**Aplicación móvil de reservas de tutorías para estudiantes de Ingeniería de Sistemas, Universidad de Pamplona, sede Villa del Rosario**

María Gabriela Espinosa Alonso- 1091964010

María Victoria Nieto Navarro- 1229329

Camilo Alfonso Mora Espinel– 1092334052

Brayan Danilo Sanchez Monroy - 1091965055

Facultad de Ingenierías y Arquitectura, Ingeniería de Sistemas

Universidad de Pamplona

Ingeniería de Software II– Ar

Fanny Casadiego Chiquillo

Villa del Rosario – Norte de Santander

2025

# Lista de Cambios

|  |  |  |  |
| --- | --- | --- | --- |
| Núm. | Fecha | Descripción | Autor/es |
| 0 | 18/10/2024 | [pág. 8] En el OBJ-003, se refactorizó el título donde aparecía "Optimizar el emparejamiento y la gestión del tiempo para las tutorías" por "Optimizar la gestión de recursos para las tutorías". | Camilo Mora |
| 1 | 18/10/2024 | [pág. 8] Se eliminó OBJ-003.2: "Implementar un algoritmo de emparejamiento que asigne a los estudiantes los tutores más adecuados, en función de su carrera, área de estudio y horarios". | Camilo Mora |
| 2 | 18/10/2024 | [pág. 8] En el OBJ-003, donde aparecía "Permitir la búsqueda avanzada de tutores por materia, carrera o disponibilidad", aparece ahora "Permitir la búsqueda avanzada y filtrado de tutores por materia, carrera u horario". | Camilo Mora |
| 3 | 18/10/2024 | [pág. 10] En el requisito REQ-INF-002, donde aparecía "Información detallada sobre tutores", aparece ahora "Información detallada sobre tutores y estudiantes". | Camilo Mora |
| 4 | 18/10/2024 | [pág. 11] Se eliminó REQ-FUN-003: "Emparejamiento automático de tutores y estudiantes" y tomó en su lugar el requerimiento titulado “Registro de interacciones y tutorías anteriores  ” | Camilo Mora |
| 5 | 12/03/2025 | [pág. 7]Cambios en los objetivos | Victoria Nieto |
| 6 | 13/03/2025 | Cambios en el documento | Gabriela Espinosa |
| 7 | 15/03/2025 | [pág. 19]Cambios en el diagrama de casos de uso | Gabriela Espinosa |
| 8 | 21/03/2025 | [pág. 12-28]Cambios en las tablas de requisitos [pág 36]Añadir análisis de riesgo (matriz DOFA) | Gabriela Espinosa |
| 9 | 27/03/2025 | [pág. 6]Añadir Antecedentes del Proyecto | Danilo Sanchez |
| 10 | 3/04/2025 | [pág. 39] Arquitectura de Software | Camilo Mora |
| 11 | 8/04/2025 | [pág. 38] Matriz de Riesgos | Gabriela Espinosa |

**Índice**

[Lista de Cambios 2](#_Toc197106926)

[Introducción 4](#_Toc197106927)

[Participantes del Proyecto 5](#_Toc197106928)

[Clientes: 5](#_Toc197106929)

[Desarrolladores: 5](#_Toc197106930)

[Descripción del Sistema 6](#_Toc197106931)

[Antecedentes 7](#_Toc197106932)

[Antecedente Internacional 7](#_Toc197106933)

[Antecedentes Nacionales 8](#_Toc197106934)

[Antecedente Regional 9](#_Toc197106935)

[Objetivos 11](#_Toc197106936)

[Catálogo de Requisitos 16](#_Toc197106937)

[Requisitos de Información 16](#_Toc197106938)

[Requisitos Funcionales 19](#_Toc197106939)

[Diagrama de Casos de Uso: 25](#_Toc197106940)

[Definición de Actores 26](#_Toc197106941)

[Requisitos No Funcionales 27](#_Toc197106942)

[Resultados de la Encuesta 31](#_Toc197106943)

[Análisis de Riesgos 36](#_Toc197106944)

[Matriz DOFA: 36](#_Toc197106945)

[Glosario 36](#_Toc197106946)

[Matriz de Riesgos 38](#_Toc197106947)

[Arquitectura de Software 39](#_Toc197106948)

[Backend: Arquitectura por Capas Enfocada al Dominio 39](#_Toc197106949)

[Frontend: Arquitectura Cliente-Servidor 40](#_Toc197106950)

[Bibliografia 41](#_Toc197106951)

# Introducción

En el contexto universitario, las tutorías académicas juegan un papel crucial al proporcionar a los estudiantes la oportunidad de recibir apoyo personalizado por parte de profesores o estudiantes de semestres superiores. No obstante, la organización eficiente de estas tutorías puede ser un desafío debido a la falta de herramientas adecuadas para gestionar las reservas, la disponibilidad de los tutores, y la asignación de espacios. Estos problemas afectan tanto a los estudiantes, que a menudo encuentran dificultades para coordinarse, como a los tutores, quienes enfrentan una sobrecarga administrativa.

Para abordar estas necesidades, este proyecto tiene como objetivo diseñar un sistema de reservas de tutorías universitarias que facilite y automatice el proceso de organización. El sistema permitirá a los estudiantes reservar sesiones de tutoría de manera fácil y rápida, con acceso a la disponibilidad en tiempo real, optimizando la gestión del tiempo y la asignación de espacios físicos o virtuales. Así mismo, proporcionará a los tutores una plataforma intuitiva para gestionar sus horarios y optimizar sus interacciones con los estudiantes.

Este proyecto busca no solo mejorar la eficiencia en la organización de las tutorías, sino también fomentar un ambiente de colaboración más estructurado dentro de la comunidad universitaria. Al implementar una solución tecnológica innovadora, se espera reducir los problemas logísticos y aumentar el acceso a tutorías, contribuyendo así al éxito académico de los estudiantes.

# Participantes del Proyecto

## Clientes:

* **Estudiantes Universitarios:** Son los usuarios principales del sistema que necesitan acceder fácilmente a las tutorías. Sus necesidades incluyen la reserva de sesiones, la visualización de la disponibilidad de tutores, y la interacción con la plataforma.
* **Tutores (profesores o estudiantes de semestres superiores)**:Son los encargados de ofrecer las tutorías. Necesitan gestionar sus horarios, confirmar o modificar las reservas.

## Desarrolladores:

* **Equipo de Desarrollo de Software**: Los desarrolladores del sistema que implementarán la plataforma. Se encargarán de la creación de funcionalidades clave como la gestión de horarios, asignación de espacios y acceso en tiempo real. También, serán los responsables de estructurar y gestionar la base datos que almacene la información sobre tutores, estudiantes, horarios y reservas.
* **Diseñadores UX/UI**: Se encargarán de crear una interfaz intuitiva para facilitar la interacción tanto de estudiantes como de tutores.

# Descripción del Sistema

El sistema de reservas de tutorías universitarias es una plataforma digital diseñada para mejorar el acceso y la organización de clases de apoyo académico. Su principal objetivo es facilitar la conexión entre estudiantes que requieren refuerzo en materias específicas y los tutores que brindan este apoyo. A través de la plataforma, los estudiantes pueden buscar, reservar y gestionar tutorías de forma rápida y sencilla, optimizando tanto su tiempo como el de los tutores.

Además de beneficiar a los estudiantes al proporcionar un acceso estructurado al apoyo académico, la plataforma facilita la gestión de horarios para tutores y profesores, promoviendo una mejor organización. También fomenta la colaboración entre los usuarios, creando un espacio en el que los estudiantes pueden encontrar el apoyo académico que necesitan de manera ágil y efectiva.

# Antecedentes

## Antecedente Internacional

Como antecedente internacional, se identificó el Proyecto Fin de Carrera "Sistema para la petición de cita de tutoría usando tecnología de Portlets", desarrollado en la Universidad de Zaragoza por Marcos Mainar Lalmolda bajo la dirección del profesor Pedro Javier Álvarez Pérez-Aradros en el curso 2010/2011. Este proyecto surge a partir de la necesidad de mejorar la gestión de tutorías universitarias, dado que tradicionalmente los profesores publicaban sus horarios de atención mediante notas físicas, correos electrónicos o sitios web personales. Sin embargo, este sistema resultaba ineficiente, ya que los alumnos solían acudir sin previo aviso, lo que generaba tiempos de espera prolongados y dificultaba la preparación de las tutorías por parte del docente (Mainar Lalmolda, 2011).

Para resolver esta problemática, el proyecto desarrolló un sistema web compuesto por dos aplicaciones: una aplicación web para profesores y una aplicación Portlet para alumnos. La aplicación docente permitía gestionar asignaturas, registrar estudiantes y establecer horarios de tutorías, además de visualizar y administrar las sesiones programadas. Por otro lado, la aplicación para alumnos facilitaba la solicitud, modificación o cancelación de citas de manera organizada. Implementado en el Departamento de Informática e Ingeniería de Sistemas, este sistema permitió optimizar el tiempo tanto de alumnos como de profesores, mejorando la planificación y personalización de las tutorías. Finalmente, fue utilizado de manera experimental, permitiendo a los estudiantes del director del proyecto acceder a la plataforma a través de su página web para gestionar sus citas de manera estructurada y eficiente (Mainar Lalmolda, 2011).

