



SDD System Design Document

Management Penta Motor

SDD 2.0 – ultima modifica 09/12/2016

Destinatario del documento:
Top Manager – Prof.ssa F. Ferrucci

Presentato da:

Gianmarco Bassano
Andrea Russo

Ernesto Pesce
Carmelo Sottile

Angela Cesarano
Roberta Gesumaria

Approvato da:

Ivan Rizzo

Storia dei cambiamenti

Data	Versione	Cambiamenti	Autori
16/11/2016	0.1	Strutturazione documento ed inserimento delle sezioni relative all'introduzione e all'architettura del sistema corrente.	Tutti
22/11/2016	0.2	Stesura ed inserimento nel documento delle sezioni decomposizione in sottosistemi, mapping hw/sw, gestione dati persistenti.	Tutti
25/11/2016	0.3	Stesura ed inserimento nel documento delle sezioni controllo degli accessi e sicurezza, controllo flusso globale del sistema, condizione limite e servizi dei sottosistemi.	Tutti
28/11/2016	1.0	Revisione.	Roberta Gesumaria
09/12/2016	2.0	Revisione finale.	Tutti

Sommario

1. Introduzione.....	4
1.1 Obiettivi del sistema	4
1.2 Design goals	4
1.3 Definizione, acronimi e abbreviazione	9
1.4 Riferimenti	10
1.5 Panoramica	11
2. Architettura del sistema corrente	11
3. Architettura del sistema proposto.....	11
3.1 Panoramica.....	11
3.2 Decomposizione in sottosistemi	12
3.2.1 Decomposizione in Layer	12
3.2.2 Decomposizione in sottosistemi	13
3.2.3 Diagramma di Deployment	14
3.3 Mapping hardware/software	14
3.4 Gestione dati persistenti	15
3.5 Controllo degli accessi e sicurezza	19
3.6 Controllo flusso globale del sistema	19
3.7 Condizione limite	20
3.7.1 Start-up	20
3.7.2 Start-up (a seguito di un fallimento)	20
3.7.3 Terminazione	20
3.7.4 Fallimento	20
4. Servizi dei sottosistemi	21

1. Introduzione

1.1 Obiettivi del sistema

Lo scopo di questo sistema è realizzare un sistema software che permetta la gestione dei dati e delle attività dell'azienda automobilistica Penta Motor e che consenta una facile gestione dei veicoli che sono in possesso dell'azienda supportando le attività lavorative dei suoi dipendenti attraverso la creazione di un portale web accessibile anche da mobile.

Lo sviluppo di una piattaforma di questo tipo permetterebbe di realizzare uno strumento in grado di ottimizzare il lavoro del personale, in quanto sarà possibile sia velocizzare le procedure di vendita dei veicoli che avere un maggiore controllo su di essi.

La piattaforma dovrebbe quindi consentire una comunicazione più immediata ed efficace tra gli attori del sistema (i vari dipendenti, gestore).

Il sistema è progettato affinché un dipendente possa vendere i veicoli disponibili, prenotarli in modo da non renderli disponibili ad altri venditori, e avere una vista dei veicoli assegnati nelle varie sezioni in base allo stato, infatti avremo veicoli immediatamente disponibili alla vendita, veicoli in arrivo, veicoli già venduti o in lavorazione. Inoltre permette loro di ricercare dei veicoli attraverso dei filtri appositamente immessi come la marca, la potenza, il colore o attraverso una parola chiave, hanno la possibilità di modificare le informazioni relative ad ogni veicolo, nel caso in cui siano state inserite scorrettamente e visualizzare le statistiche di vendita personali. Infine, permette al gestore di inserire un nuovo veicolo specificando tutte le relative caratteristiche, visualizzare le statistiche di vendita dei dipendenti ed eliminare i veicoli in qualsiasi stato si trovino.

Dato che il sistema ha accesso a dati sensibili dei clienti, deve fornire un metodo di autenticazione sicuro in modo che i dati siano protetti da accessi fraudolenti.

Inoltre per una migliore usabilità, il sistema dovrà essere facile da apprendere ed intuitivo da utilizzare, deve consentire la navigazione agevole per la fruizione delle funzionalità da lui offerte, ridurre la documentazione utente al minimo e permettere un utilizzo agevole del sistema anche senza consultare la documentazione.

1.2 Design goals

Criteri di performance

- Tempo di risposta

Per la visualizzazione di una determinata lista di veicoli il tempo di risposta sarà di 3 secondi.

Per la visualizzazione della lista degli account il tempo di risposta sarà di 3 secondi.

Per l'aggiunta di un elemento nel database il tempo di risposta sarà di 5 secondi.

Per la modifica di un elemento nel database il tempo di risposta sarà di 5 secondi.

Per la ricerca il tempo di risposta sarà di 3 secondi.

- Memoria

La dimensione complessiva del sistema dipenderà dalla memoria utilizzata per il mantenimento del database.

Criteri di affidabilità

- Robustezza

Eventuali input non validi immessi dall'utente saranno opportunamente segnalati con un messaggio di errore.

- Affidabilità

I risultati visualizzati saranno attendibili. Per quanto riguarda i veicoli, rispecchieranno la reale situazione all'interno della concessionaria, per esempio saranno visibili nella sezione "Parco auto" solo i veicoli immediatamente disponibili e presenti fisicamente in concessionaria. Gli account visualizzati rappresenteranno solo i dipendenti che attualmente lavorano presso Penta Motor. Le statistiche di vendita, relative ad un determinato periodo di tempo, saranno calcolati in funzione all'intervallo di tempo indicato.

