A dark blue vertical bar runs down the left side of the page. A blue arrow points to the right from this bar, containing the date.

10/11/2018

S.D.D- System Design Document

Team TopHw

Several thin, curved lines in shades of blue and grey sweep upwards from the bottom left corner of the page.

Partecipanti

Nome	Cognome	Matricola
Alfonso	Rianna	0512104922
Davide	Zambelli	0512104688
Raffaele	Dragone	0512105016

Scritto da

Nome	Cognome	Matricola
Alfonso	Rianna	0512104922
Davide	Zambelli	0512104688
Raffaele	Dragone	0512105016

Sommario

1. Introduzione3

1.1 Scopi del sistema	3
1.2 Design goals	3
Criteri di Performance	3
Criteri di affidabilità	3
Criteri di mantenimento	4
1.3 Definizioni, acronimi e abbreviazioni	4
Acronimi	4
1.4 Riferimento	4
2. Architettura corrente del software	5
3. Architettura del sistema proposto	5
3.1 Overview	5
3.2 Decomposizione in sottosistemi	6
3.3 Mapping Hardware/software	9
3.4 Gestione dei dati persistenti	10
3.5 Controllo degli accessi e sicurezza	11
3.5.1 Controllo degli accessi	11
3.5.2 Sicurezza	12
3.6 Gestione del controllo globale	12
3.7 Condizione di limite	13

1. Introduzione

1.1 Scopi del sistema

Il progetto prevede la realizzazione di un sito web di e-commerce, per la vendita di prodotti informatici, la gestione dell'intero sistema dipende dall'amministratore, incluso l'inserimento, la modifica e la cancellazione di prodotti.

I clienti potranno acquistare prodotti solo se sono registrati al sito. Gli utenti che non sono registrati non possono acquistare prodotti finché non effettuano la registrazione. Tutti gli utenti, registrati e non registrati, hanno la possibilità di chiedere informazioni sui prodotti ma anche dei chiarimenti su eventuali problematiche (metodi di pagamento, recesso, fatture,...).

Il sistema, inoltre, fornirà anche la possibilità di gestire il negozio in maniera automatica (entrate, uscite, visualizzazioni prodotti da parte di utenti registrati e non registrati, prenotazioni effettuate dagli utenti registrati, storico ordini, gestione e visualizzazione carrello da parte degli utenti registrati).

1.2 Design goals

Criteri di Performance

Throughput: Sarà possibile utilizzare il sistema da più macchine contemporaneamente. L'unica limitazione alle prestazioni del sistema sarà l'utilizzo di una database relazione, utilizzato come repository centrale.

Tempo di risposta: Mediamente una richiesta da parte del client è soddisfatta in un tempo inferiore ai 5 secondi. Queste prestazioni sono garantite mediante la scelta di un server capace di sopportare il carico applicativo massimo e la scelta di gestire i controlli sugli input sul client e l'utente avrà subito la segnalazione dell'eventuale non correttezza dell'input inserito.

Memoria: Prevedendo un immagazzinamento di dati quotidiano su database, riguardanti ad ordini o prodotti, si stima sia necessario almeno 1 TB di memoria di massa.

Usabilità: Il sistema sarà dotato di un'interfaccia grafica user-friendly, in modo tale che l'utente riesca ad interpretare il sistema facilmente. Inoltre dei messaggi di errore o successo aiuteranno l'utente a interpretare gli output .

Criteri di affidabilità

Affidabilità: Il sistema deve essere affidabile. Bisogna gestire eventuali crash del sistema al fine di non compromettere le sue funzionalità. Il è basato su web e su un server di database relazionale.

Efficienza: Il sistema deve svolgere i propri compiti in tempi accettabili utilizzando nel miglior modo le risorse disponibili.

Robustezza: Il software realizzato deve essere facilmente modificabile a fronte di ogni evenienza.

Funzionalità: Il sistema deve soddisfare le esigenze che vengono evidenziate nella fase di raccolta dei requisiti.

Manutenibilità: Il software deve essere facilmente modificabile in base alle esigenze dei clienti o per una futura revisione da parte dei progettisti.

Sicurezza: Il sistema deve essere sicuro. Deve essere protetto da accessi non autorizzati, ogni utente deve poter accedere alla propria area e le transizioni devono avvenire in modo sicuro.

Criteri di mantenimento

Estendibilità: E' consentito, in quanto è possibile aggiungere in futuro, nuove funzionalità al sistema (es: creare nuove classi, con l'estensione di quelle già esistenti).