## Antecedentes Nacionales

A nivel nacional, un referente importante es el proyecto "Desarrollo de un aplicativo web para la reserva de salas de informática, multimedia y equipos audiovisuales en la Universidad de San Buenaventura, Sede Bogotá", realizado en 2006 por Alexander Trujillo Velandia y Juan Carlos Valencia Ortiz. Este trabajo responde a la problemática que enfrentaban estudiantes y docentes en la gestión de reservas de espacios y equipos tecnológicos dentro de la universidad, ya que el proceso manual generaba demoras, saturación en la atención y falta de control sobre los recursos. Además, la implementación de soluciones tecnológicas representaba un reto debido a los costos asociados a licencias de software. Ante esta situación, el proyecto planteó el desarrollo de una plataforma web que modernizara el proceso y optimizara la administración de estos servicios, permitiendo a la institución mejorar la experiencia de los usuarios y garantizar un uso más eficiente de sus recursos.

El sistema propuesto no solo buscaba facilitar la reserva de salas y equipos, sino que también respondía a la necesidad de tecnificación de la universidad. Para ello, se llevó a cabo un análisis del proceso de reservas, se diseñó un modelo de base de datos adaptado a las áreas de informática y multimedia, y se desarrolló una interfaz con las funcionalidades requeridas. Su implementación permitió reducir tiempos de espera, mejorar la organización interna y optimizar la gestión de inventarios, asegurando que la institución pudiera identificar con mayor precisión sus necesidades de actualización y renovación de equipos. Con este avance, la Universidad de San Buenaventura fortaleció su compromiso con la modernización de sus procesos, alineándose con las tendencias tecnológicas en el ámbito educativo (Trujillo Velandia & Valencia Ortiz, 2006).

## Antecedente Regional

En la Universidad Francisco de Paula Santander (UFPS), la asesoría académica es una función esencial que los docentes deben cumplir como parte de su carga académica. Sin embargo, la ausencia de un sistema unificado para la programación de citas ha generado una gestión manual e ineficiente, donde cada profesor establece de manera independiente los horarios y espacios de asesoría. Esto ha derivado en problemas de comunicación y coordinación entre docentes y estudiantes, afectando la calidad del proceso de formación.

Actualmente, no existe un mecanismo institucional estandarizado para la gestión y reporte de asesorías, lo que impacta tanto la planificación académica como el cumplimiento de requisitos administrativos. A pesar de la existencia de herramientas tecnológicas que podrían facilitar este proceso, sus costos elevados y la falta de personalización impiden su adopción, ya que no generan los reportes específicos que la universidad requiere. Como resultado, los estudiantes enfrentan dificultades al programar sus citas, ya sean virtuales o presenciales. “la UFPS no establece con claridad el formato de asesoría ni el mecanismo de operacionalización de la gestión de espacios y citas, por esta razón por la cual cada profesor tiene la autonomía de proporcionar sus asesorías de manera libre. Como se mencionó anteriormente, esta situación conduce a problemas de comunicación si el profesor no define adecuadamente el protocolo de asesoría. Preguntas esenciales para el estudiante, como ‘¿dónde?’, ‘¿a qué hora?’ y ‘¿cuánto tiempo durará cada asesoría?’, quedan sin respuesta clara” (Benítez & Pacheco, 2024).

Como respuesta a esta problemática regional, se desarrolló AgendarUFPS, en el marco del proyecto titulado "Software de Gestión de Citas para Asesorías de Profesores y Estudiantes Basado en Google Calendar", elaborado por Ronald Eduardo Benítez Mejía y Jhoser Fabián Pacheco Quintero. Este software permite automatizar la reserva de espacios y estandarizar la programación de asesorías, mejorando la comunicación entre estudiantes y profesores. Diseñado bajo la metodología Design Thinking, ofrece una solución intuitiva y accesible, alineada con las necesidades institucionales. Su implementación no solo optimiza la planificación académica, sino que también facilita la generación de informes administrativos, contribuyendo a una gestión más eficiente de las asesorías dentro de la universidad.

# Objetivos

**Objetivo General:**

Diseñar una aplicación móvil de reservas de tutorías universitarias utilizando la metodología ágil Scrum para facilitar la conexión entre estudiantes que necesitan apoyo académico, mejorando la organización e incentivando la colaboración en la comunidad estudiantil.

**Objetivos Específicos:**

* Analizar los requisitos para el diseño de un sistema de reservas de tutorías universitarias, investigando las necesidades tanto de los estudiantes como de los mentores y profesores en términos de organización y comunicación.
* Crear el sistema de reservas de tutorías, incorporando un algoritmo de análisis de datos y funcionalidades que faciliten la conexión entre alumnos y mentores, optimizando la gestión del tiempo, la organización de reservas y la asignación de espacios para las tutorías.
* Optimizar el rendimiento del sistema de reservas de tutorías, implementando técnicas de computación paralela con hilos en Java para mejorar la capacidad de respuesta y la eficiencia en el procesamiento de solicitudes de reserva, asignación de mentores y gestión de disponibilidad en tiempo real.
* Validar el sistema mediante pruebas de funcionalidad y usabilidad, asegurando que cumple con los requisitos establecidos y que realmente contribuye a una mejor organización y a una mayor interacción entre los usuarios.

|  |  |
| --- | --- |
| **OBJ-001** | **Analizar los requisitos para el diseño de un sistema de reservas de tutorías universitarias, investigando las necesidades tanto de los estudiantes como de los mentores y profesores en términos de organización y comunicación.** |
| **Versión** | 2.0 (13/03/2025) |
| **Autores** | María Victoria Nieto Navarro María Gabriela Espinosa Alonso (colaboradora) |
| **Fuentes** | Estudiantes y tutores (encuesta anónima) |
| **Descripción** | El sistema deberá mejorar la comunicación entre estudiantes y tutores, facilitando una mayor coordinación y planificación eficiente de las asesorías. Además, se buscará optimizar la experiencia de usuario mediante una interfaz intuitiva y accesible, que permita una gestión sencilla de las reservas y la disponibilidad de los tutores. |
| **Subobjetivos** | * **OBJ-001.1**: Implementar un sistema de notificaciones para recordar y actualizar sobre las tutorías programadas. * **OBJ-001.2:** Permitir a los tutores actualizar su disponibilidad en tiempo real. * **OBJ-001.3:** Desarrollar una interfaz de usuario amigable y accesible para estudiantes y tutores, garantizando la usabilidad en dispositivos móviles. * **OBJ-001.4:** Integrar un sistema de retroalimentación post-tutoría para evaluar la calidad de las asesorías y mejorar el servicio. |
| **Importancia** | Alta |
| **Urgencia** | Alta |
| **Estado** | En planificación |
| **Estabilidad** | Media |
| **Comentarios** | Se identificó una falta de comunicación adecuada como un obstáculo clave en las respuestas de los estudiantes. |

|  |  |
| --- | --- |
| **OBJ-002** | **Crear el sistema de reservas de tutorías, incorporando un algoritmo de análisis de datos y funcionalidades que faciliten la conexión entre alumnos y mentores, optimizando la gestión del tiempo, la organización de reservas y la asignación de espacios para las tutorías.** |
| **Versión** | 2.0 (13/03/2025) |
| **Autor** | María Victoria Nieto  María Gabriela Espinosa Alonso (colaboradora) |
| **Fuentes** | Estudiantes (Encuesta anónima) |
| **Descripción** | El sistema de reservas de tutorías deberá integrar un algoritmo inteligente que analice datos como la disponibilidad de tutores, la demanda de los estudiantes y la ubicación de los espacios físicos. Esto permitirá una asignación eficiente de recursos, minimizando conflictos de horarios y optimizando el uso de los espacios disponibles. Además, se incluirán funcionalidades que faciliten la interacción entre alumnos y mentores, como búsquedas personalizadas y recomendaciones basadas en necesidades académicas. |
| **Subobjetivos** | * **OBJ-002.1:** Diseñar e implementar un algoritmo de análisis de datos que optimice la asignación de tutorías en función de la disponibilidad de tutores, la demanda de estudiantes y la ubicación de los espacios. * **OBJ-002.2:** Desarrollar un módulo de búsqueda y filtrado para que los estudiantes encuentren tutores según sus necesidades académicas (materia, horario, ubicación). * **OBJ-002.3:** Integrar un sistema de asignación automática de espacios físicos para las tutorías, considerando la capacidad y disponibilidad de las aulas. |
| **Importancia** | Muy alta |
| **Urgencia** | Alta |
| **Estado** | En planificación |
| **Estabilidad** | Alta |
| **Comentarios** | El algoritmo de análisis de datos está siendo probado con datos simulados para garantizar su eficiencia. |

