

**Universidad Nacional Autónoma de México**

**Programación Orientada a Objetos**

**Diseño del software**

**Versión 1.0**

**Colón Palacios Emmanuel**

**Romero Pedraza Francesca Melania**

**Roldán Sánchez Alexis**

**Sandoval Peña Dafne**

**Fecha: 07/Noviembre/2020**

***Contenido***

Histórico de Cambios 3

Descripción de la Arquitectura 4

Paquetes de la arquitectura 4

Definición del ambiente de implementación 5

Vista de despliegue 5

Descripción 6

Modelo de despliegue 6

Vista lógica 6

Descripción 6

Identificación de las clases 6

Vista de datos 8

Descripción 8

Diagrama de la base de datos 8

Vista dinámica 9

Descripción 9

Diagramas de secuencia 9

Diagramas de navegación 221

# Histórico de Cambios

|  |  |  |  |
| --- | --- | --- | --- |
| **Versión** | **Descripción** | **Responsable de Actualización** | **Fecha de actualización** |
| v0.1 | Creación del documento  (sección  1,3.2.1,3.2.2,3.2.3) | Roldán Sánchez Alexis Sandoval Peña Dafne | 02/12/2020 |
| v0.2 | Actualización del documento (secciones  2.2, 3.1, 5.2, 5.3) | Colón Palacios  Emmanuel  Romero Pedraza  Francesca Melania  Roldán Sánchez Alexis  Sandoval Peña Dafne | 06/12/2020 |
|  |  |  |  |
|  |  |  |  |

# 1. Descripción de la Arquitectura

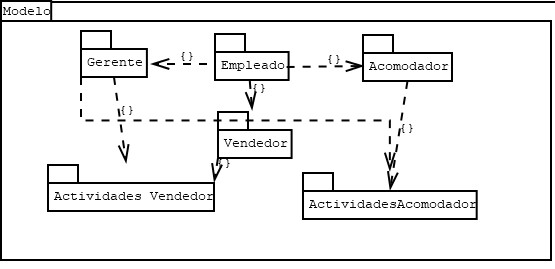
La prioridad de este sistema es facilitar la comprensión, modularidad y garantizar la portabilidad para que el usuario pueda acceder al inventario de la tienda de música.

Tomando en cuenta principalmente estas características, el sistema será implementado utilizando una arquitectura basada en el patrón de arquitectura ModelViewController.

## 1.1. Paquetes de la arquitectura

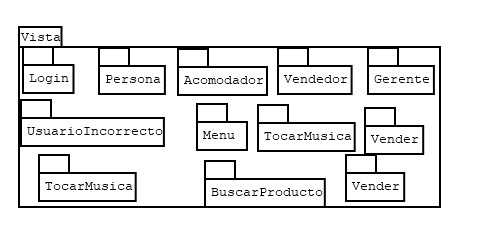
### 1.1.1. Modelo (Model)

Es la representación de la información con la cual el sistema opera, por lo tanto, gestiona todos los accesos a dicha información, tanto consultas como actualizaciones, implementando también los privilegios de acceso que se hayan descrito en las especificaciones de la aplicación (lógica de negocio). Envía a la 'vista' aquella parte de la información que en cada momento se le solicita para que sea mostrada (típicamente al usuario). Las peticiones de acceso o manipulación de información llegan al 'modelo' a través del 'controlador’. En este caso, tenemos 3 diferentes tipos de acceso a la información del proyecto, dependiente de los privilegios del usuario.



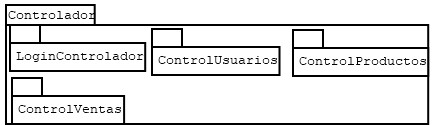
### 1.1.2. Vista (View)

Presenta el 'modelo' (información y lógica de negocio) en un formato adecuado para interactuar (usualmente la interfaz de usuario), por tanto, requiere de dicho 'modelo' la información que debe representar como salida. Las funciones que permite el programa se ven representadas principalmente aquí.



### 1.1.3. Controlador (Controller)

Responde a eventos (usualmente acciones del usuario) e invoca peticiones al 'modelo' cuando se hace alguna solicitud sobre la información (por ejemplo, un registro en la base de datos). También puede enviar comandos a su 'vista' asociada si se solicita un cambio en la forma en que se presenta el 'modelo', por tanto se podría decir que el 'controlador' hace de intermediario entre la 'vista' y el 'modelo'.



## 1.2. Definición del ambiente de implementación

|  |  |  |
| --- | --- | --- |
| **Concepto** | **Herramienta** | **Versión** |
| Lenguaje de programación | Java | 8 |
| Diagramador UML | Dia | 0.97.2 |
| IDE | Netbeans | 8.2 |
| Control de versiones | Github |  |

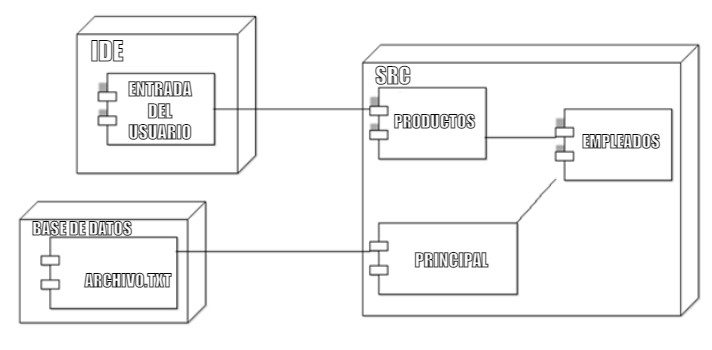
# 2. Vista de despliegue

## 2.1. Descripción

El sistema se compone principalmente de 3 paquetes que nos permitirán el manejo de toda la información y modular las acciones que son necesarias realizar en el programa.

* Productos. Aquí se encuentran toda la información sobre cada tipo de producto, incluso el manejo de estos, como editar, añadir, eliminar, buscar y listar.
* Empleados. Este módulo tiene como característica el manejo de los empleados de la tienda, principalmente sus atributos.
* Principal. En este paquete veremos el eje principal del programa, las operaciones y acciones que se deben hacer. Además de las acciones principales del programa, como vender o el módulo de aparatos de ambiente.

## 2.2. Modelo de despliegue



# 3. Vista lógica

## 3.1. Descripción

Esta sección se explica de manera general la descomposición del modelo mediante jerarquía de paquetes.

El proyecto codificado cuenta con tres paquetes

1. Principal

Este cuenta con dos clases, la principal llamada el*main* que nos servirá para ejecutar todo nuestro código y por otra parte la clase *Bdatos* que hará la función de darnos un archivo .csv que hará alusión a un ticket.

1. Empleados

En este paquete contamos con la creación de objetos los cuales simularán a *trabajadores, gerentes* y *acomodadores,* teniendo como clase madre a una clase *persona.* Cada uno contará con diversas funciones, por lo cual será aplicado el poliformismo.

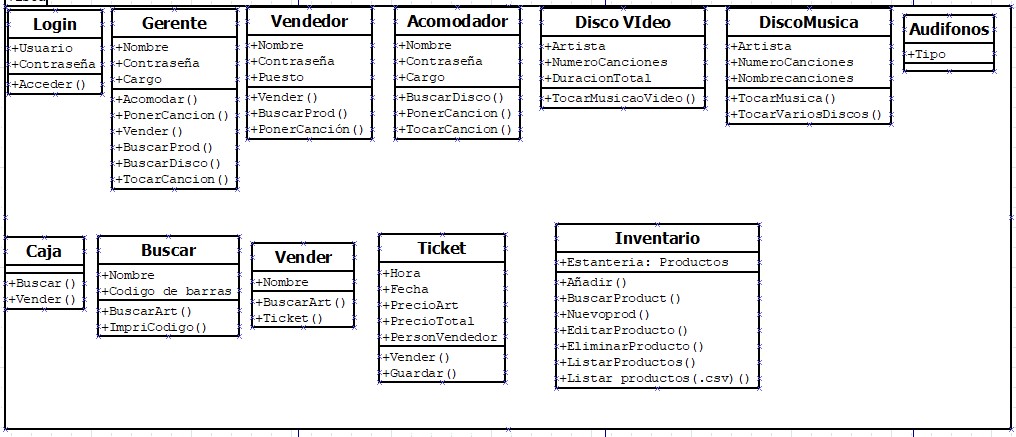
1. Productos

Por último, tenemos a nuestro paquete productos, almacenará a las clases *inventario, productos, audífonos, disco, discomusica* y *discovid* las cuales servirán para recrear la producción que se utiliza en una tienda de música.

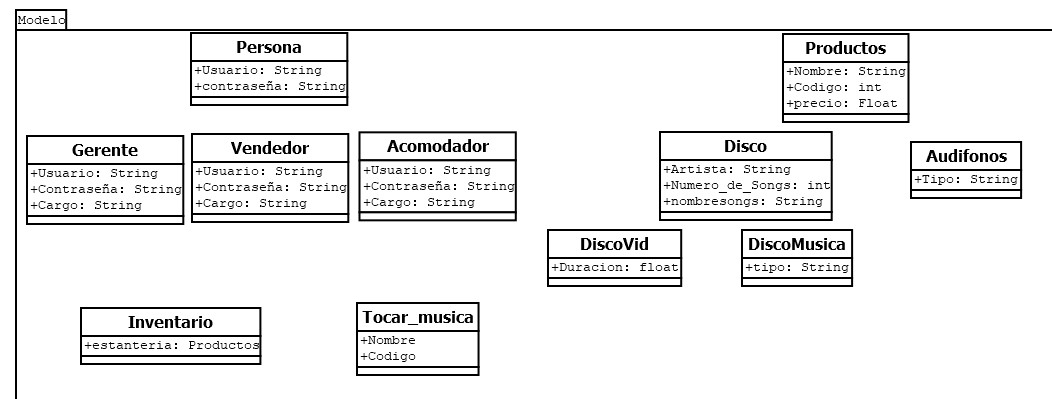
## 3.2. Identificación de las clases

Esta sección contendrá los diagramas que describen el comportamiento estático del sistema y los tipos de relaciones existentes entre las clases.

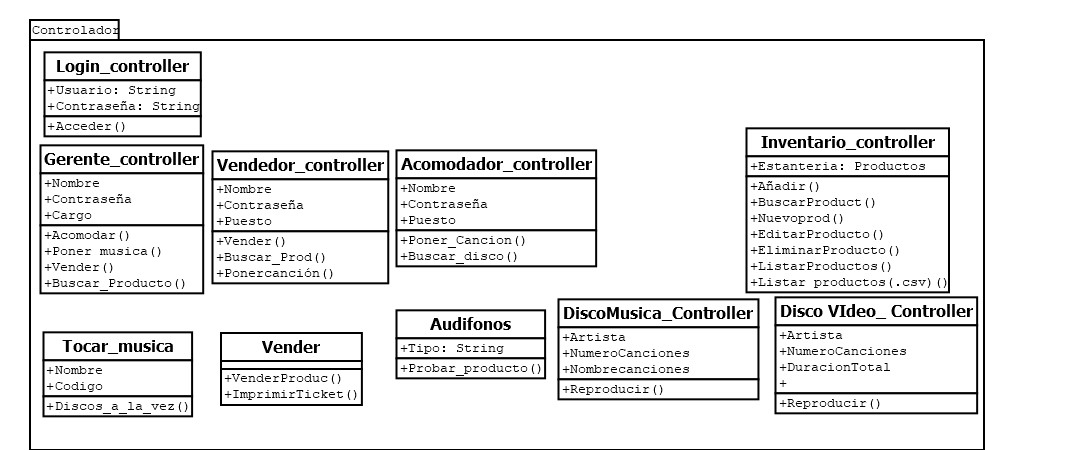
### 3.2.1. Clases de vista



### 3.2.2. Clases de modelo



### 3.2.3. Clases de controlador



# 4. Vista de datos

## 4.1. Descripción

Para hacer persistente la información del sistema de administración de asesorías se utilizará una base de datos que es un conjunto de datos pertenecientes a un mismo contexto y almacenados sistemáticamente para su posterior uso.

# 5. Vista dinámica

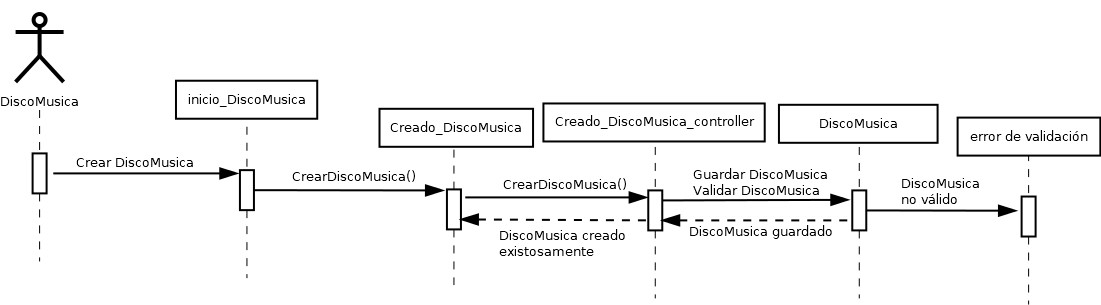
## 5.1. Descripción

Cada caso de uso es representado con un diagrama de secuencia, éste muestra cómo se comportan los objetos entre ellos a través del tiempo. Los diagramas de secuencia representarán el comportamiento dinámico de los casos de uso. El diagrama de estados, indicará cómo navega el usuario a través de las interfaces del sistema. Cada diagrama tendrá un estado inicial que se identificará con una circunferencia negra que rodea un punto negro, los estados serán representados por un óvalo con el nombre de la interfaz en la que se encuentra el usuario al realizar alguna acción, las acciones estarán determinadas en el documento por flechas con un estado origen, una acción y el estado destino.

