

# Analisi e progettazione

Progetto	VolontariON	
Gruppo	T01	
Documento	D2	rev: 1.2

## Indice

<b>Indice</b>	<b>1</b>
Scopo del documento	3
<b>Use Case diagrams</b>	<b>3</b>
Login e Registrazione	3
Requisiti	4
Use case RF1 : Effettuare login	4
Use case RF2 : Effettuare Registrazione	4
Modifica del profilo, eliminazione account e modifica password	6
Requisiti	6
Use case RF3 : Eliminazione Account	7
Use case RF4-RF5 : Modifica del profilo e Modifica password	8
Creazione, eliminazione e modifica evento	9
Requisiti	9
Use case RF6.1 : Creazione evento	10
Use case RF6.2 : Eliminazione evento	10
Use case RF6.4 : Modifica evento	11
Visualizzazione Eventi, Ricerca Eventi e Visualizzazione profilo associazione	12
Requisiti	12
Use case RF6.3 : Visualizza evento	13
Use case RF6.5 : Visualizza lista eventi	13
Use case RF6.6 : Visualizza eventi di un'associazione	14

Use case RF9 : Visualizza Profilo associazione	14
Use case RF13 : Ricerca evento	14
Iscrizione Evento, Profilo Volontario	16
Requisiti	17
Use case RF7.1 : Iscrizione ad un evento	17
Use case RF7.2 : Visualizzazione lista eventi a cui si è iscritto	18
Use case RF7.3 : Disiscrizione evento	18
Use case RF7.4 : Visualizzazione iscritti a un evento	19
Use case RF10 : Visualizzazione profilo volontario	19
Seguire associazioni, liste associazioni	20
Requisiti	21
Use case RF8 : Visualizzazione lista delle associazioni registrate	21
Use case RF11.1 : Seguire associazioni	22
Use case RF11.2 : Visualizzazione lista di associazioni seguite	22
Use case RF11.3 : Visualizzazione lista di eventi creati da associazioni seguite	23
Use case RF11.4 : Unfollow associazione	23
Use case RF12 : Ricerca associazione	24
Diagramma dei componenti	25
Componente 1: Login	25
Componente 2: Registrazione	26
Componente 3: Gestione Profilo	26
Componente 4: Gestione Database	27
Componente 5: Gestione Volontari	28
Componente 6: Gestione Eventi	28
Componente 7: Gestione Associazioni	29
Componente 8: Gestione Ricerca	30
Componente 9: Gestore Email	30
Class diagram	31
Utente non Autenticato	32
Utente Autenticato	32
Volontario	33
Associazione	34
Registrazione	35
Autenticazione	35
Email	36

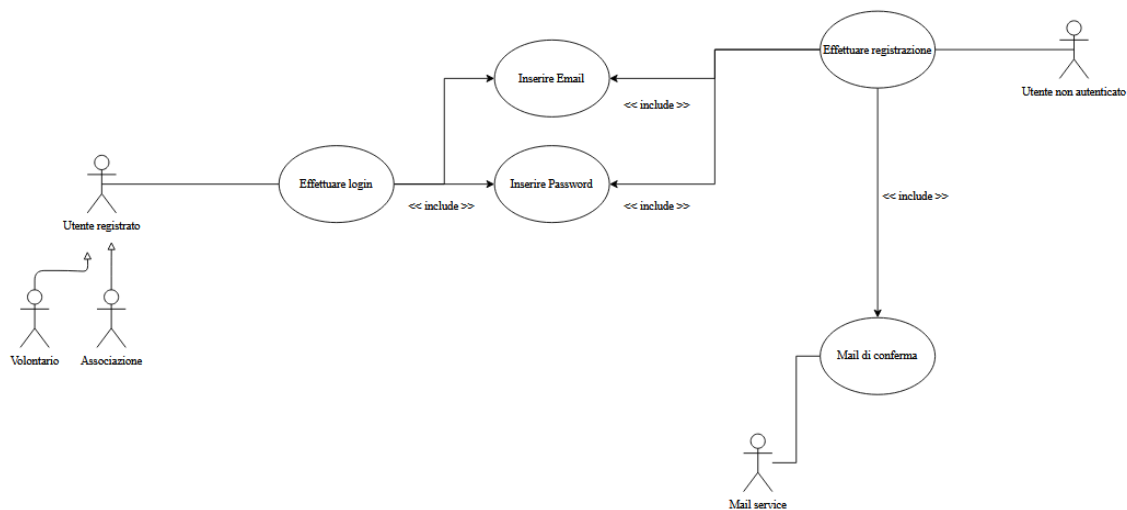
Profilo	36
Evento	37
Classi di supporto	38
Dal class diagram alle API	39

## Scopo del documento

Il presente documento ha l'obiettivo di fornire una specifica formale e dettagliata dei requisiti funzionali definiti nel Documento di Analisi dei Requisiti (D1), utilizzando sia il linguaggio naturale che UML. In particolare, verranno utilizzati diagrammi dei casi d'uso, diagrammi dei componenti e diagrammi delle classi. Segue poi la mappatura dei metodi delle classi con gli endpoint dell'API

## Use Case diagrams

### Login e Registrazione



## Requisiti

### RF1 : Login

### RF2: Registrazione

## Use case RF1 : Effettuare login

### Riassunto:

Questo use case descrive come un utente non autenticato effettua il login sulla piattaforma.

### Descrizione:

1. L'utente visualizza una schermata con un form nella quale sono presenti i campi "Email" e "Password", un bottone "Login" e un bottone per scegliere il tipo di account (se volontario o se associazione).
2. L'utente inserisce negli appositi campi la sua email e password e clicca sul bottone "Login". Se le informazioni inserite sono corrette (vi è una corrispondenza nel database), il login è stato effettuato correttamente e la pagina principale della piattaforma viene visualizzata [ Eccezione 1 ]

### Eccezioni:

1. Se l'utente inserisce una email o password errata ( l'email inserita non è stata trovata nel database / la password inserita non corrisponde a quella associata alla mail), l'applicazione restituisce un messaggio di errore e chiede di effettuare nuovamente il login.

## Use case RF2 : Effettuare Registrazione

### Riassunto:

Questo use case descrive come un utente effettua la registrazione sulla piattaforma.

**Descrizione:**

1. L'utente anonimo visualizza una schermata nella quale sceglie se vuole registrarsi come volontario o associazione.
2. Dopo la scelta, viene visualizzato un'altra schermata con un form e un bottone "Registrati". Le informazioni richieste dal form dipenderanno dalla scelta effettuata dall'utente anonimo.
3. Per un volontario è richiesto nome,cognome, email, password (obbligatori), numero di telefono , immagine del profilo, descrizione di sé, esperienze passate, competenze (opzionali).

Per un'associazione è richiesto il nome dell'associazione, email,password (obbligatori), numero di telefono, immagine del profilo/logo , descrizione dell'associazione, obiettivi dell'associazione (opzionali) . [ Eccezione 1 ][ Estensione 1 ][ Eccezione 2 ]

4. A questo punto, dopo aver premuto il pulsante "Registrati", l'utente anonimo è registrato sulla piattaforma. Viene inviata una mail di conferma di avvenuta registrazione. l'utente può effettuare il login con le credenziali appena inserite.

