

BASES DE DATOS

2CM12

---

# Manage Your Cinema

---

**ALUMNOS:**

Maya Rocha Luis Emmanuel  
Rosas Hernández Óscar Andrés  
Dominguez Lopez Humberto  
Hernández Ruiz Rafael

**PROFESOR:**

Euler Hernández Contreras

Reporte de Bases de Datos

# Índice general

<b>1. Primer Parte</b>	<b>3</b>
1.1. Introducción . . . . .	4
1.2. Problemática . . . . .	6
1.3. Objetivos . . . . .	7
1.3.1. Objetivo General: . . . . .	7
1.3.2. Objetivos Particulares: . . . . .	7
1.4. Requerimientos . . . . .	8
1.4.1. Reglas de Negocio . . . . .	8
1.4.2. Requerimientos Básicos . . . . .	9
1.4.3. Requerimientos Funcionales . . . . .	9
1.4.4. Requerimientos No Funcionales . . . . .	10
1.5. Diagrama de Contexto . . . . .	11
1.6. Análisis de Riesgos . . . . .	12
1.6.1. Riesgos en General . . . . .	12
1.6.2. Hoja de Riesgos . . . . .	13
<b>2. Segunda Parte</b>	<b>14</b>
2.1. Entidad Relación . . . . .	15
2.2. Modelo Relacional . . . . .	16
2.2.1. Diagrama Relacional con MySQLWorkBench . . . . .	16
2.2.2. Código en SQL . . . . .	17
2.3. Diccionario de Datos . . . . .	21
2.4. Diagrama de Casos de Uso . . . . .	25

---

2.5. Diagrama Secuencial . . . . .	26
2.6. Avance de la Interfaz . . . . .	31

# Capítulo 1

## Primer Parte

## 1.1. Introducción

Como proyecto final de Bases de Datos nos disponemos a crear un Sistema basado en Web (usando sobretodo PHP y el framework MaterializeCSS) para poder gestionar un Cine:

- Películas
- Horarios
- Dulceria
- Empleados
- Organización en general

Toda la información del sistema será almacenada en una base de datos creada con MySQL y que se comunicará con nuestra aplicación web usando las funciones de PHP con MySQL.



Figura 1.1: Boceto de FrontEnd de la aplicación web

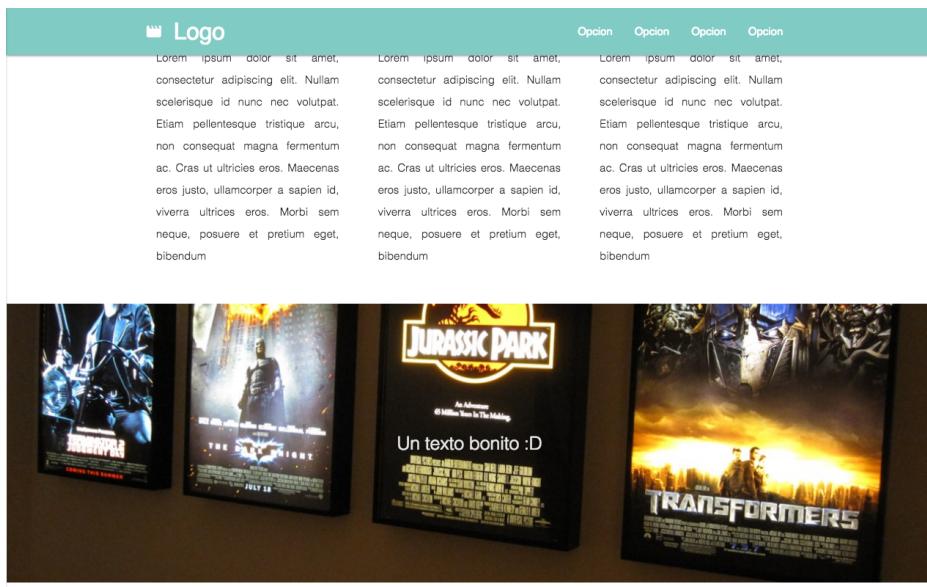


Figura 1.2: Boceto de FrontEnd de la aplicación web

## 1.2. Problemática

Buscamos crear un sistema que permita organizar toda la administración de un Cine y todas sus subdivisiones de manera consistente, confiable y segura.

Queremos que dicha aplicación web pueda ser usada con gran facilidad, importando lo menos posible el sistema operativo que usen nuestros clientes para usar la aplicación web así como el dispositivo que utilicen, sea un tablet, un celular o una computadora (de cualquier gama).

Buscamos que sea de fácil mejoramiento y que podamos actualizar y corregir de la manera más rápida cualquier error que tuviera nuestra aplicación web.

Buscamos que la información que tenga nuestra aplicación web en la base de datos sea de fácil acceso para la creación posterior de nuevas interfaces y aplicaciones así como para su mantenimiento y escalabilidad.

Buscamos la mayor seguridad en los datos, de tal manera que sean las más confiables posibles.

Buscamos que nuestra aplicación web permita diferentes perfiles de usuarios, para mejorar la confiabilidad y seguridad del sistema permitiendo a cierto tipo de usuarios solo hacer ciertas actividades y tener la capacidad de leer / modificar ciertos datos dentro del sistema.

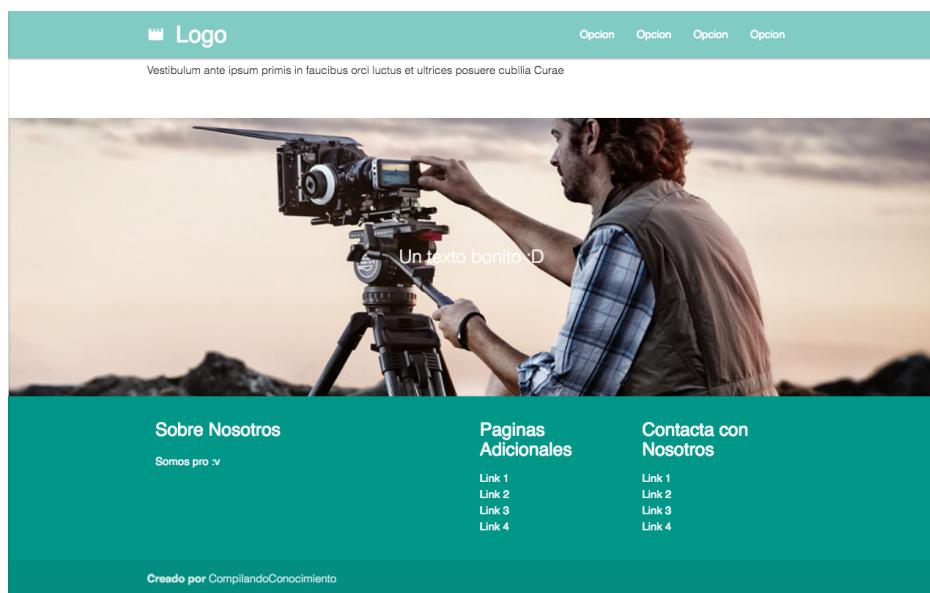


Figura 1.3: Boceto de FrontEnd de la aplicación web

## 1.3. Objetivos

### 1.3.1. Objetivo General:

Desarrollar un sistema que permita gestionar varios de los procesos necesarios en un cine; como son la dulcería, la venta de boletos y los roles de empleados.

