



UNIVERSIDAD DE LAS FUERZAS ARMADAS

DESARROLLO WEB AVANZADO

DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN

Actividad 5: Rails

Estudiantes:

Marlyn Almeida, Erick Andrade, Josué Merino

Docente:

Ing. Diego Saavedra

1. Objetivo

- Aprender los conceptos básicos de Ruby on Rails, incluyendo la configuración del entorno de desarrollo.
- Practicar la creación de modelos, controladores y vistas en Rails.
- Entender el funcionamiento y la implementación de operaciones CRUD (Crear, Leer, Actualizar, Eliminar).
- Mejorar la apariencia y la usabilidad de la aplicación mediante estilos básicos con CSS.

2. Descripción de la Actividad

En esta actividad, los estudiantes construirán una aplicación web básica de inventario utilizando Ruby on Rails. La aplicación permitirá a los usuarios realizar operaciones CRUD sobre elementos de inventario, incluyendo la creación, lectura, actualización y eliminación de ítems. También aprenderán a mejorar la apariencia de la aplicación mediante estilos CSS básicos.

3. Tareas a Realizar

3.1. Configuración del Entorno:

- Instalación de Ruby, Rails y Git.
- Creación de un nuevo proyecto de Rails utilizando el generador de Rails (`rails new`).

3.2. Desarrollo del modelo:

- Generación del modelo Item con los atributos necesarios (`name`, `description`, `quantity`).
- Migración de la base de datos para crear la tabla correspondiente (`rails db:migrate`).

3.3. Desarrollo del Controlador:

- Creación del controlador `ItemsController` con las acciones CRUD (`index`, `show`, `new`, `create`, `edit`, `update`, `destroy`).

3.4. Desarrollo de las Vistas:

- Creación de vistas para mostrar la lista de ítems (index), detalles de un ítem (show), formulario de creación (new y create), formulario de edición (edit y update).

3.5. Estilización con CSS:

- Mejora de la apariencia de la aplicación utilizando reglas CSS básicas en el archivo application.css.

3.6. Pruebas y Verificación:

- Verificación de que todas las operaciones CRUD funcionen correctamente.
- Ejecución de pruebas manuales en el navegador para crear, ver, actualizar y eliminar ítems.

4. Capturas

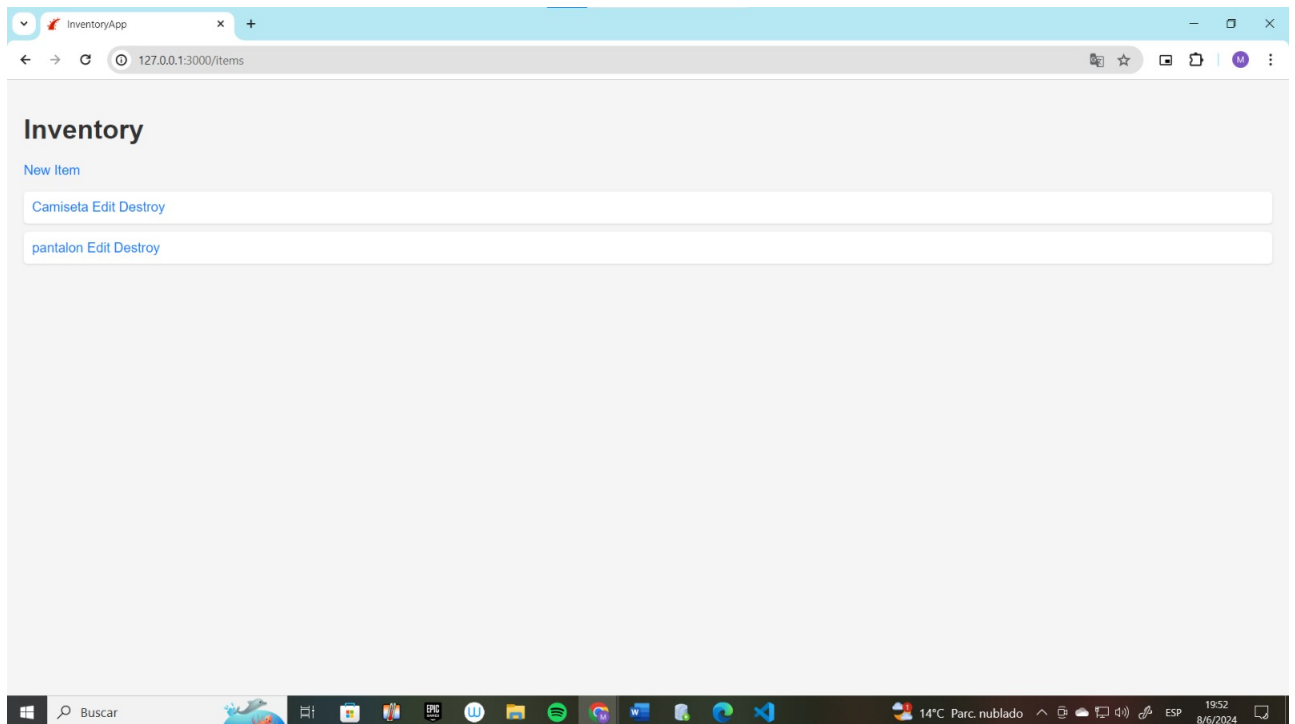


Figura 1: Inicio

Actividad 5: Rails

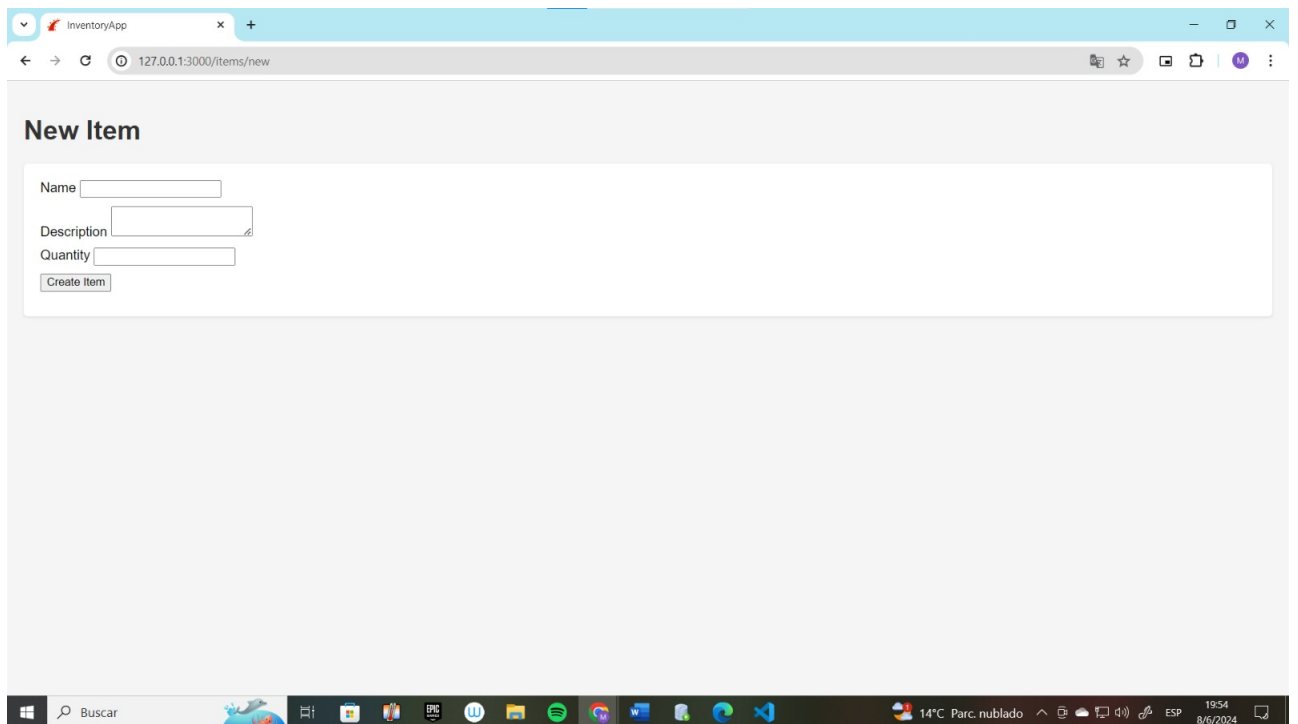


Figura 2: Crear

Actividad 5: Rails

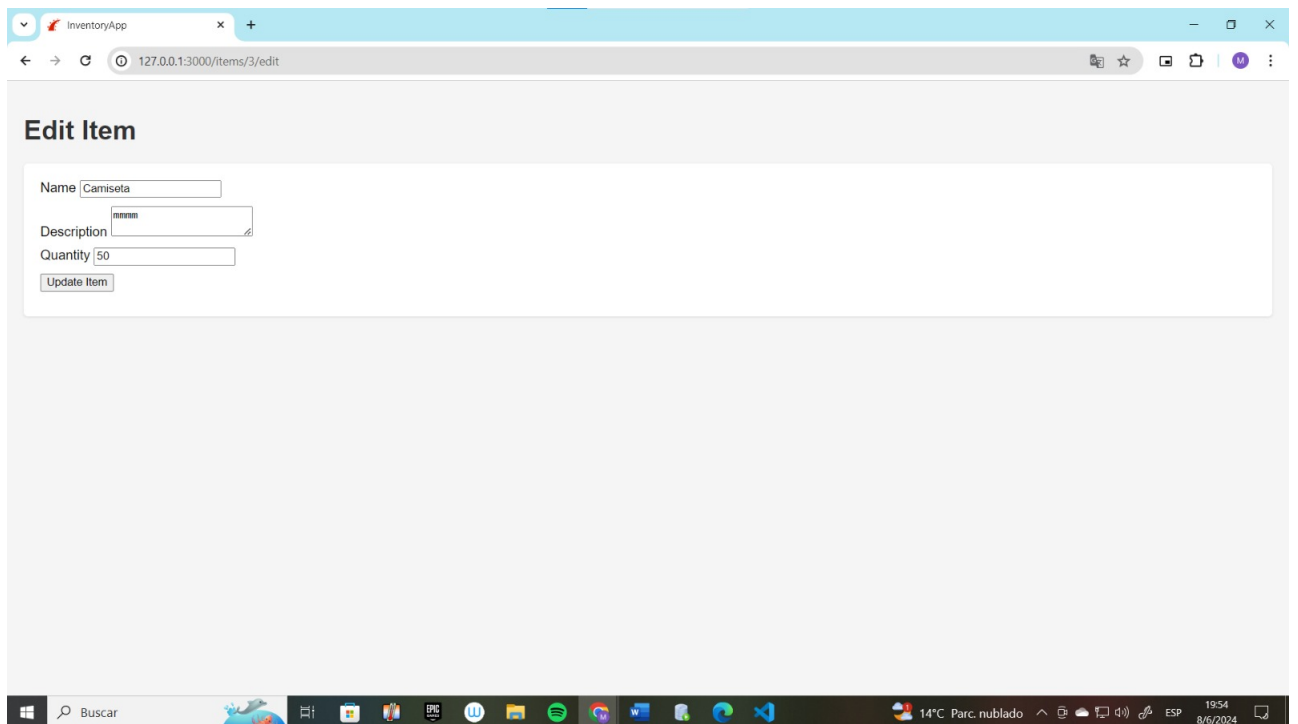


Figura 3: Editar

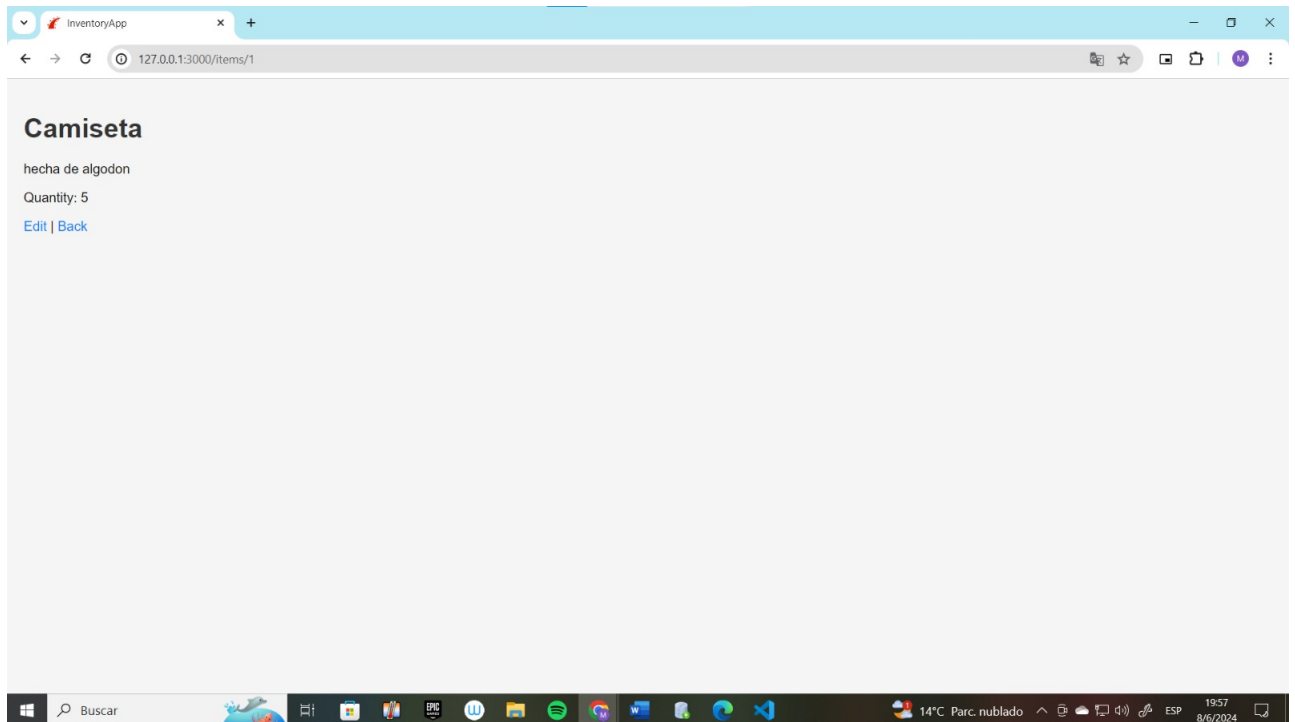


Figura 4: Ver

5. Conclusiones

5.1. Familiarización con Ruby on Rails:

Desarrollar una aplicación web de inventario permite a los estudiantes adquirir habilidades fundamentales en Ruby on Rails. Esto incluye la instalación y configuración del entorno de desarrollo, la creación de proyectos, y la utilización de generadores para crear modelos y controladores. A través de esta práctica, los estudiantes comprenden la estructura MVC (Modelo-Vista-Controlador) y cómo Rails facilita la organización del código.

5.2. Dominio de Operaciones CRUD:

Implementar operaciones CRUD (Crear, Leer, Actualizar, Eliminar) es esencial para la gestión de datos en cualquier aplicación web. En este proyecto, los estudiantes aprenden a generar modelos y controladores que permiten manipular los datos del inventario de manera eficiente. Esto les proporciona una base sólida para cualquier desarrollo web, ya que las operaciones CRUD son fundamentales en la mayoría de las aplicaciones.

