



BENEMÉRITA UNIVERSIDAD AUTÓNOMA DE PUEBLA
FACULTAD DE CIENCIAS DE LA COMPUTACIÓN

Periodo Primavera 2025



PROGRAMACIÓN AVANZADA

Profesor: Jaime Alejandro Romero Sierra

Primer Reporte del Proyecto Final

“Plataforma de Tutorías para Estudiantes”

NOMBRE:

Valeria Pérez Chávez

MATRICULA:

202451031

16 de Marzo de 2025

Introducción

Descripción

El presente proyecto tiene como objetivo el desarrollo de una plataforma digital innovadora destinada a la gestión integral de tutorías académicas. Esta solución tecnológica facilitará la interacción eficiente entre estudiantes y tutores, optimizando la organización de sesiones, el seguimiento del progreso académico y la evaluación del desempeño mediante indicadores clave de rendimiento. La plataforma está diseñada para potenciar el proceso de aprendizaje a través de la personalización de contenidos y la implementación de herramientas de seguimiento, promoviendo así un desarrollo académico continuo y adaptado a las necesidades de cada usuario.

Impacto Social

Esta iniciativa busca eliminar las barreras de acceso a tutorías de calidad, ofreciendo un sistema inclusivo y accesible que permita a los estudiantes recibir apoyo educativo de manera flexible y adaptada a sus requerimientos. Al incorporar herramientas de análisis de desempeño, la plataforma no solo beneficia los estudiantes en su proceso de aprendizaje, sino que también fortalece la labor de los tutores al proporcionarles datos precisos y valiosos para mejorar sus estrategias pedagógicas. En un contexto global donde la educación personalizada adquiere una relevancia creciente, esta plataforma representa un avance significativo hacia la democratización del conocimiento y la mejora de los índices de éxito académico, contribuyendo así a la reducción de desigualdades educativas.

Justificación del Proyecto

En un contexto educativo marcado por la creciente demanda de formación personalizada, un número significativo de estudiantes enfrenta obstáculos para acceder a servicios de tutoría que se adapten a sus necesidades específicas. Esta carencia no solo limita su capacidad para consolidar conocimientos, sino que también incrementa el riesgo de rezago académico y abandono escolar, particularmente en comunidades con acceso limitado a recurso educativos o sistemas de apoyo insuficientes.

La presente iniciativa surge como respuesta a esta problemática, proponiendo una solución tecnológica integral que optimiza la gestión de tutorías y el monitoreo del avance académico. A través de una plataforma digital intuitiva y de fácil acceso, se busca brindar a los estudiantes un acompañamiento educativo adaptado a sus ritmos y estilos de aprendizaje, potenciando así su capacidad para alcanzar sus metas académicas.

Asimismo, la incorporación de herramientas analíticas permitirá una evaluación precisa del desempeño, facilitando la identificación de áreas de mejora y la implementación de estrategias pedagógicas más efectivas. Desde una perspectiva social, este proyecto no solo contribuye a cerrar brechas educativas, sino que también promueve la inclusión y la equidad, sentando las bases para un sistema educativo más justo y eficiente.

Objetivo General

Desarrollar una plataforma digital innovadora para la gestión integral de tutorías académicas, que optimice la interacción entre estudiantes y tutores, facilite el monitoreo del progreso académico y mejore los resultados educativos mediante herramientas de análisis de datos y personalización del aprendizaje.

Objetivos Específicos

1. Crear un sistema intuitivo y accesible que permita a los estudiantes identificar y conectarse con tutores especializados según sus requerimientos académicos.
2. Implementar herramientas de seguimiento y evaluación que generen métricas detalladas sobre el desempeño estudiantil, permitiendo la mejora continua de las estrategias de enseñanza.
3. Ofrecer un entorno de aprendizaje adaptable que brinde la posibilidad de acceder a tutorías personalizadas, ya sea de forma remota o presencial, en función de las preferencias y necesidades de los usuarios.
4. Facilitar la gestión de tutorías para los tutores, proporcionándoles herramientas que simplifiquen la organización de sesiones, el registro de avances y la adaptación de metodologías pedagógicas basadas en el rendimiento de los estudiantes.
5. Fomentar la inclusión y equidad educativa, garantizando que estudiantes de diversos contextos socioeconómicos tengan acceso a oportunidades de aprendizaje de calidad a través de una plataforma tecnológica accesible y de alto impacto.

Alcance del Proyecto

Funcionalidades del Sistema

El sistema contará con un conjunto de funcionalidades diseñadas para facilitar la gestión y el seguimiento de las tutorías académicas, así como para proporcionar herramientas de análisis que permitan optimizar el proceso de enseñanza-aprendizaje. A continuación, se describen en detalle las funcionalidades principales:

1. **Registro de Estudiantes:** *El sistema permitirá el registro de estudiantes, almacenando información esencial de cada estudiante como el nombre, la edad y las materias de interés. Esta información será fundamental para identificar las necesidades académicas de cada estudiante y asignar las tutorías correspondientes de manera personalizada.*
2. **Registro de Tutores:** *Se incluirá un módulo para el registro de tutores, donde se almacenarán datos como el nombre, la especialidad y la disponibilidad horaria. Esto permitirá gestionar de manera eficiente los recursos humanos disponibles y asegurar que las tutorías se asignen en función de la experiencia y disponibilidad de los tutores.*
3. **Agenda de Tutorías:** *El sistema contará con una agenda integrada que permitirá programar tutorías, registrando fecha, hora, materia y tutor asignado. Esta funcionalidad facilitará la organización de las sesiones y garantizará que tanto estudiantes como tutores tengan acceso a la información necesaria para su participación.*
4. **Búsqueda Avanzada:** *Se implementará una herramienta de búsqueda avanzada que permitirá filtrar información por estudiante, tutor y*

materia. Esto agilizará la consulta de datos específicos y mejorará la experiencia del usuario al acceder a la información relevante de manera rápida y precisa.

5. Generación de Reportes: *El sistema incluirá un módulo de generación de reportes que proporcionará **insights** clave para la toma de decisiones. Entre los reportes disponibles se encuentran los siguientes:*

- **Materias más solicitadas:** *Identificación de las materias con mayor demanda de tutorías.*
- **Tutores más activos:** *Listado de los tutores que han impartido más sesiones.*
- **Progreso académico de los estudiantes:** *Seguimiento del desempeño académico de los estudiantes a lo largo del tiempo.*

6. Análisis de Datos: *El sistema incorporará herramientas de análisis de datos que permitirán visualizar información relevante mediante gráficos y estadísticas. Entre las funcionalidades destacadas se encuentran:*

- **Gráficos de Distribución de Tutorías por Materia:** *Visualización de la frecuencia de tutorías por cada materia.*
- **Tendencia de Solicitudes de Tutorías a lo largo del tiempo:** *Análisis temporal de la demanda de tutorías para identificar patrones y necesidades futuras.*

Estas funcionalidades, en conjunto, brindarán una solución integral para la gestión de tutorías académicas, optimizando los procesos administrativos y proporcionando herramientas valiosas para el análisis y la mejora continua.

Limitaciones o Restricciones del Proyecto

El desarrollo del proyecto, aunque con un enfoque de impacto social y académico, enfrentará una serie de **limitaciones** y **restricciones** que es importante considerar para garantizar su éxito y viabilidad. Estas limitaciones se clasifican en cinco categorías principales:

1. Limitaciones Tecnológicas:

- El sistema dependerá de herramientas y tecnologías específicas para su implementación, como frameworks de desarrollo, bases de datos y servidores. La disponibilidad y compatibilidad de estas herramientas podrían representar un desafío, especialmente si se requieren actualizaciones o adaptaciones durante el proceso de desarrollo.
- La integración de funcionalidades avanzadas, como la generación de reportes y el análisis de datos, podría requerir el uso de librerías o software especializado, lo que podría incrementar la complejidad técnica del proyecto.

