

UNIVERSIDAD METROPOLITANA



FACULTAD DE INGENIERÍAS

CARRERA DE SISTEMAS DE INFORMACIÓN

SEDE QUITO

**TRABAJO DE TITULACIÓN PREVIO A LA OBTENCIÓN DEL TÍTULO DE
INGENIERO EN SISTEMAS DE INFORMACIÓN**

TEMA:

**SISTEMA WEB PARA LA GESTIÓN EFICIENTE DE MATRÍCULAS
ESTUDIANTILES EN LA ESCUELA PATRICIO ESPINOZA BERMEO**

AUTOR:

JAIRO FABRICIO MOGROVEJO TRUJILLO

ASESOR:

MSC. MIGUEL ÁNGEL FERNÁNDEZ MARÍN

QUITO – 2025

CERTIFICACIÓN DEL TUTOR

Ing. MIGUEL ANGEL FERNÁNDEZ MARÍN MSc., en calidad de asesor del trabajo de graduación o titulación.

CERTIFICA

Que el trabajo de graduación ha sido desarrollado de manera autónoma por el señor Jairo Fabricio Mogrovejo Trujillo, portador de la cédula de identidad No 172547935-4 con el tema: "Sistema web para la gestión eficiente de matrículas estudiantiles en la Escuela Patricio Espinoza Bermeo.". El señor mencionado ha cumplido con todos los requisitos legales exigidos.

Es todo cuanto puedo decir en honor a la verdad, facultando al interesado hacer uso de la presente.

Atentamente,

Ing. Miguel Ángel Fernández Marín, Msc.

TUTOR DE TRABAJO DE TITULACIÓN

CERTIFICADO DE AUTORÍA DE TRABAJO DE TITULACIÓN

Yo, **Jairo Fabricio Mogrovejo Trujillo**, estudiante de la Universidad Metropolitana del Ecuador “UMET”, Ingeniería en Sistemas de Información, declaro en forma libre y voluntaria que el presente trabajo de titulación que versa sobre: **SISTEMA WEB PARA LA GESTIÓN EFICIENTE DE MATRÍCULAS ESTUDIANTILES EN LA ESCUELA PATRICIO ESPINOZA BERMEO** y las expresiones vertidas en la misma, son autoría de la compareciente, las cuales se han realizado en base a recopilación bibliográfica, consultas de internet y consultas de campo.

En consecuencia, asumo la responsabilidad de la originalidad de la misma y el cuidado al referirme a las fuentes bibliográficas respectivas para fundamentar el contenido expuesto.

Atentamente,

Jairo Fabricio Mogrovejo Trujillo

C.C: 1725479354

AUTOR

CESIÓN DE DERECHOS DE AUTOR

Yo, **Jairo Fabricio Mogrovejo Trujillo**, en calidad de autor y titular de los derechos morales y patrimoniales del trabajo de titulación, **sistema web para la gestión eficiente de matrículas estudiantiles en la escuela Patricio Espinoza Bermeo**, cedo a favor de la Universidad Metropolitana del Ecuador una licencia gratuita, intransferible y no exclusiva para el uso no comercial de la obra, con fines estrictamente académicos. Conservo a mi favor todos los derechos de autor sobre la obra, establecidos en la normativa citada.

Así mismo, autorizo a la Universidad Metropolitana del Ecuador para que realice la digitalización y publicación de este trabajo de titulación en el repositorio virtual, de conformidad a lo dispuesto en el Art. 144 de la Ley Orgánica de Educación Superior.

El autor declara que la obra objeto de la presente autorización es original en su forma de expresión y no infringe el derecho de autor de terceros, asumiendo la responsabilidad por cualquier reclamación que pudiera presentarse por esta causa y liberando a la Universidad de toda responsabilidad.

Atentamente,

Jairo Fabricio Mogrovejo Trujillo

C.C: 1725479354

DEDICATORIA

Dedico este proyecto con todo mi corazón a mi madre, Marcela, por su apoyo incondicional, sus sabios consejos y por inculcarme valores y principios que han guiado mi camino. A mi padre, Patricio, por creer siempre en mí, por sus palabras de aliento y por enseñarme a perseguir mis sueños con determinación. A mis hermanas, por ser mi compañía constante, por su amor y el ejemplo de fortaleza que siempre me inspiran. A mis abuelitos, cuya vida es un modelo de esfuerzo y dedicación, y quienes me han enseñado el verdadero significado de la perseverancia. Gracias a cada uno de ustedes por brindarme su confianza y apoyo incondicional. Hoy celebro este logro con ustedes, pues cada paso ha sido posible gracias a su amor y motivación.

AGRADECIMIENTO

Quito expresar mi profundo agradecimiento a Dios, que ha sido mi guía a lo largo de mi camino. A mis padres les agradezco por brindarme la oportunidad de vivir una etapa como esta que es mi formación universitaria.

A mis hermanos, por estar a mi lado cuando más lo necesite, gracias por el apoyo que me han brindado y por estar presente a lo largo de mi vida, su ejemplo me ayudado a ser perseverante en mi vida.

A mi primo Steven, quien ha estado siempre para mí como un hermano, gracias por estar en los buenos y malos momentos por el apoyo incondicional y por los valiosos consejos que me ha brindado para guiarme hacia el camino correcto, tu ayuda y aliento ha han dado las fuerzas para poder superar cualquier desafío y seguir adelante.

A los docentes de la Universidad Metropolitana, que han dedicado todo su esfuerzo para mi formación académica, gracias por sus enseñanzas y conocimientos que me han inspirado a dar lo mejor de mí en cada paso de la carrera universitaria.

Gracias.

ÍNDICE

CERTIFICADO DE AUTORÍA DE TRABAJO DE TITULACIÓN	III
CESIÓN DE DERECHOS DE AUTOR.....	IV
DEDICATORIA.....	V
AGRADECIMIENTO.....	VI
RESUMEN	XII
ABSTRACT	XIII
INTRODUCCIÓN	1
Situación problemática	2
Formulación del problema	5
Objetivo	5
Objetivo General	5
Objetivos Específicos.....	6
Hipótesis	6
Justificación de la investigación.....	6
Tipo de investigación	7
Investigación Aplicada	7
Investigación Bibliográfica.....	7
Investigación de campo	7
Estructuración del documento.....	8
CAPÍTULO I	9
1. FUNDAMENTACIÓN TEÓRICA.....	9
1.1. Conceptos asociados a la investigación	9
1.2. Objetivos estratégicos de la organización en estudio.	9
1.2.1. Misión.....	9
1.2.2. Visión	10
1.3. Objeto de estudio.....	10
1.3.1. Campo de acción	10
1.3.2. Flujo actual de proceso	10
1.4. Herramientas y tecnologías	12
1.4.1. Sistemas automatizados existentes relacionados con el campo de acción.	12
1.4.2. Comparación de sistemas informáticos existentes	14
1.5. Tendencias y tecnologías actuales.....	15

1.6. Fundamentación de la metodología utilizada	16
1.6.1. Proceso Unificado Racional RUP.....	16
1.6.2. Metodología de trabajo basada en RUP.	17
1.6.3. Enterprise Architect.....	17
1.6.4. Lenguaje Unificado de Modelado UML	18
1.6.5. Visual Studio Code.....	18
1.6.6. Lenguaje de programación Dart.....	18
1.6.7. Flutter	19
1.6.8. Firebase	19
1.7. Conclusión del capítulo	20
CAPÍTULO II.....	21
2. MODELO DE NEGOCIO, REQUISITOS FUNCIONALES Y MODELOS DEL SISTEMA	21
2.1. Descripción del entorno	21
2.1.1. Modelo de caso de uso del negocio.....	21
2.1.2. Diagrama de caso de uso del negocio	23
2.1.3. Actores del negocio.....	24
2.1.4. Diagrama de actividades.....	25
2.1.5. Reglas del negocio.....	27
2.2. Requisitos de software	28
2.2.1. Requisitos Funcionales del sistema (RF)	28
2.2.2. Interfaz de usuario	29
2.3. Diagrama de caso de uso del sistema	31
2.3.1. Listado de casos de uso.....	33
2.4. Descripción de los casos de uso del sistema	33
2.5. Descripción de los casos de uso crear registro	34
2.6. Conclusión del capítulo.....	36
CAPÍTULO III.....	38
3. DESCRIPCIÓN DE LA PROPUESTA DE SOLUCIÓN	38
3.1.1. Diseño Arquitectónico	38
3.2. Diagrama de clases de diseño.....	39
3.3. Diagrama de secuencia	42
3.4. Diagrama de componentes.....	51
3.5. Diagrama de despliegue.....	52

.....	53
CAPÍTULO IV	54
4. PRUEBAS Y ANÁLISIS DE FACTIBILIDAD	54
4.1. Breve descripción del capítulo	54
4.2. Planificación basada en uno de los métodos de estimación.....	54
4.2.1. Caracterización de los requisitos según su complejidad.....	54
4.2.2. Clasificación de requisitos funcionales.....	55
4.2.3. Definición de factores de complejidad.....	56
4.2.4. Factores técnicos	56
4.2.5. Factores ambientales.....	58
4.2.6. Métricas a medir para realizar la estimación	58
4.2.7. Porcentaje de actividades por ciclo de vida	59
4.3. Casos de prueba	61
4.4. Pruebas de caja negra y validaciones	61
4.5. Validación de la implementación del sistema	73
4.6. Conclusión del capítulo.....	74
CONCLUSIONES.....	75
RECOMENDACIONES	76
BIBLIOGRAFÍA	77

ÍNDICE DE TABLAS

Tabla 1 Comparación de sistemas informáticos existentes.....	14
Tabla 2 Comparación de Metodologías.....	16
Tabla 3 Descripción de los actores del Negocio	24
Tabla 4 Descripción del trabajador del negocio	24
Tabla 5 Interfaz de usuario.....	29
Tabla 6 Usabilidad.....	30
Tabla 7 Seguridad.....	30
Tabla 8 Confiabilidad y Disponibilidad.....	30
Tabla 9 Eficiencia del sistema	31
Tabla 10 Arquitectura	31
Tabla 11 Priorización de los casos de uso	33
Tabla 12 Descripción de los casos de uso crear registro	34
Tabla 13 Complejidad de los requisitos funcionales	55
Tabla 14 Clasificación de requisitos funcionales	55
Tabla 15 Definición de factores de complejidad.....	56
Tabla 16 Definición de factores para medir la complejidad.	56
Tabla 17 Descripción de los factores técnicos	57
Tabla 18 Factores ambientales	58
Tabla 19 Descripción de los valores de ponderación de factores ambientales	58
Tabla 20 Porcentaje de actividades por ciclo de vida.....	59
Tabla 21 Porcentaje de actividades por ciclo de vida para el sistema web	60
Tabla 22 Identificación de las clases de equivalencias:(Crear Alumno).....	61
Tabla 23 Casos de prueba para clases validad (Crear Alumno)	62
Tabla 24 Casos de prueba para clases inválida.....	63
Tabla 25 Matriz de casos de prueba	64
Tabla 26 Identificación de las clases de equivalencias:(Editar Alumno)	65
Tabla 27 Casos de prueba para clases validad (Editar Alumno).....	66
Tabla 28 Casos de prueba para clases inválida.....	67
Tabla 29 Matriz de casos de prueba (Editar Alumno)	68
Tabla 30 Identificación de las clases de equivalencias:(Eliminar Alumno).....	69
Tabla 31 Caso de prueba para clases válidas	69
Tabla 32 Caso de uso para calses invalidas	70
Tabla 33 Matriz de casos de prueba (Eliminar Alumno).....	70

Tabla 34 Identificación de las clases de equivalencias:(Listar Alumno)	71
Tabla 35 Caso de prueba para clases válidas	71
Tabla 36 Caso de uso para calses invalidas	71
Tabla 37 Matriz de casos de prueba (Listar Alumno)	72
Tabla 38 Prueba Autenticación de usuario.....	72
Tabla 39 Prueba Gestión de usuario.....	72
Tabla 40 Prueba gestionar matriculación estudiantil	73

ÍNDICE DE ILUSTRACIONES

Ilustración 1 Componentes de Caso de Uso	23
Ilustración 2 Diagrama de caso de uso del negocio	23
Ilustración 3 Diagrama de actividades- Registro de matrículas de estudiantes.	27
Ilustración 4 Diagrama casos de uso del sistema	32
Ilustración 5 Diagrama casos de uso del sistema – Gestionar Usuario.....	32
Ilustración 6 Diagrama casos de uso del sistema - Gestionar matrícual	32
Ilustración 7 Diagrama casos de uso del sistema - Generar Reporte	33
Ilustración 8 Relación entre los módulos de MVC	39
Ilustración 9 Diagrama de clases de diseño (Gestionar registro de estudiante).....	41
Ilustración 10 Diagrama de secuencia, Registrar Alumno	44
Ilustración 11 Diagrama de secuencia, Actualizar Alumno.....	46
Ilustración 12 Diagrama de secuencia, Eliminar Alumno	48
Ilustración 13 Diagrama de secuencia, Listar Alumno.....	50
Ilustración 14 Diagrama de componentes	52
Ilustración 15 Diagrama de despliegue	53
Ilustración 16 Método de estimación	60
Ilustración 17 Validación Login – Mensaje de error al ingresar	73
Ilustración 18 Mensaje de campos obligatorios creación de usuario.....	74

RESUMEN

La presente investigación propone el diseño y desarrollo de un sistema web para la gestión de matrículas estudiantiles en la Unidad Educativa Patricio Espinoza Bermeo, con el objetivo de modernizar sus procesos administrativos. La institución enfrenta desafíos derivados de la gestión manual de información, que generan retrasos, duplicidades y riesgos en la integridad de los datos.

Para ello, se implementó un sistema utilizando Flutter y Firebase, bajo el enfoque arquitectónico Modelo-Vista-Controlador (MVC) y la metodología tradicional RUP, que garantizan un desarrollo estructurado, seguro y escalable. El sistema permite realizar registros, modificaciones, eliminaciones, búsquedas y listados de estudiantes, con un acceso diferenciado según el rol del usuario.

Se priorizó una interfaz intuitiva, accesibilidad multiplataforma, validaciones de datos y medidas de seguridad como autenticación cifrada y control de accesos. Las pruebas realizadas (caja negra) demostraron su funcionalidad adecuada y facilidad de uso. La estimación de esfuerzo fue de 354.93 horas hombre, representando un ahorro significativo al no requerir contratación externa.

Palabras clave:

- Sistema web
- Matriculación estudiantil
- Flutter
- Firebase
- RUP
- MVC
- Automatización
- Educación
- Pruebas de caja negra.

ABSTRACT

This research proposes the design and development of a web-based system for managing student enrollment at the Patricio Espinoza Bermeo Educational Unit, with the goal of modernizing its administrative processes. The institution faces challenges stemming from manual information management, which generates delays, duplications, and risks to data integrity.

To this end, a system was implemented using Flutter and Firebase, under the Model-View-Controller (MVC) architectural approach and the traditional RUP methodology, which guarantees structured, secure, and scalable development. The system allows for registrations, modifications, deletions, searches, and listings of students, with differentiated access based on user roles.

Priority was given to an intuitive interface, multiplatform accessibility, data validation, and security measures such as encrypted authentication and access control. Black-box testing demonstrated its adequate functionality and ease of use. The estimated effort was 354.93 man-hours, representing significant savings by not requiring outsourcing. Keywords:

- Web system
- Student enrollment
- Flutter
- Firebase
- RUP
- MVC
- Automation
- Education
- Black-box testing

INTRODUCCIÓN

La educación en Ecuador ha sido un pilar clave para el desarrollo social y económico del país, influyendo directamente en la reducción de desigualdades y en la formación de ciudadanos capaces de afrontar desafíos. Durante varios años el sistema educativo ha ido experimentando diversas reformas teniendo como objetivo mejorar la calidad de la enseñanza, ampliar la cobertura y garantizar una educación inclusiva para todos.

Las instituciones educativas desempeñan un papel fundamental en el Ecuador. Su objetivo principal es promover el desarrollo integral de los niños para prepararlos como adultos responsables e integrados en la sociedad. Estas instituciones proporcionan orientación y enseñanzas basadas en conocimientos científicos. Por ende, un centro educativo es un espacio dedicado a la enseñanza y al refuerzo de los valores adquiridos en el entorno familiar.

Para (Mayorga Ases, Sánchez Manobanda, Páliz Ibarra, & Melo Fiallos, 2023) la gestión educativa y el desempeño docente es un papel fundamental para mejorar el ambiente de trabajo en los centros educativos. Por una parte, la gestión educativa es necesaria para que mejore el desempeño docente y brindar conocimientos de alto nivel a los estudiantes. Esto involucra aspectos como la planificación, organización, dirección y control de recursos y procesos. En Ecuador es una prioridad de gobierno buscar la calidad del servicio de educación pública.

En Ecuador los centros educativos no solo cumplen la función de impartir conocimientos, sino que también desempeñan un rol fundamental en la formación integral de los estudiantes, promoviendo valores, habilidades sociales y competencias para la vida. Mediante varias estrategias pedagógicas, programas de inclusión, las instituciones educativas tienen como objetivo garantizar una educación equitativa de calidad para todos. Sin embargo, el sistema educativo enfrenta desafíos relacionados con la infraestructura, las capacitaciones de docentes y la adopción curricular, lo que resalta la importancia de fortalecer políticas que fomenten la equidad y el acceso a una educación efectiva.

En este sentido los centros educativos son espacios donde se adquiere conocimientos de diferentes áreas académicas. Como lo fundamentan (Hernández Pico & Samada Grasst, 2021), la diversidad constituye una realidad en los centros

educativos que ha de ser atendida por los docentes, además pretende eliminar las barreras de aprendizaje, armonizando la respuesta a las necesidades educativas del alumnado, con la consecución de los objetivos de cada una de las etapas educativas. Sin embargo, la respuesta educativa a todo el alumnado se concibe a partir del principio de inclusión, entendiendo que únicamente de ese modo se garantiza su desarrollo, se favorece la equidad y se contribuye a una mayor cohesión social.

Por lo tanto, para entender la importancia de una gestión escolar eficiente, se debe analizar como la administración de los procesos educativos influye en la calidad del aprendizaje y en el desarrollo institucional. Una estructura bien organizada permitirá asignar funciones de manera efectiva, optimizar los recursos y mejorar la toma de decisiones. Con esto, gestionar la información de forma eficiente resulta clave para el buen funcionamiento de los centros educativos, ya que contribuye a la planificación, el monitoreo y la evaluación de los procesos académicos y administrativos.

Sin embargo, a pesar de la importancia de la gestión escolar eficiente, muchas instituciones educativas enfrentan dificultades en la administración de la información, lo que puede afectar la fluidez de los procesos académicos y administrativos.

Por ello, resulta necesario analizar casos específicos, como el de la unidad educativa “Patricio Espinoza Bermeo”, donde los procesos de enseñanza y de aprendizaje está desarrollada conforme a los lineamientos establecidos por el Ministerio de Educación para las instituciones educativas. Se ha detectado que los procesos educativos llevan tiempos prolongados en gestionar la información. Por esto el propósito de esta investigación es la optimización y agilización de dichos procesos, con el fin de brindar un servicio factible, más rápido y sobre todo confiable.

La unidad educativa “Patricio Espinoza Bermeo” está ubicada en la parroquia de Marcopamba La Raya, sector de Chilibulo. En la actualidad esta institución consta con alrededor de 260 estudiantes que provienen de distintas zonas de la ciudad de Quito, a quienes se les asegura una educación con excelencia en los ámbitos institucionales, administrativos y académicos.

Situación problemática

Dentro de los procesos administrativos de la institución, se presenta procesos relevantes relacionados con la organización de los documentos. Como lo es el registro

de matrículas de estudiantes que consiste en recopilar la información de los postulantes a la institución. Los aspirantes completan una ficha personal, que se utiliza para actualizar un archivo en Excel donde se registra a los estudiantes inscritos. Este documento en Excel se archiva para su posterior aprobación. Una vez aprobada la solicitud, se procede a inscribir al estudiante y elaborar el documento de inscripción o ficha de inscripción.

El manejo de la información se realiza con el apoyo de las herramientas de Microsoft como lo son el Word y Excel. Por lo que el proceso no se nota afectado por la multiplicidad de fuentes de datos, por lo que dificultará y ralentizará la búsqueda de la información. La información se encuentra archivada en carpetas físicas y en documentos digitales con un gran volumen de documentos. Donde se utiliza archivadores para mantener en buen estado y evitar el deterioro de la información. Esto implica que el espacio de almacenamiento garantice una condición óptima, lo que va ayudar a evitar la pérdida de información y asegurar su integridad a largo plazo. Bajo estas condiciones, para elaborar reportes hay que revisar la información varias veces con el objetivo de no tener ningún inconveniente ni errores. Aun así, se ha dado casos de duplicados u omisiones de información.

También, se ha dado quejas de parte de los padres de familia, los docentes de la institución e incluso de los estudiantes, debido a la baja eficiencia de los procesos dentro de la institución, ya que se presentan problemas a menudo que afectan el funcionamiento, como lo es la demora en la atención del personal administrativo, tiempos prolongados para la matriculación de su alumno que resulta en filas largas de padres y tiempo prolongado de atención, así como pérdidas de información. De igual manera los docentes manifiestan su inconformidad al tener que solicitar la información de algún estudiante con varios días de antelación, lo que provoca demora en el registro de matrícula de un estudiante o en la elaboración de un reporte sobre el estudiante.

Además, la búsqueda de datos para su verificación resulta ser un proceso lento y tedioso. Esto retrasa en la obtención de información y acrecienta el descontento de los padres de familia durante las reuniones de entrega de reportes y fichas de matrículas. Los representantes señalan que los retrasos son notables y, en ocasiones, la información proporcionada no es correcta.

Así en la actualidad, vivimos en un área caracterizada por un acelerado ritmo de cambio y avances, donde la tecnología se ha convertido en un eje fundamental para el desarrollo de diversos centros educativos. Este fenómeno global de digitalización ha dado lugar a nuevos paradigmas en la gestión de recursos y en la manera de realizar tareas manuales.

