



# Mini\_Proyecto

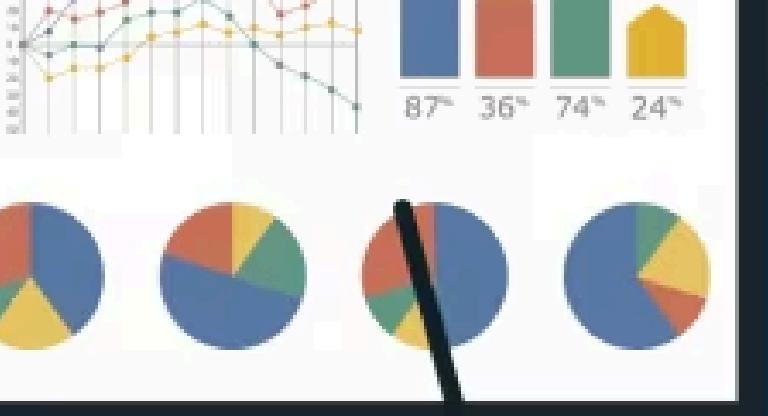
# Sistema de consultas

Flores Facundo Alexander  
Garcia Venegas Isaac  
Pérez Morales David Israel



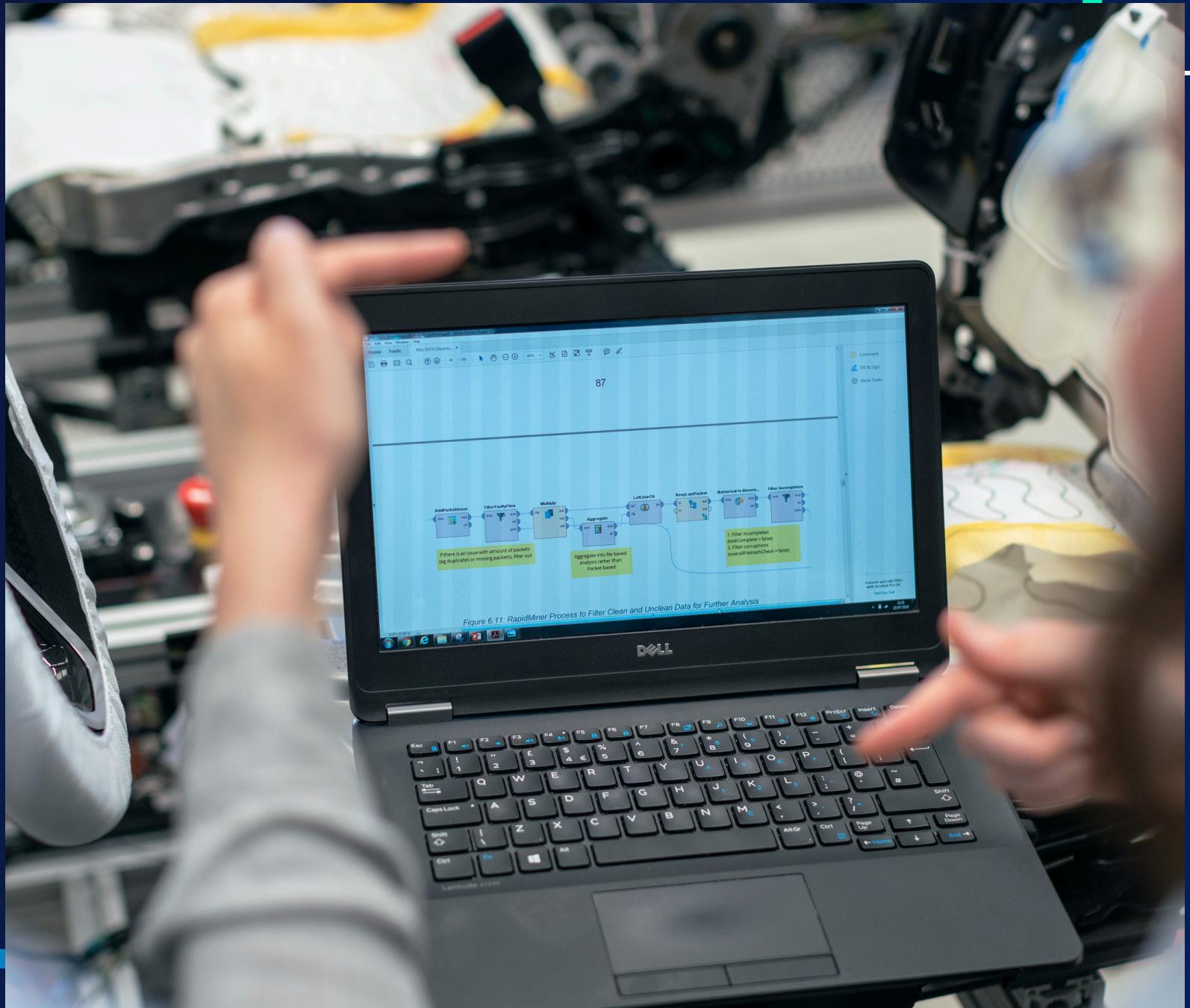
# Tabla de Contenido

- 01 Introducción
- 02 Análisis de requerimientos
- 03 Diagrama de clases
- 04 Diagrama de casos de uso
- 05 Desarrollo del prototipo
- 06 Pruebas y validación
- 07 Conclusión

