

# UNIVERSITAT<sub>DE</sub> BARCELONA

**ASSIGNATURA:** Disseny de Software

**PRÀCTICA:** TotSeries

**GRUP:** Sergi Planes, Víctor Maestro, Josep Martí

## Índex

INTRODUCCIÓ: .....	pàg. 02
DESENVOLUPAMENT:.....	pàg. 02
·Diagrama de casos d'ús: .....	pàg. 02
·Model de domini: .....	pàg. 06
·Històries d'usuari: .....	pàg. 08
CONCLUSIONS: .....	pàg. 08
APÈNDIX: .....	pàg. 10

## 1 -. INTRODUCCIÓ:

En aquesta primera pràctica es proposa la realització d'un programa anomenat **TotSeries** que ha de complir una sèrie de especificacions indicades per l'usuari. El nostre objectiu es definir els diferents casos d'ús que es poden trobar durant el funcionament de l'aplicació, indicar el model de domini que ha de complir el programa i indicar les histories d'usuari d'uns determinats casos d'ús.

## 2 -. DESENVOLUPAMENT:

### ·Diagrama de casos d'ús:

En aquest apartat s'expliquen els diferents problemes que ens hem trobat al construir els diagrames de casos d'ús dels diferents fluxos de l'aplicació.

### CU2. Veure detall de la sèrie

La *figura 1* presenta el flux bàsic corresponent a com mostrar la informació d'una sèrie seleccionada en el catàleg de series, veure *figura 2*. D'altra banda, es preveu un flux alternatiu *figura 2* on la informació que es proporcionada no està disponible i per tant, el flux porta altre cop al principi de CU3 *figura 2*, veure catàleg de la sèrie.

<b>Descripció</b>	CU2. Veure detall de la sèrie
<b>Actors</b>	Usuari o Client
<b>Precondicions</b>	Cap
<b>Flux bàsic</b>	<ol style="list-style-type: none"><li>1. L'usuari o client selecciona una sèrie dins del catàleg.</li><li>2. El Sistema ensenya la informació de la sèrie i mostren els capítols disponibles.</li></ol>
<b>Flux alternatiu</b>	2.a. La sèrie encara no esta disponible i el sistema dona la informació general de la sèrie. Es torna a CU3
<b>Postcondicions</b>	Es mostra per pantalla tota la informació de la sèrie seleccionada o de les properes estrenes.

Figura 1. Diagrama de casos d'ús: Veure detall de la sèrie

### CU3. Veure catàleg de series

La *figura 2* mostra el cas d'ús de veure el catàleg de series. Es defineix un flux bàsic que descriu els dos passos necessaris per veure el catàleg de sèries disponibles. Recordar que forma part del flux alternatiu del cas de veure detall de la sèrie CU2, veure *figura 1*.

<b>Descripció</b>	CU3. Veure catàleg de series
<b>Actors</b>	Usuari o Client
<b>Precondicions</b>	Cap
<b>Flux bàsic</b>	<ol style="list-style-type: none"><li>1. L'usuari/Client selecciona visualitzar el catàleg de series.</li><li>2. El Sistema ensenya la llista del catàleg actual de les series.</li></ol>
<b>Flux alternatiu</b>	Cap
<b>Postcondicions</b>	Mostrar el catàleg complet de l'aplicació.

Figura 2. Diagrama de casos d'ús: Veure catàleg de series

### CU6. Veure episodi

En la següent *figura 3* s'estableix un flux bàsic especificant les interaccions entre client i sistema preveient un flux alternatiu amb els errors que poden passar amb les diferents accions del client. La principal dificultat amb la qual ens vam trobar en aquest cas va ser definir correctament el flux alternatiu pels diferents possibles errors que podien produir-se al flux bàsic. Per tal de solucionar-ho, es va definir per cada interacció un possible error i es va explicar i resoldre al flux alternatiu. Visualitzar *figura 3*.

<b>Descripció</b>	CU6. Veure episodi
<b>Actors</b>	Client
<b>Precondicions</b>	Haver fet login correctament a l'aplicació.
<b>Flux bàsic</b>	<ol style="list-style-type: none"> <li>1. El Client selecciona la sèrie dins del catàleg.</li> <li>2. El Sistema mostra una llista amb els capítols de la sèrie.</li> <li>3. El Client selecciona un capítol.</li> <li>4. El Sistema ensenya la informació del capítol.</li> <li>5. El Client selecciona reproduir capítol.</li> <li>6. El Sistema passa a reproduir el capítol seleccionat.</li> </ol>
<b>Flux alternatiu</b>	<p>1.a. La sèrie encara no esta disponible i el sistema dona la informació general de la sèrie. Es torna a, CU3</p> <p>5.a Error en la reproducció, el Sistema dona un missatge d'error i es torna al 5.</p>
<b>Postcondicions</b>	Es reproduceix el capítol seleccionat dins una llista.

