

PROYECTO INTEGRAL



JUAN CARLOS MEJÍAS DOÑORO

2.º CFGS DAM

PROFESORES RESPONSABLES:

SORAYA PECEÑO CAPILLA

ISABEL GEMA MANCHADO DE SOLA

RAFAEL GALLARDO GIMÉNEZ

ÁLVARO GONZÁLEZ JIMÉNEZ

ÍNDICE

1-OBJETIVO.	3
2-ALCANCE DEL PROYECTO.	3
2.1-FUNCIONALIDADES.	3
3-IDENTIFICACIÓN DE REQUISITOS DEL SISTEMA.	4
3.1-IDENTIFICACIÓN DE DATOS.	4
3.2-IDENTIFICACIÓN REQUISITOS FUNCIONALES.	4
3.3-IDENTIFICACIÓN DE ROLES.	5
3.4-IDENTIFICACIÓN REQUISITOS NO FUNCIONALES.	6
4-IDENTIFICACIÓN DEL ENTORNO TECNOLÓGICO.	6
4.1-LENGUAJES DE PROGRAMACIÓN Y FRAMEWORKS.	6
4.2-INFRAESTRUCTURA Y DESPLIEGUE.	6
4.3-HERRAMIENTAS DE DESARROLLO Y CONTROL DE VERSIONES.	6
5-DEFINICIÓN DE CASOS DE USOS.	7
5.1-DIAGRAMA DE CASOS DE USOS.	7
5.2-ESPECIFICACIÓN DE CADA CASO DE USO.	9
CU001 - LOGIN.	9
CU002 - REGISTRO DE AUSENCIA.	10
CU003 - MODIFICACIÓN DE AUSENCIA.	12
CU004 - ELIMINACIÓN DE AUSENCIA.	14
CU005 - CONSULTA DE PROFESORADO AUSENTE CON FILTRADO.	16
CU006 - ASIGNACIÓN DE GUARDIAS.	17
CU007 - CONSULTA DE LA GUARDIA PARA UN DÍA.	18
CU008 - LISTADO DE LA INFORMACIÓN HISTÓRICA DE AUSENCIAS.	19
CU009 - LISTADO DE LA INFORMACIÓN HISTÓRICA DE GUARDIAS.	20
6-DEFINICIÓN DE PLAN DE PRUEBAS.	21
6.1-INTRODUCCIÓN.	21
6.2-ESTRUCTURA Y ORGANIZACIÓN.	21
6.3-COBERTURA DE PRUEBAS.	22
6.4-IDENTIFICACIÓN DE CASOS DE PRUEBA.	22
6.5-COBERTURA DE REQUISITOS.	23
6.6-ESTRATEGIA Y HERRAMIENTAS.	23
6.7-CRONOGRAMA.	23

6.8-DOCUMENTACIÓN DE RESULTADOS.	23
6.9-VALIDACIÓN Y ACEPTACIÓN.	24
6.10-ANÁLISIS DE RIESGOS.	24
7-DEFINICIÓN DE INTERFACES GRÁFICAS.	25
7.1-NAVEGACIÓN.	25
7.2-ESTILOS BASE.	25
7.3-WIREFRAME DIGITAL (FIGMA).	25
7.4-ENLACE A MI PROTOTIPO.	25
8-SERVICIOS INTERNOS.	26
8.1-SERVICIOS DE AUSENCIAS.	26
Servicio: Registrar Ausencia	26
Servicio: Modificar Ausencia	27
Servicio: Eliminar Ausencia	28
Servicio: Consultar Ausencias	29
Servicio: Obtener Ausencia Por Id	30
Servicio: Obtener Ausencia Por Id De Profesor	31
8.2-SERVICIOS DE GUARDIAS.	32
Servicio: Asignar Guardia	32
Servicio: Consultar Guardias	33
Servicio: Obtener Guardia Por Id	34
9-MANUAL DE USUARIO.	34

1-OBJETIVO.

Desarrollar un sistema integral para la gestión digitalizada de ausencias del profesorado y asignación de guardias en el centro educativo por parte de los profesores y la jefatura.

2-ALCANCE DEL PROYECTO.

El sistema cubrirá los siguientes procesos:

1. Gestión de ausencias.
 - Registro de ausencias por parte del profesorado.
 - Modificación/eliminación de ausencias.
2. Asignación de guardias.
 - Asignación de guardias por jefatura.
 - Notificaciones automáticas a profesores asignados.
3. Consultas e informes.
 - Listado de ausencias/guardias diarias.
 - Históricos filtrables por:
 - Rango de fechas.
 - Departamento académico.
 - Tipo de ausencia.
 - Exportación de datos (Excel/PDF).

2.1-FUNCIONALIDADES.

AUTENTICACIÓN Y ACCESO.

- Sistema de logueo para los usuarios del sistema.
- Diferenciar roles.

GESTIÓN DE AUSENCIAS.

- Registro de ausencias.
- Modificación de ausencias.
- Eliminación de ausencias.
- Consultar el profesorado que está ausente.

ASIGNACIÓN DE GUARDIAS.

- Visualizar el profesorado ausente con el rol de jefatura.
- Asignar las guardias al profesorado disponible con el rol de jefatura.

CONSULTA DE LAS GUARDIAS Y AUSENCIAS.

- Consultar la información de las guardias de un día concreto por el rol de jefatura.
- Listar la información histórica de ausencias y guardias por el rol de jefatura.

3-IDENTIFICACIÓN DE REQUISITOS DEL SISTEMA.

3.1-IDENTIFICACIÓN DE DATOS.

USUARIO.

- Todos los usuarios del sistema con sus datos básicos y rol asignado.
 - id, nick, nombre, apellidos, email, password, rol, departamento.

AUSENCIA.

- Registro de cada ausencia con sus fechas, motivo y tareas para alumnos.
 - id, profesor_id, zona_id, fecha, hora_inicio, hora_fin, motivo, estado, tareas.

GUARDIA.

- Guardias asignadas para cubrir ausencias específicas.
 - id, ausencia_id, profesor_asignado, aula_id, fecha, hora_inicio, hora_fin.

3.2-IDENTIFICACIÓN REQUISITOS FUNCIONALES.

RF001 - AUTENTICACIÓN DE USUARIOS.

- El sistema debe permitir el acceso mediante credenciales independientemente del rol que se trate.

RF002 - REGISTRO DE AUSENCIAS.

- El rol de profesor puede registrar sus propias ausencias.

RF003 - MODIFICACIÓN DE AUSENCIAS.

- El rol de profesor puede modificar sus propias ausencias.

RF004 - ELIMINACIÓN DE AUSENCIAS.

- El rol de profesor puede eliminar sus propias ausencias.

RF005 - CONSULTAR PROFESORADO AUSENTE.

- Los roles de profesor o jefatura pueden consultar los profesores que están ausentes.

RF006 - ASIGNACIÓN DE GUARDIAS.

- El rol de jefatura puede asignar guardias a profesores disponibles.

RF007 - CONSULTA DE LA GUARDIA PARA UN DÍA.

- Los roles profesor y jefatura pueden consultar los profesores que tienen guardia en un día concreto.

RF008 - LISTADO DE LA INFORMACIÓN HISTÓRICA DE AUSENCIAS.

- El rol de jefatura puede consultar el historial de ausencias.

RF009 - LISTADO DE LA INFORMACIÓN HISTÓRICA DE GUARDIAS.

- El rol de jefatura puede consultar el historial de guardias.

3.3-IDENTIFICACIÓN DE ROLES.

ROL001 - PROFESOR.

- Puede registrar, modificar y eliminar sus ausencias. Además, puede consultar los profesores que están ausentes y consultar las guardias de un día concreto.

ROL002 - JEFATURA.