- Disponibilità

Una volta installato il sistema sarà disponibile per tutti i dipendenti presso la loro postazione di lavoro.

- Tolleranza ai guasti

Il sistema potrà subire guasti dovuti al sovraccarico del database con successivo fallimento. Per ovviare al problema, periodicamente sarà previsto un salvataggio dei dati sotto forma di codice SQL necessario per la rigenerazione del database.

- Security

L'accesso al sistema sarà garantito mediante una username e una password.

Criteri di costo

- Costi di sviluppo

È stimato un costo complessivo di 300 ore per la progettazione e lo sviluppo del sistema (50 ore per ogni membro del team).

Criteri di manutenzione

- Estendibilità

Il sistema potrà essere esteso ad altre concessionarie o altri ambiti commerciali se modificato adeguatamente. Inoltre per quanto riguarda l'utilizzo nell'ambito della concessionaria potranno essere aggiunti nuovi parametri riguardanti i veicoli, i dipendenti e i clienti.

- Adattabilità

Il sistema potrà essere utilizzato da qualsiasi concessionaria o altro ambito commerciale.

- Tracciabilità dei requisiti

La tracciabilità dei requisiti sarà possibile grazie ad una matrice di tracciabilità, che permetterà di retrocedere al requisito associato ad ogni parte del progetto. La tracciabilità sarà garantita dalla fase di progettazione fino al testing.

- Portabilità

Il sistema sarà portabile in quanto l'interazione avviene mediante un browser senza interazione con il sistema sottostante, c'è quindi indipendenza dal sistema operativo.

Criteri utenti finali

- Usabilità

Il sistema sarà molto semplice da apprendere senza la consultazione di documentazione associata. L'intuitività sarà garantita in quanto il sistema avrà una buona prevedibilità, cioè la risposta del sistema ad un'azione utente sarà corrispondente alle aspettative.

- Utilità

Il lavoro dell'utente verrà supportato nel miglior modo possibile dal sistema, infatti l'utente compirà le operazioni consentite senza il carico di lavoro che deriva dal realizzare gli stessi procedimenti svolti con mezzi cartacei.

	DESIGN GOAL	PRIORITA'	RIFERIMENTI	MOTIVAZIONE
PERFORMANCE	Tempo di risposta	Media	NRF - Prestazioni	<p>Per assicurare un tempo di risposta basso utilizzeremo un database relazionale, che permette un accesso veloce e concorrente ai dati.</p> <p>Per un recupero iniziale delle informazioni si avrà una leggera perdita di velocità che andrà poi a favore dell'efficienza nell'utilizzo del sistema al fine di rendere subito disponibili i dati.</p> <p>Nel nostro sistema, poiché, ogni richiesta del client viene validata controllando la validità di username e password, il tempo di risposta del sistema potrebbe aumentare</p>

				soprattutto in situazioni di carico elevato, ma è necessario effettuare tali controlli per garantire la sicurezza.
	Memoria	Bassa	Aspetti manageriali	Lo svantaggio nella scelta del database relazionale è che richiede più spazio di memoria rispetto ad un file system.
AFFIDABILITA'	Robustezza	Alta	NRF - Usabilità	Al fine di evitare che il sistema riceva input errati e l'utente si trovi di fronte a situazioni inaspettate.
	Affidabilità	Alta	RF – Gestione veicoli, Gestione statistiche, Gestione account	Al fine di garantire l'attendibilità dei dati.
	Disponibilità	Alta	NRF – Packaging	Al fine di permettere all'utente di lavorare dalla propria postazione di lavoro.
	Tolleranza ai guasti	Alta	Aspetti manageriali	Al fine di garantire la conservazione dei dati.
	Security	Alta	NRF - Affidabilità	Al fine di garantire la sicurezza dei dati.
COSTO	Costi di sviluppo	Media	Aspetti manageriali	Sono stati preferiti altri obiettivi, dando meno importanza ai costi, come specificato nelle altre "Motivazioni".
MANUTENZIONE	Estendibilità	Alta	NRF - Supportabilità	Al fine di rendere più semplice la fase di manutenzione, anche sostenendo maggiori costi di sviluppo, scriveremo codice chiaro e facilmente modificabile.

	Adattabilità	Alta	NRF - Supportabilità	(Motivazione di estendibilità).
	Tracciabilità dei requisiti	Alta	Aspetti manageriali	Al fine di garantire la consistenza.
	Portabilità	Media	NRF - Implementazione	Per dare la possibilità all'utente di accedere al sistema anche da casa.
UTENTI FINALI	Usabilità	Alta	NRF - Usabilità	Per far sì che l'utente non si trovi in situazioni di disagio abbiamo reso il sistema il più intuitivo possibile.
	Utilità	Alta	NRF - Usabilità	Al fine di semplificare il lavoro dell'utente abbiamo puntato molto sull'utilità.

1.3 Definizione, acronimi e abbreviazione

Definizioni

Parco Auto: è l'area dove le auto a disposizione sono visibili al pubblico.

