**WoodLot**

**Problem Statement**

**Versione 3.0**

****

Data: 07/02/2023

|  |  |
| --- | --- |
| **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** |
| 13/10/2022 | 0.1 | Aggiunta requisiti funzionali | RS |
| 13/10/2022 | 0.2 | Aggiunta requisiti non funzionali | SDS |
| 13/10/2022 | 0.3 | Aggiunta scenari | AT |
| 14/10/2022 | 0.4 | Studio ed esplorazione del dominio del problema | [Tutti] |
| 18/10/2022 | 0.5 | Aggiunta deliverable & deadline | [Tutti] |
| 18/10/2022 | 0.6 | Aggiunta target environment | [Tutti] |
| 18/10/2022 | 0.7 | Revisione | [Tutti] |
| 18/10/2022 | 1.0 | Approvazione | [Tutti] |
| 20/10/2022 | 1.1 | Modifica scenari | AT |
| 21/10/2022 | 1.2 | Modifica requisiti funzionali | RS |
| 21/10/2022 | 1.3 | Modifica requisiti non funzionali | RS |
| 26/10/2022 | 1.4 | Revisione | [Tutti] |
| 26/10/2022 | 2.0 | Approvazione | [Tutti] |
| 07/02/2023 | 3.0 | Aggiunta nuovo ruolo | RS |

Sommario

[1. Introduzione 4](#_Toc126693050)

[**1.1** **Dominio del problema** 4](#_Toc126693051)

[2. Scenari 5](#_Toc126693052)

[**2.1** **S1 – Interazione utente – gestore ordini – contadino** 5](#_Toc126693053)

[**2.2** **S2 - Utente non registrato** 6](#_Toc126693054)

[**2.3** **S3 - Contadino non registrato** 7](#_Toc126693055)

[3. Requisiti funzionali 8](#_Toc126693056)

[***3.1*** ***Gestione ospite*** 8](#_Toc126693057)

[***3.2*** ***Gestione utente*** 8](#_Toc126693058)

[***3.3*** ***Gestione contadino*** 9](#_Toc126693059)

[***3.4*** ***Responsabile del catalogo*** 10](#_Toc126693060)

[***3.5*** ***Responsabile ordini*** 10](#_Toc126693061)

[4. Requisiti non funzionali 12](#_Toc126693062)

[***4.1*** ***Usabilità*** 12](#_Toc126693063)

[***4.2*** ***Affidabilità*** 12](#_Toc126693064)

[***4.3*** ***Supportabilità*** 12](#_Toc126693065)

[***4.4*** ***Implementazioni*** 13](#_Toc126693066)

[***4.5*** ***Interfaccia*** 13](#_Toc126693067)

[***4.6*** ***Packaging*** 13](#_Toc126693068)

[***4.7*** ***Legali*** 14](#_Toc126693069)

[5. Target Environment 15](#_Toc126693070)

[6. Deliverable & deadline 15](#_Toc126693071)

# Introduzione

Realizzazione di un progetto inerente all’acquisto di alberi tramite la creazione di una moderna piattaforma e-commerce: WoodLot.

## **Dominio del problema**

Il progetto WoodLot nasce dall’esigenza di permettere a migliaia di contadini di far fronte ai costi iniziali della piantumazione di nuovi alberi, attraverso l’acquisto simbolico di un albero per creare un ecosistema sostenibile.

Piantare alberi è un lavoro complesso. Non solo da un punto di vista tecnico, servono infatti competenze e risorse per creare luoghi adatti a far germinare e crescere i piccoli alberi, ma anche soprattutto da un punto di vista strategico. Piantare alberi non basta, vanno accompagnati nella crescita. Questo significa ragionare sul medio e lungo periodo e per questo è necessario individuare l’albero giusto per il posto giusto e lo scopo giusto.

L’aspetto da tenere in considerazione riguarda la capacità di realizzare benefici ambientali, che si esplicano sia a livello locale che globale.

La realizzazione di sistemi agroforestali permette di operare scelte puntuali in relazione alle specifiche caratteristiche e necessità di ogni area di progetto. Le specie arboree piantate sono native o in rispetto della biodiversità dei diversi territori. La pratica agroforestale integra, inoltre, la piantumazione di alberi in un sistema agricolo, favorendo l’interazione virtuosa tra le diverse specie ed un uso sostenibile delle risorse e dei terreni. Tutti gli alberi, infine, nel corso della propria crescita assorbono generando naturalmente un beneficio per l’intero pianeta.

