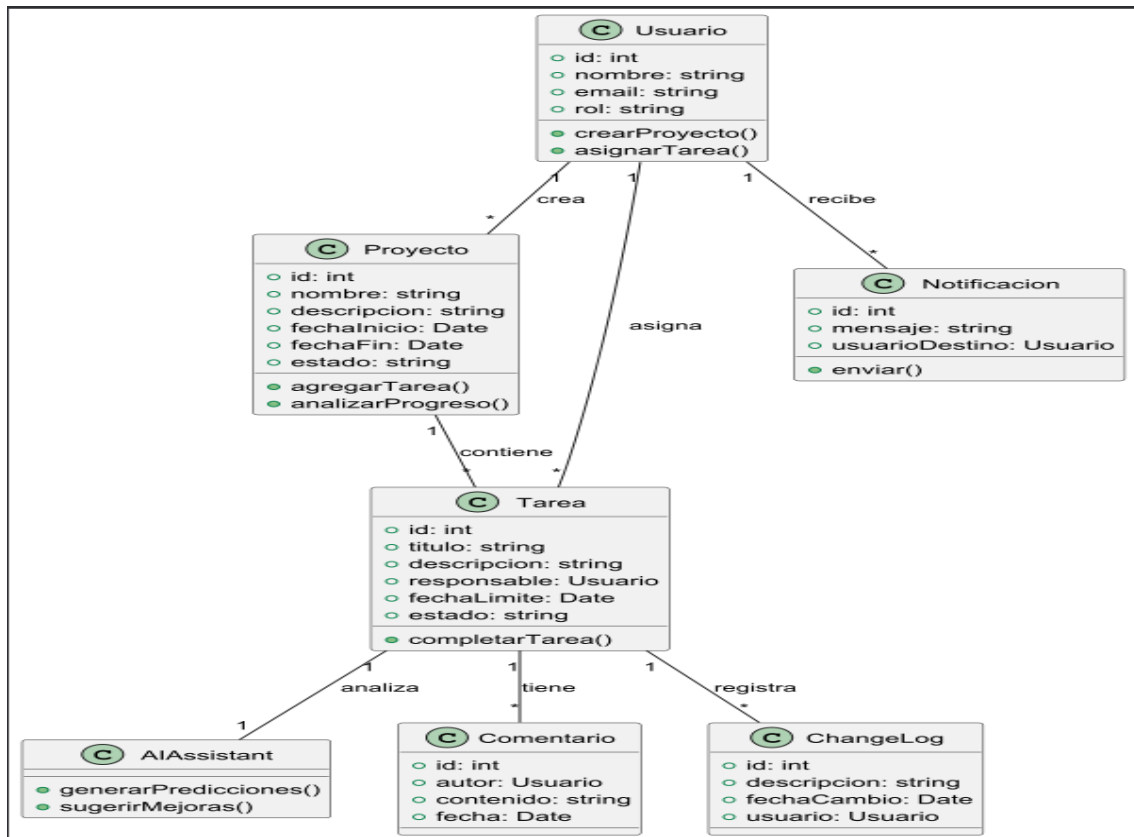


Explicación del Diagrama de Clases

Introducción

El diagrama de clases representa la estructura de la plataforma de gestión de proyectos. Este modelo define las principales entidades, sus atributos, métodos y las relaciones entre ellas, facilitando el diseño modular y la escalabilidad del sistema.



Clases y sus Funcionalidades

1. Usuario

- Representa a los usuarios del sistema con atributos como id, nombre, email y rol.
- Puede **crear proyectos** y **asignar tareas** a otros usuarios.
- Relación: Un usuario puede crear múltiples proyectos y asignar varias tareas.

2. Proyecto

- Define los proyectos con propiedades como id, nombre, descripción, fechaInicio, fechaFin y estado.
- Contiene métodos para **agregar tareas** y **analizar progreso**.
- Relación: Un proyecto puede contener varias tareas.

3. Tarea

- Representa una unidad de trabajo con atributos como id, título, descripción, responsable, fechaLímite y estado.
- Puede ser **completada** por su responsable.
- Relación: Un proyecto puede tener varias tareas y cada tarea es asignada a un usuario.

4. Notificación

- Se utiliza para informar a los usuarios sobre eventos relevantes.
- Contiene atributos como id, mensaje y usuarioDestino.
- Relación: Un usuario puede recibir varias notificaciones.

5. Comentario

- Permite agregar anotaciones a una tarea.
- Contiene atributos como id, autor, contenido y fecha.
- Relación: Una tarea puede tener varios comentarios.

6. AI Assistant

- Componente de inteligencia artificial encargado de analizar tareas.
- Puede **generar predicciones y sugerir mejoras**.
- Relación: Se asocia con la clase Tarea para optimizar su gestión.

7. ChangeLog

- Registra cambios en las tareas o proyectos.
- Contiene atributos como id, descripción, fechaCambio y usuario.
- Relación: Cada tarea puede generar múltiples registros de cambios.

Relaciones Clave

- **One to many** entre Usuario y Proyecto, ya que un usuario puede crear varios proyectos.
- **One to many** entre Proyecto y Tarea, permitiendo que cada proyecto tenga varias tareas.
- **One to many** entre Tarea y Comentario, ya que cada tarea puede recibir varios comentarios.

- **One to many** entre Tarea y ChangeLog, registrando cada cambio.
- **One to many** entre Usuario y Notificación, permitiendo que los usuarios reciban múltiples notificaciones.
- **One to One** entre Tarea y AI Assistant, ya que cada tarea puede ser analizada por la IA.

Conclusión

Este diagrama de clases proporciona una estructura clara y modular de la plataforma de gestión de proyectos. Define la interacción entre los componentes clave, asegurando un diseño escalable, mantenible y fácil de extender en el futuro.