



ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO

FACULTAD DE INFORMÁTICA Y ELECTRÓNICA

CARRERA SOFTWARE

**IMPLEMENTACIÓN DE UN SOFTWARE EDUCATIVO PARA
REFORZAR EL APRENDIZAJE DE GEOMETRÍA Y MEDIDA EN
ESTUDIANTES DE 1RO DE BACHILLERATO**

Trabajo de Integración Curricular

Tipo: Proyecto Técnico

Presentado para optar al grado académico de:

INGENIERO DE SOFTWARE

AUTOR: JOHN CRISTÓBAL SILVA ESPÍN

DIRECTOR: DR. DIEGO FERNANDO AVILA PESANTEZ

Riobamba – Ecuador

2022

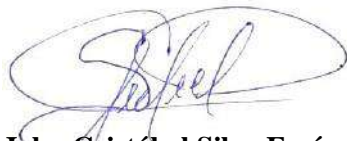
© 2022, John Cristóbal Silva Espín.

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

Yo, John Cristóbal Silva Espín, declaro que el presente Trabajo de Integración Curricular es de autoría propia y los resultados de este son auténticos. Los textos en el documento que provienen de otras fuentes están debidamente citados y referenciados.

Como autor asumo la responsabilidad legal y académica de los contenidos de este Trabajo de Integración Curricular; el patrimonio intelectual pertenece a la Escuela Superior Politécnica de Chimborazo.

Riobamba, 30 de noviembre de 2022

A handwritten signature in blue ink, appearing to read 'John', enclosed within a large, loopy oval shape.

John Cristóbal Silva Espín


1850122415

ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO

FACULTAD DE INFORMÁTICA Y ELECTRÓNICA

CARRERA SOFTWARE

El Tribunal del Trabajo de Integración Curricular certifica que: Trabajo de Integración Curricular; tipo: Proyecto Técnico, **IMPLEMENTACIÓN DE UN SOFTWARE EDUCATIVO PARA REFORZAR EL APRENDIZAJE DE GEOMETRÍA Y MEDIDA EN ESTUDIANTES DE 1RO DE BACHILLERATO**, realizado por el señor: **JOHN CRISTÓBAL SILVA ESPÍN**, ha sido minuciosamente revisado por los Miembros del Tribunal del Trabajo de Integración Curricular, el mismo que cumple con los requisitos científicos, técnicos, legales, en tal virtud el Tribunal Autoriza su presentación.

	FIRMA	FECHA
Dr. Julio Roberto Santillán Castillo PRESIDENTE DEL TRIBUNAL		2022-11-30
Dr. Diego Fernando Avila Pesantez DIRECTOR DEL TRABAJO DE INTEGRACIÓN CURRICULAR		2022-11-30
Ing. Miguel Ángel Duque Vaca ASESOR DEL TRABAJO DE INTEGRACIÓN CURRICULAR		2022-11-30

DEDICATORIA

Dedico este trabajo a mi familia, especialmente a mi madre Patricia y abuelita Mariana, gracias a sus enseñanzas, su esfuerzo, su trabajo y completo apoyo puedo culminar una anhelada meta. A mis queridos amigos y compañeros que han estado a lo largo del camino, con quienes he compartido buenos momentos, y me han entregado su confianza y ayuda; a todos ellos mi eterna gratitud.

John

AGRADECIMIENTO

Agradezco inmensamente a mi familia y amigos por todo su apoyo y confianza que me permitió avanzar y alcanzar muchas metas. A mi querida institución la Escuela Superior Politécnica de Chimborazo, por darme una excelente formación, a todos los docentes quienes compartieron su conocimiento y valores. Por último, un especial agradecimiento a mi director Ing. Diego Ávila y asesor Ing. Miguel Duque por su guía, conocimientos y experiencias que contribuyeron en la realización de este Trabajo de Integración Curricular.

John

TABLA DE CONTENIDOS

ÍNDICE DE TABLAS.....	x
ÍNDICE DE FIGURAS.....	xi
ÍNDICE DE GRÁFICOS.....	xii
ÍNDICE DE ANEXOS	xiii
RESUMEN.....	xiv
ABSTRACT	xv
INTRODUCCIÓN	1

CAPÍTULO I

1. DIAGNÓSTICO DEL PROBLEMA.....	3
1.1. Antecedentes	3
1.1.1. <i>Formulación del problema</i>	4
1.1.2. <i>Sistematización del problema</i>	4
1.2. Justificación	5
1.2.1. <i>Justificación teórica</i>	5
1.2.2. <i>Justificación aplicativa</i>	6
1.3. Objetivos	7
1.3.1. <i>Objetivo General</i>	7
1.3.2. <i>Objetivos Específicos</i>	7

CAPÍTULO II

2. FUNDAMENTOS TEÓRICOS	8
2.1. Aprendizaje y Geometría.....	8
2.1.1. <i>Conceptos de Aprendizaje</i>	8
2.1.2. <i>Teorías de Aprendizaje</i>	8
2.1.3. <i>Tipos de Aprendizaje</i>	9
2.1.3.1. <i>Aprendizaje Significativo</i>	10
2.1.3.2. <i>Aprendizaje Memorístico</i>	10

2.1.4.	<i>Fundamentos de Geometría</i>	10
2.1.5.	<i>Geometría en la educación</i>	10
2.2.	Software Educativo	11
2.2.1.	<i>Clasificación de los softwares educativos</i>	11
2.2.2.	<i>Características del software educativo</i>	11
2.2.3.	<i>Organización y presentación de contenidos en un software educativo</i>	12
2.3.	Metodología de Desarrollo de Software	12
2.3.1.	<i>Scrum</i>	12
2.3.2.	<i>Roles de Scrum</i>	13
2.3.3.	<i>Artefactos de Scrum</i>	13
2.3.4.	<i>Eventos de Scrum</i>	13
2.3.5.	<i>Características de Scrum</i>	14
2.3.6.	<i>Scrum para el desarrollo de software educativos</i>	14
2.4.	Arquitectura de Software	15
2.4.1.	<i>Jamstack</i>	15
2.4.2.	<i>RESTful API</i>	16
2.5.	Herramientas de Desarrollo de Software	17
2.5.1.	<i>Aplicaciones basadas en la WEB</i>	17
2.5.2.	<i>PostgreSQL</i>	17
2.5.3.	<i>Sistema de Gestión de Contenido sin cabeza</i>	17
2.5.3.1.	<i>Strapi</i>	18
2.5.4.	<i>Frameworks – Frontend</i>	18
2.5.4.1.	<i>NextJS</i>	18
2.5.5.	<i>Herramientas utilitarias</i>	18
2.5.5.1.	<i>Visual Studio Code</i>	18
2.5.5.2.	<i>Postman</i>	18
2.5.5.3.	<i>GeoGebra</i>	19
2.6.	Norma ISO/IEC 25010	19
2.6.1.	<i>Características de la ISO/IEC 25010</i>	19
2.6.2.	<i>Usabilidad</i>	21
2.7.	Trabajos Relacionados	21

CAPÍTULO III

3.	MARCO METODOLÓGICO.....	23
3.1.	Tipo de estudio.....	23
3.1.1.	<i>Métodos, técnicas, fuentes e instrumentos de estudio</i>	23
3.1.2.	<i>Método Analítico</i>	24
3.1.3.	<i>Método Sintético</i>	25
3.1.4.	<i>Metodología Scrum</i>	25
3.1.5.	<i>Método Inductivo</i>	25
3.1.6.	<i>Método Estadístico.....</i>	25
3.2.	Determinación de la Usabilidad	25
3.2.1.	<i>Operación Conceptual</i>	25
3.2.2.	<i>Operación Metodológica</i>	26
3.2.2.1.	<i>Características de usabilidad a evaluar</i>	27
3.2.3.	<i>Población y Muestra</i>	27
3.2.4.	<i>Herramienta para la evaluación de la usabilidad</i>	28
3.2.5.	<i>Proceso de evaluación de la Usabilidad.....</i>	29
3.3.	Desarrollo del software educativo utilizando la metodología Scrum.....	30
3.3.1.	<i>Análisis previo al desarrollo del software educativo</i>	30
3.3.1.1.	<i>Estudio de factibilidad.....</i>	30
3.3.1.2.	<i>Gestión de riesgos</i>	32
3.3.2.	<i>Fase de planificación.....</i>	33
3.3.2.1.	<i>Personas y Roles en el proyecto</i>	34
3.3.2.2.	<i>Tipos de usuarios del software educativo</i>	34
3.3.2.3.	<i>Estimación</i>	34
3.3.2.4.	<i>Product Backlog</i>	35
3.3.2.5.	<i>Historias técnicas</i>	36
3.3.2.6.	<i>Historias de usuario</i>	37
3.3.2.7.	<i>Spring Backlog</i>	38
3.3.3.	<i>Fase de desarrollo del software educativo</i>	39
3.3.3.1.	<i>Definición de la arquitectura del software educativo</i>	39
3.3.3.2.	<i>Definición del estándar de codificación.....</i>	40
3.3.3.3.	<i>Diagramas UML.....</i>	41
3.3.3.4.	<i>Diseño de la base de datos</i>	43
3.3.3.5.	<i>Diseño de las interfaces.....</i>	47
3.3.3.6.	<i>Reuniones y Entregables</i>	50

3.3.3.7. <i>Fase de pruebas</i>	51
3.3.4. <i>Fase de cierre</i>	55
3.3.4.1. <i>Gestión del proyecto</i>	55

CAPÍTULO IV

4. RESULTADOS	56
4.1. Evaluación de las subcaracterísticas de usabilidad	56
4.2. Análisis de la usabilidad del software educativo	58
4.3. Determinación de la calidad del software educativo	59
CONCLUSIONES	61
RECOMENDACIONES	62
BIBLIOGRAFÍA	
ANEXOS	

ÍNDICE DE TABLAS

Tabla 1-2: Características de las teorías de aprendizaje.....	9
Tabla 2-2: Características de la ISO/IEC 25010	20
Tabla 1-3: Métodos, Técnicas y Fuentes para el trabajo de Integración Curricular	24
Tabla 2-3: Operacionalización conceptual de variables.....	26
Tabla 3-3: Operacionalización metodológica de variables	26
Tabla 4-3: Características de usabilidad a evaluar	27
Tabla 5-3: Cuestionario final para la evaluación del software educativo	29
Tabla 6-3: Escala de Likert para la evaluación	29
Tabla 7-3: Recursos hardware.....	30
Tabla 8-3: Recursos software.....	31
Tabla 9-3: Costos de los recursos para el desarrollo.....	31
Tabla 10-3: Riesgos identificados.....	32
Tabla 11-3: Plan de Trabajo.....	33
Tabla 12-3: Personas y roles	34
Tabla 13-3: Tipos de usuarios.....	34
Tabla 14-3: Puntos Estimados T-shirt.....	35
Tabla 15-3: Product Backlog	35
Tabla 16-3: Historia técnica.....	37
Tabla 17-3: Historia de usuario.....	38
Tabla 18-3: Sprint Backlog	39
Tabla 19-3: Estándar de codificación para NextJS	41
Tabla 20-3: Estándar de codificación para Strapi	41
Tabla 21-3: Diccionario de datos	47
Tabla 22-3: Reuniones y Entregables	51
Tabla 23-3: Prueba de Aceptación.....	54
Tabla 1-4: Resultados: Facilidad pedagógica	56
Tabla 2-4: Resultados: Facilidad de entendimiento o comprensibilidad	57
Tabla 3-4: Resultados: Facilidad de uso	57
Tabla 4-4: Ponderación de la usabilidad	58
Tabla 5-4: Indicadores de calidad	59

ÍNDICE DE FIGURAS

Figura 1-2: Pasos de Scrum	13
Figura 2-2: Diagrama de la metodología Scrum para el desarrollo de videojuegos	15
Figura 3-2: Arquitectura Jamstack.....	16
Figura 1-3: Inicio de sesión	48
Figura 2-3: Pantalla principal (Listado de clases).....	49
Figura 3-3: Pantalla de la clase (Geometría dinámica).....	50
Figura 4-3: Suite de pruebas en TUSKR	52
Figura 5-3: Ejecución del caso de prueba “User account creation”.....	53
Figura 6-3: Resultado de la ejecución de los casos de pruebas.	53

ÍNDICE DE GRÁFICOS

Gráfico 1-3: Arquitectura del software educativo.....	40
Gráfico 2-3: Diagrama de clases	42
Gráfico 3-3: Diagrama de casos de uso Strapi	43
Gráfico 4-3: Diseño conceptual de la base de datos	44
Gráfico 5-3: Diseño lógico de la base de datos.....	45
Gráfico 6-3: Diseño físico de la base de datos.....	46
Gráfico 7-3: Burn Down Chart	55
Gráfico 1-4: Representación de las subcaracterísticas de usabilidad.....	58
Gráfico 2-4: Nivel de usabilidad alcanzado.....	59

ÍNDICE DE ANEXOS

- ANEXO A:** Análisis de factibilidad
- ANEXO B:** Gestión de riesgos
- ANEXO C:** Planificación del proyecto
- ANEXO D:** Historias técnicas
- ANEXO E:** Historias de usuario
- ANEXO F:** Estándares de codificación
- ANEXO G:** Diagramas UML
- ANEXO H:** Diccionario de datos
- ANEXO I:** Diseño de interfaces
- ANEXO J:** Plan de pruebas
- ANEXO K:** Pruebas de aceptación
- ANEXO L:** Cuestionario utilizado

RESUMEN

En el presente trabajo de integración curricular se estableció como objetivo principal la implementación de un software educativo que apoye el proceso de aprendizaje de geometría y medida en estudiantes de primero de bachillerato. Se utilizó la metodología de desarrollo ágil Scrum buscando flexibilidad, adaptabilidad y rapidez durante el proyecto. A través de reuniones con las partes interesadas se definieron los requerimientos funcionales y no funcionales. La arquitectura utilizada fue Jamstack aprovechando sus ventajas de velocidad, seguridad y bajo coste. Dentro de las herramientas empleadas para el desarrollo está el sistema de gestión de contenido sin cabeza Strapi, el generador de sitios estáticos NextJS, S3 de AWS para el almacenamiento de archivos, SendGrid para la gestión de correos electrónicos y Vercel para el despliegue. Para garantizar el cumplimiento de los requerimientos se realizaron pruebas de funcionalidades basadas en las historias de usuario y pruebas de aceptación. Finalmente se evaluó la característica de Usabilidad siguiendo la Norma ISO/IEC 25010 mediante la aplicación de una encuesta de 15 preguntas adaptadas de los cuestionarios USE y PSSUQ, evaluando las características de facilidad pedagógica, facilidad de entendimiento y facilidad de uso. Se tabularon y analizaron los datos obtenidos de la encuesta dando como resultado una puntuación de 4.51 para la facilidad pedagógica, 4.26 para facilidad de entendimiento y 4.28 para la facilidad de uso, todas sobre 5, concluyendo que el software educativo cumple los requerimientos establecidos, resulta atractivo y facilita el aprendizaje de geometría y medida en los estudiantes. Se recomienda utilizar técnicas de diseño responsivo para que el software educativo pueda adaptarse a diferentes dispositivos.

Palabras clave: <INGENIERÍA DE SOFTWARE>, <SCRUM>, <SOFTWARE EDUCATIVO>, <GEOMETRÍA DINÁMICA>, <JAMSTACK>, <STRAPI>, <NEXTJS>, <USABILIDAD>



ABSTRACT

The main objective of this curriculum integration work was the implementation of an educational software that supports the learning process of geometry and measurement in first year high school students. We utilized the agile development methodology Scrum, seeking flexibility, adaptability, and speed during the project. Functional and non-functional requirements were defined through meetings with stakeholders. We utilized the Jamstack architecture, taking advantage of its speed, security, and low cost. Among the tools utilized for development were the headless content management system Strapi, NextJS static site generator, AWS S3 for file storage, SendGrid for email management and Vercel for deployment. To ensure the fulfillment of the requirements, functional tests based on user stories and acceptance tests were performed. Finally, we evaluated the Usability characteristic following the ISO/IEC 25010 Standard by applying a 15-question survey adapted from the USE and PSSUQ questionnaires, evaluating the characteristics of pedagogical ease, ease of understanding and ease of use. We tabulated and analyzed the data obtained from the survey, resulting in a score of 4.51 for pedagogical ease, 4.26 for ease of understanding and 4.28 for ease of use, all out of 5. We conclude that the educational software meets the established requirements, is attractive and facilitates the learning of geometry and measurement in students. It is recommended to use responsive design techniques so that the educational software can be adapted to different devices.

Key words: <SOFTWARE ENGINEERING>, <EDUCATIONAL SOFTWARE>, <DYNAMIC GEOMETRY>, <AGILE SOFTWARE DEVELOPMENT METHODOLOGY (SCRUM)>, <WEB DEVELOPMENT ARCHITECTURE JAMSTACK>, <CONTENT MANAGEMENT SYSTEM STRAPI>, <NEXTJS FRAMEWORK >, <USABILITY>.

Revised by:



Lic. Carolina Campaña D. Mgs.

ID number: 1804191482

EFL Professor

2376-DBRA-UPT-2022

INTRODUCCIÓN

En la actualidad la integración de nuevas tecnologías en la educación es un proceso clave y necesario, debido a que influye de varias maneras en cómo se transfiere el conocimiento en los entornos educativos. Existe una gran cantidad de herramientas tecnológicas que tienen como objetivo facilitar el proceso de enseñanza y aprendizaje denominadas software educativo, cuentan con características o funciones específicas para ayudar el desarrollo de habilidades cognitivas. Sin embargo, existe poca integración de la tecnología en las aulas, debido al desconocimiento o baja formación por parte de los docentes.

A menudo las matemáticas son consideradas una de las áreas más difíciles por parte de los estudiantes, y aumenta si se habla específicamente de geometría y medida, en parte por la exigencia de su lenguaje que es gráfico y simbólico, y requiere un alto nivel de procesos mentales relacionados con la construcción de elementos geométricos, el razonamiento y la visualización espacial. Al utilizar métodos tradicionales se obliga a los estudiantes a la memorización forzada de conceptos, formulas, teoremas que son difíciles de imaginar o realizar, lo que genera un rechazo a la materia por parte de estudiantes y docentes. Una alternativa atractiva para los estudiantes son los sistemas de geometría dinámica que permiten la construcción, visualización y manipulación de elementos geométricos.

A pesar del gran número de sistemas de geometría dinámica que se pueden utilizar de manera gratuita, no se adaptan de manera completa con los contenidos propuestos en el libro de matemáticas de primero de bachillerato. Como solución a este problema se implementa un software educativo aplicando conceptos y herramientas de geometría dinámica para complementar el proceso de aprendizaje permitiendo una adquisición de conocimiento y habilidades significativas para los estudiantes.

Este documento se divide en cuatro capítulos que se detallan a continuación:

CAPÍTULO I: Corresponde al diagnóstico del problema, los antecedentes, la justificación y los objetivos planteados.

CAPÍTULO II: Expone los fundamentos teóricos sobre geometría, la educación, software educativo, metodologías de desarrollo, arquitectura del sistema, y las herramientas de desarrollo utilizadas en el proyecto.

CAPÍTULO III: Detalla el tipo de estudio, métodos y técnicas empleadas para el desarrollo y evaluación del software educativo.

CAPÍTULO IV: Presenta el análisis de resultados obtenidos en base a los objetivos planteados, las conclusiones y recomendaciones.

CAPITULO I

1. DIAGNÓSTICO DEL PROBLEMA

2.1. Antecedentes

Las matemáticas son vistas por los estudiantes como una de las áreas más difíciles, especialmente el apartado de geometría y medida porque requiere una exigencia mayor en las fórmulas y cálculos que se deben realizar. Además, la metodología o estrategia que utilizan los maestros para impartir estos temas no es atractiva y no despierta su interés, por lo cual los estudiantes tienen problemas de rendimiento académico (Mota, Oliveira y Henriques 2017).

Al analizar detenidamente las metodologías tradicionales de enseñanza de geometría, se observa que se centra en el aprendizaje forzoso o memorización completa de las fórmulas, definiciones, teoremas, y propiedades, generando una enseñanza mecánica sin fomentar el desarrollo de las destrezas necesarias para la resolución de problemas, además en algunas investigaciones se evidencia como seguir utilizando lápiz, papel y una pizarra está alejando a los estudiantes de comprometerse a estudiar (Gamboa and Ballesteros 2010).

Con este antecedente, se presentan problemas como el bajo rendimiento académico, que junto a una motivación limitada están llevando a los maestros y estudiantes a tener un rechazo a la materia, omitiéndola o realizando un estudio demasiado superficial. Sin embargo, es importante considerar que dejar de lado los contenidos geométricos impiden a los estudiantes desarrollar ciertas destrezas que les resultará útiles a lo largo de su vida estudiantil. En este sentido, (Díaz-Nunja, Rodríguez-Sosa y Lingán 2018) establece que el aprendizaje de la geometría involucra, como mínimo, realizar tres actividades cognitivas: la construcción, que alude al diseño de configuraciones mediado por instrumentos geométricos; el razonamiento, vinculado con los procesos discursivos y de argumentación; y la visualización, que se enfoca en las representaciones espaciales.

Para intentar dinamizar la manera en cómo se aprende geometría, muchos docentes hacen uso de la tecnología como soporte, sin embargo, aún existe deficiencia en como implementarla correctamente. En la investigación realizada por (Stols y Kriek 2011) se explica como la tecnología tiene el potencial de mejorar la instrucción, con respecto a la enseñanza de geometría y refuerza el aprendizaje de los alumnos mejorando la pedagogía. Los programas informáticos de matemáticas dinámicas, como GeoGebra, Cabri y Geometer's Sketchpad, se diseñaron

originalmente para la enseñanza de la geometría en la escuela secundaria. Estos programas permiten a los alumnos descubrir patrones, explorar y comprobar conjeturas construyendo sus propios bocetos. Por lo tanto, una manera de innovar los métodos tradicionales de aprendizaje es utilizar softwares educativos enfocados a geometría y medida. De aquí surge un concepto conocido como Geometría Dinámica, basada en programas para la creación y manipulación de construcciones geométricas como objetos, como puntos, líneas, círculos, etc., junto con las dependencias que pueden relacionar los objetos entre sí (Bantchev 2010).

No obstante, a pesar de la existencia de varios programas de geometría dinámica que son gratuitos, rara vez se utilizan para la enseñanza en los colegios, debido al desconocimiento por parte del docente de las posibilidades de su uso, y porque no se adaptan completamente a los temas del bloque de geometría y medida del libro de texto para 1ro de Bachillero.

En base a lo anteriormente mencionado, desarrollar un software educativo que se adapte a los contenidos de Geometría, que sugiere el ministerio de educación del Ecuador, utilizando los conceptos y herramientas dinámicas, que expanden las fuentes de información y ofrece un seguimiento del progreso y rendimiento académico, permitirá dinamizar el proceso de aprendizaje para que el conocimiento y habilidades adquiridas sean significativas.

2.1.1. Formulación del problema

¿Cómo implementar un software educativo que sea atractivo y fácil de utilizar para contribuir al proceso de aprendizaje de geometría y medida en los estudiantes de primero de Bachillerato?

2.1.2. Sistematización del problema

- ¿Cuál es la metodología o sistema actual de aprendizaje de Geometría y Medida de los estudiantes de primero de bachillerato?
- ¿Cuáles son las características que debe tener un software educativo para ser utilizado como herramienta cognitiva en el proceso de aprendizaje de geometría y medida?
- ¿Como se aplica la metodología ágil Scrum para el desarrollo de un software educativo?
- ¿Como se mide la usabilidad de un software educativo con el que se quiere facilitar el aprendizaje de geometría y medida?

2.2. Justificación

2.2.1. Justificación teórica

La tecnología sin duda alguna se ha convertido en una parte importante en la vida de los estudiantes más allá del colegio, e incluso lo podría ser dentro del aula. Al utilizar la tecnología de forma estratégica se convierte en una ayuda para aumentar la comprensión de conceptos complejos, así como dar a los estudiantes un acceso mucho más amplio a las matemáticas y otros recursos (Geiger, Faragher y Goos 2010).

En el área matemática se puede poner más énfasis en las aplicaciones prácticas mediante el modelado, la visualización y la manipulación de elementos. Si se aprovecha todo este potencial se puede mejorar la comprensión de los alumnos, estimular su interés y motivación, aumentando sus competencias en geometría y medida, y así mejorar el rendimiento de los estudiantes en los conceptos geométricos. Una de las maneras para conseguir esto es mediante el uso de materiales instructivos para promover el aprendizaje activo, al dinamizarlos se pueden mejorar aún más las habilidades de aprendizaje (National Council of Teachers of Mathematics 2015).

Los Software de Geometría Dinámica (DGS siglas en inglés) son considerados por muchos como unas de las herramientas tecnológicas más eficaces para fomentar la comprensión conceptual en el estudio del espacio y la forma, por lo que se necesita un alto nivel de capacidad de visualización, este problema se puede solucionar los DGS, ya que permiten la visualización de estas formas abstractas desde diferentes perspectivas con un estilo interactivo de exploración (Aydos 2015). Los sistemas de geometría dinámica ofrecen un nuevo enfoque de enseñanza y aprendizaje para visualizar e interactuar con conceptos matemáticos que son difíciles de imaginar o realizar (Karaibryamov, Tsareva y Zlatanov 2013).

Una de las mejores herramientas de geometría dinámica es GeoGebra, es un software gratuito y de código abierto que combina características de geometría dinámica y sistemas de álgebra computacional, y cuenta con una facilidad de uso, por lo que, es una gran ayuda para los profesores que quieren diseñar lecciones instructivas eficaces que les permitan crear objetos matemáticos y explorarlos de forma visual y dinámica (Hohenwarter, Hohenwarter y Lavicza 2009). Y en este punto se puede resaltar que los recursos creados (applets) se pueden guardar y compartir dentro de GeoGebra, también cuenta con la posibilidad utilizarlos en cualquier sitio web insertándolos de diferentes maneras. Aprovechando todos los beneficios de GeoGebra se va a crear contenido dinámico enfocado en cubrir los temas que se encuentran dentro del bloque de Geometría y medida para los estudiantes de primero de bachillerato. Sin embargo, se podría necesitar algún tipo de refuerzo en base a los conceptos y teoremas, así como poder evaluar el

conocimiento que van adquiriendo los estudiantes, es por esto por lo que el desarrollo de un software educativo que integre todos estos puntos será un complemento para innovar la manera en cómo se imparten las clases de Geometría actualmente.

A continuación, se detalla brevemente las tecnologías a utilizar, para la parte de gestión de contenido se utilizará “STRAPI”, un CMS que entrega una API desde un sistema de backend a través de la cual se gestiona y sirve todo el contenido multimedia que será almacenado a través de Amazon Web Services (AWS). Para la parte de la interfaz de usuario se utilizará “NEXTJS”, con el que se desarrollan aplicaciones estáticas que son renderizadas en el servidor con tiempos de respuestas rápidos y que pueden ser alojadas en cualquier sistema de hosting. Finalmente, la metodología ágil de desarrollo de software utilizada será Scrum.

2.2.2. Justificación aplicativa

Se va a desarrollar un software educativo al que se accede a través de la web, que permitirá crear las clases dinámicas para cada tema que se vaya a impartir de los contenidos de geometría y medida para los estudiantes de primero de bachillerato. La clase contará con una introducción teórica con inclusión de fuentes externas (enlaces a libros, informes, videos, entre otros recursos) que ayuden a los estudiantes a tener una base del tema a tratar, para proseguir con una demostración práctica e interactiva, utilizando geometría dinámica en donde se refleje de manera detallada el procesamiento para la resolución de ejercicios y para finalizar un cuestionario para comprobar el conocimiento obtenido.

En total se requieren de 14 clases, cada clase contiene 4 módulos: Introducción (base teórica), Video (multimedia), Práctica (geometría dinámica) y Evaluación (selección múltiple), además de contar con una clase introductoria que incluya una prueba de diagnóstico y finalmente una prueba final de todo el contenido presentado. De tal manera que la distribución será la siguiente:

- Introducción (Prueba de diagnóstico)
- Ecuación Vectorial
- Punto medio de un segmento
- Ecuación paramétrica de una recta
- Ecuación general de la recta
- Ecuación explícita de la recta
- Ecuación punto pendiente
- Posición relativa entre rectas
- Incidencia
- Rectas secantes
- Haces de rectas

- Ángulo entre las rectas
- Distancia entre 2 puntos
- Distancia de un punto a una recta (Cálculo directo)
- Distancia entre rectas paralelas
- Prueba Final

Para el seguimiento de los estudiantes se requieren los siguientes módulos:

- Listado de Estudiantes (Progreso)
- Calificaciones por tema
- Calificaciones finales (Promedio de todos los temas)
- Informes y analíticas

El desarrollo de este proyecto de integración curricular se rige por el eje y la línea de investigación de Tecnologías de Información y Comunicación de la ESPOCH en el programa de Tecnología Educativa, en cuanto al Plan Nacional de Desarrollo corresponde al eje Social, que tiene como uno de sus objetivos (7): “Potenciar las capacidades de la ciudadanía y promover una educación innovadora, inclusiva y de calidad en todos los niveles” y como política (7.2): “Promover la modernización y eficiencia del modelo educativo por medio de la innovación y el uso de herramientas tecnológicas.”

2.3. Objetivos

2.3.1. *Objetivo General*

Implementar un software educativo para el aprendizaje de Geometría y Medida para los estudiantes de primero de bachillerato.

2.3.2. *Objetivos Específicos*

- Describir los fundamentos teóricos, metodología actual y software educativo del proceso de aprendizaje de Geometría y Medida a nivel de educación media.
- Desarrollar el software educativo con módulos para el aprendizaje de los estudiantes y el seguimiento por parte de los maestros mediante el uso de la metodología SCRUM, el sistema de gestión de contenidos STRAPI y del framework NEXTJS.
- Evaluar el software educativo con base en estándares de Usabilidad para determinar con qué características debe contar si se busca facilitar el aprendizaje de Geometría y Medida en los estudiantes.

CAPITULO II

2. FUNDAMENTOS TEÓRICOS

En este capítulo se presentan recopilaciones de antecedentes, investigaciones previas y consideraciones teóricas en las que se sustenta este trabajo de titulación, considerando temas como el aprendizaje, la geometría en la educación, softwares educativos, metodologías y herramientas de desarrollo.

3.1. Aprendizaje y Geometría

3.1.1. *Conceptos de Aprendizaje*

Según (Zapata-Ros 2015), el aprendizaje es el medio mediante el cual se adquieren o modifican habilidades, ideas, valores, emociones, como resultado de la experiencia, razonamiento, estudio e instrucción. Por otra parte, (Garcés Cobos, Montaluisa Vivas y Salas Jaramillo 2019) mencionan que el aprendizaje humano va más allá de un simple cambio de conducta, conduce a un cambio en el significado de la experiencia, que no solo implica pensamiento, sino también afectividad y únicamente cuando se consideran en conjunto el individuo tiene la capacidad para enriquecer el significado de su experiencia.

3.1.2. *Teorías de Aprendizaje*

Desde el punto de vista de (Urbina Ramírez 1999) las teorías de aprendizaje dan una explicación organizada, racional y unificada de cómo se aprende, los límites que se presentan y por qué se olvida lo aprendido. Además, estas teorías aportan al conocimiento y con varios enfoques explican cómo se realiza el proceso de aprendizaje en los seres humanos desde un punto específico (Vega et al. 2019).

Para el desarrollo de un software educativo se requiere integrar estándares de diseño que permitan la comprensión del contenido, utilizando las bases del aprendizaje psicopedagógicas. Este es un proceso que intentar encontrar la calidad del software educativo, partiendo de las teorías de aprendizaje para tener un marco conceptual que guíe el desarrollo de los elementos didácticos que serán manejados en la aplicación requerida (Cataldi et al. 2007).

En base a todo esto, en la **Tabla 1-2** se tiene una revisión de las principales características de las teorías de aprendizaje educativas como el conductismo de Skinner (1958-1963), el cognitivismo de Gagné (1965), el constructivismo de Piaget (1989) y el Conectivismo de Siemens (2005). Analizando y utilizando los métodos y procesos en estas teorías se podrá realizar el diseño de un

marco conceptual que satisfaga las necesidades en el desarrollo de elementos didácticos para el aprendizaje de la geometría y medida.

Tabla 1-2 Características de las teorías de aprendizaje

Propiedad	Conductismo	Cognitivismo	Constructivismo	Conectivismo
¿Cómo se produce el aprendizaje?	Estímulo – respuesta Trata la conducta observable	A través de la asimilación, acomodación y adquisición	Relaciona lo intra y lo extra psicológico	Presenta una red de interconectividades, aprendizaje colaborativo
Factores que influyen	Naturaleza de recompensa, castigo o estímulo	Esquema existente, experiencias anteriores	Compromiso, participación, factores sociales y culturales	Diversidad de la red y la fuerza de los vínculos
Rol de memoria	Repetición, memorización	Codificación, almacenamiento y recuperación	Conocimiento propio	Agilidad, capacidad de selección y creación
¿Como ocurre la transferencia?	A través de un modelo mecanicista	Duplicación de las construcciones de conocimiento del “conocedor”	Socialización	Conectando o agregando redes
Otra manera de conocer la teoría	Aprendizaje basado en tareas	Razonamiento, objetivos claros y la resolución de problemas	Social, o por elaboración	Aprendizaje complejo, diversas fuentes de conocimiento

Fuente: Siemens, G., 2008

Realizado por: Silva, John., 2022

3.1.3. Tipos de Aprendizaje

En el proceso actual de enseñanza y aprendizaje cada día surgen nuevas tendencias que se apoyan en herramientas tecnológicas y medios informáticos, que son de gran utilidad en la difusión y proceso de información, y, por tanto, en los procesos de formación. Como explica (Universia 2020), conocer los tipos de aprendizaje existentes puede servir para orientar a los docentes en su forma de enseñar, de personalizar cada proceso, adaptado a las necesidades e inclinaciones del alumnado. Por otra parte, también es útil para los estudiantes que quieran conocer mejor su perfil y adoptar nuevas técnicas de estudio.

3.1.3.1. Aprendizaje Significativo

Este tipo de aprendizaje se basa en el constructivismo, intenta que la nueva información se incorpore fácilmente en la estructura cognitiva del estudiante, para que haya un proceso de incorporación de esta información con los conocimientos previos. Para (Ausubel 2015), el aprendizaje significativo se centra en edificar los conocimientos de forma armónica y coherente, por lo que es un aprendizaje que se construye a partir de conceptos sólidos, buscando así que, estos conocimientos perduren en el tiempo.

3.1.3.2. Aprendizaje Memorístico

Este tipo de aprendizaje promueve la adquisición de nuevos conocimientos reteniendo la información en la mente, buscando interiorizar el conocimiento, de tal manera que se logre explicarlo o repetirlo sin mayor esfuerzo. Esto se logra repitiendo varias veces lo que se quiere memorizar para así recordarla posteriormente. Como explica (Ausubel 2015), el aprendizaje mecánico es necesario en las primeras fases de un conjunto nuevo de conocimientos, en donde no se tienen conceptos importantes con los cuales pueda interactuar. Sin embargo, se debe priorizar el aprendizaje significativo para la adquisición de conocimiento, la retención y la transferencia de lo aprendido.

3.1.4. Fundamentos de Geometría

La palabra geometría significa “medida de la tierra”, es una rama de las matemáticas enfocada en resolver diversos problemas y comprender un mundo que ofrece amplias gamas de formas geométricas (Araya y Alfaro 2009). La geometría se desarrolla a través de la observación de los objetos del mundo que se originan desde términos indefinidos y se describen mediante definiciones. Algunos de estos objetos que estudia la geometría son puntos, rectas, planos, triángulos, polígonos, etc. (Godino y Ruiz 2002).

3.1.5. Geometría en la educación

La importancia de la geometría en la educación es señalada por (Almeida 2000), quien explica que existen algunos objetivos generales que todo estudiante debería alcanzar durante su formación: tener una cultura geométrica con visión histórica e interdisciplinaria, aplicar conocimientos geométricos para modelar, crear o resolver problemas reales, usar los diferentes lenguajes y representaciones. Como mencionan (Abrate, Pochulu y Delgado 2003) en su investigación, muchos docentes priorizan la enseñanza de las matemáticas en otras áreas y van desplazando los contenidos de geometría hacia el final del curso, esto conlleva en algunos casos a excluir temas o simplemente revisarlos de manera superficial. Además, si se analiza la metodología actual de enseñanza en los colegios, claramente los contenidos de geometría se presentan a los estudiantes como el producto acabado de la actividad matemática, que deja en segundo plano los procesos

implícitos de la construcción y de razonamiento en este conocimiento. Como consecuencia de continuar con un enfoque tradicional de enseñanza la concepción de la geometría resulta en una disciplina difícil y poco útil.

3.2. Software Educativo

Según (Noguera et al. 2011) se entiende por software educativo, a aquellos programas didácticos que pueden implementar o apoyar funciones educativas, ayudando o permitiendo el desarrollo de determinadas habilidades cognitivas. Un software educativo tiene la finalidad de ser utilizado como facilitador del proceso de aprendizaje, con algunas características particulares tales como: la facilidad de uso y la interactividad.

3.2.1. Clasificación de los softwares educativos

En base a lo que expone (Cataldi y Salgueiro 2007), los softwares educativos pueden ser clasificados como:

- *Tutoriales*, enfocados en dirigir el aprendizaje mediante una teoría subyacente conductista de la enseñanza, comparar los resultados de los alumnos contra patrones, generando nuevas ejercitaciones de refuerzo, si en la evaluación no se superaron los objetivos de aprendizaje.
- *Simuladores*, especializados en ejercitar los aprendizajes inductivo y deductivo mediante la toma de decisiones y adquisición de experiencia en situaciones imposibles de lograr desde la realidad, facilitando el aprendizaje por descubrimiento.
- *Entornos de programación*, tales como el Logo, permiten construir el conocimiento, paso a paso, facilitando la adquisición de nuevos conocimientos y el aprendizaje a partir de sus errores; y también conducen a los alumnos a la programación.
- *Herramientas de autor*, o lenguajes de autor, dirigidos a los profesores para que construyan programas del tipo tutoriales, especialmente a profesores que no disponen de grandes conocimientos de programación e informática, ya que usando muy pocas instrucciones, se pueden crear muy buenas aplicaciones de hipermedia.

3.2.2. Características del software educativo

Como expresa (Miranda Palma y Romero González 2019), un software educativo se utiliza como recurso para motivar los procesos de enseñanza y aprendizaje, cuenta con características específicas en relación con otros materiales didácticos, también utiliza en gran medida los recursos informáticos disponibles. Entre las principales características de un software educativo se encuentran la facilidad de uso, capacidad de motivación, relevancia curricular, versatilidad, enfoque pedagógico, orientación hacia los alumnos y evaluación. Por otra parte (Morejon 2011), presenta algunos principios elementales en la enseñanza que se deben considerar, entre estos: principios

de carácter educativo y científico, relación entre la teoría y la práctica, solidez en la asimilación de los conocimientos y habilidades.

3.2.3. Organización y presentación de contenidos en un software educativo

Para (Cataldi et al. 2007), seleccionar los contenidos a presentar es uno de los problemas recurrentes en la educación, comienza con el planteo de las siguientes preguntas por parte del docente ¿qué enseñar?, ¿para qué enseñar? y ¿cómo enseñar?, por lo cual, es necesario seleccionar la información a presentar y transmitir, determinando los contenidos y también su organización, la cual dependerá de la subdivisión del eje temático principal en bloques de contenido y en sub-bloques. La organización en bloques y sub-bloques se debe realizar de tal forma que permita una navegación en sentido horizontal, vertical y transversal, además debe estar de acuerdo con las diferentes estrategias de búsqueda que se preparen desde alguna de las visiones de los diferentes paradigmas educativos. Esta organización tiene que ser acorde y adecuado con el diseño de las pantallas en cada caso para una correcta presentación de los contenidos (Arias, López y Honmy J. 2015).

3.3. Metodología de Desarrollo de Software

Según (Maida y Pacienza 2015), una metodología para el desarrollo de software abarca los procesos que se deben seguir para concebir, implementar y mantener un producto de software, desde que se presenta la necesidad del producto hasta que cumplir el objetivo por el cual fue creado. Actualmente existe una gran cantidad de metodologías que se utilizan en base a las necesidades del proyecto, pero se debe considerar que todas las metodologías tienen ventajas que se pueden aprovechar, dependiendo de las condiciones del software que se pretende desarrollar, de igual forma presentan desventajas cuando no se consideran todos los factores que intervienen al realizar el trabajo.

Lo importante es utilizar siempre una metodología de software apropiada, para lo cual, es necesario conseguir información y documentación sobre ella, para lograr adaptarla lo mejor posible para obtener un producto de calidad que cumpla con los requerimientos funcionales y no funcionales (Ignacio, Paola y Fructuoso 2015).

3.3.1. Scrum

Scrum es una metodología ágil que permite trabajar de forma flexible y colaborativamente en proyectos que poseen gran variedad de requisitos, utilizando iteraciones (sprints) para su solución, de esta manera se realizan entregas parciales para buscar el beneficio del cliente del proyecto. Cabe resaltar que Scrum permite la autonomía de los equipos de trabajo, al utilizar reglas para crear un entorno ágil de administración de proyectos y no prescribe prácticas específicas de

ingeniería. En la **Figura 1-2** se muestra los pasos de esta metodología. (Srivastava, Bhardwaj y Saraswat 2017)

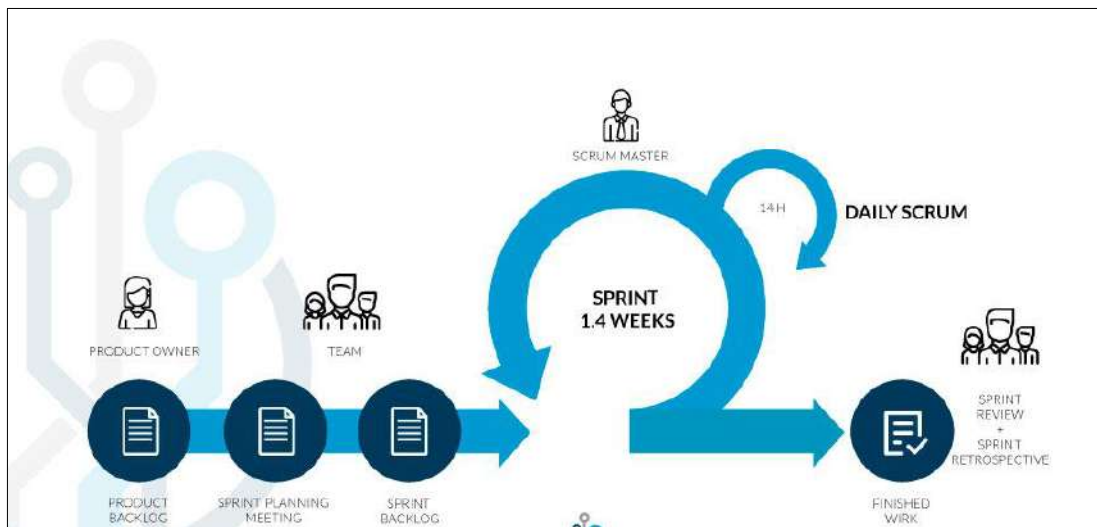


Figura 1-2: Pasos de Scrum

Fuente: Porras., 2020.

3.3.2. Roles de Scrum

Scrum se divide en tres roles principales: **Scrum Master**, es el responsable del proceso Scrum y de su correcta utilización maximizando sus beneficios; el **Product Owner**, está a cargo de la alineación del desarrollo y la definición de los objetivos de negocio, y el **Equipo**, que se ocupa de desarrollar y entregar el producto (Yin, Figueiredo y Silva 2011).

3.3.3. Artefactos de Scrum

Como expone (Fowler 2019), esta metodología identifica cuatro artefactos que son operados por el equipo Scrum a lo largo del ciclo de desarrollo:

- Product Backlog, es una lista con prioridad del trabajo necesario para concluir el producto.
- Sprint Backlog, es una lista de tareas que se realizan durante una iteración (sprint), la duración es variable desde dos hasta cuatro semanas.
- Incremento, es la colección de las tareas completadas en el Spring Backlog, cada incremento complementa los incrementos anteriores.
- Gráficos de desglose de versiones, muestran el progreso del proyecto a lo largo del tiempo.

3.3.4. Eventos de Scrum

De acuerdo con (Fowler 2019), la interacción de los roles que manejan los artefactos de Scrum se establece para las siguientes reuniones:

- Reunión de planificación de la versión, el equipo Scrum se reúne y rellena el Product Backlog.

- Reunión de Planificación del sprint, el equipo de desarrollo y el cliente discuten los avances y definen los objetivos para el siguiente sprint.
- Scrum diario, es una reunión rápida para que los desarrolladores identifiquen los problemas y las posibles mejoras en el uso de la metodología.
- Revisión del sprint, demostración del software en funcionamiento al cliente y a las partes interesadas.
- Retrospectiva del sprint, el equipo realiza una autoevaluación sobre el último sprint para buscar mejoras en el uso de Scrum y colaboración en general.

3.3.5. *Características de Scrum*

Como detallan (Schwaber y Sutherland 2020), algunas de las características de la metodología Scrum son las que a continuación se listan:

- Los procesos se dividen en iteraciones de trabajo breves (sprints).
- Los sprints se manejan mediante ensayo y error.
- Los entregables pueden ser alterados en la fase de planificación y sprint del proyecto.
- Está enfocada en la auto organización

3.3.6. *Scrum para el desarrollo de software educativos*

La metodología Scrum en el desarrollo de un software educativo se puede utilizar porque demuestra un marco que se adapta dinámicamente a las necesidades y requerimientos, además, se pueden realizar tareas simultáneamente, permite flexibilidad, mitigación de riesgos y adaptación a los cambios a los que se pueda someter en todo su ciclo de vida (Ticante Hernández et al. 2019).

(Ticante Hernández et al. 2019) presenta una adecuación de la metodología Scrum para el desarrollo de videojuegos, define 3 fases secuenciales que se listan a continuación:

- **Pre-game:** se realiza en una única iteración que comprende la definición del concepto del juego y la planificación.
- **Game:** comprende a la implementación del videojuego y se trabaja de forma iterativa e incremental.
- **Postgame:** tiene como objetivo evaluar y ajustar distintos aspectos del juego, eliminar errores detectados y, por último, entregar la versión final.

En la **Figura 2-2**, se muestra un diagrama de la metodología Scrum aplicada en el videojuego, las fases propuestas se adecuan completamente para el desarrollo de un software educativo.



Figura 2-2: Diagrama de la metodología Scrum para el desarrollo de videojuegos

Fuente: Hernández, T. et al, 2019

3.4. Arquitectura de Software

Desde el punto de vista de (Medvidovic y Taylor 2010), en el centro de cada sistema de software se encuentra su arquitectura de software, es decir, el conjunto de decisiones de diseño principales sobre el sistema. La arquitectura embebe todas las facetas principales de un sistema de software, ya que las principales decisiones de diseño pueden tomarse potencialmente en cualquier momento durante la vida útil de un sistema, y potencialmente por cualquier parte interesada. Estas decisiones incluyen aspectos estructurales, como los bloques de construcción de alto nivel, el despliegue del sistema, los requerimientos no funcionales del sistema y patrones de evolución del sistema.

3.4.1. Jamstack

Como señala (Tramullas 2020), Jamstack es una arquitectura moderna de desarrollo web que utiliza JavaScript del lado del cliente, API reutilizables y marcado prediseñado para crear sitios web y aplicaciones. A diferencia de las pilas tradicionales, la arquitectura de Jamstack es mucho más simple y se reduce a sólo tres componentes: JavaScript utilizado para añadir interactividad a las páginas y comunicarse con el backend a través de las APIs. Las APIs se utilizan en lugar de las bases de datos tradicionales y permite a las aplicaciones conectarse con servicios de terceros. El frontend está preconstruido en páginas estáticas optimizadas durante el proceso de construcción, las páginas estáticas se entregan con una red de distribución de contenidos (CDN) y se sirven como archivos HTML utilizando generadores de sitios estáticos (SSG). Como se observa en la **Figura 3-2**, hay una gran variedad de herramientas y servicios para utilizar esta arquitectura, la combinación de estas herramientas a elección del desarrollador hace que su experiencia de desarrollo sea satisfactoria y flexible.

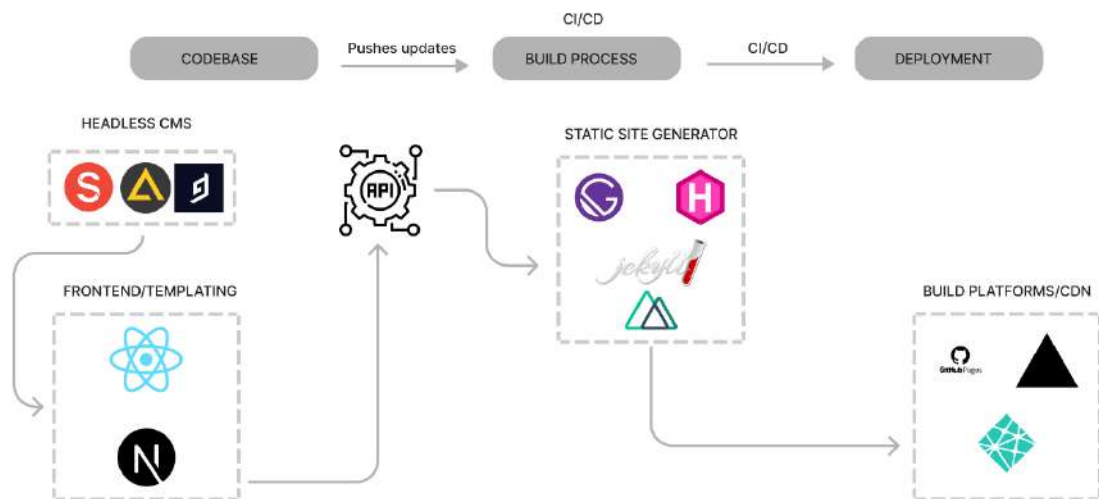


Figura 3-2: Arquitectura Jamstack

Fuente: Ikaheimo M., 2022

Como mencionan (Markovic et al. 2021), los sitios web creados utilizando Jamstack presentan una rapidez y personalización increíbles, cuando se utilizan las mejores prácticas y las herramientas correctas, pueden ser eficaces, seguros, escalables y mantenibles.

3.4.2. *RESTful API*

De acuerdo con (Halili y Ramadani 2018), REST en español denominado transferencia de estado representacional, es una arquitectura para aplicaciones basadas en redes, que indica principalmente como realizar el intercambio y manejo de datos a través de servicios web. REST se distingue por que se basa en el protocolo de aplicación HTTP. Es decir, usan los métodos y códigos de respuesta HTTP para una función específica y ampliamente reconocida por todos. Y nos permite a través de la URI, la estructuración de los recursos disponibles.

Una API RESTful se basa en REST, y resulta beneficiosa en las aplicaciones y servicios en la nube dado que la vinculación a un servicio a través de una API es una cuestión de controlar cómo se codifica la URL. Entre sus ventajas se encuentran la posibilidad de crear cliente/servidor en distintos lenguajes, así como la posibilidad de enviar la información en distintos formatos, aunque habitualmente se usa JSON, finalmente permite la creación de servicios más ligeros y simples, fácilmente visibles y utilizables a través de cualquier navegador web (Nguyen 2022).

3.5. Herramientas de Desarrollo de Software

En esta sección se presentan las diferentes herramientas que se utilizan en el desarrollo de software: lenguajes de programación, gestores de bases de datos, sistemas de gestión de contenido y framework para la web.

3.5.1. Aplicaciones basadas en la WEB

Las aplicaciones web se refieren a los programas de aplicación de software que se acceden fácilmente mediante un navegador web a través de una red y desarrolladas con lenguajes compatibles con el navegador (por ejemplo, HTML, JavaScript). Estas aplicaciones utilizan una combinación de scripts del lado del servidor para manejar los datos guardados y recuperar la información, y scripts del lado del cliente para presentar los datos a los usuarios (Al-Fedaghi 2011).

3.5.2. PostgreSQL

PostgreSQL es un gestor de bases de datos orientadas a objetos (SGBDOO o ORDBMS en sus siglas en inglés) muy conocido y usado en entornos de software libre, posee un conjunto de funcionalidades avanzadas, lo que lo sitúa al mismo o a un mejor nivel que muchos SGBD comerciales (Gibert, Ginestà y Mora 2016).

Algunas de las prestaciones más destacadas de PostgreSQL son:

- Gran conjunto de tipos de datos, permitiendo además su extensión mediante tipos y operadores definidos y programados por el usuario.
- Soporte para vistas, claves foráneas, integridad referencial, disparadores, procedimientos almacenados, subconsultas y casi todos los tipos y operadores soportados en SQL92 y SQL99.
- Su administración se basa en usuarios y privilegios.
- Es altamente confiable en cuanto a estabilidad se refiere y puede extenderse con librerías externas.

3.5.3. Sistema de Gestión de Contenido sin cabeza

Un Sistema de Gestión de Contenido sin cabeza (Headless CMS en inglés), simplemente contiene una API y un sistema de backend donde el contenido se almacena y se entrega. La expresión sin cabeza hace referencia a la idea de separar la cabeza (la interfaz) del cuerpo (backend). Los beneficios de un CMS son la flexibilidad, compatibilidad, seguridad y control (Abhishek Kothari 2022).

3.5.3.1. Strapi

Es un CMS sin cabeza de próxima generación, código abierto, basado en JavaScript, que permite crear, administrar y exponer experiencias agradables en contenido a cualquier dispositivo digital. Permite diseñar tablas de contenidos desde una interfaz fácil de usar, esta característica modela componentes específicamente adaptados a las necesidades que se tienen. Además, cada elemento es fácilmente actualizable por el administrador del sitio web con poco conocimiento técnico (Tanner 2020).

3.5.4. Frameworks – Frontend

Un framework se refiere a una estructura software compuesta de componentes personalizables e intercambiables para el desarrollo de una aplicación, se puede considerar como una aplicación genérica incompleta y configurable a la que podemos añadirle las últimas piezas para construir una aplicación concreta. Los objetivos principales que persigue un framework son acelerar el proceso de desarrollo, reutilizar código ya existente y promover buenas prácticas de desarrollo como el uso de patrones (Cuervo Gómez y Ballesteros-Ricaurte 2017).

3.5.4.1. NextJS

NextJS es un framework ligero de React utilizado para desarrollar aplicaciones estáticas y renderizadas en el servidor. NextJS viene con una configuración mínima y flexible que se puede ampliar para adaptarse al propósito de la aplicación. Además, utiliza un directorio de páginas donde se almacenan las páginas, cada una es un componente React que se asocia con una ruta basada en su nombre de archivo por lo que proporciona a la página un enrutamiento automáticamente. NextJS tiene diferentes tipos de renderización del lado del servidor. La generación estática obtiene los datos en el momento de la construcción, mientras que el renderizado del lado del servidor obtiene y renderiza los datos para cada solicitud (Johansson 2021).

3.5.5. Herramientas utilitarias

3.5.5.1. Visual Studio Code

Visual Studio Code es un editor de código que posee una gran cantidad de recursos para escribir y desarrollar código flexible y mantenible. Es una herramienta gratuita y de código abierto. Cuenta con características útiles como sistema de sugerencia o ayuda integrada, gestión de versiones con Git, instanciación de terminales, entre otros (Rubén Velasco 2021).

3.5.5.2. Postman

Postman es una aplicación que nos permite realizar pruebas API. Es un cliente HTTP que nos da la posibilidad de probar las diferentes solicitudes a través de una interfaz gráfica de usuario, por medio de la cual se obtienen diferentes tipos de respuesta que posteriormente deberán ser

validados. También se incluyen colecciones y la creación de ambientes a través de una interfaz gráfica de usuario amigable. La puesta en práctica de esta API de prueba ayuda a optimizar tiempo durante la ejecución de pruebas, especialmente cuando se trabaja con metodologías ágiles (Romero 2021).

3.5.5.3. GeoGebra

Es un software de geometría dinámica de acceso libre, con características importantes como la construcción de escenarios a partir de objetos y fórmulas, interactividad, facilidad de uso. Los escenarios construidos se pueden guardar y compartir a través de diferentes medios lo que permite contribuir en la mejora de actividades matemáticas para la resolución de problemas, elaboración de diferentes estrategias, exploración dinámica de las situaciones y aportando nuevos métodos de resolución. (Guachún Lucero y Espadero Faicán 2021)

3.6. Norma ISO/IEC 25010

La norma ISO/IEC 25010 es parte de la familia de normas ISO/IEC 25000 SQuaRE (Requisitos y evaluación de la calidad del software y del sistema), cuenta con un modelo de calidad compuesto por 8 características para la evaluación de la calidad del producto software, las propiedades y los requisitos de los usuarios (ISO2500).

3.6.1. Características de la ISO/IEC 25010

La norma ISO/IEC 25010 está compuesta por 8 características para determinar la calidad interna y externa de un producto software. La calidad del producto software es el grado que dicho software que satisface los requisitos de sus usuarios aportando de esta forma valor (Mera Paz, Miranda Gómez y Cuaran Rosas 2017). A continuación, se listan las características que se describen dentro la ISO/IEC 25010:

Tabla 2-2 Características de la ISO/IEC 25010

Característica	Subcaracterísticas
Adecuación funcional	<ul style="list-style-type: none"> • Completitud • Corrección • Pertinencia funcional.
Eficiencia de desempeño	<ul style="list-style-type: none"> • Comportamiento temporal • Utilización de los recursos • Capacidad
Compatibilidad	<ul style="list-style-type: none"> • Coexistencia • Interoperabilidad
Usabilidad	<ul style="list-style-type: none"> • Inteligibilidad • Aprendizaje • Operabilidad • Protección frente a errores de usuario • Estética • Accesibilidad
Fiabilidad	<ul style="list-style-type: none"> • Madurez • Disponibilidad • Tolerancia a fallos • Capacidad de recuperación
Seguridad	<ul style="list-style-type: none"> • Confidencialidad • Integridad • No repudio • Autenticidad • Responsabilidad
Mantenibilidad	<ul style="list-style-type: none"> • Modularidad • Reusabilidad • Analizabilidad • Capacidad de ser modificado • Capacidad de ser aprobado
Portabilidad	<ul style="list-style-type: none"> • Adaptabilidad • Facilidad de instalación • Capacidad de ser reemplazado

Fuente: ISO/IEC 25010, 2020

Realizado por: Silva, J., 2022

Para este trabajo se seleccionó la característica Usabilidad para ser evaluada en el software educativo.

3.6.2. Usabilidad

En base a lo que exponen (Balseca Chisaguano 2014), la usabilidad es la capacidad del software para ser entendido, aprendido, usado y resultar atractivo para el usuario, cuando se usa este bajo determinadas condiciones. Cuenta con algunas subcaracterísticas como la protección contra errores de usuario, estética de la interfaz de usuario, accesibilidad, inteligibilidad, capacidad de aprendizaje y operabilidad.

3.7. Trabajos Relacionados

Después de revisar trabajos similares en DSPACE de la ESPOCH, pueden resaltar los siguientes:

El diseño de una guía didáctica por parte de (Cachuput 2016), para el uso del software educativo específicamente para la geometría dirigido a los estudiantes del segundo año bachillerato de la unidad educativa “Pedro Vicente Maldonado” de la ciudad de Riobamba, entre resultados de este trabajo se destacan que el 72% de estudiantes encuestados opinan que usando las TICS se mejora el aprendizaje de la geometría y su rendimiento académico.

(Inés y Brito 2021) en el trabajo denominado “Utilización de un software como metodología alternativa para el aprendizaje del Bloque Curricular Geometría y Medida en el noveno año de educación básica de la Unidad Educativa “Isabel de Godín” durante el año lectivo 2013-2014”, presenta el software “Geomedida”, que se conforma por reflexiones, una revista, un tutorial, actividades recreativas y una evaluación del bloque curricular geometría y medida. Este software contribuye al fortalecimiento el proceso de enseñanza aprendizaje.

(Zambrano David 2015) en el trabajo de integración para la Escuela Superior Politécnica de Chimborazo desarrolló un software educativo como apoyo para la enseñanza de la asignatura de matemática I en la Escuela de Ingeniería Ambiental de la Universidad Estatal Amazónica. Para validar la importancia del software educativo en el proceso aprendizaje el autor aplicó un Pre-Test y Post-Test a los dos grupos referente al nivel de conocimientos antes y después de la ejecución del software y como resultados se obtiene que el grupo experimental obtuvo un promedio de 14.1 puntos y el 72.4% de aprobados frente a 10.1 puntos para el grupo control con solo un 51.4% de aprobado. Esto demuestra que la utilidad de un software educativo para el aprendizaje de matemáticas.

(Cocunubo-Suárez, Parra-Valencia y Otálora-Luna 2018), en el artículo de revisión “Propuesta para la evaluación de Entornos Virtuales de Enseñanza Aprendizaje con base en estándares de Usabilidad”, concluye que, para evaluar Entornos Virtuales de Aprendizaje es fundamental tener

en cuenta 5 aspectos pedagógicos, que son: contenido, modelo, evaluación, interfaz de usuario; aunque estas no tienen relación con ninguna característica del estándar ISO/IEC 25000. Cabe mencionar, que la propuesta del autor se centra en reemplazar la característica facilidad de aprendizaje por facilidad pedagógica, además que se integran 4 aspectos: soporte-ayuda-documentación, manejo de errores, flexibilidad y estándares, a diferentes características del estándar ISO/IEC 25000.

CAPITULO III

3. MARCO METODOLÓGICO

En este capítulo se describe el tipo de estudio, los métodos, técnicas, instrumentos y las fases de la metodología Scrum para el desarrollo del software educativo.

4.1. Tipo de estudio

El desarrollo de este proyecto es de tipo aplicativo, ya que se utiliza el conocimiento adquirido durante la carrera para desarrollar el software educativo que apoya el aprendizaje de Geometría y Medida en estudiantes de primero de bachillerato de la Unidad Educativa Vigotsky, enfocando y expandiendo los temas propuestos en el libro de matemáticas por el Ministerio de Educación.

4.1.1. *Métodos, técnicas, fuentes e instrumentos de estudio*

En la Tabla 1-3 se listan los métodos, las técnicas o instrumentos y las fuentes a utilizar en base a los objetivos específicos planteados.

Tabla 1-3 Métodos, Técnicas y Fuentes para el trabajo de Integración Curricular

Objetivos	Métodos	Técnicas	Fuentes
Describir los fundamentos teóricos, metodología actual y software educativo del proceso de aprendizaje de Geometría y Medida a nivel de educación media	Analítico Sintético	<ul style="list-style-type: none"> Revisión de documentación 	<ul style="list-style-type: none"> Artículos científicos Tesis Revistas Internet Bases de datos documentales
Desarrollar el software educativo con módulos para el aprendizaje de los estudiantes y el seguimiento por parte de los maestros mediante el uso de la metodología SCRUM, el sistema de gestión de contenidos STRAPI y del framework NEXTJS	Scrum	<ul style="list-style-type: none"> Entrevistas Historias de usuario Sprint Backlog Product Backlog Puntos de historia Diagramas UML Pruebas 	<ul style="list-style-type: none"> Equipo Scrum Artículos científicos Tesis Revistas Internet Bases de datos documentales
Evaluar el software educativo con base en estándares de Usabilidad para determinar con qué características debe contar si se busca facilitar el aprendizaje de Geometría y Medida en los estudiantes	Inductivo Estadístico	<ul style="list-style-type: none"> Cuestionario Revisión de documentación 	<ul style="list-style-type: none"> Estándares de usabilidad Docente Estudiantes

Realizado por: Silva, J., 2022

A continuación, se describe como se utilizará cada uno de los métodos y técnicas seleccionadas para el cumplimiento de los objetivos:

4.1.2. Método Analítico

A través de este método, se enfocará el estudio en el problema por resolver, analizando las causas y consecuencias, las ventajas de utilizar un software educativo para el aprendizaje y cuál será la solución más factible que implementar.

4.1.3. Método Sintético

Mediante este método, se resalta los aspectos más importantes del proceso de enseñanza, aprendizaje y funcionamiento de un software educativo, partiendo de la descomposición y distinción los elementos de un todo y así revisar ordenadamente cada uno de estos por separado.

4.1.4. Metodología Scrum

Se requiere tener la premisa de que los problemas no pueden ser totalmente definidos o entendidos de antemano, por tanto, con se utiliza Scrum para tener la capacidad de responder a los cambios, así como tener una participación de los usuarios en todo el proceso.

4.1.5. Método Inductivo

Aplicando este método se evaluará la usabilidad del software educativo a través de técnicas como la observación y utilización de encuestas para obtener conclusiones de esta evaluación.

4.1.6. Método Estadístico

Con este método se realizará todos los procesos para el manejo de los datos cualitativos de la investigación, con el fin de comprobar los resultados de la implementación del software educativo.

4.2. Determinación de la Usabilidad

En esta sección se describe la operación conceptual y metodológica de la usabilidad, especificando cuáles son las características por evaluar, además se determina la población y la muestra, la herramienta para la obtención de datos y finalmente el proceso de evaluación utilizado.

4.2.1. Operación Conceptual

Para el análisis conceptual de la variable a evaluar se parte desde la formulación del problema, en este caso corresponde a la usabilidad. Además, en la Tabla 2-3 se define el tipo de la variable y su concepto.

Tabla 2-3 Operacionalización conceptual de variables

Formulación del Problema	Variable	Tipo	Concepto
¿Cómo implementar un software educativo que sea atractivo y fácil de utilizar para contribuir al proceso de aprendizaje de geometría y medida en los estudiantes de primero de Bachillerato?	Usabilidad	<ul style="list-style-type: none"> • Cualitativa • Compleja 	Es la propiedad que tiene un determinado sistema para que sea ‘fácil de usar y de aprender’; tratándose de una propiedad que no es sólo aplicable a los sistemas software

Realizado por: Silva, J., 2022

4.2.2. Operación Metodológica

Después de la operación conceptual de la usabilidad se seleccionan las subcaracterísticas que serán evaluadas, también se presenta el indicador, técnicas con las que se recolectan los datos y las fuentes de cada una como se observa en la Tabla 3-3.

Tabla 3-3 Operacionalización metodológica de variables

Formulación del Problema	Variable	Subcaracterísticas	Indicador	Técnica	Fuente
¿Cómo implementar un software educativo que sea atractivo y fácil de utilizar para contribuir al proceso de aprendizaje de geometría y medida en los estudiantes de primero de Bachillerato?	Usabilidad	Facilidad Pedagógica	Materiales de aprendizaje Evaluación del aprendizaje Retroalimentación	Observación Cuestionario	Software educativo Estudiantes Maestro
		Facilidad de Entendimiento o Comprensibilidad	Interfaz de usuario perfiles (estudiante/maestro) Estructura y navegación	Observación Cuestionario	Software educativo Estudiantes Maestro
		Facilidad de Uso	Manejo de errores Notificación de estados	Observación Cuestionario	Software educativo Estudiantes Maestro

Realizado por: Silva, J., 2022

4.2.2.1. Características de usabilidad a evaluar

En la Tabla 4-3 se detallan las características de la usabilidad propuestas por (Cocunubo-Suárez et al. 2018) que serán evaluadas en el software educativo.