### 1.3.2. Objetivos Particulares:

- A través del desarrollo de un sistema gestionar los horarios de entrada y salida de los empleados en las diferentes áreas del cine.
- El sistema podrá gestionar los horarios y salas de las diferentes exhibiciones.
- El sistema gestionara la venta de boletos para las diferentes exhibiciones, además controlara la disponibilidad de boletos existentes para cada exhibición.
- El sistema controlara la dulcería, como la venta de los diferentes artículos, la existencia de estos y los proveedores.
- Se permitirá la creación de usuarios para restringir el acceso a la escritura y lectura de los diferentes datos, según el puesto que tenga el usuario en el cine.

## 1.4. Requerimientos

### 1.4.1. Reglas de Negocio

ID	Nombre	Descripción	Prioridad	Origen
RN1	Control de Empleados	Un gerente que lleva el control de los empleado(vendedores de duces boletos y los que limpian)	Alta	Propuestos
RN2	Pagos	El gerente se encarga de el pago a los proveedores	Media	Propuestos
RN3	Control de Salas	El gerente se encargar de determinar las salas para una pelicula y los horarios	Media	Propuestos
RN4	Control de Dulcería	Se debe de poder seleccionar los dulces y bebidas disponibles por ese día, se venden 5 tipos diferentes de combos, el cliente debe elegir cual de los 5 desea o puede no comprar ninguno. Existen 6 salas, las cuales a su vez tienen 30 butacas, por lo que debe controlarse el numero de personas que entran. Al tratarse de un cine popular, solo se exhiben peliculas nuevas, y las tarifas de licencia para exhibir películas, las cuales pueden ser muy altas, sobre todo para películas importantes de estreno (puedes contratar a agentes de cine para ayudar con el proceso de conseguir las películas y la aprobación para exhibirlas)	Media	Propuestos
RN4	Control de Capacidad		Baja	Propuestos

### 1.4.2. Requerimientos Básicos

ID	Nombre	Descripción	Prioridad	Origen
RB1	Seguridad	Todas las conexiones con la base de datos, así como la confiabilidad de los datos tienen que ser altas, para una mayor seguridad.	Alta	Propuestos
RB2	Compatibilidad de Navegadores	El sistema tiene que ser accesible desde los navegadores mas usados	Baja	Propuestos
RB3	Compatibilidad de Plataformas	El sistema puede ser usado desde diversas plataformas	Baja	Propuestos
RB4	Escalabilidad	La estructura tanto de la aplicación web como de la base de datos a nivel conceptual tiene que permitir el mantenimiento y la escalabilidad el proyecto.	Media	Propuestos
RB4	Roles	La aplicación web permite la existencia de distintos tipos de usuarios, con cada tipo las acciones disponibles cambian.	Media	Propuestos

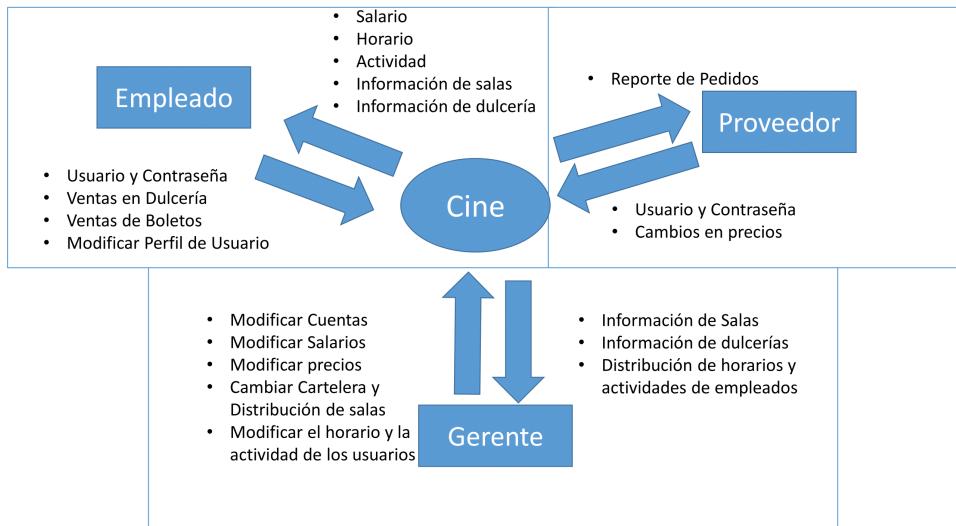
### 1.4.3. Requerimientos Funcionales

ID	Nombre	Descripción	Origen
RF1	Gestionar Sala	El Sistema permitirá al usuario gerente establecer los horarios y películas para una sala	RN2
RF2	Gestionar Empleados	El sistema permitirá establecer que actividad llevará a cabo cada empleado impiendole realizar otra que no sea la indicada	RN1
RF3	Control de Suministros	El sistema podrá llevar stock de todos los productos utilizados en venta o en mantenimiento del cine	RN3
RF4	Control de Entradas	El sistema deberá indicar el estatus de una sala al realizar la venta de un boleto en ella	RN6
RF5	Gestión de Dulcería	El sistema permitirá manejar las transacciones de venta en el departamento de dulcería	RN5

#### 1.4.4. Requerimientos No Funcionales

	Nombre	Descripción	Origen
RNF1	FrameWork	Sistema Desarrollado en: JQuery, MaterializeCSS	RB3
RNF2	Entorno de Trabajo de Aplicaciones	PHP, HTML, Javascript, CSS, XAMPP(Apache)	RB2
RNF3	Bases de Datos	MySQL Server	RB1
RNF4	Seguridad	Confidencia e integridad de los datos	RB1

## 1.5. Diagrama de Contexto



## 1.6. Análisis de Riesgos

### 1.6.1. Riesgos en General

Riesgos	Categorías	Probabilidad	Impacto
Fecha de Entrega será apretada	Riesgo de Calendario	60 %	Marginal
Fecha de Capacitación con las Herramientas	Riesgo de Apoyo	50 %	Despreciable
Personal inexperto en el Sistema	Riesgo de Apoyo	70 %	Marginal
Mayor número de usuarios que el esperado	Riesgo de Apoyo	30 %	Crítico
Personal renunciando al proyecto	Riesgo de Apoyo	40 %	Crítico
Personal accidentado	Riesgo de Apoyo	10 %	Marginal
Usuario finales ser resistentes a la interfaz	Riesgo de Rendimiento	20 %	Marginal
Implementación de algún requisito fallida	Riesgo de Rendimiento	70 %	Catastrófico

