

Universidad de La Habana
Facultad de Matemática y Computación



Sistema digital para la gestión de asignaturas electivas en la Universidad de La Habana.

Autor:

Alex Samuel Bas Beovides

Tutores:

Lic. Carlos León González

Dra. C. Lucina García Hernández

Trabajo de Diploma
presentado en opción al título de
Licenciado en Ciencia de la Computación

31 de mayo de 2025

<https://github.com/thesis-si/course-allocation/tree/main/Electia>

Resumen

El presente trabajo aborda el diseño y desarrollo de un sistema digital para la gestión de asignaturas electivas universitarias, respondiendo a la necesidad de una herramienta institucional que facilite la inscripción, aprobación y administración integral del proceso académico en la Universidad de La Habana. El sistema propuesto se centra en la creación de un entorno digital que permita a profesores, administradores y estudiantes gestionar el ciclo completo de las asignaturas electivas de manera eficiente, desde la propuesta docente hasta la matrícula estudiantil y el seguimiento académico. Se discuten los desafíos y limitaciones de los sistemas existentes de gestión académica, destacando la importancia de una solución adaptada al contexto universitario cubano que integre las particularidades organizativas de la Dirección de Formación de Pregrado.

El desarrollo del sistema se basa en una arquitectura de software robusta y escalable, utilizando tecnologías como ASP.NET Core, React.js, SQL Server y Docker para garantizar su sostenibilidad y rendimiento. En este trabajo se detallan los componentes principales del sistema, incluyendo el diseño de la base de datos relacional, la interfaz de usuario diferenciada por roles y los servicios de gestión de propuestas académicas. Se presentan los resultados de pruebas y validación experimental que demuestran la funcionalidad y eficiencia del sistema en la automatización de procesos administrativos y la generación de documentación académica oficial. Finalmente, se discuten las implicaciones de este trabajo para la comunidad académica universitaria y se proponen futuras líneas de desarrollo que permitan extender la funcionalidad a otros procesos de gestión educativa.

Abstract

This work addresses the design and development of a digital system for managing university elective courses, responding to the need for an institutional tool that facilitates enrollment, approval, and comprehensive administration of the academic process at the *Universidad de La Habana* (University of Havana). The proposed system focuses on the creation of a digital environment that enables professors, administrators, and students to efficiently manage the entire lifecycle of elective courses—from course proposal to student registration and academic monitoring. The challenges and limitations of existing academic management systems are discussed, highlighting the importance of a solution adapted to the Cuban university context and integrating the organizational particularities of the *Dirección de Formación de Pregrado* (Undergraduate Education Office).

The system's development is based on a robust and scalable software architecture, employing technologies such as ASP.NET Core, React.js, SQL Server, and Docker to ensure sustainability and performance. This work details the system's main components, including the design of the relational database, a role-based user interface, and services for managing academic proposals. Test results and experimental validation are presented, demonstrating the system's functionality and efficiency in automating administrative processes and generating official academic documentation. Finally, the implications of this work for the academic university community are discussed, and future lines of development are proposed to extend the system's functionality to other educational management processes

Índice general

Introducción	1
1. Marco teórico-conceptual	4
1.1. Ingeniería de software aplicada a la gestión de procesos académicos .	4
1.1.1. Arquitectura de Software	5
2. Concepción y diseño de la solución	7
2.1. Problemática	7
2.2. Determinación de los Requerimientos	8
2.2.1. Requerimientos Funcionales	9
2.2.2. Requerimientos No Funcionales	9
2.2.3. Requerimientos de Entorno	10
2.3. Casos de uso y flujos principales	11
2.4. Diseño de la base de datos	15
3. Detalles de Implementación y Experimentación	17
3.1. Tecnologías y herramientas utilizadas para el desarrollo del sistema .	17
3.1.1. Sistema de Gestión de Base de Datos: SQL Server	18
3.1.2. Framework de Desarrollo Backend: ASP.NET Core	18
3.1.3. Mapeo Objeto-Relacional: Entity Framework Core	19
3.1.4. Tecnología de Desarrollo Frontend: React.js	19
3.1.5. Plataforma de Contenedorización: Docker	19
3.2. Arquitectura del sistema implementado	20
3.3. Pruebas y validación experimental	20

Índice de figuras

2.1.	Diagrama de casos de uso del Administrador.	12
2.2.	Diagrama de casos de uso del Profesor.	13
2.3.	Diagrama de casos de uso del Estudiante.	14
2.4.	Modelo Entidad-Relacional Extendido de la base de datos.	16

Introducción

La gestión eficiente de las asignaturas electivas en entornos universitarios constituye un desafío clave para garantizar la calidad del proceso formativo, la equidad en el acceso a los cursos y el cumplimiento de los objetivos estratégicos de cada institución. Actualmente, las Direcciones de Formación de Pregrado requieren sistemas que no solo faciliten la inscripción y aprobación de asignaturas por parte del profesorado, sino que además aseguren una adecuada organización, control y seguimiento de la matrícula estudiantil en correspondencia con criterios académicos definidos. Esta necesidad se hace más crítica en contextos donde la oferta electiva es diversa y debe ser gestionada por roles específicos, como administradores, profesores y estudiantes, cada uno con distintos niveles de acceso y funciones.

