

NowIWant
System Design Document
Versione 1.0

LOGO PROGETTO



Data: 22/11/2017

Progetto: NowIWant	Versione: 1.0
Documento: System design document	Data: 22/11/2017

Coordinatore del progetto:

Nome	Matricola

Partecipanti:

Nome	Matricola
Ferrazzano Pompeo Alessio	0512102928
Citro Antonio	0512102922
Giovanni Lembo	0512103252
Robertazzi Gennaro Alessio	0512103792

Scritto da:	FPA, CA, RGA, LG
--------------------	------------------

Revision History

Data	Versione	Descrizione	Autore
20/11/2017	0.1	Inizio SDD	FPA, CA
20/11/2017	0.5	Architettura software proposta	RGA, GL
21/11/2017	1.0	Servizi dei sottosistemi, Glossario e revisione	FPA, RGA, GL, CA

Indice

1. INTRODUZIONE.....	4
1.1. Scopo del sistema	4
1.2. Obiettivi di design	4
1.2.1. Criteri di performance	4
1.2.2. Criteri di affidabilità	5
1.2.3. Criteri di manutenzione	6
1.2.4. Criteri per l'utente finale.....	6
1.3. Definizioni, acronimi ed abbreviazioni	6
1.4. Riferimenti	6
1.5. Panoramica	7
2. ARCHITETTURA SOFTWARE CORRENTE	8
3. ARCHITETTURA SOFTWARE PROPOSTA	8
3.1. Overview	9
3.2. Deomposizione del sistema.....	10
3.3. Mapping Hardware/Software	12
3.4. Gestione dei dati persistenti	13
3.5. Controllo degli accessi e della sicurezza.....	13
3.6. Controllo del software globale	14
3.7. Condizioni Boundary	14
4. SERVIZI DEI SOTTOSISTEMI.....	17
4.1. Gestione utenti	17
4.2. Gestione amministratore	18
4.3. Gestione carrello	19
5. GLOSSARIO	20

1. INTRODUZIONE

1.1. Scopo del sistema

NowIWant è un sito e-commerce specializzato nella vendita online di vestiti per tutti.

Il nostro scopo è offrire un vasto assortimento di brands e modelli con i vantaggi di un acquisto online sicuro e semplice. I nostri clienti non rimarranno mai indietro con le tendenze.

Mettiamo a disposizione dei nostri clienti un servizio completo, personalizzato e efficiente senza tempi di attesa lunghi.

Garantiamo ai nostri clienti la corretta gestione dei prodotti e degli ordini.

1.2. Obiettivi di design

Il sito NowIWant deve poter essere il più efficiente ed intuitivo possibile.

Tale efficienza sarà costruita attraverso rapidi tempi di risposta ad ogni genere di input.

Si punterà ad aver una buona manutenibilità attraverso il facile inserimento di nuovi prodotti da parte degli amministratori del sito. L'e-commerce sarà intuitivo attraverso facili interfacce e sarà facilmente accessibile sia agli utenti più esperti sia a quelli meno esperti.

1.2.1. Criteri di performance

- *Tempo di risposta*

NowIWant deve assicurare una risposta rapida alle richieste dell'utente.

Una semplice richiesta dell'utente deve essere elaborata in un tempo massimo di 10secs. nel caso in cui c'è una connessione lenta.

I tempi di risposta saranno più rapidi con una connessione più veloce.

- *Throughput*

Il sistema sarà capace di gestire diversi utenti contemporaneamente.

Non è possibile avere una stima certa del carico ma si prevede un carico mensile di 100 utenti con una media di 2 acquisti per utente.

- ***Robustezza***

NowIWant deve poter gestire anche gli input errati senza interrompere l'intero funzionamento del sistema attraverso dei messaggi di errore (es. Id e/o password errati).

- ***Disponibilità***

Il sito deve poter essere disponibile 24/7 agli utenti grazie ad un server sempre online.

- ***Tolleranza dell'errore***

Il sito deve poter continuare a "lavorare" durante delle condizioni di errore. Quest'ultimo infatti non deve intaccare le altre funzionalità del sito.

