



**Università degli Studi di Salerno**  
Corso di Ingegneria del Software

**CineNow**  
**ODD: Object Design Document**  
**Versione 1.0**



Data: 15/12/2024

|                                   |                  |
|-----------------------------------|------------------|
| Progetto: CineNow                 | Versione: 1.0    |
| Documento: Object Design Document | Data: 15/12/2024 |

**Coordinatore del progetto:**

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

**Partecipanti:**

| Nome              | Matricola  |
|-------------------|------------|
| Emanuele Iovane   | 0512120565 |
| Armando Vigliotti | 0512117739 |
| Antonio Caiazzo   | 0512117751 |
|                   |            |
|                   |            |
|                   |            |

|                    |                                                     |
|--------------------|-----------------------------------------------------|
| <b>Scritto da:</b> | Caiazzo Antonio, Iovane Emanuele, Vigliotti Armando |
|--------------------|-----------------------------------------------------|

**Revision History**

| Data       | Versione | Descrizione                                                                                                | Autore            |
|------------|----------|------------------------------------------------------------------------------------------------------------|-------------------|
| 02/12/2024 | 0.1      | Creazione dell'Object Design Document                                                                      | Tutto il team     |
| 03/12/2024 | 0.2      | Definizione del Packaging CineNow                                                                          | Emanuele Iovane   |
| 07/12/2024 | 0.3      | Creazione dell'Introduzione, Object design trade-offs, Interface documentation guidelines, Design Pattern. | Antonio Caiazzo   |
| 13/12/2024 | 0.4      | Definizione contratti delle classi del sistema.                                                            | Armando Vigliotti |
| 15/12/2024 | 1.0      | Revisione per primo rilascio.                                                                              | Tutto il team     |
|            |          |                                                                                                            |                   |
|            |          |                                                                                                            |                   |
|            |          |                                                                                                            |                   |

|                                   |                  |
|-----------------------------------|------------------|
| Progetto: CineNow                 | Versione: 1.0    |
| Documento: Object Design Document | Data: 15/12/2024 |

## Indice

|                                              |           |
|----------------------------------------------|-----------|
| <b>1. INTRODUZIONE</b>                       | <b>4</b>  |
| 1.1. Object design trade-offs                | 4         |
| 1.2. Interface documentation guidelines      | 5         |
| 1.3. Design Pattern                          | 6         |
| 1.4. Definizioni, acronimi e abbreviazioni   | 8         |
| 1.5. Riferimenti                             | 8         |
| <b>2. PACKAGES</b>                           | <b>9</b>  |
| 2.1 Panoramica                               | 9         |
| 2.2 Packaging del sistema                    | 9         |
| 2.3 Packaging dei layer                      | 10        |
| 2.4 Dettaglio dei singoli package            | 12        |
| 2.4.1 Package Application                    | 12        |
| Package Application: model                   | 12        |
| Package Application: gestione_programmazione | 13        |
| Package Application: gestione_registrazione  | 13        |
| Package Application: gestione_prenotazione   | 14        |
| Package Application: gestione_catalogo       | 14        |
| Package Application: gestione_sale           | 15        |
| Package Application: gestione_accesso        | 15        |
| Package Application: gestione_sede           | 16        |
| Package Application: utilities               | 16        |
| 2.4.2 Package DataAccess                     | 17        |
| Package DataAccess: DAO                      | 17        |
| Package DataAccess: database_connection      | 17        |
| <b>3. CLASS INTERFACES</b>                   | <b>18</b> |
| 3.1 Panoramica                               | 18        |
| 3.2 DataAccess                               | 18        |
| <b>4. GLOSSARIO</b>                          | <b>36</b> |

|                                   |                  |
|-----------------------------------|------------------|
| Progetto: CineNow                 | Versione: 1.0    |
| Documento: Object Design Document | Data: 15/12/2024 |

# 1. INTRODUZIONE

Il presente Object Design Document ha l'obiettivo di approfondire in modo dettagliato gli aspetti relativi all'implementazione del sistema CineNow, completando e ampliando quanto definito nei precedenti documenti, che si sono concentrati prevalentemente sull'architettura e progettazione generale del sistema. Questo documento fornisce una descrizione tecnica e completa delle decisioni progettuali prese durante le fasi di analisi e design, specificando i principali trade-off considerati, le linee guida per la documentazione delle interfacce e l'identificazione dei Design Pattern adottati.

Nel corso del documento verranno inoltre definiti i packages, le interfacce delle classi e i diagrammi delle classi che rappresentano le principali decisioni implementative del sistema. Particolare attenzione sarà dedicata alla descrizione dettagliata delle operazioni, dei parametri e delle firme delle varie interfacce e classi, assicurando una coerenza con i sottosistemi individuati nel System Design Document ([SDD\\_CineNow](#)) e con i requisiti funzionali e non funzionali descritti nel Requirements Analysis Document ([RAD\\_CineNow](#)).

La funzionalità dell'Object Design Document è fornire un modello preciso e utilizzabile per implementare tutte le funzionalità identificate, favorendo una transizione fluida verso la fase di implementazione e garantendo la manutenibilità e l'espandibilità del sistema.

## 1.1. *Object design trade-offs*

### **Funzionalità Avanzate vs. Tempi di Sviluppo:**

Come descritto nel RAD e nell'SDD, alcune funzionalità secondarie, come la cancellazione delle prenotazioni oppure l'inserimento del codice (OTP) di registrazione, non saranno implementate nella prima versione del sistema per rispettare i tempi di consegna. L'attenzione sarà focalizzata sulle funzionalità essenziali, quali la gestione delle prenotazioni, l'autenticazione degli utenti e la visualizzazione delle proiezioni. Le funzionalità avanzate verranno considerate in fasi di sviluppo successive.

### **Sicurezza vs. Efficienza:**

Considerando i tempi stretti di sviluppo, verranno implementati sistemi di sicurezza essenziali, come l'autenticazione basata su username e password crittografati, e una gestione degli accessi in base ai ruoli definiti nel RAD (cliente, gestore sede, gestore catena). Tuttavia, per mantenere l'efficienza dello sviluppo, non saranno inclusi meccanismi avanzati come l'autenticazione a due fattori. Questo compromesso garantisce un livello di sicurezza adeguato per proteggere i dati sensibili, bilanciando al contempo la velocità e la semplicità di implementazione per rispettare le scadenze.

|                                   |                  |
|-----------------------------------|------------------|
| Progetto: CineNow                 | Versione: 1.0    |
| Documento: Object Design Document | Data: 15/12/2024 |

## Comprensibilità vs. Tempo:

A causa dei tempi di sviluppo limitati, non verrà data priorità alla creazione di commenti esaustivi per ogni metodo o funzione. Tuttavia, saranno forniti commenti per le principali classi, interfacce e metodi critici al fine di garantire una comprensione di alto livello del sistema. Questa scelta consente di accelerare il processo di implementazione, mantenendo comunque una documentazione sufficiente per supportare le attività di sviluppo e manutenzione. Il compromesso consente di rispettare le scadenze stabilite sacrificando una documentazione approfondita, ma senza compromettere la leggibilità del codice per le parti essenziali del sistema.

## 1.2. Interface documentation guidelines

**Nomi dei Package:** I nomi dei package nel sistema CineNow devono essere scritti in minuscolo e non devono contenere spazi o caratteri speciali. In caso di nomi composti da più parole, è necessario utilizzare il formato *snake\_case* (es. **gestione\_sale**, **gestione\_utenti**).

**Nomi delle Classi:** I nomi delle classi devono seguire il formato *PascalCase*, iniziando con una lettera maiuscola. Devono essere descrittivi e rappresentare chiaramente l'entità o la funzionalità implementata.

**Classi DAO:** Devono terminare con il suffisso **DAO** per indicare il loro ruolo di accesso ai dati (ad esempio, **PrenotazioneDAO**, **UtenteDAO**).

**Classi Bean:** Devono terminare con il suffisso **Bean**, utilizzato per rappresentare entità del dominio (ad esempio, **UtenteBean**, **FilmBean**).

**Classi di gestione:** Devono indicare chiaramente la funzionalità principale (ad esempio, **GestionePrenotazioniControl**).

**Nomi delle Servlet:** I nomi delle Servlet non utilizzate per la logica di business, devono seguire il formato *PascalCase* e terminare con il suffisso **Servlet** (ad esempio, **CatalogoServlet**).

**Nomi dei Metodi:** I metodi devono avere nomi descrittivi che rappresentino chiaramente l'operazione eseguita. I nomi devono iniziare con una lettera minuscola e seguire il formato *camelCase* (ad esempio, **calcolaTotale()**, **aggiungiProiezione()**).

**Nomi delle Variabili:** I nomi delle variabili devono essere descrittivi e seguire il formato *camelCase*. Devono iniziare con una lettera minuscola. Per nomi composti, ogni parola interna deve iniziare con una lettera maiuscola (es. **numeroPosti**, **nomeFilm**). Possono essere utilizzate variabili temporanee ed esse possono essere abbreviate (es. **i**), ma devono essere limitate al contesto in cui sono utilizzate come ad esempio in un ciclo **for**.

|                                   |                  |
|-----------------------------------|------------------|
| Progetto: CineNow                 | Versione: 1.0    |
| Documento: Object Design Document | Data: 15/12/2024 |

**Nomi delle Costanti:** I nomi delle costanti devono essere scritti in maiuscolo, con le parole separate da underscore ( ) (ad esempio, **POSTI\_DISPONIBILI=10**).

**Nomi delle Eccezioni:** Le eccezioni devono avere nomi descrittivi che indichino chiaramente il problema che segnalano. Devono terminare con il suffisso **Exception**(ad esempio, **PostoNonDisponibileException**, **UtenteNonAutorizzatoException**).

**Nomi delle interfacce:** Le interfacce devono seguire il formato *PascalCase*, come le classi, e terminare con il suffisso **Interface**. Devono rappresentare chiaramente il comportamento che definiscono(ad esempio, **PrenotazioneInterface**).

**Nomi delle JSP:** I file JSP devono seguire il formato *camelCase* e riflettere chiaramente il contenuto della pagina(ad esempio, **loginView.jsp**, **storicoOrdiniView.jsp**).

**Nomi delle classi che implementano Strategy:** Le classi che implementano il Pattern Strategy, devono seguire il formato *PascalCase* , e devono terminare con la parola **Validator** (ad esempio, **PasswordValidator**)

#### Organizzazione delle Componenti:

- **Classi e Package:** Tutte le classi che realizzano un sottosistema devono essere racchiuse nello stesso package Java, organizzate in modo logico in base alla loro funzione in modo da mantenere un sistema coeso.
- **Risorse Statiche:** Fogli di stile, script e immagini devono essere organizzati nella directory **resources**, con sottocartelle per ciascun tipo di file (es. **resources/css**, **resources/js**, **resources/images**).

### 1.3. Design Pattern

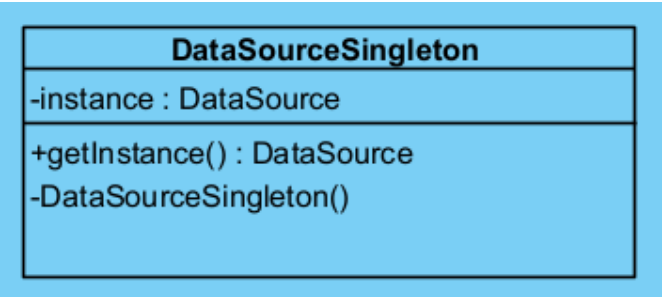
Per implementare le funzionalità del sistema CineNow, sono stati adottati due design pattern: **Singleton Pattern** e **Strategy Pattern**. Di seguito vengono riportate le motivazioni che hanno portato all'adozione dei suddetti pattern nel contesto dell'applicazione.

#### Singleton Pattern

L'applicazione necessita di un controllo centralizzato dell'accesso alla risorsa di connessione al database essendo una risorsa condivisa. L'utilizzo di un Singleton Pattern quindi risulta essere utile, per la limitazione,e l'accesso concorrente alla risorsa DataSource. L'utilizzo del pattern Singleton garantisce quindi l'esistenza di un'unica istanza di DataSource. In questo modo quindi: si previene la creazione simultanea di più connessioni ridondanti,si gestisce con maggior chiarezza il ciclo di vita della connessione condivisa e si semplifica il coordinamento dell'accesso ai dati.

|                                   |                  |
|-----------------------------------|------------------|
| Progetto: CineNow                 | Versione: 1.0    |
| Documento: Object Design Document | Data: 15/12/2024 |

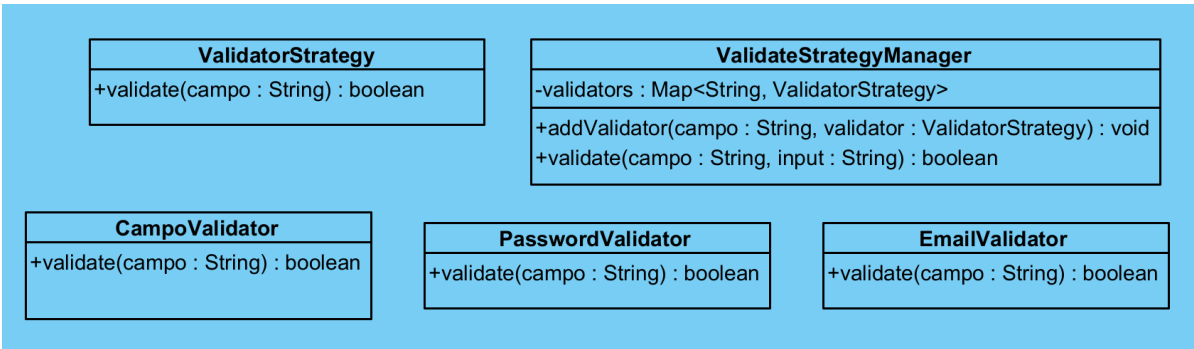
La classe `DatabaseSourceSingleton` è la specializzazione del pattern Singleton per l'applicazione CineNow. presenta un attributo statico, di tipo `DataSource`: `instance`. L'accesso al `DataSource` avviene quindi ottenendo l'attributo `instance` tramite il metodo `getConnection()`. La connessione verrà poi inserita nel contesto dell'applicazione web tramite l'uso della classe `MainContext`.