A nivel internacional, los Sistemas de Gestión del Aprendizaje (LMS, por sus siglas en inglés), como Moodle y Canvas, son ampliamente utilizados para administrar cursos y contenidos educativos en línea. Estas plataformas ofrecen funcionalidades robustas en cuanto a la creación de cursos, la gestión de usuarios y el seguimiento del desempeño académico. Sin embargo, presentan limitaciones cuando se trata de adaptar sus funcionalidades a necesidades particulares de gestión institucional. Por ejemplo, muchos de estos sistemas requieren suscripciones pagas para acceder a módulos avanzados o a servicios personalizados, y no siempre permiten configurar reglas específicas para la asignación automatizada de cursos, tales como restricciones por año académico, facultad o carrera. Estas carencias impiden su implementación efectiva en escenarios que demandan alta personalización y autonomía.

Ante las limitaciones identificadas en los sistemas actuales de inscripción y administración de asignaturas electivas, el propósito central de esta tesis es el diseño y desarrollo de una plataforma digital que permita gestionar de forma integral el proceso de inscripción, aprobación, asignación y matrícula de asignaturas electivas en el ámbito universitario cubano. Este sistema busca dar respuesta a diversas problemáticas institucionales que dificultan la organización eficiente de estas ofertas docentes, entre las que destacan:

1. La inexistencia de una herramienta digital propia que responda a las particularidades organizativas y normativas de la Dirección de Formación de Pregrado.

2. La dispersión del proceso de inscripción y validación de asignaturas electivas, que actualmente depende de métodos manuales o no integrados.
3. La dificultad para aplicar criterios diferenciados de acceso según carrera, año académico u otros parámetros establecidos por la institución.
4. La ausencia de mecanismos automáticos para el seguimiento de la matrícula, control de cupos, generación de actas y obtención de estadísticas útiles para la toma de decisiones.

Como solución, se propone el desarrollo de un sistema web de inscripción y gestión de asignaturas electivas, accesible para todos los actores del proceso educativo, que garantice la transparencia, trazabilidad y eficiencia de cada etapa. Esta solución estará basada en una arquitectura modular y una base de datos relacional, permitiendo escalabilidad, control de acceso según rol, y adaptabilidad a futuros requerimientos institucionales. Para cumplir este propósito general, se plantean los siguientes objetivos específicos:

1. Diseñar una interfaz para que el profesorado proponga nuevas asignaturas electivas, incluyendo todos los datos requeridos por la DFP para su evaluación y aprobación.
2. Implementar un módulo de revisión y validación de asignaturas por parte del rol administrador, permitiendo definir criterios y filtros personalizables.
3. Desarrollar funcionalidades de inscripción estudiantil que respeten la capacidad máxima de cada curso y permitan aplicar reglas específicas por carrera, facultad o año académico.
4. Automatizar la asignación de usuarios, generación de listados, actas académicas y reportes estadísticos, minimizando errores y tareas manuales.
5. Evaluar el desempeño del sistema mediante pruebas funcionales y de usabilidad, con el fin de asegurar su efectividad y aceptación por parte de los usuarios finales.

La presente investigación se organiza en cuatro capítulos, cada uno enfocado en una etapa clave del proceso de desarrollo del sistema propuesto:

- **Capítulo 1: Marco teórico-conceptual.** Se examinan los principios teóricos relacionados con la gestión académica de asignaturas electivas en entornos universitarios, así como la caracterización de plataformas similares, sus funcionalidades, beneficios y limitaciones, con especial atención a las necesidades del contexto cubano.

- **Capítulo 2: Concepción y diseño de la solución.** Se expone la propuesta del sistema informático, especificando los requerimientos funcionales y no funcionales, los roles de usuario, los principales casos de uso y la estructura de datos necesaria para su implementación.
- **Capítulo 3: : Implementación y experimentación.** Se describen los detalles del proceso de implementación, incluyendo las tecnologías utilizadas, la arquitectura del sistema, y los resultados obtenidos a partir de pruebas de funcionamiento, eficiencia y experiencia de usuario.

Finalmente, se incluye la bibliografía consultada, que respalda los contenidos desarrollados y las decisiones de diseño adoptadas a lo largo de la investigación.

Capítulo 1

Marco teórico-conceptual

El presente capítulo tiene como objetivo fundamentar teóricamente la problemática asociada a la gestión de asignaturas electivas en instituciones de educación superior, abordando conceptos claves relacionados con los sistemas de información académica, las arquitecturas de software aplicables y los criterios de diseño que sustentan el desarrollo de soluciones orientadas a mejorar estos procesos. Asimismo, se examinan modelos existentes de gestión educativa y su aplicabilidad en contextos con necesidades institucionales particulares, como es el caso del sistema universitario cubano.

1.1. Ingeniería de software aplicada a la gestión de procesos académicos

La ingeniería de software constituye una disciplina fundamental orientada al análisis, diseño, implementación y mantenimiento de sistemas computacionales robustos y escalables. Su aplicación trasciende los límites tradicionales de la informática, extendiéndose hacia sectores críticos como la educación superior, donde la gestión eficiente de procesos académicos representa un pilar estratégico para el funcionamiento institucional.

En el contexto universitario contemporáneo, la administración de procesos formativos demanda soluciones tecnológicas que garanticen la integración, automatización y trazabilidad de las actividades académicas. La complejidad inherente a la gestión de asignaturas electivas, matrícula estudiantil, validación docente y generación de reportes institucionales requiere la aplicación de principios sólidos de ingeniería de software que aseguren la confiabilidad, mantenibilidad y escalabilidad de las soluciones desarrolladas.

