

# PROGRAMACIÓ II - PRACTICA 3

Toni Garcia i Javier Pedragosa



Universitat de Barcelona  
1r curs

## Índex

<b>INTRODUCCIÓ .....</b>	<b>2</b>
--------------------------	----------

<b>PREGUNTES DEL ENUNCIAT .....</b>	<b>2</b>
-------------------------------------	----------

<i>Pregunta 1 .....</i>	<i>2</i>
-------------------------	----------

<i>Pregunta 2 .....</i>	<i>2</i>
-------------------------	----------

<i>Pregunta 3 .....</i>	<i>3</i>
-------------------------	----------

<i>Pregunta 4 .....</i>	<i>3</i>
-------------------------	----------

<i>Pregunta 5 .....</i>	<i>4</i>
-------------------------	----------

## 1. Introducció.

Aquesta pràctica era bastant semblant a l'anterior en quant a relació entre classes, interfícies aplicades, etc.

Les diferències substancials que hi hem trobat a la part 1 ha sigut que hem hagut d'implementar un càlcul de temps per calcular si s'havia enviat una comanda en funció de quan s'havia realitzat i el temps que es trigava en enviar-la.

A la part 2 hem hagut de dissenyar per primera vegada una interfície gràfica per a la interacció amb l'usuari.

Aquesta part no ha sigut especialment complicada, però és bastant entretinguda ja que el programador ha de posar-se en la pell de l'usuari per a saber si és suficientment intuïtiva la interfície que ha dissenyat.

## 2. Resposta a les preguntes del enunciat.

### **1. Expliqueu quines classes has pogut reutilitzar de la pràctica 3 part 1 per a fer aquest. Quins canvis sobre les classes reutilitzades has necessitat fer i perquè.**

Com es pot veure, si es comparen les dues parts del projecte, l'únic que canvia es l'apartat vista.

Tota la resta de classes no depenen de la vista i per tant les hem pogut reutilitzar. Si que hem hagut de adaptar-ne algunes classes un cop passada la vista a GUI per que ens ha permès identificar alguns errors o hi havia mètodes que ens semblava que seria més còmode canviar-los, però els canvis fets s'han pogut implementar igualment a l'altra part del projecte on la vista estava feta per consola.

### **2. Indiqueu quins tipus d'esdeveniments heu fet servir al vostre codi.**

La part gràfica del projecte està programada per esdeveniments. És un tipus de programació que permet executar una part del codi quan es produeix una interacció concreta amb l'usuari. Això fa que el flux del programa estigui marcat per el que faci el usuari (o el propi sistema).

Els esdeveniments que ens han sigut útils a la hora de programar la interfície gràfica han sigut:

- Els clics als botons
- El keyReleased per els quadres de text, ja que volíem permetre activar alguns botons només si tots els camps requerits estan emplenats.
- El mouseReleased als comboBox ja per a saber també si hem triat una opció.
- El focusGained també als comboBox per emplenar les opcions quan hi fem clic per a triar-ne una.
- El mouseReleased a la llista de comandes per a habilitar el botó d'esborrar la comanda triada.

**3. *Explica quins canvis hauríeu de realitzar en l'aplicació si es volgués afegir la funcionalitat següent: Al iniciar l'aplicació s'ha d'obrir una subfinestra amb la llista de comandes pendents d'enviar.***

Veiem dues maneres de fer-ho.

La més senzilla és fer servir un JOptionPane on s'imprimeixi la informació i tingui un botó acceptar per tancar la finestra. Aquest el podem crear des de l'esdeveniment FocusGained de la finestra principal o bé crear un mètode cridat des del constructor de la mateixa.

L'altra és dissenyar un nou JFrame i cridar-lo també des del constructor de la finestra principal.

Evidentment, s'hauria d'importar la classe que emmagatzema la llista de comandes al formulari des d'on es volguessin mostrar i també la classe Comanda per poder accedir al mètode toString.

**4. *Proves realitzades per comprovar el correcte funcionament de la pràctica, resultats obtinguts i accions derivades.***

El primer que volíem assegurar era que no es poguessin entrar dades amb un format erroni (lletres o caràcters que no fossin xifres al preu, minuts, etc.) i ho vam comprovar intentant entrar dades errònies.

Un cop assegurada aquesta part, vam començar a provar introduir clients i articles i un cop veiem que s'introduïen bé a la llista vam provar de repetir el email i el id.

Després vam afegir comandes. Com s'ha de triar un índex de la llista, hem provat a forçar un índex inexistent i quantitat menor a 1, i aquests problemes llencen una excepció. El mateix amb esborrar-les.

Hem comprovat també que, un cop enviada una comanda (ha passat el temps d'enviament) ja no es pot esborrar.

A la GUI ha sigut més fàcil fer les comprovacions i assegurar la robustesa del programa, ja que per exemple, a la hora d'esborrar una comanda, hem decidit bloquejar el botó fins que no hi hagi un índex seleccionat a la llista de comandes. D'aquesta manera evitem que es pugui demanar esborrar un índex que no existeixi.

## **5. Observacions generals.**

Considerem que la GUI permet molta més flexibilitat i robustesa a l'hora d'interactuar amb el usuari. Ja que és molt més còmode e intuïtiu treballar amb un programa que tingui interfície gràfica i a l'hora això permet al programador bloquejar alguns controls concrets en funció de les propietats o esdeveniments que ens interessin dels altres elements o del mateix.