Altrettanto importanti sono i benefici sociali per le comunità locali che partecipano, infatti, gli alberi e i loro frutti sono di proprietà dei contadini. Finanziare la piantumazione di nuovi alberi, serve a sostenere la cura nei primi anni di vita, quando non sono ancora produttivi in termini di frutti.

WoodLot sta ad indicare la possibilità di avere uno “spazio verde personale”, si propone di diventare un punto di riferimento per tutte le persone che vogliono contribuire alla crescita di aree verdi e alla diminuzione di , inoltre vorrebbe aiutare contadini che non hanno la possibilità di iniziare a piantare alberi autonomamente.

Tramite WoodLot si potranno acquistare diverse tipologie di piante, che verranno piantate per voi da contadini; il territorio, scelto per la piantumazione, verrà selezionato in base ai bisogni dell’albero, nel pieno rispetto dell’ecosistema.

# Scenari

## **S1 – Interazione utente – gestore ordini – contadino**

Carlos è un giovane contadino guatemalteco e si è da poco registrato alla piattaforma WoodLot, si autentica sulla piattaforma inserendo le sue credenziali, **username:** ‘[carlos@gmail.com](mailto:carlos@gmail.com)’ e **password:** ‘Pippo-1950’. Carlos accede al suo profilo e il sistema lo informa che deve effettuare degli aggiornamenti per degli alberi che ha piantato mesi addietro. Quindi, scatta delle foto con il suo smartphone e raccoglie la frutta e la pesa. In seguito, accedendo al sistema, compila il form per aggiornare i dati dell’albero. I dati da inserire sono: **descrizione:** ‘Il tuo albero sta crescendo, ha prodotto i primi frutti.’, **foto:** ‘4532.jpg’, **frutta raccolta:** ‘6 kg’. Il sistema valida i dati inseriti e aggiorna i dati dell’albero.

Antonio ha comprato un albero sulla piattaforma WoodLot tempo addietro ed è molto curioso di vedere i progressi. Per fare ciò, si autentica sulla piattaforma inserendo le credenziali, **username:** ‘[antonio@gmail.com](mailto:antonio@gmail.com)‘ e **password:** ‘Pippo-1234’ e accede al suo profilo e controlla gli alberi acquistati; vede che un contadino ha provveduto a caricare l’aggiornamento di un albero e può vederne le informazioni, **specie:** ‘Banano’, **stato:** ‘frutto’, **descrizione:** ‘Il tuo albero sta crescendo, ha prodotto i primi frutti.’, **dove è piantato:** ‘Guatemala’, **frutti raccolti:** ‘6 kg’.

Antonio entusiasta dei progressi del suo albero decide di comprare un altro albero; quindi, osservando gli alberi presenti nel catalogo, uno in particolare attira la sua attenzione: l’albero di ‘Pino’, decide di vedere le informazioni relativa all’albero: **nome:** ‘Pino’, **nome scientifico:** ‘Pinus halepensis’, **prezzo:** ‘22,90€’, **breve descrizione:** ‘Inconfondibile! Aveta mai provato il vino Retsina? Un vino greco a cui si aggiunge la resina… è la resina del pino di Aleppo. Gli dà l’inconfondibile sapore amaro.’, **descrizione:** ‘Il pino di Aleppo è tipico della regione mediterranea, tipicamente lo si trova in prossimità della costa, anche se nel tempo il suo areale è andato crescendo ed oggi si possono trovare pini di Aleppo anche oltre i 600 metri di altitudine. Questo albero, che cresce fino a 15 metri di altezza, si riconosce dai suoi aghi grigio-verdi, sottili, che possono essere lunghi fino a 10 cm e sono raggruppati in due sui germogli. Le piccole pigne bruno-rossastre sono simili a quelle di altri pini silvestri. La sua resina ha sentori molto caratteristici ed è utilizzata ampiamente sin dall’antichità.’, **superpoteri:** ‘anidride carbonica assorbita e salvaguardia’, **benefici:** ‘sicurezza alimentare e sviluppo economico’, **usi locali:** ‘medicina e suolo’, **paese:** ‘Perù’, **categoria:** ‘rischio estinzione’; decide, dopo aver letto le informazioni, di acquistare l’albero, per fare ciò aggiunge l’albero al carrello. Viene mostrata la pagina del carrello dove può vedere il riepilogo dei prodotti presenti nel carrello. Decide di procedere con l’acquisto e il sistema mostra la schermata di checkout che contiene un form per inserire il metodo di pagamento;

