**ESCUELA POLITÉCNICA NACIONAL**

ESCUELA DE FORMACIÓN DE TECNÓLOGOS

**TECNICAS DE ANALISIS Y DISEÑO DE SISTEMAS INFORMATICOS**

|  |  |
| --- | --- |
|  | |
| ASIGNATURA: | Técnicas de análisis y diseño de sistemas informáticos |
| PROFESOR: | Ing. Ivonne Maldonado |
| PERÍODO ACADÉMICO: |  |
|  | |

**PROYECTO FINAL**

|  |  |
| --- | --- |
|  | |
| TÍTULO:  **SISTEMA** | |
| **ESTUDIANTE** | |
| Aguilar Estefanía  Coro Andrés  Iza Jorge | |
|  | |
|  | |
|  | |
| FECHA DE REALIZACIÓN: |  |
| FECHA DE ENTREGA: | 09 de agosto de 2018 |
| CALIFICACIÓN OBTENIDA: |  |
| FIRMA DEL PROFESOR: |  |
|  | |

Contenido

[1 Propósito 3](#_Toc521559953)

[2 Alcance 3](#_Toc521559954)

[3 Documento de referencia 3](#_Toc521559955)

[4 Definiciones importantes 3](#_Toc521559956)

[4.1 Conceptos generales 3](#_Toc521559957)

[4.2 Procesos de entrada y salida 3](#_Toc521559958)

[5 Descripción de módulos 4](#_Toc521559959)

[6 Diccionario de datos 5](#_Toc521559960)

[6.1 Modelo entidad-relación 5](#_Toc521559961)

[7 Instalación y configuración 8](#_Toc521559962)

[7.1 Requisitos generales pre-instalación 8](#_Toc521559963)

[7.2 Detalles del proceso de instalación 9](#_Toc521559964)

[7.3 Detalle de configuración de la aplicación 14](#_Toc521559965)

[7.3.1 Lista de contactos técnicos 14](#_Toc521559966)

# Propósito

Este sistema tiene como propósito ayudar al usuario (dueño del taller) a tener un mejor control de registros de sus colaboradores, proveedores y clientes, además de tener un mejor control de los estados de cuentas por cobrar y pagar.

Este manual proporcionar una guía para las principales actividades técnicas del sistema <nombre del sistema>.

# Alcance

En el presente documento se pretende mostrar las limitaciones, condiciones y requerimientos que se necesitan para analizar el correcto desarrollo del sistema, de igual manera se desea mostrar las características y diseños del producto final de software que brinde la solución y que cumpla a cabalidad los requerimientos mencionados en el presente documento.

Este software está enfocado para administrador (dueño del taller) y colaboradores secretaria(o) los cuales tendran conocimientos básicos de informática y contabilidad.

# Documento de referencia

Deben indicarse todos los documentos relacionados con la solución, y cuya lectura complemente los conceptos tratados. Por ejemplo:

# Definiciones importantes

## Conceptos generales

* **Activo:** persona que está disponible en el sistema y puede realizar cualquier actividad en el sistema.
* **Importe:** Cantidad de dinero que vale una mercancía o que figura en una factura, una cuenta, una compra.
* **Abono:** Pago de una cuenta o de un dinero que se debe.
* **Referencia:** Explicación o relación de un documento.

## Procesos de entrada y salida

|  |  |
| --- | --- |
| Entradas:   * Datos generales del cliente, empleado, proveedor. * Facturas (interfaz automático). * Pagos, depuraciones, etc. | Procesos:   * Calculo de antigüedad de saldos. * Calculo de interés. * Calculo de saldo cliente, proveedor. |
| Almacenamiento:   * Listado de clientes, proveedor, empleado. * Facturas. | Salidas:   * Reporte. * Estado de cuentas. |

# 

# Descripción de módulos

* Login

**Funcionalidad:** capturar valores de los campos de usuario y contraseña y compararlos con la base de datos si estos campos coinciden tiene el acceso.