### Strategy Pattern

I requisiti di validazione dei campi input, come ad esempio email, password o altri tipi di dati, possono variare spesso e soprattutto richiedere frequenti modifiche o ampliamenti. Si è optato quindi per l'introduzione dello Strategy Pattern. Esso consente di definire un'interfaccia comune e di implementare differenti strategie di validazione, facilmente intercambiabili. Grazie a questo approccio quindi è possibile gestire con maggiore flessibilità e modularità l'introduzione di nuove logiche di validazione senza alterare l'architettura esistente, e riutilizzando quella necessaria per più istanze di classi che ne richiedono una, semplificando inoltre la manutenibilità e il testing. Per la nostra applicazione, sono stati definiti quindi:

- `ValidatorStrategy` : interfaccia che espone il metodo per la validazione del campo
- `ValidationManager`: Classe che gestisce ogni `Validator` che implementa l'interfaccia e che può essere utilizzato.
- Classi per l'implementazione dei `Validator` su tipi di dati differenti.



|                                   |                  |
|-----------------------------------|------------------|
| Progetto: CineNow                 | Versione: 1.0    |
| Documento: Object Design Document | Data: 15/12/2024 |

## 1.4. *Definizioni, acronimi e abbreviazioni*

**FR:** Requisito Funzionale.

**NFR:** Requisito Non Funzionale.

**UTC:** Utente Cliente.

**UGS:** Utente Gestore Sede.

**UGC:** Utente Gestore Catena.

## 1.5. *Riferimenti*

*Libro:* Object-Oriented Software Engineering – Using UML, Patterns and Java.

*Autori:* Bernd Bruegge & Allen H. Dutoit.

*Documenti:*

- [RAD\\_CineNow](#) (Requirements Analysis Document): Sono descritte le funzionalità individuate in fase di analisi.
- [SDD\\_CineNow](#) (System Design Document): Contiene una descrizione completa dell'architettura del sistema, inclusi i sottosistemi individuati ed i servizi forniti da ciascun sottosistema.
- ClassDiagram\_CineNow: Mostra il class diagram, per il sistema CineNow.



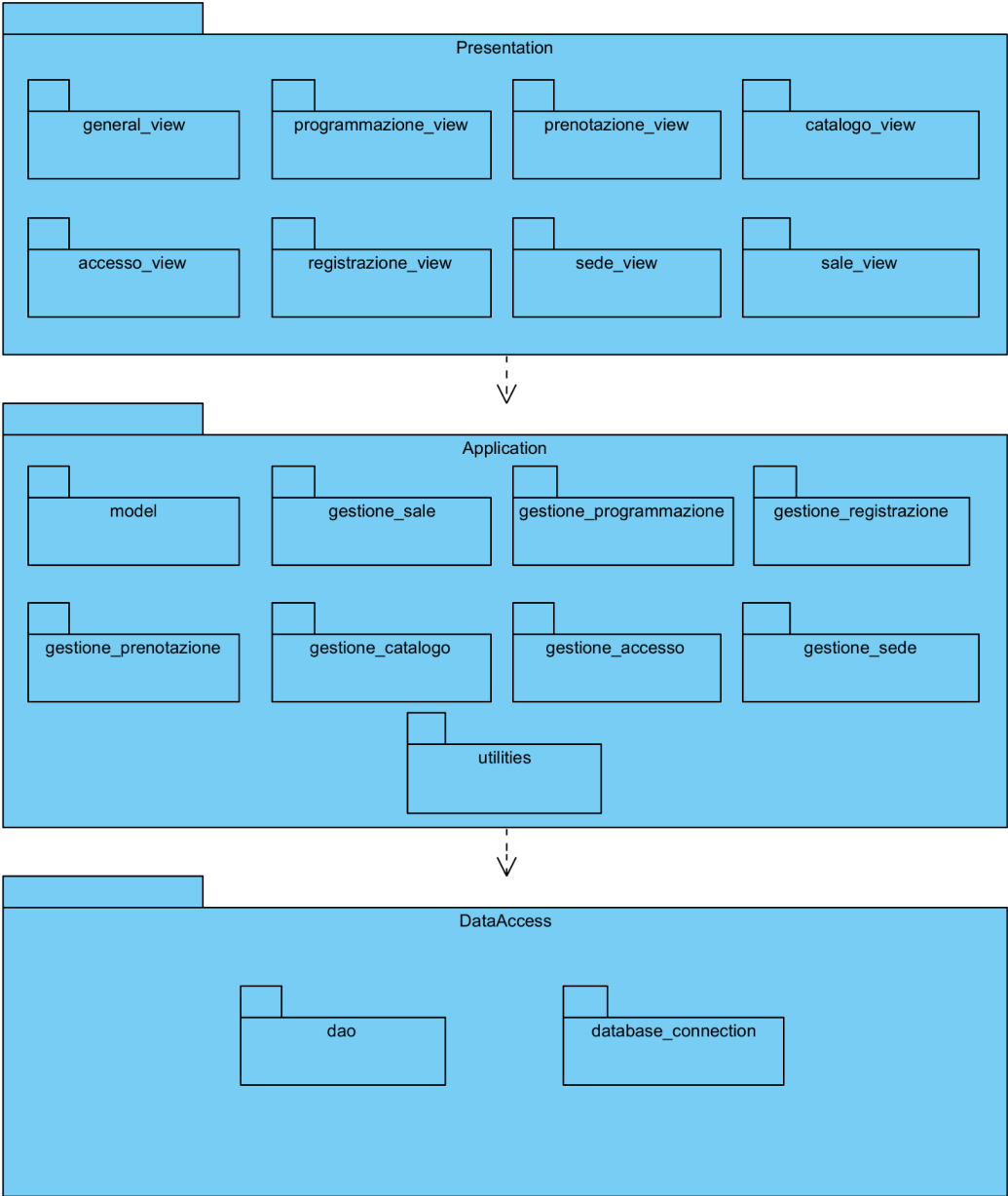
|                                   |                  |
|-----------------------------------|------------------|
| Progetto: CineNow                 | Versione: 1.0    |
| Documento: Object Design Document | Data: 15/12/2024 |

# 2. PACKAGES

## 2.1 Panoramica

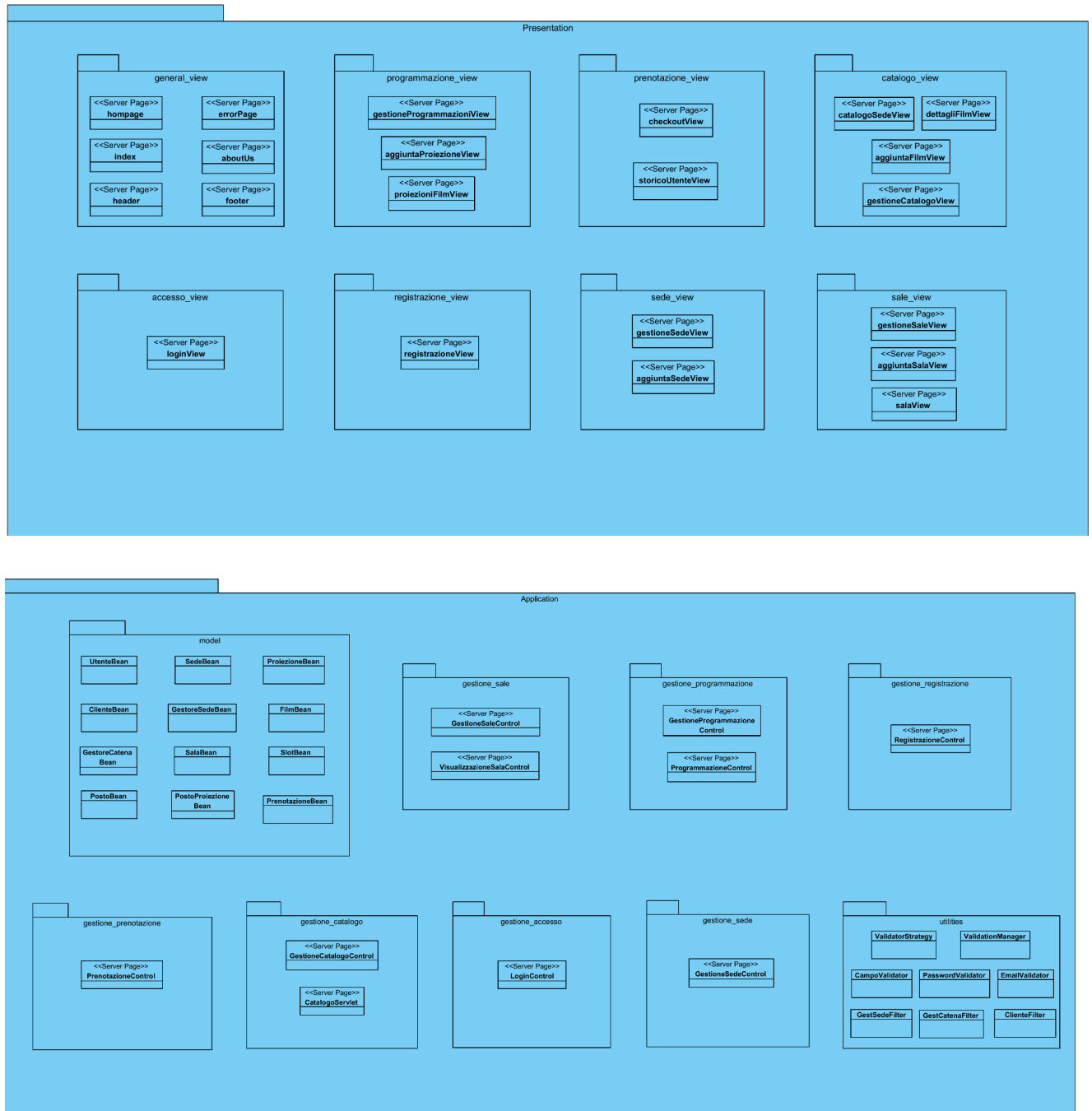
In questa sezione, verrà prima mostrata una panoramica generale del packaging completo del sistema CineNow. Dopodiché nella sezione 2.3 si entrerà più nel dettaglio, dei package dei layer Application, DataAccess e Presentation. Nella sezione 2.4 si analizzano i singoli package e le interfacce delle classi in esse contenute.

## 2.2 Packaging del sistema

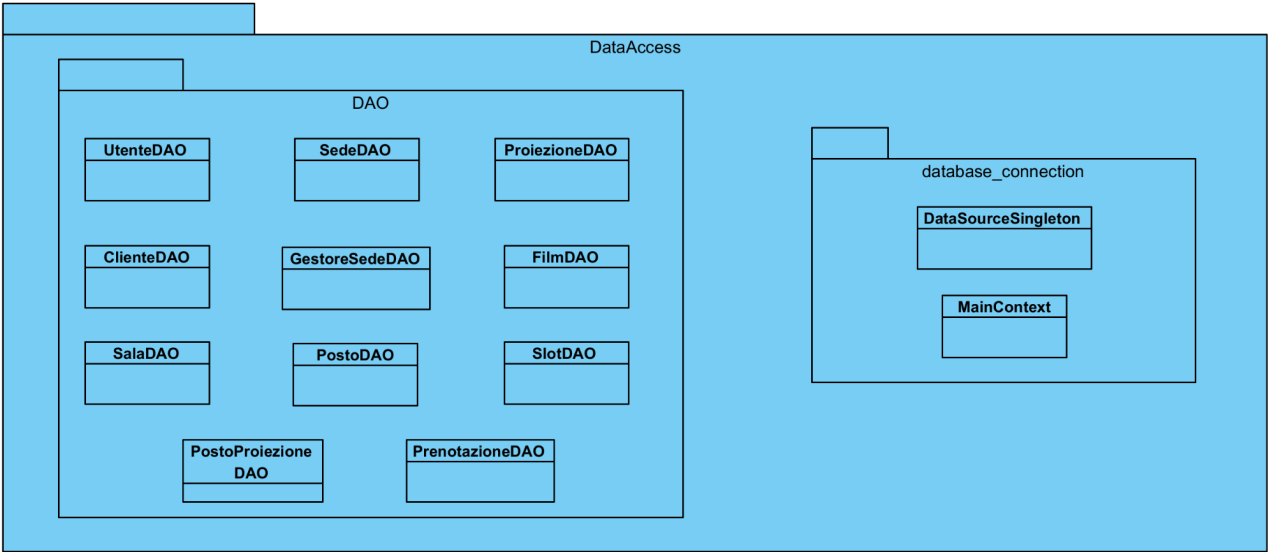


|                                   |                  |
|-----------------------------------|------------------|
| Progetto: CineNow                 | Versione: 1.0    |
| Documento: Object Design Document | Data: 15/12/2024 |

