



**UNIVERSIDAD
ANDRÉS BELLO**

Facultad de ingeniería
Ingeniería Civil informática
Ingeniería en software

“Proyecto APP móvil Joyería Fevida”

Hito 1

Grupo 8

Integrantes: Matias Cerda.
Rafael Comnte.
Victor Martinez.
Vicente Tapia.

Profesores: Pablo Schwarzenberg.
Gustavo Gatica.

Índice

1. Introducción	Pág 2
2. Planificación	Pág 3
2.1 Backlog	Pág 3
2.2 Mapas de Historia	Pág 4
3. Gestión de Requisitos	Pág 5
4. Modelo de Base de datos	Pág 6
5. Modelo 4+1	Pág 6
5.1 4+1: Diagrama de clases	Pág 6
5.2 4+1: Diagrama de Actividades	Pág 7
5.3 4+1: Diagrama de Componentes	Pág 8
5.4 4+1: Diagrama de Deployment	Pág 8
6. Código	Pág 9
6.1 Código Fuente HTML	Pág 9
6.2 Código Fuente SQL	Pág 14
7. Diseño casos de Pruebas	Pág 20

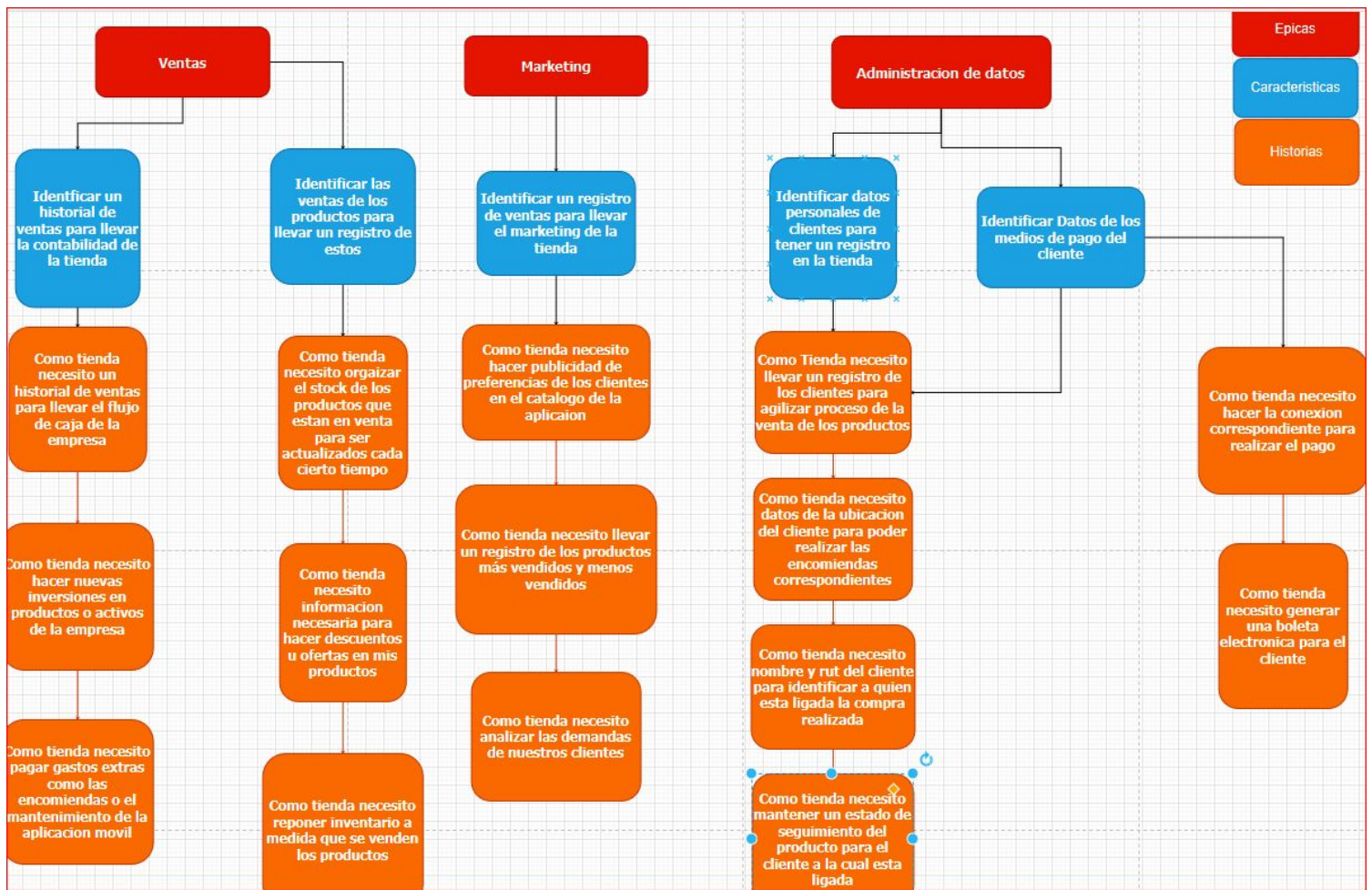
Introducción

En la actualidad, las joyerías han sido un lugar único para comprar accesorios de lujo, teniendo como limitación la cantidad de gente que abarcan con la tienda física. Con la aplicación a desarrollar “Joyas FevidaAPP” logramos llegar a una mayor cantidad de clientes (regiones) que posean un dispositivo móvil, permitir la visualización de productos como también solicitar cotizaciones, incluso la compra de estos artículos desde la comodidad del hogar.

Planificación

ID	Requerimientos	Prioridad	Estimación de esfuerzo (SP)	Sprint (Semanal)	Prueba
	Esquematizar la App para tener una idea de como empezar a desarrollarla	1	4		Tener todo planeado con anterioridad
	Programar la interfaz en HTML y CSS	2	2		Poder entrar a la App sin saltar errores y poder ver el banner
	Crear la seccion de Crear Cuenta en HTML y CSS	3	2		Poder saltar desde la interfaz hasta "Crear cuenta" al presionar el boton del mismo nombre
	Crear la seccion de Login en HTML y CSS	3	2		Poder saltar desde la interfaz hasta "Login" al presionar el boton del mismo nombre
	Crear una base de datos para almacenar a los usuarios y Articulos	4	4		Poder hacer consultas de los usuarios y articulos de prueba
	Crear la seccion Productos	5	3		Poder saltar desde la interfaz hasta "Tienda" ya sea por el boton "Tienda" o por el
	Crear la seccion Contactos	5	1		Poder saltar desde la interfaz hasta "Contactos" y visualizar la informacion pertinente
	Crear la seccion preguntas frecuentes	5	2		Poder saltar desde la interfaz hasta "Preguntas frecuentes" y visualizar la informacion pertinente
	Conectar la base de datos con la App	6	5		Si agregas los datos al crear cuenta, estos se guardaran en una base de datos
	Agregar las restricciones pertinentes para Crear Cuenta	7	3		no se puede ingresar correos o usuarios ya registrados, los correos sin @ no funcionan, las dos contraseñas
