



UNIVERSITÀ DEGLI STUDI DI NAPOLI  
**FEDERICO II**

C.d.l. triennale di Informatica  
Anno Accademico 2021/2022

# Documentazione di progetto di Ingegneria del software



**NaTour21**

## **Studenti**

Vincenzo De Simone  
N86002924  
Crescenzo Di Vano  
N86003022

## **Docenti**

Prof. Sergio Di Martino  
Prof. Francesco Cutugno  
Dott. Marco Grazioso

# Indice

<b>I Documento dei requisiti software</b>	<b>1</b>
<b>1 Descrizione dell'idea progettuale</b>	<b>2</b>
<b>2 Glossario</b>	<b>3</b>
<b>3 Specifica dei requisiti</b>	<b>4</b>
<b>3.1 Requisiti funzionali</b> . . . . .	<b>4</b>
<b>3.2 Requisiti non funzionali</b> . . . . .	<b>9</b>
<b>3.3 Requisiti di dominio</b> . . . . .	<b>11</b>
<b>3.4 Use Case Diagram</b> . . . . .	<b>12</b>
<b>3.5 Tabelle Di CockBurn</b> . . . . .	<b>13</b>
3.5.1 Inserisce nuovi itinerari . . . . .	13
3.5.2 Invia messaggi privati . . . . .	15
3.5.3 Lettura messaggi privati ricevuti . . . . .	16
<b>3.6 Wireframe</b> . . . . .	<b>17</b>
3.6.1 W-I-Miei-Itinerari . . . . .	17
3.6.2 W-Tracciato . . . . .	18
3.6.3 W-Disegna-Tracciato . . . . .	19
3.6.4 W-Disegna-Tracciato-2 . . . . .	20
3.6.5 W-Tracciato-2 . . . . .	21
3.6.6 W-Punti-Di-Interesse . . . . .	22
3.6.7 W-Informazioni-Itinerario . . . . .	23
3.6.8 W-Informazioni-Itinerario-2 . . . . .	24
3.6.9 W-Cerca-Itinerario . . . . .	25
3.6.10 W-Cerca-Itinerario-2 . . . . .	26
3.6.11 W-Itinerario . . . . .	27
3.6.12 W-Invio-Chat-Utente . . . . .	28
3.6.13 W-Invio-Chat . . . . .	29
3.6.14 W-Home . . . . .	30
3.6.15 W-Riceve-Chat . . . . .	31
3.6.16 W-Ricevete-Chat-Utente . . . . .	32

3.6.17 W-Home-Notifica . . . . .	33
<b>4 Modelli di dominio</b>	<b>34</b>
<b>4.1 Diagrammi di classi di analisi</b>	<b>34</b>
4.1.1 Si autentica . . . . .	34
4.1.2 Si autentica con social . . . . .	35
4.1.3 Si registra . . . . .	35
4.1.4 Effettua disconnessione . . . . .	36
4.1.5 Recupera Password . . . . .	36
4.1.6 Inserisce nuovi itinerari . . . . .	37
4.1.7 Visualizza e ricerca Itinerari . . . . .	37
4.1.8 Scarica informazioni sentiero . . . . .	38
4.1.9 Visualizza e modifica profilo . . . . .	39
4.1.10 Invia messaggi privati . . . . .	40
<b>4.2 Diagrammi di sequenza di analisi</b>	<b>41</b>
4.2.1 Si Autentica . . . . .	41
4.2.2 Si Registra . . . . .	42
4.2.3 Visualizza I Proprio Itinerari . . . . .	43
<b>4.3 Diagrammi di attività</b>	<b>44</b>
4.3.1 Autenticazione e Registrazione . . . . .	44
4.3.2 Modifica Profilo . . . . .	45
4.3.3 Inserimento Itinerario e Punti di Interesse . . . . .	46
4.3.4 Modifica Itinerario . . . . .	47
4.3.5 Rimozione Itinerario . . . . .	48
4.3.6 Inserimento Punti di Interesse . . . . .	49
4.3.7 Rimozone Punti Di Interesse . . . . .	50
4.3.8 Chat . . . . .	51
<b>II Documento di design del sistema</b>	<b>52</b>
<b>5 Analisi dell'architettura</b>	<b>53</b>
<b>5.1 Architettura Client</b>	<b>53</b>
<b>5.2 Architettura Server</b>	<b>54</b>
5.2.1 Servizi cloud . . . . .	54
5.2.2 API . . . . .	55
<b>6 Modelli di Design</b>	<b>56</b>
<b>6.1 Diagrammi di classi di design</b>	<b>56</b>

6.1.1	Si autentica . . . . .	56
6.1.2	Si autentica con social . . . . .	57
6.1.3	Si registra . . . . .	57
6.1.4	Effettua disconnessione . . . . .	58
6.1.5	Recupera Password . . . . .	58
6.1.6	Inserisce nuovi itinerari . . . . .	59
6.1.7	Visualizza e ricerca Itinerari . . . . .	59
6.1.8	Scarica informazioni sentiero . . . . .	60
6.1.9	Visualizza e modifica profilo . . . . .	60
6.1.10	Invia messaggi privati . . . . .	61
<b>6.2</b>	<b>Diagrammi di sequenza di design . . . . .</b>	<b>62</b>
6.2.1	Si Autentica . . . . .	62
6.2.2	Si Registra . . . . .	63
6.2.3	Visualizza I Proprio Itinerari . . . . .	64
<b>III</b>	<b>Testing del sistema . . . . .</b>	<b>65</b>
<b>7</b>	<b>Unit testing . . . . .</b>	<b>66</b>
<b>7.1</b>	<b>Strategia Black Box . . . . .</b>	<b>66</b>
7.1.1	Testing metodo <i>convertGpxIntoClass</i> . . . . .	66
7.1.2	Testing metodo <i>updateAvatar</i> . . . . .	69
<b>7.2</b>	<b>Strategia White Box . . . . .</b>	<b>71</b>
7.2.1	Testing metodo <i>sendChatNotify</i> . . . . .	71
<b>IV</b>	<b>Relazione di usabilità . . . . .</b>	<b>74</b>
<b>8</b>	<b>UX Process . . . . .</b>	<b>75</b>
<b>9</b>	<b>Target degli utenti . . . . .</b>	<b>76</b>
<b>9.1</b>	<b>Studio . . . . .</b>	<b>76</b>
<b>9.2</b>	<b>Questionario . . . . .</b>	<b>77</b>
<b>9.3</b>	<b>Personas . . . . .</b>	<b>84</b>
<b>10</b>	<b>Valutazione e Test Di Usabilità . . . . .</b>	<b>86</b>
<b>11</b>	<b>App Mapping . . . . .</b>	<b>92</b>
<b>12</b>	<b>Gerarchie Funzionali . . . . .</b>	<b>93</b>
<b>13</b>	<b>State Chart . . . . .</b>	<b>94</b>

