

SEP

TECNM

DITD

INSTITUTO
TECNOLÓGICO
SUPERIOR DE
ATLIXCO



**INSTITUTO TECNOLÓGICO SUPERIOR
DE ATLIXCO**

Organismo Público Descentralizado del Gobierno del Estado de Puebla

**NOMBRE DEL TRABAJO
ACTUALIZACIÓN DEL SISTEMA
DE INGRESOS PROPIOS ITSA**

**OPCIÓN
TESIS**

QUE PARA OBTENER EL TÍTULO DE:
INGENIERO EN SISTEMAS COMPUTACIONALES

PRESENTA:
JAVIER ORTEGA ZAMORA

ASESOR: **DRA. MARIELA J. ALONSO CALPEÑO**

ATLIXCO, PUE. OCTUBRE DE 2023

Dedicatoria

A mis queridos padres, Marco y Debora, les dedico este trabajo con profundo agradecimiento por su inquebrantable apoyo y amor. Han sido mi fortaleza en cada desafío que he enfrentado en mi camino, su sabiduría y consejos han sido la brújula que me ha guiado en las decisiones más trascendentales de mi vida.

A mis amados abuelos, aunque ya no están físicamente conmigo, su legado perdura en lo más profundo de mi corazón. Agradezco sus primeras muestras de interés en mis estudios, su constante seguimiento de mis logros y su eterna compañía, ahora desde el cielo.

A mis amigos, les agradezco por compartir conmigo momentos inolvidables a lo largo de esta travesía. Su singular personalidad ha iluminado cada instante con alegría, y juntos hemos recorrido el camino hacia la realización de nuestros sueños compartidos.

A Dios, a la vida y al universo, les dedico un profundo agradecimiento por otorgarme la salud y la oportunidad de vivir esta experiencia. Su guía ha sido fundamental en la toma de decisiones a lo largo de mi vida. Que todas estas bendiciones nunca me falten.

Agradecimientos

En el transcurso de esta travesía, cuando los desafíos comenzaron a tornarse complicados, siempre conté con el apoyo inquebrantable de las personas más importantes en mi vida. Mis padres, quienes fortalecieron mi ambición de alcanzar mis sueños, merecen un agradecimiento especial. Hoy, a punto de convertirme en un profesional, me enorgullezco de llevar sus apellidos.

También quiero expresar mi gratitud a todos mis profesores. Sus enseñanzas y consejos, tanto dentro como fuera del aula, han sido fundamentales en mi desarrollo. Gracias a su conocimiento, no solo me convierto en un profesional, sino también en un ser humano en constante crecimiento.

Agradezco de todo corazón a mi familia por su interés inquebrantable en mi éxito y por caminar junto a mí en la búsqueda de cada uno de mis sueños. Siempre han brindado amor y alegría cuando más los he necesitado.

Mi agradecimiento se extiende al Instituto Tecnológico Superior de Atlixco, que me abrió las puertas a un nuevo hogar y me permitió conocer a personas especiales que han hecho de mi estancia una experiencia única y especial.

En resumen, quiero agradecer a cada uno de ustedes por ser parte de este momento que quedará grabado en mi corazón para toda la vida. Su apoyo y compañía han sido invaluable en este viaje hacia la realización de mis metas.

Resumen

El presente proyecto tiene como finalidad actualizar el sistema web existente enfocado a la cobranza de los servicios y productos ofrecidos por el Instituto Tecnológico Superior de Atlixco, a la vez de adquirir conocimientos prácticos y teóricos sobre el desarrollo de sistemas web. El objetivo es mejorar el control de los ingresos del ITSA a través de esta actualización.

La actualización del sistema permitió proporcionar mejoras como: optimizar aún más el proceso de venta y cobranza de servicios y productos, crear reportes fiscales y de control más detallados para uso interno, y mejorar la experiencia de usuario brindada. Al mismo tiempo, nos permitió poner en práctica los conocimientos teóricos y obtener nuevos conocimientos prácticos sobre el desarrollo de sistemas web.

Abstract

The present project aims to update the existing web system focused on collecting payment for services and products offered by the Instituto Tecnológico Superior de Atlixco, while acquiring practical and theoretical knowledge about web system development. The goal is to improve ITSA's revenue control through this update.

The system update allowed improvements such as: further optimizing the sales and collection process for services and products, creating more detailed fiscal and internal control reports, and enhancing the user experience. At the same time, it allowed us to put theoretical knowledge into practice and obtain new practical knowledge about web system development.

Contenido

Dedicatoria	2
Agradecimientos.....	3
Resumen	4
Abstract.....	4
Índice de figuras	8
Índice de tablas.....	9
Introducción.....	10
Capítulo 1: Protocolo de investigación	11
1.1. Planteamiento del problema	11
1.2. Objetivo general.....	12
1.3. Objetivos específicos.....	12
1.4. Justificación	13
1.5. Hipótesis	14
1.6. Alcances y Limitaciones.....	14
Capítulo 2: Marco teórico o estado del arte.....	16
2.1. Metodología Scrum.....	16
2.1.1. Sprints.....	16
2.2 Sistema web	17
2.3 Herramientas de desarrollo.....	18
2.4. Base de datos.....	19
2.5. Pruebas de software	19

2.6. Documentación de sistema.....	20
2.7. Control de versiones	20
Capítulo 3: Metodología y planificación	23
3.1 Metodología	23
3.2 Levantamiento de requerimientos.....	23
3.2.1 Requerimientos funcionales	24
3.2.2 Requerimientos no funcionales	35
3.2.3 Reglas de negocio.....	36
3.3 Planificación (Product Backlog y Sprint Backlog)	38
3.4 Estimación (Recursos y esfuerzo)	40
3.5 Costos.....	41
3.6 Cronograma	43
3.7 Diseño de interfaces	45
3.8 Diseño de base de datos	46
3.9 Arquitectura del sistema	49
3.9.1 Diagrama de Casos de Uso	49
3.9.1.1 Administrador	49
3.9.1.2 Administrador financiero.....	50
3.9.1.3 Administrador de catálogo.....	50
3.9.2.1 Diagrama de Caso de Uso – General	51
3.9.2.2 Diagrama de Caso de Uso – Administrador	52
3.9.2.3 Diagrama de Caso de Uso – Administrador financiero.....	53
3.9.2.4 Diagrama de Caso de Uso – Administrador de catálogo.....	54

3.9.3 Diagrama de navegación	55
3.9.3.1 Inicio de Sesión y Panel de Inicio.....	55
3.9.3.2 Sección de Usuarios	55
3.9.3.3 Sección de Alumnos.....	56
3.9.3.4 Sección de Servicios	56
3.9.3.5 Sección de Referencias	57
3.9.3.6 Sección de Administrar	57
3.9.4 Modelo-Vista-Controlador	58
Capítulo 4: Desarrollo y pruebas	60
4.1 Desarrollo	60
4.2 Sprint 1 (RF-001 – RF-002)	60
4.3 Sprint 2 (RF-003)	62
4.4 Sprint 3 (RF-004)	63
4.5 Sprint 4 (RF-005 – RF-008)	65
4.6 Sprint 5 (RF-009)	70
4.7 Pruebas	73
4.8 Seguimiento y control	73
Capítulo 5: Resultados	75
5.1 Discusión de resultados	75
5.2 Conclusiones del proyecto.....	78
6. Referencias	79

Índice de figuras

Ilustración 1. Product Backlog.....	38
Ilustración 2. Sprint Backlog.....	39
Ilustración 3. Diseño de interfaces	45
Ilustración 4. Diagrama relacional - BD Principal	47
Ilustración 5. Diagrama relacional - BD Histórico	48
Ilustración 6. Diagrama de Caso de Uso - General	51
Ilustración 7. Diagrama de Caso de Uso - Administrador.....	52
Ilustración 8. Diagrama de Caso de Uso - Administrador financiero	53
Ilustración 9. Diagrama de Caso de Uso - Administrador de catálogo	54
Ilustración 10. Diagrama de navegación	58
Ilustración 11. Diagrama modelo-vista-controlador	59
Ilustración 12. Interfaz del módulo "Restaurar sistema".....	61
Ilustración 13. Interfaz del módulo "Consultar histórico"	62
Ilustración 14. Funcionalidad de generar reporte en "Consultar histórico"	63
Ilustración 15. Formato "Reporte de póliza de pago".....	64
Ilustración 16. Interfaz del módulo "Administrar alumnos".....	65
Ilustración 17. Formulario "Agregar nuevo alumno".....	66
Ilustración 18. Interfaz del módulo "Ver alumnos".....	68
Ilustración 19. Formulario "Actualizar datos del alumno".....	69
Ilustración 20. Proceso de eliminación de alumno.....	70
Ilustración 21. Interfaz del módulo "Cargar alumnos".....	71
Ilustración 22. Formato "Cargar alumnos".....	72
Ilustración 23. Proceso de carga de alumnos antes de la actualización.....	76
Ilustración 24. Proceso de carga de alumnos después de la actualización.....	77

Índice de tablas

Tabla 1. Requisito funcional 001	24
Tabla 2. Requisito funcional 002	25
Tabla 3. Requisito funcional 003	26
Tabla 4. Requisito funcional 004	28
Tabla 5. Requisito funcional 005	29
Tabla 6. Requisito funcional 006	30
Tabla 7. Requisito funcional 007	31
Tabla 8. Requisito funcional 008	32
Tabla 9. Requisito funcional 009	33
Tabla 10. Cronograma de desarrollo	43

Introducción

El objetivo principal de este trabajo es analizar el comportamiento del sistema de ingresos propios del Instituto Tecnológico Superior de Atlixco, optimizando la administración y los resultados obtenidos en las operaciones sobre dichos ingresos.

La función primordial del sistema actualizado es incorporar nuevos alumnos al registro almacenado en la base de datos, permitiendo agilizar el proceso de creación de órdenes de pago.

Analizando la problemática que motivó la actualización, es preciso mencionar que durante su desarrollo inicial se contemplaba operar en coordinación con el departamento de Control Escolar, con el fin de mantener los datos de cada estudiante constantemente actualizados.

Tras meses de operación, se detectó un amplio rezago en la actualización de información debido a la complejidad en el acceso a los datos. Esto obligó a buscar una solución alternativa en una nueva etapa de desarrollo. A su vez, se aprovechó este proceso para crear una interfaz capaz de proteger y resguardar la información capturada en la plataforma misma.

Capítulo 1: Protocolo de investigación

1.1. Planteamiento del problema

El Instituto Tecnológico Superior de Atlixco cuenta actualmente con un sistema web para la gestión de pagos de servicios por parte de alumnos y personas externas a la institución. Sin embargo, tras meses de operación, se han detectado algunas áreas de oportunidad para optimizar procesos y funcionalidades.

Específicamente, se requiere implementar un módulo de registro manual de alumnos, ya que actualmente depende únicamente de la información provista por Control Escolar, la cual debe ser operada por el primer equipo de desarrollo, esto provoca que haya retrasos en la generación de órdenes de pago y en la desactualización de la información. Asimismo, es necesario desarrollar una interfaz para el resguardo de información histórica, que permita su protección y gestión por años fiscales por parte del departamento de Finanzas. Adicionalmente, se solicitó la corrección del formato generado al descargar una póliza de pago en el sistema, para evitar inconsistencias en la presentación de la información.

De esta manera, a través de una actualización del sistema web existente, se busca optimizar procesos, corregir limitaciones detectadas y dotar de nuevas funcionalidades que mejoren el control, manejo y protección de la información de ingresos y pagos del instituto. También, el departamento de Finanzas podrá hacer uso de los registros históricos para su análisis y toma de decisiones.

1.2. Objetivo general

Actualizar el sistema web de ingresos propios del Instituto Tecnológico Superior de Atlixco con la finalidad de optimizar procesos, corregir limitaciones y dotar de nuevas funcionalidades que permitan mejorar el control, manejo y protección de los datos de ingresos y pagos de la institución.

1.3. Objetivos específicos

- Identificar y documentar de manera exhaustiva los requerimientos funcionales y no funcionales del sistema web, asegurándose de comprender las necesidades de los usuarios y las expectativas del cliente.
- Elaborar un diseño detallado de la interfaz de usuario y la arquitectura del sistema, asegurando una experiencia de usuario intuitiva y fácil navegación.
- Establecer un plan de proyecto que defina los recursos, plazos y actividades necesarias para desarrollar la actualización del sistema web de forma ordenada.
- Implementar el código del sistema web de manera eficiente y organizada, siguiendo las directrices de diseño previamente establecidas para garantizar un desarrollo coherente y efectivo.
- Realizar pruebas exhaustivas en el sistema web asegurando su funcionamiento sin errores, identificando y resolviendo cualquier problema relacionado con la presentación.
- Integrar los componentes del sistema web de manera que el ciclo de desarrollo mantenga su coherencia en todas las secciones y funcionalidades, así mismo, asegurando que los datos y la información se muestre de forma ordenada y precisa.
- Implementar la actualización del sistema web en un entorno de producción, asegurando su disponibilidad para los usuarios finales de forma ordenada y accesible.

- Crear documentación detallada y accesible que explique el funcionamiento del sistema web y su capacidad para mostrar de forma ordenada el ciclo de desarrollo del software. Esto incluye manuales de usuario y técnicos.

1.4. Justificación

El sistema web de ingresos propios del Instituto Tecnológico Superior de Atlixco necesita una actualización con el propósito de optimizar procesos, subsanar limitaciones detectadas durante la operación inicial y dotarlo de nuevas funcionalidades que permita un mejor control, manejo y protección de los datos financieros y de pagos de la institución.

El desarrollo de un módulo de registro manual de alumnos independiente del área de Control Escolar, así como una interfaz para el resguardo y gestión de registros históricos, son fundamentales para mantener los datos actualizados y protegidos ante cualquier eventualidad.