	Agregar Las restricciones pertinentes para Login	7	3		ingresadas deben coincidir estar almacenadas en la base de datos, alineados con el
	App capaz de Crear y logear cuentas correctamente	8	3		Si no existe tal usuario se puede crear desde la app en "Crear Cuenta" si ya existe en la base de datos se logeara
	Importar los productos de la base de datos a la App	8	3		Poder visualizar los productos de la base de datos
	Estetizar la pagina de Productos	9	2		la seccion de productos se ve bien y es intuitiva
	Programar el buscador	10	3		Al presionar "al carrito" la Id del articulo se almacenara en una lista
	Crear la lista de articulo en carrito	10	4		Al presionar "al carrito" la Id del articulo se almacenara en una lista
	Agregar un contador para el numero de articulos en la lista del carrito	11	1		Al presionar "al carrito" el contador aumentara en 1 su valor actual
	Llamar el contador en el banner	12	2		Al agregar un articulo al carrito la imagen del carrito se actualizara con la cantidad actual de articulos en la lista
	Importar correctamente los datos de la lista a la seccion de vista del carrito		3		los datos ya se pueden llamar correctamente a la seccion "carrito"
	Agregar todas las funciones necesarias a la seccion "carrito" (quitar del carro, ir a pagar, visualizar precios individuales y finales)	13	4		Llegar hasta la seccion de elegir opcion de pago sin contratiempos, poder visualizar
	Actualizar el contador del carrito, en la vista.	14	1		los precios y quitar productos
	Actualizar el contador del carrito, post-compra	14	1		El contador del carrito decrecera si quitas un
	La app debe disponer de algun medio de pago	15	4		El contador del carrito decrecera si quitas concretas
AD21	Disponer de los datos del cliente para saber a quien esta dirigida la compra	15	2		la compra
AD13	Disponer de los datos del cliente para poder realizar las encomiendas correspondientes	15	2		Hacer la conexión correspondiente
AD12	Generar una boleta para las compras	16	3		para realizar el pago
AD22	Disponer de envios de productos y de una app estable	17	3		Al crear la cuenta se pedirán estos datos
V13	Hacer publicidad personalizada para los clientes	18	4		Al crear la cuenta se pedirán estos datos
M11	Llevar un registro de los productos más vendidos y menos vendidos	19	2		Generar una boleta electronica para el cliente
M12	Hacer descuentos y ofertas de los productos	20	2		Pagar gastos de encomienda y mantenimiento de la App
V22					Ingresar cookies a los clientes para poder personalizar las publicidades que vera
					Analizar las ventas de los productos
					Precios más bajos por tiempo limitado y combos de productos