## 2.3 Packaging dei layer



|                                   |                  |
|-----------------------------------|------------------|
| Progetto: CineNow                 | Versione: 1.0    |
| Documento: Object Design Document | Data: 15/12/2024 |

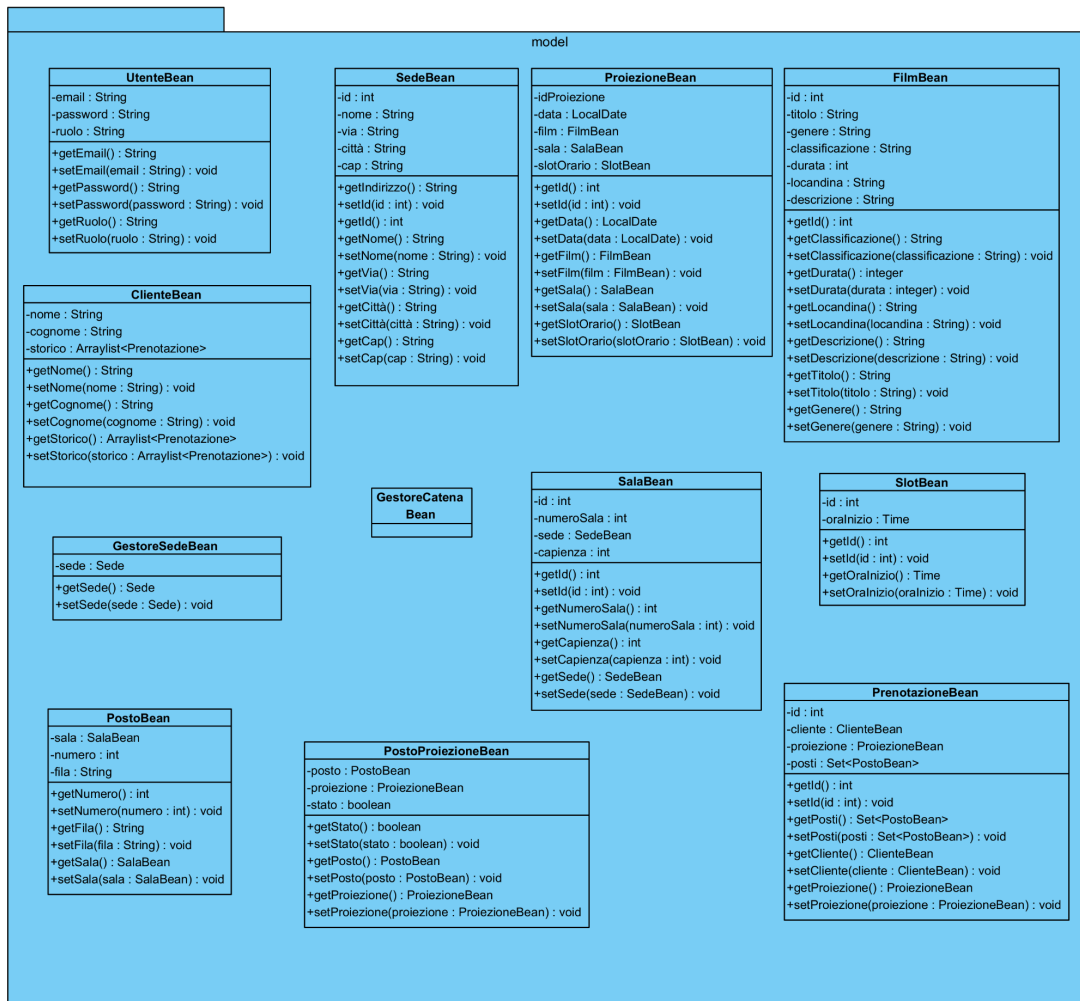


|                                   |                  |
|-----------------------------------|------------------|
| Progetto: CineNow                 | Versione: 1.0    |
| Documento: Object Design Document | Data: 15/12/2024 |

## 2.4 Dettaglio dei singoli package

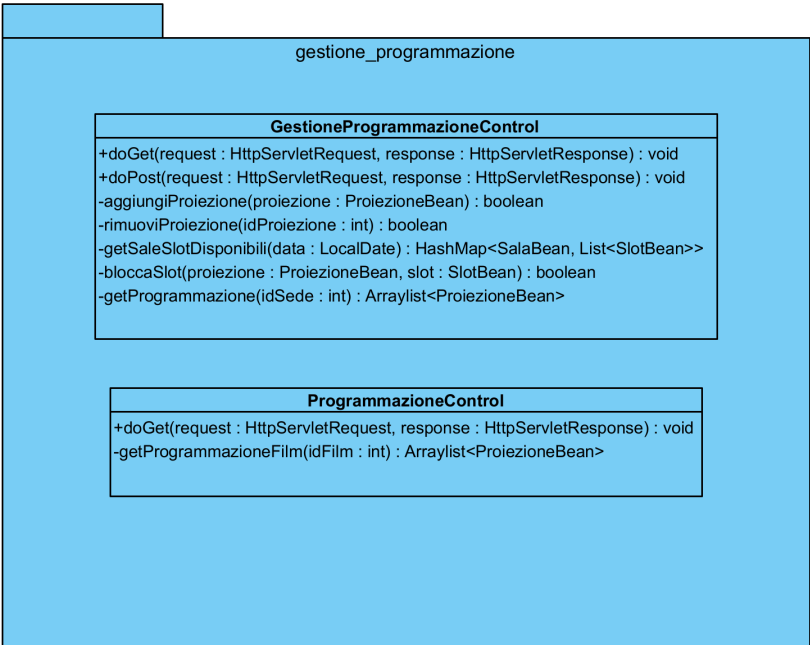
### 2.4.1 Package Application

#### Package Application: model



|                                   |                  |
|-----------------------------------|------------------|
| Progetto: CineNow                 | Versione: 1.0    |
| Documento: Object Design Document | Data: 15/12/2024 |

**Package Application: gestione\_programmazione**

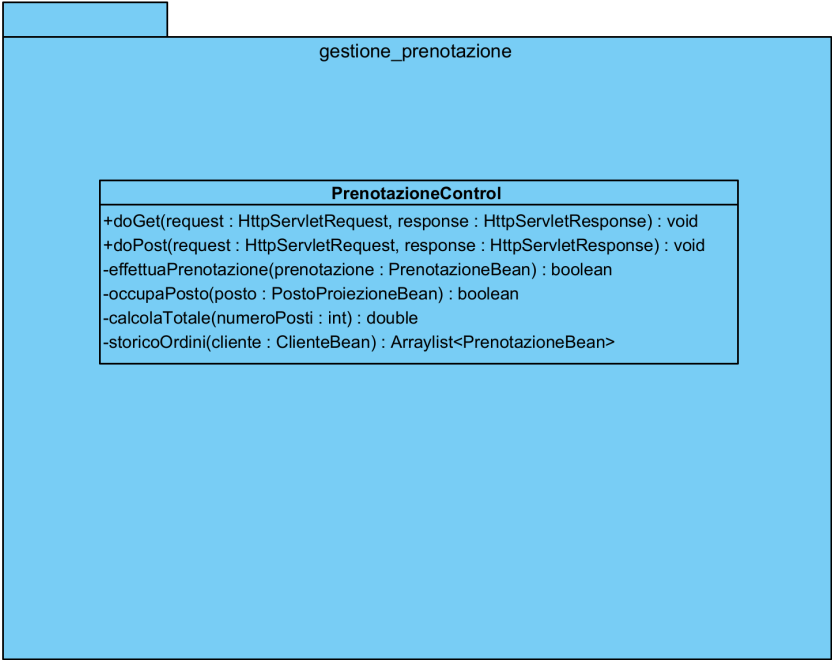


**Package Application: gestione\_registrazione**



|                                   |                  |
|-----------------------------------|------------------|
| Progetto: CineNow                 | Versione: 1.0    |
| Documento: Object Design Document | Data: 15/12/2024 |

**Package Application: gestione\_prenotazione**

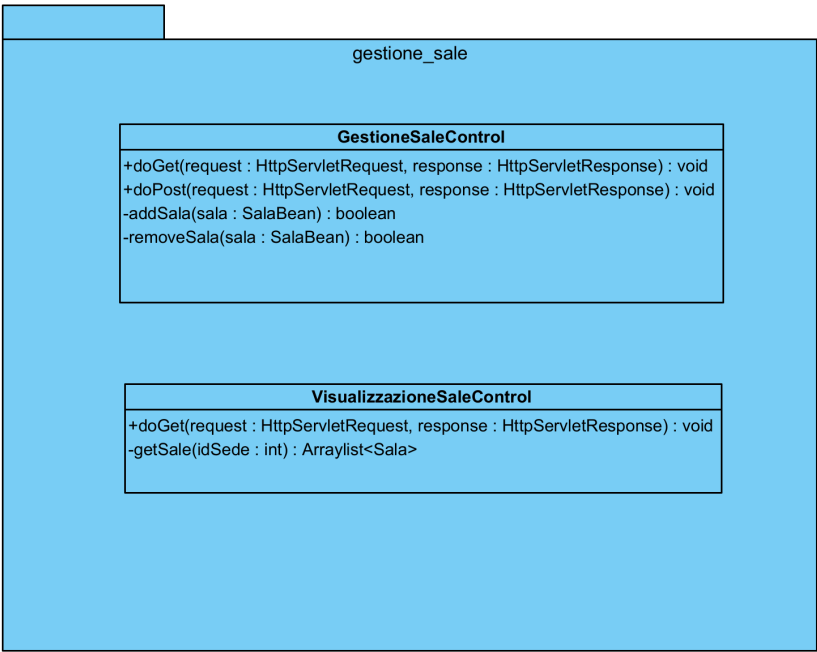


**Package Application: gestione\_catalogo**

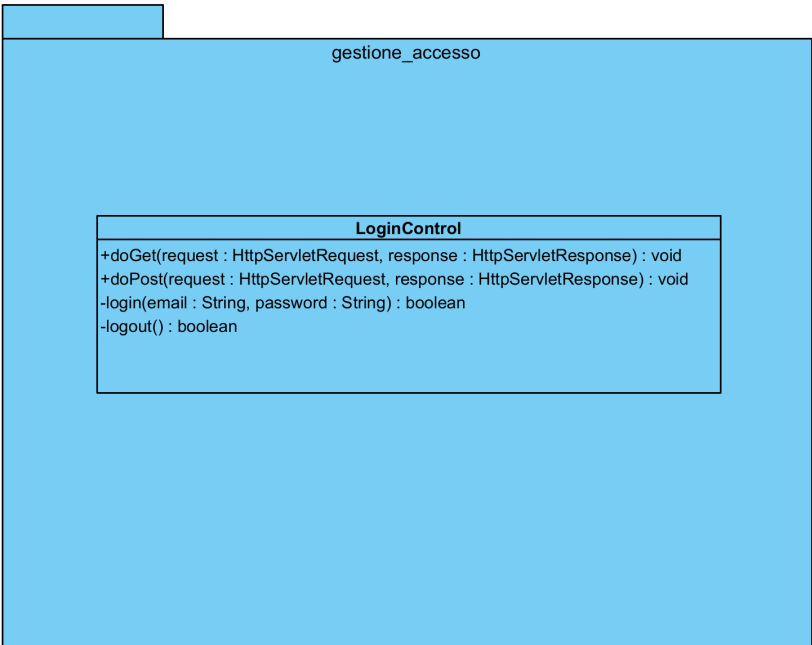


|                                   |                  |
|-----------------------------------|------------------|
| Progetto: CineNow                 | Versione: 1.0    |
| Documento: Object Design Document | Data: 15/12/2024 |

**Package Application: gestione\_sale**



**Package Application: gestione\_accesso**

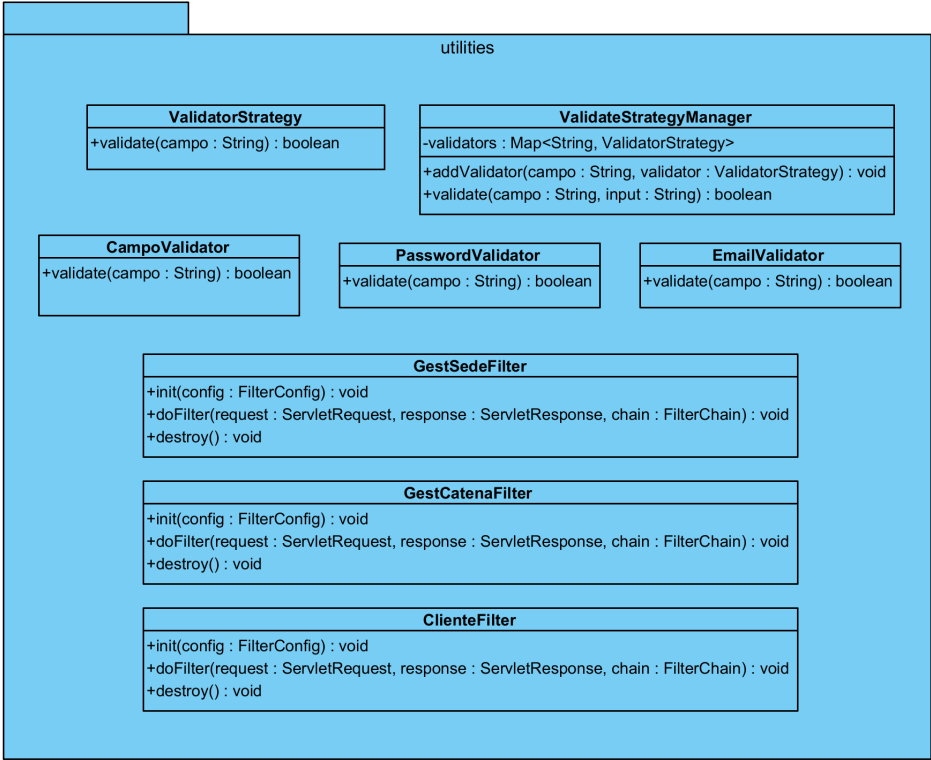


|                                   |                  |
|-----------------------------------|------------------|
| Progetto: CineNow                 | Versione: 1.0    |
| Documento: Object Design Document | Data: 15/12/2024 |