1.2.2. Criteri di affidabilità

- ***Robustezza***

Attraverso delle notifiche di errore il sito NowIWant deve poter gestire eventuali errori di input senza intoccare il funzionamento dell'intero sistema.

- ***Disponibilità***

Deve poter essere disponibile 24/7 agli utenti grazie ad un server sempre attivo.

- ***Tolleranza dell'errore***

Il sito deve essere capace di continuare a funzionare durante delle condizioni di errore. Ciò sarà reso possibile grazie all'uso di sottoparti nel sito in modo tale che l'errore verificatosi in un sottosistema non intacchi il funzionamento di altri sottosistemi.

- ***Sicurezza***

L'accesso al sistema è controllato da un sistema di login che permetterà all'utente di poter effettuare acquisti in massima sicurezza.
Il database offre sicurezza sui dati personali (es. password dell'account).

1.2.3. Criteri di manutenzione

- ***Estendibilità***

NowIWant deve essere propenso all'inserimento di nuove funzionalità. E' necessario

Che il codice scritto sia ben strutturato, quindi di facile comprensione per evitare ambiguità nella scrittura successiva. Le varie classi sono divise in pacchetti in modo che sia più facile individuare la classe da modificare.

- ***Modificabilità***

Ci deve essere la possibilità di poter effettuare eventuali modifiche al sistema (es. in caso di un bug).

Anche in questo caso il codice deve essere ben strutturato e di facile comprensione.

- ***Leggibilità***

Il codice sarà comodo da leggere grazie ad un'accurata indentazione di quest'ultimo.

1.2.4. Criteri per l'utente finale

- ***Usabilità***

NowIWant rende ogni funzione di semplice uso garantendo un'ottima esperienza all'utente grazie ad interfacce grafiche intuitive.

1.3. Definizioni, acronimi e abbreviazioni

- NowIWant: Nome del sistema che verrà sviluppato.
- RAD: **R**equirement **A**nalysis **D**ocument.
- DBMS: **D**atabase **M**enagment **S**ystem.
- Admin: Amministratore di sistema.
- Carrello: Carrello elettronico contenente i prodotti da acquistare.
- Utente: Utente registrato, o non, alla piattaforma.

1.4. Riferimenti

Documento RAD del progetto NowIWant.

1.5. Panoramica

Prima di parlare dell'architettura è importante fare un accesso al system design che costituiscono le fondamenta per l'architettura software del sistema.

- **Decomposizione del sistema:**

il sistema viene suddiviso in diversi sottoinsiemi ognuno dei quali è caratterizzato da servizi che offre ad altri sottosistemi.

- **Mapping Hardware/Software:**

riguarda la scelta della configurazione hardware del sistema, la comunicazione tra nodi ed il come vengono incapsulati i servizi di un sottosistema.

- **Gestione dei dati persistenti:**

si individuano gli oggetti che devono essere resi persistenti e quale genere di infrastruttura si deve utilizzare per memorizzare tali oggetti.

- **Politiche di accesso e sicurezza:**

che ci aiuta a rappresentare tramite delle tabelle le operazioni ed informazioni utilizzabili da ogni singolo attore.

- **Controllo del software globale:**

ci guida su quali operazioni eseguire ed in che ordine devono essere eseguite per garantire il corretto flusso di controllo del sistema.

