

# Proyecto Integrador

Anlly López Betancur

---

La aplicación desarrollada en el Proyecto Integrador es una herramienta sencilla que permite a los usuarios registrar y buscar productos, manejando funcionalidades de búsqueda por nombre o categoría y el registro de nuevos ítems en una base de datos. El objetivo principal de este proyecto fue aplicar los conocimientos adquiridos en cursos previos de desarrollo web y programación.

Durante el desarrollo de esta aplicación, se puso en práctica el uso de tecnologías front-end y back-end. En el lado del cliente, se utilizó React para construir una interfaz de usuario interactiva, apoyada por Bootstrap para aprovechar sus características de diseño responsive y estilizado. Para el back-end, se utilizó Spring Boot para la creación de una API que interactúa con una base de datos MySQL, administrada a través de MySQL Workbench.

Este trabajo no solo implicó la aplicación de habilidades técnicas en el diseño y codificación de la aplicación, sino que también proporcionó una plataforma para enfrentar y superar retos prácticos. El siguiente documento detalla todas las fases del proyecto, desde la concepción inicial hasta las pruebas finales, destacando los desafíos encontrados y las soluciones implementadas.

## Objetivos del Proyecto

El proyecto asignado tenía como eje central la creación de una aplicación web que manejara dos funcionalidades esenciales: el registro de productos y la búsqueda de estos mismos. Los objetivos específicos trazados para cumplir con esta consigna fueron los siguientes:

- Implementar de manera efectiva las funcionalidades de búsqueda y registro de productos según las especificaciones proporcionadas, asegurando que los usuarios puedan agregar productos a la base de datos y realizar búsquedas eficientes.
- Desarrollar una aplicación que no solo sea funcional sino también ordenada y agradable a la vista, emulando la estructura y el diseño de otras aplicaciones web; y además asegurando que la interfaz de usuario sea clara y fácil de usar.
- Aplicar los conocimientos técnicos adquiridos en cursos de desarrollo front-end y back-end, utilizando herramientas y frameworks como React, Bootstrap, Spring Boot y MySQL.

## Proceso de desarrollo

El desarrollo de la aplicación comenzó con una etapa de conceptualización, donde se definieron varios aspectos fundamentales que guiarían todo el proyecto. A continuación, se detallan los pasos seguidos en esta fase creativa y constructiva:

El primer paso fue elegir el tema central de la aplicación. Se decidió que el dominio de la aplicación serían los instrumentos musicales, un área que ofrece una amplia variedad de productos y que es de interés para una comunidad específica de usuarios. Con la temática en mente, se procedió a esbozar cómo se deseaba que luciera la aplicación. Esto involucró imaginar la disposición de los componentes en la pantalla, la navegación entre páginas y la interacción general del usuario con la aplicación. Los bosquejos iniciales sirvieron como una guía visual para la estructuración de la interfaz de usuario.

