Diseny de software

TotSeries

Gerard Cervilla - NIUB **16890090**

Abel Dos Ramos Sánchez - NIUB **15062106**

Índice

Diagrama de Casos de Uso ……………………………………………………………… Pág 2

Diagrama de Casos de Uso Textuales…………………………………………………... Pág 3

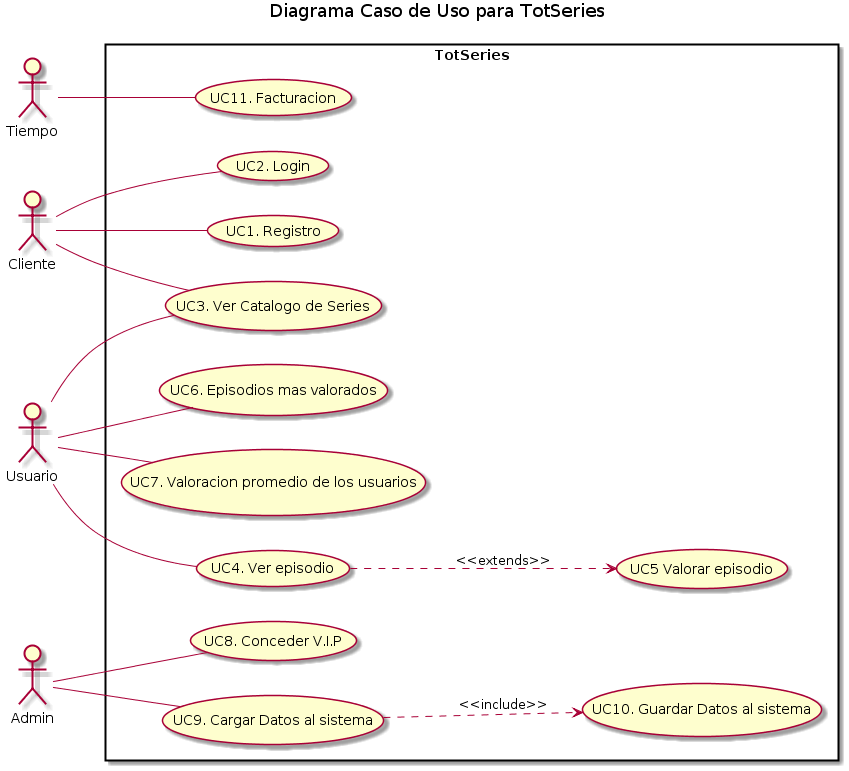
Diagrama de Dominio ……. ……………………………………………………………… Pág 10

Observaciones ……………..……………………………………………………………… Pág 12

Conclusiones..……………………………………………………………………………… Pág 14

Apéndice …………………………………………………………………………………… Pág 15

Diagrama de Caso de Uso



**Descripción**

Tenemos 4 actores: Usuario, Tiempo, Cliente y Admin.

Un cliente es una persona que no se ha identificado en el sistema. Si se ha identificado mediante (UC2) se vuelve un Usuario del sistema. Tanto el cliente como el usuario puede ver el catálogo de series. El usuario puede además, ver los episodios más valorados, ver el promedio de valoración de los usuarios, ver un episodio y valorar los episodios vistos.

El administrador por su parte maneja el sistema y puede cargar datos en el sistema (La lista de series que tiene TotSeries), guardarlos y cambiar el estado de un usuario a VIP.

Por último, el tiempo efectúa los cobros a los Usuarios por la visualización de los episodios a final de cada mes.

Diagrama de Caso de Uso Textual

|  |
| --- |
| **Descripción:** UC1. Registro |
| **Actores:** Cliente |
| **Precondiciones:** Ninguna |
| **Flujo básico:**   1. El sistema pregunta al usuario el nombre de usuario. 2. El usuario introduce un nombre de usuario. 3. El sistema pregunta al usuario su password, nombre real, nacionalidad y fecha de nacimiento. 4. El usuario introduce los datos solicitados por el sistema 5. El sistema crea un Usuario para el Cliente |
| **Flujo alternativo:**   1. El usuario introduce un nombre de usuario ya existente 2. El sistema muestra un error 3. El sistema vuelve al punto 1 del flujo básico.   4.a El usuario introduce un dato incorrecto:   1. El sistema muestra un error 2. El sistema vuelve al punto correspondiente del flujo básico.   2.b - 4.b El usuario cancela su registro:   * El sistema no registra un usuario. |
| **Postcondiciones:** Un Usuario es añadido al sistema. |

