



Enginyeria del Programari – PAC1

Presentació

Aquesta PAC és una introducció a l'Enginyeria del Programari i al paradigma d'Orientació a Objectes. L'activitat cobreix els continguts estudiats als mòduls 1 i 2 de l'assignatura.

Competències

En aquesta PAC es treballen les següents competències del Grau en Enginyeria Informàtica:

- Saber aplicar les tècniques específiques d'enginyeria del programari a les diferents etapes del cicle de vida d'un projecte.
- Saber aplicar les tècniques específiques de tractament, emmagatzematge i administració de dades.

També, es treballen les següents competències del Grau en Multimèdia:

- Capacitat d'organitzar i gestionar la informació utilitzant tecnologies de bases de dades, llenguatges i models estàndards.
- Capacitat per implementar, posar en marxa i mantenir programari i maquinari multimèdia.

Objectius

Els objectius concrets d'aquesta PAC són:

- Ser capaç d'explicar què és l'enginyeria del programari i contextualitzar-la.
- Ser capaç d'enumerar alguns dels mètodes de desenvolupament més utilitzats.
- Ser capaç d'explicar els conceptes fonamentals de l'orientació a objectes.

Descripció de la PAC a realitzar

Preguntes del Mòdul 1

Pregunta 1 (15%)

Volem que feu un exercici on reflexioneu sobre l'activitat de la “modelització”, el resultat de la qual, generalment, és una sèrie de models del sistema que faciliten la comprensió dels requisits i el disseny del sistema. Indica per cada un dels rols següents, quin paper juga en l'esmentada activitat i el motiu principal pel qual és rellevant per a la mateixa o no ho és.

Rols: Expert del domini, Analista funcional, Expert en desplegament.



Solució

- Expert del domini => És rellevant per a la modelització dels requisits. Tant podria ser que creés algun model com que validés la correctesa dels models generats pels analistes funcionals.
- Analista funcional => És el principal productor de models de requisits.
- Encarregat del desplegament => L'encarregat del desplegament pot crear models que ajudin a entendre l'estructura física del sistema com a part de la seva feina. Per tant, també és un rol rellevant per a l'activitat de modelització encara que els seus models siguin molt diferents als models que fa l'analista funcional.



Pregunta 2 (20%)

Suposem que estem desenvolupant una aplicació de consulta de les obres d'un museu. Suposem també que hem identificat una sèrie de tasques que s'han de realitzar per implementar aquesta aplicació.

Indica a quina activitat de l'enginyeria de programari correspondria cadascuna de les tasques:

1. Escriure el codi que implementa les consultes a la base de dades
[Implementació](#)
2. Fer una estimació del cost del projecte per decidir si es fa aquest any o no
[Planificació i gestió del projecte](#)
3. Decidir quina informació es mostra de cada obra
[Requisits](#)
4. Parlar amb els museus per veure quan és millor fer la càrrega inicial de dades
[Planificació i gestió del projecte](#)



Pregunta 3 (15%)

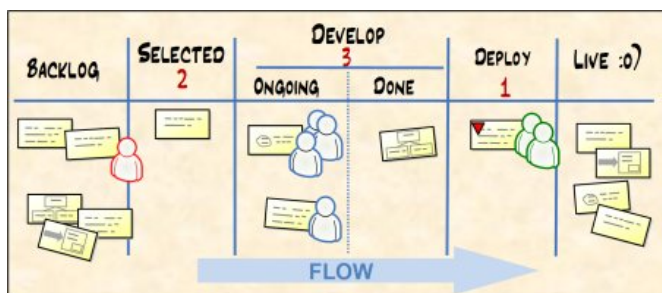
Kanban és una aproximació minimalista al desenvolupament de programari que es basa en tres principis bàsics: Visualització del “workflow”, limitació del “work in progress” i mesura del “lead time”

Feu una cerca a internet i documenteu-vos al voltant d'aquest mètode de desenvolupament de manera que pugueu fer un breu resum de cadascun d'aquests 3 principis on s'expliqui en què consisteixen, com s'implementen en la pràctica i quin és el principal benefici que se n'obté de cadascun d'ells.

Solució

Visualització del workflow fa referència a la representació, de manera visual, de les diferents etapes per les quals va passant una unitat de treball. En el cas del programari, aquesta unitat de treball pot ser una funcionalitat nova, un defecte que cal corregir, etc.

