

Università degli Studi di Salerno

Corso di Ingegneria del Software

**La Cantinella del Bonsignore
System Design Document**

Versione 1.2

LOGO PROGETTO



Data: 6/12/2017

Partecipanti

Nome	Matricola
Graziuso Catello	0512103726
Fornaro Daniele	0512103864
Ciano Francesco	0512103918

Indice

1 Introduzione

1.1 Scopo del sistema

1.2 Obbiettivi di design

1.2.1 Criteri di Performance

1.2.2 Criteri di Affidabilità

1.2.3 Criteri di Manutenzione

1.2.4 Criteri per l'utente finale

1.3 Definizione, Acronimi e Abbreviazioni

1.4 Riferimenti

1.5 Panoramica

2 Sistema corrente

3. Architettura del Sistema proposto

3.1 Panoramica

3.2 Decomposizione in sottosistemi

3.3 Mapping hardware software

3.4 Gestione dati persistenti

3.5 Controllo degli accessi e sicurezza

3.6 Controllo Globale del software

3.7 Condizione di boundary

3.7.1 Start-up

3.7.2 Terminazione

3.7.3 Fallimento

4 Servizi Dei Sottosistemi

4.1 Gestione Vini

4.2 Gestione utenti

4.3 Gestione Like

4.4 Gestione ordini

4.5 Gestione Carrello

5 GLOSSARIO

1. Introduzione

1.1.Scopo del sistema

Gli obiettivi principali del sistema sono quelli di realizzare un software che permette alle piccole imprese vinicole di espandere il proprio business inserendo i propri articoli su una piattaforma e-commerce online.

Il progetto “La Cantinella Del Bonsignore” propone quindi una piattaforma online dove gli utenti potranno scegliere se registrarsi come clienti , ed iniziare fin da subito ad acquistare vini, oppure come clienti azienda, o meglio, rappresentanti di azienda, ed iniziare a vendere i propri vini.

In entrambi i casi, ogni tipo di utente potrà accedere al catalogo del sito e comprare in poche e semplici operazioni la quantità di vino desiderata.

I Clienti azienda potranno inserire i propri prodotti in una sezione personale apposita, e all’occorrenza, modificare alcune caratteristiche di vini già inseriti.

Il software si propone di essere funzionale sia su desktop che su dispositivi mobili come tablet e smarthphone.

Inoltre punto fermo del software da implementare sarà un’interfaccia di facile comprensione da parte della clientela in quanto verrà considerato una completa inesperienza.

Il software non solo permetterà un possibile aumento dei guadagni da parte delle aziende che decideranno di esporre i propri vini sulla piattaforma ma anche la possibilità di crearsi un nome all’interno della comunità vinicola, infatti il software metterà a disposizione dei clienti un servizio che permetterà di lasciare “consensi” ai vini, così da accrescere una azienda ed aiutare altri utenti per acquisti futuri.

Infatti ogni tipo di utente potrà decidere di lasciarne un consenso positivo ad un vino, questa caratteristica oltre ad aiutare le aziende, dà a questo e-commerce un’aspetto social.

Altro aspetto importante sarà la possibilità di contattare, tramite un’apposita sezione del sito, l’admin che prontamente potrà rispondere tramite e-mail.

1.2.Obiettivi di design

Il sistema dovrà poter essere il più efficiente ed intuitivo possibile. Tale efficienza sarà data da brevi tempi di risposta ad ogni genere di input. Si punterà ad aver una buona manutenibilità attraverso il facile inserimento di nuove funzionalità dovuto ad una strutturazione del software semplice da manipolare. Altro aspetto da considerare sono le rigide politiche che eviteranno che l’utente sbagli ad inserire alcuni input.

Parte degli aspetti che il sistema avrà sono :

1.2.1Criteri di Performance

Tempo di risposta	La Cantinella del Bonsignore assicura una risposta rapida alle richieste dell’utente, la quale verrà gestita ed elaborata
-------------------	---

	entro 10 secondi. Nel caso di una connessione molto lenta tale aspetto potrebbe essere non garantito, ovviamente i tempi di risposta saranno più rapidi quanto più è veloce la tipologia di connessione utilizzata.
Memoria	La quantità di memoria che verrà utilizzata dal sito web non può essere stimata precisamente. Il sistema potrebbe essere sottoposto alla memorizzazione di centinaia di vini e migliaia di utenti, quindi una buona scalabilità in tal senso dovrà essere garantita.

1.2.2 Criteri di Affidabilità

Robustezza	La Cantinella gestirà eventuali input errati senza interrompere il funzionamento dell'intero sistema, l'utente verrà avvertito con eventuali messaggi di errore.
Sicurezza	L'accesso al sistema è controllato da un sistema di autenticazione che categorizza gli utenti non permettendo l'accesso a parti del sistema non autorizzate, oltre alla presenza di filtri che impediranno accessi non previsti.

1.2.3 Criteri di Manutenzione

Estendibilità	Il sistema dovrà essere propenso all'inserimento di nuove funzionalità. Quindi, si adotterà una metodologia di programmazione ben strutturata, dunque, di facile comprensione.
Modificabilità	Dovrà essere possibile effettuare modifiche al sistema in caso di bug. In tal senso, come nel caso dell'estendibilità, la metodologia di programmazione aiuterà a risolvere vari problemi facilmente.

1.2.4 Criteri per l'Utente Finale

Usabilità	La Cantinella del Bonsignore sarà di semplice utilizzo, grazie ad una grafica semplice ed efficace che guiderà il cliente nell'acquisto di vini.
-----------	--

1.3 Definizioni, Acronimi e Abbreviazioni

La Cantinella del Bonsignore: Nome del sistema che verrà sviluppato.