Tracciabilità dei requisiti: La tracciabilità dei requisiti a partire dal codice è relativamente semplice per via della modularità delle componenti.

Leggibilità: Grazie alla progettazione modulare del software e all'aggiunta di linee di commento ad ausilio del codice risulterà di facile comprensione la lettura di sorgenti e/o la successiva modifica.

1.3 Definizioni, acronimi e abbreviazioni

Acronimi

S.D.D. : System Design Document.

DBMS : Database Management System.

1.4 Riferimento

- Rad "Sito web e-commerce:TopHardware" .
- Bernd Bruegge & Allen H. Dutoit, Object-Oriented Software Engineering: Using UML, Patterns and Java, (3rd edition), Prentice- Hall, 2003.
- Jim Arlow, Ila Neustadt, UML e Unified Process, McGraw-Hill Italia Ian Sommerville,I. Sommerville, Software Engineering (9th edition), Addison Wesley.
- Roger S. Pressman, Principi di Ingegneria del Software (terza edizione, 2000), Mc Graw.

2. Architettura corrente del software

Il Sistema TopHw nasce come sistema Greenfield Engineering cioè lo sviluppo parte da zero. La raccolta dei requisiti è stata fatta colloquiando con il cliente, cercando di estrarre i requisiti dalle sue richieste. Sul web è comunque possibile trovare numerosi altri siti di e-commerce che verranno esaminati per capirne limiti e punti di forza, queste informazioni saranno usate per migliorare il nostro servizio.

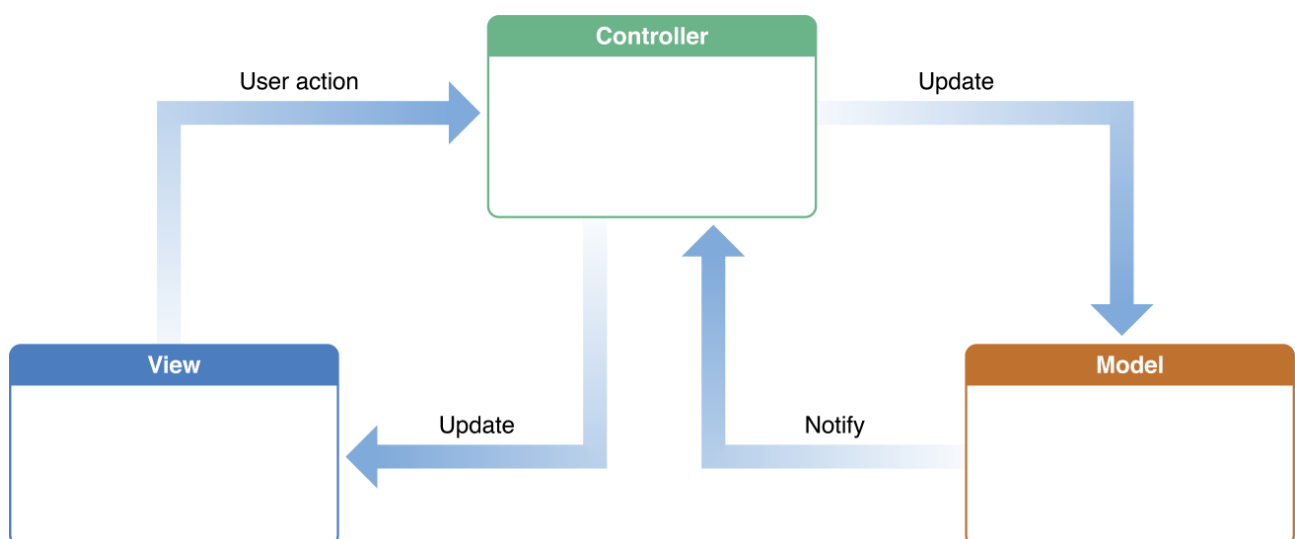
3. Architettura del sistema proposto

3.1 Overview

Il Sistema TopHw nasce come sistema Greenfield Engineering cioè lo sviluppo parte da zero.

L'architettura del sistema proposto è basata sul pattern architetturale strutturato su tre livelli logici MVC (Model - View - Controller).

Questo pattern permette di separare la logica di presentazione dei dati dalla logica applicativa e di semplificare le operazioni di mantenimento e aggiornamento, Pattern che è ampiamente utilizzato per applicazioni di tipo web.

