



ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO
FACULTAD DE INFORMÁTICA Y ELECTRÓNICA
ESCUELA DE INGENIERÍA EN SISTEMAS

DESARROLLO DEL SISTEMA OPEN COURSE WARE ESPOCH

Trabajo de titulación presentado para optar al grado académico de:
INGENIERO EN SISTEMAS INFORMÁTICOS

AUTORES: JHONNY JAVIER GUEVARA CAMAS
CARLOS MICHAEL MEJIA GRANDA
TUTOR: ING. JORGE HUILCA PALACIOS.

Riobamba-Ecuador

2016

@2015, Jhonny Javier Guevara Camas, Carlos Michael Mejía Granda

Se autoriza la reproducción total o parcial, con fines académicos, por cualquier medio o procedimiento, incluyendo la cita bibliográfica del documento, siempre y cuando se reconozca el Derecho de Autor.

ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO
FACULTAD DE INFORMÁTICA Y ELECTRÓNICA
ESCUELA DE INGENIERÍA EN SISTEMAS

El Tribunal de Trabajo de Titulación certifica que: El trabajo de titulación: DESARROLLO DEL SISTEMA OPEN COURSE WARE ESPOCH, de responsabilidad de los señores Jhonny Javier Guevara Camas y Carlos Michael Mejía Granda, ha sido minuciosamente revisado por los Miembros del Tribunal de Trabajo de Titulación, quedando autorizada su presentación.

NOMBRE	FECHA	FIRMA
DR. MIGUEL TASAMBAY, PhD. DECANO DE LA FACULTAD DE INFORMÁTICA Y ELECTRÓNICA	_____	_____
DR. JULIO SANTILLÁN CASTILLO DIRECTOR DE LA ESCUELA DE INGENIERÍA EN SISTEMAS	_____	_____
ING. JORGE HUILCA PALACIOS DIRECTOR DE TRABAJO DE TITULACIÓN	_____	_____
DR. OMAR GÓMEZ, PhD. MIEMBRO DEL TRIBUNAL	_____	_____

“Nosotros, Jhonny Javier Guevara Camas y Carlos Michael Mejía Granda, somos responsables de las ideas, doctrinas y los resultados expuestos en este documento y el patrimonio intelectual del Trabajo de Titulación pertenece a la Escuela Superior Politécnica de Chimborazo”.

Jhonny Javier Guevara Camas

Carlos Michael Mejía Granda

AGRADECIMIENTO

Agradecemos a Dios por permitirnos culminar esta etapa profesional tan importante, brindándonos salud, alegrías, tristezas, caídas y sobre todo fuerza para seguir adelante.

Gracias a nuestro tutor: Ing. Jorge Huilca, quien nos han apoyado y guiado en el proceso final de nuestra carrera universitaria; gracias por su tiempo y su predisposición en todo momento.

Finalmente, a todas esas personas que de una forma u otra manera nos han respaldado, brindándonos todo el conocimiento necesario, siempre con una sonrisa, dejando en nosotros gratitud y grandes recuerdos.

JHONNY.

CARLOS.

DEDICATORIA

El presente trabajo está dedicado primeramente a nuestras familias, quienes en el día a día, durante toda nuestra carrera y vida personal han sabido apoyarnos; a nuestros Padres quienes con su esfuerzo y consejo nos han llevado por el camino del éxito, superando todo obstáculo e impartiendo las mejores enseñanzas de vida que hasta el día de hoy hemos seguido y seguiremos para convertirnos en personas de bien.

A nuestros compañeros de estudio quienes a lo largo de la carrera y fuera de ella nos han brindado su amistad y esfuerzo conjunto y a todos nuestros profesores y demás personas que nos han extendido su mano y que sin ellas no hubiésemos logrado alcanzar esta meta. Por y para todos ustedes entregamos este trabajo, fruto de nuestro tiempo, esfuerzo y dedicación, muchas gracias desde el fondo de nuestro corazón.

JHONNY.

CARLOS.

CONTENIDO

RESUMEN.....	xxv
SUMARY.....	xxvi
INTRODUCCION.....	01
CAPÍTULO I.....	08
1 MARCO TÉORICO	
1.1 Patrón de Diseño Modelo Vista Controlador (MVC).....	08
1.2 Framework JSF.....	10
1.2.1 Características.....	11
1.2.2 Ventajas.....	11
1.2.3 Desventajas.....	11
1.3 Java Persistence API (JPA).....	11
1.3.1 Características.....	12
1.3.2 Ventajas.....	13
1.3.3 Desventajas.....	13
1.4 PostgreSQL.....	14
1.4.1 Características.....	15
1.4.2 Ventajas.....	15
1.4.3 Desventajas.....	15
1.5 Primefaces.....	16
1.5.1 Ciclo de Vida.....	16
1.5.2 Ventajas.....	16
1.5.3 Desventajas.....	17
1.6 Diseño Web Adaptable.....	17
1.6.1 Mobile First.....	18

1.7	SOAP.....	19
1.7.1	Soa.....	19
1.7.2	Restricciones.....	20
CAPÍTULO II		17
2	MARCO METODOLÓGICO.....	21
2.1	Descripción general de la metodología.....	21
2.1.1	Personas y Roles del Proyectos.....	21
2.1.2	Tipos y roles de Usuario.....	21
2.1.3	Planificación del Proyecto.....	22
2.1.4	Preparación del Proyecto.....	23
2.1.5	Requerimientos.....	23
2.1.6	Sprints del Proyecto.....	29
2.1.7	Diagramas de caso de uso.....	31
2.1.8	Diagrama de Clases.....	43
2.1.9	Desarrollo.....	66
2.1.10	Gestión del proyecto(Burndown Chart).....	165
2.2	Instalación y configuración del sistema.....	166
CAPÍTULO III.....		186
3	MARCO DE RESULTADOS, DISCUSION Y ANÁLISIS DE RESULTADO.....	167
3.2	Validación de Usabilidad.....	168
3.2.1	Identidad.....	168
3.2.2	Contenido.....	171
3.2.3	Navegación.....	173
3.2.3	Utilidad.....	175
3.2.3	Retroalimentación.....	177
3.3	Pruebas de Aceptación.....	178

CONCLUSIONES.....	179
RECOMENDACIONES.....	180
BIBLIOGRAFÍA.....	181
ANEXOS.....	182

ÍNDICE DE TABLAS

Tabla 1.1: Características DBMS PostgreSQL.....	15
Tabla 2-1: Personas y Roles del Proyecto.....	21
Tabla 2-2: Tipos y Roles de Usuario.....	22
Tabla 2-3: Niveles de Prioridad.....	24
Tabla 2-4: Product Backlog.....	26
Tabla 2-5. Sprint Backlog.....	29
Tabla 2-6: Detalle de caso de uso SU-001.....	32
Tabla 2-7: Detalle de caso de uso SU-002.....	32
Tabla 2-8: Detalle de caso de uso SU-003.....	32
Tabla 2-9: Detalle de caso de uso SU-004.....	33
Tabla 2-10: Detalle de caso de uso A-001.....	34
Tabla 2-11: Detalle de caso de uso A-002.....	34
Tabla 2-12: Detalle de caso de uso A-003.....	35
Tabla 2-13: Detalle de caso de uso A-004.....	35
Tabla 2-14: Detalle de caso de uso A-005.....	35
Tabla 2-15: Detalle de caso de uso D-001.....	36
Tabla 2-16: Detalle de caso de uso D-002.....	37
Tabla 2-17: Detalle de caso de uso D-003.....	37
Tabla 2-18: Detalle de caso de uso D-004.....	38
Tabla 2-19: Detalle de caso de uso D-005.....	38
Tabla 2-20: Detalle de caso de uso D-006.....	38
Tabla 2-21: Detalle de caso de uso D-007.....	39
Tabla 2-22: Detalle de caso de uso V-001.....	39
Tabla 2-23: Detalle de caso de uso V-002.....	40

Tabla 2-24: Detalle de caso de uso V-003.....	40
Tabla 2-25: Detalle de caso de uso V-004.....	40
Tabla 2-26: Detalle de caso de uso V-005.....	41
Tabla 2-27: Detalle de caso de uso V-006.....	41
Tabla 2-28: Detalle de caso de uso V-007.....	42
Tabla 2-29: Detalle de Clases, Métodos y Atributos.....	44
Tabla 2-30: HT1 Arquitectura del sistema.....	66
Tabla 2-31: Actividad HT1.....	67
Tabla 2-32: TI de HT1.....	67
Tabla 2-33:PA de HT1.....	68
Tabla 2-34: Estándar de Interfaz Gráfica.....	69
Tabla 2-35: Actividad de HT2.....	70
Tabla 2-36: TI de HT2.....	70
Tabla 2-37: PA de HT2.....	70
Tabla 2-38: Descripción de componentes de Interfaz.....	71
Tabla 2-39: HT1 Diseñar base de datos.....	77
Tabla 2-40: Actividad de HT3.....	78
Tabla 2-41: TI de HT3.....	78
Tabla 2-42: PA de HT3.....	78
Tabla 2-43: Tabla t_auditoria.....	81
Tabla 2-44: Tabla t_error.....	81
Tabla 2-45: Tabla t_actividad.....	81
Tabla 2-46: Tabla t_capítulo.....	81
Tabla 2-47: Tabla t_curso.....	82
Tabla 2-48: Tabla t_examen.....	82
Tabla 2-49: Tabla t_pregunta.....	82

Tabla 2-50: Tabla t_lectura.....	83
Tabla 2-51: Tabla t_respuesta.....	83
Tabla 2-52: Tabla t_recurso.	83
Tabla 2-53: Tabla t_rol.....	83
Tabla 2-54: Tabla t_tema.....	84
Tabla 2-55: Tabla t_usuario.....	84
Tabla 2-56: Actividad de HT4	84
Tabla 2-57: TI HT4	85
Tabla 2-58: PA de HT4	85
Tabla 2-59: PA de HT4.....	85
Tabla 2-60: HU1 Manejar sesiones de los usuarios	91
Tabla 2-61: Actividades de HU1	92
Tabla 2-63: Tarea 2 de HU1.....	92
Tabla 2-64: Prueba de aceptación 1 de HU1	92
Tabla 2-65: Prueba de aceptación 2 de HU1	93
Tabla 2-66: de HU2 Ingresar, editar y listar los usuarios.....	93
Tabla 2-67: Actividades de HU2.....	94
Tabla 2-68: Tarea 1 de HU2.....	94
Tabla 2-69: Tarea 2 de la HU2.....	94
Tabla 2-70: Prueba de aceptación 1 de HU2	95
Tabla 2-71: Prueba de aceptación 2 de HU2	95
Tabla 2-72: Prueba de aceptación 3 de HU2	96
Tabla 2-73: Prueba de aceptación 4 de HU2.....	96
Tabla 2-74: H3 Visualizar Materias	97
Tabla 2-75: Actividades de H3.....	97
Tabla 2-76: Tarea 1 de la H3.....	97

Tabla 2-77: Prueba de aceptación 1 de H3.....	98
Tabla 2-78: H4 Visualizar periodos.....	99
Tabla 2-79: Actividades de H4.....	99
Tabla 2-80: Tarea 1 de la HU4.....	99
Tabla 2-81: Prueba de aceptación 1 de HU4.....	99
Tabla 2-82: HU5 Editar y Visualizar el perfil de usuarios.....	100
Tabla 2-83: Actividades de HU4.....	100
Tabla 2-84 Tarea 1 de la HU5.....	101
Tabla 2-85: Prueba de aceptación 1 de HU5.....	101
Tabla 2-86: Prueba de aceptación 2 de HU5.....	102
Tabla 2-87: HU6 Ingresar, editar y listar los cursos (Administrador).....	102
Tabla 2-88: Actividades de HU6.....	103
Tabla 2-89: Tarea 1 de la HU6.....	103
Tabla 2-90: Tarea 2 de la HU6.....	103
Tabla 2-91: Prueba de aceptación 1 de HU6.....	104
Tabla 2-92: Prueba de aceptación 2 de HU6.....	104
Tabla 2-93 Prueba de aceptación 3 de HU6.....	105
Tabla 2-94: Prueba de aceptación 4 de HU6.....	105
Tabla 2-95: HU7 Listar cursos por carrera, publicar y editar perfil cursos (Docente).....	106
Tabla 2-96: Actividades de HU7.....	106
Tabla 2-97: Tarea 1 de la HU7.....	106
Tabla 2-98: Tarea 2 de la HU7.....	107
Tabla 2-99: Prueba de aceptación 1 de HU7.....	107
Tabla 2-100: Prueba de aceptación 2 de HU7.....	108
Tabla 2-101: Prueba de aceptación 3 de HU7.....	108
Tabla 2-102: HU8 Ingresar, editar, listar y visualizar las actividades de un curso.....	109

Tabla 2-103: Actividades de HU8.....	109
Tabla 2-104: Tarea 1 de la HU8.....	109
Tabla 2-105: Tarea 2 de la HU8.....	110
Tabla 2-106: Prueba de aceptación 1 de HU8.....	110
Tabla 2-107: Prueba de aceptación 2 de HU8.....	111
Tabla 2-108: Prueba de aceptación 3 de HU8.....	111
Tabla 2-109: Prueba de aceptación 4 de HU8.....	112
Tabla 2-110: Funcionalidad Módulo Docente.....	112
Tabla 2-111: Actividades de HU9.....	113
Tabla 2-112: Tarea 1 de la HU9.....	113
Tabla 2-113: Tarea 2 de la HU9.....	114
Tabla 2-114: Prueba de aceptación 1 de HU9.....	114
Tabla 2-115: Prueba de aceptación 2 de HU9.....	114
Tabla 2-116: Prueba de aceptación 3 de HU9.....	115
Tabla 2-117: Prueba de aceptación 4 de HU9.....	115
Tabla 2-118: HU10 Gestión Lecturas Docente.....	116
Tabla 2-119: Actividades de HU10.....	116
Tabla 2-120: Tarea 1 de la HU10.....	117
Tabla 2-121: Tarea 1 de la HU10.....	117
Tabla 2-122: Prueba de aceptación 1 de HU10.....	117
Tabla 2-123: Prueba de aceptación 2 de HU10.....	118
Tabla 2-124: Prueba de aceptación 3 de HU10.....	118
Tabla 2-125: Prueba de aceptación 4 de HU10.....	119
Tabla 2-126: HU11 Gestionar Recursos Docente.....	120
Tabla 2-127: HU11 Actividades de HU11.....	120
Tabla 2-128: Tarea 1 de la HU11.....	120

Tabla 2-129: Tarea 2 de la HU11.....	121
Tabla 2-130: Prueba de Aceptación 1 de la HU11.....	121
Tabla 2-131: Prueba de Aceptación 2 de la HU11.....	122
Tabla 2-132: Prueba de Aceptación 3 de la HU11.....	122
Tabla 2-133: Prueba de Aceptación 4 de la HU11.....	123
Tabla 2-134: HU12 Gestionar Temas-Curso Docente.....	123
Tabla 2-135: Actividades de HU12.....	124
Tabla 2-136: Tarea 1 de la HU12.....	124
Tabla 2-137 Tarea 2 de la HU12.....	124
Tabla 2-138: Prueba de aceptación 1 de HU12.....	125
Tabla 2-139: Prueba de aceptación 2 de HU12.....	125
Tabla 2-140: Prueba de aceptación 3 de HU12.....	126
Tabla 2-141: Prueba de aceptación 4 de HU12.....	126
Tabla 2-142: HU13 Gestionar Cursos Visitante.....	127
Tabla 2-143: Actividades de HU13.....	127
Tabla 2-144: Tarea 1 de la HU13.....	127
Tabla 2-145: Prueba de aceptación 1 de HU13.....	128
Tabla 2-146: Prueba de aceptación 2 de HU13.....	128
Tabla 2-147: Prueba de aceptación 3 de HU13.....	129
Tabla 2-148: HU14 Galería de Imágenes.....	129
Tabla 2-149: Actividades de HU14.....	130
Tabla 2-150: Tarea 1 de la HU14.....	130
Tabla 2-151: Prueba de aceptación 1 de HU14.....	130
Tabla 2-152: HU15 Gestionar Cursos Visitante.....	131
Tabla 2-153: Actividades de HU15.....	131
Tabla 2-154: Tarea 1 de la HU15.....	131

Tabla 2-155: Prueba de aceptación 1 de HU15.....	132
Tabla 2-156: HU16 Búsqueda Cursos.....	132
Tabla 2-157: Actividades de HU16.....	133
Tabla 2-158: Tarea 1 de la HU16.....	133
Tabla 2-159: Prueba de aceptación 2 de HU16.....	134
Tabla 2-160: Prueba de aceptación 2 de HU16.....	134
Tabla 2-161: Prueba de aceptación 3 de HU16.....	135
Tabla 2-162: HU17 Detalle Curso Visitante.....	135
Tabla 2-163: Actividades de HU17.....	136
Tabla 2-164: Tarea 1 de la HU17.....	136
Tabla 2-165: Prueba de aceptación 1 de HU17.....	136
Tabla 2-166: Prueba de aceptación 2 de HU17.....	137
Tabla 2-167: Prueba de aceptación 3 de HU17.....	137
Tabla 2-168: HU18 Integrar los servicios web de la Facultad.....	138
Tabla 2-169: Actividades de HU18.....	138
Tabla 2-170: Tarea 1 de la HU18.....	138
Tabla 2-171: Prueba de aceptación 1 de HU18.....	139
Tabla 2-172: HU19 Integrar los servicios web de la Escuela.....	139
Tabla 2-174: Tarea 1 de la HU19.....	140
Tabla 2-175: Prueba de aceptación 1 de HU19.....	140
Tabla 2-176: HU20 Integrar los servicios web del período.....	140
Tabla 2-177: Actividades de HU20.....	141
Tabla 2-178: Tarea 1 de la HU20.....	141
Tabla 2-179: Prueba de aceptación 1 de HU20.....	141
Tabla 2-180: HU21 Integrar los servicios web del docente.....	142
Tabla 2-181: HU21 Actividades de HU21.....	142

Tabla 2-182: Tarea 1 de la HU21.....	142
Tabla 2-183: Prueba de aceptación 1 de HU21.....	143
Tabla 2-184: HU22 Gestionar Preguntas-Examen.....	143
Tabla 2-185: Actividades de HU22.....	144
Tabla 2-186: Tarea 1 de la HU22.....	144
Tabla 2-187: Tarea 2 de la HU22.....	144
Tabla 2-188: Prueba de aceptación 1 de HU22.....	145
Tabla 2-189: Prueba de aceptación 2 de HU22.....	145
Tabla 2-190: Prueba de aceptación 3 de HU22.....	146
Tabla 2-191: Prueba de aceptación 4 de HU22.....	146
Tabla 2-192: HU23 Gestionar Examen Docente.....	147
Tabla 2-193: Actividades de HU23.....	148
Tabla 2-194: Tarea 1 de la HU23.....	148
Tabla 2-195: Prueba de aceptación 1 de HU23.....	148
Tabla 2-196: Prueba de aceptación 2 de HU23.....	149
Tabla 2-197: Prueba de aceptación 3 de HU23.....	149
Tabla 2-198: Prueba de aceptación 4 de HU23.....	150
Tabla 2-199: HU24 Ingreso Auditoria.....	150
Tabla 2-200: Actividades de HU24.....	151
Tabla 2-201: Tarea 1 de la HU24.....	151
Tabla 2-202: Prueba de aceptación 1 de HU24.....	151
Tabla 2-203: Prueba de aceptación 2 de HU24.....	152
Tabla 2-204: HU25 Ingresar el registro de error.....	152
Tabla 2-205: Actividades de HU25.....	153
Tabla 2-206: Tarea 1 de la HU25.....	153
Tabla 2-207: Prueba de aceptación 1 de HU25.....	153

Tabla 2-208: Prueba de aceptación 2 de HU25.....	154
Tabla 2-209: HT5 Preparación Servidor.....	155
Tabla 2-210: Actividades de HT5.....	155
Tabla 2-211: Tarea 1 de la HT5.....	155
Tabla 2-212: Prueba de aceptación 1 de HT5.....	156
Tabla 2-213: Prueba de aceptación 2 de HT5.....	156
Tabla 2-214: Prueba de aceptación 3 de HT5.....	157
Tabla 2-215: HT5 Despliegue App.....	157
Tabla 2-216: Actividades de HT6.....	157
Tabla 2-217: Tarea 1 de la HT6.....	158
Tabla 2-218: Prueba de aceptación 1 de HT6.....	158
Tabla 2-219: Prueba de aceptación 2 de HT6.....	159
Tabla 2-220: HT7 Pruebas App.....	159
Tabla 2-221: Actividades de HT7.....	160
Tabla 2-222: Tarea 1 de la HT7.....	160
Tabla 2-223: Prueba de aceptación 1 de HT7.....	160
Tabla 2-224: HT8 Capacitación Usuarios.....	161
Tabla 2-225: Actividades de HT8.....	161
Tabla 2-226: Tarea 1 de la HT8.....	161
Tabla 2-227: Prueba de aceptación 1 de HT8.....	162
Tabla 2-228: HT9 Mejoras al Sistema.....	162
Tabla 2-229: Actividades de HT9.....	163
Tabla 2-230: Tarea 1 de la HT9.....	163
Tabla 2-231: Prueba de aceptación 1 de HT9.....	163
Tabla 2-232: HT1 10 Documentación.....	164
Tabla 2-233: Actividades de HT10.....	164

Tabla 2-234: Tarea 1 de la HT10.....	164
Tabla 2-235: Prueba de aceptación 1 de HT10.....	164
Tabla 3.1: Estadísticas Utilización de aulas Virtuales Marzo-agosto 2015.....	186
Tabla 3.2: Estadísticas Utilización de aulas Virtuales Octubre-febrero 2015.....	186
Tabla 3-3: Preguntas Test Usabilidad-Identidad.....	187
Tabla 3-4: Usabilidad Identidad-Pregunta 1.....	188
Tabla 3-5: Usabilidad Identidad-Pregunta 2.....	188
Tabla 3-6: Usabilidad Identidad-Pregunta 3.....	189
Tabla 3-7: Usabilidad Identidad-Pregunta 4.....	189
Tabla 3.8: Usabilidad Identidad-Pregunta 5.....	189
Tabla 3.9: Usabilidad Identidad-Pregunta 6.....	190
Tabla 3-10: Preguntas Test Usabilidad-Contenido.....	190
Tabla 3-11: Test Usabilidad-Contenido Pregunta 1.....	191
Tabla 3-12: Test Usabilidad-Contenido Pregunta 2.....	191
Tabla 3-13.: Test Usabilidad-Contenido Pregunta 3.....	191
Tabla 3-14: Test Usabilidad-Contenido Pregunta.....	192
Tabla 3.15: Test Usabilidad-Contenido Pregunta 5.....	192
Tabla 3.16: Preguntas Test Usabilidad-Navegación.....	193
Tabla 3.17: Test Usabilidad-Navegación Pregunta 1.....	193
Tabla 3.18: Test Usabilidad-Navegación Pregunta 2.....	194
Tabla 3.19: Test Usabilidad-Navegación Pregunta 3.....	194
Tabla 3.20: Test Usabilidad-Navegación Pregunta 4.....	195
Tabla 3.21: Preguntas Test Usabilidad-Utilidad.....	195
Tabla 3.22: Test Usabilidad-Utilidad Pregunta 1.....	196
Tabla 3.23: Test Usabilidad-Utilidad Pregunta 2.....	196
Tabla 3.24: Test Usabilidad-Utilidad Pregunta 3.....	197

Tabla 3-25: Preguntas Test Usabilidad-Retroalimentación.....	197
Tabla 3-26: Resumen de Datos Test Usabilidad.....	198
Tabla 3-27: Total de Historias de Usuario Realizadas	199

ÍNDICE DE FIGURAS

Figura 1-1: OCW UTPL.....	02
Figura 1-2: OCW ESPOL	03
Figura 1-3: OCW EPN.....	04
Figura 1-4: Patrón MVC.....	09
Figura 1-5: Framework JSF.....	10
Figura 1-6: Nivel de Clases Arquitectura JPA.....	12
Figura 1-7: Arquitectura PostgreSQL.....	14
Figura 1.8: Características DBMS PostgreSQL.....	16
Figura 1-9. Usuario Móvil vs Usuario de Escritorio.....	18
Figura 1-10. Mobile First.....	19
Figura 1-11. Interacción a través de Servicios Web.....	20
Figura 2-1: Diagrama de Clases SuperUsuario.....	31
Figura 2-2: Casos de uso Administrador.....	33
Figura 2-3. Casos de uso Docente.....	36
Figura 2-4: Casos de uso Visitante.....	39
Figura 2-5: Diagrama de Clases.....	43
Figura 2-6: Arquitectura del Sistema.	68
Figura 2-7: Diagrama de despliegue.....	69
Figura 2-8: Página de Inicio	71
Figura 2-9: Página de Inicio Docente.....	72
Figura 2-10: Página de Inicio Administrador.....	73
Figura 2-11: Página de inicio.	73
Figura 2-12: Página de Login.	74
Figura 2-13: Menú Búsqueda de Cursos.....	74

Figura 2-14: Búsqueda de Cursos por Nivel.....	75
Figura 2-15: Búsqueda de Cursos por Nombre.....	75
Figura 2-16: Página de Curso.....	76
Figura 2-17: Listado de Instructores.....	76
Figura 2-18: Pagina Docente y sus Cursos.....	77
Figura 2-19: DER.....	79
Figura 2-20: Diagrama Físico.....	80
Figura 2-21: Estándar Tablas.....	86
Figura 2-22: Estándar Atributos.....	87
Figura 2-23: Estándar Id Secuencial.....	87
Figura 2-24: Estándar Restricción Unique.....	88
Figura 2-25: Estándar Restricciones llave principal.....	88
Figura 2-26: Estándar Restricción FK.....	88
Figura 2-27: Estándar codificación Clases.....	89
Figura 2-28: Estándar codificación Controladores.....	89
Figura 2-29: Estándar codificación Atributos Clase.....	90
Figura 2-30: Estándar codificación Métodos Controlador.....	90
Figura 2-31: Estándar codificación Atributos Clases.....	91
Figura 2-32: Burn Down Chart.....	166
Figura 2-33. Terminal Linux.....	167
Figura 2-34. Instalar Java.....	167
Figura 2-35. Progreso Instalación Java.....	168
Figura 2-36. Progreso Instalación Java.....	168
Figura 2-37. Verificar Instalación Java.....	169
Figura 2-38. Instalar Postgresql.....	169
Figura 2-39. Cambio de contraseña	169

Figura 2-40. Cambio de contraseña.....	170
Figura 2-41. Búsqueda de Padmin.....	170
Figura 2-42. Nueva Conexión	170
Figura 2-43. Configuración de Conexión	171
Figura 2-44. Crear Base de Datos.....	172
Figura 2-45. Restauración de la base de datos	172
Figura 2-46. Restauración de la base de datos.....	173
Figura 2-47. Restauración de la base de datos.....	173
Figura 2-48. Asignación de usuario para Glassfish.....	174
Figura 2-49. Permisos de Usuario.....	174
Figura 2-50. Credenciales de Glassfish.....	174
Figura 2-50. Descargar NetBeans.....	174
Figura 2-51. Instalación NetBeans.....	175
Figura 2-52. Paquetes de Netbeans.....	175
Figura 2-53. Términos de licencia.....	176
Figura 2-54. Instalar.....	176
Figura 2-55. Preparar entorno.....	177
Figura 2-56. Copiar Directorio a la raíz	177
Figura 2-57. Abrir proyecto desde NetBeans.....	177
Figura 2-58. Abrir proyecto e ir a sus propiedades.....	178
Figura 2-59. Remover librerías.....	178
Figura 2-60. Añadir librerías	179
Figura 2-61. Seleccionar y configurar librerías.....	179
Figura 2-62. Configuración servidor.....	180
Figura 2-63. Añadir Servidor.....	180
Figura 2-64. Seleccionar Servidor.....	181

Figura 2-65. Seleccionar Ruta.....	181
Figura 2-66. Siguiente paso.....	182
Figura 2-67. Ingresar Contraseña y finalizar.....	182
Figura 2-68. Seleccionar y Aceptar.....	183
Figura 2-69. Resolver problemas de servidor.....	183
Figura 2-70. Agregar conexión.....	184
Figura 2-71. Seleccionar y Aceptar.....	184
Figura 2-72. Ejecutar la aplicación.	185
Figura 2-73. Vista principal	185
Figura 3-1: Utilización de aulas Virtuales Marzo-agosto 2015.....	186
Figura 3-2: Utilización de aulas Virtuales Octubre-febrero 2015.....	187
Figura 3-3: Pregunta 1 Usabilidad Identidad.....	188
Figura 3-4: Pregunta 2 Usabilidad-Identidad.....	188
Figura 3-5: Pregunta 3 Usabilidad-Identidad.....	189
Figura 3-6: Pregunta 4 Usabilidad-Identidad.....	189
Figura 3-7: Pregunta 5 Usabilidad-Identidad.....	190
Figura 3-8: Pregunta 6 Usabilidad-Identidad.....	190
Figura 3-9: Pregunta 1 Usabilidad-Contenido.....	191
Figura 3-10: Pregunta 2 Usabilidad-Contenido.....	191
Figura 3-11: Pregunta 3 Usabilidad-Contenido.....	192
Figura 3-12: Pregunta 4 Usabilidad-Contenido.....	192
Figura 3-13: Pregunta 5 Usabilidad-Contenido.....	193
Figura 3-14: Pregunta 1 Usabilidad-Navegación.....	194
Figura 3-15: Pregunta 2 Usabilidad-Navegación.....	194
Figura 3-16: Pregunta 3 Usabilidad-Navegación.....	195
Figura 3-17: Pregunta 4 Usabilidad-Navegación.....	195

Figura 3-18: Pregunta 1 Usabilidad-Utilidad.....	196
Figura 3-19: Pregunta 2 Usabilidad-Utilidad.....	196
Figura 3-20: Pregunta 3 Usabilidad-Utilidad.....	197
Figura 3-21: Pregunta 3 Usabilidad-Retroalimentación.....	197
Figura 3-22: Resumen del Test Completo de Usabilidad.	198
Figura 3-23: Historias de Usuario.....	199

RESUMEN

Se desarrolló el Sistema Informático Open Course Ware para la gestión y publicación del material digital que generan los docentes de la Escuela Superior Politécnica de Chimborazo. Se tomó en cuenta realizar un estudio y análisis de la situación en cuanto a medios virtuales con los que contaban los docentes para publicar dichos materiales en abierto y su porcentaje de uso en los dos últimos periodos académicos. Mediante el uso de herramientas y tecnologías de uso gratuito como Java, Glassfish, NetBeans, SOAP, PostgreSQL, Primefaces y la metodología de desarrollo ágil SCRUM que permite un marco de trabajo interactivo, flexible ante el cambio de requerimientos y prioridades se logró desarrollar el sistema informático OCW ESPOCH y su integración con el sistema académico OASIS cuyas fases de análisis, capacitación y estudio de resultados se lo realizó en la institución en un lapso de 4 meses cumpliendo con el tiempo estimado. Con la implementación del sistema informático se proveyó a la institución de una plataforma web para la gestión y divulgación de los materiales digitales en abierto generados por los docentes de la institución, los mismos que fueron capacitados en el manejo del sistema y posteriormente lo aceptaron en un porcentaje de 98% obtenido de un test de usabilidad que evaluó métricas de identidad, contenido, navegabilidad y utilidad. Cabe resaltar que los materiales digitales que están expuestos en la plataforma web son difundidos bajo la licencia Creative Commons. Además, se recomienda a la ESPOCH la adquisición de servidores con mayor capacidad de almacenamiento ya que el material digital generado por los docentes aumenta con rapidez.

Palabras claves: <SISTEMA INFORMÁTICO [OCW ESPOCH]>, SISTEMA DE CALIDAD>, <METODOLOGÍA DE DESARROLLO ÁGIL [SCRUM]>, < SISTEMA DE GESTIÓN DE CONTENIDOS>, LICENCIA <CREATIVE COMMONS>, <SOFTWARE LIBRE>

SUMMARY

It was developed the Open Course Ware computer system for the management and publication of digital material that generate the teachers of Polytechnic School of Chimborazo. It took into account carry out a study an analysis of the situation on virtual media that teachers had to publish materials in open and the percentage of use in two recent academic periods. Thought the use of tools and technology for free use as java, Glassfish, NetBeans, Soap, PostgreSQL, Primefaces and methodology of agile development, SCRUM, which allows an interactive, flexible framework to changing requirements and priorities are managed to develop the computer system OCW ESPOCH and its integration with the academic system OASIS whose phases of analysis, training and study results was it conducted in the computer system in endowed to the institution of a web platform for the management and dissemination of digital materials in open, generated by the teacher of the institution, the same people who were trained in the management of the system and subsequently validated by 98% obtaining a test of usability evaluating identity, content metrics, navigability and usefulness. It is concluded that are exposed on the Polytechnic School of Chimborazo acquisition of servers with greater storage capacity.

Keywords: <COMPUTER SYSTEM [OCW ESPOCH]>, <SYSTEM OF QUALITY>, <AGILE DEVELOPMENT METHODOLOGY [SCRUM]>, <CONTENT MANAGEMENT SYSTEM>, <LICENSE [CREATIVE COMMONS]>, <FREE SOFTWARE>.

INTRODUCCIÓN

ANTECEDENTES

Como resultado del avance tecnológico se han venido innovando las formas de acceder, construir y compartir el conocimiento; así como la dinámica de interacción de la sociedad, según el sitio web de la UNESCO (Paul Resta, Alexey Semenov. (2004)). Las tecnologías de la información y la comunicación en la formación docente. 09/07/2015, de unesco.org Sitio web: <http://unesdoc.unesco.org/images/0012/001295/129533s.pdf>) es un hecho que mediante la utilización de las TICS se contribuye al proceso de aprendizaje de manera significativa aportando estrategias didácticas e innovadoras que giran en torno a un elemento principal que es el material digital.

Evidentemente en este tipo de estrategias se centra la pedagogía de la Educación Superior que desarrolla la capacidad transformadora profesional, en el cual el docente deja de ser una fuente de conocimiento y se convierte en una guía para los estudiantes facilitando recursos y herramientas que promuevan su entendimiento.

OpenCourseWare es una iniciativa editorial electrónica a gran escala, puesta en marcha en abril del 2001 por el Instituto Tecnológico de Massachusetts (MIT) en colaboración con la Fundación William and Flora Hewlett y la Fundación Andrew W. Mellon.

Forma parte de un movimiento global que promueve el acceso al conocimiento de forma libre y sin restricciones. A través de los OCW, los docentes de enseñanza superior ofrecen en abierto los materiales docentes que utilizan habitualmente en su enseñanza. Su objetivo fundamental es promover y desarrollar la educación compartiendo los recursos docentes con: otros educadores que los pueden reutilizar para sus enseñanzas, estudiantes, graduados y en general quien tenga a bien usarlos para mejorar su conocimiento personal.

Ecuador también se ha unido al movimiento OCW, actualmente existen 3 instituciones que han creado este tipo de espacios para la difusión libre del conocimiento.

UTPL

El sitio de la Universidad Técnica Particular de Loja fue el primer OCW del país, creado con EduCommons y publicado en el 2008, recientemente se incorporó a la red de Universia y se está trabajando en incorporarse al OCW Consortium.

Actualmente contiene publicaciones de las facultades de:

- Economía.
- Educación Continua.
- Ingeniería Civil.
- Pedagogía.
- Sistemas Informáticos.



Figura 1-1. OCW UTPL

Fuente: <http://ocw.utpl.edu.ec>

ESPOL

El sitio Web de la Escuela Politécnica del Litoral fue creado con EduCommons publicado en el 2009; actualmente contiene materiales de las facultades de:

- Ciencias Físicas.
- Ciencias Matemáticas.

- Ciencias Químicas y Ambientales.
- Ingeniería Eléctrica y de Computación.
- Ingeniería en Mecánica y Ciencias de la Producción.
- Ingeniería Marítima y de Ciencias del Mar.



Figura 1-2. OCW ESPOL

Fuente: <http://ocw.espoll.edu.ec>

EPN

El sitio Web de la Escuela Politécnica Nacional fue creado con EduCommons publicado en el 2011. Actualmente contiene materiales de las facultades de:

- Ciencias e Ingeniería en Sistemas.



Figura 1-3. OCW EPN

Fuente: <http://ocw.epn.edu.ec>

La Escuela Superior Politécnica de Chimborazo es una Institución pública que fundamenta su acción en los principios de: autonomía, democracia, cogobierno, libertad de cátedra e inviolabilidad de sus predios. Estimula el respeto de los valores inherentes de la persona, que garantiza la libertad de pensamiento, expresión, culto, igualdad, pluralismo, tolerancia, espíritu crítico y cumplimiento de las Leyes y normas vigentes.

La ESPOCH actualmente cuenta con un Entorno Virtual de Aprendizaje llamado elearning (Moodle) al cual solo los estudiantes de la ESPOCH pueden acceder para visualizar materiales según las asignaturas en las que se halle matriculado. Mientras que si existiera un sistema OpenCourseWare promoverá de manera libre con licencia creative commons el acceso y utilización del material digital de los cursos que son impartidos en la institución, imponiéndose de esta manera al uso de un e-mooc ya que la información se alojará en los servidores locales de la Espoch y el Código fuente será propiedad de la institución.

PLANTEAMIENTO DEL PROBLEMA

La ESPOCH teniendo como visión “Ser la institución líder de docencia con investigación, que garantice la formación profesional, la generación de ciencia y tecnología para el desarrollo humano integral, con reconocimiento nacional e internacional” no cuenta con una herramienta informática que permita transmitir y difundir el conocimiento generado por los docentes de la ESPOCH de manera libre y sin restricciones que pese a contar con un EVA(Entorno Virtual de Aprendizaje) este está enfocado a satisfacer otras necesidades importantes de un pequeño número de estudiantes según se hallen matriculados en un curso, imposibilitado el acceso a estudiantes, graduados, docentes de otras universidades y comprometiendo de esta manera su contribución en cursos y programas de vinculación con la colectividad (superior, 2013) ART.17.

Esto se debe a: falta de capacitación de los docentes en el uso de las TICS, falta de recursos para la adquisición de un sitio EMOC entre otros.

FORMULACIÓN DEL PROBLEMA

¿Qué beneficios aporta un OCW al docente en la gestión de un curso en la Escuela Superior Politécnica de Chimborazo?

Los docentes de la ESPOCH actualmente generan material para su clase que no es compartido y solo puede ser visualizando en aulas virtuales de carácter privado

JUSTIFICACIÓN TEÓRICA

Las aulas virtuales permiten la interacción del docente y el educando desde cualquier parte del hemisferio, es así que tenemos como ejemplo moodle.mayuqi.net, www.edoome.com, Com8s, y otras, pero al momento de ser utilizadas por los docentes de la Escuela Superior Politécnica de Chimborazo presentan como inconvenientes: el ser de carácter privado y no accesible a todos los usuarios.

En la actualidad existen importantes y ambiciosos proyectos de software que buscan sanear este inconveniente, es así que podemos nombrar: el OCW del (MIT), el OCW de la Universidad de Cádiz, el OCW de la Universidad de Huelva, etc

En la Escuela Superior Politécnica de Chimborazo no existe una plataforma sobre la cual se desplieguen los cursos que esta oferta con respecto a la malla curricular y plan de carreras y/o materias que son dictadas por el docente, por lo cual es necesaria la implementación del OCW ESPOCH a fin de socializar y dar libre acceso al conocimiento.

JUSTIFICACIÓN APLICATIVA

Con la finalidad de solucionar la problemática antes expuesta se desarrollará el sistema Open Course Ware ESPOCH haciendo uso de las tecnologías y herramientas tales como JAVA que es un lenguaje de programación de propósito general, concurrente, orientado a objetos, multiplataforma.

Para el almacenamiento y administración de datos del sistema se utilizará PostgreSQL que es un servidor de base de datos avanzado.

Para lograr un óptimo rendimiento del sistema en cuanto a transacciones con el servidor de base de datos se utilizará JAVA Persistence API (JPA) para el mapeo de las clases de la base de datos. Para el diseño de las interfaces de usuario se utilizará el framework PRIMEFACES y Java Server Faces de código abierto que cuenta con un conjunto de componentes enriquecidos que facilitan la creación de interfaces profesionales disminuyendo el tiempo de diseño y desarrollo de estas.

A continuación, se detallan los módulos que poseerá el sistema Open Course Ware ESPOCH que será responsivo.

Módulo Usuario permitirá:

- Ver cursos más vistos
- Ver cursos publicados
- Ver cursos nuevos
- Ver un curso en detalle
- Acceder al material de un curso
- Búsqueda de cursos por facultad, escuela y carrera.
- Ver un docente a detalle
- Búsqueda de docentes por nombre

Módulo Docente permitirá:

- Administrar el material de un curso
- Gestionar Capítulo
- Gestionar Tema
- Gestionar Actividad
- Gestionar Recurso
- Gestionar evaluación
- Autenticación

Módulo Administrador permitirá:

- Gestionar curso
- Gestionar docente
- Gestionar usuarios
- Gestionar permisos
- Autenticación
- Informes de Auditorias

OBJETIVOS

Objetivo General

Desarrollar el Sistema Open Course Ware para la Escuela Superior Politécnica de Chimborazo

Objetivos Específicos

- Analizar la situación actual en cuanto al material digital que manejan los docentes en el Entorno Virtual de Aprendizaje de la ESPOCH.
- Analizar los requerimientos del OCW ESPOCH
- Desarrollar el sistema Open Course Ware ESPOCH utilizando JSF PRIMEFACES y POSTGRESQL.
- Implementar el OCW ESPOCH para los docentes de la institución

CAPÍTULO I

1 MARCO TEÓRICO.

1.1 Patrón de Diseño Modelo Vista Controlador (MVC)

El patrón de arquitectura MVC (Modelo Vista Controlador) es uno de los patrones más conocidos, utilizados y difundidos en el ámbito de la Arquitectura de Software. MVC define la organización independiente del Modelo (Objetos de Negocio), la Vista (interfaz con el usuario u otro sistema) y el Controlador (controlador del flujo de trabajo de la aplicación).

Analizado lo anterior podemos dividimos el sistema en tres áreas o capas cómo se las llama, estas nos sirven para organizar los módulos de trabajo.

Este patrón de arquitectura “modelo vista controlador”, es una filosofía de diseño y que define el trabajo para la creación de aplicaciones robustas, este se halla compuesto por:

- Modelo

Es la capa raíz de la aplicación específicamente para gestionar todo lo que se refiere a datos del sistema, por ende aquí es donde se define los mecanismos de gestión de datos, esta se encarga de realizar transacciones con la base de datos y también acceder a datos.

- Contiene el núcleo de la funcionalidad (dominio) de la aplicación.
- Encapsula el estado de la aplicación.
- No sabe nada / independiente del Controlador y la Vista.

- Vista

La vista viene a ser la interfaz que mantiene comunicado al usuario del estado de sistema y ésta a la vez para cumplir su cometido mantiene comunicación directa con el controlador para hacer las peticiones necesarias de datos o cambios de estado en la vista.

- Encargada de presentar y recibir datos del usuario valiéndose del controlador.
 - Mantiene informado al usuario acerca del estado de los procesos que se hallen ejecutando el sistema tales como procesos con un fin exitoso o fallido.
 - Es la encargada de desatar múltiples operaciones hacia el controlador con un evento del usuario.
- Controlador

Este tiene interacción directa con el Usuario/Cliente es el encargado de responder a la petición del Cliente con la acción apropiada y los datos adecuados, según sea el caso si le es necesario datos adicionales a los que el usuario se halla visualizando deberá valerse del modelo haciéndole una petición de datos, tras recibir desde el modelo los datos peticionados si es necesario los refinara, formateara, realizara la lógica necesaria y finalmente devolverá hacia la Vista que desencadeno todo el proceso.

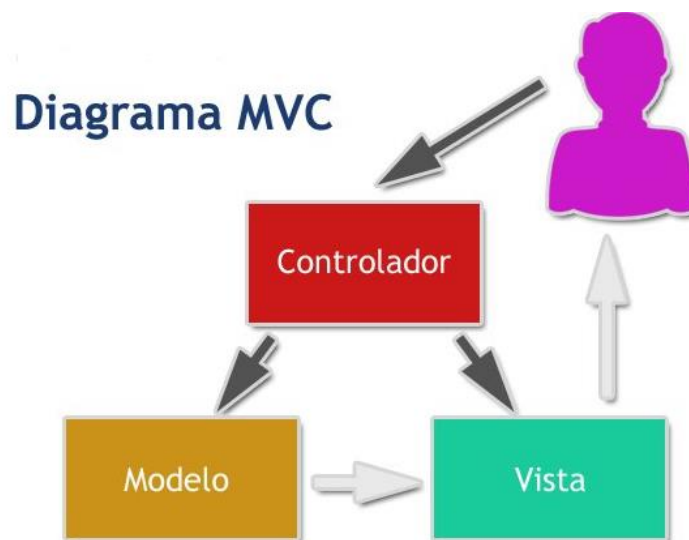


Figura 1-4. Patrón MVC

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Al final, la vista es quien lleva la responsabilidad de establecer la comunicación entre los elementos de nuestro patrón MVC.

Cuando la vista recibe un mensaje que concierne al modelo o al controlador, lo deja registrado como el modelo con el cual se comunicara y apunta con la variable controller al controlador asignado, enviándole al mismo su identificación para que el controlador establezca en su variable view el identificador de la vista y así puedan operar conjuntamente.

El responsable de deshacer estas conexiones, seguirá siendo la vista, quitándose a sí misma como dependiente del modelo y liberando al controlador.

1.2 Framework JSF

JSF es un framework para el desarrollo de aplicaciones web, está orientado a facilitar el diseño de las interfaces de usuario, liberado en marzo del 2004 por Sun Microsystems se halla basado en el patrón de diseño: Modelo Vista Controlador (MVC). (Almirón, 2012)

Una Vista en JSF está compuesta de varios archivos JSP, así como Facelets y distintos PDLs (Page Declaration Languages) los cuales son los encargados de estructurar el árbol jerárquico de componentes de JSF y vincularlos con los Managed Beans.

En cuanto al Modelo se utiliza Managed Beans que son clases de las cuales se instancia objetos responsables de administrar la lógica de la aplicación y responden en función de los eventos que generen los componentes JSF en la capa de presentación. (Lerma, 2015)

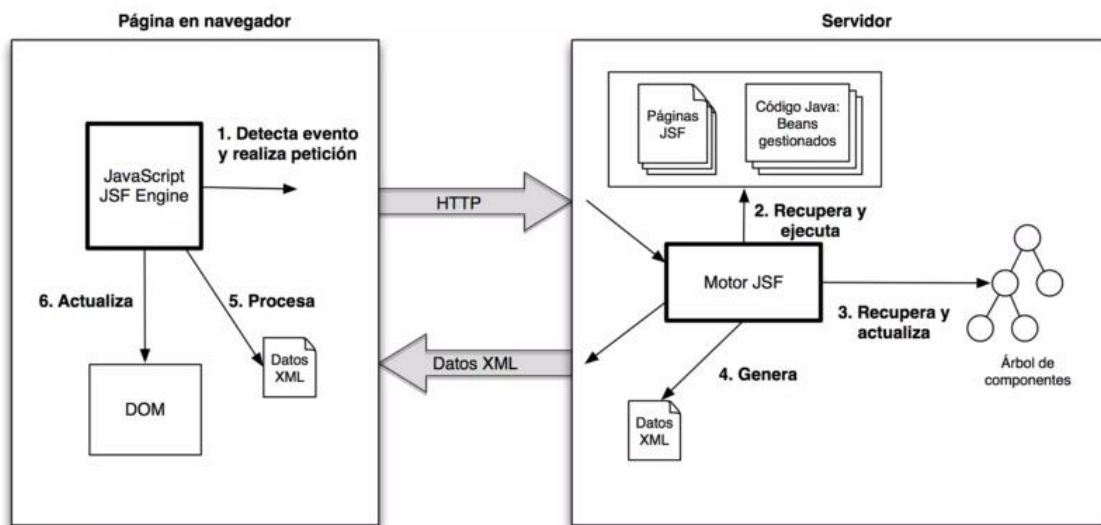


Figura 1-5. Framework JSF

Fuente: <http://www.jtech.ua.es/j2ee/publico/jsf-2012-13/sesion01-apuntes.html>

1.2.1 Características:

- Acceso y manejo de datos del usuario de manera ágil.
- Utiliza páginas JSP para generar las vistas, incluye una biblioteca de etiquetas propia para crear los elementos de los formularios HTML.
- Asocia a cada vista con formularios un conjunto de objetos java manejados por el controlador (Managed Beans) que facilitan la recogida, manipulación y visualización de los valores mostrados en los diferentes elementos de los formularios.
- Utiliza un sencillo fichero de configuración para el controlador en formato XML

1.2.2 Ventajas:

- El código JSF con el que creamos las vistas (etiquetas jsp) es muy parecido al HTML estándar. Lo pueden utilizar fácilmente desarrolladores y diseñadores web.
- JSF se integra dentro de la página JSP y se encarga de la recogida y generación de los valores de los elementos de la página
- JSF resuelve validaciones, conversiones, mensajes de error e internacionalización.
- JSF permite introducir JavaScript en la página, para acelerar la respuesta de la interfaz en el cliente (navegador del usuario).

1.2.3 Desventajas:

- El motor de servlet debe ser compatible con la especificación 2.3 y las JSP deben ser acorde a la especificación 1.2.
- JSF únicamente soporta peticiones realizadas con POST.
- La especificación no obliga a que haya validaciones en el cliente, si bien dos desarrollos como MyFaces y Shale proporcionan esta posibilidad.

1.3 Java Persistence API (JPA)

Java Persistence API (JPA) proporciona un modelo de persistencia basado en POJO's para mapear bases de datos relacionales en Java.

El Java Persistence API fue desarrollado por el grupo de expertos de EJB 3.0 como parte de JSR 220, aunque su uso no se limita a los componentes software EJB.

También puede utilizarse directamente en aplicaciones web y aplicaciones clientes; incluso fuera de la plataforma Java EE, por ejemplo, en aplicaciones Java SE. (juntadeandalucia, 2014)

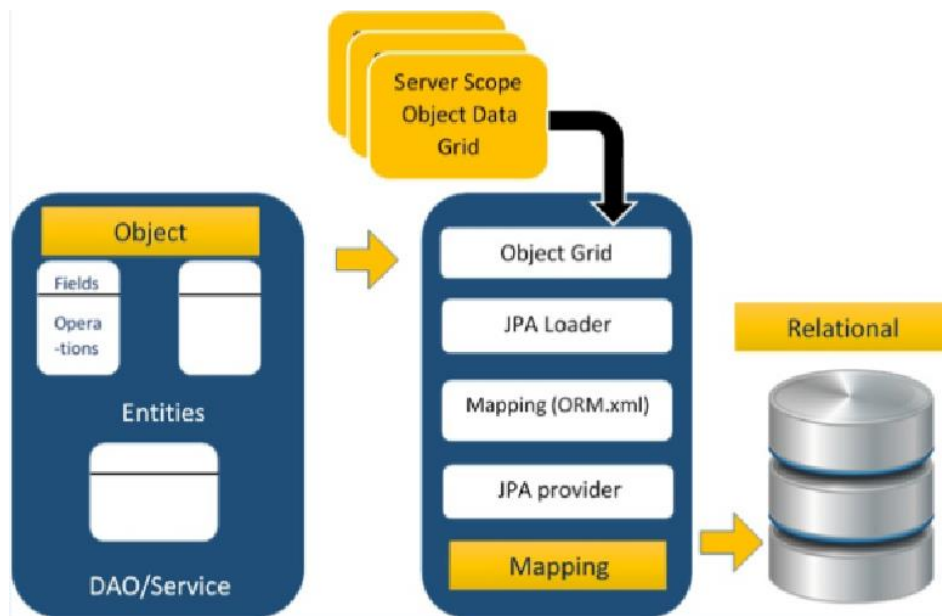


Figura 1-6. Nivel de Clases Arquitectura JPA

Fuente: http://www.tutorialspoint.com/es/jpa/jpa_architecture.htm

1.3.1 Características

El mapeo objeto/relacional, es decir, la relación entre entidades Java y tablas de la base de datos, se realiza mediante anotaciones en las propias clases de entidad, por lo que no se requieren ficheros descriptores XML.

También pueden definirse transacciones como anotaciones JPA.

- La API en sí misma, definida en el paquete `javax.persistence`
- El lenguaje de consulta Java Persistence Query Language (JPQL).
- Metadatos objeto/relacional.

1.3.2 Ventajas

Utilizar un framework de ORM simplifica enormemente la programación de la lógica de persistencia. Se trata de una idea completamente madura que cada vez se vuelve más popular. (Parr., 2012)

- Para obtener el mejor rendimiento de JPA, es importante entender los parámetros de tuning incluidos en la implementación de la JPA. La configuración de la caché es un parámetro importante de tuning (Caules., 2014)
- Si se utiliza JPA fuera del contenedor EJB, la caché y el pool de conexiones también juegan un papel importante en el rendimiento, aunque en este caso se configura el pool en el fichero persistence.xml.
- Hay otros parámetros que pueden impactar en el rendimiento de JPA. Uno es el statement cache. Éste se debe especificar explícitamente para algunas bases de datos como Oracle.

