



# TN2night

09/11/2022

D2-T02

# Indice

|  |           |
|--|-----------|
| <b>Indice</b>                          | <b>2</b>  |
| <b>Scopo del Documento</b>             | <b>3</b>  |
| <b>1. Requisiti Funzionali</b>         | <b>4</b>  |
| <b>2. Requisiti Non Funzionali</b>     | <b>16</b> |
| • RNF 1. Portabilità                   | 16        |
| • RNF 2. Compatibilità                 | 16        |
| • RNF 3. Operatività                   | 16        |
| • RNF 4. Robustezza                    | 17        |
| • RNF 5. Usabilità                     | 17        |
| • RNF 6. Accessibilità                 | 18        |
| • RNF 7. Sicurezza                     | 19        |
| • RNF 8. Privacy                       | 20        |
| • RNF 9. Scalabilità                   | 20        |
| <b>3. Altri diagrammi</b>              | <b>21</b> |
| <b>4. Analisi del Contesto</b>         | <b>22</b> |
| 4.1 Utenti e Sistemi Esterni           | 22        |
| User                                   | 22        |
| UniTn                                  | 23        |
| MongoDB                                | 23        |
| Google Maps                            | 23        |
| Google Traduttore                      | 23        |
| Google Calendar                        | 23        |
| Google Sign-in                         | 23        |
| Google Gmail                           | 23        |
| 4.2 Diagramma di Contesto              | 23        |
| <b>5. Analisi dei componenti</b>       | <b>24</b> |
| 5.1 Diagramma                          | 25        |
| <b>5.2. Definizione dei componenti</b> | <b>26</b> |
| 5.2.1 Gestione autenticazione          | 26        |
| 5.2.2 Gestione Login                   | 26        |
| 5.2.3 Gestione visualizzazione eventi  | 26        |
| 5.2.4 Gestione eventi                  | 27        |
| 5.2.5 Gestione amici                   | 27        |
| 5.2.6 Gestione visualizzazione utenti  | 28        |
| 5.2.7 Gestione ricerca utenti          | 28        |

|                                       |    |
|---------------------------------------|----|
| 5.2.7 Gestione filtro amici           | 28 |
| 5.2.8 Gestione visualizzazione spots  | 28 |
| 5.2.8 Gestione spot                   | 29 |
| 5.2.9 Gestione creazione spot         | 29 |
| 5.2.10 Creazione spot specifico       | 29 |
| 5.2.11 Creazione spot generale        | 30 |
| 5.2.12 Gestione dati da ateneo        | 30 |
| 5.2.13 Visualizzazione pagina utente  | 30 |
| 5.2.14 Gestione registrazione         | 31 |
| 5.2.15 Gestione logout                | 31 |
| 5.2.16 Gestione recupero password     | 31 |
| 5.2.17 Gestione email                 | 31 |
| 5.2.18 Gestione notifiche             | 32 |
| 5.2.19 Gestione matching              | 32 |
| 5.2.20 Gestione aggiornamento profilo | 33 |

## Scopo del Documento

Documento di specifica dei requisiti funzionali e non funzionali con diagrammi dei casi d'uso, tabelle, diagrammi di contesto e dei componenti.

# 1. Requisiti Funzionali

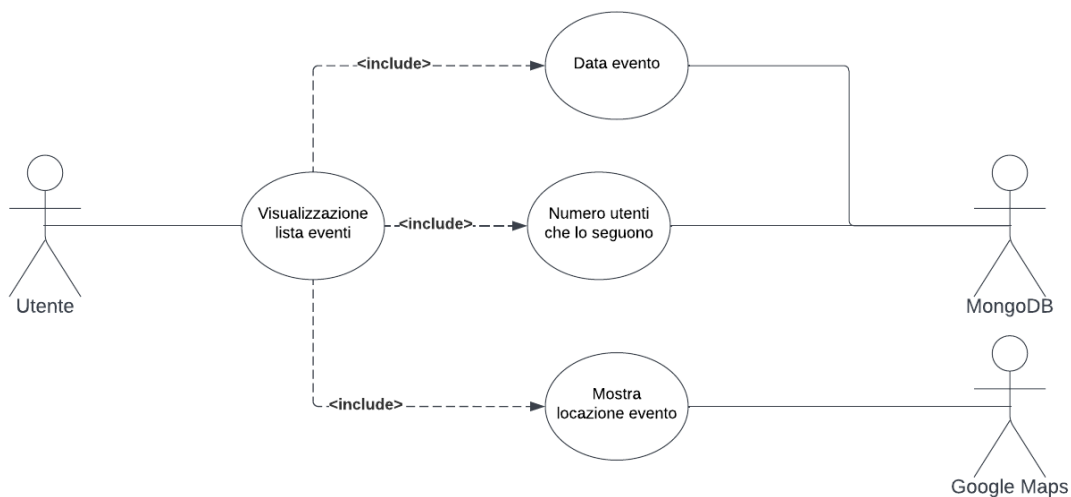
Nel presente capitolo vengono riportati i requisiti funzionali (RF) del sistema utilizzando il linguaggio naturale e Use Case Diagram (UCD) scritti in UML.

RF. 1 Lista eventi

RF. 4 Visualizzazione numero utenti che seguono l'evento

RF. 8 Mostra locazione degli eventi

RF. 9 Lista spot



## Descrizione Use Case "visualizzazione lista eventi"

Titolo: Visualizza lista eventi

Riassunto: Questo use case descrive come visualizzare la lista degli eventi che si svolgeranno nei prossimi giorni

Descrizione:

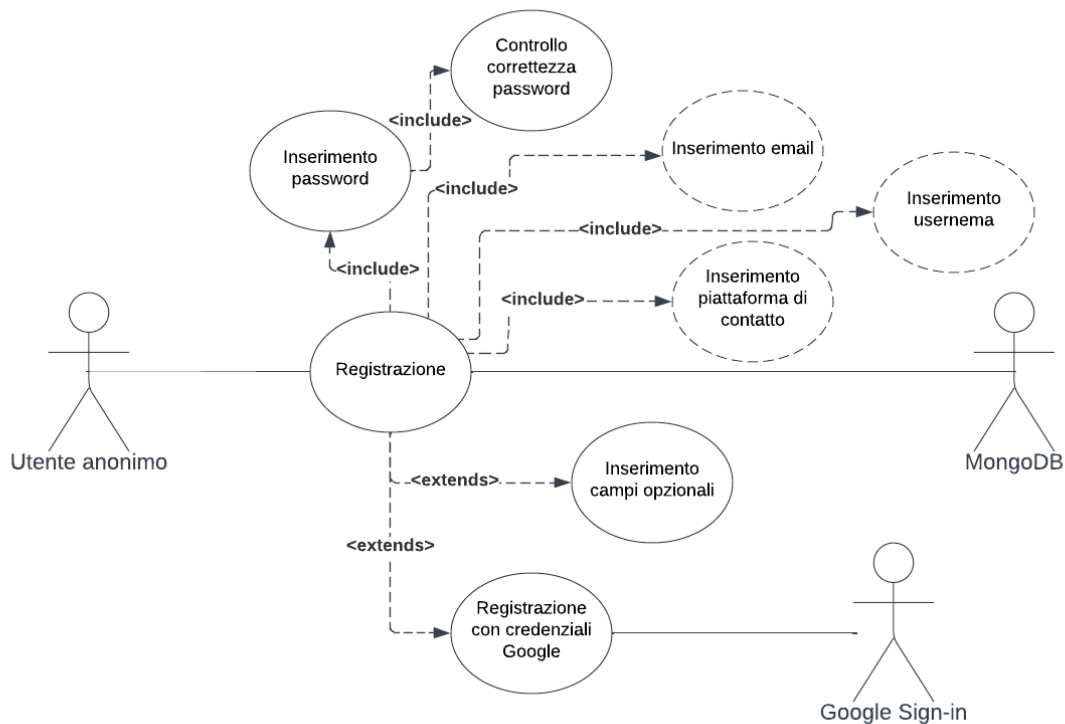
1. L'utente apre la pagina dedicata agli eventi.
2. L'applicazione richiede la lista degli eventi al database.
3. L'applicazione riceve dal database anche la data (gg/mm/aaaa hh:mm), l'indirizzo (via e città) e il numero di utenti che seguono questo evento.
4. L'applicazione mostra la lista.