<b>13.1 Inserimento Tracciato Geografico . . . . .</b>	<b>94</b>
<b>13.2 Invia Messaggio Privato Utente . . . . .</b>	<b>95</b>
<b>13.3 Lettura Messaggio Privato . . . . .</b>	<b>96</b>
<b>14 Mockup . . . . .</b>	<b>97</b>
<b>    14.1 Rappresentazione mockups . . . . .</b>	<b>97</b>
14.1.1 MU-SchermataDiAvvio . . . . .	98
14.1.2 MU-Registrazione-Autenticazione . . . . .	99
14.1.3 MU-RecuperoPassword . . . . .	100
14.1.4 MU-Registrazione . . . . .	101
14.1.5 MU-Registrazione2 . . . . .	102
14.1.6 MU-Registrazione3 . . . . .	103
14.1.7 MU-Registrazione4 . . . . .	104
14.1.8 MU-Registrazione5 . . . . .	105
14.1.9 MU-Registrazione6 . . . . .	106
14.1.10 MU-Drawer . . . . .	107
14.1.11 MU-Account . . . . .	108
14.1.12 MU-ModificaAccount . . . . .	109
14.1.13 MU-Info . . . . .	110
14.1.14 MU-Home . . . . .	111
14.1.15 MU-RicercaItinerario . . . . .	112
14.1.16 MU-RicercaItinerario2 . . . . .	113
14.1.17 MU-Itinerario . . . . .	114
14.1.18 MU-Itinerario2 . . . . .	115
14.1.19 MU-ImieItinerari . . . . .	116
14.1.20 MU-TracciatoGeografico . . . . .	117
14.1.21 MU-TutorialTracciatoGeografico . . . . .	118
14.1.22 MU-DisegnaTracciatoGeografico . . . . .	119
14.1.23 MU-DisegnaTracciatoGeografico2 . . . . .	120
14.1.24 MU-TracciatoGeografico2 . . . . .	121
14.1.25 MU-PuntiDiInteresse . . . . .	122
14.1.26 MU-TutorialPuntiDiInteresse . . . . .	123
14.1.27 MU-SceltaPuntiDiInteresse . . . . .	124
14.1.28 MU-InserisciPuntiDiInteresse . . . . .	125
14.1.29 MU-PuntiDiInteresse2 . . . . .	126
14.1.30 MU-InformazioniItinerario . . . . .	127
14.1.31 MU-InformazioniItinerario2 . . . . .	128

14.1.32 MU-Chat . . . . .	129
14.1.33 MU-Chat2 . . . . .	130
14.1.34 MU-A-Home . . . . .	131
14.1.35 MU-A-Home2 . . . . .	132
14.1.36 MU-A-Home3 . . . . .	133
14.1.37 MU-A-Drawer . . . . .	134
14.1.38 MU-A-UtentiRegistrati . . . . .	135
<b>14.2 Interazione Schermate . . . . .</b>	<b>136</b>
14.2.1 Registrazione . . . . .	136
14.2.2 Inserimento Itinerario e Punti di Interesse . . . . .	137
14.2.3 Invio e Lettura Messaggio . . . . .	138
<b>15 Valutazione usabilità sul campo . . . . .</b>	<b>139</b>
15.1 Inserisci Itinerario . . . . .	139
15.2 Modifica Itinerario . . . . .	140
15.3 Modifica Punti Di Interesse . . . . .	140
15.4 Esporta Dettagli Itinerario . . . . .	141
15.5 Esporta GPX . . . . .	141
15.6 Richiedi Info Itinerario . . . . .	142
15.7 Invia Messaggio . . . . .	142
15.8 Modifica Profilo . . . . .	143
15.9 Cerca . . . . .	143

## Parte I

# Documento dei requisiti software

# 1. Descrizione dell'idea progettuale

La presente documentazione descrive quanto richiesto dal committente SoftEngUniNA, agli studenti Vincenzo De Simone e Crescenzo Di Vano indicati come contraenti, per la realizzazione di un sistema informativo mobile denominato "NaTour21".

Il committente ha richiesto lo sviluppo di un sistema informativo mobile per appassionati di escursionismo, che implementi:

- La registrazione ed autenticazione degli utenti.
- L'inserimento di itinerari catatterizzati da un tracciato geografico facoltativo.
- L'inserimento di punti di interesse su un itinerario da parte di tutti gli utenti.
- L'esportazione dei dettagli dell'itinerario in formato PDF e del percorso geografico in formato GPX.
- La possibilità di inviare messaggi ad utenti con possibilità di risposta.
- La visualizzazione di statistiche lato Admin come numeri di utenti, numero di accessi, itinerari registrati, numero di recensioni, numero di ricerche.