JDBC: Java DataBase Connection.

DBMS: Database Management System, Sistema di gestione del database

Mysql: è il più diffuso database Open Source basato sul linguaggio SQL.

RAD: Requirements Analysis Document.

SDD: System Design Documents

User-friendly: aggettivo utilizzato per definire un software di facile utilizzo anche per persone non esperte nell'utilizzo del computer

Admin: amministratore del sistema

1.4 Riferimenti

Documento RAD del progetto **La Cantinella del Bonsignore**.

1.5 Panoramica

E' importante fare un accenno alle attività di system design che costituiscono le fondamenta per l'architettura software del sistema.

1. *Decomposizione del sistema:* Il sistema viene suddiviso in diversi sottosistemi.

Ogni sottosistema è caratterizzato da servizi che offre ad altri sottosistemi.

L'insieme dei servizi sarà denominato Interfaccia.

2. *Mapping Hardware/Software:* in questa sezione vengono prese decisioni riguardo le piattaforme hardware scelte per il sistema, la comunicazione tra nodi, e come vengano incapsulati i servizi di un sottosistema.

3. *Gestione dei dati persistenti:* In tale sezione vengono individuati gli oggetti che devono essere resi persistenti e quale software verrà utilizzato per tale scopo.

4. *Politiche di accesso e Sicurezza:* In tale sezione si individueranno grazie ad una tabella le operazioni che ogni attore può effettuare.

5. *Controllo del software globale:* che descrive il modo in cui è implementato il controllo globale del software e come si sincronizzano i sottosistemi.

6. *Condizioni Boundary:* che descriverà oltre l'avvio anche la gestione dei fallimenti, interruzione di corrente o anche a errori di progettazione.

2 SISTEMA SOFTWARE CORRENTE

Attualmente non esiste un sistema software che ha l'obiettivo di aiutare le piccole aziende vinicole a poter espandere il loro business.

Esistono vari sistemi e-commerce che danno la possibilità di poter inserire i propri articoli al fine di venderli ma nessuno di questi è specializzato nella vendita di vini.

Inoltre dei vari e-commerce presenti su internet, non tutti danno la possibilità di vendere alcolici. Pertanto il sistema che noi stiamo sviluppando è un sistema che rientra nel campo della Greenfield Engineering.

Quindi il progetto software verrà sviluppato da zero e per tale ragione necessiterà di uno studio dei requisiti che verranno ricavati dai bisogni degli utenti finali

3. Architettura del Sistema proposto

3.1 Panoramica

Il sistema che intendiamo sviluppare È un software web-Based e non sarà altro che un e-commerce incentrato nella vendita di vini.

Le tipologie di utenti si dividono in: User, cliente, cliente venditore ed infine admin.

L'utente user potrà accedere a tutti i servizi che non necessitano di Login, perciò la visualizzazione del catalogo, la sezione contact us, l'home page e ovviamente tutti i servizi che permettono la creazione dell'account.

L'utente cliente, invece, potrà accedere alle funzionalità che permettono l'acquisto dei vini e la visualizzazione delle proprie informazioni personali.

Il cliente venditore, invece, oltre ad essere un cliente potrà inserire all'interno dell'e-commerce gli articoli dell'azienda di cui è rappresentante.

Infine ultimo utente del nostro sistema sarà l'admin che gestirà l'e-commerce ed eventuali problemi che vi si possono presentare. Infatti avrà la possibilità di cancellare utenti dalla piattaforma e di visualizzare gli ordini effettuati sul sito, in modo tale da poter monitorare i propri guadagni.

L'architettura del sistema sarà Client/server.

Il server riceverà le richieste da parte del Client e cercherà di rispondere in un tempo ristretto.

I motivi di questa scelta sono:

1. Portabilità: Essendo un'applicazione web, potrà essere usata su una varietà di macchine e

- sistemi operativi e quindi sia da computer fissi che dispositivi mobili.
2. Scalabilità: il server sarà in grado di gestire un grosso numero di richieste da parte dei client.
 3. Affidabilità: entrambe le componenti client-server garantiranno affidabilità sui dati e quindi anche in seguito a guasti sarà possibile recuperare le informazioni .Per tale motivo verranno effettuati backup periodici al database con cadenza mensile.
 4. Performance: Il sistema garantirà reattività nel completamento dei task, anche se tale aspetto è molto influenzato dalla qualità della connessione ad internet che l'utente finale possiede.

Nello sviluppo del software utilizzeremo un'architettura di tipo three-tier, tale Pattern prevede che il software venga diviso in tre parti ognuna delle quali ha un compito diverso.

La view si occuperà di curare l'interazione con l'utente, Il controller riceverà informazioni della View per mandarli al model o viceversa, mentre il model si occuperà della gestione dei dati e quindi dell'iterazione con il database sottostante.

L'utilizzo di questo modello comporta numerosi vantaggi, primi tra tutti sono la facilità di manutenzione ed estensione. Infatti essendo uno standard di programmazione sarà di facile comprensione a programmatori futuri.

I model verranno realizzati utilizzando classi Java appropriate, la parte di view verrà implementata utilizzando html in pagine jsp e i control saranno realizzati tramite servlet.

3.2 Decomposizione in sottosistemi

Per la realizzazione del sistema sarà utilizzata una architettura three-tier.

