



Laurea Triennale in informatica-Università di Salerno  
Corso di Ingegneria del software – Prof. Gravino Carmine

# Statement of Work (SOW)



## MagazON Lab

<b>Riferimento</b>	nc34_magazonlab-SOW
<b>Versione</b>	v7
<b>Data</b>	30/01/2025
<b>Destinatario</b>	Studenti di Ingegneria del Software 2024/25
<b>Presentato da</b>	Battaglia Daniel, Vaiano Francesco, Gigante Ruben
<b>Approvato da</b>	Gravino Carmine
<b>GitHub</b>	<a href="#"><u>Link al repository</u></a>



Laurea Triennale in informatica-Università di Salerno  
Corso di Ingegneria del software – Prof. Gravino Carmine

## Team Members

Nome	Ruolo	Acronimo	E-mail
Daniel Battaglia	Team Member	DB	<a href="mailto:d.battaglia8@studenti.unisa.it">d.battaglia8@studenti.unisa.it</a>
Francesco Vaiano	Team Member	FV	<a href="mailto:f.vaiano3@studenti.unisa.it">f.vaiano3@studenti.unisa.it</a>
Ruben Gigante	Team Member	RG	<a href="mailto:r.gigante2@studenti.unisa.it">r.gigante2@studenti.unisa.it</a>

## Revision History

Data	Versione	Descrizione	Autore
07/10/2024	V1	Prima stesura	DB, FV, RG
11/11/2024	V2	Seconda stesura	DB, FV, RG
14/11/2024	V3	Terza stesura	DB, FV, RG
27/12/2024	V4	Quarta stesura	FV
11/01/2025	V5	Revisioni	RG
24/01/2025	V6	Inizio Revisioni Finali	RG
30/01/2025	V7	Revisione finale	DB, FV, RG



## Sommario

1. Situazione attuale.....	4
2. Obiettivi.....	4
3. Ambito del prodotto .....	5
3.1. Alcuni scenari.....	6
4. Data di Inizio e di Fine .....	7
5. Deliverables .....	7
6. Vincoli/constraints .....	8
6.1. Vincoli collaborativi e comunicativi .....	8
6.2. Vincoli tecnici.....	8
6.2.1. Analisi e specifica dei requisiti .....	8
6.2.2. System Design .....	9
6.2.3. Object Design .....	9
6.2.4. Testing .....	9
7. Criteri di accettazione/Acceptance Criteria – criteri che, se non rispettati, portano al fallimento del progetto .....	9
8. Criteri di premialità.....	10



## 1. Situazione attuale

**MagazON Lab** è una start-up italiana situata in Campania che ha l'incarico di gestire la logistica del magazzino per conto di Amazon. Attualmente, MagazON Lab ha identificato il bisogno di un software gestionale per migliorare l'organizzazione del magazzino, ottimizzare i flussi interni e garantire una gestione delle operazioni logistiche più fluida, senza necessità di aumentare il personale. L'interazione con i clienti finali non è di competenza del magazzino, che si basa esclusivamente sulle liste inviate da Amazon. Queste liste contengono informazioni sui prodotti in arrivo e sui prodotti già in deposito, e su quando ciascun prodotto deve essere rispedito: immediatamente o con una spedizione anticipata rispetto al previsto. La frequenza delle liste può variare, essendo inviate in tempo reale e anticipato.

## 2. Obiettivi

Il software gestionale di MagazON Lab è progettato per ottimizzare e monitorare i flussi di prodotti all'interno del magazzino, garantendo un controllo efficace delle operazioni quotidiane e offrendo un supporto logistico per migliorare la gestione e l'organizzazione interna. Gli obiettivi generali del sistema includono:

- **Gestione dei Prodotti e delle Categorie:** Creare una struttura chiara e organizzata per catalogare i prodotti in base alle categorie e alle caratteristiche rilevanti, semplificando così il processo di gestione e monitoraggio;
- **Gestione degli Arrivi e delle Spedizioni:** Tracciare i prodotti in ogni fase operativa, dal loro arrivo fino alla spedizione, con registrazione accurata delle date per arrivo, deposito, eventuali modifiche e spedizioni;
- **Gestione delle Liste:** Garantire la visualizzazione delle liste di prodotti che sono arrivati in magazzino in modo da permettere il corretto inserimento dei prodotti nel sistema;
- **Gestione delle Notifiche:** Consentire agli amministratori di inviare notifiche ai magazzinieri per eventuali avvisi e informazioni, garantire l'esatto invio della notifica, il corretto recapito e la capacità di visualizzarla