1.3.3 Desventajas

- JPA ha sido definida como parte de la especificación EJB 3.0 de Java EE 5, que supone una simplificación sobre versiones anteriores, por lo que ya no requiere de un contenedor EJB ni un servidor de aplicaciones Java EE. Esto implica que no podrá utilizarse en versiones anteriores de Java. (Raoul-Gabriel Urma, 2014)
- Interacciones Muchos de los desarrolladores de Java que han utilizado objetos persistentes que proporcionaban los marcos de código abierto o DAO's en lugar de Entity Beans e Enterprise Beans fue porque eran considerados demasiado pesados y complicados de utilizar.
- Hibernate la versión 3.2.0 Hibernate desarrolla la especificación JPA, con lo que ahora es posible desarrollar una capa de acceso a datos compatible con los estándares de Java en hibernate, y desplegarla en cualquier servidor de aplicaciones que soporte las especificaciones JEE5.

1.4 PostgreSQL

PostgreSQL es un potente sistema de base de datos objeto-relacional de código abierto. Cuenta con más de 15 años de desarrollo activo y una arquitectura probada que se ha ganado una sólida reputación de fiabilidad e integridad de datos. (Araya, 2012). Se ejecuta en los principales sistemas operativos que existen en la actualidad como:

- Linux
- UNIX (AIX, BSD, HP-UX, SGI IRIX, Mac OS X, Solaris, Tru64)
- Windows

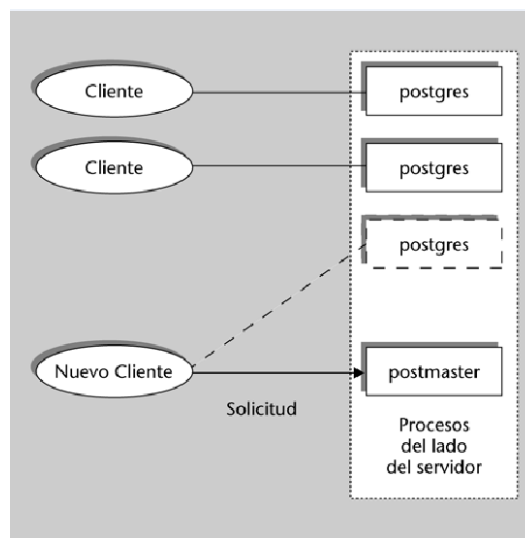


Figura 1-7. Arquitectura PostgreSQL

Fuente: <http://www.dataprix.com/72-arquitectura-postgresql>

Es una base de datos % ACID, tiene soporte completo para claves foráneas, uniones, vistas, disparadores y procedimientos almacenados (en varios lenguajes).

Incluye la mayoría de los tipos de datos del SQL 2008, incluyendo INTEGER, numérico, BOOLEAN, CHAR, VARCHAR, DATE, INTERVAL, y TIMESTAMP. (juntadeandalucia, 2015)

1.4.1 Características

Tabla 1.1: Características DBMS PostgreSQL

INDICADOR	CAPACIDAD
Tamaño máximo de la BD	Ilimitado
Tamaño máximo de las tablas	32 TB
Tamaño máximo de la fila 1.6 TB	Tamaño máximo de la fila 1.6 TB
Tamaño máximo para cada campo 1 GB	Tamaño máximo para cada campo 1 GB
Máximo de filas por tabla	Ilimitado
Máximo de columnas por tabla	250-1600 dependiendo del tipo de columna
Máximo de índices por tabla	Ilimitado

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

1.4.2 Ventajas:

- A pesar de que la velocidad de respuesta pueda parecer deficiente en bases de datos pequeñas, esa velocidad se mantiene al aumentar el tamaño de la base de datos, cosa que no sucede con otros programas, que se enlentecen brutalmente.
- Instalación ilimitada: No se puede demandar a una empresa por instalarlo en más ordenadores de los que la licencia permite, ya que no hay costo asociado a la licencia de software.
- Estabilidad y confiabilidad: No se han presentado caídas de la base de datos.
- Extensible: El código fuente está disponible de forma gratuita, para que quien necesite extender o personalizar el programa pueda hacerlo sin costes.
- Buen sistema de seguridad mediante la gestión de usuarios, grupos de usuarios y contraseñas.

1.4.3 Desventajas:

- En comparación con MySQL es más lento en inserciones y actualizaciones, ya que cuenta con cabeceras de intersección que no tiene MySQL.
- Soporte en línea: Hay foros oficiales, pero no hay una ayuda obligatoria.
- Consume más recursos que MySQL.
- La sintaxis de algunos de sus comandos o sentencias no es nada intuitiva.

1.5 PrimeFaces

Es una librería de componentes para Java Server Faces (JSF) de código abierto que cuenta con un conjunto de componentes enriquecidos que facilitan la creación de las aplicaciones web. Mantenido por Prime Technology, una compañía turca de IT especializada en consultoría ágil, JSF, Java EE y Outsourcing. El proyecto es liderado por Çağatay Çivici, un miembro del “JSF Expert Group” (y forofo del Barça).

Primefaces está bajo la licencia de Apache License V2. Una de las ventajas de utilizar Primefaces, es que permite la integración con otros componentes como por ejemplo RichFaces.

1.5.1 Ciclo de Vida



Figura 1.8: Características DBMS PostgreSQL

Fuente: www.globalmentoring.com/jsf

1.5.2 Ventajas

Soporte nativo de Ajax, incluyendo Push/Comet, es transparente para el desarrollador, aunque para activarlo deben utilizarse atributos específicos en cada uno de los componentes para lanzar un método del servidor y para indicar los componentes a actualizar, utiliza el soporte de jQuery para los efectos visuales. (Primefaces, 2013)

- Kit para crear aplicaciones web para móviles.

- Está Integrado con ThemeRoller Framework CSS. Donde se pueden elegir entre los 36 temas pre-diseñados o incluso crear los propios con la herramienta online de generador de temas
- Cuenta con muchos demos donde se incluyen su código XHTML y su Bean de Java.
- Es compatible con otras librerías de componentes, como JBoss RichFaces.
- Uso de JavaScript no intrusivo (no aparece en línea dentro de los elementos, sino dentro de un bloque <script>).

1.5.3 Desventajas:

- No podemos utilizar el soporte de Ajax de JSF 2 (mediante <f: Ajax>) con los componentes de Primefaces.
- Para utilizar el soporte de Ajax tenemos que indicarlo explícitamente, por medio de atributos específicos de cada componente.
- Varios de sus componentes no soportan la compatibilidad hacia atrás, es decir un componente que pertenece a una versión reciente por lo general no es compatible al 100% con un elemento de una versión anterior.

1.6 Diseño Web Adaptable

Actualmente en nuestro planeta existen más de 5000 millones de dispositivos conectados a Internet y debido a la gran demanda actual de terminales como Smartphones, tablets, libros electrónicos, televisiones, videoconsolas, etc.

El término Responsive Web Design se ha acuñado impuesto por la multitud de dispositivos existentes haciendo que los diseñadores comiencen a preocuparse por esta creciente tendencia, con el objetivo de llegar a todo tipo de usuario.

Aunque muchos piensan que es una tendencia de moda, realmente esta idea está desapareciendo, ya que cada día el concepto es más importante y está mayormente valorado por el usuario, transformándose en una necesidad adaptar nuestros diseños.

Además, las estadísticas a nivel mundial avalan un crecimiento de los dispositivos móviles frente a los dispositivos de escritorio, por ejemplo, existen países como India [en el cual el empleo del dispositivo móvil ya ha superado al de escritorio y a nivel global se estimó que en el año 2014 el

número de dispositivos móviles superará el número de personas en el mundo y ya es una realidad para este 2016 la tendencia se mantiene con una variabilidad que va creciendo.

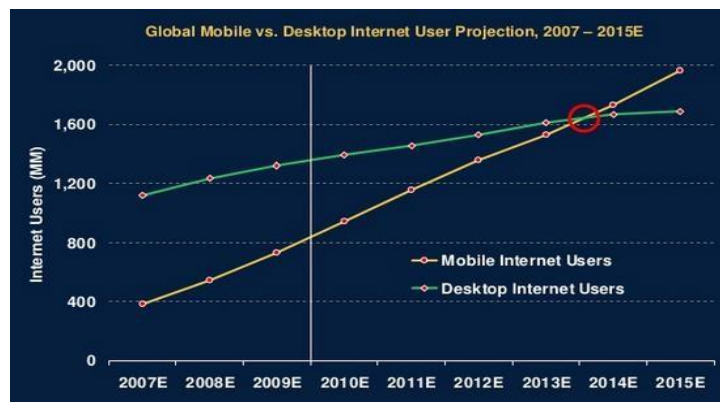


Figura 1-9: Usuario Móvil vs Usuario de Escritorio

Fuente: www.reportecaos.com

1.6.1 Mobile First

Hasta ahora todos los sitios web eran diseñados sólo para equipos de sobremesa y el proceso de navegar por la Web en los teléfonos móviles era bastante incómodo. Sin embargo, las tecnologías están cambiando y el principio de Mobile First se está convirtiendo en un concepto cada vez más extendido.

Mobile First es una filosofía la cual señala la priorización del entorno móvil frente al escritorio a la hora de desarrollar experiencias de usuario. (www.jy-soft.com, 2015)

Mobile First se trata de la práctica de desarrollar un diseño desde su forma más básica planteándolo primeramente para la pequeña pantalla de un Smartphone, basándose en un crecimiento iterativo que aporte cimientos al diseño para otros dispositivos de mayor resolución.

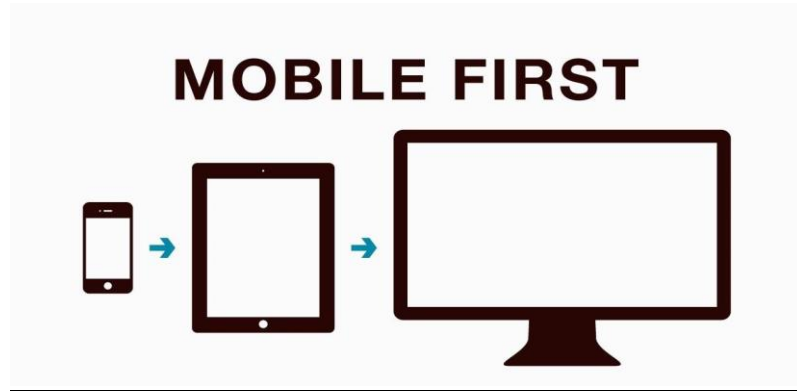


Figura 1-10. Mobile First.

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

1.7 SOAP

1.7.1 Soa

SOA (Service-Oriented Architecture o Arquitectura Orientada a Servicios en castellano) son arquitecturas de software que definen el uso de servicios como soporte a los requisitos del negocio, cuyo objetivo es alcanzar el mínimo acoplamiento posible entre agentes software.

Un servicio es una unidad de trabajo realizada por un proveedor de servicios para alcanzar un resultado final deseado por un consumidor del servicio.

Tanto el proveedor como el consumidor del servicio son roles realizados por agentes software en lugar de sus propietarios.

En un sentido general, una arquitectura orientada a servicios es una solución software que pretende permitir a la empresa organizar y hacer uso de múltiples procesos. Con SOA, las aplicaciones software ya no son enormes bloques de funciones y procesos. En cambio, estas aplicaciones se componen de servicios modulares ensamblados. (microbuffer, 2014)

¿Cómo consigue SOA desacoplar los agentes software que interactúan? Utilizando dos principios en la definición de la arquitectura:

- Escoger un pequeño conjunto de interfaces simples y distribuidas para todos los agentes de software participantes. Sólo la semántica genérica es codificada en las interfaces. Éstas deberían estar disponibles universalmente para todos los proveedores y consumidores.
- Definir mensajes descriptivos por un esquema extensible que se entrega a través de las interfaces. No se ofrece información sobre el funcionamiento del sistema a través de mensajes. Los esquemas son los encargados de limitar el vocabulario y estructura de los mensajes.

1.7.2 Restricciones

Un Servicio Web SOAP introduce las siguientes restricciones sobre las características ya citadas de SOA:

- Excepto para datos binarios anexos, los mensajes deben ser transportados sobre SOAP.
- Muestra cómo interactúa un conjunto de Servicios Web:

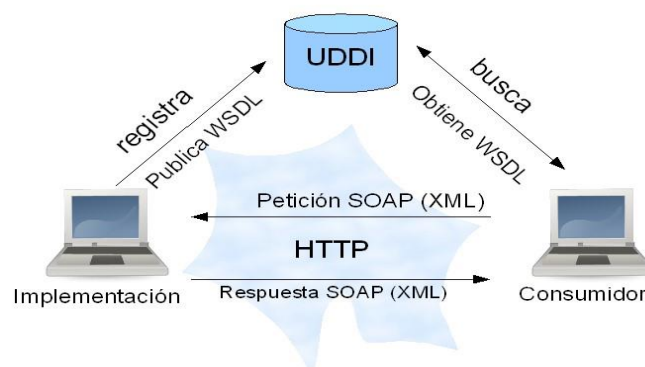


Tabla 1-11. Interacción a través de Servicios Web.

Fuente: <http://slideplayer.es/slide>

- La descripción de un servicio debe ser hecha en WSDL.
- Uso de UDDI, que son las siglas del catálogo de negocios de Internet denominado Universal Description, Discovery and Integration.

CAPÍTULO II

2 MARCO METODOLÓGICO

2.1 Descripción general de la metodología

Para el desarrollo del OCW se usó la metodología SCRUM ya que permite llevar un control oportuno del estado del proyecto al dividirlo en iteraciones o sprint (producto funcional entregable al dueño del producto); cada sprint abarca un conjunto de tareas (sprint backlog); al final se mostrará un gráfico que reflejará el esfuerzo y estado de desarrollo del proyecto (burndown chart).

2.1.1 Personas y Roles del Proyecto

Para la ejecución del Proyecto cuenta con la participación de un equipo de trabajo conformado por 5 personas cuyo rol se especifica a continuación en la siguiente tabla. (Tabla 1)

Tabla 2-1: Personas y Roles del Proyecto.

Persona	Contacto	Rol
Dr. Byron Díaz Monroy PhD. (Rector ESPOCH)	bdiaz@esPOCH.edu.ec	Product Owner
Ing. Jorge Huilca (Vicedecano FIE-ESPOCH)	jhuilca@esPOCH.edu.ec	Stakeholder
Ing. Javier Romero	wromero@esPOCH.edu.ec	Scrum Master
Carlos Mejía	michael.mejia_92@hotmail.com	Team Development
Jhonny Guevara	jhonny_gc@yahoo.com	

Realizado por: Carlos Mejía, Jhonny Guevara, 2016

2.1.2 Tipos y roles de Usuario

En el Sistema Open Course Ware ESPOCH (OCW), se describen tres tipos de usuario, los roles que cumplen cada uno se describen en la Tabla IV, a continuación.

Tabla 2-2: Tipos y Roles de Usuario.

Tipo de Usuario	Rol
Administrador	Es el usuario quien gestiona: cursos, docentes, períodos, materias, modalidad
Docente	Es el usuario quien gestiona: su perfil y contenidos de sus cursos
Visitante	Es el usuario quien puede acceder a los diferentes cursos del OCW ESPOCH.

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

2.1.3 Planificación del Proyecto

En esta etapa se detallan cada uno de los procedimientos realizados para definir los requisitos, identificar las historias de usuario y distribuirlas para su posterior desarrollo; las historias de usuario son definidas, estimadas y priorizadas. Se describen las tareas de ingeniería y las pruebas de aceptación de cada historia necesaria previo al diseño del OCW.

En la descripción de las historias de usuario se usa la métrica de puntos estimados, cada punto estimado equivale a 1 hora de trabajo.

La planificación del proyecto OCW permite proporcionar un marco de trabajo que permita al gestor hacer estimaciones razonables en base a lo planificado.

Las reuniones con el Stakeholder (Ing, Jorge Huilca) permitieron establecer los requerimientos necesarios para el desarrollo del sistema, la base para la determinar la funcionalidad del sistema fueron las necesidades recolectadas de los docentes de la ESPOCH para publicar contenidos de un curso.

El desarrollo del proyecto tiene una duración de 16 semanas tomando como fecha de inicio 02/10/15 y fecha de finalización 19/01/16, cada semana comprende 40 horas laborables distribuidas en 5 días y 8 horas por día; la estimación del proyecto se la realizó en 2 reuniones de planificación con el Ing Jorge Huilca (Stakeholder) empleando la técnica ágil de estimación de la talla de la camiseta, la cual no fue muy acertada en vista que el team development no tenía mucha

experiencia sobre proyectos OCW; se recomendaría emplear un método más acertado como el planning póker que nos permite obtener la los puntos estimados por cada historia de usuario(HU).

2.1.4 Preparación Del Proyecto

En base a las necesidades de la empresa y capacidad de despliegue de aplicaciones en servidores se planteó la creación de una aplicación web diseñada con Frameworks basados en lenguaje Java (PRIMEFACES/JPA) que permita gestionar los procesos y almacenar los datos recolectados en una base de datos relacional Postgresql, facilitando una posterior manipulación de la información generada a través de los diferentes procesos que realiza la empresa.

Para la ejecución del proyecto inicialmente se realizó un análisis de las necesidades (Requerimientos) del cliente de Software con la finalidad de establecer las funcionalidades necesarias para el sistema informático, posteriormente se definió la arquitectura del sistema para el despliegue del proyecto, se diseñó la base de datos y se estableció-configuró las herramientas de software a utilizar.

2.1.5 Requerimientos

Los requerimientos representan la funcionalidad que deberá poseer el sistema con el fin de satisfacer las necesidades del usuario, los mismos que se definen y se refinan a través de reuniones con los implicados en los procesos a automatizar con el objetivo de garantizar que el sistema cumpla ESPOCH finalmente a las expectativas del usuario final.

Un requerimiento es una característica que el sistema debe tener o es una restricción que el módulo debe satisfacer para ser aceptada por el cliente (Caldas, 2014).

La prioridad de los requerimientos se ha establecido considerando el criterio de importancia para el Stakeholder, esta información se encuentra detallada en la siguiente tabla. (Tabla 2.3).

Tabla 2-3: Niveles de Prioridad

Criterio de Prioridad	Valor de Prioridad
Alto	3
Medio	2
Bajo	1

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Los requerimientos para el proyecto fueron establecidos conjuntamente con el Stakeholder (Ing. Jorge Huilca). En las entrevistas que se mantuvo con los usuarios se determinaron 25 requisitos básicos y 10 historias técnicas; que se detallan a continuación.

Los requerimientos no funcionales del sistema son:

- **Confidencialidad:** la información manejada por el sistema tiene los controles apropiados que restringen el uso a personas no autorizadas.
- **Integridad:** la información del sistema está controlada para evitar inconsistencias y registro de datos erróneos que complican el manejo del sistema.
- **Disponibilidad:** el sistema se encuentra desplegado y disponible para cualquier usuario con acceso internet.

En las entrevistas que se mantuvo con los usuarios se determinaron 25 requisitos básicos y 10 historias técnicas; que se detallan a continuación.

HISTORIAS DE USUARIO DEL SISTEMA

1. El sistema permitirá manejar las sesiones de los usuarios: (Administrador y Docente).
2. El sistema permitirá ingresar, editar y listar los usuarios (Administrador).
3. El sistema permitirá visualizar las materias (Administrador).
4. El sistema permitirá visualizar los períodos (Administrador).

5. El sistema permitirá editar información de perfil (docente).
6. El sistema permitirá ingresar, editar y listar los cursos (Administrador).
7. El sistema permitirá listar cursos por carrera, publicar y editar perfil cursos (Docente).
8. El sistema permitirá ingresar, editar, listar y visualizar las actividades de un curso (Docente).
9. El sistema permitirá ver el perfil, vaciar contenido, ingresar, editar y listar capítulos de un curso (Docente)
10. El sistema permitirá ingresar, editar, listar y visualizar las lecturas de un curso (Docente).
11. El sistema permitirá ingresar, editar, listar y visualizar los recursos de un curso (Docente).
12. El sistema permitirá ingresar, editar, listar y visualizar los temas de un curso (Docente).
13. El sistema permitirá listar los cursos: más vistos, más recientes, publicados, (Visitante).
14. El sistema permitirá mostrar la galería de imágenes de un curso (visitante).
15. El sistema permitirá mostrar la lista de docentes que imparten los cursos (visitante).
16. El sistema permitirá buscar cursos por: facultad, escuela, carrera, nombre, nivel. (Visitante).
17. El sistema permitirá mostrar de un curso: el sílabo, calendario de actividades, material, capítulos, lecturas, temas, actividades, recursos (visitante).
18. El sistema permitirá conectarse con los servicios web de facultad desarrollados para el OASIS y mostrarlas en una lista
19. El sistema permitirá conectarse con los servicios web de escuela desarrollados para el OASIS y mostrarlas en una lista
20. El sistema permitirá conectarse con los servicios web de período desarrollados para el OASIS y mostrarlos en una lista
21. El sistema permitirá conectarse con los servicios web de docente desarrollados para el OASIS y mostrarlos en una lista
22. El Sistema Permitirá gestionar Preguntas.
23. El sistema permitirá gestionar Exámenes.
24. Ingreso de registro de auditoría
25. Ingreso de registro de error

HISTORIAS DE USUARIO TÉCNICAS

1. Diseñar la base de datos
2. Definir el estándar de codificación.
3. Diseñar la arquitectura del sistema
4. Definir el diseño de la interfaz de usuario.
5. Preparar el servidor para el despliegue de la aplicación.
6. Desplegar el OCW sobre el servidor.
7. Realizar pruebas de la aplicación
8. Elaborar la documentación del sistema
9. Capacitar a los usuarios del sistema
10. Elaborar la documentación

Mediante una reunión inicial en la que se definió la funcionalidad del sistema y se planteó veinticinco requerimientos funcionales o historias de usuario (HU) y diez historias técnicas (HT); éstas representan la configuración y despliegue adicional a los requerimientos establecidos que son necesarios para el desarrollo del sistema; las mismas que están descritas a continuación en la tabla 2.4, en donde se detalla en la primera columna un identificador de la historia, en la segunda una descripción de la misma, en la tercera el esfuerzo realizado para dicha historia y en la cuarta la prioridad asignada.

Tabla 2-4: Product Backlog

N°	Descripción	Esfuerzo	Prioridad
HT01	Como desarrollador necesito realizar el Diseño Técnico de la Arquitectura.	24	3
HT02	Como desarrollador necesito realizar el Diseño técnico de las interfaces	24	3
HT03	Como desarrollador necesito realizar el diseño técnico de la base de datos.	32	3

HT04	Como desarrollador necesito definir el estándar de codificación para el sistema	16	3
HU01	Como usuario del sistema ocw necesito poder autenticarme para gestionar mis contenidos.	16	3
HU02	Como Administrador necesito ingresar, editar y listar los usuarios que poseerá el OCW.	32	3
HU03	Como Administrador necesito visualizar las materias que poseerá el OCW.	8	3
HU04	Como Administrador necesito visualizar los períodos que poseerá el OCW.	8	3
HU05	Como Docente necesito editar información de perfil.	32	3
HU06	Como Administrador necesito ingresar, editar y listar los cursos que poseerá el OCW.	48	3
HU07	Como Docente necesito listar cursos por carrera, publicar y editar perfil cursos que poseerá el OCW.	48	3
HU08	Como Docente necesito ingresar, editar, listar y visualizar las actividades de un curso que poseerá el OCW.	48	3
HU09	Como Docente necesito ver el perfil, vaciar contenido, ingresar, editar y listar capítulos de un curso que poseerá el OCW.	48	3
HU10	Como Docente necesito ingresar, editar, listar y visualizar las lecturas de un curso que poseerá el OCW.	48	3
HU11	Como Docente necesito ingresar, editar, listar y visualizar los recursos de un curso del OCW.	48	3
HU12	Como Docente necesito ingresar, editar, listar y visualizar los temas de un curso	48	3
HU13	Como Visitante necesito listar los cursos: más vistos, más recientes, publicados.	32	3
HU14	Como Visitante necesito mostrar la galería de imágenes de un curso	16	3
HU15	Como Visitante necesito mostrar la lista de docentes que imparten los cursos.	16	2
HU16	Como Visitante necesito buscar cursos por: facultad, escuela, carrera, nombre, nivel.	32	2
HU17	Como Visitante necesito mostrar de un curso: el sílabo, calendario de actividades, material, capítulos, lecturas, temas, actividades, recursos	48	2
HU18	Como docente necesito conectarme con los servicios web de facultad desarrollados para el OASIS y mostrarlas en una lista.	24	2
HU19	Como docente necesito conectarse con los servicios web de escuela desarrollados para el OASIS y mostrarlas en una lista	24	2
HU20	Como docente necesito conectarse con los servicios web de período desarrollados para el OASIS y mostrarlos en una lista	24	2

HU21	Como Administrador necesito visualizar los datos de un docente mediante la conexión con los servicios web de docente desarrollados para el OASIS.	24	2
HU22	Como docente necesito gestionar Preguntas	48	2
HU23	Como docente necesito gestionar Exámenes	24	2
HU24	Como administrador necesito registrar las auditorías sobre acciones que se realizan en el sistema.	16	2
HU25	Como administrador necesito registrar los errores sobre acciones que se realizan en el sistema.	8	2
HT05	Como desarrollador necesito preparar el Servidor para la aplicación.	32	1
HT06	Como desarrollador necesito desplegar la aplicación para anclarla al servidor	16	1
HT07	Como desarrollador necesito realizar pruebas a la aplicación para garantizar su óptimo funcionamiento	40	1
HT08	Como desarrollador necesito capacitar a los usuarios para que conozcan la manera de operar el sistema OCW.	8	1
HT09	Como desarrollador necesito aplicar mejoras al sistema para satisfacer los colores de la institución.	8	1
HT10	Como Desarrollador necesito elaborar la documentación del sistema para registrar los eventos suscitados en el avance del proyecto.	40	1

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Las historias de usuario detalladas en la tabla anterior (Tabla 3) fueron consecuencia de los requisitos básicos propuestos por el Stakeholder, con la finalidad de ser atendidas con la mayor brevedad posible; se recomienda el uso de esta técnica puesto que permite acoplar las historias de usuario a una subdivisión que pueda ser admitida en una división de bloques para entregables de un sprint.

2.1.6 Sprints Del Proyecto

Los sprints son bloques cortos que representan un entregable; en el proyecto son los módulos en los cuales fue dividido el proyecto.

El sistema fue dividido en 21 sprints con una duración de 48 horas cada uno (6 días laborables), los que se definieron tomando en cuenta la secuencia de desarrollo y la prioridad de los requerimientos, esto permitió presentar resultados y avances del sistema al Stakeholder ordenados.

El OCW al ser un proyecto de desarrollo compartido destinado a dos personas; se lo ha distribuido de la siguiente manera:

Los sprint 1,2,20,21 han sido compartidos es decir son actividades que se han realizado en conjunto como team development; por otra parte, los sprint realizados individualmente por los desarrolladores se encuentran descritos En la (tabla 2-5) a continuación, mostrando todas las historias de usuario descritas y asignadas al sprint que le corresponde. Se especifica en la primera columna el número de sprint asignado, luego la descripción o nombre de la historia de usuario a desarrollar, la fecha de inicio y fin de cada sprint, el esfuerzo necesario para resolver el sprint y la persona o equipo responsable de cada sprint.

Tabla 2-5. Sprint Backlog

Sprint	Descripción	Inicio	Fin	Esfuerzo	Responsable
01	HT01	02/10/15	06/10/15	24	Team Development
	HT02	07/10/15	09/10/15	24	
02	HT03	12/10/15	15/10/15	32	Team Development
	HT04	16/10/15	19/10/15	16	
03	HU01	20/10/15	21/10/15	16	Carlos Mejía G.
	HU02	22/10/15	27/10/15	32	
04	HU03	20/10/15	20/10/15	8	Jhonny Guevara
	HU04	21/10/15	21/10/15	8	
	HU05	22/10/15	27/10/15	32	
05	HU06	28/10/15	04/11/15	48	Carlos Mejía G.
06	HU07	28/10/15	04/11/15	48	Jhonny Guevara
07	HU08	05/11/15	12/11/15	48	Carlos Mejía G.
08	HU09	05/11/15	12/11/15	48	Jhonny Guevara
09	HU10	13/11/15	20/11/15	48	Carlos Mejía G.
10	HU11	13/11/15	20/11/15	48	Jhonny Guevara
11	HU12	23/11/15	30/11/15	48	Carlos Mejía G.

12	HU13	23/11/15	26/11/15	32	Jhonny Guevara
	HU14	27/11/15	30/11/15	16	
13	HU15	01/12/15	02/12/15	16	Carlos Mejía G.
	HU16	03/12/15	08/12/15	32	
14	HU17	01/12/15	08/12/15	48	Jhonny Guevara
15	HU18	09/12/15	11/12/15	24	Carlos Mejía G.
	HU19	14/12/15	16/12/15	24	
16	HU20	9/12/15	11/12/15	24	Jhonny Guevara
	HU21	14/12/15	16/12/15	24	
17	HU22	17/12/15	24/12/15	48	Carlos Mejía G.
18	HU23	17/12/15	21/12/15	24	Jhonny Guevara
	HU24	22/12/15	23/12/15	16	
	HU25	24/12/15	24/12/15	8	
19	HT05	25/12/15	30/12/15	32	Team Development
	HT06	31/12/15	01/01/16	16	
20	HT07	04/01/16	08/01/16	40	Team Development
	HT08	11/01/16	11/01/16	8	
21	HT09	12/01/16	12/01/16	8	Team Development
	HT10	13/01/16	19/01/16	40	

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Los sprint del 1 al 21 cumplen con la cantidad de 48 puntos que fue nuestra métrica para dividir las tareas de desarrollo del proyecto; considero que es recomendable mantener la métrica de cinco días para futuros proyectos ya que todos los miembros del equipo de trabajo conocen exactamente el día para reuniones de inicio y fin de sprint.

2.1.7 Diagramas de caso de uso

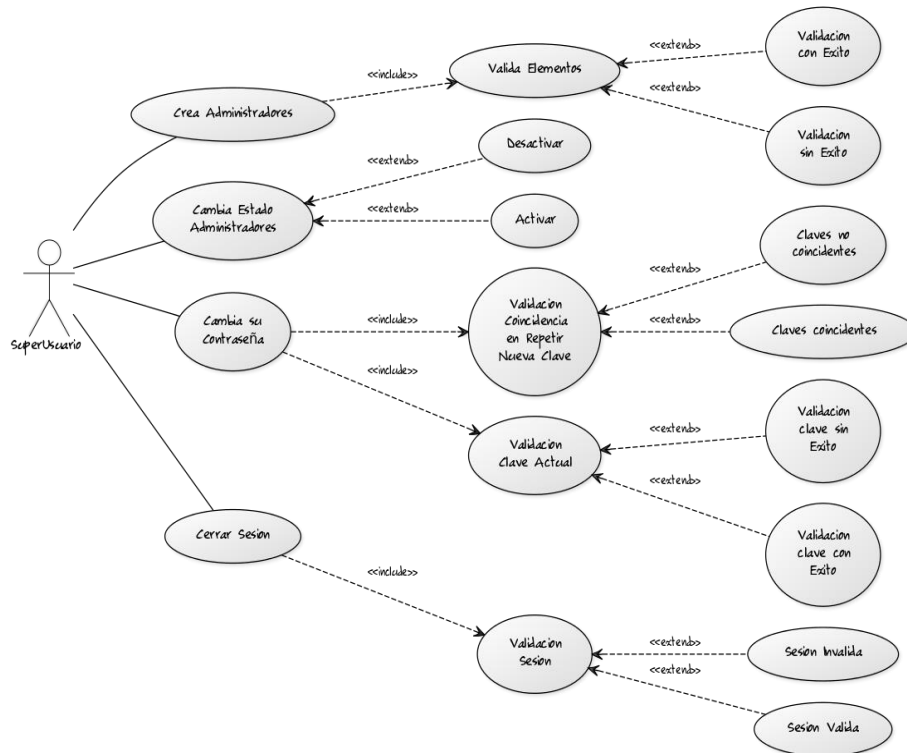


Figura 2-1: Diagrama de Clases SuperUsuario

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-6: Detalle de caso de uso SU-001

ID Caso de Uso	SU-001
Nombre de Caso de Uso	Crear Administradores
Descripción	El SuperUsuario podrá crear usuarios Administradores mediante un numero de cedula.
Actores	SuperUsuario
Precondiciones	El usuario que se intenta crear debe pertenecer al sistema académico de la ESPOCH OASIS y ser de tipo docente.
Flujo Básico de Datos	1.- Ingreso de cedula del docente 2.- Se ejecuta la búsqueda. 3.-Si la búsqueda termina con éxito, los datos del usuario a crear se cargarán.
Excepciones	Error Interno (La petición al servidor ha tardado demasiado)

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-7: Detalle de caso de uso SU-002

ID Caso de Uso	SU-002
Nombre de Caso de Uso	Cambiar el estado de los usuarios Administradores
Descripción	El SuperUsuario podrá cambiar el estado de cualquier Administrador registrado en el sistema.
Actores	SuperUsuario
Precondiciones	Se deberá seleccionar un usuario de la lista existente. Para poner el estado en Activo, deberá tener actualmente el estado en Inactivo, y viceversa.
Flujo Básico de Datos	1.- Ingreso de cedula del docente 2.- Se ejecuta la búsqueda. 3.-Si la búsqueda termina con éxito, los datos del usuario a crear se cargarán.
Excepciones	Error Interno (La petición al servidor ha tardado demasiado)

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-8: Detalle de caso de uso SU-003

ID Caso de Uso	SU-003
Nombre de Caso de Uso	Cambiar su contraseña
Descripción	El SuperUsuario podrá cambiar su contraseña
Actores	SuperUsuario
Precondiciones	Se requiere ingresar la contraseña actual para poder realizar el proceso de cambio de contraseña.
Flujo Básico de Datos	1.- Ingreso de la contraseña actual 2.- Ingreso de la nueva contraseña. 3.- Repetición de la nueva contraseña. 3.- En caso de coincidir la contraseña actual ingresada y ser idénticas las contraseñas nuevas pedidas se realiza el proceso de cambio de contraseña en el sistema
Excepciones	Advertencia (No coincide la contraseña actual con la ingresada.) Advertencia (No coincide las nuevas contraseñas) Error Interno (La petición al servidor ha tardado demasiado)

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-9: Detalle de caso de uso SU-004

ID Caso de Uso	SU-004
Nombre de Caso de Uso	Cerrar Sesión
Descripción	El SuperUsuario podrá cerrar su sesión actual
Actores	SuperUsuario
Precondiciones	Tener una sesión activa.
Flujo Básico de Datos	1.- Accionar el botón de cierre de sesión. 2.- Si la petición se realiza con éxito se re direcciona a la página de login.

Excepciones	Error Interno (La petición al servidor ha tardado demasiado)
--------------------	--

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

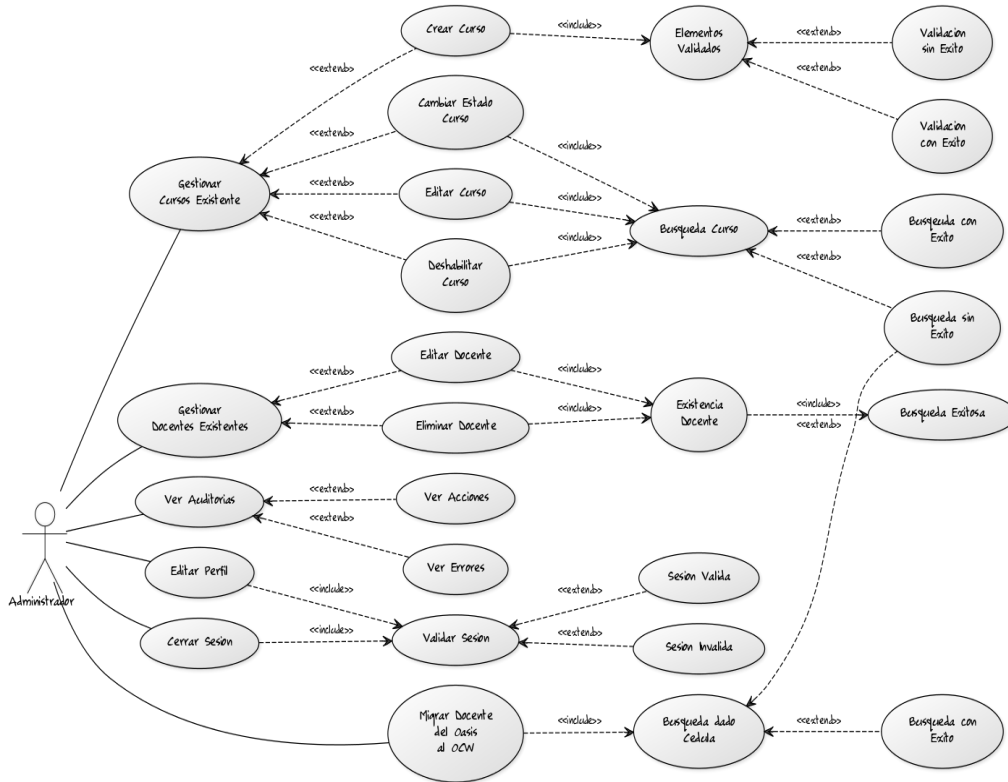


Figura 2-2: Casos de uso Administrador

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-10: Detalle de caso de uso A-001

ID Caso de Uso	A-001
Nombre de Caso de Uso	Gestionar Curso Existente
Descripción	El Administrador podrá Crear, Cambiar el estado, Editar, y desactivar un curso
Actores	SuperUsuario
Precondiciones	Situarse en página del listado de curso o Home del Administrador Seleccionar el curso a gestionar.
Flujo Básico de Datos	<ol style="list-style-type: none"> 1.- Accionar el botón para la acción correspondiente. 2.- Se traslada a la página correspondiente. <ol style="list-style-type: none"> 2.1.- En caso de crear deberá: <ol style="list-style-type: none"> 2.1.1- Ingresar la cedula de docente que impartirá ese curso y Accionar la Búsqueda. 2.1.2- Seleccionar una carrera a la que estará asociada este curso Y accionar la búsqueda. 3.-Se gestiona sus elementos. 4.-En caso de realizarse la acción con éxito se redirecciona a la lista de cursos de la que partió.

Excepciones	Error Interno (La petición al servidor ha tardado demasiado)
--------------------	--

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-11: Detalle de caso de uso A-002

ID UC	A-002
Nombre de Caso de Uso	Gestionar Docentes
Descripción	El Administrador podrá Crear, Cambiar el estado, Editar, y desactivar un docente
Actores	Administrador
Precondiciones	Situarse en página del listado de Docentes. Seleccionar un docente a gestionar.
Flujo Básico de Datos	1.- Accionar el botón para la acción correspondiente. 2.- Se traslada a la página correspondiente. 2.2- En caso de crear un docente se deberá. 2.2.1 Ingresar la cedula del docente a crear y accionar la búsqueda 3.-Se gestiona sus elementos. 4.-En caso de realizarse la acción con éxito se redirecciona a la lista de docentes de la que partió.
Excepciones	Error Interno (La petición al servidor ha tardado demasiado)

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-12: Detalle de caso de uso A-003

ID UC	A-003
Nombre de Caso de Uso	Ver Auditorias
Descripción	El Administrador podrá Ver y filtrar auditorias por fecha
Actores	Administrador
Precondiciones	Seleccionar un tipo de auditoria a ver detalle.
Flujo Básico de Datos	1.- Accionar el botón para la acción correspondiente (ítem del menú Auditoria). 2.- Se traslada a la página correspondiente. 3.-Se selecciona una fecha inicio y fin. 4.-se visualiza la lista de auditorías de entre esas fechas.
Excepciones	Error Interno (La petición al servidor ha tardado demasiado)

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-13: Detalle de caso de uso A-004

ID UC	A-004
Nombre de Caso de Uso	Editar Perfil
Descripción	El Administrador podrá editar su perfil
Actores	Administrador
Precondiciones	Situarse en la página correspondiente a esta acción.
Flujo Básico de Datos	1.- Dirigirse a la página que permite esta acción mediante el botón editar perfil ubicado en la parte superior derecha junto a la foto. 2.- Se trasladará a la página correspondiente. 3.-Debera modificar los alores requeridos y accionar el botón Actualizar. 4.-Se visualizará inmediatamente los datos actualizados en la misma pagina
Excepciones	Error Interno (La petición al servidor ha tardado demasiado)

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-14: Detalle de caso de uso A-005

ID Caso de Uso	A-005
Nombre de Caso de Uso	Cerrar Sesión
Descripción	El Administrador podrá cerrar su sesión actual
Actores	Administrador
Precondiciones	Tener una sesión activa.
Flujo Básico de Datos	1.- Accionar el botón de cierre de sesión. 2.- Si la petición se realiza con éxito se redirecciona a la página de login.
Excepciones	Error Interno (La petición al servidor ha tardado demasiado)

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

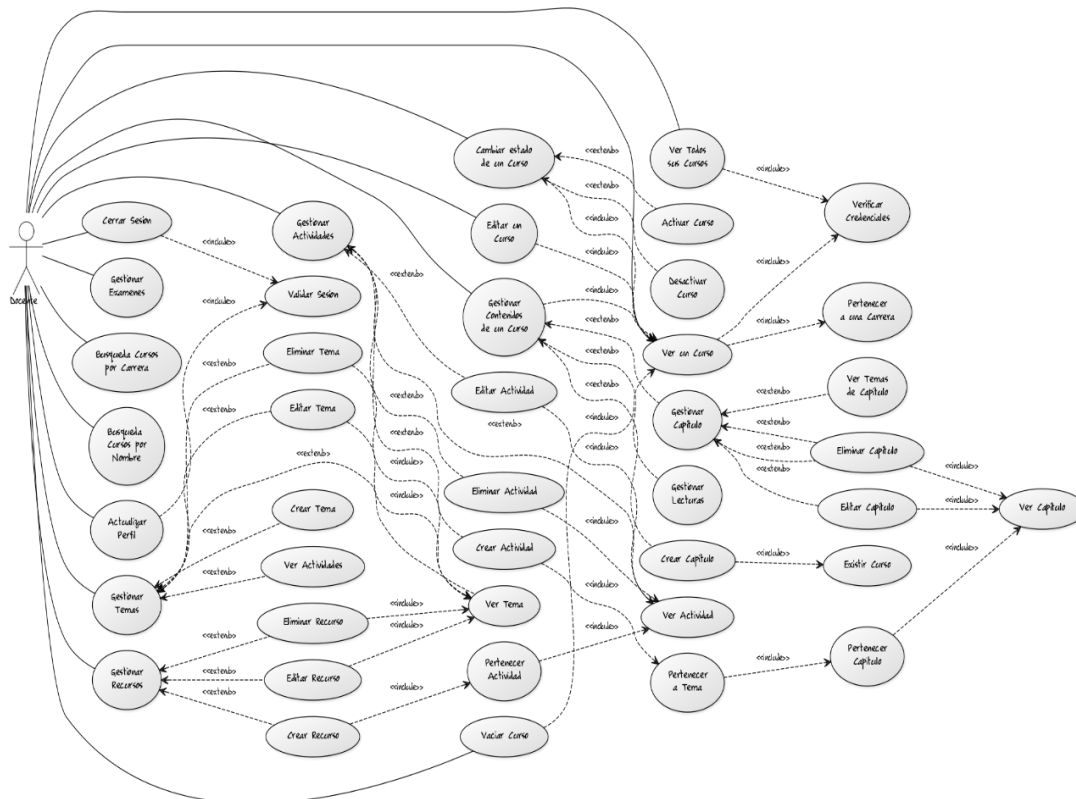


Figura 2-3. Casos de uso Docente

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-15: Detalle de caso de uso D-001

ID Caso de Uso	D-001
Nombre de Caso de Uso	Gestionar Curso Existente
Descripción	El Docente podrá Cambiar el estado, Ver, Editar, y desactivar un curso
Actores	SuperUsuario
Precondiciones	Situarse en página del listado de curso o Home del Docente Seleccionar el curso a gestionar.
Flujo Básico de Datos	<ol style="list-style-type: none"> 1.- Accionar el botón para la acción correspondiente. 2.- Se traslada a la página correspondiente. <ol style="list-style-type: none"> 2.1.- En caso de Desactivar deberá: <ol style="list-style-type: none"> 2.1.1- El curso deberá estar en estado Activo. 3.-Se gestiona sus elementos requeridos. 4.-En caso de realizarse la acción con éxito se redirecciona a la lista de cursos de la que partió, caso contrario se mantendrá en la misma página y recibirá los mensajes informativos de campos no válidos. Hasta que pasen la validación correspondiente.
Excepciones	Error Interno (La petición al servidor ha tardado demasiado)

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-16: Detalle de caso de uso D-002

ID Caso de Uso	A-002
Nombre de Caso de Uso	Gestionar Exámenes.
Descripción	El Docente podrá Crear, Editar, y Eliminar exámenes de un curso
Actores	Docente
Precondiciones	Situarse en página del listado de cursos o Home del Docente. Seleccionar la opción Gestionar Exámenes.
Flujo Básico de Datos	1.- Se traslada a la página correspondiente. 1.1.- En caso de crear deberá: 1.1.1.- Ingresa nombre del examen. 1.2.- En caso de editar deberá: 1.2.1.- Seleccionar un examen y editar sus elementos. 1.2.2.- Se validan los cambios, si todo esta correcto se emite el mensaje de actualización correcta 1.2.3.- caso contrario se mantiene en la misma página y se informa de elementos con valores no validos
Excepciones	Error Interno (La petición al servidor ha tardado demasiado)

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-17: Detalle de caso de uso D-003

ID Caso de Uso	A-003
Nombre de Caso de Uso	Gestionar Contenidos.
Descripción	El Docente podrá gestionar lecturas, capítulos, temas, actividades y recursos de un curso
Actores	Docente
Precondiciones	Situarse en página del listado de cursos o Home del Docente.
Flujo Básico de Datos	1.- Se traslada a la página correspondiente. 1.1.- Según el caso: 1.1.1.-Para crear capítulo, tema, lectura, actividad y recurso deberá: Ingresa nombre e información necesaria, seguidamente accionar en botón guardar. 1.1.2.-Para editar capítulo, tema, lectura, actividad y recurso deberá: Reemplazar el nombre e información necesaria, seguidamente accionar en botón actualizar. 1.1.2.-Para eliminar capítulo, tema, lectura, actividad y recurso deberá: Seleccionar capítulo y accionar en botón eliminar. 1.2.- Se emitirán los mensajes satisfactorios correspondientes. 1.2.3.- caso contrario se mantiene en la misma página y se Informa de elementos con valores no válidos.
Excepciones	Error Interno (La petición al servidor ha tardado demasiado)

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-18: Detalle de caso de uso D-004

ID UC	A-004
Nombre de Caso de Uso	Editar Perfil
Descripción	El Docente podrá editar su perfil
Actores	Docente
Precondiciones	Situarse en la página correspondiente a esta acción.
Flujo Básico de Datos	1.- Dirigirse a la página que permite esta acción mediante el botón editar perfil ubicado en la parte superior derecha junto a la foto. 2.- Se trasladará a la página correspondiente. 3.-Debera modificar los alores requeridos y accionar el botón Actualizar. 4.-Se visualizará inmediatamente los datos actualizados en la misma pagina
Excepciones	Error Interno (La petición al servidor ha tardado demasiado)

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-19: Detalle de caso de uso D-005

ID Caso de Uso	A-005
Nombre de Caso de Uso	Buscar Cursos por Carrera.
Descripción	El Docente podrá realizar una búsqueda de sus cursos según carrera requerida
Actores	Docente
Precondiciones	Situarse en página del listado de cursos por Carrera. Tener asignado al menos un curso.
Flujo Básico de Datos	1.- Se traslada a la página correspondiente. 1.1.- Seleccionar una carrera deseada. 1.2.- Se cargarán los cursos asociados al docente logeado y carrera seleccionada
Excepciones	Error Interno (La petición al servidor ha tardado demasiado)

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-20: Detalle de caso de uso D-006

ID Caso de Uso	A-006
Nombre de Caso de Uso	Buscar Cursos por Nombre.
Descripción	El Docente podrá realizar una búsqueda por nombre de curso
Actores	Docente
Precondiciones	Situarse en página del listado de cursos por Nombre. Tener asignado al menos un curso.
Flujo Básico de Datos	1.- Se traslada a la página correspondiente. 1.1.- En la caja de búsqueda se ingresará un nombre o indicio. 1.2.- Se cargarán los cursos asociados la palabra ingresada o indicio
Excepciones	Error Interno (La petición al servidor ha tardado demasiado)

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-21: Detalle de caso de uso D-007

ID Caso de Uso	D-007
Nombre de Caso de Uso	Cerrar Sesión
Descripción	El Docente podrá cerrar su sesión actual
Actores	Docente
Precondiciones	Tener una sesión activa.
Flujo Básico de Datos	1.- Accionar el botón de cierre de sesión. 2.- Si la petición se realiza con éxito se redirecciona a la página de login.
Excepciones	Error Interno (La petición al servidor ha tardado demasiado)

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

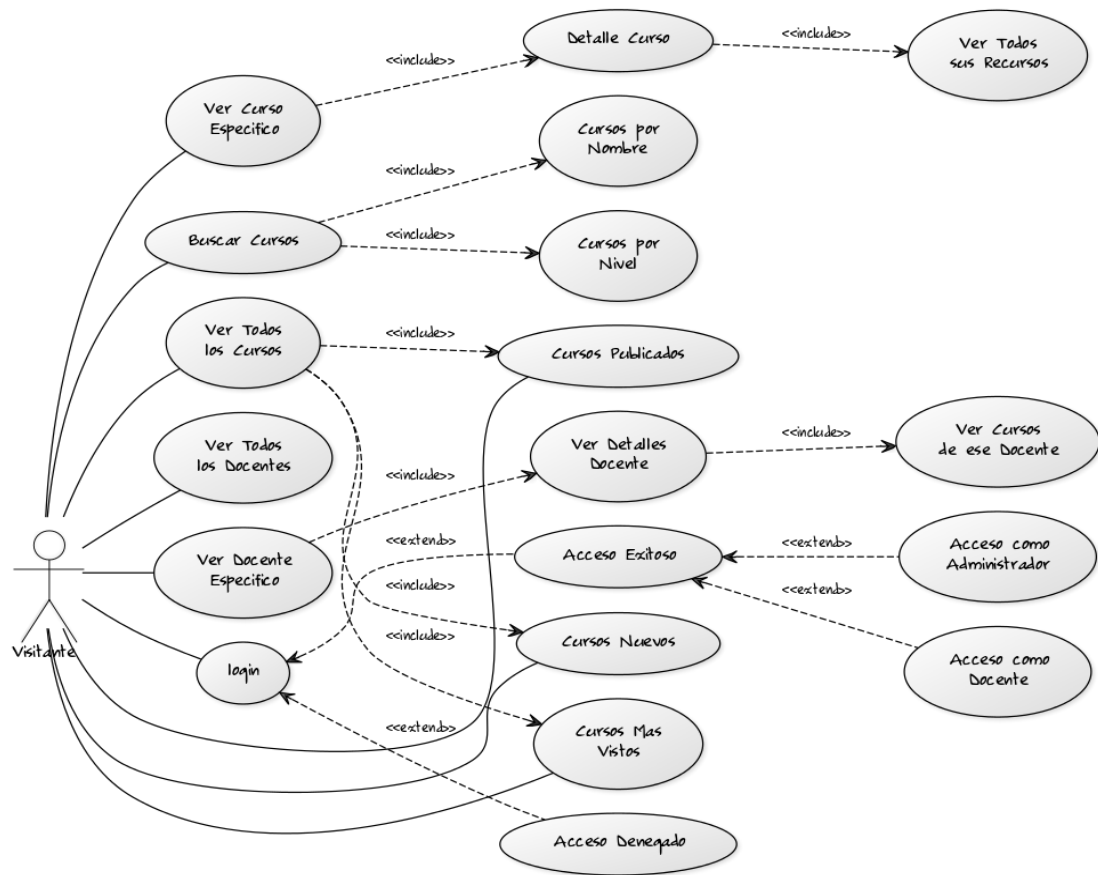


Figura 2-4: Casos de uso Visitante

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-22: Detalle de caso de uso V-001

ID Caso de Uso	V-001
Nombre de Caso de Uso	Buscar Cursos por Nombre.
Descripción	El Usuario Visitante podrá realizar una búsqueda por nombre.

