

Sistema de información para la gestión administrativa del proceso de ajuste de matrícula en
la escuela de ingeniería de sistemas e informática

Esteban David Florez Tolosa , Dilan Alessandro Corredor Diaz, Carlos Daniel Sanjuan

Argote

Trabajo de Grado Optar al Título de Ingeniero de Sistemas

Director

Juan Ramon Pernaletе Maldonado

Ingeniero de Sistemas

Codirector

Luis Carlos Gomez Florez

Dr en Ingeniería de sistemas

Universidad Industrial de Santander

Facultad de Ingenierías Fisicomecánicas

Escuela de Ingeniería de Sistemas e Informática

Bucaramanga

2025

Dedicatoria

Dedico este proyecto, con profundo agradecimiento y cariño a mis padres. Gracias por haber sido mi guía y mi apoyo incondicional en cada etapa de mi vida y de mi formación académica. Por enseñarme, con su ejemplo, los valores y principios que han sido la base de mi crecimiento personal y profesional. Por sus consejos, por sus sacrificios y por el amor inagotable que siempre me han brindado. Este logro es también suyo.

Quiero dedicarlo, además, a la memoria de mi querido abuelo Reinaldo, quien siempre me apoyó y soñó con verme convertido en profesional. Aunque no pudo estar presente para compartir este momento, su amor, enseñanzas y motivación siguen siendo una inspiración en mi camino.

Agradecimientos

Al ingeniero Luis Carlos el cual nos brindo su mano para poder realizar este proyecto, al Ingeniero Juan Pernaletе por su paciencia y compromiso, a mis compañeros de trabajo los cuales aportaron en la realización de este proyecto

A cada uno de los profesores que me orientaron y brindaron el conocimiento para forjar las bases de mi carrera profesional

Contenido

Introducción	12
1. Planteamiento y justificación del problema	16
2. Objetivos	17
2.2 Objetivos específicos	18
3. Marco de referencia	18
3.1 Marco teórico	18
3.2 Marco conceptual	19
3.2.1 Sistema de información	19
3.3 Estado del arte	21
4.1 Fase de inicio:	23
4.1.1 Gestión del proyecto.	24
4.1.1.1 Plan de desarrollo del sistema.	24
4.1.1.1.1 Vista general del proyecto.	25
4.1.1.1.2 Organización del proyecto.	29
4.1.2 Modelado de la organización.	36
4.1.2.1 Definición del negocio.	36
4.1.2.2 Descripción del negocio actual .	38
4.1.2.3 Evaluación del estatus del negocio.	40
4.2 Fase de elaboración:	41
4.2.1 Requerimientos.	42
4.2.1.1 Análisis del problema.	43
4.2.1.2 Definición del sistema.	44
4.2.1.2.2 Requerimientos no funcionales.	46
4.2.1.2.3 Requerimientos complementarios.	47
4.2.1.4 Gestión y alcance del sistema.	47
4.2.1.3 Usuarios del sistema.	48
4.2.1.4.1 Requisitos de hardware.	48
4.2.1.4.2 Requisitos de software.	49
4.2.1.4.3 Requisitos de desempeño.	50
4.2.1.4.5 Requisitos de base de datos.	52
4.2.1.4.6 Requisitos de entradas.	52
4.2.1.4.7 Requisitos de salida.	53
4.2.1.4.8 Requisitos de documentación.	54
4.2.1.5 Modelos de caso de uso.	54
4.3 Fase de construcción:	55
4.3.1 Análisis y diseño.	56
4.3.1.1 Análisis del comportamiento del sistema.	56
4.3.1.2 Definición de arquitectura.	57
4.3.1.2.1 Vista funcional o de uso de la aplicación web.	57

4.3.1.3 Diseño de base de datos.	59
4.3.1.3.1 Diccionario de datos.	63
4.3.1.4 Diseño de interfaz gráfica.	66
4.3.1.4.1 Perfil de usuarios.	66
4.3.1.4.2 Mapa de navegación.	66
4.3.1.5 Diseño de servicios.	68
4.3.1.6 Prototipo de interfaz.	69
4.3.2 Implementación.	78
4.3.2.1 Modelo de implementación.	78
4.3.2.2 Plan de integración.	79
4.3.2.3 Implementación de componentes.	81
4.3.2.3.1 Creación de base de datos.	81
4.3.2.3.2 Creación de interfaz gráfica.	82
4.3.2.3.3 Configuración de servicios.	85
4.3.2.4 Integración de componentes.	90
4.3.3 Pruebas alfa.	90
4.3.3.1 Plan de pruebas.	90
4.3.3.2 Pruebas planeadas.	91
4.3.3.2.1 Pruebas de calidad.	92
4.4 Fase de transición	95
4.4.1 Implantación.	95
4.4.1.1 Plan de implantación.	95
4.4.1.2 Desarrollo de material de apoyo.	97
4.4.1.3 Pruebas beta.	97
4.4.1.4 Seguimiento post-instalación.	104
4.4.2 Análisis de costos y beneficios.	105
4.4.2.1 Costos.	106
4.4.2.2 Estudio de mercado.	108
4.4.2.3 Beneficios de la implantación del sistema.	110
5. Conclusiones	113
6. Recomendaciones a trabajos futuros	116
6.1 Implementar un sistema de ayuda interactivo para estudiantes y administrativos	116
6.2 Ampliación del sistema a otras escuelas e integración con plataformas institucionales	116
6.3 Obtención de permisos para acceso a información en tiempo real	116
6.4 Carga automática y gestión inteligente de información de los estudiantes	117
6.5 Mejoras en la infraestructura del servidor para soportar mayor carga	117
6.6 Sistema de recomendación de horarios y asignaturas con IA	118
Referencias bibliográficas	120
Apéndices	121

Lista de figuras

1.	Funciones de un sistema de información	20
2.	Visión del proyecto	25
3.	Equipo de soporte	29
4.	Distribución de iteraciones para cada fase	32
5.	Modelo BPMN ajuste de matrícula atención por zoom	36
6.	Diagrama de casos de uso del sistema propuesto	53
7.	Diagrama de casos de uso del sistema desarrollado	57
8.	Diseño base de datos nosql	59
9.	Mapa de navegación	66
10.	Prototipo de interfaz inicio de sesión	71
11.	Prototipo de interfaz registro	71
12.	Prototipo de interfaz inicio estudiante.	72
13.	Prototipo de interfaz horario estudiante	72
14.	Prototipo de interfaz crear solicitud estudiante	73
15.	Prototipo de interfaz solicitudes	73
16.	Prototipo de interfaz gestionar solicitud	74
17.	Prototipo de interfaz usuarios	74
18.	Prototipo de interfaz estadísticas	75
19.	Diagrama de componentes del sistema	76
20.	Diagrama de actividades: plan de integración	78
21.	Esquemas de datos en MongoDB	80
22.	Diseño responsive para móvil	81
23.	Diseño responsive para laptop	82
24.	Validación de formularios	83
25.	Toast de retroalimentación para el usuario	84
26.	Alertdialog con resumen de movimientos	84
27.	Confirmación acciones irreversibles	85
28.	Autenticación de usuario por correo electrónico	86
29.	Plan de implantación	92
30.	Vista de estadísticas generales	111
31.	Vista de estadística solicitudes completadas por franja horaria	112

Lista de tablas

1.	Roles y funciones dentro del equipo de trabajo	31
2.	Estimación costos	32
3.	Hitos de cada fase	34
4.	Fortalezas y debilidades del modelo de atención por zoom	40
5.	Requerimientos funcionales del sistema	47
6.	Requerimientos no funcionales del sistema	46
7.	Alcance del sistema	47
8.	Usuarios del sistema	48
9.	Condiciones de hardware: servidor	48
10.	Condiciones del hardware: terminal	49
11.	Condiciones de software del servidor (backend)	49
12.	Condiciones de software del usuario (frontend)	50
13.	Requisitos de desempeño	50
14.	Requerimiento de diseño e interfaz	51
15.	Análisis del entorno del sistema	56
16.	Descripción de casos de uso	58
17.	Diccionario de datos. colección: usuarios	63
18.	Diccionario de datos. colección: materias	64
19.	Diccionario de datos. colección: horario	64
20.	Diccionario de datos. colección: solicitudes	65
21.	Perfil de usuarios	66
22.	Descripción modular de mapa de navegación	67
23.	Descripción de servicios	68
24.	Componentes intrínsecos del sistema	79
25.	Software de implementación de componentes	81
26.	Plan de pruebas	91
27.	Pruebas diseño e interfaz	92
28.	Pruebas funcionales	94
29.	Pruebas de integridad y seguridad de los datos	94
30.	Casos de prueba del registro de usuarios	97
31.	Casos de prueba del inicio de sesión	98
32.	Casos de prueba del registro de horario	99
33.	Casos de prueba del registro de solicitudes	100
34.	Casos de prueba de la gestión de solicitudes del admin	101
35.	Casos de prueba de la gestión de solicitudes del root	102
36.	Casos de prueba del CRUD de usuarios	103
37.	Casos de prueba de las estadísticas	104
38.	Costo de hardware: proyecto	106
39.	Costo de software: proyecto	107
40.	Costo de talento humano: proyecto	107
41.	Costo de imprevistos: proyecto	108
42.	Costo total: proyecto	108
43.	Estudio de mercado	109
44.	Tiempo de atención zoom vs sam	111

Glosario

- Ticket: Medio estructurado para recopilar, organizar y gestionar un registro único que representa una solicitud de ajuste de matrícula planteado por un usuario, en este caso, un estudiante.

Resumen

Título: Sistema de información para la gestión administrativa del proceso de ajuste de matrícula en la Escuela de Ingeniería de Sistemas e Informática de la Universidad Industrial de Santander.

Autores: Esteban David Florez Tolosa , Dilan Alessandro Corredor Diaz, Carlos Daniel Sanjuan Argote

Palabras clave: Ajuste de matrícula, Tiempos de espera, Sistema de información, Gestión de solicitudes

Descripción:

El presente trabajo se ubica dentro del contexto de un proyecto de ingeniería en el entorno de una aplicación web, cuyo objetivo es desarrollar un sistema de información para la gestión del proceso de ajuste de matrícula en la Escuela de Ingeniería de Sistemas e Informática de la Universidad Industrial de Santander (UIS). El cual busca mejorar la gestión de solicitudes, reduciendo los tiempos de espera y mejorando la experiencia de los usuarios finales

Para el desarrollo del proyecto, se aplicó la metodología RUP (Rational Unified Process) con el fin de identificar las deficiencias del sistema actual, establecer sus limitaciones y definir los requerimientos funcionales y no funcionales necesarios para su implementación.

Los resultados obtenidos demuestran una disminución significativa en los tiempos de respuesta, logrando que el 78.84% de las solicitudes fueran atendidas en solo dos de los tres días destinados al proceso, representando una mejora del 73% con respecto al semestre anterior. Adicionalmente, la aplicación ha sido diseñada con una arquitectura flexible que permite adaptabilidad de acuerdo al pènsum de cada escuela y, con el propósito de mejorar y centralizar la gestión del ajuste de matrícula. Su implementación ha demostrado ser una alternativa eficiente para agilizar el proceso y mejorar la experiencia de los usuarios finales

* Trabajo de grado

*Facultad de Ingenierías Fisicomecánicas. Escuela de Ingeniería de Sistemas. Director Juan Pernaleté. Coodirector Luis Carlos Gomez

Abstract

Title: Information system for the administrative management of the enrollment adjustment process in the School of Systems and Computer Engineering of the Universidad Industrial de Santander.

Authors: Esteban David Florez Tolosa , Dilan Alessandro Corredor Diaz, Carlos Daniel Sanjuan Argote

Keywords: Enrollment adjustment, Waiting times, Information system, Request management

Description:

The present work is located within the context of an engineering project in the environment of a web application, whose objective is to develop an information system for the management of the enrollment adjustment process in the School of Systems and Computer Engineering of the Industrial University of Santander (UIS). It seeks to improve the management of applications, reducing waiting times and improving the experience of end users

For the development of the project, the RUP (Rational Unified Process) methodology was applied in order to identify the deficiencies of the current system, establish its limitations and define the functional and non-functional requirements necessary for its implementation.

The results obtained show a significant reduction in response times, with 78.84% of the requests being handled in only two of the three days allocated to the process, representing an improvement of 73% with respect to the previous semester. Additionally, the application has been designed with a flexible architecture that allows adaptability according to the curriculum of each school and, with the purpose of improving and centralizing the management of enrollment adjustment. Its implementation has proven to be an efficient alternative to streamline the process and improve the end-user experience.

* Bachelor Thesis

*Faculty of Physical and Mechanical Engineering. School of Systems Engineering. Director Juan Pernaleté. Co-director Luis Carlos Gomez

Introducción

La matrícula académica ordinaria es un proceso fundamental en las instituciones de educación superior, a través del cual los estudiantes seleccionan las asignaturas que desean cursar durante un semestre regular. Sin embargo, la asignación final de estas materias depende de la disponibilidad de cupos y los criterios definidos por la universidad. Como resultado, los estudiantes pueden recibir una matrícula en la que algunas de sus solicitudes no sean aprobadas, se les asignen grupos u horarios distintos a los que prefieren, en algunos casos no se les matrícula asignaturas previstas dentro del periodo académico a cursar.

Para corregir estas situaciones y permitir que los estudiantes ajusten su carga académica según sus necesidades, existe el proceso de ajuste de matrícula. Este procedimiento presente en la UIS y en algunas otras instituciones de educación superior, les permite reorganizar sus horarios académicos para cumplir con los requisitos de sus planes de estudio. Durante este proceso, los estudiantes pueden agregar o retirar asignaturas, cambiar de grupo o realizar modificaciones en sus horarios con el objetivo de evitar conflictos de horario y optimizar su carga académica.

En la Escuela de Ingeniería de Sistemas e Informática de la Universidad Industrial de Santander (UIS), este procedimiento se lleva a cabo a través de reuniones síncronas en la plataforma Zoom, una herramienta no diseñada específicamente para la gestión de solicitudes administrativas. Este enfoque presenta deficiencias significativas, como largos tiempos de espera, falta de visualización integral de horarios y una sobrecarga de trabajo para el personal administrativo, especialmente durante los periodos críticos del ciclo académico.

El proceso de ajuste de matrícula es una actividad esencial en instituciones de educación superior, ya que permite a los estudiantes reorganizar sus horarios académicos de acuerdo con sus necesidades y cumplir con los requisitos establecidos en sus planes de estudio. En la

Escuela de Ingeniería de Sistemas e Informática de la Universidad Industrial de Santander (UIS), este procedimiento se lleva a cabo a través de reuniones síncronas en la plataforma Zoom, una herramienta no diseñada específicamente para la gestión de solicitudes administrativas. Este enfoque presenta deficiencias significativas, como largos tiempos de espera, falta de visualización integral de horarios y una sobrecarga de trabajo para el personal administrativo, especialmente durante los períodos críticos del ciclo académico.

A nivel internacional, diversas universidades han implementado sistemas de información para gestionar procesos de matrícula y ajuste académico, como el Sistema de Información Académica (SIA) en la Universidad Nacional de Colombia (UNAL) y el Sistema de Información Universitario (SIU) de la Universidad Nacional del Comahue (UNCo). Estos sistemas han demostrado ser efectivos en la centralización de información y la reducción de tiempos de respuesta. Sin embargo, presentan limitaciones en áreas como la presentación de información actualizada, interfaces poco intuitivas y una documentación insuficiente, lo que afecta la experiencia del usuario[1][2].

En este contexto, el presente proyecto propone el desarrollo de un sistema de información en un entorno web que permita mejorar el proceso de ajuste de matrícula en la UIS. Este sistema se diferencia al integrar funcionalidades avanzadas, como la visualización de horarios integrados, el envío de notificaciones automáticas y la priorización de solicitudes. Además, busca reducir los tiempos de espera y la carga administrativa, mientras mejora la experiencia tanto para estudiantes como para el personal encargado. Esta propuesta le brindará a la Escuela de Ingeniería de Sistemas e Informática una herramienta que centralice la captura de información y establezca un precedente para futuras mejoras en la gestión administrativa de la institución. Con esta iniciativa, se garantizará la recopilación de datos de manera integral,

segura y útil para los procesos académicos de la escuela, lo que contribuirá a un manejo más efectivo de la información disponible.

Las fuentes y técnicas de recopilación de información utilizadas para el desarrollo del proyecto incluyeron la observación directa, la revisión bibliográfica y la entrevista no estructurada. Estas técnicas junto a sus instrumentos permitieron recopilar información suficiente para identificar las debilidades del proceso actual y, a partir de ello, definir los requisitos funcionales para la implementación del proyecto.

Una vez el proyecto entró en fase de producción, se llevó a cabo un proceso de retroalimentación mediante encuestas aplicadas a los estudiantes. Esto permitió evaluar el grado de cumplimiento de los objetivos iniciales y verificar que el sistema implementa soluciones efectivas a la problemática identificada en la etapa de análisis. Durante el desarrollo del sistema de información, se modelaron tanto los datos empleados por el sistema como los procesos que actúan sobre ellos. Los diagramas de casos de uso y el diseño de las vistas de los usuarios fueron las técnicas que permitieron trabajar en diferentes niveles de abstracción, detallando procesos concretos.

Este trabajo se enmarca dentro de la metodología Rational Unified Process (RUP), siguiendo sus directrices para el desarrollo de software. RUP establece un enfoque estructurado basado en cuatro fases fundamentales, las cuales se organizan de la siguiente manera:

Fase de inicio.

Se describen los elementos correspondientes a la Fase I (Inicio) de la metodología RUP, enfocándose en el análisis de la situación actual del modelo de ajuste de matrícula. En esta fase, se examinan los procesos vigentes, identificando sus principales deficiencias, tales como

los tiempos de espera prolongados y la falta de validaciones previas a los cambios solicitados por los estudiantes.

Asimismo, se analiza la estructura organizacional y las necesidades del sistema mediante reuniones con el personal administrativo, con el propósito de recopilar los requisitos iniciales. Este análisis permite definir el alcance del proyecto, su viabilidad y sus limitaciones

Fase de elaboración.

En él se describen las actividades realizadas para la captura de los requerimientos funcionales y no funcionales, acumulando información suficiente para el plan de construcción mitigando los riesgos técnicos clave creando un modelo detallado de requisitos, asegurando una planificación efectiva para la implementación del sistema.

Fase de construcción.

En esta fase, se presentan los detalles de las actividades que garantizan la capacidad operativa del sistema. Se lleva a cabo el desarrollo e implementación conforme a los requisitos y la arquitectura establecida, utilizando ciclos iterativos para la creación y validación de funcionalidades.

Se priorizan las funcionalidades esenciales, como la creación y seguimiento de solicitudes, asegurando que el sistema cumpla con los objetivos planteados. Además, se realizan pruebas continuas en cada iteración para garantizar la calidad del software y detectar posibles mejoras.

La integración de módulos se lleva a cabo de forma progresiva, incorporando componentes clave como el sistema de notificaciones, la visualización de horarios y la validación de conflictos de horario, lo que permite una transición fluida y un sistema robusto y funcional.

Fase de transición.

Preparar el sistema para su despliegue en un entorno real para el uso de la comunidad educativa para el proceso de ajuste de matrícula, adicionalmente se incluyen pruebas, despliegue, capacitación y ajustes según el feedback recopilado de los usuarios tras la implementación, con el fin de mejorar la experiencia y funcionalidad del sistema, adicionalmente se realiza la documentación que soporta al producto de cara al usuario como el video tutorial

Finalmente, se describen las conclusiones y trabajo futuro, de acuerdo al versionamiento del sistema, apéndices, bibliografía, y toda la documentación que apoya este proyecto.

1. Planteamiento y justificación del problema

La Escuela de Ingeniería de Sistemas e Informática de la Universidad Industrial de Santander (UIS), siguiendo las directrices institucionales para la gestión de sus procedimientos administrativos y académicos propios del proceso de matrícula, el cual, se encuentra alineado con las normativas de la universidad en el reglamento académico estudiantil de pregrado capítulo 111 Artículo 43, requiere adaptaciones específicas que respondan a las particularidades y necesidades del estudiante durante el proceso de ajuste de matrícula. Actualmente, el proceso se realiza por medio de una plataforma de videoconferencias como lo es Zoom, herramienta que, aunque útil para reuniones, no está diseñada para ser un canal de gestión de solicitudes. Su uso para este fin presenta deficiencias operativas que dificultan la gestión eficiente, afectando tanto a los estudiantes como al personal administrativo. El modelo de atención por Zoom depende de reuniones síncronas en las que cada estudiante es atendido individualmente, lo cual extiende los tiempos de espera. Además, los estudiantes al no contar con una herramienta que les permita visualizar los ajustes que pueden ser realizados a su matrícula, identificando de forma temprana posibles interferencias horarias,

disponibilidad de cupos entre otras, llevan al estudiante a no preparar de manera óptima las modificaciones a su matrícula, esto los obliga al momento de ser atendidos por Zoom, revisar una por una las opciones que mejor se ajusten a su horario. Esto contribuye a extender aún más los tiempos de espera de otros estudiantes, generando un excedente de trabajo administrativo para la dirección de escuela durante períodos críticos del ciclo académico. Hasta la fecha, esta metodología de trabajo ha resultado ser poco eficaz, ya que su implementación genera dispersión en el desarrollo de otras actividades administrativas que requieren un alto nivel de dedicación, lo que genera congestión en el flujo de trabajo, en procedimientos y en la disposición del personal a cargo de la dirección de escuela.

Conforme a lo expuesto se hace necesaria la implementación de un sistema de información para la gestión de ajustes de matrícula. Esta aplicación permitiría mejorar los tiempos de respuesta de cada solicitud, lo cual facilitará la organización y priorización de las solicitudes de acuerdo a parámetros específicos como lo establece la franja horaria de atención, mejorando la gestión de grandes volúmenes de peticiones de manera efectiva, llevando un registro claro de las solicitudes pendientes, aprobadas y rechazadas, permitiendo realizar un seguimiento de cada caso.

Además para los estudiantes el sistema de ajuste de matrícula ha de brindar una experiencia más fluida y ordenada, permitiéndoles realizar sus solicitudes de manera autónoma evaluando sus opciones, recibir notificaciones automáticas sobre el estado de sus solicitudes, aprobaciones, o requerimientos adicionales.

2. Objetivos

2.1 Objetivo general

Desarrollar un sistema de información en un entorno web que mejore el proceso de

ajuste de matrícula para los estudiantes de la Escuela de Ingeniería de Sistemas e Informática, reduciendo los tiempos de espera, la carga administrativa y minimizando conflictos en la gestión de solicitudes.

2.2 Objetivos específicos

- Analizar el proceso actual de ajuste de matrícula realizado por la Escuela de Ingeniería de Sistemas e Informática, identificando las deficiencias operativas y las necesidades específicas de estudiantes y personal administrativo.
- Determinar una solución que responda a las necesidades identificadas y se ajuste a las condiciones actuales de la Escuela de Ingeniería de Sistemas e Informática.
- Desarrollar un sistema de información web basado en la gestión de tickets, que permita gestionar y resolver solicitudes de ajuste de matrícula.

3. Marco de referencia

3.1 Marco teórico

El ajuste de matrícula es un proceso administrativo fundamental en las instituciones educativas, diseñado para que los estudiantes puedan reorganizar sus horarios académicos en función de sus necesidades, respetando las normativas institucionales. Sin embargo, la implementación efectiva de este proceso requiere el apoyo de herramientas tecnológicas avanzadas que aseguren una gestión ágil, organizada y eficiente.

- **Gestión administrativa en procesos académicos**

La gestión administrativa se refiere a la organización, planificación y control de las actividades operativas dentro de una institución, con el objetivo de garantizar la eficiencia en el cumplimiento de tareas específicas [3]. En el contexto del ajuste de matrícula, esto implica la recepción, priorización y resolución de solicitudes de los estudiantes.

- **Sistemas de información en la educación**

Un sistema de información es un conjunto de componentes interrelacionados que recopilan, procesan, almacenan y distribuyen información para apoyar la toma de decisiones y la coordinación de actividades dentro de una organización [4]. En las instituciones educativas, estos sistemas permiten mejorar procesos clave, como la gestión de matrículas y la organización de horarios, reduciendo errores humanos y tiempos de respuesta.

- **Visualización y notificaciones en tiempo real**

Un aspecto crucial en la interacción entre los usuarios y un sistema de información es la capacidad de visualizar datos de forma integral y en tiempo real [5]. Estas características no solo mejoran la experiencia del usuario, sino que también la toma de decisiones. En el caso del ajuste de matrícula, la integración de notificaciones automáticas proporciona retroalimentación inmediata al estudiante, aumentando la transparencia y reduciendo la incertidumbre durante el proceso.

El diseño e implementación de un sistema de información web para el ajuste de matrícula se basa en estos fundamentos teóricos, asegurando que los estudiantes puedan gestionar sus solicitudes de manera autónoma y eficiente, mientras el personal administrativo cuenta con herramientas para organizar y priorizar las solicitudes, cumpliendo con las normativas institucionales.

3.2 Marco conceptual

3.2.1 Sistema de información

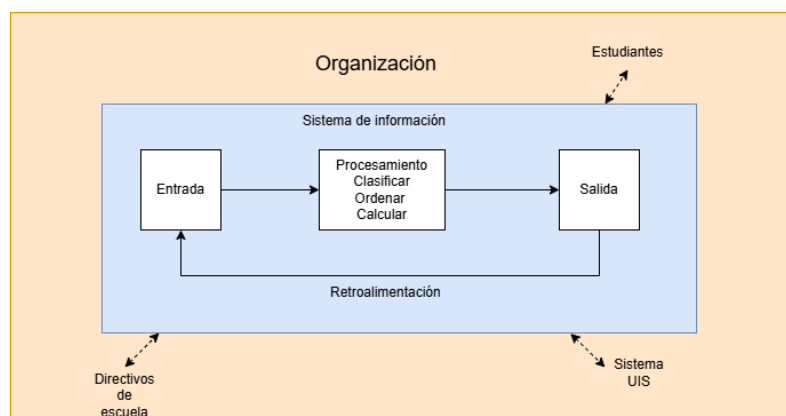
Los Sistemas de Información (SI) han adquirido un papel fundamental en la gestión organizacional, convirtiéndose en una herramienta estratégica para la optimización de

procesos administrativos. Un SI es un conjunto de componentes interdependientes diseñados para la recopilación, procesamiento, almacenamiento y distribución de información, facilitando la toma de decisiones y el control operativo dentro de una organización [6]. En la actualidad, la creciente dependencia de las organizaciones en el manejo eficiente de la información resalta la importancia de los SI como un recurso clave para mejorar la productividad, la competitividad y la eficiencia en la gestión empresarial.

Todo sistema de información se basa en el uso de datos como insumo fundamental, los cuales son capturados de su entorno, procesados y transformados con el propósito de generar información y un valor agregado para la organización. Esta información es posteriormente suministrada a los distintos usuarios del sistema, en un proceso que incluye una etapa de retroalimentación o *feedback*, mediante la cual se evalúa si la información obtenida cumple con las expectativas y requerimientos establecidos, los participantes del entorno como estudiantes, directivos de escuela y otros sistemas institucionales.(ver Figura 1).

Figura 1.

Funciones de un sistema de información



Un sistema de información tiene tres actividades básicas que producen la información que las organizaciones necesitan, entrada, proceso y salida, la entrada captura los datos necesarios de

la organización o de su entorno para procesarlos de una manera óptima y que se alinea a las necesidades de la empresa. La salida entrega la información procesada a las personas que la utilizarán.

3.3 Estado del arte

El uso de sistemas de información en la gestión académica ha evolucionado significativamente, con ejemplos en diferentes universidades que han automatizado procesos de matrícula y ajuste académico. A continuación, se presentan algunos antecedentes relevantes y su relación con el proyecto:

- **Plataforma SIA en la Universidad Nacional de Colombia**

El Sistema de Información Académica (SIA) es una herramienta diseñada para organizar procesos de matrícula y ajuste de horarios, centralizando la información académica de los estudiantes y reduciendo los tiempos de respuesta. No obstante, se han identificado áreas de mejora, como la necesidad de interfaces más intuitivas y una mejor integración con otros sistemas de información, para optimizar la experiencia del usuario y la eficiencia administrativa [1].

- **Sistema de Información Universitaria (SIU) de la Universidad Nacional del Comahue (UNCo)**

El Sistema de Información Universitaria (SIU) gestiona procesos académicos y administrativos, permitiendo a los estudiantes realizar inscripciones y consultar horarios. Sin embargo, estudios recientes han identificado limitaciones en su implementación, como la inconsistencia de datos, falta de capacitación y centralización limitada de la información. Estas deficiencias afectan tanto a estudiantes como al personal administrativo, evidenciando

la necesidad de soluciones integrales que mejoren la eficiencia y la toma de decisiones en la universidad [2].