### 1.6.2. Hoja de Riesgos

Ahora veamos en más detalle como es los riesgos que tienen un impacto mayor al 30 %

#### Riesgo: El personal renuncia del proyecto por diferentes causas

**Hoja de Riesgo:**

- Probabilidad 40 %
- Impacto Crítico
- **Refinamiento / Contexto**
  - Los integrantes de equipo pueden decidir dejar de tomar la materia por falta de tiempo
  - Pueden tener malas calificaciones y decidir dejar de trabajar al final del semestre
  - Desacuerdos entre los integrantes del equipo que provoquen que alguno abandone el proyecto
- **Mitigación**
  - Todos los integrantes del equipo deben estar al pendiente de los últimos avances del proyecto
  - Todos deben tener copia del proyecto
  - Hacer comprensibles las diferentes implementaciones del proyecto
- **Plan de Contingencia**

Repartir las actividades que realizará en integrante que ha abandonado entre los integrantes restantes del equipo

#### Riesgo: Durante la implementación alguna parte resulta disfuncional

**Hoja de Riesgo:**

- Probabilidad 60 %
- Impacto Catastrófico
- **Refinamiento / Contexto**
  - En ocasiones puede haber errores en la implementación de la aplicación web
  - Incompatibilidad entre navegadores o entre sistemas operativos
  - Se ignora alguno requisito durante la implementación
- **Mitigación**
  - Mantener siempre informados a los integrantes de los requisitos del proyecto
  - Comunicar entre los integrantes del equipo la manera en que se implementará el proyecto para evitar incompatibilidades
  - Realizar un código comentado y fácil de comprender para facilitar la detección de errores
- **Plan de Contingencia**

Tener una alternativa sencilla aunque poco eficiente para la implementación de los diferentes componentes del proyecto

