



# Problem Statament Progetto

Car – Zone

Versione 0.3

21/10/2024

|   |                  |
|---|------------------|
| Progetto: Car-Zone                        | Versione: 0.3    |
| Documento: ProblemStatement_Car-Zone_V0.3 | Data: 21/10/2024 |

### Coordinatore del progetto:

| Nome                  | Matricola  |
|-----------------------|------------|
| Francesco Pio Cataudo | 0512116773 |

### Partecipanti:

| Nome                  | Matricola  |
|-----------------------|------------|
| Francesco Pio Cataudo | 0512116773 |
| Francesco Santoro     | 0512117079 |
| Francesco Pio Bottaro | 0512118180 |
| Errico Aquino         | 0512117730 |

# Revision History

| Data       | Versione | Descrizione                                  | Autori |
|------------|----------|--|--------|
| 10/10/2024 | 0.1      | Problem Statement                            | Team   |
| 11/10/2024 | 0.2      | Revisione                                    | Team   |
| 14/10/2024 | 0.3      | Revisione generale dei vari punti e del logo | Team   |

|   |                  |
|---|------------------|
| Progetto: Car-Zone                        | Versione: 0.3    |
| Documento: ProblemStatement_Car-Zone_V0.3 | Data: 21/10/2024 |

## Sommario

|   |    |
|---|----|
| Revision History.....   | 3  |
| 1.Problem Domain.....   | 5  |
| 1.1 Current System.....   | 5  |
| 1.2 Proposed System.....  | 5  |
| 2.Description of more scenarios .....                             | 6  |
| 3. Functional Requirements .....                                  | 7  |
| 4. Nonfunctional Requirements .....                               | 9  |
| 6.Vincoli/Constraints .....                                       | 10 |
| 6.1 Vincoli Collaborativi e Comunicativi.....                     | 10 |
| 7.Ambito del Sistema .....  | 11 |
| 8. Target Enviroment.....   | 12 |
| 8.1 Presentation Layer (Livello di Presentazione) .....           | 12 |
| 8.2 Business Logic Layer (Livello della Logica di Business) ..... | 12 |
| 8.3 Data Access Layer (Livello di Accesso ai Dati) .....          | 13 |
| 9.Deliverables & deadline.....                                    | 14 |
| 10.Criteri di Accettazione/Acceptance Criteria .....              | 15 |

# Statement of Work (SOW) del Progetto

## Car-Zone

---

### 1. Problem Domain

---

Car-Zone è progettato per migliorare l'efficienza nella gestione dell'inventario, delle vendite e delle relazioni con i clienti. Il sistema aiuta i concessionari a tenere traccia del parco auto, monitorare il processo di vendita e gestire le informazioni dei clienti, migliorando così la produttività e l'esperienza del cliente

---

#### 1.1 Current System

---

Attualmente, i concessionari auto si affidano a una varietà di strumenti e processi manuali per gestire le loro operazioni chiave. Non esiste un sistema centralizzato per gestire l'inventario, le vendite e le relazioni con i clienti. Gli strumenti utilizzati includono software di contabilità, fogli di calcolo per tenere traccia dell'inventario e comunicazioni manuali per il CRM.

Le problematiche che emergono sono:

- Gestione dell'inventario inefficiente: Non c'è sincronizzazione in tempo reale tra la disponibilità dell'inventario e le vendite, causando ritardi nell'aggiornamento delle informazioni e perdite di opportunità di vendita.
  - Tracciamento delle vendite frammentato: Le informazioni sulle vendite sono memorizzate in vari sistemi, impedendo una visione complessiva delle performance aziendali.
  - Mancanza di un sistema CRM: Le relazioni con i clienti sono gestite tramite strumenti non integrati come email e telefonate, rendendo difficile seguire i clienti in modo efficace o tracciare le interazioni passate.
  - Accesso limitato ai dati: Gli operatori dei concessionari non hanno accesso centralizzato alle informazioni critiche, rendendo difficile prendere decisioni informate in modo tempestivo.
- 

#### 1.2 Proposed System

---

Il sistema che si vuole realizzare sarà una piattaforma software per i concessionari auto, progettata per ottimizzare la gestione dell'inventario, tracciare le vendite e gestire le relazioni con i clienti (CRM). Il

|   |                  |
|---|------------------|
| Progetto: Car-Zone                        | Versione: 0.3    |
| Documento: ProblemStatement_Car-Zone_V0.3 | Data: 21/10/2024 |

sistema centralizzerà tutte le informazioni in un unico database e consentirà agli amministratori, e ai clienti di interagire in modo semplice e immediato. L'architettura *three-tier* garantirà la separazione tra interfaccia utente, logica di business e gestione dei dati, rendendo il sistema scalabile e manutenibile.

