

SCHOOLME

ACADEMICS

ANÁLISIS DE SOFTWARE



Centro de la industria, la empresa y los servicios
ADSO 2900177

SCHOOLME

ACADEMICS

Grupo de Desarrollo SchollMe

Jesus Fernando Carvajal
Santiago Chaparro Riaño

jesusanacona017@gmail.com
alfasan1481@gmail.com

Tabla de contenido

1. Introducción	4
2. Propósito del análisis	4
3. Alcance del sistema	4
4. Perspectiva del producto	5
5. Funcionalidad del sistema.....	5
6. Modelo del sistema	6
6.1 Actores	6
6.2 Casos de uso	6
Diagramas de casos de uso	8
GESTION PERSONAL.....	8
GESTION ACADÉMICA.....	9
GESTIÓN ACADÉMICA.....	10
GESTION DE PERFIL Y SISTEMA	10
GESTION DE AGENDA.....	11
DOCENTE	12
Diagramas de clase	13
6.3 Flujos funcionales	13
ADMIN-ACCESO Y GESTION PERSONAL	15
MODULO PERFIL	16
PERFIL PERMISO	17
CIERRE DE PERIODO	18
ASIGNACIÓN DIRECCIÓN DE GRUPO	19
GRADOS	20
MATERIAS	21
DOCENTE	22
DOCENTE-MODULOS	23
DOCENTE-DATOS PERSONALES	24
DOCENTE-AGENDA.....	25

DOCENTE-REGISTRO DE NOTAS	26
DOCENTE-CARGA ACADÉMICA	27
DOCENTE-REPORTES	28
ESTUDIANTE	29
ESTUDIANTE-MÓDULOS	30
ESTUDIANTE-DATOS PERSONALES	31
6.4 Reglas del negocio	31
7. Arquitectura del sistema	32
7.1 Arquitectura general	32
7.2 Arquitectura web (Angular)	32
7.3 Arquitectura móvil (React Native)	32
7.4 Arquitectura API (ASP.NET Core)	33
7.5 Seguridad	33
10. Análisis de viabilidad	35
11. Conclusiones	36
12. Referencias	36

1. Introducción

Este documento presenta el análisis técnico y funcional del software SchoolMe, una plataforma web y móvil orientada a la digitalización de procesos académicos y administrativos en instituciones educativas. El análisis comprende la definición de actores, casos de uso, estructura funcional, arquitectura tecnológica, flujos del sistema, modelo de datos y evaluación de riesgos y viabilidad.

Este documento se complementa con la Especificación de Requerimientos (Documento 01), pero constituye un archivo independiente, centrado exclusivamente en el análisis técnico y el modelado del software.

2. Propósito del análisis

El propósito de este documento es describir los elementos estructurales, funcionales y arquitectónicos del sistema SchoolMe, con el fin de:

- Comprender la lógica interna del sistema.

- Analizar componentes, módulos y relaciones.

- Establecer bases para el diseño e implementación.

- Formalizar los procesos que operan en la versión web y móvil.

- Asegurar coherencia entre requisitos, arquitectura y modelado.

3. Alcance del sistema

El análisis abarca la plataforma SchoolMe en su fase actual, que incluye:

- Gestión institucional de usuarios y roles.

- Parametrización académica.

Asignación de carga docente.

Creación y administración de agendas escolares.

Registro diario por parte de docentes y directores de grupo.

Consulta y confirmación de agendas por parte de acudientes.

Arquitectura web, móvil y API.

Seguridad de acceso basada en roles.

El análisis no cubre procesos contables, pagos, matrículas o integraciones externas.

4. Perspectiva del producto

SchoolMe es un sistema modular compuesto por:

Aplicación web: para administradores, docentes y directores de grupo.

Aplicación móvil: para acudientes.

API REST: núcleo de comunicación entre las aplicaciones y la base de datos.

Base de datos relacional: repositorio de información académica y administrativa.

El sistema está diseñado para ser personalizable y escalable, con posibilidad de ampliación futura sin afectar la lógica base.

5. Funcionalidad del sistema

El sistema proporciona funcionalidades principales:

Gestión de usuarios con roles y permisos.

Parametrización escolar: grados, grupos, materias, horarios, periodos.

Asignación de carga académica docente.

Creación y mantenimiento de preguntas para la agenda escolar.

Registro de agenda diaria: global e individual.

Visualización y confirmación de agenda por acudientes.

Seguridad y auditoría básica.

6. Modelo del sistema

6.1 Actores

Actor	Descripción
Administrador	Gestiona usuarios, parámetros y configuración académica.
Docente	Registra observaciones y consulta carga académica.
Director de grupo	Crea y cierra agenda diaria; consolida información.
Acudiente	Visualiza agenda diaria y confirma lectura.
Estudiante	Usuario referencial sin acciones.

6.2 Casos de uso

Se describen los casos de uso principales del sistema:

Caso de uso 1: Gestionar personas

El administrador puede crear, editar, eliminar y consultar usuarios del sistema. Cada usuario recibe un rol que define su acceso.

Caso de uso 2: Configurar parámetros académicos

Se parametrizan grados, grupos, periodos, horarios y materias.

Caso de uso 3: Asignar carga académica

El administrador relaciona docente–materia–grupo–horario.

Caso de uso 4: Crear agenda académica

El administrador crea preguntas globales e individuales y las asigna a un grado.

Caso de uso 5: Registrar agenda diaria

El director ingresa la información global e individual de los estudiantes.

Caso de uso 6: Registrar observaciones

El docente ingresa observaciones generales relacionadas con su clase diaria.

Caso de uso 7: Cerrar agenda

El director cierra la agenda del día, inmovilizándola para edición.

Caso de uso 8: Visualizar agenda

El acudiente consulta la agenda de su estudiante.

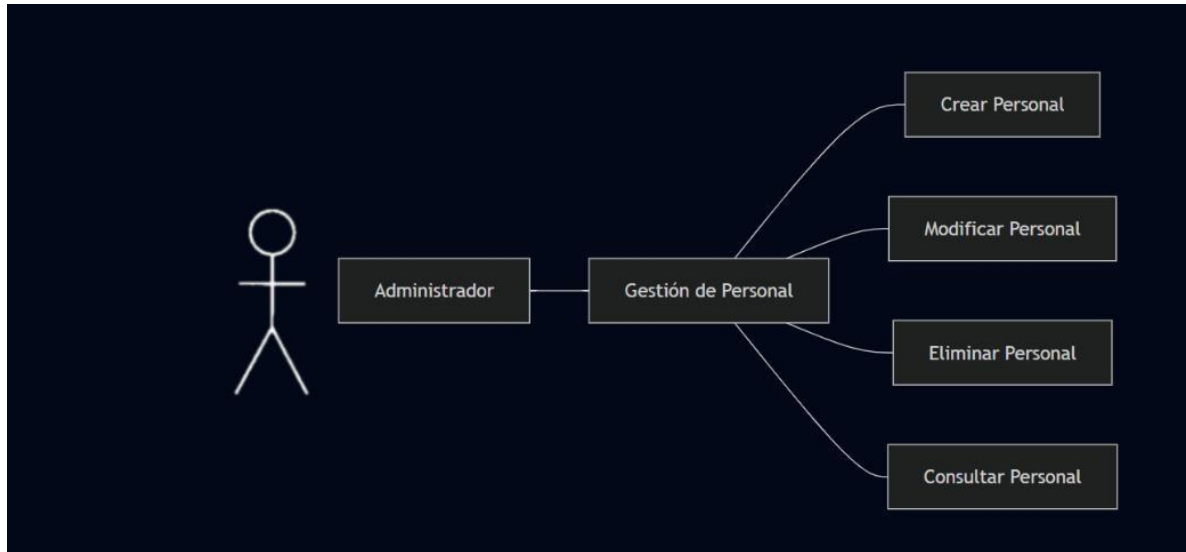
Caso de uso 9: Confirmar agenda

El acudiente registra su confirmación de lectura.