Maggiori dettagli sulle scelte progettuali ed implementative, al fine di realizzare le richieste del committente, dal prossimo paragrafo a seguire.

## 2. Glossario

<b>Use Case Diagram</b>	Sono diagrammi che descrivono come un attore interagisce con le varie funzionalità o servizi messi a disposizione dal sistema.
<b>Class Diagram</b>	I diagrammi delle classi (class diagram) sono uno dei tipi di diagrammi che possono comparire in un modello UML. In termini generali, consentono di descrivere tipi di entità, con le loro caratteristiche e le eventuali relazioni fra questi tipi.
<b>Sequence Diagram</b>	Un Sequence Diagram è un diagramma previsto dall'UML utilizzato per descrivere uno scenario. Uno scenario è una determinata sequenza di azioni in cui tutte le scelte sono state già effettuate; in pratica nel diagramma non compaiono scelte, né flussi alternativi.
<b>Activity Diagram</b>	L'Activity Diagram è un tipo di diagramma che permette di descrivere un processo attraverso dei grafi in cui i nodi rappresentano le attività e gli archi l'ordine con cui vengono eseguite.
<b>State Chart</b>	Uno State Chart è un tipo di diagramma usato per descrivere il comportamento dei sistemi, il quale viene analizzato e rappresentato tramite una serie di eventi che potrebbero accadere per ciascun stato.
<b>Wireframe</b>	I wireframe sono illustrazioni organizzative schematiche dei contenuti presenti nel progetto che stiamo realizzando. La funzione principale dei wireframe è di comunicare l'idea del progetto, focalizzando l'attenzione sull'architettura piuttosto che il design.
<b>Mockup</b>	Un mockup, è una realizzazione a scopo illustrativo di un oggetto o un sistema, senza le complete funzioni dell'originale; un mockup può rappresentare la totalità o solo una parte dell'originale di riferimento.
<b>API</b>	API è l'abbreviazione di (application programming interface), ovvero un insieme di definizioni e protocolli per la creazione e l'integrazione di software applicativi.
<b>Norme ISO</b>	Le <a href="#">norme ISO</a> sono standard di progettazione, che specificano le migliori soluzioni a un determinato problema.
<b>Policy</b>	Linee di condotta che introducono nuovi vincoli.
<b>Utente</b>	Soggetto che ha come facoltà quella di avere accesso al sistema informativo pubblico.
<b>Admin</b>	Soggetto che ha come facoltà quella di avere accesso al sistema informativo privato.
<b>Itinerario</b>	L'Itinerario è un percorso adottato per lo svolgimento di un viaggio, di una gara sportiva su strada, di una manifestazione pubblica, di una visita turistica.
<b>Punto di Interesse</b>	Un punto di interesse è un luogo specifico presente in qualche itinerario che qualcuno può trovare utile o interessante.
<b>Gpx</b>	Un file GPX è un formato che permette la memorizzazione e l'elaborazione di dati GPS (geodati).

# 3. Specifica dei requisiti

## 3.1 Requisiti funzionali

Di seguito elenchiamo i requisiti funzionali con il relativo macro-funzionamento descritto

*Nota:* le funzionalità con somenclatura "/-U" riguardano funzionalità lato utente, mentre "/-A" lato admin.

Nome	RFU1-U   Registrazione alla piattaforma
Descrizione	Tale requisito di sistema si occupa di garantire e consentire all'utente di iscriversi alla piattaforma inserendo: <ul style="list-style-type: none"><li>• Nome</li><li>• Cognome</li><li>• Data di nascita</li><li>• Genere</li><li>• E-mail</li><li>• Password</li><li>• Avatar (facoltativo)</li></ul>

Nome	RFU2-U   Autenticazione alla piattaforma
Descrizione	Tale requisito di sistema si occupa di garantire e consentire all'utente di accedere alla piattaforma indicando: <ul style="list-style-type: none"><li>• Email</li><li>• Password</li></ul>

Nome	RFU3-U   Recupero Account
Descrizione	Tale requisito di sistema si occupa di garantire e consentire all'utente di recuperare la password di accesso, indicando: <ul style="list-style-type: none"><li>• Email</li></ul>

<b>Nome</b>	<b>RFU4-U   Visualizzazione Account</b>
<b>Descrizione</b>	Tale requisito di sistema si occupa di garantire e consentire all'utente di consultare il proprio profilo, visualizzando: <ul style="list-style-type: none"> <li>• Avatar</li> <li>• Nome</li> <li>• Cognome</li> <li>• Data di registrazione alla piattaforma</li> <li>• E-mail</li> <li>• Data di nascita</li> <li>• Genere</li> <li>• Data e orario dell'ultimo accesso</li> <li>• Impostazioni notifiche messaggi</li> </ul>

<b>Nome</b>	<b>RFU5-U   Modifica Account</b>
<b>Descrizione</b>	Tale requisito di sistema si occupa di garantire e consentire all'utente di modificare il proprio account, indicando uno o più campi da modificare: <ul style="list-style-type: none"> <li>• Avatar</li> <li>• E-mail</li> <li>• Nome</li> <li>• Cognome</li> <li>• Data di nascita</li> <li>• Genere</li> <li>• Password</li> </ul>

<b>Nome</b>	<b>RFU6-U   Abilita/Disabilita notifiche chat</b>
<b>Descrizione</b>	Tale requisito di sistema si occupa di garantire e consentire all'utente di abilitare e disabilitare le notifiche della chat all'interno della propria sezione account

<b>Nome</b>	<b>RFU7-U   Contatto sviluppatori</b>
<b>Descrizione</b>	Tale requisito di sistema si occupa di garantire e consentire all'utente di contattare gli sviluppatori della piattaforma all'interno dell'applicativo nella sezione specifica

