**INSTITUTO TÉCNICO RICALDONE**

**DESARROLLO DE SOFTWARE**



**Nombre del proyecto:**

Sistema de gestión estudiantil.

Gnosis

**Docente:**

Josué Alberto Guinea Henríquez

**Integrantes**

|  |  |  |
| --- | --- | --- |
| **Nombre completo** | **# Carné** | **No trabajo (coloque una X)** |
| José Santiago Merino Herrera | **20190016** |  |
|  |  |  |
| Alejandro Antonio Méndez Letona | **20210120** |  |
|  |  |  |
| Eduardo Alfonso Barahona Vásquez | **20210451** |  |
|  |  |  |
| José Ernesto Gonzales Castillo | **20210590** |  |
|  |  |  |
| Luis Alfredo Castillo Monterrosa | **20210482** |  |

**Fecha de entrega:**

10 de junio de 2022

[San Salvador, El Salvador, CentroaméricaIntroducción 3](#_Toc37970)

[Descripción del proyecto 4](#_Toc37971)

[Modelo del dominio 5](#_Toc37972)

[Diccionario de datos 6](#_Toc37973)

[Diagrama topdown 19](#_Toc37974)

[Diagrama de clases 20](#_Toc37975)

[Diagrama E-R 21](#_Toc37976)

[Validación de login 22](#_Toc37977)

[Diagrama de casos de uso 23](#_Toc37978)

[Diagramas de actividades 28](#_Toc37979)

[Diagramas de colaboración 32](#_Toc37980)

[Instrumentos utilizados para recolectar la información 34](#_Toc37981)

[Estandarizar la codificación de errores 35](#_Toc37982)

[Definición de estándares: 37](#_Toc37983)

[Diseño de interfaces 38](#_Toc37984)

[Conclusiones 71](#_Toc37985)

[Fuentes de información 72](#_Toc37986)

**Índice.**

**Introducción.**

En este documento, se muestran todas las actividades y lineamientos requeridos para la fase de análisis y diseño del Proyecto Técnico Científico, abarcando como primer punto una descripción breve sobre el sistema, junto a su estructura y características, ejemplificando y facilitando su comprensión por medio de los diagramas UML: Modelo de Dominio, Topdown, Clases, Casos de Uso, Actividades y Colaboración son los diagramas que a continuación serán presentados, acerca del Sistema Gnosis.

Siguiendo por el diccionario de datos del sistema, y exponiendo de manera detallada los métodos en los cuales el sistema aplicará seguridad a sus usuarios y sus datos personales (contraseñas), exponiendo los instrumentos o herramientas por las cuales se justifican los datos que el sistema maneja para cumplir con su propósito, al mismo tiempo se mostrarán los estándares que cumple el sistema y determinado formato, al igual que los errores que este regula de manera predeterminada.

Concluyendo con la exposición de las interfaces que el sistema contendrá con los elementos necesarios para su correcto funcionamiento y el código que fue necesitado para realizar la base de datos a utilizar para y con el sistema.

# Descripción del proyecto.

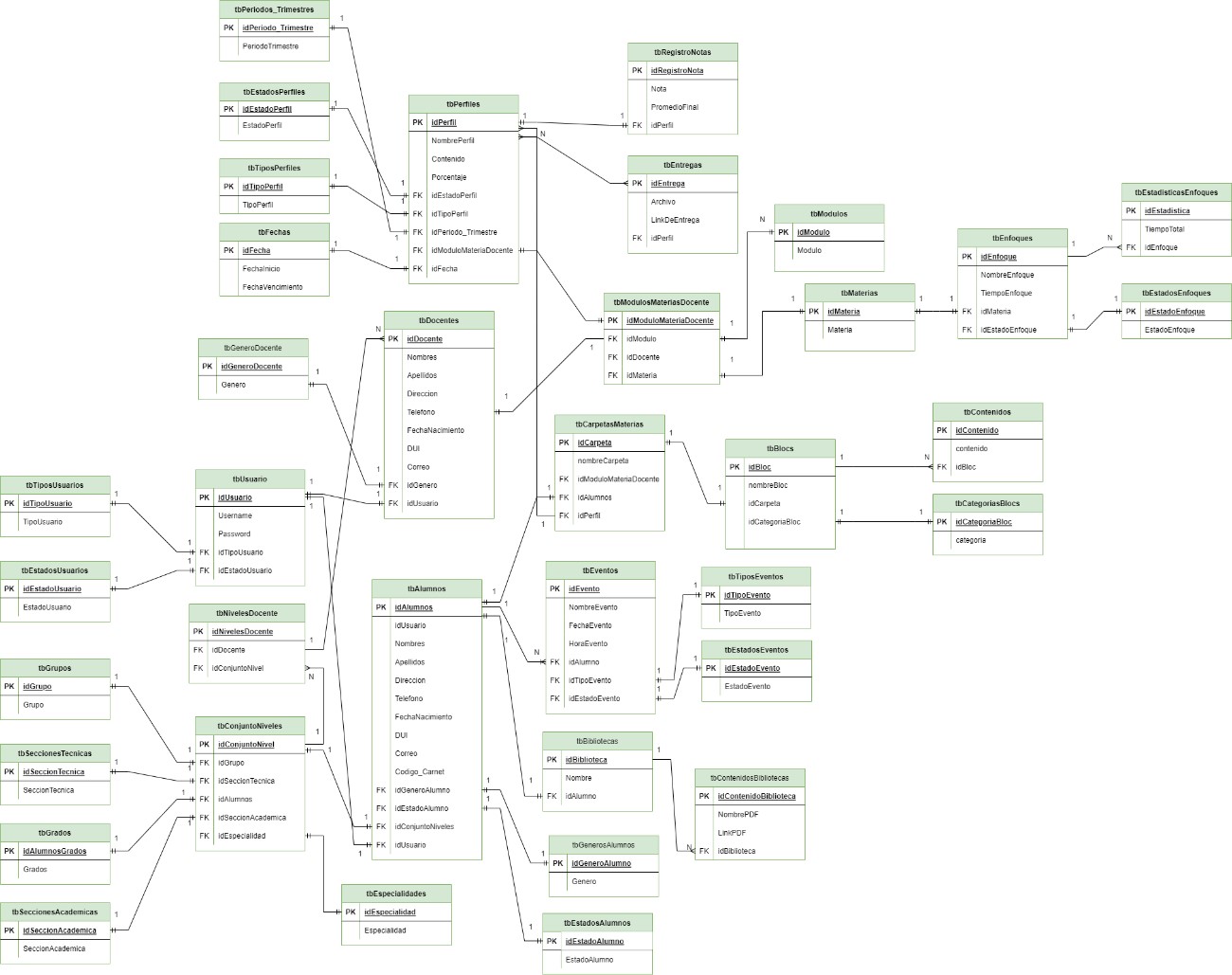
El proyecto por desglosar mediante el análisis y diseño de interfaces es el sistema Gnosis, siendo una combinación entre un sistema de gestión estudiantil (SGE) y un sistema de gestión de aprendizaje (SGA) orientado hacia el Instituto Técnico Ricaldone.

El sistema tendrá la capacidad de almacenar registros sobre los estudiantes y docentes, donde se podrá modificar, crear, eliminar dicha información de manera fácil y cómoda, siendo su mayor característica que el sistema y sus funciones se dividirán entre las interfaces de los estudiantes y docentes, permitiendo que la información entre ambos se pueda compartir a través del sistema.

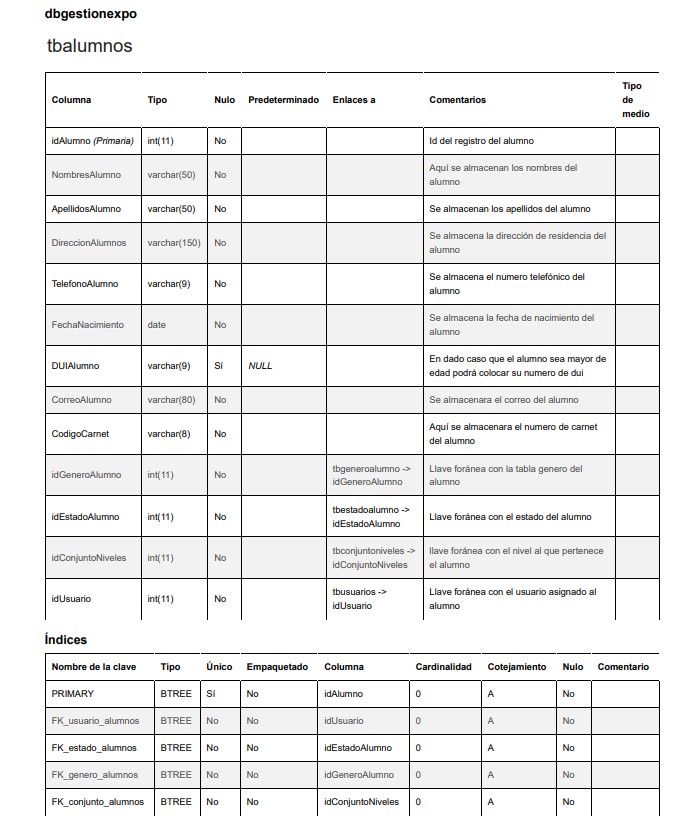
El apartado de estudiantes contendrá un dashboard donde se recopilarán algunas de las principales funciones del sistema de manera minimizada como un acceso y método de vista rápida hacia la información en los apartados, siendo los apartados: un portafolio digital por cada materia que el estudiante curse, un apartado de tareas donde el estudiante pueda consultar las tareas que tiene por realizar, donde podrá consultar en que consiste, consultar el instrumento de evaluación subido por el docente y la fecha limite para entregarlo, el apartado de calificaciones donde podrá ver sus notas sobre cada materia divididas en tres categorías, cotidianas, integradoras y objetivas, siendo capaz de ver los detalles en cada uno de los tipos de actividades, el apartado de biblioteca o recursos permitirá al estudiante almacenar links, pdf y documentos teniendo en un mismo lugar los recursos necesarios y almacenados, también se integra una calculadora que puede resolver los procedimientos básicos matemáticos (suma, resta, división, multiplicación, potencia, y raíz cuadrada), el apartado de calendario que le permite ver su horario de clases y los eventos próximos dentro de la institución, al igual que visualizar sus tareas en el propio calendario y como ultimo característica para los estudiantes podrán activar un modo de concentración que ocultará las materias no deseadas por el alumno, mientras se enfoca en realizar actividades de una sola materia, pudiendo enfocarse en dos modos distintos.

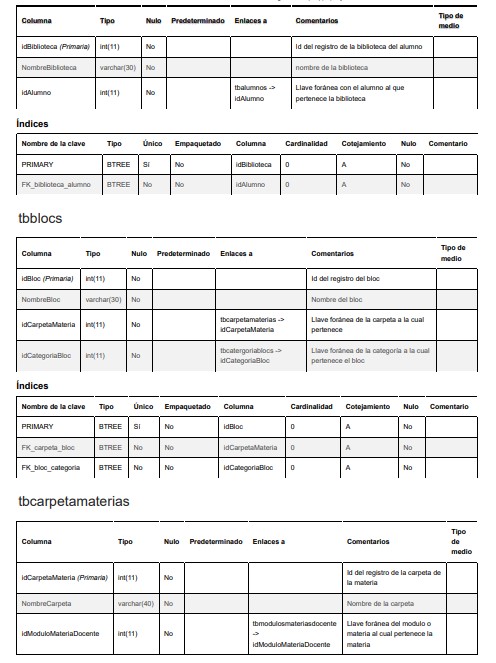
El apartado de docentes contiene las opciones de asignar actividades, asignar las notas de todas las evaluaciones, la vista de sus datos subidos a el sistema y la asignación de eventos a los estudiantes. El apartado de asignar actividades en el cual el docente pueda crear evaluaciones donde el estudiante puede revisar las activades que tiene pendientes, el apartado de asignaciones de notas permite al docente subir las notas que los alumnos tengan y que este mismo pueda visualizar su nota en su apartado, el perfil muestra todos los datos que se hayan guardado en el sistema tanto su información de docente y la asignación de su sección y técnico.

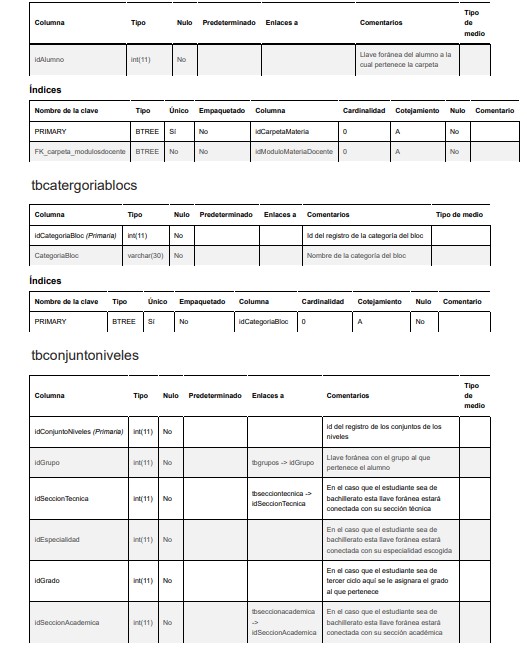
# Modelo del dominio

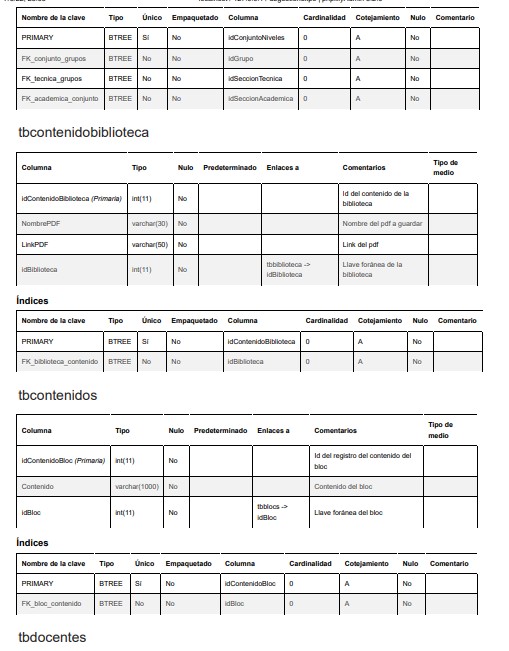


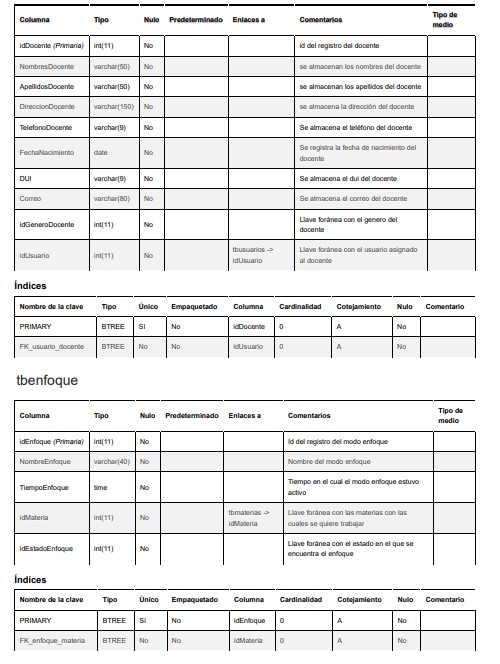
# Diccionario de datos.