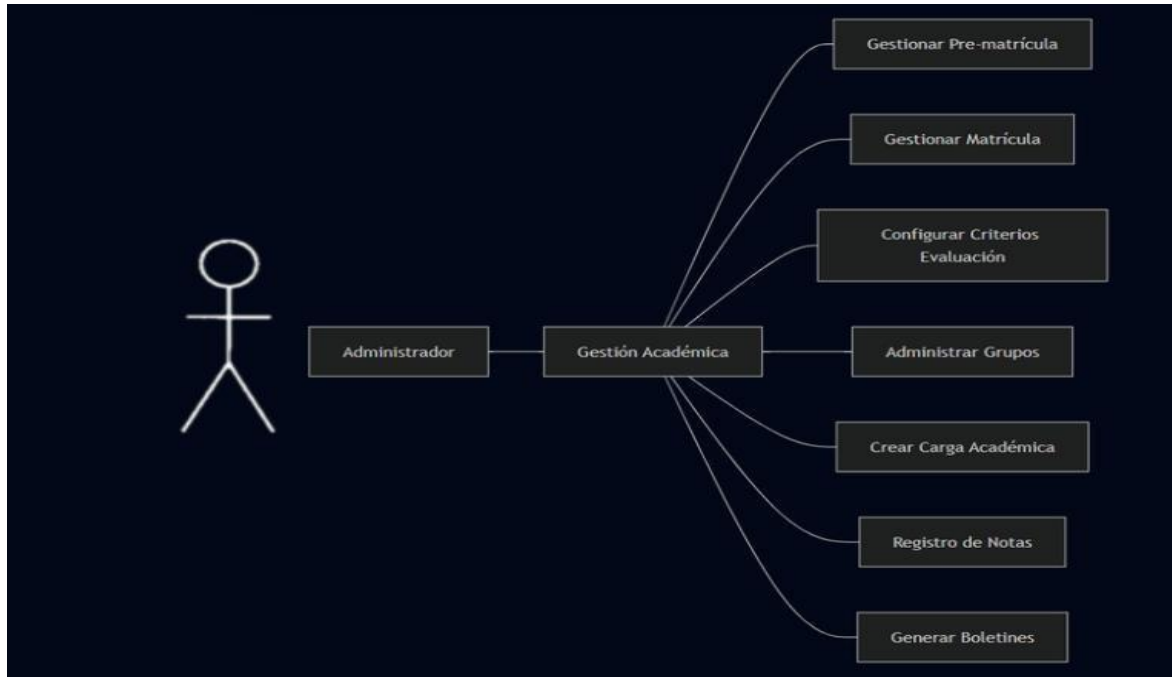
Diagramas de casos de uso

DIAGRAMAS CASOS DE USO ADMINISTRADOR

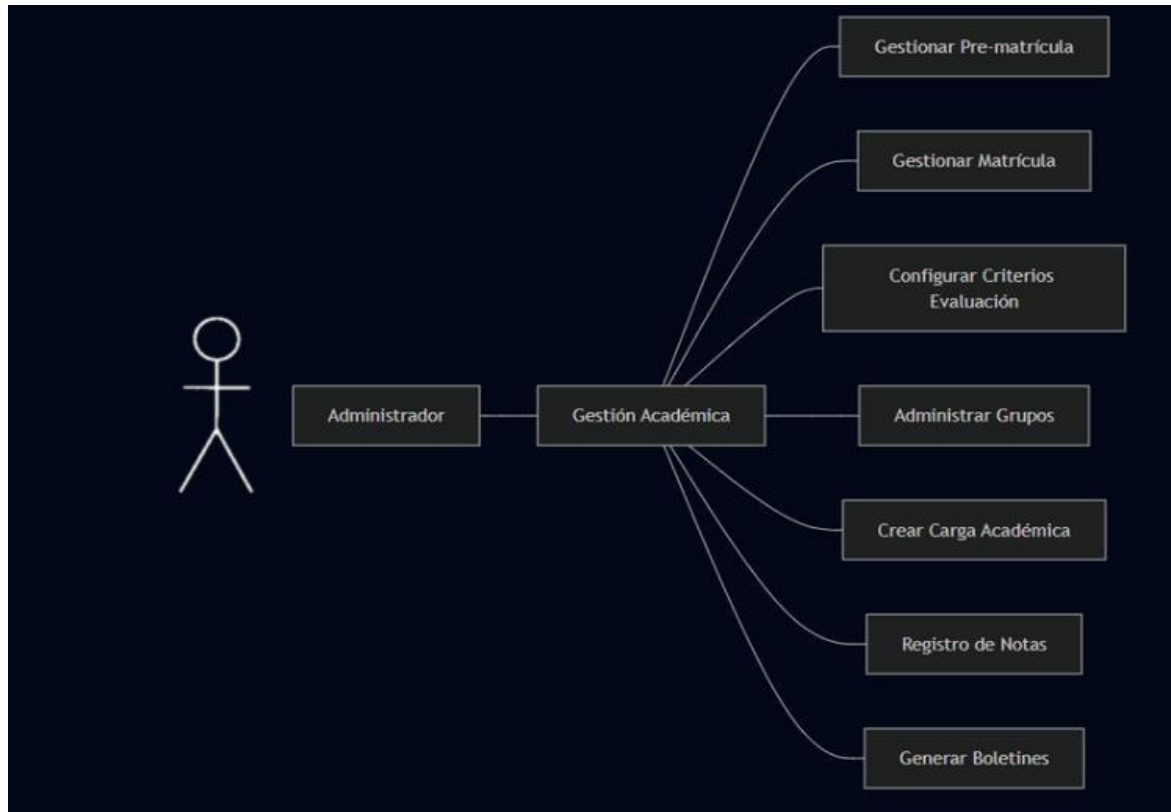
GESTION PERSONAL



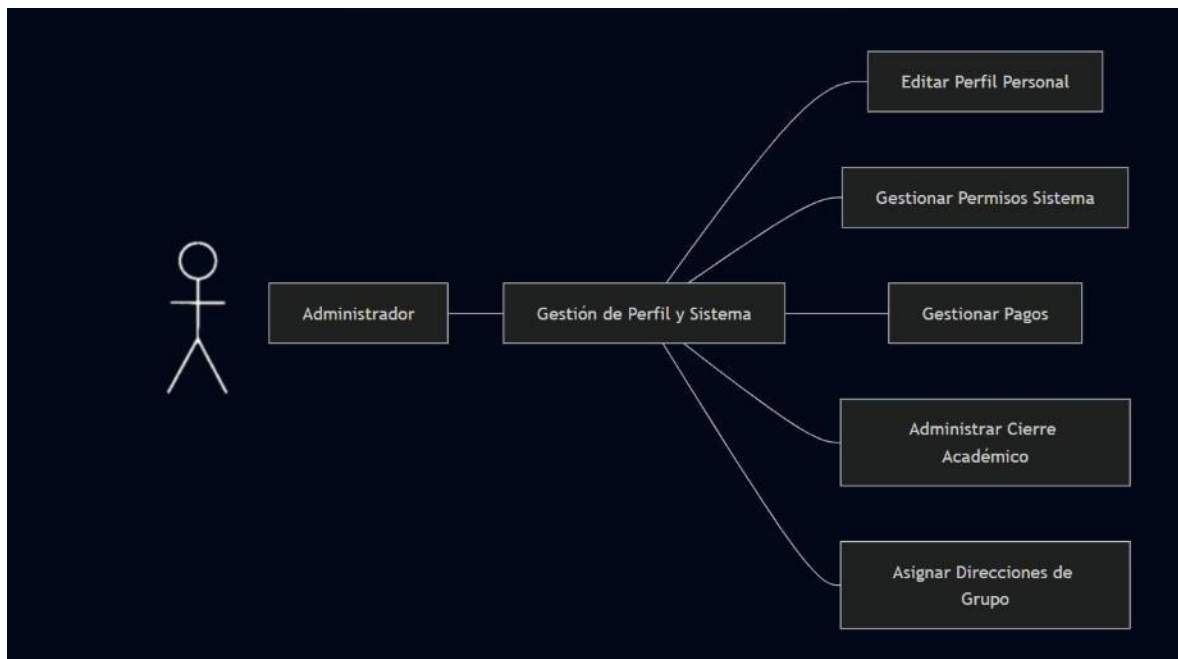
GESTIONACADÉMICA



GESTIÓN ACADÉMICA



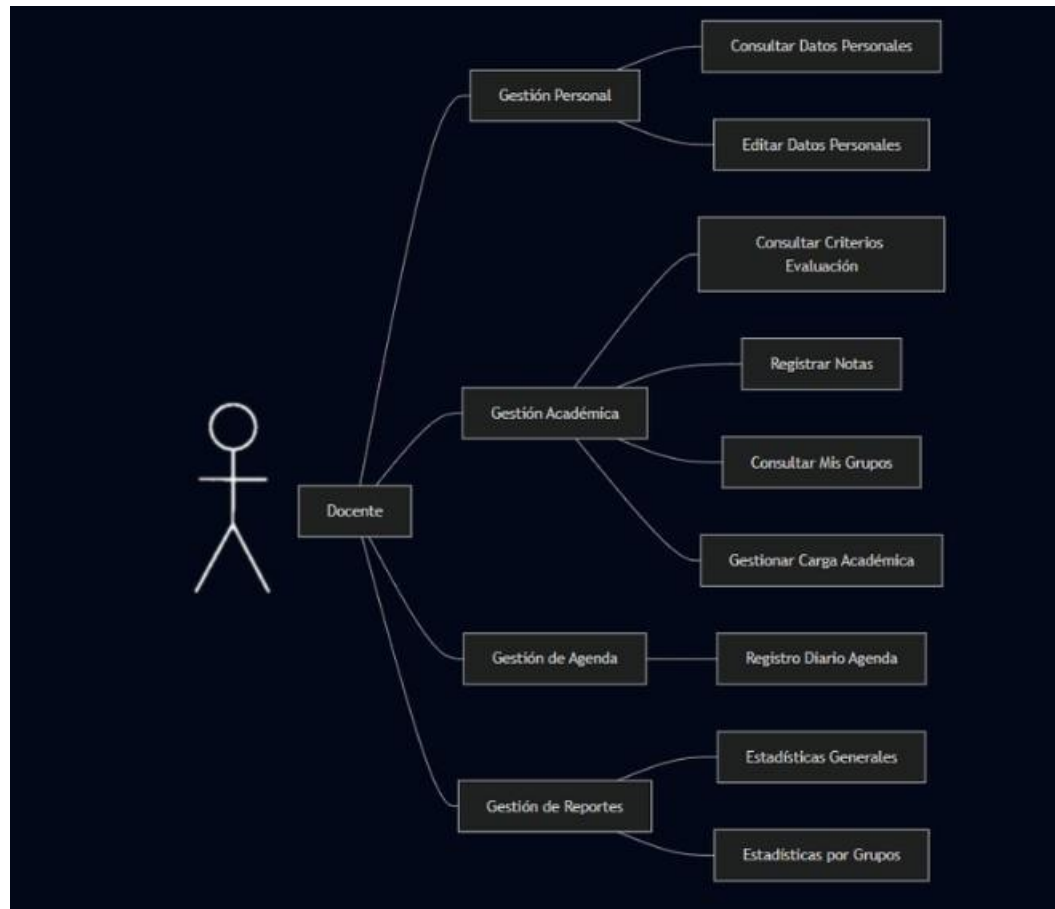
GESTIÓN DE PERFIL Y SISTEMA



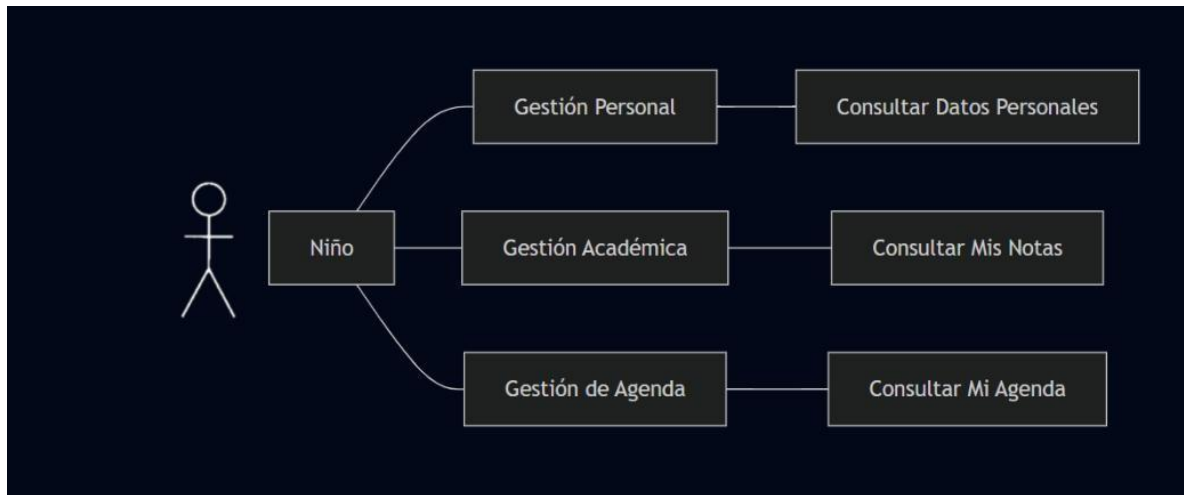
GESTION DE AGENDA



DOCENTE



ESTUDIANTE



Diagramas de clase

Se adjunta el material de los diagramas, puesto que son un poco extensos y para ello se dará esta opción para una mejor experiencia de visualización

<https://drive.google.com/file/d/1CUBnsEfnCZSJx7fqGVbPspqhDpTOJy90/view?usp=s>
haring

<https://drive.google.com/file/d/1zmETUD1GxmW9YQEskeTnaNOcXqwZpA8P/view?usp=sharing>
<https://drive.google.com/file/d/1T-IM3oKXFd24ngjMrvkMLToZyNf9P0ie/view?usp=sharing>

6.3 Flujos funcionales

El administrador define usuarios y parámetros.