Actores	Usuario Visitante
Precondiciones	Situarse en página del listado de cursos por Nombre.
Flujo Básico de Datos	1.- Se traslada a la página correspondiente. 1.1.- En la caja de búsqueda se ingresará un nombre o indicio. 1.2.- Se cargarán los cursos asociados la palabra ingresada o indicio.
Excepciones	Error Interno (La petición al servidor ha tardado demasiado)

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-23: Detalle de caso de uso V-002

ID Caso de Uso	V-002
Nombre de Caso de Uso	Buscar Cursos por Nivel.
Descripción	El Usuario Visitante podrá realizar una búsqueda por nivel.
Actores	Usuario Visitante
Precondiciones	Situarse en página del listado de cursos por Nivel.
Flujo Básico de Datos	1.- Se traslada a la página correspondiente. 1.1.- En el panel de la izquierda seleccionar un nivel. 1.2.- Se cargarán los cursos asociados al nivel seleccionado.
Excepciones	Error Interno (La petición al servidor ha tardado demasiado)

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-24: Detalle de caso de uso V-003

ID Caso de Uso	V-003
Nombre de Caso de Uso	Ver Docentes.
Descripción	El Usuario Visitante podrá Visualizar todos los docentes existentes.
Actores	Usuario Visitante
Precondiciones	Situarse en página Ver todos los docentes.
Flujo Básico de Datos	1.- Se traslada a la página correspondiente. 2.- Se despliega una lista que consta de todos los docentes del sistema con estado activo.
Excepciones	Error Interno (La petición al servidor ha tardado demasiado)

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-25: Detalle de caso de uso V-004

ID Caso de Uso	V-004
Nombre de Caso de Uso	Ver Docente Específico.
Descripción	El Usuario Visitante podrá Visualizar la información de un docente específico y sus cursos.
Actores	Usuario Visitante

Precondiciones	Seleccionar un docente específico. El docente tendrá estado activo.
Flujo Básico de Datos	1.- Se traslada a la página correspondiente. 2.- Se despliega la información del docente y a continuación un listado de cursos que dicta.
Excepciones	Error Interno (La petición al servidor ha tardado demasiado)

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-26: Detalle de caso de uso V-005

ID Caso de Uso	V-005
Nombre de Caso de Uso	Ver Cursos
Descripción	El Usuario Visitante podrá Visualizar una lista de cursos según parámetro: más vistos, nuevos.
Actores	Usuario Visitante
Precondiciones	Situarse en la página Home y seleccionar un parámetro de visualización de cursos.
Flujo Básico de Datos	1.- Se traslada a la página correspondiente. 2.- Se despliega una lista de cursos según parámetro indicado. 3.- En caso de cursos más vistos, se visualizará en orden descendente. 4.- En caso de cursos nuevos se visualizará de fecha reciente hacia atrás.
Excepciones	Error Interno (La petición al servidor ha tardado demasiado)

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-27: Detalle de caso de uso V-006

ID Caso de Uso	V-006
Nombre de Caso de Uso	Ver Curso específico
Descripción	El Usuario Visitante podrá Visualizar a detalle un curso específico.
Actores	Usuario Visitante
Precondiciones	Seleccionar un curso.
Flujo Básico de Datos	1.- Se traslada a la página correspondiente. 2.- Se despliega un menú vertical con la información y recursos que tiene un curso. 3.- Según se seleccione en el menú los ítems se mostrará la información como, por ejemplo, información básica, silabo, lecturas, calendario, galería de imágenes, capítulos, temas actividades, exámenes y recursos digitales expuestos.
Excepciones	Error Interno (La petición al servidor ha tardado demasiado)
ID Caso de Uso	V-001
Nombre de Caso de Uso	Buscar Cursos por Nombre.
Descripción	El Usuario Visitante podrá realizar una búsqueda por nombre.

Actores	Usuario Visitante
Precondiciones	Situarse en página del listado de cursos por Nombre.
Flujo Básico de Datos	1.- Se traslada a la página correspondiente. 1.1.- En la caja de búsqueda se ingresará un nombre o indicio. 1.2.- Se cargarán los cursos asociados la palabra ingresada o indicio.
Excepciones	Error Interno (La petición al servidor ha tardado demasiado)

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-28: Detalle de caso de uso V-007

ID Caso de Uso	V-007
Nombre de Caso de Uso	Login.
Descripción	El Usuario Visitante podrá realizar un proceso de autenticación.
Actores	Usuario Visitante
Precondiciones	Situarse en página de login mediante el menú (opción Iniciar Sesión).
Flujo Básico de Datos	1.- Se traslada a la página correspondiente. 1.1.- Se ingresa usuario y contraseña. 1.2.- Se acciona el botón iniciar sesión. 1.3.- En caso de ser una autenticación con éxito, se re direccionara a su página Home correspondiente según rol que posea el usuario en cuestión. 1.4. Caso contrario se mantendrá en la misma página y se emitirá el mensaje de credenciales incorrectas.
Excepciones	Error Interno (La petición al servidor ha tardado demasiado)

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

2.1.8 Diagrama de Clases

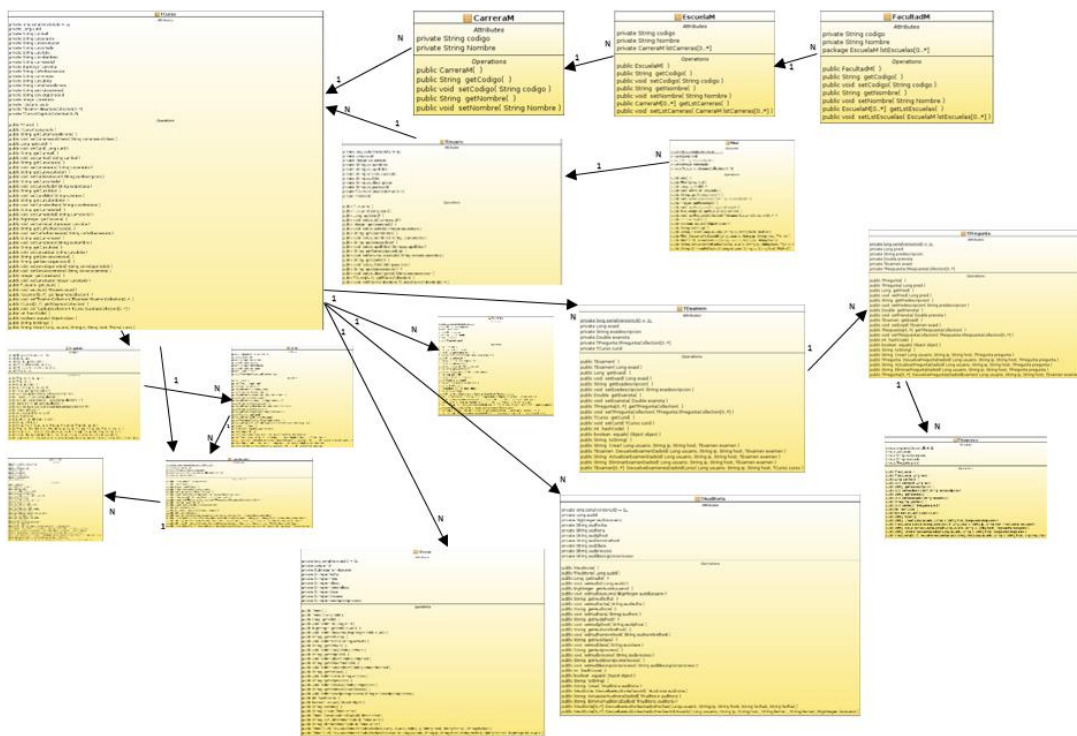


Figura 2-5: Diagrama de Clases.

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-29: Detalle de Clases, Métodos y Atributos

Nombre de la clase	Atributos	Métodos	Función
CarreraM	String codigo; String Nombre;	String getCodigo() String getNombre() Void setCodigo(String codigo) Void setNombre(String Nombre) public CarreraM()	Obtener el código. Obtener el nombre. setear el código setear el nombre Constructor de la clase
EscuelaM	String codigo; String Nombre; List<CarreraM> lstCarreras;	String getCodigo() List<CarreraM> getLstCarreras() String getNombre() Void setCodigo(String codigo) Void setLstCarreras(List<CarreraM> lstCarreras) void setNombre(String Nombre) public EscuelaM()	Obtener Código Obtener todas las carreras Obtener nombre de escuela Setear el código de escuela Setear las carreras Setear el nombre de escuela Constructor de la clase
FacultadM	String codigo; String Nombre; List<EscuelaM> lstEscuelas;	String getCodigo() List<EscuelaM> getLstEscuelas() String getNombre() void setCodigo(String codigo) void setLstEscuelas(List<EscuelaM> lstEscuelas) void setNombre(String Nombre)	Obtener el código de facultad Obtener la lista de escuelas por facultad Obtener el nombre de facultad Setear el código de facultad Setear las escuelas de facultad Setear el nombre de facultad

		FacultadM()	Constructor de la clase facultad
Nodo	String codigo; String descripcion; String codigoPadre;	String getCodigo() String getCodigoPadre() String getDescripcion() void setCodigo(String codigo) void setCodigoPadre(String codigoPadre) void setDescripcion(String descripcion) public Nodo()	Obtiene el código del nodo Obtiene el código del nodo padre Obtiene la descripción del nodo Setea el código del nodo Setea el código del nodo padre Setea la descripción del código Constructor de la clase nodo
ObjetoPregunta	String[] respuestas = new String[4]; Long[] respuestasID = new Long[4]; Boolean[] check; String preguntaDescripcion; Long idPregunta; Long opcion1Id; Long opcion2Id; Long opcion3Id; Long opcion4Id;	void calificar() void calificarConPenalizacion(List<TRespuesta> respuestas, Double valorItem, int numPenalizacion) void calificarSinPenalizacion(List<TRespuesta> respuestas, Double valorItem) Long getIdPregunta() Double getNotaObtenidaPregunta() String getOpcion1Descripcion() Long getOpcion1Id() String getOpcion2Descripcion()	Califica la pregunta Califica con penalización una pregunta Califica sin penalización una pregunta Devuelve el id de pregunta Devuelve la nota de una pregunta Devuelve la descripción de la opción 1 Devuelve el id de la opción 1

String opcion1Descripcion; String opcion2Descripcion; String opcion3Descripcion; String opcion4Descripcion; boolean opcion1Check; boolean opcion2Check; boolean opcion3Check; boolean opcion4Check; Double ValorPregunta; Double NotaObtenidaPregunta;	Long getOpcion2Id () String getOpcion3Descripcion() Long getOpcion3Id() String getOpcion4Descripcion() Long getOpcion4Id() String getPreguntaDescripcion() Double getValorPregunta() boolean isOpcion1Check() boolean isOpcion2Check() boolean isOpcion3Check() boolean isOpcion4Check() void llenarObjeto(TPregunta Pregunta, List<TRespuesta> IstRespuestas, List<TPregunta> preguntas) void llenarVectorCheck() int numeroCorrectasBase(List<TRespuesta> lst)	Devuelve la descripción de la opción 2 Devuelve el id de la opción 2 Devuelve la descripción de la opción 3 Devuelve el id de la opción 3 Devuelve la descripción de la opción 4 Devuelve el id de la opción 4 Devuelve la descripción de la pregunta Devuelve el valor (nota) de la pregunta Devuelve si la opción 1 fue chequeada Devuelve si la opción 2 fue chequeada Devuelve si la opción 3 fue chequeada Devuelve si la opción 4 fue chequeada Llena un objeto pregunta con sus respuestas Verifica si el vector de respuestas se encuentra lleno Devuelve la cantidad de respuestas correctas
--	--	---

		int	numeroCorrectasRespondidas()	Devuelve las preguntas correctas contestadas
		void	setearVariables()	Setea las variables de respuesta
		void	setIdPregunta(Long idPregunta)	Setea el id de la pregunta
		void	setNotaObtenidaPregunta(Double NotaObtenidaPregunta)	Setea la nota obtenida a la pregunta
		void	setOpcion1Check(boolean opcion1Check)	Setea la opción 1 como true o false
		void	setOpcion1Descripcion(String opcion1Descripcion)	Setea la descripción de la respuesta 1
		void	setOpcion1Id(Long opcion1Id)	Setea el id de la opción 1
		void	setOpcion2Check(boolean opcion2Check)	Setea la opción 2 como true o false
		void	setOpcion2Descripcion(String opcion2Descripcion)	Setea la descripción de la respuesta 2
		void	setOpcion2Id(Long opcion2Id)	Setea el id de la opción 2
		void	setOpcion3Check(boolean opcion3Check)	Setea la opción 3 como true o false
		void	setOpcion3Descripcion(String opcion3Descripcion)	Setea la descripción de la respuesta 3
		void	setOpcion3Id(Long opcion3Id)	Setea el id de la opción 3
		void	setOpcion4Check(boolean opcion4Check)	Setea la opción 4 como true o false
		void	setOpcion4Descripcion(String opcion4Descripcion)	Setea la descripción de la respuesta 4
		void	setOpcion4Id(Long opcion4Id)	Setea el id de la opción 4

		void setPreguntaDescripcion(String preguntaDescripcion) void setValorPregunta(Double ValorPregunta) String transformaChek(boolean check) public ObjetoPregunta()	Setea la descripción de una pregunta Setea el valor de una pregunta Trasforma un Check en boolean Constructor del objeto pregunta
PreguntaMasterExamen	String preguntaDescripcion; Long preguntaId; List<TRespuesta> lIstRespuestas;	List<TRespuesta>getLstRespuestas() String getPreguntaDescripcion() Long getPreguntaId() void llenarRespuestas(List<TRespuesta> lIstRespuestass) void setLstRespuestas(List<TRespuesta> lIstRespuestas) void setPreguntaDescripcion(String preguntaDescripcion) void setPreguntaId(Long preguntaId) public PreguntaMasterExamen()	Devuelve una lista de respuestas Devuelve la descripción de la pregunta Devuelve el id de la pregunta Setea la lista de respuestas con una lista entrante Setea la lista de respuestas contestadas por el usuario con otra lista de respuestas Setea la descripción de una pregunta Setea el id de una pregunta Constructor por defecto de la clase
TActividad	Long actid; String actdescripcion; Collection<TRecurso> tRecursoCollection; TTema temid;	String ActualizarActividadDadoId(Long usuario, String ip, String host, TActividad actividad)	Actualiza la actividad dado un objeto actividad Constructor de la clase actividad

		String Crear(Long usuario, String ip, String host, TActividad actividad) TActividad DevuelveActividadDadoId(Long usuario, String ip, String host, TActividad actividad) List<TActividad> devuelveActividadesDadoTema(Long usuario, String ip, String host, TTema tema) String EliminarActividadDadoId(Long usuario, String ip, String host, TActividad actividad) boolean equals(Object object) String getActdescripcion() Long getActid() TTema getTemid() Collection<TRecurso> getTRecursoCollection() void setActdescripcion(String actdescripcion) void setActid(Long actid) void setTemid(TTema temid) void setTRecursoCollection(Collection<TRecurso> tRecursoCollection)	Crea actividad dado un objeto actividad Devuelve actividad dado un objeto actividad Devuelve lista de actividades dado un objeto tema Elimina una actividad dado un objeto actividad Devuelve un true o false dado un object Devuelve la descripción de una actividad Devuelve el id de una actividad Devuelve el tema de una actividad Devuelve una colección de objetos recurso Setea la descripción de una actividad Setea el id de una actividad Setea el tema de una actividad Setea una lista de recursos dad una lista de recursos para actividad
--	--	---	---

		public TActividad()	Constructor de la clase actividad
TAuditoria	Long audid; BigInteger audidusuario; String audfecha; String audhora; String audiphost; String audnombrehost; String audclase; String audproceso; String auddescripcionproceso;	String ActualizarAuditoriaDadoId(TAuditoria auditoria) String Crear(TAuditoria auditoria) TAuditoria DevuelveAuditoriaDadoId(TAuditoria auditoria) List<TAuditoria> DevuelveAuditoriasDado2Fechas(Long usuario, String ip, String host, String fecha1, String fecha2) String EliminarAuditoriaDadoId(TAuditoria auditoria) String getAudclase() String getAuddescripcionproceso() String getAudfecha() String getAudhora() Long getAudid() BigInteger getAudidusuario() String getAudiphost() String getAudnombrehost() String getAudproceso() void setAudclase(String audclase)	Actualiza auditoría dado un objeto auditoría Crea auditoría dado objeto Devuelve una auditoría dado objeto auditoría Devuelve una lista de auditorías dado dos fechas Elimina auditoría dado un objeto Devuelve la clase a auditar Devuelve el proceso que se realizó Devuelve la fecha de auditoría Devuelve la hora de auditoría Devuelve el id de auditoría Devuelve el id de usuario auditado Devuelve la ip del host auditado Devuelve el nombre del host auditado Devuelve el método auditado Setea la clase a auditar

		void setAuddescripcionproceso(String auddescripcionproceso) void setAudfecha(String audfecha) void setAudhora(String audhora) void setAudid(Long audid) void setAudidusuario(BigInteger audidusuario) void setAudiphost(String audiphost) void setAudnombrehost(String audnombrehost) void setAudproceso(String audproceso) public T Auditoria()	Setea la descripción del proceso Setea la fecha de la auditoría Setea la hora de la auditoría Setea el id de la auditoría Setea el id del usuario auditado Setea la ip del host auditado Setea el nombre de host auditado Setea el proceso invocado a auditar Constructor de la clase
TCapitulo	Long capid; String capdescripcion; Collection<TTema> tTemaCollection; Collection<TLectura> tLecturaCollection; TCurso curid;	String ActualizarCapituloDadoId(Long usuario, String ip, String host, TCapitulo capitulo) String Crear(Long usuario, String ip, String host, TCapitulo capitulo) TCapitulo DevuelveCapituloDadoId(Long usuario, String ip, String host, TCapitulo capitulo) List<TCapitulo> DevuelveCapitulosDadoCursoUsuario(Long usuario, String ip, String host, TCurso curso) String EliminarCapituloDadoId(Long usuario, String ip, String host, TCapitulo capitulo) String getCapdescripcion() Long getCapid()	Actualiza un capítulo dado un objeto Crea un capítulo dado un objeto Devuelve un capítulo dado objeto Devuelve una lista de capítulos dado un objeto curso Elimina un capítulo dado objeto Devuelve la descripción de un capítulo Devuelve el id de un capítulo

		TCurso getCurid() Collection<TLectura> getTLecturaCollection() Collection<TTema> getTTemaCollection() void setCapdescripcion(String capdescripcion) void setCapid(Long capid) void setCurid(TCurso curid) void setTLecturaCollection(Collection<TLectura> tLecturaCollection) void setTTemaCollection(Collection<TTema> tTemaCollection) public TCapirulo()	Devuelve el id de un curso Devuelve las lecturas de un capítulo Devuelve los temas de un capítulo Setea la descripción de un capítulo Setea el id de un capítulo Setea el id de un curso Setea la lista de lecturas de un capítulo dado una lista de lecturas Setea una lista de temas de un capítulo dado una lista de temas Constructor por defecto de la clase
TCurso	Long curid; String curnivel; String curparalelo; String curdescripcion; String curportada; String cursilabo; String curcalendario; String curmaterial; BigInteger curvisita; String curfechacreacion;	String ActualizarCursoDadoId(Long usuario, String ip, String host, TCurso curso) String Crear(Long usuario, String ip, String host, TCurso curso) TCurso DevuelveCursoDadoId(Long usuario, String ip, String host, TCurso curso) List<TCurso> DevuelveCursos(Long usuario, String ip, String host) List<TCurso> DevuelveCursosDadoEstadoUsuarioCursoEstadoYNivel(Long usuario, String ip, String host, String nivel, Integer estadousuario, Integer estadocurso)	Actualiza curso dado objeto Crea curso dado objeto Devuelve curso dado objeto Devuelve todos los cursos Devuelve todos los cursos por estado de usuario curso y nivel

	String curnombre; String cursubida; String curtamaniofichero; String serviciocarreraid; String servicioperiodoid; Integer curestado; TUsuario usuid; Collection<TExamen> tExamenCollection; Collection<TCapitulo> tCapituloCollection;	List<TCurso> DevuelveCursosDadoEstadoUsuarioPormasActuales(Long usuario, String ip, String host, Integer estadousuario, Integer curestado) List<TCurso> DevuelveCursosDadoEstadoUsuarioPorvisitaDesc(Long usuario, String ip, String host, Integer estadousuario, Integer curestado) List<TCurso> DevuelveCursosDadoEstadoYCarrera(Long usuario, String ip, String host, Integer estado, Long carrera) List<TCurso> DevuelveCursosDadoEstadoYDocente(Long usuarioid, String ip, String host, Integer estado, TUsuario usuario) List<TCurso> DevuelveCursosDadoEstadoyUsuarioEstado(Long usuario, String ip, String host, Integer curestado, Integer usuestado) List<TCurso> DevuelveCursosDadoIdUsuarioYIdCarrera(Long usuarioid, String ip, String host, TUsuario usuario, String idcarrera) List<TLectura> DevuelveLecturasDadoIdCurso(Long usuario, String ip, String host, Integer idcurso) List<TRecurso> DevuelveRecursosDadoIdCurso(Long usuario, String ip, String host, Long curso) List<TTema> DevuelveTemasDadoIdCurso(Long usuario, String ip, String host, Long curso) List<TCurso> devuelveTopCursosNuevosDadoEstadoUsuarioYEstadoCurso(Long usuario, String ip, String host, Integer top, Integer estadousuario, Integer estadocurso) List<TCurso> DevuelveTopCursosVisitadosDadoEstadoUsuarioYCursoEstado(Long usuario, String ip, String host, Integer top, Integer estadousuario, Integer estadocurso)	Devuelve una lista de cursos más actuales Devuelve cursos dado estado de usuario y ordenados por visita descendentemente Devuelve cursos dado estado y carrera Devuelve cursos de un docente dado estado Devuelve cursos dado estado de cursos y docente Devuelve cursos dado usuario y carrera Devuelve las lecturas de un curso Devuelve los recursos de un curso Devuelve los temas de un curso dado id Devuelve los cursos más nuevos dado estado de usuario y curso Devuelve los cursos más visitados dado estado de usuario y curso
--	--	--	---

		String EliminarCursoDadoId(Long usuario, String ip, String host, TCurso curso)	Elimina un curso dado objeto
		String getCurcalendario()	Devuelve el calendario del curso
		String getCurdescripcion()	Devuelve la descripción del curso
		Integer getCurestado()	Devuelve el estado del curso
		String getCurfechacreacion()	Devuelve la fecha de un curso
		Long getCurid()	Devuelve el id de un curso
		String getCurmaterial()	Devuelve el material de un curso
		String getCurnivel()	Devuelve el nivel de un curso
		String getCurnombre()	Devuelve el nombre de un curso
		String getCurparalelo()	Devuelve el paralelo de un curso
		String getCurportada()	Devuelve la portada de un curso
		String getCursilabo()	Devuelve el sílabo de un curso
		String getCursubida()	Devuelve la subida de un curso
		String getCurtamaniofichero()	Devuelve el tamaño_fichero de un curso
		BigInteger getCurvisita()	Devuelve las visitas de un curso
		String getServiciocarreraid()	Devuelve el id de carrera
		String getServicioperiodoid()	Devuelve el id de período

		Collection<TCapitulo> getTCapituloCollection() Collection<TExamen> getTExamenCollection() TUsuario getUsuid() void setCurcalendario(String curcalendario) void setCurdescripcion(String curdescripcion) void setCurestado(Integer curestado) void setCurfechacreacion(String curfechacreacion) void setCurid(Long curid) void setCurmaterial(String curmaterial) void setCurnivel(String curnivel) void setCurnombre(String curnombre) void setCurparalelo(String curparalelo) void setCurportada(String curportada) void setCursilabo(String cursilabo) void setCursubida(String cursubida) void setCurtamanoifichero(String curtamanoifichero) void setCurvisita(BigInteger curvisita) void setServiciocarreraid(String serviciocarreraid) void setServicioperiodoid(String servicioperiodoid) void setTCapituloCollection(Collection<TCapitulo> tCapituloCollection) void setTExamenCollection(Collection<TExamen> tExamenCollection)	Devuelve los capítulos de un curso Devuelve los exámenes de un curso Setea el calendario de un curso Setea la descripción de un curso Setea el estado de un curso Setea la fecha de un curso Setea el id de un curso Setea el material de un curso Setea el nivel de un curso Setea el nombre de un curso Setea el paralelo de un curso Setea la portada de un curso Setea el silabo de un curso Setea la subida de un curso Setea el tamaño de un curso Setea la visita de un curso Setea el id de carrera Seta el id de período Setea los capítulos de un curso Setea los exámenes de un curso
--	--	--	---

		void setUsuid(TUsuario usuid) public TCurso()	Setea el usuario del curso Constructor de la clase
TError	Long errid; BigInteger erridusuario; String errfecha; String errhora; String erriphost; String errnombrehost; String errclase; String errproceso; String errdescripcionproceso;	String ActualizarErrorDadoId(TError error) String Crear(TError error) TError DevuelveErrorDadoId(TError error) List<TError> DevuelveErroresDado2Fechas(Long usuario, String ip, String host, String fecha1, String fecha2) List<TError> DevuelveErroresDado2FechasYIdUsuario(Long usuario, String ip, String host, String fecha1, String fecha2, BigInteger idusuario) String EliminarErrorDadoId(TError error) String getErrclase() String getErrdescripcionproceso() String getErrfecha() String getErrhora() Long getErrid() BigInteger getErridusuario() String getErriphost() String getErrnombrehost() String getErrproceso() void setErrclase(String errclase) void setErrdescripcionproceso(String errdescripcionproceso) void setErrfecha(String errfecha) void setErrhora(String errhora) void setErrid(Long errid)	Actualizar error dado objeto Crear error dado objeto Devolver error dado objeto Devuelve lista de errores dado dos fechas Devuelve lista de errores dado dos fechas y usuario Elimina error dado objeto Devuelve la clase a registrar error Devuelve la descripción del proceso erróneo Devuelve la fecha del sistema Devuelve la hora del sistema Devuelve el id del error Devuelve el id de usuario del error Devuelve la ip del host Devuelve el nombre del host Devuelve el proceso Setea la clase de error Setea la descripción del error Setea a fecha del error Setea la hora del error Setea el id del error

		void setErridusuario(BigInteger erridusuario) void setErriphost(String erriphost) void setErrnombrehost(String errnombrehost) void setErrproceso(String errproceso) public TError()	Setea el id del usuario Setea la ip del host Setea el nombre del host Setea el proceso de error Constructor de la clase por defecto
TExamen	Long exaid; String exadescripcion; Double exanota; Collection<TPregunta> tPreguntaCollection; TCurso curid;	String ActualizarExamenDadoId(Long usuario, String ip, String host, TExamen examen) String Crear(Long usuario, String ip, String host, TExamen examen) TExamenDevuelveExamenDadoId(Long usuario, String ip, String host, TExamen examen) List<TExamen> DevuelveExamenesDadoIdCurso(Long usuario, String ip, String host, TCurso curso) String EliminarExamenDadoId(Long usuario, String ip, String host, TExamen examen) TCurso getCurid() String getExadescripcion() Long getExaid() Double getExanota() Collection<TPregunta> getTPreguntaCollection() void setCurid(TCurso curid) void setExadescripcion(String exadescripcion) void setExaid(Long exaid) void setExanota(Double exanota)	Actualiza un examen dado objeto Crea examen dado objeto Devuelve examen dado objeto Devuelve lista de examen dado id del curso Elimina examen dado objeto Devuelve el id del curso Devuelve la descripción del examen Devuelve el id del examen Devuelve la nota del examen Devuelve las preguntas de el examen Setea el id de un curso Setea la descripción de un examen Setea el id de un examen Setea la nota de un examen

		void setTPreguntaCollection(Collection<TPregunta> tPreguntaCollection) Public TExamen()	Setea la lista de preguntas de un examen Constructor por defecto de la clase
TLectura	String lecdescripcion; String lecurl; String lecruta; String lectamano; Long lecid; TCapitulo capid;	String ActualizarLecturaDadoId(Long usuario, String ip, String host, TLectura lectura) String Crear(Long usuario, String ip, String host, TLectura lectura) TLectura DevuelveLecturaDadoId(Long usuario, String ip, String host, TLectura lectura) List<TLectura> DevuelveLecturasDadoIdCapitulo(Long usuario, String ip, String host, Long idcapitulo) List<TLectura> DevuelveLecturasDadoIdCapitulo(Long usuario, String ip, String host, TCapitulo capitulo) List<TLectura> DevuelveLecturasDadoIdCurso(Long usuario, String ip, String host, Long idcurso) Long DevuelveSumaTamanoLecturasDadoIdcurso(TCurso curso) Long DevuelveSumaTamanoLecturasDadoIdcursoYLectura(TCurso curso, TLectura lectu) String EliminarLecturaDadoId(Long usuario, String ip, String host, TLectura lectura) TCapitulo getCapid() String getLecdescripcion() Long getLecid()	Actualiza lectura dado objeto Crea lectura dado objeto Devuelve lectura dado objeto Devuelve lista de lecturas dado idcapítulo Devuelve lista de lecturas dado objeto capítulo Devuelve lista de lecturas dado id curso Devuelve la sumatoria de las lecturas de un curso dado id curso Devuelve la sumatoria de las lecturas de un curso dado id curso y nueva lectura Elimina Lectura dado objeto Devuelve el id del capítulo Devuelve la descripción de la lectura Devuelve el id de una lectura

		String getLecruta() String getLectamanio() String getLecurl() void setCapid(TCapitulo capid) void setLecdescripcion(String lecdescripcion) void setLecid(Long lecid) void setLecruta(String lecruta) void setLectamanio(String lectamanio) void setLecurl(String lecurl) Public TLectura ()	Devuelve la ruta de la lectura Devuelve el tamaño de la lectura Devuelve la url de una lectura Setea el id de un capítulo Setea la descripción de la lectura Setea el id de la lectura Setea la ruta de una lectura Setea el tamaño de lectura Setea la url de la lectura Constructor por defecto de la lectura
TPregunta	Long preid; String predescripcion; Double prenota; TExamen exaid; Collection<TRespuesta> tRespuestaCollection;	String ActualizarPreguntaDadoId(Long usuario, String ip, String host, TPregunta pregunta) String Crear(Long usuario, String ip, String host, TPregunta pregunta) TPregunta DevuelvePreguntaDadoId(Long usuario, String ip, String host, TPregunta pregunta) List<TPregunta> DevuelvePreguntasDadoIdExamen(Long usuario, String ip, String host, TExamen examen) String EliminarPreguntaDadoId(Long usuario, String ip, String host, TPregunta pregunta) TExamengetExaid() String getPredescripcion() Long getPreid() Double getPrenota()	Actualiza pregunta dado objeto pregunta Crea pregunta dado objeto Devuelve pregunta dado objeto Devuelve preguntas dado objeto examen Elimina pregunta dado objeto Devuelve el id del examen Devuelve descripción de pregunta Devuelve id de pregunta Devuelve nota de pregunta

		Collection<TRespuesta> getTRespuestaCollection() void setExaid(TExamen exaid) void setPredescripcion(String predescripcion) void setPreid(Long preid) void setPrenota(Double prenota) void setTRespuestaCollection(Collection<TRespuesta> tRespuestaCollection) public TExamen()	Las respuestas de un examen Setea examen dado objeto Setea descripción de la pregunta Setea id de pregunta Setea nota de pregunta Setea las respuestas de un examen Constructor por defecto
TRecurso	Long recid; String recnombre; String rectipo; String recurId; String recruta; String rectamano; TActividad actid;	String ActualizarRecursoDadoId(Long usuario, String ip, String host, TRecurso recurso) String Crear(Long usuario, String ip, String host, TRecurso recurso) TRecurso DevuelveRecursoDadoId(Long usuario, String ip, String host, TRecurso recurso) List<TRecurso> devuelveRecursosDadoIdActividad(Long usuario, String ip, String host, TActividad actividad) Long DevuelveSumaTamanoRecursosDadoIdcurso(TCurso curso) Long DevuelveSumaTamanoRecursosDadoIdcursoYRecurso(TCurso curso, TRecurso recurso) String EliminarRecursoDadoId(Long usuario, String ip, String host, TRecurso recurso) TActividad getActid() Long getRecid() String getRecnombre() String getRecruta()	Actualiza recurso dado objeto Crea recurso dado objeto Devuelve recurso dado objeto Devuelve recursos dado objeto actividad Devuelve sumatoria de recursos dado objeto curso Devuelve sumatoria de recursos dado objeto curso y recurso Elimina un recurso dado objeto Devuelve id de actividad Devuelve id de recurso Devuelve nombre de recurso Devuelve ruta de recurso

		String getRectamanio() String getRectipo() String getRecurl() void setActid(TActividad actid) void setRecid(Long recid) void setRecnombre(String recnombre) void setRecruta(String recruta) void setRectamanio(String rectamanio) void setRectipo(String rectipo) void setRecurl(String recurl) Public TRecurso()	Devuelve tamaño de recurso Devuelve tipo de recurso Devuelve url de recurso Setea actividad dado objeto Setea recursoid Setea recurso nombre Setea la ruta del recurso Setea el tamaño de un curso Setea tipo de recurso Setea la url de un recurso Constructor por defecto de la clase
TRespuesta	Long resid; String resdescripcion; String resestado; TPregunta preid;	String ActualizarRespuestaDadoId(Long usuario, String ip, String host, TRespuesta respuesta) String Crear(Long usuario, String ip, String host, TRespuesta respuesta) TRespuesta DevuelveRespuestaDadoId(Long usuario, String ip, String host, TRespuesta respuesta) List<TRespuesta> DevuelveRespuestasDadoIdPregunta(Long usuario, String ip, String host, long pregunta) String EliminarRespuestaDadoId(Long usuario, String ip, String host, TRespuesta respuesta) TPregunta getPreid() String getResdescripcion() String getResestado() Long getResid()	Actualiza respuesta dado objeto Crea respuesta dado objeto Devuelve respuesta dado id Devuelve respuestas dado id pregunta Elimina respuesta dado objeto Devuelve el id de la pregunta Devuelve la descripción de la respuesta Devuelve el estado de la respuesta Devuelve el id de la respuesta

		void setPreid(TPregunta preid) void setResdescripcion(String resdescripcion) void setResestado(String resestado) void setResid(Long resid) Public TRespuesta()	Setea el id de la pregunta Setea la descripción de la respuesta Setea el estado de la respuesta Setea el id de la respuesta Constructor de la clase respuesta
TRol	Long rolid; String roldescripcion; Integer rolestado; tUsuarioCollection;	String ActualizarRolDadoId(Long usuario, String ip, String host, TRol rol) String Crear(Long usuario, String ip, String host, TRol rol) TRol DevuelveRolDadoId(Long usuario, String ip, String host, TRol rol) List<TRol> DevuelveRoles(Long usuario, String ip, String host) String EliminarRolDadoId(Long usuario, String ip, String host, TRol rol) String getRoldescripcion() Integer getRolestado() Long getRolid() Collection<TUsuario> getTUsuarioCollection() void setRoldescripcion(String roldescripcion) void setRolestado(Integer rolestado) void setRolid(Long rolid) void setTUsuarioCollection(Collection<TUsuario> tUsuarioCollection) Public TRol ()	Actualiza el rol dado objeto Crea rol dado objeto Devuelve rol dado objeto Devuelve roles Elimina rol dado objeto Devuelve la descripción del rol Devuelve el estado del rol Devuelve el id del rol Devuelve la lista de usuarios de rol Setea la descripción del rol Setea el estado del rol Setea el id del rol Setea la colección de usuario del rol Constructor por defecto de la clase
TTema	Long temid; String temresumen;	String ActualizarTemaDadoId(Long usuario, String ip, String host, TTema tema)	Actualiza tema dado objeto

	String temimagen; String temdescripcion; String temtamano; TCapitulo capid; Collection<TActividad> tActividadCollection;	String Crear(Long usuario, String ip, String host, TTema tema) Long DevuelveSumaTamanoTemasDadoIdcurso(TCurso curso) Long DevuelveSumaTamanoTemasDadoIdcursoYTema(TCurso curso, TTema tema) TTema DevuelveTemaDadoId(Long usuario, String ip, String host, TTema tema) List<TTema> DevuelveTemasDadoCapituloUsuario(Long usuario, String ip, String host, TCapitulo capitulo) String EliminarTemaDadoId(Long usuario, String ip, String host, TTema tema) TCapitulo getCapid() Collection<TActividad> getTActividadCollection() String getTemdescripcion() Long getTemid() String getTemimagen() String getTemresumen() String getTemptamano() void setCapid(TCapitulo capid) void setTActividadCollection(Collection<TActividad> tActividadCollection) void setTemdescripcion(String temdescripcion) void setTemid(Long temid) void setTemimagen(String temimagen) void setTemresumen(String temresumen) void setTemptamano(String temptamano) Public TTema()	Crea tema dado objeto Devuelve sumatoria de temas dado objeto curso Devuelve sumatoria de temas dado objeto curso y tema Devuelve tema dado objeto Devuelve lista de temas dado objeto capítulo Elimina tema dado objeto Devuelve id del capítulo Devuelve actividades de un tema Devuelve descripción de un tema Devuelve id del tema Devuelve imagen del team Devuelve resumen del tema Devuelve tamaño del tema Setea id del capítulo Setea actividades de un tema Setea descripción de un tema Setea id del tema Setea imagen del tema Setea resumen del tema Setea tamaño de la imagen Constructor por defecto de clase
--	---	--	--

TUsuario	Long usuid; String usunombres; String usuapellidos; String serviciusucedula; String usufoto; String usudescpcion; String usupassword; Collection<TCurso> tCursoCollection; TRol rolid;	String ActualizarUsuarioDadoId(Long usuarioid, String ip, String host, TUsuario usuario) String Crear(Long usuarioid, String ip, String host, TUsuario usuario) TUsuario DevuelveUsuarioDadoCedula(Long usuarioid, String ip, String host, String cedula) TUsuario DevuelveUsuarioDadoCedulayPass(Long usuarioid, String ip, String host, String cedula, String password) TUsuario DevuelveUsuarioDadoId(Long usuarioid, String ip, String host, TUsuario usuario) List<TUsuario> DevuelveUsuarios(Long usuarioid, String ip, String host) List<TUsuario> DevuelveUsuariosDadoestado(Long usuarioid, String ip, String host, Integer estado) List<TUsuario> DevuelveUsuariosDadoRolyEstado(Long usuarioid, String ip, String host, Integer estadousuario, Long rolid) String EliminarUsuarioDadoId(Long usuarioid, String ip, String host, TUsuario usuario) TRol getRolid() String getServiciusucedula() Collection<TCurso> getTCursoCollection() String getUsuapellidos() String getUsudescpcion() Integer getUsuestado() String getUsufoto() Long getUsuid()	Actualiza Usuario dado objeto Crea Usuario dado objeto Devuelve Usuario dado cédula Devuelve usuario dado cédula y password Devuelve usuario dado objeto Devuelve usuarios Devuelve usuarios dado estado Devuelve usuarios dad rol y estado Elimina usuario dado objeto Devuelve el rol de un usuario Devuelve servicio cédula Devuelve colección de cursos de usuario Devuelve apellidos del usuario Devuelve descripción del usuario Devuelve estado del usuario Devuelve foto del usuario Devuelve id del usuario
----------	---	---	--

		String getUsunombres() String getUsupassword() void setRolid(TRol rolid) void setServiciosucredula(String serviciosucredula) void setTCursoCollection(Collection<TCurso> tCursoCollection) void setUsuapellidos(String usuapellidos) void setUsudescrpcion(String usudescrpcion) void setUsuestado(Integer usuestado) void setUsufoto(String usufoto) void setUsuid(Long usuid) void setUsunombres(String usunombres) void setUsupassword(String usupassword) Public TUsuario()	Devuelve nombres del usuario Devuelve password del usuario Setea rol del usuario Setea servicio cédula del usuario Setea cursos de un usuario Setea apellidos de un usuario Setea descripción de usuario Setea estado de usuario Setea foto de usuario Setea id de usuario Setea nombres de usuario Setea password de usuario Constructor por defecto de la clase
--	--	---	---

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

2.1.9 Desarrollo

Un sprint es un bloque temporal corto y fijo, cada Sprint esta descrito por una fecha de inicio, fecha de finalización, la estimación de puntos de esfuerzo requerida para cumplir con su desarrollo. En cada sprint se asignó las historias de usuario de acuerdo a la planificación realizada conjuntamente con el product owner correspondientes a un módulo (Izaurre, 2013).

A continuación, se describe brevemente cada SPRINT realizado para el cumplimiento del desarrollo sistema. Como Team Development se realizó la identificación de las necesidades de la empresa, mediante varias reuniones con El vicedecano de la FIE-ESPOCH (Ing. Jorge Huilca), Stakeholder para este proyecto, y el Ing, Javier Romero como usuario docente de dicha empresa.

El sistema fue dividido en 21 sprints con una duración de 1008 horas en total, los mismos que se definieron tomando en cuenta la secuencia de desarrollo y la prioridad de los requerimientos establecida en conjunto por el Stakeholder Ingeniero Jorge Huilca, lo cual permitió presentar resultados y avances del sistema al cliente cada mes.

2.1.9.1 Sprint 1

En el sprint 1 consta la información obtenida en la entrevista con el cliente donde se logró establecer las necesidades que va a satisfacer el sistema, con dicha información se establecieron los requerimientos a desarrollar, la arquitectura y estructura del sistema, la imagen corporativa y los bocetos de interfaz de usuario de acuerdo a la institución.

La ESPOCH cuenta con un servidor para las aplicaciones y las bases de datos de las mismas, por esta razón la arquitectura seleccionada es Cliente/Servidor

Tabla 2-30: HT1 Arquitectura del sistema

HISTORIA TÉCNICA	
Número: HT1	Sprint Asignado: 1
Fecha Inicio: 02/10/15	Fecha Fin: 06/10/15
Tiempo Estimado:	3 días
Responsable:	Jhonny Guevara
Descripción: Como administrador del sistema solicito se defina la arquitectura del sistema.	

Prueba de Aceptación: - Aprobación de la arquitectura del sistema por el administrador del sistema
--

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-31: Actividad HT1

Fecha	Actividad	Puntos estimados
02/10/15 al 06/10/15	Tarea 1: Definición de la arquitectura del sistema. - Definición de la arquitectura del sistema OCW - Documentación	24

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-32: TI de HT1

TAREA DE INGENIERIA	
Sprint: 1	Número de Tarea: 1
Nombre Historia de Usuario: Definir la arquitectura del sistema.	
Nombre de la Tarea: Definición de la arquitectura del sistema	
Tipo de Tarea: Desarrollo	Programador Responsable: Jhonny Guevara
Fecha Inicio: 02/10/15	Fecha Fin: 06/10/15
Descripción: Definir la arquitectura del sistema para el OCW.	
Pruebas de Aceptación: Aprobación de la arquitectura del sistema por parte del administrador.	

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-33:PA de HT1

PRUEBA DE ACEPTACIÓN	
Código: HT1.1	Nombre Historia de Usuario: Definir la arquitectura del sistema.
Nombre de la Prueba: Aprobación de la arquitectura del sistema por parte del administrador.	
Responsable: Jhonny Guevara	Fecha: 09/10/15
Descripción: Se realiza una reunión con el administrador para acordar y revisar la arquitectura del sistema.	
Condiciones de Ejecución:	
<ul style="list-style-type: none"> - Revisar las diferentes arquitecturas de despliegue para los sistemas. 	
Pasos de Ejecución:	
<ul style="list-style-type: none"> - Gestionar una reunión con el administrador del sistema. 	
Resultado Esperado:	
<ul style="list-style-type: none"> - Se aprueba la arquitectura del sistema 	
Evaluación de la Prueba: Satisfactoria.	

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

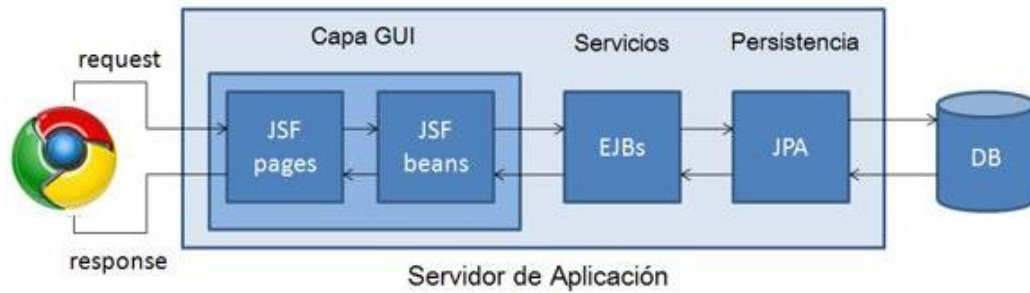


Figura 2-6: Arquitectura del Sistema

Realizado por: Carlos Mejia, Jhonny Guevara, 2016



Figura 2-7: Diagrama de despliegue

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-34: Estándar de Interfaz Gráfica.

HISTORIA TÉCNICA	
Número: HT2	Sprint Asignado: 1
Fecha Inicio: 07/10/15	Fecha Fin: 09/10/15
Tiempo Estimado:	3 días
Responsable:	Carlos Mejía
Descripción: Como administrador del sistema solicito se defina el estándar para la interfaz gráfica de usuario.	
Prueba de Aceptación:	
<ul style="list-style-type: none"> - Aprobación del estándar para interfaz gráfica de usuario por el administrador del sistema 	

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-35: Actividad de HT2.

Fecha	Actividad	Puntos estimados
07/10/15 al 09/10/15	Tarea 1: Definición del estándar para la interfaz gráfica del sistema. - Definición del estándar para la interfaz gráfica del sistema - Documentación	24

Realizado por: Carlos Mejía, Jhonny Guevara, 2016

Tabla 2-36: Actividad de HT2.

TAREA DE INGENIERIA	
Sprint: 1	Número de Tarea: 1
Nombre Historia de Usuario: Definir el estándar para la interfaz gráfica del sistema.	
Nombre de la Tarea: Definición del estándar para la interfaz gráfica del sistema	
Tipo de Tarea: Desarrollo	Programador Responsable: Carlos Mejía
Fecha Inicio: 07/10/15	Fecha Fin: 09/10/15
Descripción: Definir el estándar para la interfaz gráfica del sistema.	
Pruebas de Aceptación:	
- Aprobación del estándar para la interfaz gráfica por parte del administrador.	

Realizado por: Carlos Mejía, Jhonny Guevara, 2016

Tabla 2-37: PA de HT2.

PRUEBA DE ACEPTACIÓN	
Código: HT2.1	Nombre Historia de Usuario: Definir el estándar para la interfaz gráfica de usuario.
Nombre de la Prueba: Aprobación del estándar para la interfaz gráfica de usuario	
Responsable: Carlos Mejía	Fecha: 09/10/15
Descripción: Se realiza una reunión con el administrador para acordar y revisar el estándar para la interfaz gráfica de usuario.	
Condiciones de Ejecución:	
<ul style="list-style-type: none"> - Revisar los modelos de interfaces destinados para aplicaciones en la ESPOCH. - Definir el estándar para la interfaz gráfica de usuario. 	
Pasos de Ejecución:	
<ul style="list-style-type: none"> - Gestionar una reunión con el administrador del sistema. - Acordar y aprobar el estándar para la interfaz gráfica de usuario. 	
Resultado Esperado:	
- Se aprueba el estándar para la interfaz gráfica de usuario	
Evaluación de la Prueba:	
Satisfactoria.	

Realizado Por: Carlos Mejía, Jhonny Guevara 2016

En reuniones con el Stakeholder y el Team Development se ha llegado al acuerdo de mantener los colores de la empresa es decir predominan el color blanco y el color azul y tipo de fuente Arial. La (tabla 13) describe tonalidades y fuentes en el diseño.

Tabla 2-38: Descripción de componentes de Interfaz

COMPONENTE	LOCALIZACIÓN	COLOR FONDO /FUENTE
Menú	Header	azul / Blanco
Información	Footer	azul / Negro
Pestañas	Body	Blanco / Negro
Botones	Body, justificado a la derecha	Azul / Blanco
Mensaje de confirmación	Body, centrado y justificado a la derecha	Azul / Blanco
Mensaje de error	Body	Rojo / Blanco

Realizado por: Carlos Mejía, Jhonny Guevara, 2016



Figura 2-8: Página de Inicio

Realizado por: Carlos Mejía, Jhonny Guevara, 2016



Figura 2-9: Página de Inicio Docente

Realizado por: Carlos Mejia, Jhonny Guevara, 2016



Figura 2-10: Página de Inicio Administrador

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

ESTÁNDAR PARA LA INTERFAZ GRÁFICA DE USUARIO



Figura 2-11: Página de inicio

Realizado por: Carlos Mejia, Jhonny Guevara, 2016



Figura 2-12: Página de Login

Realizado por: Carlos Mejia, Jhonny Guevara, 2016



Figura 2-13: Menú Búsqueda de Cursos

Realizado por: Carlos Mejia, Jhonny Guevara, 2016



Figura 2-14: Búsqueda de Cursos por Nivel

Realizado por: Carlos Mejia, Jhonny Guevara, 2016



Figura 2-15: Búsqueda de Cursos por Nombre

Realizado por: Carlos Mejia, Jhonny Guevara, 2016



Figura 2-16: Página de Curso

Realizado por: Carlos Mejia, Jhonny Guevara, 2016



Figura 2-17: Listado de Instructores

Realizado por: Carlos Mejia, Jhonny Guevara, 2016



Figura 2-18: Pagina Docente y sus Cursos

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

SPRINT 2

Tabla 2-39: HT1 Diseñar base de datos

HISTORIA TÉCNICA	
Número: HT3	Sprint Asignado: 2
Fecha Inicio: 12/10/15	Fecha Fin: 15/10/15
Tiempo Estimado:	4 días
Responsable:	Team development
Descripción: Como administrador del sistema solicito se diseñe la base de datos.	
Prueba de Aceptación:	
- El diseño de la base de datos para el sistema OCW deberá ser aprobado por el administrador.	

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-40: Actividad de HT3

Fecha	Actividad	Puntos estimados
12/10/15 al 15/10/15	Tarea 1: Diseño de la Base de Datos. - Diseño Físico de la Base de Datos - Diccionario de Datos - Documentación	32

Realizado por: Carlos Mejía, Jhonny Guevara, 2016

Tabla 2-41: TI de HT3

TAREA DE INGENIERIA	
Sprint: 2	Número de Tarea: 1
Nombre Historia de Usuario: Diseñar la base de datos.	
Nombre de la Tarea: Diseño de la base de datos	
Tipo de Tarea: Desarrollo	Programador Responsable: Carlos Mejía G
Fecha Inicio: 12/10/15	Fecha Fin: 15/10/15
Descripción: Definir los Sprint del sistema.	
Pruebas de Aceptación:	
<ul style="list-style-type: none"> - Aprobación del diseño de la base de datos por parte del administrador. 	

Realizado por: Carlos Mejía, Jhonny Guevara, 2016

Tabla 2-42: PA de HT3

PRUEBA DE ACEPTACIÓN	
Código: HT3.1	Nombre Historia de Usuario: Diseñar la base de datos.
Nombre de la Prueba: Aprobación del diseño de la base de datos por parte del administrador.	
Responsable: Carlos Mejía G	Fecha: 19/10/15
Descripción: Se realiza una reunión con el administrador para acordar y revisar el diseño de la base de datos.	
Condiciones de Ejecución:	
<ul style="list-style-type: none"> - Identificar las entidades que intervendrán en el sistema. - Definir el diseño físico de la base de datos. - Definir el diccionario de datos. 	
Pasos de Ejecución:	
<ul style="list-style-type: none"> - Gestionar una reunión con el administrador del sistema. - Acordar y aprobar el diseño de la base de datos. 	
Resultado Esperado:	
<ul style="list-style-type: none"> - Se define el diseño de la base de datos 	
Evaluación de la Prueba:	
Satisfactoria.	

