**WineShop – Federico Risoli Elia Candida**

Il documento può essere scritto in inglese o in italiano. Le parti che coinvolgono il codice (e.g., un class diagram) devono essere in inglese. Le parti interne a rettangoli con bordi verdi indicano note che vanno rimosse

**Summary**

# Summary 1 Introduction 1 Revision History 1 Requirements 2 Use Case Documentation Architectural Model Detailed Software Model Functional Tests System Manual: Installation Configuration User Guide

# 

**Introduction**

Inserire la descrizione iniziale del progetto che ovviamente può evolvere nel tempo

# WineShop è un client che permetta a clienti e dipendenti di un negozio di vini, di interfacciarsi con un Sistema che permette ai clienti di comprare e ordinare vini e ai dipendenti di gestire gli ordini, clienti e altro, tutto in un unico Sistema.

**Revision History**

| **Version** | **Date** | **Author(s)** | **Description** |
| --- | --- | --- | --- |
| **1** | **Fine dicembre** | **Elia Candida-Federico Risoli** | **Inizio documentazione e progetto** |
| **2** | **Metà gennaio** | **Elia Candida-Federico Risoli** | **Implementate metà delle funzionalità** |
| **3** | **Metà marzo** | **Elia Candida-Federico Risoli** | **Bugfix, fine Progetto e documentazione ultimata** |

# **Requirements**

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Function name** | **Status** | **Priority** |
| F1 | Login | implemented | High |
| F2 | User Registration | implemented | High |
| F3 | Employee Registration | implemented | Medium |
| F4 | Modify Password | implemented | Low |
| F5 | Wine Search | implemented | High |
| F6 | Variable Wine Price | implemented | Medium |
| F7 | Popup Promozioni | implemented | Low |
| F8 | Buy Wine | implemented | High |
| F9 | Logout | implemented | Low |
| F10 | My order | implemented | Low |
| F11 | Notify | implemented | Medium |
| F12 | Complete Order | implemented | High |
| F13 | Client Search | implemented | Medium |
| F14 | Order Search | implemented | Medium |
| F15 | Delete Employee | implemented | Medium |
| F16 | Reset Password Employee | implemented | Medium |
| F17 | Generate Monthly Report | implemented | High |
| F18 | Automatic Discount | implemented | Low |
| F19 | Multiple Payment Methods | implemented | Medium |
| F20 | Automatic Orders | implemented | Medium |
| F21 | Order Wines not in Stock | implemented | High |
| F22 | Automatic Orders to Supplier for easier use | Implemented | Medium |
| F23 | Employee Can mark Order as arrived and change the price if needed | Implemented | High |
| F24 | Clients can Buy Wines that were not in stock if they still wish to do this | Implemented | High |
| F25 | Automatic Orders when Wines are Low | Implemented | High |
| F26 | Client/Server Architecture | Implemented | High |
| F27 | Round Robin Work Assignment | Implemented | High |

# 

# **Use Cases Documentation**

Definizione del diagramma dei casi uso che descrive le funzionalità, il confine e gli attori coinvolti nel sistema

**Use cases diagram**

Per ogni caso di uso devono essere compilati gli elementi indicati nel template (qui sotto indicato), ma le righe, corrispondenti a: Special Requirements, Assumptions e Notes and Issues, possono essere non compilate. Se utile, per alcuni casi di uso può essere introdotto un diagramma che descriva le funzionalità (eredità, inclusioni ed estensioni) che concorrono alla definizione del caso di uso stesso. Ulteriori informazioni sono indicate nel documento “Use Case Template GuideFile” presente su Elly

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case ID: |  | | |
| Use Case Name: |  | | |
| Created By: |  | Last Updated By: |  |
| Date Created: |  | Date Last Updated: |  |

|  |  |
| --- | --- |
| Description: |  |
| Primary Actor: |  |
| Secondary actors: |  |
| Pre-conditions: |  |
| Post-conditions: |  |
| Trigger: |  |
| Priority: |  |
| Frequency of Use: |  |
| Normal Course of Events: |  |
| Alternative Courses: |  |
| Exceptions: |  |
| Includes: |  |
| Special Requirements: |  |
| Assumptions: |  |
| Notes and Issues: |  |