## Utente anonimo

RF. 2 Registrazione

RF. 3 Suggerimento password

RF. 5 Registrazione con account google



## **Descrizione Use Case "Registrazione"**

Titolo: Registrazione

Riassunto: Permettere ad un utente anonimo di registrarsi

Descrizione:

1. L'utente apre la pagina di registrazione. Può scegliere se proseguire la registrazione con Google Sign-in oppure utilizzare il form fornito dall'applicazione.
2. L'utente, nel caso decidesse di non utilizzare Google sign-in, inserisce tutti i dati obbligatori (email, password, nome utente e piattaforma di contatto preferita). Se desidera compila anche i campi opzionali (foto profilo, bio, facoltà, anno accademico, regione di provenienza, descrizione fisica) [exception 1].
3. I dati dell'utente vengono inviati al database [exception 2] [exception 3].

## Exceptions

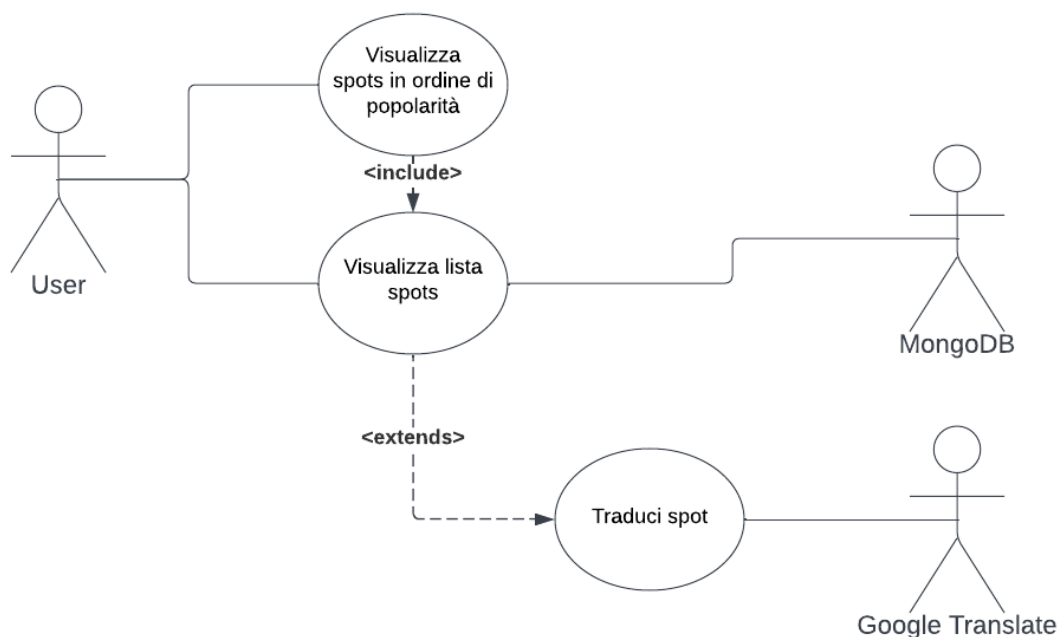
exception 1: l'utente viene notificato se la password non rispetta i criteri di sicurezza o se inserisce due password diverse nei campi dedicati.

exception 2: l'utente viene notificato che l'email è già presente nel database.

exception 3: il sistema mostra un alert se il database non è raggiungibile e i dati dell'utente non possono essere salvati.

RF. 6 Lista spot

RF. 7 Disponibilità spot in italiano e inglese



## Descrizione Use Case "visualizzazione lista spots"

Titolo: Visualizza lista degli spot

Riassunto: Viene visualizzata la lista degli spot in ordine cronologico.

Descrizione:

1. L'utente apre la pagina degli spot.
2. L'applicazione richiede la lista degli spot al database.
3. L'applicazione visualizza la lista in ordine cronologico.

## Descrizione Use Case "traduci spot"

Titolo: Traduzione di uno spot

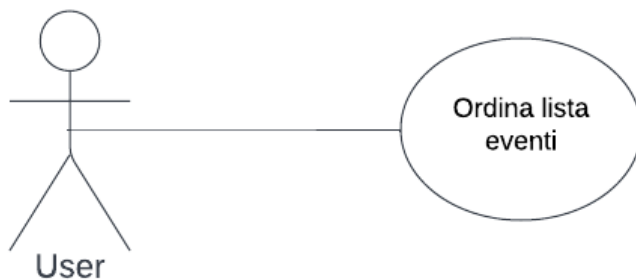
Riassunto: Uno spot selezionato viene tradotto in inglese o, viceversa in italiano.

Descrizione:

1. L'utente seleziona il tasto di traduzione presente sullo spot.
2. L'applicazione invia il testo a google translate.
3. L'applicazione sostituisce il testo con quello tradotto.

## **UTENTE REGISTRATO**

RF. 26 Visualizzazione eventi con più iscritti



### **Descrizione Use Case "ordina lista eventi"**

Titolo: Ordina lista degli eventi

Riassunto: Permettere la scelta dell'ordinamento di visualizzazione degli eventi.

Descrizione:

1. L'utente preme il tasto "Eventi".
2. L'applicazione cambia il criterio di ordinamento degli eventi: ordine cronologico (default) o ordine di popolarità.

RF. 13 Registrazione agli eventi

RF. 17 Lista amici che partecipano agli eventi

RF. 24 Aggiunta evento a calendario personale



### Descrizione Use Case "Registrazione agli eventi"

Titolo: Registrazione ad un evento

Riassunto: Permettere agli utenti registrati di registrarsi ad un evento

Descrizione:

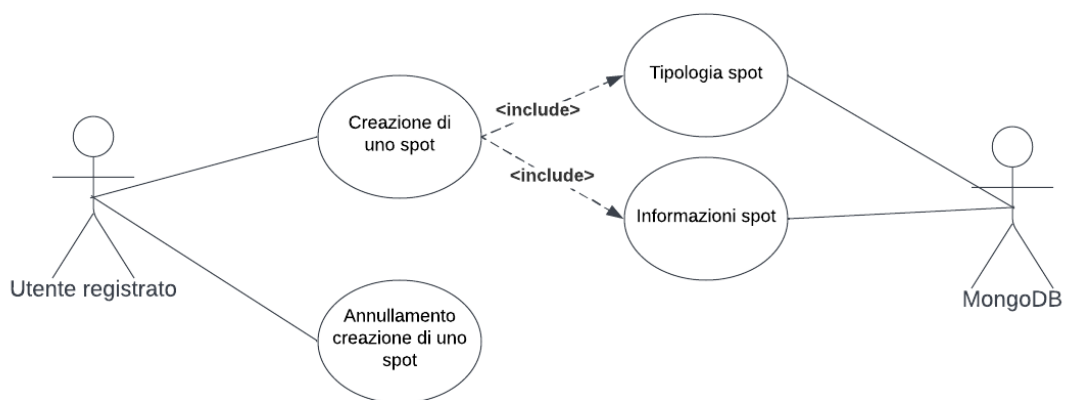
1. L'utente preme l'icona di registrazione ad un evento.
2. L'applicazione informa il database che l'utente si è registrato ad un evento.
3. L'utente risulta registrato all'evento scelto [extends 1].

Extends

extends: l'evento viene segnato sul calendario dell'utente nel caso si fosse registrato con un account Google.

RF. 19 Creazione di uno spot

RF. 20 Annullamento della creazione di uno spot





## Descrizione Use Case "Creazione di uno spot"

Titolo: creazione di uno spot.

Riassunto: Dare la possibilità ad un utente di creare uno spot.

Descrizione:

1. L'utente dalla pagina degli spot preme il tasto di creazione di uno nuovo spot.
2. L'utente seleziona il tipo di spot (default: spot generale) [extends 1].
3. L'utente scrive il suo spot [exception 1].
4. L'applicazione invia i dati al database.

Exception

exception 1: l'utente in ogni momento può annullare la creazione dello spot.

Extends

extends 1: l'utente sceglie un tipo di spot diverso da quello generale.

RF. 27 Votazione positiva spot



## Descrizione Use Case "Votazione positiva di uno spot"

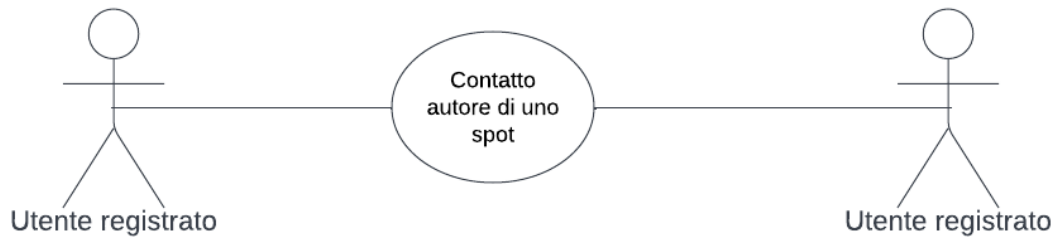
Titolo: Votazione positiva di uno spot.

Riassunto: Dare la possibilità all'utente di mettere mi piace ad uno spot.