Realizado por: Carlos Mejía, Jhonny Guevara, 2016

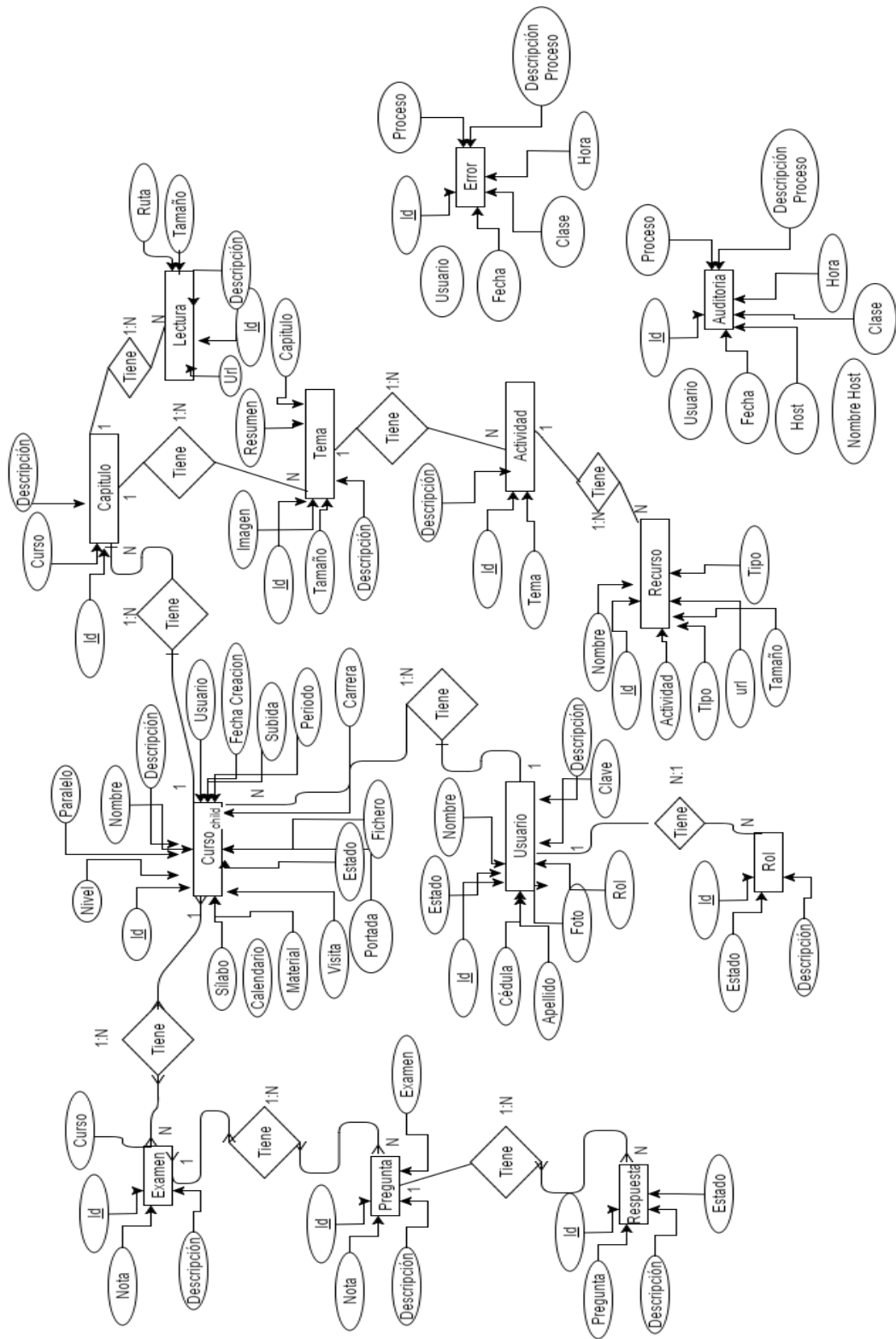


Figura 2-19: DER

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

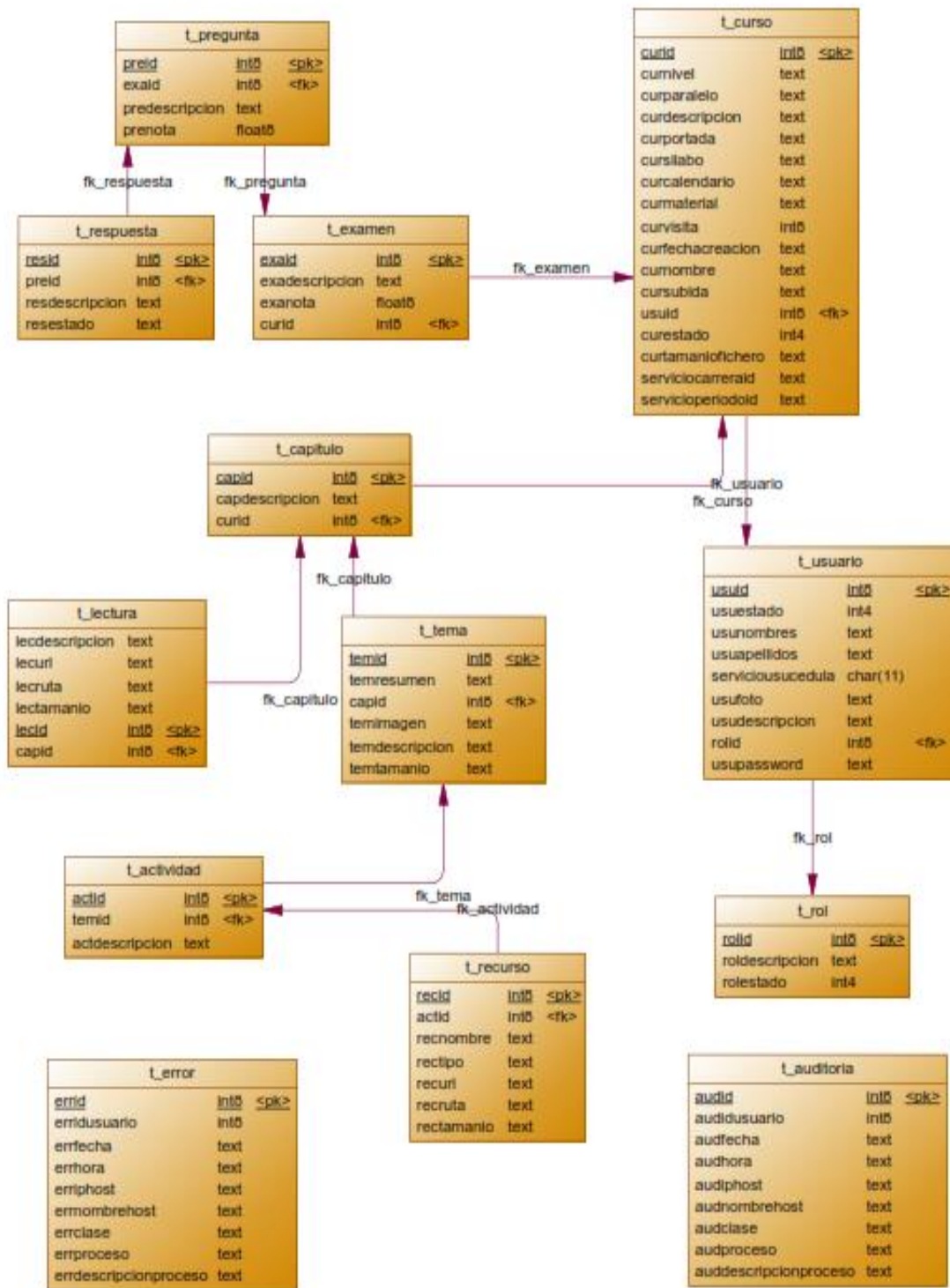


Figura 2-20: Diagrama Físico

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-43: Tabla t_auditoria

TABLA	COLUMNA	TIPO-TAMAÑO	VALIDACION
t_auditoria PK: audid	Audid	Bigserial	NOT NULL
	Audidusuario	Bigint	NOT NULL
	Audfecha	Text	NOT NULL
	Audhora	Text	NOT NULL
	Audiphost	Text	NOT NULL
	Audnombrehost	Text	NOT NULL
	Audclase	Text	NOT NULL
	Audproceso	Text	NOT NULL
	Auddescripcionproceso	Text	NOT NULL

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-44: Tabla t_error

TABLA	COLUMNA	TIPO-TAMAÑO	VALIDACION
t_error PK: errid	Erred	Bigserial	NOT NULL
	erridusuario	Bigint	NOT NULL
	errfecha	Text	NOT NULL
	errhora	Text	NOT NULL
	erriphost	Text	NOT NULL
	errnombrehost	Text	NOT NULL
	errclase	Text	NOT NULL
	errproceso	Text	NOT NULL
	errdescripcionproceso	Text	NOT NULL

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-45: Tabla t_actividad

TABLA	COLUMNA	TIPO-TAMAÑO	VALIDACION
t_actividad	Actid	Bigserial	NOT NULL
PK: actid	Temid	Biginteger	NOT NULL
FK: t_tema (temid)	temdescripcion	Text	NOT NULL

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-46: Tabla t_capitulo

TABLA	COLUMNA	TIPO-TAMAÑO	VALIDACION
t_capítulo	Capid	Bigserial	NOT NULL
PK: capid	Curid	Biginteger	NOT NULL
FK: t_curso (curid)	capdescripcion	Text	NOT NULL

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-47: Tabla t_curso

TABLA	COLUMNA	TIPO-TAMAÑO	VALIDACION
t_curso PK: curid FK: t_usuario (usuid) UK: (curid, usuid)	Curid	Bigserial	NOT NULL
	Curnivel	Text	NOT NULL
	curparalelo	Text	NOT NULL
	curdescripcion	Text	NOT NULL
	curportada	Text	NOT NULL
	Cursilabo	Text	NOT NULL
	curcalendario	Text	NOT NULL
	curmaterial	Text	NOT NULL
	Curvisita	Biginteger	DEFAULT 0
	curfechacreacion	Text	NOT NULL
	curnombre	Text	NOT NULL
	Cursubida	Text	DEFAULT 0
	Usuid	Biginteger	NOT NULL
	serviciocarreraid	Text	NOT NULL
	servicioperiodoid	Text	NOT NULL
curtamanoifichero	Text	DEFAULT 0	
curestado	Integer	DEFAULT 0	

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-48: Tabla t_examen

TABLA	COLUMNA	TIPO-TAMAÑO	VALIDACION
t_examen PK: exaid FK: t_curso (curid) UK: (curid, exaid)	exaid	Bigserial	NOT NULL
	Exadescripcion	Text	NOT NULL
	Exanota	double precisión	NOT NULL
	Curid	Biginteger	NOT NULL

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-49: Tabla t_pregunta

TABLA	COLUMNA	TIPO-TAMAÑO	VALIDACION
t_pregunta PK: preid FK: t_examen (exaid) UK: (preid, exaid)	Preid	BigSerial	NOT NULL
	Exaid	Biginteger	NOT NULL
	Predescripcion	Text	NOT NULL
	Prenota	Double precisión	DEFAULT 0

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-50: Tabla t_lectura

TABLA	COLUMNA	TIPO-TAMAÑO	VALIDACION
t_lectura PK: lecid FK: t_capitulo (capid) UK: (capid, lecdescripcion)	Lecid	Bigserial	NOT NULL
	Capid	BigInteger	NOT NULL
	lecdescripcion	Text	NOT NULL
	Lecurl	Text	NOT NULL
	Lecruta	Text	NOT NULL
	lectamano	Text	DEFAULT 0

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-51: Tabla t_respuesta

TABLA	COLUMNA	TIPO-TAMAÑO	VALIDACION
t_respuesta PK: resid FK: t_pregunta(preid) UK: (preid, resid)	Resid	BigSerial	NOT NULL
	Pried	BigInteger	NOT NULL
	Resdescripcion	Text	NOT NULL
	Resestado	Text	Default 0

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-52: Tabla t_recurso

TABLA	COLUMNA	TIPO-TAMAÑO	VALIDACION
t_recurso PK: recid FK: t_actividad (actid) UK: (actid, recnombre)	recid	Bigserial	NOT NULL
	Rectipo	Text	NOT NULL
	Recnombre	Text	NOT NULL
	Recurl	Text	NOT NULL
	Recruta	Text	NOT NULL
	rectamano	Text	DEFAULT 0
	Actid	Biginteger	NOT NULL

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-53: Tabla t_rol

TABLA	COLUMNA	TIPO-TAMAÑO	VALIDACION
t_rol PK: rolid UK: (roldescripcion)	rolid	Serial	NOT NULL
	Roldescripcion	Text	NOT NULL
	Rolestado	Integer	DEFAULT 0

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-54: Tabla t_tema

TABLA	COLUMNA	TIPO-TAMAÑO	VALIDACION
t_tema PK: temid FK: t_capitulo(capid) UK: (temdescripcion, capid)	temid	Bigserial	NOT NULL
	Temresumen	Text	NOT NULL
	Capid	Biginteger	NOT NULL
	Temimagen	Text	NOT NULL
	Temdescripcion	Text	NOT NULL
	Temtamano	Text	DEFAULT 0

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-55: Tabla t_usuario

TABLA	COLUMNA	TIPO-TAMAÑO	VALIDACION
t_usuario PK: usuid FK: t_rol (rolid)	usuid	Bigserial	NOT NULL
	Usunombres	Text	NOT NULL
	Usupassword	Text	NOT NULL
	Rolid	Integer	NOT NULL
	Uestado	Integer	DEFAULT 0
	Usuapellidos	Text	NOT NULL
	Serviciiousucedula	Character(11)	NOT NULL
	Usufoto	Text	NOT NULL
	Usudescripcion	Text	NOT NULL

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-56: HT4 estándar de Codificación

HISTORIA TÉCNICA	
Número: HT4	Sprint Asignado: 1
Fecha Inicio: 16/10/15	Fecha Fin: 19/10/15
Tiempo Estimado:	2 días
Responsable:	Carlos Mejía
Descripción: Como administrador del sistema solicito se defina el estándar de codificación.	
Prueba de Aceptación:	
- Aprobación del estándar de codificación por el administrador del sistema	

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-57: Actividad de HT4

Fecha	Actividad	Puntos Estimados
16/10/15 al 19/10/15	Tarea 1: Definición del estándar de codificación. - Definición del estándar para la base de datos - Definición del estándar para la codificación en java - Documentación	16

Realizado por: Carlos Mejía, Jhonny Guevara, 2016

Tabla 2-58: TI HT4

TAREA DE INGENIERIA	
Sprint: 2	Número de Tarea: 1
Nombre Historia de Usuario: Definir el estándar de codificación.	
Nombre de la Tarea: Definición del estándar de codificación	
Tipo de Tarea: Desarrollo	Programador Responsable: Carlos Mejía
Fecha Inicio: 16/10/15	Fecha Fin: 19/10/15
Descripción: Definir el estándar de codificación para el OCW.	
Pruebas de Aceptación: - Aprobación del estándar de codificación por parte del administrador.	

Realizado por: Carlos Mejía, Jhonny Guevara, 2016

Tabla 2-59: PA de HT4

PRUEBA DE ACEPTACIÓN	
Código: HT4.1	Nombre Historia de Usuario: Definir el estándar de codificación.
Nombre de la Prueba: Aprobación del estándar de codificación por parte del administrador.	
Responsable: Team Development	Fecha: 19/10/15
Descripción: Se realiza una reunión con el administrador para acordar y revisar el estándar de codificación.	
Condiciones de Ejecución: - Leer los distintos estándares de codificación. - Definir el estándar de codificación de la base de datos. - Definir el estándar de codificación para java.	

Pasos de Ejecución: <ul style="list-style-type: none"> - Gestionar una reunión con el administrador del sistema. - Acordar y aprobar el estándar de codificación.
Resultado Esperado: <ul style="list-style-type: none"> - Se aprueba el estándar de codificación
Evaluación de la Prueba: <p>Satisfactoria.</p>

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

ESTÁNDAR DE CODIFICACIÓN PARA BASE DE DATOS POSTGRESQL:

Nombre de las tablas en la base de datos

Cada tabla deberá ir precedida por una letra t más un guion bajo más el nombre de la tabla a representar. Ejemplo:

Este sería el ejemplo para una tabla de docente:

t_docente

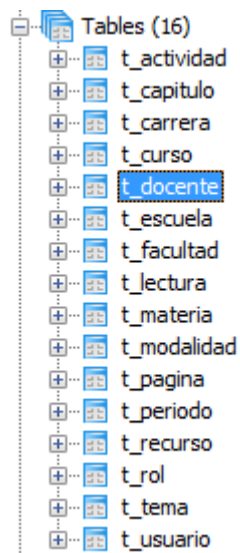


Figura 2-21: Estándar Tablas

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Atributos de las tablas:

Los atributos de cada tabla irán precedidos por las tres primeras letras de la tabla más el nombre del campo a representar, ejemplo de atributo en la tabla docente para representar un id (docid):

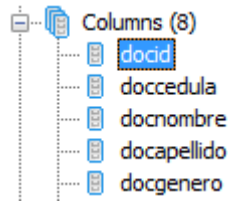


Figura 2-22: Estándar Atributos

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Nombre de secuencias para dato autoincremental

Los nombres de las secuencias serán dados por el nombre de la tabla más un guion bajo más el atributo a declarar autoincremental más un guion bajo más la abreviatura seq. Por ejemplo, de la tabla docente deseamos crear una secuencia autoincremental del atributo id:

t_docente_docid_seq

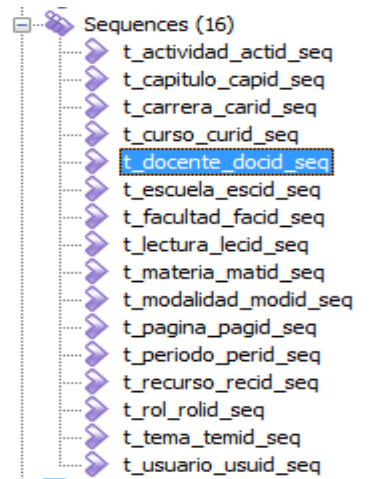


Figura 2-23: Estándar Id Secuencial

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Restricciones Unique

Será añadido el prefijo uk_ más el nombre de la tabla. Ejemplo de Unique del atributo cédula de un docente: uk_docente(doccedula)

```
CONSTRAINT uk_docente UNIQUE (doccedula)
```

Figura 2-24: Estándar Restricción Unique.

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Restricciones llave principal: Será añadido el prefijo pk_ más el nombre de la tabla. Ejemplo de primaryKey de un docente: pk_docente(docid)

```
-----  
CONSTRAINT pk_docente PRIMARY KEY (docid)
```

Figura 2-25: Estándar Restricciones llave principal.

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Restricciones llave foránea: Será añadido el prefijo fk_ más el nombre de la tabla a la que se referencia. Ejemplo de foreignKey de un docente hacia modalidad: fk_modalidad(modid)

```
CONSTRAINT fk_modalidad FOREIGN KEY (modid)  
REFERENCES t_modalidad (modid) MATCH SIMPLE  
ON UPDATE NO ACTION ON DELETE NO ACTION,
```

Figura 2-26: Estándar Restricción FK

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

ESTÁNDAR DE CODIFICACIÓN PARA CODIFICACIÓN JAVA

Nombre de las clases pertenecientes a la capa modelo: cada clase deberá ir precedida por una letra T más el nombre de la tabla a representar proveniente de la base de datos iniciando por la primera mayúscula. Ejemplo: Este sería el ejemplo para una tabla de modalidad:

TModalidad

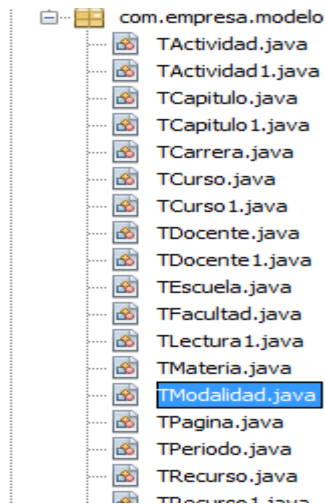


Figura 2-27: Estándar codificación Clases

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Nombre de las clases pertenecientes a la capa controlador: Cada clase deberá ir precedida por una letra T más el nombre deseado empezando con inicial mayúscula y a continuación añadida la terminación Controller. Ejemplo: Este sería el ejemplo para un controlador de modalidad:

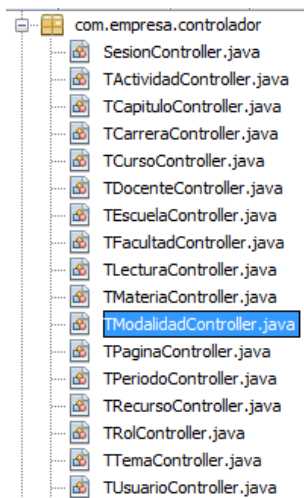


Figura 2-28: Estándar codificación Controladores

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Atributos de las tablas mapeadas en la clase modelo: Los atributos de cada tabla irán precedidos por las tres primeras letras de la tabla más el nombre del campo a representar, ejemplo de atributo en la tabla docente para representar un id (modid):

```

@Table(name = "t_modalidad")
@NamedQueries({
    @NamedQuery(name = "TModalidad.findAll", query = "SE
    @NamedQuery(name = "TModalidad.findByModid", query =
    @NamedQuery(name = "TModalidad.findByModdescripcion"
public class TModalidad implements Serializable {
    private static final long serialVersionUID = 1L;
    @Id
    @GeneratedValue(strategy = GenerationType.IDENTITY)
    @Basic(optional = false)
    @Column(name = "modid")
    private Integer modid;
    @Size(max = 100)
    @Column(name = "moddescripcion")
    private String moddescripcion;

```

Figura 2-29: Estándar codificación Atributos Clase

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Nombre de los métodos creados para las clases modelo o controlador:

Los nombres de los métodos creados para las diferentes clases comenzarán con letra minúscula más cada nombre que se desee añadir comenzando cada inicial de la palabra con mayúscula.

Por ejemplo, de la entidad docente deseamos crear un método que devuelva docente dado cédula; quedaría de la siguiente manera:

devuelveDocenteDadoCedula (String cedula)

```

public TDocente1 devuelveDocenteDadoCedula(String cedula){
    //TDocente lstDocente = new ArrayList<TDocente>();
    EntityManagerFactory emf = Persistence.createEntityManagerFactory("OpenCourseWarePU");
    EntityManager em = emf.createEntityManager();
    String jpql = "SELECT d From TDocente d where d.doccedula=?1";
    Query query = em.createQuery(jpql);
    query.setParameter(1,cedula);
    return (TDocente1) query.getResultList().get(0);
}

```

Figura 2-30: Estándar codificación Métodos Controlador

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Nombre de los atributos que estén contenidos en las clases: Los nombres de los atributos declarados en las diferentes clases comenzarán con letra minúscula más cada nombre que se desee añadir comenzando cada inicial de la palabra con mayúscula. Por ejemplo, de la entidad docente deseamos crear atributo que almacene una lista de docentes obtenida desde un query; quedaría de la siguiente manera:

IstDocente

```
public List<TDocente1> devuelveNumeroDeDocentes(Integer size){
    List<TDocente1> lstDocente= new ArrayList<TDocente1>();
    EntityManagerFactory emf = Persistence.createEntityManagerFactory("OpenCourseWare");
    EntityManager em = emf.createEntityManager();
    String jpql = "SELECT d From TDocente d ";
    Query query = em.createQuery(jpql);
    query.setMaxResults(size);
    lstDocente=query.getResultList();
    return lstDocente;
}
```

Figura 2-31: Estándar codificación Atributos Clases

Realizado por: Carlos Mejía, Jhonny Guevara, 2016

Los Sprint del 03 al 18 contemplan todas las historias de usuario desarrolladas y codificadas bajo el lenguaje de programación java y el ide Netbeans ya que incorpora ventajas como el mapeo de clases de entidad; se hizo uso para la consecución de este objetivo la configuración de la unidad de persistencia “OpenCourseWareEspochPU”, la importación de librerías como primefaces 5.3, jpa.

Las funcionalidades se las documentó a manera de historia de usuario, estas cuentan con un identificador, nombre, descripción y responsable de la misma; pruebas de aceptación con criterio, en comprobar el cumplimiento de la misma, además cuenta con tareas de ingeniería que poseen un identificador, descripción y puntos de esfuerzo.

SPRINT 3

Tabla 2-60: HU1 Manejar sesiones de los usuarios

HISTORIA DE USUARIO	
Número: 1	Sprint Asignado: 3
Fecha Inicio: 20/10/15	Fecha Fin: 21/10/15
Tiempo Estimado:	2 días
Responsable:	Carlos Mejía
Descripción: Como Administrador del sistema solicito se implemente el login y cambio de contraseña para los usuarios del OCW-ESPOCH	
Pruebas de Aceptación: <ul style="list-style-type: none">- Verificar el funcionamiento de los métodos para login.- Verificación de los métodos para cambio de contraseña y campos requeridos	

Realizado por: Carlos Mejía, Jhonny Guevara, 2016

Tabla 2-61: Actividades de HU1

Fecha	Actividad	Puntos Estimados
20/10/15 al 20/10/15	Tarea 1: Crear los métodos necesarios para la autenticación de los usuarios <ul style="list-style-type: none"> - Establecer reuniones con el Ing Jorge Huilca. - Desarrollar los métodos para autenticación - Documentación 	8
21/10/15 al 21/10/15	Tarea 2: Crear los métodos necesarios para modificar las contraseñas de los usuarios. <ul style="list-style-type: none"> - Determinar la fortaleza en seguridad de las claves - Establecer reuniones con el Ing Jorge Huilca - Desarrollar los métodos para la modificación de la contraseña. - Documentación 	8

Realizado por: Carlos Mejía, Jhonny Guevara, 2016

Tabla 2-62: Tarea 1 de la HU1

TAREA DE INGENIERIA	
Sprint: 3	Número de Tarea: 1
Nombre Historia de Usuario: Manejar las sesiones de los usuarios	
Nombre de la Tarea: Crear los métodos necesarios para la autenticación de los usuarios	
Tipo de Tarea: Desarrollo	Programador Responsable: Carlos Mejía
Fecha Inicio: 20/10/05	Fecha Fin: 20/10/15
Descripción: Creación de los métodos necesarios para la autenticación de los usuarios.	
Pruebas de Aceptación: <ul style="list-style-type: none"> - Verificar el funcionamiento de los métodos para login. 	

Realizado por: Carlos Mejía, Jhonny Guevara, 2016

Tabla 2-63: Tarea 2 de HU1

TAREA DE INGENIERIA	
Sprint: 3	Número de Tarea: 2
Nombre Historia de Usuario: Manejar las sesiones de los usuarios	
Nombre de la Tarea: Crear los métodos necesarios para modificar las contraseñas de los usuarios.	
Tipo de Tarea: Desarrollo	Programador Responsable: Carlos Mejía
Fecha Inicio: 21/10/15	Fecha Fin: 21/10/15
Descripción: Creación de los métodos necesarios para modificar las contraseñas de los usuarios.	
Pruebas de Aceptación: <ul style="list-style-type: none"> - Verificación de los métodos para cambio de contraseña y campos requeridos - Mensaje de confirmación cuando se modifique la contraseña de usuario. 	

Realizado por: Carlos Mejía, Jhonny Guevara, 2016

Tabla 2-64: Prueba de aceptación 1 de HU1

PRUEBA DE ACEPTACIÓN	
Código: HU1.1	Nombre Historia de Usuario: Manejar las sesiones de los usuarios: (Administrador y Docente).
Nombre de la Prueba: Verificar el funcionamiento de los métodos para login.	
Responsable: Carlos Mejía	Fecha: 27/10/2015
Descripción: Al instanciar los métodos para iniciar y cerrar sesión es necesario comprobar si el usuario existe en la base de datos y eliminar la sesión instanciada en el servidor respectivamente.	
Condiciones de Ejecución:	
<ul style="list-style-type: none"> - Implementar los métodos para comprobar si existe un usuario en el modelo - Implementar los métodos para comprobar si existe un usuario en el modelo - Crear las vistas necesarias para recopilar la información del usuario. 	
Pasos de Ejecución:	
<ul style="list-style-type: none"> - Clic en la opción administrar - Llenar los campos necesarios para el login - Clic en el botón ingresar. 	
Resultado Esperado:	
<ul style="list-style-type: none"> - Direccionamiento a la página home del usuario autenticado 	
Evaluación de la Prueba: Satisfactoria.	

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-65: Prueba de aceptación 2 de HU1

PRUEBA DE ACEPTACIÓN	
Código: HU1.2	Nombre Historia de Usuario: Manejar las sesiones de los usuarios.
Nombre de la Prueba: Verificación de los métodos para cambio de contraseña y campos requeridos	
Responsable: Carlos Mejía	Fecha: 27/10/2015
Descripción: Comprobar que el usuario autenticado pueda cambiar su contraseña.	
Condiciones de Ejecución:	
<ul style="list-style-type: none"> - Implementar los métodos para modificar contraseña de usuario en el modelo - Implementar los métodos para modificar contraseña de usuario en el controlador - Crear las vistas necesarias para recopilar la información del usuario. 	
Pasos de Ejecución:	
<ul style="list-style-type: none"> - Clic en la opción cambiar contraseña - Llenar los campos necesarios para cambiar contraseña 	
Resultado Esperado:	
<ul style="list-style-type: none"> - Mostrar el mensaje de confirmación cuando se modifique la contraseña - Cerrar sesión del usuario actual. 	
Evaluación de la Prueba:	
Satisfactoria.	

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-66: de HU2 Ingresar, editar y listar los usuarios

HISTORIA DE USUARIO	
Número: 2	Sprint Asignado: 3
Fecha Inicio: 22/10/15	Fecha Fin: 27/10/15
Tiempo Estimado:	4 días
Responsable:	Carlos Mejía
Descripción: Como administrador necesito Ingresar, editar y listar los usuarios.	
Pruebas de Aceptación:	
<ul style="list-style-type: none"> - Verificar el funcionamiento de los métodos para gestionar usuarios. - Mensaje de confirmación al ingresar/editar usuario. - Verificar que el nuevo usuario se encuentre en la lista. 	

Realizado por: Carlos Mejía, Jhonny Guevara, 2016

Tabla 2-67: Actividades de HU2

Fecha	Actividad	Puntos estimados
22/10/15 al 23/10/15	Tarea 1: Crear los métodos necesarios para ingresar y listar usuario - Establecer reuniones con el Ing Jorge Huilca. - Desarrollar los métodos para ingresar y listar usuario - Documentación	16
26/10/15 al 27/10/15	Tarea 2: Crear los métodos necesarios para modificar usuario - Identificar los campos no obligatorios, los obligatorios y que campos se modificaran. - Desarrollar los métodos para modificar materia - Documentación	16

Realizado por: Carlos Mejía, Jhonny Guevara, 2016

Tabla 2-68: Tarea 1 de HU2

TAREA DE INGENIERIA	
Sprint: 3	Número de Tarea: 1
Nombre Historia de Usuario: Ingresar, editar y listar usuario.	
Nombre de la Tarea: Crear los métodos necesarios para ingresar y listar usuario	
Tipo de Tarea: Desarrollo	Programador Responsable: Carlos Mejía
Fecha Inicio: 22/10/05	Fecha Fin: 23/10/15
Descripción: Creación de los métodos necesarios para la ingresar y listar usuario.	

Pruebas de Aceptación: <ul style="list-style-type: none"> - Verificar que la nueva materia se encuentre en la lista. - Validación de los campos requeridos en la GUI

Realizado por: Carlos Mejía, Jhonny Guevara, 2016

Tabla 2-69: Tarea 2 de la HU2

TAREA DE INGENIERIA	
Sprint: 3	Número de Tarea: 2
Nombre Historia de Usuario: Ingresar, editar y listar usuario.	
Nombre de la Tarea: Crear los métodos necesarios para modificar el usuario.	
Tipo de Tarea: Desarrollo	Programador Responsable: Carlos Mejía
Fecha Inicio: 26/11/15	Fecha Fin: 27/11/15
Descripción: Creación de los métodos necesarios para editar usuario.	
Pruebas de Aceptación: <ul style="list-style-type: none"> - Verificación que el usuario modificado se encuentre en la lista. - Mensaje de confirmación cuando se ingrese/modifique usuario. 	

Realizado por: Carlos Mejía, Jhonny Guevara, 2016

Tabla 2-70: Prueba de aceptación 1 de HU2

PRUEBA DE ACEPTACIÓN	
Código: HU2.1	Nombre Historia de Usuario: Ingresar, editar y listar usuario.
Nombre de la Prueba: Verificar que el nuevo usuario se encuentre en la lista.	
Responsable: Carlos Mejía	Fecha: 27/10/2015
Descripción: Al ingresar un usuario es necesario mostrarlo en una lista de usuario para corroborar que se encuentre insertado	
Condiciones de Ejecución: <ul style="list-style-type: none"> - Crear los métodos en el modelo - Crear los métodos en el controlador. - Crear la vista que permita recopilar los datos 	
Pasos de Ejecución: <ul style="list-style-type: none"> - Clic en el menú docente - Clic en el botón docente nuevo - Llenar los campos requeridos - Clic en el botón guardar 	
Resultado Esperado: <ul style="list-style-type: none"> - Guardado y despliegue de usuario en la lista. 	
Evaluación de la Prueba: Satisfactoria.	

Realizado por: Carlos Mejía, Jhonny Guevara, 2016

Tabla 2-71: Prueba de aceptación 2 de HU2

PRUEBA DE ACEPTACIÓN	
Código: HU2.2	Nombre Historia de Usuario: Ingresar, editar y listar usuario.
Nombre de la Prueba: Validación de los campos requeridos en la GUI.	
Responsable: Carlos Mejía	Fecha: 27/10/2015
Descripción: Cuando se desee guardar un usuario se debe controlar que previo se llenen los campos requeridos para garantizar la integridad de los datos	
Condiciones de Ejecución:	
<ul style="list-style-type: none"> - Crear el método para validar campos en el controlador. - Crear el método en el paquete de utilidades para mostrar el mensaje 	
Pasos de Ejecución:	
<ul style="list-style-type: none"> - Clic en el menú docente - Clic en el botón docente nuevo - No llenar algún campo requerido 	
Resultado Esperado:	
<ul style="list-style-type: none"> - Despliegue del mensaje que alerte la falta de campos requeridos 	
Evaluación de la Prueba:	
Satisfactoria.	

Realizado por: Carlos Mejía, Jhonny Guevara, 2016

Tabla 2-72: Prueba de aceptación 3 de HU2

PRUEBA DE ACEPTACIÓN	
Código: HU2.3	Nombre Historia de Usuario: Ingresar, editar y listar usuario
Nombre de la Prueba: Verificar que el usuario editado se encuentre en la lista.	
Responsable: Carlos Mejía	Fecha: 27/10/2015
Descripción: Al editar un usuario es necesario mostrarlo en una lista de usuario para corroborar que se encuentre modificado	
Condiciones de Ejecución:	
<ul style="list-style-type: none"> - Crear los métodos en el modelo - Crear los métodos en el controlador. - Crear la vista que permita recopilar los datos 	
Pasos de Ejecución:	
<ul style="list-style-type: none"> - Clic en el menú docente - Clic en el botón editar docente - Llenar los campos requeridos - Clic en el botón actualizar 	
Resultado Esperado:	
<ul style="list-style-type: none"> - Guardado y despliegue de usuario en la lista. 	
Evaluación de la Prueba:	
Satisfactoria.	

Realizado por: Carlos Mejía, Jhonny Guevara, 2016

Tabla 2-73: Prueba de aceptación 4 de HU2

PRUEBA DE ACEPTACIÓN	
Código: HU2.4	Nombre Historia de Usuario: Ingresar, editar y listar el usuario.
Nombre de la Prueba: Validación de los campos requeridos en la GUI.	
Responsable: Carlos Mejía	Fecha: 27/11/2015
Descripción: Al ingresar un usuario es necesario mostrarlo en una lista de usuarios para corroborar que se encuentre insertado	
Condiciones de Ejecución:	
<ul style="list-style-type: none"> - Crear el método para validar campos en el controlador. - Crear el método en el paquete de utilidades para mostrar el mensaje 	
Pasos de Ejecución:	
<ul style="list-style-type: none"> - Clic en el menú docente - Clic en el botón editar docente - Llenar los campos requeridos - Clic en el botón actualizar 	
Resultado Esperado:	
<ul style="list-style-type: none"> - Despliegue del mensaje que informe de la modificación/ingreso exitoso de datos 	
Evaluación de la Prueba: Satisfactoria.	

Realizado por: Carlos Mejía, Jhonny Guevara, 2016

SPRINT 4

Tabla 2-74: H3 Visualizar Materias

HISTORIA DE USUARIO	
Número: 3	Sprint Asignado: 4
Fecha Inicio: 20/10/15	Fecha Fin: 20/10/15
Tiempo Estimado:	1 día
Responsable:	Jhonny Guevara
Descripción: Como administrador necesito visualizar las materias.	
Pruebas de Aceptación:	
<ul style="list-style-type: none"> - Verificar el funcionamiento de los métodos para visualizar materias. 	

Realizado por: Carlos Mejía, Jhonny Guevara, 2016

Tabla 2-75: Actividades de H3

Fecha	Actividad	Puntos Estimados
20/10/15 al 20/10/15	Tarea 1: Crear los métodos necesarios para listar materia <ul style="list-style-type: none"> - Establecer reuniones con el Ing Jorge Huilca. - Desarrollar los métodos para listar materia - Documentación 	8

Realizado por: Carlos Mejía, Jhonny Guevara, 2016

Tabla 2-76: Tarea 1 de la H3

TAREA DE INGENIERIA	
Sprint: 4	Número de Tarea: 1
Nombre Historia de Usuario: Visualizar materia	
Nombre de la Tarea: Crear los métodos necesarios para listar materia	
Tipo de Tarea: Desarrollo	Programador Responsable: Jhonny Guevara
Fecha Inicio: 20/10/15	Fecha Fin: 20/10/15
Descripción: Creación de los métodos necesarios para la listar materia.	
Pruebas de Aceptación:	
<ul style="list-style-type: none"> - Verificar el funcionamiento de los métodos para visualizar materias. 	

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-77: Prueba de aceptación 1 de H3

PRUEBA DE ACEPTACIÓN	
Código: HU3.1	Nombre Historia de Usuario: Visualizar las materias
Nombre de la Prueba: Verificar el funcionamiento de los métodos para visualizar materias.	
Responsable: Jhonny Guevara	Fecha: 27/10/2015
Descripción: Al ingresar un curso es necesario mostrar una lista de materias para seleccionar a que materia pertenecerá	
Condiciones de Ejecución:	
<ul style="list-style-type: none"> - Crear los métodos en el modelo - Crear los métodos en el controlador. - Crear la vista que permita recopilar los datos 	
Pasos de Ejecución:	
<ul style="list-style-type: none"> - Clic en el menú curso - Clic en el botón curso nuevo - Llenar los campos requeridos - Seleccionar de la lista la materia - Clic en el botón guardar 	
Resultado Esperado:	
<ul style="list-style-type: none"> - Despliegue de la materia en la lista 	
Evaluación de la Prueba:	
Satisfactoria.	

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-78: H4 Visualizar periodos

HISTORIA DE USUARIO	
Número: 4	Sprint Asignado: 4
Fecha Inicio: 21/10/15	Fecha Fin: 21/10/15
Tiempo Estimado:	1 día
Responsable:	Jhonny Guevara
Descripción: Como administrador necesito visualizar los períodos.	
Pruebas de Aceptación:	
<ul style="list-style-type: none"> - Verificar el funcionamiento de los métodos para visualizar períodos. 	

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-79: Actividades de H4

Fecha	Actividad	Puntos Estimados
21/10/15 al 21/10/15	Tarea 1: Crear los métodos necesarios para listar período - Establecer reuniones con el Ing Jorge Huilca. - Desarrollar los métodos para listar período - Documentación	8

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-80: Tarea 1 de la HU4

TAREA DE INGENIERIA	
Sprint: 4	Número de Tarea: 1
Nombre Historia de Usuario: Visualizar período	
Nombre de la Tarea: Crear los métodos necesarios para listar período	
Tipo de Tarea: Desarrollo	Programador Responsable: Jhonny Guevara
Fecha Inicio: 21/10/15	Fecha Fin: 21/10/15
Descripción: Creación de los métodos necesarios para listar período.	
Pruebas de Aceptación:	
<ul style="list-style-type: none"> - Verificar el funcionamiento de los métodos para visualizar período. 	

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-81: Prueba de aceptación 1 de HU4

PRUEBA DE ACEPTACIÓN	
Código: HU4.1	Nombre Historia de Usuario: Visualizar los períodos

Nombre de la Prueba: Verificar el funcionamiento de los métodos para visualizar períodos.	
Responsable: Jhonny Guevara	Fecha: 27/10/2015
Descripción: Al ingresar un curso es necesario mostrar una lista de período para seleccionar a que período pertenecerá	
Condiciones de Ejecución: <ul style="list-style-type: none"> - Crear los métodos en el modelo - Crear los métodos en el controlador. - Crear la vista que permita recopilar los datos 	
Pasos de Ejecución: <ul style="list-style-type: none"> - Clic en el menú curso - Clic en el botón curso nuevo - Llenar los campos requeridos - <u>Seleccionar de la lista el período</u> 	
Resultado Esperado: - Despliegue del período en la lista	
Evaluación de la Prueba: Satisfactoria.	

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-82: HU5 Editar y Visualizar el perfil de usuarios

HISTORIA DE USUARIO	
Número: 5	Sprint Asignado: 4
Fecha Inicio: 22/10/15	Fecha Fin: 27/11/15
Tiempo Estimado:	4 días
Responsable:	Jhonny Guevara
Descripción: Como docente necesito editar y visualizar mi perfil.	
Pruebas de Aceptación: <ul style="list-style-type: none"> - Mensaje de confirmación al ingresar/editar perfil de usuario. - Verificar que el perfil de usuario se encuentre modificado 	

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-83: Actividades de HU4

Fecha	Actividad	Puntos estimados
22/10/15 al 27/10/15	Tarea 1: Crear los métodos necesarios para gestionar perfil de usuario <ul style="list-style-type: none"> - Establecer reuniones con el Ing Jorge Huilca. - Desarrollar los métodos para editar perfil de usuario - Documentación 	32

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-84 Tarea 1 de la HU5

TAREA DE INGENIERIA	
Sprint: 4	Número de Tarea: 1
Nombre Historia de Usuario: Editar y mostrar el perfil de usuario	
Nombre de la Tarea: Crear los métodos necesarios para gestionar perfil de usuario	
Tipo de Tarea: Desarrollo	Programador Responsable: Jhonny Guevara
Fecha Inicio: 22/10/15	Fecha Fin: 27/10/15
Descripción: Creación de los métodos necesarios para la modifica y visualizar el perfil de usuario.	
Pruebas de Aceptación:	
<ul style="list-style-type: none"> - Mensaje de confirmación al ingresar/editar perfil de usuario. - Verificar que el perfil de usuario se encuentre modificado 	

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-85: Prueba de aceptación 1 de HU5

PRUEBA DE ACEPTACIÓN	
Código: HU5.1	Nombre Historia de Usuario: Editar y mostrar el perfil de usuario
Nombre de la Prueba: Mensaje de confirmación al ingresar/editar perfil de usuario.	
Responsable: Jhonny Guevara	Fecha: 27/10/2015
Descripción: Al editar un perfil de usuario es necesario emitir un mensaje de modificación de perfil de usuario	
Condiciones de Ejecución:	
<ul style="list-style-type: none"> - Crear los métodos en el modelo - Crear los métodos en el controlador. - Crear la vista que permita recopilar los datos 	
Pasos de Ejecución:	
<ul style="list-style-type: none"> - Clic en el nombre de su usuario autenticado - Clic en el botón editar perfil - Llenar los campos requeridos - Clic en el botón guardar 	
Resultado Esperado:	
<ul style="list-style-type: none"> - Guardado y despliegue del mensaje usuario modificado. 	
Evaluación de la Prueba:	
Satisfactoria.	

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-86: Prueba de aceptación 2 de HU5

PRUEBA DE ACEPTACIÓN	
Código: HU5.2	Nombre Historia de Usuario: Editar y mostrar el perfil de usuario
Nombre de la Prueba: Verificar que el perfil de usuario se encuentre modificado	
Responsable: Jhonny Guevara	Fecha: 27/10/2015
Descripción: Cuando se haya editado el perfil de usuario; será direccionado a una página en la cual se muestren los datos actuales de perfil	
Condiciones de Ejecución:	
<ul style="list-style-type: none"> - Creación de la vista para mostrar los datos de perfil 	
Pasos de Ejecución:	
<ul style="list-style-type: none"> - Clic en el nombre de su usuario autenticado - Clic en el botón editar perfil - Llenar los campos requeridos - Clic en el botón guardar 	
Resultado Esperado:	
<ul style="list-style-type: none"> - Despliegue de una página en la que se muestren los datos actuales de perfil 	
Evaluación de la Prueba: Satisfactoria	

Realizado por: Carlos Mejía, Jhonny Guevara, 2016

SPRINT 5

Tabla 2-87: HU6 Ingresar, editar y listar los cursos (Administrador)

HISTORIA DE USUARIO	
Número: 6	Sprint Asignado: 5
Fecha Inicio: 28/10/15	Fecha Fin: 04/11/15
Tiempo Estimado:	6 días
Responsable:	Carlos Mejía
Descripción: Como administrador necesito ingresar, editar, listar los cursos.	
Pruebas de Aceptación:	
<ul style="list-style-type: none"> - Verificar el funcionamiento de los métodos para gestionar cursos. - Mensaje de confirmación al ingresar/editar cursos. 	

Realizado por: Carlos Mejía, Jhonny Guevara, 2016

Tabla 2-88: Actividades de HU6

Fecha	Actividad	Puntos estimados
28/11/15 al 30/11/15	Tarea 1: Crear los métodos necesarios para ingresar y listar curso - Establecer reuniones con el Ing Jorge Huilca. - Desarrollar los métodos para ingresar y listar curso	24
02/11/15 al 04/11/15	Tarea 2: Crear los métodos necesarios para modificar curso - Identificar los campos que se modificaran. - Desarrollar los métodos para modificar curso	24

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-89: Tarea 1 de la HU6

TAREA DE INGENIERIA	
Sprint: 5	Número de Tarea: 1
Nombre Historia de Usuario: Ingresar, editar, listar los cursos (Administrador).	
Nombre de la Tarea: Crear los métodos necesarios para ingresar y listar cursos	
Tipo de Tarea: Desarrollo	Programador Responsable: Carlos Mejía
Fecha Inicio: 28/10/15	Fecha Fin: 30/10/15
Descripción: Creación de los métodos necesarios para ingresar y listar curso.	
Pruebas de Aceptación: - Verificar el funcionamiento de los métodos para ingresar y listar curso.	

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-90: Tarea 2 de la HU6

TAREA DE INGENIERIA	
Sprint: 5	Número de Tarea: 2
Nombre Historia de Usuario: Ingresar, editar, listar los cursos(Administrador).	
Nombre de la Tarea: Crear los métodos necesarios para modificar curso	
Tipo de Tarea: Desarrollo	Programador Responsable: Carlos Mejía
Fecha Inicio: 02/11/15	Fecha Fin: 04/11/15
Descripción: Creación de los métodos necesarios para editar curso.	
Pruebas de Aceptación: - Verificación de los métodos para editar cursos. - Mensaje de confirmación cuando se modifique/ingrese el curso.	

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-91: Prueba de aceptación 1 de HU6

PRUEBA DE ACEPTACIÓN	
Código: HU6.1	Nombre Historia de Usuario: Ingresar, editar y listar los cursos (Administrador).
Nombre de la Prueba: Verificar que el nuevo curso se encuentre en la lista.	
Responsable: Carlos Mejía	Fecha: 04/11/2015
Descripción: Al ingresar un curso es necesario mostrarlo en una lista de cursos para corroborar que se encuentre insertado	
Condiciones de Ejecución:	
<ul style="list-style-type: none"> - Crear los métodos en el modelo - Crear los métodos en el controlador. - Crear la vista que permita recopilar los datos 	
Pasos de Ejecución:	
<ul style="list-style-type: none"> - Clic en el menú curso - Clic en el botón curso nuevo - Llenar los campos requeridos - Clic en el botón guardar 	
Resultado Esperado:	
<ul style="list-style-type: none"> - Guardado y despliegue de curso en la lista. 	
Evaluación de la Prueba: Satisfactoria.	

Realizado por: Carlos Mejía, Jhonny Guevara, 2016

Tabla 2-92: Prueba de aceptación 2 de HU6

PRUEBA DE ACEPTACIÓN	
Código: HU6.2	Nombre Historia de Usuario: Ingresar, editar y listar los cursos (Administrador).
Nombre de la Prueba: Validación de los campos requeridos en la GUI.	
Responsable: Carlos Mejía	Fecha: 04/11/2015
Descripción: Cuando se desee guardar un curso se debe controlar que previo se llenen los campos requeridos para garantizar la integridad de los datos	
Condiciones de Ejecución:	
<ul style="list-style-type: none"> - Crear el método para validar campos en el controlador. - Crear el método en el paquete de utilidades para mostrar el mensaje 	
Pasos de Ejecución:	
<ul style="list-style-type: none"> - Clic en el menú curso - Clic en el botón curso nuevo - No llenar algún campo requerido (descripción) - Clic en el botón guardar 	
Resultado Esperado:	
<ul style="list-style-type: none"> - Despliegue del mensaje que alerte la falta de campos requeridos 	
Evaluación de la Prueba: Satisfactoria	

Realizado por: Carlos Mejía, Jhonny Guevara, 2016

Tabla 2-93 Prueba de aceptación 3 de HU6

PRUEBA DE ACEPTACIÓN	
Código: HU6.3	Nombre Historia de Usuario: Ingresar, editar y listar los cursos (Administrador).
Nombre de la Prueba: Verificar que el curso editado se encuentre en la lista.	
Responsable: Carlos Mejía	Fecha: 04/11/2015
Descripción: Al editar un curso es necesario mostrarlo en una lista de cursos para corroborar que se encuentre modificado	
Condiciones de Ejecución:	
<ul style="list-style-type: none"> - Crear los métodos en el modelo - Crear los métodos en el controlador. - Crear la vista que permita recopilar los datos 	
Pasos de Ejecución:	
<ul style="list-style-type: none"> - Clic en el menú curso - Clic en el botón editar curso - Llenar los campos requeridos - Clic en el botón guardar 	
Resultado Esperado:	
<ul style="list-style-type: none"> - Guardado y despliegue del curso en la lista. 	
Evaluación de la Prueba: Satisfactoria.	

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-94: Prueba de aceptación 4 de HU6

PRUEBA DE ACEPTACIÓN	
Código: HU6.4	Nombre Historia de Usuario: Ingresar, editar y listar los cursos (Administrador).
Nombre de la Prueba: Mensaje de confirmación cuando se ingrese/modifique el usuario	
Responsable: Carlos Mejía	Fecha: 04/11/2015
Descripción: Al ingresar/modificar un curso se debe mostrar un mensaje informativo que los datos han sido guardados	
Condiciones de Ejecución:	
<ul style="list-style-type: none"> - Crear el método para validar campos en el controlador. - Crear el método en el paquete de utilidades para mostrar el mensaje 	
Pasos de Ejecución:	
<ul style="list-style-type: none"> - Clic en el menú curso - Clic en el botón curso nuevo - Llenar los campos requeridos - Clic en el botón guardar 	
Resultado Esperado: Despliegue del mensaje que informe de la modificación/ingreso exitoso de datos	
Evaluación de la Prueba: Satisfactoria	

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

SPRINT 6

Tabla 2-95: HU7 Listar cursos por carrera, publicar y editar perfil cursos (Docente).

HISTORIA DE USUARIO	
Número: 7	Sprint Asignado: 7
Fecha Inicio: 28/10/15	Fecha Fin: 04/11/15
Tiempo Estimado:	6 días
Responsable:	Jhonny Guevara
Descripción: Como docente necesito listar cursos por carrera, publicar y editar perfil cursos (Docente).	
Pruebas de Aceptación:	
<ul style="list-style-type: none"> - Verificar el funcionamiento del método listar cursos por carrera - Verificar el funcionamiento del método publicar curso - Mensaje de confirmación al editar el perfil de un curso 	

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-96: Actividades de HU7

Fecha	Actividad	Puntos estimados
28/10/15 al 30/10/15	Tarea 1: Crear los métodos necesarios para publicar y listar cursos por carrera <ul style="list-style-type: none"> - Establecer reuniones con el Ing Jorge Huilca. - Desarrollar los métodos para publicar y listar cursos - Documentación 	24
02/11/15 al 04/11/15	Tarea 2: Crear los métodos necesarios para modificar el perfil del curso <ul style="list-style-type: none"> - Identificar los campos que se modificaran. - Desarrollar los métodos para modificar actividad - Documentación 	24

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-97: Tarea 1 de la HU7

TAREA DE INGENIERIA	
Sprint: 6	Número de Tarea: 1
Nombre Historia de Usuario: listar cursos por carrera, publicar y editar perfil cursos (Docente).	
Nombre de la Tarea: Crear los métodos necesarios para publicar y listar cursos por carrera.	
Tipo de Tarea: Desarrollo	Programador Responsable: Jhonny Guevara
Fecha Inicio: 28/10/15	Fecha Fin: 30/11/15
Descripción: Creación de los métodos necesarios para la publicar y listar cursos por carrera.	

Pruebas de Aceptación: <ul style="list-style-type: none"> - Verificar el funcionamiento del método listar cursos por carrera - Verificar el funcionamiento del método publicar curso

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-98: Tarea 2 de la HU7

TAREA DE INGENIERIA	
Sprint: 6	Número de Tarea: 2
Nombre Historia de Usuario: listar cursos por carrera, publicar y editar perfil cursos (Docente).	
Nombre de la Tarea: Crear los métodos necesarios para modificar el perfil del curso	
Tipo de Tarea: Desarrollo	Programador Responsable: Jhonny Guevara
Fecha Inicio: 02/11/15	Fecha Fin: 04/11/15
Descripción: Creación de los métodos necesarios para modificar el perfil del curso	
Pruebas de Aceptación: <ul style="list-style-type: none"> - Mensaje de confirmación al editar el perfil de un curso 	

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-99: Prueba de aceptación 1 de HU7

PRUEBA DE ACEPTACIÓN	
Código: HU7.1	Nombre Historia de Usuario: listar cursos por carrera, publicar y editar perfil cursos (Docente).
Nombre de la Prueba: Verificar el funcionamiento del método listar cursos por carrera	
Responsable: Jhonny Guevara	Fecha: 04/11/2015
Descripción: Al hacer clic en una carrera, deberá mostrarse la lista de cursos de dado una carrera de un docente.	
Condiciones de Ejecución: <ul style="list-style-type: none"> - Crear los métodos en el modelo - Crear los métodos en el controlador. - Crear la vista que permita mostrar los datos 	
Pasos de Ejecución: <ul style="list-style-type: none"> - Clic en el menú mis cursos - Clic en la carrera que desee observar 	
Resultado Esperado: <ul style="list-style-type: none"> - Despliegue de los cursos dado una carrera que imparte el docente 	
Evaluación de la Prueba: Satisfactoria.	