Análisis de los antecedentes

Los ejemplos analizados muestran que, si bien existen avances significativos en la automatización de procesos académicos, persisten limitaciones importantes. La falta de integración entre las necesidades de los estudiantes y las capacidades administrativas genera obstáculos y experiencias poco óptimas para ambos grupos. Además, la falta de herramientas específicas para garantizar la accesibilidad a la documentación necesaria y la visualización integral de datos en los sistemas analizados refleja un vacío que este proyecto busca llenar, facilitando el acceso a información clave y mejorando la transparencia en los procesos académicos.

Resultados esperados

Este proyecto propone un sistema de información web diseñado específicamente para gestionar el ajuste de matrícula en la Escuela de Ingeniería de Sistemas e Informática de la Universidad Industrial de Santander. A diferencia de los trabajos previos, este sistema integrará funcionalidades que permitan:

- **Estudiantes:** Gestionar solicitudes de manera autónoma, visualizar horarios integrados y recibir notificaciones automáticas en tiempo real sobre el estado de sus solicitudes.
- **Director y coordinador académico:** Organizar, priorizar y dar seguimiento eficiente a cada solicitud, reduciendo la carga de trabajo y los tiempos de respuesta.

Se espera que este sistema transforme el proceso actual en un modelo más ágil, eficiente y transparente, beneficiando tanto a los estudiantes como al personal administrativo, y estableciendo un precedente para la mejora de procesos académicos en la UIS.

4. Metodología

4.1 Fase de inicio:

La fase de inicio en RUP no solo establece las bases técnicas del proyecto, sino que actúa como un puente entre la visión institucional y la ejecución operativa. En este contexto, la gestión del proyecto se convierte en el mecanismo para materializar los objetivos académicos en soluciones tecnológicas concretas, garantizando que el Sistema de Ajuste de Matrícula (SAM) no sea solo una herramienta funcional, sino un catalizador de transformación administrativa.

Esta fase prioriza tres ejes críticos:

1. **Precisión en la definición de necesidades:** Traducir problemáticas específicas (como la sobrecarga en Zoom) en requisitos técnicos medibles.
2. **Armonización institucional:** Alinear el desarrollo con políticas de la UIS, incluyendo seguridad de datos y cumplimiento del reglamento académico de pregrado en cuanto a los requisitos de matrícula (Cap III).
3. **Sostenibilidad del proceso:** Diseñar un sistema adaptable y flexible a futuras normativas o cambios en la demanda estudiantil.

La gestión del proyecto, detallada a continuación, operacionaliza estos principios mediante planes estructurados, asegurando que cada iteración en RUP contribuya no solo al éxito técnico del Sistema de Ajuste de Matrícula (SAM), sino a la modernización del ecosistema académico de la Escuela de Ingeniería de Sistemas e Informática.

4.1.1 Gestión del proyecto.

4.1.1.1 Plan de desarrollo del sistema.

El Plan de Desarrollo proporciona un marco estructurado para el desarrollo del Sistema de Información Web para la Gestión de Ajustes de Matrícula (SAM) de la Escuela de Ingeniería de Sistemas e Informática de la UIS, bajo la metodología RUP (Rational Unified Process).

El propósito principal de el Plan de Desarrollo es:

- **Definir la hoja de ruta:** Estructurar las fases, actividades y entregables bajo la metodología RUP, asegurando coherencia entre los objetivos del proyecto y las necesidades de la Escuela de Ingeniería de Sistemas e Informática de la UIS.
- **Alinear expectativas:** Servir como herramienta de comunicación entre stakeholders (estudiantes, personal administrativo, dirección) para garantizar que todos comprendan el alcance, riesgos y beneficios del sistema.
- **Mitigar riesgos:** Identificar posibles obstáculos técnicos, operativos o institucionales, y establecer acciones preventivas.
- **Optimizar recursos:** Estimar costos, tiempos y personal requerido, maximizando la eficiencia en el uso de recursos institucionales.

Este plan no solo es un documento de gestión, sino una herramienta dinámica que evolucionará con el proyecto. Cada iteración en RUP permitirá refinarlo, incorporando lecciones aprendidas y ajustes validados por los stakeholders (estudiantes, personal administrativo, dirección). Con esta base, el Sistema de Ajuste de Matrícula se posiciona como una solución viable, alineada con las necesidades técnicas, operativas y normativas de la UIS.

4.1.1.1.1 Vista general del proyecto.

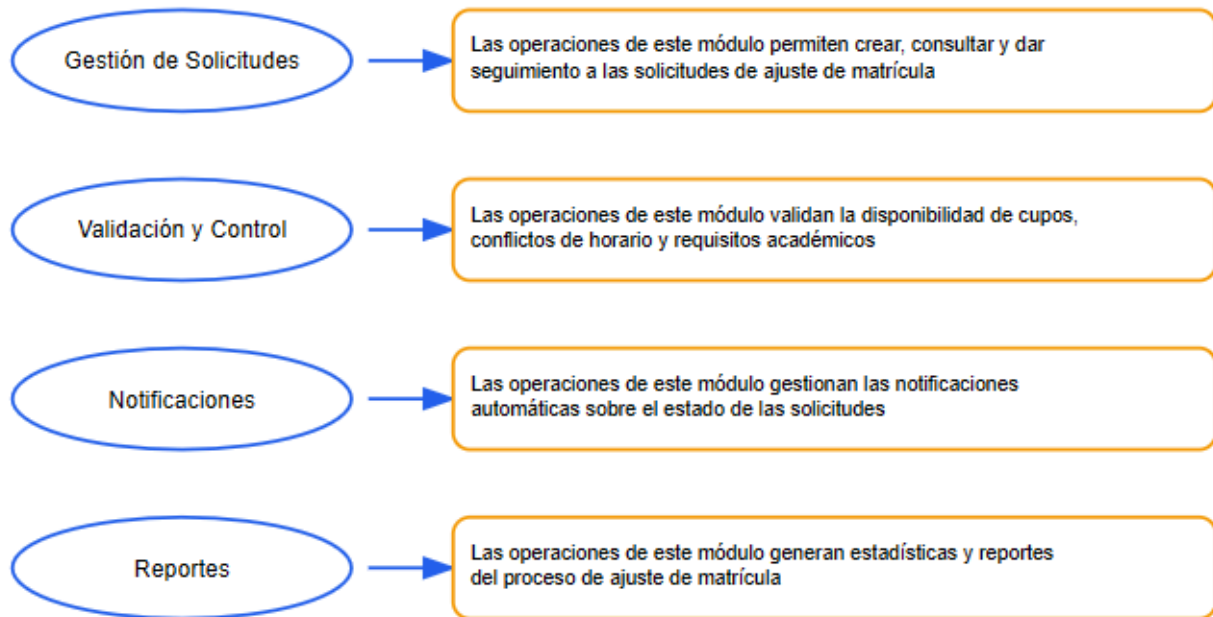
La información que a continuación se incluye ha sido extraída del análisis de los procesos institucionales de la Escuela de Ingeniería de Sistemas e Informática de la Universidad Industrial de Santander (EISI-UIS), con el objetivo de conocer la forma como se maneja el proceso de ajuste de matrícula.

Actualmente, el proceso de ajuste de matrícula se realiza de forma manual a través de reuniones en Zoom, lo que implica retardos en el proceso y en el tratamiento de la información, que se refleja en tiempos de espera prolongados, afectando tanto a los estudiantes como al personal administrativo. Por tal motivo, se hace necesaria la utilización de un sistema de información que aproveche la evolución tecnológica para mejorar este proceso.

La Escuela de Ingeniería de Sistemas e Informática demanda una forma más rápida, automática y segura para la realización de sus procesos internos. En tal sentido, el proyecto debe proporcionar una propuesta para el desarrollo de los subsistemas implicados, en mejorar el tiempo de respuesta, satisfacer las necesidades propias de los procesos y brindar información reutilizable por otras instancias institucionales. En este caso nos referimos a un Sistema de Información Web para Gestión de Ajustes de Matrícula, que debe estar conformado por los siguientes elementos.

Figura 2.

Visión del proyecto



Suposiciones y restricciones

Las suposiciones y restricciones respecto al sistema propuesto, y que se derivan directamente de la información recolectada son:

- El sistema maneja las solicitudes de ajuste de matrícula y servicios derivados como notificaciones y seguimiento de estado, pero no realiza funciones ajenas a este entorno como la matrícula inicial, planeación académica general o cambios directos en la matrícula del estudiante.
- El sistema procesa información fiable, pero esto no significa que no deba existir protección de información o control sobre el manejo del sistema por parte de los usuarios.
- El sistema estará implementado en un entorno web para proveer servicios como el seguimiento en tiempo real de solicitudes, fácil acceso para creación de solicitudes, visualización integrada de horarios académicos, notificaciones automáticas sobre el estado de las solicitudes y validación previa de conflictos horarios.

Esta implementación web garantizará el acceso continuo tanto para estudiantes como para el personal administrativo de la Escuela de Ingeniería de Sistemas e Informática, permitiendo una gestión más eficiente del proceso de ajuste de matrícula y reduciendo los tiempos de espera que actualmente se presentan en las sesiones síncronas de Zoom. Además, el sistema facilitará la centralización de la información, manteniendo un registro histórico de las solicitudes y sus resoluciones, lo que contribuirá a la toma de decisiones administrativas y a la mejora continua del proceso académico.

Entregables del Proyecto

A continuación se indican y describen cada uno de los artefactos que serán generados y utilizados por el proyecto:

- **Modelo de casos de uso del negocio:** Es un modelo de las funciones del negocio vistas desde la perspectiva de los actores externos (estudiantes, administrativos, etc.). Permite situar al sistema en el contexto organizacional haciendo énfasis en los objetivos en este ámbito.
- **Modelado del negocio:** Pieza de información significativa que permite identificar el proceso realizado por la escuela durante el ajuste de matrícula. Se refiere a las funcionalidades definidas para cada rol dentro de la escuela.
- **Visión:** Este documento define la visión del producto desde la perspectiva del usuario final, especificando las necesidades y características del sistema de ajuste de matrícula.
- **Prototipos de interfaces de usuario:** Prototipos que permiten visualizar las interfaces del sistema, incluyendo: Panel de administración, sistema de notificaciones, visualización de horarios, estadísticas de las solicitudes, panel de creación de solicitudes, panel de edición de usuarios.

- **Modelo de análisis y diseño:** Establece la realización de los casos de uso en clases, considerando la arquitectura web del sistema.
- **Modelo de datos:** Describe la representación lógica de los datos para la base de datos relacional que soportará el sistema.
- **Modelo de implementación:** Colección de componentes y subsistemas necesarios para la implementación del sistema web.
- **Modelo de despliegue:** Muestra la configuración de nodos del sistema y su despliegue en el entorno web.
- **Casos de prueba:** Documentos que establecen las condiciones de ejecución, entradas y resultados esperados para las pruebas del sistema.
- **Material de apoyo al usuario:** Documentación y guías de uso del sistema, incluyendo:
 - **Sistema:** Los ficheros del producto empaquetados y almacenados con los mecanismos apropiados para su instalación, configuración y despliegue en el entorno web de la UIS.
 - **Manual de usuario:** Videotutoriales guiados para el uso de estudiantes, coordinadores y directores de escuela.

Evolución del plan de desarrollo del sistema

El Plan de Desarrollo del Software se revisó semanalmente y se refinó antes del comienzo de cada iteración, siguiendo la metodología RUP con sus cuatro fases:

1. **Fase de inicio:** Análisis del proceso actual y definición de requisitos iniciales
2. **Fase de elaboración:** Diseño de la arquitectura y planificación detallada
3. **Fase de construcción:** Desarrollo e implementación del sistema
4. **Fase de transición:** Despliegue y capacitación de usuarios

Cada fase incluía revisiones y ajustes según la retroalimentación recibida, asegurando que el sistema cumpliera con los objetivos de mejorar el proceso de ajuste de matrícula en la Escuela de Ingeniería de Sistemas e Informática de la UIS.

4.1.1.1.2 Organización del proyecto.

Participantes en el proyecto

El equipo de desarrollo del Sistema de Información para la Gestión Administrativa del Proceso de Ajuste de Matrícula está conformado por distintos actores que desempeñan funciones clave en el diseño, implementación y validación del sistema.

- **Autor:** Este proyecto está concebido como el desarrollo de un trabajo de grado bajo la modalidad de práctica empresarial para optar al título de Ingeniero de Sistemas, por tal motivo el autor ha de cubrir todos los requisitos académicos y de conocimiento para el abordaje del proyecto y cumplir el perfil del puesto de acción.
- **Director del proyecto y tutor académico:** Deberán poseer el título de pregrado en la especialidad de la carrera a optar por el alumno, y tener como mínimo una ubicación administrativa equivalente a profesor Instructor otorgada por la Universidad Industrial de Santander.
- **Director de escuela y coordinador académico:** Son los usuarios asignados que laboran dentro de la Escuela de Ingeniería de Sistemas e Informática. Su función es evaluar el sistema, realizar observaciones y definir los requerimientos necesarios para garantizar que cumpla con su propósito.
- **Estudiantes:** Son todos los estudiantes de la Escuela de Ingeniería de Sistemas e Informática y a quienes está dirigido el sistema, realizan solicitudes de modificación de horario por conceptos de inclusión, cancelación y cambio de grupos de asignaturas.

Roles y responsabilidades

A continuación se describen las principales responsabilidades de cada uno de los roles en el equipo de desarrollo, de acuerdo al siguiente modelo de la Figura N° 3.

Figura 3.

Equipo de soporte

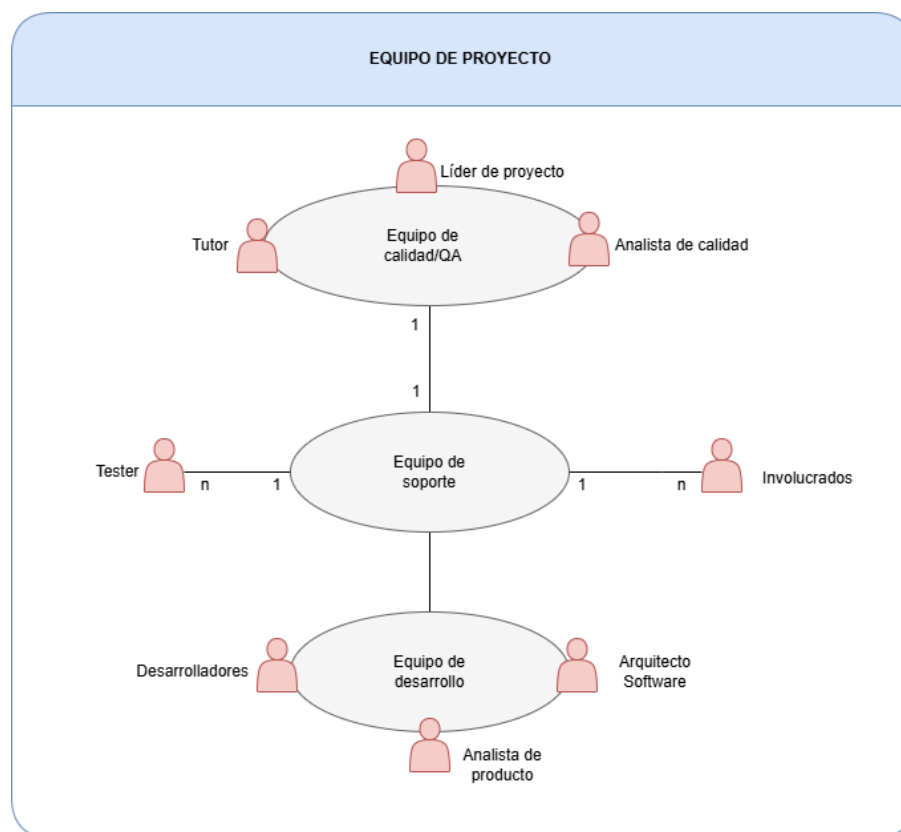


Tabla 1.*Roles y funciones dentro del equipo de trabajo*

<i>Puesto de acción</i>	<i>Roles</i>	<i>Responsabilidades</i>
<i>Tutor académico</i>	<i>Líder del proyecto</i>	<i>Dirigir y coordinar las interacciones entre los puestos de acción. Establecer las prácticas que aseguren la integridad y calidad de los artefactos del proyecto.</i>
	<i>Analista de calidad</i>	<i>Revisar todos los documentos que reflejan el avance del proyecto. Verificar que los objetivos del marco de desarrollo se cumplan.</i>
	<i>Mentor</i>	<i>Acompaña y apoya a los equipos de trabajo mediante revisiones de los artefactos. Hace recomendaciones de mejora durante todo el ciclo de vida del sistema.</i>
<i>Autores</i>	<i>Analista de producto</i>	<i>Dirigir el proceso de captura de requerimientos. Definir los actores y casos de uso y estructurar el modelo de casos de uso</i>
	<i>Arquitecto de Software</i>	<i>Establecer la arquitectura del desarrollo y refinarla en cada iteración. Construir prototipos necesarios para probar aspectos técnicos. Definir los lineamientos generales del diseño y la implementación.</i>
	<i>Desarrollador</i>	<i>Codificar los componentes en lenguaje de alto nivel. Elaborar y ejecutar pruebas unitarias.</i>

		<i>Documentar y actualizar clases bajo control de configuración.</i>
<i>Sección de Personal Docente</i>	<i>Tester</i>	<i>Realizar las pruebas identificadas y definidas previamente. Usar las instrucciones, métodos y herramientas ,necesarias</i>
<i>Estudiantes</i>	<i>Usuario final</i>	<i>Asegurar que sus necesidades sean satisfechas por el proyecto. Proporcionar retroalimentación sobre la funcionalidad del sistema</i>

Estimaciones de Costos del Proyecto

Según lo especificado por la Unidad de Apoyo a Gestión de Proyectos, la Vicerrectoría de Investigación y Extensión el presupuesto se establece considerando las modalidades actuales de trabajo y los recursos disponibles.[8]

Los costos asociados a los programadores fueron obtenidos de la plataforma Glassdoor 2024 [9], la cual provee estimaciones salariales basadas en datos reportados por empleados y empleadores. Esta información permitió establecer un rango referencial para el análisis de viabilidad económica del proyecto. A continuación se muestran los recursos y costos directos e indirectos que se han estimado:

Tabla 2.

Estimación costos

<i>Concepto</i>	<i>Responsable</i>	<i>Unidad</i>	<i>Cantidad</i>	<i>Valor unitario</i>	<i>Valor total</i>
<i>Recursos humanos</i>					

<i>Director del proyecto</i>	<i>Director</i>	<i>Hora</i>	<i>34</i>	<i>\$ 305.000/hora</i>	<i>\$ 10.370.000</i>
<i>Autor 1</i>	<i>Estudiante</i>	<i>Hora</i>	<i>170</i>	<i>\$ 16.000/hora</i>	<i>\$ 2.720.000</i>
<i>Autor 2</i>	<i>Estudiante</i>	<i>Hora</i>	<i>170</i>	<i>\$ 16.000/hora</i>	<i>\$ 2.720.000</i>
<i>Autor 3</i>	<i>Estudiante</i>	<i>Hora</i>	<i>170</i>	<i>\$ 16.000/hora</i>	<i>\$ 2.720.000</i>
<i>Sub total</i>					<i>\$ 18.530.000</i>

Recursos tecnológicos

<i>Alojamiento en la nube</i>	<i>Estudiante</i>	<i>Mes</i>	<i>4</i>	<i>\$ 66.250/més</i>	<i>\$ 265.000</i>
<i>Internet</i>	<i>Estudiante</i>	<i>Mes</i>	<i>4</i>	<i>\$ 70.000</i>	<i>\$ 280.000</i>
<i>Computador personal</i>	<i>Estudiante</i>	<i>Unidad</i>	<i>3</i>	<i>\$ 3.500.000</i>	<i>\$ 10.500.000</i>
<i>Sub total</i>					<i>\$ 11.045.000</i>

Gastos adicionales

<i>Insumos</i>	<i>Estudiante</i>	<i>Imprevistos</i>			<i>\$130.000</i>
----------------	-------------------	--------------------	--	--	------------------

Presupuesto general

Concepto	Valor
<i>Recursos humanos</i>	<i>\$ 18.530.000</i>
<i>Recursos tecnológicos</i>	<i>\$ 11.045.000</i>
<i>Gastos adicionales</i>	<i>\$130.000</i>
<i>Total</i>	<i>\$ 29.705.000</i>

Plan del proyecto

En esta sección se presenta la organización en fases e iteraciones y el calendario del proyecto.

Plan de las fases

El desarrollo se llevará a cabo en base a fases con una o más iteraciones en cada una de ellas.

La siguiente gráfica muestra la distribución de tiempos y el número de iteraciones de cada fase.

Figura 4.

Distribución de iteraciones para cada fase



Los hitos que marcan el final de cada fase se describen en la siguiente tabla:

Tabla 3.

Hitos de cada fase

Descripción	Hito
<i>Fase de inicio</i>	<i>La aceptación del director de escuela del plan de desarrollo marcan el final de esta fase.</i>
<i>Fase de elaboración</i>	<i>La validación y aceptación del diseño lógico, físico y estructural del sistema por los administradores y directivos. Además, la revisión y entrega de todos los elementos desarrollados hasta este punto son requisitos para concluir esta fase.</i>
<i>Fase de construcción</i>	<i>La finalización de esta fase se define con la disponibilidad de una versión operativa parcial del sistema, preparada para pruebas controladas por los administradores y el director de escuela</i>
<i>Fase de transición</i>	<i>La implementación del sistema en el entorno institucional, acompañada de la entrega de documentación técnica y manuales de usuario. Además, se incluyen capacitaciones al personal administrativo y la recopilación de encuestas de satisfacción para evaluar la adopción del sistema.</i>

Calendario del Proyecto

El calendario del proyecto establece las principales tareas a desarrollar dentro del enfoque iterativo e incremental de la metodología RUP. Esta metodología se caracteriza por la ejecución en paralelo de todas las disciplinas de desarrollo a lo largo del ciclo de vida del

proyecto, permitiendo que la mayoría de los elementos sean creados en etapas tempranas y refinados progresivamente en cada iteración.

En este contexto, se ha definido un cronograma en el que cada elemento evoluciona hasta alcanzar un nivel de complejidad suficiente para ser sometido a revisión y aprobación. Este proceso de validación no ocurre de manera aislada, sino que se lleva a cabo de forma continua en distintas fases del desarrollo, garantizando una mejora incremental del sistema.

Análisis de riesgos

Se lleva a cabo un análisis detallado de los posibles riesgos que podrían afectar el desarrollo del proyecto, ya sea retrasando su ejecución o disminuyendo la probabilidad de éxito. La identificación temprana de estos riesgos permite establecer estrategias de mitigación adecuadas para minimizar su impacto.

Los principales riesgos identificados son:

- El cronograma del proyecto es ajustado, lo que puede generar dificultades para cumplir con los hitos establecidos, afectando la calidad del desarrollo y las pruebas del sistema.
- La información esencial sobre los estudiantes, necesaria para el ajuste de matrícula, es administrada por la Dirección de Admisiones y Registro Académico. La incertidumbre sobre su disponibilidad y los tiempos de respuesta prolongados pueden retrasar la integración de datos en el sistema.
- Resistencia al cambio por parte de los usuarios.
- La disponibilidad y actualización de la malla curricular y requisitos académicos es fundamental para la correcta asignación de cupos y la validación de prerrequisitos. La falta de acceso oportuno a estos datos podría comprometer la precisión del sistema.

- Difusión y conocimiento del nuevo sistema para que los estudiantes realicen el ajuste de matrícula por medio del sistema

Estrategias de mitigación del riesgo

El desarrollo de un plan de acción para abordar los riesgos potenciales del proyecto permite implementar estrategias de mitigación efectivas, garantizando la ejecución del proyecto sin afectar sus objetivos ni comprometer la planificación inicial.

- Para optimizar la gestión del tiempo y la organización de tareas, se propone la adopción de metodologías ágiles como Scrum, lo que permitirá estructurar el trabajo en iteraciones cortas y garantizar la entrega progresiva de funcionalidades operativas.
- Ante la dependencia de la Dirección de Admisiones y Registro Académico para la obtención de información estudiantil, se establece como alternativa la solicitud directa al estudiante de su franja horaria y carga académica a través de SAM, en caso de que la información no esté disponible.
- Para facilitar la adopción del sistema y mejorar la experiencia del usuario, se dispondrán canales de soporte y asistencia, complementados con un video tutorial detallado que guíe a los usuarios en el uso de las funcionalidades de la plataforma.
- Adicionalmente, se plantea la implementación de un web scraping automatizado que extraiga información de la página web de la Universidad, recopilando los datos de las asignaturas correspondientes a la malla curricular y exportarlos en formato JSON, garantizando así la disponibilidad y actualización eficiente de la información académica.

4.1.2 Modelado de la organización.

4.1.2.1 Definición del negocio.

La institución de referencia para el desarrollo del sistema es la Escuela de Ingeniería de Sistemas e Informática de la Universidad Industrial de Santander. Dentro de su estructura

organizativa, la Dirección de Escuela es la entidad responsable del proceso de ajuste de matrícula, el cual se rige por las disposiciones establecidas en el Reglamento Académico Estudiantil de Pregrado, específicamente en el Capítulo III.

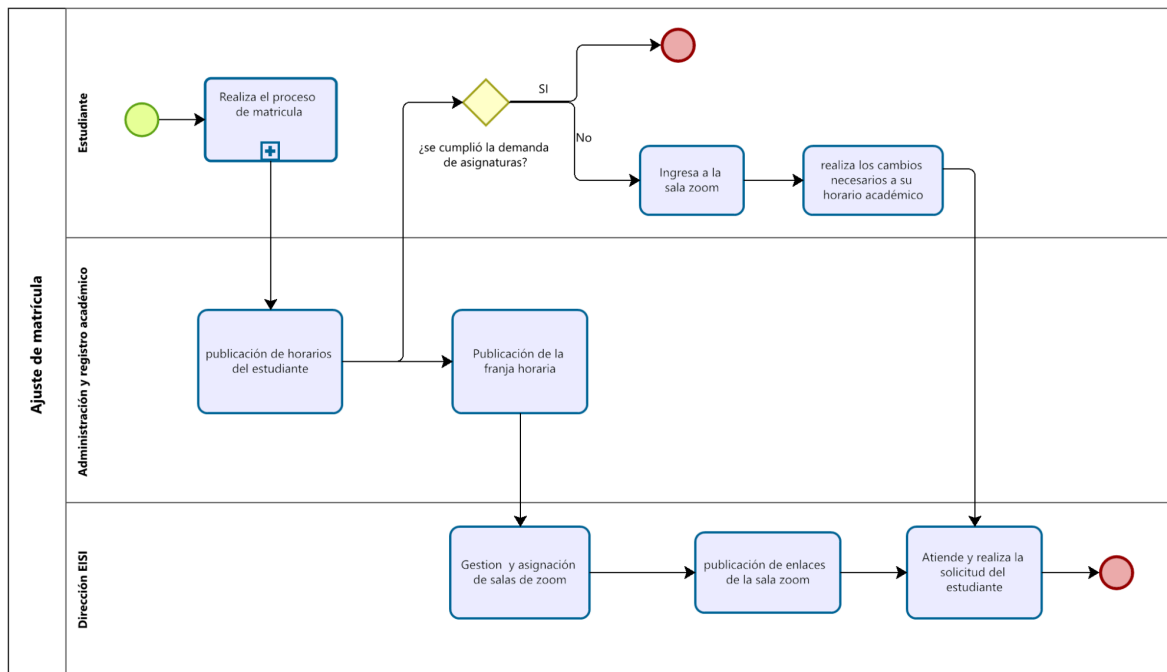
Este proceso permite a los estudiantes solicitar modificaciones en su horario académico cuando sus necesidades no han sido satisfechas en el proceso de matrícula inicial. El ajuste de matrícula se lleva a cabo en un período definido, posterior a la publicación de los horarios y las franjas horarias asignadas a cada estudiante, información que es gestionada y publicada por la Dirección de Admisiones y Registro Académico.

Una vez divulgada esta información, la Dirección de Escuela organiza la asignación de salas virtuales (Zoom), donde los estudiantes serán atendidos de acuerdo con la franja horaria previamente establecida.

A continuación, el siguiente diagrama Business Process Model and Notation (BPMN) representa de manera detallada la interacción entre los distintos actores y el flujo del proceso de ajuste de matrícula.

Figura 5.

