PROYECTO INTEGRADOR SOFTWARE II

DIEGO YANZIR GUTIÈRREZ CAMELO

DANIEL FERNANDO GUZMÁN CAICEDO

UNIVERSIDAD DE CUNDINAMARCA

ING. SISTEMAS

2020

PROYECTO INTEGRADOR SOFTWARE II

DIEGO YANZIR GUTIÈRREZ CAMELO

DANIEL FERNANDO GUZMÁN CAICEDO

DOCENTE: JOSE FERNAEDO SOLTELO

UNIVERSIDAD DE CUNDINAMARCA

ING. SISTEMAS

2020

TÍTULO

CINEMA-SOFT

DESCRIPCIÓN

Brindar a un cinema local, una gestión a la reserva de los tiquetes de las salas para el estreno de una película; ya que al momento de estreno de una película se forman filas incomodas para la compra de un tiquete.

El cliente pueda ver toda la información correspondiente de cinema, películas, estrenos, promociones; realizara la reserva de un tiquete de dicha película elegida.

OBJETIVOS

Objetivo General:

* Desarrollar un software web y una base de datos, para la gestión de las reservas de tiquetes de un cinema.

Objetivos Específicos:

* Analizar los atributos o campos que necesita realizar almacenar la información del cinema.
* Diseñar un diagrama de flujo para el proceso de reserva de tiquete.
* Diseñar la base de datos para el sistema.
* Diseñar la interfaz para la visualización de la información y la reserva de tiquetes.
* Implementar la base de datos con la interfaz para su uso.
* Realizar pruebas al sistema para detectar falencias en el sistema.

JUSTIFICACIÓN

Este software se desarrolla para mejorar la gestión de reservar un tiquete de un cinema; ya que actualmente no se puede realizar la gestión de forma virtual sino de manera presencial, en donde los usuarios que no cuentan con tiempo de anterioridad para realizarlo.

Este software ayudara la gestión de reservas de tiquetes dentro de la taquilla, mejorando el flujo de las personas dentro del cinema.

METODOLOGÍA

* MODELO XP

La metodología XP o Programación Extrema es una metodología ágil y flexible utilizada para la gestión de proyectos.

Extreme Programming se centra en potenciar las relaciones interpersonales del equipo de desarrollo como clave del éxito mediante el trabajo en equipo, el aprendizaje continuo y el buen clima de trabajo.

Esta metodología pone el énfasis en la retroalimentación continua entre cliente y el equipo de desarrollo y es idónea para proyectos con requisitos imprecisos y muy cambiantes.

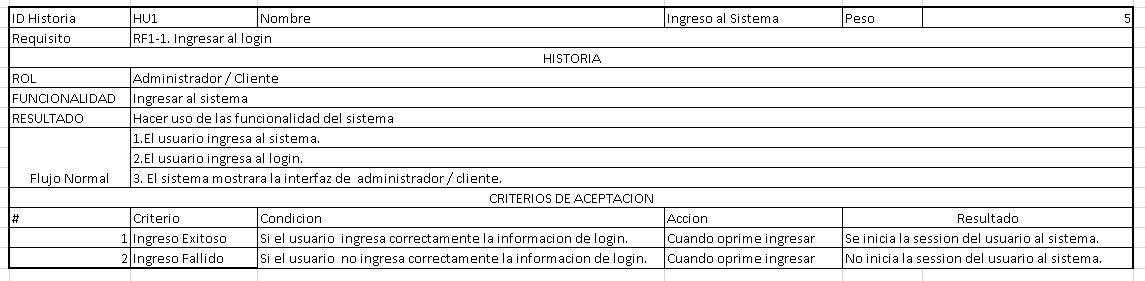
PRODUCT BACKLOG

1 Lista de Back log.

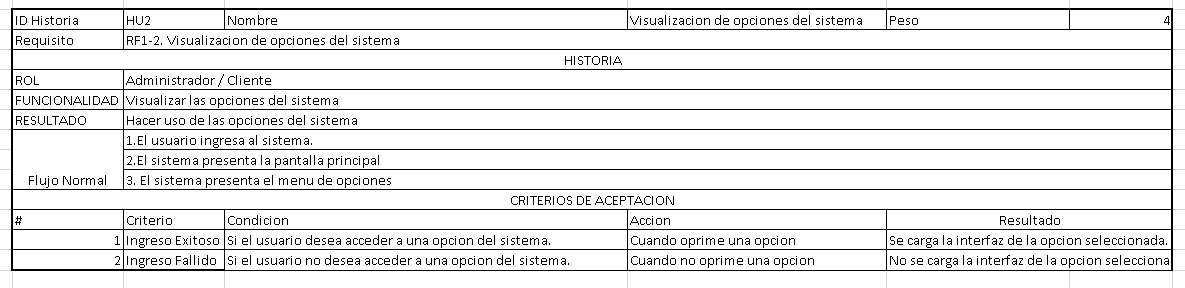
|  |  |  |
| --- | --- | --- |
| Sprint | TAREA | ESTADO |
| 1 | Historia de usuario del software  Modelo entidad relación  Modelo de clases  Modelara la base de datos | Terminada |
| 2 | Crear Interfaz de login  Validar el usuario en la base de datos  Realizar el script de la base de datos | Realizando |
| 3 | Crear Interfaz principal de usuarios  Crear Interfaz de usuarios  Gestionar y validar los usuarios en la base de datos | Realizando |
| 4 | Crear Interfaz de películas  Gestionar y validar las películas en la base de datos  Crear Interfaz de comidas  Gestionar y validar las comidas en la base de datos | Por Realizar |
| 5 | Crear Interfaz de salas  Gestionar y validar las salas en la base de datos  Crear Interfaz de tickets  Gestionar y validar los tickets en la base de datos | Por Realizar |
| 6 | Realizar pruebas y testeo a cada módulo  Realizar mantenimiento al software | Por Realizar |
| 7 | Realizar manual de usuario  Realizar manual técnico  Realizar diccionario de datos | Por Realizar |
|  |  |  |

