

Gestione Museo PRO

Luigi Smargiassi, Francesco Zaritto e Sara Montese

18 luglio 2021



Progetto di Ingegneria del Software

Anno accademico 2020/2021

Sotto la guida di

Prof. Domenico Ursino e Ing. Enrico Corradini

Sommario

Questo report specifica i vari processi e le tecniche utilizzate nella raccolta dei requisiti, progettazione, implementazione e test per sistema di gestione di un museo. Sulla base dei requisiti raccolti dagli stakeholder abbiamo creato un modello e progettato il software basato su di esso.

Indice

1	Descrizione del contesto	3
2	Intervista	4
3	Descrizione del sistema in linguaggio naturale	4
3.1	Glossario dei termini	5
4	Diagramma dei sistemi	6
5	Analisi dei requisiti	7
5.1	Requisiti funzionali	7
5.2	Requisiti non funzionali	9
6	Design del sistema	10
6.1	Diagramma dei casi d'uso	10
6.2	Casi d'uso: Attori	10
6.3	Casi d'uso: Gestione Dipendenti	11
6.4	Casi d'uso: Gestione Visite	13
6.5	Casi d'uso: Gestione Prenotazioni	14
6.6	Casi d'uso: Gestione Opere	16
6.7	Casi d'uso: Gestione Mostre	18
6.8	Casi d'uso: Calcolo Statistiche	19
6.9	Diagrammi dei casi d'uso complessivi	20
6.10	Matrice di mapping	23
6.11	Diagramma Entità-Relazioni	24
6.11.1	Schema ER	24
6.11.2	Schema logico	24
6.11.3	Implementazione del database	25
6.12	Mappa dell'architettura	27
6.13	Diagramma delle classi	30
6.14	Diagrammi di sequenza	38
6.15	Diagrammi di attività	50
7	Dettagli implementativi	60
7.1	Graphic User Interface	60
7.1.1	Login	60
7.1.2	Homepage	61
7.1.3	Amministratore	61
7.1.4	Catalogatore	64
7.1.5	Front-office	65
8	Test del software	66

1 Descrizione del contesto

Visitare un museo è come compiere un viaggio emozionante alla scoperta di civiltà lontane, talvolta scomparse. Con i suoi oggetti – opere d'arte, reperti archeologici e tanti altri materiali – il museo, in pochi secoli, è diventato uno degli strumenti principali per educare il pubblico alla conoscenza, per metterlo in contatto con la propria civiltà e con quella di mondi diversi. La visita al museo permette un contatto diretto con le opere originali, con dipinti, oggetti e reperti di cui ogni giorno vediamo le riproduzioni in televisione o sui giornali e può contribuire a far nascere un sentimento di appartenenza verso il territorio in cui si vive o da cui si proviene.

Il museo offre un servizio culturale alla comunità e ciò significa che tutte le sue attività sono organizzate per garantire lo svolgimento di alcune funzioni.

Le funzioni di un museo si raggruppano in due ambiti principali:

- Funzioni orientate verso la collezione (cura e gestione delle collezioni)
- Funzioni orientate verso il pubblico (attività educative, comunicazione e accoglienza dei visitatori)

Per quanto riguarda le collezioni, il museo si occupa della documentazione e registrazione delle opere tramite la catalogazione (descrizione fisica dell'oggetto e della sua storia). Dal punto di vista organizzativo, il rapporto del museo con il pubblico si configura come un servizio. Ogni museo osserva un orario di apertura, che tende a conciliarsi con quello del tempo libero, ma deve anche coincidere con quello scolastico, perché le scolaresche sono una delle principali fasce di utenza. Oltre alla dimensione temporale, l'accessibilità del museo ha anche una dimensione spaziale. Al visitatore vanno risparmiate per quanto possibili code all'entrata o affollamento delle sale e fornire strumenti di orientamento nel museo.

Con l'avvento del Covid, per di più, la realtà *Museo* è chiamata anch'essa ad adottare nuove norme e ad adattare il proprio sistema a nuove esigenze come ingressi contingentati e favorendo le prenotazioni online.

Il personale del museo, ossia l'insieme delle risorse umane che concorrono al suo funzionamento, si occupa sia della collezione sia del pubblico. Le funzioni che il personale deve garantire sono molteplici: restauro, conservazione, sicurezza, lo studio, la catalogazione... Di recente si sono aggiunti anche compiti che riguardano la comunicazione, il marketing e la gestione delle attività commerciali. La nostra mission è offrire a chi lavora nei musei soluzioni innovative per affrontare al meglio la trasformazione digitale, senza timori. Dotarsi di un buon software per la gestione permetterà di lavorare con maggiore sicurezza e facilità.

2 Intervista

INGEGNERE : Dalla descrizione, ci avete fornito una panoramica delle funzioni e dei processi chiave del vostro museo. Possedete già un software gestionale?

GESTORE MUSEO : No, avevamo un sistema per la semplice catalogazione delle nostre opere d'arte ma è ormai obsoleto e molto lento. Vorremmo investire in un software nuovo, innovativo, che incrementi l'efficienza del lavoro semplificando notevolmente le nostre attività, con l'obiettivo di registrare, nel lungo periodo, un importante risparmio in termini economici per tutta l'organizzazione museale.

INGEGNERE : *Gestione Museo PRO* è il software gestionale che viene incontro a tutte le esigenze di un museo moderno. Quali sono i processi che vorreste automatizzare?

GESTORE MUSEO : Il sistema dovrà, oltre a offrire un servizio di catalogazione delle opere d'arte e pianificazione delle mostre, monitorare il personale e gli incassi. Tale soluzione dovrà inoltre prevedere la funzione di biglietteria automatizzata e fornire report dettagliati sulla nostra attività.

INGEGNERE : Quali problemi vorreste che il software risolvesse?

GESTORE MUSEO : In epoca pandemica, è fondamentale per i musei adeguarsi alle nuove misure adottate dal Governo per la prevenzione e il controllo del covid-19 nei luoghi di cultura. La soluzione dovrà garantire la possibilità di gestire gli ingressi in modo contingentato per evitare assembramenti.

INGEGNERE : Chi userà il sistema?

GESTORE MUSEO : Gli operatori della biglietteria e i catalogatori devono poter accedere al sistema con le loro credenziali e saranno autorizzati ad utilizzare la sezione di sito limitata alla loro mansione. Solo l'amministratore avrà libero accesso ad ogni parte del sito e ai report.

INGEGNERE : Quali dati prevedete di raccogliere?

GESTORE MUSEO : Il software dovrà presentare statistiche periodiche sulle mostre con maggiore e minore affluenza, sui tempi di visita e sugli orari in cui vi è maggiore affollamento, per limitare il rischio di contagi e rispettare la normativa a riguardo.

3 Descrizione del sistema in linguaggio naturale

Il progetto proposto consiste nella realizzazione di un sistema informativo per la gestione di uno museo. Il software progettato e realizzato dovrà, innanzitutto, consentire la gestione degli ingressi contingentati all'interno della struttura. Il visitatore potrà prenotare preventivamente la visita e presentarsi alla cassa il giorno prestabilito per il pagamento del ticket, o acquistare direttamente il biglietto al momento dell'ingresso.

Gli ingressi avvengono ogni 30 minuti fino al raggiungimento della capienza massima; raggiunto il limite di visitatori, non sarà più possibile accedere alla struttura.

Il software dovrà consentire la catalogazione e la gestione delle opere che costituiscono il patrimonio del museo, e dovrà consentire la programmazione di mostre ed esposizioni all'interno della struttura, tenendo traccia dei periodi di permanenza delle varie opere in tali eventi.

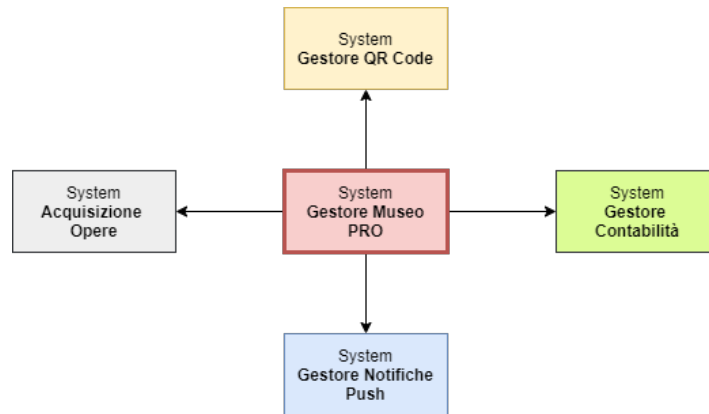
I lavoratori del museo avranno accesso al sistema con privilegi diversi in base al proprio ruolo. In particolare, l'amministratore potrà gestire gli account dei dipendenti e visualizzare le statistiche riguardo ai dati raccolti nelle varie attività del museo.

3.1 Glossario dei termini

TERMINE	SIGNIFICATO	TIPO	SINONIMO
Inventario	Descrizione di opere contenute in un archivio	Business	
Mostra	Esposizione di oggetti (o di opere) al pubblico	Business	Collezione
Catalogazione	Registrazione di un'opera o di una serie di opere in un catalogo	Business	Schedatura
Prenotazione	Atto mediante il quale viene riservato un servizio ad un cliente per un certo tempo; essa viene confermata al momento dell'ingresso	Business	
Visita	Atto mediante il quale si accede agli spazi museali	Business	Ingresso
Amministratore	Gestore del sistema. Può visualizzare i report.	Business	
Catalogatore museale	Colui che svolge attività d'inventariazione e catalogazione del patrimonio museale	Business	
Operatore Front-office	Addetto al servizio vendita e informazioni	Business	Receptionist
Sala espositiva	Sala che ospita opere del museo	Business	
Opera d'arte	Creazione artistica	Business	Composizione, quadro, tela, statua, scultura...
Biglietto	Documento che viene rilasciato al cliente dietro pagamento e che consente di visitare il museo.	Business	Ticket
Visitatore	Chi visita un museo, un'esposizione e simili		Cliente
Collezione permanente	L'esposizione permanente delle opere del museo	Business	
Collezione temporanea	Mostra temporanea delle opere del museo	Business	
Report			Statistica
Username	Identificativo di ciascun utente	Tecnico	ID
Utente	Soggetto che fruisce del sistema	Tecnico	Catalogatore, admin, Front-office
Password	Serie di caratteri alfanumerici che costituisce la parola d'ordine per accedere al sistema	Tecnico	

Figura 1: Glossario dei termini utilizzati.

4 Diagramma dei sistemi



I sistemi esterni coinvolti sono:

- *Gestore Contabilità*: sistema per la gestione degli stipendi, delle entrate e delle uscite
- *Gestore Notifiche Push*: sistema che notifica i visitatori con dettagli sulla prenotazione e aggiornamenti sugli eventi della struttura
- *Gestore QR Code*: sistema per la lettura del QR code associato ad una prenotazione
- *Acquisizione Opere*: sistema che gestisce l'acquisto di nuovi pezzi da collezione e la partecipazione ad aste per l'acquisizione di nuove opere

5 Analisi dei requisiti

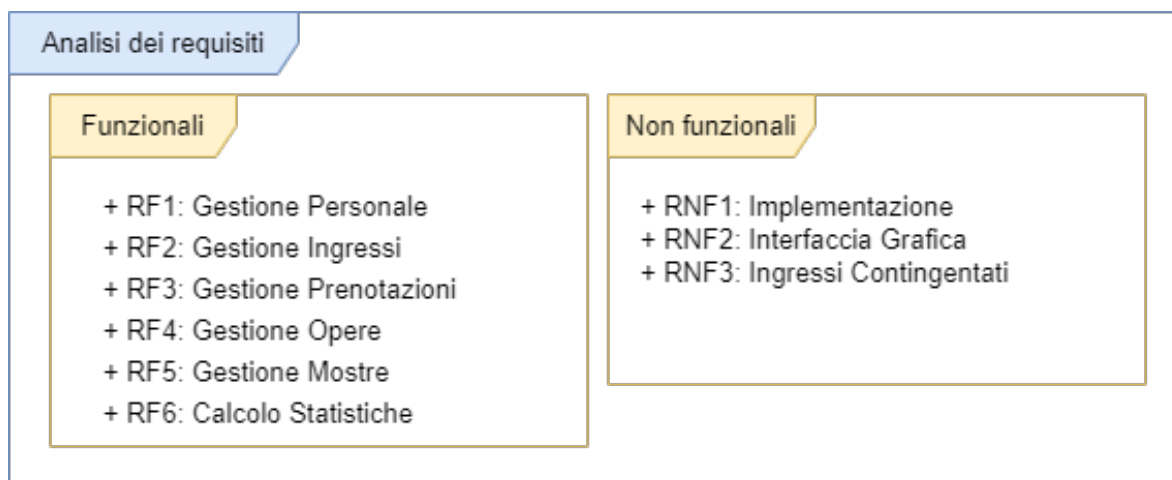


Figura 2: Schema generale dei Requisiti

5.1 Requisiti funzionali

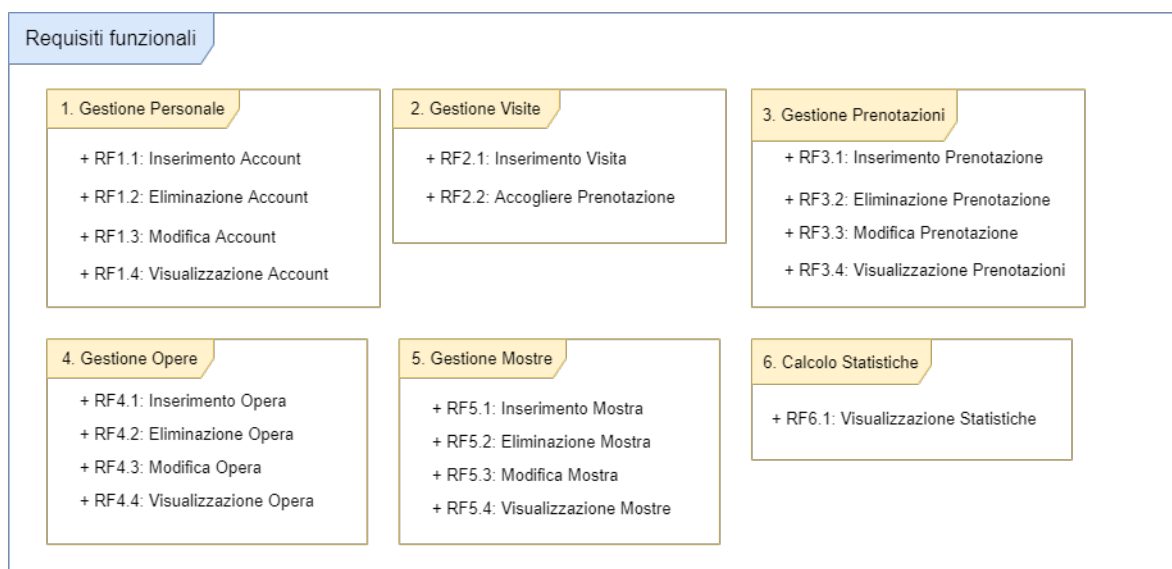


Figura 3: Schema dei Requisiti Funzionali

RF1: Gestione Personale

Requisito	Descrizione
RF1.1 Inserimento Account	Il sistema dovrà consentire all'admin di inserire account relativi ai dipendenti
RF1.2 Eliminazione Account	Il sistema dovrà consentire all'admin di eliminare gli account relativi ai dipendenti
RF1.3 Modifica Account	Il sistema dovrà consentire all'admin di modificare gli account relativi ai dipendenti, ad eccezione delle credenziali Il sistema dovrà consentire ad ogni utente di modificare la propria password
RF1.4 Visualizzazione Account	Il sistema dovrà consentire all'admin di visualizzare la lista degli account dei dipendenti

RF2: Gestione Visite

Requisito	Descrizione
RF2.1 Inserimento Visita	Il sistema dovrà registrare le referenze dei clienti e il tipo di visita che essi vogliono effettuare al momento dell'ingresso
RF2.2 Accogliere Prenotazione	Il sistema dovrà gestire gli ingressi dei visitatori con prenotazione

RF3: Gestione Prenotazioni

Requisito	Descrizione
RF3.1 Inserimento Prenotazione	Il sistema dovrà consentire la prenotazione di una visita, tenendo traccia delle referenze di chi prenota
RF3.2 Eliminazione Prenotazione	Il sistema dovrà consentire di annullare la prenotazione fino al giorno della visita
RF3.3 Modifica Prenotazione	Il sistema dovrà consentire la modifica di una prenotazione fino al giorno della visita
RF3.4 Visualizzazione Prenotazioni	Il sistema dovrà consentire di visualizzare la lista delle prenotazioni per il giorno corrente e i successivi

RF4: Gestione Opere

Requisito	Descrizione
RF4.1 Inserimento Opera	Il sistema dovrà consentire l'inserimento di un'opera nel catalogo
RF4.2 Eliminazione Opera	Il sistema dovrà consentire l'eliminazione di un'opera nel catalogo
RF4.3 Modifica Opera	Il sistema dovrà consentire la modifica di un'opera nel catalogo
RF4.4 Visualizzazione Opere	Il sistema dovrà permettere la visualizzazione degli elementi dell'archivio

RF5: Gestione Mostre

Requisito	Descrizione
RF5.1 Inserimento Mostra	Il sistema dovrà consentire all'admin di programmare una nuova mostra e di specificare le opere in essa impiegate
RF5.2 Eliminazione Mostra	Il sistema dovrà consentire all'admin di eliminare una mostra
RF5.3 Modifica Mostra	Il sistema dovrà consentire all'admin di modificare una mostra
RF5.4 Visualizzazione Mostre	Il sistema dovrà consentire all'admin di visualizzare la lista degli eventi che si terranno nel museo

RF6: Calcolo Statistiche

Requisito	Descrizione
RF6.1 Visualizzazione Statistiche	Il sistema dovrà consentire all'admin di visualizzare statistiche basate sui dati raccolti dalle altre attività del sistema

5.2 Requisiti non funzionali

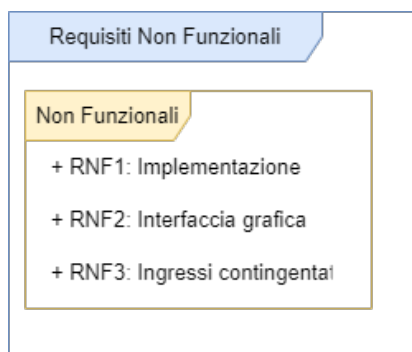


Figura 4: Descrizione dei Requisiti Non Funzionali

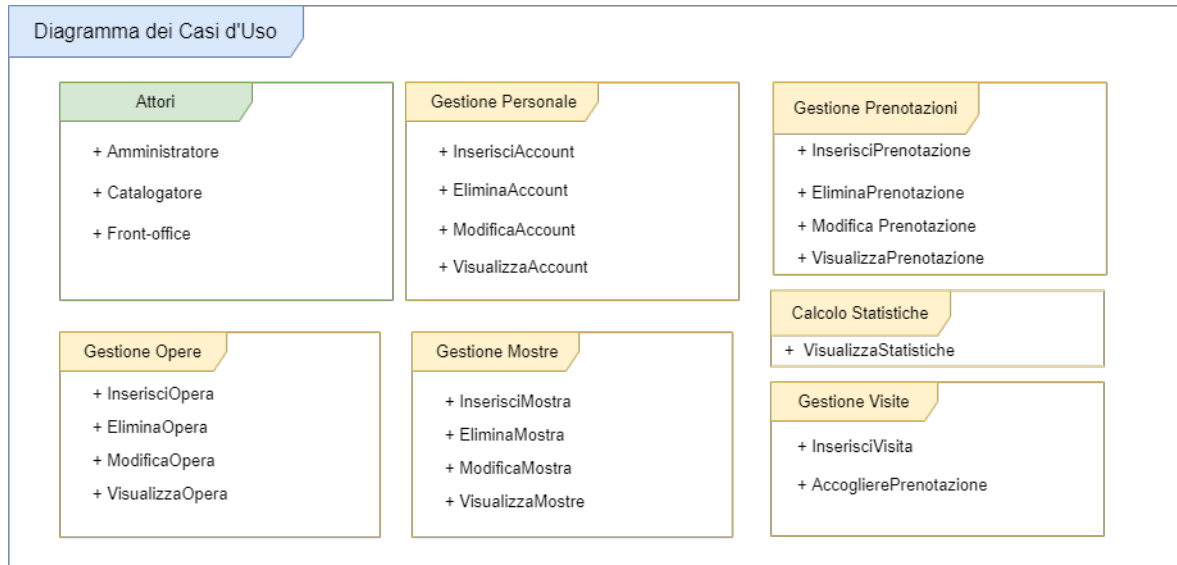
Requisito	Descrizione
RNF1 Implementazione	Il sistema dovrà essere realizzato in tecnologia Python 3.
RNF2 Interfaccia grafica	Il sistema dovrà essere dotato di interfaccia grafica
RNF3 Ingressi <u>contingentati</u>	Il sistema, nel registrare le entrate dovrà tenere conto che per ragioni di sicurezza, i visitatori potranno essere, allo stesso tempo, solo un tot, modificabile tramite file di configurazione

Figura 5: Schema dei Requisiti Non Funzionali

6 Design del sistema

6.1 Diagramma dei casi d'uso

Una volta definiti e spiegati per bene i requisiti del nostro sistema, andiamo a definire i casi d'uso, ovvero gli scenari possibili che si potranno verificare durante l'utilizzo del nostro software da parte degli utenti.

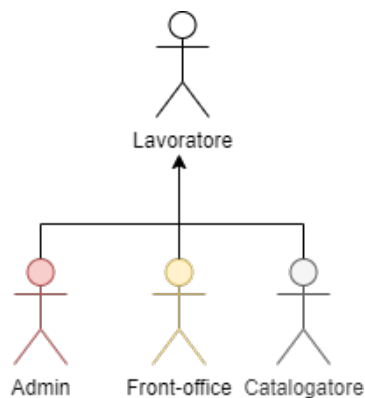


6.2 Casi d'uso: Attori

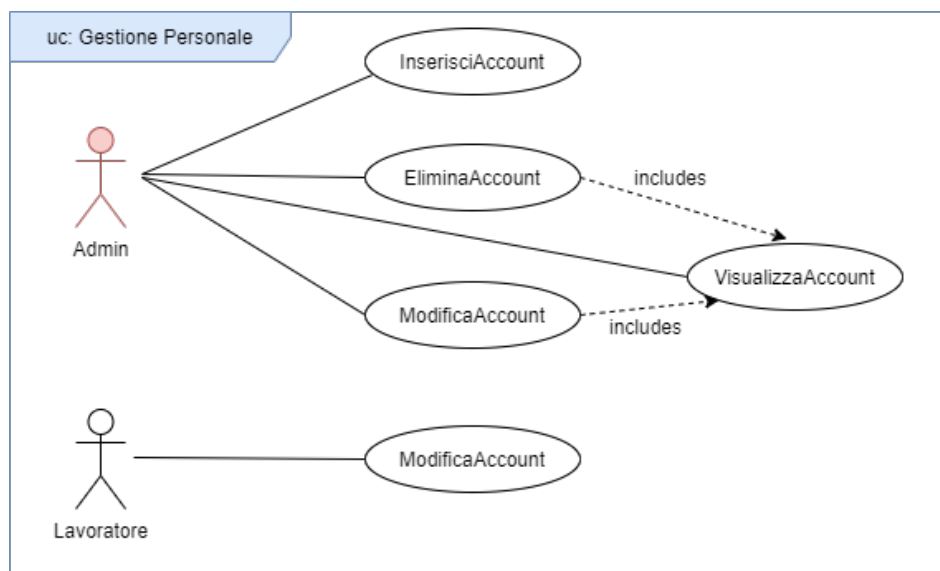
Nel nostro software abbiamo tre diversi attori che interagiscono con il sistema. Essi disporranno di un account, con username e password, e a seconda del ruolo godranno di privilegi diversi:

- Front-office: questa tipologia di utente si occupa della gestione degli ingressi e delle prenotazioni.
- Catalogatore: questa tipologia di utente si occupa della catalogazione e gestione delle opere del museo.
- Amministratore: è l'utente con più privilegi. Oltre alle funzionalità base, l'admin può gestire il personale, gli eventi che si tengono nella struttura e visualizzare le statistiche relative all'attività del museo.

Agli utenti viene assegnata una password di default; spetterà a loro definirne una personale.



6.3 Casi d'uso: Gestione Dipendenti



1) Inserisci

Attori	Utente "Amministratore"
Pre condizioni	L'attore deve essere correttamente loggato; il dipendente non deve essere già presente in elenco.
Post condizioni	Il dipendente è inserito nel relativo database e vengono forniti ID e password per il riconoscimento. Se non è stato possibile aggiungerlo, viene rigettato.
Sequenza eventi principali	<ol style="list-style-type: none"> 1) Controllo dei privilegi dell'utente che fa richiesta. 2) Richiesta di inserimento dati: nome, cognome e ruolo. 3) Controlla che non vi sia già un dipendente con stesse generalità. 4) Se è possibile, inserisce in archivio il dipendente. 5) Rilascia un ID e password.
Sequenza eventi alternativa	<p>Da 1), se l'utente non dispone di un account con i privilegi necessari, non può proseguire.</p> <p>Da 3), se non è possibile inserire il dipendente, dopo la notifica dell'impossibilità della memorizzazione, si torna al punto 2), in modo da reinserire i dati.</p>

2) Visualizza

Attori	Utente "Amministratore"
Pre condizioni	L'utente deve essere correttamente loggato.
Post condizioni	Nel caso si sia scelto di eliminare dall'elenco o modificare i dati riguardanti un dipendente, si passa al relativo caso d'uso.
Sequenza eventi principali	<ol style="list-style-type: none"> 1) Controllo dei privilegi dell'utente che fa richiesta. 2) Richiesta di inserimento filtri per i dipendenti da visualizzare. 3) Specificati i filtri, fornisce l'elenco dei dipendenti filtrato. 4) OPZIONALE: Mostra ulteriori informazioni su un determinato dipendente selezionato 5) Sarà possibile eliminare o modificare i dati dei dipendenti, e modificare nuovamente i filtri della ricerca
Sequenza eventi alternativa	Da 1), se l'utente non è loggato con un utente con i necessari requisiti, non si può proseguire.