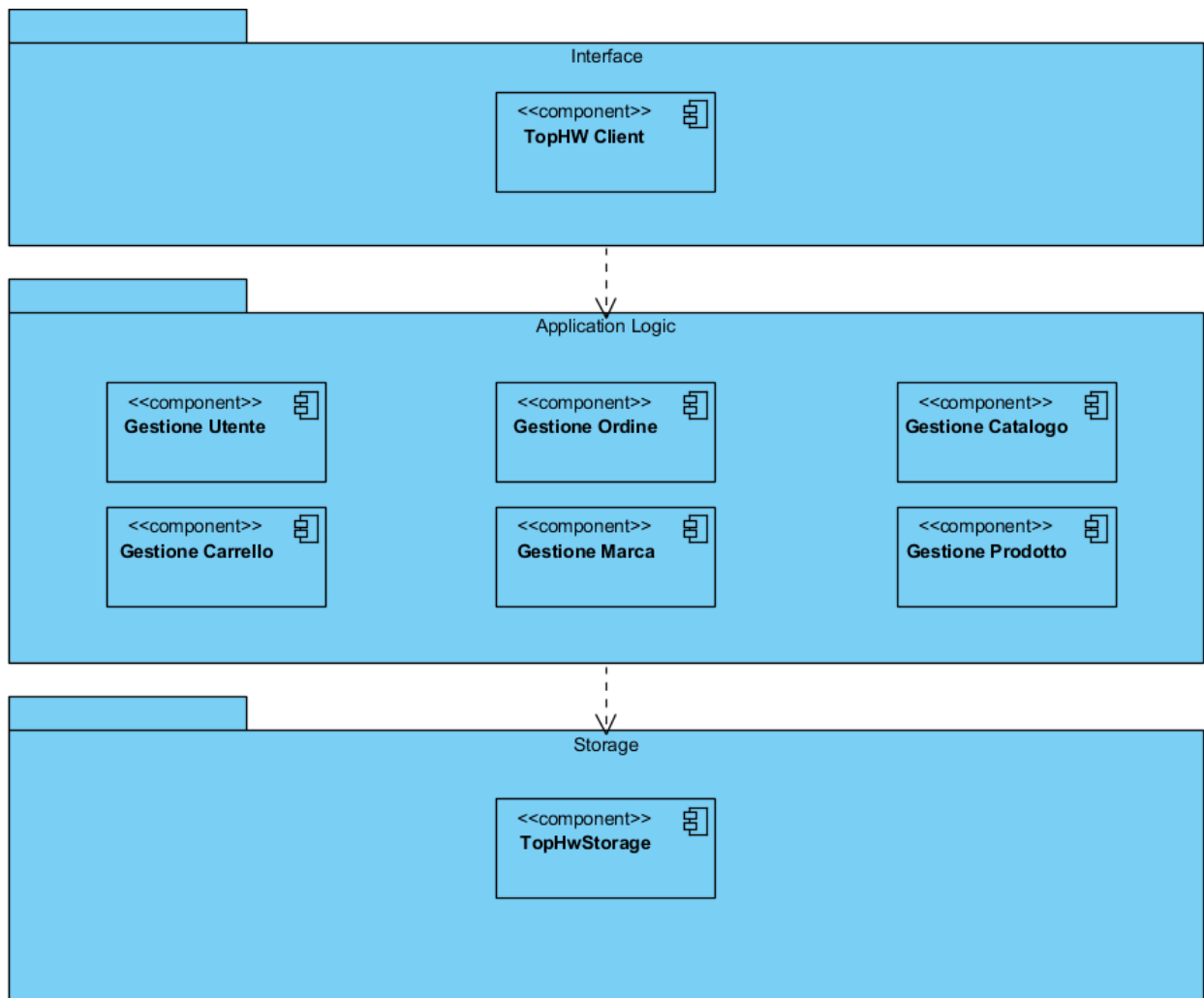


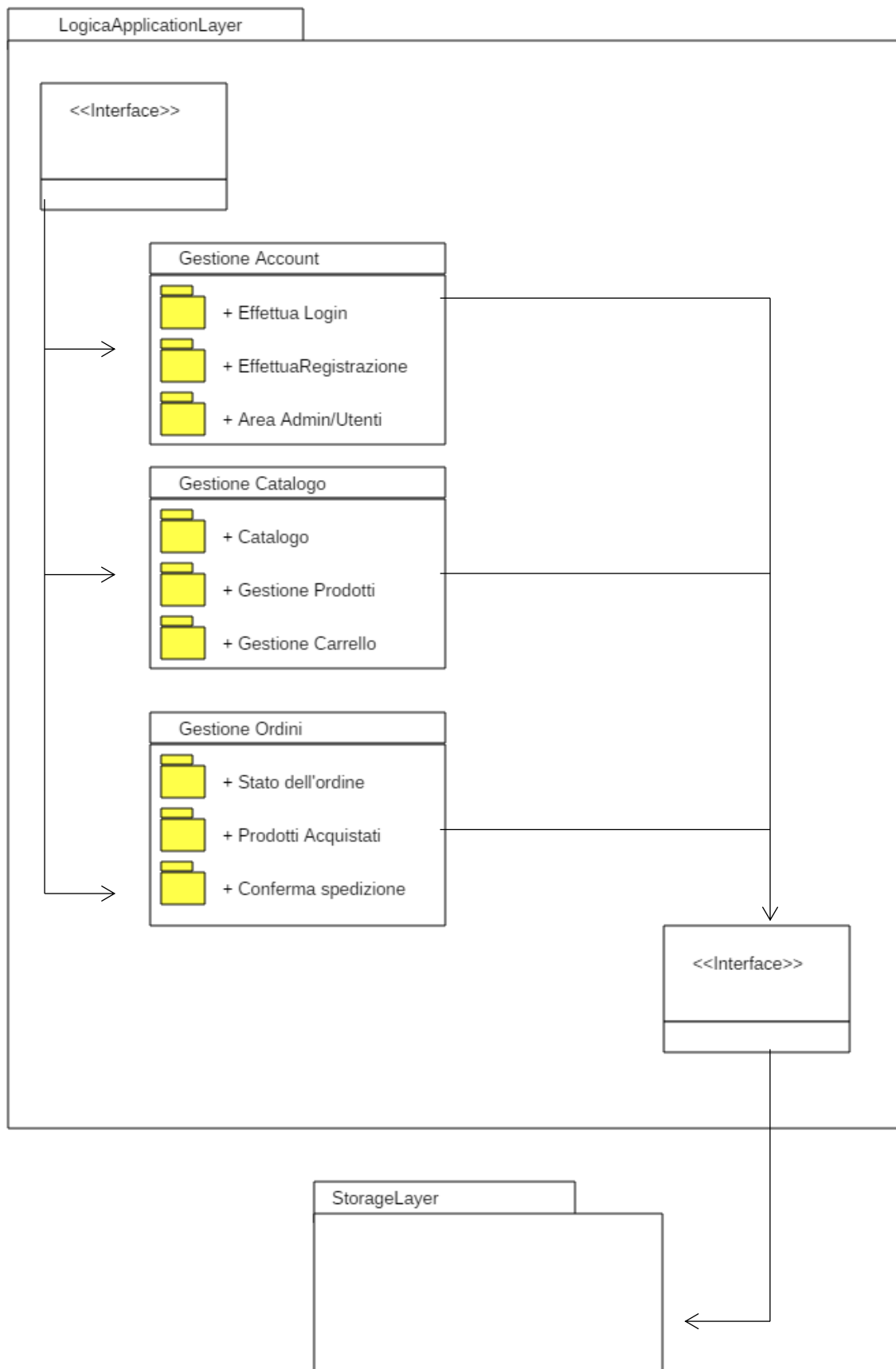
3.2 Decomposizione in sottosistemi

I tre livelli logici utilizzati nello specifico sono:

1. View Layer: che include tutti i boundary object, finestre, form ecc.
2. Control Layer: che include tutti i control object e gli entity object realizzando così tutti i processi logici e di controllo forniti dal sistema.
3. Model Layer: responsabile della gestione e della memorizzazione degli oggetti permanenti del sistema.

L'utente interagirà con le componenti View. Il Controller riceverà i comandi dell'utente, inviati tramite le View, e li metterà in atto richiamando le funzioni dei Model e le View. Il Model interagirà con il database e notificherà alla View i cambiamenti.