Mapa de historias



2. Gestión de Requisitos

Objetivo general: Desarrollar una aplicación móvil estilo e-commerce para la joyería “fevida”.

Objetivos específicos:

- Aumentar la cantidad de interesados en un 50% en el periodo del primer semestre, luego de lanzar la aplicación.
- Incrementar las ventas en un 30% dentro del primer año, luego de lanzar la aplicación.
- Aumentar el stock de mercadería internacional en un 20% al cabo de 9 meses, luego de lanzar la aplicación.
- Incrementar las ventas al por mayor en un 10% en un plazo de 2 años, luego de lanzada la aplicación.
- Mejorar el tiempo de respuesta de las cotizaciones en tiempo real otorgado por la aplicación. en el primer trimestre luego de lanzar la aplicación.

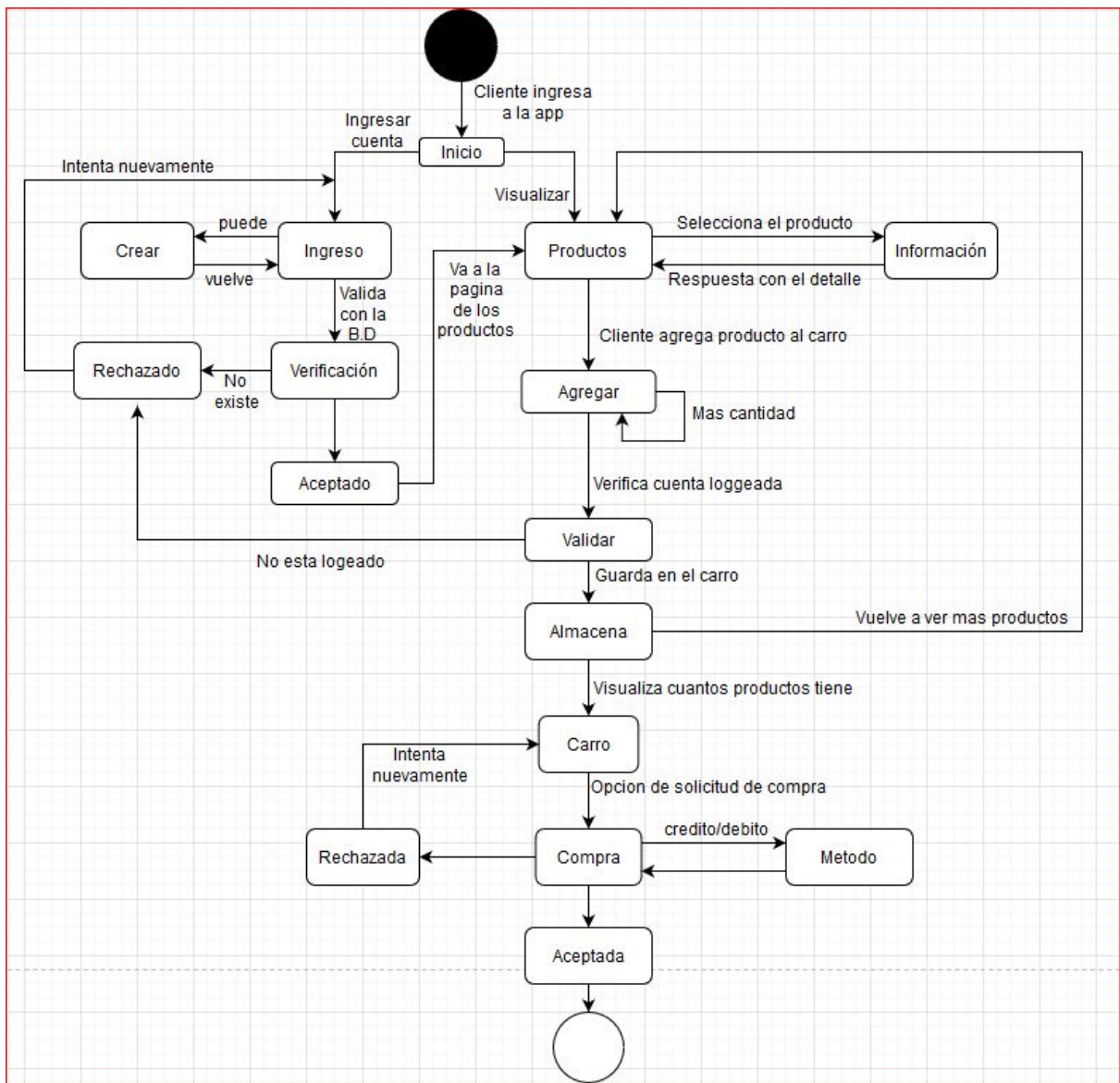
Descripción del producto: Nuestro cliente quiere implementar un sistema de aplicativo móvil para la venta, cotizaciones y manejo de stock de sus productos.