Unos de los aspectos más destacados de esta transformación es el impacto de la tecnología ha tenido en la educación y en la gestión de la información dentro de los centros educativos. Las herramientas digitales, como sistemas de gestión de datos, plataformas de aprendizaje, se han integrado dentro de los procesos académicos. Este entorno digitalizado permite un acceso más rápido y eficiente a la información, lo cual es esencial para los estudiantes y docentes, así como para los procesos administrativos.

No obstante, a pesar de las ventajas que ofrece estas tecnologías, algunas instituciones aun enfrentan desafíos significativos en su implementación. La falta de capacitaciones adecuadas, una infraestructura limitada y la falta de compatibilidad entre diversos sistemas dificultan el buen uso de los recursos digitales. Dando, así como consecuencia, el manejo de la información sigue siendo un proceso que se requiere de buena atención, ya que su adecuada administración es fundamental para mejorar la calidad educativa y facilitar la toma de decisiones dentro de la institución.

Es en este contexto, los avances tecnológicos no solo influyen en la manera en que se trasmite y se organiza el conocimiento, sino que también han revolucionado los métodos tradicionales de gestión administrativa. La optimización de procesos mediante el uso de tecnologías se ha vuelto una necesidad para mejorar la eficiencia en todos los niveles. La incorporación de nuevas herramientas no solo mejora la rapidez con la que se manejan los datos, sino que también tiene el potencial de reducir los errores humanos, y así promover un entorno educativo más equitativo y accesible.

Así (Navarrete Mendieta & Mendieta García, 2020) fundamenta que en la actualidad estamos inmerso en un desarrollo creciente de las nuevas tecnologías que brindan en el estudiante actual un conocimiento globalizado de las nuevas tendencias del mercado, la economía y los negocios en sí, la educación debe asumir una redefinición en cuanto su metodología y la adaptabilidad de ella al mundo actual.

Por ello en la actualidad, la sociedad experimenta transformaciones significativas

impulsadas por los avances tecnológicos e innovación, los cuales desempeñan un papel crucial en el desempeño humano. La constante evolución de la tecnología ha generado un impacto considerable en que las personas realizan sus actividades laborales, especialmente en el manejo de la información. En este contexto, la tecnología se ha consolidado como un recurso estratégico para las empresas, al facilitar la optimización de procesos y resultados, a lo que Martha, Isolina y Mario detallan que:

La innovación educativa se define como un proceso intencional y planificado que busca transformar y mejorar los métodos, contenidos y estructuras del sistema educativo para optimizar el aprendizaje de los estudiantes y elevar la calidad educativa. Este concepto implica la introducción de cambios significativos en las prácticas pedagógicas, la incorporación de nuevas tecnologías y la adaptación a las necesidades cambiantes de la sociedad. Según un artículo publicado en 2020, la innovación educativa "es un proceso que tiene que responder a las necesidades presentes en el ámbito educativo, de modo que debe ser eficaz y eficiente, sostenible y con resultados transferibles fuera del contexto de donde surgen. (Zavala, González, & Vázquez, 2020)

Así como las instituciones educativas tienen que enfrentar varios desafíos en la gestión de procesos administrativos, debido al incremento en la demanda de sus servicios y la necesidad de adaptarse a un cambio dentro de un entorno más digitalizado. Uno de los procesos más críticos es la matriculación de estudiantes, que involucra recolección, organización y actualización de información académica y administrativa de los estudiantes.

Formulación del problema

¿Cómo facilitar la gestión de la información relacionada con los procesos de registro de matrículas estudiantiles en la unidad educativa Patricio Espinoza Bermeo?

Objetivo

Objetivo General

Desarrollar un sistema web que mejore la gestión de la información relacionada con los procesos de registro de matrículas estudiantiles en la unidad educativa Patricio Espinoza Bermeo, usando herramientas de código abierto multiplataforma como Flutter y Fierebase que garantices la integridad de la información.

Objetivos Específicos

- Analizar el proceso de matrículas para identificar deficiencias y oportunidades de mejorar la gestión administrativa de la institución.
- Investigar metodologías y herramientas tecnológicas para el desarrollo e implementación del sistema de matriculación estudiantil.
- Diseñar un sistema de matriculación estudiantil que resuelva las necesidades dadas por la institución, garantizando el funcionamiento para la recolección y almacenamiento de información de forma segura.
- Desarrollar el sistema web de matriculación en la escuela Patricio Espinoza Bermeo, y de esta manera dar capacitación al personal administrativo del correcto uso y funcionamiento.
- Describir el análisis de costos relacionado con el desarrollo del sistema web a través del método de Puntos Casos de Uso.
- Validar el sistema con pruebas de caja negra, evaluando las respuestas de las interfaces.

Hipótesis

El desarrollo de un sistema web mejora la gestión de la información relacionada con los procesos de registro de matrículas estudiantiles en la unidad educativa Patricio Espinoza Bermeo, usando herramientas de código abierto multiplataforma como Flutter y Fierebase que garantices la integridad de la información.

Justificación de la investigación

Para la Unidad Educativa Patricio Espinoza Bermeo, es fundamental tener un sistema web que les facilite la gestión de matrículas estudiantiles. Esto permitirá manejar los procesos de forma adecuada y organizada, garantizando la generación de reportes confiables de manera más rápida y eficiente.

Al almacenar la información en la base de datos del sistema, se garantiza una correcta persistencia de los datos. Esto optimiza los registros, procesos y actividades, mejorando tanto la calidad como la precisión de la información, ya que esta será validada automáticamente al momento de su registro. Además, el trabajo del personal administrativo se verá reducido, permitiendo que puedan dedicarse a otras tareas relacionadas con el funcionamiento de la institución.

El desarrollo de un sistema web para la gestión de matrículas estudiantiles se plantea como una respuesta directa a estas necesidades. Este sistema estará diseñado para agilizar tareas clave, como la matriculación de estudiantes, dando así la edición y actualización de datos, teniendo un almacenamiento seguro de datos académicos y administrativos. También se tendrá una plataforma amigable para los padres de familia, estudiantes y personal administrativo, la cual permitirá agilizar los procesos y mejorar la comunicación institucional.

La implementación de un sistema web no solo sirve para optimizar los procesos internos, sino también para contribuir a su modernización y la capacidad de adaptarse a un entorno tecnológico que está en constante cambio.

Este proyecto no tiene como objetivo principal mejorar los procesos administrativos de la Escuela “Patricio Espinoza Bermeo”, sino también crear un modelo de gestión donde se pueda replicar en otras instituciones educativas con las mismas necesidades. De esta manera, el presente proyecto se destaca en un esfuerzo por contribuir al desarrollo de soluciones tecnológicas accesibles, que promuevan la mejora continua en el sector educativo.

Tipo de investigación

De acuerdo al proyecto que se desarrollará el tipo de investigación que se realizará será aplicada, y las estrategias de estudio serán bibliográfica y de campo.

Investigación Aplicada

La investigación aplicada es aquella que se lleva a cabo con el objetivo de resolver un problema puntual. Este tipo de investigación se centra en el proceso, utilizando tanto la teoría como los resultados obtenidos para abordar dichas problemáticas. (Kiss, 2024).

Investigación Bibliográfica

La investigación documental o bibliográfica es una técnica de investigación a través de la cual se explora lo que se ha escrito y publicado antes acerca un determinado tema. (Zorrilla, 2021).

Investigación de campo

El objetivo específico de una investigación de campo es obtener información directamente de la realidad. Según cuál sea el objeto de estudio, esto se puede

realizar mediante la observación directa, entrevistas, encuestas o recopilación de datos primarios. (Kiss, 2024).

Estructuración del documento

El capítulo I presenta el marco teórico, donde se analizan los antecedentes de la investigación, incluyendo sistemas informáticos relacionados con el área de estudio. Además, se describen las herramientas tecnológicas y metodológicas que se aplicaran durante el desarrollo del sistema web.

El capítulo II aborda el análisis del modelo de negocio detallando las reglas y especificaciones necesarias. En esta sección, se establecen los requerimientos funcionales y no funcionales que servirán como base para la solución del problema.

El capítulo III se centra en el desarrollo del software, apoyándose en la metodología seleccionada. Se incluye diagramas de clases, diseño físico y lógico, así como la estructura de la base de datos, mostrando las relaciones necesarias para la construcción del sistema.

El capítulo IV está dedicado a la validación del software, mediante la aplicación de pruebas, como la caja negra y evaluaciones técnicas. También se presentan los resultados obtenidos y la aceptación del sistema por parte de los usuarios.

CAPÍTULO I

1. FUNDAMENTACIÓN TEÓRICA.

En este capítulo se abordará toda la problemática relacionada con la gestión de matrículas estudiantiles y también examinará los antecedentes que involucra este proceso. Existirá una comparación de sistemas similares al tema de investigación. Además, se entrará en detalle con las tecnologías, herramientas y lenguajes de programación que se empleará en el desarrollo para la solución de esta propuesta.

1.1. Conceptos asociados a la investigación

Registro de estudiantes: Proceso que comprende la recolección, validación, almacenamiento y el mapeo seguro de la información personal, académica y administrativa de los estudiantes. Este procedimiento abarca la matrícula, revisión de información, récord académico, manejo de matrículas con el objetivo de facilitar la organización institucional.

Notificaciones y Seguimiento: Sistema físico de documentación y control implementado mediante un libro de reporte, que se encarga de reportar fechas importantes, avance del alumno, sucesos y avisos relevantes.

Gestiona Académica: Conjunto de procesos administrativos y estructurados diseñados para planificar, organizar y supervisar las tareas académicas de una institución, como horarios, cursos y distribución de profesores.

Análisis de datos educativos: Proceso de recopilación y evaluación de datos generados en la interacción de los estudiantes con los sistemas educativos, con el fin de optimizar la toma de decisiones y mejorar el rendimiento académico.

1.2. Objetivos estratégicos de la organización en estudio.

1.2.1. Misión

La institución educativa PATRICIO ESPINOSA BERMEO, forma estudiantes desde los niveles de Inicial I y II, Preparatoria, Elemental y Media. La misión de la institución es desarrollar competencias básicas, formando estudiantes reflexivos, críticos que aportan en la sociedad, fomentando la formación integral a través de una metodología activa y colaborativa con la comunidad educativa. (Escuelas Ecuador, 2023)

1.2.2. Visión

La visión de la IE “XX” desarrollará procesos de enseñanza y aprendizaje con un enfoque constructivista que formará estudiantes reflexivos, críticos en beneficio de su entorno social y natural; seguros de sí mismo, con alta autoestima y comprometidos con su aprendizaje; impulsando que los docentes apliquen metodologías que fomentan la lectura, el trabajo cooperativo en vinculación con la comunidad educativa a fin de buscar soluciones oportunas con la participación de padres de familia como agentes responsables, colaborativos que se involucren con los objetivos y metas de la institución. (Escuelas Ecuador, 2023)

1.3. Objeto de estudio

Procesos requeridos al registro de matrículas de estudiantes en las unidades educativas del Ecuador.

1.3.1. Campo de acción

Sistemas informatizados orientados a la gestión del registro de matrículas estudiantiles en unidades educativas en Ecuador.

1.3.2. Flujo actual de proceso

Análisis crítico de la ejecución de los procesos

Los procesos dentro de cualquier institución educativa en el Ecuador siguen un orden secuencial y requieren el cumplimiento de procedimientos específicos, como es el caso del registro de matrículas estudiantiles. Sin embargo, en muchas instituciones se han identificado dificultades en la gestión de la información, que impactan en la eficiencia de los procesos.

Específicamente, en la institución Patricio Espinoza Bermeo frecuentemente, se presentan casos de duplicidad, ausencia o redacción deficiente de la información, especialmente al utilizar hojas de cálculo en Excel y el Word, esto afecta la calidad de reportes, que no reflejan con exactitud la realidad de la información. Además, se han registrado pérdidas de datos debido a fallos técnicos en los equipos de cómputo y físicos donde se almacena la información. Por otro lado, los padres de familia experimentan tiempos de espera prolongados para realizar la matrícula de sus hijos, lo que retrasa la atención y alarga innecesariamente el proceso de matriculación del estudiante, afectando también la jornada laboral de los padres de familia.

La inserción y manipulación de datos implican largas jornadas de trabajo para el área de administración, dando como resultado que la generación de reportes no sea inmediata, lo que dificulta la toma de decisiones oportuna. Esta falta de agilidad en la extracción de información genera insatisfacción y afecta el desempeño institucional.

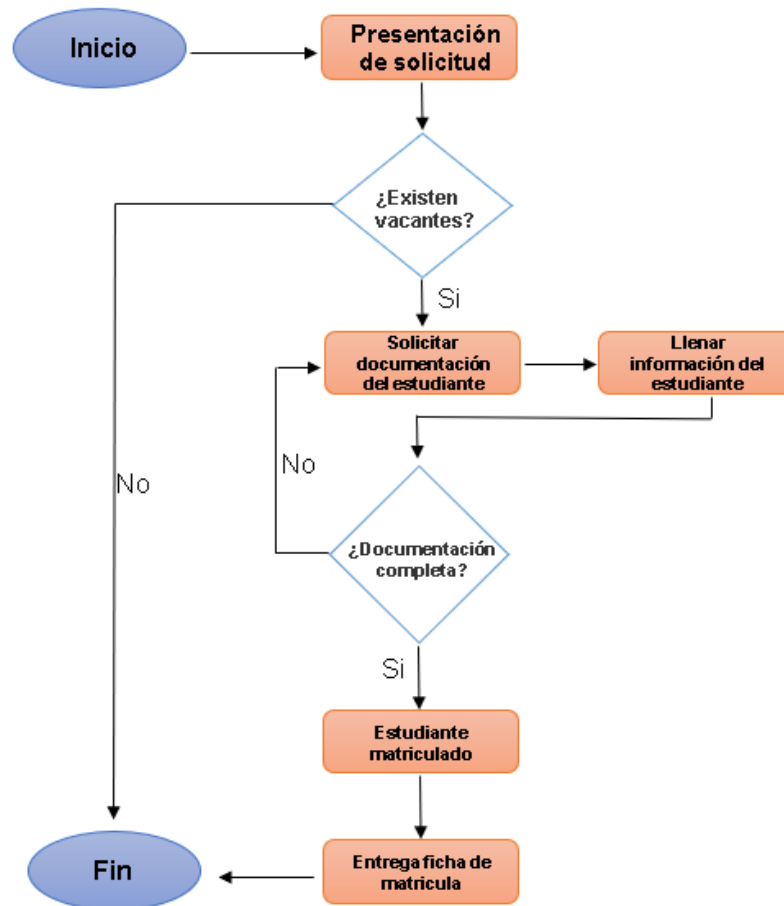
Por otro lado, la gestión de documentos en la institución presenta deficiencias significativas debido a la falta de control por parte del área administrativa. La ausencia de un sistema organizado y centralizado no solo ha hecho que incremente el riesgo de pérdida, deterioro o inaccesibilidad de los registros académicos, sino que también favorece la acumulación desordenada de documentos históricos. Esta situación dificulta la búsqueda de la información de estudiantes que ya no forman parte de la institución, retrasa los procesos administrativos y aumenta la posibilidad de que existan errores al tratar la documentación.

Procesos de estudio

El registro de matrículas en la institución es importante ya que es el que permite llevar el control ordenado y preciso de los estudiantes inscritos. Esta información es esencial ya que a través de ella se realiza planificaciones académicas, asignación de cursos y procesos administrativos para el estudiante. Además, garantiza la legalidad del proceso de admisión y facilita un seguimiento óptimo sobre un estudiante en específico.

Contar con un registro de matrículas bien establecido, es clave para un buen funcionamiento en la institución educativa, ya que permite tener organizado de manera eficiente los cursos con sus respectivos estudiantes, horarios, materias, así como mantener actualizada la base de datos estudiantil de la institución.

Figura 1 Proceso de matriculación



En este diagrama de flujo se describe el proceso de matriculación de un estudiante asegurando que cumpla con todos los requisitos necesarios para el registro. El proceso inicia con presentar una solicitud, seguida de una verificación de vacantes, si hay campos, se le solicitara la documentación completa del estudiante para poder llenar el registro de la matrícula, dada esta información se verificara si está completa o de lo contrario si hace falta algún documento. Una vez validada la información, el estudiante queda matriculado y recibirá su ficha de matrícula, finalizando el proceso.

1.4. Herramientas y tecnologías

1.4.1. Sistemas automatizados existentes relacionados con el campo de acción.

Múltiples son las instituciones que se han dado la tarea de informatizar sus áreas y que han logrado agilizar los procesos internos como la matriculación y otros.

Una muestra de ellos es el

1.4.1.1. Sistema de Matriculación y carnetización en la “Unidad Educativa Pablo Picasso”

Este sistema de matriculación y carnetización en la “Unidad Educativa Pablo Picasso” es un software implementado por (Salazar Acosta, 2018). Fue creado con el propósito de automatizar procesos que se llevan a cabo en la Unidad Educativa “Pablo Picasso”, ubicada en la ciudadela “El Recreo”, cantón Durán. Dichos procesos realizados de forma manual necesitaban periodos extensos para su registro, además de la pérdida de información y falta de seguridad, con todo esto, se requería de forma urgente una pronta solución. El módulo de matriculación de la Unidad Educativa “Pablo Picasso”, contiene los procesos de gestión de la aplicación (mantenimientos), seguridad, carnetización, reportes y matriculación, desarrollados en lenguaje PHP. Además, se utiliza JavaScript para manejar los efectos dinámicos del sistema.

Este sistema demuestra que la automatización del proceso de registro de estudiantes optimiza tiempos, reduce la pérdida de información y fortalece la seguridad de los datos, por lo que, en base a esta experiencia, la propuesta para el sistema web en la institución Patricio Espinoza Bermeo busca implementar un sistema que garantice la integridad y disponibilidad de los registros en la nube y de esta manera poder solventar con el riesgo asociados a los sistemas locales tradicionales.

1.4.1.2. Sistema de matriculación y gestión de calificaciones para la “Unidad Educativa Ati li Pillahuaso”

Este sistema de matriculación fue desarrollado por (Rojas Borbor & Tandazo Luna, 2019) con el objetivo de mejorar el flujo de actividades y ayude en la verificación de la información, con un sistema web para la generación de matrículas y calificaciones, de esta manera lograr optimizar la entrega de reportes e información. Este sistema web automatiza todo el proceso de inscripción de los alumnos, y también el proceso de calificaciones.

Esto evidencia que la implementación de un sistema web da como resultado una optimización y flujo de actividades, automatiza los procesos de matriculación y facilita la generación de reportes. Este enfoque respalda la propuesta para la unidad educativa Patricio Espinoza Bermeo con Flutter y Firebase, teniendo como objetivo la

mejora de la gestión de matrículas y asegurando la disponibilidad de los registros en tiempo real.

1.4.1.3. Sistema informático para el registro de matrículas de estudiantes, cobro de pensiones y pago de sueldos a docentes en la Unidad Educativa Louis Víctor de Broglie.

El sistema de (Landi Paucar, 2019) propone una implementación de un sistema automatizado para los procesos administrativos como es el caso de la matriculación estudiantil, cobro de pensiones y pago de sueldos a los docentes de la Unidad Educativa Louis Víctor de Broglie, este sistema mejora la rapidez y la organización del flujo de trabajo, reduciendo tiempos de espera y facilitando el acceso a la información, este sistema está desarrollado en Java con un gestor de base de datos como lo es MariaDB.

Este sistema aporta en la propuesta para la unidad educativa Patricio Espinoza Bermeo, ya que demuestra una automatización en el proceso de matriculación, mejora la rapidez, organización y acceso a la información, reduciendo los tiempos de espera.

1.4.2. Comparación de sistemas informáticos existentes

Los requisitos de la unidad educativa “Patricio Espinoza Bermeo” se detalla en la tabla 1, La adquisición de cualquier de estos requisitos, según lo especificado en la tabla garantiza una solución integral que satisface las necesidades del cliente.

Tabla 1 Comparación de sistemas informáticos existentes

Característica	UNIDAD EDUCATIVA PABLO PICASSO	UNIDAD EDUCATIVA ATI II PILLAHUASO	UNIDAD EDUCATIVA LOUIS VÍCTOR DE BROGLIE
Funcionalidades	Sistema de matriculación, y de carnets	Sistema de matriculación y gestión de calificaciones	Registro de matrículas, cobro de pensiones, pago de sueldos a docentes.
Software	PHP, JavaScript	PHP, JavaScript, Laravel	Java
Gestor de base de datos	MariaDB	PostgreSQL	MariaDB

Arquitectura	MVC	MVC	MVC
Metodología de desarrollo	Proceso Unificado Racional RUP.	Metodologías Ágiles, Scrum Framework	Proceso Unificado Racional RUP.

La institución para la cual se va a desarrollar el sistema, en su plan de informatización, no cuenta con recursos económicos suficientes para adquirir soluciones tecnológicas existentes como lo son el caso de los sistemas anteriormente estudiados. Esto se debe a que la institución no cobra pensiones, no es autofinanciada y atiende a estudiantes de bajos recursos económicos. Dicho esto, se ha optado por reutilizar ideas de sistemas estudiados anteriormente, como la matriculación online de estudiantes, adaptándolas a las necesidades específicas de la institución.

Cabe recalcar que estas propuestas consideradas han sido adecuadas exclusivamente para la operación de esta institución. Además, destacar que es una solución de desarrollo de manera desinteresada, teniendo en cuenta que esos sistemas no son comerciales ni reutilizables para otras instituciones debido a que responden a necesidades particulares que no aplican de forma general para otras instituciones. El nuevo sistema busca dar un aporte para los procesos académicos de la institución Patricio Espinoza Bermeo, cumpliendo con los objetivos y necesidades institucionales sin comprometer recursos financieros.

Por esta razón, se propone desarrollar una nueva solución enfocada en la gestión de matrículas, con el objetivo de reducir tiempos de trabajo, tiempos de búsqueda de información y asegurar que sea solventadas las necesidades de la institución. Esta propuesta se basará también en la arquitectura Modelo – Vista – Controlador (MVC), que nos dan como solución los sistemas anteriormente nombrados y se aplicara una metodología RUP para guiar el proceso de desarrollo de manera estructurada, aprovechando las buenas prácticas implementadas en los sistemas nombrados en la tabla y de esta manera asegurar una integración coherente para nuestra institución educativa.