|  |  |
| --- | --- |
| **OBJ-003** | **Optimizar el rendimiento del sistema de reservas de tutorías, implementando técnicas de computación paralela con hilos en Java para mejorar la capacidad de respuesta y la eficiencia en el procesamiento de solicitudes de reserva, asignación de mentores y gestión de disponibilidad en tiempo real.** |
| **Versión** | 2.0 (13/03/2025) |
| **Autor** | María Victoria Nieto  María Gabriela Espinosa Alonso (colaboradora) |
| **Fuentes** | Estudiantes (Encuesta anónima) |
| **Descripción** | El sistema de reservas de tutorías requiere una optimización significativa en su rendimiento para manejar un alto volumen de solicitudes simultáneas. Para ello, se implementarán técnicas de computación paralela utilizando hilos en Java, lo que permitirá procesar múltiples solicitudes de reserva, asignaciones de mentores y actualizaciones de disponibilidad de manera concurrente. Esto mejorará la capacidad de respuesta del sistema y garantizará una experiencia fluida para los usuarios, incluso en momentos de alta demanda. |
| **Subobjetivos** | * **OBJ-003.1**: Identificar y analizar los cuellos de botella en el sistema actual mediante pruebas de carga y estrés. * **OBJ-003.2**: Diseñar e implementar un modelo de computación paralela utilizando hilos en Java para procesar solicitudes de reserva de manera concurrente. * **OBJ-003.3:** Optimizar el algoritmo de asignación de mentores para que funcione de manera eficiente en un entorno paralelo. * **OBJ-003.4:** Implementar un sistema de gestión de disponibilidad en tiempo real que utilice hilos para actualizar y sincronizar la información de manera eficaz. * **OBJ-003.5:** Realizar pruebas de rendimiento para validar la mejora en la capacidad de respuesta y la eficiencia del sistema. |
| **Importancia** | Alta |
| **Urgencia** | Media |
| **Estado** | En planificación |
| **Estabilidad** | Baja |
| **Comentarios** | * Las pruebas de carga han identificado cuellos de botella en el procesamiento de solicitudes simultáneas, especialmente durante los picos de demanda. * Se está evaluando el uso de bibliotecas como **Java Concurrent API** para simplificar la implementación de hilos. * El equipo de desarrollo ha sugerido explorar técnicas adicionales, como el uso de cachés en memoria, para complementar la optimización del rendimiento. |

|  |  |
| --- | --- |
| **OBJ-004** | **Validar el sistema mediante pruebas de funcionalidad y usabilidad, asegurando que cumple con los requisitos establecidos y que realmente contribuye a una mejor organización y a una mayor interacción entre los usuarios.** |
| **Versión** | 2.0 (13/03/2025) |
| **Autor** | María Victoria Nieto  María Gabriela Espinosa Alonso (colaboradora) |
| **Fuentes** | Estudiantes y Tutores |
| **Descripción** | El sistema de reservas de tutorías debe ser validado exhaustivamente para garantizar que cumple con los requisitos establecidos y que ofrece una experiencia de usuario óptima. Esto incluye pruebas de funcionalidad para verificar que todas las características operan correctamente, así como pruebas de usabilidad para asegurar que la interfaz es intuitiva, accesible y facilita la interacción entre estudiantes y tutores. El objetivo final es confirmar que el sistema contribuye a una mejor organización y a una comunicación más efectiva entre los usuarios. |
| **Subobjetivos** | * **OBJ-004.1:** Diseñar y ejecutar pruebas de funcionalidad para validar que todas las características del sistema operan según lo especificado (reservas, notificaciones, asignación de espacios, etc.). * **OBJ-004.2:** Realizar pruebas de usabilidad con usuarios reales (estudiantes, tutores y administradores) para evaluar la intuitividad y eficiencia de la interfaz. * **OBJ-004.3:** Identificar y documentar errores, fallos o áreas de mejora a partir de los resultados de las pruebas. * **OBJ-004.4:** Implementar correcciones y ajustes basados en el feedback obtenido durante las pruebas. * **OBJ-004.5:** Validar que el sistema cumple con los estándares de accesibilidad y experiencia de usuario (UX) establecidos. |
| **Importancia** | Muy Alta |
| **Urgencia** | Alta |
| **Estado** | En planificación |
| **Estabilidad** | Media |
| **Comentarios** | Las pruebas de usabilidad se realizarán con un grupo diverso de usuarios para captar diferentes perspectivas y necesidades. |

# Catálogo de Requisitos

## Requisitos de Información

|  |  |  |
| --- | --- | --- |
| **REQ-INF-001** | **Disponibilidad de información en tiempo real** | |
| **Versión** | 1.0 (23/10/2024) | |
| **Autores** | Camilo Alfonso Mora | |
| **Fuentes** | Estudiantes y Tutores | |
| **Objetivos asociados** | **OBJ-001:** Analizar los requisitos para el diseño de un sistema de reservas de tutorías universitarias.  **OBJ-002:** Crear el sistema de reservas de tutorías, incorporando un algoritmo de análisis de datos y funcionalidades. | |
| **Requisitos asociados** | **REQ-INF-001.1:** Mostrar la disponibilidad de horarios de tutoría en tiempo real.  **REQ-INF-001.2:** Ofrecer las notificaciones en tiempo real en caso de cambios de horario. | |
| **Descripción** | El sistema deberá proporcionar información relevante de forma actualizada y notificar cualquier cambio. | |
| **Datos específicos** | * **Frecuencia de actualización**: Cada 10 segundos o cada vez que ocurra un cambio en la disponibilidad. * **Fuente de datos**: Calendarios de tutores y estudiantes, registros del sistema. | |
| **Tiempo de vida** | **Medio** | **Máximo** |
| 6 meses a 1 año sin intervención manual para ajustes menores o mejoras. | 2 años antes de necesitar una actualización tecnológica significativa. |
| **Ocurrencias simult.** | **Medio** | **Máximo** |
| entre 300 y 400 usuarios simultáneos realizando consultas sobre la disponibilidad en tiempo real. | El sistema deberá soportar hasta 1000 usuarios simultáneos en momentos de alta demanda. |
| **Importancia** | Alta | |
| **Urgencia** | Alta | |
| **Estado** | Pendiente de verificación | |
| **Estabilidad** | Media | |
| **Comentarios** | El sistema deberá prever periodos de alta demanda, como los días previos a exámenes, para optimizar los recursos disponibles y evitar saturación.  La notificación en tiempo real debe activarse inmediatamente cuando se produzca cualquier cambio de disponibilidad para asegurar una experiencia de usuario eficiente y sin demoras. | |

|  |  |  |
| --- | --- | --- |
| **REQ-INF-002** | **Información detallada sobre tutores y estudiantes** | |
| **Versión** | 1.0 (23/10/2024) | |
| **Autores** | Camilo Alfonso Mora | |
| **Fuentes** | Estudiantes y Tutores | |
| **Objetivos asociados** | **OBJ-001:** Analizar los requisitos para el diseño de un sistema de reservas de tutorías universitarias.  **OBJ-002:** Crear el sistema de reservas de tutorías, incorporando un algoritmo de análisis de datos y funcionalidades. | |
| **Requisitos asociados** | **REQ-INF-002.1**: Visualización de perfiles de tutores  **REQ-INF-002.2**: Información sobre estudiantes para los tutores | |
| **Descripción** | El sistema deberá mostrar información detallada sobre los tutores, incluyendo su especialidad, materias que enseñan y horarios disponibles. | |
| **Datos específicos** | * **Información sobre tutores**: Nombre completo, especialidad, materias que enseñan, años de experiencia, horarios disponibles, evaluaciones de otros estudiantes, y modalidad (presencial o en línea). * **Información sobre estudiantes**: Nombre completo, carrera, semestre. | |
| **Tiempo de vida** | **Medio** | **Máximo** |
| 1 año con actualizaciones menores en la información de los perfiles, como cambios de horario o nuevas evaluaciones. | 2 años antes de necesitar una revisión completa de los perfiles para asegurar la precisión de la información y añadir nuevas funcionalidades. |
| **Ocurrencias simult.** | **Medio** | **Máximo** |
| hasta 500 consultas simultáneas de perfiles de tutores y estudiantes. | hasta 1000 consultas simultáneas durante picos de uso. |
| **Importancia** | Alta | |
| **Urgencia** | Alta | |
| **Estado** | Pendiente de verificación | |
| **Estabilidad** | Bajo | |
| **Comentarios** | La precisión de la información debe ser garantizada mediante un proceso regular de actualización de perfiles, realizado por los tutores y supervisado por el sistema.  Se recomienda incluir una funcionalidad para que los tutores puedan actualizar sus horarios de manera dinámica. | |