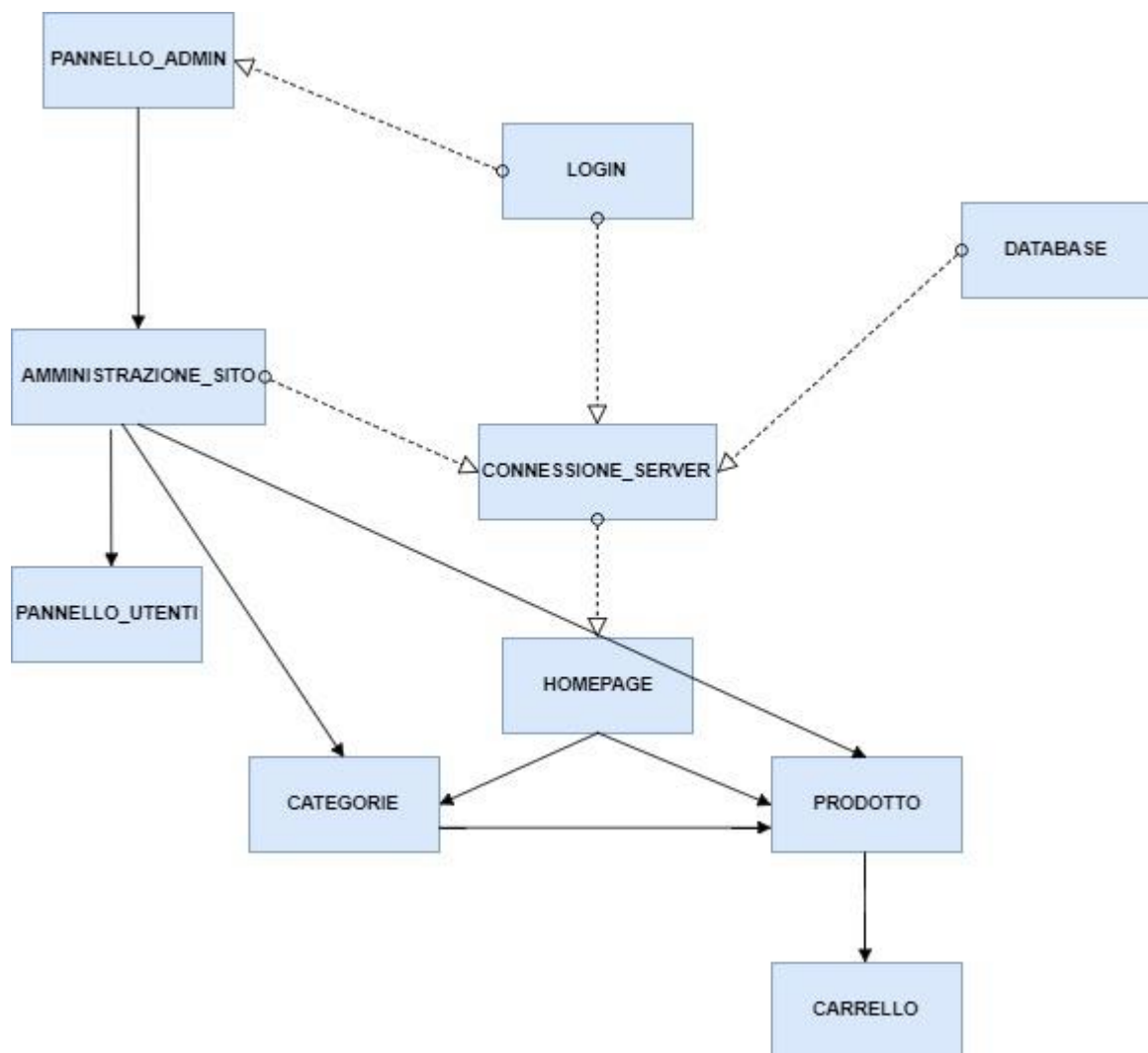
- **Condizioni Boundary:**

includono: l'avvio, lo shutdown, la gestione dei fallimenti dovuti all'invecchiamento del sistema, interruzione della corrente o anche errori di progettazione.

2. ARCHITETTURA SOFTWARE CORRENTE

Nella progettazione del sistema i progettisti si sono potuti avvalere della possibilità di confrontarsi con altre piattaforme. La concorrenza attuale è di fascia altissima in quanto ci sono infiniti siti di e-commerce in giro per il web.

3. ARCHITETTURA SOFTWARE PROPOSTA



3.1. Overview

L'architettura di NowIWant è di tipo client/server. Il server riceve le richieste da parte dei client e risponde in tempo utile.

I motivi di questa scelta sono:

-portabilità: il sistema potrà essere utilizzato su migliaia di macchine e sistemi operativi sia da pc che dal telefono.

-performance: il client sarà in grado di supportare task interattivi ed il server dovrà fornire operazioni CPU-sensitive.

-flessibilità: per ogni tipologia di utente che effettua l'accesso ci saranno delle form aggiuntive (in questo caso per un utente amministratore).

