

**System Design Document**

[Versione 1]

**Sommario**

[**Informazioni sul documento** 2](#_Toc156991718)

[**Generalità** 2](#_Toc156991719)

[**Team Project** 2](#_Toc156991720)

[**Revision History** 2](#_Toc156991721)

[**1. Introduzione** 4](#_Toc156991722)

[**1.1 Obiettivo del sistema** 4](#_Toc156991723)

[**1.2 Obiettivi di design (Design goals)** 5](#_Toc156991724)

[**1.3 Definizioni, acronimi e abbreviazioni** 7](#_Toc156991725)

[**1.4 Riferimenti** 8](#_Toc156991726)

[**1.5 Organizzazione del documento** 8](#_Toc156991727)

[**2. Architettura del sistema software corrente** 9](#_Toc156991728)

[**3. Architettura del sistema software proposto** 9](#_Toc156991729)

**Informazioni sul documento**

**Generalità**

* **Progetto**: TechHeaven
* **Versione**: [Versione 1]
* **Documento**: Documento di design del sistema
* **Data**: [23/01/2024]

**Team Project**

|  |  |  |  |
| --- | --- | --- | --- |
| Nome Membro | Matricola | Ruolo | Contatti |
| Dorotea Serrelli | 0512113740 | Project manager | d.serrelli1@studenti.unisa.it |
| Raffaella Sabatino | 0512115114 | Team member | r.sabatino17@studenti.unisa.it |

**Revision History**

|  |  |  |  |
| --- | --- | --- | --- |
| Data | Versione | Descrizione | Autore |
| 23/01/2024 | 0.1 | Scrittura delle sezioni “Introduzione” e “Architettura del sistema software corrente” | Tutto il team |

**1. Introduzione**

**1.1 Obiettivo del sistema**

La società TechHeavenSrl è responsabile dall’anno 2000 della gestione del negozio “TechHeaven – Il paradiso digitale”, specializzato nella vendita di prodotti elettronici, elettrodomestici, telefonia.

Tale negozio è, attualmente, un punto vendita di riferimento nella zona per lo smercio, la qualità e il prezzo dei prodotti, al punto che soddisfa un grande bacino di utenza, quasi esclusivamente residente nella provincia.

La società intende espandere i confini della propria attività ed ampliare la clientela, avvalendosi di un sistema software che consenta, sotto il profilo soggettivo, una maggiore conoscibilità della società e dell’affidabilità della stessa; sotto il profilo oggettivo, favorisca l’incremento della vendita dei prodotti.

Il sistema software verrà sviluppato per fornire alla clientela informazioni sulla società, sul punto vendita e sui prodotti trattati.

La piattaforma permetterà, infatti, al cliente di registrarsi, in modo da poter visionare i prodotti in vendita ed acquistarli, tenere traccia dello stato degli ordini effettuati presso il negozio online, creare una lista di prodotti desiderati (wishlist).

La piattaforma, inoltre, consentirà l’accesso ai seguenti dipendenti:

* Gestore degli ordini: responsabile del processo di acquisizione, registrazione ed evasione degli ordini dei clienti, nonché dell’elaborazione di richieste di approvvigionamento di prodotti da inoltrare, poi, all’ufficio acquisti.
* Gestore del catalogo: responsabile della presentazione, organizzazione e gestione del catalogo dei prodotti venduti dal negozio.

La piattaforma, quindi, consentirà al gestore degli ordini di visionare gli ordini commissionati dai clienti al negozio e gli ordini che sono stati spediti, preparare un ordine alla spedizione e fare richiesta di approvvigionamento di prodotti mancanti.

Essa, inoltre, permetterà al gestore del catalogo di visionare il catalogo e di poter inserire, cancellare e modificare un prodotto nel catalogo.

**1.2 Obiettivi di design (Design goals)**

In questa sezione sono riportati i Design Goals, ovvero le caratteristiche qualitative che deve possedere il sistema software, la cui formalizzazione esplicita sarà costantemente utilizzata come linea guida per le scelte progettuali che si affronteranno nelle attività successive.

In base alle linee guida delineate dagli autori del libro *Object Oriented Software Engineering* , Bernd Bruegge e Allen H. Dutoit, i design goal sono stati raggruppati nelle seguenti categorie:

* Performance: includono i requisiti di spazio e velocità imposti sul sistema.
* Dependability: determinano quanto sforzo deve essere speso per minimizzare i fallimenti del sistema (crash, falle di sicurezza) e le loro conseguenze.
* Maintenance: determina quanto sforzo è necessario per modificare il sistema dopo il suo rilascio.
* End User: includono qualità che sono desiderabili dal punto di vista dell’utente, ma che non sono state coperte dai criteri di Performance e Dependability.