Tabla 4-3 Características de usabilidad a evaluar

Características	Definición
Facilidad pedagógica	Hace referencia a las funcionalidades que el entorno provee a docentes y alumnos para el desarrollo del proceso de enseñanza y aprendizaje.
Facilidad de entendimiento o comprensibilidad	Hace referencia a todas aquellas características del software educativo que facilitan su entendimiento; como puede ser utilizada para la realización de tareas en condiciones de uso particulares.
Facilidad de uso	Hace referencia a todos aquellos atributos del software educativo que permiten controlarla y operarla adecuadamente. Su definición se basa en los términos controlabilidad, tolerancia a fallos y conformidad con las expectativas del usuario.

Realizado por: Silva, J., 2022

4.2.3. Población y Muestra

Para la evaluación del software educativo en la Unidad Educativa Vigotsky se considera el paralelo A de primer año de bachillerato que con un tamaño de la población de 70 estudiantes. Para la obtención de la muestra se aplica la siguiente fórmula:

$$n = \frac{N * Z^2 * p * q}{e^2 * (N - 1) + Z^2 * p * q}$$

Donde:

n = tamaño de la muestra

N = Tamaño del universo

Z = Nivel de confianza

p = probabilidad de éxito

q = probabilidad de fracaso

e = Error de estimación máximo aceptado

Para la evaluación del software educativo se tienen los siguientes valores presentados a continuación, obteniendo una muestra de 40 estudiantes.

N = 70 estudiantes

Z = 1,96 que corresponde al nivel de confianza de 95%

e = 10% de error admitido

p = 0,5 de probabilidad de éxito

q = 0,5 de probabilidad de fracaso

$$n = \frac{70 * 1.96^2 * 0.5 * 0.5}{0.1^2 * (70 - 1) + 1.96^2 * 0.5 * 0.5} = 40 \text{ estudiantes}$$

4.2.4. Herramienta para la evaluación de la usabilidad

Para la evaluación de la usabilidad del software educativo se utilizan como base cuestionarios estandarizados que se emplean para medir la satisfacción de los usuarios con respecto a un producto de software. El primero es el PSSUQ (Post-Study System Usability Questionnaire) de IBM que consta de 16 preguntas con 7 opciones que van desde “totalmente de acuerdo” hasta totalmente en desacuerdo y se incluye una opción NA en caso de que no corresponda o se aplique. El siguiente es el cuestionario USE (Usefulness, Satisfaction, and Ease of use) propuesto por Arnold Lund que cuenta con 30 preguntas con aspectos específicos de la usabilidad de un software, al igual que PSSUQ tiene 7 opciones que van desde “muy en desacuerdo” hasta “muy de acuerdo” con una opción en caso de no aplicar.

En la Tabla 5-3 se detallan las 15 preguntas adaptadas para el cuestionario final de cada característica. Para evaluar la capacidad de entendimiento o comprensibilidad hay 6 preguntas qué tienen que ver con la estética de la interfaz, la estructura de la información y la navegabilidad. Para la facilidad de uso u operabilidad hay 7 preguntas referentes a la tolerancia a fallos, el manejo de errores y las notificaciones de estados.

Tabla 5-3 Cuestionario final para la evaluación del software educativo

Características	Preguntas
Facilidad Pedagógica	<ul style="list-style-type: none"> - La información de la introducción y el video de cada clase es clara y fácil de entender - La evaluación de cada clase tiene preguntas acordes a la temática y las puedo responder sin mayor problema. - Al finalizar la evaluación puede revisar mis respuestas correctas o erróneas, y se explica el procedimiento o la manera de resolver cada pregunta.
Facilidad de entendimiento o comprensibilidad	<ul style="list-style-type: none"> - La pantalla de las clases es agradable. - Te gustó interactuar con las figuras interactivas. - Tuve complicaciones para entrar en la pantalla de calificaciones. - La pantalla de inicio es agradable y puede seleccionar la clase que quiera de manera fácil. - Al ir a las diferentes pantallas pude regresar a donde estaba antes. - Recomendaría este software educativo a un amigo.
Facilidad de uso u operabilidad	<ul style="list-style-type: none"> - No tuve ningún error mientras utilizaba el software educativo. - Si tuve un error pude volver atrás y seguir navegando. - Se mostró un mensaje después de completar una clase o actualizar mi perfil. - Al poner un correo electrónico equivocado para iniciar sesión se mostró un mensaje. - Es cómodo navegar desde la “Introducción” de la clase hasta la “Prueba”. - Es fácil interactuar con todo el software educativo.

Realizado por: Silva, J., 2022

4.2.5. *Proceso de evaluación de la Usabilidad*

Escala de evaluación

Para cada pregunta del cuestionario se utiliza una escala cualitativa, la cual es Likert de 5 puntos que se describe en la Tabla 6-3 a continuación.

Tabla 6-3 Escala de Likert para la evaluación

Puntuación	1	2	3	4	5
Escala cualitativa	Totalmente en desacuerdo	En desacuerdo	Ni de acuerdo ni en desacuerdo	De acuerdo	Totalmente de acuerdo

Realizado por: Silva, J., 2022

4.3. Desarrollo del software educativo utilizando la metodología Scrum

4.3.1. Análisis previo al desarrollo del software educativo

En esta etapa se realiza un estudio previo al desarrollo del software educativo, para analizar la factibilidad y verificar posibles riesgos que puedan ocurrir durante el desarrollo evitando inconvenientes posteriores.

4.3.1.1. Estudio de factibilidad

Mediante este estudio se comprueba la viabilidad de la realización del proyecto, para lo cual se toma en cuenta recursos que se tienen disponibles, así como los necesarios para cumplir con todas las actividades planteadas.

Factibilidad técnica

En la factibilidad técnica se analizan los recursos técnicos necesarios para la realización del proyecto, en la Tabla 7-3 se especifican los recursos hardware y en la Tabla 8-3 los recursos software.

Tabla 7-3 Recursos hardware

Cantidad	Recurso	Características
1	Computadora	AMD Ryzen 9
1	Servidor	Heroku
1	S3 AWS	Bucket S3 de AWS

Realizado por: Silva, J., 2022

En cuanto a los recursos hardware se requiere de una computadora para el desarrollo, de un servidor para el backend y un bucket S3 de AWS para el almacenamiento de archivos multimedia.

Tabla 8-3 Recursos software

Recurso	Descripción	Características
Visual Studio Code	Editor de código	Libre
Strapi	Sistema de gestión de contenido sin cabeza	Gratis
Insomnia	Pruebas para API	Gratis
PostgreSQL	Sistema de base de datos	Libre
GeoGebra	Applets dinámicos	Libre
Vercel	Despliegue del cliente	Gratis

Realizado por: Silva, J., 2022

Los recursos software que se necesitan en mayor medida son libres o gratuitos por lo que es factible el desarrollo.

Factibilidad económica

En la factibilidad económica se establecen y analizan todos los gastos que se incluyen para la elaboración del proyecto, para obtener un costo total aproximado y determinar si es factible la realización de este. En la Tabla 9-3 se detallan los costos de los recursos para el desarrollo.

Tabla 9-3 Costos de los recursos para el desarrollo

Tipo del recurso	Recurso	Cantidad	Fuente de financiamiento	Precio unitario \$	Precio total \$
Hardware	Computadora	1	Personal	1700	1700
	Servidor	1		100 / año	100 / año
	Bucket S3	1		200 / año	200 / año
Software	Visual Studio Code	1	Personal	0	0
	Strapi	1		0	0
	Insomnia	1		0	0
	PostgreSQL (Plugin Heroku)	1		60 / año	60 / año
	GeoGebra	1		0	0
	Vercel	1		0	0
				Total	2.060

Realizado por: Silva, J., 2022

Después de analizar la Tabla 9-3 se tiene que el costo total del proyecto es de 2060 dólares, puede ser financiado por lo que el desarrollo es factible.

Estimaciones

Para realizar las estimaciones se utiliza el modelo COCOMO II para determinar el tiempo en meses y el número de programadores requeridos para el desarrollo de este proyecto. Se inicia con el análisis para obtener los puntos de fusión, y posterior el número de lianas de código. De tal manera se ha obtenido un total de 160 puntos de fusión y un valor de KLOCs de 4.60.

El tiempo de desarrollo estimado es de 6.4 meses, un esfuerzo constante de 11.99 hombre-mes, y un total de 1.82. Por lo tanto, es necesario realizar un ajuste para que la realización de ese proyecto sea factible. Las estimaciones realizadas se encuentran en el ANEXO A.

4.3.1.2. Gestión de riesgos

En esta sección se identificarán y estudiarán las diferentes fuentes de riesgo con el propósito de controlarlos y eliminarlos antes de que se conviertan en amenazas que interfieran en la correcta finalización de este proyecto.

Identificación de riesgos

Como se observa en la Tabla 10-3 se plantean los posibles riesgos identificados que se pueden presentar durante el desarrollo del software educativo. En total se identificaron 6 posibles riesgos, 4 corresponden a riesgos del proyecto y 2 a riesgos técnicos, Además se detallan las diferentes consecuencias que podrían presentar.

Tabla 10-3 Riesgos identificados

ID	Descripción	Tipo de Riesgo	Consecuencia
RI-01	Cambios no viables de requerimientos	Proyecto	Cambios graves en el software educativo, incompatibilidad de funcionalidades, alto gasto de recursos
RI-02	Falta de partición del cliente	Proyecto	Retraso en la planificación, pérdida de calidad del producto
RI-03	Asignación incorrecta de recursos	Proyecto	Finalización tardía del desarrollo, abandono del proyecto
RI-04	Daño de equipos	Proyecto	Aumento del costo del proyecto
RI-05	Herramientas de desarrollo incompatibles	Técnico	Retraso y reajuste en la planificación
RI-06	Diseño erróneo de base de datos	Técnico	Fallas en el manejo de datos, problemas de persistencia e integridad.

En el Anexo B, se especifican los criterios para la valoración de probabilidades e impactos de los riesgos, su análisis y la priorización.

4.3.2. Fase de planificación

Como se observa en la Tabla 11-3 se detallan las tareas que se realizan en el desarrollo del software educativo, desde las reuniones con los usuarios para determinar el alcance y los requerimientos hasta las pruebas y despliegue del producto. En cada tarea se detalla la fecha de inicio y finalización propuestas incluyendo el tiempo en horas requerido.

En el Anexo C se presenta el diagrama de Gantt con todas las tareas para este proyecto.

Tabla 11-3 Plan de Trabajo

	Nombre de tarea	Inicio	Fin	Horas
Marco metodológico		5/3/2022	6/30/2022	408
	Determinación de los tipos de investigación	5/3/2022	5/4/2022	16
	Determinación de métodos y técnicas	5/3/2022	5/4/2022	16
Desarrollo del software educativo		5/6/2022	6/30/2022	384
Análisis		5/6/2022	5/11/2022	40
	Entrevistas	5/6/2022	5/7/2022	16
	Recolección y análisis de información	5/7/2022	5/9/2022	16
	Selección de requerimientos	5/10/2022	5/11/2022	16
	Análisis y definición de las herramientas de desarrollo	5/11/2022	5/11/2022	8
Planificación		5/12/2022	5/16/2022	32
	Roles y personas	5/12/2022	5/12/2022	8
	Tipos de usuarios	5/12/2022	5/12/2022	8
	Historias de usuario Historias técnicas	5/13/2022	5/14/2022	16
	Product Backlog	5/14/2022	5/16/2022	16
	Spring Backlog	5/14/2022	5/16/2022	16
Desarrollo y Pruebas		5/16/2022	6/16/2022	224
	Arquitectura del Sistema	5/16/2022	5/17/2022	16
	Diagramas UML	5/17/2022	5/18/2022	16
	Estándares de codificación	5/16/2022	5/18/2022	24
	Diseño de base de datos	5/19/2022	5/19/2022	8
	Diseño de interfaces	5/19/2022	5/19/2022	8
	Codificación del software educativo	5/20/2022	6/6/2022	120
	Pruebas unitarias	5/20/2022	6/6/2022	120
	Pruebas de integración	5/20/2022	6/6/2022	120
	Cuestionario para evaluación de la usabilidad	6/7/2022	6/8/2022	16
	Pruebas de aceptación	6/10/2022	6/16/2022	48

Cierre	6/16/2022	6/30/2022	104
Correcciones Finales	6/16/2022	6/25/2022	72
Despliegue del software educativo	6/27/2022	6/29/2022	24
Conclusión de documentación	6/16/2022	6/30/2022	104

Realizado por: Silva, J., 2022

4.3.2.1. Personas y Roles en el proyecto

En la Tabla 12-3 se listan las personas involucradas en el proyecto con el respectivo rol que desempeña.

Tabla 12-3 Personas y roles

Persona	Rol	Contacto
Ing, Diego Avila	Scrum Master	diego.avila@epoch.edu.ec
Ing. Miguel Duque	Scrum Master	miguel.duque@epoch.edu.ec
Ing. Miriam Avila	Product Owner	miriam.avila@epoch.edu.ec
John Silva	Desarrollador	john.silva@epoch.edu.ec

Realizado por: Silva, J., 2022

4.3.2.2. Tipos de usuarios del software educativo

En la Tabla 13-3 se describen los tipos de usuarios del software educativo. En total hay 4, administrador y editor, los cuales manejan el sistema de gestión de contenidos Strapi. También maestro y estudiante que interactúan con el software educativo.

Tabla 13-3 Tipos de usuarios

Tipo	Actividades	Módulos
Administrador	Gestión de usuarios, contenidos y permisos	Gestor de Contenidos
Editor	Gestión de contenidos	Gestor de Contenidos
Maestro	Gestión de contenidos, estudiantes, calificaciones y reportes	Gestor de Contendidos, Seguimiento estudiantil
Estudiante	Visualización de contenidos, pruebas y avance	Estudiantes

Realizado por: Silva, J., 2022

4.3.2.3. Estimación

Para realizar la estimación de las tareas se utilizará el método de la talla de camiseta o T-Shirt.

En donde las estimaciones se realizan con métodos XS, S, M, L y XL, como se presenta en la Tabla 14-3, a continuación:

Tabla 14-3 Puntos Estimados T-shirt

Talla	Puntos Estimados	Horas de Trabajo
XL	40	40
L	20	20
M	15	15
S	10	10
XS	5	5

Realizado por: Silva, J., 2022

4.3.2.4. Product Backlog

Con el product backlog se representan de manera clara todos los requerimientos denominados como historias de usuario y técnicas para el software educativo. El product backlog consta del ID de la historia, el prefijo HT corresponde a la historia técnica, y el prefijo HU corresponde a una historia de usuario acompañado con su respectivo número, además se tiene el nombre de la historia, su prioridad, y su estimación, tal y como se especifica en la Tabla 15-3.

Tabla 15-3 Product Backlog

Product Backlog			
ID	Nombre de la historia	Esfuerzo	Prioridad
HU-01	Registro de usuarios	40	Alta
HU-02	Autenticación de usuarios	16	Alta
HU-03	Restablecer contraseña mediante correo electrónico	16	Media
HU-04	Crear, modifica eliminar y visualizar las clases	8	Alta
HU-05	Crear, modifica eliminar y visualizar las categorías	8	Alta
HU-06	Crear, modifica eliminar y visualizar las pruebas	8	Alta
HU-07	Crear, modifica eliminar y visualizar las preguntas	8	Alta
HU-08	Enlazar temas a una categoría	12	Alta
HU-09	Enlazar preguntas a una prueba	12	Alta
HU-10	Enlazar prueba a un tema	8	Alta
HU-11	Modificar información personal del estudiante	12	Media
HU-12	Visualizar listado de temas	24	Alta
HU-13	Visualizar una introducción a la clase	16	Alta
HU-14	Visualizar contenido multimedia de la clase	16	Alta
HU-15	Visualizar e interactuar con un modelo geométrico sobre la clase	82	Alta

HU-16	Realizar una evaluación sobre la clase	56	Alta
HU-17	Obtener una puntuación mínima para finalizar la clase	4	Baja
HU-18	Retroalimentación de cada evaluación	16	Alta
HU-19	Visualizar los temas completos e incompletos	8	Media
HU-20	Visualizar un resumen sobre las calificaciones de las clases	32	Alta
HU-21	Visualizar los resultados finales y el progreso	32	Alta
HU-22	Obtener un certificado al finalizar todas las clases	32	Alta
HU-23	Visualizar listado de estudiantes	32	Alta
HU-24	Visualizar calificaciones por clase	32	Alta
HU-25	Visualizar calificaciones generales	32	Alta
HU-26	Gráficas de calificaciones	32	Alta

Realizado por: Silva, J., 2022

4.3.2.5. Historias técnicas

Las historias técnicas o elementos no funcionales en la metodología Scrum hacen referencia a las tareas que deben realizarse, pero no son un entregable ni se encuentran directamente relacionadas con alguna historia específica por lo tanto no son de valor inmediato para el dueño del producto. Se pueden identificar diferentes tipos de historias técnicas por ejemplo para la arquitectura infraestructura del producto o equipo refactorización entre otros. En la Tabla 16-3 se muestra un ejemplo de la historia técnica y la estructura utilizada en este proyecto. Las historias técnicas se encuentran con más detalle en el anexo D.

Tabla 16-3 Historia técnica

HISTORIA TÉCNICA			
ID	HT-01	Nombre de la Historia	Definir los requerimientos del software educativo
Rol	Desarrollador	Sprint	
Prioridad	Alta	Puntos Estimados	48
Riesgo		Puntos Reales	48
Fecha (Inicio)	06/05/2022	Fecha (Fin)	11/05/2022
Descripción	Como desarrollador requiero entrevistas con el cliente para recolectar información, analizarla y definir los requerimientos del software educativo		
PRUEBAS DE ACEPTACIÓN			
ID_PA	Descripción		Estado
HT01 – PA01	Analizar el nivel de cumplimiento de los objetivos con los requerimientos definidos		Finalizada
TAREAS DE INGENIERÍA			
ID_TI	Descripción		Esfuerzo
HT01 – TI01	Realizar las reuniones con el cliente hasta que se cumpla con el nivel de cumplimiento de los objetivos con los requerimientos definidos		24
HT01 – TI02	Documentar los requerimientos definidos en base a la metodología Scrum		24

Realizado por: Silva, J., 2022

4.3.2.6. Historias de usuario

Las historias de usuario se utilizan para representar los requerimientos del software educativo, cuentan con la siguiente estructura para escribirlas: Como “rol”, quiero “evento”, para “funcionalidad”. Algunos de los campos que se requieren son: ID de la historia de usuario, nombre de la historia, el usuario, la prioridad, el sprint en que se desarrollará, además de parámetros para su aceptación y las tareas de ingeniería que le corresponden. En la Tabla 17-3 se muestra un ejemplo de las historias de usuario y su estructura utilizada en este proyecto.

Tabla 17-3 Historia de usuario

HISTORIA DE USUARIO			
ID	HU-01	Nombre de la Historia	Registro de usuarios
Usuario	Administrador, maestro, estudiante		Sprint
Prioridad	Alta	Puntos Estimados	40
Riesgo		Puntos Reales	40
Fecha (Inicio)	20/05/2022	Fecha (Fin)	22/05/2022
Descripción	Como usuario quiero registrarme en el software educativo para poder acceder a los contenidos		
PRUEBAS DE ACEPTACIÓN			
ID_PA	Descripción		Estado
HU01 – PA01	Validación de campos requeridos para el registro		Finalizada
HU01 – PA02	Verificar que el usuario sea único		Finalizada
HU01 – PA03	Notificación y verificación a través de correo electrónico		Finalizada
HU01 – PA04	Verificar almacenamiento de usuarios en Strapi		Finalizada
TAREAS DE INGENIERÍA			
ID_TI	Descripción		Esfuerzo
HU01 – TI01	Desarrollar la interfaz de registro e inicio de sesión		8
HU01 – TI02	Implementar el servicio web para la comunicación del software educativo con Strapi		32

Realizado por: Silva, J., 2022

En total se tienen 26 historias de usuario las que se pueden encontrar en el Anexo E.

4.3.2.7. Spring Backlog

En el Sprint Backlog se detalla el número de iteraciones (sprint) para este proyecto. Dentro de cada sprint se presentan las historias técnicas o las historias de usuario que le corresponden, con la fecha de inicio, la fecha final y los puntos de estimados de cada tarea. Además, en cada sprint se señalan los puntos totales de esfuerzo y el responsable. A continuación, en la Tabla 18-3 se visualiza el sprint backlog con todos los detalles.

Tabla 18-3 Sprint Backlog

Sprint	ID Historia	Fecha Inicio	Fecha Fin	Puntos estimados	Puntos totales	Responsable
1	HT-01	06/05/2022	11/05/2022	48	152	John Silva
	HT-02	16/05/2022	17/05/2022	32		
	HT-03	17/05/2022	18/05/2022	8		
	HT-04	19/05/2022	19/05/2022	32		
	HT-05	19/05/2022	20/05/2022	32		
2	HU-01	20/05/2022	22/05/2022	40	128	John Silva
	HU-02	23/05/2022	23/05/2022	16		
	HU-03	24/05/2022	24/05/2022	16		
	HU-04	24/05/2022	24/05/2022	8		
	HU-05	25/05/2022	25/05/2022	8		
	HU-06	25/05/2022	25/05/2022	8		
	HU-07	25/05/2022	25/05/2022	8		
	HU-08	25/05/2022	25/05/2022	8		
	HU-09	25/05/2022	25/05/2022	8		
	HU-10	25/05/2022	25/05/2022	8		
3	HU-11	26/05/2022	26/05/2022	12	156	John Silva
	HU-12	27/05/2022	27/05/2022	24		
	HU-13	27/05/2022	28/05/2022	16		
	HU-14	29/05/2022	29/05/2022	16		
	HU-15	30/05/2022	03/06/2022	82		
	HU-16	03/06/2022	04/06/2022	56		
	HU-17	04/06/2022	04/06/2022	4		
	HU-18	04/06/2022	05/06/2022	16		
	HU-19	05/06/2022	05/06/2022	8		
	HU-20	05/06/2022	05/06/2022	32		
4	HU-21	06/06/2022	06/06/2022	32	92	John Silva
	HU-22	06/06/2022	06/06/2022	16		
	HU-23	06/06/2022	06/06/2022	12		
	HU-24	06/06/2022	06/06/2022	8		
	HU-25	07/06/2022	07/06/2022	8		
	HU-26	07/06/2022	07/06/2022	16		

Realizado por: Silva, J., 2022

4.3.3. Fase de desarrollo del software educativo

4.3.3.1. Definición de la arquitectura del software educativo

El diseño de la arquitectura del software educativo se la realizó de acuerdo con las necesidades del usuario. Como se observa en el Gráfico 1-3 el enfoque dispuesto es Jamstack debido a que los

estudiantes serán quienes utilicen el software educativo desde distintos puntos a través de un navegador web (cliente), mientras que la fuente de respuestas a las solicitudes de los estudiantes será el backend compuesto por Strapi conectado con diferentes microservicios. El cliente utiliza el framework NEXTJS con tecnología de generación de sitios estáticos para garantizar una experiencia de rapidez al usuario. Para el servidor se utiliza Strapi que implementa el patrón modelo, vista, controlador (MVC) junta a la arquitectura REST para la comunicación. Para el almacenamiento de archivos se utiliza S3 de AWS, la gestión de correos electrónicos con SendGrid y la gestión de formularios mediante Typeform.

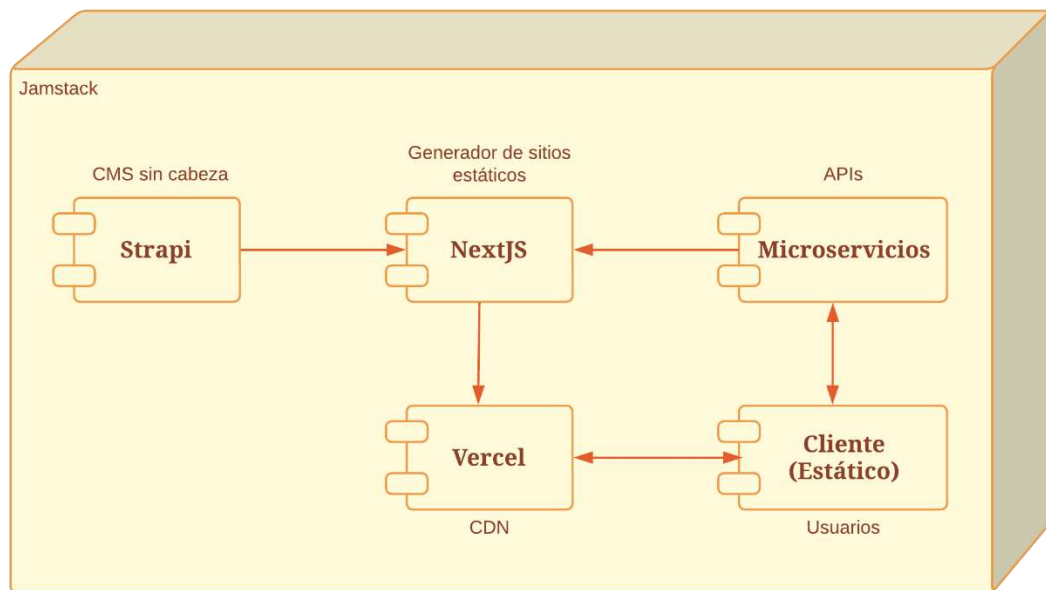


Gráfico 4-3: Arquitectura del software educativo

Realizado por: Silva, J., 2022.

4.3.3.2. Definición del estándar de codificación

El estándar de codificación se define principalmente para las clases, métodos y variables para tener consistencia y mejorar la lectura y entendimiento del código fuente para posteriores actualización o mantenimiento del software educativo. En la Tabla 19-3 y Tabla 20-3 se presentan los estándares utilizados los diferentes elementos en NextJS, y Strapi respectivamente.

Tabla 19-3 Estándar de codificación para NextJS

Elemento	Estándar	Ejemplo
Clases	Pascal Case	UserClass
Métodos	Camel Case	getUserInfo
Variables	Snake Case	user_id

Realizado por: Silva, J., 2022

Tabla 20-3 Estándar de codificación para Strapi

Elemento	Estándar	Ejemplo
Colecciones y componentes	Pascal Case	Quiz
Atributos	Snake Case	quiz_title

Realizado por: Silva, J., 2022

En el Anexo F se describen las especificaciones de codificación en NextJS, y la creación de colecciones en Strapi.

4.3.3.3. Diagramas UML

Diagrama de clases

El diagrama de clases permite visualizar los atributos, métodos y la relación entre los objetos. En el Grafico 2-3 se muestra un total de clases que se usaran para crear las colecciones en Strapi.

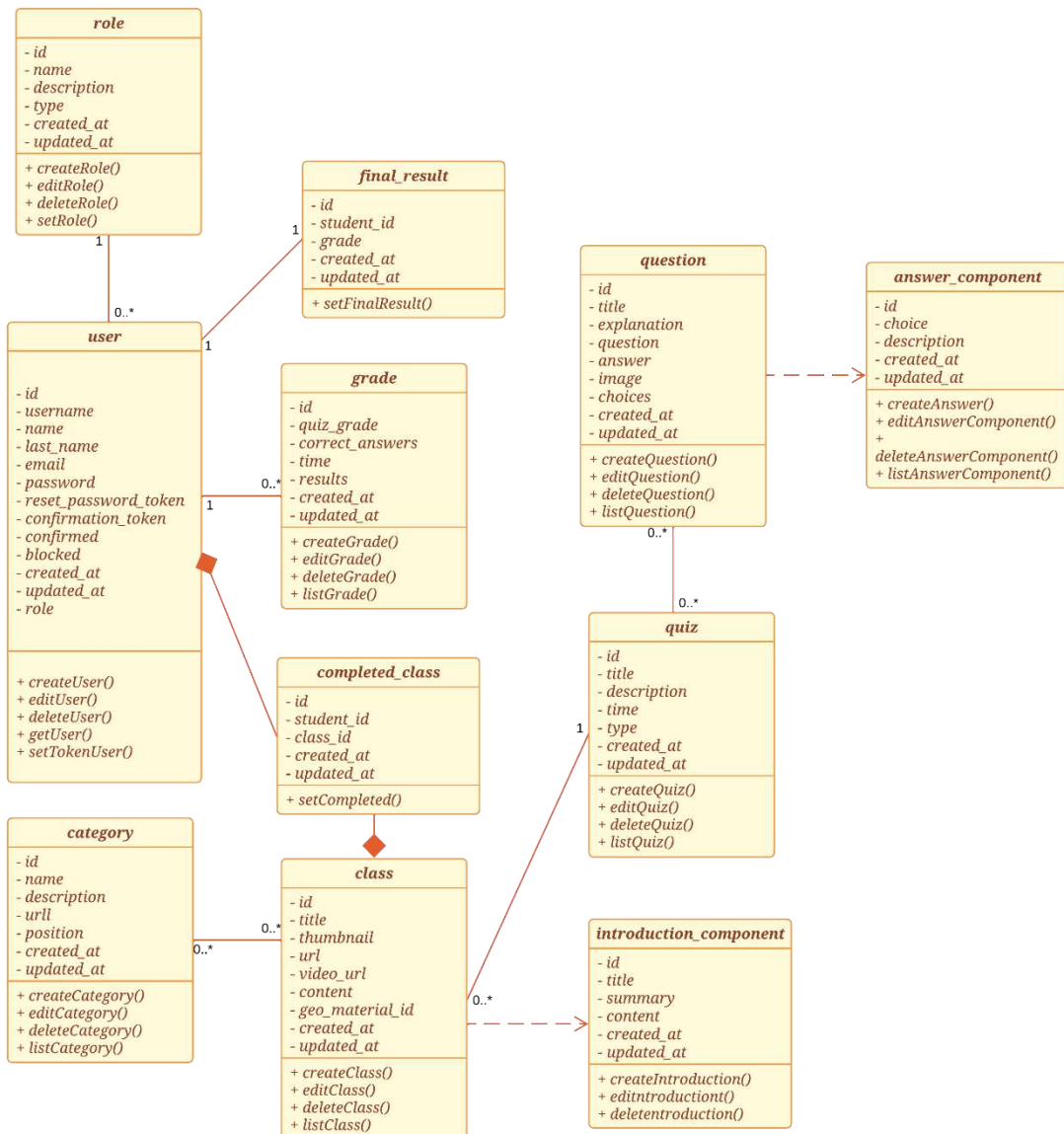


Gráfico 2-3: Diagrama de clases

Realizado por: Silva, J., 2022.

Diagrama de casos de uso

El diagrama de casos de uso permite visualizar las acciones o funcionalidades de software educativo y las personas que interactúan. En el Gráfico 3-3 se muestra el diagrama de casos de uso del administrador o maestro con Strapi.

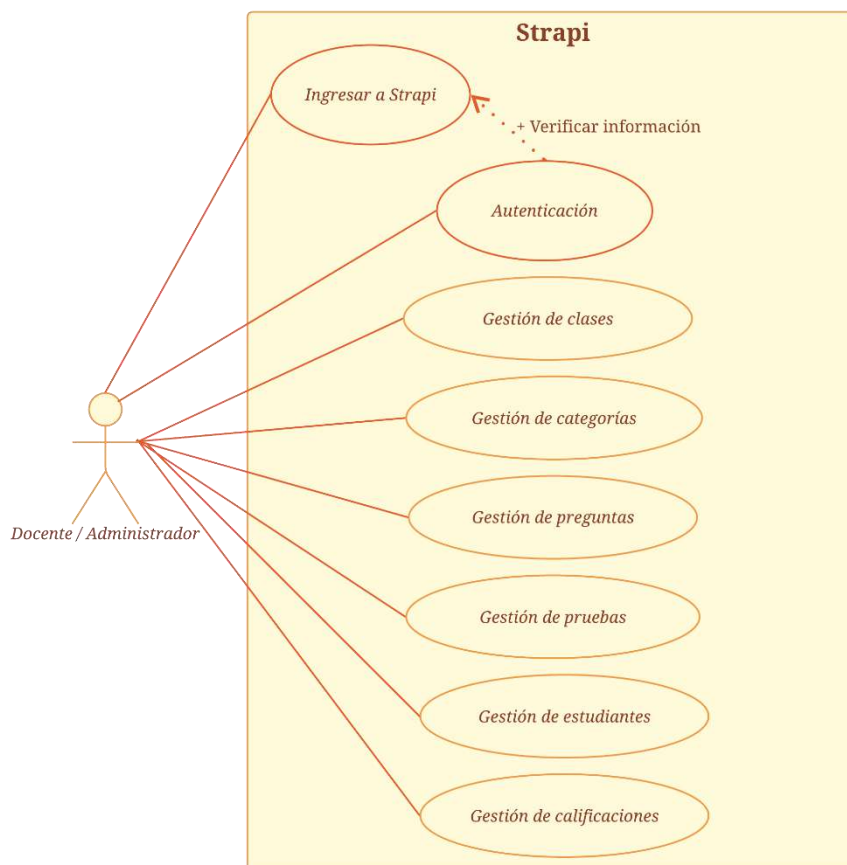


Gráfico 3-3: Diagrama de casos de uso Strapi

Realizado por: Silva, J., 2022.

En el anexo G se encuentran los diagramas de casos de uso para los demás módulos.

4.3.3.4. Diseño de la base de datos

Con el diseño de la base de datos se busca tener persistencia y acceso a la información actualizada de manera precisa, por lo que, se parte de análisis previo del escenario junto con el proceso que este tenga, teniendo así el diseño conceptual, diseño lógico, diseño físico y diccionario de datos.

Diseño conceptual

Con el propósito de mostrar el escenario de forma entendible a continuación se muestra en el Gráfico 4-3 el diseño conceptual de la base de datos.

