

UmmahSpace

Documentazione del Progetto

M. Marulli, M. Baka

Indice

1	Specifiche dei requisiti software	2
1.1	Introduzione	2
1.2	Panoramica del sistema	2
1.3	Requisiti hardware e software	2
1.4	Sistemi Correlati, pro e contro	2
1.5	Requisiti Funzionali	3
1.6	Diagramma dei Casi d'Uso	4
1.7	UC Internal Steps – Iscrizione evento	4
1.8	UC Internal Steps – Gestione eventi	5
2	Storyboards	6
3	Design	8
3.1	VOPC	8
3.2	Design-level diagram	9
3.2.1	Singleton	9
3.2.2	Facade	10
3.2.3	Factory	10
3.2.4	Adapter	10
3.3	Activity Diagram	11
3.4	Sequence Diagram	13
3.5	State Diagram	15

1 Specifiche dei requisiti software

1.1 Introduzione

Questo documento ha lo scopo di fornire una descrizione dettagliata di **UmmahSpace**, delineando le funzionalità principali, l'architettura del sistema e i requisiti tecnici necessari per il suo sviluppo e utilizzo.

1.2 Panoramica del sistema

L'app è progettata per supportare la comunità musulmana nella crescita spirituale quotidiana. Gli utenti possono accedere agli orari di preghiera, monitorare i progressi spirituali e partecipare a eventi online offerti dagli organizzatori. Gli organizzatori, dal canto loro, possono creare eventi educativi, caricare contenuti aggiuntivi e gestire incontri online, offrendo risorse spirituali e di apprendimento per la community.

1.3 Requisiti hardware e software

- **Software:** Java SDK, IDE: IntelliJ IDEA, database (es. MySQL).
- **Hardware:** Computer con almeno 4GB di RAM, connessione a internet.

1.4 Sistemi Correlati, pro e contro

- **OnePath Network.** Si tratta di un'app che offre contenuti educativi e motivazionali per la crescita personale e spirituale, inclusi video, articoli e podcast creati da esperti religiosi. Fornisce video, podcast e articoli educativi da esperti, accessibili on-demand, che aiutano gli utenti a crescere spiritualmente; ma non offre un tracker spirituale per monitorare le pratiche religiose giornaliere né la possibilità di iscriversi a eventi o partecipare a incontri dal vivo.
- **Pillars.** Si tratta di un'app focalizzata sull'assistenza alla preghiera, con orari di preghiera precisi, direzione della Qibla, e contenuti educativi per supportare le pratiche religiose quotidiane. Fornisce orari di preghiera estremamente accurati grazie a dati aggiornati e basati sulla geolocalizzazione, molto utile per utenti che cercano una precisione assoluta, ma non include una community o gruppi tematici, limitando l'interazione sociale e il supporto tra gli utenti.

sectionUser Stories

- **Utente:**
 - *Come utente, voglio utilizzare il tracker spirituale, per monitorare i miei progressi nella lettura del Corano, nel digiuno e nella preghiera.*
 - *Come utente, voglio partecipare agli eventi, per accrescere le mie conoscenze sulla religione.*
 - *Come utente, voglio visualizzare gli orari della preghiera, per poter organizzare il mio tempo per pregare nei momenti corretti.*
- **Organizzatore:**
 - *Come organizzatore, voglio gestire i miei eventi, per modificarli o eliminarli in caso di imprevisti.*

- *Come organizzatore, voglio stampare i report dei miei eventi, per visualizzarne le partecipazioni.*
- *Come organizzatore, voglio aggiungere il link per l'evento, così che gli utenti iscritti possano accedervi facilmente al momento opportuno.*

