

Pràctiques de Laboratori

Lliurament 2

**Enginyeria del
Programari**

ETS Enginyeria
Informàtica
DSIC – UPV

Abast del lliurament

En el lliurament 2 l'alumne presentarà el treball d'implementació desenvolupat en les últimes sessions de pràctiques sobre el cas d'estudi ***Gestió d'Esports***:

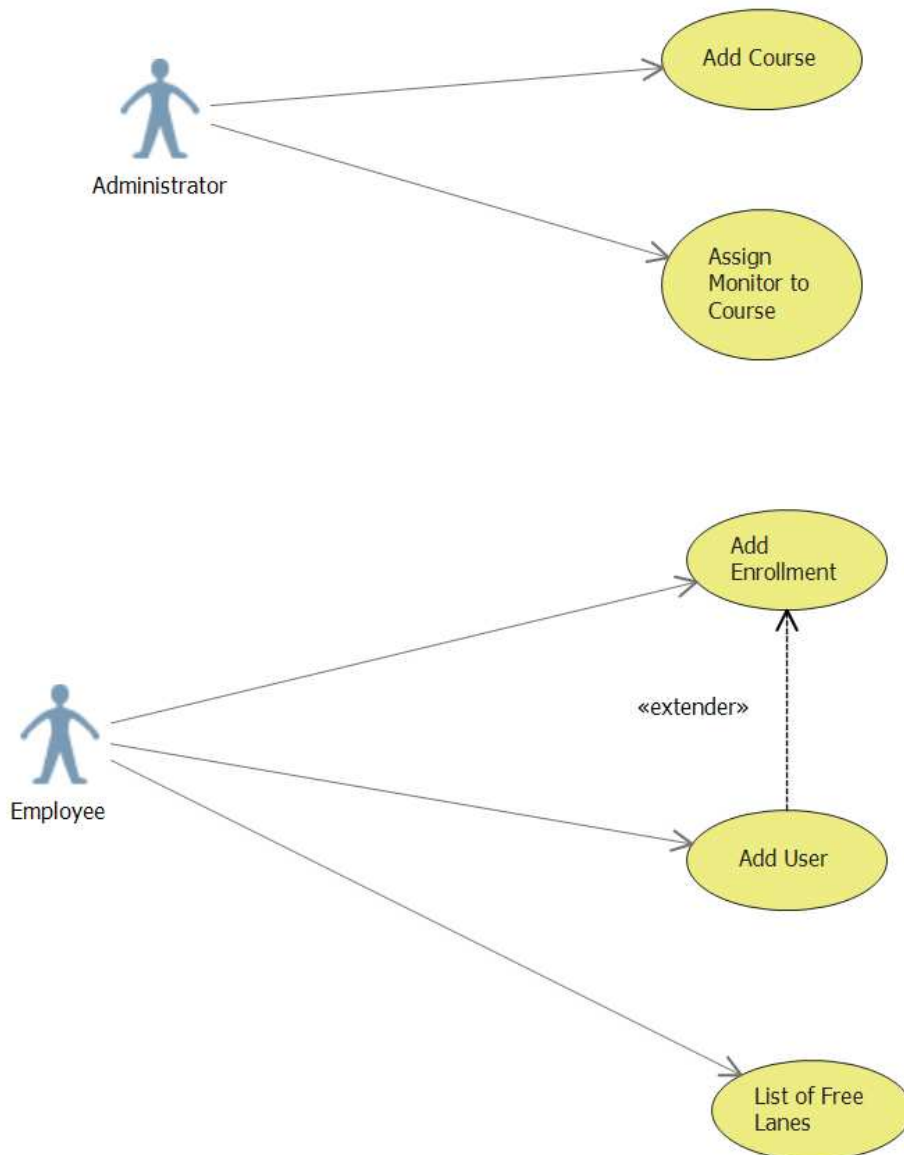
- Pràctica 3: Disseny de Classes.
- Pràctica 4: Entity Framework.

En concret, l'alumne haurà de demostrar que ha implementat:

- La capa de negoci per a totes les classes del model de disseny presentat en el butlletí de la pràctica 3.
- La implementació completa de les capes de negoci i persistència, seguint les guies dels respectius butlletins, per als següents casos d'ús (tal com es va indicar en els butlletins de la pràctica 3 i 4 i els seminaris vistos en classe de teoria):
 - **Donar d'alta un curset** (actor: Administrador)
 - **Assignar monitor a un curset** (actor: Administrador)
 - **Inscriure un usuari en un curset** (actor: Empleat)
 - **Llistat de carrers lliures** (actor: Empleat)

Descripció dels casos d'ús a desenvolupar

A continuació es detalla la funcionalitat esperada dels casos d'ús que caldrà implementar. Compte, per a aquest lliurament **solament** es demana les capes de negoci i persistència. **No es demana la capa d'interfície gràfica** que serà objecte del següent lliurament.



