**WoodLot**

**System Design Document**

**Versione 1.0**

****

Data: 21/12/2022

|  |  |
| --- | --- |
| **Coordinatore del progetto:** | |
| **Nome e Cognome:** | **Matricola:** |
| Alessia Ture | 0512107758 |

|  |  |
| --- | --- |
| **Partecipanti:** | |
| **Nome e Cognome:** | **Matricola:** |
| Raffaella Spagnuolo | 0512107794 |
| Salvatore Di Sanza | 0512106066 |

|  |  |
| --- | --- |
| **Scritto da:** | Raffaella Spagnuolo |

|  |  |  |  |
| --- | --- | --- | --- |
| **Revisione History** | | | |
| **Data** | **Versione** | **Descrizione** | **Autore** |
| 01/12/2022 | 0.1 | Stesura decomposizione del sottosistema | RS |
| 01/12/2022 | 0.1 | Stesura servizi dei sottosistemi | AT, SDS |
| 01/12/2022 | 0.1 | Inserimento disegno mappatura hardware/software | SDS |
| 01/12/2022 | 0.1 | Stesura gestione dati persistenza | RS, AT |
| 01/12/2022 | 0.1 | Stesura controllo degli accessi e sicurezza | AT |
| 01/12/2022 | 0.1 | Stesura controllo globale del software | AT |
| 19/12/2022 | 0.2 | Correzione degli errori | RS |
| 19/12/2022 | 0.3 | Stesura Introduzione | RS |
| 19/12/2022 | 0.3 | Stesura Visione d’insieme | RS |
| 19/12/2022 | 0.3 | Stesura Glossario | RS |
| 21/12/2022 | 1.0 | Revisione documento | RS |

Sommario

[1. Introduzione 4](#_Toc123312298)

[**1.1** **Scopo del sistema** 4](#_Toc123312299)

[**1.2** **Design Goals** 4](#_Toc123312300)

[**1.2.1** **Criteri di Usabilità** 4](#_Toc123312301)

[**1.2.2** **Criteri di Affidabilità** 4](#_Toc123312302)

[**1.2.3** **Criteri di Modularità** 5](#_Toc123312303)

[**1.3** **Trade-off** 5](#_Toc123312304)

[**1.4** **Definizioni, acronimi e abbreviazioni** 5](#_Toc123312305)

[**1.4.1** **Definizioni** 5](#_Toc123312306)

[**1.4.2** **Acronimi e Abbreviazioni** 5](#_Toc123312307)

[**1.5** **Riferimenti** 6](#_Toc123312308)

[**1.6** **Panoramica** 6](#_Toc123312309)

[2. Architettura software proposta 6](#_Toc123312310)

[**2.1** **Visione d’insieme** 6](#_Toc123312311)

[**2.2** **Decomposizione dei sottosistemi** 6](#_Toc123312312)

[**2.3** **Mappatura hardware/software** 8](#_Toc123312313)

[**2.4** **Gestione dei dati persistenti** 9](#_Toc123312314)

[**2.5** **Controllo degli accessi e sicurezza** 9](#_Toc123312315)

[**2.6** **Controllo globale del software** 11](#_Toc123312316)

[**2.7** **Condizioni di confine** 11](#_Toc123312317)

[3. Servizi di sottosistemi 11](#_Toc123312318)

[4. Glossario 12](#_Toc123312319)

# Introduzione

## **Scopo del sistema**

Il nostro progetto permette all’utenza di effettuare un acquisto simbolico di un albero. L’obiettivo principale è quello di contribuire alla piantumazione di aree verdi e di aiutare contadini che non hanno la possibilità di iniziare a piantare alberi autonomamente. La piattaforma sarà estremamente semplice da utilizzare, offrendo un’interfaccia minimale ed intuitiva, in modo tale che qualsiasi utente riesca ad effettuare ogni operazione senza alcun problema.

## **Design Goals**

La piattaforma WoodLot punterà ad essere il più lineare ed intuitiva possibile, in modo da offrire questa caratteristiche alche nell’interfaccia grafica. Il sistema cercherà di far trasparire tramite la sua semplice interfaccia grafica, tutte le funzionalità disponibili all’utente, il quale potrà utilizzarle con facilità, evitando problemi ed inserimenti di dati errati.