Ogni layer contiene al suo interno diversi moduli, di cui riportiamo una breve descrizione:

- ***Livello di interfaccia (Interface Layer):***

View	Insieme delle componenti che rappresentano l'interfaccia utente.
------	--

- ***Livello applicativo (Application Layer):***

Controllo	Classe che, in base al tipo di richiesta effettuata dall'utente, comunica con i model e/o mostra un'altra view.
Modello Prodotto	Classi che interagiscono con il database e notificano i cambiamenti nelle view.
Modello Utente	

- ***Livello Dati (Data Layer):***

Database	Sistema di archivio in cui vengono registrate le informazioni di tutti i dati contenuti nel sito.
----------	---

3.3 Mapping Hardware/software

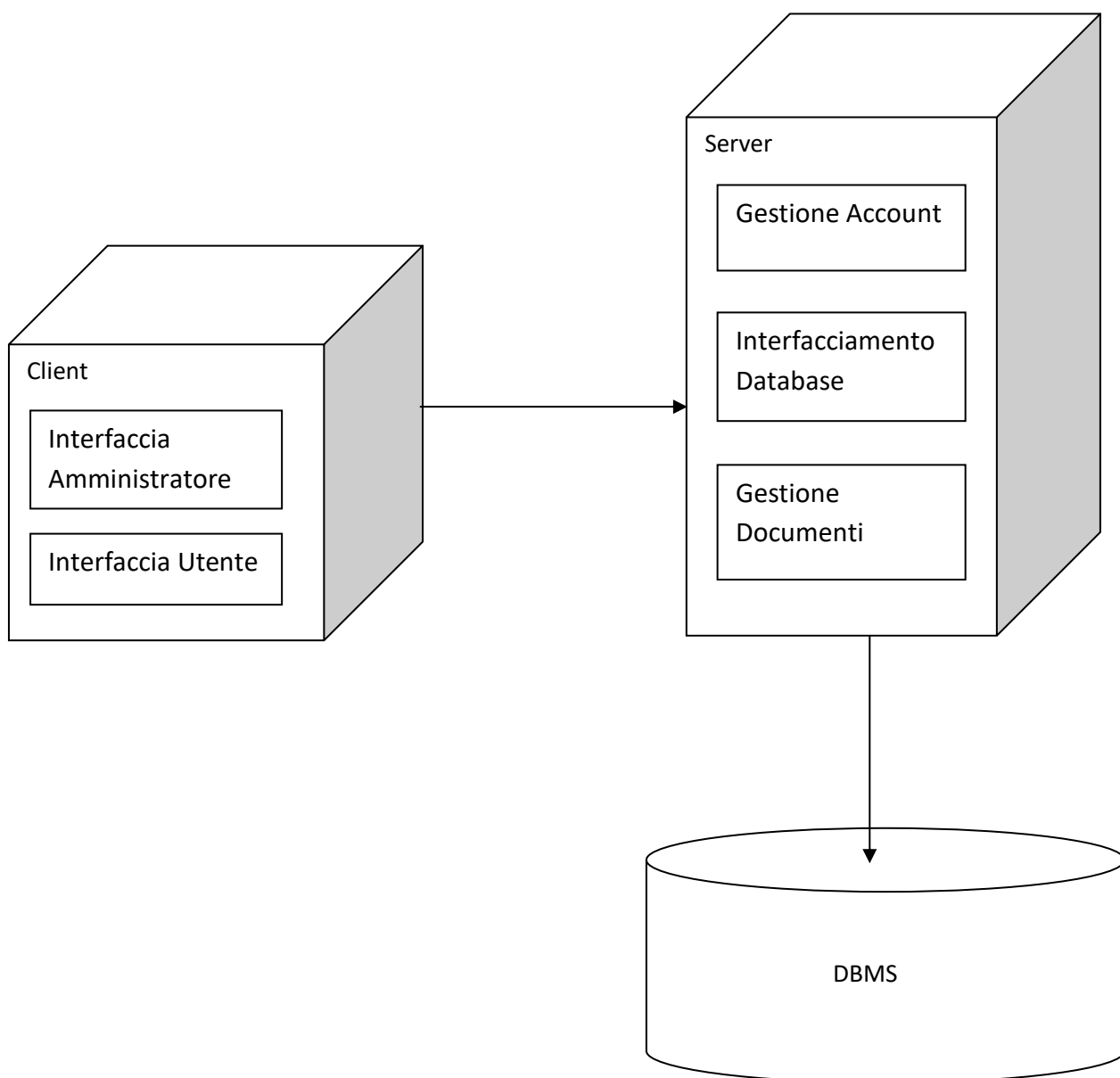
Il sistema che sarà realizzato si basa su un'architettura Web-based, composta da vari moduli client che richiedono servizi a un modulo server.