<b>Nome</b>	<b>RFU8-U   Disconnessione dalla piattaforma</b>
<b>Descrizione</b>	Tale requisito di sistema si occupa di garantire e consentire all'utente di disconnettersi dalla piattaforma

<b>Nome</b>	<b>RFU9-U   Visualizzazione itinerari</b>
<b>Descrizione</b>	Tale requisito di sistema si occupa di garantire e consentire all'utente di visualizzare tutti gli itinerari presenti all'interno della piattaforma

<b>Nome</b>	<b>RFU10-U   Scarica informazioni itinerari</b>
<b>Descrizione</b>	Tale requisito di sistema si occupa di garantire e consentire all'utente di scaricare le informazioni degli itinerari, compresi quelli inseriti da lui

<b>Nome</b>	<b>RFU11-U   Invio messaggi in chat ad altri utenti</b>
<b>Descrizione</b>	Tale requisito di sistema si occupa di garantire e consentire all'utente di inviare messaggi in chat ad altri utenti per richiedere informazioni pertinenti a itinerari

<b>Nome</b>	<b>RFU12-U   Aggiunta/Rimozione punti di interesse</b>
<b>Descrizione</b>	Tale requisito di sistema si occupa di garantire e consentire all'utente di aggiungere o rimuovere punti di interesse pertinenti a itinerari, compresi quelli inseriti da lui

<b>Nome</b>	<b>RFU13-U   Visualizzazione recensioni</b>
<b>Descrizione</b>	Tale requisito di sistema si occupa di garantire e consentire all'utente di visualizzare le recensioni per ogni itinerario

<b>Nome</b>	<b>RFU14-U   Ricerca itinerari</b>
<b>Descrizione</b>	Tale requisito di sistema si occupa di garantire e consentire all'utente di effettuare delle ricerche degli itinerari presenti all'interno della piattaforma. La ricerca degli itinerari può essere fatta per nome e filtrata per livello e categoria

<b>Nome</b>	<b>RFU15-U   Visualizzazione dei propri itinerari</b>
<b>Descrizione</b>	Tale requisito di sistema si occupa di garantire e consentire all'utente di visualizzare i propri itinerari inseriti

<b>Nome</b>	<b>RFU16-U   Aggiunta itinerari</b>
<b>Descrizione</b>	<p>Tale requisito di sistema si occupa di garantire e consentire all'utente di aggiungere nuovi itinerari alla piattaforma, indicando:</p> <ul style="list-style-type: none"> <li>• Tracciato geografico (facoltativo) <ul style="list-style-type: none"> <li>– Inserimento manuale con mappa interattiva</li> <li>– File in formato GPX</li> </ul> </li> <li>• Punti di interesse (Facoltativo)</li> <li>• Informazioni itinerario <ul style="list-style-type: none"> <li>– Nome itinerario</li> <li>– Livello di difficoltà</li> <li>– CATEGORIA</li> <li>– Distanza</li> <li>– Durata</li> <li>– Descrizione (facoltativa)</li> </ul> </li> </ul>

<b>Nome</b>	<b>RFU17-U   Modifica dei propri itinerari</b>
<b>Descrizione</b>	<p>Tale requisito di sistema si occupa di garantire e consentire all'utente di modificare i propri itinerari inseriti. L'utente può modificare:</p> <ul style="list-style-type: none"> <li>• Tracciato geografico, se esiste</li> <li>• Informazioni itinerario</li> <li>• Punti di interessi</li> </ul>

<b>Nome</b>	<b>RFU18-U   Eliminazione di un proprio itinerario</b>
<b>Descrizione</b>	Tale requisito di sistema si occupa di garantire e consentire all'utente di eliminare un proprio itinerario inserito

<b>Nome</b>	<b>RFU1-A   Autenticazione alla piattaforma amministratore</b>
<b>Descrizione</b>	<p>Tale requisito di sistema si occupa di garantire e consentire all'amministratore di accedere alla piattaforma amministratore, indicando:</p> <ul style="list-style-type: none"> <li>• Username</li> <li>• Password</li> </ul>

<b>Nome</b>	<b>RFU2-A   Visualizzazione numero utenti registrati</b>
<b>Descrizione</b>	Tale requisito di sistema si occupa di garantire e consentire all'amministratore di visualizzare il numero di utenti registrati alla piattaforma

<b>Nome</b>	<b>RFU3-A   Visualizzazione numero accessi utenti</b>
<b>Descrizione</b>	Tale requisito di sistema si occupa di garantire e consentire all'amministratore di visualizzare il numero di accessi eseguiti da ogni utente

<b>Nome</b>	<b>RFU4-A   Visualizzazione numero itinerari</b>
<b>Descrizione</b>	Tale requisito di sistema si occupa di garantire e consentire all'amministratore di visualizzare il numero di itinerari per ogni categoria presenti all'interno della piattaforma

<b>Nome</b>	<b>RFU5-A   Visualizzazione numero recensioni</b>
<b>Descrizione</b>	Tale requisito di sistema si occupa di garantire e consentire all'amministratore di visualizzare il numero di recensioni per ogni livello di recensione presente

<b>Nome</b>	<b>RFU6-A   Visualizzazione numero ricerche</b>
<b>Descrizione</b>	Tale requisito di sistema si occupa di garantire e consentire all'amministratore di visualizzare il numero di ricerche fatte dall'utente per ogni itinerario

## 3.2 Requisiti non funzionali

Di seguito elenchiamo la descrizione dei seguenti requisiti non funzionali.

<b>Nome</b>	<b>RNFU1   Prestazioni di avvio</b>
<b>Descrizione</b>	Tale requisito di sistema si occupa di garantire un tempo di avvio dell'applicazione inferiore a 3 secondi nel 90% dei casi.

