

Universidad La Salle



Construcción de Software

Informe Construcción de Software

Integrantes:

Luisfelipe Mamani Arosquipa, Patrick Paredes
Neira

Docente:

Msc. Vicente Machaca Arceda
Séptimo Semestre - Ingeniería de Software

2023

1. Introducción:

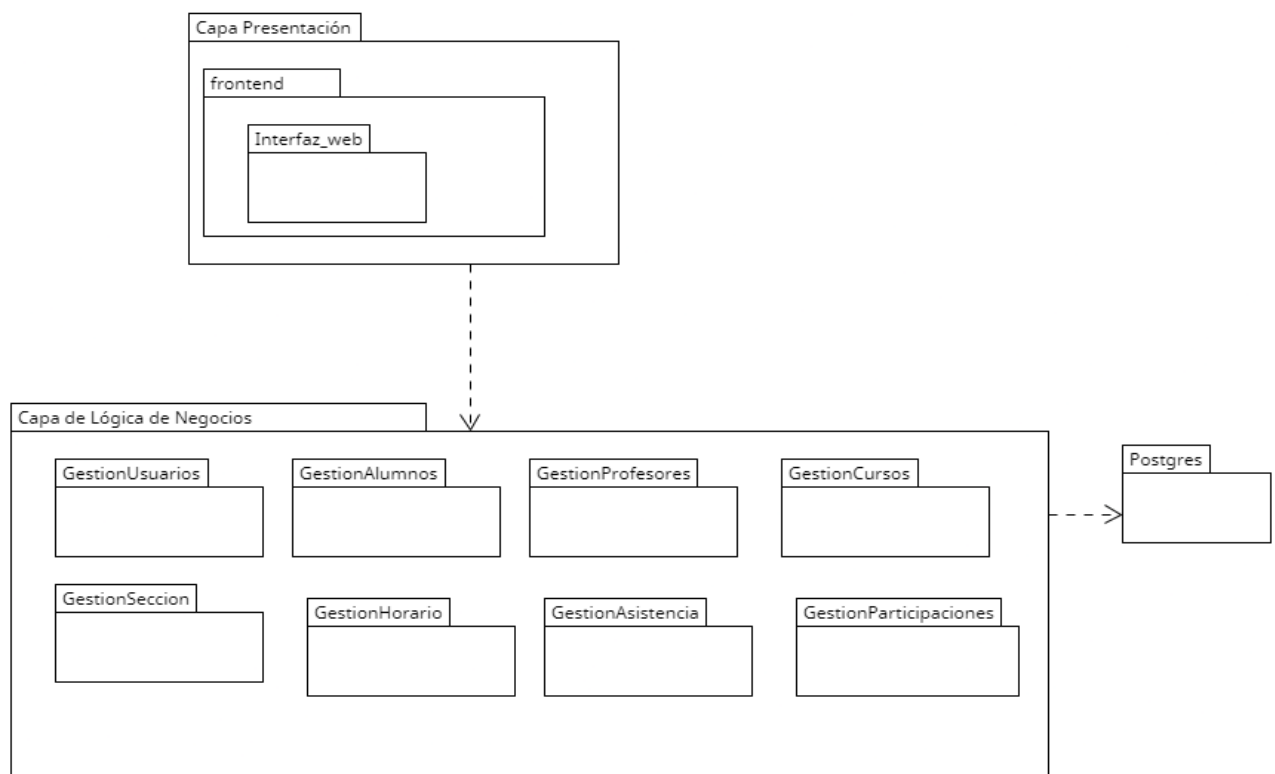
En la actualidad sabemos que las aplicaciones web cada vez son más populares y es común encontrarnos con proyectos que requieran consumir APIs, es también de nuestro conocimiento que el sistema académico con el que contamos se encuentra diseñado para situaciones nuevas como es el caso en que en un curso sea dictado por dos docentes al mismo tiempo o que un curso pueda tener varias secciones, es por ello que en este proyecto trabajaremos con la implementación de una API restful que se adapte a las necesidades futuras de la universidad y que pueda ser escalable, el uso de una API restful nos permite consumir APIs de manera muy efectiva, el objetivo de nuestro proyecto es poder hacer un sistema de registro de asistencia con reconocimiento facial.

Nuestro proyecto consta en realizar los siguientes procesos:

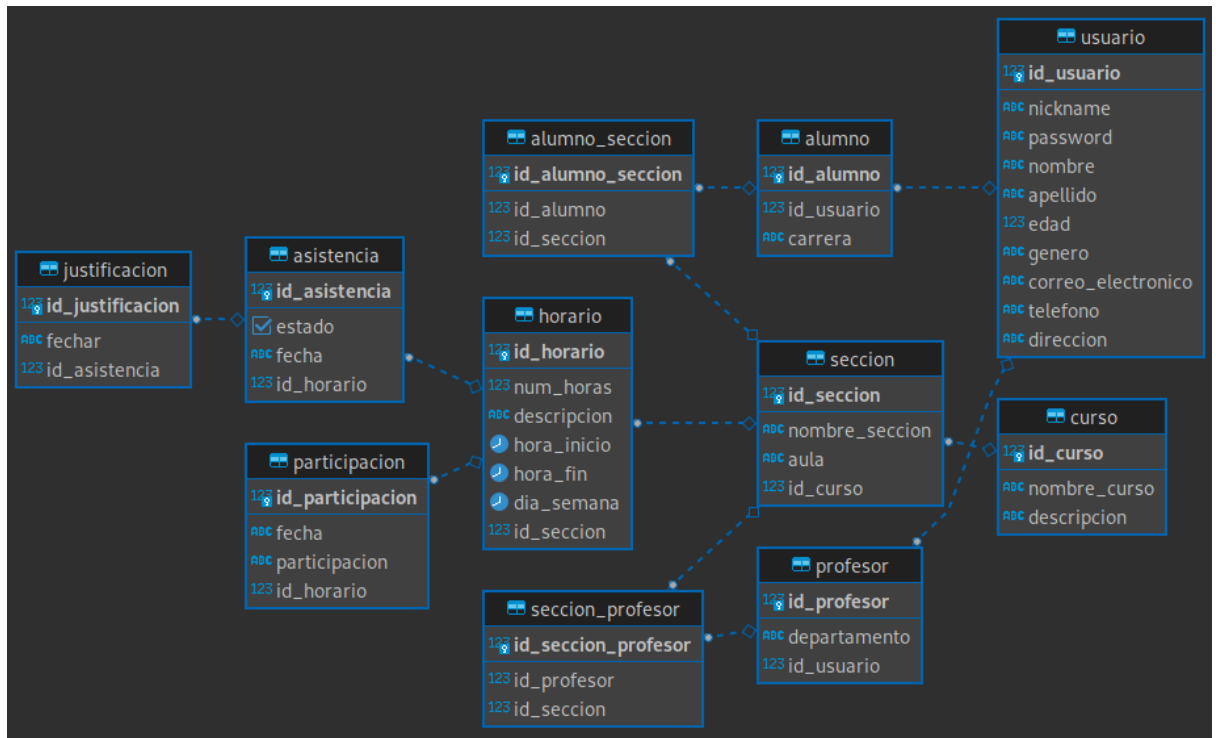
- contabilizar las asistencias y las intervenciones que tengan los alumnos a lo largo de su periodo de dictado de clases.
- mejorar la confiabilidad de los registros de asistencia añadiendo un reconocimiento facial el cual nos permitirá saber si la persona que registró la asistencia se encuentra presente.