Asimismo, resulta indispensable incorporar requerimientos específicos del área de Finanzas, principal usuaria del sistema, con el fin de que la plataforma se ajuste de mejor manera a sus procesos y necesidades.

De esta forma, el proyecto permitirá contar con un sistema web optimizado, corregido y fortalecido en sus capacidades de registro, resguardo y gestión de la información de ingresos y pagos del Instituto Tecnológico Superior de Atlixco.

1.5. Hipótesis

La utilidad de un sistema de información se incrementa cuando se agrega la funcionalidad que da libertad al usuario administrador sobre la actualización de los catálogos del sistema, sin depender necesariamente de la carga de datos por parte de terceros.

1.6. Alcances y Limitaciones

El presente proyecto de actualización del sistema web de ingresos propios tiene definido un alcance específico que abarca elementos clave como el desarrollo de nuevos módulos, incorporación de requerimientos y pruebas de funcionamiento. Asimismo, existen limitaciones inherentes, tanto en relación con el sistema de información del Instituto Tecnológico Superior de Atlixco, como en cuanto a disponibilidad de recursos y participación de áreas involucradas.

Los siguientes puntos describen de manera concisa los alcances y limitaciones que enmarcan este proyecto de mejora al sistema web existente.

Alcances:

- Se desarrollará e implementará un módulo de registro manual de alumnos.
- Se creará una interfaz para la gestión y resguardo de registros históricos financieros.
- Se actualizarán formatos generados por el sistema para mejorar su presentación en los procesos del área de finanzas.
- Se realizarán pruebas de funcionalidad sobre las mejoras y nuevos desarrollos.

- Se elaborarán manuales de usuario y técnico del sistema actualizado.

Limitaciones:

- La implementación de mejoras dependerá de la autorización de los responsables técnicos y administrativos.
- El levantamiento de requerimientos fue coordinado y aprobado por los responsables administrativos.
- Las pruebas no contemplan la migración de datos históricos futuros, solo la migración de datos previos.
- El módulo de registro manual de alumnos deberá ajustarse al formato y estructura de datos definidos inicialmente en la base de datos del sistema.
- El entorno de producción se rige bajo las restricciones de una red local.

Capítulo 2: Marco teórico o estado del arte

Este capítulo presenta el marco teórico que sustenta el proceso de actualización del sistema web de gestión de pagos del Instituto Tecnológico Superior de Atlixco.

2.1. Metodología Scrum

Scrum es un proceso que aplica un conjunto de buenas prácticas para el trabajo en equipo y obtener el mejor resultado en un proyecto. Estas prácticas están seleccionadas a partir del estudio de equipos altamente productivos.

En Scrum se realizan entregas parciales y regulares del producto final, priorizadas por el valor que aportan al cliente. Por ello, Scrum es adecuado para proyectos complejos donde se requiere entregar resultados pronto, con requisitos cambiantes o poco definidos inicialmente, y donde la innovación, competitividad, flexibilidad y productividad son claves.

Scrum promueve la inspección y adaptación constantes ante cambios. Se enfoca en la funcionalidad operativa del producto y un equipo autosuficiente y multifuncional. Estas características hacen a Scrum ideal para actualizar sistemas de software existentes de manera ágil ante nuevos requerimientos [1].

2.1.1. Sprints

Los sprints son periodos de trabajo de duración fija y corta (generalmente de 2 a 4 semanas) en los que el equipo Scrum se enfoca en completar un incremento de producto potencialmente entregable. Los sprints son la unidad básica de desarrollo en Scrum [1].

Al inicio de cada sprint, se lleva a cabo una reunión de planificación donde se seleccionan los elementos prioritarios del Product Backlog para implementar en ese periodo. Durante el sprint, el Equipo Scrum trabaja en conjunto para alcanzar el objetivo establecido y crear un incremento de producto funcionando.

Realizar sprints de manera continua, con objetivos claros y duración fija, permite entregar valor al cliente de forma temprana y obtener feedback rápido para incorporar mejoras. Esto resulta esencial en proyectos ágiles de desarrollo o actualización de software.

2.2 Sistema web

Un sistema web consiste en un conjunto de componentes interconectados que interactúan para llevar a cabo ciertos procesos y entregar funcionalidades a los usuarios a través de internet o intranets [2].

Las aplicaciones web son un tipo de sistema clientes-servidor donde el cliente es el navegador y el servidor consta de tecnologías como servidores web y lenguajes de programación. Están diseñadas para ser utilizadas a través de navegadores web, ya sea en internet o redes internas [3].

2.3 Herramientas de desarrollo

Para llevar a cabo la actualización del sistema web se utilizaron principalmente las siguientes herramientas de desarrollo provistas por Microsoft:

- **Visual Studio:** entorno de desarrollo integrado para lenguajes como C# y .NET usados en la programación del sistema. Permite la edición, compilación y depuración del código, además de funciones de control de versiones y testing.
- **SQL Server:** sistema de gestión de bases de datos relacionales que almacena los datos del sistema web. Provee herramientas para la administración y optimización del rendimiento de las bases de datos.
- **SQL Server Management Studio:** interfaz gráfica para la administración de bases de datos SQL Server. Sirve para consultar, modificar y optimizar la estructura de las bases de datos.
- **Servidor IIS:** servidor web desarrollado por Microsoft para sistemas Windows. Se utiliza para hospedar el sitio web y las aplicaciones desarrolladas con ASP.NET.

El uso combinado de estas herramientas Microsoft facilitó el proceso de actualización del sistema en los aspectos de programación, bases de datos y deploy.

2.4. Base de datos

Las bases de datos relacionales permiten organizar la información en tablas interrelacionadas, lo cual reduce redundancias, previene inconsistencias y facilita el mantenimiento de integridad de los datos ante actualizaciones.

Al centralizar los datos en SQL Server se obtiene mayor seguridad, capacidad de consulta y optimización del rendimiento. Además, permite el acceso simultáneo para distintos procesos y usuarios del sistema web.

2.5. Pruebas de software

Las pruebas de software son una parte fundamental en el desarrollo de sistemas computacionales, ya que permiten verificar que el código cumple con los requerimientos y funciona según lo esperado. Existen diferentes tipos de pruebas que se pueden implementar durante el ciclo de vida del desarrollo de software.

Las pruebas unitarias validan el funcionamiento de módulos o componentes individuales del sistema. Por otro lado, las pruebas de integración verifican que los diferentes módulos o componentes trabajan correctamente al interactuar entre ellos. Las pruebas del sistema en su conjunto, conocidas como pruebas de sistema, evalúan el comportamiento general bajo condiciones normales y anormales.

Otro tipo importante de pruebas son las pruebas de humo, las cuales se enfocan en la funcionalidad crítica y se ejecutan al tener una primera versión ejecutable del sistema. Las pruebas de regresión buscan detectar nuevos errores y verificar que no se hayan introducido fallas en funciones que previamente pasaron las pruebas.

La implementación de un proceso de pruebas riguroso con diferentes tipos y niveles de prueba tanto manuales como automatizadas, es esencial para el éxito en el desarrollo de un sistema computacional de calidad.

2.6. Documentación de sistema

La documentación de software juega un rol clave en el desarrollo de sistemas, pues captura información esencial sobre el diseño, funciones y operación de las aplicaciones. Una adecuada documentación beneficia a diversos actores del ciclo de vida del software como desarrolladores, usuarios y equipos de mantenimiento.

Los documentos típicos en el desarrollo de software incluyen especificaciones de requerimientos, manuales de usuario, documentación técnica, documentos de arquitectura, entre otros. Las especificaciones funcionales y técnicas son útiles durante el desarrollo y testing. Los manuales de usuario facilitan la adopción.

Mantener la documentación de un sistema de información permite la transferencia de conocimiento entre equipos, reduce la curva de aprendizaje para nuevos miembros y facilita el mantenimiento y escalabilidad del sistema.

2.7. Control de versiones

Los sistemas de control de versiones son un tipo de software que ayuda a hacer un seguimiento de los cambios realizados en el código a lo largo del tiempo. A medida que un desarrollador edita el código, el sistema de control de versiones toma una instantánea de los archivos. Después, guarda esa instantánea de forma permanente para que se pueda recuperar más adelante si es necesario [14].

Los beneficios clave del control de versiones incluyen:

- Registro de cambios: Cada modificación queda registrada, lo que permite comprender la evolución del código.
- Trabajo en equipo: Los desarrolladores pueden trabajar en paralelo en distintas funcionalidades.
- Integración de código: Se pueden combinar los cambios de diferentes ramas de desarrollo.
- Reversión: Se puede regresar a versiones anteriores en caso de introducir errores.
- Resolución de conflictos: Las herramientas ayudan a resolver inconsistencias al integrar código.
- Backup y recuperación: El repositorio actúa como backup de cada versión del código.
- Administración de releases: Se pueden marcar versiones estables de entrega.
- Ramas de desarrollo: Permite separar el trabajo en ramas aisladas.
- Flujo de trabajo: Se definen políticas de contribución de código.

En general, el control de versiones automatiza tareas que de otro modo requerirían procedimientos manuales complejos y propensos a errores. El uso de un sistema de control de versiones se considera una práctica esencial en la ingeniería de software moderna [5].

Entre los conceptos fundamentales se encuentran el repositorio como lugar centralizado para almacenar todas las versiones, el sistema de commits para registrar

modificaciones, el branching para crear líneas divergentes de desarrollo, y el merging para integrar ramas [5].

Diferentes sistemas de control de versiones implementan estos conceptos utilizando arquitecturas centralizadas o distribuidas. Independientemente de la herramienta específica, la incorporación del control de versiones representa una ventaja significativa en la administración de proyectos de software de mediana y gran escala.

Capítulo 3: Metodología y planificación

3.1 Metodología

Para llevar a cabo la actualización del sistema web se decidió emplear la metodología ágil Scrum, enfocada en entregar valor al cliente de forma iterativa e incremental mediante sprints de trabajo colaborativo.

La actualización se realizó en 6 sprints, durante un periodo total de 3 meses. Al inicio de cada sprint se hizo una reunión de planificación para establecer los objetivos y tareas del incremento a desarrollar. Al finalizar se llevó a cabo una demo con los usuarios y una retrospectiva para mejorar procesos.

3.2 Levantamiento de requerimientos

Para esta etapa es esencial identificar y documentar los requerimientos para desarrollar la actualización del sistema web, así como cualquier otro proyecto de software. Se aplicaron como técnicas para la obtención de requerimientos los siguientes puntos:

1. Observación
2. Entrevistas

Se implementó la grabación de voz y la transcripción a texto durante las entrevistas realizadas en cada una de las sesiones programadas para garantizar la calidad del proceso.

La documentación de requerimientos fue elaborada bajo el estándar ANSI/IEEE 830 de 1998, mismo estándar recomendado por la IEEE para la especificación de requerimientos de software.

3.2.1 Requerimientos funcionales

Los requerimientos funcionales definen las funcionalidades y servicios que el sistema web debe ofrecer.

Tabla 1. Requisito funcional 001

Requerimiento funcional	Detalles
Nombre	RF-001 Restablecer sistema
Descripción	El sistema debe proporcionar un mecanismo seguro que permita al administrador restaurar el sistema a un estado inicial.
Entradas	El administrador deberá ingresar su contraseña para iniciar el proceso.
Procesamiento	El sistema comparará el usuario y la contraseña proporcionada con la información almacenada en la base de datos. Si los datos coinciden, el sistema iniciará el proceso. Si los datos no coinciden, el sistema denegará la solicitud y proporcionará un mensaje de error.

Salidas	Si los datos coinciden, el sistema ejecutará el respaldo de información y el vaciado de la base de datos principal como proceso de restauración. Si los datos no coinciden, el sistema mostrará un mensaje de error.
Restricciones	<ul style="list-style-type: none"> • Solo los usuarios autorizados podrán ejecutar este proceso. • Una vez iniciado el proceso cualquier cambio será irreversible.

Tabla 2. Requisito funcional 002

Requerimiento funcional		Detalles
Nombre	RF-002 Generar respaldo de histórico	
Descripción	El sistema deberá permitir generar un respaldo del histórico de referencias en la base de datos principal y trasladarlo a la base de datos de respaldo anual. Esto con el fin de liberar espacio en la DB principal y conservar registros de años anteriores.	
Entradas	El sistema recibirá como entrada la contraseña del usuario con permisos para ejecutar el proceso de respaldo histórico. Esta deberá ser ingresada en la interfaz de "Restaurar sistema" para poder iniciar el procedimiento.	

Procesamiento	El sistema validará que la contraseña ingresada corresponda a un usuario autorizado. Finalmente ejecutará un script transfiriendo los registros y liberando espacio en la DB principal.
Salidas	Si el proceso ha sido completado correctamente, el sistema redireccionará al usuario hacia el panel principal, en caso contrario, mostrará un mensaje de error.
Restricciones	<ul style="list-style-type: none"> • Solo los usuarios autorizados podrán ejecutar este proceso. • Se debe contar con espacio suficiente en el servidor antes de iniciar el proceso.

Tabla 3. Requisito funcional 003

Requerimiento funcional	Detalles
Nombre	RF-003 Consultar histórico
Descripción	El sistema deberá permitir consultar y generar reportes con las referencias que fueron respaldadas en la base de datos histórica durante procesos previos de restauración.

Entradas	El sistema recibirá como entradas las fechas de intervalo del histórico a consultar y los filtros de búsqueda para referencias específicas.
Procesamiento	El módulo recuperará de la base de datos histórica las referencias correspondientes al intervalo de fechas indicadas, aplicando cualquier filtro adicional especificado.
Salidas	Como salida se entregará un archivo Excel con el reporte de referencias históricas según los criterios de búsqueda.
Restricciones	<ul style="list-style-type: none"> • Solo los usuarios autorizados podrán ejecutar este proceso. • El reporte en Excel seguirá una plantilla predefinida.

