**Plan de pruebas de software**

***[Rest API]***

***Fecha: [18/11/2022]***

**Tabla de contenido**

Historial de Versiones 4

Información del Proyecto 4

Aprobaciones 4

Resumen Ejecutivo 5

Alcance de las Pruebas 5

Elementos de Pruebas 5

Nuevas Funcionalidades a Probar 6

Pruebas de Regresión 6

Funcionalidades a No Probar 7

Enfoque de Pruebas (Estrategia) 7

Criterios de Aceptación o Rechazo 8

Criterios de Aceptación o Rechazo 8

Criterios de Suspensión 8

Criterios de Reanudación 9

Entregables 9

Recursos 10

Requerimientos de Entornos – Hardware 10

Requerimientos de Entornos – Software 10

Herramientas de Pruebas Requeridas 11

Personal 11

Entrenamiento 12

Planificación y Organización 12

Procedimientos para las Pruebas 12

Matriz de Responsabilidades 13

Cronograma 13

Premisas 14

Dependencias y Riesgos 14

Referencias 15

Glosario 15

# Historial de versiones

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Fecha** | **Versión** | **Autor** | **Organización** | **Descripción** |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

# Información del proyecto

|  |  |
| --- | --- |
| Empresa / Organización | TdeA |
| Proyecto | Rest API node Js |
| Fecha de preparación |  |
| Cliente |  |
| Patrocinador principal |  |
| Gerente / Líder de proyecto |  |
| Gerente / Líder de pruebas de software |  |

# Aprobaciones

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Nombre y Apellido** | **Cargo** | **Departamento u organización** | **Fecha** | **Firma** |
| **Juan Campiño** | **Estudiante** | **TdeA** |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

# Resumen ejecutivo

Plan de pruebas realizado a una API rest hecha con javascript donde se probarán los métodos api esperando una buena respuesta del servidor y que los cambios se vean reflejados en la base de datos para concluir una prueba con resultados positivos y poder lanzar la API

# Alcance de las pruebas

## Elementos de pruebas

* Ruta de personas en sus métodos:
  + GET
  + POST
  + PUT
  + DELETE
* Ruta index en sus método Get

## Funcionalidades a no probar

* Ruta usuarios

## Enfoque de pruebas (estrategia)

Se realizará una prueba con una API llamada thunder client en VS donde se probaran las respuestas del servidor ante el llamado de las rutas y de los llamados de los métodos comprobando su correcto funcionamiento.

# Criterios de aceptación o rechazo

## Criterios de aceptación o rechazo

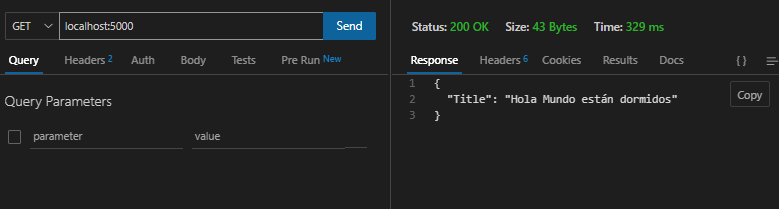
Para dar por completada esta prueba se deben realizar todos los métodos definidos en las rutas con respuesta del servidor HTTP 200 y que se vean reflejados los cambios en la base de datos.

## Criterios de suspensión

Esta prueba se suspende en el momento que algunas de las rutas devuelvan un resultado HTTP 400 o 500, enviando los datos y métodos correctamente, también se suspende cuando los métodos no afecten la base de datos.

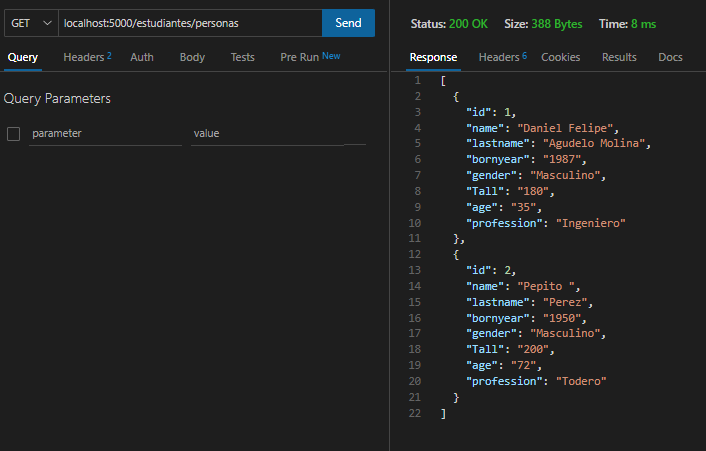
# Entregables

EL primer método a probar sería la raíz de la api:

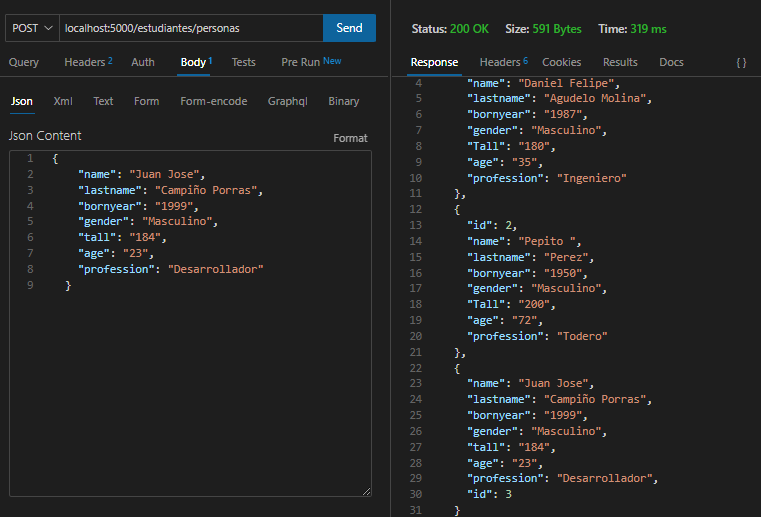




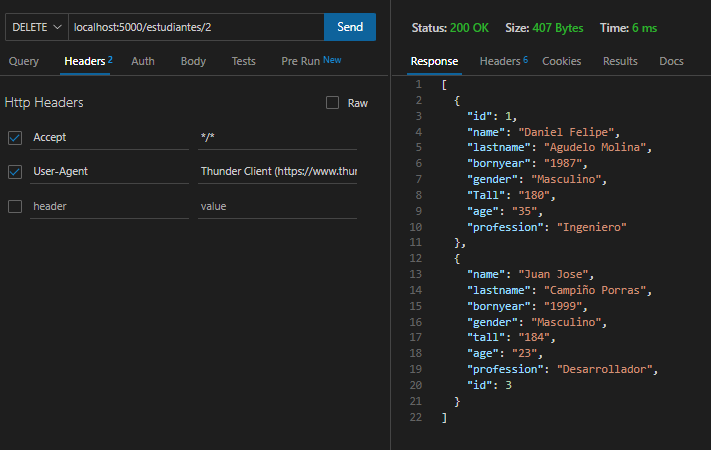
Vemos que la respuesta del servidor es 200 y el json devuelto es el especificado en el metodo



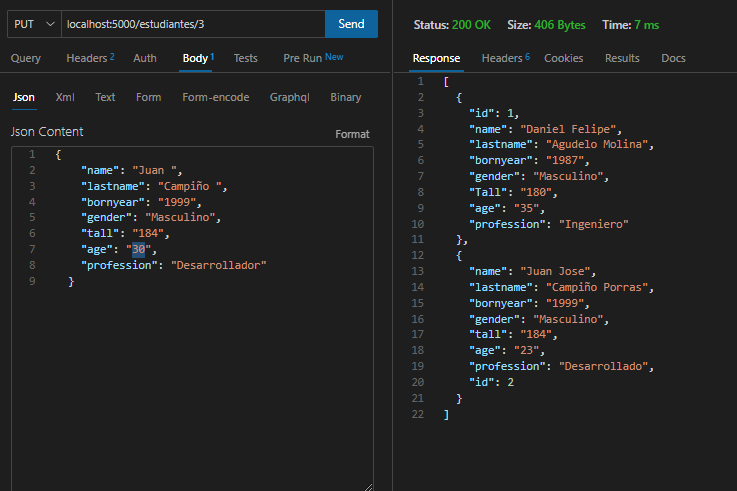
EL método Get en la ruta personas nos devuelve correctamente los elementos de la base de datos