Utilizzeremo una particolare architettura multi-tier in cui la logica dell'applicazione verrà suddivisa in tre parti detti layer:

1. Presentation layer: composto da tutte le interfacce grafiche in particolare dai boundary object come le form che verranno compilate dagli utenti.
2. Application layer: composta dagli oggetti che si occuperanno dell'elaborazione dati e di notificare cambiamenti al presentation layer. Inoltre questo strato interagirà con il database sottostante tramite lo storage layer.
3. Storage layer: Si occupa della memorizzazione di dati persistenti e del loro recupero dal database attraverso query

Il sottosistema presentation layer è stato suddiviso in quattro sottosistemi

1. Sottosistema user: include tutte le interfacce grafiche a cui l'user può accedere ovvero: visualizzare il catalogo, la form contact us e la homepage.
2. Sottosistema cliente: oltre a tutte le interfacce grafiche che può visualizzare l'user, potrà

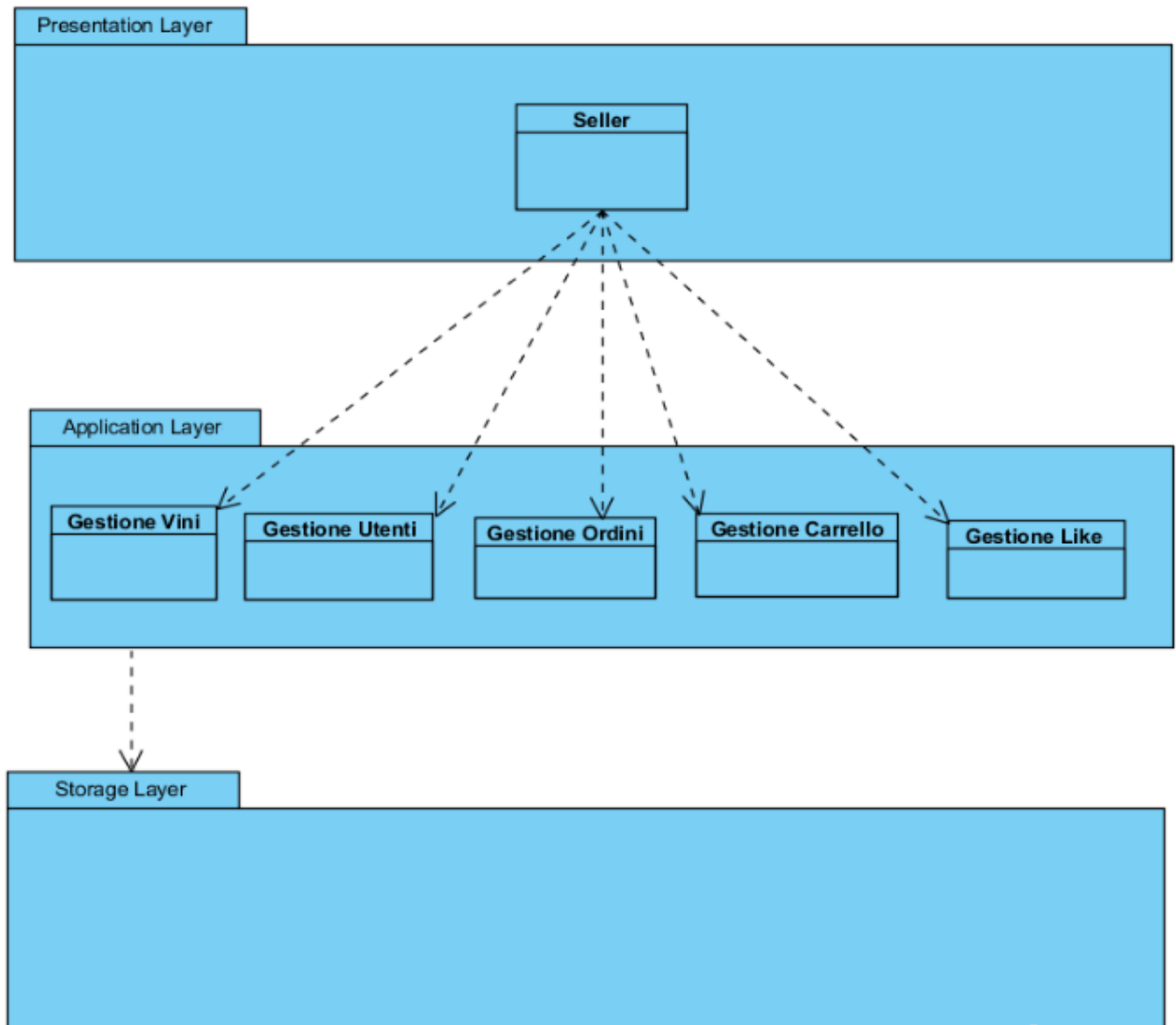
- accedere alla funzionalità di acquistare vini. Inoltre potrà accedere ad interfacce grafiche aggiuntive come quella per la visualizzazione delle informazioni personali ed il carrello.
3. Sottosistema cliente azienda: tale utente oltre ad essere cliente, e quindi poter accedere a tutte le sue funzionalità, potrà inserire all'interno dell'e-commerce gli articoli dell'azienda che rappresenta.
 4. Sottosistema admin: include le interfacce accessibili all'admin come quella che permetterà la cancellazione degli utenti dal sistema oppure quella che permetterà la visualizzazione degli ordini effettuati sulla piattaforma.

