

**CIMAT**

# **Artefactos de Software**

## **Profesor Titular**

José Arturo Mora  
Soto

---

## **Tarea 2**

**Sistemas Distribuidos**

**Cliente - Servidor**

## **Alumno**

Iván E. Tinajero Díaz

19-Septiembre-2014

# INDICE GENERAL

## TABLA DE CONTENIDO

1. INTRODUCCIÓN.....	1
2. DESCRIPCION DEL PARADIGMA CLIENTE-SERVIDOR .....	1
3. EJEMPLOS DE SISTEMAS CLIENTE-SERVIDOR .....	2
3.1 EJEMPLO 1.SERVIDOR HTTP JAVA .....	2
3.2 EJEMPLO 2: SERVIDOR DE BASE DE DATOS MYSQL.....	3
3. JAVA APLICADO A SISTEMAS CLIENTE-SERVIDOR .....	4
4. DESCRIPCION DE LA APLICACIÓN DESARROLLADA.....	6
4.1 SCREENSHOT – APLICACIÓN CORRIENDO EN UN NAVEGADOR DE PCs.....	7
4.2 SCREENSHOT – APLICACIÓN CORRIENDO EN DISPOSITIVOS MOVILES. ....	10
4.3 INSTALACIÓN DE LA APLICACIÓN .....	11
5. BIBLIOGRAFIA.....	11

## INDICE DE FIGURAS

Figura 1: Arquitectura Cliente-Servidor .....	2
Figura 2: Arquitectura Tomcat HTTP Server .....	3
Figura 3: Arquitectura Mysql Server .....	4
Figura 4: Arquitectura de la aplicación Cliente-Servidor creada con JAVA.....	5
Figura 5: Diagrama Entidad-Relación de la aplicación web.....	6
Figura 6: Página inicial – Chrome .....	7
Figura 7: Pantalla Acerca - Chrome .....	7
Figura 8: Vendedor después de hacer Login - Chrome.....	8
Figura 9: Formulario para capturar venta - Chrome .....	8
Figura 10: Reporte de ventas de la Sucursal – Chrome.....	9
Figura 11: Reporte de ventas de todas las sucursales (Perfil Administrador) Chrome .....	9
Figura 12: Pantalla inicio en dispositivo Samsung Android .....	10
Figura 13: Pantalla nueva venta en dispositivo iPhone 4 .....	10
Figura 14: Pantalla inicio en dispositivo LG Android 4.....	11
Figura 15: Pantalla nueva venta en dispositivo LG Android 4 .....	11

## 1. INTRODUCCIÓN

Hoy en día el Internet está teniendo un gran impacto en nuestra vida diaria por ejemplo en la forma de comunicarnos, la forma de trabajar, la forma de comprar o vender artículos, incluso la forma de divertirnos. Es por eso que el uso de software y en general los sistemas de información se están convirtiendo en una necesidad de la sociedad.

La mayoría de los sistemas de información están así como los diferentes dispositivos que hacen uso del software están conectados a través de Internet y como cada día crece el número de dispositivos conectados a la red se necesitan sistemas más rápidos y eficientes. Es por esto que la mayoría de los sistemas actuales son conocidos como sistemas distribuidos.

Los **Sistemas Distribuidos** (Coulouris, Dollimore, Kindberg, & Blair, 2012) están formados por recursos de computación (hardware y software) físicamente distribuidos e interconectados a través de una red, que comunican mediante paso de mensajes y cooperan para realizar una determinada tarea. Existen diferentes paradigmas de Sistemas Distribuidos como por ejemplo cliente-servidor, paso de mensajes, servicios en red, llamada a métodos remotos, etc.

El presente documento tiene como objetivo principal explicar de forma general el paradigma Cliente-Servidor (C/S), además se mostrarán algunos ejemplos prácticos de donde es aplicado dicho paradigma y finalmente se mostrará una aplicación web desarrollada con Java y Mysql donde es aplicado dicho paradigma.

## 2. DESCRIPCION DEL PARADIGMA CLIENTE-SERVIDOR

La arquitectura C/S es un modelo de aplicación distribuida en el que las tareas se reparten entre los proveedores de recursos o servicios, llamados servidores, y los demandantes, llamados clientes. Un cliente realiza peticiones a otro programa, el servidor, quien le da respuesta. Esta idea también se puede aplicar a programas que se ejecutan sobre una sola computadora, aunque es más ventajosa en un sistema operativo multiusuario distribuido a través de una red de computadoras.

De forma general, básicamente existen dos roles principales en la arquitectura C/S:

- Servidor
  - Ofrece un servicio. Ejemplo una página WEB utilizando el protocolo HTTP.
  - A este rol se le conoce como Pasivo, ya que siempre está a la espera de peticiones de los clientes.
- Cliente
  - Solicita un servicio. Ejemplo un dispositivo móvil que accede a una página web.
  - A este rol se le conoce como Activo, ya que es el que hace las peticiones a los servidores

Un punto importante a tomar en cuenta es que este paradigma de C/S está pensado principalmente para servicios centralizados. Algunos ejemplos muy comunes de esta arquitectura son servicios de internet como HTTP, FTP, DNS, RDBMS, etc.