Parametri veicolo: rappresentano le informazioni relative ad ogni veicolo:

1. **Marca;**
2. **Modello;**
3. **Categoria** (berlina, cabrio, city car, coupé, furgone, monovolume, station wagon, SUV);
4. **Numero telaio;**
5. **Colore;**
6. **Cilindrata;**
7. **Potenza (Cv);**
8. **Alimentazione;**
9. **Anno immatricolazione;**
10. **Numero porte;**
11. **Allestimento** (base, economy, full optional);
12. **Km;**
13. **Cambio** (automatico, manuale, semi-automatico);
14. **Classe emissione** (Euro 1-6);
15. **Usata** (si/no);
16. **Prezzo;**
17. **Stato:** rappresenta un'informazione associata ad un veicolo relativa alla sua situazione di vendita.
Può essere:
 - 17.1 **Disponibile:** nel caso in cui ci sia la possibilità dell'immediata fruizione del veicolo.
 - 17.2 **In arrivo:** nel caso in cui il veicolo sia già stato ordinato dal gestore e siamo solo in attesa del suo arrivo nel parco auto per poter essere venduta.
 - 17.3 **Venduto:** nel caso in cui il veicolo sia già stato acquistato da un cliente.
 - 17.4 **In lavorazione:** nel caso in cui il veicolo sia stato prenotato da un cliente e siamo solo in attesa del pagamento.
18. **Dipendente:** indica il dipendente che ha venduto il veicolo;
19. **Data vendita.**

Dipendente: rappresenta un qualsiasi addetto alle vendite, ognuno avente un proprio account in cui saranno specificate le seguenti informazioni:

1. **Codice fiscale;**
2. **Nome;**
3. **Cognome;**
4. **Data di nascita;**
5. **Indirizzo;**
6. **Email;**
7. **Numero di telefono;**
8. **Username** (generato automaticamente dal sistema con la seguente notazione: nome_cognome_dipx) con x=1,2,3...;
9. **Password;**
10. **Ruolo:** indica il ruolo del dipendente per distinguerlo dal gestore.

Gestore: rappresenta una specializzazione del concetto di dipendente con le funzionalità aggiuntive di poter visualizzare le statistiche periodiche, le statistiche dei vari dipendenti, inserire, modificare e eliminare un veicolo. Ha inoltre funzioni di gestione degli account.

Sezione: indica la suddivisione dei veicoli in base allo stato e può essere:

1. **Parco auto:** sezione in cui sono visibili i veicoli a disposizione.
2. **In arrivo:** sezione in cui sono visibili i veicoli ordinati dal gestore.
3. **In lavorazione:** sezione in cui sono visibili i veicoli venduti tramite finanziamento.
4. **Venduti:** sezione in cui sono visibili i veicoli venduti.

Cliente: rappresenta un generico acquirente di un veicolo. Al momento dell'acquisto dovranno essere specificati i seguenti parametri:

1. **Codice fiscale;**
2. **Nome;**
3. **Cognome;**
4. **Data di nascita;**
5. **Comune di nascita;**
6. **Indirizzo;**
7. **Email;**
8. **Numero di telefono.**

Acronimi

MPM = Management Penta Motor.

RAD = Requirement Analysis Document.

MVC = Model View Controller.

RF = Requisito Funzionale.

RNF = Requisito Non Funzionale.

SQL = Structured Query Language.

E – R = Entity Relationship.

DBMS = DataBase Management System.

NA = Nessuna.

Km = Chilometri.

NB = Nota Bene.

1.4 Riferimenti

- <http://www.pentamotor.it>
- Bernd Bruegge & Allen H. Dutoit, Object-Oriented Software Engineering: Using UML, Patterns and Java, (2nd edition), Prentice-Hall, 2004.
- Ian Sommerville, Ingegneria del software, (8^a edizione), Pearson, 2007.
- MPM_RAD_Vers3.0.

1.5 Panoramica

Al secondo punto del documento verrà presentato il sistema corrente. Al terzo punto verrà presentata l'architettura del sistema proposto in cui gestiremo la decomposizione in sottosistemi, il mapping hardware/software, i dati persistenti, il controllo degli accessi e sicurezza, il controllo del flusso globale del sistema, le condizioni limite. Al quarto punto verranno presentati i servizi dei sottosistemi.

2. Architettura del sistema corrente

Attualmente non esiste un sistema software che si occupa di gestire questa problematica, ossia l'amministrazione dei vari documenti che vengono utilizzati nella concessionaria Penta Motor. Essa, infatti, è gestita tramite documenti cartacei. Questo provoca numerosi disagi ai dipendenti e soprattutto uno spreco di risorse che potrebbero essere utilizzate diversamente. Quindi si tratta di un sistema che rientra nel campo della Greenfield Engineering. Infatti in un Greenfield Project lo sviluppo comincia da zero, non esiste nessun sistema a priori e i requisiti sono ottenuti dall'utente finale e dal cliente. Nasce, perciò, a partire dai bisogni dell'utente.

3. Architettura del sistema proposto

3.1 Panoramica

Il sistema da noi proposto è un'applicazione web per la concessionaria Penta Motor. L'obiettivo che si pone è quello di fornire uno strumento per la gestione interna della vendita e per la visione generale dei veicoli disponibili.

Il sistema è rivolto al gestore e ai dipendenti dell'autoconcessionaria. Ogni utente potrà accedervi attraverso una login. In base alla tipologia di utente (gestore o dipendenti) si avrà una visione diversa del sistema.

Si darà la possibilità di inserire nuovi veicoli con le relative caratteristiche, di avere una visione delle vendite per dipendente.

Altre funzioni che il sistema fornirà saranno legate alla vendita, quindi, la possibilità di effettuare ricerche specifiche sui veicoli, mostrare i veicoli disponibili, quelli in arrivo, venduti e quelli in lavorazione e modificare lo stato dei veicoli.