El desarrollo sostenible de las instituciones educativas exige la implementación de

entornos digitales que faciliten la colaboración entre los diversos actores del proceso formativo, promoviendo la transparencia administrativa y optimizando la asignación de recursos académicos. En el ámbito de la gestión universitaria, la interacción efectiva entre administradores, docentes y estudiantes resulta esencial para el logro de los objetivos institucionales. Sin embargo, una de las principales limitaciones identificadas radica en la ausencia de sistemas integrados que permitan la gestión centralizada y automatizada de los procesos de inscripción y administración de asignaturas electivas.

Aunque existen diversas plataformas de gestión académica en el mercado, muchas de ellas presentan restricciones significativas en términos de personalización, costos de implementación y adaptabilidad a las particularidades normativas de cada institución. La mayoría de estos sistemas comerciales requieren inversiones considerables en licenciamiento y capacitación, además de no ofrecer la flexibilidad necesaria para configurar reglas específicas de asignación según criterios institucionales diferenciados, tales como restricciones por carrera, año académico o disponibilidad de cupos.

Por tanto, resulta imperativo desarrollar soluciones tecnológicas propias que permitan la gestión integral y eficiente de los procesos académicos, incorporando principios de ingeniería de software que garanticen la calidad, seguridad y escalabilidad del sistema resultante. La aplicación de arquitecturas bien definidas, patrones de diseño apropiados y metodologías de desarrollo ágiles constituyen elementos clave para lograr soluciones que respondan efectivamente a las necesidades específicas del entorno universitario cubano.

1.1.1. Arquitectura de Software

La arquitectura de software constituye un elemento esencial para el diseño e implementación de sistemas computacionales robustos destinados a la gestión de procesos académicos universitarios. Según Bass et al. (2021), la arquitectura de software puede definirse como la estructura fundamental de un sistema, que comprende sus componentes, las relaciones entre estos componentes y los principios que rigen su diseño y evolución a lo largo del tiempo. Esta disciplina proporciona los mecanismos necesarios de comunicación y coordinación entre los diversos componentes que integran el sistema, estableciendo las bases para soluciones estructurales que satisfagan tanto los requisitos técnicos como los operativos, mientras optimiza atributos críticos de calidad tales como el rendimiento, la seguridad y la escalabilidad.

La definición arquitectónica trasciende el ámbito puramente técnico, constituyendo un reflejo directo de las prioridades estratégicas institucionales. En el contexto específico de sistemas de gestión de asignaturas electivas, la arquitectura debe contemplar no solo la funcionalidad inmediata del sistema, sino también su capacidad de adaptación a futuros requerimientos institucionales, su mantenibilidad a largo plazo y su capacidad de integración con otros sistemas existentes en el entorno universita-

rio. Las decisiones arquitectónicas comprenden la selección de patrones estructurales apropiados, la especificación de interfaces entre módulos del sistema y la gestión efectiva de dependencias, todo ello orientado a equilibrar las demandas operativas de la gestión académica con las restricciones tecnológicas existentes (Gebrekidan 2023).

Uno de los elementos fundamentales de la arquitectura de software radica en su capacidad para gestionar la complejidad inherente a los sistemas académicos contemporáneos, complejidad que no representa únicamente un desafío técnico, sino también organizacional. La ley de Conway (Bird 2010) establece que la estructura de un sistema de software tiende a reflejar la estructura de comunicación de la organización que lo desarrolla (Richards y Ford 2020). Por consiguiente, una arquitectura adecuadamente diseñada para entornos universitarios debe ser técnicamente sólida y estar alineada con la cultura organizacional y los procesos administrativos propios de la institución educativa.

Los principios fundamentales que sustentan una arquitectura de software sólida incluyen la separación de responsabilidades, la modularidad, la escalabilidad, la reutilización de componentes y la facilidad de mantenimiento. En sistemas de gestión de asignaturas electivas, estos principios se traducen en la necesidad de separar claramente las funcionalidades relacionadas con la gestión de usuarios, la administración de cursos, el procesamiento de inscripciones y la generación de reportes, permitiendo que cada módulo pueda ser desarrollado, probado y mantenido de forma independiente.

La escalabilidad representa un aspecto particularmente relevante en el diseño de sistemas académicos, dado que las instituciones universitarias experimentan variaciones significativas en la carga de trabajo según los períodos de inscripción, matrícula y cierre de semestres. Una arquitectura bien diseñada debe ser capaz de manejar picos de demanda durante los períodos críticos de inscripción sin comprometer el rendimiento del sistema, mientras mantiene una utilización eficiente de recursos durante períodos de menor actividad.

En el ámbito específico de la gestión de asignaturas electivas, la arquitectura debe contemplar la implementación de mecanismos robustos de control de acceso y autorización, que permitan definir permisos diferenciados según el rol del usuario y garanticen que cada actor del proceso pueda acceder únicamente a las funcionalidades y datos correspondientes a sus responsabilidades institucionales. Esta compartimentación de accesos resulta fundamental para mantener la integridad del proceso de aprobación docente, inscripción estudiantil y administración general del sistema, mientras la interoperabilidad con sistemas de información estudiantil existentes, plataformas de gestión académica institucionales y servicios de autenticación centralizados evita la creación de silos de información que dificulten la gestión integral de los procesos formativos.

