**Especificación de Diseño 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 Diseño de Pruebas. |
| --- | --- |
| **Referencia** | <https://github.com/JoseAngelRE/VerificacionYValidacion/tree/main/Plan_de_Pruebas> |
| **Autor** | José Ángel Ruiz Escalante |
| **Fecha** | 7/11/22 |

**Histórico de versiones**

| **Versión** | **Fecha** | **Estado** | **Responsable** | **Nombre del archivo** |
| --- | --- | --- | --- | --- |
| 1.0 | 1/11/22 | A | José Ángel Ruiz Escalante | Especificación de Diseño de Pruebas.docx |
| 1.2 | 7/11/22 | A | José Ángel Ruiz Escalante | Especificación de Diseño 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 Diseño de Pruebas. |
| 1.1 | 6/11/22 | Se realizaron los diseños de prueba para cada caso de prueba. |
| 1.2 | 7/11/22 | Se revisó el formato del documento y se corrigieron las observaciones. |

**Índice**

[**Identificador del documento**](#_77oe8lwyv01b) **4**

[**Documentos relacionados**](#_hv3nuknkq9b3) **4**

[**Tabla 1. Diseño de prueba DDP-001**](#_cismratr2ull) **4**

[**Tabla 2. Diseño de prueba DDP-002**](#_udufynw6of3k) **5**

[**Tabla 3. Diseño de prueba DDP-003**](#_ng9jy4svpmgk) **6**

[**Tabla 4. Diseño de prueba DDP-004**](#_cd2wspjt1hrf) **7**

[**Tabla 5. Diseño de prueba DDP-005**](#_xd7zpqgy8uer) **8**

[**Tabla 6. Diseño de prueba DDP-006**](#_2ccfdtboh5dm) **9**

# 

# **Identificador del documento**

VYV-EDP-2022-JAJD

# **Documentos relacionados**

* Plan de Pruebas VYV-PDP-2022-JAJD.
* Especificación de Procedimiento de Pruebas VYV-DEPP-2022-JAJD.
* Especificación de Casos de Pruebas VYV-ECP-2022-JAJD.

# Tabla 1. Diseño de prueba DDP-001

| **Identificador de diseño de prueba** | DDP-001 |
| --- | --- |
| **Componentes a probar** | F-1 createNewBook |
| **Enfoque** | |
| La función “createNewBook” recibe un objeto que contiene los atributos y los datos de un libro que se quiere agregar a la base de datos. Los atributos que debe tener el objeto que se le pasa como argumento son:   * id * image * title * author * price * description * isbn   Si el libro no se encuentra en la base de datos, se devuelve un mensaje de éxito, en caso de que el libro ya se encuentre en la base de datos o que suceda un error durante el proceso, se devuelve un mensaje de error.  El mensaje de éxito se devuelve en un objeto con dos atributos:   * message: “Book created successfully” * data: Este atributo representa la información del libro que se agregó.   El mensaje de error devuelve un objeto con dos atributos:   * message: “Something went wrong while creating a book” * data: error | |
| **Identificador del caso de prueba** | CDP-003 |
| **Identificador del procedimiento** | PR-001 |
| **Criterios de éxito** | Se le pasan todos los datos del libro correctamente y devuelve el mensaje de éxito. |
| **Criterios de fallo** | Se le pasan todos los datos del libro correctamente y devuelve un mensaje de error que no coincide con el mensaje esperado. |

# Tabla 2. Diseño de prueba DDP-002

| **Identificador de diseño de prueba** | DDP-002 |
| --- | --- |
| **Componentes a probar** | F-2 getAllBooks |
| **Enfoque** | |
| La función “getAllBooks” no recibe ningún dato de entrada y se encarga de pedir todos los recursos de la biblioteca y obtenerlos exitosamente con el mensaje debido.  Si se recuperan correctamente todos los libros, se devuelve un mensaje de éxito, en caso contrario se devuelve un mensaje de error.  El mensaje de éxito se devuelve en un objeto con dos atributos:   * message: “All books fetched successfully” * data: Este atributo representa un arreglo con objetos que representan la información de todos los libros almacenados en la base de datos.   El mensaje de error devuelve un mensaje con dos atributos:   * message: “Something went wrong while fetching users” * data: error | |
| **Identificador del caso de prueba** | CDP-006 |
| **Identificador del procedimiento** | PR-001 |
| **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 fallo** | 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. |

# Tabla 3. Diseño de prueba DDP-003

| **Identificador de diseño de prueba** | DDP-003 |
| --- | --- |
| **Componentes a probar** | F-3 deleteBookById |
| **Enfoque** | |
| La función “deleteBookById” recibe el id de un libro almacenado en la base de datos y lo elimina.  La función devuelve un mensaje de éxito en caso de que el libro se haya encontrado y eliminado exitosamente de la base de datos, en caso contrario, es decir, el libro no se encuentra o ocurre un error durante el proceso se devuelve un mensaje de error.  El mensaje de éxito se devuelve en un objeto con dos atributos:   * message: “Book deleted successfully” * data: Este atributo representa el objeto con la información del libro que se ha eliminado.   El mensaje de error se devuelve en un objeto con dos atributos:   * message: “Something went wrong while deleting a book” * data: error | |
| **Identificador del caso de prueba** | CDP-004 |
| **Identificador del procedimiento** | PR-001 |
| **Criterios de éxito** | Se le pasan todos los datos del libro correctamente y devuelve el mensaje de éxito. |
| **Criterios de fallo** | Se le pasan todos los datos del libro correctamente y devuelve un mensaje de error que no coincide con el mensaje esperado. |