Il sottosistema application layer è stato suddiviso in vari sottosistemi

1. Sottosistema Gestione vini che include tutte le operazioni che permettono la visualizzazione del catalogo, l'aggiunta di un vino e la sua modifica, l'eliminazione di un vino, la visualizzazione degli articoli tramite un filtro e la visualizzazione delle informazioni relative alla descrizione di un vino
2. Sottosistema Gestione ordini che include tutte le operazioni di creazione di un ordine e la visualizzazione
3. Sottosistema Gestione utenti che include tutte le operazioni di creazione di un utente, visualizzazione delle informazioni personali, eliminazione di un utente e la verifica dei dati in fase di Login.
4. Sottosistema Gestione like che include tutte le operazioni di aggiunta di un like e la sua rimozione
5. Sottosistema Gestione Carrello che include tutte le operazioni di gestione del carrello quali l'aggiunta di un articolo in esso, la rimozione dei vini dal carrello, e la sua visualizzazione

Ora mostriamo nello specifico ogni layer:

Cliente Azienda



Gestione vini: Il cliente Azienda potrà visualizzare tutti i vini presenti nel database e la loro descrizione.

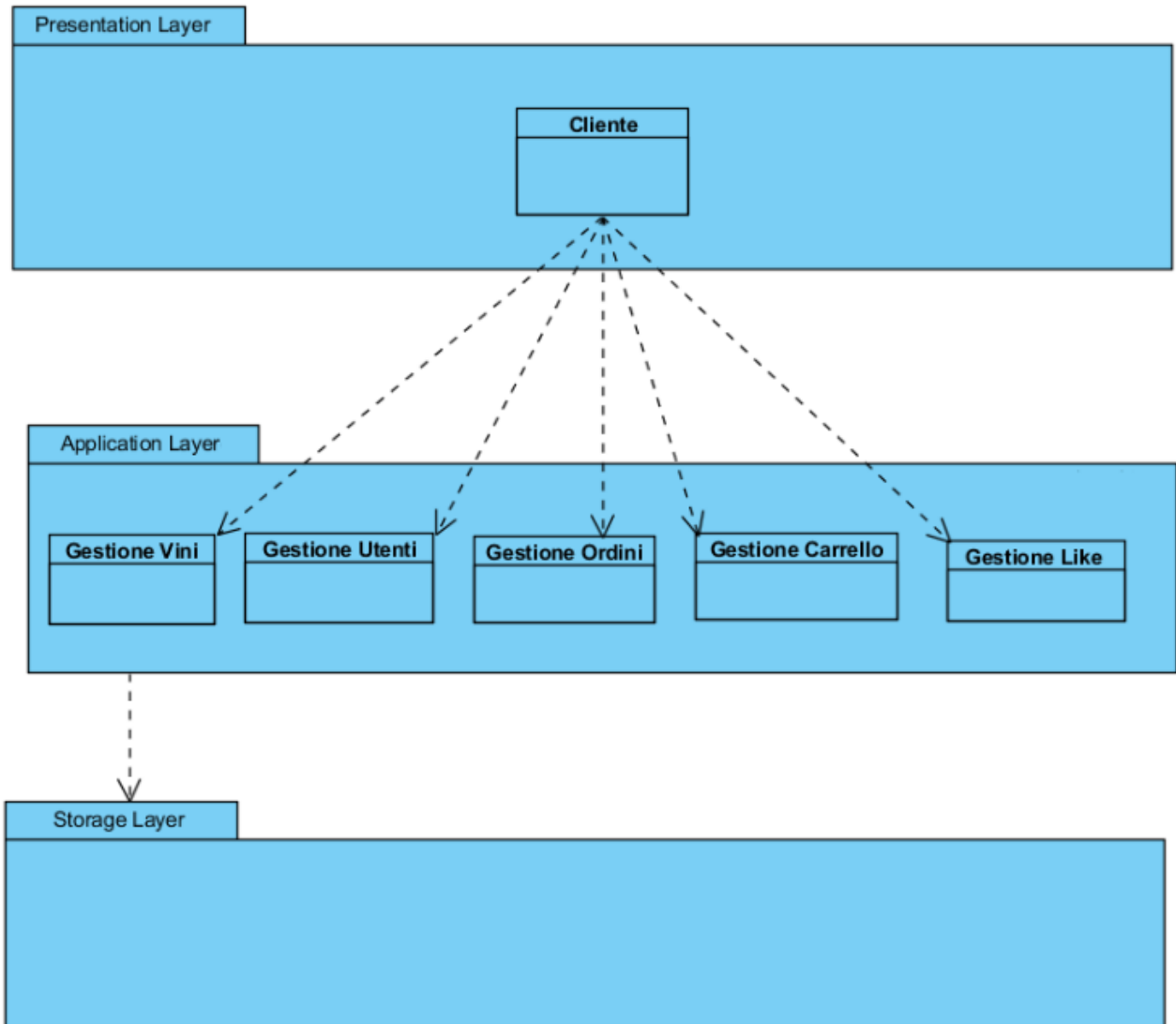
Avrà la possibilità di aggiungere vini al catalogo ed eventualmente modificare prezzo e quantità. Infine avrà la possibilità di cancellare i vini da lui inseriti.

Gestione utenti: Il cliente Azienda avrà la possibilità di loggarsi al sistema e quindi verificare i propri dati. Inoltre potrà accedere alla propria sezione personale dove visualizzerà le informazioni inserite durante la registrazione.

Gestione Ordini: in seguito alla scelta dei vini da acquistare, il cliente Azienda potrà effettuare ordini.

Gestione Carrello: Il cliente Azienda potrà aggiungere articoli al carrello, ed eliminare gli stessi da esso. Ovviamente potrà visualizzare il carrello (vuoto o pieno che sia)

Gestione Like: Il cliente Azienda potrà inserire un consenso e successivamente eliminarlo.
Cliente



Gestione vini: Il cliente potrà visualizzare tutti i vini presenti nel database e la loro descrizione.