Modelo BPMN ajuste de matrícula atención por zoom



4.1.2.2 Descripción del negocio actual.

La Dirección de Escuela tiene entre sus responsabilidades la captura, gestión y procesamiento de las solicitudes de ajuste de matrícula realizadas por los estudiantes vía zoom, información que deben capturar y llevar a través del Sistema de Información de la Universidad Industrial de Santander en el cual el director debe ingresar la solicitud realizada por el estudiante, posteriormente el sistema evalúa cada solicitud mediante una serie de validaciones automáticas, determinando si la solicitud es aprobada o rechazada con base en criterios como conflictos de horario entre asignaturas, disponibilidad de cupos y requisitos académicos.

Actualmente, la Dirección de Escuela cuenta con un equipo compuesto por cuatro personas, distribuidas de la siguiente manera: Director de Escuela, Secretaría, Coordinador y Técnico de Soporte Académico. Sin embargo, debido a la alta demanda de estudiantes que requieren atención durante el proceso de ajuste de matrícula, es necesario reforzar el equipo con dos personas adicionales del área de personal docente, con el fin de agilizar la gestión y disminuir los tiempos de respuesta.

A continuación, se describen los roles existentes dentro de la Dirección de Escuela y sus respectivas funciones en el proceso de ajuste de matrícula.

Director de Escuela

- Supervisa y coordina el proceso de ajuste de matrícula, asegurando su correcta ejecución según las normativas institucionales.
- Toma decisiones estratégicas para la optimización del proceso y la asignación de recursos.
- Valida y autoriza ajustes especiales que requieran intervención directa, como ampliación de cupos o creación de nuevos grupos.

Secretaría

- Coordina la programación de las salas de atención para los ajustes de matrícula.
- Realiza la publicación de enlaces de las salas zoom en los medios de comunicación oficiales de la escuela.

Coordinador Académico

- Evalúa la viabilidad de las solicitudes de ajuste de matrícula en función de los criterios del sistema (conflictos de horario, cupos disponibles y requisitos académicos).
- Se encarga de la asignación y reubicación de estudiantes en las diferentes asignaturas.
- Apoya la revisión y validación de las solicitudes procesadas en el sistema.
- Actúa como enlace entre la Dirección de Escuela y los docentes para resolver incidencias en la asignación de horarios y cupos.

Personal Docente

- Refuerzan la atención a los estudiantes en la revisión y validación de sus solicitudes.

- Contribuyen a la gestión eficiente del ajuste de matrícula, reduciendo los tiempos de espera.

4.1.2.3 Evaluación del estatus del negocio.

Esta sección tiene como objetivo evaluar y analizar el estado actual del proceso, así como definir los objetivos del modelado correspondiente. Actualmente, la herramienta asignada por la Universidad Industrial de Santander para la gestión de solicitudes no está optimizada para este tipo de procesos. Sin embargo, el crecimiento en el número de estudiantes matriculados en la escuela ha incrementado significativamente la demanda de cupos y cambios entre asignaturas, lo que ha derivado en un aumento considerable de los trámites administrativos asociados. En este contexto, el uso de Zoom se ha vuelto insostenible debido al elevado volumen de solicitudes y estudiantes que requiere gestionar en la actualidad.

Los retrasos en el tiempo de respuesta a las solicitudes estudiantiles constituyen el principal factor de perturbación en el proceso. Esto se debe a la dificultad de realizar un preprocesamiento eficiente que permita validar previamente la disponibilidad de cupos, los requisitos entre asignaturas y los posibles conflictos de horario. La ausencia de este filtrado previo ocasiona que el personal de la escuela invierta tiempo en procesar solicitudes que por restricciones académicas o administrativas, no pueden ser atendidas, desviando recursos de aquellas que sí podrían ser gestionadas eficazmente. a continuación se resumen las fortalezas y debilidades del proceso actual.

Tabla 4.

Fortalezas y debilidades del modelo de atención por zoom

<i>Debilidades</i>	<i>Causas</i>	<i>Sugerencias</i>
--------------------	---------------	--------------------

<i>La herramienta por la cual se realiza la gestión de solicitudes no está optimizada para este tipo de procesos</i>	<i>La plataforma asignada por la universidad no se diseñó para apoyar este tipo de procesos</i>	<i>Implementar un sistema de información especializado para la gestión de solicitudes académicas.</i>
<i>No hay un criterio que defina la prioridad para atender solicitudes</i>	<i>La herramienta no cuenta con una funcionalidad que permita organizar solicitudes</i>	
<i>.El uso de Zoom como medio de gestión de solicitudes es insostenible</i>	<i>Alto volumen de solicitudes y estudiantes</i>	<i>Desarrollar un sistema de información con la capacidad para procesar alto volumen de peticiones</i>
<i>Desvío de recursos hacia solicitudes que no pueden ser procesadas .</i>	<i>Falta de filtrado previo que descarte solicitudes que no cumplen requisitos académicos o administrativos.</i>	<i>Falta de filtrado previo que descarte solicitudes que no cumplen requisitos académicos o administrativos.</i>
<i>Los estudiantes deben permanecer atentos a la llamada de Zoom hasta ser atendidos</i>	<i>Zoom no puede notificará al estudiante cuando será atendido</i>	<i>Desarrollar una herramienta que capture la información y esta sea procesada sin la necesidad de tener al estudiante en línea</i>

Fortaleza	Causas	Sugerencias
<i>Atención personalizada</i>	<i>La plataforma por ser una herramienta de videoconferencias</i>	<i>Desarrollar una funcionalidad que permita observaciones adicionales en la solicitud para el estudiante y para el directivo</i>
<i>Retroalimentación al estudiante al instante</i>	<i>Interacción directa entre el estudiante y el directivo</i>	<i>Incorporar el envío de correo electrónico notificando al estudiante el estado de sus solicitudes y un módulo para ver el estado de sus peticiones junto con el motivo de rechazo</i>

Como solución a esta problemática, se propone el desarrollo de un Sistema de Información web para mejorar el proceso de ajuste de matrícula. Este sistema estará alineado con los procedimientos administrativos actuales y permitirá registrar las solicitudes estudiantiles, incorporando un mecanismo de preprocesamiento que valide automáticamente los requisitos entre asignaturas y los posibles conflictos de horario. De esta manera, se garantizará que solo las solicitudes viables sean gestionadas por el personal directivo, reduciendo la carga administrativa y mejorando los tiempos de respuesta.

4.2 Fase de elaboración:

La fase de Elaboración para SAM representa un punto de transformación crítico donde se transforma la visión conceptual en una arquitectura tecnológica sólida y viable. Este proceso va más allá de un simple diseño técnico, constituyéndose como un puente estratégico entre los requerimientos académicos de la Escuela de Ingeniería de Sistemas e Informática de la UIS y una solución informática que responda integralmente a sus necesidades.

El objetivo central es construir una base arquitectónica que no solo resuelva la problemática actual del proceso de ajuste de matrícula, sino que también anticipe futuras evoluciones administrativas y tecnológicas. Esto implica un análisis multidimensional que contemple:

- Evaluación exhaustiva de las restricciones operativas actuales
- Modelado de escenarios de uso proyectados
- Definición de una infraestructura flexible y escalable
- Integración de mecanismos de seguridad y trazabilidad
- Preparación para posibles ampliaciones o modificaciones normativas

Al concluir la fase de Elaboración, se contará con un documento que incluirá un modelo de sistema web completamente especificado para ajuste de matrícula, con definición precisa de arquitectura técnica, modelado de casos de uso críticos, prototipo de componentes funcionales, plan de gestión de riesgos, especificaciones técnicas detalladas, modelo de datos preliminar y una hoja de ruta clara para la fase de construcción, todo alineado con los requerimientos académicos y administrativos de la Escuela de Ingeniería de Sistemas e Informática de la UIS.

4.2.1 Requerimientos.

El objetivo principal de esta disciplina es establecer las funciones que se quiere que satisfaga el sistema a construir. Los requerimientos pueden ser divididos en dos grupos, los requerimientos funcionales, los cuales describen las funciones que el software va a ejecutar;

y los requerimientos no funcionales, los cuales especifican criterios que pueden usarse para juzgar la operación de un sistema en lugar de sus funciones específicas, como parte de los requerimientos de facilidad de uso.

4.2.1.1 Análisis del problema.

El diagnóstico del proceso actual de ajuste de matrícula revela una problemática multifacética que trasciende la simple ineficiencia administrativa. La Escuela de Ingeniería de Sistemas e Informática de la Universidad Industrial de Santander (UIS) enfrenta un escenario donde la plataforma Zoom, originalmente concebida para videoconferencias, se ha convertido en el núcleo de un proceso académico crítico, generando fricciones operativas significativas.

La gestión actual se caracteriza por una serie de limitaciones estructurales. Los tiempos de espera se dilatan considerablemente para los estudiantes, producto de un modelo de atención secuencial que somete a estudiantes y administrativos a largas sesiones de videoconferencia. La visualización de horarios carece de integralidad, obligando a los estudiantes a realizar cálculos manuales complejos para identificar potenciales conflictos académicos de horario o requisitos.

La sobrecarga administrativa es otro elemento crítico. El personal de la escuela destina recursos humanos desproporcionados para gestionar y recopilar manualmente solicitudes que podrían automatizarse, restando eficiencia a otros procesos académicos estratégicos. La ausencia de trazabilidad implica que cada solicitud se convierte en un evento aislado, sin posibilidad de análisis sistémico o generación de conocimiento institucional.

Necesidad tecnológica transformacional.

La solución demanda un sistema web que no sea meramente un repositorio de solicitudes, sino un ecosistema tecnológico que transforme radicalmente la experiencia académica. Se requiere una plataforma capaz de gestionar integralmente el proceso de ajuste de matrícula, con capacidades que superen las limitaciones actuales.

La arquitectura tecnológica debe contemplar una infraestructura web moderna, fundamentada en principios de escalabilidad, seguridad e interoperabilidad. Una base de datos no relacional flexible y veloz actuará como columna vertebral, permitiendo no solo el almacenamiento, sino la generación de conocimiento a partir de los datos académicos.

4.2.1.2 Definición del sistema.

Objetivo estratégico

El sistema se configura como una herramienta de transformación digital para la gestión académica, cuyo propósito fundamental trasciende la automatización: busca redefinir la experiencia de ajuste de matrícula, equilibrando las necesidades de estudiantes y la administración académica.

Arquitectura conceptual

La plataforma se estructurará como un ecosistema de módulos interconectados, cada uno diseñado para resolver dimensiones específicas del proceso académico. El módulo de solicitudes permitirá a los estudiantes gestionar sus ajustes con autonomía y precisión. El sistema de validación académica incorporará reglas para garantizar el cumplimiento de requisitos curriculares y evitar los conflictos entre materias por el horario de manera instantánea.

Un componente diferenciador será el sistema de notificaciones por medio del correo institucional, que transformará la comunicación de ser reactiva a proactiva, manteniendo informados a los estudiantes en tiempo real sobre el estado de sus solicitudes.

Componentes de Valor

El panel administrativo no será simplemente una interfaz de gestión, sino un tablero de inteligencia operativa. Generará reportes estadísticos que permitirán comprender patrones de matrícula, optimizar la oferta académica y tomar decisiones estratégicas basadas en datos.

Beneficios Projectados

Más allá de la eficiencia operativa, el sistema se configura como un catalizador de transformación institucional. La reducción de tiempos de proceso, la mejora en la experiencia estudiantil y la optimización de recursos administrativos son resultados inmediatos. A largo plazo, se espera una reconfiguración de los flujos de trabajo académico, fundamentada en datos y centrada en la experiencia del usuario.

Conclusión estratégica

El Sistema de Ajuste de Matrícula representa más que una solución tecnológica: es una respuesta estratégica a las complejidades de la gestión académica contemporánea, diseñada para evolucionar junto con las necesidades de la institución.

4.2.1.2.1 Requerimientos funcionales.

Son los requerimientos que le proporcionan funcionalidad al sistema. Podemos observar una lista de estos requerimientos en la siguiente tabla:

Tabla 5.

Requerimientos funcionales del sistema

No.	Requerimientos funcionales del sistema
01	<i>El sistema debe permitir a los estudiantes registrar su horario matriculado, visualizando los horarios de las asignaturas integradas</i>
02	<i>El sistema debe permitir a los estudiantes realizar solicitudes de ajuste de matrícula, registrando distintas peticiones (inclusión, cancelación o cambio de grupo), visualizando los horarios de las asignaturas integradas</i>
03	<i>El sistema debe registrar estudiantes mediante autenticación con correo institucional y código único, enviando un enlace de verificación al correo electrónico institucional para activar la cuenta</i>
04	<i>El sistema debe permitir al Director y Coordinador académico priorizar solicitudes según criterios predefinidos (ej: turno asignado, hora de creación).</i>
05	<i>El sistema debe generar estadísticas para el personal administrativo (solicitudes pendientes, aprobadas, rechazadas, creadas).</i>
06	<i>El sistema debe notificar por correo electrónico institucional el estado de las solicitudes a los estudiantes (aprobación, rechazo, requerimientos adicionales)</i>
07	<i>El sistema debe permitir la recuperación de contraseña mediante un enlace enviado al correo institucional registrado.</i>

08	<i>El sistema debe permitir al root registrar administrativos y estudiantes.</i>
09	<i>El sistema debe permitir al administrativo enviar solicitudes que no pueda resolver al root</i>
10	<i>El sistema debe permitir a los estudiantes ver el historial de sus solicitudes, el estado de estas solicitudes y la aprobación o rechazo de cada una de las peticiones</i>
11	<i>El sistema debe permitir a los estudiantes crear asignaturas que no se encuentren integradas en el sistema para agregar en su horario o solicitud, también debe validar que no existan ya estas asignaturas creadas</i>
12	<i>El sistema debe registrar en logs dentro de las solicitudes cada uno de los usuarios que interceden en ella(estudiante, administrativos, root)</i>
13	<i>El sistema debe asignar al admin una solicitud a la vez y que no pueda resolver otra hasta terminar con la actual</i>

4.2.1.2.2 Requerimientos no funcionales.

Son requerimientos que imponen restricciones en la implementación, así como restricciones en el diseño.

Tabla 6.

Requerimientos no funcionales del sistema

No	Requerimientos no funcionales del sistema
01	<i>El sistema se desarrollará utilizando React.js para el frontend y Node.js con Express.js para el backend. La base de datos será MongoDB. Se utilizará Git para el control de versiones del código, con ramas separadas para desarrollo y producción</i>
02	<i>El sistema debe responder en máximo 3 segundos para operaciones críticas como el envío de solicitudes y el proceso de registro.</i>
03	<i>El sistema debe ser multiusuario, soportando al menos 100 solicitudes simultáneas sin degradación de rendimiento.</i>
04	<i>El sistema debe utilizar encriptación bcrypt para almacenamiento seguro de contraseñas, HTTPS para transmisión de datos y JWT para el manejo de sesiones con tokens de acceso que expiran cada hora.</i>
05	<i>El sistema debe mantener una disponibilidad del 99% durante los periodos críticos de inscripción de horarios, con un tiempo máximo de recuperación ante fallos de 30 minutos</i>
06	<i>La interfaz de usuario será responsive, funcionando correctamente en dispositivos móviles y de sobremesa, con soporte garantizado para las últimas versiones de Chrome, Firefox y Safari entre otros navegadores.</i>
07	<i>El tiempo de carga inicial no superará los 1.5 segundos y se implementará soporte para modo oscuro y claro. La interfaz seguirá los principios de diseño de Material design para mantener la consistencia visual.</i>
08	<i>El manual de usuario será mediante videotutoriales del uso del sistema</i>

4.2.1.2.3 Requerimientos complementarios.

- Disponer de equipos de cómputo con acceso a internet para la implementación y uso del sistema
- Cumplir con las normas de comunicación TCP/IP
- El personal administrativo debe recibir capacitación en el uso del sistema
- Se debe mantener una copia de seguridad actualizada del código fuente en repositorios Git

Se debe contar con un plan de contingencia en caso de fallos durante períodos críticos

Mantener documentación técnica actualizada del sistema para futuros desarrollos

4.2.1.4 Gestión y alcance del sistema.

A continuación se mostrará un listado con los beneficios que obtendrá el cliente a partir del producto:

Tabla 7.*Alcance del sistema*

<i>Alcance del sistema</i>	<i>Beneficio del cliente</i>	<i>Justificación</i>
<i>Captura de la información para procesos</i>	<i>Optimización del tiempo en la gestión de solicitudes</i>	<i>Facilita el acceso y reutilización de datos en futuras consultas y reportes.</i>
<i>Base de datos centralizada con la información de todas las solicitudes</i>	<i>Reducción de errores y mejora en la calidad de la información</i>	<i>Garantiza que todas las solicitudes cumplan un formato estandarizado, facilitando su procesamiento.</i>
<i>Validación automatizada de requisitos y disponibilidad.</i>	<i>Disminución de tiempos de respuesta y carga administrativa</i>	<i>Evita que el personal procese solicitudes inviables, optimizando recursos.</i>
<i>Generación de reportes y estadísticas.</i>	<i>Análisis de rendimiento en el procesamiento de las solicitudes.</i>	<i>Permite identificar patrones y realizar mejoras en la asignación de cupos</i>

<i>Interfaz web accesible desde cualquier dispositivo.</i>	<i>Facilidad de acceso y uso desde cualquier ubicación</i>	<i>El sistema es adaptable a distintos dispositivos, permitiendo su uso remoto.</i>
--	--	---

4.2.1.3 Usuarios del sistema.

Se refiere a los usuarios directos del sistema, su responsabilidad en el manejo y jerarquía en la gestión del sistema.

Tabla 8.

Usuarios del sistema

Nombre	Descripción	Responsabilidades	Jerarquía
<i>Root</i>	<i>Usuario con los privilegios más altos del sistema.</i>	<i>Procesamiento de solicitudes especiales (aumentar cupos, crear nuevos grupos)</i>	<i>Director de escuela máxima autoridad.</i>
<i>Administrador</i>	<i>Encargado de la gestión operativa.</i>	<i>Procesar solicitudes y redirigir al root en caso de ser requerido.</i>	<i>Coordinador Académico y Técnico de Soporte Académico.</i>
<i>Estudiantes</i>	<i>Usuarios que realizan solicitudes de ajuste de matrícula</i>	<i>Ingresar y registrar sus solicitudes de cupos y</i>	<i>Usuario final</i>

4.2.1.4 Requisitos del sistema.

4.2.1.4.1 Requisitos de hardware.

Un servidor WEB y de base de datos, la configuración de hardware recomendada para el servidor mencionado es la siguiente:

Servidor (Backend)

Tabla 9.

Condiciones de hardware: servidor

<i>Componente</i>	<i>Especificación</i>	<i>Justificación</i>
<i>Procesador.</i>	<i>8 vCPU (64 bits, 2.4 GHz o superior).</i>	<i>Soporta 5 instancias paralelas para manejar alta concurrencia de solicitudes.</i>
<i>Memoria RAM.</i>	<i>8 GB DDR4.</i>	<i>Optimiza el rendimiento en operaciones intensivas</i>
<i>Almacenamiento.</i>	<i>SSD de 100 GB.</i>	<i>Velocidad en lectura/escritura para bases de datos y archivos temporales.</i>

Tabla 10.

Condiciones del hardware: terminal

<i>Componente</i>	<i>Especificación</i>	<i>Justificación</i>
<i>Procesador.</i>	<i>Dual Core 1.6 GHz o superior.</i>	<i>Ejecución fluida de navegadores modernos (Chrome, Firefox).</i>
<i>Memoria RAM.</i>	<i>4 GB.</i>	<i>Multitarea básica (pestañas abiertas, aplicaciones en segundo plano).</i>
<i>Conexión a Internet</i>	<i>10 Mbps estable.</i>	<i>Acceso sin interrupciones al sistema web.</i>

4.2.1.4.2 Requisitos de software.

Estos requerimientos son los medios intangibles para el desarrollo y el óptimo funcionamiento del sistema, a continuación los detalles:

Tabla 11.

Condiciones de software del servidor (backend)

<i>Descripción</i>	<i>Justificación</i>
<i>Linux</i>	<i>Sistema operativo estable para servidores, compatible con entornos cloud.</i>
<i>Nest.js v10+</i>	<i>Framework backend escalable y modular, ideal para APIs RESTful.</i>
<i>MongoDB 6.0+</i>	<i>Base de datos NoSQL flexible, adecuada para datos no estructurados como horarios y solicitudes</i>
<i>TypeScript</i>	<i>Lenguaje base para Nest.js, agrega tipado estático y mejora la mantenibilidad</i>

Tabla 12.*Condiciones de software del usuario (frontend)*

Descripción	Justificación
React v18+	Biblioteca frontend para interfaces dinámicas y reactivas, ideal para gestión de solicitudes en tiempo real.
AWS Amplify	Plataforma para despliegue automático, hosting escalable y CI/CD integrado.
Axios	Cliente HTTP para comunicación eficiente con APIs de Nest.js
React Router v6	Navegación SPA (Single Page Application) fluida y estructurada.
Vite	Bundler rápido para desarrollo y optimización de builds en producción.
Navegadores	Chrome v100+, Firefox v100+, Edge v100+. Compatibilidad con React y estándares modernos.

4.2.1.4.3 Requisitos de desempeño.

A continuación se detallan los requisitos de actuación con respecto al sistema:

Tabla 13.*Requisitos de desempeño*

Requisito	Descripción
Tiempo de Respuesta	≤ 3 segundos para operaciones críticas.
Disponibilidad	99% (con redundancia en railway y balanceo de carga entre instancias).
Concurrencia	Soporte para 100 usuarios simultáneos durante picos de matrícula.
Escalabilidad	5 instancias de backend para escalado automático.
Seguridad	<ul style="list-style-type: none"> JWT para autenticación y acceso a vistas. Encriptación bcrypt.
Rendimiento Frontend	Carga inicial de la app en ≤ 1.5 segundos (optimizada con Vite y CDN de Amplify).

Este ajuste garantiza un sistema moderno, seguro y escalable, aprovechando las mejores prácticas de React y Nest.js.

5.2.1.4.4 Requisitos De Diseño Y De Interfaz.

El diseño del Sistema de Ajuste de Matrícula (SAM) busca garantizar una experiencia de usuario intuitiva y profesional. A continuación se detallan los principios y elementos clave:

Tabla 14.*Requerimiento de diseño e interfaz*

Principio	Descripción	Ejemplo
<i>Facilidad de acceso</i>	<i>Acceso rápido y sin barreras para estudiantes y administrativos.</i>	<i>Autenticación simplificada con correo institucional y código único.</i>
<i>Velocidad de carga</i>	<i>Optimización de recursos para reducir tiempos de espera. Uso de colores monocromáticos y elementos gráficos coherentes.</i>	<i>Uso de Vite para carga dinámica de componentes. Uso de colores blanco y negro con distintos gradientes aplicados, para mostrar prioridades.</i>
<i>Armonía visual</i>	<i>Consistencia en iconos, tipografía y disposición de elementos.</i>	<i>Iconos de Lucide-React y tipografía Montserrat en todas las pantallas.</i>
<i>Homogeneidad</i>	<i>Priorización de áreas críticas (ej: formularios de solicitud, panel administrativo).</i>	<i>Diseño modular con secciones destacadas para "Generar Solicitud" y "Validar Horarios".</i>
<i>Funcionalidad</i>		

Elementos Clave**Componentes Reutilizables:**

- Botones con estilos predefinidos (ej: "Enviar Solicitud", "Cancelar").
- Menús desplegables para selección de asignaturas y horarios.

Multimedia:

- Uso limitado de animaciones (solo en transiciones críticas, como confirmación de solicitudes).
- Imágenes oficiales: Logo UIS y escudo institucional en la cabecera.

Responsividad:

- Adaptación a dispositivos móviles (ej: pantallas de hasta 320px de ancho).
- Grids flexibles con CSS Grid para organización de horarios.

4.2.1.4.5 Requisitos de base de datos.

Para el almacenamiento de la información dentro del sistema, es necesario el uso de una base de datos. Esta debe estar debidamente diseñada, normalizada y validada a fin de garantizar el registro adecuado de la información.

Se seleccionó MongoDB, por ser un sistema de gestión de bases de datos NoSQL orientado a documentos, altamente escalable y flexible, que permite almacenar datos en formato JSON. Además, su compatibilidad con NestJS y Mongoose facilita la integración con la API del sistema.

La base de datos debe cumplir con los siguientes requisitos:

- Estructura bien definida con modelos de datos para usuarios, solicitudes, horarios y asignaturas.
- Validación de datos en los esquemas para evitar inconsistencia en la información almacenada.
- Índices adecuados para optimizar las consultas y mejorar el rendimiento.
- Implementación de medidas de seguridad como cifrado de contraseñas y control de acceso a los datos.

4.2.1.4.6 Requisitos de entradas.

El sistema debe permitir la captura de datos a través de diferentes dispositivos de entrada, incluyendo computadores, tabletas y dispositivos móviles, asegurando una experiencia de usuario óptima en cada uno de ellos.

Se deben aplicar validaciones y mecanismos de control para garantizar la integridad de los datos ingresados:

- Diseño adaptable a diferentes dispositivos, con formularios optimizados para pantallas de escritorio y móviles.

- Minimización de la intervención manual, mediante menús desplegables, calendarios, tabs y botones, que facilitan la entrada de datos en los formatos requeridos.
- Validación de credenciales de acceso, asegurando que solo usuarios autorizados puedan ingresar al sistema mediante correo y contraseña.
- Emisión de mensajes de error, notificando al usuario cuando se introducen caracteres incorrectos o datos inválidos.
- Validación de campos obligatorios, impidiendo el envío de formularios sin completar los datos esenciales.
- Validación de tipos de datos, garantizando que los valores ingresados sean correctos según el campo correspondiente.
- Verificación de conflictos en horarios, para evitar solapamientos de asignaturas al momento de hacer ajustes en la matrícula.
- Confirmaciones en acciones críticas como envío de solicitudes o guardado de horarios, mediante mensajes de confirmación.

4.2.1.4.7 Requisitos de salida.

Las salidas del sistema están basadas en la información procesada y almacenada, con el objetivo de mejorar la experiencia del usuario y reducir el tiempo en tareas administrativas.

A continuación, se detallan los distintos tipos de salida que debe generar el sistema:

Visualización de horarios, asignaturas y solicitudes:

- Los estudiantes podrán ver su horario de clases con materias y horarios asignados.
- Los estudiantes podrán ver sus solicitudes y el estado de cada una de estas.
- Los administrativos podrán visualizar todas las solicitudes resueltas por ellos y cada una de sus peticiones.
- El root podrá visualizar todas las solicitudes y cada una de sus peticiones.

Reportes y estadísticas:

- El sistema debe generar reportes sobre las solicitudes aprobadas, rechazadas y pendientes para el root.

Generación y Envío de Documentos

- Confirmaciones de aprobación o rechazo de solicitudes, enviadas al correo institucional del estudiante.

4.2.1.4.8 Requisitos de documentación.

Documentación para Estudiantes

La capacitación para los estudiantes se proporcionará mediante un video explicativo, donde se detallará el uso del sistema, incluyendo:

- Registro e inicio de sesión.
- Consulta y ajustes de horarios.
- Envío de solicitudes y seguimiento de estado.
- Recepción de notificaciones al correo.

Documentación para Administradores

Los administrativos adicional al video explicativo recibirán capacitación por parte del equipo responsable del sistema, con una sesión explicativa en la que se cubrirán:

- Gestión de solicitudes de estudiantes.
- Aprobación, modificación y rechazo de solicitudes.
- Administración de usuarios.