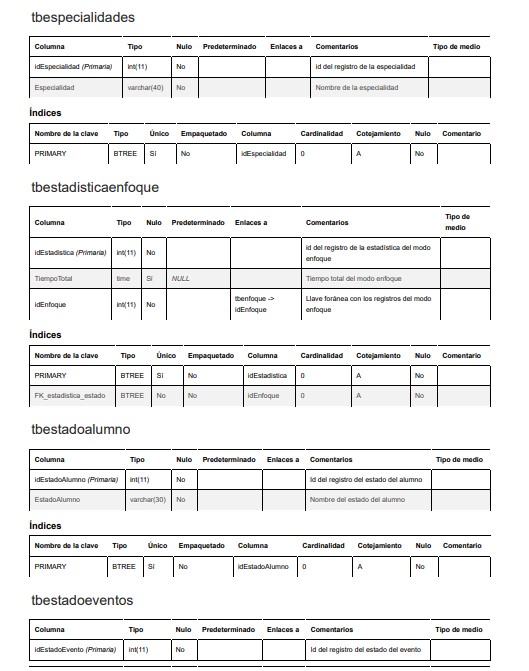


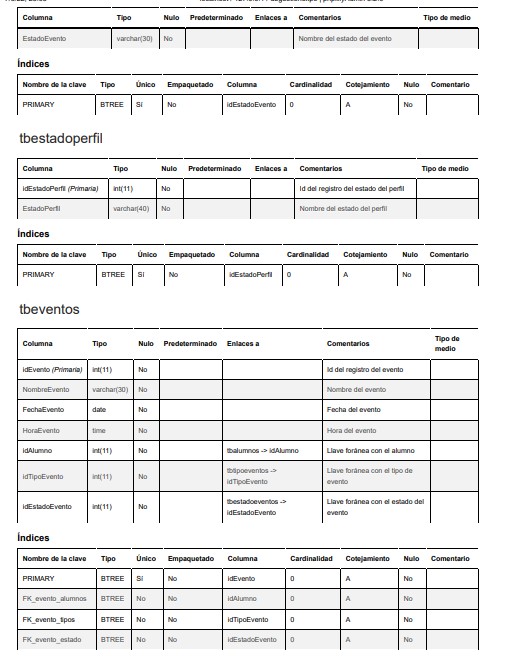


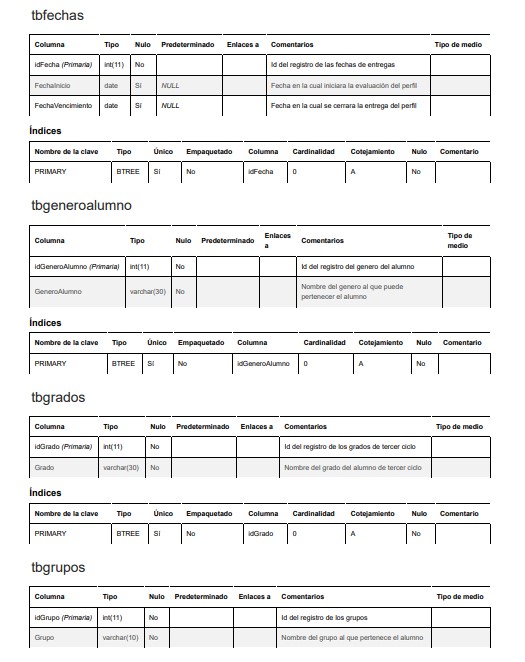


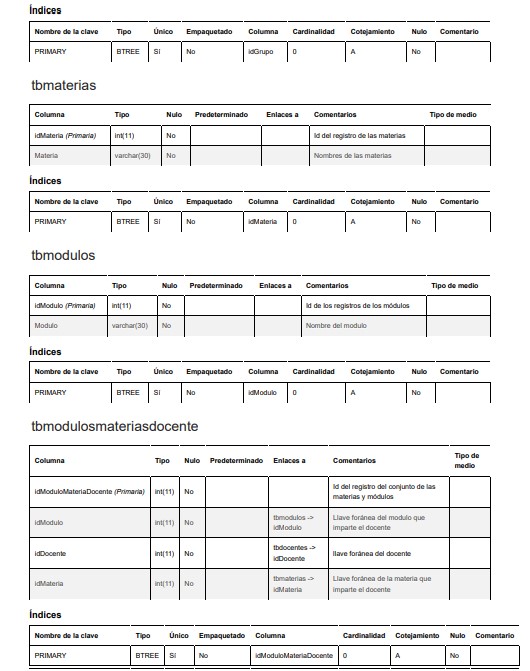


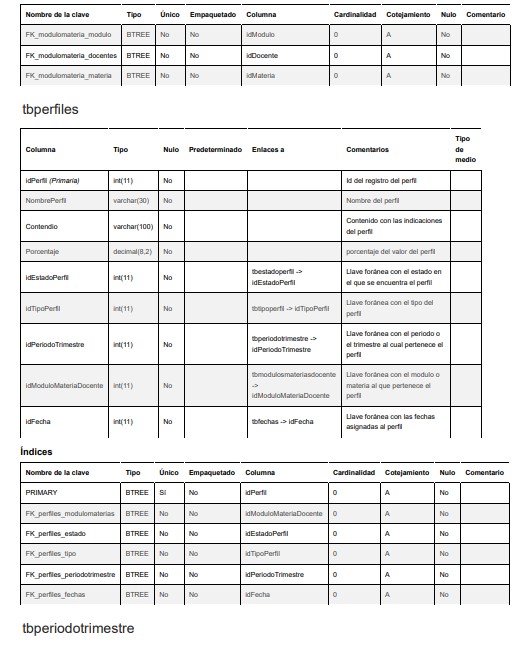


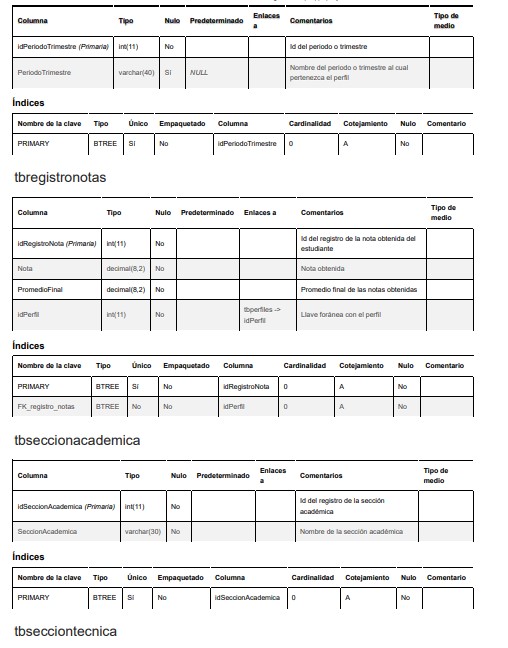


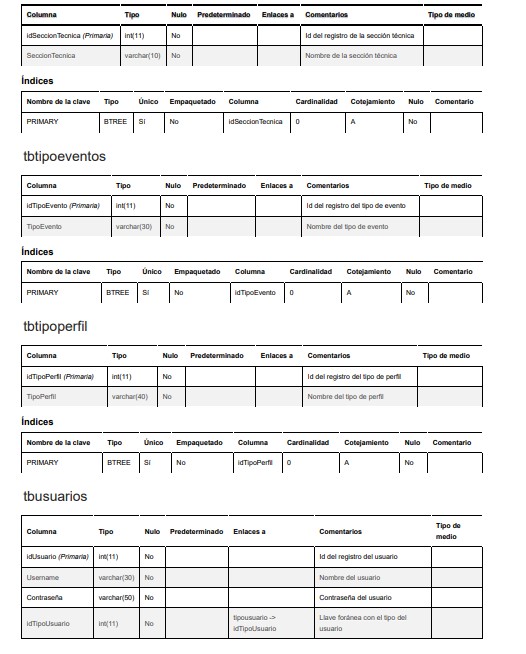


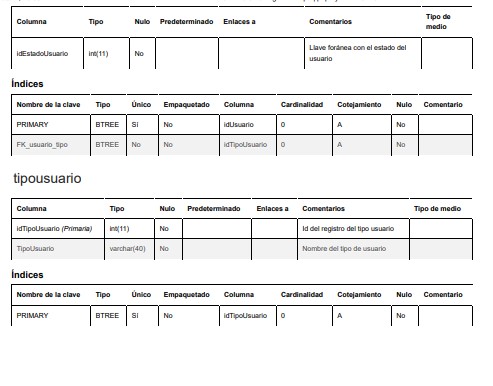




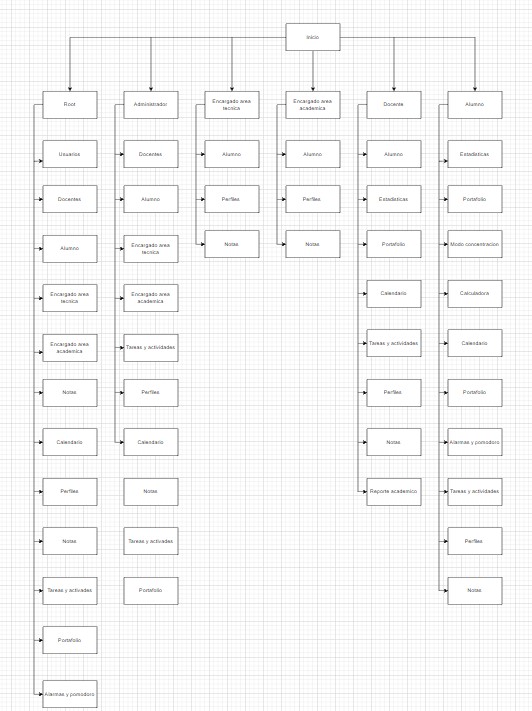




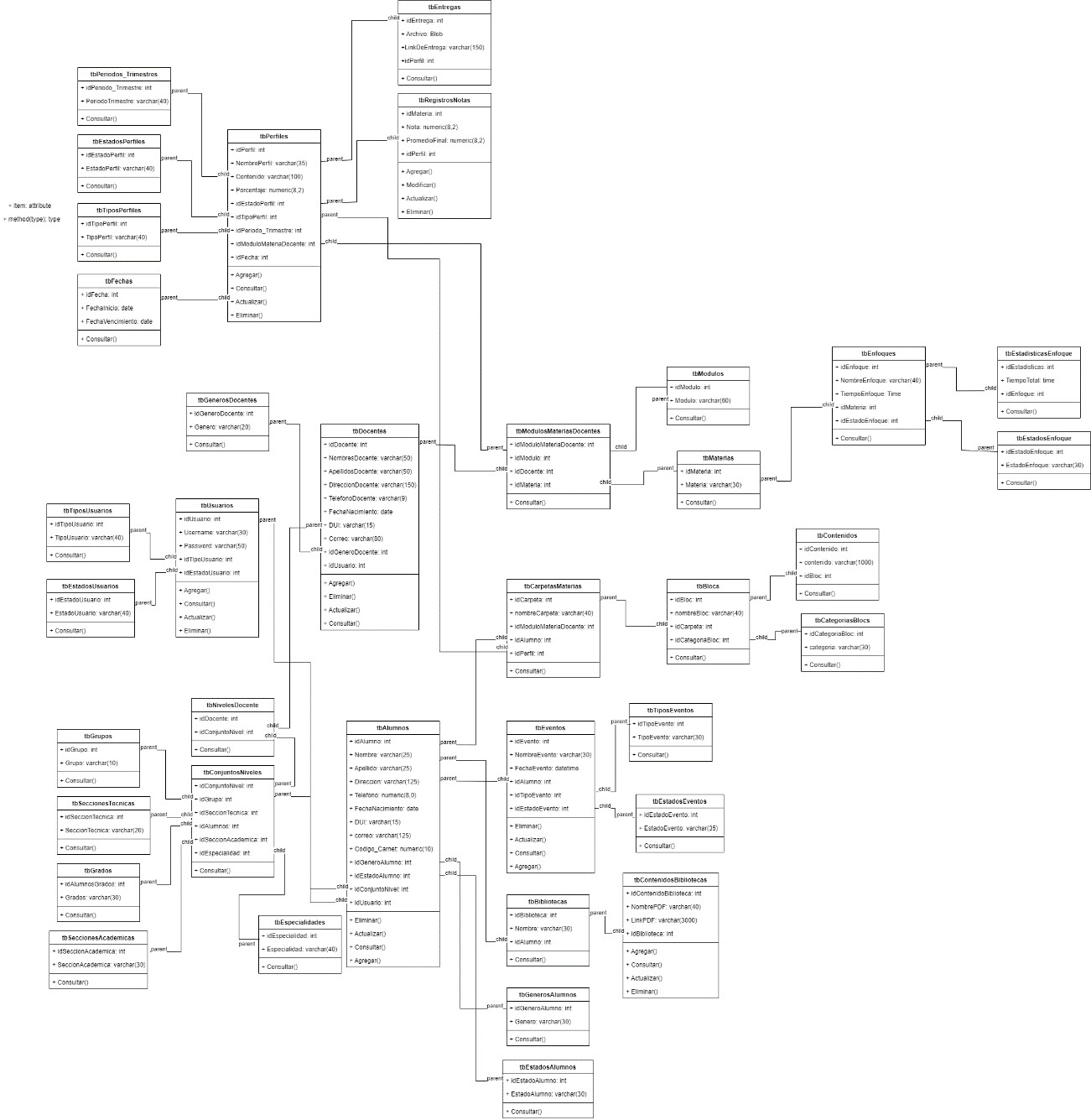




# Diagrama topdown



# Diagrama de clases

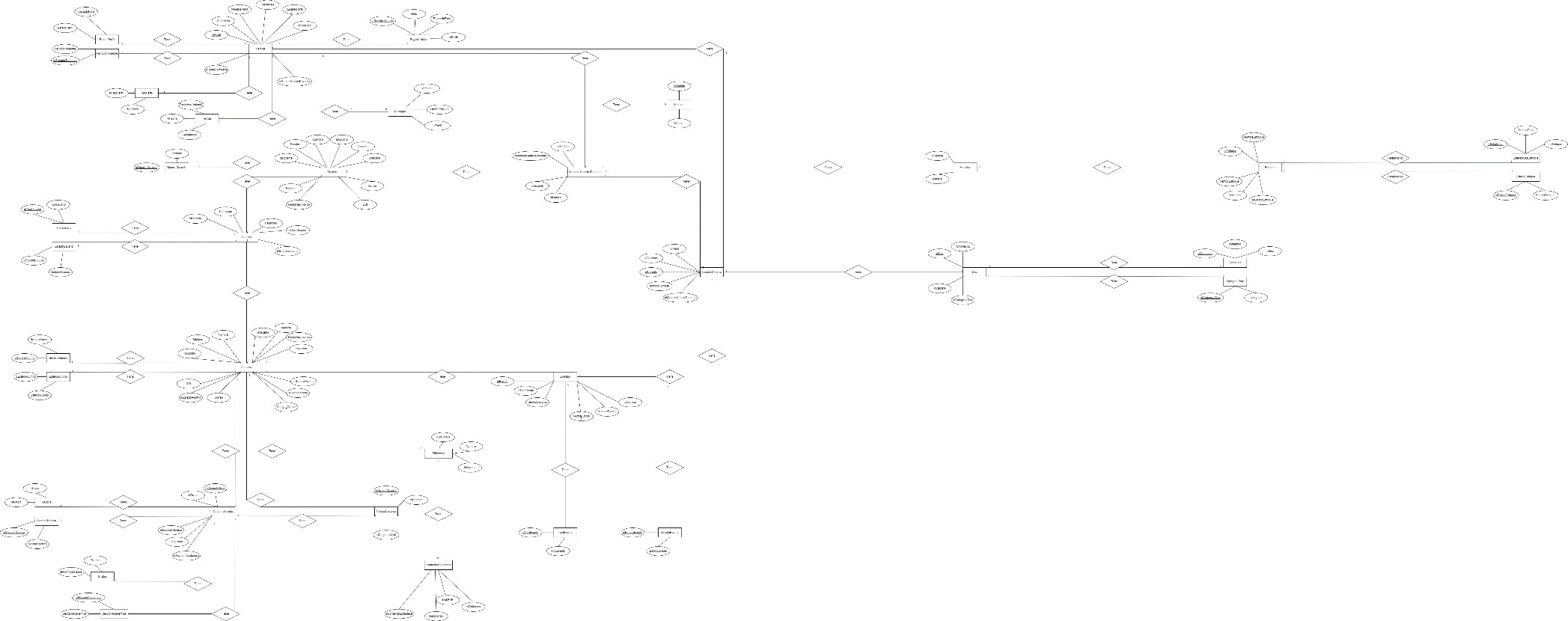


Desarrollo de Software

Instituto Técnico Ricaldone

# Diagrama E-R

21

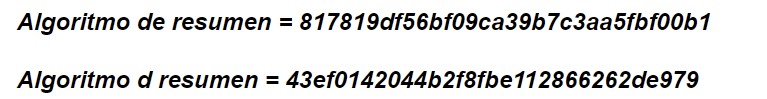


# Validación de login

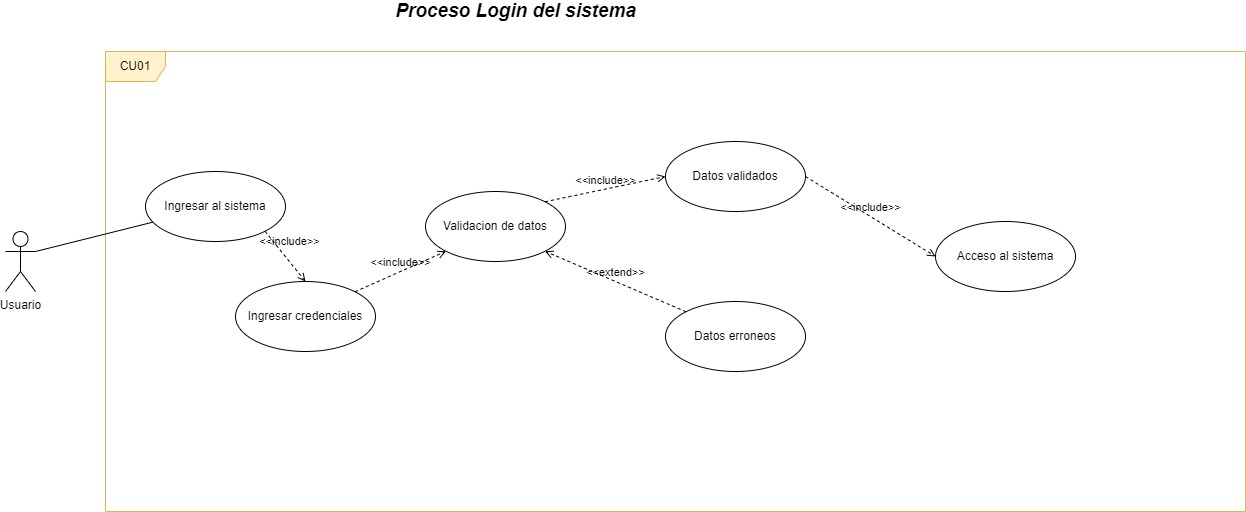
El método de encriptación que se utilizará será por medio del MD5. El método MD5 siendo este un método de encriptación seguro, sencillo y popular, con una función de hash. Es un algoritmo que proporciona un código asociado a un archivo o texto concretos. De esta forma, a la hora de descargar un determinado archivo como puede ser un instalador, el código generado por el algoritmo, también llamado hash, viene "unido" al archivo Los hash son algoritmos que consiguen crear a partir de una entrada de datos, una salida alfanumérica de longitud. El método es sencillo creando una clase que contenga un método que reciba una cadena de texto y regrese otra encriptada.

MD5 permite encriptar el texto que se ingresa, pero no permite desencriptarla.