2. Propuesta:

2.1 Arquitectura:



2.2 Base de datos:



2.3 Documentación de Endpoints

❖ Curso

➤ Crear Curso

- http://127.0.0.1:5000/create_curso
- Donde se va a crear un nuevo curso
- Request y response

POST http://127.0.0.1:5000/create_curso Send 200 OK 41.4 ms 48 B

JSON Auth Query Headers 1 Docs

```
1 {
2   "nombre_curso": "botanica",
3   "descripcion": "este curso se lleva 5 horas a la semana"
4 }
```

Preview Headers 5 Cookies Timeline

```
1 {
2   "ID_CURSO": 7,
3   "message": "Usuario OK"
4 }
```

■ Fields

Field	Type	Required?	Description
nombre_curso	varchar	required	Es el nombre que tiene el curso
descripcion	varchar	required	Es la descripción que tiene el curso

■ Post Objects:

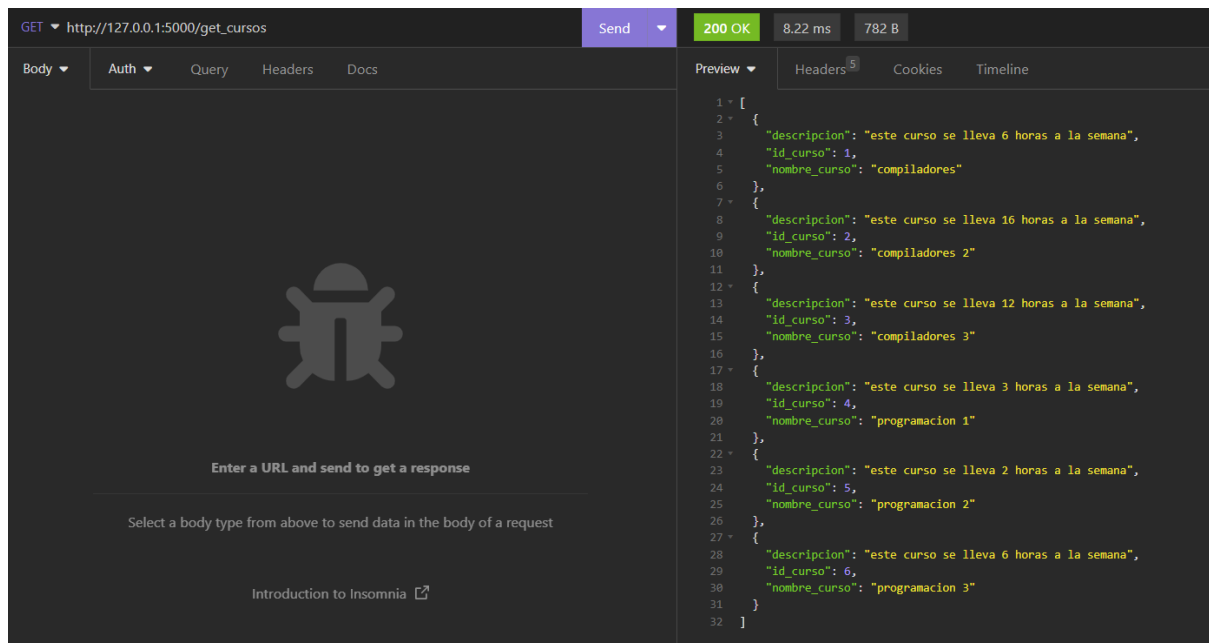
Field	Type	Description
id_curso	int	Representa el id que se le asigna luego de la inserción en la tabla
message	varchar	Representa la comprobación de que se ingreso correctamente los datos en la tabla

■ Errores_posibles

Error	Descripción
400 Bad Request	Alguno de los campos que fueron ingresados no existen

➤ Listar Curso

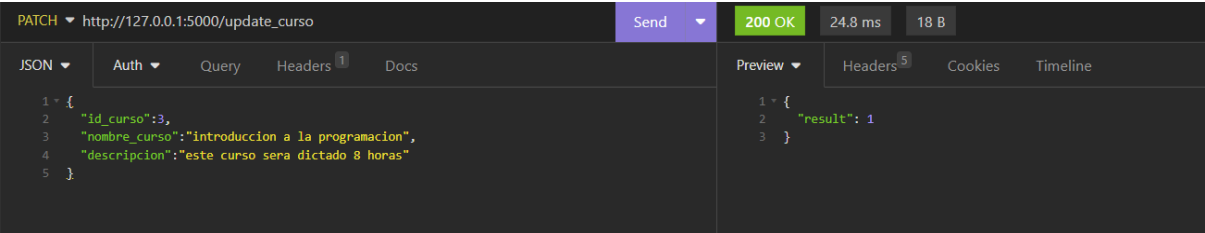
- http://127.0.0.1:5000/get_cursos
- Donde se va visualizar los cursos
- Response



■ Get Objects:

Field	Type	Description
id_curso	int	Representa el id que tiene el curso en la base de datos curso
descripción	varchar	Representa la descripción de cada curso que vamos ingresando a la base de datos
nombre_curso	varchar	Representa el nombre del curso el cual nos muestra

- Listar Curso por id
- Actualizar Curso
 - http://127.0.0.1:5000/update_curso
 - Donde se harán actualizaciones de los cursos
 - Request y response



■ Fields

Field	Type	Required?	Description
id_curso	int	required	Es el identificador del curso
nombre_curso	varchar	required	Es el nombre del curso
descripcion	varchar	required	Es la descripción del curso

■ Patch Objects:

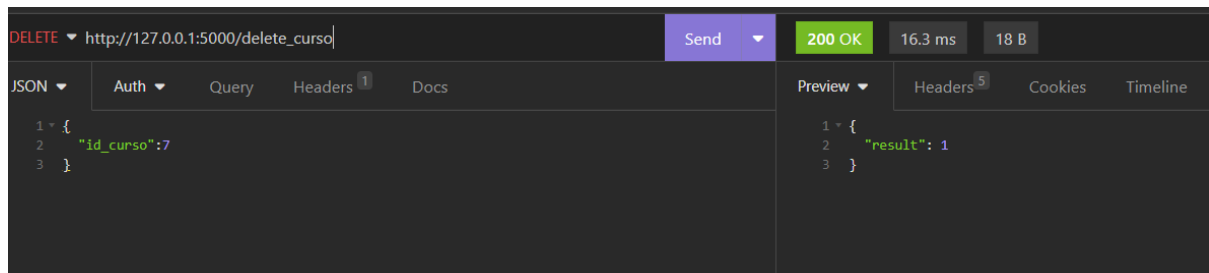
Field	Type	Description
result	int	Representa que se actualizó exitosamente la tabla curso

■ Errores_posibles

Error	Descripción
400 Bad Request	Alguno de los campos que fueron ingresados no existen

➤ Borrar Curso

- `http://127.0.0.1:5000/delete_curso`
- Donde se va a eliminar el curso con respecto a un identificador
- Request y response