|  |
| --- |
| **Descripción:** UC2. Login |
| **Actores:** Cliente |
| **Precondiciones:** Ninguna |
| **Flujo básico:**   1. El sistema pregunta al Cliente el nombre de usuario 2. El Cliente introduce un nombre de usuario. 3. El sistema pregunta al Cliente su password 4. El Cliente introduce un password. 5. El sistema identifica el Cliente como un Usuario |
| **Flujo alternativo:**  2.a - 4.a El Cliente introduce un dato incorrecto:   1. El sistema muestra un error 2. El sistema vuelve al punto correspondiente del flujo básico.   2.b - 10.b El Cliente cancela su identificación:   * El sistema no identifica al usuario. |
| **Postcondiciones:** El cliente es identificado y pasa a ser un Usuario |

|  |
| --- |
| **Descripción:** UC3. Ver Catálogo |
| **Actores:** Cliente, Usuario |
| **Precondiciones:** El catálogo debe estar cargado en el sistema |
| **Flujo básico:**   1. El sistema muestra al cliente/usuario el catálogo de series que tiene. |
| **Flujo alternativo:** Ninguna |
| **Postcondiciones:** Ninguna |

|  |
| --- |
| **Descripción:** UC4. Ver Episodio |
| **Actores:** Usuario |
| **Precondiciones:** Ver el Catálogo (UC3) |
| **Flujo básico:**   1. El usuario busca la serie que quiere ver en el catálogo 2. El usuario selecciona la temporada y episodio que quiera ver. 3. El usuario clica en reproducir. 4. El sistema muestra el streaming del video. 5. El usuario mira el video 6. Al finalizar la reproducción, el Sistema pregunta al usuario si quiere valorar el episodio. 7. El cliente acepta y va al (UC5) 8. El usuario sale de la pantalla de streaming |
| **Flujo alternativo:**   1. El usuario sale de la reproducción antes de finalizar el video  * La visualización se cancela y el sistema no cobra al usuario.  1. El usuario abre otro video mientras mantiene abierto el actual 2. El sistema informará al usuario que solo puede ver 1 video a la vez. 3. El sistema no mostrará el segundo video.  * La segunda visualización se cancela y el sistema no cobra al usuario.   1.a - 2.a - 3.a El usuario cambia de opinión y decide ver otra serie   * La visualización se cancela y el sistema no cobra al usuario.  1. El usuario rechaza valorar un episodio y continua en el paso 8. |
| **Postcondiciones:** El usuario puede ver otro video.  Se añade 1 euro a la factura de visualización del Usuario.  Se añade un descuento a la facturación si el Usuario es un Vip. |

|  |
| --- |
| **Descripción:** UC5. Valorar Episodio |
| **Actores:** Usuario |
| **Precondiciones:** El usuario viene de (UC4) punto 7. |
| **Flujo básico:**   1. El sistema pregunta al usuario su Id 2. El usuario introduce su Id. 3. El sistema pregunta al usuario la nota que desea conceder entre 1-5 (por defecto 3). 4. El usuario selecciona su nota. 5. El sistema añade la valoración a la serie. 6. El usuario retorna al (UC4) punto 8. |
| **Flujo alternativo:**   1. El usuario introduce mal su Id 2. El sistema muestra un error 3. El sistema vuelve al punto 1 del flujo básico.   2.a, 4.a El usuario cancela su valoración:   1. El usuario retorna al (UC4) punto 8.  * El sistema no guarda la valoración a la serie. |
| **Postcondiciones:** Una valoración es añadida a un episodio. |

|  |
| --- |
| **Descripción:** UC6. Episodios más valorados |
| **Actores:** Usuario |
| **Precondiciones:** Ninguna |
| **Flujo básico:**   1. El sistema devuelve al Usuario una lista con los episodios más valorados |
| **Flujo alternativo:** Ninguna |
| **Postcondiciones:** Ninguna |

|  |
| --- |
| **Descripción:** UC7. Valoración promedio de los usuarios |
| **Actores:** Usuario |
| **Precondiciones:** Ninguna |
| **Flujo básico:**   1. El sistema pregunta al Usuario de que serie quiere saber la valoración media de los usuarios. 2. El sistema muestra la lista de series. 3. El Usuario selecciona una serie. 4. El sistema muestra la media de todos los episodios de la serie seleccionada. |
| **Flujo alternativo:** Ninguna |
| **Postcondiciones:** Ninguna |