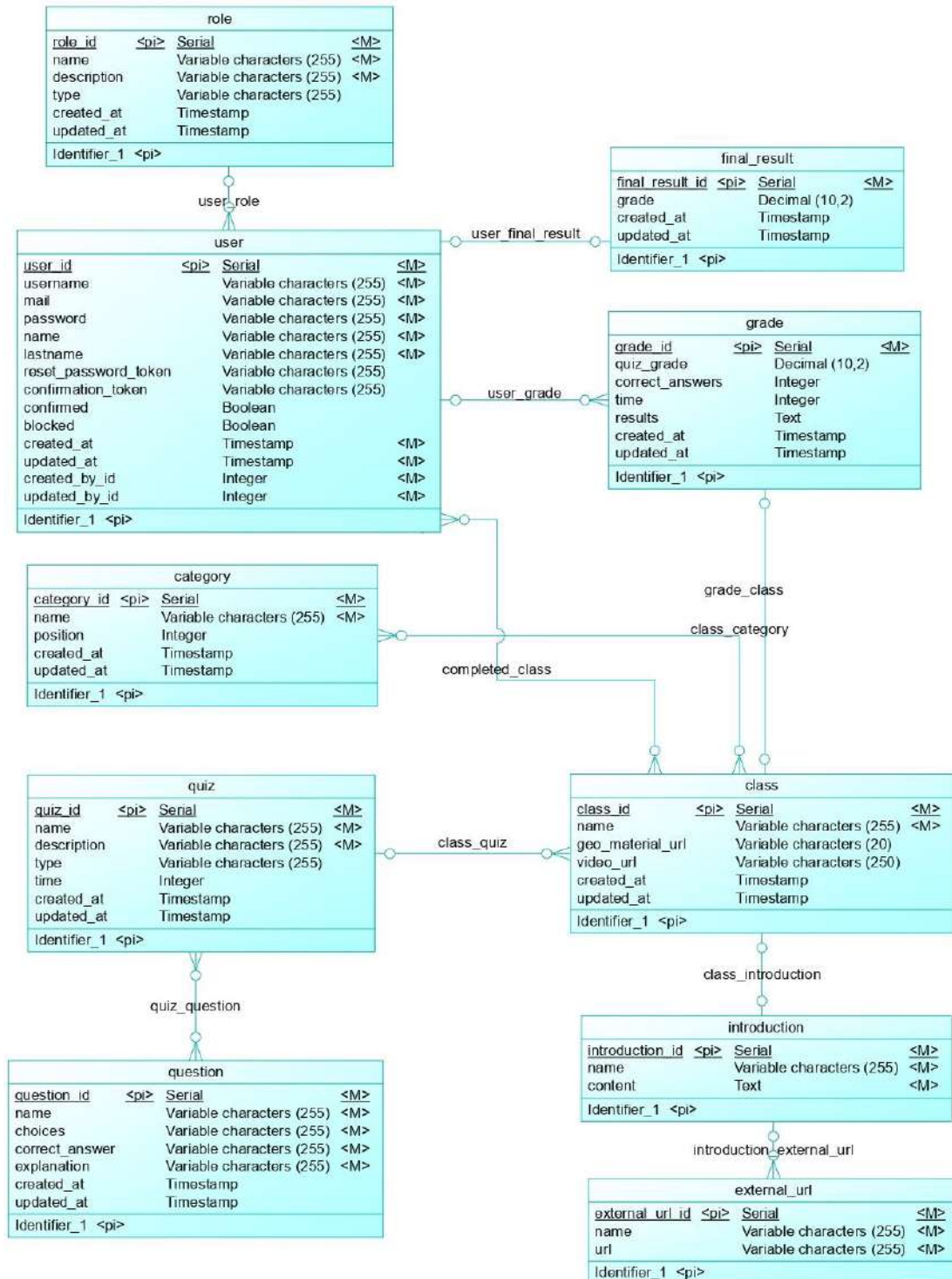


Gráfico 4-3: Diseño conceptual de la base de datos

Realizado por: Silva, J., 2022.

Diseño lógico

En esta etapa se transforma el modelo conceptual en un conjunto de objetos con el propósito de ser parte funcional de lo que va a usar el diseño físico. A continuación, se muestra en el Gráfico 5-3 el diseño lógico de la base de datos.

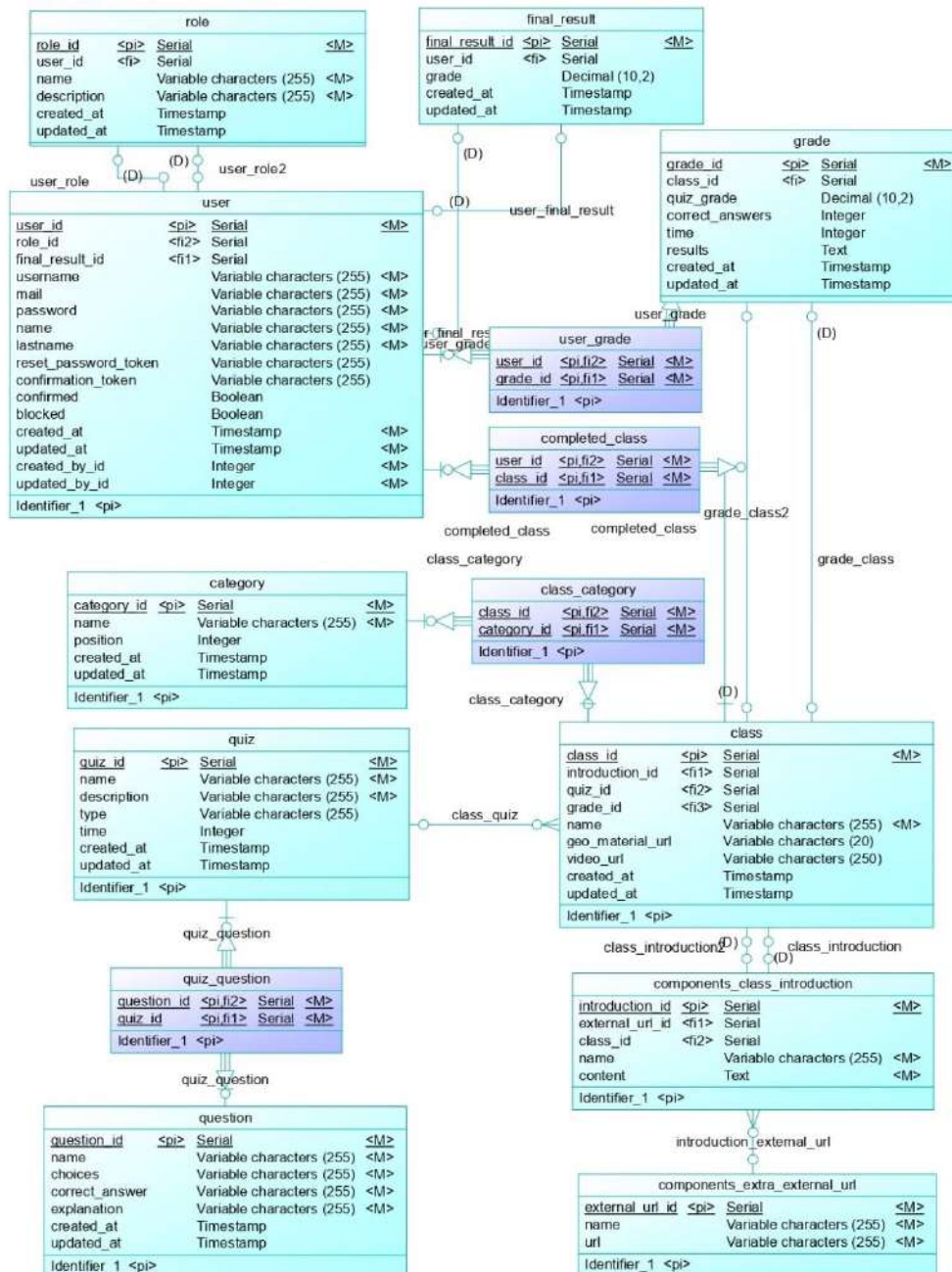
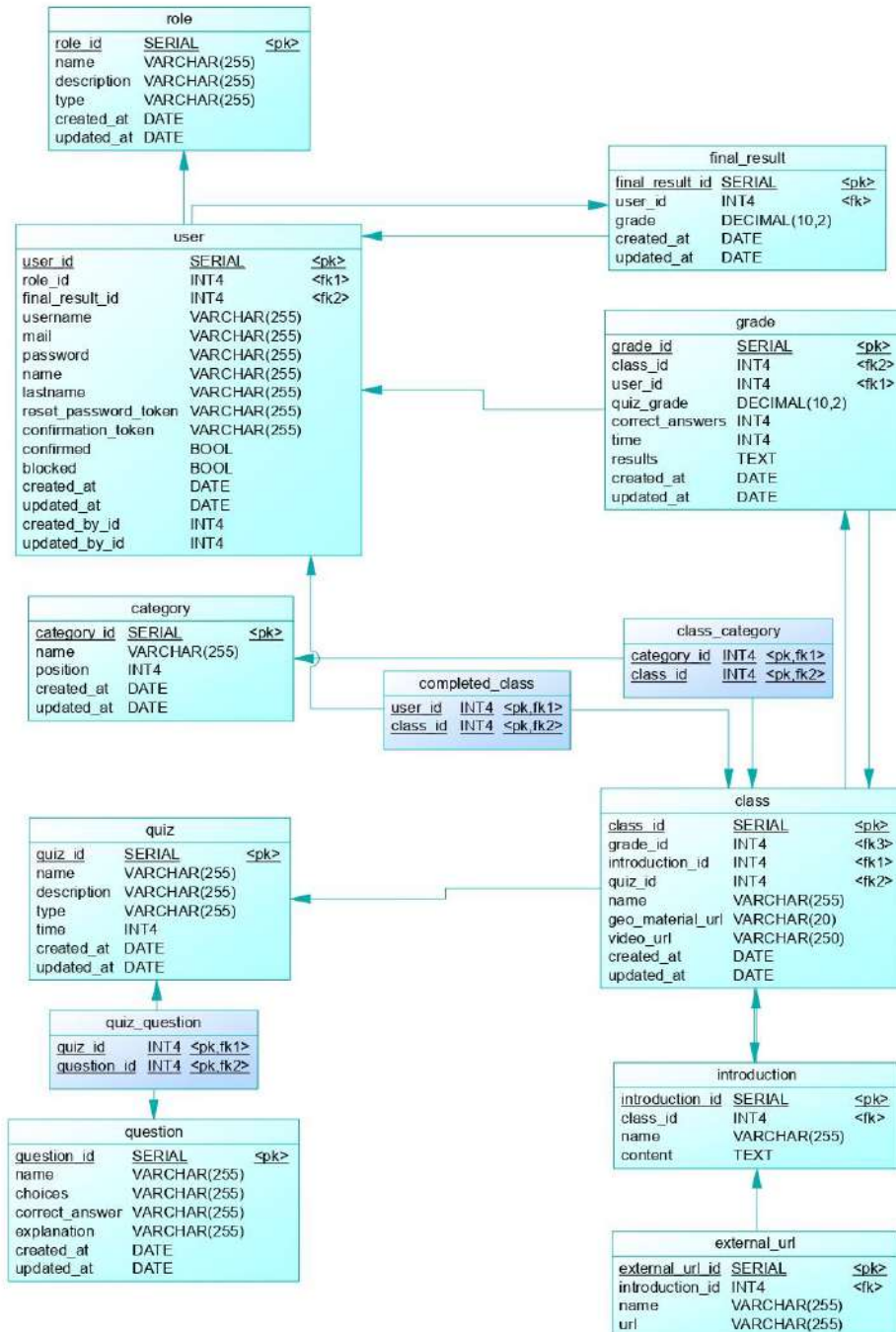


Gráfico 5-3: Diseño lógico de la base de datos

Realizado por: Silva, J., 2022.

Diseño físico

En esta etapa se transforma el diseño lógico en una solución la cual es la que se va a implementar en el sistema, así como se observa a continuación en el Gráfico 6-3 el diseño físico de la base de datos.



1 / 1

Gráfico 6-3: Diseño físico de la base de datos

Realizado por: Silva, J., 2022.

Diccionario de datos

El diccionario de datos describe las características de los atributos que se manejan dentro de la base de datos, en este caso el campo, el tipo de dato, la obligatoriedad y las restricciones. En la Tabla 21-3 está descrita la entidad **user**. En el Anexo H se encuentra el diccionario de datos de las tablas del modelo físico de la base de datos.

Tabla 21-3 Diccionario de datos

Nombre:	USER				
Descripción:	Usuario del software educativo				
Campo	Descripción	is_key	Tipo de dato	NULL	Restricción
id	ID del usuario	PK	integer(32)	NOT NULL	
username	Nickname del usuario		character varying(255)	NULL	
email	Correo electrónico		character varying(255)	NULL	
password	Contraseña del usuario		character varying(255)	NULL	codificación: base64
reset_password_token	Token para restablecer la contraseña		character varying(255)	NULL	json
confirmation_token	Token de confirmación		character varying(255)	NULL	json
confirmed	Estado de confirmación del usuario		boolean	NULL	
blocked	Estado de bloqueo del usuario		boolean	NULL	
created_at	Fecha y hora de creación		timestamp without time zone(6)	NULL	created_at <= current_date
updated_at	Fecha y hora de actualización		timestamp without time zone(6)	NULL	updated_at <= current_date

Realizado por: Silva, J., 2022

4.3.3.5. Diseño de las interfaces

Pantalla de Inicio de Sesión

En la Figura 1-3, se observa el diseño conceptual de la pantalla de inicio de sesión del software educativo.

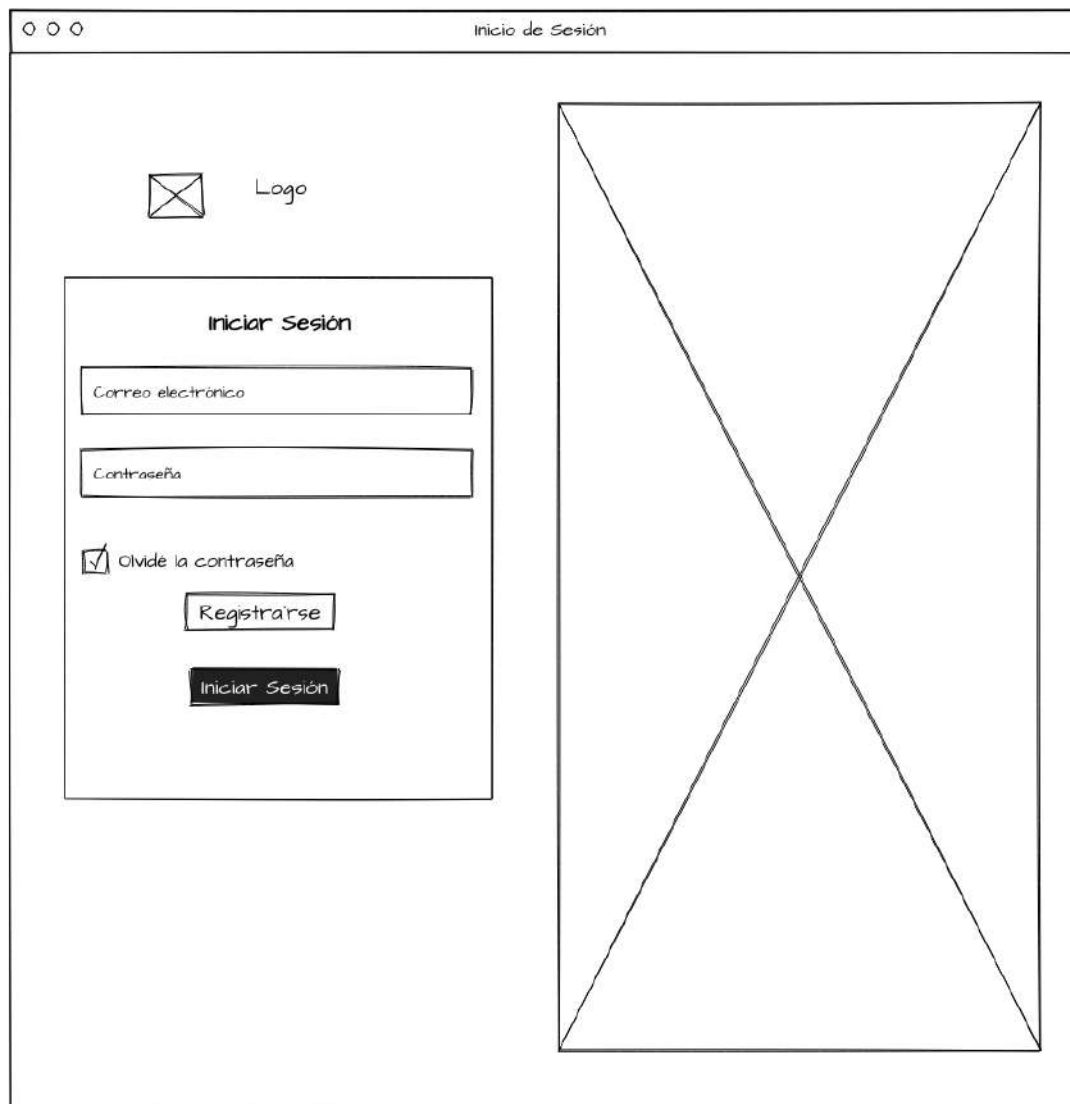


Figura 1-3: Inicio de sesión

Realizado por: Silva, J., 2022.

Pantalla de principal (listado de clases)

En la Figura 2-3, se observa el diseño de la pantalla la pantalla principal con el listado de clases, cada clase presenta su título, una carátula y el estado (“completo”, “incompleto”). La estructura de la información se la realiza con categorías, la disposición a través de una grilla (Grid) y la navegación es vertical.

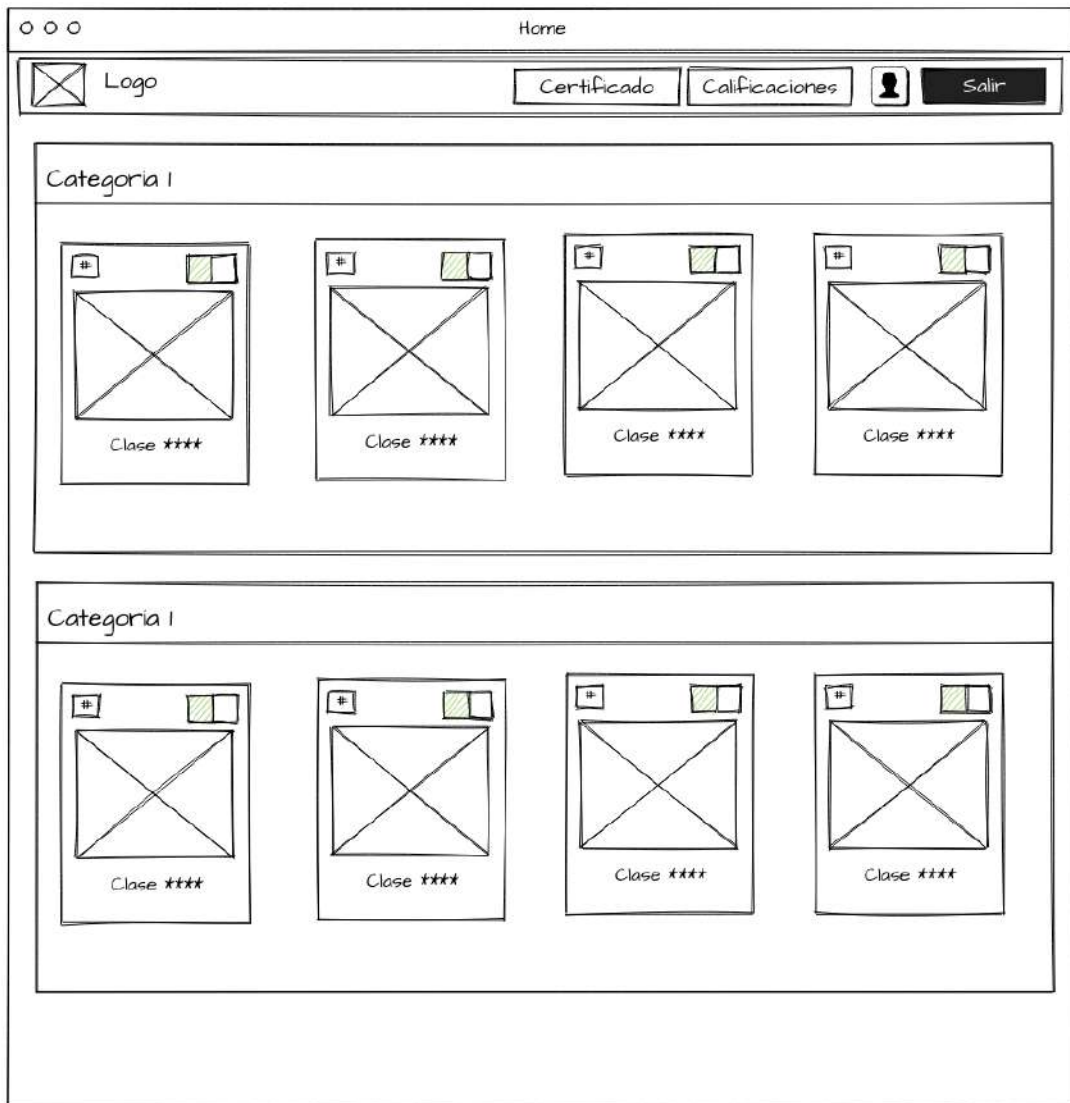


Figura 2-3: Pantalla principal (Listado de clases)

Realizado por: Silva, J., 2022.

Pantalla de cada clase (Geometría dinámica)

En la Figura 3-3, se presenta el diseño de la pantalla para cada clase, se tiene el título, el estado en el que se encuentra y el contenido correspondiente. La navegación del contenido se la realiza de manera horizontal a través de tabulación, la primera sección corresponde al contenido teórico, la segunda sección es un vídeo explicativo, la tercera sección (Figura 6-3) corresponde a geometría dinámica y la última sección es una prueba de selección múltiple.

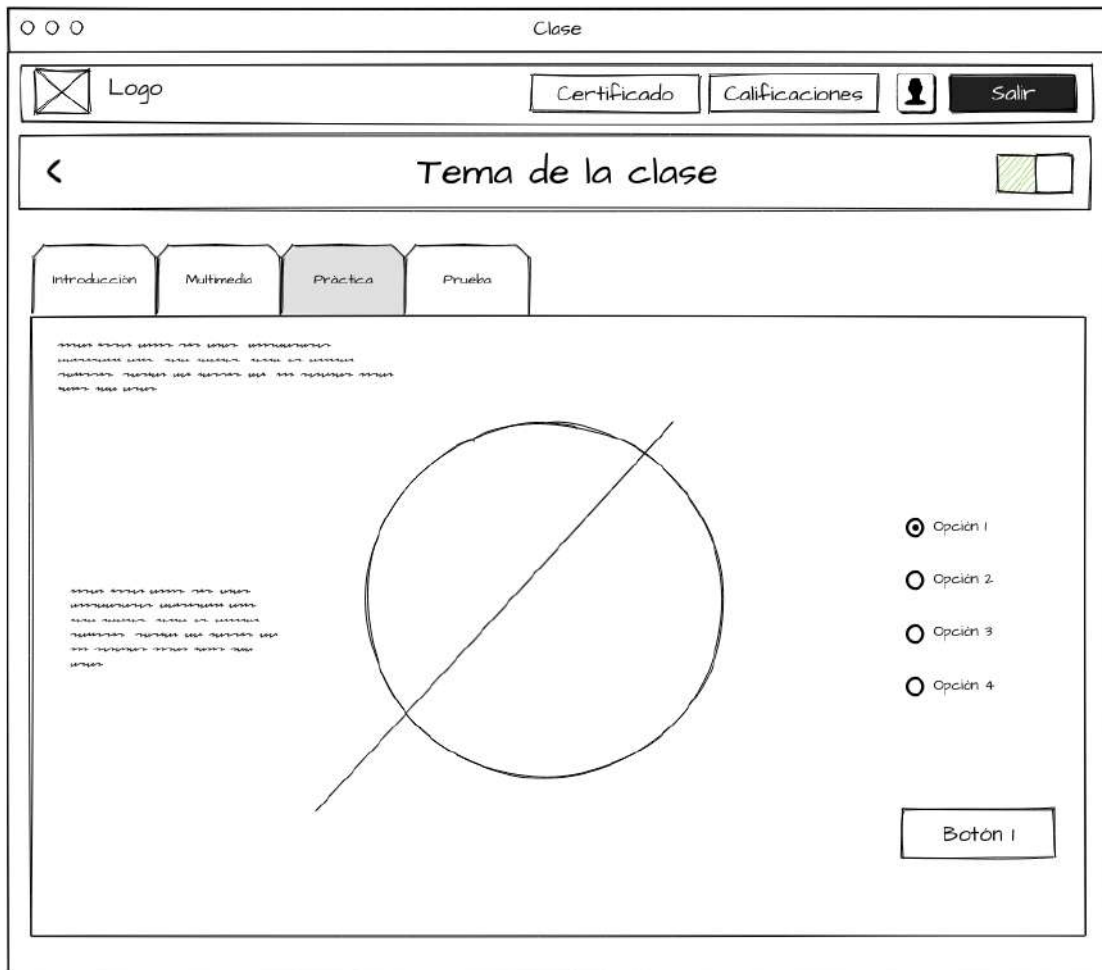


Figura 3-3: Pantalla de la clase (Geometría dinámica)

Realizado por: Silva, J., 2022.

Las interfaces de los otros módulos del software educativo se encuentran en mayor detalle en el Anexo I.

4.3.3.6. Reuniones y Entregables

La participación del cliente la Ing. Mirian Avila es indispensable para el correcto avance del proyecto, junto con el equipo Scrum se realizaron reuniones constantes para la recolección, análisis y presentación de la información sobre el software educativo, como se detalla en la Tabla 22-3, en cada reunión se presentaron los resultados de las actividades desarrolladas, al igual que

las pruebas de aceptación para verificar y validar el productor software buscando aumentar su calidad.

Tabla 22-3 Reuniones y Entregables

N#	Fecha	Actividades	Responsables
1	06/05/2022	Análisis del problema, definición de requerimientos	Ing. Diego Avila Ing. Miriam Avila Est. John Silva
2	16/05/2022	Detalle de historias técnicas y de usuario, diseño de arquitectura del software educativo, diseño de base de datos, definición de estándares de codificación y diseño de las interfaces.	Ing. Diego Avila Ing. Miriam Avila Est. John Silva
3	20/05/2022	Sistema de autenticación y recuperación de contraseñas	Ing. Diego Avila Ing. Miriam Avila Ing. Miguel Duque Est. John Silva
4	24/05/2022	Creación de colecciones principales en Strapi	Ing. Miriam Avila Est. John Silva
5	26/05/2022	Codificación y presentación de pantallas de usuario para las clases	Ing. Miriam Avila Est. John Silva
6	04/06/2022	Codificación y presentación de pantallas para las evaluaciones y calificaciones, pantalla para los certificados	Ing. Miriam Avila Est. John Silva
7	10/06/2022	Codificación y presentación de pantallas para el seguimiento de los estudiantes	Ing. Miriam Avila Est. John Silva
8	22/06/2022	Despliegue de prueba y demo con estudiantes	Ing. Diego Avila Ing. Miriam Avila Est. John Silva
9	13/07/2022	Corrección de errores y despliegue a producción	Ing. Diego Avila Ing. Miguel Duque Est. John Silva

Realizado por: Silva, J., 2022

4.3.3.7. Fase de pruebas

El desarrollo y la ejecución de pruebas se divide en 2 categorías que corresponden a Strapi (Backend) que se realizan con la herramienta Postman para verificar que el servicio API REST funciona correctamente y con NextJS (Frontend) se utiliza Selenium y Chrome DevTools para realizar pruebas de exploratorias automatizadas sobre la interfaz de usuario. Las suites y casos de prueba serán gestionados desde TUSKR que se encuentra integrado con Jira para la generación de informes del proyecto.

En la Figura 4-3 se observa las suites de pruebas consideradas, en total 7 que corresponden a los módulos principales del software educativo, con un total de 69 casos de pruebas.

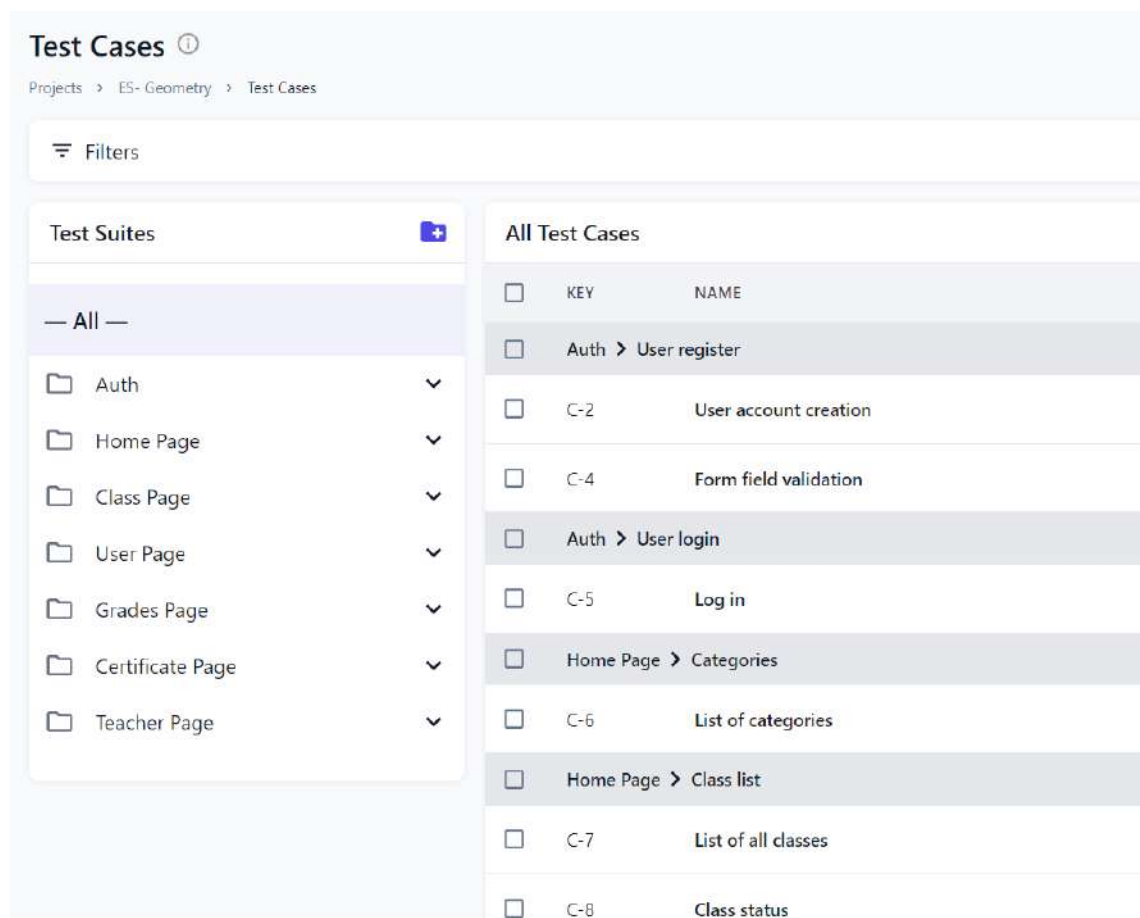


Figura 4-3: Suite de pruebas en TUSKR

Realizado por: Silva, J., 2022.



Después se procede con la ejecución de cada caso de prueba, en la Figura 5-3 se observa el procedimiento para completar un caso de prueba, y en la Figura 6-3 se presenta los resultados de la ejecución de las pruebas.

C-2 · User account creation

Auth > User register > C-2

Passed Assigned To: JSE Estimated Time: 5 minutes Actual Time: 3 minutes

Steps

INSTRUCTIONS	EXPECTED RESULT
<p>1 Open web browser and access the following link https://e-geometry-js.vercel.app/</p>	<p>Access to educational software on the login page.</p> 
<p>2 Select the option: "create an account"</p>	<p>The form opens to enter the user data.</p> 

Add Result

Status*: Passed

Reassign To: John Silva Espín

Time Spent: 12 minutes

Comments: Test case: User registration completed successfully

Field Set:

Attachments

Max 5 files of 15 MB each.

Upload

out.rar

Save Save & Next

Figura 5-3: Ejecución del caso de prueba “User account creation”

Realizado por: Silva, J., 2022.

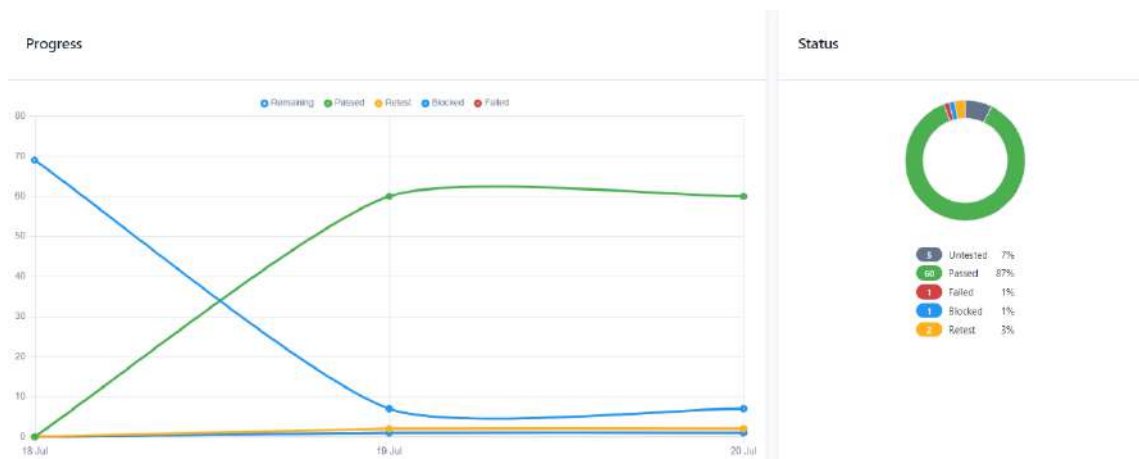


Figura 6-3: Resultado de la ejecución de los casos de pruebas.