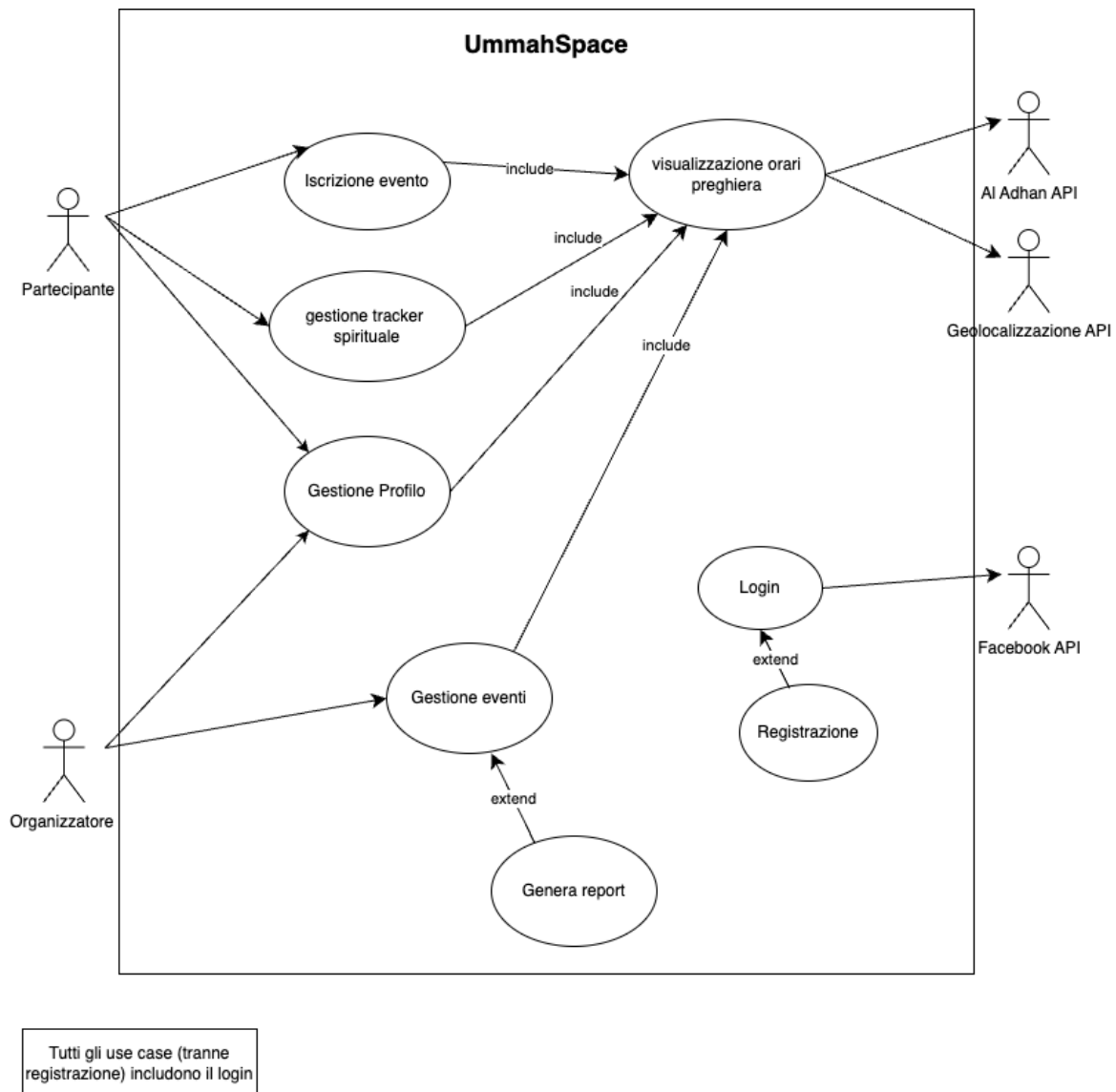
1.5 Requisiti Funzionali

Di seguito i principali requisiti funzionali offerti dal sistema:

1. **Calcolo orari.** Il sistema deve localizzare la posizione dell'utente a ogni accesso per poter calcolare e visualizzare gli orari delle preghiere.
2. **Registrazione e Accesso.** Il sistema deve permettere agli attori di registrarsi e accedere.
3. **Organizzazione calendario.** Il sistema deve visualizzare una lista personale con gli eventi a cui l'utente è iscritto.
4. **Aggiornamento del numero di iscritti.** Il sistema deve aggiornare il numero di iscritti all'evento quando l'utente conferma o annulla l'iscrizione.
5. **Monitoraggio Progressi.** Il sistema deve monitorare e registrare i progressi del tracker spirituale di ogni utente.
6. **Validazione e salvataggio dati.** Il sistema deve validare i dati inseriti dall'utente (ad esempio, l'e-mail deve avere un formato corretto e i campi obbligatori non possono essere vuoti) e salvare le modifiche solo se i dati sono validi.

1.6 Diagramma dei Casi d'Uso

Il diagramma rappresenta le funzionalità principali di UmmahSpace, evidenziando le interazioni tra partecipanti, organizzatori e il sistema.