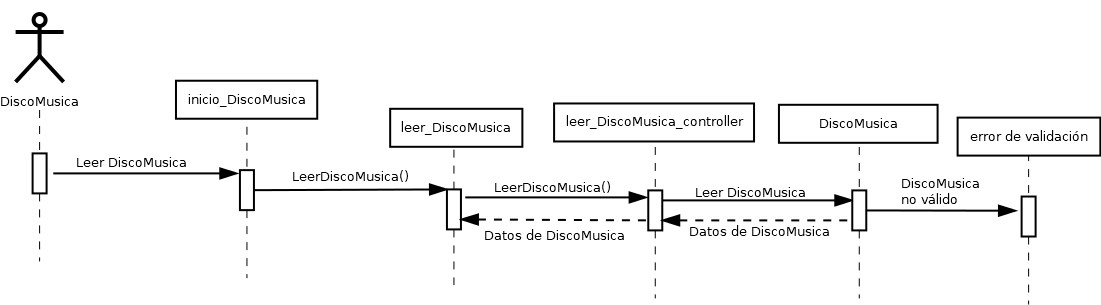
**5.2. Diagramas de secuencia**

## 5.3. Productos

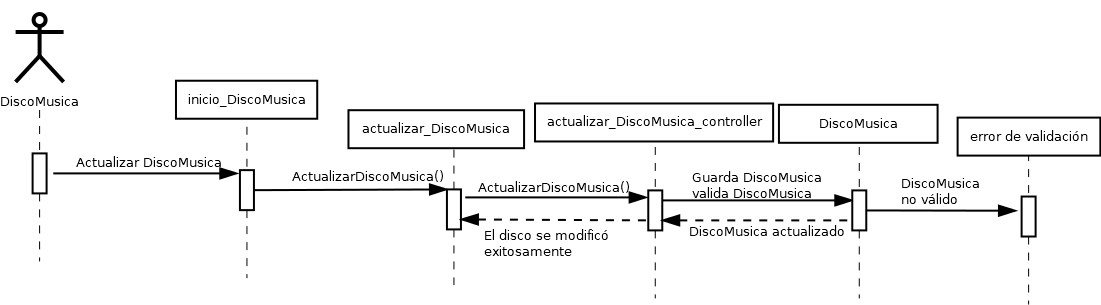
### 5.3.1. Crear DiscoMusica



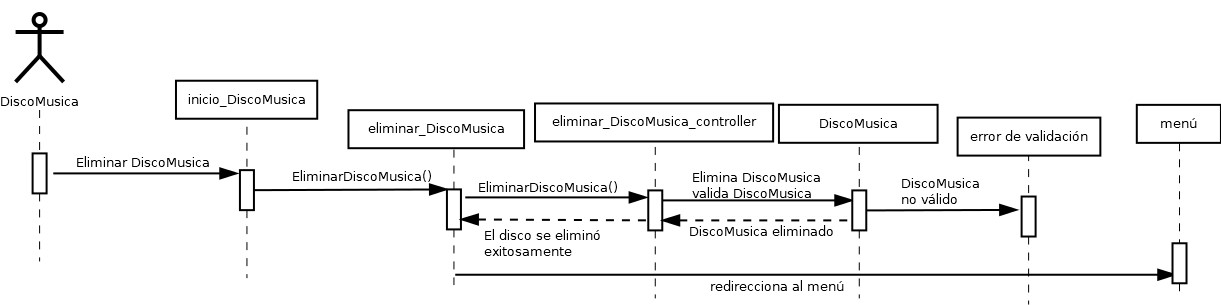
### 5.3.2. Leer DiscoMusica



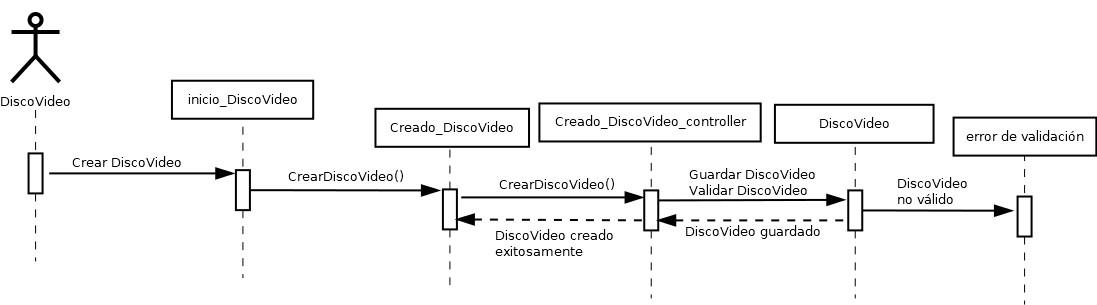
### 5.3.3. Actualizar DiscoMusica



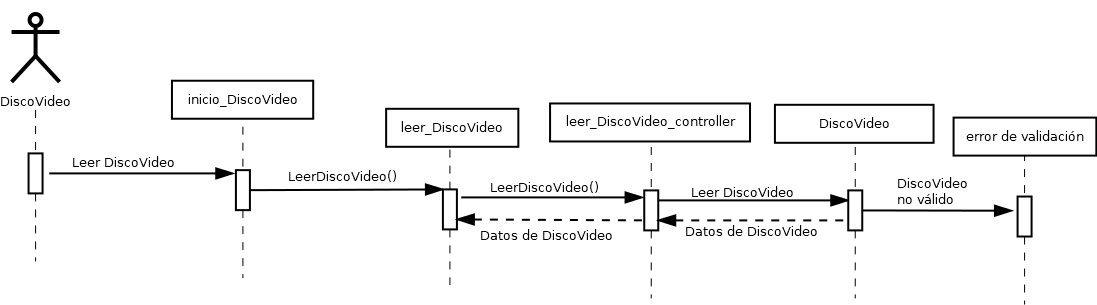
### 5.3.4. Eliminar DiscoMusica



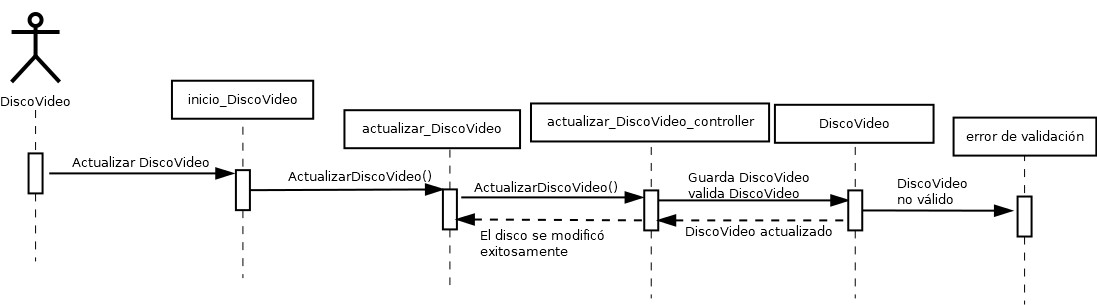
### 5.3.5. Crear DiscoVideo



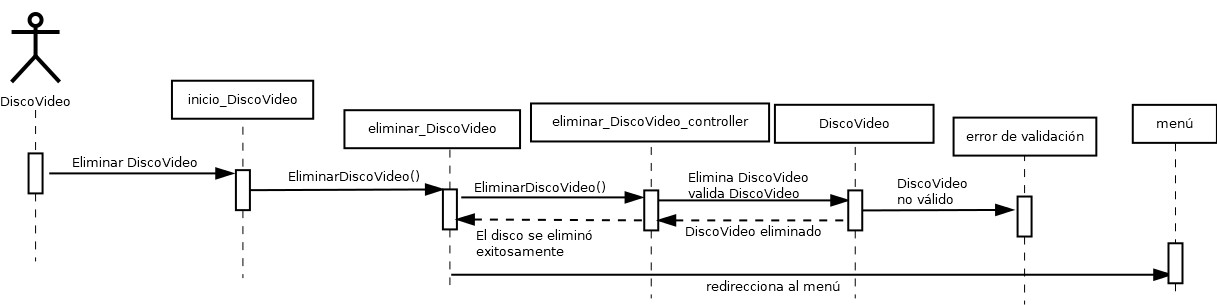