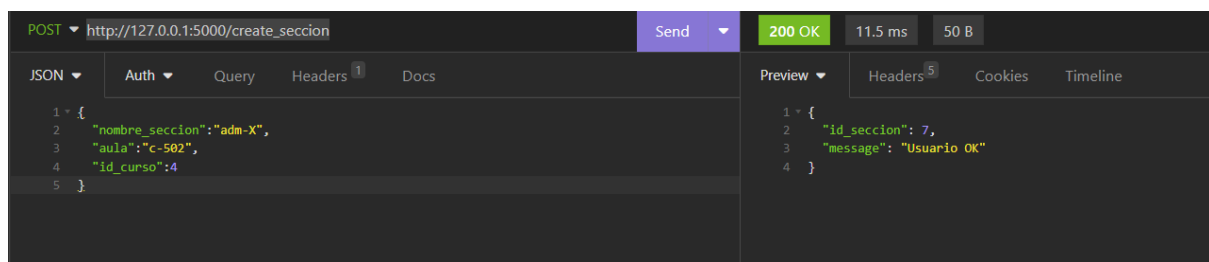
Posibles Errores

Error	Descripción
400 Bad Request	Alguno de los campos que fueron ingresados en curso no existen

❖ Sección

➤ Crear Sección

- http://127.0.0.1:5000/create_seccion
- response y request
- Se va a insertar datos en sección



■ Fields

Field	Type	Required?	Description
nombre_seccion	varchar	required	Es el nombre que tiene la seccion
aula	varchar	required	Es el aula de la seccion
id_curso	int	required	Es el identificador del curso

■ Post Objects:

Field	Type	Description
id_seccion	int	Representa el id que se le asigna luego de la inserción en la tabla seccion
message	varchar	Representa la comprobación de que se ingreso correctamente los datos en la tabla seccion

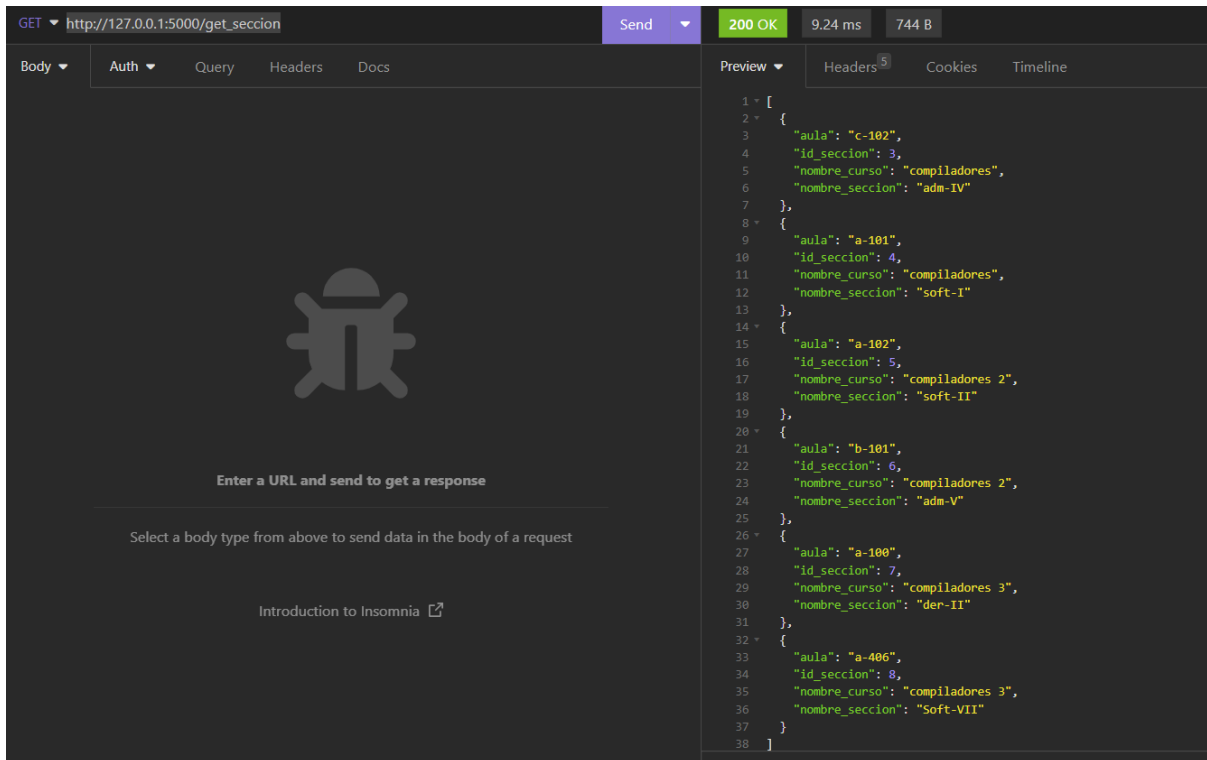
■ Errores_posibles

Error	Descripción
400 Bad Request	Alguno de los campos que fueron ingresados no existen

➤ Listar Sección

- http://127.0.0.1:5000/get_seccion

- response
- donde se va a mostrar los datos que esta en listar

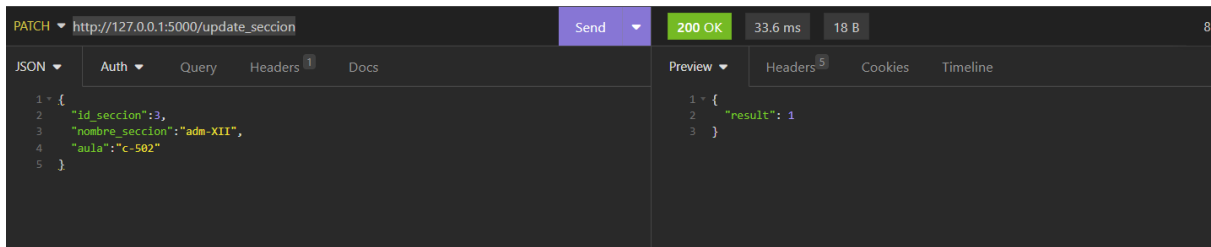


■ Post_object:

Field	Type	Description
id_seccion	int	Representa el id que tiene la sección en la base de datos curso
nombre_curso	varchar	Representa el nombre del curso de la tabla de cursos
nombre_seccion	varchar	representa el nombre de la seccion
aula	varchar	representa la aula

- Listar Sección por id
- Actualizar Sección
 - http://127.0.0.1:5000/update_seccion
 - Donde se van a actualizar los datos de la tabla

■ Request y response



■ Fields

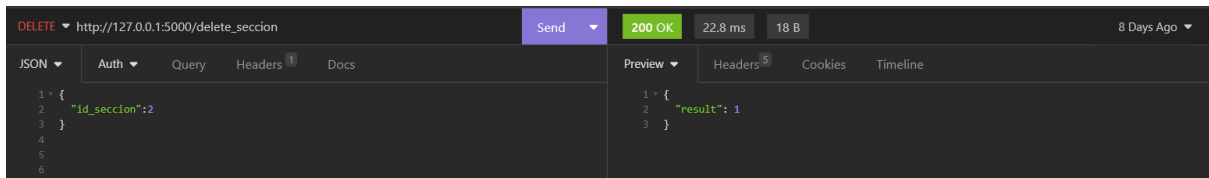
Field	Type	Required?	Description
id_seccion	int	required	Es id de la seccion
nombre_seccion	varchar	required	Es el nombre de la seccion
aula	varchar	required	Es el nombre del aula