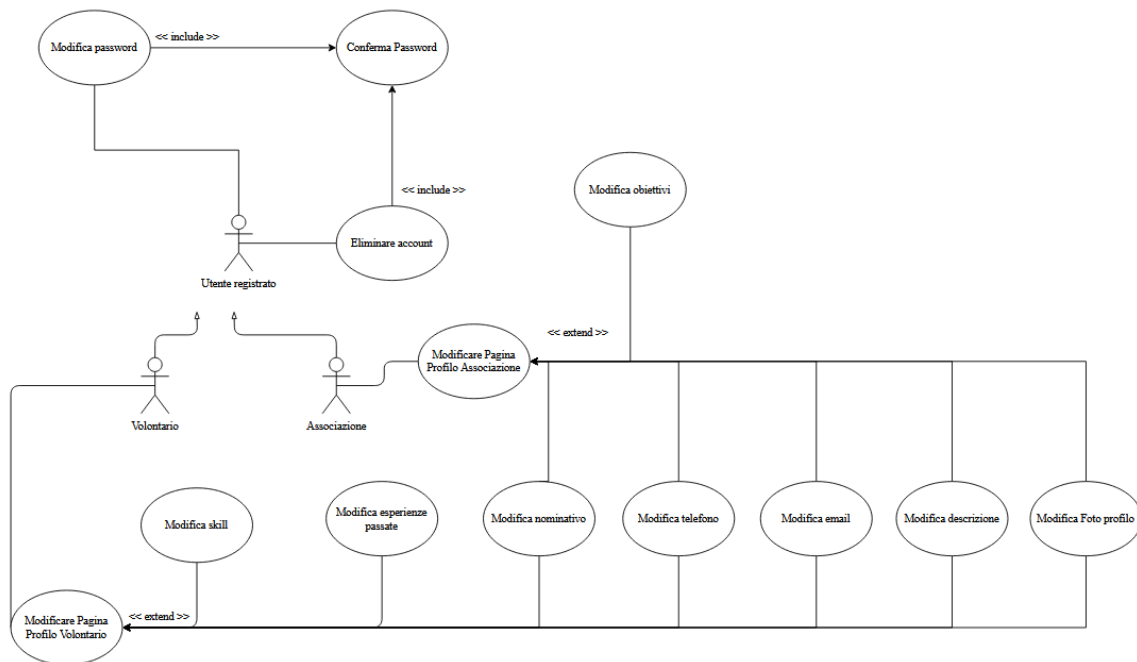
**Estensioni:**

1. Le email devono essere univoche per tipologia di account ovvero due o più associazioni non possono avere la stessa email e due o più volontari non possono avere la stessa email, mentre un'associazione e un volontario possono avere la stessa email.

**Eccezioni:**

1. Se un campo obbligatorio non è compilato o non è corretto (es: l'email non ha il formato giusto) viene visualizzato un messaggio di errore e viene chiesto di reinserire le informazioni correttamente.
2. In relazione all'Estensione 1, se l'email inserita appartiene già a un'altro account viene visualizzato un messaggio di errore e viene chiesto di inserire un'email diversa

## Modifica del profilo, eliminazione account e modifica password



## Requisiti

**RF3 : Eliminazione account**

**RF4 : Modifica del profilo**

**RF5 : Modifica password**

## Use case RF3 : Eliminazione Account

### Riassunto:

Questo use case descrive il processo tramite il quale un utente registrato può eliminare il proprio account dalla piattaforma.

### Descrizione:

1. L'utente accede alla sezione "Profilo" la quale contiene il bottone "Elimina Account".
2. Una volta cliccato, viene visualizzato un form di conferma contenente un campo "Password" e i bottoni "Conferma" e "Annulla".
3. Se l'utente completa il campo "Password" e clicca su "Conferma" il sistema avvia il processo di eliminazione che include: La rimozione di tutti i dati associati all'utente (informazioni personali, eventi, associazioni seguite, etc.). [ Eccezione 1]
4. Se l'utente clicca su "Annulla" il processo di eliminazione viene annullato e l'utente viene riportato alla schermata precedente.

### Eccezioni

1. Se il campo "password" è vuoto o non combacia con la password dell'account viene visualizzato un messaggio di errore e viene chiesto di reinserire la password.

## Use case RF4-RF5 : Modifica del profilo e Modifica password

### Riassunto:

Questo use case descrive come un utente registrato può modificare i dati del suo profilo.

### Descrizione:

1. L'utente accede alla sua sezione "Profilo" dove sono presenti i dati inseriti in fase di registrazione (nome, email, cellulare, descrizione,...) a seconda del tipo di account (Volontario/Associazione) e un bottone "Modifica Profilo" e un bottone "Modifica password".
2. Se l'utente clicca il bottone "Modifica Profilo" viene visualizzato un form nella quale è possibile modificare i dati del profilo.
3. Se l'utente clicca il bottone "Modifica password" viene visualizzato un form con i campi "Vecchia password", "Nuova password" e "Conferma nuova password". È quindi necessario immettere la vecchia password e che gli ultimi due campi combaciano.

[ Eccezione 2 ][ Eccezione 3 ]

4. Nei form di entrambi i casi è presente un bottone "Conferma" per finalizzare le modifiche e un bottone "Annulla" per annullare il processo di modifica.
5. Dopo aver cliccato sul bottone "Conferma", le modifiche vengono salvate permanentemente. [ Eccezione 1 ]

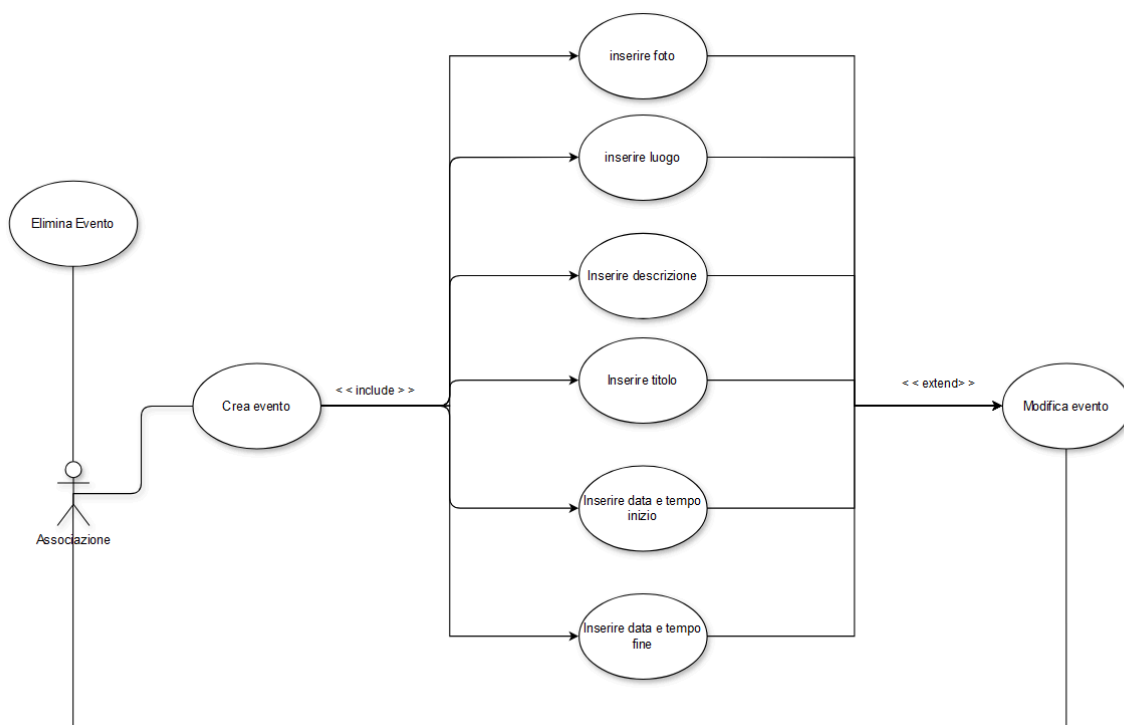
### Eccezioni

1. Se i campi del form sono vuoti o non corretti e viene cliccato il bottone "Conferma" viene visualizzato un messaggio di errore e viene chiesto di completare correttamente i campi.
2. Se il contenuto del campo "Vecchia password" non combacia con la vera password dell'account viene visualizzato un messaggio di errore e viene chiesto di inserire correttamente la vecchia password



3. Se il contenuto dei campi "Nuova password" e "Conferma nuova password" non combaciano viene visualizzato un messaggio di errore e viene chiesto di completare correttamente i due campi

## Creazione, eliminazione e modifica evento



## Requisiti

**RF6.1: Creazione evento**

**RF6.2: Eliminazione evento**

**RF6.4: Modifica evento**

## Use case RF6.1 : Creazione evento

**Riassunto:**

Questo use case descrive come un'associazione può creare un evento

**Descrizione:**

1. L'associazione accede alla sua sezione di gestione dei suoi eventi.
2. Preme sul bottone "Crea evento".
3. Viene visualizzato un form che contiene i bottoni "Crea evento" e "Annulla" e i seguenti campi: nome dell'evento, data e tempo di inizio, data e tempo di fine, luogo/indirizzo dove si terrà l'evento (obbligatori), foto e descrizione (opzionali).
4. Una volta che l'associazione completa i campi e clicca "Crea evento", l'evento viene creato [ Eccezione 1]
5. Il bottone "Annulla" cancella il processo di creazione evento e riporta l'utente alla schermata precedente.

**Eccezioni**

1. Se un campo obbligatorio è mancante o uno o più campi sono errati (es : data di fine che precede la data di inizio) viene visualizzato un messaggio di errore e viene chiesto di completare correttamente il form.