3) Modifica

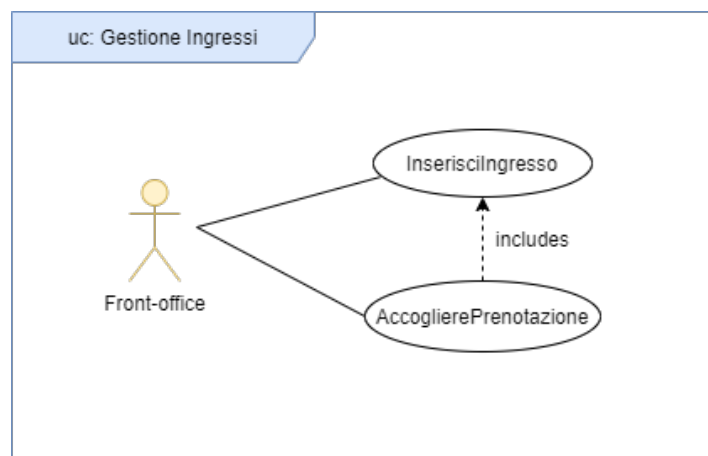
Attori	Utente "Amministratore"
Pre condizioni	Il dipendente deve essere presente nell'elenco.
Post condizioni	I dati del dipendente sono modificati nel relativo database, oppure la modifica viene rigettata se non è stato possibile modificarli. L'username e la password non possono essere modificati.
Sequenza eventi principali	1) Richiesta di inserimento dati per la modifica. 2) Se è possibile, inserisce in archivio i dati sul dipendente.
Sequenza eventi alternativa	Da 2), se non è possibile effettuare la modifica, si notifica dell'impossibilità dell'azione e si torna al punto 1).

Attori	Utente generico
Pre condizioni	L'utente deve essere correttamente loggato.
Post condizioni	La password dell'utente viene modificata nel relativo database
Sequenza eventi principali	1. Richiesta di inserimento dati per la modifica. 2. Inseriti i dati, controlla 3. Se è possibile, inserisce in archivio i dati sul dipendente.
Sequenza eventi alternativa	Da 2), se non è possibile modificare il dato, dopo la notifica dell'impossibilità della modifica, si torna al punto 1), in modo da reinserire i dati.

4) Elimina

Attori	Utente "Amministratore"
Pre condizioni	Il dipendente deve essere presente nell'elenco.
Post condizioni	Il dipendente è eliminato dall'elenco.
Sequenza eventi principali	1) Il dipendente selezionato è eliminato dall'elenco.
Sequenza eventi alternativa	/

6.4 Casi d'uso: Gestione Visite



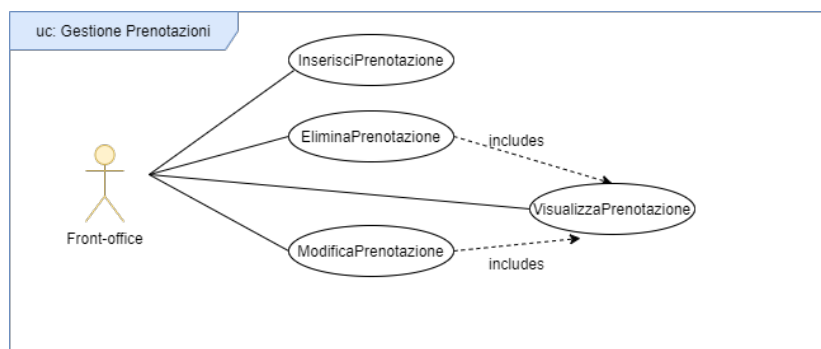
1) InserisciIngresso

Attori	Utente "Front Office"
Pre condizioni	L'utente deve essere correttamente loggato .
Post condizioni	La visita è inserita nel relativo database.
Sequenza eventi principali	<ol style="list-style-type: none">1) Controllo dei privilegi dell'utente che fa richiesta.2) Richiesta di inserimento dati: numero visitatori, generalità e tipo di visita.3) Inseriti i dati, l'orario di ingresso è posto appena il numero di visitatori all'interno sommato alle prenotazioni per la giornata odierna scende sotto il massimo ammesso.4) OPZIONALE: All'uscita memorizza anche l'orario di uscita.
Sequenza eventi alternativa	Da 1), se l'utente non è loggato con un utente con i necessari requisiti, non si può proseguire.

2) AccoglierePrenotazione

Attori	Utente "Front Office"
Pre condizioni	L'utente deve essere correttamente loggato ; la prenotazione deve essere presente in elenco nella data odierna.
Post condizioni	La prenotazione è convertita in visita e inserita nel relativo database e viene fornito un ID per il riconoscimento della visita. Viene rigettata se non è stato possibile aggiungerla.
Sequenza eventi principali	<ol style="list-style-type: none">1) Controllo dei privilegi dell'utente che fa richiesta.2) Tramite l'ID si risale alla prenotazione in elenco.3) Se la prenotazione corrisponde la data odierna, viene generata la relativa visita; l'orario di ingresso è posto appena il numero di visitatori all'interno scende sotto il massimo ammesso.4) OPZIONALE: All'uscita memorizza anche l'orario di uscita.
Sequenza eventi alternativa	Da 1), se l'utente non è loggato con un utente con i necessari requisiti, non si può proseguire.

6.5 Casi d'uso: Gestione Prenotazioni



1) Inserisci

Attori	Utente "Front Office"
Pre condizioni	L'utente deve essere correttamente loggato; nella data in cui andrà inserita la prenotazione devono essere presenti meno prenotazioni del limite massimo consentito.
Post condizioni	La prenotazione è inserita nel relativo database, oppure viene rigettata se non è stato possibile aggiungerla, e viene fornito un ID per il riconoscimento della prenotazione.
Sequenza eventi principali	<ol style="list-style-type: none"> 1) Controllo dei privilegi dell'utente che fa richiesta. 2) Richiesta di inserimento dati per la prenotazione: data visita, numero visitatori, generalità e tipo di visita. 3) Inseriti i dati, controlla la possibilità che la visita abbia luogo. 4) Se è possibile, inserisce in archivio la prenotazione. 5) Rilascia un ID univoco relativo la prenotazione.
Sequenza eventi alternativa	<p>Da 1), se l'utente non è loggato con un utente con i necessari requisiti, non si può proseguire.</p> <p>Da 3), se non è possibile inserire la prenotazione, dopo la notifica dell'impossibilità della memorizzazione della prenotazione, si torna al punto 2), in modo da reinserire i dati.</p>

2) Visualizza

Attori	Utente "Front Office"
Pre condizioni	L'utente deve essere correttamente loggato.
Post condizioni	Nel caso si sia scelto di eliminare o modificare una prenotazione si passa al relativo caso d'uso.
Sequenza eventi principali	<ol style="list-style-type: none"> 1) Controllo dei privilegi dell'utente che fa richiesta. 2) Richiesta di inserimento filtri per le prenotazioni da visualizzare. 3) Inseriti i dati, fornisce l'elenco delle prenotazioni filtrate. 4) OPZIONALE: Mostra ulteriori informazioni su una determinata prenotazione selezionata 5) Sarà possibile eliminare o modificare le prenotazioni, e modificare nuovamente i filtri della ricerca
Sequenza eventi alternativa	Da 1), se l'utente non è loggato con un utente con i necessari requisiti, non si può proseguire.

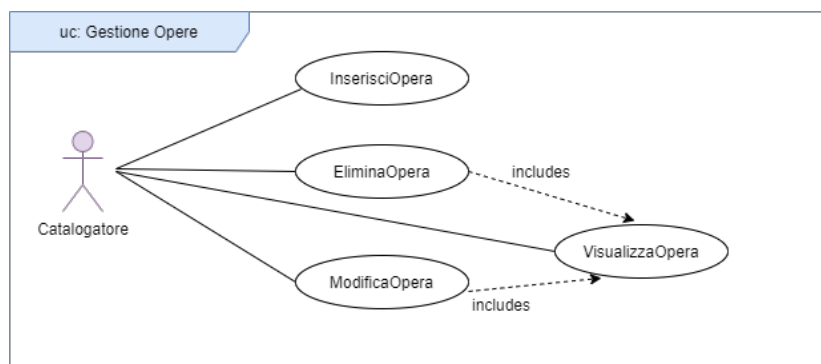
3) Modifica

Attori	Utente "Front Office"
Pre condizioni	La prenotazione deve essere presente nell'elenco; la nuova prenotazione modificata deve rispettare il numero massimo di prenotazioni per la data in cui va inserita.
Post condizioni	I dati della prenotazione sono modificati nel relativo database, oppure viene rigettata se non è stato possibile modificarla. L'ID non può essere modificato.
Sequenza eventi principali	<ol style="list-style-type: none">1) Inserimento identificativo della prenotazione da modificare.2) Inseriti i dati, controlla la possibilità che la visita abbia luogo.3) Se è possibile, inserisce in archivio la prenotazione.
Sequenza eventi alternativa	Da 2), se non è possibile inserire la prenotazione, dopo la notifica dell'impossibilità della modifica della prenotazione, si torna al punto 1), in modo da reinserire i dati.

4) Elimina

Attori	Utente "Front Office"
Pre condizioni	La prenotazione deve essere presente nell'elenco.
Post condizioni	La prenotazione è eliminata dall'elenco.
Sequenza eventi principali	<ol style="list-style-type: none">1) La prenotazione selezionata viene eliminata dall'elenco
Sequenza eventi alternativa	/

6.6 Casi d'uso: Gestione Opere



1) Inserisci

Attori	Utente "Catalogatore"
Pre condizioni	L'utente deve essere correttamente loggato, non deve essere già presente l'elemento nel catalogo.
Post condizioni	L'elemento è inserito nel catalogo, oppure viene rifiutato, perché già presente.
Sequenza eventi principali	<ol style="list-style-type: none"> 1) Controllo dei privilegi dell'utente che fa richiesta. 2) Richiesta di inserimento dati dell'elemento: nome opera, autore, tipo di opera o stato della stessa. 3) Inseriti i dati, controlla se l'opera è già presente nel catalogo. 4) Se è possibile, inserisce in archivio l'opera. 5) Rilascia un ID univoco relativo l'elemento inserito in catalogo.
Sequenza eventi alternativa	<p>Da 1), se l'utente non è loggato con un utente con i necessari requisiti, non si può proseguire.</p> <p>Da 3), se l'opera è già presente in catalogo si torna al punto 3).</p>

2) Visualizza

Attori	Utente "Catalogatore"
Pre condizioni	L'utente deve essere correttamente loggato.
Post condizioni	Nel caso si sia scelto di eliminare o modificare un'opera si passa al relativo caso d'uso.
Sequenza eventi principali	<ol style="list-style-type: none"> 1) Controllo dei privilegi dell'utente che fa richiesta. 2) Richiesta di inserimento filtri per le opere da visualizzare, autore, stato oppure tipo di opera o direttamente l'ID 3) Inseriti i dati, fornisce l'elenco delle opere filtrate. 4) OPZIONALE: Mostra ulteriori informazioni su una determinata opera selezionata. 5) Sarà possibile eliminare o modificare le opere, e modificare nuovamente i filtri della ricerca.
Sequenza eventi alternativa	Da 1), se l'utente non è loggato con un utente con i necessari requisiti, non si può proseguire.

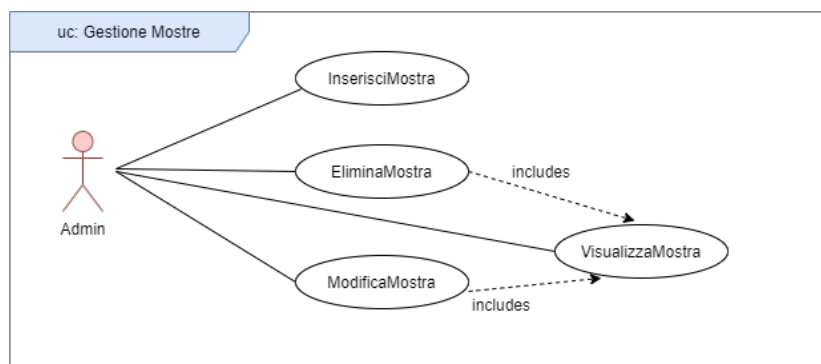
3) Modifica

Attori	Utente "Catalogo"
Pre condizioni	L'opera deve essere presente nell'elenco.
Post condizioni	I dati dell'opera sono modificati nel relativo database, oppure viene rigettata se non è stato possibile modificarla. L'ID non può essere modificato.
Sequenza eventi principali	1) Richiesta di inserimento del codice dell'opera da modificare 2) Inserito l'identificativo, controlla che l'opera sia nel catalogo 3) Se è possibile, modifica l'opera nel catalogo.
Sequenza eventi alternativa	Da 2), se non è possibile modificare l'opera, dopo la notifica dell'impossibilità, si torna al punto 1), in modo da reinserire i dati.

4) Elimina

Attori	Utente "Catalogo"
Pre condizioni	L'opera deve essere presente nell'elenco.
Post condizioni	L'opera è eliminata dall'elenco.
Sequenza eventi principali	1. L'opera selezionata viene eliminata dall'elenco
Sequenza eventi alternativa	/

6.7 Casi d'uso: Gestione Mostre



1) Inserisci

Attori	Utente "Amministratore"
Pre condizioni	L'utente deve essere correttamente loggato, le opere impiegate nell'evento non possono esserlo in altre nella stessa data.
Post condizioni	La mostra è inserita nel relativo database e viene fornito un ID per il riconoscimento della mostra. Se non è stato possibile aggiungerla, viene rigettata.
Sequenza eventi principali	<ol style="list-style-type: none"> 1) Controllo dei privilegi dell'utente che fa richiesta. 2) Richiesta di inserimento dati per la mostra: data, nome e opere impiegate. Sarà possibile salvare il tipo di evento e caricarlo in seguito. 3) Inseriti i dati, controlla la possibilità che la mostra abbia luogo. 4) Se è possibile, inserisce in archivio l'evento. 5) Rilascia un ID univoco relativo la mostra.
Sequenza eventi alternativa	<p>Da 1), se l'utente non è loggato con un utente con i necessari requisiti, non si può proseguire.</p> <p>Da 3), se non è possibile inserire la mostra, dopo la notifica dell'impossibilità della memorizzazione dell'evento, si torna al punto 2), in modo da reinserire i dati.</p>

2) Visualizza

Attori	Utente "Amministratore"
Pre condizioni	L'utente deve essere correttamente loggato.
Post condizioni	Nel caso si sia scelto di eliminare o modificare una mostra si passa al relativo caso d'uso.
Sequenza eventi principali	<ol style="list-style-type: none"> 1) Controllo dei privilegi dell'utente che fa richiesta. 2) Richiesta di inserimento filtri per le mostre da visualizzare. 3) Inseriti i dati, fornisce l'elenco delle mostre filtrate. 4) OPZIONALE: Mostra ulteriori informazioni su una determinata mostra selezionata 5) Sarà possibile eliminare o modificare gli eventi, e modificare nuovamente i filtri della ricerca
Sequenza eventi alternativa	Da 1), se l'utente non è loggato con un utente con i necessari requisiti, non si può proseguire.

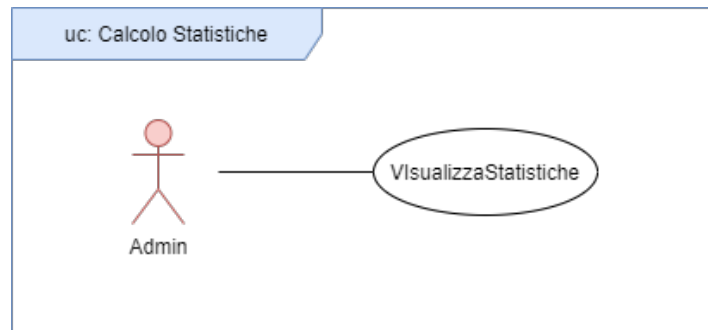
3) Modifica

Attori	Utente "Amministratore"
Pre condizioni	La mostra deve essere presente nell'elenco, il nuovo evento modificato non deve impiegare opere già utilizzate in altre mostre nelle date in cui andrà svolto.
Post condizioni	I dati della mostra sono modificati nel relativo database, oppure viene rigettata se non è stato possibile modificarla. L'ID non può essere modificato.
Sequenza eventi principali	<ol style="list-style-type: none"> 1) Richiesta di inserimento dati per la prenotazione modificata: data, nome ed opere impiegate. 2) Inseriti i dati, controlla la possibilità che l'evento abbia luogo. 3) Se è possibile, inserisce in archivio la mostra.
Sequenza eventi alternativa	Da 2), se non è possibile inserire la mostra, dopo la notifica dell'impossibilità della modifica della prenotazione, si torna al punto 1), in modo da reinserire i dati.

4) Elimina

Attori	Utente "Amministratore"
Pre condizioni	La mostra deve essere presente nell'elenco.
Post condizioni	La mostra è eliminata dall'elenco.
Sequenza eventi principali	1. La mostra selezionata viene eliminata dall'elenco
Sequenza eventi alternativa	/

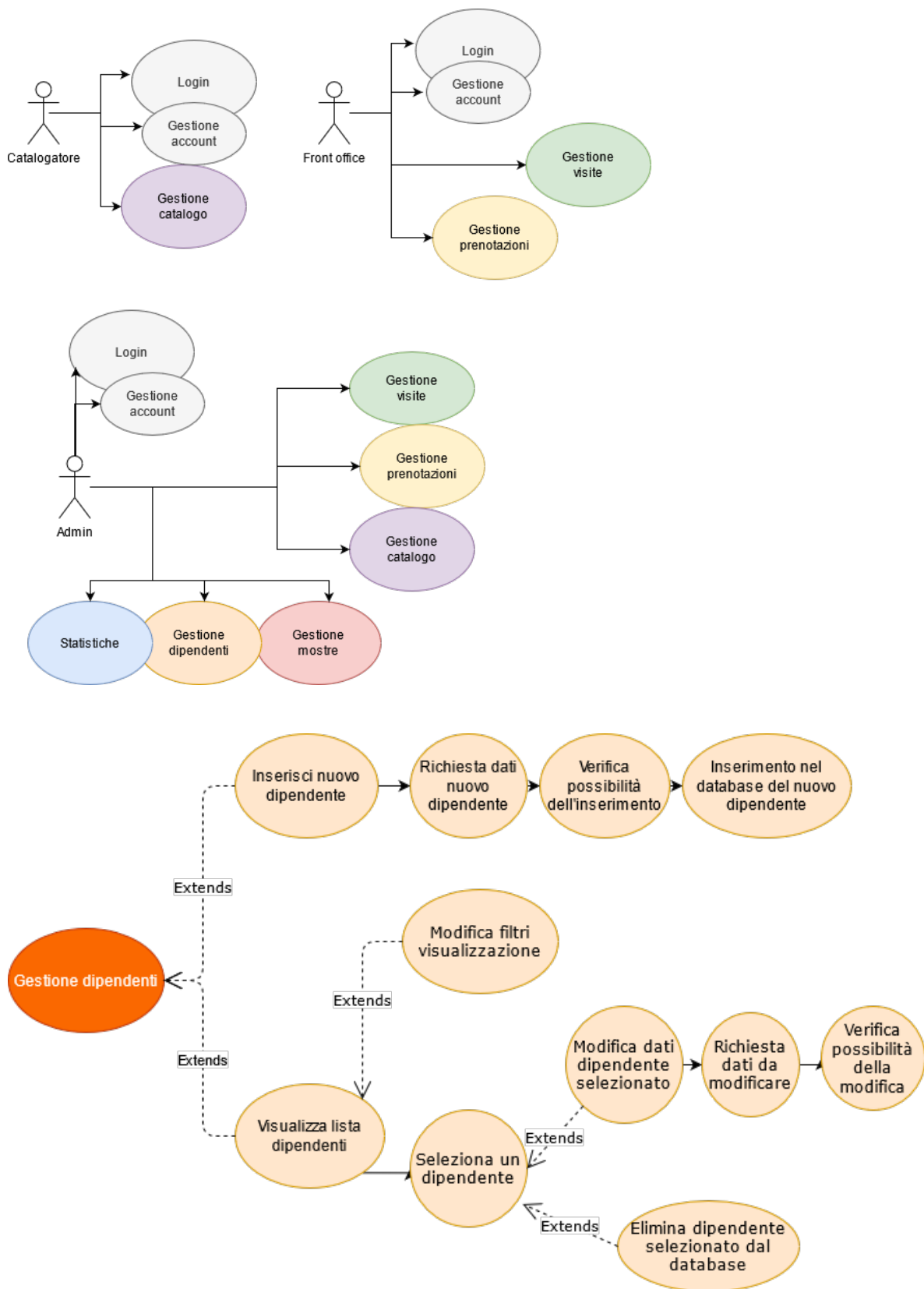
6.8 Casi d'uso: Calcolo Statistiche

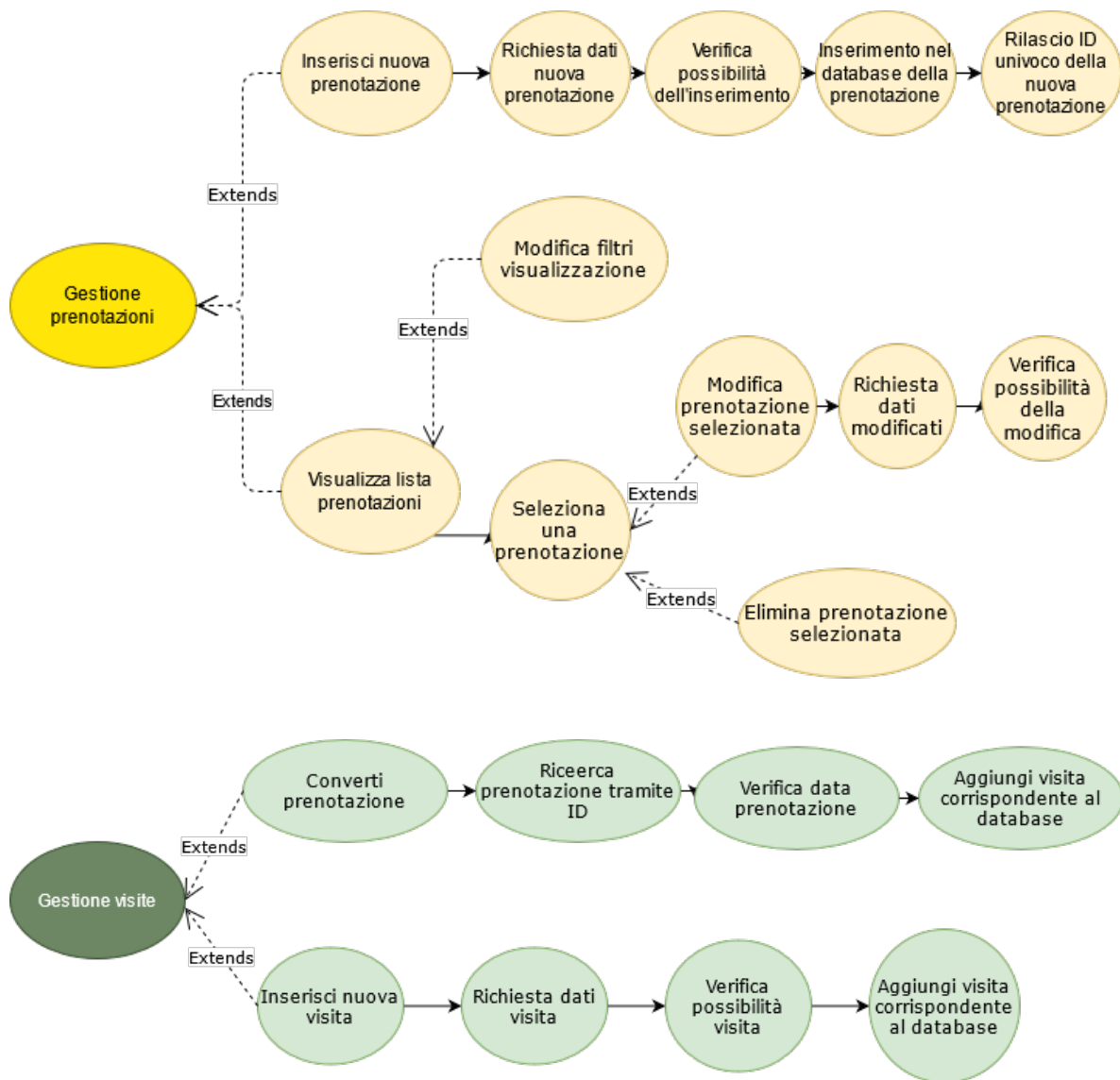


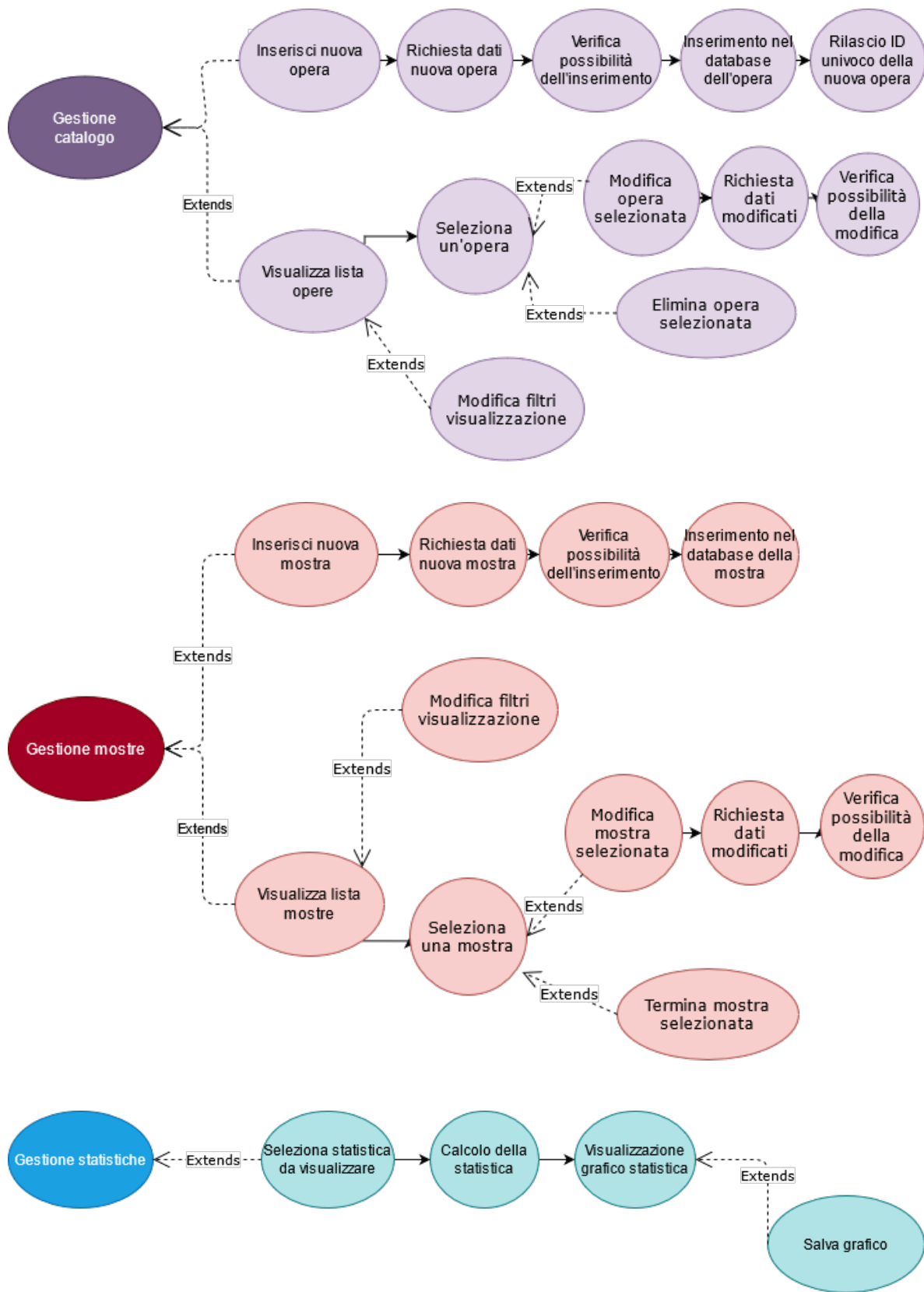
1. Visualizza

Attori	Utente "Amministratore"
Pre condizioni	L'utente deve essere correttamente loggato.
Post condizioni	/
Sequenza eventi principali	<ol style="list-style-type: none"> 1) Controllo dei privilegi dell'utente che fa richiesta. 2) Selezionare il tipo di statistica da visualizzare e i filtri. 3) Attingendo ai dati, si calcolano le statistiche e le si visualizzano.
Sequenza eventi alternativa	Da 1), se l'utente non è loggato con un utente con i necessari requisiti, non può proseguire.

