



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

**“DESARROLLO DE UNA APLICACIÓN WEB PARA
AUTOMATIZAR EL “CONTROL DE SERVICIOS” DEL
DEPARTAMENTO DE EQUIDAD, GÉNERO Y ACCIÓN SOCIAL-
CEFAS”**

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

AUTOR: DUCHITANGA CUMBE JULIO CRISTIAN

TUTOR: ING. LORENA AGUIRRE

Macas, Ecuador

2016

©2016, Julio Cristian Duchitanga Cumbe

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.

Julio Cristian Duchitanga Cumbe

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

El Tribunal del trabajo de titulación certifica que: “DESARROLLO DE UNA APLICACIÓN WEB PARA AUTOMATIZAR EL “CONTROL DE SERVICIOS” DEL DEPARTAMENTO DE EQUIDAD, GÉNERO Y ACCIÓN SOCIAL-CEFAS ”, de responsabilidad del señor Julio Cristian Duchitanga Cumbe, ha sido minuciosamente revisado por los Miembros del Tribunal del trabajo de titulación, quedando autorizada su presentación.

ING. WASHINGTON LUNA

DECANO DE LA FACULTAD DE -----
INFORMÁTICA Y ELECTRÓNICA

ING. PATRICIO MORENO

DIRECTOR DE ESCUELA DE -----
INGENIERÍA EN SISTEMAS

ING. LORENA AGUIRRE

DIRECTOR -----

ING. EDUARDO VILLA

MIEMBRO DE TRIBUNAL -----

DECLARACIÓN DE AUTENTICIDAD

Yo, Julio Cristian Duchitanga Cumbe, declaro que el presente trabajo de titulación es de mi autoría y que los resultados del mismo son auténticos y originales. Los textos constantes en el documento que provienen de otra fuente están debidamente citados y referenciados.

Como autor, asumo la responsabilidad legal y académica de los contenidos de este trabajo de titulación.

Riobamba, 21, septiembre 2016

Julio Cristian Duchitanga Cumbe

1400738694

DEDICATORIA

A mi madre María por haberme apoyado durante todo este trayecto de mi vida estudiantil sin perecer, además quiero dedicar este trabajo a mi esposa Gloria y a mi hijo Maykel por ser una fuente de inspiración en los momentos más difíciles, y a todos aquellos que aportaron un granito de arena para cumplir esta meta, quiero dedicar de manera especial al ser más importante del mundo a mi Dios sagrado por mostrarme el camino correcto y ayudarme a superar las travesías engañosas de la vida.

AGRADECIMIENTO

Quiero agradecerá mis profesores quienes fueron una pieza fundamental para mi formación académica además por enseñarme el lado humanístico que debe poseer un profesional de calidad, además debo agradecer a toda mi familia en especial a mi madre y esposa por ser un apoyo moral, sentimental y económico que permitieron cumplir esta meta ansiada.

TABLA DE CONTENIDO

PORTADA.....	i
DERECHOS DE AUTOR	ii
CERTIFICACIÓN.....	iii
DECLARACIÓN DE AUTENTICIDAD.....	iv
DEDICATORIA.....	v
AGRADECIMIENTO	vi
TABLA DE CONTENIDO.....	vii
ÍNDICE DE TABLAS.....	xi
ÍNDICE DE FIGURAS.....	xv
ÍNDICE DE ANEXOS	xviii
RESUMEN.....	xix
SUMMARY.....	xx

INTRODUCCIÓN

CAPITULO I

1. MARCO TEÓRICO	7
1.1. Aplicación web.....	7
<i>1.1.1. ¿Como funcionan?.....</i>	<i>7</i>
<i>1.1.2. Ventajas</i>	<i>7</i>
<i>1.1.3. Desventajas</i>	<i>7</i>
1.2. Framework	7
<i>1.2.1. Ventajas</i>	<i>8</i>
<i>1.2.2. Desventajas</i>	<i>8</i>
1.3. Patrón MVC(Modelo, Vista, Controlador).....	8
<i>1.3.1. Modelo</i>	<i>8</i>
<i>1.3.2. Vista</i>	<i>8</i>
<i>1.3.3. Controlador</i>	<i>8</i>
1.4. Metodología XP(Programación Extrema)	9
<i>1.4.1. Proceso de desarrollo</i>	<i>9</i>
<i>1.4.1.1. Objetivos</i>	<i>9</i>
<i>1.4.1.2. Características</i>	<i>9</i>
<i>1.4.1.3. Ciclo de Vida.....</i>	<i>9</i>

1.4.1.4.	<i>Fases</i>	10
1.4.1.5.	<i>Principios</i>	10
1.4.1.6.	<i>Ventajas</i>	10
1.4.1.7.	<i>Desventajas</i>	11
1.5.	PostgreSQL	11
1.5.1.	<i>Ventajas</i>	11
1.5.2.	<i>Desventajas</i>	11
1.6.	PgAdmin 3	11
1.6.1.	<i>Características</i>	12
1.7.	Python	12
1.7.1.	<i>Características</i>	12
1.7.2.	<i>Ventajas</i>	12
1.8.	Framework DJANGO	13
1.8.1.	<i>Patrón de diseño MTV</i>	13
1.8.2.	<i>Proyecto Django</i>	13
1.8.3.	<i>Servidor de desarrollo</i>	14
1.8.4.	<i>Sistema de plantilla</i>	14
1.8.4.1.	<i>Herencia de plantillas</i>	15
1.8.5.	<i>Modelos</i>	16
1.8.5.1.	<i>Configuración de base de datos</i>	16
1.8.5.2.	<i>Aplicación Django</i>	17
1.8.5.3.	<i>Como crear Modelos</i>	18
1.8.6.	<i>Sitio de administración</i>	18
1.8.6.1.	<i>Interfaz de administración</i>	18
1.8.7.	<i>Formularios usando clases</i>	19
1.8.8.	<i>URLconf</i>	20
1.8.9.	<i>Vistas</i>	21
1.8.9.1.	<i>Vistas Genéricas basadas en clases</i>	21
1.8.10.	<i>Sesiones</i>	21
1.8.11.	<i>Usuarios e identificación</i>	22
1.8.12.	<i>Seguridad</i>	22
1.9.	ReportLab	22
1.9.1.	<i>Ventajas</i>	23
1.9.2.	<i>Desventajas</i>	23
1.10.	Bootstrap	23
1.10.1.	<i>Ventajas</i>	23

1.10.2.	<i>Desventajas</i>	23
1.11.	JavaScript	24
1.11.1.	<i>Uso</i>	24
1.11.2.	<i>Ventajas</i>	24
1.11.3.	<i>Desventajas</i>	25
1.12.	Ajax	25
1.12.1.	<i>Ventajas</i>	25
1.12.2.	<i>Desventajas</i>	25

CAPITULO II

2.	MARCO METODOLÓGICO	26
2.1.	Metodología XP (programación extrema)	27
2.2.	Fase de Planificación	28
2.2.1.	<i>Planificación Inicial</i>	28
2.3.1.1.	<i>Recopilación de información</i>	28
2.2.2.	<i>Arquitectura de la aplicación web</i>	29
2.2.3.	<i>Módulos de la aplicación web SIGEGAS</i>	29
2.2.4.	<i>Cronograma de actividades</i>	30
2.2.5.	<i>Planificación de Publicaciones</i>	31
2.3.5.1.	<i>Iteraciones</i>	31
2.2.6.	<i>Historias de usuario</i>	33
2.3.	Fase de Diseño	37
2.3.1.	<i>Diseño de base de datos</i>	37
2.3.2.	<i>Diseño de Interfaces</i>	39
2.3.3.	<i>Casos de uso</i>	47
2.3.4.	<i>Diagrama de procesos</i>	67
2.4.4.1.	<i>Diagramas del módulo de posada y comedor</i>	67
2.4.4.2.	<i>Diagramas del módulo de ayuda social</i>	69
2.4.4.3.	<i>Diagramas del módulo de capacitaciones</i>	70
2.4.4.4.	<i>Diagrama de Reportes</i>	71
2.3.5.	<i>Arquitectura de desarrollo</i>	72
2.3.6.	<i>Tareas por historias de usuario</i>	72
2.3.6.1.	<i>Tareas primera iteración.</i>	72
2.3.6.2.	<i>Tareas segunda iteración.</i>	78
2.3.6.3.	<i>Tareas tercera iteración.</i>	82

2.4.	Fase de Codificación	87
2.4.1.	Estándar de diseño de base de datos	87
2.4.1.1.	<i>Nombre de entidades</i>	87
2.4.1.2.	<i>Nombre de atributos</i>	87
2.4.2.	Estándar de codificación	87
2.4.2.1.	<i>Tipos de archivos</i>	88
2.4.3.	SIGEGAS	88
2.4.3.1.	<i>Modelos (M)</i>	89
2.4.3.2.	<i>Vistas (V)</i>	89
2.4.3.3.	<i>Plantillas (T)</i>	90
2.4.4.	Diagrama de despliegue	90
2.4.5.	Diagrama de componentes	91
2.5.	Fase de Pruebas	92
2.5.1.	Pruebas primera iteración	92
2.5.2.	Pruebas segunda iteración	99
2.5.3.	Pruebas tercera iteración	104

CAPÍTULO III

3.	MARCO DE RESULTADO	109
3.1.	Análisis de resultados	111
3.1.1.	<i>Valores observados totales</i>	121
3.2.	Ventajas de la aplicación web SIGEGAS	122
CONCLUSIONES		123
RECOMENDACIONES		124
BIBLIOGRAFÍA		
ANEXOS		

ÍNDICE DE TABLAS

Tabla 1-2:	Ventajas y Desventajas de Django.....	26
Tabla 2-2:	Roles del Equipo de Trabajo	28
Tabla 3-2:	Historias de Usuario – Primera Iteración	31
Tabla 4-2:	Historias de Usuario – Segunda Iteración	32
Tabla 5-2:	Historias de Usuario – Tercera Iteración.....	33
Tabla 6-2:	Acceso al sistema	33
Tabla 7-2:	Historias de Usuario Gestión de usuario	34
Tabla 8-2:	Historia de Usuario Gestión de beneficiario.....	34
Tabla 9-2:	Historias de Usuario Asignar usuarios a un grupo	34
Tabla 10-2:	Historias de Usuario Gestión de Posada.....	35
Tabla 11-2:	Historias de Usuario Seguimiento de entrada y salida de Posantes.....	35
Tabla 12-2:	Historias de Usuario Gestión de alimentación diaria	35
Tabla 13-2:	Historias de Usuario Gestión de ayudas sociales	36
Tabla 14-2:	Historias de Usuario Gestión de recursos.....	36
Tabla 15-2:	Historias de Usuario Gestión de capacitaciones.....	36
Tabla 16-2:	Historias de Usuario Inscripción a capacitaciones	37
Tabla 17-2:	Historia de Usuario Generar Reportes.....	37
Tabla 18-2:	Caso de uso Autenticar.....	48
Tabla 19-2:	Caso de uso Gestionar Usuario.....	49
Tabla 20-2:	Caso de uso Asignar Grupo.....	49
Tabla 21-2:	Caso de uso Gestionar Funcionario.....	50
Tabla 22-2:	Caso de uso Gestionar Beneficiario.....	50
Tabla 23-2:	Caso de uso Gestionar Cantones.....	51
Tabla 24-2:	Caso de uso Gestionar Parroquias.....	51
Tabla 25-2:	Caso de uso Gestionar Comunidades.....	52
Tabla 26-2:	Caso de uso Gestionar Programas.....	52
Tabla 27-2:	Caso de uso Autenticar Responsable del Centro Intercultural.....	53
Tabla 28-2:	Caso de uso Crear Posada.....	54
Tabla 29-2:	Caso de uso Buscar Beneficiario.....	54
Tabla 30-2:	Caso de uso Registrar Beneficiario.....	55
Tabla 31-2:	Caso de uso Registrar Alimentación.....	55
Tabla 32-2:	Caso de uso Generar Ficha.....	56
Tabla 33-2:	Caso de uso Marcar Salida.....	56
Tabla 34-2:	Caso de uso Autenticar a Ayuda Social.....	57

Tabla 35-2:	Caso de uso Crear Ayuda Social.	58
Tabla 36-2:	Caso de uso Buscar Beneficiario para Ayuda Social.....	58
Tabla 37-2:	Caso de uso Registrar Beneficiario para Ayuda Social.	59
Tabla 38-2:	Caso de uso Crear consultar historial de ayudas.	59
Tabla 39-2:	Caso de uso Gestionar Recursos.....	60
Tabla 40-2:	Caso de uso Autenticar Capacitador.....	61
Tabla 41-2:	Caso de uso inscribir beneficiarios.....	61
Tabla 42-2:	Caso de uso Aprobar o Reprobar Capacitación.....	62
Tabla 43-2:	Caso de uso Finalizar Capacitación.....	62
Tabla 44-2:	Caso de uso Buscar Beneficiario para Inscribir a Capacitación.	63
Tabla 45-2:	Caso de uso Registrar Beneficiario para Capacitación.....	63
Tabla 46-2:	Caso de uso Autenticar Asistente Administrativo	64
Tabla 47-2:	Caso de uso Crear Capacitación.	65
Tabla 48-2:	Caso de uso Publicar Capacitación.....	65
Tabla 49-2:	Caso de uso Generar Certificado.	66
Tabla 50-2:	Tareas de HU Acceso al sistema.	72
Tabla 51-2:	Configuración del entorno de desarrollo.	72
Tabla 52-2:	Diseño e implementación de base de datos.	73
Tabla 53-2:	Diseño de plantilla para login.....	73
Tabla 54-2:	Implementación de vistas en Django.....	73
Tabla 55-2:	Tareas de HU Gestión de usuario.....	73
Tabla 56-2:	Instalar tablas propias de Django.....	74
Tabla 57-2:	Activar la interfaz administrativa de Django.....	74
Tabla 58-2:	Tareas de HU gestión de beneficiarios.	74
Tabla 59-2:	Crear formulario basado en clases.....	75
Tabla 60-2:	Diseñar plantilla para ingresar, editar.....	75
Tabla 61-2:	Crear patrón url y vista (funciones) Django.	75
Tabla 62-2:	Tareas de HU Asignar usuarios a un grupo.....	75
Tabla 63-2:	Registrar grupos mediante la interfaz de administración.	76
Tabla 64-2:	Asignar usuarios registrados a un grupo.....	76
Tabla 65-2:	Tareas de HU Gestión de posadas.	76
Tabla 66-2:	Crear formulario basado en clases.....	76
Tabla 67-2:	Diseñar plantilla para crear o registrar posada.	77
Tabla 68-2:	Crear patrón url y vista (funciones) Django.	77
Tabla 69-2:	Listar posadas creadas.	77
Tabla 70-2:	Generar ficha de posada creada.	77
Tabla 71-2:	Tareas de HU Seguimiento de entrada y salida de posantes.	78

Tabla 72-2:	Crear patrón url y vista para salida de posante.....	78
Tabla 73-2:	Tareas de HU Gestión de alimentación diaria.....	78
Tabla 74-2:	Crear formulario basado en clases.....	78
Tabla 75-2:	Diseñar plantilla para crear o registrar alimentación.....	79
Tabla 76-2:	Crear patrón url y vista (funciones) Django.....	79
Tabla 77-2:	Tareas de HU Gestión de ayudas sociales.....	80
Tabla 78-2:	Crear formulario basado en clases.....	80
Tabla 79-2:	Diseñar plantilla para crear o registrar ayuda.....	80
Tabla 80-2:	Crear patrón url y vista (funciones) Django.....	81
Tabla 81-2:	Tareas de HU Gestión de recursos.....	81
Tabla 82-2:	Crear formulario basado en clases.....	81
Tabla 83-2:	Diseñar plantilla para crear o registrar recursos.....	81
Tabla 84-2:	Crear patrón url y vista (funciones) Django.....	82
Tabla 85-2:	Tareas de HU Gestión de capacitaciones.....	82
Tabla 86-2:	Crear formulario basado en clases.....	82
Tabla 87-2:	Diseñar plantilla para crear o registrar capacitaciones.....	83
Tabla 88-2:	Crear patrón url y vista (funciones) Django.....	83
Tabla 89-2:	Tareas de HU Inscripción a capacitaciones.....	84
Tabla 90-2:	Crear formulario basado en clases.....	84
Tabla 91-2:	Diseñar plantilla para inscripciones.....	84
Tabla 92-2:	Crear patrón url y vista (funciones) Django.....	85
Tabla 93-2:	Tareas de HU Generar reportes.....	85
Tabla 94-2:	Instalar y configurar la librería Reportlab en Django.....	85
Tabla 95-2:	Crear formulario basado en clases.....	86
Tabla 96-2:	Diseñar plantilla Django para reportes simples y con parámetros.....	86
Tabla 97-2:	Crear patrón url y vista (funciones) Django.....	86
Tabla 98-2:	Extensiones de archivos.....	88
Tabla 99-3:	Descripción de procesos manuales a automatizar.....	109
Tabla 100-3:	Resultados proceso 1.....	111
Tabla 101-3:	Resultados proceso 2.....	112
Tabla 102-3:	Resultados proceso 3.....	112
Tabla 103-3:	Resultados proceso 4.....	113
Tabla 104-3:	Resultados proceso 5.....	114
Tabla 105-3:	Resultados proceso 6.....	114
Tabla 106-3:	Resultado proceso 7.....	115
Tabla 107-3:	Resultados pregunta 8.....	116
Tabla 108-3:	Resultados proceso 9.....	116

Tabla 109-3:	Resultados proceso 10.	117
Tabla 110-3:	Resultados proceso 11.	118
Tabla 111-3:	Resultados proceso 12.	118
Tabla 112-3:	Resultados proceso 13.	119
Tabla 113-3:	Resultados proceso 14.	120
Tabla 114-3:	Valores observados.....	121

ÍNDICE DE FIGURAS

Figura 1-1	Ciclo de vida metodología XP.....	9
Figura 2-1	Plantilla inventario.htm.	15
Figura 3-1	Plantilla base.htm.	16
Figura 4-1	Configuración para PostgreSQL.	17
Figura 5-1	Modelos de tablas de base de datos.....	18
Figura 6-1	Índice principal de la administración Django.....	19
Figura 7-1	Formulario basado en clases Django.....	19
Figura 8-1	Vista recurso () para formulario.	20
Figura 9-1	Ejemplo de Urlconf Django.	20
Figura 10-1	Patrones comunes para Urlconf Django.....	21
Figura 11-1	Uso de JavaScript.....	24
Figura 12-2	Arquitectura Metodología XP (Programación Extrema).....	27
Figura 13-2	Arquitectura de la aplicación web.	29
Figura 14-2	Módulos de la aplicación web SIGEGAS.	29
Figura 15-2	Cronograma de actividades.	31
Figura 16-2	Tiempos de entrega de la primera iteración.	32
Figura 17-2	Tiempos de entrega de la segunda iteración.....	32
Figura 18-2	Tiempos de entrega de la segunda iteración.....	33
Figura 19-2	Esquema de base de datos.	38
Figura 20-2	Esquema de base de datos Django.	38
Figura 21-2	Diseño de interfaz base.	39
Figura 22-2	Gestión de posada.....	39
Figura 23-2	Ingreso a posada.	40
Figura 24-2	Seguimiento de entrada y salida de posante.....	40
Figura 25-2	Gestión de habitaciones.....	40
Figura 26-2	Registro de beneficiarios.....	41
Figura 27-2	Reg. De alimentación (beneficiarios).....	41
Figura 28-2	Registro de alimentación (funcionarios).	42
Figura 29-2	Lista de deuda alimentación (funcionario).....	42
Figura 30-2	Registro de Ayudas Sociales.....	43
Figura 31-2	Registro de recursos.	43
Figura 32-2	Lista de recursos con stock cero.....	43
Figura 33-2	Aumentar Stock.....	44
Figura 34-2	Historial de ayudas por beneficiario.....	44

Figura 35-2	Registro de capacitación.....	44
Figura 36-2	Registro de programas.....	45
Figura 37-2	Lista de capacitaciones a publicar.....	45
Figura 38-2	Publicar capacitación.....	45
Figura 39-2	Capacitación publicada.	46
Figura 40-2	Inscribir a capacitación.....	46
Figura 41-2	Aprobar o reprobar participantes.....	46
Figura 42-2	Capacitaciones terminada.....	47
Figura 43-2	Generar o enlazar reportes.....	47
Figura 44-2	Diagrama de caso de uso administrador.....	48
Figura 46-2	Diagrama de caso de uso Centro Intercultural.	53
Figura 47-2	Diagrama de caso de uso Ayuda Social.	57
Figura 48-2	Diagrama de caso de uso Capacitador.....	60
Figura 49-2	Diagrama de caso de uso Asistente Administrativo.....	64
Figura 50-2	Proceso para acceso al sistema.....	67
Figura 51-2	Crear usuario y asignar grupo.	68
Figura 52-2	Crear posada.....	68
Figura 53-2	Salida de posante.....	69
Figura 54-2	Registro de alimentación.....	69
Figura 55-2	Registro de ayuda social.....	70
Figura 56-2	Registro de recurso.....	70
Figura 57-2	Registro de una capacitación.....	70
Figura 58-2	Inscripción de personas.....	71
Figura 59-2	Emitir reporte.	71
Figura 60-2	Arquitectura django.....	72
Figura 61-2	Estructura SIGEGAS.....	89
Figura 62-2	Estructura de módulos SIGEGAS.....	89
Figura 63-2	Diagrama de despliegue.....	90
Figura 64-2	Diagrama de componentes.....	91
Figura 65-3	Representación estadística proceso 1.....	111
Figura 66-3	Representación estadística proceso 2.....	112
Figura 67-3	Representación estadística proceso 3.....	113
Figura 68-3	Representación estadística proceso 4.....	113
Figura 69-3	Representación estadística proceso 5.....	114
Figura 70-3	Representación estadística proceso 6.....	115
Figura 71-3	Representación estadística proceso 7.....	115
Figura 72-3	Representación estadística proceso 8.....	116

Figura 73-3	Representación estadística proceso 9.	117
Figura 74-3	Representación estadística proceso 10.	117
Figura 75-3	Representación estadística proceso 11.	118
Figura 76-3	Representación estadística proceso 12.	119
Figura 77-3	Representación estadística proceso 13.	119
Figura 78-3	Representación estadística proceso 14.	120

ÍNDICE DE ANEXOS

- Anexo A:** Certificado por haber implementado la aplicación web SIGEGAS.
- Anexo B:** Capacitación al personal del departamento de Equidad, Genero y Acción Social.
- Anexo C:** Acuestas proceso manual.
- Anexo E:** Acuestas proceso automatizado.
- Anexo E:** Manual de usuario.

RESUMEN

Desarrollar una aplicación web de Gestión de Equidad, Género y Acción social (SIGEGAS) para automatizar los servicios del Departamento de Equidad, Género y Acción Social en la Coordinación de Equidad, Formación y Acción Social-CEFAS ubicado en la ciudad de Macas provincia de Morona Santiago. Para el desarrollo de la aplicación web se analizó el tiempo de los procesos manuales del departamento en las siguientes áreas: Posada y Comedor, Ayuda Social y Capacitaciones. La finalidad de la aplicación web fue la automatización de los procesos para agilizar el trabajo que realiza el personal de este departamento. Se utilizó el framework DJANGO, como herramienta de desarrollo. La reutilización de código en tareas comunes, y las clases genéricas que reducen el tiempo de desarrollo son una de sus características. La aplicación tiene la finalidad de optimizar los procesos garantizando mayor productividad al trabajo que realiza el personal del departamento. Se planteó la implementación de la metodología de desarrollo Programación Extrema (XP) como guía durante el ciclo de desarrollo de la aplicación. SIGEGAS cuenta con los siguientes módulos: Posada y Comedor, Ayuda Social, Capacitación y reportes desarrollados. Como resultado se determina que SIGEGAS reduce el tiempo en los procesos en comparación a procesos manuales en un 91.23 % de efectividad; Concluyendo que minimiza el tiempo al ejecutar los servicios en cada una de las áreas del departamento. Se recomienda al departamento de Equidad; Género y Acción Social la utilización de la aplicación web para mejorar los servicios que prestan.

Palabras Claves: < COORDINACIÓN DE EQUIDAD, FORMACIÓN Y ACCIÓN SOCIAL (CEFAS)>, <SISTEMA DE GESTIÓN DE EQUIDAD GENERO Y ACCIÓN SOCIAL (SIGEGAS)> <PROGRAMACIÓN EXTREMA (XP)> <FRAMEWORK DJANGO> <APLICACIÓN WEB> <MACAS (CANTÓN)> <MORONA SANTIAGO (PROVINCIA)>

SUMMARY

Develop a web application of Equity, Gender, and Social Action (SIGEGAS) management to automate the services of the Department of Equity, Gender and Social Action in the Equity Coordination, Training and Social Action-CEFAS located in the city of Macas, Morona Santiago Province. To develop the web application, the department's time of manual processes was analyzed in the following areas: Inn and Dinning, Social help and Training. The purpose of the web application was the automatization of the process to make easier tasks done by the department's staff. The DJANGO framework was used as a tool of development. Code reuse in common tasks, and the generic classes that reduce development time are some of its features. The application aims to optimize processes ensuring greater productivity work done by department staff. The implementation of the Extreme Programming (XP) methodology development was arise as a guide. SIGEGAS has the following modules: Inn and Dinning, Social Help, Training and development reports. As a result it is determined that SIGEGAS reduces the time in the processes in comparison to other manuals, 91,23% effectivity; concluding that minimizes time when running the services in each one the department's areas. It is recommended that the department of Equity, Genre and Social Action use the web application to improve the services provide.

Key Words: <EQUITY TRAINING AND SOCIAL ACTION COORDINATION (CEFAS)>
<SYSTEM OF GENRE EQUITY AND SOCIAL ACTION (SIGEGAS)> <EXTREME PROGRAMMING (XP)> <FRAMEWORK DJANGO> <WEB APPLICATION> <MACAS (CANTON)> <MORONA SANTIAGO (PROVINCE)>

INTRODUCCIÓN

Con el paso del tiempo el mundo institucional y empresarial ha buscado disminuir el tiempo en ejecutar procesos manuales, dando así, paso a las herramientas informáticas para solucionar esta carencia. Dentro de esta problemática se encuentra el departamento de Equidad, Género y Acción social que pertenece la institución pública solidaria CEFAS. Actualmente los procesos en este departamento se efectúan de forma manual y están divididos en Posada y Comedor, Ayudas Sociales y Capacitaciones.

El objetivo principal de este trabajo de titulación es desarrollar una aplicación web para automatizar el "control de servicios" del departamento de equidad y género y acción social-CEFAS; para de alguna forma agilizar los proceso y dar un seguimiento de la información, que sirva como ayuda en la toma de decisiones.

Se ha planteado realizar una aplicación web, considerado una herramienta informática para acceder desde cualquier parte, independiente de la plataforma que se utilice. Además para el diseño del front-end se considera los siguientes: Html, Css3, JavaScripts y Ajax.

Para el desarrollo se utilizó Django; un framework de alto nivel escrito en el lenguaje de programación Python, que ayuda al desarrollo y diseño rápido de aplicaciones. Finalmente para llevar un desarrollo organizado y que garantice la calidad de software se tomó como guía la metodología ágil XP (Programación Extrema).

El trabajo de titulación consta de 3 capítulos que se describen a continuación:

Capítulo I Marco Teórico, comprende conceptos, definiciones y características principales de las herramientas utilizadas en el desarrollo del ciclo de vida de software.

Capítulo II Desarrollo de la Aplicación web **SIGEGAS** (Sistema de Gestión de Equidad, Genero y Acción Social.), también para la documentación del desarrollo se basa en la metodología XP.

Capítulo III En este apartado se determinó el análisis de resultados con y sin sistema para determinar las ventajas de los procesos con la aplicación web como marco de resultados.

Para finalizar se determina que las aplicación SIGEGAS minimiza los tiempo en los procesos de: posada y comedor, ayuda social y capacitaciones del departamento de Equidad y Género y Acción social-CEFAS, al proporcionar una interfaz amigable e intuitiva para el usuario.

ANTECEDENTES

Con el paso del tiempo el mundo institucional y empresarial ha buscado ampliar su forma de prestar sus servicios, y es aquí donde el desarrollo de aplicaciones web va tomando fuerza día a día; sin embargo para llevar acabo estos grandes avances es necesario que el desarrollo de nuevas aplicaciones, esté basadas en estándares, que permitan la compatibilidad entre tecnologías. Para esto, se fundaron compañías encargadas de crear nuevas especificaciones o estándares como por ejemplo W3C (World Wide Web), en las cuales empresas creadoras de tecnología deben basarse.

Para contribuir con el desarrollo de nuevas Aplicaciones Web existen una variedad de lenguajes para crear sitios dinámicos, a continuación una lista de los más relevantes PYTHON, PHP, JSP, ASP, RUBY. Los cuales se caracterizan por un desarrollo literalmente puro. Aumentando notablemente tiempos de desarrollo; para optimizar nacen los famosos Framework pero sin perder la filosofía de su base.

Particularmente los Framework se basan en una estructura digital que facilita el desarrollo ágil, dependiendo del lenguaje normalmente integran componentes que simplifican la producción de Aplicaciones Web como el soporte de programa, bibliotecas, plantillas y más.

En la actualidad las Aplicaciones Web se están convirtiendo en una herramienta fundamental para la comunidad actual, ya que no son dependientes de una plataforma, tan solo requieren un intérprete que vendría a ser un navegador web y conexión a internet.

Entre las ventajas que ofrecen:

- Compatibilidad multiplataforma.
- Menos requerimientos de memoria.
- Continúa Actualización.
- Múltiples Usuarios Concurrentes.
- Ahorro de tiempo.
- Reducción de costos.

El crecimiento de las tecnologías de la información y comunicación ponen en desafío constante a las instituciones públicas y privadas, que tienen disminuir la brecha tecnológica existente, siendo parte de este cambio global que les permitirá crecer notablemente con el tiempo.

Gracias a la innovadora propuesta que se aspira crear con el presente proyecto, se alcanzara la automatización del proceso de control de servicios del departamento de Equidad, Género y Acción Social en la Coordinación De Equidad, Formación Y Acción Social De Morona Santiago (CEFAS).

FORMULACIÓN DEL PROBLEMA

Los problemas del departamento de Equidad, Género y Acción Social del CEFAS son los siguientes:

- Tiempo elevado en la recopilación de la información para la generación de reportes.
- Pérdida de tiempo en la gestión y análisis de información.
- El director no tiene acceso a la información inmediata.
- El usuario pierde mucho tiempo en acceder a un servicio concreto.
- No se puede acceder a la información desde cualquier lugar.
- Problemas en la asignación de habitaciones y alimentación.
- Lentitud en el proceso de registro de capacitación, ayudas sociales, posada y comedor.

La institución necesita un instrumento que les permita obtener una mayor eficiencia, información relacionada con los servicios que presta: Capacitaciones, Ayudas sociales y Posada y comedor.

Estos controles existen dentro de la institución, pero son realizados de forma manual, incurriendo en mayor tiempo para su obtención y distribución. Con la implantación de la aplicación web, estos controles conllevaran menos tiempo en su ejecución, además de hacer más fluida la trasmisión y acceso a la información dentro y fuera de la institución.

SISTEMATIZACIÓN DEL PROBLEMA

¿Existe procesos automatizados para control de capacitaciones, Ayudas Sociales y Posada y Comedor Popular del departamento de equidad y género?

¿Se realiza un control eficaz de los recursos materiales entregados mediante ayudas sociales a grupos de atención prioritaria?

¿Qué parte del Sistema permitirá realizar los Registro, la Gestión y los reportes de capacitaciones, Ayudas Sociales y Posada y Comedor Popular?

¿Existe un sistema para el control de Servicios del departamento de Equidad y Género en la institución CEFAS?

JUSTIFICACIÓN DEL TRABAJO DE GRADO

JUSTIFICACIÓN TEÓRICA

El adelanto tecnológico que ha crecido a través de uso oportuno de la informática está causando un impacto social notable en las instituciones públicas, para lidiar con esta brecha tecnológica muchas instituciones han visto la necesidad de automatizar sus procesos.

La aplicación web para el “Control de Servicios” en el departamento de Equidad y Género, es una solución absoluta y viable para optimizar operaciones de la institución. Con un único fin simplificar procesos manuales para un mejor trabajo dentro del departamento. Teniendo como principal objetivo automatizar el proceso de control de servicios prestados.

Para el desarrollo de la aplicación web propuesta; se empleara el framework o entorno de desarrollo web Django 1.8.3, además se aspira estudiar y comprender todo lo necesario de este Framework para desarrollar la aplicación web. Ya que este entorno fomenta el desarrollo rápido y el diseño limpio y pragmático.

Como guía de desarrollo se utilizara la metodología ágil XP (Programación Extrema), la cual permite aumentar la productividad al desarrollar el software basándose en la simplicidad

JUSTIFICACIÓN APLICATIVA

Se procederá a desarrollar una aplicación web para la automatización del control de servicios del departamento de Equidad, Género y Acción Social en el CEFAS; que contara con los siguientes módulos: capacitaciones, ayuda social y posada y comedor.

A continuación se describe las funciones que realizara el usuario en la aplicación propuesta.

Administrador.- Su cuenta le permitirá la gestión de:

- Cuentas de Usuario
- Beneficiarios

- Programas
- Reportes

Secretaria Posada.- Se encargara del registro de beneficiarios, Registro de Posada y Generar reportes de posantes.

Secretara Ayuda Social.- Se encargara del registro de Ayudas, Recursos, además puede registrar beneficiarios. Generar reportes de recursos, historial de ayudas por beneficiario.

Técnico.- Se encargara de crear una capacitación y publicar esa capacitación, además generar reportes de personas que aprobaron una determinada capacitación.

Capacitador.- Se encargara de registrar beneficiarios que no existan para inscribir a la capacitación, aprobar o reprobar a beneficiarios inscritos en una capacitación, además dar por terminado una capacitación.

Con la aplicación web optimizaremos los siguientes procesos:

- Registro manual de posantes, alimentación, beneficiarios, ayudas sociales, capacitaciones, recursos.
- Consulta de historial de ayudas sociales.
- Emisión de reportes.

Parámetros de usabilidad de la aplicación:

- Interfaz amigable para el usuario.
- Interactiva con el usuario.
- Mensajes de ayuda o advertencia.

Entre las ventajas de su implementación:

- Emitir reportes de forma rápida y oportuna como apoyo a la toma de decisiones.
- Reducción de tiempos al prestar un servicio.
- Control de seguridad de la información de los servicios prestados.
- Integridad de la información.
- Accesibilidad al sistema desde cualquier lugar con acceso a internet.

OBJETIVOS

OBJETIVO GENERAL

Desarrollar una Aplicación Web para automatizar el “Control de Servicios” del departamento de Equidad, Género Y Acción Social - CEFAS.

OBJETIVOS ESPECÍFICOS

- Medir el tiempo que se demora en los servicios el departamento de Equidad, Género, Y Acción Social.
- Estudiar y Aplicar el framework Django para el desarrollo de la aplicación web.
- Implementar la metodología de desarrollo XP en la ingeniería de la aplicación web.
- Desarrollar los módulos de Capacitaciones, Acción Social y Posada y Comedor Popular.
- Determinar las ventajas de los procesos en el departamento con la aplicación web.

CAPITULO I

1. MARCO TEÓRICO

1.1. Aplicación web

Generalmente las aplicaciones web se denominan así, porque se ejecutan o funcionan en el internet. Esto quiere decir que todos los datos o archivos en los que se trabaja son ejecutados y almacenados dentro de la web; estas son independientes de una plataforma, por lo que no necesitan instalación; tan solo conexión a internet.

1.1.1. *¿Como funcionan?*

El recurso se encuentra en la internet, por lo que todo parte de una solicitud del usuarios a una página web, que a su vez realiza una solicitud a un servidor web que se encargar de procesar la información requerida y enviar una respuesta al cliente. De forma general así es como funciona y este procesamiento se serializa en milisegundos.

1.1.2. *Ventajas*

- La compatibilidad es una ventaja importante.
- Escaso requerimiento de memoria.
- Puede soportar la concurrencia de usuarios.

1.1.3. *Desventajas*

- La compatibilidad de los navegadores con estilos de diseño.
- La seguridad de la información puede ser restringido.

1.2. Framework

Se podría considerar como un conjunto de elementos reutilizables basados en una arquitectura personalizable e universal con opción a integrar fragmentos para elaborar una aplicación específica; como ejemplo podemos considerar al framework Django basado en el lenguaje de programación Python.

1.2.1. Ventajas

- Acrecentar la productividad en el desarrollo.
- Importante organización, por que presenta un camino a seguir basado en una arquitectura por capas.
- Generalmente el framework se encarga de la seguridad básica.

1.2.2. Desventajas

- Se requiere mucho tiempo para adiestrarse, no basta con discernir el lenguaje en que fue escrito.
- Posee una arquitectura estricta a seguir.

1.3. Patrón MVC(Modelo, Vista, Controlador)

Es una arquitectura que permite ver de otra forma el desarrollo de software, permitiendo separar las partes que componen un sistema o sus bases; para una mayor organización durante el desarrollo.

1.3.1. Modelo

Aquí está todo lo relacionado con la base de datos, las clases, métodos que permitan interactuar con los datos, ya sea para almacenar, consultar o actualizar información.

1.3.2. Vista

Está muy relacionado con la interfaz de usuario, podemos decir que es lo que ve el usuario; además permite capturar información para ejecutar determinadas acciones ejecutadas por el usuario. Esta capa está ligada con el controlador y de forma indirecta con las vistas.

1.3.3. Controlador

En esta parte se encuentra la lógica de la programación, este se encarga de procesar una petición en base a una solicitud y decidir qué operación realizar, en base a su lógica definida por el desarrollador. Las reglas de negocio definidas aquí permiten procesar información de mejor manera para cumplir un proceso; está ligada con la capa Vista y el Modelo.

1.4. Metodología XP (Programación Extrema)

1.4.1. Proceso de desarrollo

1.4.1.1. Objetivos

- Aumentar básicamente la productividad en el desarrollo.
- Garantizar un software de calidad.
- Proporcionar prácticas eficaces de Ingeniería de Software en el desarrollo de proyectos.

1.4.1.2. Características

- Basada en entregas pequeñas o iteraciones funcionales.
- El cliente llega a ser un miembro más del equipo de desarrollo.
- El costo por cambios durante las etapas del ciclo de vida del software son tolerables.
- Susceptible al cambio de requerimientos funcionales.
- El grupo de trabajo es reducido pero compacto; básicamente el trabajo se desarrolla en parejas.
- La capacidad del grupo de trabajo con formación progresista.

1.4.1.3. Ciclo de Vida

Es ciclo de vida de esta metodología está representada en la figura 1-1, que a continuación se muestra.

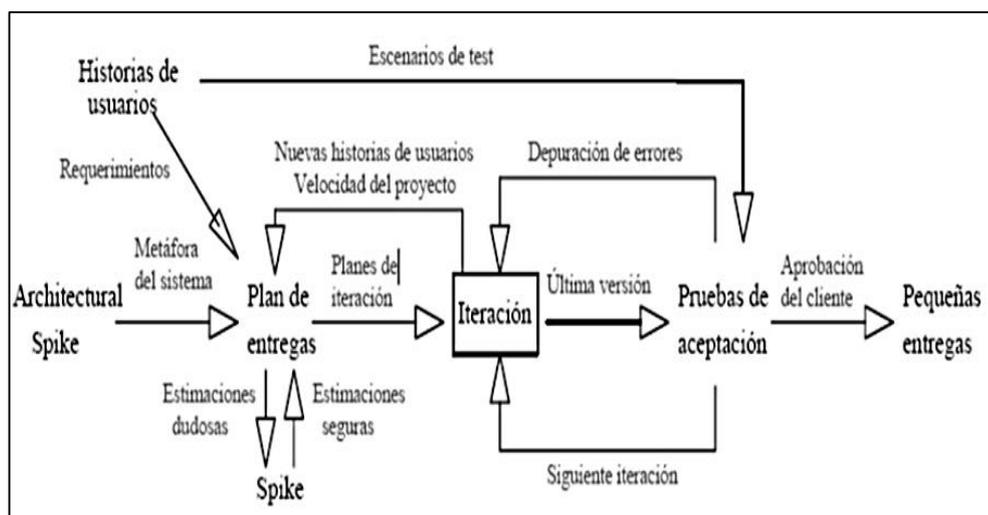


Figura 1-1 Ciclo de vida metodológica XP

Fuente: (Piskamen, 2013, <http://es.slideshare.net/Piskamen/metodologa-xp>)

1.4.1.4. Fases

- Planificación
 - Historias de Usuarios.
 - Release Planing
 - Iteraciones.
 - Velocidad del proyecto.
 - Reuniones Diarias.

- Diseño
 - Diseños simples.
 - Glosario de términos.
 - Riesgo.
 - Funcionalidad extra.
 - Tarjetas C.R.C.
- Codificación
- Pruebas.

(Castillo, Figueroa y Sevilla , <http://programacionextrema.tripod.com/fases.htm#inicio>)

1.4.1.5. Principios

- Simplicidad.
- Decisión.
- Comunicación.
- Planificación.
- Diseño simple.

(Piskamen, 2013, <http://es.slideshare.net/Piskamen/metodologa-xp>)

1.4.1.6. Ventajas

- La programación es notablemente estructurada.
- La el porcentaje de errores es mínima.
- Favorece el agrado del programador.
- Es adaptable a cambios.
- Ahorro de tiempo y dinero para cliente.

- Flexible a todo lenguaje de programación.
- Las prioridades de desarrollo las limita el cliente.
- Durante el desarrollo se realizan pruebas constantes.

1.4.1.7. Desventajas

- Recomendable para proyectos cortos.
- Una mala planificación, repercute altos costos.
- Puede considerarse complicado a diferencia de la implementación tradicional.

1.5. PostgreSQL

Es un sistema de gestión de base de datos orientado a objetos, basado en modelo relacional de código abierto, puede trabajar con distintos sistemas operativos.

1.5.1. Ventajas

- Estabilidad y confiabilidad legendarias.
- Extensible.
- Multiplataforma.
- Diseño para ambiente de alto volumen.

1.5.2. Desventajas

- En comparación con MySQL trabaja lento en inserciones y actualizaciones, por contar con cabeceras de intersección.
- La sintaxis de algunos de sus comandos tienen un cierto grado de dificultad.

<http://postgresql-dbms.blogspot.com/p/limitaciones-puntos-de-recuperacion.html>

1.6. PgAdmin 3

Herramienta para el diseño y utilización de bases de datos PostgreSQL. Compatible con PostgreSQL 7.3 en adelante y se encuentra disponible para la mayoría de plataformas. Sus funcionalidades principales son: Elaborar sencillas y complejas consultas SQL, construir engorrosas bases de datos. Su interfaz sencilla y de fácil gestión.