La siguiente figura muestra un esquema general de cómo es un sistema distribuido basado en la arquitectura C/S.

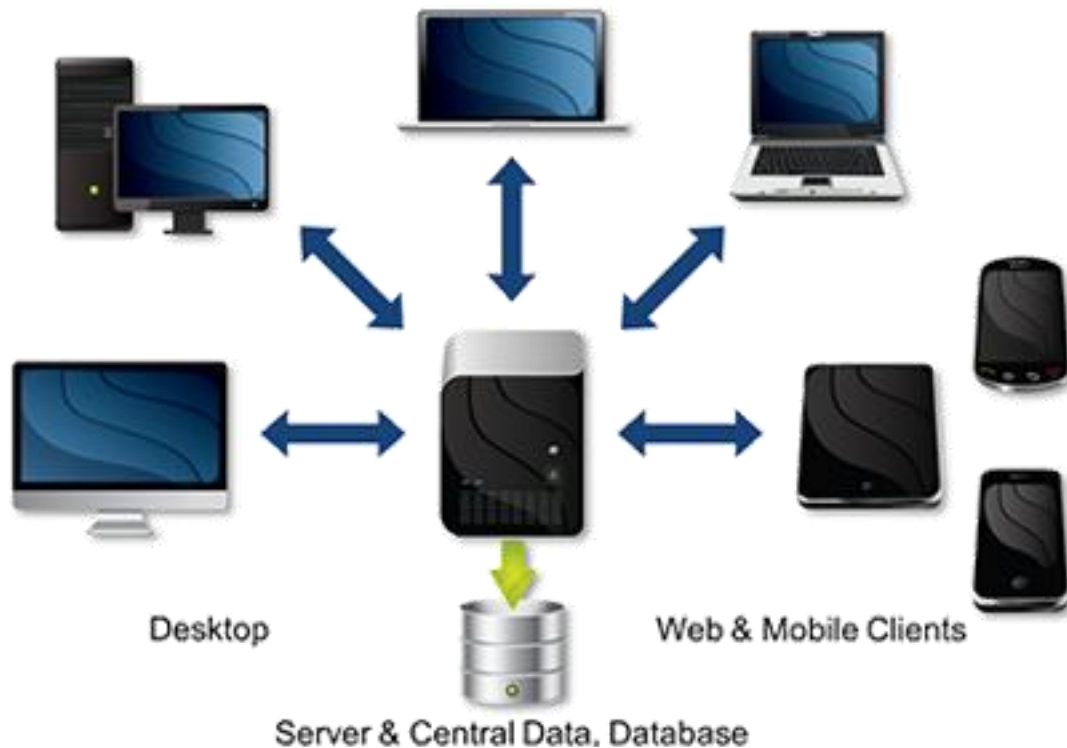


Figura 1: Arquitectura Cliente-Servidor

Nota: La aplicación web desarrollada para representar la arquitectura C/S es similar a la que es presentada en la figura anterior, es decir es una aplicación WEB en donde los clientes pueden ser diferentes dispositivos como PC, tabletas, celulares, etc.

### 3. EJEMPLOS DE SISTEMAS CLIENTE-SERVIDOR

A continuación se describen dos ejemplos muy usados de arquitecturas C/S. Cabe destacar estos ejemplos de arquitecturas fueron las usadas para desarrollar la aplicación de ejemplo.

#### 3.1 EJEMPLO 1.SERVIDOR HTTP JAVA

Apache Tomcat (Apache, 2014) (también llamado Jakarta Tomcat o simplemente Tomcat) funciona como un contenedor de servlets desarrollado bajo el proyecto Jakarta en la Apache Software Foundation. Tomcat implementa las especificaciones de los servlets y de JavaServer Pages (JSP) de Oracle Corporation. Para efecto de la explicación de la arquitectura C/S Apache Tomcat actuará como un Servidor HTTP el cual permite servir páginas web creadas utilizando el lenguaje de programación JAVA de Oracle.

En el caso de los clientes podría ser navegadores de Internet como Firefox, Internet Explorer, incluso navegadores que corren en dispositivos móviles.

Resumiendo, esta arquitectura sería:

- Servidor: Apache Tomcat (HTTP)
- Clientes: Navegadores de internet para PC y móviles.