<b>Nome</b>	<b>RNFU2   Prestazioni caricamento mappa</b>
<b>Descrizione</b>	Tale requisito di sistema si occupa di garantire un tempo di caricamento della mappa inferiore a 2 secondi nel 90% dei casi.

<b>Nome</b>	<b>RNFU3   Prestazioni di notifiche push</b>
<b>Descrizione</b>	Tale requisito di sistema si occupa di garantire un tempo di ricezione di una notifica push inferiore a 3 secondi nel 90% dei casi.

<b>Nome</b>	<b>RNFU4   Prestazioni di ricerca</b>
<b>Descrizione</b>	Tale requisito di sistema si occupa di garantire un tempo di risposta di ricerca inferiore a 4 secondi nel 90% dei casi.

<b>Nome</b>	<b>RNFU5   Applicazione User-Friendly</b>
<b>Descrizione</b>	Tale requisito di sistema si occupa di garantire un'usabilità alta del sistema, l'utente deve ben comprendere tutte le funzionalità del sistema, interagendo con facilità in tutte le schermate dell'applicazione.

<b>Nome</b>	<b>RNFU6   Policy attributi utente in Registrazione</b>
<b>Descrizione</b>	Le informazioni inserite dall'utente devono rispettare determinati vincoli: <ul style="list-style-type: none"><li>• Nome<ul style="list-style-type: none"><li>– Lunghezza minima 2 caratteri</li><li>– Lunghezza massima 30 caratteri</li></ul></li><li>• Cognome<ul style="list-style-type: none"><li>– Lunghezza minima 2 caratteri</li><li>– Lunghezza massima 30 caratteri</li></ul></li><li>• Email<ul style="list-style-type: none"><li>– Formato valido</li></ul></li><li>• Password<ul style="list-style-type: none"><li>– Lunghezza minima 8 caratteri</li><li>– Almeno una lettera maiuscola</li><li>– Almeno un carattere speciale</li><li>– Almeno un numero</li></ul></li></ul>

<b>Nome</b>	<b>RNFU7   Policy password utente</b>
<b>Descrizione</b>	Tale requisito di sistema si occupa di garantire il salvataggio delle password utente attraverso un sistema che fornisce funzioni crittografiche. Questo requisito garantisce anche che nessun amministratore ha accesso alle password dell'utente

<b>Nome</b>	<b>RNFU8   Policy informazioni itinerario in Aggiunta Itinerario</b>
<b>Descrizione</b>	<p>Le informazioni inserite dall'utente devono rispettare determinati vincoli:</p> <ul style="list-style-type: none"> <li>• Nome itinerario <ul style="list-style-type: none"> <li>– Lunghezza minima 3 caratteri</li> <li>– Lunghezza massima 35 caratteri</li> </ul> </li> <li>• Descrizione <ul style="list-style-type: none"> <li>– Lunghezza massima 200 caratteri</li> </ul> </li> </ul>

<b>Nome</b>	<b>RNFU9   Policy accesso alla base di dati</b>
<b>Descrizione</b>	L'applicazione non deve comunicare direttamente con la base di dati, le operazioni devono quindi essere effettuate e controllate da un server back end.

<b>Nome</b>	<b>RNFU10   Utilizzo API Google Maps</b>
<b>Descrizione</b>	Tale requisito di sistema si occupa di garantire l'utilizzo della mappa tramite le API di Google Maps.

<b>Nome</b>	<b>RNFU11   Portabilità sistema in ambienti diversi</b>
<b>Descrizione</b>	Tale requisito di sistema si occupa di garantire che l'applicazione funziona anche in un ambiente diverso.

<b>Nome</b>	<b>RNFU12   Adattabilità sistema in ambienti diversi</b>
<b>Descrizione</b>	Tale requisito di sistema si occupa di garantire che l'applicazione ha quella capacità di adattarsi al cambiamento in un ambiente senza alcun cambiamento nel suo comportamento.

### 3.3 Requisiti di dominio

Di seguito elenchiamo la descrizione dei seguenti requisiti di dominio.

<b>Nome</b>	<b>RDO1   ISO 25010:2020</b>
<b>Descrizione</b>	Tale requisito afferma che il sistema deve essere conforme allo standard ISO 25010:2020, requisiti della qualità di sistemi e del software.

<b>Nome</b>	<b>RDO2   ISO 27001:2013</b>
<b>Descrizione</b>	Tale requisito afferma che il sistema deve essere conforme allo standard ISO 27001:2013 sulla sicurezza delle informazioni.

<b>Nome</b>	<b>RDO3   ISO 27018:2020</b>
<b>Descrizione</b>	Tale requisito afferma che il sistema deve essere conforme allo standard ISO 27018:2020 sulla protezione dei dati personali trattati in cloud.

<b>Nome</b>	<b>RDO4   ISO 9241-110:2020</b>
<b>Descrizione</b>	Tale requisito afferma che il sistema deve essere conforme allo standard ISO 9241-110:2020 sui principi alla base dell'interazione tra un utente e un sistema.

<b>Nome</b>	<b>RDO5   ISO 27701:2019</b>
<b>Descrizione</b>	Tale requisito afferma che il sistema deve essere conforme allo standard ISO 27701:2019 estensione della precedente ISO27001:2013 per la protezione dei dati personali, migliorando il sistema di gestione della sicurezza delle informazioni e utile per la conformità al GDPR

## 3.4 Use Case Diagram

Di seguito viene riportato il diagramma dei casi d'uso relativo all'applicazione.