**Package Application: gestione\_sede**



**Package Application: utilities**

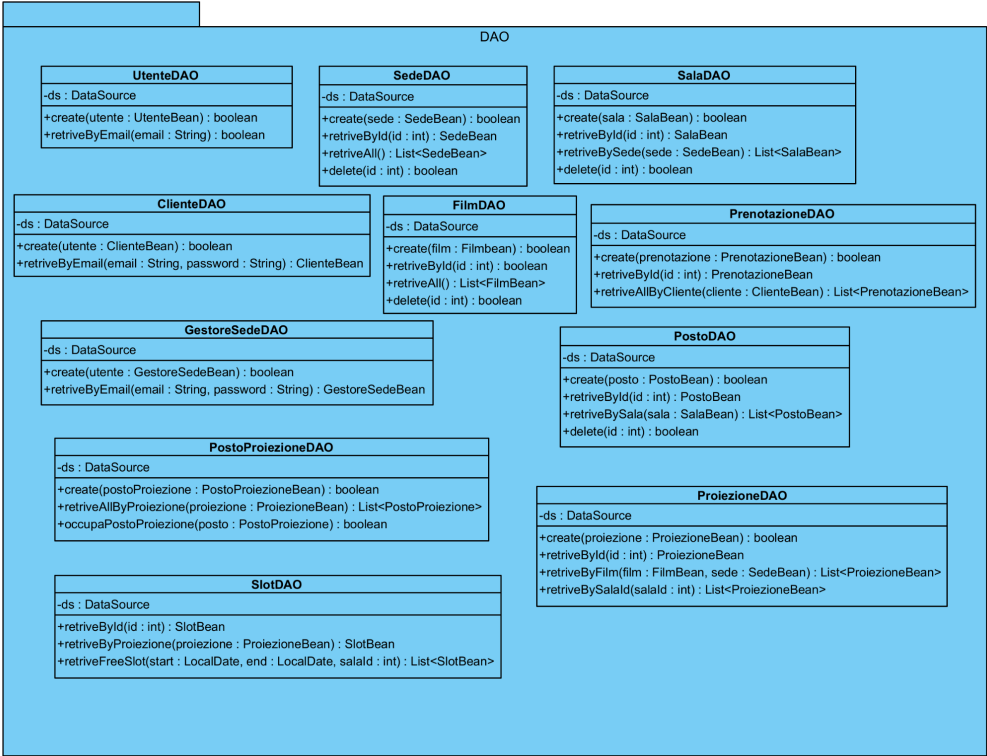




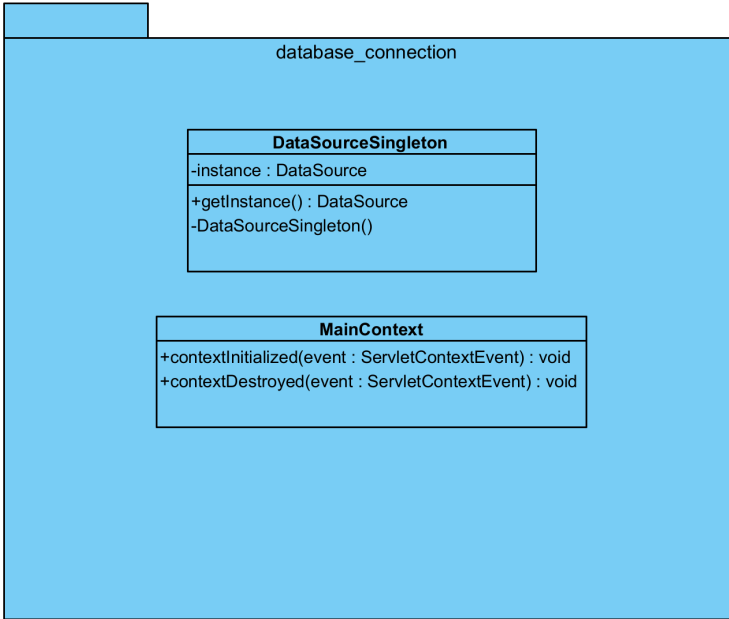
|                                   |                  |
|-----------------------------------|------------------|
| Progetto: CineNow                 | Versione: 1.0    |
| Documento: Object Design Document | Data: 15/12/2024 |

2.4.2 Package DataAccess

Package DataAccess: DAO



Package DataAccess: database\_connection



|                                   |                  |
|-----------------------------------|------------------|
| Progetto: CineNow                 | Versione: 1.0    |
| Documento: Object Design Document | Data: 15/12/2024 |

# 3. CLASS INTERFACES

## 3.1 Panoramica

Di seguito, in dettaglio, vengono mostrate le classi e i loro metodi, nonché i contratti definiti tramite OCL. Per il class diagram completo, visitare: `ClassDiagram_CineNow`.

## 3.2 DataAccess

**Package: DAO**

| NOME CLASSE | UtenteDAO                                                                               |
|-------------|-----------------------------------------------------------------------------------------|
| METODI      | +create(utente : UtenteBean) : boolean<br>+retrieveByEmail(email : String) : UtenteBean |
| INVARIANTI  |                                                                                         |

| NOME METODO     | +create(utente : UtenteBean) : boolean                                                                                                                                                                                       |
|-----------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| DESCRIZIONE     | Questo metodo consente di inserire un nuovo utente                                                                                                                                                                           |
| PRE-CONDIZIONE  | <b>context:</b> UtenteDAO::create(utente: UtenteBean)<br><br><b>pre:</b><br>utente!= NULL AND<br>utente.email != NULL AND<br>utente.password!=null AND<br>Database.utente->!exists(Utente   Utente.email ==<br>utente.email) |
| POST-CONDIZIONE | <b>context:</b> UtenteDAO::create(utente: UtenteBean)<br><br><b>post:</b> Database.Utente->include(utente)                                                                                                                   |

|                                   |                  |
|-----------------------------------|------------------|
| Progetto: CineNow                 | Versione: 1.0    |
| Documento: Object Design Document | Data: 15/12/2024 |

|                 |                                                                                                                                                           |
|-----------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------|
| NOME METODO     | +retriveByEmail(email : String) : UtenteBean                                                                                                              |
| DESCRIZIONE     | Questo metodo permette di ottenere l'utente associato alla email                                                                                          |
| PRE-CONDIZIONE  | <b>context:</b> UtenteDAO::retriveByEmail(email: String): UtenteBean<br><br><b>pre:</b> email != NULL                                                     |
| POST-CONDIZIONE | <b>context:</b> UtenteDAO::retriveByEmail(email: String): UtenteBean<br><br><b>post:</b> result = Database.Utente->select(Utente   Utente.email == email) |

|             |                                                                                                                                         |
|-------------|-----------------------------------------------------------------------------------------------------------------------------------------|
| NOME CLASSE | SedeDAO                                                                                                                                 |
| METODI      | +create(sede: SedeBean) : boolean<br>+retriveById(id : int) : SedeBean<br>+retriveAll() : List<SedeBean><br>+delete(id : int) : boolean |
| INVARIANTI  |                                                                                                                                         |

|                |                                                                                                                                                                       |
|----------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| NOME METODO    | +create(sede: SedeBean) : boolean                                                                                                                                     |
| DESCRIZIONE    | Questo metodo permette di aggiungere una nuova sede                                                                                                                   |
| PRE-CONDIZIONE | <b>context:</b> SedeDAO::create(sede: SedeBean): boolean<br><br><b>pre:</b><br>sede!=NULL AND<br>sede.nome != NULL AND<br>sede.nome != "" AND<br>sede.via != NULL AND |

|                                   |                  |
|-----------------------------------|------------------|
| Progetto: CineNow                 | Versione: 1.0    |
| Documento: Object Design Document | Data: 15/12/2024 |

|                 |                                                                                                                        |
|-----------------|------------------------------------------------------------------------------------------------------------------------|
|                 | sede.via != "" AND<br>sede.cap != NULL AND<br>sede.cap.length() == 5 AND<br>sede.città != NULL AND<br>sede.città != "" |
| POST-CONDIZIONE | <b>context:</b> SedeDAO::create(sede: SedeBean):<br>boolean<br><br><b>post:</b> Database.Sede->include(sede)           |

|                 |                                                                                                                                                     |
|-----------------|-----------------------------------------------------------------------------------------------------------------------------------------------------|
| NOME METODO     | +retriveById(id : int) : SedeBean                                                                                                                   |
| DESCRIZIONE     | Questo metodo permette di ottenere una sede in base all'id                                                                                          |
| PRE-CONDIZIONE  | <b>context:</b> SedeDAO::retriveById(id : int) : SedeBean<br><br><b>pre:</b> id != NULL AND id>0                                                    |
| POST-CONDIZIONE | <b>context:</b> SedeDAO::retriveById(id : int) : SedeBean<br><br><b>post:</b> result = Database.Sede->select(Sede   Sede.id == id) AND result!=null |

|                 |                                                                                                    |
|-----------------|----------------------------------------------------------------------------------------------------|
| NOME METODO     | +retriveAll() : List<SedeBean>                                                                     |
| DESCRIZIONE     | Questo metodo permette di ottenere tutte le sedi presenti nel database                             |
| PRE-CONDIZIONE  | <b>context:</b> SedeBean::retriveAll() : List<SedeBean><br><br><b>pre:</b>                         |
| POST-CONDIZIONE | <b>context:</b> SedeBean::retriveAll() : List<SedeBean><br><br><b>post:</b> result = Database.Sede |

|                                   |                  |
|-----------------------------------|------------------|
| Progetto: CineNow                 | Versione: 1.0    |
| Documento: Object Design Document | Data: 15/12/2024 |

|                 |                                                                                                                  |
|-----------------|------------------------------------------------------------------------------------------------------------------|
| NOME METODO     | +delete(id : int) : boolean                                                                                      |
| DESCRIZIONE     | Questa operazione permette al Gestore Catena di eliminare una sede                                               |
| PRE-CONDIZIONE  | <b>context:</b> SedeDAO::delete(id : int) : boolean<br><b>pre:</b> Database.Sede->exists(Sede   Sede.id == id)   |
| POST-CONDIZIONE | <b>context:</b> SedeDAO::delete(id : int) : boolean<br><b>post:</b> Database.Sede->!exists(Sede   Sede.id == id) |

|             |                                                                                                                                                     |
|-------------|-----------------------------------------------------------------------------------------------------------------------------------------------------|
| NOME CLASSE | <b>SalaDAO</b>                                                                                                                                      |
| METODI      | +create(sala: SalaBean) : boolean<br>+retriveById(id: int) : SalaBean<br>+retiveBySede(SedeId: int) : List<SalaBean><br>+delete(id : int) : boolean |
| INVARIANTI  |                                                                                                                                                     |

|                 |                                                                                                                                                                                   |
|-----------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| NOME METODO     | +create(sala: SalaBean) : boolean                                                                                                                                                 |
| DESCRIZIONE     | Questo metodo permette l'aggiunta di una nuova sala dal Gestore Sede                                                                                                              |
| PRE-CONDIZIONE  | <b>context:</b> SalaDAO::create(sala: SalaBean) : boolean<br><br><b>pre:</b><br>sala.numero != NULL AND<br>sala.capienza != NULL AND<br>sala.capienza > 0<br>sala.id_sede != NULL |
| POST-CONDIZIONE | <b>context:</b> SalaDAO::create(sala: SalaBean) : boolean<br><br><b>post:</b> Database.Sala->include(sala)                                                                        |

|                                   |                  |
|-----------------------------------|------------------|
| Progetto: CineNow                 | Versione: 1.0    |
| Documento: Object Design Document | Data: 15/12/2024 |

|                 |                                                                                                                               |
|-----------------|-------------------------------------------------------------------------------------------------------------------------------|
| NOME METODO     | +retriveById(id: int) : SalaBean                                                                                              |
| DESCRIZIONE     | Questo metodo permette di ottenere una sala tramite il suo id                                                                 |
| PRE-CONDIZIONE  | <b>context:</b> SalaDAO::retriveById(id: int) : SalaBean<br><b>pre:</b> id != NULL AND id>0                                   |
| POST-CONDIZIONE | <b>context:</b> SalaDAO::retriveById(id: int) : SalaBean<br><b>post:</b> result = Database.Sala->select(Sala   Sala.id == id) |