Descrizione:

1. L'utente preme l'icona per votare uno spot.
2. L'applicazione aggiorna il contatore di "mi piace" di uno spot nel database.

RF. 21 Contatto autore di uno spot



### **Descrizione Use Case "Votazione positiva di uno spot"**

Titolo: Votazione positiva di uno spot.

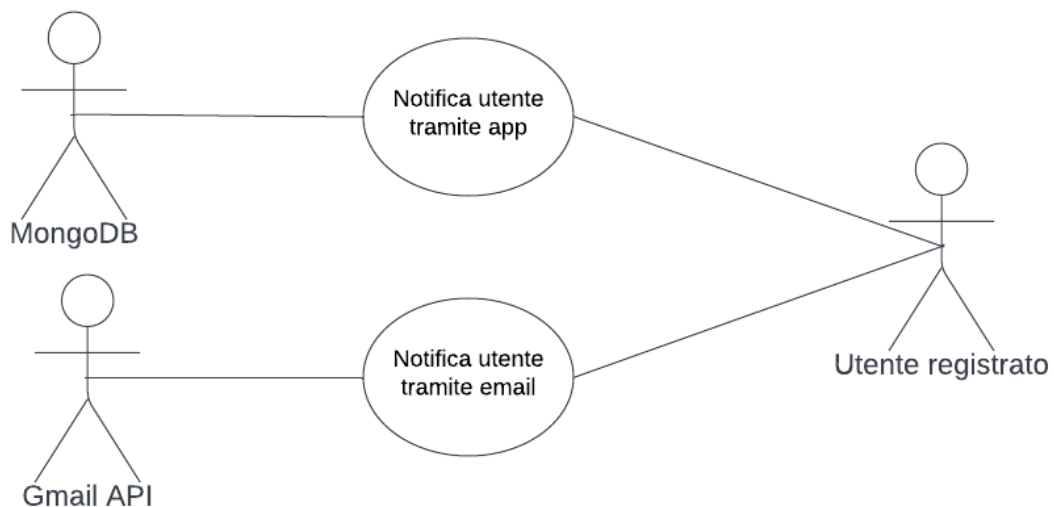
Riassunto: Dare la possibilità all'utente di mettere mi piace ad uno spot.

Descrizione:

1. Un utente seleziona lo spot a cui è interessato.
2. L'applicazione lo reindirizza al profilo dell'autore dello spot.
3. L'utente prende le informazioni di contatto dell'autore dal suo profilo.

RF. 22 Notificare utenti tramite app

RF. 23 Notificare utenti tramite email



### **Descrizione Use Case "Notifica utente tramite app"**

Titolo: Notifica utente tramite app.

Riassunto: Un utente corrispondente alla descrizione di uno spot viene notificato dall'applicazione.

Descrizione:

1. Dopo la creazione di uno spot l'applicazione trova i profili che corrispondono maggiormente alla descrizione nello spot.

2. L'applicazione fa apparire un pop-up di notifica agli utenti corrispondenti [exception 1].

Exception

exception 1: se l'utente non è online l'applicazione salva la notifica nella casella delle notifiche dell'utente nel database. Quando l'utente si collega l'applicazione invia le notifiche tramite pop-up.

## Descrizione Use Case "Notifica utente tramite email"

Titolo: Notifica utente tramite email.

Riassunto: Un utente corrispondente alla descrizione di uno spot viene notificato dall'applicazione tramite email.

Descrizione:

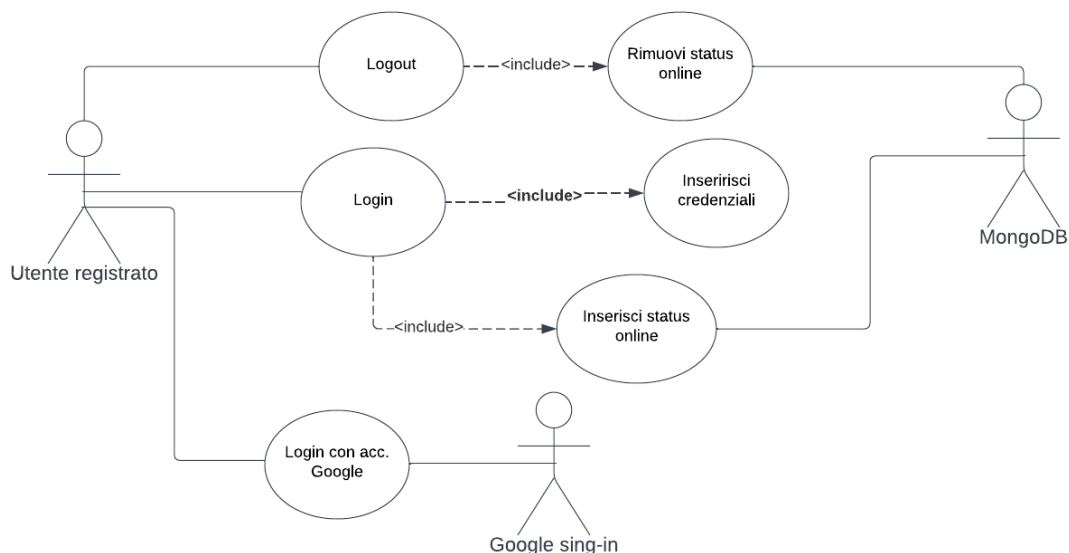
1. Dopo la creazione di uno spot l'applicazione trova i profili che corrispondono maggiormente alla descrizione nello spot.
2. L'applicazione tramite le API di gmail invia una email agli utenti corrispondenti.

RF. 10 Login

RF. 12 Login con account google

RF. 28 Visualizzazione stato attività amici

RF. 30 Logout



## **Descrizione Use Case "Login"**

Titolo: Login

Riassunto: Login dell'utente e aggiornamento del suo status online

Descrizione:

1. L'utente accede alla pagina di login.
2. L'utente inserisce le proprie credenziali (email e password).
3. L'applicazione verifica con il database la correttezza delle credenziali [exception 1].
4. Se le credenziali sono corrette l'applicazione aggiorna lo status online dell'utente nel database.
5. All'utente viene garantito l'accesso alle funzioni dedicate agli utenti registrati.

Exception

exception 1: se le credenziali sono errate l'applicazione mostra un messaggio di errore all'utente.

## **Descrizione Use Case "Logout"**

Titolo: Logout

Riassunto: Logout dell'utente e aggiornamento del suo status offline

Descrizione:

1. L'utente accede alla pagina del profilo.
2. L'utente preme il tasto del logout.
3. L'applicazione aggiorna sul database lo status dell'utente ad offline.
4. L'utente dispone solo delle funzionalità dedicate ad utenti anonimi sull'applicazione.

## **Descrizione Use Case "Login con account Google"**

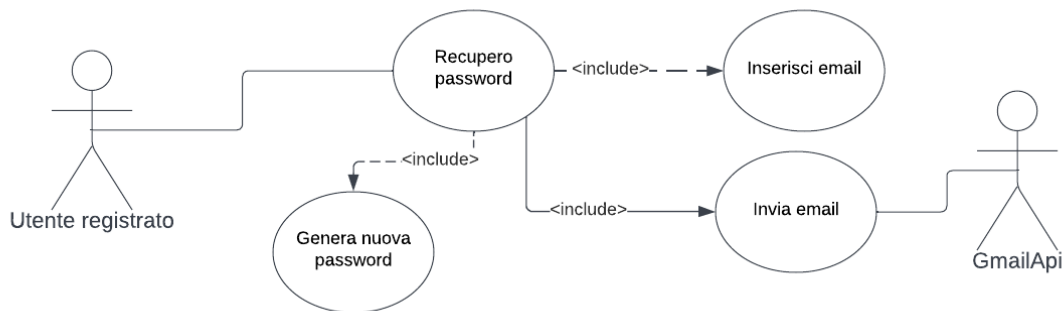
Titolo: Login con account Google.

Riassunto: L'utente svolge il login con la funzionalità Google sign-in e le sue credenziali Google.

Descrizione:

1. L'utente accede alla pagina di login.
2. L'utente preme il tasto 'Usa Google'.
3. Una volta confermato l'accesso da Google Sign-in l'applicazione aggiorna lo status online dell'utente.
4. All'utente viene garantito l'accesso alle funzioni dedicate agli utenti registrati.

## RF. 11 Recupero password



### Descrizione Use Case "Recupero password"

Titolo: Recupero password.

Riassunto: Il sistema manda una email con una password generata automaticamente all'utente.