- La aplicación debe contar con cuentas para usuarios donde guardará la información necesaria, para luego enviar los productos por encomienda.
- La aplicación debe permitir editar la información del usuario en caso de cambio de domicilio.
- La aplicación debe contar con un apartado de búsqueda que permita ir mas facil a un producto específico.
- Debe tener categorías generales para los tipos de productos que posee la joyería.
- Debe tener ofertas en la pantalla de inicio de la aplicación.
- Al visualizar el producto este debe permitir elegir la cantidad, color, ver cantidad de stock disponible, la descripción del producto y obviamente agregar al carro.
- En el apartado del carro de compras se pueden eliminar o cambiar la cantidad del producto nuevamente, se debe poder realizar directamente la compra o enviar una cotización.
- Realizar el pago vía webpay (agregar sistema externo)

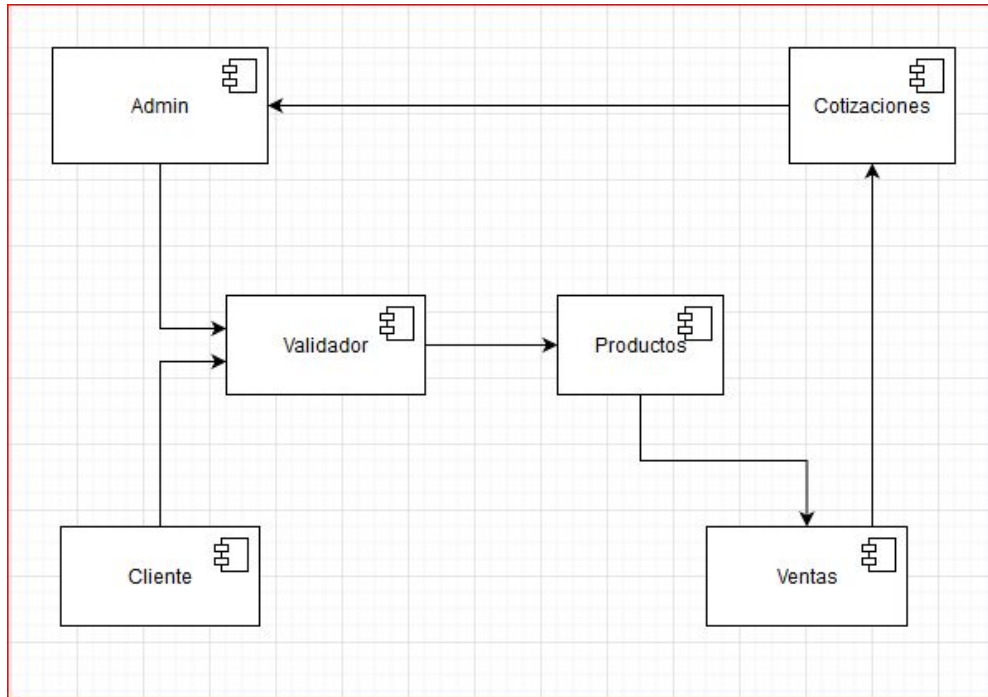
Propuestas por parte del equipo de trabajo aceptadas por el cliente:

- Ofertas personalizadas para clientes con compras habituales a cierto producto.
- Chat con administradores dentro de la aplicación.
- Modo daltonico para visualizar colores en los productos.
- En el perfil contar un con historial de compras.

