**Requerimientos Funcionales**

* El sistema permitirá a los estudiantes visualizar
* El sistema permitirá a los docentes registrar calificaciones, incluyendo la posibilidad de generar reportes de calificaciones.
* El sistema permitirá a los estudiantes podrán consultar sus horarios académicos de forma detallada
* El sistema facilitará la matrícula de asignaturas, incluyendo la selección de cursos, la generación de horarios
* Los estudiantes podrán consultar la oferta académica disponible y obtener información detallada sobre cada materia
* El sistema enviará notificaciones a los estudiantes sobre eventos importantes como cambios de horario, resultados de exámenes,
* El sistema permitirá a los estudiantes gestionar su perfil personal, cambiar contraseñas, actualizar datos y solicitar certificados académicos.
* El sistema permitirá a los estudiantes realizar un seguimiento de su progreso académico, visualizando su historial académico y estableciendo metas.

**Requerimientos No Funcionales**

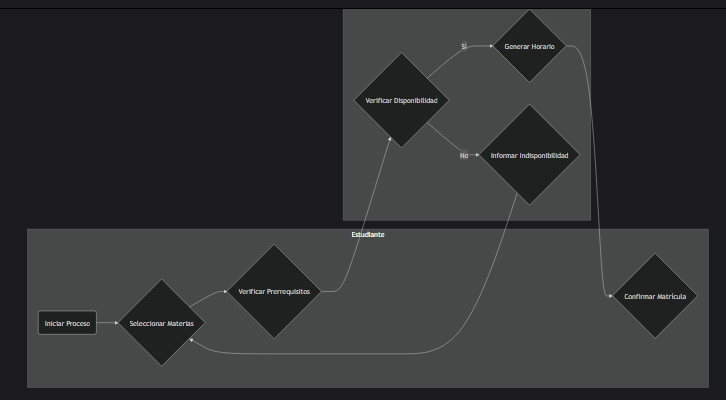
* La interfaz de usuario debe ser intuitiva para los estudiantes
* El sistema debe permitir la integración con calendarios personales como Google Calendar o Outlook para facilitar la gestión de tareas y eventos académicos.
* El sistema debe proporcionar herramientas de análisis de datos para que los estudiantes puedan visualizar su progreso académico.

Diagrama de clases

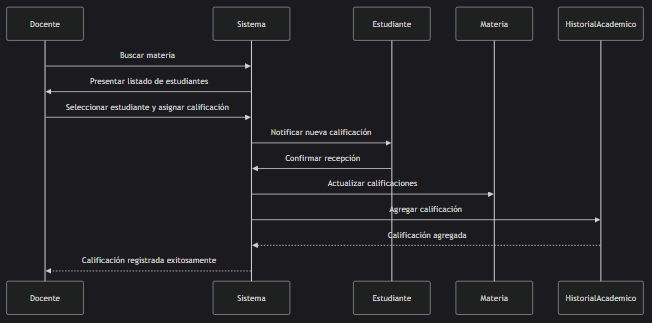
Diagrama

Descripción generada automáticamente

**Diagrama de Actividades**



**Diagrama de Secuencia**

  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
Toma el rol de un estudiante de 8vo semestre el cual tiene un trabajo de hacer un diagra de caso de uso con un trabajo el cual es:

Una universidad te contrata para desarrollar su software de Academico, para esto debe tener en cuenta el patron de diseño MVC, en una aplicacion de Spring boot, aplicando los conocimientos adquiridos

Este diagrama es:

Diagrama de Clases

El diagrama de clases es fundamental para visualizar las entidades y sus relaciones en nuestro sistema. Basándonos en los modelos de datos propuestos anteriormente, podríamos tener un diagrama como el siguiente:

Explicación:

\* Estudiante: Con atributos como ID, nombre, correo electrónico, etc.

\* Docente: Similar al estudiante, pero con atributos específicos como departamento y área de especialización.

\* Materia: Con atributos como código, nombre, descripción, créditos, etc.

\* Horario: Relaciona una materia con un día y horario específico.

\* Calificación: Guarda la calificación de un estudiante en una materia específica.

\* Matrícula: Representa la inscripción de un estudiante en una materia.

\* Notificación: Contiene información sobre las notificaciones enviadas a los usuarios.

\* Perfil: Contiene información adicional sobre el perfil del usuario.

\* HistorialAcadémico: Guarda el historial académico de un estudiante.

\* Meta: Representa una meta establecida por un estudiante.

Relaciones:

\* Estudiante se relaciona con Matrícula, Calificación, HistorialAcadémico y Meta.

\* Docente se relaciona con Materia.

\* Materia se relaciona con Horario, Calificación y Matrícula.

\* Horario se relaciona con Materia.

\* Calificación se relaciona con Estudiante y Materia.

\* Matrícula se relaciona con Estudiante y Materia.

Realiza este diagrama en este tipo de formato

graph TD Estudiante((Estudiante)) -->|Se inscribe en| Matrícula Matrícula -->|Asigna horario| Horario Horario -->|Se dicta en| Materia