- Puede consultar el listado de profesores ausentes, asignar guardias, consultar las guardias de un día concreto y consultar el historial de ausencias y guardias.

3.4-IDENTIFICACIÓN REQUISITOS NO FUNCIONALES.

RNF001 - SEGURIDAD.

- Los profesores no deben acceder a las secciones del rol de jefatura.

RNF002 - USABILIDAD.

- La interfaz debe ser intuitiva y accesible para todos los usuarios.

RNF003 - COMPATIBILIDAD.

- Debe ser accesible desde distintos dispositivos y navegadores.

RNF004 - RENDIMIENTO.

- El sistema debe responder en <2 segundos el 95% de las consultas.

4-IDENTIFICACIÓN DEL ENTORNO TECNOLÓGICO.

4.1-LENGUAJES DE PROGRAMACIÓN Y FRAMEWORKS.

- Frontend: Angular.
- Backend: Java con Spring Boot y JPA.
- Base de Datos: MySQL.

4.2-INFRAESTRUCTURA Y DESPLIEGUE.

- Despliegue: Spring Tools Suite.
- Autenticación: JWT.
- Acceso a Datos: JDBC y JPA.

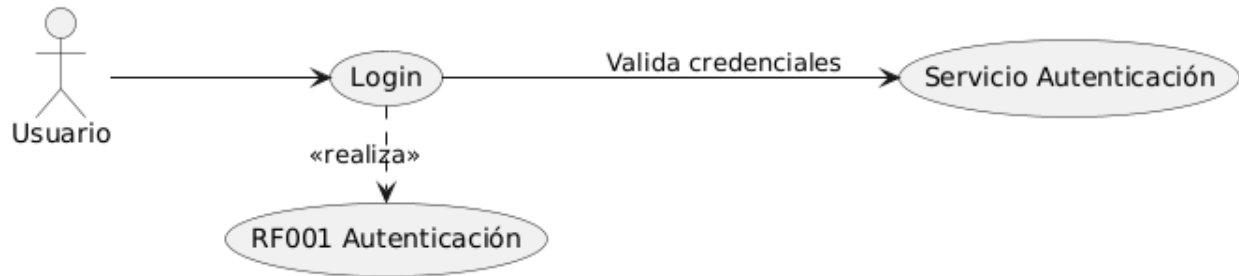
4.3-HERRAMIENTAS DE DESARROLLO Y CONTROL DE VERSIONES.

- Repositorio: GitHub.
- Testing: JUnit.
- IDE: Visual Studio Code y Spring Tools Suite.
- Prototipado: Figma.

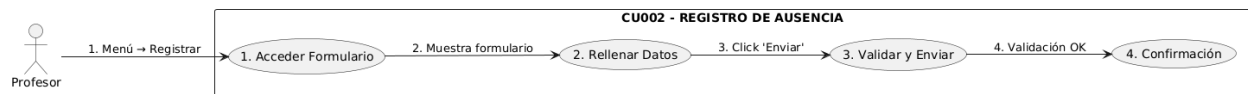
5-DEFINICIÓN DE CASOS DE USOS.

5.1-DIAGRAMA DE CASOS DE USOS.

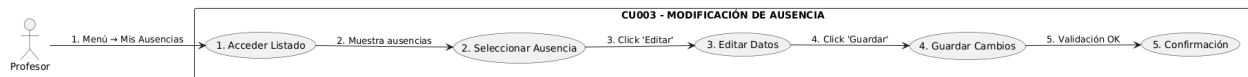
CU001 - LOGIN.



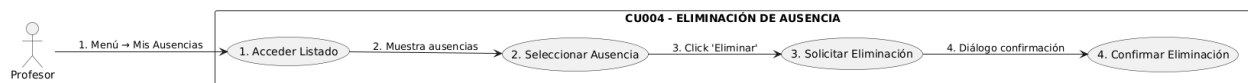
CU002 - REGISTRO DE AUSENCIA.



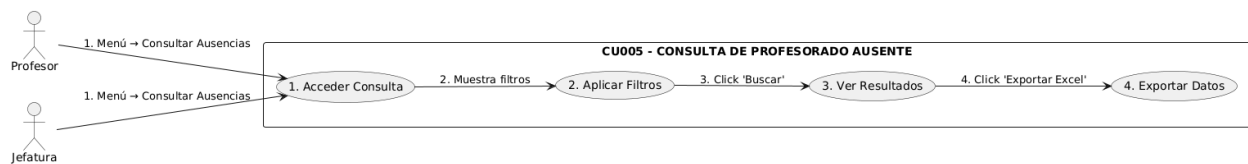
CU003 - MODIFICACIÓN DE AUSENCIA.



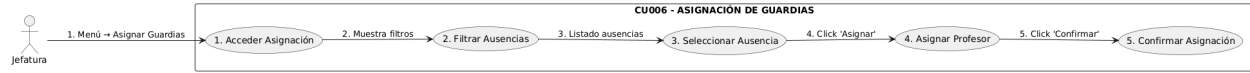
CU004 - ELIMINACIÓN DE AUSENCIA.



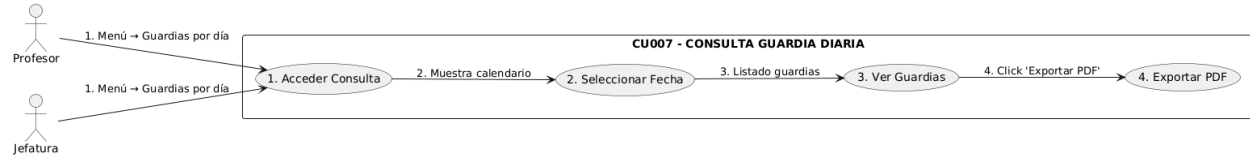
CU005 - CONSULTA DE PROFESORADO AUSENTE CON FILTRADO.



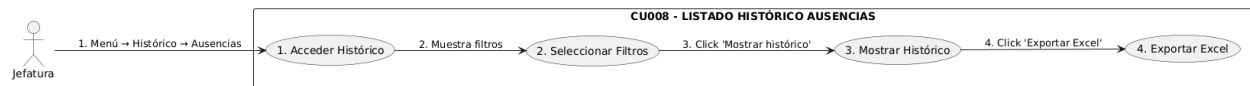
CU006 - ASIGNACIÓN DE GUARDIAS.



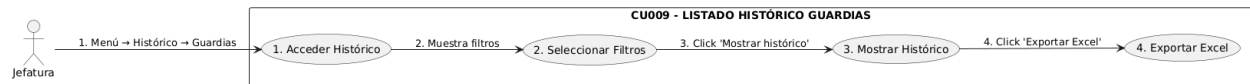
CU007 - CONSULTA DE LA GUARDIA PARA UN DÍA.



CU008 - LISTADO DE LA INFORMACIÓN HISTÓRICA DE AUSENCIAS.



CU009 - LISTADO DE LA INFORMACIÓN HISTÓRICA DE GUARDIAS.



5.2-ESPECIFICACIÓN DE CADA CASO DE USO.

CU001 - LOGIN.

- ❖ **Roles:** Todos.
- ❖ **Descripción:** Permite a los usuarios autenticarse en el sistema mediante integración con servicio externo (Jesús Luna Cordero).
- ❖ **Punta de entrada:** Acceder a URL → Formulario de login.
- ❖ **Precondición:**
 - El usuario no tiene una sesión activa.
- ❖ **Requisitos funcionales:**
 - RF001 - Autenticación de usuarios.
- ❖ **Secuencia de pasos:**
 1. El usuario introduce credenciales en un formulario.
 2. El sistema llama a servicio externo con usuario/clave.
 3. El servicio devuelve token JWT si las credenciales son válidas.
 4. El sistema llama a segundo servicio con usuario y token.
 5. El servicio devuelve la lista de módulos accesibles según el rol.
 6. El sistema muestra el menú adaptado al rol.
- ❖ **Postcondición:**
 - Usuario autenticado con menú adaptado a su rol.
 - Token de sesión almacenado.
- ❖ **Excepciones:**
 - Credenciales incorrectas: Mostrar mensaje “Credenciales no válidas”.
 - Servicio no disponible: Mostrar mensaje “Servicio no disponible, inténtalo de nuevo más tarde”.

