

Lliurament 2

1. Objectius

Organitzar imatges en una biblioteca i poder crear diferents àlbums amb les imatges de la biblioteca. En el lliurament 1 heu implementat les classes **TaulaImatges** i **LlistaImatges** que permeten gestionar un conjunt d'imatges. La classe **TaulaImatges** era un exercici introductori, que permetia aprendre a utilitzar l'entorn de programació i repassar els coneixements de Programació 1. A partir d'aquest lliurament, utilitzarem només la classe **LlistaImatges**. A partir d'aquesta classe, definirem una classe **BibliotecaImatges** que serà una llista d'imatges amb les següents propietats:

- No permetrà que hi hagin dues imatges iguals
 - Entenem que dues imatges són iguals si el camí on es troben, el seu nom de fitxer i la seva extensió són iguals.
- Quan s'afegeix una imatge a la biblioteca, es verificarà que el fitxer associat a la imatge existeix.
- No hi haurà cap límit en el nombre d'imatges

A més a més, definirem una classe **AlbumImatges** que serà una llista d'imatges amb les següents propietats:

- Permet imatges duplicades
- Permetrà afegir imatges que no existeixen en el disc.
- Estarà limitat a un màxim d' N imatges, on N serà un valor que per defecte serà 10, però que es podrà canviar.
- Cada àlbum portarà associat:
 - Un títol
 - Una posició GPS de referència (longitud i latitud)
 - La imatge a mostrar com a representació de l'àlbum.
 - En cas que no hi hagi cap imatge a l'àlbum, serà null.
 - Per defecte serà la primera imatge afegida a l'àlbum.
 - Es podrà seleccionar la imatge que es desitgi, però ha d'estar a l'àlbum.

Programació 2. Projecte de Pràctiques

Grau d'Enginyeria Informàtica. Facultat de Matemàtiques. UB
Curs 2012-2013. Professorat: Laura Igual, Xavier Baró i Santi Seguí

L'aplicació podrà tenir una i només una biblioteca però un nombre indeterminat d'àlbums. Els àlbums quedaran identificats pel seu títol, i no podrem tenir-ne de repetits.

Per tal d'interactuar amb la llista, caldrà definir un menú amb les opcions següents, aprofitant tot el possible del lliurament 1:

1. Gestió Biblioteca: Dona accés a un menú per a la gestió de la biblioteca.
 - 1.1. Afegir Imatge: Demanarà les dades d'un fitxer d'imatge i l'afegirà a la biblioteca.
 - 1.2. Mostrar Biblioteca: Mostra el contingut de la biblioteca, mostrant davant de cada fitxer, el número de la seva posició a la llista començant per 1.
 - 1.3. Eliminar Imatge: Elimina de la biblioteca la imatge corresponent a una posició donada.
 - 1.4. Visualitzar Imatge: Permet visualitzar una imatge de la biblioteca.
 - 1.5. Menú Anterior: Tornarà al menú principal
2. Gestió Àlbums: Dona accés a un menú per a la gestió dels àlbums, que conté les següents opcions:
 - 2.1 Afegir Àlbum: Demana les dades d'un nou àlbum i l'afegeix a l'aplicació.
 - 2.2 Mostrar Àlbums: Mostra els àlbums disponibles a l'aplicació, mostrant davant de cada àlbum, el número de la seva posició a la llista començant per 1.
 - 2.3 Eliminar Àlbum: Elimina de l'aplicació l'àlbum corresponent a una posició donada.
 - 2.4 Gestionar Àlbum: Permet seleccionar un àlbum dels disponibles a l'aplicació, i obre un menú per a gestionar-lo. Aquest menú tindrà les següents opcions:
 - 2.4.1 Afegir Imatge: Mostra les imatges de la biblioteca i permet seleccionar-ne una, que passarà a formar part de l'àlbum.
 - 2.4.2 Mostrar àlbum: Mostra el contingut de l'àlbum, mostrant davant de cada imatge, el número de la seva posició a la llista començant per 1.
 - 2.4.3 Eliminar Imatge: Mostra les imatges de l'àlbum i permet seleccionar-ne una, que serà eliminada de l'àlbum.
 - 2.4.4 Modificar Dades: Permet modificar els atributs de l'àlbum (nom, posició i imatge de portada)
 - 2.4.5 Visualitzar Imatge: Permet visualitzar una imatge de l'àlbum.
 - 2.4.6 Menú Anterior: Tornarà al menú anterior.
 - 2.5 Menú Anterior: Tornarà al menú principal.
3. Guardar Dades: Guarda les dades de l'aplicació a un fitxer.
4. Recuperar Dades: Carrega les dades de l'aplicació prèviament guardades d'un fitxer.
5. Sortir: Surt de l'aplicació.