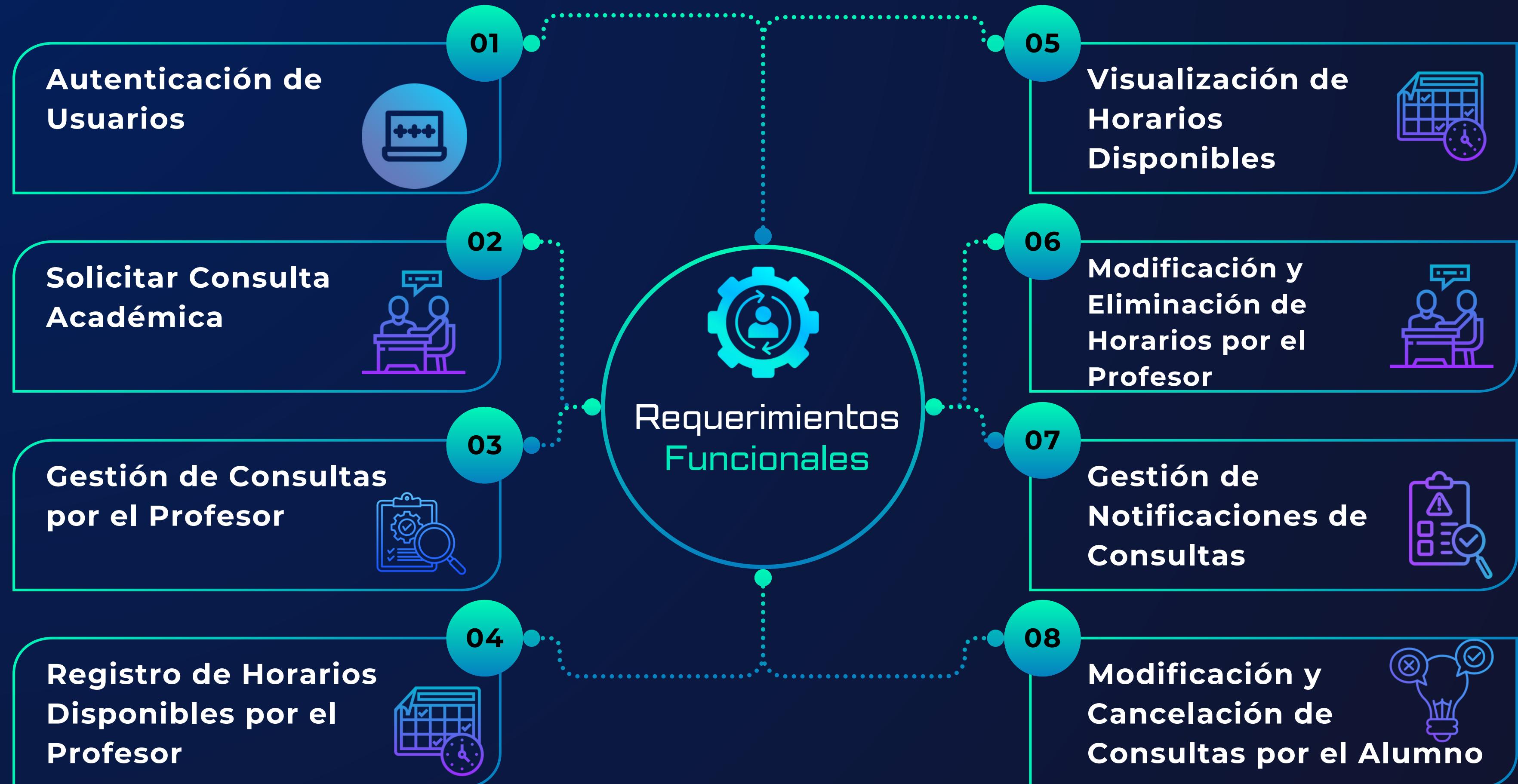


# Introducción

El propósito del presente proyecto de ingeniería de software va enfocado en reconocer, y analizar de manera práctica las fases del ciclo de vida del desarrollo de software.