CU002 - REGISTRO DE AUSENCIA.

- ❖ **Rol:** Profesor.
- ❖ **Descripción:** Permite registrar una ausencia con fecha, horario, motivo y tareas.
- ❖ **Punto de entrada:** Menú → Registrar ausencia.
- ❖ **Precondición:**
 - El profesor debe estar autenticado (CU001).
 - No debe haber otra ausencia registrada en el mismo horario.
- ❖ **Requisitos funcionales:**
 - RF002 - Registro de ausencias.
- ❖ **Secuencia de pasos:**
 - 1. Acceso al formulario:**
 - El profesor hace clic en "Registrar ausencia" en el menú lateral.
 - El sistema muestra un formulario vacío con:
 - Selector de fecha/hora de inicio.
 - Selector de fecha/hora de fin.
 - Campo de texto "Motivo" (Enfermedad, Formación, Otros...).
 - Campo de texto "Tareas para el alumnado".
 - Opción para adjuntar documento:
 - Formato: PDF, DOC, JPG o PNG.
 - Tamaño: máx. 5MB.
 - Botones "Enviar" y "Cancelar".
 - 2. Rellenado de datos:**
 - El profesor completa:
 - Fecha y hora inicio.
 - Fecha y hora fin.
 - Selecciona motivo.
 - Describe tareas.
 - Adjunta documento de las tareas (opcional).
 - 3. Validación y envío:**
 - El profesor hace clic en "Enviar".
 - El sistema verifica:
 - Que la fecha fin sea posterior a la fecha inicio.
 - Que no existan solapamientos con otras ausencias.
 - Que los campos obligatorios estén completos.
 - Que el documento tenga formato y tamaño correcto.

- 4. **Confirmación:**
 - El sistema muestra el mensaje: "Ausencia registrada correctamente (ID: AUS-XXXXX)".
 - Se redirige automáticamente a la pantalla "Mis ausencias".
- ❖ **Postcondición:**
 - La ausencia queda registrada en el sistema con estado "Pendiente de guardia".
 - ID único registrado para la ausencia.
 - El profesor recibe confirmación del registro.
- ❖ **Excepciones:**
 1. **Fecha fin anterior a la fecha inicio:**
 - Muestra mensaje: "No puede registrar la ausencia porque la fecha fin es anterior a la fecha inicio".
 2. **Solapamiento:**
 - El sistema detecta conflicto con otra ausencia.
 - Muestra mensaje: "Ya tiene una ausencia registrada en ese horario".
 3. **Datos incompletos:**
 - El sistema resalta en rojo los campos faltantes.
 - Muestra mensaje: "Complete los campos requeridos".
 4. **Documento no válido:**
 - Muestra mensaje: "El documento debe ser PDF, DOC, JPG o PNG (máx. 5MB)".
- ❖ **Dependencia con otro compañero:**
 - Servicio getModulos (Jesús Luna Cordero).

CU003 - MODIFICACIÓN DE AUSENCIA.

- ❖ **Rol:** Profesor.
- ❖ **Descripción:** Permite a un profesor modificar una ausencia registrada.
- ❖ **Punto de entrada:** Menú → Mis ausencias → Seleccionar ausencia → Editar.
- ❖ **Precondición:**
 - El profesor debe estar autenticado (CU001).
 - El profesor debe tener una ausencia registrada (CU002).
 - No debe tener una guardia asignada en ese horario (CU006).
- ❖ **Requisitos funcionales:**
 - RF003 - Modificación de ausencias.
- ❖ **Secuencia de pasos:**
 - 1. Acceso al listado:**
 - El profesor selecciona "Mis ausencias".
 - El sistema muestra tabla con:
 - ID Ausencia (no modificable).
 - Fecha y hora inicio.
 - Fecha y hora fin.
 - Motivo.
 - Estado.
 - Botón "Editar".
 - 2. Selección para edición:**
 - El profesor hace clic en "Editar" para una ausencia.
 - El sistema muestra el formulario precargado.
 - 3. Modificación:**
 - Campos editables:
 - Fecha y hora inicio.
 - Fecha y hora fin.
 - Motivo.
 - Tarea del alumnado.
 - Documento adjunto (opcional).

4. Guardado:

- El profesor hace clic en "Guardar cambios".
- El sistema verifica:
 - Que la fecha fin sea posterior a la fecha inicio.
 - Que no existan solapamientos con otras ausencias.
 - Que los campos obligatorios estén completos.
 - Que el documento tenga formato y tamaño correcto.

5. Solicitud de modificación:

- El sistema muestra mensaje de confirmación:
 - “¿Modificar ausencia?”.
 - Botones: “Confirmar”/“Cancelar”.

6. Confirmación:

- Si confirma:
 - El sistema verifica que no tenga guardias asignadas.
 - Elimina registro permanentemente.
 - Muestra mensaje: “Ausencia modificada correctamente (ID: AUS-XXXXX)”.
 - Se redirige automáticamente a la pantalla "Mis ausencias".
- Si cancela:
 - Se redirige automáticamente a la pantalla "Mis ausencias".

❖ Postcondición:

- La ausencia se actualiza en el sistema.
- El profesor recibe confirmación de la modificación.

❖ Excepciones:

1. Fecha fin anterior a la fecha inicio:

- Muestra mensaje: “No puede registrar la ausencia porque la fecha fin es anterior a la fecha inicio”.

2. Solapamiento:

- El sistema detecta conflicto con otra ausencia.
- Muestra mensaje: “Ya tiene una ausencia registrada en ese horario”.

3. Datos incompletos:

- Aunque el formulario aparece precargado, el sistema resalta en rojo los campos faltantes si el profesor ha dejado algún campo obligatorio vacío.
- Muestra mensaje: “Complete los campos requeridos”.

4. Documento no válido:

- Muestra mensaje: “El documento debe ser PDF, DOC, JPG o PNG (máx. 5MB)”.

❖ Dependencia con otro compañero:

- Servicio getModulos (Jesús Luna Cordero).

CU004 - ELIMINACIÓN DE AUSENCIA.

❖ Rol: Profesor.

❖ Descripción: Permite a un profesor eliminar una ausencia registrada.

❖ Punto de entrada: Menú → Mis ausencias → Seleccionar ausencia → Eliminar.

❖ Precondición:

- El profesor debe estar autenticado (CU001).
- El profesor debe tener una ausencia registrada (CU002).
- La ausencia debe estar en estado “Pendiente de guardia”.
- No debe tener una guardia asignada en ese horario (CU006).

❖ Requisitos funcionales:

- RF004 - Eliminación de ausencias.

❖ Secuencia de pasos:

1. Acceso al listado:

- El profesor selecciona “Mis ausencias”.
- El sistema muestra tabla con:
 - ID Ausencia (no modificable).
 - Fecha y hora inicio.
 - Fecha y hora fin.
 - Motivo.
 - Estado.
 - Botón "Eliminar".

2. Solicitud de eliminación:

- El profesor hace clic en “Eliminar”.
- El sistema muestra mensaje de confirmación:
 - “¿Eliminar ausencia del XX/XX/XXXX (XX:XX) al XX/XX/XXXX (XX:XX)? Esta acción no se puede deshacer”.
 - Botones: “Confirmar”/“Cancelar”.