4.2.1.5 Modelos de caso de uso.

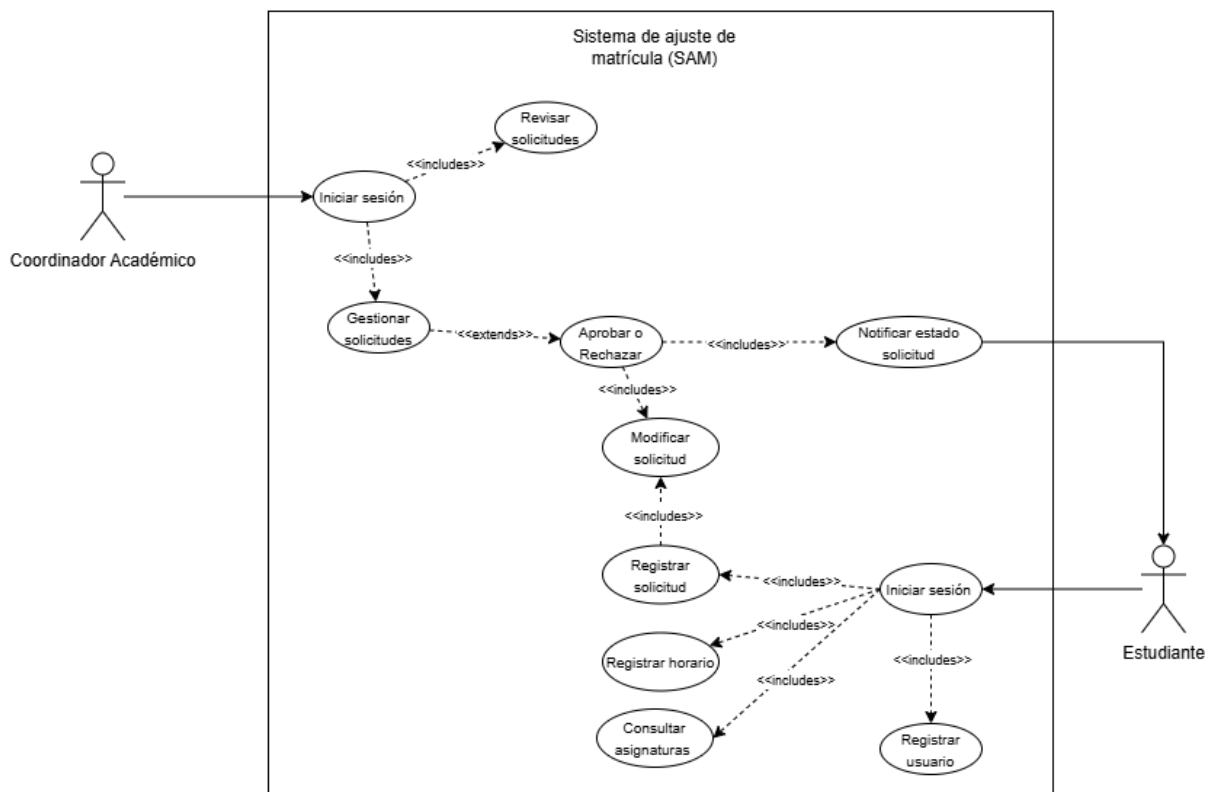
Representa el refinamiento del modelo preliminar de casos de uso, resultante en el proceso de visión del proyecto, en esta oportunidad se modela y describe de forma más precisa las acciones de los usuarios como las posibles reacciones del sistema en el entorno de un sistema de información, por lo quedará especificada las

acciones de los usuarios con cada módulo del sistema, así como la relación de la base de datos con los servicios previstos en la interfaz gráfica.

A continuación se presentan los diagramas de casos de uso planteados para cada uno de los subsistemas definidos para el diseño de la propuesta.

Figura 6.

Diagrama de casos de uso del sistema propuesto



4.3 Fase de construcción:

El objetivo de esta fase es alcanzar la capacidad operativa del producto de manera incremental a través de iteraciones sucesivas. Durante este proceso, se lleva a cabo la integración de todos los componentes del sistema y la implementación de los requisitos establecidos. Asimismo, se realizan pruebas funcionales para verificar el cumplimiento de las especificaciones, asegurando la estabilidad y operatividad del producto en su versión final.

El hito de esta fase se alcanza con el desarrollo completo del sistema en su versión final, cumpliendo con los estándares de calidad de producción y estando plenamente operativo para el inicio de la fase de transición. En este punto, se han implementado y superado satisfactoriamente todas las pruebas de calidad, garantizando la estabilidad, funcionalidad y conformidad del sistema con los requisitos establecidos.

4.3.1 Análisis y diseño.

El principal objetivo del análisis y diseño es transformar los requerimientos en un modelo estructurado que describa la implementación del sistema. El análisis se enfoca en definir una visión clara del sistema a desarrollar, especificando qué debe hacer, con base en los requerimientos funcionales. Por otro lado, el diseño se orienta a la solución técnica, abordando los requerimientos no funcionales y determinando cómo el sistema cumplirá sus objetivos, considerando aspectos como arquitectura, rendimiento, seguridad y escalabilidad.

4.3.1.1 Análisis del comportamiento del sistema.

Esta actividad facilita el análisis de las necesidades tanto de la dirección de la escuela como de los estudiantes, quienes actúan como usuarios finales del sistema. Su propósito es garantizar que la solución propuesta satisfaga sus expectativas y optimice los procesos académicos y administrativos.

Tabla 15.

Análisis del entorno del sistema

<i>Actividades</i>	<i>Características</i>
--------------------	------------------------

<i>Análisis de audiencia</i>	<i>El sistema está dirigido a estudiantes y personal académico-administrativo. Los usuarios tienen distintos niveles de alfabetización digital, por lo que se requiere una interfaz intuitiva y accesible.</i>
<i>Entorno organizacional</i>	<i>Involucra a la Dirección Académica, Secretaría de Registro, Coordinaciones de Carrera y estudiantes, quienes interactúan para gestionar modificaciones en la matrícula.</i>
<i>Entorno tecnológico</i>	<i>Requiere integración con bases de datos académicas, plataformas de gestión educativa, servidores, redes y aplicaciones web/móviles para el acceso de los usuarios.</i>
<i>Entorno operativo</i>	<i>Funciona en períodos específicos del semestre académico, interactuando con otros sistemas (Sistema de información UIS) y soportando alta demanda durante estos procesos.</i>
<i>Entorno social y de usuarios</i>	<i>Considera la resistencia al cambio, la necesidad de respuestas rápidas y la asistencia técnica en caso de problemas con el sistema.</i>

4.3.1.2 Definición de arquitectura.

A partir del modelado del negocio, la definición de requisitos y el análisis de casos de uso, se desarrolló un conjunto de vistas que conforman la arquitectura del sistema. Estas vistas incluyen la funcional, estructural y de comportamiento, las cuales representan los productos finales de este diseño.

4.3.1.2.1 Vista funcional o de uso de la aplicación web.

Esta vista define la versión final del modelo de casos de uso desarrollado durante la fase de elaboración. En esta etapa, se lleva a cabo un modelado más detallado de las acciones que los usuarios ejecutan en interacción con el sistema, permitiendo una mayor precisión en la especificación de los procesos.

A continuación, se presenta el diagrama de casos de uso diseñados para representar cada una de las funcionalidades establecidas en la construcción de la propuesta. Estos diagramas reflejan la relación entre los actores del sistema y las operaciones que pueden realizar, proporcionando una visión clara de los requisitos funcionales del sistema.

Figura 7.

Diagrama de casos de uso del sistema desarrollado

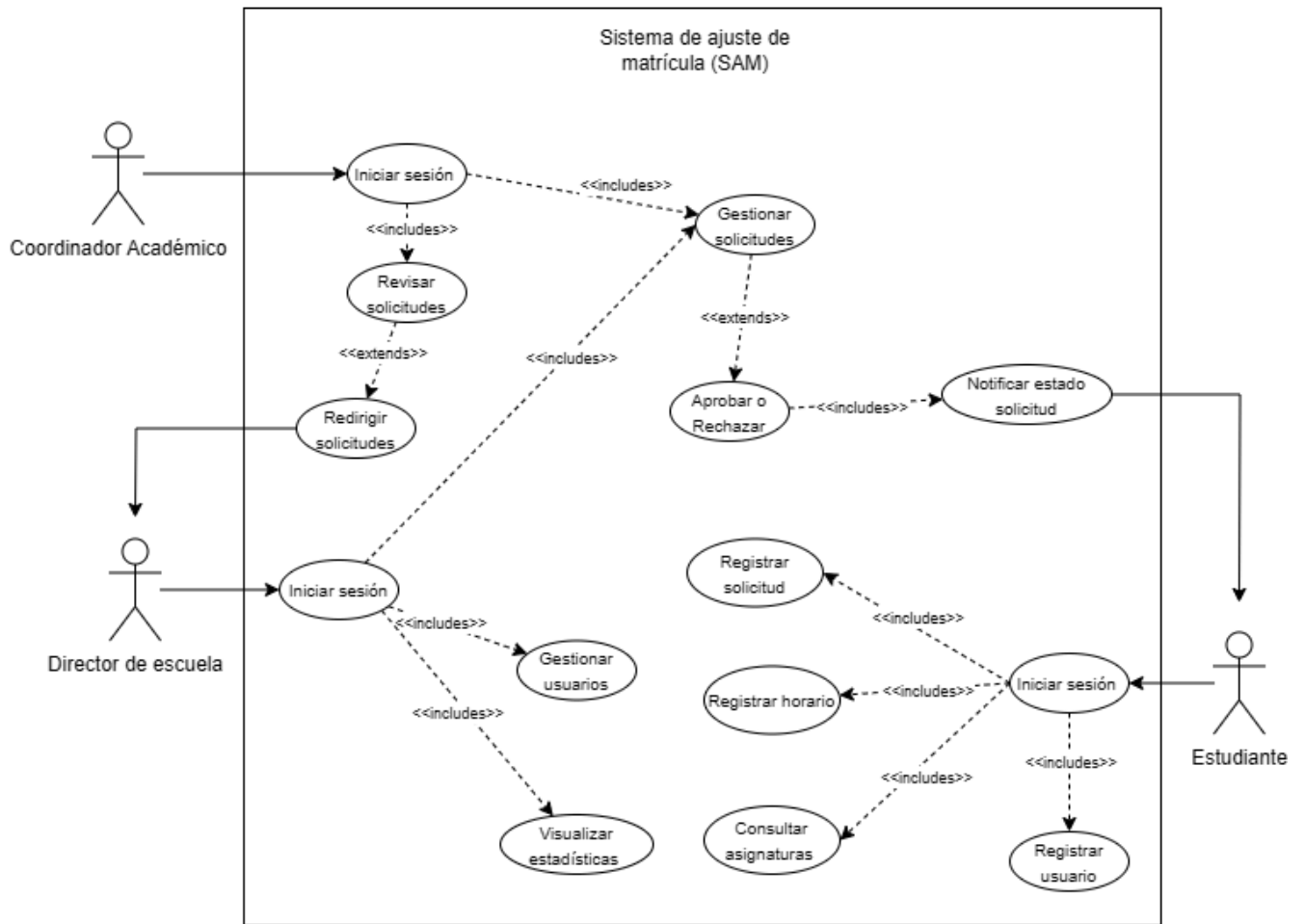


Tabla 16.

Descripción de casos de uso

Actor	Casos de uso	Flujo de eventos
	Administrar usuario	<ul style="list-style-type: none"> Realiza CRUD de usuarios de tipo estudiante y de tipo administrador Asigna login y contraseña al usuario de tipo administrador
Root	Supervisar solicitudes de ajuste de matrícula Revisión de estadísticas	<ul style="list-style-type: none"> Supervisa el estado de las solicitudes y las operaciones de los administradores Consulta métricas sobre el número de ajustes de matrícula, tiempos de procesamiento y tendencias de uso.

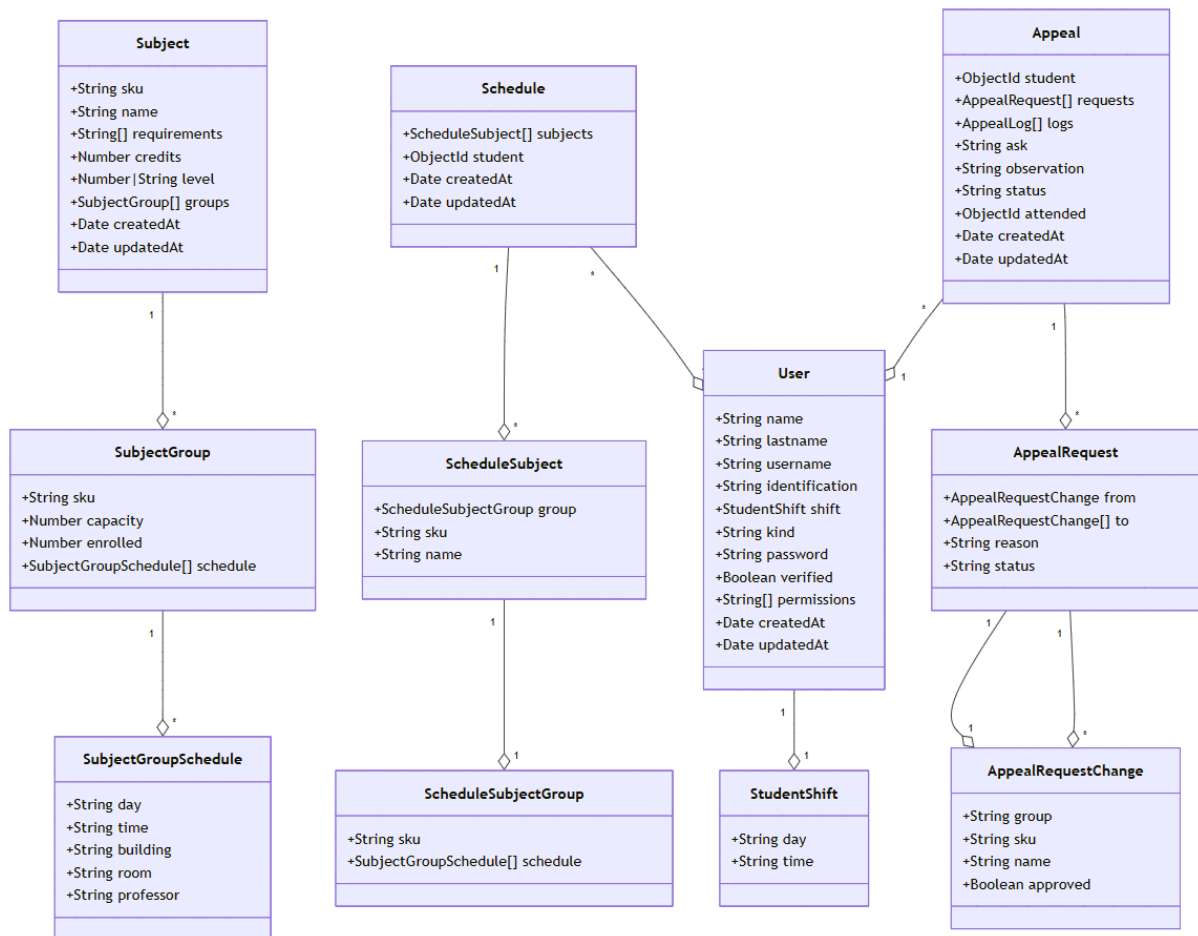
	<i>Procesar solicitudes</i>	<ul style="list-style-type: none"> ● <i>Evalúa cada solicitud verificando cupos, requisitos y restricciones.</i> ● <i>Aprueba o rechaza la solicitud y notifica al estudiante.</i> ● <i>recibe solicitudes de ajuste de matrícula enviadas por el administrador</i>
<i>Administrador</i>	<i>Redirige solicitudes al director</i>	<ul style="list-style-type: none"> ● <i>En caso de no tenerlos los permisos necesarios para poder realizar una solicitud, podrá enviarla a l director para ser procesada</i> ● <i>Aprueba o rechaza la solicitud y notifica al estudiante.</i>
	<i>Gestionar solicitudes</i>	
<i>Estudiante</i>	<i>Registrar horario</i>	<ul style="list-style-type: none"> ● <i>Guarda el horario en su perfil</i>
	<i>Crear solicitudes</i>	<ul style="list-style-type: none"> ● <i>Selecciona el tipo de solicitud (agregar, eliminar o cambiar una asignatura).</i> ● <i>Ingresa los detalles de la asignatura y justificación del cambio.</i> ● <i>Envía la solicitud para su revisión por parte del administrador.</i> ● <i>Recibe una notificación sobre el estado de la solicitud</i>
	<i>Consultar asignaturas</i>	<ul style="list-style-type: none"> ● <i>Visualiza las asignaturas disponibles, con detalles como cupos, horarios y profesores.</i>
	<i>Ver historial de solicitudes</i>	<ul style="list-style-type: none"> ● <i>Visualiza el estado y detalle de todas las solicitudes de ajuste de matrícula realizadas.</i> ● <i>Crear asignaturas faltantes en el listado</i>
	<i>Crear asignaturas no registradas</i>	

4.3.1.3 Diseño de base de datos.

En esta actividad se identifican los esquemas de diseño que formarán parte de la base de datos del sistema, teniendo en cuenta que este es de un tipo no relacional.

Figura 8.

Diseño base de datos nosql



A partir de este diseño de una base de datos NoSQL desarrollada con MongoDB se construyó el sistema de ajuste de matrícula (SAM). Este sistema se ha concebido para manejar de manera eficiente la información relacionada con estudiantes, materias, horarios y solicitudes en un entorno educativo moderno.

La arquitectura de la base de datos se fundamenta en un modelo documental que aprovecha las ventajas de la flexibilidad y escalabilidad propias de MongoDB. El sistema se estructura en cuatro colecciones principales que interactúan entre sí para proporcionar una solución integral: Users, Subjects, Schedule y Appeals.

Usuarios

La colección Users constituye el núcleo del sistema, almacenando la información fundamental de todos los usuarios del sistema de ajuste de matrícula. Cada documento en esta colección representa a un usuario individual, ya sea estudiante, administrador o usuario root, conteniendo sus datos personales, credenciales de acceso y permisos específicos. Para los estudiantes, se incluye información adicional como su turno asignado, que se implementa como un subdocumento embebido, permitiendo una gestión eficiente de los horarios y disponibilidad.

Materias

En cuanto a la colección Subjects, esta almacena el catálogo completo de materias disponibles en la carrera de ingeniería de sistemas, además de la adición de electivas, contextos y materias libres de crédito. Cada materia se representa como un documento que contiene no solo la información básica como código y nombre, sino también una estructura compleja de grupos y horarios. Los grupos se implementan como arreglos de subdocumentos, donde cada grupo contiene su propia información de capacidad, cantidad de estudiantes inscritos y los horarios específicos con detalles sobre profesor, aula y edificio.

Horarios

La gestión de horarios se realiza a través de la colección Schedule, que establece una relación entre estudiantes y sus materias matriculadas en el horario. Esta colección implementa un sistema de referencias mediante ObjectId para mantener la integridad referencial con la colección Users, mientras que almacena la información específica de las materias y grupos seleccionados por cada estudiante. Esta estructura permite consultas eficientes para obtener los horarios individuales y facilita la verificación de conflictos de horario.

Solicitudes

El sistema de solicitudes, representado por la colección Appeals, se diseñó para manejar las solicitudes de cambios y modificaciones que los estudiantes pueden requerir en sus horarios. Cada documento de solicitud mantiene un registro detallado del estado actual de la solicitud, los cambios solicitados y el seguimiento administrativo de la solicitud.

La implementación de este diseño se realizó considerando patrones óptimos para bases de datos NoSQL. Se utilizó una combinación estratégica de documentos embebidos y referencias. Los documentos embebidos se emplearon para información que típicamente se consulta y actualiza en conjunto, como los horarios dentro de los grupos de materias. Las referencias, por otro lado, se utilizaron para establecer relaciones entre colecciones independientes, como la relación entre estudiantes y sus horarios.

Para optimizar el rendimiento, se implementaron índices estratégicos en campos clave. La colección Users mantiene índices únicos en campos como username e identification, mientras que la colección Appeals incluye índices compuestos para facilitar las búsquedas por estado y estudiante. Estos índices mejoran significativamente la velocidad de las consultas más frecuentes del sistema.

El sistema implementa un robusto esquema de validación utilizando el framework @nestjs/mongoose, que garantiza la integridad de los datos ingresados. Se definieron enumeraciones para estados y tipos de usuarios, asegurando la consistencia en los valores almacenados. Además, todas las colecciones principales incluyen timestamps automáticos que registran las fechas de creación y última modificación de cada documento.

La base de datos se diseñó pensando en la escalabilidad horizontal, permitiendo el crecimiento natural del sistema sin comprometer el rendimiento. La estructura modular de las colecciones facilita la adición de nuevas funcionalidades o la modificación de las existentes sin afectar la integridad del sistema.

Este diseño ha demostrado ser eficiente en la práctica, permitiendo operaciones comunes como la consulta de horarios, la verificación de disponibilidad horaria en grupos y el seguimiento de solicitudes de manera rápida y confiable. La combinación de documentos embebidos y referencias proporciona un balance óptimo entre rendimiento y flexibilidad, mientras que los índices estratégicos garantizan tiempos de respuesta consistentes incluso con grandes volúmenes de datos.

La implementación de este diseño de base de datos NoSQL ha resultado en un sistema robusto y eficiente que cumple con los requerimientos de un entorno académico moderno, proporcionando la flexibilidad necesaria para adaptarse a futuras necesidades y cambios en los procesos institucionales.

4.3.1.3.1 Diccionario de datos.

Los diccionarios de datos constituyen un componente fundamental en el análisis estructurado de sistemas de información. Mientras que los diagramas de flujo de datos proporcionan una visión general del sistema, los diccionarios de datos profundizan en la estructura y organización de la información, describiendo detalladamente cómo los datos están estructurados para satisfacer tanto los requerimientos de los usuarios como las necesidades organizacionales.

Para el presente sistema, se ha diseñado una estructura de base de datos NoSQL que se compone de 4 colecciones principales, cada una con sus respectivos subdocumentos embebidos. A continuación, se describe el diccionario de datos que detalla la funcionalidad

de cada colección, las características de sus campos y su capacidad de almacenamiento proyectada.

Tabla 17.

Diccionario de datos. colección: usuarios

Campo	Tipo	Descripción	Registros
<i>_id</i>	<i>ObjectId (12B)</i>	<i>Clave primaria (generada por MongoDB)</i>	
<i>name</i>	<i>String (50)</i>	<i>Nombre del usuario.</i>	
<i>lastname</i>	<i>String (50)</i>	<i>Apellido del usuario</i>	
<i>username</i>	<i>String (20)</i>	<i>Correo del usuario (único)</i>	278
<i>identificatio n</i>	<i>String (20)</i>	<i>Código del estudiante (única, opcional)</i>	
<i>kind</i>	<i>Enum</i>	<i>Tipo de usuario (STUDENT, ADMIN, ROOT)</i>	
<i>password</i>	<i>String (60)</i>	<i>Hash de contraseña</i>	
<i>permissions</i>	<i>Array[String]</i>	<i>Lista de permisos</i>	
<i>verified</i>	<i>Boolean (1B)</i>	<i>Estado de verificación de la cuenta</i>	

Tabla 18.

Diccionario de datos. colección: materias

Campo	Tipo	Descripción	Registros
<i>_id</i>	<i>ObjectId (12B)</i>	<i>Clave primaria (generada por MongoDB)</i>	
<i>sku</i>	<i>String (20)</i>	<i>Código único de la materia.</i>	
<i>name</i>	<i>String (100)</i>	<i>Nombre de la materia</i>	
<i>groups</i>	<i>Array[Object]</i>	<i>Grupos dentro de la materia</i>	
<i>level</i>	<i>Mixed (Number, String)</i>	<i>Nivel académico</i>	104
<i>requirement s</i>	<i>Array[String]</i>	<i>Materias requeridas (referencia a SKUs)</i>	
<i>credits</i>	<i>Number (4B)</i>	<i>Créditos académicos</i>	

Tabla 19.

Diccionario de datos. colección: horario

Campo	Tipo	Descripción	Registros
<i>_id</i>	<i>ObjectId (12B)</i>	<i>Clave primaria (generada por MongoDB)</i>	
<i>student</i>	<i>ObjectId (12B)</i>	<i>Referencia a Users (FK)</i>	249
<i>subjects</i>	<i>Array[Object]</i>	<i>Materias inscritas</i>	

Tabla 20.

Diccionario de datos. colección: solicitudes

Campo	Tipo	Descripción	Registros
<i>_id</i>	<i>ObjectId (12B)</i>	<i>Clave primaria (generada por MongoDB)</i>	
<i>student</i>	<i>ObjectId (12B)</i>	<i>Referencia a Users (FK)</i>	
<i>status</i>	<i>Enum</i>	<i>Estado de la solicitud (PENDING, APPROVED, REJECTED, PARTIAL_REJECTED, REVIEW)</i>	
<i>request</i>	<i>Array[Object]</i>	<i>Lista de peticiones</i>	393
<i>s</i>	<i>ObjectId (12B)</i>	<i>Referencia a Users (admin, root)</i>	
<i>attended</i>	<i>String</i>	<i>Descripción de la solicitud</i>	
<i>ask</i>	<i>String</i>	<i>Observaciones administrativas</i>	
<i>observation</i>			

Tipos de datos clave

- ObjectId → Identificadores únicos en MongoDB.
- String → Texto en formato UTF-8.
- Number → Enteros o decimales (4 bytes).
- Boolean → true o false.
- Array → Listas de datos.
- Object → Documentos embebidos

4.3.1.4 Diseño de interfaz gráfica.

Este diseño de interfaz de usuario describe el esquema que tendrá la aplicación y cómo son las interfaces principales. Se describen las características de los usuarios, se define un mapa de navegación y se detallan los elementos de diseño de la interfaz de los distintos usuarios, permite crear un diseño de la interfaz de usuario que apoye el razonamiento y el realce de su usabilidad.

4.3.1.4.1 Perfil de usuarios.

El perfil de usuario permite describir aquellas personas que están en contacto directo con el sistema y hacen uso de los servicios que el producto ofrece. A continuación la siguiente tabla:

Tabla 21.

Perfil de usuarios

<i>Usuario</i>	<i>Perfil</i>
<i>Root</i>	<i>Tiene acceso total al sistema. Gestiona usuarios (crear, modificar), accede a estadísticas avanzadas y gestiona solicitudes complejas que le envía el personal administrativo. Puede ver y resolver todas las solicitudes creadas en el sistema.</i>
<i>Admin</i>	<i>Gestiona solicitudes de ajuste de matrícula: aprueba, rechaza o deriva casos al Root. Se le asignan las solicitudes 1 a 1 automáticamente al completar una.</i>
<i>Student</i>	<i>Interactúa con el sistema para solicitar ajustes de matrícula, visualizar o registrar su horario actual, consultar disponibilidad de grupos y recibir notificaciones mediante el correo. Solo puede acceder y modificar su propia información.</i>

4.3.1.4.2 Mapa de navegación.

Proporcionan una representación esquemática de la estructura del hipertexto, indicando los principales conceptos incluidos en el espacio de la información y las interrelaciones que existen entre ellos, y orientar al lector/usuario durante el recorrido o para facilitarle acceso directo al lugar que le interese. Los módulos de la Aplicación Web, se definieron a partir de los requerimientos realizados inicialmente, del análisis de los datos para el desarrollo de este

trabajo, además se creó la estructura de navegación basada en el criterio y necesidades de los usuarios finales.

Figura 9.

Mapa de navegación

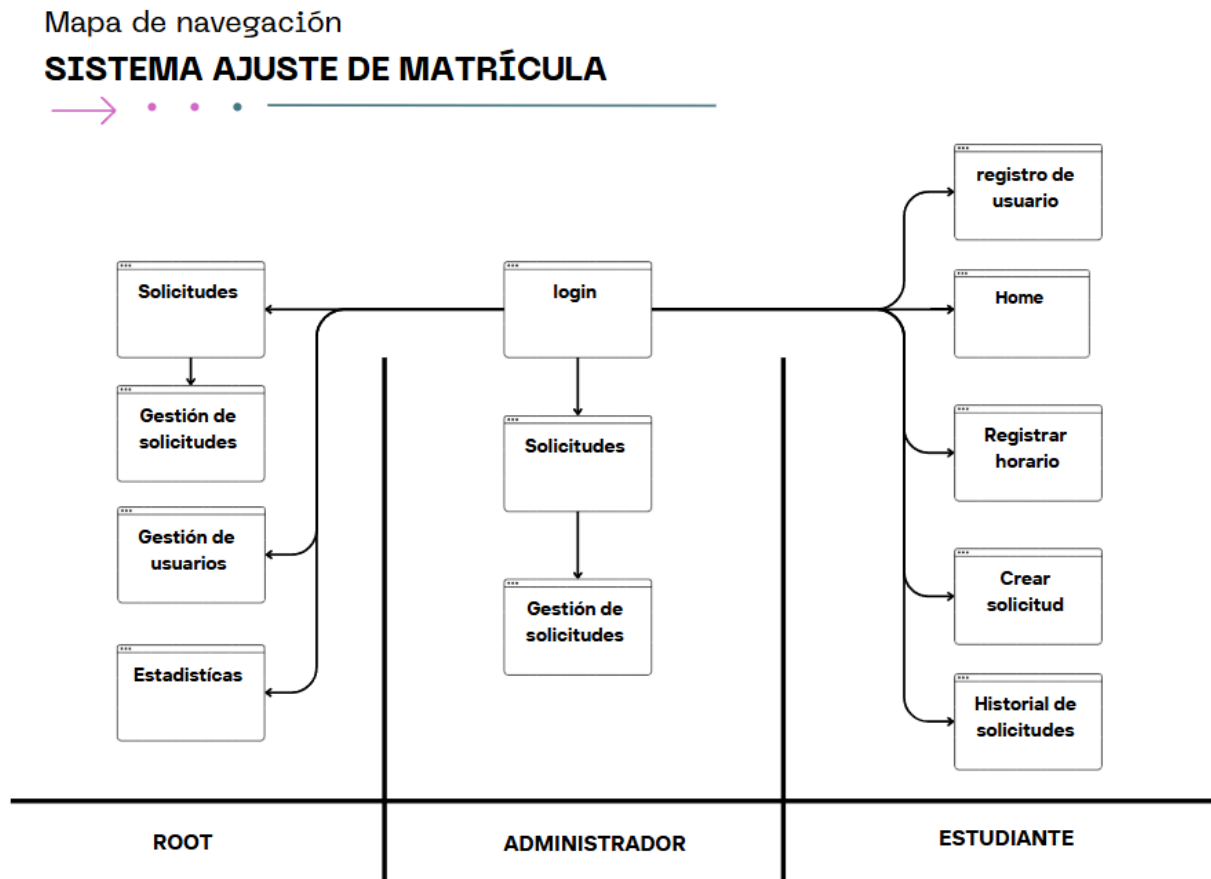


Tabla 22.