-affidabilità: entrambi i componenti client e server devono essere affidabili ed essere in grado di mantenere i propri dati anche in seguito a guasti. Per fare ciò deve essere possibile effettuare backup periodici al database.

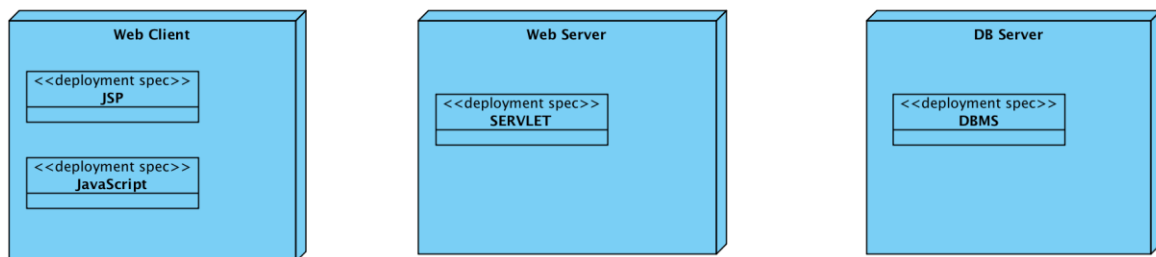
L'applicazione è strutturata da 3 livelli(three-tier):