Figura 3. Diagrama de casos d'ús: Veure episodi.

## CU7. Valorar episodi

En *figura 4* es mostra el flux bàsic per a valorar un episodi després d'haver-lo vist. La principal dificultat és novament localitzar els possibles errors que podien existir al intentar valorar un episodi. Per a resoldre-ho vam definir el flux bàsic primerament i un cop definit vam localitzar totes les fonts de possibles errors possibles i vam donar una sortida alternativa al flux. Visualitzar *figura 4*.

<b>Descripció</b>	CU7. Valorar episodi
<b>Actors</b>	Client
<b>Precondicions</b>	Haver finalitzat la visualització d'un episodi.
<b>Flux bàsic</b>	<ol style="list-style-type: none"> <li>1. El client escull l'opció de valorar l'episodi.</li> <li>2. El sistema demana l'identificador d'usuari.</li> <li>3. El client indica l'identificador d'usuari.</li> <li>4. El Sistema mostra una taula amb valors de l'1 al 5 per puntuar el capítol.</li> <li>5. El client selecciona una puntuació.</li> <li>6. El sistema guarda la puntuació indicada pel client</li> <li>7. El sistema mostra la puntuació a l'informació de l'episodi</li> </ol>
<b>Flux alternatiu</b>	<ol style="list-style-type: none"> <li>1.a El client vol valorar un episodi que no ha acabat de veure. Es mostra un error i torna al 1</li> <li>3.a L'identificador d'usuari indicat pel client no es correcte. Es torna al 2</li> </ol>
<b>Postcondicions</b>	El client entra a l'aplicació i pot veure la puntuació dels capítols valorats

Figura 4. Diagrama de casos d'ús: Valorar episodi.

## CU8. Veure episodis més valorats

En *figura 5* s'estableix un flux bàsic per veure els episodis més valorats. La major dificultat que vam trobar va ser localitzar si podia haver un flux alternatiu i definir-lo altre cop. Es veure que si no existien episodis valorats, el sistema havia de indicar-ho a l'usuari. Veure *figura 5*.

<b>Descripció</b>	CU8. Veure episodis més valorats
<b>Actors</b>	Client/Admin/Usuari
<b>Precondicions</b>	Haver fet login correctament a l'aplicació.
<b>Flux basic</b>	<ol style="list-style-type: none"> <li>1. El client/admin/usuari selecciona veure episodis més valorats</li> <li>2. El Sistema mostra una llista ordenada de més a menys amb els episodis amb més valoracions per part dels clients i la nota mitja de l'episodi.</li> </ol>
<b>Flux alternatiu</b>	2.a No hi ha episodis valorats. El Sistema mostra un missatge i es torna al 1.
<b>Postcondicions</b>	Cap

Figura 5. Diagrama de casos d'ús: Veure episodis més valorats.

#### ·Model de domini:

En aquest apartat es mostra el procés per a fer el model de domini i es comenten alguns aspectes rellevants.

Per crear el model de domini es va utilitzar un programa especialitzat en la creació de models de domini: plantuml. La principal dificultat en aquest apartat va ser entendre com funcionava aquesta eina i poder incloure totes les propietats que necessitàvem al model de domini. Veure *figura 6*.

Amb l'ajut de les indicacions al Campus Virtual vam poder utilitzar-lo correctament amb uns tests d'altres models de domini.

Un cop ja sabíem com usar el plantuml vam definir les diferents classes que es necessitaven en aquesta aplicació i vam definir les relacions que hi havia entre elles. Els majors problemes amb els que ens vam trobar va ser relacionar en el model de domini als usuaris i els episodis ja que veiem que no era correcte relacionar-los directament. Per tant

vam crear una nova classe anomenada UsuariRegistrat que ens permetia fer de pont entre els usuaris i els episodis. Un altre aspecte rellevant va ser indicar la doble relació que aquesta classe tenia amb episodis, ja que tant podies veure'ls com valorar-los; raó per la qual finalment ens vam decantar per indicar aquesta relació amb una doble connexió (una per les visualitzacions i l'altre per les valoracions). Així mateix va sorgir la qüestió de com relacionar la classe **TotSeries** i les sèries, ja que faltava una classe on incloure totes les sèries. Per fer-ho vam crear la classe catàleg, que ens va permetre relacionar a més a més l'usuari amb les series que visualitzava.