Descripción modular de mapa de navegación

Módulo relacionado	Funciones
Registro de usuarios	Permite a los estudiantes crear una cuenta en el sistema, registrando sus datos personales y credenciales de acceso para autenticarse y utilizar las funcionalidades del sistema.
Login	Verifica las credenciales ingresadas por los usuarios, asignando un token de sesión para mantener la autenticación y controlar el acceso a los diferentes módulos del sistema.
Solicitudes	Permite al usuario root y administrador visualizar las solicitudes registradas por los estudiantes y el estado en el cual se encuentran Facilita a los administradores la revisión y procesamiento de solicitudes de ajuste de matrícula, asegurando un flujo de aprobación o rechazo con sus motivos respectivos.

<i>Gestión de solicitudes</i>	<i>Permite al usuario root administrar las cuentas del sistema, incluyendo la creación, modificación y eliminación de usuarios de tipo estudiantes y administradores.</i>
<i>Gestión de usuarios</i>	<i>Proporciona al usuario root reportes sobre la cantidad de solicitudes creadas, porcentaje de solicitudes atendidas y pendientes, tiempo promedio de respuesta, cantidad de solicitudes por franja.conteo de intentos de ajuste de matrícula por estudiante y solicitudes atendidas en total por cada franja.</i>
<i>Estadísticas</i>	<i>Permite a los estudiantes ingresar y gestionar su horario académico.</i>
<i>Registro de horario académico</i>	<i>Por medio de este módulo los estudiantes podrán registrar una solicitud de ajuste de matrícula, validando posibles conflictos entre asignaturas Los estudiantes pueden revisar las solicitudes creadas, revisar el estado en el que se encuentran y poder ver los motivos de rechazo en caso de ser necesario.</i>
<i>Crear solicitud</i>	
<i>Historial de solicitudes</i>	

4.3.1.5 Diseño de servicios.

Tabla 23.

Descripción de servicios

Módulo	Servicio	Contenido
<i>Autenticación y Seguridad</i>	<i>Gestión de Usuarios</i>	<i>Formulario de registro (código UIS, correo institucional, contraseña). Asignación de roles (Root, Administrador, Estudiante). Validación de correos institucionales (@uis.edu.co). Formulario de selección de asignaturas y grupos.</i>
<i>Solicitudes de Ajuste</i>	<i>Generar Solicitud</i>	<i>Validación automática de conflictos horarios. Envío de solicitud con indicaciones opcionales. Panel administrativo para aprobar/rechazar solicitudes.</i>
<i>Listado Solicitudes</i>	<i>Gestionar Solicitudes</i>	<i>Priorización por franja asignada. Derivación de casos complejos al Root. Interfaz interactiva con calendario semanal.</i>
<i>Horarios Académicos</i>	<i>Visualización de Horarios Generar Estadísticas</i>	<i>Detección visual de conflictos horarios. Formulario de selección de asignaturas y grupos. Gráficos de estados, porcentajes y materias. Dashboards en tiempo real para Root.</i>
<i>Estadísticas</i>	<i>Envío de alertas</i>	<i>Notificaciones en la interfaz (Toast messages). Correos automáticos de confirmación de solicitudes.</i>
<i>Notificaciones</i>		

4.3.1.6 Prototipo de interfaz.

Después de definir los perfiles de usuario, servicios y contenido, se desarrolló el prototipo de interfaz de usuario considerando que la aplicación será utilizada por estudiantes, profesores y personal administrativo del ámbito académico. El diseño se realizó siguiendo principios fundamentales de usabilidad y diseño moderno, que se detallan a continuación:

Principios de Diseño Implementados

Comunicación

- Implementación de un sistema de notificaciones por correo claro
- Indicadores de ubicación en la navegación
- Tooltip de retroalimentación para todas las acciones
- Estado visible de solicitudes

Consistencia

- Diseño monocromático
- Patrones de interfaz consistentes en todas las vistas
- Sistema de navegación uniforme
- Componentes reutilizables con estilos homogéneos

Autonomía Controlada

- Sistema de autenticación robusto
- Perfiles de usuario diferenciados (estudiante, administrador, root)
- Acceso restringido según niveles de permiso
- Navegación intuitiva dentro de las áreas autorizadas

Eficiencia

- Interfaz minimalista y enfocada en tareas principales
- Accesos directos a funciones frecuentes
- Diseño responsivo que se adapta a diferentes dispositivos
- Optimización de pasos para completar tareas comunes

Flexibilidad

- Navegación clara con opciones de retroceso
- Diseño adaptable a diferentes resoluciones de pantalla
- Sistema de corrección de errores intuitivo
- Múltiples rutas para acceder a funcionalidades

Enfoque

- Priorización de información relevante
- Jerarquía visual clara
- Contenido organizado por relevancia
- Eliminación de elementos distractores

Características Técnicas del Diseño**Responsividad**

- Implementación de diseño mobile-first
- Layouts fluidos que se adaptan a diferentes tamaños
- Elementos flexibles que mantienen la usabilidad

Esquema de Colores

- Paleta monocromática
- Uso de tonos neutros para información secundaria
- Contraste optimizado para legibilidad
- Indicadores de estado con colores semánticos

Navegación

- Menú lateral colapsable
- Barra superior con accesos rápidos
- Menú móvil optimizado

Componentes Principales

Header

- Logo institucional
- Perfil de usuario
- Videotutorial de ayuda
- Menú de navegación principal
- Logout

Área de Contenido

- Títulos claros y descriptivos
- Información organizada en cards
- Formularios simplificados
- Tablas responsive

Elementos Interactivos

- Botones con estados visibles
- Formularios con validación instantánea
- Modales para acciones importantes
- Tooltips informativos

El diseño implementado prioriza la simplicidad y usabilidad, manteniendo una estética profesional y moderna que facilita la interacción con el sistema. La interfaz se ha optimizado para proporcionar una experiencia consistente en todos los dispositivos, manteniendo la funcionalidad y accesibilidad como principios fundamentales.

A continuación los prototipos de pantalla:

Figura 10.

Prototipo de interfaz inicio de sesión



El prototipo de interfaz de inicio de sesión, titulado "Ajuste de Matricula", presenta un diseño limpio y funcional. El encabezado incluye el título "Ajuste de Matricula" y el subtítulo "Ingrese sus credenciales para acceder al sistema". El formulario contiene dos campos de entrada: "Correo electrónico" con el valor "dilan2191976@correo.uis.edu.co" y "Contraseña" con caracteres ocultos por puntos y un ícono de ojo para alternar la visibilidad. Debajo de los campos se encuentra un botón de "Iniciar sesión" con un ícono de flecha a la derecha. En la parte inferior, se ofrecen enlaces para "¿Olvidó su Contraseña?" y "¿No tiene una cuenta? Regístrese aquí".

Figura 11.

Prototipo de interfaz registro

Registro de usuario

Crea una cuenta para acceder al sistema de ajuste de matrícula

? **Sistema de Ajuste de Matrícula (SAM)**
 Sistema exclusivo para estudiantes de Ingeniería de Sistemas.

Nombre

Apellido

Código

Correo institucional

Contraseña


Confirmar contraseña

+ Registrarse

[¿Ya tienes una cuenta? Ingresar aquí](#)

Figura 12.

Prototipo de interfaz inicio estudiante



Inicio Horario Solicitudes 🔍 👤

Bienvenido al sistema de ajuste de matrícula

⚠️ **¡Importante!**
 El éxito de su solicitud de ajuste de matrícula depende de que su horario y franja de atención estén registrados correctamente. Asegúrese de revisar y confirmar su horario antes de crear una solicitud de ajuste.

📅 **Cargar Horario**

Visualiza y organiza tu horario académico

En esta sección podrás:

- Registrar tu horario actual.
- Visualizar las materias registradas.
- Editar el horario en caso de errores.

Gestionar horario

📄 **Crear Solicitud de Ajuste**

Solicita cambios en tu matrícula académica

En esta sección podrás:

- Incluir o cancelar materias.
- Cambiar los grupos de las materias.
- Solicitar excepciones académicas.

Crear solicitud

Figura 13.

Prototipo de interfaz horario estudiante

Horario del estudiante

Hora	Lunes	Martes	Miércoles	Jueves	Viernes	Sábado
6-7		Cálculo I Grupo A1		Cálculo I Grupo A1	Cálculo I Grupo A1	
7-8						
8-9						
9-10						
10-11						
11-12						
12-13						
13-14						
14-15						
15-16						
16-17						
17-18		Estadística II Grupo F1		Estadística II Grupo F1		
18-19						
19-20						
20-21						
21-22						

Enviar

Asignaturas disponibles

Buscar asignatura +

Interacción Hombre Computador

Créditos: 4
Nivel: Electivas

Grupo B1

- MARTES, 8:00 - 10:00
JORGE LEONARDO SOTO GARNICA
- JUEVES, 8:00 - 10:00
JORGE LEONARDO SOTO GARNICA

+

Procesamiento de imágenes digitales

Algoritmos I

Análisis de datos a gran escala

Trabajo de investigación I

Optimización convexa

Principios y prácticas de...

Figura 14.

Prototipo de interfaz crear solicitud estudiante

Solicitud ajuste de matricula

Hora	Lunes	Martes	Miércoles	Jueves	Viernes	Sábado
6-7						
7-8		Cálculo I Grupo A1		Cálculo I Grupo A1	Cálculo I Grupo A1	
8-9	Cálculo I Grupo B1	Cálculo I Grupo B1		Cálculo I Grupo B1		
9-10						
10-11		Fundamentos de Programación Grupo PC1		Fundamentos de Programación Grupo PC1		
11-12						
12-13						
13-14						
14-15						
15-16						
16-17						
17-18						
18-19						
19-20						
20-21						
21-22						

Enviar

Asignaturas disponibles

Buscar asignatura +

Modelos de negocios en la sociedad de la información

Créditos: 4
Nivel: Electivas
Requisitos: Sistemas de Información

Grupo B1

- MARTES, 8:00 - 10:00
SONIA CRISTINA GAMBOA SARMIENTO
- JUEVES, 8:00 - 10:00
SONIA CRISTINA GAMBOA SARMIENTO

+

Visión por computador

Arquitectura Empresarial

Gestión del conocimiento

BALONCESTO (FEMENINO)

FUTBOL

FITRO | SAI | MASC | II INDI

Figura 15.

Prototipo de interfaz solicitudes

Nombre Estudiante	Código Estudiante	Franja Horaria	Estado	Atendido por	Tiempo Total	Acciones
Jhon Amaya	2220092	Jueves - Mañana	Parcialmente Completada	Juan Pernalete	1d 15h	Ver detalles
Camilo Andrés Campos Gamboa	2220099	Jueves - Mañana	Fallida	Juan Pernalete	1d 12h	Ver detalles
Brayan Julian Barrera Hernandez	2220097	Jueves - Mañana	Parcialmente Completada	Juan Pernalete	1d 11h	Ver detalles
Daniel Fernando Leal Ayala	2191430	Jueves - Mañana	Fallida	Juan Pernalete	1d 10h	Ver detalles
Luis Alejandro Vecino Daza	2220096	Jueves - Mañana	Fallida	Juan Pernalete	1d 10h	Ver detalles
Rafael Pinilla	2221929	Jueves - Mañana	Completada	Juan Pernalete	1d 10h	Ver detalles
DAVID ESTEBAN SUAREZ LOZANO	2221909	Jueves - Mañana	Parcialmente Completada	Juan Pernalete	1d 10h	Ver detalles
Sebastian Alvarado	2191969	Miércoles - Mañana	Fallida	Juan Pernalete	10h 4m	Ver detalles
Juan José Rojas Vanegas	2180043	Miércoles - Mañana	Completada	Juan Pernalete	2h 35m	Ver detalles
Cristhian Trisanchco	2192532	Miércoles - Mañana	Fallida	Juan Pernalete	1h 19m	Ver detalles

Figura 16.

Prototipo de interfaz gestionar solicitud

← Volver

Atendida por: Juan Pernalete

Estado de la solicitud: Parcialmente Completada

Datos del estudiante

Código del estudiante
2230052

Nombre del estudiante
Cristian Jacob Quintero Portilla

Franja de atención
Jueves - Mañana

Tiempo en espera de la solicitud
20h 16m

Lista de peticiones

Petición #1
Cambio del grupo B1 al A1 en la materia 22957 - Electricidad y Electrónica ✓ ✕

Petición #2
Cambio del grupo G1 al F2, F1 en la materia 22960 - Base de Datos II ✓ ✕

Capacidad de grupo

Indicaciones estudiante
No hay indicaciones del estudiante.

Observaciones adicionales
Ingrese observaciones adicionales para la solicitud del estudiante.

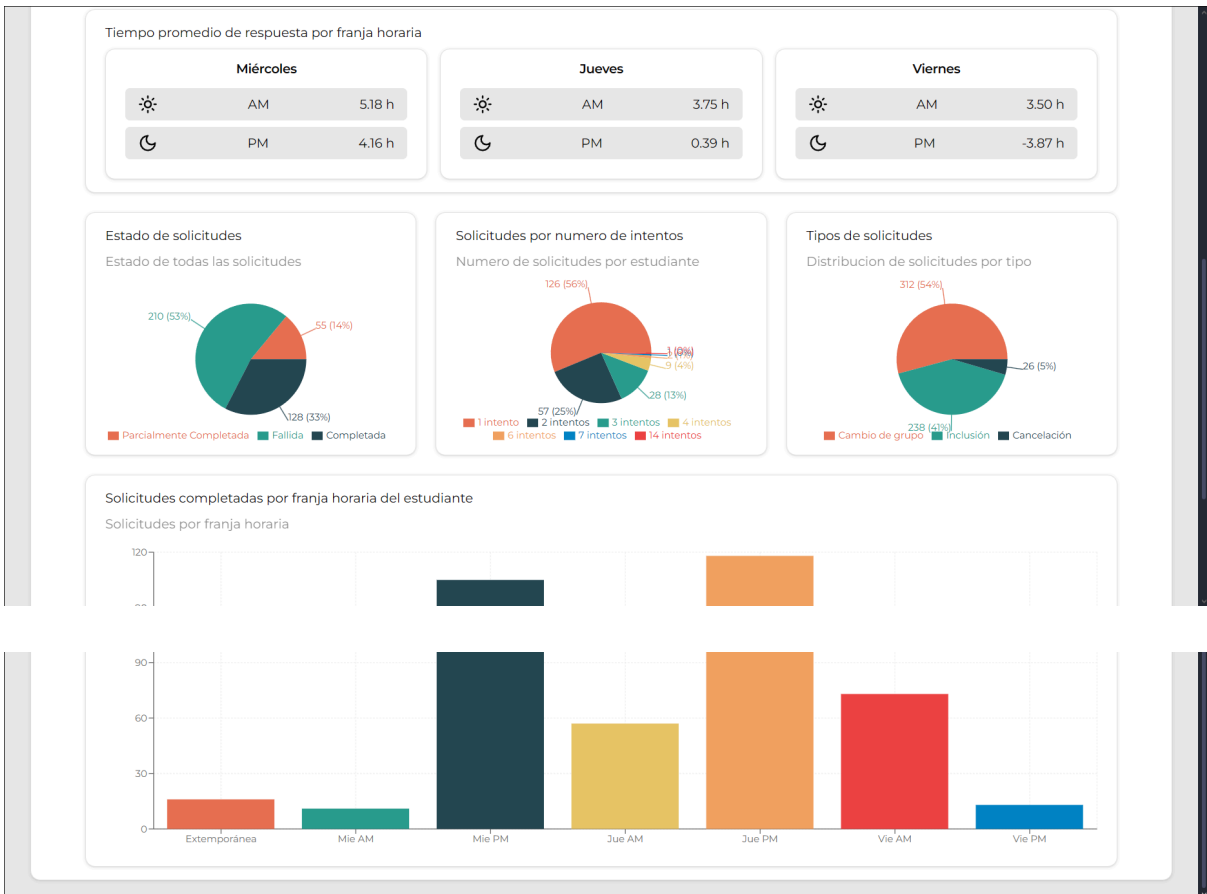
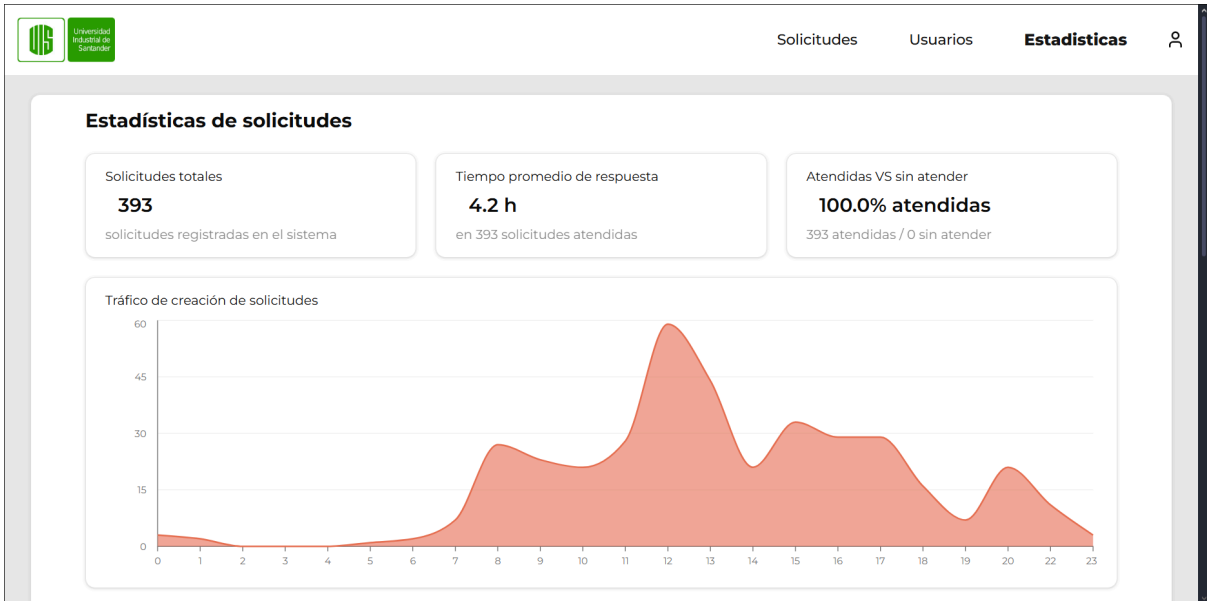
✓ Completada

Figura 17.*Prototipo de interfaz usuarios*

The screenshot displays a web interface for user management. At the top left is the logo of the Universidad Autónoma de Bucaramanga. The navigation menu includes 'Solicitudes', 'Usuarios' (highlighted), and 'Estadísticas', along with a user profile icon. The main content area is titled 'Usuarios' and features a search bar, a '+ Crear usuario' button, and a table of users. The table has columns for 'Nombre', 'Apellido', 'Email', 'Tipo', and 'Acciones'. The 'Acciones' column contains edit and delete icons. A pagination bar at the bottom shows '1' selected, with other numbers 2, 3, 4, 5, and 28, and a right arrow.

Nombre ↕	Apellido ↕	Email ↕	Tipo ↕	Acciones
Juan	Pernalete	eisi.laboratorios@correo.uis.edu.co	ADMIN	
Esteban	David	esteban2191940@correo.uis.edu.co	STUDENT	
Laura	Méndez	laura2191951@correo.uis.edu.co	STUDENT	
Jhon	Amaya	jhon2220092@correo.uis.edu.co	STUDENT	
Dilan	Corredor	dilan2191976@correo.uis.edu.co	STUDENT	
Santiago	Quintero	santiago2180445@correo.uis.edu.co	STUDENT	
Cesar	Hurtado Rodríguez	cesar2191957@correo.uis.edu.co	STUDENT	
Camilo Andrés	Campos Gamboa	camilo2220099@correo.uis.edu.co	STUDENT	
José Alejandro	Gómez Vargas	jose2220077@correo.uis.edu.co	STUDENT	
Luis Alejandro	Vecino Daza	luis2220096@correo.uis.edu.co	STUDENT	

Figura 18.*Prototipo de interfaz estadísticas*



4.3.2 Implementación.

4.3.2.1 Modelo de implementación.

El modelo de implementación del Sistema de Ajuste de Matrícula (SAM) describe la estructura física y lógica de los componentes que integran el sistema, incluyendo archivos, servicios, bibliotecas y dependencias. A continuación, se detalla la composición técnica y las relaciones entre los elementos clave:

Figura 19.

Diagrama de componentes del sistema

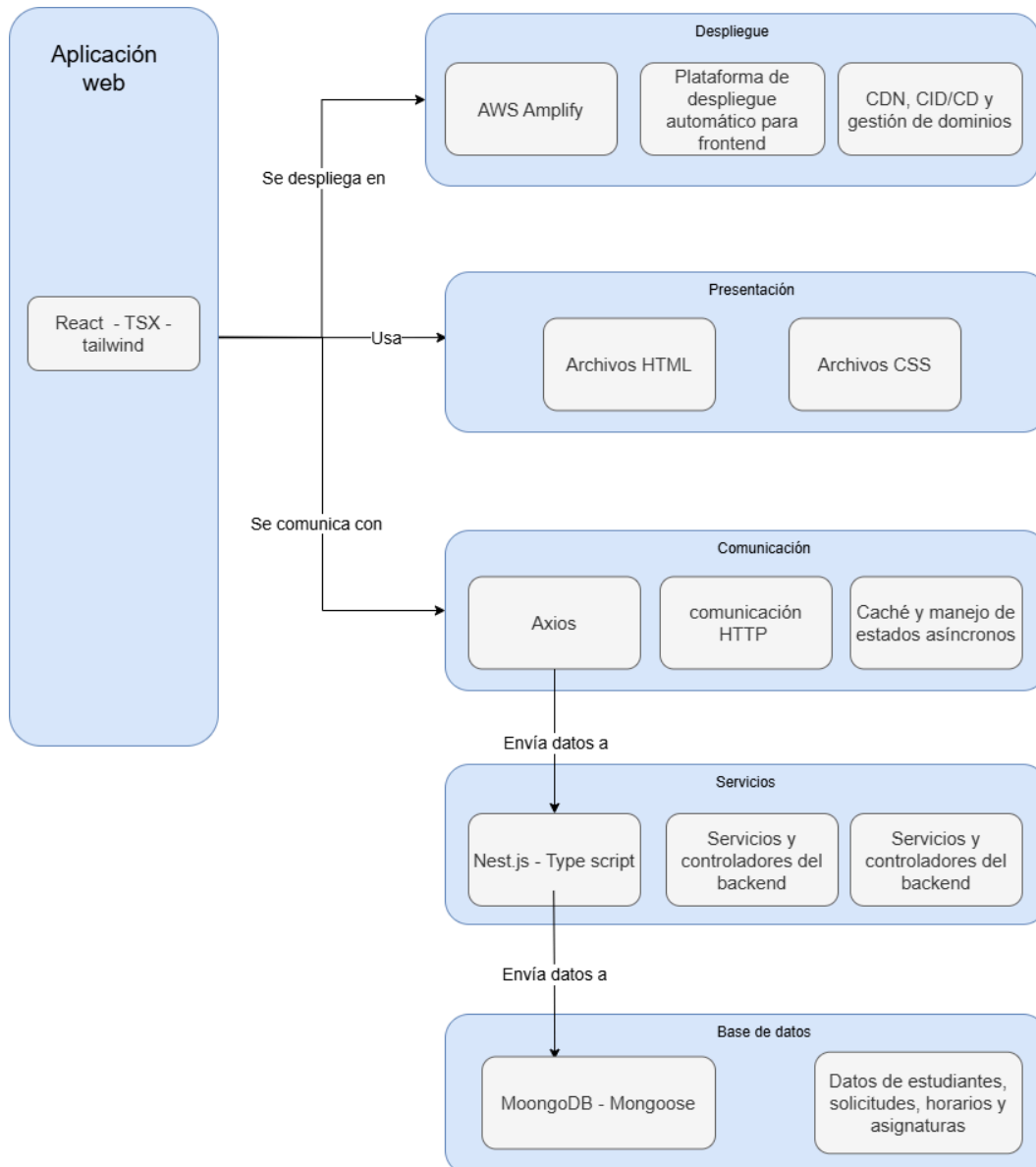


Tabla 24.*Componentes intrínsecos del sistema*

Componente	Descripción
<i>React (TSX y TAILWIND)</i>	<i>Archivos encargados de la interfaz gráfica (UI), estilos y navegación. Desarrollados en React 18+ con TypeScript y Tailwind.</i>
<i>Nest.js (TypeScript)</i>	<i>Servicios y controladores del backend que gestionan lógica de negocio, autenticación e integraciones. Escritos en TypeScript y basados en arquitectura modular.</i>
<i>MongoDB (Mongoose)</i>	<i>Base de datos NoSQL que almacena datos de estudiantes, solicitudes, horarios y asignaturas. Utiliza Mongoose como ODM para validación y consultas.</i>
<i>AWS Amplify</i>	<i>Panel administrativo para aprobar/rechazar solicitudes.</i>
<i>Axios</i>	<i>Plataforma de despliegue automático para el frontend, con integración de CDN, CI/CD y gestión de dominios.</i>
	<i>Bibliotecas para comunicación HTTP entre frontend y backend, incluyendo caché y manejo de estados asíncronos.</i>

4.3.2.2 Plan de integración.

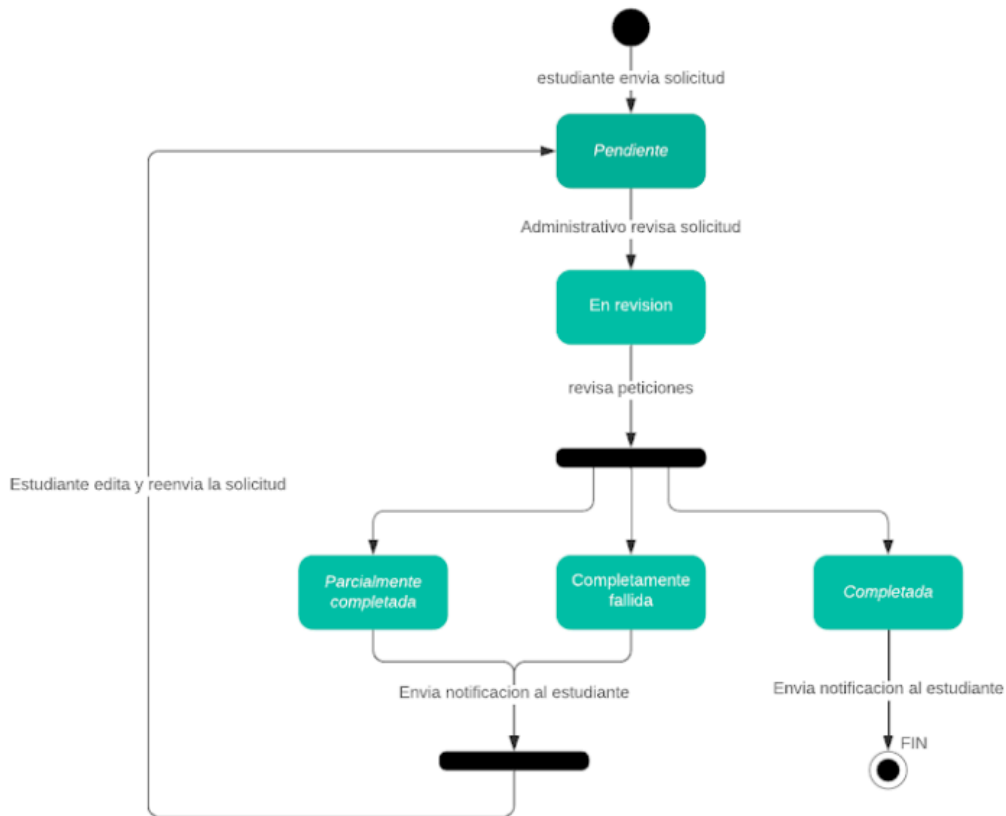
El propósito de este plan es definir el orden en que los componentes del sistema

deben llevarse a cabo, los resultados al integrar el sistema y cómo serán evaluados.