# Tabla 4. Diseño de prueba DDP-004

| **Identificador de diseño de prueba** | DDP-004 |
| --- | --- |
| **Componentes a probar** | F-4 updateABook |
| **Enfoque** | |
| La función “updateABook” recibe un objeto que contiene los atributos y los datos de un libro que se quiere actualizar en la base de datos. Los atributos que debe tener el objeto que se le pasa como argumento son:   * id * image * title * author * price * description * isbn   Si el libro se encuentra en la base de datos devuelve un mensaje de éxito, en caso contrario que no se encuentre en la base de datos o que suceda un error durante el proceso, se devuelve un mensaje de error.  El mensaje de éxito se devuelve en un objeto con dos atributos:   * message: “Book updated successfully” * data: Este atributo representa un arreglo con objetos que representan la información de todos los libros almacenados en la base de datos.   El mensaje de error devuelve un objeto con dos atributos:   * message: “Something went wrong while updating a book” * data: error | |
| **Identificador del caso de prueba** | CDP-005 |
| **Identificador del procedimiento** | PR-001 |
| **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 fallo** | 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. |

# Tabla 5. Diseño de prueba DDP-005

| **Identificador de diseño de prueba** | DDP-005 |
| --- | --- |
| **Componentes a probar** | F-5 createNewUser |
| **Enfoque** | |
| La función “createNewUser” recibe un objeto que contiene los atributos y los datos de un usuario que se quiere registrar en el sistema. Los atributos del usuario son los siguientes:   * userName * email * password   Si el usuario no se encuentra en la base de datos y se registra exitosamente en la base de datos, se devuelve un mensaje de éxito. Por otro lado, si el usuario ya está registrado o ha ocurrido un error en el proceso, o los datos son incorrectos se devuelve un mensaje de error.  El mensaje de éxito se devuelve en un objeto con dos atributos:   * message: Sign up successfully” * data: Este atributo representa un objeto con los datos del usuario que se acaba de registrar en el sistema.   El mensaje de error se devuelve en un objeto con dos atributos:   * message: “Something went wrong while creating an user” * data: error | |
| **Identificador del caso de prueba** | CDP-001 |
| **Identificador del procedimiento** | PR-001 |
| **Criterios de éxito** | Se envían todos los datos correctamente y el sistema envía un mensaje de éxito. |
| **Criterios de fallo** | Se envían todos los datos correctamente y el sistema envía un mensaje de error. |

# Tabla 6. Diseño de prueba DDP-006

| **Identificador de diseño de prueba** | DDP-006 |
| --- | --- |
| **Componentes a probar** | F-6 userLogin |
| **Enfoque** | |
| La función “userLogin” recibe un objeto que contiene los atributos y los datos de un usuario que se quiere validar con la base de datos. Los atributos que debe tener el objeto que se le pasa como argumento son:   * email * password   Si el usuario se encuentra en la base de datos, se devuelve un mensaje de éxito, en caso de que el usuario no se encuentre en la base de datos o que suceda un error durante el proceso, se devuelve un mensaje de error.  El mensaje de éxito se devuelve en un objeto con dos atributos:   * message: “Login successfully” * userName: Este atributo representa el nombre del usuario que se encuentra en la base de datos. * token: Este atributo representa un valor que el sistema tiene que enviar al hacer solicitudes para autentificarse durante una sesión.   Si el email que recibe la función es una cadena vacía, se devuelve un mensaje de error con dos atributos:   * message: ”Error invalid email, please enter a valid email” * data: Este atributo representa un objeto vacío.   Si el email y contraseña no coinciden con ningún usuario registrado en la base de datos, se devuelve un mensaje de error con los atributos:   * message: “Error user not found” * data: Este atributo representa un objeto con los datos ingresados por el usuario.   Si se encuentra un usuario con el correo electrónico, pero la contraseñas no coinciden se devuelve un mensaje de error con los siguientes atributos:   * message: "Error invalid password" * email: Este atributo representa la información proporcionada por el usuario.   Si los datos proporcionados por el usuario son correctos y se encuentra registrado en el sistema, pero ocurre un error durante el proceso, se devuelve un mensaje de error con los siguientes atributos:   * message: "Something went wrong while fetching a user credentials" * data: error | |
| **Identificador del caso de prueba** | CDP-002 |
| **Identificador del procedimiento** | PR-001 |
| **Criterios de éxito** | Se ingresan los datos correctamente, el usuario ya se encuentra registrado en el sistema y se devuelve el mensaje de inicio de sesión exitoso. |
| **Criterios de fallo** | Se ingresan los datos correctamente, y se encuentra registrado, y el sistema devuelve un mensaje de error. |