Universidad Autónoma Gabriel René Moreno

Facultad de Ingeniería en Ciencias de la Computación y Telecomunicaciones

**SOFTWARE PARA LA ADMINISTRACION DE UN VIDEO CLUD**



**NOMBRE:** ORELLANA VASQUEZ OSWALDO

**MATERIA:** ARQUITECTURA DE SOFTWARE (INF552-SA)

**DOCENTE:** ING. VEIZAGA GONZALES JOSUÉ OBED

**TABLA DE CONTENIDO**

[1.- Flujo de trabajo: Requerimientos 3](#_Toc86040685)

[**1.1.- Requisitos funcionales** 3](#_Toc86040686)

[1.1.1.- Lista casos de uso 3](#_Toc86040687)

[1.1.2.- Lista de actores 3](#_Toc86040688)

[1.1.3.- Detallar casos de usos 4](#_Toc86040689)

[2.- Flujo de trabajo: Análisis 8](#_Toc86040690)

[**2.1.- Análisis de arquitectura** 8](#_Toc86040691)

[2.1.1.- Diagrama de comunicación 8](#_Toc86040692)

[3.- Flujo de trabajo: Diseño 9](#_Toc86040693)

[**3.1.- Diseño de la arquitectura** 9](#_Toc86040694)

[**3.2.- Diseño de la base de datos** 9](#_Toc86040695)

[3.2.1.- Diseño conceptual 9](#_Toc86040696)

[3.2.2.- Diseño físico (Script) 10](#_Toc86040697)

[**3.3.- Diseño de detalle procedimental** 13](#_Toc86040698)

# **1.- Flujo de trabajo: Requerimientos**

## **1.1.- Requisitos funcionales**

### 1.1.1.- Lista casos de uso

* CU1: Gestionar genero
* CU2: Gestionar película
* CU3: Gestionar Nota de Alquiler

### 1.1.2.- Lista de actores



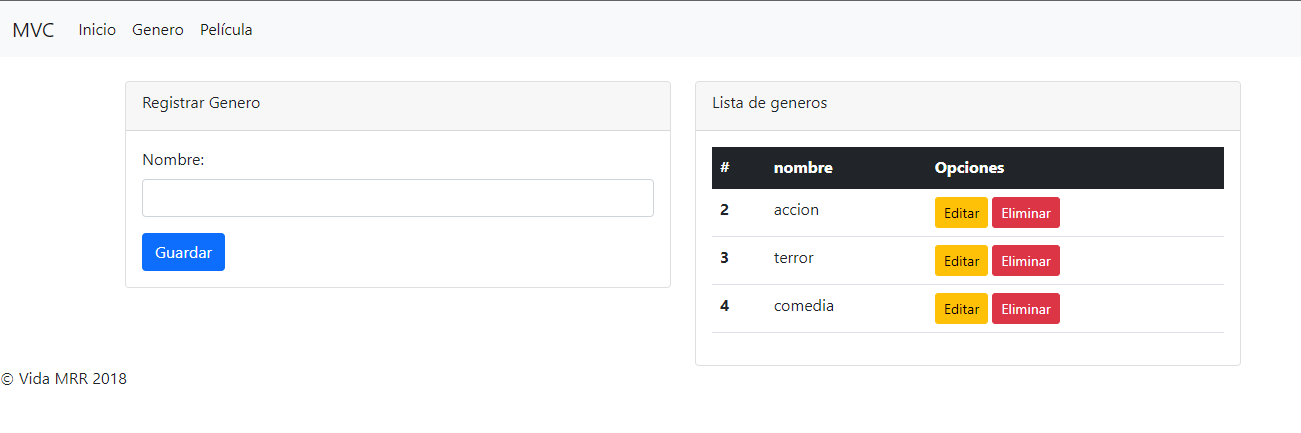
* Administrador: Persona encargada de mantener el software y de realizar el registro y baja de nuevos bibliotecarios.