|  |  |  |
| --- | --- | --- |
| **REQ-INF-003** | **Registro de interacciones y tutorías anteriores** | |
| **Versión** | 1.0 (23/10/2024) | |
| **Autores** | Camilo Alfonso Mora | |
| **Fuentes** | Estudiantes y Tutores | |
| **Objetivos asociados** | **OBJ-001:** Analizar los requisitos para el diseño de un sistema de reservas de tutorías universitarias.  **OBJ-004:** Validar el sistema mediante pruebas de funcionalidad y usabilidad. | |
| **Requisitos asociados** | **REQ-INF-003.1**: Visualización del historial de tutorías  **REQ-INF-003.2:** Evaluaciones post-tutoría | |
| **Descripción** | El sistema deberá almacenar y mostrar un historial de tutorías programadas y completadas por cada estudiante. | |
| **Datos específicos** | * **Información a almacenar**: Fecha y hora de la tutoría, tutor asignado, materia tratada, modalidad (en línea o presencial), duración, y evaluaciones recibidas por ambas partes. * **Acceso al historial**: Tanto estudiantes como tutores podrán consultar el historial de tutorías previas, filtrado por fecha, materia o tutor. * **Almacenamiento**: Los registros deberán almacenarse por un periodo mínimo de 2 años y ser accesibles en cualquier momento desde la cuenta del usuario. * **Seguridad**: Los registros deben estar protegidos y solo visibles para el estudiante y el tutor involucrado, conforme a las normativas de privacidad. | |
| **Tiempo de vida** | **Medio** | **Máximo** |
| El sistema deberá mantener los registros de interacciones durante al menos 1 año para permitir revisiones posteriores. | Los registros podrán almacenarse hasta 2 años antes de ser archivados o eliminados, a menos que los estudiantes o tutores soliciten su conservación por más tiempo. |
| **Ocurrencias simult.** | **Medio** | **Máximo** |
| hasta 300 consultas simultáneas al historial de tutorías. | hasta 1500 consultas simultáneas durante periodos de alta demanda. |
| **Importancia** | Media | |
| **Urgencia** | Baja | |
| **Estado** | Pendiente de verificación | |
| **Estabilidad** | Media | |
| **Comentarios** | La posibilidad de revisar interacciones anteriores es clave para mejorar la calidad y efectividad de futuras tutorías.  Es importante realizar copias de seguridad periódicas de los registros para evitar la pérdida de datos. | |

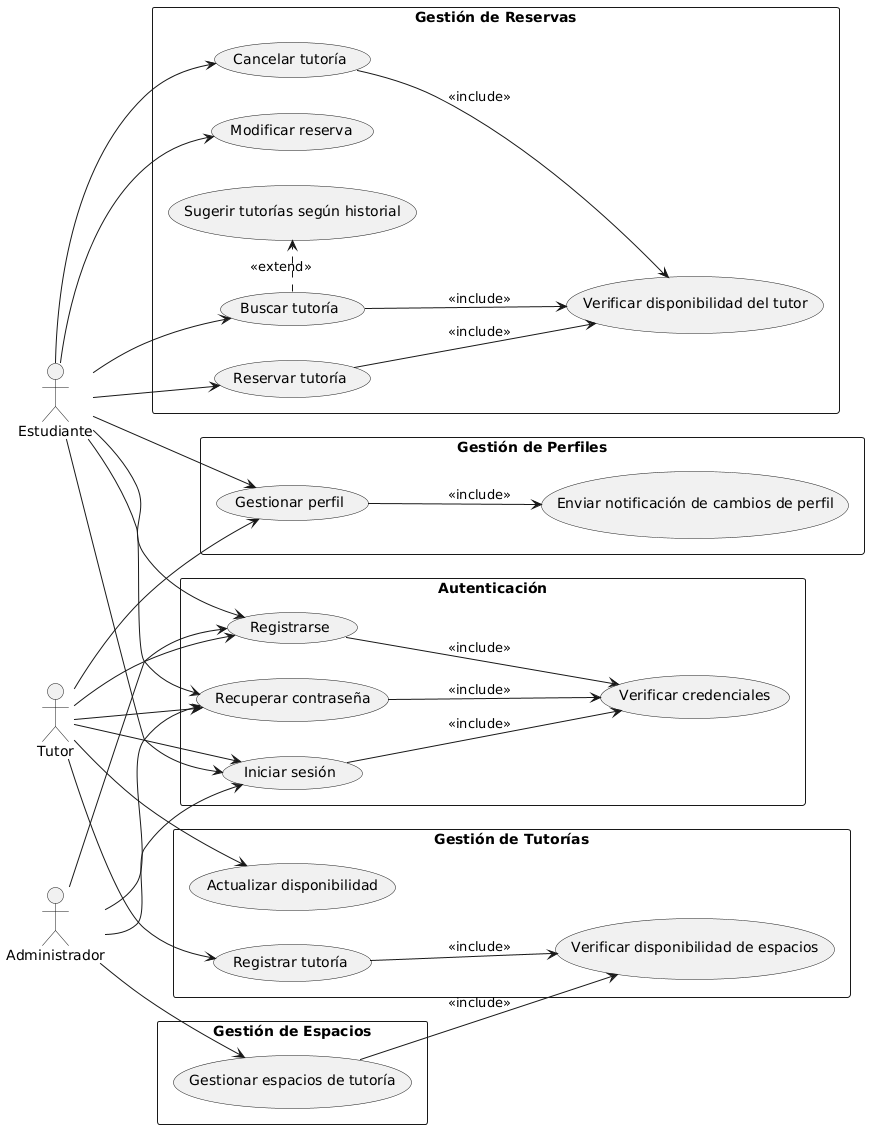
## Requisitos Funcionales

|  |  |  |
| --- | --- | --- |
| **REQ-FUN-001** | **Sistema de notificaciones** | |
| **Versión** | 1.0 (23/10/2024) | |
| **Autores** | Camilo Alfonso Mora | |
| **Fuentes** | Estudiantes y Tutores | |
| **Objetivos asociados** | **OBJ-001:** Analizar los requisitos para el diseño de un sistema de reservas de tutorías universitarias.  **OBJ-004:** Validar el sistema mediante pruebas de funcionalidad y usabilidad | |
| **Requisitos asociados** | **REQ-FUN-001.1**: Notificaciones de recordatorio  **REQ-FUN-001.2**: Notificaciones de cambios de disponibilidad  **REQ-FUN-001.3**: Personalización de notificaciones | |
| **Descripción** | El sistema deberá implementar un sistema de notificaciones que envíe alertas a los estudiantes y tutores sobre tutorías programadas o cambios en la disponibilidad. | |
| **Datos específicos** | **Tipos de notificaciones**:   * Programación de una tutoría confirmada. * Cambios en la disponibilidad de los tutores (cancelaciones, modificaciones de horario). * Recordatorios de tutorías próximas (con al menos 24 horas de antelación). * Evaluaciones pendientes tras finalizar una tutoría.   **Frecuencia de envío**: Inmediata para cambios en la disponibilidad y recordatorios automáticos según el calendario. | |
| **Tiempo de vida** | **Medio** | **Máximo** |
| 1 año antes de requerir mejoras en la infraestructura de notificaciones, para optimizar la capacidad y adaptarse a nuevas tecnologías. | 2 años antes de una revisión completa del sistema de notificaciones. |
| **Ocurrencias simult.** | **Medio** | **Máximo** |
| hasta 500 notificaciones simultáneas | hasta 2000 notificaciones simultáneas sin retraso o pérdida de información. |
| **Importancia** | Alta | |
| **Urgencia** | Alta | |
| **Estado** | Pendiente de verificación | |
| **Estabilidad** | Media | |
| **Comentarios** | Se podrían añadir notificaciones opcionales, como alertas de nuevas tutorías disponibles o cambios de última hora por parte de los tutores. | |

|  |  |  |
| --- | --- | --- |
| **REQ-FUN-002** | **Sistema de reservas en tiempo real** | |
| **Versión** | 1.0 (23/10/2024) | |
| **Autores** | Camilo Alfonso Mora | |
| **Fuentes** | Estudiantes y Tutores | |
| **Objetivos asociados** | **OBJ-001:** Analizar los requisitos para el diseño de un sistema de reservas de tutorías universitarias.  **OBJ-002:** Crear el sistema de reservas de tutorías, incorporando un algoritmo de análisis de datos y funcionalidades.  **OBJ-003:** Optimizar el rendimiento del sistema de reservas de tutorías, implementando técnicas de computación paralela con hilos en Java.  **OBJ-004:** Validar el sistema mediante pruebas de funcionalidad y usabilidad | |
| **Requisitos asociados** | **REQ-FUN-002.1**: Confirmación de reserva  **REQ-FUN-002.2**: Sistema de cancelación de tutorías  **REQ-FUN-002.3**: Filtro de búsqueda avanzada de tutores | |
| **Descripción** | El sistema deberá permitir a los estudiantes reservar tutorías en tiempo real, mostrando la disponibilidad actualizada de los tutores. | |
| **Datos específicos** | * **Funcionalidad principal**: El sistema deberá mostrar la disponibilidad de los tutores en tiempo real, permitiendo que los estudiantes reserven tutorías de forma inmediata. * **Proceso de reserva**: Los estudiantes podrán seleccionar un tutor, visualizar su horario disponible y confirmar la reserva con un solo clic. * **Cancelaciones**: Los estudiantes y tutores podrán cancelar tutorías con una notificación enviada a la contraparte, permitiendo la actualización inmediata del horario disponible. | |
| **Tiempo de vida** | **Medio** | **Máximo** |
| El sistema deberá mantenerse operativo y actualizado durante al menos 1 año, con revisiones periódicas de rendimiento. | Deberá ser capaz de operar durante 2 años sin requerir una reestructuración significativa, con capacidad para adaptarse a cambios menores en la interfaz o funciones. |
| **Ocurrencias simult.** | **Medio** | **Máximo** |
| hasta 500 reservas o consultas simultáneas. | hasta 2000 reservas o consultas simultáneas en periodos de alta demanda. |
| **Importancia** | Alta | |
| **Urgencia** | Alta | |
| **Estado** | Pendiente de verificación | |
| **Estabilidad** | Alta | |
| **Comentarios** | El sistema deberá ofrecer la opción de filtrar a los tutores por disponibilidad inmediata, especialidad, o modalidad (en línea o presencial). | |