Tabla 4. Requisito funcional 004

Requerimiento funcional	Detalles
Nombre	RF-004 Actualizar reporte de póliza
Descripción	El sistema debe ser capaz de generar correctamente un reporte de póliza después de conciliar referencias. Siguiendo el formato sugerido por el departamento de finanzas.
Entradas	El usuario ha conciliado referencias y solicita a través de un botón la descarga del formato "Reporte de póliza".
Procesamiento	El sistema recopila la información almacenada en la base de datos y genera un archivo con formato "xls" mediante el que se incluye el formato solicitado.
Salidas	Una vez finalizada la generación del archivo, el sistema web inicia la descarga de este a través del navegador del usuario.
Restricciones	<ul style="list-style-type: none"> • Solo los usuarios autorizados podrán ejecutar este proceso. • El reporte de póliza solo estará disponible para su descarga durante el proceso de conciliación de referencias. Una vez finalizado este proceso, ningún usuario podrá acceder al mismo reporte.

Tabla 5. Requisito funcional 005

Requerimiento funcional	Detalles
Nombre	RF-005 Crear alumno
Descripción	<p>El sistema debe permitir el registro de nuevos alumnos con sus datos asociados como:</p> <ul style="list-style-type: none"> • Nombre • Apellidos • Matrícula • Correo electrónico institucional • Datos de domicilio
Entradas	El usuario captura los datos del nuevo alumno a registrar.
Procesamiento	El sistema recopila la información y valida que los datos obligatorios estén registrados, verifica que la matrícula no exista previamente y realiza el registro del alumno si el proceso fue completado correctamente.
Salidas	El sistema web envía al usuario un mensaje de confirmación o de error de acuerdo con el estado en el que se completó el proceso.

Restricciones	<ul style="list-style-type: none"> • Solo los usuarios autorizados podrán ejecutar este proceso. • La matrícula del alumno debe ser única y no repetirse para otro alumno. • Los campos obligatorios no pueden ir vacíos o nulos. • El correo electrónico institucional debe tener la terminación “@itsatlixco.edu.mx”
----------------------	--

Tabla 6. Requisito funcional 006

Requerimiento funcional	Detalles
Nombre	RF-006 Consultar alumno
Descripción	El sistema debe permitir buscar y visualizar los datos de un alumno registrado.
Entradas	<p>El usuario utiliza como criterios de búsqueda:</p> <ul style="list-style-type: none"> • Nombre completo • Matrícula <p>Así como el apoyo de la paginación para agilizar el proceso.</p>
Procesamiento	El sistema valida los parámetros de entrada y realiza una consulta SQL que construye y ejecuta para mostrar los resultados encontrados.

Salidas	El sistema web muestra un listado paginado con los resultados de alumnos que coinciden con los criterios de búsqueda.
Restricciones	<ul style="list-style-type: none"> • Solo los usuarios autorizados podrán ejecutar este proceso. • Los criterios de búsqueda serán paginados con 10 registros por página como valor máximo. • Solo puede utilizarse un criterio de búsqueda.

Tabla 7. Requisito funcional 007

Requerimiento funcional	Detalles
Nombre	RF-007 Actualizar alumno
Descripción	El sistema debe permitir modificar los datos de un alumno previamente registrado.
Entradas	El usuario identifica al alumno del que desea actualizar su información y completado el llenado del formulario correctamente.
Procesamiento	El sistema valida la información actualizada y verifica que los datos obligatorios aún estén presentes.

Salidas	El sistema web envía un mensaje de confirmación o de error en base al estado final del proceso realizado.
Restricciones	<ul style="list-style-type: none"> • Solo los usuarios autorizados podrán ejecutar este proceso. • Los campos obligatorios no pueden quedar vacíos o nulos.

Tabla 8. Requisito funcional 008

Requerimiento funcional	Detalles
Nombre	RF-008 Eliminar alumno
Descripción	El sistema debe permitir eliminar o dar de baja un alumno registrado.
Entradas	El usuario identifica al alumno del que desea eliminar su registro.
Procesamiento	El sistema valida que no existan registros que se vean afectados tras la eliminación del alumno.

Salidas	El sistema web envía un mensaje de confirmación o error de acuerdo con el estado final del proceso realizado.
Restricciones	<ul style="list-style-type: none"> • Solo los usuarios autorizados podrán ejecutar este proceso. • No se permitirá la eliminación física de un registro de alumno si éste ya tiene relaciones o dependencias con otros registros en otras tablas de la base de datos.

Tabla 9. Requisito funcional 009

Requerimiento funcional	Detalles
Nombre	RF-009 Registrar alumnos mediante formato CSV
Descripción	El sistema deberá permitir la carga masiva de registros de alumnos a través de un archivo CSV con formato preestablecido.
Entradas	El sistema recibirá como entrada un archivo tipo CSV con los campos preestablecidos. El archivo CSV deberá seguir el formato y estructura especificados en la documentación técnica.

Procesamiento	<p>El módulo validará que la estructura y contenido del archivo CSV cumplan con las especificaciones del formato requerido. Luego, iterará sobre cada línea del archivo extrayendo los campos y registrando una nueva entrada en la base de datos por cada alumno en el CSV.</p>
Salidas	<p>El sistema web enviará un mensaje de confirmación o error de acuerdo con el estado final del proceso realizado.</p>
Restricciones	<ul style="list-style-type: none">• Solo los usuarios autorizados podrán ejecutar este proceso.• La primera fila del CSV siempre deberá incluir los encabezados de campo.• Los campos no deberán contenedor saltos de línea o retornos de carro.• El único formato CSV valido, será el proporcionado por el propio sistema.

3.2.2 Requerimientos no funcionales

RNF-001 | Usabilidad en nuevas interfaces

Las nuevas interfaces incorporadas deberán seguir los estándares de usabilidad definidos para el sistema en cuanto a simplicidad, mensajes guía y flujos de proceso.

RNF-002 | Corrección de errores

Se deberán solucionar los errores detectados en las funcionalidades existentes que afecten la experiencia de usuario.

RNF-003 | Rendimiento

El sistema deberá responder en tiempos adecuados tras la optimización de rendimiento realizada.

RNF-004 | Innovación

Los nuevos procesos introducidos deberán aportar valor a través de ideas y conceptos innovadores.

RNF-005 | Mantenibilidad

El código deberá seguir buenas prácticas que faciliten su lectura y mantenimiento futuro.

RNF-006 | Portabilidad

La actualización no deberá afectar la compatibilidad del sistema con distintos navegadores y dispositivos.

RNF-007 | Seguridad

Se deberá mantener el esquema de autenticación y permisos del sistema.

RNF-008 | Documentación

Se deberá actualizar la documentación técnica y de usuario con los cambios realizados.

3.2.3 Reglas de negocio

RN-001 | Generación de archivos de pago

Se mantendrá la generación diaria de archivos de pago antes de las 7:00 pm con las referencias conciliadas por la institución de cobro.

RN-002 | Compra de servicios

Se mantendrá la restricción de solo permitir servicios de una misma área por compra.

RN-003 | Registro masivo de alumnos

El sistema validará que los campos sean correctos y únicos al cargar registros de alumnos mediante el formato CSV.

RN-004 | Datos de estudiante

Se mantendrá el proceso de llenado automático de “Datos del cliente” con la matrícula de un alumno al realizar una compra.

RN-005 | Acceso a usuarios

Los accesos a módulos del sistema estarán restringidos o permitidos dependiendo el rol del usuario.

RN-006 | Mantenimiento del sistema

Se mantendrá la funcionalidad de inicialización de parámetros por parte del administrador antes de cada ciclo.

RN-007 | Consulta de histórico

La consulta de histórico requerirá seleccionar el intervalo de fechas a consultar previo a la búsqueda.

RN-008 | Reportes en Excel

Los reportes generados en Excel seguirán plantillas predefinidas por tipo de reporte.

3.3 Planificación (Product Backlog y Sprint Backlog)

Siguiendo los principios de la metodología ágil Scrum, la planificación del proyecto del sistema web se realizó mediante el uso de un Product Backlog y los Sprint Backlogs.

El Product Backlog contiene el listado completo de requerimientos y funcionalidades priorizadas para el sistema web resultante del trabajo conjunto entre el Product Owner y el equipo de desarrollo. Cada elemento del Product Backlog o historia de usuario estaba descrita brevemente e incluía detalles como su prioridad, estado, esfuerzo estimado y la iteración asignada.

Identificador (ID) de la Historia	Enunciado de la Historia	Alias	Estado	Dimensión / Esfuerzo	Iteración (Sprint)	Prioridad	Comentarios
HU-001	Como administrador quiero que el sistema permita realizar una administración de los alumnos, las opciones para agregar, leer, actualizar y eliminar los datos generales. Los campos que solicita para crear un nuevo alumno son: matrícula, nombre completo del alumno, correo electrónico, domicilio, ciudad, estado y un botón "Registrar alumno". Para la eliminación de un alumno se selecciona un alumno y aparecerá una ventana con la información general y se elimina con el botón "Eliminar". Para visualizar los alumnos registrados se utiliza una tabla que incluya los campos matrícula, nombre, estado.	Administración de Alumnos	Completado	1 semana	1	Media	El CRUD de alumnos estará en una misma interfaz para la comodidad del usuario.
HU-002	Como administrador quiero que los alumnos de nuevo ingreso se registren al sistema mediante un archivo .csv, para esto se debe tener en cuenta que los alumnos ya registrados no sean afectados a menos que tenga algún cambio en su información.	Registrar alumnos	Completado	1 semana	2	Alta	La información que sea registrada en la base de datos sea nueva o tenga alguna modificación en la información ya registrada.
HU-003	Como administrador quiero que haya un respaldo de todo el sistema del ejercicio fiscal que comprende ese año para después comenzar el nuevo año desde cero, se creara una BD nueva con la toda la información registrada.	Respaldo del sistema	Completado	1 semana	3	Media	La información que se respaldará sea del año anterior en que se uso el sistema
HU-004	Como administrador quiero que haya un registro histórico para que se puedan consultar las referencias de anteriores años teniendo un reporte para poder consultar esta información, de los servicios, las referencias realizadas, etc.	Consultar registro historico	Completado	1 semana	4	Alta	La información consultada sea del año anterior en que se uso el sistema.
HU-005	Como administrador quiero que el sistema incluya una opción para limpiar los datos que hay en el sistema excepto a los usuarios registrados y se inicializara todo el proceso de las referencias.	Limpieza de datos	En proceso	2 semanas	5	Alta	El administrador tendrá acceso a esta opción del sistema y se eliminarán todos los datos del año en el que fue utilizado.

Ilustración 1. Product Backlog.

El Product Owner se encargó de refinar, priorizar y mantener actualizado el Product Backlog a lo largo del proyecto en coordinación con el equipo Scrum.

Para cada sprint, el equipo Scrum seleccionó del Product Backlog las historias de usuario de mayor prioridad que se comprometían a completar en la iteración. Estas

historias conformaron el Sprint Backlog. Al final de cada sprint, el equipo Scrum revisó el Sprint Backlog y volvió a estimar el esfuerzo pendiente para los sprints siguientes.

Identificador (ID) de ítem de product backlog	Enunciado del ítem de Product Backlog	Tarea	Dueño / Voluntario	Estatus	Horas estimadas totales	Día 1		Día 2		Día 3		Día 4		Día 5		Día 6		Día 7		Día 8	
						Cons.	Rest.														
HU-001	Como administrador quiero que el sistema permita realizar una administración de los alumnos, las opciones para agregar, leer, actualizar y eliminar los datos generales. Los campos que solicita para crear un nuevo alumno son: matrícula, nombre completo del alumno, correo electrónico, domicilio, ciudad, estado y un botón "Registrar alumno". Para la eliminación de un alumno se selecciona un alumno y aparecerá una ventana con la información general y se elimina con el botón "Eliminar". Para visualizar los alumnos registrados se utiliza una tabla que incluya los campos matrícula, nombre, estado.	Diseñar las interfaces del CRUD	Javier Ortega	Completado	8	3	5	2	3	1	2	2	2	2	2	2	2	2	2	2	2
		Diseñar el query en SQL para la inserción, actualización, eliminación y lectura de los registros de alumnos	Javier Ortega	Completado	4	2	2	2	0	0	0	0	0	0	0	0	0	0	0	0	0
		Programar las acciones de lectura, insertar, eliminar y actualizar en C#	Javier Ortega	Completado	20	3	17	4	13	3	10	4	6	0	6	3	3	2	1	1	1
		Realizar la conexión con la base de datos para registrar los datos	Javier Ortega	Completado	5	3	2	2	0	0	0	0	0	0	0	0	0	0	0	0	0

Ilustración 2. Sprint Backlog.

De esta manera, el uso de Product y Sprint Backlogs como herramientas de planificación promovió el desarrollo incremental, adaptable y colaborativo del sistema web siguiendo los valores de Scrum.

3.4 Estimación (Recursos y esfuerzo)

Una vez definidos los Sprints Backlogs, se estimó el esfuerzo requerido para desarrollar cada requerimiento utilizando el método de Story Points. Este método se basa en que el equipo Scrum asigna un puntaje a cada elemento del backlog en función de tres criterios: riesgo, cantidad de esfuerzo y complejidad.

Inicialmente, el equipo realizó una primera estimación rápida de los puntos para cada elemento. Posteriormente, se llevó a cabo una reestimación más detallada de los Story Points asignados, con el fin de afinar las mediciones de esfuerzo para el desarrollo de los requerimientos. Este proceso iterativo de estimación y reestimación permitió obtener una evaluación más precisa del trabajo necesario en cada parte del proyecto.