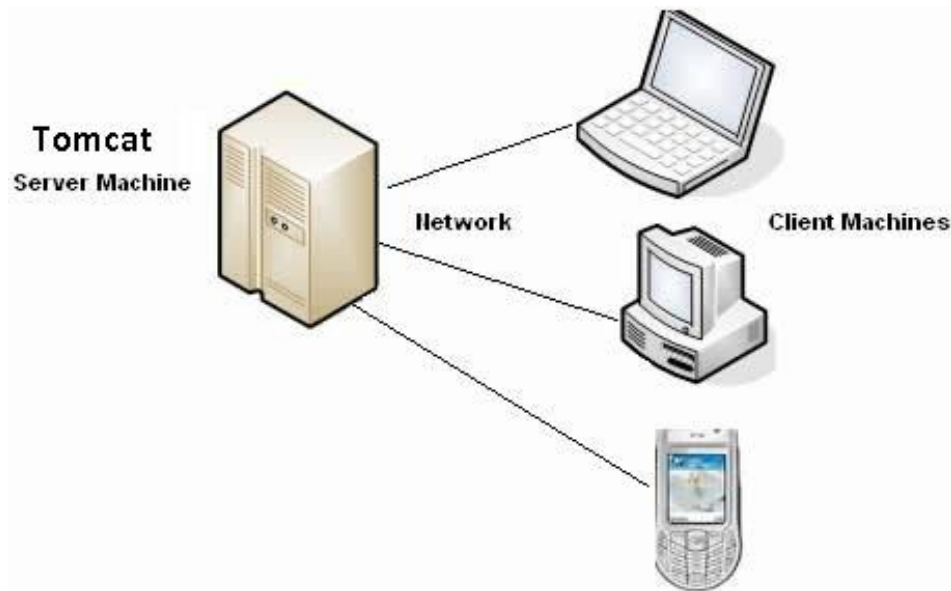


Figura 2: Arquitectura Tomcat HTTP Server

### 3.2 EJEMPLO 2: SERVIDOR DE BASE DE DATOS MYSQL.

MYSQL es un sistema de gestión de bases de datos relacional, multihilo y multiusuario que corre en plataformas como Windows, Linux, MAC, etc. Este es un buen ejemplo de la arquitectura C/S ya que MYSQL en realidad es un servidor de bases de datos el cual esta ejecutándose en espera de clientes los cuales pueden ser por ejemplo el cliente nativo de MYSQL ejecutándose desde la consola, una aplicación web corriendo en apache, herramientas graficas de administración (GUI), etc.

Resumiendo, esta arquitectura sería:

- Servidor: MYSQL Server:
- Clientes: phpMyAdmin, Mysql Workbench, una aplicación hecha en PHP, etc.

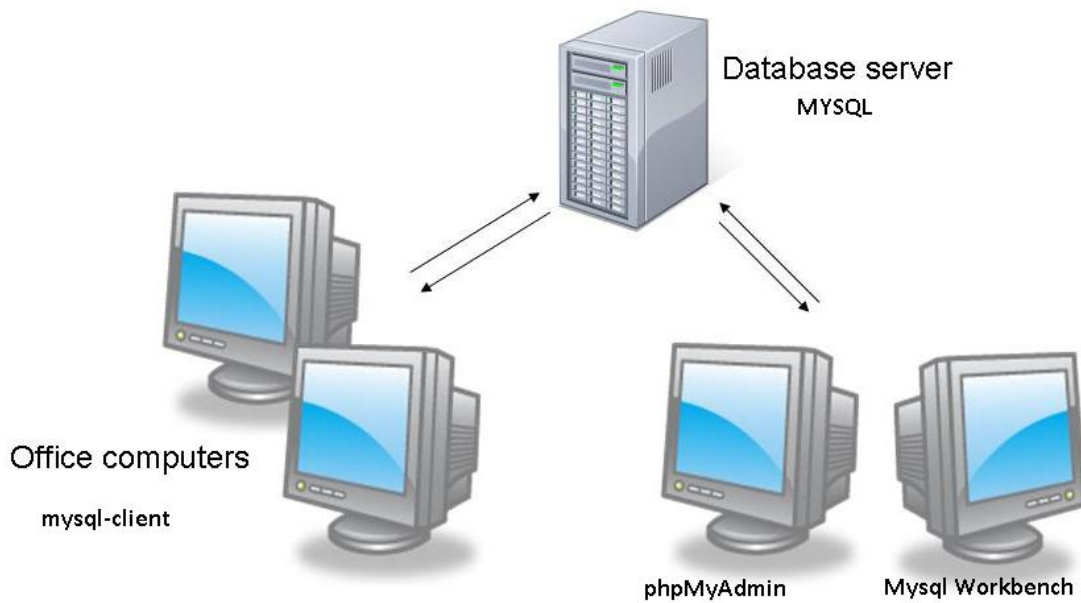


Figura 3: Arquitectura Mysql Server

### 3. JAVA APLICADO A SISTEMAS CLIENTE-SERVIDOR

JAVA es un lenguaje de programación muy robusto hoy en día cuyo propósito general es ser multiuso es decir puede ser usado para crear un sinfín de aplicaciones que van desde aplicaciones de escritorio, aplicaciones móviles, servidores de bases de datos como Neo4J, etc. JAVA cuenta con una extensa API la cual ya cuenta con librerías y paquetes los cuales están listos para ser usados. Además de esa API, existe un gran número de frameworks con los cuales puede ser aún más extendido el lenguaje.