# **Architectural Model**

**Architecture diagram**

**…**

# **Detailed Software Model**

**Class diagram(s)**

**…**

# **Functional Tests**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **ID** | **Test name** | **Steps** | **Expected result** | **Test result** | **Comments** |
| 1. | Unit Test Carrello.java | 1. Setup pre test 2. Testo i metodi set/get/clear/non-disponibile della classe | pass | pass |  |
| 2. | Unit Test Data.java | 1. Testo tutte le funzioni per vedere se funzionano correttamente | pass | pass |  |
| 3. | Unit Test PersonnelPageController.java | 1. Testo nei casi limite le funzioni roundtodecimalplaces e isValidDate | Pass | pass |  |
| 4 | Unit Test ConcurrentServer.java | 1. Setup con clear delle variabili che rimangono da un test all’altro. 2. Testo le funzioni che Aggiungono/rimuovono/ottengono gli impiegati e gli ordini di vendita registrati nel server | Pass | pass |  |
| 5 | Test UI | 1. Entrare in una pagina 2. Testare il corretto funzionamento di tutti i suoi elementi 3. Ripetere per ogni Controller | Pass | Pass |  |
| 6 | Test Generale Sviluppatori | 1. Testare tutto il sistema in maniera insolita per evidenziarne i bug | Pass | Pass | Eseguito da Elia e Federico non sono stati trovati bug malgrado ci fossero; quindi il test è passato malgrado bug nascosti (i test non assicurano l’assenza di bug) |
| 7 | Alpha Test 1 | 1. Testare tutto il sistema in maniera insolita per evidenziarne i bug | Pass | Fail | Eseguito da un amico degli sviluppatori per avere visioni differenti sul programma. Ha evidenziato un bug quindi il test conta come fallito |
| 8 | Alpha Test 2 | 1. Testare tutto il sistema in maniera insolita per evidenziarne i bug | Pass | Pass | Eseguito dopo l’Alpha Test 1 e dopo il fix del bug trovato. |

# **System Manual**

Deve descrivere cosa serve e come si installa il sistema

**Installation**

**…**

**Configuration**

Deve descrivere cosa bisogna fare per configurare il sistema

**…**

**User Guide**

Deve descrivere quale sono le operazioni che devono fare gli utenti per usufruire delle funzionalità offerte dal sistema

**Clienti**

**1-Login**

* Utenti Già Registrati
  1. Inserire username e password
  2. Cliccare su Login
* Utenti non Registrati
  1. Cliccare su “Registrati Cliente”
  2. Compilare tutti i campi
  3. Cliccare su Registra

**2-Comprare/Ordinare/Ricercare Vini**

1. Selezionare i vini che si desiderano comprare con l’apposita checkbox sul lato sinistro della tabella
   1. Si possono ricercare I vini compilando gli apposite campi ”Nome Prodotto” o selezionare l’annata
   2. Per Resettare la ricerca cliccare su “Clear”
2. Inserire la quantità per ogni prodotto selezionato
3. Cliccare il tasto “Compra”
4. Selezionare il metodo di pagamento e cliccare su “procedi al pagamento”
5. Compilare i campi e cliccare su “Paga”

* **NB se i vini selezionati non sono disponibili**
  + - 1. Inserire l’indirizzo di consegna nell’ apposite campo
      2. Cliccare su Ordina

**3-Visualizzare i propri ordini**

1. Nella pagina principale cliccare su “I miei Ordini”

**4-Notifiche-Ordini Pronti**

* 1. Cliccare su Notifiche
  2. Selezionare metodo di pagamento se si vuole acquistare e cliccare su accetta
     1. (altrimenti premere annulla)
  3. Compilare i campi richiesti dal metodo di pagamento e cliccare su “Paga”