Protocollo Richiesto: HTTP

Memorizzazione Dati: DMBS MySQL

WebServer: GlassFish

Linguaggi di programmazione utilizzati: Html, CSS , Java, Javascript.



3.4 Gestione dei dati persistenti

La scelta dei dati persistenti riguarda gli oggetti che devono essere inseriti in opportuni Database e che devono perdurare quando il sistema viene chiuso, sia in maniera controllata che in seguito ad un crash.

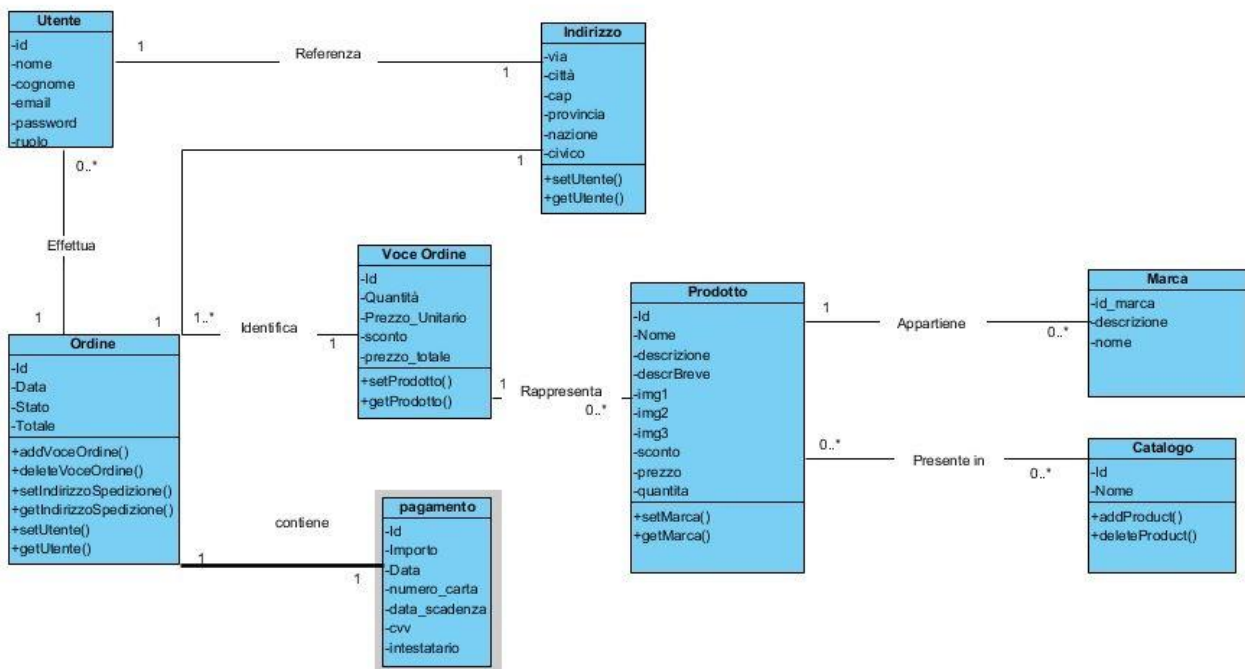
- Le credenziali per gli utenti saranno conservati in un database così da impedire l'accesso agli altri utenti non autorizzati.
- Scorte: in funzione di una buona gestione di un sito web di e-commerce è necessario tenere traccia dei consumi avvenuti durante gli acquisti effettuati dai clienti registrati.
- Ordini: è necessario avere un elenco degli ordini effettuati.

Gli oggetti individuati sono:

- Amministratore
- Utente registrato
- Utente non registrato

Per la memorizzazione dei dati utilizzeremo un database relazionale. Nel database i dati vengono memorizzati in tabelle che comprendono una lista di attributi.

Le tabelle saranno così strutturate:



3.5 Controllo degli accessi e sicurezza

3.5.1 Controllo degli accessi

Il sito web di e-commerce “Top Hardware” prevede tre tipi di utenti: Amministratore, utente registrato e utente non registrato. Ogni attore può accedere alle diverse funzionalità del sistema ma con diritti d’accesso organizzati in base alle differenti tipologie di utenza. Per documentare i diritti di accesso e per tenerne traccia all’interno del sistema usufruiamo di una tabella di controllo di accesso che descrive le varie operazioni permesse agli attori sui diversi oggetti. Le colonne della matrice rappresentano gli attori del sistema mentre le righe rappresentano gli oggetti su cui sono regolamentati gli accessi.