ID	1
Caso d'Ús	Donar d'alta un curset
Actors	Administrador
Propòsit	Afegir un curset
Resum	<ol style="list-style-type: none"> 1. L'usuari introdueix la informació específica del curset que s'impartirà (data de començament, de finalització, dies de la setmana, carrers de la piscina, preus, etc) 2. El sistema crea el nou curset.
Precond	--
Postcond	El curset es creat i emmagatzemat

Extensió síncrona	En 1, si el curset es programa en dates en les que els carrers estan ocupats, el sistema mostrarà un error i l'empleat podrà canviar les dades.
-------------------	---

ID	2
Cas d'Ús	Assignar monitor a un curset
Actors	Administrador
Propòsit	Assignar un monitor a un curset
Resum	<ol style="list-style-type: none"> 1. El sistema mostrarà tots els curset. 2. L'usuari triarà un curset. 3. El sistema mostrarà la informació detallada del curset incloent si el mateix ja te un monitor assignat. 4. El sistema mostrarà tots els monitors que poden impartir el curset tenint en compte les possibles restriccions horàries que pogueren existir. 5. L'usuari triarà el monitor que l'impartirà. 6. El sistema assignarà el monitor al curset.
Precond	--
Postcond	La assignació es crea i s'emmagatzema en el sistema.

Un monitor no pot donar un curset si en el lapse de temps que dura aquest el seu horari solapa amb altres horaris de cursets que ja te assignats el monitor. Un curset te la següent informació temporal:

- StartDate: el dia que comença la primera sessió del curset.
Exemple: "16/11/2017"
- FinishDate: el dia de l'última sessió del curset.
Exemple: "29/6/2018"
- CourseDays: els dies de la setmana que s'imparteix el curset.
Exemple: Dilluns, Dimecres i Divendres
- StartHour: hora a la que comença una sessió del curset.
Exemple: "9:30"
- Duration: temps que dura cadascuna de les sessions del curset.
Exemple: "0:45"

ID	3
Cas d'Ús	Inscriure un usuari en un curset
Actors	Empleat
Propòsit	Inscriure un usuari en algun del cursets programats
Resum	<ol style="list-style-type: none"> 1. El sistema mostra la llista de cursets amb una data de començament posterior a la actual o que s'estén impartint en l'actualitat. 2. El empleat tria un dels cursets.

	3. El sistema sol·licita el DNI del usuari. 4. El empleat introdueix el DNI. 5. El sistema busca el usuari per DNI i mostra les seues dades. Calcula el preu de la primera quota i la mostra. 6. El empleat confirma la inscripció del usuari en el curset. 7. El sistema emmagatzema la inscripció i genera el primer pagament.
Precond	-
Postcond	Una nova inscripció queda registrada en el sistema.
Extensió síncrona	En 5, si no es troba l'usuari, es preguntarà si vol donar d'alta un nou usuari i sollicitarà les dades del mateix. En 5, si el usuari ja està assignat a eixe curset el sistema informa del error.

ID	4
Cas d'Ús	Llistat de carrers lliures
Actors	Empleat
Propòsit	Proporcionar informació dels carrers que queden lliures una setmana concreta
Resum	1. El sistema mostra les piscines que hi ha. 2. El empleat tria una de les piscines. 3. El sistema demana la data del dilluns de la setmana per a la que va a fer el càlcul. 4. El sistema calcula els carrers lliures que hi haurà en cada franja horària* generant una taula com la que es mostra a continuació.
Precond	-
Postcond	
Extensió síncrona	

* Franja horària: Trams de 45 minuts que comencen amb l'hora d'apertura de la piscina.

Exemple del llistat:

Piscina 1 - Setmana del 18/12/2017

	Dilluns	Dimarts	Dimecres	Dijous	Divendres	Dissabte
08:00-08:45	6	6	6	6	6	6

08:45-09:30	6	6	6	6	6	6
09:30-10:15	3	6	3	6	3	2
10:15-11:00	0	3	0	3	0	2
11:00-11:45	6	6	6	6	6	2
11:45-12:30	6	6	6	6	6	6
12:30-13:15	6	6	6	6	6	6
13:15-14:00	6	6	6	6	6	6
...

Detalls de la capa lògica

En les sessions de pràctica anteriors s'ha construït la part de la funcionalitat que té a veure amb el disseny de les classes i el seu emmagatzematge persistent.

En aquesta pràctica, es deu d'implementar el controlador (o proveïdor de serveis) de la lògica de negoci. Aquest controlador haurà d'oferir a les capes superiors (IU) accés a tota la funcionalitat que esta necessite.

El conjunt de tots els serveis oferts per la IGU deu d'estar definit en una interfície anomenada *IGestDepService*. Els mètodes d'aquesta interfície seran implementats per una classe anomenada *GestDepService*. La interfície i la seua implementació deuen trobar-se a la carpeta *BusinessLogic/Services* i formar part del espai de noms *GestDepLib.Services*.