## 3.5 Tabelle Di CockBurn

### 3.5.1 Inserisce nuovi itinerari

Use Case #1	Inserisce nuovi itinerari	
Goal	L'utente vuole inserire nuovi itinerari (sentieri), caratterizzandoli da un nome, durata, livello di difficoltà, punto di inizio, una descrizione e da un tracciato geografico (facoltativo) inserito manualmente con una mappa interattiva oppure tramite file GPX.	
Preconditions	L'utente deve essere autenticato.	
Success End Conditions	L'utente inserisce nuovi itinerari (sentieri), caratterizzandoli con diversi attributi di appartenenza.	
Failed End Conditions	L'utente non inserisce un nuovo itinerario.	
Actor	Utente autenticato.	
Trigger	Pressione sul pulsante (+) <b>Aggiungi Itinerario</b> in <b>W-i-miel-Itinerari</b>	
<b>Step</b>	<b>Utente autenticato</b>	<b>Sistema</b>
1	Preme sul pulsante (+) <b>Aggiungi Itinerario</b> in <b>W-i-miel-Itinerari</b>	Mostra il <b>W-Tracciato</b>
2		Mostra il <b>W-Disegna-Tracciato</b>
3	Preme sul pulsante <b>Disegna Tracciato</b>	Mostra il <b>W-Disegna-Tracciato</b>
4		Mostra il <b>W-Disegna-Tracciato-2</b>
5	Disegna tracciato geografico manualmente con una mappa interattiva	
6	Preme sul pulsante <b>Conferma tracciato</b>	
7		Mostra il <b>W-Disegna-Tracciato-2</b>
8	Preme sul pulsante <b>Avanti</b>	Mostra il <b>W-Tracciato-2</b>
9		Mostra il <b>W-Tracciato-2</b>
10	Preme sul pulsante <b>Avanti</b>	Mostra il <b>W-Punti-di-Interesse</b>
11		Mostra il <b>W-Tracciato-2</b>
12	Preme sul pulsante <b>Avanti</b>	Mostra il <b>W-Informazioni-Itinerario</b>
13		Mostra il <b>W-Informazioni-Itinerario</b>
14	Inserisce correttamente le informazioni dell'itinerario: nome (itinerario, livello, categoria, distanza, durata, descrizione (facoltativa))	
15	Preme sul pulsante <b>Inserisci Itinerario</b>	
16		Verifica la validità di ogni campo inserito
17		Mostra un messaggio di dialog di conferma inserimento itinerario
18		Mostra il <b>W-i-miel-Itinerari</b>
Extension #1: Non vuole inserire l'itinerario		
3.1	Preme sul pulsante <b>Torna indietro</b>	
4.1		Mostra il <b>W-i-miel-Itinerari</b>
Extension #2: Non vuole disegnare un tracciato geografico		
5.2	Preme sul pulsante <b>Torna indietro</b>	
6.2		Mostra il <b>W-Tracciato</b>
Extension #3: Non inserisce tutti i campi obbligatori		
14.3	Non inserisce uno o più campi obbligatori e preme sul pulsante <b>Inserisci Itinerario</b>	
15.3		Mostra messaggi di errore vicino ai campi non inseriti
Extension #4: Inserisce un nome dell'itinerario troppo lungo		
14.4	Inserisce un nome dell'itinerario che supera i 40 caratteri	
15.4		Mostra messaggio di errore vicino al campo di testo corrispondente (nome itinerario)
Extension #5: Inserisce una descrizione dell'itinerario troppo lunga		
14.5	Inserisce una descrizione dell'itinerario che supera i 400 caratteri	
15.5		Mostra un messaggio di errore vicino al campo di testo corrispondente (descrizione)
Sub-Variations #1: Carica il percorso con file GPX		
3.2	Preme sul pulsante <b>Carica Percorso GPX</b>	
4.2		Mostra l'interfaccia <b>SO</b> per caricare il percorso in file GPX
5.2	Sceglie il file da caricare	

*Continua a pagina successiva*

5.2	Sceglie il file da caricare	
6.2		Carica il percorso e mostra il <b>W-Tracciato-2</b> . Torna allo step 8 del main scenario.
Sub-Variations #2: Annulla l'inserimento del marker durante il disegno del tracciato geografico		
6.1	Preme sul pulsante <b>Annulla</b>	
7.1		Mostra il <b>W-Disegna-Tracciato</b> con il marker eliminato
Sub-Variations #3: Visualizza o modifica un percorso disegnato manualmente		
10.3	Preme sul pulsante <b>Visualizza o Modifica</b>	
11.3		Mostra il <b>W-Disegna-Tracciato</b>
Sub-Variations #4: Elimina un percorso disegnato manualmente		
10.4	Preme sul pulsante <b>Elimina</b>	
11.4		Mostra il <b>W-Tracciato</b>
Sub-Variations #5: Visualizza o modifica un percorso GPX		
10.5	Preme sul pulsante <b>Visualizza o Modifica</b>	
11.5		Mostra il <b>W-Disegna-Tracciato</b>
Sub-Variations #6: Elimina un percorso GPX		
10.6	Preme sul pulsante <b>Elimina</b>	
11.6		Mostra il <b>W-Tracciato</b>

### 3.5.2 Invia messaggi privati

Use Case #2	Invia messaggi privati	
Goal	L'utente vuole inviare un messaggio privato a un altro utente per chiedere informazioni su un itinerario da lui inserito.	
Preconditions	L'utente deve essere autenticato.	
Success End Conditions	L'utente invia un messaggio privato a un altro utente.	
Failed End Conditions	L'utente non invia un messaggio privato.	
Actor	Utente autenticato.	
Trigger	Pressione sul pulsante Chat in W-Itinerario	
Step	Utente autenticato	Sistema
1	Sceglie un itinerario dalla lista in W-Cerca-Itinerario	Mostra il W-Itinerario
2	Preme sul pulsante Chat	Mostra il W-Invio-Chat-Utente
3	Scrive messaggio nel campo di testo	
4	Preme sul pulsante Invia messaggio	Mostra il W-Invio-Chat-Utente-2
5		
Extension #1: Non vuole inviare messaggio privato		
5.1	Preme sul pulsante Torna indietro	
6.1		Mostra il W-Itinerario
Sub-Variations #1: Invia messaggio privato a un utente con chat già esistente		
1.1	Preme sul pulsante Chat presente sulla Bottom Bar in W-Cerca-Itinerario	
2.1		Mostra il W-Invio-Chat
3.1	Selezione o ricerca il messaggio nella lista	
4.1		Mostra il W-Invio-Chat-Utente-3. Torna allo step 5 del main scenario.
Sub-Variations #2: Invia immagine ad un utente		
5.2	Preme sul pulsante Immagine	
6.2		Mostra l'interfaccia SO per caricare un'immagine
7.2	Selezione l'immagine da inviare	
8.2		Mostra il W-Invio-Chat-Utente-4