Para crear sistemas distribuidos utilizando la arquitectura Cliente/Servidor existen muchas posibilidades utilizando JAVA. La decisión de cual utilizar dependerá del tipo de aplicación que se requiera desarrollar. Para efectos de demostración de un sistema distribuido utilizando JAVA como lenguaje de desarrollo se utilizó la siguiente arquitectura mostrada en la siguiente figura:

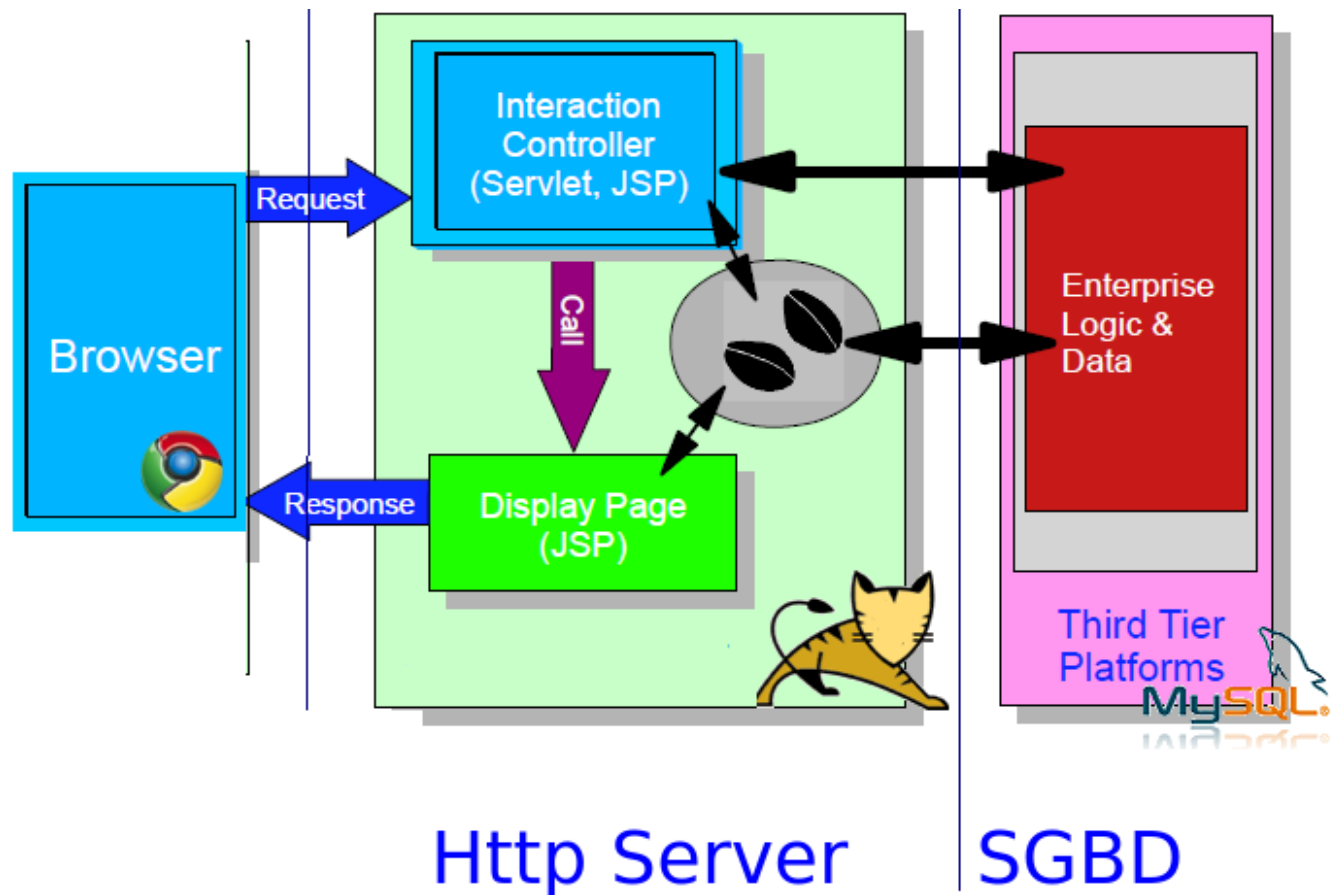


Figura 4: Arquitectura de la aplicación Cliente-Servidor creada con JAVA

A continuación se describe de forma general que principales elementos del lenguaje JAVA que fueron utilizados para hacer posible la aplicación creada.

- **JavaBeans:** Permiten representar el modelo de negocio de una aplicación y son los que permiten transferir información entre las diferentes capas de la arquitectura MVC. Por lo general estas clases representan la capa Model en la arquitectura MVC.
- **JSP (Java Server Pages):** Son archivos especiales los cuales tiene como finalidad mostrar la vista final del usuario, por lo general son las paginas HTML. Por lo general estos archivos representan la capa View en la arquitectura MVC.
- **Servlets:** Los servlets son clases especiales las cuales tiene como finalidad controlar el flujo de la aplicación. Por lo general estas clases representan la capa Controller en la arquitectura MVC.