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-100: Prueba de aceptación 2 de HU7

PRUEBA DE ACEPTACIÓN	
Código: HU7.2	Nombre Historia de Usuario: listar cursos por carrera, publicar y editar perfil cursos (Docente).
Nombre de la Prueba: Verificar el funcionamiento del método publicar curso	
Responsable: Jhonny Guevara	Fecha: 04/11/2015
Descripción: Cuando el usuario decida publicar un curso debe haber la opción para activarlo o desactivarlo.	
Condiciones de Ejecución: <ul style="list-style-type: none"> - Crear el método para activar/desactivar cursos. - Crear la vista que permita gestionar el curso a desactivar/activar 	
Pasos de Ejecución: <ul style="list-style-type: none"> - Clic en el menú mis cursos - Clic en la carrera que desee observar - Seleccione un curso 	
Resultado Esperado: <ul style="list-style-type: none"> - Actualización del estado de un curso a activo/inactivo 	
Evaluación de la Prueba: Satisfactoria	

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-101: Prueba de aceptación 3 de HU7

PRUEBA DE ACEPTACIÓN	
Código: HU7.3	Nombre Historia de Usuario: listar cursos por carrera, publicar y editar perfil cursos (Docente).
Nombre de la Prueba: Mensaje de confirmación al editar el perfil de un curso.	
Responsable: Jhonny Guevara	Fecha: 04/11/2015
Descripción: Al editar el perfil de un curso es necesario desplegar un mensaje que informe sobre si la actualización se realizó.	
Condiciones de Ejecución: <ul style="list-style-type: none"> - Crear los métodos en el modelo - Crear los métodos en el controlador. - Crear la vista que permita mostrar y recopilar los datos 	
Pasos de Ejecución: <ul style="list-style-type: none"> - Clic en el menú mis cursos - Clic en la carrera que desee observar - Seleccione un curso - Clic en el botón editar actividad - Llenar los campos requeridos 	
Resultado Esperado: Despliegue del mensaje Curso Actualizado	
Evaluación de la Prueba: Satisfactoria.	

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

SPRINT 7

Tabla 2-102: HU8 Ingresar, editar, listar y visualizar las actividades de un curso

HISTORIA DE USUARIO	
Número: 8	Sprint Asignado: 10
Fecha Inicio: 05/11/15	Fecha Fin: 12/11/15
Tiempo Estimado:	6 días
Responsable:	Carlos Mejía
Descripción: Como docente necesito Ingresar, editar, listar y visualizar las actividades de un curso.	
Pruebas de Aceptación: <ul style="list-style-type: none"> - Verificar el funcionamiento de los métodos para gestionar actividades. - Mensaje de confirmación al ingresar/editar actividades. - Verificar que las actividades se encuentren en la lista 	

Realizado por: Carlos Mejía, Jhonny Guevara, 2016

Tabla 2-103: Actividades de HU8

Fecha	Actividad	Puntos estimados
05/11/15 al 07/11/15	Tarea 1: Crear los métodos necesarios para ingresar y listar actividad - Establecer reuniones con el Ing Jorge Huilca. - Desarrollar los métodos para ingresar y listar actividad	24
10/11/15 al 12/11/15	Tarea 2: Crear los métodos necesarios para modificar la actividad - Identificar los campos que se modificaran. - Desarrollar los métodos para modificar actividad	24

Realizado por: Carlos Mejía, Jhonny Guevara, 2016

Tabla 2-104: Tarea 1 de la HU8

TAREA DE INGENIERIA	
Sprint: 7	Número de Tarea: 1
Nombre Historia de Usuario: Ingresar, editar, listar y visualizar las actividades de un curso (Docente).	
Nombre de la Tarea: Crear los métodos necesarios para ingresar y listar actividad	
Tipo de Tarea: Desarrollo	Programador Responsable: Carlos Mejía
Fecha Inicio: 05/11/15	Fecha Fin: 07/11/15
Descripción: Creación de los métodos necesarios para la ingresar y listar modalidad.	

<p>Pruebas de Aceptación:</p> <ul style="list-style-type: none"> - Verificar que la nueva actividad se encuentre en la lista. - Validación de los campos requeridos en la GUI.

Realizado por: Carlos Mejía, Jhonny Guevara, 2016

Tabla 2-105: Tarea 2 de la HU8

TAREA DE INGENIERIA	
Sprint: 7	Número de Tarea: 2
Nombre Historia de Usuario: Ingresar, editar, listar y visualizar las actividades de un curso (Docente).	
Nombre de la Tarea: Crear los métodos necesarios para modificar las actividades	
Tipo de Tarea: Desarrollo	Programador Responsable: Carlos Mejía
Fecha Inicio: 10/11/15	Fecha Fin: 12/11/15
Descripción: Creación de los métodos necesarios para editar la actividad	
<p>Pruebas de Aceptación:</p> <ul style="list-style-type: none"> - Verificación de los métodos para editar la actividad. - Mensaje de confirmación cuando se modifique/ingrese la actividad. 	

Realizado por: Carlos Mejía, Jhonny Guevara, 2016

Tabla 2-106: Prueba de aceptación 1 de HU8

PRUEBA DE ACEPTACIÓN	
Código: HU8.1	Nombre Historia de Usuario: Ingresar, editar, listar y visualizar las actividades de un curso (Docente).
Nombre de la Prueba: Verificar que la nueva actividad se encuentre en la lista.	
Responsable: Carlos Mejía	Fecha: 12/11/2015
Descripción: Al ingresar una actividad es necesario mostrarla en una lista de actividades para corroborar que se encuentre insertado	
<p>Condiciones de Ejecución:</p> <ul style="list-style-type: none"> - Crear los métodos en el modelo - Crear los métodos en el controlador. - Crear la vista que permita recopilar los datos 	
<p>Pasos de Ejecución:</p> <ul style="list-style-type: none"> - Clic en el menú modalidad - Clic en el botón modalidad nueva - Llenar los campos requeridos - Clic en el botón guardar 	
<p>Resultado Esperado:</p> <ul style="list-style-type: none"> - Guardado y despliegue de actividad en la lista. 	
<p>Evaluación de la Prueba:</p> <p>Satisfactoria.</p>	

Realizado por: Carlos Mejía, Jhonny Guevara, 2016

Tabla 2-107: Prueba de aceptación 2 de HU8

PRUEBA DE ACEPTACIÓN	
Código: HU8.2	Nombre Historia de Usuario: Ingresar, editar, listar y visualizar las actividades de un curso (Docente).
Nombre de la Prueba: Validación de los campos requeridos en la GUI.	
Responsable: Carlos Mejía	Fecha: 12/11/2015
Descripción: Cuando se desee guardar una actividad se debe controlar que previo se llenen los campos requeridos para garantizar la integridad de los datos	
Condiciones de Ejecución:	
<ul style="list-style-type: none"> - Crear el método para validar campos en el controlador. - Crear el método en el paquete de utilidades para mostrar el mensaje 	
Pasos de Ejecución:	
<ul style="list-style-type: none"> - Clic en el menú actividad - Clic en el botón actividad nueva - No llenar algún campo requerido (descripción) - Clic en el botón guardar 	
Resultado Esperado:	
<ul style="list-style-type: none"> - Despliegue del mensaje que alerte la falta de campos requeridos 	
Evaluación de la Prueba: Satisfactoria	

Realizado por: Carlos Mejía, Jhonny Guevara, 2016

Tabla 2-108: Prueba de aceptación 3 de HU8

PRUEBA DE ACEPTACIÓN	
Código: HU8.3	Nombre Historia de Usuario: Ingresar, editar, listar y visualizar las actividades de un curso (Docente).
Nombre de la Prueba: Verificar que la modalidad editada se encuentre en la lista.	
Responsable: Carlos Mejía	Fecha: 12/11/2015
Descripción: Al editar una actividad es necesario mostrarla en una lista de actividades para corroborar que se encuentre modificado	
Condiciones de Ejecución:	
<ul style="list-style-type: none"> - Crear los métodos en el modelo - Crear los métodos en el controlador. - Crear la vista que permita recopilar los datos 	
Pasos de Ejecución:	
<ul style="list-style-type: none"> - Clic en el menú actividad - Clic en el botón editar actividad - Llenar los campos requeridos 	
Resultado Esperado:	
<ul style="list-style-type: none"> - Guardado y despliegue de la actividad en la lista. 	
Evaluación de la Prueba: Satisfactoria.	

Realizado por: Carlos Mejía, Jhonny Guevara, 2016

Tabla 2-109: Prueba de aceptación 4 de HU8

PRUEBA DE ACEPTACIÓN	
Código: HU8.4	Nombre Historia de Usuario Ingresar, editar, listar y visualizar las actividades de un curso (Docente).
Nombre de la Prueba: Mensaje de confirmación cuando se ingrese/modifique el usuario	
Responsable: Carlos Mejía	Fecha: 12/11/2015
Descripción: Al ingresar/modificar una actividad se debe mostrar un mensaje informativo que los datos han sido guardados	
Condiciones de Ejecución:	
<ul style="list-style-type: none"> - Crear el método para validar campos en el controlador. - Crear el método en el paquete de utilidades para mostrar el mensaje 	
Pasos de Ejecución:	
<ul style="list-style-type: none"> - Clic en el menú actividad - Clic en el botón actividad nueva - Llenar los campos requeridos - Clic en el botón guardar 	
Resultado Esperado:	
<ul style="list-style-type: none"> - Despliegue del mensaje que informe de la modificación/ingreso exitoso de datos 	
Evaluación de la Prueba:	
Satisfactoria	

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

SPRINT 8

Tabla 2-110: Funcionalidad Módulo Docente

HISTORIA DE USUARIO	
Número: 9	Sprint Asignado: 8
Fecha Inicio: 05/11/15	Fecha Fin: 12/11/15
Tiempo Estimado:	6 días
Responsable:	Jhonny Guevara
Descripción: Como docente necesito Ver el perfil, vaciar contenido, ingresar, editar y listar capítulos de un curso.	
Pruebas de Aceptación:	
<ul style="list-style-type: none"> - Verificar el funcionamiento de los métodos para gestionar capítulos. - Mensaje de confirmación al ingresar/editar capítulos. - Verificar que los capítulos se encuentren en la lista 	

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-111: Actividades de HU9

Fecha	Actividad	Puntos estimados
05/11/15 al 07/11/15	Tarea 1: Crear los métodos necesarios para ingresar y listar capítulo - Establecer reuniones con el Ing Jorge Huilca. - Desarrollar los métodos para ingresar y listar capítulo	24
10/11/15 al 12/11/15	Tarea 2: Crear los métodos necesarios para modificar el capítulo, vaciar y ver perfil de curso - Identificar los campos que se modificarán. - Desarrollar los métodos para modificar recurso	24

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-112: Tarea 1 de la HU9

TAREA DE INGENIERIA	
Sprint: 8	Número de Tarea: 1
Nombre Historia de Usuario: Ver el perfil, vaciar contenido, ingresar, editar y listar capítulos de un curso.	
Nombre de la Tarea: Crear los métodos necesarios para ingresar y listar recurso	
Tipo de Tarea: Desarrollo	Programador Responsable: Jhonny Guevara
Fecha Inicio: 05/11/15	Fecha Fin: 07/11/15
Descripción: Creación de los métodos necesarios para ingresar y listar recurso	
Pruebas de Aceptación:	
<ul style="list-style-type: none"> - Verificar el funcionamiento de los métodos para gestionar capítulos. - Mensaje de confirmación al ingresar/editar capítulos. 	

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-113: Tarea 2 de la HU9

TAREA DE INGENIERIA	
Sprint: 13	Número de Tarea: 2
Nombre Historia de Usuario: Ver el perfil, vaciar contenido, ingresar, editar y listar capítulos de un curso.	
Nombre de la Tarea: Crear los métodos necesarios para modificar el capítulo, vaciar y ver perfil de curso.	
Tipo de Tarea: Desarrollo	Programador Responsable: Jhonny Guevara
Fecha Inicio: 10/11/15	Fecha Fin: 12/11/15
Descripción: Creación de los métodos necesarios para modificar el capítulo, vaciar y ver perfil de curso.	

<p>Pruebas de Aceptación:</p> <ul style="list-style-type: none"> - Verificar que los capítulos se encuentren en la lista - Verificar que el contenido del curso sea vaciado
--

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-114: Prueba de aceptación 1 de HU9

PRUEBA DE ACEPTACIÓN	
Código: HU9.1	Nombre Historia de Usuario: Ver el perfil, vaciar contenido, ingresar, editar y listar capítulos de un curso.
Nombre de la Prueba: Verificar el funcionamiento de los métodos para gestionar capítulos.	
Responsable: Jhonny Guevara	Fecha: 12/11/2015
Descripción: Al ingresar un capítulo es necesario mostrarlo en una lista de recursos para corroborar que se encuentre insertado	
Condiciones de Ejecución:	
<ul style="list-style-type: none"> - Crear los métodos en el modelo - Crear los métodos en el controlador. - Crear la vista que permita recopilar los datos 	
Pasos de Ejecución:	
<ul style="list-style-type: none"> - Clic en el menú capítulo - Clic en el botón capítulo nuevo - Llenar los campos requeridos - Clic en el botón guardar 	
Resultado Esperado:	
<ul style="list-style-type: none"> - Guardado y despliegue de capítulo en la lista. 	
Evaluación de la Prueba:	
Satisfactoria.	

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-115: Prueba de aceptación 2 de HU9

PRUEBA DE ACEPTACIÓN	
Código: HU9.2	Nombre Historia de Usuario: Ver el perfil, vaciar contenido, ingresar, editar y listar capítulos de un curso.
Nombre de la Prueba: Mensaje de confirmación al ingresar/editar capítulos.	
Responsable: Jhonny Guevara	Fecha: 12/11/2015
Descripción: Cuando se desee insertar/ editar un capítulo se debe mostrar un mensaje que informe sobre el ingreso/modificación del capítulo	
Condiciones de Ejecución:	
<ul style="list-style-type: none"> - Crear el método para validar campos en el controlador. - Crear el método en el paquete de utilidades para mostrar el mensaje 	

Pasos de Ejecución: <ul style="list-style-type: none"> - Clic en el menú recurso - Clic en el botón recurso nuevo - Llenar los campos requeridos - Clic en el botón guardar
Resultado Esperado: <ul style="list-style-type: none"> - Despliegue del mensaje que el capítulo se ha ingresado
Evaluación de la Prueba: Satisfactoria
Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-116: Prueba de aceptación 3 de HU9

PRUEBA DE ACEPTACIÓN	
Código: HU9.3	Nombre Historia de Usuario: Ver el perfil, vaciar contenido, ingresar, editar y listar capítulos de un curso.
Nombre de la Prueba: Verificar que los capítulos se encuentren en la lista	
Responsable: Jhonny Guevara	Fecha: 12/11/2015
Descripción: Al editar un capítulo es necesario mostrarlo en una lista de capítulos para corroborar que se encuentre modificado	
Condiciones de Ejecución: <ul style="list-style-type: none"> - Crear los métodos en el modelo - Crear los métodos en el controlador. - Crear la vista que permita recopilar los datos 	
Pasos de Ejecución: <ul style="list-style-type: none"> - Clic en el menú capítulo - Clic en el botón editar capítulo - Llenar los campos requeridos 	
Resultado Esperado: - Guardado y despliegue del capítulo en la lista.	
Evaluación de la Prueba: Satisfactoria.	
Realizado por: Carlos Mejia, Jhonny Guevara, 2016	

Tabla 2-117: Prueba de aceptación 4 de HU9

PRUEBA DE ACEPTACIÓN	
Código: HU9.4	Nombre Historia de Usuario: Ver el perfil, vaciar contenido, ingresar, editar y listar capítulos de un curso.
Nombre de la Prueba: Verificar que el contenido del curso sea vaciado	
Responsable: Jhonny Guevara	Fecha: 12/11/2015
Descripción: Al vaciar el contenido de un curso se desea visualizar que el contenido de este haya sido vaciado.	

Condiciones de Ejecución: <ul style="list-style-type: none"> - Crear los métodos en el modelo - Crear los métodos en el controlador. - Crear la vista que permita gestionar los datos
Pasos de Ejecución: <ul style="list-style-type: none"> - Clic en el menú mis cursos - Seleccione el curso a vaciar el contenido - Clic en el botón vaciar - Clic en el botón explorar del curso
Resultado Esperado: <ul style="list-style-type: none"> - Despliegue de las listas de contenidos para un curso vacías.
Evaluación de la Prueba: Satisfactoria
Realizado por: Carlos Mejia, Jhonny Guevara, 2016

SPRINT 9

Tabla 2-118: HU10 Gestión Lecturas Docente

HISTORIA DE USUARIO	
Número: 10	Sprint Asignado: 9
Fecha Inicio: 13/11/15	Fecha Fin: 20/11/15
Tiempo Estimado:	6 días
Responsable:	Carlos Mejía
Descripción: Como docente necesito Ingresar, editar, listar y visualizar las lecturas de un curso (Docente)	
Pruebas de Aceptación: <ul style="list-style-type: none"> - Verificar el funcionamiento de los métodos para gestionar lecturas. - Mensaje de confirmación al ingresar/editar lecturas. - Verificar que las lecturas se encuentren en la lista 	

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-119: Actividades de HU10

Fecha	Actividad	Puntos estimados
13/11/15 al 15/11/15	Tarea 1: Crear los métodos necesarios para ingresar y listar lectura <ul style="list-style-type: none"> - Establecer reuniones con el Ing Jorge Huilca. - Desarrollar los métodos para ingresar y listar lectura 	24
18/11/15 al 20/11/15	Tarea 2: Crear los métodos necesarios para modificar lectura <ul style="list-style-type: none"> - Identificar los campos que se modificaran. - Desarrollar los métodos para modificar lectura - Documentación 	24

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-120: Tarea 1 de la HU10

TAREA DE INGENIERIA	
Sprint: 9	Número de Tarea: 1
Nombre Historia de Usuario: Ingresar, editar, listar y visualizar las lecturas de un curso (Docente)	
Nombre de la Tarea: Crear los métodos necesarios para ingresar y listar lectura	
Tipo de Tarea: Desarrollo	Programador Responsable: Carlos Mejía
Fecha Inicio: 13/11/15	Fecha Fin: 15/11/15
Descripción: Creación de los métodos necesarios para ingresar y listar lectura	
Pruebas de Aceptación: <ul style="list-style-type: none"> - Verificar que la nueva lectura se encuentre en la lista. - Validación de los campos requeridos en la GUI. 	

Realizado por: Carlos Mejía, Jhonny Guevara, 2016

Tabla 2-121: Tarea 1 de la HU10

TAREA DE INGENIERIA	
Sprint: 9	Número de Tarea: 2
Nombre Historia de Usuario: Ingresar, editar, listar y visualizar las lecturas de un curso (Docente)	
Nombre de la Tarea: Crear los métodos necesarios para modificar lectura.	
Tipo de Tarea: Desarrollo	Programador Responsable: Carlos Mejía
Fecha Inicio: 18/11/15	Fecha Fin: 20/11/15
Descripción: Creación de los métodos necesarios para editar lectura	
Pruebas de Aceptación: <ul style="list-style-type: none"> - Verificación de los métodos para editar lectura. - Mensaje de confirmación cuando se modifique/ingrese lectura. 	

Realizado por: Carlos Mejía, Jhonny Guevara, 2016

Tabla 2-122: Prueba de aceptación 1 de HU10

PRUEBA DE ACEPTACIÓN	
Código: HU10.1	Nombre Historia de Usuario: Ingresar, editar, listar y visualizar las lecturas de un curso (Docente).
Nombre de la Prueba: Verificar que la nueva lectura se encuentre en la lista.	
Responsable: Carlos Mejía	Fecha: 20/11/2015
Descripción: Al ingresar una lectura es necesario mostrarla en una lista de lecturas para corroborar que se encuentre insertada	

Condiciones de Ejecución: <ul style="list-style-type: none"> - Crear los métodos en el modelo - Crear los métodos en el controlador. - Crear la vista que permita recopilar los datos
Pasos de Ejecución: <ul style="list-style-type: none"> - Clic en el menú lectura - Clic en el botón lectura nueva - Llenar los campos requeridos - Clic en el botón guardar
Resultado Esperado: <ul style="list-style-type: none"> - Guardado y despliegue de lectura en la lista.
Evaluación de la Prueba: Satisfactoria.

Realizado por: Carlos Mejía, Jhonny Guevara, 2016

Tabla 2-123: Prueba de aceptación 2 de HU10

PRUEBA DE ACEPTACIÓN	
Código: HU10.2	Nombre Historia de Usuario: Ingresar, editar, listar y visualizar las lecturas de un curso (Docente)
Nombre de la Prueba: Validación de los campos requeridos en la GUI.	
Responsable: Carlos Mejía	Fecha: 20/11/2015
Descripción: Cuando se desee guardar una lectura se debe controlar que previo se llenen los campos requeridos para garantizar la integridad de los datos	
Condiciones de Ejecución: <ul style="list-style-type: none"> - Crear el método para validar campos en el controlador. - Crear el método en el paquete de utilidades para mostrar el mensaje 	
Pasos de Ejecución: <ul style="list-style-type: none"> - Clic en el menú lectura - Clic en el botón lectura nueva - No llenar algún campo requerido (descripción) - Clic en el botón guardar 	
Resultado Esperado: <ul style="list-style-type: none"> - Despliegue del mensaje que alerte la falta de campos requeridos 	
Evaluación de la Prueba: Satisfactoria	

Realizado por: Carlos Mejía, Jhonny Guevara, 2016

Tabla 2-124: Prueba de aceptación 3 de HU10

PRUEBA DE ACEPTACIÓN	
Código: HU10.3	Nombre Historia de Usuario: Ingresar, editar, listar y visualizar las lecturas de un curso (Docente)
Nombre de la Prueba: Verificar que la lectura editada se encuentre en la lista.	
Responsable: Carlos Mejía	Fecha: 20/11/2015

Descripción: Al editar una lectura es necesario mostrarla en una lista de lecturas para corroborar que se encuentre modificada
Condiciones de Ejecución: <ul style="list-style-type: none"> - Crear los métodos en el modelo - Crear los métodos en el controlador. - Crear la vista que permita recopilar los datos
Pasos de Ejecución: <ul style="list-style-type: none"> - Clic en el menú lectura - Clic en el botón editar lectura - Llenar los campos requeridos - Clic en el botón guardar
Resultado Esperado: <ul style="list-style-type: none"> - Guardado y despliegue de lectura en la lista.
Evaluación de la Prueba: Satisfactoria.
Realizado por: Carlos Mejía, Jhonny Guevara, 2016

Tabla 2-125: Prueba de aceptación 4 de HU10

PRUEBA DE ACEPTACIÓN	
Código: HU10.4	Nombre Historia de Usuario Ingresar, editar, listar y visualizar las lecturas de un curso (Docente)
Nombre de la Prueba: Mensaje de confirmación cuando se ingrese/modifique la lectura	
Responsable: Carlos Mejía	Fecha: 20/11/2015
Descripción: Al ingresar/modificar una lectura se debe mostrar un mensaje informativo que los datos han sido guardados	
Condiciones de Ejecución: <ul style="list-style-type: none"> - Crear el método para validar campos en el controlador. - Crear el método en el paquete de utilidades para mostrar el mensaje 	
Pasos de Ejecución: <ul style="list-style-type: none"> - Clic en el menú lectura - Clic en el botón lectura nueva - Llenar los campos requeridos - Clic en el botón guardar 	
Resultado Esperado: <ul style="list-style-type: none"> - Despliegue del mensaje que informe de la modificación/ingreso exitoso de datos 	
Evaluación de la Prueba: Satisfactoria	
Realizado por: Carlos Mejía, Jhonny Guevara, 2016	

SPRINT 10

Tabla 2-126: HU11 Gestionar Recursos Docente

HISTORIA DE USUARIO	
Número: 11	Sprint Asignado: 10
Fecha Inicio: 13/12/15	Fecha Fin: 20/11/15
Tiempo Estimado:	6 días
Responsable:	Jhonny Guevara
Descripción: Como docente necesito Ingresar, editar, listar y visualizar los recursos de un curso (Docente).	
Pruebas de Aceptación: <ul style="list-style-type: none">- Verificar el funcionamiento de los métodos para gestionar recursos.- Mensaje de confirmación al ingresar/editar recurso.- Verificar que los recursos se encuentren en la lista	

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-127: HU11 Actividades de HU11

Fecha	Actividad	Puntos estimados
13/11/15 al 15/11/15	Tarea 1: Crear los métodos necesarios para ingresar y listar recurso <ul style="list-style-type: none">- Establecer reuniones con el Ing Jorge Huilca.- Desarrollar los métodos para ingresar y listar recurso- Documentación	24
18/11/15 al 20/11/15	Tarea 2: Crear los métodos necesarios para modificar el recurso <ul style="list-style-type: none">- Identificar los campos que se modificaran.- Desarrollar los métodos para modificar recurso- Documentación	24

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-128: Tarea 1 de la HU11

TAREA DE INGENIERIA	
Sprint: 10	Número de Tarea: 1
Nombre Historia de Usuario: Ingresar, editar, listar y visualizar los recursos de un curso	
Nombre de la Tarea: Crear los métodos necesarios para ingresar y listar recurso	
Tipo de Tarea: Desarrollo	Programador Responsable: Jhonny Guevara
Fecha Inicio: 13/11/15	Fecha Fin: 15/11/15
Descripción: Creación de los métodos necesarios para ingresar y listar recurso	

Pruebas de Aceptación: <ul style="list-style-type: none"> - Verificar que el nuevo recurso se encuentre en la lista. - Validación de los campos requeridos en la GUI.
--

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-129: Tarea 2 de la HU11

TAREA DE INGENIERIA	
Sprint: 10	Número de Tarea: 2
Nombre Historia de Usuario: Ingresar, editar, listar y visualizar los recursos de un curso	
Nombre de la Tarea: Crear los métodos necesarios para modificar recurso.	
Tipo de Tarea: Desarrollo	Programador Responsable: Jhonny Guevara
Fecha Inicio: 18/11/15	Fecha Fin: 20/11/15
Descripción: Creación de los métodos necesarios para editar recurso	
Pruebas de Aceptación: <ul style="list-style-type: none"> - Verificación de los métodos para editar recurso. - Mensaje de confirmación cuando se modifique/ingrese recurso. 	

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-130: Prueba de Aceptación 1 de la HU11

PRUEBA DE ACEPTACIÓN	
Código: HU11.1	Nombre Historia de Usuario: Ingresar, editar, listar y visualizar los recursos de un curso
Nombre de la Prueba: Verificar que el nuevo recurso se encuentre en la lista.	
Responsable: Jhonny Guevara	Fecha: 20/11/2015
Descripción: Al ingresar un recurso es necesario mostrarlo en una lista de recursos para corroborar que se encuentre insertado	
Condiciones de Ejecución: <ul style="list-style-type: none"> - Crear los métodos en el modelo - Crear los métodos en el controlador. - Crear la vista que permita recopilar los datos 	
Pasos de Ejecución: <ul style="list-style-type: none"> - Clic en el menú recurso - Clic en el botón recurso nuevo - Llenar los campos requeridos 	
Resultado Esperado: <ul style="list-style-type: none"> - Guardado y despliegue de recurso en la lista. 	
Evaluación de la Prueba: Satisfactoria.	

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-131: Prueba de Aceptación 2 de la HU11

PRUEBA DE ACEPTACIÓN	
Código: HU11.2	Nombre Historia de Usuario: Ingresar, editar, listar y visualizar los recursos de un curso
Nombre de la Prueba: Validación de los campos requeridos en la GUI.	
Responsable: Jhonny Guevara	Fecha: 20/11/2015
Descripción: Cuando se desee guardar un recurso se debe controlar que previo se llenen los campos requeridos para garantizar la integridad de los datos	
Condiciones de Ejecución:	
<ul style="list-style-type: none"> - Crear el método para validar campos en el controlador. - Crear el método en el paquete de utilidades para mostrar el mensaje 	
Pasos de Ejecución:	
<ul style="list-style-type: none"> - Clic en el menú recurso - Clic en el botón recurso nuevo - No llenar algún campo requerido (descripción) - Clic en el botón guardar 	
Resultado Esperado:	
<ul style="list-style-type: none"> - Despliegue del mensaje que alerte la falta de campos requeridos 	
Evaluación de la Prueba: Satisfactoria	

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-132: Prueba de Aceptación 3 de la HU11

PRUEBA DE ACEPTACIÓN	
Código: HU11.3	Nombre Historia de Usuario: Ingresar, editar, listar y visualizar los recursos de un curso
Nombre de la Prueba: Verificar que el recurso editado se encuentre en la lista.	
Responsable: Jhonny Guevara	Fecha: 20/11/2015
Descripción: Al editar un recurso es necesario mostrarlo en una lista de recurso para corroborar que se encuentre modificado	
Condiciones de Ejecución:	
<ul style="list-style-type: none"> - Crear los métodos en el modelo - Crear los métodos en el controlador. - Crear la vista que permita recopilar los datos 	
Pasos de Ejecución:	
<ul style="list-style-type: none"> - Clic en el menú recurso - Clic en el botón editar recurso - Llenar los campos requeridos - Clic en el botón guardar 	
Resultado Esperado: - Guardado y despliegue de recurso en la lista.	
Evaluación de la Prueba: Satisfactoria.	

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-133: Prueba de Aceptación 4 de la HU11

PRUEBA DE ACEPTACIÓN	
Código: HU11.4	Nombre Historia de Usuario Ingresar, editar, listar y visualizar los recursos
Nombre de la Prueba: Mensaje de confirmación cuando se ingrese/modifique recurso	
Responsable: Jhonny Guevara	Fecha: 20/11/2015
Descripción: Al ingresar/modificar un recurso se debe mostrar un mensaje informativo que los datos han sido guardados	
Condiciones de Ejecución: <ul style="list-style-type: none"> - Crear el método para validar campos en el controlador. - Crear el método en el paquete de utilidades para mostrar el mensaje 	
Pasos de Ejecución: <ul style="list-style-type: none"> - Clic en el menú recurso - Clic en el botón recurso nuevo - Llenar los campos requeridos - Clic en el botón guardar 	
Resultado Esperado: <ul style="list-style-type: none"> - Despliegue del mensaje que informe de la modificación/ingreso exitoso de datos 	
Evaluación de la Prueba: Satisfactoria	

Realizado por: Carlos Mejía, Jhonny Guevara, 2016

SPRINT 11

Tabla 2-134: HU12 Gestionar Temas-Curso Docente

HISTORIA DE USUARIO	
Número: 12	Sprint Asignado: 11
Fecha Inicio: 23/11/15	Fecha Fin: 30/11/15
Tiempo Estimado:	6 días
Responsable:	Carlos Mejía
Descripción: Como docente necesito Ingresar, editar, listar y visualizar los temas de un curso (Docente).	
Pruebas de Aceptación: <ul style="list-style-type: none"> - Verificar el funcionamiento de los métodos para gestionar tema. - Mensaje de confirmación al ingresar/editar tema. - Verificar que los temas se encuentren en la lista 	

Realizado por: Carlos Mejía, Jhonny Guevara, 2016

Tabla 2-135: Actividades de HU12

Fecha	Actividad	Puntos estimados
23/11/15 al 25/11/15	Tarea 1: Crear los métodos necesarios para ingresar y listar tema - Establecer reuniones con el Ing Jorge Huilca. - Desarrollar los métodos para ingresar y listar tema - Documentación	24
28/11/15 al 30/11/15	Tarea 2: Crear los métodos necesarios para modificar el tema - Identificar los campos que se modificaran. - Desarrollar los métodos para modificar tema - Documentación	24

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-136: Tarea 1 de la HU12

TAREA DE INGENIERIA	
Sprint: 11	Número de Tarea: 1
Nombre Historia de Usuario: Ingresar, editar, listar y visualizar los temas de un curso	
Nombre de la Tarea: Crear los métodos necesarios para ingresar y listar tema	
Tipo de Tarea: Desarrollo	Programador Responsable: Carlos Mejía
Fecha Inicio: 23/11/15	Fecha Fin: 25/11/15
Descripción: Creación de los métodos necesarios para ingresar y listar tema	
Pruebas de Aceptación: - Verificar que el nuevo tema se encuentre en la lista. - Validación de los campos requeridos en la GUI.	

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-137 Tarea 2 de la HU12

TAREA DE INGENIERIA	
Sprint: 11	Número de Tarea: 2
Nombre Historia de Usuario: Ingresar, editar, listar y visualizar los temas de un curso	
Nombre de la Tarea: Crear los métodos necesarios para modificar tema.	
Tipo de Tarea: Desarrollo	Programador Responsable: Carlos Mejía
Fecha Inicio: 28/11/15	Fecha Fin: 30/11/15
Descripción: Creación de los métodos necesarios para editar tema	
Pruebas de Aceptación: - Verificación de los métodos para editar tema. - Mensaje de confirmación cuando se modifique/ingrese tema.	

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-138: Prueba de aceptación 1 de HU12

PRUEBA DE ACEPTACIÓN	
Código: HU12.1	Nombre Historia de Usuario: Ingresar, editar, listar y visualizar los temas de un curso
Nombre de la Prueba: Verificar que el nuevo tema se encuentre en la lista.	
Responsable: Carlos Mejía	Fecha: 30/11/2015
Descripción: Al ingresar un tema es necesario mostrarlo en una lista de temas para corroborar que se encuentre insertado	
Condiciones de Ejecución:	
<ul style="list-style-type: none"> - Crear los métodos en el modelo - Crear los métodos en el controlador. - Crear la vista que permita recopilar los datos 	
Pasos de Ejecución:	
<ul style="list-style-type: none"> - Clic en el menú tema - Clic en el botón tema nuevo - Llenar los campos requeridos - Clic en el botón guardar 	
Resultado Esperado:	
Guardado y despliegue del tema en la lista.	
Evaluación de la Prueba:	
Satisfactoria.	

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-139: Prueba de aceptación 2 de HU12

PRUEBA DE ACEPTACIÓN	
Código: HU12.2	Nombre Historia de Usuario: Ingresar, editar, listar y visualizar los temas de un curso
Nombre de la Prueba: Validación de los campos requeridos en la GUI.	
Responsable: Carlos Mejía	Fecha: 30/11/2015
Descripción: Cuando se desee guardar un tema se debe controlar que previo se llenen los campos requeridos para garantizar la integridad de los datos	
Condiciones de Ejecución:	
<ul style="list-style-type: none"> - Crear el método para validar campos en el controlador. - Crear el método en el paquete de utilidades para mostrar el mensaje 	
Pasos de Ejecución:	
<ul style="list-style-type: none"> - Clic en el menú tema - Clic en el botón tema nuevo - No llenar algún campo requerido (descripción) 	
Resultado Esperado:	
<ul style="list-style-type: none"> - Despliegue del mensaje que alerte la falta de campos requeridos 	
Evaluación de la Prueba: Satisfactoria	

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-140: Prueba de aceptación 3 de HU12

PRUEBA DE ACEPTACIÓN	
Código: HU12.3	Nombre Historia de Usuario: Ingresar, editar, listar y visualizar los temas de un curso
Nombre de la Prueba: Verificar que el tema editado se encuentre en la lista.	
Responsable: Carlos Mejía	Fecha: 30/11/2015
Descripción: Al editar un tema es necesario mostrarlo en una lista de temas para corroborar que se encuentre modificado	
Condiciones de Ejecución:	
<ul style="list-style-type: none"> - Crear los métodos en el modelo - Crear los métodos en el controlador. - Crear la vista que permita recopilar los datos 	
Pasos de Ejecución:	
<ul style="list-style-type: none"> - Clic en el menú tema - Clic en el botón editar tema - Llenar los campos requeridos - Clic en el botón guardar 	
Resultado Esperado: Guardado y despliegue del tema en la lista.	
Evaluación de la Prueba: Satisfactoria.	

Realizado por: Carlos Mejía, Jhonny Guevara, 2016

Tabla 2-141: Prueba de aceptación 4 de HU12

PRUEBA DE ACEPTACIÓN	
Código: HU12.4	Nombre Historia de Usuario: Ingresar, editar, listar y visualizar los temas de un curso
Nombre de la Prueba: Mensaje de confirmación cuando se ingrese/modifique el tema	
Responsable: Carlos Mejía	Fecha: 30/11/2015
Descripción: Al ingresar/modificar un tema se debe mostrar un mensaje informativo que los datos han sido guardados	
Condiciones de Ejecución:	
<ul style="list-style-type: none"> - Crear el método para validar campos en el controlador. - Crear el método en el paquete de utilidades para mostrar el mensaje 	
Pasos de Ejecución:	
<ul style="list-style-type: none"> - Clic en el menú tema - Clic en el botón tema nuevo - Llenar los campos requeridos - Clic en el botón guardar 	
Resultado Esperado: Despliegue del mensaje que informe de la modificación/ingreso exitoso de datos	
Evaluación de la Prueba:	
Satisfactoria	

Realizado por: Carlos Mejía, Jhonny Guevara, 2016

SPRINT 12

Tabla 2-142: HU13 Gestionar Cursos Visitante

HISTORIA DE USUARIO	
Número: 13	Sprint Asignado: 12
Fecha Inicio: 23/11/15	Fecha Fin: 26/11/15
Tiempo Estimado:	4 días
Responsable:	Jhonny Guevara
Descripción: Como visitante necesito listar los cursos: más vistos, más recientes, publicados, (Visitante).	
Pruebas de Aceptación:	
<ul style="list-style-type: none"> - Verificar que se muestren los cursos más vistos. - Verificar que se muestren los cursos más recientes. - Verificar que se muestren los cursos publicados 	

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-143: Actividades de HU13

Fecha	Actividad	Puntos estimados
23/11/15 al 26/11/15	Tarea 1: Crear los métodos necesarios para listar los cursos: más vistos, más recientes, publicados. - Establecer reuniones con el Ing Jorge Huilca. - Desarrollar los métodos para listar cursos	32

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-144: Tarea 1 de la HU13

TAREA DE INGENIERIA	
Sprint: 12	Número de Tarea: 1
Nombre Historia de Usuario: Listar los cursos: más vistos, más recientes, publicados, (Visitante).	
Nombre de la Tarea: Crear los métodos necesarios para listar los cursos: más vistos, más recientes, publicados.	
Tipo de Tarea: Desarrollo	Programador Responsable: Jhonny Guevara
Fecha Inicio: 23/11/15	Fecha Fin: 26/11/15
Descripción: Creación de los métodos necesarios para ingresar y listar recurso	

<p>Pruebas de Aceptación:</p> <ul style="list-style-type: none"> - Verificar que se muestren los cursos más vistos. - Verificar que se muestren los cursos más recientes. - Verificar que se muestren los cursos publicados

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-145: Prueba de aceptación 1 de HU13

PRUEBA DE ACEPTACIÓN	
Código: HU13.1	Nombre Historia de Usuario: Listar los cursos: más vistos, más recientes, publicados, (Visitante).
Nombre de la Prueba: Verificar que se muestren los cursos más vistos.	
Responsable: Jhonny Guevara	Fecha: 30/11/2015
Descripción: Al acceder a la página de inicio del OCW deberán mostrarse los cursos ingresados en el sistema	
<p>Condiciones de Ejecución:</p> <ul style="list-style-type: none"> - Crear los métodos en el modelo - Crear los métodos en el controlador. - Crear la vista que permita mostrar los datos 	
<p>Pasos de Ejecución:</p> <ul style="list-style-type: none"> - Abrir la página de inicio del sistema 	
Resultado Esperado: Despliegue de los cursos más vistos en un carrusel web.	
Evaluación de la Prueba: Satisfactoria.	

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-146: Prueba de aceptación 2 de HU13

PRUEBA DE ACEPTACIÓN	
Código: HU13.2	Nombre Historia de Usuario: Listar los cursos: más vistos, más recientes, publicados, (Visitante).
Nombre de la Prueba: Verificar que se muestren los cursos más recientes.	
Responsable: Jhonny Guevara	Fecha: 30/11/2015
Descripción: Al acceder a la página de inicio del OCW deberán mostrarse los cursos ingresados en el sistema	
<p>Condiciones de Ejecución:</p> <ul style="list-style-type: none"> - Crear los métodos en el modelo - Crear los métodos en el controlador. - Crear la vista que permita mostrar los datos 	
<p>Pasos de Ejecución:</p> <ul style="list-style-type: none"> - Abrir la página de inicio del sistema 	
Resultado Esperado:	
<ul style="list-style-type: none"> - Despliegue de los cursos más recientes en un carrusel web. 	
Evaluación de la Prueba:	
Satisfactoria.	

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-147: Prueba de aceptación 3 de HU13

PRUEBA DE ACEPTACIÓN	
Código: HU13.3	Nombre Historia de Usuario: Listar los cursos: más vistos, más recientes, publicados, (Visitante).
Nombre de la Prueba: Verificar que se muestren los cursos publicados.	
Responsable: Jhonny Guevara	Fecha: 30/11/2015
Descripción: Al acceder a la página de inicio del OCW deberán mostrarse los cursos ingresados en el sistema	
Condiciones de Ejecución:	
<ul style="list-style-type: none"> - Crear los métodos en el modelo - Crear los métodos en el controlador. - Crear la vista que permita mostrar los datos 	
Pasos de Ejecución:	
<ul style="list-style-type: none"> - Abrir la página de inicio del sistema 	
Resultado Esperado:	
<ul style="list-style-type: none"> - Despliegue de los cursos publicados en un carrusel web. 	
Evaluación de la Prueba:	
Satisfactoria.	

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-148: HU14 Galería de Imágenes

HISTORIA DE USUARIO	
Número: 14	Sprint Asignado: 12
Fecha Inicio: 15/12/15	Fecha Fin: 16/12/15
Tiempo Estimado:	2 días
Responsable:	Jhonny Guevara
Descripción: Como visitante necesito mostrar la galería de imágenes de un curso (visitante).	
Pruebas de Aceptación:	
<ul style="list-style-type: none"> - Verificar que se muestren las imágenes de cada curso del OCW. 	

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-149: Actividades de HU14

Fecha	Actividad	Puntos estimados
27/11/15 al 30/11/15	Tarea 1: Crear los métodos necesarios para mostrar la galería de imágenes de un curso - Establecer reuniones con el Ing Jorge Huilca. - Desarrollar los métodos para mostrar imágenes de un curso - Documentación	16

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-150: Tarea 1 de la HU14

TAREA DE INGENIERIA	
Sprint: 12	Número de Tarea: 1
Nombre Historia de Usuario: Mostrar la galería de imágenes de un curso (visitante).	
Nombre de la Tarea: Crear los métodos necesarios para mostrar la galería de imágenes de un curso (visitante).	
Tipo de Tarea: Desarrollo	Programador Responsable: Jhonny Guevara
Fecha Inicio: 27/11/15	Fecha Fin: 30/11/15
Descripción: Creación de los métodos necesarios para mostrar la galería de imágenes de un curso (visitante).	
Pruebas de Aceptación: - Verificar que se muestren las imágenes de cada curso del OCW.	

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-151: Prueba de aceptación 1 de HU14

PRUEBA DE ACEPTACIÓN	
Código: HU14.1	Nombre Historia de Usuario: Mostrar la galería de imágenes de un curso (visitante).
Nombre de la Prueba: Verificar que se muestren las imágenes de cada curso del OCW.	
Responsable: Jhonny Guevara	Fecha: 30/11/2015
Descripción: Al acceder al contenido de un curso deberán mostrarle las imágenes asociadas a ese curso.	
Condiciones de Ejecución: - Crear los métodos en el modelo - Crear los métodos en el controlador. - Crear la vista que permita mostrar los datos	

Pasos de Ejecución: <ul style="list-style-type: none"> - Abrir la página de inicio del sistema - Doble clic sobre un curso deseado desde un carrusel - Clic en la opción galería de imágenes
Resultado Esperado: <ul style="list-style-type: none"> - Despliegue de la galería de imágenes para cada curso en un carrusel.
Evaluación de la Prueba: Satisfactoria.

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

SPRINT 13

Tabla 2-152: HU15 Gestionar Cursos Visitante

HISTORIA DE USUARIO	
Número: 15	Sprint Asignado: 13
Fecha Inicio: 01/12/15	Fecha Fin: 02/12/15
Tiempo Estimado:	2 días
Responsable:	Carlos Mejía
Descripción: Como visitante necesito listar los docentes que imparten los cursos (visitante)	
Pruebas de Aceptación: <ul style="list-style-type: none"> - Verificar que se muestren los docentes del OCW. 	

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-153: Actividades de HU15

Fecha	Actividad	Puntos estimados
01/12/15 al 02/12/15	Tarea 1: Crear los métodos necesarios para listar los docentes que imparten los cursos. <ul style="list-style-type: none"> - Establecer reuniones con el Ing Jorge Huilca. - Desarrollar los métodos para listar docentes - Documentación 	16

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-154: Tarea 1 de la HU15

TAREA DE INGENIERIA	
Sprint: 13	Número de Tarea: 1
Nombre Historia de Usuario: Listar los docentes que imparten los cursos (visitante)	

Nombre de la Tarea: Crear los métodos necesarios para listar los docentes que imparten los cursos (visitante)	
Tipo de Tarea: Desarrollo	Programador Responsable: Carlos Mejía
Fecha Inicio: 01/12/15	Fecha Fin: 02/12/15
Descripción: Creación de los métodos necesarios para listar los docentes que imparten los cursos (visitante)	
Pruebas de Aceptación: Verificar que se muestren los docentes del OCW.	

Realizado por: Carlos Mejía, Jhonny Guevara, 2016

Tabla 2-155: Prueba de aceptación 1 de HU15

PRUEBA DE ACEPTACIÓN	
Código: HU15.1	Nombre Historia de Usuario: Listar los docentes que imparten los cursos (visitante)
Nombre de la Prueba: Verificar que se muestren los docentes del OCW.	
Responsable: Carlos Mejía	Fecha: 08/12/2015
Descripción: Al acceder a la página de inicio del OCW deberán mostrarse los docentes registrados en el sistema	
Condiciones de Ejecución:	
<ul style="list-style-type: none"> - Crear los métodos en el modelo - Crear los métodos en el controlador. - Crear la vista que permita mostrar los datos 	
Pasos de Ejecución:	
<ul style="list-style-type: none"> - Abrir la página de inicio del sistema 	
Resultado Esperado:	
<ul style="list-style-type: none"> - Despliegue de los docentes del OCW en un carrusel web. 	
Evaluación de la Prueba:	
Satisfactoria.	

Realizado por: Carlos Mejía, Jhonny Guevara, 2016

Tabla 2-156: HU16 Búsqueda Cursos

HISTORIA DE USUARIO	
Número: 16	Sprint Asignado: 13
Fecha Inicio: 03/12/15	Fecha Fin: 08/12/15
Tiempo Estimado:	4 días
Responsable:	Carlos Mejía
Descripción: Como visitante deseo buscar cursos por: facultad, escuela, carrera, nombre, nivel. (Visitante).	

Pruebas de Aceptación:

- Verificar que se muestren los cursos por facultad/escuela/carrera.
- Verificar que se muestren los cursos por nombre.
- Verificar que se muestren los cursos por nivel.

Realizado por: Carlos Mejía, Jhonny Guevara, 2016

Tabla 2-157: Actividades de HU16

Fecha	Actividad	Puntos estimados
03/12/15 al 08/12/15	Tarea 1: Crear los métodos necesarios para buscar cursos por: facultad, escuela, carrera, nombre, nivel. (Visitante). - Establecer reuniones con el Ing Jorge Huilca. - Desarrollar los métodos para buscar cursos - Documentación	32

Realizado por: Carlos Mejía, Jhonny Guevara, 2016

Tabla 2-158: Tarea 1 de la HU16

TAREA DE INGENIERIA	
Sprint: 13	Número de Tarea: 1
Nombre Historia de Usuario: Buscar cursos por: facultad, escuela, carrera, nombre, nivel. (Visitante).	
Nombre de la Tarea: Crear los métodos necesarios para buscar cursos por: facultad, escuela, carrera, nombre, nivel. (Visitante).	
Tipo de Tarea: Desarrollo	Programador Responsable: Carlos Mejía
Fecha Inicio: 03/12/15	Fecha Fin: 08/12/15
Descripción: Crear los métodos necesarios para buscar cursos por: facultad, escuela, carrera, nombre, nivel. (Visitante).	
Pruebas de Aceptación:	
<ul style="list-style-type: none"> - Verificar que se muestren los cursos por facultad/escuela/carrera. - Verificar que se muestren los cursos por nombre. - Verificar que se muestren los cursos por nivel. 	

Realizado por: Carlos Mejía, Jhonny Guevara, 2016

Tabla 2-159: Prueba de aceptación 1 de HU16

PRUEBA DE ACEPTACIÓN	
Código: HU16.1	Nombre Historia de Usuario: Buscar cursos por: facultad, escuela, carrera, nombre, nivel. (Visitante).
Nombre de la Prueba: Verificar que se muestren los cursos por facultad/escuela/carrera.	
Responsable: Carlos Mejía	Fecha: 08/12/2015
Descripción: Mostrar la lista de cursos por facultad/escuela/carrera	
Condiciones de Ejecución:	
<ul style="list-style-type: none"> - Crear los métodos en el modelo - Crear los métodos en el controlador. - Crear la vista que permita mostrar los datos 	
Pasos de Ejecución:	
<ul style="list-style-type: none"> - Abrir la página de inicio del sistema - Hacer clic en el menú curso - Hacer curso en facultad/escuela/carrera - Dar doble clic en una facultad/escuela/carrera 	
Resultado Esperado:	
<ul style="list-style-type: none"> - Despliegue del listado de curso por facultad/escuela/carrera 	
Evaluación de la Prueba: Satisfactoria.	

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-160: Prueba de aceptación 2 de HU16

PRUEBA DE ACEPTACIÓN	
Código: HU16.2	Nombre Historia de Usuario: Buscar cursos por: facultad, escuela, carrera, nombre, nivel. (Visitante).
Nombre de la Prueba: Verificar que se muestren los cursos por nombre.	
Responsable: Carlos Mejía	Fecha: 08/12/2015
Descripción: Mostrar la lista de cursos por facultad/escuela/carrera con un nombre específico	
Condiciones de Ejecución:	
<ul style="list-style-type: none"> - Crear los métodos en el modelo - Crear los métodos en el controlador. - Crear la vista que permita mostrar los datos 	
Pasos de Ejecución:	
<ul style="list-style-type: none"> - Abrir la página de inicio del sistema - Hacer clic en el menú curso - Hacer curso en facultad/escuela/carrera - Dar doble clic en una facultad/escuela/carrera - Ingresar un nombre en el campo título. 	
Resultado Esperado:	
<ul style="list-style-type: none"> - Despliegue del listado de curso por facultad/escuela/carrera con un nombre específico 	
Evaluación de la Prueba: Satisfactoria.	

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-161: Prueba de aceptación 3 de HU16

PRUEBA DE ACEPTACIÓN	
Código: HU16.3	Nombre Historia de Usuario: Buscar cursos por: facultad, escuela, carrera, nombre, nivel. (Visitante).
Nombre de la Prueba: Verificar que se muestren los cursos por nivel.	
Responsable: Carlos Mejía	Fecha: 08/12/2015
Descripción: Mostrar la lista de cursos por facultad/escuela/carrera con un nivel específico	
Condiciones de Ejecución: <ul style="list-style-type: none"> - Crear los métodos en el modelo - Crear los métodos en el controlador. - Crear la vista que permita mostrar los datos 	
Pasos de Ejecución: <ul style="list-style-type: none"> - Abrir la página de inicio del sistema - Hacer clic en el menú curso - Hacer curso en facultad/escuela/carrera - Dar doble clic en una facultad/escuela/carrera - Ingresar un nombre en el campo nivel. 	
Resultado Esperado: <ul style="list-style-type: none"> - Despliegue del listado de curso por facultad/escuela/carrera con un nivel específico 	
Evaluación de la Prueba: Satisfactoria.	