Il sistema proposto rispetterà i seguenti **Criteri di Design**:

### **Criteri di Usabilità**

|  |  |  |
| --- | --- | --- |
| ID | Nome | Descrizione |
| DG\_CU\_01 | Responsiveness | L’interfaccia grafica del sistema deve adattarsi alle grandezze dei principali dispositivi in commercio quali: computer, tablet e smatphone. La responsiveness verrà implementata tramite framework Bootstrap |
| DG\_CU\_02 | EvidenziareErrori | Il sistema deve riconoscere ed evidenziare gli errori di compilazione di un form con un’apposita colorazione che consenta all’utente di capire la provenienza di un determinato errore, il sistema deve inoltre fornire suggerimenti utili a correggerli utilizzando JavaScript |
| DG\_CU\_03 | InterfacciaGrafica | WoodLot deve rendere le operazioni accessibili a qualsiasi categoria di utente indipendentemente dalle loro capacità informatiche tramite l’utilizzo di un’interfaccia grafica minimale e di facile comprensione, nello specifico verranno utilizzati fogli di stile CSS ed il frameeork BootStrap, la grafica ha priorità più elevata rispetto a tutto il resto anche a discapito di performance e manutenibilità |

### **Criteri di Affidabilità**

|  |  |  |
| --- | --- | --- |
| ID | Nome | Descrizione |
| DG\_CA\_01 | ConsistenzaDati | Il sistema registra dati e transazioni, garantendo il loro salvataggio in un database e la loro usabilità |
| DG\_CA\_02 | SicurezzaDati | Le credenziali di accesso devono essere criptate all’interno del database e di queste non deve essere possibile il recupero, ma soltanto la sostituzione. Questi dati sensibili saranno criptati in MD5 direttamente all’interno del database e manipolati nel codice utilizzando le apposite funzioni di libreria |
| DG\_CA\_03 | TolleranzaGuasti | Il sistema deve garantire che i dati inseriti dall’utente siano corretti, in modo da non immettere dati errati all’interno del database. Questi controlli verranno implementati con Java in modo che sia il server a poterli effettuare |

### **Criteri di Modularità**

|  |  |  |
| --- | --- | --- |
| ID | Nome | Descrizione |
| DG\_CM\_01 | Modificabilità | Il sistema permette di apportare modifiche alle funzionalità già implementate senza la necessità di modificare i sottosistemi |
| DG\_CM\_02 | Commenti | Il codice della piattaforma deve avere dei commenti utili alla comprensione delle principali funzionalità |

## **Trade-off**

* **Usabilità vs Performance:** il sistema sarà implementato in modo da preferire l’usabilità alla performance, in modo da rendere l’interfaccia grafica più accessibile. Si è deciso di non preferire le performance visto che non si prevedono grandi afflussi di utenti.
* **Sicurezza vs Performance:** il sistema sarà implementato in modo da preferire la sicurezza alla performance, in modo da rendere il sistema più sicuro.

## **Definizioni, acronimi e abbreviazioni**

### **Definizioni**

* **Responsiveness:** capacità di una pagina web di adattarsi al tipo di dispositivi in uso;
* **Fault:** qualsiasi errore o malfunzionamento critico che non sia stato adeguatamente gestito porterebbe all’arresto del sistema-

### **Acronimi e Abbreviazioni**

* **SDD:** System Design Document
* **DG:** Design Goals
* **CU:** Criteri di Usabilità
* **CA:** Criteri di Affidabilità
* **CM:** Criteri di Modularità

## **Riferimenti**

* 2022\_RAD\_WoodLot
* Object Oriented Software Engineering Using UML

## **Panoramica**

Il documento si compone di **3 parti fondamentali**:

1. **Introduzione:** una breve panoramica dell'architettura software e degli obiettivi di progettazione. Fornisce inoltre riferimenti ad altri documenti e informazioni sulla tracciabilità.
2. **Architettura del sistema proposto:** viene illustrato il modello di progettazione del nuovo sistema.
3. **Servizi dei sottosistemi e glossario:** descrive i servizi forniti da ciascun sottosistema in termini di operazioni.