2. Limitaciones de Tiempo:

- El proyecto está sujeto a un cronograma ajustado, lo que limita la capacidad para implementar todas las funcionalidades deseadas en la fase inicial. Será necesario priorizar las

características más críticas y dejar otras para futuras actualizaciones.

- La disponibilidad de tiempo para dedicarle al desarrollo, pruebas y corrección de errores podría ser un factor limitante, especialmente al combinar esta tarea con otras responsabilidades académicas y personales.

3. Limitaciones de Recursos:

- El proyecto podría enfrentar restricciones presupuestarias, especialmente en lo relacionado con la adquisición de herramientas tecnológicas o servicios de hosting para el sistema.

4. Restricciones de Normativas y de Cumplimiento:

- Aunque el proyecto no está destinado a un entorno comercial en esta etapa, es importante considerar el manejo responsable de los datos personales de estudiantes y tutores. Esto implica implementar medidas básicas de privacidad y seguridad, lo que podría requerir de tiempo y esfuerzo profesional adicional.

5. Otras:

- La falta de acceso a datos históricos o reales para pruebas y validación podría limitar la precisión de algunas funcionalidades, como el análisis de tendencias de tutorías.
- La escalabilidad del sistema podría verse limitada si el número de usuarios crece significativamente, ya que el proyecto inicial no está diseñado para manejar grandes volúmenes de datos o usuarios simultáneos.

Estas limitaciones y restricciones no representan obstáculos insuperables, pero es fundamental identificarlas y planificar estrategias para mitigarlas. Esto permitirá garantizar que el proyecto se desarrolle de manera eficiente y cumpla con los objetivos establecidos, incluso bajo las condiciones actuales. Además, enfrentar estos desafíos contribuirá al desarrollo de habilidades transversales como la resolución de problemas, la gestión del tiempo y la adaptabilidad, que son esenciales en el ámbito profesional.

Beneficiarios

El sistema está diseñado para beneficiar a dos grupos de usuarios finales: **estudiantes** y **tutores**, quienes interactuarán directamente con la plataforma. Además, el proyecto tendrá un impacto positivo en la comunidad educativa en general, al facilitar el acceso a tutorías personalizadas y mejorar el rendimiento académico de los estudiantes. A continuación, se describen los beneficios y su relación con el sistema:

1. Estudiantes:

- Los estudiantes serán los principales beneficiarios del sistema, ya que les permitirá acceder a tutorías personalizadas en las materias que más necesiten. Podrán registrarse en la plataforma, indicar sus áreas de interés y programar sesiones con sus tutores favoritos.
- El sistema brindará herramientas para mejorar su rendimiento académico, como el seguimiento de su progreso y la posibilidad de identificar áreas de oportunidad.

- Además, los estudiantes se beneficiarán de la flexibilidad que ofrece la plataforma ya que podrán elegir horarios y tutores según sus necesidades y disponibilidad.

2. Tutores:

- Los tutores también serán usuarios clave del sistema, ya que les permitirá gestionar su disponibilidad, especialidades y sesiones programadas. Podrán registrarse en la plataforma, indicar sus áreas de profesionalización y recibir solicitudes de tutorías.
- El sistema les brindará herramientas para optimizar su tiempo y recursos, como la posibilidad de visualizar su agenda y generar reportes sobre su actividad.
- Además, los tutores se beneficiarán de la visibilidad que les brindará la plataforma, ya que podrán conectarse con estudiantes que requieran sus servicios de forma virtual.

3. Comunidad Educativa:

- Aunque no son usuarios directos del sistema, la comunidad educativa (*como las instituciones*) se verán beneficiadas indirectamente. Al facilitar el acceso a tutorías personalizadas, el sistema contribuirá a mejorar el rendimiento académico de los estudiantes y a reducir el número de reprobados.
- Las instituciones educativas podrán utilizar los datos generados por el sistema (*como las materias más solicitadas o el progreso de los estudiantes*) para tomar decisiones informadas y mejorar sus programas de apoyo académico.