6.9 Diagrammi dei casi d'uso complessivi







6.10 Matrice di mapping

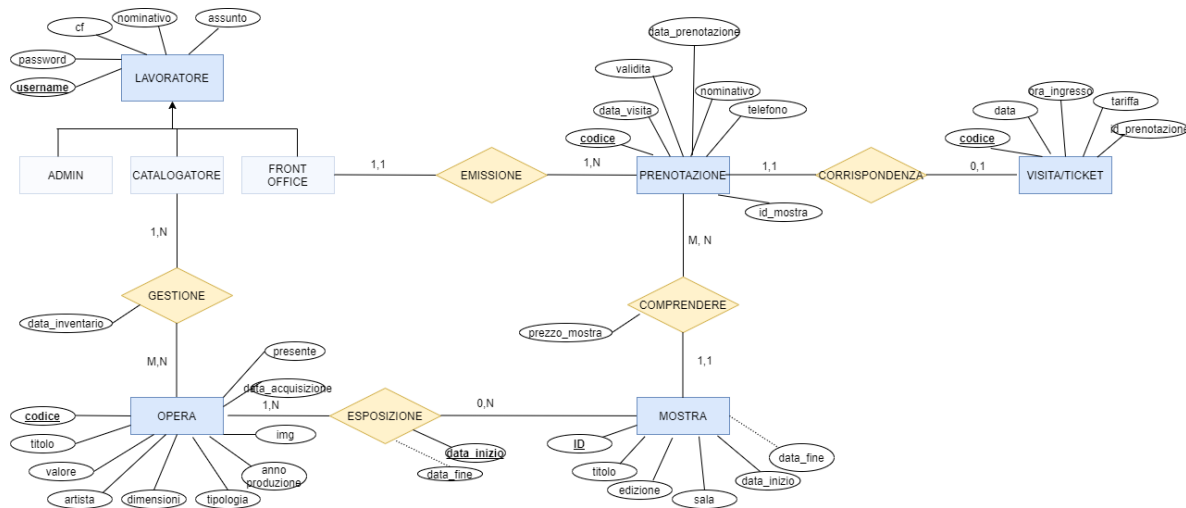
Di seguito è riportata la matrice di mapping tra requisiti funzionali e casi d'uso.

	RF1.1 Inserimento Account	RF1.2 Eliminazione Account	RF1.3 Modifica Account	RF1.4 Visualizzazione Account	RF2.1 Inserimento Visita	RF2.2 Accogliete Prenotazione	RF3.1 Inserimento Prenotazione	RF3.2 Eliminazione Prenotazione	RF3.3 Modifica Prenotazione	RF3.4 Visualizzazione Prenotazione	RF4.1 Inserimento Opera	RF4.2 Eliminazione Opera	RF4.3 Modifica Opera	RF4.4 Visualizzazione Opera	RF5.1 Inserimento Mostra	RF5.2 Eliminazione Mostra	RF5.3 Modifica Mostra	RF5.4 Visualizzazione Mostra	RF6.1 Visualizzazione Statistiche
InserisciAccount	•																		
EliminaAccount		•		•															
ModificaAccount			•	•															
VisualizzaAccount				•															
InserisciPrenotazione							•			•									
EliminaPrenotazione								•		•									
ModificaPrenotazione									•	•									
VisualizzaPrenotazione										•									
InserisciOpera											•								
EliminaOpera												•		•					
ModificaOpera													•	•					
VisualizzaOpera														•					
InserisciMostra															•			•	
EliminaMostra																•		•	
ModificaMostra																	•	•	
VisualizzaMostra																			•
VisualizzaStatistiche																			
InserisciVisita					•														
AccoglierePrenotazione					•	•													

6.11 Diagramma Entità-Relazioni

6.11.1 Schema ER

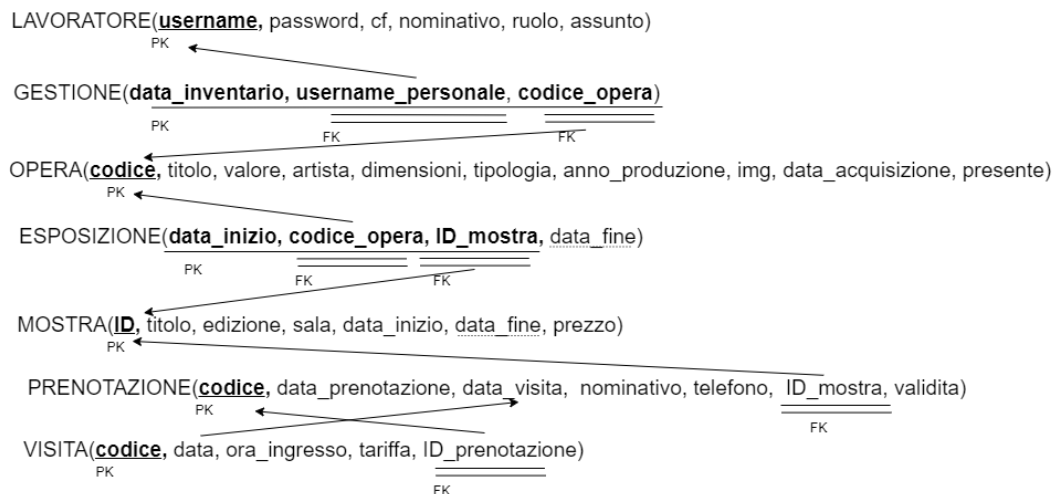
Lo schema ER mostra le entità coinvolte nel sistema e le relazioni tra esse.



Ipotesi:

- Per ogni mostra è previsto un biglietto a sé e una prenotazione a sé
- Non si tiene traccia dei dati personali dei visitatori
- La prenotazione è gratuita e obbligatoria; se il cliente non ha prenotato, la prenotazione viene creata contestualmente al momento dell'ingresso
- Essendo le statistiche calcolate su dati relativi a visite, prenotazione e mostre salvati nel database, esse non compaiono nello schema
- La specializzazione di Lavoratore viene implementata aggiungendo un campo ruolo alla tabella Lavoratore, che può assumere esclusivamente i valori 'admin', 'catalogatore' e 'front_office'

6.11.2 Schema logico



6.11.3 Implementazione del database

Tabella **Lavoratore**

```
CREATE TABLE IF NOT EXISTS lavoratore (  
    username TEXT PRIMARY KEY ,  
    password TEXT NOT NULL,  
    cf TEXT NOT NULL CHECK (length(CF) == 16) ,  
    nominativo TEXT NOT NULL,  
    ruolo TEXT NOT NULL  
    CHECK (ruolo IN ('admin', 'catalogatore', 'front_office' )),  
    assunto INTEGER NOT NULL DEFAULT 1  
)WITHOUT ROWID;
```

Tabella **Opera**

```
CREATE TABLE IF NOT EXISTS opera (  
    codice INTEGER PRIMARY KEY,  
    titolo TEXT NOT NULL,  
    valore REAL NOT NULL,  
    artista TEXT NOT NULL,  
    dimensioni TEXT NOT NULL,  
    tipologia TEXT NOT NULL,  
    anno_produzione TEXT NOT NULL,  
    img TEXT NOT NULL,  
    data_acquisizione TEXT NOT NULL,  
    presente INTEGER NOT NULL DEFAULT 1  
    CHECK(presente IN (0, 1))  
);
```

Tabella **Gestione**

```
CREATE TABLE IF NOT EXISTS gestione (  
    data_inventario TEXT NOT NULL,  
    username_lavoratore TEXT NOT NULL,  
    codice_opera INTEGER NOT NULL,  
    PRIMARY KEY(data_inventario, username_lavoratore, codice_opera)  
    FOREIGN KEY (username_lavoratore) REFERENCES lavoratore(username)  
    ON UPDATE CASCADE ON DELETE NO ACTION,  
    FOREIGN KEY (codice_opera) REFERENCES opera(codice)  
    ON UPDATE CASCADE ON DELETE NO ACTION  
) WITHOUT ROWID;
```

Tabella **Mostra**

```
CREATE TABLE IF NOT EXISTS mostra (  
    ID INTEGER PRIMARY KEY,  
    titolo TEXT NOT NULL,  
    edizione INTEGER NOT NULL CHECK(edizione>0),  
    sala TEXT NOT NULL,  
    data_inizio TEXT NOT NULL,  
    data_fine TEXT,  
    prezzo REAL CHECK(prezzo>=0)  
);
```

Tabella **Esposizione**

```
CREATE TABLE IF NOT EXISTS esposizione (  
    data_inizio TEXT,  
    codice_opera INTEGER, ID_mostra INTEGER,  
    data_fine TEXT,  
    PRIMARY KEY(data_inizio, codice_opera, ID_mostra )  
    FOREIGN KEY (codice_opera) REFERENCES opera(codice)  
    ON UPDATE CASCADE ON DELETE NO ACTION,  
    FOREIGN KEY (ID_mostra) REFERENCES mostra(ID)  
    ON UPDATE CASCADE ON DELETE NO ACTION  
);
```

Tabella **Prenotazione**

```
CREATE TABLE IF NOT EXISTS prenotazione (  
    codice INTEGER PRIMARY KEY,  
    data_prenotazione TEXT NOT NULL,  
    data_visita TEXT NOT NULL,  
    nominativo TEXT NOT NULL,  
    telefono TEXT NOT NULL CHECK(length(telefono)>9),  
    ID_mostra INTEGER NOT NULL,  
    validita INTEGER NOT NULL DEFAULT 1 CHECK(validita IN (0, 1)),  
    FOREIGN KEY (ID_mostra) REFERENCES mostra(ID)  
    ON UPDATE CASCADE ON DELETE NO ACTION  
);
```

Tabella **Visita**

```
CREATE TABLE IF NOT EXISTS visita (  
    codice INTEGER PRIMARY KEY,  
    data TEXT NOT NULL,  
    ora_ingresso TEXT NOT NULL,  
    tariffa TEXT NOT NULL CHECK (tariffa IN ('gratuito', 'ridotto', 'intero')),  
    codice_prenotazione INTEGER NOT NULL  
);  
ALTER TABLE visita  
    ADD CONSTRAINT visita_fk_1 FOREIGN KEY (data)  
    REFERENCES prenotazione(data_visita),  
    ADD CONSTRAINT visita_fk_2 FOREIGN KEY (codice_prenotazione)  
    REFERENCES prenotazione(codice);
```

6.12 Mappa dell'architettura

Per la nostra applicazione abbiamo optato per un'architettura MVC (Model-View-Controller).

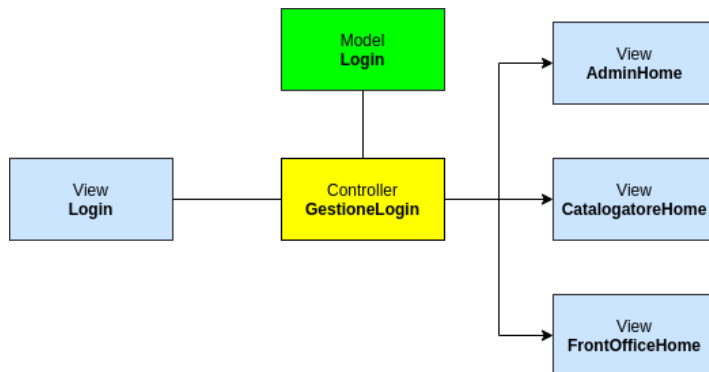


Figura 6: Autenticazione

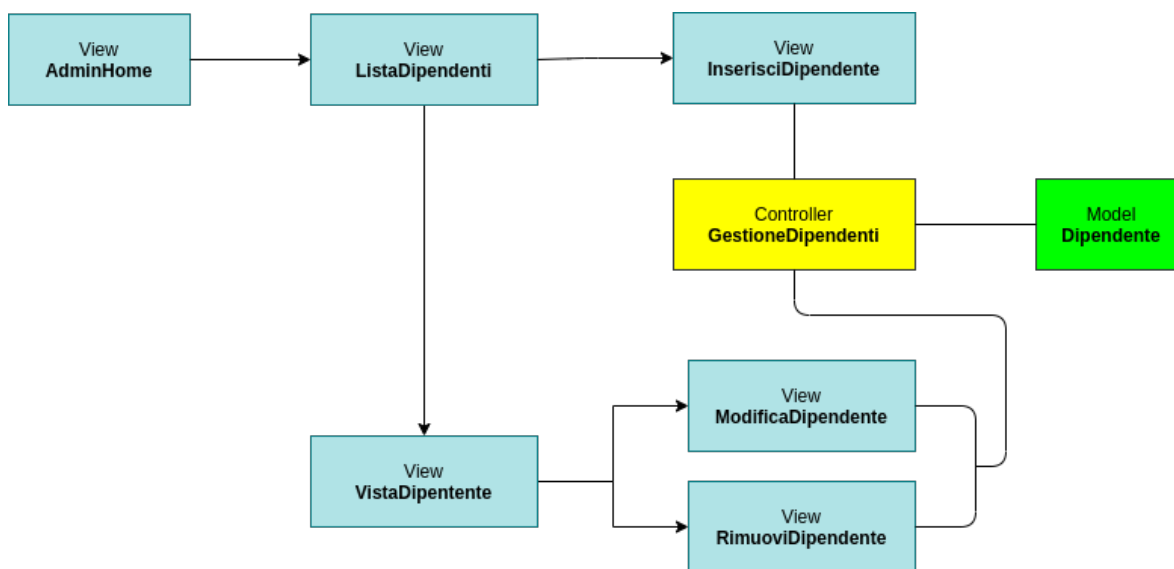


Figura 7: Dipendenti

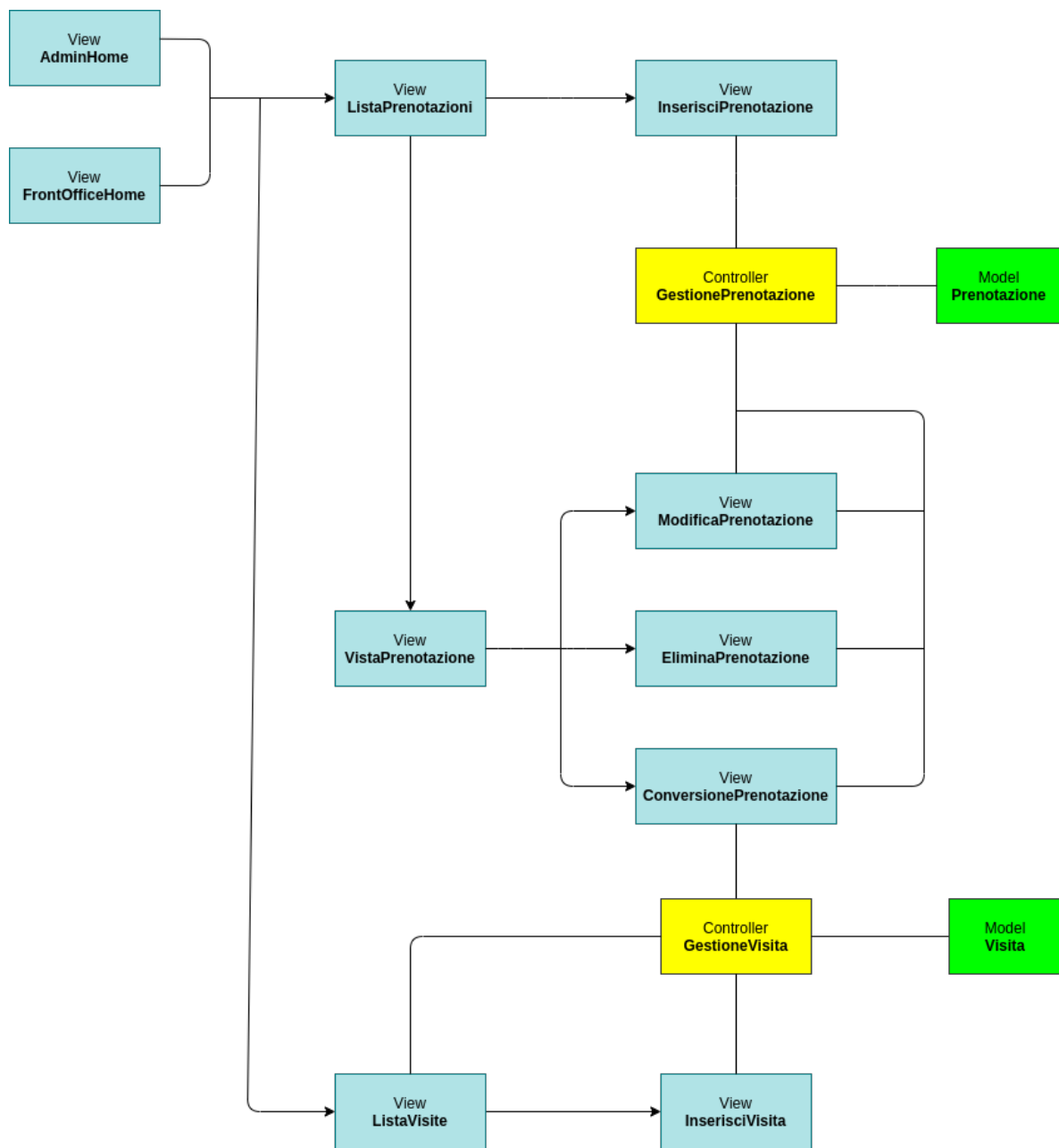


Figura 8: Prenotazioni e Visite

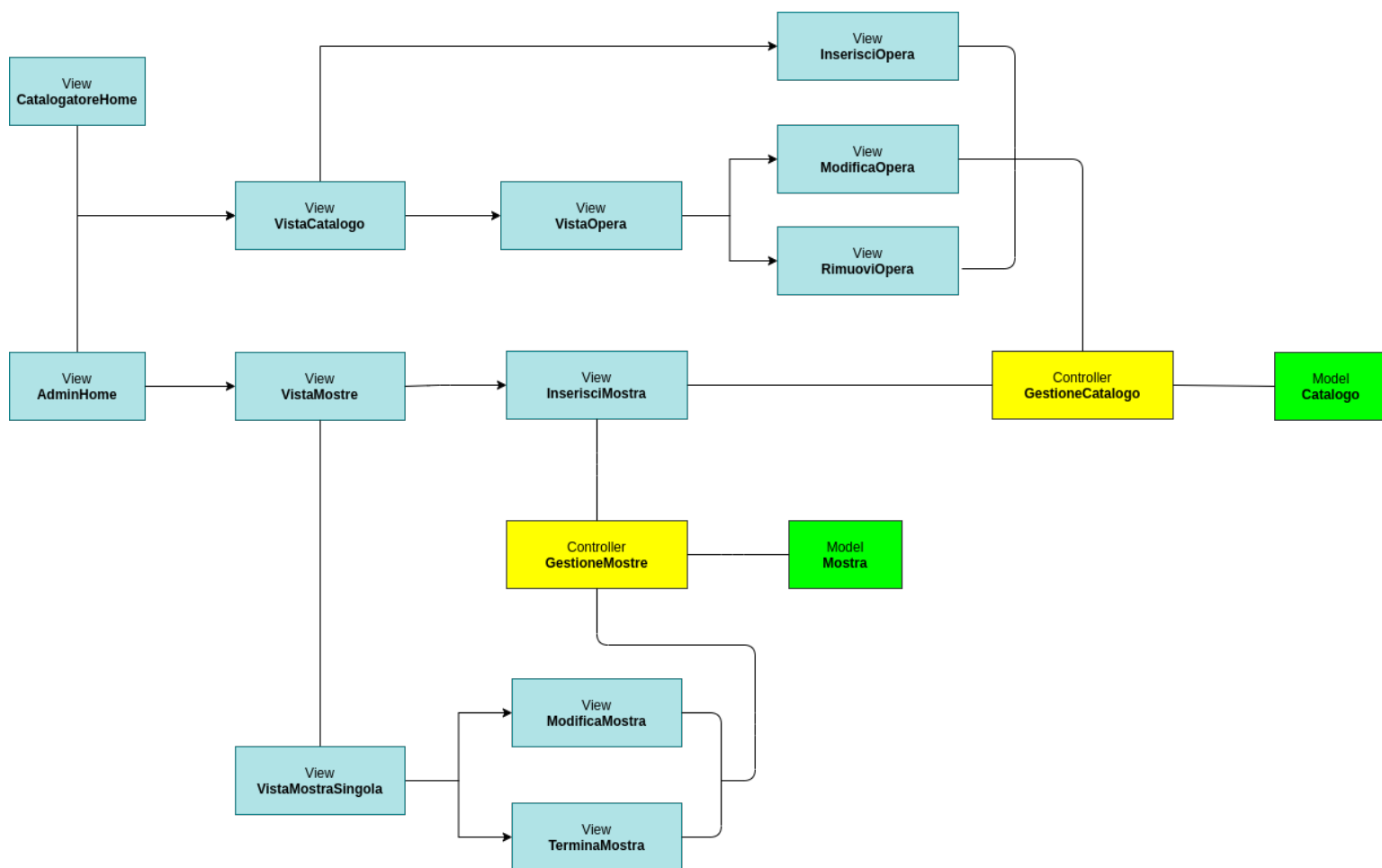


Figura 9: Opere e Mostre

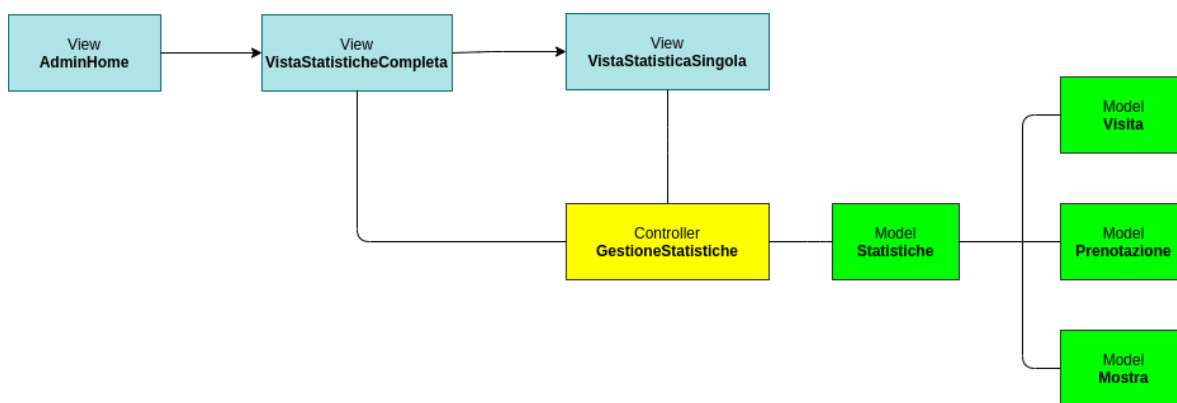


Figura 10: Statistiche

6.13 Diagramma delle classi

Il diagramma delle classi mostra le differenti classi coinvolte nel software e come esse sono collegate. Per ogni classe, sono mostrati gli attributi e i metodi principali.