HISTORIAS DE USUARIO

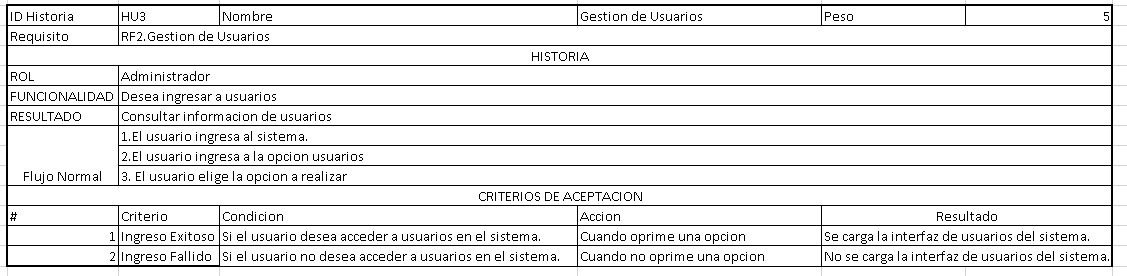
1 Ingresar al login



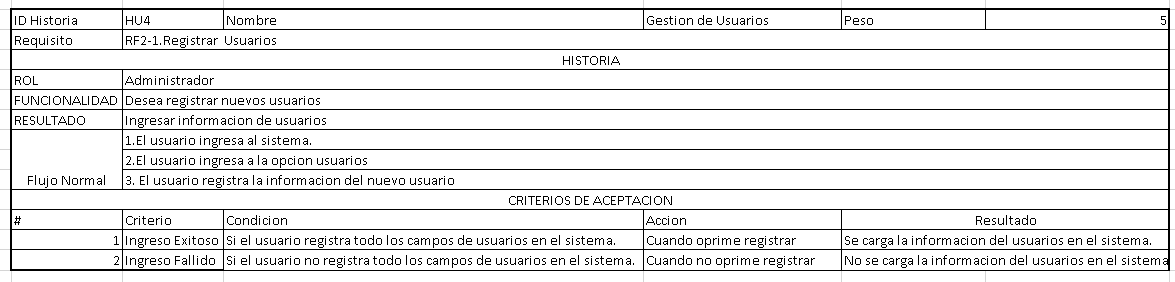
2 Visualización de opciones del sistema



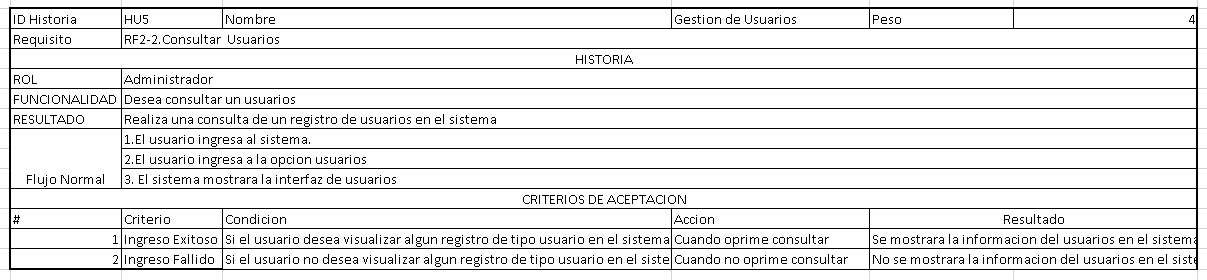
3 Gestión de Usuarios



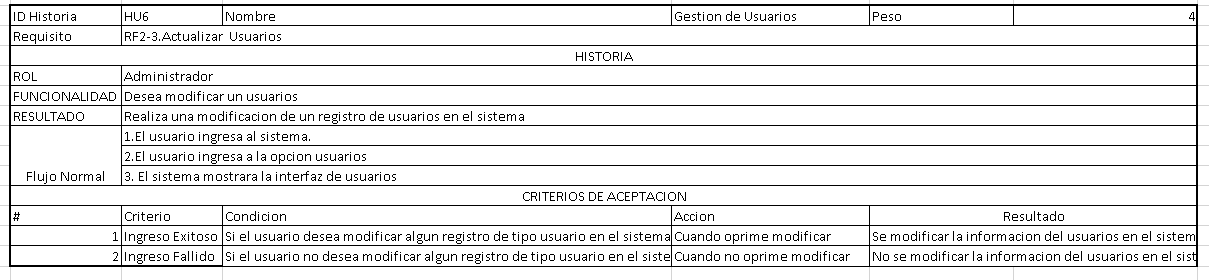
4 Registrar Usuarios



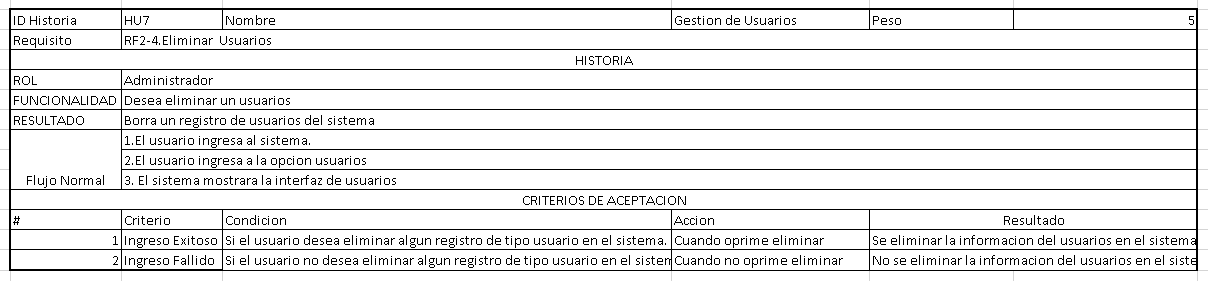
5 Consultar Usuarios



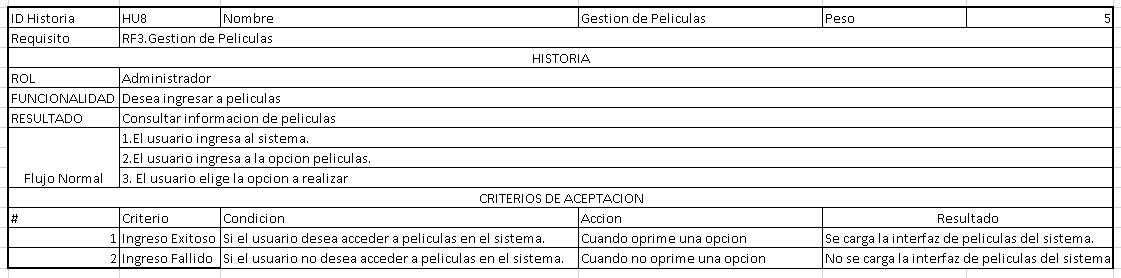
6 Actualizar Usuarios



