



Università degli Studi di Salerno
Corso di Ingegneria del Software

JapanWorld
Objective Design Document
Versione 1.1



Data: 17/08/2020

Progetto: JapanWorld	Versione: 1.1
Documento: Objective Design Document	Data: 25/08/2020

Coordinatore del progetto:

Nome	Matricola
Emanuele Patella	0512104730

Partecipanti:

Nome	Matricola
Emanuele Patella	0512104730

Scritto da:	Emanuele Patella
--------------------	------------------

Revision History

Data	Versione	Descrizione	Autore
20/08/2020	1.0	Objective Design Document Versione 1	Emanuele Patella
25/08/2020	1.1	Revisione Documento	Emanuele Patella

Indice

- 1. Introduzione.....4
 - 1.1. Objective design trade-offs.....4
 - 1.2. Interface documentation guide-lines.....4
- 2. Packages.....4
 - 1. Class Interface Glossary.....5

1. Introduzione

1.1. Obiettivi di design e trade-off

Il nostro principale obiettivo è quello di fornire un sistema che privilegi la rapidità di risposta e la semplicità di utilizzo a fronte di possibili errori del sistema, sempre nei limiti di gestibilità da parte di un utente.

Il trade-off più ingente lo si ha nella procedura di acquisto, visto che, qualora un prodotto(del quale supponiamo sia rimasta una sola unità in magazzino) venisse aggiunto da più di un utente al carrello contemporaneamente, solo il primo di tali utenti avrebbe effettivamente ricevuto quello già presente in magazzino, mentre gli altri si troveranno costretti ad attendere un rifornimento da parte del venditore o la restituzione dell'importo.

1.2. Linee guida per l'utilizzo dell'interfaccia

I file che andranno a comporre l'interfaccia saranno nominati e suddivisi nel modo seguente:

- I file che andranno a comporre l'interfaccia grafica avranno nomi composti da sostantivi che fanno riferimento all'attività la cui pagina è legata(es.: L'interfaccia della pagina di Login si chiamerà: "Login.html" ecc.);
- I nominativi dei controller, cioè, dei file che si occupano di processare i dati prima di passarli ai model per la query al database, saranno composti da brevi frasi che fanno riferimento all'operazione che la loro query svolgerà, quindi sono contraddistinti da verbi(es.: La funzione che processa i dati e permette di acquistare i prodotti presenti nel proprio carrello prende il nome di: "Acquisto.java");
- I nominativi dei model, infine, saranno dei sostantivi e fanno riferimento all'entità del database con cui andranno a interfacciarsi per effettuare la query(es.: Il model nel quale sono racchiuse le funzioni che interagiscono con i prodotti da acquistare si chiamerà: "Carrello.java")

2. Pacchetti

La decomposizione del progetto in pacchetti è sostanzialmente composta di tre macro-blocchi che interagiscono tra di loro per permettere al software di funzionare correttamente:

- Il primo pacchetto è quello dell'interfaccia grafica, il quale può essere visualizzato anche senza la presenza degli altri due pacchetti di cui è composto il progetto, ma risulterebbe solo e puramente estetico, in quanto deve interagire con gli altri due package per eseguire le operazioni più importanti per chi il sito sta venendo sviluppato;
- Il secondo package è quello dei controller, si tratta di un pacchetto le cui funzioni sono molto semplici e brevi, sono il tramite tra le funzioni model che effettuano le query e le funzioni di interfaccia. In quanto tramite, sono dipendenti dalle funzioni model per dare un output sensato al sistema e dipendono anche dall'interfaccia grafica per l'input dei dati. Questo tipo di file non hanno dei veri e propri errori proprietari da ritornare al software, se non quelli causati da problemi di libreria, ma restituiscono gli stessi errori dei model;
- Il terzo package, nonché più importante tra i 3, è quello dei model, che si occupa di effettuare le query al database e restituire i risultati in output ai controller, che li formattano e li inviano all'interfaccia appropriata per la visualizzazione.

3. Class interface glossary

La sezione di codice che andrò ad implementare sarà composta da:

- 6/7 file di interfaccia grafica, i quali avranno il compito di prendere in input i dati che l'utente dovrà fornire per le operazioni e mostrare l'output delle stesse. Prenderanno i nomi di:
 - Home-page.jsp, Login.html, Iscrizione.html(solo qualora l'utente non avesse già un profilo), RicercaNaruto.jsp, Carrello.jsp, CampiPagamento.jsp, Scontrino.jsp;
- 6 controller, che si occuperanno di processare tutti i dati in ingresso per rendere possibile al query al database. I loro nomi saranno:

- RicercaMangaNaruto.java, Iscrizione.java, Acquisto.java, VisualCarrello.java, Fattura.java, VisualFattura.java;
- 5 model, che conterranno tutte le funzioni necessarie a effettuare le query con i dati forniti dai controller. I loro nomi saranno:
 - DriverManagerConnectionPool.java, Carrello.java, Login.java, Prodotti.java, Utente.java.