3.5 Costos

A continuación, se presenta una estimación ilustrativa de los costos involucrados en el desarrollo de este proyecto:

Tipo	Descripción	Cantidad	Tiempo	Precio por hora	Total
Hardware	Equipo de computo	1	N/A	N/A	873.50
Software	Windows 10	1	N/A	N/A	1, 000.00
	SQL Server Express 2019	1	N/A	N/A	N/A
	Visual Studio Community 2019	1	N/A	N/A	N/A
	Pencil	1	N/A	N/A	N/A
Personal	SCRUM Team	3	360 horas	100 por hora	108, 000.00
	Asesores del sistema	1	72 horas	75 por hora	5, 400.00
	Tester	1	20 horas	75 por hora	1, 500.00
	Asesor de Ingeniería de Software	1	10 horas	75 por hora	750.00
Total					117, 523.50

Cabe mencionar que, al ser este un proyecto realizado en su totalidad por estudiantes de Ingeniería en Sistemas Computacionales, no representó un costo real para la institución.

Algunas de las licencias de software utilizadas, como Visual Studio Community 2019, SQL Server Express 2019 y Pencil, no generaron costo al tratarse de herramientas libres o de código abierto.

Los precios unitarios considerados se obtuvieron consultando tarifas reales del mercado de desarrollo de software y servicios de programación. Además, se contó con el apoyo de los asesores para estimar dichos costos.

En el caso de los equipos de cómputo, se calculó un costo diario dividiendo el precio entre la vida útil de 5 años y luego multiplicando por los días de trabajo.

Finalmente, para estimar el costo del Scrum Team, asesores y tester, los asesores proporcionaron un precio unitario por hora de trabajo. Con base en esto y las horas trabajadas según el cronograma, se obtuvo el costo total de recursos humanos.

3.6 Cronograma

El siguiente cronograma muestra las fechas de inicio y fin de cada sprint junto con las actividades realizadas en Scrum para la actualización del sistema.

Tabla 10. Cronograma de desarrollo

Nombre	Fecha inicio	Fecha fin
Levantamiento de requerimientos	16/01/2023	20/01/2023
Análisis de desarrollo	23/01/2023	27/01/2023
Diseño de interfaces (Front End)		
Prototipo 1	01/02/2023	06/02/2023
Ejecución		
Historias de usuario	06/02/2023	10/02/2023
Pruebas de aceptación	08/02/2023	10/02/2023
Construcción de interfaces	10/02/2023	14/02/2023
Diseño de base de datos	14/02/2023	17/02/2023
Programación de interfaces (Back End)		
Sprint 1 RF-001 – RF-002		
Desarrollo	17/02/2023	02/03/2023
Pruebas	02/03/2023	03/03/2023

Sprint 2 RF-003		
Desarrollo	07/03/2023	13/03/2023
Pruebas	13/03/2023	14/03/2023
Sprint 3 RF-004		
Desarrollo	15/03/2023	20/03/2023
Pruebas	20/03/2023	22/03/2023
Sprint 4 RF-005 – RF-008		
Desarrollo	23/03/2023	21/04/2023
Pruebas	21/04/2023	25/04/2023
Sprint 5 RF-009		
Desarrollo	26/04/2023	02/05/2023
Pruebas	03/05/2023	04/05/2023
Actualización de base de datos	08/05/2023	08/05/2023
Seguimiento y control		
Pruebas finales	10/05/2023	12/05/2023
Corrección de errores	11/05/2023	16/05/2023
Cierre		
Entrega de sistema	17/05/2023	19/05/2023

3.7 Diseño de interfaces

Para esta etapa de actualización y mejora de interfaces de usuario del sistema, se realizó un rediseño de los prototipos utilizando la herramienta Pencil. Esto permitió explorar y validar distintos diseños centrados en la experiencia de usuario antes de implementarlos.

En total se crearon alrededor de 15 nuevos prototipos de alta fidelidad, incluyendo las interfaces actualizadas y los nuevos módulos desarrollados como el de registro manual de alumnos y consulta de histórico. Esto sentó las bases para la etapa de implementación de los cambios en el código y recursos del sistema.

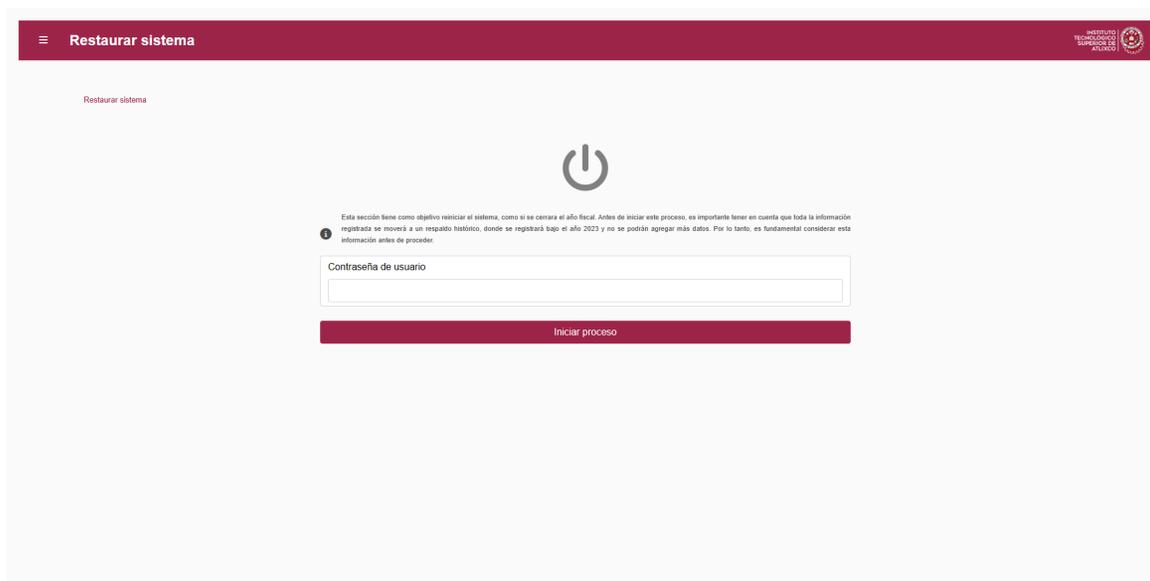


Ilustración 3. Diseño de interfaces

El uso de Pencil facilitó la iteración ágil de diseño en esta fase de rediseño de la experiencia de usuario. Los prototipos digitales creados en esta herramienta

permitieron validar los nuevos diseños de forma simple y sin elevar tiempos de desarrollo.

3.8 Diseño de base de datos

Para la etapa de actualización se mantuvo la estructura general de la base de datos relacional utilizada en la versión inicial del sistema, realizando modificaciones en algunas tablas para soportar nuevas funcionalidades.

Específicamente, se ajustaron campos en la tabla de PerfilesAd, para permitir el acceso a los nuevos módulos como registro manual y recuperación de información histórica.

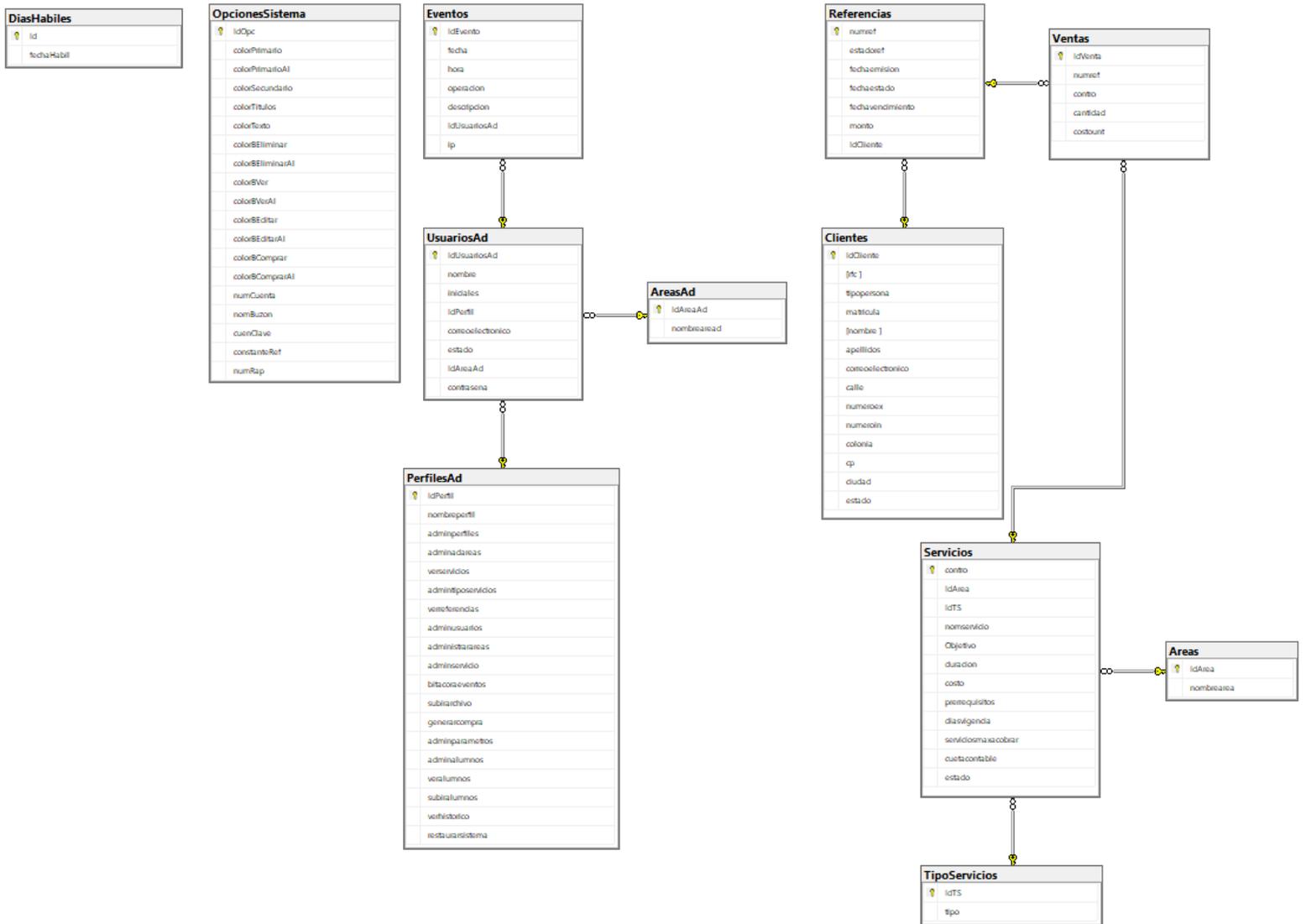


Ilustración 4. Diagrama relacional - BD Principal

Además, se diseñó e implementó una nueva base de datos para almacenar los registros históricos de referencias migrados desde la base de datos principal. Esta nueva base de datos cuenta con 4 tablas que mantienen los datos de referencias de años anteriores, optimizando el rendimiento de la base de datos principal.

Si bien las tablas de la base de datos histórica no tienen relaciones basadas en llaves primarias con las tablas principales, sí existe una relación lógica interpretada por la aplicación para vincular la información durante los procesos de generación de reportes y consultas.



Ilustración 5. Diagrama relacional - BD Histórico

El diseño de estas bases de datos relacionales, con los ajustes y adiciones realizados, buscó proveer una estructura óptima y flexible para soportar los nuevos requerimientos funcionales en el sistema.

3.9 Arquitectura del sistema

3.9.1 Diagrama de Casos de Uso

En este punto, se abordará la representación gráfica y la descripción detallada de los distintos roles y sus interacciones dentro del sistema, a través de los Diagramas de Casos de Uso. Estos diagramas son esenciales para comprender la funcionalidad del sistema y cómo los usuarios interactúan con él para llevar a cabo diversas tareas.

Hasta el momento de este documento, se han identificado tres roles fundamentales en el sistema: Administrador, Administrador de catálogo y Administrador financiero. Estos roles se derivan de los requerimientos recopilados durante la fase de levantamiento de requerimientos. Aunque el sistema tiene la flexibilidad para crear roles adicionales en el futuro, para los fines de este documento, se trabajará con estos tres roles primarios.

3.9.1.1 Administrador

El rol de Administrador posee un nivel de autoridad completo dentro del sistema. Sus responsabilidades abarcan acciones críticas como la creación, actualización y eliminación de elementos diversos, como usuarios, servicios, áreas y tipos de áreas. Además, el Administrador es el responsable de la seguridad y la integridad de los datos almacenados en el sistema, asegurando la confidencialidad y la confiabilidad de la información.

3.9.1.2 Administrador financiero

El Administrador financiero desempeña una función esencial en la gestión financiera del sistema. Este rol se encarga de incorporar los archivos de pago proporcionados por la institución de cobro y consultar el estado de las referencias almacenadas en el sistema. La correcta ejecución de estas acciones garantiza una administración financiera coherente y transparente.

3.9.1.3 Administrador de catálogo

El Administrador de catálogo es responsable de la gestión de servicios, incluyendo tipos de servicios y áreas. Este rol asegura que la estructura y la organización del catálogo sean cohesivas y eficaces, mejorando la navegación y la usabilidad del sistema.

Los Diagramas de Casos de Uso, presentados a continuación, proporcionan una visualización clara de las interacciones entre los roles y el sistema. Estos diagramas ilustran las distintas acciones que cada rol puede llevar a cabo, ayudando a comprender cómo se realizan las tareas y cómo se conectan entre sí.