Capítulo 2

Concepción y diseño de la solución

El desarrollo de sistemas de información efectivos requiere una comprensión profunda de las necesidades institucionales y una traducción precisa de dichas necesidades en especificaciones técnicas funcionales. La fase de análisis y diseño constituye el puente fundamental entre la identificación de problemáticas organizacionales y la materialización de soluciones tecnológicas viables, donde interactúan desarrolladores y usuarios para analizar la realidad desde distintas perspectivas y definir los componentes esenciales del sistema propuesto.

En este capítulo se desarrolla el análisis detallado de los requerimientos funcionales y no funcionales que fundamentan la propuesta de solución, se caracterizan los roles y perfiles de usuario del sistema, y se presenta el diseño conceptual que soportará la funcionalidad de la plataforma. Asimismo, se establecen los casos de uso principales y la arquitectura de datos necesaria para garantizar que la solución responda de manera integral a las necesidades identificadas en el contexto de la gestión de asignaturas electivas universitarias.

2.1. Problemática

Las instituciones de educación superior enfrentan desafíos significativos en la gestión de asignaturas electivas, particularmente cuando estas deben coordinarse entre múltiples facultades, carreras y niveles académicos. En el contexto universitario cubano, este proceso tradicionalmente se ha caracterizado por una alta dependencia de métodos manuales y sistemas fragmentados que no logran integrar de manera efectiva las diferentes etapas del ciclo de gestión académica. Esta situación genera ineficiencias operativas, limita la transparencia del proceso y dificulta la toma de decisiones basada en datos confiables.

La problemática se manifiesta de manera particular en la Dirección de Formación de Pregrado, donde convergen las necesidades de múltiples actores con diferentes

niveles de responsabilidad y acceso a la información. Los profesores requieren mecanismos ágiles para proponer nuevas asignaturas electivas, especificando contenidos, requisitos y modalidades de evaluación. Simultáneamente, los administradores académicos necesitan herramientas que faciliten la revisión, validación y aprobación de estas propuestas, aplicando criterios institucionales específicos. Por su parte, los estudiantes demandan procesos de inscripción claros, equitativos y que respeten las particularidades de su plan de estudios.

El sistema propuesto aborda esta problemática mediante el desarrollo de una plataforma integrada que centraliza la gestión de asignaturas electivas en un entorno digital unificado. La solución contempla la creación de **roles diferenciados** que reflejan las responsabilidades específicas de cada actor en el proceso académico. Los **profesores** dispondrán de interfaces especializadas para la creación y gestión de propuestas de asignaturas, mientras que los **administradores** contarán con módulos de supervisión y control que permitan la aplicación sistemática de criterios de evaluación institucionales. Los **estudiantes**, por su parte, podrán consultar las asignaturas disponibles y realizar su inscripción a través del sistema, de manera análoga al entorno EVEA, accediendo únicamente a aquellas aprobadas por la DFP.

Para garantizar la integridad y coherencia del proceso, el sistema implementará **flujos de trabajo estructurados** que guíen cada etapa desde la propuesta inicial hasta la matrícula final del estudiante. Estos flujos incluirán mecanismos de validación automática, notificaciones de estado y controles de capacidad que aseguren el cumplimiento de los límites establecidos para cada asignatura. Adicionalmente, se incorporarán funcionalidades de **trazabilidad y auditoría** que permitan el seguimiento detallado de todas las acciones realizadas en el sistema, generando registros históricos útiles para la evaluación y mejora continua del proceso académico.

La arquitectura del sistema contempla además la generación automatizada de reportes y estadísticas que faciliten la supervisión institucional y la planificación académica a largo plazo. Estos elementos de inteligencia institucional permitirán identificar patrones de demanda, evaluar la efectividad de las ofertas electivas y optimizar la asignación de recursos académicos en función de las necesidades reales de la comunidad universitaria.

A continuación se detallan los requerimientos específicos que fundamentan el desarrollo de esta solución tecnológica.

2.2. Determinación de los Requerimientos

Los requerimientos del sistema se agrupan en tres categorías fundamentales: funcionales, no funcionales y de entorno. Estos fueron identificados mediante un análisis detallado de la problemática institucional y las necesidades específicas de la Dirección de Formación de Pregrado en el contexto de la gestión de asignaturas electivas

universitarias.

2.2.1. Requerimientos Funcionales

Las siguientes funcionalidades han sido definidas para permitir una gestión integral del proceso de inscripción, aprobación y matrícula de asignaturas electivas:

- **Definición de roles diferenciados:**

Se contemplan tres perfiles de usuario:

- **Administrador:** Acceso completo al sistema, con funciones de supervisión, aprobación y configuración de parámetros institucionales.
- **Profesor:** Propone asignaturas, gestiona información académica y administra calificaciones.
- **Estudiante:** Consulta la oferta de asignaturas, realiza su inscripción y accede a su información académica.

- **Gestión del proceso de propuesta de asignaturas:**

La plataforma permite el registro detallado de propuestas, incluyendo datos generales, docentes responsables, contenidos académicos, modalidades, cupos, requisitos y vínculos institucionales estratégicos.

- **Módulo de revisión y aprobación administrativa:**

Herramientas especializadas permiten a los administradores evaluar propuestas según criterios institucionales, aprobar asignaturas y habilitar accesos para los profesores responsables.

- **Funcionalidades de inscripción estudiantil:**

Los estudiantes acceden al catálogo de asignaturas aprobadas e inscriben según disponibilidad de cupos y reglas configurables por el administrador (facultad, carrera, año, etc.).

- **Generación automatizada de documentación académica:**

El sistema genera listados de inscritos, actas académicas y documentos oficiales con estructura y formato requeridos para certificación institucional.