1.5. Tendencias y tecnologías actuales.

La metodología es la disciplina que estudia el conjunto de técnicas o métodos que se usan en las investigaciones científicas para alcanzar

los objetivos planteados. Todos los métodos de investigación deben seguir una metodología, que se vale de una teoría normativa, descriptiva y comparativa acerca del método, sumado al proceder del investigador. (Editorial Etecé, 2024)

Seleccionar la metodología que más se semeje a las necesidades de la institución educativa la cual nos provea de pautas específicas para la planeación, ejecución e inspección del proyecto.

Tabla 2 Comparación de Metodologías

Metodología Tradicional	Metodología Ágil
Resistencia a los cambios	Preparados por cambios
Impuestos por el equipo	Impuestas externamente
Arquitectura esencial expresada mediante modelos	Menos énfasis en la arquitectura del software
Más roles	Pocos roles
Más artefactos	Pocos artefactos
Grupo grandes y distribuidos	Grupos pequeños, en el mismo sitio
Procesos controlados, con muchas normas y políticas	Procesos menos controlados, con pocos principios
Proceso rígido	Proceso flexible con adaptación
El cliente interactúa con el equipo de desarrollo	El cliente es parte del equipo de desarrollo
Basada en normas provenientes de estándares seguidos por el entorno de desarrollo	Basadas en heurísticas provenientes de prácticas de producción de código

Fuente: (Kuz, Falco, & Giandini, 2018)

1.6. Fundamentación de la metodología utilizada

Tras examinar todos los elementos, se optó por emplear una metodología tradicional debido a su enfoque estructurado y sistemático, que se adapta a la necesidad de la institución. En especial RUP, se centra en los procedimientos y etapas claras del desarrollo, lo que garantiza una planificación precisa y un control riguroso de cada fase. Esta metodología da como prioridad la organización y proporciona un marco robusto para para gestionar los riesgos, y de esta manera asegurar que todos los requisitos de la propuesta sean cubiertos de manera correcta. En este contexto, RUP es la opción adecuada para la implementación ordenada del sistema en la institución y satisfacer las necesidades.

1.6.1. Proceso Unificado Racional RUP.

La metodología RUP consiste en una estructura de trabajo de proceso con el objetivo del producto y por tanto basada en el modelo Unified Modeling Language (UML), cuando se habla de programación orientada a objetos. El UML

compone un lenguaje para definir una secuencia de artefactos y ayudar en la ejecución de las tareas del sistema a desarrollar, a través de diferentes tipos de diagramas. (Barrientos Rodríguez, y otros, 2022)

RUP es un sistema adaptable para crear software, creado para ajustarse a las demandas de las empresas y grupos de trabajo. Facilita la elección y la modificación de componentes del proceso de acuerdo con las necesidades particulares de cada proyecto.

Una de sus ventajas es que este proceso es de los más utilizados para el desarrollo del software por la mayoría de las empresas, pues su enfoque es bastante optimo y tiende a ser una metodología viable para la mayoría de estas.

1.6.2. Metodología de trabajo basada en RUP.

En la siguiente tabla se describe la metodología de trabajo RUP que guiará el proceso de desarrollo de este trabajo.

Figura 2 : Metodología de trabajo RUP

DISCIPLINA (PROCESO)	FASES							
	INICIO		ELABORACIÓN		CONSTRUCCIÓN		TRANSICIÓN	
	Actividades	Artefacto	Actividades	Artefacto	Actividades	Artefacto	Actividades	Artefacto
MODELADO DEL NEGOCIO	Elaborar casos de uso del negocio	Casos de uso del negocio			Emplear casos de uso del negocio	Casos de uso del negocio		
	Realizar diagrama de actividades							
REQUERIMIENTOS	Establecer requerimientos funcionales		Desarrollar casos de uso del sistema		Emplear casos de uso del sistema	Casos de uso del sistema		
ANALISIS Y DISEÑO			Elaborar la arquitectura del sistema	Diagrama MVC				
			Realizar diagrama de clases	Diagrama de clases				
IMPLEMENTACIÓN			Crear diagrama de secuencias		Emplear diagrama de secuencias	Diagrama de secuencias		
			Crear diagrama de componentes		Emplear diagrama de componentes	Diagrama de componentes		
PRUEBAS					Preparar pruebas y análisis de la factibilidad		Mostrar formulario de prueba	Caso de prueba

Fuente: (Landi Paucar, 2019)

1.6.3. Enterprise Architect

Es una herramienta CASE para diseñar y construir sistemas de software, como lo son diagramas de casos de uso, diagrama de actividades, diagrama de procesos y para propósitos de modelado más generalizados como es el caso del lenguaje de modelado UML. Integra y conecta una amplia gama de información estructural y de

comportamiento, lo que ayuda a construir un modelo arquitectónico coherente y verificable, ya sea lo que es o lo que será. (Systems, Sparx, 2020)

1.6.4. Lenguaje Unificado de Modelado UML

Se utilizará UML como lenguaje de modelado dentro del desarrollo del sistema web de matriculación y de esta manera poder tener una documentación exhaustiva y precisa. Su capacidad para generar diagramas de caso de uso comprensibles permitirá visualizar las interacciones entre los actores del sistema (estudiantes, personal administrativo, docentes). La elaboración de diagramas de clases, secuencias de actividades facilitara la comprensión de la estructura y comportamiento del sistema, al ser un lenguaje de modelado estandarizado y enfocado en la documentación, UML asegura que cada aspecto del sistema de matriculación este detallado, desde los requerimientos iniciales hasta la implementación final, lo resulta fundamental para la claridad y exactitud del proyecto.

En este contexto (Lizcano Bueno , 2016) define a UML como una herramienta que permite modelar software orientado a objetos a través de un amplio vocabulario gráfico enfocado a la representación conceptual y física de los sistemas de software. UML no está limitado al modelado de software.

1.6.5. Visual Studio Code

Es un editor de código para programadores gratuito, de código abierto y multiplataforma. Está desarrollado por Microsoft, una compañía con una dilatada experiencia en la creación de IDEs (entornos de desarrollo integrados), que ha conseguido plasmar su larga tradición en el sector para ofrecer una herramienta ligera y práctica que la comunidad ha adoptado en masa.

Se trata de un editor de código, esto quiere decir, una herramienta que nos permite editar el texto plano de los archivos de código para programación. Otro de los factores de su rápida propagación es su capacidad para admitir una amplia variedad de lenguajes y frameworks, por lo que muchas comunidades lo usan en áreas tan dispares como el front-end o back-end. (García de Zúñiga, 2024).

1.6.6. Lenguaje de programación Dart

Dart es un lenguaje open source desarrollado en Google con el objetivo de permitir a los desarrolladores utilizar un lenguaje orientado a objetos y con análisis

estático de tipo. A diferencia de muchos lenguajes, Dart se diseñó con el objetivo de hacer el proceso de desarrollo lo más cómodo y rápido posible para los desarrolladores. Por eso, viene con un conjunto bastante extenso de herramientas integrado, como su propio gestor de paquetes, varios compiladores i/o transpiladores, un analizador y formateador. (Martínez Cordón, 2021)

1.6.7. Flutter

Flutter es un marco de código abierto desarrollado y compatible con Google. Es compatible principalmente con el desarrollo de aplicaciones móviles, con el desarrollo de aplicaciones en seis plataformas: iOS, Android, web, Windows, MacOS y Linux. (Cristancho, 2022)

En este contexto (Flutter Docs, 2023) aporta que Flutter se ha convertido en una herramienta de desarrollo muy potente gracias a su rendimiento superior y su capacidad para conseguir interfaces y experiencias de usuario muy bien logradas, atractivas y personalizables. Cuenta con el respaldo de Google y de una comunidad sumamente activa, lo cual asegura un soporte continuo y mejoras constantes.

1.6.8. Firebase

Firebase es una plataforma creada por la empresa Google, para la creación de aplicaciones móviles y web, que proporcionan diversos recursos y servicios en la nube. Tienen como objetivo facilitar la creación y gestión de aplicaciones sin necesidad de preocuparse por la infraestructura de los servidores. Entre sus funcionalidades se destaca que incluyen base de datos en tiempo real, cuenta con autenticación de usuario, almacenamiento en la nube, hosting y análisis de rendimiento.

En este contexto (López Mora, 2020) nos aporta que sus herramientas son variadas y de fácil uso, considerando que su agrupación simplifica las tareas de gestión a una misma plataforma. Las finalidades de las mismas se pueden dividir en cuatro grupos: desarrollo, crecimiento, monetización y análisis. Es especialmente interesante para que los desarrolladores no necesiten dedicarle tanto tiempo al backend, tanto en cuestiones de desarrollo como de mantenimiento.

1.7. Conclusión del capítulo

El análisis de los conceptos claves nos ayudó a establecer el contenido del tema, proporcionando así una visión más clara sobre el desafío a enfrentar. La revisión de los sistemas informáticos similares permitió concluir que si cumplen con las necesidades actuales de nuestra unidad educativa y así poder guiarnos para poder dar una solución óptima.

En este capítulo también se definieron las herramientas, que se va a utilizar para desarrollar el sistema web de matriculación en la institución, los cuales nos van ayudar a resolver el problema de la gestión de la información y de esta manera brindarles una solución más óptima y automatizada para la manipulación de la información.

CAPÍTULO II

2. MODELO DE NEGOCIO, REQUISITOS FUNCIONALES Y MODELOS DEL SISTEMA

En este capítulo se dará a conocer el entorno que se va a automatizar, a través de diagramas de casos de usos de negocio, actores involucrados y diagramas de actividades de negocio. También, se dará a conocer los requisitos funcionales teniendo en cuenta su complejidad y se modela y se describe los casos de usos del sistema.

2.1. Descripción del entorno

La unidad educativa Patricio Espinoza Bermeo está integrada por una comunidad comprometida, padres de familia, estudiantes, docentes y personal administrativo. La misión principal es ofrecer una educación de calidad, aportando al desarrollo académico, personal y social para cada uno de los alumnos de la institución.

Actualmente, los procesos administrativos se desarrollan de manera manual, como es el caso de las matrículas estudiantiles, cuya automatización permitirá brindar una mayor eficiencia y organización, con el propósito de modernizar la gestión de la institución, buscando la implementación de herramientas tecnológicas que faciliten el desarrollo de la misma, dando este beneficio a toda la comunidad de la institución.

2.1.1. Modelo de caso de uso del negocio






Los usuarios del negocio y las actividades del negocio forman parte del diagrama de casos de uso del negocio, como una técnica para capturar requisitos o información de cómo un sistema o negocio trabaja. (Fernández Marín & González Tolmo, 2020)

El modelo de casos de usos del negocio es una técnica que se utiliza en la ingeniería de software y análisis de sistemas, donde se describe como una organización o empresa lleva los procesos del negocio desde una perspectiva de alto nivel. Se identifican actores (clientes, empleados) y las interacciones que tienen con el negocio para cumplir con su objetivo.

Así de esta manera (González Hernández, 2021) nos aporta que el modelo de casos de uso del negocio es un modelo que describe los procesos de un negocio (casos de uso del negocio) y su interacción con elementos externos (actores), tales como socios y clientes, es decir, describe las funciones que el negocio pretende realizar y su objetivo básico es describir cómo el negocio es utilizado por sus clientes y socios. Implica la determinación de los actores y casos de uso del negocio. Con esta actividad se pretende: Identificar los procesos en el negocio, definir las fronteras del negocio que van a modelarse, Definir quién y qué interactuarán con el negocio y crear diagramas del modelo de casos de uso del negocio.

En la siguiente tabla se explica los estereotipos que se utilizaran durante el modelo de negocio:

Ilustración 1 Componentes de Caso de Uso

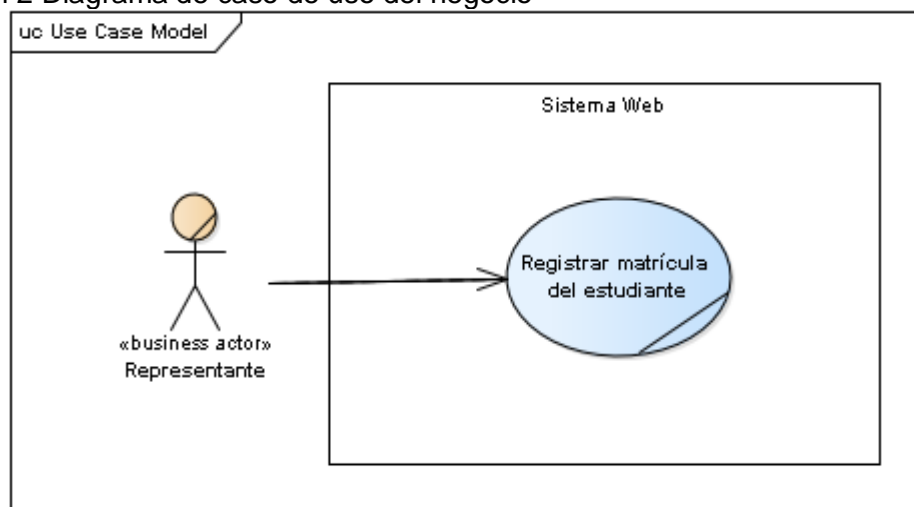
Componentes de un Caso de Uso		
Figura	Nombre	Definición
	Actor	Un actor en un caso de uso se define como una entidad externa que puede ser: personas, sistemas, que interactúan con el sistema para llevar a cabo un proceso o una función específica
	Caso de Uso	Un caso de uso representa las distintas capacidades de un sistema. Cada caso de uso recibe un nombre y una descripción, la cual detalla el inicio y su final. Además, incluye una explicación de la funcionalidad que representan, junto con escenarios complementarios y requerimientos no funcionales.
	Conectores	Los conectores en un caso de uso son relaciones visuales que indican como los actores y los casos de usos interactúan dentro de un sistema. Se representan en los diagramas de casos de uso utilizando líneas y flechas, permiten comprender mejor la estructura y el flujo de las
	Extensión	entre casos de uso en la que uno extiende a otro. Es decir, el caso de uso extendido agrega o modifica el comportamiento de otro caso de uso bajo ciertas condiciones o circunstancias adicionales
	Inclusión	Se utiliza para representar una relacion entre casos de uso en la que un caso de uso incluye el comportamiento de otro caso de uso de manera obligatoria.

Fuente: (Kimmel, 2019)

2.1.2. Diagrama de caso de uso del negocio

En el siguiente diagrama se muestra el caso de usos del negocio relacionado con el actor del negocio.

Ilustración 2 Diagrama de caso de uso del negocio



2.1.3. Actores del negocio

Se definen como actores del negocio a cualquier individuo, entidad, grupo, organización, con los que el negocio va a interactuar.

Tabla 3 Descripción de los actores del Negocio

Descripción de los actores del Negocio	
Actores del negocio	Función
Representante	Padres de familia legales de los estudiantes

Trabajador del negocio

El trabajador del negocio, es el que ejecuta actividades clave dentro de un proceso empresarial, sin ser necesariamente un usuario del sistema. Su rol es fundamental para entender las operaciones que el sistema debe apoyar, ya que proporciona el conocimiento contextual del negocio y sus flujos de trabajo.

Tabla 4 Descripción del trabajador del negocio

Descripción del trabajador del negocio	
Actores del negocio	Función
Secretaria	Encargada de gestionar toda la información relacionada con el estudiante para el registro de matrículas.

Descripción de los casos de uso del negocio

El proceso de matriculación de un nuevo estudiante comienza cuando la secretaria entrega la ficha de registro al representante del estudiante. Esta se llena con información requerida del alumno y luego se devuelve a la secretaria, quien se encarga de validar, revisar y analizar la solicitud para su aprobación. Para completar el registro del nuevo postulante, se verifica la información de la ficha de registro que contiene:

- Numero de cedula
- Casta de servicios básicos de máximo tres meses de expedición
- Fecha de nacimiento
- Ficha de la estudiante proporcionada por la institución
- Récord académico de la escuela

Cuando se realiza la verificación de la información, se procede a ingresar los datos en un archivo de Excel en el cual consta los estudiantes matriculados. Una vez ya registrados se procede a verificar si el estudiante es elegible para ser matriculado en la institución, y si existen cupos para la misma.

Una vez sea el estudiante elegible para la institución se informará el día y la hora de asistencia para su día de clases y se procederá a dar la ficha de matrícula. Este se encuentra en un archivo de Excel en el cual se modifica los datos del nuevo postulante, sin embargo, este método de almacenamiento no garantiza la seguridad de la información, lo que podría representar un riesgo en términos de privacidad y protección de datos.

2.1.4. Diagrama de actividades

Es una herramienta gráfica que representa el flujo de actividades y decisiones en un proceso, permitiendo modelar comportamientos dinámicos y flujos de trabajo tanto en sistemas computacionales como en procesos organizacionales. Según (Siewe & Ngounou, 2025) estos diagramas son fundamentales para describir el comportamiento dinámico de los sistemas, facilitando la comprensión y verificación formal de los procesos modelados. Los diagramas de actividades muestran el flujo de trabajo desde el punto de inicio hasta el punto final detallando muchas de las rutas de decisiones que existen en el progreso de eventos contenidos en la actividad. Estos también pueden usarse para detallar situaciones donde el proceso paralelo puede ocurrir en la ejecución de algunas actividades.

En el desarrollo del sistema web de matriculación estudiantil, se va a identificar los procesos principales que garanticen un flujo eficiente y estructurado de la gestión de inscripciones. Entre los aspectos claves es la matriculación de estudiantes, un proceso que se debe optimizar para satisfacer las necesidades de la institución, de esta manera se definirá los actores con sus respectivas actividades.

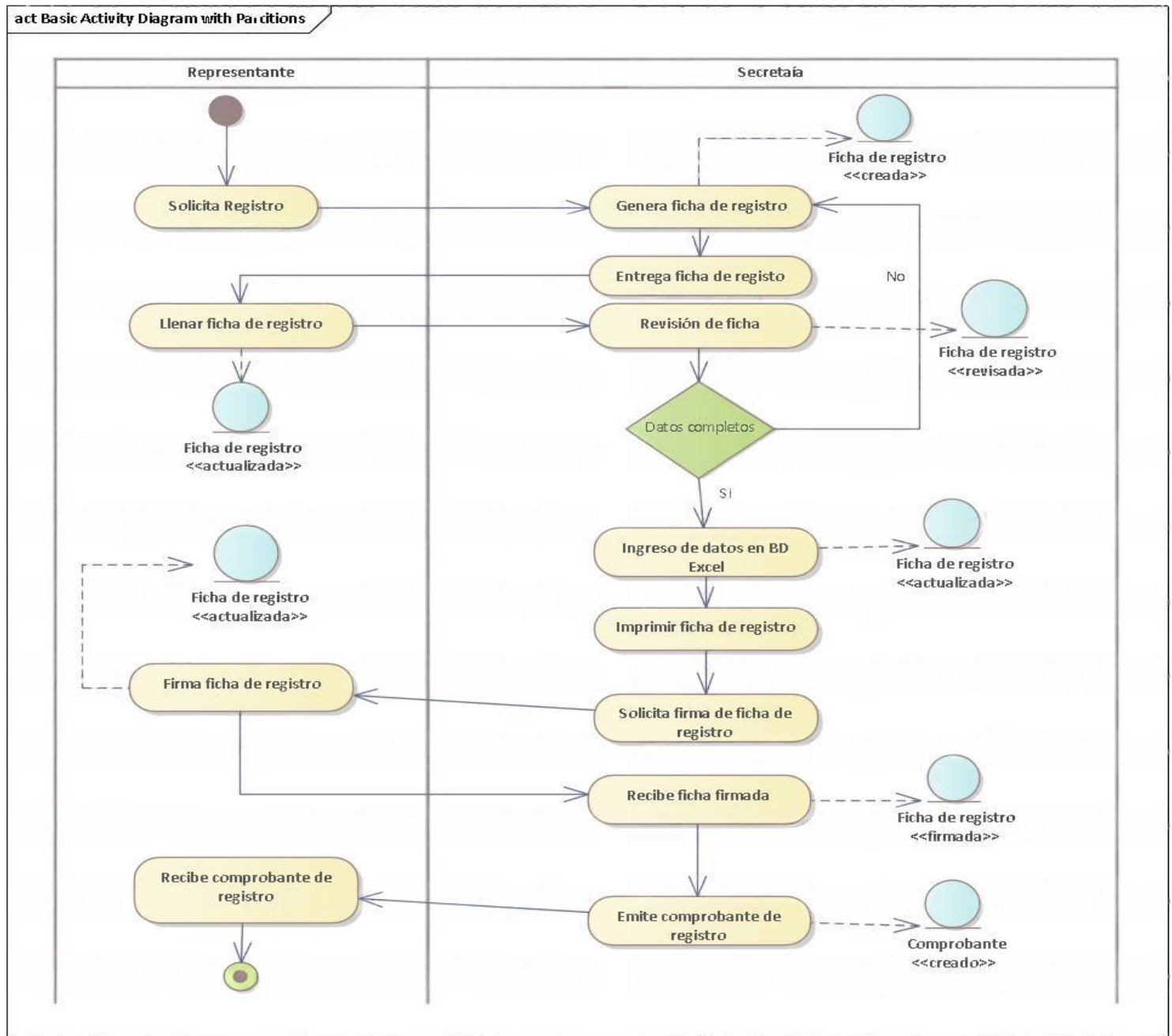
Así de esta manera el diagrama de actividades en este sistema permitirá visualizar de una forma más detallada los procesos, asegurando que cada paso este claramente definido y alineado con los objetivos y necesidades de la institución educativa.

Beneficios de los diagramas de actividades

- Demostrar la lógica de un algoritmo.

- Describir los pasos realizados en un caso de uso UML.
- Ilustrar un proceso de negocios o flujo de trabajo entre los usuarios y el sistema.
- Simplificar y mejorar cualquier proceso clarificando casos de uso complicados.
- Modelar elementos de arquitectura de software, tales como método, función y operación. (Lucidchart, 2019)

Ilustración 3 Diagrama de actividades- Registro de matrículas de estudiantes.