Diagramma delle classi relativo alla gestione dei dipendenti.

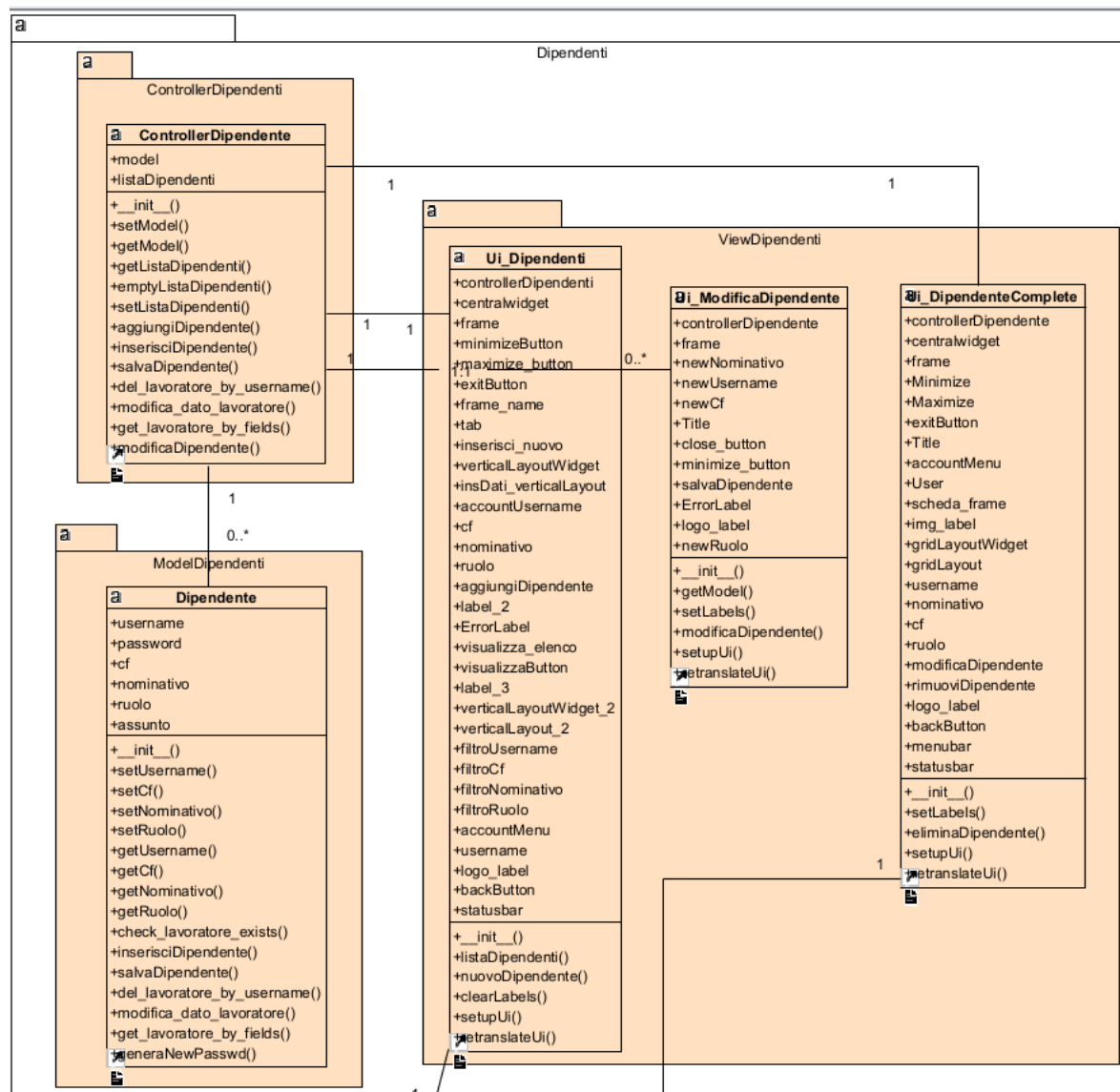


Diagramma delle classi relativo alla gestione delle opere.

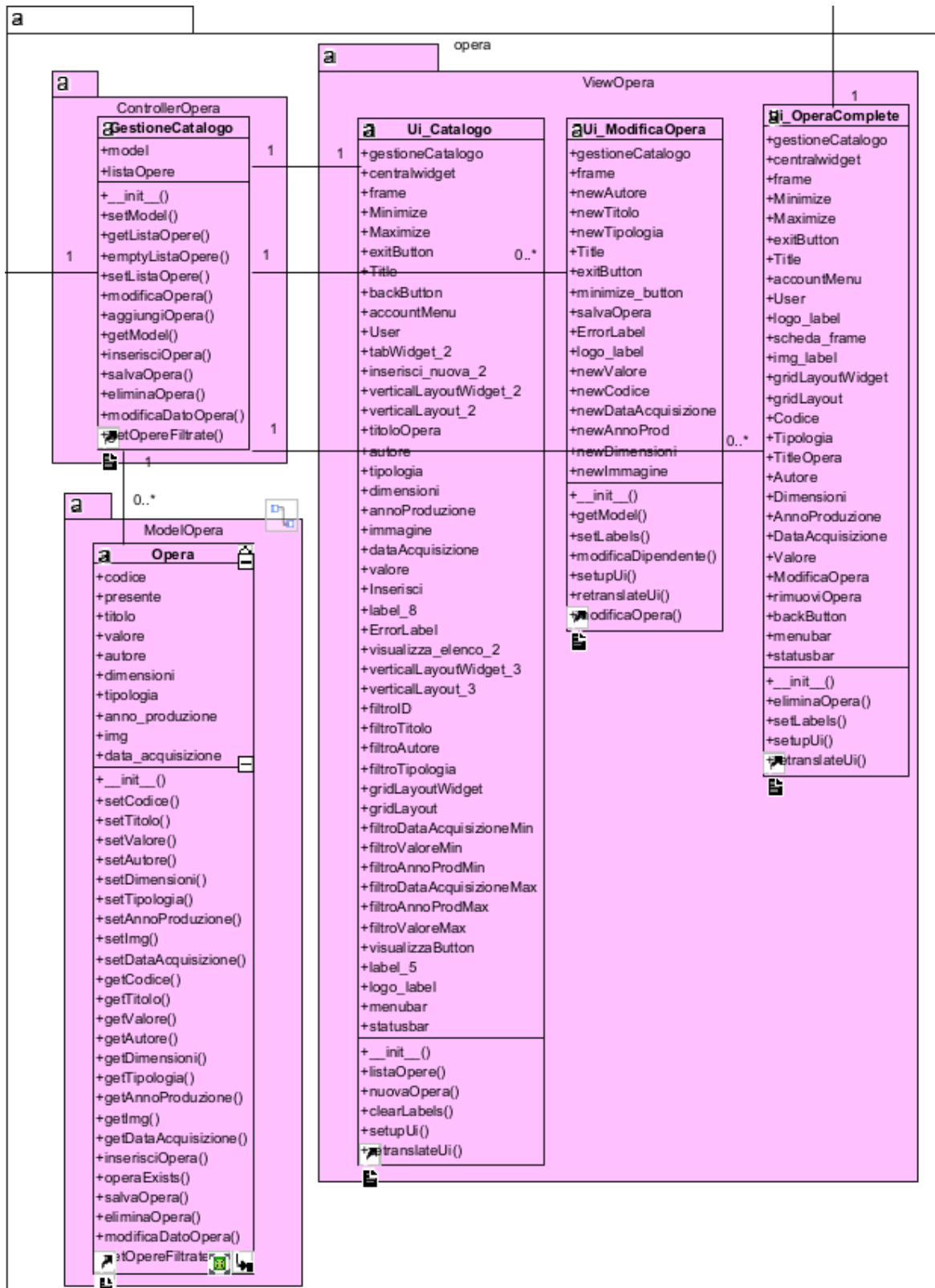


Diagramma delle classi relativo alla gestione degli eventi nel museo.

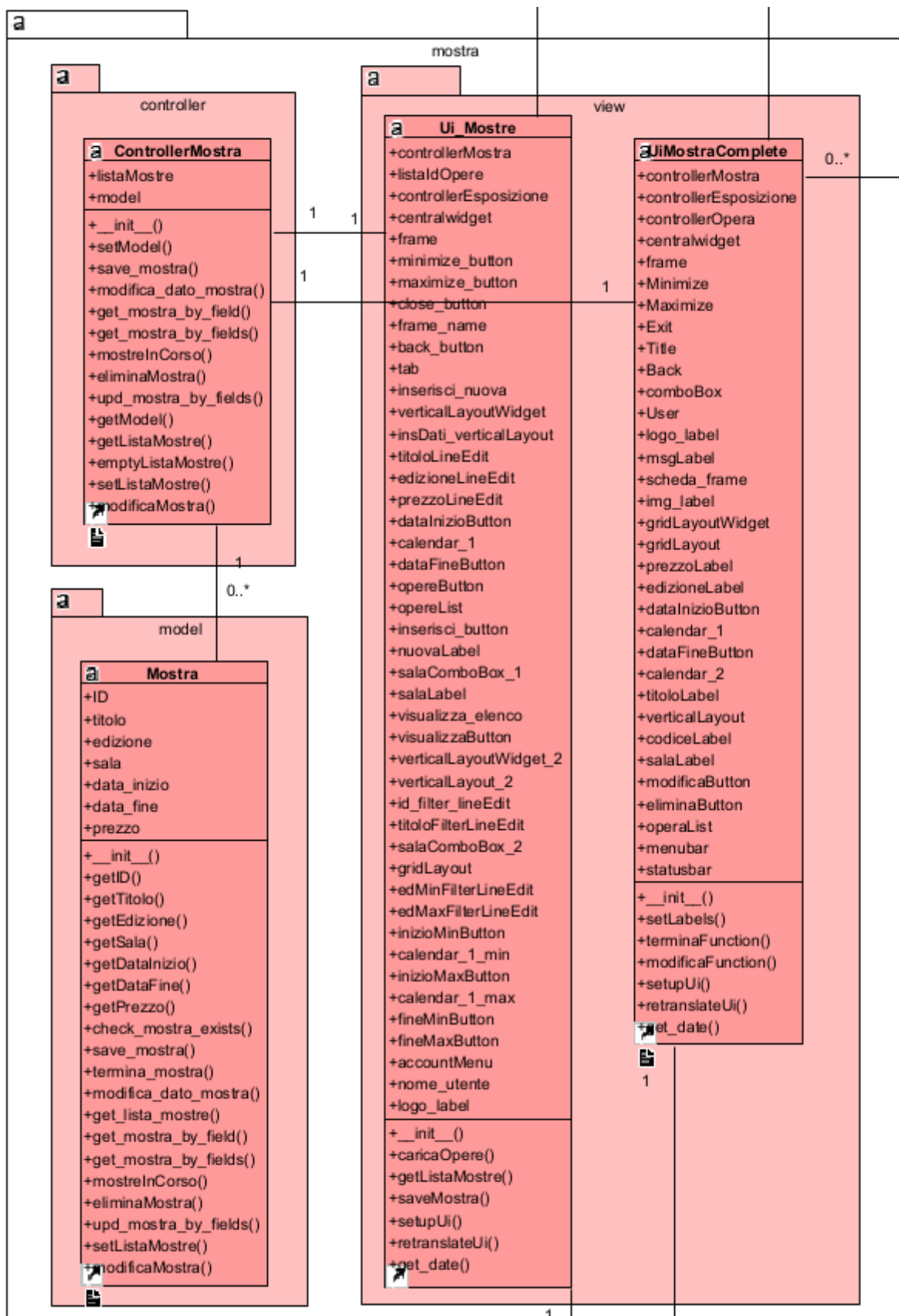


Diagramma delle classi relativo alla gestione delle prenotazioni.

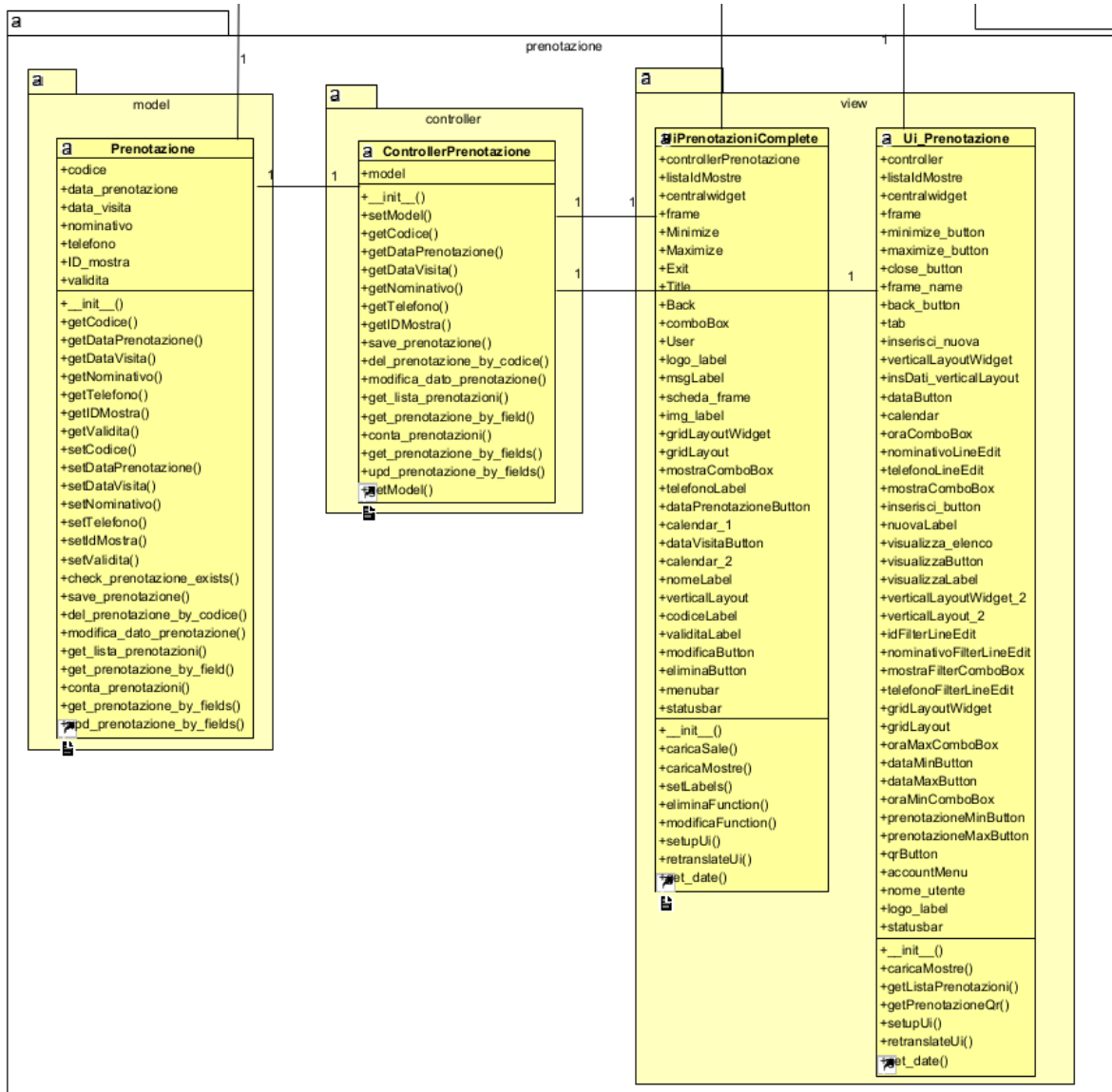


Diagramma delle classi relativo alla gestione delle visite al museo.

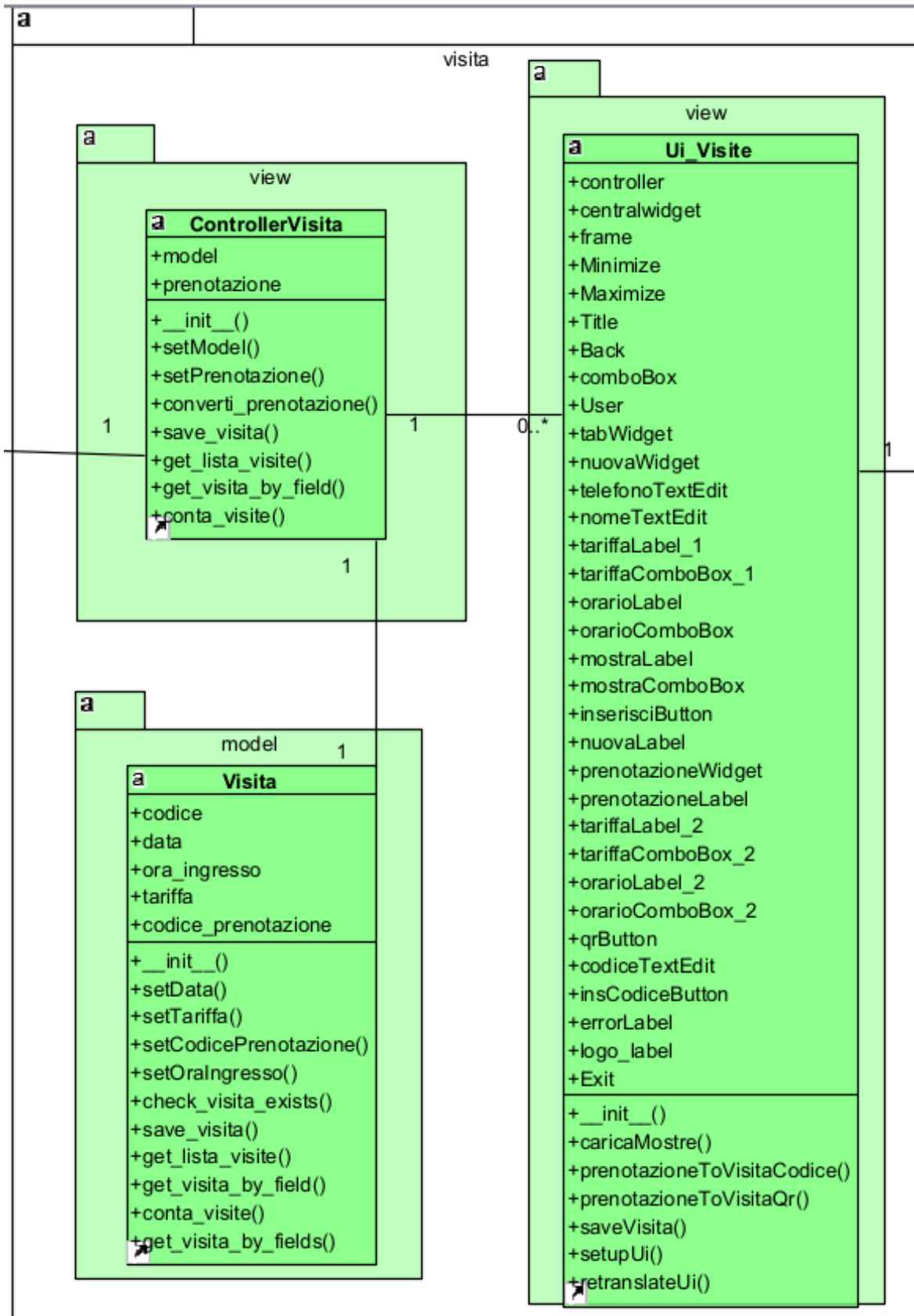


Diagramma delle classi relativo alla gestione dell'autenticazione al sistema.