Il sistema verrà suddiviso in client e server, il client gestirà la parte di presentazione e la parte di logica direttamente connessa all'interfaccia grafica, il server invece gestirà la parte di logica relativa ai dati, che saranno salvati in un database salvato sul server.

La divisione in classi nei vari sottosistemi sarà individuata in base alle funzionalità per poter rendere massima la coesione e minimo l'accoppiamento tra i sottosistemi in modo che i cambiamenti in un sottosistema non influiscano sugli altri.

Le funzionalità saranno divise in layer logici in base alle differenti funzionalità utilizzando il modello MVC: sistema di memorizzazione (Model), presentazione (View) e business logic (Controller).

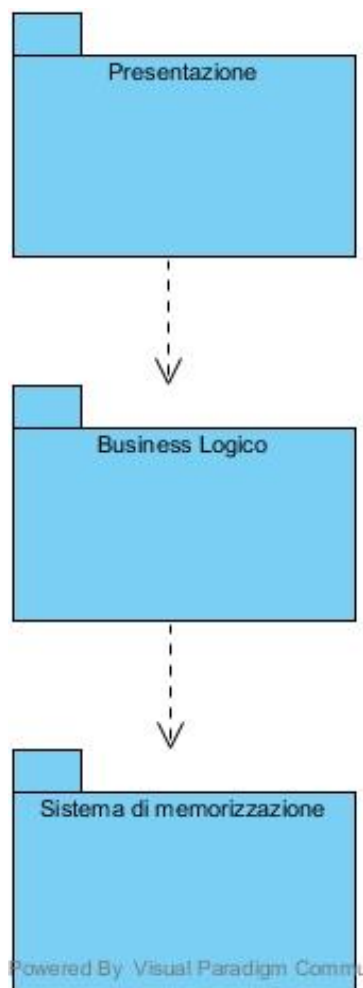
3.2 Decomposizione in sottosistemi

3.2.1 Decomposizione in Layer

E' stata adottata l'architettura MVC (Model, View, Controller) perché adatta per applicazioni web interattive come la nostra. Questo stile di architettura presenta una netta divisione tra interfaccia, rappresentato dal sottosistema View, controllo della gestione dei dati persistenti, rappresentato dal Model, e controllo della logica di business, rappresentato dal Controller.

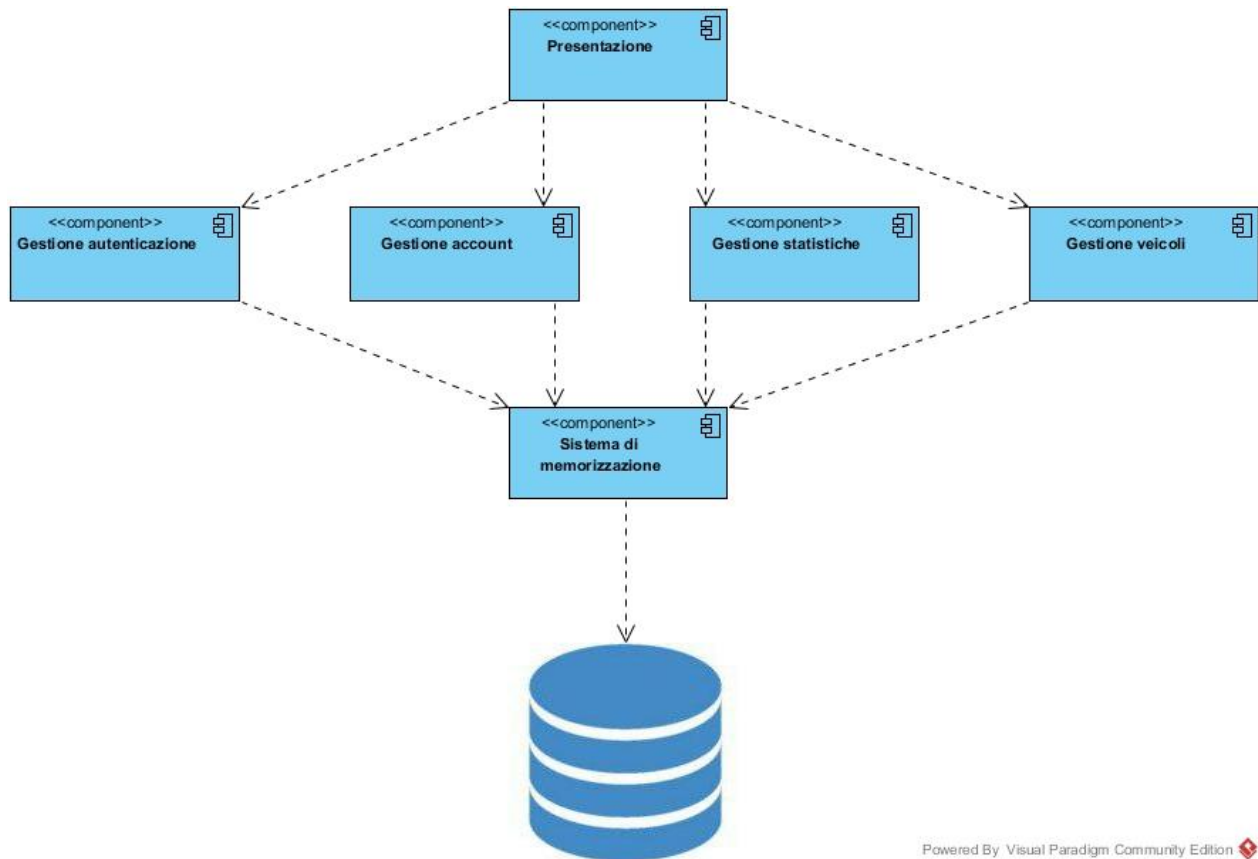
Decomponiamo, quindi, il sistema in tre layer che si occupano di gestire aspetti e funzionalità differenti:

- sistema di memorizzazione (Model): si occupa della gestione e dello scambio dei dati tra i sottosistemi;
- presentazione (View): raccoglie e gestisce l'interfaccia grafica e gli eventi generati dall'utente;
- business logico (Controller): si occupa della gestione della logica del sistema.