3.9.2.1 Diagrama de Caso de Uso – General

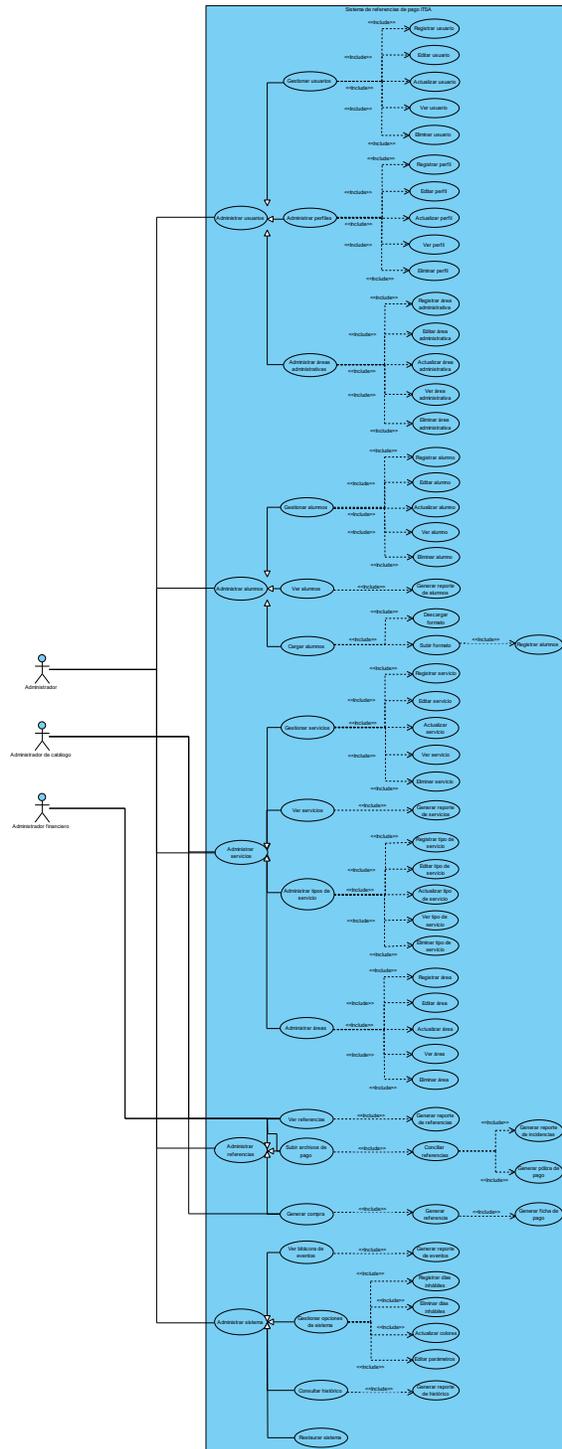


Ilustración 6. Diagrama de Caso de Uso - General

3.9.2.2 Diagrama de Caso de Uso – Administrador

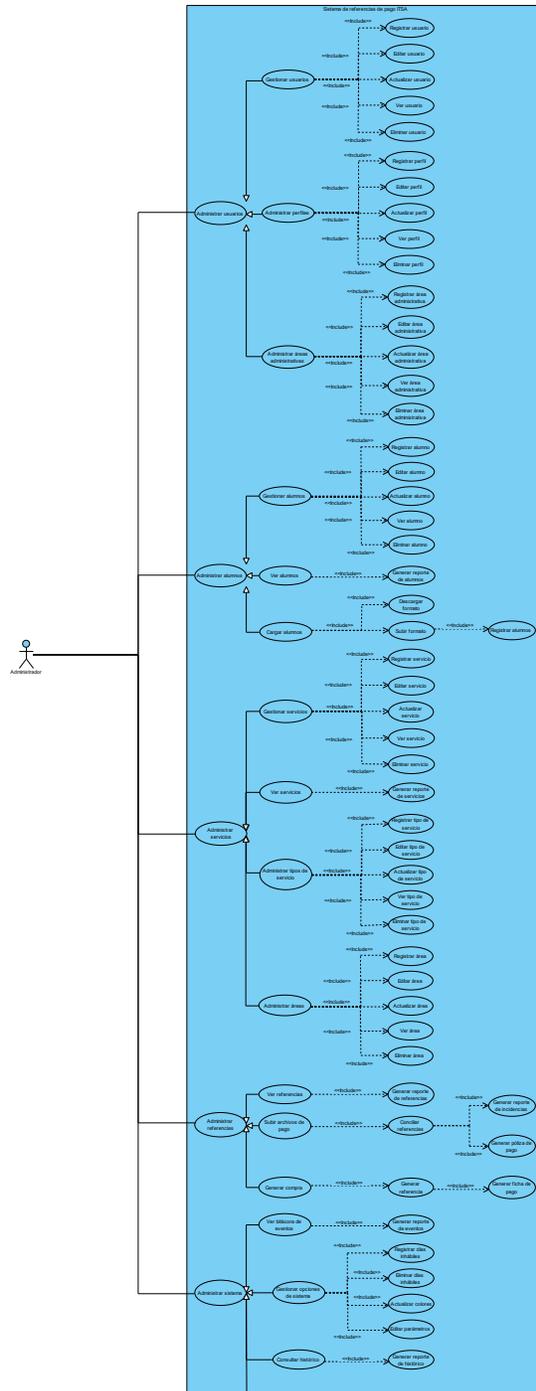


Ilustración 7. Diagrama de Caso de Uso - Administrador

3.9.2.3 Diagrama de Caso de Uso – Administrador financiero

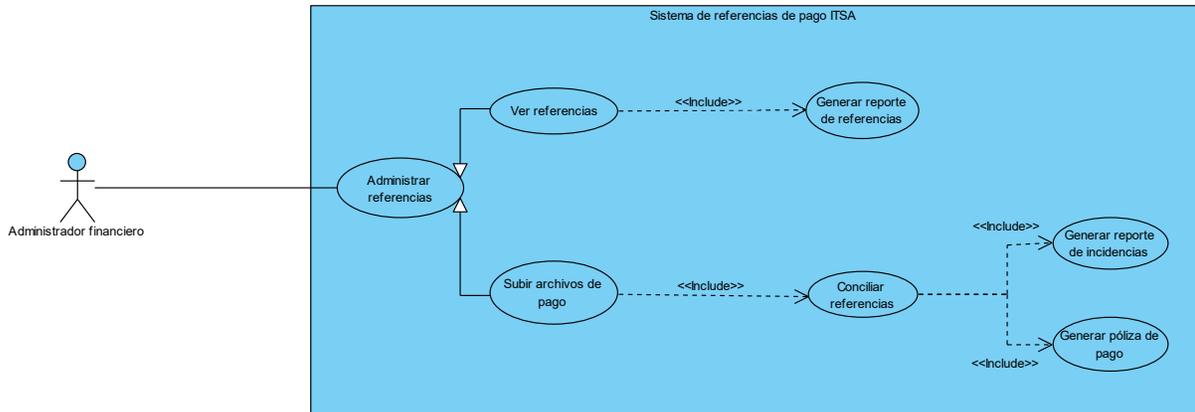


Ilustración 8. Diagrama de Caso de Uso - Administrador financiero

3.9.2.4 Diagrama de Caso de Uso – Administrador de catálogo

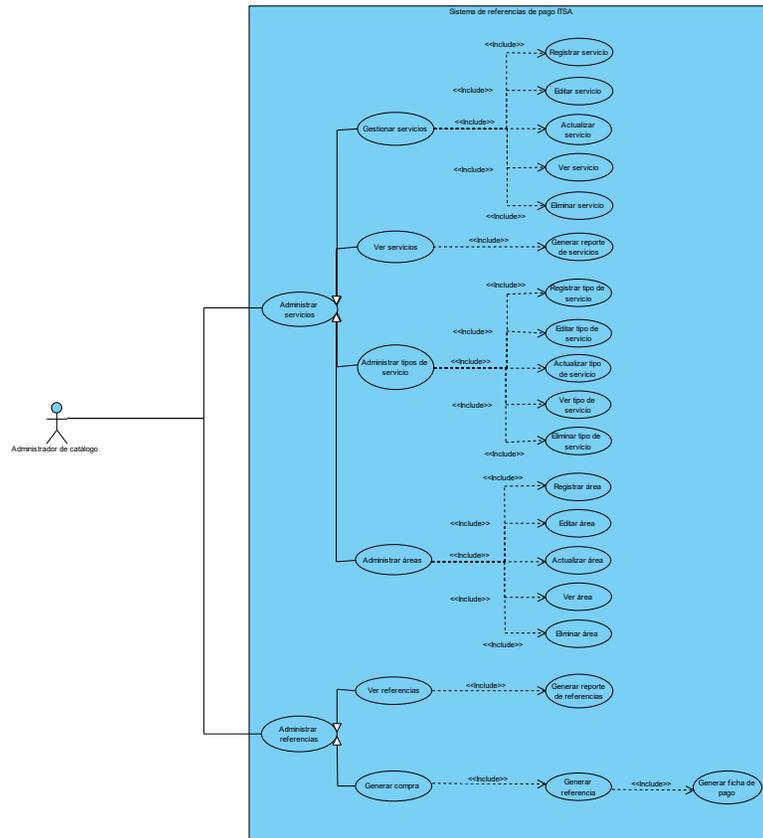


Ilustración 9. Diagrama de Caso de Uso - Administrador de catálogo

3.9.3 Diagrama de navegación

El Diagrama de Navegación constituye un elemento esencial en el diseño y la comprensión de la estructura de navegación dentro del sistema. En esta sección, se presentará una descripción detallada de las pantallas y los flujos de navegación que los usuarios experimentarán al interactuar con el sistema actualizado. Aunque algunos aspectos de la navegación han sido modificados, la base conceptual se ha mantenido para garantizar la continuidad y la familiaridad para los usuarios.

3.9.3.1 Inicio de Sesión y Panel de Inicio

El proceso de inicio del sistema desencadena la presentación de la pantalla de inicio de sesión. Cada usuario deberá iniciar sesión de acuerdo con su perfil asignado. Una vez autenticado, el usuario es dirigido al Panel de Inicio, donde se encuentran los enlaces a diferentes secciones y funciones del sistema.

3.9.3.2 Sección de Usuarios

Dentro de esta sección, se han implementado mejoras para agilizar la administración de usuarios:

1. **Administrar usuarios:** Permite la gestión completa de los usuarios del sistema, incluyendo su creación, edición y eliminación.
2. **Administrar perfiles:** Facilita la asignación de permisos específicos a cada usuario, garantizando un control preciso sobre las funciones disponibles para cada rol.
3. **Administrar áreas administrativas:** Muestra la asignación de cada usuario a una determinada área administrativa, lo que contribuye a una mejor organización y responsabilidad.

3.9.3.3 Sección de Alumnos

La nueva sección "Alumnos" se integra con las siguientes funcionalidades:

1. **Administrar alumnos:** Permite la gestión de información de alumnos, incluyendo su registro, actualización y eliminación.
2. **Ver alumnos:** Proporciona una vista de todos los alumnos registrados en el sistema, mostrando únicamente matrícula y nombre completo para su consulta.
3. **Cargar alumnos:** Permite la carga masiva de datos de alumnos al sistema mediante un formato CSV.

3.9.3.4 Sección de Servicios

La sección de servicios se ha mantenido con las siguientes funcionalidades:

1. **Administrar servicios:** Permite la gestión completa de los servicios, incluyendo la creación, edición y eliminación de registros.
2. **Ver servicios:** Proporciona una vista de todos los servicios previamente registrados.
3. **Administrar áreas:** Permite la gestión eficaz de las áreas involucradas en el manejo de servicios.
4. **Administrar tipos de servicio:** Facilita la gestión de los tipos de servicios y su asignación a categorías.

3.9.3.5 Sección de Referencias

La sección de referencias se ha reorganizado para agilizar las operaciones:

1. **Ver referencias:** Muestra una lista de las referencias generadas en el sistema.
2. **Subir archivo de pago:** Permite a los usuarios subir archivos de pago y llevar a cabo el proceso de conciliación de referencias.
3. **Generar compra:** Facilita la compra de servicios o productos para generar referencias.

3.9.3.6 Sección de Administrar

La sección de administrar ha sido mejorada para ofrecer una experiencia más eficiente:

1. **Bitácora de eventos:** Registra todos los cambios realizados en el sistema, proporcionando un seguimiento detallado de las acciones y modificaciones.
2. **Opciones de sistema:** Permite la configuración de aspectos como el esquema de colores y ciertos parámetros de contabilidad, brindando flexibilidad en la adaptación del sistema a las necesidades específicas.
3. **Consultar histórico:** Consulta las referencias generadas y almacenadas actualmente en la base de datos de histórico.
4. **Restaurar sistema:** Restaura el sistema a un estado inicial y realiza la copia de seguridad de las referencias generadas para reubicarlas en la base de datos de histórico.

El diagrama de navegación ha sido ajustado para reflejar las nuevas secciones y accesos, proporcionando una experiencia de usuario más completa y eficiente. Estas

modificaciones se han implementado con el propósito de mejorar la usabilidad y la funcionalidad del sistema en su conjunto.

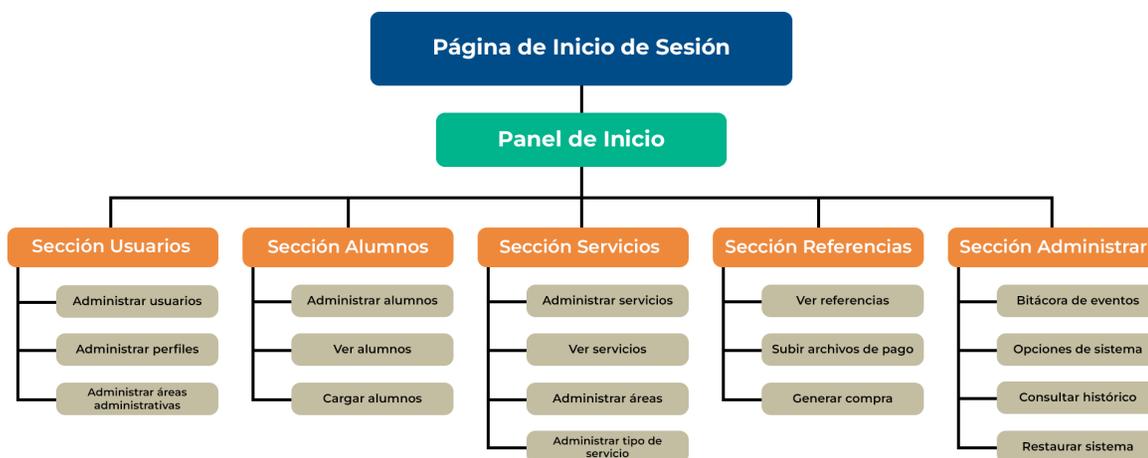


Ilustración 10. Diagrama de navegación

3.9.4 Modelo-Vista-Controlador

A continuación, se presentará el diagrama modelo-vista-controlador que representa la estructura y las relaciones entre los distintos componentes del sistema desarrollado en Visual Studio utilizando el lenguaje de programación C# y la biblioteca ASP.NET Framework en el modelo MVC. Además, se emplea SQL Server Express para la gestión de la base de datos.