1.7 UC Internal Steps – Iscrizione evento

1. L'utente effettua il login.
2. Il sistema apre la pagina home, visualizzando il calendario con celle colorate se ci sono eventi.
3. L'utente seleziona il giorno di interesse.
4. Il sistema apre una lista degli eventi presenti per il giorno selezionato.
5. L'utente seleziona l'evento a cui vuole partecipare.
6. L'utente si iscrive per confermare la partecipazione all'evento.
7. Il sistema verifica che non si sia raggiunto il limite massimo di partecipanti.

8. Il sistema verifica che l'evento non sia chiuso.
9. Il sistema aggiorna la lista dei partecipanti e la lista personale dell'utente aggiungendo l'evento.

Extensions:

- **6a. Evento duplicato:** il sistema rileva che l'utente è già iscritto all'evento, mostra un messaggio "Sei già iscritto a questo evento". Il sistema rimanda l'utente al punto 2.
- **7a. Superamento del limite massimo di iscrizioni per l'evento:** il sistema rileva che il numero massimo di partecipanti per l'evento è stato raggiunto e informa l'utente che non può più iscriversi. Il sistema rimanda l'utente al punto 2.
- **8a. Chiusura dell'evento:** il sistema rileva che l'evento è stato chiuso e informa l'utente che non può più iscriversi. Il sistema rimanda l'utente al punto 2.

1.8 UC Internal Steps – Gestione eventi

1. L'organizzatore effettua il login.
2. Il sistema apre la pagina home, visualizzando il calendario.
3. L'organizzatore seleziona un giorno del calendario.
4. Il sistema salva la data selezionata e apre il modulo di creazione dell'evento.
5. L'organizzatore compila i campi richiesti (titolo, descrizione, orario inizio, orario fine, link, limite partecipante).
6. Il sistema valida i dati inseriti, verificando che tutti i campi obbligatori siano compilati e che i formati siano corretti.
7. L'organizzatore aggiunge l'evento.
8. Il sistema salva l'evento nel database.
9. Il sistema aggiorna la lista degli eventi dell'organizzatore.
10. Il sistema modifica il colore della cella del calendario per indicare la presenza di un evento in quella data.
11. Il sistema aggiorna la lista degli eventi del giorno selezionato per includere il nuovo evento.

Extensions:

- **4a. Validazione dei dati fallita:** il sistema rileva che uno o più campi contengono errori (es. data in formato errato, campo vuoto), mostra un messaggio di errore e invita l'organizzatore a correggere i dati. All'inserimento dei nuovi dati si ritorna al punto 6 fino alla validazione dei dati.
- **6a. Annulla aggiunta evento:** l'organizzatore annulla la creazione, ritorna al punto 2.
- **7a. Errore salvataggio nel DB:** il sistema riscontra un problema nel salvataggio dell'evento, notifica l'organizzatore e ritorna al punto 2.

2 Storyboards

Questa sezione illustra, attraverso una serie di *storyboards*, il percorso dell'utente all'interno dell'app UmmahSpace, evidenziando le principali funzionalità e interazioni.



The login screen features a title "Accedi al tuo spazio spirituale" in a stylized font. Below it are two input fields for "E-mail" and "Password", each with a placeholder text "Inserisci la tua e-mail" and "Inserisci la tua password" respectively. A checkbox labeled "Sei un organizzatore?" is positioned below the password field. At the bottom, there are two buttons: a green "Login" button and a white "Accedi con Google" button with a red border.

Figura 1: Vista sulla schermata di accesso



The main dashboard has a decorative header with the text "Benvenuto utente!". On the right side of the header are two buttons: "Profilo" and "Esci". Below the header, there are three main sections: "Preghiera passata" (Fajr - 5:30), "Oggi" (5 Novembre 2025), and "Preghiera futura" (Zuhr - 12:15). Below these sections is a navigation bar with three tabs: "Calendario", "Tracker", and "I Miei Eventi". The "Calendario" tab is active, showing a calendar for November 2025. The calendar is a grid with days of the week (Lun, Mar, Mer, Gio, Ven, Sab, Dom) and dates (1 to 28). The dates are displayed in a grid format, with the current date (5) highlighted.

Lun	Mar	Mer	Gio	Ven	Sab	Dom
					1	2
3	4	5	6	7	8	9
10	11	12	13	14	15	16
17	18	19	20	21	22	23
24	25	26	27	28		

Figura 2: Vista sulla schermata principale

Benvenuto utente!