1.6.1. Características

- Se puede ver las tareas que se están ejecutando de manera gráficas.
- La comunicación de PgAdmin con PostgreSQL lo realiza utilizando el lenguaje SQL; esto permite ver la funcionalidad de PostgreSQL y minimizar tiempos.
- Permite exportar fácilmente resultados en diferentes formatos.
- Permite realizar backups y restauraciones de manera gráfica.

(Teba, 2013, <http://www.solvetic.com/tutoriales/article/299-postgresql-pgadmin/>)

1.7. Python

Python se puede considerar uno de los lenguajes de programación más sencillos de aprender y de fácil uso. El código de Python es claro tanto al leerlo como escribirlo, y es preciso sin ser complicado, permitiendo escribir menos líneas de código.

Además es un lenguaje de plataforma cruzada: en general, el mismo programa Python puede ejecutarse en diferentes sistemas operativos como por ejemplo Windows y Unix como Linux, BSD y MacOS X. Python es un lenguaje orientado a objetos. (Summerfield, 2009, pp. 17-18)

1.7.1. Características

- Simple.
- Libre y fuente abierta.
- Lenguaje de alto nivel.
- Interpretado.
- Orientado a objetos.

(Swaroop, CH, 2014, http://dev.laptop.org/~edsiper/byteofpython_spanish/ch01s02.html)

1.7.2. Ventajas

- La depuración desarrollada y de fácil uso.
- La flexibilidad de Python permite que se use para el desarrollo rápido de diferentes tipos de aplicaciones.

(JaGonzalez, 2016, <http://jagonzalez.org/deberias-usar-python-para-el-desarrollo-web>)

1.8. Framework DJANGO

Framework para el desarrollo web de alto nivel, lenguaje base Python, que permite un desarrollo rápido, un diseño sin complicaciones y tareas genéricas definidas. Soluciona gran parte de los problemas del desarrollo de aplicaciones web. El desarrollador no debe preocuparse por tareas comunes y así puede enfocarse en codificar su aplicación a un menor esfuerzo.

1.8.1. Patrón de diseño MTV

Django es conocido como un Framework MTV.

- **M** (Modelo), acceso a la base de datos. La información que contiene esta capa está relacionada a: Acceder a los datos, validar datos, maneja las relaciones existentes entre datos.
- **T** (Plantilla), presentación. Se encarga de ver cómo será mostrada la información al usuario final.
- **V** (Vista), lógica de negocios. Se encarga del back-end de la aplicación, enlazando los datos con la presentación a través de las URLs.

(Saul, M, 2015: p.75-76, <https://pythonizame.s3.amazonaws.com/media/Book/guia-definitiva-django-18/file/34ba425e-5985-11e5-964d-04015fb6ba01.pdf>)

1.8.2. Proyecto Django

Es un conjunto de configuraciones para una petición de Django, incluye configuraciones de base de datos, alternativas genéricas de Django y configuraciones definidas para aplicaciones.

Comando encargado de crear un proyecto django:

```
- django-admin.py startproject misitio (nombre de ejemplo)
```

Este comando crea los archivos siguientes:

- **misitio/**: Carpeta que contiene el proyecto general.
- **manage.py**: Línea de comandos que ayuda a interactuar con el proyecto Django de diferentes maneras.
- **misitio/misitio/**: Es un directorio interno que alberga el paquete Python para un proyecto.
- **__init__.py**: Archivo que permite tratar a un proyecto creado como una agrupación de módulos.

- **settings.py:** Lista de opciones para configurar un proyecto Django.
- **urls.py:** Permite declarar las urls para un proyecto.
- **wsgi.py:** Permite poner un proyecto en producción.

(Saul, M, 2015: p.19, <https://pythonizame.s3.amazonaws.com/media/Book/guia-definitiva-django-18/file/34ba425e-5985-11e5-964d-04015fb6ba01.pdf>)

1.8.3. Servidor de desarrollo

Django contiene un servidor Web liviano (Que es llamado con el comando runserver) para usar durante el desarrollo web.

Ejecutar el servidor de desarrollo Django.

```
- python manage.py runserver
```

El servidor se inicia en el puerto 8000 (el puerto es configurable), para conexiones locales.

(Saul, M, 2015: p.20, <https://pythonizame.s3.amazonaws.com/media/Book/guia-definitiva-django-18/file/34ba425e-5985-11e5-964d-04015fb6ba01.pdf>)

1.8.4. Sistema de plantilla

Una plantilla es un enlace de texto para separar los datos de la presentación del documento. La plantilla define una lógica básica, por ejemplo etiquetas de plantillas que se encargan de manejar la presentación de un documento.

```

inventario.html
1  {% extends "base1.html" %}
2  {% load staticfiles %}
3  {%block title%}Inventario{%endblock%}
4  {%block content%}
5  <h4>LISTA DE ARTICULOS</h4>
6  <table class="table table-hover" ><thead>
7      <tr><td id='cool'><b><center>Categoria</b></td>
8          <td id='cool'><b><center>Artículo</b></td>
9          <td id='cool' ><b><center>Stock</b></td></tr>
10 </thead><tbody>
11     {% for dato in datos%}
12         {%if dato.Stock == 0 %}
13             <tr onclick="muestraPR(this)">
14                 <td>{{dato.Articulo}}</td>
15                 <td style="color:red">{{dato.Stock}}</td>
16             </tr>
17             {%else%}
18                 <tr><td>{{dato.Articulo}}</td>
19                 <td >{{dato.Stock}}</td></tr>
20             {%endif%}
21         {% empty %}
22             <h4>No hay articulos que mostrar</h4>
23         {%endfor%}
24     <tr><td >{{datos|length}}</span></td></tr>
25 </tbody></table>
26 {%endblock%}

```

Figura 2-1 Plantilla inventario.htm.

Fuente: DUCHITANGA, JULIO. 2016.

Además las plantillas admiten el uso de filtros de datos para tratar la información proveniente del servidor.

(Saul, M, 2015: pp.42- 44, <https://pythonizame.s3.amazonaws.com/media/Book/guia-definitiva-django-18/file/34ba425e-5985-11e5-964d-04015fb6ba01.pdf>)

1.8.4.1. Herencia de plantillas

Es una estructura de un sitio web, alberga todas las partes comunes que tendrá un sitio web; para ello se utiliza bloques que serán sobrescritos por las plantillas hijas al heredar la base.

A continuación en la figura 3-1 un ejemplo de platilla base:

```
base.html *
1  {% load staticfiles %}
2  <!DOCTYPE html>
3  <html lang="es">
4  <head>
5  <title> {% block title %}Base{% endblock %}</title>
6  </head>
7  <body>
8  <h4> Plantilla Base</h4>
9  <section>
10 <{% block content %}{% endblock %}>
11 </section>
12 <footer>
13 <{% block footer %}
14 <p>| © Copyright:2016|Cristian Duchitanga</p>
15 <{% endblock %}>
16 </footer>
17 </body>
18 </html>
```

Figura 3-1 Plantilla base.htm.
Fuente: DUCHITANGA, JULIO. 2016.

1.8.5. Modelos

1.8.5.1. Configuración de base de datos

La configuración de la base de datos se encuentra en el archivo settings.py, se debe editar la variable DATABASES, el cual acepta los siguientes parámetros.

- ENGINE indica a django el tipo de base de datos a utilizar.
- NAME: Nombre de la base de datos.
- USER: Nombre de usuario para la conexión.
- PASSWORD: Contraseña usado para la conexión.
- HOST: Dirección donde se encuentre la base de datos.
- POST: Puerto a utilizar.

(Saul, M, 2015: pp.76-77, <https://pythonizame.s3.amazonaws.com/media/Book/guia-definitiva-django-18/file/34ba425e-5985-11e5-964d-04015fb6ba01.pdf>)

En la figura 4-1 se muestra un ejemplo de configuración para una base de datos PostgreSQL.

```
settings.py
103     DATABASES = {
104         'default': {
105             'ENGINE': 'django.db.backends.postgresql_psycopg2',
106             'NAME': 'db_Cefas_1.0',
107             'USER': 'Django_Cefas',
108             'PASSWORD': '12345678swdent@',
109             'HOST': 'localhost',
110             'PORT': '5433',
111         }
112     }
```

Figura 4-1 Configuración para PostgreSQL.

Fuente: DUCHITANGA, JULIO. 2016.

1.8.5.2. Aplicación Django

Conjunto de archivos de código fuente, que incluye vistas y modelos, que se encuentran en un paquete Python determinado. Convirtiéndose en un aplicación Django.

Crear una aplicación django:

- Python manage.py **startapp** (nombre de la aplicación)

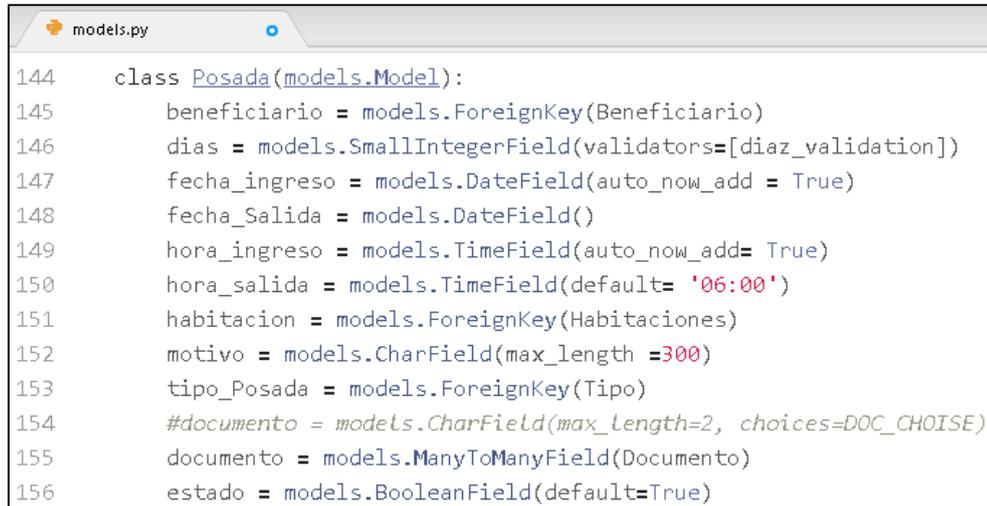
nombreakp/
__init__.py
admin.py
models.py
tests.py
views.py
migrations/
__init__.py

Esta es la estructura de una aplicación Django. Es importante mencionar que un proyecto puede albergar n aplicaciones que pueden trabajar conjuntamente como uno solo o independientemente.

(Saul, M, 2015: p.80, <https://pythonizame.s3.amazonaws.com/media/Book/guia-definitiva-django-18/file/34ba425e-5985-11e5-964d-04015fb6ba01.pdf>)

1.8.5.3. Como crear Modelos

Para crear un modelo se debe tener ya definida un modelado de base de datos; a continuación se presenta un ejemplo de cómo se crear un modelo en el fichero models.py.



```
models.py
144 class Posada(models.Model):
145     beneficiario = models.ForeignKey(Beneficiario)
146     dias = models.SmallIntegerField(validators=[diaz_validation])
147     fecha_ingreso = models.DateField(auto_now_add = True)
148     fecha_Salida = models.DateField()
149     hora_ingreso = models.TimeField(auto_now_add= True)
150     hora_salida = models.TimeField(default= '06:00')
151     habitacion = models.ForeignKey(Habitaciones)
152     motivo = models.CharField(max_length =300)
153     tipo_Posada = models.ForeignKey(Tipo)
154     #documento = models.CharField(max_length=2, choices=DOC_CHOISE)
155     documento = models.ManyToManyField(Documento)
156     estado = models.BooleanField(default=True)
```

Figura 5-1 Modelos de tablas de base de datos.

Fuente: DUCHITANGA, JULIO. 2016.

1.8.6. Sitio de administración

Esta característica que funciona leyendo un grupo de datos de los modelos definidos con el objetivo de brindar una interfaz potente y preparada para producción; para el uso de administradores. (Saul, M, 2015: p.102, <https://pythonizame.s3.amazonaws.com/media/Book/guia-definitiva-django-18/file/34ba425e-5985-11e5-964d-04015fb6ba01.pdf>)

1.8.6.1. Interfaz de administración

Diseñada para los usuarios comunes, con la información necesaria para ser lo suficientemente entendible. (Saul, M, 2015: p.104, <https://pythonizame.s3.amazonaws.com/media/Book/guia-definitiva-django-18/file/34ba425e-5985-11e5-964d-04015fb6ba01.pdf>)

A continuación se muestra en la figura 6-1 la interfaz de administración genérica de Django.



Figura 6-1 Índice principal de la administración Django.
Fuente: DUCHITANGA, JULIO. 2016.

Contiene enlaces para editar y guardar objetos, que dirigen a páginas concretas como: listas de cambio y formularios de edición de registros. La lista muestra todos los registros que contiene la base de datos. El formulario de edición permite ejecutar las siguientes acciones: modificar, registrar y eliminar registros concretos de la base de datos.

(Saul, M, 2015: p.105, <https://pythonizame.s3.amazonaws.com/media/Book/guia-definitiva-django-18/file/34ba425e-5985-11e5-964d-04015fb6ba01.pdf>)

1.8.7. Formularios usando clases

Django trae incorporado la librería `django.forms`; para el uso de formularios; para validar y presentar un documento HTML. La figura 7-1 muestra un ejemplo de formulario basada en clases.

```

forms.py
1  from django import forms
2
3  class recurso(forms.ModelForm):
4
5      class Meta:
6          model= Recurso
7          fields=['Categoria', 'Articulo', 'Detalle', 'Stock', 'Observacion']
8          widgets = {
9              'Observacion': Textarea(attrs={'cols': 2, 'rows': 3}),
10         }

```

Figura 7-1 Formulario basado en clases Django.
Fuente: DUCHITANGA, JULIO. 2016.

A continuación en la figura 8-1 un ejemplo de cómo ligar formularios a vistas.

```
views.py
137 def recurso(request):
138     if request.method == 'POST':
139         form = recurso(request.POST)
140         if form.is_valid():
141
142             Tipo_Ayuda = form.cleaned_data['Categoria']
143             Nombre = form.cleaned_data['Articulo']
144
145             Detalle = form.cleaned_data['Detalle']
146             Stock = form.cleaned_data['Stock']
147             Observacion = form.cleaned_data['Observacion']
148
149             form.save()
150             messages.success(request, 'Se registro la recurso %s'%Nombre)
151
152             return HttpResponseRedirect('/social/recurso' )
153         else:
154             messages.success(request, 'Parece que falta llenar campos')
155
156     else:
157         form = recurso()
158         cero = Recurso.objects.filter(Stock =0)
159         return render (request, 'recurso.html', {'form':form, 'cero':cero})
```

Figura 8-1 Vista recurso () para formulario.

Fuente: DUCHITANGA, JULIO. 2016.

1.8.8. URLconf

Considerado como un índice de contenidos para un sitio web. Básicamente, es un mapeo entre los patrones URL y las funciones de vista que deben ser llamadas por esos patrones URL.

Cada definición de URLconf debe asociar una vista, pasando metodos.

```
urls.py
1 from django.conf.urls import url
2
3 from . import views
4
5 urlpatterns = [
6     url(r'^recurso/', views.recur, name='recurso'),
```

Figura 9-1 Ejemplo de Urlconf Django.

Fuente: DUCHITANGA, JULIO. 2016.

Están basadas en expresiones regulares Python para especificar patrones en un texto; a continuación una selección patrones comunes.

Símbolo	Coincide con
.	Cualquier carácter
\d	Cualquier dígito
[A-Z]	Cualquier carácter, A-Z (mayúsculas)
[a-z]	Cualquier carácter, a-z (minúsculas)
[A-Za-z]	Cualquier carácter, a-z (no distingue entre mayúscula y minúscula)
+	Una o más ocurrencias de la expresión anterior (ejemplo, \d+ coincidirá con uno o más dígitos)
[^/]+	Cualquier carácter excepto la barra o diagonal.
?	Cero o una ocurrencia (ejemplo \d? coincidirá con cero o un dígito)
*	Cero o más ocurrencias de la expresión anterior (ejemplo, \d* coincidirá con cero o más dígitos)
{1,3}	Entre una y tres (inclusive) ocurrencias de la expresión anterior (ejemplo \d{1,3} coincidirá con uno, dos o tres dígitos)

Figura 10-1 Patrones comunes para Urlconf Django.

Fuente: <https://pythonizame.s3.amazonaws.com/media/Book/guia-definitiva-django-18/file/34ba425e-5985-11e5-964d-04015fb6ba01.pdf>.

1.8.9. Vistas

Las vistas de Django son las encargadas de procesar las solicitudes que provienen desde una URL; para ello se apoya de funciones, clases propias de Python. Las vistas en Django son código Python.

1.8.9.1. Vistas Genéricas basadas en clases

Agrupar patrones y estilos habituales dentro del desarrollo de vistas, abstrayéndolos; de tal forma que se pueda escribir vistas en un tiempo reducido a un menor esfuerzo, sin repetir código.

Django posee vistas genéricas para realizar las siguientes tareas:

- Redirigir a una página diferente a un usuario y renderizar una plantilla dada.
- Mostrar páginas de listado y detalle para un solo objeto.
- Listar objetos basados en fechas en páginas de archivo de tipo día/mes/año, su detalle asociado, y las páginas más recientes.
- Permitir a los usuarios registrar, modificar y borrar registros.

(Saul, M, 2015: p.219, <https://pythonizame.s3.amazonaws.com/media/Book/guia-definitiva-django-18/file/34ba425e-5985-11e5-964d-04015fb6ba01.pdf>)

1.8.10. Sesiones

El entorno de sesiones permite guardar y recuperar datos basándose en la sesión del usuario, guarda la información relevante en el servidor para mayor seguridad. (Saul, M, 2015: p.290)

1.8.11. Usuarios e identificación

Las sesiones permiten almacenar información a lo largo de las diferentes peticiones del navegador; por otra parte utiliza esas sesiones para hacer login.

El sistema de autenticación de usuarios de Django maneja cuentas de usuarios, grupos, permisos y sesiones basadas en cookies. Realiza dos verificaciones básicas a los usuarios:

- Verifica que el usuario posea una cuenta.
- Verifica los permisos del usuario.

El sistema autenticación y autorización tiene los siguientes componentes:

- **Usuarios:** Personas registradas en tu sitio web.
- **Permisos:** Permite ver o asignar permisos a un usuario.
- **Grupos:** Forma genérica para aplicar permisos a varios usuarios con funciones similares.
- **Mensajes:** Permite enviar y mostrar mensajes del sistema hacia el usuario.
- **Perfiles:** Permite incorporar más campos a los usuarios del sistema.

(Saul, M, 2015: pp.296-297, <https://pythonizame.s3.amazonaws.com/media/Book/guia-definitiva-django-18/file/34ba425e-5985-11e5-964d-04015fb6ba01.pdf>)

1.8.12. Seguridad

Django intenta mitigar esta dificultad. Está diseñado para protegerte automáticamente de muchos de los errores de seguridad comunes de los desarrolladores Web. (Saul, M, 2015, p.381, <https://pythonizame.s3.amazonaws.com/media/Book/guia-definitiva-django-18/file/34ba425e-5985-11e5-964d-04015fb6ba01.pdf>)

1.9. ReportLab

Es una librería para crear reportes en formato de documento portable (PDF), compatible con el lenguaje de programación Python y con todas sus implementaciones como por ejemplo Django. (Vilchez, 2014, <http://menteal.blogspot.com/2014/01/reportlab-libreria-pdf-para-python-00.html>)

Permite crear reportes utilizando dos librerías:

- Canvas.
- Platypus (framework de Reportlab).

1.9.1. Ventajas

- Posee un marco amplio para el diseño, porque trabaja directamente con Python y hace uso de canvas.
- Trae con consigo un framework para facilitar la codificación y el diseño (platypus).

1.9.2. Desventajas

- La curva de aprendizaje es muy amplia por lo que no proporciona un marco de trabajo acorde al desarrollo ágil.

1.10. Bootstrap

Bootstrap es un framework que permite diseñar interfaces web con CSS y Javascript que se adaptan al tamaño del dispositivo en el que se vea. La interfaz web se verá perfecta y se creará de un modo automático, ya sea en un desktop, portátil, tablet o teléfono móvil. Propone una solución perfecta para los diseños adaptables. (Kinaweb!, 2015, <http://www.kinaweb.es/que-es-bootstrap>)

1.10.1. Ventajas

- Utiliza componentes y servicios como: HTML5 shim, Normalize.css, OOCSS, jQuery UI, LESS y GitHub.
- Implementa HTML5 + CSS3
- Grid system, que por defecto incluye 12 columnas fijas, dependiendo del diseño será adaptable o no a dispositivos.

1.10.2. Desventajas

- Es necesario adaptarse a su forma de trabajo, a pesar de poseer una curva de aprendizaje liviana, se debe comprender y familiarizarse con su estructura y nomenclatura.
- Estricta adaptación al diseño grid de 12 columnas para que sea acorde a diferentes dispositivos.
- Si necesitas añadir componentes que no existen, debes hacerlos por cuenta propia utilizando CSS puro y cuidando que mantenga la coherencia en diseño y adaptabilidad.

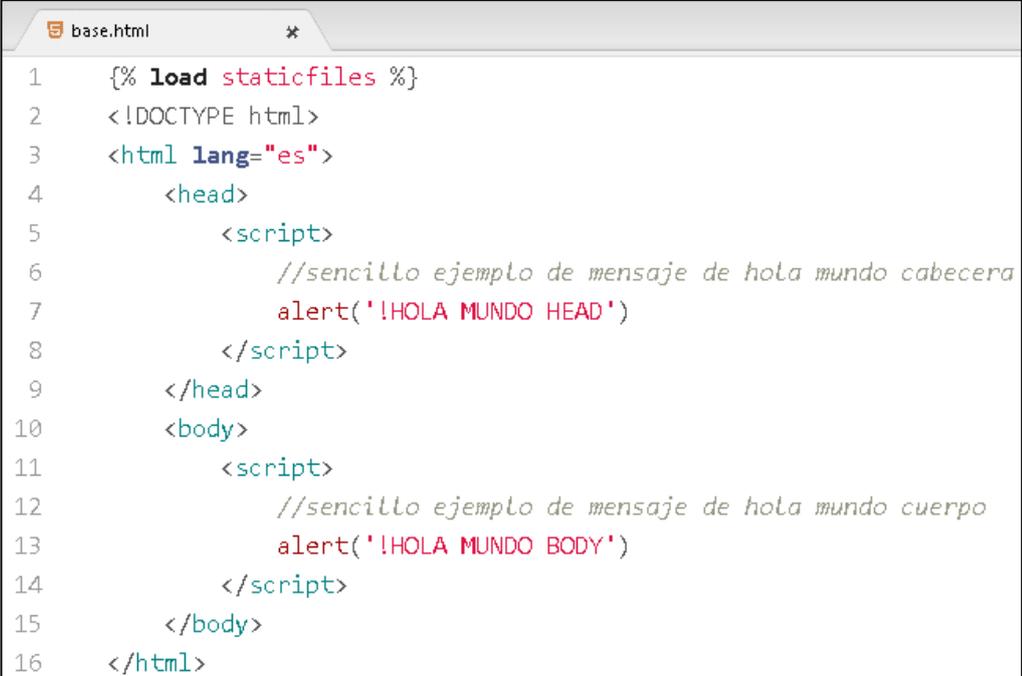
(Jorge, L., 2014, <http://jorgelessin.com/ventajas-y-desventajas-de-usar-bootstrap/>)

1.11. JavaScript

Es un lenguaje de programación que permite añadir interactividad a las páginas web. JavaScript esta contenido dentro de una página HTML o también puede residir en un archivo externo. (Zakas, et al., 2008)

1.11.1. Uso

Para utilizar JS (JavaScript) se debe usar las etiquetas HTML `<Script></Script>`, dentro ellas se puede escribir código JS. Estas etiquetas pueden residir dentro de las etiquetas head o body del HTML como menciona (Zakas, et al., 2008, pp. 23-30). A continuación un ejemplo de uso de JS.



```
1  {% load staticfiles %}
2  <!DOCTYPE html>
3  <html lang="es">
4      <head>
5          <script>
6              //sencillo ejemplo de mensaje de hola mundo cabecera
7              alert('!HOLA MUNDO HEAD')
8          </script>
9      </head>
10     <body>
11         <script>
12             //sencillo ejemplo de mensaje de hola mundo cuerpo
13             alert('!HOLA MUNDO BODY')
14         </script>
15     </body>
16 </html>
```

Figura 11-1 Uso de JavaScript.

Fuente: DUCHITANGA, JULIO. 2016

1.11.2. Ventajas

- Es extraordinario para implementar validaciones y hacer cálculos que no requieran del lado del servidor (back-end).
- Permite crear magníficos efectos de diseño que permiten al sitio ser amigable.
- Posee compatibilidad total con la mayoría de los navegadores modernos.

1.11.3. Desventajas

- Puede ser desactivado del navegador por el cliente.
- Carece de opciones 3D.
- Tiene que descargarse totalmente para ejecutar sus funciones.
- Código visible que puede ser leído o copiado por terceras personas.

1.12. Ajax

Ajax se considera como una forma de crear aplicaciones web interactivas. Permite una reacción dinámica a una petición de un usuario reduciendo el tiempo de espera, para conseguir esto Ajax carga solo las partes solicitadas de un sitio web evitando así recargar todo el sitio, obteniendo una mayor fluidez al responder una solicitud.

Ajax es la abreviatura de Asynchronous JavaScript and XML, Ajax es simplemente una pequeña parte de JavaScript.

(Zakas., 2008. pp. 23-30)

1.12.1. Ventajas

- La mayor parte de procesamiento de la aplicación tiene lugar en el navegador del usuario.
- Las solicitudes al servidor son relativamente cortas, evitando recargar todo un sitio.

1.12.2. Desventajas

- Normalmente requieren más tiempo para su desarrollo.
- Su implementación solo debe realizarse cuando el desarrollo tradicional no satisfaga la experiencia del usuario.
- Cambia la forma de interactuar con el sitio web.
- Estrictamente necesario JavaScript para su funcionamiento óptimo.

CAPITULO II

DESARROLLO DE LA APLICACIÓN WEB SIGEGAS PARA AUTOMATIZAR EL “CONTROL DE SERVICIOS” DEL DEPARTAMENTO DE EQUIDAD, GÉNERO Y ACCIÓN SOCIAL-CEFAS.

2. MARCO METODOLÓGICO

En este capítulo se describe el proceso de desarrollo de la aplicación web propuesta. La aplicación automatiza los procesos manuales efectuados por los usuarios del departamento de Equidad, Género y Acción Social. Las personas que ejecutan estos procesos son: responsable del centro intercultural (Posada y Comedor), trabajadora social encargada de acción social (Ayudas Sociales), capacitadores y asistentes administrativos (técnicos) encargados de Capacitaciones.

Para tener claro el camino a seguir durante el ciclo de vida de desarrollo de la aplicación web, se utilizó la metodología XP (Programación Extrema), que permite seguir un proceso basado en fases de desarrollo las cuales constan de: planificación, diseño, codificación y pruebas, además de esto se obtuvo una aplicación web enmarcada en la calidad. Se debe tomar en cuenta que esta metodología está orientada a grupos de trabajo, pero para esta implementación se modificó para trabajar de forma individual.

En el estudio del framework Django; después de hacer un análisis se determinó las principales ventajas y desventajas de utilizar el framework Django; a continuación se detalla en la tabla 1-2:

Tabla 1- 2: Ventajas y Desventajas de Django.

VENTAJAS	DESVENTAJAS
<ul style="list-style-type: none">• Fomenta un desarrollo rápido, minimizando procesos de desarrollo.• Utiliza la regla de no te repitas; consiste en no repetir bloques de código y así fomentar la reutilización.• Posee un ORM potente para hacer las consultas más simples y evitar usar SQL puro.• Una ventaja en la que Django sobresale es poseer un sitio de administración activo.• Fue construido bajo la necesidad de ahorrar tiempo, por lo que se considera un framework para desarrollo rápido de aplicaciones.• Esta escrito bajo el lenguaje de programación Python, por lo que hereda la facilidad de escribir y entender el código.• Posee soporte amplio de librerías para agilizar el desarrollo.• Compatibilidad con bases de datos actuales.	<ul style="list-style-type: none">• Poner en producción una aplicación web construida en Django puede llegar hacer tedioso.• Estéticamente el manejo de llaves foráneas no es de lo más sofisticado.• El ORM no satisface al 100% las consultas complejas; lo que conlleva a utilizar SQL puro.

<ul style="list-style-type: none"> • Permite construir consultas SQL sin utilizar el ORM. • Incorpora un servidor de desarrollo propio. • La documentación oficial es consistente. • La seguridad es manejada por el propio framework, garantizando la fiabilidad de los datos. • Genera una estructura CRUD (Crear, leer, modificar y eliminar) automáticamente sobre el sitio de administración. • El desarrollo está ligado al patrón de desarrollo MTV (Modelo Vista Plantillas); permitiendo tener una estructura flexible. 	
--	--

Fuente: DUCHITANGA, JULIO. 2016.

Para el desarrollo de la aplicación web se aplicó el framework Django por las amplias ventajas que proporciona para el desarrollo web. Se integró a Django el sistema de gestión de base de datos PostgreSQL, para el almacenamiento de los datos siguiendo el modelo relacional; aplicado al sistema de gestión de equidad género y acción social (SIGEGAS).

2.1. Metodología XP (programación extrema)

Para la etapa de ingeniería de la aplicación web se implementó la metodología XP, por ser una metodología basada en el desarrollo ágil, además que permite una comunicación permanente con el cliente y el equipo de desarrollo, también se adapta a los cambios de requerimientos porque maneja iteraciones, permitiendo simplificar el desarrollo.

La figura 12-2 muestra la arquitectura de la metodología XP.



Figura 12-2 Arquitectura Metodología XP (Programación Extrema)

Fuente: DUCHITANGA, JULIO. 2016

2.2. Fase de Planificación

2.2.1. Planificación Inicial

En esta etapa denominada planificación inicial se realiza el estudio y análisis de problema que presenta el proyecto, permitiendo la recopilación de requerimientos funcionales para la aplicación web utilizando las historias de usuarios que permiten describir un requerimiento funcional, cabe recalcar que las historias de usuarios fueron mejorando según se avanzaba en la etapa de planificación.

A continuación en la tabla 2-2 se especifica el equipo de trabajo para el desarrollo de la aplicación web SIGEGAS.

Tabla 2-2: Roles del Equipo de Trabajo

Metodología XP		
Nombre de miembro	Tipo	Roles XP
Julio Duchitanga C.	Tesista	Programador, Tester, Tracker
Ing. Lorena Aguirre	Director	Entrenador
Ing. Eduardo Villa	Miembro	Entrenador

Fuente: DUCHITANGA, JULIO. 2016

2.3.1.1. Recopilación de información

La importancia de este proceso dentro de proyecto radica en especificar las funcionalidades de la aplicación web, las cuales se obtuvieron utilizando mecanismos como: entrevistas, recopilación de documentos e información verbal del cliente; que permitieron definir con claridad como empieza y termina un proceso manual.

Se realizaron entrevistas a los encargados de las diferentes áreas del departamento de Equidad, Género y Acción Social, como son: responsable del centro intercultural, la trabajadora social encargada de Ayudas Sociales, capacitadores y Asistente Administrativo encargados de capacitaciones.

La información obtenida permitió elaborar las historias de usuarios con mayor veracidad cumpliendo así la recopilación de requerimientos para la elaboración de cada uno de los módulos de la aplicación web SIGEGAS.

2.2.2. Arquitectura de la aplicación web

A continuación en la figura 13-2 se muestra la arquitectura de la aplicación web.

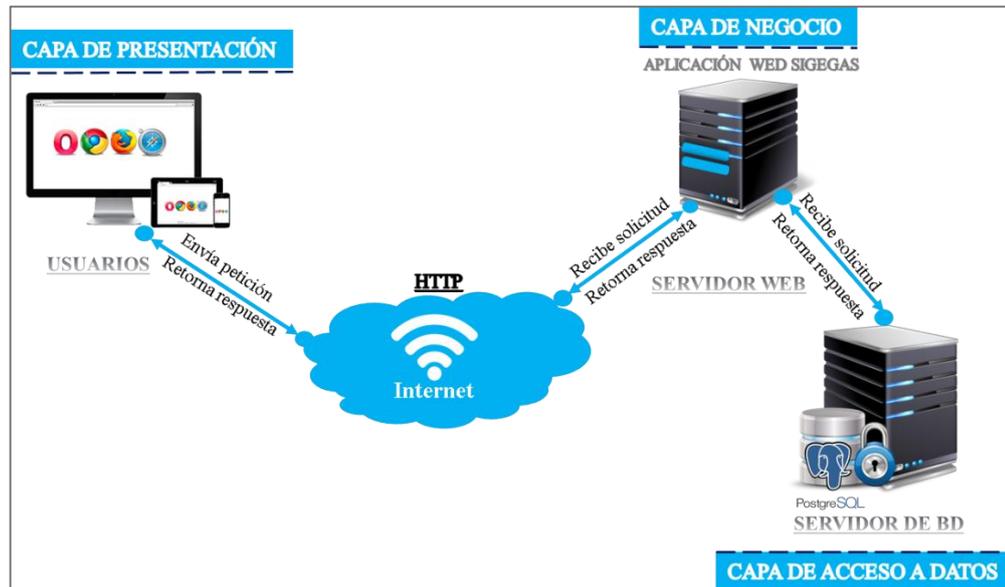


Figura 13-2 Arquitectura de la aplicación web.

Fuente: DUCHITANGA, J., 2016

2.2.3. Módulos de la aplicación web SIGEGAS



Figura 14-2 Módulos de la aplicación web SIGEGAS.

Fuente: DUCHITANGA, J., 2016

- Posada y Comedor

Este módulo se encargará del área denominado centro intercultural del departamento de Equidad Género y Acción Social; el cual efectúa la automatización de: gestión de beneficiarios, gestión de posadas, gestión de alimentación en lo que compete a comedor, y generar fichas al gestionar una posada.

- Ayuda social

Este módulo permite llevar un registro cronológico de ayudas brindadas a personas de escasos recursos económicos, permitiendo tener un historial de ayudas brindadas, por persona, además llevar un registro de recursos entregados y la disponibilidad de los mismos.

- Capacitaciones

Permite llevar un registro de personas por capacitación, registro de capacitaciones desarrolladas, aprobar o reprobar personas de una determinada capacitación y generar certificados para personas que aprueben de acuerdo al criterio del capacitador; para ello utiliza dos tipos de usuarios: capacitador y el técnico o asistente administrativo.

Además de estos módulos principales, se considera un módulo complementario para generar reportes. Entre otras funciones: asignar roles a usuarios de acuerdo a un grupo predefinido en la aplicación web, mediante autenticación, con sus respectivas sesiones para los usuarios de la aplicación.

2.2.4. Cronograma de actividades

De acuerdo a las historias de usuario adquiridas mediante entrevistas con los encargados de las tres áreas del departamento de equidad género y acción social como son: Capacitaciones, posada y comedor y acción social (Ayuda social), se elaboró una lista de historias, ordenadas en base a tres módulos como son: posada (1), ayuda social (2) y capacitaciones (3), generando un cronograma de actividades; repartidas en tres iteraciones a realizarse.

A continuación en la figura 15-2 se detalla el cronograma de actividades.

Nombre de tarea	Duración	Comienzo	Fin
▲ Cronograma de actividades	81,1 días	vie 01/04/16	lun 04/07/16
Inicio	0 días	vie 01/04/16	vie 01/04/16
▲ Primera iteración	43 días	vie 01/04/16	vie 20/05/16
▲ Historias de Usuario (M. Posada)	43 días	vie 01/04/16	vie 20/05/16
HU1 Acceso al sistema	18 días	vie 01/04/16	jue 21/04/16
HU2 Gestión de usuarios	4 días	sáb 23/04/16	mié 27/04/16
HU3 Gestión de Beneficiarios	4 días	jue 28/04/16	lun 02/05/16
HU4 Asignar usuarios a un grupo	4 días	mar 03/05/16	vie 06/05/16
HU5 Gestión de Posada	10 días	vie 06/05/16	mar 17/05/16
HU6 Seguimiento de entrada y salida de pos	3 días	mié 18/05/16	vie 20/05/16
▲ Segunda iteración	16 días	sáb 21/05/16	mié 08/06/16
▲ Historias de Usuario (M. Posada)	4 días	sáb 21/05/16	mié 25/05/16
HU7 Gestión de alimentación diaria	4 días	sáb 21/05/16	mié 25/05/16
▲ Historias de Usuario (M. Ayuda social)	12 días	jue 26/05/16	mié 08/06/16
HU8 Gestión de ayudas sociales	6 días	jue 26/05/16	mié 01/06/16
HU9 Gestión de bienes y servicios	6 días	jue 02/06/16	mié 08/06/16
▲ Tercera iteración	22 días	jue 09/06/16	lun 04/07/16
▲ Historias de Usuario (M. Capacitaciones)	12 días	jue 09/06/16	mié 22/06/16
HU10 Gestión de Capacitaciones	6 días	jue 09/06/16	mié 15/06/16
HU11 Inscripciones a capacitaciones	6 días	jue 16/06/16	mié 22/06/16
▲ Historia de Usuario (Reportes)	10 días	jue 23/06/16	lun 04/07/16
HU12 Generar Reporte	10 días	jue 23/06/16	lun 04/07/16
Fin	0 días	lun 04/07/16	lun 04/07/16

Figura 15-2 Cronograma de actividades.

Fuente: DUCHITANGA, JULIO. 2016

2.2.5. Planificación de Publicaciones

2.3.5.1. Iteraciones

Primera Iteración: A continuación se detalla las historias de usuario que son parte de esta iteración con su respectivo tiempo, como se muestra en la Tabla 3-2.

Tabla 3-2: Historias de Usuario – Primera Iteración

N°	HISTORIA DE USUARIO	FECHA	SEMANAS
HU1	Acceso al sistema	vie 01/04/16 - jue 21/04/16	3
HU2	Gestión de usuarios	sáb 23/04/16 - mie 27/04/16	0.5
HU3	Gestión de beneficiarios	jue 28/04/16 - lun 02/05/16	0.5
HU4	Asignar usuarios a un grupo	mar 03/05/16 - vie 06/05/16	0.5
HU5	Gestión de Posadas	vie 06/05/16 - mar 17/05/16	1,5
HU6	Seguimiento de entrada y salida de posantes.	mie 18/05/16 - vie 20/05/16	0.5

Fuente: DUCHITANGA, JULIO. 2016

A continuación en la figura 16-2 se presenta la planificación de las historias de usuario que pertenecen a la primera iteración.

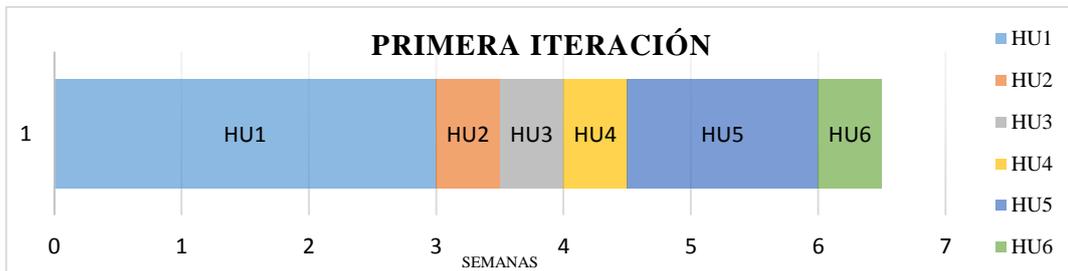


Figura 16-2 Tiempos de entrega de la primera iteración.

Fuente: DUCHITANGA, JULIO. 2016

Segunda Iteración: A continuación se describe las historias de usuario que son parte de esta iteración con su respectivo tiempo, como se muestra en la Tabla 4-2.

Tabla 4-2: Historias de Usuario – Segunda Iteración

N°	HISTORIA DE USUARIO	FECHA	SEMANAS
HU7	Gestión de alimentación diaria	sáb 21/05/16 - mié 25/05/16	0.5
HU8	Gestión de ayudas sociales.	jue 26/05/16 - mié 01/06/16	1
HU9	Gestión de recursos	jue 02/06/16 - mié 08/06/16	1

Fuente: DUCHITANGA, JULIO. 2016

A continuación en la figura 17-2 se presenta se presenta la planificación de las historias de usuario que pertenecen a la segunda iteración.

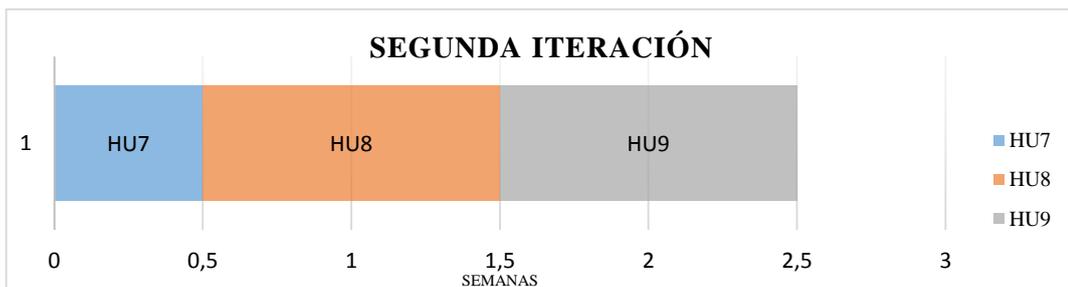


Figura 17-2 Tiempos de entrega de la segunda iteración.

Fuente: DUCHITANGA, JULIO. 2016

Tercera Iteración: A continuación se describe las historias de usuario que son parte de esta iteración con su respectivo tiempo, como se muestra en la Tabla 5-2.

Tabla 5-2: Historias de Usuario – Tercera Iteración

N°	HISTORIA DE USUARIO	FECHA	SEMANAS
HU10	Gestión de Capacitaciones	jue 09/06/16 - mié 15/06/16	1
HU11	Inscripciones a capacitaciones	jue 16/06/16 - mié 22/06/16	1
HU12	Generar Reportes	jue 23/06/16 - lun 04/07/16	1,5

Fuente: DUCHITANGA, JULIO. 2016

A continuación en la figura 18-2 se presenta se presenta la planificación de las historias de usuario que pertenecen a la tercera iteración.