Realizado por: Silva, J., 2022.

Los reportes e información adicionales se encuentran en el Plan de pruebas (Anexo J).

También se realizaron las pruebas de aceptación del software educativo por parte del cliente, para validar que el software educativo cumple con las necesidades presentadas. En la tabla se presenta el formato utilizado para documentar las pruebas. En la Tabla 23-3 se detalla la prueba de aceptación para las actividades relacionadas con la autenticación de usuarios.

Tabla 23-3 Prueba de Aceptación

PRUEBA DE ACEPTACIÓN			
ID	PA-01	Nombre	Autenticación de usuarios
Responsables	Ing. Diego Avila Ing. Miriam Avila Ing. Miguel Duque Est. John Silva		Fecha 18/05/2022
Descripción	Se verificará que los usuarios puedan crear una cuenta, iniciar sesión y recuperar su contraseña.		
Estado de la prueba	EXITOSA		
ACTIVIDADES EVALUADAS			
ID_PA	Descripción		Estado
HU01 – PA01	Validación de campos requeridos para el registro		Finalizada
HU01 – PA02	Verificar que el usuario sea único		Finalizada
HU01 – PA03	Notificación y verificación a través de correo electrónico		Finalizada
HU01 – PA04	Verificar almacenamiento de usuarios en Strapi		Finalizada
HU02 – PA01	Ingreso al software educativo con las funcionalidades dependiendo del rol		Finalizada
HU02 – PA02	Notificación de error de inicio de sesión		Finalizada
HU03 – PA01	Verificar que el usuario esté registrado		Finalizada
HU03 – PA02	Enviar enlace de recuperación a correo electrónico		Finalizada
TAREAS DE INGENIERÍA			
ID_TI	Descripción		Resultado
HU01 – TI01	Desarrollar la interfaz de registro e inicio de sesión		Exitoso
HU01 – TI02	Implementar el servicio web para la comunicación del software educativo con Strapi		Exitoso
HU02 – TI01	Implementar el servicio para la autenticación de los usuarios en el software educativo		Exitoso
HU03 – TI01	Desarrollo de la interfaz de recuperación de contraseña		Exitoso
HU03 – TI02	Verificación de existencia de usuario en la base de datos		Exitoso
HU03 – TI03	Envío de enlace de recuperación al correo electrónico		Exitoso

Realizado por: Silva, J., 2022

En el anexo K se detallan las demás pruebas de aceptación.

4.3.4. Fase de cierre

4.3.4.1. Gestión del proyecto

La gestión de proyecto se observa en el Gráfico 7-3, el cual demuestra un valor de total de 160 puntos estimados (Eje Y) divididos en un total de 4 sprints (Eje X). Los puntos estimados se representan con color azul y los puntos reales con color rojo. El avance en el desarrollo de las historias de usuario entre el sprint 1 y 2 estuvo ajustado debido a problemas de incompatibilidad de ciertas herramientas, pero a partir del sprint 3 el avance fue correcto permitiendo la finalización a tiempo del proyecto.

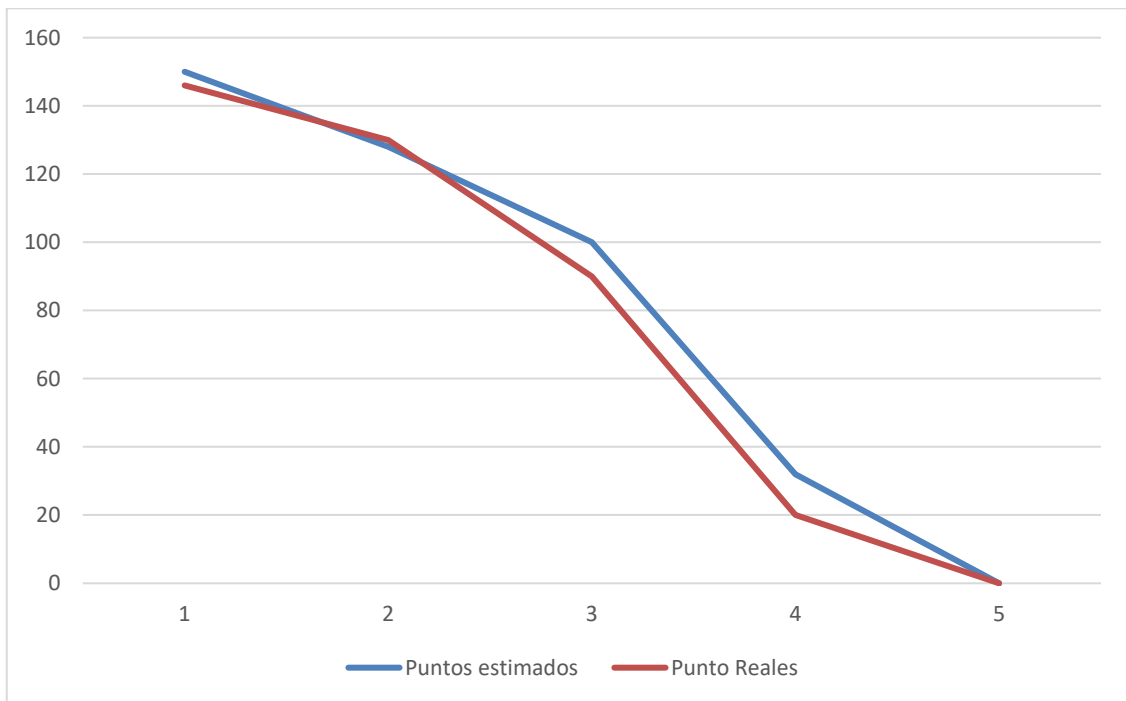


Gráfico 7-3: Burn Down Chart

Realizado por: Silva, J., 2022.

CAPITULO IV

4. RESULTADOS

En este capítulo se analizan y describen los resultados obtenidos de la medición de la usabilidad del software educativo.

5.1. Evaluación de las subcaracterísticas de usabilidad

Para la evaluación de la usabilidad se seleccionó 3 subcaracterísticas en relación con el estudio de (Cocunubo-Suárez, Parra-Valencia y Otálora-Luna 2018), que tiene como objetivo describir los aspectos necesarios para evaluar los sistemas virtuales de aprendizaje teniendo como base el estándar ISO 25000-SQuaRE. La recolección de datos se realizó con la aplicación de un cuestionario (Anexo L) a 40 estudiantes de primero de bachillerato de la Unidad educativa Vigotsky para determinar su grado de satisfacción con el software educativo.

En la Tabla 1-4 se presentan los resultados del cuestionario para la subcaracterística de Facilidad pedagógica, en donde se observa que la pregunta con mayor puntaje es la 3 relacionada al indicador de “retroalimentación” con un promedio de 4.7 sobre 5, mientras que la pregunta 2 correspondiente al indicador “evaluación de aprendizaje” es la de menor puntaje con 4.3 sobre 5.

Tabla 1-4 Resultados: Facilidad pedagógica

Pregunta	Sumatoria (Máximo75)	Promedio sobre 5
1. La información de la introducción y el video de cada clase es clara y fácil de entender	69	4.6
2. La evaluación de cada clase tiene preguntas acordes a la temática y las puedo responder sin mayor problema	64	4.3
3. Al finalizar la evaluación puedo revisar mis respuestas correctas o erróneas, y se explica el procedimiento o manera de resolver cada pregunta	70	4.7
Promedio		4.51

Realizado por: Silva, J., 2022

En la Tabla 2-4 se presentan los resultados para la subcaracterística de Facilidad de entendimiento o comprensibilidad, la pregunta con mayor puntaje es la 9 con un promedio de 4.6, mientras que la pregunta 6 es la de menor puntaje con 4.0.

Tabla 2-4 Resultados: Facilidad de entendimiento o comprensibilidad

Pregunta	Sumatoria (Máximo75)	Promedio sobre 5
4. La pantalla de las clases es agradable	66	4.4
5. Me gustó interactuar con las figuras interactivas	63	4.2
6. Tuve complicaciones para entrar en la pantalla de calificaciones	60	4.0
7. La información de la introducción y el video de cada clase es clara y fácil de entender	66	4.4
8. Al ir a las diferentes pantallas pude regresar a donde estaba antes.	61	4.1
9. Recomendaría este software educativo a un amigo	69	4.6
Promedio		4.26

Realizado por: Silva, J., 2022

Los resultados para la subcaracterística Facilidad de uso están en la Tabla 3-4, la pregunta con mayor puntaje es la 14 con un promedio de 4.8, mientras que la pregunta 11 es la de menor puntaje con 3.9.

Tabla 3-4 Resultados: Facilidad de uso

Pregunta	Sumatoria (Máximo75)	Promedio sobre 5
10. No tuve ningún error mientras utilizaba el software educativo	60	4.0
11. Si tuve un error pude volver atrás y seguir navegando	58	3.9
12. Se mostró un mensaje después de completar una clase o actualizar mi perfil	60	4.0
13. Al poner un correo electrónico equivocado para iniciar sesión se mostró un mensaje	67	4.5
14. Es cómodo navegar desde la Introducción de la clase hasta la Prueba	72	4.8
15. Es fácil interactuar con todo el software educativo	66	4.4
Promedio		4.28

Realizado por: Silva, J., 2022

Para realizar el análisis de las subcaracterísticas se obtuvo el promedio de cada grupo, dando como resultados que la Facilidad pedagógica es la mejor puntuada con 4.51, le sigue la Facilidad de uso con 4.28 y finalmente la Facilidad de entendimiento con 4.26.

Por lo tanto, las funcionalidades que el software educativo provee a los estudiantes para el desarrollo del proceso de aprendizaje son las más valoradas, dentro de las que destacan la claridad y facilidad de los materiales de aprendizaje, los modelos geométricos interactivos y la retroalimentación después de realizar las pruebas. Las siguientes funcionalidades necesarias para que el software educativo sirva de apoyo son una correcta interfaz de usuario, la estructura de los contenidos y una fácil navegación.

5.2. Análisis de la usabilidad del software educativo

Para obtener el nivel de usabilidad del software educativo se asignó ponderaciones porcentuales a las sus características evaluadas como se detalla en la Tabla 4-4, y en el Gráfico 1-4 se observa la distribución del porcentaje de cada subcaracterística en relación con el 100%:

Tabla 4-4 Ponderación de la usabilidad

Subcaracterística	Promedio	Porcentaje alcanzado	Ponderación
Facilidad Pedagógica	4.51	27.07%	30%
Facilidad de entendimiento o comprensibilidad	4.26	34.22%	40%
Facilidad de uso u operabilidad	4.28	25.53%	30%
Total		86.82%	100%

Realizado por: Silva, J., 2022

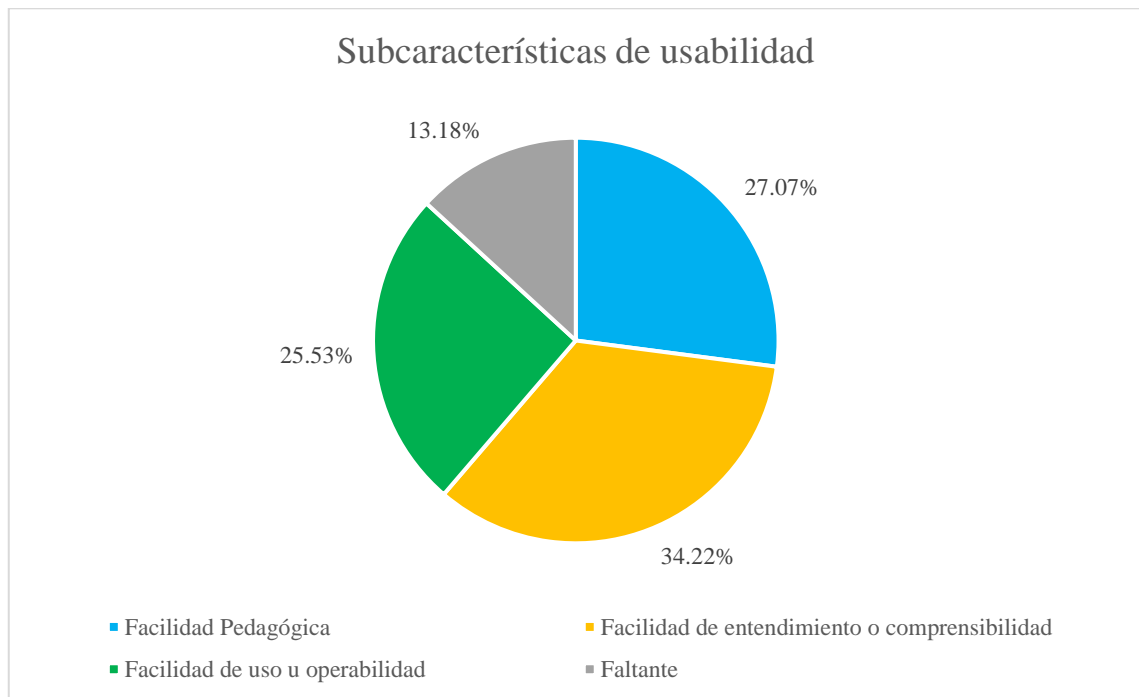


Gráfico 1-4: Representación de las subcaracterísticas de usabilidad

Realizado por: Silva, J., 2022.

A continuación, en el Gráfico 2-4 se representa el nivel total de usabilidad alcanzado y el porcentaje no cubierto.

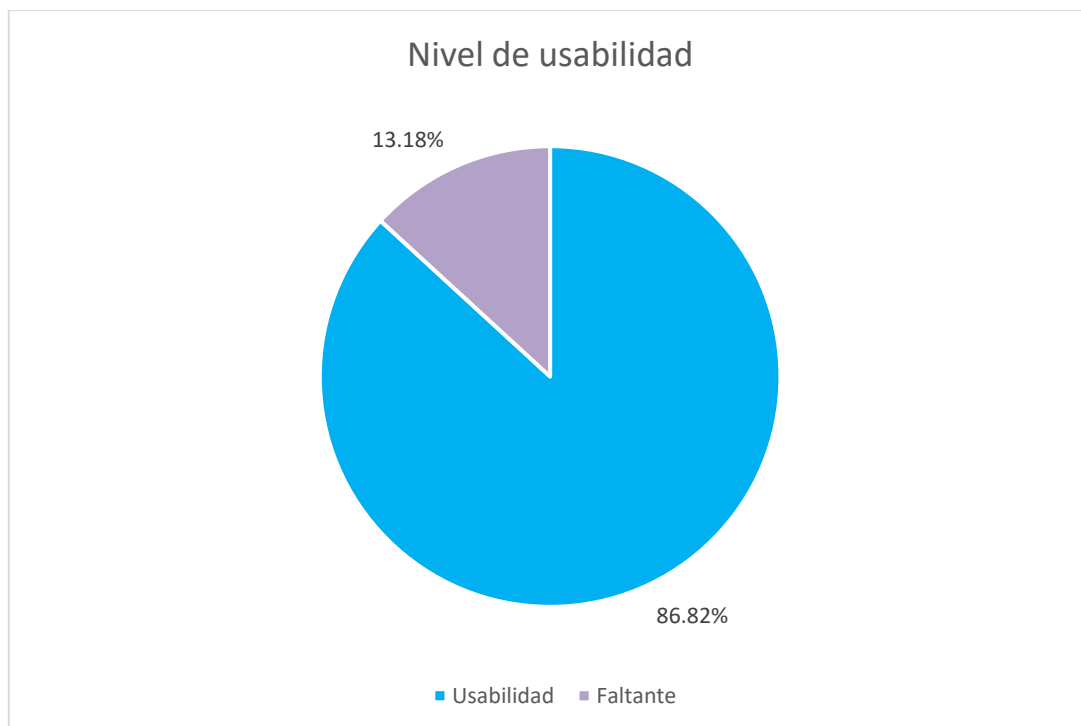


Gráfico 2-4: Nivel de usabilidad alcanzado

Realizado por: Silva, J., 2022.

5.3. Determinación de la calidad del software educativo

Para determinar el nivel de calidad del software educativo se utilizan los indicadores propuestos por (Quichimbo Pereira et al. 2021) que relacionan una escala de medición, con una puntuación y el nivel de satisfacción cómo se presenta en la Tabla 5-4 a continuación:

Tabla 5-4 Indicadores de calidad

Escala de medición	Puntuación	Grado de satisfacción
87.5%-100%	Cumple con los requisitos	Muy satisfactorio
50 % - 87.4 %	Aceptable	Satisfactorio
27.5% - 49%	Mínimamente aceptable	Insatisfactorio
0-27,4%	Inaceptable	

Fuente: Pereira et al. 2021

Realizado por: Silva, J., 2022

En base a esto se concluye que el software educativo al contar con un porcentaje de usabilidad del 86,82% está dentro de la escala de medición de [50% a 87,4%] lo que otorga una puntuación de “Aceptable” contando con un grado de satisfacción “Satisfactorio”.

CONCLUSIONES

- Después de la revisión de literatura se determinó que mediante un correcto aprendizaje de la geometría se desarrollan habilidades importantes como la resolución de problemas, el pensamiento crítico, razonamiento lógico y espacial. Las metodologías de aprendizaje actuales todavía utilizan la memorización de fórmulas y teoremas, dando como resultado un aprendizaje mecánico. Los maestros y estudiantes disponen de recursos informáticos como aplicaciones de geometría dinámica de las que no se aprovecha todo su potencial.
- Para el desarrollo del software educativo se utilizó la metodología Scrum, obteniendo resultados tempranos con entregas continuas de funcionalidades, comunicación con el cliente, una correcta gestión de cambios y mayor productividad. Con la arquitectura Jamstack se construyó un software educativo a medida. Con Strapi se cubrieron requerimientos específicos del proyecto gracias a la integración de servicios internos y externos, mientras que con NextJS las páginas se crearon de manera automática únicamente a partir de los datos entregados por Strapi, al finalizar con la característica de generación de sitios estáticos de NEXTJS se prepararon los archivos para el despliegue en Vercel y DigitalOcean.
- Se determinó el nivel de usabilidad de Geoline a través de un análisis descriptivo e inferencial aplicando un cuestionario para medir la satisfacción de los estudiantes en relación con el software educativo. Entre las características necesarias para que el software educativo sirva de apoyo en el proceso de aprendizaje se destacan: la claridad y facilidad de los materiales de aprendizaje, los modelos geométricos interactivos y la retroalimentación después de realizar las pruebas. Las siguientes características son una correcta interfaz de usuario, la estructura de los contenidos y una fácil navegación. Finalmente, se obtuvo un 86.82 % concluyendo que el software educativo es aceptable y satisfactorio.

RECOMENDACIONES

- La selección de esta pila de tecnologías si se desea implementar páginas auto generadas de manera dinámica en base a rutas administradas desde un gestor de contenidos.
- El uso de estándares de codificación ayuda a tener un código limpio y completamente incorporado tanto en la parte del servidor como del cliente.
- El uso de herramientas de geometría dinámica que trabajan sobre la web y pueden ser integradas en cualquier sitio web para ampliar los recursos dinámicos del software educativo.
- Seguir el enfoque de las metodologías ágiles permite tener un mejor desarrollo del proyecto, al estar en constante comunicación con el cliente las funcionalidades pueden ser modificadas o introducidas sin mayor problema.
- El uso de técnicas de diseño responsivo para que el software educativo se pueda utilizar desde cualquier dispositivo.

BIBLIOGRAFÍA

ABHISHEK KOTHARI, 2022. *11 CMS sin cabeza a considerar para aplicaciones modernas*. [en línea]. [Consulta: 18 abril 2022]. Disponible en: <https://geekflare.com/es/headless-cms/>.

ABRATE, R., POCHULU, M. y DELGADO, G., 2003. *Caracterización de las actividades de Geometría que proponen los textos de Matemática*. *Revista Iberoamericana de Educación* [en línea], vol. 39, no. 2001, pp. 1-9. [Consulta: 15 enero 2022]. ISSN 1681-5653. DOI 10.35362/RIE3912598. Disponible en: [//rieoei.org/RIE/article/view/2598](http://rieoei.org/RIE/article/view/2598).

AL-FEDAGHI, S., 2011. *Developing web applications*. *International Journal of Software Engineering and its Applications*, vol. 5, no. 2, pp. 57-68. ISSN 17389984. DOI 10.1016/b978-192899448-0/50015-x.

ALMEIDA, M., 2000. *Desarrollo Profesional Docente en Geometría: análisis de un proceso de Formación a Distancia* [en línea]. S.l.: s.n. [Consulta: 15 enero 2022]. Disponible en: <https://dialnet.unirioja.es/servlet/tesis?codigo=3172&info=resumen&idioma=SPA>.

ARAYA, R.G. y ALFARO, E.B., 2009. *Algunas reflexiones sobre la didáctica de la geometría*. En: G. BALINT, B. ANTALA, C. CARTY, J.-M.A. MABIEME, I.B. AMAR y A. KAPLANOVA (eds.), *Cuadernos de Investigación y Formación en Educación Matemática* [en línea], pp. 343-354. [Consulta: 31 agosto 2022]. ISSN 2215-5627. DOI 10.2/JQUERY.MIN.JS. Disponible en: <https://revistas.ucr.ac.cr/index.php/cifem/article/view/6915>.

ARIAS, M., LÓPEZ, Á. y HONMY J., R., 2015. *Metodología Dinámica para el Desarrollo de Software Educativo*. [en línea], [Consulta: 31 agosto 2022]. Disponible en: <https://repositorial.cuaieed.unam.mx:8443/xmlui/handle/20.500.12579/4325>.

AUSUBEL, D., 2015. *Adquisición Y Retención Del Conocimiento: Una Perspectiva Cognitiva*. [en línea], [Consulta: 18 abril 2022]. Disponible en: https://issuu.com/luisorbegoso/docs/ausubel_-_adquisicion_y_retencion_d.

AYDOS, M., 2015. *The impact of teaching mathematics with geogebra on the conceptual understanding of limits and continuity: The case of Turkish gifted and talented students*. [en línea], [Consulta: 1 enero 2022]. Disponible en: <http://repository.bilkent.edu.tr/handle/11693/30052>.

BALSECA CHISAGUANO, E.A., 2014. *EVALUACIÓN DE CALIDAD DE PRODUCTOS SOFTWARE EN EMPRESAS DE DESARROLLO DE SOFTWARE APLICANDO LA NORMA ISO/IEC 25000* [en línea]. S.l.: s.n. [Consulta: 15 enero 2022]. Disponible en: <http://bibdigital.epn.edu.ec/handle/15000/9113>.

BANTCHEV, B.B., 2010. *A Brief Tour to Dynamic Geometry Software*. *CaMSP February 2015 Newsletter*, pp. 1-6.

- CACHUPUT, J.**, 2016. *Desarrollo de una guía didáctica como herramienta para la enseñanza de la geometría utilizando TICS, dirigido a los estudiantes del Segundo Año Bachillerato de la Unidad Educativa Pedro Vicente Maldonado*. [en línea], [Consulta: 15 enero 2022]. Disponible en: <http://dspace.esPOCH.edu.ec/handle/123456789/4999>.
- CATALDI, Z., LAGE, F., PESSACQ, R. y GARCÍA-MARTÍNEZ, R.**, 2007. *Metodología extendida para la creación de software educativo desde una visión integradora*. *Revista Latinoamericana de Tecnología Educativa-RELATEC* [en línea], vol. 2, no. 1, pp. 9-40. [Consulta: 15 enero 2022]. ISSN 1695-288X. Disponible en: <https://dialnet.unirioja.es/servlet/articulo?codigo=1252631&info=resumen&idioma=SPA>.
- CATALDI, Z. y SALGUEIRO, F.**, 2007. *Software libre y código abierto en educación*. [en línea]. S.l.: [Consulta: 9 abril 2019]. Disponible en: www.idc.com.
- COCUNUBO-SUÁREZ, J.I., PARRA-VALENCIA, J.A. y OTÁLORA-LUNA, J.E.**, 2018. *Propuesta para la evaluación de Entornos Virtuales de Enseñanza Aprendizaje con base en estándares de Usabilidad*. *TecnoLógicas*, vol. 21, no. 41, pp. 135-147. ISSN 0123-7799. DOI 10.22430/22565337.732.
- CUERVO GÓMEZ, W.O. y BALLESTEROS-RICAURTE, J.A.**, 2017. *Framework para desarrollo de aplicaciones educativas móviles, basado en modelos de enseñanza*. *Praxis & Saber* [en línea], vol. 8, no. 17, pp. 125-153. [Consulta: 18 abril 2022]. ISSN 2216-0159. DOI 10.19053/22160159.v8.n17.2018.7204. Disponible en: https://revistas.uptc.edu.co/index.php/praxis_saber/article/view/7204.
- DIAZ-NUNJA, L., RODRÍGUEZ-SOSA, J. y LINGÁN, S.K.**, 2018. *Enseñanza de la geometría con el software GeoGebra en estudiantes secundarios de una institución educativa en Lima*. *Propósitos y Representaciones*, vol. 6, no. 2, pp. 217. ISSN 2307-7999. DOI 10.20511/PYR2018.V6N2.251.
- FOWLER, F.M.**, 2019. *Scrum Artifacts. Navigating Hybrid Scrum Environments* [en línea]. S.l.: Apress, Berkeley, CA, pp. 55-57. [Consulta: 15 enero 2022]. Disponible en: https://link.springer.com/chapter/10.1007/978-1-4842-4164-6_8.
- GAMBOA, R. y BALLESTERO, E.**, 2010. *The Students' Perspective of Geometry Teaching and Learning in High School*. *Revista Electrónica Educare*, vol. XIV, pp. 1409-1451.
- GARCÉS COBOS, L.F., MONTALUISA VIVAS, Á. y SALAS JARAMILLO, E.**, 2019. *El aprendizaje significativo y su relación con los estilos de aprendizaje*. *Revista Anales* [en línea], vol. 1, no. 376, pp. 231-248. [Consulta: 18 abril 2022]. ISSN 1390-7891. DOI 10.29166/anales.v1i376.1871. Disponible en: <https://revistadigital.uce.edu.ec/index.php/anales/article/view/1871>.
- GEIGER, V., FARAGHER, R. y GOOS, M.**, 2010. *Cas-enabled technologies as 'agents provocateurs' in teaching and learning mathematical modelling in secondary school classrooms*. *Mathematics Education Research Journal* 2010 22:2 [en línea], vol. 22, no. 2, pp. 48-68.

[Consulta: 1 enero 2022]. ISSN 2211-050X. DOI 10.1007/BF03217565. Disponible en: <https://link.springer.com/article/10.1007/BF03217565>.

GIBERT, M., GINESTÀ, O. y MORA, P., 2016. *Bases de datos en PostgreSQL.* ,

GODINO, J.D. y RUIZ, F., 2002. *Geometría y su didáctica para maestros. Matemáticas y su Didáctica para Maestros. Manual para el Estudiante.* [en línea], pp. 453-606. [Consulta: 15 enero 2022]. Disponible en:

https://www.academia.edu/14804358/Geometr%C3%ADa_y_su_did%C3%A1ctica_para_maestros.

GUACHÚN LUCERO, P. y ESPADERO FAICÁN, G., 2021. *El software GeoGebra como recurso para la enseñanza de vectores: Una experiencia didáctica. REMATEC* [en línea], vol. 16, no. 37, pp. 46-60. [Consulta: 31 agosto 2022]. ISSN 1980-3141. DOI 10.37084/rematec.1980-3141.2021.n37.p46-60.id315. Disponible en: <http://www.sinewton.org/>.

HALILI, F. y RAMADANI, E., 2018. *Web Services: A Comparison of Soap and Rest Services. Modern Applied Science*, vol. 12, no. 3, pp. 175. ISSN 1913-1844. DOI 10.5539/mas.v12n3p175.

HOHENWARTER, J., HOHENWARTER, M. y LAVICZA, Z., 2009. *Introducing dynamic mathematics software to secondary school teachers: The case of GeoGebra. ... of Computers in Mathematics ...* [en línea], vol. 28, pp. 135-146. [Consulta: 1 enero 2022]. ISSN ISSN-0731-9258. Disponible en: <http://www.editlib.org/p/30304?nl>.

IGNACIO, C., PAOLA, V. y FRUCTUOSO HERNÁNDEZ, J., 2015. *Metodologías actuales de desarrollo de software. Artículo Revista Tecnología e Innovación Diciembre* [en línea], vol. 2, no. 5, pp. 980-986. [Consulta: 18 abril 2022]. Disponible en: www.ecorfan.org/bolivia.

IKAHEIMO, M., 2022. *What is Jamstack? / Jamstack 101.* [en línea]. [Consulta: 31 agosto 2022]. Disponible en: <https://ikius.com/blog/what-is-jamstack>.

INÉS, S. y BRITO, T., 2021. *Utilización de un software como metodología alternativa para el aprendizaje del Bloque Curricular Geometría y Medida en el noveno año de educación básica de la Unidad Educativa "Isabel de Godín" durante el año lectivo 2013-2014.* [en línea], [Consulta: 15 enero 2022]. ISSN 2013-2014. Disponible en: <http://dspace.esPOCH.edu.ec/handle/123456789/14530>.

JOHANSSON, J., 2021. *Create React App vs NextJS: A comparison of two ReactJS based web application frameworks.* [en línea], [Consulta: 30 diciembre 2021]. Disponible en: <http://urn.kb.se/resolve?urn=urn:nbn:se:miun:diva-42390>.

KARAIBRYAMOV, S., TSAREVA, B. y ZLATANOV, B., 2013. *Optimization of the Courses in Geometry by the Usage of Dynamic Geometry Software Sam. The Electronic Journal of Mathematics and Technology*, vol. 7, no. 1.

MAIDA, E.G. y PACIENZA, J., 2015. *Metodología de desarrollo de Software* [en línea]. S.l.: s.n. [Consulta: 18 abril 2022]. Disponible en: <https://repositorio.uca.edu.ar/handle/123456789/522>.

- MARKOVIC, D., CICHETTI, A., BUCAIONI, A. y JAMSTACK, S.,** 2021. *UNDERSTANDING JAMSTACK AND ITS PERCEPTION IN WEB DEVELOPMENT*. [en línea], pp. 2-3. [Consulta: 25 julio 2022]. Disponible en: <https://www.diva-portal.org/smash/record.jsf?pid=diva2%3A1564626&dswid=-9184>.
- MEDVIDOVIC, N. y TAYLOR, R.N.,** 2010. *Software architecture: Foundations, theory, and practice. Proceedings - International Conference on Software Engineering*. S.l.: s.n., pp. 471-472. ISBN 9781605587196. DOI 10.1145/1810295.1810435.
- MERA PAZ, J., MIRANDA GÓMEZ, M.Y. y CUARAN ROSAS, S.,** 2017. *Análisis sistemático de información de la Norma ISO 25010 como base para la implementación en un laboratorio de Testing de software en la Universidad Cooperativa de Colombia Sede Popayán* [en línea]. 25 julio 2017. S.l.: s.n. [Consulta: 15 enero 2022]. Disponible en: <https://revistas.utp.ac.pa/index.php/memoutp/article/view/1483/html>.
- MIRANDA PALMA, C.A. y ROMERO GONZÁLEZ, R.M.,** 2019. *Un software educativo como una herramienta pedagógica en la mejora de las habilidades de lectoescritura utilizando el método ecléctico. Revista Tecnología, Ciencia y Educación* [en línea], vol. 13, pp. 172-186. [Consulta: 31 agosto 2022]. ISSN 2444-250X. DOI 10.51302/tce.2019.291. Disponible en: <https://www.tecnologia-ciencia-educacion.com/index.php/TCE/article/view/291>.
- MOREJON, S.,** 2011. *El Software Educativo un medio de enseñanza eficiente. Cuadernos de Educación y Desarrollo* [en línea]. [Consulta: 31 agosto 2022]. Disponible en: <https://ideas.repec.org/a/erv/cedced/y2011i2982.html>.
- MOTA, A.I., OLIVEIRA, H. y HENRIQUES, A.,** 2017. *El desarrollo de la capacidad de Resiliencia Matemática: La voz de los estudiantes sobre el uso de las TIC en la aula. Electronic Journal of Research in Education Psychology* [en línea], vol. 14, no. 38, pp. 67-88. [Consulta: 19 diciembre 2021]. ISSN 1696-2095. DOI 10.25115/ejrep.38.15041. Disponible en: <https://ojs.ual.es/ojs/index.php/EJREP/article/view/1677>.
- NATIONAL COUNCIL OF TEACHERS OF MATHEMATICS,** 2015. *Strategic use of technology in teaching and learning mathematics: A position of the National Council of Teachers of Mathematics. National Council of Teachers of Mathematics* [en línea]. [Consulta: 1 enero 2022]. Disponible en: <http://www.iste.org/standards/nets-for-teachers.aspx>.
- NGUYEN, A.,** 2022. *Building an E-commerce Website Using Next Js, Mantine, and Strapi*. [en línea], [Consulta: 31 agosto 2022]. Disponible en: <http://www.theseus.fi/handle/10024/751226>.
- NOGUERA, E.G. de, ALTUVE, G.N., ALEJANDRA, M. y GOTTBORG, N.,** 2011. *Propuesta de desarrollo de software para la enseñanza universitaria esta pedagógica: Una metodología. Duarte* [en línea], vol. LXI, no. 50, pp. 49-57. [Consulta: 15 enero 2022]. ISSN 0041-8935. Disponible en: <https://www.redalyc.org/articulo.oa?id=37319837005>.
- QUICHIMBO PEREIRA, G., PARDO JIMÉNEZ, K., ARCOS-MEDINA, G. y AVILA PESANTEZ, M.,** 2021. *Gestión y seguimiento de pacientes en sus dietas nutricionales utilizando*

un sitio web. *Ecuadorian Science Journal* [en línea], vol. 5, no. 2, pp. 15-30. [Consulta: 19 julio 2022]. ISSN 2602-8077. DOI 10.46480/esj.5.2.106. Disponible en: <https://journals.gdeon.org/index.php/esj/article/view/106>.