Típicament, la visualització del workflow es fa mitjançant un panell on es situen, en columnes, les diferents etapes i on cada unitat de treball és representada per una tarjeta:



D'aquesta manera, podem veure, “visualment” en quin punt del procés està cada unitat de treball. El principal benefici que s'obté de la visualització del workflow és fer més evidents els problemes en presentar-se la informació de manera visual.

Limitació del “work in progress” consisteix en assignar, a cadascuna de les etapes del procés (columnes del panell) un nombre màxim d'unitats de treball de manera que, quan es supera aquest límit, detectem que hi ha un problema en algun punt del procés. En aquest punt s'ha d'estudiar i analitzar el problema i aprofitar per introduir alguna millora al procés per tal d'evitar que es torni a repetir la situació.

El principal benefici de la limitació de la feina “en progrés” és que els problemes es fan evidents molt més aviat i, per tant, s'identifiquen abans les oportunitats de millora.

El **“lead time”** és el temps total necessari per completar una unitat de treball; des de que entra al procés fins que està disponible per als usuaris. Una altra mesura relacionada és el “cycle time” que mesura el temps que passa des de que es comença a treballar en la unitat de treball fins que està disponible per als usuaris (i, per tant, no compta el temps que s'està “esperant” a ser atesa).



El principal benefici de mesurar el “lead time” és que ens dóna una mesura objectiva de la capacitat de resposta de l'equip de desenvolupament a les necessitats dels stakeholders i els usuaris. Això també ens servirà per mesurar, per exemple, si una acció de millora al procés ha sigut realment efectiva.

Preguntes del Mòdul 2

Pregunta 4 (10%)

Poseu un exemple del món real, de coneixement comú, d'un concepte que il·lustri una associació, amb els seus rols. Cal que sigui un exemple nou, que no aparegui als materials. Justifiqueu la vostra resposta.

Solució

En podem trobar molts exemples. Un exemple clàssic seria la relació entre una obra d'art i els seus autors. Una obra està associada amb un o més artistes que juguen el rol “autor” a l'associació. Un artista està associat a una o més obres; en aquest cas, no tenim una paraula especial per al rol que juga l'obra i, per tant, diríem que juga el rol “obra”.

**Pregunta 5 (15%)**

Suposem un sistema orientat a objectes amb les classes *Llibre* i *Lector*. La classe *Llibre* té un atribut protegit *isbn*, un atribut privat *dataCreacio* i un atribut públic *titol*.

Per cada supòsit (tres en total), respon de manera justificada cada pregunta. Cada supòsit aplica a la situació inicial descrita en el paràgraf anterior.

1er supòsit: Creem una subclasse de *Llibre* amb nom *Novel·laHistòrica*.

1.1 Una instància de la classe *Novel·laHistòrica* té l'atribut *dataCreacio*?

Si. És una subclasse de *Llibre* i, per tant, hereta l'atribut tot i la visibilitat privada.

1.2 En cas que no el tingui, quins canvis caldria aplicar a *Llibre* i/o a *Novel·laHistorica* per tal que les dues tinguin l'atribut *dataCreacio*?

No cal fer cap canvi.

2on supòsit: Definim una operació a la classe *Lector*, de nom *llegir()*.

2.1 Des d'aquesta operació, podem accedir a l'atribut *isbn*?

No, donat que és un atribut protegit i, per tant, només és visible a les operacions definides a la classe *Llibre* o a alguna de les seves subclasses.

2.2 En cas que no es pugui, quins canvis caldria aplicar a *Llibre* i/o a *Novel·laHistòrica* per tal l'operació *llegir* tingués accés a l'atribut *isbn*?

Caldria fer que l'atribut fos públic.

3er supòsit: Creem una subclasse de *Llibre* amb nom *Novel·laHistòrica* amb un atribut privat *època*.

3.1 Una instància de la classe *Llibre* té l'atribut *època*?

Depèn. Una instància de *Llibre* que també ho sigui de *Novel·laHistòrica* sí que el té, però una instància de *Llibre* que no sigui instància de *Novel·laHistòrica* no el tindrà.

3.2 Si no el té, quins canvis caldria aplicar perquè el tingués?

L'única manera de que totes les instàncies de *Llibre* tinguin l'atribut *època* seria definint aquest atribut a la classe *Llibre* en comptes de *Novel·laHistòrica*.