Programació 2. Projecte de Pràctiques

Grau d'Enginyeria Informàtica. Facultat de Matemàtiques. UB
Curs 2012-2013. Professorat: Laura Igual, Xavier Baró i Santi Seguí

Cal que tingueu en compte que quan una imatge s'elimina de la biblioteca, també cal eliminar-la dels àlbums que la contenen.

Seguint la organització Model-Vista-Controlador (MVC), definirem una classe **DadesVisor** que contingui totes les dades de l'aplicació. Totes les accions que afectin a les dades s'hauran de fer a través d'aquesta classe. També crearem la classe **CtrlVisor** que implementi el controlador de la nostra aplicació.

A més a més, volem evitar que la vista utilitzi directament classes del model, per delegar aquesta tasca al controlador. Per tant, la vista només podrà utilitzar el controlador, i aquest s'encarregarà de fer les crides oportunes a la classe del model, la qual utilitzarà les classes que guarden les dades.



3 Descripció del lliurament

A continuació us anirem plantejant els diferents passos per resoldre la pràctica proposada. Us recomanem que seguiu aquests passos.

3.1 Creació del projecte

El primer pas serà crear un projecte, al qual li heu de posar com a nom **Cognom1Cognom2Nom**, tenint en compte les següents consideracions:

- La primera lletra de cada part en majúscula i la resta en minúscula.
- Eviteu utilitzar accents i caràcters estranys (ñ o ç).
- La classe principal s'ha de dir **VisorUB2**, i el paquet per defecte **edu.ub.prog2.Cognom1Cognom2Nom.vista**. En el nom del paquet utilitzeu els mateixos criteris anteriors.

Per exemple, una estudiant amb nom Dolça Martínez Castaña, hauria de crear un projecte amb el nom *MartínezCastanaDolça*, i la seva classe principal hauria de ser *edu.ub.prog2.MartínezCastanaDolça.vista.VisorUB2*.

3.2 Implementació del menú d'opcions i la lògica del programa

Seguint l'exemple del lliurament 1, creeu la lògica del programa, utilitzant la classe **Menu** de la llibreria i mostrant un missatge per a cada opció. Comproveu especialment que podeu passar d'un menú a un altre correctament i que l'aplicació finalitza correctament.

Programació 2. Projecte de Pràctiques

Grau d'Enginyeria Informàtica. Facultat de Matemàtiques. UB
Curs 2012-2013. Professorat: Laura Igual, Xavier Baró i Santi Seguí

3.3 Creació de les classes principals de l'aplicació

Tal com s'indica a la descripció del lliurament, en aquest lliurament començarem a utilitzar el disseny MVC. Per facilitar la separació dels diferents components, utilitzarem una classe principal per a cada component. Pel que fa a la Vista, ja heu definit la classe **VisorUB2**. Ara crearem les classes per al model i per al controlador.

Crea la classe **DadesVisor** que contindrà totes les dades de l'aplicació. De moment pots deixar la classe totalment en blanc. Quan comencem a implementar les opcions del menú ja aniràs afegint els mètodes que necessitis. Igual que totes les classes relacionades amb el model, aquesta classe haurà d'estar dins el [paquet Model] (**edu.ub.prog2.Congom1Cognom2Nom.model**).

Ara crea la classe **CtrlVisor**, que exercirà de controlador de l'aplicació, controlant els diferents processos que es duguin a terme en el visor. Aquesta classe rebrà les indicacions de l'usuari per mitjà de la vista i farà les accions pertinents, comunicant-se amb el model quan sigui necessari. Igual que totes les classes que tinguin a veure amb el control de l'aplicació, aquesta classe anirà dins del [paquet Controlador] (**edu.ub.prog2.Congom1Cognom2Nom.controlador**). Tot i que per aquest lliurament encara no ho necessiteu, per poder resoldre el següent lliurament, caldrà que el vostre controlador hereti les funcionalitats d'un controlador bàsic que us hem proporcionat a la llibreria. Per fer això, definiu aquesta classe com:

```
public class CtrlVisor extends BasicCtrl {  
  
}
```

Un cop definides les classes principals, caldrà que defineixis els atributs necessaris per connectar aquestes classes, tenint en compte que la vista només pot accedir al model a través del controlador.