|  |  |  |
| --- | --- | --- |
| **REQ-FUN-003** | **Actualización de disponibilidad de tutores** | |
| **Versión** | 1.0 (23/10/2024) | |
| **Autores** | Camilo Alfonso Mora | |
| **Fuentes** | Estudiantes y Tutores | |
| **Objetivos asociados** | **OBJ-001:** Analizar los requisitos para el diseño de un sistema de reservas de tutorías universitarias.  **OBJ-002:** Crear el sistema de reservas de tutorías, incorporando un algoritmo de análisis de datos y funcionalidades.  **OBJ-003:** Optimizar el rendimiento del sistema de reservas de tutorías, implementando técnicas de computación paralela con hilos en Java. | |
| **Requisitos asociados** | **REQ-FUN-003.1**: Actualización de información  **REQ-FUN-003.2:** Notificaciones automáticas de cambios  **REQ-FUN-003.3**: Configuración de disponibilidad recurrente | |
| **Descripción** | El sistema deberá permitir a los tutores actualizar su disponibilidad en tiempo real, para facilitar la coordinación de las tutorías. | |
| **Datos específicos** | * **Actualización de disponibilidad**: Cualquier cambio en la disponibilidad de los tutores deberá reflejarse en tiempo real en el sistema, incluyendo cancelaciones o nuevas franjas horarias. * **Interfaz de actualización**: El sistema deberá ofrecer una interfaz sencilla para que los tutores modifiquen, agreguen o eliminen franjas horarias de disponibilidad. * **Notificaciones**: Los estudiantes afectados por cambios en la disponibilidad recibirán notificaciones automáticas informándoles sobre la actualización. * **Seguridad**: Solo los tutores podrán modificar su propia disponibilidad, y todos los cambios quedarán registrados para referencia futura. | |
| **Tiempo de vida** | **Medio** | **Máximo** |
| El sistema deberá permitir la actualización continua de disponibilidad durante al menos 1 año sin requerir una reestructuración significativa. | El sistema deberá funcionar sin interrupciones durante 2 años antes de necesitar una revisión técnica importante. |
| **Ocurrencias simult.** | **Medio** | **Máximo** |
| hasta 300 actualizaciones simultáneas de disponibilidad por parte de los tutores en periodos normales. | hasta 1000 actualizaciones simultáneas sin afectar el rendimiento. |
| **Importancia** | Alta | |
| **Urgencia** | Alta | |
| **Estado** | Pendiente de verificación | |
| **Estabilidad** | Alta | |
| **Comentarios** | Los tutores deberán tener la opción de configurar su disponibilidad de forma recurrente (por ejemplo, los mismos días y horas cada semana).  El sistema deberá registrar todas las actualizaciones para propósitos de auditoría, en caso de disputas o problemas de coordinación. | |
| **REQ-FUN-004** | **Registro de tutores y estudiantes** | |
| **Versión** | 1.0 (23/10/2024) | |
| **Autores** | Camilo Alfonso Mora | |
| **Fuentes** | Estudiantes y Tutores | |
| **Objetivos asociados** | **OBJ-001:** Analizar los requisitos para el diseño de un sistema de reservas de tutorías universitarias.  **OBJ-002:** Crear el sistema de reservas de tutorías, incorporando un algoritmo de análisis de datos y funcionalidades. | |
| **Requisitos asociados** | **REQ-FUN-003.1**: Validación de datos de registro  **REQ-FUN-003.2:** Asignación de roles y permisos  **REQ-FUN-003.3**: Integración con sistemas universitarios | |
| **Descripción** | El sistema deberá permitir el registro detallado de tutores y estudiantes, recolectando información clave para garantizar un proceso de tutorías eficiente y personalizado. | |
| **Datos específicos** | * **Información requerida para los tutores**: Nombre completo, correo institucional, número de contacto, especialidades y materias que enseñan, disponibilidad horaria, modalidad de tutorías (en línea, presencial o ambas) * **Información requerida para los estudiantes:** Nombre completo, correo institucional, número de contacto, carreras o áreas de estudio, semestre cursado, materias en las que busca apoyo. | |
| **Tiempo de vida** | **Medio** | **Máximo** |
| Los datos de registro deberán mantenerse actualizados y ser revisados por los usuarios cada 6 meses para garantizar la vigencia de la información. | El sistema deberá mantener los registros por un periodo mínimo de 2 años antes de necesitar una depuración o actualización masiva de datos. |
| **Ocurrencias simult.** | **Medio** | **Máximo** |
| hasta 200 registros simultáneos de tutores y estudiantes sin afectar su rendimiento. | hasta 500 registros o actualizaciones simultáneas. |
| **Importancia** | Alta | |
| **Urgencia** | Alta | |
| **Estado** | Pendiente de verificación | |
| **Estabilidad** | Alta | |
| **Comentarios** | La información recolectada deberá cumplir con los estándares de protección de datos personales (GDPR o equivalente) para garantizar la privacidad y seguridad de los usuarios.  El sistema deberá incluir un proceso de validación para verificar la autenticidad de la información proporcionada, especialmente en el caso de los tutores.  Los usuarios podrán actualizar su información en cualquier momento, y se deberá registrar un historial de los cambios realizados. | |

## Diagrama de Casos de Uso:



## Definición de Actores

|  |  |
| --- | --- |
| **ACT - <01>** | Estudiante |
| **Versión** | 2.0 (15/03/2025) |
| **Autores** | María Gabriela Espinosa Alonso |
| **Fuentes** | Lucidchart |
| **Descripción** | El estudiante es el usuario principal que utiliza el sistema para gestionar sus reservas de tutorías. Puede cancelar o modificar reservas, recibir sugerencias de tutorías basadas en su historial y gestionar su perfil. |
| **Comentarios** | El estudiante tiene acceso a funciones de autenticación como registrarse, recuperar contraseña y verificar credenciales. También puede gestionar su perfil y recibir notificaciones sobre cambios en el mismo. |

|  |  |
| --- | --- |
| **ACT - <02>** | Tutor |
| **Versión** | 2.0 (15/03/2025) |
| **Autores** | María Gabriela Espinosa Alonso |
| **Fuentes** | Lucidchart |
| **Descripción** | El tutor es responsable de gestionar sus propias tutorías, actualizar su disponibilidad y verificar la disponibilidad de espacios para las tutorías. |
| **Comentarios** | El tutor tiene la capacidad de registrar nuevas tutorías y asegurarse de que los espacios estén disponibles para las sesiones programadas. |

|  |  |
| --- | --- |
| **ACT - <03>** | Administrador |
| **Versión** | 2.0 (15/03/2025) |
| **Autores** | María Gabriela Espinosa Alonso |
| **Fuentes** | Lucidchart |
| **Descripción** | El administrador es responsable de la gestión general de los espacios de tutoría dentro del sistema. |
| **Comentarios** | El administrador tiene la capacidad de gestionar y organizar los espacios disponibles para las tutorías, asegurando que estén correctamente asignados y disponibles para su uso. |