2.1.5. Reglas del negocio

Las reglas de negocio son directrices y restricciones que ayudan a regular las operaciones de una empresa. Las reglas fueron creadas para colaborar con el funcionamiento, ya sea de una sociedad, una escuela, un juego... y no sería diferente en las organizaciones. (Sydle, 2023).

Entre las reglas del negocio que se toman en cuenta para este sistema web son:

- Cada estudiante debe pertenecer a un nivel de estudio.
- Los estudiantes registrados deben ser mayores a 5 años de edad.
- Cada estudiante debe contar con un representante legal para su matriculación.
- La inscripción solo será válida si se realiza dentro del período de matrícula establecida por la institución.
- El número de estudiantes por aula es limitado, 25 estudiantes por grupo.
- Los estudiantes que no hayan aprobado el año anterior deben repetir el curso antes de avanzar al siguiente nivel.
- Los estudiantes deben tener el promedio mínimo asignado por la institución para pasar al siguiente nivel.

2.2. Requisitos de software

Son los que definen la funcionalidad y el propósito de una pieza particular de software o aplicación. Sin estos bien definidos, es difícil crear algo que satisfaga las necesidades y expectativas del usuario. (Northware, 2022).

Requerimientos funcionales

Describen las acciones, los comportamientos y las funciones específicas que debe realizar un sistema. Definen lo que hará el sistema, desde las interacciones del usuario hasta las respuestas del sistema. requerimientos no funcionales, Sin embargo, se centran en el rendimiento del sistema y abordan aspectos como el rendimiento, la seguridad, la escalabilidad y la confiabilidad, para garantizar que el sistema no solo funcione como se espera, sino que también brinde una experiencia de usuario de alta calidad. (Northware, 2022)

2.2.1. Requisitos Funcionales del sistema (RF)

1. RF1: Autenticación de usuario: Encargado de validar las credenciales del usuario que accederá al sistema.
2. RF2: Gestión de usuario: Encargado de tratar los datos requeridos de un usuario para el sistema.
 - RF 2.1 Crear Usuario
 - RF 2.2 Eliminar Usuario
 - RF 2.3 Modificar Usuario
 - RF 2.4 Buscar Usuario

3. RF3: Gestionar matriculación estudiantil: Encargado de ingresar y tratar la información persona de cada estudiante acorde a los requerimientos dados por la institución.
 - RF 3.1 Crear registro de estudiante
 - RF 3.2 Eliminar Registro de estudiante
 - RF 3.3 Modificar Registro de estudiante
 - RF 3.4 Buscar Registro de estudiante
 - RF 3.5 Listar estudiantes registrados
4. RF4: Generar reporte: Se encarga de generar comprobantes con la información solicitada sea del docente o estudiantes en forma de reporte, esta se entregará en formato PDF
 - RF 4.1 Reporte de registro de estudiantes.

Requerimientos no funcionales (RNF)

Los requisitos no funcionales se refieren a los atributos de calidad de un sistema que definen cómo funciona en lugar de qué hace. A diferencia de los requisitos funcionales, que especifican las acciones y tareas que debe realizar un sistema, los requisitos no funcionales se centran en las características generales y el comportamiento del sistema en diversas condiciones. Abordan aspectos como el rendimiento, la facilidad de uso, la fiabilidad y la escalabilidad, garantizando que el sistema cumpla con los estándares de calidad y proporcione una experiencia de usuario satisfactoria. (Northware, 2022).

2.2.2. Interfaz de usuario

Interfaz de usuario

Tabla 5 Interfaz de usuario

RNF Interfaz de usuario (1)		
ID	Requisitos no Funcionales	Prioridad del requerimiento
RNF 1.1	El sistema garantizara una interfaz intuitiva y fácil de usar para los usuarios.	Alta
RNF 1.2	El sistema tendrá un diseño adaptable para dispositivos móviles, tabletas y computadoras.	Alta
RNF 1.3	La interfaz deba permitir la personalización de vistas y configuraciones según el rol de usuario	Baja

Usabilidad

Tabla 6 Usabilidad

RNF Usabilidad (2)		
ID	Requisitos no Funcionales	Prioridad del requerimiento
RNF 2.1	El sistema garantizará la seguridad de los datos para todos los usuarios.	Alta
RNF 2.2	El sistema evitara pasos innecesarios y minimizar la cantidad de datos manuales que deba ingresar el usuario	Alta
RNF 2.3	El sistema debe contar con autocompletado en formularios para reducir errores y acelerar la inscripción.	Alta
RNF 2.4	El sistema mostrara errores con mensajes claros y soluciones para el usuario.	Alta

Seguridad

Tabla 7 Seguridad

RNF Seguridad (3)		
ID	Requisitos no Funcionales	Prioridad del requerimiento
RNF 3.1	Implementar autenticación segura mediante credenciales cifradas, para brindar seguridad en todas las capas del sistema	Alta
RNF 3.2	Para el acceso a diferentes módulos se va a priorizar y restringir según el rol del usuario.	Alta
RNF 3.3	Implementar bloqueo de cuenta tras varios intentos fallidos de inicio de sesión	Alta
RNF 3.4	El sistema garantizara la protección de datos mediante regulaciones locales de privacidad	Alta

Confiabilidad y Disponibilidad

Tabla 8 Confiabilidad y Disponibilidad

RNF Confiabilidad y Disponibilidad (4)		
ID	Requisitos no Funcionales	Prioridad del requerimiento
RNF 4.1	El sistema va a realizar copias de seguridad automáticas diariamente para evitar pérdida de datos	Alta
RNF 4.2	Los errores presentados se presentarán en idioma español	Alta
RNF 4.3	Los formularios contaran con guardado automático para evitar pérdida de datos en caso de cierre inesperado.	Alta

Eficiencia del sistema

Tabla 9 Eficiencia del sistema

RNF Eficiencia del sistema (5)		
ID	Requisitos no Funcionales	Prioridad del requerimiento
RNF 5.1	El sistema debe procesar solicitudes de matriculación en un tiempo máximo de 3 segundos	Alta
RNF 5.2	La búsqueda de estudiantes deberá completarse en menos de 5 segundos con base de datos de hasta 1.000 registros.	Alta
RNF 5.3	La generación de reportes no tardara menos de 5 segundos con grandes cantidades de datos.	Alta

Arquitectura

Tabla 10 Arquitectura

RNF Arquitectura (6)		
ID	Requisitos no Funcionales	Prioridad del requerimiento
RNF 6.1	El sistema contará con una estructura robusta que garantizará su funcionamiento continuo sin interrupciones durante los procesos	Alta
RNF 6.2	El sistema debe tener la posibilidad de ejecutarse en cualquier dispositivo	Alta
RNF 6.3	El sistema tendrá como servidor de base de datos Firebase y será un servicio en línea en tiempo real	Alta

2.3. Diagrama de caso de uso del sistema

Los diagramas de caso de uso para (Zaman, Nadeem, & Sindhu, 2020) se construyen al recopilar y especificar requisitos y representan los requisitos funcionales, que al implementarse serán las funcionalidades del sistema, es decir, muestran lo que hace el sistema y no cómo lo hace.

En este contexto los diagramas de caso de uso representan la funcionalidad de sistema mediante actores y caso de uso. Los casos de usos son actividades o a su vez funciones que el sistema ofrece para sus usuarios.

Ilustración 4 Diagrama casos de uso del sistema

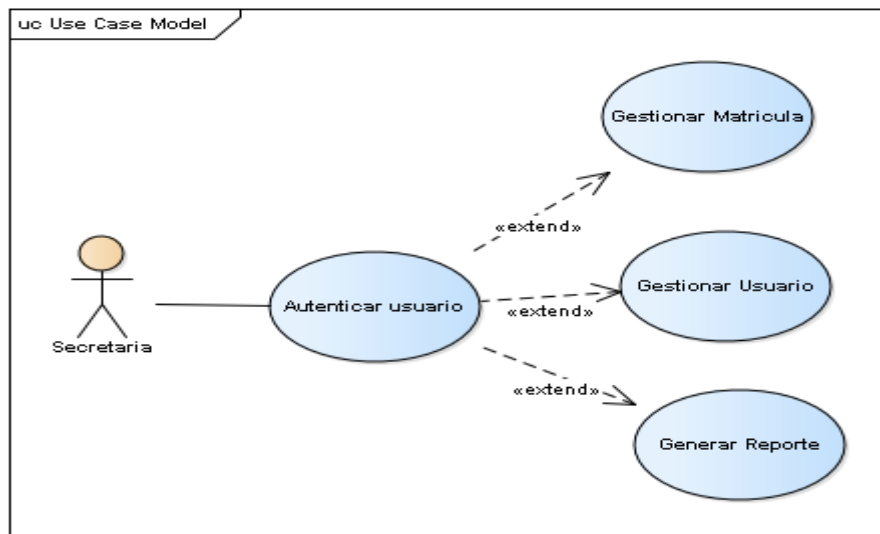


Ilustración 5 Diagrama casos de uso del sistema – Gestionar Usuario

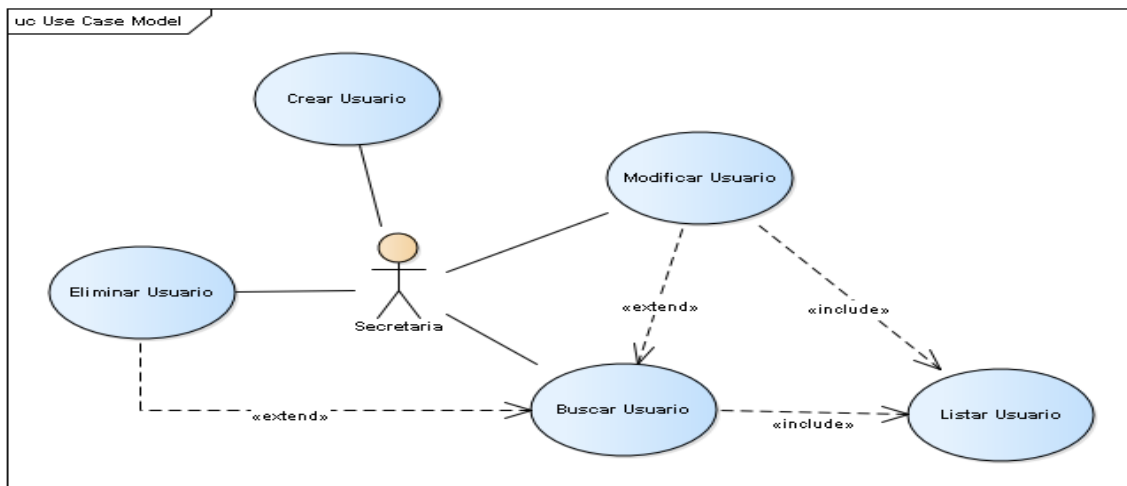


Ilustración 6 Diagrama casos de uso del sistema - Gestionar matricul

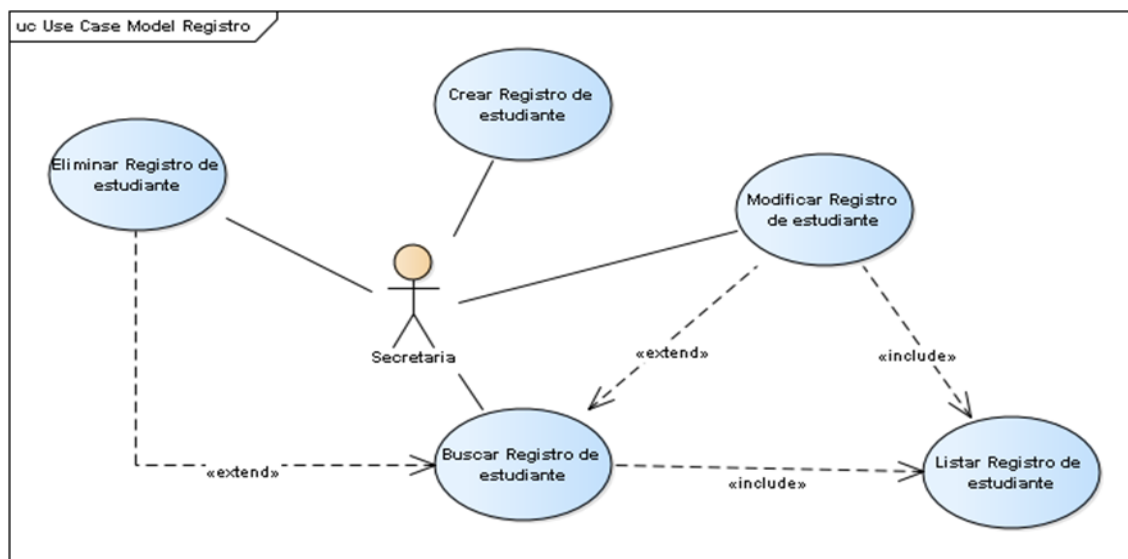
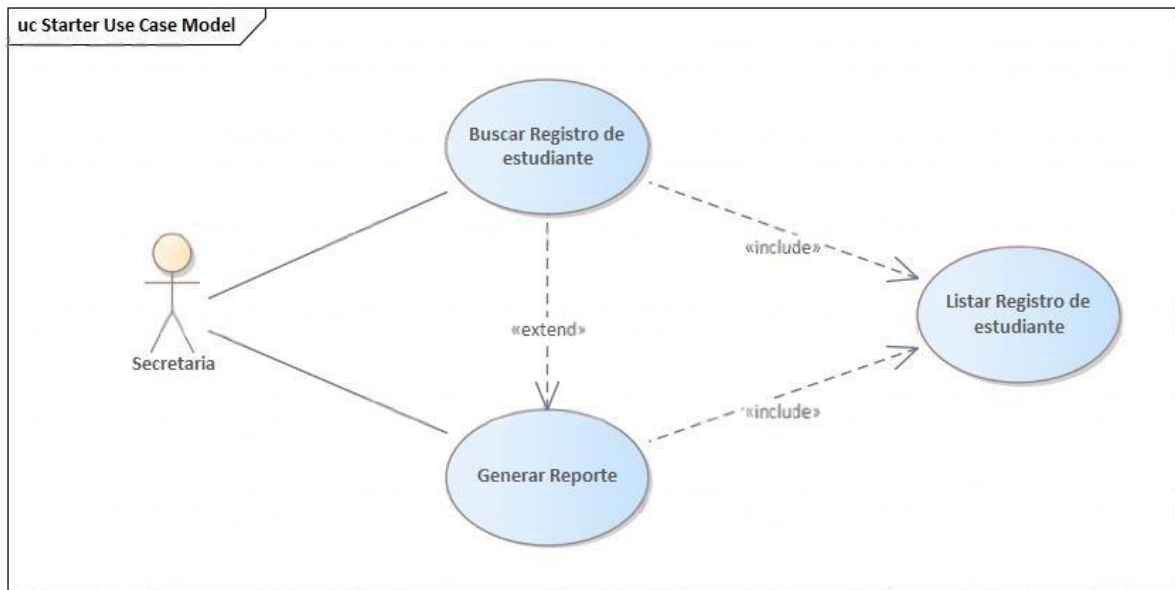


Ilustración 7 Diagrama casos de uso del sistema - Generar Reporte



2.3.1. Listado de casos de uso

Tabla 11 Priorización de los casos de uso

N°	Nombre del caso de uso	Contenido
1	Autenticar Usuario	
2	Gestionar Usuario	2.1 Crear Usuario 2.2 Eliminar Usuario 2.3 Modificar Usuario 2.4 Buscar Usuario
3	Gestionar matriculación estudiantil	3.1 Crear registro de estudiante 3.2 Eliminar Registro de estudiante 3.3 Modificar Registro de estudiante 3.4 Buscar Registro de estudiante 3.5 Listar estudiantes registrados
4	Generación de reporte	4.1 Reporte de registro de estudiantes

2.4. Descripción de los casos de uso del sistema

En la presente investigación se va a describir el caso de uso 3 ya que es el que describe el negocio. De cada uno de estos solo se documentará el escenario crear.

La descripción del caso de uso tendrá en cuenta lo siguiente:

- **Nombre del caso de uso:** nombre del caso de uso que se va a describir
- **Propósito:** descripción del objetivo del caso de uso que se está modelando

- **Actor:** es el actor que va a iniciar el caso de uso
- **Precondición:** Es el conjunto de requisitos que deben cumplirse para que un caso de uso pueda comenzar, en muchas situaciones, implica realización de casos de usos anteriores, representando una condición o estado en lugar de una secuencia de acciones previas al caso de uso.

2.5. Descripción de los casos de uso crear registro

Tabla 12 Descripción de los casos de uso crear registro

CU-3	Gestionar Registro de estudiantes
CU-3.1	Crear registro de estudiantes
RF: 3.1	
Actores	Secretaría
Propósito	Crear un nuevo registro
Resumen	El caso de uso comienza cuando la secretaria accede a la pestaña de crear un nuevo registro y elige entre las opciones estudiante o docente. Luego, se muestra la interfaz correspondiente para administrar la información, permitiendo la creación de un nuevo registro
Precondición	La secretaria esta autenticada en el sistema
Flujo normal del evento	
Acción del actor	Respuesta del sistema
1.- Selecciona la pestaña creación de nuevo registro	2.- Muestra las opciones estudiante o docente.
3.- Selecciona la opción deseada	4.- Muestra la ventana de registro
5.- Selecciona la opción "Nuevo"	6.- Se habilita la ficha para el registro
	7.- Se genera el código automáticamente
8.- Registra toda la información requerida	
9.- Selecciona la opción guardar del menú	10.- se realiza la validación de toda la información ingresada
Flujos alternos	
6.1- En caso de errores den la información ingresada o este incompleta en el nuevo registro no se guardará hasta que el actor de toda la información correcta y completa	
CU-3.2	Eliminar Registro de estudiante
RF: 3.2	
Actores	Secretaría
Propósito	Eliminar un registro

Resumen	El caso de uso comienza cuando la secretaria accede a la pestaña de los registros y selecciona eliminar registro, elige entre las opciones estudiante o docente. Luego, se muestra la interfaz correspondiente para administrar la información, permitiendo eliminar un registro.
Precondición	La secretaria esta autenticada en el sistema
Flujo normal del evento	
Acción del actor	Respuesta del sistema
1.- Selecciona la pestaña eliminar registro	2.- Muestra las opciones estudiante o docente.
3.- Selecciona la opción deseada	4.- Muestra la ventana de registro
5.- Selecciona la opción eliminar	6.- Elimina el registro
	7.- Actualiza la base de datos
Flujos alternos	
6.2- En caso de errores no se actualizará la base de datos y seguirá con todos los registros anteriores.	
CU-3.3	Modificar Registro de estudiante
RF: 3.3	
Actores	Secretaría
Propósito	Modificar un registro
Resumen	El caso de uso comienza cuando la secretaria accede a la pestaña de los registros y selecciona modificar registro, elige entre las opciones estudiante o docente. Luego, se muestra la interfaz correspondiente para administrar la información, permitiendo modificar un registro.
Precondición	La secretaria esta autenticada en el sistema
Flujo normal del evento	
Acción del actor	Respuesta del sistema
1.- Selecciona la pestaña modificar registro	2.- Muestra las opciones estudiante o docente.
3.- Selecciona la opción deseada	4.- Muestra la ventana de registro
5.- Selecciona la opción modificar	6.- Muestra el registro a modificar
7.- Modifica el registro deseado	8.- Actualiza la base de datos
9.- Selecciona la opción guardar del menú	10.- se realiza la validación de toda la información ingresada
Flujos alternos	
6.3- En caso de errores den la información ingresada o este incompleta en el registro no se guardará hasta que el actor de toda la información correcta y completa	
CU-3.4	Buscar Registro de estudiante

RF: 3.4	
Actores	Secretaría
Propósito	Buscar Registro de estudiante
Resumen	El caso de uso comienza cuando la secretaria accede a la pestaña de los registros y selecciona buscar registro, elige entre las opciones estudiante o docente. Luego, se muestra la interfaz correspondiente para administrar la información, permitiendo la búsqueda.
Precondición	La secretaria esta autenticada en el sistema
Flujo normal del evento	
Acción del actor	Respuesta del sistema
1.- Selecciona la pestaña búsqueda de registro	2.- Muestra las opciones estudiante o docente.
3.- Selecciona la opción deseada	4.- Muestra la ventana de registro
5.- Realiza la búsqueda del estudiante	6.- Muestra los registros
Flujos alternos	
6.4- En caso de errores al insertar la información, este incompleta o no exista en el registro no se mostrará los registros solicitados	
CU-3.5	Listar estudiantes registrados
RF: 3.5	
Actores	Secretaría
Propósito	Buscar Registro de estudiante
Resumen	El caso de uso comienza cuando la secretaria accede a la pestaña de los registros y selecciona listar, elige entre las opciones estudiante. Luego, se muestra la interfaz correspondiente de la lista de estudiantes.
Precondición	La secretaria esta autenticada en el sistema
Flujo normal del evento	
Acción del actor	Respuesta del sistema
1.- Selecciona la pestaña lista de registro	2.- Muestra las opciones estudiante o docente.
3.- Selecciona la opción deseada	4.- Muestra la ventana de la lista de registros
Flujos alternos	
6.4- En caso de errores al no tener registros que listar no se mostrará nada, y solo se tendrá un mensaje de error.	

2.6. Conclusión del capítulo

En este capítulo se dio a conocer toda la arquitectura que se va a emplear para el modelado del sistema web, teniendo en cuenta que cada elemento tiene su

descripción detallada, a través de diagrama de clases, diagrama de casos de uso, diagrama de actividades de esta manera poder tener una documentación exhaustiva y precisa.

Así mismo se define los requisitos funcionales y no funcionales del sistema teniendo en cuenta el nivel de complejidad, también se detalla todas las reglas del sistema y de esta manera poder englobar todos estos requisitos para tener un sistema web bien planeado.