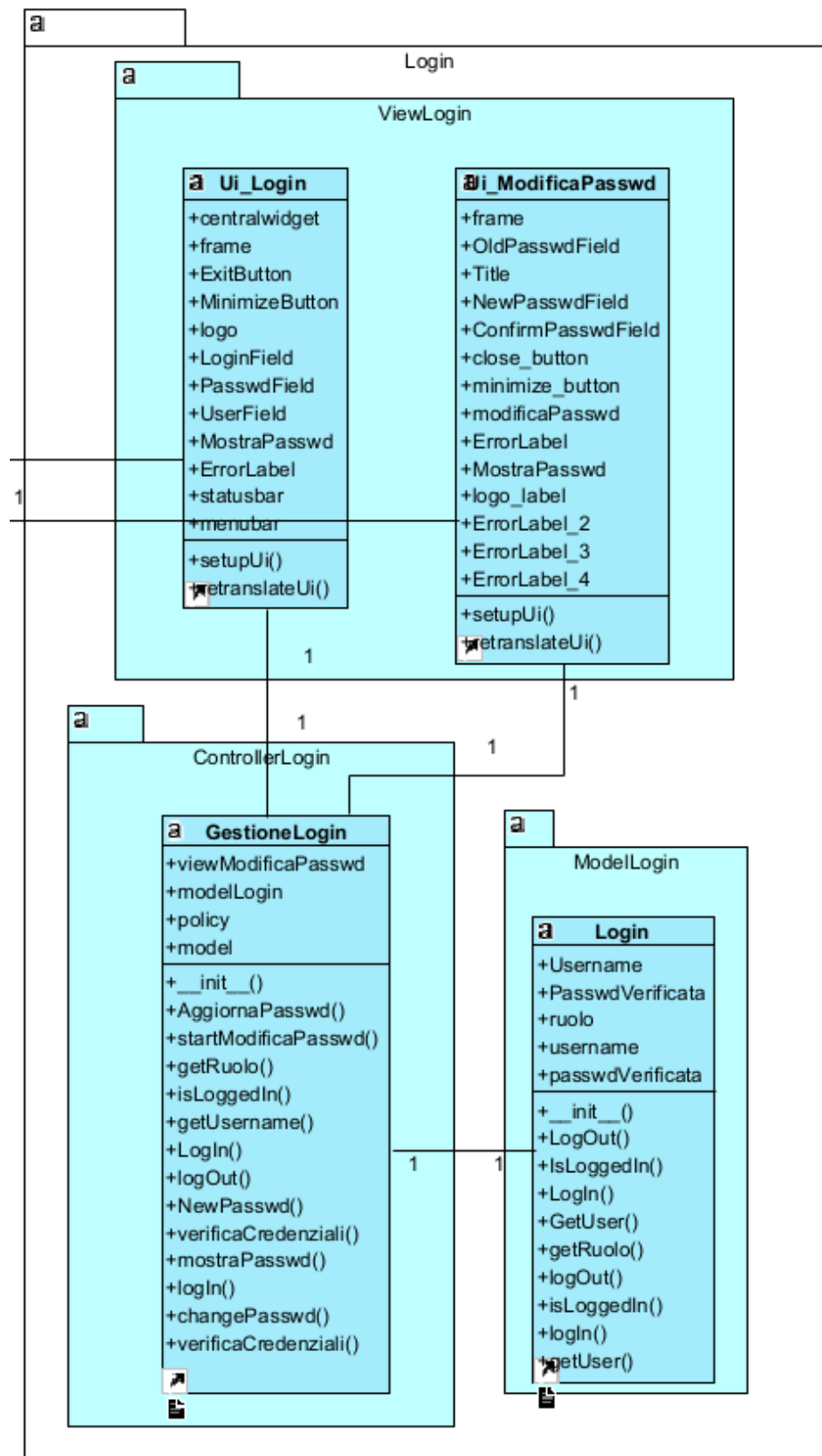
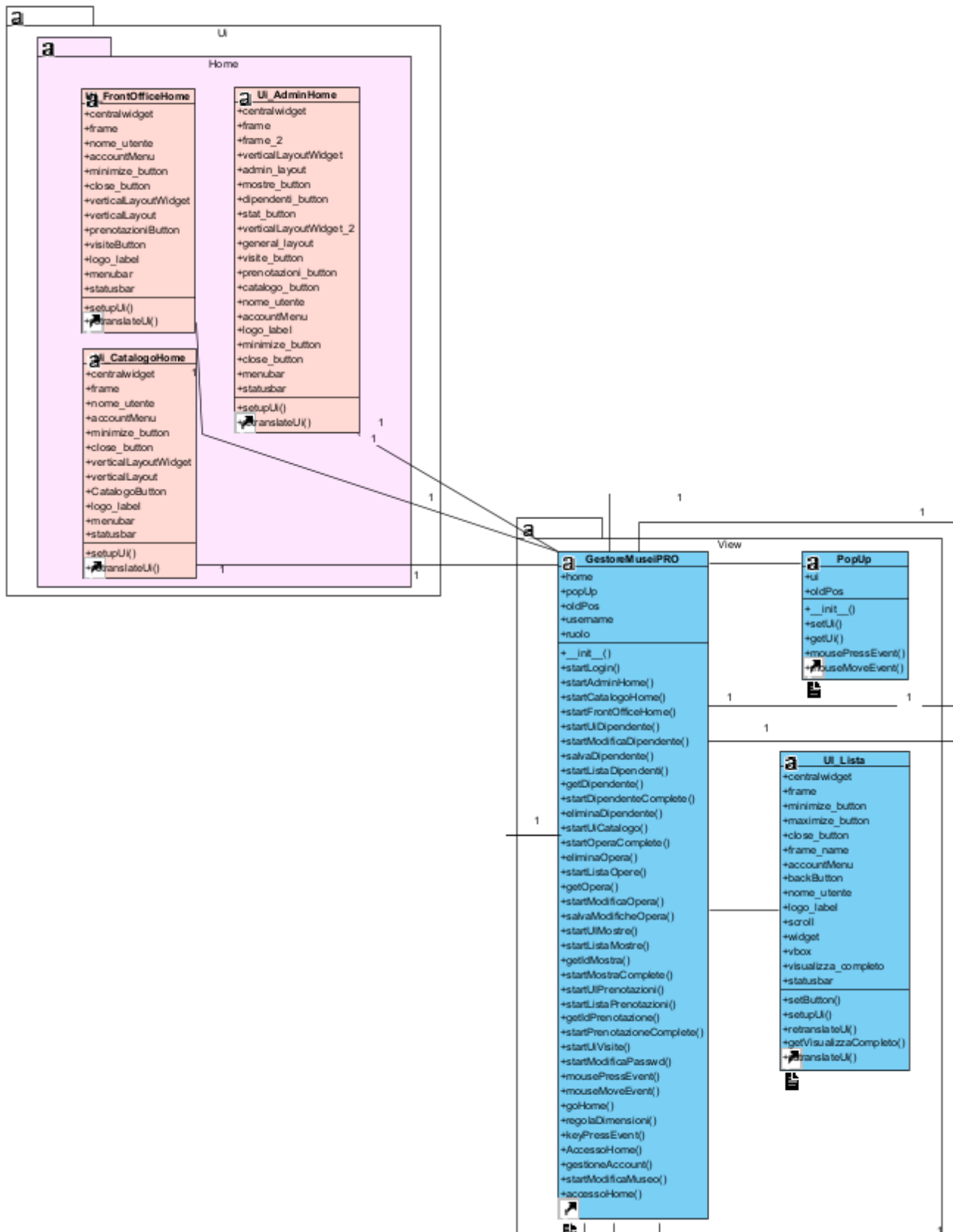
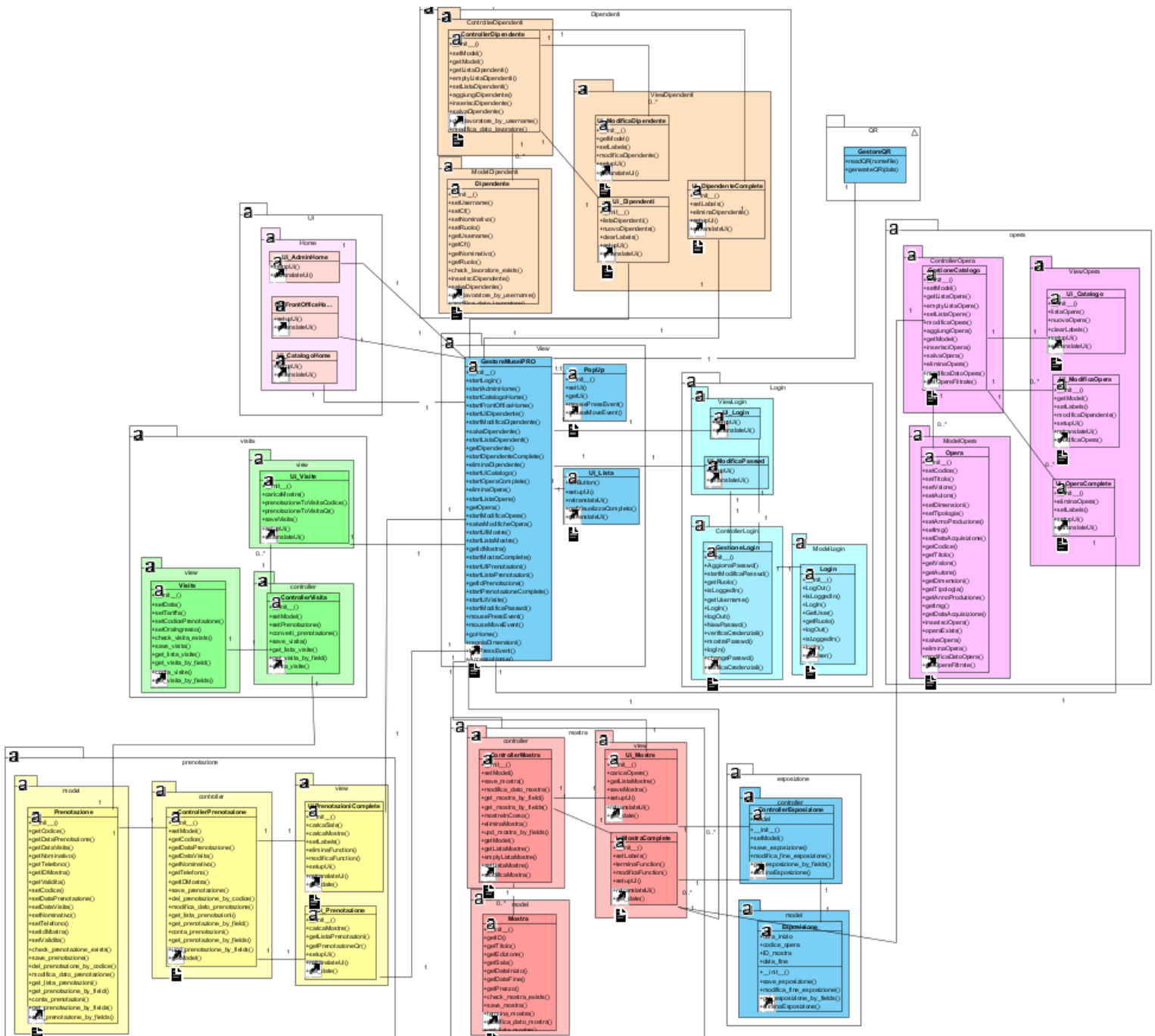


Diagramma delle classi che realizzano il funzionamento complessivo del sistema, collegando le varie interfacce.

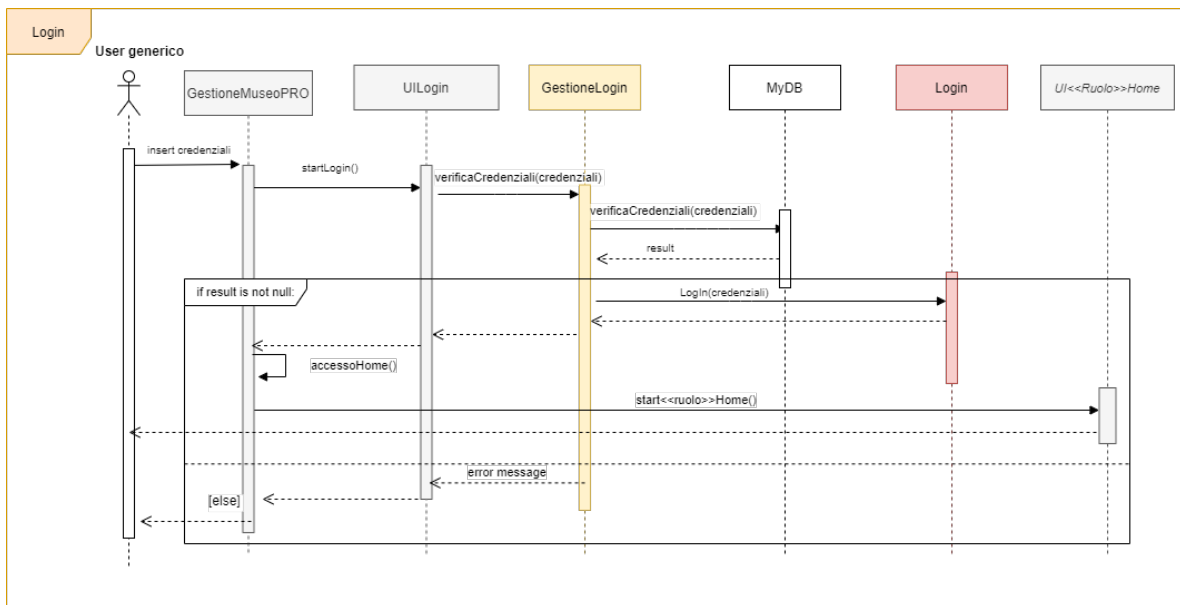


Infine il diagramma delle classi complessivo, in forma compatta.

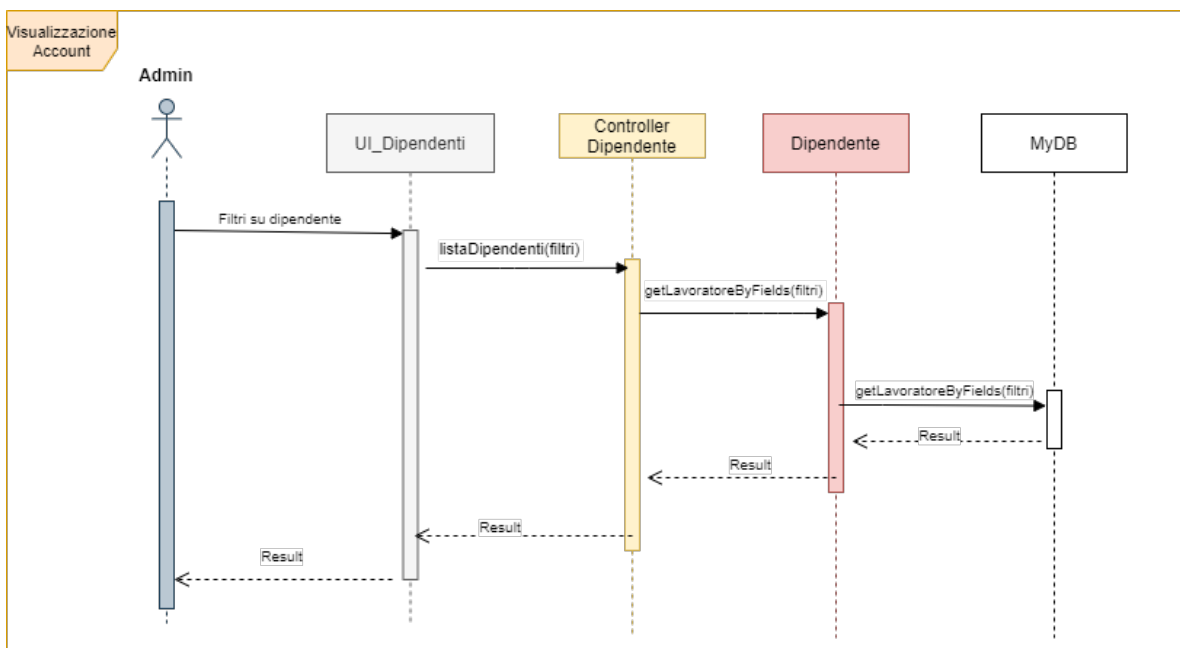


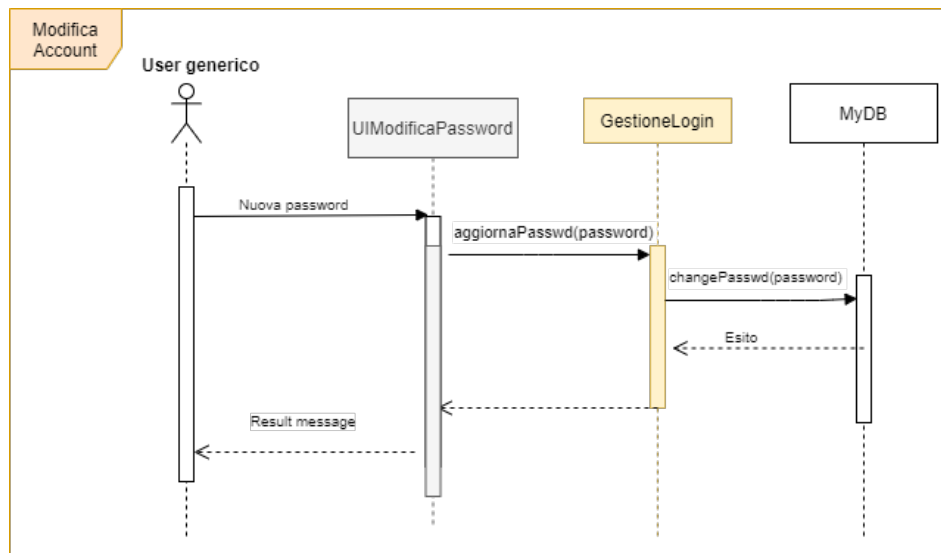
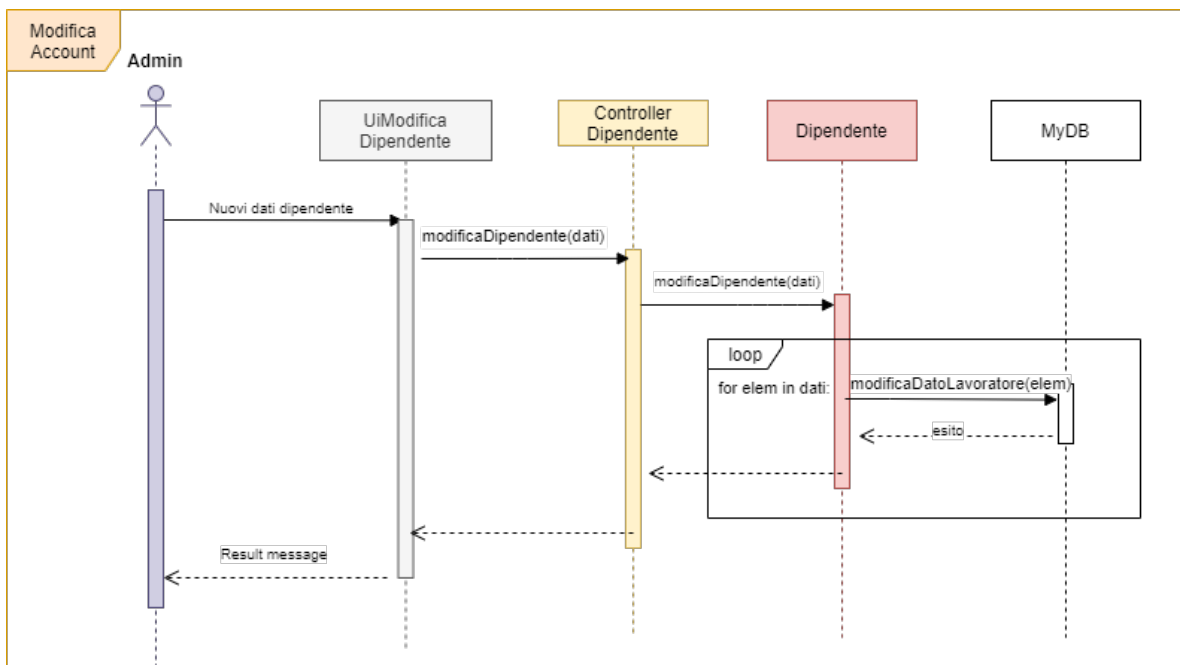
6.14 Diagrammi di sequenza

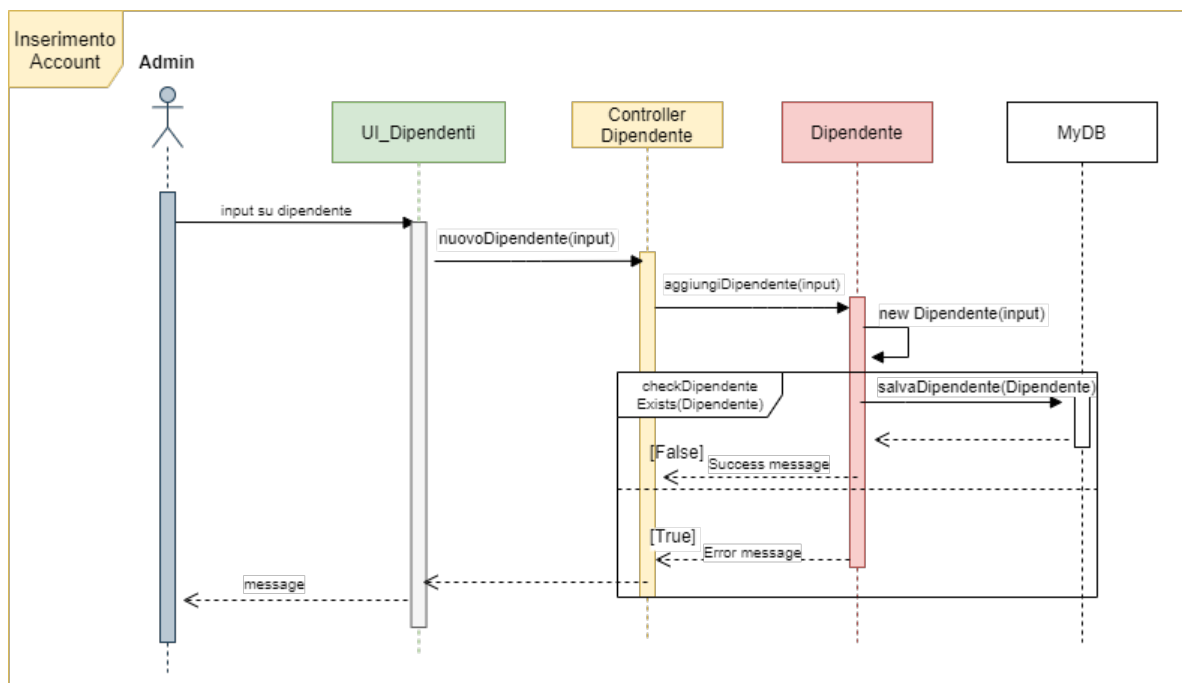
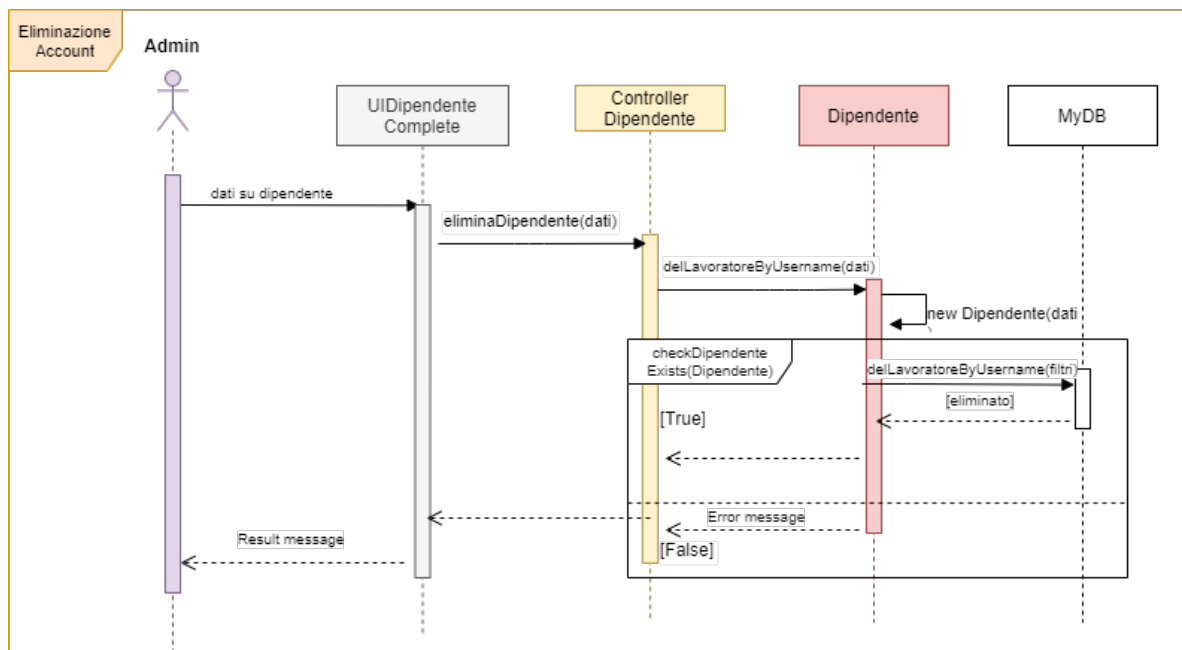
Diagramma di sequenza relativo all'autenticazione.



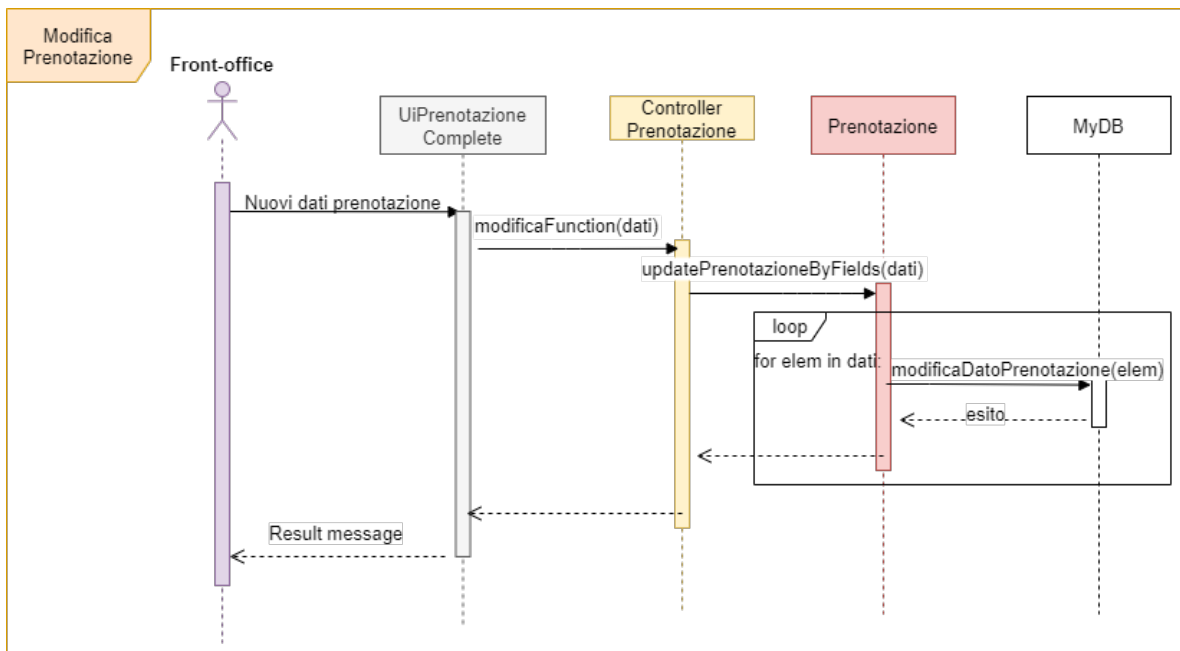
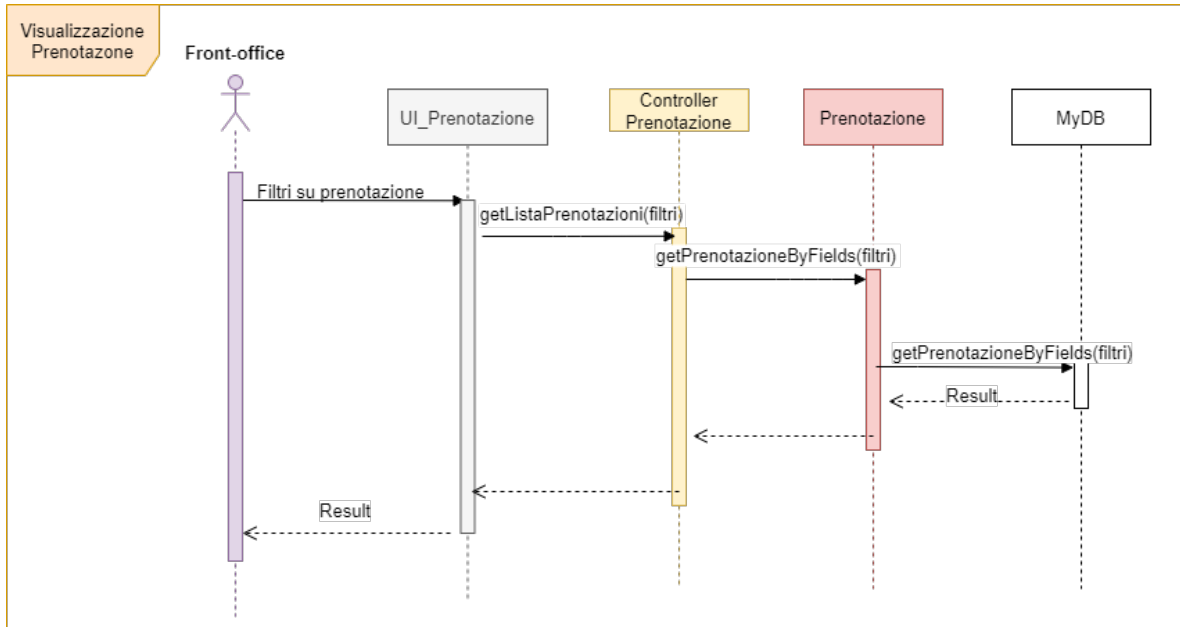
Diagrammi di sequenza relativi alla gestione degli account.