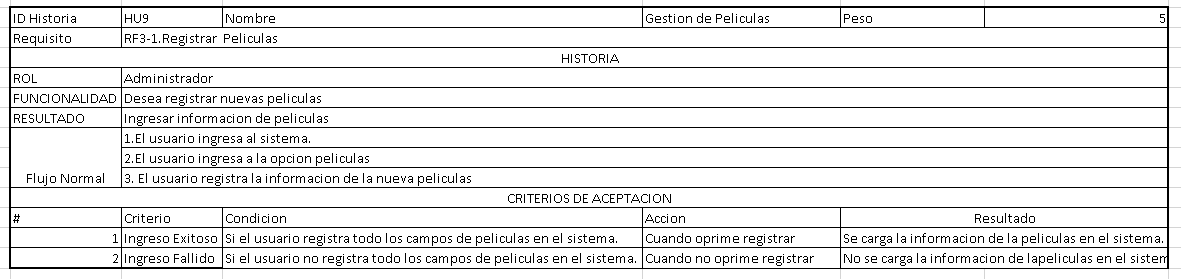
7 Eliminar Usuarios



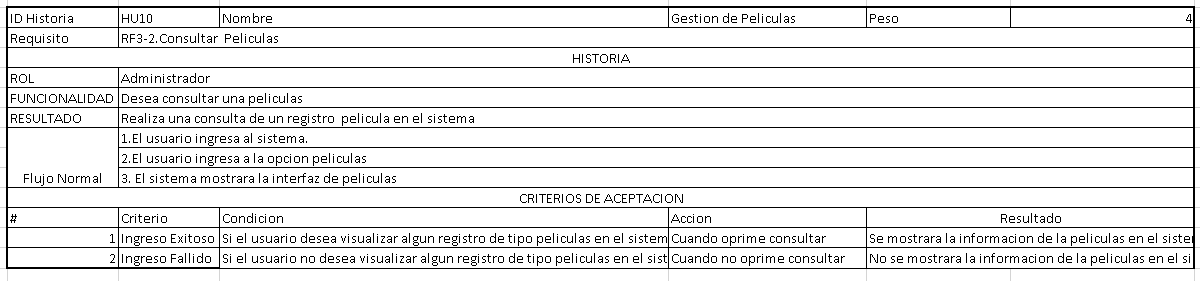
8 Gestión de Películas



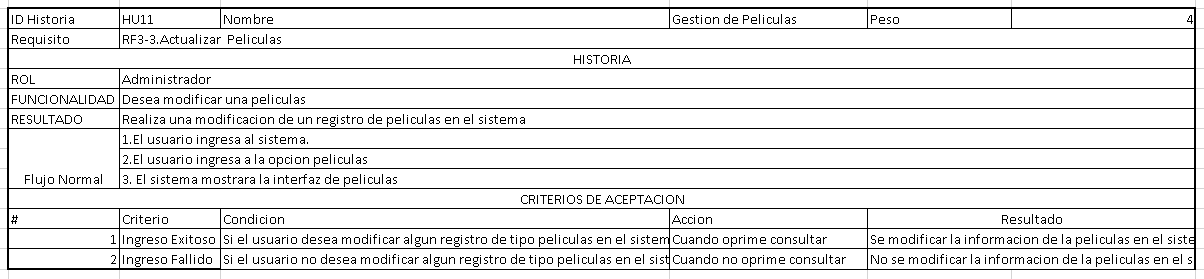
9 Registrar Películas



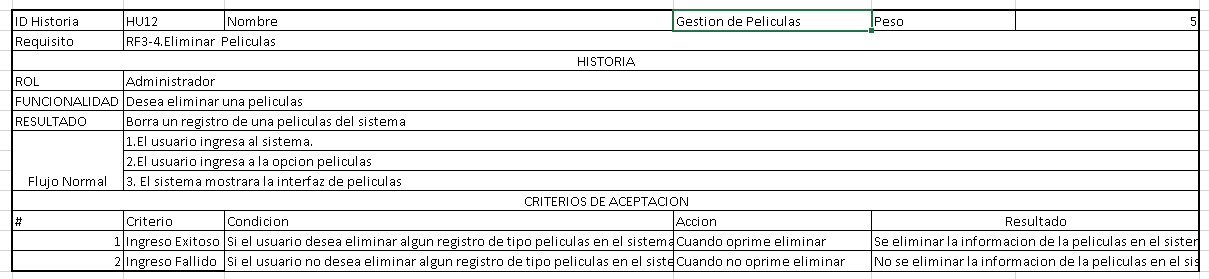
10 Consultar Películas



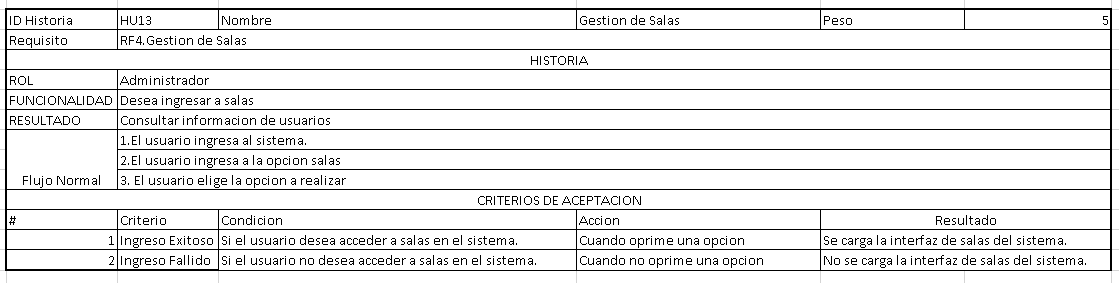
11 Actualizar Películas



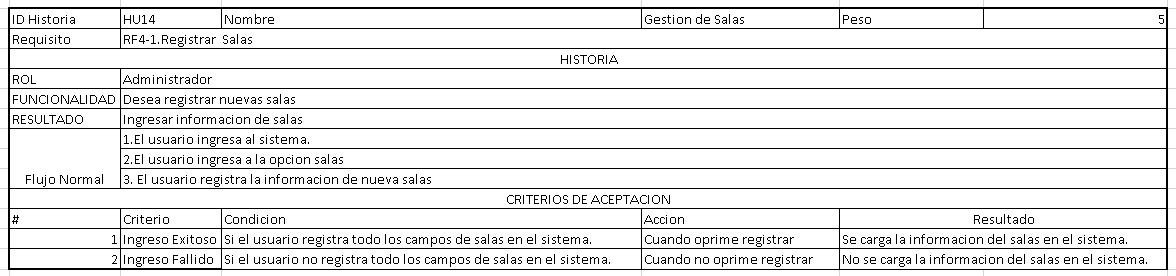
12 Eliminar Películas



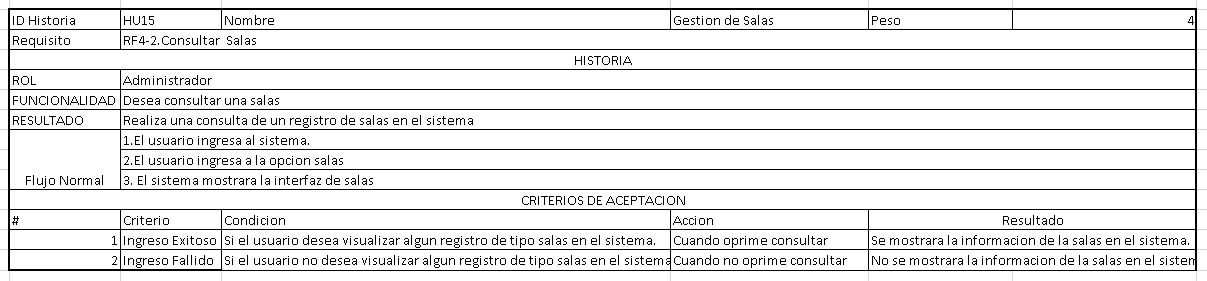