- L’amministratore accede al sito tramite la sua pagina di login. Egli si occupa della gestione dei prodotti quali inserimento, modifica e cancellazione. Inoltre si occupa anche di rispondere alle mail degli utenti qualora questi ultimi dovessero effettuare delle domande/richieste per ogni tipo di informazioni che riguarda i prodotti.
- Gli Utenti potranno accedere al sito, tramite la sezione dedicata al login, possono effettuare le prenotazioni sui prodotti che desiderano acquistare, controllare i prodotti inseriti nel carrello e controllare lo storico ordini. La modifica e/o la cancellazione del prodotto dal carrello potrà essere effettuata sempre dalla propria area utente, tutti gli utenti possono chiedere delle informazioni riguardanti i prodotti entrando nell’area riservata alla richiesta delle informazioni.
- Coloro che non sono registrati al sito possono solo visualizzare i prodotti, senza poterli acquistare. Inoltre posso contattare l’amministratore per ricevere le informazioni sui prodotti.

Attori Oggetti	Amministratore	Utente Registrato	Utente Non registro
Effettua Login/Logout	X	X	
Registrazione Utente			X
Descrizione Prodotti	X	X	X
Catalogo	X	X	X
Inserimento Prodotti	X		

Modifica Prodotti	X		
Cancellazione Prodotti	X		
Gestione Ordini	X		
Gestione Carrello		X	
Storico Prodotti		X	
Modalità di Pagamento		X	
Output ricerca Prodotti	X	X	X

3.5.2 Sicurezza

Il sito web di e-commerce “Top Hardware” è protetto da accessi indesiderati grazie ad un meccanismo di autenticazione che, tramite una combinazione di username e password fornita ad ogni utente, regola gli accessi al sistema. Il sistema analizzerà questi campi e verificherà che nel database sia presente la tupla contenente l’Username e la Password specificata, dopodiché nel caso in cui l’accesso venga negato verrà presentato all’utente un messaggio di errore e riproposto l’inserimento dei dati necessari per l’accesso, oppure, il sistema riconoscerà automaticamente quali sono i servizi destinati e permessi all’utente che si è autenticato nel sistema.

Nello specifico i dettagli della combinazione Username/Password sono:

- Mail: è richiesta un indirizzo email valido, che rispetti il formato standard.
- Password: è richiesta una lunghezza compresa fra un minimo di 6 e un massimo di 15 caratteri alfanumerici.

3.6 Gestione del controllo globale

Per consentire di impiegare appieno le potenzialità messe a disposizione dall'ambiente di sviluppo, il nostro sito web sarà gestito da un “Event Driven Control”.

Facendo riferimento a questo gestore, il flusso principale attende il verificarsi di eventi che attivano funzionalità specifiche.

Al verificarsi di un evento, il più delle volte generato da un utente tramite l'interfaccia grafica, il controllo viene trasferito al relativo oggetto che lo gestisce.

Quest'ultimo ricevuto il controllo manda in esecuzione un sottosistema.

Nella maggior parte dei casi, ogni operazione termina con un messaggio di notifica all'utente utilizzatore del Client.

3.7 Condizione di limite

Le boundary conditions di installazione, start up e shut-down saranno così definite:

- **Installazione:** richiede lato server la presenza del DBMS e del Webserver con il quale si accederà a determinati file inseriti in un particolare percorso di file. Lato client occorre la connessione ad internet e un browser compatibile con Html, CSS, Javascript, AJAX.
- **Start-up:** comporta l'avvio del DBMS e del Webserver.
- **Shut-down:** viene interrotto il servizio fornito dal DBMS e dal Webserver.

4. Servizi dei sottosistemi

Sottosistema	Descrizione
Gestione Utente	Sottosistema che gestisce la registrazione di un utente, la sua autenticazione e le operazioni necessarie alla sua gestione.
Servizio	Descrizione
doRetriveByEmailAndPassword()	Permette di visualizzare tutti gli utenti con la determinata email e password.
doRetriveByEmail()	Permette di visualizzare tutti gli utenti con la determinata email.
doRetriveByAll()	Permette di visualizzare tutti gli utenti registrati.
doRetriveById()	Permette di visualizzare tutti un utente con determinato id.
doSave()	Permette di inserire un nuovo utente nel database.

doDelete()	Permette di eliminare un utente dal database.
doUpdate()	Permette di modificare gli attributi di un utente nel database.