Antonio, inserisce i dati di pagamento, **tipo di carta:** ‘Visa’, **titolare carta:** ‘Antonio Avino’, **numero carta:** ‘1234 1234 1234 1234’, **data di scadenza:** ‘06/24’, **cvv:** ‘123’ e conferma l’acquisto. Il sistema, una volta validati i dati inseriti, mostra all’utente la conferma di ordine andato a buon fine. Il sistema salva l’ordine e attende che il responsabile degli ordini assegna l’albero a un contadino.

Il responsabile scegli di assegnare l’ordine di Antonio al contadino Pedro.

Pedro accede al suo profilo, e vedere che ha un nuovo albero da piantare, prende visione delle informazioni dell’albero, **specie:** ‘Pino’, **data di assegnazione:** ‘8/10/22’ e si adopera per piantare l’albero. Dopo aver finito la piantumazione, scatta una foto e aggiorna lo stato dell’albero compilando il form (già descritto in precedenza).

Il responsabile degli ordini accede all’elenco degli ordini da controllare per emettere il pagamento e selezione l’albero di ‘Pino’ piantato da Pedro, controlla la foto ed effettua la convalida

della piantumazione. Ora può emettere il pagamento compilando il form: **nome del titolare del conto:** ‘Pedro Garcia’, **codice SWIFT:** ‘UNCRITMM789’, **totale:** ‘300,00€’, **causale:** ‘Sostentamento per la crescita dell’albero.’.

Pedro riceverà i soldi, dal sistema, per sostentare il nuovo albero fino a quando non produrrà i frutti.

Antonio, passati alcuni giorni, decide di controllare i progressi del suo nuovo albero di ‘Pino’; perciò, effettua nuovamente il login al sito e accede al suo profilo. In particolare, visualizza la sezione dove sono presenti tutti i suoi alberi e vede che sono presenti due alberi: ‘Banano’ (già descritto in precedenza) e ‘Pino’.

Antonio seleziona l’albero di ‘Pino’ e legge gli aggiornamenti inseriti dal contadino, descritti precedentemente.

Antonio avendo acquistato due alberi decide di creare una foresta, il sistema gli mostra il form di creazione della foresta, **nome foresta:** ‘Breaking Forest’, **foto foresta:** ‘34433.jpg’, **alberi che vuoi facciano parte di questa foresta:** ‘pino e banano’, **tipologia di foresta:** ‘pubblica’. Il sistema valida i dati inseriti e crea la foresta di Antonio. D’ora in avanti la foresta di Antonio avrà una sua pagina dedicata e sarà visibile anche ad altri utente che potranno decidere di piantare i loro alberi nella foresta, gli altri utenti potranno raggiungere la foresta effettuando una ricerca.

## **S2 - Utente non registrato**

Salvatore ha ricevuto in regalo da Antonio un Treecode per riscattare un’albero sulla piattaforma WoodLot, per fare ciò dalla home preme per riscattare i codici e il sistema mostra un form, dove Salvatore inserisce il **codice ricevuto via e-mail:** ‘TRCD34434343’, ma non essendo registrato sulla piattaforma il sistema mostra un messaggio di errore che gli comunica di non poter completare l’operazione perché non registrato e lo reindirizza al form di registrazione. Il form contiene vari campi, **nome:** ‘Salvatore’, **cognome:** ‘Di Sanza’, **data di nascita:** ‘13/10/1998’, **e-mail:** ‘[salvatore@gmail.com](mailto:salvatore@gmail.com)’, **password:** ‘SaLv?’, **conferma password:** ‘SaLv?’.

Il sistema controlla che i dati inseriti dall’utente siano corretti, ma la password non rispetta il formato e il sistema mostra un messaggio di errore, per comunica che la lunghezza della password deve essere di minimo otto carattere e contenere almeno un carattere speciale, al di sotto del campo Password. Salvatore, può correggere i due campi inserendo una nuova password: **password:** ‘SaLv456?’, **conferma password:** ‘SaLv456?’.