5.3. Desarrollo de Interfaces de Usuario:

Crear vistas para las diferentes acciones (index, show, new, edit) permite a los estudiantes entender cómo construir interfaces de usuario dinámicas y funcionales. Aprenden a utilizar formularios para la entrada de datos y a mostrar información de manera organizada y accesible. Además, esta parte del proyecto destaca la importancia de la interacción entre el backend y el frontend.

..

5.4. Estilización y Diseño Básico con CSS:

Mejorar la apariencia de la aplicación utilizando CSS básica enseña a los estudiantes cómo estilizar una aplicación web para hacerla más atractiva y fácil de usar. Aunque el enfoque es básico, comprender cómo aplicar estilos CSS y trabajar con el archivo `application.css` es crucial para el desarrollo web, ya que una buena apariencia puede mejorar significativamente la experiencia del usuario.

Pruebas y Verificación de Funcionalidades: La verificación de que todas las operaciones CRUD funcionan correctamente y la realización de pruebas manuales son pasos importantes en el desarrollo de software. Estas actividades enseñan a los estudiantes la importancia de la calidad y la fiabilidad en el desarrollo de aplicaciones. Aprenden a identificar y corregir errores, asegurando que la aplicación cumple con los requisitos y funciona como se espera.

6. Referencias

1. Instagram, P. O. (2016). Instagram. Online unter: <https://www.wissenschaftskommunikation.de/format/instagram>.
2. Escalante, F., & Castillo, O. (2021). Arquitectura de microservicios para una plataforma de red social: Un caso de estudio de Instagram. *Tecnociencia*, 25(2), 233-249.
3. Autor: Hernández-Ortiz, J. C., & Díaz-Oviedo, R. (2022). *Arquitecturas de software escalables: Patrones, tecnologías y casos prácticos*. Madrid, España: Editorial ESIC.
4. López-López, M. A. (2020). *Análisis de la escalabilidad y la eficiencia de las arquitecturas de microservicios en aplicaciones web de gran escala: Un estudio de caso en Instagram*. [Tesis doctoral, Universidad Politécnica de Madrid].

5. Zhang, T., & Cheng, S. (2019). Arquitectura de Instagram: Un análisis en profundidad
6. Instagram Engineering. (2023). Cómo funciona Instagram: Una mirada detrás de escena. <https://instagram-engineering.com/>
7. Imagen recopilada de: Microservicios para Instagram. (2022, enero 29). SACAViX Tech - Te Ayudamos a Ser Un Mejor Desarrollador Backend. <https://sacavix.com/2022/01/microservicios-para-instagram/>