13 Gestión de Salas



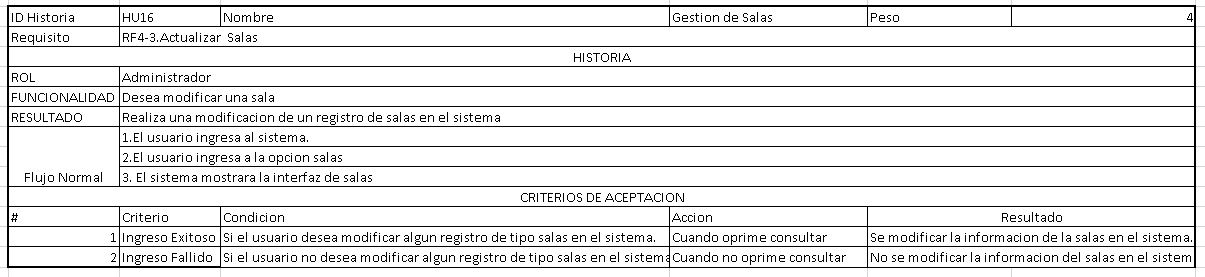
14 Registrar Salas



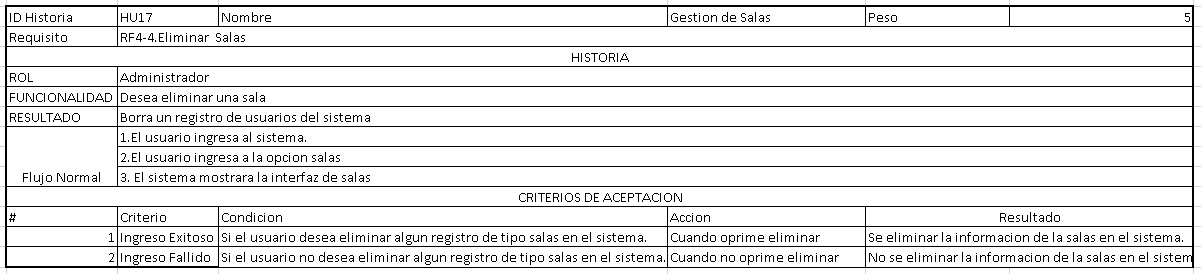
15 Consultar Salas



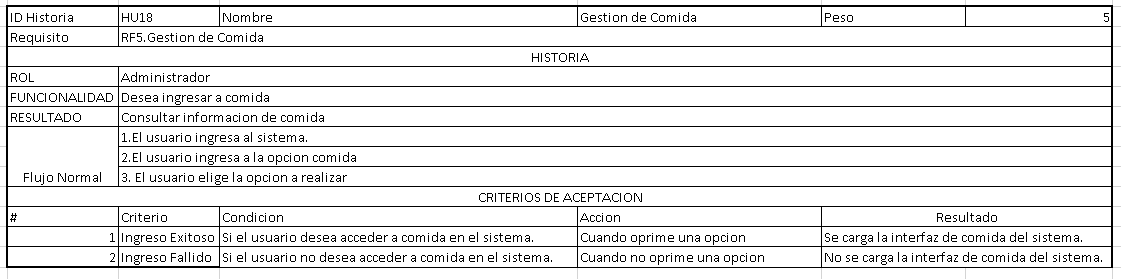
16 Actualizar Salas



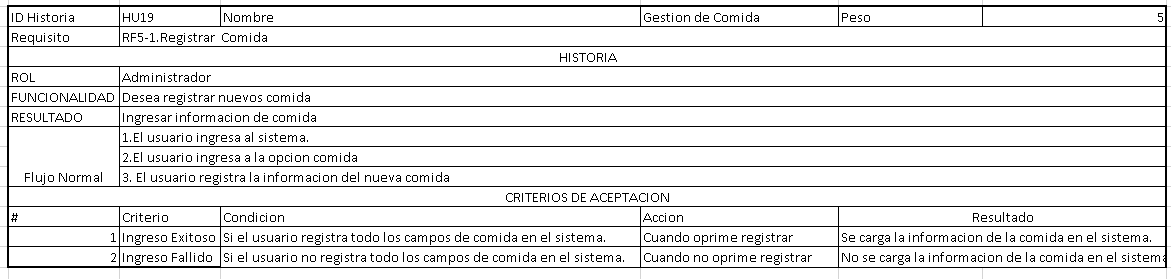
17 Eliminar Salas



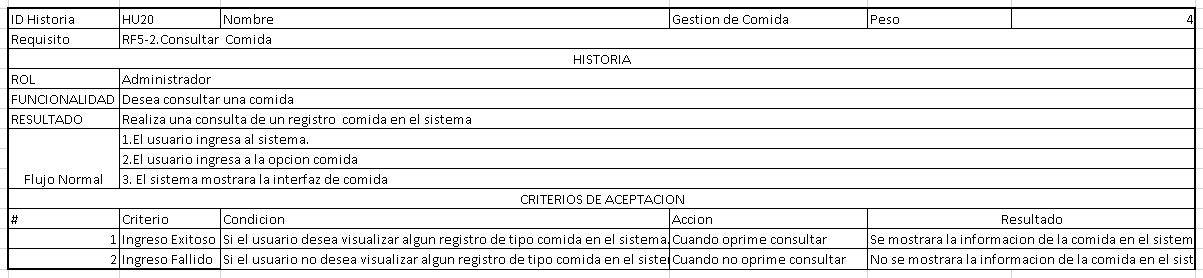
18 Gestión de Comida



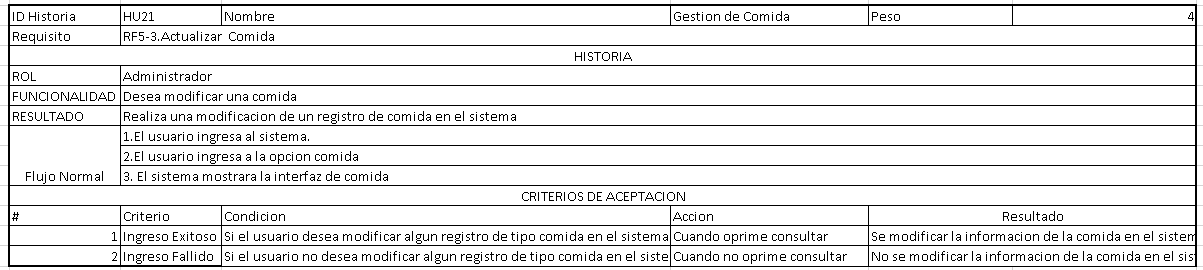
19 Registrar Comida



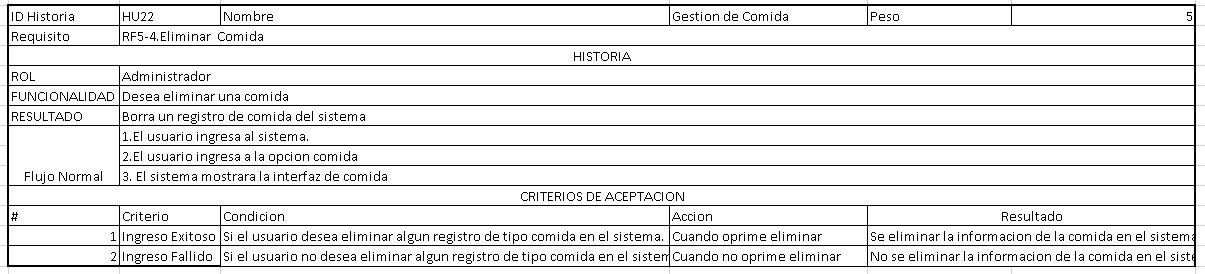
20 Consultar Comida



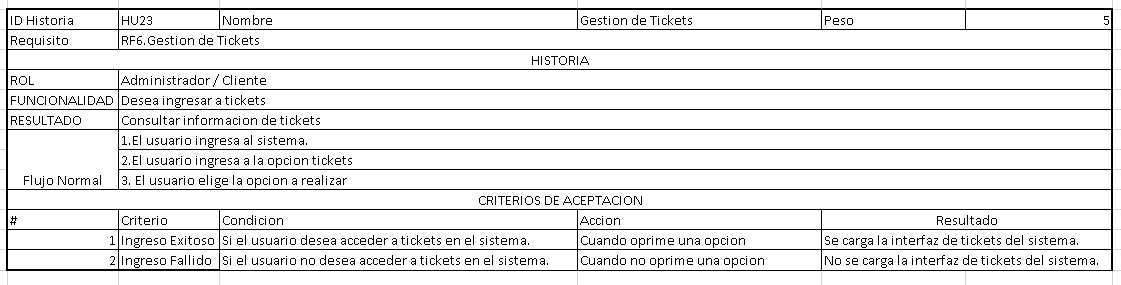
21 Actualizar Comida



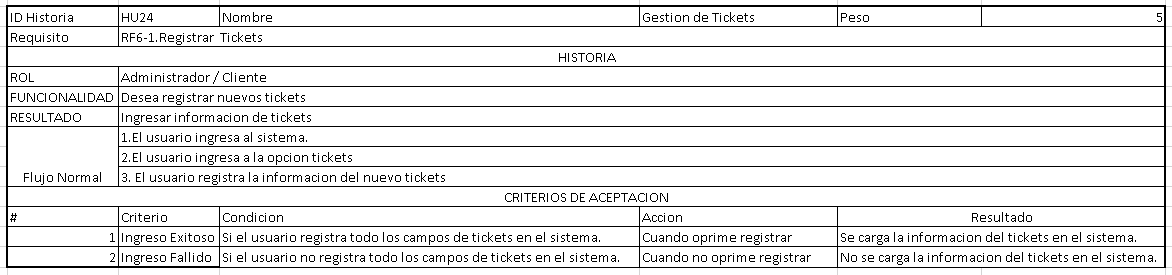
22 Eliminar Comida



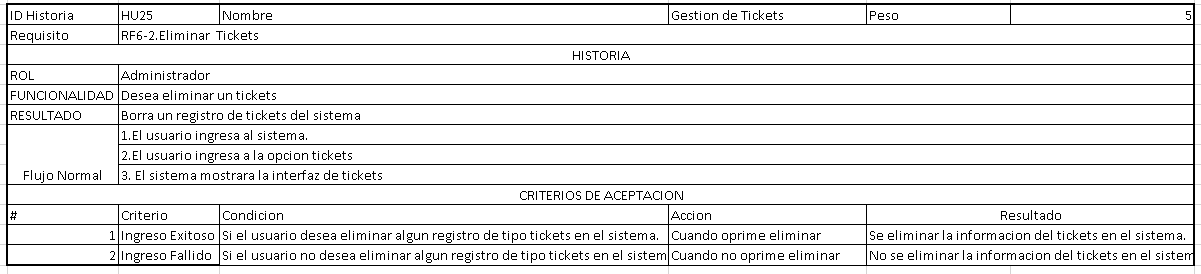
23 Gestión de Tickets



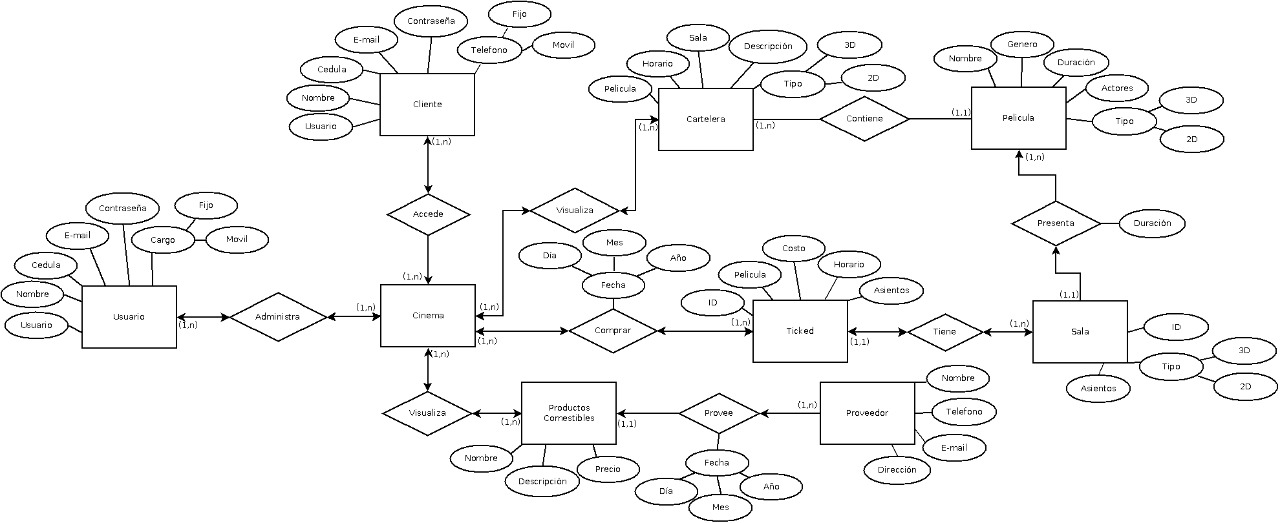
24 Registrar Tickets



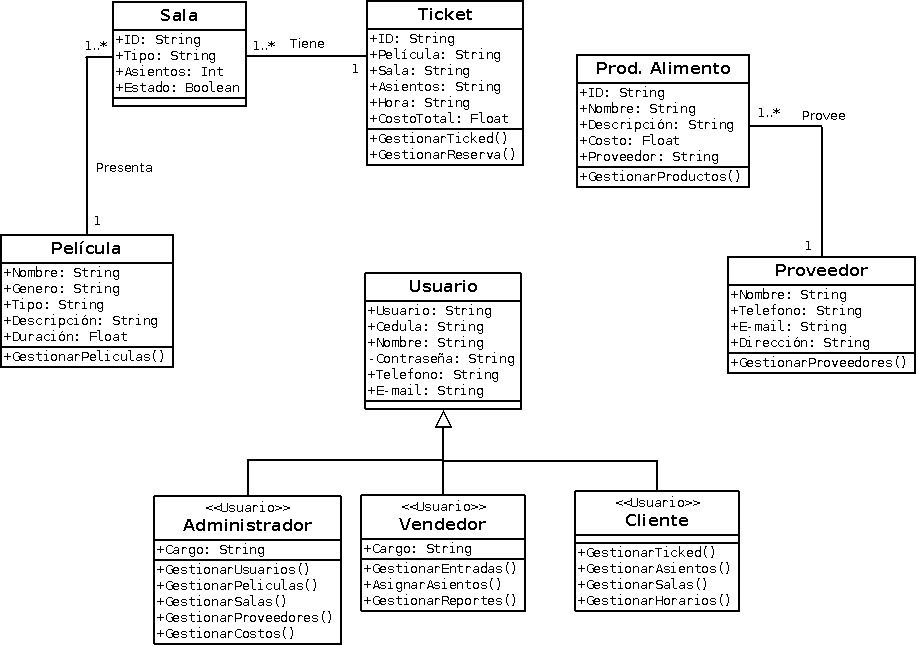
25 Eliminar Tickets



MODELO ENTIDAD RELACION



MODELO DE CLASES



SCRIPT DE LA BASE DE DATOS

CREATE TABLE usuario (  
 idUsuario int(11) UNSIGNED NOT NULL,  
 Dni varchar(10) DEFAULT NULL,  
 Nombres varchar(50) DEFAULT NULL,  
 Direccion varchar(25) DEFAULT NULL,  
 Email varchar(25) DEFAULT NULL,  
 Password varchar(25) DEFAULT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8;

CREATE TABLE reserva (  
 idReserva int(11) UNSIGNED NOT NULL,  
 idUsuario int(11) UNSIGNED NOT NULL,  
 idCartelera int(11) UNSIGNED NOT NULL,  
 idDetalleC int(10) UNSIGNED NOT NULL,  
 FechaReserva varchar(11) DEFAULT NULL,  
 idEstado int(11) UNSIGNED NOT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8;

CREATE TABLE detalle\_reserva (  
 idDetalleP int(10) UNSIGNED NOT NULL,  
 idReserva int(11) UNSIGNED NOT NULL,  
 Cantidad int(11) UNSIGNED DEFAULT NULL,  
 Descripcion varchar(10) DEFAULT NULL,  
 Precio double DEFAULT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8;