Il sistema controlla che i dati inseriti dall’utente siano corretti, quindi, regista l’utente e mostra nuovamente il form per riscattare il regalo sul quale si trovava in precedenza. Salvatore inserisce il codice e questa volta l’operazione avviene con successo, mostrando al un messaggio di avvenuta riscossione del regalo.

L’utente accede al suo profilo, dove può visionare le informazioni dell’albero appena riscattato (già descritti in precedenza).

Salvatore decide di piantare l’albero nella foresta di Antonio: ‘Breaking Forest’, in questo modo anche se in modo virtuale i loro alberi potranno crescere insieme. Quindi clicca sul barra di ricerca, dove digita il nome della foresta: ‘Breaking Forest’ e il sistema mostra l’elenco dei risultati. L’utene seleziona il risultato che aveva cercato e il sistema mostra la pagina della foresta, ora Salvatore visualizza i dati della foresta: **nome:** ‘Breaking Forest’, **il numero di alberi piantati nella foresta:** ‘2’ e il **numero di custodi:** ‘1’. Salvatore può inserire il suo albero nella foresta compilando il form inserendo l’albero che vuole inserire, **nome:** ‘Baobab’. Il sistema salva l’aggiunta alla foresta e l’utente visualizza il suo albero all’interno della foresta.

## **S3 - Contadino non registrato**

Pedro è un contadino peruviano e vorrebbe registrarsi, per fare ciò accede al form di registrazione tramite la pagina dedicata ai contadini. Il form contiene vari campi: **nome:** ‘Pedro’, **cognome:** ‘Garcia’, **e-mail:** ‘[pedrogarcia@gmail.com](mailto:pedrogarcia@gmail.com)’, **password:** ‘!brew456?’, **conferma password:** ‘SLv456?’, **data di nascita:** ‘10/09/88’, **stato in cui pianterò gli alberi:** ‘Perù’, **nome del titolare del conto:** ‘Pedro Garcia’, **codice SWIFT:** ‘UNCRITMM789’. Il sistema valida i dati inseriti, ma il campo ‘Password’ e il campo ‘Conferma password’ non coincidono; quindi, mostra un messaggio di errore al di sotto del campo Conferma password. Pedro inserisce nuovamente il campo **password:** ‘!brew456?’, **conferma password:** ‘!brew456?’, il sistema controlla che i dati inseriti dal contadino siano corretti, quindi, regista il contadino.

# Requisiti funzionali

## ***Gestione ospite***

|  |  |  |
| --- | --- | --- |
| ID | Nome | Descrizione |
| RF\_GO\_01 | Registrazione | Il sistema deve permettere all’ospite di registrarsi. |

## ***Gestione utente***

|  |  |  |
| --- | --- | --- |
| ID | Nome | Descrizione |
| RF\_GU\_01 | Login | Il sistema deve permettere all’utente di autenticarsi. |
| RF\_GU\_02 | Logout | Il sistema deve permettere all’utente di effettuare il logout. |
| RF\_GU\_03 | ReimpostaPassword | Il sistema deve permettere il cambio password all’utente. |
| RF\_GU\_04 | VisualizzazioneProfilo | Il sistema deve consentire all’utente di visualizzare la pagina personale. |
| RF\_GU\_05 | ModificaDatiPersonali | Il sistema deve consentire all’utente di modificare i dati personali. |
| RF\_GU\_06 | VisualizzazioneCatalogo | Il sistema deve consentire all’utente di visualizzare il catalogo. |
| RF\_GU\_07 | AggiungereProdottoCarrello | Il sistema deve permettere all’utente di aggiungere un prodotto al carrello. |
| RF\_GU\_08 | ModificareQuantitaCarrelo | Il sistema deve permettere all’utente di modificare la quantità di un prodotto nel carrello. |
| RF\_GU\_09 | SvuotareCarrello | Il sistema deve permettere all’utente di svuotare il carrello. |
| RF\_GU\_10 | RimuovereProdottoCarrello | Il sistema deve permettere all’utente di rimuovere un prodotto dal carrello. |
| RF\_GU\_11 | EffettuareOrdine | Il sistema deve consentire all’utente di effettuare un ordine. |
| RF\_GU\_12 | RegalareAlgero | Il sistema deve consentire all’utente di rendere un prodotto del carrello un regalo, in un ordine ci possono essere più regali  (il regalo deve essere formato da un solo prodotto). |
| RF\_GU\_13 | VisualizzazioneStoricoOrdini | Il sistema deve permettere all’utente di visualizzare lo storico dei suoi ordini. |
| RF\_GU\_14 | VisualizzazioneAlberi | Il sistema deve permettere all’utente di visualizzare i suoi alberi. |
| RF\_GU\_15 | RiscattareTreecode | Il sistema deve permettere all’utente di riscattare un Treecode (codice regalo). |
| RF\_GU\_16 | CreazioneForesta | Il sistema deve consentire all’utente, che possiede almeno due alberi, di creare una foresta. |
| RF\_GU\_17 | AggingereAlberoNellaForesta | Il sistema deve consentire all’utente di aggiungere un albero in una foresta. |
| RF\_GU\_18 | ModificaDatiForesta | Il sistema deve permettere all’utente di modificare i dati della propria foresta. |
| RF\_GU\_19 | ForestaPubblica | Il sistema deve impedire all’utente di modificare una foresta pubblica in una foresta privata. |
| RF\_GU\_20 | Ricerca | Il sistema deve permettere all’utente di ricercare un prodotto, un utente o una foresta. |