2.2.2. Requerimientos No Funcionales

Para garantizar un funcionamiento óptimo, la solución debe cumplir con los siguientes requerimientos no funcionales:

- **Seguridad y control de acceso:** Autenticación y autorización según roles; protección de datos personales y académicos.
- **Usabilidad y experiencia de usuario:** Interfaces intuitivas, accesibles y adecuadas para usuarios con distintos niveles de competencia tecnológica.
- **Administración y mantenimiento:** Herramientas para configurar parámetros, gestionar contenidos y usuarios, y actualizar el sistema con mínima interrupción.
- **Accesibilidad multiplataforma:** Aplicación web compatible con diversos dispositivos y navegadores sin necesidad de instalación adicional.
- **Trazabilidad y auditoría:** Registro detallado de acciones para seguimiento, auditoría y resolución de incidentes.
- **Capacidades de análisis y reporte:** Estadísticas y reportes sobre matrícula, desempeño y uso de recursos para apoyar la toma de decisiones.

2.2.3. Requerimientos de Entorno

Para una implementación exitosa y operación estable, se establecen los siguientes requerimientos del entorno:

- **Arquitectura multiplataforma:** Compatibilidad con distintos sistemas operativos y entornos de servidor.
- **Alojamiento en infraestructura nacional:** Uso de servidores nacionales para garantizar autonomía tecnológica y cumplimiento de normativas de protección de datos.
- **Validación de identidad con sistemas institucionales existentes:** El sistema debe permitir la verificación de credenciales institucionales (correo y contraseña universitarios) a través de mecanismos de autenticación con servicios de la universidad, garantizando que los usuarios registrados sean miembros válidos de la comunidad académica.
- **Escalabilidad y rendimiento:** Capacidad de manejar crecimiento en usuarios y datos sin afectar el rendimiento, incluso en periodos de alta demanda.

2.3. Casos de uso y flujos principales

La definición de casos de uso constituye una herramienta fundamental para representar las interacciones entre el sistema y sus diferentes actores, estableciendo un marco conceptual que facilita la comprensión de las funcionalidades requeridas y los flujos de trabajo necesarios para su implementación. A través del análisis de casos de uso se logra identificar de manera precisa las responsabilidades y capacidades de cada rol dentro del sistema, proporcionando una base sólida para el diseño de la arquitectura funcional.

En el contexto del sistema de gestión de asignaturas electivas, se han identificado tres actores principales que interactúan con la plataforma, cada uno con responsabilidades específicas y niveles de acceso diferenciados que reflejan su función dentro del proceso académico institucional.

Actores del sistema

El sistema contempla tres perfiles de usuario claramente diferenciados:

Administrador (DFP): Representa el rol de mayor autoridad dentro del sistema, correspondiente a la Dirección de Formación de Pregrado. Este actor posee privilegios completos para supervisar, configurar y administrar todos los aspectos operativos de la plataforma, incluyendo la aprobación de asignaturas propuestas, la gestión de reglas de inscripción y el acceso a información estadística integral.

Profesor: Actor responsable de la creación y gestión académica de las asignaturas electivas. Su participación en el sistema abarca desde la propuesta inicial de cursos hasta la administración de calificaciones y generación de documentación académica oficial.

Estudiante: Usuario final que accede al sistema para consultar la oferta académica disponible, realizar inscripciones en asignaturas de su interés y consultar su información académica personal.

Interacciones del Administrador

El administrador del sistema ejerce funciones de supervisión y control que garantizan el cumplimiento de los estándares académicos institucionales. Sus interacciones principales incluyen la evaluación y aprobación de propuestas de asignaturas presentadas por los profesores, aplicando criterios específicos establecidos por la institución.

La gestión de entidades constituye una función central del administrador, quien puede crear, editar y eliminar elementos fundamentales del sistema como facultades, carreras, modalidades y otros componentes estructurales. Esta capacidad de configuración permite adaptar el sistema a las necesidades específicas de la institución y mantenerlo actualizado ante cambios organizacionales.

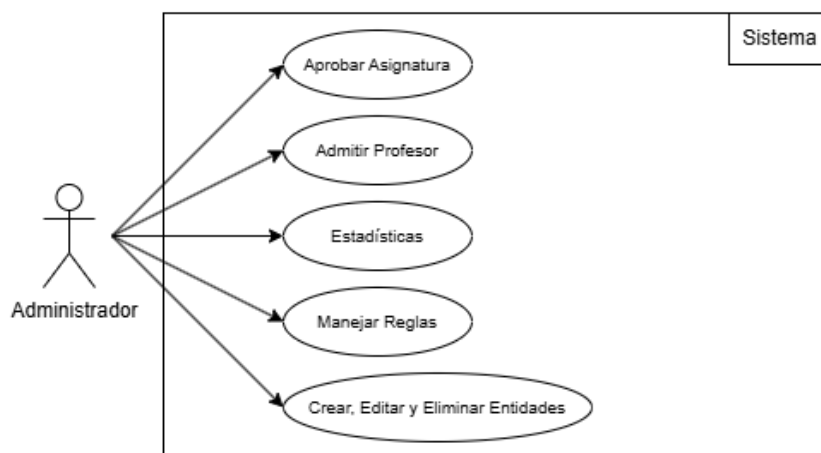


Figura 2.1: Diagrama de casos de uso del Administrador.