# Análisis de requerimientos



# Análisis de requerimientos

## No Funcionales

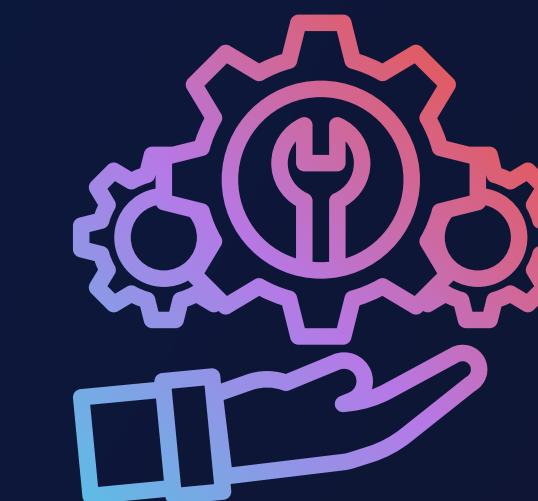
Usabilidad



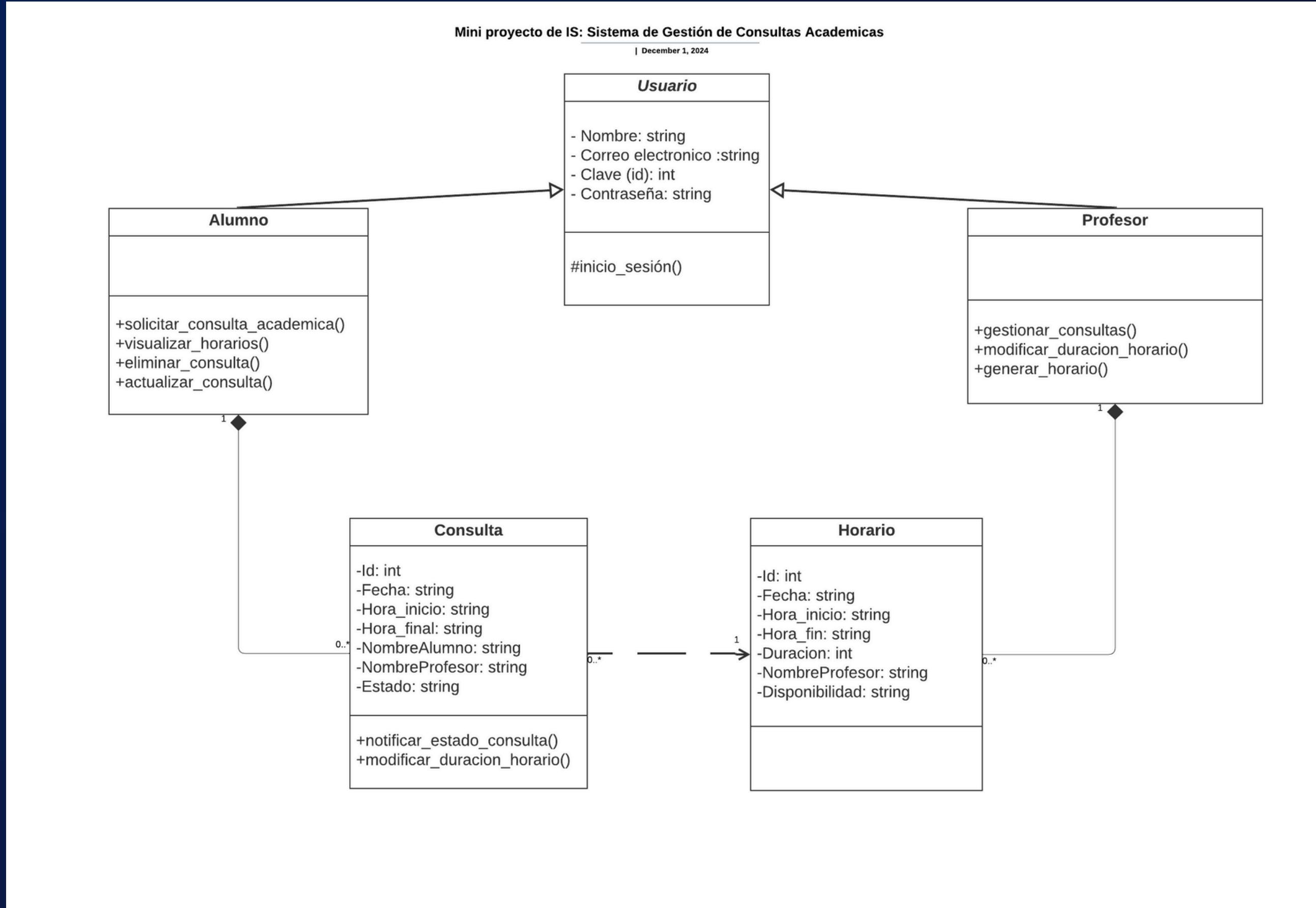
Rendimiento



Tolerancia a  
fallos



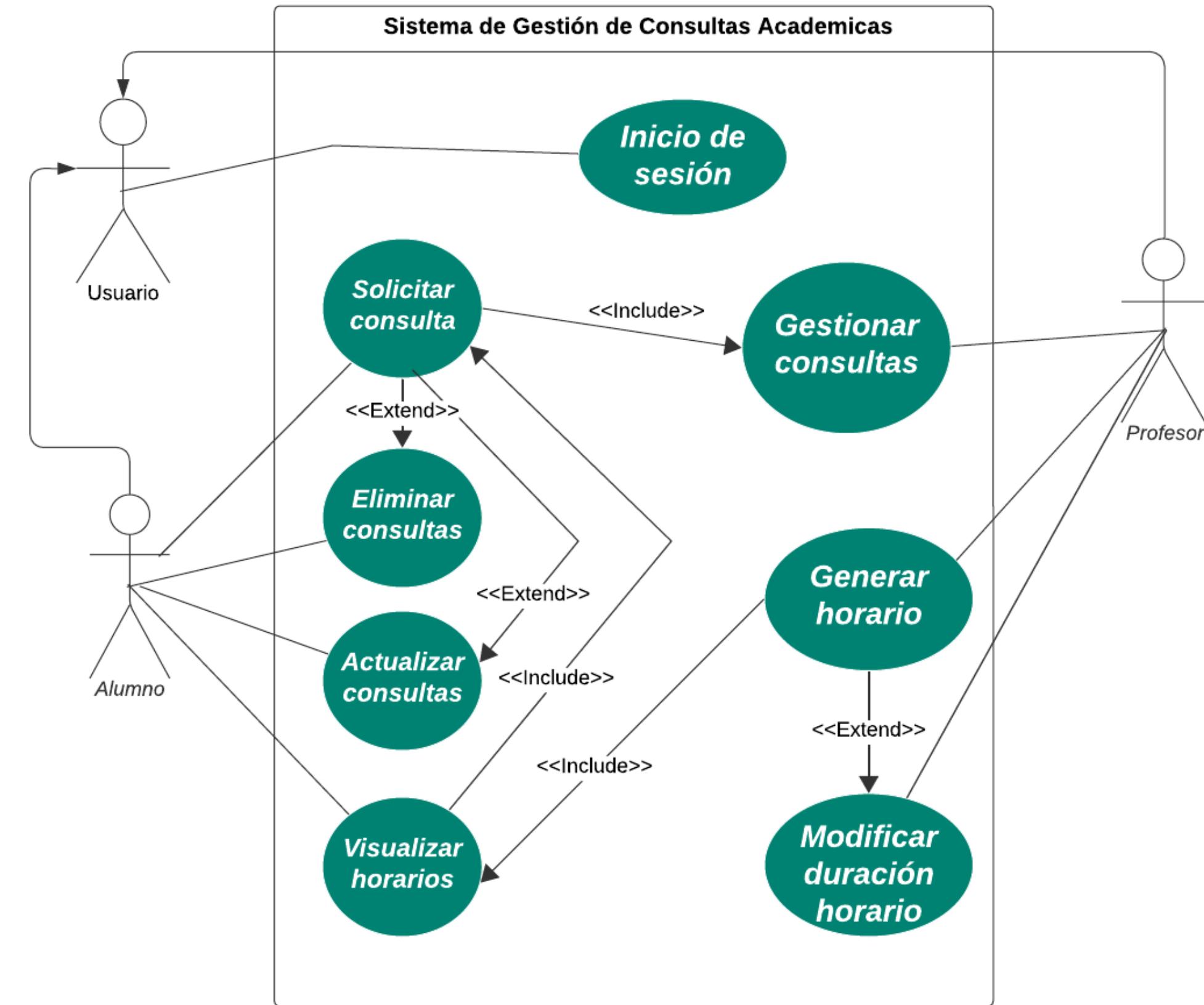
# Diagrama UML: Diagrama de clases



# Diagrama UML: Diagrama de casos de uso

## Mini proyecto de IS: Sistema de Gestión de Consultas Académicas

| December 1, 2024



# Desarrollo de prototipo

```
class Usuario:  
    def __init__(self, nombre, correo, contraseña, clave):  
        self.correo = correo  
        self.contraseña = contraseña  
        self.clave = clave  
  
class Estudiante(Usuario):  
    def __init__(self, nombre, correo, contraseña, clave):  
        super().__init__(nombre, correo, contraseña, clave)  
        self.consultas = []  
  
    def solicitar_consulta(self, consulta):  