3.4 Gestió de la Biblioteca

A partir de les indicacions de l'enunciat, defineix la classe **Bibliotecalmatges** que permet gestionar les imatges de la biblioteca.

Crea tots els mètodes necessaris per fer la gestió de les opcions 1.1, 1.2 i 1.3, tenint en compte que equivalen a les opcions 1, 2 i 3 del lliurament 1, i per tant, tot i que cal reestructurar el projecte seguint el model MVC, has de poder reutilitzar gran part del codi. Recorda que disposes de la classe **Llistalmatges** del lliurament 1.

Per comprovar si un fitxer existeix o no, teniu el mètode **exists** de la classe **File**.

Programació 2. Projecte de Pràctiques

Grau d'Enginyeria Informàtica. Facultat de Matemàtiques. UB
Curs 2012-2013. Professorat: Laura Igual, Xavier Baró i Santi Seguí

3.5 Visualitzar una imatge

El fet de que en el lliurament 1, la classe **Imatge** heretés de la classe **ImageFile**, era per permetre que poguéssiu utilitzar els mètodes de visualització. Per tal de visualitzar una imatge, podeu utilitzar el mètode **show** per visualitzar qualsevol imatge.

Implementa la opció 1.4, de manera que permeti seleccionar una imatge de la biblioteca i la visualitzi. Un cop seleccionada una imatge, executeu-ne el mètode **show** passant un valor **true** com a paràmetre per tal que la execució quedi parada mentre no tanqueu la imatge.

3.6 Gestió dels àlbums

A partir de les indicacions de l'enunciat, defineix la classe **AlbumImatges** que permet gestionar les imatges d'un àlbum. Prenent com a exemple la gestió de la biblioteca, implementa tots els mètodes necessaris per fer la gestió dels àlbums.

3.7 Persistència de dades

Igual que al lliurament 1, volem poder guardar totes les dades de l'aplicació. En aquest cas, ja no tenim una llista d'imatges, sinó que tenim una biblioteca i uns àlbums. Seguint la persistència del lliurament 1, implementa les opcions 3 i 4 del menú principal. Recorda que les dades de l'aplicació es troben al model.

4 Material pel lliurament

Per aquest lliurament us proporcionem una llibreria **UtilsProg2.jar** que conté les classes:

- **ImageFile**
- **ImageList**
- **BasicCtrl**
- **Menu**

Haureu d'utilitzar aquestes classes en el desenvolupament del lliurament. Podeu trobar aquesta llibreria al Campus Virtual i afegir-la al vostre projecte.

Programació 2. Projecte de Pràctiques

Grau d'Enginyeria Informàtica. Facultat de Matemàtiques. UB
Curs 2012-2013. Professorat: Laura Igual, Xavier Baró i Santi Seguí

5 Ajuda pel lliurament

Utilització de la llibreria **UtilsProg2.jar**:

- Podeu instanciar objectes de la classe **Menu** a la vostra classe vista per implementar la gestió del menú del visor. Teniu un exemple de com utilitzar la classe **Menu** al Campus Virtual.
- Les classes **ImageFile** i **ImageList** són classes abstractes que heu d'utilitzar per implementar les vostres classes **Imatge**, **LlistatImatges** i **TaulaImatges**.
- La classe **BasciCtrl** és una classe capaç de respondre davant de certs events. Us servirà per realitzar tasques automàtiques al controlador, i per tant cal que la classe **CtrlVisor** hereti d'aquesta classe.

6 Format del lliurament

El lliurament consistirà en tot el codi generat en els diferents punts de l'enunciat, juntament amb la documentació especificada en aquest apartat.

En concret, cal generar un fitxer amb el nom:

Cognom1Cognom2Nom_L2.tar.gz

Que conté:

- el directori "src"
- el fitxer build.xml
- i un fitxer "Cognom1Cognom2Nom_L2.jar" amb els .class

Tot el codi generat ha d'estar correctament comentat per a poder executar el JavaDoc, generant automàticament la documentació en línia del codi.

També cal lliurar una **memòria** amb els següent punts (a més dels descrits en la normativa de pràctiques)

- Introducció al problema tractat en el lliurament (No s'acceptarà una còpia directa de l'enunciat).
- Explicació de les classes implementades per cadascuna de les parts del lliurament.
- Explicar què has pogut reutilitzar del lliurament anterior i quins són els principals canvis.
- Proves realitzades per comprovar el correcte funcionament de la pràctica, resultats obtinguts i accions derivades.
- Observacions generals.