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

SPRINT 14

Tabla 2-162: HU17 Detalle Curso Visitante

HISTORIA DE USUARIO	
Número: 17	Sprint Asignado: 14
Fecha Inicio: 01/12/15	Fecha Fin: 08/12/15
Tiempo Estimado:	6 días
Responsable:	Jhonny Guevara
Descripción: Como visitante necesito ver de un curso: el sílabo, calendario de actividades, material, capítulos, lecturas, temas, actividades, recursos (visitante).	
Pruebas de Aceptación: <ul style="list-style-type: none"> - Verificar que se muestre el sílabo/calendario de actividades. - Verificar que se muestre el material/capítulos/lecturas - Verificar que se muestren los recursos/temas/actividades 	

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-163: Actividades de HU17

Fecha	Actividad	Puntos estimados
01/12/15 al 08/12/15	Tarea 1: Crear los métodos necesarios para ver de un curso: el sílabo, calendario de actividades, material, capítulos, lecturas, temas, actividades, recursos (visitante). <ul style="list-style-type: none"> - Establecer reuniones con el Ing Jorge Huilca. - Desarrollar los métodos para mostrar información del curso - Documentación 	48

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-164: Tarea 1 de la HU17

TAREA DE INGENIERIA	
Sprint: 14	Número de Tarea: 1
Nombre Historia de Usuario: Mostrar de un curso: el sílabo, calendario de actividades, material, capítulos, lecturas, temas, actividades, recursos (visitante).	
Nombre de la Tarea: Crear los métodos necesarios para mostrar de un curso: el sílabo, calendario de actividades, material, capítulos, lecturas, temas, actividades, recursos (visitante).	
Tipo de Tarea: Desarrollo	Programador Responsable: Jhonny Guevara
Fecha Inicio: 01/12/15	Fecha Fin: 08/12/15
Descripción: Creación de los métodos necesarios para mostrar la información de un curso.	
Pruebas de Aceptación:	
<ul style="list-style-type: none"> - Verificar que se muestre el sílabo/calendario de actividades. - Verificar que se muestre el material/capítulos/lecturas - Verificar que se muestren los recursos/temas/actividades 	

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-165: Prueba de aceptación 1 de HU17

PRUEBA DE ACEPTACIÓN	
Código: HU17.1	Nombre Historia de Usuario: Mostrar de un curso: el sílabo, calendario de actividades, material, capítulos, lecturas, temas, actividades, recursos (visitante).
Nombre de la Prueba: Verificar que se muestre el sílabo/calendario de actividades.	
Responsable: Jhonny Guevara	Fecha: 08/12/2015
Descripción: Al acceder a la página de contenidos de un curso se desea visualizar el sílabo y calendario de actividades	
Condiciones de Ejecución:	
<ul style="list-style-type: none"> - Crear los métodos en el modelo - Crear los métodos en el controlador. - Crear la vista que permita mostrar los datos 	

Pasos de Ejecución:
<ul style="list-style-type: none"> - Abrir la página de inicio del sistema - Clic sobre el curso que se desee visualizar
Resultado Esperado: - Despliegue del sílabo y calendario de un curso seleccionado
Evaluación de la Prueba: Satisfactoria.

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-166: Prueba de aceptación 2 de HU17

PRUEBA DE ACEPTACIÓN	
Código: HU17.2	Nombre Historia de Usuario: Mostrar de un curso: el sílabo, calendario de actividades, material, capítulos, lecturas, temas, actividades, recursos (visitante).
Nombre de la Prueba: Verificar que se muestre el material/capítulos/lecturas	
Responsable: Jhonny Guevara	Fecha: 08/12/2015
Descripción: Al acceder a la página de contenidos de un curso se desea visualizar el material/capítulos/lecturas	
Condiciones de Ejecución:	
<ul style="list-style-type: none"> - Crear los métodos en el modelo - Crear los métodos en el controlador. - Crear la vista que permita mostrar los datos 	
Pasos de Ejecución:	
<ul style="list-style-type: none"> - Abrir la página de inicio del sistema - Clic sobre el curso que se desee visualizar - Clic sobre el botón material/capítulos/lecturas 	
Resultado Esperado:	
- Despliegue del material/capítulos/lectura de un curso seleccionado	
Evaluación de la Prueba:	
Satisfactoria.	

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-167: Prueba de aceptación 3 de HU17

PRUEBA DE ACEPTACIÓN	
Código: HU17.3	Nombre Historia de Usuario: Mostrar de un curso: el sílabo, calendario de actividades, material, capítulos, lecturas, temas, actividades, recursos (visitante).
Nombre de la Prueba: Verificar que se muestren los recursos/temas/actividades	
Responsable: Jhonny Guevara	Fecha: 08/12/2015
Descripción: Al acceder a la página de contenidos de un curso se desea visualizar los recursos/temas/actividades	
Condiciones de Ejecución:	
<ul style="list-style-type: none"> - Crear los métodos en el modelo - Crear los métodos en el controlador. - Crear la vista que permita mostrar los datos 	

Pasos de Ejecución:
<ul style="list-style-type: none"> - Abrir la página de inicio del sistema - Clic sobre el curso que se desee visualizar - Clic sobre el botón recursos/temas/actividades
Resultado Esperado:
- Despliegue de recursos/temas/actividades de un curso seleccionado
Evaluación de la Prueba:
Satisfactoria.

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

SPRINT 15

Tabla 2-168: HU18 Integrar los servicios web de la Facultad

HISTORIA DE USUARIO	
Número: 18	Sprint Asignado: 19
Fecha Inicio: 09/12/15	Fecha Fin: 11/12/15
Tiempo Estimado:	3 días
Responsable:	Carlos Mejía
Descripción: Como Administrador solicito visualizar las facultades del OASIS para el OCW.	
Pruebas de Aceptación:	
<ul style="list-style-type: none"> - Verificar que se muestren las facultades en el ocw. 	

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-169: Actividades de HU18

Fecha	Actividad	Puntos Estimados
09/12/15 al 11/12/15	Tarea 1: Crear los métodos necesarios para integrar y visualizar las facultades de la ESPOCH <ul style="list-style-type: none"> - Establecer reuniones con el Ing Jorge Huilca. - Desarrollar los métodos para obtener y mostrar las facultades de la ESPOCH 	24

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-170: Tarea 1 de la HU18

TAREA DE INGENIERIA	
Sprint: 15	Número de Tarea: 1
Nombre Historia de Usuario: Integrar los servicios web de la Facultad.	
Nombre de la Tarea: Crear los métodos necesarios para integrar y visualizar las facultades de la ESPOCH	

Tipo de Tarea: Desarrollo	Programador Responsable: Carlos Mejia
Fecha Inicio: 09/12/15	Fecha Fin: 11/12/15
Descripción: Creación de los métodos necesarios para integrar y visualizar las facultades de la ESPOCH.	
Pruebas de Aceptación:	
<ul style="list-style-type: none"> - Verificar que se muestren las facultades del oasis en el ocw. 	

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-171: Prueba de aceptación 1 de HU18

PRUEBA DE ACEPTACIÓN	
Código: HU18.1	Nombre Historia de Usuario: Integrar los servicios web de la Facultad.
Nombre de la Prueba: Verificar que se muestren las facultades del oasis en el ocw.	
Responsable: Carlos Mejía	Fecha: 16/12/2015
Descripción: Como administrador deseo mostrar la lista de facultades del oasis en el sistema OCW.	
Condiciones de Ejecución:	
<ul style="list-style-type: none"> - Crear los métodos en el modelo - Crear los métodos en el controlador. - Crear la vista que permita mostrar los datos 	
Pasos de Ejecución:	
<ul style="list-style-type: none"> - Abrir la página de inicio del sistema - Autenticarse como docente 	
Resultado Esperado: Despliegue facultades del docente autenticado.	
Evaluación de la Prueba: Satisfactoria.	

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-172: HU19 Integrar los servicios web de la Escuela.

HISTORIA DE USUARIO	
Número: 19	Sprint Asignado: 15
Fecha Inicio: 14/12/15	Fecha Fin: 16/12/15
Tiempo Estimado:	3 días
Responsable:	Carlos Mejía
Descripción: Como Administrador solicito visualizar las escuelas del OASIS para el OCW.	
Pruebas de Aceptación:	
<ul style="list-style-type: none"> - Verificar que se muestren las escuelas en el ocw. 	

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-173: Actividades de HU19

Fecha	Actividad	Puntos estimados
14/12/15 al 16/12/15	Tarea 1: Crear los métodos necesarios para integrar y visualizar las escuelas de la ESPOCH <ul style="list-style-type: none"> - Establecer reuniones con el Ing Jorge Huilca. - Desarrollar los métodos para obtener y mostrar las escuelas de la ESPOCH - Documentación 	24

Realizado por: Carlos Mejía, Jhonny Guevara, 2016

Tabla 2-174: Tarea 1 de la HU19

TAREA DE INGENIERIA	
Sprint: 15	Número de Tarea: 1
Nombre Historia de Usuario: Integrar los servicios web de la escuela.	
Nombre de la Tarea: Crear los métodos necesarios para integrar y visualizar las escuelas de la ESPOCH	
Tipo de Tarea: Desarrollo	Programador Responsable: Carlos Mejía
Fecha Inicio: 14/12/15	Fecha Fin: 16/12/15
Descripción: Creación de los métodos necesarios para integrar y visualizar las escuelas de la ESPOCH.	
Pruebas de Aceptación:	
<ul style="list-style-type: none"> - Verificar que se muestren las escuelas del oasis en el ocw. 	

Realizado por: Carlos Mejía, Jhonny Guevara, 2016

Tabla 2-175: Prueba de aceptación 1 de HU19

PRUEBA DE ACEPTACIÓN	
Código: HU19.1	Nombre Historia de Usuario: Integrar los servicios web de la escuela.
Nombre de la Prueba: Verificar que se muestren las escuelas del oasis en el ocw.	
Responsable: Carlos Mejía	Fecha: 16/12/2015
Descripción: Como administrador deseo mostrar la lista de escuelas del oasis en el sistema OCW.	
Condiciones de Ejecución:	
<ul style="list-style-type: none"> - Crear los métodos en el modelo - Crear los métodos en el controlador. - Crear la vista que permita mostrar los datos 	
Pasos de Ejecución:	
<ul style="list-style-type: none"> - Abrir la página de inicio del sistema - Autenticarse como docente 	
Resultado Esperado:	
<ul style="list-style-type: none"> - Despliegue escuelas del docente autenticado. 	
Evaluación de la Prueba:	
Satisfactoria.	

Realizado por: Carlos Mejía, Jhonny Guevara, 2016

SPRINT 16

Tabla 2-176: HU20 Integrar los servicios web del período.

HISTORIA DE USUARIO	
Número: 20	Sprint Asignado: 19
Fecha Inicio: 09/12/15	Fecha Fin: 11/12/15
Tiempo Estimado:	3 días
Responsable:	Jhonny Guevara
Descripción: Como Administrador solicito visualizar los períodos del OASIS para el OCW.	
Pruebas de Aceptación:	
- Verificar que se muestren las escuelas en el ocw.	

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-177: Actividades de HU20

Fecha	Actividad	Puntos estimados
09/12/15 al 11/12/15	Tarea 1: Crear los métodos necesarios para integrar y visualizar los períodos de la ESPOCH - Establecer reuniones con el Ing Jorge Huilca. - Desarrollar los métodos para obtener y mostrar los períodos de la ESPOCH	24

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-178: Tarea 1 de la HU20

TAREA DE INGENIERIA	
Sprint: 16	Número de Tarea: 1
Nombre Historia de Usuario: Integrar los servicios web del período.	
Nombre de la Tarea: Crear los métodos necesarios para integrar y visualizar los períodos de la ESPOCH	
Tipo de Tarea: Desarrollo	Programador Responsable: Jhonny Guevara
Fecha Inicio: 09/12/15	Fecha Fin: 11/12/15
Descripción: Creación de los métodos necesarios para integrar y visualizar los períodos de la ESPOCH.	
Pruebas de Aceptación:	
- Verificar que se muestren los períodos del oasis en el ocw.	

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-179: Prueba de aceptación 1 de HU20

PRUEBA DE ACEPTACIÓN	
Código: HU20.1	Nombre Historia de Usuario: Integrar los servicios web del período.
Nombre de la Prueba: Verificar que se muestren los períodos del oasis en el ocw.	
Responsable: Jhonny Guevara	Fecha: 16/12/2015
Descripción: Como administrador deseo mostrar la lista de períodos del oasis en el sistema OCW.	
Condiciones de Ejecución:	
<ul style="list-style-type: none"> - Crear los métodos en el modelo - Crear los métodos en el controlador. - Crear la vista que permita mostrar los datos 	
Pasos de Ejecución:	
<ul style="list-style-type: none"> - Abrir la página de inicio del sistema - Autenticarse como administrador - Clic sobre crear curso 	
Resultado Esperado: - Despliegue períodos del OASIS en el OCW	
Evaluación de la Prueba Satisfactoria.	

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-180: HU21 Integrar los servicios web del docente.

HISTORIA DE USUARIO	
Número: 21	Sprint Asignado: 16
Fecha Inicio: 14/12/15	Fecha Fin: 16/12/15
Tiempo Estimado:	3 días
Responsable:	Jhonny Guevara
Descripción: Como Administrador solicito visualizar los docentes del OASIS para el OCW.	
Pruebas de Aceptación:	
<ul style="list-style-type: none"> - Verificar que se muestren los docentes en el ocw. 	

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-181: HU21 Actividades de HU21

Fecha	Actividad	Puntos
14/12/15 al 16/12/15	Tarea 1: Crear los métodos necesarios para integrar y visualizar los docentes de la ESPOCH <ul style="list-style-type: none"> - Establecer reuniones con el Ing Jorge Huilca. - Desarrollar los métodos para obtener y mostrar los docentes de 	24

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-182: Tarea 1 de la HU21

TAREA DE INGENIERIA	
Sprint: 16	Número de Tarea: 1
Nombre Historia de Usuario: Integrar los servicios web del docente.	
Nombre de la Tarea: Crear los métodos necesarios para integrar y visualizar los docentes de la ESPOCH	
Tipo de Tarea: Desarrollo	Programador Responsable: Jhonny Guevara
Fecha Inicio: 14/12/15	Fecha Fin: 16/12/15
Descripción: Creación de los métodos necesarios para integrar y visualizar los docentes de la ESPOCH.	
Pruebas de Aceptación:	
<ul style="list-style-type: none"> - Verificar que se muestren los docentes del oasis en el ocw. 	

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-183: Prueba de aceptación 1 de HU21

PRUEBA DE ACEPTACIÓN	
Código: HU21.1	Nombre Historia de Usuario: Integrar los servicios web del docente.
Nombre de la Prueba: Verificar que se muestren los docentes del oasis en el ocw.	
Responsable: Jhonny Guevara	Fecha: 16/12/2015
Descripción: Como administrador deseo mostrar la lista de docentes del oasis en el sistema OCW.	
Condiciones de Ejecución:	
<ul style="list-style-type: none"> - Crear los métodos en el modelo - Crear los métodos en el controlador. - Crear la vista que permita mostrar los datos 	
Pasos de Ejecución:	
<ul style="list-style-type: none"> - Abrir la página de inicio del sistema - Autenticarse como administrador - Clic sobre crear docente - Clic en nuevo 	
Resultado Esperado:	
<ul style="list-style-type: none"> - Despliegue docentes del OASIS en el OCW 	
Evaluación de la Prueba:	
Satisfactoria.	

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

SPRINT 17

Tabla 2-184: HU22 Gestionar Preguntas-Examen

HISTORIA DE USUARIO	
Número: 22	Sprint Asignado: 20
Fecha Inicio: 17/12/15	Fecha Fin: 24/12/15
Tiempo Estimado:	6 días
Responsable:	Jhonny Guevara
Descripción: Como docente necesito Ingresar, editar, listar y visualizar las preguntas de un examen (Docente).	
Pruebas de Aceptación:	
<ul style="list-style-type: none"> - Verificar el funcionamiento de los métodos para gestionar pregunta. - Mensaje de confirmación al ingresar/editar pregunta. - Verificar que las preguntas se encuentren en la lista 	

Realizado por: Carlos Mejía, Jhonny Guevara, 2016

Tabla 2-185: Actividades de HU22

Fecha	Actividad	Puntos
17/12/15 al 19/12/15	Tarea 1: Crear los métodos necesarios para ingresar y listar preguntas <ul style="list-style-type: none"> - Establecer reuniones con el Ing Jorge Huilca. - Desarrollar los métodos para ingresar y listar preguntas - Documentación 	24
22/12/15 al 24/12/15	Tarea 2: Crear los métodos necesarios para modificar la pregunta <ul style="list-style-type: none"> - Identificar los campos que se modificaran. - Desarrollar los métodos para modificar pregunta - Documentación 	24

Realizado por: Carlos Mejía, Jhonny Guevara, 2016

Tabla 2-186: Tarea 1 de la HU22

TAREA DE INGENIERIA	
Sprint: 20	Número de Tarea: 1
Nombre Historia de Usuario: Ingresar, editar, listar y visualizar las preguntas de un examen.	
Nombre de la Tarea: Crear los métodos necesarios para ingresar y listar preguntas	
Tipo de Tarea: Desarrollo	Programador Responsable: Carlos Mejía
Fecha Inicio: 17/12/15	Fecha Fin: 19/12/15

Descripción: Creación de los métodos necesarios para ingresar y listar preguntas
Pruebas de Aceptación:
<ul style="list-style-type: none"> - Verificar que la nueva pregunta se encuentre en la lista. - Validación de los campos requeridos en la GUI.
Realizado por: Carlos Mejía, Jhonny Guevara, 2016

Tabla 2-187: Tarea 2 de la HU22

TAREA DE INGENIERIA	
Sprint: 20	Número de Tarea: 2
Nombre Historia de Usuario: Ingresar, editar, listar y visualizar las preguntas de un examen	
Nombre de la Tarea: Crear los métodos necesarios para modificar preguntas.	
Tipo de Tarea: Desarrollo	Programador Responsable: Carlos Mejía
Fecha Inicio: 22/12/15	Fecha Fin: 24/12/15
Descripción: Creación de los métodos necesarios para editar preguntas de un examen	
Pruebas de Aceptación:	
<ul style="list-style-type: none"> - Verificación de los métodos para editar pregunta. - Mensaje de confirmación cuando se modifique/ingrese pregunta. 	

Realizado por: Carlos Mejía, Jhonny Guevara, 2016

Tabla 2-188: Prueba de aceptación 1 de HU22

PRUEBA DE ACEPTACIÓN	
Código: HU22.1	Nombre Historia de Usuario: Ingresar, editar, listar y visualizar las preguntas de un examen
Nombre de la Prueba: Verificar que la nueva pregunta se encuentre en la lista.	
Responsable: Carlos Mejía	Fecha: 24/12/2015
Descripción: Al ingresar una pregunta es necesario mostrarla en una lista de preguntas para corroborar que se encuentre insertada	
Condiciones de Ejecución:	
<ul style="list-style-type: none"> - Crear los métodos en el modelo - Crear los métodos en el controlador. - Crear la vista que permita recopilar los datos 	
Pasos de Ejecución:	
<ul style="list-style-type: none"> - Clic en el menú pregunta - Clic en el botón pregunta nueva - Llenar los campos requeridos 	
Resultado Esperado:	
<ul style="list-style-type: none"> - Guardado y despliegue de la pregunta en la lista. 	
Evaluación de la Prueba:	
Satisfactoria.	

Realizado por: Carlos Mejía, Jhonny Guevara, 2016

Tabla 2-189: Prueba de aceptación 2 de HU22

PRUEBA DE ACEPTACIÓN	
Código: HU22.2	Nombre Historia de Usuario: Ingresar, editar, listar y visualizar las preguntas de un examen
Nombre de la Prueba: Validación de los campos requeridos en la GUI.	
Responsable: Carlos Mejía	Fecha: 24/12/2015
Descripción: Cuando se desee guardar una pregunta se debe controlar que previo se llenen los campos requeridos para garantizar la integridad de los datos	
Condiciones de Ejecución: <ul style="list-style-type: none"> - Crear el método para validar campos en el controlador. - Crear el método en el paquete de utilidades para mostrar el mensaje 	
Pasos de Ejecución: <ul style="list-style-type: none"> - Clic en el menú pregunta - Clic en el botón pregunta nueva - No llenar algún campo requerido (descripción) - Clic en el botón guardar 	
Resultado Esperado: <ul style="list-style-type: none"> - Despliegue del mensaje que alerte la falta de campos requeridos 	
Evaluación de la Prueba: Satisfactoria	

Realizado por: Carlos Mejía, Jhonny Guevara, 2016

Tabla 2-190: Prueba de aceptación 3 de HU22

PRUEBA DE ACEPTACIÓN	
Código: HU22.3	Nombre Historia de Usuario: Ingresar, editar, listar y visualizar las preguntas de un curso.
Nombre de la Prueba: Verificar que la pregunta editada se encuentre en la lista.	
Responsable: Carlos Mejía	Fecha: 24/12/2015
Descripción: Al editar una pregunta es necesario mostrarla en una lista de preguntas para corroborar que se encuentre modificada	
Condiciones de Ejecución: <ul style="list-style-type: none"> - Crear los métodos en el modelo - Crear los métodos en el controlador. - Crear la vista que permita recopilar los datos 	
Pasos de Ejecución: <ul style="list-style-type: none"> - Clic en el menú pregunta - Clic en el botón editar pregunta - Llenar los campos requeridos - Clic en el botón guardar 	
Resultado Esperado: <ul style="list-style-type: none"> - Guardado y despliegue de la pregunta en la lista. 	
Evaluación de la Prueba: Satisfactoria.	

Realizado por: Carlos Mejía, Jhonny Guevara, 2016

Tabla 2-191: Prueba de aceptación 4 de HU22

PRUEBA DE ACEPTACIÓN	
Código: HU22.4	Nombre Historia de Usuario Ingresar, editar, listar y visualizar las preguntas de un examen
Nombre de la Prueba: Mensaje de confirmación cuando se ingrese/modifique la pregunta	
Responsable: Carlos Mejía	Fecha: 24/12/2015
Descripción: Al ingresar/modificar una pregunta se debe mostrar un mensaje informativo que los datos han sido guardados	
Condiciones de Ejecución: <ul style="list-style-type: none"> - Crear el método para validar campos en el controlador. - Crear el método en el paquete de utilidades para mostrar el mensaje 	
Pasos de Ejecución: <ul style="list-style-type: none"> - Clic en el menú pregunta - Clic en el botón pregunta nueva - Llenar los campos requeridos 	
Resultado Esperado: Despliegue del mensaje que informe de la modificación/ingreso exitoso de datos	
Evaluación de la Prueba: Satisfactoria	

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

SPRINT 18

Tabla 2-192: HU23 Gestionar Examen Docente

HISTORIA DE USUARIO	
Número: 23	Sprint Asignado: 18
Fecha Inicio: 17/12/15	Fecha Fin: 21/12/15
Tiempo Estimado:	3 días
Responsable:	Jhonny Guevara
Descripción: Como docente necesito Ingresar, editar, listar y visualizar un examen (Docente).	
Pruebas de Aceptación: <ul style="list-style-type: none"> - Verificar el funcionamiento de los métodos para gestionar examen. - Mensaje de confirmación al ingresar/editar examen. - Verificar que los exámenes se encuentren en la lista 	

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-193: Actividades de HU23

Fecha	Actividad	Puntos estimados
17/12/15 al 21/12/15	Tarea 1: Crear los métodos necesarios para ingresar, modificar y listar examen - Establecer reuniones con el Ing Jorge Huilca. Identificar los campos que se modificarán. - Desarrollar los métodos para gestionar examen - Documentación	24

Realizado por: Carlos Mejía, Jhonny Guevara, 2016

Tabla 2-194: Tarea 1 de la HU23

TAREA DE INGENIERIA	
Sprint: 18	Número de Tarea: 1
Nombre Historia de Usuario: Ingresar, editar, listar y visualizar un examen.	
Nombre de la Tarea: Crear los métodos necesarios para ingresar y listar examen	
Tipo de Tarea: Desarrollo	Programador Responsable: Jhonny Guevara
Fecha Inicio: 17/12/15	Fecha Fin: 21/12/15
Descripción: Creación de los métodos necesarios para ingresar, editar y listar examen	
Pruebas de Aceptación:	
<ul style="list-style-type: none"> - Verificar que el nuevo examen se encuentre en la lista. - Validación de los campos requeridos en la GUI. - Verificación de los métodos para editar examen. - Mensaje de confirmación cuando se modifique/ingrese examen. 	

Realizado por: Carlos Mejía, Jhonny Guevara, 2016

Tabla 2-195: Prueba de aceptación 1 de HU23

PRUEBA DE ACEPTACIÓN	
Código: HU23.1	Nombre Historia de Usuario: Ingresar, editar, listar y visualizar las preguntas de un examen
Nombre de la Prueba: Verificar que el nuevo examen se encuentre en la lista.	
Responsable: Jhonny Guevara	Fecha: 24/12/2015
Descripción: Al ingresar un examen es necesario mostrarlo en una lista de exámenes para corroborar que se encuentre insertado	
Condiciones de Ejecución:	
<ul style="list-style-type: none"> - Crear los métodos en el modelo - Crear los métodos en el controlador. - Crear la vista que permita recopilar los datos 	

Pasos de Ejecución: <ul style="list-style-type: none"> - Clic en el menú examen - Clic en el botón examen nuevo - Llenar los campos requeridos - Clic en el botón guardar
Resultado Esperado: <ul style="list-style-type: none"> - Guardado y despliegue del examen en la lista.
Evaluación de la Prueba: Satisfactoria.

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-196: Prueba de aceptación 2 de HU23

PRUEBA DE ACEPTACIÓN	
Código: HU23.2	Nombre Historia de Usuario: Ingresar, editar, listar y visualizar un examen
Nombre de la Prueba: Validación de los campos requeridos en la GUI.	
Responsable: Jhonny Guevara	Fecha: 24/01/2016
Descripción: Cuando se desee guardar un examen se debe controlar que previo se llenen los campos requeridos para garantizar la integridad de los datos	
Condiciones de Ejecución: <ul style="list-style-type: none"> - Crear el método para validar campos en el controlador. - Crear el método en el paquete de utilidades para mostrar el mensaje 	
Pasos de Ejecución: <ul style="list-style-type: none"> - Clic en el menú examen - Clic en el botón examen nuevo - No llenar algún campo requerido (descripción) - Clic en el botón guardar 	
Resultado Esperado: <ul style="list-style-type: none"> - Despliegue del mensaje que alerte la falta de campos requeridos 	
Evaluación de la Prueba: Satisfactoria	

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-197: Prueba de aceptación 3 de HU23

PRUEBA DE ACEPTACIÓN	
Código: HU23.3	Nombre Historia de Usuario: Ingresar, editar, listar y visualizar un curso.
Nombre de la Prueba: Verificar que el examen editado se encuentre en la lista.	
Responsable: Jhonny Guevara	Fecha: 24/12/2015
Descripción: Al editar un examen es necesario mostrarlo en una lista de exámenes para corroborar que se encuentre modificado	

Condiciones de Ejecución: <ul style="list-style-type: none"> - Crear los métodos en el modelo - Crear los métodos en el controlador. - Crear la vista que permita recopilar los datos
Pasos de Ejecución: <ul style="list-style-type: none"> - Clic en el menú examen - Clic en el botón editar examen - Llenar los campos requeridos - Clic en el botón guardar
Resultado Esperado: Guardado y despliegue del examen en la lista.
Evaluación de la Prueba: Satisfactoria.

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-198: Prueba de aceptación 4 de HU23

PRUEBA DE ACEPTACIÓN	
Código: HU23.4	Nombre Historia de Usuario: Ingresar, editar, listar y visualizar un examen
Nombre de la Prueba: Mensaje de confirmación cuando se ingrese/modifique un examen	
Responsable: Jhonny Guevara	Fecha: 24/12/2015
Descripción: Al ingresar/modificar un examen se debe mostrar un mensaje informativo que los datos han sido guardados	
Condiciones de Ejecución: <ul style="list-style-type: none"> - Crear el método para validar campos en el controlador. - Crear el método en el paquete de utilidades para mostrar el mensaje 	
Pasos de Ejecución: <ul style="list-style-type: none"> - Clic en el menú examen - Clic en el botón examen nueva - Llenar los campos requeridos 	
Resultado Esperado: Despliegue del mensaje que informe de la modificación/ingreso exitoso de datos	
Evaluación de la Prueba: Satisfactoria	

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-199: HU24 Ingreso Auditoria

HISTORIA DE USUARIO	
Número: 24	Sprint Asignado: 18
Fecha Inicio: 22/12/15	Fecha Fin: 23/12/15
Tiempo Estimado:	2 días
Responsable:	Jhonny Guevara
Descripción: Como administrador necesito se registre la actividad que realizan los usuarios sobre el sistema	

Pruebas de Aceptación: <ul style="list-style-type: none"> - Verificar el funcionamiento de los métodos para ingresar auditoría. - Verificar que la auditoría se encuentre en la base de datos
--

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-200: Actividades de HU24

Fecha	Actividad	Puntos
22/12/15 al 23/12/15	Tarea 1: Crear los métodos necesarios para ingresar auditoría - Establecer reuniones con el Ing Jorge Huilca. examen - Documentación	16

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-201: Tarea 1 de la HU24

TAREA DE INGENIERIA	
Sprint: 18	Número de Tarea: 1
Nombre Historia de Usuario: Ingresar el registro de la auditoría.	
Nombre de la Tarea: Crear los métodos necesarios para ingresar auditoría	
Tipo de Tarea: Desarrollo	Programador Responsable: Jhonny Guevara
Fecha Inicio: 22/12/15	Fecha Fin: 23/12/15
Descripción: Creación de los métodos necesarios para ingresar auditoría	
Pruebas de Aceptación: <ul style="list-style-type: none"> - Verificar el funcionamiento de los métodos para ingresar auditoría. - Verificar que la auditoría se encuentre en la base de datos 	

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-202: Prueba de aceptación 1 de HU24

PRUEBA DE ACEPTACIÓN	
Código: HU24.1	Nombre Historia de Usuario: Ingresar el registro de la auditoría.
Nombre de la Prueba: Verificar el funcionamiento de los métodos para ingresar auditoría.	
Responsable: Jhonny Guevara	Fecha: 24/12/2015
Descripción: Al ingresar realizar alguna actividad del OCW se vuelve necesario registrarla en la base de datos para llevar un seguimiento y seguridad de los datos	

Condiciones de Ejecución: <ul style="list-style-type: none"> - Crear los métodos en el modelo - Crear los métodos en el controlador.
Pasos de Ejecución: <ul style="list-style-type: none"> - Autenticarse como docente - Clic en el menú examen - Clic en el botón examen nuevo - Llenar los campos requeridos - Clic en el botón guardar
Resultado Esperado: <ul style="list-style-type: none"> - Guardado de la actividad de ingreso de un examen en la base de datos.
Evaluación de la Prueba: Satisfactoria.

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-203: Prueba de aceptación 2 de HU24

PRUEBA DE ACEPTACIÓN	
Código: HU24.2	Nombre Historia de Usuario: Ingresar el registro de la auditoría.
Nombre de la Prueba: Verificar que la auditoría se encuentre en la base de datos	
Responsable: Jhonny Guevara	Fecha: 24/12/2015
Descripción: Luego de realizar una acción sobre la aplicación ocw Epoch se debe registrar la actividad en la base de datos para garantizar la integridad de la información	
Condiciones de Ejecución: <ul style="list-style-type: none"> - Ingresar un examen desde el Ocw 	
Pasos de Ejecución: <ul style="list-style-type: none"> - Abrir el administrador de Postgresql (pgadim) - Conectarse a la base de datos (OpenCourseWareEpoch) - Expandir el esquema auditoria - Mostrar los registros de la tabla t_auditoria 	
Resultado Esperado: <ul style="list-style-type: none"> - Despliegue del registro de la actividad realizada 	
Evaluación de la Prueba: Satisfactoria	

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-204:HU25 Ingresar el registro de error.

HISTORIA DE USUARIO	
Número: 25	Sprint Asignado: 18
Fecha Inicio: 24/12/15	Fecha Fin: 24/12/15
Tiempo Estimado:	1 día
Responsable:	Jhonny Guevara

Descripción: Como administrador necesito se registre errores en la actividad que realizan los usuarios sobre el sistema
Pruebas de Aceptación:
<ul style="list-style-type: none"> - Verificar el funcionamiento de los métodos para ingresar error. - Verificar que el registro de error se encuentre en la base de datos

Realizado por: Carlos Mejía, Jhonny Guevara, 2016

Tabla 2-205: Actividades de HU25

Fecha	Actividad	Puntos Estimados
24/12/15 al	Tarea 1: Crear los métodos necesarios para ingresar error	8
24/12/15	<ul style="list-style-type: none"> - Establecer reuniones con el Ing Jorge Huilca. - Documentación 	

Realizado por: Carlos Mejía, Jhonny Guevara, 2016

Tabla 2-206: Tarea 1 de la HU25

TAREA DE INGENIERIA	
Sprint: 18	Número de Tarea: 1
Nombre Historia de Usuario: Ingresar el registro de error.	
Nombre de la Tarea: Crear los métodos necesarios para ingresar el registro de error	
Tipo de Tarea: Desarrollo	Programador Responsable: Jhonny Guevara
Fecha Inicio: 24/12/15	Fecha Fin: 24/12/15
Descripción: Creación de los métodos necesarios para ingresar de error	
Pruebas de Aceptación:	
<ul style="list-style-type: none"> - Verificar el funcionamiento de los métodos para registrar error. - Verificar que el registro de error se encuentre en la base de datos 	

Realizado por: Carlos Mejía, Jhonny Guevara, 2016

Tabla 2-207: Prueba de aceptación 1 de HU25

PRUEBA DE ACEPTACIÓN	
Código: HU25.1	Nombre Historia de Usuario: Ingresar el registro de error.
Nombre de la Prueba: Verificar el funcionamiento de los métodos para ingresar el registro de error.	
Responsable: Jhonny Guevara	Fecha: 24/12/2015
Descripción: Al ingresar fallar alguna actividad del OCW se vuelve necesario registrarla en la base de datos para llevar un seguimiento y seguridad de los datos	
Condiciones de Ejecución:	
<ul style="list-style-type: none"> - Crear los métodos en el modelo - Crear los métodos en el controlador. 	

Pasos de Ejecución: <ul style="list-style-type: none"> - Autenticarse como docente - Clic en el menú examen - Clic en el botón examen nuevo
Resultado Esperado: <ul style="list-style-type: none"> - Guardado del registro de error de ingreso de un examen en la base de datos.
Evaluación de la Prueba: Satisfactoria.

Realizado por: Carlos Mejía, Jhonny Guevara, 2016

Tabla 2-208: Prueba de aceptación 2 de HU25

PRUEBA DE ACEPTACIÓN	
Código: HU25.2	Nombre Historia de Usuario: Ingresar el registro de error
Nombre de la Prueba: Verificar que el registro de error se encuentre en la base de datos	
Responsable: Jhonny Guevara	Fecha: 24/12/2015
Descripción: Luego de realizar una acción y esta falle sobre la aplicación ocw Epoch se debe registrar el error en la base de datos para garantizar la integridad de la información	
Condiciones de Ejecución: <ul style="list-style-type: none"> - Ingresar un examen desde el Ocw - Dejar un campo en blanco en el formulario 	
Pasos de Ejecución: <ul style="list-style-type: none"> - Abrir el administrador de Postgresql (pgadim) - Conectarse a la base de datos (OpenCourseWareEpoch) - Expandir el esquema auditoria - Mostrar los registros de la tabla t_error 	
Resultado Esperado: <ul style="list-style-type: none"> - Despliegue del registro de error en la actividad realizada 	
Evaluación de la Prueba: Satisfactoria	

Realizado por: Carlos Mejía, Jhonny Guevara, 2016

IMPLANTACIÓN DEL SISTEMA

En el Sprint 19 del proyecto OCW, se realizó la implantación del servidor de producción del sistema; se seleccionó GNU/Linux CentOS 5.9 de 64 bits como el sistema operativo huésped con un disco duro de 2GB, ya que la ESPOCH maneja sus servidores de aplicaciones web creadas en java con sistemas operativos basados en GNU/LINUX; el servidor de aplicaciones Glassfish 4.0 fue el elegido para el alojamiento y despliegue de la aplicación web; y como servidor de bases de datos se seleccionó Postgresql por política de la ESPOCH de trabajar con este DBMS; para la gestión de contenidos que poseerá el OCW se seleccionó el directorio /recursos; el mismo que con datos de 5 usuarios su tamaño creció a 100MB incluyendo la base de datos por lo que se sugiere aumentar la capacidad de almacenamiento a 1TB.

SPRINT 19

Tabla 2-209: HT5 Preparación Servidor

HISTORIA DE USUARIO	
Número: HUT8	Sprint Asignado: 19
Fecha Inicio: 25/12/15	Fecha Fin: 30/12/15
Tiempo Estimado:	4 día
Responsable:	Team Development
Descripción: Como administrador del sistema solicito preparar un servidor con S.O GNU/LINUX para el despliegue de la aplicación	
Pruebas de Aceptación:	
<ul style="list-style-type: none"> - Verificar que se instale GlassFish 4.0. - Verificar que se encuentre instalada la Jdk 7 - Verificar que se pueda acceder a la consola de administrador de GlassFish 	

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-210: Actividades de HT5

Fecha	Actividad	Puntos estimados
25/12/15 al 30/12/15	Tarea 1: Preparar un servidor con S.O GNU/LINUX para el despliegue de la aplicación <ul style="list-style-type: none"> - Establecer reuniones con el Ing Jorge Huilca. - Instalar GlassFish 4.0/JDK 7 - Documentación 	32

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-211: Tarea 1 de la HT5

TAREA DE INGENIERIA	
Sprint: 19	Número de Tarea: 1
Nombre Historia de Usuario: Preparación del servidor para la aplicación	
Nombre de la Tarea: Preparar un servidor con S.O GNU/LINUX para el despliegue de la aplicación	
Tipo de Tarea: Desarrollo	Programador Responsable: team development
Fecha Inicio: 25/12/15	Fecha Fin: 30/12/15
Descripción: Instalar el software necesario (GlassFish4.0/Jdk7) para desplegar la aplicación OCW-ESPOCH	
Pruebas de Aceptación:	
<ul style="list-style-type: none"> - Verificar que se instale GlassFish 4.0. - Verificar que se encuentre instalada la Jdk 7 - Verificar que se pueda acceder a la consola de administrador de GlassFish 	

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-212: Prueba de aceptación 1 de HT5

PRUEBA DE ACEPTACIÓN	
Código: HT5.1	Nombre Historia de Usuario: Preparación del servidor para la aplicación.
Nombre de la Prueba: Verificar que se instale GlassFish 4.0.	
Responsable: team development	Fecha: 01/01/2016
Descripción: Se desea saber que se haya instalado el servidor GlassFish Para aplicaciones web	
Condiciones de Ejecución:	
<ul style="list-style-type: none"> - Descargar el instalador desde la página oficial - Instalar GlassFish 4.0 - Anclar GlassFish al arranque del sistema 	
Pasos de Ejecución:	
<ul style="list-style-type: none"> - Abrir el terminal de su distribución GNU/LINUX (DEBIAN) - Autenticarse como usuario root o privilegios similares - Ejecutar la sentencia: service glassfish restart 	
Resultado Esperado:	
<ul style="list-style-type: none"> - Reinicio del servicio GlassFish 	
Evaluación de la Prueba:	
Satisfactoria.	

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-213: Prueba de aceptación 2 de HT5

PRUEBA DE ACEPTACIÓN	
Código: HT5.2	Nombre Historia de Usuario: Preparación del servidor para la aplicación.
Nombre de la Prueba: Verificar que se encuentre instalada la Jdk 7	
Responsable: Team Development	Fecha: 01/01/2016
Descripción: Se desea saber que se encuentre instalada la jdk7 sobre la cual fue desarrollado el sistema OCW-ESPOCH	
Condiciones de Ejecución:	
<ul style="list-style-type: none"> - Descargar el instalador desde la página oficial - Instalar jdk7 	
Pasos de Ejecución:	
<ul style="list-style-type: none"> - Abrir el terminal de su distribución GNU/LINUX (DEBIAN) - Autenticarse como usuario root o privilegios similares - Ejecutar la sentencia: java -version 	
Resultado Esperado:	
<ul style="list-style-type: none"> - Despliegue de la versión jdk7 instalada en el sistema operativo 	
Evaluación de la Prueba:	
Satisfactoria.	

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-214: Prueba de aceptación 3 de HT5

PRUEBA DE ACEPTACIÓN	
Código: HT5.3	Nombre Historia de Usuario: Preparación del servidor para la aplicación.
Nombre de la Prueba: Verificar que se pueda acceder a la consola de administrador de GlassFish	
Responsable: Team development	Fecha: 01/01/2016
Descripción: Al acceder a la dirección ip localhost:4848, mediante un navegador web se debe acceder a la consola de configuración de GlassFish	
Condiciones de Ejecución:	
<ul style="list-style-type: none"> - Instalar Jdk 7 - Instalar GlassFish 4 - Ejecutar el servicio de Glassfish 	
Pasos de Ejecución:	
<ul style="list-style-type: none"> - Abrir el navegador web preferido - Ingresar la url localhost:4848 - Ingresar las credenciales en el formulario de registro - Acceder a la consola 	
Resultado Esperado:	
<ul style="list-style-type: none"> - Despliegue de las herramientas de configuración web para GlassFish 	
Evaluación de la Prueba:	
Satisfactoria.	

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-215: HT5 Despliegue App

HISTORIA DE USUARIO	
Número: HT6	Sprint Asignado: 19
Fecha Inicio: 31/12/15	Fecha Fin: 01/01/16
Tiempo Estimado:	2 días
Responsable:	Team Development
Descripción: Como administrador del sistema solicito se despliegue de la aplicación en el servidor	
Pruebas de Aceptación:	
<ul style="list-style-type: none"> - Verificar que se cree la conexión JDBC con la base de datos. - Verificar que la aplicación se encuentre desplegada 	

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-216: Actividades de HT6

Fecha	Actividad	Puntos Estimados
31/12/15 al 01/01/16	Tarea 1: Desplegar la aplicación en el servidor GlassFish <ul style="list-style-type: none"> - Establecer reuniones con el Ing Jorge Huilca. - Desplegar el OCW-ESPOCH en el servidor GlassFish - Documentación 	16

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-217: Tarea 1 de la HT6

TAREA DE INGENIERIA	
Sprint: 19	Número de Tarea: 1
Nombre Historia de Usuario: Despliegue de la aplicación en el servidor	
Nombre de la Tarea: Desplegar la aplicación en el servidor GlassFish	
Tipo de Tarea: Desarrollo	Programador Responsable: Team Development
Fecha Inicio: 31/12/15	Fecha Fin: 01/01/16
Descripción: Instalar el software necesario (GlassFish4.0/Jdk7) para desplegar la aplicación OCW-ESPOCH	
Pruebas de Aceptación: <ul style="list-style-type: none"> - Verificar que se cree la conexión JDBC con la base de datos. - Verificar que la aplicación se encuentre desplegada 	

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-218: Prueba de aceptación 1 de HT6

PRUEBA DE ACEPTACIÓN	
Código: HT6.1	Nombre Historia de Usuario: Despliegue de la aplicación en el servidor
Nombre de la Prueba: Verificar que se cree la conexión JDBC con la base de datos.	
Responsable: Team development	Fecha: 01/01/2016
Descripción: Se desea verificar que la conexión a la base de datos se encuentre en funcionamiento para el OCW-ESPOCH	
Condiciones de Ejecución: <ul style="list-style-type: none"> - Servidor GlassFish ejecutándose - Acceder a la consola de configuración para GlassFish - Crear un nuevo pool de conexiones JDBC - Crear la unidad de persistencia JNDI 	
Pasos de Ejecución: <ul style="list-style-type: none"> - Acceder a los recursos JDBC del servidor Glassfish - Revisar que tenga conexión a la base de datos 	
Resultado Esperado: <ul style="list-style-type: none"> - Añadidura de la conexión JDBC en el servidor GlassFish 	
Evaluación de la Prueba: Satisfactoria.	

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-219: Prueba de aceptación 2 de HT6

PRUEBA DE ACEPTACIÓN	
Código: HT6.2	Nombre Historia de Usuario: Despliegue de la aplicación en el servidor
Nombre de la Prueba: Verificar que la aplicación se encuentre desplegada	
Responsable: Team development	Fecha: 01/01/2016
Descripción: Se desea comprobar que la aplicación OCW-ESPOCH se encuentre desplegada.	
Condiciones de Ejecución:	
<ul style="list-style-type: none"> - Generar el archivo .war para subirlo al servidor - Tener creada la conexión JDBC para la aplicación web 	
Pasos de Ejecución:	
<ul style="list-style-type: none"> - Abrir la consola de administración web de GlassFish - Hacer clic en la sección applications - Hacer clic en el botón deploy - Seleccionar la aplicación 	
Resultado Esperado:	
<ul style="list-style-type: none"> - Despliegue de la aplicación mostrándome la URL de acceso a ella 	
Evaluación de la Prueba:	
Satisfactoria.	

Realizado por: Carlos Mejía, Jhonny Guevara, 2016

CAPACITACIÓN DE USUARIOS

En el sprint 20 se realizó la capacitación a los usuarios del sistema, en total 6 personas, de las cuales 2 eran Docentes de la escuela de sistemas, 2 más eran docentes de la escuela de ingeniería electrónica y por último 2 eran docentes de la escuela de diseño gráfico, de esta capacitación se aprobó el sistema OCW para la ESPOCH por parte de los usuarios y se emitió como recomendación añadir más funcionalidad al sistema y añadir la opción para compartir los contenidos de los cursos del sistema en redes sociales y blogs; Se explicó que el sistema fue planificado hasta esa fase; por lo que recomendamos continuar con el desarrollo del sistema OCW-ESPOCH

SPRINT 20

Tabla 2-220: HT7 Pruebas App

HISTORIA DE USUARIO	
Número: HT7	Sprint Asignado: 20
Fecha Inicio: 04/01/16	Fecha Fin: 08/01/16
Tiempo Estimado:	5 días
Responsable:	Team Development

Descripción: Como administrador del sistema solicito se realicen pruebas de concurrencia a la aplicación.
Pruebas de Aceptación:
<ul style="list-style-type: none"> - Verificar que la aplicación se desempeñe correctamente con concurrencia simultanea de usuarios. - Verificar que la aplicación se encuentre desplegada

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-221: Actividades de HT7

Fecha	Actividad	Puntos Estimados
04/01/16 al 08/01/16	Tarea 1: Pruebas de aceptación en la aplicación. <ul style="list-style-type: none"> - Establecer reuniones con el Ing Jorge Huilca. - Test de la aplicación en laboratorios de la ESPOCH-ESPOCH - Documentación 	40

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-222: Tarea 1 de la HT7

TAREA DE INGENIERIA	
Sprint: 20	Número de Tarea: 1
Nombre Historia de Usuario: Pruebas de la Aplicación	
Nombre de la Tarea: Pruebas a la Aplicación	
Tipo de Tarea: Desarrollo	Programador Responsable: Team development
Fecha Inicio: 04/01/16	Fecha Fin: 08/01/16
Descripción: Como administrador deseo conocer si la aplicación web soportará la concurrencia múltiple de usuarios	
Pruebas de Aceptación: Verificar que la aplicación se desempeñe correctamente con concurrencia simultanea de usuarios.	

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-223: Prueba de aceptación 1 de HT7

PRUEBA DE ACEPTACIÓN	
Código: HT7.1	Nombre Historia de Usuario: Pruebas de la Aplicación
Nombre de la Prueba: Verificar que la aplicación se desempeñe correctamente con concurrencia simultanea de usuarios.	
Responsable: Team development	Fecha: 11/01/2016
Descripción: Se desea verificar si la aplicación web soportará la concurrencia múltiple de usuarios	
Condiciones de Ejecución:	
<ul style="list-style-type: none"> - Aplicación web desplegada - Poseer varios usuarios tipo docente en el sistema - Poseer varios usuarios tipo administrador en el sistema 	
Pasos de Ejecución:	
<ul style="list-style-type: none"> - Acceder a la aplicación web desde los distintos ordenadores de los laboratorios de la ESPOCH - Realizar operaciones de inserción, actualización y eliminación de elementos existentes en el OCW-ESPOCH 	
Resultado Esperado: Funcionamiento normal de la aplicación web	

Evaluación de la Prueba: Satisfactoria.

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-224: HT8 Capacitación Usuarios

HISTORIA DE USUARIO	
Número: HT8	Sprint Asignado: 20
Fecha Inicio: 11/01/16	Fecha Fin: 11/01/16
Tiempo Estimado:	1 día
Responsable:	Team Development
Descripción: Como administrador del sistema solicito capacitar a los usuarios del OCW-ESPOCH	
Pruebas de Aceptación:	
<ul style="list-style-type: none"> - Firmas de asistencia de los participantes en la capacitación del OCW-ESPOCH 	

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-225: Actividades de HT8

Fecha	Actividad	Puntos estimados
11/01/16 al 11/01/16	Tarea 1: Capacitar a los usuarios del OCW-ESPOCH <ul style="list-style-type: none"> - Establecer reuniones de coordinación con el Ing Jorge Huilca. - Capacitar a los usuarios en un laboratorio de la ESPOCH - Documentación 	8

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-226: Tarea 1 de la HT8

TAREA DE INGENIERIA	
Sprint: 20	Número de Tarea: 1
Nombre Historia de Usuario: Capacitación a usuarios	
Nombre de la Tarea: Capacitar a los usuarios del OCW-ESPOCH	
Tipo de Tarea: Desarrollo	Programador Responsable: Team Development
Fecha Inicio: 11/01/16	Fecha Fin: 11/01/16
Descripción: Se desea capacitar a los usuarios para operar el OCW-ESPOCH y aportar ideas para futuras mejoras.	
Pruebas de Aceptación:	
<ul style="list-style-type: none"> - Firma de asistencia de los participantes en la capacitación del OCW-ESPOCH 	

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-227: Prueba de aceptación 1 de HT8

PRUEBA DE ACEPTACIÓN	
Código: HT8.1	Nombre Historia de Usuario: Capacitación a usuarios
Nombre de la Prueba: Firma de asistencia de los participantes en la capacitación del OCW-ESPOCH	
Responsable: Team Development	Fecha: 11/01/2016
Descripción: Se desea tener una evidencia física de la capacitación a usuarios	
Condiciones de Ejecución:	
<ul style="list-style-type: none"> - Despliegue de la aplicación - Usuarios a capacitarse presentes en el sistema OCW-ESPOCH 	
Pasos de Ejecución:	
<ul style="list-style-type: none"> - Abrir la url del sistema OCW-ESPOCH - Autenticarse como usuario - Probar y explicar el sistema a usuarios 	
Resultado Esperado:	
<ul style="list-style-type: none"> - Capacitación del sistema a usuarios completa y recolección de mejoras a la interfaz gráfica de usuario 	
Evaluación de la Prueba:	
Satisfactoria.	

Realizado por: Carlos Mejía, Jhonny Guevara, 2016

SPRINT 21

Tabla 2-228: HT9 Mejoras al Sistema

HISTORIA DE USUARIO	
Número: HT9	Sprint Asignado: 21
Fecha Inicio: 12/01/16	Fecha Fin: 12/01/16
Tiempo Estimado:	1 día
Responsable:	Team Development
Descripción: Como administrador solicito se mejoren los colores de la interface para acoplarse a la identidad institucional	
Pruebas de Aceptación:	
<ul style="list-style-type: none"> - Revisión en conjunto con el Ing Jorge Huilca 	

Realizado por: Carlos Mejía, Jhonny Guevara, 2016

Tabla 2-229: Actividades de HT9

Fecha	Actividad	Puntos estimados
12/01/16 al 12/01/16	Tarea 1: Acoplar los colores del sistema a la identidad institucional - Establecer reuniones de coordinación con el Ing Jorge Huilca. - Revisión de las interfaces del sistema OCW-ESPOCH - Documentación	8

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-230: Tarea 1 de la HT9

TAREA DE INGENIERIA	
Sprint: 21	Número de Tarea: 1
Nombre Historia de Usuario: Mejoras al sistema	
Nombre de la Tarea: Acoplar los colores del sistema a la identidad institucional	
Tipo de Tarea: Desarrollo	Programador Responsable: Team Development
Fecha Inicio: 12/01/16	Fecha Fin: 12/01/16
Descripción: Se desea acoplar los colores del sistema a la identidad institucional	
Pruebas de Aceptación:	
- Revisión y aceptación del sistema por parte del ingeniero Jorge Huilca	

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-231: Prueba de aceptación 1 de HT9

PRUEBA DE ACEPTACIÓN	
Código: HT9.1	Nombre Historia de Usuario: Mejoras al sistema
Nombre de la Prueba: Revisión y aceptación del sistema por parte del ingeniero Jorge Huilca	
Responsable: Team Development	Fecha: 19/01/2016
Descripción: Se desea acoplar la identidad corporativa a los colores de la facultad	
Condiciones de Ejecución:	
- Identificar los elementos a cambiar. - Implementar los cambios	
Pasos de Ejecución:	
- Revisión de las interfaces con el Ing. Jorge Huilca	
Resultado Esperado:	
- Aprobación de las interfaces por parte del ingeniero Jorge Huilca	
Evaluación de la Prueba:	
Satisfactoria.	