■ Patch_objects

Field	Type	Description
result	int	Representa que se hizo la inserción de manera exitosa

➤ Borrar Sección

- http://127.0.0.1:5000/delete_seccion
- request y response
- Donde se va a eliminar la seccion con el id especificado



■ Errores posibles

Error	Descripción
400 Bad Request	Alguno de los campos que fueron ingresados no existen

❖ Horario

➤ Crear Horario

- http://127.0.0.1:5000/create_horario
- request y response
- Donde se va a crear un horario con el id especificado

■ Fields

Field	Type	Required?	Description
num_horas	varchar	required	Es el numero de horas
descripcion	varchar	required	Es la descripcion
hora_inicio	varchar	required	Es la hora de inicio
hora_fin	varchar	required	Es la hora de finalizacion
dia_semana	varchar	required	Es el dia de la semana
id_seccion	int	required	Es el id de la seccion

■ Post Objects:

Field	Type	Description
id_horario	int	Representa el id que se le asigna luego de la inserción en la tabla horario
message	varchar	Representa la comprobación de que se ingreso correctamente los datos en la tabla horario

■ Errores_posibles

Error	Descripción
400 Bad Request	Alguno de los campos que fueron ingresados no existen

➤ Listar Horario

- http://127.0.0.1:5000/get_horario
- response
- Donde se va a enlistar con el horario

❖ Post Objects:

Field	Type	Description
id_horario	int	Representa el id que se le asigna luego de la inserción en la tabla horario
message	varchar	Representa la comprobación de que se ingreso correctamente los datos en la tabla horario

➤ **Actualizar Horario**

- http://127.0.0.1:5000/update_horario
- request y response
- Donde se va a actualizar el horario con el id especificado

❖ Fields

Field	Type	Required?	Description
id_horario	int	required	es el id del horario
num_horas	varchar	required	Es el numero de horas
descripcion	varchar	required	Es el la descripcion del horario
hora_inicio	varchar	required	hora de inicio
hora_fin	varchar	required	hora de fin
dia_semana	varchar	required	dia de la semana
id_seccion	int	required	identificador de la seccion

❖ Post Objects:

Field	Type	Description
message	varchar	Representa la comprobación de que se realizaron los cambios correspondientes

❖ Horario

- http://127.0.0.1:5000/delete_horario

- request y response
- Donde se va a eliminar el horario con el id especificado

❖ Errores_posibles

Error	Descripción
400 Bad Request	Alguno de los campos que fueron ingresados no existen

❖ Asistencia

➤ Crear Asistencia

- http://127.0.0.1:5000/create_asistencia
- request y response
- Donde se va a eliminar la seccion con el id especificado

❖ Fields

Field	Type	Required?	Description
fecha	varchar	required	Es la fecha
id_horario	int	required	Es el identificador
vector	varchar	required	Es el vector
foto	varchar	required	Es la foto
dni	varchar	required	Es el dni

❖ Post Objects:

Field	Type	Description
id_asistencia	int	Representa el id que se le asigna luego de la inserción en la tabla asistencia

message	varchar	Representa la comprobación de que se ingreso correctamente los datos en la tabla asistencia
---------	---------	---

❖ Errores_posibles

Error	Descripción
400 Bad Request	Alguno de los campos que fueron ingresados no existen

➤ **Listar Asistencia**

- http://127.0.0.1:5000/get_asistencia
- **response**
- **Donde se va a enlistar la asistencia**

❖ Post Objects:

Field	Type	Description
dni	varchar	Representa el id que se le asigna luego de la inserción en la tabla horario
estado	varchar	Representa la comprobación de que se ingreso correctamente los datos en la tabla horario
fecha	varchar	Representa la fecha en que se tomó la asistencia
foto	varchar	Representa la dirección de la foto que se usó para la asistencia
id_asistencia	int	Representa el id de la asistencia
id_horario	int	Representa el id del horario de la

		asistencia
vector_comp	varchar	Representa la distancia entre las 2 fotos

➤ **Actualizar Asistencia**

- http://127.0.0.1:5000/update_asistencia
- request y response
- Donde se va a actualizar la asistencia

➤ **Borrar Asistencia**

- http://127.0.0.1:5000/delete_asistencia
- request y response
- Donde se va a eliminar la asistencia con el id especificado

❖ Errores_posibles

Error	Descripción
400 Bad Request	Alguno de los campos que fueron ingresados no existen

❖ **Justificación**

➤ **Crear Justificación**

- http://127.0.0.1:5000/create_justificacion
- request y response
- Donde se va a eliminar la justificacion con el id especificado

❖ Fields

Field	Type	Required?	Description
fechar	varchar	required	Es el numero de horas
id_asistencia	varchar	required	Es la descripcion

❖ Post Objects:

Field	Type	Description
id_asistencia	int	Representa el id que se le asigna luego de la inserción en la tabla asistencia
message	varchar	Representa la comprobación de que se ingreso correctamente los datos en la tabla asistencia

❖ Errores_posibles

Error	Descripción
400 Bad Request	Alguno de los campos que fueron ingresados no existen

➤ Listar Justificación

- http://127.0.0.1:5000/get_justificacion
- response
- Donde se va a enlistar las justificaciones

❖ Post Objects:

Field	Type	Description
id_justificacion	int	Representa el identificador de la justificación
fechar	varchar	Representa la fecha de la justificacion
id_asistencia	int	identificador de la asistencia

➤ Actualizar Justificación

- http://127.0.0.1:5000/update_justificacion
- request y response
- Donde se va a actualizar la justificacion

❖ Fields

Field	Type	Required?	Description
id_justificacion	int	required	Es el identificador de justificacion
fechar	varchar	required	Es la fecha de la justificacion

❖ Patch Objects:

Field	Type	Description
result	int	Representa que se actualizó exitosamente la tabla curso

❖ Errores_posibles

Error	Descripción
400 Bad Request	Alguno de los campos que fueron ingresados no existen

➤ **Borrar Justificación**

❖ Errores_posibles

Error	Descripción
400 Bad Request	Alguno de los campos que fueron ingresados no existen

❖ Participación

➤ Crear Participación

- http://127.0.0.1:5000/create_seccion
- request y response
- Donde se va a crear la participacion con el id especificado

❖ Fields

Field	Type	Required?	Description
fecha	varchar	required	Es la fecha
participacion	varchar	required	Es la participacion
id_horario	varchar	required	Es el horario

❖ Post Objects:

Field	Type	Description
id_horario	int	Representa el id que se le asigna luego de la inserción en la tabla horario
message	varchar	Representa la comprobación de que se ingreso correctamente los datos en la tabla horario

❖ Errores_posibles

Error	Descripción
400 Bad Request	Alguno de los campos que fueron ingresados no existen

➤ **Listar Participación**

❖ Post Objects:

Field	Type	Description
fecha	varchar	Representa la fecha de la participación
id_horario	int	Representa el id del horario en que se tomo la participación
id_participacion	int	Representa el id de la participación
participacion	varchar	Representa la descripción de la participación

➤ **Actualizar Participación**

❖ Fields

Field	Type	Required?	Description
fecha	int	required	Es la fecha
participacion	varchar	required	Es el nombre de la participacion
id_horario	int	required	Es el id de horario