Nella seguente tabella si riportano le informazioni dei design goals individuati dal RAD : il nome, la descrizione, la categoria di appartenenza e il requisito non funzionale da cui discende.

|  |  |  |  |
| --- | --- | --- | --- |
| Nome | Descrizione | Categoria | RNF di origine |
| DG\_1  Quantità dei dati | Il sistema dovrà garantire un’alta capacità di memorizzazione per poter conservare tutte le informazioni necessarie per l’erogazione dei servizi che offre. | Performance | RNF\_11 |
| DG\_2  Gestione permessi sistema | Il sistema deve garantire la separazione netta delle operazioni, sulla base degli utenti che possono effettuarle. | Dependability | RNF\_3 |
| DG\_3  Affidabilità | Il sistema deve garantire che tutte le operazioni avvengano con successo. | Dependability | RNF\_4 |
| DG\_4  Robustezza | Il sistema deve essere in grado di gestire l’inserimento di dati non validi da parte dell’utente, generando messaggi di errore. | Dependability | RNF\_5 |
| DG\_5  Crittografia password utente | Il sistema deve memorizzare la password di accesso di un utente non in chiaro, ossia applicare su di essa un algoritmo crittografico. | Dependability | RNF\_6 |
| DG\_6  Protezione da attacchi SQLInjection | Il sistema deve essere protetto da attacchi di tipo SQLInjection. | Dependability | RNF\_7 |
| DG\_7  Protezione da Cross-site scripting | Il sistema deve difendersi da attacchi di tipo Cross-site scripting. | Dependability | RNF\_8 |
| DG\_8  Protocollo HTTPS | Il sistema deve proteggere l’integrità e la riservatezza dei dati scambiati tra client e server, mediante l’uso di un certificato SSL/TLS. | Dependability | RNF\_9 |
| DG\_9  Portabilità | Il sistema sarà disponibile tramite pagina web da qualsiasi dispositivo (principalmente tablet, smartphone, computer). Ciò rende il sistema altamente portabile in quanto non è necessario adattare il sistema tra diversi sistemi operativi. | Dependability | RNF\_18  RNF\_14 |
| DG\_10  Manutenibilità | Il sistema dovrà essere sviluppato seguendo i principali standards per la buona manutenibilità. | Maintenance | RNF\_12 |
| DG\_11  Estendibilità | Il sistema dovrà essere sviluppato seguendo i principali standards che garantiscono una buona estendibilità delle funzionalità. | Maintenance | RNF\_13 |
| DG\_12  Facilità di utilizzo | Il sistema deve essere facilmente utilizzabile anche da un utente non esperto. | End user | RNF\_1 |
| DG\_13  Interfaccia intuitiva | L’interfaccia utente della piattaforma deve permettere di eseguire azioni in modo chiaro e semplice, rendendo ben esplicita la funzionalità di ogni elemento visuale. | End user | RNF\_2 |
| DG\_14  Sistema responsive | Il sistema dovrà offrire un’interfaccia grafica utente che sia in grado di adattarsi, principalmente, allo schermo di uno dei seguenti dispositivi: smartphone, tablet e computer. | End user | RNF\_10 |

**1.3 Definizioni, acronimi e abbreviazioni**

In questa sezione si riportano di seguito l’insieme delle definizioni, degli acronimi e delle abbreviazioni che verranno utilizzati in questo documento:

* Design goals: le caratteristiche qualitative che il sistema software deve possedere, ottenute dal dominio applicativo e dal cliente/dagli utenti finali;
* Sottosistema: una parte sostituibile del sistema software che offre un sottoinsieme di servizi che soddisfano una o più funzionalità definite nel RAD;
* Dati persistenti: i dati che sopravvivono alla singola esecuzione del sistema e che, dunque, devono essere memorizzati;
* Mapping hardware/software: lo studio della connessione tra le componenri logiche e le compnenti fisiche che costituiscono il sistema;
* RAD: Requirements Analysis Document;
* SDD: System Design Document.

**1.4 Riferimenti**

Di seguito si elencano i documenti utili alla comprensione del documento:

* Problem statement;
* RAD;
* …

**1.5 Organizzazione del documento**

Il presente documento di System Design consta di quattro sezioni:

1. **Introduzione**: si descrivono lo scopo del sistema, gli obiettivi di design su cui il sistema deve essere svluppato, le definizioni e gli acronimi utilizzati nel documento ed i riferimenti agli altri documenti rilasciati.
2. **Architettura del sistema software corrente**: si delinea lo stato attuale dell’architettura del software già presente.
3. **Architettura del sistema software proposto**: si illustra come il sistema verrà decomposto nei sottosistemi, il mapping Hardware/Software dei sottosistemi, la gestione dei dati persistenti, le politiche di sicurezza e di gestione degli accessi adottate e il flusso di controllo del sistema; inoltre, verranno presentate la struttura dei singoli sottosistemi e le boundary conditions riguardanti l’intero sistema.
4. **Glossario:** riporta la lista dei termini usati nel documento con annessa spiegazione.

**2. Architettura del sistema software corrente**

La società, al momento, non presenta un sistema informatico in grado di raccogliere gli ordini dei clienti in modo automatico, bensì prende in carico solo gli ordini effettuati da clienti che risiedono nella regione in cui è localizzato il negozio.

Attualmente, gli ordini richiesti dai clienti vengono inviati al negozio tramite telefono o fax.

Come si evince dal contesto appena descritto, questa modalità di gestione degli ordini comporta perdita di tempo per la raccolta dei dati ed oneri aggiuntivi per la relativa gestione, a discapito della produttività dell’azienda.

**3. Architettura del sistema software proposto**