### 3. Ambito del prodotto

Per soddisfare le esigenze di MagazON Lab, il software gestionale deve fornire un set completo di funzionalità specifiche. Di seguito, sono riportati i dettagli operativi e tecnici per ciascun tipo di utente.

- **Gestione dei Prodotti e delle Categorie:** Ogni prodotto è catalogato in una categoria specifica e associato a dettagli come stato attuale e settore di appartenenza. Le funzionalità per i magazzinieri comprendono:
  - Inserimento di nuovi prodotti con tutte le informazioni necessarie, inclusa la data di arrivo;
  - Eliminazione di prodotti non più presenti o necessari;
  - Aggiornamento delle informazioni esistenti per ciascun prodotto, in modo da garantire una gestione accurata e tempestiva;
  - Visualizzazione dei prodotti in base alla categoria e alle loro caratteristiche, con la possibilità di visualizzare eventuali modifiche delle date di spedizione o di arrivo.
- **Gestione degli Arrivi e delle Spedizioni:** Il sistema permette di monitorare le quantità e la posizione dei prodotti in magazzino nelle seguenti fasi:
  - **Arrivi:** Ogni prodotto è registrato al momento dell'ingresso in magazzino, includendo origine, quantità e data di arrivo;
  - **Spedizioni:** Per i prodotti destinati all'uscita, il sistema traccia le date di spedizione effettiva o anticipate e registra la quantità spedita;
  - **Funzionalità per i Magazzinieri:** I magazzinieri possono monitorare i movimenti e aggiornare lo stato dei prodotti, avendo sempre una panoramica completa dei prodotti in arrivo;
  - **Gestione delle Liste:** Il sistema offre ad ogni amministratore una sezione per poter inserire, modificare, visualizzare ed eliminare le liste amazon. I magazzinieri avranno una propria sezione dove potranno solo visualizzarle;
  - **Funzionalità per i Magazzinieri:** Avranno il compito di visualizzare le liste inserite e poi aggiungere i prodotti che sono presenti sulla lista.



- **Gestione degli Utenti e Comunicazioni:** Gli utenti accedono al sistema tramite login, con ruoli differenziati per dipendenti e amministratori, ciascuno dotato di una sezione notifiche dedicata.
  - **Magazzinieri:** Hanno accesso alle operazioni quotidiane di gestione dei prodotti e possono ricevere notifiche sia dall'amministratore che dal sistema. Le notifiche possono riguardare aggiornamenti, promemoria o avvisi legati alle operazioni di magazzino;
  - **Amministratori:** Dispongono di accesso completo a tutte le funzionalità e possono inviare notifiche a tutti gli utenti per comunicazioni urgenti o aggiornamenti operativi.

### 3.1. Alcuni scenari

#### - **Scenario 1: Organizzazione dei Prodotti in Arrivo - Antonio (Magazziniere)**

Antonio riceve una lista di nuovi prodotti inviata da Amazon, con i dettagli di arrivo e le categorie specifiche. Inserisce i prodotti nel sistema gestionale, registra la data di arrivo e assegna i prodotti alle rispettive categorie. Una volta catalogati, aggiorna il sistema con i dettagli di settore per rendere i prodotti facilmente reperibili e pronti per il deposito.

#### - **Scenario 2: Monitoraggio delle Spedizioni - Marco (Magazziniere)**

Marco riceve una notifica sul sistema riguardante una lista di prodotti destinati a una spedizione urgente. Accede alla sezione dedicata alle spedizioni e visualizza i dettagli: i prodotti, le quantità richieste e la data di consegna prevista. Dopo aver raccolto i prodotti dai rispettivi settori del magazzino, Marco aggiorna il sistema per registrare che i prodotti sono stati prelevati. Infine, conferma la spedizione, e il sistema aggiorna automaticamente lo stato dei prodotti come "spediti".