## Use case RF6.2 : Eliminazione evento

**Riassunto:**

Questo use case descrive come un'associazione può eliminare un suo evento

**Descrizione:**

1. L'associazione accede alla sua sezione di gestione dei suoi eventi la quale contiene la lista di tutti gli eventi creati dall'associazione.
2. Per ogni evento è presente l'opzione "Elimina evento".
3. Se viene cliccato, tale evento viene eliminato dalla piattaforma.

## Use case RF6.4 : Modifica evento

### Riassunto:

Questo use case descrive come un'associazione può modificare un suo evento

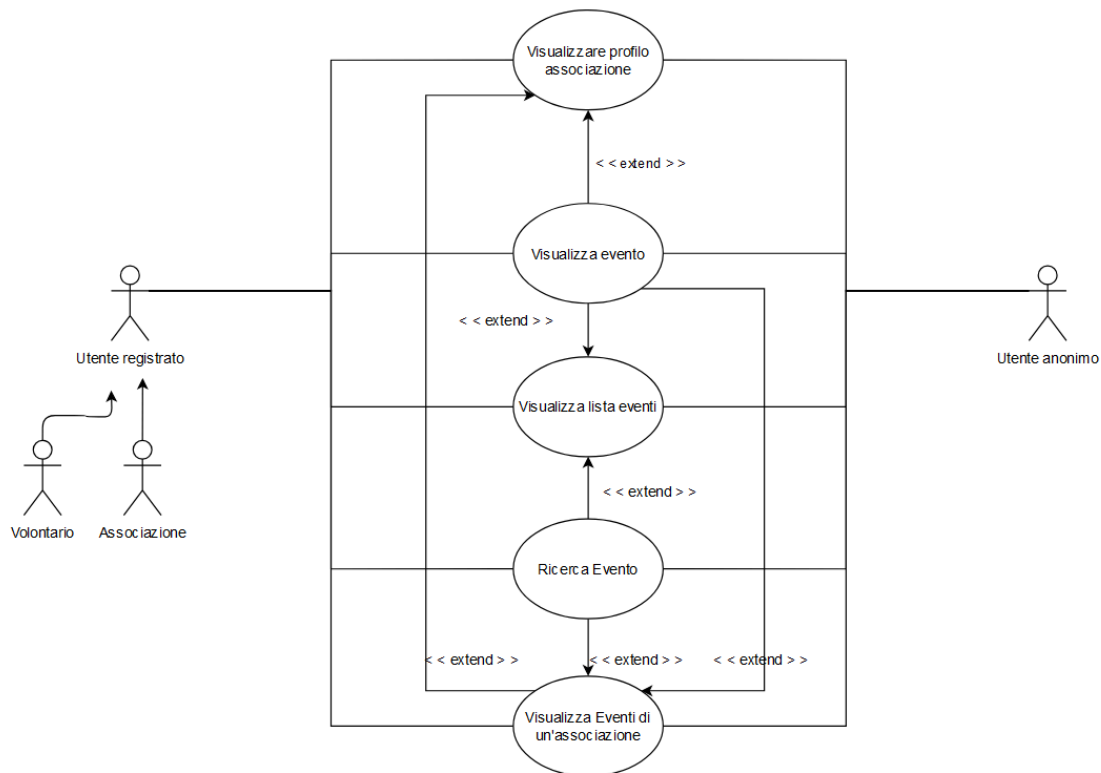
### Descrizione:

1. L'associazione accede alla sua sezione di gestione dei suoi eventi "I miei eventi" la quale contiene la lista di tutti gli eventi creati dall'associazione.
2. Per ogni evento è presente l'opzione "Modifica evento".
3. Se cliccato, viene visualizzato un form che contiene i bottoni "Modifica evento" e "Annulla" i seguenti campi: nome dell'evento, data e tempo di inizio, data e tempo di fine, luogo/indirizzo dove si terrà l'evento, foto e descrizione.
4. Una volta che l'associazione completa i campi e clicca "Modifica evento", l'evento viene modificato[ Eccezione 1]
5. Il bottone "Annulla" cancella il processo di modifica dell'evento e riporta l'utente alla schermata precedente.

### Eccezioni

1. Se uno o più campi sono errati (es : data di fine che precede la data di inizio) viene visualizzato un messaggio di errore e viene chiesto di completare correttamente il form.

## Visualizzazione Eventi, Ricerca Eventi e Visualizzazione profilo associazione



## Requisiti

**RF6.3: Visualizzazione evento**

**RF6.5: Visualizzazione lista eventi**

**RF6.6: Visualizzazione eventi di un'associazione**

**RF9: Visualizzazione profilo associazione****RF13: Ricerca Evento**

## **Use case RF6.3 : Visualizza evento**

**Riassunto:**

Questo use case descrive come un utente può visualizzare un evento

**Descrizione:**

1. L'utente sceglie l'evento che intende visualizzare.
2. Premendo il bottone "Visualizza" esso viene mandato alla pagina dell'evento.
3. La pagina contiene tutte le informazioni riguardanti l'evento: nome dell'evento, data e tempo di inizio, data e tempo di fine, luogo/indirizzo dove si terrà l'evento, foto e descrizione.
4. L'utente se vuole può dirigersi al profilo dell'associazione tramite un link (Use Case visualizza profilo associazione)

## **Use case RF 6.5 : Visualizza lista eventi**

**Riassunto:**

Questo use case descrive come un utente può visualizzare la lista di tutti gli eventi presenti sulla piattaforma

**Descrizione:**

1. L'utente si dirige verso la pagina "Lista eventi".
2. Vengono visualizzate tutti gli eventi caricati sul sistema
3. L'utente scelto un evento può cliccare su "Visualizza" per dirigersi alla pagine dell'evento (Use Case "Visualizza Evento").

## Use case RF6.6 : Visualizza eventi di un'associazione

### Riassunto:

Questo use case descrive come un utente può visualizzare la lista di tutti gli eventi appartenenti a un'associazione.

### Descrizione:

1. L'utente sceglie l'associazione della quale visualizzare gli eventi.
2. Cliccando su "Eventi" vengono presentati tutti gli eventi creati da tale associazione.
3. l'utente scelto un evento può cliccare su "Visualizza" per dirigersi alla pagina dell'evento (Use Case "Visualizza Evento").

## Use case RF9 : Visualizza Profilo associazione

### Riassunto:

Questo use case descrive come un utente può visualizzare il profilo di un'associazione

### Descrizione:

1. L'utente sceglie l'associazione della quale vuole visualizzare il profilo.
2. Vengono fornite tutte le informazioni inerenti l'associazione :nome dell'associazione, email, numero di telefono, immagine del profilo/logo , descrizione dell'associazione e obiettivi dell'associazione.
3. L'utente inoltre può scegliere di visualizzare tutti gli eventi creati da tale associazione (Use Case Visualizza Eventi di un'associazione).

## Use case RF13 : Ricerca evento

### Riassunto:

Questo use case descrive come un utente può cercare uno specifico evento da una lista di eventi e filtrare in base alla data di inizio e fine

### Descrizione:

1. L'utente inserisce in una barra di ricerca il nome dell'evento che vuole cercare
2. La lista di eventi viene aggiornata in base al contenuto della barra di ricerca.
3. L'utente può inoltre filtrare gli eventi in base al fatto che siano eventi prossimi (data attuale < data di inizio), passati (data di fine < data attuale). [ Estensione 1]

### Estensione:

1. Se il risultato della ricerca non produce nessun risultato, viene visualizzato il messaggio "Nessun evento trovato"

## Iscrizione Evento, Profilo Volontario





## Requisiti

**RF7.1: Iscrizione ad un evento**

**RF7.2: Disiscrizione da un evento**

**RF7.3: Visualizzazione lista eventi a cui un volontario si è iscritto**

**RF7.4: Visualizzazione iscritti a un evento**

**RF10: Visualizzazione profilo Volontario**

## Use case RF7.1 : Iscrizione ad un evento

### Riassunto:

Questo use case descrive come un volontario può iscriversi ad un evento.

### Descrizione:

1. L'utente visualizza la pagina dell'evento scelto.
2. Nella pagina è presente un bottone "Iscriviti".
3. L'utente clicca sul bottone.
4. Il sistema registra l'iscrizione dell'utente all'evento.
5. Il sistema conferma l'avvenuta iscrizione con un messaggio di successo. [ Estensione 1]

### Estensione:

1. Se il volontario è già iscritto all'evento, viene visualizzato un messaggio informativo "Sei già iscritto"

## Use case RF7.2 : Visualizzazione lista eventi a cui si è iscritto

### Riassunto:

Questo use case descrive come un volontario può visualizzare tutti gli eventi a cui si è iscritto

### Descrizione:

1. Il volontario si dirige alla sua pagina "I miei eventi".
2. Il sistema recupera la lista degli eventi a cui il volontario è iscritto.
3. Viene visualizzato la lista degli eventi con le seguenti informazioni per ciascun evento: foto, nome, data e ora di inizio e fine, luogo.
4. Il volontario può selezionare un evento e visualizzarne la pagina ( use case Visualizza Evento)
5. Il volontario può cercare eventi per nome e filtrare in base alla data ( use case Ricerca Evento)

## Use case RF7.3 : Disiscrizione evento

### Riassunto:

Questo use case descrive come un utente può disiscriversi da un evento.