|                 |                                                                                                                                         |
|-----------------|-----------------------------------------------------------------------------------------------------------------------------------------|
| NOME METODO     | +retiveBySede(SedeId: int) : List<SalaBean>                                                                                             |
| DESCRIZIONE     | Questo metodo permette di ottenere tutte le sale di un sede                                                                             |
| PRE-CONDIZIONE  | <b>context:</b> retiveBySede(SedeId: int) : List<SalaBean><br><b>pre:</b> SedeId != NULL AND SedeId>0                                   |
| POST-CONDIZIONE | <b>context:</b> retiveBySede(SedeId: int) : List<SalaBean><br><b>post:</b> result = Database.Sala->select(Sala   sala.id_sede = SedeId) |

|                 |                                                                                                       |
|-----------------|-------------------------------------------------------------------------------------------------------|
| NOME METODO     | +delete(id : int) : boolean                                                                           |
| DESCRIZIONE     | Questo metodo permette al Gestore Sede di eliminare una sala                                          |
| PRE-CONDIZIONE  | <b>context:</b> delete(id : int) : boolean<br><b>pre:</b> Database.Sala->exists(Sala   Sala.id == id) |
| POST-CONDIZIONE | <b>context:</b> delete(id : int) : boolean                                                            |

|                                   |                  |
|-----------------------------------|------------------|
| Progetto: CineNow                 | Versione: 1.0    |
| Documento: Object Design Document | Data: 15/12/2024 |

|  |                                                           |
|--|-----------------------------------------------------------|
|  | <b>post:</b> Database.Sala->!exists(Sala   Sala.id == id) |
|--|-----------------------------------------------------------|

|                    |                                                                                       |
|--------------------|---------------------------------------------------------------------------------------|
| <b>NOME CLASSE</b> | <b>ClienteDAO</b>                                                                     |
| <b>METODI</b>      | +create(utente: ClienteBean) : boolean<br>+retriveByEmail(email: String): ClienteBean |
| <b>INVARIANTI</b>  |                                                                                       |

|                        |                                                                                                                                                                                                                                                                                                                 |
|------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>NOME METODO</b>     | +create(utente: ClienteBean) : boolean                                                                                                                                                                                                                                                                          |
| <b>DESCRIZIONE</b>     | Questo metodo consente di inserire un nuovo cliente                                                                                                                                                                                                                                                             |
| <b>PRE-CONDIZIONE</b>  | <b>context:</b> ClienteDAO::create(utente: ClienteBean) : boolean<br><br><b>pre:</b><br>cliente!= NULL AND<br>cliente.email != NULL AND<br>cliente.password!=null AND<br>cliente.email.validate() AND<br>cliente.password.validate() AND<br>Database.Cliente->!exists(Cliente   Cliente.email == cliente.email) |
| <b>POST-CONDIZIONE</b> | <b>context:</b> ClienteDAO::create(utente: ClienteBean) : boolean<br><br><b>post:</b> Database.Cliente->exists(Cliente)                                                                                                                                                                                         |

|                       |                                                                         |
|-----------------------|-------------------------------------------------------------------------|
| <b>NOME METODO</b>    | +retriveByEmail(email: String): ClienteBean                             |
| <b>DESCRIZIONE</b>    | Questo metodo permette di ottenere il cliente associato all'email       |
| <b>PRE-CONDIZIONE</b> | <b>context:</b> ClienteDAO:: retriveByEmail(email: String): ClienteBean |

|                                   |                  |
|-----------------------------------|------------------|
| Progetto: CineNow                 | Versione: 1.0    |
| Documento: Object Design Document | Data: 15/12/2024 |

|                 |                                                                                                                                                                |
|-----------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------|
|                 | <b>pre:</b> email != NULL                                                                                                                                      |
| POST-CONDIZIONE | <b>context:</b> ClienteDAO::retriveByEmail(email: String): ClienteBeanb<br><br><b>post:</b> result = Database.Cliente->select(Cliente   Cliente.email = email) |

| NOME CLASSE | FilmDAO                                                                                                                             |
|-------------|-------------------------------------------------------------------------------------------------------------------------------------|
| METODI      | +create(film: FilmBean) : boolean<br>+retriveById(id: int) : boolean<br>+retriveAll(): List<FilmBean><br>+delete(id: int) : boolean |
| INVARIANTI  |                                                                                                                                     |

|                |                                                                                                                                                                                                                                                                                                                                                                                                                                                            |
|----------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| NOME METODO    | +create(film: FilmBean) : boolean                                                                                                                                                                                                                                                                                                                                                                                                                          |
| DESCRIZIONE    | Questo metodo permette all'utente Gestore Catena di aggiungere un film al catalogo                                                                                                                                                                                                                                                                                                                                                                         |
| PRE-CONDIZIONE | <b>context:</b> FilmDAO::create(film: FilmBean) : boolean<br><br><b>pre:</b><br>film != NULL AND<br>film.titolo != NULL AND<br>film.titolo != "" AND<br>film.genere != NULL AND<br>film.genere != "" AND<br>film.classificazione != NULL AND<br>film.classificazione != "" AND<br>film.durata != NULL AND<br>film.durata > 52 (minuti) AND<br>film.locandina != NULL AND<br>film.locandina != "" AND<br>film.descrizione != NULL<br>film.descrizione != "" |



|                                   |                  |
|-----------------------------------|------------------|
| Progetto: CineNow                 | Versione: 1.0    |
| Documento: Object Design Document | Data: 15/12/2024 |

|                 |                                                                                                            |
|-----------------|------------------------------------------------------------------------------------------------------------|
| POST-CONDIZIONE | <b>context:</b> FilmDAO::create(film: FilmBean) : boolean<br><br><b>post:</b> Database.Film->include(film) |
|-----------------|------------------------------------------------------------------------------------------------------------|

|                 |                                                                                                                                  |
|-----------------|----------------------------------------------------------------------------------------------------------------------------------|
| NOME METODO     | +retrieveById(id: int) : boolean                                                                                                 |
| DESCRIZIONE     | Questo metodo permette di ottenere il film in base all'id                                                                        |
| PRE-CONDIZIONE  | <b>context:</b> FilmDAO::retrieveById(id: int) : boolean<br><br><b>pre:</b> id!= NULL AND id > 0                                 |
| POST-CONDIZIONE | <b>context:</b> FilmDAO::retrieveById(id: int) : boolean<br><br><b>post:</b> result = Database.Film->select(Film   Film.id = id) |

|                 |                                                                                                   |
|-----------------|---------------------------------------------------------------------------------------------------|
| NOME METODO     | +retrieveAll(): List<FilmBean>                                                                    |
| DESCRIZIONE     | Questo metodo permette di ottenere tutta la lista di Film presenti nel Database                   |
| PRE-CONDIZIONE  | <b>context:</b> FilmDAO::retrieveAll(): List<FilmBean><br><br><b>pre:</b>                         |
| POST-CONDIZIONE | <b>context:</b> FilmDAO::retrieveAll(): List<FilmBean><br><br><b>post:</b> result = Database.Film |

|                |                                                                            |
|----------------|----------------------------------------------------------------------------|
| NOME METODO    | +delete(id: int) : boolean                                                 |
| DESCRIZIONE    | Questo metodo permette al Gestore Catena di eliminare un film dal catalogo |
| PRE-CONDIZIONE | <b>context:</b> FilmDAO::delete(id: int) : boolean                         |

|                                   |                  |
|-----------------------------------|------------------|
| Progetto: CineNow                 | Versione: 1.0    |
| Documento: Object Design Document | Data: 15/12/2024 |

|                 |                                                                                                              |
|-----------------|--------------------------------------------------------------------------------------------------------------|
|                 | <b>pre:</b> Database.Film->exists(Film  Film.id = id )                                                       |
| POST-CONDIZIONE | <b>context:</b> FilmDAO::delete(id: int) : boolean<br><b>post:</b> Database.Film->!exists(Film Film.id = id) |

| NOME CLASSE | PrenotazioneDAO                                                                                                                                              |
|-------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------|
| METODI      | +create(prenotazione: PrenotazioneBean) : boolean<br>+retriveById(id: int): prenotazioneBean<br>+retriveAllByCliente(clienteId: int): List<PrenotazioneBean> |
| INVARIANTI  |                                                                                                                                                              |

|                 |                                                                                                                                                                                                                     |
|-----------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| NOME METODO     | +create(prenotazione: PrenotazioneBean) : boolean                                                                                                                                                                   |
| DESCRIZIONE     | Questo metodo permette di creare una nuova prenotazione                                                                                                                                                             |
| PRE-CONDIZIONE  | <b>context:</b><br>PrenotazioneDAO::create(prenotazione: PrenotazioneBean) : boolean<br><br><b>pre:</b><br>prenotazione != NULL AND<br>prenotazione.email_cliente != NULL AND<br>prenotazione.id_proiezione != NULL |
| POST-CONDIZIONE | <b>context:</b><br>PrenotazioneDAO::create(prenotazione: PrenotazioneBean) : boolean<br><br><b>post:</b><br>Database.Prenotazione->include(prenotazione)                                                            |

|             |                                         |
|-------------|-----------------------------------------|
| NOME METODO | +retriveById(id: int): prenotazioneBean |
|-------------|-----------------------------------------|

|                                   |                  |
|-----------------------------------|------------------|
| Progetto: CineNow                 | Versione: 1.0    |
| Documento: Object Design Document | Data: 15/12/2024 |

|                 |                                                                                                                                                               |
|-----------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------|
| DESCRIZIONE     | Questo metodo permette di ottenere una prenotazione in base all'id                                                                                            |
| PRE-CONDIZIONE  | <b>context:</b> PrenotazioneDAO::retriveById(id: int): prenotazioneBean<br><br><b>pre:</b> id != NULL AND id > 0                                              |
| POST-CONDIZIONE | <b>context:</b> PrenotazioneDAO::retriveById(id: int): prenotazioneBean<br><br><b>post:</b> Database.Prenotazione->select(Prenotazione Prenotazione.id = id ) |

|                 |                                                                                                                                                                                                                |
|-----------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| NOME METODO     | +retriveAllByCliente(clienteId: int): List<PrenotazioneBean>                                                                                                                                                   |
| DESCRIZIONE     | Questo metodo permette di ottenere tutte le prenotazioni effettuate dal cliente                                                                                                                                |
| PRE-CONDIZIONE  | <b>context:</b> PrenotazioneDAO::retriveAllByCliente(clienteId: int): List<PrenotazioneBean><br><br><b>pre:</b> DA CAMBIARE                                                                                    |
| POST-CONDIZIONE | <b>context:</b> PrenotazioneDAO::retriveAllByCliente(email_cliente: int): List<PrenotazioneBean><br><br><b>post:</b> result = Database.Prenotazione(Prenotazione   Prenotazione.email_cliente = email_cliente) |

|             |                                                                                                |
|-------------|------------------------------------------------------------------------------------------------|
| NOME CLASSE | <b>GestoreSedeDAO</b>                                                                          |
| METODI      | +create(utente: GestoreSedeBean) : boolean<br>+retriveByEmail(email: String): GestioneSedeBean |
| INVARIANTI  |                                                                                                |

|                                   |                  |
|-----------------------------------|------------------|
| Progetto: CineNow                 | Versione: 1.0    |
| Documento: Object Design Document | Data: 15/12/2024 |

|                 |                                                                                                                                                                                                         |
|-----------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| NOME METODO     | +create(utente: GestoreSedeBean) : boolean                                                                                                                                                              |
| DESCRIZIONE     | Questo metodo permette di inserire un nuovo Gestore Sede                                                                                                                                                |
| PRE-CONDIZIONE  | <b>context:</b> GestoreSedeDAO::create(utente: GestoreSedeBean) : boolean<br><br><b>pre:</b><br>utente != NULL AND<br>utente.email != NULL AND<br>utente.email.validate() AND<br>utente.id_sede != NULL |
| POST-CONDIZIONE | <b>context:</b> GestoreSedeDAO::create(utente: GestoreSedeBean) : boolean<br><br><b>post:</b> Database.Gest_Sede->include(utente)                                                                       |

|                 |                                                                                                                                                                                    |
|-----------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| NOME METODO     | +retriveByEmail(email: String): GestioneSedeBean                                                                                                                                   |
| DESCRIZIONE     | Questo metodo permette di ottenere il Gestore Sede associato alla email                                                                                                            |
| PRE-CONDIZIONE  | <b>context:</b><br>GestoreSedeDAO::retriveByEmail(email: String): GestioneSedeBean<br><br><b>pre:</b> email != NULL                                                                |
| POST-CONDIZIONE | <b>context:</b><br>GestoreSedeDAO::retriveByEmail(email: String): GestioneSedeBean<br><br><b>post:</b> result =<br>Database.Gest_Sede->select(Gest_Sede   Gest_Sede.email = email) |

|             |               |
|-------------|---------------|
| NOME CLASSE | ProiezioneDAO |
|-------------|---------------|

|                                   |                  |
|-----------------------------------|------------------|
| Progetto: CineNow                 | Versione: 1.0    |
| Documento: Object Design Document | Data: 15/12/2024 |

|            |                                                                                                                                                                                                         |
|------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| METODI     | +create(proiezione: ProiezioneBean) : boolean<br>+retriveById(id: int) : ProiezioneBean<br>+retriveByFilmId(filmId: int) : List<ProiezioneBean><br>+retriveBySalaId(salaId: int) : List<ProiezioneBean> |
| INVARIANTI |                                                                                                                                                                                                         |