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-232: HT1 10 Documentación

HISTORIA DE USUARIO	
Número: HT10	Sprint Asignado: 21
Fecha Inicio: 13/01/16	Fecha Fin: 19/01/16
Tiempo Estimado:	5 días
Responsable:	Team Development
Descripción: Como administrador solicito se elabore la documentación (manual de usuario y técnico del OCW-ESPOCH)	
Pruebas de Aceptación: Revisión en conjunto con el Ing Jorge Huilca	

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-233: Actividades de HT10

Fecha	Actividad	Puntos estimados
13/01/16 al 19/01/16	Tarea 1: Elaboración del manual técnico y manual de usuario del OCW-ESPOCH - Establecer reuniones de coordinación con el Ing Jorge Huilca. - Revisión de los manuales del sistema OCW-ESPOCH - Documentación	40

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-234: Tarea 1 de la HT10

TAREA DE INGENIERIA	
Sprint: 21	Número de Tarea: 1
Nombre Historia de Usuario: Documentación	
Nombre de la Tarea: Elaboración del manual técnico y manual de usuario del OCW-ESPOCH	
Tipo de Tarea: Desarrollo	Programador Responsable: Team Development
Fecha Inicio: 13/01/16	Fecha Fin: 19/01/16
Descripción: Se desea elaborar el manual técnico y de usuario del OCW-ESPOCH	
Pruebas de Aceptación: - Revisión y aceptación de la documentación por parte del ingeniero Jorge Huilca	

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 2-235: Prueba de aceptación 1 de HT10

PRUEBA DE ACEPTACIÓN	
Código: HT10.1	Nombre Historia de Usuario: Mejoras al sistema
Nombre de la Prueba: Revisión y aceptación de la documentación por parte del ingeniero Jorge Huilca	

Responsable: Team Development	Fecha: 19/01/2016
Descripción: Se desea obtener el manual técnico y de usuario del OCW-ESPOCH	
Condiciones de Ejecución:	
<ul style="list-style-type: none"> - Identificar las historias de usuario. - Identificar los distintos usuarios del sistema 	
Pasos de Ejecución:	
<ul style="list-style-type: none"> - Elaborar el manual técnico. - Elaborar el manual de usuario - Revisión de los manuales. 	
Resultado Esperado:	
<ul style="list-style-type: none"> - Aprobación de los manuales técnico y de usuario 	
Evaluación de la Prueba:	
Satisfactoria.	

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

2.1.10 GESTIÓN DEL PROYECTO (Burndown Chart)

Un "Burndown Chart" es una representación gráfica del trabajo a realizar en el tiempo (Oliveira, 2012).

Las fechas del Sprint se representan en el eje X en la que se visualiza la fecha de inicio y fin que corresponden a los días comprendidos entre el 02 de octubre del 2015 al 19 de enero del 2016, mientras que el esfuerzo se representa en horas sobre el eje Y con un total de 1008(1008 puntos de función), además el gráfico está representado por dos líneas; la línea de color azul muestra el desarrollo real del proyecto la línea de color rojo plasma el desarrollo ideal.

En la Ilustración 27, se aprecia que la entrega de la funcionalidad fluctúa levemente con respecto a la planificación de sprint en sprint, esto se debe a que en el desarrollo del proyecto existieron retrasos debido a cambios en la requerimentación obtenida; estos cambios fueron replanificados oportunamente y consecuentemente se entregó la funcionalidad en el tiempo acordado inicialmente.

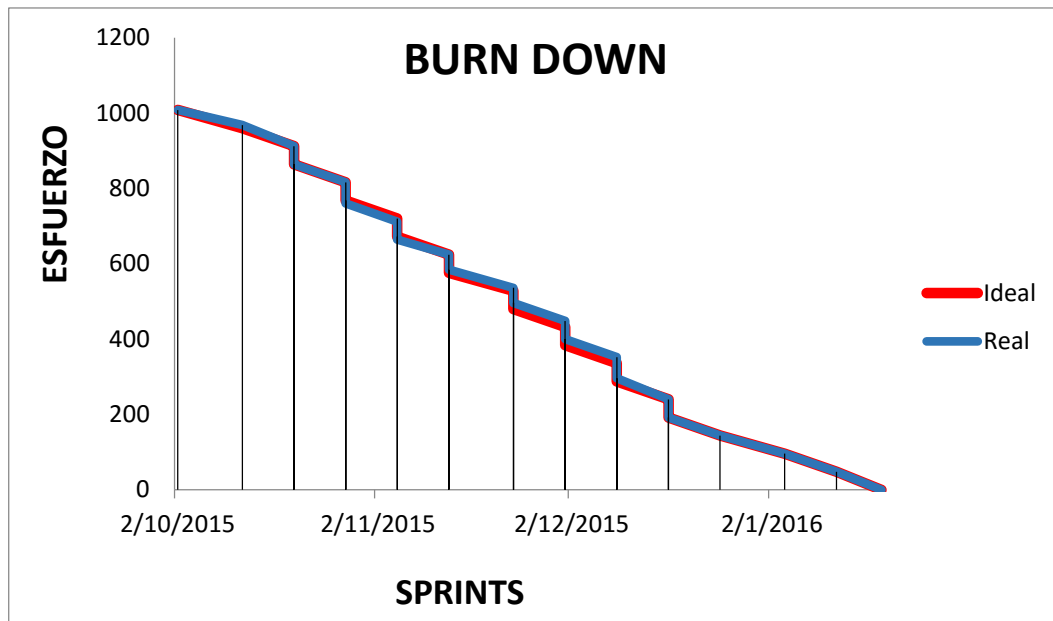


Figura 2-32. Burn Down Chart

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

2.2 INSTALACIÓN Y CONFIGURACIÓN

- Requerimiento de hardware para la máquina virtual VMWare Workstation 9

El servidor Ubuntu 14.04.2 debe tener las siguientes características:

- ✓ Memoria RAM: mínimo 2 GB
- ✓ Disco Duro: 20 GB
- ✓ Procesador: 2 núcleos
- ✓ Adaptador de Red: Bridged (Automático)

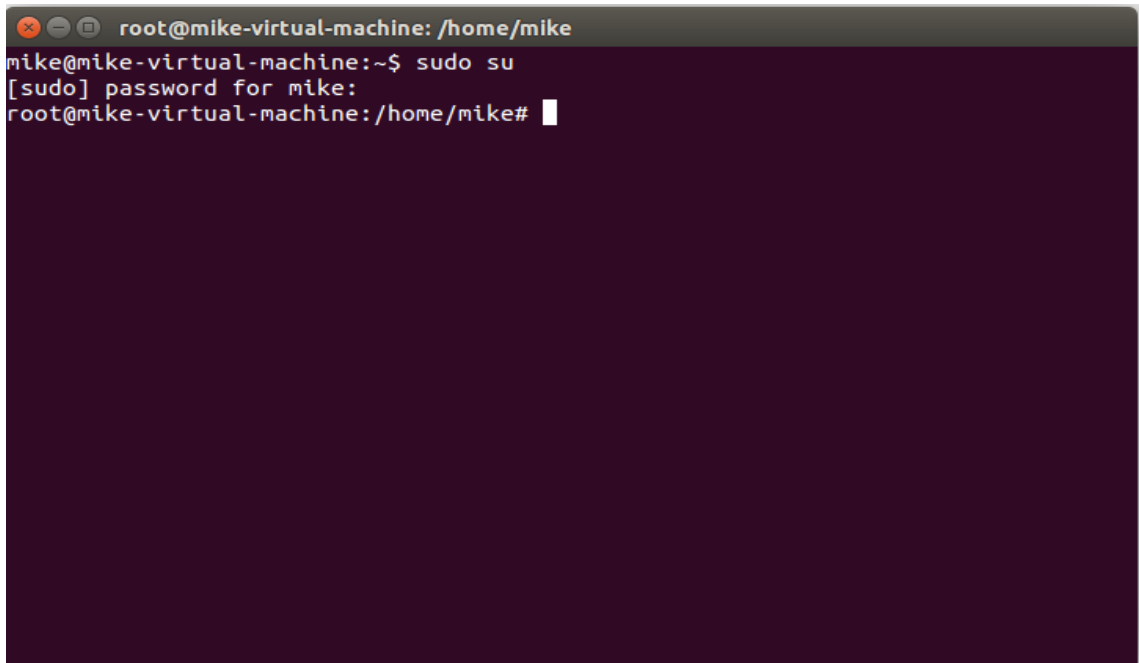
- Requerimiento de software

Los requerimientos de software que el sistema OPEN COURSE WARE (OCW-ESPOCH) son:

- ✓ Instalar JRE 1.7
- ✓ Instalar el servidor GlassFish 4
- ✓ Instalar el motor de base de datos PostgreSQL 9.3
- ✓ Salida permanente a Internet, se recomienda contar con un ancho de banda de 1mb como óptimo.

Pasos para instalar JRE 1.7

Abrir el terminal de Linux como súper usuario

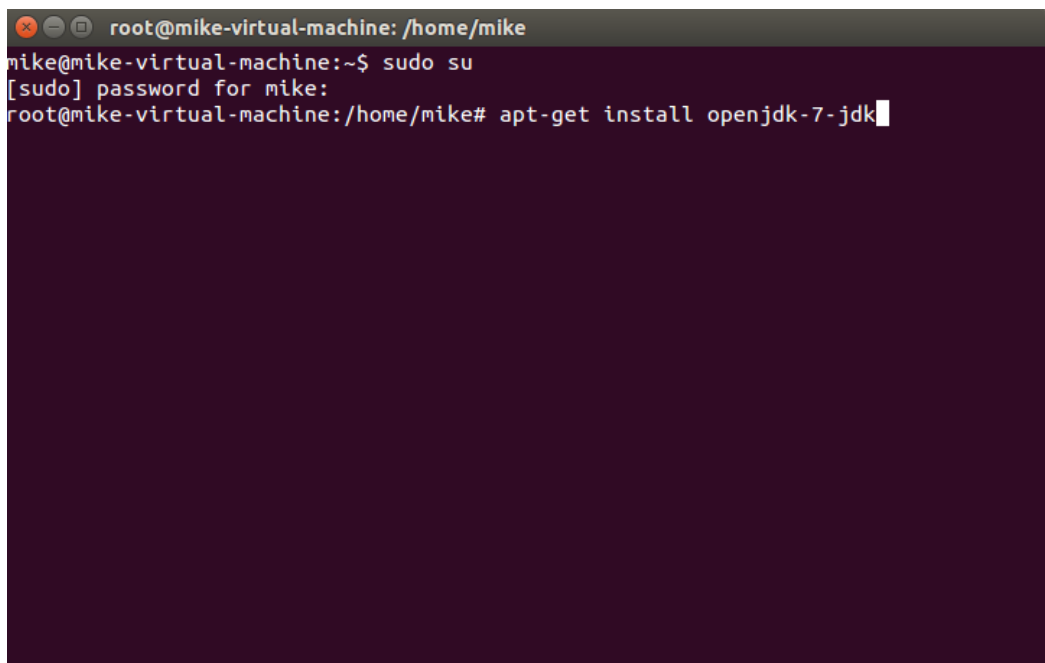


```
root@mike-virtual-machine: /home/mike
mike@mike-virtual-machine:~$ sudo su
[sudo] password for mike:
root@mike-virtual-machine: /home/mike#
```

Figura 2-33. Terminal Linux

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Ejecutar la siguiente línea de comandos y esperar a que se instale java (apt-get install openjdk-7-jdk)



```
root@mike-virtual-machine: /home/mike
mike@mike-virtual-machine:~$ sudo su
[sudo] password for mike:
root@mike-virtual-machine: /home/mike# apt-get install openjdk-7-jdk
```

Figura 2-34. Instalar Java

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

- Progreso de instalación y descarga.

```

root@mike-virtual-machine: /home/mike
2 [19,9 kB]
Des:30 http://pe.archive.ubuntu.com/ubuntu/ trusty/main libxau-dev amd64 1:1.0.8
-1 [11,1 kB]
Des:31 http://pe.archive.ubuntu.com/ubuntu/ trusty/main libxdmcp-dev amd64 1:1.1
.1-1 [26,9 kB]
Des:32 http://pe.archive.ubuntu.com/ubuntu/ trusty/main x11proto-input-dev all 2
.3-1 [139 kB]
Des:33 http://pe.archive.ubuntu.com/ubuntu/ trusty/main x11proto-kb-dev all 1.0.
6-2 [269 kB]
Des:34 http://pe.archive.ubuntu.com/ubuntu/ trusty-updates/main xtrans-dev all 1
.3.5-1~ubuntu14.04.1 [70,3 kB]
Des:35 http://pe.archive.ubuntu.com/ubuntu/ trusty/main libxcb1-dev amd64 1.10-2
ubuntu1 [76,6 kB]
Des:36 http://pe.archive.ubuntu.com/ubuntu/ trusty/main libx11-dev amd64 2:1.6.2
-1ubuntu2 [629 kB]
Des:37 http://pe.archive.ubuntu.com/ubuntu/ trusty/main libx11-doc all 2:1.6.2-1
ubuntu2 [1.448 kB]
Des:38 http://pe.archive.ubuntu.com/ubuntu/ trusty/main libxt-dev amd64 1:1.1.4-
1 [455 kB]
Des:39 http://pe.archive.ubuntu.com/ubuntu/ trusty/main lksctp-tools amd64 1.0.1
5+dfsg-1 [51,3 kB]
Des:40 http://pe.archive.ubuntu.com/ubuntu/ trusty-updates/main openjdk-7-jdk ar
d64 7u95-2.6.4-0ubuntu0.14.04.1 [16,3 MB]
77% [40 openjdk-7-jdk 1.846 kB/16,3 MB 11%] 241 kB/s 59seg.

```

Figura 2-35. Progreso Instalación Java

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

```

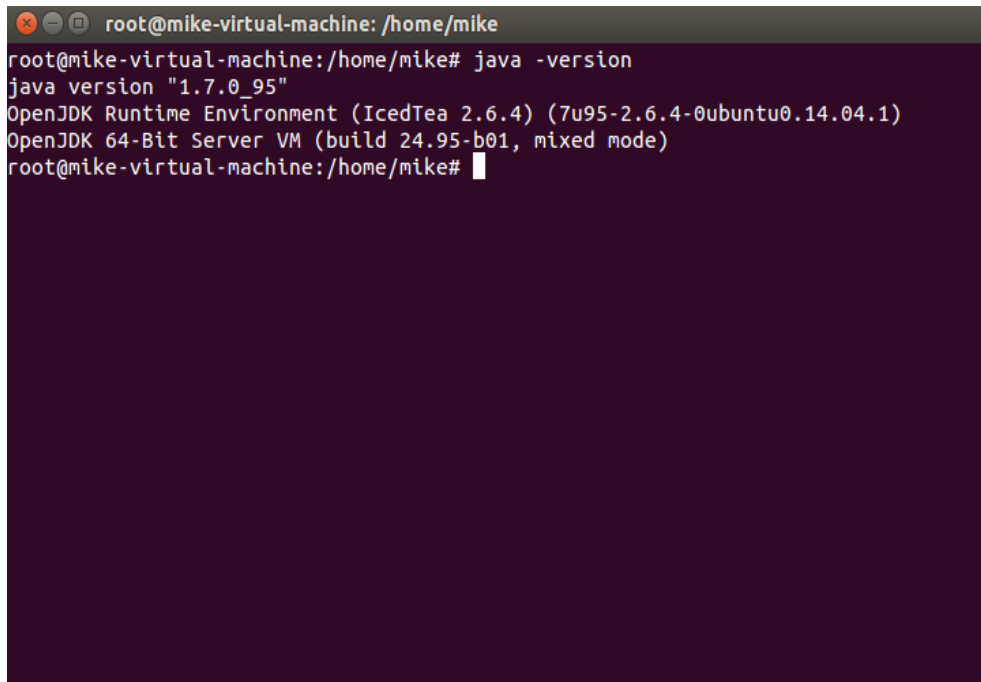
root@mike-virtual-machine: /home/mike
proveer /usr/bin/jstat (jstat) en modo automático
update-alternatives: utilizando /usr/lib/jvm/java-7-openjdk-amd64/bin/jstatd par
a proveer /usr/bin/jstatd (jstatd) en modo automático
update-alternatives: utilizando /usr/lib/jvm/java-7-openjdk-amd64/bin/native2asci
i para proveer /usr/bin/native2ascii (native2ascii) en modo automático
update-alternatives: utilizando /usr/lib/jvm/java-7-openjdk-amd64/bin/rmic para
proveer /usr/bin/rmic (rmic) en modo automático
update-alternatives: utilizando /usr/lib/jvm/java-7-openjdk-amd64/bin/schemagen
para proveer /usr/bin/schemagen (schemagen) en modo automático
update-alternatives: utilizando /usr/lib/jvm/java-7-openjdk-amd64/bin/serialver
para proveer /usr/bin/serialver (serialver) en modo automático
update-alternatives: utilizando /usr/lib/jvm/java-7-openjdk-amd64/bin/wsgen para
proveer /usr/bin/wsgen (wsgen) en modo automático
update-alternatives: utilizando /usr/lib/jvm/java-7-openjdk-amd64/bin/wsimport p
ara proveer /usr/bin/wsimport (wsimport) en modo automático
update-alternatives: utilizando /usr/lib/jvm/java-7-openjdk-amd64/bin/xjc para p
roveer /usr/bin/xjc (xjc) en modo automático
Processing triggers for libc-bin (2.19-0ubuntu6.5) ...
Processing triggers for ca-certificates (20130906ubuntu2) ...
Updating certificates in /etc/ssl/certs... 0 added, 0 removed; done.
Running hooks in /etc/ca-certificates/update.d....
done.
done.
root@mike-virtual-machine: /home/mike#

```

Figura 2-36. Progreso Instalación Java 2

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

- Verificamos la version de Java instalada con el comando java-version



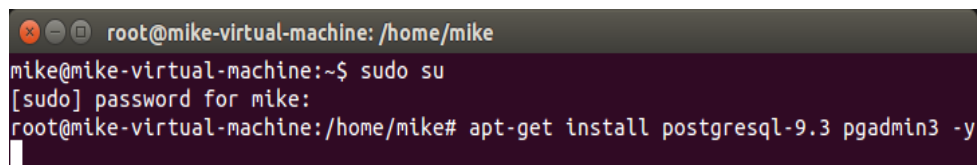
```
root@mike-virtual-machine: /home/mike
root@mike-virtual-machine:/home/mike# java -version
java version "1.7.0_95"
OpenJDK Runtime Environment (IcedTea 2.6.4) (7u95-2.6.4-0ubuntu0.14.04.1)
OpenJDK 64-Bit Server VM (build 24.95-b01, mixed mode)
root@mike-virtual-machine:/home/mike#
```

Figura 2-37. Verificar Instalación Java

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Pasos para instalar PostgreSQL 9.3

Ejecutar como súper usuario en un terminal el comando mostrado en pantalla y esperar que termine la instalación:

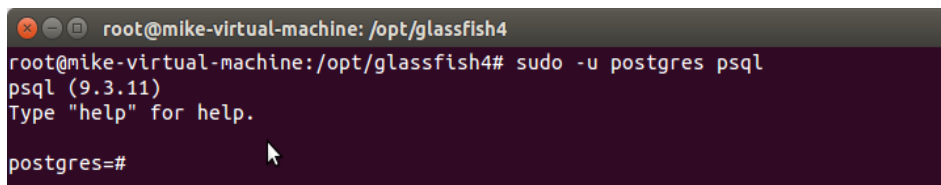


```
root@mike-virtual-machine: /home/mike
mike@mike-virtual-machine:~$ sudo su
[sudo] password for mike:
root@mike-virtual-machine:/home/mike# apt-get install postgresql-9.3 pgadmin3 -y
```

Figura 2-38. Instalar Postgresql

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Cambiamos la contraseña al servidor postgres con el siguiente comando



```
root@mike-virtual-machine:/opt/glassfish4
root@mike-virtual-machine:/opt/glassfish4# sudo -u postgres psql
psql (9.3.11)
Type "help" for help.

postgres=#
```

Figura 2-39. Cambio de contraseña

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Ingresamos la nueva contraseña

```
root@mike-virtual-machine: /opt/glassfish4
root@mike-virtual-machine:/opt/glassfish4# sudo -u postgres psql postgres
psql (9.3.11)
Type "help" for help.

postgres=# \password postgres
Enter new password:
Enter it again:
postgres=#
```

Figura 2-40. Cambio de contraseña

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Buscamos el administrador pgadmin III:

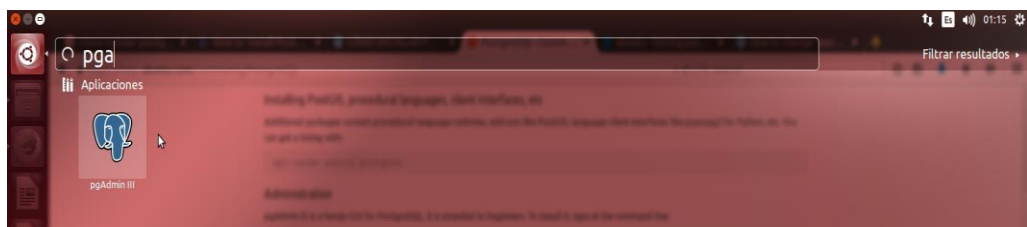


Figura 2-41. Búsqueda de Pgadmin

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Abrimos el administrador y hacemos clic en crear una nueva conexión

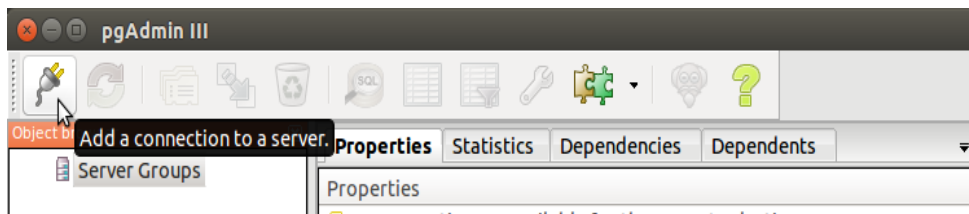


Figura 2-42. Nueva Conexión

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Llenamos los datos del formulario incluyendo la contraseña de nuestro usuario postgres y damos clic en aceptar.

The image shows a 'New Server Registration' dialog box with the following fields and values:

Field	Value
Name	mike
Host	localhost
Port	5432
Service	
Maintenance DB	postgres
Username	postgres
Password
Store password	<input checked="" type="checkbox"/>
Colour	
Group	Servers

Buttons at the bottom: Ayuda, Aceptar, Cancelar.

Figura 2-43. Configuración de Conexión

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Pasos para restaurar la base de datos

Expandimos la nueva conexión mike, navegamos hasta la sección database y damos clic derecho crear nueva base de datos:

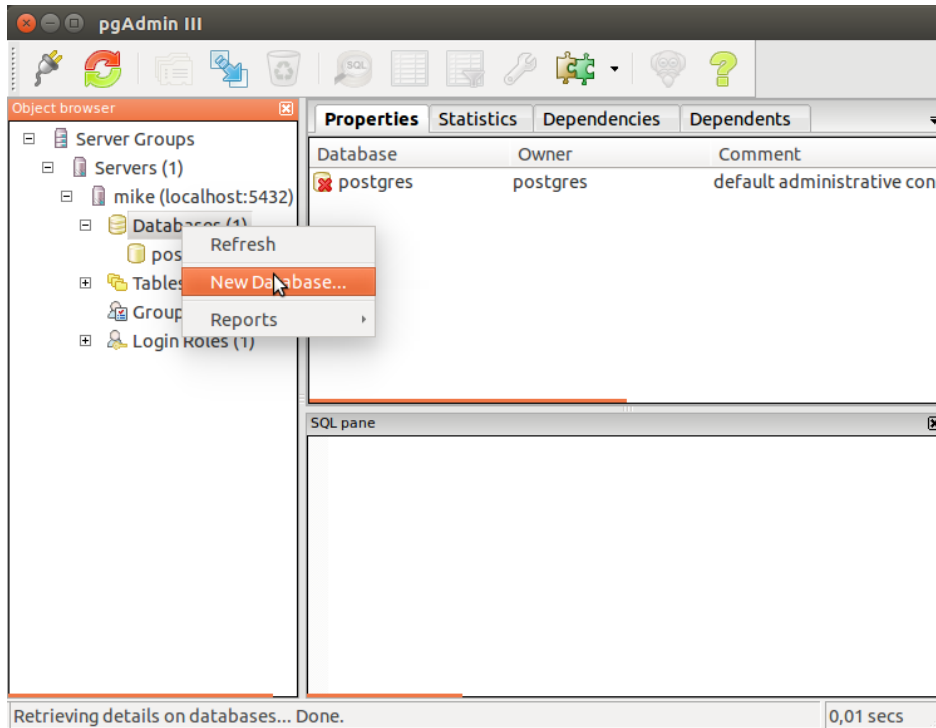


Figura 2-44. Crear Base de Datos

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Clic derecho en la base OpenCourseWare y seleccionamos la opción restaurar

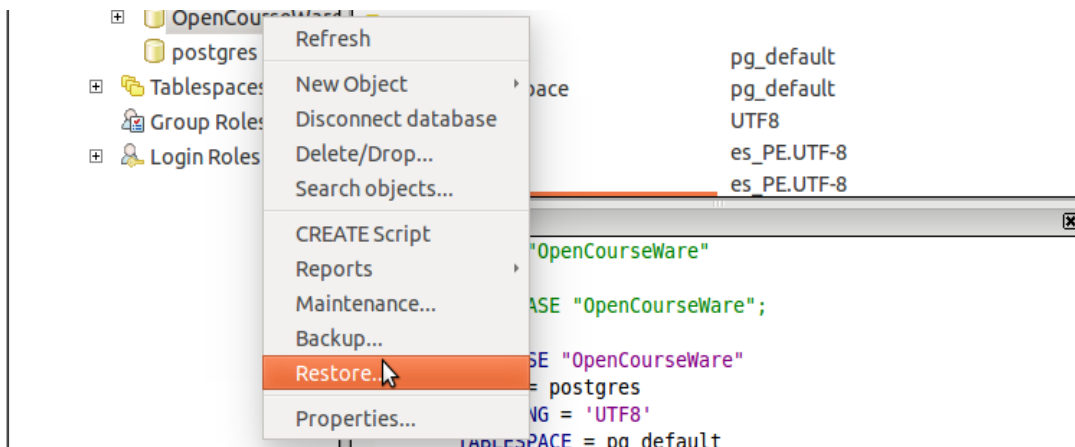


Figura 2-45. Restauración de la base de datos

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Obtendrá un mensaje confirmación para la restauración; clic en done:

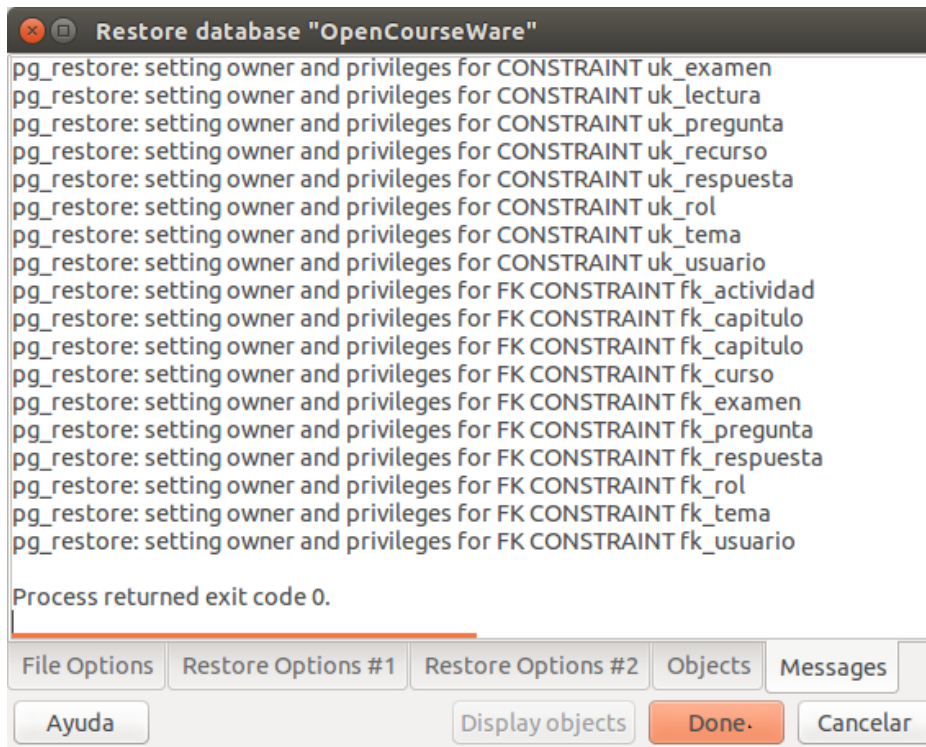


Figura 2-46. Restauración de la base de datos

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Pasos para instalar GlassFish 4

- Ejecutar este comando `wget download.java.net/glassfish/4.0/release/glassfish-4.0.zip` para descargar el instalador de glassfish

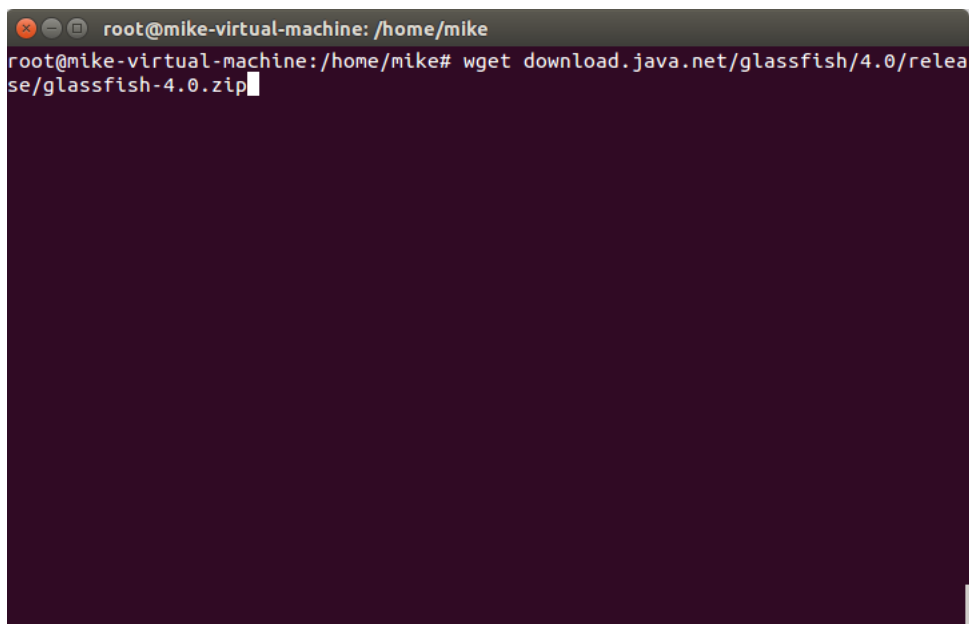


Figura 2-47. Restauración de la base de datos

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

- Creamos un usuario que ejecute el glassfish con el comando `useradd --system glassfish -d /opt/glassfish4/`

```
root@mike-virtual-machine:/opt# useradd --system glassfish -d /opt/glassfish4/
root@mike-virtual-machine:/opt#
```

Figura 2-48. Asignación de usuario para Glassfish

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

- Cambiamos el dueño de los archivos del directorio glassfish y asignamos permisos de ejecución con los comandos mostrados en pantalla

```
root@mike-virtual-machine:/opt# chown -R glassfish glassfish4
root@mike-virtual-machine:/opt# chgrp -R sudo glassfish4
root@mike-virtual-machine:/opt# chmod -R +x glassfish4/bin/
root@mike-virtual-machine:/opt# chmod -R +x glassfish4/glassfish/bin
```

Figura 2-49. Permisos de Usuario

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

- Cambiamos el password del servidor glassfish con el siguiente comando

```
root@mike-virtual-machine:/opt/glassfish4# sudo -u glassfish bin/asadmin change-
admin-password
Enter admin user name [default: admin]>
Enter the admin password>
Enter the new admin password>
Enter the new admin password again>
Command change-admin-password executed successfully.
root@mike-virtual-machine:/opt/glassfish4#
```

Figura 2-50. Credenciales de Glassfish

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Pasos para instalar Netbeans 7.4

- Descargar Netbeans <https://netbeans.org/downloads/7.4/index.html> y descargue NetBeans 7.4

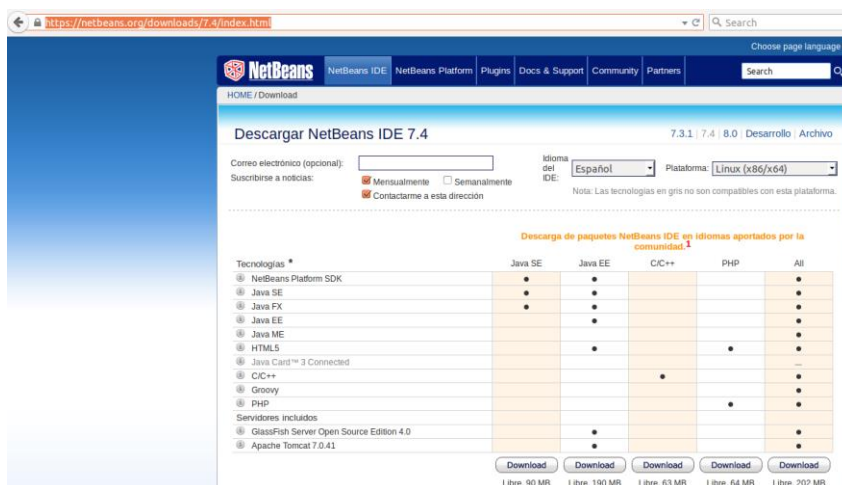


Figura 2-50. Descargar NetBeans

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

- Ejecute el siguiente comando para comenzar la instalación de NetBeans

```
root@mike-virtual-machine: /opt/glassfish4
root@mike-virtual-machine: /opt/glassfish4# '/home/mike/Descargas/netbeans-7.4-linux.sh'
bash: /home/mike/Descargas/netbeans-7.4-linux.sh: Permiso denegado
root@mike-virtual-machine: /opt/glassfish4# bash '/home/mike/Descargas/netbeans-7.4-linux.sh'
Configurando el instalador...
Buscando la JVM en el sistema...
Extrayendo los datos de la instalación...
Ejecutando el asistente de instalación...
█
```

Figura 2-51. Instalación NetBeans

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

- Vamos a personalizar los paquetes a instalarse con Netbeans

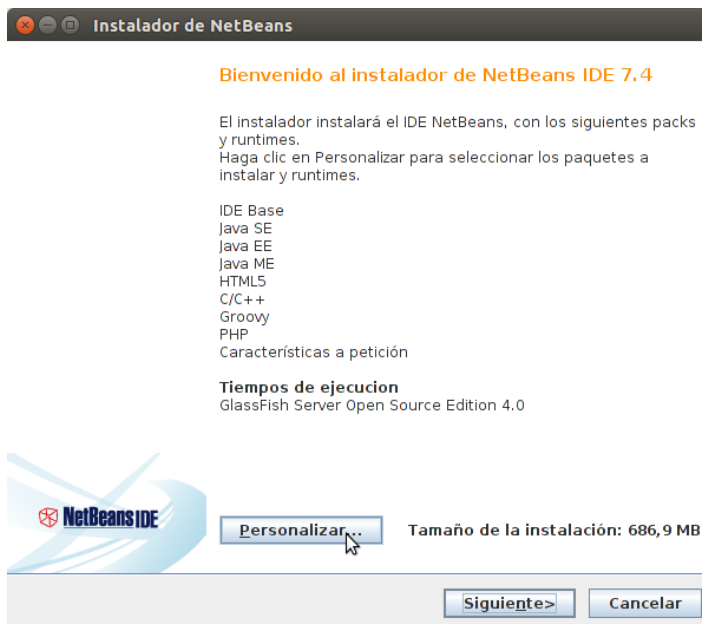


Figura 2-52. Paquetes de Netbeans

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

1. Acepte el contrato de licencia y clic en siguiente

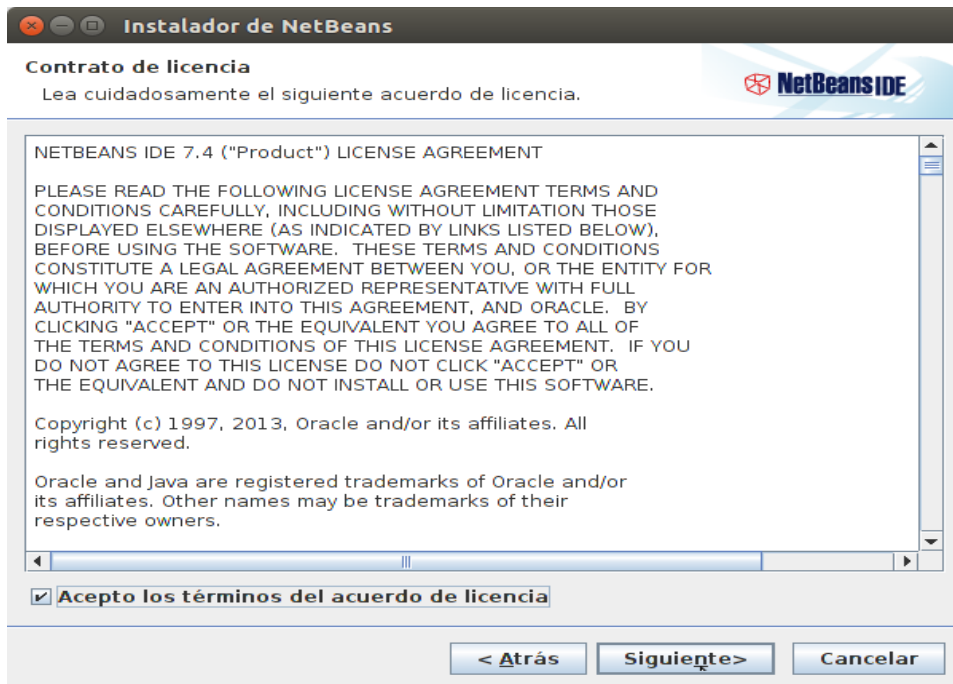


Figura 2-53. Términos de licencia

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

- Clic en instalar y esperar que concluya la instalación

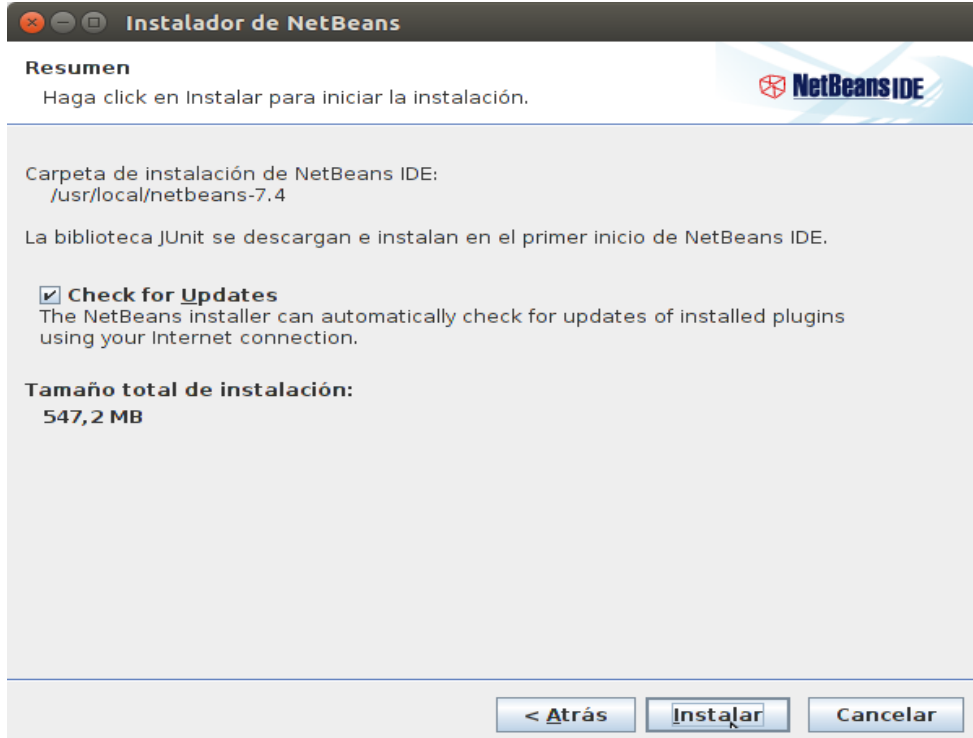


Figura 2-54. Instalar

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Pasos para desplegar la aplicación

- Copiar la carpeta prepara al escritorio de su servidor

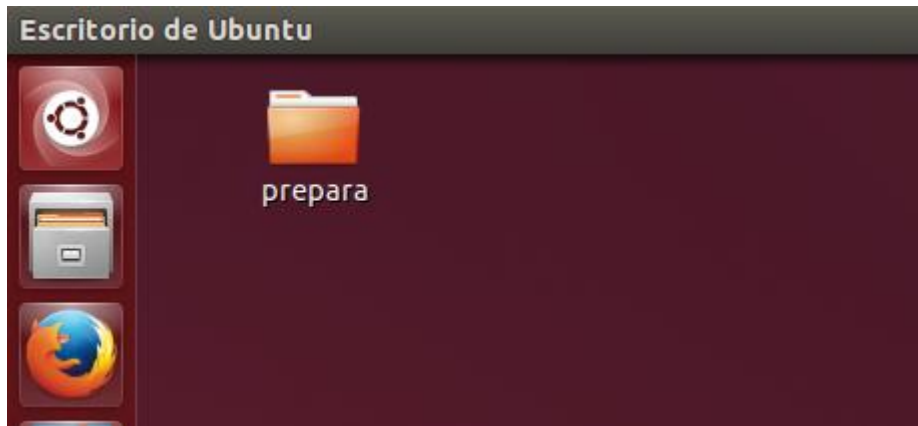


Figura 2-55. Preparar entorno

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

- Copiar la carpeta recursos de manera recursiva hacia es directorio raíz con el siguiente comando

```
root@mike-virtual-machine: /home/mike
root@mike-virtual-machine:/home/mike# cp -R '/home/mike/Escritorio/prepara/recursos' /
root@mike-virtual-machine:/home/mike#
```

Figura 2-56. Copiar Directorio a la raíz

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

- Abrir el Netbeans 7.4 previamente instalado y seleccione abrir proyecto

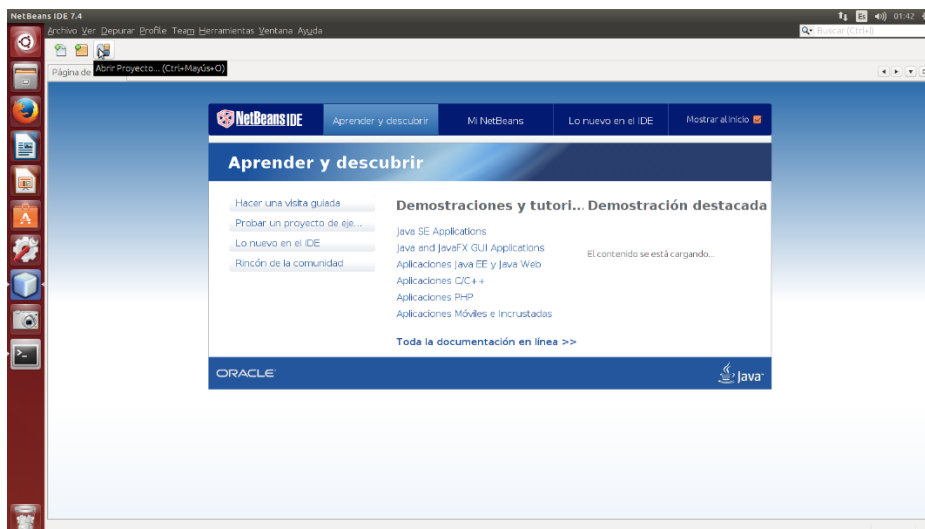


Figura 2-57. Abrir proyecto desde NetBeans

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

- Acceder a las propiedades del proyecto OpenCourseWare

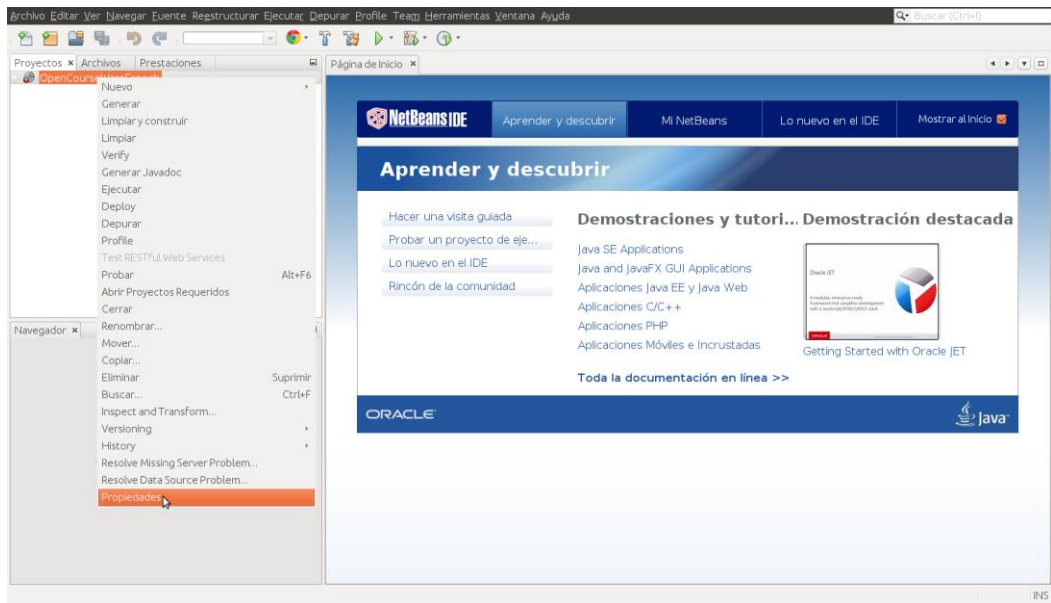


Figura 2-58. Abrir proyecto e ir a sus propiedades

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Ir a propiedades y del proyecto y remover las librerías que se muestran en la imagen

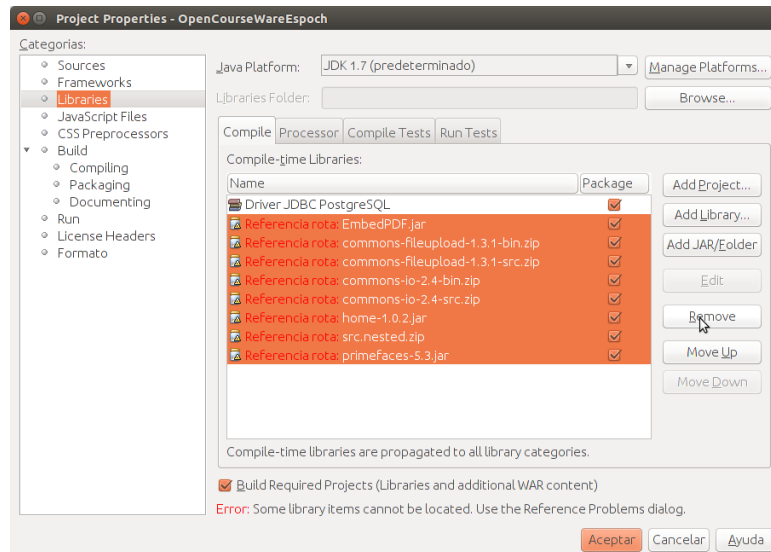


Figura 2-59. Remover librerías

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

- Clic en añadir Jar/folder

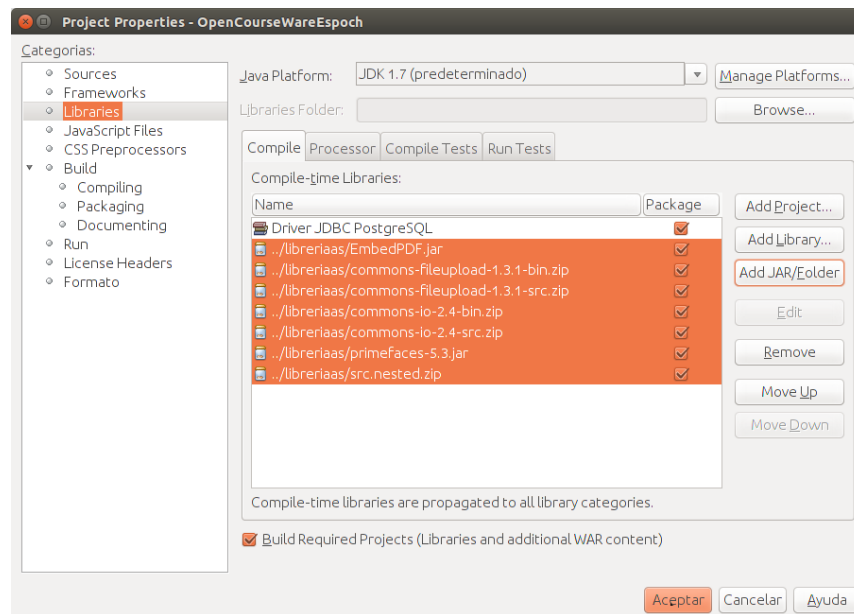


Figura 2-60. Añadir librerías

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

- Navegar hasta el directorio ../libreriaas y añadir las siguientes marcando la opción ruta relativa y clic en aceptar.

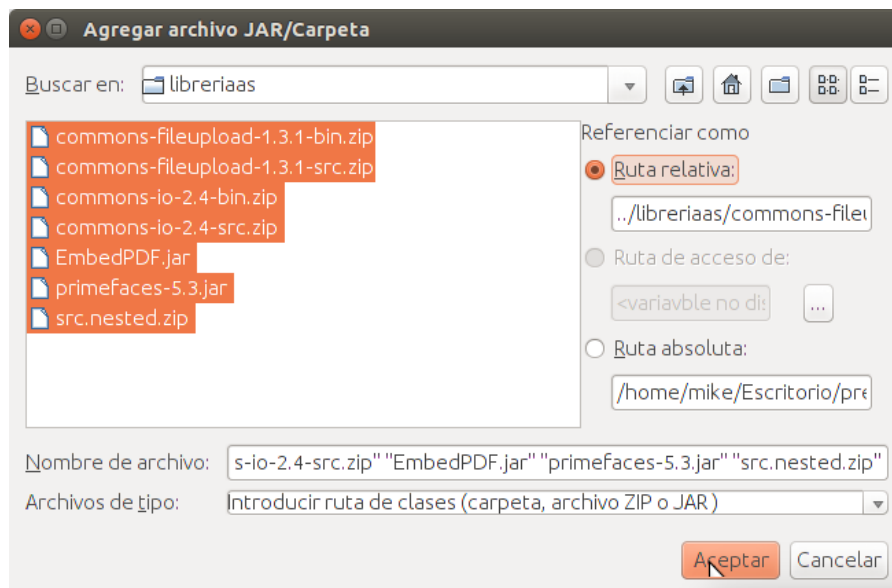


Figura 2-61. Seleccionar y configurar librerías

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

1. Clic derecho sobre el proyecto y seleccionar resolver problemas de servidor perdido

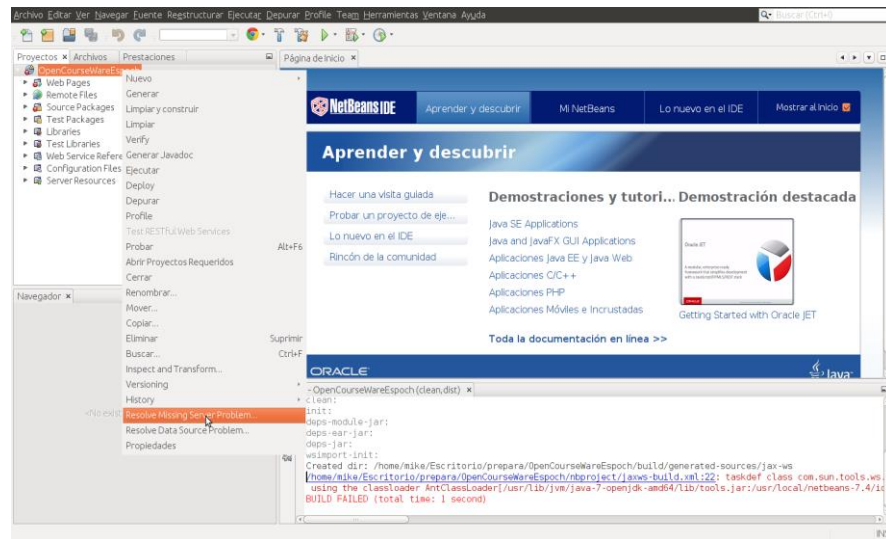


Figura 2-62. Configuración servidor

Realizado por: Carlos Mejía, Jhonny Guevara, 2016

2. Clic en la opción añadir servidor

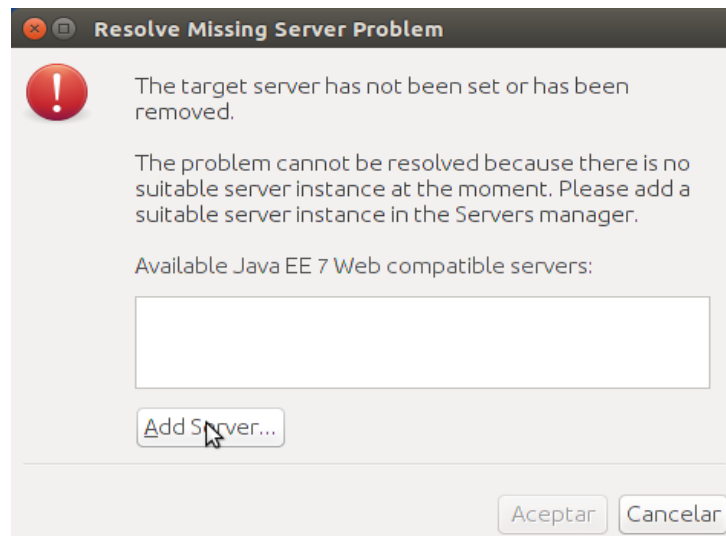


Figura 2-63. Añadir Servidor

Realizado por: Carlos Mejía, Jhonny Guevara, 2016

- Seleccionar el servidor glassfish y clic en siguiente.