### 5.3.6. Leer DiscoVideo



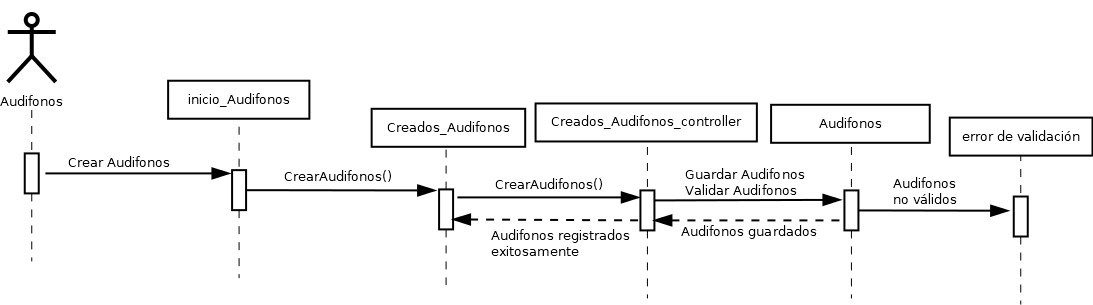
### 5.3.7. Actualizar DiscoVideo



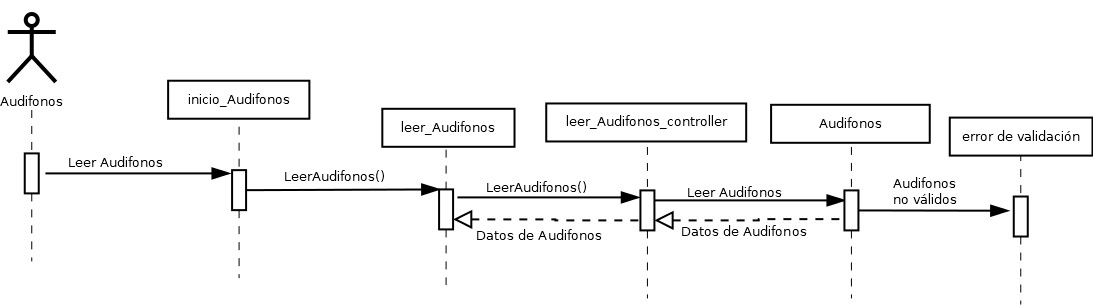
### 5.3.8. Eliminar DiscoVideo



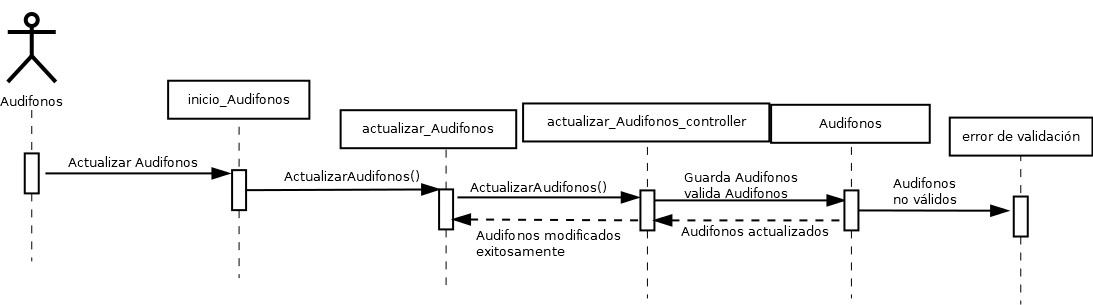
### 5.3.9. Crear Audifonos



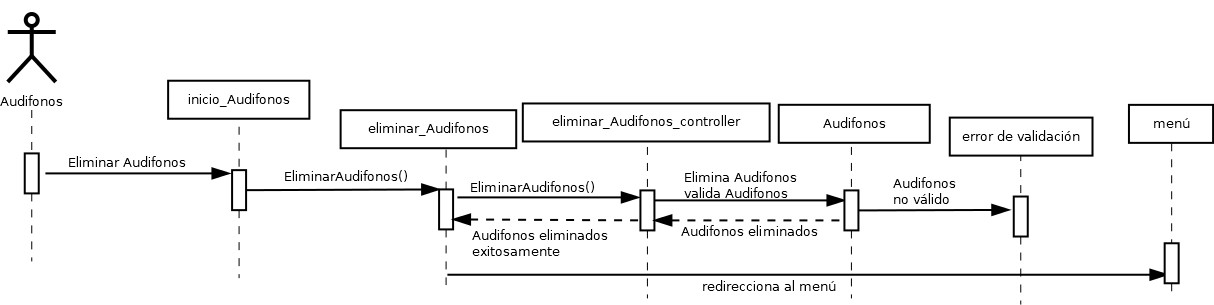
### 5.3.10. Leer Audifonos



### 5.3.11. Actualizar audifonos

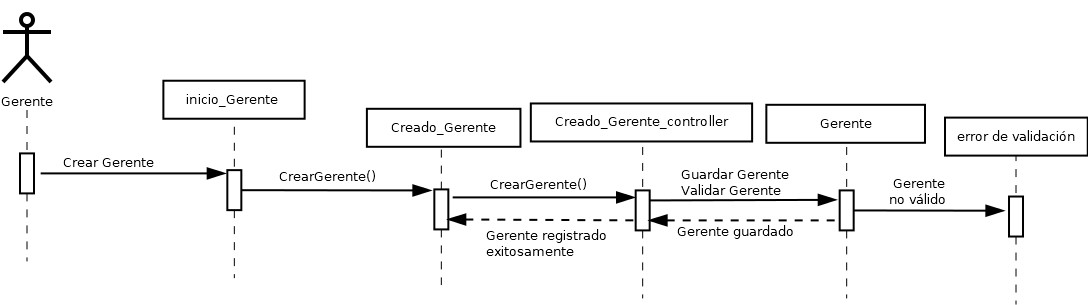


### 5.3.12. Eliminar Audifonos

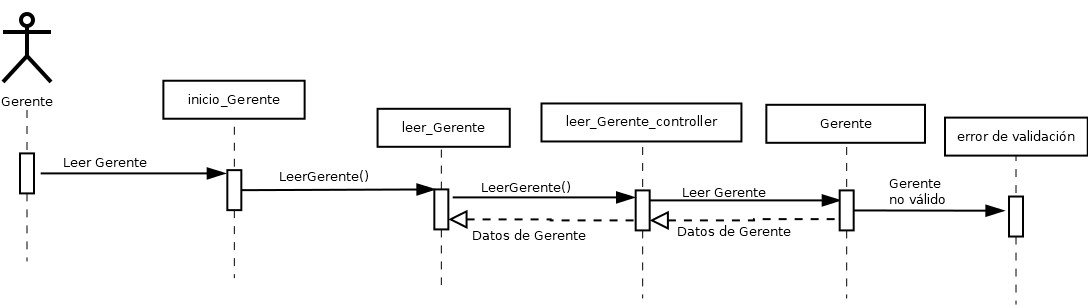


## 5.4. Login

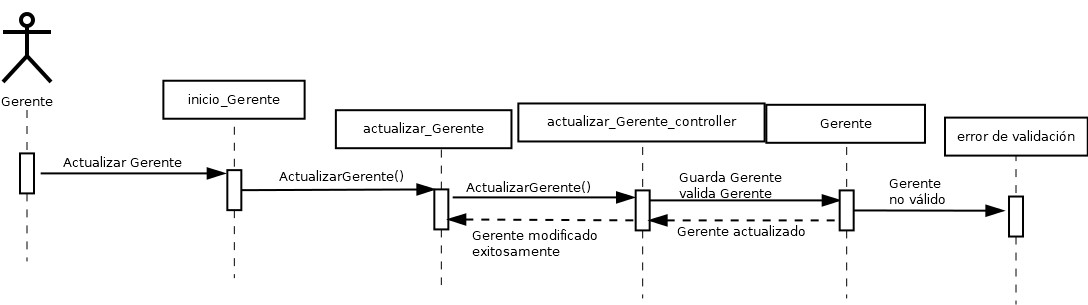