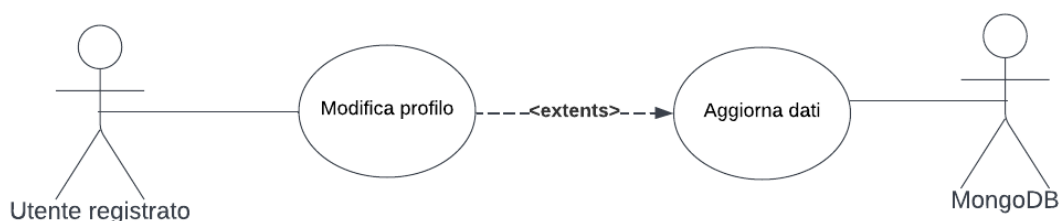
Descrizione:

1. L'utente preme il tasto 'Forgot password?'
2. L'utente inserisce la propria mail.
3. L'applicazione verifica che la mail sia presente sul database [exception 1].
4. Il sistema genera una password casuale automaticamente.
5. Il sistema invia una email contenente questa password con l'utilizzo delle API di Gmail.

### Exception

exception 1: se la mail non è presente nel database il sistema mostra un messaggio di errore.

## RF. 25 Modifica profilo



## Descrizione Use Case "Modifica profilo"

Titolo: Modifica profilo.

Riassunto: L'utente modifica i dati del proprio profilo.

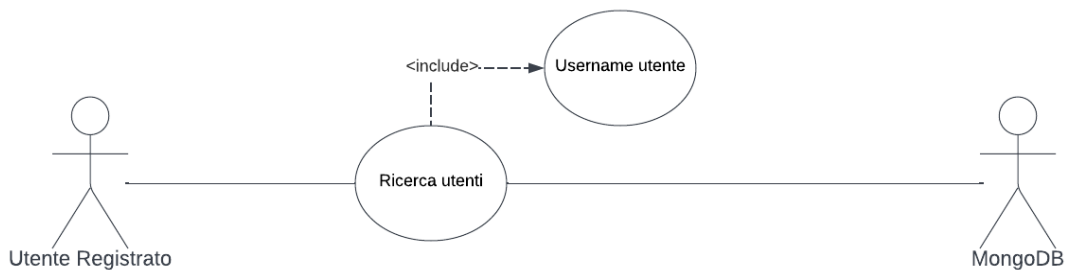
Descrizione:

1. L'utente dalla pagina del profilo modifica i dati desiderati.
2. L'applicazione invia gli aggiornamenti al database [extension 1].

Extension

extension 1: se l'utente non modifica nessun campo allora l'applicazione non invia nessun comando al database.

## RF. 14 Ricerca Utenti



## Descrizione Use Case "Ricerca utenti"

Titolo: Ricercare utenti.

Riassunto: L'utente ricerca altri utenti registrati nell'applicazione.

Descrizione:

1. L'utente apre la pagina di ricerca utenti.
2. L'utente inserisce l'username dell'utente della persona che desidera cercare.
3. L'applicazione invia la ricerca al database. [exception 1]
4. L'applicazione mostra la lista degli utenti con un username che include l'username cercato.

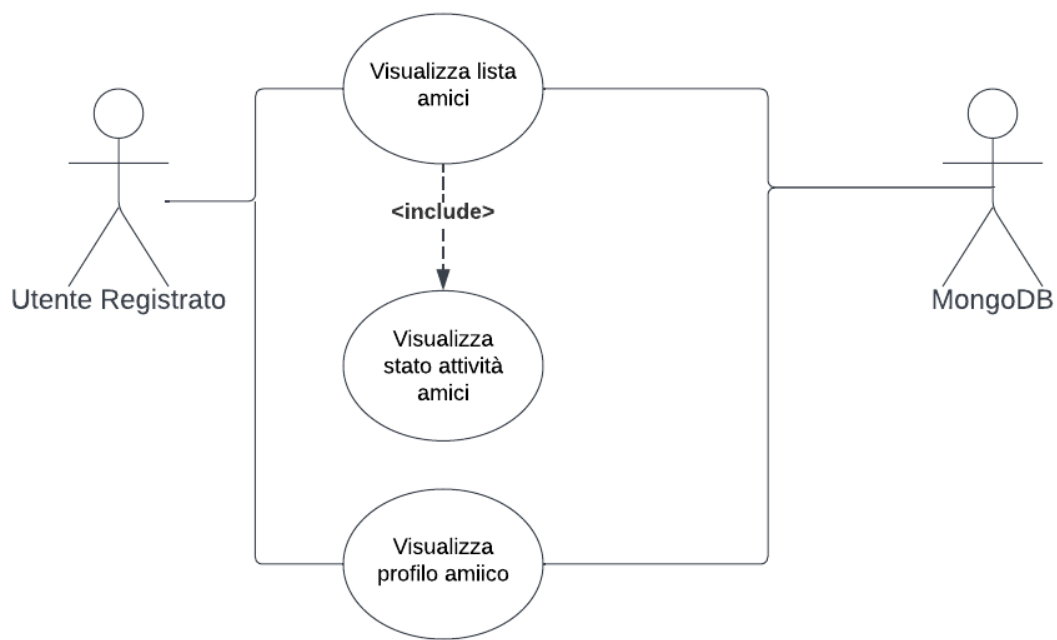
Exception

exception 1: Il sistema mostra un messaggio d'errore se non ci sono corrispondenze con l'username cercato.

RF. 18 Visualizza lista amici

RF. 28 Visualizza stato attività amici

RF. 31 Visualizza profilo amico



### **Descrizione Use Case “Visualizza lista amici”**

Titolo: Visualizza amici.

Riassunto: L’utente visualizza la sua lista amici.

Descrizione:

1. L’utente preme l’icona per visualizzare la lista degli amici.
2. L’applicazione ricerca gli amici nel database.
3. L’applicazione visualizza la lista degli amici.
4. L’applicazione mostra al fianco di ogni amico lo status (online o offline).

### **Descrizione Use Case “Visualizza profilo amico”**

Titolo: Visualizza profilo amico.

Riassunto: L’utente visualizza il profilo di un amico.

Descrizione:

1. L’utente preme sul profilo di un amico.
2. L’applicazione richiede le informazioni sul profilo dell’amico ricercato al database.
3. L’applicazione mostra il profilo dell'amico.

## 2. Requisiti Non Funzionali

Nel presente capitolo vengono riportati i requisiti non funzionali (RNF) del sistema utilizzando tabelle strutturate e specificando misure facilmente verificabili.

- **RNF 1. Portabilità**

| Proprietà    | Descrizione   | Misura   |
|--------------|---|--|
| Indipendenza | Il sistema deve avere le stesse funzionalità e avere la stessa interfaccia su tutti i dispositivi (mobile, tablet, computer). | L'applicazione deve avere le stesse funzionalità su computer e dispositivi mobile con età inferiore ai 5 anni. |

- **RNF 2. Compatibilità**

| Proprietà     | Descrizione  | Misura  |
|---------------|--|---|
| Compatibilità | Il software deve funzionare sui browser più utilizzati e con una certa versione. | Il software deve funzionare sui seguenti browser: Chrome, Safari, Opera, Firefox, Edge. Tali browser devono essere aggiornati alle loro ultime versioni a partire dal 2020. |

- **RNF 3. Operatività**

| Proprietà   | Descrizione   | Misura  |
|-------------|---|---|
| Operatività | Il sistema deve avere un tempo di uptime elevato e limitare il tempo di down al minimo. | Il sistema deve avere un uptime pari al 99.1% ogni mese: 23 h 47 min 2 s al giorno. |



• **RNF 4. Robustezza**

| Proprietà                       | Descrizione   | Misura   |
|---------------------------------|---|--|
| Tempo di soluzione del database | Il database deve essere robusto e in caso di malfunzionamento deve tornare operativo in un tempo relativamente breve.                           | Il database deve essere in grado di ritornare operativo entro 1 giorno dopo un qualsiasi imprevisto grazie ad un sistema di back-up. |
| Unicità dell'email              | Un utente non deve potersi iscrivere alla stessa piattaforma più di una volta con la stessa password.   | Il sistema non deve permettere la registrazione di più account utilizzando lo stesso indirizzo email.                                |
| Unicità dello username          | Tutti gli utenti devono avere dei nomi univoci che li identificano: non deve essere quindi possibile che due utenti abbiano lo stesso username. | Il sistema non deve permettere la registrazione di più account con lo stesso username.   |

• **RNF 5. Usabilità**

| Proprietà         | Descrizione  | Misura  |
|-------------------|--|---|
| Velocità di avvio | L'applicazione deve avviarsi in maniera veloce.                                | Il sistema deve mostrare la schermata di autenticazione entro 3 secondi dalla richiesta dell'utente.  |
| UI/UX design      | Il software deve soddisfare i moderni principi di <a href="#">UI/UX Design</a> | Conforme ai seguenti punti principali: <ul style="list-style-type: none"><li>- white space</li><li>- color balance</li><li>- text hierarchy</li></ul> |

|                          |   |  |
|--------------------------|---|--|
| Semplicità di utilizzo   | Il sistema deve essere facile ed intuitivo da utilizzare.   | Un nuovo utente deve essere in grado di utilizzare tutte le funzionalità entro 9 minuti senza dover ricorrere al supporto di un manuale di utilizzo. |
| Divisione delle funzioni | Il sistema deve essere chiaro nella divisione ed allocazione delle funzioni all'interno del prodotto. | Ogni funzione specificata negli obiettivi del progetto deve avere una pagina dedicata.   |