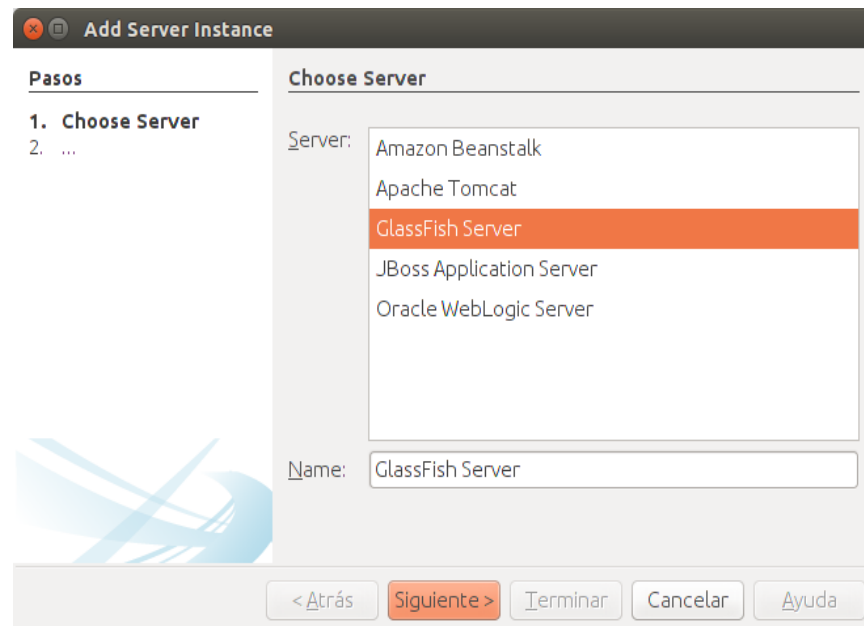


Figura 2-64. Seleccionar Servidor

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

- Ir hacia la ruta /opt/glassfish4 y hacer clic en escoger

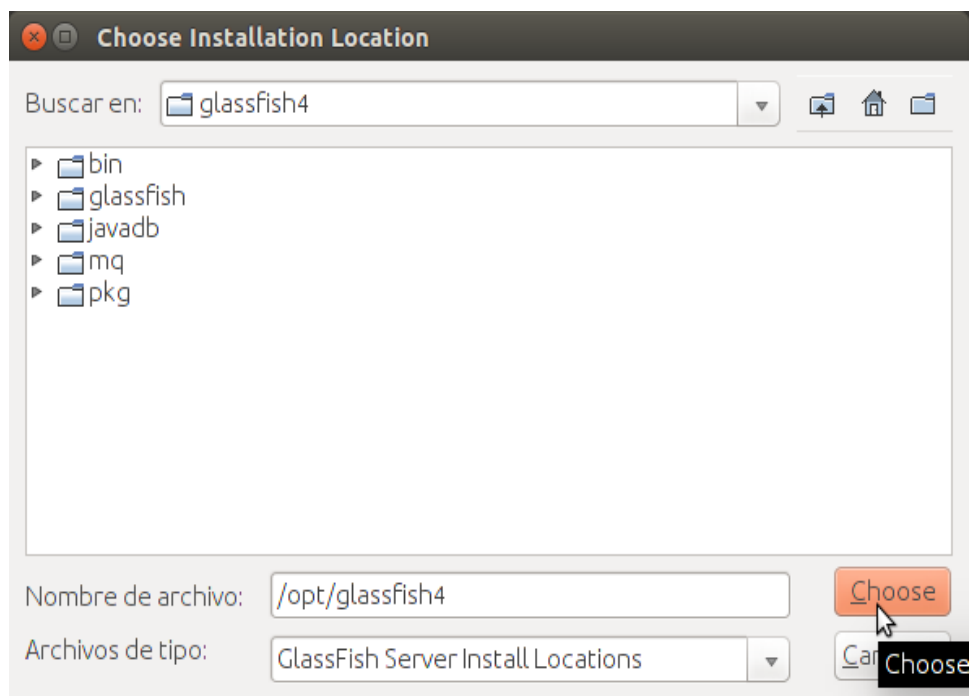


Figura 2-65. Seleccionar Ruta

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

- Clic en siguiente

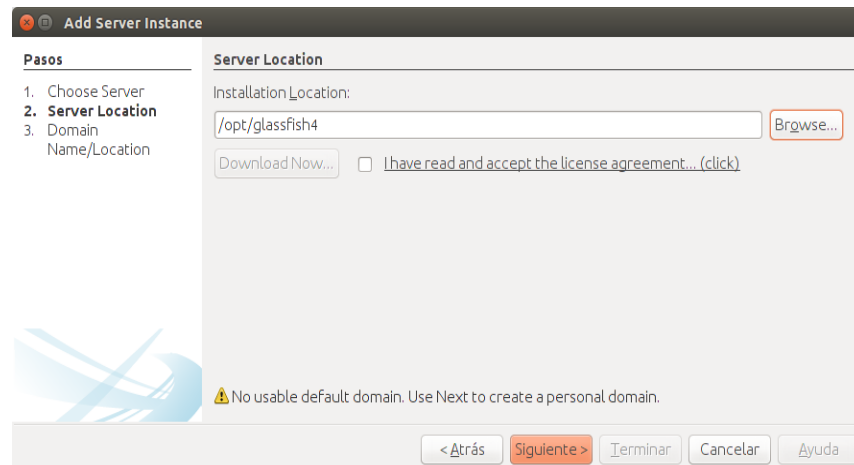


Figura 2-66. Siguiete paso

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

- Escribir usuario y contraseña para el servidor glassfish y dar clic en terminar

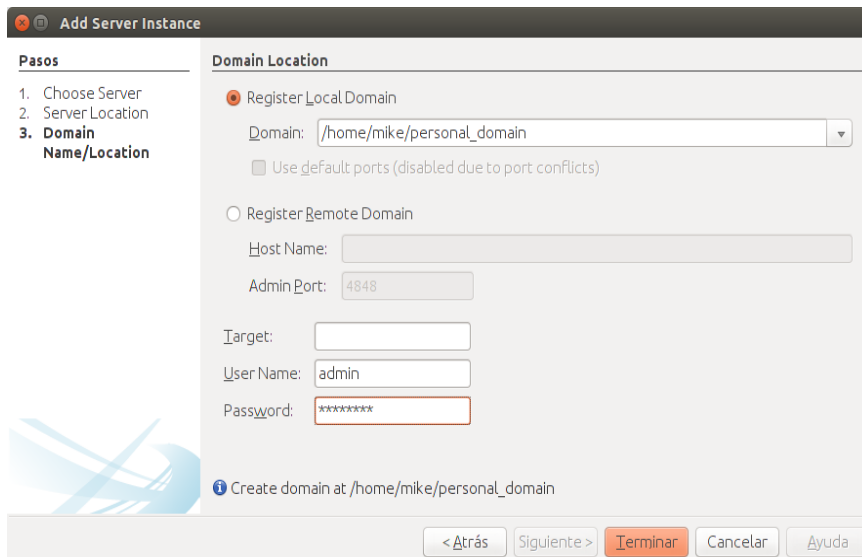


Figura 2-67. Ingresar Contraseña y finalizar

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Clic en aceptar

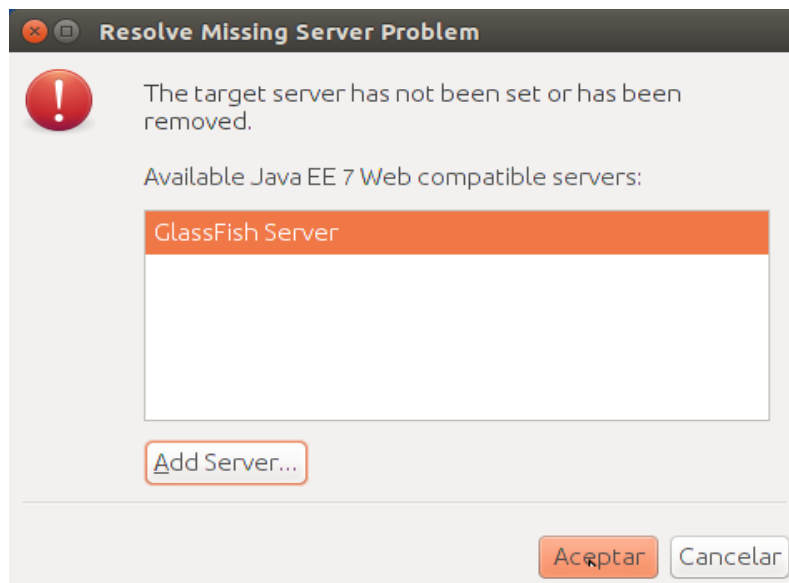


Figura 2-68. Seleccionar y Aceptar

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Clic derecho en el proyecto y seleccionamos resolve data source problem

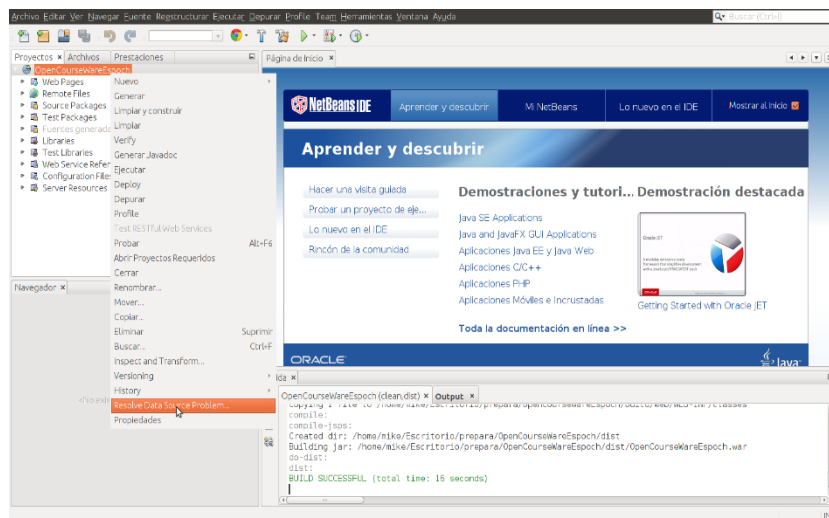


Figura 2-69. Resolver problemas de servidor

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Clic en add connection

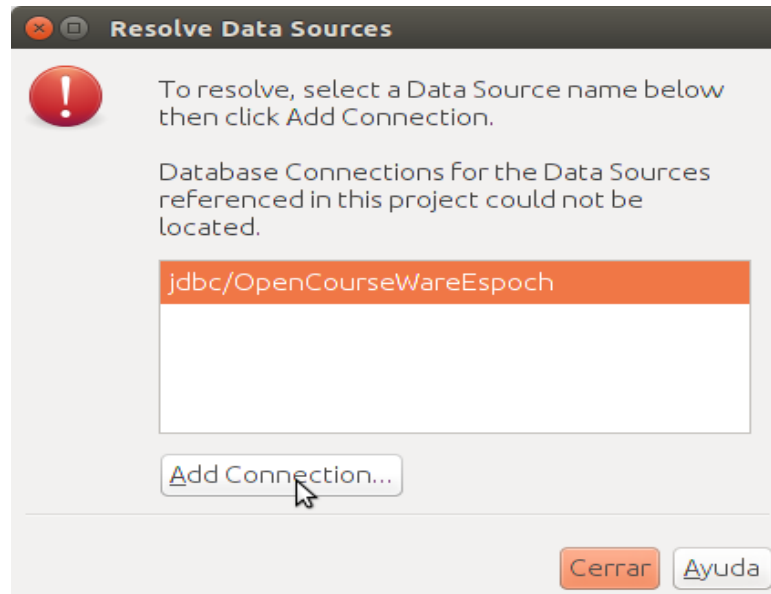


Figura 2-70. Agregar conexión

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Clic en terminar



Figura 2-71. Seleccionar y Aceptar

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Clic Derecho sobre el proyecto y seleccionamos la opción ejecutar

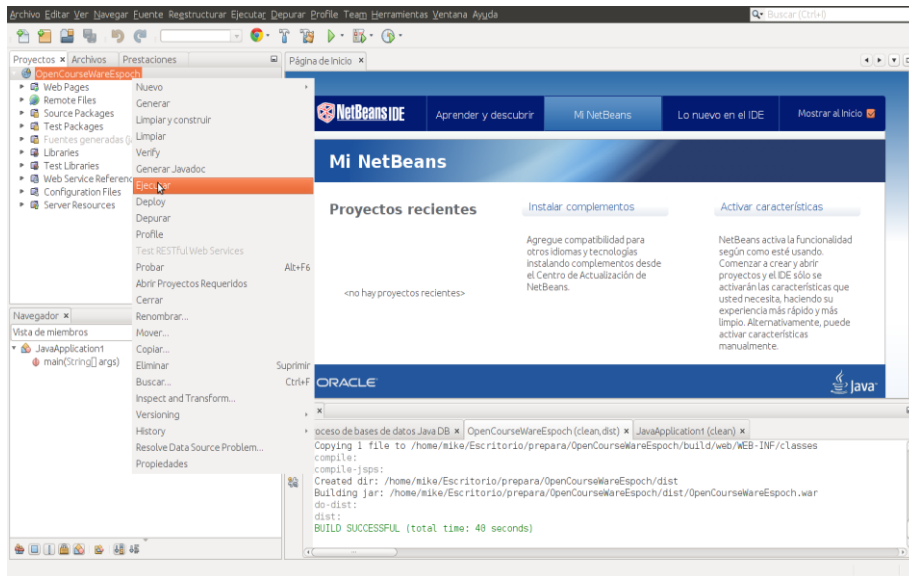


Figura 2-72. Ejecutar la aplicación.

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

El OpenCourseWare Espoch será desplegado sobre el servidor glassfish que lo hemos sincronizado al Netbeans 7.4; esto implica que cualquier despliegue del Netbeans será automáticamente subido al servidor real de glassfish 4.0; a continuación, se mostrará



Figura 2-73. Vista principal

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

CAPÍTULO III

3 MARCO DE RESULTADOS, DISCUSION Y ANALISIS DE RESULTADO

Actualmente la Escuela Superior Politécnica de Chimborazo cuenta con 655 docentes impartiendo materias que se les son asignadas en cada periodo académico, de la misma manera se le son asignadas aulas virtuales en el EVA (Entorno Virtual de Aprendizaje) de la ESPOCH también conocido como Elearning.

Según estadísticas del DTIC de la ESPOCH de las 655 aulas virtuales destinadas para los docentes solo 360 aulas virtuales han sido utilizadas en el pasado periodo Marzo-agosto 2015 y 393 en el actual Periodo octubre 2015-febrero 2016, en la siguiente tabla se detallan los resultados.

Tabla 3.1: Estadísticas Utilización de aulas Virtuales Marzo-agosto 2015

Período	#Docentes	Aulas Virtuales Asignadas	Aulas Virtuales Utilizadas	%
Marzo-agosto 2015	655	655	360	54.9%

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

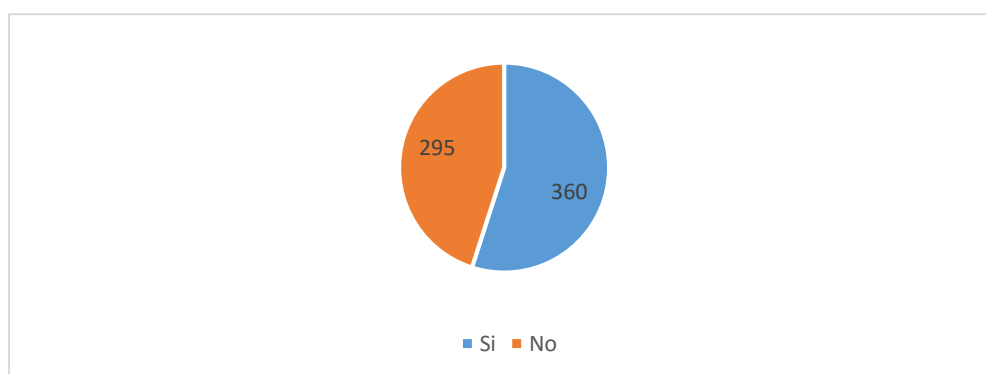


Figura 3-1: Utilización de aulas Virtuales Marzo-agosto 2015

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 3.2: Estadísticas Utilización de aulas Virtuales Octubre-febrero 2015

Período	#Docentes	Aulas Virtuales Asignadas	Aulas Virtuales Utilizadas	%
Octubre -febrero 2016	655	655	393	60%

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

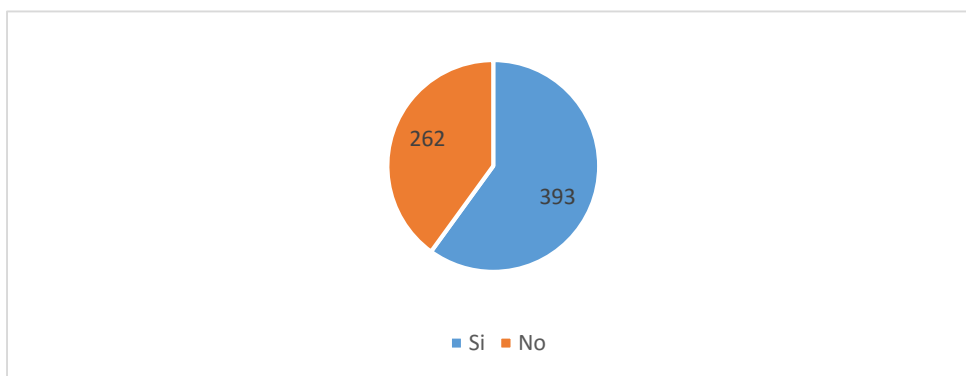


Figura 3-2: Utilización de aulas Virtuales Octubre-febrero 2015

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Como podemos apreciar no todos los docentes utilizan las aulas virtuales, teniendo un promedio de 377 usuarios activos en los 2 últimos periodos, de esto podemos tomar una referencia en cuanto al número de usuarios que posiblemente harían uso del sistema OCW ESPOCH ya que se parte de la premisa de un aproximado de 655 docentes al servicio de la ESPOCH.

3.2 Validación de Usabilidad

Con el fin de asegurar cumplir los objetivos planteados que tiene el sistema Open Course Ware para con la institución se ha puesto a consideración de un grupo de 10 docentes el producto desarrollado mediante una sesión en la que se dio a conocer a detalle cada uno de los módulos y funcionalidades que posee, dicha capacitación duro aproximadamente 1 hora, tiempo en el cual se atendió las diferentes inquietudes y sugerencias que tuvieron los usuarios acerca de los procesos que implica el manejo del sistema, posteriormente se puso a consideración un test de usabilidad en el cual se evaluó aspectos tales como identidad, contenido, navegación, utilidad y retroalimentación.

Tabla 3-3: Preguntas Test Usabilidad-Identidad

Identidad	Si	No	%
1.- ¿Con la información que se ofrece en la página principal, es posible saber a qué institución corresponde el sitio?	10	0	100
2.- ¿Hay algún elemento gráfico o de texto que le haya ayudado a entender más claramente a qué institución pertenece el sitio?	9	1	90
3.- ¿Relaciona los colores predominantes en el sitio web con la institución?	8	2	80
4.- ¿De los elementos que muestra esta pantalla, hay algo que usted crea que está fuera de lugar?	0	10	100
5.- ¿Distingue alguna imagen que represente (logotipo) a la institución y cree que aparece en un lugar importante dentro de la página?	10	0	100

6.- ¿Cree usted que este sitio está dirigido hacia Graduados de la Institución, autodidactas y estudiantes de otras universidades?	9	1	90
TOTAL			93.3

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 3-4: Usabilidad Identidad-Pregunta 1

Identidad	Si	No	%
1.- ¿Con la información que se ofrece en la página principal, es posible saber a qué institución corresponde el sitio?	10	0	100

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

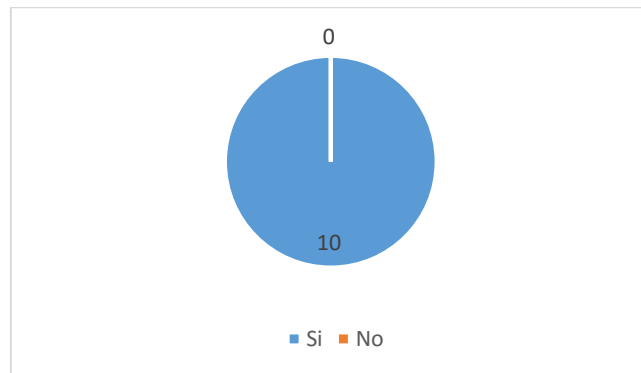


Figura 3-3: Pregunta 1 Usabilidad Identidad

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 3-5: Usabilidad Identidad-Pregunta 2

Identidad	Si	No	%
2.- ¿Hay algún elemento gráfico o de texto que le haya ayudado a entender más claramente a qué institución pertenece el sitio?	9	1	90

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

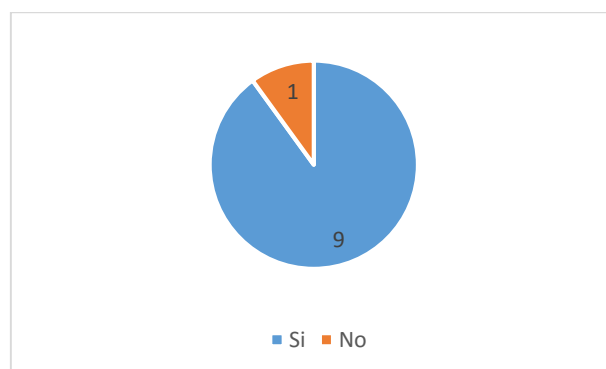


Figura 3-4: Pregunta 2 Usabilidad-Identidad

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 3-6: Usabilidad Identidad-Pregunta 3

Identidad	Si	No	%
3.- ¿Relaciona los colores predominantes en el sitio web con la institución?	8	2	80

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

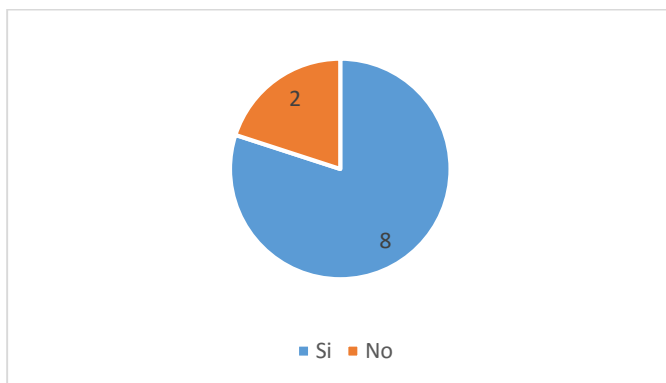


Figura 3-5: Pregunta 3 Usabilidad-Identidad

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 3-7: Usabilidad Identidad-Pregunta 4

Identidad	Si	No	%
4.- ¿De los elementos que muestra esta pantalla, hay algo que usted crea que está fuera de lugar?	0	10	100

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

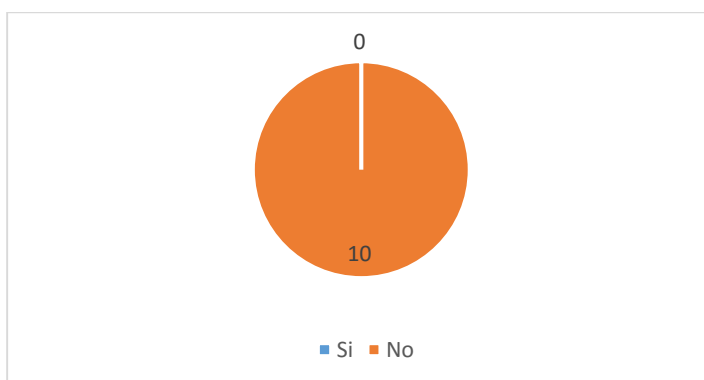


Figura 3-6: Pregunta 4 Usabilidad-Identidad

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 3-8: Usabilidad Identidad-Pregunta 5

Identidad	Si	No	%
5.- ¿Distingue alguna imagen que represente (logotipo) a la institución y cree que aparece en un lugar importante dentro de la página?	10	0	100

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

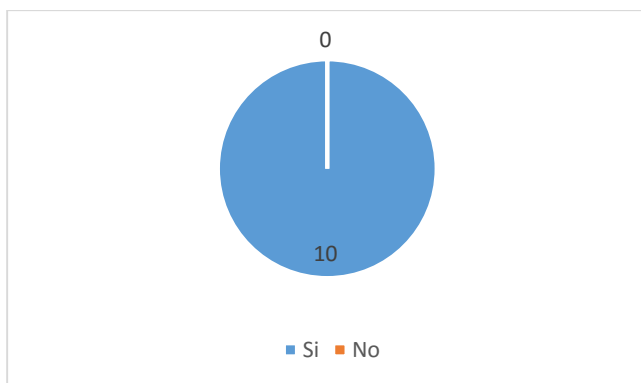


Figura 3-7: Pregunta 5 Usabilidad-Identidad

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 3-9: Usabilidad Identidad-Pregunta 6

Identidad	Si	No	%
6.- ¿Cree usted que este sitio está dirigido hacia Graduados de la Institución, autodidactas y estudiantes de otras universidades?	9	1	90

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

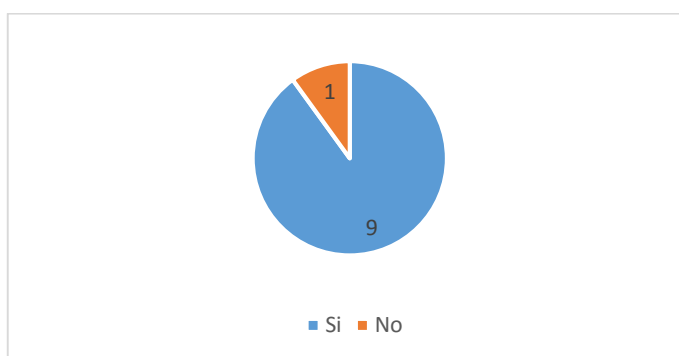


Figura 3-8: Pregunta 6 Usabilidad-Identidad

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Análisis: Se realizaron 6 interrogantes que tienen que ver con Identidad de las cuales se obtuvo una aceptación del 93.3% lo que nos dice que la Identidad del sitio es aceptada en dicho porcentaje.

Tabla 3-10: Preguntas Test Usabilidad-Contenido

Contenido	Si	No	%
1.- ¿Le parece adecuada la selección de contenidos destacados en la aplicación web?	10	0	100
2.- ¿Al ver la aplicación web, pudo distinguir de una sola mirada cuál era el contenido más relevante que se ofrecía?	10	0	100
3.- ¿Es fácil distinguir los datos que deben ser ingresados en la aplicación web?	10	0	100
4.- ¿Los datos a ser ingresados son suficientemente descriptivos?	9	1	90
5.- ¿Encontró información redundante en la aplicación web?	1	9	90
TOTAL			96

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 3-11: Test Usabilidad-Contenido Pregunta 1

Contenido	Si	No	%
1.- ¿Le parece adecuada la selección de contenidos destacados en la aplicación web?	10	0	100

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

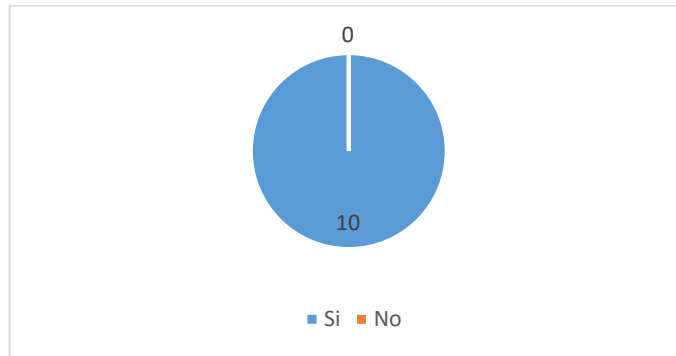


Figura 3-9: Pregunta 1 Usabilidad-Contenido

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 3-12: Test Usabilidad-Contenido Pregunta 2

Contenido	Si	No	%
2.- ¿Al ver la aplicación web, pudo distinguir de una sola mirada cuál era el contenido más relevante que se ofrecía?	10	0	100

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

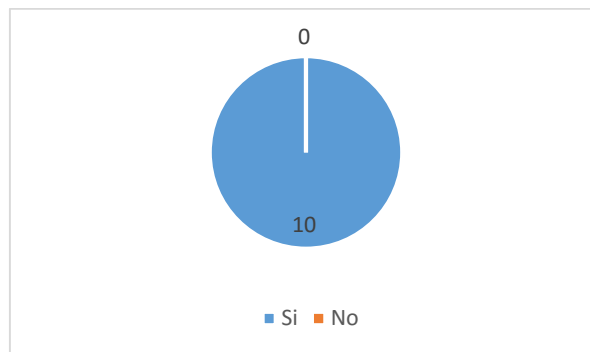


Figura3-10: Pregunta 2 Usabilidad-Contenido

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 3-13.: Test Usabilidad-Contenido Pregunta 3

Contenido	Si	No	%
3.- ¿Es fácil distinguir los datos que deben ser ingresados en la aplicación web?	10	0	100

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

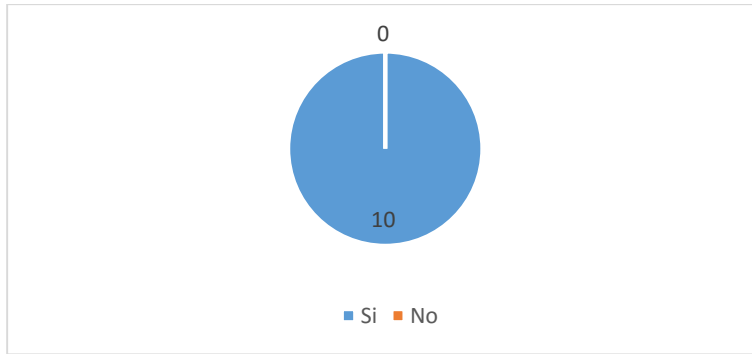


Figura 3-11: Pregunta 3 Usabilidad-Contenido

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 3-14: Test Usabilidad-Contenido Pregunta

Contenido	Si	No	%
4.- ¿Los datos a ser ingresados son suficientemente descriptivos?	9	1	90

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

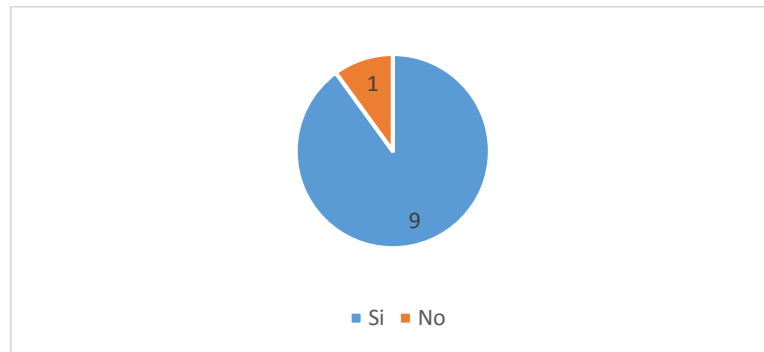


Figura 3-12: Pregunta 4 Usabilidad-Contenido

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 3.15: Test Usabilidad-Contenido Pregunta 5

Contenido	Si	No	%
5.- ¿Encontró información redundante en la aplicación web?	1	9	90

Realizado por: Carlos Mejia, Jhonny Guevara, 2016



Figura 3-13: Pregunta 5 Usabilidad-Contenido

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Análisis: Se realizaron 5 interrogantes que tienen que ver con el contenido del sitio de las cuales se obtuvo una aceptación del 96% lo que nos dice que el contenido del sitio es aceptado en dicho porcentaje.

Observaciones Pregunta 5: Redunda el nivel de un curso en la página principal y la página de ese curso.

Tabla 3.16: Preguntas Test Usabilidad-Navegación

Navegación	Si	No	%
1.- ¿La forma en que se navega por la aplicación web, y sus diferentes opciones, es clara? ¿Se distingue fácilmente?	10	0	100
2.- ¿Existen elementos dentro de las páginas, que le permitan saber exactamente dónde se encuentra dentro de la aplicación web y cómo volver atrás sin usar los botones del navegador?	10	0	100
3.- ¿Ve alguna forma de volver a la página de inicio desde cualquier página de la aplicación web, que no sea presionando algún botón del navegador?	10	0	100
4. ¿Logra distinguir gráficamente los datos que ya han sido registrados?	10	0	100
TOTAL			100

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 3.17: Test Usabilidad-Navegación Pregunta 1

Navegación	Si	No	%
1.- ¿La forma en que se navega por la aplicación web, y sus diferentes opciones, es clara? ¿Se distingue fácilmente?	10	0	100

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

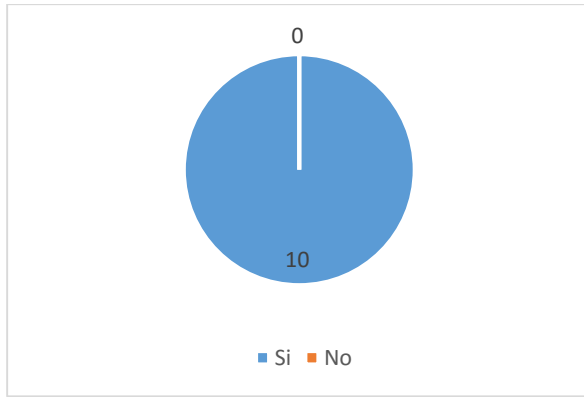


Figura 3-14: Pregunta 1 Usabilidad-Navegación

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 3.18: Test Usabilidad-Navegación Pregunta 2

Navegación	Si	No	%
2.- ¿Existen elementos dentro de las páginas, que le permitan saber exactamente dónde se encuentra dentro de la aplicación web y cómo volver atrás sin usar los botones del navegador?	10	0	100

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

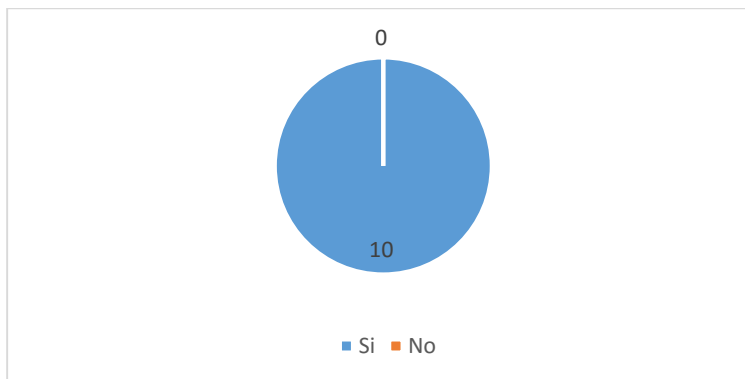


Figura 3-15: Pregunta 2 Usabilidad-Navegación

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 3.19: Test Usabilidad-Navegación Pregunta 3

Navegación	Si	No	%
3.- ¿Ve alguna forma de volver a la página de inicio desde cualquier página de la aplicación web, que no sea presionando algún botón del navegador?	10	0	100

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

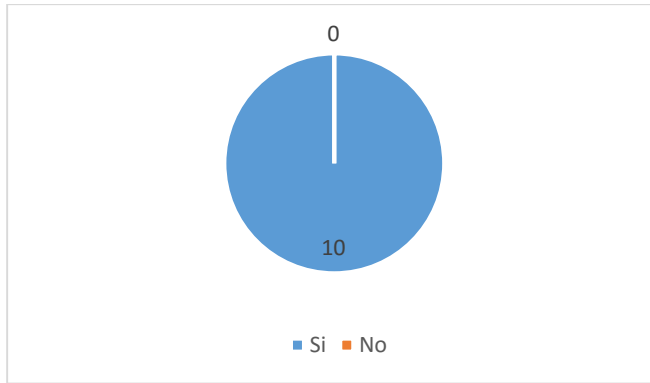


Figura 3-16: Pregunta 3 Usabilidad-Navegación

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 3.20: Test Usabilidad-Navegación Pregunta 4

Navegación		Si	No	%
4.	¿Logra distinguir gráficamente los datos que ya han sido registrados?	10	0	100

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

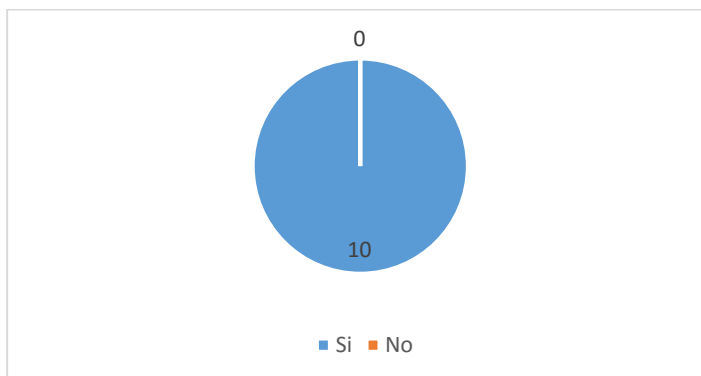


Figura 3-17: Pregunta 4 Usabilidad-Navegación

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Análisis: Se realizaron 4 interrogantes que tienen que ver con la navegación del sitio de las cuales se obtuvo una aceptación del 100% lo que nos dice que la navegación del sitio es aceptada en su totalidad

Tabla 3.21: Preguntas Test Usabilidad-Utilidad

Utilidad		Si	No	%
1.-	¿Tras una primera mirada, le queda claro cuál es el objetivo de la aplicación web, contenidos y servicios que ofrece?	9	0	90

2.- ¿Cree que los contenidos y servicios que se ofrecen en esta aplicación son de utilidad para su caso personal?	10	0	100
3.- ¿Qué es lo que más le llamó la atención positivamente o negativamente de la utilidad que ofrece la aplicación web? (si=positivamente; no=negativamente)	10	0	100
TOTAL			96.67

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 3.22: Test Usabilidad-Utilidad Pregunta 1

Utilidad	Si	No	%
1.- ¿Tras una primera mirada, le queda claro cuál es el objetivo de la aplicación web, contenidos y servicios que ofrece?	9	0	90

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

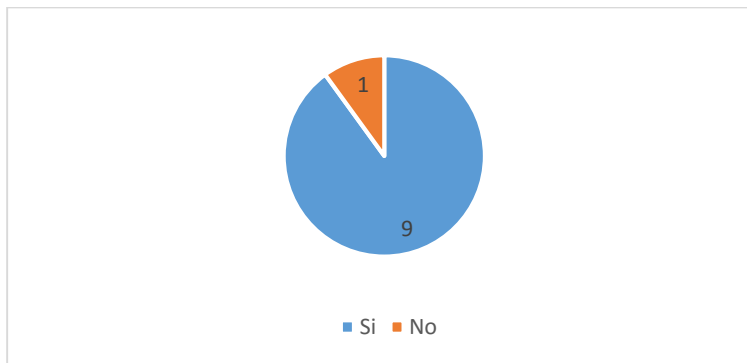


Figura 3-18: Pregunta 1 Usabilidad-Utilidad

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 3.23: Test Usabilidad-Utilidad Pregunta 2

Utilidad	Si	No	%
2.- ¿Cree que los contenidos y servicios que se ofrecen en esta aplicación son de utilidad para su caso personal?	10	0	100

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

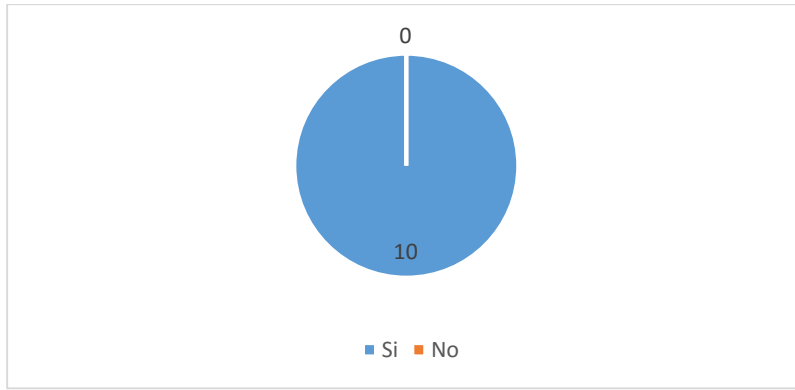


Figura 3-19: Pregunta 2 Usabilidad-Utilidad

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Tabla 3.24: Test Usabilidad-Utilidad Pregunta 3

Utilidad	Si	No	%
3.- ¿Qué es lo que más le llamó la atención positivamente o negativamente de la utilidad que ofrece la aplicación web? (si=positivamente; no=negativamente)	10	0	100

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

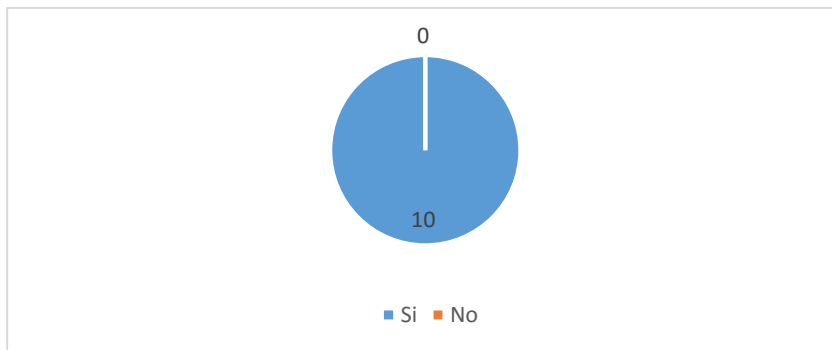


Figura 3-20: Pregunta 3 Usabilidad-Utilidad

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Análisis: Se realizaron 3 interrogantes que tienen que ver con la utilidad del sitio de las cuales se obtuvo una aceptación del 96.67% lo que nos dice que la navegación del sitio es aceptada en dicho porcentaje.

Tabla 3-25: Preguntas Test Usabilidad-Retroalimentación

Retroalimentación	Si	No	%
1.- Al momento de registrar los datos en la aplicación web ¿Ocurrió algún error? Describa el evento	0	10	100
TOTAL			100

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

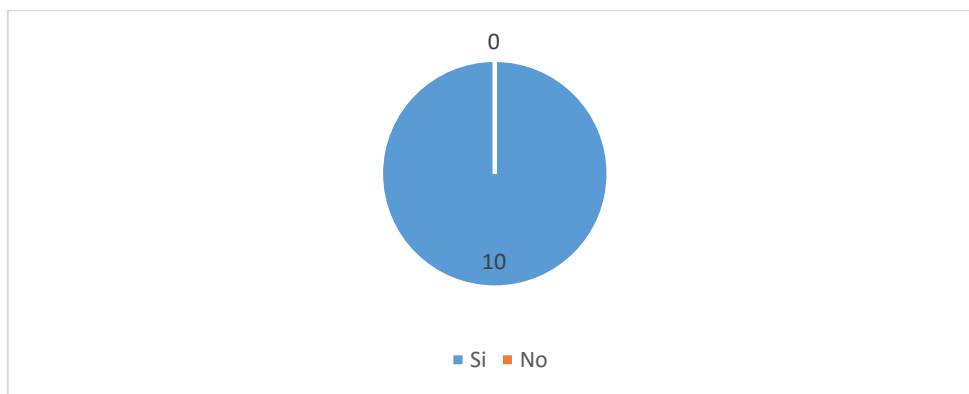


Figura 3-21: Pregunta 3 Usabilidad-Retroalimentación

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

2.- Puede expresar sus comentarios que permitan mejorar la calidad de la aplicación.

Observaciones Pregunta 2:

1. Sería bueno tener un botón de compartir con Facebook
2. Quisiera que los visitantes puedan dejar comentarios
3. Sería deseable un chat con los docentes
4. Me gustaría que la aplicación se pueda usar desde este momento.
5. **Análisis:** Se realizaron 2 interrogantes que tienen que ver con la retroalimentación del sitio de las cuales se obtuvo la primera una aceptación positiva del 100% y en la segunda en un espacio para dejar sugerencias se obtuvo las siguientes

Tabla 3-26: Resumen de Datos Test Usabilidad

Parámetro	%
Identidad	19
Contenido	19
Utilidad	20
Navegación	20
Retroalimentación	20
No aceptado	2
TOTAL	98

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

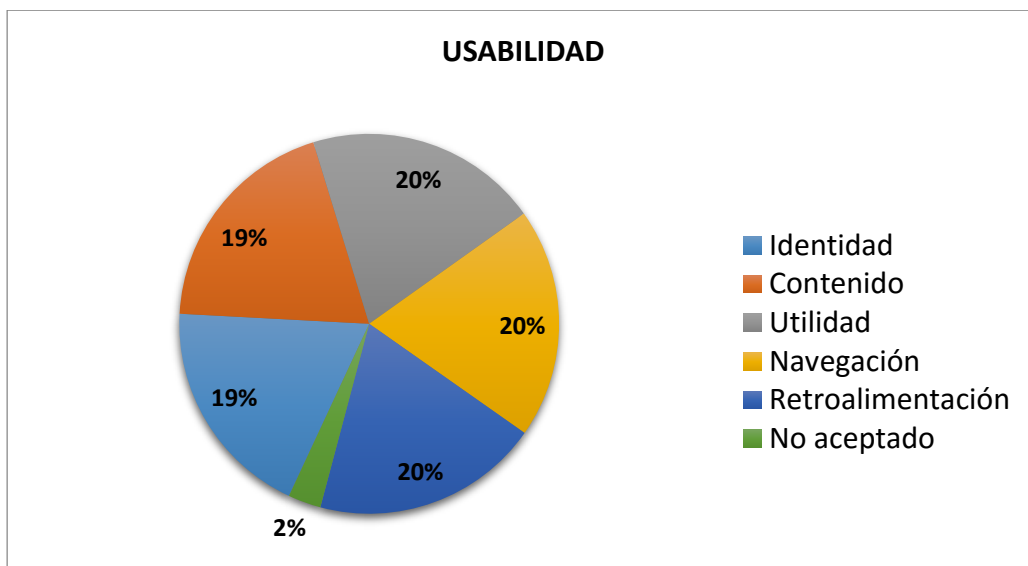


Figura 3-22: Resumen del Test Completo de Usabilidad

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

Análisis Final: con la evaluación de la identidad, contenido, utilidad, navegación y retroalimentación realizada por los docentes se logró evaluar a la usabilidad del sistema se establece como aceptada obteniendo un resultado del 98%, considerado un producto novedoso y de manejo intuitivo.

3.2.1 Pruebas de aceptación

Las Pruebas de Aceptación son las características con las que debe cumplir un requerimiento (historia de usuario) para ser aceptado, está relacionado principalmente con la funcionalidad.

Las pruebas de aceptación son evaluadas por el facilitador que en este caso es el administrador de aplicaciones.

Se debe tener en cuenta que las revisiones fueron realizadas mes a mes según la planificación de los Sprints y versiones a evaluar desde la finalización del primer sprint hasta la finalización del último sprint

Como resultado de la evaluación de las pruebas de determinó la siguiente tabla:

Tabla 3-27: Total de Historias de Usuario Realizadas

Detalle	Fecha Inicio Revisión	Fecha Fin Revisión	Pruebas Exitosas	Pruebas Fallidas
36 Historias de Usuario	10-10-2015	04-02-2016	100	0

Realizado por: Carlos Mejia, Jhonny Guevara, 2016



Figura 3-23: Historias de Usuario

Realizado por: Carlos Mejia, Jhonny Guevara, 2016

El detalle de cada una de las Historias Técnicas, Historias de Usuario y pruebas de aceptación se encuentran en el Manual Técnico Adjunto en el Anexo.

CONCLUSIONES

- El número de usuarios que actualmente soporta el entorno Virtual de Aprendizaje de la ESPOCH es de 655 de los cuales solo 376 lo utilizaron en los dos últimos periodos académicos representando un 57.45%. Estos datos sirvieron como un indicador en cuanto a la cantidad de usuarios que se estima que harán uso del sistema Open Course Ware ESPOCH.
- Se realizó el análisis de requerimientos conjuntamente con los usuarios finales del producto definiendo así un Product Backlog de 25 historias de usuario y una Pila de Sprints de 21 según el marco de trabajo utilizado SCRUM.
- El sistema Open Course Ware fue integrado con el sistema OASIS de la institución mediante el consumo de servicios web que provee el DTICS dando mayor productividad de los datos existentes y evitando la réplica de los mismos.
- El porcentaje de usabilidad que se obtuvo en la capacitación de los usuarios fue de un 98%, demostrando que el sistema informático Open Course Ware cumple exitosamente con las métricas de efectividad, eficiencia y satisfacción impuestas por el indicador de usabilidad.
- El desarrollo del sistema informático Open Course Ware concluyo satisfactoriamente en los 4 meses planificados cumpliendo así con la requerimentación y tiempo acordado.

RECOMENDACIONES.

- La utilización de una metodología o marco de trabajo ágil en el desarrollo de un proyecto permite llevar un control y respuesta flexible ante el cambio dando así la posibilidad de administrar de mejor manera aspectos importantísimos como tiempo, recursos humanos, recursos económicos etc.
- En el desarrollo de un proyecto es vital la integración del usuario o usuarios implicados en el grupo de trabajo ya que todo lo que se desarrolla gira en torno sus necesidades y aceptación constante

BIBLIOGRAFÍA

- Almirón, C. G.** (2012). *adictosaltrabajo*. Recuperado el 27/04/2016 de <https://www.adictosaltrabajo.com/tutoriales/introduccion-jsf-java/>
- Araya, P.** (2012). *Administración del SGDB PostgreSQL*. Naucalpan de Juárez - Mexico. Pearson.
- Caules., C. Á.** (2014). *JPA Domain Driven Design*. USA. Alfa Omega.
- Izaurrealde, M. P.** (2013). *Requerimientos en metodologías ágiles*. Recuperado el 25/04/2016 de http://www.institucional.frc.utn.edu.ar/sistemas/lidicalso/pub/file/Tesis/Anteproyecto_Requerimientos_en_Metodologías_Agiles.pdf
- Jesús, C. S.** (2014). *slideshare - Arquitectura de integración de servicios*. Recuperado el 19/04/2016 de <http://es.slideshare.net/Coatzozon20/arquitectura-de-integracin-de-servicios>
- juntadeandalucia.** (2014). *Contenido y gestión de recursos*. Recuperado el 01/04/2016 de <http://www.juntadeandalucia.es/servicios/madeja/sites/default/files/historico/1.4.0/contenido-recurso-96.html>
- Lerma, E. V.** (2015). *Introducción a primefaces*. Recuperado el 06/04/2016 de <https://www.adictosaltrabajo.com/tutoriales/introduccion-primefaces/>
- microbuffer.** (2014). *Microbuffer*. Recuperado el 15/04/2016 de <https://microbuffer.wordpress.com/2011/05/04/que-es-postgresql/>
- Oliveira, H. D.** (2012). *Burndown Chart*. Recuperado el 25/03/2016 de <http://agilizandoando.blogspot.com/2012/06/burndown-chart-quemando-tareas.html>
- Parr., D. B.** (2012). *Java para Estudiantes*. Naucalpan de Juárez - Mexico. Pearson.
- Primefaces.** (2013). *PRIMEFACES USER GUIDE*. USA. Alfa Omega.
- Raoul-Gabriel Urma, M. F.** (2014). *Java EE 7 in action*. USA. Manning Publications.
- superior, c. d.** (2013). *CES*. Recuperado el 23/03/2016 de <http://www.ces.gob.ec/descargas/reglamento-general-de-la-ley-organica-de-educacion-superior>
- Vega, A. A.** (2015). *Manejo de memoria*. Recuperado el 25/03/2016 de <http://dspace.uah.es/dspace/bitstream/handle/10017/19972/Memoria.pdf?sequence=1>

Vigo, U. d. (2009). *Revisión de los servicios web SOAP*. Recuperado el 25/04/2016 de http://www.albertolsa.com/wp-content/uploads/2009/07/mdsw-revision-de-los-servicios-web-soap_rest-alberto-los-santos.pdf

www.jy-soft.com. (2015). *J-Query*. Recuperado el 18/03/2016 de <http://www.jy-soft.com/Soporte.php>

ANEXOS