**Propósito:** controlar el acceso de entradas al sistema.

**Dependencias funcionales:** depende de la base de datos.

* Menú

**Funcionalidad:** indicar y presentar las opciones o herramientas de la aplicación, dispuestas en menús desplegables.

**Propósito:** dar una mejor navegación entre las diferentes interfaces.

**Dependencias funcionales:** del login quien tuvo el acceso puede interactuar en el menú.

* Vehículo

**Funcionalidad:** registrar a los vehículos y listar todos los autos que ya han sido atendidos anteriormente en el taller.

**Propósito:** tener un manejo adecuado de los autos que ingresan al taller.

**Dependencias funcionales:** tener un dueño del vehículo (cliente).

* Cliente

**Funcionalidad:** registrar clientes que ingresan al taller y listar a todos los clientes antiguos.

**Propósito:** tener un control de quienes son nuestros clentes.

**Dependencias funcionales:** del login el usuario administrador puede registrar al cliente.

* Orden trabajo

**Funcionalidad:** registrar todas las ordenes de trabajo que se van a realizar al auto.

**Propósito:** tener un registro de servicios que se le realizan al auto.

**Dependencias funcionales:** del cliente y el automóvil.

* Proveedor

**Funcionalidad:** registrar todos los proveedores y listar a todos los principales proveedores del taller mecánico.

**Propósito:** listar a los principales proveedores del taller mecánico.

**Dependencias funcionales:** de la empresa y productos.

* Empleado

**Funcionalidad:** registrar a todos nuestros colaboradores que son los empleados y de igual manera listar a todos los empleados.

**Propósito:** listar a todos nuestros colaboradores.

**Dependencias funcionales:** de la empresa.

* Almacén

**Funcionalidad:** registrar productos y servicios.

**Propósito:** tener un control de stock de productos.

**Dependencias funcionales:** de los productos y servicios.

* Contabilidad

**Funcionalidad:** registrar facturas y cuentas por cobrar y pagar así también como los abonos de los clientes y proveedores.

**Propósito:** tener un control de las cuentas.

**Dependencias funcionales:** facturación y órdenes de trabajo.

* Reporte

**Funcionalidad:** reportar todas las órdenes de trabajo y facturas.

**Propósito:** entregar al cliente un comprobante de su orden o factura.

**Dependencias funcionales:** ordenes de trabajo y facturas.

* Salir

**Funcionalidad:** dar por finalizado a la aplicación cuando ya no se la vaya a utilizar.

**Propósito:** salir del menú principal.

**Dependencias funcionales:** cierre de sesión.

# Diccionario de datos

## Modelo entidad-relación

CREATE TABLE usuario

(id INT NOT NULL UNIQUE AUTO\_INCREMENT,

nombre VARCHAR(25) NOT NULL UNIQUE,

email VARCHAR(255) NOT NULL UNIQUE,

passw VARCHAR(255),

fecha\_registro DATE NOT NULL,

passadmin VARCHAR(255),

tipousuario VARCHAR (60) NOT NULL,

PRIMARY KEY (id)

);

CREATE TABLE empresa

(RucEmpre NUMERIC (20) UNIQUE NOT NULL,

NomEmpre VARCHAR(30) NOT NULL,

DirEmpre VARCHAR(200) NOT NULL,

TlfEmpre NUMERIC (10) NOT NULL,

PRIMARY KEY (RucEmpre)

);

CREATE TABLE empleado

(RucEmpl NUMERIC (20) UNIQUE NOT NULL,

NomEmpl VARCHAR (60) NOT NULL,

ApelEmpl VARCHAR(30) NOT NULL,

DirEmpl VARCHAR (60) NOT NULL,

TlfEmpl NUMERIC (10),

EmailEmpl VARCHAR(255)UNIQUE,

Activo TINYINT NOT NULL,

RucEmpre NUMERIC (20) NOT NULL,

FOREIGN KEY (RucEmpre) REFERENCES empresa (RucEmpre)

ON UPDATE CASCADE,

PRIMARY KEY (RucEmpl)

);

CREATE TABLE proveedor

(RucProv NUMERIC (20) UNIQUE NOT NULL,

NomProv VARCHAR(30) NOT NULL,

ApelProv VARCHAR(30) NOT NULL,

DirProv VARCHAR(30) NOT NULL,

TlfProv NUMERIC (10) NOT NULL,

EmailProv VARCHAR(255) UNIQUE,

Activo TINYINT NOT NULL,

PRIMARY KEY (RucProv)