### Descrizione:

1. Il volontario si dirige alla sua pagina "I miei eventi".
2. Il sistema recupera la lista degli eventi a cui il volontario è iscritto.
3. Viene visualizzato la lista degli eventi con le seguenti informazioni per ciascun evento: nome, data e ora di inizio e fine, luogo.
4. Ogni evento ha un bottone "Disiscriviti".

5. Una volta cliccato il sistema disiscrive il volontario dall'evento.
6. Un messaggio informativo di avvenuta disiscrizione viene visualizzato.

## Use case RF7.4 : Visualizzazione iscritti ad un evento

### Riassunto:

Questo use case descrive come un'associazione può visualizzare tutti i volontari che si sono iscritti a un suo evento

### Descrizione:

1. L'associazione accede alla sua sezione di gestione dei suoi eventi "I miei eventi" la quale contiene la lista di tutti gli eventi creati dall'associazione.
2. Il sistema recupera la lista di tutti eventi creati dall'associazione.
3. Viene visualizzato la lista degli eventi con le seguenti informazioni per ciascun evento: foto, nome, data e ora di inizio e fine, luogo.
4. Ogni evento ha un bottone "Visualizza iscritti".
5. Una volta cliccato il sistema visualizza la lista di tutti i volontari iscritti a tale evento contenente il nome,cognome,foto profilo, email e cellulare di ogni volontario.
6. Scelto un volontario, l'associazione può cliccare su "Visualizza volontario" per dirigersi alla pagina profilo di tale volontario. (Use case "Visualizzazione profilo volontario")

## Use case RF10 : Visualizzazione profilo volontario

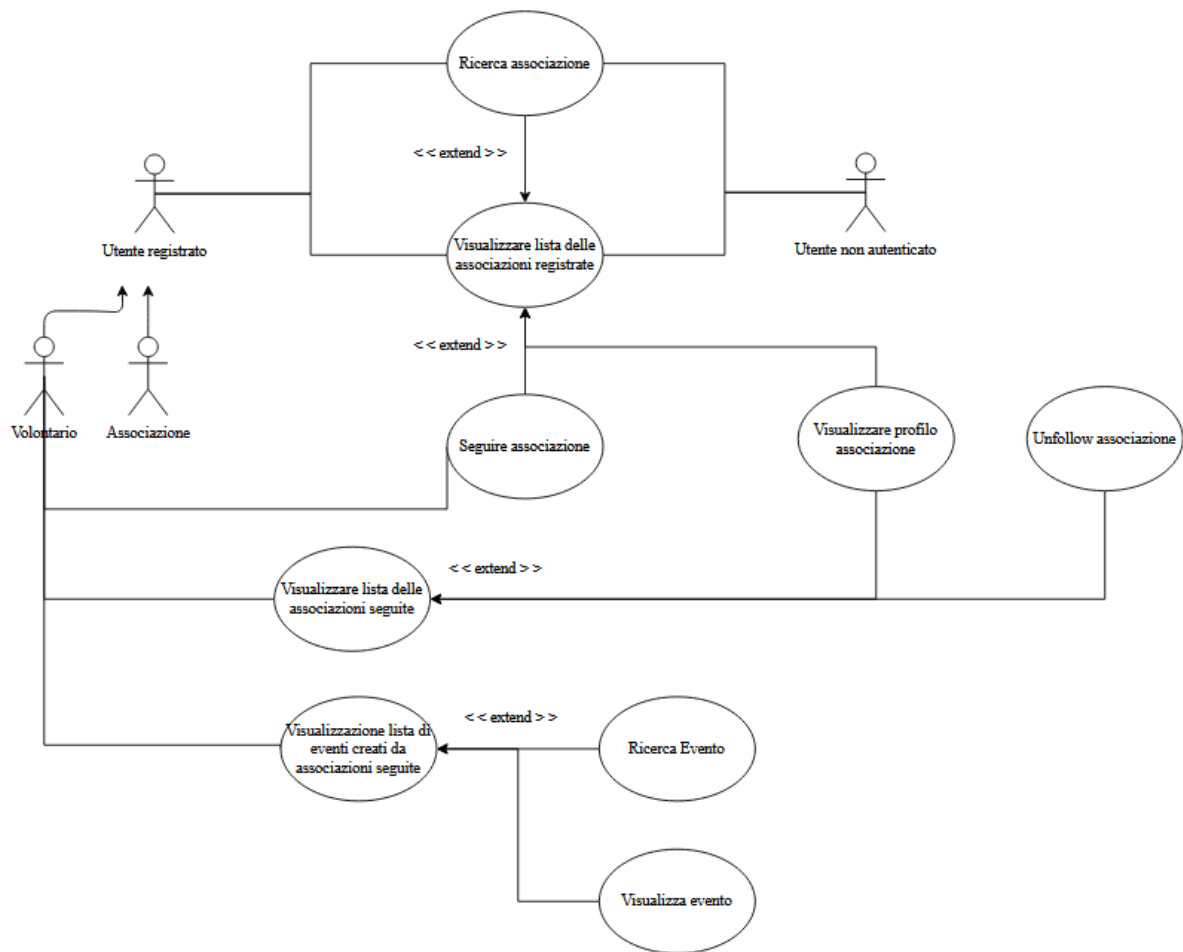
### Riassunto:

Questo use case descrive come un'associazione può visualizzare il profilo di un volontario che si è iscritto a un suo evento

**Descrizione:**

1. L'associazione accede alla lista degli iscritti a un evento.
2. L'associazione sceglie un volontario dall'elenco e clicca sul bottone "Visualizza".
3. L'associazione viene reindirizzata alla pagina profilo del volontario che contiene le seguenti informazioni: foto profilo, Nome, cognome, email, telefono, Esperienza e competenze.

## Seguire associazioni, liste associazioni



## Requisiti

**RF8 : Visualizzazione lista delle associazioni registrate**

**RF11.1 : Seguire associazioni**

**RF11.2 : Visualizzazione lista di associazioni seguite**

**RF11.3 : Visualizzazione lista di eventi creati da associazioni seguite**

**RF11.4 : Unfollow associazione****RF12 : Ricerca associazione**

## **Use case RF8 : Visualizzazione lista delle associazioni registrate**

**Riassunto:**

Questo use case descrive come un utente può visualizzare tutte le associazioni che si sono registrate sulla piattaforma.

**Descrizione:**

1. L'utente si dirige alla sezione "Associazioni".
2. Il sistema recupera la lista di tutte le associazioni che si sono registrate sulla piattaforma.
3. Viene visualizzato la lista delle associazioni con le seguenti informazioni per ciascuna associazione: foto profilo, nome, email, telefono.
4. Ogni associazione ha un bottone "Visualizza" che indirizza l'utente alla pagina profilo dell'associazione (Use case "Visualizzare profilo associazione")

## **Use case RF11.1 : Seguire associazioni**

**Riassunto:**

Questo use case descrive come un volontario può seguire un associazione

**Descrizione:**

1. Il volontario visita la pagina profilo di un'associazione.
2. Cliccando il bottone "Segui",
3. Il sistema registra l'associazione tra quelle seguite dall'utente.
4. Il sistema conferma l'azione con un messaggio di successo [Estensione 1]

**Estensioni**

1. Se il volontario sta già seguendo l'associazione, viene visualizzato un messaggio informativo "Stai già seguendo".

## **Use case RF11.2 : Visualizzazione lista di associazioni seguite**

**Riassunto:**

Questo use case descrive come un volontario può visualizzare tutte le associazioni che sta seguendo

**Descrizione:**

1. Il volontario si dirige alla sezione "Associazioni seguite"
2. Il sistema recupera la lista di tutte le associazioni che l'utente sta attualmente seguendo.
3. Viene visualizzato la lista delle associazioni con le seguenti informazioni per ciascuna associazione: foto, nome, data e ora di inizio e fine, luogo.
4. Ogni evento ha un bottone "Visualizza" che indirizza l'utente alla pagina profilo dell'evento (Use case "Visualizzare profilo associazione")

## Use case RF11.3 : Visualizzazione lista di eventi creati da associazioni seguite

### Riassunto:

Questo use case descrive come un volontario può visualizzare tutti gli eventi creati da associazioni che segue.