- 1) Interface Layer
- 2) Application Logic Layer
- 3) Data Layer

Web-Client: JSP e Javascript

Web-Server: Servlet

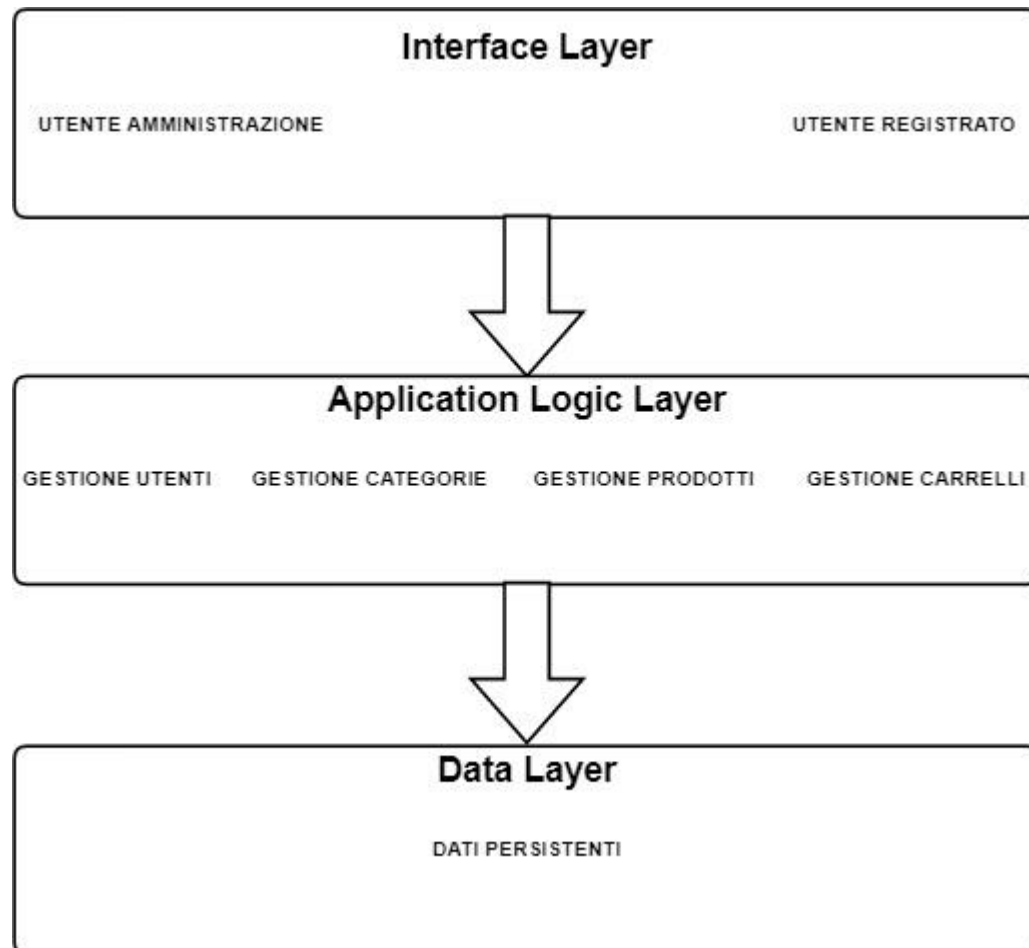
DB-Server: DBMS



- 1) L'Interface Layer rappresenta l'interfaccia che permette all'utente di poter interagire col sistema. Esso ricopre il ruolo di Client in quanto utilizza un browser per richiedere pagine web al server.
- 2) L'application Logic Layer ha il compito di elaborare i dati che vengono inviati al client. Spesso interroga il database, tramite lo Storage Layer, per accedere ai dati persistenti.
- 3) Il Data Layer ha il compito di memorizzare i dati sensibili del sistema utilizzando un DBMS. Inoltre riceve le varie richieste dal livello precedente (Application Logic Layer) inoltrandole al DBMS e restituendo i dati richiesti.

3.2. Decomposizione del sistema

Diagramma e descrizione di ogni metodo.



Interface Layer	
Utente Amministratore	Modulo che si occupa di gestire le funzionalità del sistema riservate all'amministratore, come la gestione degli utenti, delle categorie e dei prodotti. C'è la possibilità di aggiungere, modificare ed eliminare le categorie e i prodotti.
Utente visitatore del sito	Modulo che si occupa di gestire le funzionalità del sistema riservate ad un utente, come la visualizzazione dei prodotti e l'aggiunta di un prodotto al carrello per poi effettuare l'acquisto.

Application Logic Layer	
Gestione Utenti	Modulo che permette la registrazione di un nuovo utente al sistema ed inoltre gestisce l'accesso degli utenti.
Gestione Categorie	Modulo che si occupa di visualizzare, creare, modificare, eliminare le categorie dei prodotti (es. Uomo-Donna).
Gestione Prodotti	Modulo che si occupa di visualizzare, creare, modificare, eliminare i prodotti visualizzati sulla piattaforma.
Gestione Carrelli	Modulo che si occupa di visualizzare, creare, modificare, eliminare il carrello degli utenti da parte di un utente amministratore. Anche un utente normale può cancellare il suo carrello.

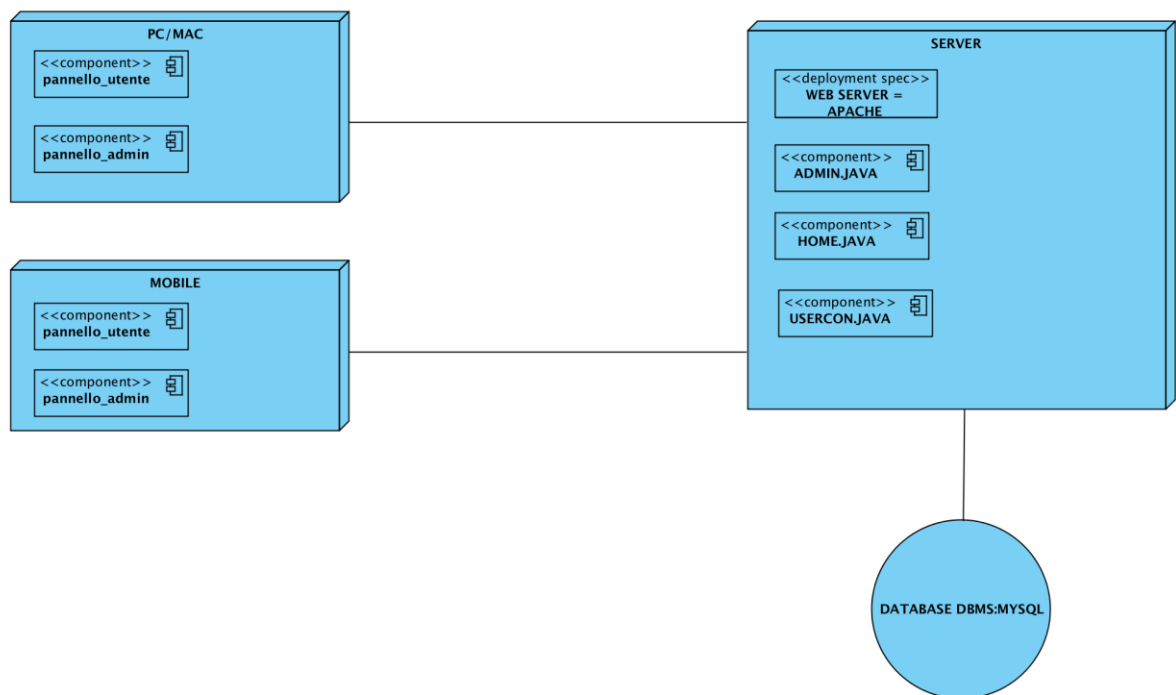
Data Layer	
Dati persistenti	Modulo che si occupa di memorizzare i dati fondamentali degli utenti e dei prodotti in memoria in modo tale da poter essere prelevati e modificati in modo corretto.