3.2.2 Decomposizione in sottosistemi

Dopo un'attenta e dettagliata analisi funzionale, abbiamo scelto di dividere il nostro sistema nel seguente modo in quanto, per la divisione in componenti, avevamo bisogno di un basso accoppiamento e un'elevata coesione tra i servizi offerti tra i componenti interni. Abbiamo deciso di suddividere le funzionalità per area di gestione e creato un'interfaccia intermedia tra i sistemi della logica di business e il database, in modo da non dover fare grossi cambiamenti nel caso in cui sia necessario cambiare database. Il sistema si compone di sei componenti che si occupano di gestire aspetti e funzionalità differenti:



Powered By Visual Paradigm Community Edition

Il livello Presentazione prevede un unico sottosistema:

- Presentazione: sistema che gestisce l'interfaccia grafica e gli eventi generati dall'interazione con il sistema.

Il livello Business Logico prevede a sua volta una suddivisione in 4 sottosistemi:

- Gestione autenticazione;
- Gestione account;
- Gestione statistiche;
- Gestione veicoli;

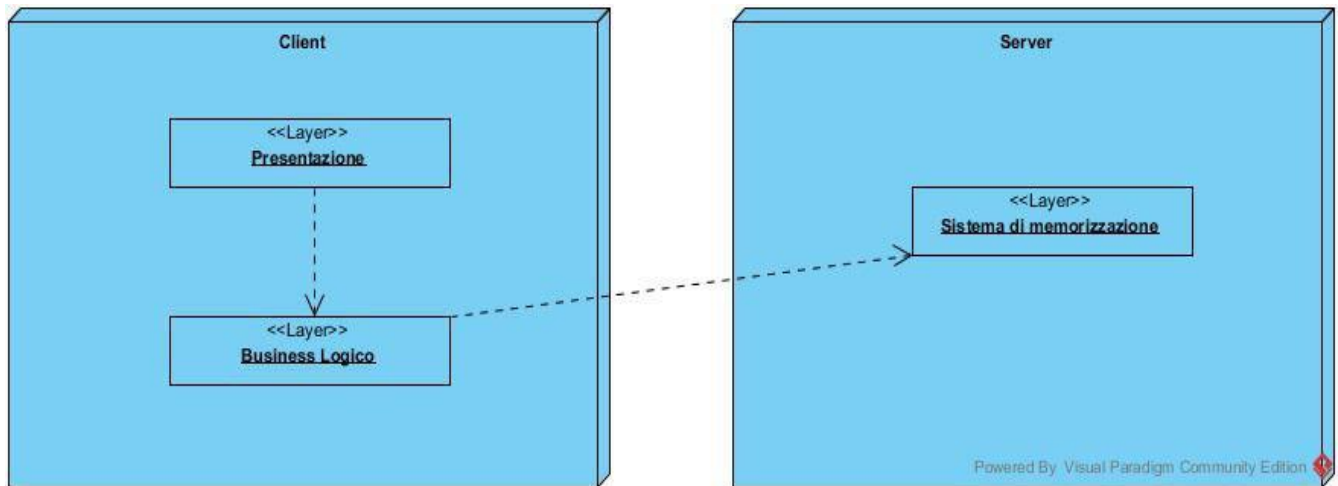
Gestione account fa riferimento alle funzionalità attive per il gestore mentre Gestione autenticazione, Gestione statistiche e Gestione veicoli inglobano le funzionalità accessibili anche al dipendente.

Il livello Sistema di memorizzazione prevede un unico sottosistema:

- Sistema di memorizzazione: sistema che gestisce ed immagazzina i dati persistenti.

3.2.3 Diagramma di Deployment

L'utente richiede le funzionalità tramite l'interfaccia che il sistema mette a disposizione. Il client ospita lo strato di presentazione e logica del sistema. La parte server gestisce la persistenza dei dati.



3.3 Mapping hardware/software

Per una migliore gestione, il sistema che si vuole sviluppare sarà diviso in client e server che saranno installati su Altrivista, per il DBMS verrà utilizzato PHPMyAdmin, mentre per l'interfaccia si usufruirà del framework Bootstrap.

Il client fornirà: interfacce utente, elaborazione al front-end delle informazioni, gestione dei dati centralizzata, integrità, consistenza e sicurezza dei dati e operazioni concorrenti (multiple access).

Il server fornirà: elaborazione centralizzata (archiviazione e interazione col database).

Su richiesta del cliente, successivamente, il client potrà essere separato dal server e installato sui dispositivi da lui richiesti.

3.4 Gestione dati persistenti

Per la gestione dei dati persistenti abbiamo deciso di utilizzare un database relazionale poiché le informazioni che andremo a memorizzare richiedono un accesso multiplo opportunamente gestito da questo tipo di memorizzazione.

