

**INGEGNERIA DEL SOFTWARE 2016-2017**

**Test Plan 0.1**

**Fiorazon**

**16/12/2016**

**Sommario**

[**1.** **Introduction** 2](#_Toc469652401)

[**2.** **Relationship to other documents** 3](#_Toc469652402)

[**3.** **System overview** 3](#_Toc469652403)

[**4.** **Features to be tested/not to be tested** 3](#_Toc469652404)

[**5.** **Pass/Fail criteria** 3](#_Toc469652405)

[**6.** **Approach** 3](#_Toc469652406)

[**7.** **Suspension and Resumption** 3](#_Toc469652407)

[**8.** **Testing materials** 3](#_Toc469652408)

[**9.** **Test Cases** 3](#_Toc469652409)

[**10.** **Testing Schedule** 4](#_Toc469652410)

[**11.** **Team** 4](#_Toc469652411)

[**12.** **Storia delle versioni** 4](#_Toc469652412)

1. **Introduction**

Questo è il test plan document dell’applicazione web “Fiorazon”. L’obiettivo del test è quello di provare il corretto funzionamento dei metodi delle classi di controllo e manager del sistema.  
Si utilizzerà la modalità di test “black box” in quanto, data la semplicità dell’applicazione, si è interessati unicamente al comportamento delle singole classi.  
Si procederà in prima fase con il test di unità, successivamente verrà eseguito anche un test di integrazione per verificare la corretta interdipendenza delle varie classi.

1. **Relationship to other documents**

Per la redazione di questo documento si è dovuto far riferimento ai documenti RAD, SDD e ODD; questo per comprendere al meglio il funzionamento delle singole classi e i vincoli di inserimento dei valori (come ad esempio espressi nel RAD).   
Analizzando il documento RAD e in particolare i requisiti funzionali e non funzionali si sono potuti evidenziare i casi di test su tutte le operazioni riguardanti il carrello, il catalogo, l’ordine, l’utente.  
Per quanto riguarda i documenti SDD e ODD, sono serviti ad individuare le relazioni tra le classi e i metodi che le classi espongono.

1. **System overview**

Saranno testate nella fase di testing le classi .java dei package carrelloPackage, utentiPackage, prodottiPackage, acquistoPackage.  
Del package carrelloPackage saranno testati i metodi delle classi Carrello.java, ControlloCarrello.java, CarrelloManager.java.  
Del package utentiPackage saranno testati i metodi delle classi Utente.java, ControlloUtenti.java, UtentiManager.java.  
Del package prodottiPackage saranno testati i metodi delle classi Prodotto.java, ControlloProdotti.java, ProdottiManager.java.  
Del package acquistoPackage saranno testati i metodi delle classi Ordine.java, ControlloOrdini.java, OrdiniManager.java.

Ci sono queste relazioni tra le classi dello stesso package.  
 In carrelloPackage, ControlloCarrello.java usa CarrelloManager.java e Carrello.java.  
 In utentiPackage, ControlloUtenti.java usa UtentiManager.java e Utente.java.  
 In prodottiPackage, ControlloProdotti.java usa ProdottiManager.java e Prodotto.java.  
 In acquistoPakcage, ControlloOrdini.java usa OrdiniManager.java e Ordine,java.

1. **Features to be tested/not to be tested**

**Funzionalità da testare:**

Vanno testati tutti i metodi delle classi ControlloUtenti.java, UtentiManager.java, ControlloProdotti.java, ProdottiManager.java, ControlloCarrello.java, CarrelloManager.java, ControlloOrdini.java, OrdiniManager.java come esposto nell’ODD.  
Inoltre vanno testate le relazioni tra i metodi di ControlloUtenti.java e UtentiManager.java, tra ControlloProdotti.java e ProdottiManager.java, tra ControlloCarrello.java e CarrelloManager.java, tra ControlloOrdini.java e OrdiniManager.java. In dettaglio vanno testate le relazioni tra le seguenti coppie di metodi:   
ControlloCarrello.controlloAcquistoCarrello() – CarrelloManager.controlloAcquistoCarrello()  
ControlloCarrello.aggiungiProdottoCarrello() – CarrelloManager.aggiungiProdottoCarrello()

ControlloCarrello.rimuoviProdottoCarrello() – CarrelloManager.rimuoviProdottoCarrello()

ControlloProdotti.aggiungiProdotto() – ProdottiManager.aggiungiProdotto()  
ControlloProdotti.caricaProdotti() - ProdottiManager.caricaProdotti()   
ControlloProdotti.cancellaProdotto() - ProdottiManager.cancellaProdotto()   
ControlloProdotti.modificaProdotto() - ProdottiManager.modificaProdotto()   
ControlloProdotti.ricercaProdotti() ProdottiManager.ricercaProdotti()

ControlloUtenti.controlloCancella() - UtentiManager.controlloCancella()

ControlloUtenti.controlloLoginAmministratore() - UtentiManager.controlloLoginAmministratore  
ControlloUtenti.controlloLoginUtente() - UtentiManager.controlloLoginUtente  
ControlloUtenti.controlloModificaDati() - UtentiManager.controlloModificaDati()  
ControlloUtenti.controlloRegistrazione() - UtentiManager.controlloRegistrazione()

ControlloAcquisto.caricamentoOrdini() - AcquistoManager.caricamentoOrdini()  
ControlloAcquisto.controlloSpedizione() - AcquistoManager.controlloSpedizione()

ControlloAcquisto.controlloPagamento() - AcquistoManager.controlloPagamento()

**Funzionalità da non testare:**

Non vanno testati i metodi delle classi Utente.java, Prodotto.java, Carrello.java, Ordine.java in quanto si tratta, per queste classi, unicamente di metodi “get” e “set”.

1. **Pass/Fail criteria**

Gli esiti dei test si baseranno su quanto detto nei documenti precedenti, in particolare nel RAD per quanto riguarda i vincoli di inserimento e nell’ODD per quanto riguarda l’output atteso dai metodi delle classi testate.

1. **Approach**

Nella fase di testing verrà utilizzato un approccio “Black box”.   
Ogni classe da testare verrà testata prima individualmente tramite il test d’unità, successivamente le componenti verranno testate nel test d’integrazione con una strategia bottom up, accoppiando prima i model ai control, successivamente accoppiando ai control i bean. In particolare verrà seguito questo ordine.  
Unit Testing di tutte le classi (prese individualmente).   
Inoltre verranno testati gli accoppiamenti:

CarrelloManager.java – ControlloCarrello.java – Carrello.java  
UtentiManager.java – ControlloUtenti.java – Utente.java  
ProdottiManager.java – ControlloProdotti.java – Prodotto.java

OrdiniManager.java – ControlloOrdini.java – Ordine.java

1. **Suspension and Resumption**

A causa della semplicità del sistema, non sono presenti criteri di sospensione del testing oppure attività che devono essere eseguite quando il testing viene riavviato.

1. **Testing materials**
2. **Test Cases**
3. **Testing Schedule**
4. **Team**

|  |  |
| --- | --- |
| **Nome** | **Matricola** |
| Giuseppe Siani | 0512102958 |
| Carmine D’ Alessandro | 0512103402 |
| Aleandro Giuseppe Libano | 0512102976 |

1. **Storia delle versioni**

|  |  |  |  |
| --- | --- | --- | --- |
| DATA | VERSIONE | DESCRIZIONE | AUTORE |
| 16/12/2016 | 0.1 | Parziale inserimento dei punti del documento | Giuseppe Siani, Carmine D’Alessandro |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |