|  |
| --- |
| A logo with numbers and letters  Description automatically generated  **Università degli Studi di Salerno Corso di Ingegneria del Software** |

**Rated  
System Design Document  
Versione 1.0**

**A black and white logo

Description automatically generated**

Data: 24/11/2024

**Coordinatore del progetto:**

|  |  |
| --- | --- |
| Nome | Matricola |
|  |  |
|  |  |

**Partecipanti:**

|  |  |
| --- | --- |
| Nome | Matricola |
| Francesco Rao | 0512116836 |
| Bruno Nesticò | 0512117268 |
|  |  |
|  |  |
|  |  |
|  |  |

|  |  |
| --- | --- |
| **Scritto da:** | Francesco Rao, Bruno Nesticò |

**Revision History**

|  |  |  |  |
| --- | --- | --- | --- |
| Data | Versione | Descrizione | Autore |
| 24/11/2024 | 1.0 | Prima stesura completa | Francesco Rao, Bruno Nesticò |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

**Indice**

1. Introduzione 4

1.1 Scopo del Sistema 4

1.2 Obiettivi di progettazione 4

1.2.1 Criteri di usabilità 4

1.2.2 Criteri di affidabilità 4

1.3 Riferimenti 4

2. Architettura Software Proposta 5

2.1 Decomposizione in Sottosistemi 5

2.2 Mapping Hardware/Software 6

2.3 Gestione dei dati persistenti 7

2.4 Controllo degli accessi e sicurezza 8

2.5 Controllo Software 9

2.6 Condizioni di confine 10

2.7 Servizi dei Sottosistemi 11

**1. Introduzione**

**1.1. Scopo del sistema**

Il sistema “Rated” è concepito come una piattaforma web dedicata agli appassionati di cinema, offrendo loro un ambiente per condividere recensioni, valutare i contributi della community e interagire con altri utenti. Gli utenti possono: Pubblicare recensioni sui film, valutare i contenuti pubblicati da altri membri e godere di un sistema reputazionale che premia i recensori più attivi e apprezzati, favorendo la visibilità dei contenuti di qualità.

La gestione della piattaforma è supportata da figure dedicate, come i Gestori del catalogo, responsabili dell’aggiornamento continuo dell’offerta di film, e i Moderatori, incaricati di garantire un ambiente rispettoso e contenuti conformi alle linee guida.

**1.2. Obiettivi di progettazione**

**1.2.1 Criteri di usabilità**

Per garantire un’esperienza utente ottimale, la piattaforma adotterà i seguenti criteri di usabilità:

*Validazione degli input*: saranno implementati meccanismi per prevenire errori durante l’inserimento di dati. Messaggi di errore chiari guideranno l’utente nella correzione dei valori errati.

*Design responsive*: l’interfaccia sarà ottimizzata per adattarsi a diversi dispositivi (PC, tablet, smartphone), rendendo l’esperienza uniforme e accessibile.

*Navigazione intuitiva*: ogni pagina presenterà una barra di navigazione per facilitare l’accesso rapido alle diverse sezioni.

**1.2.2 Criteri di affidabilità**

Il sistema garantirà robustezza e protezione attraverso:

*Controllo avanzato degli input****:*** oltre alla validazione primaria, saranno effettuati ulteriori controlli per gestire scenari non previsti e prevenire errori critici.

*Sicurezza dei dati****:*** saranno adottati protocolli di crittografia per proteggere le informazioni sensibili, come le credenziali degli utenti. Le password dovranno rispettare requisiti di complessità (es. lunghezza minima e inclusione di caratteri speciali).

*Connessioni protette****:*** l’intero sistema utilizzerà protocolli HTTPS per garantire comunicazioni sicure.