La plataforma genera carga docente por grupo y materia.

El director genera y abre agenda diaria.

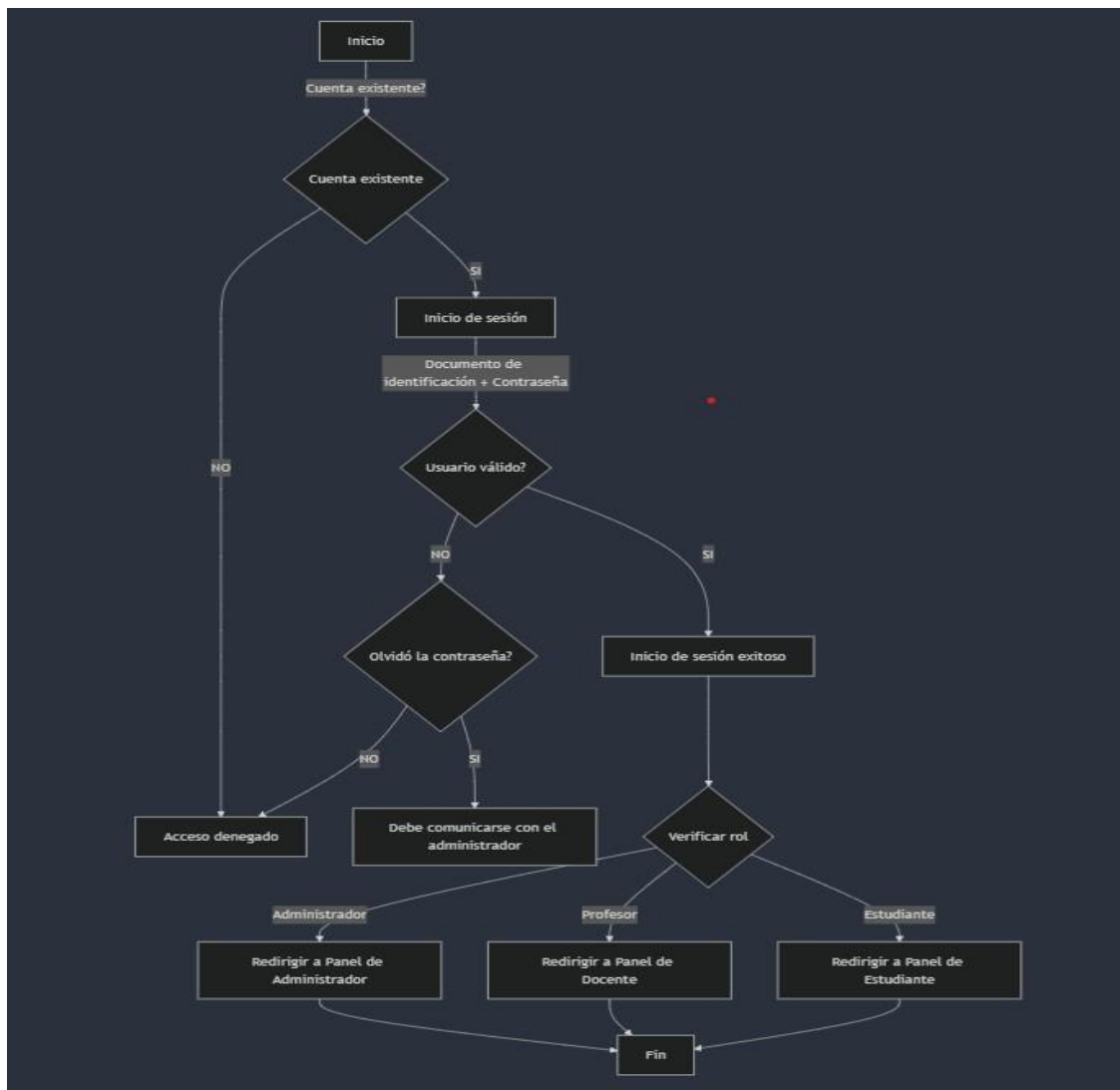
Los docentes registran observaciones individuales y globales.

El director consolida la información y cierra la agenda.

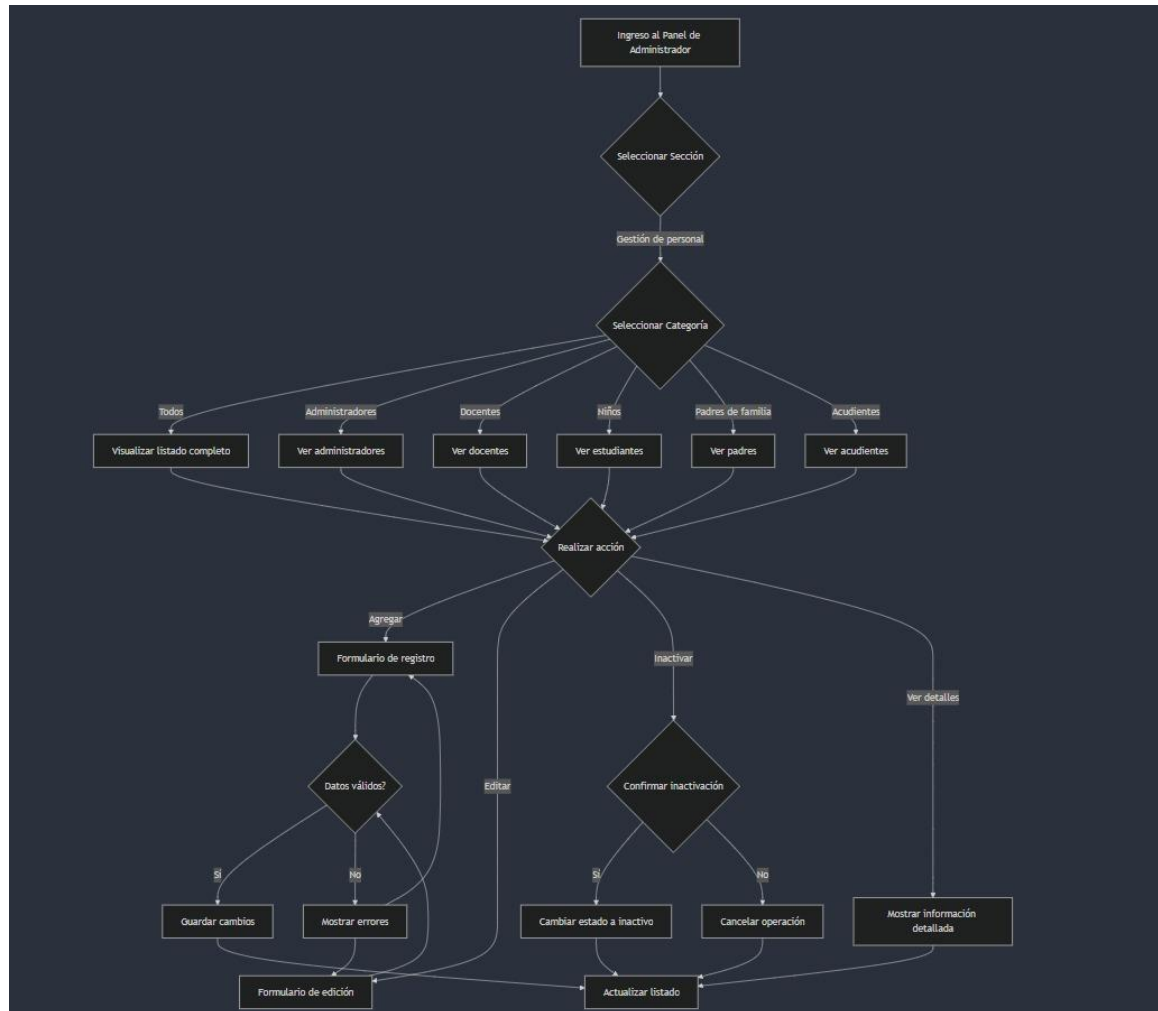
Los acudientes reciben la agenda, la leen y confirman.