3. Confirmación:

- Si confirma:
 - El sistema verifica que no tenga guardias asignadas.
 - Elimina registro permanentemente.
 - Muestra mensaje: “Ausencia eliminada correctamente (ID: AUS-XXXXX)”.
 - Se redirige automáticamente a la pantalla "Mis ausencias".
- Si cancela:
 - Se redirige automáticamente a la pantalla "Mis ausencias".

❖ Postcondición:

- La ausencia se elimina del sistema.
- El profesor recibe confirmación de la eliminación.

❖ Excepciones:

1. Guardia asignada:

- Muestra mensaje: “No se puede eliminar esta ausencia: tiene una guardia asignada”.

❖ Dependencia con otro compañero:

- Servicio getModulos (Jesús Luna Cordero).

CU005 - CONSULTA DE PROFESORADO AUSENTE CON FILTRADO.

- ❖ **Roles:** Profesor y jefatura.
- ❖ **Descripción:** Permite consultar la lista de profesores ausentes aplicando filtros.
- ❖ **Punto de entrada:** Menú → Consultar ausencias.
- ❖ **Precondición:**
 - El usuario debe estar autenticado (CU001).
- ❖ **Requisitos funcionales:**
 - RF005 - Consulta de profesorado ausente.
- ❖ **Secuencia de pasos:**
 - 1. Pantalla de filtros:**
 - Filtros:
 - Rango de fechas.
 - Departamento.
 - Estado.
 - 2. Aplicar filtros:**
 - Pulsar el botón “Buscar”.
 - 3. Resultados:**
 - Tabla con columnas:
 - Profesor | Fecha | Horario | Motivo | Estado
 - Opciones:
 - Paginación (20 resultados/página).
 - 4. Exportación:**
 - Botón “Exportar a Excel”.
 - Genera archivo con todos los resultados.
- ❖ **Postcondición:**
 - Se visualiza el listado filtrado.
 - Opción de exportación de resultados a Excel.
- ❖ **Excepciones:**
 - 1. Sin resultados:**
 - El sistema muestra el mensaje "No hay ausencias".
- ❖ **Dependencia con otro compañero:**
 - Servicio getModulos (Jesús Luna Cordero).

CU006 - ASIGNACIÓN DE GUARDIAS.

- ❖ **Rol:** Jefatura.
- ❖ **Descripción:** Permite asignar guardias a profesores para cubrir ausencias.
- ❖ **Punto de entrada:** Menú → Asignar guardias.
- ❖ **Precondición:**
 - Usuario autenticado como jefatura (CU001).
 - Existencia de ausencias pendientes (CU005).
- ❖ **Requisitos funcionales:**
 - RF006 - Asignación de guardias.
- ❖ **Secuencia de pasos:**
 1. **Visualización inicial:**
 - El sistema muestra:
 - Filtros (fecha, departamento).
 - Listado de ausencias pendientes.
 - Tabla con columnas:
 - Profesor | Fecha | Horario | Motivo | Botón “Asignar”
 2. **Proceso de asignación:**
 - Jefatura:
 1. Aplica filtros si es necesario.
 2. Selecciona “Asignar” en una ausencia.
 3. El sistema muestra:
 - Panel izquierdo: Detalle de la ausencia.
 - Panel derecho:
 - Lista de profesores disponibles.
 - Selector de horario.
 - Checkbox “Notificar al profesor”.
 3. **Confirmación:**
 - Jefatura:
 1. Selecciona profesor.
 2. Confirma asignación.
- ❖ **Postcondición:**
 - La guardia queda registrada.
 - Estado de ausencia actualizado a “Guardia asignada”.
 - Notificación enviada al profesor asignado.

❖ **Excepciones:**

1. **Profesor no disponible:**

- Muestra mensaje: “El profesor seleccionado tiene conflicto de horario”.

2. **Datos incompletos:**

- Resalta campos faltantes.
- Muestra mensaje: “Seleccione profesor y horario”.

❖ **Dependencia con otro compañero:**

- Servicio getModulos (Jesús Luna Cordero).

CU007 - CONSULTA DE LA GUARDIA PARA UN DÍA.

❖ **Rol:** Profesor y jefatura.

❖ **Descripción:** Visualización de guardias asignadas por día.

❖ **Punto de entrada:** Menú → Guardias por día.

❖ **Precondición:**

- El usuario debe estar autenticado (CU001).

❖ **Requisitos funcionales:**

- RF007 - Consulta de la guardia para un día.

❖ **Secuencia de pasos:**

1. **Selector de fecha:**

- El sistema muestra:
 - Calendario para selección de día.
 - Por defecto: fecha actual.

2. **Visualización:**

- Listado completo de guardias en el día seleccionado.

3. **Acciones:**

- Botón “Exportar a PDF”.

❖ **Postcondición:**

- Se visualiza el resumen de dicho día.

❖ **Excepciones:**

1. **Sin guardias asignadas:**

- Muestra mensaje: “No hay guardias en este día”.

❖ **Dependencia con otro compañero:**

- Servicio getModulos (Jesús Luna Cordero).

CU008 - LISTADO DE LA INFORMACIÓN HISTÓRICA DE AUSENCIAS.

- ❖ **Rol:** Jefatura.
- ❖ **Descripción:** Permite consultar registros históricos de ausencias.
- ❖ **Punto de entrada:** Menú → Histórico → Ausencias.
- ❖ **Precondición:**
 - Jefatura debe estar autenticada (CU001).
- ❖ **Requisitos funcionales:**
 - RF008 - Listado de la información histórica de ausencias.
- ❖ **Secuencia de pasos:**
 - 1. Pantalla inicial:**
 - Filtros:
 - Fecha y hora inicio.
 - Fecha y hora fin.
 - Botón “Exportar a Excel”.
 - 2. Generación del reporte:**
 - Jefatura:
 - Selecciona el rango de fechas.
 - Hace clic en “Mostrar histórico”.
 - 3. Visualización:**
 - El sistema muestra tabla con:
 - ID | Fecha | Hora | Profesor | Motivo | Estado
 - 4. Exportación:**
 - Opción para exportar a Excel.
- ❖ **Postcondición:**
 - Se obtiene el histórico.
- ❖ **Excepciones:**
 - 1. Sin resultados:**
 - Muestra mensaje: “No hay registros para los filtros seleccionados”.
 - 2. Rango inválido:**
 - Muestra mensaje: “La fecha y hora final debe ser posterior a la inicial”.
- ❖ **Dependencia con otro compañero:**
 - Servicio getModulos (Jesús Luna Cordero).

CU009 - LISTADO DE LA INFORMACIÓN HISTÓRICA DE GUARDIAS.

- ❖ **Rol:** Jefatura.
- ❖ **Descripción:** Permite consultar registros históricos de guardias.
- ❖ **Punto de entrada:** Menú → Histórico → Guardias.
- ❖ **Precondición:**
 - Jefatura debe estar autenticada (CU001).
- ❖ **Requisitos funcionales:**
 - RF009 - Listado de la información histórica de guardias.
- ❖ **Secuencia de pasos:**
 - 1. Pantalla inicial:**
 - Filtros:
 - Fecha y hora inicio.
 - Fecha y hora fin.
 - Botón “Exportar a Excel”.
 - 2. Generación del reporte:**
 - Jefatura:
 - Selecciona el rango de fechas.
 - Hace clic en “Mostrar histórico”.
 - 3. Visualización:**
 - El sistema muestra tabla con:
 - ID | Fecha | Hora | Profesor | Aula
 - 4. Exportación:**
 - Opción para exportar a Excel.
- ❖ **Postcondición:**
 - Se obtiene el histórico.
- ❖ **Excepciones:**
 - 1. Sin resultados:**
 - Muestra mensaje: “No hay registros para los filtros seleccionados”.
 - 2. Rango inválido:**
 - Muestra mensaje: “La fecha y hora final debe ser posterior a la inicial”.
- ❖ **Dependencia con otro compañero:**
 - Servicio getModulos (Jesús Luna Cordero).