### 5.4.1. Crear gerente



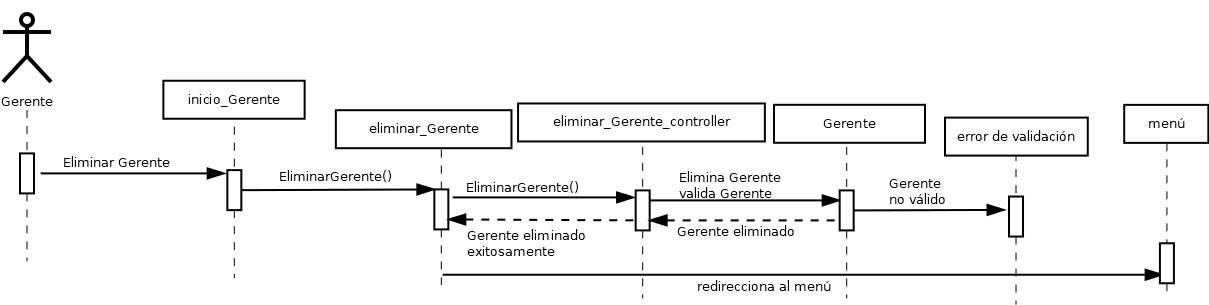
### 5.4.2. Leer gerente



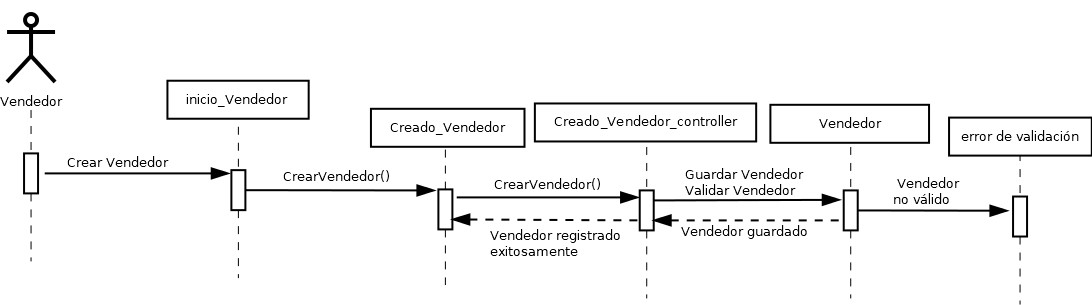
### 5.4.3. Actualizar gerente



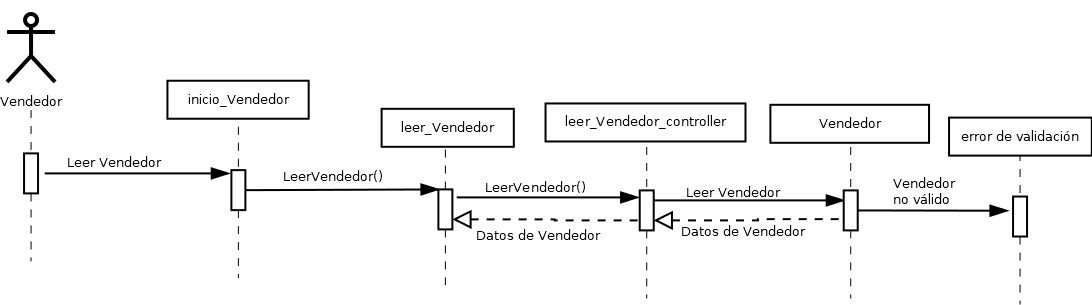
### 5.4.4. Eliminar gerente



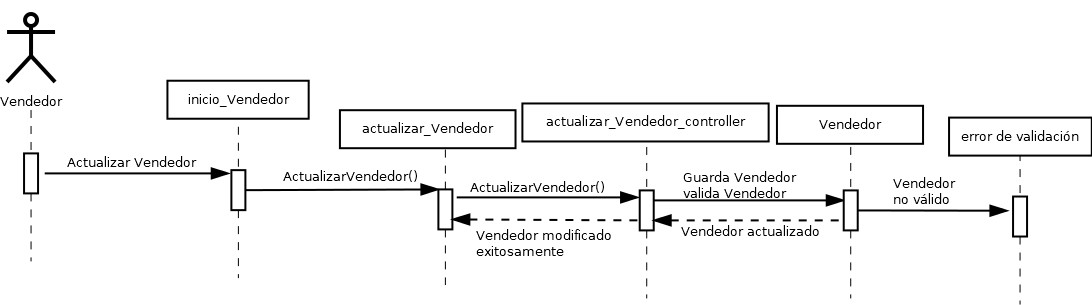
### 5.4.5. Crear vendedor



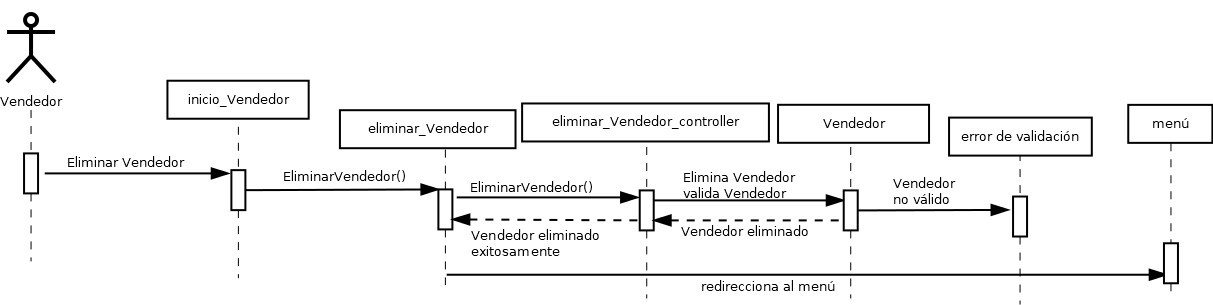
### 5.4.6. Leer vendedor



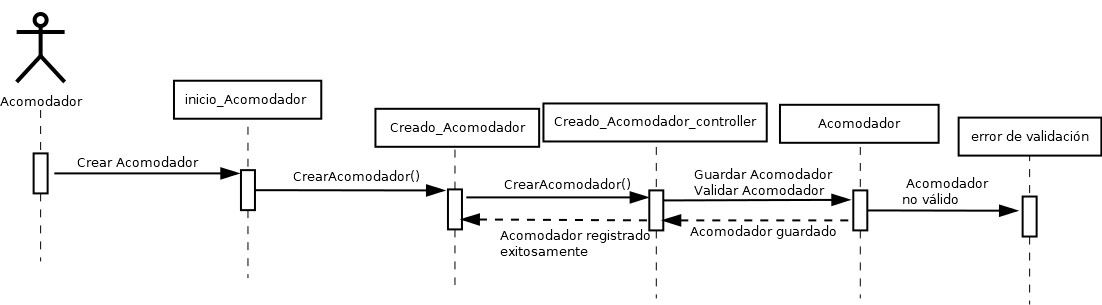
### 5.4.7. Actualizar vendedor



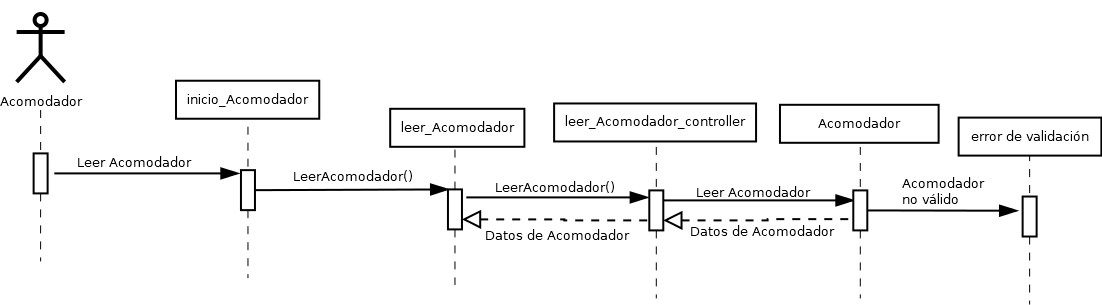