DIAGRAMAS DE FLUJO

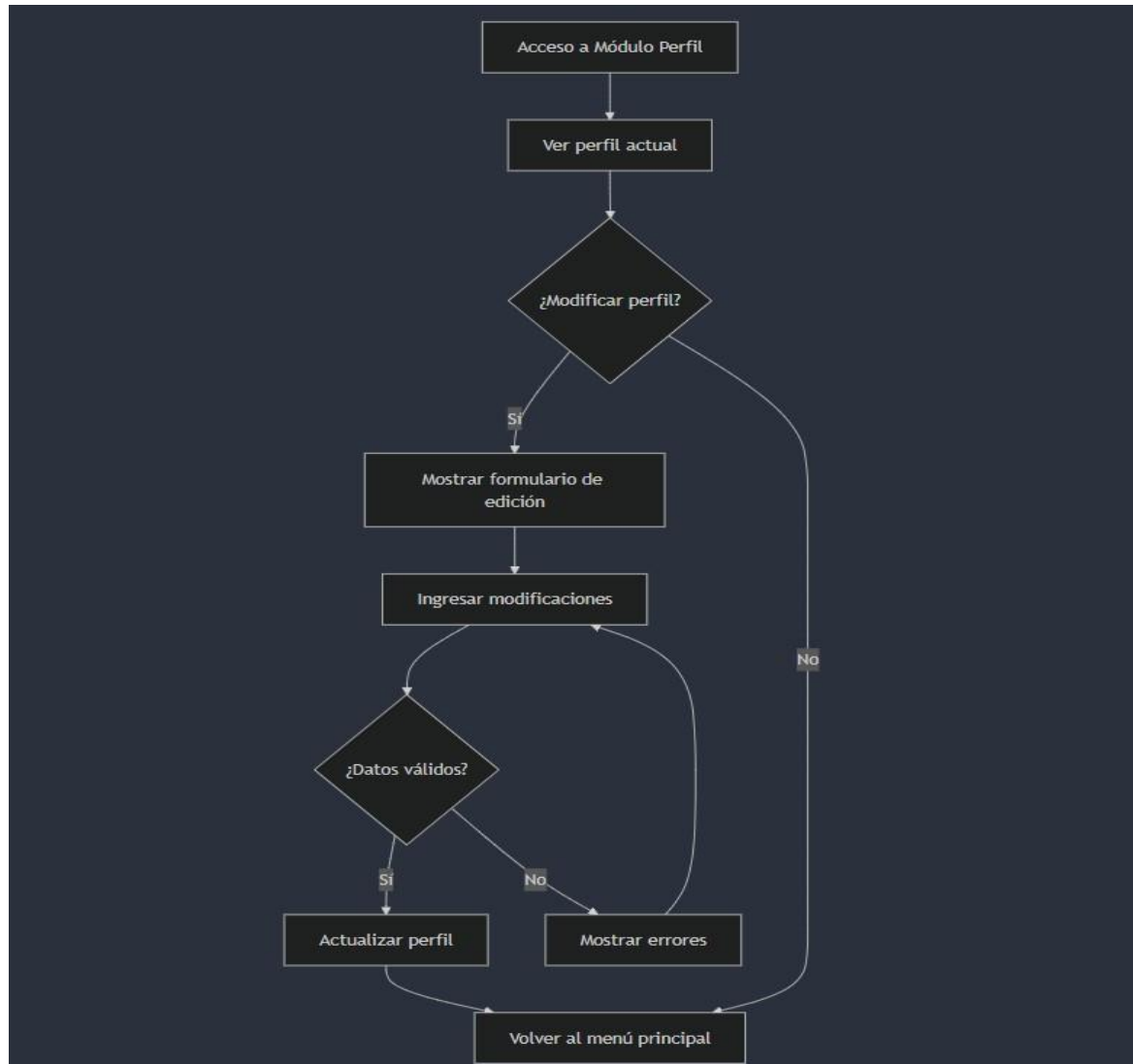
LOGIN



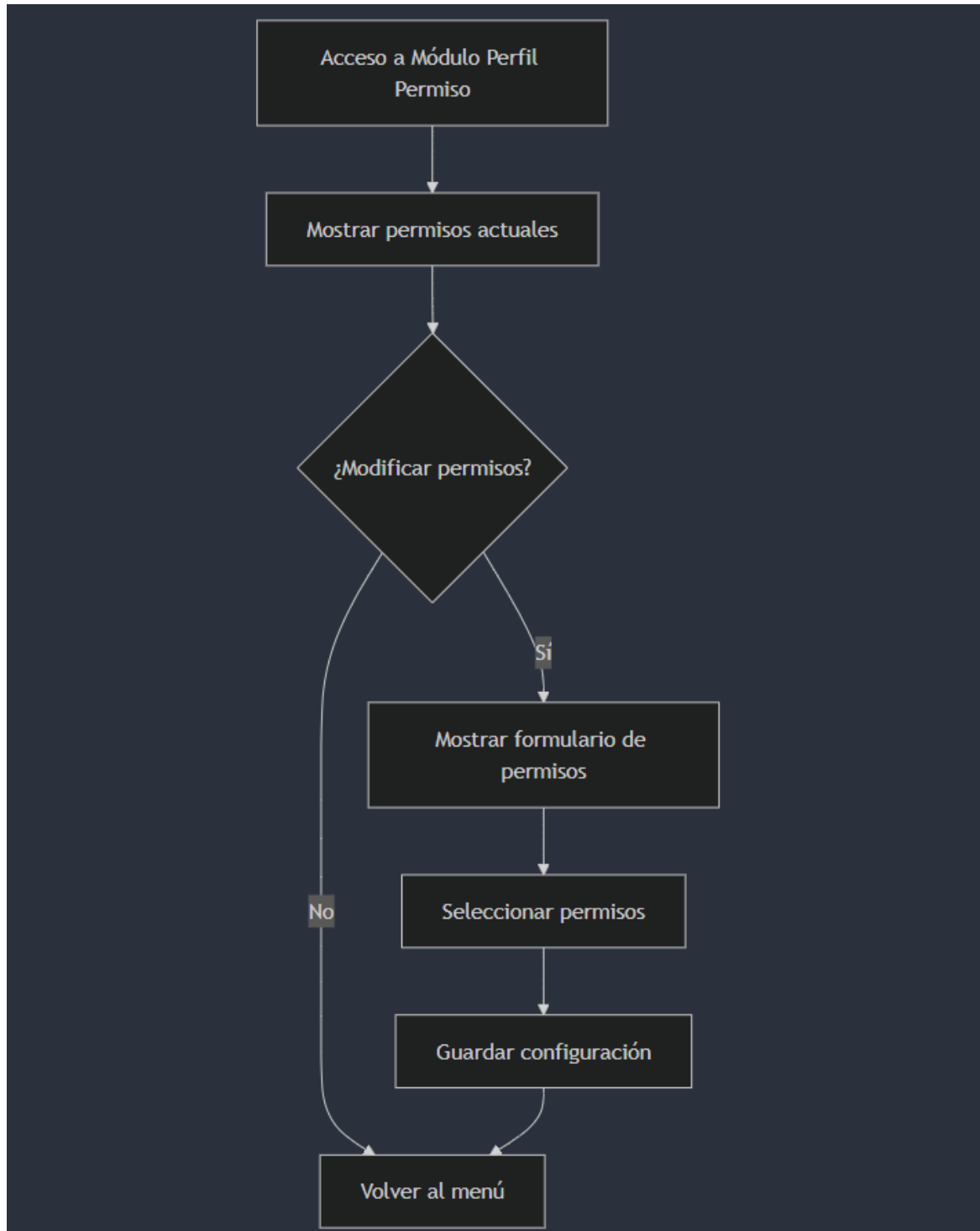
ADMIN-ACCESO Y GESTION PERSONAL



MODULO PERFIL



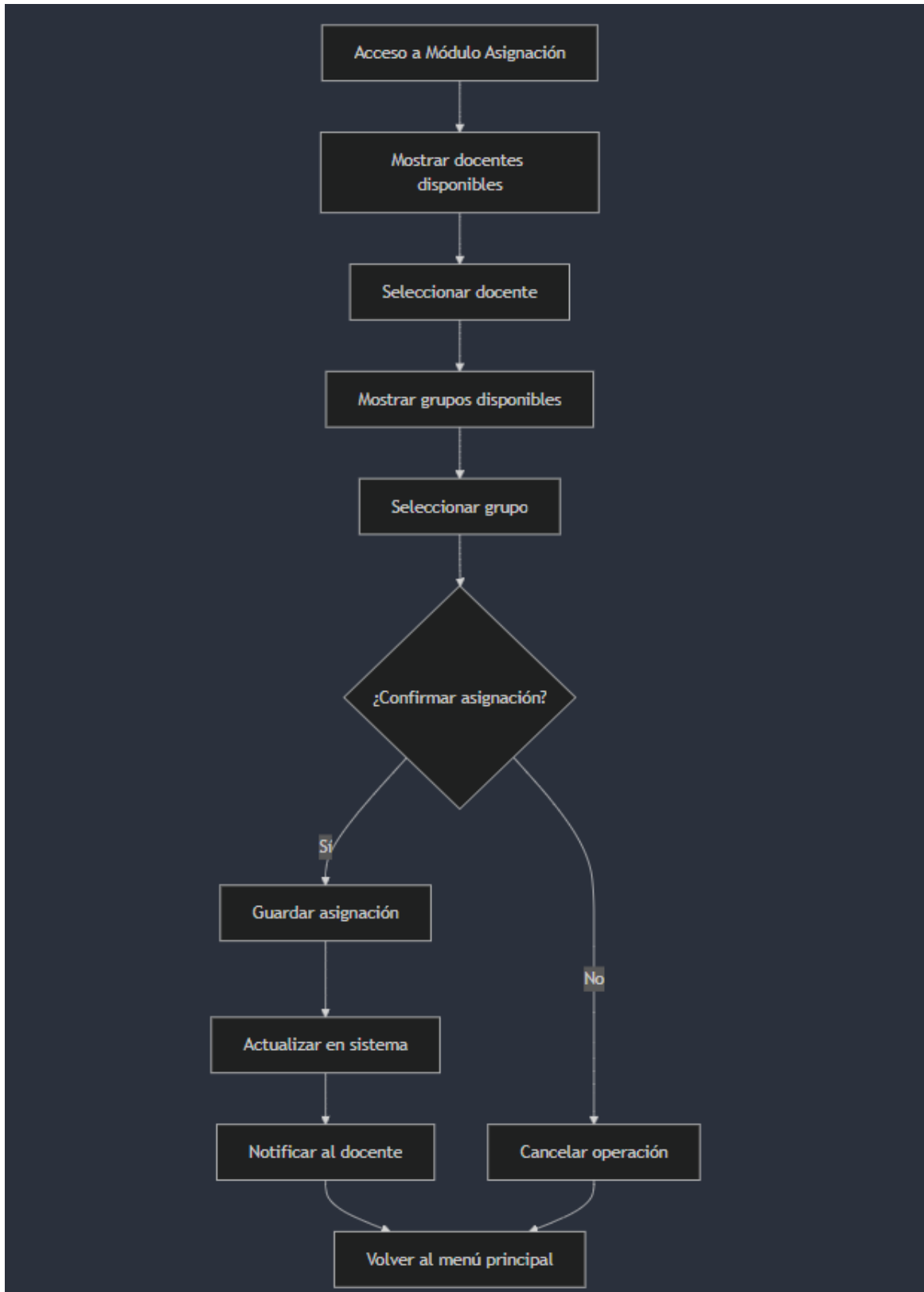
PERFIL PERMISO



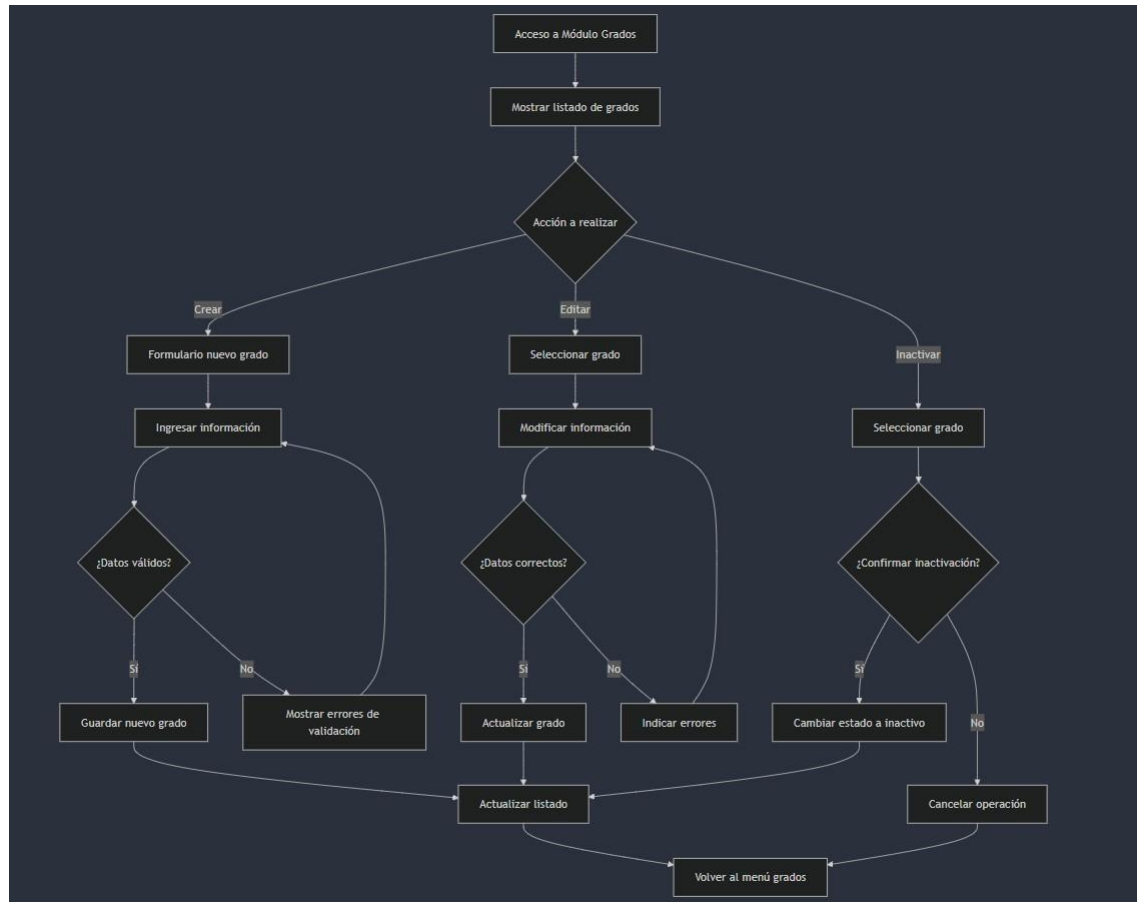
CIERRE DE PERIODO



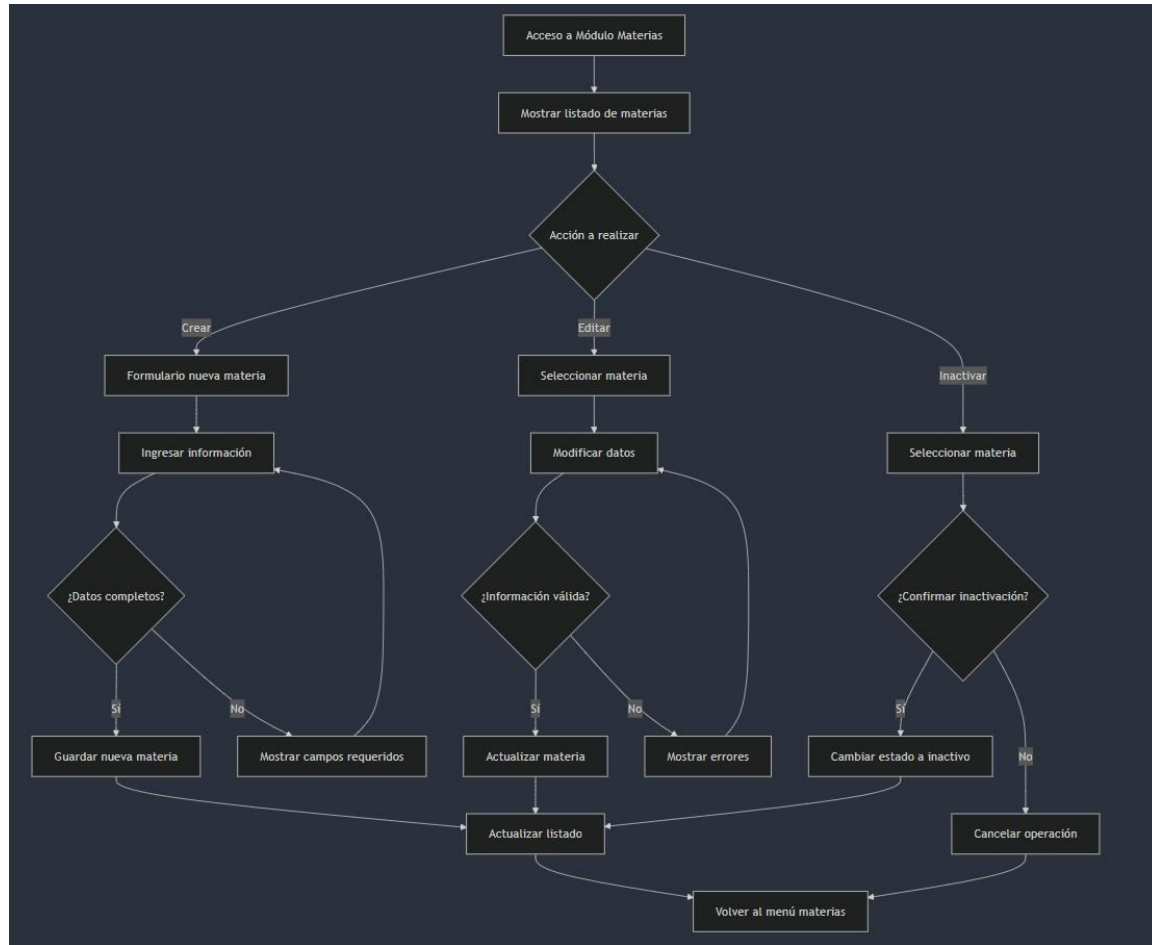
ASIGNACIÓN DIRECCIÓN DE GRUPO



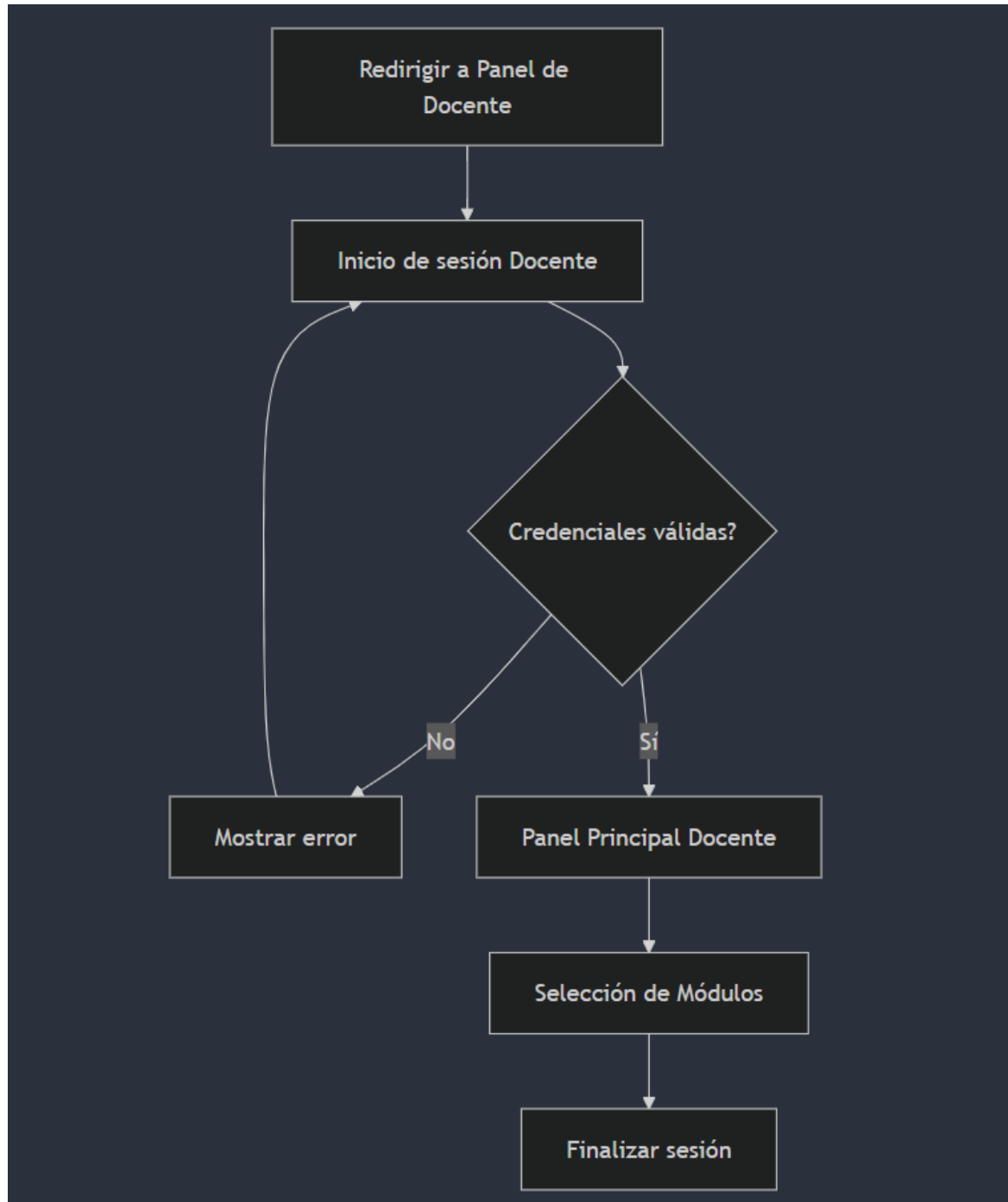
GRADOS



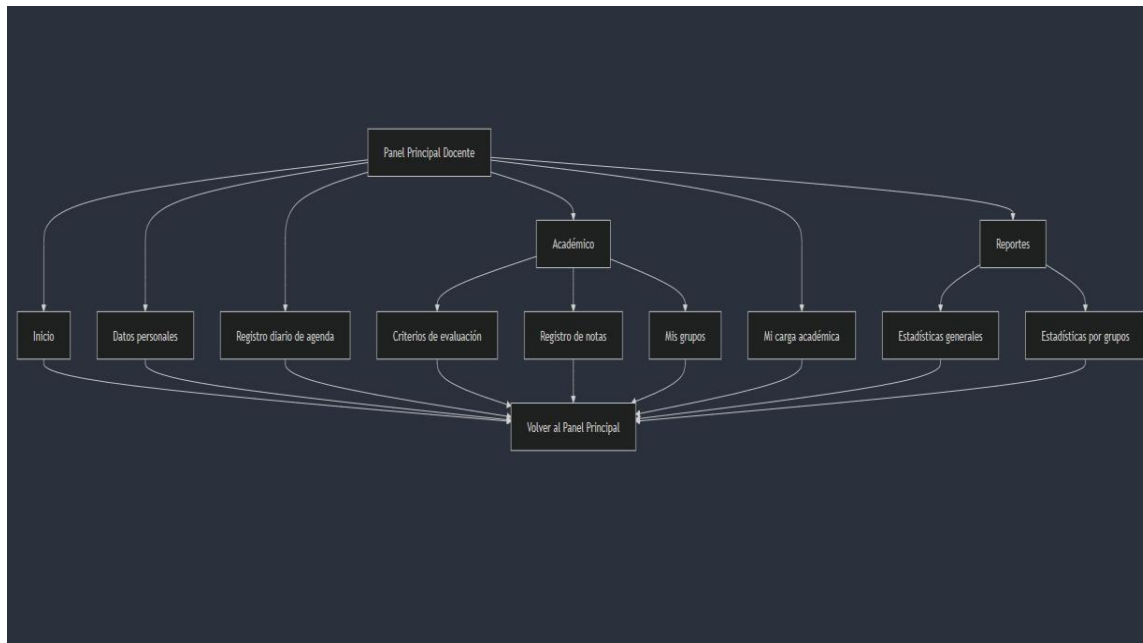
MATERIAS



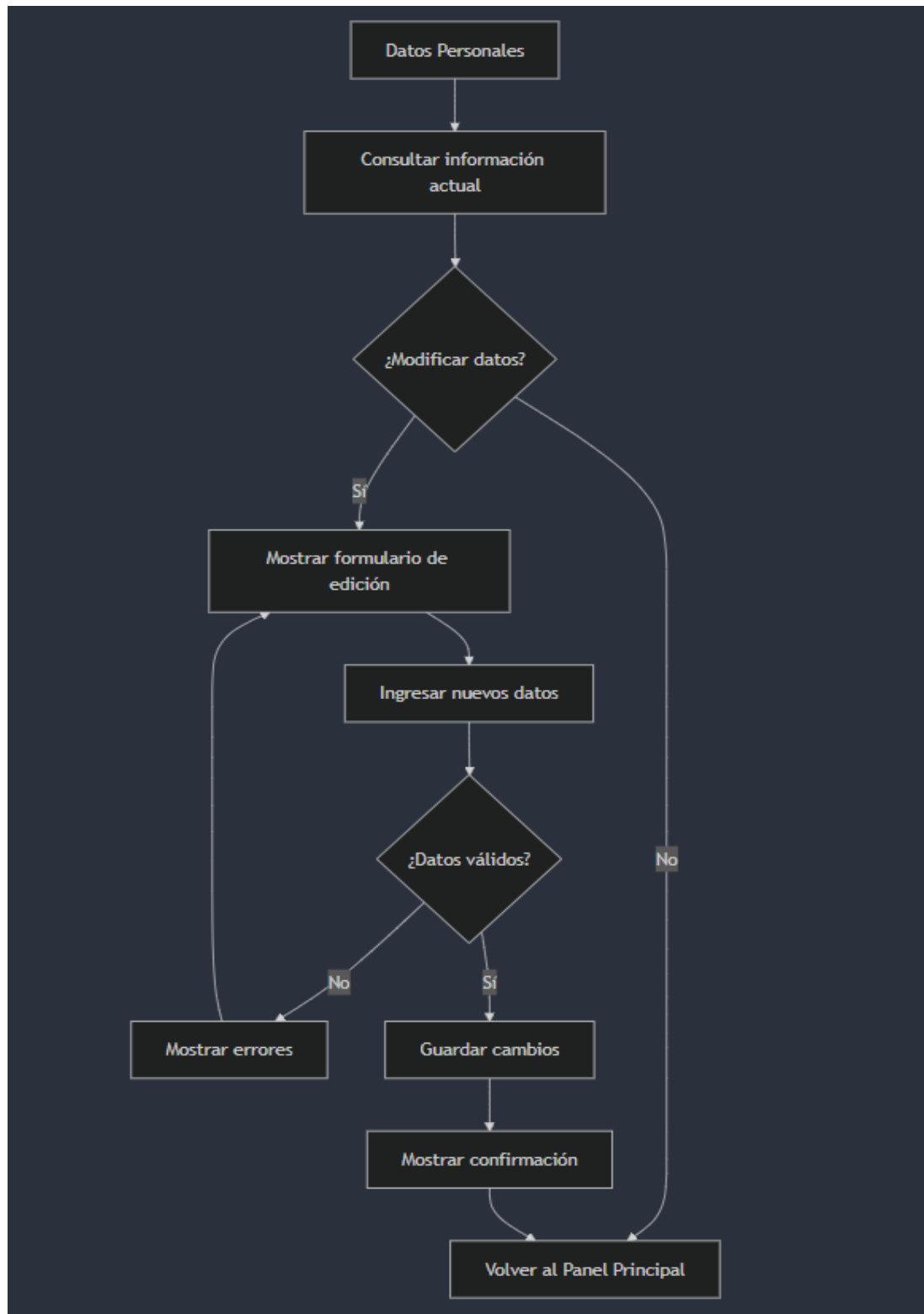
DOCENTE



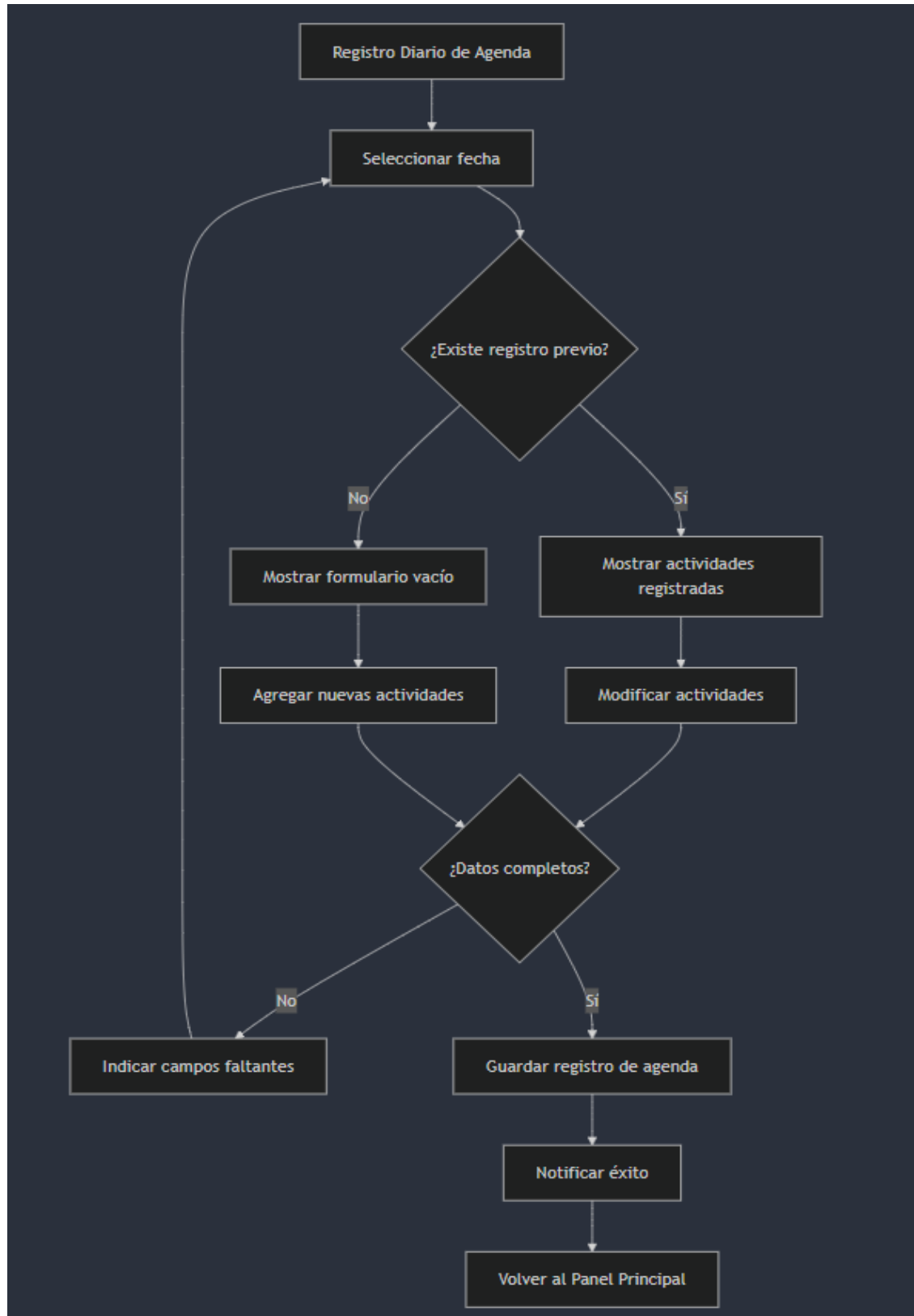
DOCENTE-MODULOS



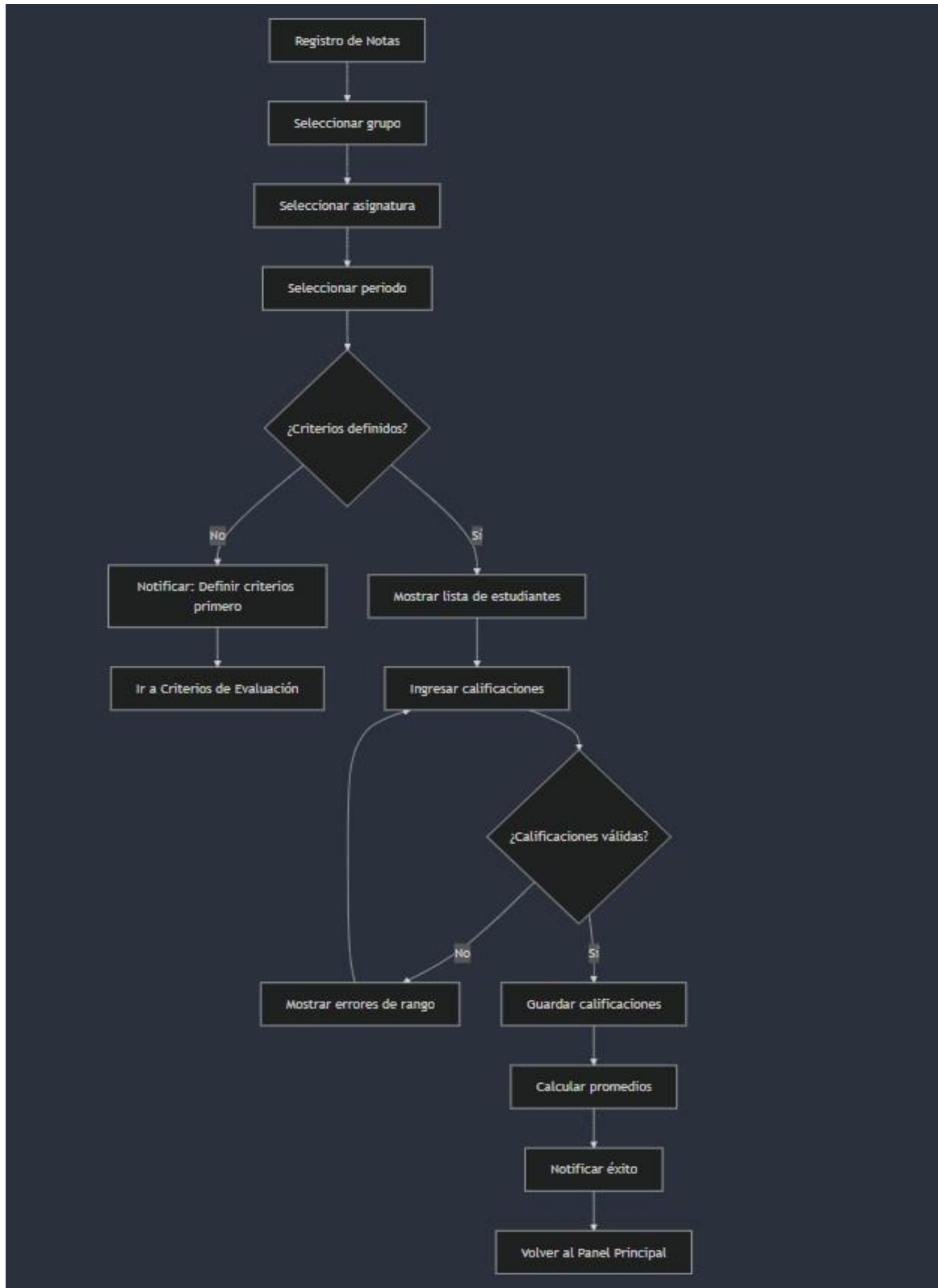
DOCENTE-DATOS PERSONALES



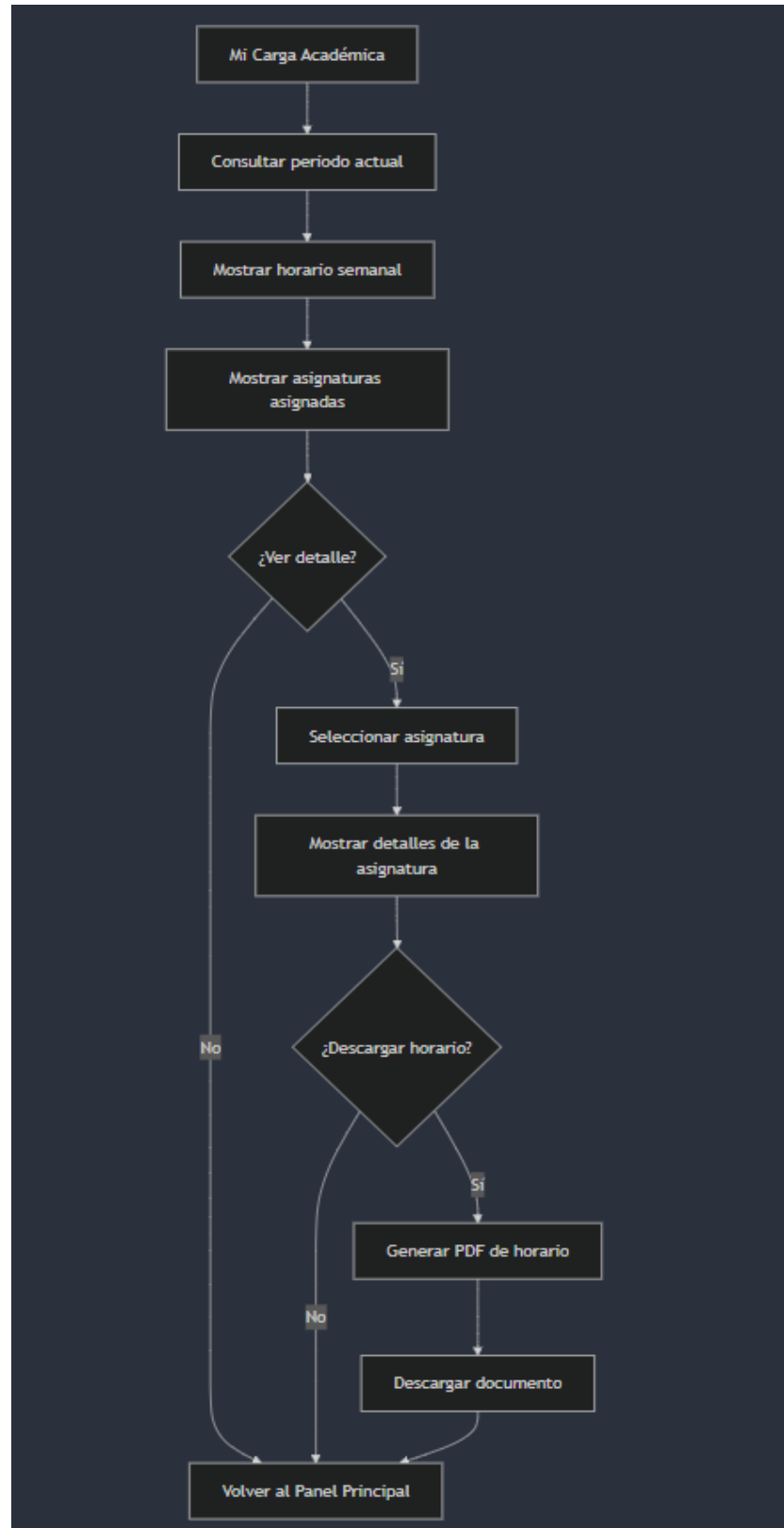
DOCENTE-AGENDA



DOCENTE-REGISTRO DE NOTAS



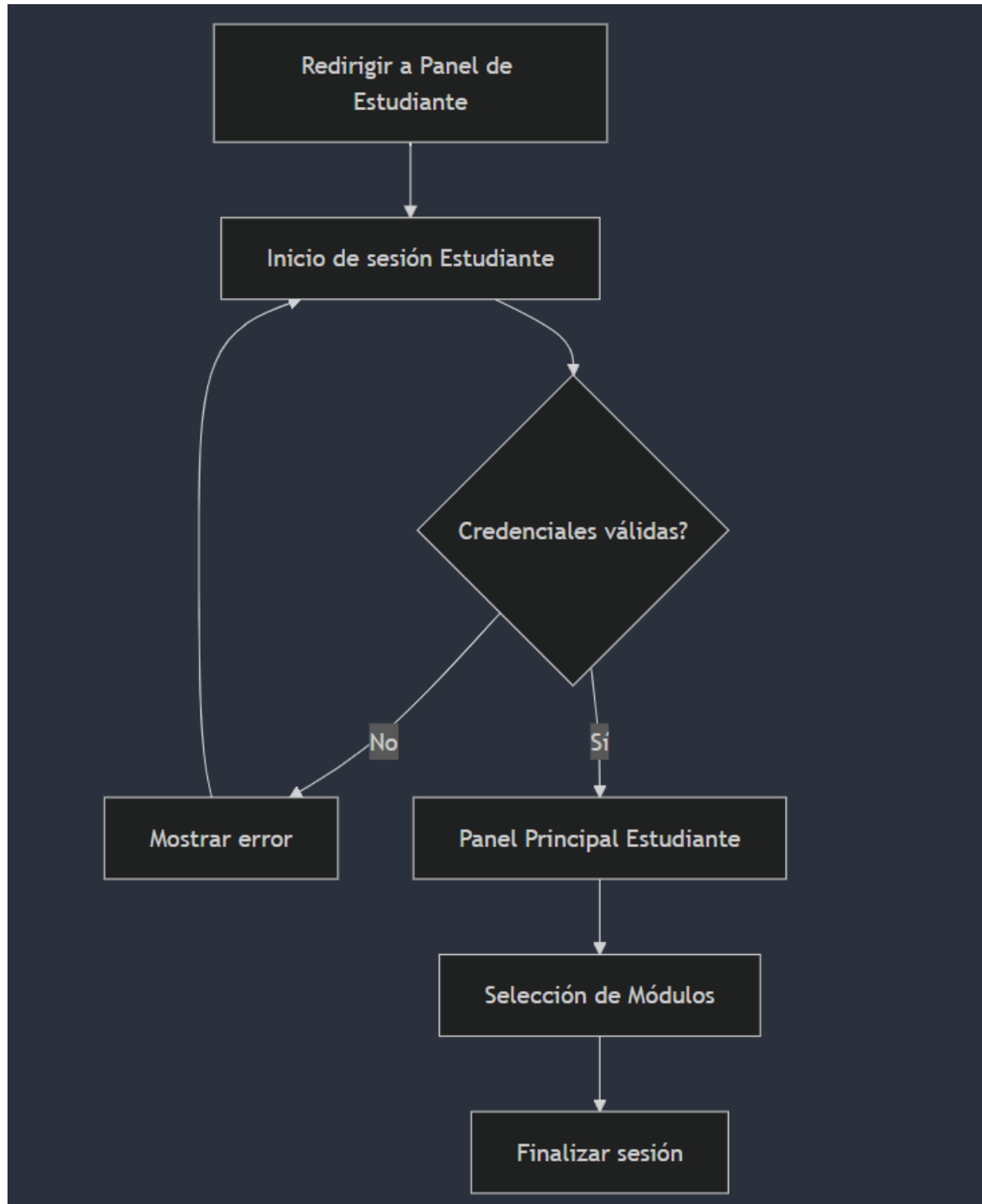
DOCENTE-CARGA ACADÉMICA



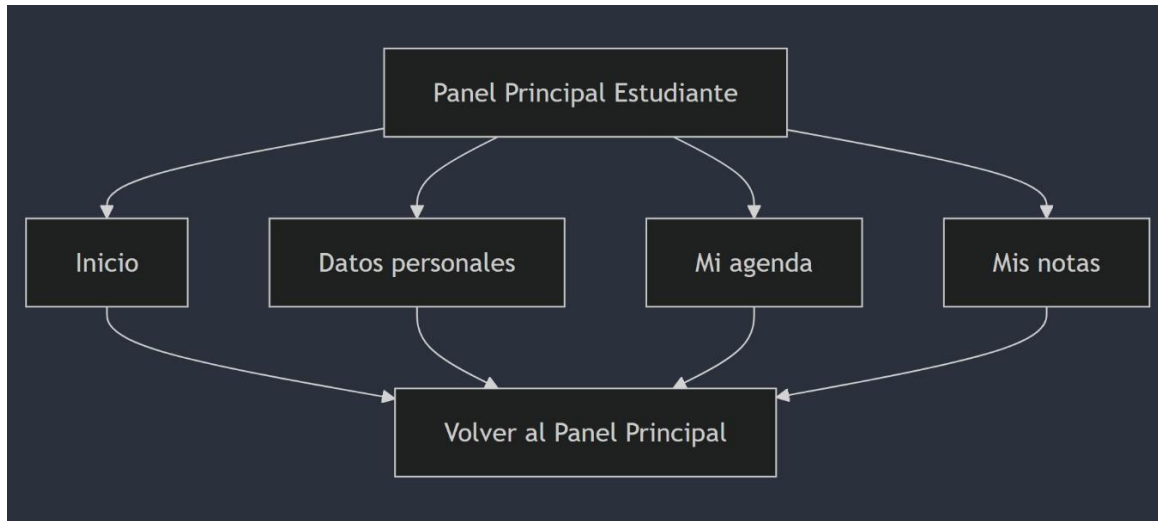