6-DEFINICIÓN DE PLAN DE PRUEBAS.

6.1-INTRODUCCIÓN.

OBJETIVO.

Verificar que el sistema cumple con los requisitos funcionales y no funcionales, garantizando su correcto funcionamiento en la gestión de ausencias y asignación de guardias en el centro.

ALCANCE.

- Pruebas funcionales (autenticación, gestión de ausencias, asignación de guardias).
- Pruebas no funcionales (seguridad, usabilidad, compatibilidad).
- Pruebas de rendimiento (tiempos de respuesta, carga concurrente).

6.2-ESTRUCTURA Y ORGANIZACIÓN.

1. Introducción.
2. Alcance.
3. Objetivos.
4. Tipos de prueba y cobertura.
5. Casos de prueba.
6. Herramientas.
7. Cronograma.
8. Documentación de resultados.
9. Validación y aceptación.
10. Análisis de riesgos.
11. Presentación y estilo.

6.3-COBERTURA DE PRUEBAS.

FUNCIONALES.

- Autenticación y control de acceso por roles (RF001).
- Registro, modificación y eliminación de ausencias (RF002, RF003, RF004).
- Asignación y consulta de guardias (RF006, RF007).
- Histórico de ausencias/guardias (RF008).

NO FUNCIONALES.

- **Seguridad:** Acceso restringido por roles (RNF001).
- **Usabilidad:** Interfaz intuitiva (RNF002).
- **Compatibilidad:** Navegadores y dispositivos (RNF003).

PRUEBAS DE RENDIMIENTO.

- Tiempo de respuesta < 1 segundo en operaciones críticas (ej: login, consultas).
- Carga concurrente: 50 usuarios simultáneos.

6.4-IDENTIFICACIÓN DE CASOS DE PRUEBA.

ID	FUNCIONALIDAD	ENTRADA	SALIDA ESPERADA	CRITERIO DE ÉXITO
CP01	LOGIN (CU001)	Credenciales válidas.	Acceso según rol.	Redirección correcta.
CP02	Registrar ausencia (CU002)	Fecha, horario, motivo.	Ausencia registrada en sistema.	Confirmación visual.
CP03	Asignar guardia (CU006)	Profesor disponible y horario.	Guardia asignada y notificada.	Registro en BBDD y notificación.
CP04	Consultar histórico (CU008)	Rango de fecha.	Listado de ausencias / guardias.	Datos consistentes.
CP05	Seguridad (RNF001)	Intento de acceso sin autenticación.	Redirección a login.	Bloqueo de acceso no autorizado.

6.5-COBERTURA DE REQUISITOS.

- RF001 -> CP01
- RF002 -> CP02
- RF003 -> CP03
- RF004 -> CP04
- RNF001 -> CP05

6.6-ESTRATEGIA Y HERRAMIENTAS.

TIPO DE PRUEBA	ESTRATEGIA	HERRAMIENTAS
Unitarias	Validar componentes.	JUnit, Mockito.
Integración	Comunicación frontend-backend.	Postman.
Seguridad	Validar autenticación JWT y restricciones por rol.	Postman.
Rendimiento	Simulación de carga.	JMeter.
Usabilidad	Evaluar interfaz intuitiva y accesibilidad.	Pruebas manuales.

6.7-CRONOGRAMA.

FASE	ACTIVIDAD	FECHA INICIO	FECHA FIN
Preparación	Diseño de casos y entorno.	01/05/2025	03/05/2025
Ejecución	Pruebas funcionales/no funcionales.	04/05/2025	10/05/2025
Evaluación	Análisis de resultados.	11/05/2025	12/05/2025
Aceptación	Validación con profesores.	13/05/2025	14/05/2025

6.8-DOCUMENTACIÓN DE RESULTADOS.

- **Registro en informe:**
 - ID de caso, resultado (exitoso/fallido), evidencias.
 - Gestión de incidencias (severidad, prioridad).
- **Herramientas:** Excel.

6.9-VALIDACIÓN Y ACEPTACIÓN.

Criterios:

- 100% de casos críticos aprobados.
- $\leq 3\%$ de casos no críticos fallidos.
- Resolución de defectos de alta prioridad antes de despliegue.

6.10-ANÁLISIS DE RIESGOS.

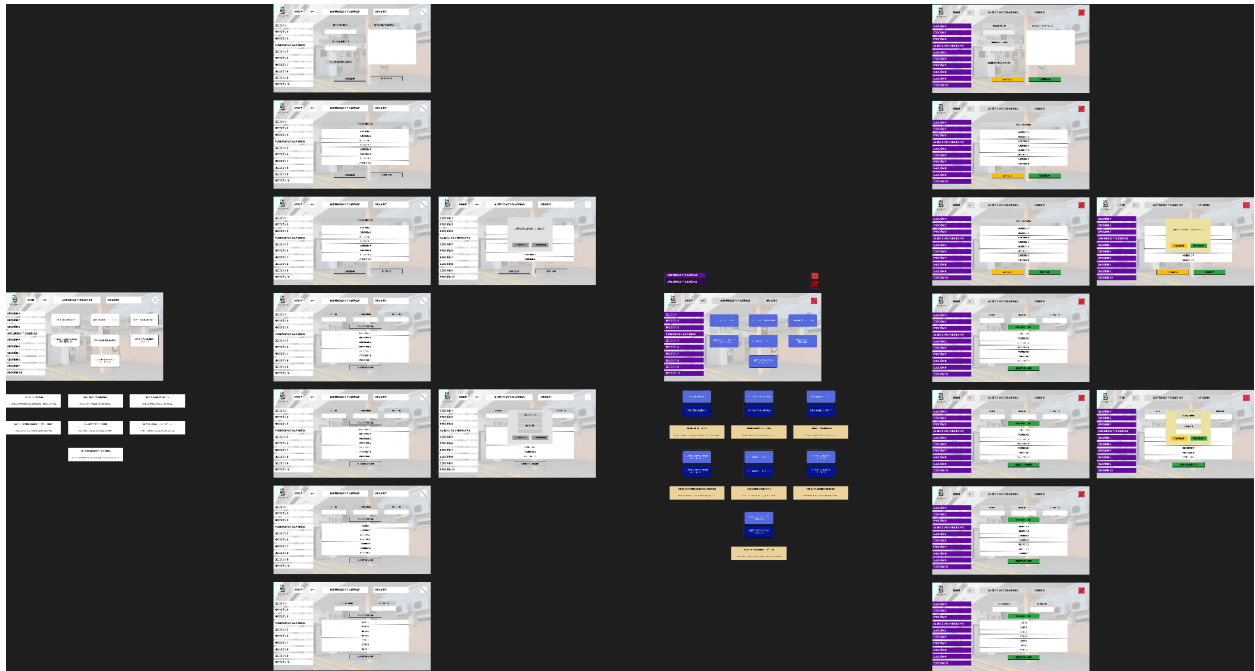
RIESGO	PROBABILIDAD	IMPACTO	MITIGACIÓN
Cambios en requisitos	Media	Alto	Reuniones semanales con profesores.
Retraso en pruebas	Baja	Medio	Priorización de casos críticos.
Fallos de integración	Alta	Alto	Pruebas automatizadas.

7-DEFINICIÓN DE INTERFACES GRÁFICAS.

7.1-NAVEGACIÓN.

7.2-ESTILOS BASE.

7.3-WIREFRAME DIGITAL (FIGMA).