|  |
| --- |
| **Descripción:** UC8. Conceder VIP |
| **Actores:** Admin |
| **Precondiciones:** Ninguna |
| **Flujo básico:**   1. El sistema devuelve una lista de Usuarios con estado básico 2. El admin selecciona uno de ellos 3. El sistema cambia el estado de un usuario básico a usuario VIP |
| **Flujo alternativo:**Ninguna |
| **Postcondiciones:** El estado de un usuario cambia a VIP |

|  |
| --- |
| **Descripción:** UC9. Cargar Datos del sistema |
| **Actores:** Admin |
| **Precondiciones:** Ninguna |
| **Flujo básico:**   1. El sistema pide al Admin que introduzca la ruta del sistema del fichero a cargar 2. El Admin introduce la ruta del sistema del fichero 3. El sistema lee los datos del fichero y crea su contenido 4. El sistema redirige al Admin al (UC10) |
| **Flujo alternativo:**   1. El sistema encuentra algún error con el fichero.    1. El sistema muestra un mensaje de error    2. El sistema retorna al Admin al punto 2 del flujo básico  * Los datos NO son cargados en el sistema  1. El admin cancela la Carga de datos.    1. El sistema cerrará la aplicación  * Los datos NO son cargados en el sistema |
| **Postcondiciones:** Los datos del sistema son cargados en el sistema |

|  |
| --- |
| **Descripción:** UC10. Guardar datos del sistema |
| **Actores:** Admin |
| **Precondiciones:** El Admin viene de (UC9) punto 4. |
| **Flujo básico:**   1. El sistema pide al Admin que confirme que los datos cargados son correctos y que quiere persistir los. 2. El Admin confirma. 3. El sistema persiste los datos cargados por el Admin. |
| **Flujo alternativo:**   1. El Admin selecciona que no quiere persistir los datos cargados del fichero    1. El sistema retorna al Admin al (UC9)  * Los datos del sistema NO son persistidos |
| **Postcondiciones:** Los datos del sistema son persistidos |

|  |
| --- |
| **Descripción:** UC11. Facturar Usuario |
| **Actores:** Tiempo |
| **Precondiciones:** La fecha es final de mes |
| **Flujo básico:**   1. El sistema cobra a cada usuario la cantidad generada por cada visualización de los episodios. |
| **Flujo alternativo:** |
| **Postcondiciones:** El sistema cobra a cada usuario una cantidad de dinero |

Diagrama de Dominio

Hay 18 entidades:

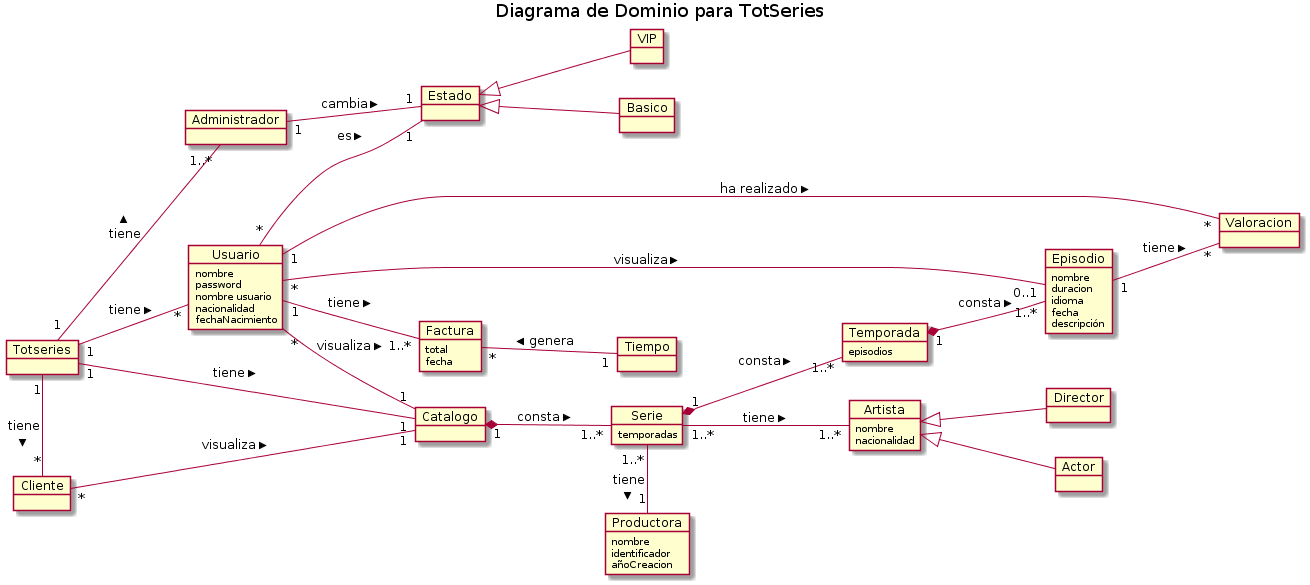
*TotSeries* tiene *Clientes*, *Usuarios*, *Admins* y un *Catálogo*.