Pregunta 6 (25%)

Suposeu que volem modelar, usant orientació a objectes, part de la informació necessària per a gestionar un aparcament municipal. D'aquesta manera tenim informació sobre les places:

- *A la primera planta de l'aparcament del carrer Bailèn (ubicat al carrer Bailèn número 42 de Barcelona) hi ha les places amb número del 101 al 143*
- *A la segona planta del mateix aparcament hi ha les places 201 a 240*
- *L'aparcament del carrer Balmes (ubicat al carrer Balmes número 34 de Barcelona) només té una planta amb places 101 a 190*

Volem saber quantes places lliures hi ha a cada planta. Per a fer-ho, necessitem saber, per cada plaça, si està o no ocupada i, en cas afirmatiu, quin és el vehicle que l'ocupa. Del vehicle volem saber, també, quantes vegades ha aparcat a un dels aparcaments gestionats pel nostre sistema

- *La plaça 101 de l'aparcament del carrer Bailèn està ocupada per un vehicle amb matrícula 1234-JIF que ha aparcat 48 vegades en un dels aparcaments gestionats pel nostre sistema.*
- *En canvi, la plaça 102 del mateix aparcament no està ocupada per cap vehicle*

Algunes places estan reservades per a un vehicle en particular durant un temps (una plaça només pot tenir una reserva enregistrada al sistema i, per tant, no cal conèixer les reserves en el passat).

- *La plaça 108 de l'aparcament del carrer Balmes està reservada per al vehicle amb matrícula 3921-HHX fins el 4 d'agost del 2014.*

Es demana:

- (10%) Feu una llista de totes les classes que pugueu identificar. Per a cada classe, feu una llista dels seus atributs. Per a aquells atributs que siguin opcionals i/o multiavaluats i/o derivats, indiqueu-ho al costat del seu nom.

Solució

- **Aparcament**
 - nom
 - adreça
- **Planta**
 - número
 - placesLliures (derivat)
- **Plaça**
 - Número
- **PlaçaReservada**



- dataFinalReserva
- Vehicle
 - matrícula
 - numVegadesAparcat

Nota:

Els atributs derivats no comptabilitzaran en la solució de l'exercici perquè no s'expliquen ens els mòduls implicats en aquesta PAC. Malgrat això, hem volgut citar-los i així proporcionar-vos un primer exemple d'aquest tipus d'informació.

- b) (5%) Identifiqueu les relacions d'herència entre classes.

Solució

PlaçaReservada és una subclasse de Plaça.

- c) (10%) Feu una llista de totes les associacions que identifiqueu. Per a cada associació indiqueu, si fa falta, els noms de rol de cada classe que participa.

Solució

- ocupa: Entre Vehicle i Plaça per indicar quin vehicle ocupa cada plaça
- reserva: Entre Vehicle i PlaçaReservada per indicar per a quin vehicle està reservada la plaça
- pertany: Entre Planta i Plaça per indicar a quina planta pertany cada plaça
- pertany: Entre Planta i Aparcament per indicar a quin aparcament pertany una planta

Nota:

- Es podria haver resolt l'exercici sense la classe Planta si no fos per l'atribut derivat placesLliures.
- Es podria haver resolt l'exercici sense la subclasse PlaçaReservada (fent, per exemple, que l'associació amb Vehicle fos opcional) però el fet que hi hagi dues unitats d'informació (el vehicle i la data) desaconsellen aquesta solució (l'atribut dataFinalReserva només podria tenir valor si l'associació reserva també té valor).

Recursos

Recursos Bàsics

- Mòdul didàctic 1: Introducció a l'Enginyeria del Programari
- Mòdul didàctic 2: Orientació a Objectes

Criteris d'avaluació



- La PAC s'ha de resoldre **de forma individual**. En cas de detectar còpies (siguin o no entre membres d'un mateix grup de debat) es penalitzarà l'activitat amb una D com a nota.
- El pes de cada pregunta està indicat dins l'enunciat.
- És necessari justificar la resposta a cadascuna de les preguntes. Es valorarà tant la correctesa de la resposta com la justificació donada.

Format i data de lliurament

Cal lliurar un únic document PDF amb les respostes a tots els exercicis. Aquest document s'ha de lliurar a l'espai de *Lliurament i Registre d'AC* de l'aula abans de les **23:59 hores del dia 14 d'octubre de 2014**. No s'acceptaran lliuraments fora de termini.