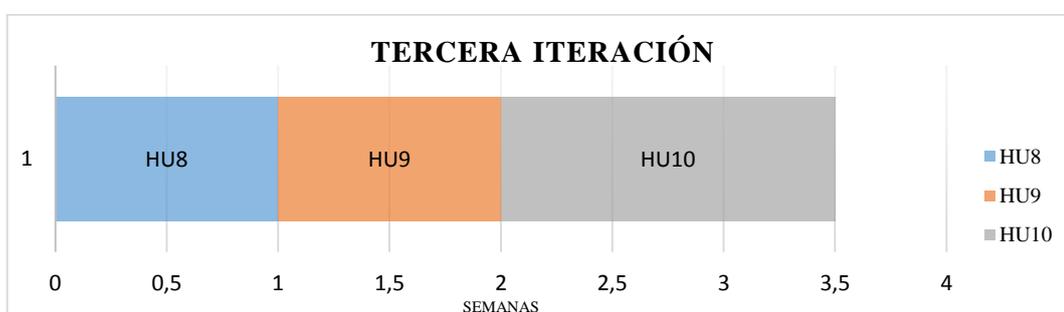


Figura 18-2 Tiempos de entrega de la segunda iteración.

Fuente: DUCHITANGA, JULIO. 2016

A continuación se describe las historias de usuario que pertenecen a cada iteración como se denotó anteriormente.

2.2.6. Historias de usuario

A continuación se muestra las historias del usuario por iteración; se optado por una valoración de Prioridad, Riesgo, Esfuerzo; para elaborar las historias de usuarios con valores de representación de Alto, Medio y Bajo.

Historias de Usuario de la primera iteración:

Tabla 6-2: Acceso al sistema

Historia de Usuario	
Numero: 1	Usuario: Programador
Nombre de Historia: Acceso al sistema	
Prioridad en negocio: Alto	Riesgo en desarrollo: Alto
Esfuerzo: Alto	Iteración: 1
Programador Responsable: Julio Duchitanga	

Descripción: Para tener acceso al sistema el usuario deberá autenticarse, para ello se presentara un formulario de login donde se llenará sus datos (Usuario y Contraseña).
Observaciones: El usuario debe estar registrado en el sistema.

Fuente: DUCHITANGA, JULIO. 2016

Tabla 7-2: Historias de Usuario Gestión de usuario

Historia de Usuario	
Numero: 2	Usuario: Programador
Nombre de Historia: Gestión de usuario	
Prioridad en negocio: Alto	Riesgo en desarrollo: Bajo
Esfuerzo: Medio	Iteración: 1
Programador Responsable: Julio Duchitanga	
Descripción: La aplicación permitirá el registro y edición de usuarios para que puedan acceder a un servicio determinado.	
Observaciones:	

Fuente: DUCHITANGA, JULIO. 2016

Tabla 8-2: Historia de Usuario Gestión de beneficiario

Historia de Usuario	
Numero: 3	Usuario: Programador
Nombre de Historia: Gestión de beneficiario	
Prioridad en negocio: Alto	Riesgo en desarrollo: Alto
Esfuerzo: Medio	Iteración: 1
Programador Responsable: Julio Duchitanga	
Descripción: La aplicación permitirá el registro y edición de beneficiarios para acceder a cualquiera de los tres módulos del sistema (Posada y Comedor, Ayudas Sociales, Capacitaciones)	
Observaciones:	

Fuente: DUCHITANGA, JULIO. 2016

Tabla 9-2: Historias de Usuario Asignar usuarios a un grupo

Historia de Usuario	
Numero: 4	Usuario: Programador
Nombre de Historia: Asignar usuarios a un grupo	
Prioridad en negocio: Alto	Riesgo en desarrollo: Medio
Esfuerzo: Medio	Iteración: 1
Programador Responsable: Julio Duchitanga	
Descripción: La aplicación permitirá asignar a usuarios del sistema a un determinado grupo predefinido en para el acceso al sistema, el grupo está definido de la siguiente manera: <ul style="list-style-type: none"> - Social para el acceso al módulo de ayuda social. - Posada para el acceso al módulo de posada y comedor. - Capacitador para el acceso de capacitadores al módulo de capacitaciones - Técnico para el acceso del técnico o administrador a módulo de capacitaciones. 	
Observaciones: Asignar usuarios a un grupo es obligatorio caso contrario no podrá acceder a su cuenta de usuario a pesar de haber sido registrado en el sistema.	

Fuente: DUCHITANGA, JULIO. 2016

Tabla 10-2: Historias de Usuario Gestión de Posada

Historia de Usuario	
Numero: 5	Usuario: Secretaria de posada
Nombre de Historia: Gestión de Posadas	
Prioridad en negocio: Medio	Riesgo en desarrollo: Medio
Esfuerzo: Medio	Iteración: 1
Programador Responsable: Julio Duchitanga	
Descripción: Se llevará un registro de posadas (Crear Fichas) para en seguimientos de estadías con los respectivos controles, como por ejemplo como que tipo de posante.	
Observaciones:	

Fuente: DUCHITANGA, JULIO. 2016

Tabla 11-2: Historias de Usuario Seguimiento de entrada y salida de Posantes

Historia de Usuario	
Numero: 6	Usuario: Secretaria de posada
Nombre de Historia: Seguimiento de entrada y salida de posantes	
Prioridad en negocio: Medio	Riesgo en desarrollo: Medio
Esfuerzo: Bajo	Iteración: 1
Programador Responsable: Julio Duchitanga	
Descripción: Permitirá ver que personas están y cuánto tiempo les queda como posantes, además se podrá marcar la salida del posante.	
Observaciones:	

Fuente: DUCHITANGA, JULIO. 2016

Historias de Usuario de la segunda iteración:**Tabla 12-2:** Historias de Usuario Gestión de alimentación diaria

Historia de Usuario	
Numero: 7	Usuario: Secretaria de Comedor
Nombre de Historia: Gestión de alimentación diaria	
Prioridad en negocio: Bajo	Riesgo en desarrollo: Bajo
Esfuerzo: Bajo	Iteración: 1
Programador Responsable: Julio Duchitanga	
Descripción: Permitir llevar el registro de personas que accedieron a un almuerzo en un determinado día, viendo el tipo de usuario (por ejemplo funcionario, posante, ocasionales).	
Observaciones:	

Fuente: DUCHITANGA, JULIO. 2016

Tabla 13-2: Historias de Usuario Gestión de ayudas sociales

Historia de Usuario	
Numero: 8	Usuario: Trabajadora Social
Nombre de Historia: Gestión de ayudas sociales	
Prioridad en negocio: Alto	Riesgo en desarrollo: Medio
Esfuerzo: Medio	Iteración: 2
Programador Responsable: Julio Duchitanga	
Descripción: Se llevara un registro de los usuarios que accedieron a una ayuda social de un determinada comunidad y que tipo de ayuda recibió, con las respectivas restricciones.	
Observaciones: Como prerrequisito el beneficiario debe ser registrado con anterioridad, además deben existir recursos en stock para extender una ayuda.	

Fuente: DUCHITANGA, JULIO. 2016

Tabla 14-2: Historias de Usuario Gestión de recursos

Historia de Usuario	
Numero: 9	Usuario: Trabajadora Social
Nombre de Historia: Gestión de recursos	
Prioridad en negocio: Medio	Riesgo en desarrollo: Medio
Esfuerzo: Bajo	Iteración: 2
Programador Responsable: Julio Duchitanga	
Descripción: Gestionar recursos que se entregan en ayudas sociales, permitiendo consultar si existe un bien para ser entregado.	
Observaciones: No se llevara un control sofisticado para este ámbito	

Fuente: DUCHITANGA, JULIO. 2016

Historias de Usuario de la tercera iteración:**Tabla 15-2:** Historias de Usuario Gestión de capacitaciones

Historia de Usuario	
Numero: 10	Usuario: Asistente administrativo
Nombre de Historia: Gestión de Capacitaciones	
Prioridad en negocio: Alto	Riesgo en desarrollo: Medio
Esfuerzo: Medio	Iteración: 3
Programador Responsable: Julio Duchitanga	
Descripción: Registrar capacitaciones que serán impartidas en una determinada comunidad. Además deberá permitir generar un certificado al término de una capacitación, para ellos recibirá la información del capacitador al marcar como terminada la capacitación.	
Observaciones:	

Fuente: DUCHITANGA, JULIO. 2016

Tabla 16-2: Historias de Usuario Inscripción a capacitaciones

Historia de Usuario	
Numero:11	Usuario: Capacitador
Nombre de Historia: inscripción a capacitaciones	
Prioridad en negocio: Medio	Riesgo en desarrollo: Medio
Esfuerzo: Medio	Iteración: 3
Programador Responsable: Julio Duchitanga	
Descripción: Los usuarios podrán inscribirse a una capacitación que se encuentre vigente. Las personas inscritas una vez finalizado el taller o capacitación deben ser calificados con aprobado o reprobado según el criterio del capacitador. Además deberá finalizar la capacitación para enviar la información al técnico para que se genere los certificados.	
Observaciones:	

Fuente: DUCHITANGA, JULIO. 2016

Tabla 17-2: Historia de Usuario Generar Reportes

Historia de Usuario	
Numero:12	Usuario: Directivos
Nombre de Historia: Generar reportes	
Prioridad en negocio: Alto	Riesgo en desarrollo: Medio
Esfuerzo: Medio	Iteración: 3
Programador Responsable: Julio Duchitanga	
Descripción: La aplicación permitirá a los encargados de cada sección del departamento de Equidad, Genero y acción social generar reportes de acuerdo a sus necesidades principales.	
Observaciones:	

Fuente: DUCHITANGA, JULIO. 2016

2.3. Fase de Diseño

2.3.1. Diseño de base de datos

El diseño de base de datos se considera como el pilar fundamental para el funcionamiento de la aplicación web. Para lo cual se elaboró un esquema de base de datos utilizando DeZing for Databases, para el desarrollo de los módulos de posada y comedor, ayudas sociales y capacitaciones en la entidad pública CEFAS como caso práctico. A continuación en la figura 19-2 se muestra el esquema general de BD (Base de Datos).

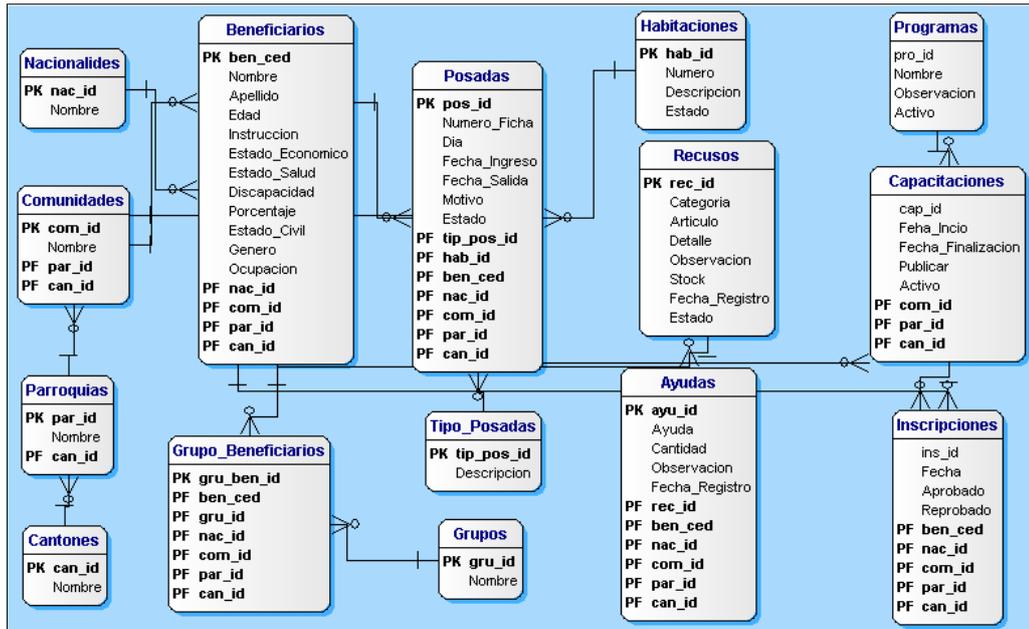


Figura 19-2 Esquema de base de datos.

Fuente: DUCHITANGA, JULIO. 2016

A continuación en la figura 20-2 se muestra las tablas necesaria que proporciona el framework Django de forma predeterminada, durante la instalación del mismo. Estas tablas se encargan de controlar el sistema de administración de Django; además se implementan para el control sesiones, usuarios, grupos, permisos. Para el manejo de autenticación de usuarios en la aplicación.

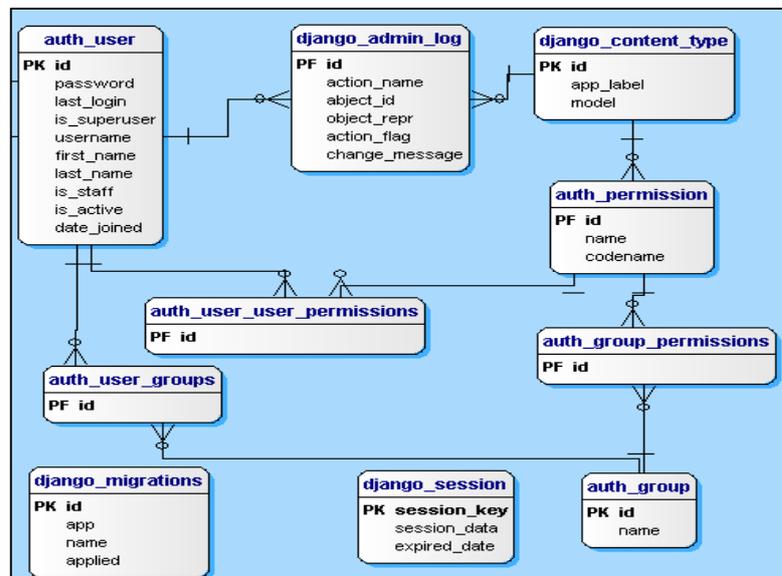


Figura 20-2 Esquema de base de datos Django.

Fuente: DUCHITANGA, J., 2016

2.3.2. Diseño de Interfaces

Dentro de este apartado se realizó el diseño interfaces de usuario, que son un componente importante dentro de cualquier sistema informático y mucho más para la implementación de la aplicación web SIGEGAS. Su función principal radica en permitir interactuar a los usuarios con la aplicación. El diseño de interfaces permite tener una visión clara de las plantillas a utilizadas y la estructura de la página, el diseño ayuda a mantener un estándar de diseño para toda la aplicación relacionada al front-end.

A continuación en la figura 21-2 se muestra la interfaz que se utilizó como base para todas las plantillas de la aplicación.



Figura 21-2 Diseño de interfaz base.

Fuente: DUCHITANGA, J., 2016

A continuación en la figura 22-2 se muestra la interfaz que será como guía para el desarrollo de gestión de posada.

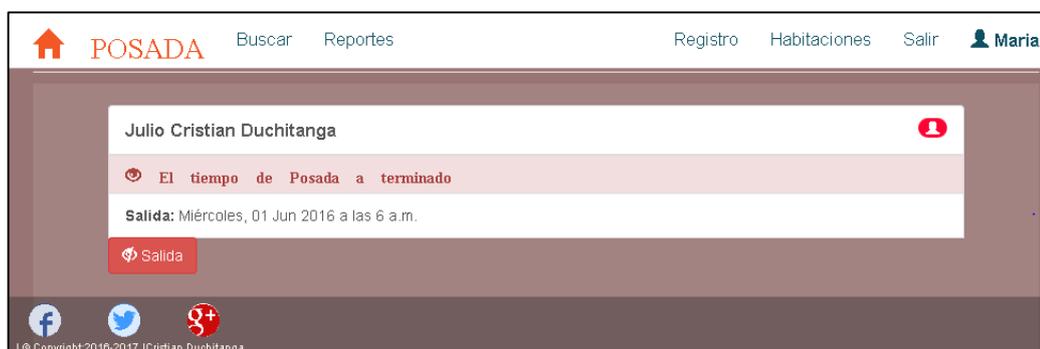


Figura 22-2 Gestión de posada.

Fuente: DUCHITANGA, J., 2016

A continuación en la figura 23-2 se muestra la interfaz guía para el desarrollo de plantillas gestión de posada.

HOSPEDAJE

Beneficiario: Julio Cristian Duchitanga

Tipo Posada:

Dias:

Hora salida: 06:00

Habitación:

Motivo:

Documento:

Guardar

Figura 23-2 Ingreso a posada.
Fuente: DUCHITANGA, J., 2016

A continuación en la figura 24-2 se muestra la interfaz guía para el desarrollo de plantilla para el seguimiento de entrada y salida de posantes.

Julio Cristian Duchitanga

El tiempo de Posada a terminado

Salida: Miércoles, 01 Jun 2016 a las 6 a.m.

Salida

Figura 24-2 Seguimiento de entrada y salida de posante.
Fuente: DUCHITANGA, J., 2016

A continuación en la figura 25-2 se muestra la interfaz guía para el desarrollo de plantilla para la gestión de habitaciones.

Agregar habitaciones

Por favor, corrija los siguientes errores.

Habitacion: Este campo es obligatorio.

Numero: Este campo es obligatorio.

Ubicacion: Este campo es obligatorio.

Estado

Guardar y agregar otro Guardar y continuar editando GUARDAR

Figura 25-2 Gestión de habitaciones.
Fuente: DUCHITANGA, J., 2016

A continuación en la figura 26-2 se muestra la interfaz guía para el desarrollo de plantilla para el registro de beneficiarios, esta interfaz sirve para los tres módulos desarrollados: posada y comedor, ayudas sociales y capacitaciones.

Figura 26-2 Registro de beneficiarios.
Fuente: DUCHITANGA, J., 2016

A continuación en la figura 27-2 se muestra la interfaz guía para la implementación del registro de alimentación tanto para beneficiarios ocasionales, permanentes y posantes.

Figura 27-2 Reg. De alimentación (beneficiarios).
Fuente: DUCHITANGA, J., 2016

A continuación en la figura 28-2 se muestra la interfaz guía para la implementación del registro de alimentación para funcionarios del CEFAS.

COMEDOR

Funcionario

Almuerzo

Costo
 .00 \$

Pagado

Observacion

Guardar y agregar otro Guardar

Figura 28-2 Registro de alimentación (funcionarios).
 Fuente: DUCHITANGA, J., 2016

A continuación en la figura 29-2 se muestra la interfaz guía para mostrar la lista de deudas de alimentación de funcionarios, además permitirá marcar la deuda como cancelada.

Fecha	Alimentación	Deuda	Acción
Miércoles, 15 Jun 2016	COMPLETO	2.00	\$
Viernes, 17 Jun 2016	COMPLETO	2.00	\$
TOTAL		4.00 \$	

Figura 29-2 Lista de deuda alimentación (funcionario).
 Fuente: DUCHITANGA, J., 2016

A continuación en la figura 30-2 se muestra la interfaz guía para el registro de ayudas sociales para un determinado beneficiario.

Ayuda Social

Beneficiario *

Ayuda *

Cantida *

Observacion

Guardar Cerrar

Figura 30-2 Registro de Ayudas Sociales.
Fuente: DUCHITANGA, J., 2016

A continuación en la figura 31-2 se muestra la interfaz guía para el registro de recursos que son utilizados en ayudas sociales.

RECURSO

Categoria

Artículo

Detalle
Ninguna

Stock

Observacion

Guardar

Figura 31-2 Registro de recursos.
Fuente: DUCHITANGA, J., 2016

A continuación en la figura 32-2 se muestra la interfaz guía para listar los recursos que se encuentran en stock cero y para aumentar el mismo si fuera el caso.

LISTA DE ARTICULOS PARA AYUDAS SOCIALES					
Categoría	Artículo	Tipo	Stock	Observación	Fecha Registro
Ayuda Tecnica	Silla postural	Para Niños	0	Solo niños mayode 3 años	29 de Mayo de 2016 a las 00:47
Ayuda social	NUEVO	Ninguna	0	Verificar	15 de Junio de 2016 a las 03:00
			N° registros 2		

Figura 32-2 Lista de recursos con stock cero.
Fuente: DUCHITANGA, J., 2016

A continuación en la figura 33-2 se muestra la interfaz guía para aumentar el stock de un recurso.

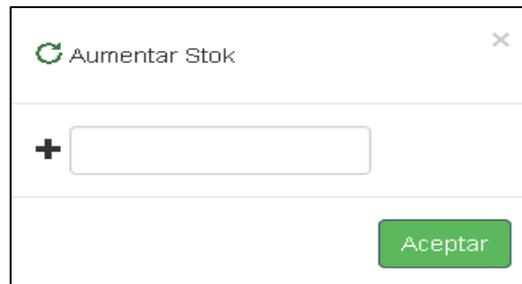


Figura 33-2 Aumentar Stock.
Fuente: DUCHITANGA, J., 2016

A continuación en la figura 34-2 se muestra la interfaz guía para listar el historial de ayudas recibidas por un beneficiario en concreto.



TIPO	ENTREGADO	CANTIDAD	OBSERVACIÓN	FECHA
Ayuda Tecnica	Silla postural Para Niños	1	Para un año	miércoles, 22 jun 2016

Figura 34-2 Historial de ayudas por beneficiario.
Fuente: DUCHITANGA, J., 2016

A continuación en la figura 35-2 se muestra la interfaz guía para el registro de capacitaciones a efectuarse en una fecha determinada.



Figura 35-2 Registro de capacitación.
Fuente: DUCHITANGA, J., 2016

A continuación en la figura 36-2 se muestra la interfaz guía para el registro de programas.



Figura 36-2 Registro de programas.
Fuente: DUCHITANGA, J., 2016

A continuación en la figura 37-2 se muestra la interfaz guía para listar las capacitaciones con opción a ser publicadas.



Taller	Comunidad	Parroquia	Canton	Inicia	Finaliza	Capacitador	Accion
Panaderia	Sinaí	Sinaí	Morona	4 de Julio de 2016	8 de Julio de 2016	Cristian Duchitanga	
N° registros 1							
Observación El apartado muestra o mostrará la lista de capacitaciones que seran ejecutadas en un cierto tiempo, para ello deben ser publicadas para que sean enviadas a los respectivos capacitadores.							

Figura 37-2 Lista de capacitaciones a publicar.
Fuente: DUCHITANGA, J., 2016

A continuación en la figura 38-2 se muestra la interfaz guía para publicar una capacitación.



Figura 38-2 Publicar capacitación.
Fuente: DUCHITANGA, J., 2016

A continuación en la figura 39-2 se muestra la interfaz guía para ejecutar las siguientes acciones: inscribir participantes a capacitaciones, listar participantes inscritos a ser aprobados o reprobados y dar por finalizada una capacitación.

LISTA CAPACITACIONES A REALIZAR							
Taller	Comunidad	Parroquia	Canton	Inicia	Finaliza	Capitador	Accion
Panaderia	Sinaí	Sinaí	Morona	4 de Julio de 2016	8 de Julio de 2016	Cristian Duchitanga	  
N° registros 1							
...?							

Figura 39-2 Capacitación publicada.

Fuente: DUCHITANGA, J., 2016

A continuación en la figura 40-2 se muestra la interfaz guía para inscribir a participantes a una capacitación.

REGISTRO

Taller

Participante

Figura 40-2 Inscribir a capacitación.

Fuente: DUCHITANGA, J., 2016

A continuación en la figura 41-2 se muestra la interfaz guía para aprobar o reprobar participantes de una capacitación.

PERSONAS REGISTRADAS					
Cedula	Nombre	Procedencia	Estado Civil	Ocupacion	Acción
1400738694	Julio Cristian	Sinaí	Soltero/a	Desarrollador	 
Taller de Panaderia en Sinaí					

Figura 41-2 Aprobar o reprobar participantes.

Fuente: DUCHITANGA, J., 2016

A continuación en la figura 42-2 se muestra la interfaz guía para listar las capacitaciones terminadas y generar los respectivos certificados para los participantes.

CAPACITACIONES TERMINADAS							
Taller	Cominidad	Parroquia	Canton	Inicia	Finaliza	Capacitador	Accion
Panaderia	Sinaí	Sinaí	Morona	26 de Junio de 2016	1 de Julio de 2016	Cristian Duchitanga	 
N° registros 1							
<p>Observación El apartado muestra o mostrará las capacitaciones finalizadas que esten a su cargo, para la generación de certificados; el capacitador es el único que puede dar por terminado una capacitación.</p>							

Figura 42-2 Capacitaciones terminada.
Fuente: DUCHITANGA, J., 2016

A continuación en la figura 43-2 se muestra la interfaz guía para generar o enlazar a los respectivos reportes.



Figura 43-2 Generar o enlazar reportes.
Fuente: DUCHITANGA, J., 2016

2.3.3. Casos de uso

Este apartado permitirá comprender las funciones de la aplicación, tanto para desarrolladores como para cliente final, además proporciona un enfoque claro sobre el comportamiento de la aplicación.

Para mejor entendimiento los casos de uso están organizados por módulos y los usuarios a cargo de cada módulo, además estos casos de uso involucran más de una historia de usuario.

A continuación en la figura 44-2 se muestra el caso de uso que se encarga de la administración: gestión de usuarios, gestión de funcionarios, gestión de beneficiarios, gestión de cantones, gestión de parroquias, gestión de comunidades, gestión de programas.

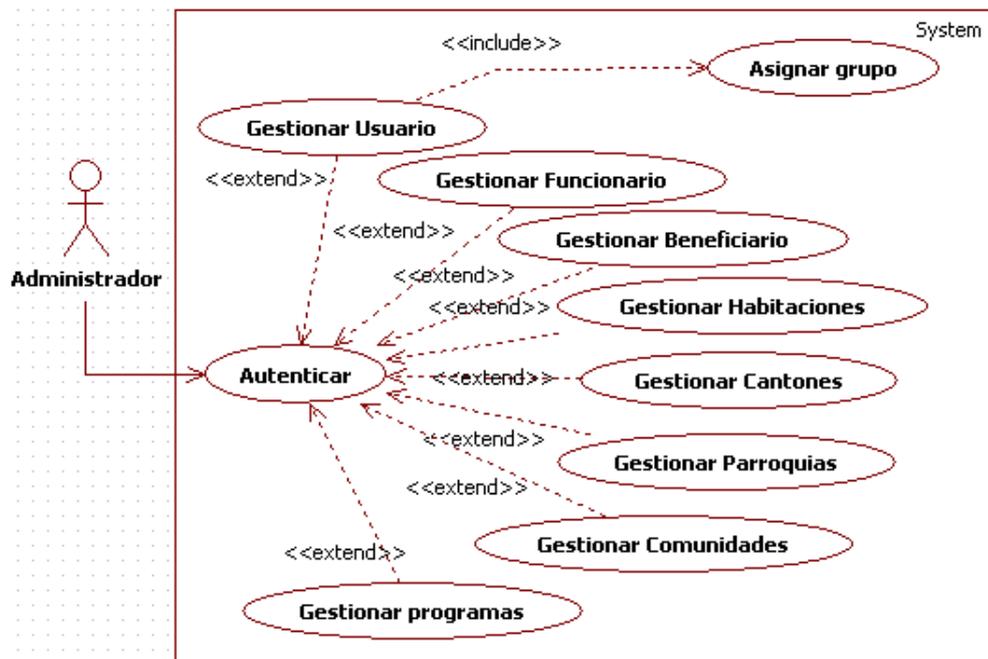


Figura 44-2 Diagrama de caso de uso administrador.
Fuente: DUCHITANGA, J., 2016

En la figura 18-2 se describe específicamente el caso de uso autenticar.

Tabla 18-2: Caso de uso Autenticar.

Identificador: 001	Nombre: Autenticar
Actores: Administrador	
Precondiciones: El administrador debe poseer una cuenta tipo súper usuario, predefinido por el sistema.	
Propósito: Loguear a cuenta de administrador.	
Descripción: El administrador debe ingresar usuario y contraseña para tener acceso al sitio de administración.	
Referencias	HU1
Tipo	Primario
Curso principal de eventos	
Acciones de Actores	Respuesta del sistema
1. Administrador solicita interfaz de logueo. 3. Administrador tipear Usuario y Contraseña y enviar los datos.	2. Visualiza interfaz para ejecutar acción solicitada por el administrador. 4. Valida datos ingresados y muestra interfaz de administración.
Curso eventos alternativos	
5. <u>Mensaje de advertencia (datos erróneos). Volver paso 3.</u>	

Fuente: DUCHITANGA, J., 2016

En la tabla 19-2 se describe específicamente el caso de uso gestionar usuario.

Tabla 19-2: Caso de uso Gestionar Usuario.

Identificador: 002	Nombre: Gestionar Usuario
Actores: Administrador	
Precondiciones: El administrador debe poseer una cuenta tipo súper usuario, además deben estar definidos los grupos de usuarios.	
Propósito: Crear, modificar y eliminar usuarios.	
Descripción: El administrador una vez autenticado, podrá gestionar (crear, modificar o eliminar) los usuarios que tendrán acceso al sistema.	
Referencias	HU1,HU2
Tipo	Primario
Curso principal de eventos	
Acciones de Actores	Respuesta del sistema
1. El usuario administrador inicia sesión en la aplicación haciendo uso de su nombre de usuario y contraseña.	3. Visualiza interfaz para ejecutar acción solicitada por el administrador.
2. Administrador solicita interfaz para realizar cualquiera de las siguientes operaciones: Crear, Modificar o eliminar Usuarios.	5. Valida datos ingresados y ejecuta acción solicitada.
4. Administrador efectúa la operación de una de las tareas listadas en el <u>paso2</u> .	
Curso eventos alternativos	
6. Mensaje de advertencia (datos requeridos). Volver paso 4.	

Fuente: DUCHITANGA, J., 2016

En la tabla 20-2 se describe específicamente el caso de uso asignar grupo.

Tabla 20-2: Caso de uso Asignar Grupo.

Identificador: 003	Nombre: Asignar Grupo
Actores: Administrador	
Precondiciones: El administrador debe poseer una cuenta tipo súper usuario predefinido por el sistema, deben existir usuarios registrados, además grupos predefinidos con sus respectivos permisos.	
Propósito: Asignar a usuario a un grupo predefinido.	
Descripción: El administrador una vez autenticado, podrá asignar un grupo a determinados usuarios (el usuario podrá pertenecer solo a un grupo).	
Referencias	HU1,HU4
Tipo	Primario
Curso principal de eventos	
Acciones de Actores	Respuesta del sistema
1. El administrador solicita lista de usuarios registrados.	2. Visualiza interfaz para ejecutar acción solicitada por el administrador (lista de usuarios).
3. Selecciona un usuario en específico, para asignar grupo.	4. Muestra interfaz para asignación de grupo.
6. Asigna un grupo al usuario seleccionado.	7. Valida y guarda la asignación.
Curso eventos alternativos	
8. Mensaje de advertencia (datos erróneos). Volver paso 6.	

Fuente: DUCHITANGA, J., 2016

En la tabla 21-2 se describe específicamente el caso de uso gestionar funcionario.

Tabla 21-2: Caso de uso Gestionar Funcionario.

Identificador: 003	Nombre: Gestionar Funcionario
Actores: Administrador	
Precondiciones: El administrador debe poseer una cuenta tipo súper usuario.	
Propósito: Crear, modificar y eliminar funcionario.	
Descripción: El administrador una vez autenticado, podrá gestionar (crear, modificar o eliminar) funcionarios para que tengan acceso al servicio de alimentación.	
Referencias	HU1,HU3
Tipo	Primario
Curso principal de eventos	
Acciones de Actores	Respuesta del sistema
1. El usuario administrador inicia sesión en la aplicación haciendo uso de su nombre de usuario y contraseña.	3. Visualiza interfaz para ejecutar acción solicitada por el administrador.
2. Administrador solicita interfaz para: Crear, Modificar o eliminar funcionario.	5. Valida datos ingresados y ejecuta acción solicitada.
4. Administrador efectúa la operación de una de las tareas listadas en el <u>paso2</u> .	
Curso eventos alternativos	
6. Mensaje de advertencia (datos erróneos o requeridos). Volver paso 4.	

Fuente: DUCHITANGA, J., 2016

En la tabla 22-2 se describe específicamente el caso de uso gestionar beneficiario.

Tabla 22-2: Caso de uso Gestionar Beneficiario.

Identificador: 004	Nombre: Gestionar Beneficiario
Actores: Administrador	
Precondiciones: El administrador debe poseer una cuenta tipo súper usuario.	
Propósito: Crear, modificar y eliminar beneficiario.	
Descripción: El administrador una vez autenticado, podrá gestionar (crear, modificar o eliminar) beneficiarios para que tengan acceso al servicio de posada y alimentación.	
Referencias	HU1,HU3
Tipo	Primario
Curso principal de eventos	
Acciones de Actores	Respuesta del sistema
1. Administrador solicita interfaz para: Crear, Modificar o eliminar beneficiario.	2. Visualiza interfaz para ejecutar acción solicitada por el administrador.
3. Administrador efectúa la operación de una de las tareas listadas en el <u>paso1</u> .	4. Valida datos ingresados y ejecuta acción solicitada.
Curso eventos alternativos	
6. Mensaje de advertencia (datos erróneos o requeridos). Volver paso 3.	

Fuente: DUCHITANGA, J., 2016

En la tabla 23-2 se describe específicamente el caso de uso gestionar cantones.

Tabla 23-2: Caso de uso Gestionar Cantones.

Identificador: 005	Nombre: Gestionar Cantones
Actores: Administrador	
Precondiciones: El administrador debe poseer una cuenta tipo súper usuario.	
Propósito: Crear, modificar y eliminar cantones.	
Descripción: El administrador una vez autenticado, podrá gestionar (crear, modificar o eliminar) cantones que serán parte de la información de beneficiarios.	
Referencias	HU1,HU3
Tipo	Primario
Curso principal de eventos	
Acciones de Actores	Respuesta del sistema
1. Administrador solicita interfaz para: Crear, Modificar o eliminar cantón.	2. Visualiza interfaz para ejecutar acción solicitada por el administrador.
3. Administrador efectúa la operación de una de las tareas listadas en el <u>paso1</u> .	4. Valida datos ingresados y ejecuta acción solicitada.
Curso eventos alternativos	
6. Mensaje de advertencia (datos erróneos o requeridos). Volver paso 3.	

Fuente: DUCHITANGA, J., 2016

En la tabla 24-2 se describe específicamente el caso de uso parroquias.

Tabla 24-2: Caso de uso Gestionar Parroquias.

Identificador: 006	Nombre: Gestionar Parroquias
Actores: Administrador	
Precondiciones: El administrador debe poseer una cuenta tipo súper usuario. Además deben existir cantones registrados.	
Propósito: Crear, modificar y eliminar parroquias.	
Descripción: El administrador una vez autenticado, podrá gestionar (crear, modificar o eliminar) parroquias que serán parte de la información de beneficiarios.	
Referencias	HU1,HU3
Tipo	Primario
Curso principal de eventos	
Acciones de Actores	Respuesta del sistema
1. Administrador solicita interfaz para: Crear, Modificar o eliminar parroquia.	2. Visualiza interfaz para ejecutar acción solicitada por el administrador.
3. Administrador efectúa la operación de una de las tareas listadas en el <u>paso1</u> .	4. Valida datos ingresados y ejecuta acción solicitada.
Curso eventos alternativos	
6. Mensaje de advertencia (datos erróneos o requeridos). Volver paso 3.	

Fuente: DUCHITANGA, J., 2016

En la tabla 25-2 se describe específicamente el caso de uso gestionar comunidades.

Tabla 25-2: Caso de uso Gestionar Comunidades.

Identificador: 007	Nombre: Gestionar Comunidades
Actores: Administrador	
Precondiciones: El administrador debe poseer una cuenta tipo súper usuario. Además debe estar registrado el cantón y la parroquia a la que pertenecerá la comunidad.	
Propósito: Crear, modificar y eliminar comunidad.	
Descripción: El administrador una vez autenticado, podrá gestionar (crear, modificar o eliminar) comunidades que serán parte de la información de beneficiarios.	
Referencias	HU1,HU3
Tipo	Primario
Curso principal de eventos	
Acciones de Actores	Respuesta del sistema
1. Administrador solicita interfaz para: Crear, Modificar o eliminar comunidad.	2. Visualiza interfaz para ejecutar acción solicitada por el administrador.
3. Administrador efectúa la operación de una de las tareas listadas en el <u>paso1</u> .	4. Valida datos ingresados y ejecuta acción solicitada.
Curso eventos alternativos	
6. Mensaje de advertencia (datos erróneos o requeridos). Volver paso 3.	

Fuente: DUCHITANGA, J., 2016

En la tabla 26-2 se describe específicamente el caso de uso gestionar programas.

Tabla 26-2: Caso de uso Gestionar Programas.

Identificador: 008	Nombre: Gestionar Programas
Actores: Administrador	
Precondiciones: El administrador debe poseer una cuenta tipo súper usuario.	
Propósito: Crear, modificar y eliminar programas.	
Descripción: El administrador una vez autenticado, podrá gestionar (crear, modificar o eliminar) programas que serán parte de la información de capacitaciones.	
Referencias	HU1,HU10
Tipo	Primario
Curso principal de eventos	
Acciones de Actores	Respuesta del sistema
1. Administrador solicita interfaz para: Crear, Modificar o eliminar programa.	2. Visualiza interfaz para ejecutar acción solicitada por el administrador.
3. Administrador efectúa la operación de una de las tareas listadas en el <u>paso1</u> .	4. Valida datos ingresados y ejecuta acción solicitada.
Curso eventos alternativos	
6. Mensaje de advertencia (datos erróneos o requeridos). Volver paso 3.	

Fuente: DUCHITANGA, J., 2016

A continuación en la figura 46-2 se muestra las funciones generales del módulo de posada y comedor en el diagrama de caso de uso para el usuario del centro intercultural del departamento de equidad, género y acción social.

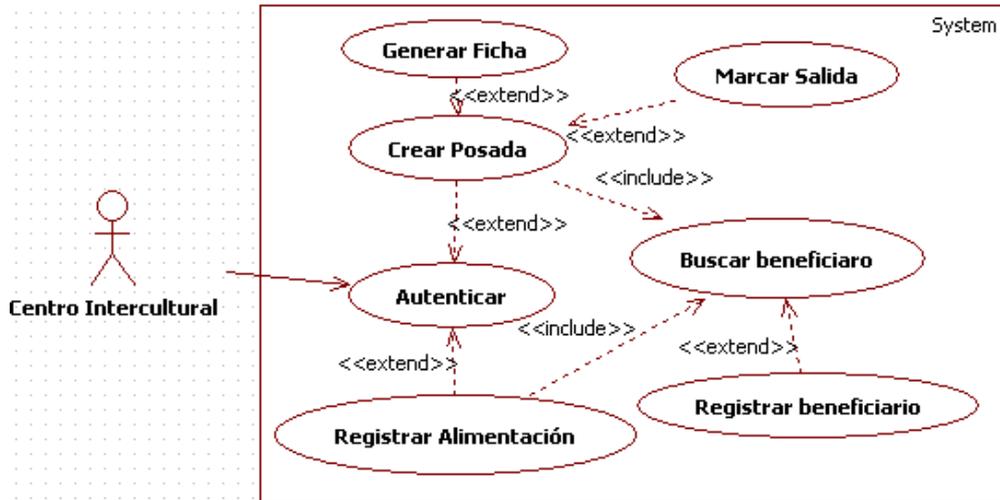


Figura 45-2 Diagrama de caso de uso Centro Intercultural.
Fuente: DUCHITANGA, J., 2016

En la tabla 27-2 se describe específicamente el caso de uso autenticar.

Tabla 27-2: Caso de uso Autenticar Responsable del Centro Intercultural.

Identificador: 009	Nombre: Autenticar
Actores: Responsable Centro intercultural (CI)	
Precondiciones: Contar con una cuenta de usuario y pertenecer a un grupo (Posada).	
Propósito: Loguear a módulo de Posada.	
Descripción: El responsable debe ingresar usuario y contraseña para tener acceso al módulo de posada y comedor.	
Referencias	HU1,HU2
Tipo	Primario
Curso principal de eventos	
Acciones de Actores	Respuesta del sistema
1. El responsable de CI solicita interfaz de logueo.	2. Visualiza interfaz para ejecutar acción solicitada por el responsable de CI.
3. El responsable de CI deberá tipear Usuario y Contraseña y enviar los datos.	4. Valida datos ingresados y muestra interfaz del módulo de posada.
Curso eventos alternativos	
5. Mensaje de advertencia (datos erróneos o requeridos). Volver paso 3.	

Fuente: DUCHITANGA, J., 2016

En la tabla 28-2 se describe específicamente el caso de uso crear posada.

Tabla 28-2: Caso de uso Crear Posada.

Identificador: 010	Nombre: Crear Posada
Actores: Responsable Centro intercultural (CI)	
Precondiciones: Contar con una cuenta de usuario y pertenecer a un grupo (Posada), el beneficiario debe estar registrado.	
Propósito: Crear posada para un beneficiario.	
Descripción: El responsable de CI una vez autenticado, tendrá la facultad de crear posadas para beneficiarios.	
Referencias	HU1,HU5
Tipo	Primario, esencial
Curso principal de eventos	
Acciones de Actores	Respuesta del sistema
1. El responsable de CI solicita interfaz para crear posada.	2. Visualiza interfaz para ejecutar acción solicitada por el responsable de CI.
3. El responsable de CI deberá tipear los parámetros requeridos y enviar los datos.	4. Valida y guarda los datos ingresados y muestra mensaje de éxito.
Curso eventos alternativos	
5. Mensaje de advertencia (datos erróneos o requeridos). Volver paso 3.	

Fuente: DUCHITANGA, J., 2016

En la tabla 29-2 se describe específicamente el caso de uso buscar beneficiario.

Tabla 29-2: Caso de uso Buscar Beneficiario.

Identificador: 011	Nombre: Buscar Beneficiario
Actores: Responsable Centro intercultural (CI)	
Precondiciones: Contar con una cuenta de usuario y pertenecer a un grupo (Posada) además puede o no estar registrado el beneficiario.	
Propósito: Buscar beneficiario.	
Descripción: El responsable de CI una vez autenticado, tendrá la facultad de buscar a un beneficiario para asignarle una posada o alimentación.	
Referencias	HU1,HU3
Tipo	Primario
Curso principal de eventos	
Acciones de Actores	Respuesta del sistema
1. El responsable de CI solicita interfaz de búsqueda.	2. Visualiza interfaz para ejecutar acción solicitada por el responsable de CI.
3. El responsable de CI debe tipear la cédula del beneficiario.	4. Valida los datos y realiza la búsqueda.
	5. Muestra la información del beneficiario.
Curso eventos alternativos	
5. Mensaje de advertencia (datos erróneos o requeridos). Volver paso 3.	

Fuente: DUCHITANGA, J., 2016

En la tabla 30-2 se describe específicamente el caso de uso registrar beneficiario.

Tabla 30-2: Caso de uso Registrar Beneficiario.

Identificador: 012	Nombre: Registrar Beneficiario
Actores: Responsable Centro intercultural (CI)	
Precondiciones: Contar con una cuenta de usuario y pertenecer a un grupo (Posada), el beneficiario no debe estar registrado.	
Propósito: Registrar beneficiario cuando no exista.	
Descripción: El responsable de CI una vez autenticado, tendrá la facultad de registrar beneficiario si no está registrado al pasar el proceso de búsqueda.	
Referencias	HU1,HU3
Tipo	Primario
Curso principal de eventos	
Acciones de Actores	Respuesta del sistema
1. El responsable de CI solicita interfaz para registrar beneficiario.	2. Visualiza interfaz para ejecutar acción solicitada por el responsable de CI.
3. El responsable de CI deberá tipear los parámetros requeridos y enviar los datos.	4. Valida y guarda los datos ingresados y muestra mensaje de éxito.
Curso eventos alternativos	
5. <u>Mensaje de advertencia (datos erróneos o requeridos). Volver paso 3.</u>	

Fuente: DUCHITANGA, J., 2016

En la tabla 31-2 se describe específicamente el caso de uso registrar alimentación.

Tabla 31-2: Caso de uso Registrar Alimentación.

Identificador: 013	Nombre: Registrar Alimentación
Actores: Responsable Centro intercultural (CI)	
Precondiciones: Contar con una cuenta de usuario y pertenecer a un grupo (Posada), el beneficiario debe estar registrado.	
Propósito: Registrar alimentación para un beneficiario.	
Descripción: El responsable de CI una vez autenticado, tendrá la facultad de registrar alimentación para beneficiarios registrados.	
Referencias	HU1,HU7
Tipo	Primario
Curso principal de eventos	
Acciones de Actores	Respuesta del sistema
1. El responsable de CI solicita interfaz para registrar alimentación.	2. Visualiza interfaz para ejecutar acción solicitada por el responsable de CI.
3. El responsable de CI debe tipear los parámetros requeridos y enviar los datos.	4. Valida y guarda los datos ingresados y muestra mensaje de éxito.
Curso eventos alternativos	
5. <u>Mensaje de advertencia (datos erróneos o requeridos). Volver paso 3.</u>	

Fuente: DUCHITANGA, J., 2016

En la tabla 32-2 se describe específicamente el caso de uso generar ficha.

Tabla 32-2: Caso de uso Generar Ficha.

Identificador: 014	Nombre: Generar Ficha
Actores: Responsable Centro intercultural (CI)	
Precondiciones: Contar con una cuenta de usuario y pertenecer a un grupo (Posada), debe existir una posada creada.	
Propósito: Generar ficha de posada.	
Descripción: El responsable de CI una vez autenticado, tendrá la facultad de generar ficha para legalizar la posada.	
Referencias	HU1,HU12
Tipo	Primario
Curso principal de eventos	
Acciones de Actores	Respuesta del sistema
1. El responsable de CI solicita lista de posadas.	2. Visualiza interfaz para ejecutar acción solicitada por el responsable de CI.
3. El responsable de CI deberá escoger una posada y seleccionar generar ficha.	4. Realiza consulta y genera ficha en formato imprimible.
Curso eventos alternativos	
5. Mensaje de advertencia (datos requeridos). Volver paso 3.	

Fuente: DUCHITANGA, J., 2016

En la tabla 33-2 se describe específicamente el caso de uso marcar salida.

Tabla 33-2: Caso de uso Marcar Salida.

Identificador: 015	Nombre: Marcar Salida
Actores: Responsable Centro intercultural (CI)	
Precondiciones: Contar con una cuenta de usuario y pertenecer a un grupo (Posada), debe existir una posada creada.	
Propósito: Marcar salida de un posante.	
Descripción: El responsable de CI una vez autenticado, tendrá la facultad de marcar la salida de un posante de la lista de posadas vencidas de acuerdo a la fecha de salida.	
Referencias	HU1,HU6
Tipo	Primario
Curso principal de eventos	
Acciones de Actores	Respuesta del sistema
1. El responsable de CI solicita lista de posadas.	2. Visualiza interfaz para ejecutar acción solicitada por el responsable de CI.
3. El responsable de CI deberá escoger una posada y seleccionar marcar salida.	4. Realiza la acción ejecutada y muestra mensaje de éxito.
Curso eventos alternativos	
5. Mensaje de advertencia (datos requeridos). Volver paso 3.	

Fuente: DUCHITANGA, J., 2016

A continuación en la figura 47-2 se muestra el caso de uso que generaliza las funciones del módulo de ayuda social de acuerdo a las historias de usuarios descritas en apartados anteriores obtenidos del departamento de equidad, género y acción social.

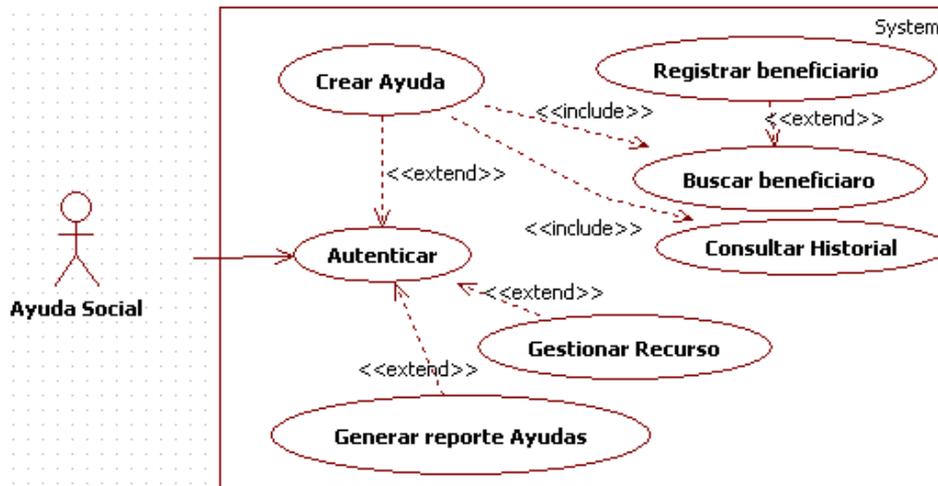


Figura 46-2 Diagrama de caso de uso Ayuda Social.
Fuente: DUCHITANGA, J., 2016

En la tabla 34-2 se describe específicamente el caso de uso autenticar.

Tabla 34-2: Caso de uso Autenticar a Ayuda Social.

Identificador: 016	Nombre: Autenticar
Actores: Trabajadora social	
Precondiciones: Contar con una cuenta de usuario y pertenecer a un grupo (Social).	
Propósito: Loguear a módulo de Ayuda Social.	
Descripción: El responsable debe ingresar usuario y contraseña para tener acceso al módulo de ayuda social.	
Referencias	HU1,HU2
Tipo	Primario
Curso principal de eventos	
Acciones de Actores	Respuesta del sistema
1. La Trabajadora social solicita interfaz de logueo.	2. Visualiza interfaz para ejecutar acción solicitada por la Trabajadora social.
3. La Trabajadora social tendrá que tipear Usuario y Contraseña y enviar los datos.	4. Valida datos ingresados y muestra interfaz del módulo de ayuda social.
Curso eventos alternativos	
5. Mensaje de advertencia (datos erróneos o requeridos). Volver paso 3.	

Fuente: DUCHITANGA, J., 2016

En la tabla 35-2 se describe específicamente el caso de uso crear ayuda.

Tabla 35-2: Caso de uso Crear Ayuda Social.

Identificador: 017	Nombre: Crear Ayuda
Actores: Trabajadora Social	
Precondiciones: Contar con una cuenta de usuario y pertenecer a un grupo (Social), el beneficiario tiene que estar registrado, además deben existir recursos registrados.	
Propósito: Crear ayuda social para un beneficiario que cumpla con los requerimientos para acceder al mismo.	
Descripción: La trabajadora social una vez autenticada; podrá crear ayuda para un beneficiarios, para efectuar esta operación el beneficiario debe estar registrado.	
Referencias	HU1, HU8
Tipo	Primario, esencial
Curso principal de eventos	
Acciones de Actores	Respuesta del sistema
1. La trabajadora social solicita interfaz para crear una ayuda social.	2. Visualiza interfaz para ejecutar acción solicitada por la trabajadora social
3. La trabajadora social debe llenar los parámetros solicitados y enviar los datos.	4. Valida y guarda los datos ingresados y muestra mensaje de éxito.
Curso eventos alternativos	
6. Mensaje de advertencia (datos erróneos o requeridos). Volver paso 3.	

Fuente: DUCHITANGA, J., 2016

En la tabla 36-2 se describe específicamente el caso de uso buscar beneficiario.

Tabla 36-2: Caso de uso Buscar Beneficiario para Ayuda Social.

Identificador: 018	Nombre: Buscar Beneficiario
Actores: Trabajadora social	
Precondiciones: Contar con una cuenta de usuario y pertenecer a un grupo (Social) además puede o no estar registrado el beneficiario.	
Propósito: Buscar beneficiario.	
Descripción: La trabajadora social una vez autenticado, tendrá la facultad de buscar a un beneficiario para asignarle una ayuda social.	
Referencias	HU1,HU3
Tipo	Primario
Curso principal de eventos	
Acciones de Actores	Respuesta del sistema
1. La trabajadora social solicita interfaz de búsqueda.	2. Visualiza interfaz para ejecutar acción solicitada por la trabajadora social.
3. La trabajadora deberá tipear la cédula del beneficiario.	4. Valida los datos y realiza la búsqueda.
	5. Muestra la información del beneficiario.
Curso eventos alternativos	
5. Mensaje de advertencia (datos erróneos o requeridos). Volver paso 3.	

Fuente: DUCHITANGA, J., 2016

En la tabla 37-2 se describe específicamente el caso de uso registrar beneficiario.

Tabla 37-2: Caso de uso Registrar Beneficiario para Ayuda Social.

Identificador: 019	Nombre: Registrar Beneficiario
Actores: Trabajadora social	
Precondiciones: Contar con una cuenta de usuario y pertenecer a un grupo (social), el beneficiario no debe estar registrado.	
Propósito: Registrar beneficiario cuando no exista.	
Descripción: La trabajadora social una vez autenticado, tendrá la facultad de registrar beneficiario si no está registrado al pasar el proceso de búsqueda.	
Referencias	HU1,HU3
Tipo	Primario
Curso principal de eventos	
Acciones de Actores	Respuesta del sistema
1. La trabajadora social solicita interfaz para registrar beneficiario.	2. Visualiza interfaz para ejecutar acción solicitada por la trabajadora social.
3. La trabajadora social deberá tipear los parámetros requeridos y enviar los datos.	4. Valida y guarda los datos ingresados y muestra mensaje de éxito.
Curso eventos alternativos	
5. Mensaje de advertencia (datos erróneos o requeridos). Volver paso 3.	

Fuente: DUCHITANGA, J., 2016

En la tabla 38-2 se describe específicamente el caso de uso consultar historial.

Tabla 38-2: Caso de uso Crear consultar historial de ayudas.

Identificador: 020	Nombre: Consultar Historial
Actores: Trabajadora social	
Precondiciones: Tener una cuenta de usuario asignado al grupo social, debe existir un registro de ayudas.	
Propósito: Consultar historial de ayudas entregadas para un beneficiario en específico.	
Descripción: La trabajadora social una vez autenticado; podrá consultar historia de ayudas entregadas.	
Referencias	HU1, HU8
Tipo	Primario
Curso principal de eventos	
Acciones de Actores	Respuesta del sistema
1. La trabajadora social solicita interfaz buscar historial de beneficiario.	2. Visualiza interfaz para ejecutar acción solicitada por la trabajadora social.
3. En la interfaz presentada ingresar cédula del beneficiario y enviar datos.	4. Valida datos y realiza consulta.
	5. Muestra el historial del beneficiario
Curso eventos alternativos	
6. Mensaje de advertencia (dato erróneo o requerido). Volver paso 3.	

Fuente: DUCHITANGA, J., 2016

En la tabla 39-2 se describe específicamente el caso de uso gestionar recursos.

Tabla 39-2: Caso de uso Gestionar Recursos.

Identificador: 021	Nombre: Gestionar Recursos
Actores: Trabajadora social	
Precondiciones: Contar con una cuenta de usuario y pertenecer a un grupo (social).	
Propósito: Crear, modificar y eliminar recursos para ayudas sociales.	
Descripción: La trabajadora social una vez autenticado, podrá gestionar (crear, modificar o eliminar) recursos que serán parte de la información de ayudas sociales.	
Referencias	HU1, HU9
Tipo	Primario
Curso principal de eventos	
Acciones de Actores	Respuesta del sistema
1. La trabajadora social solicita interfaz para: Crear, Modificar o eliminar programa.	2. Visualiza interfaz para ejecutar acción solicitada por la trabajadora social.
3. La trabajadora social deberá cumplir con los parámetros de una de las tareas listadas en el <u>paso1</u> .	4. Valida datos ingresados y ejecuta acción solicitada mostrando un mensaje de éxito.
Curso eventos alternativos	
6. Mensaje de advertencia (datos erróneos o requeridos). Volver paso 3.	

Fuente: DUCHITANGA, J., 2016

A continuación en la figura 48-2 se muestra el caso de uso que generaliza las funciones del usuario denominado capacitador de acuerdo a las historias de usuarios descritas en apartados anteriores obtenidos del departamento de equidad, género y acción social.

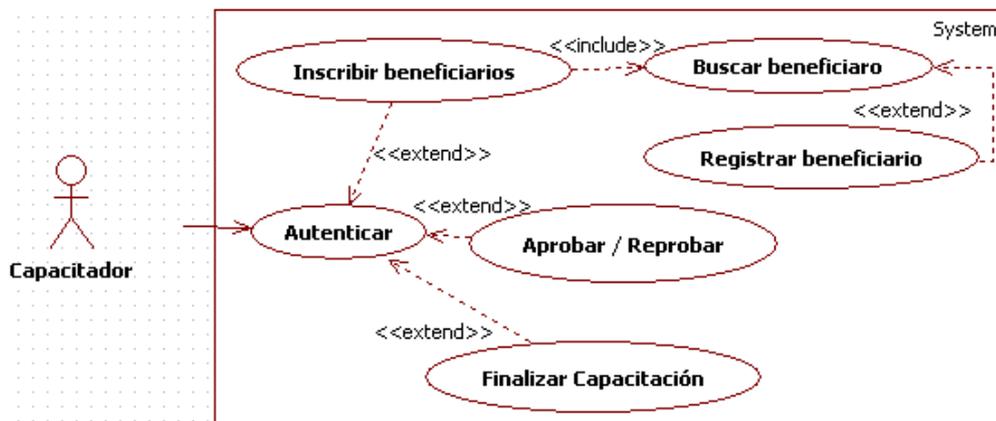


Figura 47-2 Diagrama de caso de uso Capacitador.

Fuente: DUCHITANGA, J., 2016

Para una mejor comprensión del caso de uso usuario autenticar se describe el mismo en la tabla 40-2.

Tabla 40-2: Caso de uso Autenticar Capacitador.

Identificador: 022	Nombre: Autenticar
Actores: Capacitador	
Precondiciones: Contar con una cuenta de usuario y pertenecer a un grupo (Capacitador).	
Propósito: Loguear a la cuenta de usuario capacitador.	
Descripción: El capacitador debe ingresar usuario y contraseña para tener acceso al módulo de capacitaciones.	
Referencias	HU1,HU2,HU4
Tipo	Primario
Curso principal de eventos	
Acciones de Actores	Respuesta del sistema
1. El capacitador social solicita interfaz de logueo.	2. Visualiza interfaz para ejecutar acción solicitada por el capacitador.
3. El capacitador tendrá que tipear Usuario y Contraseña y enviar los datos.	4. Valida datos ingresados y muestra interfaz del módulo de capacitaciones.
Curso eventos alternativos	
5. Mensaje de advertencia (datos erróneos o requeridos). Volver paso 3.	

Fuente: DUCHITANGA, J., 2016

En la tabla 41-2 se describe específicamente el caso de uso inscribir beneficiarios.

Tabla 41-2: Caso de uso inscribir beneficiarios.

Identificador: 023	Nombre: Inscribir beneficiarios
Actores: Capacitador	
Precondición: Tener una cuenta de usuario asignada al grupo de capacitadores, además el beneficiario y la capacitación deben estar registrados.	
Propósito: Inscribir beneficiarios a una capacitación	
Descripción: El capacitador una vez autenticado tendrá la facultad de inscribir beneficiarios a una capacitación, para ello el beneficiario deberá estar registrado con anterioridad.	
Referencias	HU1,HU11
Tipo	Primario, esencial
Curso principal de eventos	
Acciones de Actores	Respuesta del sistema
1. Capacitador solicita interfaz para inscribir un beneficiario.	2. Visualiza interfaz para ejecutar acción solicitada por el capacitador
3. Capacitador deberá tipear los parámetros solicitados y enviar los datos.	4. Valida los datos ingresados y guarda el registros de inscripción.
	5. Muestra un mensaje de éxito.
Curso eventos alternativos	
6. Mensaje de advertencia (datos erróneos o requeridos). Volver paso 3.	

Fuente: DUCHITANGA, J., 2016

En la tabla 42-2 se describe específicamente el caso de uso Aprobar o reprobar.

Tabla 42-2: Caso de uso Aprobar o Reprobar Capacitación.

Identificador: 024	Nombre: Aprobar / Reprobar
Actores: Capacitador	
Precondición: Tener una cuenta de usuario asignada al grupo de capacitadores, además el beneficiario debe estar inscrito a una capacitación.	
Propósito: Aprobar o reprobar beneficiario de una capacitación	
Descripción: El capacitador una vez autenticado tendrá la facultad de aprobar o reprobar a beneficiarios que participan de una capacitación bajo el criterio del capacitador.	
Referencias	HU1,HU11
Tipo	Primario
Curso principal de eventos	
Acciones de Actores	Respuesta del sistema
1. Capacitador solicita lista de beneficiarios de una capacitación. 3. Capacitador elegir un beneficiario de la lista y seleccionar aprobar o reprobar según su criterio y enviar datos.	2. Visualiza interfaz con la lista de beneficiarios solicitada por el capacitador. 4. Valida los datos ingresados y actualiza el perfil de la inscripción. 5. Muestra un mensaje de éxito.
Curso eventos alternativos	
6. Mensaje de advertencia (datos erróneos o requeridos). Volver paso 3.	

Fuente: DUCHITANGA, J., 2016

En la tabla 43-2 se describe específicamente el caso de uso finalizar capacitación.

Tabla 43-2: Caso de uso Finalizar Capacitación.

Identificador: 025	Nombre: Finalizar Capacitación
Actores: Capacitador	
Precondición: Tener una cuenta de usuario asignada al grupo de capacitadores.	
Propósito: Aprobar o reprobar beneficiario de una capacitación	
Descripción: El capacitador una vez autenticado tendrá la facultad de finalizar capacitaciones, para ello la capacitación deben haber cumplido la fecha de finalización.	
Referencias	HU1,HU10
Tipo	Primario
Curso principal de eventos	
Acciones de Actores	Respuesta del sistema

1. Capacitador solicita lista de capacitaciones activas.	2. Visualiza interfaz con la lista de capacitaciones solicitada por el capacitador.
3. Capacitador deberá elegir una capacitación de la lista y seleccionar terminar capacitación.	4. Valida la solicitud y solicita confirmación.
4. Capacitador confirma solicitud.	5. Muestra un mensaje de éxito.
Curso eventos alternativos	
6. Mensaje de advertencia (datos erróneos). Volver paso 3.	

Fuente: DUCHITANGA, J., 2016

En la tabla 44-2 se describe específicamente el caso de uso buscar beneficiario.

Tabla 44-2: Caso de uso Buscar Beneficiario para Inscribir a Capacitación.

Identificador: 026	Nombre: Buscar Beneficiario
Actores: Capacitador	
Precondiciones: Tener una cuenta de usuario asignada al grupo de capacitadores, además puede o no estar registrado el beneficiario.	
Propósito: Buscar beneficiario.	
Descripción: El capacitador una vez autenticado, tendrá la facultad de buscar a un beneficiario para inscribirle a una capacitación.	
Referencias	HU1,HU3
Tipo	Primario
Curso principal de eventos	
Acciones de Actores	Respuesta del sistema
1. El capacitador solicita interfaz de búsqueda.	2. Visualiza interfaz para ejecutar acción solicitada por el capacitador.
3. El capacitador deberá tipear la cédula del beneficiario.	4. Valida los datos y realiza la búsqueda.
	5. Muestra la información del beneficiario.
Curso eventos alternativos	
5. Mensaje de advertencia (datos erróneos o requeridos). Volver paso 3.	

Fuente: DUCHITANGA, J., 2016

En la tabla 45-2 se describe específicamente el caso de uso registrar beneficiario.

Tabla 45-2: Caso de uso Registrar Beneficiario para Capacitación.

Identificador: 027	Nombre: Registrar Beneficiario
Actores: Capacitador	
Precondiciones: Tener una cuenta de usuario asignada al grupo de capacitadores, el beneficiario no debe estar registrado.	
Propósito: Registrar beneficiario cuando no exista.	
Descripción: El capacitador una vez autenticado, tendrá la facultad de registrar beneficiario si no está registrado al pasar el proceso de búsqueda.	
Referencias	HU1,HU3
Tipo	Primario

Curso principal de eventos	
Acciones de Actores	Respuesta del sistema
1. El capacitador solicita interfaz para registrar beneficiario.	2. Visualiza interfaz para ejecutar acción solicitada por el capacitador.
3. El capacitador deberá tipear los parámetros requeridos y enviar los datos.	4. Valida y guarda los datos ingresados y muestra mensaje de éxito.
Curso eventos alternativos	
5. Mensaje de advertencia (datos erróneos o requeridos). Volver paso 3.	

Fuente: DUCHITANGA, J., 2016

A continuación en la figura 49-2 se muestra el caso de uso que generaliza las funciones del usuario denominado asistente administrativo de acuerdo a las historias de usuarios descritas en apartados anteriores obtenidos del departamento de equidad, género y acción social.

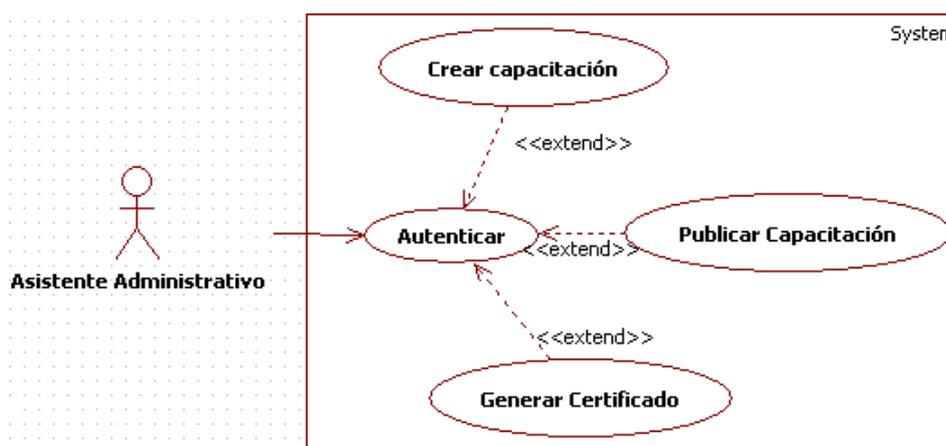


Figura 48-2 Diagrama de caso de uso Asistente Administrativo.

Fuente: DUCHITANGA, J., 2016

Para una mejor comprensión del caso de uso autenticar se describe el mismo en la tabla 46-2.

Tabla 46-2: Caso de uso Autenticar Asistente Administrativo

Identificador: 028	Nombre: Autenticar
Actores: Asistente administrativo	
Precondiciones: Tener una cuenta de usuario asignada al grupo de Técnicos	
Propósito: Loguear a la cuenta de usuario capacitador.	
Descripción: El Asistente administrativo debe ingresar usuario y contraseña para tener acceso al módulo de capacitaciones.	
Referencias	HU1,HU2,HU4
Tipo	Primario
Curso principal de eventos	
Acciones de Actores	Respuesta del sistema

1. El Asistente administrativo social solicita interfaz de logueo.	2. Visualiza interfaz para ejecutar acción solicitada por el Asistente administrativo.
3. El Asistente administrativo tendrá que tipear Usuario y Contraseña y enviar los datos.	4. Valida datos ingresados y muestra interfaz del módulo de capacitaciones.
Curso eventos alternativos	
5. Mensaje de advertencia (datos erróneos o requeridos). Volver paso 3.	

Fuente: DUCHITANGA, J., 2016

En la tabla 47-2 se describe específicamente el caso de uso crear capacitación.

Tabla 47-2: Caso de uso Crear Capacitación.

Identificador: 029	Nombre: Crear Capacitación
Actores: Asistente administrativo	
Precondición: Tener una cuenta de usuario asignada al grupo de Técnicos, además debe existir programas y capacitadores registrados.	
Propósito: Crear o registrar una capacitación.	
Descripción: El Asistente administrativo una vez autenticado, podrá crear capacitaciones asignándolas a un determinado capacitador para posteriormente ser publicadas.	
Referencias	HU10
Tipo	Primario, esencial
Curso principal de eventos	
Acciones de Actores	Respuesta del sistema
1. Asistente administrativo solicita interfaz para crear una capacitación. 3. Asistente administrativo llena los parámetros requeridos en el registro y envía los datos.	2. Visualiza interfaz para ejecutar acción solicitada por el asistente administrativo. 4. Valida datos ingresados y ejecuta acción solicitada. 5. Muestra mensaje de éxito.
Curso eventos alternativos	
6. Mensaje de advertencia (datos erróneos o requeridos). Volver paso 3.	

Fuente: DUCHITANGA, J., 2016

En la tabla 48-2 se describe específicamente el caso de uso publicar capacitación.

Tabla 48-2: Caso de uso Publicar Capacitación.

Identificador: 030	Nombre: Publicar Capacitación
Actores: Asistente administrativo	
Precondición: Tener una cuenta de usuario asignada al grupo de Técnicos, además debe existir capacitaciones creadas o registradas.	
Propósito: Publicar capacitación para enviar al capacitador asignado.	
Descripción: El Asistente administrativo una vez autenticado tendrá la facultad del publicar	

capacitaciones registradas, con el propósito de ser enviadas al capacitador.	
Referencias	HU1,HU10
Tipo	Primario, esencial
Curso principal de eventos	
Acciones de Actores	Respuesta del sistema
1. Asistente administrativo solicita interfaz con la lista de capacitaciones sin publicar. 3. Asistente administrativo elige una capacitación y deberá seleccionar publicar. 5. Asistente administrativo confirma solicitud.	2. Visualiza interfaz para ejecutar acción solicitada por el asistente administrativo. 4. Valida datos ingresados y solicita confirmación 6. Muestra mensaje de éxito.
Curso eventos alternativos	
6. Mensaje de advertencia (datos erróneos o requeridos). Volver paso 3.	

Fuente: DUCHITANGA, J., 2016

En la tabla 49-2 se describe específicamente el caso de uso generar certificado.

Tabla 49-2: Caso de uso Generar Certificado.

Identificador: 031	Nombre: Generar Certificado
Actores: Asistente administrativo	
Precondición: Tener una cuenta de usuario asignada al grupo de Técnicos, la capacitación marcada por el capacitador como finalizado y beneficiarios aprobados.	
Propósito: Generar certificado para beneficiarios que aprobaron una capacitación.	
Descripción: El Asistente administrativo una vez autenticado tendrá la facultad de generar certificado en un formato imprimible, para ello la capacitación debe haber sido finalizado con participantes aprobados.	
Referencias	HU1, HU10, HU11, HU12
Tipo	Primario, esencial
Curso principal de eventos	
Acciones de Actores	Respuesta del sistema
1. Asistente administrativo solicita interfaz con la lista de capacitaciones terminadas. 3. Asistente administrativo deberá elegir una capacitación y solicita la lista de beneficiarios que aprobaron la capacitación. 5. Asistente administrativo deberá elegir un beneficiario y seleccionar generar certificado.	2. Visualiza interfaz para ejecutar acción solicitada por el asistente administrativo. 4. Visualiza interfaz con la lista de beneficiarios aprobados. 6. Valida y realiza consulta de la información solicitada. 7. Genera certificado.
Curso eventos alternativos	
6. Mensaje de advertencia Volver paso 5.	

Fuente: DUCHITANGA, J., 2016

2.3.4. Diagrama de procesos

Los diagramas que a continuación se muestran permiten identificar de mejor manera los procesos que realizará la aplicación web dentro de un área respectiva del departamento de Equidad y Género y Acción Social, para ello se diseñara un diagrama para cada módulo.

2.4.4.1. Diagramas del módulo de posada y comedor

A continuación en la figura 50-2 se muestra el proceso que llevará la aplicación para el control de acceso a SIGEGAS, este proceso involucra la historia de usuario 1 (HU1) de la primera iteración descritas en procesos anteriores.

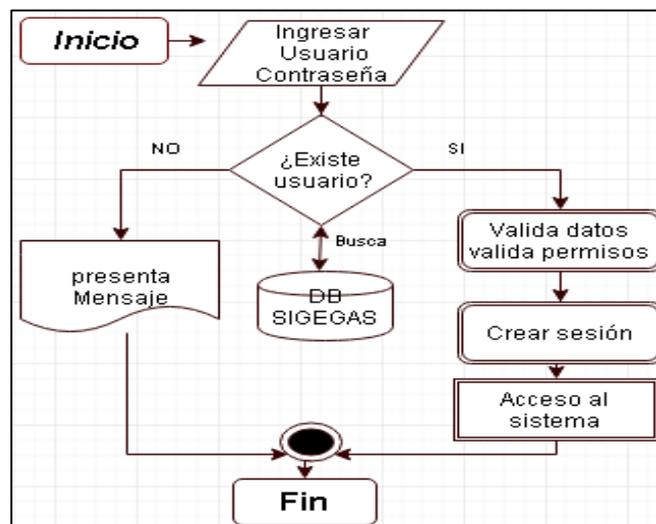


Figura 49-2 Proceso para acceso al sistema.
Fuente: DUCHITANGA, J., 2016

A continuación en la figura 51-2 se muestra el proceso que llevará la aplicación para crear un usuario y asignarle a un grupo dentro de SIGEGAS, este proceso involucra la historia de usuario 2 y 4 (HU2, HU4) de la primera iteración descritas en procesos anteriores.

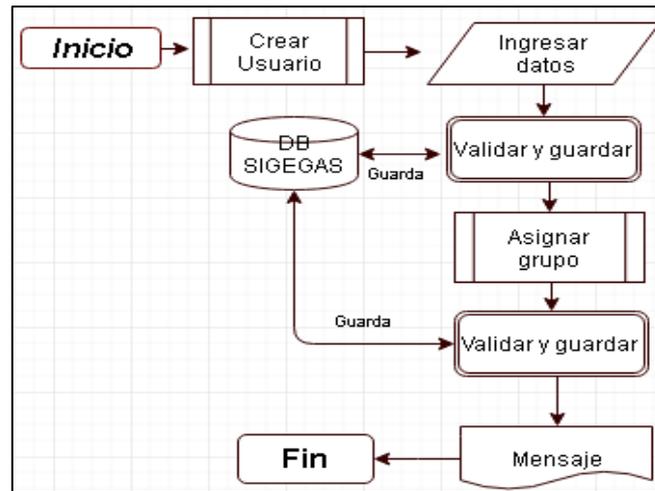


Figura 50-2 Crear usuario y asignar grupo.
 Fuente: DUCHITANGA, J., 2016

A continuación en la figura 52-2 se muestra el proceso que llevará la aplicación para crear una posada; este proceso involucra dos historias de usuarios (HU3, HU5) de la primera iteración descritas en procesos anteriores.

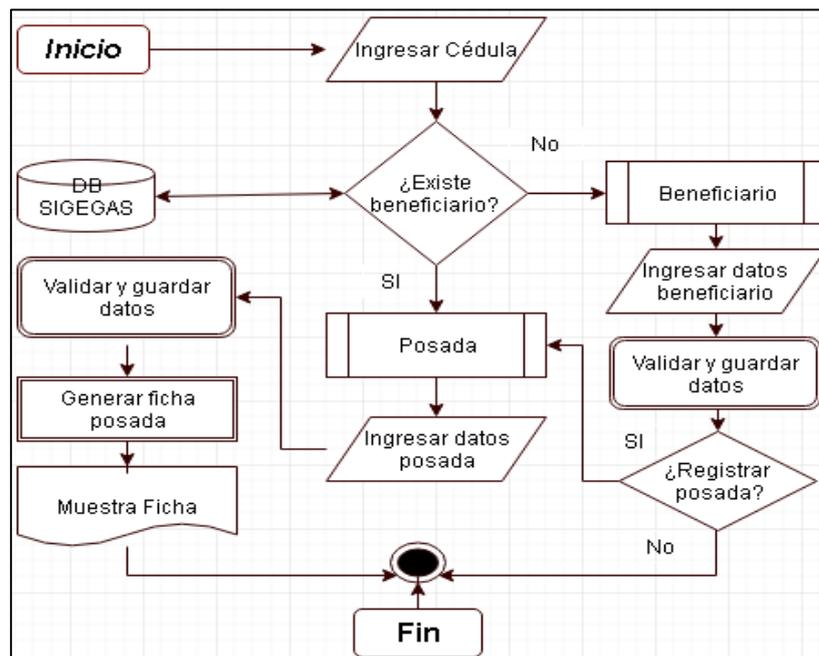


Figura 51-2 Crear posada.
 Fuente: DUCHITANGA, J., 2016

A continuación en la figura 53-2 se muestra el proceso que llevará la aplicación para marcar la salida de posantes, este proceso involucra una historia de usuario (HU6) de la primera iteración descrita en procesos anteriores.

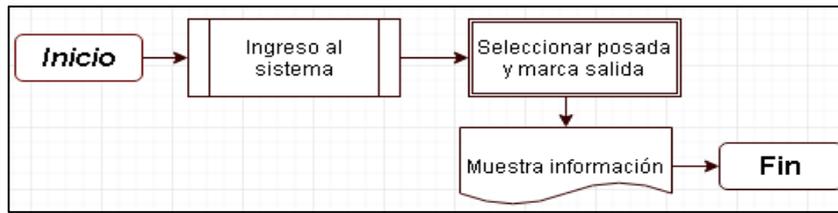


Figura 52-2 Salida de posante.
 Fuente: DUCHITANGA, J., 2016

A continuación en la figura 54-2 se muestra el proceso que llevará la aplicación para registro de alimentación para beneficiarios y funcionarios, este proceso involucra una historia de usuario (HU3) de la primera iteración y una historia de usuario (HU7) de la segunda iteración descrita en procesos anteriores.

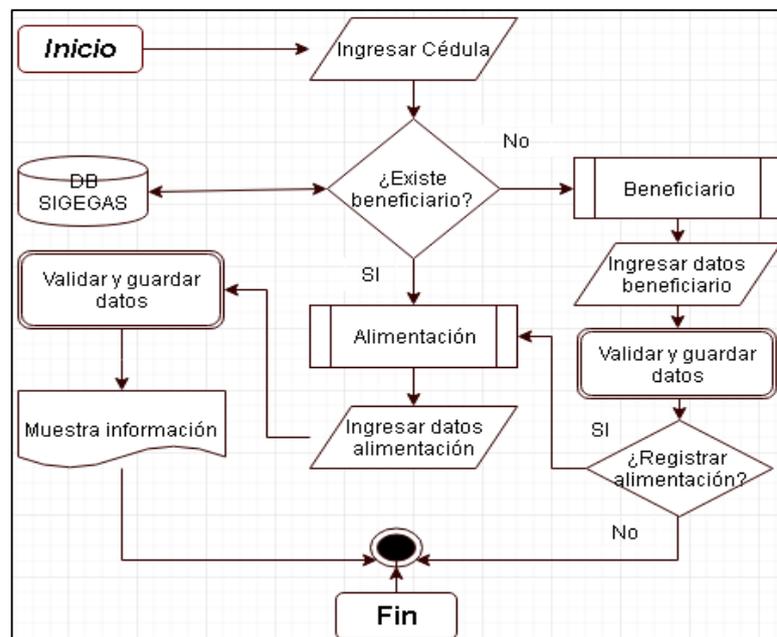


Figura 53-2 Registro de alimentación.
 Fuente: DUCHITANGA, J., 2016

2.4.4.2. Diagramas del módulo de ayuda social

A continuación en la figura 55-2 se muestra el proceso que llevará la aplicación para registrar una ayuda social para un beneficiario, este proceso involucra una historia de usuario (HU3) de la primera iteración y una historia de usuario (HU8) de la segunda iteración descrita en procesos anteriores.

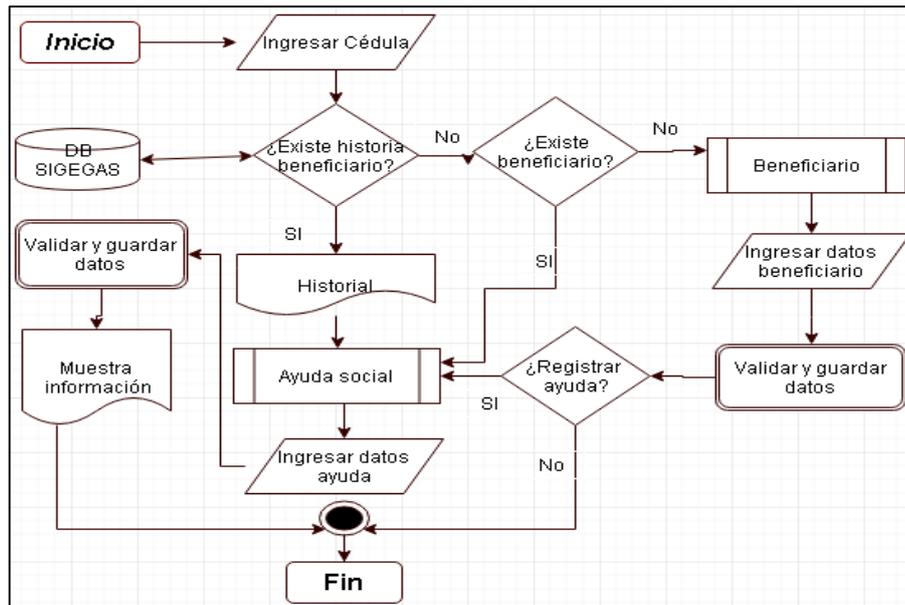


Figura 54-2 Registro de ayuda social.
Fuente: DUCHITANGA, J., 2016

A continuación en la figura 56-2 se muestra el proceso que llevará la aplicación para registrar un recurso para ser utilizado en ayudas sociales, este proceso involucra una historia de usuario (HU9) de la segunda iteración descrita en procesos anteriores.

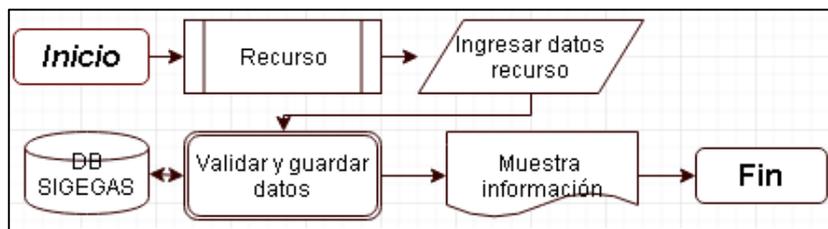


Figura 55-2 Registro de recurso.
Fuente: DUCHITANGA, J., 2016

2.4.4.3. Diagramas del módulo de capacitaciones

A continuación en la figura 57-2 se muestra el proceso que llevará la aplicación para registrar capacitaciones, este proceso involucra una historia de usuario (HU10) de la tercera iteración descrita en procesos anteriores.



Figura 56-2 Registro de una capacitación.
Fuente: DUCHITANGA, J., 2016

A continuación en la figura 58-2 se muestra el proceso que llevará la aplicación para la inscripción de personas (beneficiarios) a una capacitación, este proceso involucra una historia de usuario (HU11) de la tercera iteración descrita en procesos anteriores.

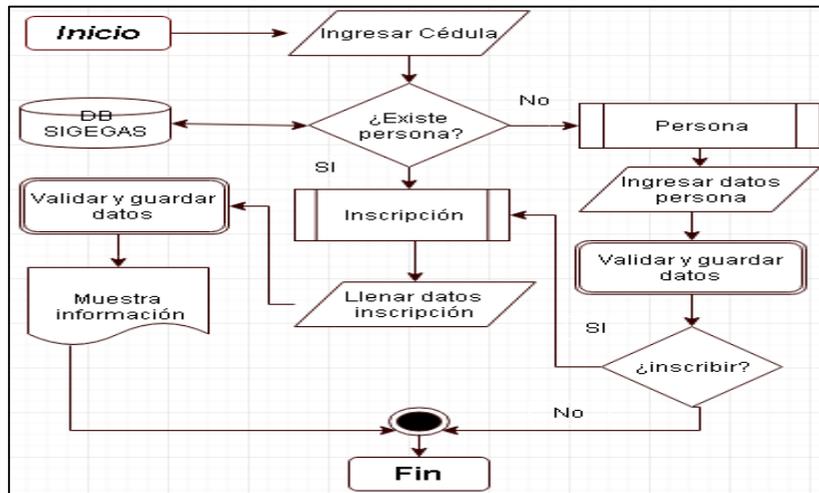


Figura 57-2 Inscripción de personas.
Fuente: DUCHITANGA, J., 2016

2.4.4.4. Diagrama de Reportes

A continuación en la figura 59-2 se muestra el diagrama general para la emisión de reportes para todos los módulos principales de SIGEGAS, este proceso involucra una historia de usuario (HU12) descrita en procedimientos anteriores.

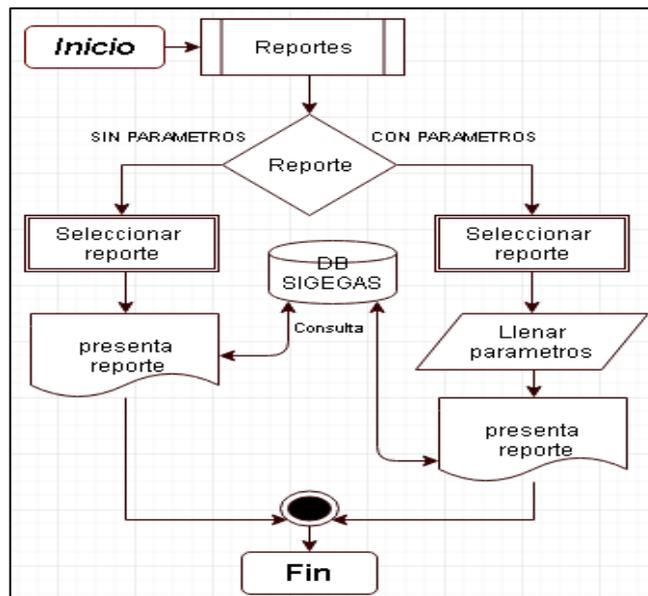


Figura 58-2 Emitir reporte.
Fuente: DUCHITANGA, J., 2016

2.3.5. Arquitectura de desarrollo

En la figura 60-2 se muestra la arquitectura utilizada en el desarrollo de la aplicación web SIGEGAS, el cual está basado en el patrón MTV del framework Django utilizado para el desarrollo.

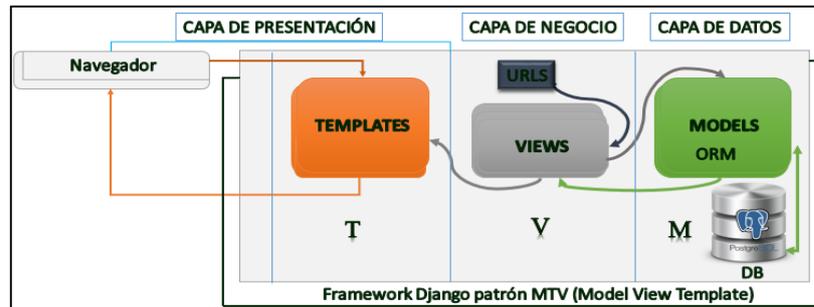


Figura 59-2 Arquitectura django.
Fuente: DUCHITANGA, J., 2016

2.3.6. Tareas por historias de usuario

2.3.6.1. Tareas primera iteración.

Tabla 50-2: Tareas de HU Acceso al sistema.

HU1: Acceso al sistema		
Número	Nombre de tarea	Tiempo (semanas)
1	Configuración del entorno de desarrollo	1
2	Diseño e implementación de base de datos	1
3	Diseño de plantilla para login	0.5
4	Implementación del vistas en Django	0.5

Fuente: DUCHITANGA, J., 2016

Tabla 51-2: Configuración del entorno de desarrollo.

Nombre Tarea: Configuración del entorno de desarrollo	
Numero tarea: 1	Numero de historia: 1
Tipo de tarea: Desarrollo	Tiempo estimado: 1
Fecha inicio: 01/04/2016	Fecha fin: 05/04/2016
Programador responsable: Julio Duchitanga	
Descripción: Instalación y configuración del entorno de desarrollo web haciendo uso de software libre; a continuación se lista los componentes a utilizar:	
Plataforma de desarrollo: Python.	
Framework de desarrollo: Django.	
Base de datos: PostgreSQL 9.4, pgAdminIII.	
Editor de texto: Atom.	
Servidor de desarrollo: Propio de Django.	
Librerías: Bootstrap3, JQuery.	

Fuente: DUCHITANGA, J., 2016

Tabla 52-2: Diseño e implementación de base de datos.

Nombre Tarea: Diseño e implementación de base de datos	
Numero tarea: 2	Numero de historia: 1
Tipo de tarea: Desarrollo	Tiempo estimado: 1
Fecha inicio: 07/04/2016	Fecha fin: 11/04/2016
Programador responsable: Julio Duchitanga	
Descripción: Analizar, diseñar e implementar la base de datos para la aplicación web SIGEGAS, además configurara la base de datos propia de Django para utilizar la interfaz administrativa por defecto que trae el framework.	

Fuente: DUCHITANGA, J., 2016

Tabla 53-2: Diseño de plantilla para login.

Nombre Tarea: Diseño de plantilla para login	
Numero tarea: 3	Numero de historia: 1
Tipo de tarea: Desarrollo	Tiempo estimado: 0.5
Fecha inicio: 12/04/2016	Fecha fin: 15/04/2016
Programador responsable: Julio Duchitanga	
Descripción: Diseño de la plantilla del formulario login haciendo uso de técnicas de etiquetado y estilos basados en HTML5 como lo son CSS3 puro y librerías Bootstrap3, JQuery.	
Además se procederá a crear el archivo Urls.py (controlador) para acceder a los datos desde las vistas en Django, para ello se creará una urlConf para esta plantilla.	

Fuente: DUCHITANGA, J., 2016

Tabla 54-2: Implementación de vistas en Django.

Nombre Tarea: Implementación de vistas en Django	
Numero tarea: 4	Numero de historia: 1
Tipo de tarea: Desarrollo	Tiempo estimado: 0.5
Fecha inicio: 16/04/2016	Fecha fin: 21/04/2016
Programador responsable: Julio Duchitanga	
Descripción: Implementación de funciones Python en el archivo views.py para el acceso a datos del usuario de las tablas propias de Django, además se utilizará funciones que proporciona el framework. A continuación una lista algunas de estas:	
home_page(): función para validar datos recibidos del formulario y autenticar usuario e iniciar sesión	
Authenticate (): método para autenticar usuario.	
is_Active (): método para ver si el usuario está activo.	
login: método para iniciar sesión	
logout_ (): función para cerrar sesión haciendo uso de método propio de Django logout.	

Fuente: DUCHITANGA, J., 2016

Tabla 55-2: Tareas de HU Gestión de usuario.

HU2: Gestión de Usuario		
Número	Nombre de tarea	Tiempo(semanas)
1	Instalar tablas propias de Django	0.2
2	Activar la interfaz administrativa de Django.	0.3

Fuente: DUCHITANGA, J., 2016

Tabla 56-2: Instalar tablas propias de Django.

Nombre Tarea: Instalar tablas propias de Django	
Numero tarea: 1	Numero de historia: 2
Tipo de tarea: Desarrollo	Tiempo estimado: 0.2
Fecha inicio: 23/04/2016	Fecha fin: 24/04/2016
Programador responsable: Julio Duchitanga	
<p>Descripción: Crear las tablas propias del framework Django dentro de las cuales se encuentra la entidad llamada user(dentro de la aplicación django.contrib.auth) con todas las consultas necesarias para insertar, modificar o eliminar. A continuación el comando para crear tablas:</p> <p>Python manage.py migrate: Verifica el esquema de base de datos para comprobar si las tablas existen, caso contrario crea.</p>	

Fuente: DUCHITANGA, J., 2016

Tabla 57-2: Activar la interfaz administrativa de Django.

Nombre Tarea: Activar la interfaz administrativa de Django.	
Numero tarea: 2	Numero de historia: 2
Tipo de tarea: Desarrollo	Tiempo estimado: 0.3
Fecha inicio: 25/04/2016	Fecha fin: 27/04/2016
Programador responsable: Julio Duchitanga	
<p>Descripción: Configuración del archivo setting.py para habilitar el centro de administración Django. El sitio de administración al ser habilitado proporciona las tareas básicas para administrar sitios web, como por ejemplo gestionar usuarios.</p> <p>django.contrib.admin: agregar esta aplicación a la variable INSTALLED_APP para activar y usar la interfaz de administración. Verificar que las siguientes dependencias estén activadas.</p> <ul style="list-style-type: none"> • django.contrib.auth • django.contrib.contenttypes • django.contrib.messages • django.contrib.sessions <p>Python manage.py createsuperuser: crear un súper-usuario para el sitio, para ello elegir un usuario y contraseña para acceder al mismo.</p> <p>Agregar o habilitar el patrón url para acceder al sitio, para ello acceder al archivo urls.py</p> <p>Python manage.py runserver: iniciar el servidor de pruebas</p> <p>Localhost: 8000/Admin: acceder a la interfaz administrativa.</p>	

Fuente: DUCHITANGA, J., 2016

Tabla 58-2: Tareas de HU gestión de beneficiarios.

HU3: Gestión de beneficiarios		
Número	Nombre de tarea	Tiempo(semanas)
1	Crear formulario basado en clases	0.1
2	Diseñar plantilla para ingresar, editar.	0.1
3	Crear patrón url y vista (funciones) Django.	0.3

Fuente: DUCHITANGA, J., 2016

Tabla 59-2: Crear formulario basado en clases.

Nombre Tarea: Crear formulario basado en clases.	
Numero tarea: 1	Numero de historia: 3
Tipo de tarea: Desarrollo	Tiempo estimado: 0.1
Fecha inicio: 28/04/2016	Fecha fin: 28/04/2016
Programador responsable: Julio Duchitanga	
<p>Descripción: implementar una clase para crear automáticamente el formulario para validar y mostrar HTML mediante la librería llamada django.forms para el modelo beneficiarios. createBeneficiario(): instancia el modelo beneficiario para utilizar sus campos y utilizar en el formulario.</p>	

Fuente: DUCHITANGA, J., 2016

Tabla 60-2: Diseñar plantilla para ingresar, editar.

Nombre Tarea: Diseñar plantilla para ingresar, editar.	
Numero tarea: 2	Numero de historia: 3
Tipo de tarea: Desarrollo	Tiempo estimado: 0.1
Fecha inicio: 29/04/2016	Fecha fin: 29/04/2016
Programador responsable: Julio Duchitanga	
<p>Descripción: Diseñar una plantilla haciendo el lenguaje de plantillas de django para instanciar el formulario creado anteriormente para beneficiario, para cumplir esta tarea se utilizará librerías como bootstrap y CSS3 aplicados a etiquetas HTML para generar un aspecto amigable y funcional para el usuario final .</p> <p>Plantilla beneficiario: formulario para el registro y edición de beneficiarios. Además efectuar validación a nivel del front-end usando la librería JQuery.</p>	

Fuente: DUCHITANGA, J., 2016

Tabla 61-2: Crear patrón url y vista (funciones) Django.

Nombre Tarea: Crear patrón url y vista (funciones) Django.	
Numero tarea: 3	Numero de historia: 3
Tipo de tarea: Desarrollo	Tiempo estimado: 0.3
Fecha inicio: 30/04/2016	Fecha fin: 02/05/2016
Programador responsable: Julio Duchitanga	
<p>Descripción: Crear un patrón url dentro del archivo urls.py para acceder al formulario además las funciones necesarias para hacer el control del lado del servidor para el almacenamiento de la información; para ello implementar las siguientes funciones dentro del archivo views.py.</p> <p>url(r'^beneficiario/', views.crearBene,): patrón para acceder a formulario y enviar datos a la vista. def crearBene(): Recibe los datos mediante POST, los valida y guarda.</p>	

Fuente: DUCHITANGA, J., 2016

Tabla 62-2: Tareas de HU Asignar usuarios a un grupo.

HU4: Asignar usuarios a un grupo		
Número	Nombre de tarea	Tiempo(semanas)
1	Registrar grupos mediante la interfaz de administración	0.1
2	Asignar usuarios registrados a un grupo	0.4

Fuente: DUCHITANGA, J., 2016

Tabla 63-2: Registrar grupos mediante la interfaz de administración.

Nombre Tarea: Registrar grupos mediante la interfaz de administración.	
Numero tarea: 1	Numero de historia: 4
Tipo de tarea: Desarrollo	Tiempo estimado: 0.1
Fecha inicio: 03/05/2016	Fecha fin: 03/05/2016
Programador responsable: Julio Duchitanga	
<p>Descripción: Se registra los grupos a los que pertenecerán los usuarios del sistema para tener acceso a un perfil, para ello se hará uso de la interfaz de administración de Django que facilita este proceso.</p> <p>Estos grupos de usuario serán los siguientes: Posada: Para el acceso al módulo de posada y comedor. Social: Para el acceso al módulo de ayuda social. Capacitador: Para el acceso al módulo de capacitaciones. Técnico: Para el acceso al módulo de capacitaciones.</p>	

Fuente: DUCHITANGA, J., 2016

Tabla 64-2: Asignar usuarios registrados a un grupo.

Nombre Tarea: Asignar usuarios registrados a un grupo.	
Numero tarea: 2	Numero de historia: 4
Tipo de tarea: Desarrollo	Tiempo estimado: 0.4
Fecha inicio: 04/05/2016	Fecha fin: 06/05/2016
Programador responsable: Julio Duchitanga	
<p>Descripción: Una vez registrado un usuario el administrador tendrá la facultad de asignar al usuario a un grupo creado con anterioridad; un usuario solo podrá ser parte de un grupo.</p>	

Fuente: DUCHITANGA, J., 2016

Tabla 65-2: Tareas de HU Gestión de posadas.

HU5: Gestión de Posadas		
Número	Nombre de tarea	Tiempo(semanas)
1	Crear formulario basado en clases	0.1
2	Diseñar plantilla para crear o registrar posada.	0.2
3	Crear patrón url y vista (funciones) Django.	0.5
4	Listar posadas creadas.	0.2
5	Generar ficha de posada creada.	0.5

Fuente: DUCHITANGA, J., 2016

Tabla 66-2: Crear formulario basado en clases.

Nombre Tarea: Crear formulario basado en clases.	
Numero tarea: 1	Numero de historia: 5
Tipo de tarea: Desarrollo	Tiempo estimado: 0.1
Fecha inicio: 06/05/2016	Fecha fin: 06/05/2016
Programador responsable: Julio Duchitanga	
<p>Descripción: Implementar la clase ficha() para representar los atributos heredados del modelo Posada haciendo uso de la librería django.forms para crear automáticamente el formulario que validará y mostrará HTML; para ello se deberá contar con los siguientes prerrequisitos:</p> <p>Posada.models: Representación de la tabla Posada en la base de datos. Django.forms: Librería Django que crea automáticamente formularios HTML representando etiquetas mediante código Python. DateTimePicker: Librería para el manejo de fechas.</p>	

Fuente: DUCHITANGA, J., 2016

Tabla 67-2: Diseñar plantilla para crear o registrar posada.

Nombre Tarea: Diseñar plantilla para crear o registrar posada.	
Numero tarea: 2	Numero de historia: 5
Tipo de tarea: Desarrollo	Tiempo estimado: 0.2
Fecha inicio: 07/05/2016	Fecha fin: 08/05/2016
Programador responsable: Julio Duchitanga	
<p>Descripción: Haciendo uso de las librerías bootstrap, jquery sobre etiquetas HTML & CSS3 bajo el lenguaje de plantillas de Django que generará automáticamente el formulario gracias a la implementación del archivo forms.py, se diseñará el formulario de registro de una posada.</p> <p>Reservaficha.html: formulario para el registro de posadas. Además se implementará funciones jquery para la validación de campos del formulario. sumaFecha(): Asignar automáticamente la fecha de salida al recibir los días de estancia del posante.</p>	
Fuente: DUCHITANGA, J., 2016	

Tabla 68-2: Crear patrón url y vista (funciones) Django.

Nombre Tarea: Crear patrón url y vista (funciones) Django.	
Numero tarea: 3	Numero de historia: 5
Tipo de tarea: Desarrollo	Tiempo estimado: 0.5
Fecha inicio: 09/05/2016	Fecha fin: 13/05/2016
Programador responsable: Julio Duchitanga	
<p>Descripción: En esta tarea se creará el patrón url para acceder al formulario reservaficha.html y enlazar los datos devueltos de los formularios mediante POST a la vista reserva() del lado del servidor, para ello se deberá crear también la vista (función python).</p> <p>Def reserva(): Vista que se encargará de la validación y almacenamiento de posadas del lado del servidor.</p> <p>Como requisito se importa del archivo forms.py la clase ficha() para ligar el formulario a la vista.</p>	
Fuente: DUCHITANGA, J., 2016	

Tabla 69-2: Listar posadas creadas.

Nombre Tarea: Listar posadas creadas.	
Numero tarea: 4	Numero de historia: 5
Tipo de tarea: Desarrollo	Tiempo estimado: 0.2
Fecha inicio: 13/05/2016	Fecha fin: 14/05/2016
Programador responsable: Julio Duchitanga	
<p>Descripción: Crear la vista genérica posadaListView() haciendo uso de la librería django.views.generic.list que trae Django.</p> <p>posadaListView: Lista las posadas que se registran al momento.</p> <p>Activas.html: plantilla que mostrará las posadas con las siguientes funciones:</p> <ul style="list-style-type: none"> - Crear ficha (documento para legalizar posada) - Marcar salida. - Alimentación (registrar alimentación a posantes) 	
Fuente: DUCHITANGA, J., 2016	

Tabla 70-2: Generar ficha de posada creada.

Nombre Tarea: Generar ficha de posada creada.	
Numero tarea: 5	Numero de historia: 5
Tipo de tarea: Desarrollo	Tiempo estimado: 0.5
Fecha inicio: 14/05/2016	Fecha fin: 17/05/2016

Programador responsable: Julio Duchitanga
Descripción: Generar documento PDF como reporte para legalizar el registro de una posada, para ello se usará la librería Reportlab para el diseño del documento mediante código Python. def encabezado(): Para crear el encabezado del documento PDF. def pie(): Para crear el pie de página del documento PDF. def generarPdf(): Crear el cuerpo del documento con datos obtenidos mediante consultas a la base de datos haciendo uso del ORM del Django.
Fuente: DUCHITANGA, J., 2016

Tabla 71-2: Tareas de HU Seguimiento de entrada y salida de posantes.

HU6: Seguimiento de entrada y salida de posantes.		
Número	Nombre de tarea	Tiempo(semanas)
1	Crear patrón url y vista para salida de posante.	0.5

Fuente: DUCHITANGA, J., 2016

Tabla 72-2: Crear patrón url y vista para salida de posante.

Nombre Tarea: Crear patrón url y vista para salida de posante.	
Numero tarea: 1	Numero de historia: 6
Tipo de tarea: Desarrollo	Tiempo estimado: 0.5
Fecha inicio: 18/05/2016	Fecha fin: 20/05/2016
Programador responsable: Julio Duchitanga	
Descripción: proceso para implementar la vista que permita poner en estado inactivo una posada, además crear el patrón url para enlazar la plantilla creada con anterioridad (activas.html) con la vista. def salida: Función Python que permitirá poner una posada en estado inactivo.	

Fuente: DUCHITANGA, J., 2016

2.3.6.2. Tareas segunda iteración.

Tabla 73-2: Tareas de HU Gestión de alimentación diaria.

HU7: Gestión de alimentación diaria		
Número	Nombre de tarea	Tiempo(semanas)
1	Crear formulario basado en clases	0.1
2	Diseñar plantilla para crear o registrar alimentación.	0.1
3	Crear patrón url y vista (funciones) Django.	0.3

Fuente: DUCHITANGA, J., 2016

Tabla 74-2: Crear formulario basado en clases.

Nombre Tarea: Crear formulario basado en clases.	
Numero tarea: 1	Numero de historia: 7
Tipo de tarea: Desarrollo	Tiempo estimado: 0.1
Fecha inicio: 21/05/2016	Fecha fin: 21/05/2016
Programador responsable: Julio Duchitanga	
Descripción: Implementar las clases necesarias para registrar alimentación que permita representar los atributos de modelos del módulo de posada, haciendo uso de la librería django.forms para crear automáticamente el formulario que validará y mostrará HTML; para ello se deberá contar con los siguiente:	

<p>ComedorF.models: Representación de la tabla comedor funcionario en la base de datos. ComedorB.models: Representación de la tabla comedor beneficiario en la base de datos. Django.forms: Librería Django que crea automáticamente formularios HTML representando etiquetas mediante código Python.</p> <p>createComedorF(): Clase para alimentación de funcionarios que representa el formulario en HTML. createComedorB(): Clase para alimentación de beneficiarios que representa el formulario en HTML.</p> <p>Fuente: DUCHITANGA, J., 2016</p>

Tabla 75-2: Diseñar plantilla para crear o registrar alimentación.

Nombre Tarea: Diseñar plantilla para crear o registrar alimentación.	
Numero tarea: 2	Numero de historia: 7
Tipo de tarea: Desarrollo	Tiempo estimado: 0.1
Fecha inicio: 22/05/2016	Fecha fin: 22/05/2016
Programador responsable: Julio Duchitanga	
<p>Descripción: Haciendo uso de las librerías bootstrap, jquery sobre etiquetas HTML & CSS3 bajo el lenguaje de plantillas de Django que generará automáticamente el formulario gracias a la implementación de clases dentro del archivo forms.py, se diseñará el formulario de registro de alimentación:</p> <p>base.html: plantilla base que heredaran todas las plantillas siguientes. buscaB.html: plantilla para buscar si el beneficiario existe en la aplicación mediante AJAX. buscaF.html: plantilla para buscar si funcionario existe en la aplicación mediante AJAX. comedorB.html: formulario para registro de alimentación beneficiario. comedorF.html: formulario para registro de alimentación funcionario.</p> <p>Además crear un archivo llamado beneficiario.js para implementar código JavaScript para realizar la búsqueda de funcionarios y beneficiarios mediante AJAX.</p> <p>Fuente: DUCHITANGA, J., 2016</p>	

Tabla 76-2: Crear patrón url y vista (funciones) Django.

Nombre Tarea: Crear patrón url y vista (funciones) Django.	
Numero tarea: 3	Numero de historia: 7
Tipo de tarea: Desarrollo	Tiempo estimado: 0.3
Fecha inicio: 23/05/2016	Fecha fin: 25/05/2016
Programador responsable: Julio Duchitanga	
<p>Descripción: Crear los patrones urls para acceder a formularios de alimentación e interactuar con la información de los modelos mediante las vistas, para ello se creará vistas que permitan ejecutar acciones tanto para el registro y búsqueda de funcionarios y beneficiarios.</p> <p>def traer(): función que permite buscar un beneficiario recibe como id la cedula a través de la url; retorna un JSON. def detalleF(): función que permite buscar un funcionario, recibe como id la cedula a través de la url; retorna un JSON. def comedorB(): función que registra alimentación de beneficiarios; datos obtenidos a través de POST. def comedorF(): función que registra alimentación de funcionarios; datos obtenidos a través de POST. def pagar(): función que permite marca cancelado adeudos (alimentación) de funcionarios, recibe como parámetro la cedula del funcionario. class deudoresListView(): vista genérica ListView que permite listar deuda de un funcionario para ello recibe como parámetro la cédula.</p> <p>Fuente: DUCHITANGA, J., 2016</p>	

Tabla 77-2: Tareas de HU Gestión de ayudas sociales.

HU8: gestión de ayudas sociales		
Número	Nombre de tarea	Tiempo(semanas)
1	Crear formulario basado en clases	0.1
2	Diseñar plantilla para crear o registrar ayuda.	0.4
3	Crear patrón url y vista (funciones) Django.	0.5

Fuente: DUCHITANGA, J., 2016

Tabla 78-2: Crear formulario basado en clases.

Nombre Tarea: Crear formulario basado en clases.	
Numero tarea: 1	Numero de historia: 8
Tipo de tarea: Desarrollo	Tiempo estimado: 0.1
Fecha inicio: 26/05/2016	Fecha fin: 27/05/2016
Programador responsable: Julio Duchitanga	
<p>Descripción: Implementar las clases necesarias que permita representar los atributos de modelos del módulo de ayuda social, haciendo uso de la librería django.forms para crear automáticamente el formulario que validará y mostrará HTML; para ello se deberá contar o crear lo siguiente:</p> <p>Ayuda.models: representación de la tabla ayuda en la base de datos. Django.forms: librería Django que crea automáticamente formularios HTML representando etiquetas mediante código Python. class ayuda(): clase Python ayuda que representará etiquetas en el formulario HTML.</p>	

Fuente: DUCHITANGA, J., 2016

Tabla 79-2: Diseñar plantilla para crear o registrar ayuda.

Nombre Tarea: Diseñar plantilla para crear o registrar ayuda.	
Numero tarea: 2	Numero de historia: 8
Tipo de tarea: Desarrollo	Tiempo estimado: 0.4
Fecha inicio: 27/05/2016	Fecha fin: 29/05/2016
Programador responsable: Julio Duchitanga	
<p>Descripción: Haciendo uso de las librerías bootstrap, jquery sobre etiquetas HTML & CSS3 bajo el lenguaje de plantillas de Django que generará automáticamente el formulario gracias a la implementación de clases dentro del archivo forms.py, se diseñará el formulario de registro de alimentación, además se creará las plantillas necesarias para dar el funcionalidad la gestión de ayudas.</p> <p>base.html: plantilla base que heredaran todas las plantillas de ayudas. busca.html: plantilla para buscar si el beneficiario existe en la aplicación o si cuenta con un historial de ayudas recibidas mediante AJAX. ayuda.html: formulario para registro de ayudas sociales.</p> <p>Además se implementará un archivo llamado beneficiario.js con código JavaScript para realizar la búsqueda de beneficiarios y mostrar el detalle de recursos a entregar mediante AJAX.</p>	

Fuente: DUCHITANGA, J., 2016

Tabla 80-2: Crear patrón url y vista (funciones) Django.

Nombre Tarea: Crear patrón url y vista (funciones) Django.	
Numero tarea: 3	Numero de historia: 8
Tipo de tarea: Desarrollo	Tiempo estimado: 0.5
Fecha inicio: 29/05/2016	Fecha fin: 01/06/2016
Programador responsable: Julio Duchitanga	
<p>Descripción: Crear los patrones urls para acceder a formularios de ayudas e interactuar con la información de los modelos mediante las vistas, para ello se creará vistas que permitan ejecutar acciones tanto para el registro y búsqueda beneficiarios.</p> <p>def detalle(): función que permitirá buscar un beneficiario en la aplicación, recibe como id la cedula a través de la url; retorna un JSON.</p> <p>def busqueda(): función que permitirá buscar las ayudas que ha recibido un funcionario, recibe como id la cedula a través de la url; retorna un JSON.</p> <p>def ayudaS(): función que registrará ayudas de beneficiarios; datos obtenidos a través de POST.</p> <p>def arti(): función que retornará el detalle del recurso a entregar, recibe como parámetro el id del recurso y retorna un JSON.</p>	

Fuente: DUCHITANGA, J., 2016

Tabla 81-2: Tareas de HU Gestión de recursos.

HU9: gestión de recursos		
Número	Nombre de tarea	Tiempo(semanas)
1	Crear formulario basado en clases	0.1
2	Diseñar plantilla para crear o registrar recursos.	0.4
3	Crear patrón url y vista (funciones) Django.	0.5

Fuente: DUCHITANGA, J., 2016

Tabla 82-2: Crear formulario basado en clases.

Nombre Tarea: Crear formulario basado en clases.	
Numero tarea: 1	Numero de historia: 9
Tipo de tarea: Desarrollo	Tiempo estimado: 0.1
Fecha inicio: 02/06/2016	Fecha fin: 03/05/2016
Programador responsable: Julio Duchitanga	
<p>Descripción: Implementar las clases necesarias que permita representar los atributos de modelos del módulo de ayuda social, haciendo uso de la librería django.forms para crear automáticamente el formulario que validará y mostrará HTML; para ello se deberá contar o crear lo siguiente:</p> <p>Recurso.models: representación de la tabla recurso en la base de datos, para registrar recursos de ayudas sociales.</p> <p>Django.forms: librería Django que crea automáticamente formularios HTML representando etiquetas mediante código Python.</p> <p>class recurso(): clase Python recurso que representará atributos en etiquetas para el formulario HTML tomado del modelo recurso.</p>	

Fuente: DUCHITANGA, J., 2016

Tabla 83-2: Diseñar plantilla para crear o registrar recursos.

Nombre Tarea: Diseñar plantilla para crear o registrar recursos.	
Numero tarea: 2	Numero de historia: 9
Tipo de tarea: Desarrollo	Tiempo estimado: 0.4
Fecha inicio: 03/06/2016	Fecha fin: 05/05/2016
Programador responsable: Julio Duchitanga	

<p>Descripción: Haciendo uso de las librerías Bootstrap, Jquery sobre etiquetas HTML & CSS3 bajo el lenguaje de plantillas de Django que generará automáticamente el formulario gracias a la implementación de clases dentro del archivo forms.py, se diseñará el formulario de registro de recursos, además se creará las plantillas necesarias para dar el funcionalidad la gestión de recursos.</p> <p>base.html: plantilla base que heredaran todas las plantillas de recursos. inventario.html: plantilla que permitirá listar los recursos con stock cero, además presentará una interfaz para aumentar el stock de recursos. recurso.html: formulario para registro de recursos.</p> <p>Además se realizará validaciones del front-end de estas interfaces al capturar información haciendo uso de Jquery.</p> <p>Fuente: DUCHITANGA, J., 2016</p>
--

Tabla 84-2: Crear patrón url y vista (funciones) Django.

Nombre Tarea: Crear patrón url y vista (funciones) Django.	
Numero tarea: 3	Numero de historia: 9
Tipo de tarea: Desarrollo	Tiempo estimado: 0.5
Fecha inicio: 05/06/2016	Fecha fin: 08/06/2016
Programador responsable: Julio Duchitanga	
<p>Descripción: Crear los patrones urls para acceder a formularios de recursos e interactuar con la información de los modelos mediante las vistas, para ello se creará vistas que permitan ejecutar acciones tanto para el registro, detalle y aumentar stock de recursos. def detalle(): función que permitirá buscar un beneficiario en la aplicación, recibe como id la cedula a través de la url; retorna un JSON. def busqueda(): función que permitirá buscar las ayudas que ha recibido un funcionario, recibe como id la cedula a través de la url; retorna un JSON. def ayudaS(): función que registrará ayudas de beneficiarios; datos obtenidos a través de POST. def arti(): función que retornará el detalle del recurso a entregar, recibe como parámetro el id del recurso y retorna un JSON.</p> <p>Fuente: DUCHITANGA, J., 2016</p>	

2.3.6.3. Tareas tercera iteración.

Tabla 85-2: Tareas de HU Gestión de capacitaciones.

HU10: gestión de capacitaciones		
Número	Nombre de tarea	Tiempo(semanas)
1	Crear formulario basado en clases	0.1
2	Diseñar plantilla para crear o registrar capacitaciones.	0.4
3	Crear patrón url y vista (funciones) Django.	0.5

Fuente: DUCHITANGA, J., 2016

Tabla 86-2: Crear formulario basado en clases.

Nombre Tarea: Crear formulario basado en clases.	
Numero tarea: 1	Numero de historia: 10
Tipo de tarea: Desarrollo	Tiempo estimado: 0.1
Fecha inicio: 09/06/2016	Fecha fin: 10/06/2016
Programador responsable: Julio Duchitanga	
<p>Descripción: Implementar las clases necesarias que permita representar los atributos de modelos del módulo de capacitaciones, haciendo uso de la librería django.forms para crear automáticamente el formulario que validará y mostrará HTML; para ello se deberá contar o crear lo siguiente:</p>	

<p>Capacitacion.models: representación de la tabla capacitación en la base de datos, para registrar capacitaciones.</p> <p>Programa.models: representación de la tabla programa en la base de datos, para registrar los tipos de capacitaciones denominado programa.</p> <p>Django.forms: librería Django que crea automáticamente formularios HTML representando etiquetas mediante código Python.</p> <p>class capacita(): clase Python que representará atributos en etiquetas para el formulario HTML tomado del modelo capacitación; para ello se usará el lenguaje de plantillas Django.</p> <p>class programa(): clase Python que representará atributos en etiquetas para el formulario HTML tomado del modelo programa; para ello se usará el lenguaje de plantillas Django.</p> <p>Fuente: DUCHITANGA, J., 2016</p>
--

Tabla 87-2: Diseñar plantilla para crear o registrar capacitaciones.

Nombre Tarea: Diseñar plantilla para crear o registrar capacitaciones.	
Numero tarea: 2	Numero de historia: 10
Tipo de tarea: Desarrollo	Tiempo estimado: 0.4
Fecha inicio: 10/06/2016	Fecha fin: 12/05/2016
Programador responsable: Julio Duchitanga	
<p>Descripción: Haciendo uso de las librerías Bootstrap, Jquery sobre etiquetas HTML &CSS3 bajo el lenguaje de plantillas de Django que generará automáticamente el formulario gracias a la implementación de clases dentro del archivo forms.py, se diseñara el formulario de registro de capacitaciones, además se creará las plantillas necesarias para dar el funcionalidad la gestión de capacitaciones.</p> <p>base.html: plantilla base que heredaran todas las plantillas de capacitaciones.</p> <p>capacitacion.html: plantilla para el registro de capacitaciones.</p> <p>programa.html: plantilla para el registro de programas.</p> <p>listaCapTer.html: plantilla que listará las capacitaciones finalizadas.</p> <p>listaCapT.html: plantilla que listará las capacitaciones a ser publicadas.</p> <p>aprobados.html: plantilla que listará participantes que aprobaron una capacitación para generar certificado.</p> <p>Además se realizará validaciones del front-end de estas interfaces al capturar información haciendo uso de Jquery.</p> <p>Fuente: DUCHITANGA, J., 2016</p>	

Tabla 88-2: Crear patrón url y vista (funciones) Django.

Nombre Tarea: Crear patrón url y vista (funciones) Django.	
Numero tarea: 3	Numero de historia: 10
Tipo de tarea: Desarrollo	Tiempo estimado: 0.5
Fecha inicio: 12/06/2016	Fecha fin: 15/06/2016
Programador responsable: Julio Duchitanga	
<p>Descripción: Crear los patrones urls para acceder a formularios de capacitaciones e interactuar con la información de los modelos mediante las vistas, para ello se creará vistas que permitan ejecutar acciones tanto para el registro, listar, publicar y generar certificado de una capacitación además para el registro de programas.</p> <p>def capacitar(): función que permitirá registrar una capacitación en la aplicación.</p> <p>def programa(): función que permitirá registrar programas en la aplicación.</p> <p>class capacitacionesTListView(): vista genérica Django que listará capacitaciones registradas sin ser publicadas.</p> <p>class capacitacionesTerListView(): vista genérica Django que listará capacitaciones finalizadas.</p> <p>def publicar(): función que permitirá publicar capacitaciones.</p> <p>def aprobadosListView(): vista genérica Django que retornará una lista de todas las personas que aprobaron una capacitación; recibe como parámetro el id de la capacitación.</p> <p>def certificado(): función que generará certificado para participantes en formato PDF.</p> <p>Fuente: DUCHITANGA, J., 2016</p>	

Tabla 89-2: Tareas de HU Inscripción a capacitaciones.

HU11: inscripción a capacitaciones		
Número	Nombre de tarea	Tiempo(semanas)
1	Crear formulario basado en clases	0.1
2	Diseñar plantilla para inscripciones (registrar, aprobar personas).	0.4
3	Crear patrón url y vista (funciones) Django.	0.5

Fuente: DUCHITANGA, J., 2016

Tabla 90-2: Crear formulario basado en clases.

Nombre Tarea: Crear formulario basado en clases.	
Numero tarea: 1	Numero de historia: 11
Tipo de tarea: Desarrollo	Tiempo estimado: 0.1
Fecha inicio: 16/06/2016	Fecha fin: 17/06/2016
Programador responsable: Julio Duchitanga	
Descripción: Implementar las clases necesarias que permita representar los atributos de modelos del módulo de capacitaciones, haciendo uso de la librería django.forms para crear automáticamente el formulario que validará y mostrará HTML; para ello se deberá contar o crear lo siguiente:	
Inscripcion.models: representación de la tabla inscripción en la base de datos, para registrar personas a una capacitación.	
Django.forms: librería Django que crea automáticamente formularios HTML representando etiquetas mediante código Python.	
class registro(): clase Python que representará atributos en etiquetas para el formulario HTML tomado del modelo inscripción; para ello se usará el lenguaje de plantillas Django.	

Fuente: DUCHITANGA, J., 2016

Tabla 91-2: Diseñar plantilla para inscripciones.

Nombre Tarea: Diseñar plantilla para inscripciones.	
Numero tarea: 2	Numero de historia: 10
Tipo de tarea: Desarrollo	Tiempo estimado: 0.4
Fecha inicio: 17/06/2016	Fecha fin: 19/06/2016
Programador responsable: Julio Duchitanga	
Descripción: Haciendo uso de las librerías bootstrap, jquery sobre etiquetas HTML & CSS3 bajo el lenguaje de plantillas de Django que generará automáticamente el formulario gracias a la implementación de clases dentro del archivo forms.py, se diseñará el formulario de inscripciones, además se creará las plantillas para listar personas registradas.	
base.html: plantilla base que heredaran todas las plantillas de capacitaciones.	
registro.html: plantilla que permitirá registrar personas a capacitaciones.	
listapersonas.html: plantilla que mostrará personas inscritas en capacitaciones con las siguientes funcionalidades: aprobar o reprobar.	
listacap.html: plantilla que mostrará capacitaciones a realizarse, además permitirá funcionalidades como botón para inscripción, listar personas y terminar capacitación.	
aprobados.html: plantilla que listará participantes que aprobaron una capacitación para generar certificado.	
Además se realizará validaciones del front-end de estas interfaces al capturar información haciendo uso de JQuery.	

Fuente: DUCHITANGA, J., 2016

Tabla 92-2: Crear patrón url y vista (funciones) Django.

Nombre Tarea: Crear patrón url y vista (funciones) Django.	
Numero tarea: 3	Numero de historia: 10
Tipo de tarea: Desarrollo	Tiempo estimado: 0.5
Fecha inicio: 19/06/2016	Fecha fin: 22/06/2016
Programador responsable: Julio Duchitanga	
<p>Descripción: Crear los patrones urls para acceder a las plantillas e interactuar con la información de los modelos mediante las vistas, para ello se creará vistas que permitan ejecutar acciones tanto para inscripción, listar personas y capacitaciones, aprobar o reprobar personas, terminar capacitaciones.</p> <p>def aprobar(): función que permitirá aprobar una capacitación; recibirá como parámetro la cedula dela persona.</p> <p>def reprobar(): función que permitirá reprobar una capacitación; recibira como parámetro la cedula dela persona.</p> <p>class personasListView(): vista genérica Django que retornará una lista de personas registradas en una capacitación para ello recibirá como parámetro el id de la capacitación.</p> <p>class capacitacionesListView(): vista genérica Django que retornará una lista de capacitaciones activas a ser desarrolladas.</p> <p>def terminar(): función que permitirá dar por finalizado una capacitación.</p> <p>def registro(): función que realizará la inscripción de personas a capacitaciones.</p>	

Fuente: DUCHITANGA, J., 2016

Tabla 93-2: Tareas de HU Generar reportes.

HU12: generar reportes		
Número	Nombre de tarea	Tiempo(semanas)
1	Instalar y configurar la librería Reportlab en Django.	0.1
2	Crear formulario basado en clases para reportes con parámetros.	0.1
2	Diseñar plantilla Django para reportes simples y con parámetros.	0.3
3	Crear patrón url y vista Django para reportes con o sin parámetros.	0.5

Fuente: DUCHITANGA, J., 2016

Tabla 94-2: Instalar y configurar la librería Reportlab en Django.

Nombre Tarea: Instalar y configurar la librería Reportlab en Django.	
Numero tarea: 1	Numero de historia: 12
Tipo de tarea: Desarrollo	Tiempo estimado: 0.1
Fecha inicio: 23/06/2016	Fecha fin: 24/06/2016
Programador responsable: Julio Duchitanga	
<p>Descripción: Descargar la librería Reportlab gratuita de la página oficial e instalar sobre Django. Posteriormente configurar el archivo setting.py de la aplicación web.</p>	

Fuente: DUCHITANGA, J., 2016

Tabla 95-2: Crear formulario basado en clases.

Nombre Tarea: Crear formulario basado en clases.	
Numero tarea: 2	Numero de historia: 12
Tipo de tarea: Desarrollo	Tiempo estimado: 0.1
Fecha inicio: 24/06/2016	Fecha fin: 25/06/2016
Programador responsable: Julio Duchitanga	
Descripción: Implementar las clases necesarias haciendo uso de la librería django.forms para crear automáticamente el formulario que validará y mostrará HTML para los reportes con parámetros. Las clases se crearán para cada módulo de SIGEGAS, dependiendo de los reportes solicitados.	
Fuente: DUCHITANGA, J., 2016	

Tabla 96-2: Diseñar plantilla Django para reportes simples y con parámetros.

Nombre Tarea: Diseñar plantilla Django para reportes simples y con parámetros.	
Numero tarea: 3	Numero de historia: 12
Tipo de tarea: Desarrollo	Tiempo estimado: 0.3
Fecha inicio: 25/06/2016	Fecha fin: 28/06/2016
Programador responsable: Julio Duchitanga	
Descripción: Haciendo uso de las librerías Bootstrap, JQuery sobre etiquetas HTML & CSS3 bajo el lenguaje de plantillas de Django que generará automáticamente el formulario gracias a la implementación de clases dentro del archivo forms.py, se diseñara los formularios para reportes. base.html: plantilla base que heredaran todas las plantillas de reportes. reporteparametros.html: plantilla que permitirá recoger datos para hacer la consulta y generar reporte. reporte.html: plantilla para reportes sin parámetros. Este proceso o tarea se efectuara para los tres módulos de la aplicación web SIGEGAS.	
Fuente: DUCHITANGA, J., 2016	

Tabla 97-2: Crear patrón url y vista (funciones) Django.

Nombre Tarea: Crear patrón url y vista (funciones) Django.	
Numero tarea: 4	Numero de historia: 12
Tipo de tarea: Desarrollo	Tiempo estimado: 0.5
Fecha inicio: 29/06/2016	Fecha fin: 04/07/2016
Programador responsable: Julio Duchitanga	
Descripción: Crear los patrones urls para acceder a las plantillas e interactuar con la información de los modelos mediante las vistas, para ello se creará vistas que permitan realizar consultas a la base de datos para generar los reportes. def certificado(): función Python que interactúa con la librería Reportlab y canvas para generar un certificado de capacitaciones en formato PDF. def ficha (): función Python que interactúa con la librería Reportlab y canvas para generar una ficha de una posada en formato PDF. Las vistas se crearan para los tres módulos de la aplicación web para realizar las consultas y retornar información a las plantillas para generar reportes en formato PDF.	
Fuente: DUCHITANGA, J., 2016	

2.4. Fase de Codificación

2.4.1. Estándar de diseño de base de datos

El estándar seguido en esta etapa de desarrollo se lo realizó para evitar conflictos con los estándares que predica el framework Django para los modelos y así ganar tiempo en desarrollo.

2.4.1.1. Nombre de entidades

Para nombrar las entidades de la base de datos se manejó de la siguiente manera: el nombre de acuerdo a su propósito, primera letra mayúscula y pluralizada. Para entidades combinadas se tomó el nombre de cada entidad separado por un guion bajo (_).

Ejemplo:

- Beneficiarios (nombre de entidad).
- Grupo_Beneficiarios (nombre de entidad combinada).

2.4.1.2. Nombre de atributos

Para nombrar los atributos de cada entidad se tomó de la siguiente manera:

El nombre en minúsculas de acuerdo a sus propósitos, para nombres compuesto por más de una palabra se las une con un guion bajo (_) para evitar espacios. Para nombrar llaves primarias y foráneas se escribió las tres primeras letras de la entidad contenedora seguido por las tres primeras letras del atributo unidos por un guion bajo (_).

Ejemplo:

- apellido (nombre de atributo).
- estado_civil (nombre de atributo compuesto por dos palabras).
- ben_ced (nombre de llave primaria).

2.4.2. Estándar de codificación

Para mejorar la legibilidad del código se ha optado un estándar que facilite y optimice la programación de la aplicación.

Clases y funciones

Los nombres de las clases y las funciones están compuestas de la siguiente forma: nombre en minúscula, si contiene más de una palabra la siguiente palabra debe iniciar con mayúscula sin espacios entre palabras. El nombre debe ser acorde a la tarea que realiza.

Ejemplo:

- `posadaListView` (clase)
- `salida` (función)

Variables

El nombre de estas debe ser minúsculas, si contiene más de una palabra la siguiente palabra debe iniciar con mayúscula sin espacios entre palabras. Los nombres no necesariamente deben estar acorde a la tarea o acción que realiza.

2.4.2.1. Tipos de archivos

A continuación en la tabla 98-2 se muestra las extensiones de archivos que se utiliza en el desarrollo de la aplicación basada en el framework Django, para una mejor utilización.

Tabla 98-2: Extensiones de archivos.

Archivo	Extensión	Ejemplo
Python	.py	Views.py
Plantillas	.html	Index.html
Estilos	.css	Base.css
Java Scrip	.js	stock.js

Fuente: DUCHITANGA, J., 2016

2.4.3. SIGEGAS

La aplicación web está desarrollada sobre Django framework de Python, para el desarrollo ágil de la aplicación, está compuesta por tres módulos principales y uno complementario. Además cada módulo sigue el patrón de desarrollo MTV que fomenta Django. A continuación en la figura 61-2 se muestra la estructura general de la aplicación.

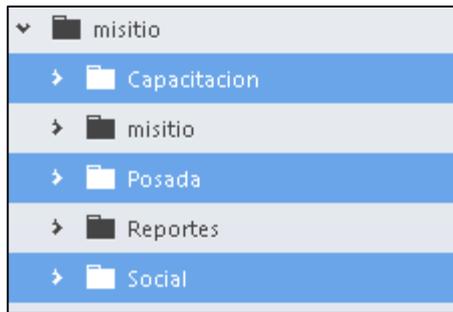


Figura 60-2 Estructura SIGEGAS.
Fuente: DUCHITANGA, J., 2016

Los principales módulos de SIGEGAS son: capacitación, posada y social cada uno de estos sigue e patrón MTV con si fuera un módulo independiente. En la figura 62-2 se muestra la estructura de los módulos de la aplicación.

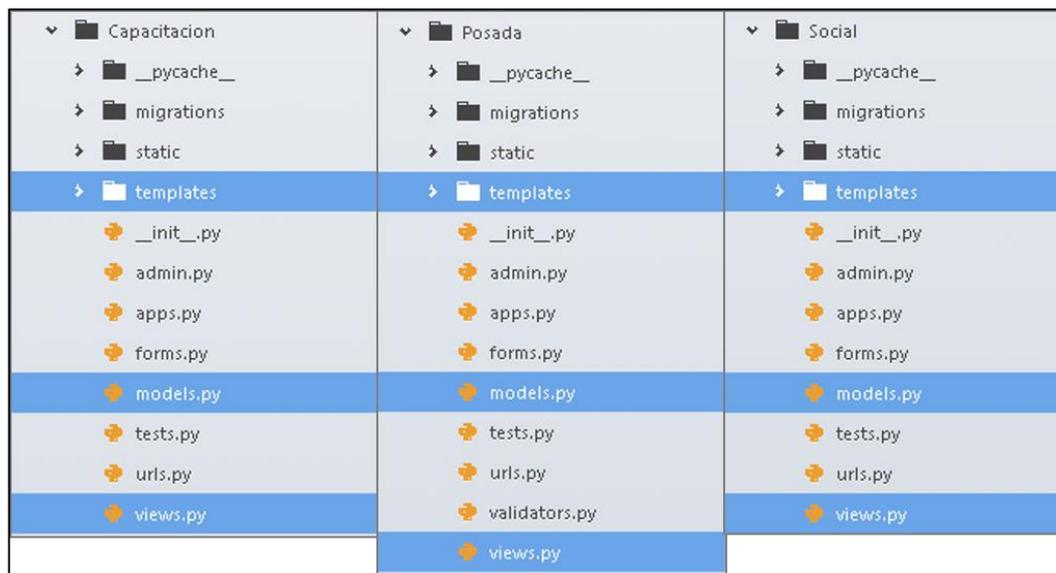


Figura 61-2 Estructura de módulos SIGEGAS.
Fuente: DUCHITANGA, J., 2016

2.4.3.1. Modelos (M)

La capa de datos; aquí se encuentra los modelos que representa cada entidad de la base de datos PostgreSQL. Para ello Django utiliza el archivo `models.py` para representar las entidades mediante clases sin utilizar código SQL.

2.4.3.2. Vistas (V)

La capa de negocio; dentro de esta capa están las vistas que contienen clases y funciones necesarias para proporcionar la iteración con los modelos y presentar la lógica necesaria para

interactuar con las plantillas a través de las urls, tanto para receptor y responder solicitudes. Django utiliza dos archivos Python para controlar este proceso: views.py y urls.py.

2.4.3.3. Plantillas (T)

La capa de presentación; aquí se encuentra todas los archivos con extensión .html que se utiliza para para la recepción y envío de información hacia el servidor, y toda la información necesaria para el diseño y validación del información del lado del cliente.

Para mayor detalle de la estructura de la aplicación web y su contenido como: clases, funciones, plantillas, archivos complementarios y configuraciones diríjase al proyecto que contiene la aplicación que se encuentra en el cd.

2.4.4. Diagrama de despliegue

En la figura 63-2 se visualiza el diagrama de despliegue de la aplicación web.

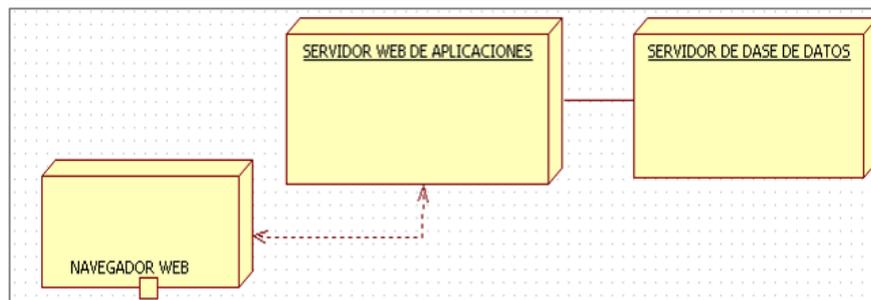


Figura 62-2 Diagrama de despliegue.

Fuente: DUCHITANGA, J., 2016

2.4.5. Diagrama de componentes

En la figura 64-2 se visualiza el diagrama de componentes de la aplicación web.

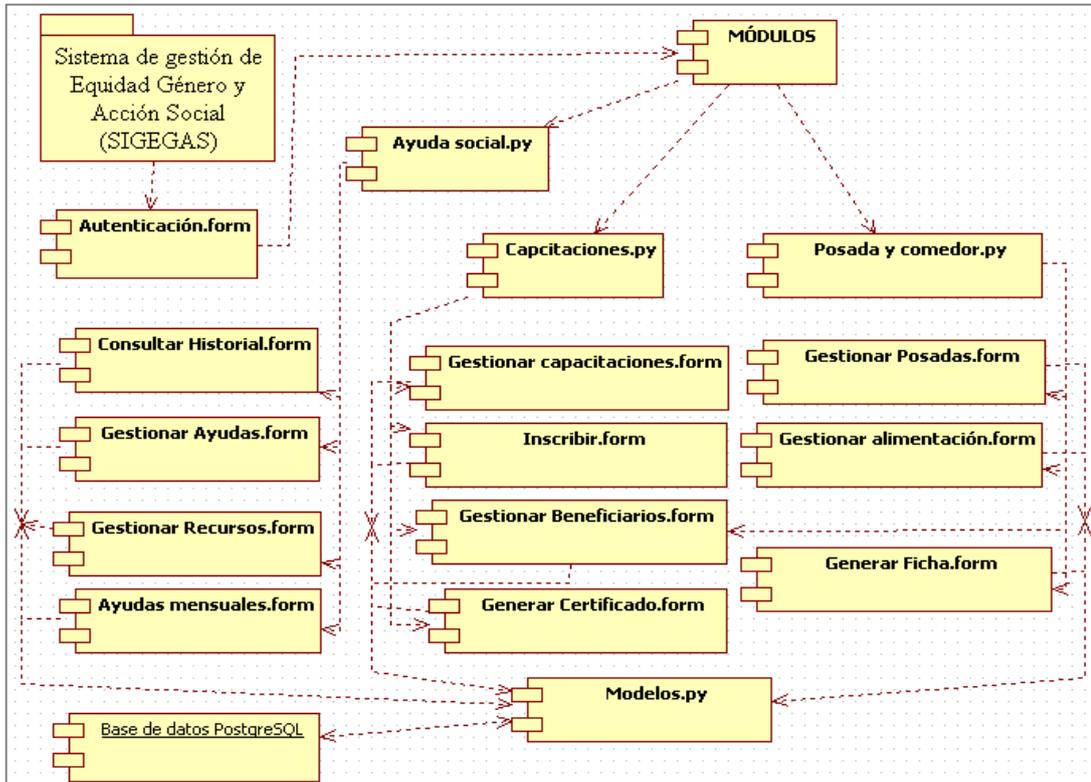


Figura 63-2 Diagrama de componentes.

Fuente: DUCHITANGA, J., 2016

2.5. Fase de Pruebas

2.5.1. Pruebas primera iteración

Historia de usuario 1: Acceso al sistema.

Registro de información correcta

- **Descripción:**

Una vez que el usuario ingresa a la página de login debe llenar los campos que solicita el formulario de ingreso, los datos serán validados internamente para permitir el acceso al sistema.

- **Condiciones de ejecución:**

El usuario debe haber sido registrado y asignado roles para ingresar a un perfil del sistema.

- **Entrada:**

Usuario y contraseña

- **Resultado esperado:**

Una vez ingresado los datos solicitados, y la validación de los mismos es correcta accede al sistema.

- **Evaluación de prueba:**

Prueba satisfactoria

Registro de información incorrecta

- **Descripción:**

Si los datos del usuario ingresados en los formularios de login no coinciden, el sistema presentará alertas que describen el error cometido.

- **Condiciones de ejecución:**

Los datos ingresados por el usuario deben ser incorrectos.

- **Entrada:**

Usuario y contraseña

- **Resultado esperado:**

Si no existe coincidencia entre los datos del sistema y los ingresados no accederán al sistema.

- **Evaluación de prueba:**

Prueba satisfactoria

Historia de usuario 2: Gestión de usuarios.

Registro de información correcta

- **Descripción:**

El administrador es el único usuario que puede registrar, actualizar o eliminar usuario, para ello dentro de su perfil seleccionará 'agregar usuario' y aparecerá un formulario en el cual el usuario debe ingresar nombre de usuario, contraseña y confirmar contraseña; todos los campos solicitados por los formularios son obligatorios.

- **Condiciones de ejecución:**

El usuario debe permanecer al grupo administrador para registrar usuarios.

- **Entrada:**

- Una vez seleccionado 'agregar usuario' para registrar un usuario.
- El administrador deberá ingresar los siguientes campos: nombre de usuario máximo 30 caracteres, contraseña y confirmar contraseña.
- El validará los datos ingresados para proceder a registrar.

- **Resultado esperado:**

Si los datos ingresados son validados correctamente por el sistema se registra un nuevo usuario.

- **Evaluación de prueba:**

Prueba satisfactoria.

Registro de información incorrecta

- **Descripción:**

El administrador es el único usuario que puede registrar, actualizar o eliminar usuario, para ello dentro de su perfil seleccionará 'agregar usuario' y aparecerá un formulario en el cual el usuario deberá ingresar nombre de usuario, contraseña y confirmar contraseña; el sistema validará los datos de presentar errores emitirá alertas que permitirán identificar el error.

- **Condiciones de ejecución:**

El administrador deberá ingresar datos incorrectos.

- **Entrada:**

- Una vez seleccionado 'agregar usuario' para registrar un usuario.
- El administrador deberá ingresar los siguientes campos: nombre de usuario máximo 30 caracteres, contraseña mínimo 8 caracteres y confirmar contraseña, el sistema validará la información, de presentar errores mostrará mensajes de alerta con el error cometido.

- **Resultado esperado:**

El formulario de registro de usuario presenta mensajes de alerta de no cumplirse con las condiciones para realizar un registro satisfactorio como: la coincidencia entre contraseñas o el número de caracteres de los campos no se registrara al usuario en el sistema.

- **Evaluación de prueba:**

Prueba satisfactoria.

Historia de usuario 3: Gestión de beneficiarios.

Registro de información correcta

- **Descripción:**

Los usuarios que dispongan de acceso al sistema tendrán los permisos para registrar beneficiarios, para ello luego de haber seleccionado ‘registrar’ se mostrara un formulario con los campos requeridos sobre beneficiarios.

- **Condiciones de ejecución:**

El usuario que registra un beneficiario deberá pertenecer a un grupo de usuario (permisos) preestablecido.

- **Entrada:**

- Una vez seleccionado ‘registrar’ para registrar un beneficiario.
- El usuario deberá ingresar los siguientes campos: cédula (no admite guion), nombre, apellido, edad, instrucción, ocupación, estado económico, estado de salud, etnia, genero, procedencia, discapacidad, porcentaje y grupo de atención.
- El sistema validará los datos ingresados para proceder a registrar.

- **Resultado esperado:**

Si los datos ingresados son validados correctamente y el beneficiario no está registrado entonces el sistema registrará un nuevo beneficiario.

- **Evaluación de prueba:**

Prueba satisfactoria.

Registro de información incorrecta

- **Descripción:**

Los usuarios que dispongan de acceso al sistema tendrán los permisos para registrar beneficiarios, para ello luego de haber seleccionado ‘registrar’ se mostrara un formulario con los

campos requeridos sobre beneficiarios; el sistema mostrará mensajes de alerta en caso de existir errores.

- **Condiciones de ejecución:**

La información proporcionada deberá ser incorrecta.

- **Entrada:**

- Una vez seleccionado 'registrar' para registrar un beneficiario.
- El usuario deberá ingresar los siguientes campos: cédula (no admite guion), nombre, apellido, edad, instrucción, ocupación, estado económico, estado de salud, etnia, genero, procedencia, discapacidad, porcentaje y grupo de atención.
- El sistema validará los datos; si existe errores presentara mensajes de alerta al usuario.

- **Resultado esperado:**

Si los datos ingresados no son válidos o el beneficiario existe en el sistema, se mostrara mensajes de alerta con los inconvenientes presentados y no se registrará el beneficiario.

- **Evaluación de prueba:**

Prueba satisfactoria.

Historia de usuario 4: Asignar usuarios a un grupo.

Registro de información correcta

- **Descripción:**

Los usuarios pueden pertenecer a un solo grupo de usuarios, para ello el administrador seleccionará de la lista de usuarios registrados a un usuario para asignarle a un grupo.

- **Condiciones de ejecución:**

El usuario deberá tener permisos de administrador para asignar grupos a beneficiarios.

- **Entrada:**

- Del menú de administración el usuario deberá seleccionar 'usuarios' para listar los usuarios registrados.
- El administrador deberá seleccionar un usuario y llenar el siguiente campo: grupos (seleccionar un grupo).
- El sistema validará los datos ingresados y actualizara la información del usuario.

- **Resultado esperado:**

Si la asignación a un grupo fue validado correctamente, se actualizara la información del usuario y tendrá acceso al sistema de acuerdo a los permisos de cada grupo.

- **Evaluación de prueba:**

Prueba satisfactoria.

Registro de información incorrecta

- **Descripción:**

Los usuarios pueden pertenecer a un solo grupo de usuarios, para ello el administrador seleccionará de la lista de usuarios registrados a un usuario para asignarle a un grupo. Si la asignación no es correcta el sistema mostrará mensaje de alerta para informar del error existente.

- **Condiciones de ejecución:**

La información proporcionada deberá ser incorrecta.

- **Entrada:**

- Del menú de administración el usuario deberá seleccionar ‘usuarios’ para listar los usuarios registrados.
- El administrador deberá seleccionar un usuario y llenar el siguiente campo: grupos (seleccionar un grupo); el sistema validará los datos ingresados.

- **Resultado esperado:**

Si la validación es incorrecta mostrara mensajes de alerta para informar del error al administrador, como consecuencia el usuario no tendrá acceso al sistema.

- **Evaluación de prueba:**

Prueba satisfactoria.

Historia de usuario 5: Gestión de posada.

Registro de información correcta

- **Descripción:**

El usuario debe tener acceso al módulo de posada el cual tendrá la facultad de crear posadas para beneficiarios registrados o no registrados en el sistema.

- **Condiciones de ejecución:**

El usuario deberá tener permiso para acceder al módulo de posada.

- **Entrada:**

- Del menú de posada el usuario deberá seleccionar registrar y luego ‘posada/almuerzo’ y se mostrara un formulario para realizar una búsqueda de beneficiario.
- El usuario ingresará la siguiente información: Cédula (no se acepta guion, solo números, máximo 10 caracteres).
- El sistema mostrará el detalle del beneficiario y seleccionar ‘crear posada’ y llenar los siguientes campos del formulario de posada: beneficiario (automático), días (numero), tipo

posada (automático), fecha salida (automático), hora salida (predefinido con opción a cambio), habitación, motivo y documento.

- Sistema validará los datos y registrará la posada.

- **Resultado esperado:**

Si la validación fue exitosa se creará la posada, el sistema mostrará un mensaje de éxito y mostrará en una lista la posada creada al usuario con opción a generar una ficha en formato PDF imprimible a ser legalizada.

- **Evaluación de prueba:**

Prueba satisfactoria.

Registro de información incorrecta

- **Descripción:**

El usuario deberá tener acceso al módulo de posada el cual tendrá la facultad de crear posadas para beneficiarios registrados o no registrados en el sistema. El sistema mostrara mensajes de alerta (errores o éxito) en cada uno de los procesos efectuados.

- **Condiciones de ejecución:**

El usuario deberá ingresar información errónea, además el beneficiario tendrá que estar registrado previamente.

- **Entrada:**

- Del menú de posada el usuario deberá seleccionar registrar y luego 'posada/almuerzo' y se mostrara un formulario para realizar una búsqueda de beneficiario.

- Para realizar la búsqueda del beneficiario el usuario deberá ingresa la siguiente información: Cédula (no se acepta guion, solo números, máximo 10 caracteres).

- El sistema mostrará el detalle del beneficiario y seleccionar 'crear posada' y llenar los siguientes campos del formulario de posada: beneficiario (automático), días (numero), tipo posada (automático), fecha salida (automático), hora salida (predefinido con opción a cambio), habitación, motivo y documento.

- **Resultado esperado:**

Si la validación es incorrecta no se creará la posada, el sistema mostrará un mensaje de alerta (error) y no presentara ningún registro en pantalla (no se podrá generar la ficha).

- **Evaluación de prueba:**

Prueba satisfactoria.

Historia de usuario 6: seguimiento de entrada y salida de posantes.

Registro de información correcta

- **Descripción:**

El usuario con permisos para acceder al módulo de posada, tendrá la facultad de ver la lista de posadas registradas en pantalla, con opción a marcar la salida del posante cuando este haya culminado el tiempo de estadía; para ello deberá seleccionar 'salida' de la lista de posantes.

- **Condiciones de ejecución:**

El usuario deberá tener permiso para acceder al módulo de posada, deben existir posadas registradas.

- **Entrada:**

- En el menú principal seleccionar una posada y pulsar en 'salida'.
- El sistema validará y modificará el estado de la posada.

- **Resultado esperado:**

La posada marcada como salida se deberá quitar de la lista de posadas en pantallas y mostrará un mensaje de éxito.

- **Evaluación de prueba:**

Prueba satisfactoria.

Registro de información incorrecta

- **Descripción:**

El usuario con permisos para acceder al módulo de posada, tendrá la facultad de ver la lista de posadas registradas en pantalla, con opción a marcar la salida del posante cuando este haya culminado el tiempo de estadía; para ello deberá seleccionar 'salida' de la lista de posantes. Además el sistema mostrara mensajes de alerta (error) que permitirán al usuario estar informado.

- **Condiciones de ejecución:**

El usuario no deberá marcar la salida luego o antes de la fecha de culminación de estadía.

- **Entrada:**

- En el menú principal seleccionar una posada y pulsar en 'salida'.
- El sistema validará y modificará el estado de la posada.

- **Resultado esperado:**

El sistema alertará al usuario mediante mensajes, todas las posadas que sobrepasaron el tiempo de estadía y que deben ser marcadas como salida.

- **Evaluación de prueba:**

Prueba satisfactoria.

2.5.2. *Pruebas segunda iteración*

Historia de usuario 7: gestión de alimentación diaria.

Registro de información correcta

- **Descripción:**

El usuario con acceso al módulo de posada el cual tendrá la facultad de gestionar alimentación para beneficiarios y funcionarios registrados o no registrados en el sistema. Para ello en primera instancia deberá realizar una búsqueda para verificar la existencia de los mismos el sistema de lo contrario se procederá a registrarlos; para poder efectuar el registro de alimentación.

- **Condiciones de ejecución:**

El usuario deberá tener permiso para acceder al módulo de posada, el beneficiario y el funcionario deben estar registrados con anterioridad.

- **Entrada (beneficiario):**

- Del menú de posada el usuario deberá seleccionar registrar y luego 'posada/almuerzo' y se mostrara un formulario para realizar una búsqueda de beneficiario.
- Para realizar la búsqueda del beneficiario el usuario deberá ingresa la siguiente información: Cédula (no se acepta guion, solo números, máximo 10 caracteres).
- El sistema mostrará el detalle del beneficiario y seleccionar 'registrar almuerzo' y llenar los siguientes campos del formulario de alimentación: beneficiario (automático), categoría (opcional, permanente, posante), almuerzo (automático), observación (campo no obligatorio).
- El sistema validará los datos y almacenara el registro de alimentación.
- El proceso se considera como finalizado.

- **Entrada (funcionarios):**

- Del menú de posada el usuario deberá seleccionar comedor y luego 'funcionarios', se mostrara un formulario para realizar una búsqueda de funcionarios.
- Para realizar la búsqueda deberá ingresa la siguiente información: Cédula (no se acepta guion, solo números, máximo 10 caracteres).
- El sistema mostrará el información del funcionario y seleccionar 'registrar almuerzo' y llenar los siguientes campos del formulario de alimentación: funcionario (automático), almuerzo (completo, seco y jugo, sopa), costo (automático al seleccionar el almuerzo), pagado (seleccionar si fue cancelado el almuerzo) y observación (campo opcional).

- El sistema validará los datos y mostrara mensajes de alerta (error).
- El proceso se considera como finalizado.

- **Resultado esperado:**

Si la validación de los datos del formulario de alimentación es correcta se almacenara el registro en la base de datos y el sistema mostrara un mensaje de éxito al usuario.

- **Evaluación de prueba:**

Prueba satisfactoria.

Registro de información incorrecta

- **Descripción:**

El usuario con acceso al módulo de posada el cual tendrá la facultad de gestionar alimentación para beneficiarios y funcionarios registrados o no registrados en el sistema. Para ello en primera instancia deberá realizar una búsqueda para verificar la existencia de los mismos el sistema de lo contrario se procederá a registrarlos; para poder efectuar el registro de alimentación. El sistema por su parte validara la información y mostrará mensaje de alerta (error) para mantener informado al usuario.

- **Condiciones de ejecución:**

El usuario deberá tener permiso para acceder al módulo de posada, el beneficiario y el funcionario deben estar registrados con anterioridad.

- **Entrada (beneficiario):**

- Del menú de posada el usuario deberá seleccionar registrar y luego ‘posada/almuerzo’ y se mostrara un formulario para realizar una búsqueda de beneficiario.
- Para realizar la búsqueda del beneficiario el usuario deberá ingresa la siguiente información: Cédula (no se acepta guion, solo números, máximo 10 caracteres).
- El sistema mostrará el detalle del beneficiario y seleccionar ‘registrar almuerzo’ y llenar los siguientes campos del formulario de alimentación: beneficiario (automático), categoría (opcional, permanente, posante), almuerzo (automático), observación (campo no obligatorio).
- El sistema validará los datos y mostrara mensajes de alerta (error).
- El proceso se considera como finalizado.

- **Entrada (funcionarios):**

- Del menú de posada el usuario deberá seleccionar comedor y luego ‘funcionarios’, se mostrara un formulario para realizar una búsqueda de funcionarios.
- Para realizar la búsqueda deberá ingresa la siguiente información: Cédula (no se acepta guion, solo números, máximo 10 caracteres).

- El sistema mostrará el información del funcionario y seleccionar ‘registrar almuerzo’ y llenar los siguientes campos del formulario de alimentación: funcionario (automático), almuerzo (completo, seco y jugo, sopa), costo (automático al seleccionar el almuerzo), pagado (seleccionar si fue cancelado el almuerzo) y observación (campo opcional).
- El sistema validará los datos y mostrara mensajes de alerta (error).
- El proceso se considera como finalizado.

- **Resultado esperado:**

Cuando los datos son incorrectos no se procederá al registro de alimentación, tanto para beneficiarios como para funcionarios, el sistema informara del error al usuario mediante mensajes de alerta (error).

- **Evaluación de prueba:**

Prueba satisfactoria.

Historia de usuario 8: gestión de ayudas sociales.

Registro de información correcta

- **Descripción:**

El usuario con acceso al módulo de ayuda social tendrá la facultad de gestionar ayudas a beneficiarios. Para ello en primera instancia deberá realizar una búsqueda para verificar la existencia del mismo en el sistema de lo contrario se procederá a registrarlo; para poder efectuar un registro de ayuda. Además el sistema permitirá llevar un seguimiento de ayudas entregadas a una determinada persona.

- **Condiciones de ejecución:**

El usuario deberá tener permiso para acceder al módulo de ayuda social, el beneficiario deberá estar registrado con anterioridad. Además tiene que existir recursos (bines) registrados a entregar el la ayuda.

- **Entrada:**

- Dentro del menú principal se deberá realizar la búsqueda del beneficiario para ello seleccionar ‘buscar’, el usuario deberá ingresa la siguiente información: Cédula (no se acepta guion, solo números, máximo 10 caracteres).
- El sistema mostrará información de la persona y seleccionar ‘crear ayuda y llenar los siguientes campos del formulario de ayudas: beneficiario (automático), ayuda, cantidad y observación (campo opcional).
- El sistema validará los datos y almacenara el registro de alimentación.
- El proceso se considera como finalizado.

- **Resultado esperado:**

Durante el proceso de registro de ayudas, si la validación del sistema es correcta; se registrará la ayuda y mostrará un mensaje de éxito al usuario.

- **Evaluación de prueba:**

Prueba satisfactoria.

Registro de información incorrecta

- **Descripción:**

El usuario con acceso al módulo de ayuda social tendrá la facultad de gestionar ayudas a beneficiarios. Para ello en primera instancia deberá realizar una búsqueda para verificar la existencia del mismo en el sistema de lo contrario se procederá a registrarlo; para poder efectuar un registro de ayuda. Además el sistema permitirá llevar un seguimiento de ayudas entregadas a una determinada persona. La información proporcionada es incorrecta el sistema mostrará mensajes de alerta informando del error cometido.

- **Condiciones de ejecución:**

El usuario encargado de ayudas sociales deberá proporcionar información incorrecta al sistema.

- **Entrada:**

- Dentro del menú principal se deberá realizar la búsqueda del beneficiario para ello seleccionar 'buscar', el usuario tendrá que ingresar la siguiente información: Cédula (no se acepta guion, solo números, máximo 10 caracteres).
- El sistema mostrará información de la persona y seleccionar 'crear ayuda y llenar los siguientes campos del formulario de ayudas: beneficiario (automático), ayuda, cantidad y observación (campo opcional).
- El sistema validará los datos y visualizará un mensaje de alerta (error).
- El proceso se considera como finalizado.

- **Resultado esperado:**

Si la información de registro de ayudas no es válida, el sistema no almacenará la información en la base de datos, informando al usuario del error mediante mensajes de alerta.

- **Evaluación de prueba:**

Prueba satisfactoria.

Historia de usuario 9: gestión de recursos

Registro de información correcta de recursos

- **Descripción:**

El usuario con acceso al módulo de ayuda social tendrá la facultad de gestionar recursos para ayudas sociales. Se realizará el respectivo registro y modificación de recursos.

- **Condiciones de ejecución:**

El usuario deberá tener permiso para acceder al módulo de ayuda social.

- **Entrada:**

- Dentro del menú principal seleccionar 'recurso', posteriormente 'nuevo' se presentara un formulario donde se deberá llenar los siguientes campos del formulario: categoría, artículo, detalle, stock, observación (no obligatorio).
- El sistema validará los datos y almacenará la información en la base de datos.
- El proceso se considera como finalizado.

- **Resultado esperado:**

Debe almacenarse los datos de recurso, si toda la información cumple con lo establecido.

- **Evaluación de prueba:**

Prueba satisfactoria.

Registro de información incorrecta de recursos

- **Descripción:**

El usuario con acceso al módulo de ayuda social tendrá la facultad de gestionar recursos para ayudas sociales, se realizará el respectivo registro y modificación de recursos. El sistema visualizara mensajes de error si la información proporcionada no es la adecuada.

- **Condiciones de ejecución:**

La información proporcionada por el usuario tiene que ser incorrecta.

- **Entrada:**

- Dentro del menú principal seleccionar 'recurso', posteriormente 'nuevo', se presentara un formulario donde se deberá llenar los siguientes campos del formulario: categoría, artículo, detalle, stock, observación (no obligatorio).
- El sistema validará los datos y visualizará en pantalla alertas.
- El proceso se considera como finalizado.

- **Resultado esperado:**

La información de un nuevo recurso no deberá almacenarse, si toda la información no cumple con las reglas establecidas en los campos del formulario.

- **Evaluación de prueba:**

Prueba satisfactoria.

2.5.3. *Pruebas tercera iteración*

Historia de usuario 10: gestión de capacitaciones

Registro de información correcta de capacitaciones

- **Descripción:**

El usuario con acceso al módulo de capacitaciones con permisos de técnico tendrá la facultad de registrar, modificar capacitaciones. Una vez registrada correctamente se listara las capacitaciones para ser publicadas.

- **Condiciones de ejecución:**

El usuario deberá acceder al módulo de capacitación con permisos de técnico, además tiene que estar registrados usuarios con permisos de capacitadores para ser asignados a las capacitaciones, y por último deben estar registrados programas asignados a técnicos.

- **Entrada:**

- Dentro del menú principal seleccionar ‘capitación,’ se presentara un formulario donde se deberá llenar los siguientes campos del formulario: nombre (programa), fecha de inicio, fecha de finalización, horas, lugar y responsable (capitador).
- El sistema validará los datos y almacenará la información en la base de datos.
- El proceso se considera como finalizado.

- **Resultado esperado:**

Si toda la información cumple con lo establecido el sistema almacenarse los datos de capacitación para continuar con el proceso de publicación.

- **Evaluación de prueba:**

Prueba satisfactoria.

Registro de información incorrecta de capacitaciones

- **Descripción:**

El usuario con acceso al módulo de capacitaciones con permisos de técnico tendrá la facultad de registrar, modificar capacitaciones. Si los datos ingresados no cumplen con los parámetros de validación el sistema visualizara en pantalla mensajes de alerta para el usuario.

- **Condiciones de ejecución:**

Los datos proporcionados deberán ser incorrectos o faltantes.

- **Entrada:**

- Dentro del menú principal seleccionar ‘capacitación,’ se presentara un formulario donde se deberá llenar los siguientes campos del formulario: nombre (programa), fecha de inicio, fecha de finalización, horas, lugar y responsable (capacitador).
- El sistema validará los datos y mostrara alertas en pantalla.
- El proceso se considera como finalizado.

- **Resultado esperado:**

No se efectuará el registro de la capacitación, si la información proporcionada es incorrecta o faltante.

- **Evaluación de prueba:**

Prueba satisfactoria.

Historia de usuario 11: inscripciones a capacitaciones

Registro de información correcta de inscripciones

- **Descripción:**

La inscripción de beneficiarios a capacitaciones lo realizará el usuario con permisos de capacitador, además podrá aprobar o reprobar y finalizar la capacitación con el objetivo de generar el respectivo certificado.

- **Condiciones de ejecución:**

Tener permisos de capacitador en el sistema, deberá haberse publicado una capacitación asignando al capacitador, caso contrario no se mostrara la capacitación en el perfil del capacitador. Además realizar el registro previo de beneficiarios.

- **Entrada:**

- Para iniciar el proceso es necesario realizar una búsqueda del beneficiario en el sistema para lo cual se deberá llenar los siguientes campos en el formulario: Cédula (no se acepta guion, solo números, máximo 10 caracteres).

- El sistema visualizará en pantalla la información de la persona, luego seleccionar 'inscribir' donde visualizará el formulario de inscripción con los siguientes datos requeridos: participante (automático) y taller.
- Sistema guardará la información en la base de datos.
- El proceso se considera finalizado.

- **Resultado esperado:**

Los datos serán almacenados con la correcta información proporcionada para continuar con el proceso de aprobar o reprobar.

- **Evaluación de prueba:**

Prueba satisfactoria.

Registro de información incorrecta de inscripciones

- **Descripción:**

La inscripción de beneficiarios a capacitaciones lo realizará el usuario con permisos de capacitador, además podrá aprobar o reprobar y finalizar la capacitación con el objetivo de generar el respectivo certificado. Si la información es incorrecta no se efectuará la inscripción, como respuesta el sistema visualizara mensajes de alerta (error) para el usuario.

- **Condiciones de ejecución:**

Proporcionar información correcta en el formulario de inscripción.

- **Entrada:**

- Para iniciar el proceso es necesario realizar una búsqueda del beneficiario en el sistema para lo cual se deberá llenar los siguientes campos en el formulario: Cédula (no se acepta guion, solo números, máximo 10 caracteres).
- El sistema visualizará en pantalla la información de la persona, luego seleccionar 'inscribir' donde visualizará el formulario de inscripción con los siguientes datos requeridos: participante (automático) y taller.
- El sistema validará y mostrará en pantalla alertas con los problemas presentados.
- El proceso se considera finalizado.

- **Resultado esperado:**

No deberá realizarse la inscripción si existe información faltante o si el beneficiario ya ha sido inscrito en el taller; como respuesta el sistema informará por pantalla el inconveniente.

- **Evaluación de prueba:**

Prueba satisfactoria.

Historia de usuario 11: generar reportes

Esta historia es generalizada para los tres módulos del sistema SIGEGAS donde se especifica la prueba de realizar reportes con parámetro o sin parámetros.

Registro de información correcta de reportes

- **Descripción:**

Los reportes podrán ser generados por todos los usuarios que tengan acceso al sistema de acuerdo a sus permisos o asignación de grupo. Se efectuara reportes sin parámetros y con parámetros.

- **Condiciones de ejecución:**

El usuario deberá poseer los para acceder a los módulos, deberá existir información almacenada.

- **Entrada (reportes sin parámetros):**

- En el menú principal del perfil seleccionar ‘reportes’, se mostrara una lista de reportes seleccionar una.
- El sistema realizará la consulta y mostrara la información solicitada en formato PDF.
- Proceso culminado.

- **Entrada (reportes con parámetros):**

- En el menú principal del perfil seleccionar ‘reportes’, se mostrara una lista de reportes seleccionar una.
- El sistema mostrara un formulario donde se deberá llenar los datos requeridos.
- El sistema realizará la consulta y mostrara la información solicitada en formato PDF.
- Proceso culminado.

- **Resultado esperado:**

Los datos serán consultados si la información proporcionada es correcta para continuar con el proceso de generar el reporte en formato PDF.

- **Evaluación de prueba:**

Prueba satisfactoria.

Registro de información correcta de reportes

- **Descripción:**

Los reportes podrán ser generados por todos los usuarios que tengan acceso al sistema de acuerdo a sus permisos o asignación de grupo. Se efectuara reportes sin parámetros y con parámetros. No se presentará la información si los datos son incorrectos en el caso de los reportes con parámetros, el sistema visualizará en pantalla una alerta.

- **Condiciones de ejecución:**

Proporcionar información incorrecta para los reportes con parámetros.

- **Entrada (reportes con parámetros):**

- En el menú principal del perfil seleccionar 'reportes', se mostrara una lista de reportes seleccionar una.
- El sistema mostrara un formulario donde se deberá llenar los datos requeridos.
- El sistema validará la información y mostrara una alerta.
- Proceso culminado.

- **Resultado esperado:**

No se deberá generar el reporte si la información proporcionada es incorrecta.

- **Evaluación de prueba:**

Prueba satisfactoria.

CAPÍTULO III

3. MARCO DE RESULTADOS

Este capítulo se centra en el análisis de los servicios que presta el departamento de Equidad, Género y Acción Social en cuanto al tiempo que se demora en ejecutarse un determinado servicio manualmente. Por otra parte se determina las ventajas en tiempo que ofrece la aplicación web SIGEGAS en los procesos automatizados dentro del departamento con respecto al tiempo.

A continuación en la tabla 99-3 se muestra los procesos generales que se efectúan para acceder a los servicios de: Posada y Comedor, Ayuda Social y Capacitaciones del departamento de Equidad, Género y Acción Social.

Tabla 99-3: Descripción de procesos manuales a automatizar.

Nombre del servicio	Descripción de procesos
Posada y Comedor	<p>Proceso para acceder a Posada:</p> <ol style="list-style-type: none">1. El beneficiario debe presentar un oficio de solicitud de posada al director del CEFAS.2. El director debe dar el visto bueno.3. Una vez aprobada la solicitud de posada por parte del director, el beneficiario de debe presentar la documentación necesaria para efectuar el registro de posada.4. <u>El encargado de centro intercultural crea la ficha para legalizar la posada.</u>5. El beneficiario debe firmar la ficha.6. <u>El encargado de centro intercultural lleva un registro de posantes diarios donde controla la fecha de salida de acuerdo a los días de estadía para marcar la salida del posante.</u>7. El posante debe firmar un registro de salida. <p>Proceso para acceder a Comedor (beneficiarios):</p> <ol style="list-style-type: none">1. el beneficiario debe presentar una copia de cédula en el centro intercultural del CEFAS.2. <u>El encargado del centro intercultural realiza el registro de la alimentación para el beneficiario.</u>3. el beneficiario debe firmar un documento para legalizar la entrega de alimentación.4. El encargado del centro intercultural entrega un tique al beneficiario y finaliza el proceso. <p>Proceso para acceder a Comedor (funcionarios):</p>

	<ol style="list-style-type: none"> 1. <u>El encargado del centro intercultural lleva un registro de funcionarios del CEFAS.</u> 2. <u>El funcionario realiza la compra la compra del almuerzo; puede pagar o adeudar el almuerzo.</u> 3. <u>El encargado del centro intercultural llena el registro de alimentación.</u> 4. El funcionario firma el registro de alimentación y finaliza el proceso.
Ayuda Social	<p>Proceso para acceder a Ayudas Sociales</p> <ol style="list-style-type: none"> 1. El beneficiario debe presentar un oficio de solicitud detallando la ayuda que requiere al director del CEFAS. 2. El director debe aprobar la solicitud a la brevedad posible. 3. El beneficiario debe presentar el oficio aprobado y documentos comentarios en el área de acción social. 4. La trabajadora social realizará una entrevista personal al beneficiario, para determinar su estatus económico. 5. <u>La trabajadora social realiza un análisis del historial del beneficiario.</u> 6. <u>La trabajadora social verifica si existe el recurso solicitado.</u> 7. <u>La trabajadora social crea o registra la ayuda solicitada.</u> 8. La trabajadora social crear el informe social para legalizar la entrega de la ayuda solicitada y finaliza el proceso.
Capacitaciones	<p>Proceso para acceder a Capacitaciones:</p> <ol style="list-style-type: none"> 1. Debe existir un plan operativo anual, donde se describe las capacitaciones que se realizarán en una determinada comunidad y las fechas específicas. 2. De acuerdo al plan operativo el asistente administrativo socializara a la comunidad que recibirá la capacitación. 3. <u>El asistente administrativo lleva un registro de las capacitaciones que están a su cargo.</u> 4. <u>El asistente administrativo asigna al capacitador acorde al área le la capacitación.</u> 5. <u>El capacitador realizará la inscripción de las personas durante la socialización; para ello la persona debe presentar una copia de cédula.</u> 6. <u>Luego de finalizado la capacitación el capacitador presentara la lista de personas que aprobaron el taller.</u> 7. <u>El asistente administrativo crea los certificados que se entregarán el último día de la capacitación y finaliza el proceso.</u>

Fuente: DUCHITANGA, J., 2016

La aplicación web SIGEGAS no automatiza todos los procesos que se efectúa en cada uno de los servicios del departamento, pero si ha tomado en consideración los siguientes procesos que se describen a continuación:

- Para el servicio de posada la aplicación automatiza el apartado número 4 y 6.
- Para el servicio de comedor (beneficiarios), la aplicación automatiza el apartado número 2 y 6.

- Para el servicio de comedor (funcionarios), la aplicación automatiza los apartados 1, 2,3.
- Para el servicio de ayudas sociales la aplicación automatiza los apartados 5, 6, 7.
- Finalmente para el servicio de capacitaciones la aplicación automatiza los apartados 3, 4, 5, 6, 7.

Para determinar los tiempos que se demora en cada uno de los servicios se tomó solo los procesos especificados anteriormente.

3.1. Análisis de resultados.

De acuerdo a la medición realizada a los procesos que se efectúan en las áreas de Posada y Comedor, Ayudas Social y Capacitaciones se obtuvo los siguientes valores para los procesos descritos a continuación.

Proceso 1.

Registro de Beneficiarios.

Tabla 100-3: Resultados proceso 1.

Proceso	Tiempo en minutos
Manual	8
Automatizado	2.5

Fuente: DUCHITANGA, JULIO. 2016

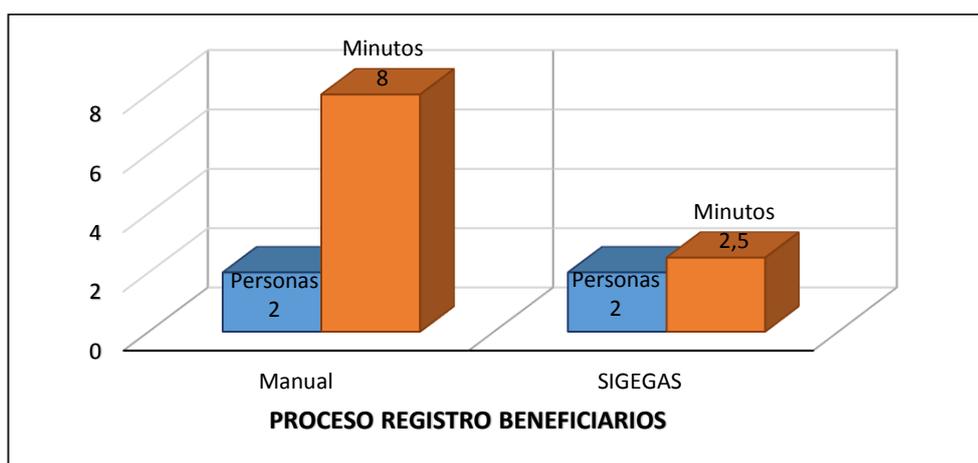


Figura 64-3 Representación estadística proceso 1.