Requerimientos del Sistema

Requerimientos Funcionales

Los requerimientos funcionales del sistema describen las capacidades y características que se debe implementar para cumplir con los objetivos establecidos. A continuación, se presenta una lista detallada de las funcionalidades que el sistema debe incluir:

1. Registro de Usuarios:

- El sistema permitirá el registro de dos tipos de usuarios: **estudiantes y tutores.**
- Se validarán campos obligatorios como nombre, edad, correo y contraseña.
- Se garantizará que los correos electrónicos sean únicos para evitar duplicados.

2. Autenticación y Seguridad:

- El sistema implementará un módulo de autenticación seguro que permita a los usuarios iniciar sesión con correo electrónico y contraseña.
- Se cifrarán las contraseñas antes de almacenarlas en la base de datos.
- Se incluirá un mecanismo de recuperación de contraseña mediante correo electrónico.

3. Agenda de Tutorías:

- El sistema permitirá a los estudiantes seleccionar fecha, hora, materia y tutor disponible para programar una tutoría.
- Se validará que no se sobrepongan las tutorías programadas para un mismo tutor en la misma fecha y hora.
- Se notificará a los estudiantes y tutores cuando una tutoría sea programada, modificada o cancelada.

4. Búsqueda Avanzada:

- El sistema incluirá un motor de búsqueda que permita filtrar información por estudiante, tutor o materia.
- Se garantizará que los resultados de búsqueda sean precisos y se muestren en un tiempo breve.

5. Generación de Reportes:

- El sistema generará reportes automáticos en formatos descargables (*PDF, Excel*).
- Se incluirá la opción de filtrar reportes por rangos de fecha, materia o tutores.
- Se garantizará que los reportes sean precisos y reflejen la información más reciente.

6. Análisis de Datos:

- El sistema incluirá herramientas de visualización de datos, como gráficos y tablas dinámicas.
- Se permitirá a los usuarios seleccionar los parámetros de análisis.

- Se garantizará que los datos mostrados sean consistentes y actualizados.

7. Notificaciones y Recordatorios:

- El sistema enviará notificaciones automáticas a estudiantes y tutores vía correo electrónico o a través de la plataforma.
- Se permitirá a los usuarios configurar sus preferencias de notificaciones.

8. Interfaz Intuitiva:

- El sistema contará con una interfaz de usuario responsive, que se adapte a diferentes dispositivos (*escritorio, móvil, tablet*).
- Se garantizará que la navegación sea intuitiva y que las acciones más comunes sean fáciles de realizar.

Estos requerimientos funcionales definen las capacidades técnicas que el sistema debe implementar para garantizar su correcto funcionamiento y cumplir con las necesidades de los usuarios.

Requerimientos No Funcionales

Los requerimientos no funcionales del sistema describen los atributos y características que aseguran su eficiencia, seguridad y usabilidad. Estos aspectos complementan las funcionalidades del sistema y son esenciales para su correcto funcionamiento y aceptación por parte de los usuarios. A continuación, se detallan:

1. Rendimiento:

- El sistema soportará un mínimo de 100 usuarios concurrentes sin experimentar una degradación en su rendimiento.

- Las operaciones críticas, como búsquedas y consultas, se completarán en un tiempo máximo de 2 segundos.
- La plataforma deberá mantener una disponibilidad del 99.9%, con un tiempo de respuesta promedio inferior a 500 ms.

2. Seguridad:

- El sistema implementará medidas de seguridad básicas, como el cifrado de contraseñas y la protección contra ataques comunes (*por ejemplo, inyección SQL*).
- Los datos personales de los usuarios estarán protegidos, y el sistema cumplirá con normativas de privacidad y protección de datos.
- Se incluirá un mecanismo de autenticación seguro, como la validación de dos factores (2FA), para acceder a funciones críticas.

3. Usabilidad:

- El sistema contará con una interfaz de usuario intuitiva y fácil de usar, que permitirá a los usuarios realizar sus tareas con un mínimo de clicks y esfuerzo.
- La plataforma será accesible para usuarios con discapacidades, cumpliendo con estándares de accesibilidad.
- Se proporcionará una guía de usuario y tutoriales integrados para facilitar la adaptación del sistema.