Legenda priorità:

Obbligatorio	O
Trascurabile	T

Veicolo

PARAMETRO	PRIORITA'
Numero telaio	O
Marca	O
Modello	O
Categoria	O
Colore	O
Cilindrata	O
Potenza	O
Alimentazione	O
Anno immatricolazione	O
Numero porte	O
Allestimento	O
Km	O
Cambio	O
Classe di emissione	O
Usata	O
Prezzo	O
Stato	O
Dipendente	O
Cliente	O
Data vendita	O

Cliente





PARAMETRO	PRIORITA'
Codice fiscale	O
Nome	O
Cognome	O
Data di nascita	T
Comune di nascita	T
Indirizzo	T
Email	T
Numero di telefono	O

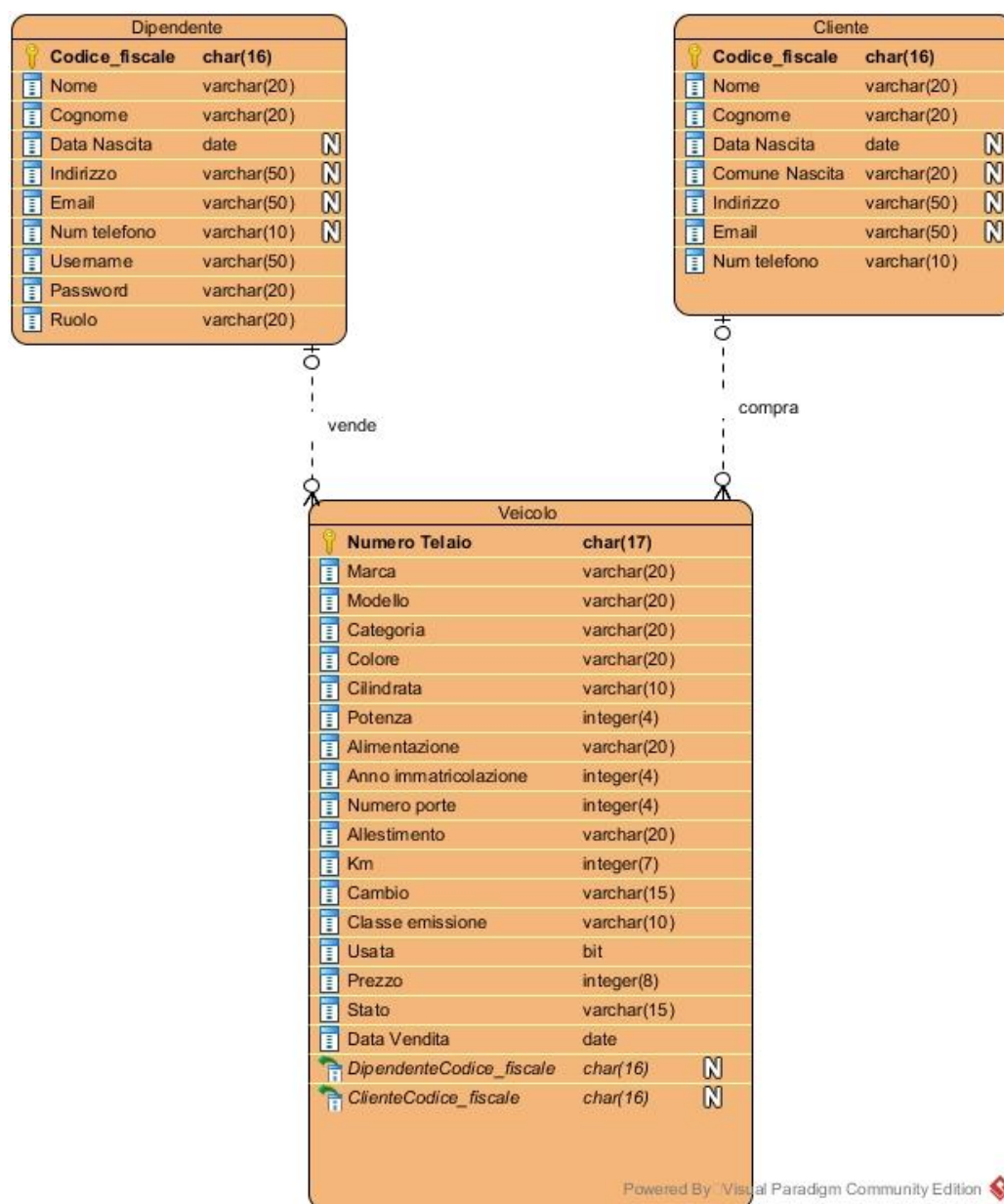
Dipendente

PARAMETRO	PRIORITA'
Codice fiscale	O
Nome	O
Cognome	O
Data di nascita	T
Indirizzo	T
Email	T
Numero di telefono	T
Username	O
Password	O
Ruolo	O

Schema E-R

Legenda:

	Colonna
	Chiave primaria
	Chiave esterna
	Nullable
+ --- ○	Relazione 1 a molti



Lista di attributi per ogni tabella

Veicolo

Numero telaio	
Marca	
Modello	
Categoria	
Colore	
Cilindrata	
Potenza	
Alimentazione	
Anno immatricolazione	
Numero porte	
Allestimento	
Km	
Cambio	
Classe emissione	
Usata	
Prezzo	
Stato	
Data vendita	
ClienteCodice_fiscale	
DipendenteCodice_fiscale	

Cliente

Codice fiscale	
Nome	
Cognome	
Data nascita	
Comune nascita	
Indirizzo	
Email	
Numero telefono	