Para definir la primera instancia de un plan de integración, es preciso que a partir de los requerimientos funcionales, se establezcan los servicios que el sistema debe ofrecer, y en función de estos servicios el diseño de la estructura de la base de datos, y la interfaz gráfica que permitiría interactuar a los usuarios con el sistema.

Figura 20.

Diagrama de actividades: plan de integración



A continuación el sistema manejará los siguientes estados:

- **Pendiente:** Se asigna cuando el estudiante registra su solicitud y esta se encuentra en la cola de espera para ser atendida.
- **En revisión:** Este estado se aplica cuando el directivo ha iniciado el proceso de revisión de la solicitud.
- **Completada:** Indica que todas las peticiones de la solicitud han sido procesadas exitosamente.
- **Completamente fallida:** Se asigna cuando ninguna de las peticiones de la solicitud pudo ser procesada.
- **Parcialmente completada:** Este estado refleja que una o más peticiones de la solicitud fueron procesadas de manera satisfactoria.

4.3.2.3 Implementación de componentes.

Esta actividad abarca una parte fundamental de la implementación del sistema, permitiendo su distribución para la integración. A continuación, se detallan las herramientas utilizadas en la implementación de cada componente:

Tabla 25.

Software de implementación de componentes

Herramientas	Función
<i>React (TSX y TAILWIND)</i>	<i>Framework de frontend basado en componentes reutilizables con TypeScript y Tailwind CSS para el diseño.</i>
<i>Nest.js (TypeScript)</i>	<i>Servicios y controladores del backend que gestionan lógica de negocio, autenticación e integraciones. Escritos en TypeScript y basados en arquitectura modular.</i>
<i>MongoDB (Mongoose)</i>	<i>Base de datos NoSQL que almacena datos de estudiantes, solicitudes, horarios y asignaturas. Utiliza Mongoose como ODM para validación y consultas.</i>
<i>AWS Amplify</i>	<i>Plataforma de despliegue automático para el frontend, con integración de CDN, CI/CD y gestión de dominios.</i>
<i>Axios</i>	<i>Bibliotecas para comunicación HTTP entre frontend y backend, incluyendo caché y manejo de estados asíncronos.</i>
<i>Google Chrome / Mozilla Firefox</i>	<i>Navegadores web utilizados para pruebas y validación del frontend.</i>

4.3.2.3.1 Creación de base de datos.

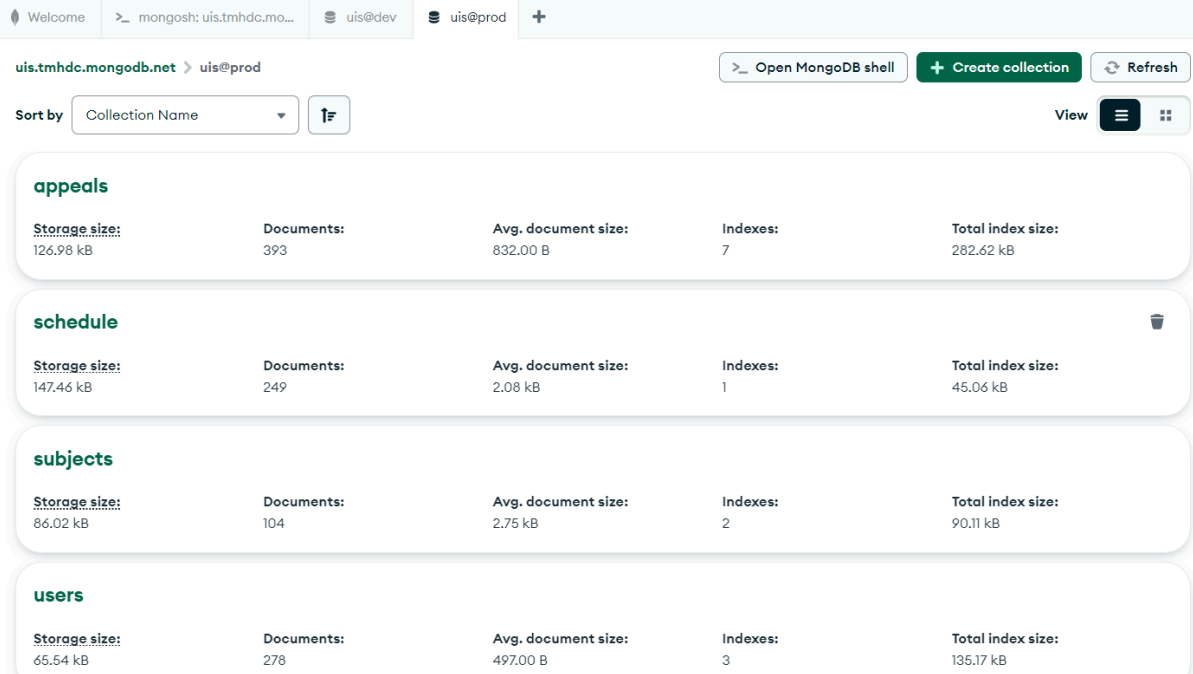
Dada la alta demanda de procesamiento y almacenamiento de información en el Sistema de Ajuste de Matrícula (SAM), se ha optado por la implementación de una base de datos NoSQL, seleccionando MongoDB como motor de almacenamiento. Esta elección responde a la necesidad de contar con una solución altamente escalable, flexible y optimizada para gestionar grandes volúmenes de datos con acceso eficiente y en tiempo real ya que MongoDB ofrece múltiples ventajas para el SAM, como Optimización en consultas y rendimiento: Su modelo de indexación avanzada y consultas optimizadas mejoran la velocidad de recuperación de datos, lo que se traduce en respuestas más rápidas para los usuarios al

consultar y procesar solicitudes de matrícula, además de facilitar el almacenamiento y consulta de información estructurada y semiestructurada sin depender de esquemas rígidos, permitiendo adaptaciones dinámicas a los cambios en los requerimientos del sistema.

Como parte del proceso de implementación, se diseñaron y generaron los esquemas de datos en MongoDB, asegurando una arquitectura eficiente y alineada con los requerimientos operativos del SAM. Esta infraestructura garantiza integridad, eficiencia y escalabilidad, optimizando la gestión de información académica y mejorando la experiencia de los usuarios del sistema.

Figura 21.

Esquemas de datos en MongoDB



Collection Name	Storage size	Documents	Avg. document size	Indexes	Total index size
appeals	126.98 kB	393	832.00 B	7	282.62 kB
schedule	147.46 kB	249	2.08 kB	1	45.06 kB
subjects	86.02 kB	104	2.75 kB	2	90.11 kB
users	65.54 kB	278	497.00 B	3	135.17 kB

4.3.2.3.2 Creación de interfaz gráfica.

Para el desarrollo de las interfaces del sistema, se empleó Shadcn/UI, una biblioteca de componentes de interfaz de usuario (UI) que permite crear aplicaciones web modernas,

funcionales y visualmente atractivas. Su implementación se basó en el diseño y esquema predefinido de la interfaz gráfica, garantizando el cumplimiento de los requerimientos funcionales y no funcionales, Se hizo especial énfasis en la experiencia del usuario (UX), priorizando la intuitividad, accesibilidad e interacción continua con el sistema. Además, se optimizó el diseño para garantizar una experiencia completamente responsiva, permitiendo que la aplicación se adapte de manera fluida a distintos dispositivos y resoluciones de pantalla, mejorando así la usabilidad en entornos móviles y de escritorio.

Figura 22.

Diseño responsive para móvil

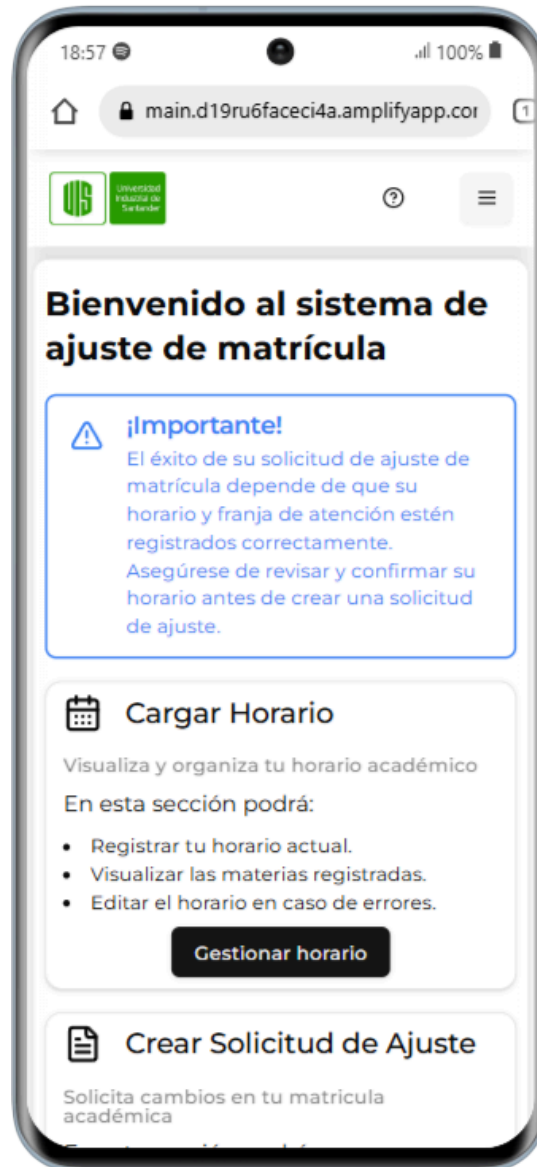
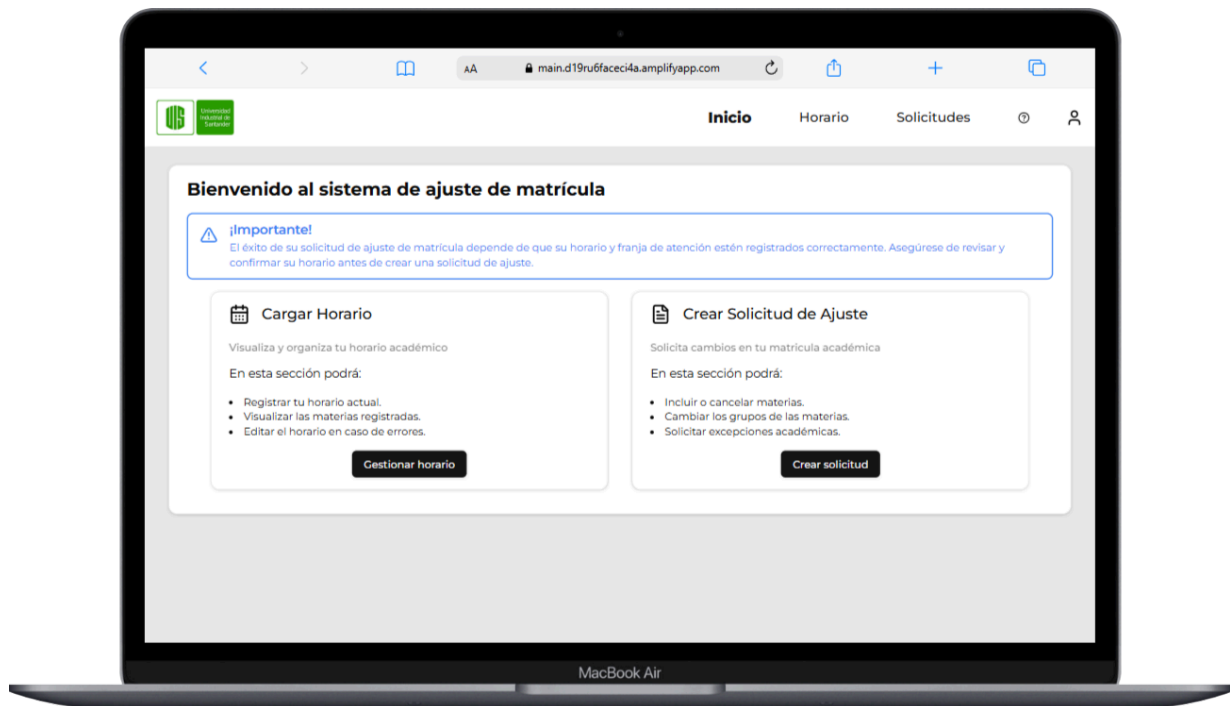


Figura 23.

Diseño responsive para laptop



4.3.2.3 Configuración de servicios.

La configuración de servicios del Sistema de Ajuste de Matrícula (SAM) se ha diseñado para satisfacer los requerimientos funcionales y estructurales del sistema, asegurando su correcto funcionamiento y eficiencia. Para ello, se ha implementado una infraestructura de software robusta que abarca elementos clave como la interfaz de usuario y la integridad de la base de datos y el control de acceso. A continuación, se detallan los aspectos más relevantes para mantener la seguridad de la información y los procesos del sistema mediante validaciones y buenas prácticas.

Interfaz de usuario

La interfaz de usuario ha sido diseñada de tal forma que que llevan al mínimo la posibilidad de errores por parte del usuario, entre los principales aspectos considerados se encuentran:

- Implementación de formularios asegurando la correcta entrada de datos mediante la restricción de campos numéricos, alfanuméricos, caracteres especiales, validación de correos, contraseñas y control de campos vacíos

Figura 24.

Validación de formularios

Registro de usuario
Crea una cuenta para acceder al sistema de ajuste de matrícula

? **Sistema de Ajuste de Matrícula (SAM)**
Sistema exclusivo para estudiantes de Ingeniería de Sistemas.

Nombre Apellido

Codi ! Completa este campo

Correo institucional

Contraseña 👁

Confirmar contraseña 👁

👤 Registrarse

¿Ya tienes una cuenta? Ingresar aquí

Registro fallido
Las contraseñas no coinciden

Registro fallido
El correo debe ser de la universidad UIS

Emisión de mensajes de orientación y alerta para los usuarios en caso de operaciones incorrectas y de operaciones realizadas con éxito.

Figura 25.

Toast de retroalimentación para el usuario

<p>Horario guardado El horario ha sido guardado exitosamente.</p>	<p>Requisitos no cumplidos La materia no cumple el requisito de Cálculo I</p>
<p>Solicitud guardada Los cambios han sido guardados exitosamente</p>	<p>Solicitud completada La solicitud ha sido marcada como completada</p>

Figura 26.

Alertdialog con resumen de movimientos

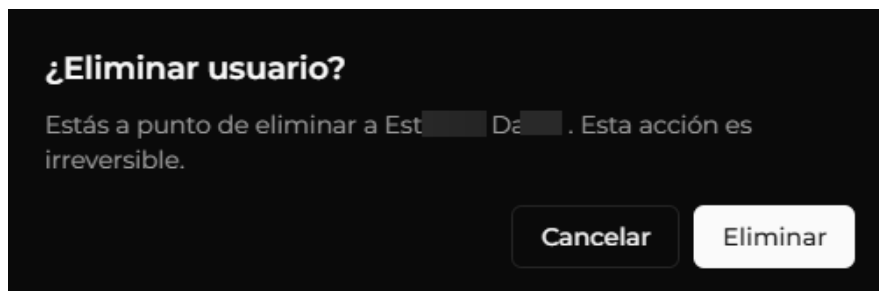


Presentación detallada en pantalla un resumen de las operaciones realizadas, incluyendo la opción de revertir los cambios.

Confirmación de Acción Irreversible para eliminación de usuarios.

Figura 27.

Confirmación acciones irreversibles



validación incremental evitando que un usuario de tipo admin o root puedan avanzar en las solicitudes si no han completado la anterior.

Integridad de la Base de Datos y Control de Acceso

Para garantizar la seguridad, confidencialidad e integridad de la información dentro del Sistema de Ajuste de Matrícula (SAM), se han implementado diversas medidas de protección alineadas con las mejores prácticas de seguridad definidas por la Open Web Application Security Project (OWASP).

Uno de los principales riesgos identificados por OWASP es la Pérdida de Control de Acceso (Broken Access Control), una vulnerabilidad crítica que puede permitir a atacantes obtener privilegios indebidos dentro del sistema. Para mitigar este riesgo, el SAM fue diseñado con un modelo de control de acceso robusto, implementando medidas como autenticación segura, validación estricta de permisos y encriptación avanzada.

- Control de Acceso Basado en Roles (RBAC - Role-Based Access Control) SAM implementa un modelo RBAC que restringe el acceso a funcionalidades y datos según el rol del usuario.

Estudiantes: Solo pueden gestionar y visualizar sus propias solicitudes de ajuste de matrícula.

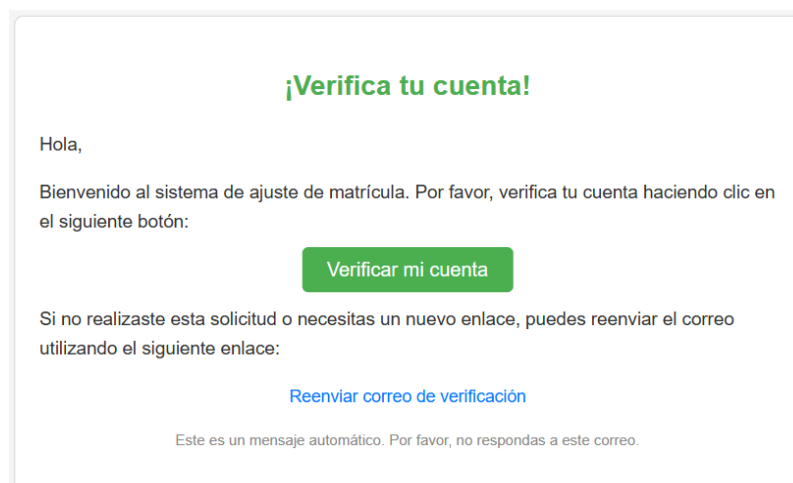
Administradores: Gestionan y procesan únicamente solicitudes que les fueron asignadas, el procesamiento es de forma secuencial y solo pueden visualizar el histórico de solicitudes procesadas únicamente por el mismo

Superadministrador (Root): Tiene acceso a la gestión de usuarios, puede crear usuarios de tipo admin y asignarles una contraseña por defecto, revisa todas las solicitudes registradas por los estudiantes, único usuario con acceso al módulo de estadísticas

- Autenticación Segura y Gestión de Sesiones; para prevenir accesos no autorizados, el Sistema de Ajuste de Matrícula (SAM) implementa autenticación mediante JSON Web Tokens (JWT). Esto garantiza que cada solicitud realizada por un usuario sea validada antes de permitir el acceso a los recursos protegidos. El registro de usuarios se lleva a cabo exclusivamente mediante correos electrónicos institucionales, los cuales son validados a través de un mensaje enviado al correo para confirmar la identidad del usuario y evitar suplantaciones. Además, los tokens tienen una vigencia limitada y deben ser renovados periódicamente para mantener una sesión activa sin exponer credenciales sensibles. Adicionalmente, el registro de las contraseñas de los usuarios se almacenan encriptadas utilizando bcrypt.

Figura 28 .

Autenticación de usuario por correo electrónico



4.3.2.4 Integración de componentes.

A medida que se desarrollaron y validaron los componentes mediante pruebas funcionales, estos fueron integrados progresivamente con el resto de las funcionalidades del sistema. Este proceso de integración continua permitió evaluar la interoperabilidad entre los módulos, asegurando la coherencia y estabilidad del sistema en cada iteración. Como resultado, se garantiza la integridad del producto final, cumpliendo con los requerimientos definidos y asegurando su correcto funcionamiento en el entorno operativo.

4.3.3 Pruebas alfa.

El objetivo de esta disciplina es evaluar la calidad del producto que se está desarrollando, mediante la aplicación de pruebas concretas, para validar que las suposiciones hechas en el diseño y los requerimientos se estén cumpliendo satisfactoriamente, esto permite verificar que el producto funcione como se diseñó y que los requerimientos son satisfechos cabalmente. Esta disciplina está presente en todo el ciclo de vida del desarrollo del sistema para ir refinando durante todo el proceso.

4.3.3.1 Plan de pruebas.

El plan de pruebas establece los niveles de verificación del sistema mediante la ejecución de distintos tipos de pruebas diseñadas para garantizar la calidad del software. Este enfoque abarca pruebas unitarias e integración, evaluando tanto la funcionalidad de los módulos individuales como su interoperabilidad. Además, se contemplan distintos criterios de validación, incluyendo usabilidad y fiabilidad, asegurando el cumplimiento de los requerimientos definidos. A continuación, se detallan estos aspectos en la siguiente tabla.

Tabla 26.*Plan de pruebas*

<i>Módulo</i>	<i>Características a testear</i>	<i>Tipos de prueba</i>
<i>Registro de usuarios</i>	<i>Ingreso de un correo diferente al institucional Ingreso de contraseñas débiles Registrar un usuario ya registrado Confirmación de contraseña diferente a la contraseña creada</i>	<i>Pruebas unitarias.</i>
<i>Login de Usuarios</i>	<i>Intentar ingresar al sistema con un usuario no autenticado Intentar ingresar al sistema con un usuario NO registrado validación de usuario y contraseña incorrecta</i>	<i>Pruebas unitarias.</i>
<i>Registro y manejo de solicitudes de matrícula</i>	<i>No se puede enviar una nueva solicitud hasta que la anterior haya sido revisada Se registran todas las peticiones de la solicitud La solicitud captura la información del estudiante y es mostrada a root y administrador Actualizar estado de solicitudes</i>	<i>Pruebas de integración.</i>
<i>Notificaciones por correo electrónico</i>	<i>Enviar correo de verificación de cuenta Envío de correo recuperación de contraseña Notificación cambio de estado de la solicitud Conflicto de horario Requisitos entre asignaturas Asignaturas cursadas</i>	<i>Pruebas de integración. Pruebas unitarias.</i>
<i>Validaciones de asignaturas</i>	<i>Registro de nuevos usuarios Validación para no permitir usuarios duplicados Asignación de permisos de acuerdo al rol creado</i>	<i>Pruebas unitarias.</i>
<i>Crud de usuarios</i>		

4.3.3.2 Pruebas planeadas.

A continuación se realiza un resumen de las pruebas aplicadas y los resultados obtenidos en cada una de ellas

4.3.3.2.1 Pruebas de calidad.

Las pruebas fueron ejecutadas por los miembros del equipo de desarrollo con la supervisión y aprobación del tutor y el director del proyecto. Estas pruebas se llevaron a cabo de manera progresiva, en paralelo con la integración de los distintos componentes del sistema, permitiendo evaluar aspectos clave como la correcta ejecución de los procesos, la interacción con la interfaz, el rendimiento del sistema y la precisión en el procesamiento de la información

Pruebas de Diseño de Interfaz

Se llevaron a cabo pruebas manuales aplicando el enfoque de pruebas de caja negra, con el objetivo de evaluar el comportamiento del sistema sin necesidad de analizar su código interno. Estas pruebas permitieron validar aspectos clave como la navegación y usabilidad del sistema, la correcta adaptación a diferentes dispositivos (responsividad), el rendimiento en diversas condiciones de uso y la coherencia en el diseño visual, asegurando la consistencia de elementos gráficos como colores, tipografías y estilos. A continuación se detallan los escenarios

Tabla 27.

Pruebas diseño e interfaz

Resultados obtenidos Correcciones realizadas

<i>El texto de las asignaturas no era visible debido a la falta de contraste entre el color de fondo y el de la fuente.</i>	<i>Se ajustaron los estilos CSS para mejorar la legibilidad, cambiando el color de fuente y asegurando un contraste adecuado</i>
<i>La asignatura eliminada seguía ocupando un espacio visual, lo que impedía registrar asignaturas en ese horario.</i>	<i>Se configuró la función que se activaba al dar clic en el icono de la papelera para que la vista de horario se renderiza automáticamente</i>
<i>Algunos elementos de la interfaz se desalinearon en dispositivos móviles.</i>	<i>Se mejoró la responsividad aplicando grid y flexbox para la distribución de los componentes en diferentes tamaños de ventanas</i>
<i>El botón "Enviar al director" no había sido desarrollado</i>	<i>Se realizó la lógica en el backend y se implementó en el frontend para poder ser utilizado por el usuario</i>

Pruebas de carga y estrés

Es fundamental definir los límites de capacidad del servidor para gestionar de manera eficiente la alta concurrencia de solicitudes durante el proceso de ajuste de matrícula. Esto permite mejorar el rendimiento del sistema, evitar la degradación del servicio y reducir el riesgo de fallos o interrupciones en momentos de máxima demanda. Para mitigar estos impactos, se ha implementado el backend con cinco instancias paralelas en Railway, lo que permite distribuir la carga de trabajo de manera equilibrada garantizando la disponibilidad continua del sistema en escenarios de alta concurrencia.

Pruebas funcionales

Se ejecutaron en cada uno de los módulos con el fin de asegurar que cada característica del software funcione de acuerdo con los requisitos y especificaciones definidos.

Tabla 28.

Pruebas funcionales

Resultados obtenidos	Correcciones realizadas
<i>La asignación de solicitudes presentaba retardos significativos, afectando la eficiencia del proceso.</i>	<i>Se realizó el despliegue del backend en 5 instancias para mejorar los tiempos de respuesta del sistema</i>
<i>Al aplicar filtros en las tablas, los datos no se actualizaban correctamente</i>	<i>Se corrigió la lógica de filtrado en las consultas a la base de datos y se optimizó la manipulación de estados en el frontend para asegurar una actualización precisa.</i>
<i>El cálculo del tiempo promedio de respuesta calculaba valores errados debido a fallas en la segmentación por franjas horarias.</i>	<i>Se ajustó la fórmula de cálculo considerando el tiempo a partir en que la solicitud se encuentra en su franja hasta cuando es atendida</i>

Prueba de Integridad y Seguridad de los Datos

Esta prueba permite verificar el nivel de seguridad de la aplicación, en cuanto al manejo de la información almacenada, a través de la comprobación de los controles de acceso y permisos para cada usuario y a la base de datos. Para la realización de estas pruebas se utilizan datos que permitan la validación de los usuarios, controlar las sesiones y los privilegios de cada usuario.

Tabla 29.

Pruebas de integridad y seguridad de los datos

Resultados obtenidos	Correcciones realizadas
<i>El usuario estudiante podía ver las solicitudes de otros estudiantes</i>	<i>Se modifica la consulta a la base de datos para que solo recuperen solicitudes asociadas al usuario autenticado.</i>

4.4 Fase de transición

Tiene como objetivo general entregar el producto funcional en manos de los usuarios finales una vez realizadas las pruebas por un grupo especial de usuarios, para lo que se requerirá desarrollar nuevas versiones actualizadas del producto, entrenar a los usuarios en el manejo del sistema, completar la documentación, y en general tareas relacionadas con la configuración, instalación y usabilidad del producto.

Los objetivos específicos de esta fase son:

- Garantizar que el usuario aprenda a operar y mantener el sistema.
- Conseguir un producto final que cumpla los requerimientos esperados.

El hito en la fase de transición corresponde a haber decidido si los objetivos se cumplieron y el comienzo de otro ciclo de desarrollo. El cliente debe haber revisado y aceptado los artefactos que le han sido entregados.

4.4.1 Implantación.

Esta disciplina tiene como objetivo distribuir e instalar con éxito el sistema elaborado por el equipo de desarrollo y asegurar la disponibilidad del producto para los usuarios finales.

4.4.1.1 Plan de implantación.

El plan de implantación se estructura en 5 fases críticas, diseñadas para minimizar riesgos y asegurar una adopción efectiva del sistema:

Figura 29.

Plan de implantación

FASES PLAN DE IMPLANTACIÓN		SEMANA 1					SEMANA 2					
Preparación	Configuración entorno AWS	■	■									
	Migración de Datos iniciales			■	■							
Despliegue	Implementación del Backend				■	■						
	Despliegue del Frontend						■					
Pruebas	Pruebas de Carga y Seguridad						■	■				
	Validación con Stakeholders								■			
Capacitación	Taller para Administrativos								■			
	Videotutorial para Estudiantes									■		
Lanzamiento	Lanzamiento y Monitoreo Inicial									■	■	■

Preparación:

- Configuración de la infraestructura en AWS (Amplify para frontend).
- Migración de datos de asignaturas programadas (Sistema UIS) a MongoDB.

Despliegue:

- Implementación del backend en 5 instancias paralelas de Railway.
- Despliegue automático del frontend en AWS Amplify con integración de CDN.