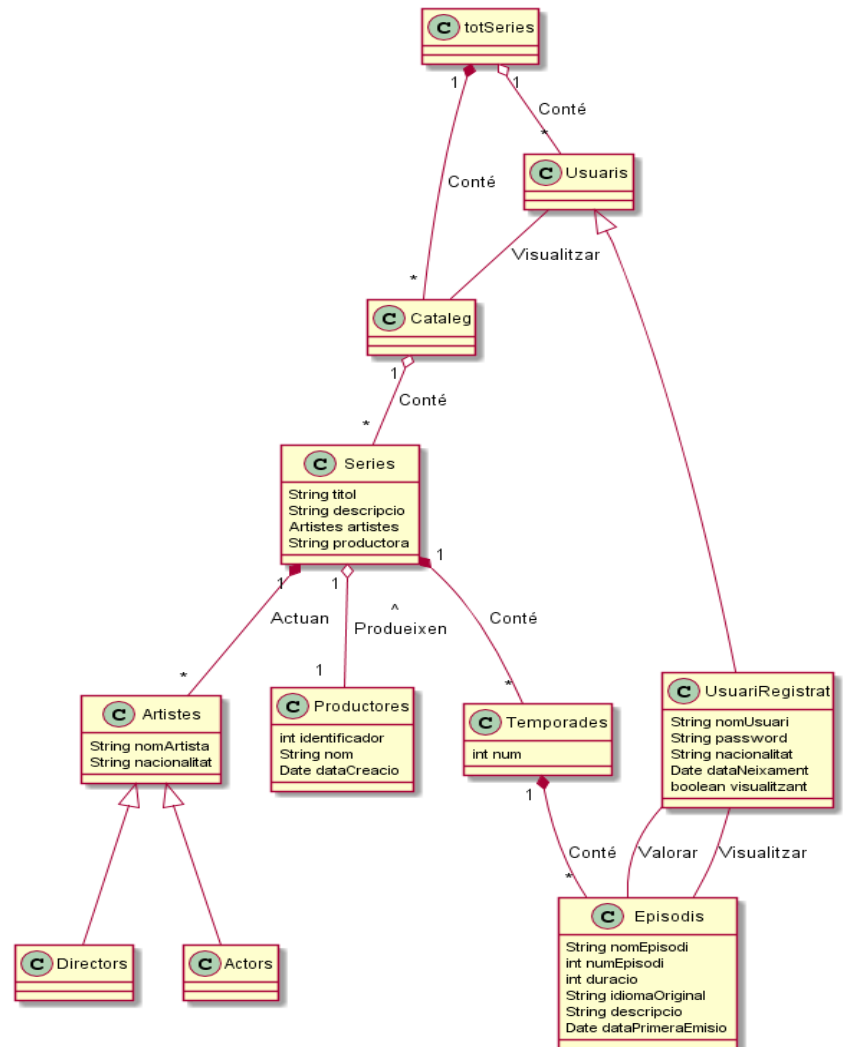


Figura 6. Model de domini de TotSeries.

L'últim pas va ser indicar els valors a banda i banda de les relacions, i localitzar els llocs on la relació fos una agregació o una composició. Per fer-ho vam usar els exemples vistos a teoria i vam poder definir les classes que eren necessàries per a que funcionés la classe amb la qual estava relaciona. Per exemple, vam veure que sense artistes no podies fer una sèrie, o que una sèrie estava composta per temporades, que a la vegada estaven compostes



per episodis. Finalment, per indicar els valors numèrics de les relacions vam fixar-nos amb com es relacionaven les diferents classes i vam anar traient poc a poc els valors indicats.

#### ·Històries d'usuari:

En aquest apartat explicarem com vam crear les històries d'usuari per visualitzar i valorar els episodis. Tant per la creació de la història d'usuari relacionada amb la visualització com la valoració vam usar la taula del campus virtual com a guia.

El primer pas va ser definir tots els criteris d'acceptació que vam trobar, relacionant cada un d'ells amb una possible opció que ens podíem trobar al recórrer el flux. A continuació, vam definir a la casella resultats què indicava l'enunciat de la pràctica respecte el funcionament de les diferents opcions del flux, per tal que la història d'usuari fos correcta.