❖ Patch Objects:

Field	Type	Description
result	int	Representa que se actualizó exitosamente la tabla curso

❖ Errores_posibles

Error	Descripción
400 Bad Request	Alguno de los campos que fueron ingresados no existen

➤ **Borrar Participación**

❖ Errores_posibles

Error	Descripción
400 Bad Request	Alguno de los campos que fueron ingresados no existen

❖ Usuario

➤ **Crear Usuario**

- http://127.0.0.1:5000/create_user
- request y response
- Donde se va a crear un usuario con el id especificado

❖ Fields

Field	Type	Required?	Description
nickname	varchar	required	Es el nickname
password	varchar	required	Es la password
nombre	varchar	required	Es el nombre
apellido	varchar	required	Es el apellido
edad	varchar	required	Es la edad

genero	varchar	required	Es el genero
correo_electronico	varchar	required	Es el correo
telefono	varchar	required	Es el telefono
foto	varchar	required	Es la foto
vector	varchar	required	Es el vector de la foto

❖ Post Objects:

Field	Type	Description
id_usuario	int	Representa el id que se le asigna luego de la inserción en la tabla usuario
message	varchar	Representa la comprobación de que se ingreso correctamente la insercion

❖ Errores_posibles

Error	Descripción
400 Bad Request	Alguno de los campos que fueron ingresados no existen

➤ Listar Usuario

- http://127.0.0.1:5000/get_user
- response
- Donde se va a enlistar el usuario

❖ Post Objects:

Field	Type	Description
-------	------	-------------

id_usuario	int	Representa el id del usuario
apellido	varchar	Representa el apellido del usuario
correo	varchar	Representa el correo del usuario
direccion	varchar	Representa la dirección del usuario
dni	varchar	Representa el dni del usuario
edad	int	Representa la edad del usuario
foto	varchar	Representa la dirección de foto del usuario
genero	varchar	Representa el género del usuario
nickname	varchar	Representa el nickname del usuario
nombre	varchar	Representa el nombre del usuario
password	varchar	Representa el password del usuario
telefono	varchar	Representa el telefono del usuario
vector	varchar	Representa el vector de la foto del usuario

➤ Actualizar Usuario

- http://127.0.0.1:5000/update_user
- request y response
- Donde se va a actualizar el usuario con el id especificado

❖ Fields

Field	Type	Required?	Description
id_usuario	int	required	Es el identificador del usuario
nickname	varchar	required	Es el nick del usuario
password	varchar	required	Es el password del usuario

nombre	varchar	required	Es el nombre del usuario
apellido	varchar	required	Es el apellido del usuario
edad	int	required	Es la edad del usuario
genero	varchar	required	Es el género del usuario
corre_electronic o	varchar	required	Es el correo del usuario
telefono	varchar	required	Es el teléfono del usuario
direccion	varchar	required	Es la dirección del usuario
vector	varchar	required	Es el vector de la foto del usuario, que se obtiene al consumir la api de openface
foto	file	required	Es la foto del usuario

❖ Patch Objects:

Field	Type	Description
id_usuario	int	Representación de la creación del identificador del usuario
nickname	varchar	Representación de la creación del nick del usuario
password	varchar	Representación de la creación del password del usuario
nombre	varchar	Representación de la creación

		del nombre del usuario
apellido	varchar	Representación de la creación del apellido del usuario
edad	int	Representación de la creación de la edad del usuario
genero	varchar	Representación de la creación del género del usuario
corre_electronico	varchar	Representación de la creación del correo del usuario
telefono	varchar	Representación de la creación del teléfono del usuario
direccion	varchar	Representación de la creación de la dirección del usuario
vector	varchar	Representación de la creación del vector de la foto del usuario
foto	file	Representación de la creación de la foto del usuario

❖ Errores_posibles

Error	Descripción
400 Bad Request	Alguno de los campos que fueron ingresados no existen

➤ Borrar Usuario

- http://127.0.0.1:5000/delete_user
- request y response
- Donde se va a eliminar la usuario con el id especificado

❖ Errores_posibles

Error	Descripción
400 Bad Request	Alguno de los campos que fueron ingresados no existen

❖ Alumno

➤ Crear Alumno

- http://127.0.0.1:5000/create_alumno
- request y response
- Donde se va a crear un alumno con el id especificado

❖ Fields

Field	Type	Required?	Description
num_horas	varchar	required	Es el numero de horas
descripcion	varchar	required	Es la descripcion
hora_inicio	varchar	required	Es la hora de inicio
hora_fin	varchar	required	Es la hora de finalizacion
dia_semana	varchar	required	Es el dia de la semana
id_seccion	int	required	Es el id de la seccion

❖ Post Objects:

Field	Type	Description
-------	------	-------------

id_horario	int	Representa el id que se le asigna luego de la inserción en la tabla horario
message	varchar	Representa la comprobación de que se ingreso correctamente los datos en la tabla horario

❖ Errores_posibles

Error	Descripción
400 Bad Request	Alguno de los campos que fueron ingresados no existen

➤ Listar Alumno

- http://127.0.0.1:5000/get_alumno
- response
- Donde se va a enlista los alumnos

❖ Post Objects:

Field	Type	Description
id_alumno	int	Representa el identificador del alumno
carrera	varchar	Representa la carrera que estudia el alumno

➤ Actualizar Alumno

- http://127.0.0.1:5000/update_alumno
- request y response
- Donde se va a actualizar el alumno

❖ Fields

Field	Type	Required?	Description
-------	------	-----------	-------------

id_alumno	int	required	Es el identificador de alumno
carrera	varchar	required	Es el nombre de la carrera

❖ Patch Objects:

Field	Type	Description
result	int	Representa que se actualizó exitosamente la tabla curso

❖ Errores_posibles

Error	Descripción
400 Bad Request	Alguno de los campos que fueron ingresados no existen

➤ **Borrar Alumno**

- http://127.0.0.1:5000/delete_alumno
- request y response
- Donde se va a borrar el alumno con el id especificado

❖ Errores_posibles

Error	Descripción
400 Bad Request	Alguno de los campos que fueron ingresados no existen

❖ Profesor

➤ Crear Profesor

- http://127.0.0.1:5000/create_profesor
- request y response
- Donde se va a crear un profesor con el id especificado

❖ Fields

Field	Type	Required?	Description
num_horas	varchar	required	Es el numero de horas
descripcion	varchar	required	Es la descripcion
hora_inicio	varchar	required	Es la hora de inicio
hora_fin	varchar	required	Es la hora de finalizacion
dia_semana	varchar	required	Es el dia de la semana
id_seccion	int	required	Es el id de la seccion

❖ Post Objects:

Field	Type	Description
id_horario	int	Representa el id que se le asigna luego de la inserción en la tabla horario
message	varchar	Representa la comprobación de que se ingreso correctamente los datos en la tabla horario

❖ Errores_posibles

Error	Descripción
400 Bad Request	Alguno de los campos que fueron ingresados no existen

➤ Listar Profesor

- http://127.0.0.1:5000/get_profesor
- response
- Donde se va a listar el profesor

❖ Post Objects:

Field	Type	Description
id_profesor	int	Representa el identificador del profesor
departamento	varchar	Representa el departamento que proviene el profesor