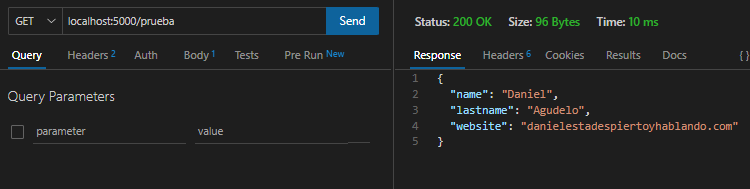
AL utilizar el método POST y enviar un objeto en el body teniendo en cuenta el nombre de las llaves del JSON, nos devuelve un status 200 y también nos retorna los elementos de la base de datos



Al cambiar la ruta a “localhost:5000/estudiantes/2” y ejecutar el método DELETE vemos que las respuesta es ok y nos devuelve los elementos de la base de datos y confirmamos que se eliminó el estudiante con el id 2



Al ejecutar el método PUT, realizando pequeños cambios para el estudiante con el id 3, vemos que se realizan correctamente y además se actualiza el id que antes era 3 y ya al solo tener 2 estudiantes queda con id 2



La última ruta es “localhost:5000/prueba” el cual nos devuelve un JSON que es creado en el mismo método y lo devuelve con status 200.

# Recursos

## Requerimientos de entornos – Hardware

* Computador

## Requerimientos de entornos – Software

Para realizar está prueba vamos a utilizar los siguientes componentes:

* Visual Studio Code
* Lenguaje JavaScript
* Navegador web

## Herramientas de pruebas requeridas

* API Thunder client

## Personal

Lista del personal necesario para completar las actividades de pruebas, especificando sus roles, por ejemplo: Un (1) líder de pruebas, cinco (5) analistas de pruebas (Testers), dos (2) especialistas en automatización de pruebas, entre otros.

## Entrenamiento

Necesidades de entrenamiento en el sistema o aplicación, así como en las herramientas de prueba a utilizar.

# Planificación y organización

## Procedimientos para las pruebas

Especifica los procedimientos o metodología de pruebas a emplear durante la ejecución del plan de pruebas de software.

## Matriz de responsabilidades

Lista cada una de las personas integrantes del equipo de QA y sus responsabilidades. Se puede hacer uso de una matriz RACI (responsable, aprobador, consultado, informado).

## Cronograma

Debe estar basado en estimaciones de actividades realizadas por el equipo de prueba. En él se identifican los hitos relevantes en las pruebas de software, se establecen las dependencias (actividades predecesoras) y demás aspectos componentes de un cronograma.

## Premisas

Las premisas relacionadas con las tareas de pruebas de software, incluyendo limitaciones de tiempo, disponibilidad de recursos que se asumen, uso de una metodología de pruebas, uso de una herramienta, entre otros.

## Dependencias y Riesgos

Aquí se listan los riesgos asociados con el proceso de pruebas de software, por ejemplo, algunas fuentes de riesgos suelen ser:

* Dependencias con desarrollos.
* Dependencias con otros proyectos.
* Disponibilidad de recursos.
* Restricciones de tiempo.
* Premisas que resulten no ser ciertas.

Los riesgos se pueden clasificar en función de su probabilidad e impacto, cada uno debe contemplar un plan de mitigación para evitar que ocurra o plan de contingencia cuando el riesgo no puede mitigarse y tiene que aceptarse.

# Referencias

Lista de todos los documentos que pueden citarse como apoyo o para ampliar el contenido del plan de pruebas. Algunos ejemplos de lo que se puede hacer referencia aquí son:

* Plan de proyecto.
* Especificaciones de requerimientos.
* Diseño general.
* Diseño detallado.
* Procedimientos y estándares de desarrollo.
* Procedimientos y estándares de pruebas.
* Metodologías, procedimientos y estándares corporativos.

# Glosario

Definiciones de términos usados en la documentación, y general sobre el área de pruebas.