



Laurea Triennale in informatica-Università di Salerno
Corso di *Ingegneria del Software*- Prof. Gravino

Statement of Work

Progetto

StellarLibrary

Riferimento	
Versione	1.0
Data	09/01/2026
Destinatario	Studenti di Ingegneria del Software 2025/26
Presentato da	Christian Antonio Genovese, Biagio Radino, Riccardo Odierna, Antonio Plumitallo
Approvato da	



Revision History

Data	Versione	Descrizione	Autori
05/10/2025	0.1	Prima stesura	Christian Antonio Genovese, Biagio Radino, Riccardo Odierna, Antonio Plumitallo



Statement of Work (SOW) del Progetto

StellarLibrary

1. Scopo del Sistema

Negli anni la Libreria di Salerno, Stellar Library, ha costruito un rapporto di fiducia fatto di consigli, abitudini e percorsi di lettura; oggi intende estendere quell'esperienza oltre gli orari e gli spazi del negozio, rendendo il catalogo consultabile quando serve e valorizzando il confronto tra lettori come aiuto alla scelta dei libri. In questa prospettiva, la consultazione non si esaurisce più tra gli scaffali: novità, collane e autori restano a disposizione con continuità, così che la scoperta di un libro possa avvenire anche fuori sede, senza rinunciare all'attenzione che caratterizza la libreria.

L'obiettivo del progetto è fornire una piattaforma nella quale visitatori, clienti registrati e amministratori interagiscono naturalmente ed efficacemente tra di loro.

Il sistema deve supportare:

- Navigazione e ricerca del catalogo con schede prodotto complete di metadati essenziali (titolo, autore, genere, prezzo, descrizione), possibilità di visualizzare catalogo libri filtrato e ricercare libri specifici con titolo.
- Registrazione, autenticazione e gestione profilo, con storici ordini, come base per abilitare funzionalità riservate agli utenti loggati, inclusa la pubblicazione di commenti e l'assegnazione di valutazioni numeriche ai libri.
- Carrello accessibile a tutti gli utenti, nel quale è possibile inserire e rimuovere libri, e visualizzare l'importo totale.
- Checkout con prezzo totale, selezione metodi di pagamento/spedizione e conferma ordine accessibile solo agli utenti autenticati.
- Funzioni utente autenticato per commenti e valutazioni dei libri, con possibilità di:
 - Inserire un commento e una valutazione numerica relativo a un libro, visualizzabile da tutti nella scheda relativa a esso.
- Area amministrativa per:
 - Gestione catalogo (aggiunta e rimozione libro).
 - Riepilogo centralizzato di tutti gli ordini, con importo e metodo di pagamento e spedizione.
 - Riepilogo centralizzato dei commenti e delle valutazioni relative a tutti i libri.
 - Moderazione dei commenti e rimozione di essi.



2. Data di Inizio e di Fine

Inizio: ottobre 2025

Fine: gennaio-Febbraio 2026. È possibile concordare la data di consegna che potrà essere una delle seguenti:

- I: circa metà Gennaio 2026

3. Deliverables

4. RAD, SDD, cenni ODD, Matrice di Tracciabilità, Test Plan, Test Case Specification, Test incident Report, Test Summary Report.

Vincoli collaborativi e comunicativi.

- Rispetto scadenze delle scadenze intermedie/di fine progetto **definite nello statement of work**
- Uso di sistemi di versioning - GitHub in particolare
- Utilizzo di un sistema di versioning, dove tutti i membri del team forniscono il loro contributo
- Utilizzo di tool di per la suddivisione dei task e attività (Trello o similare)
- Utilizzo di tool di comunicazione tracciabile (Slack)

Vincoli tecnici

Analisi e specifica dei requisiti

- Specifica di minimo 2 e massimo 4 scenari per ogni membro del team;
- Diagramma degli use
- Esattamente uno use case per ogni membro del team - i casi d'uso aggiuntivi non saranno valutati;
- Specifica dei requisiti funzionali e non funzionali relativi a uno use case selezionato.
- Specifica degli oggetti boundary, control e entity per gli use case specificati.
- Un sequence diagram ogni due membri del team - il sequence diagram deve far riferimento agli use case specificati.
- Specifica di un class diagram per team - (gli object diagram non verranno valutati).
- Almeno un activity diagram per sistema esistente o sistema proposto e almeno uno statechart diagram ogni due membri del team.

System Design

- Specifica dei design goal e analisi dei trade-off relativi ad almeno due coppie di design goal.



- Definizione dell'architettura del sistema, Strategie relative alla gestione dei dati

persistenti, Il flusso di controllo globale, Le politiche di controllo degli accessi

Object Design

- Solo individuazione di dove poter utilizzare due design pattern, indicandone l'obiettivo e come sarebbero implementati.
- Uso di UML;

Testing

- Ogni studente dovrà effettuare il testing di unità, tramite category partition, dei casi d'uso specificati.

5. Criteri di Accettazione/Acceptance Criteria

- Utilizzo appropriato di GitHub, che preveda il rispetto delle linee guida definite nel contesto del lab.
- Adeguato utilizzo del pull-based development, che preveda il rispetto delle linee guida definite nel contesto del lab.
- Adeguato utilizzo di un tool di comunicazione, che preveda il rispetto delle linee guida definite nel contesto del lab.
- Adeguato utilizzo di un tool per la gestione dei task, che preveda il rispetto delle linee guida definite nel contesto del lab.
- Documentazione adeguata. Verranno usati tool di plagiarism detection per identificare casi in cui gli studenti hanno copiato da progetti di anni precedenti e/o da altre fonti.
- Appropriato testing dei casi d'uso.

6. Criteri di premialità

- Uso adeguato di sistemi di build;
- Uso adeguato di un processo di continuous integration tramite Travis;
- Adozione di processi di code review;
- Uso adeguato di tool avanzati di testing (e.g., Mockito, Cobertura, etc.).