Pruebas:

- Simulación de 100 usuarios concurrentes para validar rendimiento.
- Auditoría de seguridad.

Capacitación:

- Talleres prácticos para personal administrativo (aprobación de solicitudes, envío de solicitudes al director).
- Videotutorial como guía interactiva para estudiantes, administrativos y directores de escuela (creación de solicitudes, visualización de horarios, gestión de solicitudes). Anexo A y B.

Lanzamiento:

- Liberación gradual del sistema a los estudiantes para monitorear impacto.

- Monitoreo constante durante los días asignados al proceso de ajuste de matrícula

4.4.1.2 Desarrollo de material de apoyo.

En esta actividad se produjo el material de apoyo necesario para que los usuarios utilicen correctamente el sistema. Para este proyecto se hace referencia a documentos de vital importancia como son:

Manual de Usuario: Videotutorial explicando el uso completo del sistema para los estudiantes, administrativos y directores de escuela, al cual pueden acceder desde el sistema en la barra de navegación.

4.4.1.3 Pruebas beta.

Una vez realizadas las pruebas alfa se asegura que el sistema ha alcanzado una versión beta o una versión final preliminar, considerada como la primera iteración completamente operativa del producto. Sin embargo para garantizar la calidad del producto y estabilidad, se implementa un plan de pruebas beta. Estas pruebas permiten una evaluación exhaustiva del sistema en un entorno más representativo del uso real, identificando posibles ajustes y optimizaciones necesarias para mejorar la experiencia del usuario y alinear la funcionalidad del software con sus expectativas y requerimientos. A continuación se muestran los casos de prueba realizados en los diferentes módulos del sistema.

Tabla 30.

Casos de prueba del registro de usuarios

<i>CP-01</i>	<i>Paso</i>	<i>Registrar usuario</i>
--------------	-------------	--------------------------

<i>Precondición</i>	<ul style="list-style-type: none"> ● <i>El Usuario debe estar en la página de registro</i> ● <i>Debe tener acceso al correo institucional</i>
<i>Descripción</i>	<i>Este caso de prueba verifica que un nuevo usuario pueda registrarse en el sistema</i>
<i>Secuencia normal</i>	
0	<i>El usuario accede a la página de registro.</i>
1	<i>Ingresa los datos requeridos (nombre, correo electrónico, contraseña, confirmación de contraseña)</i>
2	<i>Presiona el botón de "Registrarse".</i>
<i>Postcondición</i>	<ul style="list-style-type: none"> ● <i>Se crea correctamente la cuenta del usuario.</i> ● <i>Se muestra un mensaje de confirmación.</i> ● <i>Se redirige al usuario a la pantalla de inicio de sesión o al dashboard inicial.</i>
<i>Excepciones</i>	<ul style="list-style-type: none"> ● <i>Correo electrónico ya registrado → Se muestra un mensaje de error.</i> ● <i>Contraseña y confirmación no coinciden → Se notifica al usuario.</i> ● <i>Formato de correo inválido → Se solicita ingresar un correo válido.</i> ● <i>Campos obligatorios vacíos → Se indica que todos los campos deben completarse.</i>

Tabla 31.

Casos de prueba del inicio de sesión

CP-02	Paso	Iniciar Sesión
<i>Precondición</i>	<ul style="list-style-type: none"> ● <i>El usuario debe estar registrado en el sistema.</i> ● <i>Debe tener credenciales válidas (correo electrónico y contraseña).</i> 	
<i>Descripción</i>	<i>Este caso de prueba verifica que un usuario registrado puede acceder al sistema ingresando sus credenciales correctamente.</i>	
<i>Secuencia normal</i>		
0	<i>El usuario accede a la página de inicio de sesión.</i>	
1	<i>Ingresa su correo electrónico y contraseña en los campos correspondientes.</i>	
2	<i>Presiona el botón de "Iniciar sesión".</i>	
3	<i>Si las credenciales son correctas, el usuario es redirigido a la vista principal</i>	
<i>Postcondición</i>	<ul style="list-style-type: none"> ● <i>El usuario accede exitosamente a su cuenta.</i> ● <i>Se muestra la pantalla principal o dashboard con la información correspondiente</i> 	
<i>Excepciones</i>	<ul style="list-style-type: none"> ● <i>Credenciales incorrectas</i> ● <i>Usuario no registrado</i> ● <i>Formato de correo inválido</i> ● <i>Campos vacíos</i> ● <i>Cuenta no autenticada</i> 	

Tabla 32.*Casos de prueba del registro de horario*

CP-03	Paso	Registrar horario académico
<i>Precondición</i>		<i>El usuario debe ser de tipo estudiante el cual le da permisos para registrar horarios</i>
<i>Descripción</i>		<i>Este caso de prueba verifica que un usuario autorizado pueda registrar un horario académico asignando materias, docentes, días y horarios correspondientes.</i>
<i>Secuencia normal</i>		
	0	<i>El usuario accede al módulo de "Registrar horario".</i>
	1	<i>Selecciona la franja horaria</i>
	2	<i>Selecciona las materias registradas en su horario .</i>
	3	<i>Presiona el botón "Guardar" para registrar el horario.</i>
	4	<i>Da clic en "aceptar" en el modal con el resumen de asignaturas y el horario queda registrado en el sistema.</i>
<i>Postcondición</i>		<ul style="list-style-type: none"> ● <i>El horario académico se registra correctamente y es visible en el sistema.</i> ● <i>Los estudiantes pueden visualizar el horario asignado.</i>
<i>Excepciones</i>		<ul style="list-style-type: none"> ● <i>Credenciales incorrectas</i> ● <i>Usuario no registrado</i> ● <i>Formato de correo inválido</i> ● <i>Campos vacíos</i>

Tabla 33.*Casos de prueba del registro de solicitudes*

CP-04	Paso	Registrar Solicitud de Ajuste de Matrícula
<i>Precondición</i>		<ul style="list-style-type: none"> ● <i>El usuario debe estar autenticado en el sistema como estudiante.</i> ● <i>No debe haber una solicitud de ajuste pendiente de revisión por el administrador o root.</i>
<i>Descripción</i>		<i>Este caso de prueba verifica que un estudiante pueda registrar una solicitud de ajuste de matrícula, asegurando que no se creen múltiples solicitudes sin revisión previa.</i>
<i>Secuencia normal</i>		
	0	<i>El usuario accede al módulo de "Registrar solicitud".</i>
	1	<i>Selecciona el tipo de ajuste (agregar, eliminar o cambiar asignaturas).</i>
	2	<i>Ingresa los detalles del ajuste en caso de ser necesario</i>
	3	<i>Presiona el botón "Enviar".</i>
	4	<i>Se muestra un mensaje de confirmación indicando que la solicitud ha sido enviada para revisión</i>
<i>Postcondición</i>		<ul style="list-style-type: none"> ● <i>La solicitud de ajuste de matrícula queda registrada con estado "Pendiente".</i> ● <i>El estudiante no puede generar otra solicitud hasta que la actual sea revisada por el administrador o root.</i>
<i>Excepciones</i>		<ul style="list-style-type: none"> ● <i>El administrador recibe la solicitud en la vista de solicitudes.</i>
		<i>N/A</i>

Tabla 34.*Casos de prueba de la gestión de solicitudes del admin*

CP-05	Paso	Gestión de Solicitudes por el administrador
<i>Precondición</i>	<ul style="list-style-type: none"> ● El usuario debe estar autenticado como Administrador en el sistema. ● Deben existir solicitudes pendientes de revisión. 	
<i>Descripción</i>		<i>Este caso de prueba verifica que el administrador pueda gestionar las solicitudes de ajuste de matrícula en orden secuencial, asegurando que una solicitud no se marque como completada hasta resolver todas las peticiones asociadas. Además, el administrador debe poder enviar la solicitud al director en caso de no poder resolverla..</i>
<i>Secuencia normal</i>	<p>0</p> <p>1</p> <p>2</p> <p>3</p> <p>4</p> <p>5</p>	<p><i>El usuario accede al módulo de "Solicitudes".</i></p> <p><i>Se muestra una única solicitud basado en el orden de prioridad definido</i></p> <p><i>El administrador revisa las peticiones dentro de la solicitud</i></p> <p><i>Si puede resolver todas las peticiones, realiza las modificaciones correspondientes en el sistema de la universidad</i></p> <p><i>Una vez todas las peticiones de la solicitud han sido resueltas, presiona "Marcar como Completada".</i></p> <p><i>Se muestra un mensaje de confirmación y la solicitud cambia de estado a "Completada".</i></p>
<i>Postcondición</i>		<ul style="list-style-type: none"> ● <i>La solicitud es gestionada correctamente y queda registrada en el sistema como "Completada".</i> ● <i>Si el administrador no puede resolver una solicitud, la envía al director para su revisión.</i> ● <i>si no se puede gestionar ninguna solicitud queda registrada en el sistema como "Completamente fallida"</i> ● <i>Otros usuarios ya no pueden modificar la solicitud una vez resuelta.</i>
<i>Excepciones</i>		<ul style="list-style-type: none"> ● <i>El botón "Marcar como completada" estará inhabilitado en caso de que el administrador intente marcar la solicitud como completada sin resolver todas las peticiones</i>

Tabla 35.*Casos de prueba de la gestión de solicitudes del root*

CP-06	Paso	Gestión de Solicitudes por el Usuario Root
<i>Precondición</i>	<ul style="list-style-type: none"> ● El usuario debe estar autenticado como root en el sistema. ● Deben existir solicitudes pendientes de revisión. 	
<i>Descripción</i>		<i>Este caso de prueba verifica que el usuario Root pueda gestionar cualquier solicitud en el orden que desee, asegurando que dos usuarios no revisen la misma solicitud al mismo tiempo.</i>
<i>Secuencia normal</i>	<p>0 El usuario accede al módulo de "Solicitudes".</p> <p>1 Se muestra una lista de todas las solicitudes.</p> <p>2 El usuario selecciona una solicitud para revisar.</p> <p>3 Root revisa las peticiones dentro de la solicitud.</p> <p>4 Una vez todas las peticiones de la solicitud han sido resueltas, presiona "Marcar como Completada".</p> <p>5 Se muestra un mensaje de confirmación y la solicitud cambia de estado a "Completada".</p>	
<i>Postcondición</i>	<ul style="list-style-type: none"> ● La solicitud es gestionada correctamente y queda registrada en el sistema como "Completada". ● Otros usuarios ya no pueden modificar la solicitud una vez resuelta. ● si no se puede gestionar ninguna solicitud queda registrada en el sistema como "Completamente fallida" 	
<i>Excepciones</i>	<ul style="list-style-type: none"> ● Si un Administrador ya está revisando la solicitud, Root no podrá ingresar a editarla. ● Si intenta marcar la solicitud como completada sin resolver todas las peticiones, el sistema muestra una advertencia. 	

Tabla 36.*Casos de prueba del CRUD de usuarios*

CP-07	Paso	CRUD de Usuarios
<i>Precondición</i>	<ul style="list-style-type: none"> ● El usuario debe estar autenticado como Root en el sistema. ● Deben existir usuarios registrados en la base de datos. 	
<i>Descripción</i>		<i>Este caso de prueba verifica que el usuario Root pueda gestionar completamente el CRUD de usuarios (crear, leer, actualizar y eliminar), asegurando que los cambios se reflejen correctamente en el sistema y que se mantenga la integridad de los datos.</i>
<i>Secuencia normal 0</i>		<i>El usuario Root accede al módulo de gestión de usuarios.</i>
	1	<i>Root selecciona la opción de crear un nuevo usuario e ingresa los datos requeridos.</i>
	2	<i>Root selecciona un usuario existente y elige la opción de editar. Modifica los datos y guarda los cambios..</i>
	3	<i>Root accede a la lista de usuarios y selecciona un usuario para eliminarlo.</i>
	4	<i>Root confirma la eliminación del usuario..</i>
	5	<i>Root intenta acceder a los datos del usuario eliminado.</i>
<i>Postcondición</i>	<ul style="list-style-type: none"> ● Los usuarios creados, editados o eliminados quedan correctamente registrados en la base de datos. ● No se permite la duplicación de usuarios con los mismos datos críticos (ej. correo electrónico o ID). ● Los cambios realizados son reflejados en tiempo real en la interfaz del sistema. ● Si un usuario es eliminado, no podrá autenticarse en el sistema ni aparecer en la lista de usuarios. 	
<i>Excepciones</i>	<ul style="list-style-type: none"> ● Si Root intenta crear un usuario con un correo ya registrado el sistema muestra un mensaje de error indicando la duplicidad. ● Si Root deja campos obligatorios vacíos al crear o editar un usuario, el sistema impide guardar los cambios y muestra una advertencia. ● Si Root intenta eliminar un usuario que está activo en una sesión. El sistema bloquea la eliminación y muestra un mensaje de error. 	

Tabla 37.*Casos de prueba de las estadísticas*

CP-08	Paso	Módulo de Estadísticas
<i>Precondición</i>	<ul style="list-style-type: none"> ● El usuario debe estar autenticado como Root en el sistema. ● Deben existir registros de solicitudes en la base de datos. ● El módulo de estadísticas debe estar correctamente integrado con la base de datos. 	
<i>Descripción</i>		<i>Este caso de prueba verifica que el usuario Root pueda acceder y visualizar correctamente las estadísticas del sistema, incluyendo la cantidad de solicitudes, el tiempo promedio de respuesta, la cantidad de solicitudes sin atender y el gráfico de cantidad de solicitudes por hora.</i>
<i>Secuencia normal</i>	<p>0</p> <p>1</p> <p>2</p> <p>3</p> <p>4</p>	<p><i>Root visualiza el número total de solicitudes registradas en el sistema, se muestra la cantidad exacta de solicitudes en tiempo real.</i></p> <p><i>Root visualiza el número total de solicitudes registradas en el sistema, se muestra la cantidad exacta de solicitudes en tiempo real.</i></p> <p><i>Root consulta el tiempo promedio de respuesta de las solicitudes, se muestra el cálculo del tiempo promedio en función de las solicitudes resueltas.</i></p> <p><i>Root visualiza la cantidad de solicitudes sin atender, se muestra el número de solicitudes que aún no han sido gestionadas.</i></p> <p><i>Root accede al gráfico de cantidad de solicitudes por hora, se genera una visualización gráfica de las solicitudes procesadas en cada franja horaria.</i></p>
<i>Postcondición</i>	<ul style="list-style-type: none"> ● Las estadísticas reflejan datos actualizados y coherentes con la base de datos. ● La interfaz gráfica debe presentar la información de manera clara y comprensible. 	
<i>Excepciones</i>	<i>N/A</i>	

4.4.1.4 Seguimiento post-instalación.

El seguimiento post-instalación del Sistema de Ajuste de Matrícula (SAM) se enfoca en garantizar su adaptación continua a los cambios normativos, técnicos y operativos propios del entorno universitario. Este proceso se articula mediante mecanismos de actualización, soporte técnico y retroalimentación, asegurando que el sistema evolucione de manera sostenible y alineada con las necesidades de la Escuela de Ingeniería de Sistemas e Informática.

La actualización contractual y normativa es un eje prioritario. Dado que el reglamento académico (Artículo 43) y las convenciones colectivas del sector pueden sufrir modificaciones. Por esto se realizan auditorías semestrales manuales para verificar el cumplimiento de nuevas disposiciones, priorizando la actualización de algoritmos críticos, como los de priorización de solicitudes o detección de choques horarios.

La documentación jugará un rol central en este seguimiento. El Manual de Usuario incluirá instrucciones detalladas para estudiantes, administrativos y directores de escuela sobre el manejo completo del sistema. Cada nueva versión incorporará capturas de pantalla actualizadas, tutoriales paso a paso y un log de cambios que detalla las mejoras técnicas, como la corrección de errores o la integración con nuevos servicios. Por ejemplo, si se añade una función para exportar horarios, el manual explicará su uso con ejemplos visuales.

Finalmente, la retroalimentación continua se obtendrá mediante encuestas semestrales dirigidas a estudiantes. Estas encuestas medirán aspectos como la usabilidad de la interfaz, la claridad de las notificaciones y el tiempo de respuesta del sistema. Los resultados se analizarán para monitorear métricas clave como el número de solicitudes procesadas o el porcentaje de aprobaciones. Cualquier tendencia negativa, como un aumento inusual en solicitudes rechazadas, activará alertas proactivas para investigar y resolver causas subyacentes.

4.4.2 Análisis de costos y beneficios.

El desarrollo de cualquier proyecto involucra recursos humanos, económicos y materiales a fin de satisfacer el logro de los objetivos planteados. En este sentido se hace una inversión, una vez concluido el proceso de desarrollo debe realizarse el análisis de costos y beneficios,

identificando todos los costos y beneficios tanto tangibles como intangibles, así como realizar un análisis del retorno de la inversión.

4.4.2.1 Costos.

Para el desarrollo de la aplicación Web los autores realizaron una inversión económica producto de los diversos recursos técnicos y humanos involucrados.

Según lo especificado por la Unidad de Apoyo a Gestión de Proyectos La Vicerrectoría de Investigación y Extensión el presupuesto se establece considerando las modalidades actuales de trabajo y los recursos disponibles.

Los costos asociados a los programadores fueron obtenidos de la plataforma Glassdoor 2024, la cual provee estimaciones salariales basadas en datos reportados por empleados y empleadores. Esta información permitió establecer un rango referencial para el análisis de viabilidad económica del proyecto.

Estos costos se describen a continuación y conforman el monto total que implicó el desarrollo del sistema:

Costos de Hardware

Se refiere al costo de los equipos necesarios para el desarrollo e implantación del sistema.

Para este proyecto se utilizaron los siguientes componentes:

Tabla 38.

Costo de hardware: proyecto

<i>Componentes</i>	<i>Cantidad</i>	<i>Valor unitario</i>	<i>Total en Cop.</i>
<i>Computador Personal</i>	<i>3</i>	<i>\$ 3.500.000</i>	<i>\$ 10.500.000</i>

Costos de Software y Servicios Tecnológicos

Son los gastos que se realizan para el desarrollo y correcto funcionamiento del sistema, en este caso se realizaron gastos en el alojamiento en la nube del proyecto en Railway, así como también el internet usado para el desarrollo del software y el costo de la plataforma para el envío de correos a los usuarios.

Estos fueron los únicos costos asociados al software ya que la base de datos, desarrollo del cliente web y demás fueron bajo licencia de software libre.

Tabla 39.

Costo de software: proyecto

<i>Componentes</i>	<i>Cantidad (meses)</i>	<i>Valor unitario</i>	<i>Total en Cop.</i>
<i>Alojamiento en la nube</i>	<i>1</i>	<i>\$ 20.000</i>	<i>\$ 20.000</i>
<i>Servicio de internet</i>	<i>3</i>	<i>\$ 70.000</i>	<i>\$ 210.000</i>
<i>Plataforma de envío de correos</i>	<i>1</i>	<i>\$ 60.000</i>	<i>\$ 60.000</i>
<i>Total Costos</i>			<i>\$ 290.000</i>

Costo Talento Humano

Está relacionado con la remuneración percibida por las personas involucradas en el análisis y desarrollo del sistema.

Tabla 40.

Costo de talento humano: proyecto

<i>Concepto</i>	<i>Cantidad (horas)</i>	<i>Valor unitario</i>	<i>Total en Cop.</i>
<i>Director del Proyecto</i>	<i>34</i>	<i>\$ 305.000/hora</i>	<i>\$ 10.370.000</i>
<i>Autor 1</i>	<i>170</i>	<i>\$ 16.000/hora</i>	<i>\$ 2.720.000</i>
<i>Autor 2</i>	<i>170</i>	<i>\$ 16.000/hora</i>	<i>\$ 2.720.000</i>

<i>Autor 3</i>	<i>170</i>	<i>\$ 16.000/hora</i>	<i>\$ 2.720.000</i>
<i>Total Costos</i>			<i>\$ 18.530.000</i>

Gastos Adicionales

Corresponde a los costos generados por insumos y gastos imprevistos relacionados con el proyecto.

Tabla 41.

Costo de imprevistos: proyecto

<i>Concepto</i>	<i>Total en Cop.</i>
<i>Insumos e imprevistos</i>	<i>\$ 130.000</i>

Costo Total de Desarrollo

El costo total del proyecto se calcula sumando todos los componentes anteriores:

Tabla 42.

Costo total: proyecto

<i>Concepto</i>	<i>Total en Cop.</i>
<i>Recursos Humanos</i>	<i>\$ 18.530.000</i>
<i>Recursos Tecnológicos</i>	<i>\$ 290.000</i>
<i>Gastos Adicionales</i>	<i>\$ 130.000</i>
<i>Total Proyecto</i>	<i>\$ 18.950.000</i>

4.4.2.2 Estudio de mercado.

Para establecer un punto de comparación del sistema de gestión de solicitudes de ajuste de matrícula desarrollado, se realizó una investigación de mercado sobre soluciones similares disponibles actualmente. Se analizaron diferentes plataformas especializadas en la gestión de tickets y solicitudes, con énfasis en aquellas que son utilizadas en entornos académicos.

A continuación se presentan los productos más relevantes que ofrecen funcionalidades similares a las propuestas en este proyecto

Tabla 43.

Estudio de mercado

Producto	Características	Precio (Primer Año)
<i>Freshdesk</i>	<ul style="list-style-type: none"> ● <i>Sistema de tickets personalizable</i> ● <i>Automatización de flujos</i> ● <i>Integración con correo</i> 	<i>\$ 38.000.000</i>
<i>Zendesk for Education</i>	<ul style="list-style-type: none"> ● <i>Portal estudiantil personalizado</i> ● <i>Gestión de solicitudes académicas</i> ● <i>Automatización de procesos</i> ● <i>Integración con sistemas universitarios</i> ● <i>Analytics avanzados</i> 	<i>\$ 44.500.000</i>
<i>SysAid Education</i>	<ul style="list-style-type: none"> ● <i>Diseñado para instituciones educativas</i> ● <i>Help desk completo</i> ● <i>Flujos académicos predefinidos</i> ● <i>Reportes personalizados</i> 	<i>\$ 36.000.000</i>

Análisis Comparativo

La evaluación de estas soluciones comerciales frente a nuestro desarrollo nos permite establecer varios puntos importantes:

1. Costo Promedio del Mercado:

- El costo promedio de las soluciones comerciales es de \$39.500.000 COP
- Nuestro desarrollo tiene un costo de \$18.950.000 COP
- Representa un ahorro promedio de \$20.550.000 COP (52.02%)

2. Ventajas Económicas:

- Eliminación de costos recurrentes de licenciamiento
- No hay cobros por usuario adicional

- Control total sobre costos de mantenimiento
 - Inversión única vs. pagos anuales
- 3. Beneficios Adicionales del Desarrollo Propio:**
- Personalización completa para procesos institucionales específicos
 - Integración directa con sistemas existentes
 - Independencia de proveedores externos
 - Flexibilidad para modificaciones futuras
 - Propiedad intelectual del desarrollo

Conclusión del Estudio de Mercado

El análisis demuestra que nuestro desarrollo representa una alternativa económicamente viable frente a las soluciones comerciales disponibles. Si bien las opciones del mercado dan características robustas y probadas, nuestro sistema provee una solución más específica y adaptada a las necesidades institucionales a un costo considerablemente menor.

La diferencia en costos, sumada a los beneficios de tener un sistema totalmente personalizado y sin costos recurrentes de licenciamiento, justifica la decisión del desarrollo propio. Además, la posibilidad de realizar modificaciones y mejoras sin depender de terceros representa una ventaja estratégica importante para la institución.

4.4.2.3 Beneficios de la implantación del sistema.

Estos beneficios están asociados a la rentabilidad económica y al incremento de la eficiencia, producto de la implementación del sistema en comparación con el proceso manual actual de gestión de solicitudes académicas.

Beneficios tangibles

La razón principal para llevar a cabo el proyecto fue reducir significativamente los tiempos de atención y espera de las solicitudes de los estudiantes. Para evaluar si se logró este

objetivo, se registraron los tiempos de atención a través de Zoom para 40 solicitudes, obteniendo los siguientes resultados.

Tabla 44.

Tiempo de atención zoom vs sam

Modelo de atención por zoom		
TIEMPO PROMEDIO DE RESPUESTA [mins]	TIEMPO MÁXIMO [mins]	TIEMPO MÍNIMO [mins]
2:28	8:30	0:40
Modelo de atención por SAM		
0:40	1:10	0:15
Porcentaje de reducción		
73%	86.27%	62.5%

Según los datos presentados en la tabla anterior, se observa una reducción del 73% en el tiempo promedio de respuesta al implementar el Sistema de Ajuste de Matrícula (SAM). Este indicador representa el intervalo desde que una solicitud comienza a ser revisada por un administrador hasta su procesamiento inicial. En el modelo basado en Zoom, dicho tiempo corresponde al periodo desde que el directivo solicita al estudiante activar la cámara hasta su salida de la sala.

Gracias a la reducción de los tiempos de atención con la implementación del Sistema de Ajuste de Matrícula (SAM), se logró reducir la cantidad de personal necesario para gestionar las solicitudes. En el modelo basado en Zoom, los tiempos elevados generaban un atasco, lo que provocaba una acumulación significativa de estudiantes en la última franja horaria sin

haber podido ajustar su horario académico. Para mitigar esta situación, se requería la asignación de personal adicional, con dos personas durante los primeros dos días del ajuste y tres el último día para procesar todas las solicitudes pendientes.

En contraste, con la implementación del SAM en el semestre 2025-1, la gestión de solicitudes fue más eficiente, permitiendo que solo dos directivos atendieran un alto porcentaje de los trámites durante los primeros dos días del proceso, optimizando así los recursos humanos y reduciendo la carga operativa.

Beneficios Intangibles

La implementación del sistema ofrece una serie de beneficios cualitativos de gran relevancia para la institución:

1. Modernización Tecnológica

- Digitalización de procesos tradicionalmente manuales
- Adopción de estándares tecnológicos modernos
- Mejora en la experiencia de usuario

2. Optimización de Procesos

- Automatización de flujos de trabajo
- Eliminación de procedimientos redundantes
- Estandarización de procesos
- Reducción de errores humanos

3. Calidad de Información

- Mayor precisión en el seguimiento de solicitudes
- Trazabilidad completa de procesos
- Generación de estadísticas en tiempo real
- Mejora en la toma de decisiones

4. Eficiencia Operativa

- Reducción en tiempos de respuesta
- Mayor capacidad de atención a solicitudes
- Mejor distribución de carga laboral
- Optimización de recursos humanos

5. Satisfacción de Usuarios

- Mejor experiencia para estudiantes y docentes
- Acceso 24/7 a la plataforma
- Transparencia en el proceso de solicitudes
- Comunicación más efectiva

6. Gestión Institucional

- Descentralización de procesos administrativos
- Mejor coordinación entre departamentos
- Facilidad de escalamiento del sistema
- Adaptabilidad a cambios institucionales

7. Impacto en la Comunidad Académica

- Mayor autonomía para estudiantes y docentes
- Mejora en la percepción institucional
- Fortalecimiento de la cultura digital

La implementación del sistema representa una transformación significativa en la manera en que la institución gestiona las solicitudes académicas, proporcionando beneficios tangibles e intangibles que justifican ampliamente la inversión realizada.

5. Conclusiones

El sistema de ajuste de matrícula (SAM) fue diseñado con el fin de mejorar los tiempos de atención a solicitudes de ajuste de matrícula permitiéndole a la escuela realizar este proceso

de una forma más organizada clarificando el orden de prioridades y haciendo prevalidaciones para descartar solicitudes inviables, de esta forma el director o coordinador de escuela pueden dedicar más tiempo para atender un mayor número de solicitudes. SAM le permite al estudiante permanecer fuera de línea durante el procesamiento de sus solicitudes, lo que les permite seguir con sus actividades personales, una funcionalidad que no era posible en el modelo anterior de atención mediante videollamadas en Zoom

Los objetivos establecidos al inicio del proyecto fueron abordados en cada fase metodológica descrita anteriormente con la formalización de una propuesta en forma de un prototipo funcional, por medio del cual se realizó el proceso de ajuste de matrícula de 277 estudiantes de la Escuela de Ingeniería de Sistemas e Informática durante el periodo 2025-1, atendiendo en total 393 solicitudes de acuerdo a la siguiente figura.

Figura 30.