);

CREATE TABLE producto

(IdProd INT UNIQUE AUTO\_INCREMENT NOT NULL,

Descripcion VARCHAR (60) not null,

Cantidad INT not null,

CostoUnit DECIMAL (10,2) not null,

PrecioUnit DECIMAL (10,2) not null,

RucProv NUMERIC (20) NOT NULL,

FOREIGN KEY (RucProv) REFERENCES proveedor (RucProv),

PRIMARY KEY (IdProd)

);

CREATE TABLE cliente

(RucCli NUMERIC (20) UNIQUE NOT NULL,

NomCli VARCHAR (60) NOT NULL,

ApelCli VARCHAR(30) NOT NULL,

DirCli VARCHAR (60) NOT NULL,

TlfCli NUMERIC (10),

EmailCli VARCHAR(255) UNIQUE,

Activo TINYINT NOT NULL,

PRIMARY KEY (RucCli)

);

CREATE TABLE vehiculo

(Placa VARCHAR (10) UNIQUE NOT NULL,

Tipo VARCHAR (60) NOT NULL,

Modelo VARCHAR (60) NOT NULL,

Color VARCHAR (60),

RucCli NUMERIC (20) NOT NULL,

FOREIGN KEY (RucCli) REFERENCES cliente (RucCli)

ON UPDATE CASCADE,

PRIMARY KEY (Placa)

);

CREATE TABLE servicio

(IdServ INT UNIQUE AUTO\_INCREMENT NOT NULL,

Descripcion VARCHAR (60) not null,

PrecioServ DECIMAL (10,2) not null,

PRIMARY KEY (IdServ)

);

CREATE TABLE empleado\_encargado

(IdCargo NUMERIC (10),

RucEmpl NUMERIC (20),

Placa VARCHAR (10) NOT NULL,

FOREIGN KEY (Placa) REFERENCES vehiculo (Placa)

ON UPDATE CASCADE,

FOREIGN KEY (RucEmpl) REFERENCES empleado (RucEmpl)

ON UPDATE CASCADE,

PRIMARY KEY (IdCargo)

);

CREATE TABLE orden\_trabajo

(IdOrden NUMERIC (10) UNIQUE NOT NULL,

RucCli NUMERIC (20) NOT NULL,

Placa VARCHAR (10) NOT NULL,

FechaHora\_recepcion date NOT NULL,

Kilometraje NUMERIC (20) NOT NULL,

Mano\_de\_obra DECIMAL (10,2) NOT NULL,

Otros DECIMAL (10,2) NOT NULL,

Subtotal DECIMAL (10,2) NOT NULL,

Iva DECIMAL (10,2) NOT NULL,

Total DECIMAL (10,2) NOT NULL,

facturado varchar (2) NULL,

FOREIGN KEY (RucCli) REFERENCES cliente (RucCli)

ON UPDATE CASCADE,

FOREIGN KEY (Placa) REFERENCES vehiculo (Placa)

ON UPDATE CASCADE,

PRIMARY KEY (IdOrden)

);

CREATE TABLE det\_ordenServicio

(IdOrdenServi NUMERIC (10) NOT NULL,

IdServ INT,

IdOrden NUMERIC (10),

FOREIGN KEY (IdServ) REFERENCES servicio (IdServ)

ON UPDATE CASCADE,

FOREIGN KEY (IdOrden) REFERENCES orden\_trabajo (IdOrden)

ON UPDATE CASCADE,

PRIMARY KEY (IdOrdenServi)

);

CREATE TABLE det\_ordenProducto

(IdOrdenProducto NUMERIC (10) NOT NULL,

IdProd INT,

IdOrden NUMERIC (10),

FOREIGN KEY (IdProd) REFERENCES servicio (IdServ)

ON UPDATE CASCADE,

FOREIGN KEY (IdProd) REFERENCES producto (IdProd)

ON UPDATE CASCADE,

PRIMARY KEY (IdOrdenProducto)

);