DOCENTE-REPORTES



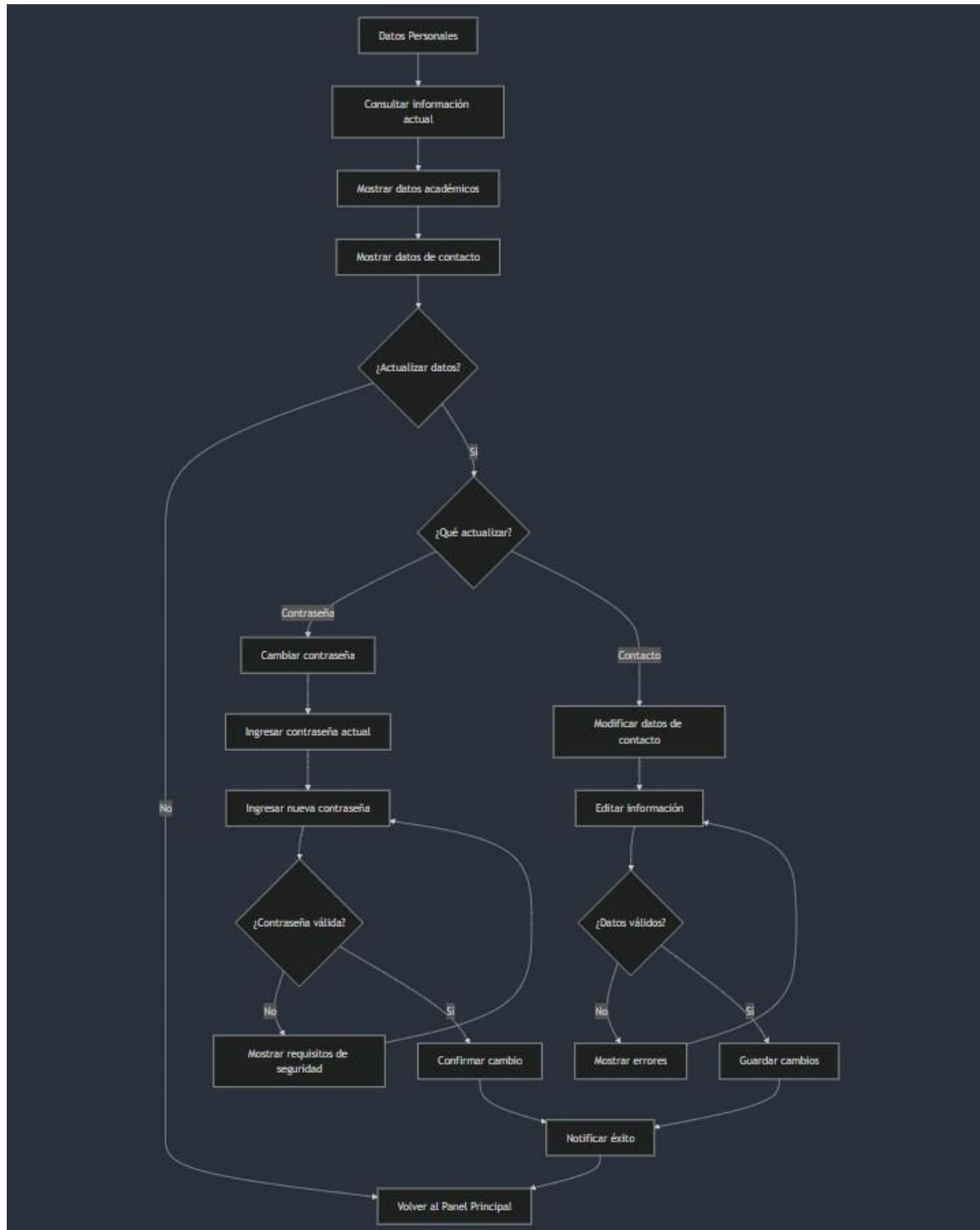
ESTUDIANTE



ESTUDIANTE-MODULOS



ESTUDIANTE-DATOS PERSONALES



6.4 Reglas del negocio

Un acudiente solo puede confirmar una agenda una vez al día.

Un docente no puede registrar observaciones si la agenda está cerrada.

Un grupo solo puede tener un director asignado.

El sistema debe impedir solapamientos de horarios.

La agenda de un día no puede reabrirse.

7. Arquitectura del sistema

7.1 Arquitectura general

SchoolMe adopta una arquitectura en capas:

Capa de presentación (web y móvil)

Capa API (REST)

Capa de lógica de negocio

Capa de datos

7.2 Arquitectura web (Angular)

SPA modular

Servicios HTTP inyectables

Componentes por módulo

Routing jerárquico

Material Design

7.3 Arquitectura móvil (React Native)

Pantallas por flujo funcional

Fetch para consumo API