|                 |                                                                                                                                                                                                                                                             |
|-----------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| NOME METODO     | +create(proiezione: ProiezioneBean) : boolean                                                                                                                                                                                                               |
| DESCRIZIONE     | Questo metodo permette al Gestore Sede di aggiungere una nuova proiezione                                                                                                                                                                                   |
| PRE-CONDIZIONE  | <b>context:</b> ProiezioneDAO::create(proiezione: ProiezioneBean) : boolean<br><br><b>pre:</b><br>proiezione != NULL AND<br>proiezione.data != NULL AND<br>proiezione.id_film != NULL AND<br>proiezione.id_sala != NULL AND<br>proiezione.id_orario != NULL |
| POST-CONDIZIONE | <b>context:</b> create(proiezione: ProiezioneBean) : boolean<br><br><b>post:</b><br>Database.Proiezione->include(proiezione)                                                                                                                                |

|                 |                                                                                                               |
|-----------------|---------------------------------------------------------------------------------------------------------------|
| NOME METODO     | +retriveById(id: int) : ProiezioneBean                                                                        |
| DESCRIZIONE     | Questo metodo permette di ottenere la proiezione in base all'id                                               |
| PRE-CONDIZIONE  | <b>context:</b> ProiezioneDAO::retriveById(id: int) : ProiezioneBean<br><br><b>pre:</b> id != NULL AND id > 0 |
| POST-CONDIZIONE | <b>context:</b> ProiezioneDAO::retriveById(id: int) : ProiezioneBean                                          |

|                                   |                  |
|-----------------------------------|------------------|
| Progetto: CineNow                 | Versione: 1.0    |
| Documento: Object Design Document | Data: 15/12/2024 |

|  |                                                                                          |
|--|------------------------------------------------------------------------------------------|
|  | <b>post:</b> result =<br>Database.Proiezione->select(Proiezione  <br>Proiezione.id = id) |
|--|------------------------------------------------------------------------------------------|

|                 |                                                                                                                                                                                        |
|-----------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| NOME METODO     | +retriveByFilmId(filmId: int) :<br>List<ProiezioneBean>                                                                                                                                |
| DESCRIZIONE     | Questo metodo permette di ottenere una lista di proiezioni del film                                                                                                                    |
| PRE-CONDIZIONE  | <b>context:</b><br>ProiezioneDAO::retriveByFilmId(filmId: int) :<br>List<ProiezioneBean><br><br><b>pre:</b> filmId != NULL AND filmId > 0                                              |
| POST-CONDIZIONE | <b>context:</b><br>ProiezioneDAO::retriveByFilmId(filmId: int) :<br>List<ProiezioneBean><br><br><b>post:</b> result = Database.Proiezione(Proiezione  <br>Proiezione.id_film = filmId) |

|                 |                                                                                                                                                                                                     |
|-----------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| NOME METODO     | +retriveBySalaId(salaId: int) : List<ProiezioneBean>                                                                                                                                                |
| DESCRIZIONE     | Questo metodo permette di ottenere la lista di proiezione di quella determinata sala                                                                                                                |
| PRE-CONDIZIONE  | <b>context:</b><br>PrenotazioneDAO::retriveBySalaId(salaId: int)<br>: List<ProiezioneBean><br><br><b>pre:</b> salaId != NULL AND salaId > 0                                                         |
| POST-CONDIZIONE | <b>context:</b><br>PrenotazioneDAO::retriveBySalaId(salaId: int)<br>: List<ProiezioneBean><br><br><b>post:</b> result =<br>Database.Proiezione->select(Proiezione  <br>Proiezione.id_sala = salaId) |

|                                   |                  |
|-----------------------------------|------------------|
| Progetto: CineNow                 | Versione: 1.0    |
| Documento: Object Design Document | Data: 15/12/2024 |

| NOME CLASSE | PostoDAO                                                                                                                                      |
|-------------|-----------------------------------------------------------------------------------------------------------------------------------------------|
| METODI      | +create(posto: PostoBean): boolean<br>+retriveById(id: int): SalaBean<br>+retriveBySala(idSala): List<PostoBean><br>+delete(id: int): boolean |
| INVARIANTI  |                                                                                                                                               |

| NOME METODO     | +create(posto: PostoBean): boolean                                                                                                                                                                                                                                                                          |
|-----------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| DESCRIZIONE     | Questo metodo permette di creare un posto all'interno della sala                                                                                                                                                                                                                                            |
| PRE-CONDIZIONE  | <b>context:</b> PostoDAO::create(posto: PostoBean): boolean<br><br><b>pre:</b><br>posto != NULL AND<br>posto.id_sala != NULL AND<br>posto.id_sala > 0 NAD<br>posto.numero != NULL AND<br>posto.fila != NULL AND<br>Database.Posto->!exists(Posto   Posto.numero = posto.numero AND Posto.fila = posto.fila) |
| POST-CONDIZIONE | <b>context:</b> create(posto: PostoBean): boolean<br><br><b>post:</b> Database.Posto->include(posto)                                                                                                                                                                                                        |

| NOME METODO     | +retriveById(id: int): SalaBean                                                                   |
|-----------------|---------------------------------------------------------------------------------------------------|
| DESCRIZIONE     | Questo metodo restituisce la sala in cui è situato il posto                                       |
| PRE-CONDIZIONE  | <b>context:</b> PostoDAO::retriveById(id: int): SalaBean<br><br><b>pre:</b> id != NULL AND id > 0 |
| POST-CONDIZIONE | <b>context:</b> retriveById(id: int): SalaBean                                                    |

|                                   |                  |
|-----------------------------------|------------------|
| Progetto: CineNow                 | Versione: 1.0    |
| Documento: Object Design Document | Data: 15/12/2024 |

|  |                                                                  |
|--|------------------------------------------------------------------|
|  | <b>post:</b> result = Database.Sala->select(Sala   Sala.id = id) |
|--|------------------------------------------------------------------|

|                 |                                                                                                                                                  |
|-----------------|--------------------------------------------------------------------------------------------------------------------------------------------------|
| NOME METODO     | +retriveBySala(idSala): List<PostoBean>                                                                                                          |
| DESCRIZIONE     | Questo metodo restituisce una lista di posti della sala                                                                                          |
| PRE-CONDIZIONE  | <b>context:</b> PostoDAO::retriveBySala(idSala): List<PostoBean><br><b>pre:</b> idSala != NULL AND idSala > 0                                    |
| POST-CONDIZIONE | <b>context:</b> PostoDAO::retriveBySala(idSala): List<PostoBean><br><b>post:</b> result = Database.Posto->select(Posto   Posto.id_sala = idSala) |

|                 |                                                                                                                                        |
|-----------------|----------------------------------------------------------------------------------------------------------------------------------------|
| NOME METODO     | +delete(id: int): boolean                                                                                                              |
| DESCRIZIONE     | Questo metodo permette al Gestore Sede di eliminare il posto                                                                           |
| PRE-CONDIZIONE  | <b>context:</b> Posto::delete(id: int): boolean<br><b>pre:</b><br>id != NULL AND<br>Database.Posto->exists(Posto   Posto.id_sala = id) |
| POST-CONDIZIONE | <b>context:</b> Posto::delete(id: int): boolean<br><b>post:</b> Database.Posto->!exists(Posto   Posto.id_sala = id)                    |

|             |                                                                                                        |
|-------------|--------------------------------------------------------------------------------------------------------|
| NOME CLASSE | <b>PostoProiezioneDAO</b>                                                                              |
| METODI      | +create(postoProiezione: PostoProiezioneBean): boolean<br>+retriveAllByProiezione(proiezioneId: int) : |



|                                   |                  |
|-----------------------------------|------------------|
| Progetto: CineNow                 | Versione: 1.0    |
| Documento: Object Design Document | Data: 15/12/2024 |

|            |                                                                                     |
|------------|-------------------------------------------------------------------------------------|
|            | List<PostoProiezione><br>+occupaPostoProiezione(posto:<br>PostoProiezione): boolean |
| INVARIANTI |                                                                                     |

|                 |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     |
|-----------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| NOME METODO     | +create(postoProiezione:PostoProiezioneBean):<br>boolean                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |
| DESCRIZIONE     | Questo metodo permette di inserire il posto<br>della prenotazione                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |
| PRE-CONDIZIONE  | <b>context:</b><br>PostoProiezioneDAO::create(postoProiezione:<br>PostoProiezioneBean): boolean<br><br><b>pre:</b><br>PostoProiezione != NULL AND<br>PostoProiezione.id_sala != NULL AND<br>Database.Sala->exists(Sala   Sala.id_sala =<br>id_sala) AND<br>PostoProiezione.fila != NULL AND<br>Database.Posto->exists(Posto   Posto.fila = fila)<br>AND<br>PostoProiezione.numero_posto != NULL AND<br>Database.Posto->exists(Posto   Posto.numero =<br>numero_posto) AND<br>PostoProiezione.id_proiezione != NULL<br>PostoProiezione.stato = False |
| POST-CONDIZIONE | <b>context:</b><br>PostoProiezioneDAO::create(postoProiezione:<br>PostoProiezioneBean): boolean<br><br><b>post:</b><br>Database.Posto_Proiezione->include(postoProi<br>ezione)                                                                                                                                                                                                                                                                                                                                                                      |

|             |                                                                       |
|-------------|-----------------------------------------------------------------------|
| NOME METODO | +retriveAllByProiezione(proiezioneId: int) :<br>List<PostoProiezione> |
|-------------|-----------------------------------------------------------------------|

|                                   |                  |
|-----------------------------------|------------------|
| Progetto: CineNow                 | Versione: 1.0    |
| Documento: Object Design Document | Data: 15/12/2024 |

|                 |                                                                                                                                                                                                                                              |
|-----------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| DESCRIZIONE     | Questo metodo restituisce la lista dei posti per quella prenotazione                                                                                                                                                                         |
| PRE-CONDIZIONE  | <b>context:</b><br>PostoProiezioneDAO::retriveAllByProiezione(proiezioneId: int) : List<PostoProiezione><br><br><b>pre:</b><br>proiezioneId != NULL AND<br>Database.Posto_Proiezione->exists(PostoPr   PostoPr.id_proiezione = proiezioneId) |
| POST-CONDIZIONE | <b>context:</b><br>PostoProiezioneDAO::retriveAllByProiezione(proiezioneId: int) : List<PostoProiezione><br><br><b>post:</b> result =<br>Database.Posto_Proiezione->select(PostoPr   PostoPr.id_proiezione = proiezioneId)                   |

|                 |                                                                                                                                      |
|-----------------|--------------------------------------------------------------------------------------------------------------------------------------|
| NOME METODO     | +occupaPostoProiezione(posto: PostoProiezione): boolean                                                                              |
| DESCRIZIONE     | Questo metodo permette di occupare un posto                                                                                          |
| PRE-CONDIZIONE  | <b>context:</b><br>PostoProiezioneDAO::occupaPostoProiezione(posto: PostoProiezione): boolean<br><br><b>pre:</b><br>posto != NULL    |
| POST-CONDIZIONE | <b>context:</b><br>PostoProiezioneDAO::occupaPostoProiezione(posto: PostoProiezione): boolean<br><br><b>post:</b> posto.stato = True |

|             |                                 |
|-------------|---------------------------------|
| NOME CLASSE | SlotDAO                         |
| METODI      | +retriveById(id: int): SlotBean |

|                                   |                  |
|-----------------------------------|------------------|
| Progetto: CineNow                 | Versione: 1.0    |
| Documento: Object Design Document | Data: 15/12/2024 |

|            |                                                                                                                           |
|------------|---------------------------------------------------------------------------------------------------------------------------|
|            | +retriveByProiezione(proiezioneId: int) : SlotBean<br>+retriveFreeSlot(start: LocalDate, end: LocalDate) : List<SlotBean> |
| INVARIANTI |                                                                                                                           |

|                 |                                                                                                                                            |
|-----------------|--------------------------------------------------------------------------------------------------------------------------------------------|
| NOME METODO     | +retriveById(id: int): SlotBean                                                                                                            |
| DESCRIZIONE     | Questo metodo permette di ottenere lo slot tramite l'id                                                                                    |
| PRE-CONDIZIONE  | <b>context:</b> SlotDAO::retriveById(id: int): SlotBean<br><br><b>pre:</b><br>id != NULL AND<br>Database.Slot->exists(Slot   Slot.id = id) |
| POST-CONDIZIONE | <b>context:</b> SlotDAO::retriveById(id: int): SlotBean<br><br><b>post:</b> result = Database.Slot->select(Slot   Slot.id = id)            |

|                 |                                                                                                                                                                          |
|-----------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| NOME METODO     | +retriveByProiezione(proiezioneId: int) : SlotBean                                                                                                                       |
| DESCRIZIONE     | Questo metodo permette di ottenere lo slot per quella determinata proiezione                                                                                             |
| PRE-CONDIZIONE  | <b>context:</b><br>SlotDAO::retriveByProiezione(proiezioneId: int) : SlotBean<br><br><b>pre:</b> Database.Proeizione->existst(Proiezione   Proiezione.id = proiezioneId) |
| POST-CONDIZIONE | <b>context:</b><br>SlotDAO::retriveByProiezione(proiezioneId: int) : SlotBean                                                                                            |

|                                   |                  |
|-----------------------------------|------------------|
| Progetto: CineNow                 | Versione: 1.0    |
| Documento: Object Design Document | Data: 15/12/2024 |

|  |                                                                             |
|--|-----------------------------------------------------------------------------|
|  | <b>post:</b> Database.Proiezione->select(Proiezione   Proiezione.id_orario) |
|--|-----------------------------------------------------------------------------|

|                 |                                                                                                              |
|-----------------|--------------------------------------------------------------------------------------------------------------|
| NOME METODO     | +retrieveFreeSlot(start: LocalDate, end: LocalDate) : List<SlotBeam>                                         |
| DESCRIZIONE     | Questo metodo permette di ottenere una lista di slot liberi in una determinata fascia oraria                 |
| PRE-CONDIZIONE  | <b>context:</b> SlotDAO::retrieveFreeSlot(start: LocalDate, end: LocalDate) : List<SlotBeam><br><b>pre:</b>  |
| POST-CONDIZIONE | <b>context:</b> SlotDAO::retrieveFreeSlot(start: LocalDate, end: LocalDate) : List<SlotBeam><br><b>post:</b> |