CREATE TABLE cab\_fact

(NoFact INT ,

RucCli NUMERIC (20) NOT NULL,

fechaFact date not null,

IdOrden NUMERIC (10) NOT NULL,

RucEmpre NUMERIC (20) null,

Subtotal DECIMAL (10,2) NOT NULL,

Iva DECIMAL (10,2) NOT NULL,

Total DECIMAL (10,2) NOT NULL,

FOREIGN KEY (RucCli) REFERENCES cliente (RucCli),

FOREIGN KEY (RucEmpre) REFERENCES empresa (RucEmpre),

FOREIGN KEY (IdOrden) REFERENCES orden\_trabajo (IdOrden),

PRIMARY KEY (NoFact)

);

CREATE TABLE cuenta\_x\_cobrar

(CodCuenta INT UNIQUE AUTO\_INCREMENT NOT NULL,

Importe DECIMAL (10,2) NOT NULL,

NoFact INT NOT NULL,

fecha\_registro DATE NOT NULL,

FOREIGN KEY (NoFact) REFERENCES cab\_fact (NoFact),

PRIMARY KEY (CodCuenta)

);

CREATE TABLE cuenta\_x\_pagar

(CodCuenta INT NOT NULL,

Importe DECIMAL (10,2) NOT NULL,

NoFact INT NOT NULL,

fecha\_registro DATE NOT NULL,

RucEmpre NUMERIC (20),

RucProv NUMERIC (20) NOT NULL,

FOREIGN KEY (RucEmpre) REFERENCES empresa (RucEmpre),

FOREIGN KEY (RucProv) REFERENCES proveedor (RucProv),

PRIMARY KEY (CodCuenta)

);

# Instalación y configuración

Cada uno de estos apartados debe ser agrupado por servidor para garantizar su adecuada descripción. Debe quedar claro el inventario de software de cada equipo. Este detalle debe darse también para realizar la instalación de equipos- cliente de los servidores.

## Requisitos generales pre-instalación

Requisitos sistema operativo:

1. Windows 7 ( x32 y x64)
2. Windows 8 ( x32 y x64)
3. Windows 8.1 ( x32 y x64)
4. Windows 10 ( x32 y x64)

Requisitos de hadware:

1. Procesador: 1 Ghz or greater ( x32 y x64) Procesador with SSE2 instruction set
2. Memory: 4 GB RAM (32 bit) / 4GB RAM (60 bit)
3. Hard disk: 2.0 GB available
4. Graphics: Graphics hardware acceleration requires DirectX10 graphics card 1024x576 minimun resolution.

## Detalles del proceso de instalación

Para la administración de Base de datos hemos utilizado phpMyAdmin que es una herramienta escrita en PHP con la intención de manejar la administración de MySQL a través de páginas web, utilizando Internet.

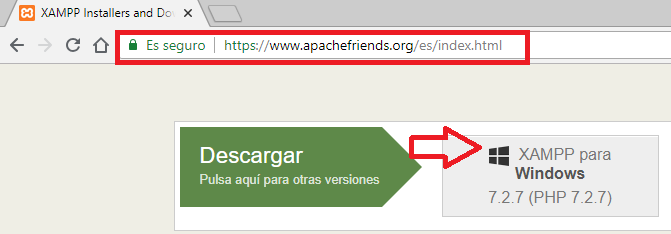
Hemos escogido este administrador de base de datos ya que nos permite crear y eliminar Bases de Datos, crear, eliminar y alterar tablas, borrar, editar y añadir campos, ejecutar cualquier sentencia SQL en Netbeans administrar claves en campos

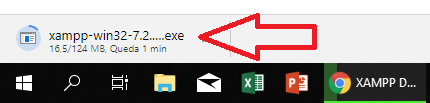
**Instalar Xampp**

XAMPP es un servidor web de plataforma, software libre, que consiste principalmente en el sistema de gestión de bases de datos MySQL, el servidor web Apache.

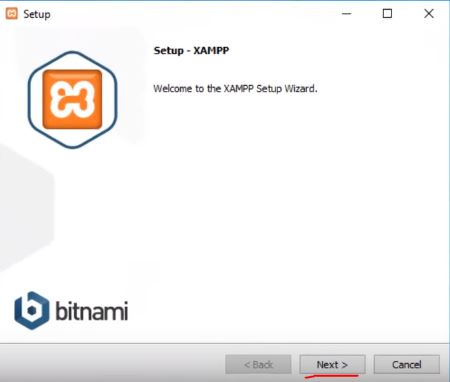
La razón que instalamos este servidor es porque XAMPP nos permite conectarnos con la base de datos ya que para poder realizar alguna modificación o queramos obtener alguna información de la mismo no podremos hacerlo si no levantamos el servidor XAMPP.