CREATE TABLE pelicula (  
 idPelicula int(11) UNSIGNED NOT NULL,  
 Nombre varchar(255) DEFAULT NULL,  
 Foto longblob,  
 Descripcion varchar(255) DEFAULT NULL,  
 Formato varchar(25) DEFAULT NULL,  
 Genero varchar(25) DEFAULT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8;

CREATE TABLE comestible (  
 idComestible int(11) UNSIGNED NOT NULL,  
 idProveedor int(11) UNSIGNED NOT NULL,   
 Nombre varchar(255) DEFAULT NULL,  
 Foto longblob,  
 Descripcion varchar(255) DEFAULT NULL,  
 Precio double DEFAULT NULL,  
 Stock int(11) UNSIGNED DEFAULT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8;

CREATE TABLE detalle\_comida (  
 idDetalleC int(10) UNSIGNED NOT NULL,  
 idComestible int(11) UNSIGNED NOT NULL,  
 idReserva int(11) UNSIGNED NOT NULL,  
 Cantidad int(11) UNSIGNED DEFAULT NULL,  
 PrecioComida double DEFAULT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8;

CREATE TABLE proveedor (  
 idProveedor int(11) UNSIGNED NOT NULL,  
 Nombres varchar(255) DEFAULT NULL,  
 Direccion varchar(255) DEFAULT NULL,  
 Email varchar(255) DEFAULT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8;

CREATE TABLE cartelera (  
 idCartelera int(11) UNSIGNED NOT NULL,  
 idSala int(11) UNSIGNED NOT NULL,  
 idPelicula int(11) UNSIGNED NOT NULL,  
 horaInicio int(11) UNSIGNED NOT NULL,  
 horaFin int(11) UNSIGNED NOT NULL,  
 idEstado int(11) UNSIGNED NOT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8;

CREATE TABLE sala (  
 idSala int(11) UNSIGNED NOT NULL,  
 Nombre varchar(9) DEFAULT NULL,  
 numFila int(11) UNSIGNED NOT NULL,  
 numColumna int(11) UNSIGNED NOT NULL,  
 Foto longblob  
) ENGINE=InnoDB DEFAULT CHARSET=utf8;

CREATE TABLE asiento (  
 idAsiento int(11) UNSIGNED NOT NULL,  
 idSala int(11) UNSIGNED NOT NULL,  
 Fila int(9) UNSIGNED NOT NULL,  
 Columna varchar(9) DEFAULT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8;

CREATE TABLE reserva\_asientos (  
 idUsuario int(11) UNSIGNED NOT NULL,  
 idAsiento int(11) UNSIGNED NOT NULL,  
 idEstado int(11) UNSIGNED NOT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8;

CREATE TABLE estado (  
 idEstado int(11) UNSIGNED NOT NULL,  
 Descripcion varchar(10) DEFAULT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8;

ALTER TABLE usuario ADD PRIMARY KEY (idUsuario);  
ALTER TABLE reserva ADD PRIMARY KEY (idReserva);  
ALTER TABLE detalle\_reserva ADD PRIMARY KEY (idDetalleP);  
ALTER TABLE pelicula ADD PRIMARY KEY (idPelicula);  
ALTER TABLE comestible ADD PRIMARY KEY (idComestible);  
ALTER TABLE detalle\_comida ADD PRIMARY KEY (idDetalleC,idComestible,idReserva);  
ALTER TABLE proveedor ADD PRIMARY KEY (idProveedor);  
ALTER TABLE cartelera ADD PRIMARY KEY (idCartelera);  
ALTER TABLE sala ADD PRIMARY KEY (idSala);  
ALTER TABLE asiento ADD PRIMARY KEY (idAsiento);  
ALTER TABLE reserva\_asientos ADD PRIMARY KEY (idUsuario);  
ALTER TABLE estado ADD PRIMARY KEY (idEstado);

ALTER TABLE reserva ADD FOREIGN KEY (idUsuario) REFERENCES usuario(idUsuario);  
ALTER TABLE reserva ADD FOREIGN KEY (idCartelera) REFERENCES cartelera(idCartelera);  
ALTER TABLE reserva ADD FOREIGN KEY (idDetalleC) REFERENCES comestible(idDetalleC);  
ALTER TABLE reserva ADD FOREIGN KEY (idEstado) REFERENCES estado(idEstado);  
ALTER TABLE comestible ADD FOREIGN KEY (idProveedor) REFERENCES proveedor(idProveedor);  
ALTER TABLE detalle\_comida ADD FOREIGN KEY (idComestible) REFERENCES comestible(idComestible);  
ALTER TABLE detalle\_comida ADD FOREIGN KEY (idReserva) REFERENCES reserva(idReserva);  
ALTER TABLE detalle\_reserva ADD FOREIGN KEY (idReserva) REFERENCES reserva(idReserva);  
ALTER TABLE cartelera ADD FOREIGN KEY (idSala) REFERENCES sala(idSala);  
ALTER TABLE cartelera ADD FOREIGN KEY (idPelicula) REFERENCES pelicula(idPelicula);  
ALTER TABLE cartelera ADD FOREIGN KEY (idEstado) REFERENCES estado(idEstado);  
ALTER TABLE asiento ADD FOREIGN KEY (idSala) REFERENCES sala(idSala);  
ALTER TABLE reserva\_asientos ADD FOREIGN KEY (idAsiento) REFERENCES asiento(idAsiento);  
ALTER TABLE reserva\_asientos ADD FOREIGN KEY (idEstado) REFERENCES estado(idEstado);