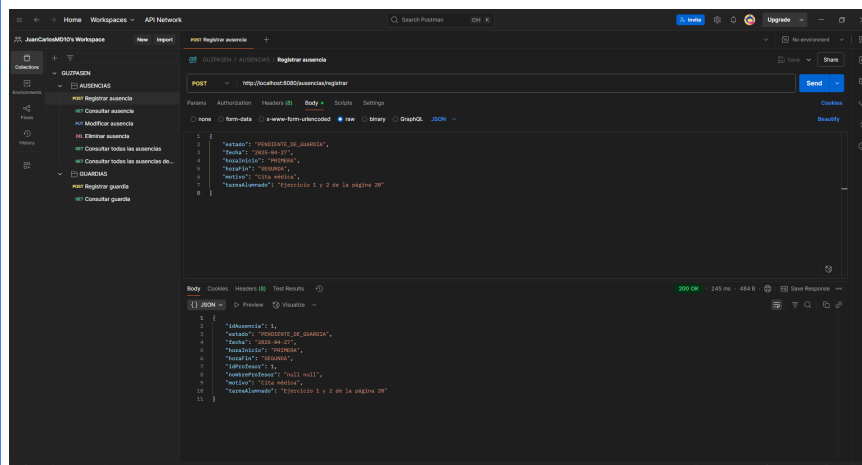


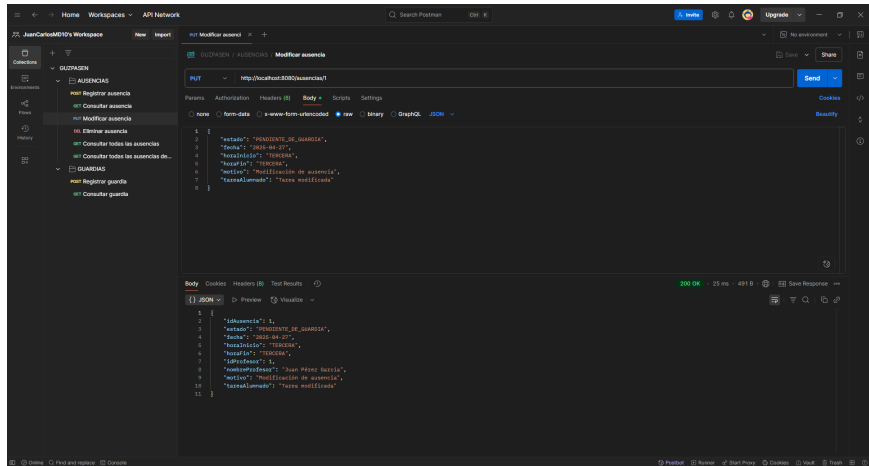
7.4-ENLACE A MI PROTOTIPO.

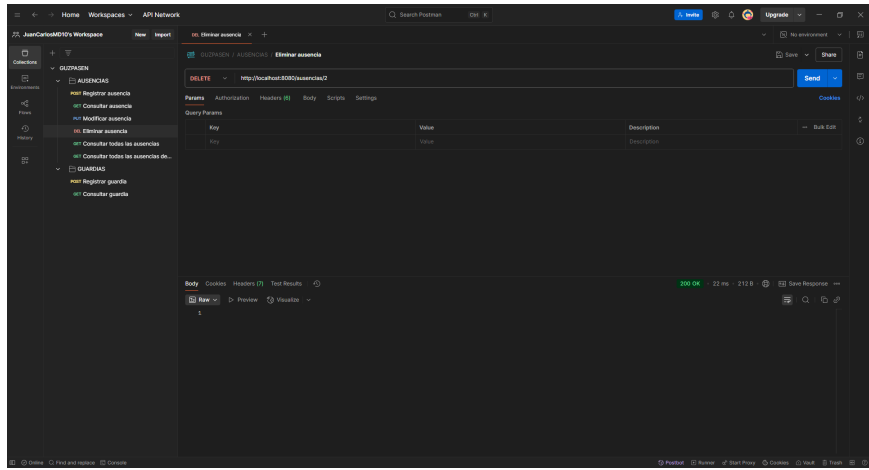
[ENLACE AL PROTOTIPO](#)

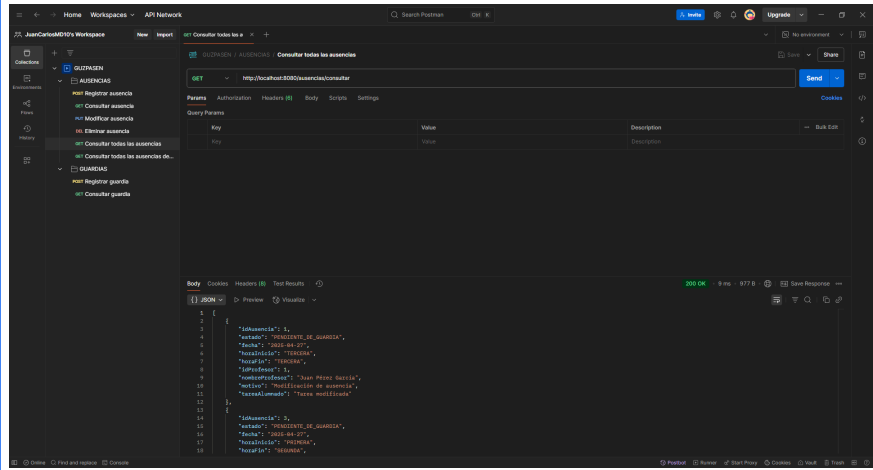
8-SERVICIOS INTERNOS.

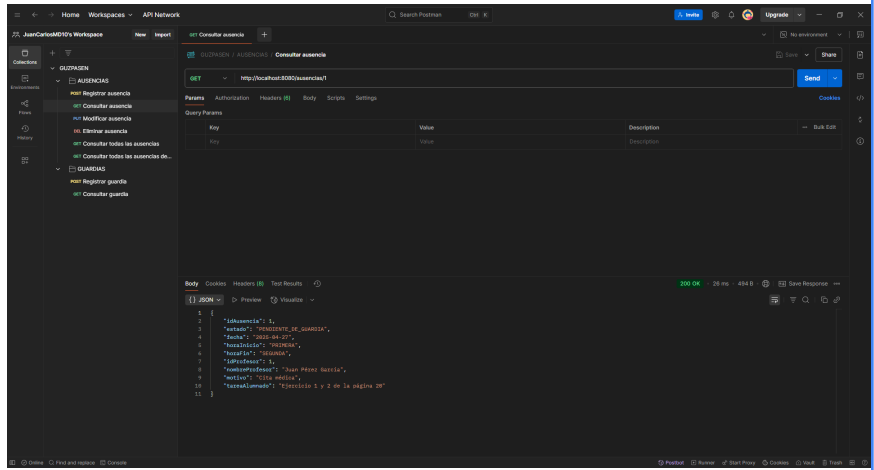
8.1-SERVICIOS DE AUSENCIAS.