El diagrama modelo-vista-controlador es una representación visual que resalta la separación de responsabilidades entre los componentes de la interfaz de usuario (vista), la lógica de negocios (controlador) y el acceso a datos (modelo). Esto proporciona una visión clara de cómo interactúan estos elementos.

El diagrama modelo-vista-controlador completo, que ilustra las relaciones y las interacciones entre los componentes mencionados, se presenta a continuación:

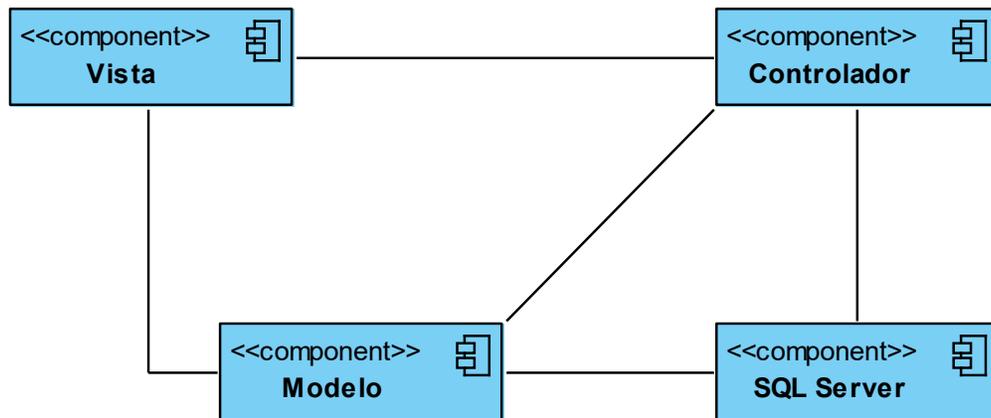


Ilustración 11. Diagrama modelo-vista-controlador

El diagrama modelo-vista-controlador muestra cómo los diferentes componentes trabajan juntos para lograr la funcionalidad y la interacción esperadas dentro del sistema. La Vista se comunica con el Controlador, que a su vez interactúa con el Modelo para acceder y actualizar los datos almacenados en SQL Server Express.

Este enfoque basado en componentes permite una separación clara de las responsabilidades y facilita la escalabilidad, el mantenimiento y la expansión del sistema en el futuro.

Capítulo 4: Desarrollo y pruebas

4.1 Desarrollo

Para la etapa de actualización del backend del sistema web se mantuvo el entorno de desarrollo ASP.NET en Visual Studio 2022, realizando mejoras y ampliaciones en el código para incorporar las nuevas funcionalidades requeridas.

Se implementaron los controladores adicionales necesarios en el backend con la lógica de procesos para los nuevos módulos de registro manual de alumnos y consulta de histórico financiero. Asimismo, se agregaron las vistas o interfaces de usuario asociadas, manteniendo consistencia con el diseño establecido.

El proceso de actualización del backend demandó 5 meses de codificación y pruebas hasta lograr un sistema optimizado, corregido e incorporando nuevas funcionalidades de acuerdo con los objetivos planteados.

4.2 Sprint 1 (RF-001 – RF-002)

RF-001: Restaurar sistema

Se creó un nuevo módulo en el sistema web mediante la implementación del diseño de interfaces, con el propósito de habilitar la restauración del sistema, según los requisitos establecidos. Para dar inicio al proceso de restauración, se utiliza la contraseña del usuario que ha iniciado sesión como clave de activación. Esta medida asegura que el acceso al módulo esté reservado para usuarios con autorización, determinada por su rol en el sistema.

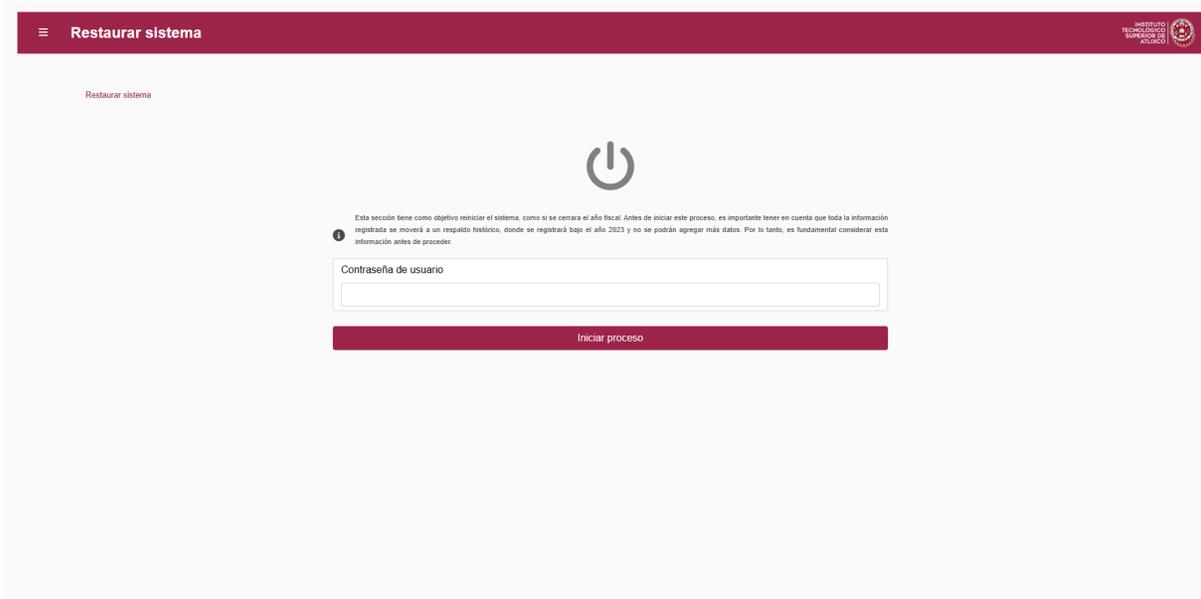


Ilustración 12. Interfaz del módulo "Restaurar sistema".

La contraseña del usuario en sesión se emplea como factor de autenticación, y al ingresarla se confirma su legitimidad para acceder al módulo. La solicitud para iniciar el proceso de restauración se efectúa mediante la activación del botón "Iniciar proceso". Estos pasos de ingreso de contraseña y solicitud de proceso se diseñan como mecanismos de confirmación que el sistema utiliza para discernir el procedimiento a ejecutar.

RF-002: Generar respaldo histórico

El resguardo de referencias en un registro histórico se lleva a cabo de manera simultánea al inicio de un proceso de restauración del sistema por parte del usuario. Una vez que el sistema ha iniciado este proceso, se activa un procedimiento en el cual se realiza una transferencia de información al exportar registros desde la base de datos principal e importarlos a la base de datos histórica. Esta implementación garantiza que

la información se procese de manera segura y sin alteraciones, a diferencia de un proceso de exportación manual. Consultar ilustración 10.

4.3 Sprint 2 (RF-003)

RF-003: Consultar histórico

Se ha desarrollado el módulo "Consultar histórico" con el propósito de brindar a los usuarios con acceso a este módulo la capacidad de revisar todos los respaldos efectuados en el sistema web. El diseño de este módulo garantiza la integridad de los datos al restringir las acciones a lectura únicamente.

Ver	Año	Referencia	Estado	Fecha de emisión	Monto
	2022	220000132186220	CANCELADA	22/8/2022	\$585.00
	2022	220000232186233	CONCILIADA	22/8/2022	\$585.00
	2022	220000332186246	CONCILIADA	22/8/2022	\$585.00
	2022	220000432186259	CONCILIADA	22/8/2022	\$585.00
	2022	220000532186272	CADUCA	22/8/2022	\$585.00
	2022	220000632186285	CONCILIADA	22/8/2022	\$585.00
	2022	220000732186201	CONCILIADA	22/8/2022	\$585.00
	2022	220000832186214	CADUCA	22/8/2022	\$585.00
	2022	220000932186227	CONCILIADA	22/8/2022	\$585.00
	2022	220001032186224	CONCILIADA	22/8/2022	\$585.00

Ilustración 13. Interfaz del módulo "Consultar histórico"

Además de su funcionalidad principal, se ha incorporado una característica adicional que permite la generación de reportes a través de la especificación de parámetros de búsqueda. Estos parámetros incluyen:

- Intervalo de fechas
- Estado de referencia: Emitida, Caduca, Conciliada, Cancelada

Los reportes generados siguen un formato predefinido y son compatibles con la suite de Microsoft Office, en particular, con Excel. Este enfoque asegura que la información presentada sea coherente y facilite su análisis por parte de los usuarios.

The screenshot shows a web application interface for 'Consultar histórico'. At the top, there is a 'Generar reporte' button and a search bar. Below is a table with columns: Ver, Año, Referencia, Estado, Fecha de emisión, and Monto. The table contains several rows of data. A modal window titled 'Generar reporte Excel' is open, allowing users to filter records by 'Estado' (set to 'TODAS') and 'Fecha inicial' and 'Fecha final' (both set to 'dd/mm/aaaa'). There are 'Descargar Excel' and 'Cerrar' buttons in the modal.

Ver	Año	Referencia	Estado	Fecha de emisión	Monto
	2022	220000132186220	CANCELADA	22/8/2022	\$585.00
	2022	220000232186233	CONCILIADA	22/8/2022	\$585.00
	2022				\$585.00
	2022				\$585.00
	2022				\$585.00
	2022				\$585.00
	2022				\$585.00
	2022	220000832186214	CADUCA	22/8/2022	\$585.00
	2022	220000932186227	CONCILIADA	22/8/2022	\$585.00
	2022	220001032186224	CONCILIADA	22/8/2022	\$585.00

Ilustración 14. Funcionalidad de generar reporte en "Consultar histórico"

4.4 Sprint 3 (RF-004)

RF-004: Generar reporte de póliza

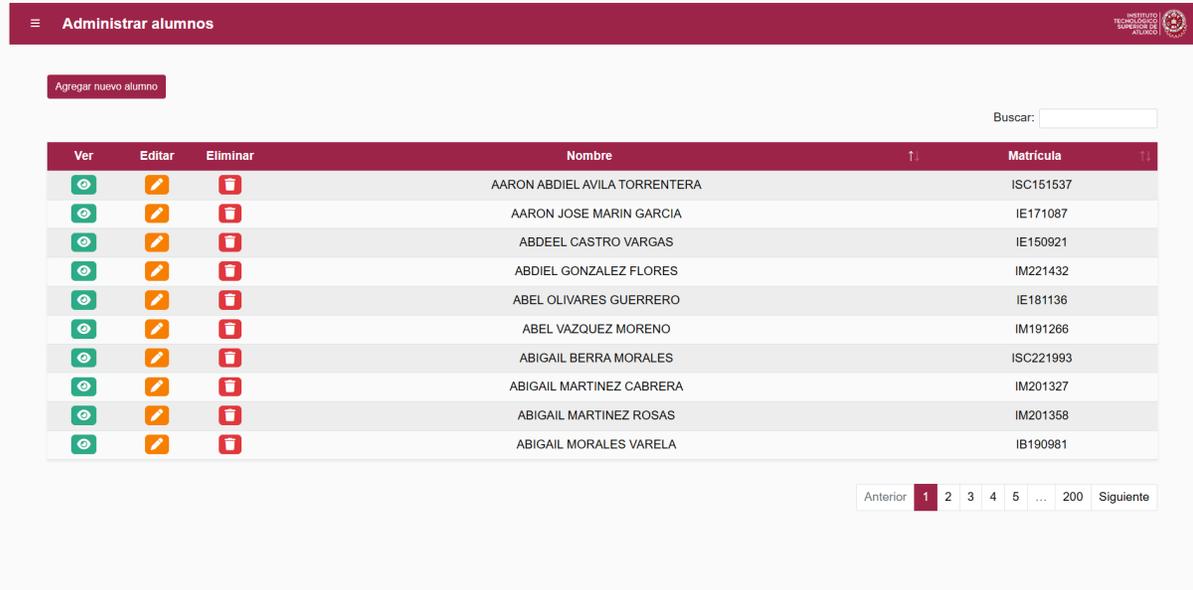
El requisito demandaba una modificación en el diseño del "reporte de póliza de pago" que el sistema genera. Esta actualización implicaba la adaptación de la plantilla estándar, agregando campos en línea con el formato utilizado por el departamento de Finanzas.

Una vez realizada la conciliación de referencias desde el módulo "Subir archivo de pago", el sistema web procederá a crear un informe. Dicho informe estará compuesto por los campos que fueron indicados en el requerimiento. En este proceso, se asegura que los datos se ajusten a las necesidades identificadas y se generen reportes pertinentes y actualizados.