CAPÍTULO III

3. DESCRIPCIÓN DE LA PROPUESTA DE SOLUCIÓN

En este capítulo se va abordar el análisis y diseño del sistema web propuesto, dando a conocer la arquitectura empleada, la cual se graficará en los diagramas de diseño mediante uso de tecnologías, lenguajes y metodologías previamente descritas.

Se establece la implementación de sistema de acuerdo a la arquitectura propuesta por el diseño. Para esto, se dará a conocer un modelo de datos que representa la base de datos a utilizar, junto con diagramas de componentes y despliegue, lo que nos ayudará con el modelo de implementación. También se dará el tratamiento de errores, métodos de seguridad y estilos aplicados en el desarrollo del sistema web.

3.1.1. Diseño Arquitectónico

MVC es un patrón de diseño que considera dividir una aplicación en tres módulos claramente identificables y con funcionalidad bien definida: El Modelo, las Vistas y el Controlador.

El modelo

El modelo es un conjunto de clases que representan la información del mundo real que el sistema debe procesar, así por ejemplo un sistema de administración de datos climatológicos tendrá un modelo que representará la temperatura, la humedad ambiental, el estado del tiempo esperado, etc. Para (Enríquez, Fierro, Flores, Imbaquingo Esparza, & Michelena, 2023) el modelo representa la lógica de negocio y los datos de la aplicación. Es responsable de acceder a la base de datos, realizar cálculos y aplicar las reglas del negocio.

Las vistas

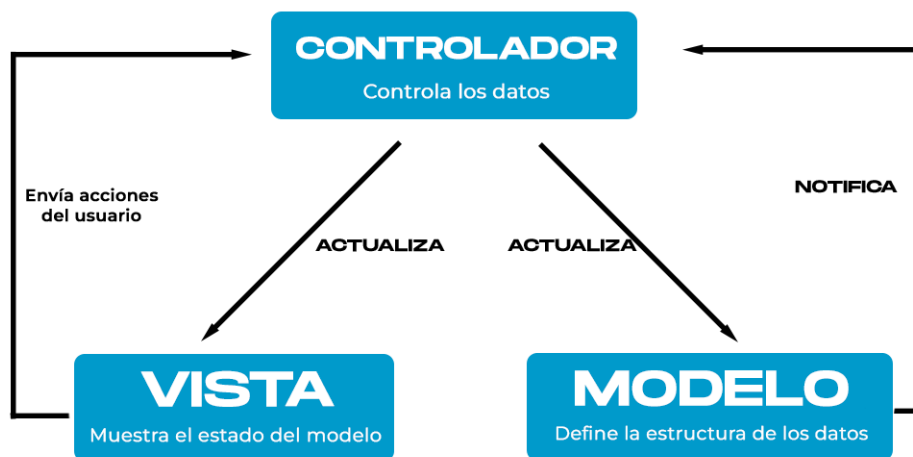
Las vistas son el conjunto de clases que se encargan de mostrar al usuario la información contenida en el modelo. Una vista obtiene del modelo solamente la información que necesita para desplegar y se actualiza cada vez que el modelo del dominio cambia por medio de notificaciones generadas por el modelo de la aplicación. Es la interfaz de usuario que presenta los datos al usuario. Se encarga de mostrar la

información contenida en el modelo de manera adecuada. (Enríquez, Fierro, Flores, Imbaquingo Esparza, & Michelena, 2023)

El controlador

El controlador es un objeto que se encarga de dirigir el flujo del control de la aplicación debido a mensajes externos, como datos introducidos por el usuario u opciones del menú seleccionadas por él. A partir de estos mensajes, el controlador se encarga de modificar el modelo o de abrir y cerrar vistas. Actúa como intermediario entre la Vista y el Modelo. Recibe las entradas del usuario desde la Vista, las procesa y determina qué operaciones realizar en el Modelo. (Enríquez, Fierro, Flores, Imbaquingo Esparza, & Michelena, 2023)

Ilustración 8 Relación entre los módulos de MVC



3.2. Diagrama de clases de diseño

El diagrama de clases de diseño es una herramienta fundamental en la ingeniería de software orientada a objetos. Forma parte del Lenguaje Unificado de Modelado (UML) y representa la estructura estática del sistema, detallando las clases, sus atributos, métodos y las relaciones entre ellas. Este diagrama es esencial durante las fases de análisis y diseño, ya que permite visualizar y planificar la arquitectura del sistema antes de su implementación. (Ucán Pech, Aguilar Vera, Díaz Mendoza, & Gómez Gómez, 2022)

A continuación, se describe el diagrama de clases, este diagrama tiene un modelo arquitectónico MVC (Modelo Vista Controlador), como centro de nuestro diseño tenemos la clase "CC_RegistroAlumno", la cual funciona como clase controladora, esta clase tiene métodos fundamentales como registrar, listar, guardar, eliminar y

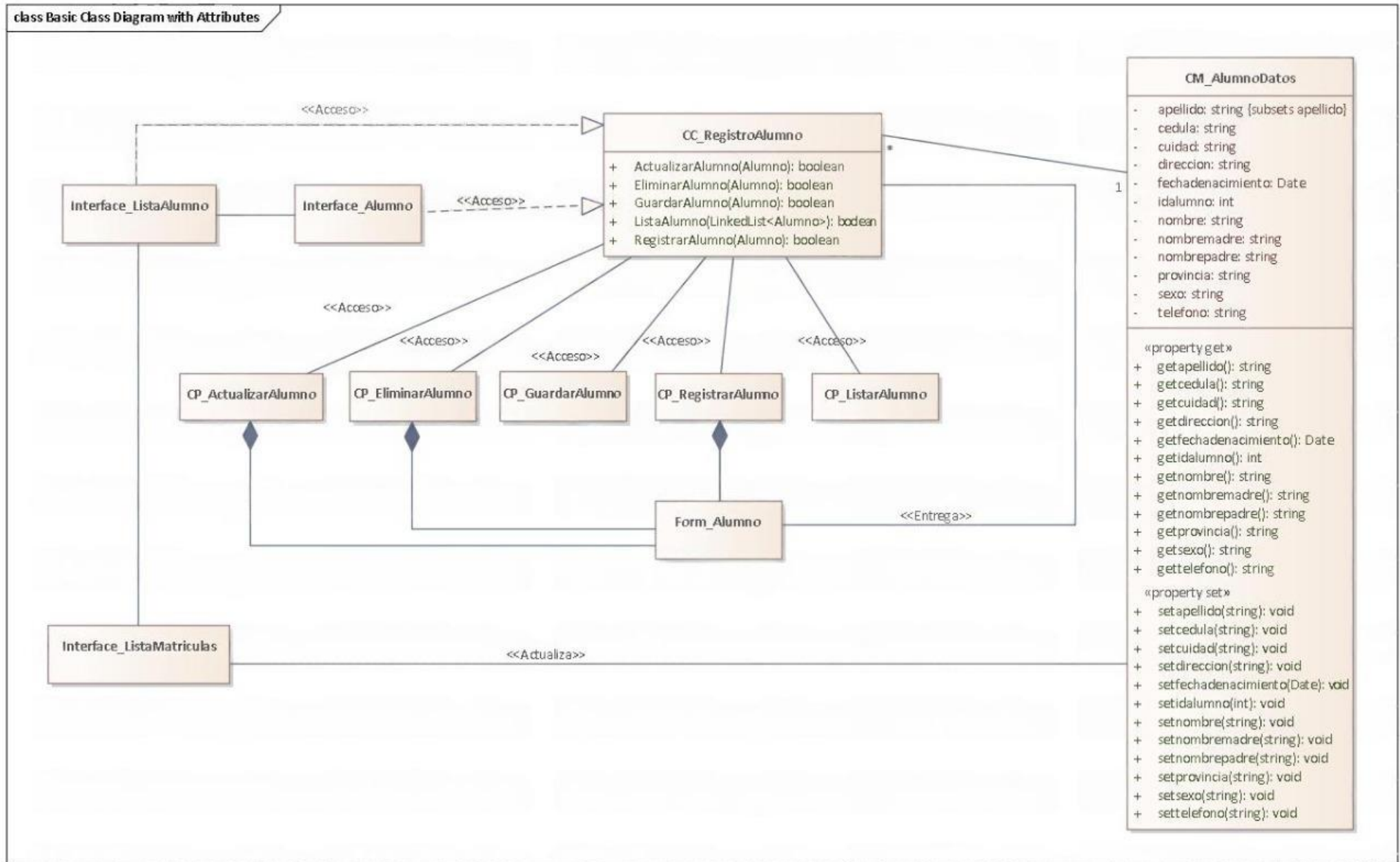
actualizar los registros de los estudiantes, dentro de esta clase controladora tenemos varios client pages que son CP_ActualizarAlumno, CP_EliminarAlumno, CP_GuardarAlumno, CP_ListarAlumno, CP_RegistrarAlumno.

Cada una de estas clases tiene un uso específico, y van agregadas con un formulario que tiene como nombre Form_Alumno, esto va hacer que al entrar con una client page se nos despliegue un formulario dependiendo la operación que se desee realizar.

Los datos de cada estudiante están enfocados en la clase "CM_AlumnoDatos", esta es una clase modelo de datos, donde se almacena todos los atributos de cada alumno como su nombre, dirección, fecha de nacimiento, sexo, etc. Dicha clase modelo contiene también sus métodos para poder acceder a los datos mediante getters y setters.

Para la interacción con el usuario se presenta varias interfaces que son Interface_Alumno, Interface_ListaAlumno y Interface_ListaMatricula, estas interfaces son las que se nos va a desplegar en el sistema web las cuales el usuario va a poder interactuar y realizar operaciones que sea de su necesidad.

Finalmente, este diseño también cuenta con sus conexiones entre cada componente del sistema y están relacionadas con etiquetas como <<Acceso>>, <<Entrega>> y <<Actualiza>>. Estas etiquetas nos ayudan a identificar como interactúan las distintas clases entre sí, quien accede a los componentes, cual clase entrega información y cuál de estas clases actualiza.



3.3. Diagrama de secuencia

Según (Cillero, 2023) el diagrama de secuencia es un tipo de diagrama de interacción cuyo objetivo es describir el comportamiento dinámico del sistema de información haciendo énfasis en la secuencia de los mensajes intercambiados por los objetos.

Un diagrama de secuencia tiene dos dimensiones, el eje vertical representa el tiempo y el eje horizontal los diferentes objetos. El tiempo avanza desde la parte superior del diagrama hacia la inferior. Normalmente, en relación al tiempo solo es importante la secuencia de los mensajes, sin embargo, en aplicaciones de tiempo real se podría introducir una escala en el eje vertical. Respecto a los objetos, es irrelevante el orden en que se representan, aunque su colocación debería poseer la mayor claridad posible.

El diagrama de secuencia se representa de acuerdo a la realización de los diagramas de clases de diseño, se podrá observar cual es la secuencia para cumplir con las funciones de registrar solicitud de petición, generar reporte de ganancias, gestionar usuario, cliente, servicios y productos.

Diagrama de secuencia Registrar Alumno

El diagrama que se presenta a continuación representa el proceso de registro de un alumno nuevo en el sistema, desde como interactúa la secretaria hasta como se van almacenar los datos.

La secuencia comienza con una secretaria, quien es una trabajadora del sistema, todo inicia al ejecutar la acción Seleccionar alumno (), aquí se va a mostrar una interfaz de lista de alumnos (Lista Alumno), lo cual indica que en esta interfaz la secretaria va a poder seleccionar la opción de crear nuevo alumno.

En la interfaz Lista Alumno se llama al método Crear Alumno (), esta representa la apertura de un formulario para poder registrar los datos del nuevo estudiante. Una vez estando en esta interfaz se ejecuta la operación IngresoDatos (), el cual envía todos los datos a la clase controladora CC_RegistroAlumno, la cual su responsabilidad es organizar la lógica del sistema, el controlador recibe los datos y envía al método GuardarAlumno () en la clase modelo CM_AlumnoDatos, el cual es una base de datos donde va almacenar todo el dato del nuevo estudiante.

Finalmente, después del proceso de guardado, la clase modelo CM_AlumnoDatos devuelve un mensaje al controlador de confirmación usando `Return (): AlumnoGuardado`, que indica que los datos han sido guardados correctamente. A su vez el controlador da una respuesta a la interfaz Alumno con `Return (): Alumno`, dando como exitosa a la operación. Para finalizar la interfaz Lista Alumno llama a `VisualizarAlumno ()`, mostrando el registro del nuevo estudiante registrado.

Este diagrama describe de forma clara todo el flujo de interacciones entre las capas del sistema, desde la secretaria, pasando por las interfaces del usuario, la clase controladora, y la clase modelo.

Ilustración 10 Diagrama de secuencia, Registrar Alumno

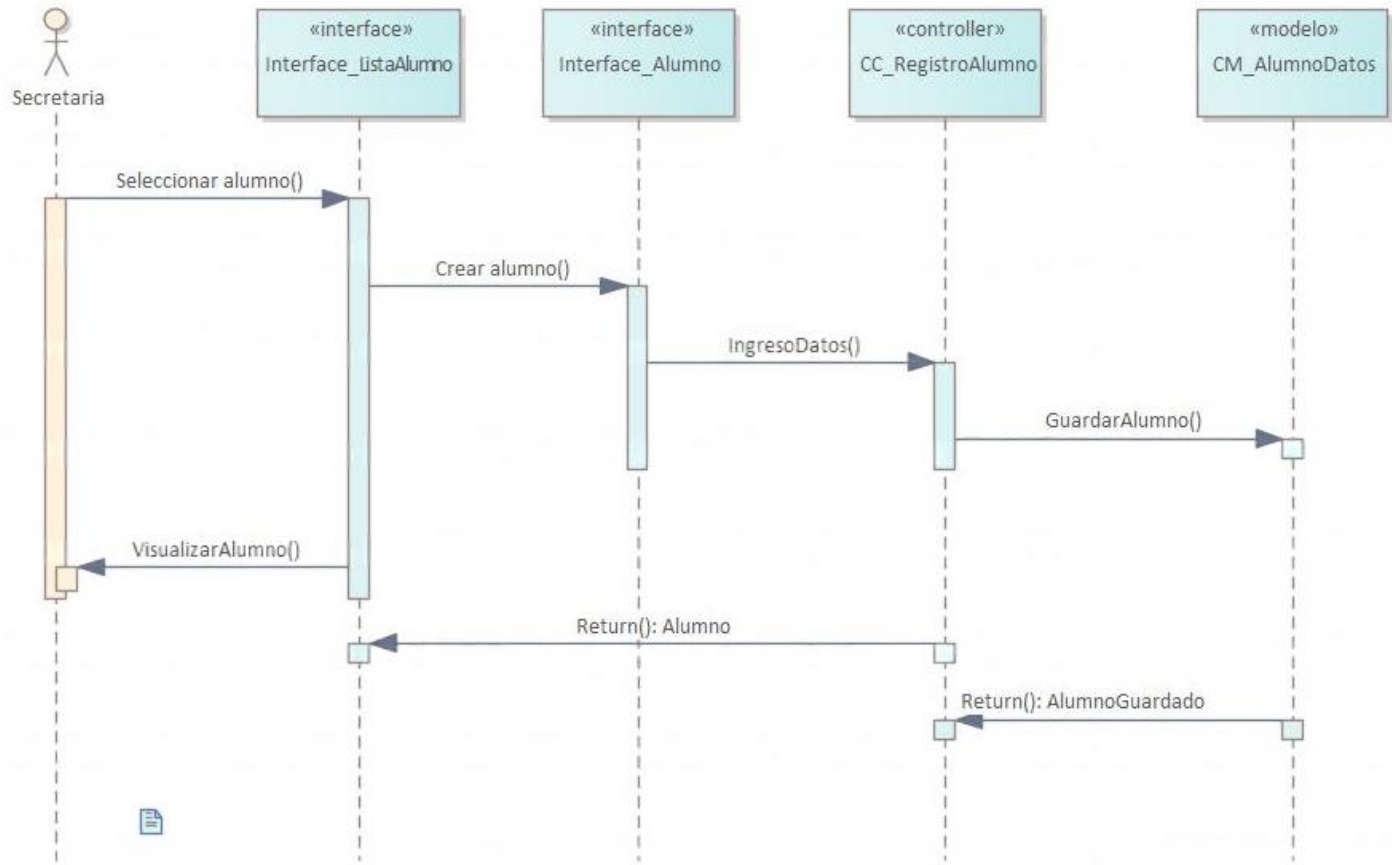


Diagrama de secuencia Actualizar Alumno

El diagrama describe el proceso de actualización de los datos de un alumno ya matriculado.

Empieza la secretaría va a interactuar con la interfaz `Interface_listaAlumno` el cual le va a listar todos los alumnos que estén matriculados seguido de esto la secretaria va a seleccionar el alumno que se va actualizar, y se llama al método `ActualizarAlumno ()` en la `Interface_Alumno`.

Dentro de la `Interface_Alumno` se solicita la entrada de datos mediante el método `IngresoDatos ()` el cual va a recopilar la información que se va a cambiar, seguido de esto se llama a la clase controladora `CC_RegistroAlumno` la cual va a recibir los datos y llamara al método `ActualizarAlumno ()` para que se ejecute en la clase modelo `CM_AlumnoDatos` para persistir la actualización.

Finalmente, en la clase `CM_AlumnoDatos` devuelve una confirmación de alumno actualizado mediante `Return (): AlumnoActualizado` al controlador `CC_RegistroAlumno`, este a su vez nos va a retornar la información actualizada del estudiante mediante `Return (): Alumno` a la `Interface_Alumno`. Para finalizar la interfaz `Interface_listaAlumno` va a mostrar al alumno actualizado a la secretaria a través del método `VisualizarAlmuno ()`, dando como finalizada la operación de actualización.

Ilustración 11 Diagrama de secuencia, Actualizar Alumno

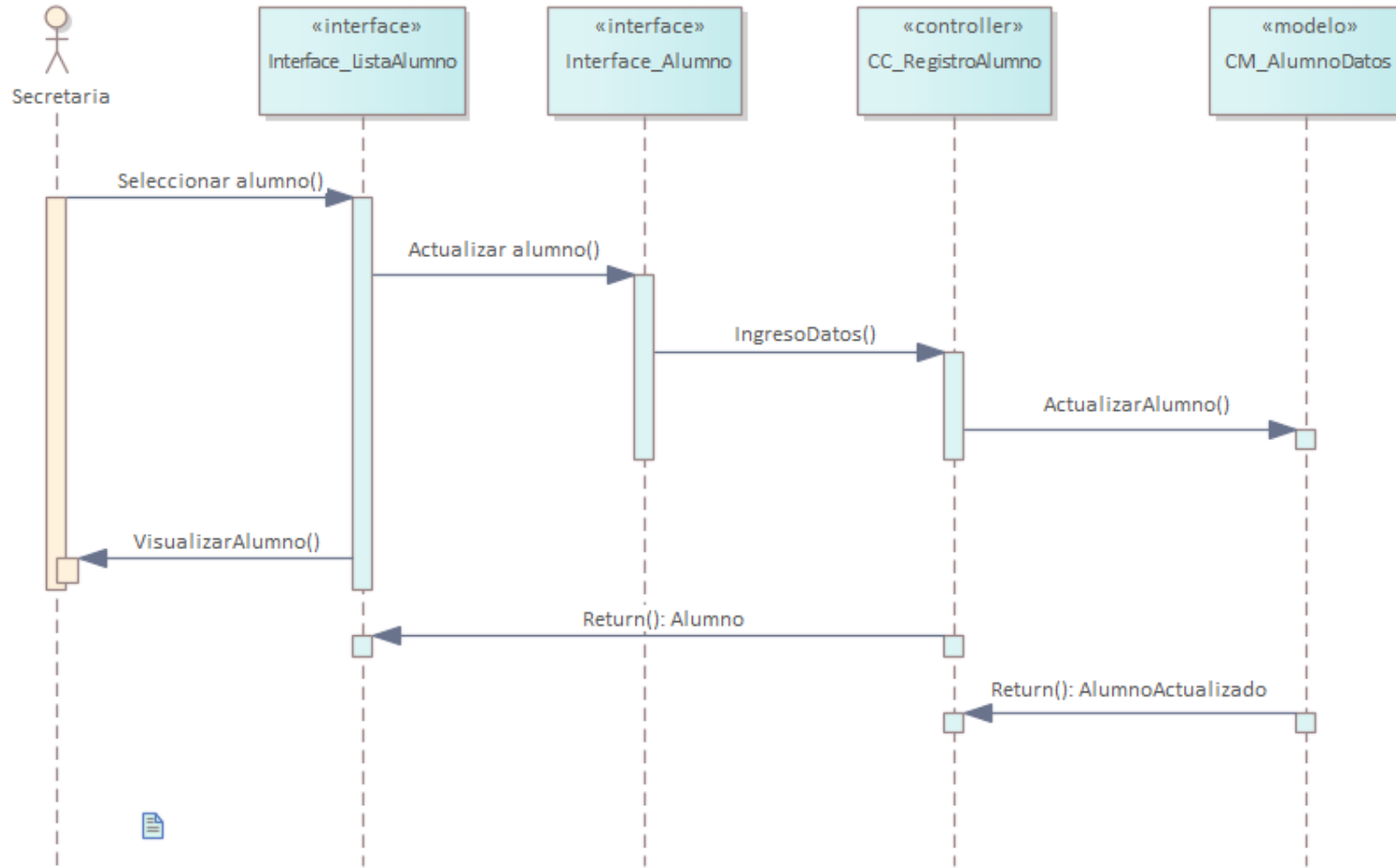


Diagrama de secuencia Eliminar Alumno

La secuencia empieza cuando la secretaría va a interactuar con la interfaz `Interface_listaAlumno` el cual le va a listar a los alumnos, seguido de esto la secretaria va a seleccionar el alumno que desea eliminar, y se llama al método `EliminarAlumno ()` en la `Interface_Alumno`.

Posterior dentro de la `Interface_Alumno` se solicita la selección de datos del alumno mediante el método `SeleccionarDatos ()`, seguido de esto se llama a la clase controladora `CC_RegistroAlumno` la cual tiene los datos del alumno que se desea eliminar y llamara al método `EliminarAlumno ()` para que se ejecute en la clase modelo `CM_AlumnoDatos` y de esta manera eliminar el alumno.