El establecimiento y modificación de reglas de inscripción representa otra responsabilidad crítica del administrador. Estas reglas determinan los criterios de elegibilidad para que los estudiantes puedan inscribirse en asignaturas específicas, considerando factores como año académico, facultad de procedencia, carrera y otros parámetros relevantes para la planificación académica.

La generación de estadísticas y reportes proporciona al administrador herramientas analíticas para evaluar el desempeño del sistema, identificar tendencias de matrícula y tomar decisiones informadas sobre la oferta académica. Estos análisis incluyen datos de participación por facultad, carrera y año académico, así como indicadores de rendimiento.

Interacciones del Profesor

Los profesores constituyen los proveedores de contenido académico dentro del sistema, con responsabilidades que abarcan desde la concepción inicial de las asignaturas hasta la evaluación final de los estudiantes participantes.

La inscripción de asignaturas representa el punto de entrada del profesor al sistema, requiriendo la presentación de información detallada sobre el curso propuesto. Este proceso incluye la especificación de datos generales, información del docente responsable y colaboradores, contenidos académicos, metodología de evaluación y requisitos para los estudiantes participantes.

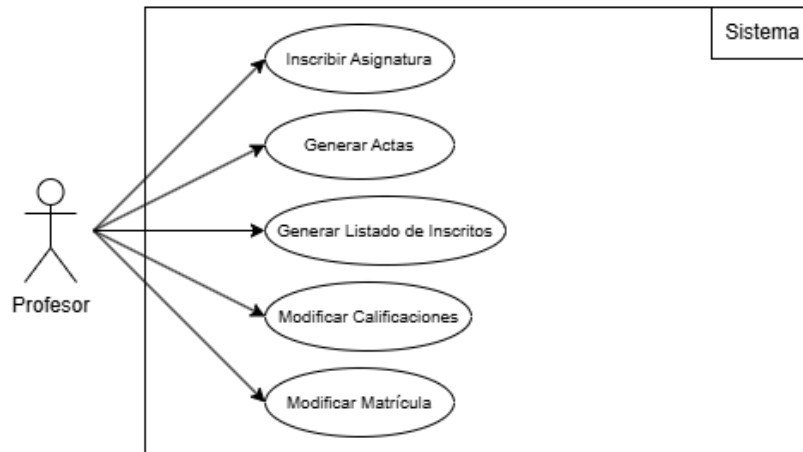


Figura 2.2: Diagrama de casos de uso del Profesor.

Una vez aprobada una asignatura, el profesor adquiere capacidades de gestión académica que incluyen la administración de la matrícula estudiantil. Esta función permite realizar ajustes en la lista de participantes según criterios académicos específicos, respetando siempre los límites de capacidad establecidos y las reglas institucionales vigentes.

La generación de listados de estudiantes inscritos y actas académicas constituye una funcionalidad esencial para la documentación oficial del proceso educativo. El sistema proporciona herramientas automatizadas para crear estos documentos con formato institucional, incluyendo espacios para firmas y sellos requeridos por la normativa universitaria.

La gestión de calificaciones permite al profesor registrar y modificar las evaluaciones de los estudiantes participantes, con controles de validación que aseguran la integridad de la información académica. El sistema implementa mecanismos de trazabilidad que registran todos los cambios realizados, proporcionando un historial completo de modificaciones.

Interacciones del Estudiante

Los estudiantes representan los beneficiarios finales del sistema, con acceso a funcionalidades específicamente diseñadas para facilitar su participación en el proceso de selección e inscripción de asignaturas electivas.

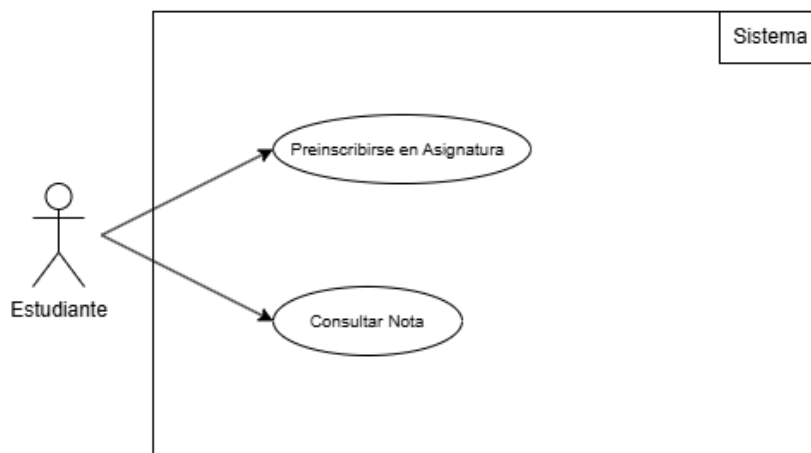


Figura 2.3: Diagrama de casos de uso del Estudiante.

La consulta de asignaturas disponibles proporciona a los estudiantes acceso al catálogo completo de cursos electivos aprobados por la administración. Esta funcionalidad presenta información detallada sobre cada asignatura, incluyendo objetivos, contenidos, modalidad de impartición, requisitos y disponibilidad de cupos.

El proceso de inscripción en asignaturas permite a los estudiantes formalizar su participación en cursos de su interés, sujeto a la disponibilidad de cupos y el cumplimiento de requisitos establecidos. El sistema implementa validaciones automáticas que verifican la elegibilidad del estudiante según las reglas configuradas por el administrador.