### 5.4.8. Eliminar vendedor



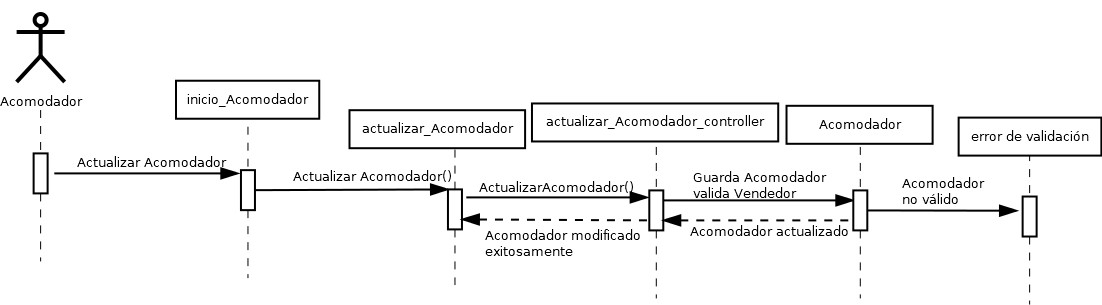
### 5.4.9. Crear acomodador



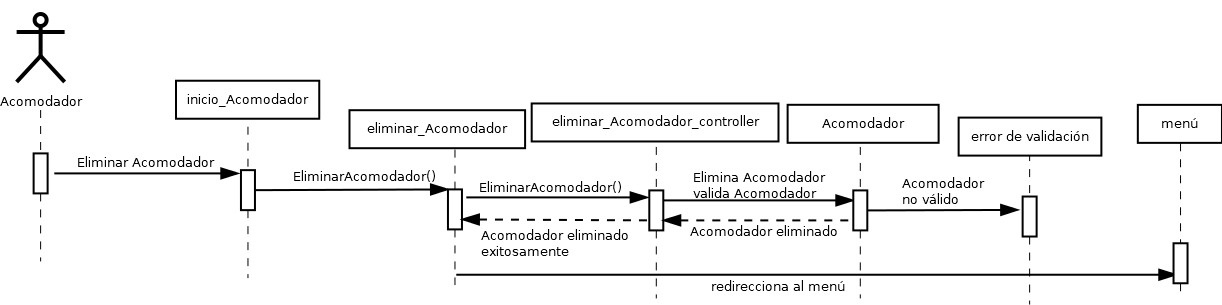
### 5.4.10. Leer acomodador



### 5.4.11. Actualizar acomodador

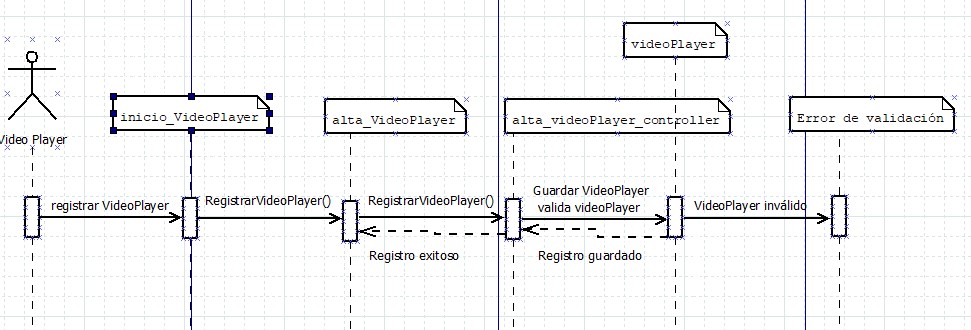


### 5.4.12. Eliminar acomodador

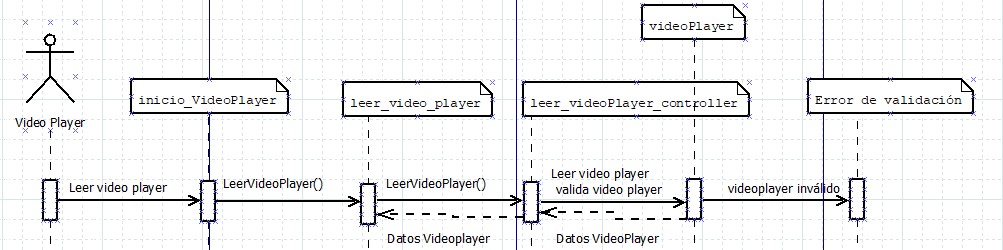


## 5.5. Aparato de ambiente

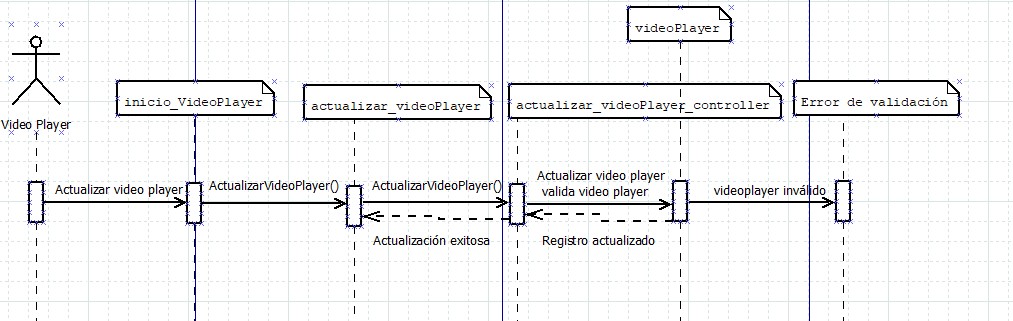
### 5.4.1. Crear VideoPlayer



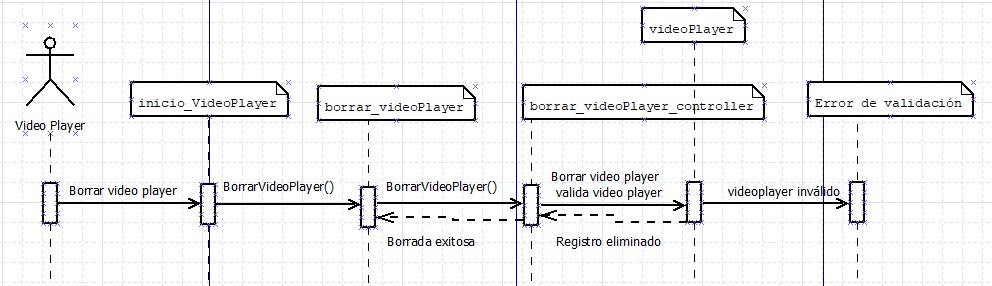