Profilo Esci

Preghiera passata
Fajr - 5:30

Oggi
10 Febbraio 2025

Preghiera futura
Zuhr - 12:15

Calendario Tracker I Miei Eventi

Lettura del Corano

Obiettivo giornaliero: 100
Pagine lette: 50

Imposta obiettivo Aggiungi lettura

Hai digiunato oggi?

Aggiungi eventuali note ...

Motivazione:

☐ Sunnah ☐ Recupero ☐ Volontario Specifico ☐ Volontario Generale

Salva

Preghiere

Salva

Figura 3: Vista sulla schermata del Tracker

Il tuo profilo

Nome:
Mario

Cognome:
Rossi

Username:
mario.rossi

E-mail:
mario.rossi@example.com

▼ Cambia Password

Password attuale: Inserisci password attuale

Nuova password: Inserisci la nuova password

Conferma Password: Conferma la tua password

Modifica Salva Modifiche Indietro

Figura 4: Vista sulla schermata di gestione profilo

3 Design

I seguenti diagrammi sono disponibili, per una visualizzazione più approfondita, nella cartella **/deliverables** del progetto.

3.1 VOPC

Nel contesto del VOPC, adottiamo un'implementazione MVC focalizzata sui due principali attori del sistema: Organizzatore e Partecipante.