Finalmente, en la clase `CM_AlumnoDatos` devuelve un mensaje de confirmación de alumno eliminado mediante `Return (): AlumnoEliminado`, al controlador `CC_RegistroAlumno`, este a su vez nos va a retornar un mensaje de alumno eliminado y también a todos los alumnos que no fueron eliminados mediante el `Return (): Mensaje` a la `Interface_Alumno`.

Para finalizar la interfaz `Interface_listaAlumno` va a mostrar a todos los alumnos que no fueron eliminados a la secretaria a través del método `VisualizarAlmuno ()`, dando como finalizada la operación de eliminar.

Ilustración 12 Diagrama de secuencia, Eliminar Alumno

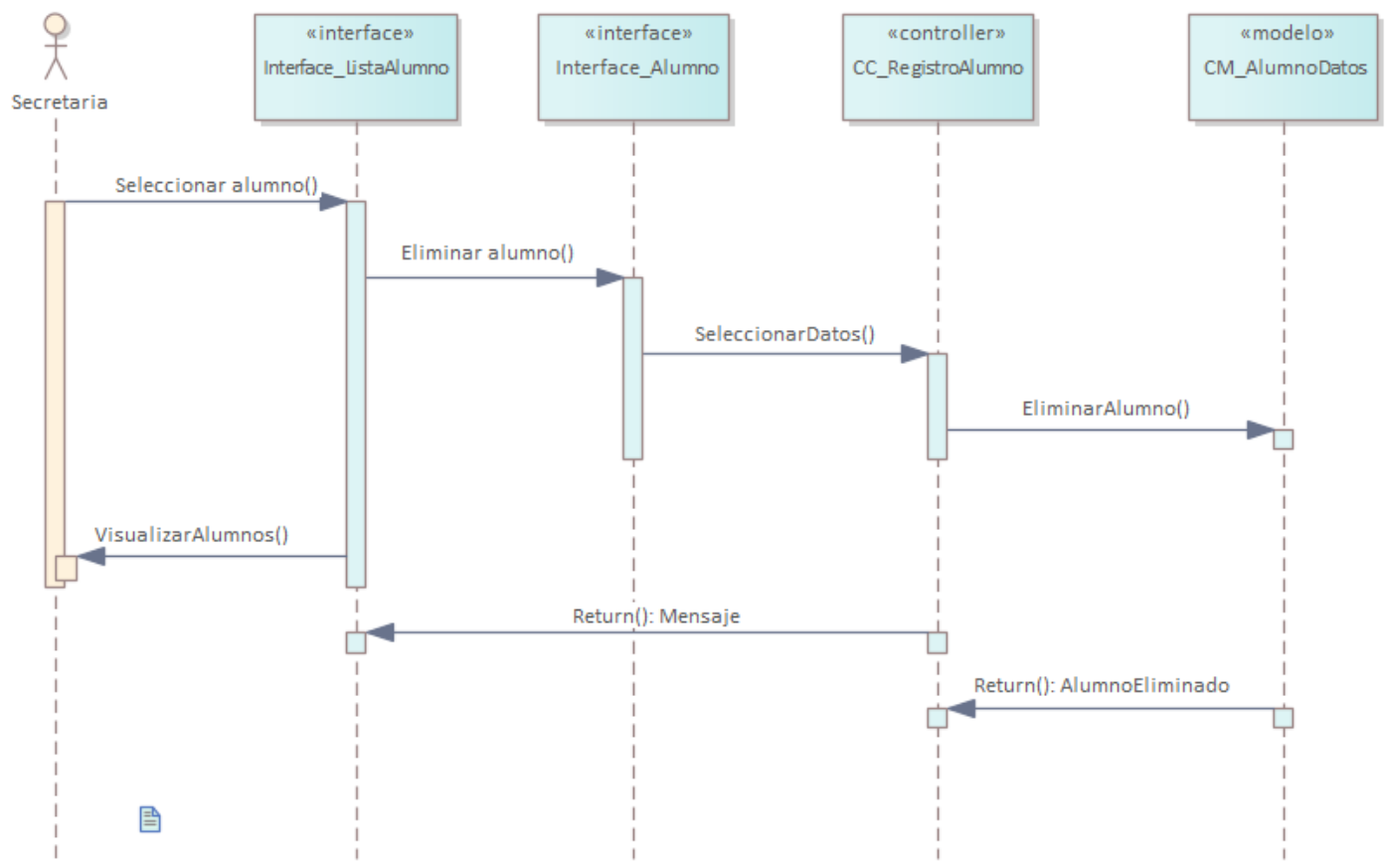


Diagrama de secuencia Listar Alumno

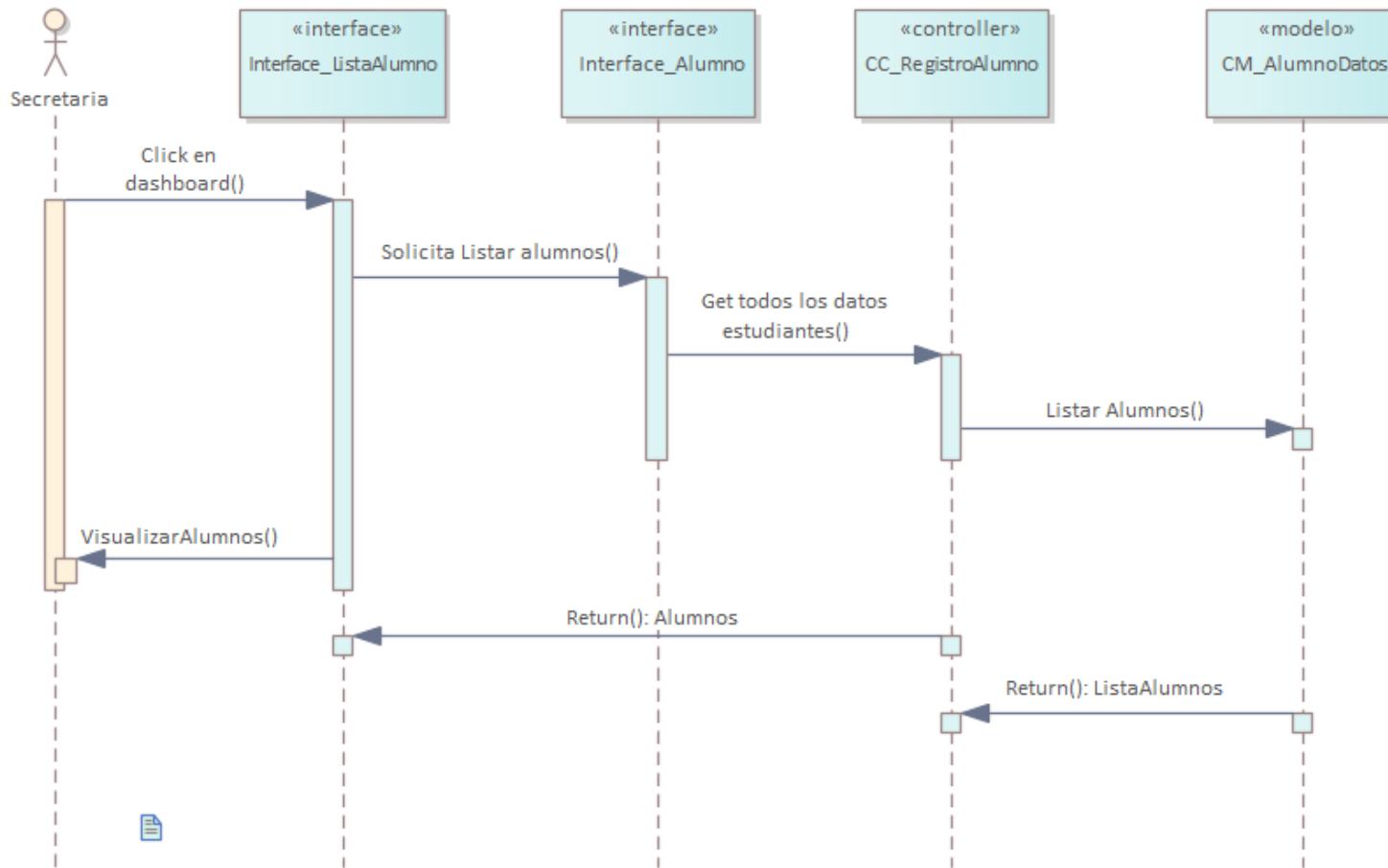
En este diagrama se muestra como el sistema va a listar a los alumnos empezando cuando la secretaria interactúa con la Interface_ListaAlumno en la cual da clic en el dashboard de la pantalla principal, este clic hará que la Interface_ListaAlumno invoque al método solicita listar alumnos () dentro de la Interface_Alumno.

Siguiendo la secuencia la Interface_Alumno solicita todos los datos que se necesite para ser listados a través del método Get_todos_los_datos_ estudiantes () dentro del controlador CC_RegistroAlumno. El controlador, va solicitar al método ListarAlumnos () dentro de la clase modelo CM_AlumnoDatos que será para poder recopilar toda la información de los alumnos de la base de datos.

Cuando la clase modelo CM_AlumnoDatos haya recopilado todos los datos que se solicitó, nos va a retornar una lista de alumnos mediante el método Return (): Lista Alumnos, el controlador posteriormente nos va a retornar toda la información del alumno mediante el Return (): Alumnos a la Interface_ListaAlumno.

Finalmente, la secretaria visualizara la lista de todos los estudiantes que se solicitó a través del método VisualizarAlumnos (), completando con esto el diagrama de secuencia Listar Alumno.

Ilustración 13 Diagrama de secuencia, Listar Alumno



3.4. Diagrama de componentes

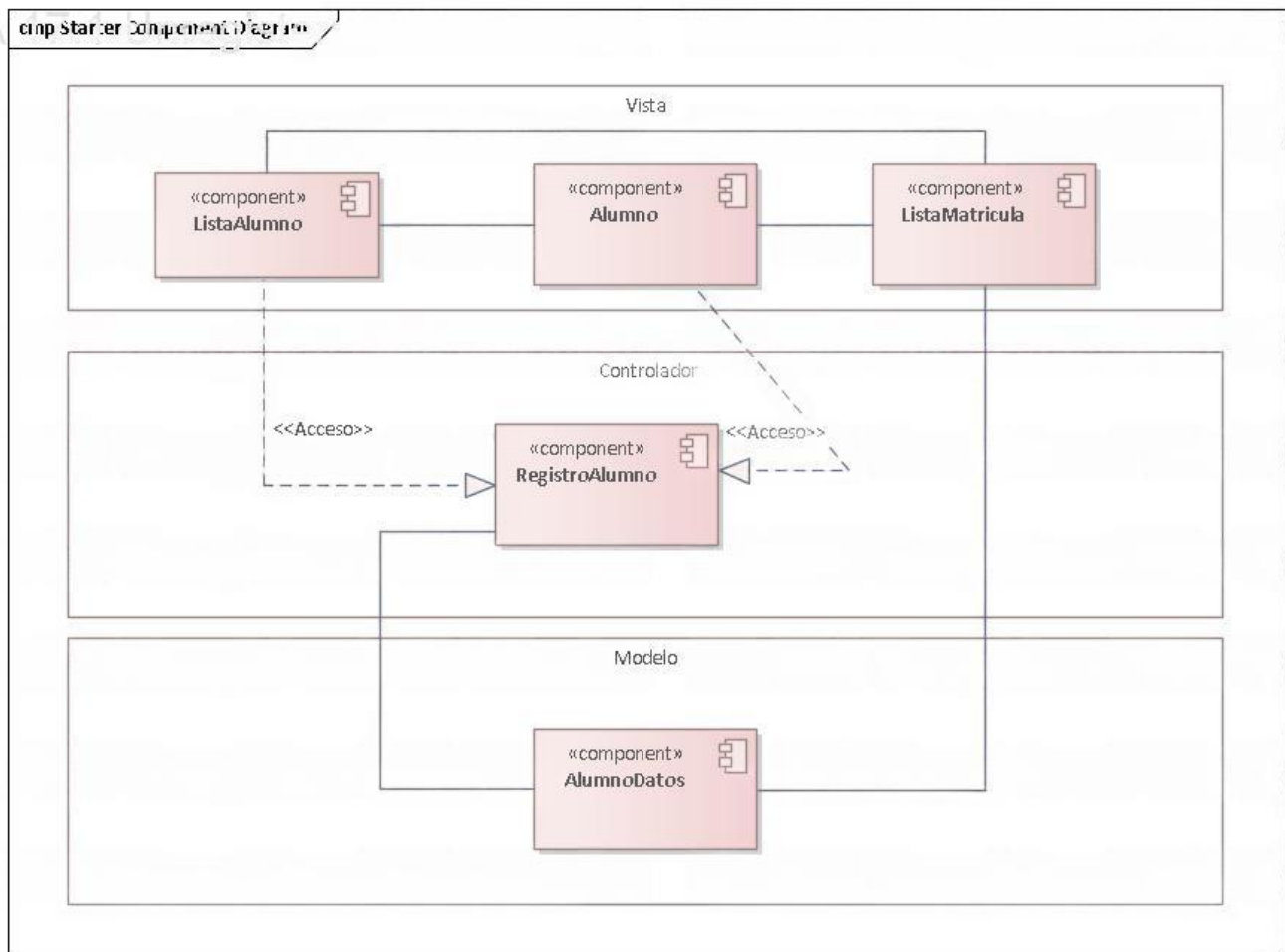
Estos diagramas enseñan la estructura de implementación de un sistema, por lo que los desarrolladores saben cómo mantenerlo y modificarlo. También muestran dependencias entre componentes, lo que puede ayudar a identificar problemas. Los diagramas de componentes muestran la estructura física o aspectos físicos de un sistema. Pueden ser de alto nivel o detallados. Cada componente modelado proporciona un servicio específico y se comporta de una manera bien definida. Los conectores entre ellos también tienen significados específicos. (Lucidchart, 2019).

En el siguiente diagrama de componentes se demuestra la arquitectura del sistema web, que se va a tener la arquitectura MVC (Modelo, Vista, Controlador), dentro de la capa Vista tenemos los componentes como "ListarAlumno", "Alumno" y "Lista Matrícula", que son las interfaces del usuario por donde se va a reflejar los datos, permitiendo que el usuario pueda interactuar con la información de los alumnos.

La capa del Controlador tenemos el componente "RegistroAlumno", que es un intermediario ya que es el que procesa todas las solicitudes de la Vista, de esta manera este organiza y reparte las operaciones que se desee desarrollar. "RegistroAlumno" tiene como relaciones de "Acceso" con "ListaAlumno" y "Alumno", lo que nos indica que administra todas las interacciones para poder reflejar toda la información del alumno.

Por último, en la capa de Modelo tenemos el componente "AlumnoDatos", el cual encapsula la lógica del negocio y persistencia de los datos de los alumnos. En el controlador el "RegistroAlumno" va a estar conectado con "AlumnoDatos" debido a que son 2 partes fundamentales ya que para poder realizar las operaciones y manipular los datos se va a necesitar que "AlumnoDatos" nos de los datos de los estudiantes para que en el controlador se pueda realizar cualquier operación.

Ilustración 14 Diagrama de componentes



3.5. Diagrama de despliegue

Un diagrama de despliegue es un tipo de diagrama del Lenguaje Unificado de Modelado (UML) que se utiliza para representar el despliegue físico de los componentes de un sistema. Muestra cómo se asignan los artefactos de software a los componentes de hardware, conocidos como nodos, e ilustra las configuraciones físicas de software y hardware que son críticas para la ejecución, operación y escalabilidad de un sistema. (Lucidchart, 2019).

Nodo Aplicación

Representa la interfaz gráfica del sistema y muestra las opciones con las que el sistema cuenta

Nodo Pc Usuario

Representan a los equipos que son manipuladas por los usuarios por donde se accede a la información

Nodo Servidor

Es la máquina del servidor en donde se va alojar la aplicación, este nodo se encarga de enviar la información mediante otros nodos:

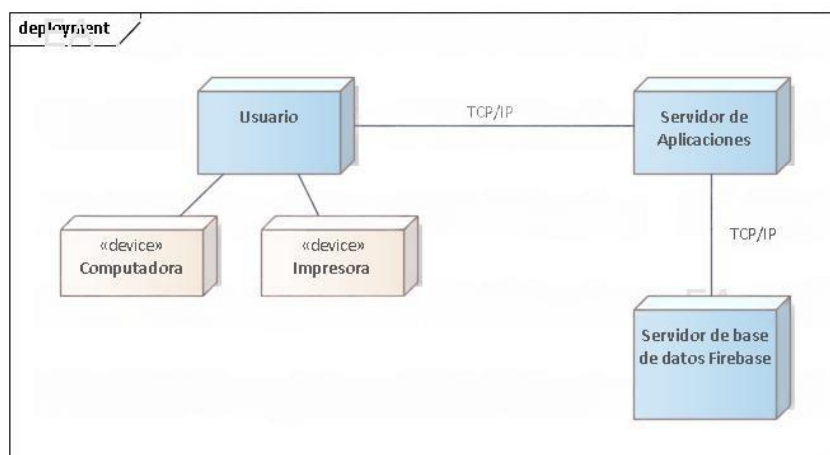
- Con el nodo gestor base de datos: Firebase
- Con el nodo de aplicación

En el siguiente diagrama se observa que tenemos un nodo principal que es el Usuario, el cual representa el entorno donde los usuarios finales van a tener interacciones con el sistema. Este nodo Usuario tiene a dos dispositivos («device») una computadora y una impresora. Esto indica que cada uno de los usuarios pueden acceder al sistema mediante la computadora y realizar operaciones que sean necesarias de la impresora.

Desde el nodo Usuario se establece una conexión con el Servidor de Aplicaciones mediante TCP/IP, este servidor de aplicaciones es el que se en carga de operar toda la lógica del negocio del sistema y recibir las solicitudes que desee el usuario.

A su vez, el servidor de aplicaciones está conectado con el servidor de base de datos Firebase, esta conexión igual esta realizada mediante TCP/IP, lo que da a conocer que el almacenamiento y gestión de los datos del sistema será en la base de datos Firebase, es por esto que el servidor de aplicaciones debe acceder al el para manipular información.

Ilustración 15 Diagrama de despliegue



CAPÍTULO IV

4. PRUEBAS Y ANÁLISIS DE FACTIBILIDAD

4.1. Breve descripción del capítulo

En este capítulo, se evalúa la estimación del esfuerzo para el desarrollo del software, para este estudio se utilizará medos de pintos de casos de uso. Las pruebas de comportamiento de la aplicación se realizaron mediante pruebas, esto tiene el propósito de identificar las deficiencias en el funcionamiento de las interfaces.

4.2. Planificación basada en uno de los métodos de estimación

Para realizar el análisis de costos, relacionados al sistema web se va a implementar el método de Puntos Casos de Uso fue desarrollado por Gustav Kamer, bajo la supervisión de Ivar Jacobson (creador de los casos de uso y gran promovedor del desarrollo de UML y el Proceso Unificado). Este método ha sido ampliamente utilizado por la empresa Racional. Su principal ventaja es su rápida adaptación a empresas que ya estén utilizando la técnica de Casos de Uso. (Gómez, 2019).

La técnica tendrá como objetivos:

- Medir libremente la tecnología utilizada en la implementación del sistema
- Proporcionar una métrica del tamaño.
- Proporcionar un instrumento para la estimación del software

4.2.1. Caracterización de los requisitos según su complejidad

Los requisitos funcionales se caracterizan según la complejidad teniendo en cuenta el nivel de esfuerzo obligatorio para el diseño, implementación e integración de un requisito funcional.

Tabla 13 Complejidad de los requisitos funcionales

VALOR	DESCRIPCIÓN
ALTA	El requisito funcional contiene más de 8 servicios, se instancian para consulta, modificación, o creación más de 5 entidades de sistema, la figura que accede a la misma es una persona y la interfaz diseñada posee elementos de animación, y/o gráficos, y/o componentes tradicionales de captura y muestra de información. Además, no se utilizan elementos de reutilización, el Requisito funcional se desarrolla en su totalidad por un programador.
MEDIA	El Requisito funcional contiene de 5 a 8 transacciones, se instancian para consulta, modificación, o creación entre 3 y 5 entidades de sistema, el actor que accede a la misma es una persona o un sistema externo que accede mediante un protocolo o una interfaz basada en texto (Servicios Web, XML, Correo electrónico); y la interfaz diseñada posee solo componentes tradicionales de captura y muestra de información. Además, se utilizan elementos donde es necesario ajustar la codificación antes de la generación de la implementación del CU.
BAJA	El Requisito funcional contiene de 1 a 4 servicios, se instancian para consulta, modificación, o creación hasta 2 entes de sistema, la figura que accede a la misma es una persona o un sistema externo que aprueba mediante una interfaz de programación (API, DLL); y la interfaz diseñada posee solo mecanismos tradicionales de captura y muestra de información o no dispone de interfaz asociada. Además, se utilizan elementos que realiza el requisito funcional y sólo es necesario concertar los atributos de uso.

Fuente: (Landi Paucar, 2019)

4.2.2. Clasificación de requisitos funcionales

Tabla 14 Clasificación de requisitos funcionales

ID	Nombre de Requisito	Complejidad
RF1.1	Crear Usuario	Baja
RF1.2	Eliminar Usuario	Baja
RF1.3	Modificar Usuario	Baja
RF1.4	Buscar Usuario	Baja
RF2.1	Crear Estudiante	Baja
RF2.2	Eliminar Estudiante	Baja
RF2.3	Modificar Estudiante	Baja
RF2.4	Buscar Estudiante	Baja
RF3,1	Reporte de registro de estudiante	Media

4.2.3. Definición de factores de complejidad

Se define la complejidad de los requisitos de la siguiente manera:

Tabla 15 Definición de factores de complejidad

ID	Complejidad del requisito	Peso
C1	Alta	3
C2	Media	2
C3	Baja	1

Para medir la complejidad de los actores que manipulan el sistema, los factores que se utilizará se muestra en la siguiente clasificación.

