



Università degli Studi Federico II

Facoltà di Informatica
Corso di Ingegneria del Software

DietiEstates25

Documentazione del Progetto di Ingegneria Del Software

Autori:

Nome Cognome 1

Nome Cognome 2

Anno Accademico 2024/2025

Contents

1	Introduzione	2
1.1	Scopo del Documento	2
1.2	Contesto	2
1.3	Struttura	2
2	Documento dei Requisiti	4
2.1	Modellazione dei casi d'uso	4
2.1.1	Use Case Diagram	5
2.2	Individuazione degli utenti target e Personas	5
3	Architettura del Sistema	6
4	Implementazione	7
5	Conclusioni	8

1. Introduzione

1.1. Scopo del Documento

Il presente documento è stato redatto con l'obiettivo di delineare i requisiti necessari per la progettazione e realizzazione del sistema DietiEstates25. Si tratta di una piattaforma software progettata per ottimizzare la gestione dei servizi immobiliari, offrendo uno strumento di interazione tra agenzie immobiliari e utenti finali. Il sistema mira a semplificare le operazioni di gestione e comunicazione, migliorando l'esperienza degli utenti. Il documento rappresenta un riferimento fondamentale per tutte le fasi di realizzazione del progetto, costituendo una base solida per la progettazione, implementazione e verifica del sistema. Ogni sezione è stata redatta seguendo i principi dell'Ingegneria del Software, assicurando chiarezza e completezza dei requisiti.

1.2. Contesto

Il settore immobiliare rappresenta uno dei pilastri dell'economia globale, caratterizzato da una forte dinamicità e da una crescente necessità di modernizzazione attraverso strumenti digitali. Tuttavia, l'implementazione di sistemi software in questo scenario, comporta alcune sfide significative, come la necessità di garantire sicurezza e protezione dei dati sensibili, mantenere alte prestazioni durante i picchi di utilizzo e offrire un'esperienza utente intuitiva. Inoltre, l'integrazione di servizi esterni, come API per mappe o previsioni meteo, aggiunge complessità alla progettazione e alla manutenzione del sistema. DietiEstates25 è progettata per operare al meglio all'interno di tale contesto, offrendo una piattaforma digitale avanzata, scalabile e sicura, progettata per migliorare l'esperienza utente e supportare la gestione delle operazioni immobiliari. Grazie alla sua modularità, il sistema permette un'elevata flessibilità nell'evoluzione futura delle funzionalità, rispondendo così alle esigenze in continua trasformazione del settore immobiliare.

1.3. Struttura

Il documento è strutturato in modo tale da rendere fluida e piacevole la lettura e l'analisi della realizzazione del progetto. Di seguito, sono presentate le varie sezioni:

- **Analisi dei requisiti**, che include un glossario, la descrizione del target di utenti, i diagrammi UML e la formalizzazione dei requisiti. Aspetti formalizzati nel Documento dei Requisiti Software.

- **Progettazione del sistema**, che dettaglia le scelte architetturali e tecnologiche, i modelli UML e lo schema di persistenza dei dati.
- **Implementazione e testing**, che documenta il codice sorgente, le strategie di testing automatico e le valutazioni di usabilità.

2. Documento dei Requisiti

Descrizione dei requisiti funzionali e non funzionali. e tutta la parte che viene prima