Un Usuario tiene un estado que puede ser VIP o Básico y múltiples facturas que se generan cada mes(Tiempo), con el total a pagar y la fecha de pago.

Un *Catálogo* se compone de *Series*, estas se componen de *Temporadas*, y estas a su vez de *Episodios*.

Una *Serie* ha sido hecha por una *Productora* y por varios *Artistas*, ya sean *Actores* o *Directores*

Un *Episodio* tiene muchas *Valoraciones* que han sido hechas por múltiples *Usuarios* después que dichos *Usuarios* hayan visualizado un *Episodio*.



Observaciones

Diagrama de Caso de Uso:

Actores:

Tenemos 4 actores.

El actor *Cliente* es una persona que ha entrado en la aplicación pero no está identificada como *Usuario*. Esta persona puede ver el catálogo de series (UC3) y puede crear una cuenta de usuario (UC1) o identificarse como tal (UC2). Este actor no puede heredar de Usuario ni viceversa porque cada uno tiene acciones que el otro no tiene.

El actor *Usuario* es un *Cliente* que se ha identificado mediante el Login (UC2) . Este *Usuario* puede, ver los episodios más valorados por la comunidad (UC6) , ver la media de las valoraciones hechas por los Usuarios, de las series (UC7) y puede visualizar un episodio que haya elegido (UC4) . Una vez finalizado dicho episodio podrá dejar, si así lo desea, su valoración sobre el episodio visto (UC5).

En el diagrama de caso de uso textual, UC4-UC7 tenían la precondición de estar logeados en el sistema, pero dado que el actor era un Usuario, ya estaba implícito que se había identificado en el sistema, de modo que esta precondición fue removida.

El actor Admin es el encargado de manejar y velar para que el sistema funcione. Un Admin al iniciar la aplicación ha de cargar la información de todas las series (UC9) y si todo va bien, guardar los cambios (UC10). También puede cambiar el estado de un Usuario para que este sea un VIP (UC7).

En un principio VIP era una un actor que heredaba de Usuario, pero más tarde lo descartamos porque en el diagrama de dominio, hicimos que VIP fuera un estado de Usuario.

El actor Tiempo es una tarea encargada por el sistema para que se ejecute cada cierto tiempo. En este caso, cada mes el sistema ha de cobrar a los Usuarios la cantidad de proporcional a las visualizaciones que haya realizado (UC11).

Observaciones

Diagrama de Dominio:

Catálogo, Series, Temporadas, Episodios, están formados por composición uno respecto a otro. Se barajó la idea de que estas asociaciones fueran agrupaciones en vez de composiciones, porque TotSeries podría tener series que únicamente fueran anunciadas pero que no hayan salido, de forma que no permitiese su visualización. Se descartó por el hecho que TotSeries es una plataforma de visualización y si una serie no se puede visualizar, no tendría sentido estar en la plataforma.

VIP en un principio era una entidad que heredaba de Usuario, pero se decidió cambiarlo a un tipo de estado. Ahora un Usuario tiene un estado, que puede ser Básico, o Vip. Estas dos entidades son herencia de Estado.

Artista en un principio contenía un nombre y tenía herencia de Actor, Director y Productora, pero luego se cambio a que productora fuera una entidad separada y que Artista contuviera nombre y nacionalidad y que únicamente tuviera herencia de Actor y Director, ya que conceptualmente no tiene sentido que una productora sea un artista.