3.3. Mapping Hardware/Software

La struttura Hardware proposta è costituita da un server centrale e dalle postazioni utenti che possono essere costituite da computer portatili, e/o fissi, nei quali possono girare sistemi operativi diversi tra loro. I Client si collegano al server ed il database e le postazioni client dovranno effettuare richieste al server per eseguire le operazioni legate alle loro rispettive funzionalità.

Il database conterrà esclusivamente i dati relativi ai prodotti messi in vendita.

Le postazioni client ed il server saranno connesse tramite una rete che utilizzerà il protocollo TCP/IP.



- **Web Server**
Il server utilizzato è Apache Tomcat
- **Interface Layer**
L'utente utilizza il sistema mediante il browser installato sulla propria macchina (es. Google Chrome).
- **Application Logic Layer**
Il sistema, e quindi le funzionalità, sono implementate in linguaggio HTML. Il codice viene inviato al browser del client
- **Storage Layer**
Rappresenta il collegamento con il server da parte del sistema e si occupa di tutte le richieste di accesso, modifica ed eliminazione sui dati permanenti presenti nel database.
- **Database Server**
Il DBMS usato è HeidiSQL.

3.4. Gestione dei dati persistenti

Ci si ricollega al documento “NowIWant – System Design Document – Gestione Dati Persistenti.

3.5. Controllo degli accessi e della sicurezza

NowIWant è un sistema multiutente, quindi permette a più utenti di effettuare diverse operazioni contemporaneamente. Per schematizzare al meglio il controllo degli accessi abbiamo suddiviso, per tipologia di utente, le azioni consentite al fine di ottenere una visione più compatta e dettagliata.

- **ADMIN**

Oggetto	Operazioni
Prodotti	Inserisci Modifica Rimuovi
Utente	Inserisci Modifica Rimuovi

- **UTENTE**

Oggetto	Operazioni
Carrello	Aggiungi prodotti Elimina prodotti Acquista prodotti

3.6. Controllo del software globale

Il controllo del flusso software viene gestito da classi HTML, Java e JSP che, interagendo con il client, svolgono le varie operazioni. Il server smista le varie richieste alla classe adeguata inoltrando poi la risposta al client.

3.7. Condizioni Boundary

Le condizioni limite riguardano l'accensione e lo spegnimento del sistema per quanto riguarda il lato server. Dal lato Client si riferiscono agli errori di connessione al server.

- **SCENARI**

Nome scenario	Start Server
Istanze di attori partecipanti	Admin
Flusso di eventi	<ol style="list-style-type: none">1. Un admin decide di avviare il sistema e quindi clicca sul pulsante "Avvia"2. Il sistema, con le sue procedure di avvio, attiva il server e i relativi servizi in remoto rendendoli disponibili ad eventuali richieste.3. Il sistema notifica il successo della procedura.