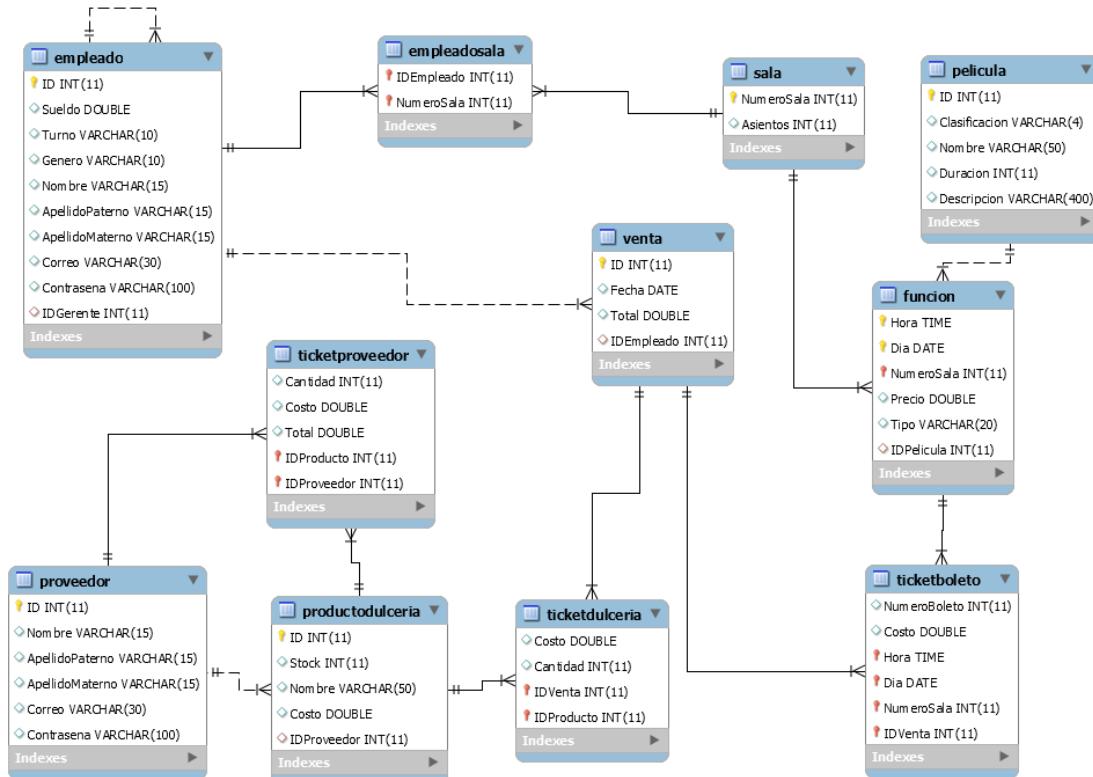
## Capítulo 2

### Segunda Parte

## 2.1. Entidad Relación

## 2.2. Modelo Relacional

### 2.2.1. Diagrama Relacional con MySQLWorkBench



### 2.2.2. Código en SQL

```
1 # =====
2 # ====== TABLES =====
3 # =====
4
5 DROP DATABASE IF EXISTS Proyect;
6 CREATE DATABASE Proyect;
7 USE Proyect;
8
9
10 CREATE TABLE Pelicula (
11     ID          INT NOT NULL PRIMARY KEY ,
12     Clasificacion    VARCHAR(4) ,
13     Nombre        VARCHAR(50) ,
14     Duracion      INT ,
15     Descripcion   VARCHAR(400)
16 );
17
18
19 CREATE TABLE Sala (
20     NumeroSala    INT NOT NULL PRIMARY KEY ,
21     Asientos       INT
22 );
23
24
25
26 CREATE TABLE Empleado (
27     ID          INT NOT NULL PRIMARY KEY ,
28     Sueldo       REAL ,
29     Turno        VARCHAR(10) ,
30     Genero       VARCHAR(10) ,
31     Nombre        VARCHAR(15) ,
32     ApellidoPaterno VARCHAR(15) ,
33     ApellidoMaterno  VARCHAR(15) ,
34     Correo        VARCHAR(30) ,
35     Contrasena    VARCHAR(100) ,
36     IDGerente     INT ,
37
38     FOREIGN KEY (IDGerente)
39         REFERENCES Empleado(ID)
40         ON DELETE CASCADE
41         ON UPDATE CASCADE
42
43 );
```

```
1 CREATE TABLE EmpleadoSala (
2     IDEmppleado      INT ,
3     NumeroSala       INT ,
4
5     FOREIGN KEY (IDEmpleado)
6         REFERENCES Empleado(ID)
7             ON DELETE CASCADE
8             ON UPDATE CASCADE ,
9
10    FOREIGN KEY (NumeroSala)
11        REFERENCES Sala(NumeroSala)
12            ON DELETE CASCADE
13            ON UPDATE CASCADE ,
14
15    PRIMARY KEY (IDEmpleado , NumeroSala)
16 );
17
18
19 CREATE TABLE Proveedor (
20     ID              INT NOT NULL PRIMARY KEY ,
21     Nombre          VARCHAR(15) ,
22     ApellidoPaterno VARCHAR(15) ,
23     ApellidoMaterno VARCHAR(15) ,
24     Correo          VARCHAR(30) ,
25     Contrasena      VARCHAR(100)
26 );
27
28
29 CREATE TABLE Funcion (
30     Hora           TIME ,
31     Dia            DATE ,
32     NumeroSala    INT ,
33     Precio         REAL ,
34     Tipo           VARCHAR(20) ,
35     IDPelicula    INT ,
36
37     FOREIGN KEY (NumeroSala)
38         REFERENCES Sala(NumeroSala) ,
39
40     FOREIGN KEY (IDPelicula)
41         REFERENCES Pelicula (ID)
42             ON DELETE CASCADE
43             ON UPDATE CASCADE ,
44
45     PRIMARY KEY (Hora , Dia , NumeroSala)
46 );
```

```
1 CREATE TABLE Venta (
2     ID          INT NOT NULL PRIMARY KEY ,
3     Fecha       DATE ,
4     Total       REAL ,
5     IDEmppleado INT ,
6
7     FOREIGN KEY (IDEmpleado)
8         REFERENCES Empleado (ID)
9             ON DELETE CASCADE
10            ON UPDATE CASCADE
11 );
12
13
14 CREATE TABLE TicketBoleto (
15     NumeroBoleto    INT ,
16     Costo           REAL ,
17     Hora            TIME ,
18     Dia             DATE ,
19     NumeroSala     INT ,
20     IDVenta         INT ,
21
22     FOREIGN KEY (IDVenta)
23         REFERENCES Venta (ID)
24             ON DELETE CASCADE
25             ON UPDATE CASCADE ,
26
27     FOREIGN KEY (Hora, Dia, NumeroSala)
28         REFERENCES Funcion (Hora, Dia, NumeroSala)
29             ON DELETE CASCADE
30             ON UPDATE CASCADE ,
31
32     PRIMARY KEY (IDVenta, Hora, Dia, NumeroSala)
33 );
34
35
36 CREATE TABLE ProductoDulceria (
37     ID          INT NOT NULL PRIMARY KEY ,
38     Stock       INT ,
39     Nombre      VARCHAR(50) ,
40     Costo       REAL ,
41     IDProveedor INT ,
42
43     FOREIGN KEY (IDProveedor)
44         REFERENCES Proveedor (ID)
45             ON DELETE CASCADE
46             ON UPDATE CASCADE
47 );
```

```
1 CREATE TABLE TicketDulceria (
2     Costo          REAL ,
3     Cantidad       INT ,
4     IDVenta        INT ,
5     IDProducto     INT ,
6
7     FOREIGN KEY (IDVenta)
8         REFERENCES Venta (ID)
9             ON DELETE CASCADE
10            ON UPDATE CASCADE ,
11
12    FOREIGN KEY(IDProducto)
13        REFERENCES ProductoDulceria (ID)
14            ON DELETE CASCADE
15            ON UPDATE CASCADE ,
16
17    PRIMARY KEY (IDVenta, IdProducto)
18 );
19
20
21 CREATE TABLE TicketProveedor (
22     Cantidad       INT ,
23     Costo          REAL ,
24     Total          REAL ,
25     IDProducto     INT ,
26     IDProveedor    INT ,
27
28     FOREIGN KEY (IDProducto)
29         REFERENCES ProductoDulceria (ID)
30             ON DELETE CASCADE
31             ON UPDATE CASCADE ,
32
33     FOREIGN KEY (IDProveedor)
34         REFERENCES Proveedor (ID)
35             ON DELETE CASCADE
36             ON UPDATE CASCADE ,
37
38     PRIMARY KEY (IDProducto, IDProveedor)
39 );
```

## 2.3. Diccionario de Datos

### Relación: Función

Aquella relación que almacena los datos que caracterizan a la presentación de una película a una hora determinada, en una sala específica.

Una película puede presentarse en varias funciones y varias funciones pueden presentarse en una sala.

Campo	Tipo de Datos (tamaño)	Constraint
Hora	TIME	PK
Dia	DATE	PK
NumeroSala	INT(11)	PK , FK
Precio	REAL	
Tipo	VARCHAR(20)	
IDPelicula	INT(11)	FK

### Relación: Película

Relación que guarda los datos sobre una determinada película, como su descripción, clasificación y duración, es necesario saber qué película se presenta en cada función, recordando que una película puede presentarse en varias funciones, pero una función solo puede presentar una película.

Campo	Tipo de Datos (tamaño)	Constraint
ID	INT (11)	PK
Clasificacion	VARCHAR(4)	
Nombre	VARCHAR(50)	
Duracion	INT(11)	
Descripcion	VARCHAR(400)	

## Relación: TicketBoleto

Relación que permite llevar un control sobre los boletos que tiene cada función, guardando también las características de la venta.

Campo	Tipo de Datos (tamaño)	Constraint
NombreBoletos	INT(11)	
Costos	REAL	
Hora	TIME	PK , FK
Dia	DATE	PK , FK
NumeroSala	INT(11)	PK , FK
IDVenta	INT(11)	PK , FK

## Relación: TicketDulceria

Aquella relación que permite tener un control sobre la comida/dulces vendidos.

Campo	Tipo de Datos (tamaño)	Constraint
Costos	REAL	
Cantidad	INT(11)	
IPProducto	INT(11)	PK , FK
IDVenta	INT(11)	PK , FK

## Relación: ProductoDulceria

Relación encargada de almacenar los productos de la dulcería, tales como dulces, palomitas y refrescos. Se guardan también sus costos y el proveedor que los provee.

Campo	Tipo de Datos (tamaño)	Constraint
IPProducto	INT(11)	PK
Stock	INT(11)	
Nombre	VARCHAR(50)	
Costo	REAL	
IDProveedor	INT(11)	FK

## Relación: TicketProoveedor

Relación destinada a relacionar los proveedores con los productos de dulcería, almacenando la cantidad, el total y la fecha en la que se proporcionaron los dulces, por parte del proveedor.

Campo	Tipo de Datos (tamaño)	Constraint
Fecha	DATE	
Total	REAL	
Cantidad	INT(11)	
IDProducto	INT(11)	PK , FK
IDProveedor	INT(11)	PK , FK

## Relación: Proveedor

Aquella relación destinada a almacenar la información del proveedor como lo es su nombre y correo (el correo será único, y tendrá su contraseña).