Il sistema proposto consentirà di:

- Gestire in tempo reale l'inventario delle auto.
- Tracciare e automatizzare il processo di vendita, dalla gestione del cliente fino alla conferma dell'acquisto.
- Facilitare il follow-up e il customer care tramite strumenti CRM integrati.

## 2.Description of more scenarios

### Scenario AS IS – Analisi del contesto e dello stato attuale

#### Scenario 1: Ricerca di un'auto

Marco, un giovane di 25 anni, si trova nella fase cruciale di scegliere la sua prima auto. Per lui, questo è un passo importante e significativo. Attualmente, Marco visita diversi concessionari fisici, cercando di ottenere informazioni su vari modelli e prezzi. Tuttavia, si sente sopraffatto dalla quantità di informazioni disponibili. Ogni concessionario offre veicoli e opzioni diverse, rendendo difficile il confronto diretto. Marco si ritrova a raccogliere dati da molteplici fonti online, come siti di comparazione e forum, il che si rivela frustrante e dispendioso in termini di tempo. La mancanza di un'unica piattaforma che centralizzi le informazioni lo fa sentire confuso e indeciso, aumentando la sua ansia nel prendere una decisione.

#### Scenario 2: Verifica dello stato dell'ordine

Giulia ha recentemente ordinato un'auto da un concessionario locale e attende con trepidazione la consegna. Purtroppo, dopo aver completato l'acquisto, non riceve alcun aggiornamento riguardo al suo ordine. Ogni giorno, si trova a controllare la sua email e a chiamare il concessionario per chiedere notizie. Questa incertezza la rende ansiosa e insoddisfatta, poiché teme che ci possano essere stati dei problemi con il suo ordine. La mancanza di comunicazioni chiare da parte del concessionario contribuisce a un'esperienza di acquisto poco soddisfacente, facendola sentire trascurata e non supportata.

### Scenari Visionari

#### Scenario 1: Registrazione e ricerca di un'auto

Luca, un giovane di 30 anni, decide di visitare il sito web Car-Zone dopo aver sentito parlare della sua interfaccia user-friendly e delle numerose funzionalità. Dopo un rapido processo di registrazione tramite un modulo online, crea il suo profilo. Una volta effettuato il login, Luca esplora la funzione di ricerca,

che gli consente di filtrare le auto disponibili in base al nome dell'auto. Grazie a questa funzionalità, riesce a trovare rapidamente l'auto dei suoi sogni. Accede a tutte le informazioni necessarie, comprese le specifiche tecniche, le foto. Questa esperienza semplificata lo fa sentire più sicuro nella sua decisione d'acquisto.

### **Scenario 2: Dettagli del veicolo**

Sara, una giovane professionista, naviga nel sito Car-Zone alla ricerca di un'auto nuova. Dopo aver visto un modello che l'ha colpita, decide di cliccare sul link per visualizzarne i dettagli. Qui, può accedere a una descrizione approfondita del veicolo, che include specifiche tecniche, e immagini di alta qualità. Questa chiarezza e trasparenza nelle informazioni le permettono di avere un'idea chiara del veicolo e di prendere una decisione informata senza fretta.

### **Scenario 3: Contatto con il concessionario**

Dopo aver trovato un'auto che le interessa particolarmente, Giulia decide di utilizzare il modulo di contatto presente sul sito Car-Zone per richiedere ulteriori informazioni. Compila il modulo con i suoi dati personali e specifica le sue domande riguardo al veicolo. Questa funzionalità non solo le offre un modo semplice per comunicare con il concessionario, ma le dà anche la tranquillità di sapere che qualcuno si occuperà della sua richiesta.

### **Scenario 4: Aggiornamenti sull'ordine**

Dopo aver completato con successo l'acquisto della sua auto, Luca accede nuovamente al sito Car-Zone per controllare lo stato dell'ordine. Trova una sezione dedicata agli aggiornamenti degli ordini, dove può visualizzare in tempo reale il progresso della sua pratica. Luca si sente informato e in controllo della situazione, il che riduce la sua ansia e aumenta la sua soddisfazione.