- **Scenario 3: Inserimento di una Lista Amazon - Laura (Amministratrice)**

Laura, amministratrice del sistema, riceve un nuovo file contenente una lista di prodotti in arrivo da Amazon. Accede alla sezione delle liste, carica il file e verifica che tutti i dettagli (categorie, quantità e date di arrivo previste) siano corretti. Successivamente, assegna la lista ai magazzinieri per il trattamento.

- **Scenario 4: Revisione delle Categorie - Antonio (Magazziniere)**

Antonio si accorge che alcuni prodotti catalogati non corrispondono esattamente alla categoria assegnata. Accede alla sezione gestione prodotti, identifica quelli con categorie errate e aggiorna il sistema con i dati corretti.

## 4. Data di Inizio e di Fine

Inizio: Ottobre 2024

Fine: Gennaio-Febbraio 2025. E' possibile concordare la data di consegna che potrà essere una delle seguenti:

- I: circa metà Gennaio 2025
- II: fine Gennaio 2025
- III: prima decade di Febbraio 2025

## 5. Deliverables

- RAD, SDD, ODD, Matrice di Tracciabilità, Test Plan, Test Case Specification, Test incident Report, Test Summary Report, Manuale D'Uso, Manuale Installazione e ogni altro documento richiesto per lo sviluppo del sistema.



## 6. Vincoli/constraints

### 6.1. 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)

### 6.2. Vincoli tecnici

#### 6.2.1. Analisi e specifica dei requisiti

- Specifica di **minimo** 2 e **massimo** 4 scenari per ogni membro del team;
- Specifica di **minimo** 2 e **massimo** 4 requisiti funzionali e non funzionali per ogni membro del team;
- **Esattamente** uno use case per ogni membro del team - i casi d'uso aggiuntivi **non** saranno valutati;
- **Esattamente** un sequence diagram ogni due membri del team - i sequence diagram aggiuntivi **non** saranno valutati;
- **Esattamente** un diagramma a scelta tra statechart e activity diagram ogni due membri del team - ulteriori diagrammi **non** verranno valutati;
- Specifica di un class diagram per team - eventuali object diagram **non** verranno valutati.





### 6.2.2. System Design

- Specifica di **minimo** 2 e **massimo** 4 design goal per ogni membro del team.
- Definizione di **un diagramma** di decomposizione dei sottosistemi per team, con annessa descrizione e motivazione all'uso.
- Definizione di **un deployment diagram** per team, con annessa descrizione e motivazione all'uso.

### 6.2.3. Object Design

- Uso di **minimo** uno e **massimo** due design pattern per team (devono essere selezionati tra quelli presentati a lezione);
- Uso di UML;

### 6.2.4. Testing

- Ogni studente dovrà effettuare il testing di unità, tramite category partition, di **esattamente** un metodo di una classe sviluppata.
- Ogni studente dovrà effettuare il testing di sistema di **esattamente** una funzionalità del sistema sviluppato.

## 7. Criteri di accettazione/Acceptance Criteria – criteri che, se non rispettati, portano al fallimento del progetto

- Utilizzo appropriato di GitHub, che preveda il **rispetto delle linee guida definite nel contesto del primo lab**.
- Adeguato utilizzo del pull-based development, che preveda il **rispetto delle linee guida definite nel contesto del primo lab**.
- Adeguato utilizzo di Slack, che preveda il **rispetto delle linee guida definite nel contesto del secondo lab**.



- Adeguato utilizzo di Trello, che preveda il **rispetto delle linee guida definite nel contesto del secondo 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 test di unità di un metodo sviluppato, che preveda il **rispetto dei vincoli**.
- Appropriato test di sistema di una funzionalità del sistema sviluppato, che preveda il **rispetto dei vincoli**

## 8. Criteri di premialità

- Uso adeguato di sistemi di **build**.
- Uso adeguato di un processo di **continuous integration** tramite Travis.
- Uso adeguato di tool di controllo della qualità (ad esempio, **CheckStyle**).
- Adozione di processi di **code review**.
- Uso adeguato di tool avanzati di testing (e.g., **Mockito**, **Cobertura**, etc.).