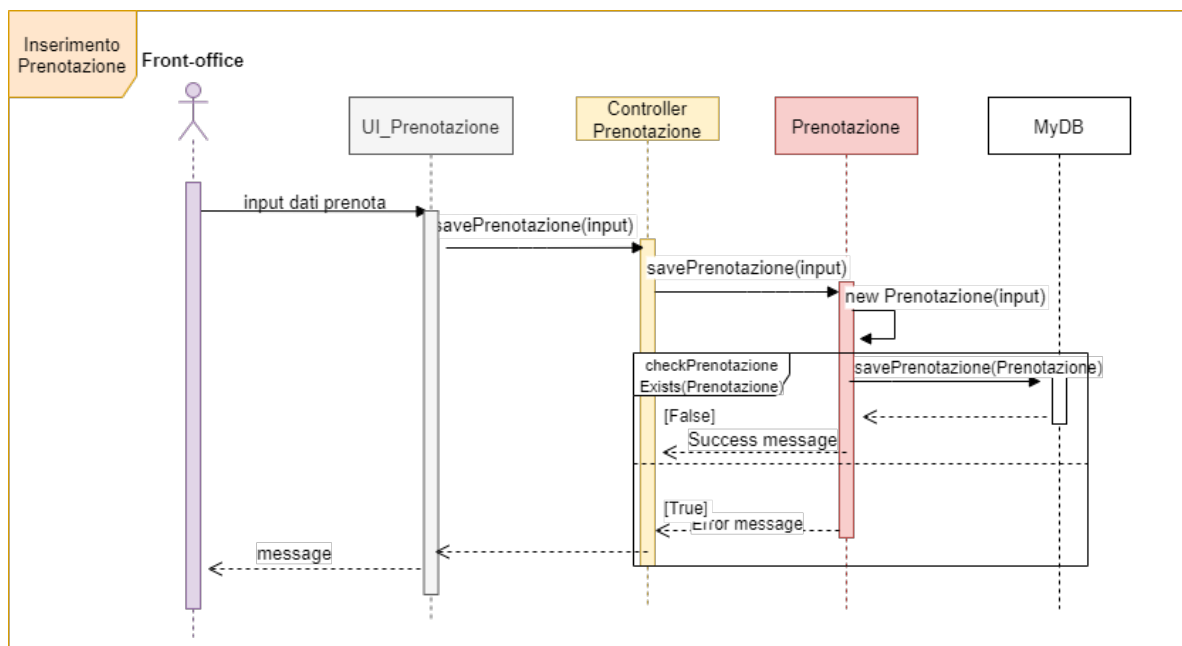
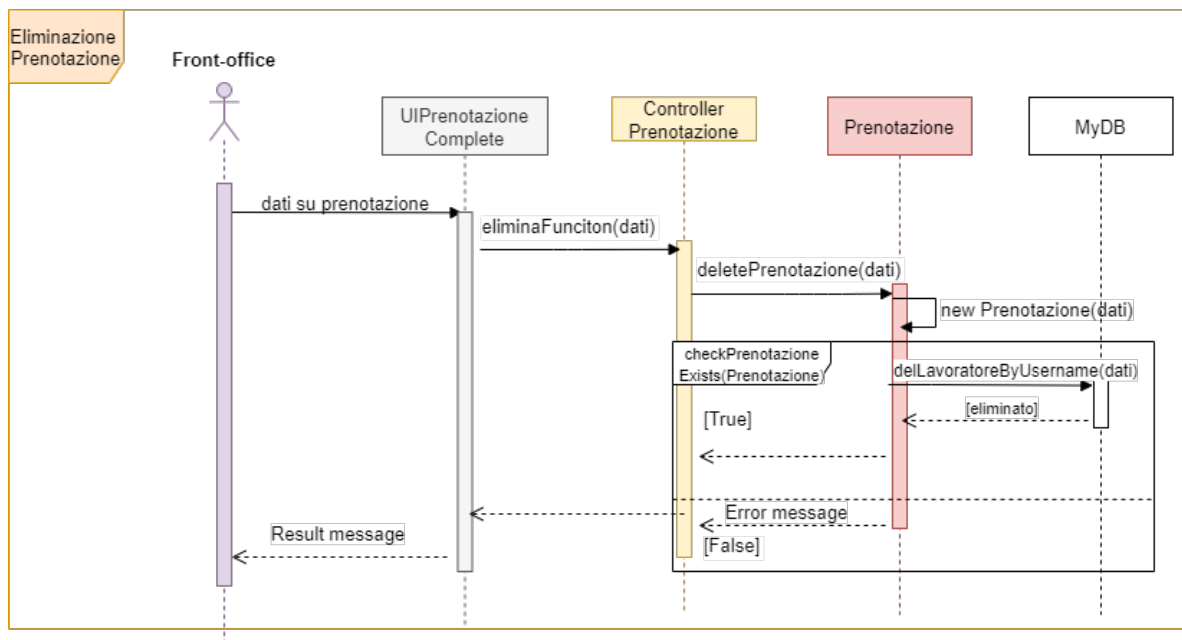






Diagrammi di sequenza relativi alla gestione delle prenotazioni.



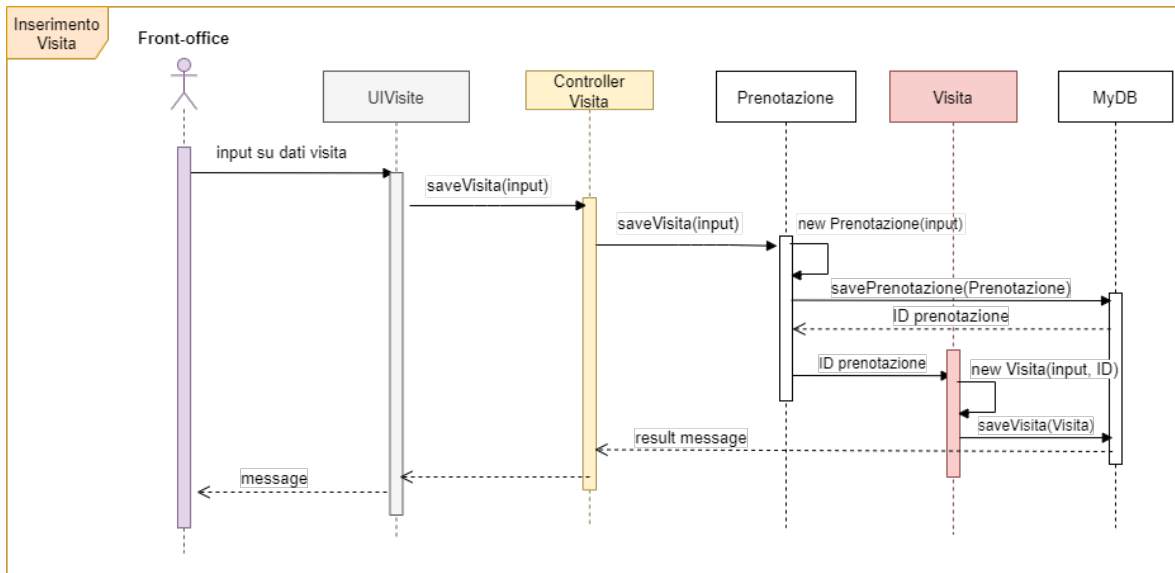


Nel nostro progetto abbiamo previsto che il software si interfacci con un sistema esterno *Gestore Qr code* per la gestione dei biglietti elettronici. Ogni qual volta viene effettuata una prenotazione, il cliente riceve un QR code associato al codice identificativo della prenotazione stessa. Tale codice dovrà essere poi mostrato alla biglietteria al momento della visita effettiva.

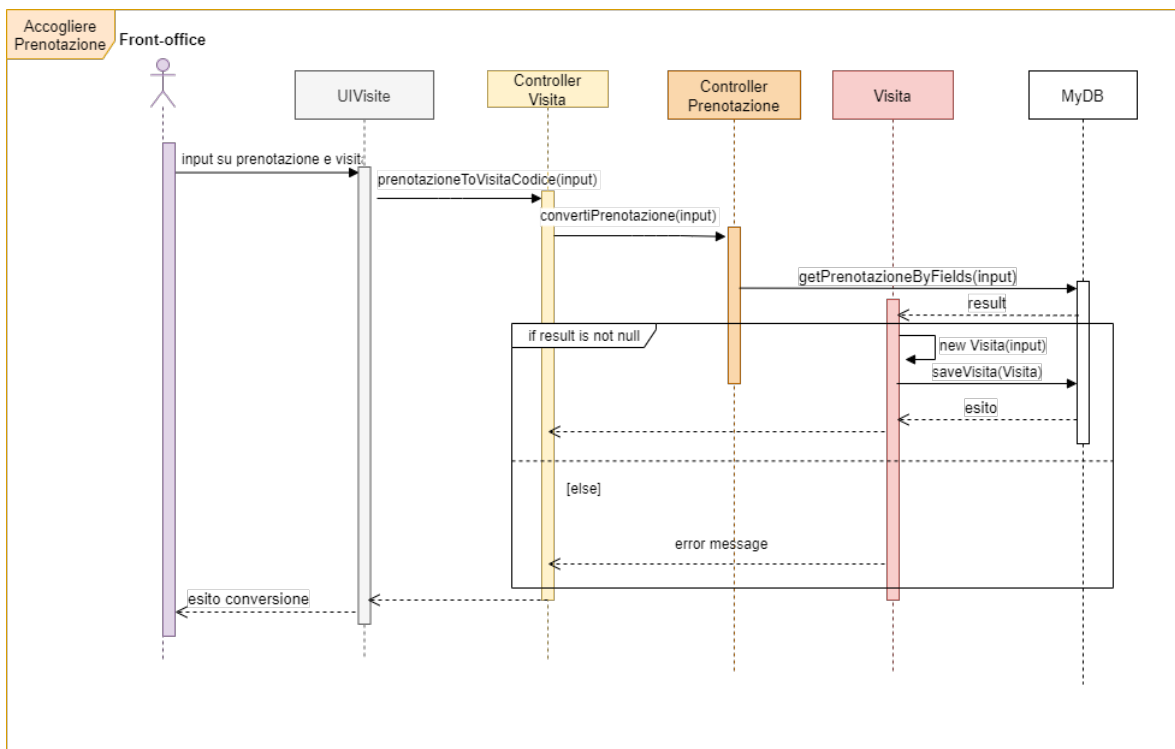
Diagrammi di sequenza relativi alla gestione delle visite.

Quando un visitatore si reca presso la biglietteria del museo, distinguiamo 3 casi:

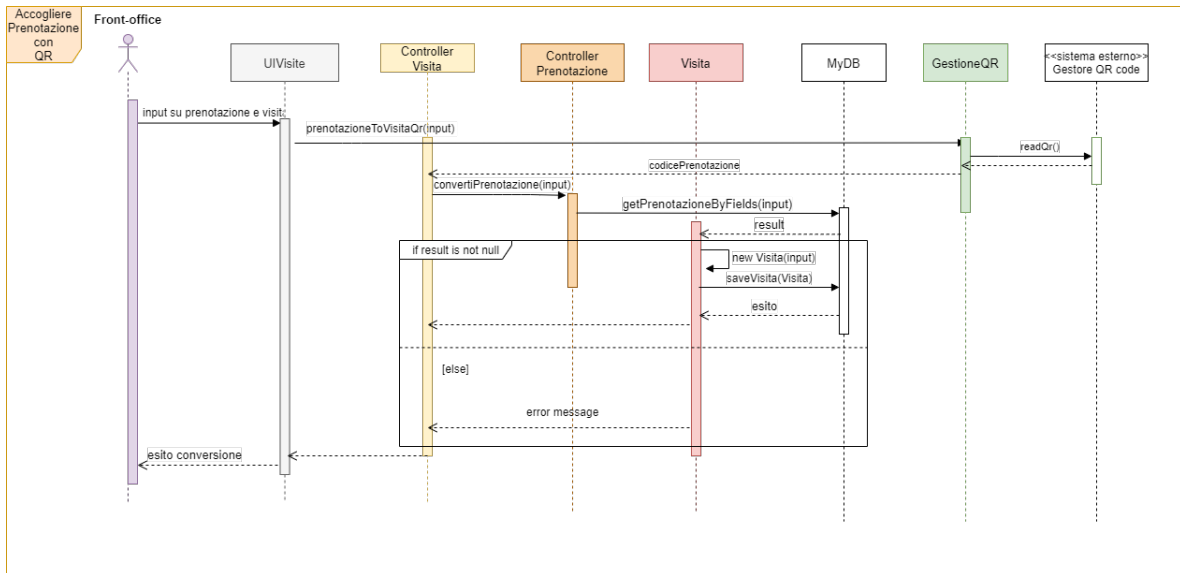
- Il visitatore non ha prenotato: in questo caso, se ci sono posti disponibili, la prenotazione viene creata contestualmente all'ingresso e viene effettuata la visita



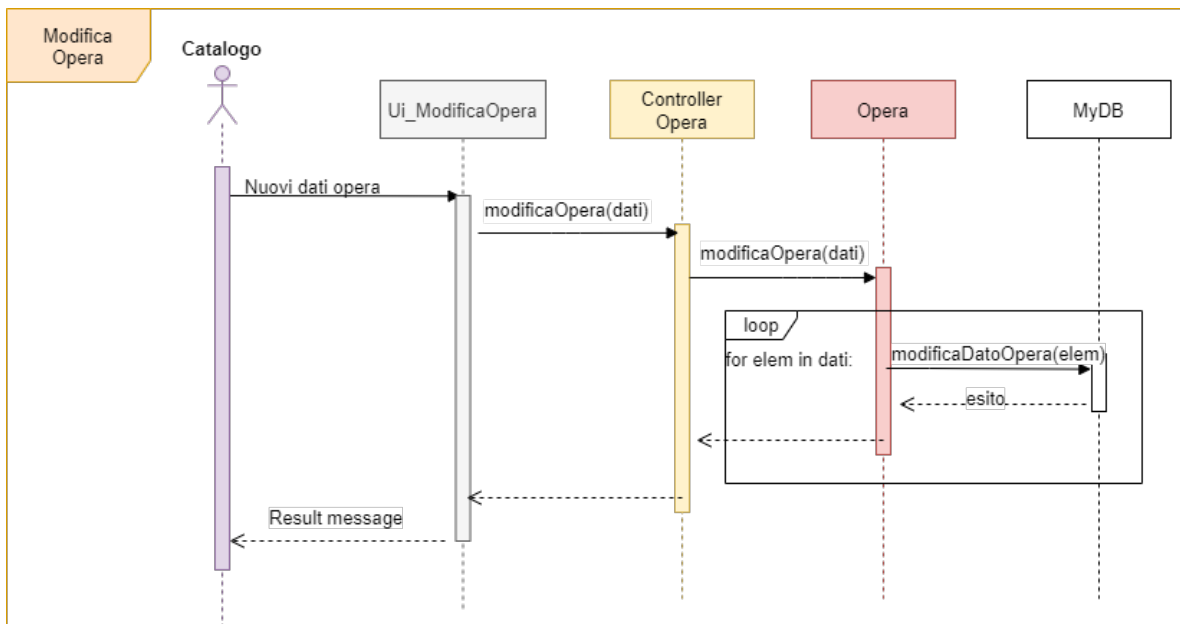
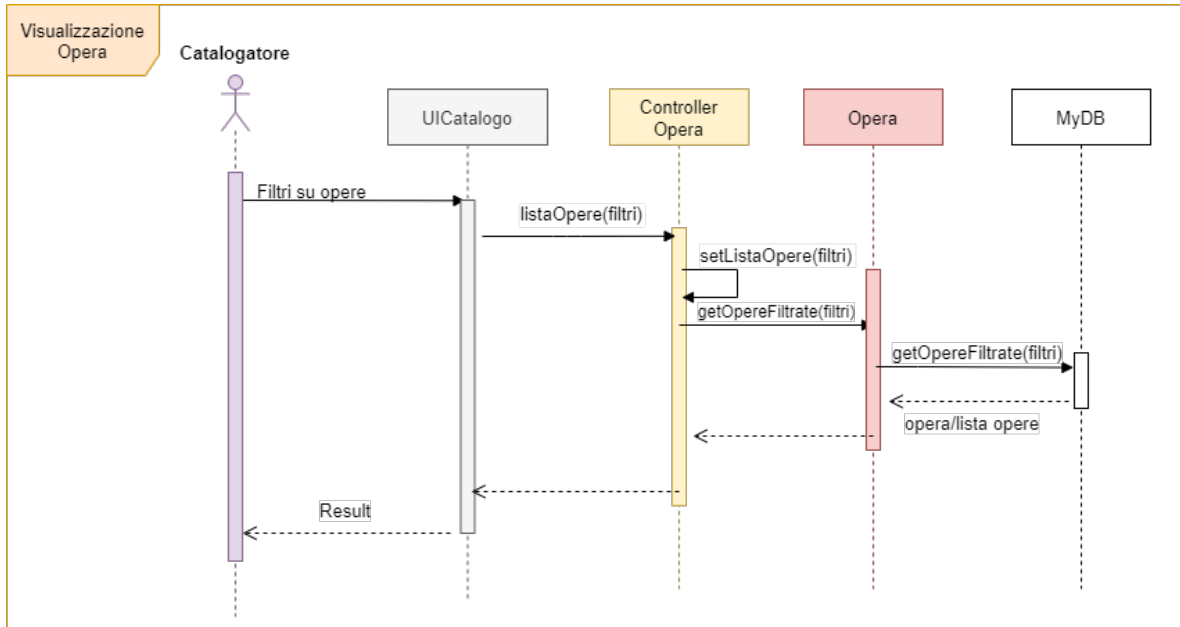
- Il visitatore ha prenotato e ha il codice della prenotazione: la visita viene effettuata dopo aver verificato la validità della prenotazione

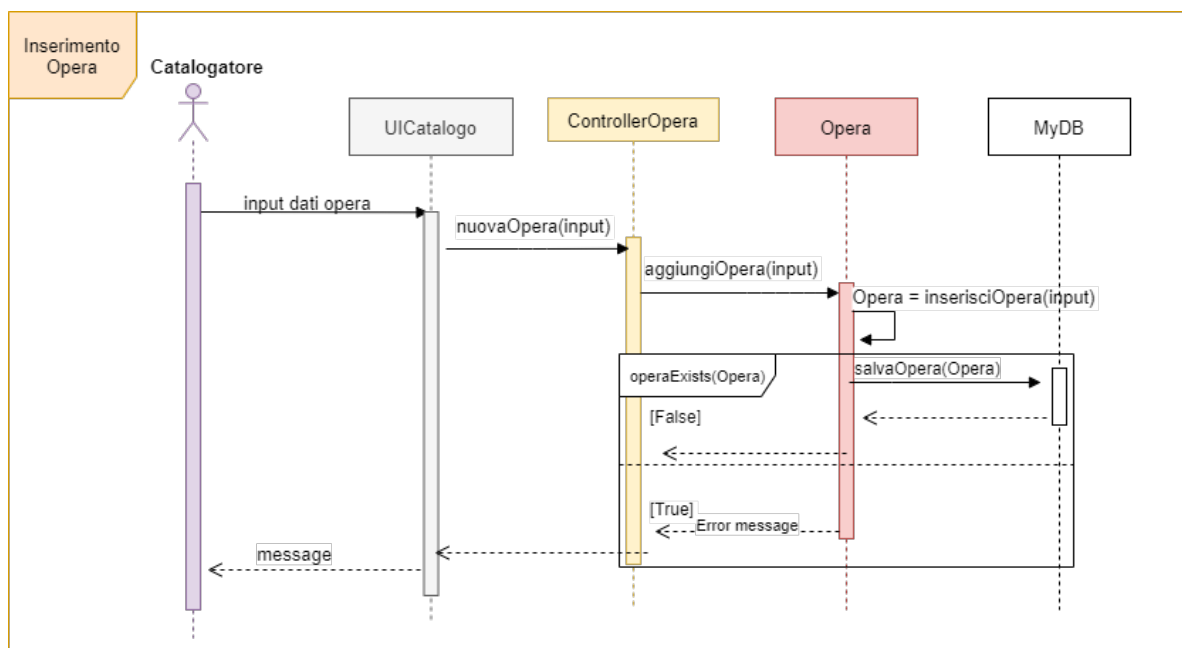
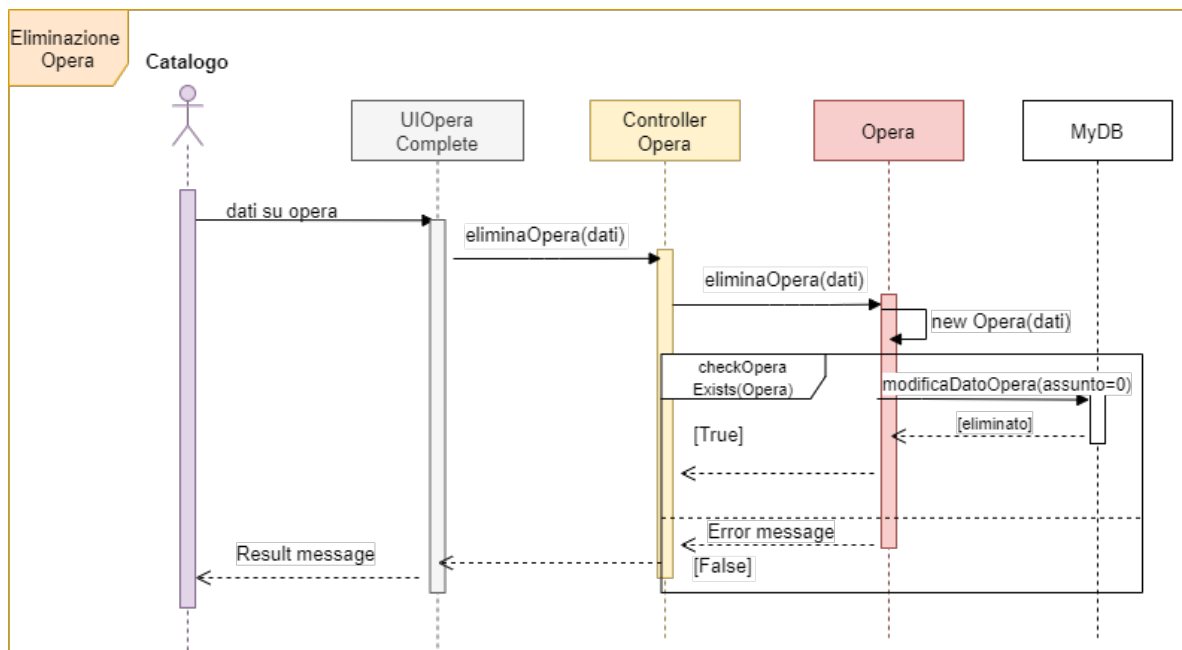


- Il visitatore ha prenotato e ha il codice QR della prenotazione: grazie al sistema esterno *Gestore QR code*, si rileva dal QR il codice identificativo della prenotazione. Dopo averne verificato la validità, la visita può avere luogo.