Vista de estadísticas generales



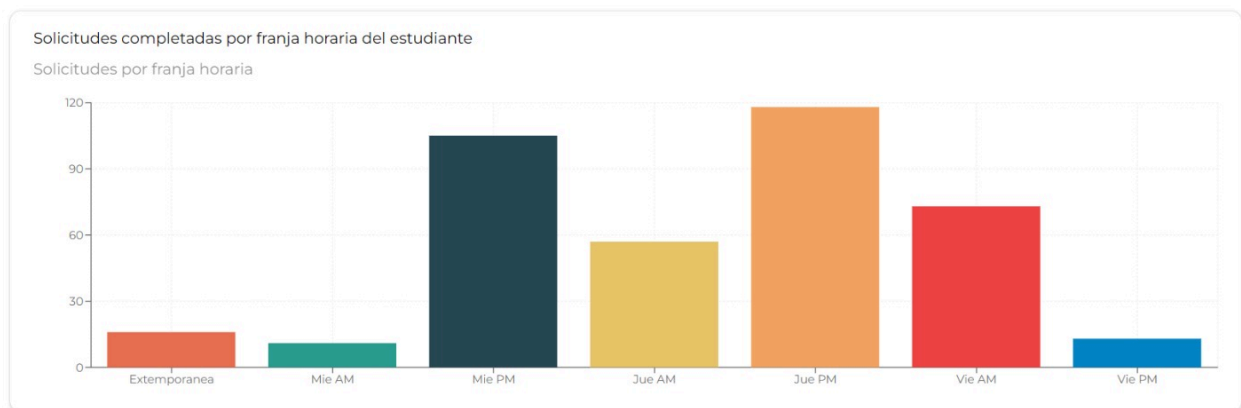
La atención de las solicitudes mediante SAM fue realizada por un único usuario administrador, por ende, se obtuvo un tiempo promedio de respuesta un poco alto con un total de 4.2 horas, para 393 solicitudes atendidas. Al comparar estos resultados con los tiempos

obtenidos mediante Zoom en el semestre anterior (2024-2), donde se atendieron 61 solicitudes en un solo día, se puede inferir que para procesar 393 solicitudes a través de Zoom se requerirían aproximadamente 6 días. En contraste, SAM permitió atender el mismo volumen en tan solo 3 días, evidenciando una significativa reducción en los tiempos de respuesta.

El proceso de ajuste de matrícula se lleva a cabo durante un periodo de 3 días, en los cuales la cantidad de estudiantes solicitando ajustes es constante hasta el último día, con la puesta en marcha de SAM se sintió un fuerte impacto en este patrón ya que casi todos los estudiantes fueron atendidos en los primeros dos días, en el tercer día del proceso las solicitudes fueron considerablemente bajas como podemos observar a continuación.

Figura 31.

Vista de estadística solicitudes completadas por franja horaria



Este comportamiento hizo innecesario incorporar personal adicional para la atención, como había ocurrido en semestres anteriores permitiendo a la escuela no solo reducir tiempos si no también gestionar de una mejor forma la asignación de recursos humanos para otras actividades.

Desde la perspectiva del estudiante, se observó una buena aceptación del sistema, lo cual se confirmó a través de la encuesta de satisfacción presentada en la sección de anexos. Los resultados indican que el 94,76% de los estudiantes recomendaría realizar el proceso de ajuste de matrícula mediante SAM, y un 78% prefiere utilizar esta plataforma. Sin embargo, el tiempo de respuesta no obtuvo una valoración positiva, ya que solo un usuario estaba gestionando las solicitudes a través de SAM. En este aspecto, el 41% de los encuestados se siente muy satisfecho con el tiempo de respuesta, mientras que un 28,2% manifiesta estar satisfecho.

6. Recomendaciones a trabajos futuros

Para futuras mejoras y expansiones del sistema, se proponen las siguientes recomendaciones:

6.1 Implementar un sistema de ayuda interactivo para estudiantes y administrativos

Desarrollar un módulo de soporte con tutoriales interactivos, guías paso a paso o módulos de preguntas frecuentes, permitiendo que los usuarios comprendan mejor el uso del sistema sin necesidad de asistencia manual. Esto facilita la adaptación al sistema y reduciría la carga de soporte.

6.2 Ampliación del sistema a otras escuelas e integración con plataformas institucionales

Se recomienda explorar la posibilidad de expandir el sistema a otras escuelas y facultades de la UIS, permitiendo que más estudiantes y administrativos gestionen sus horarios y solicitudes de manera centralizada.

6.3 Obtención de permisos para acceso a información en tiempo real

Gestionar los permisos necesarios para que el sistema pueda consultar en tiempo real las asignaturas programadas del semestre, evitando inscripciones en cursos sin capacidad o

inexistentes y optimizando la gestión de horarios así como la creación de solicitudes que no se pueden realizar. Esto implicaría la integración con bases de datos institucionales y la implementación de API seguras para la sincronización de información.

6.4 Carga automática y gestión inteligente de información de los estudiantes

Se recomienda implementar un sistema que permita cargar automáticamente la información académica de cada estudiante, obteniendo datos en tiempo real desde los sistemas institucionales, esto obteniendo los permisos necesarios con la universidad. Esto incluiría:

- **Horario matriculado:** Sincronización automática con el sistema de matrícula para que los estudiantes no tengan que ingresar manualmente sus asignaturas.
- **Franja horaria asignada:** Registro del turno asignado al estudiante.
- **Créditos disponibles:** Consulta de la cantidad de créditos restantes que el estudiante puede matricular en ese semestre.
- **Materias ya vistas:** Identificación de asignaturas aprobadas previamente para evitar registros duplicados.
- **Materias restringidas:** Filtrado de asignaturas que el estudiante no puede ver, ya sea por: No cumplir los requisitos o prerrequisitos establecidos por cada materia o pertenecer a un nivel superior en su plan de estudios.

Con esta mejora, la gestión académica sería más eficiente y los estudiantes tendrían una experiencia más fluida en la organización de su carga académica.

6.5 Mejoras en la infraestructura del servidor para soportar mayor carga

Se recomienda mejorar la infraestructura del servidor para manejar un mayor volumen de solicitudes simultáneas, especialmente en periodos de alta demanda como la inscripción de asignaturas. Algunas mejoras incluyen:

- **Escalabilidad horizontal y vertical:** Uso de contenedores y balanceo de carga para distribuir el tráfico.
- **Monitoreo en tiempo real:** Integración con herramientas de monitoreo para detectar cuellos de botella y prevenir fallas en el sistema.

Implementación de solicitudes compartidas entre estudiantes

Desarrollar una funcionalidad que permita a los estudiantes crear solicitudes compartidas para intercambio de grupos. En este caso, un estudiante podría crear una solicitud de cambio e invitar a otro estudiante a intercambiar grupos, enviándole una petición que el otro estudiante debe aceptar. Una vez ambos estudiantes confirmen, la solicitud se enviará como un único trámite para su aprobación. Esto agiliza el proceso de cambios de grupo y evitaría la creación de solicitudes duplicadas.

6.6 Sistema de recomendación de horarios y asignaturas con IA

Implementar un motor de recomendación basado en IA que sugiera asignaturas y horarios óptimos para cada estudiante, teniendo en cuenta:

- **Disponibilidad de horarios:** Preferencias de franja horaria y cursos con cupos disponibles.
- **Rendimiento y dificultad:** Materias en las que estudiantes con perfiles similares han tenido éxito.
- **Optimización del tiempo:** Minimización de huecos en el horario y balance de carga académica.

Optimización de cambios de grupo con IA

Implementar un sistema de IA que analice la cola de solicitudes en tiempo real para identificar oportunidades de intercambio o inclusión más eficientes.

- **Emparejamiento inteligente:** Si un estudiante quiere cambiar del grupo B al A y otro de A al B, el sistema los va a vincular y sugerirá el intercambio directo al administrador.

- **Gestión de cupos en grupos llenos:** Si un estudiante quiere incluir un grupo y este está lleno pero hay una solicitud de cancelación, se notificará a los administradores para procesar ambas solicitudes juntas.
- **Automatización para administradores:** Panel con coincidencias detectadas, reduciendo rechazos innecesarios y optimizando la asignación de cupos.

Referencias bibliográficas

- [1] Margoth Acuña Garzón (2016). Evaluación del Sistema de Información Académica (SIA) de la Universidad Nacional de Colombia. Recuperado de <https://repositorio.unal.edu.co/bitstream/handle/unal/59044/MargothAcuñaGarzón.2016.pdf?sequence=1&isAllowed=y>
- [2] Briceño Santiago, Wheeler Jonatan (2024). Los sistemas de información para la gestión académica y la toma de decisiones de la Universidad Nacional del Comahue. Período post pandemia. Recuperado de <https://rdi.uncoma.edu.ar/bitstream/handle/uncomaid/18409/Tesina%20%20-%20WHEELER%20-%20BRICEÑO.pdf?sequence=1&isAllowed=y>
- [3] Chiavenato, I. (2009). *Introducción a la teoría general de la administración*. McGraw-Hill.
- [4] Laudon, K. C., & Laudon, J. P. (2021). *Management Information Systems: Managing the Digital Firm*. Pearson.
- [5] O'Brien, J. A., & Marakas, G. M. (2021). *Introduction to Information Systems*. McGraw-Hill.
- [6] Laudon *Sistemas De Informacion Gerencial 14 Edicion : Free Download, Borrow, and Streaming : Internet Archive*. (2020, 2 mayo). Internet Archive. <https://archive.org/details/laudon-sistemas-de-informacion-gerencial-14-edicion>
- [7] Ortega, L. (2022, 22 octubre). Metodología RUP: ¿Qué es, cuál es su objetivo y cómo se utiliza? Lean Management Blog. <https://lean-management.site/rup/>
- [8] Anaya Jaimes, Y. M. (2022). Presupuesto: Unidad de Apoyo a Gestión de Proyectos, Vicerrectoría de Investigación y Extensión. Universidad Industrial de Santander. Recuperado de <https://uis.edu.co/wp-content/uploads/2022/08/TARIFAS-INSTITUCIONALES-2022-2.pdf>
- [9] Glassdoor. 2024. *Sueldo promedio de desarrolladores web en Bogotá, Colombia*. Glassdoor. Recuperado de https://www.glassdoor.com.mx/Sueldos/bogot%C3%A1-desarrollador-web-sueldo-SRCH_IL_0,6_IM1064_KO7,24.htm

Apéndices

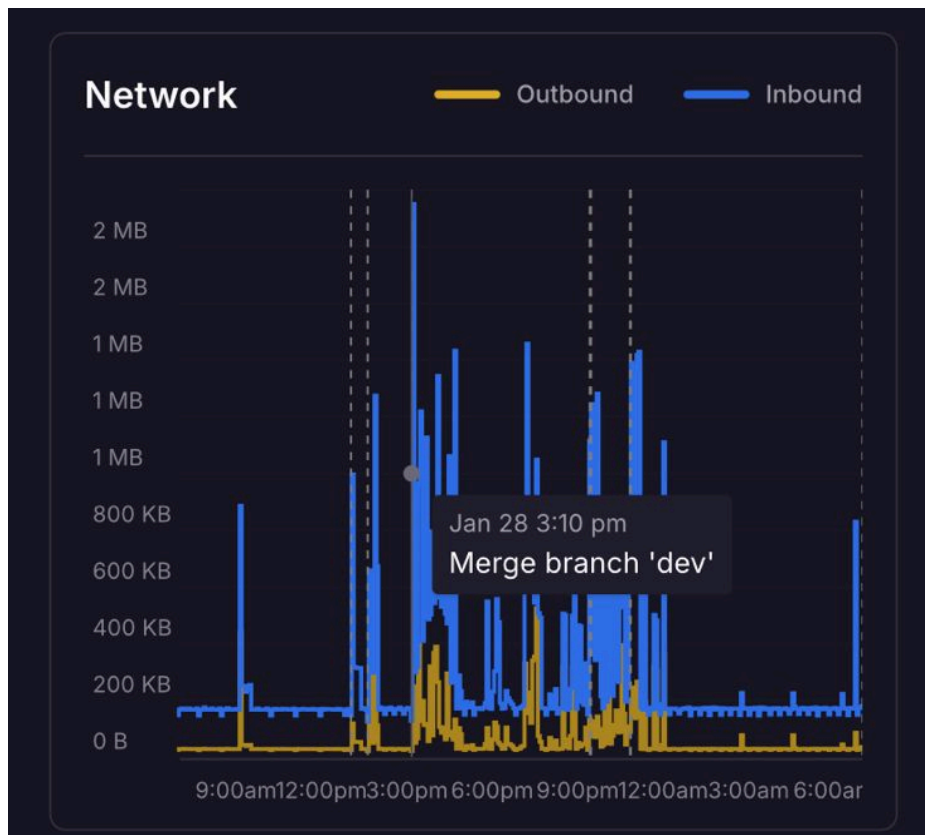
Apéndice a. enlace video tutorial para el estudiante sobre el uso del sistema

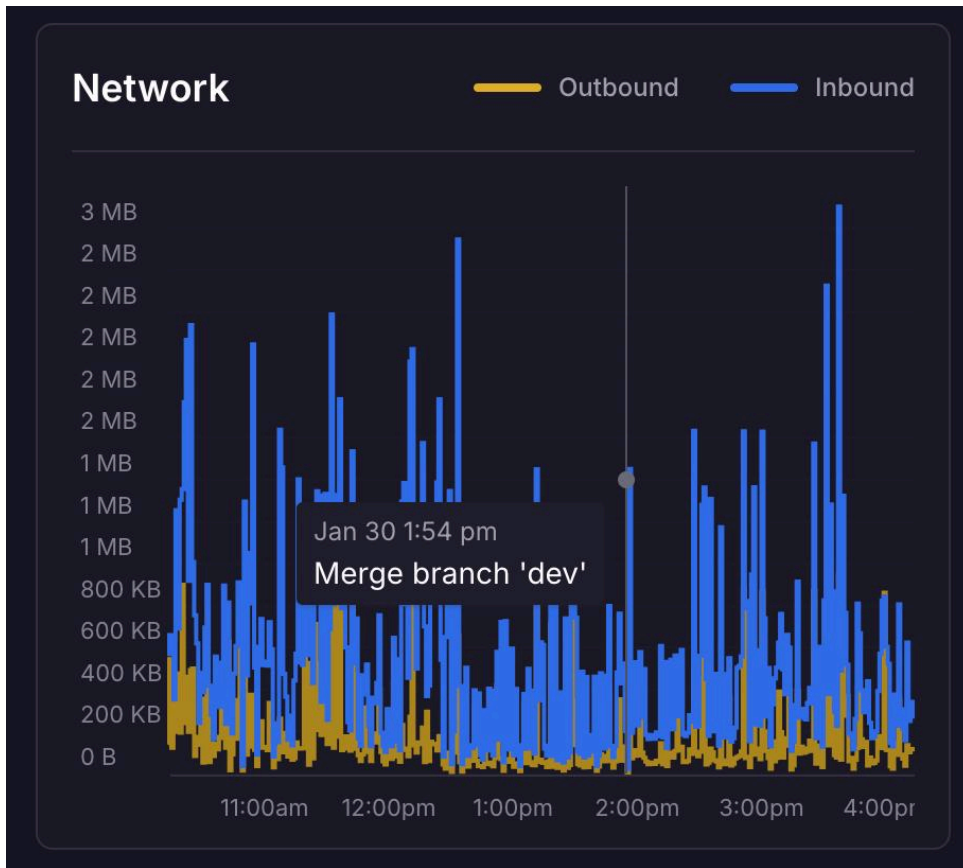
https://youtu.be/A2i_13bcRB0?si=CbOFI-MVA-m-Zaa6

Apéndice b. enlace video tutorial para coordinador y director académico sobre el uso del sistema

<https://youtu.be/HUkOLLJIG44?si=286huy2xy7ofNyds>

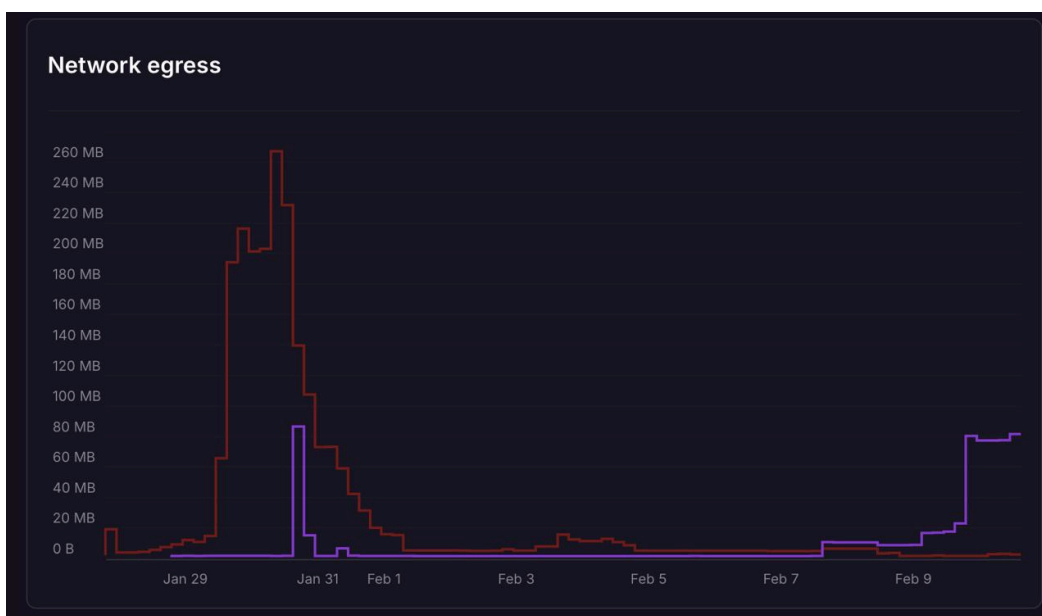
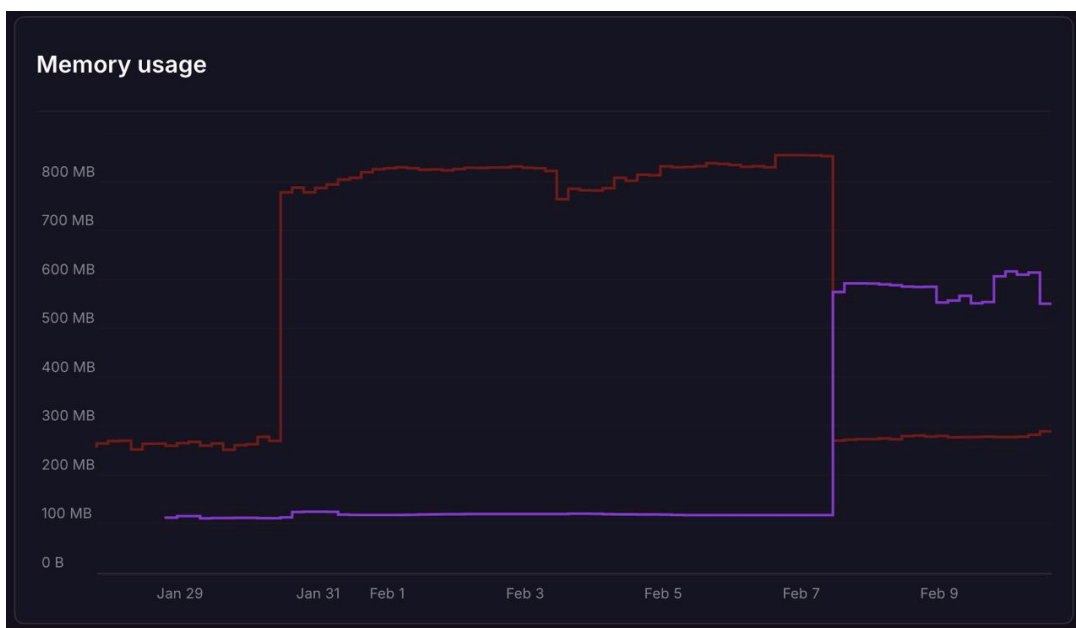
Apéndice c. capturas del tráfico de red tras su implantación

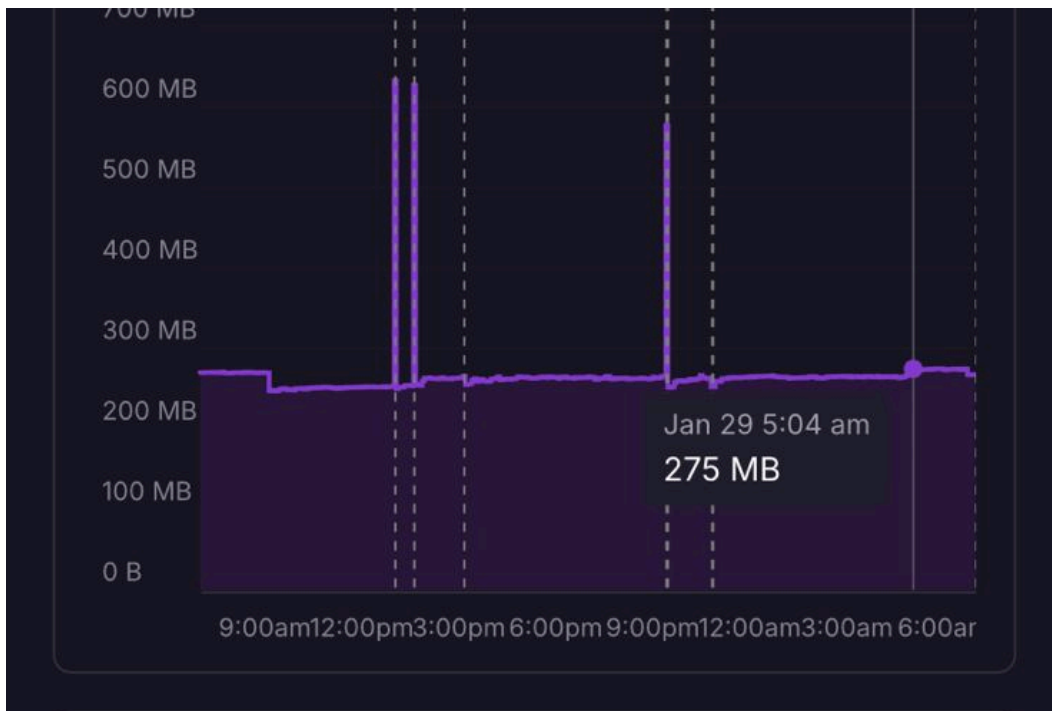




Apéndice d. capturas del consumo de recursos del servidor tras implantación



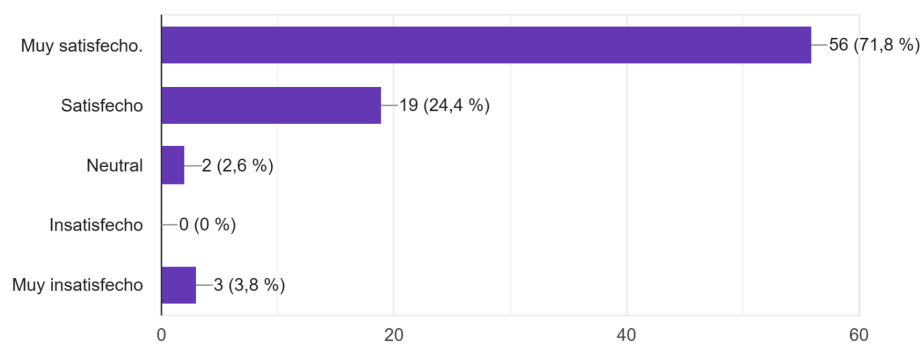




Apéndice e. encuesta realizada para medir la satisfacción de los estudiantes que realizaron el proceso de ajuste de matrícula por sam

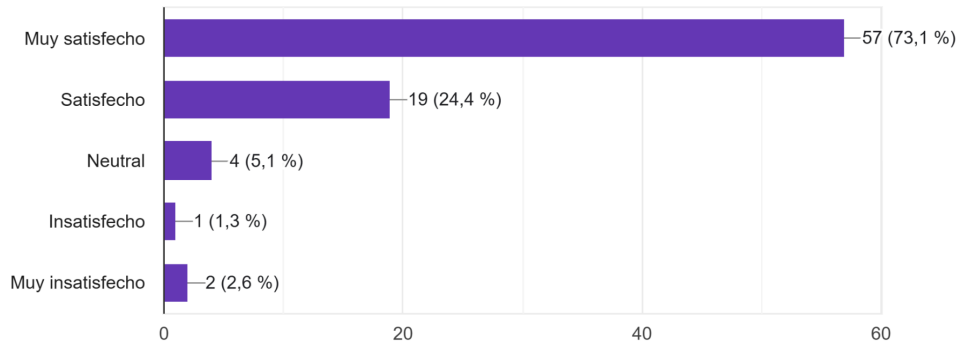
¿Qué tan satisfech@ se siente con la facilidad de navegación y uso del sistema?

78 respuestas



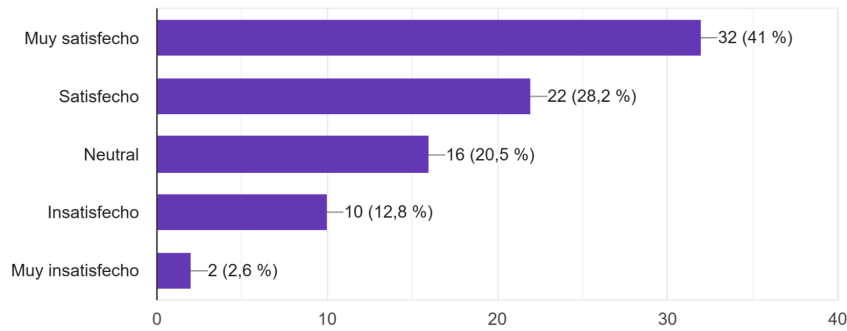
¿Qué tan satisfech@ está con la claridad de la información presentada, por ejemplo: datos de asignaturas, grupos, horarios, etc?

78 respuestas



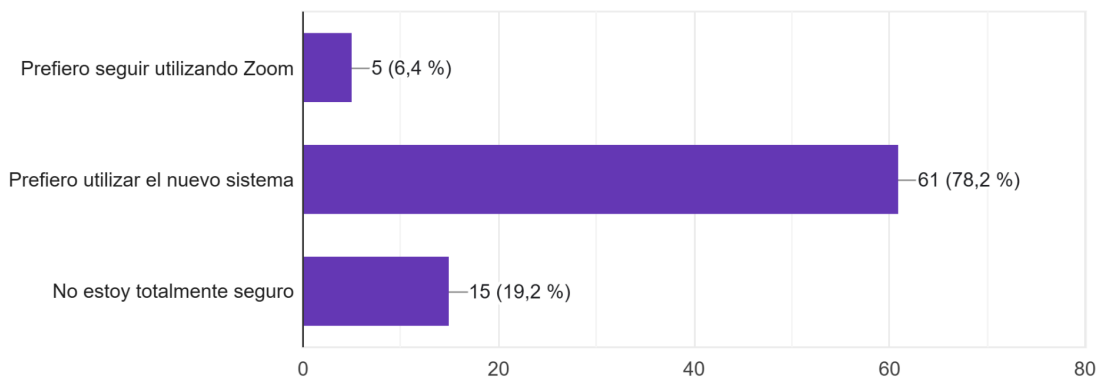
¿Qué tan satisfech@ se siente con el tiempo de respuesta y el rendimiento del sistema durante el proceso de matrícula?

78 respuestas



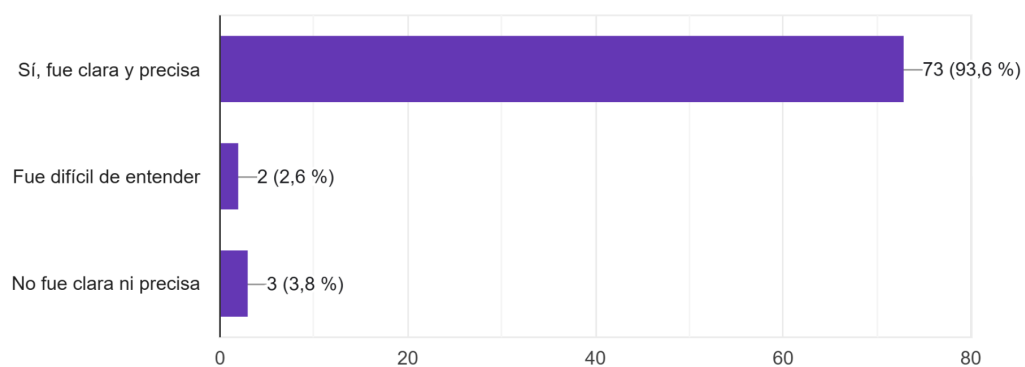
¿Preferirías seguir realizando el proceso de ajuste de matrícula mediante la plataforma Zoom o utilizar el nuevo sistema de información web?

78 respuestas



¿La retroalimentación que recibiste por correo electrónico sobre el estado de tu solicitud de ajuste de matrícula fue clara y precisa?

78 respuestas



Recomendarías a otro estudiante realizar el proceso de ajuste de matrícula mediante el sistema web?

78 respuestas