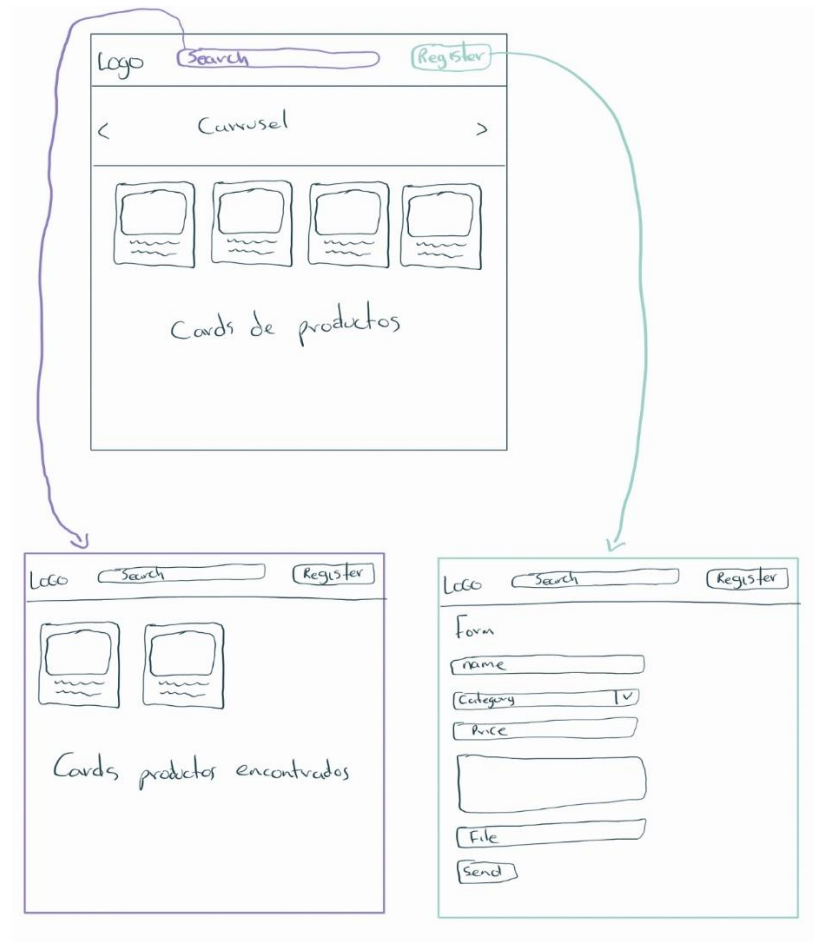


Figura 1. Bosquejo de la aplicación.

Posteriormente, se dedicó tiempo a la creación de la identidad de la aplicación, lo que incluyó la elección de un nombre y un eslogan, el siguiente paso fue diseñar el logo, seleccionar una paleta de colores, y definir el estilo de la fuente.



Figura 2. Logotipo y eslogan de la aplicación.

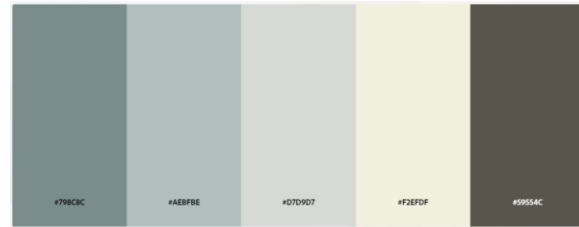


Figura 3. Paleta de colores

Con una visión clara del diseño visual, se comenzó la creación de la interfaz de usuario empleando React. Durante esta etapa, se aplicaron los principios de diseño de interfaces aprendidos y se escribió código CSS para personalizar los componentes. Además, se utilizaron algunos elementos de Bootstrap.

Para facilitar el desarrollo del front-end sin depender de la finalización del back-end, se utilizó una API de simulación que permitió emular los datos que posteriormente serían proporcionados por la API propia de la aplicación. Esta estrategia permitió un desarrollo paralelo e independiente de las capas de presentación y lógica.

La construcción del back-end comenzó con la creación de la base de datos en MySQL utilizando MySQL Workbench. Este paso implicó el diseño de las tablas, estableciendo todos los campos necesarios y las relaciones entre ellas. Con la base de datos listada y operativa, la atención se desplazó hacia la creación de la aplicación utilizando Spring Boot Initializr, que proporcionó una plantilla de proyecto con todas las dependencias necesarias para un inicio rápido. La configuración inicial del proyecto incluyó: entidades JPA, se definieron clases de entidad que representaban las tablas product y category en la base de datos, utilizando la funcionalidad de JPA para mapear estos objetos a registros en la base de datos; conexión a la Base de Datos, donde se escribió y probó el código necesario para establecer la conexión con la base de datos, utilizando JPA para facilitar la interacción. Una vez asegurada la conexión con la base de datos, se procedió a implementar los componentes restantes de la arquitectura de la aplicación: DTOs para transferir datos entre subsistemas de la aplicación, DAOs para abstraer la lógica de acceso a la base de datos, controladores para operaciones clave como recuperar todos los productos, buscar productos por ID o por nombre/categoría, añadir nuevos productos a la base de datos y eliminar productos por ID.

## Metodología de Pruebas

### - Pruebas Automatizadas con Postman

Para validar la funcionalidad de la API, se utilizó Postman para crear una serie de pruebas automatizadas. Estas pruebas permitieron verificar el correcto funcionamiento de los endpoints, así como asegurar que los códigos de estado y las respuestas de la API fueran adecuados y consistentes con la lógica de negocio esperada.