• **RNF 6. Accessibilità**

| Proprietà                           | Descrizione  | Misura   |
|-------------------------------------|--|--|
| Lingue di sistema                   | L'intero sistema deve essere disponibile in due lingue.                        | Ogni pagina del software deve rispecchiare le preferenze di sistema (host) dell'utente: ogni pagina deve essere disponibile in italiano e inglese. Quest'ultima impostazione verrà scelta per tutti gli utenti che non hanno l'italiano come lingua preferita. |
| Standard per gli utenti non vedenti | Il sistema deve permettere l'utilizzo del software anche a utenti non vedenti. | Le diverse pagine del sistema devono essere conformi con gli standard di accessibilità per i non vedenti specificate nel seguente link: <a href="#">HTML Accessibility</a> .   |

• **RNF 7. Sicurezza**

| Proprietà           | Descrizione  | Misura  |
|---------------------|--|---|
| Password            | La password deve essere <i>forte</i> contro eventuali attacchi.  | La password inserita dall'utente in fase di registrazione deve essere lunga: <ul style="list-style-type: none"><li>- almeno 8 caratteri</li><li>- meno di 126 caratteri</li></ul> e deve contenere almeno: <ul style="list-style-type: none"><li>- 1 carattere maiuscolo (A-Z)</li><li>- 1 carattere minuscolo (a-z)</li><li>- una cifra (0-9)</li><li>- un carattere speciale (!£\$%&amp;?+ #)</li></ul> |
| Verifica dell'email | Il sistema deve verificare che l'email inserita in fase di registrazione sia effettivamente appartenente all'utente  | Conferma tramite link inviato con Gmail   |
| GDPR                | Il sistema deve garantire la sicurezza del database da eventuali attacchi tramite l'utilizzo di strumenti specifici in modo da rispettare il principio di <a href="#">integrità e confidenzialità espessi nel GDPR</a> | Conforme: <ul style="list-style-type: none"><li>• Misure di sicurezza tecniche ed organizzative</li><li>• Sicurezza adeguata al rischio</li></ul>   |
| Connessione sicura  | Il sistema deve fornire una connessione crittografata tra l'utente ed il server  | Il sistema deve usare una connessione HTTPS   |

|                |  |   |
|----------------|--|---|
| Scadenza token | Dopo un tempo determinato il sistema deve effettuare il logout dell'utente | Dopo 172800s (48h), a un utente registrato verrà chiesto di autenticarsi nuovamente |
|----------------|--|---|

• **RNF 8. Privacy**

| Proprietà | Descrizione   | Misura   |
|-----------|---|--|
| GDPR      | Rispetto dei principi di trasparenza, scopo limitato, raccolta dei dati strettamente necessari, accuratezza, trattenimento dei dati per il tempo strettamente necessario e di responsabilità previsti dal <a href="#">GDPR</a> , tramite un'esplicita conferma. | Conforme: <ul style="list-style-type: none"> <li>• Trattamento lecito, equo e trasparente</li> <li>• Il responsabile della protezione dei dati</li> <li>• Il consenso</li> </ul> |

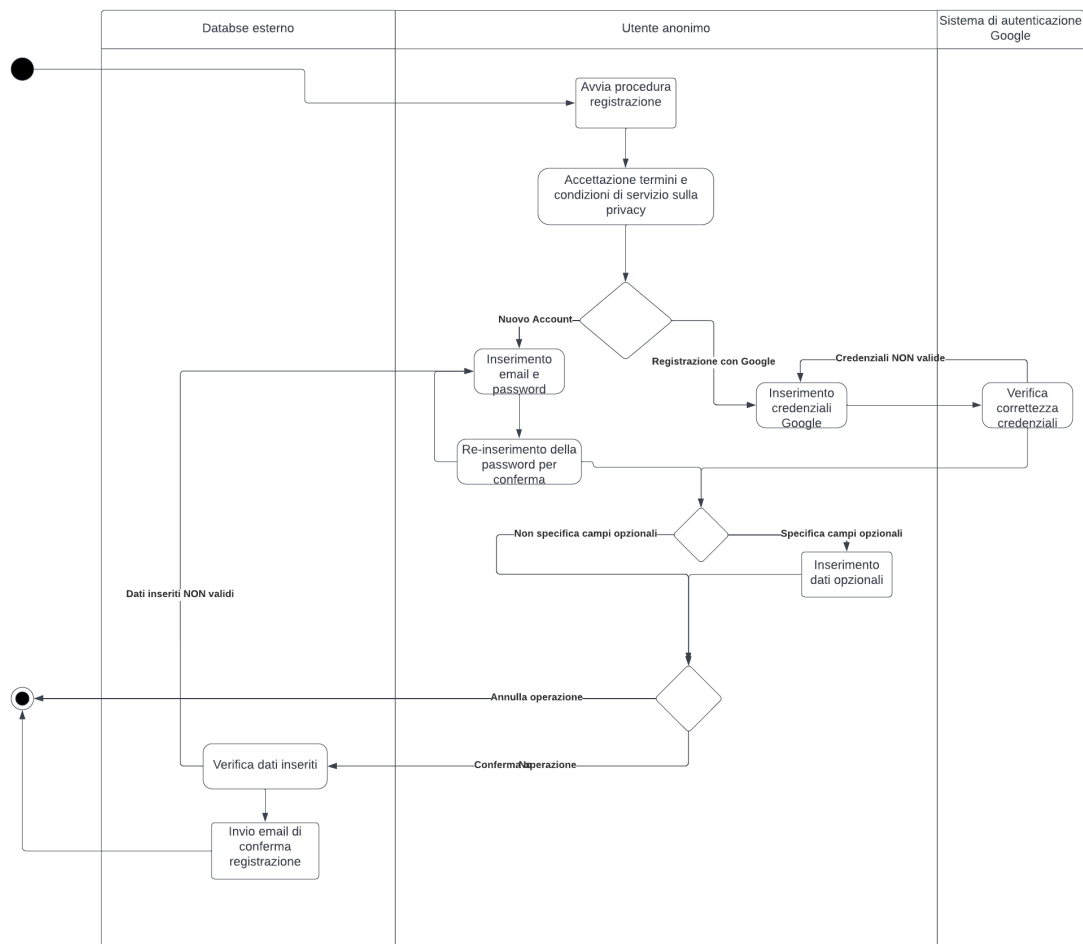
• **RNF 9. Scalabilità**

| Proprietà   | Descrizione                       | Misura   |
|-------------|-----------------------------------|--|
| Scalabilità | Il sistema deve essere scalabile. | L'intero sistema deve funzionare correttamente con 50'000 utenti che la utilizzano simultaneamente |

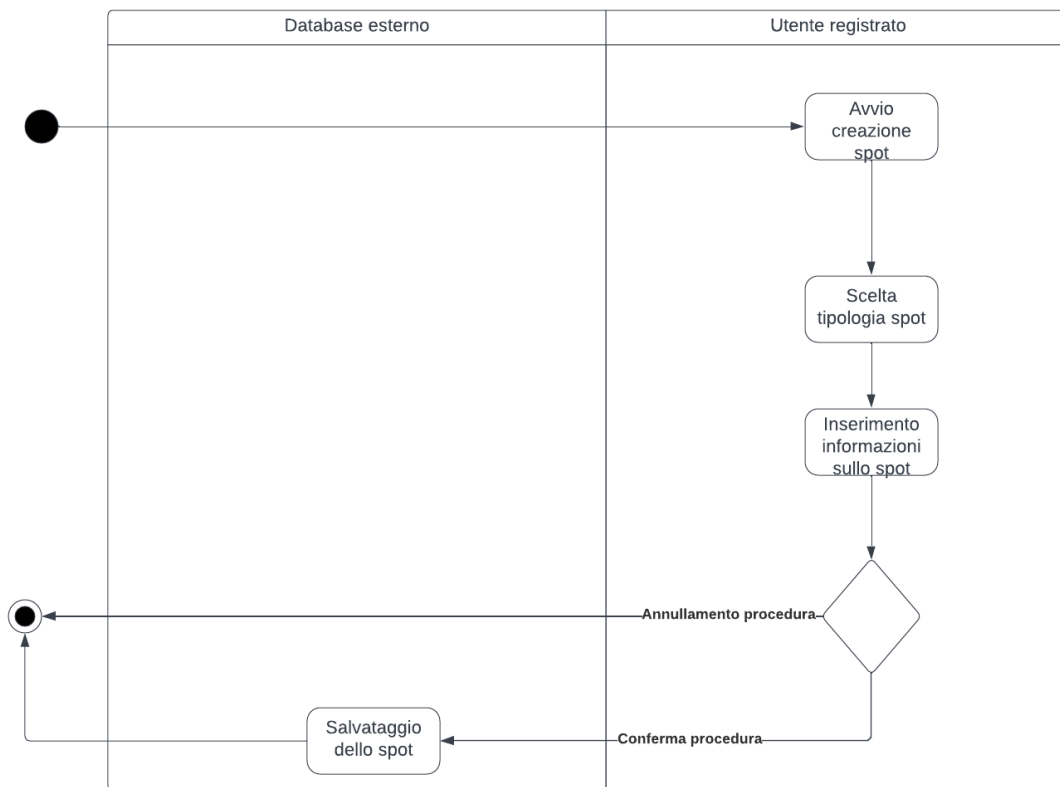
### 3. Altri diagrammi

Al fine di dettagliare meglio alcune procedure vengono riportati i seguenti diagrammi:

#### Diagramma appendice 1: Registrazione Utente



## Diagramma Appendice 2: Creazione di uno spot



## 4. Analisi del Contesto

Nel presente capitolo viene discusso il contesto di funzionamento del sistema, fornendo una descrizione testuale e una rappresentazione grafica basata su Context Diagram.

Nella seguente parte della sezione vengono presentati gli attori e i sistemi esterni con cui l'applicazione TN2night si interfacerà.

### 4.1 Utenti e Sistemi Esterni

#### User

Colui che utilizza l'applicazione per visualizzare i contenuti della stessa. Una volta effettuata l'autenticazione (registrazione **RF.2**, accesso **RF.10**), l'utente potrà usufruire delle funzionalità descritte al **RF.13**, **RF.14** e **RF.19**.

## UniTn

Utilizzato per fornire le facoltà e i corsi che un utente frequenta nel periodo in cui effettua la registrazione al sistema per la prima volta, o per eventuali future modifiche al profilo (**RF.2**, **RF.25**).

## MongoDB

Utilizzato per salvare le informazioni degli utenti, degli eventi e degli spot. L'intera applicazione si basa sulla memorizzazione di dati in questo database esterno. Tale sistema è utilizzato anche per poter reperire le informazioni salvate nel database nel momento del bisogno.

## Google Maps

Sistema utilizzato per visualizzare il luogo dell'evento selezionato, come specificato nel **RF.8**.

## Google Traduttore

Sistema utilizzato per la traduzione degli spot, come specificato in **RF.7**.

## Google Calendar

Utilizzato quando l'utente vuole salvare la data e l'ora dell'evento nel proprio calendario personale GCalendar (**RF.24**).

## Google Sign-in

Servizio utilizzato per effettuare il login tramite account google, come specificato in **RF.5** e **RF.12**.

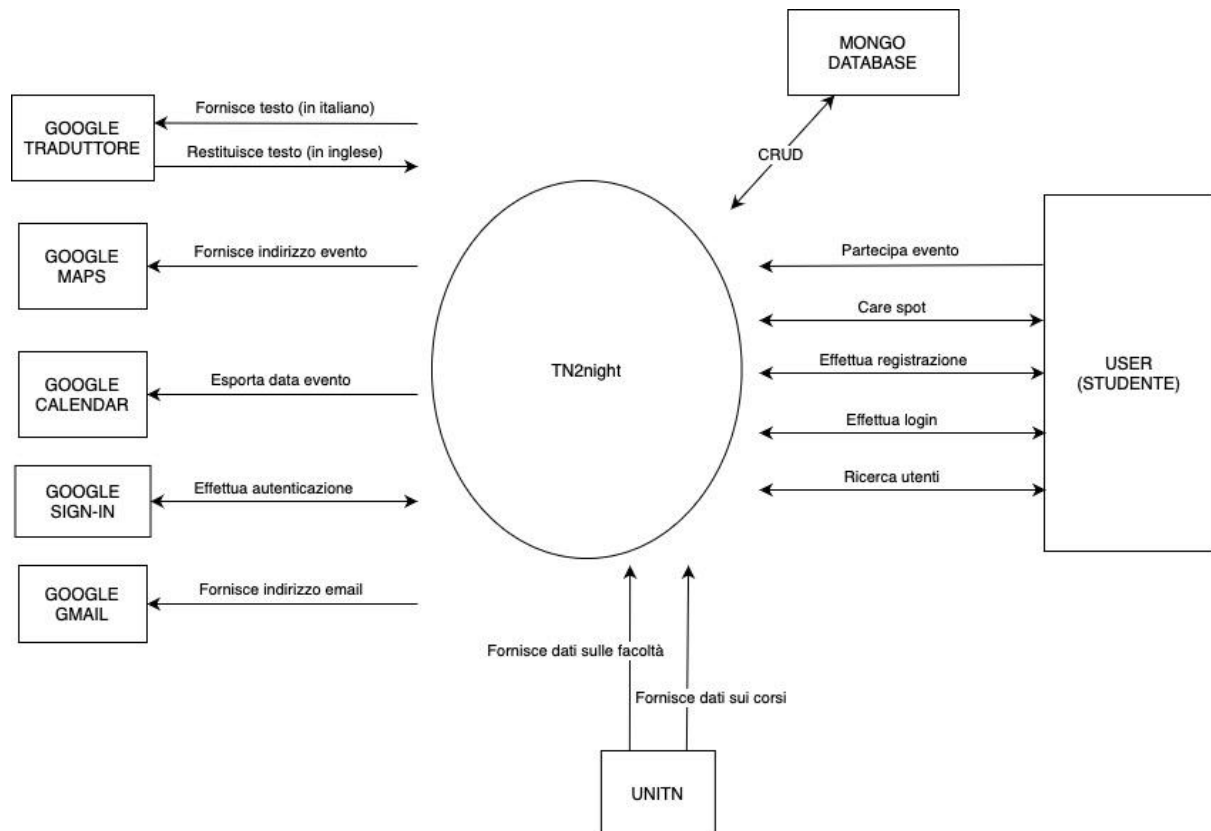
## Google Gmail

Usato per notificare gli utenti in caso si verifichi il matching con uno spot (**RF.23**). Sistema utilizzato anche per il recupero della password: se l'utente si dovesse dimenticare della password per accedere all'applicazione richiederà di ri-impostarla tramite un link inviatogli con Gmail (**RF.11**).

## 4.2 Diagramma di Contesto

L'utente si registra nell'applicazione come specificato in **RF.2** successivamente effettua il login (**RF.10**), da questo momento in poi

l'utente visualizza gli eventi (**RF.1**) che può decidere se tradurli (**RF.7**), vederne la loro locazione geografica su google maps (**RF.8**). Può inoltre decidere di salvare l'evento (**RF.13**) in questo caso l'applicazione aggiunge direttamente l'evento al google calendar dell'email utilizzata dall'utente (**RF.24**). Inoltre l'utente può scegliere se creare uno spot (**RF.19**).



