**** 

­

**Web Application Development**

Team 4

MEMBERS:

* Pardo Rómulo
* Pila Nataly
* Pilligua Washington
* Romero Cristhian

Theme:

Client - Server Web Development

**Activity 1**

Tutor:

PhD. JORGE EDISON LASCANO Mgs

**Definición del Proyecto**

**Actividad de Aprendizaje N.1**

**Tema:** Sistema para ventas y stock utilizando una arquitectura cliente servidor.

**Introducción:** La tienda tecnológica TECH STORE, es un negocio que inició sus actividades en el 2015 con el objetivo de vender productos tecnológicos, partes y piezas para computadoras en la ciudad de Quito.

Actualmente, el negocio no cuenta con un sistema efectivo para llevar a cabo los anuncios sus productos, el cual consiste en anuncios en redes sociales únicamente y lo registros del stock de productos se lo lleva manualmente, tampoco se tiene un registro óptimo de la cantidad, de los precios y de las descripciones de los productos. Lo cual no es muy efectivo en el manejo del negocio por lo expuesto anteriormente.

Conociendo la problemática del negocio, se plantea la necesidad de crear una aplicación web que permita administrar de una forma efectiva las ventas y los inventarios del negocio, llevando un registro de los productos, sus precios, cantidad, tipos y descripción. Con el fin de tener un sistema más estable que genere rentabilidad.



Figura 1. Logo de la tienda.

**Objetivo:**

Implementar un sistema de ventas para la PYME “Tech Store” que permita simplificar los procesos de venta de productos y control de stock de manera automática, mediante la arquitectura cliente servidor.

**Problema:**

Se requiere una aplicación web que permita gestionar las ventas de los productos del negocio mediante una base de datos y a su vez se muestre una interfaz (Página web) 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: Apache NetBeans IDE 14**

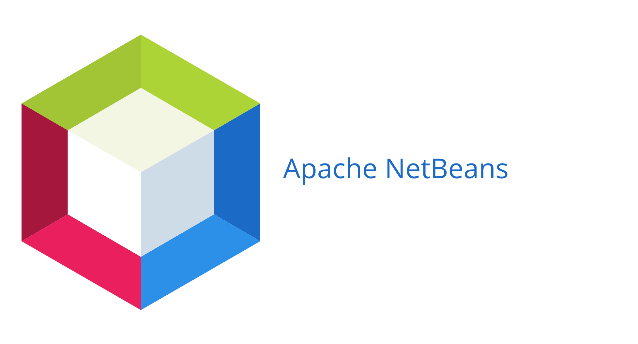


Figura 2. IDE NetBeans.

Es un entorno de desarrollo integrado gratuito y de código abierto para desarrollar aplicaciones en los sistemas operativos Windows, Mac, Linux y Solaris. El IDE simplifica el desarrollo de aplicaciones web, empresariales, de escritorio y móviles utilizando plataformas Java y HTML5.

**SERVIDOR WEB: PAYARA SERVER**



Figura 3. Servidor Apache Tomcat.

Payara Server es la plataforma de código abierto de elección para el desarrollo de aplicaciones Java EE de producción. Actualmente es el sustituto lógico para el servidor GlassFish edición de código abierto con la tranquilidad que contiene mejoras, correcciones de errores y parches.

Payara Server cuenta con un soporte de producción de 24/7, es una solución creíble sobre la que construir su plataforma de middleware Java, es una alternativa atractiva, derivado de GlassFish.

**Base de Datos: Atlas MongoDB**

****

Figura 4. Logo Sistema 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, que ofrece un sistema rápido y eficiente para la conectividad y elimina la necesidad de instalar y administrar entornos de BBDD. Para la ejecución de MongoDB Atlas se puede acceder mediante el siguiente enlace y crear una cuenta gratuita limitada: <https://www.mongodb.com/cloud/atlas/register>