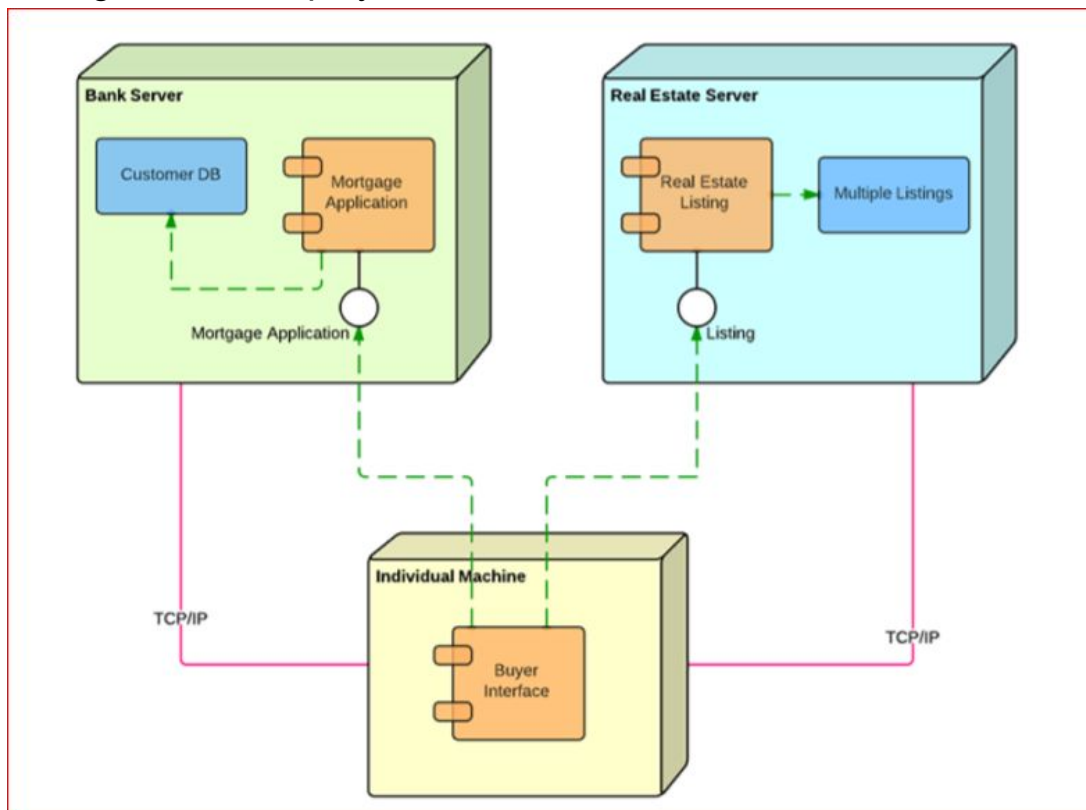
Modelo de Base de datos



4+1: Diagrama de Componentes



4+1: Diagrama de Deployment



Código Fuente

(Interfaz)

```
<!DOCTYPE html>
<html>
<head>
<title>Fevida</title>
<style>
body{
  margin: 0 0;
}
ul {
  list-style-type: none;
  margin: 0;
  padding: 0;
  overflow: hidden;
  background-color: #162B89;

  li {
    float: left;
  }

  li a, .dropbtn {
    display: inline-block;
    color: white;
    text-align: center;
    padding: 14px 16px;
    text-decoration: none;
  }

  li a:hover, .dropdown:hover .dropbtn {
    background-color: blue;
  }

  li.dropdown {
    display: inline-block;
  }

  .dropdown-content {
    display: none;
    position: absolute;
```

```

background-color: lightblue;
min-width: 10px;
box-shadow: 0px 8px 16px 0px violet;
z-index: 1;
}

.dropdown-content a {
  color: black;
  padding: 12px 16px;
  text-decoration: none;
  display: block;
  text-align: left;
}

.dropdown-content a:hover {background-color: lightblue}

.dropdown:hover .dropdown-content {
  display: block;
}

#p_solo{
  background-color: red;
}

div.left{
  float:left;
}

div.right{
  float:right;
}

div.center{
  float: center;
  padding: 14px;
}

</style>
</head>
<body>

<ul>
  <div class="left">
    <li></li>
  </div>
  <div class="left">