### 3.5.3 Lettura messaggi privati ricevuti

Use Case #3	Legge messaggi privati ricevuti	
Goal	L'utente legge un messaggio privato ricevuto da un altro utente per richiedere informazioni su un itinerario inserito.	
Preconditions	L'utente deve essere autenticato.	
Success End Conditions	L'utente legge un messaggio privato ricevuto da un altro utente.	
Failed End Conditions	L'utente non legge un messaggio privato.	
Actor	Utente autenticato.	
Trigger	Pressione sul pulsante <b>Chat</b> in W-Home	
Step	Utente autenticato	Sistema
1	Preme sul pulsante <b>Chat</b> presente sulla Bottom Bar in W-Home	
2		Mostra il W-Riceve-Chat
3	Seleziona il messaggio ricevuto	
4		Mostra il W-Riceve-Chat-Utente
5	Legge il messaggio ricevuto	
Extension #1: Non legge messaggio ricevuto		
3.1	Preme sul pulsante <b>Home</b> presente sulla Bottom Bar in W-Home	
4.1		Mostra il W-Home
Sub-Variations #1: Legge messaggio dalla notifica		
1.2	Legge messaggio dalla notifica in W-Home-Notifica	
2.2	Preme sulla notifica in W-Home-Notifica	
		Mostra il W-Riceve-Chat-Utente

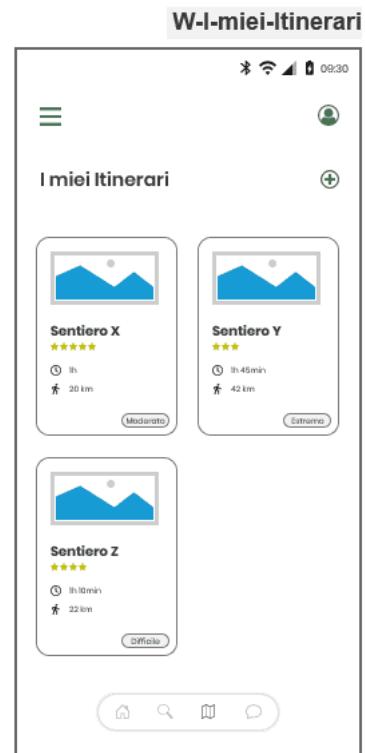
## 3.6 Wireframe

Di seguito vengono rappresentati i Wireframe dell'applicazione mobile derivati dalle descrizioni testuali (Cockburn) rappresentate nel capitolo precedente. I wireframe presi in considerazione sono quelli di media fedeltà. La rappresentazione dell'intero processo della UI (Wireframe di bassa fedeltà, media fedeltà e Mockups finali) è esposta nel capito 4 dedicato alla relazione di usabilità.

### 3.6.1 W-I-Miei-Itinerari

Il wireframe in figura mostra la disposizione degli elementi all'interno della schermata dedicata alla visualizzazione dei propri itinerari. Gli elementi presenti sono:

- 3 tasti funzionali. Rispettivamente *apri menu*, *il mio profilo* ed *aggiungi itinerario* che porterà l'utente alla schermata W-Tracciato.
- Una lista di card cliccabili che da l'accesso alla visualizzazione dell'itinerario.
- Una barra fluttuante composta da tasti, con la finalità di navigare tra le schermate.



### 3.6.2 W-Tracciato

Il wireframe in figura mostra la disposizione degli elementi all'interno della schermata dedicata all'inserimento / caricamento di un tracciato geografico. Gli elementi presenti sono:

- 2 immagini cliccabili. Rispettivamente *disegna tracciato geografico* che porterà l'utente alla schermata W-Disegna Tracciato, *carica tracciato (gpx)*.
- 1 tasto. Rispettivamente *torna alla schermata precedente*
- Una barra fluttuante composta da tasti, con la finalità di navigare tra le schermate di inserimento itinerario.

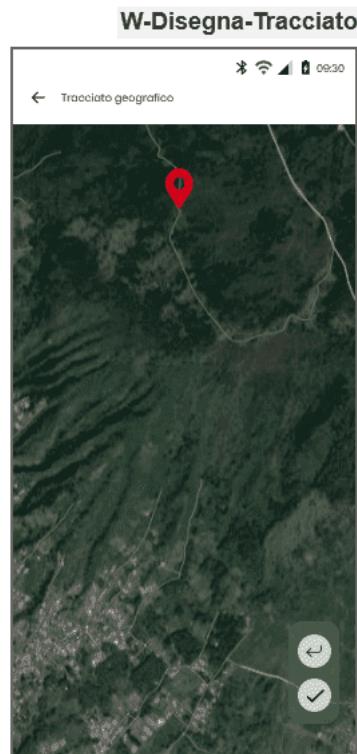


### 3.6.3 W-Disegna-Tracciato

Il wireframe in figura mostra la disposizione degli elementi all'interno della schermata dedicata nel disegnare il tracciato geografico.

Gli elementi presenti sono:

- 3 tasti funzionali. Rispettivamente *confirma segna posto* che riporterà l'utente alla schermata W-Tracciato, *torna al segna posto precedente*, *torna alla schermata precedente*.