Compatible con Android

7.4 Arquitectura API (ASP.NET Core)

Controladores REST

Patrón DTO

Middleware de autenticación

Validación de modelos

Servicios y repositorios

Dapper o EF Core según configuración

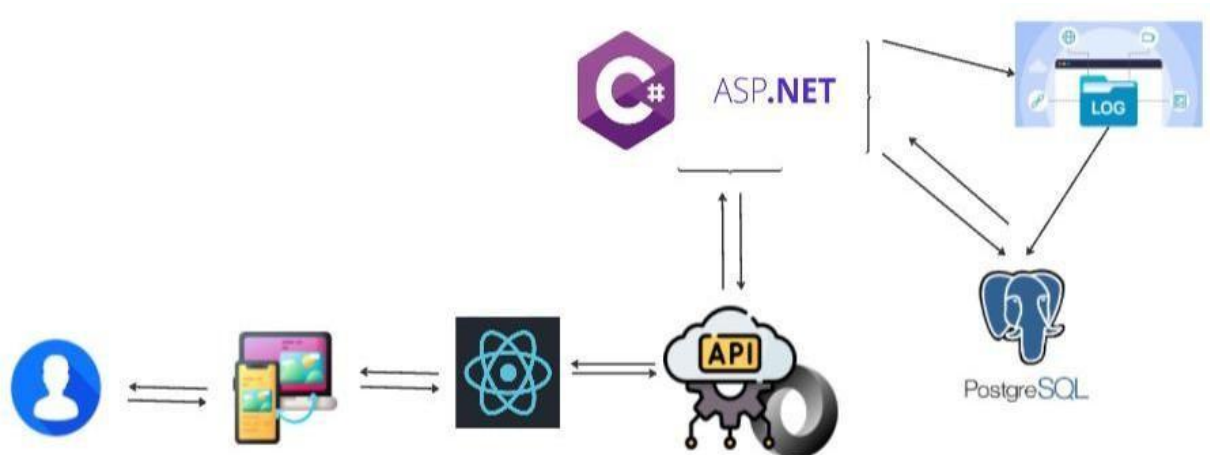
7.5 Seguridad

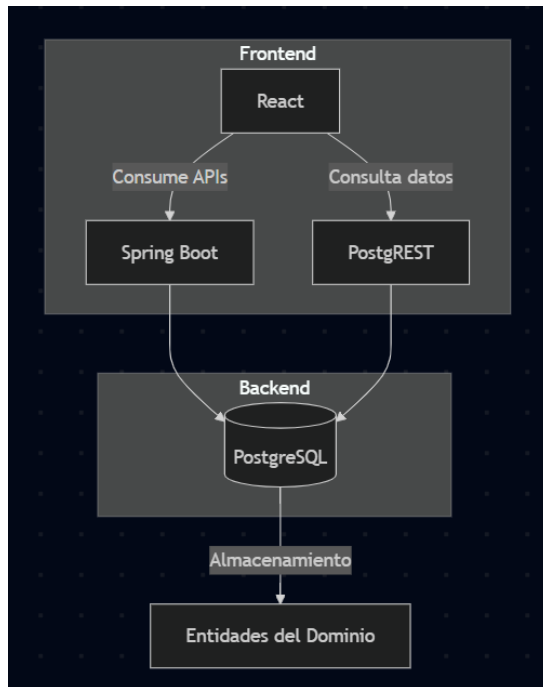
Autenticación con JWT

Autorización por roles

Control de acceso por formularios

Prevención de accesos no autorizados





8. Modelo de datos

El modelo de datos incluye las siguientes entidades principales:

Entidad	Descripción
Users	Registra información de usuarios.
Roles	Define permisos y niveles de acceso.
Students	Datos académicos del estudiante.
Teachers	Información docente.
Groups	Grupos asignados por grado.
Subjects	Materias académicas.
AcademicLoad	Asignaciones por docente.
Agenda	Registro diario por grupo.