### Descrizione:

1. Il volontario si dirige alla sezione "Eventi associazioni iscritte"
2. Il sistema recupera la lista di tutti gli eventi creati da associazioni che l'utente sta attualmente seguendo.
3. Viene visualizzato la lista degli eventi con le seguenti informazioni per ciascun evento: foto profilo,nome,email,telefono.
4. Ogni associazione ha un bottone "Visualizza" che indirizza l'utente alla pagina dell'evento (Use case "Visualizza Evento")

## Use case RF11.4 : Unfollow associazione

### Riassunto:

Questo use case descrive come un volontario smettere di seguire un associazione che sta al momento seguendo.

### Descrizione:

1. Il volontario si dirige alla sezione "Associazioni seguite"
2. Il sistema recupera la lista di tutte le associazioni che l'utente sta attualmente seguendo.
3. Viene visualizzato la lista delle associazioni con le seguenti informazioni per ciascuna associazione:foto, nome,data e ora di inizio e fine,luogo.



4. Ogni evento ha un bottone "Unfollow".
5. Una volta cliccato il sistema registra che il volontario non sta più seguendo l'associazione.
6. Un messaggio di successo viene visualizzato che conferma l'operazione.

## Use case RF12 : Ricerca associazione

### Riassunto:

Questo use case descrive come un utente può cercare un'associazione registrata sulla piattaforma per nome.

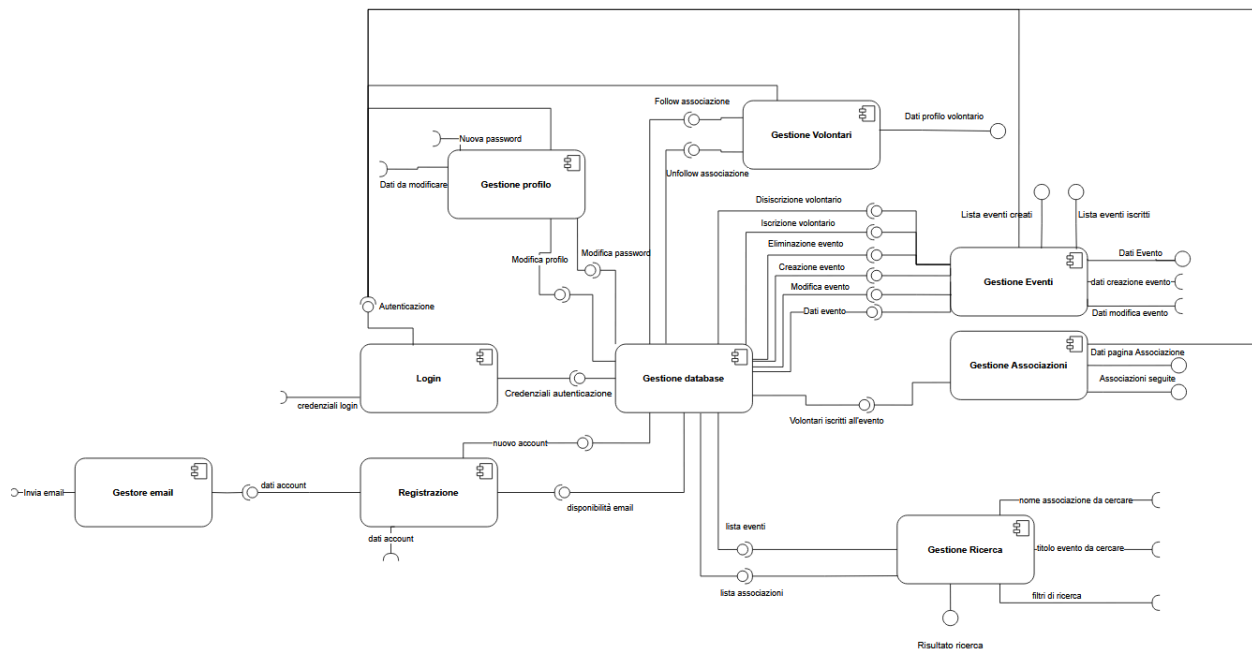
### Descrizione:

1. L'utente inserisce in una barra di ricerca il nome dell'associazione che vuole cercare
2. La lista di associazioni risultanti viene aggiornata in base al contenuto della barra di ricerca. [ Estensione 1]

### Estensione:

1. Se il risultato della ricerca non produce nessun risultato, viene visualizzato il messaggio "Nessun associazione trovata"

## Diagramma dei componenti



## Componente 1: Login

**Descrizione:** Gestisce l'autenticazione degli utenti, consentendo l'accesso alla piattaforma. Permette all'utente di scegliere se loggarsi come volontario o utente.

### Interfacce fornite:

- Autenticazione: Fornisce il token di autenticazione dell'utente a altri componenti.

### Interfacce richieste:

- Credenziali login: Riceve le credenziali (email e password ) dell'utente per effettuare il login.

- Credenziali autenticazione : recupera le credenziali dal database per verificarne la correttezza

## Componente 2: Registrazione

**Descrizione:** Permette la creazione di un nuovo account per volontari o associazioni.

**Interfacce fornite:**

- Dati account: Invia i dettagli del nuovo utente al gestore email.
- Nuovo account : invia i dettagli del nuovo utente al database per salvarne i dati

**Interfacce richieste:**

- Dati account: Riceve i dati dell'utente per la registrazione.
- Disponibilità email: Verifica se un'email è già utilizzata.

## Componente 3: Gestione Profilo

**Descrizione:** Consente agli utenti di modificare le informazioni del proprio profilo e di cambiare la password.

**Interfacce fornite:**

- Modifica profilo: Invia le modifiche al database per salvare i dati.
- Modifica password : invia la nuova password al database.

**Interfacce richieste:**

- Nuova password: Riceve la nuova password per aggiornare le credenziali.
- Dati da modificare: Riceve le modifiche da apportare.
- Autenticazione: Riceve Il token di autenticazione dell'utente.

## Componente 4: Gestione Database

**Descrizione:** Gestisce la memorizzazione e il recupero dei dati relativi a utenti, eventi e associazioni, iscrizioni e follows.

### Interfacce fornite:

- Lista eventi : Fornisce l'elenco degli eventi.
- Dati evento : Fornisce i dettagli di un evento.
- Credenziali autenticazione : fornisce le credenziali di accesso per verificarne la correttezza.
- Lista associazioni: Fornisce l'elenco delle associazioni registrate.
- Volontari iscritti all'evento: Fornisce i dati sui volontari iscritti a un evento.
- Disponibilità email : Verifica se un'email è già utilizzata.

### Interfacce richieste:

- Iscrizione volontario: Registra un volontario a un evento.
- Disiscrizione volontario: Rimuove un volontario da un evento.
- Creazione evento: Registra un nuovo evento.
- Modifica evento: Aggiorna i dettagli di un evento.
- Eliminazione evento: Cancella un evento.
- Follow associazione: Riceve l'operazione di follow e lo memorizza.
- Unfollow associazione: Riceve l'operazione di unfollow e lo memorizza.
- Modifica profilo: riceve le modifiche del profilo.
- Modifica password :riceve la nuova password.
- Nuovo account : riceve i dettagli del nuovo utente appena registrato.

## Componente 5: Gestione Volontari

**Descrizione:** Gestisce le azioni disponibili per i volontari, come seguire un'associazione o iscriversi agli eventi.

### Interfacce fornite:

- Dati profilo volontario: Fornisce informazioni sul volontario.

### Interfacce richieste:

- Follow associazione: Permette al volontario di seguire un'associazione.
- Unfollow associazione: Permette di smettere di seguire un'associazione.
- Autenticazione: Riceve Il token di autenticazione dell'utente.

## Componente 6: Gestione Eventi

**Descrizione:** Si occupa della creazione, modifica, eliminazione, visualizzazione degli eventi e delle iscrizioni agli eventi sulla piattaforma.

### Interfacce fornite:

- Lista eventi iscritti: Restituisce l'elenco degli eventi a cui un volontario è iscritto.
- Lista eventi creati: Restituisce l'elenco degli eventi che l'associazione ha creato.
- Dati evento: Restituisce i dettagli di un evento specifico (titolo, descrizione, luogo, data, immagine, organizzatore).
- Creazione evento: restituisce l'evento appena creato.
- Modifica evento: restituisce le modifiche di un evento da salvare.