4.1 Gestione Utente

4.2 Gestione ordine

Sottosistema	Descrizione
Gestione ordine	Sottosistema che gestisce tutte le operazioni relative ai prodotti e alla loro gestione nel sistema.
Servizio	Descrizione
doRetrieveById()	Permette di visualizzare tutti gli ordini in base al loro id.
doRetrieveByStato()	Permette di visualizzare tutti gli ordini in base al loro stato.
doRetrieveByData_Creazione()	Permette di visualizzare tutti gli ordini in base alla loro data di creazione.
doRetrieveAll()	Permette di visualizzare tutti gli ordini effettuati.
doRetrieveByProdotto()	Permette di visualizzare tutti gli ordini che contengono un determinato prodotto.
doRetrieveByUtente()	Permette di visualizzare tutti gli ordini effettuati da un determinato utente.
doSave()	Permette di inserire un nuovo ordine nel database.
doDelete()	Permette di eliminare un ordine dal database.
doUpdate()	Permette di modificare gli attributi di un ordine nel database.
addVoceOrdine()	Aggiunge un articolo con le relative informazioni a un ordine.

deleteOrderVoice()	Elimina un articolo da un ordine.
--------------------	-----------------------------------

4.3 Gestione Catalogo

Sottosistema	Descrizione
Gestione Catalogo	Sottosistema che gestisce le operazioni e la gestione relative a un catalogo.
Servizio	Descrizione
doRetriveByNome()	Permette di visualizzare un catalogo in base al nome.
doRetriveById()	Permette di visualizzare un catalogo in base al suo id.
doSave()	Permette di inserire un nuovo catalogo nel database
doDelete()	Permette di eliminare un catalogo dal database.
doUpdate()	Permette di modificare gli attributi di un catalogo nel database.
addProduct()	Permette di aggiungere un prodotto ad un determinato catalogo.
deleteProduct()	Permette di eliminare un prodotto da un determinato catalogo.

4.4 Gestione Carrello

Sottosistema	Descrizione
Gestione Carrello	Sottosistema che gestisce le operazioni e la gestione di un carrello.
Servizio	Descrizione
addProduct()	Permette di aggiungere un prodotto al carrello.
updateProdotto()	Permette di modificare un prodotto all'interno del carrello

4.5 Gestione Marca

Sottosistema	Descrizione
Gestione Marca	Sottosistema che gestisce le operazioni e la gestione di un brand.
Servizio	Descrizione
doRetrieveById()	Permette di visualizzare un brand con un determinato id.
doRetrieveByNome()	Permette di visualizzare un brand in base al nome.
doSave()	Permette di inserire un nuovo brand nel database.
doDelete()	Permette di eliminare un brand dal database.
doUpdate()	Permette di modificare gli attributi di un brand nel database.

4.6 Gestione prodotto

Sottosistema	Descrizione
Gestione Prodotto	Sottosistema che gestisce le operazioni e la gestione relative a un prodotto.
Servizio	Descrizione
doRetriveByLikeNome()	Permette di visualizzare i prodotti che contengono la stringa all'interno del nome.
doRetrieveByNome()	Permette di visualizzare tutti i prodotti con un determinato nome.
doRetrieveAll()	Permette di visualizzare tutti i prodotti.
doRetrieveByBrand()	Permette di visualizzare tutti i prodotti con un determinato brand.
doRetrieveByBestSeller()	Permette di visualizzare i prodotti in evidenza.
doSave()	Permette di inserire un nuovo prodotto nel database.
doDelete()	Permette di eliminare un prodotto dal database.
doUpdate()	Permette di modificare gli attributi di un prodotto nel database.

5. Glossario

- TopHardware: nome del sistema che si sta sviluppando.
- Amministratore: il termine identifica la persona che gestisce la piattaforma TopHardware
- Utente loggato: il termine identifica le persone che sono correttamente loggate al sistema
- Utente non loggato: il termine identifica le persone che non sono loggate al sistema, che quindi, avranno un insieme di funzionalità disponibili più ristretto rispetto agli utenti loggati
- RAD: documento di Analisi dei Requisiti
- DBMS: sistema di gestione di basi di dati
- Database: insieme organizzato di dati persistenti