## Requisitos No Funcionales

|  |  |  |
| --- | --- | --- |
| **REQ-NF-001** | **Escalabilidad del sistema** | |
| **Versión** | 1.0 (23/10/2024) | |
| **Autores** | Camilo Alfonso Mora | |
| **Fuentes** | Estudiantes y Tutores | |
| **Objetivos asociados** | **OBJ-001:** Analizar los requisitos para el diseño de un sistema de reservas de tutorías universitarias.  **OBJ-002:** Crear el sistema de reservas de tutorías, incorporando un algoritmo de análisis de datos y funcionalidades.  **OBJ-003:** Optimizar el rendimiento del sistema de reservas de tutorías, implementando técnicas de computación paralela con hilos en Java. | |
| **Requisitos asociados** | **REQ-NF-001.1**: Monitoreo del rendimiento del sistema  **REQ-NF-001.2**: Optimización de consultas a la base de datos | |
| **Descripción** | El sistema deberá ser escalable para manejar un aumento en el número de usuarios sin afectar su rendimiento. | |
| **Datos específicos** | * **Tecnologías utilizadas**: El sistema deberá estar construido sobre una arquitectura que permita la escalabilidad, como microservicios o contenedores, para facilitar el aumento de capacidad de manera eficiente. * **Recursos de hardware**: La infraestructura deberá permitir la adición de recursos de hardware (como servidores) según la demanda, permitiendo el balanceo de carga y la distribución del tráfico. * **Escalabilidad vertical y horizontal**: El sistema deberá ser capaz de escalar verticalmente (aumentando la capacidad de los servidores existentes) y horizontalmente (agregando nuevos servidores) según sea necesario. | |
| **Tiempo de vida** | **Medio** | **Máximo** |
| El sistema deberá mantener su rendimiento óptimo con un crecimiento del 50% en el número de usuarios durante al menos 2 años. | Deberá ser capaz de soportar un crecimiento del 200% en el número de usuarios durante un periodo de 4 años sin necesidad de reestructuración completa. |
| **Ocurrencias simult.** | **Medio** | **Máximo** |
| hasta 3000 usuarios concurrentes sin degradar su rendimiento. | hasta 10,000 usuarios concurrentes durante picos de actividad, como al inicio de un nuevo semestre. |
| **Importancia** | Alta | |
| **Urgencia** | Media | |
| **Estado** | Pendiente de verificación | |
| **Estabilidad** | Alta | |
| **Comentarios** | Se recomienda realizar pruebas de carga periódicas para garantizar que el sistema se mantenga dentro de los parámetros de rendimiento deseados a medida que se añadan nuevos usuarios.  La escalabilidad deberá considerar tanto el crecimiento en la cantidad de usuarios como el aumento en la cantidad de datos almacenados, asegurando que la base de datos también pueda escalar según sea necesario. | |

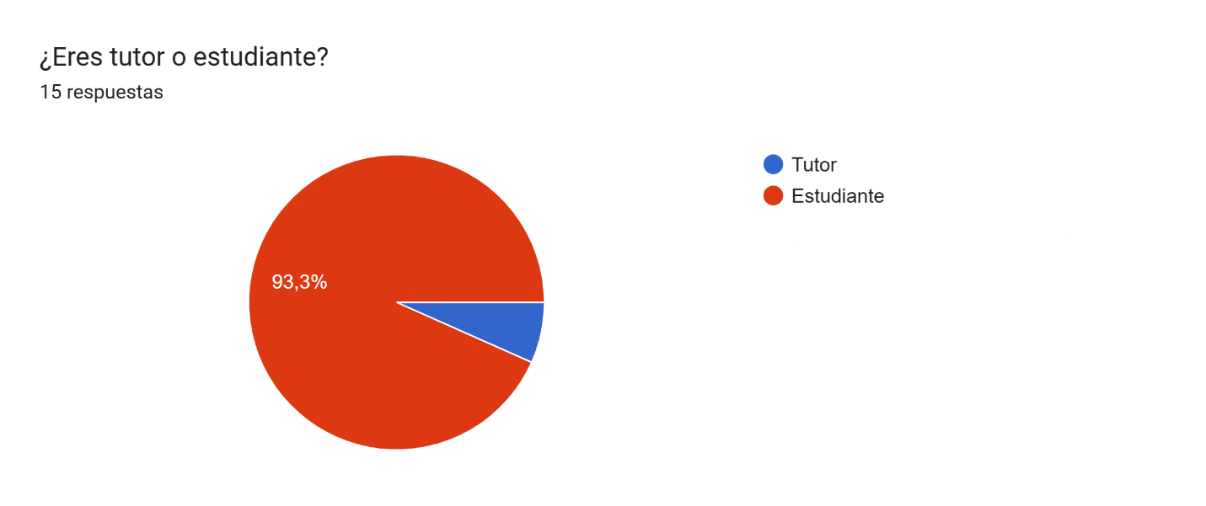
|  |  |  |
| --- | --- | --- |
| **REQ-NF-002** | **Seguridad y protección de datos** | |
| **Versión** | 1.0 (23/10/2024) | |
| **Autores** | Camilo Alfonso Mora | |
| **Fuentes** | Estudiantes y Tutores | |
| **Objetivos asociados** | **OBJ-001:** Analizar los requisitos para el diseño de un sistema de reservas de tutorías universitarias.  **OBJ-004:** Validar el sistema mediante pruebas de funcionalidad y usabilidad. | |
| **Requisitos asociados** | **REQ-NF-002.1**: Control de acceso y permisos  **REQ-NF-002.2**: Monitoreo de actividad sospechosa  **REQ-NF-002.3**: Políticas de privacidad y cumplimiento | |
| **Descripción** | El sistema deberá garantizar la seguridad de la información personal de los estudiantes y tutores, cumpliendo con las normativas de protección de datos. | |
| **Datos específicos** | * **Cifrado de datos**: Todos los datos sensibles (como información personal, académica y de contacto) deberán estar cifrados tanto en tránsito (usando HTTPS) como en reposo (en la base de datos). * **Autenticación y autorización**: Implementar un sistema de autenticación robusto (por ejemplo, autenticación de dos factores) para el acceso de tutores y estudiantes. Los permisos de acceso deberán estar claramente definidos para asegurar que los usuarios solo puedan acceder a la información que les corresponde. | |
| **Tiempo de vida** | **Medio** | **Máximo** |
| Las medidas de seguridad deben ser revisadas y actualizadas al menos una vez al año. | La infraestructura de seguridad debe ser capaz de soportar cambios y nuevas regulaciones sin una reestructuración completa cada 2 años. |
| **Ocurrencias simult.** | **Medio** | **Máximo** |
| El sistema deberá manejar al menos 3000 usuarios simultáneos sin comprometer las medidas de seguridad. | Deberá poder soportar hasta 10,000 usuarios concurrentes, garantizando que todas las sesiones estén protegidas. |
| **Importancia** | Alta | |
| **Urgencia** | Media | |
| **Estado** | Pendiente de verificación | |
| **Estabilidad** | Alta | |
| **Comentarios** | La capacitación del personal y de los usuarios sobre la importancia de la seguridad de datos y la privacidad será fundamental.  Implementar políticas de respaldo y recuperación de datos para minimizar la pérdida de información en caso de incidentes. | |

|  |  |  |
| --- | --- | --- |
| **REQ-NF-003** | **Disponibilidad y tiempo de respuesta** | |
| **Versión** | 1.0 (23/10/2024) | |
| **Autores** | Camilo Alfonso Mora | |
| **Fuentes** | Estudiantes y Tutores | |
| **Objetivos asociados** | **OBJ-001:** Analizar los requisitos para el diseño de un sistema de reservas de tutorías universitarias.  **OBJ-002:** Crear el sistema de reservas de tutorías, incorporando un algoritmo de análisis de datos y funcionalidades.  **OBJ-003:** Optimizar el rendimiento del sistema de reservas de tutorías, implementando técnicas de computación paralela con hilos en Java.  **OBJ-004:** Validar el sistema mediante pruebas de funcionalidad y usabilidad | |
| **Requisitos asociados** | **REQ-NF-003.1**: Optimización de consultas  **REQ-NF-003.2**: Estrategia de escalamiento  **REQ-NF-003.3**: Mantenimiento proactivo | |
| **Descripción** | El sistema deberá garantizar una disponibilidad mínima del 99% y un tiempo de respuesta de menos de 2 segundos para solicitudes de reservas o consulta de disponibilidad. | |
| **Datos específicos** | * **Métricas de rendimiento**: Se deberán implementar herramientas de monitoreo para medir la disponibilidad y el tiempo de respuesta del sistema en tiempo real. * **Planes de contingencia**: Se deberán establecer procedimientos para manejar caídas del sistema, incluyendo la identificación y solución rápida de problemas, y la comunicación efectiva con los usuarios. * **Mantenimiento programado**: El sistema deberá realizar mantenimiento de forma programada, con notificación anticipada a los usuarios, para minimizar el impacto en la disponibilidad. | |
| **Tiempo de vida** | **Medio** | **Máximo** |
| El sistema debe mantener la disponibilidad del 99% durante un periodo de 6 meses, con pruebas periódicas de rendimiento. | La solución debe ser capaz de mantener esta disponibilidad durante al menos 2 años, con revisiones anuales de la infraestructura. |
| **Ocurrencias simult.** | **Medio** | **Máximo** |
| al menos 3000 usuarios simultáneos con un tiempo de respuesta de menos de 2 segundos en un 95% de las solicitudes. | hasta 10,000 usuarios concurrentes, manteniendo el tiempo de respuesta especificado. |
| **Importancia** | Alta | |
| **Urgencia** | Media | |
| **Estado** | Pendiente de verificación | |
| **Estabilidad** | Alta | |
| **Comentarios** | La alta disponibilidad es crítica para el éxito del sistema, especialmente durante períodos de alta demanda, como el inicio de semestre o exámenes.  La implementación de copias de seguridad y sistemas de recuperación ante desastres será esencial para garantizar la disponibilidad continua del sistema. | |

# Resultados de la Encuesta

1. **¿Eres tutor o estudiante?**

La mayoría de los encuestados son estudiantes. Sin embargo, también hubo respuesta proviene de un tutor.