	A	B	C	D
1	SISTEMA AUTOMATIZADO DE ADMINISTRACION Y CONTABILIDAD GUBERNAMENTAL SAACG.NET			
2	FECHA:	01/08/2023		
3	TIPO DE POLIZA:	D		
4	NO. CHEQUE:			
5	CONCEPTO:	Deposito del dia		
6	BENEFICIARIO:	Instituto Tecnológico Superior de Atlixco		
7	SUMAS IGUALES	1461	0	
8	CUENTA	CARGO	ABONO	CONCEPTO POR MOVIMIENTO
9		585		230181334626296
0		876		230281334626296
1		120.5		230381334626296
2				
3				
4				

Ilustración 15. Formato "Reporte de póliza de pago".

4.5 Sprint 4 (RF-005 – RF-008)



The screenshot shows the 'Administrar alumnos' interface. At the top, there is a header with a hamburger menu icon and the text 'Administrar alumnos'. Below the header, there is a button labeled 'Agregar nuevo alumno' and a search box labeled 'Buscar:'. The main content is a table with the following columns: 'Ver', 'Editar', 'Eliminar', 'Nombre', and 'Matricula'. The table contains 10 rows of student records. At the bottom right, there is a pagination control showing 'Anterior', '1', '2', '3', '4', '5', '...', '200', and 'Siguiente'.

Ver	Editar	Eliminar	Nombre	Matricula
			AARON ABDIEL AVILA TORRENTERA	ISC151537
			AARON JOSE MARIN GARCIA	IE171087
			ABDEEL CASTRO VARGAS	IE150921
			ABDIEL GONZALEZ FLORES	IM221432
			ABEL OLIVARES GUERRERO	IE181136
			ABEL VAZQUEZ MORENO	IM191266
			ABIGAIL BERRA MORALES	ISC221993
			ABIGAIL MARTINEZ CABRERA	IM201327
			ABIGAIL MARTINEZ ROSAS	IM201358
			ABIGAIL MORALES VARELA	IB190981

Ilustración 16. Interfaz del módulo "Administrar alumnos".

RF-005: Crear alumno

Se desarrolló el módulo "Administrar alumnos" en el sistema web con el fin de facilitar las operaciones CRUD relacionadas con los datos de los estudiantes en la base de datos (Ver Ilustración 14). La funcionalidad inicial de este módulo permite llevar a cabo el registro manual de los alumnos.

The screenshot shows a web interface for managing students. At the top, there is a navigation bar with the text 'Administrar alumnos' and a logo for 'INSTITUTO TECNOLÓGICO SUPERIOR DE PUERTO RICO'. Below the navigation bar, there is a button labeled 'Agregar nuevo alumno'. The main content area is divided into two parts: a table of existing students and a modal form for adding a new student.

The table of existing students has the following data:

Matricula
ISC151537
IE171087
IE150921
IM221432
IE181136
IM191266
ISC221993
IM201327
IM201358
IB190981

The 'Agregar nuevo alumno' form contains the following fields:

- Nombre (Name)
- Apellidos (Surnames)
- Matricula (Enrollment Number)
- Correo electrónico (Email)
- Calle (Street)
- Número exterior (External Number)
- Número interior (Internal Number)
- Colonia (Neighborhood)
- Código postal (Postal Code)
- Ciudad (City)
- Estado (State)

At the bottom of the form, there are two buttons: 'Añadir nuevo alumno' and 'Cerrar'.

Ilustración 17. Formulario "Agregar nuevo alumno".

Para llevar a cabo este registro, se emplea un formulario específicamente diseñado. Este formulario se adapta a los parámetros esenciales y sigue el formato adecuado en cada campo, conforme a los estándares de ingreso utilizados en el sistema web para la tabla de Alumnos. Este enfoque garantiza que la información se ingrese de manera consistente y se alinee con las convenciones establecidas en el sistema, contribuyendo así a la integridad y calidad de los datos almacenados.

RF-006: Consultar alumno

La interfaz del módulo "Administrar alumnos" (Ver Ilustración 14) está diseñada para presentar, al iniciar su carga, una tabla con los registros paginados de todos los alumnos que han sido registrados en el sistema. Esta tabla enriquece la capacidad de consulta de los alumnos mediante la implementación de un filtro de búsqueda. Dicho filtro se basa en los siguientes campos:

- Nombre completo.
- Matrícula.

De manera similar, se ha desarrollado el módulo "Ver alumnos". Este módulo tiene como finalidad atender a aquellos usuarios con acceso restringido que necesiten la posibilidad de acceder a la información de los alumnos para fines de generación de referencias. La presentación de la información se limita únicamente al nombre completo del alumno y a la matrícula estudiantil asignada. Con esta estrategia, se garantiza que los usuarios autorizados puedan acceder a la información relevante sin exponer datos confidenciales innecesarios.

Ver Alumnos

Buscar:

Nombre	Matrícula
AARON ABDIEL AVILA TORRENTERA	ISC151537
AARON JOSE MARIN GARCIA	IE171087
ABDEEL CASTRO VARGAS	IE150921
ABDIEL GONZALEZ FLORES	IM221432
ABEL OLIVARES GUERRERO	IE181136
ABEL VAZQUEZ MORENO	IM191266
ABIGAIL BERRA MORALES	ISC221993
ABIGAIL MARTINEZ CABRERA	IM201327
ABIGAIL MARTINEZ ROSAS	IM201358
ABIGAIL MORALES VARELA	IB190981

Anterior 1 2 3 4 5 ... 200 Siguiente

Ilustración 18. Interfaz del módulo "Ver alumnos".

RF-007: Actualizar alumno

Dentro del módulo "Administrar alumnos", a través de la tabla que presenta el listado de estudiantes, se ha implementado un botón que habilita la actualización de los datos del alumno seleccionado (consulte Ilustración 14). Al hacer clic en este botón, se despliega un formulario que muestra los detalles susceptibles de modificación. Este formulario se configura de tal manera que solo exhiba la información que es sujeta de actualización, agilizando así el proceso.

Una vez que el usuario completa la actualización, el sistema procede a evaluar la integridad de los datos modificados y a prevenir la posibilidad de duplicación. En caso de que los datos actualizados puedan afectar la coherencia de la información o resultar en duplicados, el sistema detiene la actualización y presenta un mensaje de error. Por otro lado, si no se detectan problemas, el sistema confirma la actualización de la información del alumno. Este proceso asegura que los datos se mantengan confiables y coherentes a lo largo del sistema.

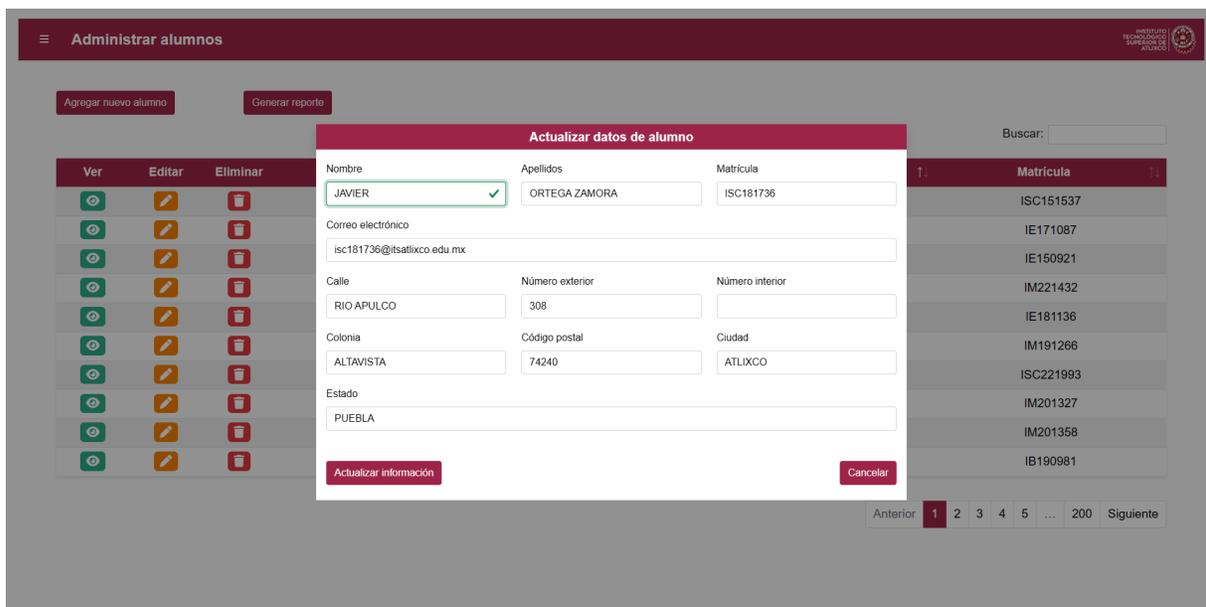


Ilustración 19. Formulario "Actualizar datos del alumno".

RF-008: Eliminar alumno

La funcionalidad final disponible desde la interfaz "Administrar alumnos" es la capacidad de eliminar registros, siempre y cuando estos registros no estén relacionados con datos en otras tablas de la base de datos (Ver Ilustración 14). En situaciones en las que existe tal relación, el sistema detiene el proceso de eliminación para preservar la integridad de la información.

Para llevar a cabo esta acción, simplemente se requiere presionar el botón de eliminación correspondiente al alumno en cuestión. Este botón es visible en la tabla paginada que lista a los alumnos. Después de esta acción, el sistema desencadena un mensaje de confirmación antes de iniciar el proceso de eliminación.

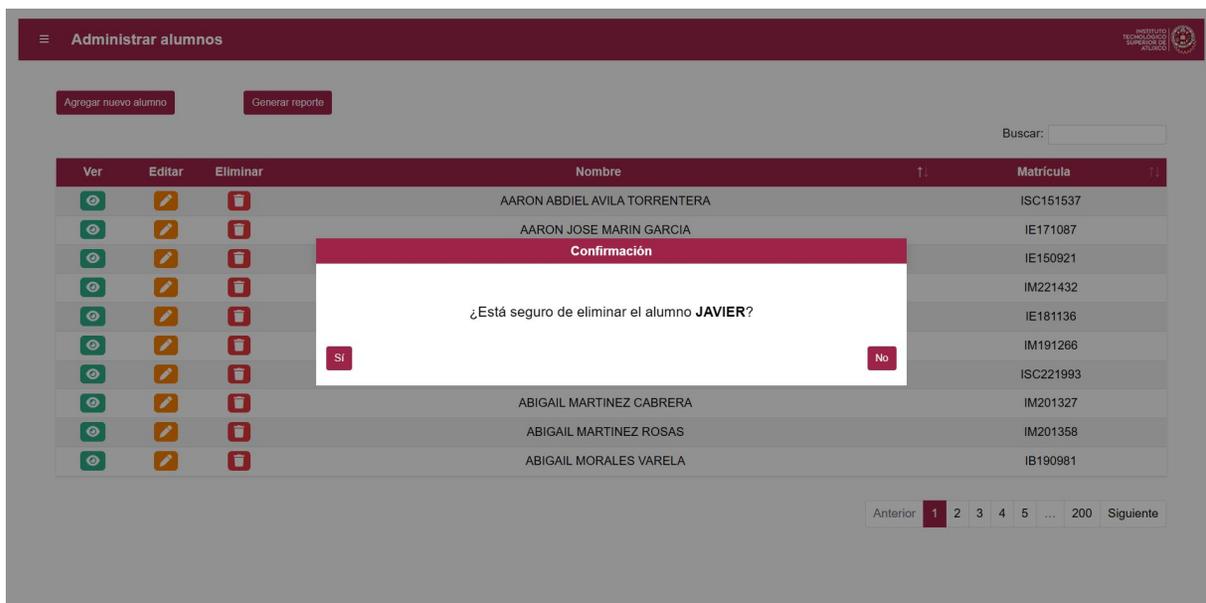


Ilustración 20. Proceso de eliminación de alumno.

Este proceso de eliminación es diseñado con precaución para asegurar que los datos se manejen de manera responsable y que la información valiosa no se pierda de manera accidental.

4.6 Sprint 5 (RF-009)

RF-009: Carga masiva CSV

Esta característica de alta prioridad se encuentra implementada a través del módulo "Cargar alumnos". Este módulo proporciona un formulario destinado a la carga de archivos en formato CSV. Además, el formulario ofrece la opción de descargar una plantilla específica diseñada para facilitar el llenado con los registros de nuevos alumnos.

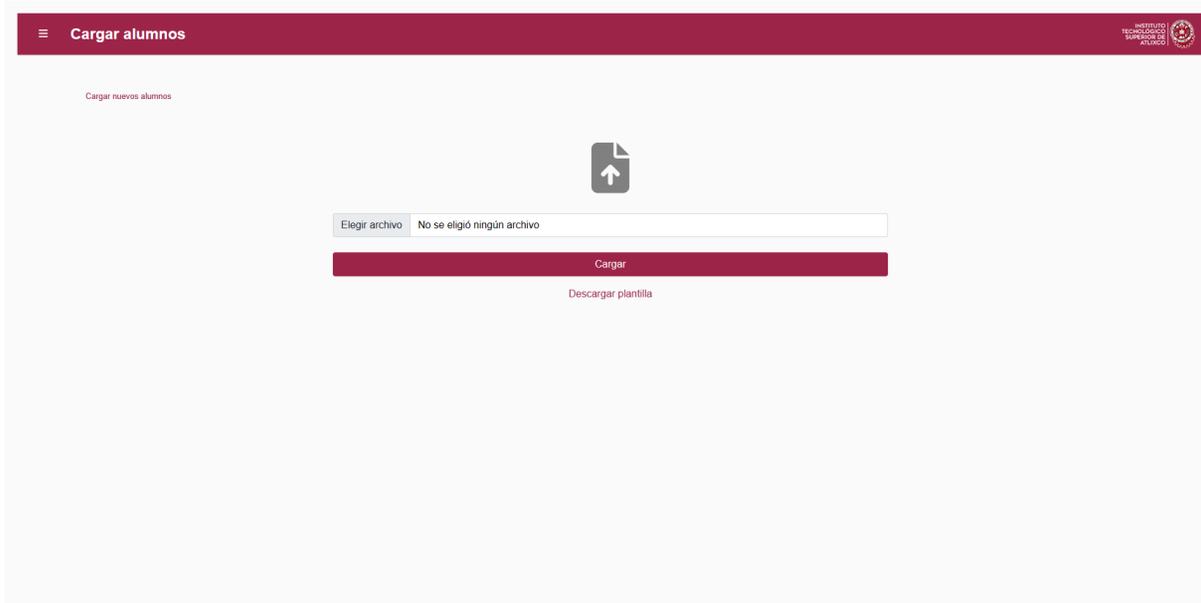


Ilustración 21. Interfaz del módulo "Cargar alumnos".