Fuente: DUCHITANGA, JULIO. 2016

En el proceso registro de beneficiarios, de acuerdo a la figura 65-3 manualmente toma un tiempo de 8 minutos y con la aplicación web SIGEGAS un tiempo de 2.5 minutos; minimizando un 68.75% respecto al proceso manual.

Proceso 2.

Registrar una posada

Tabla 101-3: Resultados proceso 2.

Proceso	Tiempo en minutos
Manual	10
Automatizado	2

Fuente: DUCHITANGA, JULIO. 2016

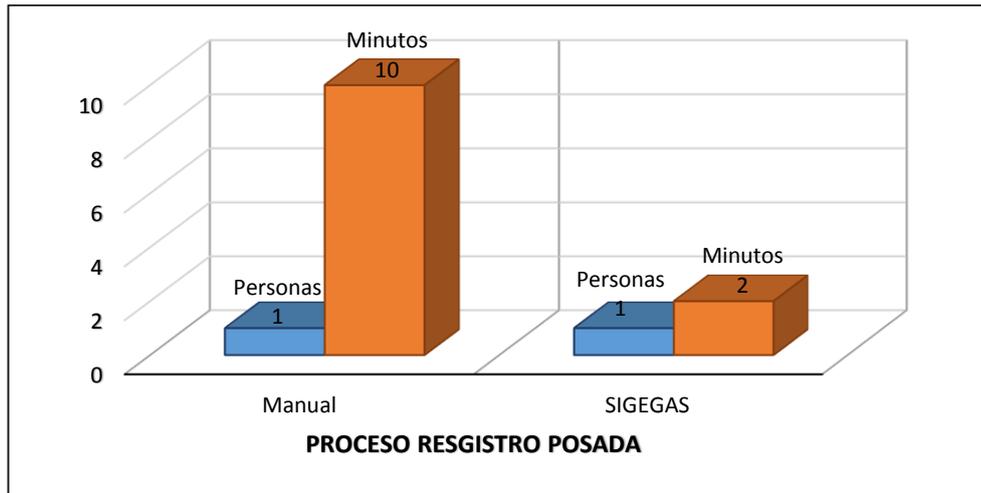


Figura 65-3 Representación estadística proceso 2.

Fuente: DUCHITANGA, JULIO. 2016

En el proceso registro de posada, de acuerdo a la figura 66-3 manualmente toma un tiempo de 10 minutos y con la aplicación web SIGEGAS un tiempo de 2 minutos; minimizando un 80% respecto al proceso manual.

Proceso 3.

Registrar a un beneficiario a comedor

Tabla 102-3: Resultados proceso 3.

Proceso	Tiempo en minutos
Manual	10
Automatizado	2

Fuente: DUCHITANGA, JULIO. 2016

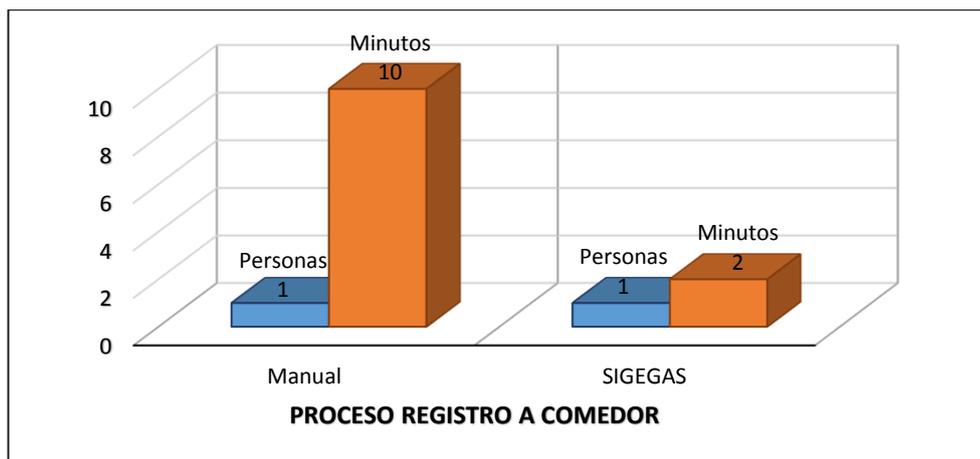


Figura 66-3 Representación estadística proceso 3.

Fuente: DUCHITANGA, JULIO. 2016

En el proceso registro de beneficiarios a comedor, de acuerdo a la figura 67-3 manualmente toma un tiempo de 10 minutos y con la aplicación web SIGEGAS un tiempo de 2 minutos; minimizando un 80% respecto al proceso manual.

Proceso 4.

Crear reporte de registro de posantes

Tabla 103-3: Resultados proceso 4.

Proceso	Tiempo en minutos
Manual	90
Automatizado	1

Fuente: DUCHITANGA, JULIO. 2016

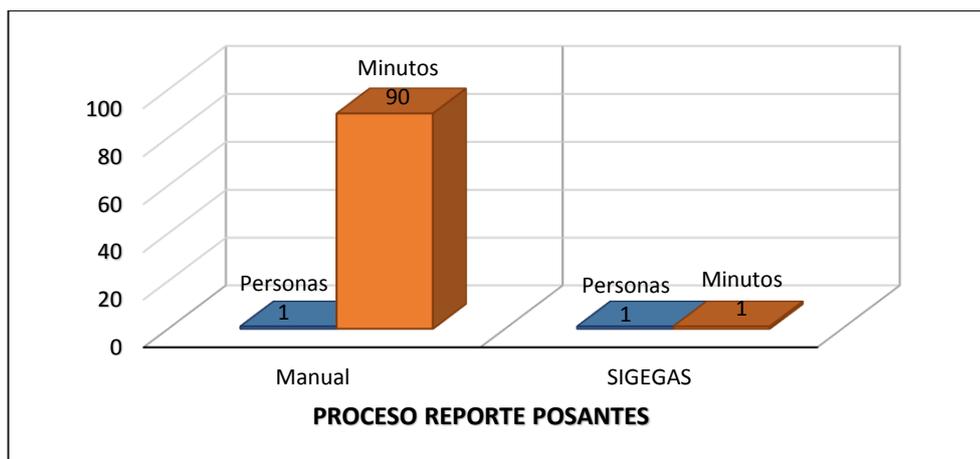


Figura 67-3 Representación estadística proceso 4.

Fuente: DUCHITANGA, JULIO. 2016

En el proceso de reporte de posantes, de acuerdo a la figura 68-3 manualmente toma un tiempo de 90 minutos y con la aplicación web SIGEGAS un tiempo de 1 minutos; minimizando un 98.88% respecto al proceso manual.

Proceso 5.

Dar de alta (salida) a un posante.

Tabla 104-3: Resultados proceso 5.

Proceso	Tiempo en minutos
Manual	5
Automatizado	1

Fuente: DUCHITANGA, JULIO. 2016

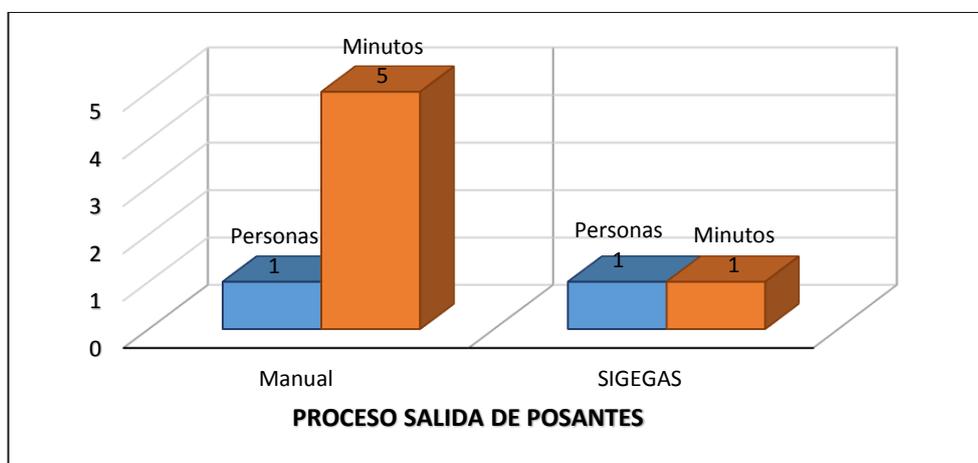


Figura 68-3 Representación estadística proceso 5.

Fuente: DUCHITANGA, JULIO. 2016

En el proceso de salida de posantes, de acuerdo a la figura 69-3 manualmente toma un tiempo de 5 minutos y con la aplicación web SIGEGAS un tiempo de 1 minutos; minimizando un 80% respecto al proceso manual.

Proceso 6.

Registrar una ayuda social.

Tabla 105-3: Resultados proceso 6.

Proceso	Tiempo en minutos
Manual	15
Automatizado	2

Fuente: DUCHITANGA, JULIO. 2016

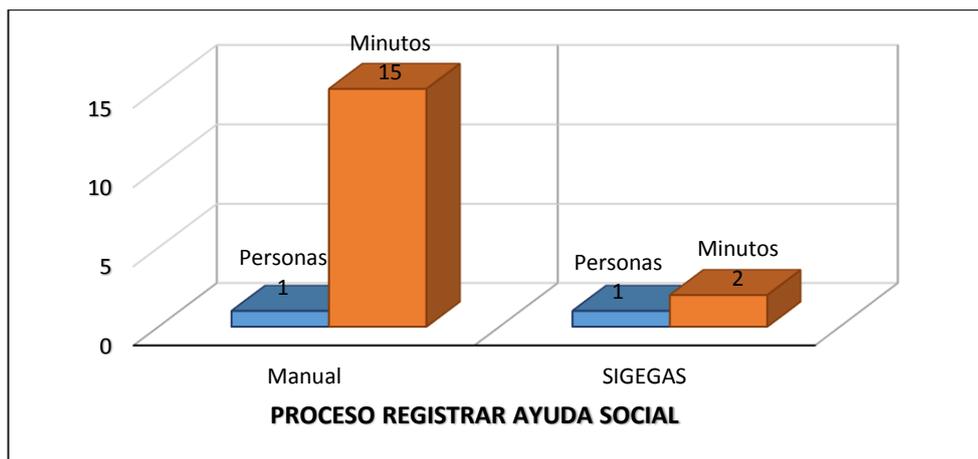


Figura 69-3 Representación estadística proceso 6.

Fuente: DUCHITANGA, JULIO. 2016

En el proceso de salida de posantes, de acuerdo a la figura 70-3 manualmente toma un tiempo de 15 minutos y con la aplicación web SIGEGAS un tiempo de 2 minutos; minimizando un 86.67% respecto al proceso manual.

Proceso 7.

Buscar el historial de ayudas para un beneficiario

Tabla 106-3: Resultado proceso 7

Proceso	Tiempo en minutos
Manual	10
Automatizado	1

Fuente: DUCHITANGA, JULIO. 2016

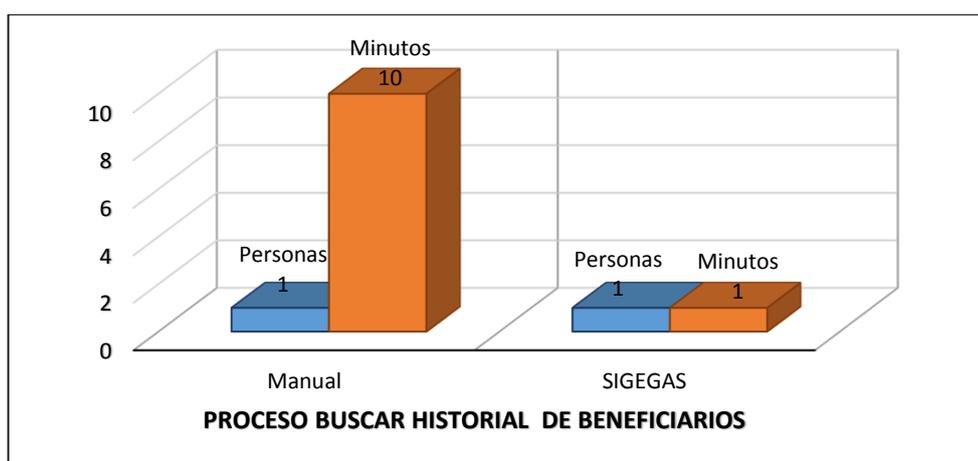


Figura 70-3 Representación estadística proceso 7.

Fuente: DUCHITANGA, JULIO. 2016

En el proceso de buscar el historial de beneficiarios, de acuerdo a la figura 71-3 manualmente toma un tiempo de 10 minutos y con la aplicación web SIGEGAS un tiempo de 1 minutos; minimizando un 90% respecto al proceso manual.

Proceso 8.

Generar un reporte de ayudas entregadas

Tabla 107-3: Resultados pregunta 8.

Proceso	Tiempo en minutos
Manual	15
Automatizado	1.5

Fuente: DUCHITANGA, JULIO. 2016

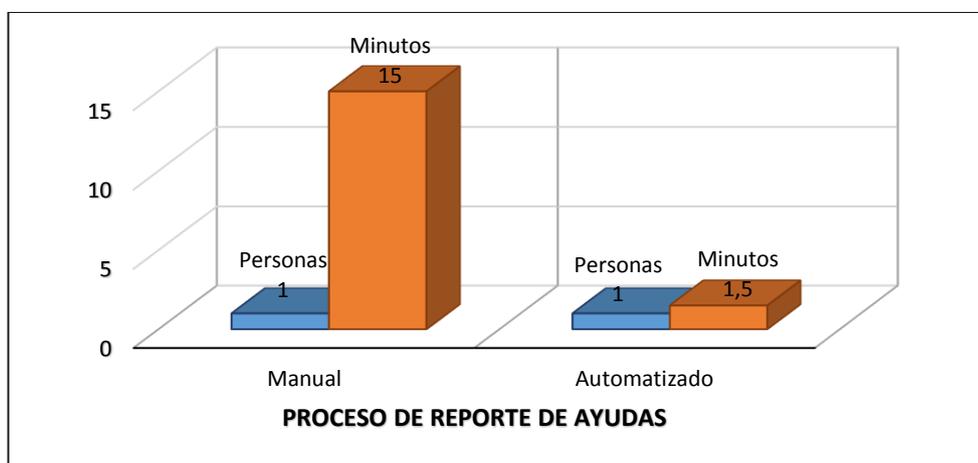


Figura 71-3 Representación estadística proceso 8.

Fuente: DUCHITANGA, JULIO. 2016

En el proceso de buscar el historial de beneficiarios, de acuerdo a la figura 72-3 manualmente toma un tiempo de 15 minutos y con la aplicación web SIGEGAS un tiempo de 1.5 minutos; minimizando un 90% respecto al proceso manual.

Proceso 9.

Revisar el stock de recursos para ayudas sociales

Tabla 108-3: Resultados proceso 9

Proceso	Tiempo en minutos
Manual	15
Automatizado	1.5

Fuente: DUCHITANGA, JULIO. 2016

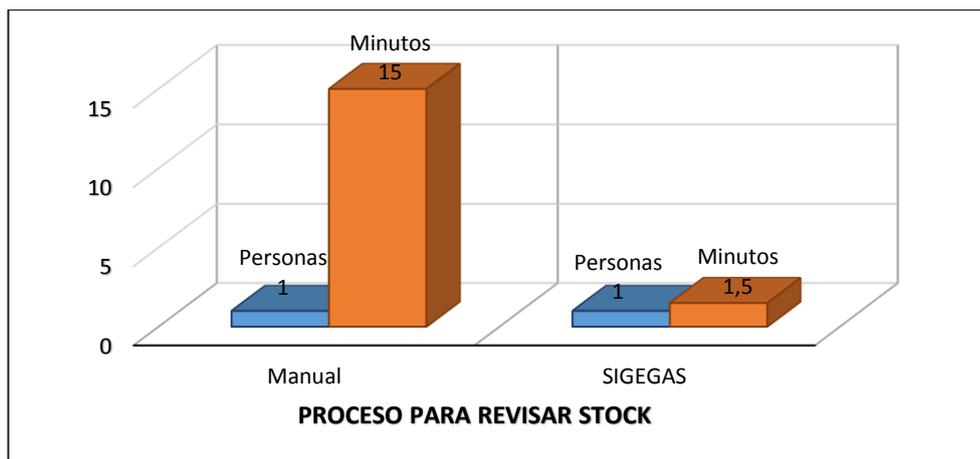


Figura 72-3 Representación estadística proceso 9.
Fuente: DUCHITANGA, JULIO. 2016

En el proceso para revisar el stock de recursos, de acuerdo a la figura 73-3 manualmente toma un tiempo de 15 minutos y con la aplicación web SIGEGAS un tiempo de 1.5 minutos; minimizando un 90% respecto al proceso manual.

Proceso 10.

Registrar beneficiario a una capacitación

Tabla 109-3: Resultados proceso 10.

Proceso	Tiempo en minutos
Manual	12
Automatizado	3

Fuente: DUCHITANGA, JULIO. 2016

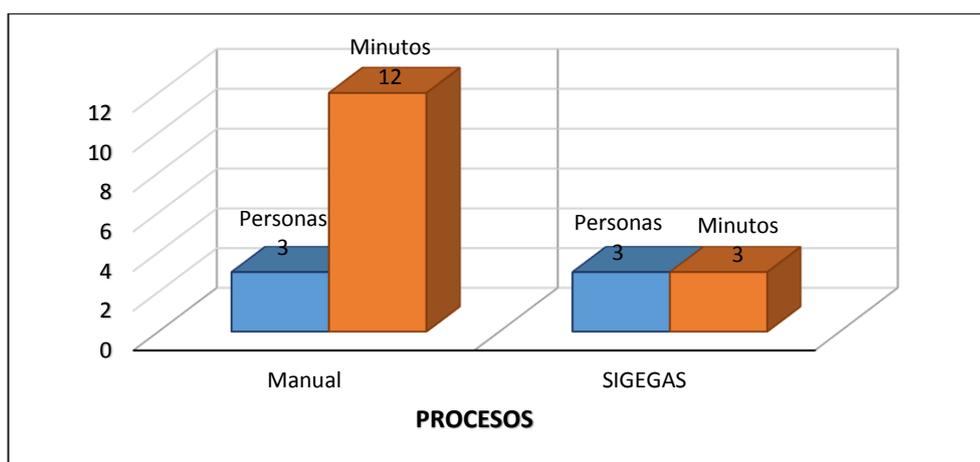


Figura 73-3 Representación estadística proceso 10.
Fuente: DUCHITANGA, JULIO. 2016

En el proceso de buscar el historial de beneficiarios, de acuerdo a la figura 74-3 manualmente toma un tiempo de 12 minutos y con la aplicación web SIGEGAS un tiempo de 3 minutos; minimizando un 75% con respecto al proceso manual.

Proceso 11.

Registrar una capacitación

Tabla 110-3: Resultados proceso 11.

Proceso	Tiempo en minutos
Manual	12
Automatizado	2.5

Fuente: DUCHITANGA, JULIO. 2016

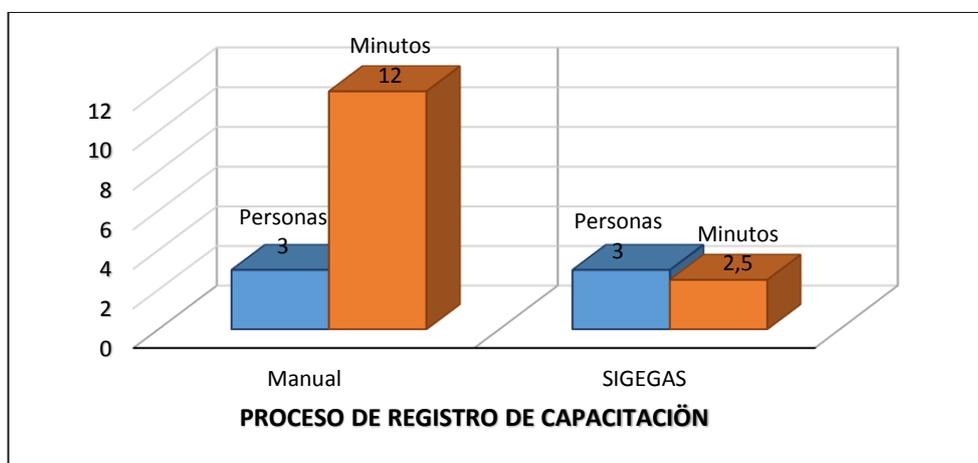


Figura 74-3 Representación estadística proceso 11.

Fuente: DUCHITANGA, JULIO. 2016

En el proceso de buscar el historial de beneficiarios, de acuerdo a la figura 75-3 manualmente toma un tiempo de 12 minutos y con la aplicación web SIGEGAS un tiempo de 2.5 minutos; minimizando un 79.17% respecto al proceso manual.

Proceso 12.

Generar un reporte de participantes de una capacitación

Tabla 111-3: Resultados proceso 12.

Proceso	Tiempo en minutos
Manual	40
Automatizado	3

Fuente: DUCHITANGA, JULIO. 2016

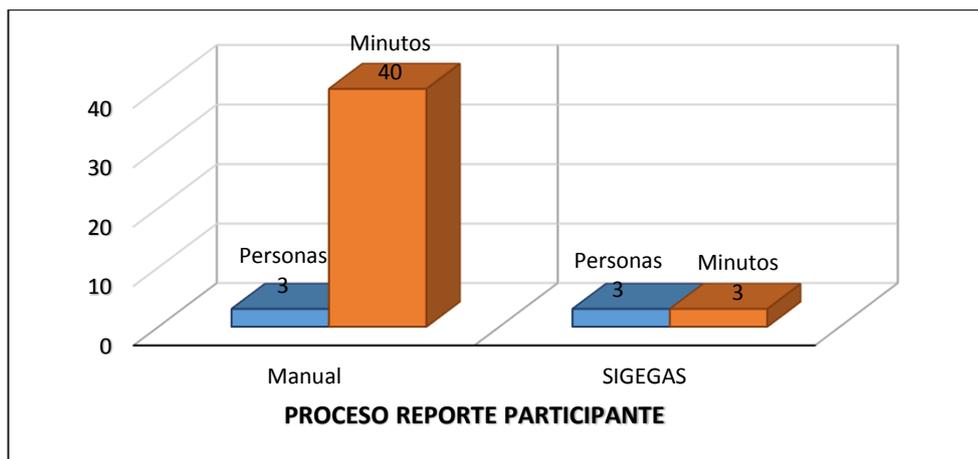


Figura 75-3 Representación estadística proceso 12.

Fuente: DUCHITANGA, JULIO. 2016

En el proceso para generar reporte de participantes, de acuerdo a la figura 76-3 manualmente toma un tiempo de 40 minutos y con la aplicación web SIGEGAS un tiempo de 3 minutos; minimizando un 92.5% respecto al proceso manual.

Proceso 13.

Generar un certificado para un participante

Tabla 112-3: Resultados proceso 13.

Proceso	Tiempo en minutos
Manual	28
Automatizado	1,3

Fuente: DUCHITANGA, JULIO. 2016

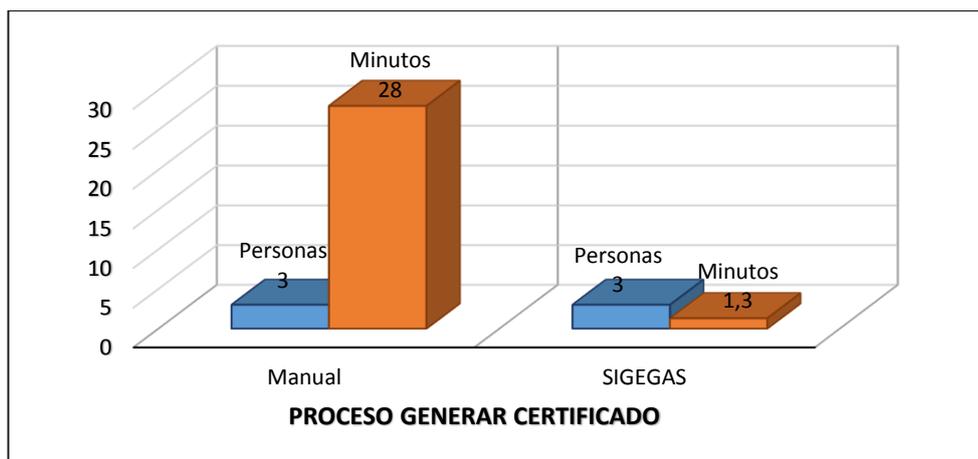


Figura 76-3 Representación estadística proceso 13.

Fuente: DUCHITANGA, JULIO. 2016

En el proceso de generar certificado, de acuerdo a la figura 77-3 manualmente toma un tiempo de 28 minutos y con la aplicación web SIGEGAS toma un tiempo de 1.3 minutos; minimizando un 95.36% respecto al proceso manual.

Proceso 14.

Elaborar una lista de capacitaciones mensuales

Tabla 113-3: Resultados proceso 14.

Proceso	Tiempo en minutos
Manual	30
Automatizado	2

Fuente: DUCHITANGA, JULIO. 2016

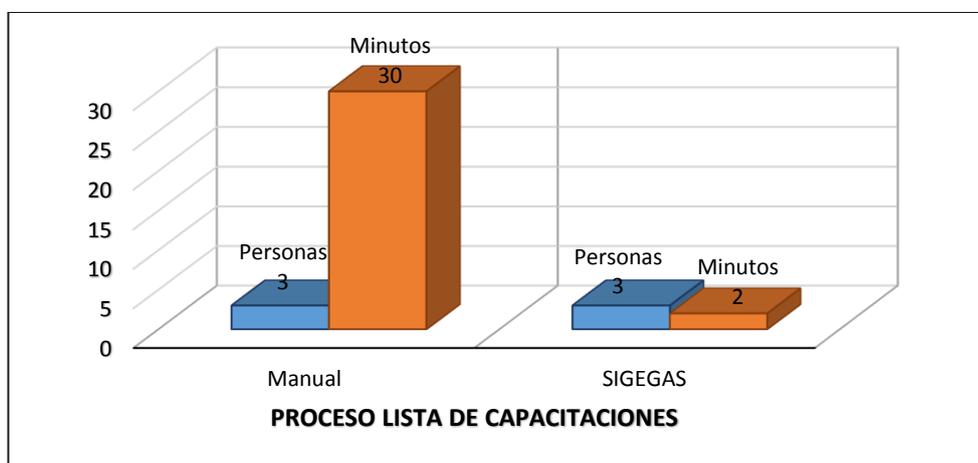


Figura 77-3 Representación estadística proceso 14.

Fuente: DUCHITANGA, JULIO. 2016

En el proceso generar lista de capacitaciones, de acuerdo a la figura 78-3 manualmente toma un tiempo de 35 minutos y con la aplicación web SIGEGAS un tiempo de 2 minutos; minimizando un 93.33% respecto al proceso manual.

3.1.1. Valores observados totales

De acuerdo al análisis de resultados se obtuvo los siguientes valores tanto para los procesos manuales como para los automatizados con la aplicación web SIGEGAS (ver tabla 114-3).

Tabla 114-3: Valores observados

PROCESOS	TIEMPO EN MINUTOS		MEJORA
	MANUAL	SIGEGAS	%
Registrar un beneficiario (P1)	8	2,5	68,75
Registrar una posada (P2)	10	2	80
Registrar beneficiario a comedor (P3)	10	2	80
Reporte de posantes (P4)	90	1	98,88
Marcar salida de posante (P5)	5	1	80
Registrar ayuda social (P6)	15	2	86,67
Buscar historial de ayudas por beneficiario (P7)	10	1	90
Reporte de ayudas sociales (P8)	15	1,5	90
Revisar Stock de recursos para ayudas sociales (P9)	15	1,5	90
Registrar una capacitación (P10)	12	3	75
Inscribir un beneficiario a capacitación (P11)	12	2,5	79,17
Reporte de beneficiarios de una capacitación (P12)	40	3	92,5
Generar certificado para beneficiario (P13)	28	1,3	95,36
Generar una lista de capacitaciones mensuales (P14)	30	2	93,33
TOTAL TIEMPO	300	26,3	

Fuente: DUCHITANGA, JULIO. 2016

El porcentaje de eficacia de la aplicación web en los procesos respecto a los procesos manuales se calculó utilizando la fórmula:

$$P_s = \frac{(\sum V_m - \sum V_s)(100)}{\sum V_m}$$

$$P_s = \frac{(300 - 26,3)(100)}{300}$$

$$P_s = 91.23 \%$$

Interpretación:

$\sum v_m$ = La sumatoria de los valores observados (Manual).

$\sum v_s$ = La sumatoria de los valores observados (SIGEGAS).

3.2. Ventajas de la aplicación web SIGEGAS

- La aplicación web optimiza los procesos manuales del departamento de Equidad Género y Acción Social, con una clara reducción en tiempos del 91.23% al ejecutar los procesos que gestiona la aplicación.
- Reduce el consumo recursos materiales y económicos como: esferos, lápices, borradores, transporte (al enviar la lista de beneficiarios que aprobaron una capacitación).
- Permite tener la información disponible en todo momento a un menor esfuerzo.
- Interfaz amigable para el usuario facilitando el ingreso correcto de la información.
- La información puede ser accedida desde cualquier parte con acceso a internet.
- Minimiza el tiempo del personal al ejecutar los procesos, teniendo más tiempo para ejecutar otras tareas.

CONCLUSIONES

1. Se concluye que actualmente los tiempos en los procesos manuales en el departamento de Equidad, Género y Acción Social son muy elevados en comparación a los procesos automatizados de acuerdo a los valores observados.
2. Django es un framework que permite optimizar el desarrollo de aplicaciones web que requieran un tiempo desarrollo corto; al proporcionar ventajas que facilitan el trabajo al desarrollador aplicando el principio DRY (Don't repeat Yourself) que fomenta la reutilización del código.
3. La metodología XP es caracterizada por la facilidad de su implementación; permite abstraer de mejor manera los requerimientos funcionales, al involucrar al cliente como un miembro activo durante todas las fases del desarrollo del software.
4. Concluido el desarrollo de la aplicación, se efectuó pruebas reales con el personal encargado de: Posada y Comedor, Ayudas Sociales y Capacitaciones, concluyendo que la aplicación web satisface los requerimientos propuestos a través de las historias de usuario del cliente.
5. Finalmente se realizó un análisis de los procesos automatizados lo que permite concluir que la aplicación web SIGEGAS minimiza tiempos en un 91.23% de eficacia respecto a los procesos manuales.

RECOMENDACIONES

1. Hacer uso de una metodología ágil para el desarrollo de aplicaciones orientadas a la web, se recomienda hacer uso de XP (Programación Extrema) por tratarse de una metodología que está estrechamente ligada al usuario, tomándole como otro miembro del equipo de desarrollo.
2. Utilizar el Framework DJANGO para el desarrollo de aplicaciones; porque permite un desarrollo ágil predicando la reutilización de código, además por tener una curva de aprendizaje alta.
3. Capacitar al personal sobre herramienta informáticas que permitan aprovechar al máximo la aplicación web SIGEGAS. Y así evitar inconvenientes futuros para con la aplicación.
4. Debido al constante crecimiento de mundo tecnológico se recomienda estar siempre actualizado con las nuevas herramientas de desarrollo o actualizaciones de herramientas de su elección; que le faciliten ejecutar tareas a un menor tiempo con un esfuerzo mínimo.
5. Implementar un módulo para no videntes a la aplicación SIGEGAS, que permita ampliar el rango de disponibilidad de la aplicación.

BIBLIOGRAFÍA

CASTILLO, Oswaldo, FIGUEROA, Daniel y SEVILLA , Hector. *Fases de la Programación Extrema.* [En línea].

<http://programacionextrema.tripod.com/fases.htm>.

[Consulta: 01 de mayo 2016.].

DEBERÍAS USAR PYTHON PARA EL DESARROLLO WEB. [En línea]. 2014.

<http://jagonzalez.org/deberias-usar-python-para-el-desarrollo-web/>.

[Consulta: 10 marzo 2016.].

FUNDATION, Django Software. *Django Documentation.* [En línea] 2016.

<https://docs.djangoproject.com/en/1.8/>

[Consulta: 10 de abril 2016.].

GARCIA Saul. *La guia definitiva de django.* Mexico : 2015. [pdf].

<https://pythonizame.s3.amazonaws.com/media/Book/guia-definitiva-django-18/file/34ba425e-5985-11e5-964d-04015fb6ba01.pdf>

[Consulta: 03 de marzo de 2016.].

POSTGRESQL. *Características, limitaciones y ventajas.* [En línea]. 2012.

<http://postgresql-dbms.blogspot.com/p/limitaciones-puntos-de-recuperacion.html>.

[Consulta: 12 de mayo 2016.].

PISKAMEN. *Metodología xp.* [En línea]. 2013.

<http://es.slideshare.net/Piskamen/metodologa-xp>.

[Consulta: 12 de mayo de 2016.].

¿QUÉ SON LOS FRAMEWORKS? [En línea]. 2014.

<http://blog.nubelo.com/que-son-los-frameworks/>.

[Consulta: 12 de marzo de 2016.].

QUE ES BOOTSTRAP. [En línea] 2015.

<http://www.kinaweb.es/que-es-bootstrap>.

[Consulta: 22 de abril de 2016.].

SUMMERFIELD, Mark. *Programación Python 3*. Madrid-España : Ediciones Anaya multimedia, 2009. pp 17 -18.

TEBA, Alejandro. *PostgreSQL - pgAdmin*. [En línea]. 2013.
<http://www.solvetic.com/tutoriales/article/299-postgresql-pgadmin/>.
[Consulta: 15 de 03 de 2016.].

VILCHEZ, Beger. *ReportLab*. [En línea] 2014.
<http://menteleal.blogspot.com/2014/01/reportlab-libreria-pdf-para-python-00.html>.
[Consulta: 10 de junio de 2016.].

ZAKAS, Nicholas, MCPEACK, Jeremy y otros. *Profesional AJAX*. Madrid-España : Ediciones Anaya multimedia, 2008. pp.23-30.

ANEXOS

ANEXO A: Certificado por haber implementado la aplicación web SIGEGAS.



CEFAS DE MORONA SANTIAGO
Coordinación de Equidad, Formación y Acción Social
Macas - Morona Santiago - Ecuador

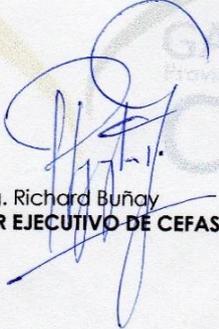
A petición verbal de la parte interesada,

CERTIFICO:

Que, el señor **Julio Cristian Duchitanga Cumbe**, portador de la cedula de ciudadanía N° 1400738694, estudiante de la carrera de Ingeniera en Sistemas de la Escuela Superior Politécnica de Chimborazo "ESPOCH" extensión Morona Santiago, desarrolló e implementó la aplicación web SIGEGAS para el Departamento de Equidad, Genero y Acción Social de CEFAS Patronato del GAPMS.

Es todo cuanto puedo certificar en honor a la verdad, autorizando al interesado hacer uso de la presente certificación en lo que creyere conveniente.

Macas, 11 de octubre del 2016.



Ing. Richard Buñay
DIRECTOR EJECUTIVO DE CEFAS

Dir: Gavino Rivadeneira y 24 de Mayo
Tel: 072 703 023 / 072 703 390
www.cefaspms.gob.ec
Macas - Morona Santiago - Ecuador

 CEFAS Morona Santiago
 @CEFASMS
 CEFAS Morona Santiago

Somos ejemplo de Responsabilidad!!!

Figura 1: Certificado de implementación SIGEGAS.

ANEXO B: Capacitación al personal del departamento de Equidad, Género y Acción Social.



Figura 1: Capacitación área de Ayuda Social.
Fuente: Netto Toño Tseremp



Figura 2: Capacitación área de Posada Y Comedor.
Fuente: Netto Toño Tseremp



Figura 3: Capacitación área de Capacitaciones.
Fuente: Netto Toño Tseremp

ANEXO C: Encuesta proceso manual.

8/6/2016

ENCUESTA POSADA Y COMEDOR

ENCUESTA POSADA Y COMEDOR

Medir el tiempo que se demora en este servicio

1. ¿Cual es el tiempo que lleva en registrar a un beneficiario?
Marca solo un óvalo.

- Bajo(5min)
 Medio(10m)
 Alto(15min)
 Otro:

2. ¿Cual es el tiempo que lleva en registrar una posada?
Marca solo un óvalo.

- Bajo(5min)
 Medio(10min)
 Alto(15min)
 Otro:

3. ¿Cual es el tiempo que se demora en registrar a un beneficiario a comedor
Se refiere al tiempo que lleva en dar el servicios de comedor a un beneficiario
Marca solo un óvalo.

- Bajo(5min)
 Medio(10min)
 Alto(15min)
 Otro:

4. ¿Cual es el tiempo que lleva en crear reporte de registro de posantes
Marca solo un óvalo.

- Bajo(5min)
 Medio(10min)
 Alto(15min)
 Otro: 2 horas

5. ¿Cual es el tiempo que se demora en dar de alta(salida) a un posante.
Se refiere al la salida de los posantes
Marca solo un óvalo.

- Bajo(5min)
 Medio(10min)
 Alto(15min)
 Otro:

Con la tecnología de
 Google Forms



Figura 1: Encuesta área de Posada Y Comedor.

ENCUESTA AYUDA SOCIAL

Conocer el tiempo que lleva en ejecutar este proceso

1. ¿Cual es el tiempo que lleva en registrar una ayuda social?

Se refiere al tiempo de en el registro de la ayuda luego de haber efectuador las respectiva entrevista al beneficiario

Marca solo un óvalo.

- Bajo(5min)
- Medio(10min)
- Alto(15min)
- Otro: _____

2. ¿Cual es el tiempo que lleva en registrar a un beneficiario

Marca solo un óvalo.

- Bajo(5min)
- Medio(10min)
- Alto(15min)
- Otro: _____

3. ¿Cual es el tiempo que lleva en buscar el historial de un beneficiario?

Se refiere al tiempo que lleva en buscar registros anteriores del beneficiario, si es que ha recibido una ayuda y el tipo de ayuda recibida

Marca solo un óvalo.

- Bajo(5min)
- Medio(10min)
- Alto (15min)
- Otro: _____

4. ¿Cual es el tiempo que lleva en generar un reporte de ayudas entregadas?

Se refiere a sacar del registro de las ayudas entregas una lista de ayudas entregadas

Marca solo un óvalo.

- Bajo(5min)
- Medio(10min)
- Bajo(15)min
- Otro: _____

5. ¿Cual es el tiempo que lleva en revisar el Stock de recursos para Ayudas sociales?

Se refiere a ver si existe un producto o artículo; dentro de almacenamiento para entregar en una ayuda social

Marca solo un óvalo.

- Bajo(5min)
- Medio(10min)
- Alto(15min)
- Otro: _____



Figura 1: Encuesta área de trabajo social.

ENCUESTA PARA CAPACITACIÓN

Conocer el tiempo que lleva en realizar los procesos dentro de capacitaciones (Proceso manual)

1. ¿Cual es el tiempo que lleva en el registro de una capacitaciones?

Marca solo un óvalo.

- Bajo(5min)
- Medio(10min)
- Alto(15min)
- Otro:

2. ¿Cual es el tiempo que lleva registrar un beneficiario a una capacitación?

Se refiere a la inscripción de usuarios a una capacitación.

Marca solo un óvalo.

- Bajo(5min)
- Medio(10min)
- Alto(15min)
- Otro:

3. ¿cual es el tiempo que lleva en generar un reporte de participantes de una capacitación?

Se refiere a elaborar una lista de participantes de una determinada capacitación.

Marca solo un óvalo.

- Bajo(5min)
- Medio(10min)
- Alto(15min)
- Otro: 1 Hora

4. ¿Cual es el tiempo que se demora en generar un certificado para un participante?

Se refiere al certificado que se entrega al final de una capacitación.

Marca solo un óvalo.

- Bajo(5min)
- Medio(10min)
- Alto(15)
- Otro: 1. Hora

5. ¿Cual es el tiempo que lleva en elaborar un lista de capacitaciones semestrales?

Se refiere a elabora una lista de capacitaciones realizadas.

Marca solo un óvalo.

- Bajo(5min)
- Medio(10min)
- Alto(15min)
- Otro: 30 minutos



Figura 3: Encuesta área de capacitaciones.

ENCUESTA PARA CAPACITACIÓN

Conocer el tiempo que lleva en realizar los procesos dentro de capacitaciones (Proceso manual)

1. ¿Cual es el tiempo que lleva en el registro de una capacitaciones?

Marca solo un óvalo.

- Bajo(5min)
 Medio(10min)
 Alto(15min)
 Otro:

2. ¿Cual es el tiempo que lleva registrar un beneficiario a una capacitación?

Se refiere a la inscripción de usuarios a una capacitación.

Marca solo un óvalo.

- Bajo(5min)
 Medio(10min)
 Alto(15min)
 Otro:

3. ¿cual es el tiempo que lleva en generar un reporte de participantes de una capacitación?

Se refiere a elaborar una lista de participantes de una determinada capacitación.

Marca solo un óvalo.

- Bajo(5min)
 Medio(10min)
 Alto(15min)
 Otro: 30 min

4. ¿Cual es el tiempo que se demora en generar un certificado para un participante?

Se refiere al certificado que se entrega al final de una capacitación.

Marca solo un óvalo.

- Bajo(5min)
 Medio(10min)
 Alto(15)
 Otro:

5. ¿Cual es el tiempo que lleva en elaborar un lista de capacitaciones semestrales?

Se refiere a elabora una lista de capaciotaciones realizadas.

Marca solo un óvalo.

- Bajo(5min)
 Medio(10min)
 Alto(15min)
 Otro: 1h.

Figura 4: Encuesta área de capacitaciones.

ENCUESTA PARA CAPACITACIÓN

Conocer el tiempo que lleva en realizar los procesos dentro de capacitaciones (Proceso manual)

1. ¿Cual es el tiempo que lleva en el registro de una capacitaciones?

Marca solo un óvalo.

- Bajo(5min)
 Medio(10min)
 Alto(15min)
 Otro: _____

2. ¿Cual es el tiempo que lleva registrar un beneficiario a una capacitación?

Se refiere a la inscripción de usuarios a una capacitación.

Marca solo un óvalo.

- Bajo(5min)
 Medio(10min)
 Alto(15min)
 Otro: _____

3. ¿cual es el tiempo que lleva en generar un reporte de participantes de una capacitación?

Se refiere a elaborar una lista de participantes de una determinada capacitación.

Marca solo un óvalo.

- Bajo(5min)
 Medio(10min)
 Alto(15min)
 Otro: 30 M

4. ¿Cual es el tiempo que se demora en generar un certificado para un participante?

Se refiere al certificado que se entrega al final de una capacitación.