---

## 3. Functional Requirements

---

L'Utente:

- Registrazione utente
- Ricerca auto
- Consultazione catalogo
- Visualizzazione modello

Il Cliente:

- Login
- Logout
- Gestione account
- Modifica Account
- Visualizzazione dati personali

|   |                  |
|---|------------------|
| Progetto: Car-Zone                        | Versione: 0.3    |
| Documento: ProblemStatement_Car-Zone_V0.3 | Data: 21/10/2024 |

- Visualizzazione Storico Cliente
- Conferma Ordine
- Richiesta Preventivo
- Conferma Preventivo
- Eliminazione Account
- Gestione Ordine Cliente
- Gestione Preventivo Cliente

#### Il Consulente:

- Login
- Logout
- Gestione account
- Modifica Account
- Visualizzazione dati personali
- Visualizzazione Clienti
- Gestione Ordine Consulente
- Eliminazione Ordine
- Approvazione Ordine
- Stipulazione Preventivo
- Gestione Preventivo Consulente
- Visualizzazione Storico Consulente

#### L'Amministratore:

- Login
- Logout
- Gestione account
- Modifica Account
- Visualizzazione dati personali
- Aggiunta Auto al catalogo
- Modifica Auto del catalogo
- Eliminazione Auto dal catalogo
- Eliminazione Consulente



---

## 4. Nonfunctional Requirements

---

### Usabilità

- L'interfaccia utente deve essere semplice e intuitiva, in modo che gli utenti possano navigare il sistema senza la necessità di una formazione specifica. Il 90% dei dispositivi dotati di un browser web comune deve poter accedere e utilizzare il sistema senza problemi.

### Affidabilità

- Il sistema deve essere disponibile 24/7, tranne che in casi di manutenzione programmata o imprevisti. Deve inoltre garantire una protezione adeguata dei dati, utilizzando protocolli di autenticazione sicura per prevenire accessi non autorizzati.

### Prestazioni

- Il sistema deve essere in grado di gestire contemporaneamente fino a 1000 utenti attivi. Le notifiche via email devono essere inviate entro 30 secondi dalla presa in carico di una richiesta, nel 90% dei casi.

### Sostenibilità

- L'architettura del sistema deve essere scalabile per poter gestire un aumento progressivo del carico utente, e il codice deve essere scritto in modo riusabile per facilitare future modifiche e aggiornamenti.

### Implementazione

- Il sistema deve essere sviluppato utilizzando il linguaggio di programmazione Java e un database MySQL DB per la gestione dei dati. Per il front-end, si utilizzeranno HTML5, CSS, JavaScript e il framework Bootstrap.

### Supportabilità

- Il sistema prodotto dev'essere facilmente scalabile e manutenibile per gli sviluppatori che in futuro andranno ad apportare delle modifiche o a risolvere dei problemi.

|   |                  |
|---|------------------|
| Progetto: Car-Zone                        | Versione: 0.3    |
| Documento: ProblemStatement_Car-Zone_V0.3 | Data: 21/10/2024 |

---

## 6.Vincoli/Constraints

---

### 6.1 Vincoli Collaborativi e Comunicativi

---

- Rispetto scadenze delle scadenze intermedie/di fine progetto
- Budget/Effort non superiore a  $50 \cdot n$  ore dove  $n$  sono i membri del team (compresi PM)
- Uso di sistemi di versioning - GitHub in particolare
- utilizzo di un sistema di versioning, dove tutti i membri del team forniscono il loro contributo

---

## 7.Ambito del Sistema

---

L'obiettivo del progetto Car-Zone è quello di sviluppare una piattaforma innovativa che fornisca un supporto completo alle esigenze dei clienti durante il processo di acquisto di un'auto, migliorando significativamente la loro esperienza e soddisfazione. La piattaforma sarà progettata per rendere l'intero processo di acquisto più semplice, trasparente e interattivo, consentendo agli utenti di navigare tra le varie opzioni disponibili con facilità e convenienza.

