|  |
| --- |
| 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 |
| 16/12/2024 | 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

1.3 Linee guida per la documentazione dell’interfaccia 4

1.4 Ottimizzazione del modello a oggetti 4

2. Packages 6

2.1 Struttura del progetto 6

3. Interfaccia delle classi 9

4. Design Pattern 17

**1. Introduzione**

**1.1. Scopo del sistema**

Lo scopo del sistema “Rated” è offrire 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. Le operazioni principali saranno: 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 sarà affidata a 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**

**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.

**Riusabilità**

Il sistema garantirà la riusabilità, tramite l’ereditarietà fornita dal paradigma di programmazine object oriented e tramite l’utilizzo di diversi design pattern.

**Affidabilità**

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

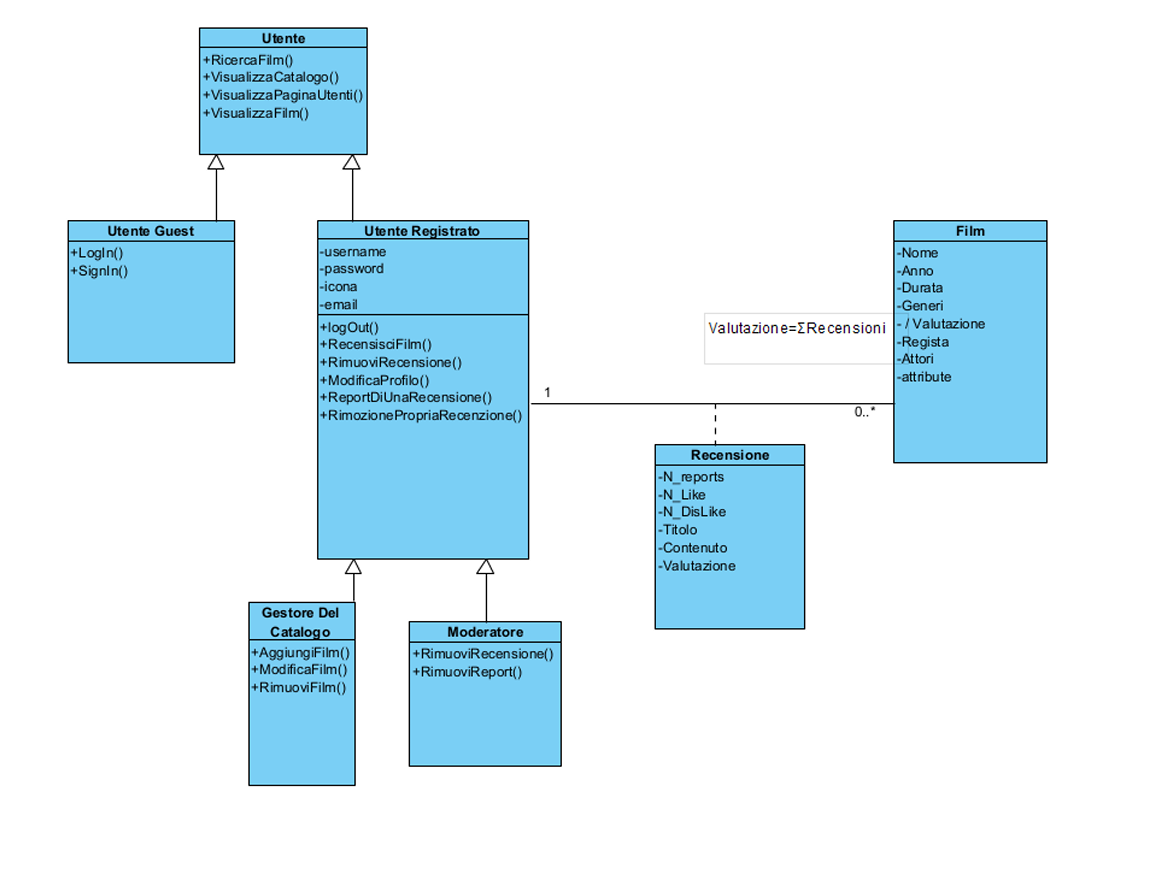
*Sicurezza dei dati****:*** saranno adottati protocolli di crittografia per proteggere le informazioni sensibili, come le credenziali degli utenti. Le password dovranno rispettare requisiti di complessità (es. lunghezza minima e inclusione di caratteri speciali).

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

**1.3 Linee guida per la documentazione dell’interfaccia**

Le convenzioni utilizzate nell’implementazione delle funzionalità del sistema faranno riferimento alle specifiche di Java Sun.

**1.4 Ottimizzazione del modello a oggetti**



Il modello ad oggetti è rimasto invariato rispetto alla versione proposta nel RAD in quanto non sono stati trovati errori o imprecisioni.

**2. Packages**

**2.1. Struttura del progetto**

Di seguito viene indicata la struttura organizzativa di file e cartelle che compongono la parte implementativa del sistema.

**.mvn**, contenente i file di configurazione di Maven

**src**, contente i file sorgente del progetto

**main**

**java,** contentente I package files di Java

**resources,** contenente le risorse relative all’interfaccia utente

**static,** contenente le risorse statiche come CSS e JS files

**templates,** contenente file HTML dinamici

**test**

**java,** contenente le classi di testing

**target,** contenente i file build i Maven

Il sistema è formato da un package generale chiamato *it.unisa.rated* all’interno del quale sarà presente un package per ogni sottosistema individuato. Oltre a questi sarà presente anche un package model , contenente le class entity e i DAO.