Nome Scenario	Shoutdown Server
Istanze di attori partecipanti	Admin
Flusso di eventi	<ol style="list-style-type: none">1. L'admin decide di voler arrestare il sistema e quindi accede alla pagina dedicata e clicca il pulsante "Arresta" (Arresta server Tomcat).2. Il sistema controlla se ci sono ancora delle richieste in sospeso.3. Il sistema porta a termine (se esistenti) le richieste in sospeso.4. Tramite le procedure di arresto il sistema disattiva tutti i servizi ed il server.5. Il sistema notifica il successo della procedura.

- **CASI D'USO**

ID	Use Case Startup
Nome caso d'uso	Startup Server
Istanze di attori partecipanti	Admin
Condizione d'entrata	L'admin accede al sistema.
Flusso di eventi	<p>Utente: l'utente accede al sistema tramite il suo web browser.</p> <p>Sistema: NowIWant accende il server ed attiva i servizi in remoto rendendosi disponibile per le richieste notificando il successo dell'operazione all'utente.</p>
Condizione d'uscita	Il server è attivo e i relativi servizi sono disponibili.
Eccezioni	Errori nello Startup.

ID	Use case shutdown
Nome caso d'uso	Shutdown Server
Istanze di attori partecipanti	Admin
Condizione di entrata	L'amministratore accede al sistema.
Flusso di eventi	Utente: l'utente accede al sistema e preme il pulsante spegni.
Condizione di uscita	Il server si è spento correttamente.
Eccezioni	Errore shutdown.

4. SERVIZI DEI SOTTOSISTEMI

4.1. Gestione utenti

Sottosistema	Gestione utenti
Descrizione	Sottosistema che gestisce la registrazione di un utente, l'autenticazione di tutti gli utenti.
Servizi offerti	
Servizio	Descrizione
Registra utente	Permette di inserire un nuovo utente al database.
Login	Permette all'utente di poter accedere al sistema.
Logout	Permette all'utente di poter uscire dal sistema.

4.2. Gestione amministratore

Sottosistema	Gestione amministratore.
Descrizione	Sottosistema che gestisce le operazioni che un amministratore può effettuare sulla piattaforma.
Servizi offerti	
Servizio	Descrizione
Login	Permette ad un amministratore di poter effettuare l'accesso al sistema.
Logout	Permette ad un amministratore di uscire dal sistema.
Visualizza ordini	Permette di visualizzare a video tutti gli ordini effettuati dagli utenti.
Aggiungi prodotto	Permette all'amministratore di poter aggiungere un prodotto al sistema.
Rimuovi prodotto	Permette all'amministratore di poter rimuovere un prodotto dal sistema.
Modifica prodotto	Permette all'amministratore di poter modificare un prodotto presente nel sistema.
Visualizza utenti	Permette all'amministratore di poter visualizzare tutti gli utenti registrati al sistema.
...	...

Aggiungi categoria	Permette all'amministratore di poter aggiungere una categoria al sistema.
Modifica categoria	Permette all'amministratore di poter aggiungere una categoria nel sistema.
Elimina categoria	Permette all'amministratore di poter eliminare una categoria presente nel sistema.

4.2. Gestione carrello

Sottosistema	Gestione carrello.
Descrizione	Sottosistema che gestisce le operazioni che un utente può effettuare all'interno del carrello elettronico.
Servizi offerti	
Servizio	Descrizione
Elimina carrello	Permette di eliminare il carrello con tutti i prodotti al suo interno.
Salva carrello	Permette di salvare il carrello.
Vedi storico	Permette di guardare tutti gli acquisti precedenti.
Acquista prodotti	Permette di acquistare i prodotti inseriti nel carrello elettronico.
Elimina prodotto	Permette di eliminare il prodotto selezionato nel carrello.

5. GLOSSARIO

Admin: il termine identifica la persona che amministra il sito internet che può essere identificato come una persona che gestisce la configurazione del sistema.

Utente: E' l'utente del sistema che ha la possibilità di creare test esercitativi, valutarli e sottoporli agli utenti.

RAD: Documento di Analisi dei Requisiti.

Carrello: il termine identifica il mezzo mediante il quale l'utente effettua un acquisto.

DBMS: Database Management System per la gestione del database.

Database: Insieme dei dati persistenti del sistema.

Prodotto: Il termine identifica un prodotto in vendita sulla piattaforma.