- Eliminazione evento: Richiede l'eliminazione di un evento dalla piattaforma.
- Iscrizione volontario: Riceve la richiesta di iscrizione di un volontario a un evento.
- Disiscrizione volontario: Riceve la richiesta di un volontario di annullare la propria iscrizione a un evento.

**Interfacce richieste:**

- Dati creazione evento: Riceve le informazioni per la creazione di un nuovo evento.
- Dati modifica evento: Riceve le informazioni aggiornate per modificare un evento esistente.
- Autenticazione: Riceve Il token di autenticazione dell'utente.

**Componente 7: Gestione Associazioni**

**Descrizione:** Permette di visualizzare informazioni sulle associazioni, le associazioni seguite da un volontario e gli iscritti a un evento.

**Interfacce fornite:**

- Dati pagina associazione: Fornisce i dettagli di un'associazione.
- Associazioni seguite: Fornisce l'elenco delle associazioni seguite dai volontari.

**Interfacce richieste:**

- Volontari iscritti all'evento: Riceve la lista di volontari iscritti a un evento.
- Autenticazione: Riceve Il token di autenticazione dell'utente.

## Componente 8: Gestione Ricerca

**Descrizione:** Permette di cercare associazioni ed eventi in base a diversi criteri.

**Interfacce fornite:**

- Risultato ricerca:Fornisce i risultati delle ricerche effettuate.

**Interfacce richieste:**

- Nome dell'associazione da cercare:Permette di cercare un'associazione specifica per nome.
- Titolo evento da cercare:Permette di cercare un evento specifico per titolo.
- Filtri di ricerca:Permette di applicare criteri di filtraggio alla ricerca come la data degli eventi.
- Lista eventi : Riceve l'elenco degli eventi.
- Lista associazioni: Riceve l'elenco delle associazioni registrate.

## Componente 9: Gestore mail

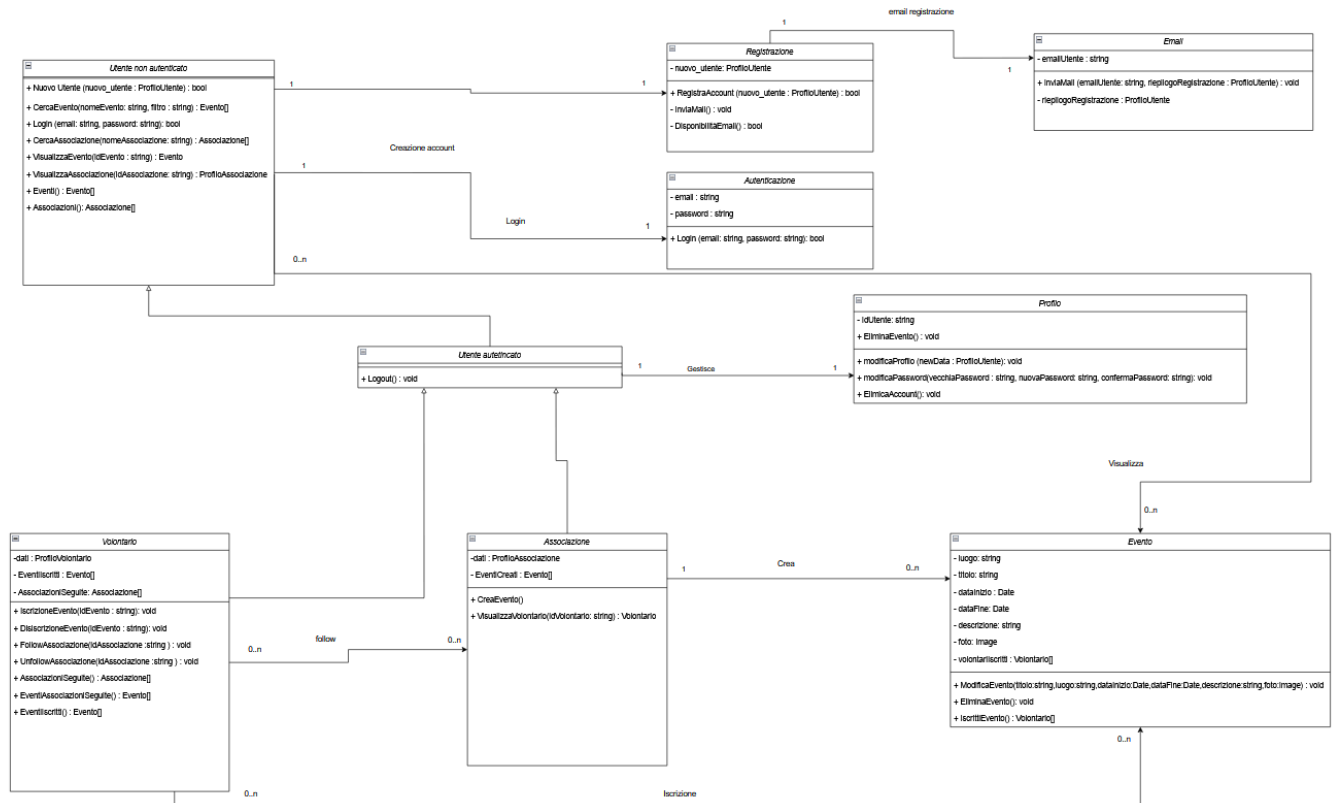
**Descrizione:** Gestisce l'invio di email per conferma di registrazione.

**Interfacce richieste:**

- Dati account: riceve le informazioni dell'account appena creato

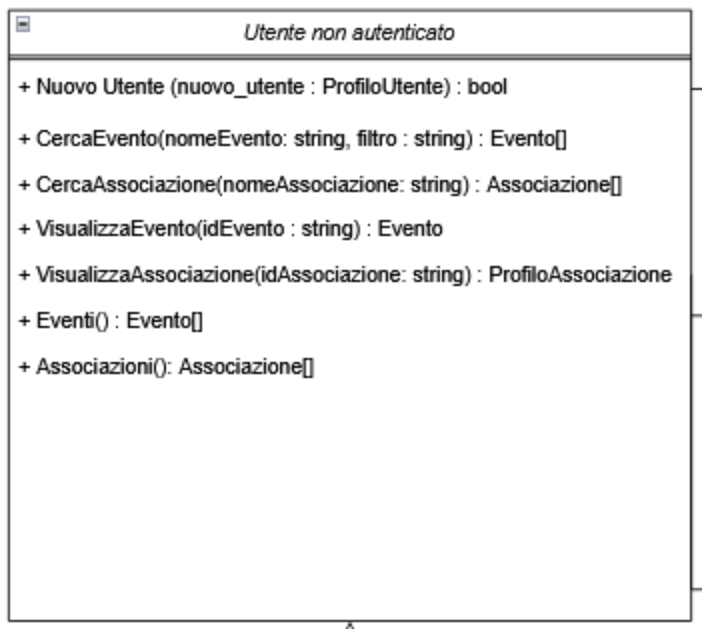
**Interfacce fornite:**

- Invia email:Manda l'email di avvenuta registrazione.





## Utente non Autenticato



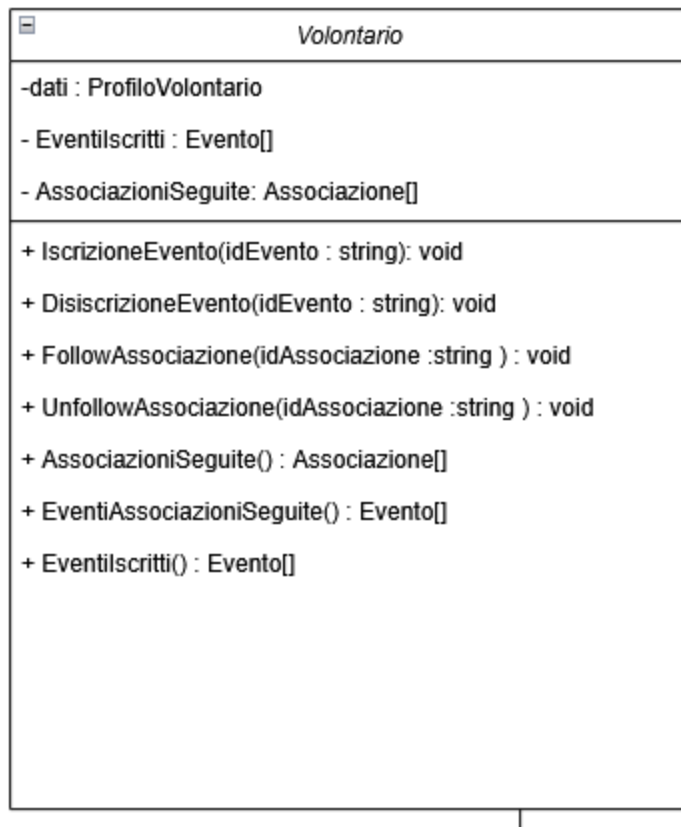
Rappresenta un utente che non ha effettuato l'accesso al sistema.