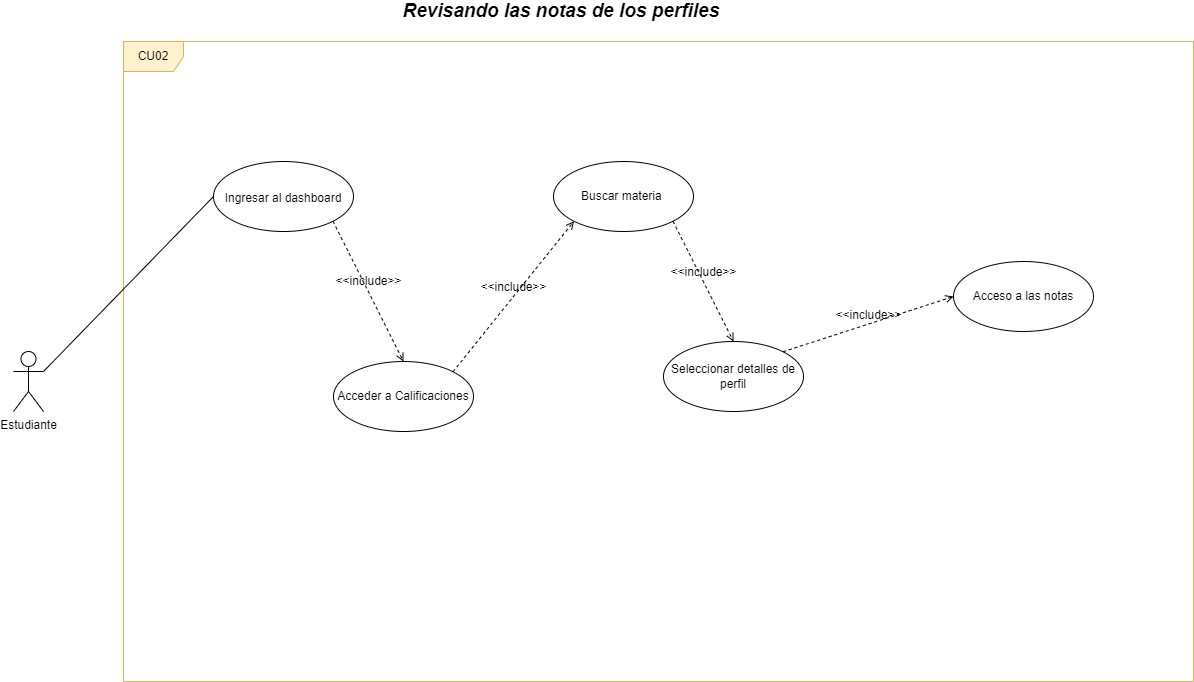
Cuando el usuario introduzca su contraseña, esta es encriptada con MD5 y este texto se va a comparar con lo que se encuentre en la base de datos, es una transformación de visualización que evitan también problemas de codificación. De esta manera la contraseña no se encuentra en texto plano, la persona que logre tener acceso a la base de datos podrá ver el MD5, pero será incapaz de decodificarlo, incapaz de encontrar la contraseña original. Lo que se hace es que el usuario, al iniciar sesión e introducir la contraseña, esta será procesado y sometida al algoritmo MD5, luego se comparará los dos MD5 si son exactamente iguales, se le dará acceso, de lo contrario no.



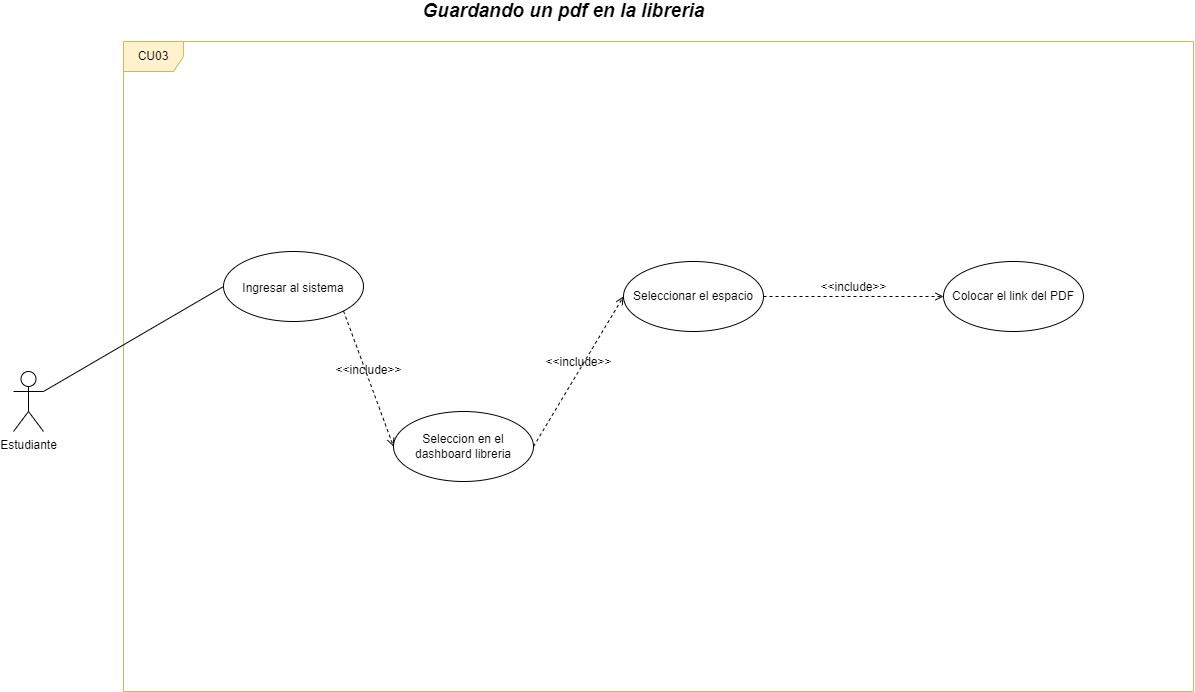
# Diagrama de casos de uso



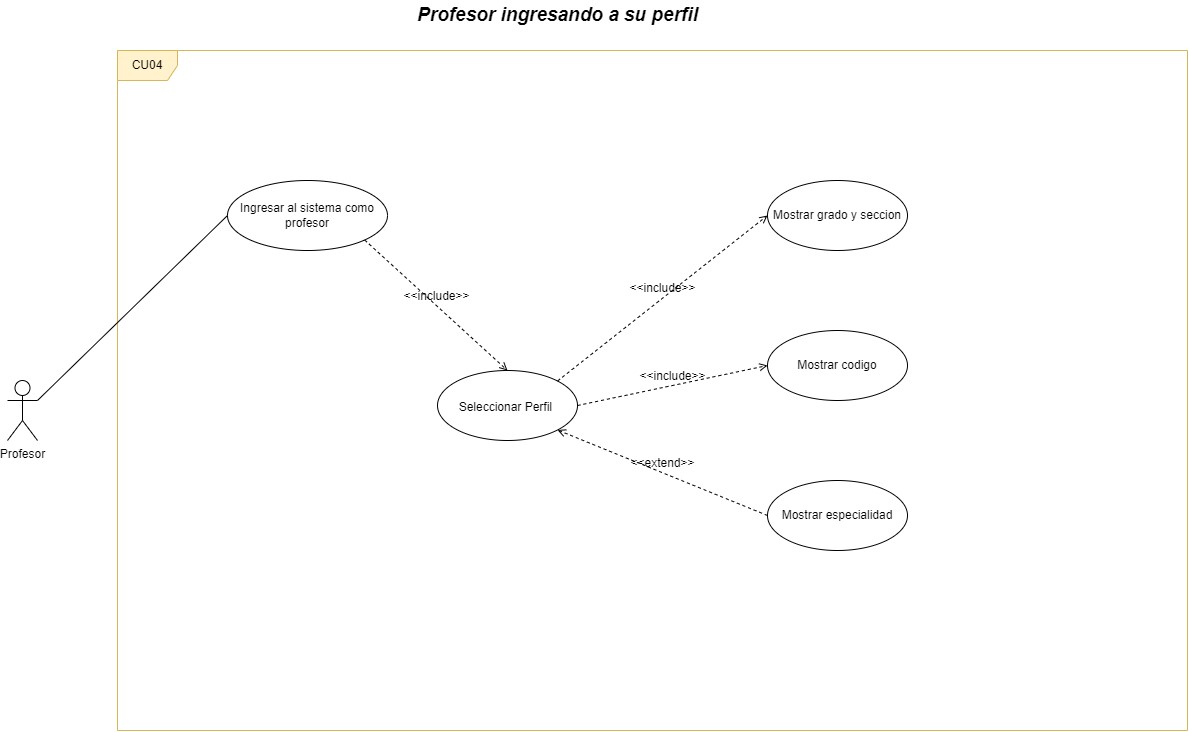
|  |  |
| --- | --- |
| **Nombre del caso de uso:** Proceso Login del sistema. | |
| **Área:** Sistema | |
| **Actores:** Estudiante, Docente, sistema | |
| **Interesados:** Administrador, Root, Supervisores. | |
| **Nivel:** Azul | |
| **Descripción:** El estudiante hace el proceso para poder validarse y acceder al sistema y tener los datos que necesita a la mano. | |
| **Evento desencadenador:** La necesidad de acceder al sistema por una propia razón. | |
| **Tipo de desencadenador:**  Tipo externo Tipo temporal | |
| **Pasos realizados (ruta principal).**  **1. Ingresar al sistema.** | **Información para los pasos.**  Ingresar al sistema por medio de el dispositivo. |
| **2. Ingresar las credenciales.** | Al momento de haber ingresado al sistema este accede al sistema y se le solicita sus datos. |
| **3. Validación de datos.** | El sistema accede a la base de datos para poder comprobar que las credenciales si concuerdan con lo que el estudiante ha colocado. |
| **3.1. Datos erróneos.** | La base de datos no encontró ningún dato relacionado con el que se coloco. |
| **4. Datos validados.** | La base de datos encontró los datos que coincidan. |
| **5. Acceso al sistema.** |  |
| **Precondiciones:** El usuario tiene su cuenta con su Usuario y contraseña en el sistema. | |
| **Postcondiciones:** El usuario pudo acceder con total éxito al sistema | |
| **Suposiciones:** El sistema esta correctamente conectado con la base de datos. | |
| **Garantía de éxito:** El sistema dirige al usuario a sus dashboard según el tipo de usuario que sea. | |
| **Garantía de mínima:** El sistema provee el acceso correctamente. | |
| **Requerimientos cumplidos:** Tener un dispositivo que permita la conexión con la base de datos. | |
| **Cuestiones pendientes: ¿?** | |
| **Prioridad:** Alta | |
| **Riesgo:** Alta | |



|  |  |
| --- | --- |
| **Nombre del caso de uso:** Revisando las notas de perfiles | |
| **Área:** El sistema | |
| **Actores:** Estudiante | |
| **Interesados:** Docente, Encargados, Administradores. | |
| **Nivel:** Azul | |
| **Descripción:** El estudiante accede al sistema y revisa las calificaciones asignadas por el docente en los perfiles correspondientes de la materia en cuestión. | |
| **Evento desencadenador:** El estudiante requiere de ver sus notas de una actividad o perfil. | |
| **Tipo de desencadenador:**  Externo Temporal | |
| **Pasos realizados (ruta principal).**  **1. Ingresar al dashboard.** | **Información para los pasos.**  El estudiante accede al sistema y con un acceso exitoso se dirige al dashboar. |
| **2. Acceder a Calificaciones.** | En el dashboard en el menú de navegación selecciona el apartado Calificaciones. |
| **3. Buscar materia.** | En el apartado de Calificaciones busca la materia. |
| **4. Seleccionar detalles de perfil.** | En Calificaciones se selecciona a mostrar detalles para poder observar las notas de cada perfil y la nota global. |
| **5. Acceso a las notas.** |  |
| **Precondiciones:** El estudiante ya ha accedido al sistema y su nota está ingresada. | |
| **Postcondiciones:** El estudiante pudo observar su nota en el sistema | |
| **Suposiciones:** Que el docente haya puesto su nota. | |
| **Garantía de éxito:** El estudiante puede observar su nota en el perfil y su nota global en la materia. | |
| **Garantía de mínima:** Se muestra el estado de la nota en el sistema. | |
| **Requerimientos cumplidos:** Permitir al estudiante poder observar sus calificaciones. | |
| **Cuestiones pendientes: ¿**Mostrar los datos exactos de su evaluación**?** | |
| **Prioridad:** Alta | |
| **Riesgo:** Alta | |



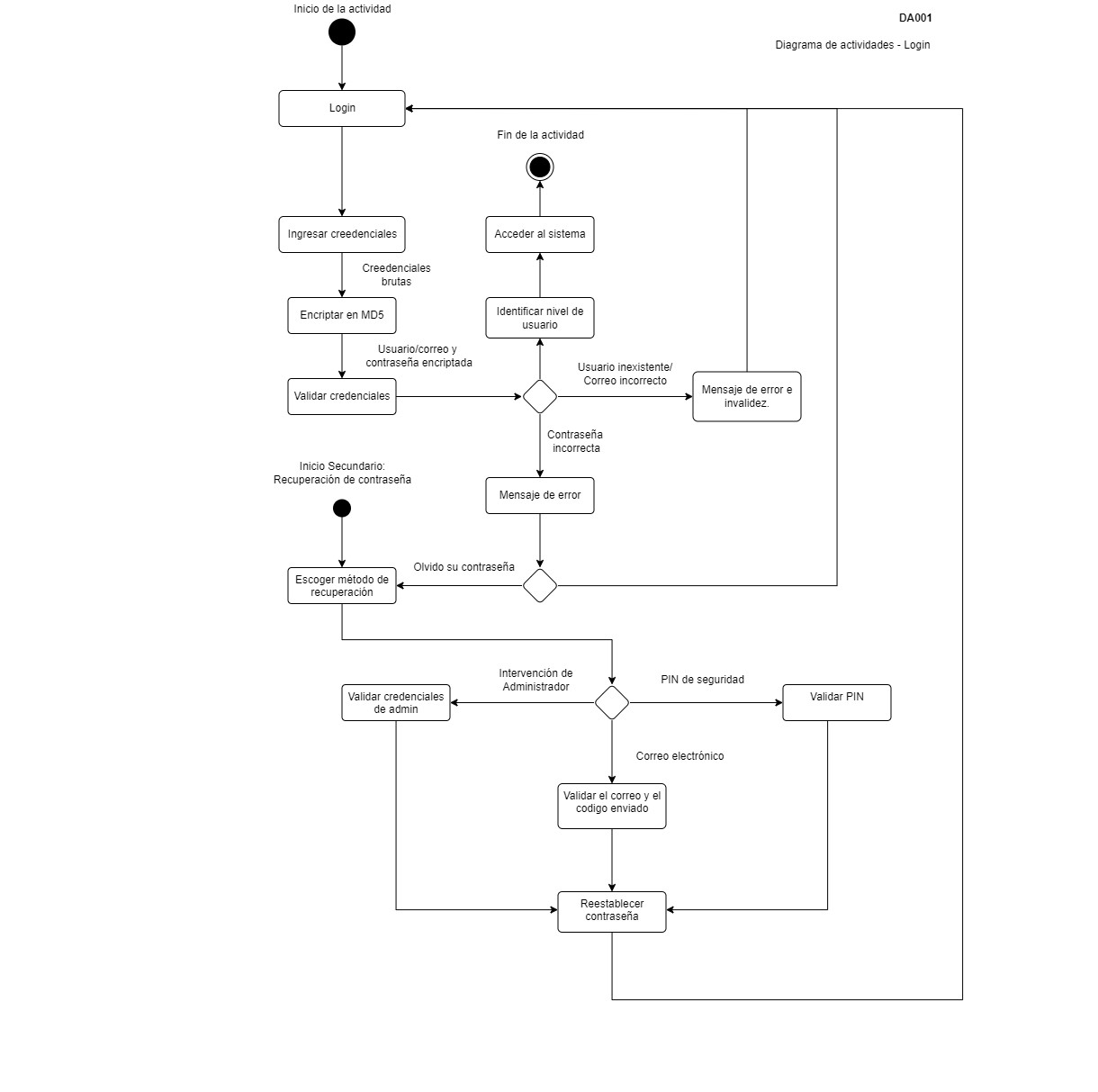
|  |  |
| --- | --- |
| **Nombre del caso de uso:** Guardando un PDF en la librería. | |
| **Área: Librería.** | |
| **Actores:** Estudiante, Sistema. | |
| **Interesados: Docente, Administradores.** | |
| **Nivel:** Azul | |
| **Descripción: Permitir que el estudiante pueda guardar los links de los PDF que considere importante para su estudio.** | |
| **Evento desencadenador: El estudiante obtiene o consigue un PDF que le interesa y le conviene mucho.** | |
| **Tipo de desencadenador: Tipo externo Temporal** | |
| **Pasos realizados (ruta principal).**  **1.** Ingresar al sistema | **Información para los pasos.**  El estudiante ingresa a su cuenta al sistema con las credenciales de estudiante. |
| **2.** Selección en el dashboard librería | En el dashboard seleccionar en el menú de navegación la opción de librería. |
| **3.** Seleccionar el espacio | En el espacio de librería adjuntar el link del PDF y guardar lo con el nombre que quiere tener. |
| 4. Colocar el link del PDF |  |
| **Precondiciones:** El estudiante ya tiene el PDF y de que categoría es. | |
| **Postcondiciones:** El sistema pudo guardar el nombre y el link del PDF para el estudiante. | |
| **Suposiciones:** El estudiante tiene corriendo el sistema perfectamente. | |
| **Garantía de éxito:** El PDF Esta guardado con éxito y con su categoría. | |
| **Garantía de mínima:** El PDF esta guardado con su nombre y link. | |
| **Requerimientos cumplidos:** Los PDF están guardados en las librerías de los estudiantes correctamente. | |
| **Cuestiones pendientes:** ¿Cómo guardar PDF desde los archivos correctamente? | |
| **Prioridad:** Media | |
| **Riesgo:** Medio | |