1. Descargamos el xampp de la página oficial la versión más actual para Windows.

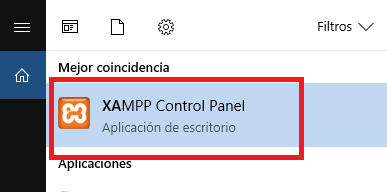




1. Instalamos Xampp hasta finalizar damos siguiente hasta finalizar.

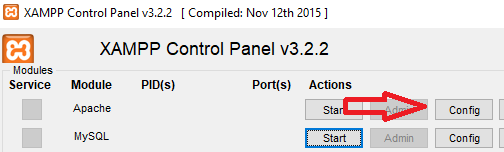


1. Buscamos en el ordenador si ya está el icono de xampp

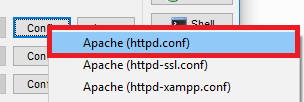


**Configuración del servidor XAMPP**

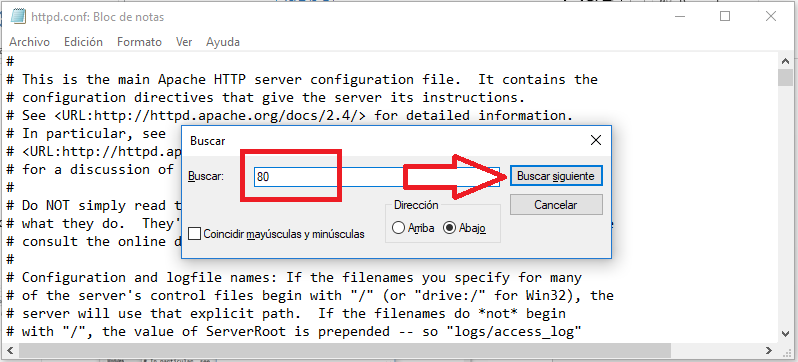
1. Antes de dar clic en el botón Start en el modulo de Apache de XAMPP debemos antes configurar al servidor dando clic en config del servidor Apache.



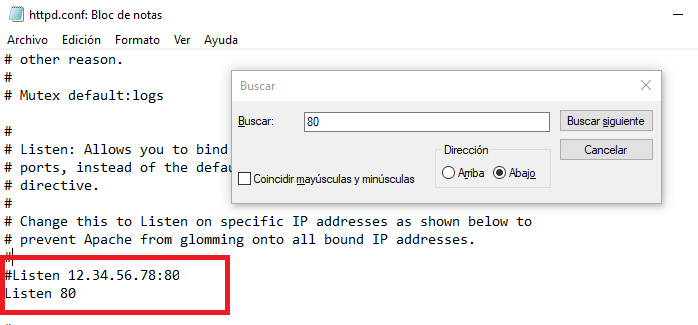
1. Damos clic en el botón conf y seleccionamos la primera opción Apache (httpd.conf)

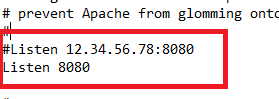


1. Con el teclado presionamos las tecla CTRL + B y nos aparecerá un recuadro de búsqueda en el cual pondremos 80 y damos clic en buscar siguiente hasta encontrar el puerto 80

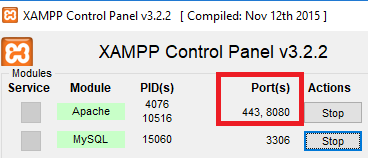


1. Cambiamos el puerto por 80 por 8080





1. Finalmente cerramos el bloc de notas y damos clic en Start en los botones de los modulos Apache y MYSQL y como nos podemos dar cuenta inicializa sin ningún problema por el puerto 8080.