## Utente Autenticato

Rappresenta un utente che ha effettuato l'accesso al sistema. Ha privilegi aggiuntivi rispetto all'utente non autenticato, come la gestione del proprio profilo.

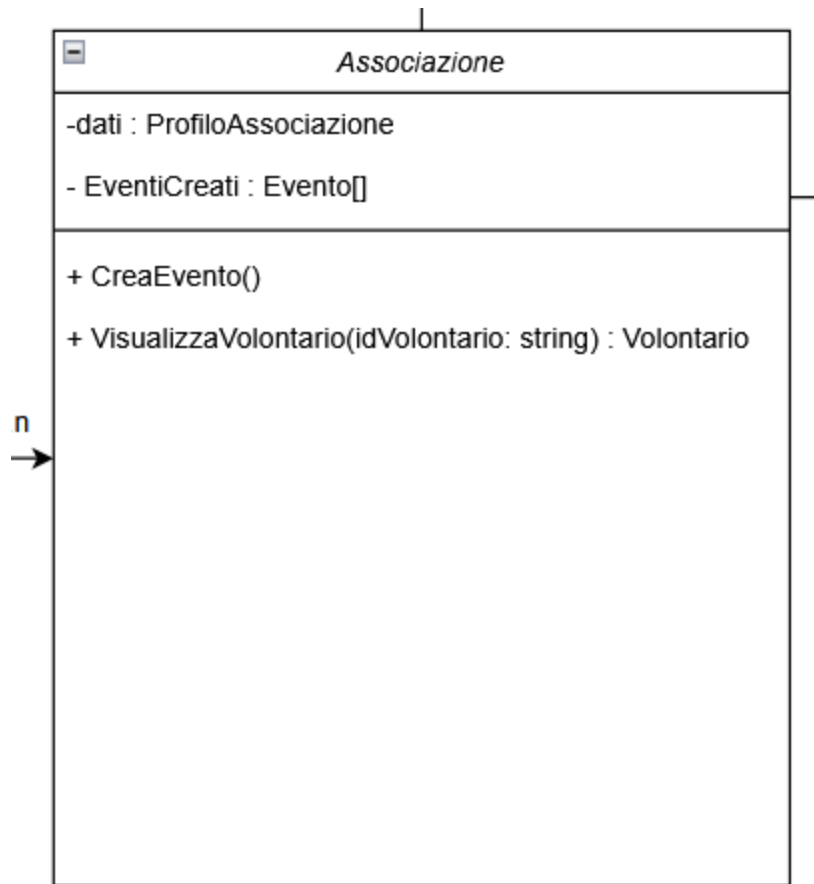
Questa classe funge da base per altre entità come "Volontario" e "Associazione", che derivano dagli utenti autenticati e hanno funzionalità più specifiche.

## Volontario



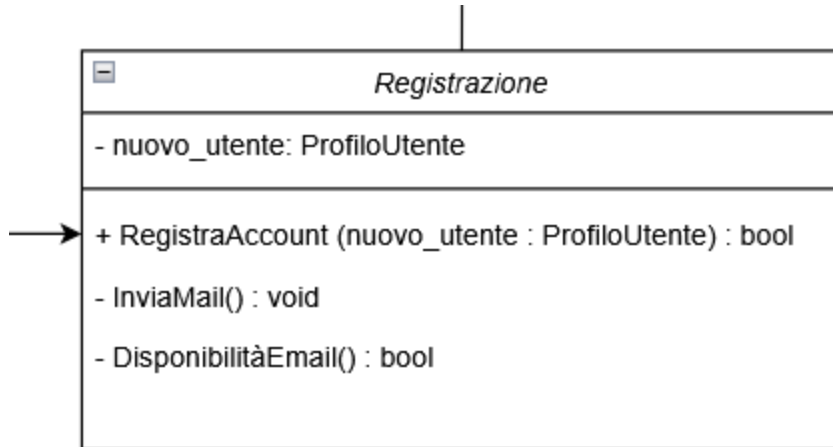
Un volontario è un utente autenticato che può iscriversi agli eventi e seguire associazioni.

## Associazione



Rappresenta un'associazione che può creare e organizzare eventi.

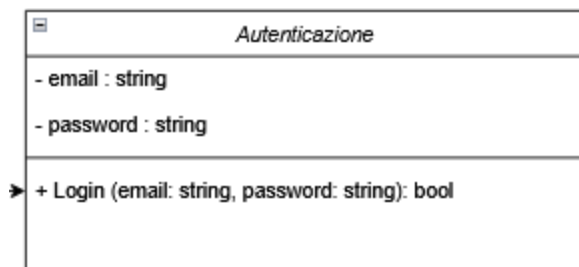
## Registrazione



La classe Registrazione si occupa della creazione di nuovi account utente all'interno del sistema. Gestisce la registrazione di un nuovo utente e verifica la disponibilità dell'email.

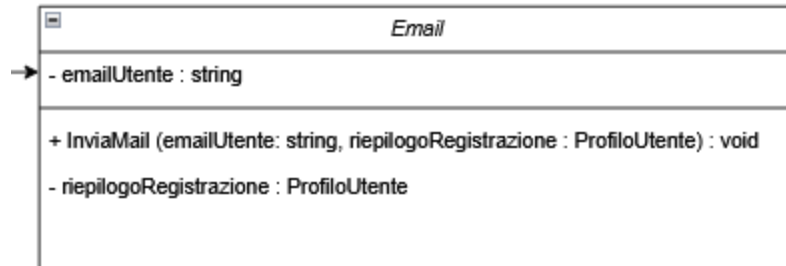
La classe Registrazione è collegata a Email, che gestisce l'invio di messaggi di avvenuta registrazione. Interagisce con "Utente non autenticato", permettendogli di creare un nuovo profilo.

## Autenticazione



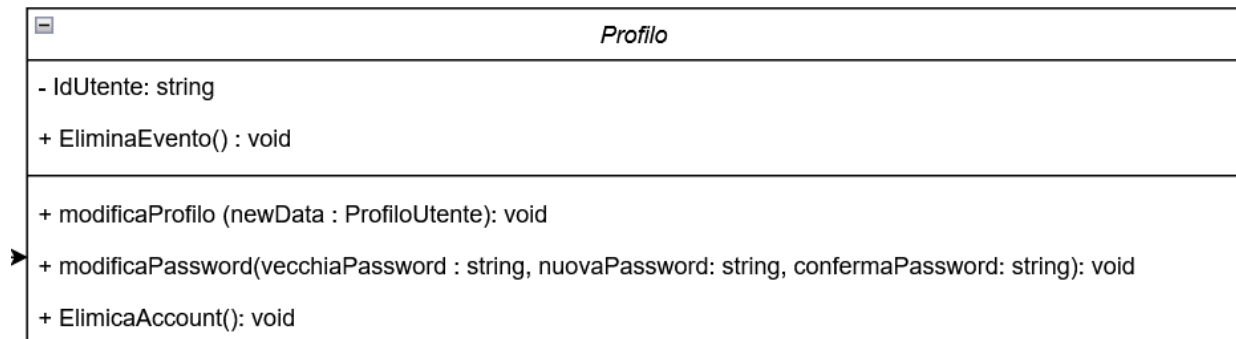
La classe Autenticazione gestisce il processo di login e di autenticazione. Permette agli utenti di accedere al loro account.