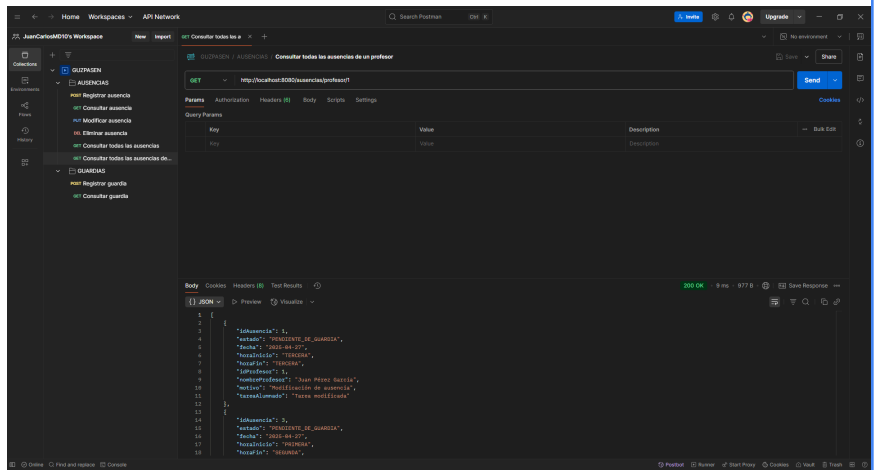
Servicio: Registrar Ausencia	
Ruta URL	http://localhost:8080/ausencias/regarstrar
Método	POST
Entrada: valores y tipo	JSON en el cuerpo de la petición { "estado": "PENDIENTE_DE_GUARDIA", "fecha": "2025-04-27", "horaInicio": "PRIMERA", "horaFin": "SEGUNDA", "motivo": "Cita médica", "tareaAlumnado": "Ejercicio 1 y 2 de la página 20" }
Modo de Acceso	Postman/Frontend web.
Formato Respuesta	JSON
Código Respuesta	200 OK
Descripción	Registra una nueva ausencia de profesor.
Captura	

Servicio: Modificar Ausencia	
Ruta URL	http://localhost:8080/ausencias/{id}
Método	PUT
Entrada: valores y tipo	<p>JSON en el cuerpo de la petición</p> <pre>{ "estado": "PENDIENTE_DE_GUARDIA", "fecha": "2025-04-27", "horaInicio": "TERCERA", "horaFin": "TERCERA", "motivo": "Modificación de ausencia", "tareaAlumnado": "Tarea modificada" }</pre>
Modo de Acceso	Postman/Frontend web.
Formato Respuesta	JSON
Código Respuesta	200 OK
Descripción	Modifica los datos de una ausencia existente.
Captura	 <p>The screenshot shows the Postman interface for a PUT request. The URL is <code>http://localhost:8080/ausencias/1</code>. The body is a JSON object: <code>{ "estado": "PENDIENTE_DE_GUARDIA", "fecha": "2025-04-27", "horaInicio": "TERCERA", "horaFin": "TERCERA", "motivo": "Modificación de ausencia", "tareaAlumnado": "Tarea modificada" }</code>. The response status is <code>200 OK</code> with a response time of <code>25 ms</code>. The response body is a JSON array containing the updated absence object.</p>

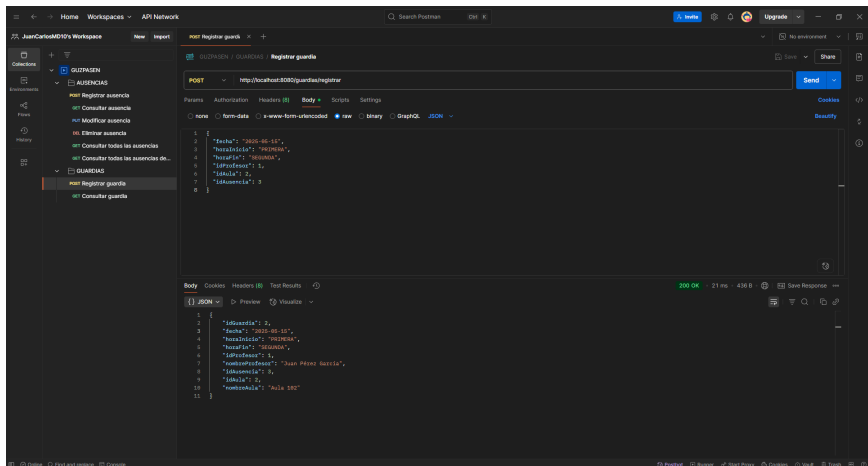
Servicio: Eliminar Ausencia	
Ruta URL	http://localhost:8080/ausencias/{id}
Método	DELETE
Entrada: valores y tipo	ID en la URL
Modo de Acceso	Postman/Frontend web.
Formato Respuesta	Ninguno
Código Respuesta	200 OK
Descripción	Elimina una ausencia existente.
Captura	 <p>The screenshot shows the Postman interface with a DELETE request configured. The URL is <code>http://localhost:8080/ausencias/2</code>. The response status is <code>200 OK</code> with a response time of 22 ms and a size of 212 B. The response body is empty, indicated by a single character 't' in the raw view.</p>

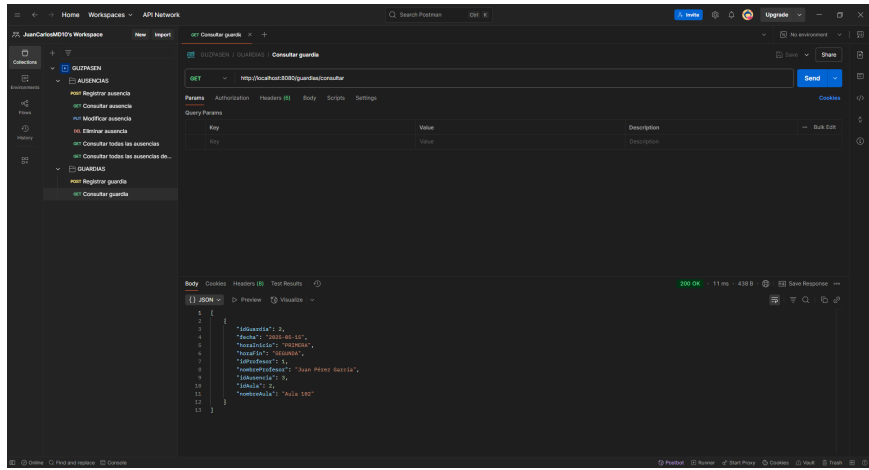
Servicio: Consultar Ausencias	
Ruta URL	http://localhost:8080/ausencias/consultar
Método	GET
Entrada: valores y tipo	Según estado
Modo de Acceso	Postman/Frontend web.
Formato Respuesta	JSON
Código Respuesta	200 OK
Descripción	Lista de ausencias.
Captura	 <p>The screenshot shows a Postman interface with a GET request to the URL 'http://localhost:8080/ausencias/consultar'. The response is a JSON array of objects, each representing an absence record. The status of all records is 'FRENTE'.</p> <pre> [{ "idAusencia": 1, "estado": "FRENTE", "fecha": "2021-08-27", "horaInicio": "10:00:00", "horaFin": "12:00:00", "nombreEmpleado": "Juan Pérez García", "motivo": "Permiso de ausencia", "observaciones": "Permiso de ausencia" }, { "idAusencia": 2, "estado": "FRENTE", "fecha": "2021-08-27", "horaInicio": "10:00:00", "horaFin": "12:00:00", "nombreEmpleado": "Juan Pérez García", "motivo": "Permiso de ausencia", "observaciones": "Permiso de ausencia" }] </pre>

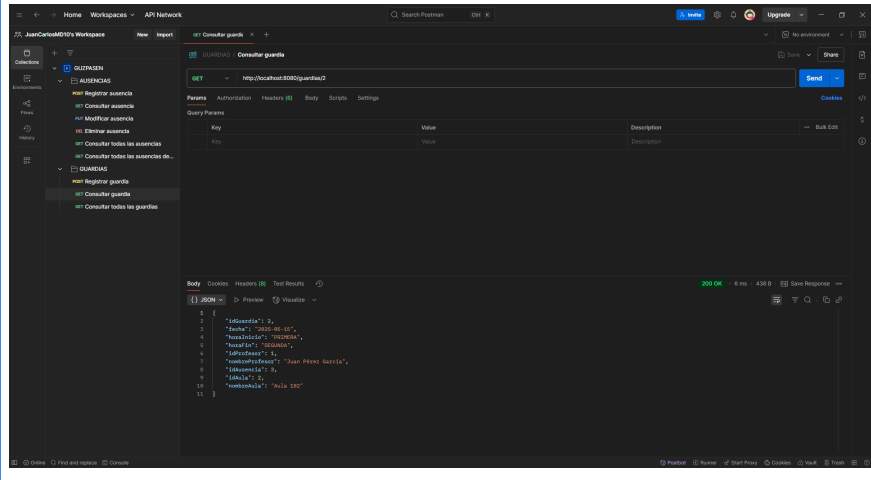
Servicio: Obtener Ausencia Por Id	
Ruta URL	http://localhost:8080/ausencias/{id}
Método	GET
Entrada: valores y tipo	ID en la URL
Modo de Acceso	Postman/Frontend web.
Formato Respuesta	JSON
Código Respuesta	200 OK
Descripción	Devuelve la ausencia correspondiente al ID.
Captura	 <p>The screenshot shows the Postman interface with a GET request to <code>http://localhost:8080/ausencias/1</code>. The response is a 200 OK status with a JSON body containing the following data:</p> <pre> { "idAusencia": 1, "estado": "PENDING_OT_GABRIELA", "fecha": "2020-06-27", "ausenciaId": "PENDING", "fechaFin": "2020-06-27", "idEmpleado": 1, "nombreEmpleado": "Juan Pedro Riquelme", "motivo": "Falta médica", "observaciones": "Exposición a y 2 de la página 28" } </pre>