**1.3 Riferimenti**

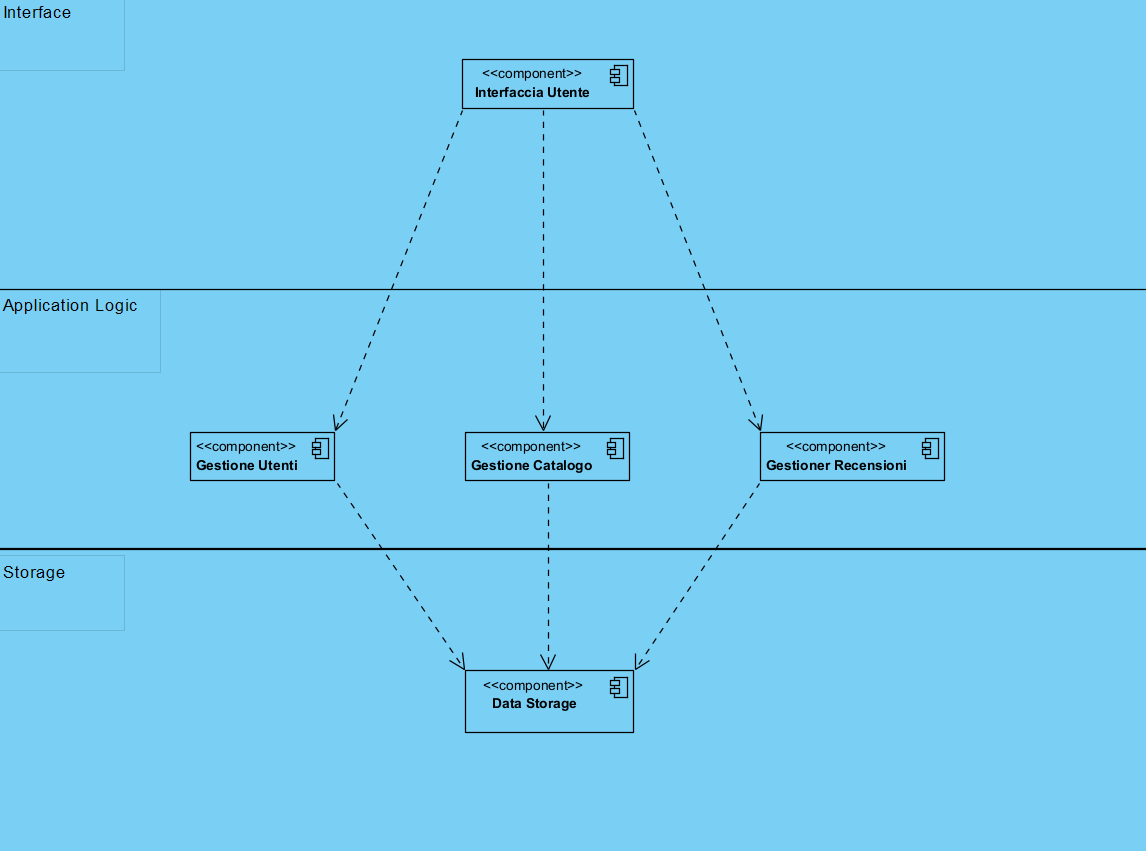
RequirementAnalysisDocument\_Rated

**2. Architettura Software Proposta**

**2.1 Decomposizine in Sottosistemi**

l sistema prodotto sarà caratterizzato da una arhitettura a tre livelli: Livello Interface, Livello Application Logic e Livello Storage.

**Component Diagram**

****

**Interface**

L’Interfaccia Utente è il sottosistema responsabile dell’interazione con l’utente finale. Fornisce le classi e le operazioni necessarie per acquisire input, visualizzare output e interagire con i servizi della piattaforma. Si occupa inoltre di:

*Coordinare la logica applicativa,* delegando le operazioni principali ai sottosistemi del livello sottostante.

*Validare i dati in ingresso*, assicurando la conformità alle specifiche del sistema e restituendo eventuali messaggi di errore in caso di input errati.

**Application Logic**

La Logica Applicativa è suddivisa in tre sottosistemi principali:

*Gestione Utenti:* si occupa dell’autenticazione, della creazione e dell’eliminazione degli account. Garantisce la protezione delle credenziali attraverso l’adozione di protocolli sicuri per la gestione dei dati sensibili.

*Gestione Catalogo:* consente ai Gestori di inserire, aggiornare e rimuovere film dal catalogo, mantenendo un’offerta costantemente aggiornata e coerente con le preferenze della community.

*Gestione Recensioni:* supporta l’aggiunta, la valutazione e la moderazione delle recensioni. Questo include meccanismi per segnalare contenuti inappropriati e l’assegnazione di punteggi che influenzano il sistema reputazionale.

**Storage**  
Il Data Storage rappresenta il sottosistema responsabile della memorizzazione e gestione degli oggetti persistenti, come:

*Dati utente:* informazioni personali, credenziali e preferenze.

*Dati di catalogo:* elenco dei film disponibili con relative informazioni.

*Dati delle recensioni:* contenuti testuali, valutazioni e metadati associati.

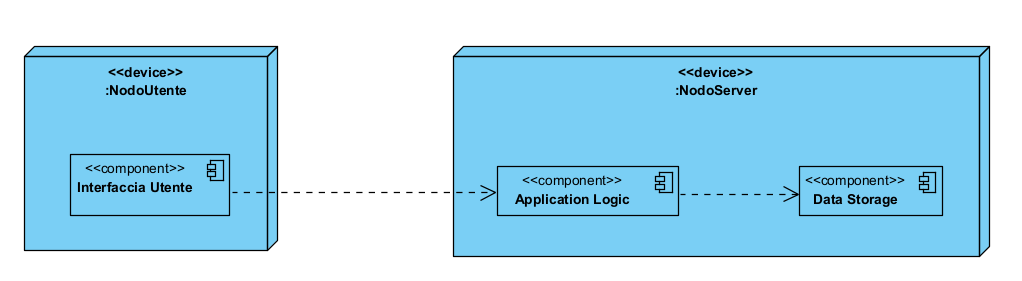
La persistenza dei dati è garantita attraverso l’utilizzo di un database relazionale ottimizzato per supportare operazioni frequenti e simultanee.

Il sistema adotta un’**architettura a strati chiusa**, in cui ogni livello può comunicare solo con lo strato immediatamente inferiore. Questa configurazione favorisce:

* *Alta manutenibilità:* le modifiche a uno strato non influenzano direttamente gli altri, rendendo più semplice la gestione del codice.
* *Flessibilità:* la separazione dei sottosistemi consente di aggiornare o sostituire componenti senza compromettere l’intero sistema.

**2.2 Mapping Hardware/Software**

**Deployment Diagram**

****

**2.3 Gestione dei dati persistenti**