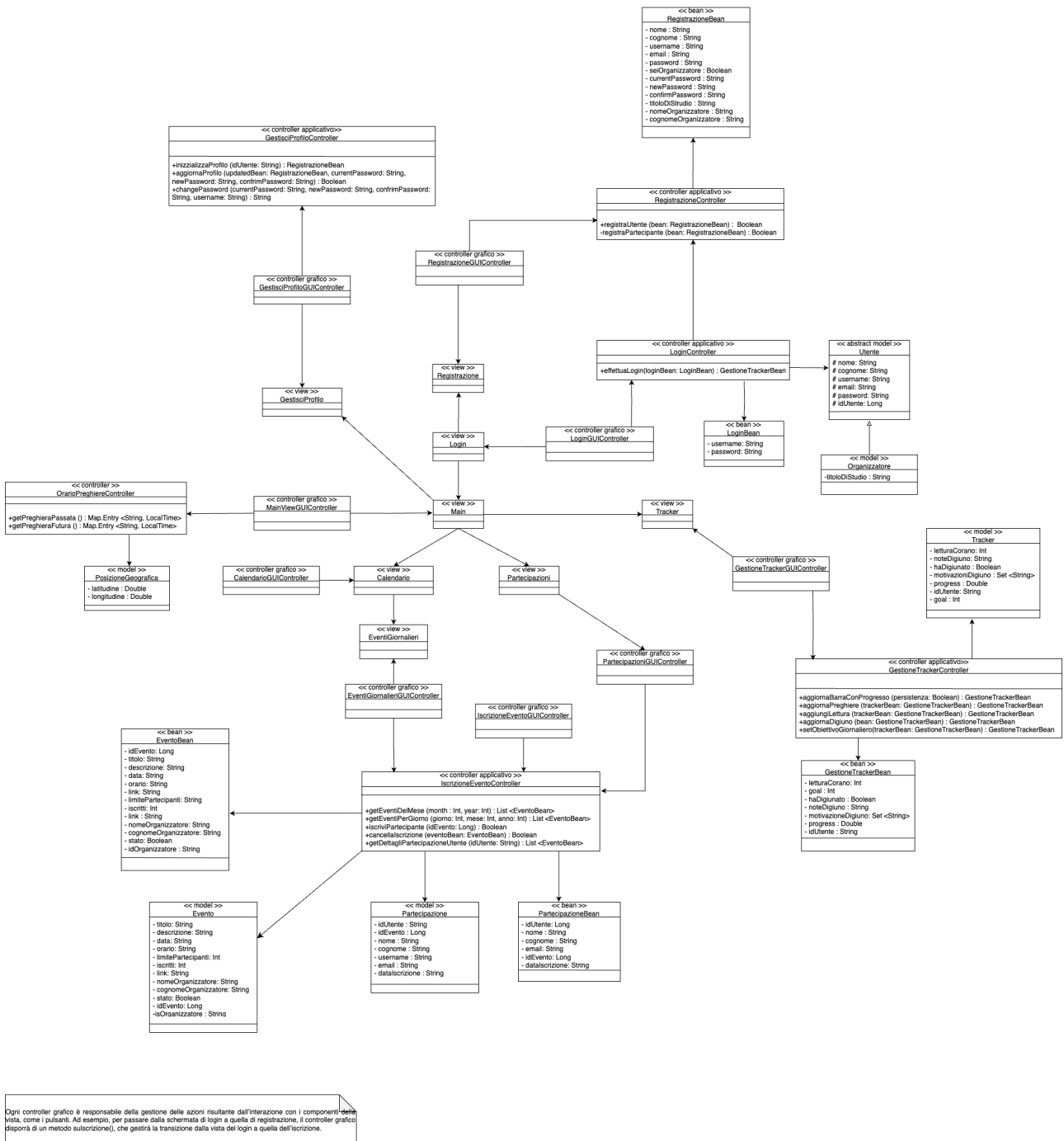


Figura 5: MVC Partecipante

- **Connessione al Database:** Il *Database Session Manager (DBSM)* utilizza un'istanza Singleton per gestire in modo efficiente le connessioni al database, evitando la creazione ridondante di nuove connessioni e migliorando le prestazioni dell'applicazione.
- **Model:** La classe `Model` segue il pattern Singleton per mantenere un'unica istanza della `ViewFactory`, ottimizzando il caricamento delle schermate grafiche e centralizzando la gestione della UI. Questo approccio riduce il consumo di risorse, evitando istanze multiple della `ViewFactory`.
- **Facade:** Il pattern Singleton è stato applicato alla classe `Facade`, che funge da interfaccia di alto livello per nascondere la complessità dell'accesso ai dati e delle operazioni di aggiornamento. Grazie al Singleton, la `Facade` garantisce un punto di accesso unico e condiviso per la gestione degli eventi e delle iscrizioni, migliorando la coerenza delle operazioni nel sistema.

Per rendere il Singleton sicuro nei contesti multi-thread, è stato talvolta implementato il **Double-Checked Locking**, il quale impedisce che due thread creino più istanze concorrenti del Singleton.

3.2.2 Facade

Il pattern Facade nasconde la complessità della gestione dei **DAO** e della creazione dei modelli, fornendo un'interfaccia unica per eseguire operazioni sui dati. Consente di gestire eventi, partecipazioni e utenti senza interagire direttamente con il database, semplificando l'accesso ai dati. L'accesso alla **Facade** avviene tramite il metodo `getInstance()`, garantendo un'unica istanza centralizzata che gestisce tutte le operazioni principali dell'applicazione.

3.2.3 Factory

Nel nostro progetto, il pattern Factory è stato adottato principalmente nella classe `ViewFactory` per gestire la creazione delle viste grafiche dell'applicazione. Ogni metodo della `ViewFactory` è responsabile della creazione di una specifica vista, caricando il relativo file `FXML`, istanziando il corrispondente controller e iniettandolo nella scena grafica.

3.2.4 Adapter

Il pattern Adapter è stato adottato per consentire l'integrazione tra diverse API esterne senza modificare il codice applicativo principale. In particolare, sono stati sviluppati due adapter:

- **GeolocalizzazioneIPAdapter:** si interfaccia con l'API di `ip-api.com` per ottenere la posizione geografica dell'utente in base all'indirizzo IP.
- **AlAdhanAdapter:** utilizza l'API `AlAdhan` per calcolare gli orari delle preghiere giornaliere in base alla posizione geografica.

Grazie a questa implementazione, il controller applicativo funge da doppio client, interagendo con entrambe le API tramite le interfacce `GeolocalizzazioneAPI` e `OrarioPreghiereAPI`.

Gli Activity Diagram scelti illustrano i processi chiave della gestione degli eventi e del monitoraggio del tracker spirituale del partecipante.