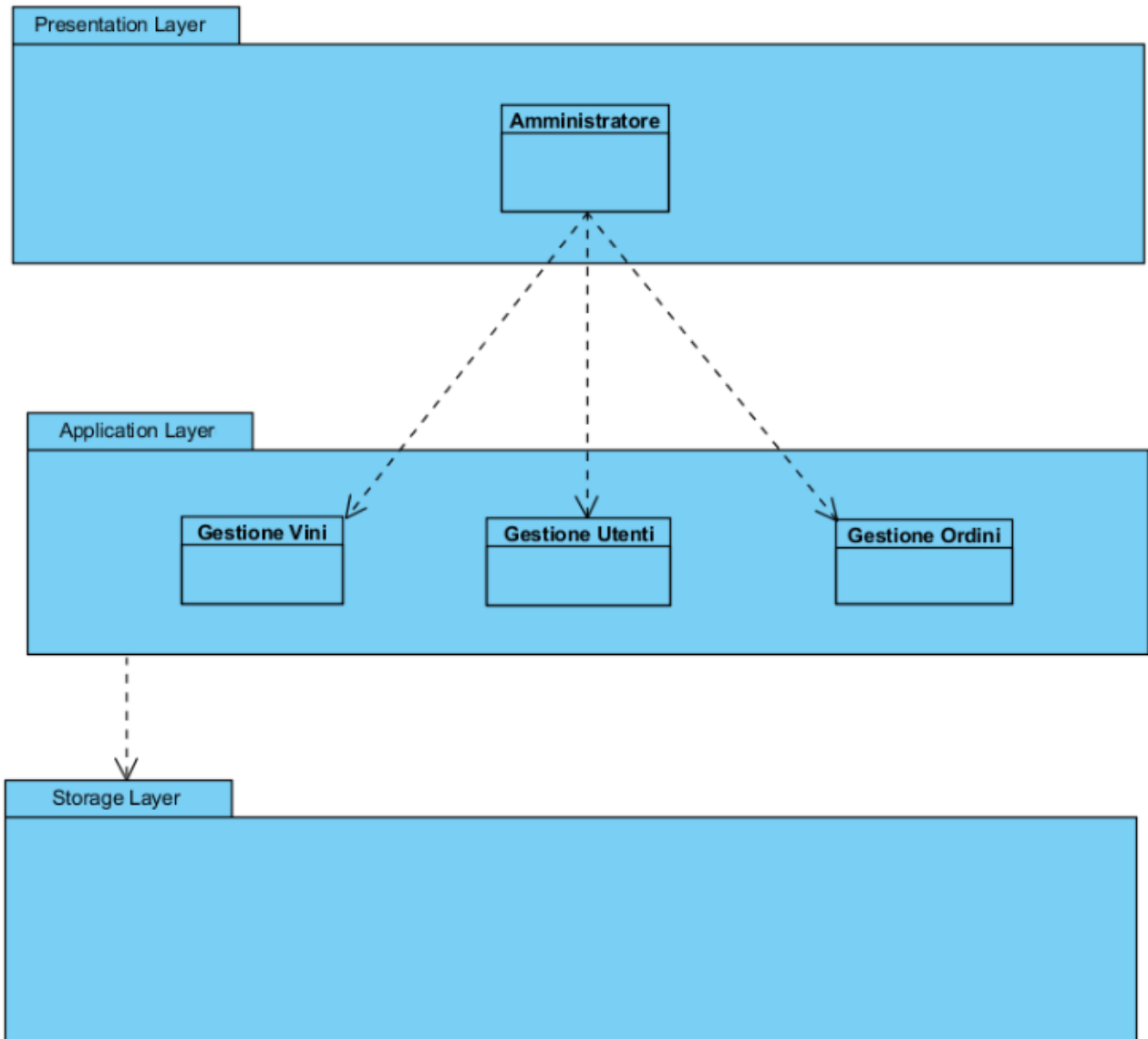
Gestione utenti: Il cliente avrà la possibilità di loggarsi al sistema e quindi verificare i propri dati. Inoltre potrà accedere alla propria sezione personale dove visualizzerà le informazioni inserite durante la registrazione.

Gestione Ordini: in seguito alla scelta dei vini da acquistare, il cliente potrà effettuare ordini.

Gestione Carrello: Il cliente potrà aggiungere articoli al carrello, ed eliminare gli stessi da esso. Ovviamente potrà visualizzare il carrello (vuoto o pieno che sia)

Gestione Like: Il cliente potrà inserire un consenso e successivamente eliminarlo.

Amministratore (Admin)