        self.consultas.append(consulta)  
  
class Profesor(Usuario):  
    def __init__(self, nombre, correo, contraseña, clave):  
        super().__init__(nombre, correo, contraseña, clave)  
        self.horarios = []  
        self.consultas_gestionadas = []  
  
    def registrar_horario(self, horario):  
        self.horarios.append(horario)  
  
    def gestionar_consulta(self, consulta, nuevo_estado):  
        consulta.estado = nuevo_estado  
  
class Horario:  
    def __init__(self, fecha, hora_inicio, hora_final, duracion, id_horario, profesor):  
        self.fecha = fecha  
        self.hora_inicio = hora_inicio  
        self.hora_final = hora_final  
        self.duracion = duracion  
        self.id_horario = id_horario
```

# Pruebas y validación

Consulta C1 - Estado actual: solicitud

Seleccione el nuevo estado:

Aprobar  
 Rechazar

[Actualizar](#)  
[Volver](#)

 Éxito

 Consulta C1 actualizada a 'aprobar'

[Aceptar](#)

Fecha (YYYY-MM-DD)  
2024-12-02

Hora Inicio (HH:MM)  
10:00

Hora Final (HH:MM)  
11:00

[Registrar](#)  
[Volver](#)

 Éxito

 Horario registrado correctamente

[Aceptar](#)

Correo Electrónico: [profesor1@mail.com](mailto:profesor1@mail.com)

Contraseña: \*\*\*\*\*

[Iniciar Sesión](#)

Seleccione un horario disponible:  
H1: 2024-12-01 de 10:00 a 11:00  
H2: 2024-12-02 de 10:00 a 11:00

 H2

[Solicitar](#)  
[Volver](#)

 Éxito

 Consulta solicitada correctamente

[Aceptar](#)

# Conclusión

Se logró identificar la importancia de analizar y llevar a cabo de manera estructurada el desarrollo de software.

Finalizando con una retroalimentación de varias de las situaciones que deberíamos mejorar.



Muchas  
Gracias