# Architettura software proposta

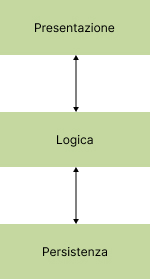
## **Visione d’insieme**

Il sistema prevederà l’utilizzo di un’architettura Client-Server: il lato Cliente di occuperà delle pagine di presentazione del sistema, quindi delle view, mentre il lato Server si occuperà della logica applicativa. Per il server sarà utilizzato il WebServer Apache Tomcat, per la gestione dei dati sarà utilizzato il DBMS MySQL.

## **Decomposizione dei sottosistemi**

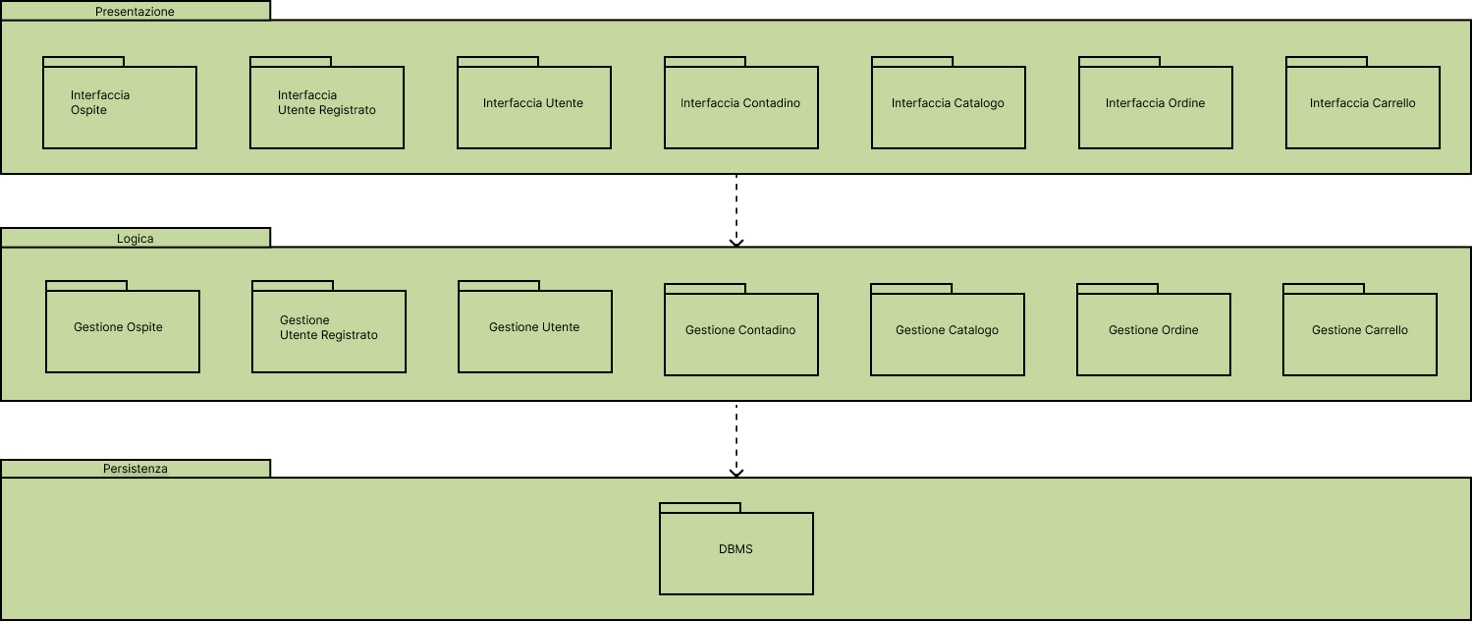
La decomposizione prevista per il sistema è composta da tre layer che si occupano di gestire aspetti e funzionalità differenti, in particolare è stato scelto il modello Three-Tier:

* Presentazione: contiene tutte le possibili viste con cui l’utente può interagire.
* Logica: include gli oggetti relativi al controllo e all’entità.
* Persistenza: si occupa di ricercare e gestire i dati persistenti.