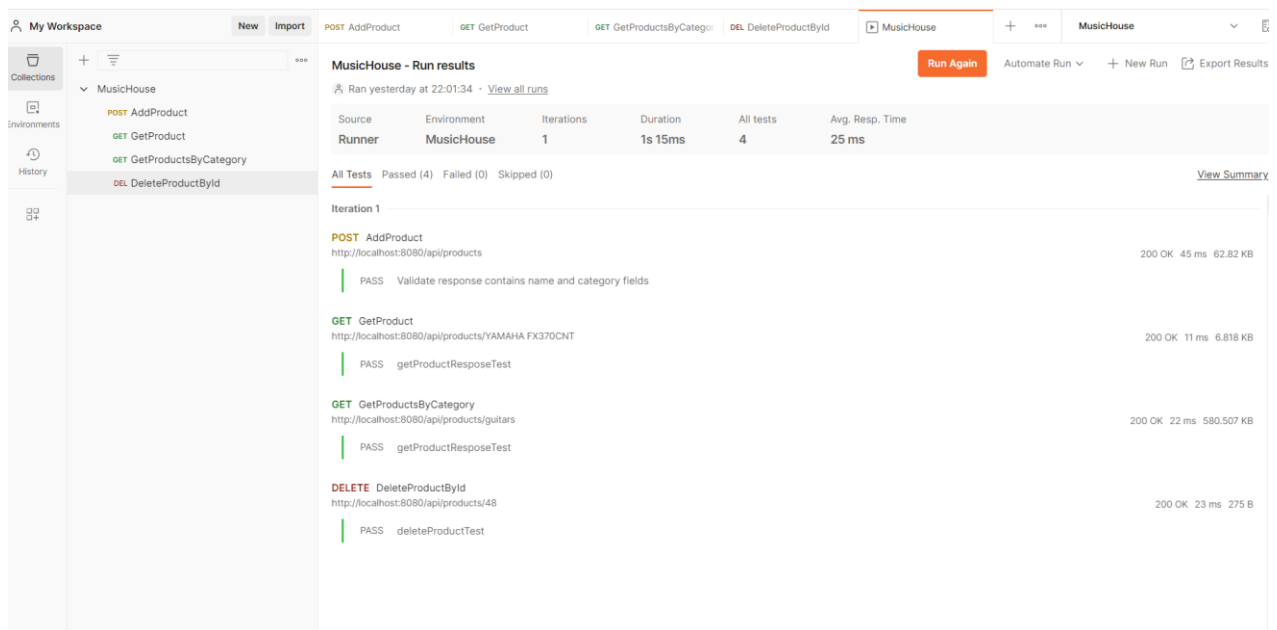


Figura 4. Test en Postman

## - Pruebas Manuales

Además de las pruebas automatizadas, se diseñaron y ejecutaron casos de prueba manuales para las funcionalidades clave de la aplicación: la búsqueda por nombre o categoría y el registro de un nuevo producto.

Los casos de prueba manual se dividieron en dos categorías principales:

- Casos de Prueba Positivos: Estos casos se enfocaron en escenarios donde se esperaba que la aplicación funcionara según lo previsto, garantizando que las funcionalidades estuvieran implementadas correctamente y que la aplicación manejara los casos de uso estándar sin errores.
- Casos de Prueba Negativos: En estos casos, se probaron escenarios de error para asegurar que la aplicación manejara adecuadamente las entradas inválidas, mostrando mensajes de error apropiados y evitando comportamientos inesperados o fallas del sistema.

## Casos de prueba

### Búsqueda por nombre

Id caso de prueba	CP-001
Título	Búsqueda por nombre
Descripción	Verificar que la búsqueda por nombre devuelve las cards correspondientes.
Precondiciones	La aplicación está abierta y hay datos precargados en la base de datos.
Pasos	1. Ingresar el nombre exacto de un ítem en el cuadro de búsqueda. 2. Presionar la tecla Enter.
Datos de prueba	Nombre de un ítem existente en la base de datos.