## 4. DESCRIPCION DE LA APLICACIÓN DESARROLLADA

A efectos de demostración de una aplicación Cliente-Servidor utilizando JAVA como lenguaje de programación se desarrollo una aplicación web responsiva la cual funciona tanto en navegadores de Internet de PC, así como en los principales navegadores web móviles. La ventaja de ser responsiva es que el tamaño de los controles se adapta según el tamaño del dispositivo.

La aplicación web desarrollada es un punto de venta (POS) para registrar ventas en diferentes sucursales (zapaterías) y tiene reporte de ventas por sucursal y ventas globales que puede ver un administrador. El sistema tiene dos perfiles:

- **Perfil Vendedor:** Este usuario está asignado a una sucursal y básicamente tiene dos opciones en el sistema:
  - Capturar Venta
  - Ver reporte de ventas de esa sucursal.
- **Perfil Administrador:** Este perfil solo tiene una opción que es ver un reporte general de ventas de todas las sucursales.

Para el desarrollo de esta aplicación se utilizaron las siguientes tecnologías:

- A. Servidor Linux: Ubuntu Server 14.04 LTS 64Bits
- B. Apache Tomcat Versión 7
- C. JDK Versión 7
- D. jQuery Mobile Versión 1.4.4 (Mobile, 2014)
- E. Mysql Versión 5.5

La siguiente figura muestra el modelo Entidad-Relación de la base de datos MYSQL.