2.1. Modellazione dei casi d'uso

I casi d'uso sono uno strumento fondamentale nell'ingegneria del software, poiché rappresentano in modo chiaro e sistematico le interazioni tra gli utenti e il sistema. Attraverso tale modello, è possibile definire quali funzionalità il sistema deve offrire e come queste saranno utilizzate dai diversi attori. Questa tecnica consente al team di sviluppo di avere una visione condivisa del funzionamento del sistema, aiutando a individuare i requisiti in modo completo e a progettare soluzioni che rispondano alle necessità degli utenti. Un caso d'uso descrive un flusso di interazione tra un attore e il software, con l'obiettivo di raggiungere un risultato specifico, aiutando non solo a definire le funzionalità principali, ma anche a identificare i confini del sistema, evidenziando dove sono richiesti interventi tecnologici o decisioni progettuali. Nei progetti complessi, è facile perdere di vista quali siano le funzionalità essenziali e quali invece possano essere rimandate. I casi d'uso aiutano a identificare i processi critici su cui concentrarsi per garantire che il sistema soddisfi le esigenze fondamentali degli utenti. Infine, hanno un ruolo chiave anche nella fase di testing del software: ognuno di essi, rappresenta uno scenario reale che il sistema deve essere in grado di gestire. Questo consente di progettare test basati su situazioni pratiche, garantendo che il sistema sia robusto e risponda correttamente alle interazioni previste.

Nel contesto del nostro sistema, l'attore principale è sicuramente l'utente, ossia colui che interagisce direttamente con il sistema per raggiungere obiettivi concreti come cercare un immobile, prenotare visite o fare offerte. Questo attore rappresenta il fulcro delle interazioni e le sue esigenze guidano la progettazione delle funzionalità principali del sistema. Accanto all'attore principale, ci sono anche altre figure secondarie che supportano e influenzano l'operatività del sistema:

- Admin (Amministratore): figura che gestisce il sistema a livello globale o supervisione. Si occupa di configurazioni di alto livello, come la gestione degli agenti immobiliari o l'intervento su eventuali problemi operativi.
- Agente Immobiliare: l'attore responsabile della gestione delle inserzioni immobiliari. Interagisce con il sistema per aggiungere, modificare o eliminare annunci, gestire prenotazioni di visite e rispondere alle offerte dei clienti.

- Servizi API Esterni (Google, Facebook, GitHub): questi servizi fungono da attori secondari tecnici, fornendo funzionalità essenziali come l'autenticazione tramite social network. Consentono agli utenti di accedere al sistema utilizzando le proprie credenziali già esistenti, semplificando il processo di registrazione e login.

Di seguito, viene riportato l'intero "Use Case Diagram" progettato per DietiEstates25.

2.1.1. Use Case Diagram

2.2. Individuazione degli utenti target e Personas

Quando si progetta un sistema software, il rischio più grande è creare un prodotto che, pur tecnicamente avanzato, non soddisfi pienamente le esigenze degli utenti a cui è rivolto. Per evitare questo, le **personas** diventano uno strumento essenziale. Grazie a queste rappresentazioni realistiche degli utenti, il team di sviluppo può mantenere sempre al centro dell'attenzione chi realmente utilizzerà il sistema, orientando ogni decisione progettuale, al fine di rendere l'esperienza di utilizzo quanto più intuitiva, semplice e soddisfacente. Tali infatti, offrono una guida concreta per il design e lo sviluppo del sistema. Non si tratta solo di sapere chi sono gli utenti, ma di comprendere a fondo come interagiranno con il prodotto, quali difficoltà potrebbero incontrare e quali funzionalità possono semplificare il loro percorso. Ogni persona include dettagli come:

- Nome e ruolo,
- Età e professione,
- Obiettivi e criticità,
- Caratteristiche personali,
- Competenze tecniche.

In definitiva, sono degli strumenti attraverso cui il design e lo sviluppo vengono guidati in una direzione precisa. Ogni decisione tecnologica e funzionale, dalla realizzazione dell'interfaccia utente alla logica di business, viene influenzata dalla comprensione delle esigenze e degli obiettivi reali dei consumatori finali. Questo non solo aumenta l'efficacia del sistema, ma garantisce anche che l'utente percepisca il prodotto come uno strumento pensato appositamente per lui.

3. Architettura del Sistema

Spiegazione dell'architettura.

4. Implementazione

Dettagli sull'implementazione.

5. Conclusioni

Conclusioni e sviluppi futuri.