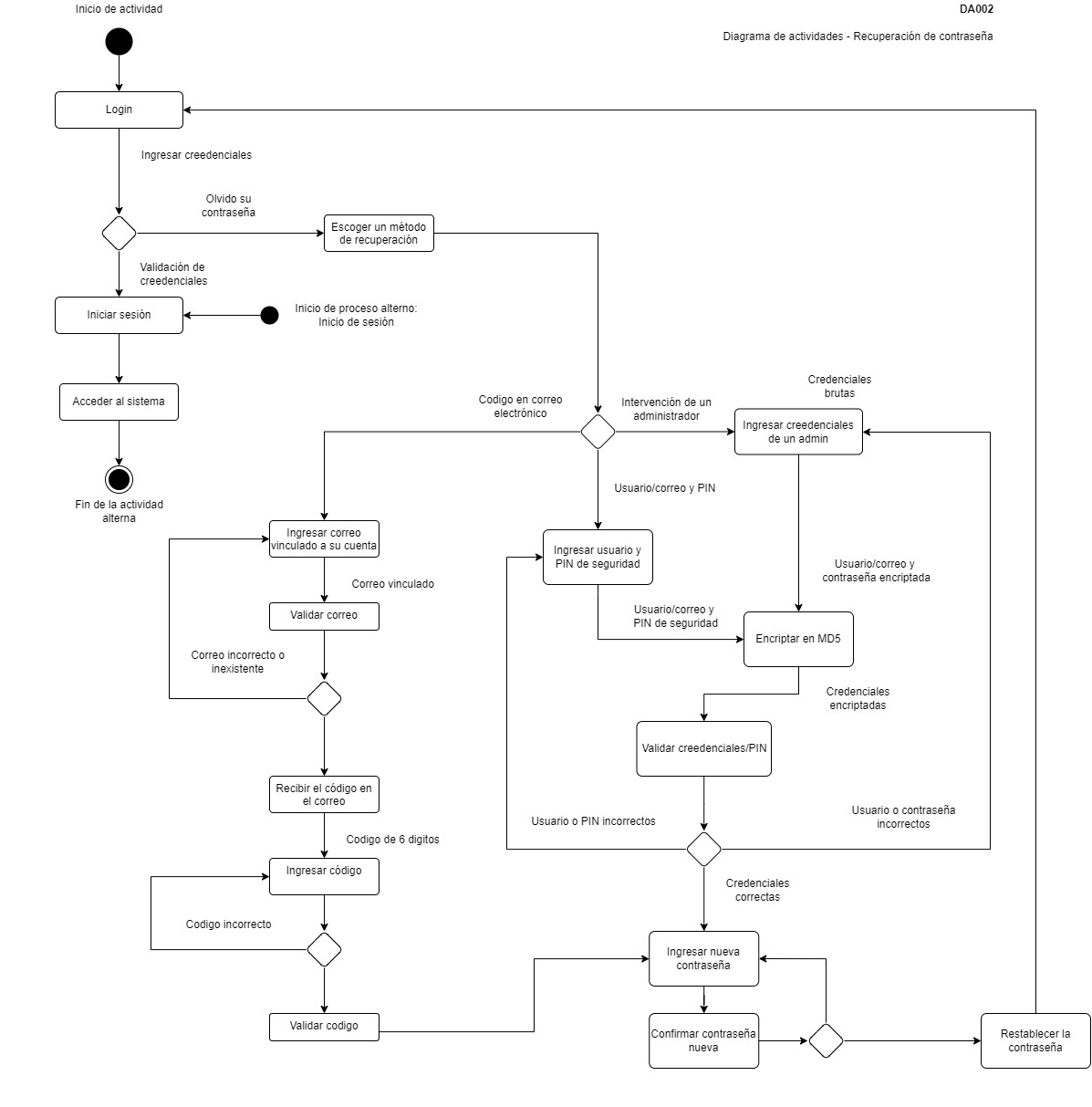


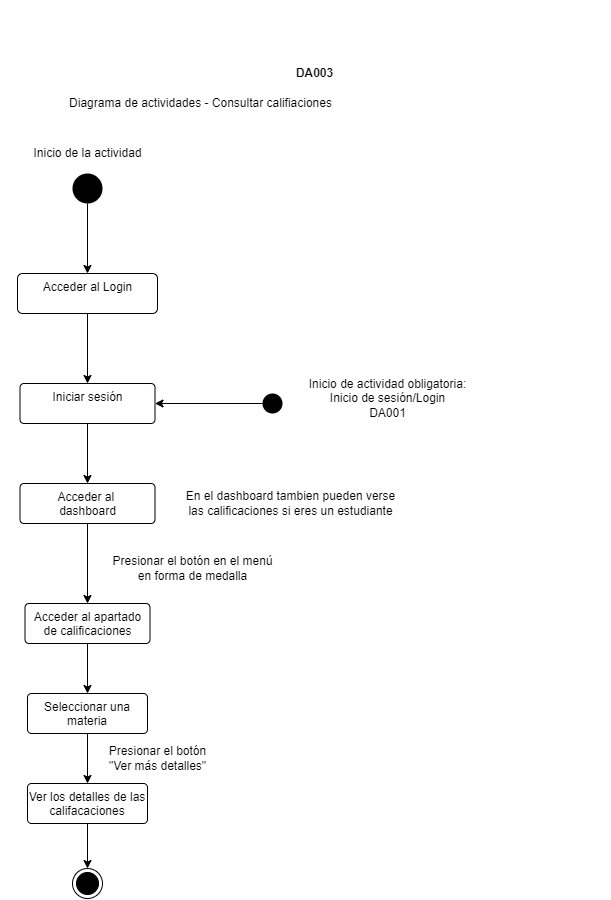
**Nombre del caso de uso:** Profesor ingresando a su perfil.

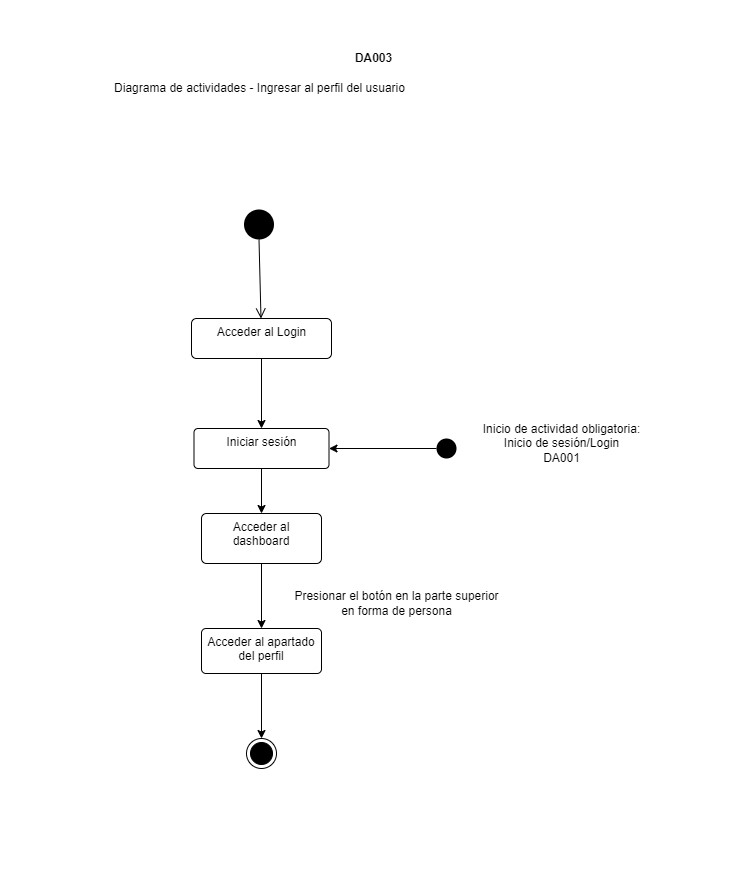
|  |  |
| --- | --- |
| **Área:** Perfiles | |
| **Actores:** Docentes | |
| **Interesados:** Encargados, Administradores | |
| **Nivel:** Azul | |
| **Descripción:** El docente accede al sistema para poder observar sus datos colocados en el sistema. | |
| **Evento desencadenador:** El docente accede a la sección de perfil. | |
| **Tipo de desencadenador:**  externo temporal | |
| **Pasos realizados (ruta principal).**  **1. Ingresar al sistema como profesor.** | **Información para los pasos.**  El Docente con sus credenciales accede al sistema como un Docente. |
| **2. Seleccionar perfil.** | Después de acceder como docente en el sistema este se traslada al menú de navegación y selecciona el apartado Perfil |
|  | Se muestra todos sus datos subidos al sistema. |
| **Precondiciones:** El docente se encuentra en el sistema. | |
| **Postcondiciones:** El docente puede confirmar todos los datos almacenados en el sistema. | |
| **Suposiciones:** La institución cargo todos los datos del docente. | |
| **Garantía de éxito:** Se muestran todos los datos correctamente. | |
| **Garantía de mínima:** El docente tiene acceso a su perfil en el sistema. | |
| **Requerimientos cumplidos:** El sistema posee todos los datos que la institución a proporcionado almacenados en el apartado de su perfil. | |
| **Cuestiones pendientes:** ¿Se puede mostrar información personal del docente? | |
| **Prioridad:** Alta | |
| **Riesgo:** Media | |

# Diagramas de actividades.

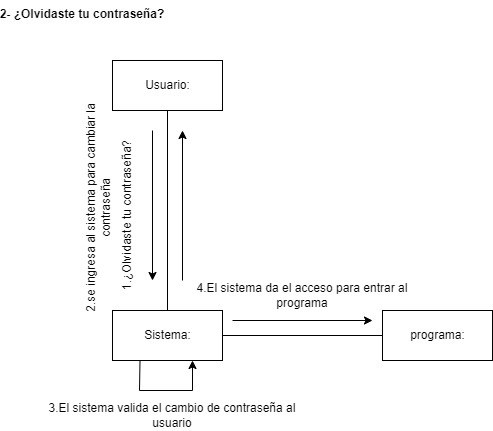
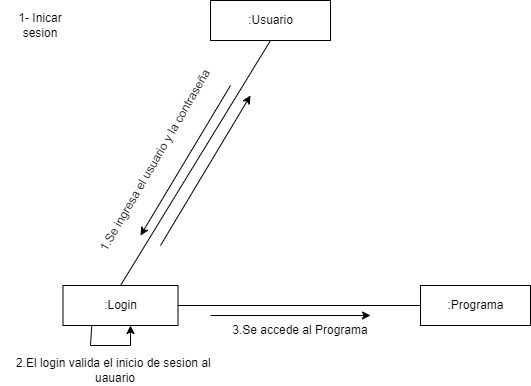


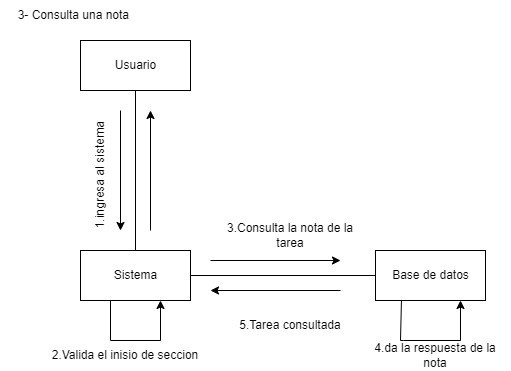


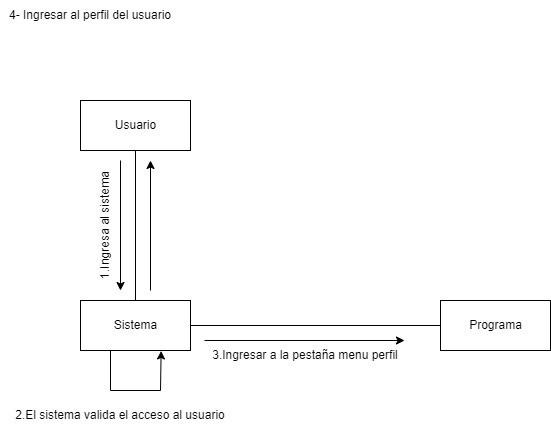




# Diagramas de colaboración.





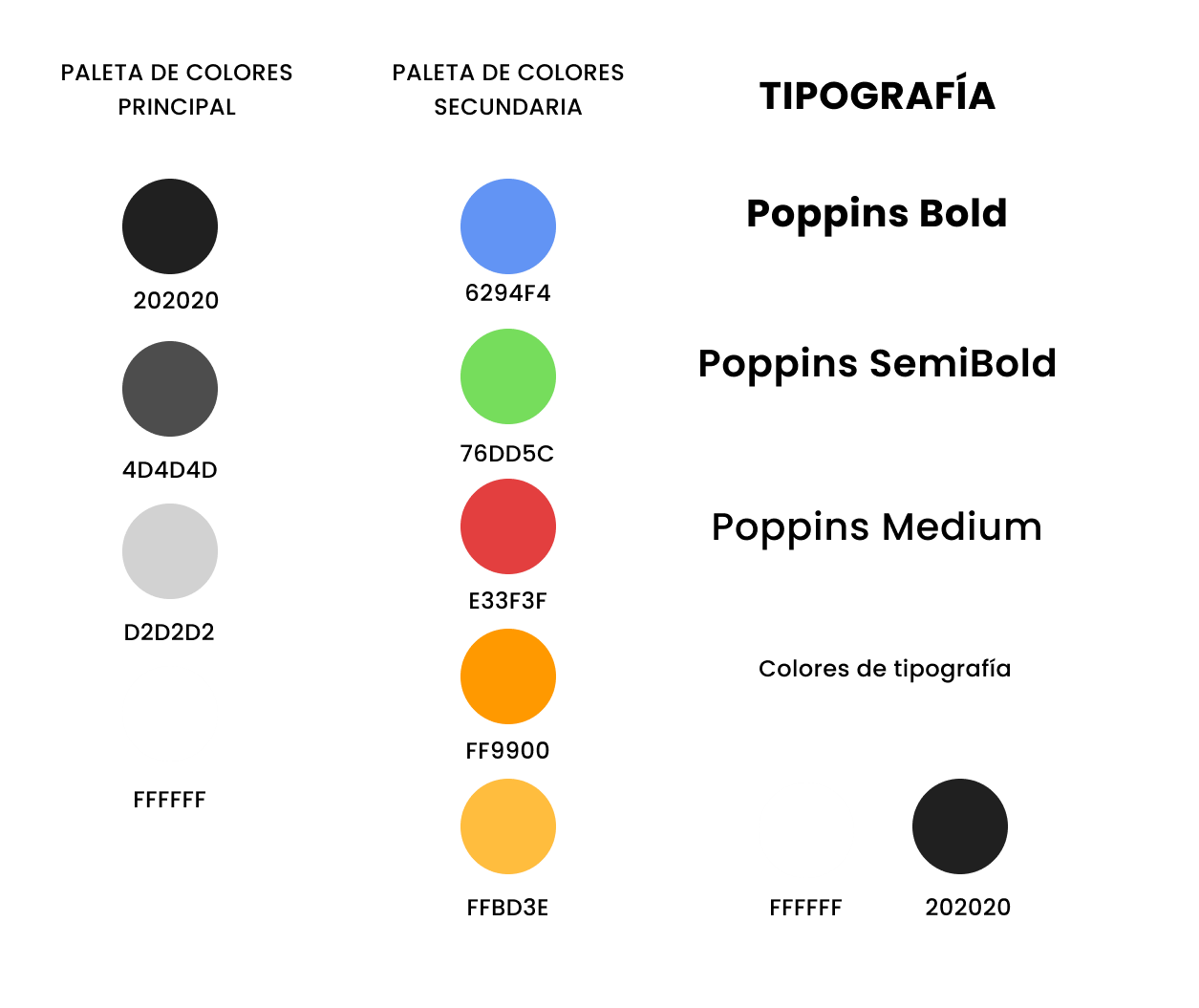
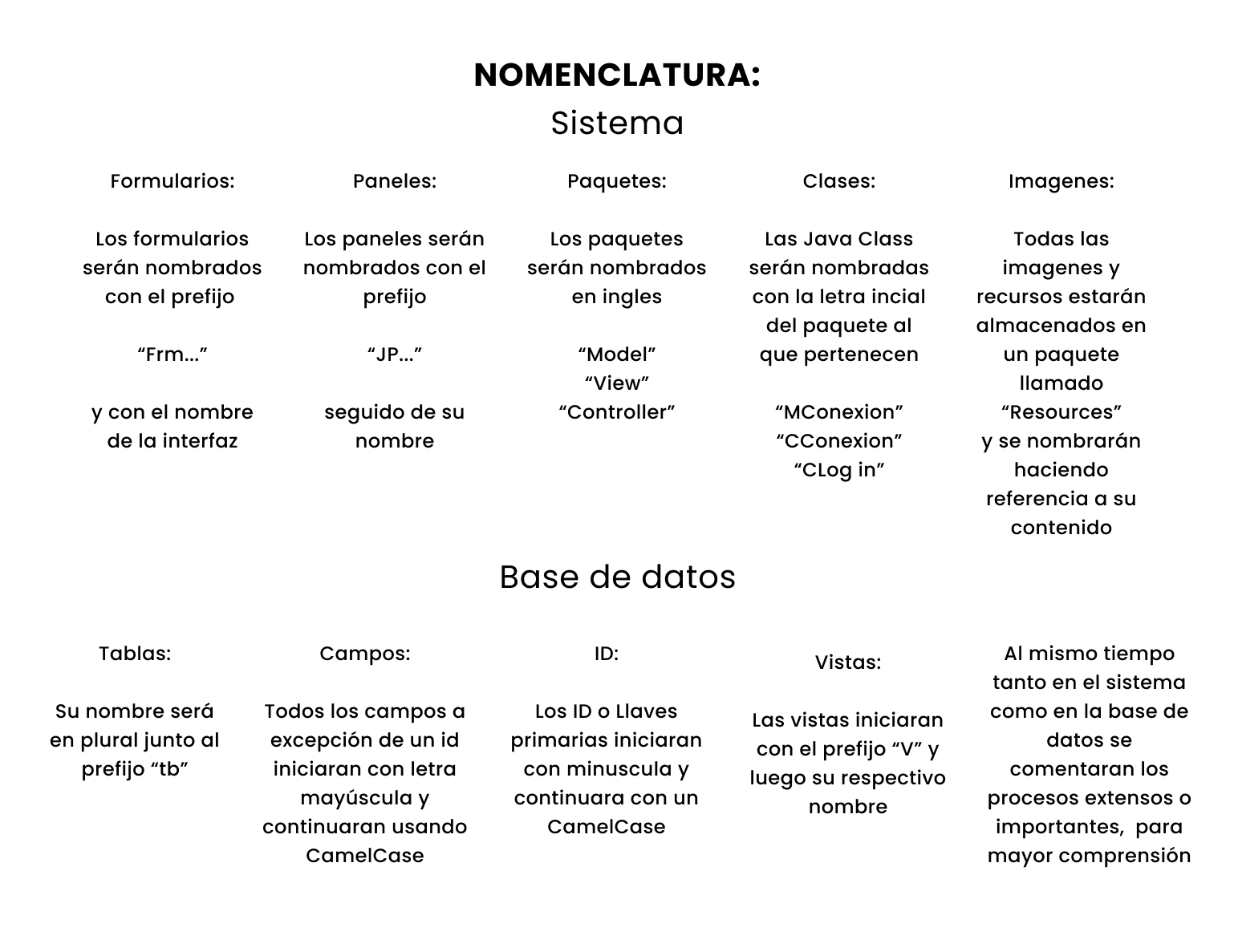