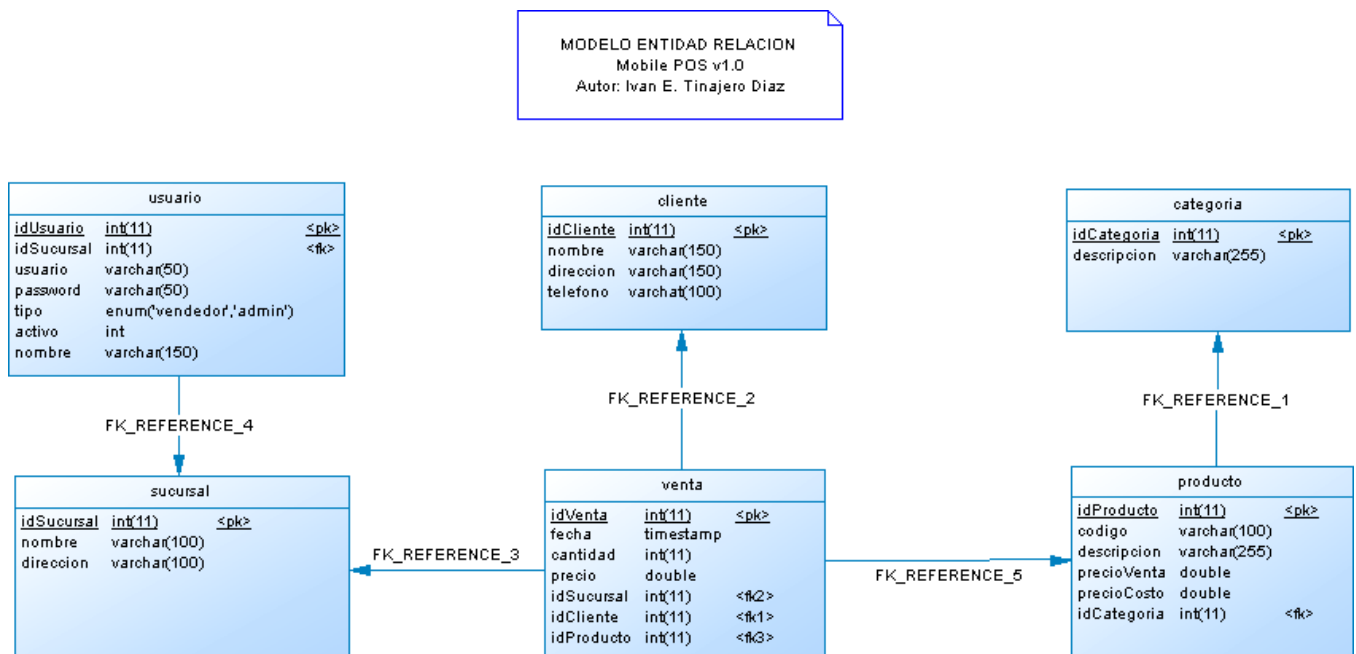


Figura 5: Diagrama Entidad-Relación de la aplicación web



#### 4.1 SCREENSHOT – APLICACIÓN CORRIENDO EN UN NAVEGADOR DE PCs.

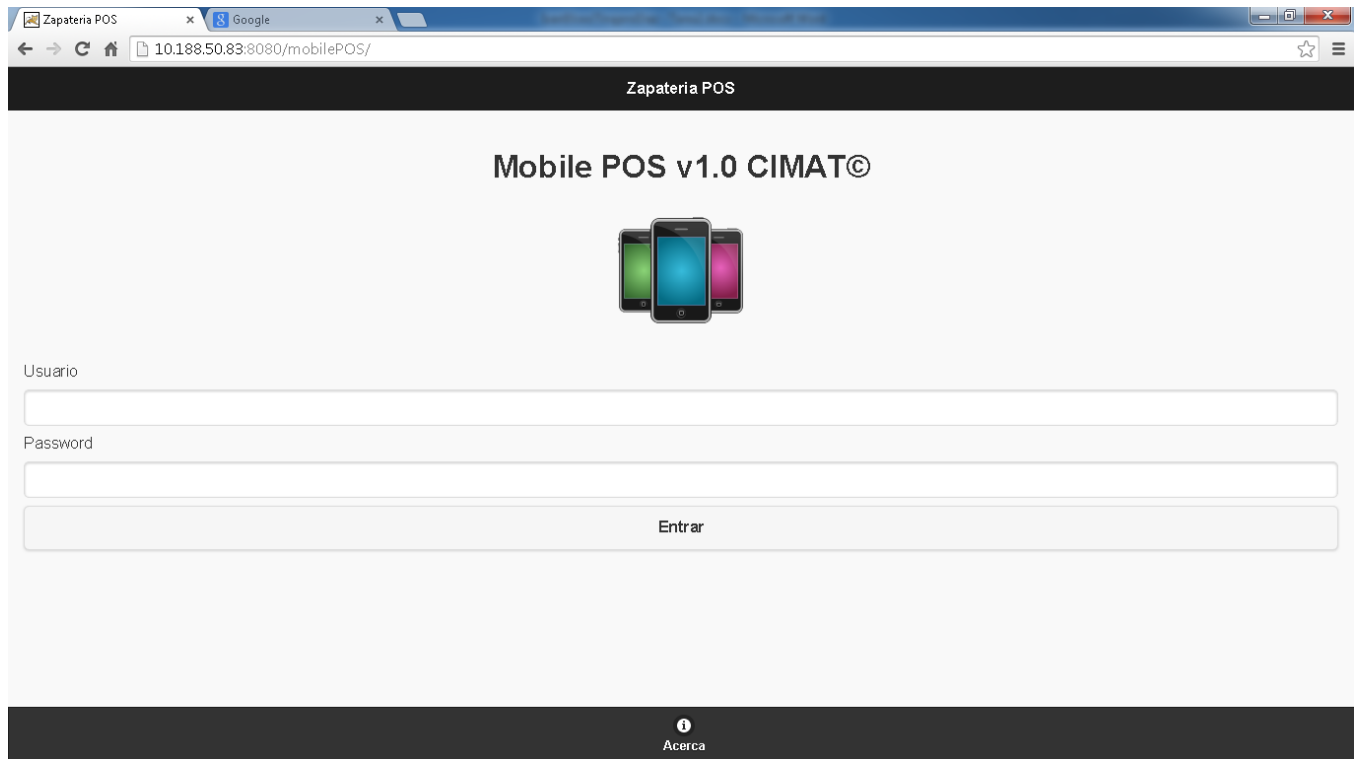


Figura 6: Página inicial – Chrome



Figura 7: Pantalla Acerca - Chrome

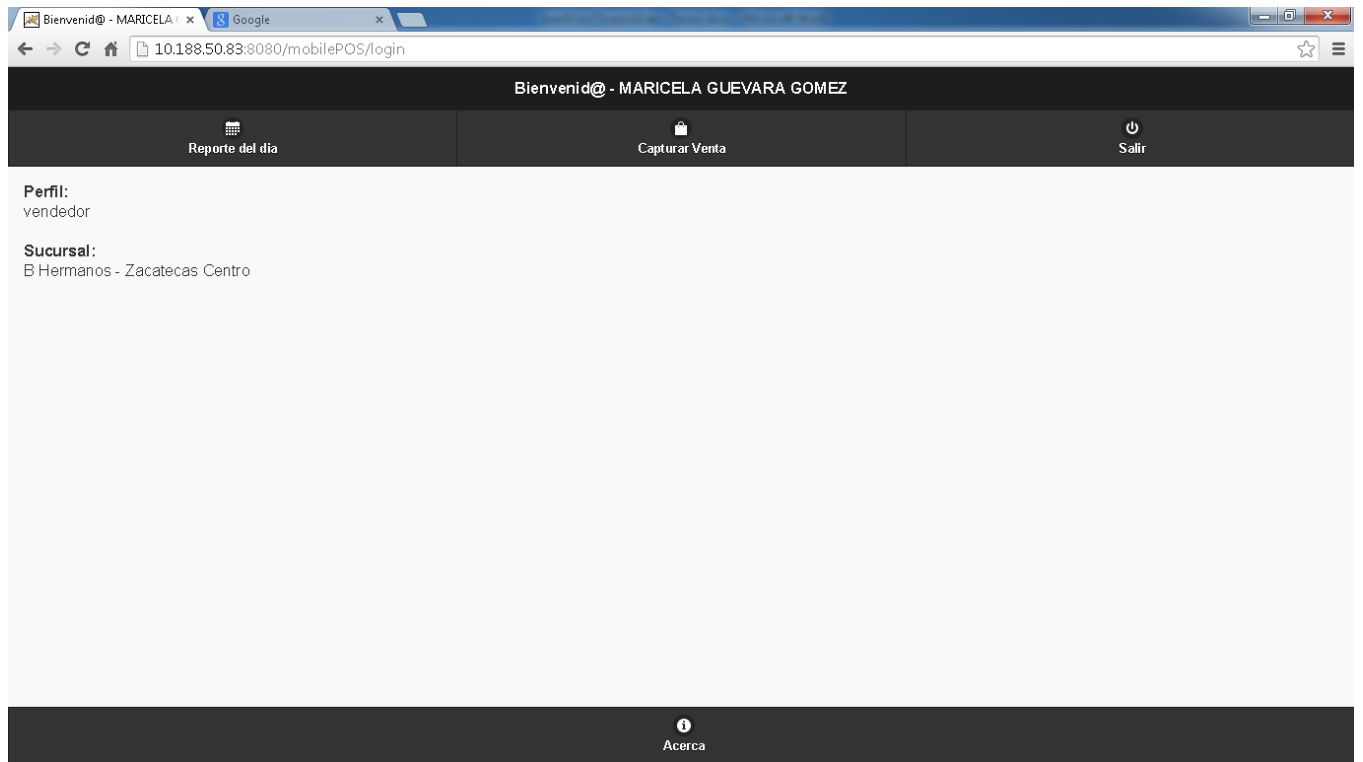


Figura 8: Vendedor después de hacer Login - Chrome

Figura 9: Formulario para capturar venta - Chrome

Reporte del día

Capturar Venta

Salir

Listado de ventas

Columns to display...

Sucursal	Fecha	Producto	Cantidad	Precio	Total
B Hermanos - Zacatecas Centro	2014-09-19 13:55:00.0	Tennis Nike Jordan 52 #6	1	2200.0	\$2200.0
B Hermanos - Zacatecas Centro	2014-09-19 10:20:00.0	Zapatos para Mujer Gucci #4	3	1500.0	\$4500.0
B Hermanos - Zacatecas Centro	2014-09-19 10:14:00.0	Tennis Nike Modelo 1524 #7	3	700.0	\$2100.0
B Hermanos - Zacatecas Centro	2014-09-18 18:47:00.0	Zapatos para Mujer Gucci #4	5	1500.0	\$7500.0
B Hermanos - Zacatecas Centro	2014-09-18 18:38:00.0	Tennis Running Mod 774 # 7	6	1400.0	\$8400.0
B Hermanos - Zacatecas Centro	2014-09-18 18:37:00.0	Zapatos para Mujer Gucci #4	1	1500.0	\$1500.0

Acerca