4. Escalabilidad:

- El sistema estará diseñado para ser escalable, permitiendo un crecimiento futuro en el número de usuarios y datos sin requerir cambios significativos en las bases.
- La plataforma podrá manejar un aumento de 50% en la carga de trabajo sin afectar su rendimiento.

5. Compatibilidad:

- El programa será compatible con los navegadores web más utilizados y en sus versiones más recientes.
- La plataforma funcionará correctamente en dispositivos móviles y tablets, con una interfaz responsive.

6. Mantenibilidad:

- El código del sistema estará documentado y seguirá buenas prácticas de desarrollo, lo que facilitará su mantenimiento y actualización en el futuro.
- Se incluirá un sistema de registro de errores que permitirá identificar y solucionar problemas de manera eficiente.

7. Portabilidad:

- El sistema será portable, lo que permitirá su implementación en diferentes entornos sin requerir cambios significativos.

Estos requerimientos no funcionales asegurarán que el sistema no solo cumpla con sus objetivos funcionales, sino que también sea eficiente, seguro, usable y preparado para su crecimiento a futuro.

Diagramas LMS

Variables a Ocupar

1. Clase: Usuario

- ATRIBUTOS:

- `id` (*String*): Identificador único del usuario.
- `nombre` (*String*): Nombre completo del usuario.
- `correo` (*String*): Correo electrónico del usuario.
- `contrasena` (*String*): Contraseña del usuario.
- `fecha_registro` (*DateTime*): Fecha en la que el usuario se registró al sistema.

- MÉTODOS:

- `iniciar_sesion(correo, contrasena)`: Permite al usuario iniciar sesión en el sistema.
- `cerra_sesion()`: Permite al usuario cerrar sesión.
- `cambiar_contrasena(nueva_contrasena)`: Permite al usuario cambiar su contraseña.

2. Clase: Estudiante(Usuario)

- ATRIBUTOS:

- `edad` (*Int*): Edad del estudiante.
- `materias_interes` (*List<String>*): Lista de materias de interés del estudiante.
- `tutorias_programadas` (*List<Tutoria>*): Lista de tutorías programadas para el estudiante.
- `progreso_academico` (*Map<String, float>*): Mapa que relaciona materias con calificaciones.

- **MÉTODOS:**

- `programar_tutoria(fecha, hora, materia, tutor)`:
Permite al estudiante programar una tutoría.
- `ver_progreso()`: Muestra el progreso académico del estudiante.

3. Clase: Tutor(Usuario)

- **ATRIBUTOS:**

- `especialidad` (*String*): Área de especialización del tutor.
- `disponibilidad` (*List<DateTime>*): Lista de fechas y horas disponibles para tutorías.
- `tutorias_asignadas` (*List<Tutoria>*): Lista de tutorías asignadas al tutor.
- `calificacion_promedio` (*Float*): Calificación promedio del tutor basada en las calificaciones recibidas.

- **MÉTODOS:**

- `actualizar_disponibilidad(fecha, hora)`: Permite al tutor actualizar su disponibilidad.
- `ver_solicitudes_tutoria()`: Muestra las solicitudes de tutorías recibidas por el tutor.

4. Clase: Tutoría

- **ATRIBUTOS:**

- `id_tutoria` (*String*): Identificador único de tutoría.
- `fecha` (*DateTime*): Fecha de la tutoría.
- `hora` (*DateTime*): Hora de la tutoría.
- `materia` (*String*): Materia a impartir en la tutoría.

- `estudiante` (*Estudiante*): Estudiante que solicita la tutoría.
 - `tutor` (*Tutor*): Tutor asignado para la tutoría.
 - `estado` (*String*): Estado actual de la tutoría (Programada, En progreso, Finalizada, Cancelada).
- **MÉTODOS:**
- `crear_tutoria(estudiante, tutor, fecha, hora, materia)`: Crea una nueva tutoría.
 - `cancelar_tutoria()`: Cancela la tutoría programada.
 - `enviar_reordatorio()`: Envía un recordatorio de la tutoría al estudiante y al tutor.