### 5.4.2. Leer VideoPlayer



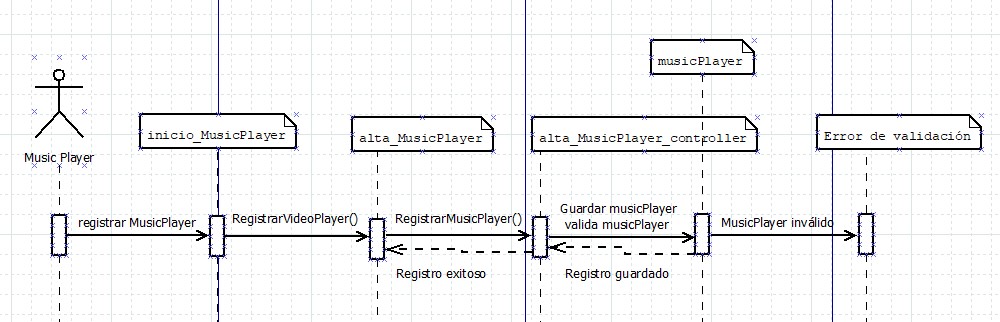
### 5.4.3. Actualizar VideoPlayer



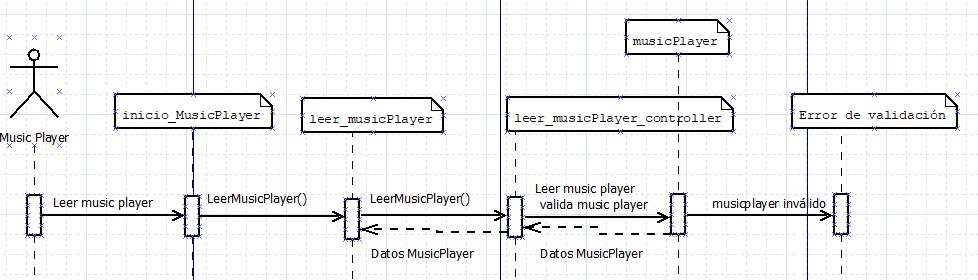
### 5.4.4 Borrar VideoPlayer



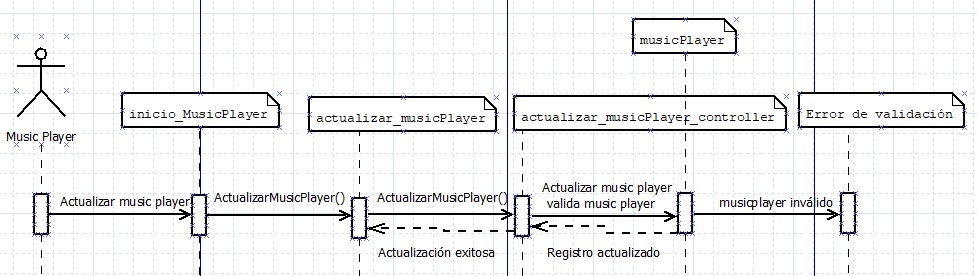
### 5.4.5 Crear MusicPlayer



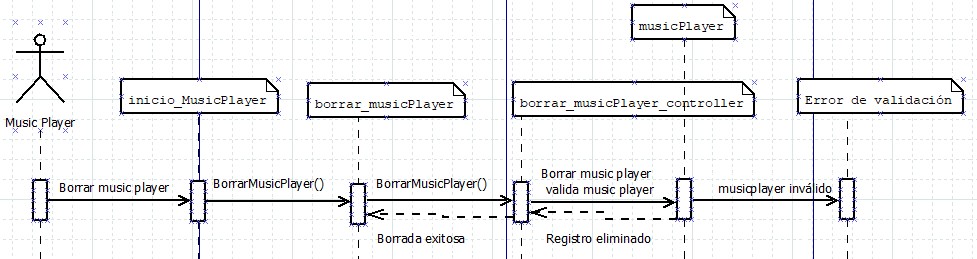
### 5.4.6. Leer MusicPlayer



### 5.4.7. Actualizar MusicPlayer

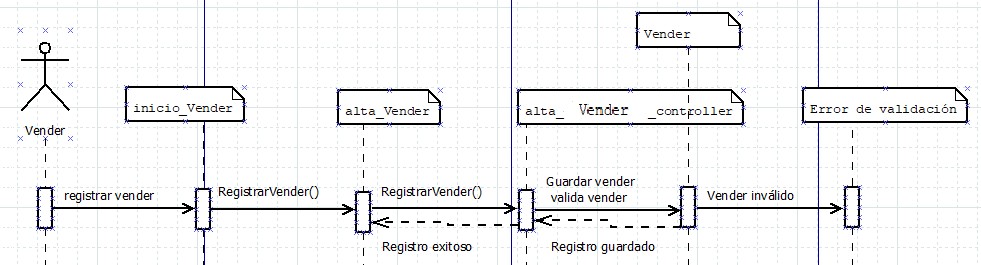


### 5.4.8. Borrar MusicPlayer

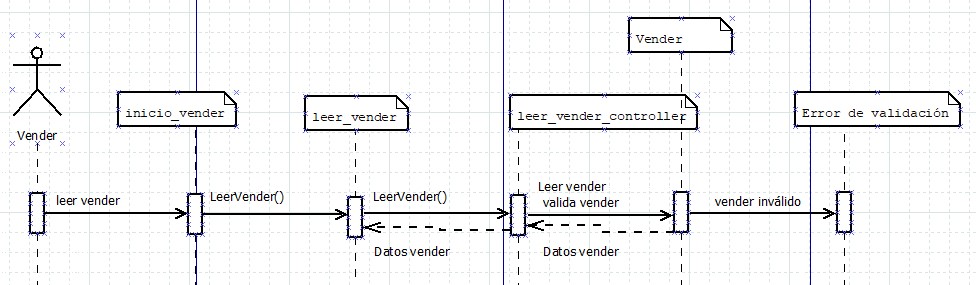


## 5.6. Caja

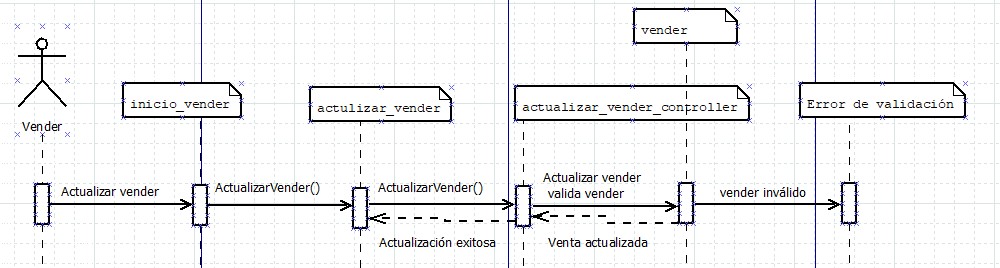