Dipendente

Codice fiscale	
Nome	
Cognome	
Data nascita	
Indirizzo	
Email	
Numero telefono	
Username	
Password	
Ruolo	

Specifica delle caratteristiche degli attributi

Dipendente

Nome	Tipo	Null	Key
Codice fiscale	Char(16)	Not null	Primary key
Nome	Varchar(20)	Not null	
Cognome	Varchar(20)	Not null	
Data nascita	Date	Nullable	
Indirizzo	Varchar(50)	Nullable	
Email	Varchar(50)	Nullable	
Numero telefono	Varchar(10)	Nullable	
Username	Varchar(50)	Not null	
Password	Varchar(20)	Not null	
Ruolo	Varchar(20)	Not null	

Cliente

Nome	Tipo	Null	Key
Codice fiscale	Char(16)	Not null	Primary key
Nome	Varchar(20)	Not null	
Cognome	Varchar(20)	Not null	
Data nascita	Date	Nullable	
Comune nascita	Varchar(20)	Nullable	
Indirizzo	Varchar(50)	Nullable	
Email	Varchar(50)	Nullable	
Numero telefono	Varchar(10)	Not null	

Veicolo

Nome	Tipo	Null	Key
Numero telaio	Varchar(17)	Not null	Primary key
Marca	Varchar(20)	Not null	
Modello	Varchar(20)	Not null	
Categoria	Varchar(20)	Not null	
Colore	Varchar(20)	Not null	
Cilindrata	Integer(4)	Not null	
Potenza	Integer(4)	Not null	
Alimentazione	Varchar(20)	Not null	
Anno immatricolazione	Integer(4)	Not null	
Numero porte	Integer(4)	Not null	
Allestimento	Varchar(20)	Not null	
Km	Integer(7)	Not null	
Cambio	Varchar(15)	Not null	
Classe emissione	Varchar(10)	Not null	
Usata	Boolean	Not null	
Prezzo	Integer(8)	Not null	
Stato	Varchar(15)	Not null	
Data vendita	Date	Not null	
ClienteCodice_fiscale	Char(16)	Nullable	Foreign key
DipendenteCodice_fiscale	Char(16)	Nullable	Foreign key

NB Usata = 0 indica false, Usata = 1 indica true.

3.5 Controllo degli accessi e sicurezza

Il controllo degli accessi è garantito tramite l'utilizzo di username e password per il gestore, o dipendente, che verranno richieste per ogni singolo accesso a MPM.

La sicurezza sui dati sensibili è garantita dall'accesso controllato in quanto solo il gestore può avere accesso ai dati relativi agli acquirenti.

Le operazioni che l'utente del MPM può fare sono:

SOTTOSISTEMA	FUNZIONI	ATTORE	
		Gestore	Dipendente
Gestione Autenticazione	Login	√	√
	Logout	√	√
Gestione Account	Registra Account	√	X
	Modifica Account	√	X
	Elimina Account	√	X
Gestione Statistiche	Statistiche Personali	√	√
	Statistiche dipendenti	√	X
Gestione Veicoli	Inserisci veicolo	√	X
	Prenota veicolo	√	√
	Elimina veicolo	√	X
	Modifica veicolo	√	X
	Vendita veicolo	√	√
	Visualizza veicolo	√	√
	Ricerca veicolo con parola chiave	√	√
	Ricerca veicolo con filtri	√	√

3.6 Controllo flusso globale del sistema

Il sistema MPM fornisce funzionalità che richiedono una continua interazione da parte dell'utente, per questo motivo abbiamo scelto come meccanismo di controllo del flusso globale l'event-driven control.

3.7 Condizione limite

3.7.1 Start-up

Per il primo start-up del sistema MPM è necessario l'avvio di un web server che fornisca il servizio di un Database PHPMyAdmin per la gestione dei dati persistenti e l'interpretazione ed esecuzione del codice lato server. In seguito, tramite l'interfaccia di Login, sarà possibile autenticarsi tramite opportune credenziali (username e password) come utente con accesso alle funzionalità del sistema. Una volta effettuato l'accesso, MPM presenterà la home relativa al tipo di utente loggato, dalla quale si potranno effettuare tutte le operazioni che il sistema gli fornisce.

3.7.2 Start-up (a seguito di un fallimento)

Il sistema può subire guasti dovuti al sovraccarico del database con successivo fallimento. Per ovviare al problema, periodicamente è previsto un salvataggio dei dati sotto forma di codice SQL. All'avvio a seguito di tale fallimento, oltre alle normali procedure previste per lo start-up, l'ultimo codice SQL memorizzato sarà eseguito per la rigenerazione del database.