1. **¿Consideras que la forma en la que se gestionan actualmente las tutorías es óptima?**   
   La gran mayoría de los estudiantes no considera que la gestión actual de las tutorías sea óptima.

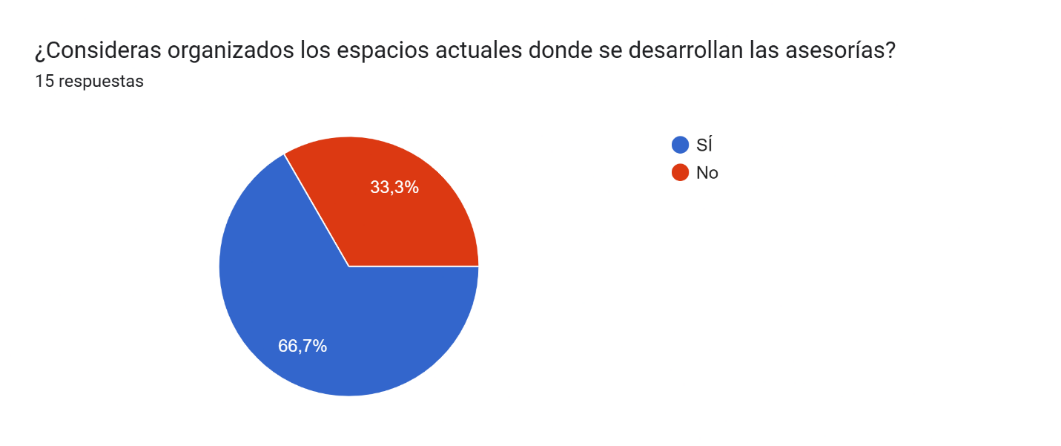
Gráfico de respuestas de formularios. Título de la pregunta:  ¿Consideras que la forma en la que se gestionan actualmente las tutorías es óptima?  
. Número de respuestas: 15 respuestas.

1. **¿Los medios de comunicación que se utilizan actualmente para organizar las tutorías te facilitan el acceso y te recuerdan adecuadamente sobre este beneficio?**

Al igual que la pregunta anterior, la mayoría de los estudiantes respondieron que no. Sin embargo, también hubo muchas respuestas que creen que los medios de comunicación que se usan actualmente si nos proveen un acceso asertivo a las asesorías.

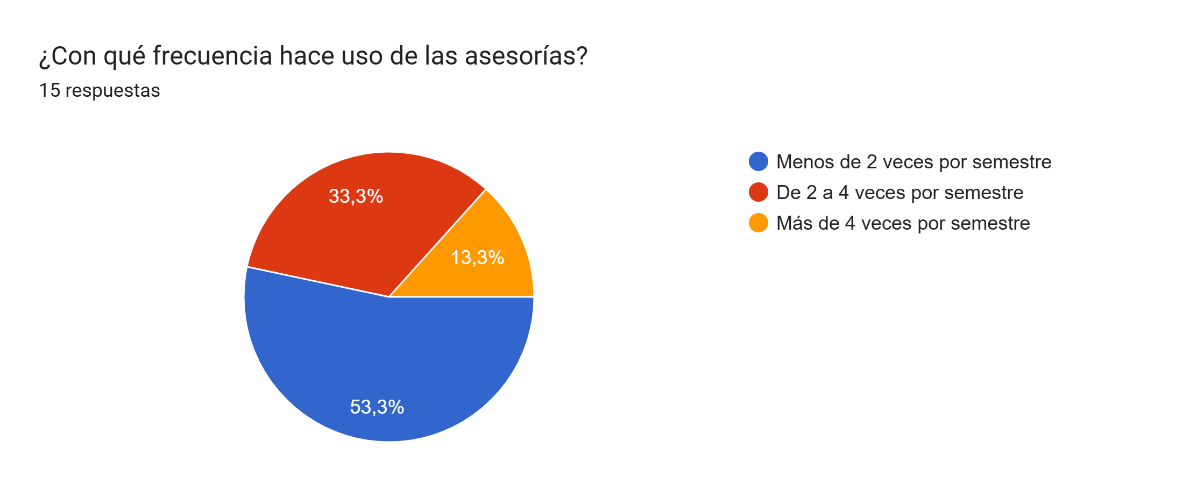
Gráfico de respuestas de formularios. Título de la pregunta:  ¿Los medios de comunicación que se utilizan actualmente para organizar las tutorías te facilitan el acceso y te recuerdan adecuadamente sobre este beneficio?  
. Número de respuestas: 15 respuestas.

1. **¿Consideras organizado los espacios actuales donde se desarrollan las asesorías?**   
   Las respuestas a esta pregunta se dividen casi por igual. Algunos estudiantes encuentran los espacios organizados, mientras que otros no.

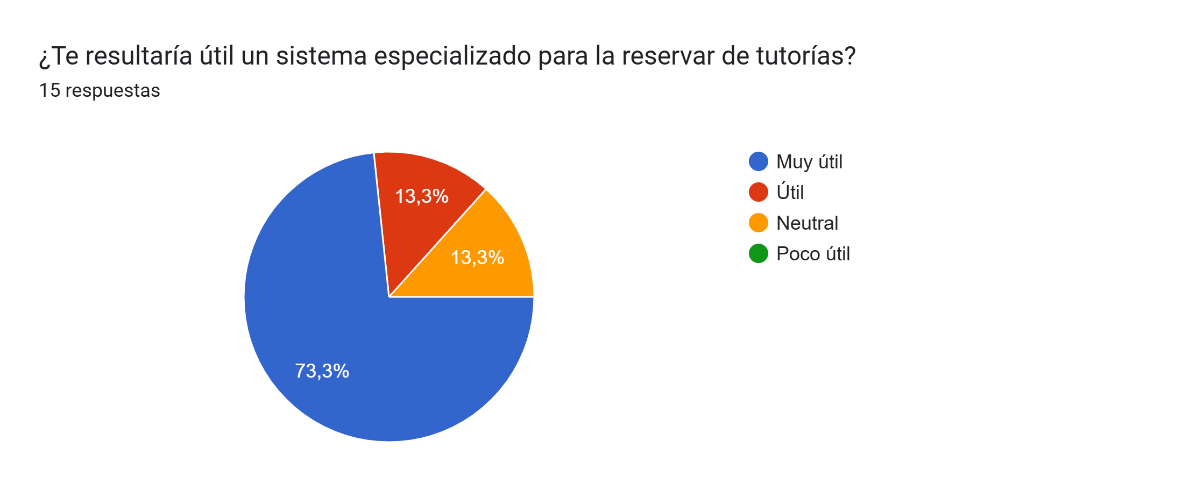


1. **¿Con qué frecuencia haces uso de las asesorías?**

La frecuencia de uso de las asesorías varía entre los estudiantes. Algunos las usan menos de 2 veces por semestre, otros entre 2 y 4 veces, y algunos más de 4 veces.



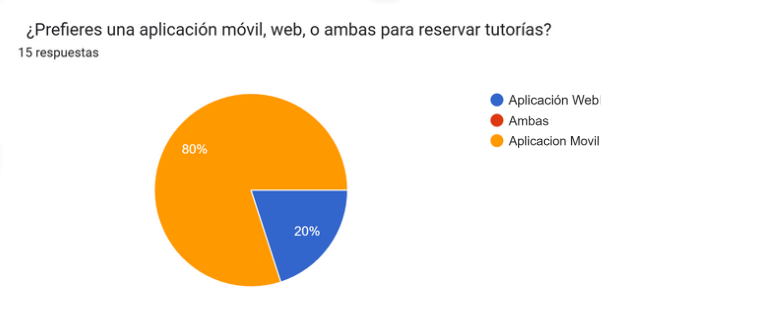
1. **¿Te resultaría útil un sistema especializado para la reserva de tutorías?**   
   La gran mayoría de los estudiantes considera muy útil un sistema de este tipo.1234... Solo un estudiante lo consideró simplemente "útil".



1. **¿Consideras que la creación de un sistema para gestionar las reservas de tutorías en la universidad podría mejorar la organización y facilitar la conexión entre estudiantes y tutores para acceder de manera más eficiente a las tutorías?**   
   La mayoría de los estudiantes están de acuerdo en que este tipo de sistema mejoraría la organización y la conexión entre estudiantes y tutores.

Gráfico de respuestas de formularios. Título de la pregunta: ¿Consideras que la creación de un sistema para gestionar las reservas de tutorías en la universidad podría mejorar la organización y facilitar la conexión entre estudiantes y tutores para acceder de manera más eficiente a las tutorías?
. Número de respuestas: 15 respuestas.

