**Especificación de Casos de Pruebas para el sistema The BookStore**

****

**José Ángel Ruiz Escalante**

**Jesús Andrei Torres Landero**

**David Abraham Paredes Coob**

**Julian Alejandro Perez Koo**

**Verificación y Validación de Software**

**M.I.T. Edwin Jesús León Bojorquez**

**Control de la documentación**

**Control de configuración**

| **Título** | Especificación de Casos de Pruebas. |
| --- | --- |
| **Referencia** | <https://github.com/JoseAngelRE/VerificacionYValidacion/tree/main/Plan_de_Pruebas> |
| **Autor** | José Ángel Ruiz Escalante |
| **Fecha** | 07/11/10 |

**Histórico de versiones**

| **Versión** | **Fecha** | **Estado** | **Responsable** | **Nombre del archivo** |
| --- | --- | --- | --- | --- |
| 1.0 | 1/11/22 | A | José Ángel Ruiz Escalante | Especificación de Casos de Pruebas.docx |
| 1.3 | 7/11/22 | A | José Ángel Ruiz Escalante | Especificación de Casos de Pruebas.docx |

Estado (B)orrador, (R)evisión, (A)probado

**Histórico de cambios**

| **Versión** | **Fecha** | **Cambios** |
| --- | --- | --- |
| 1.0 | 1/11/22 | Se realizó la primera versión de la Especificación de Casos de Pruebas. |
| 1.1 | 5/11/22 | Se realizó el formato para cada caso de prueba. |
| 1.2 | 6/11/22 | Se agregó la información de cada caso de prueba. |
| 1.3 | 7/11/22 | Se revisó el documento y se realizaron modificaciones en el formato. |

**Índice**

[**Identificador del documento**](#_7b2mzawfcywa) **3**

[**Casos de prueba**](#_vwud4fs5jab3) **3**

[Tabla 1. Caso de prueba CDP-001](#_u0clttspugja) 3

[Tabla 2. Caso de prueba CDP-002](#_x5wr3c9gx4eg) 4

[Tabla 3. Caso de prueba CDP-003](#_6fzvga8zs1od) 5

[Tabla 4. Caso de prueba CDP-004](#_u0vgvtjkewyi) 6

[Tabla 5. Caso de prueba CDP-005](#_lkssvbshj6fr) 7

[Tabla 6. Caso de prueba CDP-006](#_9fcbsdgsu5zu) 8

# Identificador del documento

VYV-ECP-2022-JAJD

# Casos de prueba

| Caso de prueba | *CDP-001* | | |
| --- | --- | --- | --- |
| Identificador de característica | *TBC-1: Registrarse en el sistema* | | |
| Función probar | *F-5: createNewUser* | | |
| Objetivo | *Validar que la petición de registro de usuario funciona correctamente y que el sistema manda el mensaje correcto en caso de éxito o fallo.* | | |
| Criterios de éxito | *Se envían todos los datos correctamente y el sistema envía un mensaje de éxito.* | | |
| Criterios de falla | *Se envían todos los datos correctamente y el sistema envía un mensaje de error.* | | |
| Precondiciones | *Se debió haber ingresado los datos en el formato de registro correspondiente.* | | |
| Necesidades para el caso de prueba | *Es necesario realizar esta prueba para asegurarse que la petición para registrar un usuario funciona correctamente.* | | |
| Autor | *José Ángel Ruiz Escalante* | | |
| Fecha de creación | *24/10/2022* | | |
| Entradas | **Salidas** | | |
| {  fullname: “Jose Ruiz”,  email: “[jose@example.com](mailto:jose@example.com)”,  password: “passwordtest1”  } | {  message: “Sign up successfully”,  data: {  id: 1  fullname: “Jose Ruiz”,  email: “[jose@example.com](mailto:jose@example.com)”,  password: “passwordtest1”  }  } | | |
| Flujo del caso de prueba | **No paso** | **Usuario del sistema** | **Sistema** |
| *1* | *Se le envía a la función un objeto tipo JSON con los datos del usuario correctos.* | *Devuelve un objeto tipo JSON con un mensaje de éxito y los datos del usuario que se registró.* |
| Post condiciones | No Aplica. | | |