3.7.3 Terminazione

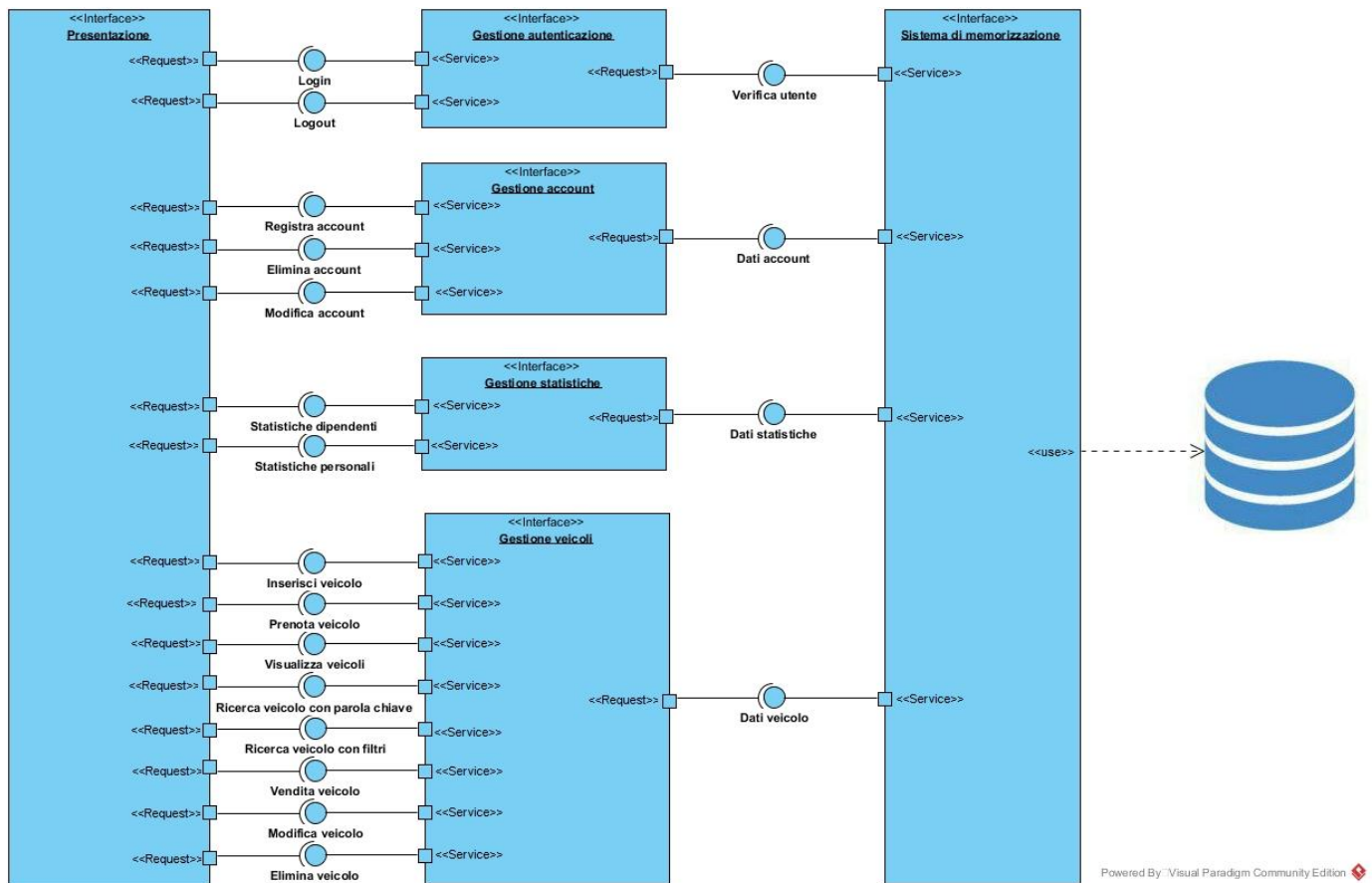
Al momento della chiusura dell'applicativo si ha la terminazione del sistema con un regolare Logout dal sistema. Viene assicurata la consistenza dei dati, annullando eventuali operazioni che erano in esecuzione.

3.7.4 Fallimento

Possono verificarsi diversi casi di fallimento del sistema:

- Nel caso di guasti dovuti al sovraccarico del database con successivo fallimento dello stesso, è prevista come procedura preventiva il salvataggio periodico dei dati sotto forma di codice SQL per la successiva rigenerazione del database.
- Nel caso in cui si verifichi un'interruzione inaspettata dell'alimentazione, non sono previsti metodi che ripristinino lo stato del sistema a prima dello spegnimento inaspettato.
- Un altro caso di fallimento potrebbe derivare dal software stesso che causa una chiusura inaspettata dovuta ad errori commessi durante la fase di implementazione, non sono previste politiche correttive, l'unico processo che potrà essere eseguito è la chiusura del sistema e il suo successivo riavvio.
- Un altro caso di fallimento potrebbe essere dovuto ad un errore critico nell'hardware, non è prevista alcuna misura correttiva.

4. Servizi dei sottosistemi



Presentazione: sistema che gestisce l'interfaccia grafica e gli eventi generati dall'interazione con il sistema.

Gestione autenticazione offre 2 servizi all'interfaccia di Presentazione:

- Login
- Logout

Gestione account offre 3 servizi all'interfaccia di Presentazione:

- Registra account
- Elimina account
- Modifica account

Gestione statistiche offre 2 servizi all'interfaccia di Presentazione:

- Statistiche dipendenti
- Statistiche personali

Gestione veicoli offre 8 servizi all'interfaccia di Presentazione:

- Inserisci veicolo
- Prenota veicolo
- Visualizza veicoli
- Ricerca veicolo con parola chiave



- Ricerca veicolo con filtri
- Vendita veicolo
- Modifica veicolo
- Elimina veicolo

Sistema di memorizzazione offre un unico servizio a Gestione autenticazione:

- Verifica utente

Sistema di memorizzazione offre un unico servizio a Gestione account:

- Dati account

Sistema di memorizzazione offre un unico servizio a Gestione statistiche:

- Dati statistiche

Sistema di memorizzazione offre un unico servizio a Gestione veicoli:

- Dati veicolo