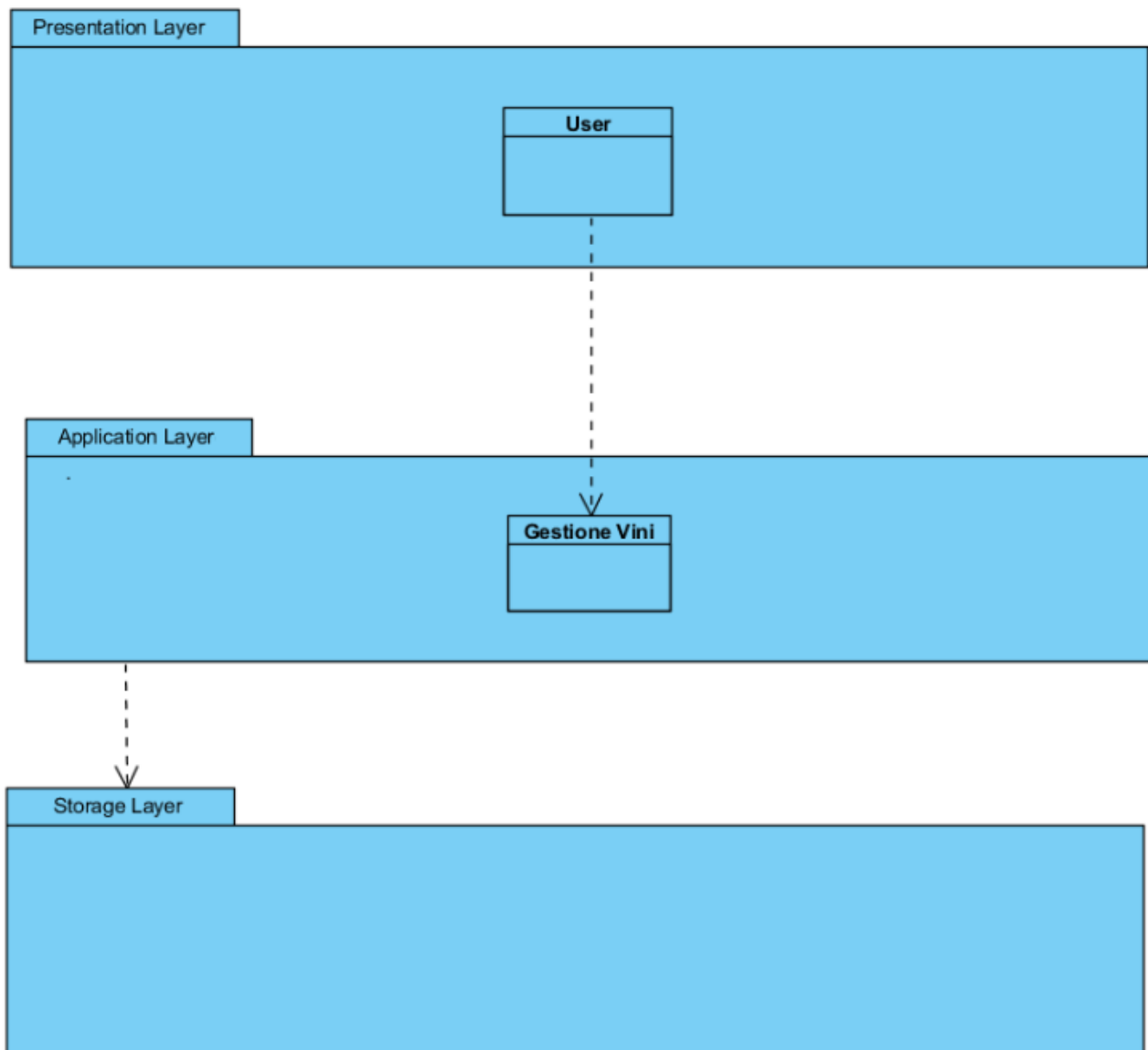


Gestione vini: L'amministratore modera implicitamente i vini, valendosi sull'utente registrato.

Gestione utenti: L'amministratore avrà la possibilità di loggarsi al sistema e quindi verificare i propri dati. Inoltre potrà accedere alla sezione dedicata alla gestione degli utenti, ed eventualmente eliminarne qualcuno.

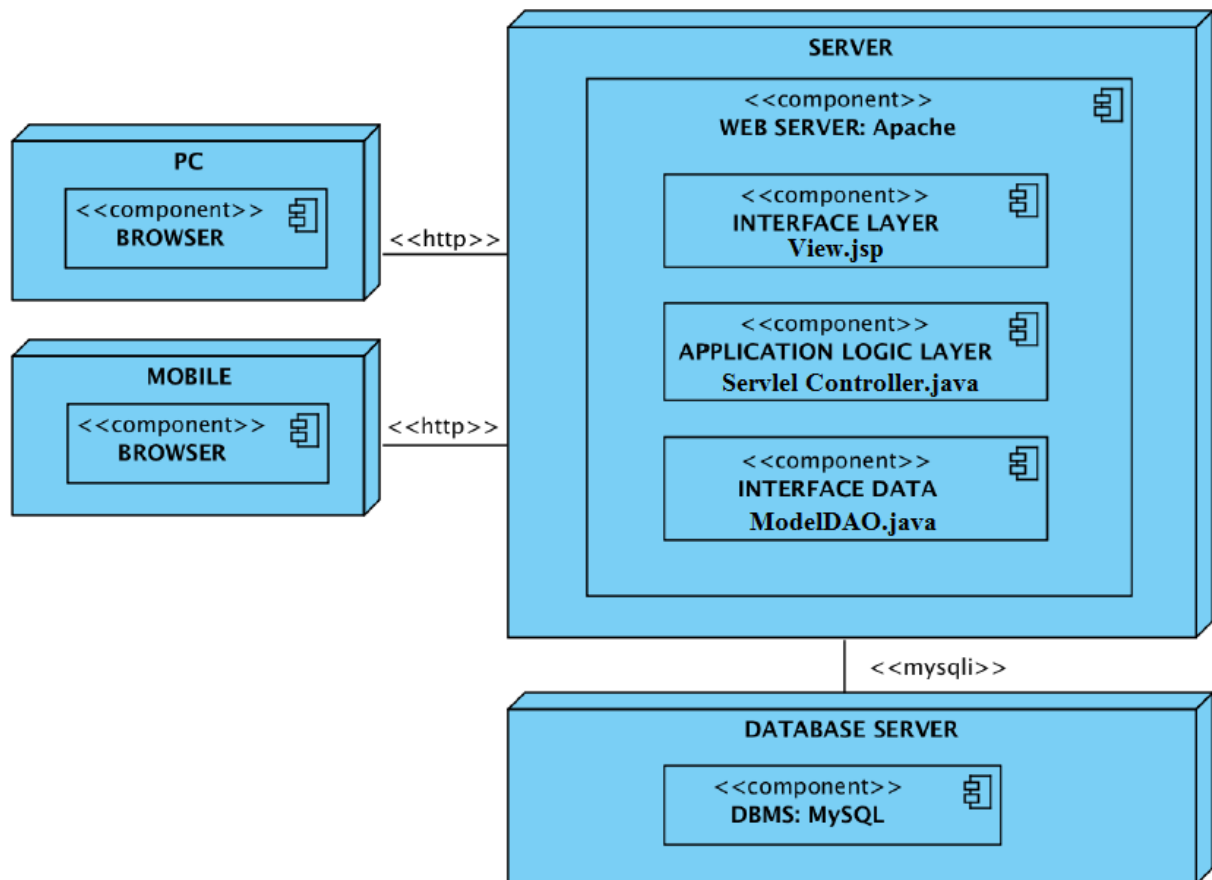
Gestione Ordini: L'amministratore potrà visualizzare gli ordini effettuati sul sito.