# Instrumentos utilizados para recolectar la información.

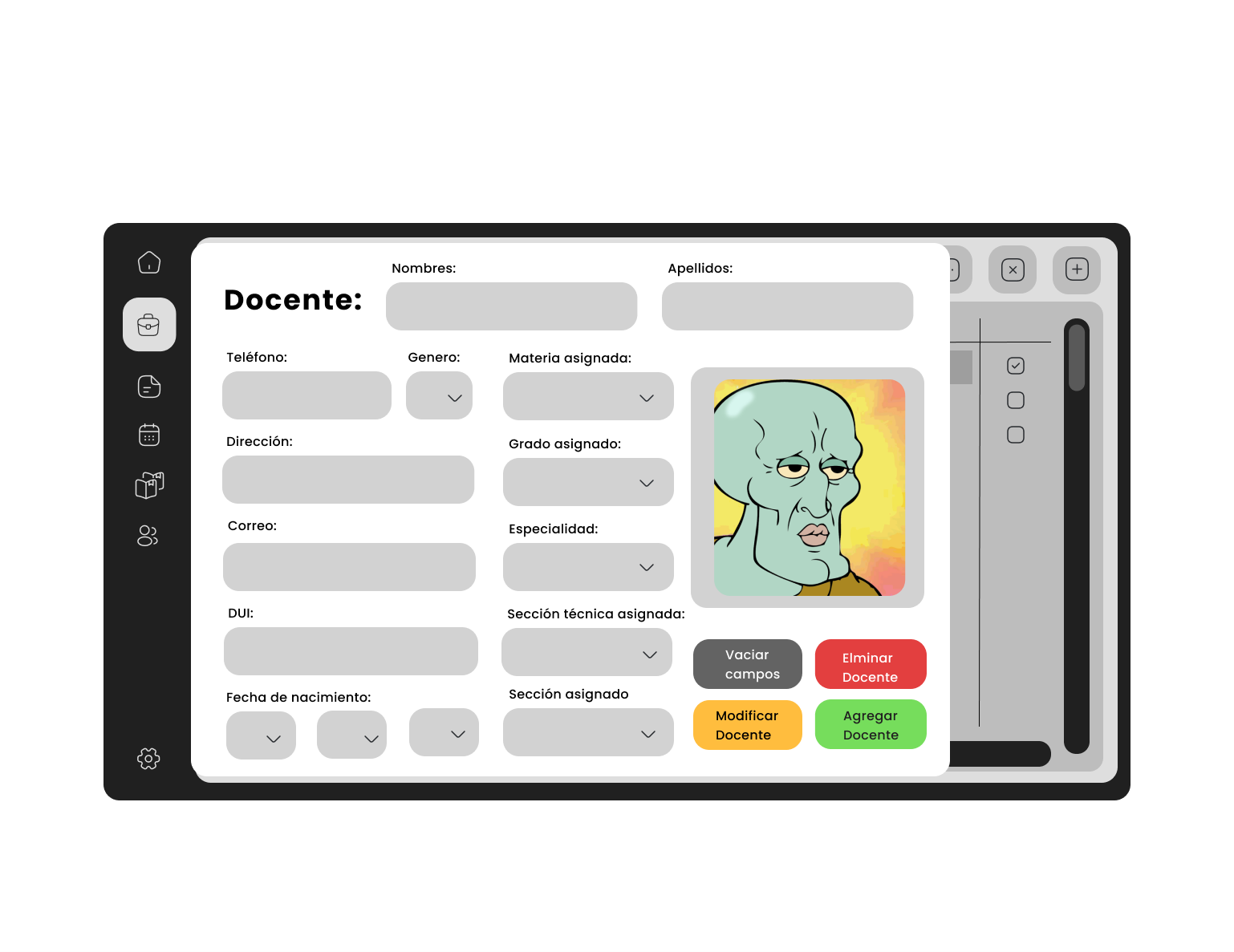
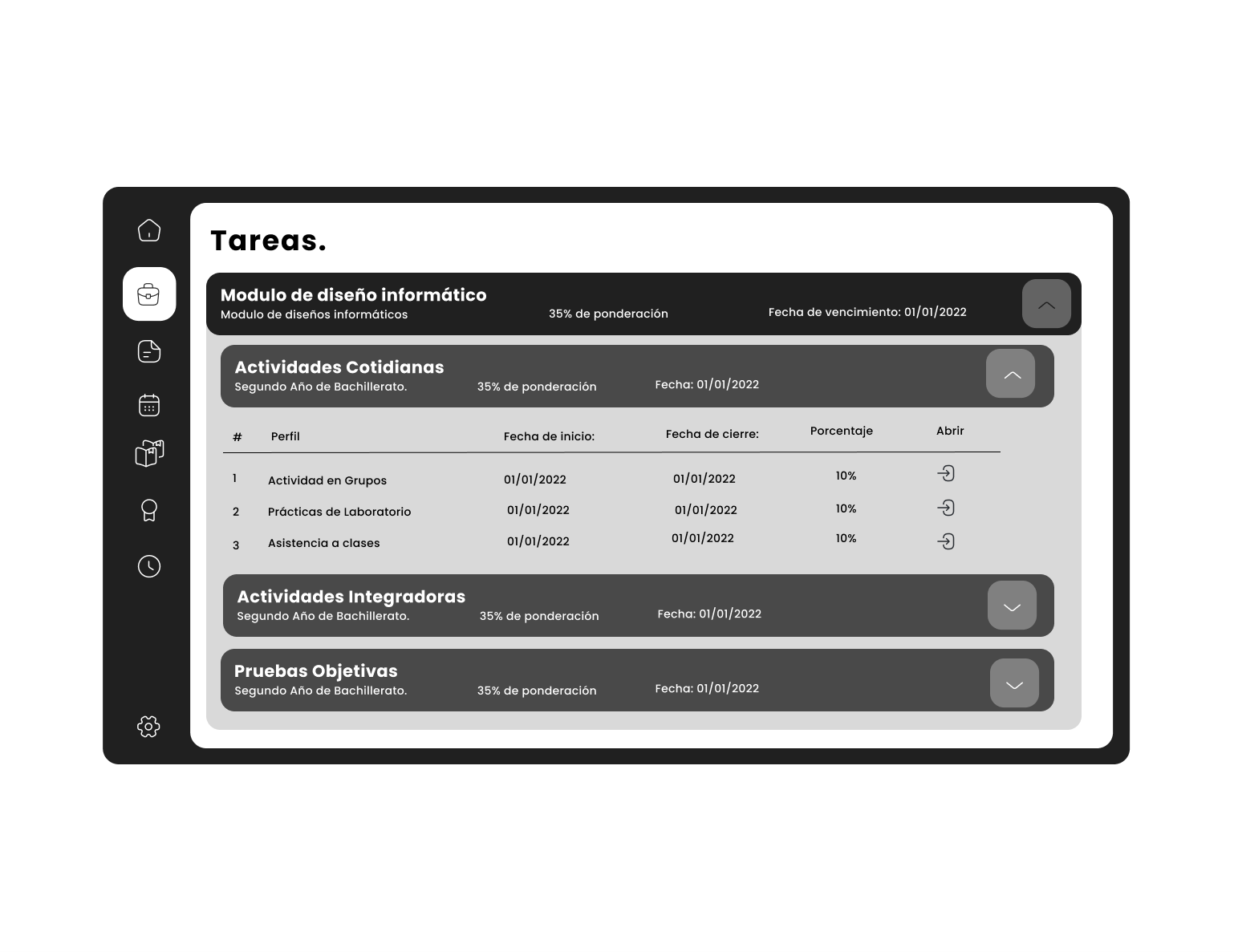
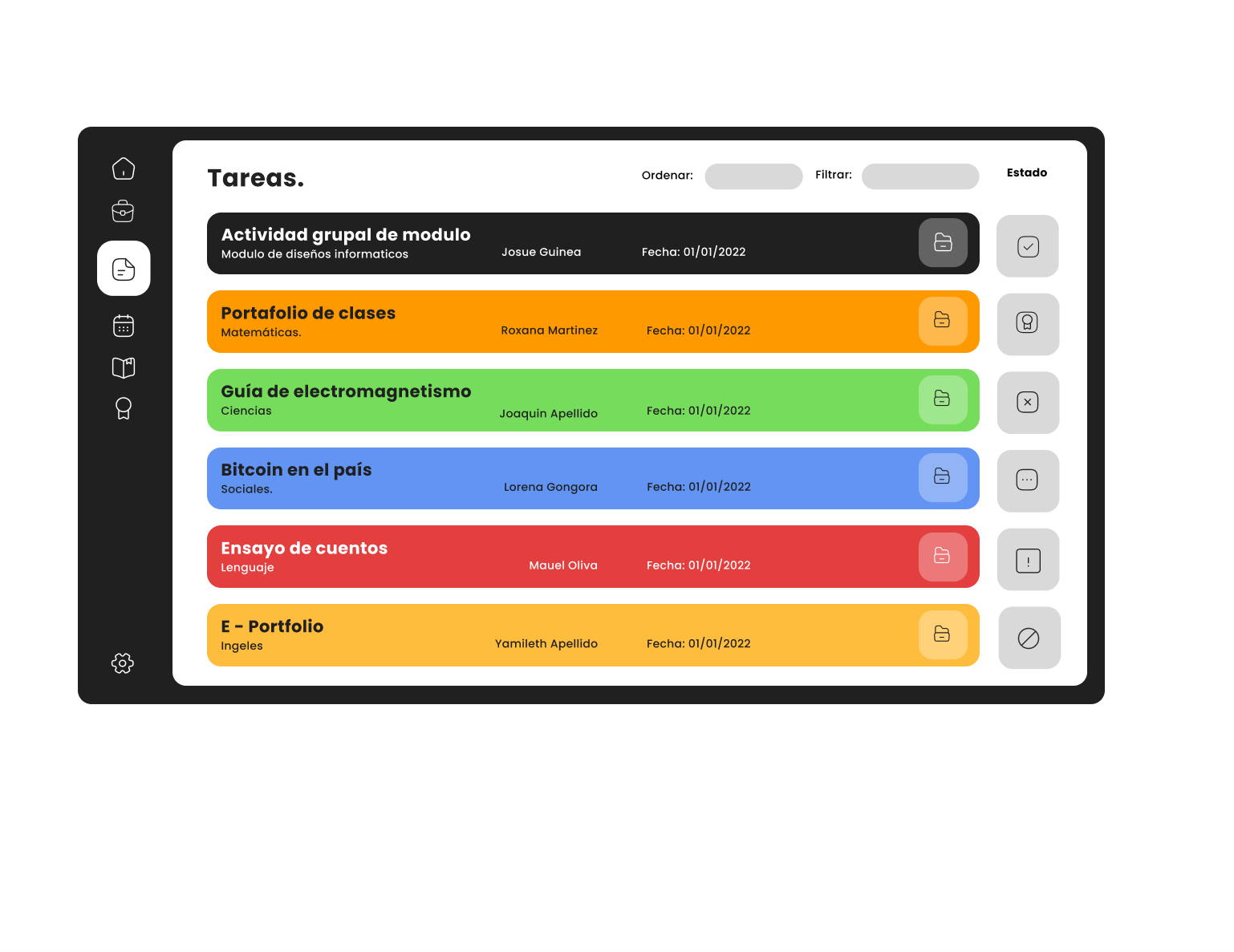
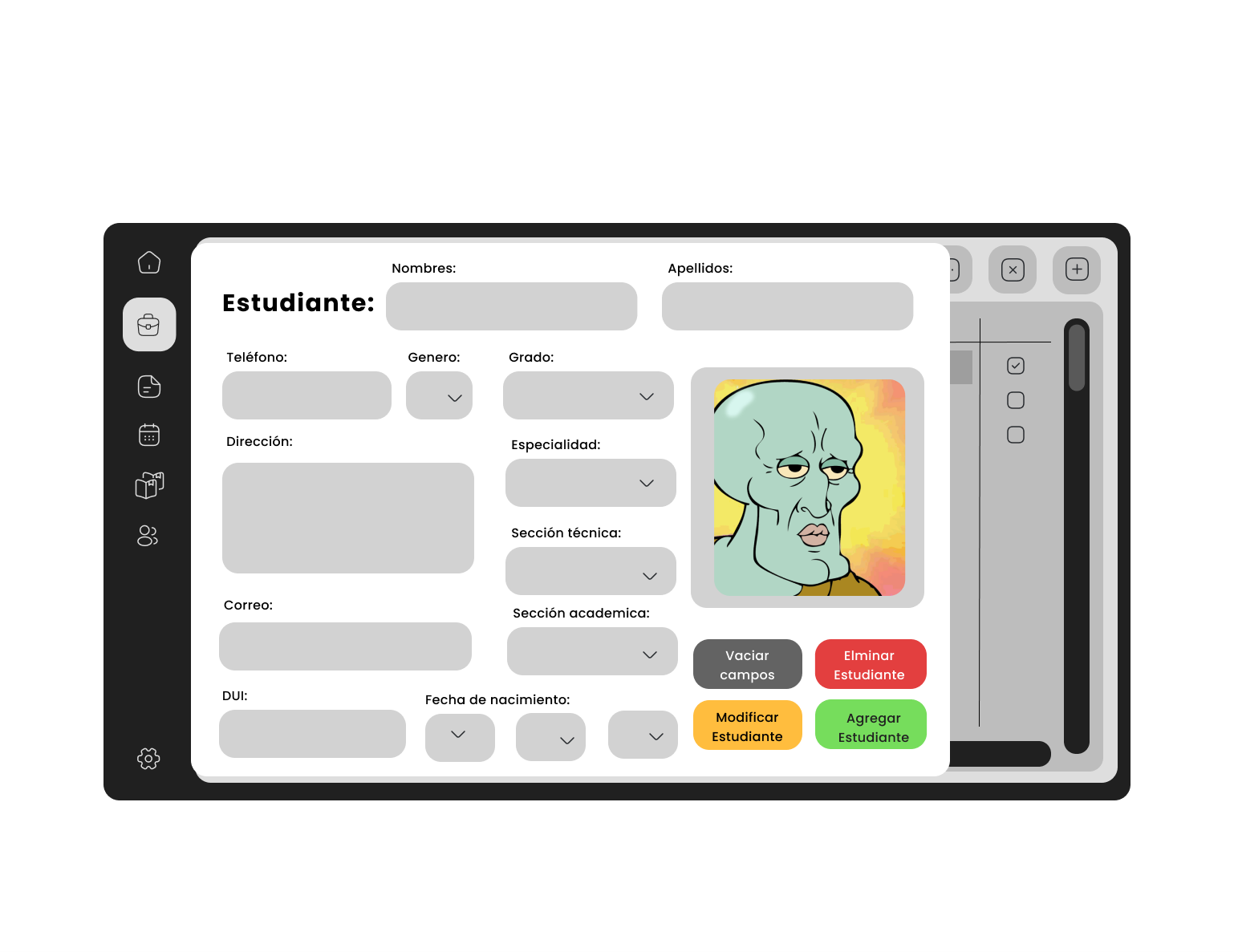
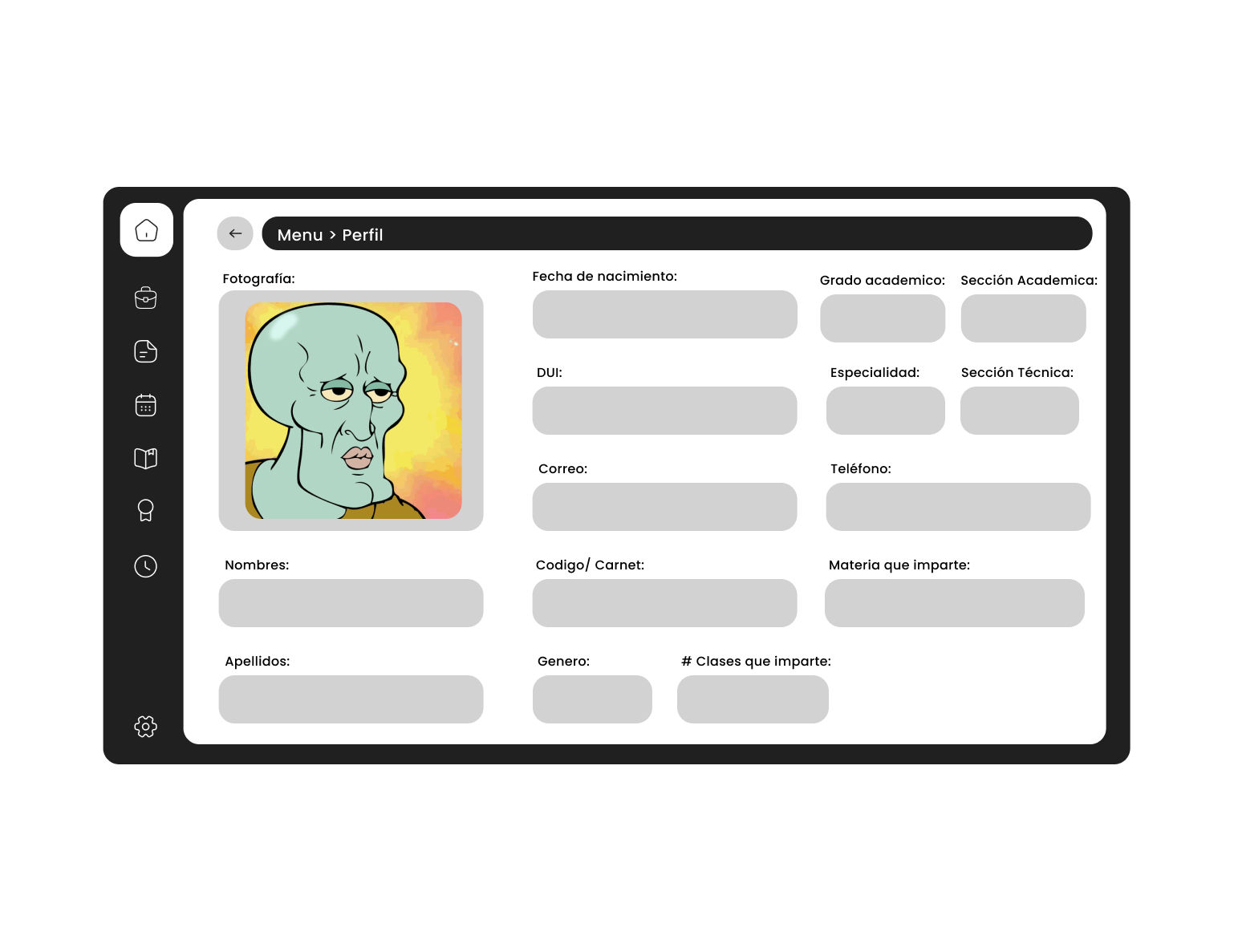
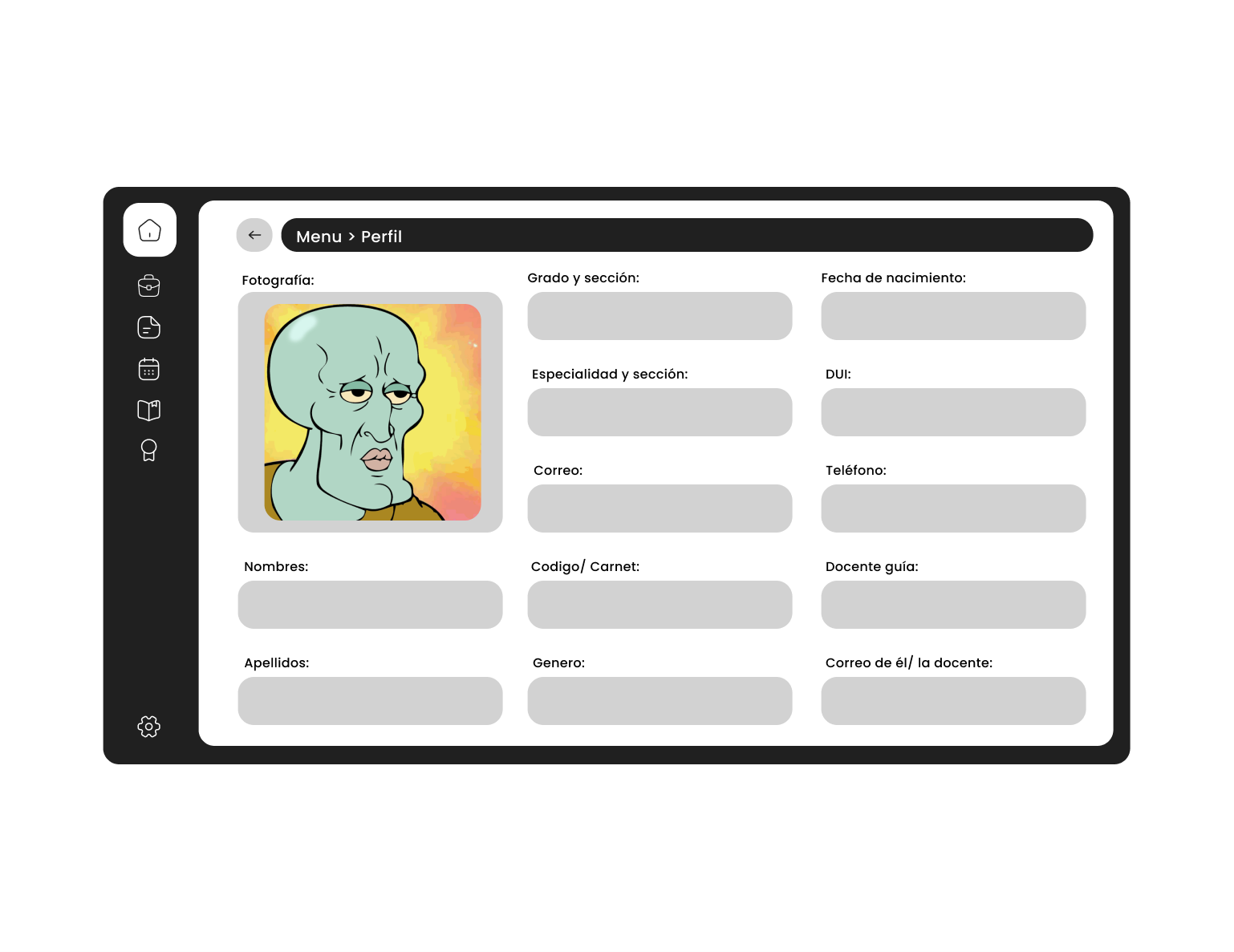
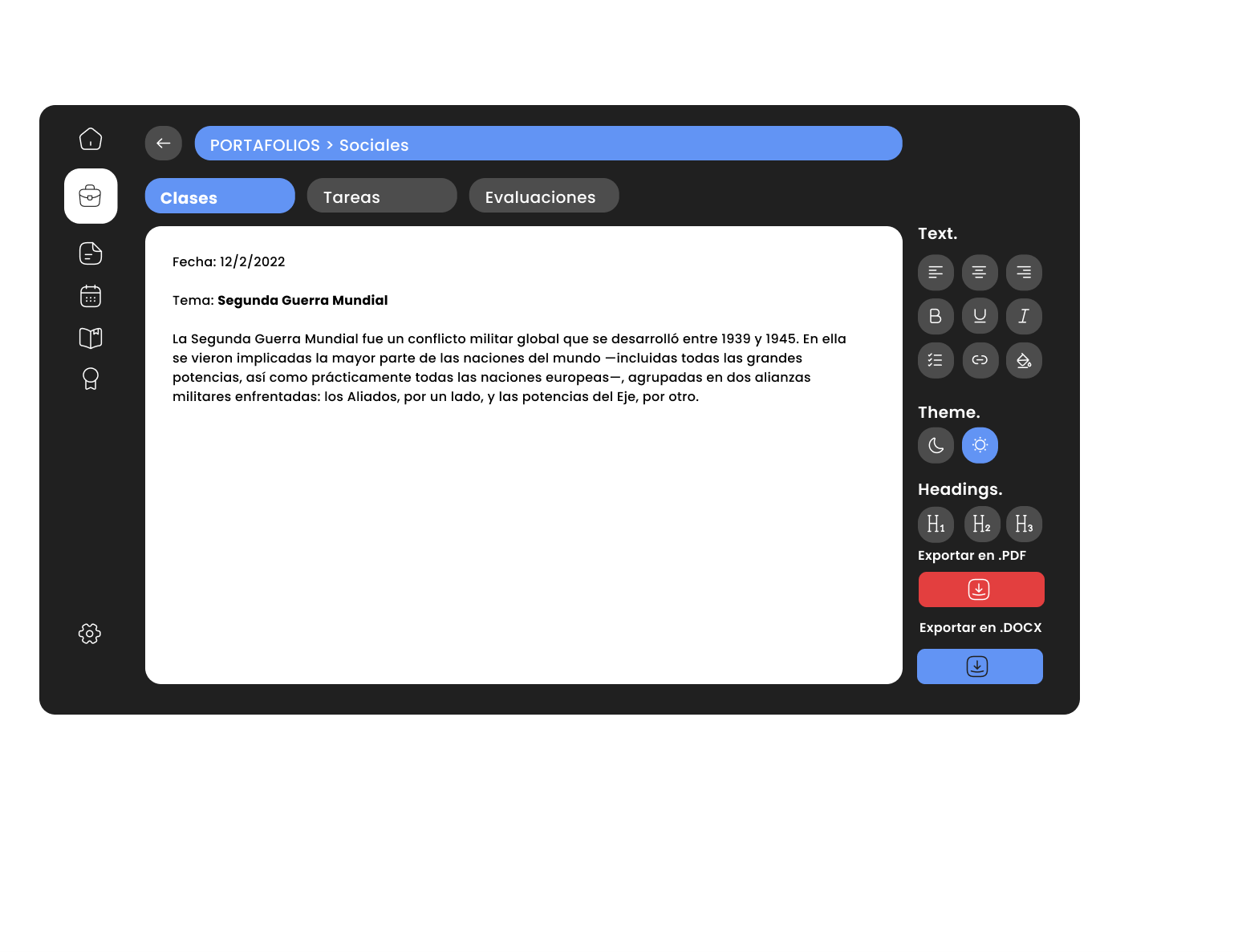