```

```

        <li class="dropdown">
            <a href="Fevida.html" class="dropbtn">Home</a>
        </li>
    <li class="dropdown">
        <a href="#" class="dropbtn">Tienda</a>
        <div class="dropdown-content">
            <a href="#">Todo</a>
            <a href="#">Collar</a>
            <a href="#">Pulsera</a>
            <a href="#">Anillo</a>
            <a href="#">Aro</a>
            <a href="#">Conjunto</a>
        </div>
    </li>
    <li>
        <div class="dropdown">
            <a href="#" class="dropbtn">Contacto</a>
        </div>
    </li>
    <li><a href="Preguntas Frecuentes">Preguntas Frecuentes</a></li>
</div>
<div class="center">
    <li><select name="Tipo de Contenido">
        <option value="Todo">Todo</option>
        <option value="Collares">Collar</option>
        <option value="Pulseras">Pulsera</option>
        <option value="Anillos">Anillo</option>
        <option value="Aros">Aro</option>
        <option value="Conjuntos">Conjunto</option>
    </select></li>
    <li><form action="/action_page.php">
        <input list="browsers">
        <input type="submit">
    </form>
    </li>
</div>

<div class="right">
    <li><a href="Crear_Cuenta.html">Crear Cuenta</a></li>
    <li><a href="Login.html">Iniciar Sesión</a></li>

</div>
</ul>

```

```

    <h1 style="text-align: center">¿Que Buscas?</h1>
    

</body>
</html>

```

(Crear Cuenta)

```

<!DOCTYPE html>

<html>
  <head>
    <title>Fevida</title>
    <style>
      body{
        margin: 0 0;
      }
      div.centrar{
        float: center;
        text-align: center;
        padding: 300px;
        background-color: lightblue;
      }

    </style>
  </head>
  <body>
    <div class="centrar">
      <form action="/action_page.php">
        E-mail:<br>
        <input type="text" name="E-mail" value="">
        <br><br>
        Username:<br><br>
        <input type="text" name="Username" value="">
        <br><br>
        Password:<br>
        <input type="text" name="Password" value="">
        <br>
        Repetir Password:<br>
        <input type="text" name="Password" value="">
        <br><br><br>
      </form>
    </div>
  </body>
</html>

```

```
        <input type="radio">
        <span>Acepto las condiciones de uso</span>
        <br><br>
        <input type="submit" value="Submit">
        </form>
    </div>

</body>
</html>
```


(Login)

(MySql)

```
-- phpMyAdmin SQL Dump
-- version 5.0.2
-- https://www.phpmyadmin.net/
--
-- Servidor: 127.0.0.1
-- Tiempo de generación: 08-10-2020 a las 03:30:22
-- Versión del servidor: 10.4.14-MariaDB
-- Versión de PHP: 7.4.10
```

```
SET SQL_MODE = "NO_AUTO_VALUE_ON_ZERO";
START TRANSACTION;
SET time_zone = "+00:00";
```

```
/*!40101 SET
@OLD_CHARACTER_SET_CLIENT=@@CHARACTER_SET_CLIENT */;
/*!40101 SET
@OLD_CHARACTER_SET_RESULTS=@@CHARACTER_SET_RESULTS */;
/*!40101 SET
@OLD_COLLATION_CONNECTION=@@COLLATION_CONNECTION */;
/*!40101 SET NAMES utf8mb4 */;
```

```
--
-- Base de datos: `ecommerce`
--
```

```
-- -----
```

```
--
-- Estructura de tabla para la tabla `clientes`
--
```

```
CREATE TABLE `clientes` (
  `id` int(11) NOT NULL,
  `email` varchar(200) NOT NULL,
  `pass` varchar(255) NOT NULL,
  `nombre` varchar(255) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;
```

```
-- -----
```

```

--
-- Estructura de tabla para la tabla `detalleventas`
--

CREATE TABLE `detalleventas` (
  `id` int(11) NOT NULL,
  `idProd` int(5) NOT NULL,
  `idVenta` int(5) NOT NULL,
  `cantidad` int(5) NOT NULL,
  `precio` double NOT NULL,
  `subTotal` double NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;

-----

--
-- Estructura de tabla para la tabla `productos`
--

CREATE TABLE `productos` (
  `id` int(11) NOT NULL,
  `nombre` varchar(200) NOT NULL,
  `precio` double NOT NULL,
  `existencia` int(5) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;

-----

--
-- Estructura de tabla para la tabla `usuarios`
--

CREATE TABLE `usuarios` (
  `id` int(11) NOT NULL,
  `email` varchar(200) NOT NULL,
  `pass` varchar(255) NOT NULL,
  `nombre` varchar(255) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;

--
-- Volcado de datos para la tabla `usuarios`
--