## Tabla 1. Caso de prueba CDP-001

| Caso de prueba | *CDP-002* | | |
| --- | --- | --- | --- |
| Identificador de característica | *TBC-2: Iniciar sesión en el sistema* | | |
| Función probar | *F-6: userLogin* | | |
| Objetivo | *Validar que la petición de inicio de sesión de un usuario funciona correctamente y que el sistema manda el mensaje correcto en caso de éxito o fallo.* | | |
| Criterios de éxito | *Se ingresan los datos correctamente, ya se encuentra registrado en el sistema y se devuelve el mensaje de inicio de sesión exitoso.* | | |
| Criterios de falla | *Se ingresan los datos correctamente, y se encuentra registrado, y el sistema devuelve un mensaje de error.* | | |
| Precondiciones | *Debe estar el usuario registrado en el sistema.* | | |
| Necesidades para el caso de prueba | *Es necesario realizar esta prueba para asegurarse que la petición de los usuarios a comparar sean las correctas, si no son entonces se presentarán casos de administradores que tendrán respuestas diferentes.* | | |
| Autor | *Jesús Andrei Torres Landero* | | |
| Fecha de creación | *25/10/2022* | | |
| Entrada | **Salidas** | | |
| {  email: “[jesus@example.com](mailto:jose@example.com)”,  password: “passwordtest1”  } | {  message: “Login successfully”,  userName: “Jesus”,  token  } | | |
| Flujo del caso de prueba | **No paso** | **Usuario del sistema** | **Sistema** |
| *1* | *Se le envía a la función un objeto de tipo JSON con el correo electrónico y la contraseña de un usuario registrado.* | *Devuelve un objeto de tipo JSON con un mensaje exitoso, el nombre del usuario y un token.* |
| Post condiciones | No aplica. | | |

## Tabla 2. Caso de prueba CDP-002

| Caso de prueba | *CDP-003* | | |
| --- | --- | --- | --- |
| Identificador de característica | *TBC-3: Añadir recursos a la biblioteca* | | |
| Función probar | *F-1: createNewBook* | | |
| Objetivo | *Validar que la función devuelve el mensaje correcto cuando se le envía un libro con los datos completos y correctos.* | | |
| Criterios de éxito | *Se le pasan todos los datos del libro correctamente y devuelve el mensaje de éxito.* | | |
| Criterios de falla | *Se le pasan todos los datos del libro correctamente y devuelve un mensaje de error que no coincide con el mensaje esperado.* | | |
| Precondiciones | *Se debe haber ingresado sus datos en el formato de registro correspondiente.* | | |
| Necesidades para el caso de prueba | *Es necesario realizar esta prueba para asegurarse que la petición para añadir un nuevo libro funciona correctamente.* | | |
| Autor | *David Abraham Paredes Coob* | | |
| Fecha de creación | *26/10/2022* | | |
| Entradas | **Salidas** | | |
| {  id: 123,  image: “<https://imagen.com/500/500>,  title: “Harry Potter”,  author: “J. K. Rowling”,  price: 100,  description: “Libro de texto”,  isbn: “113-112”  } | {  message: “Book created successfully”,  data: {  id: 123,  image: “<https://imagen.com/500/500>,  title: “Harry Potter”,  author: “J. K. Rowling”,  price: 100,  description: “Libro de texto”,  isbn: “113-112”  }  } | | |
| Flujo del caso de prueba | **No paso** | **Usuario del sistema** | **Sistema** |
| *1* | *Se le envía un objeto de tipo JSON con los datos completos y correctos del libro que se quiere añadir.* | *Devuelve un objeto tipo JSON con un mensaje de éxito y los datos del libro que se añadió.* |
| Post condiciones | No aplica. | | |

## Tabla 3. Caso de prueba CDP-003

| Caso de prueba | *CDP-004* | | |
| --- | --- | --- | --- |
| Identificador de característica | *TBC-4: Eliminar recursos de la biblioteca* | | |
| Función probar | *F-3: deleteBookById* | | |
| Objetivo | *Validar que la función devuelve el mensaje correcto cuando se le envía un libro con los datos completos y correctos.* | | |
| Criterios de éxito | *Se le pasan todos los datos del libro correctamente y devuelve el mensaje de éxito.* | | |
| Criterios de falla | *Se le pasan todos los datos del libro correctamente y devuelve un mensaje de error que no coincide con el mensaje esperado.* | | |
| Precondiciones | *Se debe haber ingresado sus datos en el formato de registro correspondiente.* | | |
| Necesidades para el caso de prueba | *Es necesario realizar esta prueba para asegurarse que la petición para eliminar un libro funciona correctamente.* | | |
| Autor | *Julian Alejandro Perez Koo* | | |
| Fecha de creación | *26/10/2022* | | |
| Entradas | **Salidas** | | |
| {  id: 123  } | {  message: “Book deleted successfully”,  data: {  id: 123,  image: “<https://imagen.com/500/500>,  title: “Harry Potter”,  author: “J. K. Rowling”,  price: 100,  description: “Libro de texto”,  isbn: “113-112”  }  } | | |
| Flujo del caso de prueba | **No paso** | **Usuario del sistema** | **Sistema** |
| *1* | *Se le envía un objeto de tipo JSON con los datos completos y correctos del libro que se quiere eliminar.* | *Devuelve un objeto tipo JSON con un mensaje de éxito y los datos del libro que se eliminó.* |
| Post condiciones | No aplica. | | |

## Tabla 4. Caso de prueba CDP-004

| Caso de prueba | *CDP-005* | | |
| --- | --- | --- | --- |
| Identificador de característica | *TBC-5: Editar los recursos de la biblioteca* | | |
| Función probar | *F-4: updateABook* | | |
| Objetivo | *Validar que la función devuelve un mensaje de éxito y los datos actualizados del libro que se editó.* | | |
| Criterios de éxito | *Los datos nuevos del libro que se quiere actualizar son correctos y completos, y el sistema devuelve un mensaje de éxito y todos los libros incluyendo el libro actualizado.* | | |
| Criterios de falla | *Los datos nuevos del libro que se quiere actualizar son correctos y completos, y el sistema devuelve un mensaje de error que no coincide con el mensaje esperado.* | | |
| Precondiciones | *El libro debe estar registrado en el sistema y los datos que se envían deben ser correctos y completos.* | | |
| Necesidades para el caso de prueba | *Es necesario realizar esta prueba para validar que cuando se envían los datos correctos de un libro que se quiere actualizar, se devuelve el mensaje correcto.* | | |
| Autor | *José Ángel Ruiz Escalante* | | |
| Fecha de creación | *27/10/2022* | | |
| Entradas | **Salidas** | | |
| {  id: 123,  image: “<https://imagen.com/500/500>,  title: “Harry Potter Actualizado”,  author: “J. K. Rowling”,  price: 200,  description: “Libro de texto”,  isbn: “113-112” | {  message: “Book updated successfully”,  data: books  } | | |
| Flujo del caso de prueba | **No paso** | **Usuario del sistema** | **Sistema** |
| *1* | *Se le envía a la función un objeto de tipo JSON con la información actualizada del libro.* | *Devuelve un objeto de tipo JSON con un mensaje de éxito y con una lista de los libros incluyendo el libro actualizado.* |
| Post condiciones | No aplica. | | |

## Tabla 5. Caso de prueba CDP-005

| Caso de prueba | *CDP-006* | | |
| --- | --- | --- | --- |
| Identificador de característica | *TBC-6: Obtener todos los recursos de la biblioteca* | | |
| Función probar | *F-2: getAllBooks* | | |
| Objetivo | *Pedir todos los recursos de la biblioteca y obtenerlos exitosamente con el mensaje debido.* | | |
| Criterios de éxito | *La función devuelve el mensaje correcto y que todos los datos de los libros en la biblioteca obtenidos se encuentran completos y correctos cuando se hace la petición.* | | |
| Criterios de falla | *La función devuelve el mensaje incorrecto y/o algún dato de los libros en la biblioteca obtenidos no se encuentre completo y/o correctos cuando se hace la petición.* | | |
| Precondiciones | *Los libros deben estar registrados en el sistema y los datos que se envían deben ser correctos y completos.* | | |
| Necesidades para el caso de prueba | *Es necesario realizar esta prueba para validar que cuando se piden todos los libros los datos sean los correctos y el mensaje también.* | | |
| Autor | *Jesús Andrei Torres Landero* | | |
| Fecha de creación | *27/10/2022* | | |
| Entradas | **Salidas** | | |
| Ninguna | {  message: “All books fetched successfully”,  data: books  } | | |
| Flujo del caso de prueba | **No paso** | **Usuario del sistema** | **Sistema** |
| *1* | *Se hace la petición con todos los libros en la biblioteca.* | *Devuelve un objeto de tipo JSON con un mensaje de éxito y con una lista de todos los libros.* |
| Post condiciones | No aplica. | | |

## Tabla 6. Caso de prueba CDP-006