Tabla 16 Definición de factores para medir la complejidad.

ID	Actores	Peso	Criterio de clasificación
A1	Alta	0,5	Persona que interactúe con el sistema mediante una interfaz gráfica.
A2	Media	0,3	Un sistema o personas que interactúen a través de una interfaz de modo consola
A3	Baja	0	Otro sistema que interactúa con el sistema a desarrollar mediante una interfaz de programación

4.2.4. Factores técnicos

Los elementos técnicos son aquellos que incluyen de manera favorable o desfavorable dentro de los requisitos del software. Los factores ambientales, se pueden manipular ya que están asociados directamente con la complejidad tecnológica y requerimientos no funcionales del sistema. Estos factores intervienen directamente en el esfuerzo, fundamentalmente en la etapa de implementación.

Tabla 17 Descripción de los factores técnicos

ID	Factor Técnico	Peso (0,5-2)	Valor (0,1)	Descripción
FT1	Sistema distribuido.	2	Toma el valor definido.	Se corresponde con requisitos cuyos componentes hardware y/o software, se encuentran en otros ordenadores conectados en red, se comunican y coordinan sus acciones mediante el paso de mensajes, para el logro de un objetivo.
FT2	Rendimiento y Concurrencia.	0.5	Toma el valor definido.	Se refiere a la capacidad del requisito para cumplir con sus funciones, por lo general va muy relacionado con la respuesta que ostenta el requisito ante el acceso de varios usuarios al mismo tiempo y su tiempo de respuesta.
FT3	Complejidad de diseño gráfico.	0.5	Toma el valor definido.	Se refiere a la complejidad visual diseñada en el prototipo del requisito.
FT4	Algoritmos y procesamiento interno.	0.5	Toma el valor definido.	Se refiere a la complejidad en la definición del requisito y complejidad de los algoritmos a utilizar, ejemplo: Implementación de funciones matemáticas, gráficos, etc.
FT5	Seguridad.	1	Toma el valor definido.	Se refiere a la característica del requisito para manejar la identidad de las entidades, los accesos a los recursos, el registro cronológico de las acciones, verificar que la información esté disponible solo para las entidades permitidas, integridad de la información, disponibilidad de la información por las entidades autorizadas, validar la propiedad de la información en procesos de comunicación.
FT6	Volumen de validaciones.	1	Toma el valor definido.	Se refiere a la cantidad de validaciones a realizar en cierta acción en el sistema.
FT7	Reutilización.	1	Toma el valor definido.	Se refiere a la capacidad o posibilidad de la reutilización de código ya implementado en otro requisito o sistema.
FT8	Utilización de estándares.	0.5	Toma el valor definido.	Necesidad de implementar estándares asociado al requisito.
FT9	Integración con dispositivos externos	0.5	Toma el valor definido.	Necesidad de integración de la implementación del requisito con hardware externo (lector de huella, lector de código de barra, cámaras, etc.).
FT10	Utilización de librerías externas.	1	Toma el valor definido.	Necesidad de incorporación de librerías, APIs, componentes desarrollados por terceros para completar el funcionamiento del requisito.
FT11	Interoperabilidad.	0.5	Toma el valor definido.	Capacidad de operación del requisito en otras plataformas y dispositivos.
FT12	Entrenamiento especial de usuario.	0.5	Toma el valor definido.	Se requiere de entrenamiento especial para los usuarios finales en la utilización del requisito.

4.2.5. Factores ambientales

El factor ambiental indica el grado en que el ambiente de pruebas influye sobre la productividad y se define mediante las seis variables ambientales siguientes: herramientas de pruebas, pruebas del desarrollo, bases de pruebas, ambiente de desarrollo, ambiente de pruebas y el repositorio de pruebas. Para estimar el factor ambiental se define para cada variable ambiental una escala nominal; el valor seleccionado para la variable debe ser uno de los propuestos en la escala, ya que los valores intermedios no se permiten. (Torres Ricaurte, 2013, pág. 21)

Tabla 18 Factores ambientales

ID	Factor ambiental	Peso (0,5-2)
F1	Familiar con RUP	1.5
F2	Experiencia en la aplicación	0,5
F3	Experiencia con orientación a objetos	1
F4	Capacidades de análisis	0.5
F5	Motivación	1
F6	Estabilidad de requisitos	2
F7	Trabajadores a tiempo parcial	-1
F8	Lenguaje complejo	-1
F9	Distribución del equipo geográficamente	-1
F10	Dificultad del lenguaje de programación	-1

A continuación se describe la los valores de ponderación de los factores ambientales:

Tabla 19 Descripción de los valores de ponderación de factores ambientales

Valor	Descripción
3	Ausencia del factor ambiental. (Mejor caso)
2	Presencia o influencia parcial del factor ambiental.
1	Presencia o influencia total del factor ambiental. (peor caso)

4.2.6. Métricas a medir para realizar la estimación

1. Las métricas a evaluar son:
2. Puntos de Requisitos Sin Ajustar (PRSA).
3. Factor de Peso del Actor Sin Ajustar (FPASA).

4. Factor de Peso de Requisitos Sin Ajustar (FPRSA).

- Puntos de Requisitos Sin Ajustar (PRSA): $PRSA = FPASA + FPRSA$
- Factor de Peso del Actor Sin Ajustar (FPASA): $FPASA = \sum_{i=1}^n \text{Peso}_i * \text{Actor}_i$
- Factor de Peso de Requisitos Sin Ajustar (FPRSA): $FPRSA = \text{Peso}$
- Factor de Complejidad Técnica (FCT): $FCT = 0,6 + (\sum_{i=1}^n \text{Peso}_i * \text{Valor}_i) / 100$
- Factor Ambiental (FA): $FA = 1,4 - (0,03 * \sum_{i=1}^n \text{Peso}_i * \text{Valor}_i)$
- Puntos de Requisitos Ajustado (PRA): $PRA = PRSA * FTC * FA$
- El esfuerzo estimado en horas persona: $E = PRA * HH$
- HH = 10 horas

4.2.7. Porcentaje de actividades por ciclo de vida

Con el propósito de obtener la duración a cada fase del ciclo de vida del proyecto, considerando las adaptaciones del modelo de estimación aplicadas en el sistema web de la institución Patricio Espinoza Bermeo, la cual está orientada a la gestión de proceso de matriculación estudiantil,

Para el ciclo de vida de proyectos bajo el enfoque de desarrollo tradicional, se han definido los siguientes porcentajes:

Tabla 20 Porcentaje de actividades por ciclo de vida

Fase	Porcentaje
Actividades de Gestión	20%
Modelado del negocio	10%
Requerimientos	10%
Análisis y Diseño	10%
Implementación(Codificación)	30%
Pruebas	10%
Despliegue	10%
Soporte	NA

Se determinaron los valores de porcentaje tomando en cuenta los porcentos de distribución para el esfuerzo total en el desarrollo del sistema propuesto por (Gómez, 2019), en el método inicial ya que en la institución no se tiene información histórica que sirva para obtener un valor de porcentaje propio. Los porcentos se irán

ajustando conforme existan datos históricos de la duración real de cada proceso en un periodo de tiempo adecuado por ende los porcentos se irán ajustando con respecto al método descrito anteriormente.

En la siguiente tabla se establecerá los resultados al implementar el método de estimación para el sistema web de la institución Patricio Espinoza Bermeo:

Ilustración 16 Método de estimación

ID Rquisito	Complejidad	FACTORES TÉCNICOS														FACTORES AMBIENTALES										FA	PAR	HH	E(Horas)					
		A1	A2	A3	FPASA	FPRSA	PRSA	FT1	FT2	FT3	FT4	FT5	FT6	FT7	FT8	FT9	FT10	FT11	FT12	FCT	FA1	FA2	FA3	FA4	FA5					FA6	FA7	FA8	FA9	FA10
RF1	Baja	1	0	0	0,5	3	3,5	1	1	1	1	1	0	1	0	0	1	1	1	0,71	1	1	0	0	1	0	1	0	1	1	1,33	3,27	10,00	32,69
RF1.1	Baja	1	0	0	0,5	3	3,5	1	1	1	1	1	0	1	0	0	1	1	1	0,71	1	1	0	0	1	0	1	0	1	1	1,33	3,27	10,00	32,69
RF1.2	Baja	1	0	0	0,5	3	3,5	1	1	1	1	1	0	1	0	0	1	1	1	0,71	1	1	0	0	1	0	1	0	1	1	1,33	3,27	10,00	32,69
RF1.3	Baja	1	0	0	0,5	3	3,5	1	1	1	1	1	0	1	0	0	1	1	1	0,71	1	1	0	0	1	0	1	0	1	1	1,33	3,27	10,00	32,69
RF1.4	Baja	1	0	0	0,5	3	3,5	1	1	1	1	1	0	1	0	0	1	1	1	0,71	1	1	0	0	1	0	1	0	1	1	1,33	3,27	10,00	32,69
RF2.1	Baja	1	0	0	0,5	3	3,5	1	1	1	1	1	0	1	0	0	1	1	1	0,71	1	1	0	0	1	0	1	0	1	1	1,33	3,27	10,00	32,69
RF2.2	Baja	1	0	0	0,5	3	3,5	1	1	1	1	1	0	1	0	0	1	1	1	0,71	1	1	0	0	1	0	1	0	1	1	1,33	3,27	10,00	32,69
RF2.3	Baja	1	0	0	0,5	3	3,5	1	1	1	1	1	0	1	0	0	1	1	1	0,71	1	1	0	0	1	0	1	0	1	1	1,33	3,27	10,00	32,69
RF2.4	Baja	1	0	0	0,5	3	3,5	1	1	1	1	1	0	1	0	0	1	1	1	0,71	1	1	0	0	1	0	1	0	1	1	1,33	3,27	10,00	32,69
RF3	Media	1	0	0	0,5	6	6,5	1	1	1	1	1	0	1	0	0	1	1	1	0,71	1	1	0	0	1	0	1	0	1	1	1,33	6,07	10,00	60,72
																	TOTAL DE ESFUERZO													354,93				

Tabla 21 Porcentaje de actividades por ciclo de vida para el sistema web

Fase	Porcentaje	Tiempo (Horas)
Actividades de Gestión	20%	70,90
Modelado del negocio	10%	35,49
Requerimientos	10%	35,49
Análisis y Diseño	10%	35,49
Implementación(Codificación)	30%	106,48
Pruebas	10%	35,49
Despliegue	10%	35,49
Soporte	No aplica	

La ilustración 16 muestra todo el esfuerzo para desarrollar el sistema es de 354.93 horas/ hombre.

Teniendo en cuenta que el sueldo mínimo de un ingeniero en sistemas en el ecuador asciende a 8 USD/hora, el presupuesto que se necesita para poder desarrollar este sistema web, asciende a 2.839,44 USD. Además, recalando que un mes de trabajo consta de 160h, el software se construyó en un tiempo estimado de 4 meses aproximadamente y se aplica todo el mes de vida tradicional para un proceso de desarrollo de software para los recursos. Por ello al ser un proyecto de titulación da un beneficio a la unidad educativa, ya que al ser una institución de recursos limitados se ahorrará este presupuesto que posiblemente se lo podría destinar a una empresa externa para el desarrollo del sistema.

4.3. Casos de prueba

En esta parte se realizará la comparación de las características de sistemas por lo que se toman los requisitos funcionales y no funcionales del sistema. Por medio de tablas se detallará los requisitos funcionales y no funcionales usados como objeto de prueba.

4.4. Pruebas de caja negra y validaciones

Las pruebas de caja negra son una técnica de pruebas de software que se centra en verificar el comportamiento externo de una aplicación según sus requisitos, sin conocer la estructura ni la lógica del código interno. Se utilizan principalmente para validar que el sistema cumple con sus especificaciones funcionales. (Giró Gracia & Sancho Gil, 2022).

Crear alumno

Tabla 22 Identificación de las clases de equivalencias:(Crear Alumno)

Parámetro de entrada	Regla Heurística a aplicar	Clases válidas (CV)	Clases inválidas (CI)
Nombre Completo	Llenar campo con el nombre completo del estudiante	CV1: Llenar el campo Nombre	CI1: No llenar el campo
Sexo	Seleccionar el campo correcto respectivamente	CV2: Seleccionar Masculino CV3: Seleccionar Femenino	CI2: No llenar el campo
Cedula	Llenar el campo con la cedula del estudiante, se valida que la cedula tenga 10 dígitos	CV4: Llenar el campo	CI3: No llenar el campo
País	Llenar campo con el país de nacimiento del estudiante	CV5: Llenar el campo	CI4: No llenar el campo
Fecha de nacimiento	Llenar campo con la fecha de nacimiento del estudiante	CV6: Llenar el campo fecha de nacimiento	CI5: No llenar el campo
Provincia	Llenar campo con la provincia donde reside el estudiante	CV7: Llena el campo	CI6: No llenar el campo
Cantón	Llenar campo con el cantón donde reside el estudiante	CV8: Llenar el campo	CI7: No llenar el campo
Dirección	Llenar campo con la dirección exacta donde vive	CV9: Llena el campo con la dirección completa del estudiante	CI8: No llenar el campo

Tabla 23 Casos de prueba para clases validad (Crear Alumno)

Nombre Completo	Cedula	Sexo	Fecha de nacimiento	País	Provincia	Cantón	Dirección	Clases válidas cubiertas
Carlos Alberto Terán Hinojosa	1711465211	Masculino	25/8/2002	Ecuador	Pichincha	Quito	Av 12 de octubre y Colon	CV1 CV2 CV3 CV4 CV5 CV6 CV7 CV8 CV9 CV10
Martina Angelina Trujillo Córdova	1725489565	Femenino	25/8/2005	Ecuador	Pichincha	Quito	Av Mariscal sucre y chilibulo	CV1 CV2 CV3 CV4 CV5 CV6 CV7 CV8 CV9 CV10

Tabla 24 Casos de prueba para clases inválida

Nombre Completo	Sexo	Cedula	País	Fecha de nacimiento	Provincia	Cantón	Dirección	Clases válidas cubiertas
Se deja vacío	Masculino	1711465211	Ecuador	Se deja vacío	Pichincha	Quito	Av 12 de octubre y Colon	CI1, CI5
Martina Angelina Trujillo Córdova	Femenino	Se deja vacío	Se deja vacío	Se deja vacío	Pichincha	Quito	Se deja vacío	CI3, CI4, CI8
Carlos Alberto Terán Hinojosa	Se deja vacío	Se deja vacío	Ecuador	25/8/2008	Pichincha	Quito	Av 12 de octubre y Colon	CI2
Martina Angelina Trujillo Córdova	Femenino	1753538273	Ecuador	Se deja vacío	Pichincha	Se deja vacío	Av Mariscal sucre y chilibulo	CI5, CI7
Carlos Alberto Terán Hinojosa	Masculino	1767562627	Se deja vacío	25/8/2014	Pichincha	Quito	Se deja vacío	CI4, CI8
Martina Angelina Trujillo Córdova	Se deja vacío	1781586981	Ecuador	Se deja vacío	Pichincha	Quito	Se deja vacío	CI2, CI5, CI8
Carlos Alberto Terán Hinojosa	Masculino	1795611335	Ecuador	25/8/2020	Pichincha	Quito	Av 12 de octubre y Colon	CI3, CI6
Martina Angelina Trujillo Córdova	Se deja vacío	1809635689	Se deja vacío	Se deja vacío	Se deja vacío	Quito	Se deja vacío	CI2, CI4, CI5, CI6, CI8
Carlos Alberto Terán Hinojosa	Se deja vacío	1823660043	Ecuador	Se deja vacío	Pichincha	Quito	Se deja vacío	CI2, CI5, CI8
Martina Angelina Trujillo Córdova	Femenino	Se deja vacío	Ecuador	25/8/2029	Se deja vacío	Quito	Av Mariscal sucre y chilibulo	CI3, CI6

Matriz de casos de prueba:

Tabla 25 Matriz de casos de prueba

Clases de equivalencia	Resultados Esperados	Resultados Real
CV1, CV2, CV3, CV4, CV5, CV6, CV7, CV8, CV9	Estudiante Registrado	Estudiante Registrado
CV1, CV2, CV3, CV4, CV5, CV6, CV7, CV8, CV9	Estudiante Registrado	Estudiante Registrado
CI1	Mensaje de "Falta llenar el campo Nombre"	Mensaje de "Falta llenar el campo Nombre"
CI2	Mensaje de "Falta llenar el campo Sexo"	Mensaje de "Falta llenar el campo Sexo"
CI3	Mensaje de "Falta llenar el campo Cedula"	Mensaje de "Falta llenar el campo Cedula"
CI4	Mensaje de "Falta llenar el campo país"	Mensaje de "Falta llenar el campo país"
CI5	Mensaje de "Falta llenar el campo Fecha de nacimiento"	Mensaje de "Falta llenar el campo Fecha de nacimiento"
CI6	Mensaje de "Falta llenar el campo provincia"	Mensaje de "Falta llenar el campo provincia"
CI7	Mensaje de "Falta llenar el campo cantón"	Mensaje de "Falta llenar el campo cantón"
CI8	Mensaje de "Falta llenar el campo Dirección"	Mensaje de "Falta llenar el campo Dirección"

Editar alumno

Identificación de clases equivalente

Tabla 26 Identificación de las clases de equivalencias:(Editar Alumno)

Parámetro de entrada	Regla Heurística a aplicar	Clases válidas (CV)	Clases inválidas (CI)
Nombre Completo	Cambiar el contenido del campo con completo del estudiante	CV1: Cambiar el campo Nombre	CI1: Dejar vacío
Sexo	Cambiar el campo correcto respectivamente	CV2: Seleccionar Masculino CV3: Seleccionar Femenino	CI2: Dejar vacío
Cedula	Cambiar el campo con la cedula del estudiante, se valida que la cedula tenga 10 dígitos	CV4: Cambiar el campo	CI3: Dejar vacío
País	Cambiar campo del país de nacimiento del estudiante	CV5: Cambiar el campo	CI4: Dejar vacío
Fecha de nacimiento	Cambiar campo de la fecha de nacimiento del estudiante	CV6: Cambiar el campo fecha de nacimiento	CI5: Dejar vacío
Provincia	Cambiar campo de la provincia donde reside el estudiante	CV7: Cambiar el campo	CI6: Dejar vacío
Cantón	Cambiar campo del cantón donde reside el estudiante	CV8: Cambiar el campo	CI7: Dejar vacío
Dirección	Cambiar campo de la dirección exacta donde vive	CV9: Cambiar el campo con la dirección completa del estudiante	CI8: Dejar vacío

Tabla 27 Casos de prueba para clases validad (Editar Alumno)

Nombre Completo	Cedula	Sexo	Fecha de nacimiento	País	Provincia	Cantón	Dirección	Clases válidas cubiertas
Juan Diego Procel Pérez	1711465211	Masculino	15/8/2002	Ecuador	Pichincha	Quito	Av General Rumiñahui	CV1 CV2 CV3 CV4 CV5 CV6 CV7 CV8 CV9 CV10
Ana María Castro Castillo	1725489565	Femenino	05/8/2005	Ecuador	Pichincha	Quito	Av Mariscal sucre	CV1 CV2 CV3 CV4 CV5 CV6 CV7 CV8 CV9 CV10

Tabla 28 Casos de prueba para clases inválida

Nombre Completo	Sexo	Cedula	País	Fecha de nacimiento	Provincia	Cantón	Dirección	Clases válidas cubiertas
Juan Diego Procel Pérez	Masculino	Se deja vacío	Ecuador	15/8/2002	Pichincha	Quito	Av General Rumiñahui	CI3
Ana María Castro Castillo	Femenino	1725489565	Se deja vacío	05/8/2005	Se deja vacío	Quito	Av Mariscal sucre	CI4, CI6
Se deja vacío	Masculino	1711465211	Ecuador	15/8/2002	Se deja vacío	Quito	Av General Rumiñahui	CI1, CI6
Ana María Castro Castillo	Femenino	1725489565	Se deja vacío	05/8/2005	Pichincha	Quito	Av Mariscal sucre	CI4
Juan Diego Procel Pérez	Masculino	1711465211	Ecuador	15/8/2002	Pichincha	Se deja vacío	Av General Rumiñahui	CI7
Ana María Castro Castillo	Femenino	1725489565	Se deja vacío	05/8/2005	Pichincha	Quito	Av Mariscal sucre	CI4
Juan Diego Procel Pérez	Masculino	Se deja vacío	Ecuador	15/8/2002	Pichincha	Se deja vacío	Av General Rumiñahui	CI3, CI7
Ana María Castro Castillo	Femenino	1725489565	Se deja vacío	05/8/2005	Pichincha	Quito	Av Mariscal sucre	CI4
Juan Diego Procel Pérez	Masculino	1711465211	Se deja vacío	15/8/2002	Pichincha	Se deja vacío	Av General Rumiñahui	CI4, CI7
Ana María Castro Castillo	Femenino	1725489565	Se deja vacío	05/8/2005	Pichincha	Quito	Av Mariscal sucre	CI4

Matriz de casos de prueba:

Tabla 29 Matriz de casos de prueba (Editar Alumno)

Clases de equivalencia	Resultados Esperados	Resultados Real
CV1, CV2, CV3, CV4, CV5, CV6, CV7, CV8, CV9	Estudiante Actualizado	Estudiante Actualizado
CV1, CV2, CV3, CV4, CV5, CV6, CV7, CV8, CV9	Estudiante Actualizado	Estudiante Actualizado
CI1	Mensaje de "Falta llenar el campo Nombre"	Mensaje de "Falta llenar el campo Nombre"
CI2	Mensaje de "Falta llenar el campo Sexo"	Mensaje de "Falta llenar el campo Sexo"
CI3	Mensaje de "Falta llenar el campo Cedula"	Mensaje de "Falta llenar el campo Cedula"
CI4	Mensaje de "Falta llenar el campo país"	Mensaje de "Falta llenar el campo país"
CI5	Mensaje de "Falta llenar el campo Fecha de nacimiento"	Mensaje de "Falta llenar el campo Fecha de nacimiento"
CI6	Mensaje de "Falta llenar el campo provincia"	Mensaje de "Falta llenar el campo provincia"
CI7	Mensaje de "Falta llenar el campo cantón"	Mensaje de "Falta llenar el campo cantón"
CI8	Mensaje de "Falta llenar el campo Dirección"	Mensaje de "Falta llenar el campo Dirección"