➤ Actualizar Profesor

- http://127.0.0.1:5000/update_profesor
- request y response
- Donde se va a eliminar el profesor con el id especificado

❖ Fields

Field	Type	Required?	Description
id_profesor	int	required	Es el identificador del profesor
departamento	varchar	required	Es el nombre del departamento

❖ Patch Objects:

Field	Type	Description
result	int	Representa que se actualizó exitosamente la tabla curso

❖ Errores_posibles

Error	Descripción
400 Bad Request	Alguno de los campos que fueron ingresados no existen

➤ **Borrar Profesor**

❖ Errores_posibles

Error	Descripción
400 Bad Request	Alguno de los campos que fueron ingresados no existen

❖ Seccion_Profesor

➤ **Crear Seccion_Profesor**

- http://127.0.0.1:5000/create_seccion_profesor
- request y response
- Donde se va a crear la seccion_profesor tabla que intermedia de seccion y profesor con el id especificado

❖ Fields

Field	Type	Required?	Description
num_horas	varchar	required	Es el numero de horas
descripcion	varchar	required	Es la descripcion
hora_inicio	varchar	required	Es la hora de inicio
hora_fin	varchar	required	Es la hora de finalizacion
dia_semana	varchar	required	Es el dia de la semana
id_seccion	int	required	Es el id de la seccion

❖ Post Objects:

Field	Type	Description
id_horario	int	Representa el id que se le asigna luego de la inserción en la tabla horario
message	varchar	Representa la comprobación de que se ingreso correctamente los datos en la tabla horario

❖ Errores_posibles

Error	Descripción
400 Bad Request	Alguno de los campos que fueron ingresados no existen

➤ **Listar Seccion_Profesor**

- http://127.0.0.1:5000/get_seccion_profesor
- response
- Donde se va a eliminar la seccion con el id especificado

❖ Post Objects:

Field	Type	Description
id_seccion_profesor	int	Representa el id de la sección de un profesor
id_seccion	int	Representa el id de la sección a la que pertenece el profesor
id_profesor	int	Representa el id del profesor

➤ **Actualizar Seccion_Profesor**

- http://127.0.0.1:5000/update_seccion_profesor
- request y response
- Donde se va a actualizar la seccion_profesor

❖ Fields

Field	Type	Required?	Description
id_seccion_profesor	int	required	Es el identificador de seccion profesor
id_seccion	int	required	Es el identificacion de la seccion
id_profesor	int	required	Es la identificacion del profesor

❖ Patch Objects:

Field	Type	Description
-------	------	-------------

result	int	Representa que se actualizó exitosamente la tabla curso
--------	-----	---

❖ Errores_posibles

Error	Descripción
400 Bad Request	Alguno de los campos que fueron ingresados no existen

➤ **Borrar Seccion_Profesor**

- http://127.0.0.1:5000/delete_seccion_profesor
- request y response
- Donde se va a eliminar la seccion_profesor con el id especificado

❖ Errores_posibles

Error	Descripción
400 Bad Request	Alguno de los campos que fueron ingresados no existen

❖ **Alumno _Seccion**

➤ **Crear Alumno_Seccion**

- http://127.0.0.1:5000/create_alumno_seccion
- request y response
- Donde se va a eliminar la seccion con el id especificado

❖ Fields

Field	Type	Required?	Description
id_alumno	varchar	required	Es el numero de horas
descripcion	varchar	required	Es la descripcion
hora_inicio	varchar	required	Es la hora de inicio
hora_fin	varchar	required	Es la hora de finalizacion
dia_semana	varchar	required	Es el dia de la semana
id_seccion	int	required	Es el id de la seccion

❖ Post Objects:

Field	Type	Description
id_horario	int	Representa el id que se le asigna luego de la inserción en la tabla horario
message	varchar	Representa la comprobación de que se ingreso correctamente los datos en la tabla horario

❖ Errores_posibles

Error	Descripción
400 Bad Request	Alguno de los campos que fueron ingresados no existen

➤ **Listar Alumno_Seccion**

- http://127.0.0.1:5000/get_alumno_seccion
- response
- Donde se va a enlistar alumno_seccion

❖ Post Objects:

Field	Type	Description
id_alumno_seccion	int	Representa el id de la sección de un alumno
id_alumno	int	Representa el id del alumno que pertenece a la sección
id_seccion	int	Representa el id de la seccion

➤ Actualizar Alumno_Seccion

- http://127.0.0.1:5000/update_alumno_seccion
- request y response
- Donde se va a actualizar alumno_seccion

❖ Fields

Field	Type	Required?	Description
id_alumno_seccion	int	required	Es el identificador de alumno_seccion
id_alumno	int	required	Es el identificacion de alumno
id_seccion	int	required	Es el identificador de seccion

❖ Patch Objects:

Field	Type	Description
result	int	Representa que se actualizó exitosamente la tabla alumno_seccion

❖ Errores_posibles

Error	Descripción
400 Bad Request	Alguno de los campos que fueron ingresados no existen

➤ **Borrar Alumno_Seccion**

- http://127.0.0.1:5000/delete_alumno_seccion
- request y response
- Donde se va a eliminar el alumno_seccion con el id especificado

❖ Errores_posibles

Error	Descripción
400 Bad Request	Alguno de los campos que fueron ingresados no existen

RESULTADOS:

USUARIOS:

THUNDER CLIENT

New Request

Activity Collections Env

filter collections

asistencias_db_user_request

- POST get_users 1 hour ago
- POST get_user 15 days ago
- PUP create_user 26 mins ago
- DEL delete_user 1 hour ago
- PATCH update_user 1 hour ago

asistencias_proyect_curso_requests

asistencias_proyect_seccion_requests

asistencias_proyect_alumnos_requests

asistencias_proyect_profesores_request

asistencias_proyect_horario_requests

POST http://127.0.0.1:5050/users Send

Query Headers 2 Auth Body Tests Pre Run

Json Xml Text Form Form-encode GraphQL Binary

Json Content

1

Format

Status: 200 OK Size: 32.26 KB Time: 243 ms

Response Headers 3 Cookies Results Docs {}

```
1 [
2 {
3   "apellido": "paraaedes",
4   "correo": "asssdasd",
5   "direccion": "sasssdadasdasd",
6   "dni": null,
7   "edad": 216,
8   "foto": null,
9   "genero": "ms",
10  "id_usuario": 2,
11  "nickname": "patol122",
12  "nombre": "patricsssk",
13  "password": "12233",
14  "telefono": "5145454",
15  "vector_foto": null
16 },
17 {
18   "apellido": "paredaaes",
19   "correo": "assssasd",
20   "direccion": "saaaadasdasd",
21   "dni": null,
22   "edad": 26,
23   "foto": null,
24   "genero": "mc",
25   "id_usuario": 3,
26   "nickname": "pato3",
27   "nombre": "patssrick",
28   "password": "12433",
```

