

# DOCUMENTACIÓN DEL PROYECTO

## DESCRIPCIÓN

El proyecto PRINCE\_PROJECT consta de una serie de end-points que permiten el acceso a algunas funcionalidades listadas a continuación.

- ASIGNATURAS: (Adicionar, Listar, Modificar, Eliminar)
- LUGARES: (Adicionar, Listar, Modificar, Eliminar)
- GRUPOS: (Adicionar, Listar, Modificar, Eliminar)
  
- CONSULTAS EXTRAS:
  - Asignaturas que tienen asignado un mismo lugar.
  - Asignaturas con más de un grupo.
  - Asignaturas que tiene asignado el mismo horario.

## EJECUCION DEL PROGRAMA:

Para la correcta ejecución del programa se recomienda tener instalado en la maquina Java 17.

Para la ejecución del programa siga estos pasos:

1. Ingrese a la carpeta "PRINCE\_PROJECT\_2V"
2. Acceda a la carpeta "EJECUTABLE"
3. En la barra de direcciones escriba "cmd" para abrir una ventana del símbolo del sistema de Windows, a continuación, copie la siguiente línea "java -jar PRINCE\_PROJECT\_2V-0.0.1-SNAPSHOT.jar"

De está manera el programa empezara a ejecutarse de manera local en el puerto 8080.

## ACCESO A LOS END-POINTS:

Para la correcta ejecución de los end-points se recomienda tener instalado el software POSTMAN (aplicación que permite probar API web).

### Pasos para Probar los Endpoints en Postman:

1. **Abrir Postman:** Inicie la aplicación de Postman en su máquina.
2. **Seleccionar el Método y la URL:**
  - En la parte superior izquierda de la ventana de Postman, seleccione el método HTTP adecuado (GET, POST, DELETE, PUT.) según sea el caso.  
  
***GET:*** Listar y obtener consultas.  
***POST:*** Adicionar.  
***DELETE:*** Eliminar.  
***PUT:*** Modificar.
  - En el campo de URL, ingrese la dirección del servidor seguida del end-point específico que deseas probar. Por ejemplo:  
<http://localhost:8080/subjects/add>.
3. **Agregar Parámetros** (si es necesario):
  - Si el endpoint requiere parámetros, puede especificarlos en la sección adecuada de la ventana de Postman. Por ejemplo, para un método POST, puede agregar un cuerpo JSON en la pestaña "Body" subsección "raw".
4. **Enviar la Solicitud:**
  - Una vez configurados los detalles de la solicitud, haga clic en el botón "Send" (Enviar) para enviar la solicitud al servidor.

## 5. Observar la Respuesta:

- Postman mostrará la respuesta del servidor en la parte inferior de la ventana. Aquí puede ver el código de estado HTTP, y el cuerpo de la respuesta.

## EJEMPLOS DE ACCESO A END-POINTS:

### ASIGNATURAS:

#### 1. Agregar Asignatura

Método: POST

URL: `http://localhost:8080/subjects/add`

Descripción: Agrega una nueva asignatura al sistema.

Ejemplo de Cuerpo:

```
{  
  "name": "Cálculo 1",  
  "code": "CALC001"  
}
```

#### 2. Listar Asignaturas

Método: GET

URL: `http://localhost:8080/subjects/getAll`

Descripción: Listar todas las asignaturas.

#### 3. Eliminar Asignatura

Método: DELETE

URL: `http://localhost:8080/subjects/delete/{id}`

Descripción: Eliminar una asignatura.

Ejemplo de URL: `http://localhost:8080/subjects/delete/CALC001`

#### 4. Modificar Asignatura

Método: PUT

URL: `http://localhost:8080/subjects/update/{code}`

Descripción: Modificar una asignatura.

Ejemplo de URL: `http://localhost:8080/subjects/update/CALC001`

Ejemplo de Cuerpo:

```
{
  "name": "Cálculo 2",
  "code": "CALC001"
}
```

#### LUGARES:

##### 1. Agregar Lugar

Método: POST

URL: `http://localhost:8080/places/add`

Descripción: Agrega un nuevo lugar al sistema.

Ejemplo de Cuerpo:

```
{
  "id": "C123",
  "name": "Aula 123",
  "location": "Edificio Central, Piso 1"
}
```

##### 2. Listar Lugares

Método: GET

URL: `http://localhost:8080/places/getAll`

Descripción: Listar todos los lugares.

##### 3. Eliminar Lugar

Método: DELETE

URL: `http://localhost:8080/places/delete/{id}`

Descripción: Eliminar un lugar.

Ejemplo de URL: `http://localhost:8080/places/delete/C123`

#### 4. Modificar Lugar

Método: PUT

URL: `http://localhost:8080/places/update/{code}`

Descripción: Modificar un lugar.

Ejemplo de URL: `http://localhost:8080/places/update/C123`

Ejemplo de Cuerpo:

```
{
  "id": "C123",
  "name": "Aula 123",
  "location": "Edificio Rafael Azula, Piso 4"
}
```

### GRUPOS:

#### 1. Agregar Grupo

Método: POST

URL: `http://localhost:8080/groups/add`

Descripción: Agrega un nuevo grupo al sistema.

Ejemplo de Cuerpo:

```
{
  "id": "G001",
  "subjectCode": "C120241",
  "placeId": "A123",
  "schedules": ["Lunes 8:00-10:00", "Miércoles 10:00-12:00"]
}
```

#### 2. Listar Grupos

Método: GET

URL: `http://localhost:8080/groups/getAll`

Descripción: Listar todos los lugares.

### 3. Eliminar Grupo

Método: DELETE

URL: `http://localhost:8080/groups/delete/{id}`

Descripción: Eliminar un lugar.

Ejemplo de URL: `http://localhost:8080/groups/delete/G001`

### 4. Modificar Lugar

Método: PUT

URL: `http://localhost:8080/groups/update/{code}`

Descripción: Modificar un lugar.

Ejemplo de URL: `http://localhost:8080/groups/update/G001`

Ejemplo de Cuerpo:

```
{
  "id": "G001",
  "subjectCode": "C120241",
  "placeId": "A24",
  "schedules": ["Martes 8:00-10:00", "Jueves 10:00-12:00"]
}
```

### CONSULTAS:

#### 1. Asignaturas que tienen asignado un mismo lugar.

Método: GET

URL: `http://localhost:8080/groups/samePlaceQuery/{placeId}`

Descripción: Listar todas las asignaturas que tienen asignado el lugar que se manda como parámetro.

Ejemplo de URL: `http://localhost:8080/groups/samePlaceQuery/A123`

## **2. Asignaturas con más de un grupo.**

Método: GET

URL: <http://localhost:8080/groups/multipleGroups>

Descripción: Listar todas las asignaturas que tienen más de un grupo.

## **3. Asignaturas que tienen asignado un mismo lugar.**

Método: GET

URL: <http://localhost:8080/groups/sameScheduleQuery>

Descripción: Listar todas las asignaturas que tienen el horario que se manda como parámetro en el "body".

Ejemplo de Cuerpo:

["Lunes 8:00-10:00", "Miércoles 10:00-12:00"]

## **CONCLUSIONES:**

Este documento proporciona una guía para interactuar con los end-points de la API. Nuevamente se recomienda utilizar Postman u otra herramienta similar para probar y explorar la funcionalidad completa de la API.