### 5.5.1. Crear Vender



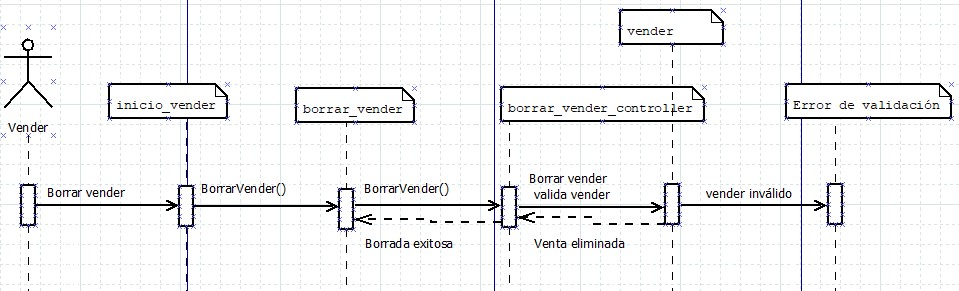
### 5.5.2. Leer Vender



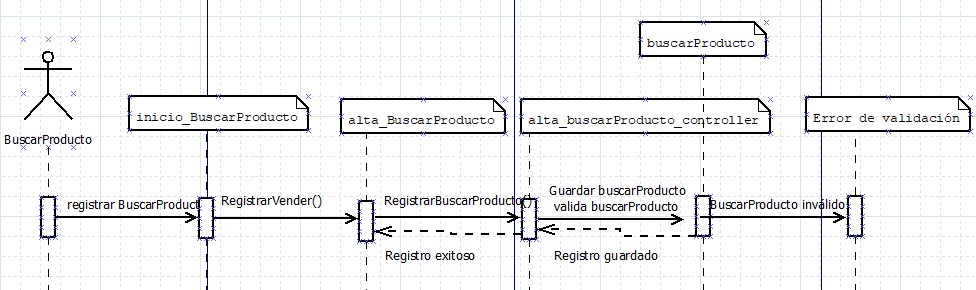
### 5.5.3. Actualizar Vender



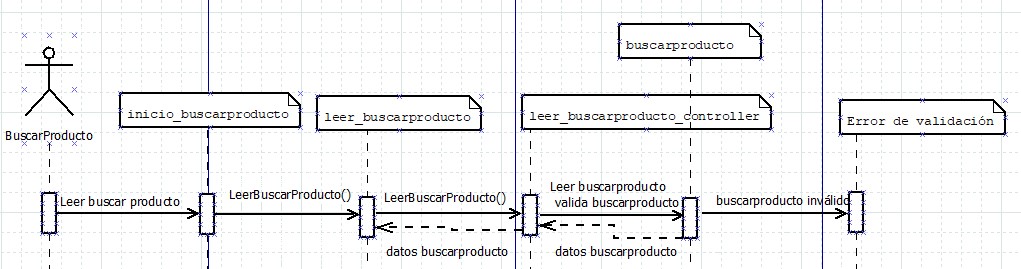
### 5.5.4. Borrar Vender



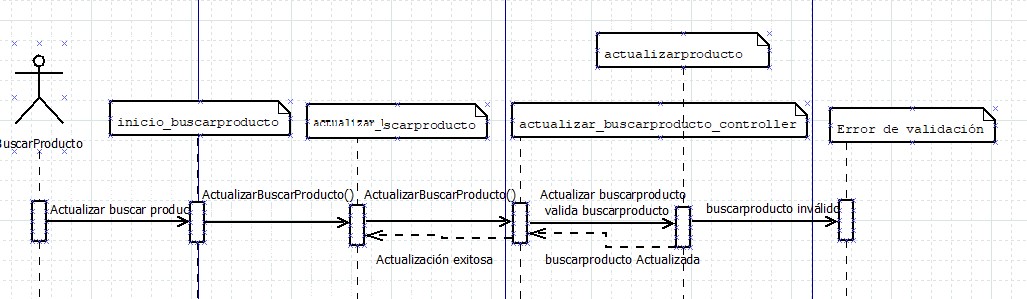
### 5.5.5. Crear BuscarProducto



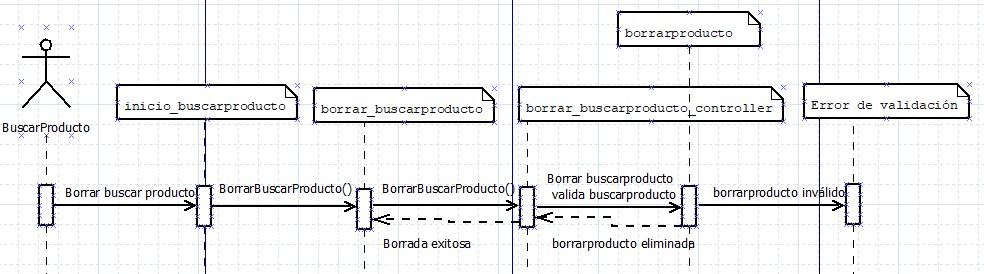
### 5.5.6. Leer BuscarProducto



### 5.5.7. Actualizar BuscarProducto



### 5.5.8. Borrar BuscarProducto



## 5.7. Diagramas de navegación