THUNDER CLIENT

New Request

Activity Collections Env

filter collections

asistencias_db_user_request

- POST get_users 1 hour ago
- POST get_user 15 days ago
- PUP create_user 26 mins ago
- DEL delete_user 1 hour ago
- PATCH update_user 1 hour ago

asistencias_proyect_curso_requests

asistencias_proyect_seccion_requests

asistencias_proyect_alumnos_requests

asistencias_proyect_profesores_request

asistencias_proyect_horario_requests

POST http://127.0.0.1:5050/user Send

Query Headers 2 Auth Body 1 Tests Pre Run

Json Xml Text Form Form-encode GraphQL Binary

Json Content

```
1 {
2   "id_usuario":1
3 }
```

Format

Status: 200 OK Size: 324 Bytes Time: 79 ms

Response Headers 3 Cookies Results Docs {}

```
1 {
2 {
3   "apellido": "paredes",
4   "correo": "asdasd",
5   "direccion": "sadasdasd",
6   "dni": null,
7   "edad": 26,
8   "foto": "fotol",
9   "genero": "m",
10  "id_usuario": 1,
11  "nickname": "pato",
12  "nombre": "patrick",
13  "password": "1233",
14  "telefono": "545454",
15  "vector_foto": null
16 }
17 }
```

THUNDER CLIENT

New Request

Activity Collections Env

filter collections

asistencias_db_user_request

POST get_users 1 hour ago

POST get_user 15 days ago

PUT create_user just now

DEL delete_user 1 hour ago

PATCH update_user 1 hour ago

asistencias_proyect_curso_requests

asistencias_proyect_seccion_requests

asistencias_proyect_alumnos_requests

asistencias_proyect_profesores_request

asistencias_proyect_horario_requests

PUT http://127.0.0.1:5050/user Send

Query Headers 2 Auth Body 1 Tests Pre Run

Json Xml Text Form Form-encode GraphQL Binary

apellido mamani

genero m

correo_electronico correoluis

telefono telaluis

direccion dir1

dni 75575441

edad 21

field name value

Files

foto Choose File file2.png

field name Choose File Select file

Status: 200 OK Size: 1.8 KB Time: 3.62 s

Response Headers 5 Cookies Results Docs {}

```
1 {
2   "apellido": "mamani",
3   "correo_electronico": "correoluis",
4   "direccion": "dir1",
5   "dni": "75575441",
6   "edad": "21",
7   "genero": "m",
8   "nickname": "luis201",
9   "nombre": "luis",
10  "password": "secreto",
11  "telefono": "telaluis",
12  "vector": "[-0.11292028 -0.06097206 -0.05607069
0.00111997 0.00834676 0.06708492\n -0
.0894036 0.07954498 0.04564694 -0.01467148
0.03628109 -0.02179129\n 0.0251426 0
.0491619 0.10369043 0.03022784 0.06532107
-0.06391694\n 0.09843206 -0.00510458 -0
.00258851 -0.13006359 0.1763816 -0
.00942208\n 0.09396623 -0.00418245 0
.0945325 -0.11552594 0.00132898 -0
.16068751\n 0.25385582 -0.07114045 0
.16395135 -0.043857 -0.04449619 0
.11143877\n 0.00443648 0.11425914 -0
.07100575 -0.01055785 0.07958804 -0
.02399931\n -0.05355907 0.05052274 -0
.11888666 -0.12084354 0.0217115 -0
.07608157\n 0.00959319 -0.09343958 -0.175861
-0.16504183 0.0828352 -0.06299359\n -0
```

THUNDER CLIENT

New Request

Activity Collections Env

filter collections

asistencias_db_user_request

POST get_users 1 hour ago

POST get_user 15 days ago

PUT create_user just now

DEL delete_user just now

PATCH update_user 1 hour ago

asistencias_proyect_curso_requests

asistencias_proyect_seccion_requests

asistencias_proyect_alumnos_requests

asistencias_proyect_profesores_request

asistencias_proyect_horario_requests

DELETE http://127.0.0.1:5050/user Send

Query Headers 2 Auth Body 1 Tests Pre Run

Json Xml Text Form Form-encode GraphQL Binary

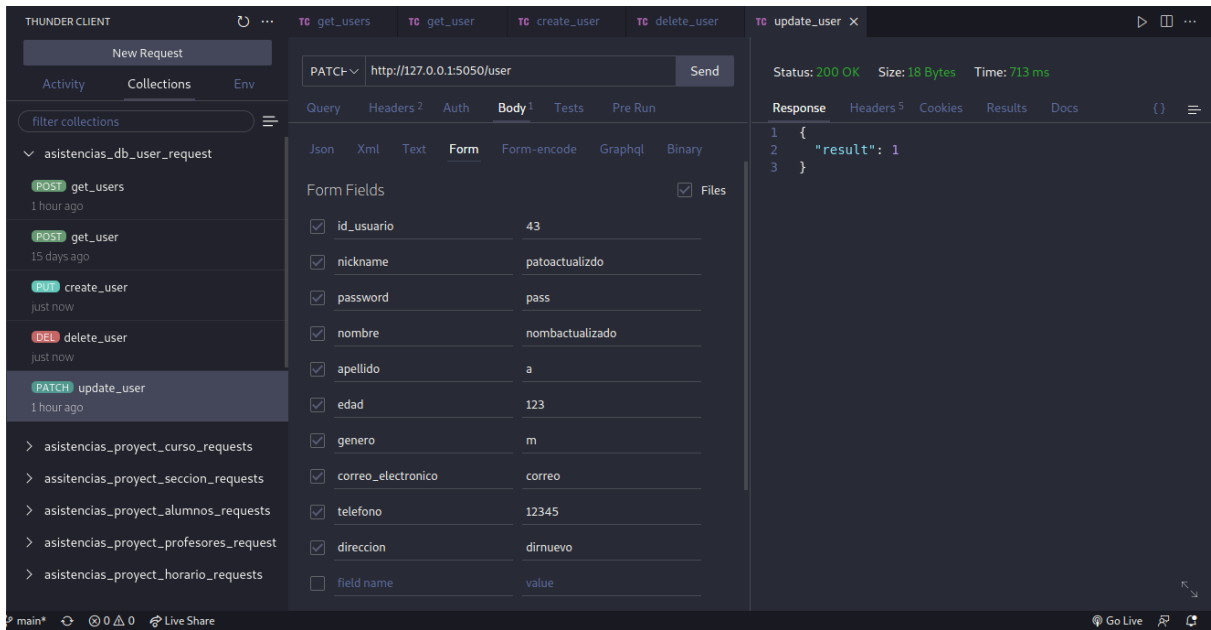
Json Content Format

```
1 {
2   "id_usuario" : 31
3 }
```