Eliminar alumno

Identificación de clases equivalente

Tabla 30 Identificación de las clases de equivalencias:(Eliminar Alumno)

Parámetro de entrada	Regla Heurística a aplicar	Clases válidas (CV)	Clases inválidas (CI)
Cedula	Ingresar una cedula de algún estudiante ingresado en el sistema	CV1: Ingresar una cedula valida	CI1: Ingresar cedula que no existe
Formato Cedula	Ingresar una cedula con formato correcto	CV2: Ingresar cedula con longitud numérica válida	CI2: Cedula incorrecta
Estado de matrícula	Validar que el usuario no tenga una matrícula activa	CV3: Alumno sin matricula activa	CI3: Alumno con matrícula activa
Permiso del usuario	Validar que el usuario tenga los permisos necesarios	CV4: Usuario con rol administrativo	CI4: Usuario sin rol

Tabla 31 Caso de prueba para clases válidas

Caso de prueba	Descripción del caso de prueba	Entrada	Resultado Esperado	Clase Valida
CP1	Eliminar alumno con cedula existente	1795611335	Alumno eliminado correctamente	CV1
CP2	Ingreso de cedula con formato correcto	1767562627	Alumno eliminado correctamente	CV2
CP3	Alumno sin matricula activa	1753538273	Alumno eliminado correctamente	CV3
CP4	Usuario con rol de administrador	usuario: jairomogro12@gmail.com	Proceso permitido, alumno eliminado	CV4

Tabla 32 Caso de uso para calses invalidas

Caso de prueba	Descripción del caso de prueba	Entrada	Resultado Esperado	Clase Valida
CP5	Ingresar una cedula no existente	1758478556	"Alumno no encontrado"	CI1
CP6	Ingreso de cedula con formato correcto	AV125486	Cedula incorrecta, alumno no encontrado	CI2
CP7	Ingresar alumno con matrícula activa	1753538273 (matrícula activa)	Alumno matriculado no se puede eliminar	CI3
CP8	Usuario con rol de padre de familia	usuario: edgar12@gmail.com	Acceso no autorizado	CI4

Matriz de casos de prueba:

Tabla 33 Matriz de casos de prueba (Eliminar Alumno)

Clases de equivalencia	Resultados Esperados	Resultados Real
CV1, CV2, CV3, CV4	Alumno eliminado correctamente	Alumno eliminado correctamente
CV1, CV2, CV3, CV4,	Alumno eliminado correctamente	Alumno eliminado correctamente
CI1	Mensaje de "Alumno no encontrado"	Mensaje de "Alumno no encontrado"
CI2	Mensaje de "Cedula invalida"	Mensaje de "Cedula invalida"
CI3	Mensaje de "No se puede eliminar alumno"	Mensaje de "No se puede eliminar alumno"
CI4	Mensaje de "Acceso denegado "	Mensaje de "Acceso denegado "

Listar alumno

Identificación de clases equivalente

Tabla 34 Identificación de las clases de equivalencias:(Listar Alumno)

Parámetro de entrada	Regla Heurística a aplicar	Clases válidas (CV)	Clases inválidas (CI)
Nombre Cedula	El sistema listo automáticamente cuando se crea un alumno	CV1: Sistema lista todos los alumnos	CI1: Sistema no muestra ningún alumno
usuario	Usuario con permisos para poder acceder	CV2: Usuario Autorizado	CI2: Usuario no autorizado

Tabla 35 Caso de prueba para clases válidas

Caso de prueba	Descripción del caso de prueba	Entrada	Resultado Esperado	Clase Valida
CP9	Listar todos los alumnos	Sin filtro	Se muestra los alumnos registrados	CV1
CP10	Usuario con permisos para acceder a la lista	Usuario: jairomogro12@gmail.com	Se muestra los alumnos registrados	CV2

Tabla 36 Caso de uso para calses invalidas

Caso de prueba	Descripción del caso de prueba	Entrada	Resultado Esperado	Clase Valida
CP11	Sistema no muestra alumnos	Sin filtros	No se muestra ningún alumno	CI1
CP12	Usuario sin permisos para acceder a la lista	Usuario: pdero12@gmail.com	Acceso denegado no puede ver ese apartado	CI2

Matriz de casos de prueba:

Tabla 37 Matriz de casos de prueba (Listar Alumno)

Clases de equivalencia	Resultados Esperados	Resultados Real
CV1, CV2	Lista de alumnos registrados en el sistema	Lista de alumnos registrados en el sistema
CV1, CV2	Lista de alumnos registrados en el sistema	Lista de alumnos registrados en el sistema
CI1	No se muestra ningún alumno porque no hay registros	No se muestra ningún alumno porque no hay registros
CI2	No tiene permisos para poder ver esa sección	No tiene permisos para poder ver esa sección

Casos de prueba de GUI

Tabla 38 Prueba Autenticación de usuario

ID	CP01
Requerimiento Asociado	RF1
Nombre	Autenticación de usuario
Descripción	Encargado de validar las credenciales del usuario que accederá al sistema.
Peor Valor	Las credenciales no son correctas
Valor planificado	Usuario ingresado Correctamente
Tipo de usuario	Administrador
Esta (cumple/no cumple)	Cumple

Tabla 39 Prueba Gestión de usuario

ID	CP02
Requerimiento Asociado	RF2
Nombre	Gestión de usuario
Descripción	Encargado de tratar los datos requeridos de un usuario para el sistema.
Peor Valor	Falta llenar campos
Valor planificado	Campos llenados correctamente
Tipo de usuario	Administrador
Esta (cumple/no cumple)	Cumple

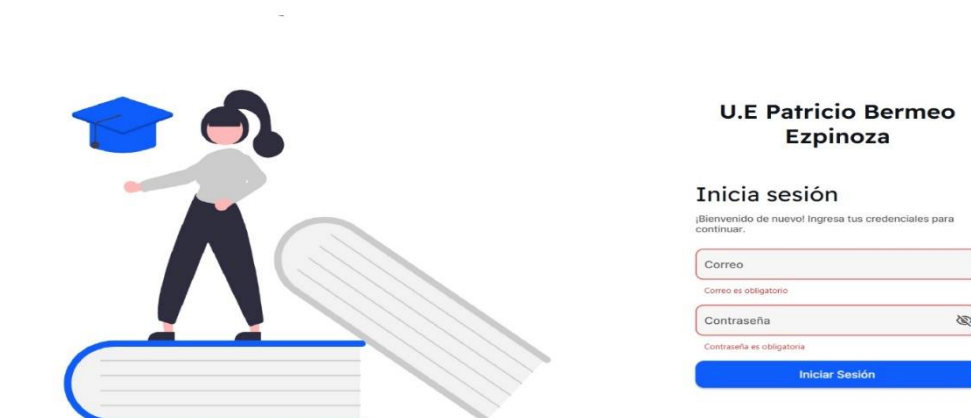
Tabla 40 Prueba gestionar matriculación estudiantil

ID	CP03
Requerimiento Asociado	RF3
Nombre	Gestionar matriculación estudiantil
Descripción	Encargado de ingresar y tratar la información persona de cada estudiante acorde a los requerimientos dados por la institución.
Peor Valor	Falta campos por llenar, para la creación de un registro
Valor planificado	Registro creado correctamente
Tipo de usuario	Administrador
Esta (cumple/no cumple)	Cumple

4.5. Validación de la implementación del sistema


En las siguientes ilustraciones se mostrará algunas de las validaciones que se realizó al sistema.

Ilustración 17 Validación Login – Mensaje de error al ingresar



En la ilustración 17 se valida el inicio de sesión para el sistema, en este caso no se está ingresando ningún correo ni contraseña, por lo que el sistema arroja mensajes de correo es obligatorio y contraseña es obligatoria, ya que sin estas credenciales no se podrá ingresar al sistema.

Ilustración 18 Mensaje de campos obligatorios creación de usuario



**U.E Patricio Bermeo
Ezpiñoza**

Crear cuenta

Rellenar todos los campos por favor

Correo es obligatorio

Password es obligatorio

Confirmar password es obligatorio

[Create Account](#)

¿Ya tienes una cuenta? [Iniciar sesión](#)

En la ilustración 18 tenemos el apartado de crear una cuenta para poder tener un registro en el sistema y de esta manera poder ingresar sin ningún problema, es por esto que nos notifica que se necesita llenar los campos obligatoriamente, de lo contrario nos arrojará un mensaje como es el caso de correo obligatorio, contraseña es obligatoria, y el confirmar contraseña también es obligatorio.

4.6. Conclusión del capítulo

Dentro de este capítulo se logró detectar y solucionar errores presentes dentro del sistema web. También se pudo evaluar la viabilidad de su implementación, además se realizaron pruebas de las distintas funcionalidades con el fin de verificar el funcionamiento correcto del sistema.

Se aplicó el método de estimación de esfuerzos por el cual nos permite ver el esfuerzo para realizar o mantener un sistema por su complejidad a partir de las funcionalidades e interacciones que tiene el usuario con el sistema.

CONCLUSIONES

El especificar los procesos de negocio orientados a la automatización, nos facilitó comprender detalladamente las debilidades existentes, como actividades que dificultaban a un mejor desarrollo administrativo dentro de la Unidad Educativa Patricio Espinoza Bermeo.

El tener un análisis en el proceso de matriculación, facilitó detectar algunos fallos, tales como la repetición de información, demoras en el ámbito administrativos y deficiencia con el manejo de los datos de estudiantes. Esto dio como consecuencia la necesidad de poder tener un sistema automatizado para la gestión administrativa y evitar errores o pérdida de información.

Se tuvo una investigación de deferentes metodologías y sobre todo herramientas actuales para el desarrollo del sistema, llegando a optar por tecnologías y bases de datos relacionadas de mejor manera, garantizando un sistema amigable y sobre todo accesible para los usuarios.

Los requerimientos funcionales y no funcionales se definieron en base a las necesidades de la institución, estableciendo estándares de calidad que serán integrados en el diseño del sistema, para de esta manera asegurar un cumplimiento de los objetivos que se plantearon a un inicio y sobre todo probando la factibilidad del sistema.

El análisis de costos relacionado con el desarrollo del sistema web a través del método de Puntos Casos de Uso determinó que el esfuerzo para el desarrollo del sistema fue de 354.93 horas hombres y el costo de realizarlo es de 2839.44 dólares.

Las pruebas de caja negra realizadas ayudaron a corregir errores y permitió validar el funcionamiento correcto del sistema, demostrando de esta manera que el sistema puede ser utilizado en el área administrativa de la Unidad Educativa Patricio Espinoza Bermeo.

RECOMENDACIONES

Capacitar a al personal administrativo de la Unidad Educativa Patricio Espinoza Bermeo, con la finalidad de tener el manejo correcto y eficaz del sistema, especificación cada una de sus funcionalidades estipuladas anteriormente.

Se sugiere implementar nuevos procesos que pueda realizar dentro del sistema, como sistema de notas para los estudiantes o a su vez una plataforma por donde se pueda subir y enviar deberes educativos.

Implementar un módulo donde los docentes puedan realizar sus planificaciones académicas, actividades y algunos recursos didácticos.

Se sugiere implementar un módulo donde los estudiantes van a poder acceder a material didáctico, material de apoyo para su proceso estudiantil.

Se sugiere el desarrollo de una aplicación móvil como complemento del sistema web, permitiendo a usuario que se le haga más accesible usar la aplicación desde su teléfono.

Implementar notificaciones sobre fechas festivas o a su vez situaciones problemáticas sobre el estudiante que les llegara a cada padre de familia a su dispositivo móvil.

BIBLIOGRAFÍA

- Barrientos Rodríguez, M., Zacca González, G., Castro Peraza, M., Álvarez Gainza, D., Vidal Ledo, M., & Valdés Santiago, D. (2 de febrero de 2022). Metodología para el desarrollo del sistema Web para la gestión de los programas de maestría del Instituto "Pedro Kouri". *Revista Cubana de Informática Médica*. Recuperado el 20 de febrero de 2025, de <http://scielo.sld.cu/pdf/rcim/v14n2/1684-1859-rcim-14-02-e537.pdf>
- Cillero, M. (05 de marzo de 2023). *Diagrama de secuencia*. Recuperado el 20 de marzo de 2025, de <https://manuel.cillero.es/doc/metodologia/metrica-3/tecnicas/diagrama-de-interaccion/diagrama-de-secuencia/>
- Cristancho, F. (26 de julio de 2022). *¿Qué es Flutter?* Recuperado el 26 de febrero de 2025, de <https://talently.tech/blog/que-es-flutter/>
- Editorial Etecé. (26 de Septiembre de 2024). *Metodología*. Recuperado el 20 de febrero de 2025, de <https://concepto.de/metodologia/>
- Enríquez, F., Fierro, S., Flores, B., Imbaquingo Esparza, D., & Michelena, J. (29 de julio de 2023). Influencia del patrón modelo vista controlador (MVC) en la seguridad, interoperabilidad y usabilidad del ciclo de vida de un sistema informático. *Ingeniería y Ciencias Aplicadas en la Industria (EASI)*. Recuperado el 15 de marzo de 2025, de <https://revistas.ug.edu.ec/index.php/easi/en/article/view/821>
- Escuelas Ecuador. (30 de octubre de 2023). *Escuela Ing. Patricio Espinosa Bermeo*. Recuperado el 15 de enero de 2025, de <https://www.escuelasecuador.com/ing-patricio-espinosa-bermeo-pichincha-quito-17h00546>
- Fernández Marín, M. Á., & González Tolmo, D. (Septiembre de 2020). Propuesta de fusión de una metodología para multimedia con el proceso unificado evidenciado en un caso real. *Revista Metropolitana de Ciencias Aplicadas*. Recuperado el 5 de marzo de 2025, de <https://www.redalyc.org/pdf/7217/721778107017.pdf>

Flutter Docs. (4 de septiembre de 2023). *Descripción general de la arquitectura de Flutter*. Recuperado el 26 de febrero de 2025, de <https://docs.flutter.dev/resources/architectural-overview>

García de Zúñiga, F. (30 de Octubre de 2024). *¿Qué es Visual Studio Code y cuáles son sus ventajas?* Recuperado el 26 de febrero de 2025, de <https://www.arsys.es/blog/que-es-visual-studio-code-y-cuales-son-sus-ventajas>

Giró Grácia, X., & Sancho Gil, J. (24 de Enero de 2022). La Inteligencia Artificial en la educación: Big data, cajas negras y solucionismo tecnológico. *Revista Latinoamericana de Tecnología Educativa*. Recuperado el 5 de abril de 2025, de Universidad de Extremadura: <https://dehesa.unex.es/server/api/core/bitstreams/77f756cf-c49d-4375-b928-2cc590ae809b/content>

Gómez, J. (14 de Febrero de 2019). *Método de Estimación Puntos Casos de Uso (Use Case Points)*. Recuperado el 25 de abril de 2025, de <https://www.laboratorioti.com/2013/02/14/metodo-de-estimacion-puntos-casos-de-uso-use-case-points/>

González Hernández, A. (15 de Marzo de 2021). Identificación de procesos de negocio. *Ingeniería Industrial* . Recuperado el 5 de marzo de 2025, de <https://www.redalyc.org/pdf/3604/360433558004.pdf>

Hernández Pico, P. A., & Samada Grasst, Y. (01 de Septiembre de 2021). La educación inclusiva desde el marco legal educativo en el Ecuador. *Revista de Ciencias Humanísticas y Sociales (ReHuSo)*. Recuperado el 3 de diciembre de 2024, de http://scielo.senescyt.gob.ec/scielo.php?pid=S2550-65872021000300063&script=sci_arttext

Kimmel, P. (18 de Marzo de 2019). *Manual de UML*. Recuperado el 5 de marzo de 2025, de https://www.academia.edu/29446479/Manual_de_UML_-_Paul_Kimmel.pdf

- Kiss, T. (20 de Mayo de 2024). *Investigación aplicada*. Recuperado el 18 de febrero de 2025, de <https://concepto.de/investigacion-aplicada/>
- Kuz, A., Falco, M., & Giandini, R. (30 de 03 de 2018). Comprendiendo la Aplicabilidad de Scrum en el Aula: Herramientas y Ejemplos. *Revista Iberoamericana de Tecnología en Educación y Educación en Tecnología*. Recuperado el 20 de febrero de 2025, de TE&ET: <https://portal.amelica.org/ameli/journal/24/2414011/html/>
- Landi Paucar, Á. C. (2019). *Sistema informático para el registro de matrículas de estudiantes, cobro de pensiones y pago de sueldos a docentes en la unidad educativa Louis Víctor de Broglie*. Quito: Universidad Metropolitana.
- Lizcano Bueno , L. I. (18 de junio de 2016). UML: Un Lenguaje de Modelo de Objetos. *Respuestas Journal of Engineering Sciences*. Recuperado el 25 de febrero de 2025, de <https://revistas.ufps.edu.co/index.php/respuestas/article/view/527>
- López Mora, S. (17 de 05 de 2020). *Firestore: qué es, para qué sirve, funcionalidades y ventajas*. Obtenido de <https://digital55.com/blog/que-es-firebase-funcionalidades-ventajas-conclusiones/#:~:text=Firestore%20de%20Google%20es%20una,r%C3%A1pido%20trabajar%20en%20el%20desarrollo.>
- Lucidchart. (12 de junio de 2019). *Tutorial de diagrama de actividades UML*. Recuperado el 6 de marzo de 2025, de <https://www.lucidchart.com/pages/es/tutorial-diagrama-de-actividades-uml>
- Martínez Cordón, M. J. (30 de Junio de 2021). *¿Qué es el lenguaje de programación Dart?* Recuperado el 26 de febrero de 2025, de <https://www.hiberus.com/crecemos-contigo/que-es-el-lenguaje-de-programacion-dart/>
- Mayorga Ases, M. J., Sánchez Manobanda, K. P., Páliz Ibarra, S. J., & Melo Fiallos, D. F. (19-29 de Julio de 2023). *Gestión educativa y desempeño docente en las escuelas de Ecuador*. AlfaPublicaciones. Recuperado el 3 de diciembre de

2024, de
<https://alfapublicaciones.com/index.php/alfapublicaciones/article/view/374>

Navarrete Mendieta, G., & Mendieta García, R. C. (2020). Las tic y la educación ecuatoriana en tiempos de internet. *Revista Iberoamericana para la Investigación y el Desarrollo Educativo (RIDE)*. Recuperado el 15 de febrero de 2025, de <https://www.revistaespirales.com/index.php/es/article/view/220>

Northware. (26 de Mayo de 2022). *Requerimientos en el desarrollo de software y aplicaciones*. Recuperado el 8 de marzo de 2025, de <https://www.northware.mx/blog/requerimientos-en-el-desarrollo-de-software-y-aplicaciones/>

Rojas Borbor, G. R., & Tandazo Luna, J. S. (abril de 2019). *Implementación de un sistema de matriculación y gestión de calificaciones para la Unidad Educativa ATI II Pillahuaso*. Recuperado el 19 de febrero de 2025, de Universidad Politécnica Salesiana: <https://dspace.ups.edu.ec/bitstream/123456789/17388/1/UPS-GT002664.pdf>

Salazar Acosta, L. I. (octubre de 2018). *Implementación de sistema de matriculación y carnetización en la unidad educativa Pablo Picasso*. Recuperado el 3 de diciembre de 2024, de Universidad Politécnica Salesiana: <https://dspace.ups.edu.ec/handle/123456789/16844>

Siewe, F., & Ngounou, G. M. (27 de febrero de 2025). Sobre la ejecución y verificación en tiempo de ejecución de diagramas de actividad UML. *Multidisciplinary Digital Publishing Institute*. doi:<https://doi.org/10.3390/software4010004>

Sydle. (01 de Agosto de 2023). *¿Qué son las reglas de negocio y qué importancia tienen? Conoce algunos ejemplos*. Recuperado el 8 de marzo de 2025, de <https://www.sydle.com/es/blog/reglas-de-negocio-5f6333be1e43744c69d995e0>

Systems, Sparx. (12 de junio de 2020). *Diagrama de Actividades UML 2*. Recuperado el 20 de febrero de 2025, de http://www.sparxsystems.com.ar/resources/tutorial/uml2_activitydiagram.html

Torres Ricaurte, D. M. (junio de 2013). *Un método para medir la productividad del equipo de pruebas en la estimación del esfuerzo de pruebas de software*. Recuperado el 28 de septiembre de 2020, de Universidad Nacional de Colombia:

<https://repositorio.unal.edu.co/bitstream/handle/unal/20798/30081835.2013.pdf?sequence=1&isAllowed=y>

Ucán Pech, J. P., Aguilar Vera, R. A., Díaz Mendoza, J. C., & Gómez Gómez, O. S. (2022). Faltas en el aprendizaje del modelado de clases y casos de uso: una revisión sistemática. *Revista Científica*, 46(1), 93-106. Recuperado el 15 de marzo de 2025, de <https://www.redalyc.org/journal/5043/504375197008/html/>

Zaman, Q. u., Nadeem, A., & Sindhu, M. A. (20 de Abril de 2020). Formalización del modelo de casos de uso: un enfoque basado en modelos. *Plos One*. doi:<https://doi.org/10.1371/journal.pone.0231534>

Zavala, M., González, I., & Vázquez, M. A. (2020). Modelo de innovación educativa según las experiencias de docentes y estudiantes universitarios. *Revista Iberoamericana para la Investigación y el Desarrollo Educativo (RIDE)*. Recuperado el 15 de febrero de 2025, de https://www.scielo.org.mx/scielo.php?script=sci_arttext&pid=S2007-74672020000100103

Zorrilla, A. (8 de 1 de 2021). *¿Cómo se realiza una investigación documental o bibliográfica?* Recuperado el 18 de febrero de 2025, de <https://campusidyd.com/investigacion-documental-o-bibliografica/>