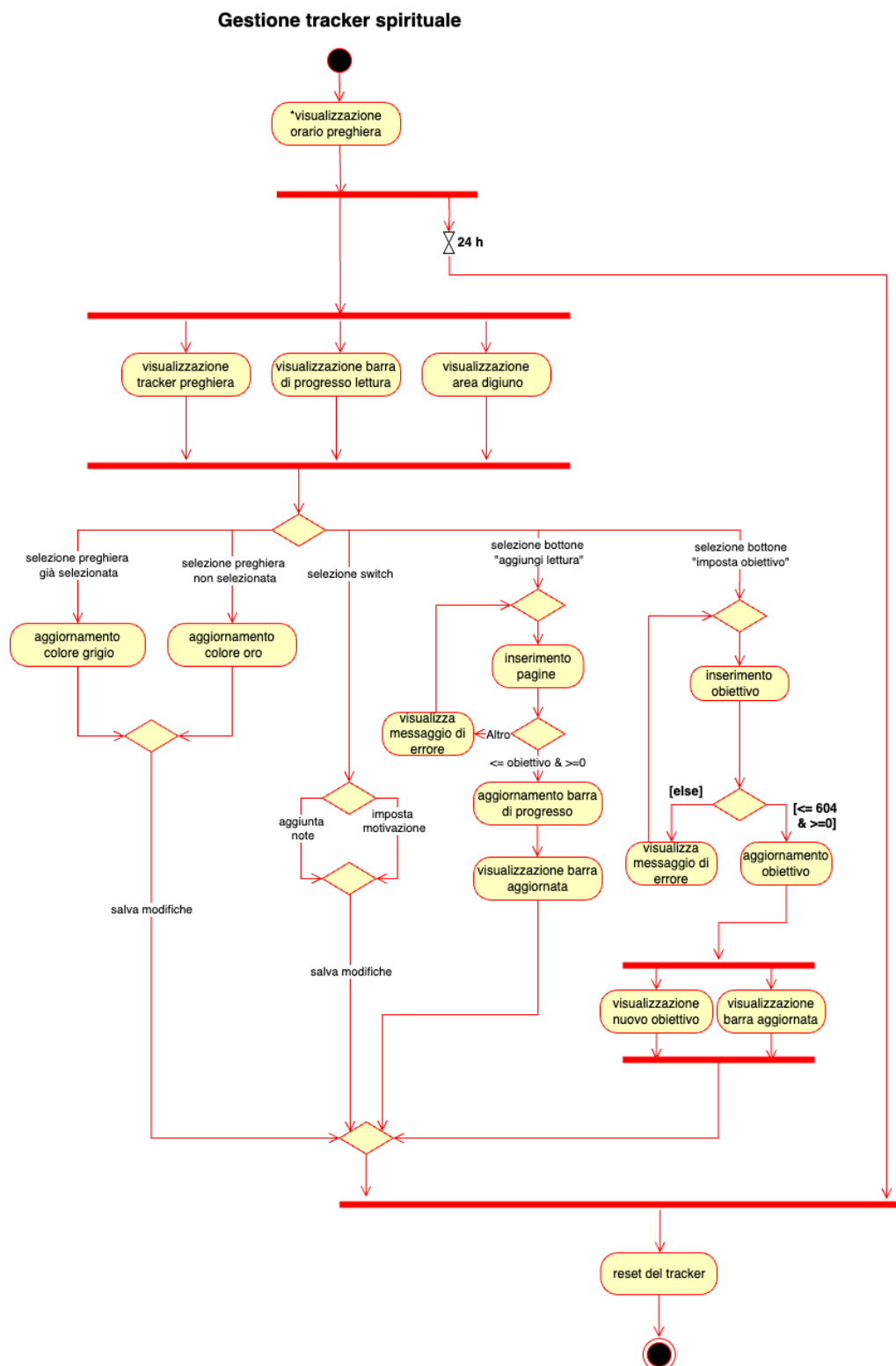


Figura 8: Activity Diagram per la gestione del tracker spirituale

3.4 Sequence Diagram

I Sequence Diagram selezionati approfondiscono il processo di iscrizione a un evento e la gestione del tracker spirituale del partecipante.