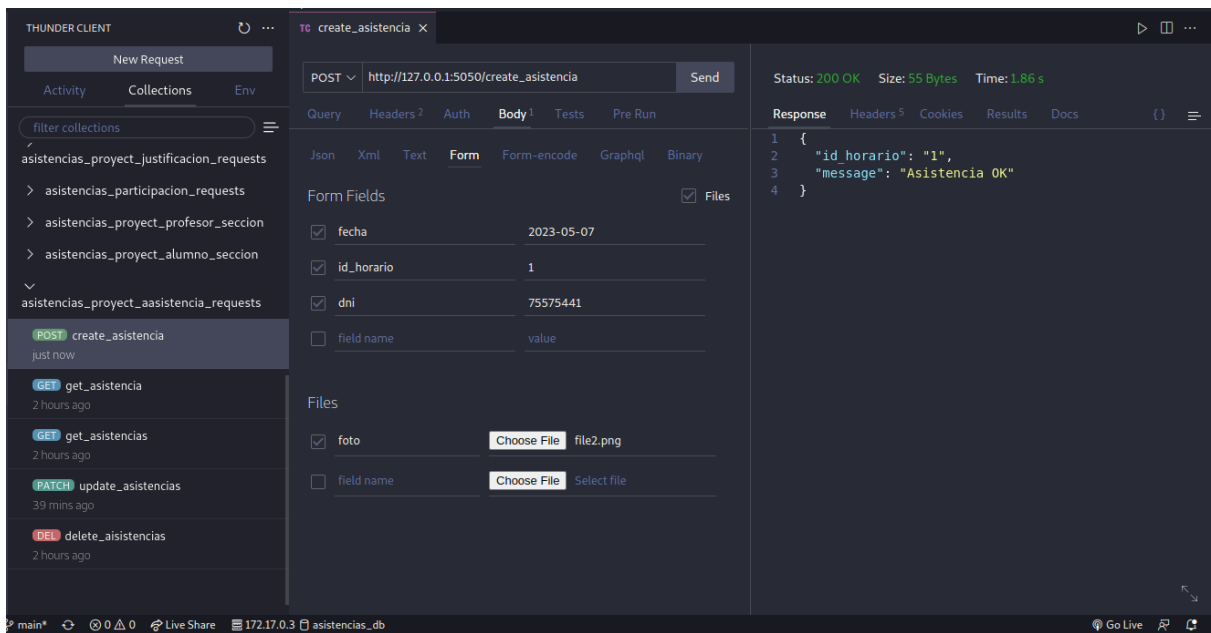
Status: 200 OK Size: 18 Bytes Time: 23 ms

Response Headers 5 Cookies Results Docs {}

```
1 {
2   "result": 1
3 }
```



Asistencias:



THUNDER CLIENT

TC create_asistencia TC get_asistencia X

New Request

Activity Collections Env

filter collections

- asistencias_project_justificacion_requests
- > asistencias_participacion_requests
- > asistencias_project_profesor_seccion
- > asistencias_project_alumno_seccion
- asistencias_project_aasistencia_requests
 - POST create_asistencia just now
 - GET get_asistencia just now
 - GET get_asistencias 2 hours ago
 - PATCH update_asistencias 39 mins ago
 - DEL delete_asistencias 2 hours ago

Query Headers Auth Body Tests Pre Run

Json Xml Text Form Form-encode GraphQL Binary

Form Fields

<input checked="" type="checkbox"/>	id_asistencia	5
<input type="checkbox"/>	field name	value

Files

Status: 200 OK Size: 314 Bytes Time: 28 ms

Response Headers Cookies Results Docs

```
1 [
2 {
3   "dni": "75575456",
4   "estado": false,
5   "fecha": "Tue, 09 May 2023 00:00:00 GMT",
6   "foto": "/home/luisfelipe/Proyectos
   /construccion_Software/proyecto_final
   /photos_asistencia/75575456file2.png",
7   "id_asistencia": 5,
8   "id_horario": 2,
9   "vector_comparacion": "1.382154326253344"
10 }
11 ]
```

main* 0 0 0 Live Share 172.17.0.3 asistencias_db Go Live

THUNDER CLIENT

TC create_asistencia TC get_asistencia TC get_asistencias X

New Request

Activity Collections Env

filter collections

- asistencias_project_justificacion_requests
- > asistencias_participacion_requests
- > asistencias_project_profesor_seccion
- > asistencias_project_alumno_seccion
- asistencias_project_aasistencia_requests
 - POST create_asistencia just now
 - GET get_asistencia just now
 - GET get_asistencias 2 hours ago
 - PATCH update_asistencias 39 mins ago
 - DEL delete_asistencias 2 hours ago

Query Headers Auth Body Tests Pre Run

Query Parameters

<input type="checkbox"/>	parameter	value
--------------------------	-----------	-------

Status: 200 OK Size: 3.88 KB Time: 27 ms

Response Headers Cookies Results Docs

```
1 [
2 {
3   "dni": "12345679",
4   "estado": false,
5   "fecha": "Mon, 01 May 2023 00:00:00 GMT",
6   "foto": "/home/luisfelipe/Proyectos
   /construccion_Software/proyecto_final
   /photos_asistencia
   /nuevos847ea22cef1014c0b5e4833.png",
7   "id_asistencia": 2,
8   "id_horario": 1,
9   "vector_comparacion": "1"
10 },
11 {
12   "dni": "75575456",
13   "estado": false,
14   "fecha": "Tue, 09 May 2023 00:00:00 GMT",
15   "foto": "/home/luisfelipe/Proyectos
   /construccion_Software/proyecto_final
   /photos_asistencia/75575456file2.png",
16   "id_asistencia": 5,
17   "id_horario": 2,
18   "vector_comparacion": "1.382154326253344"
19 },
20 {
21   "dni": "75575456",
22   "estado": false,
23   "fecha": "Thu, 04 May 2023 00:00:00 GMT",
```

main* 0 0 0 Live Share 172.17.0.3 asistencias_db Go Live

THUNDER CLIENT

TC create_asistencia TC get_asistencia TC get_asistencias TC update_asistencias x

New Request

Activity Collections Env

filter collections

- asistencias_project_justificacion_requests
- > asistencias_participacion_requests
- > asistencias_project_profesor_seccion
- > asistencias_project_alumno_seccion
- asistencias_project_aasistencia_requests
 - POST create_asistencia just now
 - GET get_asistencia just now
 - GET get_asistencias 2 hours ago
 - PATCH update_asistencias 40 mins ago
 - DEL delete_asistencias 2 hours ago

PATCH+ http://127.0.0.1:5050/update_asistencia Send

Query Headers 2 Auth Body 1 Tests Pre Run

Json Xml Text Form Form-encode GraphQL Binary

Form Fields

☒ id_asistencia 3

☒ estado true

☒ fecha 2023-05-01

☒ id_horario 1

☐ field name value

Files

☒ foto Choose File 5847ea22cef1014c0b5e4f

☐ field name Choose File Select file

Status: 200 OK Size: 18 Bytes Time: 41 ms

Response Headers 5 Cookies Results Docs {}

```
1 {
2   "result": 1
3 }
```

main* 0 0 0 Live Share 172.17.0.3 asistencias_db Go Live

THUNDER CLIENT

TC create_asistencia TC get_asistencia TC get_asistencias TC update_asistencias TC delete_asistencias x

New Request

Activity Collections Env

filter collections

- asistencias_project_justificacion_requests
- > asistencias_participacion_requests
- > asistencias_project_profesor_seccion
- > asistencias_project_alumno_seccion
- asistencias_project_aasistencia_requests
 - POST create_asistencia just now
 - GET get_asistencia just now
 - GET get_asistencias 2 hours ago
 - PATCH update_asistencias 40 mins ago
 - DEL delete_asistencias just now

DELETE http://127.0.0.1:5050/delete_asistencia Send

Query Headers 2 Auth Body 1 Tests Pre Run

Json Xml Text Form Form-encode GraphQL Binary

Json Content

```
1 {
2   "id_asistencia":4
3 }
```

Status: 200 OK Size: 18 Bytes Time: 27 ms

Response Headers 5 Cookies Results Docs {}

```
1 {
2   "result": 1
3 }
```

main* 0 0 0 Live Share 172.17.0.3 asistencias_db Go Live