La consulta de calificaciones proporciona acceso a la información académica personal, permitiendo a los estudiantes monitorear su progreso y desempeño en las asignaturas en las que participan. Esta funcionalidad mantiene un registro histórico completo de todas las evaluaciones recibidas.

Flujos de trabajo principales

Los casos de uso identificados se organizan en flujos de trabajo secuenciales que reflejan el ciclo completo de gestión de asignaturas electivas. El flujo principal inicia con la propuesta de asignaturas por parte de los profesores, continúa con la evaluación y aprobación administrativa, prosigue con la habilitación de inscripciones estudiantiles y culmina con la generación de documentación académica oficial.

Este diseño de flujos garantiza la trazabilidad completa del proceso, desde la concepción inicial hasta la certificación final, proporcionando mecanismos de control y validación en cada etapa. La estructura modular de los casos de uso facilita la implementación incremental del sistema y permite la incorporación de nuevas funcionalidades según evolucionen las necesidades institucionales.

2.4. Diseño de la base de datos

Con el fin de estructurar y gestionar eficientemente la información del sistema, se ha diseñado una base de datos relacional que contempla las principales entidades involucradas. A continuación, se describen las entidades definidas junto con sus atributos y relaciones clave.

Definición de Entidades:

- **Curso:** ID (PK ¹), Título, ID Centro (FK ²), Modalidad, Capacidad de matrícula, ID Profesor principal (FK), Justificación del curso, Objetivo general, Objetivos específicos, Temario, Bibliografía básica, Bibliografía complementaria, Sistema de evaluación, Justificación de modalidad, Requisitos básicos, Lugar de encuentro, Ejes estratégicos, Sectores estratégicos, Ruta de carta de autorización, URL de imagen, Fecha de creación, Fecha de actualización, Estatus de aprobación.
- **Instancia de Curso:** ID (PK), ID Curso (FK), Fecha de inicio, Fecha de fin.
- **Aplicación a Curso:** ID (PK), ID Estudiante (FK), ID Instancia de Curso (FK), Carta de motivación, Año académico, Estatus, Fecha de aplicación.
- **Calificación de Curso:** ID (PK), ID Instancia de Curso (FK), ID Estudiante (FK), Calificación, Comentario.
- **Profesor:** ID Usuario (PK), Nombre completo, Email principal, Categoría docente, Grado académico, Teléfono fijo, Número de teléfono, Email secundario, Estatus de aprobación.
- **Estudiante:** ID Usuario (PK), Nombre completo, Número de identificación, Email principal, Nombre de usuario Evea, Número de teléfono, ID Facultad (FK), ID Carrera (FK), Email secundario.
- **Centro:** ID (PK), Nombre.

¹Primary Key - Llave primaria de la entidad

²Foreign Key - Llave foránea que referencia a otra tabla

- **Facultad:** ID (PK), Nombre.
- **Carrera:** ID (PK), Nombre.

Para facilitar la comprensión de la estructura de la base de datos, el diagrama de la Figura 2.1 ilustra el Modelo Entidad-Relacional-Extendido (MERX) del sistema, que refleja los conjuntos de entidades principales y sus interrelaciones clave.

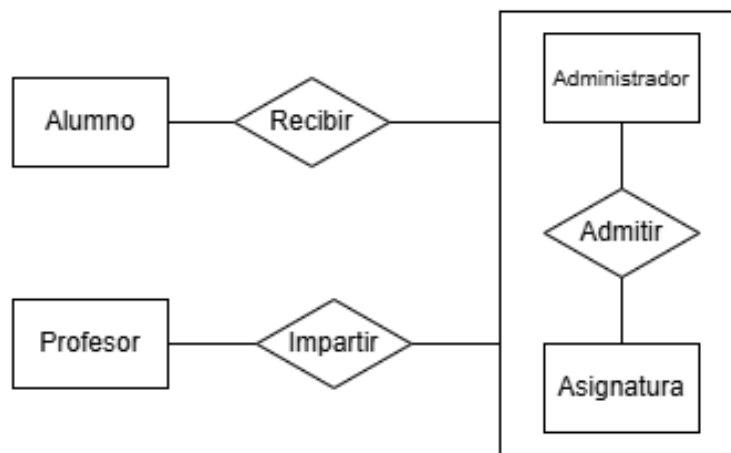


Figura 2.4: Modelo Entidad-Relacional Extendido de la base de datos.

Capítulo 3

Detalles de Implementación y Experimentación

Toda solución computacional, una vez concebida y diseñada, requiere una implementación técnica rigurosa para transformarse en una herramienta funcional y accesible para los usuarios finales. En este capítulo se abordan los aspectos fundamentales del proceso de desarrollo de la plataforma de gestión de asignaturas electivas, desde la selección de tecnologías hasta la validación experimental del sistema implementado. Se detallan las decisiones técnicas adoptadas, justificando la elección de cada componente tecnológico en función de los requerimientos identificados y las particularidades del contexto académico universitario.

El proceso de implementación se estructura en múltiples dimensiones que abarcan desde la arquitectura de datos hasta la experiencia de usuario final. Se presenta un análisis exhaustivo de las herramientas tecnológicas seleccionadas, considerando criterios de escalabilidad, mantenibilidad, seguridad y compatibilidad con el ecosistema tecnológico institucional. Asimismo, se documentan las pruebas experimentales realizadas para validar tanto la funcionalidad técnica como la efectividad operacional de la solución propuesta.