### 1.1.3.- Detallar casos de usos

CU1: Gestionar genero



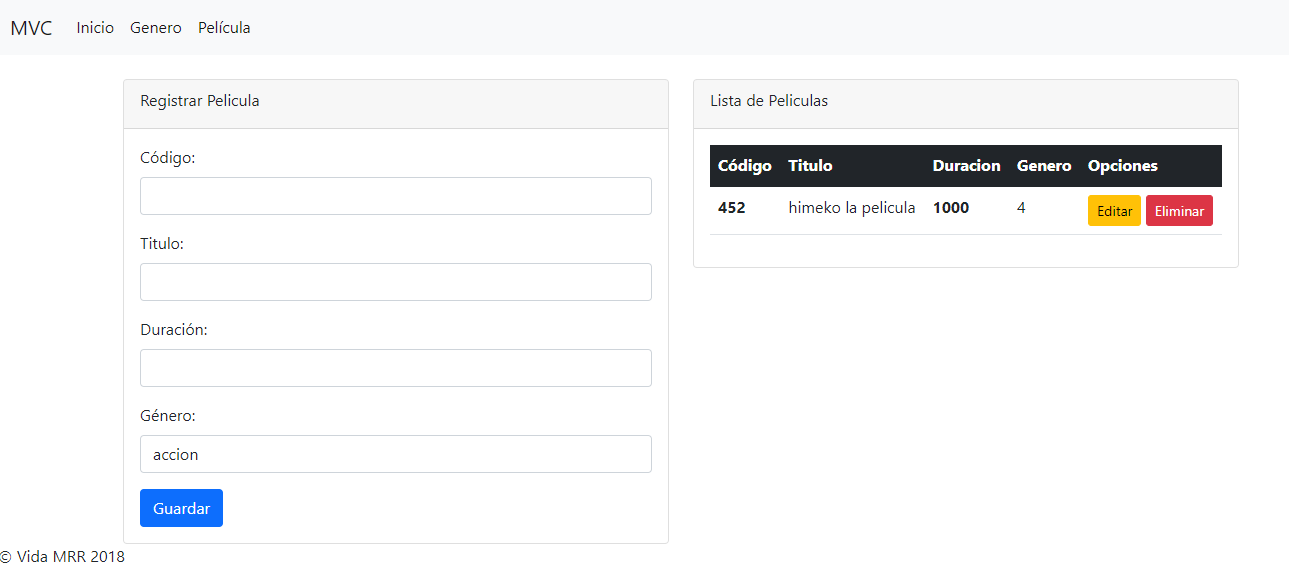
|  |  |
| --- | --- |
| Caso de Uso | CU. Gestionar genero |
| Propósito | Gestión de géneros |
| Descripción | Permite la gestión de los diferentes géneros que posee las películas adquiridas por la tienda |
| Actores | Administrador |
| Actor Iniciador | Administrador |
| Precondición | - |
| Proceso | 1. Registrar genero    1. Agregar datos    2. Insertar datos    3. Guardar datos 2. Modificar genero    1. Modificar datos    2. Guardar datos 3. Eliminar genero    1. Confirmar acción |
| Postcondición | -------- |
| Excepciones | 3.- La eliminación de géneros requiere una reconfirmación del bibliotecario para proceder la acción |