## Email



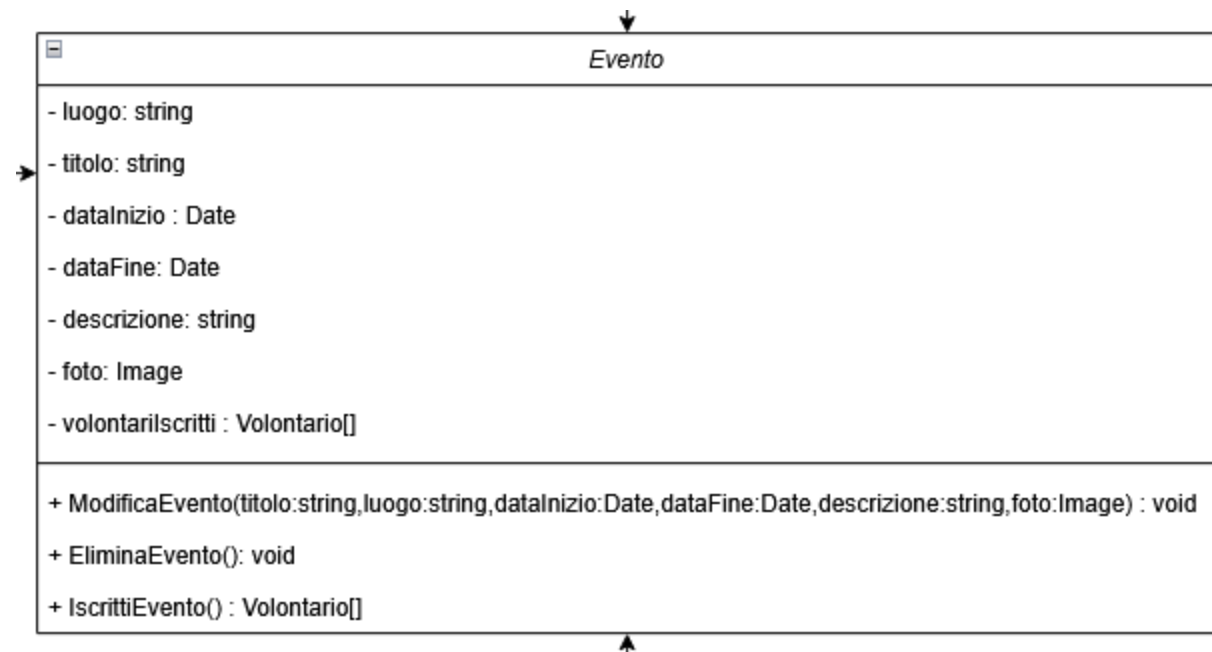
La classe Email gestisce l'invio di email di registrazione.

## Profilo



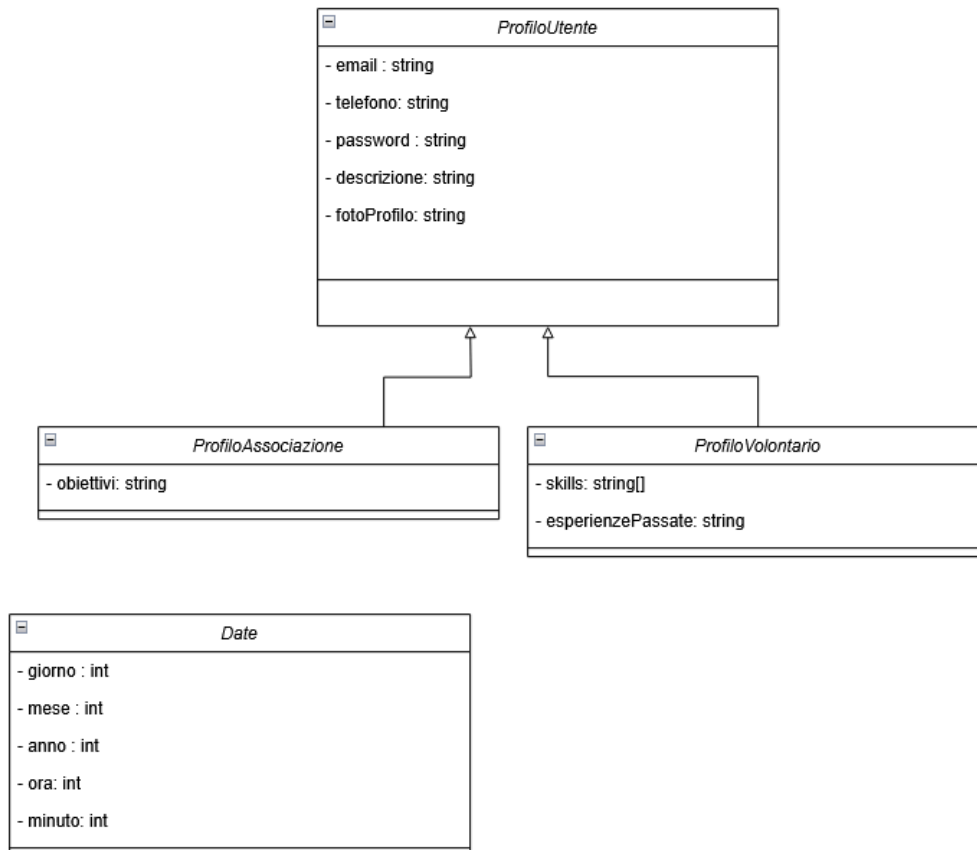
Rappresenta le informazioni e le operazioni che un utente autenticato può eseguire sul proprio account. Consente la gestione dei dati personali, cambio della password e la cancellazione dell'account.

## Evento



La classe *Evento* rappresenta un evento creato da un'associazione a cui i volontari possono iscriversi.

## Classi di supporto



## Dal class diagram alle API

Classe	Metodo	HTTP method	URI + query params	Request	Response	Note
Utente non autenticato	NuovoUtente (nuovo_utente)	POST	/users/registrazioneVolontario	Volontario{ name, surname, age, email, phone, password, description, experience, skills, profilePicture }	201	Registrazione account volontario
	NuovoUtente (nuovo_utente)	POST	/associazioni/registrazioneAssociazione	Associazione{ name, email, phone, password, description, objectives, profilePicture }	201	Registrazione account associazione
	Login ( email, password)	POST	/users/login	{email,password}	201 {token}	Login volontario
	Login ( email, password)	POST	/associazioni/login	{email,password}	201 {token}	Login associazioni
	CercaEvento(nomeEvento, filtro)					Implementato lato frontend



	CercaAssociazione(nomeAssociazione)					Implementato lato frontend
	VisualizzaEvento(idEvento)	GET	/eventi/getEvento/idEvento		201 Evento{ name, startDate, endDate, location, description }	
	VisualizzaAssociazione(idAssociazione)	GET	/eventi/getAssociazione/idAssociazione		201 Associazione	
	Eventi()	GET	/eventi/getEventi		201 Evento[]	
	Associazioni()	GET	/associazioni/getAssociazioni		201 Associazione[]	
Utente Autenticato	logout()	POST	/logout		201	
Profilo	modificaProfilo (newData)	POST	/user/ModifyProfile	Volontario	201	modifica Profilo volontario
	modificaProfilo (newData)	POST	/associazioni/ModifyProfile	Associazione	201	modifica Profilo associazione
	modificaPassword(vecchiaPassword , nuovaPassword, confermaPassword)	POST	/user/changePassword	{vecchiaPassword ,nuovaPassword}	201	-cambio password

						volontari o -Il check su nuovaPa ssword e conferm aPasswo rd è fatto sul frontend
	modificaPassword(vecchiaPassw ord , nuovaPassword, confermaPassword)	POST	/associazioni/ changePassw ord	{vecchiaPassword ,nuovaPassword}	201	-cambio passwor d associazi oni -Il check su nuovaPa ssword e conferm aPasswo rd è fatto sul frontend
	EliminaAccount()	DELETE	/deleteAccoun t		201	non è impleme ntata
Volontario	IscrizioneEvento(idEvento)	POST	/eventi/subscri beEvent	{idEvento}	201	
	DisiscrizioneEvento(idEvento)	POST	/eventi/unsubs cribeEvent	{idEvento}	201	
	FollowAssociazione(idAssociazio ne)	POST	/user/seguirAs sociazione	{idAssociazione}	201	
	UnfollowAssociazione(idAssociaz ione)	POST	/user/unsubsc ribeAssociazio	{idAssociazione}	201	

			ne			
	AssociazioniSeguite()	GET	/user/getAssociazionilscritte		201 Associazione[]	
	EventiAssociazioniSeguite()	GET	/eventi/eventiAssociazionilscritte		201 Evento[]	
	Eventilscritti()	GET	/eventi/getMyEventiVolontario		201 Evento[]	
Associazione	CreaEvento()	POST	/eventi/creaEvento	Evento	201	
	VisualizzaVolontario(idVolontario)	GET	/user/getVolontario/idVolontario		201 Volontario	
Evento	ModificaEvento(titolo,luogo,datainizio,dataFine,descrizione,foto)	POST	/eventi/modifyEvent	Evento	201	
	EliminaEvento()	DELETE	/eventi/deleteEvent	{idEvento}	201	
	IscrittiEvento()	GET	/associazioni/getVolontarilscrittiEvento	{idEvento}	201 Volontario[]	