Resultado esperado	Las cards que coinciden exactamente con el nombre ingresado se muestran en los resultados de búsqueda.
Resultado real	Las cards que coinciden con el nombre ingresado en el cuadro de búsqueda son mostradas en pantalla

#### Búsqueda por cadena vacía

<b>Id caso de prueba</b>	<b>CP-002</b>
Título	Búsqueda con cadena vacía
Descripción	Verificar que el sistema maneja adecuadamente una cadena de búsqueda vacía.
Precondiciones	La aplicación está abierta y operativa.
Pasos	1. Dejar el cuadro de búsqueda vacío. 2. Presionar la tecla Enter.
Datos de prueba	Ninguno (cadena vacía).
Resultado esperado	No se realiza ninguna acción
Resultado real	No se realiza ninguna acción en la aplicación

#### Registro de Producto con Todos los Campos Llenos

<b>Id caso de Prueba</b>	<b>CP-003</b>
Título	Registro exitoso de producto con todos los campos llenos
Descripción	Verificar que un producto se puede registrar correctamente cuando todos los campos requeridos están llenos y la descripción es opcional.
Precondiciones	La aplicación está abierta.
Pasos	1. Hacer clic en el botón "Register Product". 2. Llenar el campo "Name" con un nombre válido. 3. Seleccionar una categoría en el "Select Category". 4. Introducir un precio válido en "Price". 5. (Opcional) Agregar una descripción en "Description". 6. Hacer clic en el botón para cargar un archivo y seleccionar un archivo válido. 7. Hacer clic en el botón "Save".
Datos de prueba	Nombre: "Ukulele" Categoría: Guitars Precio: "230000" Archivo: ukulele.jpg
Resultado esperado	El formulario se envía sin errores, el producto se registra en la base de datos, y todos los campos del formulario se restablecen a su estado vacío inicial.
Resultado real	El formulario se envía sin ningún problema, todos los campos se reestablecen a su estado vacío inicial.

#### Registro de Producto sin seleccionar categoría

<b>Id caso de Prueba</b>	<b>CP-003</b>
Título	Registro de producto sin seleccionar categoría
Descripción	Verificar que el sistema impide el registro de un producto si la categoría no ha sido seleccionada.

Precondiciones	La aplicación está abierta.
Pasos	<ol style="list-style-type: none"> <li>1. Hacer clic en el botón "Register Product".</li> <li>2. Llenar el campo "Name" con un nombre válido.</li> <li>3. Dejar la selección de "Select Category" en su estado por defecto (sin seleccionar).</li> <li>4. Introducir un precio válido en "Price".</li> <li>5. Cargar un archivo válido.</li> <li>6. Hacer clic en el botón "Save".</li> </ol>
Datos de prueba	Nombre: "ukulele" Categoría: [No seleccionada] Precio: "10000" Archivo: ukulele.jpg
Resultado esperado	El sistema no permite el envío del formulario y muestra un mensaje de error indicando que la categoría es obligatoria.
Resultado real	El formulario no puede ser enviado, en el campo category aparece el siguiente mensaje de error: "Please selecta an item in the list"

## Desafíos y Obstáculos

Durante el desarrollo del proyecto, me enfrenté a una serie de desafíos que dificultaron su realización. Uno de los primeros obstáculos fue el refrescar y aplicar conocimientos previos, también aprender conceptos que no se habían visto en los cursos, esto implicó un repaso intensivo de materiales de cursos anteriores y la práctica de conceptos y herramientas para la programación y el diseño de aplicaciones.

Otra limitación significativa fue la falta de acceso a infraestructura de servicios en la nube como AWS. Esto me llevó a tomar la decisión de almacenar imágenes directamente en la base de datos, en lugar de utilizar soluciones especializadas como Amazon S3. A pesar de que esta solución no es ideal desde el punto de vista de la escalabilidad y el rendimiento, permitió seguir avanzando en el proyecto dentro de las restricciones existentes.

El tiempo fue otro factor crítico que afectó el desarrollo del proyecto. Con un plazo muy corto, se priorizaron las funcionalidades esenciales por encima de mejoras en el desempeño y el diseño. Esto significó aceptar que algunas de las características deseables del proyecto tendrían que ser pospuestas para futuras versiones de la aplicación.

A pesar de estos desafíos, cada obstáculo superado ha sido una oportunidad de aprendizaje, enseñándome no solo sobre el desarrollo de software sino también sobre la resiliencia y la gestión eficaz del tiempo y recursos.

## Apéndices

El código fuente, la Postman collection y el dump de la base de datos puede encontrarse en el siguiente repositorio: <https://github.com/anlly1796/ProyectoIntegrador>