Servicio: Obtener Ausencia Por Id De Profesor	
Ruta URL	http://localhost:8080/ausencias/profesor/{idProfesor}
Método	GET
Entrada: valores y tipo	ID en la URL
Modo de Acceso	Postman/Frontend web.
Formato Respuesta	JSON
Código Respuesta	200 OK
Descripción	Devuelve las ausencias correspondientes al ID del profesor.
Captura	 <p>The screenshot shows the Postman interface. On the left, a collection named 'API REST' is expanded, showing a folder 'API REST' and a request 'Consultar todas las ausencias de un profesor'. The main panel shows the details of this request: a GET method to the URL 'http://localhost:8080/ausencias/profesor/1'. The 'Body' tab is selected, showing a JSON response with a status of '200 OK'. The response body is a JSON array of two objects, each representing an absence record with fields like 'idAusencia', 'idProfesor', 'fecha', 'hora', 'descripcion', and 'estado'.</p>

8.2-SERVICIOS DE GUARDIAS.

Servicio: Asignar Guardia	
Ruta URL	http://localhost:8080/guardias/registrar
Método	POST
Entrada: valores y tipo	<p>JSON en el cuerpo de la petición</p> <pre>{ "fecha": "2025-05-15", "horaInicio": "PRIMERA", "horaFin": "SEGUNDA", "idProfesor": 1, "idAula": 2, "idAusencia": 3 }</pre>
Modo de Acceso	Postman/Frontend web.
Formato Respuesta	JSON
Código Respuesta	200 OK
Descripción	Asigna una nueva guardia a un profesor.
Captura	 <p>The screenshot shows the Postman interface. On the left, a collection named 'GUARDIAS' is expanded, showing a request 'Registrar guardia'. The main panel shows the details of this POST request. The URL is 'http://localhost:8080/guardias/registrar'. The body is a JSON object: { "fecha": "2025-05-15", "horaInicio": "PRIMERA", "horaFin": "SEGUNDA", "idProfesor": 1, "idAula": 2, "idAusencia": 3 }. The response is shown as '200 OK' with a JSON body: { "idAusencia": 2, "fecha": "2025-05-15", "horaInicio": "PRIMERA", "horaFin": "SEGUNDA", "idProfesor": 1, "idAusencia": 2, "idAula": 2, "horaFin": "SEGUNDA", "idProfesor": 1, "idAusencia": 2, "idAula": 2, "horaFin": "SEGUNDA" }.</p>

Servicio: Consultar Guardias	
Ruta URL	http://localhost:8080/guardias/consultar
Método	GET
Entrada: valores y tipo	-
Modo de Acceso	Postman/Frontend web.
Formato Respuesta	JSON
Código Respuesta	200 OK
Descripción	Lista las guardias de profesores.
Captura	 <p>The screenshot shows the Postman interface with a GET request to the URL <code>http://localhost:8080/guardias/consultar</code>. The response status is <code>200 OK</code> with a response time of <code>11 ms</code> and a size of <code>438 B</code>. The response body is a JSON array containing 11 objects, each representing a guardia with fields like <code>idGuardia</code>, <code>fecha</code>, <code>horaInicio</code>, <code>horaFin</code>, <code>descripcion</code>, <code>profesor</code>, and <code>asignatura</code>.</p> <pre> [{ "idGuardia": 1, "fecha": "2023-05-10", "horaInicio": "08:00", "horaFin": "09:00", "descripcion": "Clase de Matemáticas", "profesor": "Juan Pérez García", "asignatura": "Matemáticas" }, { "idGuardia": 2, "fecha": "2023-05-10", "horaInicio": "09:00", "horaFin": "10:00", "descripcion": "Clase de Física", "profesor": "María López", "asignatura": "Física" }, { "idGuardia": 3, "fecha": "2023-05-10", "horaInicio": "10:00", "horaFin": "11:00", "descripcion": "Clase de Química", "profesor": "Carlos Rodríguez", "asignatura": "Química" }, { "idGuardia": 4, "fecha": "2023-05-10", "horaInicio": "11:00", "horaFin": "12:00", "descripcion": "Clase de Biología", "profesor": "Ana Martínez", "asignatura": "Biología" }, { "idGuardia": 5, "fecha": "2023-05-10", "horaInicio": "12:00", "horaFin": "13:00", "descripcion": "Clase de Historia", "profesor": "Pedro Sánchez", "asignatura": "Historia" }, { "idGuardia": 6, "fecha": "2023-05-10", "horaInicio": "13:00", "horaFin": "14:00", "descripcion": "Clase de Geografía", "profesor": "Lucía Fernández", "asignatura": "Geografía" }, { "idGuardia": 7, "fecha": "2023-05-10", "horaInicio": "14:00", "horaFin": "15:00", "descripcion": "Clase de Inglés", "profesor": "Roberto Díaz", "asignatura": "Inglés" }, { "idGuardia": 8, "fecha": "2023-05-10", "horaInicio": "15:00", "horaFin": "16:00", "descripcion": "Clase de Arte", "profesor": "Sofía Navarro", "asignatura": "Arte" }, { "idGuardia": 9, "fecha": "2023-05-10", "horaInicio": "16:00", "horaFin": "17:00", "descripcion": "Clase de Música", "profesor": "Diego Morales", "asignatura": "Música" }, { "idGuardia": 10, "fecha": "2023-05-10", "horaInicio": "17:00", "horaFin": "18:00", "descripcion": "Clase de Educación Física", "profesor": "Elena Ruiz", "asignatura": "Educación Física" }, { "idGuardia": 11, "fecha": "2023-05-10", "horaInicio": "18:00", "horaFin": "19:00", "descripcion": "Clase de Tecnología", "profesor": "Alejandro Torres", "asignatura": "Tecnología" }] </pre>

Servicio: Obtener Guardia Por Id	
Ruta URL	http://localhost:8080/guardias/{id}
Método	GET
Entrada: valores y tipo	ID en la URL
Modo de Acceso	Postman/Frontend web.
Formato Respuesta	JSON
Código Respuesta	200 OK
Descripción	Devuelve la guardia correspondiente al ID.
Captura	 <p>The screenshot shows a Postman interface with a GET request to <code>http://localhost:8080/guardias/2</code>. The response is a JSON object with the following structure:</p> <pre> { "idGuardia": 2, "fecha": "2020-06-01", "descripcion": "Reserva", "usuario": "Juan Pérez Realista", "telefono": 9, "telefonoReserva": "Juan Pérez Realista", "idMala": 2, "telefonoMala": "Raja 180" } </pre>

9-MANUAL DE USUARIO.

Este manual se encuentra en la carpeta `src/main/resources` del backend del proyecto para no hacer infinito este documento.