Figura 10: Reporte de ventas de la Sucursal – Chrome

Reporte del día

Salir

Listado de ventas

Columns to display...

Sucursal	Fecha	Producto	Cantidad	Precio	Total
B Hermanos - Zacatecas Centro	2014-09-19 13:55:00.0	Tennis Nike Jordan 52 #6	1	2200.0	\$2200.0
B Hermanos - Jerez	2014-09-19 11:26:00.0	Tennis Adidas modelo 4511 #5	2	1800.0	\$3600.0
B Hermanos - Jerez	2014-09-19 11:26:00.0	Tennis Nike Jordan 52 #6	1	2200.0	\$2200.0
B Hermanos - Fresnillo	2014-09-19 10:51:00.0	Zapatos para Mujer Gucci #4	3	1500.0	\$4500.0
B Hermanos - Zacatecas Centro	2014-09-19 10:20:00.0	Zapatos para Mujer Gucci #4	3	1500.0	\$4500.0
B Hermanos - Zacatecas Centro	2014-09-19 10:14:00.0	Tennis Nike Modelo 1524 #7	3	700.0	\$2100.0
B Hermanos - Fresnillo	2014-09-18 19:21:00.0	Zapatos para Mujer Gucci #4	3	1500.0	\$4500.0
B Hermanos - Zacatecas Centro	2014-09-18 18:47:00.0	Zapatos para Mujer Gucci #4	5	1500.0	\$7500.0
B Hermanos - Zacatecas Centro	2014-09-18 18:38:00.0	Tennis Running Mod 774 # 7	6	1400.0	\$8400.0
B Hermanos - Zacatecas Centro	2014-09-18 18:37:00.0	Zapatos para Mujer Gucci #4	1	1500.0	\$1500.0

Acerca

Figura 11: Reporte de ventas de todas las sucursales (Perfil Administrador) Chrome