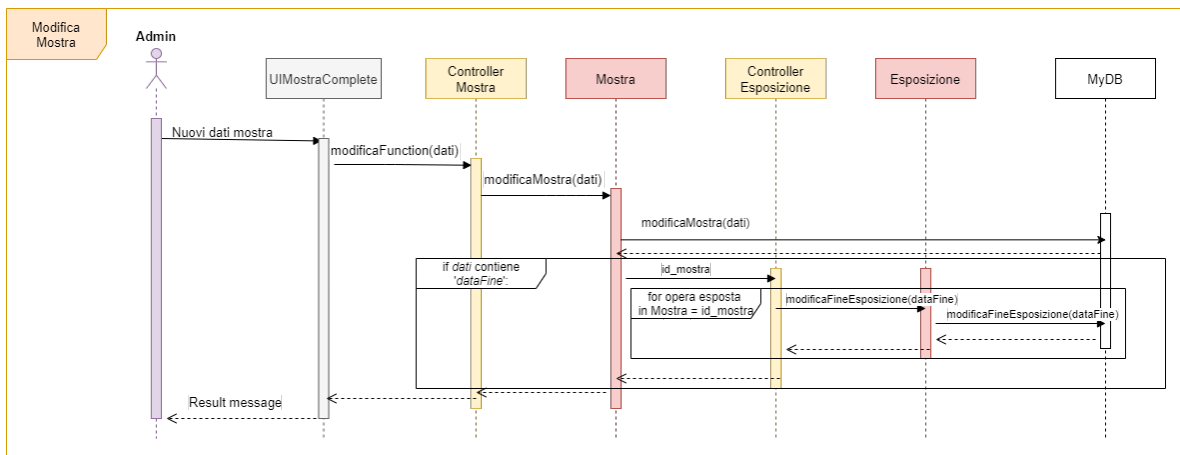
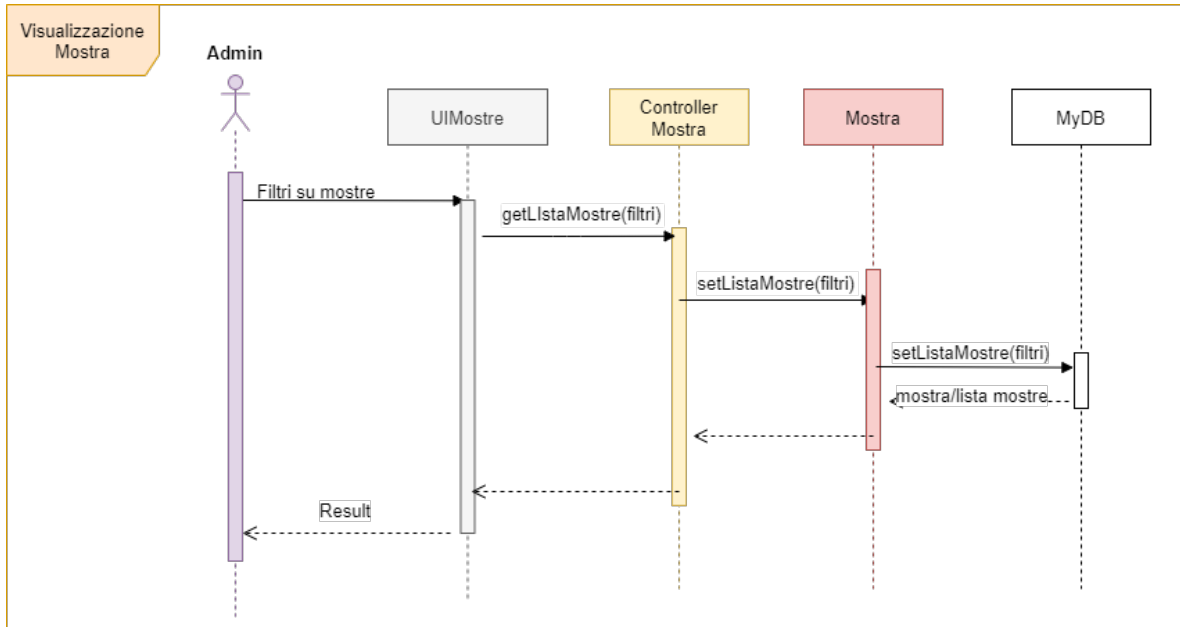


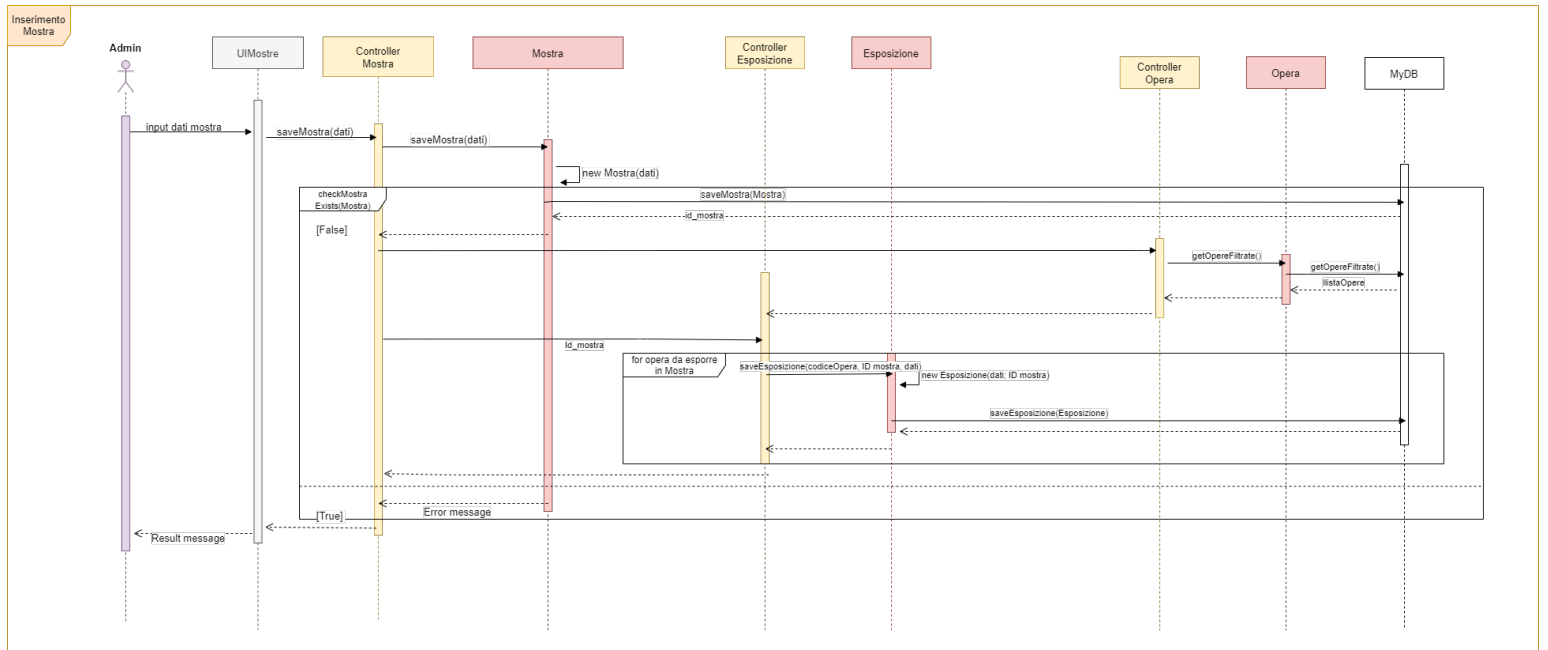
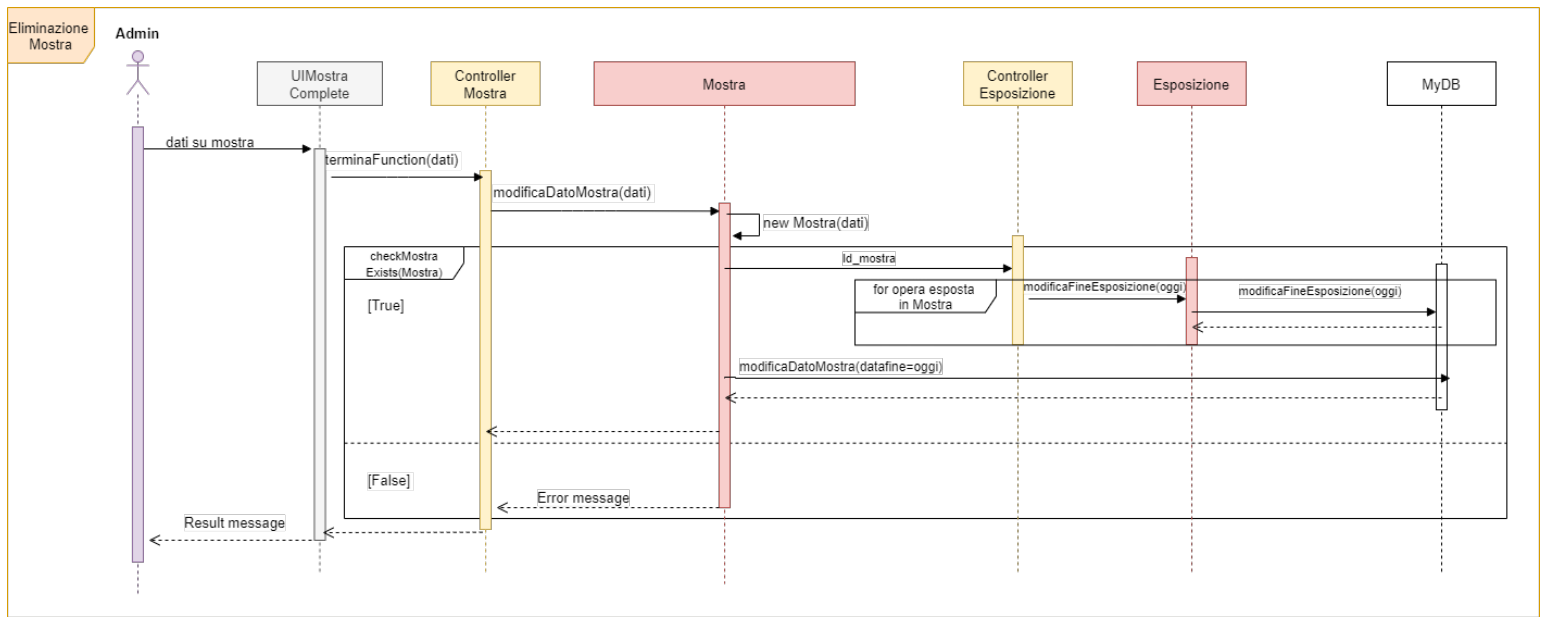
Diagrammi di sequenza relativi alla gestione delle opere.



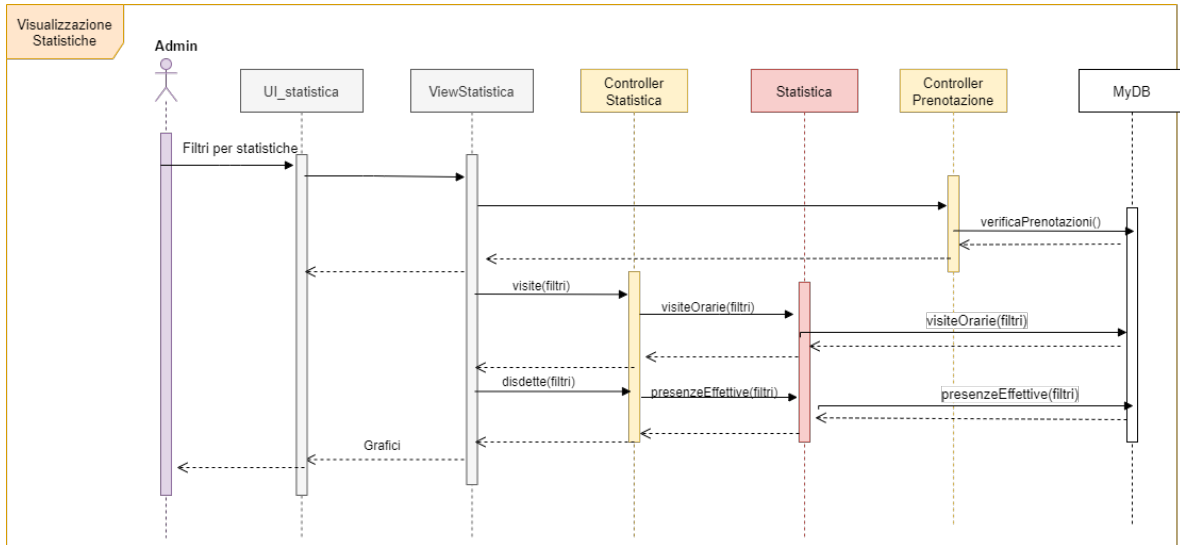


Diagrammi di sequenza relativi alla gestione delle mostre.



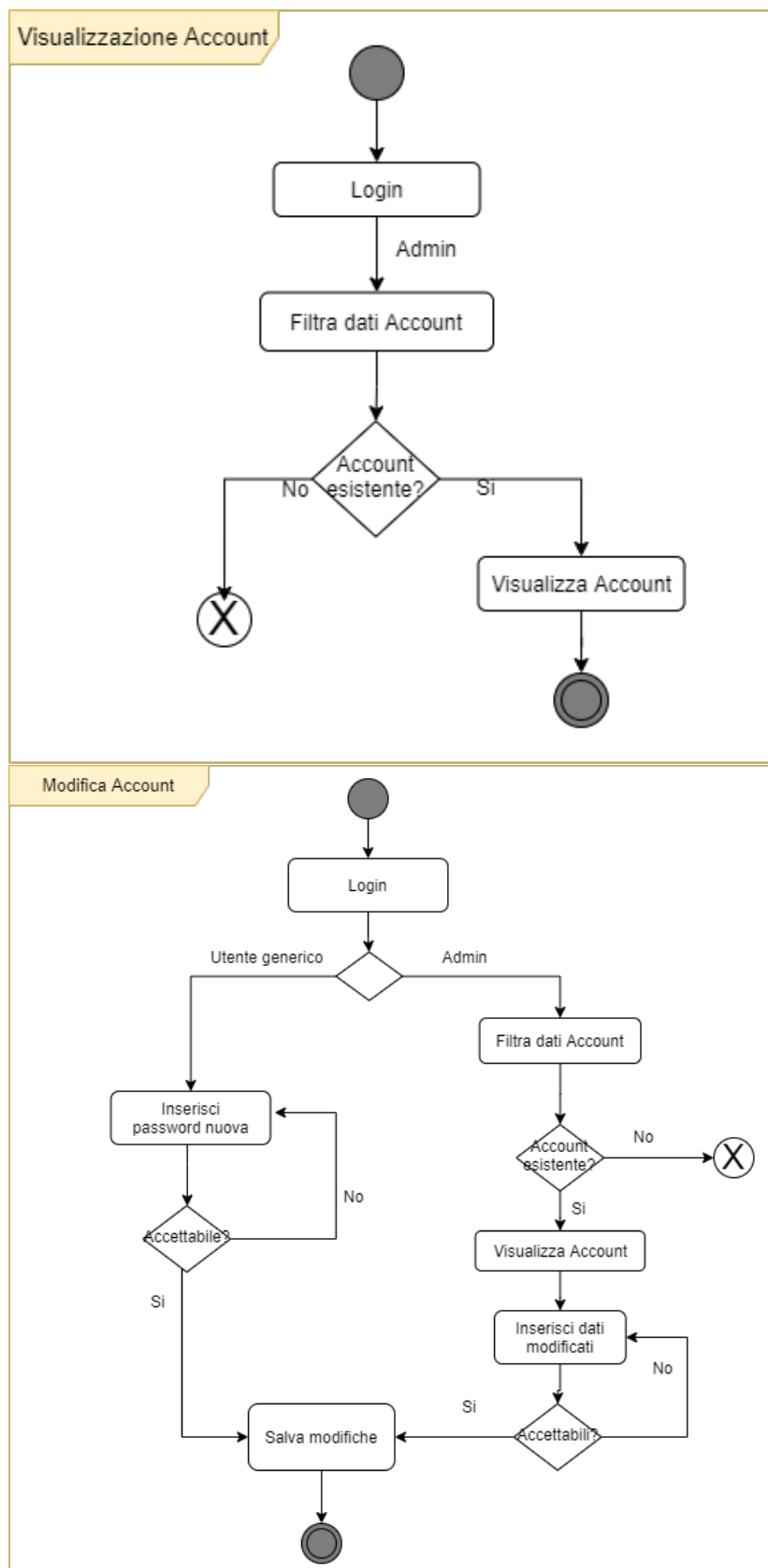


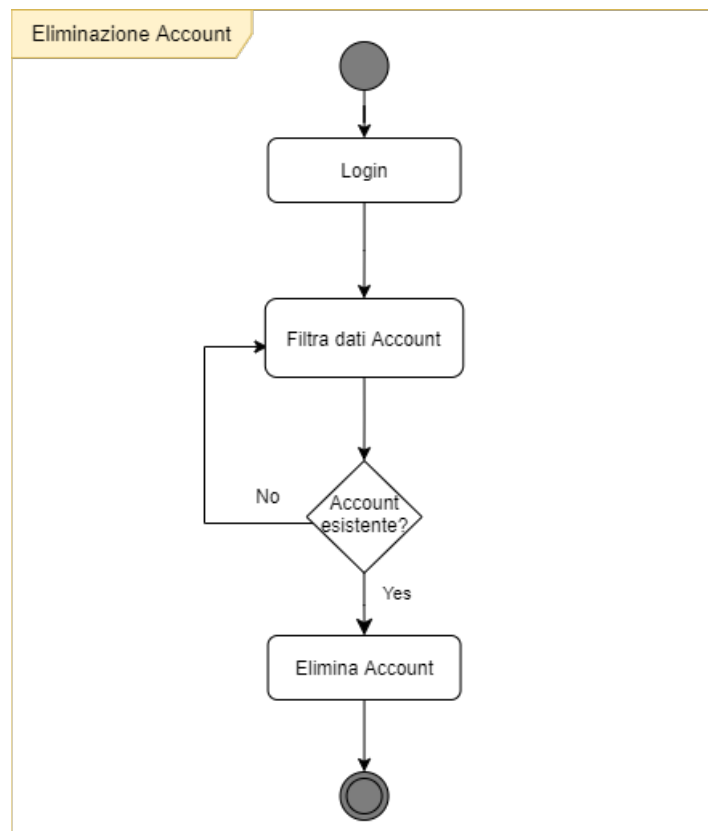
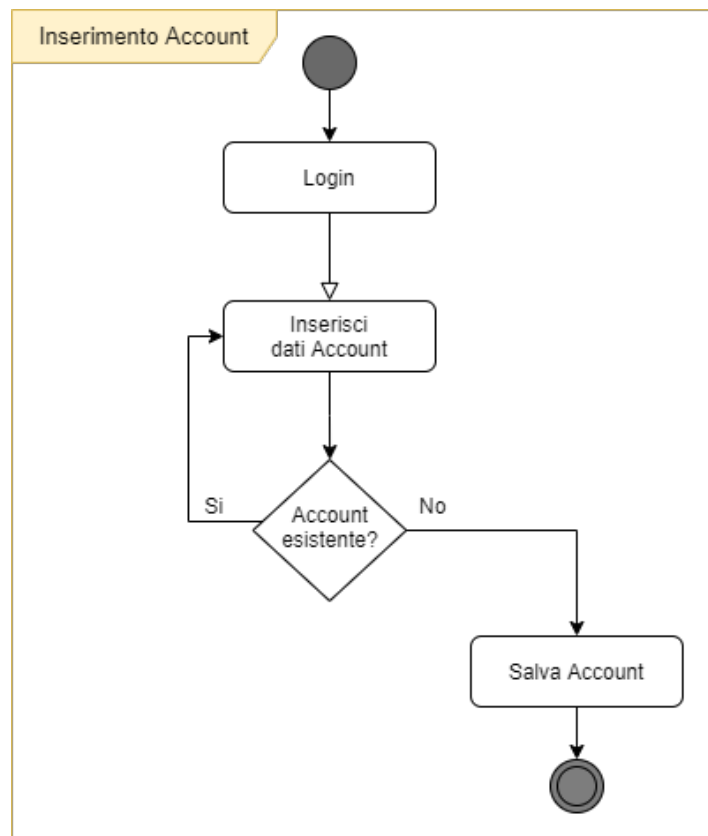
Diagrammi di sequenza relativi alla gestione delle statistiche.



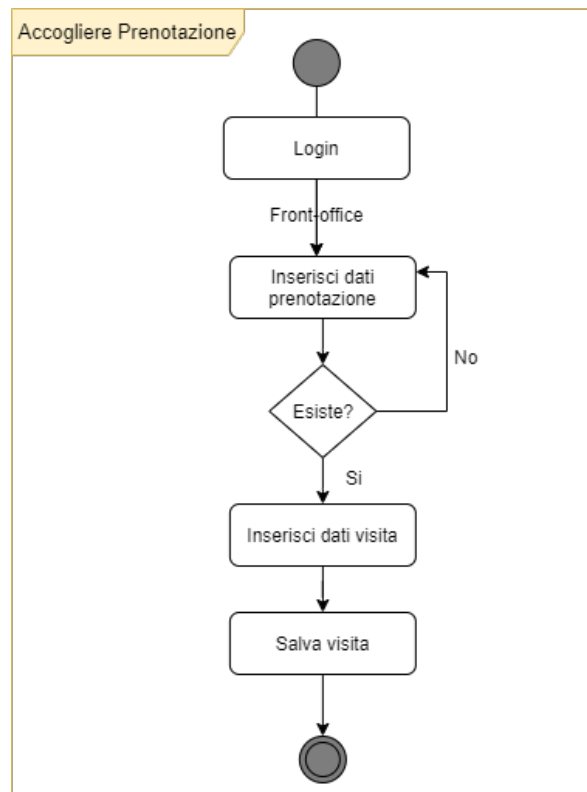
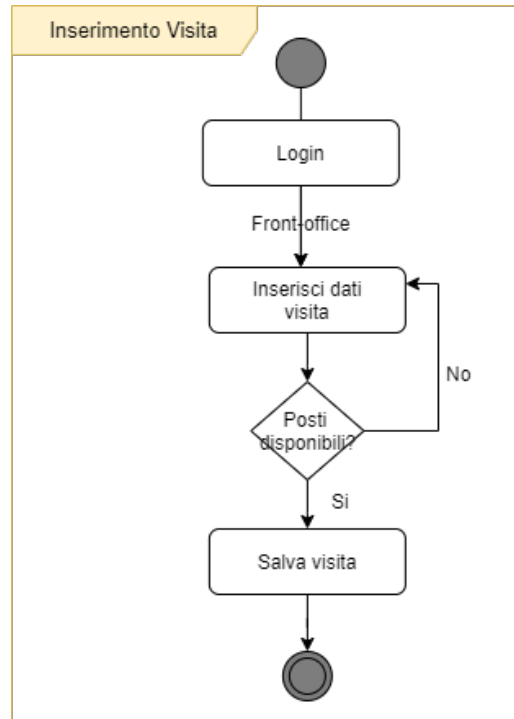
6.15 Diagrammi di attività

Attività: Gestione Account

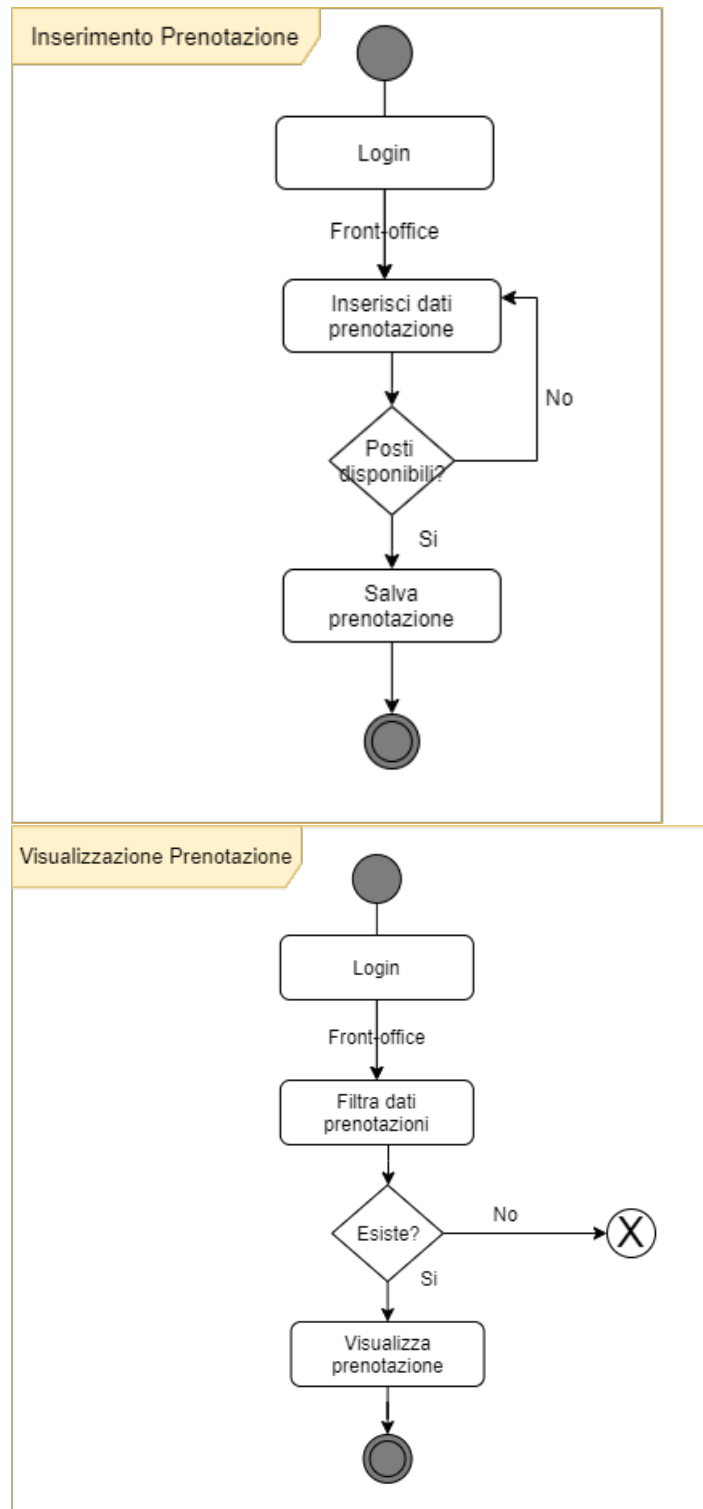


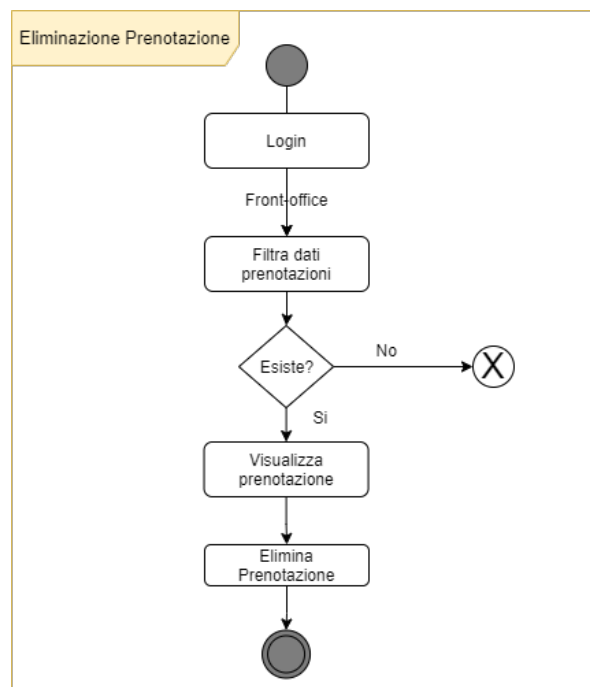
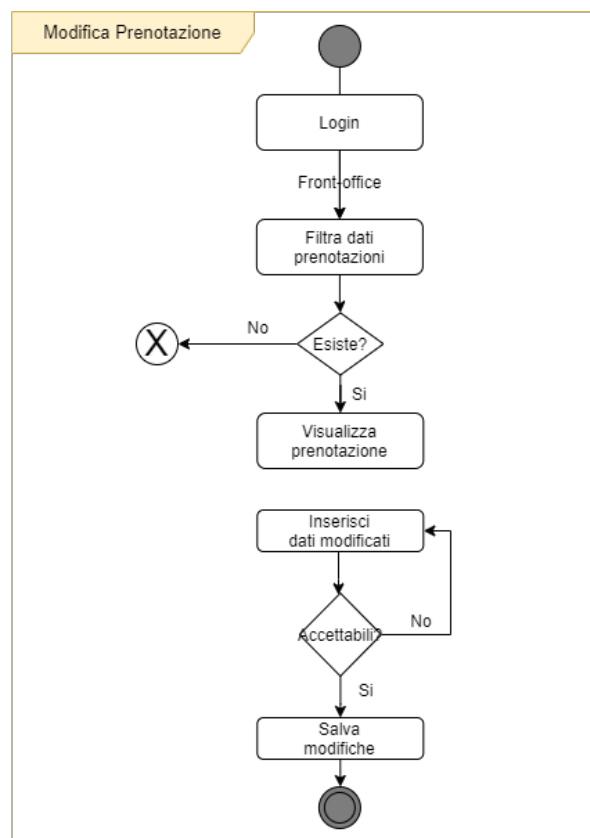