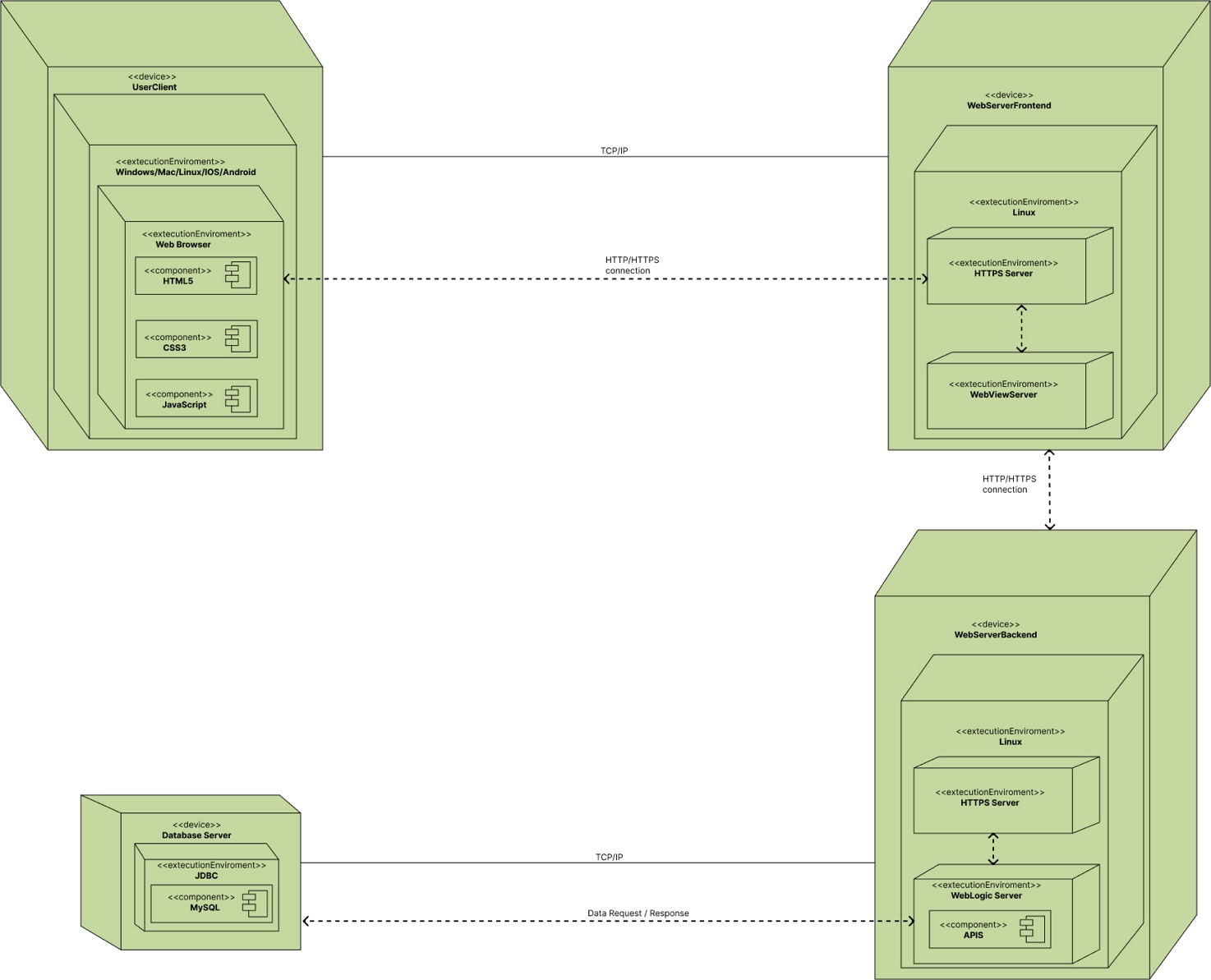
Il sistema principale è suddiviso in vari sottosistemi che comunica tra di loro, che sono:

* **Gestione Ospite:** si occupa di gestire la funzione di registrazione;
* **Gestione Utente Registrato**: si occupa di gestire le funzioni di login, logout e reimpostazione della password;
* **Gestione Utente:** è responsabile della creazione e modifica di una foresta, dell’aggiunta di un albero alla foresta e del riscatto dei TreeCode;
* **Gestione Contadino:** si occupa di gestire le funzionalità riguardanti il contadino, ovvero, la modifica dei dati di pagamento e l’aggiornamento dello stato di un albero;
* **Gestione Catalogo:** è responsabile dell’aggiunta, modifica ed eliminazione di un prodotto dal catalogo e del filtraggio degli articoli del catalogo;
* **Gestione Ordine:** si occupa di gestire gli ordini effettuati dai clienti e il conseguente pagamento del contadino per evadere l’ordine;
* **Gestione Carrello:** è responsabile dell’aggiunta ed eliminazione di un prodotto dal carrello, della modifica della quantità di un prodotto e dell’eliminazione di tutti i prodotti dal carrello.



## **Mappatura hardware/software**

Il sistema utilizzerà una struttura hardware costituita da due server denominati ServerFrontend e ServerBackend. Il ServerBackend risponderà alle richieste del ServerFrontend mentre il ServerFrontend risponderà alle richieste inviate tramite interazione del client. Il client è rappresentato da una qualsiasi macchina con una connessione Internet e una web browser installato su di esso, in modo da potersi connettere al ServerFrontend per interagire col sistema. Il ServerFrontend contiene la Web Application utilizzabile dagli utenti. Invece, il ServerBackend è rappresentato da una qualsiasi macchina con una connesione Internet, dove è presente il DBMS MySQL ed è capace di immagazzinare una certa quantità di dati. Esso, inoltre, gestisce la logica e i dati persistenti contenuti nel database. La comunicazione tra il client e il ServerBackend è basata sul protocollo HTTPS a cui il client invierà delle richieste e il server provvederà alle risposte.



## **Gestione dei dati persistenti**

La documentazione associata a questo paragrafo è stata inserita in un documento a parte denominato “Gestione dati persistenti”.

## **Controllo degli accessi e sicurezza**

WoodLot è un sistema multiutente, vi sono differenti attori ognuno con l’autorizzazione per compiere differenti operazioni. Il controllo degli accessi è garantito tramite l’utilizzo di username e password, che garantiranno l’accesso alle operazioni permesse in base al tipo di utente registrato. Si ricorrerà all’utilizzo della sessione del server per tenere traccia dell’utente loggato.

Per questioni di efficienza, la sessione sarà attiva per soli 20 minuti dopo l’ultima interazione dell’utente col sistema. Allo scadere della sessione in cui si opera verrà richiesto l’inserimento delle apposite credenziali. Il salvataggio delle password nel database sarà criptato.

Di seguito riportiamo la Global Access Table per il controllo degli accessi.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Attori/  Oggetti | Ospite | Utente | Contadino | Responsabile Catalogo |
| Gestione Ospite | Registrazione |  |  |  |
| Gestione Utente Registrato |  | Autenticazione, Logout, Reimpostare Password | Autenticazione, Logout, Reimpostare Password | Autenticazione, Logout, Reimpostare Password |
| Gestione Utente |  | Riscattare TreeCode, Ricercare foresta, Creare foresta, Aggiungere albero foresta, Modificare dati foresta. |  |  |
| Gestione Contadino |  |  | Modifica dati pagamento, Aggiornare stato. |  |
| Gestione Catalogo |  |  |  | Aggiungere prodotto, Modificare prodotto, Eliminare prodotto. |
| Gestione Ordine |  | Effettuare ordine |  |  |
| Gestione Carrello |  | Aggiungere prodotto, Eliminare prodotto, Modificare quantità, Svuotare. |  |  |

L’ospite può navigare in una porzione limitata di WoodLot, può, quindi, visualizzare la pagina “Home” e registrare un nuovo accountalla piattaforma come utenti oppure accedere ad un accountesistente; può visualizzare la pagina “Diventa un contadino” per registrarsi come contadino e visualizzare il catalogo dei prodotti e le pagine degli alberi.

L’utente oltre a poter visualizzare il catalogo e le pagine dei prodotti può interagire con il carrello ed effettuare un ordine, può inoltre riscattare un TreeCode. L’utente ha la possibilità di creare una foresta (o modificare i dati di una sua foresta) e di piantare alberi in una foresta.