AgendaQuestions	Preguntas de la agenda.

AgendaResponses	Respuestas globales e individuales.
ParentConfirmation	Confirmaciones por acudiente.

9. Análisis de riesgos			
Riesgo	Probabilidad	Impacto	Mitigación
Caída del servidor	Media	Alta	Backup, redundancia
Errores en registros diarios	Alta	Media	Validaciones en backend
Saturación por usuarios	Baja	Alta	Escalabilidad horizontal
Accesos no autorizados	Media	Alta	Roles, permisos y logs
Fallas en móviles antiguos	Media	Media	Pruebas por dispositivos

10. Análisis de viabilidad

Viabilidad técnica

El proyecto es viable dada la disponibilidad de tecnologías modernas:

Angular

ASP.NET Core

SQL Server / PostgreSQL

React Native

Viabilidad económica

Los costos se reducen debido a que el proyecto se basa en tecnologías gratuitas o open-source.

Viabilidad operativa

Los actores pueden utilizar el sistema con mínima capacitación.

11. Conclusiones

SchoolMe es una plataforma sólida que responde a necesidades reales de instituciones educativas. El análisis estructural, funcional y arquitectónico demuestra que el sistema puede implementarse con un nivel alto de fiabilidad, escalabilidad y seguridad. Este documento establece el marco técnico necesario para la etapa de diseño e implementación, complementando la especificación de requerimientos elaborada previamente.

12. Referencias

Formato APA 7:

- Pressman, R. (2020). *Software Engineering: A Practitioner's Approach* (8th ed.). McGraw-Hill.
- Sommerville, I. (2019). *Software Engineering* (10th ed.). Pearson.
- Organización de las Naciones Unidas para la Educación, la Ciencia y la Cultura. (2021). *Digital learning and transformation*. UNESCO. <https://www.unesco.org/en/digital-education>