3.1. Tecnologías y herramientas utilizadas para el desarrollo del sistema

La arquitectura tecnológica del sistema incorpora principalmente herramientas de código abierto, buscando optimizar la sostenibilidad económica y facilitar la autonomía tecnológica de la institución, aunque también se ha integrado software propietario como SQL Server por su confiabilidad y soporte en entornos académicos. La selección de cada componente tecnológico se fundamenta en un análisis comparativo que

evalúa criterios de rendimiento, escalabilidad, seguridad, facilidad de mantenimiento y compatibilidad con estándares de desarrollo empresarial.

El enfoque metodológico adoptado para la selección tecnológica considera la necesidad de facilitar la colaboración entre desarrolladores y la incorporación de nuevos miembros al equipo de desarrollo en futuras iteraciones del proyecto. Esta consideración resulta fundamental para garantizar la continuidad del desarrollo y el mantenimiento a largo plazo de la plataforma académica.

3.1.1. Sistema de Gestión de Base de Datos: SQL Server

Para la gestión de datos del sistema se ha seleccionado Microsoft SQL Server como sistema de gestión de base de datos relacional. Esta elección se fundamenta en la robustez y confiabilidad que caracteriza a esta plataforma de base de datos empresarial, así como en su amplia adopción en entornos académicos y corporativos que garantiza la disponibilidad de recursos técnicos especializados.

SQL Server proporciona funcionalidades avanzadas de seguridad, backup y recuperación que resultan esenciales para la protección de datos académicos sensibles. Su arquitectura permite implementar mecanismos de control de acceso granular y auditoría detallada, aspectos fundamentales para el cumplimiento de regulaciones de protección de datos en el contexto educativo. Adicionalmente, las capacidades de escalabilidad horizontal y vertical de la plataforma aseguran el soporte adecuado para el crecimiento proyectado de usuarios y volumen de información académica.

3.1.2. Framework de Desarrollo Backend: ASP.NET Core

Para la implementación de la lógica de negocio y los servicios del sistema se ha adoptado ASP.NET Core como framework de desarrollo backend. Esta tecnología de Microsoft representa una plataforma de desarrollo moderna, multiplataforma y de alto rendimiento que facilita la creación de aplicaciones web empresariales escalables y seguras.

La elección de ASP.NET Core se justifica por su arquitectura modular que permite el desarrollo de servicios web RESTful con capacidades avanzadas de autenticación, autorización y gestión de sesiones. El framework proporciona herramientas integradas para la implementación de patrones de diseño empresariales como la inyección de dependencias, el control de versiones de API y la documentación automática de servicios. Estas características facilitan el mantenimiento del código y la colaboración entre desarrolladores en equipos distribuidos.

3.1.3. Mapeo Objeto-Relacional: Entity Framework Core

Para la abstracción de la capa de acceso a datos se utiliza Entity Framework Core, el mapeador objeto-relacional oficial de Microsoft para .NET. Esta herramienta facilita la interacción con la base de datos mediante el uso de objetos .NET, eliminando la necesidad de escribir código SQL específico para la mayoría de operaciones de acceso a datos.

Entity Framework Core proporciona funcionalidades avanzadas como migraciones automáticas de esquema de base de datos, lazy loading de entidades relacionadas y optimización automática de consultas. Estas características reducen significativamente el tiempo de desarrollo y minimizan la posibilidad de errores en la implementación de operaciones de persistencia de datos. Además, su integración nativa con ASP.NET Core facilita la implementación de patrones arquitectónicos como Repository y Unit of Work.

3.1.4. Tecnología de Desarrollo Frontend: React.js

Para la implementación de la interfaz de usuario se ha seleccionado React.js, una biblioteca de JavaScript desarrollada por Meta que facilita la construcción de interfaces de usuario interactivas y responsivas. React.js se ha consolidado como una de las tecnologías frontend más utilizadas en el desarrollo de aplicaciones web modernas, con un ecosistema maduro y una comunidad activa de desarrolladores.

La arquitectura basada en componentes de React permite la creación de interfaces modulares y reutilizables que facilitan el mantenimiento y la extensión de funcionalidades. Su modelo de gestión de estado y el virtual DOM proporcionan un rendimiento óptimo para aplicaciones con alta interactividad, aspecto fundamental para la experiencia de usuario en procesos académicos que requieren navegación fluida entre múltiples formularios y vistas de datos.

3.1.5. Plataforma de Contenedorización: Docker

Para el empaquetado y despliegue de la aplicación se ha adoptado Docker como plataforma de contenedorización. Esta tecnología permite encapsular la aplicación y todas sus dependencias en contenedores ligeros y portátiles que garantizan la consistencia del entorno de ejecución independientemente de la infraestructura de despliegue.

La utilización de Docker facilita la implementación de prácticas de integración continua y despliegue continuo, aspectos fundamentales para el mantenimiento eficiente de aplicaciones en producción. Los contenedores Docker proporcionan aislamiento de procesos y recursos que mejora la seguridad y estabilidad del sistema, mientras que

su portabilidad garantiza la flexibilidad de migración entre diferentes entornos de infraestructura sin dependencias externas.

La estrategia de contenedorización adoptada contempla la separación de la aplicación en múltiples contenedores especializados para cada componente del sistema, incluyendo el frontend React, el backend ASP.NET Core y la base de datos SQL Server. Esta arquitectura de microservicios facilita el escalado independiente de cada componente según las demandas específicas de carga y proporciona mayor flexibilidad para actualizaciones y mantenimiento del sistema.

3.2. Arquitectura del sistema implementado

3.3. Pruebas y validación experimental