### 3.2 Application

|             |                                                                                                                                                                                                                                           |
|-------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| NOME CLASSE | <b>LoginControl</b>                                                                                                                                                                                                                       |
| METODI      | +doGet(request : HttpServletRequest, response : HttpServletResponse) : void<br>+doPost(request : HttpServletRequest, response : HttpServletResponse) : void<br>-login(email : String, password : String) : boolean<br>-logout() : boolean |
| INVARIANTI  |                                                                                                                                                                                                                                           |

|             |                                                     |
|-------------|-----------------------------------------------------|
| NOME METODO | -login(email : String, password : String) : boolean |
| DESCRIZIONE | Questo metodo permette di loggarsi.                 |

|                                   |                  |
|-----------------------------------|------------------|
| Progetto: CineNow                 | Versione: 1.0    |
| Documento: Object Design Document | Data: 15/12/2024 |

|                 |                                                                                                                                                                              |
|-----------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| PRE-CONDIZIONE  | <b>context:</b> LoginControl::login(email:String, password:String)<br><br><b>pre:</b><br>not email = NULL and not password = NULL                                            |
| POST-CONDIZIONE | <b>context:</b> LoginControl::login(email:String, password:String)<br><br><b>post:</b> result = (User.allInstances()->exists(u   u.email = email and u.password = password)) |

|                 |                                                                               |
|-----------------|-------------------------------------------------------------------------------|
| NOME METODO     | -logout() : boolean                                                           |
| DESCRIZIONE     | Questo metodo permette di effettuare il logout                                |
| PRE-CONDIZIONE  | <b>context:</b> LoginControl::logout()<br><br><b>pre:</b> sessionUser != NULL |
| POST-CONDIZIONE | <b>context:</b> LoginControl::logout()<br><br><b>post:</b> sessionUser = NULL |

|                 |                                                                                                                                                          |
|-----------------|----------------------------------------------------------------------------------------------------------------------------------------------------------|
| NOME METODO     | +doGet(request : HttpServletRequest, response : HttpServletResponse) : void                                                                              |
| DESCRIZIONE     | Il metodo che risponde alle richieste di tipo GET                                                                                                        |
| PRE-CONDIZIONE  | <b>context:</b><br>LoginControl::doGet(request:HttpServletRequest, response:HttpServletResponse)<br><br><b>pre:</b> request != NULL AND response != NULL |
| POST-CONDIZIONE | <b>context:</b><br>LoginControl::doGet(request:HttpServletRequest, response:HttpServletResponse)<br><br><b>post:</b>                                     |

|                                   |                  |
|-----------------------------------|------------------|
| Progetto: CineNow                 | Versione: 1.0    |
| Documento: Object Design Document | Data: 15/12/2024 |

|                 |                                                                                                                                                          |
|-----------------|----------------------------------------------------------------------------------------------------------------------------------------------------------|
| NOME METODO     | +doPost(request : HttpServletRequest, response : HttpServletResponse) : void                                                                             |
| DESCRIZIONE     | Il metodo che risponde alle richieste di tipo POST                                                                                                       |
| PRE-CONDIZIONE  | <b>context:</b><br>LoginControl::doGet(request:HttpServletRequest, response:HttpServletResponse)<br><br><b>pre:</b> request != NULL AND response != NULL |
| POST-CONDIZIONE | <b>context:</b><br>LoginControl::doGet(request:HttpServletRequest, response:HttpServletResponse)<br><br><b>post:</b>                                     |

|             |                                                                                                                                                                                                                                                                                                   |
|-------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| NOME CLASSE | <b>GestioneCatalogoControl</b>                                                                                                                                                                                                                                                                    |
| METODI      | +doGet(request : HttpServletRequest, response : HttpServletResponse) : void<br>+doPost(request : HttpServletRequest, response : HttpServletResponse) : void<br>-addFilmCatalogo(film : FilmBean) : boolean<br>-removeFilmCatalogo(idFilm : int) : boolean<br>-getCatalogo() : ArrayList<FilmBean> |
| INVARIANTI  | <b>inv:</b> self.getCatalogo()->isUnique(FilmBean   FilmBean.idFilm)                                                                                                                                                                                                                              |

|             |                                              |
|-------------|----------------------------------------------|
| NOME METODO | -addFilmCatalogo(film : FilmBean) : boolean  |
| DESCRIZIONE | Questo metodo permette di aggiungere un film |

|                                   |                  |
|-----------------------------------|------------------|
| Progetto: CineNow                 | Versione: 1.0    |
| Documento: Object Design Document | Data: 15/12/2024 |

|                 |                                                                                                                                                            |
|-----------------|------------------------------------------------------------------------------------------------------------------------------------------------------------|
|                 | al catalogo                                                                                                                                                |
| PRE-CONDIZIONE  | <b>context:</b><br>GestioneCatalogoControl::addFilmCatalogo(film:FilmBean)<br><br><b>pre:</b><br>not film = NULL AND<br>self.getCatalogo()->excludes(film) |
| POST-CONDIZIONE | <b>context:</b><br>GestioneCatalogoControl::addFilmCatalogo(film:FilmBean)<br><br><b>post:</b> self.getCatalogo()->includes(film)                          |

|                 |                                                                                                                                               |
|-----------------|-----------------------------------------------------------------------------------------------------------------------------------------------|
| NOME METODO     | -removeFilmCatalogo(idFilm : int) : boolean                                                                                                   |
| DESCRIZIONE     | Questo metodo permette di rimuovere un film.                                                                                                  |
| PRE-CONDIZIONE  | <b>context:</b><br>GestioneCatalogoControl::removeFilmCatalogo(idFilm:int)<br><br><b>pre:</b> idFilm > 0                                      |
| POST-CONDIZIONE | <b>context:</b><br>GestioneCatalogoControl::removeFilmCatalogo(idFilm:int)<br><br><b>post:</b> result =<br>self.getCatalogo()->excludes(film) |

|                                   |                  |
|-----------------------------------|------------------|
| Progetto: CineNow                 | Versione: 1.0    |
| Documento: Object Design Document | Data: 15/12/2024 |

|                 |                                                                           |
|-----------------|---------------------------------------------------------------------------|
| NOME METODO     | -getCatalogo() : ArrayList<FilmBean>                                      |
| DESCRIZIONE     | Questo metodo permette di ottenere il catalogo.                           |
| PRE-CONDIZIONE  | <b>context:</b><br>GestioneCatalogoControl::getCatalogo()<br><b>pre:</b>  |
| POST-CONDIZIONE | <b>context:</b><br>GestioneCatalogoControl::getCatalogo()<br><b>post:</b> |

|                 |                                                                                                                                                                 |
|-----------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------|
| NOME METODO     | +doGet(request : HttpServletRequest, response : HttpServletResponse) : void                                                                                     |
| DESCRIZIONE     | Il metodo che risponde alle richieste di tipo GET                                                                                                               |
| PRE-CONDIZIONE  | <b>context:</b><br>GestioneCatalogoControl::doGet(request:HttpServletRequest, response:HttpServletResponse)<br><b>pre:</b> request != NULL AND response != NULL |
| POST-CONDIZIONE | <b>context:</b><br>GestioneCatalogoControl::doGet(request:HttpServletRequest, response:HttpServletResponse)<br><b>post:</b>                                     |

|                |                                                                                                             |
|----------------|-------------------------------------------------------------------------------------------------------------|
| NOME METODO    | +doPost(request : HttpServletRequest, response : HttpServletResponse) : void                                |
| DESCRIZIONE    | Il metodo che risponde alle richieste di tipo POST                                                          |
| PRE-CONDIZIONE | <b>context:</b><br>GestioneCatalogoControl::doGet(request:HttpServletRequest, response:HttpServletResponse) |



|                                   |                  |
|-----------------------------------|------------------|
| Progetto: CineNow                 | Versione: 1.0    |
| Documento: Object Design Document | Data: 15/12/2024 |

|                 |                                                                                                                                 |
|-----------------|---------------------------------------------------------------------------------------------------------------------------------|
|                 | <b>pre:</b> request != NULL AND response != NULL                                                                                |
| POST-CONDIZIONE | <b>context:</b><br>GestioneCatalogoControl::doGet(request:HttpServletRequest, response:HttpServletResponse)<br><br><b>post:</b> |

| NOME CLASSE | PrenotazioneControl                                                                                                                                                                                                                                                                                                                                                                                         |
|-------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| METODI      | +doGet(request : HttpServletRequest, response : HttpServletResponse) : void<br>+doPost(request : HttpServletRequest, response : HttpServletResponse) : void<br>-effettuaPrenotazione(prenotazione : PrenotazioneBean) : boolean<br>-occupaPosto(posto : PostoProiezioneBean) : boolean<br>-calcolaTotale(numeroPosti : int) : double<br>-storicoOrdini(cliente : ClienteBean) : ArrayList<PrenotazioneBean> |
| INVARIANTI  |                                                                                                                                                                                                                                                                                                                                                                                                             |

|                |                                                                                                                                            |
|----------------|--------------------------------------------------------------------------------------------------------------------------------------------|
| NOME METODO    | -effettuaPrenotazione(prenotazione : PrenotazioneBean) : boolean                                                                           |
| DESCRIZIONE    | Questo metodo permette di effettuare una prenotazione.                                                                                     |
| PRE-CONDIZIONE | <b>context:</b><br>PrenotazioneControl::effettuaPrenotazione(prenotazione:PrenotazioneBean)<br><br><b>pre:</b> NOT prenotazione = NULL AND |

|                                   |                  |
|-----------------------------------|------------------|
| Progetto: CineNow                 | Versione: 1.0    |
| Documento: Object Design Document | Data: 15/12/2024 |

|                 |                                                                                                                                                                                                  |
|-----------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|                 | prenotazione.numeroPosti > 0                                                                                                                                                                     |
| POST-CONDIZIONE | <b>context:</b><br>PrenotazioneControl::effettuaPrenotazione(prenotazione:PrenotazioneBean)<br><br><b>post:</b> result =<br>self.storicoOrdini(prenotazione.cliente)-><br>includes(prenotazione) |

|                 |                                                                                                                                                     |
|-----------------|-----------------------------------------------------------------------------------------------------------------------------------------------------|
| NOME METODO     | -occupaPosto(posto : PostoProiezioneBean) : boolean                                                                                                 |
| DESCRIZIONE     | Questo metodo permette di occupare un posto.                                                                                                        |
| PRE-CONDIZIONE  | <b>context:</b><br>PrenotazioneControl::occupaPosto(posto:PostoProiezioneBean)<br><br><b>pre:</b> not posto = NULL AND not<br>posto.occupato = true |
| POST-CONDIZIONE | <b>context:</b><br>PrenotazioneControl::occupaPosto(posto:PostoProiezioneBean)<br><br><b>post:</b> result = (posto.occupato = true)                 |

|                 |                                                                                                      |
|-----------------|------------------------------------------------------------------------------------------------------|
| NOME METODO     | -calcolaTotale(numeroPosti : int) : double                                                           |
| DESCRIZIONE     | Questo metodo permette di ottenere il catalogo.                                                      |
| PRE-CONDIZIONE  | <b>context:</b><br>PrenotazioneControl::calcolaTotale(numeroPosti:int)<br><b>pre:</b> numeroPosti >0 |
| POST-CONDIZIONE | <b>context:</b><br>PrenotazioneControl::calcolaTotale(numeroPosti:int)<br><b>post:</b> result > 0.0  |

|                                   |                  |
|-----------------------------------|------------------|
| Progetto: CineNow                 | Versione: 1.0    |
| Documento: Object Design Document | Data: 15/12/2024 |

|                 |                                                                                                                                        |
|-----------------|----------------------------------------------------------------------------------------------------------------------------------------|
| NOME METODO     | -storicoOrdini(cliente : ClienteBean) :<br>ArrayList<PrenotazioneBean>                                                                 |
| DESCRIZIONE     | Il metodo che restituisce la lista delle prenotazioni di un cliente.                                                                   |
| PRE-CONDIZIONE  | <b>context:</b><br>PrenotazioneControl::storicoOrdini(cliente:ClienteBean)<br><br><b>pre:</b> not cliente = NULL                       |
| POST-CONDIZIONE | <b>context:</b><br>PrenotazioneControl::storicoOrdini(cliente:ClienteBean)<br><br><b>post:</b> result->forAll(p   p.cliente = cliente) |