Se discutió si Cliente y Tiempo eran entidades porque eran actores en el diagrama de casos de uso. Se decidió que eran entidades porque, aunque no guarden atributos interactúan con el sistema y en el diagrama de Dominio puede haber entidades que más tarde no serán Clases.

Conclusiones

Viendo los cambios que hemos realizado o pudimos haber realizado en el apartado de Observaciones, nos damos cuenta que no hay una única solución y la planificación es subjetiva a cómo cada uno interpreta el texto por lo que es muy importante que los requisitos que se quieren del sistema sean muy explicitos.

Se Observa la importancia que tienen realizar correctamente la entrevista a los posibles usuarios que pueden participar. Al momento de realizar la entrevista se deben hacer recomendaciones sobre el sistema para al mismo tiempo saber que mejoras se pueden realizar a futuro una vez el sistema cumpla los requisitos mínimos solicitados y así poder hacer el sistema en base a futuras mejoras.

Luego habría que ver la viabilidad de nuestro diseño cuando se realice el diagrama de Clases y posteriormente la escritura de código, pues otra vez, un fallo de diseño que se descubra en esta etapa puede conllevar a rehacer todo o parcialmente el diseño anterior.

Apéndice

Codigo Diagrama de Uso PlantText

@startuml

left to right direction

skinparam packageStyle rect

title Diagrama Caso de Uso para TotSeries

rectangle TotSeries {

usecase (UC3. Ver Catalogo de Series) as UC3

usecase (UC1. Registrarse) as UC1

usecase (UC2. Logearse) as UC2

usecase (UC4. Ver episodio ) as UC4

usecase (UC5 Valorar episodio) as UC5

usecase (UC6. Episodios mas valorados) as UC6

usecase (UC7. Valoracion promedio de los usuarios) as UC7

usecase (UC8. Conceder V.I.P) as UC8

usecase (UC9. Cargar Datos al sistema) as UC9

usecase (UC10. Guardar Datos al sistema) as UC10

usecase (UC11. Facturacion) as UC11

}

actor Tiempo

actor Cliente

actor Usuario

actor VIP

actor Admin

Cliente -- UC1

Cliente -- UC2

Cliente -- UC3

Usuario -- UC3

Usuario -- UC4

Usuario -- UC6

Usuario -- UC7

Tiempo -- UC11

Usuario <|- VIP

Admin -- UC8

Admin -- UC9

UC9 -.> UC10 : <<include>>

UC4 -.> UC5 :<<extends>>

Codigo Diagrama de Dominio PlantText

@startuml

left to right direction

title Diagrama de Dominio para TotSeries

object Totseries

object Episodio {

nombre

duracion

idioma

fecha

descripción

}

object Artista {

nombre

nacionalidad

}

object Productora {

nombre

identificador

añoCreacion

}

object Usuario {

nombre

password

nombre usuario

nacionalidad

fechaNacimiento

}

Object Factura{

total

fecha

}

Object Temporada{

episodios

}

object Catalogo

object Serie{

temporadas

}

Object Valoracion

Object Tiempo

Object Cliente

Object VIP

Object Administrador

Object Actor

Object Director

Object Estado

Totseries "1" -- "1" Catalogo : tiene >

Totseries "1" - "\*" Cliente : tiene >

Totseries "1" -- "\*" Usuario : tiene >

Totseries "1" -- "1..\*" Administrador : tiene >

Usuario "1" -- "1..\*" Factura : tiene >

Usuario "1" -- "\*" Valoracion : ha realizado >

Usuario "\*" -- "1" Catalogo : visualiza >

Usuario "\*" -- "1" Estado : es >

Estado <|-- VIP

Estado <|-- Basico

Administrador "1" --"1" Estado : cambia >

Cliente "\*" -- "1" Catalogo: visualiza >

Catalogo "1" \*-- "1..\*" Serie : consta >

Serie "1" \*-- "1..\*" Temporada : consta >

Serie "1..\*" -- "1..\*" Artista : tiene >

Serie "1..\*"- "1" Productora : tiene >

Artista <|-- Actor

Artista <|-- Director

Temporada "1" \*-- "1..\*" Episodio : consta >

Episodio "0..1" -- "\*" Usuario : visualiza <

Episodio "1" -- "\*" Valoracion : tiene >

Factura "\*" -- "1" Tiempo : genera <