## ***Gestione contadino***

|  |  |  |
| --- | --- | --- |
| ID | Nome | Descrizione |
| RF\_GC\_01 | ModificaDatiPersonali | Il sistema deve consentire al contadino di modificare i dati personali |
| RF\_GC\_02 | AggiornamentoStatoAlbero | Il sistema deve aggiornare lo stato di un albero in base ai dati inseriti dal contadino. |
| RF\_GC\_03 | VisualizzazioneAlberi | Il sistema deve permettere al contadino di visualizzare l’elenco degli alberi che gli sono stati assegnati. |

## ***Responsabile del catalogo***

|  |  |  |
| --- | --- | --- |
| ID | Nome | Descrizione |
| RF\_RC\_01 | VisualizzazioneCatalogo | Il sistema deve consentire al gestore del catalogo di visualizzare i prodotti presenti nel catalogo. |
| RF\_RC\_02 | InserireProdotto | Il sistema deve consentire al gestore del catalogo di inserire un prodotto nel catalogo. |
| RF\_RC\_03 | ModificareProdotto | Il sistema deve consentire al gestore del catalogo di modificare un prodotto del catalogo. |
| RF\_RC\_04 | EliminareProdotto | Il sistema deve consentire al gestore del catalogo di eliminare un prodotto dal catalogo. |
| RF\_RC\_05 | Ricerca | Il sistema deve permettere al gestore del catalogo di ricercare un prodotto. |

## ***Responsabile ordini***

|  |  |  |
| --- | --- | --- |
| ID | Nome | Descrizione |
| RF\_RO\_01 | AssegnazioneAlberi | Il sistema deve suggerire al responsabile ordini la migliore assegnazione. |
| RF\_RO\_02 | RiassegnazionaAlberoNonPiantato | Il sistema deve consentire al responsabile ordini di riassegnare un albero, che dopo una settimana non è stato ancora piantato. |
| RF\_RO\_03 | PagamentoContadino | Il sistema deve permettere al responsabile ordini di effettuare il pagamento ai contadini, dopo aver dato la conferma della piantumazione. |
| RF\_RO\_04 | VisualizzazioneElencoCotadini | Il sistema deve permettere al responsabile ordini di visualizzare l’elenco dei contadini. |
| RF\_RO\_05 | VisualizzazioneElencoOrdini | Il sistema deve permettere al responsabile ordini di visualizzare l’elenco degli ordini. |

# Requisiti non funzionali

## ***Usabilità***

|  |  |  |
| --- | --- | --- |
| ID | Nome | Descrizione |
| RNF\_US\_01 | Usabilità | Il sistema deve avere un filtro che gestisca la visualizzazione del catalogo. |
| RNF\_US\_02 | Usabilità | Il sistema deve avere delle FAQ d’utilizzo. |
| RNF\_US\_03 | Usabilità | Il sistema deve avere un menu contestuale per facilitare l’interazione. |
| RNF\_US\_04 | Usabilità | Il sistema deve fornire un feedback visuale all’utente dopo le operazioni. |