|                 |                                                                                                                                                                    |
|-----------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| NOME METODO     | +doGet(request : HttpServletRequest, response : HttpServletResponse) : void                                                                                        |
| DESCRIZIONE     | Il metodo che risponde alle richieste di tipo GET                                                                                                                  |
| PRE-CONDIZIONE  | <b>context:</b><br>PrenotazioneControl::doGet(request:HttpServletRequest, response:HttpServletResponse)<br><br><b>pre:</b> not(request = NULL AND response = NULL) |
| POST-CONDIZIONE | <b>context:</b><br>PrenotazioneControl::doGet(request:HttpServletRequest, response:HttpServletResponse)<br><br><b>post:</b>                                        |

|             |                                                                              |
|-------------|------------------------------------------------------------------------------|
| NOME METODO | +doPost(request : HttpServletRequest, response : HttpServletResponse) : void |
|-------------|------------------------------------------------------------------------------|

|                                   |                  |
|-----------------------------------|------------------|
| Progetto: CineNow                 | Versione: 1.0    |
| Documento: Object Design Document | Data: 15/12/2024 |

|                 |                                                                                                                                                                 |
|-----------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------|
| DESCRIZIONE     | Il metodo che risponde alle richieste di tipo POST                                                                                                              |
| PRE-CONDIZIONE  | <b>context:</b><br>PrenotazioneControl::doGet(request:HttpServletRequest, response:HttpServletResponse)<br><br><b>pre:</b> request != NULL AND response != NULL |
| POST-CONDIZIONE | <b>context:</b><br>PrenotazioneControl::doGet(request:HttpServletRequest, response:HttpServletResponse)<br><br><b>post:</b>                                     |

|             |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |
|-------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| NOME CLASSE | GestioneProgrammazioneControl                                                                                                                                                                                                                                                                                                                                                                                                                                                                         |
| METODI      | +doGet(request : HttpServletRequest, response : HttpServletResponse) : void<br>+doPost(request : HttpServletRequest, response : HttpServletResponse) : void<br>-aggiungiProiezione(proiezione : ProiezioneBean) : boolean<br>-rimuoviProiezione(idProiezione : int) : boolean<br>-getSaleSlotDisponibili(data : LocalDate) : HashMap<SalaBean, List<SlotBean>><br>-bloccaSlot(proiezione : ProiezioneBean, slot : SlotBean) : boolean<br>-getProgrammazione(idSede : int) : ArrayList<ProiezioneBean> |
| INVARIANTI  |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |

|                                   |                  |
|-----------------------------------|------------------|
| Progetto: CineNow                 | Versione: 1.0    |
| Documento: Object Design Document | Data: 15/12/2024 |

|                 |                                                                                                                                                                                               |
|-----------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| NOME METODO     | -aggiungiProiezione(proiezione :<br>ProiezioneBean) : boolean                                                                                                                                 |
| DESCRIZIONE     | Questo metodo permette di effettuare una prenotazione.                                                                                                                                        |
| PRE-CONDIZIONE  | <b>context:</b><br>GestioneProgrammazioneControl::aggiungiProiezione(proiezione:ProiezioneBean)<br><br><b>pre:</b> NOT proiezione= NULL AND proiezione.id > 0                                 |
| POST-CONDIZIONE | <b>context:</b><br>GestioneProgrammazioneControl::aggiungiProiezione(proiezione:ProiezioneBean)<br><br><b>post:</b> result = self.getProgrammazione(proiezione.idSede)->includes(proiezione)) |

|                 |                                                                                                                                                                           |
|-----------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| NOME METODO     | -rimuoviProiezione(idProiezione : int) :<br>boolean                                                                                                                       |
| DESCRIZIONE     | Questo metodo permette di occupare un posto.                                                                                                                              |
| PRE-CONDIZIONE  | <b>context:</b><br>GestioneProgrammazioneControl::rimuoviProiezione(idProiezione:int)<br><br><b>pre:</b> idProiezione > 0                                                 |
| POST-CONDIZIONE | <b>context:</b><br>GestioneProgrammazioneControl::rimuoviProiezione(idProiezione:int)<br><br><b>post:</b> Database.proiezione->forAll(p   p.idProiezione != idProiezione) |

|                                   |                  |
|-----------------------------------|------------------|
| Progetto: CineNow                 | Versione: 1.0    |
| Documento: Object Design Document | Data: 15/12/2024 |

|                 |                                                                                                                     |
|-----------------|---------------------------------------------------------------------------------------------------------------------|
| NOME METODO     | -getSaleSlotDisponibili(data : LocalDate) :<br>HashMap<SalaBean, List<SlotBean>>                                    |
| DESCRIZIONE     | Questo metodo permette di ottenere gli slot disponibili per ogni sala.                                              |
| PRE-CONDIZIONE  | <b>context:</b><br>GestioneProgrammazioneControl::getSaleSlotDisponibili(data:LocalDate)<br><b>pre:</b> data!= NULL |
| POST-CONDIZIONE | <b>context:</b><br>PrenotazioneControl::calcolaTotale(numeroPosti:int)<br><b>post:</b> result > 0.0                 |

|                 |                                                                                                                                                     |
|-----------------|-----------------------------------------------------------------------------------------------------------------------------------------------------|
| NOME METODO     | -bloccaSlot(proiezione : ProiezioneBean, slot : SlotBean) : boolean                                                                                 |
| DESCRIZIONE     | Questo metodo permette bloccare gli slot per una proiezione                                                                                         |
| PRE-CONDIZIONE  | <b>context:</b><br>GestioneProgrammazioneControl::bloccaSlot(proiezione:ProiezioneBean, slot:SlotBean)<br><b>pre:</b> not(data= NULL AND slot=NULL) |
| POST-CONDIZIONE | <b>context:</b><br>GestioneProgrammazioneControl::bloccaSlot(proiezione:ProiezioneBean, slot:SlotBean)<br><b>post:</b>                              |

|                                   |                  |
|-----------------------------------|------------------|
| Progetto: CineNow                 | Versione: 1.0    |
| Documento: Object Design Document | Data: 15/12/2024 |

|                 |                                                                                                                                       |
|-----------------|---------------------------------------------------------------------------------------------------------------------------------------|
| NOME METODO     | -getProgrammazione(idSede : int) :<br>ArrayList<ProiezioneBean>                                                                       |
| DESCRIZIONE     | Questo metodo permette di ottenere la programmazione per una sede.                                                                    |
| PRE-CONDIZIONE  | <b>context:</b><br>GestioneProgrammazioneControl::getProgrammazione(idSede:int)<br><b>pre:</b> idSede>0                               |
| POST-CONDIZIONE | <b>context:</b><br>GestioneProgrammazioneControl::getProgrammazione(idSede:int)<br><b>post:</b> result->forAll(p   p.idSede = idSede) |

|             |                                                                                                                                                    |
|-------------|----------------------------------------------------------------------------------------------------------------------------------------------------|
| NOME CLASSE | <b>ProgrammazioneControl</b>                                                                                                                       |
| METODI      | +doGet(request : HttpServletRequest, response : HttpServletResponse) : void<br>-getProgrammazioneFilm(idFilm : int) :<br>ArrayList<ProiezioneBean> |
| INVARIANTI  |                                                                                                                                                    |

|                 |                                                                                                                                   |
|-----------------|-----------------------------------------------------------------------------------------------------------------------------------|
| NOME METODO     | -getProgrammazioneFilm(idFilm : int) :<br>ArrayList<ProiezioneBean>                                                               |
| DESCRIZIONE     | Questo metodo permette di ottenere la programmazione per una sede.                                                                |
| PRE-CONDIZIONE  | <b>context:</b><br>ProgrammazioneControl::getProgrammazioneFilm(idFilm:int)<br><b>pre:</b> idFilm>0                               |
| POST-CONDIZIONE | <b>context:</b><br>ProgrammazioneControl::getProgrammazioneFilm(idFilm:int)<br><b>post:</b> result->forAll(p   p.idFilm = idFilm) |

|                                   |                  |
|-----------------------------------|------------------|
| Progetto: CineNow                 | Versione: 1.0    |
| Documento: Object Design Document | Data: 15/12/2024 |

| NOME CLASSE | RegistrazioneControl                                                                                                                                                    |
|-------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| METODI      | +doGet(request : HttpServletRequest, response : HttpServletResponse) : void<br>-registrazione(cliente : ClienteBean) : boolean<br>-checkEmail(email : String) : boolean |
| INVARIANTI  |                                                                                                                                                                         |

| NOME METODO     | -registrazione(cliente : ClienteBean) : boolean                                                                                                                                                                          |
|-----------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| DESCRIZIONE     | Questo metodo permette la registrazione di un nuovo Cliente.                                                                                                                                                             |
| PRE-CONDIZIONE  | <b>context:</b><br>RegistrazioneControl::registrazione(cliente : ClienteBean)<br><b>pre:</b> not(cliente = NULL AND cliente.email = NULL AND cliente.email = ' ' AND cliente.password = NULL AND cliente.password = ' ') |
| POST-CONDIZIONE | <b>context:</b><br>RegistrazioneControl::registrazione(cliente : ClienteBean)<br><b>post:</b> Database.Cliente->excludes(cliente)                                                                                        |

| NOME METODO     | -checkEmail(email : String) : boolean                                                                              |
|-----------------|--------------------------------------------------------------------------------------------------------------------|
| DESCRIZIONE     | Questo metodo permette di verificare se un email è già presente.                                                   |
| PRE-CONDIZIONE  | <b>context:</b><br>RegistrazioneControl::checkEmail(email:String)<br><b>pre:</b> not(email = NULL AND email = ' ') |
| POST-CONDIZIONE | <b>context:</b> RegistrazioneControl::checkEmail(email:String)                                                     |



|                                   |                  |
|-----------------------------------|------------------|
| Progetto: CineNow                 | Versione: 1.0    |
| Documento: Object Design Document | Data: 15/12/2024 |

|  |                                                                        |
|--|------------------------------------------------------------------------|
|  | <b>post:</b> result = (Database.Cliente->forAll(c   c.email != email)) |
|--|------------------------------------------------------------------------|

|                    |                                                                                                                                                                         |
|--------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>NOME CLASSE</b> | <b>GestioneSaleControl</b>                                                                                                                                              |
| <b>METODI</b>      | +doGet(request : HttpServletRequest, response : HttpServletResponse) : void<br>-registrazione(cliente : ClienteBean) : boolean<br>-checkEmail(email : String) : boolean |
| <b>INVARIANTI</b>  |                                                                                                                                                                         |

|                        |                                                                                                                                         |
|------------------------|-----------------------------------------------------------------------------------------------------------------------------------------|
| <b>NOME METODO</b>     | -addSala(sala : SalaBean) : boolean                                                                                                     |
| <b>DESCRIZIONE</b>     | Questo metodo permette di aggiungere una sala                                                                                           |
| <b>PRE-CONDIZIONE</b>  | <b>context:</b><br>GestioneSaleControl::addSala(sala:SalaBean)<br><b>pre:</b> not(sala = NULL AND sala.id>0)                            |
| <b>POST-CONDIZIONE</b> | <b>context:</b><br>GestioneSaleControl::addSala(sala:SalaBean)<br><b>post:</b> result = Database.Sala->select(Sala   Sala.id = sala.id) |

|                        |                                                                                                                                  |
|------------------------|----------------------------------------------------------------------------------------------------------------------------------|
| <b>NOME METODO</b>     | -removeSala(sala:SalaBean) : boolean                                                                                             |
| <b>DESCRIZIONE</b>     | Questo metodo permette di aggiungere una sala                                                                                    |
| <b>PRE-CONDIZIONE</b>  | <b>context:</b> GestioneSaleControl::removeSala(sala :SalaBean)<br><b>pre:</b> not(sala = NULL AND sala.id>0)                    |
| <b>POST-CONDIZIONE</b> | <b>context:</b> GestioneSaleControl::removeSala(sala :SalaBean)<br><b>post:</b> Database.Sala->!exists(Sala   Sala.id = sala.id) |

|                                   |                  |
|-----------------------------------|------------------|
| Progetto: CineNow                 | Versione: 1.0    |
| Documento: Object Design Document | Data: 15/12/2024 |

## 4. GLOSSARIO

|                            |                                                                                                                                 |
|----------------------------|---------------------------------------------------------------------------------------------------------------------------------|
| <b>Prenotazione</b>        | Rappresenta l'acquisto di uno o più tickets da parte di un cliente registrato del sistema                                       |
| <b>Checkout</b>            | Inserimento dei dati per effettuare l'acquisto                                                                                  |
| <b>Catalogo</b>            | Elenco di film disponibili in una sede                                                                                          |
| <b>Cliente (UTC)</b>       | Utente registrato, utilizza il sistema per prenotare i biglietti                                                                |
| <b>Gestore Sede(UGS)</b>   | Utente registrato, che gestisce una sede specifica della catena Movieplex. Egli gestisce le proiezioni della sua sede e le sale |
| <b>Gestore Catena(UGC)</b> | Utente registrato, che gestisce la catena Movieplex. Egli gestisce i film proiettabili e le sedi della catena                   |
| <b>Piantina sala</b>       | Visualizzazione grafica che modella la sala fisica della sede.                                                                  |
| <b>Proiezione</b>          | Specifico spettacolo di un film, a una data e orario definiti, associato a una sala.                                            |
| <b>Programmazione</b>      | Insieme delle proiezioni di un cinema.                                                                                          |