1. **¿Prefieres una aplicación móvil, web, o ambas para reservar tutorías?**La preferencia es que desean una aplicación movil.



1. **¿Qué funcionalidades crees que serían más útiles en un sistema de reservas de tutorías? (Selecciona todas las que apliquen)**

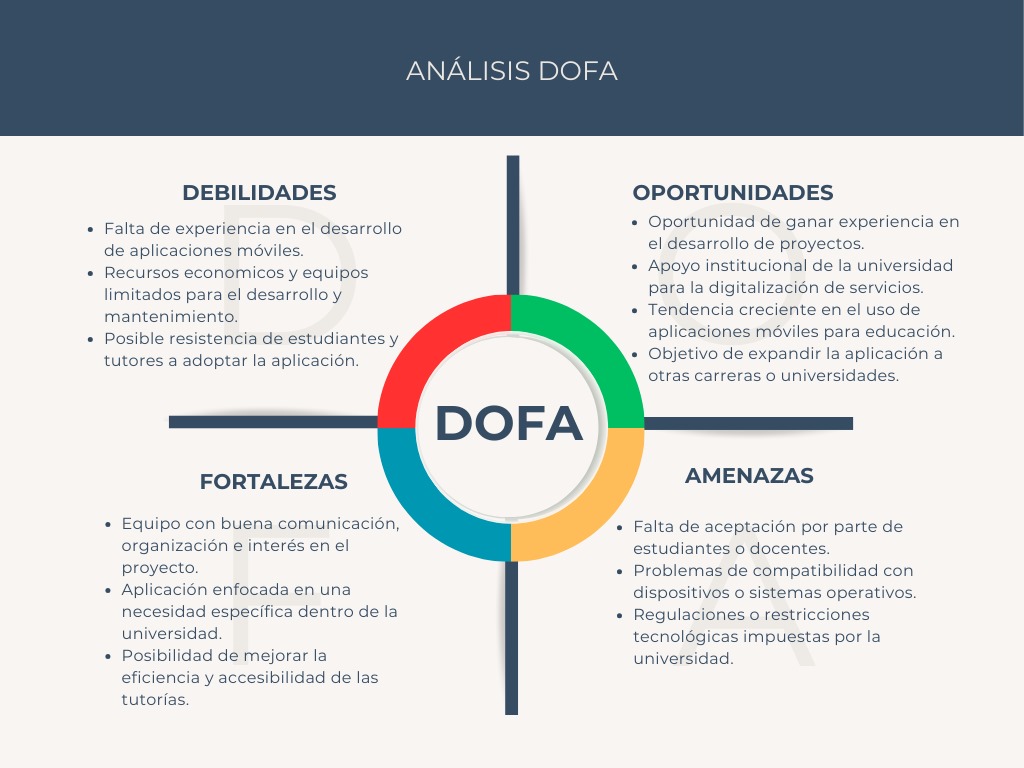
Gráfico de respuestas de formularios. Título de la pregunta: ¿Qué funcionalidades crees que serían más útiles en un sistema de reservas de tutorías? (Selecciona todas las que apliquen)  
. Número de respuestas: 15 respuestas.

1. **¿Qué tan probable es que utilices este sistema si se implementa?**   
   La mayoría de los estudiantes indica una alta probabilidad de usar el sistema.

Gráfico de respuestas de formularios. Título de la pregunta: ¿Qué tan probable es que utilices este sistema si se implementa?  
. Número de respuestas: 15 respuestas.

# Análisis de Riesgos

## Matriz DOFA:



# Glosario

* **Sistema de Gestión de Tutorías**: Plataforma digital que permite la organización, reserva, y seguimiento de las tutorías. Incluye funcionalidades como la disponibilidad en tiempo real y la asignación de espacios.
* **Interfaz de Usuario (UI):** Parte visual del sistema con la que los usuarios (tutores y estudiantes) interactúan. Se diseñará para ser intuitiva y de fácil uso, facilitando la gestión de las tutorías.
* **Experiencia de Usuario (UX**): Sensación general que tienen los usuarios al interactuar con el sistema. Incluye la facilidad de navegación, la comprensión de las funcionalidades y la eficiencia del sistema.
* **Plataforma Virtual**: Herramienta tecnológica que permite realizar tutorías de forma remota, utilizando aplicaciones como Zoom, Microsoft Teams, Google Meet, u otros servicios de videollamadas.
* **Automatización**: Proceso mediante el cual ciertas funciones del sistema, como la asignación de horarios o espacios, se realizan automáticamente sin intervención manual, optimizando el tiempo y reduciendo errores.
* **Escalabilidad**: Capacidad del sistema para adaptarse a un aumento en el número de usuarios o sesiones de tutoría sin comprometer su rendimiento o funcionalidad.

# Matriz de Riesgos

| **Nº** | **Categoría** | **Descripción del Riesgo** | **Probabilidad** | **Impacto** | **Plan de Mitigación / Acción** |
| --- | --- | --- | --- | --- | --- |
| 1 | Técnico | Fallas en la plataforma o herramientas utilizadas | Media | Alto | Establecer backups frecuentes y contar con herramientas alternativas |
| 2 | Requisitos | Cambios inesperados en los requisitos del sistema | Alta | Alto | Realizar reuniones periódicas con el cliente para validar requisitos |
| 3 | Recursos Humanos | Baja disponibilidad del equipo de desarrollo | Media | Medio | Establecer un cronograma flexible y roles compartidos |
| 4 | Cronograma | Retrasos en la entrega de funcionalidades | Media | Alto | Estimar tiempos con márgenes de contingencia y realizar seguimientos semanales |
| 5 | Seguridad | Pérdida de datos o accesos no autorizados | Baja | Alto | Implementar protocolos de seguridad, autenticación y control de accesos |
| 6 | Comunicación | Mala comunicación entre los miembros del equipo | Media | Medio | Utilizar herramientas colaborativas y realizar reuniones constantes |
| 7 | Validación | Errores no detectados en la etapa de pruebas | Media | Alto | Plan de pruebas exhaustivo y pruebas automatizadas |
| 8 | Financiero | Exceso en los costos estimados del proyecto | Baja | Medio | Llevar control detallado del presupuesto y buscar recursos adicionales si es necesario |
| 9 | Cliente/Usuario | Falta de compromiso del cliente en las etapas clave | Alta | Medio | Establecer cronograma de entregas parciales con validación obligatoria |
| 10 | Implementación | Dificultad para desplegar el sistema en el entorno del cliente | Baja | Medio | Hacer pruebas de integración previas y documentar bien el proceso de instalación |

# Arquitectura de Software

## Backend: Arquitectura por Capas Enfocada al Dominio

El backend está diseñado bajo una arquitectura por capas, con enfoque en el dominio, para garantizar separación de responsabilidades, mantenibilidad y escalabilidad. La estructura sigue el esquema mostrado en el diagrama a continuación:

Imagen de la pantalla de un celular con la imagen de un video juego

El contenido generado por IA puede ser incorrecto.

**Explicación:**

* **Web (Capa de Entrada)**  
  Contiene los controladores REST (@RestController) que reciben las peticiones del cliente y delegan la lógica al servicio correspondiente.
* **Domain (Lógica del Negocio)**  
  Se divide en varias partes:
  + dto: Objetos de transferencia de datos.
  + model: Modelos del dominio.
  + service: Lógica del negocio (@Service).
  + repository: Interfaz del repositorio que abstrae la persistencia.  
    Esta capa define la lógica central de la aplicación, sin depender de detalles técnicos de infraestructura.
* **Persistence (Capa de Infraestructura)**  
  Implementa los detalles de acceso a datos:
  + entity: Entidades JPA (@Entity).
  + mapper: Interfaces de mapeo entre entidades y modelos (@Mapper).
  + repository: Repositorios reales con @Repository y extendidos de JpaRepository.  
    Esta capa se encarga de la persistencia, implementando la interfaz definida en la capa de dominio.

Esta estructura permite un desarrollo limpio, testeable y desacoplado de frameworks específicos.

## Frontend: Arquitectura Cliente-Servidor

El frontend se basa en el modelo cliente-servidor, donde:

* **Cliente**  
  Aplicación móvil (prioridad según encuestas), que permite a estudiantes y tutores interactuar con el sistema de tutorías: búsqueda, reserva, gestión de disponibilidad, historial y retroalimentación.
* **Servidor**  
  Expone una API REST consumida por el cliente. Se encarga de manejar la lógica, acceder a la base de datos, y garantizar la seguridad y consistencia de la información.

# Bibliografia

* <https://core.ac.uk/reader/289971000>
* <http://biblioteca.usbbog.edu.co:8080/Biblioteca/BDigital/37874.pdf>
* <https://repositorio.ufps.edu.co/bitstream/handle/ufps/9186/TG_1151813_1151807.pdf?sequence=1&isAllowed=y>