La plantilla se ha diseñado con columnas organizadas de manera coherente para garantizar la correcta interpretación por parte del sistema. De esta forma, el sistema puede validar cada registro de manera precisa antes de añadirlo a la tabla de alumnos.

La incorporación de esta funcionalidad optimiza el proceso de registro de alumnos al inicio de cada ciclo escolar. Permite que cualquier usuario con acceso a este módulo realice la tarea con facilidad y en un tiempo significativamente menor en comparación con el registro manual de cada alumno desde el módulo "Administrar alumnos". Este enfoque se traduce en una mayor eficiencia y una reducción del tiempo dedicado a tareas administrativas repetitivas.

	A	B	C	D	E	F	G	H	I	J	K
1	Matrícula	Nombre	Apellidos	Correo	Calle	Número exte	Número inter	Colonia	CP	Ciudad	Estado
2											
3											
4											
5											
6											
7											
8											
9											
10											
11											
12											
13											
14											
15											
16											
17											
18											
19											
20											
21											
22											
23											
24											
25											

Ilustración 22. Formato "Cargar alumnos".

4.7 Pruebas

Luego de finalizar la etapa de desarrollo de la actualización, se ejecutó un proceso completo de pruebas sobre el sistema antes de su puesta en producción. Estas pruebas buscaron validar que los nuevos componentes y funcionalidades se ejecuten de acuerdo con los requerimientos y sin errores.

Se realizaron pruebas unitarias sobre cada unidad de código nueva o modificada, garantizando su funcionamiento independiente. También se llevaron a cabo pruebas de integración y de sistema para validar el comportamiento conjunto de los módulos actualizados.

Las pruebas de aceptación ejecutadas por parte de los usuarios resultaron fundamentales para verificar que el sistema actualizado cubre de forma correcta los procesos involucrados. Durante esta etapa se detectaron bugs y oportunidades de mejora, las cuales fueron incorporadas antes del despliegue final [5].

En total, el ciclo de pruebas sobre la actualización del sistema demandó 1 semana de trabajo para garantizar su calidad antes del lanzamiento.

4.8 Seguimiento y control

Durante la etapa de desarrollo de la actualización se establecieron mecanismos de seguimiento y control, tanto para el avance de las actividades como para la calidad del producto final.

Se utilizó la metodología Scrum para dar seguimiento continuo al progreso mediante reuniones periódicas de retrospectiva y revisión del Product Backlog.

Para el aseguramiento de calidad se efectuaron rigurosas pruebas sobre los nuevos módulos y cambios realizados, incluyendo pruebas unitarias, de integración, de sistema y de aceptación. Esto permitió identificar y solventar problemas antes del despliegue de la actualización.

Entre los componentes y funcionalidades críticas verificadas se encuentran: el módulo de registro manual de alumnos, la consulta de histórico financiero, la generación de reportes y el proceso de respaldo de información histórica, detallados en los requerimientos.

Capítulo 5: Resultados

5.1 Discusión de resultados

Los resultados obtenidos en este proyecto demuestran que la integración del módulo para la actualización de catálogos por parte del administrador optimizó los procesos relacionados con el control y manejo de la información del sistema web de ingresos propios.

Anteriormente, la actualización de los catálogos dependía completamente de la recepción de la información por parte de terceros, generalmente con retrasos en la entrega que provocaban desface en los datos y dificultades para el usuario administrador. Con la implementación de este nuevo módulo, el administrador tiene ahora la libertad de ingresar y modificar los datos de los catálogos en el momento que lo requiera, sin tener que esperar la entrega por un externo.

Proceso de carga de alumnos (por horas)

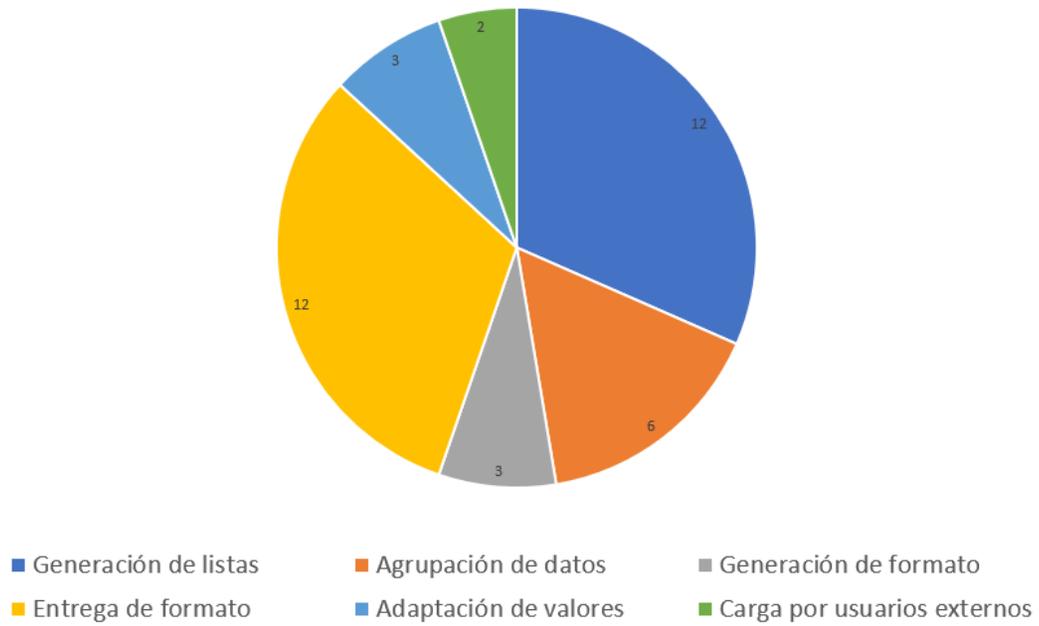


Ilustración 23. Proceso de carga de alumnos antes de la actualización.

Proceso de carga de alumnos (por horas)

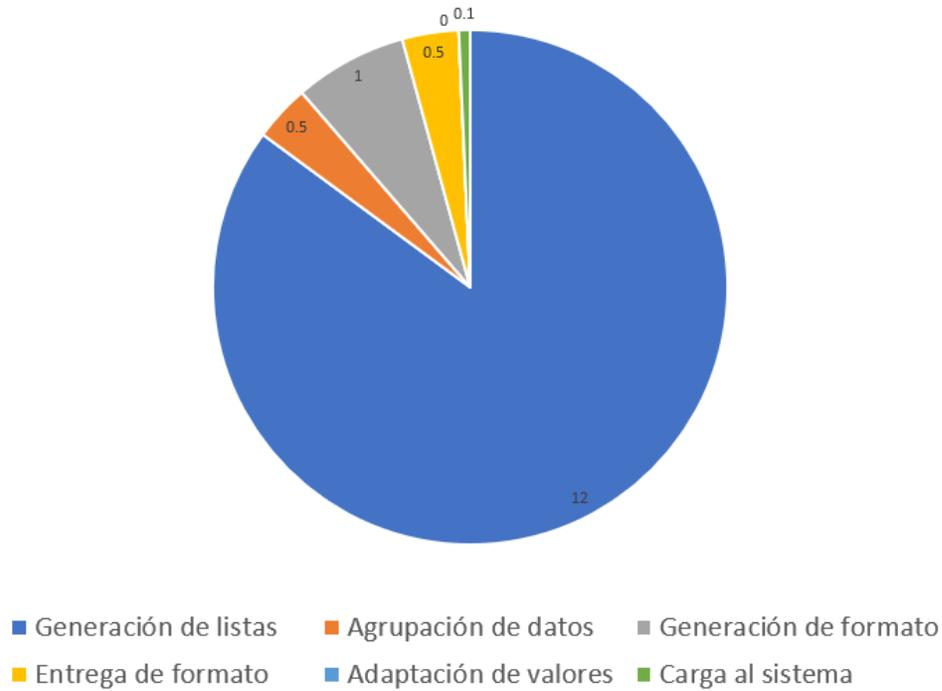


Ilustración 24. Proceso de carga de alumnos después de la actualización.

Esta funcionalidad no solo optimizó los tiempos y procesos internos, sino que también mejoró el control y protección de los datos sensibles de ingresos y pagos al evitar retrasos e inconsistencias. Además, dotó al sistema web de mayor autonomía y flexibilidad para adaptarse a las necesidades cambiantes de la institución.

5.2 Conclusiones del proyecto

En conclusión, los resultados confirman que la hipótesis planteada al inicio de este proyecto fue correcta. La utilidad y eficiencia del sistema de información de ingresos propios del Instituto Tecnológico Superior de Atlixco efectivamente se incrementó al otorgar libertad al administrador para actualizar los catálogos sin depender de terceros. Asimismo, se cumplió satisfactoriamente el objetivo general de actualizar el sistema web mediante la optimización de procesos, corrección de limitaciones y adición de nueva funcionalidad para mejorar el control y manejo de datos.

Mi experiencia en el desarrollo de este proyecto fue muy enriquecedora, ya que pude aplicar y ampliar los conocimientos adquiridos durante la carrera, además de enfrentar retos interesantes durante la implementación que me permitieron crecer profesionalmente. El resultado final me deja muy satisfecho, al comprobar que la solución propuesta efectivamente optimizó el sistema y cumplió con los objetivos planteados inicialmente.

6. Referencias

[1] A. Djandrw. “¿Qué es Scrum”? Medium. Accedido el 10 de septiembre de 2023. [En línea]. Disponible: <https://medium.com/@andrewdjandrw/qu%C3%A9-es-scrum-674c6b791af4>

[2] Microsoft. “Visual Studio: IDE y Editor de código para desarrolladores de software y Teams”. Visual Studio. Accedido el 01 de septiembre de 2023. [En línea]. Disponible: <https://visualstudio.microsoft.com/es/>

[3] M. López Sanz. “Programación web en el entorno servidor”. RA-MA Editorial. Accedido el 08 de septiembre de 2023. [En línea]. Disponible: <https://elibro.net/es/lc/itsatlixco/titulos/106487>

[4] G. Martínez. “¿Qué es un servidor web y para qué sirve”? Webempresa. Accedido el 10 de septiembre de 2023. [En línea]. Disponible: <https://www.webempresa.com/hosting/que-es-servidor-web.html>

[5] R. Pressman. “Ingeniería del software: un enfoque práctico (8va edición)”. McGraw Hill. Accedido el 11 de septiembre de 2023. [En línea]. Disponible: https://www.academia.edu/download/58273346/IS1_Cap3_Desarrollo_Agil.pdf

[6] S. Luján-Mora. “Programación de aplicaciones web: Clientes web y AJAX”. Club Universitario. Accedido el 11 de septiembre de 2023. [En línea]. Disponible: <https://sergiolujanmora.es/verpdf/42>

[7] K.S. Rubin y P. Rubin. “Continuous Delivery with Visual Studio ALM 2015”. Apress. Accedido el 10 de septiembre de 2023. [En línea]. Disponible: <https://github.com/apress/continuous-delivery-w-visual-studio-alm-2015>

[8] K. Schwaber y J. Sutherland. “La Guía Definitiva de Scrum: La Reglas del Juego”. Scrum.org. Accedido el 11 de septiembre de 2023. [En línea]. Disponible: <https://scrumguides.org/docs/scrumguide/v2020/2020-Scrum-Guide-Spanish-European.pdf>

[9] K. Beck. “Test Driven Development: By Example”. Addison-Wesley Professional. Accedido el 12 de septiembre de 2023. [En línea]. Disponible: [https://github.com/clarabez/SoftwareTestingBooks/blob/master/Test-Driven%20Development%20By%20Example%20\(Kent%20Beck\).pdf](https://github.com/clarabez/SoftwareTestingBooks/blob/master/Test-Driven%20Development%20By%20Example%20(Kent%20Beck).pdf)

[10] M.Y. Jiménez Capel. “Bases de datos relacionales y modelado de datos”. IC Editorial. Accedido el 12 de septiembre de 2023. [En línea]. Disponible: <https://elibro.net/es/lc/itsatlixco/titulos/44139>

[11] J.L. Cisneros. “Panorama sobre bases de datos. Un enfoque práctico”. Universidad Autónoma de Baja California. Accedido el 12 de septiembre de 2023. [En línea]. Disponible: <https://books.google.com.mx/books?id=UmLBLPGniSoC>

[12] S. P. José Manuel. “Pruebas de Software. Fundamentos y Técnicas”. Universidad Politécnica de Madrid. Accedido el 13 de septiembre de 2023. [En línea]. Disponible: https://oa.upm.es/40012/1/PFC_JOSE_MANUEL_SANCHEZ_PENO_3.pdf

[13] E.G. Maida y J. Pacienza. “Metodologías de desarrollo de software”. Universidad Católica Argentina. Accedido el 13 de septiembre de 2023. [En línea]. Disponible: <https://repositorio.uca.edu.ar/bitstream/123456789/522/1/metodologias-desarrollo-software.pdf>

[14] Learn Microsoft. “¿Qué es el control de versiones?”. Microsoft. Accedido el 15 de septiembre de 2023. [En línea]. Disponible: <https://learn.microsoft.com/es-es/devops/develop/git/what-is-version-control>