```

```
INSERT INTO `usuarios` (`id`, `email`, `pass`, `nombre`) VALUES
(1, 'email@prueba.cl', 'e10adc3949ba59abbe56e057f20f883e', 'Matias');
```

```
-- -----
```

```
--
-- Estructura de tabla para la tabla `ventas`
--
```

```
CREATE TABLE `ventas` (
  `id` int(11) NOT NULL,
  `idCli` int(5) NOT NULL,
  `fecha` datetime NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;
```

```
--
-- Índices para tablas volcadas
--
```

```
--
-- Indices de la tabla `clientes`
--
```

```
ALTER TABLE `clientes`
  ADD PRIMARY KEY (`id`);
```

```
--
-- Indices de la tabla `detalleventas`
--
```

```
ALTER TABLE `detalleventas`
  ADD PRIMARY KEY (`id`),
  ADD KEY `fkIdProd` (`idProd`),
  ADD KEY `fkIdVenta` (`idVenta`);
```

```
--
-- Indices de la tabla `productos`
--
```

```
ALTER TABLE `productos`
  ADD PRIMARY KEY (`id`);
```

```
--
-- Indices de la tabla `usuarios`
--
```

```
ALTER TABLE `usuarios`
  ADD PRIMARY KEY (`id`);
```

```

--
-- Indices de la tabla `ventas`
--
ALTER TABLE `ventas`
  ADD PRIMARY KEY (`id`),
  ADD KEY `fkIdCli` (`idCli`);

--
-- AUTO_INCREMENT de las tablas volcadas
--

--
-- AUTO_INCREMENT de la tabla `clientes`
--
ALTER TABLE `clientes`
  MODIFY `id` int(11) NOT NULL AUTO_INCREMENT;

--
-- AUTO_INCREMENT de la tabla `detalleventas`
--
ALTER TABLE `detalleventas`
  MODIFY `id` int(11) NOT NULL AUTO_INCREMENT;

--
-- AUTO_INCREMENT de la tabla `productos`
--
ALTER TABLE `productos`
  MODIFY `id` int(11) NOT NULL AUTO_INCREMENT;

--
-- AUTO_INCREMENT de la tabla `usuarios`
--
ALTER TABLE `usuarios`
  MODIFY `id` int(11) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=2;

--
-- AUTO_INCREMENT de la tabla `ventas`
--
ALTER TABLE `ventas`
  MODIFY `id` int(11) NOT NULL AUTO_INCREMENT;

--
-- Restricciones para tablas volcadas

```

```

--

--
-- Filtros para la tabla `detalleventas`
--
ALTER TABLE `detalleventas`
  ADD CONSTRAINT `idProd` FOREIGN KEY (`idProd`) REFERENCES `productos`
  (`id`) ON DELETE CASCADE ON UPDATE CASCADE,
  ADD CONSTRAINT `idVenta` FOREIGN KEY (`idVenta`) REFERENCES `ventas`
  (`id`) ON DELETE CASCADE ON UPDATE CASCADE;

--
-- Filtros para la tabla `ventas`
--
ALTER TABLE `ventas`
  ADD CONSTRAINT `idCli` FOREIGN KEY (`idCli`) REFERENCES `clientes` (`id`)
  ON DELETE CASCADE ON UPDATE CASCADE;
COMMIT;

/*!40101 SET CHARACTER_SET_CLIENT=@OLD_CHARACTER_SET_CLIENT */;
/*!40101 SET
CHARACTER_SET_RESULTS=@OLD_CHARACTER_SET_RESULTS */;
/*!40101 SET COLLATION_CONNECTION=@OLD_COLLATION_CONNECTION */;

```

Diseños casos de Pruebas