Campo	Tipo de Datos (tamaño)	Constraint
IDProveedor	INT(11)	PK
Nombre	VARCHAR(15)	
Apellido1	VARCHAR(15)	
Apellido2	VARCHAR(15)	
Correo	VARCHAR(30)	
Contrasena	VARCHAR(100)	

## Relación: EmpleadoSala

Relación que asocia al empleado y sala. Viene dada de una cardinalidad muchos a muchos, y sirve para almacenar los empleados que revisan una sala, y las salas que son revisadas por un empleado.

Campo	Tipo de Datos (tamaño)	Constraint
ID	INT(11)	PK , FK
NumSala	INT(11)	PK, FK

## Relación: Empleado

Relación que guarda la información del usuario o cliente, almacenando ambos con una relación unitaria para saber que gerente dirige a los usuarios. Los usuarios y gerentes se distinguen entre sí por el campo rol. Un gerente puede dirigir a varios empleados, pero un empleado solo tiene un gerente.

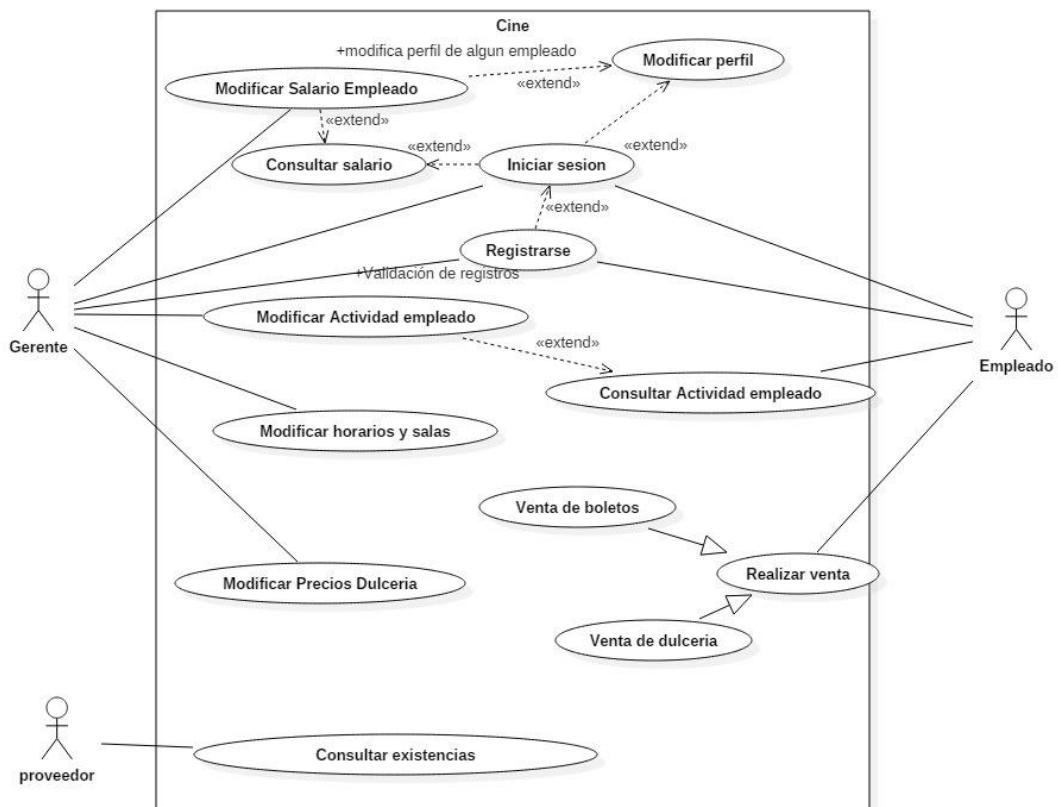
Campo	Tipo de Datos (tamaño)	Constraint
ID	INT(11)	PK
Nombre	VARCHAR(15)	
Apellido1	VARCHAR(15)	
Apellido2	VARCHAR(15)	
Correo	VARCHAR(30)	
Contrasena	VARCHAR(100)	
Sueldo	REAL	
Turno	VARCHAR(10)	
Rol	VARCHAR(15)	
Genero	VARCHAR(10)	
IDGerente	INT(11)	FK
NumEmpleados	INT(11)	
ID_Ger	INT(11)	FK (Llave recursiva)

## Relación: Sala

Aquella relación que almacena las salas y el número de asientos disponible en cada una.

Campo	Tipo de Datos (tamaño)	Constraint
ID	INT(11)	PK
Asientos	INT(11)	