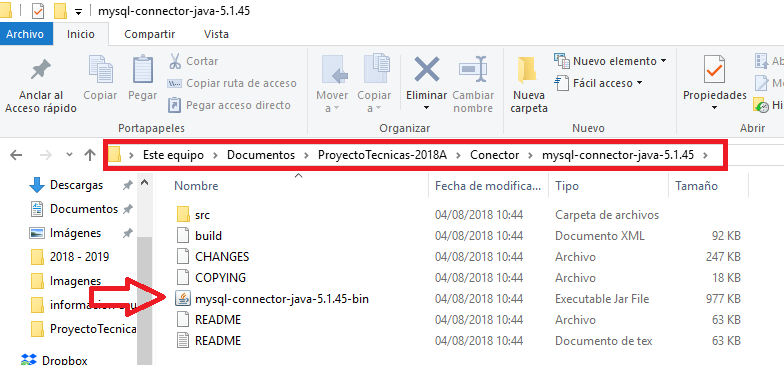


**Instalar conecctor**

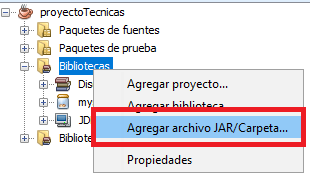
1. Procederemos con la instalación del driver que nos va a permitir la conexión de nuestra base de datos con NetBeans para ello ingresaremos a la página de dev.mysql/ donde descargaremos la versión 5.1.45 de3.6 M “Conector J 5.1.45”. la versión puede cambiar.



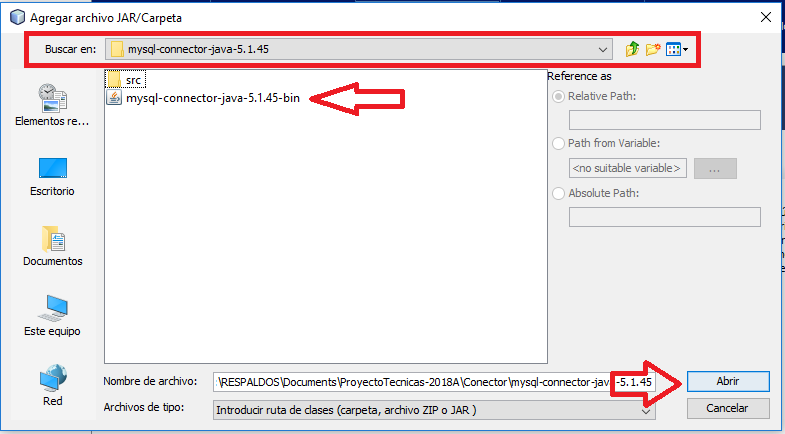
1. El driver se descargará en un archivo .rar lo cual deberemos descomprimirlo y de preferencia ubicarlos dentro de la carpeta donde se encuentre el proyecto que realizaremos en Netbeans.



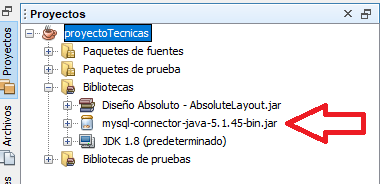
1. Abrimos el netbeans y cargamos el proyecto a continuación nos dirigimos a la carpeta del proyecto llamada bibliotecas y damos click derecho y buscamos la opción: Agregar archivo JAR/Carpeta.



1. Buscamos el directorio donde guardamos el conector que fue el caso en el mismo proyecto en la carpeta conector una vez seleccionamos el conector y damos clic en abrir.



1. Finalmente se cargara el conector que nos servirá para hacer conexión con la base de datos del servidor XAMPP



## Detalle de configuración de la aplicación

### Lista de contactos técnicos

|  |  |  |  |
| --- | --- | --- | --- |
| **Nombre** | **Empresa/Unidad Ejecutora** | **Módulo que atiende** | **Teléfonos** |
| Aguilar Estefanía | SoftSolution AEJ | * Registro /listado cliente * Registro /listado empleado * Registro /listado proveedor | 0984790230 |
| Coro Andrés | SoftSolution AEJ | * Ordenes de trabajo * Facturación * Registro /listado productos * Registro /listado servicios | 0984999817 |
| Iza Jorge | SoftSolution AEJ | * Login * Contabilidad * Cuentas por cobrar y cuentas por pagar * Registro /listado vehiculos | 0992245389 |