## 4.2 SCREENSHOT – APLICACIÓN CORRIENDO EN DISPOSITIVOS MOVILES.

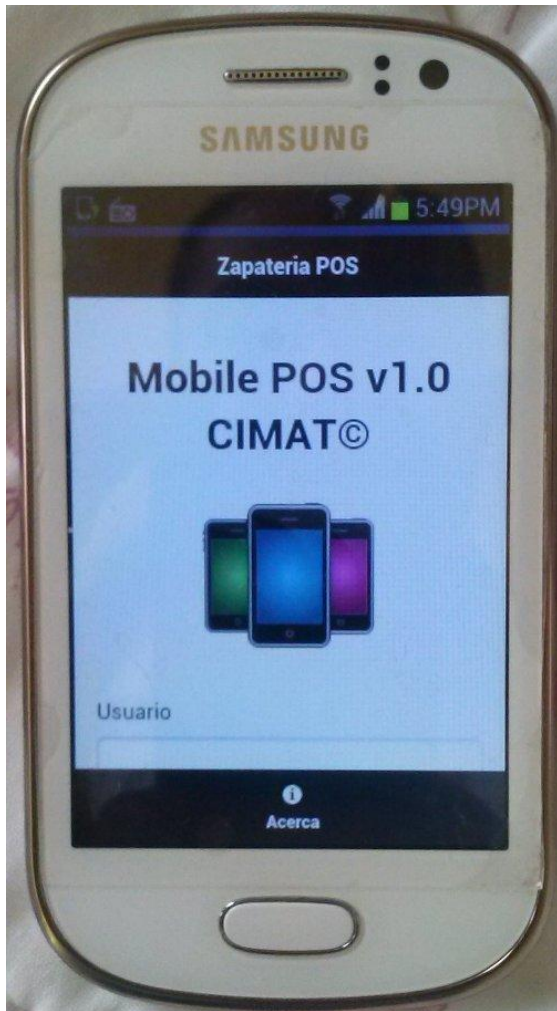


Figura 12: Pantalla inicio en dispositivo Samsung Android



Figura 13: Pantalla nueva venta en dispositivo iPhone 4



Figura 14: Pantalla inicio en dispositivo LG Android 4

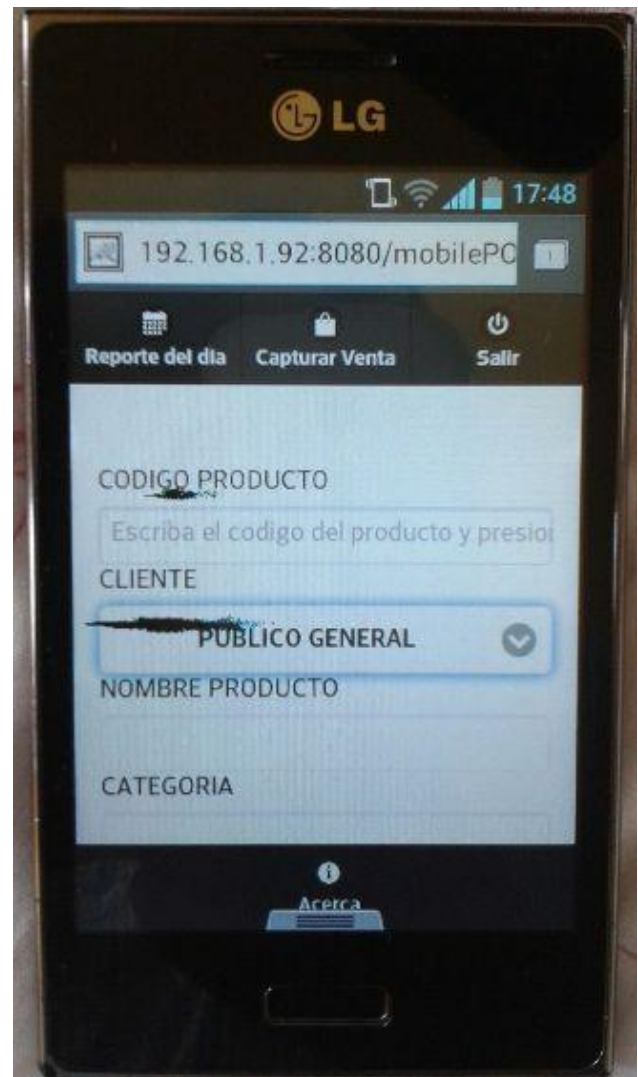


Figura 15: Pantalla nueva venta en dispositivo LG Android 4

### 4.3 INSTALACIÓN DE LA APLICACIÓN

Las instrucciones de instalación están detalladas en el archivo README.pdf

## 5. BIBLIOGRAFIA

Apache, T. (19 de Septiembre de 2014). *Tomcat*. Recuperado el 19 de Septiembre de 2014, de Tomcat: <http://tomcat.apache.org/download-70.cgi>

Coulouris, G., Dollimore, J., Kindberg, T., & Blair, G. (2012). *Distributed Systems, Concepts and Design* (Quina Edición ed.). United States of America: Addison-Wesley.

Mobile, j. (19 de Septiembre de 2014). *A Touch-Optimized Web Framework*. Recuperado el 12 de Septiembre de 2014, de A Touch-Optimized Web Framework: <http://jquerymobile.com/>