Tots els errors que puguin sorgir en l'execució dels mètodes implementats s'hauran de reportar mitjançant excepcions.

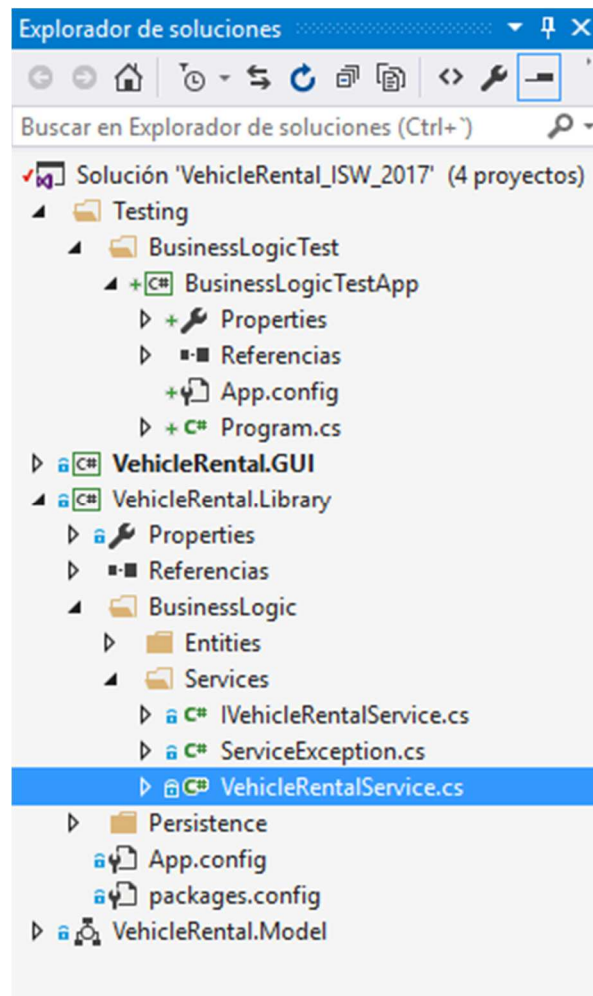
Per a reportar els errors generats en la capa lògica, en la carpeta *BusinessLogic/Services* s'haurà de crear una subclasse de *Exception* anomenada *ServiceException*. Quan es creen objectes d'aquesta classe per a informar d'algun error, la propietat *Message* d'aquesta classe contindrà el missatge de text amb l'error generat.

Per exemple, en el cas d'estudi d'exemple que hi ha en PoliformaT anomenat *VehicleRental_ISW_2017* quan es vol afegir una persona seria un error afegir-la si ja existeix una amb el mateix dni que la que es vol crear. A continuació, es pot veure un tros de codi del controlador en el qual abans d'afegir una persona es comprova que no existisca una amb el mateix dni. En cas contrari, es llança una excepció.

```
public void addPerson(Person person)
{
    if (dal.GetById<Person>(person.Dni) == null)
    {
        dal.Insert<Person>(person);
        dal.Commit();
    }
    else throw new ServiceException("Person already exists.");
}
```

Amb l'objectiu de facilitar la identificació dels mètodes a implementar en el controlador, en PoliformaT es podrà descarregar un projecte que conté un programa de test de la lògica. Aquest programa assumirà que existeix un controlador que implementa certs mètodes i els provarà. Els alumnes han d'afegir a la carpeta *Testing* del seu projecte una nova carpeta de solució anomenada *BusinessLogicTest*, en la qual hauran d'agregar el projecte de consola que es descarregaran de PoliformaT anomenat *BusinessLogicTestApp*. Aquest programa conté les proves que ha de superar el *GestDepService* implementat.

En la següent figura es pot veure l'aparença que té explorador de solucions per a l'exemple *VehicleRental_ISW_2017* amb els continguts creats en la sessió. Destaca la interfície *IVehicleRentalService* la seua implementació anomenada *VehicleRentalService*, l'excepció *ServiceException* i el programa de test *BusinessLogicTestApp*.



Data de lliurament

La data de lliurament serà la corresponent a la 8a setmana de pràctiques en els laboratoris. TOTS els membres del grup deuen acudir al lliurament.

Forma de lliurament

S'obrirà una tasca en PoliformaT en la qual cada grup haurà de pujar el fitxer comprimit del projecte. També es realitzarà una operació de protegir sobre el repositori de Team Services amb el comentari "Lliurament 2". El projecte ha de pujar-se a la tasca **ABANS** del començament de la sessió d'avaluació.

Avaluació

En la sessió de laboratori, el professor avaluarà a cadascun dels grups comprovant que s'ha implementat correctament la funcionalitat sol·licitada i fent preguntes a TOTS els membres del grup. S'utilitzarà la rúbrica d'avaluació adjunta per a valorar la funcionalitat implementada.