**Figura 1. Diagramma di contesto per l'applicazione TN2night**

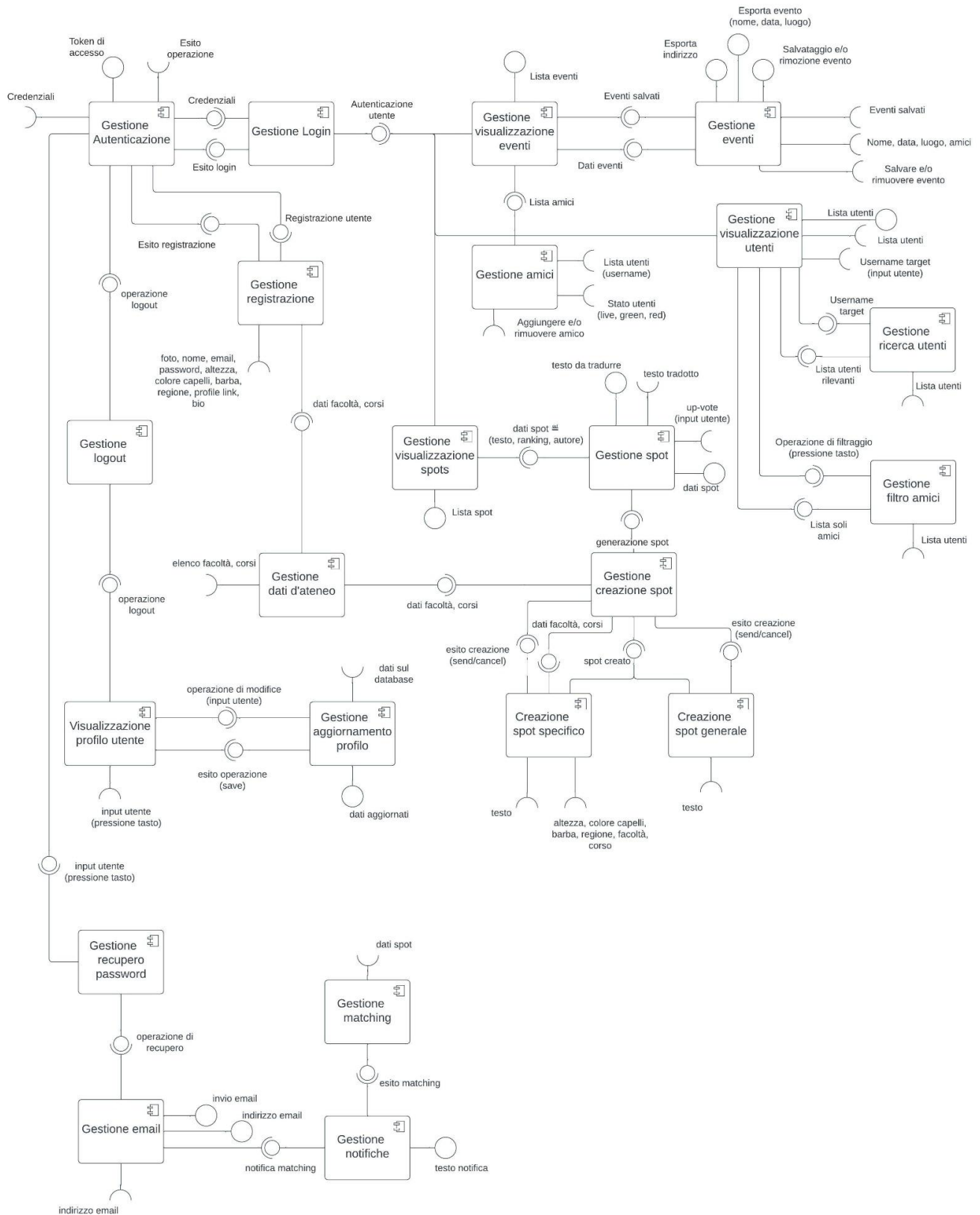
La Figura 1 mostra le principali interazioni tra attori o sistemi esterni e l'applicazione TN2night.

## 5. Analisi dei componenti

Nel presente capitolo viene presentata l'architettura in termini di componenti interni al sistema definiti sulla base dei requisiti analizzati nei precedenti documenti. Viene poi adottato l'uso di Component Diagram per rappresentare l'interconnessione tra i vari componenti, identificando quindi le interfacce tra questi e verso sistemi esterni. Viene infine valutato il livello di *accoppiamento* tra i componenti.



## 5.1 Diagramma



## 5.2. Definizione dei componenti

In questa sezione vengono definiti i componenti.

### 5.2.1 Gestione autenticazione

Motivazione: I Requisiti **RF.2**, **RF.10** **RF.12** per essere soddisfatti, devono interfacciarsi prima con la pagina di login. È stato quindi identificato un componente **Gestione autenticazione** che si interfaccerà con il sistema di google sign-in e con i sistemi gestione login e registrazione permettendo così l'autenticazione dell'utente.

Interfaccia richiesta - credenziali: le credenziali includono email e password e vengono fornite in input dall'utente.

Interfaccia fornita - credenziali: [Rif.](#)

Interfaccia richiesta - esito login: l'esito del login può essere un successo o meno in base agli errori rilevati

Interfaccia fornita - registrazione utente: indica ad altre componenti che è necessario passare alla modalità registrazione.

Interfaccia richiesta - esito registrazione: indica se la registrazione dell'utente è andata a buon fine o meno, in base agli errori rilevati

Interfaccia richiesta - esito operazione: indica se il login con account google è andato a buon fine.

Interfaccia fornita - token di accesso: è necessario per identificare l'utente al servizio di google sign-in.

### 5.2.2 Gestione Login

Motivazione: Dato il requisito **RF.10** e **RF.12** è necessario gestire l'autenticazione, questo è effettuato tramite il componente Gestione Login

Interfaccia fornita - autenticazione utente: fornisce i dati dell'utente nel caso in cui si sia autenticato

Interfaccia richiesta - credenziali: [Rif.](#)

Interfaccia fornita - esito login: [Rif.](#)

### 5.2.3 Gestione visualizzazione eventi

Motivazione: Dato il requisito **RF.1**, **RF.17** e **RF.26**, è stato individuato il componente *Gestione visualizzazione eventi* che permette di gestirli.

Interfaccia richiesta - autenticazione utente: [Rif.](#)

Interfaccia fornita - lista eventi: fornisce la lista degli eventi all'utente, con le relative informazioni

Interfaccia richiesta - Eventi Salvati: fornisce la lista che l'utente ha deciso di salvare sul suo calendario

Interfaccia richiesta - Dati eventi: fornisce gli tutti i dati identificativi dell'evento

Interfaccia richiesta - Lista amici: riguarda la lista amici dell'utente per verificare quanti di questi partecipano agli eventi

#### 5.2.4 Gestione eventi

Descrizione: questo componente, che soddisfa gli **RF 1, 4, 8, 9, 13, 24, 26**, gestisce tutti gli eventi presenti sulla piattaforma e le funzionalità associate a questi. Questo componente si interfaccia con il componente homepage in quanto quest'ultimo permette la visualizzazione di tutti gli eventi e offre tutti i tasti necessari per interagire.

Interfaccia richiesta - Eventi salvati: [Rif.](#)

Interfaccia fornita - Dati eventi: [Rif.](#)

Interfaccia richiesta - Nome,data,luogo,amici: gli amici sono tutti gli amici dell'utente che partecipano a quello specifico evento, nome,data,luogo sono altri parametri per identificare l'evento.

Interfaccia fornita - Salvataggio/ rimozione evento: fornisce al DB gli eventi che l'utente ha deciso di salvare o rimuovere.

Interfaccia richiesta - Salvare/rimuovere evento: [Rif.](#)

Interfaccia richiesta - Esporta indirizzo: interfaccia necessaria a visualizzare il luogo dell'evento su altre piattaforme.

Interfaccia richiesta - Esporta evento: interfaccia necessaria a salvare il proprio evento sul calendario.

#### 5.2.5 Gestione amici

Motivazione: il componente soddisfa gli **RF 14, 15, 16** implementando le funzionalità per gestire, modificare e visualizzare la lista amici.

Interfaccia fornita - Lista amici: [Rif.](#)

Interfaccia richiesta - Lista utenti: il db fornisce l'interfaccia contenente tutti gli utenti registrati

Interfaccia richiesta - Stato utenti: il db fornisce anche lo stato degli utenti (online/offline) se questi ultimi risultano essere amici dell'utente

Interfaccia richiesta - Aggiungi/Rimuovi amico: Interfaccia necessaria a rimuovere un amico dalla lista di amici dell'utente

### 5.2.6 Gestione visualizzazione utenti

Descrizione: questo componente offre un'interfaccia la quale permette di ricercare utenti registrati nel sistema. L'utente inserisce lo username da ricercare e il componente restituisce una lista contenente gli username più affini a quello cercato. Soddisfa l'**RF 14**.

Interfaccia fornita - Lista utenti: [Rif.](#)

Interfaccia richiesta - Username target: contiene l'username da cercare, fornito dall'utente

Interfaccia fornita - Lista utenti rilevanti: contiene la lista degli utenti a cui è stato applicato il filtro

Interfaccia fornita - Operazione filtraggio: è fornita dall'utente nel momento in cui preme l'icona che permette di filtrare la lista

Interfaccia richiesta - Lista utenti rilevati: [Rif.](#)

Interfaccia richiesta - Lista soli amici: fornisce la lista di utenti che sono amici dell'utente.

### 5.2.7 Gestione ricerca utenti

Motivazione: questo componente si interfaccia con il database contenente gli utenti ed effettua una operazione di ricerca partendo dall'username fornito. Questo componente serve per soddisfare l'**RF 14**.

Interfaccia fornita - Lista utenti rilevati: contiene la lista degli utenti a cui è stato applicato il filtro

Interfaccia richiesta - Lista utenti: [Rif.](#)

Interfaccia richiesta - Username target: [Rif.](#)

### 5.2.7 Gestione filtro amici

Motivazione: questo componente soddisfa l'**RF 18** in quanto permette di visualizzare la lista degli amici dell'utente tramite l'utilizzo di un pulsante.