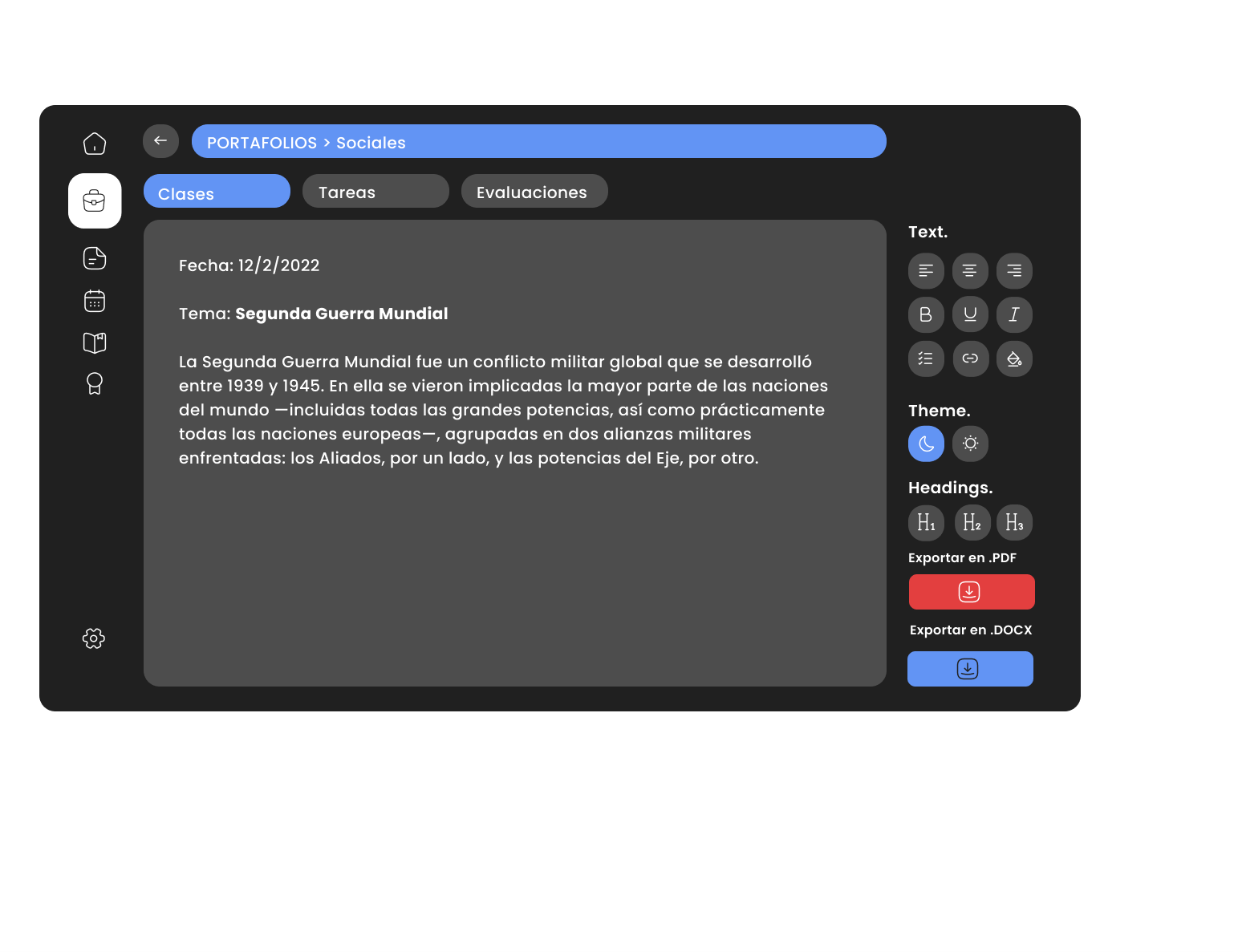
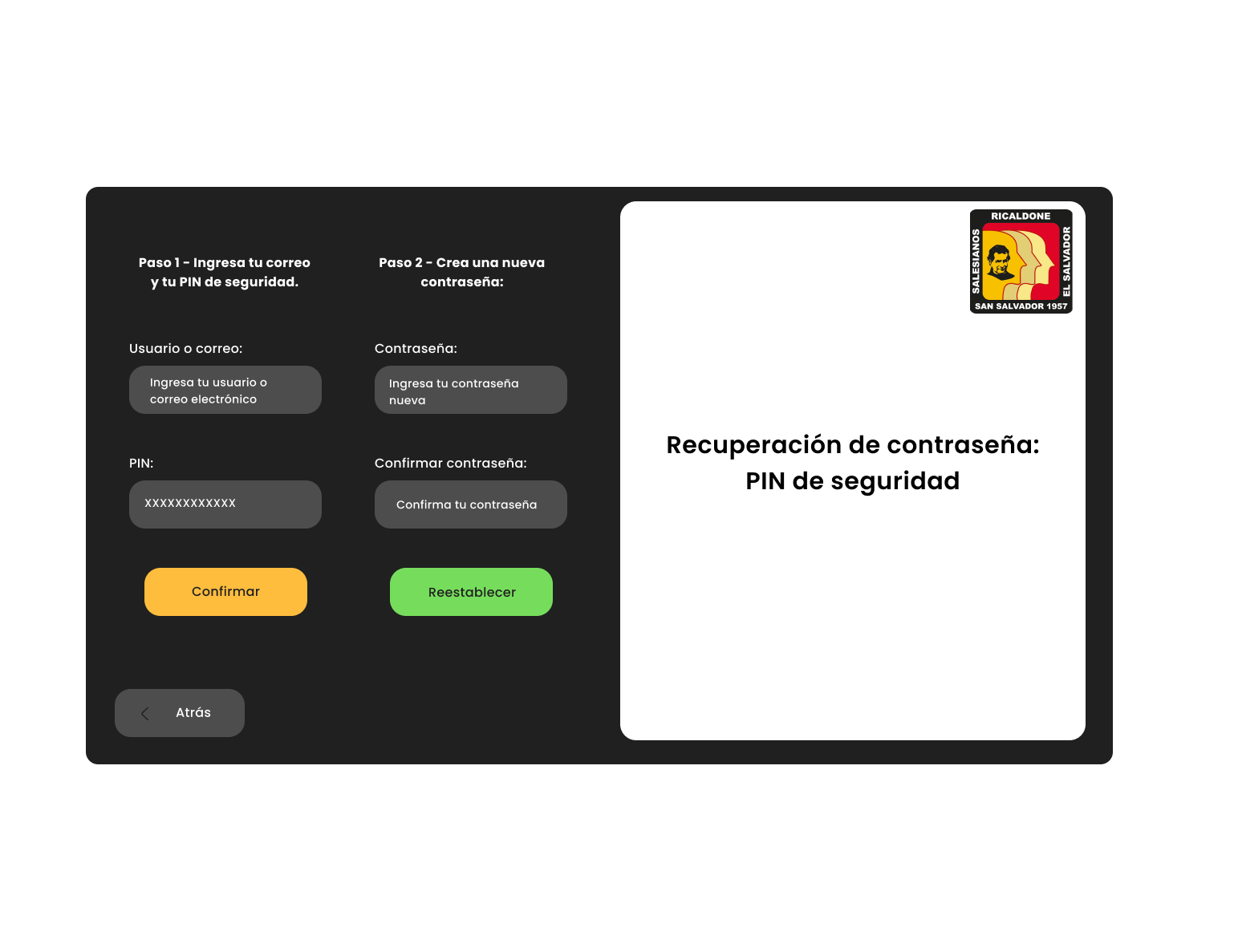
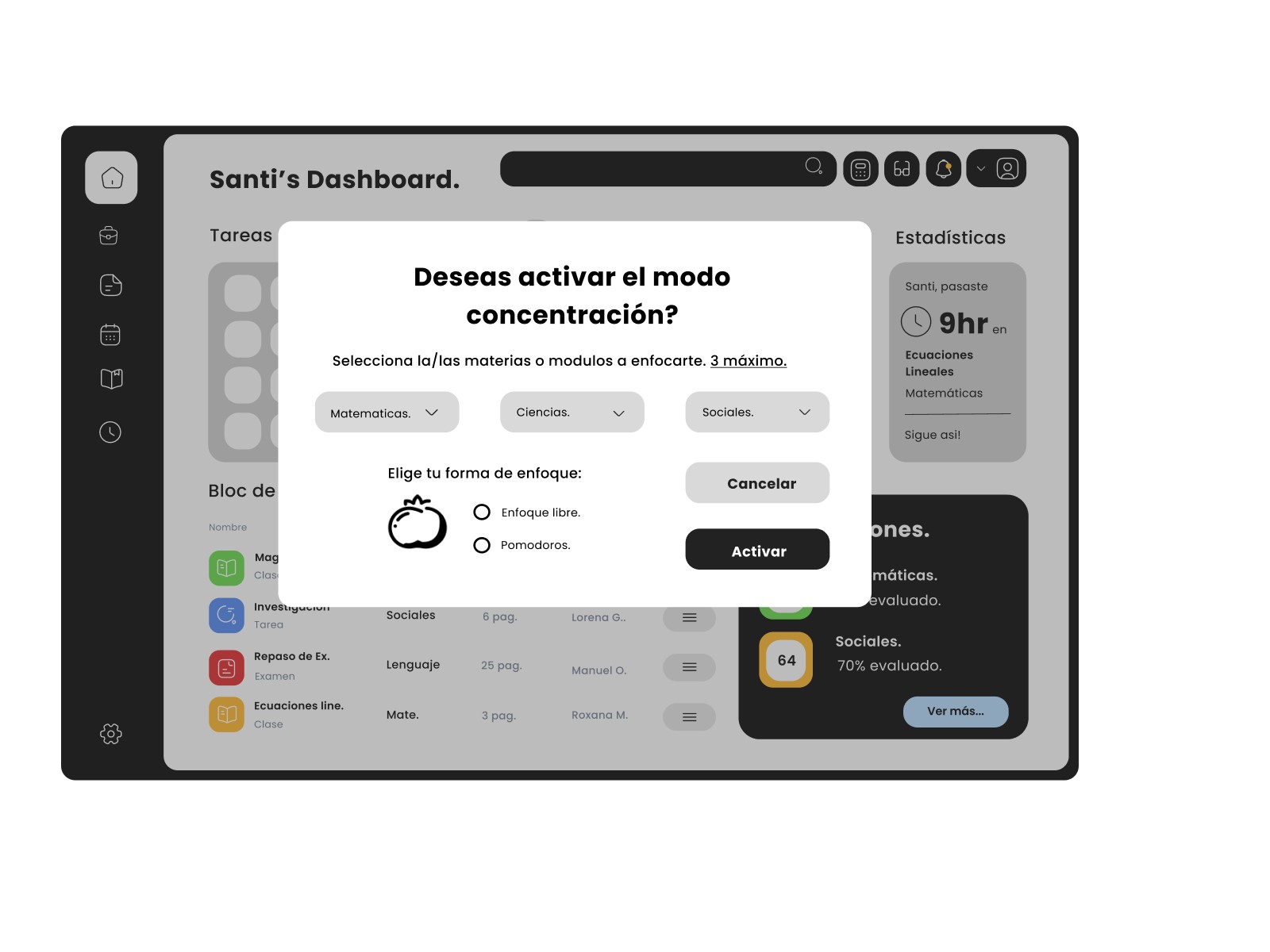
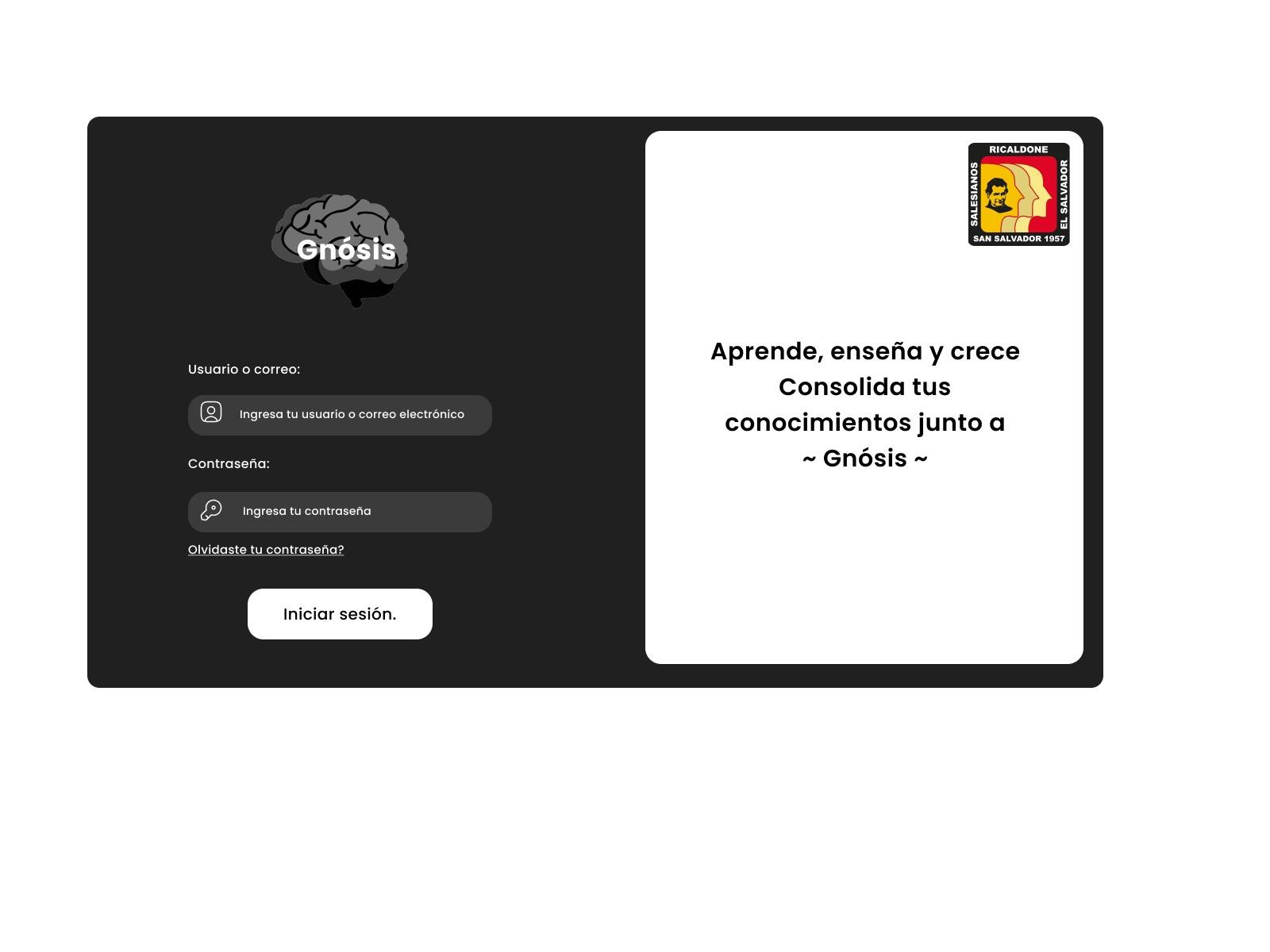
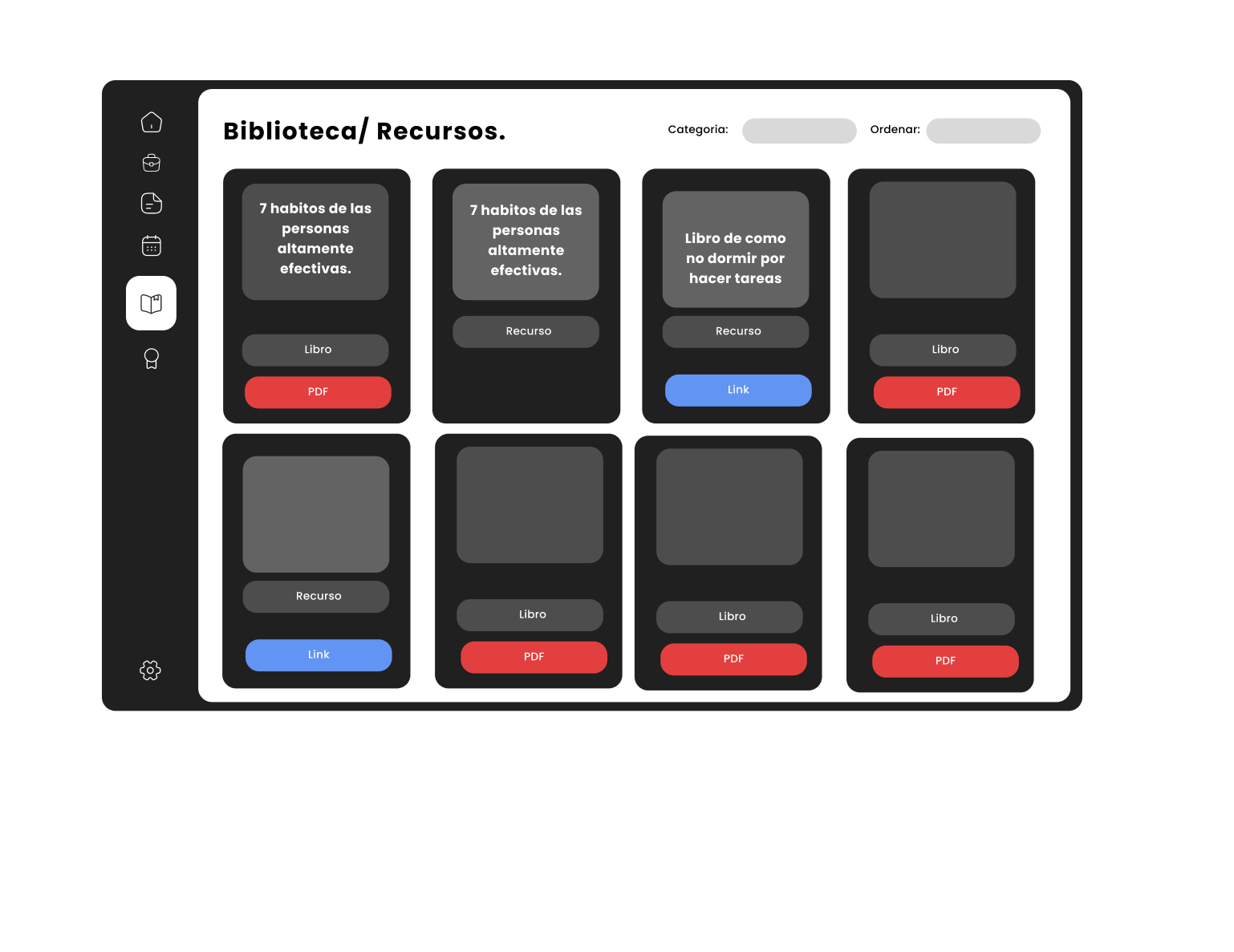
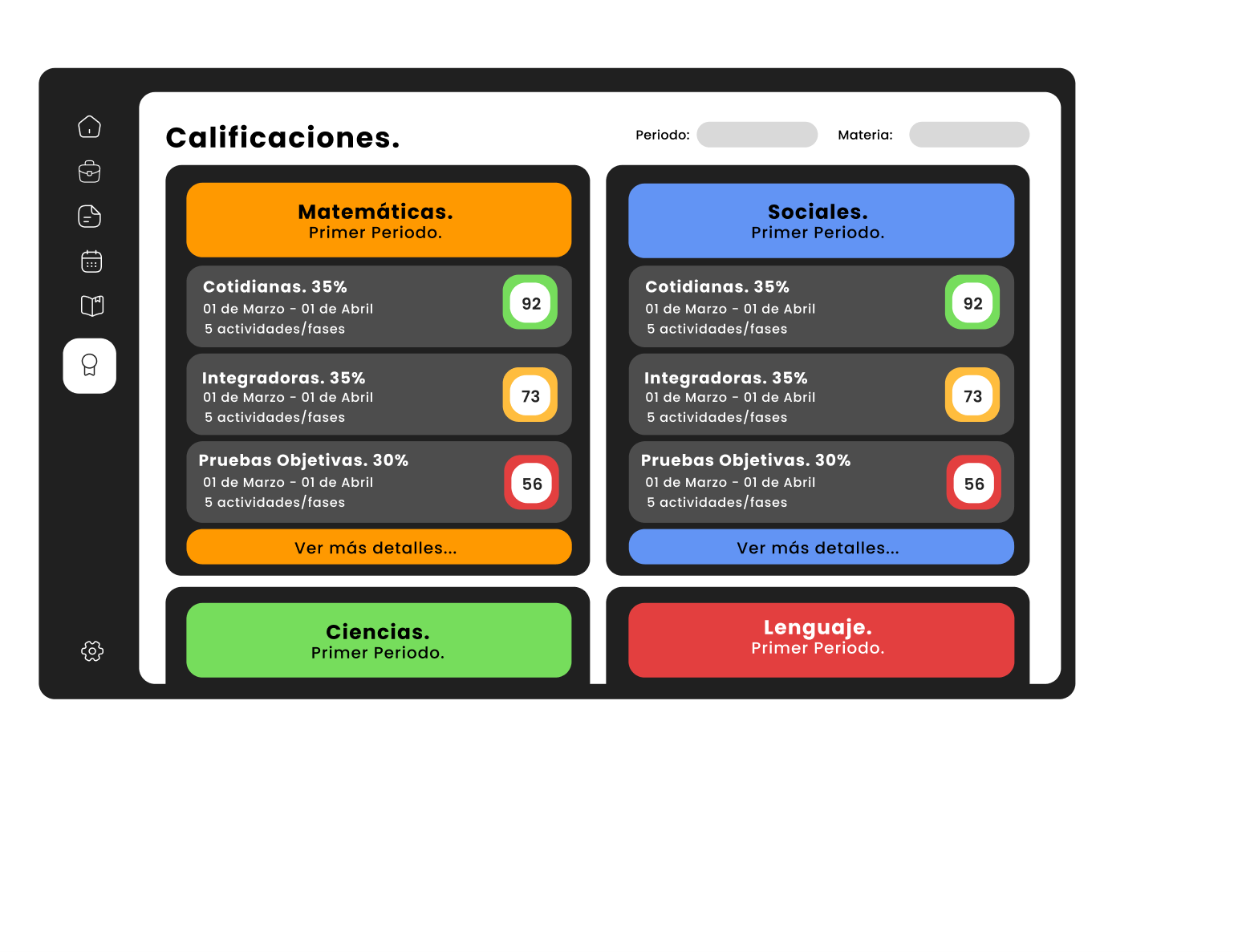
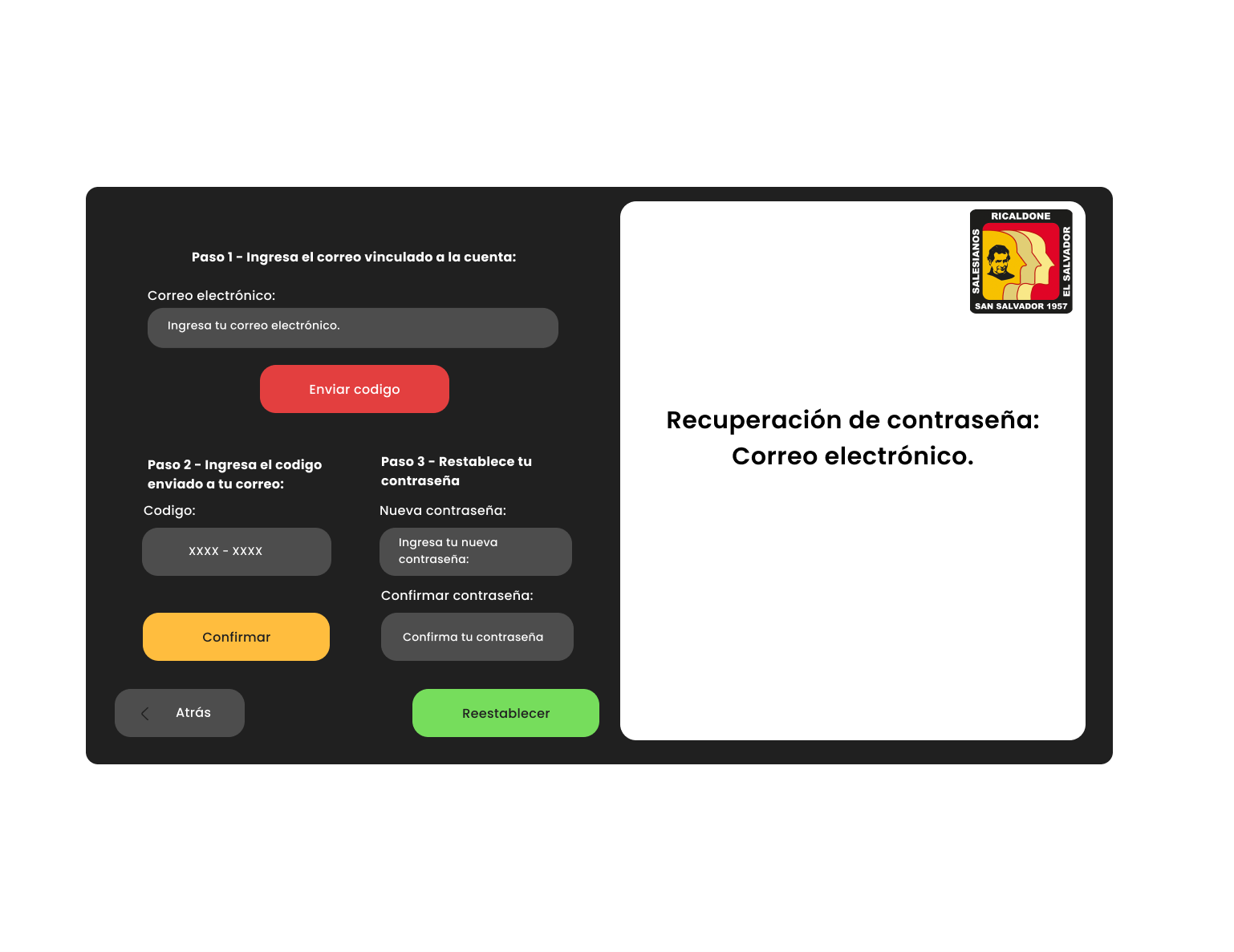
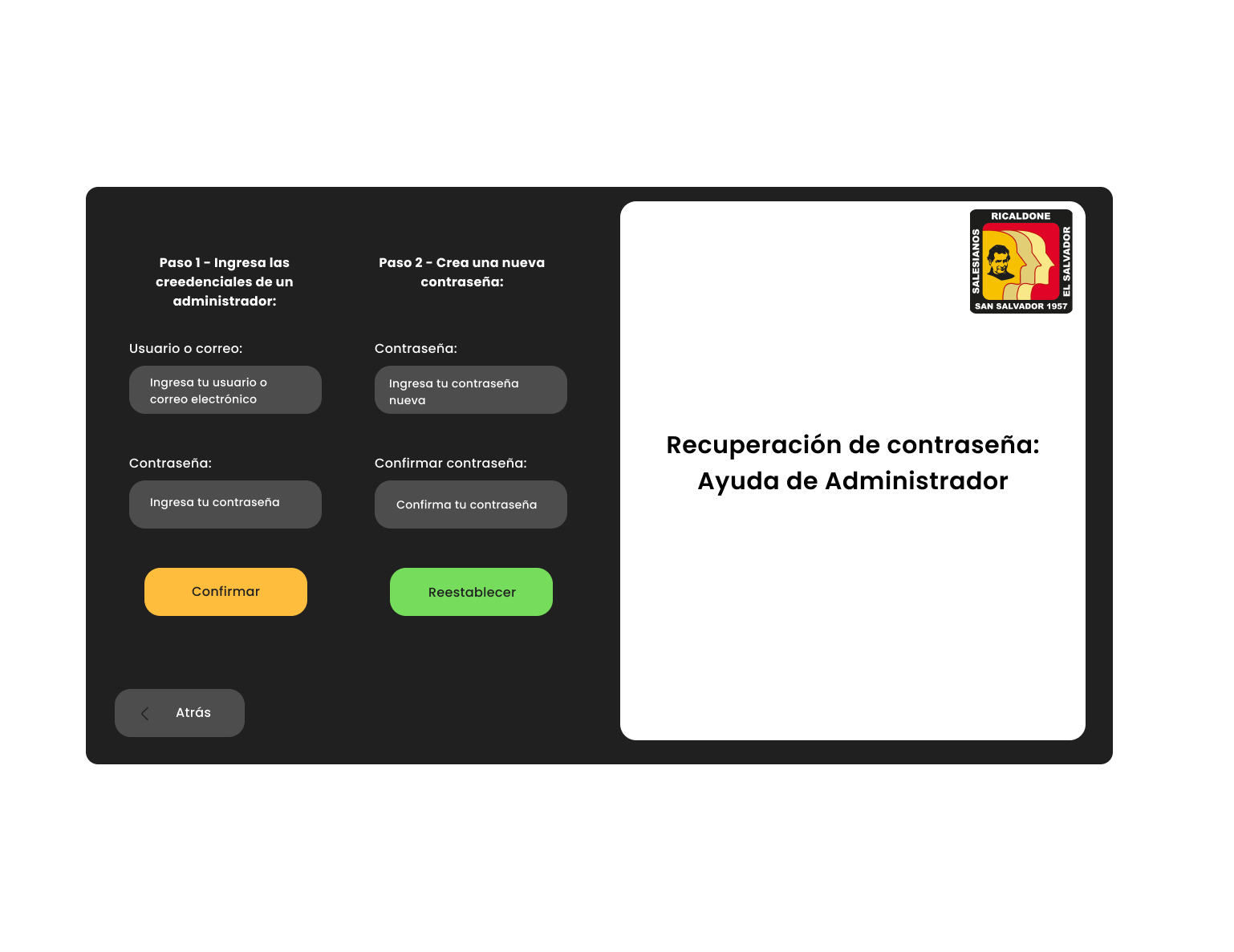
# Estandarizar la codificación de errores.