## ***Affidabilità***

|  |  |  |
| --- | --- | --- |
| ID | Nome | Descrizione |
| RNF\_RE\_01 | Affidabilità | Il sistema deve prevedere l’autenticazione tramite e-mail e password. |
| RNF\_RE\_02 | Affidabilità | Il sistema deve criptare le password e salvarle nel DB. |
| RNF\_RE\_03 | Affidabilità | Il sistema deve rispondere a input non validi. |
| RNF\_RE\_04 | Affidabilità | Le password devono rispettare dei canoni di sicurezza, al fine di non essere facilmente individuabili. |
| RNF\_RE\_05 | Affidabilità | Il sistema deve utilizzare un protocollo di navigazione sicuro HTTPS. |
| RNF\_RE\_06 | Affidabilità | Il sistema non deve permettere l’accesso in aree riservate ad utenti non autorizzati. |

## ***Supportabilità***

|  |  |  |
| --- | --- | --- |
| ID | Nome | Descrizione |
| RNF\_SU\_01 | Sostenibilità | Il sistema deve avere un’architettura in tre livelli con separazione di presentazione, application logic e data layer. |
| RNF\_SU\_02 | Sostenibilità | Il sistema deve adottare un’architettura modularizzabile per migliorare la manutenzione. |
| RNF\_SU\_03 | Sostenibilità | Il sistema deve supportare l’utilizzo su diversi dispositivi. |

## ***Implementazioni***

|  |  |  |
| --- | --- | --- |
| ID | Nome | Descrizione |
| PR\_IM\_01 | Implementazioni | Il sistema deve essere web-based, accessibile da qualsiasi dispositivo che sia connesso ad Internet. |
| PR\_IM\_02 | Implementazioni | L’implementazione deve avvenire tramite l’utilizzo del framework di Java Enterprise: Spring-Boot e l’utilizzo di Database relazionali. |
| PR\_IM\_03 | Implementazioni | Le dipendenze di libreria del sistema e il running del sito dovranno essere gestiti tramite l’ausilio di un tool di building come Maven. |
| PR\_IM\_04 | Implementazioni | La grafica del sito sarà sviluppata tramite il framework Bootstrap. |
| PR\_IM\_05 | Implementazioni | Gli aspetti dinamici dell’interfaccia saranno gestiti tramite l’ausilio del linguaggio Java-script. |
| PR\_IM\_06 | Implementazioni | La logica core del front-end sarà gestita tramite i linguaggi HTML, CSS e JSP. |

## ***Interfaccia***

|  |  |  |
| --- | --- | --- |
| ID | Nome | Descrizione |
| PR\_IN\_01 | Interfaccia | La gestione della sessione deve consentire lo scambio dati fra logica di business e front-end. |
| PR\_IN\_02 | Interfaccia | L’approccio classico di query sul DB deve essere applicato per l’interfaccia tra logica di business e logica persistente. |

## ***Packaging***

|  |  |  |
| --- | --- | --- |
| ID | Nome | Descrizione |
| PR\_PA\_01 | Packaging | Il DB deve essere installato tramite script SQL dedicati. |
| PR\_PA\_02 | Packaging | L’utilizzo del sistema necessita di una connessione ad internet. |

## ***Legali***

|  |  |  |
| --- | --- | --- |
| ID | Nome | Descrizione |
| PR\_LE\_01 | Legali | Il sistema deve garantire le norme sulla privacy secondo il GDPR 2016/679. |
| PR\_LE\_02 | Legali | Il sistema deve avere una politica dei cookie. |

# Target Environment

L’applicazione è sviluppata e distribuita su tre livelli:

* Database: verrà utilizzato un database relazionale per il salvataggio e l’utilizzo dei dati.
* Server: verrà utilizzato un server Tomcat per il caricamento del sito web.
* Client: un qualsiasi dispositivo dotato di browser web.

Altre tecnologie utilizzate per la creazione del sito web:

* Apache Tomcat, Apache Maven
* JDK, Angular CLI
* IntelliJ IDEA
* JavaScript, AJAX, JSON
* MySQL, MySQL Workbench, JDBC
* Spring-Boot
* HTML, CSS, Bootstrap, Spring Framework

# Deliverable & deadline

|  |  |
| --- | --- |
| **Problem Statement:** | 19 ottobre 2022 |
| **Requirements Analysis Document:** | 16 novembre 2022 |
| **System Design Document:** | 30 novembre 2022 |
| **Object Design Document:** | 11 gennaio 2023 |
| **Text plan:** | 11 gennaio 2023 |
| **Text case specification:** | 11 gennaio 2023 |