ROMERO, G., 2021. *Cómo realizar pruebas automatizadas con Postman*. [en línea]. [Consulta: 2 mayo 2022]. Disponible en: <https://www.encora.com/es/blog/como-realizar-pruebas-automatizadas-con-postman>.

RUBÉN VELASCO, 2021. *Visual Studio Code: editor de texto de código abierto para programar*. <https://www.softzone.es/programas/utilidades/visual-studio-code/> [en línea]. [Consulta: 15 enero 2022]. Disponible en: <https://www.softzone.es/programas/utilidades/visual-studio-code/>.

SCHWABER, K. y SUTHERLAND, J., 2020. *The Scrum Guide*. [en línea]. [Consulta: 31 agosto 2022]. Disponible en: <https://scrumguides.org/scrum-guide.html>.

SRIVASTAVA, A., BHARDWAJ, S. y SARASWAT, S., 2017. *SCRUM model for agile methodology. Proceeding - IEEE International Conference on Computing, Communication and Automation, ICCCA 2017*. S.l.: Institute of Electrical and Electronics Engineers Inc., pp. 864-869. ISBN 9781509064717. DOI 10.1109/CCAA.2017.8229928.

STOLS, G. y KRIEK, J., 2011. *Why don't all maths teachers use dynamic geometry software in their classrooms? Australasian Journal of Educational Technology* [en línea], vol. 27, no. 1, pp. 137-151. [Consulta: 18 diciembre 2021]. ISSN 14495554. DOI 10.14742/ajet.988. Disponible en: <https://ajet.org.au/index.php/AJET/article/view/988>.

TANNER, M., 2020. *Implementation of the new Swiss-Hema website using headless CMS and React.js. Bachelor's Thesis Degree Programme in Business Information Technology*,

TICANTE HERNÁNDEZ, A.C., HERRERA ORDUÑA, C.M., ARGUIJO, P., MELÉNDEZ ARMENTA, R.Á. y VÁZQUEZ LÓPEZ, A.H., 2019. *Videojuego educativo para ayudar a comprender los principios básicos de la programación y desarrollar la habilidad lógica en niños de educación básica. Research in Computing Science*, vol. 148, no. 7, pp. 127-139. ISSN 1870-4069. DOI 10.13053/rcs-148-7-10.

TRAMULLAS, J., 2020. *Elaboración de productos de información con JAMstack: del sistema de gestión de contenidos al web estático. Anuario ThinkEPI* [en línea], vol. 14. [Consulta: 31 agosto 2022]. ISSN 2564-8837. DOI 10.3145/THINKEPI.2020.E14F05. Disponible en: <https://thinkepi.profesionaldelainformacion.com/index.php/ThinkEPI/article/view/thinkepi.2020.e14f05>.

UNIVERSIA, 2020. *¿Cuáles son los tipos de aprendizaje?*. [en línea]. [Consulta: 2 mayo 2022]. Disponible en: <https://www.universia.net/mx/actualidad/vida-universitaria/cuales-son-tipos-aprendizaje-aqui-te-lo-desvelamos-1143835.html>.

URBINA RAMÍREZ, S., 1999. *Informática y teorías del aprendizaje*. [en línea], [Consulta: 15 enero 2022]. ISSN 1133-8482. Disponible en: <https://idus.us.es/handle/11441/45480>.

VEGA, N., FLORES-JIMÉNEZ, R., FLORES-JIMÉNEZ, I., HURTADO-VEGA, B. y RODRÍGUEZ-MARTÍNEZ, J.S., 2019. *Teorías del aprendizaje. XIKUA Boletín Científico de la Escuela Superior de Tlahuelilpan* [en línea], vol. 7, no. 14, pp. 51-53. [Consulta: 30 agosto 2022]. ISSN 2007-4948. DOI 10.29057/xikua.v7i14.4359. Disponible en: <https://repository.uaeh.edu.mx/revistas/index.php/xikua/article/view/4359>.

YIN, A., FIGUEIREDO, S. y SILVA, M.M. da, 2011. *Scrum Maturity Model: Validation for IT organizations' roadmap to develop software centered on the client role. Proceedings of the Sixth International Conference on Software Engineering Advances (ICSEA 2011), Barcelona, Spain, 23-29 October 23, 2011* [en línea], no. January 2011, pp. 20-29. [Consulta: 3 mayo 2022]. Disponible en: https://www.academia.edu/1185850/Scrum_Maturity_Model_Validation_for_IT_organizations_roadmap_to_develop_software_centered_on_the_client_role.

ZAMBRANO DAVID, 2015. *Elaboración de un software educativo como apoyo para la enseñanza de la asignatura de matemática en la Escuela de Ingeniería Ambiental de la Universidad Estatal Amazónica.* [en línea], [Consulta: 25 abril 2022]. Disponible en: <http://dspace.esPOCH.edu.ec/handle/123456789/4473>.

ZAPATA-ROS, M., 2015. *Teorías y modelos sobre el aprendizaje en entornos conectados y ubicuos. Bases para un nuevo modelo teórico a partir de una visión crítica del “conectivismo”.* *Education in the Knowledge Society (EKS)*, vol. 16, no. 1, pp. 69-102. DOI 10.14201/eks201516169102.

ANEXOS

ANEXO A: Análisis de factibilidad

Estimaciones

Tabla 1: Archivos (ILF, Internal Logical Files)

ID	Fichero Lógico Interno	Complejidad	PF
1	Usuario	Simple	5
2	Clase	Simple	5
3	Pregunta	Simple	5
4	Prueba	Simple	5
5	Calificación por clase	Simple	5
6	Calificación final	Simple	5
7	Clase completada	Simple	5
TOTAL			35

Realizado por: Silva, J., 2022

Tabla 2: Entradas (EI, External Inputs)

ID	Entrada	Función	Complejidad	PF
1	El software educativo permitirá ingresar los datos de una clase	Ingresar	Media	5
2	El software educativo permitirá ingresar los datos de una categoría	Ingresar	Simple	3
3	El software educativo permitirá ingresar los datos de una prueba	Ingresar	Media	5
4	El software educativo permitirá ingresar los datos de una pregunta	Ingresar	Media	5
5	El software educativo permitirá ingresar los datos de un estudiante	Ingresar	Media	5
6	El software educativo permitirá ingresar los datos de una calificación de cada clase	Ingresar	Baja	3
7	El software educativo permitirá ingresar los datos de una calificación final de las clases	Ingresar	Baja	3
8	El software educativo permitirá ingresar los datos de una clase completada	Ingresar	Baja	3
9	El software educativo permitirá modificar los datos de una clase	Modificar	Media	5

10	El software educativo permitirá modificar los datos de una categoría	Modificar	Simple	3
11	El software educativo permitirá modificar los datos de una prueba	Modificar	Media	5
12	El software educativo permitirá modificar los datos de una pregunta	Modificar	Media	5
13	El software educativo permitirá modificar los datos de un estudiante	Modificar	Media	5
14	El software educativo permitirá modificar los datos de una calificación de cada clase	Modificar	Baja	3
15	El software educativo permitirá modificar los datos de una calificación final de las clases	Modificar	Baja	3
16	El software educativo permitirá modificar los datos de una clase completada	Modificar	Baja	3
17	El software educativo permitirá eliminar los datos de una clase	Eliminar	Baja	3
18	El software educativo permitirá eliminar los datos de una categoría	Eliminar	Baja	3
19	El software educativo permitirá eliminar los datos de una prueba	Eliminar	Baja	3
20	El software educativo permitirá eliminar los datos de una pregunta	Eliminar	Baja	3
21	El software educativo permitirá eliminar los datos de un estudiante	Eliminar	Baja	3
TOTAL				79

Realizado por: Silva, J., 2022

Tabla 3: Salidas (EO, External Outputs)

ID	Salida	Función	Complejidad	PF
1	El software educativo permitirá visualizar el listado de clases	Pantalla	Simple	3
2	El software educativo permitirá visualizar el listado de categorías	Pantalla	Simple	3
3	El software educativo permitirá visualizar el listado de pruebas	Pantalla	Simple	3
4	El software educativo permitirá visualizar el listado de preguntas	Pantalla	Simple	3
5	El software educativo visualizar el listado de estudiantes	Pantalla	Simple	3

6	El software educativo permitirá visualizar las calificaciones por clases	Pantalla / Reporte	Media	5
7	El software educativo permitirá visualizar las calificaciones finales de los estudiantes	Pantalla / Reporte	Media	5
8	El software educativo permitirá visualizar gráficas estadísticas del rendimiento de los estudiantes	Pantalla / Reporte	Media	5
TOTAL				35

Realizado por: Silva, J., 2022

Tabla 4: Consultas (EQ, External Inquiries)

ID	Consulta	Función	Complejidad	PF
1	El software educativo permitirá mostrar las clases de una categoría.	Pantalla	Media	5
2	El software educativo permitirá mostrar las calificaciones de un estudiante.	Pantalla	Simple	3
3	El software educativo permitirá mostrar los datos de un estudiante.	Pantalla	Simple	3
TOTAL				11

Realizado por: Silva, J., 2022

Total de puntos de fusión

Tabla 5: Total de puntos de fusión

Tipo	PF
Archivos	35
Entradas	79
Salidas	35
Consultas	11
TOTAL	160

Realizado por: Silva, J., 2022

Para el cálculo de las líneas de código se utiliza en los puntos de fusión totales y un promedio de líneas de código (20) para un punto de fusión en el lenguaje de programación JavaScript, dando como resultado 4640 líneas de código.

$$KLOC = PF * SLOC = \frac{160 * 20}{1000} = 4.64$$

Estimación mediante COCOMO

Para este proyecto se utiliza el modelo orgánico de COCOMO, por lo tanto, se tienen los siguientes coeficientes:

$$C1 = 2.40, C2 = 1.05, C3=0.38$$

Esfuerzo Hombre – Mes

$$H - M = c1 * KLOC^{c2}$$
$$H - M = 2.4 * 4.63^{1.05} = 11.99$$

Tiempo de desarrollo

$$Tdev = 2.5 * (H - M)^{c3}$$
$$Tdev = 2.5 * (11.99)^{0.38} = 6.42$$

Número de programadores

$$Nprog = \frac{H - M}{Tdev}$$
$$Nprog = \frac{11.99}{6.42} = 1.82$$

Como resultados de la aplicación de COCOMO se tiene que el esfuerzo Hombre-Mes es de 11.99, con un tiempo estimado de desarrollo de 6.42 meses y un total de 2 programadores, es necesario realizar ajustes para que el proyecto sea factible.

ANEXO B: Gestión de riesgos

1. Criterios para la valoración de la probabilidad e impacto de los riesgos

Después de identificar los riesgos es pertinente definir criterios para la para la valoración de los riesgos en base a su probabilidad de impacto. En la Tabla 1 se detalla los criterios para la evaluación de la probabilidad de los posibles riesgos identificados.

Tabla 1: Valoración de probabilidad de riesgos

Rango (Probabilidad)	Descripción	Valor asignado
1 % al 33%	Baja	1
34 % al 66%	Media	2
67 % al 100%	Alta	3

Realizado por: Silva, J., 2022

En la Tabla 2 se detalla los criterios para la evaluación del impacto de los posibles riesgos identificados.

Tabla 2: Valoración del impacto de los riesgos

Descripción	Valor asignado
Bajo	1
Medio	2
Alto	3
Crítico	4

Realizado por: Silva, J., 2022

Tabla 3: Valoración de la exposición del riesgo

Descripción	Valor asignado (probabilidad * impacto)
Bajo	1
Medio	4
Alto	9
Crítico	16

Realizado por: Silva, J., 2022

2. Análisis de los riesgos

En la Tabla 4 se realiza un análisis de los riesgos, tomando en cuenta su probabilidad de ocurrencia, el impacto que tendrían en el desarrollo del software educativo y de esta manera se obtuvo el valor de la exposición de estos riesgos durante el proyecto.

Tabla 4: Análisis de riesgos

ID	Descripción	Probabilidad		Impacto		Exposición	
		Porcentaje	Valor	Porcentaje	Valor	Porcentaje	Valor
RI-01	Cambios no viables de requerimientos	Alta (70%)	3	Alta	3	Alta	3
RI-02	Falta de partición del cliente	Baja (15%)	1	Medio	2	Media	2
RI-03	Asignación incorrecta de recursos	Media (50%)	2	Media	2	Media	4
RI-04	Daño de equipos	Baja (15%)	1	Baja	1	Baja	1
RI-05	Herramientas de desarrollo incompatibles	Baja (15%)	1	Baja	1	Baja	1
RI-06	Diseño erróneo de base de datos	Media (15%)	2	Media	2	Media	4

Realizado por: Silva, J., 2022

3. Priorización de los riesgos

Con el análisis de riesgos realizado se procede a la asignación de prioridades de cada uno, con el fin de considerar los que tienen mayor prioridad y que deberán ser gestionados antes de convertirse en amenazas para el proyecto. Como se observa en la Tabla 5, el riesgo con mayor prioridad es el de exposición alta (RI-01) y será el primero en gestionarse, después de existen 3 riesgos con exposición media (RI-01, RI-02, RI-06) que se gestionarán en la fase inicial del proyecto y 2 riesgos con exposición baja (RI-04, RI-05) que se gestionarán a medida que avance el proyecto.

Tabla 5: Priorización de riesgos

ID	Descripción	Exposición	Valor	Prioridad
RI-01	Cambios no viables de requerimientos	Alta	9	1
RI-02	Falta de partición del cliente	Media	4	2
RI-03	Asignación incorrecta de recursos	Media	4	3
RI-06	Diseño erróneo de base de datos	Media	4	4
RI-05	Herramientas de desarrollo incompatibles	Baja	1	5
RI-04	Daño de equipos	Baja	1	6

Realizado por: Silva, J., 2022

ANEXO C: Planificación del proyecto

ESPOCH | Trabajo de Titulación

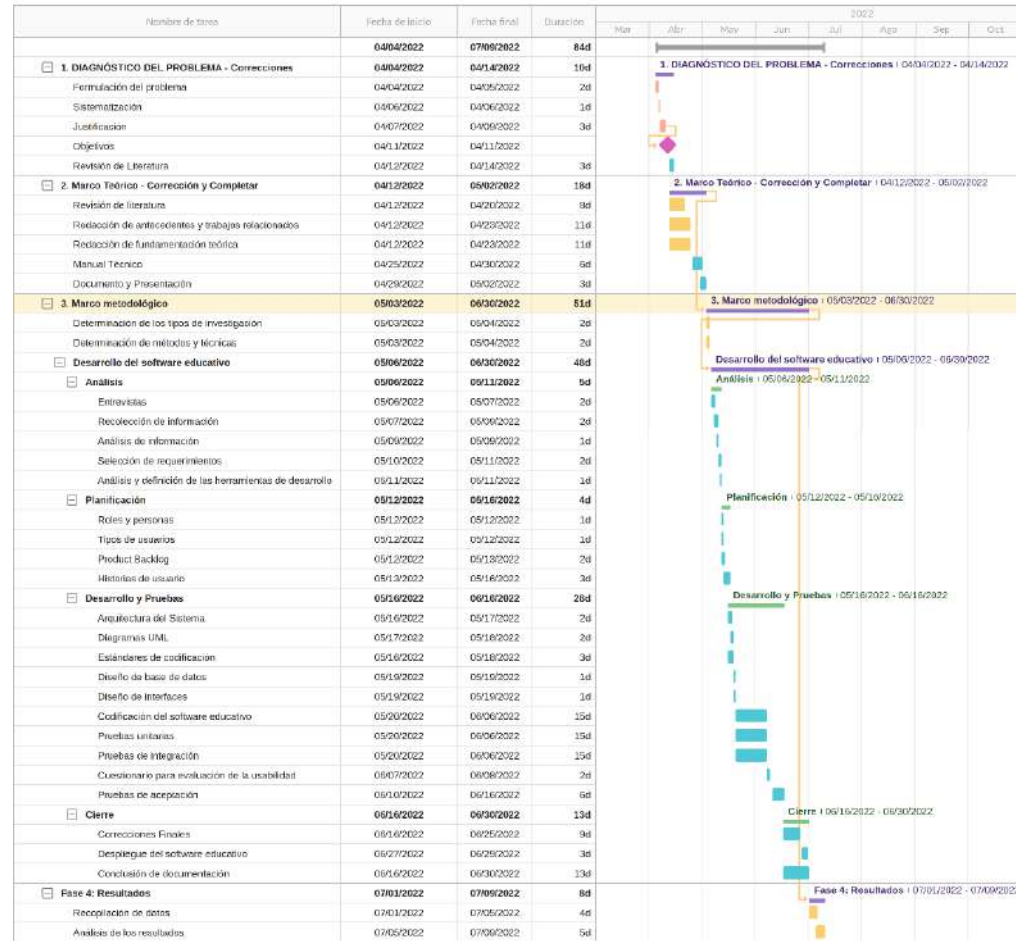


Figura 1: Planificación del Proyecto

Realizado por: Silva, J., 2022

ANEXO D: Historias técnicas

HISTORIA TÉCNICA			
ID	HT-01	Nombre de la Historia	Definir los requerimientos del software educativo
Rol	Desarrollador		Sprint
Prioridad	Alta		Puntos Estimados 48
Riesgo			Puntos Reales 48
Fecha (Inicio)	06/05/2022		Fecha (Fin) 11/05/2022
Descripción	Como desarrollador requiero entrevistas con el cliente para recolectar información, analizarla y definir los requerimientos del software educativo		
PRUEBAS DE ACEPTACIÓN			
ID_PA	Descripción		Estado
HT01 – PA01	Analizar el nivel de cumplimiento de los objetivos con los requerimientos definidos		Finalizada
TAREAS DE INGENIERÍA			
ID_TI	Descripción		Esfuerzo
HT01 – TI01	Realizar las reuniones con el cliente hasta que se cumpla con el nivel de cumplimiento de los objetivos con los requerimientos definidos		24
HT01 – TI02	Documentar los requerimientos definidos en base a la metodología SCRUM		24

Realizado por: Silva, J., 2022

HISTORIA TÉCNICA			
ID	HT-02	Nombre de la Historia	Definir la arquitectura del software educativo
Rol	Desarrollador		Sprint
Prioridad	Alta		Puntos Estimados 32
Riesgo			Puntos Reales 32
Fecha (Inicio)	16/05/2022		Fecha (Fin) 17/05/2022
Descripción	Como desarrollador requiero definir la arquitectura del software educativo para una correcta interacción entre todos los módulos que lo conforman		
PRUEBAS DE ACEPTACIÓN			
ID_PA	Descripción		Estado
HT02 – PA01	Analizar los diseños de arquitectura para aplicaciones web		Finalizada
HT02 – PA02	Verificar la correcta interacción de cada módulo del software educativo		-
TAREAS DE INGENIERÍA			
ID_TI	Descripción		Esfuerzo
HT02 – TI01	Implementación de la arquitectura cliente servidor		16
HT02 – TI02	Documentación de la arquitectura del software educativo a través de Diagramas UML		16

Realizado por: Silva, J., 2022

HISTORIA TÉCNICA			
ID	HT-03	Nombre de la Historia	Definir el estándar de codificación
Rol	Desarrollador	Sprint	
Prioridad	Alta	Puntos Estimados	8
Riesgo		Puntos Reales	8
Fecha (Inicio)	17/05/2022	Fecha (Fin)	18/05/2022
Descripción	Como desarrollador requiero definir un estándar de codificación para mejorar la seguridad, protección y homogeneidad del código		
PRUEBAS DE ACEPTACIÓN			
ID_PA	Descripción		Estado
HT03 – PA01	Determinar un estándar de codificación adecuado para el software educativo		Finalizada
TAREAS DE INGENIERÍA			
ID_TI	Descripción		Esfuerzo
HT03 – TI01	Seleccionar herramientas que faciliten la implementación del estándar de codificación seleccionado		8

Realizado por: Silva, J., 2022

HISTORIA TÉCNICA			
ID	HT-04	Nombre de la Historia	Diseño de la base de datos
Rol	Desarrollador	Sprint	
Prioridad	Alta	Puntos Estimados	32
Riesgo		Puntos Reales	32
Fecha (Inicio)	19/05/2022	Fecha (Fin)	19/05/2022
Descripción	Como desarrollador requiero diseñar y crear la base de datos correspondiente para el desarrollo del software educativo		
PRUEBAS DE ACEPTACIÓN			
ID_PA	Descripción		Estado
HT04 – PA01	Verificar la normalización y las relaciones entre entidades de la base de datos		Finalizada
HT04 – PA01	Verificar las colecciones en Strapi		
TAREAS DE INGENIERÍA			
ID_TI	Descripción		Esfuerzo
HT04 – TI01	Realizar el diseño lógico y conceptual de la base de datos		16
HT04 – TI02	Realizar el diccionario de datos		16

Realizado por: Silva, J., 2022

HISTORIA TÉCNICA			
ID	HT-05	Nombre de la Historia	Diseñar las interfaces del software educativo
Rol	Desarrollador	Sprint	

Prioridad	Alta	Puntos Estimados	32
Riesgo		Puntos Reales	32
Fecha (Inicio)	19/05/2022	Fecha (Fin)	20/05/2022
Descripción	Como desarrollador requiero diseñar y prototipar las interfaces del software educativo		
PRUEBAS DE ACEPTACIÓN			
ID_PA	Descripción	Estado	
HT05 – PA01	Determinar la paleta de colores para el software educativo	Finalizada	
HT05 – PA02	Determinar estructura de la información del software educativo	Finalizada	
TAREAS DE INGENIERÍA			
ID_TI	Descripción	Esfuerzo	
HT05 – TI01	Creación del Sistema de Diseño (Design System) para el software educativo	16	
HT05 – TI02	Realizar mockups de las interfaces principales	16	

Realizado por: Silva, J., 2022

ANEXO E: Historias de usuario

HISTORIA DE USUARIO			
ID	HU-01	Nombre de la Historia	Registro de usuarios
Usuario	Administrador, maestro, estudiante		Sprint
Prioridad	Alta		Puntos Estimados 40
Riesgo			Puntos Reales 40
Fecha (Inicio)	20/05/2022		Fecha (Fin) 22/05/2022
Descripción	Como usuario quiero registrarme en el software educativo para poder acceder a los contenidos		
PRUEBAS DE ACEPTACIÓN			
ID_PA	Descripción		Estado
HU01 – PA01	Validación de campos requeridos para el registro		Finalizada
HU01 – PA02	Verificar que el usuario sea único		-
HU01 – PA03	Notificación y verificación a través de correo electrónico		-
HU01 – PA04	Verificar almacenamiento de usuarios en Strapi		Finalizada
TAREAS DE INGENIERÍA			
ID_TI	Descripción		Esfuerzo
HU01 – TI01	Desarrollar la interfaz de registro e inicio de sesión		8
HU01 – TI02	Implementar el servicio web para la comunicación del software educativo con Strapi		32

Realizado por: Silva, J., 2022

HISTORIA DE USUARIO			
ID	HU-02	Nombre de la Historia	Autenticación de usuarios
Usuario	Administrador, maestro, estudiante		Sprint
Prioridad	Alta		Puntos Estimados 16
Riesgo			Puntos Reales 16
Fecha (Inicio)	23/05/2022		Fecha (Fin) 23/05/2022
Descripción	Como usuario quiero autenticarme en el software educativo para poder acceder a los contenidos		
PRUEBAS DE ACEPTACIÓN			
ID_PA	Descripción		Estado
HU02 – PA01	Ingreso al software educativo con las funcionalidades dependiendo del rol		Finalizada
HU02 – PA02	Notificación de error de inicio de sesión		Finalizada
TAREAS DE INGENIERÍA			
ID_TI	Descripción		Esfuerzo
HU02 – TI01	Implementar el servicio para la autenticación de los usuarios en el software educativo		16

Realizado por: Silva, J., 2022

HISTORIA DE USUARIO			
ID	HU-03	Nombre de la Historia	Restablecer contraseña mediante correo electrónico

Usuario	Administrador, maestro, estudiante	Sprint	
Prioridad	Media	Puntos Estimados	16
Riesgo		Puntos Reales	16
Fecha (Inicio)	24/05/2022	Fecha (Fin)	24/05/2022
Descripción	Como usuario quiero restablecer la contraseña mediante correo electrónico en caso de pérdida		
PRUEBAS DE ACEPTACIÓN			
ID_PA	Descripción	Estado	
HU03 – PA01	Verificar que el usuario esté registrado	Finalizada	
HU03 – PA02	Enviar enlace de recuperación a correo electrónico	Finalizada	
TAREAS DE INGENIERÍA			
ID_TI	Descripción	Esfuerzo	
HU03 – TI01	Desarrollo de la interfaz de recuperación de contraseña	4	
HU03 – TI02	Verificación de existencia de usuario en la base de datos	4	
HU03 – TI03	Envío de enlace de recuperación al correo electrónico	8	

Realizado por: Silva, J., 2022

HISTORIA DE USUARIO			
ID	HU-04	Nombre de la Historia	Crear, modifica eliminar y visualizar las clases
Usuario	Maestro	Sprint	
Prioridad	Alta	Puntos Estimados	8
Riesgo		Puntos Reales	8
Fecha (Inicio)	25/05/2022	Fecha (Fin)	25/05/2022
Descripción	Como maestro quiero crear, modificar, eliminar y visualizar las clases		
PRUEBAS DE ACEPTACIÓN			
ID_PA	Descripción	Estado	
HU04 – PA01	Gestionar la información de las clases en Strapi	Finalizada	
TAREAS DE INGENIERÍA			
ID_TI	Descripción	Esfuerzo	
HU04 – TI01	Creación de la colección “clases”	4	
HU04 – TI02	Configuración de permisos de la colección	4	

Realizado por: Silva, J., 2022

HISTORIA DE USUARIO			
ID	HU-05	Nombre de la Historia	Crear, modifica eliminar y visualizar las categorías
Usuario	Maestro	Sprint	
Prioridad	Alta	Puntos Estimados	8
Riesgo		Puntos Reales	8
Fecha (Inicio)	25/05/2022	Fecha (Fin)	25/05/2022
Descripción	Como maestro quiero crear, modificar, eliminar y visualizar las categorías		
PRUEBAS DE ACEPTACIÓN			
ID_PA	Descripción	Estado	
HU05 – PA01	Gestionar la información de las categorías en Strapi	Finalizada	

TAREAS DE INGENIERÍA		
ID_TI	Descripción	Esfuerzo
HU05 – TI01	Creación de la colección “categorías”	4
HU05 – TI02	Configuración de permisos de la colección	4

Realizado por: Silva, J., 2022

HISTORIA DE USUARIO			
ID	HU-06	Nombre de la Historia	Crear, modifica eliminar y visualizar las pruebas
Usuario	Maestro	Sprint	
Prioridad	Alta	Puntos Estimados	8
Riesgo		Puntos Reales	8
Fecha (Inicio)	25/05/2022	Fecha (Fin)	25/05/2022
Descripción	Como maestro quiero crear, modificar, eliminar y visualizar las pruebas		

PRUEBAS DE ACEPTACIÓN

ID_PA	Descripción	Estado
HU06 – PA01	Gestionar la información de las pruebas en Strapi	Finalizada

TAREAS DE INGENIERÍA

ID_TI	Descripción	Esfuerzo
HU06 – TI01	Creación de la colección “pruebas”	4
HU06 – TI02	Configuración de permisos de la colección	4

Realizado por: Silva, J., 2022

HISTORIA DE USUARIO			
ID	HU-07	Nombre de la Historia	Crear, modifica eliminar y visualizar las preguntas
Usuario	Maestro	Sprint	
Prioridad	Alta	Puntos Estimados	8
Riesgo		Puntos Reales	8
Fecha (Inicio)	25/05/2022	Fecha (Fin)	25/05/2022
Descripción	Como maestro quiero crear, modificar, eliminar y visualizar las preguntas		

PRUEBAS DE ACEPTACIÓN

ID_PA	Descripción	Estado
HU07 – PA01	Gestionar la información de las preguntas en Strapi	Finalizada

TAREAS DE INGENIERÍA

ID_TI	Descripción	Esfuerzo
HU07 – TI01	Creación de la colección “preguntas”	4
HU07 – TI02	Configuración de permisos de la colección	4

Realizado por: Silva, J., 2022

HISTORIA DE USUARIO			
ID	HU-08	Nombre de la Historia	Enlazar temas a una categoría
Usuario	Maestro	Sprint	
Prioridad	Alta	Puntos Estimados	12
Riesgo		Puntos Reales	12

Fecha (Inicio)	25/05/2022	Fecha (Fin)	25/05/2022
Descripción	Como maestro quiero seleccionar los temas que pertenecen a una determinada categoría		
PRUEBAS DE ACEPTACIÓN			
ID_PA	Descripción	Estado	
HU08 – PA01	Verificar que los temas se encuentren en determinada categoría	Finalizada	
TAREAS DE INGENIERÍA			
ID_TI	Descripción	Esfuerzo	
HU08 – TI01	Creación de la colección “categorías”	4	
HU08 – TI02	Relación entre clases y categorías	4	
HU08 – TI03	Configuración de permisos de la colección	4	

Realizado por: Silva, J., 2022

HISTORIA DE USUARIO			
ID	HU-09	Nombre de la Historia	Enlazar preguntas a una prueba
Usuario	Maestro	Sprint	
Prioridad	Alta	Puntos Estimados	12
Riesgo		Puntos Reales	12
Fecha (Inicio)	25/05/2022	Fecha (Fin)	25/05/2022
Descripción	Como maestro quiero seleccionar las preguntas que pertenecen a una determinada prueba		
PRUEBAS DE ACEPTACIÓN			
ID_PA	Descripción	Estado	
HU09 – PA01	Verificar que los preguntas se encuentren en determinada prueba	Finalizada	
TAREAS DE INGENIERÍA			
ID_TI	Descripción	Esfuerzo	
HU09 – TI01	Creación de la colección “preguntas”	4	
HU09 – TI02	Relación entre pruebas y preguntas	4	
HU09 – TI03	Configuración de permisos de la colección	4	

Realizado por: Silva, J., 2022

HISTORIA DE USUARIO			
ID	HU-10	Nombre de la Historia	Enlazar prueba a una clase
Usuario	Maestro	Sprint	
Prioridad	Alta	Puntos Estimados	8
Riesgo		Puntos Reales	8
Fecha (Inicio)	25/05/2022	Fecha (Fin)	25/05/2022
Descripción	Como maestro quiero enlazar una prueba a una clase		
PRUEBAS DE ACEPTACIÓN			
ID_PA	Descripción	Estado	
HU10 – PA01	Verificar que una prueba se encuentre en determinada clase	Finalizada	
TAREAS DE INGENIERÍA			
ID_TI	Descripción	Esfuerzo	
HU10 – TI01	Relación entre pruebas y clases	4	
HU10 – TI03	Configuración de permisos de la colección	4	

Realizado por: Silva, J., 2022

HISTORIA DE USUARIO			
ID	HU-11	Nombre de la Historia	Modificar información personal del estudiante
Usuario	Estudiante	Sprint	
Prioridad	Media	Puntos Estimados	12
Riesgo		Puntos Reales	12
Fecha (Inicio)	26/05/2022	Fecha (Fin)	26/05/2022
Descripción	Como estudiante quiero modificar la información personal		
PRUEBAS DE ACEPTACIÓN			
ID_PA	Descripción		Estado
HU11 – PA01	Verificar que la información modificada se actualice en Strapi y el software educativo		Finalizada
TAREAS DE INGENIERÍA			
ID_TI	Descripción		Esfuerzo
HU11 – TI01	Validación de token del estudiante		4
HU11 – TI02	Creación del servicio para modificar la información		4
HU11 – TI03	Creación de la interfaz para modificar la información		4

Realizado por: Silva, J., 2022

HISTORIA DE USUARIO			
ID	HU-12	Nombre de la Historia	Visualizar listado de clases
Usuario	Estudiante	Sprint	
Prioridad	Alta	Puntos Estimados	24
Riesgo		Puntos Reales	24
Fecha (Inicio)	27/05/2022	Fecha (Fin)	27/05/2022
Descripción	Como estudiante quiero visualizar las todas las clases		
PRUEBAS DE ACEPTACIÓN			
ID_PA	Descripción		Estado
HU12 – PA01	Verificar que al iniciar sesión el estudiante vio un listado de todas las clases		Finalizada
TAREAS DE INGENIERÍA			
ID_TI	Descripción		Esfuerzo
HU12 – TI01	Desarrollar la interfaz para las clases		8
HU12 – TI02	Creación del servicio para obtener el listado de todas las clases		8
HU12 – TI03	Agrupar las clases según la categoría		8

Realizado por: Silva, J., 2022

HISTORIA DE USUARIO			
ID	HU-13	Nombre de la Historia	Visualizar una introducción sobre la clase

Usuario	Estudiante	Sprint	
Prioridad	Alta	Puntos Estimados	16
Riesgo		Puntos Reales	16
Fecha (Inicio)	27/05/2022	Fecha (Fin)	28/05/2022
Descripción	Como estudiante quiero que al ingresar a una clase se muestre una información introductoria de esta		
PRUEBAS DE ACEPTACIÓN			
ID_PA	Descripción		Estado
HU13 – PA01	Al acceder a una clase se muestra una sección introductoria		Finalizada
TAREAS DE INGENIERÍA			
ID_TI	Descripción		Esfuerzo
HU13 – TI01	Desarrollar la interfaz para visualizar la introducción de la clase		8
HU13 – TI02	Creación del servicio para obtener la información de la clase		8

Realizado por: Silva, J., 2022

HISTORIA DE USUARIO			
ID	HU-14	Nombre de la Historia	Visualizar contenido multimedia de la clase
Usuario	Estudiante	Sprint	
Prioridad	Alta	Puntos Estimados	16
Riesgo		Puntos Reales	16
Fecha (Inicio)	29/05/2022	Fecha (Fin)	29/05/2022
Descripción	Como estudiante quiero que al ingresar a una clase se muestre un video sobre la clase		
PRUEBAS DE ACEPTACIÓN			
ID_PA	Descripción		Estado
HU14 – PA01	Al acceder a una clase se muestra un video sobre la clase		Finalizada
TAREAS DE INGENIERÍA			
ID_TI	Descripción		Esfuerzo
HU14 – TI01	Desarrollar la interfaz para visualizar un video sobre de la clase		16

Realizado por: Silva, J., 2022

HISTORIA DE USUARIO			
ID	HU-15	Nombre de la Historia	Visualizar e interactuar con un modelo geométrico sobre la clase
Usuario	Estudiante	Sprint	
Prioridad	Alta	Puntos Estimados	82
Riesgo		Puntos Reales	82
Fecha (Inicio)	29/05/2022	Fecha (Fin)	03/06/2022
Descripción	Como estudiante quiero observar un modelo geométrico sobre la clase y poder interactuar con este		
PRUEBAS DE ACEPTACIÓN			
ID_PA	Descripción		Estado
HU15 – PA01	Al acceder a una clase se muestra un modelo geométrico interactivo		Finalizada
TAREAS DE INGENIERÍA			

ID_TI	Descripción	Esfuerzo
HU15 – TI01	Desarrollar la interfaz para visualizar el modelo geométrico sobre de la clase	16
HU15 – TI02	Desarrollar el servicio de comunicación del modelo con el software educativo	16
HU15 – TI03	Crear el modelo geométrico de cada clase	40

Realizado por: Silva, J., 2022

HISTORIA DE USUARIO			
ID	HU-16	Nombre de la Historia	Realizar una evaluación sobre la clase
Usuario	Estudiante	Sprint	
Prioridad	Alta	Puntos Estimados	56
Riesgo		Puntos Reales	56
Fecha (Inicio)	03/06/2022	Fecha (Fin)	04/06/2022
Descripción	Como estudiante quiero realizar una evaluación sobre la clase		
PRUEBAS DE ACEPTACIÓN			
ID_PA	Descripción		Estado
HU16 – PA01	Al acceder a una clase se muestra una evaluación sobre la clase		Finalizada
HU16 – PA02	Las calificaciones son enviadas a Strapi		Finalizada
TAREAS DE INGENIERÍA			
ID_TI	Descripción		Esfuerzo
HU16 – TI01	Desarrollar la interfaz para la evaluación de cada clase		16
HU16 – TI02	Desarrollar el servicio para obtener la prueba con sus preguntas de cada clase		16
HU16 – TI03	Desarrollar el módulo de tiempo para la prueba		8
HU16 – TI04	Desarrollar el servicio para enviar la calificación obtenida a Strapi		16

Realizado por: Silva, J., 2022

HISTORIA DE USUARIO			
ID	HU-17	Nombre de la Historia	Obtener una puntuación mínima para finalizar la clase
Usuario	Estudiante	Sprint	
Prioridad	Baja	Puntos Estimados	4
Riesgo		Puntos Reales	4
Fecha (Inicio)	04/06/2022	Fecha (Fin)	04/06/2022
Descripción	Como estudiante quiero obtener una puntuación mínima en la prueba para de finalizar la clase		
PRUEBAS DE ACEPTACIÓN			
ID_PA	Descripción		Estado
HU17 – PA01	Verificar que tenga la calificación mínima para finalizar la clase		Finalizada
TAREAS DE INGENIERÍA			
ID_TI	Descripción		Esfuerzo
HU17 – TI01	Desarrollar la interfaz para visualizar un mensaje de puntuación mínima requerida		4

Realizado por: Silva, J., 2022

HISTORIA DE USUARIO			
---------------------	--	--	--

ID	HU-18	Nombre de la Historia	Retroalimentación de cada evaluación	
Usuario	Estudiante		Sprint	
Prioridad	Alta		Puntos Estimados	16
Riesgo			Puntos Reales	16
Fecha (Inicio)	04/06/2022		Fecha (Fin)	05/06/2022
Descripción	Como estudiante quiero ver los resultados de mi evaluación de cada clase y tener retroalimentación sobre los errores cometido			

PRUEBAS DE ACEPTACIÓN

ID_PA	Descripción	Estado
HU18 – PA01	Verificar se muestre retroalimentación en cada prueba incluida la resolución de cada pregunta	Finalizada

TAREAS DE INGENIERÍA

ID_TI	Descripción	Esfuerzo
HU18 – TI01	Desarrollar la interfaz para ver los resultados de la prueba	16

Realizado por: Silva, J., 2022

HISTORIA DE USUARIO

ID	HU-19	Nombre de la Historia	Visualizar las clases completas e incompletas	
Usuario	Estudiante		Sprint	
Prioridad	Media		Puntos Estimados	8
Riesgo			Puntos Reales	8
Fecha (Inicio)	05/06/2022		Fecha (Fin)	05/06/2022
Descripción	Como estudiante quiero distinguir entre las clases completas e incompletas			

PRUEBAS DE ACEPTACIÓN

ID_PA	Descripción	Estado
HU19 – PA01	Verificar que se muestra un indicativo de las clases completas y sin completar	Finalizada

TAREAS DE INGENIERÍA

ID_TI	Descripción	Esfuerzo
HU19 – TI01	Desarrollar el componente sobre el estado de la clase	8

Realizado por: Silva, J., 2022

HISTORIA DE USUARIO

ID	HU-20	Nombre de la Historia	Visualizar un resumen sobre las calificaciones de las clases	
Usuario	Estudiante		Sprint	
Prioridad	Alta		Puntos Estimados	32
Riesgo			Puntos Reales	32
Fecha (Inicio)	05/06/2022		Fecha (Fin)	05/06/2022
Descripción	Como estudiante quiero ver las calificaciones que obtuve en cada clase			

PRUEBAS DE ACEPTACIÓN

ID_PA	Descripción	Estado
--------------	--------------------	---------------

HU20 – PA01	Verificar que se muestran los datos de las calificaciones obtenidas en cada clase	Finalizada
TAREAS DE INGENIERÍA		
ID_TI	Descripción	Esfuerzo
HU20 – TI01	Desarrollar la interfaz de visualización de calificaciones de las clases	16
HU20 – TI02	Creación del servicio para obtener los detalles de las calificaciones de las clases	16

Realizado por: Silva, J., 2022

HISTORIA DE USUARIO			
ID	HU-21	Nombre de la Historia	Visualizar los resultados finales y el progreso
Usuario	Estudiante	Sprint	
Prioridad	Alta	Puntos Estimados	32
Riesgo		Puntos Reales	32
Fecha (Inicio)	06/06/2022	Fecha (Fin)	06/06/2022
Descripción	Como estudiante quiero ver los resultados finales y el progreso a lo largo de las clases		
PRUEBAS DE ACEPTACIÓN			
ID_PA	Descripción		Estado
HU21 – PA01	Verificar que se muestra los resultados finales y gráficas con el progreso del estudiante		Finalizada
TAREAS DE INGENIERÍA			
ID_TI	Descripción		Esfuerzo
HU21 – TI01	Desarrollar la interfaz de visualización de resultados finales y gráficas		32

Realizado por: Silva, J., 2022

HISTORIA DE USUARIO			
ID	HU-22	Nombre de la Historia	Obtener un certificado al finalizar todas las clases
Usuario	Estudiante	Sprint	
Prioridad	Alta	Puntos Estimados	16
Riesgo		Puntos Reales	16
Fecha (Inicio)	06/06/2022	Fecha (Fin)	06/06/2022
Descripción	Como estudiante quiero obtener un certificado al concluir todas las clases		
PRUEBAS DE ACEPTACIÓN			
ID_PA	Descripción		Estado
HU22 – PA01	Verificar que se presente un certificado de finalización		Finalizada
HU22 – PA02	Verificar que sólo al completar todas las clases se pueda visualizar el certificado		Finalizada
TAREAS DE INGENIERÍA			
ID_TI	Descripción		Esfuerzo
HU22 – TI01	Desarrollar la interfaz del certificado de finalización		8
HU22 – TI02	Comprobar que el estudiante haya finalizado todas las clases		8

Realizado por: Silva, J., 2022

HISTORIA DE USUARIO			
---------------------	--	--	--

ID	HU-23	Nombre de la Historia	Visualizar listado de estudiantes	
Usuario	Maestro		Sprint	
Prioridad	Alta		Puntos Estimados	12
Riesgo			Puntos Reales	12
Fecha (Inicio)	06/06/2022		Fecha (Fin)	06/06/2022
Descripción	Como maestro quiero visualizar el listado de los estudiantes y su progreso			
PRUEBAS DE ACEPTACIÓN				
ID_PA	Descripción			Estado
HU23 – PA01	Verificar que solo el maestro puede listar los estudiantes			Finalizada
HU23 – PA02	Verificar que se visualiza listado de estudiantes y su progreso			Finalizada
TAREAS DE INGENIERÍA				
ID_TI	Descripción			Esfuerzo
HU23 – TI01	Desarrollar la interfaz del listado de estudiantes y el indicador de progreso en las clases			4
HU23 – TI02	Visualización de información dependiendo del rol “maestro”			4
HU23 – TI03	Creación del servicio para obtener el listado de estudiantes y su progreso			4

Realizado por: Silva, J., 2022

HISTORIA DE USUARIO				
ID	HU-24	Nombre de la Historia	Visualizar calificaciones por clase	
Usuario	Maestro		Sprint	
Prioridad	Alta		Puntos Estimados	8
Riesgo			Puntos Reales	8
Fecha (Inicio)	07/06/2022		Fecha (Fin)	07/06/2022
Descripción	Como maestro quiero visualizar las calificaciones agrupadas por clase			
PRUEBAS DE ACEPTACIÓN				
ID_PA	Descripción			Estado
HU24 – PA01	Verificar que se muestren las calificaciones dependiendo de cada clase			Finalizada
TAREAS DE INGENIERÍA				
ID_TI	Descripción			Esfuerzo
HU24 – TI01	Desarrollar la interfaz del listado de calificaciones por clase			4
HU24 – TI02	Creación del servicio para obtener el listado de clases y sus calificaciones			4

Realizado por: Silva, J., 2022

HISTORIA DE USUARIO				
ID	HU-25	Nombre de la Historia	Visualizar calificaciones generales	
Usuario	Maestro		Sprint	
Prioridad	Alta		Puntos Estimados	32
Riesgo			Puntos Reales	32
Fecha (Inicio)	07/06/2022		Fecha (Fin)	07/06/2022

Descripción	Como maestro quiero visualizar las calificaciones de todas las clases en un listado general de estudiantes	
PRUEBAS DE ACEPTACIÓN		
ID_PA	Descripción	Estado
HU25 – PA01	Verificar que se muestren las calificaciones de todos los estudiantes	Finalizada
TAREAS DE INGENIERÍA		
ID_TI	Descripción	Esfuerzo
HU25 – TI01	Desarrollar la interfaz del listado de calificaciones generales	16
HU25 – TI02	Creación del servicio para obtener el listado de estudiantes y calificaciones	16

Realizado por: Silva, J., 2022

HISTORIA DE USUARIO			
ID	HU-26	Nombre de la Historia	Gráficas de calificaciones
Usuario	Maestro	Sprint	
Prioridad	Alta	Puntos Estimados	32
Riesgo		Puntos Reales	32
Fecha (Inicio)	07/06/2022	Fecha (Fin)	07/06/2022
Descripción	Como maestro quiero visualizar el progreso y las calificaciones de los estudiantes a través de gráficas estadísticas		
PRUEBAS DE ACEPTACIÓN			
ID_PA	Descripción		Estado
HU26 – PA01	Verificar que se muestran gráficas de estadísticas con la información sobre el progreso y las calificaciones de los estudiantes		Finalizada
TAREAS DE INGENIERÍA			
ID_TI	Descripción		Esfuerzo
HU26 – TI01	Desarrollar la interfaz para visualizar las gráficas		16
HU26 – TI02	Creación del servicio para analizar y graficar el progreso y las calificaciones de los estudiantes		16

Realizado por: Silva, J., 2022

ANEXO F: Estándares de codificación

Especificaciones de codificación en NextJS

1. Archivos

- Cada componente se encuentra en una carpeta, cada archivo contiene solo un componente, el nombre del archivo y componente será el de la carpeta.
- La exportación se la realiza a través de un archivo index.js
- El archivo SCSS tiene el mismo nombre del componente.

2. Sintaxis

- Se utiliza la sintaxis JavaScript ES6.

3. Nomenclatura

- Para el nombramiento de carpetas y archivos se utiliza Pascal Case.

4. Iteraciones

- Se utiliza iteración en línea para la lógica de operación.

5. Comentarios

- Dentro de los componentes los comentarios se incluyen con la notación de código usando llaves {}.

6. Uso de comillas

- Se utilizan comillas dobles.
- En el caso de concatenar con variables se usan plantillas literales.

7. Condiciones

- Se utiliza operador ternario.

Especificaciones de codificación en Strapi

1. Colecciones y componentes

- Las colecciones y componentes utilizan Pascal Case.

2. Atributos

- Los atributos utilizan Snake Case.

ANEXO G: Diagramas UML

Caso de uso - Maestro

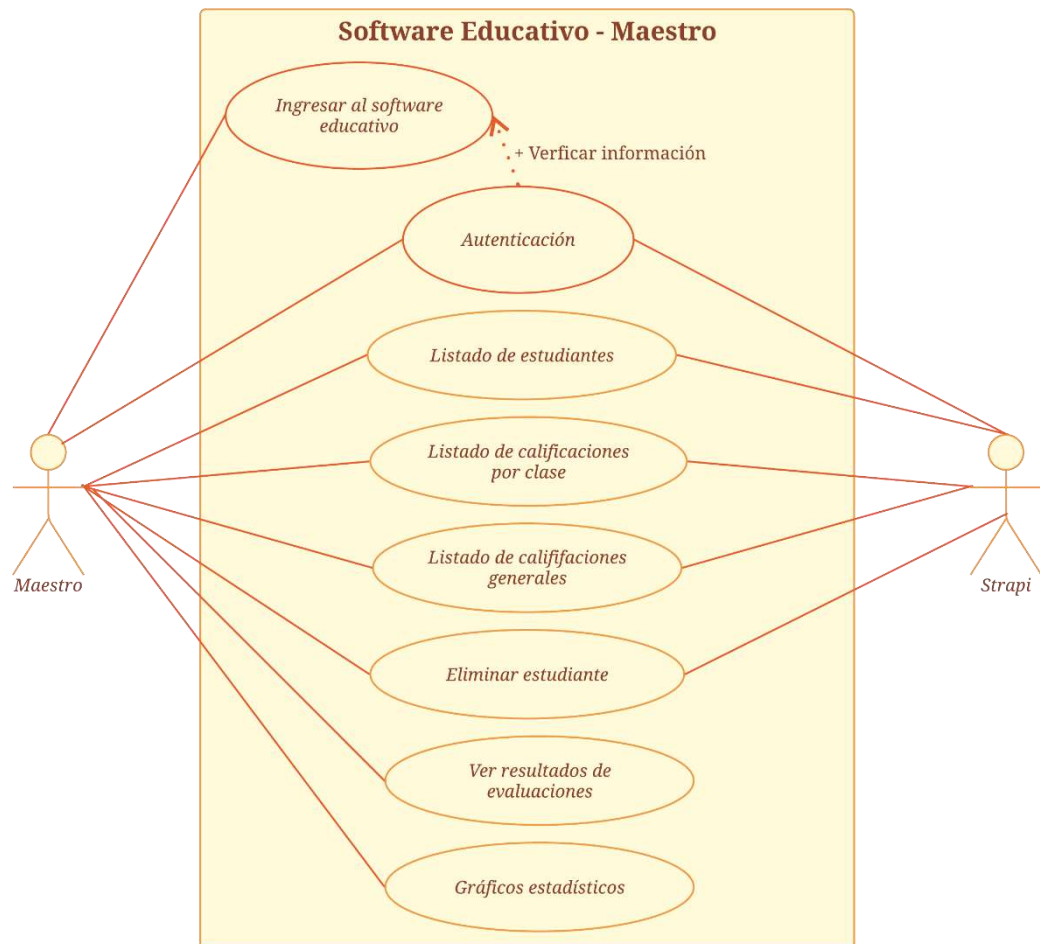


Figura 1: Diagrama de casos de uso - Maestro

Realizado por: Silva, J., 2022

Caso de uso - Estudiante

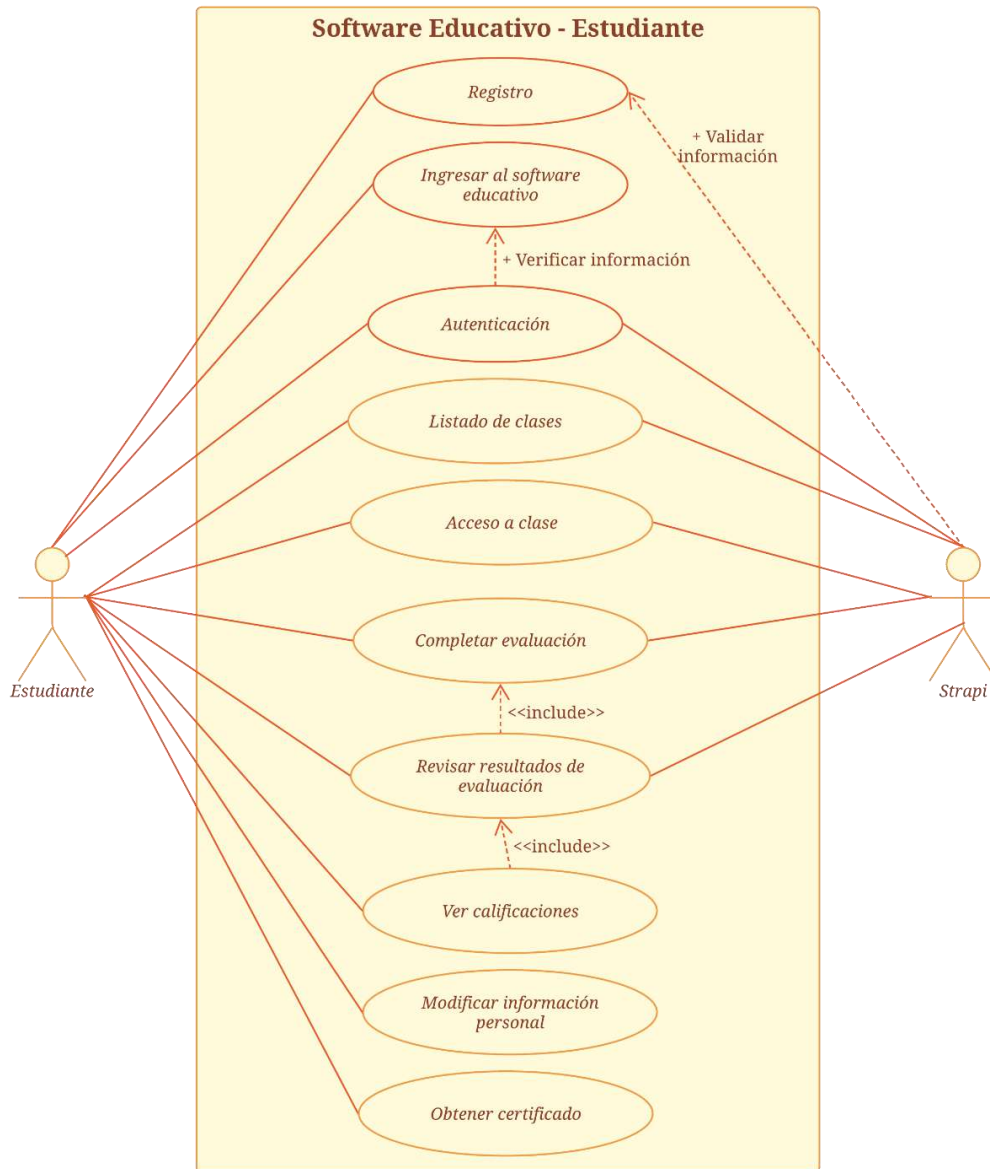


Figura 1: Diagrama de casos de uso - Maestro

Realizado por: Silva, J., 2022

ANEXO H: Diccionario de datos

Tabla 1: category

Nombre:	CATEGORY				
Descripción:	Categoría de las clases				
Campo	Descripción	is_key	Tipo de dato	NULL	Restricción
id	ID de la categoría	PK	integer(32)	NOT NULL	Serial
name	Nombre de la categoría		character varying(255)	NULL	
description	Descripción de la categoría		character varying(255)	NULL	
url	Ruta de la categoría	UK	character varying(255)	NULL	Autogenerado de name
position	Posición en el menú de la categoría		integer(32)	NULL	
created_at	Fecha y hora de creación		timestamp without time zone(6)	NULL	
updated_at	Fecha y hora de actualización		timestamp without time zone(6)	NULL	
published_at	Fecha y hora de publicación		timestamp without time zone(6)	NULL	
created_by_id	ID del usuario que crea la categoría	FK	integer(32)	NULL	created_at <= current_date
updated_by_id	ID del usuario que modifica la categoría	FK	integer(32)	NULL	updated_at <= current_date

Realizado por: Silva, J., 2022

Tabla 2: class

Nombre:	CLASS				
Descripción:	Información de la clase				
Campo	Descripción	is_key	Tipo de dato	NULL	Restricción
id	ID de la clase	PK	integer(32)	NOT NULL	Serial
title	Nombre de la clase		character varying(255)	NULL	
url	Ruta de la clase	UK	character varying(255)	NULL	Autogenerado de title
video_url	Enlace de YouTube de la clase		character varying(255)	NULL	
geo_materia_l_id	ID del applet de GeoGebra		character varying(255)	NULL	

created_at	Fecha y hora de creación		timestamp without time zone(6)	NULL	
updated_at	Fecha y hora de actualización		timestamp without time zone(6)	NULL	
published_at	Fecha y hora de publicación		timestamp without time zone(6)	NULL	
created_by_id	ID del usuario que crea la clase	FK	integer(32)	NULL	created_at <= current_date
updated_by_id	ID del usuario que modifica la clase	FK	integer(32)	NULL	updated_at <= current_date

Realizado por: Silva, J., 2022

Tabla 3: introduction

Nombre:	INTRODUCTION				
Descripción:	Información de la Introducción				
Campo	Descripción	is_key	Tipo de dato	NULL	Restricción
id	ID de la introducción	PK	integer(32)	NOT NULL	
title	Título		character varying(255)	NULL	
summary	Resumen de la introducción		text	NULL	
content	Contenido principal		text	NULL	

Realizado por: Silva, J., 2022

Tabla 4: external_url

Nombre:	EXTERNAL_URL				
Descripción:	Información de los enlaces externos				
Campo	Descripción	is_key	Tipo de dato	NULL	Restricción
id	ID del enlace externo	PK	integer(32)	NOT NULL	
title	Nombre de la página		character varying(255)	NULL	
url	Enlace de la página		character varying(255)	NULL	

Realizado por: Silva, J., 2022

Tabla 5: quiz

Nombre:	QUIZ				
Descripción:	Información de la prueba				
Campo	Descripción	is_key	Tipo de dato	NULL	Restricción
id	ID de la prueba	PK	integer(32)	NOT NULL	
name	Nombre de la prueba		character varying(255)	NULL	

description	Descripción de la prueba		text	NULL	
type	Tipo de prueba		character varying(255)	NULL	type: "diagnostic ", "class", "exam"
created_at	Fecha y hora de creación		timestamp without time zone(6)	NULL	
updated_at	Fecha y hora de actualización		timestamp without time zone(6)	NULL	created_at <= current_date
published_at	Fecha y hora de publicación		timestamp without time zone(6)	NULL	updated_at <= current_date
created_by_id	ID del usuario que crea la prueba	FK	integer(32)	NULL	
updated_by_id	ID del usuario que modifica la prueba	FK	integer(32)	NULL	

Realizado por: Silva, J., 2022

Tabla 6: question

Nombre:	QUESTION				
Descripción:	Información de la pregunta				
Campo	Descripción	is_key	Tipo de dato	NULL	Restricción
id	ID de la pregunta	PK	integer(32)	NOT NULL	
name	Nombre de la pregunta		character varying(255)	NULL	
question	Pregunta		text	NULL	
explanation	Explicación del procedimiento		text	NULL	
created_at	Fecha y hora de creación		timestamp without time zone(6)	NULL	
updated_at	Fecha y hora de actualización		timestamp without time zone(6)	NULL	
published_at	Fecha y hora de publicación		timestamp without time zone(6)	NULL	
created_by_id	ID del usuario que crea la nota	FK	integer(32)	NULL	created_at <= current_date
updated_by_id	ID del usuario que modifica la nota	FK	integer(32)	NULL	updated_at <= current_date

Realizado por: Silva, J., 2022

Tabla 7: grade

Nombre:	GRADE
Descripción:	Información de la nota

Campo	Descripción	is_key	Tipo de dato	NULL	Restricción
id	ID de la nota	PK	integer(32)	NOT NULL	
quiz_grade	Calificación de la nota		numeric(10,2)	NULL	quiz_grade <= 10
results	Respuestas de la prueba		jsonb	NULL	json
time_spent	Tiempo empleado		integer(32)	NULL	
correct_answers	Numero de respuestas correctas		integer(32)	NULL	correct_answers <= 4
created_at	Fecha y hora de creación		timestamp without time zone(6)	NULL	created_at <= current_date
updated_at	Fecha y hora de actualización		timestamp without time zone(6)	NULL	updated_at <= current_date
published_at	Fecha y hora de publicación		timestamp without time zone(6)	NULL	
created_by_id	ID del usuario que crea la nota	FK	integer(32)	NULL	

Realizado por: Silva, J., 2022

Tabla 8: final_result

Nombre:	FINAL_RESULT				
Descripción:	Información de la calificación final				
Campo	Descripción	is_key	Tipo de dato	NULL	Restricción
id	ID de la calificación final	PK	integer(32)	NOT NULL	
grade	Calificación final		numeric(10,2)	NULL	grade <= 10
created_at	Fecha y hora de creación		timestamp without time zone(6)	NULL	created_at <= current_date
updated_at	Fecha y hora de actualización		timestamp without time zone(6)	NULL	updated_at <= current_date
published_at	Fecha y hora de publicación		timestamp without time zone(6)	NULL	published_at <= current_date
created_by_id	ID del usuario que crea la nota	FK	integer(32)	NULL	
updated_by_id	ID del usuario que modifica la nota	FK	integer(32)	NULL	

Realizado por: Silva, J., 2022

ANEXO I: Diseño de interfaces

Seguimiento de estudiantes (Módulo Docente)

Listado de estudiantes

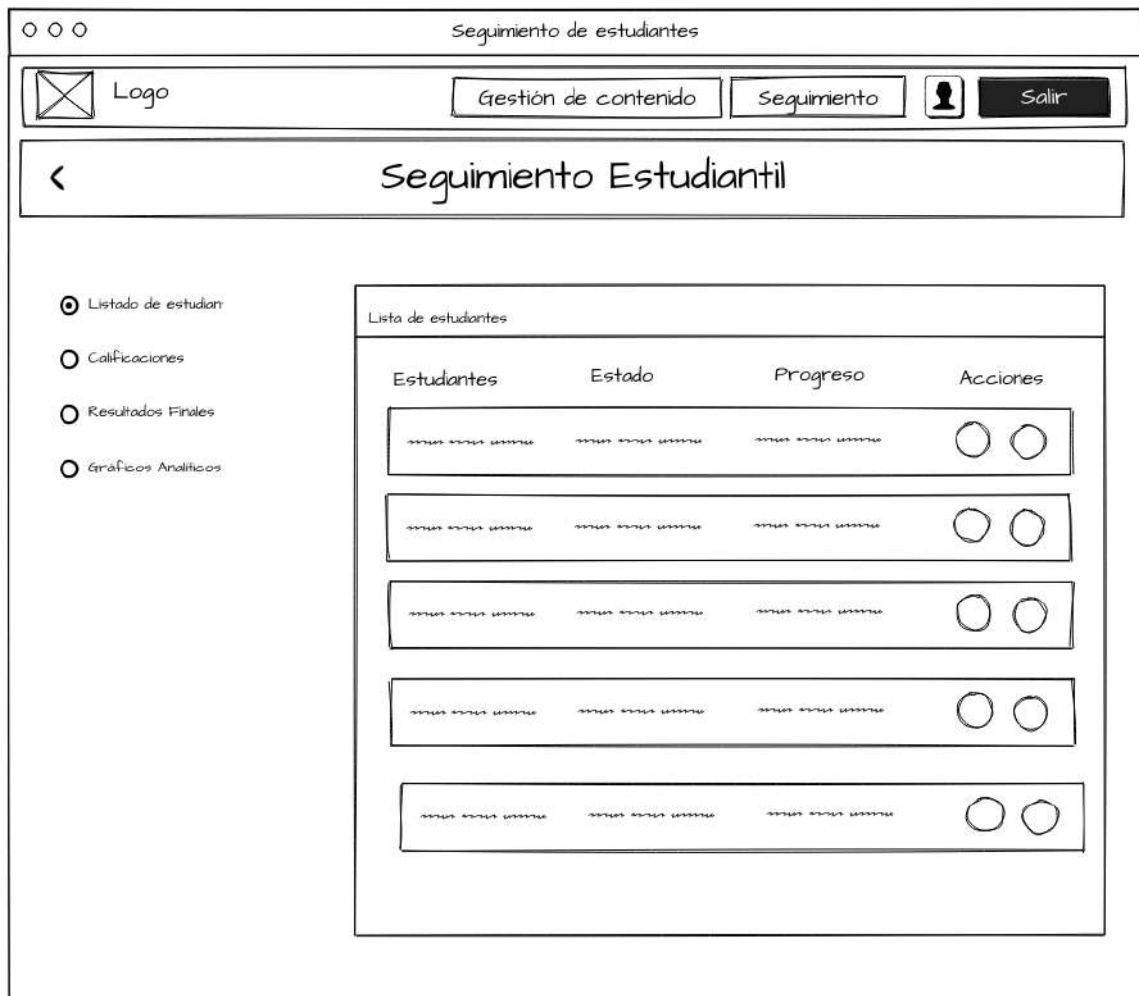


Figura 1: Pantalla de seguimiento estudiantil (Listado de estudiantes)

Realizado por: Silva, J., 2022

Seguimiento de estudiantes (Módulo Docente)

Calificaciones por tema

The screenshot shows a web application interface for student tracking. At the top, there is a navigation bar with a logo placeholder, a 'Gestión de contenido' button, a 'Seguimiento' button, a user profile icon, and a 'Salir' button. Below this is a header for 'Seguimiento Estudiantil' with a back arrow. On the left, there is a sidebar menu with four options: 'Listado de estudiantes', 'Calificaciones' (selected), 'Resultados Finales', and 'Gráficos Analíticos'. The main content area is titled 'Calificaciones de las clases' and displays two tables of scores.