La piattaforma permetterà:

- **A tutti gli utenti di esplorare e acquistare veicoli:** Grazie a un'interfaccia intuitiva e user-friendly, chiunque avrà la possibilità di cercare e selezionare il veicolo che meglio soddisfa le proprie esigenze. Gli utenti potranno filtrare le auto in base a diversi criteri, come marca, modello, prezzo e anno di produzione, rendendo il processo di scelta più mirato e personalizzato.
- **A chiunque di visualizzare l'intero parco auto** disponibile presso il concessionario: La piattaforma offrirà un catalogo completo delle auto in vendita, con dettagli approfonditi su ogni veicolo, comprese specifiche tecniche, fotografie di alta qualità e informazioni sullo stato del veicolo. Questo permetterà ai clienti di fare scelte informate e consapevoli, migliorando la loro fiducia nel processo di acquisto.
- **Ai clienti registrati di consultare lo storico degli ordini** e monitorare lo stato delle proprie transazioni: Gli utenti potranno accedere facilmente al proprio profilo per visualizzare gli ordini effettuati, le fatture e lo stato delle consegne. Questa funzionalità non solo aumenta la trasparenza, ma fornisce anche ai clienti la comodità di gestire tutte le informazioni relative ai loro acquisti in un unico luogo.

---

## 8. Target Enviroment

---

### 8.1 Presentation Layer (Livello di Presentazione)

---

Il Presentation Layer è responsabile dell'interazione con gli utenti e della presentazione delle informazioni. Le tecnologie utilizzate in questo livello includono:

- HTML/CSS: Per la struttura e lo stile delle pagine web.
- JavaScript: Per aggiungere interattività alle pagine.
- AJAX: Per aggiornare il contenuto della pagina senza ricaricarla.
- JQuery: Per semplificare le manipolazioni DOM e AJAX.
- JSP (JavaServer Pages): Per generare dinamicamente contenuti HTML dal server utilizzando Java.

Funzionalità:

- Dashboard interattiva per le vendite e l'inventario.
- Gestione dei veicoli e delle informazioni sui clienti tramite form e interfacce.

---

### 8.2 Business Logic Layer (Livello della Logica di Business)

---

Il Business Logic Layer è responsabile dell'elaborazione delle regole di business e della gestione dei dati. Le tecnologie utilizzate sono:

- Java: Il linguaggio principale per l'implementazione della logica di business.
- Servlet: Per gestire le richieste HTTP e determinare le risposte da inviare.
- JSP: Anche per la generazione dinamica di contenuti.

Funzionalità:

- Gestione dell'inventario: Aggiunta, modifica e rimozione dei veicoli.
- Gestione delle vendite: Tracciamento delle transazioni e calcolo delle commissioni.

- Gestione CRM: Tracciamento delle interazioni con i clienti

---

### 8.3 Data Access Layer (Livello di Accesso ai Dati)

---

Il Data Access Layer si occupa della gestione dei dati e delle operazioni di persistenza. Le tecnologie includono:

- MySQL: DBMS relazionale per l'archiviazione di dati.
- JDBC: Per interagire con il database MySQL.

Funzionalità:

- Gestione e memorizzazione delle informazioni sull'inventario.
- Tracciamento delle vendite e gestione delle relazioni con i clienti.

---

## 9.Deliverables & deadline

---

- Project Management: business case, charter, team contract, scope statement, WBS, schedule, PM Plan, cost baseline, status reports, final project presentation, final project report, lessons-learned report, e ogni altro documento richiesto per gestire il progetto.
- Di Prodotto: 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.

Inizio: Ottobre 2024

1. Formazione gruppi di progetto (invio al docente delle schede informative e della proposta di progetto): 2 ottobre 2024
2. Start-up progetto (creazione repository gitHub, invio del file excel con le informazioni su progetto e partecipanti, invito docente e tutor su repository gitHub, kick-off meeting): 7 ottobre 2024
3. Problem Statement: 14 ottobre 2024
4. Requisiti e casi d'uso: 28 ottobre 2024
5. Requirements Analysis Document: 11 novembre 2024
6. System Design Document: 25 novembre 2024
7. Specifica delle interfacce dei moduli del sottosistema da implementare (parte dell'Object Design Document): 16 dicembre 2024
8. Piano di test di sistema e specifica dei casi di test per il sottosistema da implementare: 16 dicembre 2024

Fine: Gennaio-Febbraio 2025.

---

## 10.Criteri di Accettazione/Acceptance Criteria

---

- 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.**