Interfaccia richiesta - Operazione filtraggio: [Rif.](#)

Interfaccia richiesta - Lista utenti: [Rif.](#)

Interfaccia fornita - lista soli amici: [Rif.](#)

### 5.2.8 Gestione visualizzazione spots

Motivazione: il componente permette di visualizzare la lista degli spot registrati nel sistema. Soddisfando il **RF 6**. Questo componente inoltre permette di ordinare gli spot come definito nel **RF 29**.

Interfaccia fornita - lista spot: fornisce la lista degli spot presenti del database

Interfaccia richiesta - Testo, ranking, autore: fornisce i dati dello spot che verrà mostrato

### 5.2.8 Gestione spot

Descrizione: questo componente per soddisfare gli **RF 7, 19, 20, 21, 27** permette all'utente di effettuare diverse azioni su un singolo spot e la creazione stessa di uno spot. Tra queste ci sono la traduzione, l'up-vote e la visualizzazione dei dati dello spot. Per creare lo spot l'utente viene indirizzato ad una pagina e un componente separato.

Interfaccia fornita - Testo, ranking, autore: [Rif.](#)

Interfaccia fornita - Testo da tradurre: fornisce il testo che l'utente ha scelto di tradurre

Interfaccia richiesta - Testo tradotto: ottiene dall'api di google translate il testo tradotto

Interfaccia richiesta - up-vote: è un up-vote fornito dall'utente

Interfaccia fornita - dati spot: mostra tutti i dati dello spot all'utente

Interfaccia richiesta - generazione spot: contiene lo spot appena creato

### 5.2.9 Gestione creazione spot

Descrizione: questo componente permette ad un utente di creare uno spot. È necessario specificare diversi dati tra cui la descrizione e il messaggio. Inoltre l'utente deve selezionare se desidera fare uno spot specifico ad un evento o uno generale. In base all'ultima scelta il componente chiede informazioni ad un altro componente.

Interfaccia richiesta - dati facolt: corsi: vengono fornite dal portale dell'ateneo

Interfaccia richiesta - esito creazione (specifico): eseguita con successo oppure annullata

Interfaccia fornita - dati facoltà, corsi: [Rif.](#)

Interfaccia richiesta - spot creato: è fornita sia dalla creazione di uno spot generale che specifico

Interfaccia richiesta - esito creazione (generale): [Rif.](#)

Interfaccia fornita - generazione spot: contiene lo spot appena creato

### 5.2.10 Creazione spot specifico

Descrizione: questo componente assiste nella creazione di uno spot permettendo all'utente di specificare dei dettagli sul luogo associato allo spot.

Interfaccia fornita - sport creato: [Rif.](#)

Interfaccia fornita - esito creazione: [Rif.](#)

Interfaccia richiesta - dati facoltà, corsi o eventi: necessari nella creazione di questo tipo di spot

Interfaccia richiesta - testo: dato fornito dall'utente

Interfaccia richiesta - altezza, colore capelli, barba, regione, facoltà, corso: dati sulla persona forniti dall'utente

### 5.2.11 Creazione spot generale

Descrizione: questo componente permette di creare uno spot che non contiene alcune informazioni aggiuntive, tipo il luogo o l'evento a cui è associato.

Interfaccia fornita - esito creazione: [Rif.](#)

Interfaccia fornita - sport creato: [Rif.](#)

Interfaccia richiesta - testo: [Rif.](#)

### 5.2.12 Gestione dati da ateneo

Descrizione: questo componente serve per interagire con i dati di ateneo, principalmente la lista dei corsi. Il componente si interfaccia con le API dell'università e condivide i dati ricevuti con altri componenti del sistema.

Interfaccia fornita - dati facoltà, corsi (spot):

Interfaccia richiesta - elenco facoltà, corsi: dati richiesti all'api di UniTN

Interfaccia fornita - dati facoltà, corsi (registrazione): [Rif.](#)

### 5.2.13 Visualizzazione pagina utente

Descrizione: questo componente permette la visualizzazione del profilo dell'utente. Questa pagina contiene i dati dell'utente, modificabili passando il controllo al componente di modifica, e il tasto per effettuare il logout.

Interfaccia fornita - operazione logout: trasmette ad altre interfaccia l'intenzione dell'utente di effettuare il logout

Interfaccia richiesta - input utente: consiste in un pulsante che permette all'utente di effettuare il logout

Interfaccia fornita - operazione di modifica: il componente ottiene le modifiche fatte dall'utente al proprio profilo.

Interfaccia richiesta - esito operazione: una volta terminata l'operazione di aggiornamento il componente notifica l'utente (che durante tutta l'operazione si trovava sulla pagina profilo) del successo dell'operazione.

#### 5.2.14 Gestione registrazione

Descrizione: questo componente permette di gestire la registrazione dell'utente, include una pagina dedicata, andando così a soddisfare il **RF.2**.

Interfaccia richiesta - registrazione utente: [Rif.](#)

Interfaccia richiesta - informazioni utente: Contiene le informazioni che l'utente ha deciso di inserire (foto, nome, email, password, altezza, colore capelli, barba, regione, profile link, bio)

interfaccia richiesta - dati facoltà, corsi: [Rif.](#)

interfaccia fornita - esito registrazione: indica se la registrazione è andata buon fine o meno

#### 5.2.15 Gestione logout

Descrizione: questo componente permette di passare le informazioni dalla profile page alla gestione autenticazione in modo tale che l'utente risulti offline sulla piattaforma soddisfacendo il **RF.30**

Interfaccia fornita - operazione logout: [Rif.](#)

Interfaccia richiesta - operazione logout: [Rif.](#)

#### 5.2.16 Gestione recupero password

Motivazione: dato l'**RF 11** il sistema richiede un componente che permetta all'utente di recuperare la propria password. L'utente accede a questo componente premendo il tasto per resettare la password nella pagina di Login. Il componente genera una nuova password casuale e poi la invia all'utente.

Interfaccia richiesta - email: il componente richiede all'utente la sua mail per potergli inviare la nuova password

Interfaccia fornita - operazione di recupero: il componente si affida al componente di gestione email per inviare un'email all'utente contenente la nuova password.

#### 5.2.17 Gestione email

Motivazione: il componente si occupa di inviare email agli utenti come richiesto dall'**RF 23**. Viene richiesto il suo utilizzo quando un utente deve recuperare la propria password o quando bisogna notificare gli utenti che vengono spottati.

Interfaccia richiesta - indirizzo email: il componente richiede l'indirizzo email dell'utente al quale deve inviare una mail.

Interfacci richiesta - notifica matching: quando avviene un match questo componente viene utilizzato per notificare gli utenti individuati tramite email.-

Interfaccia fornita - invio email: il componente crea intestazione e corpo dell'email che poi verranno inviate utilizzando il servizio Gmail.

### 5.2.18 Gestione notifiche

Motivazione: come richiesto dall'**RF 22** questo componente notifica un utente da app quanto viene pubblicato uno spot che lo riguarda.

Interfaccia richiesta - esito matching: questo componente viene attivato quando un 'match' viene individuato tra uno spot e la descrizione di un utente.

Interfaccia fornita - testo notifica: il componente fornisce agli utenti individuati grazie all'interfaccia 'esito matching' una notifica contenente lo spot associato a loro.

Interfaccia fornita - notifica matching: il componente invoca la gestione email per notificare l'utente di avvenuto 'match' come per **RF 23**.

### 5.2.19 Gestione matching

Motivazione: ogni volta che uno spot viene creato il sistema utilizza questo componente per verificare se la descrizione di qualche utente è simile alla descrizione fornita nello spot. In caso affermativo vengono attivati i componenti propedeutici per notificare gli utenti dell'avvenuto 'match'.

Interfaccia richiesta - dati spot: il componente riceve i dati di un nuovo spot creato.

Interfaccia richiesta - descrizione utente: il componente riceve la descrizione di un utente che andrà a confrontare con i dati dello spot appena creato.



Interfaccia fornita - esito matching: in caso il confronto nel punto prima si positivo il componente invia i dati dello spot e il nome dell'utente individuato al componente di gestione notifiche.

#### 5.2.20 Gestione aggiornamento profilo

Motivazione: per **RF 25** un utente può modificare il proprio profilo. Questo componente riceve dati dalla pagina profilo dell'utente e se si sono svolte modifiche aggiorna i dati nel database.

Interfaccia richiesta - dati sul database: il componente riceve in input i dati dal database per confrontarli con quelli inseriti dall'utente ed aggiornare in seguito solo quelli modificati.

Interfaccia richiesta - operazioni di modifica: [Rif.](#)

Interfaccia fornita - dati aggiornati: il componente fornisce i dati profilo aggiornati al database esterno che andrà a modificare i dati dell'utente.

Interfaccia fornita - esito operazione: [Rif.](#)