5. Clase: Reporte

- **ATRIBUTOS:**
- `id_reporte` (*String*): Identificador único del reporte.
 - `tipo_reporte` (*String*): Tipo de reporte(Ejemplo, “Materias más solicitadas”).
 - `fecha_generacion` (*DateTime*): Fecha en que se generó el reporte.
 - `datos` (*List<Object>*): Datos utilizados para generar el reporte.
- **MÉTODOS:**
- `generar_reporte(tipo_reporte, datos)`: Genera un reporte basado en los datos proporcionados.
 - `exportar_reporte(formato)`: Exporta el reporte en un formato específico (PDF, Excel).

6. Clase: Sistema

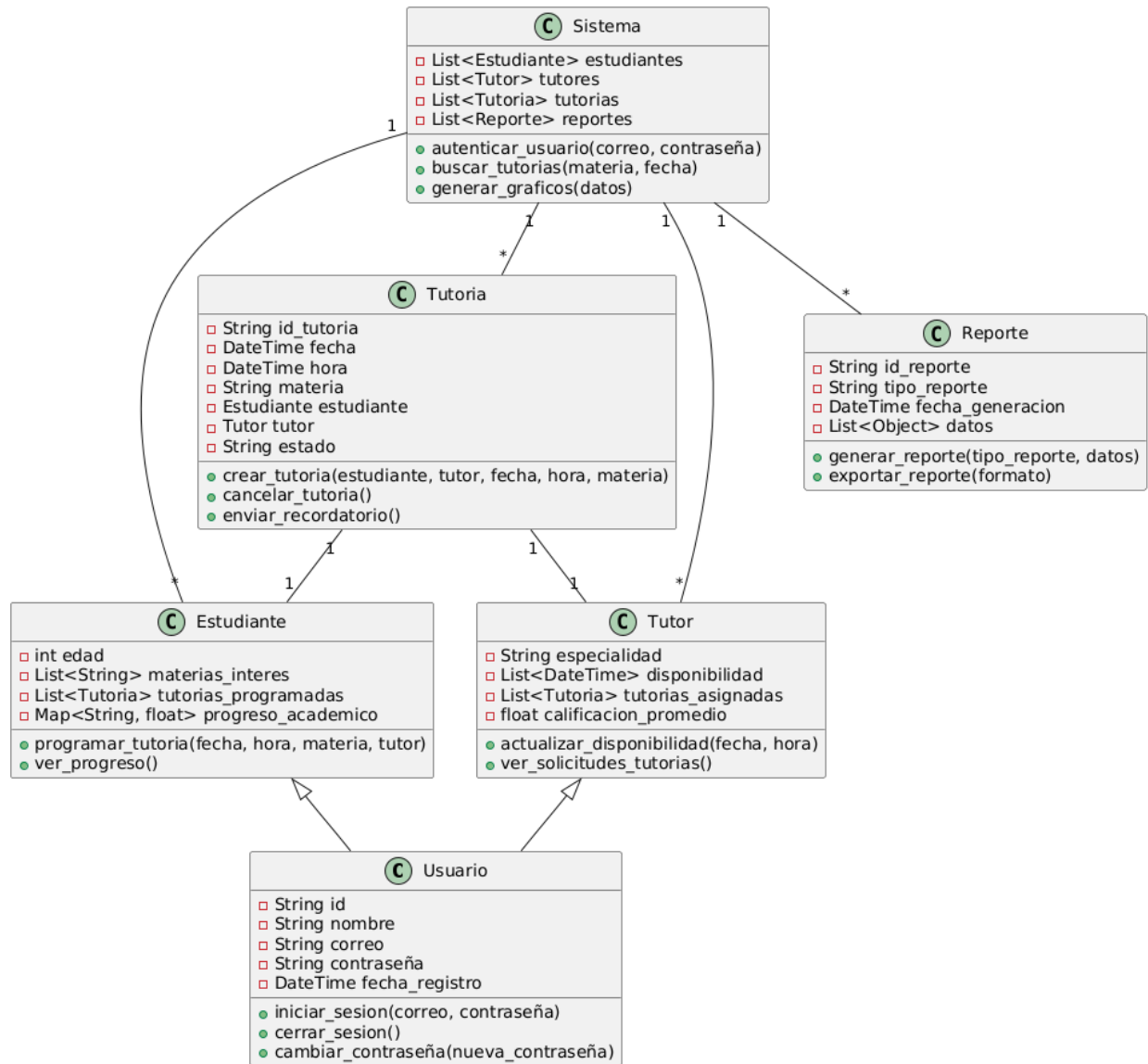
- ATRIBUTOS:

- `estudiantes` (*List<Estudiante>*): Lista de estudiantes registrados.
- `tutores` (*List<Tutor>*): Lista de tutores registrados.
- `tutorias` (*List<Tutoria>*): Lista de tutorías programadas.
- `reportes` (*List<Reporte>*): Lista de reportes generados.

- MÉTODOS:

- `autenticar_usuario(correo, contrasena)`: Autentica a un usuario en el sistema.
- `buscar_tutorias(materia, fecha)`: Buscar tutorías disponibles según la materia y fecha.
- `generar_graficos(datos)`: Generar gráficos basados en los datos proporcionados.

Diagrama de Modelo



Tecnologías a Utilizar

El desarrollo del sistema se llevará a cabo utilizando las siguientes herramientas, seleccionadas por su facilidad de uso y compatibilidad con los requerimientos del proyecto:

1. Lenguaje de programación:

- **Python:** Se eligió Python por su sintaxis clara y legible, lo que facilita el desarrollo rápido y mantenible. Además, cuenta con una amplia gama de librerías y frameworks que aceleran la implementación de funcionalidades.

2. Frameworks y librerías:

- **Tkinter:** Framework para interfaces gráficas de usuario (GUI) que permitirá desarrollar las ventanas y componentes visuales del sistema, como la ventana de inicio de sesión, el dashboard de estudiantes y tutores, y la ventana de programación de tutorías.
- **Pandas:** Librería para el análisis y manipulación de datos, útil en la generación de reportes como "Materias más solicitadas" o "Tutores más activos".
- **Matplotlib:** Librería para la generación de gráficos y visualización de datos, ideal para mostrar tendencias y distribuciones de tutorías.

3. Herramientas de desarrollo:

Para el desarrollo del sistema, se utilizará Visual Studio Code (VS Code) como editor de código principal. VS Code es una herramienta ligera y altamente personalizable, con soporte nativo para Python y extensiones como Python (soporte oficial), Pylance (autocompletado avanzado) y GitLens (integración con Git), que facilitan el desarrollo y la depuración del código. Para el manejo de datos, se empleará SQLite, una base de datos ligera y embebida que no requiere configuración adicional, lo que la hace ideal para proyectos pequeños o medianos. Además, SQLite viene integrada con Python, lo que simplifica su implementación. Finalmente, se utilizará Git como sistema de control de versiones para gestionar los cambios en el código, junto con GitHub, una plataforma que aloja el repositorio del proyecto y facilita la colaboración y el seguimiento de issues.

Conclusiones

Este proyecto busca desarrollar un sistema de gestión de tutorías académicas que facilite la interacción entre estudiantes y tutores, optimizando la organización y el seguimiento de las sesiones. A través de funcionalidades como el registro de usuarios, la programación de tutorías, la generación de reportes y el análisis de datos, se espera mejorar el rendimiento académico de los estudiantes y optimizar el tiempo de los tutores. Además, al enfocarse en un sistema accesible y eficiente, se busca generar un impacto social positivo, promoviendo una educación más inclusiva y personalizada. Este proyecto no solo demuestra la aplicación de conocimientos técnicos, sino también el compromiso con la resolución de problemas reales y la creación de soluciones tecnológicas con propósito.