User



Gestione vini: L'user potrà visualizzare tutti i vini presenti nel database e la loro descrizione.

3.3 Mapping hardware software



Web Server

Il server utilizzato è Apache versione 9.

Interface layer

L'utente utilizza il sistema mediante un Browser installato all'interno del suo calcolatore (ad es. Opera, Firefox, Chrome).

Application Logic layer

Il sistema, e quindi le funzionalità, sono implementate in linguaggio HTML5 + java.

Storage layer

Rappresenta il collegamento con il server da parte del sistema e si occupa di tutte le richieste di accesso e modifiche sui dati permanenti presenti nel database.

Database Server

Il DBMS usato è MySQL.

3.4 Gestione dati persistenti

Cliente

NOME	TIPO	VINCOLI	KEY
CF	varchar(45)	NOT NULL	PRIMARY KEY
Nome	varchar(25)	NOT NULL	
Cognome	varchar(25)	NOT NULL	
User	varchar(45)	NOT NULL	
Password	varchar(45)	NOT NULL	
email	varchar(50)	NOT NULL	
isSeller	boolean		
Nome Azienda	varchar(45)	DEFAULT NULL	
Partita iva	varchar(50)	DEFAULT NULL	

Vino

NOME	TIPO	VINCOLI	KEY
idVino	int(11)	UNSIGNED, NOT NULL	PRIMARY KEY
CFUser	varchar(45)	NOT NULL	FOREIGN KEY
Nome	varchar(25)	NOT NULL	
Descrizione	varchar(1000)	NOT NULL	
Anno	varchar(45)	NOT NULL	
Prezzo	varchar(45)	NOT NULL	
Immagine	varchar(500)		
Tipo	varchar(15)	NOT NULL	
Quantita	int(225)	NOT NULL	
Feedback	int(225)	DEFAULT 0	

Feedback

NOME	TIPO	VINCOLI	KEY
idVino	int(11)	UNSIGNED, NOT NULL	FOREIGN KEY
CF	varchar(45)	NOT NULL	FOREIGN KEY

Ordine

NOME	TIPO	VINCOLI	KEY
idOrdine	int(225)	UNSIGNED, NOT NULL	PRIMARY KEY
user	varchar(45)	NOT NULL	FOREIGN KEY
totale	varchar(225)	NOT NULL	
Descrizione	varchar(500)	NOT NULL	
email	varchar(50)	NOT NULL	
indirizzo	varchar(50)	NOT NULL	
zipcode	int(10)	NOT NULL	
percentuale	varchar(255)	NOT NULL	

3.5 Controllo degli accessi e sicurezza

All' interno del sistema i vari attori hanno il permesso di eseguire operazioni diverse sui vari sottosistemi. Per schematizzare meglio il controllo si è deciso di usare la matrice degli accessi riportata di seguito:

Attore Sottosistema	Cliente	Cliente Azienda	Admin	User
Utenti	×Login ×Logout ×Visualizzazione info personali	×Login ×Logout ×Visualizzazione info personali	×Login ×Logout ×eliminare utenti	
Vini	×Visualizza catalogo ×Visualizza informazioni vino	×Visualizza catalogo ×Visualizza informazioni vino ×Aggiungere vino ×Eliminare vino inserito ×Modificare vino		×Visualizza catalogo ×Visualizza informazioni vino
Carrello	×Aggiungere articolo al carrello ×Eliminare articolo dal carrello ×Visualizzare Carrello	×Aggiungere articolo al carrello ×Eliminare articolo dal carrello ×Visualizzare Carrello		
Ordini	×Effettuare ordine	×Effettuare ordine	×Visualizzare ordini effettuati sul sistema	
Like	×Inserire like ×eliminare like inserito	×Inserire like ×eliminare like inserito		

3.6 Controllo Globale del software

Il flusso di controllo globale è la sequenza di azioni nel sistema. Il sistema ha un flusso guidato di eventi; le funzionalità richiedono un'interazione continua da parte dell'utente; per questo motivo, il controllo del flusso globale che utilizziamo è di tipo event-driven.

Quindi non abbiamo una sequenza di operazioni prestabilite ma, è l'utente che sceglie l'operazione da eseguire.