Marca solo un óvalo.

- Bajo(5min)
 Medio(10min)
 Alto(15)
 Otro: _____

5. ¿Cual es el tiempo que lleva en elaborar un lista de capacitaciones semestrales?

Se refiere a elabora una lista de capaciitaciones realizadas.

Marca solo un óvalo.

- Bajo(5min)
 Medio(10min)
 Alto(15min)
 Otro: _____

Figura 5: Encuesta área de capacitaciones.

ANEXO D: Encuesta proceso automatizado.

9/10/2016

ENCUESTA POSADA Y COMEDOR

ENCUESTA POSADA Y COMEDOR

Medir el tiempo que se demora en estos procesos (Automatizado)

1. ¿Cual es el tiempo que lleva en registrar a un beneficiario?

Marca solo un óvalo.

Bajo(3min)

Otro: (2 min)

2. ¿Cual es el tiempo que lleva en registrar una posada?

Marca solo un óvalo.

Bajo(3min)

Otro: (2 min)

3. ¿Cual es el tiempo que se demora en registrar a un beneficiario a comedor

Se refiere al tiempo que lleva en dar el servicios de comedor a un beneficiario

Marca solo un óvalo.

Bajo(3min)

Otro: (1 min)

4. ¿Cual es el tiempo que lleva en crear reporte de registro de posantes

Marca solo un óvalo.

Bajo(3min)

Otro: (80 seg)

5. ¿Cual es el tiempo que se demora en dar de alta(salida) a un posante.

Se refiere al la salida de los posantes

Marca solo un óvalo.

Bajo(3min)

Otro: (1 min)

Con la tecnología de
 Google Forms



Figura 1: Encuesta área de Posada Y Comedor.

ENCUESTA AYUDA SOCIAL

Conocer el tiempo que lleva en ejecutar estos procesos (automatizados)

1. ¿Cual es el tiempo que lleva en registrar una ayuda social?

Se refiere al tiempo de en el registro de la ayuda luego de haber efectuador las respectiva entrevista al beneficiario

Marca solo un óvalo.

Bajo(3min)

Otro: (1 minuto)

2. ¿Cual es el tiempo que lleva en registrar a un beneficiario

Marca solo un óvalo.

Bajo(3min)

Otro: (1 minuto)

3. ¿Cual es el tiempo que lleva en buscar el historial de un beneficiario?

Se refiere al tiempo que lleva en buscar registros anteriores del beneficiario, si es que ha recibido una ayuda y el tipo de ayuda recibida

Marca solo un óvalo.

Bajo(1min)

Otro: (1 minuto)

4. ¿Cual es el tiempo que lleva en generar un reporte de ayudas entregadas?

Se refiere a sacar del registro de las ayudas entregas una lista de ayudas entregadas

Marca solo un óvalo.

Bajo(3min)

Otro: (30 segundos)

5. ¿Cual es el tiempo que lleva en revisar el Stock de recursos para Ayudas sociales?

Se refiere a si existe un producto o articulo; dentro de almacenamiento para entregar en una ayuda social

Marca solo un óvalo.

Bajo(3min)

Otro: (1 minuto)

Con la tecnología de
 Google Forms



Figura 2: Encuesta área de Trabajo Social.

ENCUESTA PARA CAPACITACIÓN

Conocer el tiempo que lleva en realizar los procesos dentro de capacitaciones (Proceso automatizado)

1. ¿Cual es el tiempo que lleva en el registro de una capacitaciones?

Marca solo un óvalo.

Bajo(1min)

Otro: _____

2. ¿Cual es el tiempo que lleva registrar un beneficiario a una capacitación?

Se refiere a la inscripción de usuarios a una capacitación.

Marca solo un óvalo.

Bajo(3min)

Otro: _____

3. ¿cual es el tiempo que lleva en generar un reporte de participantes de una capacitación?

Se refiere a elaborar una lista de participantes de una determinada capacitación.

Marca solo un óvalo.

Bajo(1min)

Otro: _____

4. ¿Cual es el tiempo que se demora en generar un certificado para un participante?

Se refiere al certificado que se entrega al final de una capacitación.

Marca solo un óvalo.

Bajo(1min)

Otro: _____

5. ¿Cual es el tiempo que lleva en elaborar un lista de capacitaciones mensual?

Se refiere a elabora una lista de capacitaciones realizadas.

Marca solo un óvalo.

Bajo(1min)

Otro: 2 minutos

Con la tecnología de
 Google Forms


140071708-6

Figura 3: Encuesta área de Capacitaciones.

ENCUESTA PARA CAPACITACIÓN

Conocer el tiempo que lleva en realizar los procesos dentro de capacitaciones (Proceso automatizado)

1. ¿Cual es el tiempo que lleva en el registro de una capacitaciones?

Marca solo un óvalo.

Bajo(1min)

Otro: _____

2. ¿Cual es el tiempo que lleva registrar un beneficiario a una capacitación?

Se refiere a la inscripción de usuarios a una capacitación.

Marca solo un óvalo.

Bajo(3min)

Otro: 2 min

3. ¿cual es el tiempo que lleva en generar un reporte de participantes de una capacitación?

Se refiere a elaborar una lista de participantes de una determinada capacitación.

Marca solo un óvalo.

Bajo(1min)

Otro: _____

4. ¿Cual es el tiempo que se demora en generar un certificado para un participante?

Se refiere al certificado que se entrega al final de una capacitación.

Marca solo un óvalo.

Bajo(1min)

Otro: _____

5. ¿Cual es el tiempo que lleva en elaborar un lista de capacitaciones mensual?

Se refiere a elabora una lista de capacitaciones realizadas.

Marca solo un óvalo.

Bajo(1min)

Otro: _____

Con la tecnología de
Google Forms

Rolando J. J. J.

Figura 4: Encuesta área de Capacitaciones.

ENCUESTA PARA CAPACITACIÓN

Conocer el tiempo que lleva en realizar los procesos dentro de capacitaciones (Proceso automatizado)

1. ¿Cual es el tiempo que lleva en el registro de una capacitaciones?

Marca solo un óvalo.

Bajo(1min)

Otro: 2 min

2. ¿Cual es el tiempo que lleva registrar un beneficiario a una capacitación?

Se refiere a la inscripción de usuarios a una capacitación.

Marca solo un óvalo.

Bajo(3min)

Otro: 3

3. ¿cual es el tiempo que lleva en generar un reporte de participantes de una capacitación?

Se refiere a elaborar una lista de participantes de una determinada capacitación.

Marca solo un óvalo.

Bajo(1min)

Otro:

4. ¿Cual es el tiempo que se demora en generar un certificado para un participante?

Se refiere al certificado que se entrega al final de una capacitación.

Marca solo un óvalo.

Bajo(1min)

Otro:

5. ¿Cual es el tiempo que lleva en elaborar un lista de capacitaciones mensual?

Se refiere a elabora una lista de capaciaciones realizadas.

Marca solo un óvalo.

Bajo(1min)

Otro:

Con la tecnología de
 Google Forms


Juan Campoverde

Figura 5: Encuesta área de Capacitaciones.

ANEXO E: Manual de usuario.

SISTEMA DE EQUIDAD GÉNERO Y ACCIÓN SOCIAL (SIGEGAS)

1. OBJETIVO

Ayudar a las personas a entender e utilizar los procesos automatizados de la aplicación web SIGEGAS.

2. ¿QUIÉNES PUEDEN UTILIZAR ESTE MANUAL?

Todas las personas que actúen directamente en las áreas de: Posada y Comedor; Ayudas sociales y Capacitaciones.

3. SIGEGAS

Es una aplicación web encargada de la gestión de posadas, ayudas sociales y capacitaciones, los cuales están divididos en tres módulos de acuerdo los procesos que efectúan dentro del departamento.

Además maneja cinco grupos de usuarios con sus respectivos permisos asignados que a continuación se listan:

- ▲ Administrador.
- ▲ Posada.
- ▲ Social.
- ▲ Tecnico.
- ▲ Capacitador.

3.1. Requerimientos de hardware y software

Hardware mínimo:

- ▲ Procesador Core 2DUO.
- ▲ Memoria RAM 2gb.
- ▲ Tarjeta de red inalámbrica lo incorporada.

Software mínimo:

Solo se requiere un navegador de los mencionados a continuación:

- ▲ Opera, Google Chrome, Mozilla Firefox, Safari.
- ▲ Acceso a internet.

3.2. ¿Cómo ingresar a SIGEGAS?

Aquí se detalla como el usuario debe ingresar a la aplicación web.

Paso1. Abrir el navegador web (ver figura 1).



Figura 1: Ejecutar navegador

Paso2. En la barra de direcciones tipear la siguiente dirección: <http://cefasms.herokuapp.com> (Ver figura 2).



Figura 2: Ingresar la dirección en navegador web.

Fuente: DUCHITANGA, JULIO. 2016.

Nota: La misma mecánica para cualquier navegador.

Luego de ejecutar la dirección la aplicación mostrará la página de inicio de SIGEGAS (ver figura 3)



Figura 3: Página de inicio SIGEGAS

Fuente: DUCHITANGA, JULIO. 2016.

3.3. INGRESAR A LOS MODULO DE SIGEGAS

Paso 1: En el formulario de SIGEGAS ingresar Usuario y Contraseña proporcionada por el administrador (ver figura 4).



Figura 4: Formulario de autenticación

Fuente: DUCHITANGA, JULIO. 2016.

3.4. Módulo de Posada y Comedor



Figura 5: Pantalla principal de Posada y Comedor

Fuente: DUCHITANGA, JULIO. 2016.

3.4.1 Registrar un beneficiario.

Paso1: De la barra de menú de posada seleccionar **Posadas** luego **Reg. Posada/Almuerzo** (ver figura 6).



Figura 6: Selección de Reg. Posada/Almuerzo

Fuente: DUCHITANGA, JULIO. 2016.

Paso2: Este proceso mostrará el siguiente formulario en el cual debe ingresar la **cédula** del beneficiario para buscar si existe o no (ver figura 7).

The screenshot shows a search form for beneficiaries. The form has a title 'BENEFICIARIO' and a search input field with the placeholder 'Cédula ejemplo 1400738694'. A green 'Buscar' button is below the input field. A note below the button reads: 'Recuerde que!! Es necesario realizar una búsqueda para verificar que la existencia del cliente en el sistema; si no existe proceda a registrarlo. Por favor ingrese la cédula sin guion.'

Figura 7: búsqueda para beneficiarios

Fuente: DUCHITANGA, JULIO. 2016.

Paso3: Si el beneficiario no está registrado en el sistema SIGEGAS mostrará un mensaje en el cual se debe seleccionar **Registrar ahora** (ver figura 7).



Figura 7: Seleccionar Registrar Ahora
Fuente: DUCHITANGA, JULIO. 2016.

Paso 4: Mostrará un formulario de registro para beneficiarios en el cual debe llenar los campos obligatorios y los no obligatorios (opcional) y luego **Guardar** o **Cancelar** para volver a la página principal (ver figura 8).

Figura 8: Registrar beneficiario
Fuente: DUCHITANGA, JULIO. 2016.

3.4.2. *Modificar un beneficiario*

Paso 1: Si al ejecutar el paso 1 del apartado 3.4.1.1 el beneficiario existe; SIGEGAS mostrara información del beneficiario (ver figura 9) con tres opciones a ejecutar:

- ▲ Actualizar datos.
- ▲ Registrar Almuerzo.
- ▲ Crear Posada.

Nombres: Julio Cristian Duchitanga Cumbe
Cedula: 1400738694
Edad: 25
Nacionalidad: Mestizo
Estado Civil: Soltero/a
Instrucción: Superior
Ocupación: Desarrollador
Discapacidad: Ninguna
Domicilio: Cantón Morona, Parroquia Sinaí, Comunidad Sinaí

Actualizar Crear Posada!! Registrar Almuerzo!!

Figura 9: Actualizar, Crear Posada o Registrar Almuerzo
Fuente: DUCHITANGA, JULIO. 2016.

Paso2: Seleccionada la opción **actualizar** SIGEGAS mostrará el formularios para modificar la información del usuario (seguir el procedimiento del paso4 del apartado 3.4.1.1.).

3.4.3. Registrar una posada.

Paso1: Seleccionar **Crear posada**, seguir el procedimiento descrito en el **paso1** del apartado 3.4.1.2. (Ver figura 9).

Paso2: Ejecutada la operación la aplicación mostrar el formulario para registro de posada (ver figura 10) en el cual debe llenar los campos y **guardar o cancelar** para volver al menú principal.

POSADA

Los campos con este simbolo (*) son obligatorios

Beneficiario *
Julio Cristian Duchitanga Cumbe

Dias de estadía *
[Empty field]

Tipo de posada *
[Dropdown menu]

Selección automatica al proporcionar el dia.

Fecha de salida *
[Empty field]

Calcula automaticamente la fecha al proporcionar el dia.

Habitacion *
[Dropdown menu]

Guardar Cancelar

Figura 10: Registrar beneficiario
Fuente: DUCHITANGA, JULIO. 2016.

Paso2: Una vez guardada la posada la aplicación mostrará la posada activa, indicando la fecha de salida, con tres opciones a ejecutar (ver figura 11):

- ▲ Generar ficha ().
- ▲ Marcar salida.
- ▲ Registrar almuerzo para posante ().

Gloria Esperanza Castro Samaniego 		
Ingreso: Lunes, 17 Oct 2016 a las 1:24 a.m.		
Posada: 1 Dias (transcurrido 1 hora,24 minutos)		
Tipo: Ocasional		
Salida: Martes, 18 Oct 2016 a las 6 a.m.		
	 Salida	

Figura 11: seleccionar comedor, salida o ficha
Fuente: DUCHITANGA, JULIO. 2016.

3.4.4. Generar ficha de posada.

Paso1: Seleccionar *ficha*  (ver figura 11) para mostrar el documento para legalizar la posada.

3.4.1.5. Marcar salida de posada.

Paso1: Seleccionar *Salida* (ver figura 11) para marcar la salida del posante; la aplicación mostrará un mensaje (ver figura 12).

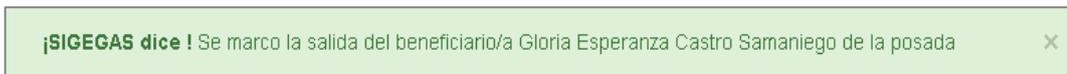


Figura 12: Mensaje salida posada
Fuente: DUCHITANGA, JULIO. 2016.

3.4.5. Registrar almuerzo para Posantes.

Paso1: Seleccionar *Almuerzo*  (ver figura 11).

Paso2: La aplicación mostrar un formulario en el cual debe seleccionar *Guardar* (ver figura 13) o *Cancelar* para volver al menú principal.

 **ALMUERZO**

Los campos con este simbolo (*) son obligatorios

Beneficiario *

Seleccionar tipo de beneficiario *

Observación

Figura 13: Guardar para registrar Almuerzo.
Fuente: DUCHITANGA, JULIO. 2016.

Nota: Se debe considerar que puede o no puede agregar una observación.

3.4.6. Registrar almuerzo para Ocasionales y Permanentes.

Recuerde que para ejecutar este proceso deberá seguir la misma temática de buscar al beneficiario y ejecutar la opción deseada descrita en pasos anteriores.

Paso1: Seleccionar Almuerzo (ver figura 9).

Paso2: La aplicación mostrará un formulario en el cual deberá seleccionar el tipo de beneficiario (Ocasional, Permanente o Posante) y seleccionar guardar (ver figura 14).

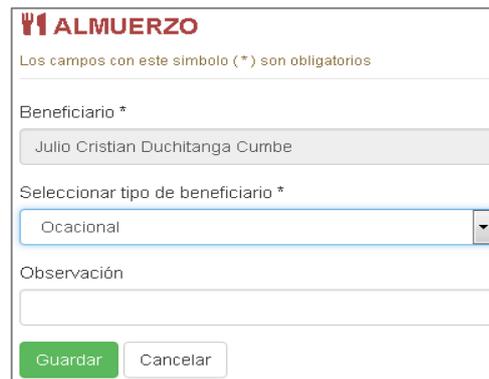


Figura 13: Registró de almuerzos.

Fuente: DUCHITANGA, JULIO. 2016.

Nota: En este proceso el formulario muestra por defecto *Ocasional* como tipo de beneficiario; recuerde que se debe seleccionar el más adecuado.

3.4.7. Registrar, modificar o eliminar funcionario.

Paso1: Seleccionar funcionarios en el menú de posada, luego seleccionar lista de funcionarios (ver figura 14).



Figura 14: Seleccionar lista de funcionarios.

Fuente: DUCHITANGA, JULIO. 2016.

Paso1: Ejecutada la acción, la aplicación mostrará la siguiente interfaz (ver figura 15) con tres opciones a ejecutar:

- ▲ Nuevo (registrar funcionario).
- ▲ Modificar.
- ▲ Eliminar.

LISTA DE FUNCIONARIOS		
Nombre y Apellido	N° Cédula	Acción
Buñay Guevara Richard Gerson	1710173145	<input type="button" value="Modificar"/> <input type="button" value="Eliminar"/>
Arévalo Torres Cruz Beatriz	1400831382	<input type="button" value="Modificar"/> <input type="button" value="Eliminar"/>
Duchitanga Cristian	1400738694	<input type="button" value="Modificar"/> <input type="button" value="Eliminar"/>
Aguilar Lojano Laura Mercedes	1400668370	<input type="button" value="Modificar"/> <input type="button" value="Eliminar"/>
<input type="button" value="Nuevo"/>		

Figura 15: Seleccionar Nuevo, Modificar o Eliminar.
Fuente: DUCHITANGA, JULIO. 2016.

Paso2: Seleccionar Nuevo para registrar un funcionario.

Ejecutada la acción SIGEGAS mostrará una ventana modal donde deberá ingresar los campos solicitados y enviar a guardar (ver figura 16).

REGISTRAR FUNCIONARIO

Los campos con este simbolo (*) son obligatorios

Cedula*

Nombre*

Apellido*

Figura 16: Registro de funcionarios.
Fuente: DUCHITANGA, JULIO. 2016.

Paso4: Seleccionar Modificar para actualizar información del funcionario.

Ejecutada la acción SIGEGAS mostrará una ventana modal donde deberá ingresar los campos solicitados y enviar a guardar (ver figura 16).

Paso5: Seleccionar Eliminar para borrar un funcionario.

Ejecutada la acción SIGEGAS mostrará una ventana modal de confirmación donde deberá aceptar o cancelar la acción (ver figura 17).

Esta seguro de realizar esta operación?

Figura 17: Eliminar funcionarios.
Fuente: DUCHITANGA, JULIO. 2016.

Nota: Recuerde este procedimiento es irreversible.

3.4.8. Registrar almuerzo para beneficiario.

Paso1: De la barra de menú de posada seleccionar **Funcionarios** luego **Registrar Almuerzo** (ver figura 18).



Figura 18: Seleccionar registrar almuerzo.

Fuente: DUCHITANGA, JULIO. 2016.

Paso2: Este proceso mostrará el siguiente formulario en el cual debe ingresar la ***cédula*** del funcionario para buscar si existe o no (ver figura 19).



Figura 19: Buscar funcionario.

Fuente: DUCHITANGA, JULIO. 2016.

Paso2: Si existe el funcionario existe, la aplicación mostrará los datos (ver figura 20) con dos opciones a ejecutar:

- ▲ Registrar almuerzo.
- ▲ Ver deuda.



Figura 20: Seleccionar registrar almuerzo o ver deuda.

Fuente: DUCHITANGA, JULIO. 2016.

Paso2: Seleccionar **Registrar Almuerzo**.

La aplicación mostrará el formulario para el registro de alimentación (ver figura 21).

Figura 21: Registro de almuerzo funcionarios.
Fuente: DUCHITANGA, JULIO. 2016.

Nota: El formulario calcula automáticamente el costo de acuerdo al almuerzo seleccionado, además permitirá enviar como deuda si así se lo requiere; existen dos botones de guardado:

- ▲ **Guardar y agregar otro** (guarda y retorna al mismo formulario; para registrar otro almuerzo para el mismo funcionario).
- ▲ **Terminar** (Guarda y retorna al formulario de búsqueda.)

3.4.9. Ver y cobrar deuda por beneficiario.

Paso1: Seleccionar Ver deuda (ver figura 20); Siguiendo los pasos descritos para la búsqueda de un beneficiario (ver figura 19).

Paso2: Ejecutado el procedimiento la aplicación mostrará la lista de deudas (ver figura 22) con una opción a ejecutar:

- ▲ Cobrar deuda

Fecha	Alimentación	Deuda	Acción
Jueves, 08 Sep 2016	SECO Y JUGO	1.50	\$
Jueves, 08 Sep 2016	COMPLETO	2.00	\$
TOTAL		3.50	\$

Figura 22: Lista de deuda por beneficiario.
Fuente: DUCHITANGA, JULIO. 2016.

Paso3: **Cobrar deuda**, hacer clic en cobrar (\$). SIGEGAS mostrara un mensaje (ver figura 23).

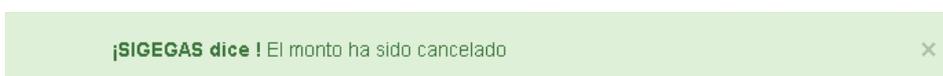


Figura 23: Deuda cancelada.
Fuente: DUCHITANGA, JULIO. 2016

3.4.10. Registrar habitaciones.

Paso 1: Seleccionar habitaciones en el menú de Posada, luego lista de habitaciones (ver figura 24)



Figura 24: Seleccionar lista de habitaciones.

Fuente: DUCHITANGA, JULIO. 2016

Paso 2: Ejecutado el paso anterior SIGEGAS mostrará la siguiente interfaz; donde debe seleccionar Nuevo (ver figura 25).

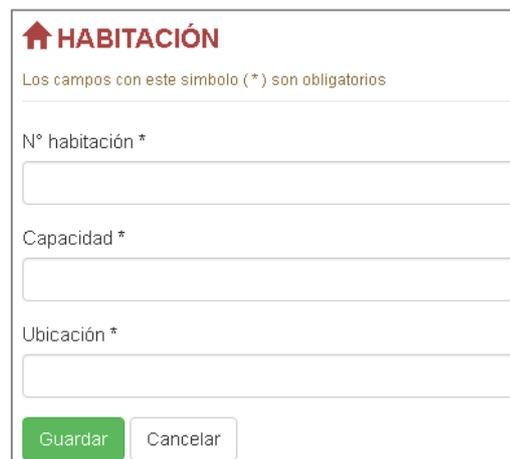


N° Habitación	Capacidad Actual	Ubicación
100	1	Primer piso
101	2	Primer piso
300	2	Planta baja

Figura 25: Seleccionar nuevo.

Fuente: DUCHITANGA, JULIO. 2016

Paso 3: SIGEGAS presentará un formulario, en el cual se debe llenar todos los campos requeridos. Luego seleccionar guardar o cancelar para volver a la lista de capacitaciones (ver figura 26).



HABITACIÓN

Los campos con este simbolo (*) son obligatorios

N° habitación *

Capacidad *

Ubicación *

Guardar Cancelar

Figura 26: Llenar campos y guardar.

Fuente: DUCHITANGA, JULIO. 2016

3.4.11. Reportes del módulo Posada y Comedor.

Todos los reportes serán generados en formato PDF.

3.4.11.1. Lista de beneficiario de comedor del mes actual.

Paso 1: En la interfaz principal de posada hacer clic en el icono  (ver figura 27).



Figura 27: Total personas del mes actual.

Fuente: DUCHITANGA, JULIO. 2016

Nota: el documento se mostrará con o sin información dependiendo si existe datos. Paso el último día del mes los datos no se mostraran, e iniciara la cuenta para el nuevo mes.

3.4.11.2. Lista de posantes inactivos del mes actual.

Paso1: En la interfaz principal de posada hacer clic en el icono  (ver figura 28).

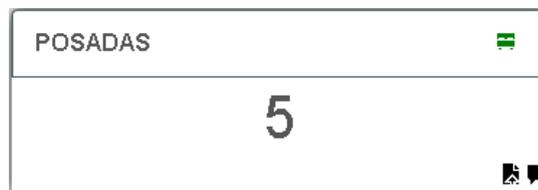


Figura 28: Total posantes del mes actual.

Fuente: DUCHITANGA, JULIO. 2016

Nota: Pasado el mes actual, mostrará la información perteneciente al siguiente mes.

3.4.11.3. Lista comedor funcionarios (diario).

Paso1: En la interfaz principal de posada hacer clic en el icono  (ver figura 29).

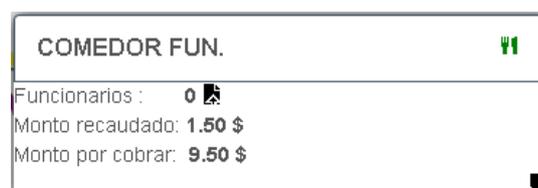


Figura 29: Lista diaria comedor Fun.

Fuente: DUCHITANGA, JULIO. 2016

Nota: Muestra la lista de los funcionarios que accedieron a comedor diariamente.

3.4.11.4. *Liste de personas (permanentes).*

Paso 1: En el menú principal de posada seleccionar **Beneficiarios** luego **lista de permanentes** (ver figura 30).



Figura 30: Seleccionar lista de permanentes.

Fuente: DUCHITANGA, JULIO. 2016

Paso 2: SIGEGAS lista las personas que comieron en el mes, además las veces que comieron. Con opción a ver un detalle por persona en formato PDF para ello hacer clic en **detalle** (ver figura 31).



Nombre y apellidos	Cédula	Almuerzos	Acción
Christian Duchitanga	1450300551	1	Detalle
Gloria Esperanza Castro Samaniego	1401199300	1	Detalle

Figura 31: Número de veces que comió un beneficiario en el mes actual.

Fuente: DUCHITANGA, JULIO. 2016

3.4.11.5. *Lista de personas (posantes).*

Paso 1: Seleccionar **Lista de Posantes** siguiendo el proceso del apartado 3.4.11.4 (ver figura 30).

Paso 2: SIGEGAS listará los posantes del mes y las veces que se hospedaron con opción a ver el detalle de cada posante; para ello hacer clic en **detalle.** (Ver figura 32)



Nombre y apellidos	Cédula	Posadas	Acción
Gloria Esperanza Castro Samaniego	1401199300	1	Detalle

N° registros 1

Figura 32: número de posada por beneficiario.

Fuente: DUCHITANGA, JULIO. 2016

3.4.11.6. *Lista de deudores.*

Paso 1: Seleccionar **Lista de deudores** del menú de Posada.

Paso 2: SIGEGAS mostrará la lista de deudores con el monto de deuda con opción a cancelar deuda. (Ver figura 33)

LISTA DE DEUDORES			
Nombre y apellidos	Cédula	Deuda	Acción
Aguilar Lojano Laura Mercedes	1400668370	2.00 \$	Cobrar
Arévalo Torres Cruz Beatriz	1400831382	4.00 \$	Cobrar
Buñay Guevara Richard Gerson	1710173145	3.50 \$	Cobrar
Total por cobrar		9.50 \$	

Figura 33: Deudores.
Fuente: DUCHITANGA, JULIO. 2016

3.5. Módulo de Ayuda Social.

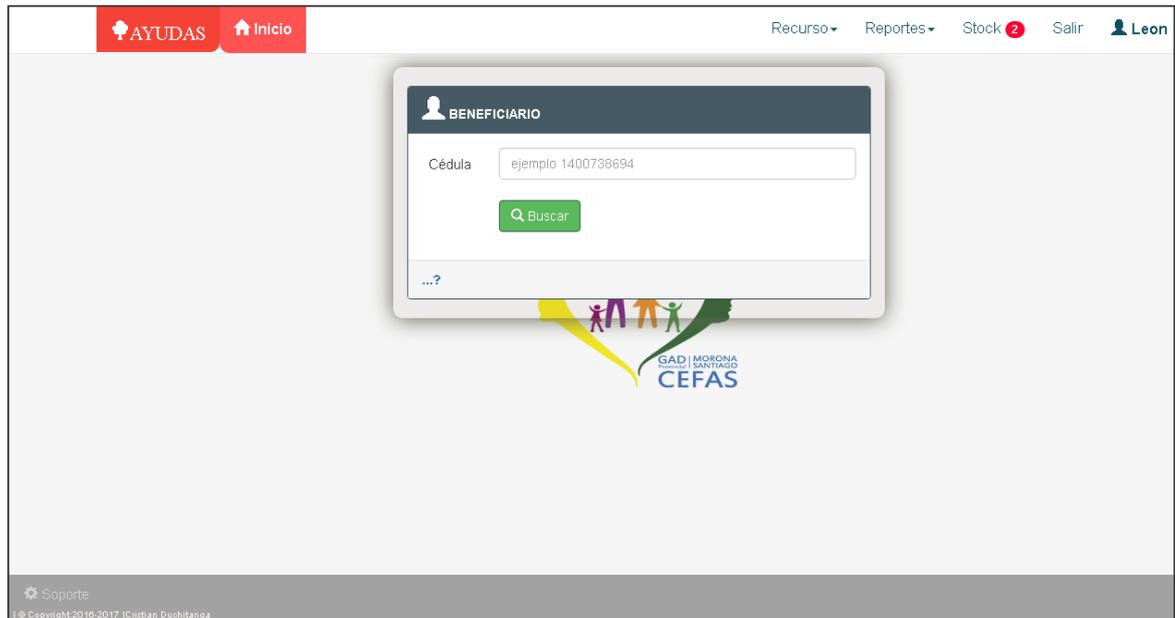


Figura 33: Página principal de Ayuda Social.
Fuente: DUCHITANGA, JULIO. 2016

3.5.1. Crear Ayuda Social

Paso 1: Buscar al beneficiario ingresando la cédula (ver figura 34).

Figura 33: Ingresar cédula.
Fuente: DUCHITANGA, JULIO. 2016

Paso2: SIGEGAS mostrará la información del beneficiario, luego seleccionar Crear Ayuda. (Ver figura 35).

Nombres: Julio Cristian Duchitanga Cumbe
Cedula: 1400738694
Edad: 25
Nacionalidad: Mestizo
Estado Civil: Soltero/a
Instrucción: Superior
Ocupación: Desarrollador
Discapacidad: Ninguna
Domicilio: Cantón Morona, Parroquia Sinaí, Comunidad Sinaí

Figura 35: Seleccionar crear ayuda.
Fuente: DUCHITANGA, JULIO. 2016

Paso 3: Llenar el formulario de Ayudas y guardar o cancelar para volver a página principal.
(Ver figura 36).

Ayuda Social

Los campos con este simbolo (*) son obligatorios

Beneficiario *

Julio Cristian Duchitanga Cumbe

Seleccionar ayuda *

Muletas axilares para niño/a

Agregar cantidad *

2

Se recomienda "ver stok" antes de agregar cantidad.

Agregar observación

valido para 3 meses|

Figura 36: Llenar formulario y guardar.
Fuente: DUCHITANGA, JULIO. 2016

3.5.2. Historial de ayudas

Paso1: Seleccionar Historial; seguir el paso 1 y paso2 del apartado 3.5.1 (ver figura 35).

Paso2: SIGEGAS presentará la listas de ayudas entregadas por beneficiario (ver figura 37).

Julio Cristian Duchitanga Cumbe					
TIPO	RECURSO ENTREGADO	CANTIDAD	OBSERVACIÓN	FECHA	ACIÓN
Ayuda Tecnica	Muletas axilares para niño/a	1		15 de Octubre de 2016	
Ayuda Tecnica	Muletas axilares para niño/a	1		15 de Octubre de 2016	
Ayuda Tecnica	Sillas para baño	2		15 de Octubre de 2016	
Ayuda Tecnica	Muletas axilares para niño/a	1	Entregado para un año	10 de Octubre de 2016	
Ayuda Tecnica	Sillas posturales PCI para adulto	1		8 de Octubre de 2016	

Figura 37: Historial.

Fuente: DUCHITANGA, JULIO. 2016

Nota: Registrar ayuda permite acceder a crear ayuda.

3.5.3. Agregar y editar Recurso

Paso1: seleccionar Recurso luego agregar nuevo en el menú de Ayuda. (Ver figura 38).



Figura 38: Seleccionar agregar nuevo.

Fuente: DUCHITANGA, JULIO. 2016

Paso2: Llenar el formulario y guardar o cancelar para volver a página principal. (Ver figura 39)

RECURSO

Los campos con este simbolo (*) son obligatorios

Seleccionar tipo de artículo *

Nombre de artículo *

Seleccionar categoría *

Se puede dejar vacío este campo; cuando el artículo no tenga categoría.

Agregar cantidad *

Agregar descripción

Figura 39: Llenar y guardar.

Fuente: DUCHITANGA, JULIO. 2016

Paso 3: Seleccionar Recurso luego Editar recurso en el menú de Ayuda.

Paso 4: SIGEGAS visualizará los recursos almacenados; Hacer clic en icono  para editar un recurso en específico. (Veri figura 40)



Tipo	Nombre	Stock	Descripción	Acción
Ayuda Tecnica	Muletas axilares para niño/a	1	Pares	
Ayuda Tecnica	Choches para niño/a	10		

Figura 40: Seleccionar editar de la lista de recursos.

Fuente: DUCHITANGA, JULIO. 2016

3.5.4. Aumentar Stock

Paso 1: seleccionar Stock en el menú de Ayudas (ver figura 41); SIGEGAS lista los recursos con stock cero.



Figura 41: Seleccionar Stock.

Fuente: DUCHITANGA, JULIO. 2016

Paso 2: De la lista de recursos mostrados hacer clic en el icono  para aumentar stock. (Ver figura 42)

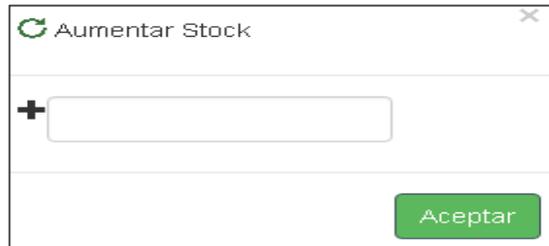


Figura 42: Ingresar valor y aceptar.

Fuente: DUCHITANGA, JULIO. 2016

3.5.5. Reportes para ayuda social.

3.5.5.1 Recursos.

Paso 1: seleccionar Reportes luego Recursos en el menú de ayudas.

SIGEGAS mostrará la información en formato PDF.

3.5.5.1 Ayudas.

Paso1: En el menú de Ayudas seleccionar Reportes luego Ayudas.

Paso 2: SIGEGAS mostrará in formulario, donde bene seleccionar la ayuda (ver figura 43).

Figura 42: Generar reporte.
Fuente: DUCHITANGA, JULIO. 2016

Nota: el reporte será presentado en formato de PDF

3.6. Modulo Capacitaciones (Técnico)

Taller	Comunidad	Parroquia	Canton	Inició	Finalizó	Capacitador	Accion
Gastronomia	Sinai	Sinai	Morona	1 de Septiembre de 2016	2 de Septiembre de 2016	Cristian Duchitanga	[Grid] [Checkmark]
Gastronomia	Sinai	Sinai	Morona	5 de Octubre de 2016	7 de Octubre de 2016	Cristian Duchitanga	[Grid] [Checkmark]

Figura 43: Menú principal de Capacitaciones (Técnico).
Fuente: DUCHITANGA, JULIO. 2016

3.6.1 Registrar capacitación

Paso 1: En el menú de capacitaciones seleccionar Capacitaciones luego agregar nuevo.

Paso 2: Llenar el formulario y guardar o cancelar para volver al menú principal (ver figura 44)

Figura 44: Llenar y guardar.
Fuente: DUCHITANGA, JULIO. 2016

3.6.2 Publicar capacitación

Paso 1: En el menú del técnico seleccionar publicar.

SUGEGAS visualizará la lista de capacitaciones a ser publicadas con tres opciones:

- ▲ Publicar 
- ▲ Editar
- ▲ Eliminar

Paso 2: Hacer clic en el icono  para publicar capacitación (ver figura 45).

Taller: Gastronomía
Capacitador: Cristian Duchitanga
Inicia: 6 de Octubre de 2016
Finaliza: 13 de Octubre de 2016
Lugar: Macas

Figura 45: Si para publicar.
Fuente: DUCHITANGA, JULIO. 2016

3.6.3 Reportes Técnico

3.6.3.1 Generar Certificado

Paso 1: en la página principal hacer clic en icono  para listar las personas.

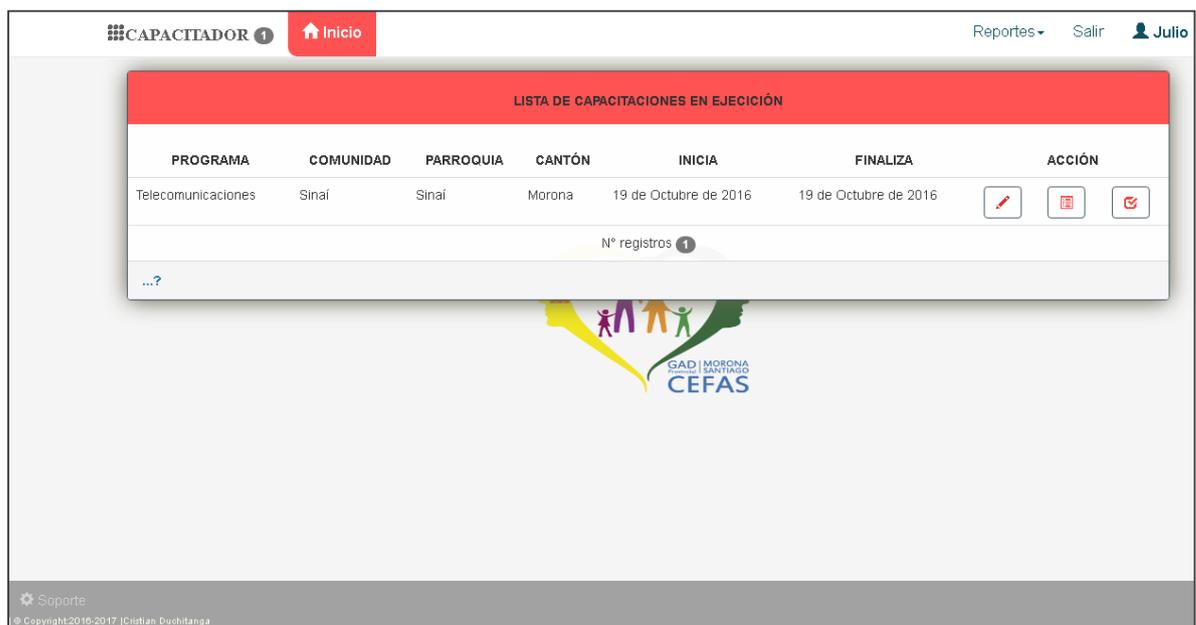
Paso 2: Hacer clic en el icono  para generar certificado. (Ver figura 46)

PARTICIPANTES APROBADOS				
Cedula	Nombre	Edad	Procedencia	Acción
1400738694	Julio Cristian	25	Sinaí	
1401199300	Gloria Esperanza	23	Sinaí	
Taller de Gastronomía en Sinaí				

Figura 45: Generar certificado.

Fuente: DUCHITANGA, JULIO. 2016

3.6. Modulo Capacitaciones (Capacitador)



The screenshot shows the 'CAPACITADOR' interface. At the top, there is a navigation bar with 'Inicio', 'Reportes', 'Salir', and a user profile 'Julio'. The main content area displays a table titled 'LISTA DE CAPACITACIONES EN EJECICIÓN'. The table has columns for 'PROGRAMA', 'COMUNIDAD', 'PARROQUIA', 'CANTÓN', 'INICIA', 'FINALIZA', and 'ACCIÓN'. A single record is visible for 'Telecomunicaciones' in 'Sinaí' parish, 'Morona' canton, starting and ending on '19 de Octubre de 2016'. The 'ACCIÓN' column contains icons for edit, print, and refresh. Below the table, it indicates 'N° registros 1' and a search icon. The footer includes 'Soporte' and '© Copyright 2016-2017 | Cristian Duchitanga'.

PROGRAMA	COMUNIDAD	PARROQUIA	CANTÓN	INICIA	FINALIZA	ACCIÓN
Telecomunicaciones	Sinaí	Sinaí	Morona	19 de Octubre de 2016	19 de Octubre de 2016	  

Figura 46: Página principal de Capacitadores.

Fuente: DUCHITANGA, JULIO. 2016

3.6.1 Registrar participantes

Paso 1: En el menú principal hacer clic en el icono 

Paso 2: En la interfaz mostrada, ingresar la cedula del beneficiario para ver si se encuentra en el sistema (ver figura 47)

INSCRIPCIÓN A TELECOMUNICACIONES

Ingresar cédula

Cédula	Nombres y Apellidos	Acción
1400738694	Julio Cristian Duchitanga Cumbe	 Inscribir!!

Lugar: Sinaí Horas: 10

...?

Figura 46: Ingresar Cédula.
Fuente: DUCHITANGA, JULIO. 2016

Paso 3: Hacer clic en  Inscribir.

SIGEGAS mostrara interfaz con información detallada de la de la inscripción. (Ver figura 47)

 SIGEGAS

El beneficiario será inscrito al taller de: Telecomunicaciones.

Figura 47: Aceptar para inscribir.
Fuente: DUCHITANGA, JULIO. 2016

3.6.2 Aprobar o Reprobar Participantes

Paso 1: En el menú principal hacer clic en el icono.

SIGEGAS visualizará una interfaz con los beneficiarios registrados en la capacitación seleccionada.

Paso2: Hacer clic en icono  para aprobar o en el icono  para reprobar (ver figura 48)

PERSONAS REGISTRADAS					
Cedula	Nombres y Aellidos	Procedencia	Edad	Autodfinición	Acción
1400738694	Julio Cristian Duchitanga Cumbe	Sinaí	25	Mestizo	 
1401199300	Gloria Esperanza Castro Samaniego	Sinaí	23	Mestizo	 

Taller de Telecomunicaciones en Sinaí

Figura 48: Aprobar o Reprobar.
Fuente: DUCHITANGA, JULIO. 2016

3.6.3 Reportes de capacitadores

3.6.3.1 Participantes por capacitación

Paso 1: En el menú de capacitadores seleccionar Reportes.

SIGEGAS mostrará la lista de capacitaciones desarrolladas.

Paso 2: Hacer clic en el icono  para generar el reporte. (Ver figura 49)

LISTA DE CAPACITACIONES FINALIZADAS						
PROGRAMA	COMUNIDAD	PARROQUIA	CANTÓN	FECHA	HORAS	ACCIÓN
Peluquería	Sinaí	Sinaí	Morona	Del 6 al 7 Sep 2016	10	
Peluquería	Sinaí	Sinaí	Morona	Del 29 al 30 Sep 2016	10	

Figura 48: Generar reporte por capacitación.

Fuente: DUCHITANGA, JULIO. 2016