## 2.4. Diagrama de Casos de Uso



## 2.5. Diagrama Secuencial

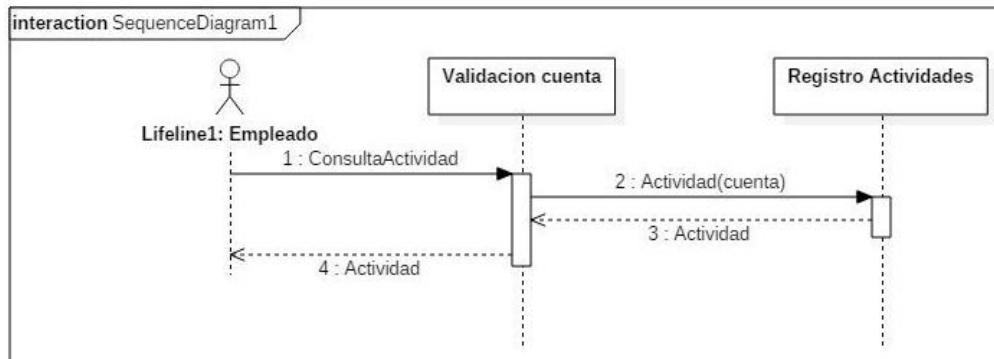


Figura 2.1: Consultar Actividad

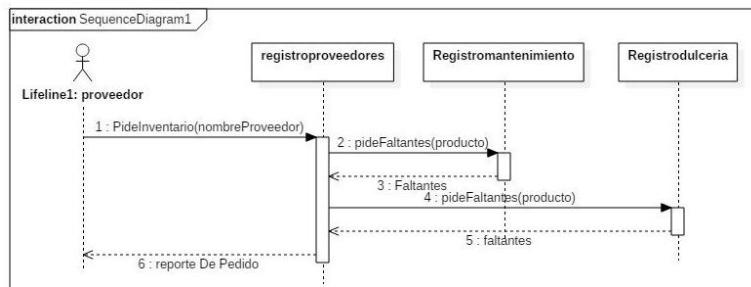


Figura 2.2: Consultar Existencias

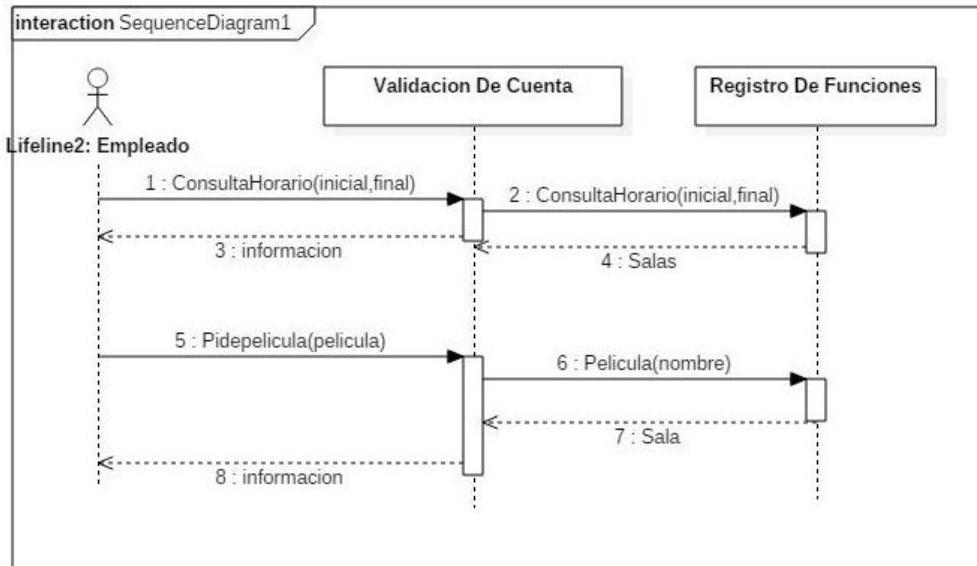


Figura 2.3: Consultar Horarios y Salas

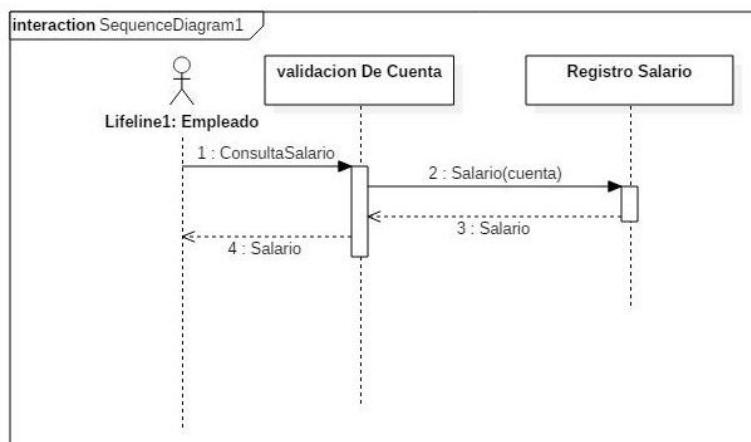


Figura 2.4: Consultar Salarios

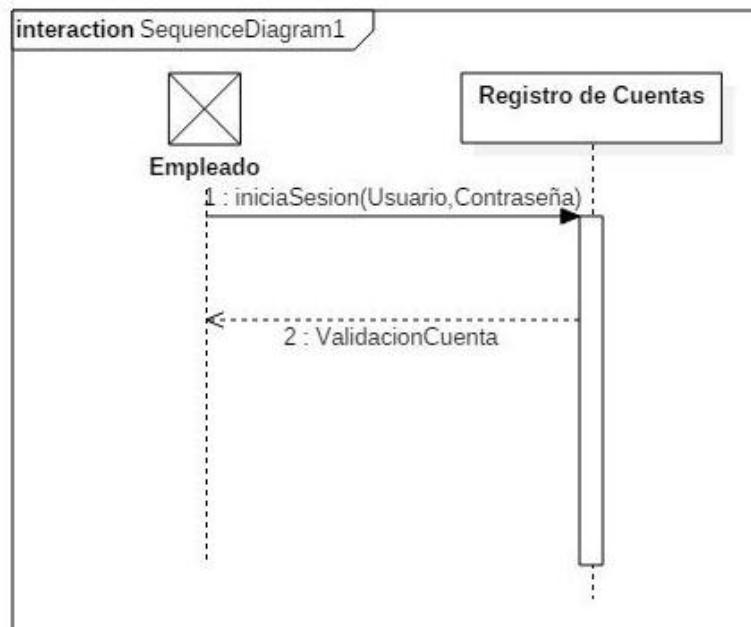


Figura 2.5: Iniciar Sesión

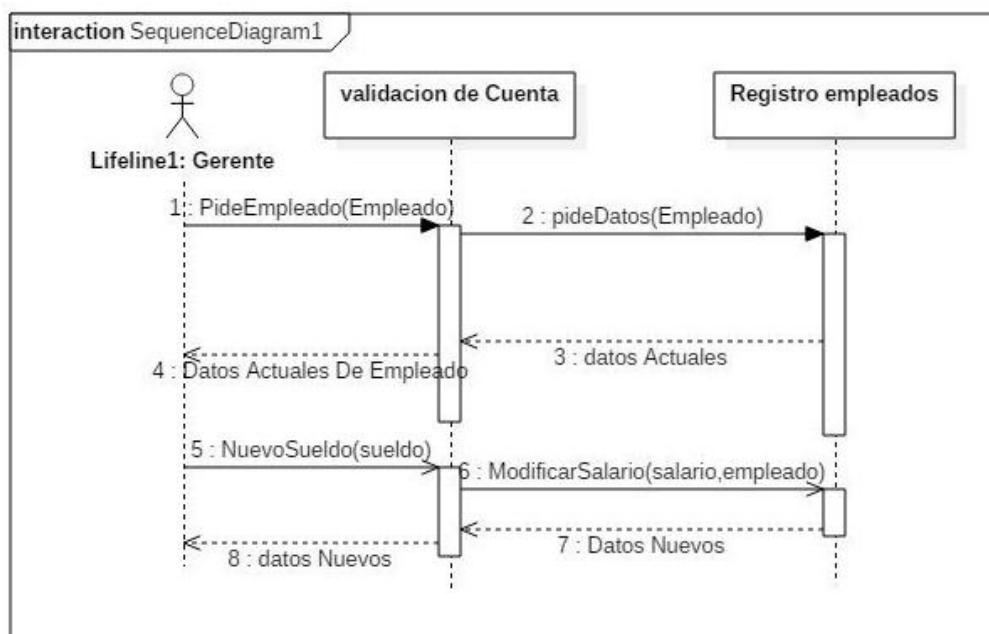


Figura 2.6: Modificar Actividad Empleado

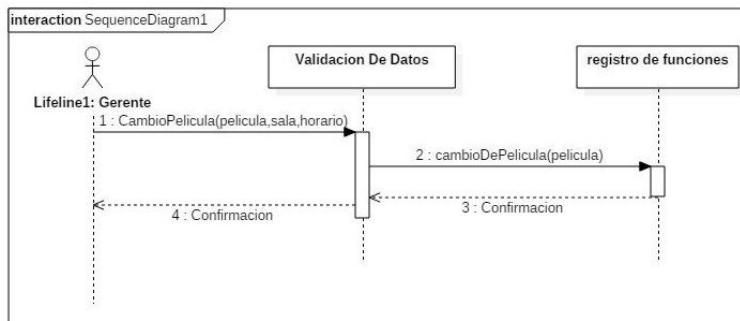


Figura 2.7: Modificar Horarios Salas

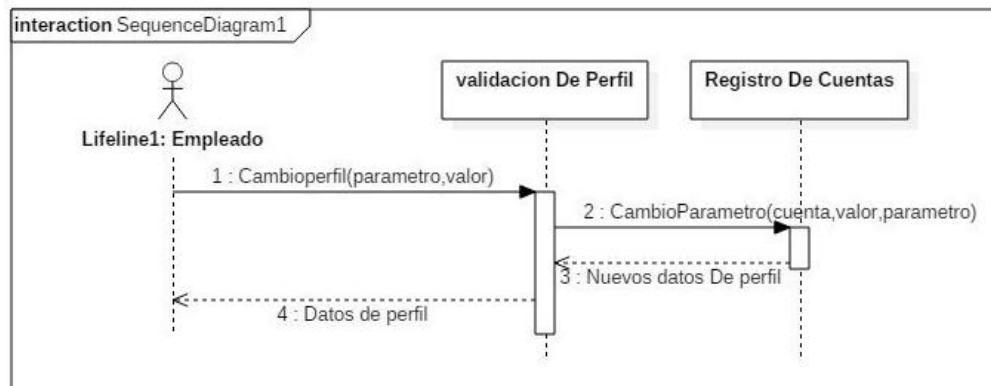


Figura 2.8: Modificar Perfil

