



**Definición del Proyecto**

**Actividad de Aprendizaje N.1**

**Tema:** Sistema de control de inventarios y stock utilizando una arquitectura cliente servidor.

**Introducción:** La tienda Fruit Shop ‘Happy Avocado’, es un negocio que inició sus actividades en el 2019 con el objetivo de vender frutas de calidad en el sector de la Mitad del Mundo.

Sin embargo, los dueños del negocio iniciaron sus actividades con un sistema poco efectivo para controlar sus productos o inventarios, el cual consiste en un cuaderno de inventarios en donde se registraba todas las entradas y salidas de productos de los inventarios con esfero y hojas, lo cual a futuro le causó problemas en el ámbito económico del establecimiento dado que no se tenía un registro óptimo de la cantidad, de los precios y de las descripciones de los productos.

Es por ello, que ante estos problemas se planteó la necesidad de crear una aplicación web que permita controlar de una forma eficaz los inventarios del negocio, llevando un registro de los productos, sus precios, cantidad, tipos y descripción. De esta manera el negocio podrá tener un sistema más estable que le genere ganancias y rentabilidad a largo plazo.



Figura 1. Logo de la empresa.

**Objetivo:**

Implementar un sistema de inventarios para la empresa “Fruit Shop: Happy Avocado” que nos permita simplificar los procesos de monitoreo de productos y stock de manera automática, mediante la arquitectura cliente servidor.

**Problema:**

Se necesita implementar un sistema o aplicación web que permita registrar la información de las frutas del negocio mediante una base de datos y a su vez se muestre una interfaz (Página web o index) para que los diferentes usuarios puedan interactuar con la aplicación mediante las diferentes funcionalidades (CRUD y reglas del negocio) del mismo de acuerdo a las tareas que requieran hacer.

**Herramientas y Entornos de Desarrollo:**

Las herramientas y entornos de desarrollo se escogieron y aplicaron de acuerdo al sistema y lenguaje de programación utilizado, que en este caso fue JAVA. Se trabajó mediante un IDE (Entorno de Desarrollo Integrado) para generar el código del aplicativo, de igual manera, se aplicaron conceptos de servidores web en el cual se decidió utilizar el servidor Apache Tomcat para un desarrollo más óptimo. Finalmente, se hizo uso de un sistema de base de datos para el manejo de los datos y la información, que para este proyecto fue MongoDB.

A continuación se describe de una manera más detallada cada herramienta usada en el proyecto, tomando énfasis en las versiones y características aplicadas al sistema que en nuestro caso, el entorno de desarrollo se ejecutó bajo el sistema operativo de Windows 10.

**IDE: IntelliJ IDEA Version: 2022.1.1**



Figura 2. IDE IntelliJ.

Se trata de un IDE inteligente que reconoce el contexto para trabajar con Java y otros lenguajes JVM como Kotlin, Scala y Groovy en todo tipo de aplicaciones., además de que permite desarrollar aplicaciones web completas, gracias a sus potentes herramientas integradas, soporte para JavaScript y tecnologías relacionadas, y soporte avanzado para marcos populares como Spring, Spring Boot, Jakarta EE, Micronaut, Quarkus, Helidon.

En este caso se tomó en cuenta este IDE para el desarrollo de nuestro aplicativo web.

Para la descarga e instalación del IDE se puede acceder mediante el siguiente enlace: <https://www.jetbrains.com/idea/download/#section=windows>

**Módulo de Proyecto: Spring Boot**

****

Spring Boot es un módulo del proyecto Spring creado para simplificar el desarrollo de aplicaciones mediante Spring Framework bajo la licencia Apache 2.0.

**SERVIDOR WEB: Apache Tomcat**

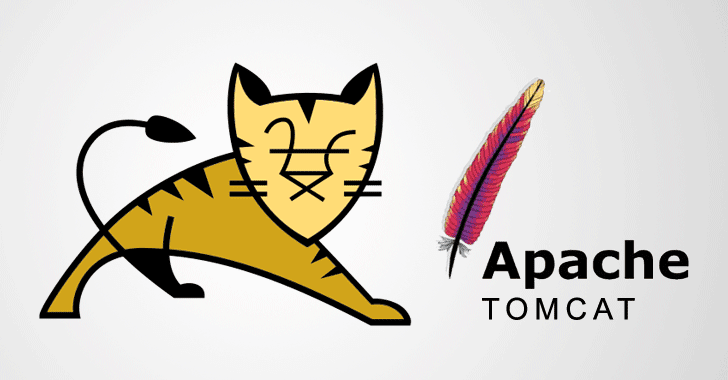
****

Figura 3. Servidor Apache Tomcat.

Apache Tomcat, también conocido como Tomcat Server, demuestra ser una opción popular para los desarrolladores web que crean y mantienen aplicaciones y sitios web dinámicos basados ​​en la plataforma de software Java.

En palabras simples, podemos decir que [Apache Tomcat](https://www.javatpoint.com/java-and-tomcat-setup-for-jenkins) es en realidad un servidor y un contenedor de servlets. En fin, en nuestro caso se hizo uso de este servidor, por el hecho de que proporciona todas las funcionalidades necesarias y estándares requeridos para poder operar un servidor, lo que significa que brinda una carga y redespliegue muy rápidos en comparación con sus diversas alternativas.

Para la descarga e instalación de de Payara se puede acceder mediante el siguiente enlace: <https://www.payara.fish/downloads/>

**Base de Datos: MongoDB**

****

Figura 4. Sistema de BD mongoDB.

Sistema de base de datos NoSQL (No es un modelo de datos relacional), orientado a documentos de código abierto y que se maneja bajo el lenguaje de C++. Este sistema se maneja bajo estructuras de datos BSON (similar a JSON) con un esquema dinámico, a diferencia de otros sistemas de BD que guardan los datos en tablas.

Este sistema trabaja bajo dos modalidades que son:

Aplicativo MOngoDB: Que trabaja mediante un host (físico o virtual).

Nube MongoDB Atlas: Este sistema de MongoDB permite almacenar base de datos en la nube mediante clusters.

Para nuestro sistema aplicaremos el concepto de MongoDB Atlas, el cual ofrece un sistema más rápido y eficiente para la conectividad y reserva de la información y elimina la necesidad de instalar y administrar entornos de BBDD. Para la ejecución de MongoDB Atlas se puede acceder mediante el siguiente enlace:<https://www.mongodb.com/atlas/database>