Calificaciones de las clases

Introducción a la Geometría

Estudiante	Respuestas Correctas	Calificación
Estudiante 1	1 de 5	2 / 10
Estudiante 2	2 de 5	4 / 10
Estudiante 3	5 de 5	10 / 10
Estudiante 4	4 de 5	8 / 10
Estudiante 5	0 de 5	0 / 10

Punto medio de un segmento

Estudiante	Respuestas Correctas	Calificación
Estudiante 1	1 de 5	2 / 10
Estudiante 2	2 de 5	4 / 10
Estudiante 3	5 de 5	10 / 10

Figura 2: Pantalla de seguimiento estudiantil (Calificaciones por tema)

Realizado por: Silva, J., 2022

Seguimiento de estudiantes (Módulo Docente)

Resultados Finales

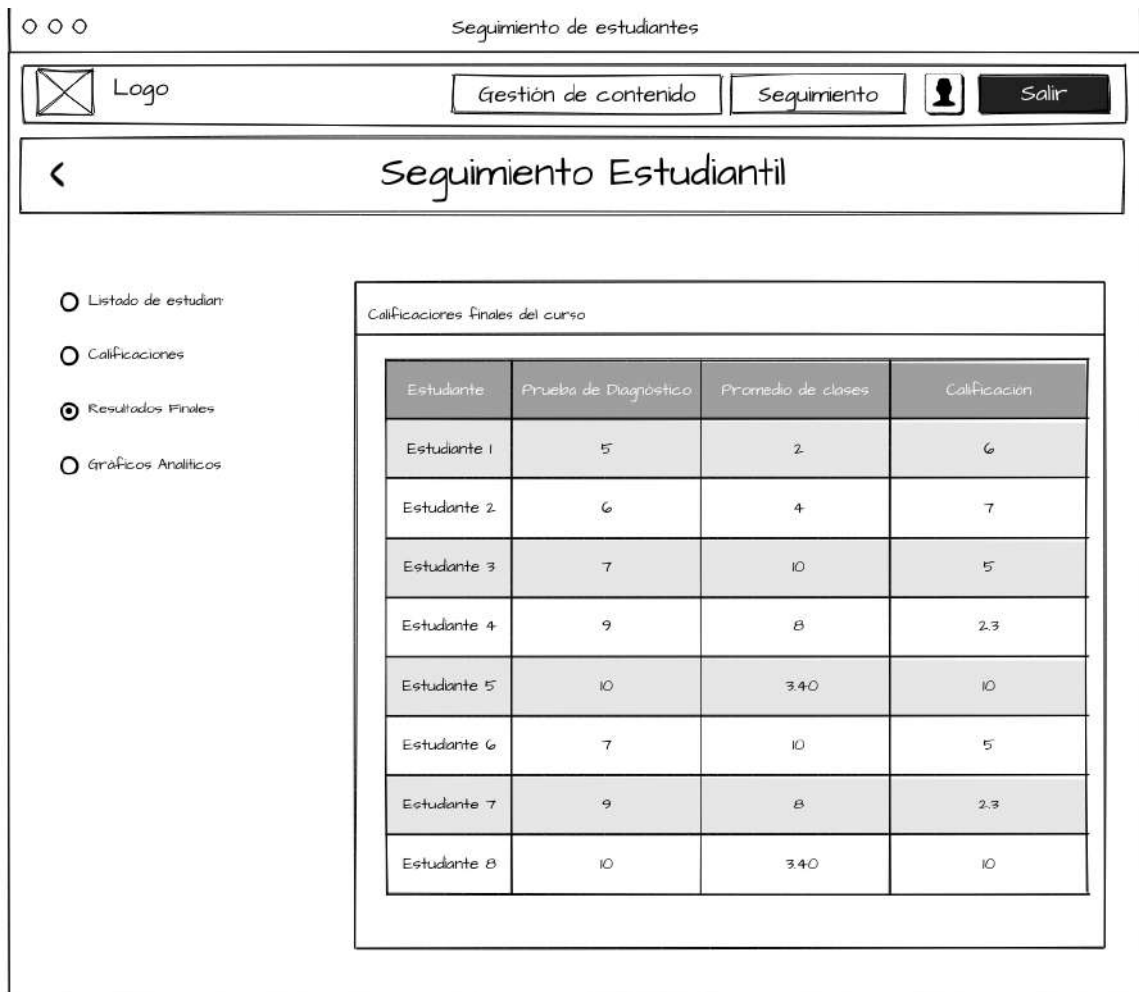


Figura 3: Pantalla de seguimiento estudiantil (Resultados Finales)

Realizado por: Silva, J., 2022

Seguimiento de estudiantes (Módulo Docente)

Gráficas Analíticas

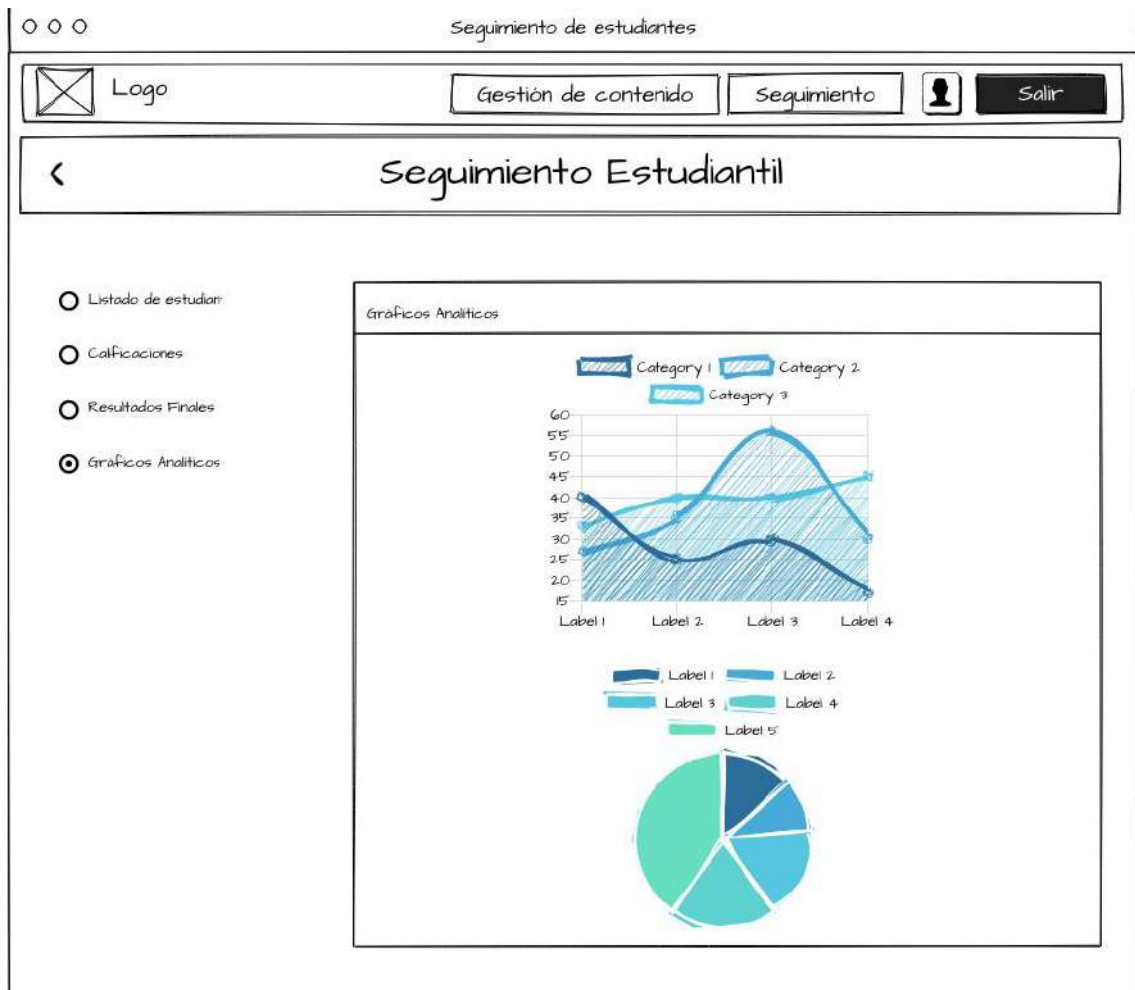


Figura 2: Pantalla de seguimiento estudiantil (Gráficas Analíticas)

Realizado por: Silva, J., 2022

ANEXO J: Plan de pruebas

Tabla 1: Historial de versiones

Fecha	Versión	Autor	Descripción
03-06-2022	1.0	John Silva	Definición de plan de pruebas
24-06-2022	1.1.0	John Silva	Pruebas de la versión 1.0.0 de E-Geometry

Realizado por: Silva, J., 2022

Tabla 2: Información del proyecto

Proyecto	Software educativo: Geometría y medida
Fecha de preparación	04-04-2022
Cliente	Ing. Miriam Avila
Líder de proyecto	Ing. Diego Avila Ing. Miguel Duque
Líder de pruebas de software	John Silva

Realizado por: Silva, J., 2022

RESUMEN

E-Geometry es un software educativo para reforzar el aprendizaje de geometría y medida en estudiantes de primero de bachillerato enfocándose y expandiendo los temas propuestos en el libro de matemáticas por el ministerio de educación. tiene como principal enfoque la utilización de geometría dinámica para la creación de modelos geométricos con los que los estudiantes puedan interactuar.

Este plan de pruebas abarca desde el alcance de las pruebas es decir lo que se va a aprobar los elementos que están fuera del alcance los tipos de prueba a realizar juntamente con la estrategia, los recursos hardware, software, herramientas, la planificación y los datos obtenidos en las pruebas.

ALCANCE DE LAS PRUEBAS

Elementos de pruebas

Se van a aprobar las siguientes historias de usuario definidas en el product backlog:

Tabla 3: Historias de usuario a probar del Product Backlog

Elementos de pruebas	
ID	Nombre de la historia
HT-01	Definir los requerimientos del software educativo
HT-02	Definir la arquitectura del software educativo
HT-03	Definir el estándar de codificación
HT-04	Diseño de la base de datos
HT-05	Diseñar las interfaces del software educativo
HU-01	Registro de usuarios
HU-02	Autenticación de usuarios
HU-03	Restablecer contraseña mediante correo electrónico
HU-04	Crear, modifica eliminar y visualizar las clases
HU-05	Crear, modifica eliminar y visualizar las categorías
HU-06	Crear, modifica eliminar y visualizar las pruebas
HU-07	Crear, modifica eliminar y visualizar las preguntas
HU-08	Enlazar temas a una categoría
HU-09	Enlazar preguntas a una prueba
HU-10	Enlazar prueba a un tema
HU-11	Modificar información personal del estudiante
HU-12	Visualizar listado de temas
HU-13	Visualizar una introducción a la clase
HU-14	Visualizar contenido multimedia de la clase
HU-15	Visualizar e interactuar con un modelo geométrico sobre la clase
HU-16	Realizar una evaluación sobre la clase
HU-17	Obtener una puntuación mínima para finalizar la clase
HU-18	Retroalimentación de cada evaluación
HU-19	Visualizar los temas completos e incompletos
HU-20	Visualizar un resumen sobre las calificaciones de las clases
HU-21	Visualizar los resultados finales y el progreso
HU-22	Obtener un certificado al finalizar todas las clases
HU-23	Visualizar listado de estudiantes
HU-24	Visualizar calificaciones por clase
HU-25	Visualizar calificaciones generales
HU-26	Gráficas de calificaciones

Realizado por: Silva, J., 2022

Funcionalidades por no probar

A continuación, se detallan los elementos de pruebas que están fuera del alcance además de especificar los posibles riesgos al no ser controlados.

Tabla 4: Pruebas fuera del alcance

Fuera de alcance		
ID	Elementos	Posibles inconvenientes
FA-01	Pruebas de rendimiento	En condiciones de alta carga o concurrencia el software educativo podría fallar
FA-02	Pruebas de seguridad	El software educativo podría presentar filtración o pérdida de datos
FA-03	Pruebas de accesibilidad	Grupos de usuarios específicos podrían no utilizar todas las funciones del software educativo
FA-04	Diseño responsivo	El software educativo podría no ser utilizado en dispositivos móviles

Realizado por: Silva, J., 2022

Enfoque de pruebas (estrategia)

A continuación, se listan las estrategias utilizadas para la realización de pruebas:

1. Pruebas de funcionalidades basadas en las historias de usuario
2. Pruebas de exploración
3. Prueba en diferentes navegadores (Cross browser testing)
4. Pruebas de aceptación

CRITERIOS DE ACEPTACIÓN O RECHAZO

Para considerar completo el plan de pruebas se deberán completar entre 95 a 100% las pruebas de aceptación descritas en cada una de las historias de usuario seleccionadas, lo que incluye:

- Las pruebas de aceptación descritas en cada una de las historias de usuario seleccionadas.
- Los casos de prueba exitosos deberán cubrir un 90% o superior.
- Los componentes que serán cubiertos son los que tienen prioridad alta o crítica.
- El porcentaje de los defectos corregidos debe ser superior al 90%.

ENTREGABLES

Como parte de la ejecución del plan de pruebas se entregará este documento, los casos de prueba generados, evidencia de pruebas y los reportes emitidos por las herramientas de prueba utilizadas.

RECURSOS

Hardware

Tabla 5: Recursos hardware

Cantidad	Recurso	Características
1	Computadora	AMD Ryzen 9
1	Servidor	Heroku

Realizado por: Silva, J., 2022

Software

Tabla 6: Recursos software

Recurso	Descripción	Características
Visual Studio Code	Editor de código	Libre
Strapi	Sistema de gestión de contenido sin cabeza	Gratis
Postman	Pruebas para API	Gratis
PostgreSQL	Sistema de base de datos	Libre
GeoGebra	Applets dinámicos	Libre
Vercel	Despliegue del cliente	Gratis
Jira	Gestor de proyectos	Gratis

Realizado por: Silva, J., 2022

Herramientas de pruebas requeridas

Tabla 7: Herramientas de pruebas requeridas

Herramienta	Descripción
Postman	Pruebas para APIs de Strapi.
Jira	Gestión de las historias de usuario, sprints, versiones.
TestRail	Organización, gestión y rastreo del proceso de pruebas.
Selenium	Pruebas exploratorias asistidas por automatización.
Chrome DevTools	Herramienta de desarrollo integrada en el buscador de Chrome para simulación de diferentes dispositivos.

Realizado por: Silva, J., 2022

PLANIFICACIÓN Y ORGANIZACIÓN

Procedimientos para las pruebas

Las pruebas estarán divididas en 2 categorías las que corresponden a Strapi (backend) que serán realizadas utilizando Postman para verificar que las APIs generadas funcionen de manera correcta. En NextJS (frontend) se utilizará Selenium y Chrome DevTools Para realizar pruebas de exploratorias automatizadas sobre la interfaz de usuario.

Las suites y casos de prueba serán gestionados desde TUSKR que se encuentra integrado con Jira para la generación de informes del proyecto.

EJECUCIÓN DE PRUEBAS

NextJS

Resultados de la ejecución de todos los casos de pruebas:

Overview			
STATUS		TEST CASES	%
 Passed		60	87
 Retest		2	3
 Failed		1	1
 Blocked		1	1
 Untested		3	4
Total		69	89%

Test Cases

KEY	NAME	STATUS
Auth > User register		
C-2	User account creation	Passed
C-4	Form field validation	Passed
Auth > User login		
C-5	Log in	Passed
Home Page > Categories		
C-6	List of categories	Passed
Home Page > Class list		
C-7	List of all classes	Passed
C-8	Class status	Passed
Home Page > Menu bar		
C-9	Menu bar	Passed
Class Page > Introduction		
C-10	Class title and bibliographic information	Passed

KEY	NAME	STATUS
C-18	Images and videos	Passed
C-19	Tables and diagrams	Passed
C-20	Formulas and equations with mathematical format	Passed
C-21	Embed information	Passed
C-22	URL	Passed
Class Page > Media video		
C-11	Multimedia video about the class	Passed
C-23	Video controls	Passed
Class Page > Dynamic geometry		
C-12	Dynamic geometry component	Passed
C-13	Manipulation of geometric models	Passed
C-14	Keyboard data entry	Passed
C-15	Reset chart to initial state	Passed
C-16	Display graph full screen	Passed
C-17	View model explanation	Passed
Class Page > Quiz		

KEY	NAME	STATUS
C-24	Quiz about the selected class	Passed
C-25	Number of questions	Passed
C-26	Time elapsed	Passed
C-27	Time left	Passed
C-28	Validate response selection	Passed
C-29	Picture question format	Passed
C-30	Minimum score required	Passed
C-31	Random questions	Passed
C-32	Results feedback	Passed
User Page > Email section		
C-33	Current email	Passed
C-34	Change email	Passed
C-35	Field validation	Passed
C-36	Successful update	Passed
User Page > Names section		
C-37	Current name	Passed

KEY	NAME	STATUS
C-38	Field validation	Passed
C-39	Successful update	Passed
C-40	Change username	Passed
User Page > Password section		
C-41	Current password	Passed
C-42	Change password	Retest
C-43	Field validation	Failed
C-44	Successful update	Blocked
Grades Page > Class grades		
C-45	Class grades list	Passed
Grades Page > Final results		
C-46	Diagnostic and final grade	Passed
C-47	Average obtained	Passed
Grades Page > Progress charts		
C-48	Graphs of statistics on the progress of the classes	Passed
Certificate Page > Generated certificate		

KEY	NAME	STATUS
C-49	Certificate generation	Passed
C-50	User information	Passed
C-51	Progress validation	Passed
Certificate Page > Certificate download		
C-52	Certificate download	Passed
C-53	Data validation	Untested
C-54	Generated document	Untested
Teacher Page > Student list		
C-55	List of registered students	Passed
C-56	Class progress	Passed
C-57	Student account status	Passed
C-58	Remove student	Passed
Teacher Page > Grades grouped by class		
C-59	Grouped list of class grades	Passed
C-60	Table with data	Passed
C-61	View responses	Passed

KEY	NAME	STATUS
C-66	Pdf report generation	Passed
C-67	Data export to Excel	Passed
Teacher Page > Overall results		
C-62	Student matrix and grades	Retest
C-63	General average of the classes	Untested
C-64	Pdf report generation	Untested
C-65	Data export to Excel	Untested
Teacher Page > Statistical graphs		
C-68	Area chart	Passed
C-69	Bar chart	Passed
C-70	Charts by categories	Passed
C-71	General average of the categories	Passed

ANEXO K: Pruebas de aceptación

PRUEBA DE ACEPTACIÓN			
ID	PA-01	Nombre	Autenticación de usuarios
Responsables	Ing. Diego Avila Ing. Miriam Avila Ing. Miguel Duque Est. John Silva		Fecha 18/05/2022
Descripción	Se verificará que los usuarios puedan crear una cuenta, iniciar sesión y recuperar su contraseña.		
Estado de la prueba	EXITOSA		
ACTIVIDADES EVALUADAS			
ID_PA	Descripción		Estado
HU01 – PA01	Validación de campos requeridos para el registro		Finalizada
HU01 – PA02	Verificar que el usuario sea único		Finalizada
HU01 – PA03	Notificación y verificación a través de correo electrónico		Finalizada
HU01 – PA04	Verificar almacenamiento de usuarios en Strapi		Finalizada
HU02 – PA01	Ingreso al software educativo con las funcionalidades dependiendo del rol		Finalizada
HU02 – PA02	Notificación de error de inicio de sesión		Finalizada
HU03 – PA01	Verificar que el usuario esté registrado		Finalizada
HU03 – PA02	Enviar enlace de recuperación a correo electrónico		Finalizada
TAREAS DE INGENIERÍA			
ID_TI	Descripción		Resultado
HU01 – TI01	Desarrollar la interfaz de registro e inicio de sesión		Exitoso
HU01 – TI02	Implementar el servicio web para la comunicación del software educativo con Strapi		Exitoso
HU02 – TI01	Implementar el servicio para la autenticación de los usuarios en el software educativo		Exitoso
HU03 – TI01	Desarrollo de la interfaz de recuperación de contraseña		Exitoso
HU03 – TI02	Verificación de existencia de usuario en la base de datos		Exitoso
HU03 – TI03	Envío de enlace de recuperación al correo electrónico		Exitoso

PRUEBA DE ACEPTACIÓN			
ID	PA-02	Nombre	Operaciones CRUD para clases, categorías, pruebas y preguntas
Responsables	Ing. Diego Avila Ing. Miriam Avila Ing. Miguel Duque Est. John Silva		Fecha 25/05/2022
Descripción	Se verificará la creación, lectura, edición y eliminación de registros para las clases, categorías, pruebas y preguntas.		

Estado de la prueba	EXITOSA	
ACTIVIDADES EVALUADAS		
ID_PA	Descripción	Estado
HU04 – PA01	Gestionar la información de las clases en Strapi	Finalizada
HU05 – PA01	Gestionar la información de las categorías en Strapi	Finalizada
HU06 – PA01	Configuración de permisos de la colección	Finalizada
HU07 – PA01	Gestionar la información de las preguntas en Strapi	Finalizada
TAREAS DE INGENIERÍA		
ID_TI	Descripción	Resultado
HU04 – TI01	Creación de la colección “clases”	Exitoso
HU04 – TI02	Configuración de permisos de la colección “clases”	Exitoso
HU05 – TI01	Creación de la colección “categorías”	Exitoso
HU05 – TI02	Configuración de permisos de la colección “categorías”	Exitoso
HU06 – TI01	Creación de la colección “pruebas”	Exitoso
HU06 – TI02	Configuración de permisos de la colección “pruebas”	Exitoso
HU07 – TI01	Creación de la colección “preguntas”	Exitoso
HU07 – TI02	Configuración de permisos de la colección “preguntas”	Exitoso

Realizado por: Silva, J., 2022

PRUEBA DE ACEPTACIÓN			
ID	PA-03	Nombre	Relación entre las clases con categorías y las pruebas con sus preguntas
Responsables	Ing. Diego Avila Ing. Miriam Avila Ing. Miguel Duque Est. John Silva	Fecha	26/05/2022
Descripción	Se verificará la relación entre las diferentes clases con una determinada categoría y la relación de varias preguntas con su respectiva prueba.		
Estado de la prueba	EXITOSA		
ACTIVIDADES EVALUADAS			
ID_PA	Descripción	Estado	
HU08 – PA01	Verificar que los temas se encuentren en determinada categoría	Finalizada	
HU09 – PA01	Verificar que los preguntas se encuentren en determinada prueba	Finalizada	
TAREAS DE INGENIERÍA			
ID_TI	Descripción	Resultado	
HU08 – TI01	Creación de la colección “categorías”	Exitoso	
HU08 – TI02	Relación entre clases y categorías	Exitoso	
HU08 – TI03	Configuración de permisos de la colección	Exitoso	
HU09 – TI01	Relación entre pruebas y preguntas	Exitoso	

Realizado por: Silva, J., 2022

PRUEBA DE ACEPTACIÓN			
ID	PA-04	Nombre	Modificación y actualización de la información personal del estudiante
Responsables	Ing. Diego Avila Ing. Miriam Avila Ing. Miguel Duque Est. John Silva	Fecha	26/05/2022
Descripción	Se verificará la actualización de los datos personales del estudiante		
Estado de la prueba	EXITOSA		
ACTIVIDADES EVALUADAS			
ID_PA	Descripción		Estado
HU11 – PA01	Verificar que la información modificada se actualice en Strapi y el software educativo		Finalizada
TAREAS DE INGENIERÍA			
ID_TI	Descripción		Resultado
HU11 – TI01	Validación de token del estudiante		Exitoso
HU11 – TI02	Creación del servicio para modificar la información		Exitoso
HU11 – TI03	Creación de la interfaz para modificar la información		Exitoso

Realizado por: Silva, J., 2022

PRUEBA DE ACEPTACIÓN			
ID	PA-05	Nombre	Visualizar información de las clases en el software educativo
Responsables	Ing. Diego Avila Ing. Miriam Avila Ing. Miguel Duque Est. John Silva	Fecha	04/06/2022
Descripción	Se verificará la correcta visualización del listado de todas las clases, las introducciones, vídeos, modelos geométricos interactivos y pruebas de cada clase.		
Estado de la prueba	EXITOSA		
ACTIVIDADES EVALUADAS			
ID_PA	Descripción		Estado
HU12 – PA01	Verificar que al iniciar sesión el estudiante vio un listado de todas las clases		Finalizada
HU13 – PA01	Al acceder a una clase se muestra una sección introductoria		Finalizada
HU14 – PA01	Al acceder a una clase se muestra un video sobre la clase		Finalizada
HU15 – PA01	Al acceder a una clase se muestra un modelo geométrico interactivo		Finalizada
HU16 – PA01	Al acceder a una clase se muestra una evaluación sobre la clase		Finalizada
HU16 – PA02	Las calificaciones son enviadas a Strapi		Finalizada
TAREAS DE INGENIERÍA			
ID_TI	Descripción		Resultado

HU12 – TI01	Desarrollar la interfaz para las clases	Exitoso
HU12 – TI02	Creación del servicio para obtener el listado de todas las clases	Exitoso
HU12 – TI03	Agrupar las clases según la categoría	Exitoso
HU13 – TI01	Desarrollar la interfaz para visualizar la introducción de la clase	Exitoso
HU13 – TI02	Creación del servicio para obtener la información de la clase	Exitoso
HU14 – TI01	Desarrollar la interfaz para visualizar un video sobre de la clase	Exitoso
HU15 – TI01	Desarrollar la interfaz para visualizar el modelo geométrico sobre de la clase	Exitoso
HU15 – TI02	Desarrollar el servicio de comunicación del modelo con el software educativo	Exitoso
HU15 – TI03	Crear el modelo geométrico de cada clase	Exitoso
HU16 – TI01	Desarrollar la interfaz para la evaluación de cada clase	Exitoso
HU16 – TI02	Desarrollar el servicio para obtener la prueba con sus preguntas de cada clase	Exitoso
HU16 – TI03	Desarrollar el módulo de tiempo para la prueba	Exitoso
HU16 – TI04	Desarrollar el servicio para enviar la calificación obtenida a Strapi	Exitoso

Realizado por: Silva, J., 2022

PRUEBA DE ACEPTACIÓN			
ID	PA-06	Nombre	Puntuación mínima requerida en pruebas y retroalimentación
Responsables	Ing. Diego Avila Ing. Miriam Avila Ing. Miguel Duque Est. John Silva	Fecha	05/06/2022
Descripción	Se verificará la configuración de una puntuación mínima en las pruebas para dar como finalizada la clase y la visualización de una retroalimentación de cada evaluación		
Estado de la prueba	EXITOSA		
ACTIVIDADES EVALUADAS			
ID_PA	Descripción		Estado
HU17 – PA01	Verificar que tenga la calificación mínima para finalizar la clase		Finalizada
HU18 – PA01	Verificar se muestre retroalimentación en cada prueba incluida la resolución de cada pregunta		Finalizada
TAREAS DE INGENIERÍA			
ID_TI	Descripción		Resultado
HU17 – TI01	Desarrollar la interfaz para visualizar un mensaje de puntuación mínima requerida		Exitoso
HU18 – TI01	Desarrollar la interfaz para ver los resultados de la prueba		Exitoso

Realizado por: Silva, J., 2022

PRUEBA DE ACEPTACIÓN			
ID	PA-07	Nombre	Visualizar las clases completas de incompletas, las calificaciones de las clases y los resultados finales con el progreso de las clases
Responsables	Ing. Diego Avila Ing. Miriam Avila Ing. Miguel Duque Est. John Silva	Fecha	05/06/2022
Descripción	Se verificará la visualización de las clases según su estado, un listado con las calificaciones de las clases, gráficos con los resultados y el progreso.		
Estado de la prueba	EXITOSA		
ACTIVIDADES EVALUADAS			
ID_PA	Descripción		Estado
HU19 – PA01	Verificar que se muestra un indicativo de las clases completas y sin completar		Finalizada
HU20 – PA01	Verificar que se muestran los datos de las calificaciones obtenidas en cada clase		Finalizada
HU21 – PA01	Verificar que se muestra los resultados finales y gráficas con el progreso del estudiante		Finalizada
TAREAS DE INGENIERÍA			
ID_TI	Descripción		Resultado
HU19 – TI01	Desarrollar el componente sobre el estado de la clase		Exitoso
HU20 – TI01	Desarrollar la interfaz de visualización de calificaciones de las clases		Exitoso
HU20 – TI02	Creación del servicio para obtener los detalles de las calificaciones de las clases		Exitoso
HU21 – TI01	Desarrollar la interfaz de visualización de resultados finales y gráficas		Exitoso

Realizado por: Silva, J., 2022

PRUEBA DE ACEPTACIÓN			
ID	PA-08	Nombre	Generación de certificado al concluir todas las clases
Responsables	Ing. Diego Avila Ing. Miriam Avila Ing. Miguel Duque Est. John Silva	Fecha	06/06/2022
Descripción	Se verificará que al completar todas las clases se genere un certificado con los Datos del estudiante y la fecha de finalización.		
Estado de la prueba	EXITOSA		
ACTIVIDADES EVALUADAS			
ID_PA	Descripción		Estado
HU22 – PA01	Verificar que se presente un certificado de finalización		Finalizada

HU22 – PA02	Verificar que sólo al completar todas las clases se pueda visualizar el certificado	Finalizada
TAREAS DE INGENIERÍA		
ID_TI	Descripción	Resultado
HU22 – TI01	Desarrollar la interfaz del certificado de finalización	Exitoso
HU22 – TI02	Comprobar que el estudiante haya finalizado todas las clases	Exitoso

Realizado por: Silva, J., 2022

PRUEBA DE ACEPTACIÓN			
ID	PA-10	Nombre	Seguimiento académico por parte del docente a los estudiantes
Responsables	Ing. Diego Avila Ing. Miriam Avila Ing. Miguel Duque Est. John Silva	Fecha	07/06/2022
Descripción	Se verificará el listado con todos los estudiantes, la visualización de las calificaciones agrupadas por clase, una matriz con las calificaciones generales y gráficas sobre el rendimiento académico.		
Estado de la prueba	EXITOSA		
ACTIVIDADES EVALUADAS			
ID_PA	Descripción	Estado	
HU23 – PA01	Verificar que solo el maestro puede listar los estudiantes	Finalizada	
HU23 – PA02	Verificar que se visualiza listado de estudiantes y su progreso	Finalizada	
HU24 – PA01	Verificar que se muestren las calificaciones dependiendo de cada clase	Finalizada	
HU25 – PA01	Verificar que se muestren las calificaciones de todos los estudiantes	Finalizada	
HU26 – PA01	Verificar que se muestran gráficas de estadísticas con la información sobre el progreso y las calificaciones de los estudiantes	Finalizada	
TAREAS DE INGENIERÍA			
ID_TI	Descripción	Resultado	
HU23 – TI01	Desarrollar la interfaz del listado de estudiantes y el indicador de progreso en las clases	Exitoso	
HU23 – TI02	Visualización de información dependiendo del rol “maestro”	Exitoso	
HU23 – TI03	Creación del servicio para obtener el listado de estudiantes y su progreso	Exitoso	
HU24 – TI01	Desarrollar la interfaz del listado de calificaciones por clase	Exitoso	
HU24 – TI02	Creación del servicio para obtener el listado de clases y sus calificaciones	Exitoso	
HU25 – PA01	Verificar que se muestren las calificaciones de todos los estudiantes	Exitoso	
HU25 – PA01	Verificar que se muestren las calificaciones de todos los estudiantes	Exitoso	
HU26 – TI01	Desarrollar la interfaz para visualizar las gráficas	Exitoso	
HU26 – TI02	Creación del servicio para analizar y graficar el progreso y las calificaciones de los estudiantes	Exitoso	

Realizado por: Silva, J., 2022

ANEXO L: Cuestionario utilizado

Hola! Comenzar cuestionario de satisfacción del software educativo

Dime qué piensas presiona Enter ↵

1→ La información de la introducción y el video de cada clase es clara y fácil de entender *

A Totalmente de acuerdo

B De acuerdo

C Ni de acuerdo ni en desacuerdo

D En desacuerdo

E Totalmente en desacuerdo

2 → La evaluación de cada clase tiene preguntas acordes a la temática y las puedo responder sin mayor problema *

A Totalmente de acuerdo

B De acuerdo

C Ni de acuerdo ni en desacuerdo

D En desacuerdo

E Totalmente en desacuerdo

Aceptar ✓

3 → Al finalizar la evaluación puedo revisar mis respuestas correctas o erróneas, y se explica el procedimiento o manera de resolver cada pregunta *

A Totalmente de acuerdo

B De acuerdo

C Ni de acuerdo ni en desacuerdo

D En desacuerdo

E Totalmente en desacuerdo

Aceptar ✓

4 → La pantalla de las clases es agradable *

A Totalmente de acuerdo

B De acuerdo

C Ni de acuerdo ni en desacuerdo

D En desacuerdo

E Totalmente en desacuerdo

Aceptar ✓

5 → Me gustó interactuar con las figuras interactivas *

A Totalmente de acuerdo

B De acuerdo

C Ni de acuerdo ni en desacuerdo

D En desacuerdo

E Totalmente en desacuerdo

Aceptar ✓

6 → Tuve complicaciones para entrar en la pantalla de calificaciones *

A Totalmente de acuerdo

B De acuerdo

C Ni de acuerdo ni en desacuerdo

D En desacuerdo

E Totalmente en desacuerdo

Aceptar ✓

7 → La pantalla de inicio es agradable y puedo seleccionar la clase que quiera de manera fácil *

A Totalmente de acuerdo

B De acuerdo

C Ni de acuerdo ni en desacuerdo

D En desacuerdo

E Totalmente en desacuerdo

Aceptar ✓

8 → Al ir a las diferentes pantallas pude regresar a donde estaba antes *

A Totalmente de acuerdo

B De acuerdo

C Ni de acuerdo ni en desacuerdo

D En desacuerdo

E Totalmente en desacuerdo

Aceptar ✓

9 → Recomendaría este software educativo a un amigo *

A Totalmente de acuerdo

B De acuerdo

C Ni de acuerdo ni en desacuerdo

D En desacuerdo

E Totalmente en desacuerdo

Aceptar ✓

10 → No tuve ningún error mientras utilizaba el software educativo *

A Totalmente de acuerdo

B De acuerdo

C Ni de acuerdo ni en desacuerdo

D En desacuerdo

E Totalmente en desacuerdo

Aceptar ✓

11 → Cuando tuve un error pude volver atrás y seguir navegando *

A Totalmente de acuerdo

B De acuerdo

C Ni de acuerdo ni en desacuerdo

D En desacuerdo

E Totalmente en desacuerdo

Aceptar ✓

12 → Se mostró un mensaje después de actualizar información de mi perfil *

A Totalmente de acuerdo

B De acuerdo

C Ni de acuerdo ni en desacuerdo

D En desacuerdo

E Totalmente en desacuerdo

Aceptar ✓

13 → Al poner un correo electrónico equivocado para iniciar sesión se mostró un mensaje *

A Totalmente de acuerdo

B De acuerdo

C Ni de acuerdo ni en desacuerdo

D En desacuerdo

E Totalmente en desacuerdo

Aceptar ✓

14 → Es cómodo navegar desde la "Introducción" de la clase hasta la "Evaluación" *

A Totalmente de acuerdo

B De acuerdo

C Ni de acuerdo ni en desacuerdo

D En desacuerdo

E Totalmente en desacuerdo

Aceptar ✓

15 → Es fácil interactuar con todo el software educativo *

A Totalmente de acuerdo

B De acuerdo

C Ni de acuerdo ni en desacuerdo

D En desacuerdo

E Totalmente en desacuerdo

Aceptar ✓

“ Hemos terminado! Muchas gracias!!

Finalizar pulsa Ctrl + Enter ↵




ESCUELA SUPERIOR POLITÉCNICA DE
CHIMBORAZO

DIRECCIÓN DE BIBLIOTECAS Y RECURSOS DEL
APRENDIZAJE



UNIDAD DE PROCESOS TÉCNICOS
REVISIÓN DE NORMAS TÉCNICAS, RESUMEN Y BIBLIOGRAFÍA

Fecha de entrega: 16 / 01 / 2023

INFORMACIÓN DEL AUTOR
Nombres – Apellidos: John Cristóbal Silva Espín
INFORMACIÓN INSTITUCIONAL
Facultad: Facultad de Informática y Electrónica - FIE
Carrera: Software
Título a optar: Ingeniero de Software
f. Analista de Biblioteca responsable:  Ing. Fernanda Arévalo M.



2376-DBRA-UPT-2022