Il contadino è un utente di WoodLot a cui vengono assegnati degli alberi. All’atto della registrazione fornisce delle credenziali bancarie che può successivamente modificare accedendo al suo profilo. Inoltre il contadino può aggiornare lo stato di un albero da lui curato.

Il responsabile del catalogo è una figura amministrativa che può aggiungere, modificare o eliminare un albero dal catalogo.

## **Controllo globale del software**

Lo stile architetturale scelto è “Controller-Service-Repository”, in particolare il Controller si occupa di gestire opportunamente le richieste dei client e di smistare la richiesta al servizio appropriato, il Service layer conterrà la logica di business, mentre il Repository layer è responsabile dell'archiviazione e del recupero di alcuni set di dati.

Il controllo software globale è di tipo event-driven e risiede in un dispatcher che chiama le funzioni mediante callback. Il Web Server si occupa di gestire le richieste effettuate dagli utenti. Quando l’utente interagisce con l’interfaccia grafica viene generata una richiesta che con i Control (dunque Java Servlet), gestiranno la richiesta eventualmente interagendo con i Service e risponde aggiornando l’interfaccia grafica.

## **Condizioni di confine**

Visto che il sistema sarà gestito da un web server non dovrà essere prodotto codice aggiuntivo , in quanto sarà gestito in automatica.

# Servizi di sottosistemi

|  |  |
| --- | --- |
| Gestione Ospite | |
| Registrazione | Consente la registrazione ad un ospite |

|  |  |
| --- | --- |
| Gestione Utente Registrato | |
| Logout | Consente all’utente che ha effettuato l’accesso di scollegarsi dal sistema |
| Autenticazione | Consente all’utente di effettuare l’accesso al sistema |
| Reimpostare password | Consente all’utente di reimpostare la password del suo profilo |

|  |  |
| --- | --- |
| Gestione Utente | |
| Riscattare TreeCode | Consente ad un utente di riscattare un TreeCode |
| Ricercare foresta | Consente ad un utente di ricercare una foresta |
| Creare foresta | Consente ad un utente di creare una foresta |
| Aggiungere albero foresta | Consente ad un utente di aggiungere un albero ad una foresta |
| Modificare dati foresta | Consente ad un utente proprietario di una foresta di modificare i dati della foresta |

|  |  |
| --- | --- |
| Gestione Contadino | |
| Modificare dati pagamento | Permette al contadino di modificare le coordinate bancarie per ricevere il pagamento |
| Aggiornare stato | Permette al contadino di aggiornare lo stato di un albero |
| Filtrare Catalogo | Consente di visualizzare il catalogo applicando dei filtri |
| Inserimento prodotto | Consente al responsabile catalogo di aggiungere un albero al catalogo |
| Modificare prodotto | Consente al responsabile catalogo di modificare i dati di un albero presente nel catalogo |
| Eliminare prodotto | Consente al responsabile catalogo di eliminare un prodotto |

|  |  |
| --- | --- |
| Gestione Ordine | |
| Effettuare ordine | Permette all’utente di effettuare l’acquisto dei prodotti inseriti nel carrello |
| Pagamento contadino | Consente al contadino di ricevere il sostentamento della pianta |

|  |  |
| --- | --- |
| Gestione Carrello | |
| Aggiungere prodotto | Consente ad un utente di aggiungere un prodotto al carrello |
| Eliminare prodotto | Consente ad un utente di eliminare un prodotto al carrello |
| Modificare quantità | Consente ad un utente di modificare un prodotto al carrello |
| Svuotare | Consente ad un utente di eliminare tutti i prodotti presenti nel carrello. |

# Glossario

A

B

C

**:** Anidride Carbonica

D

E

F

G

H

**HTTPS:** Hyper Text Transfer Protocol Secure, protocollo di trasferimento ipertestuale sicuro. È una dicitura visualizzata negli URL di un sito Web protetto con un certificato SSL.

I

L

**Login:** Procedura di accesso a un sistema informatico, che prevede l’inserimento di un codice identificativo e di una parola d’ordine da parte dell’utente.

**Logout:** Procedura per scollegarsi da un sistema informatico a cui si era avuto accesso tramite un’operazione di login.

M