CU2: Gestionar película



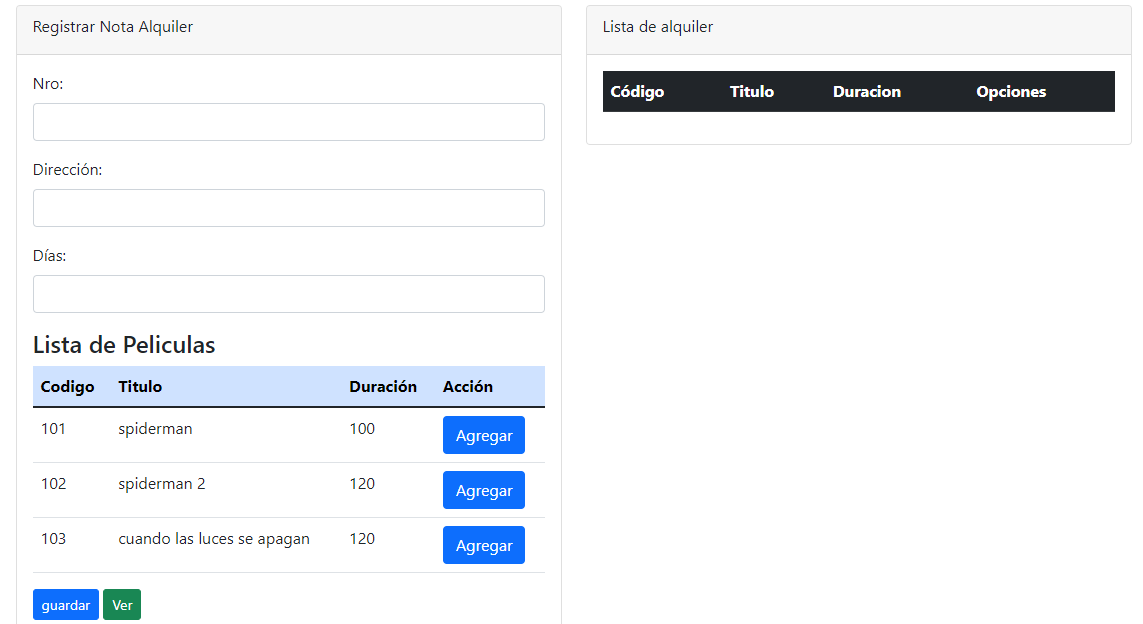
|  |  |
| --- | --- |
| Caso de Uso | CU. Gestionar película |
| Propósito | Gestión de película |
| Descripción | Permite la gestión de las diferentes películas que posee la tienda video clud. |
| Actores | Administrador |
| Actor Iniciador | Administrador |
| Precondición | - |
| Proceso | 1. Registrar película    1. Agregar datos    2. Insertar datos    3. Guardar datos 2. Modificar película    1. Modificar datos    2. Guardar datos 3. Eliminar película    1. Confirmar acción |
| Postcondición | -------- |
| Excepciones |  |



CU2: Gestionar nota alquiler



|  |  |
| --- | --- |
| Caso de Uso | CU. Gestionar Nota alquila |
| Propósito | Gestión de nota de alquiler |
| Descripción | Permite la gestión de las diferentes notas de alquiler que posee la tienda video clud. |
| Actores | Administrador |
| Actor Iniciador | Administrador |
| Precondición | - |
| Proceso | 1. Registrar nota de alquiler    1. Agregar datos    2. Insertar datos    3. Guardar datos 2. Ver nota de alquiler    1. Insertar un numero    2. Buscar 3. Eliminar nota de alquiler    1. Seleccionar nota eliminar    2. Eliminar |
| Postcondición | -------- |
| Excepciones |  |



# **2.- Flujo de trabajo: Análisis**

## **2.1.- Análisis de arquitectura**

### 2.1.1.- Diagrama de comunicación

CU1: Gestionar genero



CU2: Gestionar película



CU3: Gestionar Nota Alquila



# **3.- Flujo de trabajo: Diseño**

## **3.1.- Diseño de la arquitectura**



## **3.2.- Diseño de la base de datos**

### 3.2.1.- Diseño conceptual

### 

### 

### 3.2.2.- Diseño físico (Script)

SET FOREIGN\_KEY\_CHECKS=0

;

/\* Drop Tables \*/

DROP TABLE IF EXISTS `detallesalquila` CASCADE

;

DROP TABLE IF EXISTS `Generos` CASCADE

;