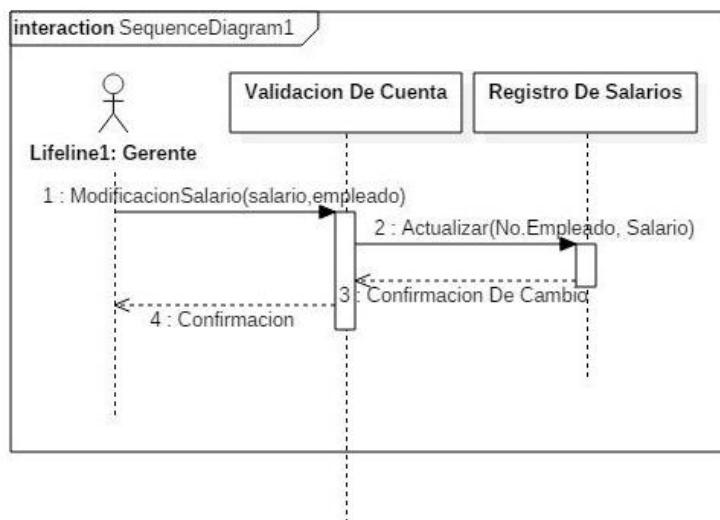


Figura 2.9: Modificar Precios Dulcería

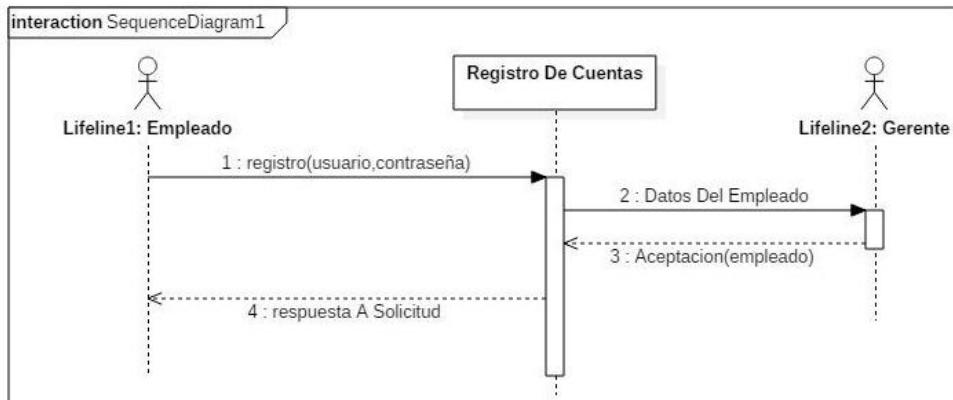


Figura 2.10: Modificar Salario

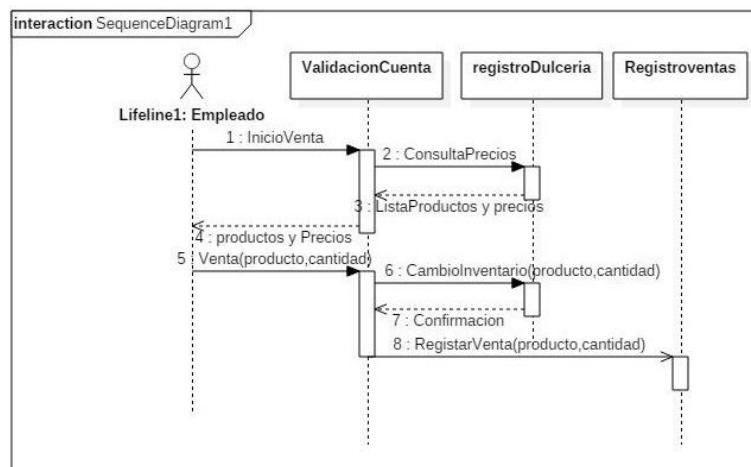


Figura 2.11: Registrarse

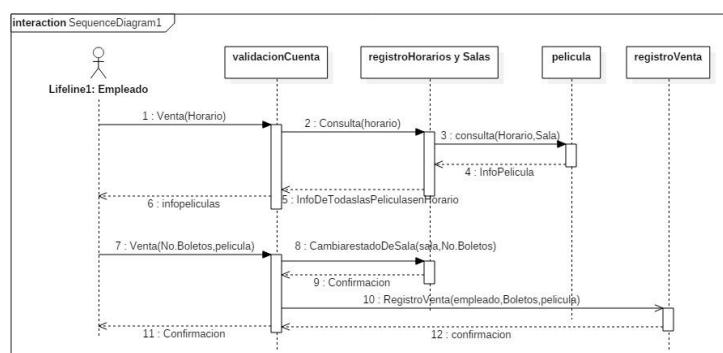


Figura 2.12: Venta Boletos

## 2.6. Avance de la Interfaz

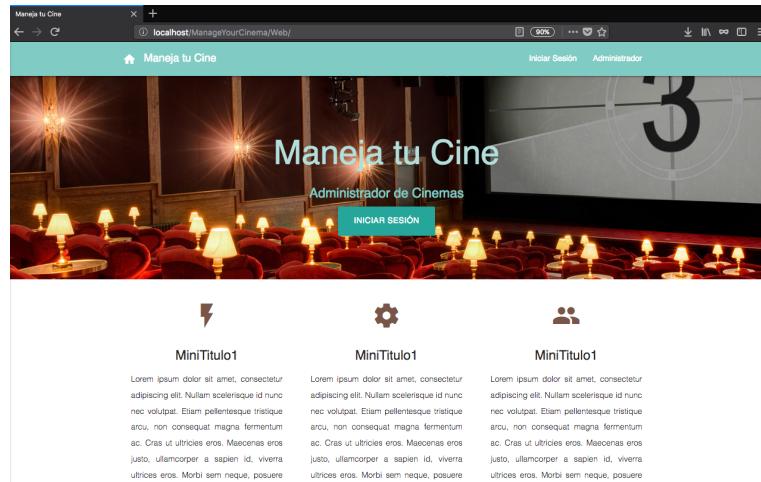


Figura 2.13: Página Principal

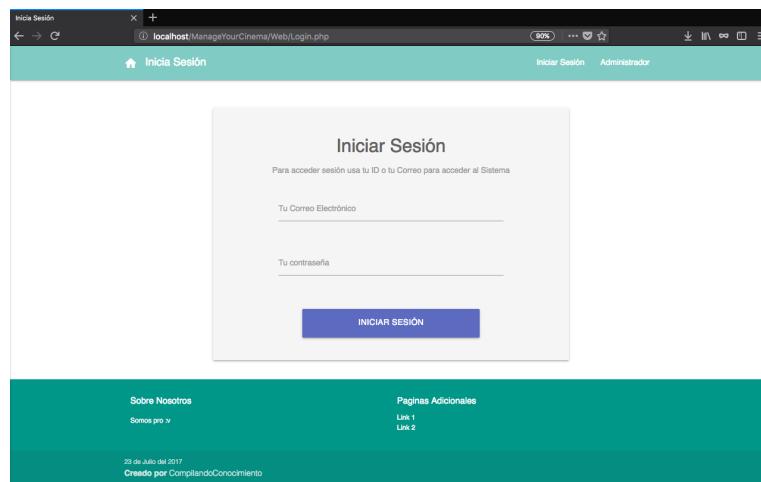


Figura 2.14: Menú de Inicio

ID	Sueldo	Turno	Genero	Nombre	Apellido 1	Apellido 2	Correo	Contraseña	ID del Gerente
1	50	M	M	Oscar	Rosas	Hernandez	SoyOscarRH@gmail.com	hola	
2	150	V	F	Leilan	Marmolejo	Robles	Leilan@gmail.com	adios	