#### CONCLUSIONS:

Aquesta primera pràctica ens ha ajudat molt a entendre i consolidar molts dels conceptes donats a classe de teoria d'aquesta assignatura. A la primera part de la pràctica hem entès què és un diagrama de casos d'ús i la utilitat que té, si es fa bé, per saber com i què has de programar abans de començar a escriure el codi d'un programa, tenint en compte els diferents fluxos que pot seguir l'aplicació i els avisos i errors que et pots trobar mentre desenvolupes el flux principal del codi. A més a més també serveix com a ajuda a l'hora de veure quines funcions es necessari introduir al teu codi i quines no.

Pel que fa al model de domini ens ha ajudat a estructurar el programa correctament, creant vincles entre les diferents classes, ajudant a construir una correcta estructura del programa. A més a més també ajuda a veure com modificar i introduir els valors necessaris pel funcionament correcte del programa, com és per exemple la classe UsuariRegistrat, una classe que conté informació dels episodis per tal que no tinguem la informació dels episodis directament vinculada amb l'usuari. Creiem que és una eina molt útil per estructurar un programa i saber quines classes són imprescindibles al teu codi abans de posar-te a programar.

Finalment, creiem que les histories d'usuari són molt útils per saber si el teu programa realment compleix totes les condicions i reacciona a tots els fluxos tal i com havien demanat que ho fes, ja que és una forma visual i pràctica de saber en cada cas com ha de reaccionar el teu programa als diferents esdeveniments en els que es pot trobar.

En conclusió, creiem que les eines usades en aquesta pràctica és necessari utilitzar-les abans de llençar-te a programar si vols que el teu programa compleixi totes les condicions requerides i a més tingui una estructura fàcil d'entendre i amb lògica.

## APÈNDIX:

**CU1:Logar-se**

En el diagrama de *figura 7* el principal problema amb el qual ens vam trobar va ser definir com havia de reaccionar el flux alternatiu als error de inici de sessió. Per controlar els dos tipus de error vam crear dos fluxos diferents, un flux que controlava com reaccionar si la contrasenya indicada era la correcta i un altre flux per indicar que l'usuari indicat no estava registrat.

<b>Descripció</b>	CU1. Logar-se
<b>Actors</b>	Client/Admin
<b>Precondicions</b>	Disposar de un client registrat al seu nom.
<b>Flux bàsic</b>	<ol style="list-style-type: none"> <li>1. El Sistema demana el nom d' usuari.</li> <li>2. El Client/Admin diu quin es el seu nom usuari.</li> <li>3. El Sistema demana el contrasenya.</li> <li>4. El Client/Admin diu quin es el seu contrasenya .</li> <li>5. El Sistema loga al Usuari.</li> </ol>
<b>Flux alternatiu</b>	<p>4.a El Client diu alguna informació incorrecta i el Sistema mostra un missatge de error. Es torna al 1.</p> <p>4.a El Client intenta accedir sense estar registrat i el Sistema mostra un missatge de error. Es torna al 1.</p>
<b>Postcondicions</b>	El client entra a la aplicació i pot veure els capítols i fer “rating”.

Figura 7. Diagrama de casos d'ús: Logar-se valorats.

**CU5. Donar de baixa client**

En *figura 8* vam definir el flux bàsic que seguiria l'administrador per poder eliminar un usuari de l'aplicació. El principal problema amb el que ens vam trobar va ser localitzar el flux alternatiu, definir-lo i donar una solució per tal que el programa pogués seguir funcionant correctament. Per solucionar-ho vam preveure que al no confirmar l'eliminació, el sistema informés que no s'havia eliminat correctament l'usuari indicat i el flux tornés a dirigir-se al pas número 1 del flux bàsic.

<b>Descripció</b>	CU5. Donar de baixa client
<b>Actors</b>	Admin
<b>Precondicions</b>	Tenir un usuari amb privilegis de administrador.
<b>Flux bàsic</b>	<ol style="list-style-type: none"><li>1. El Admin selecciona l'opció donar de baixa.</li><li>2. El Sistema ensenya un llista dels usuaris registrats.</li><li>3. El Admin selecciona un usuari de la llista.</li><li>4. El Sistema demana confirmació.</li><li>5. El Admin confirma la baixa.</li><li>6. El Sistema dona de baixa l'usuari de l'aplicació.</li></ol>
<b>Flux alternatiu</b>	5.a El Admin s'equivoca al donar de baixa un usuari i no confirma la baixa. El Sistema mostra un missatge i es torna al 1.
<b>Postcondicions</b>	El client s'elimina correctament de l'aplicació.

Figura 8. Diagrama de casos d'ús: Donar de baixa client.