DROP TABLE IF EXISTS `notasalquila` CASCADE

;

DROP TABLE IF EXISTS `Peliculas` CASCADE

;

/\* Create Tables \*/

CREATE TABLE `detallesalquila`

(

    `id` INT NOT NULL AUTO\_INCREMENT,

    `notaalquila\_nro` INT NOT NULL,

    `pelicula\_cod` INT NOT NULL,

    CONSTRAINT `PK\_detallesalquila` PRIMARY KEY (`id` ASC)

)

;

CREATE TABLE `Generos`

(

    `id` INT NOT NULL AUTO\_INCREMENT,

    `nombre` VARCHAR(50) NULL,

    CONSTRAINT `PK\_Generos` PRIMARY KEY (`id` ASC)

)

;

CREATE TABLE `notasalquila`

(

    `nro` INT NOT NULL,

    `fecha` DATE NOT NULL,

    `direccion` VARCHAR(50) NOT NULL,

    `dias` VARCHAR(50) NOT NULL,

    CONSTRAINT `PK\_notasalquila` PRIMARY KEY (`nro` ASC)

)

;

CREATE TABLE `Peliculas`

(

    `codigo` INT NOT NULL,

    `titulo` VARCHAR(50) NULL,

    `duracion` VARCHAR(50) NULL,

    `genero\_id` INT NULL,

    CONSTRAINT `PK\_Peliculas` PRIMARY KEY (`codigo` ASC)

)

;

/\* Create Primary Keys, Indexes, Uniques, Checks \*/

ALTER TABLE `detallesalquila`

 ADD INDEX `IXFK\_detallesalquila\_notasalquila` (`notaalquila\_nro` ASC)

;

ALTER TABLE `detallesalquila`

 ADD INDEX `IXFK\_detallesalquila\_Peliculas` (`pelicula\_cod` ASC)

;

ALTER TABLE `Peliculas`

 ADD INDEX `IXFK\_Peliculas\_Generos` (`genero\_id` ASC)

;

/\* Create Foreign Key Constraints \*/

ALTER TABLE `detallesalquila`

 ADD CONSTRAINT `FK\_detallesalquila\_notasalquila`

    FOREIGN KEY (`notaalquila\_nro`) REFERENCES `notasalquila` (`nro`) ON DELETE Cascade ON UPDATE Cascade

;

ALTER TABLE `detallesalquila`

 ADD CONSTRAINT `FK\_detallesalquila\_Peliculas`

    FOREIGN KEY (`pelicula\_cod`) REFERENCES `Peliculas` (`codigo`) ON DELETE Cascade ON UPDATE Cascade

;

ALTER TABLE `Peliculas`

 ADD CONSTRAINT `FK\_Peliculas\_Generos`

    FOREIGN KEY (`genero\_id`) REFERENCES `Generos` (`id`) ON DELETE Cascade ON UPDATE Cascade

;

SET FOREIGN\_KEY\_CHECKS=1

;

Mapeo

|  |  |
| --- | --- |
| Tabla generos | |
| PK |  |
| ID | NOMBRE |

|  |  |  |  |
| --- | --- | --- | --- |
| Tabla películas | | | |
| PK |  |  | FK |
| ID | TITULO | DURACION | GENERO\_ID |

|  |  |  |
| --- | --- | --- |
| Tabla detallesAlquila | | |
| PK | FK | FK |
| ID | PELICULA\_COD | NOTAALQUILA\_NRO |

|  |  |  |  |
| --- | --- | --- | --- |
| Tabla notasAlquila | | | |
| PK |  |  |  |
| NRO | FECHA | DIRECCION | DIAS |

## **3.3.- Diseño de detalle procedimental**

**CU1: Gestionar genero**

Diagrama de clase dinámica



Diagrama de secuencia



CU2: Gestionar película

Diagrama de clase dinámico 



CU3: Gestionar Nota de alquiler



Diagrama de secuencia