[VE LOS EMPLEADOS](#)

Figura 2.15: Tabla que muestra los Datos

ID	1	2
Sueldo	50	150
Turno	M	V
Genero	M	F
Nombre	Oscar	Leilan
Apellido 1	Rosas	Marmolejo
Apellido 2	Hernandez	Robles
Correo	SoyOscarRH@gmail.com	Leilan@gmail.com
Contraseña	hola	adios
ID del Gerente		

Figura 2.16: Empezamos a hacerlo Responsive para pantallas pequeñas (modificación del menú y creamos menú hamburguesa)

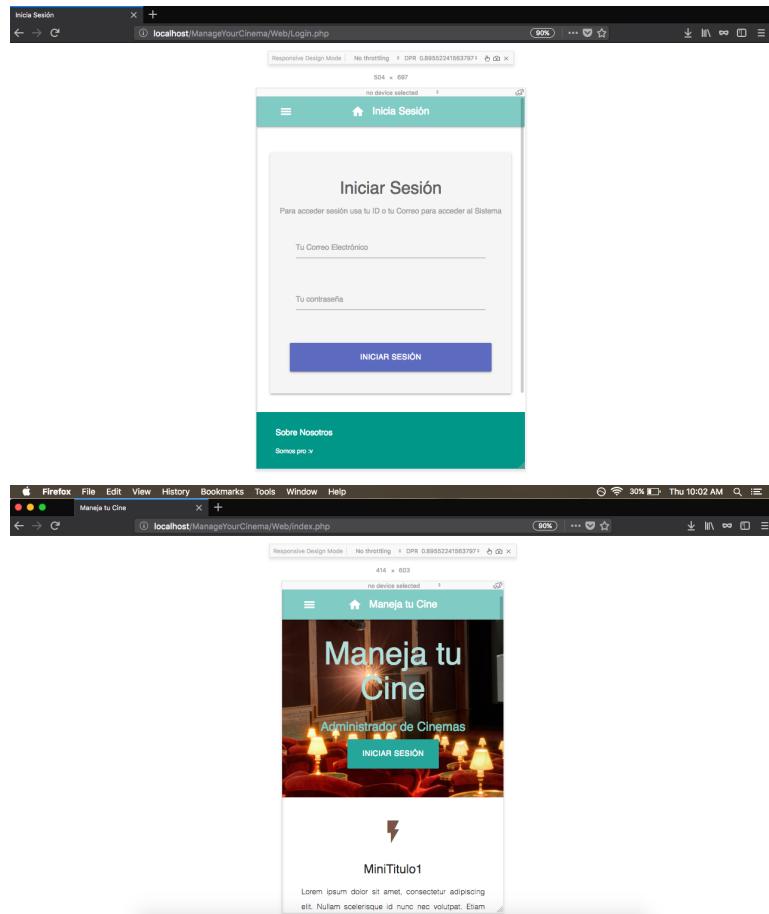


Figura 2.17: Empezamos a hacerlo Responsive para pantallas pequeñas (modificación del menú y creamos menú hamburguesa)

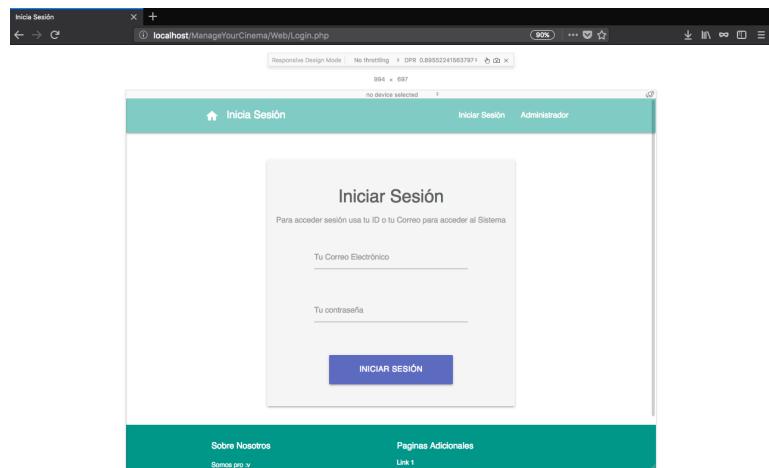


Figura 2.18: Empezamos a hacerlo Responsive para pantallas medianas