|  |  |  |  |
| --- | --- | --- | --- |
| # de Error | Código de Error | Error | Posibles soluciones |
| 1 | GN001 | Error de conexión con la base de datos | -Verificar conexión de internet    -Verificar los datos de conexión con la base de datos en el sistema |
| 2 | GN002 | Error en validación de credenciales | -Verificar las credenciales ingresadas en el sistema    -Verificar si ya hay una cuenta existente con esas credenciales |
| 3 | GN003 | Error al cargar los datos en el sistema | -Volver a iniciar sesión en el sistema |
| 4 | GN004 | Error al cargar los combobox | -volver a iniciar sesión en el sistema    -Recargar el formulario |
| 5 | GN005 | Error al cargar las notas del estudiante | -Volver a iniciar  sesión    -Reiniciar el sistema |
| 6 | GN006 | Error al cargar el perfil | -Volver a iniciar sesión |
| 7 | GN007 | Erro al recuperar la contraseña con administrador | -Verificar las credenciales a recuperar    -Verificar las credenciales del administrador |
| 8 | GN008 | Error al recuperar la contraseña por correo electrónico | -Verificar el correo electrónico    -Verificar el código enviado al correo electrónico |
| 9 | GN009 | Error al activar el  modo  concentración | -Verificar si se seleccionaron materias |
| 10 | GN010 | Error al cargar las estadísticas | -Reiniciar el sistema  -Volver a iniciar sesión |