Attività: Gestione Visite

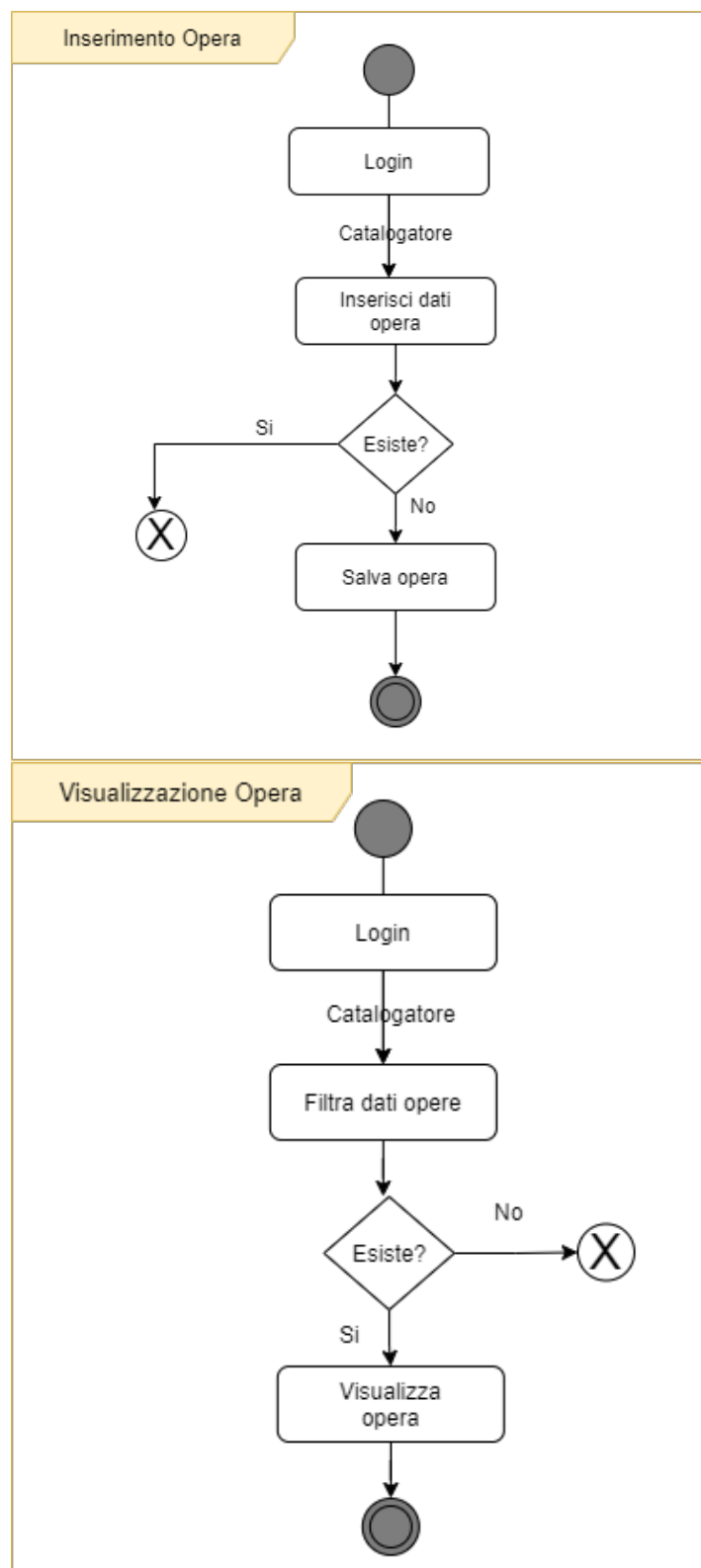


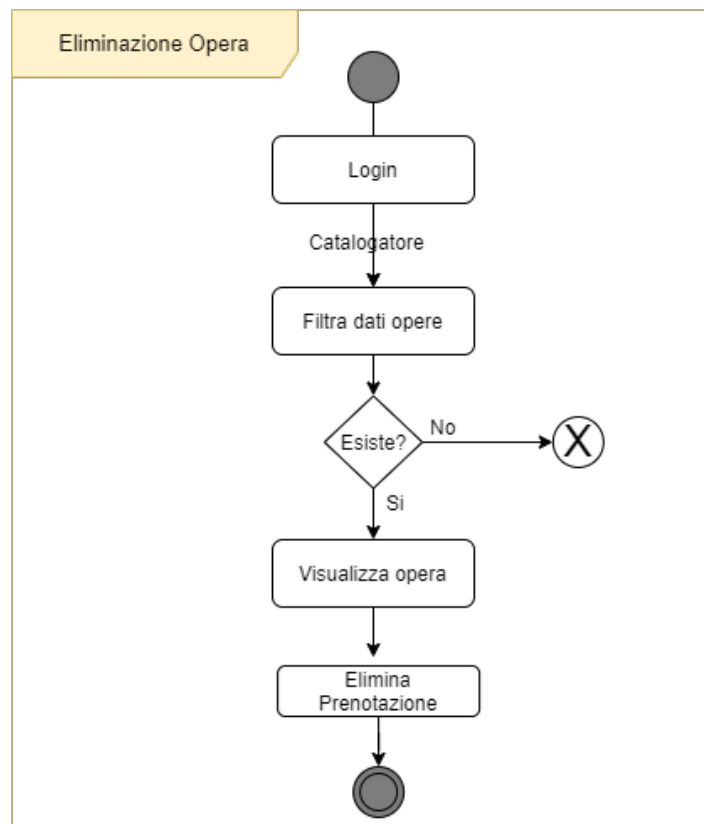
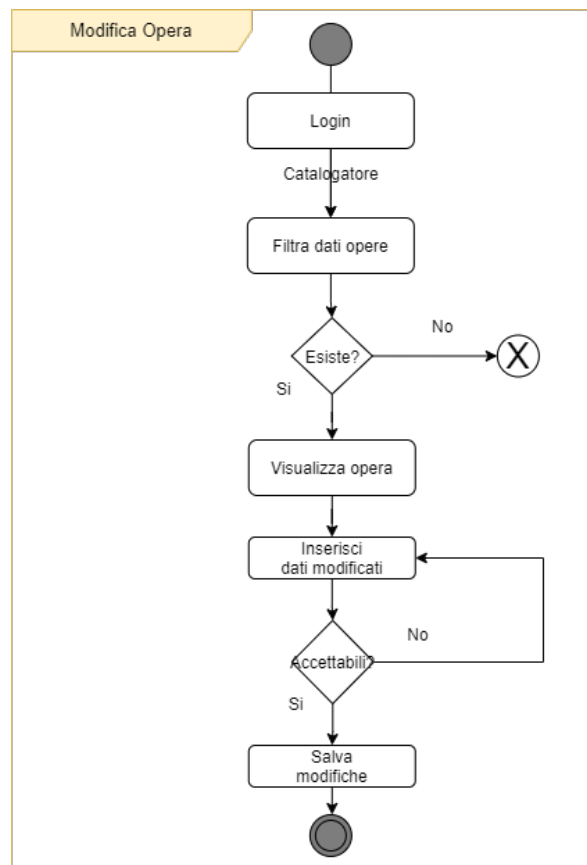
Attività: Gestione Prenotazioni



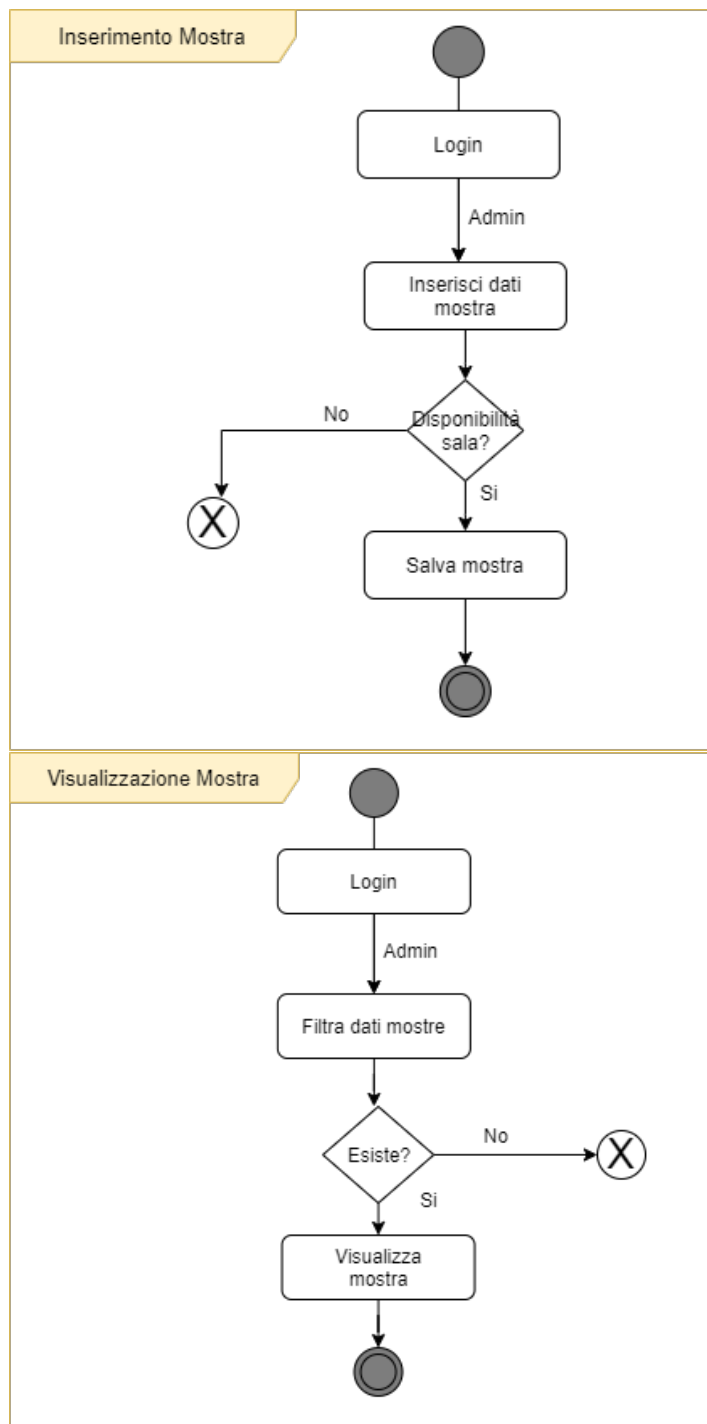


Attività: Gestione Opere

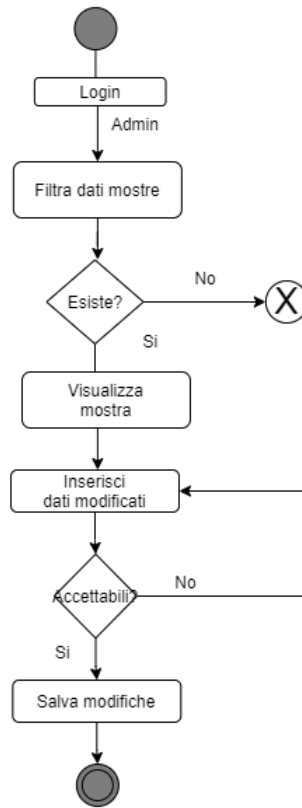




Attività: Gestione Mostre



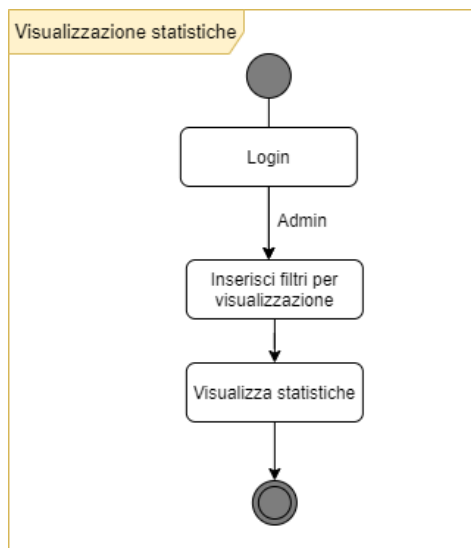
Modifica Mostra



Eliminazione Mostra



Attività: Calcolo Statistiche



7 Dettagli implementativi

Il linguaggio scelto per l'implementazione del software è Python 3, l'IDE di riferimento per lo sviluppo è PyCharm.

L'interfaccia grafica è sviluppata tramite PyQt, generata tramite il software *QT Designer*.

Per quanto riguarda la gestione dei dati, si utilizza la libreria *sqlite3* che implementa un DBMS incorporabile all'interno dell'applicazione.

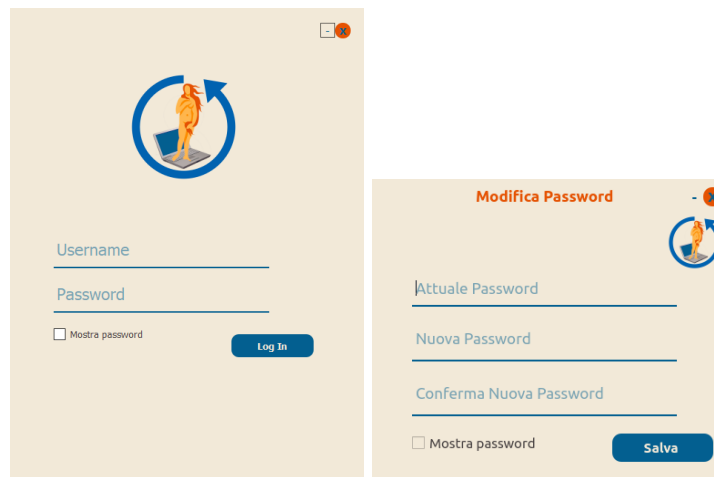
In generale, i pacchetti esterni richiesti dal progetto sono specificati nel file *requirements.txt*.

Nel package Configurazione, inoltre, troviamo il file *StrutturaMuseo.json*, dove vengono definite le sale per le esposizioni e il n. massimo di prenotazioni giornaliere.

7.1 Graphic User Interface

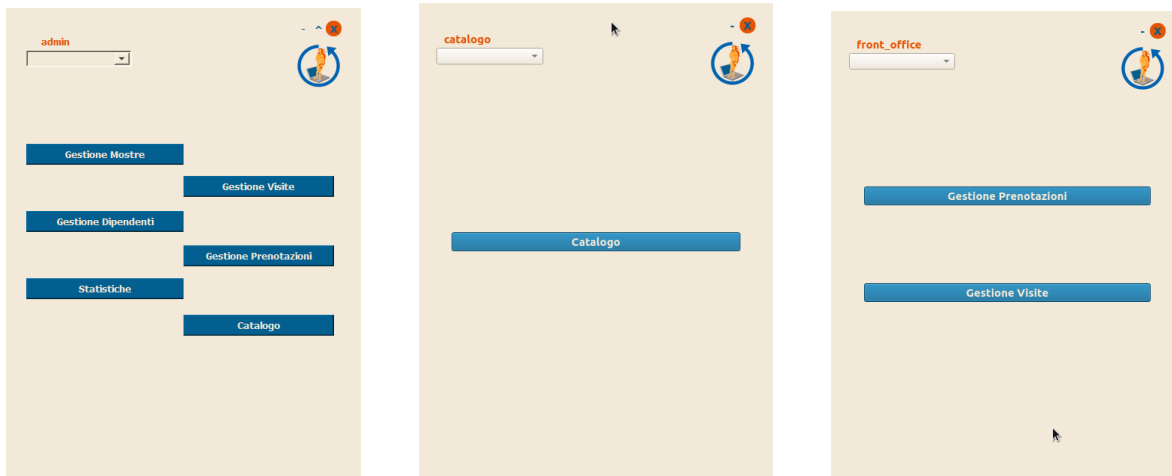
7.1.1 Login

A ciascun utente è assegnato un username univoco che lo identifica e una password dall'amministratore del sistema; avranno la possibilità poi di modificare la password in seguito.



7.1.2 Homepage

Dopo aver effettuato il login con successo, all'utente viene presentata l'homepage del sistema. In base ai privilegi dell'utente che ha eseguito l'accesso, verranno presentate Homepage con diverse funzionalità.



7.1.3 Amministratore

Le figure sottostanti mostrano come l'amministratore possa, attraverso l'interfaccia, gestire gli account dei vari dipendenti:

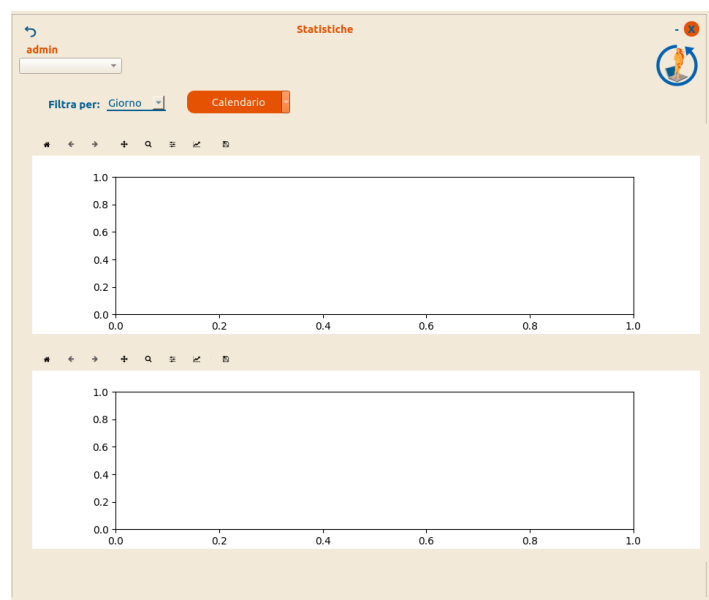


(a) Lista dei dipendenti

(b) Visualizzazione di un account

(c) Modifica di un account

L'amministratore è l'unico a poter visualizzare le statistiche mensili e giornaliere sui dati raccolti dal museo, in formato grafico:



(Per visualizzare le statistiche prendere in riferimento i mesi di Aprile/Maggio 2021.)

Per quanto riguarda la gestione degli eventi nel museo:

(a) Inserimento di una mostra

(b) Visualizzazione di una mostra

Se desidera, inoltre, può modificare la struttura del museo, aggiungendo nuove sale e ridefinendo il n. massimo di prenotazioni giornaliere (default: 210 visitatori):

7.1.4 Catalogatore

Le figure sottostanti mostrano come il catalogatore possa, attraverso l'interfaccia, gestire il patrimonio artistico del museo:

The screenshot shows the 'Gestione Catalogo' interface. At the top, there is a user dropdown menu set to 'admin' and a profile icon. Below this, a list of artworks is displayed. Each entry includes the title, author, and a 'Visualizza Scheda Opera' button.

Disegno Urlato	Quadro
Niccolò Calvi	
1	Visualizza Scheda Opera
Natura morta con gamberi	Quadro
Oswaldo Paniccia	
2	Visualizza Scheda Opera
Free Hugs	Quadro
Joan Cornellà	
3	Visualizza Scheda Opera

(a) Lista delle opere

The screenshot shows the 'Inserisci Nuova' form in the 'Gestione Catalogo' interface. It has a title bar 'Gestione Catalogo' and a user dropdown set to 'admin'. Below the title bar are two tabs: 'Inserisci Nuova' (active) and 'Visualizza elenco'. The form contains several input fields for artwork details, followed by an 'Inserisci' button.

Inserisci opera #ID

Titolo _____

Autore _____

Tipologia _____

Dimensioni _____

Anno di produzione _____

Immagine _____

Data acquisizione _____

Valore _____

Inserisci

(b) Inserimento di un'opera

The screenshot shows the 'Visualizza Scheda Opera' interface. It has a title bar 'Visualizza Scheda Opera' and a user dropdown set to 'admin'. Below the title bar is a table displaying the details of an artwork.

Titolo	Codice
Disegno Urlato	1
Autore	Anno Produzione
Niccolò Calvi	2018
Tipologia	Dimensioni
Quadro	50x70
Data Acquisizione	Valore
2021-01-23	250000.0

Modifica Opera **Elimina Opera**

(c) Visualizzazione di un'opera

The screenshot shows the 'Modifica Opera' interface. It has a title bar 'Modifica Opera' and a user dropdown set to 'admin'. Below the title bar is a form for editing an artwork's details, followed by a 'Salva' button.

Modifica Opera

Disegno Urlato _____

Niccolò Calvi _____

Quadro _____

1 250000.0

50x70 -

2018 _____

2021-01-23 _____

Salva

Figura 17: Modifica di un'opera

7.1.5 Front-office

Le figure sottostanti mostrano come il front-office possa, attraverso l'interfaccia, gestire gli ingressi e le prenotazioni del museo:



(a) Visualizza una prenotazione



(b) Cercare una prenotazione

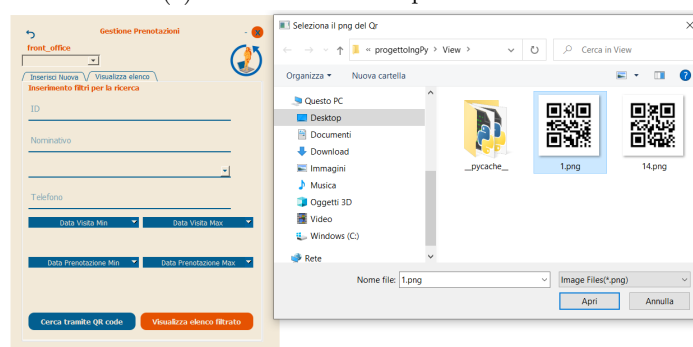


(c) Inserimento di una visita

In particolare, per quanto riguarda la creazione e la rilevazione dei codici QR:



(a) Inserimento di una prenotazione



(b) Verifica prenotazione con QR code

8 Test del software

Il completamento del software è raggiunto dopo che esso è stato accuratamente testato. Per eseguire i test delle componenti del sistema è stato utilizzato il framework *UnitTest*, una libreria che permette di verificare se metodi e classi si comportano come previsto.

Le componenti testate riguardano la gestione del login, dei dipendenti, delle opere, delle mostre, delle prenotazioni e delle visite.

I relativi test sono raccolti in un package a sé chiamato *Tests*.

Vediamo di seguito un esempio di Unit Test riguardante la gestione dell'autenticazione:

```
from unittest import TestCase

from database.MyDB import db
from Login.ControllerLogin.GestioneLogin import GestioneLogin

class TestGestioneLogin(TestCase):
    def setUp(self):
        sql = " INSERT OR REPLACE INTO lavoratore VALUES('test','af4500a0be019f949d939b38" \
            "9585f95665b03e2b879e0a6a474744ac81a4bc95262647f11661ac3f1f4880c69117a56a9e3e61bc" \
            "81a12e4139f039bb28ed7cad46de211dff08df6b11b0efd63abe0c8e09e12c22d039cd331a8c3270c7f52dc7'," \
            " 'testtesttesttest', 'test test', 'admin', 1) "
        db.connection.execute(sql)
        self.gestioneLogin = GestioneLogin()

    def tearDown(self):
        sql = "DELETE FROM lavoratore WHERE username = 'test'"
        db.connection.cursor().execute(sql)
        db.connection.commit()
```

I metodi `setUp()` e `tearDown()` consentono di definire le istruzioni che verranno eseguite prima e dopo ogni metodo di test.

```

def test_changePasswd(self):
    # login dell'utente che vuole effettuare la modifica password
    self.assertTrue(self.gestioneLogin.verificaCredenziali(password='test', username='test'))

    # modifica password fallita (errore nella conferma della vecchia password)
    self.assertRaises(Exception, lambda: self.gestioneLogin.changePasswd('test', 'nuova_password'))

    # modifica password fallita (errore nella conferma della nuova password)
    self.assertRaises(Exception, lambda: self.gestioneLogin.changePasswd('test', 'test1'))

    # modifica password fallita (policy non rispettata --> password debole)
    self.assertRaises(Exception, lambda: self.gestioneLogin.changePasswd('test1', 'test1'))

    # password aggiornata con successo
    self.assertTrue(self.gestioneLogin.changePasswd('S3cur3P@ssw0rd', 'S3cur3P@ssw0rd'))

    self.gestioneLogin.logout()

    # eseguo il login con la nuova password
    self.assertTrue(self.gestioneLogin.verificaCredenziali(password='S3cur3P@ssw0rd', username='test'))

```

Figura 20: Metodo per testare la modifica della password di un utente

```

✓ Tests passed: 1 of 1 test – 270 ms

C:\Users\saram\AppData\Local\Programs\Python\Python39\python.exe "C:\Program Files\JetBrains\PyCharm 2021.1.3
Testing started at 09:39 ...
Launching unittests with arguments python -m unittest test_GestioneLogin.TestGestioneLogin.test_changePasswd

Process finished with exit code 0

Ran 1 test in 0.268s

OK ←

```

Figura 21: Esito del test

Dopo aver accertato il funzionamento del software, esso può ritenersi concluso.

Grazie per l'attenzione!