3.7 Condizione di boundary

3.7.1 Start-up

Per lo start-up iniziale del sistema è necessario l'avvio di un web server che fornisca il servizio di un Database MySql per gestire i dati persistenti e l'interpretazione e l'esecuzione del codice dal lato server. Tramite l'interfaccia di Login, sarà possibile autenticarsi tramite credenziali(username e password) come utente, e sarà permesso l'accesso a tutte le funzionalità del sistema.

3.7.2 Terminazione

Alla corretta chiusura del servizio, il sistema termina con un Logout regolare, e vengono inviati gli aggiornamenti ai sottosistemi, al fine di gestire correttamente i dati in fase di chiusura. La consistenza dei dati, viene assicurata annullando eventuali operazioni in esecuzione.

3.7.3 Fallimento

I casi di fallimento del sistema, possono essere diversi:

1. Guasti dovuti al sovraccarico e successivo fallimento del database. In questo caso, come provvedimento, si ha quello del salvataggio periodico dei dati sotto forma di codice SQL per la successiva rigenerazione del DB.
2. Interruzione inaspettata dell'alimentazione, non sono previsti metodi di ripristino dello stato del sistema prima dello spegnimento.
3. Software che causa la chiusura inaspettata dovuta ad errori commessi durante la fase di implementazione. In questo caso non sono previste politiche di correzione; l'unico processo eseguibile, sarà la chiusura del sistema e il suo riavvio.
4. **Errore critico nell'hardware. In questo caso non sono previste risoluzioni.**

4 SERVIZI DEI SOTTOSISTEMI

4.1 Gestione Vini

Sottosistema	Autenticazione
Descrizione	Sottosistema che gestisce tutte le operazioni riguardante i vini.
Servizi Offerti	
Servizio	Descrizione
Visualizzazione catalogo	Questa funzionalità permette di visualizzare il catalogo dei vini disponibili sul sistema.
Visualizzazione catalogo	Questa funzionalità permette di visualizzare il catalogo dei vini disponibili sul sistema tramite filtro.
Visualizzazione informazioni vino	Questa funzionalità permette di visualizzare le informazioni di un vino.
Aggiunta di un vino	Questa funzionalità permette di aggiungere un nuovo articolo nel database.
Modifica di un vino	Questa funzionalità permette di modificare le caratteristiche di un vino, quali la quantità nel database ed il suo prezzo.
Eliminazione di un vino	Questa funzionalità permette di eliminare un vino dal database.

4.2 Gestione utenti

Sottosistema	Autenticazione
Descrizione	Sottosistema che gestisce tutte le operazioni riguardante gli utenti del sito.
Servizi Offerti	
Servizio	Descrizione
Login	Questa funzionalità permette di effettuare l'accesso al sistema inserendo le proprie credenziali.
Logout	Questa funzionalità permette di uscire dal sistema.
InfoCliente	Questa funzionalità permette di visualizzare le informazioni inserite durante la registrazione.
Aggiunta Cliente	Questa funzionalità permette di registrarsi al sistema e quindi inserire un nuovo cliente
Elimina Cliente	Questa funzionalità permette di eliminare un cliente dal sistema.

4.3 Gestione Like

Sottosistema	Autenticazione
Descrizione	Sottosistema che gestisce tutte le operazioni riguardante i consensi agli vini.
Servizi Offerti	
Servizio	Descrizione
Aggiungi like	Questa funzionalità permette di aggiungere un like ad un vino.
Elimina like	Questa funzionalità permette eliminare un like inserito.

4.4 Gestione ordini

Sottosistema	Autenticazione
Descrizione	Sottosistema che gestisce tutte le operazioni riguardante gli ordini effettuati.
Servizi Offerti	
Servizio	Descrizione
Visualizza ordini	Questa funzionalità permette di visualizzare gli ordini effettuati sul sito web.
Creazione ordine	Questa funzionalità permette di aggiungere un nuovo ordine nel sistema.

4.5 Gestione Carrello

Sottosistema	Autenticazione
Descrizione	Sottosistema che gestisce tutte le operazioni riguardante il carrello.
Servizi Offerti	
Servizio	Descrizione
Aggiungere articoli al carrello	Questa funzionalità permette di aggiungere un articolo al carrello
Eliminare articolo dal carrello	Questa funzionalità permette di eliminare un articolo presente nel carrello
Visualizzazione del carrello	Questa funzionalità permette di visualizzare il carrello ed eventuali articoli contenuti in esso

5 GLOSSARIO

Client: componente che accede ai servizi o alle risorse di un'altra componente, detta server.

Deployment Diagram: Schema che descrive la struttura dinamica del sistema

DBMS: programma informatico (o, più frequentemente, un insieme di programmi) progettato per gestire un database, ovvero un insieme di numerosi dati strutturati. Le operazioni, normalmente, sono richieste da un gran numero di utenti.

Login: Procedura attraverso la quale ci si collega con un qualsiasi servizio in linea. All'utente viene assegnato un nome di login ed una password che vengono richiesti dal sistema ogni volta che ci si collega.

Layer: E' un insieme di classi con funzionalità simile (tipicamente raggruppati in un unico package).

Logout: Operazione attraverso la quale si termina un collegamento con un sistema al quale si ha accesso attraverso un nome utente e una password (vedi login).

MySQL: Database management system relazionale, composto da un client con interfaccia a caratteri e un server, disponibile su molte piattaforme.

Server: Programma di gestione di un servizio che invia informazioni in un particolare formato ricevuto e interpretato da un programma Client dal lato ricevente.

Shutdown: fase di terminazione del sistema.

Start-up: processo di accensione e di avvio di un computer, di un dispositivo o di un sistema.