# Definición de estándares:



# Diseño de interfaces.



**Script de base de datos.** create database dbGnosis; go

use dbGnosis; go

/\*Perfiles\*/

create table tbPeriodosTrimestres( idPeriodoTrimestre int identity(1,1) not null primary key,

PeriodoTrimestre varchar(40) not null

);

create table tbEstadosPerfiles( idEstadoPerfil int identity(1,1) not null primary key,

EstadoPerfil varchar(40) not null

);

create table tbTiposPerfiles( idTipoPerfil int identity(1,1) not null primary key,

TipoPerfil varchar(40) not null

);

create table tbFechas( idFecha int identity(1,1) not null primary key,

FechaInicio date not null,

FechaVencimiento date not null

);

create table tbPerfiles(

idPerfil int identity(1,1) not null primary key,

NombrePerfil varchar(30) not null,

Contendio varchar(100) not null, Porcentaje numeric(8,2) not null, idEstadoPerfil int not null, idTipoPerfil int not null, idPeriodoTrimestre int not null, idModuloMateriaDocente int not null, idFecha int not null

);

create table tbModulosMateriasDocente( idModuloMateriaDocente int identity(1,1) not null primary key,

idModulo int null, idDocente int not null, idMateria int null

);

create table tbRegistroNotas( idRegistroNota int identity(1,1) not null primary key,

Nota numeric(8,2) not null, PromedioFinal numeric(8,2) not null, idPerfil int not null

);

create table tbEntregas( idEntega int identity(1,1) not null primary key,

Archivo varbinary not null,

LinkDeEntrega varchar(150) null,

idPerfil int not null

);

alter table tbEntregas add constraint FK\_perfiles\_entregas foreign key (idPerfil) references tbPerfiles(idPerfil);

alter table tbPerfiles add constraint FK\_perfiles\_estado foreign key (idEstadoPerfil) references tbEstadosPerfiles(idEstadoPerfil);

alter table tbPerfiles add constraint FK\_perfiles\_tipo foreign key (idTipoPerfil) references tbTiposPerfiles(idTipoPerfil);

alter table tbPerfiles add constraint FK\_perfiles\_periodotrimestre foreign key (idPeriodoTrimestre) references tbPeriodosTrimestres(idPeriodoTrimestre);

alter table tbPerfiles add constraint FK\_perfiles\_modulomaterias foreign key (idModuloMateriaDocente) references tbModulosMateriasDocente(idModuloMateriaDocente);

alter table tbPerfiles add constraint FK\_perfiles\_fechas foreign key (idFecha) references tbFechas(idFecha);

alter table tbRegistroNotas add constraint FK\_registro\_notas foreign key (idPerfil) references tbPerfiles(idPerfil);

/\*Modulos-Materias-Docentes-enfonque\*/ create table tbModulos( idModulo int identity(1,1) not null primary key,

Modulo varchar(30) not null

);

create table tbMaterias( idMateria int identity(1,1) not null primary key,

Materia varchar(30)not null

);

create table tbEnfoques( idEnfoque int identity(1,1) not null primary key,

NombreEnfoque varchar(40) not null, TiempoEnfoque time not null, idMateria int not null, idEstadoEnfoque int not null

);

create table tbEstadosEnfoques( idEstadoEnfoque int identity(1,1) not null primary key,

EstadoEnfoque varchar(30) not null

);

create table tbEstadisticasEnfoques( idEstadistica int identity(1,1) not null primary key,

TiempoTotal time,

idEnfoque int not null

);

create table tbDocentes( idDocente int identity(1,1) not null primary key, NombresDocente varchar(50) not null,

ApellidosDocente varchar(50) not null,

DireccionDocente varchar(150) not null,

TelefonoDocente varchar(9) not null,

FechaNacimiento date not null,

DUI varchar(9) not null, Correo varchar(80) not null, idGeneroDocente int not null, idUsuario int not null

);

create table tbGenerosDocentes( idGeneroDocente int identity(1,1) not null primary key,

GeneroDocente varchar(30) not null

);

alter table tbModulosMateriasDocente add constraint FK\_modulomateria\_modulo foreign key (idModulo) references tbModulos(idModulo);

alter table tbModulosMateriasDocente add constraint FK\_modulomateria\_docentes foreign key (idDocente) references tbDocentes(idDocente);

alter table tbModulosMateriasDocente add constraint FK\_modulomateria\_materia foreign key (idMateria) references tbMaterias(idMateria);

alter table tbEnfoques add constraint FK\_enfoque\_materia foreign key (idMateria) references tbMaterias(idMateria);

alter table tbEnfoques add constraint FK\_enfoque\_estado foreign key (idEstadoEnfoque) references tbEstadosEnfoques(idEstadoEnfoque);

alter table tbEstadisticasEnfoques add constraint FK\_estadistica\_estado foreign key (idEnfoque) references tbEnfoques(idEnfoque);

alter table tbDocentes add constraint FK\_genero\_docente foreign key (idGeneroDocente) references tbGenerosDocentes(idGeneroDocente); create table TiposUsuarios( idTipoUsuario int identity(1,1) not null primary key,

TipoUsuario varchar(40) not null

);

create table EstadosUsuarios( idEstadoUsuario int identity(1,1) not null primary key,

EstadoUsuario varchar(40) not null

);

create table tbUsuarios( idUsuario int identity(1,1) not null primary key,

Username varchar(30) not null, Contraseña varchar(50) not null, idTipoUsuario int not null, idEstadoUsuario int not null

);

alter table tbDocentes add constraint FK\_usuario\_docente foreign key (idUsuario) references tbUsuarios(idUsuario);

alter table tbUsuarios add constraint FK\_usuario\_tipo foreign key (idTipoUsuario) references TiposUsuarios(idTipoUsuario);

alter table tbUsuarios add constraint FK\_usuario\_estado foreign key (idEstadoUsuario) references EstadosUsuarios(idEstadoUsuario);

/\*Alumnos\*/

create table tbAlumnos( idAlumno int identity(1,1) not null primary key, NombresAlumno varchar(50) not null,

ApellidosAlumno varchar(50) not null,

DireccionAlumnos varchar(150) not null,

TelefonoAlumno varchar(9) not null,

FechaNacimiento date not null,

DUIAlumno varchar(9) null,

CorreoAlumno varchar(80) not null, CodigoCarnet varchar(8) not null, idGeneroAlumno int not null, idEstadoAlumno int not null, idConjuntoNiveles int not null, idUsuario int not null

);

alter table tbAlumnos add constraint FK\_usuario\_alumnos foreign key (idUsuario) references tbUsuarios(idUsuario);

create table tbGenerosAlumnos( idGeneroAlumno int identity(1,1) not null primary key,

GeneroAlumno varchar(30) not null

);

create table tbEstadosAlumnos( idEstadoAlumno int identity(1,1) not null primary key,

EstadoAlumno varchar(30) not null

);

alter table tbAlumnos add constraint FK\_estado\_alumnos foreign key (idEstadoAlumno) references tbEstadosAlumnos(idEstadoAlumno);

alter table tbAlumnos add constraint FK\_genero\_alumnos foreign key (idGeneroAlumno) references tbGenerosAlumnos(idGeneroAlumno);

/\*Conjunto Niveles\*/

create table tbGrupos( idGrupo int identity(1,1) not null primary key,

Grupo varchar(10) not null

);

create table tbSeccionesTecnicas( idSeccionTecnica int identity(1,1) not null primary key,

SeccionTecnica varchar(10) not null

);

create table tbEspecialidades( idEspecialidad int not null identity(1,1) primary key, Especialidad varchar(40) not null

);

create table tbGrados( idGrado int not null identity(1,1) primary key,

Grado varchar(30)not null

);

create table tbSeccionesAcademicas( idSeccionAcademica int not null identity(1,1) primary key,

SeccionAcademica varchar(30)not null

);

create table tbConjuntoNiveles( idConjuntoNiveles int identity (1,1) not null primary key,

idGrupo int not null, idSeccionTecnica int null, idEspecialidad int null, idGrado int null, idSeccionAcademica int null

);

alter table tbAlumnos add constraint FK\_conjunto\_alumnos foreign key (idConjuntoNiveles) references tbConjuntoNiveles(idConjuntoNiveles);

alter table tbConjuntoNiveles add constraint FK\_conjunto\_grupos foreign key (idGrupo)

references tbGrupos(idGrupo);

alter table tbConjuntoNiveles add constraint FK\_tecnica\_grupos foreign key (idSeccionTecnica) references tbSeccionesTecnicas(idSeccionTecnica);

alter table tbConjuntoNiveles add constraint FK\_academica\_conjunto foreign key (idSeccionAcademica) references tbSeccionesAcademicas(idSeccionAcademica);

alter table tbConjuntoNiveles add constraint FK\_especialidad\_conjunto foreign key (idEspecialidad) references tbEspecialidades(idEspecialidad);

alter table tbConjuntoNiveles add constraint FK\_conjunto\_grado foreign key (idGrado) references tbGrados(idGrado);

/\*Eventos\*/

create table tbEventos( idEvento int identity(1,1) not null primary key,

NombreEvento varchar(30) not null,

FechaEvento date not null, HoraEvento time null, idAlumno int not null, idTipoEvento int not null, idEstadoEvento int not null,

);

create table tbTipoEventos( idTipoEvento int identity(1,1) not null primary key,

TipoEvento varchar(30) not null

);

create table tbEstadoEventos( idEstadoEvento int identity(1,1) not null primary key,

EstadoEvento varchar(30) not null

);

alter table tbEventos add constraint FK\_evento\_alumnos foreign key (idAlumno) references tbAlumnos(idAlumno);

alter table tbEventos add constraint FK\_evento\_tipos foreign key (idTipoEvento) references tbTipoEventos(idTipoEvento);

alter table tbEventos add constraint FK\_evento\_estado foreign key (idEstadoEvento) references tbEstadoEventos(idEstadoEvento);

/\*carpeta bloc\*/

create table tbCarpetaMaterias( idCarpetaMateria int identity(1,1) not null primary key, NombreCarpeta varchar(40) not null, idModuloMateriaDocente int not null, idAlumno int not null

);

create table tbBlocs( idBloc int identity(1,1) not null primary key, NombreBloc varchar(30) not null, idCarpetaMateria int not null, idCategoriaBloc int not null

);

create table tbCatergoriaBlocs( idCategoriaBloc int identity(1,1) not null primary key,

CategoriaBloc varchar(30) not null,

);

create table tbContenidos( idContenidoBloc int identity(1,1) not null primary key, Contenido varchar(1000) not null, idBloc int not null

);

alter table tbCarpetaMaterias add constraint FK\_carpeta\_modulosdocente foreign key (idModuloMateriaDocente) references tbModulosMateriasDocente(idModuloMateriaDocente); alter table tbBlocs add constraint FK\_carpeta\_bloc foreign key (idCarpetaMateria) references tbCarpetaMaterias(idCarpetaMateria);

alter table tbBlocs add constraint FK\_bloc\_categoria foreign key (idCategoriaBloc) references tbCatergoriaBlocs(idCategoriaBloc);

alter table tbContenidos add constraint FK\_bloc\_contenido foreign key (idBloc) references tbBlocs(idBloc);

/\*Biblioteca\*/

create table tbBibliotecas( idBiblioteca int identity(1,1) not null primary key, NombreBiblioteca varchar(30) not null, idAlumno int not null

);

create table tbContenidoBibliotecas( idContenidoBiblioteca int identity(1,1) not null primary key,

NombrePDF varchar(30) not null, LinkPDF varchar(50) not null, idBiblioteca int not null

);

alter table tbBibliotecas add constraint FK\_biblioteca\_alumno foreign key (idAlumno) references tbAlumnos(idAlumno);

alter table tbContenidoBibliotecas add constraint FK\_biblioteca\_contenido foreign key (idBiblioteca)

references tbBibliotecas(idBiblioteca);

# Conclusiones.

Al completar la fase de análisis y diseño del proyecto se concluye en lo indispensable y útil que pueden llegar a hacer los procesos de diagramación, para la comprensión del sistema, y la realización de este, mientras provee una vista panorámica acerca del uso del sistema en cuestión, sus alcances y limitantes. Permitiendo de esta manera visualizar los datos y la información a manejar de una manera más amplia, comprensible y ajustable a las problemáticas que puedan llegar a presentarse durante el proceso de ejecución del sistema.

Mostrando que diseñar interfaces en un medio ajeno al de programación permite visualizar el producto final o lo que se quiere llegar a lograr, facilitando la etapa de ejecución del sistema, siendo estos diseños los puntos de referencia perfectos para asegurar la presentación del sistema en su etapa final. De esta forma por medio de la investigación de tendencias en los recursos gráficos, formas de implementación en Java, librerías, son la forma de alcanzar el objetivo deseado.

Por lo que la realización previa de la base de datos es de vital importancia, para llegar a comprender el sistema y todos los datos que este manejará, y como lo hará de manera que el usuario este cómodo, estandarizando y controlando los errores y regulaciones que el sistema contendrá en sí mismo.

# Fuentes de información.

1. MikeRayMSFT. (2022, March 22). *Data types (Transact-SQL) - SQL Server*. Microsoft.com. [https://docs.microsoft.com/en-us/sql/t-sql/data-types/data-types-](https://docs.microsoft.com/en-us/sql/t-sql/data-types/data-types-transact-sql?view=sql-server-ver16)

[transact-sql?view=sql-server-ver16](https://docs.microsoft.com/en-us/sql/t-sql/data-types/data-types-transact-sql?view=sql-server-ver16)

1. *Guía para la redacción de casos de uso | Marco de Desarrollo de la Junta de*

*Andalucía*. (2022). Juntadeandalucia.es.

<https://www.juntadeandalucia.es/servicios/madeja/contenido/recurso/416>

1. *Capítulo 9 -Apéndice 9.1 Definición de Casos de uso 9.1.1 Qué es un caso de uso*. (n.d.). [http://sedici.unlp.edu.ar/bitstream/handle/10915/3956/9\_-](http://sedici.unlp.edu.ar/bitstream/handle/10915/3956/9_-_Ap%C3%A9ndice.pdf?sequence=13)

[\_Ap%C3%A9ndice.pdf?sequence=13](http://sedici.unlp.edu.ar/bitstream/handle/10915/3956/9_-_Ap%C3%A9ndice.pdf?sequence=13)