Iscrizione Evento

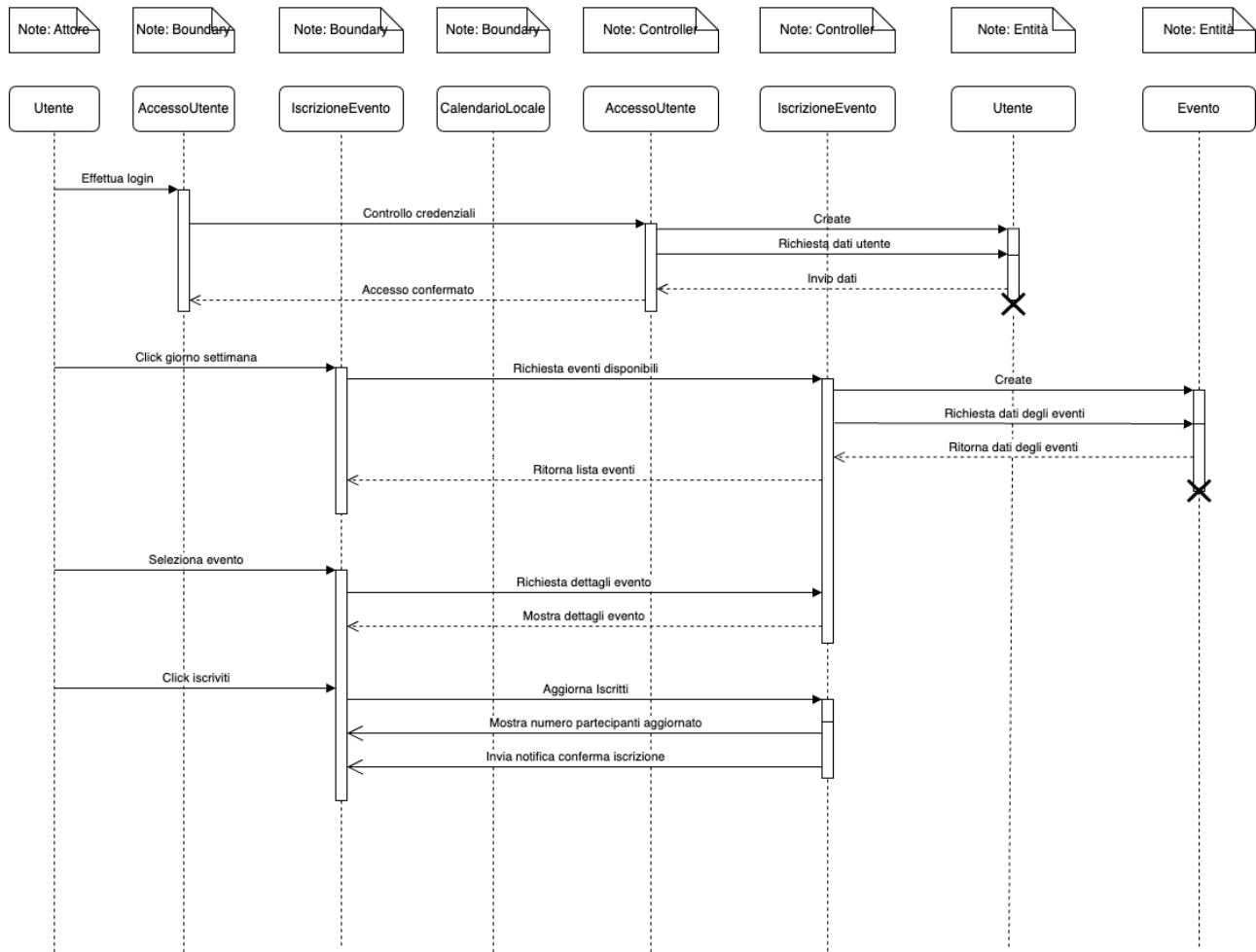


Figura 9: Sequence Diagram per l'iscrizione all'evento

Gestione Tracker Spirituale

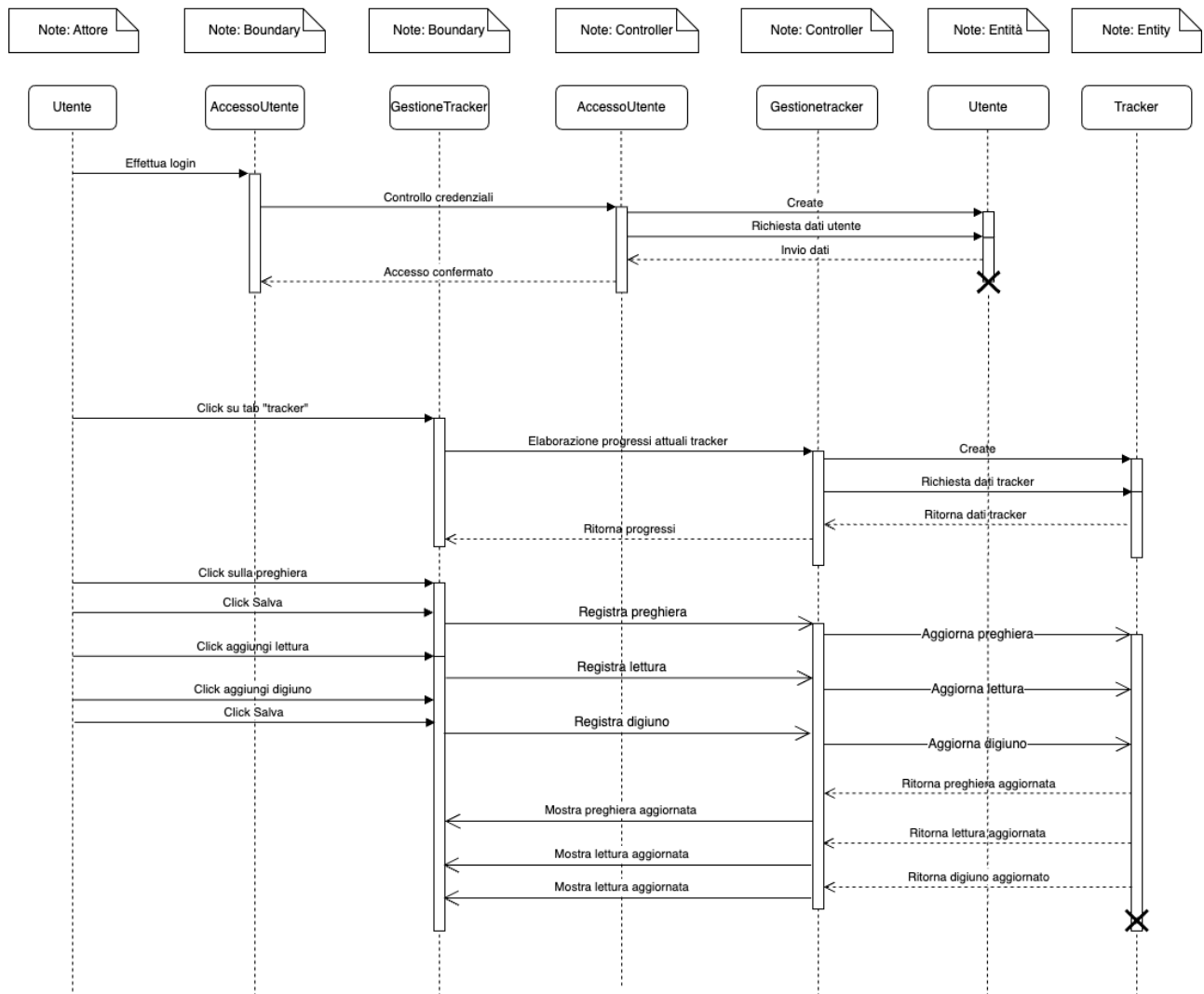
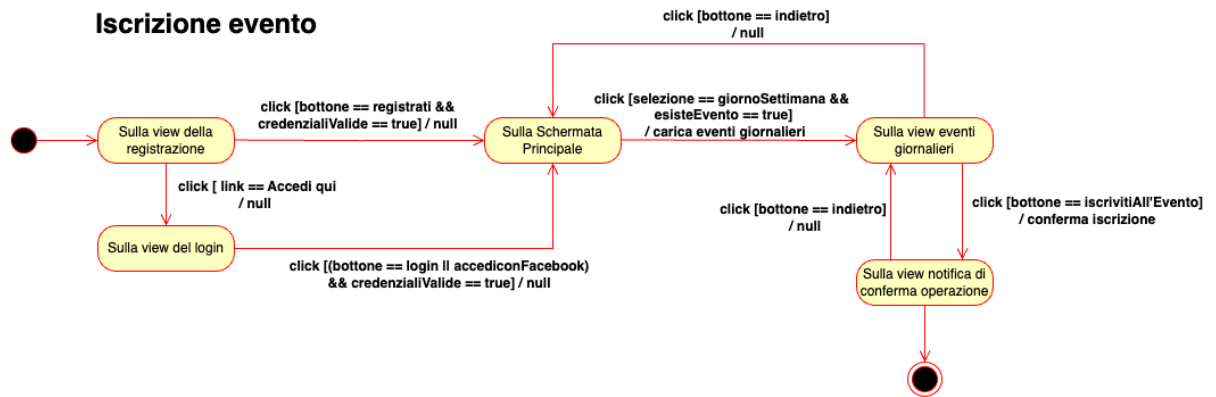


Figura 10: Sequence Diagram per la gestione del Tracker Spirituale

3.5 State Diagram

In conclusione, mostriamo gli State Diagram riguardanti il processo di iscrizione a un evento e la gestione degli eventi. Questi diagrammi rappresentano le transizioni di stato delle interfacce grafiche in risposta alle azioni dell'utente.

Iscrizione evento



Gestione evento