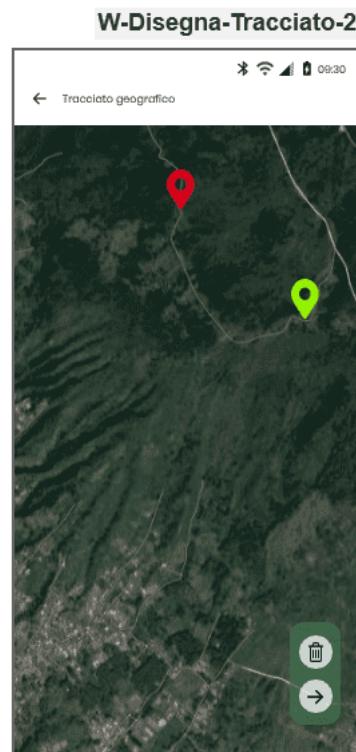


### 3.6.4 W-Disegna-Tracciato-2

Il wireframe in figura mostra la disposizione degli elementi all'interno della schermata dedicata alla conferma del tracciato geografico.

Gli elementi presenti sono:

- 3 tasti funzionali. Rispettivamente *conferma tracciato* che porterà l'utente alla schermata W-Tracciato-2, *elimina tracciato* che riporterà l'utente alla schermata W-Disegna-Tracciato, *torna alla schermata precedente*.



### 3.6.5 W-Tracciato-2

Il wireframe in figura mostra la disposizione degli elementi all'interno della schermata dedicata alla modifica / eliminazione del tracciato geografico. Gli elementi presenti sono:

- 2 immagini cliccabili. Rispettivamente *modifica disegno tracciato*, *elimina tracciato*.
- 1 tasto. Rispettivamente *torna alla schermata precedente*
- Una barra fluttuante composta da tasti, con la finalità di navigare tra le schermate di inserimento itinerario.



### 3.6.6 W-Punti-Di-Interesse

Il wireframe in figura mostra la disposizione degli elementi all'interno della schermata dedicata all'aggiunta dei punti di interesse. Gli elementi presenti sono:

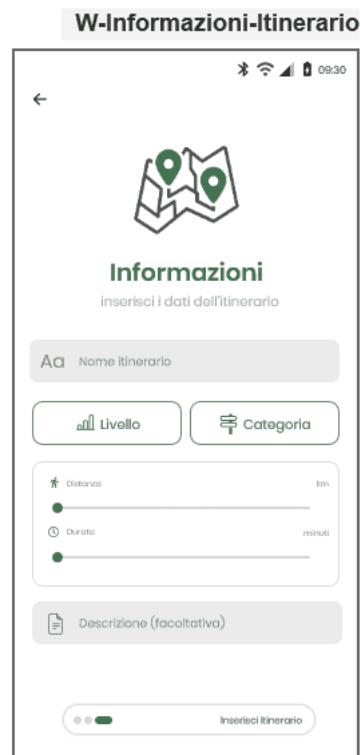
- 2 tasti. Rispettivamente *torna alla schermata precedente, aggiungi punti di interesse*
- Una barra fluttuante composta da tasti, con la finalità di navigare tra le schermate di inserimento itinerario.



### 3.6.7 W-Informazioni-Itinerario

Il wireframe in figura mostra la disposizione degli elementi all'interno della schermata dedicata all'inserimento dei dettagli dell'itinerario. Gli elementi presenti sono:

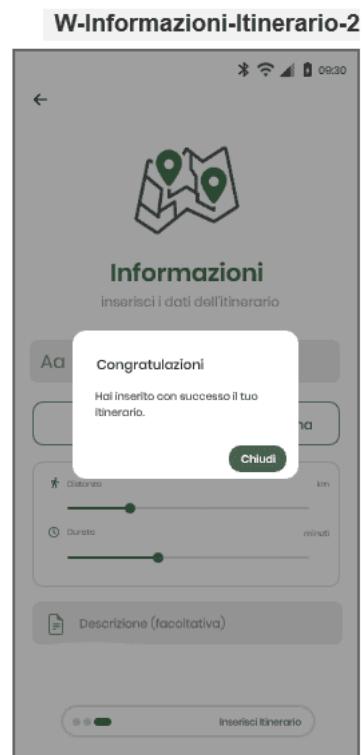
- 2 campi di testo. Rispettivamente *nome dell'itinerario, descrizione*
- 3 tasti. Rispettivamente *seleziona livello di difficoltà, seleziona categoria, torna alla schermata precedente*
- 2 slider. Rispettivamente *imposta lunghezza dell'itinerario, imposta durata dell'itinerario*
- Una barra fluttuante composta da tasti, con la finalità di navigare tra le schermate di inserimento itinerario e di inserimento dell'itinerario.



### 3.6.8 W-Informazioni-Itinerario-2

Il wireframe in figura mostra la disposizione degli elementi all'interno della schermata di successo dell'inserimento dell'itinerario. Gli elementi presenti sono:

- 1 finestra di dialog con messaggi di sistema.
- 1 Tasto *chiudi* che riporterà l'utente alla schermata W-I-Miei-Itinerari.



### 3.6.9 W-Cerca-Itinerario

Il wireframe in figura mostra la disposizione degli elementi all'interno della schermata dedicata alla ricerca degli itinerario. Gli elementi presenti sono:

- 3 tasti funzionali. Rispettivamente *apri menu*, *il mio profilo* ed *filtri di ricerca* che porterà l'utente alla schermata W-Cerca-Itinerario-2.
- Una lista di card cliccabili come risultato di ricerca che porteranno l'utente alla schermata W-Itinerario se cliccate.
- Una barra fluttuante composta da tasti, con la finalità di navigare tra le schermate.

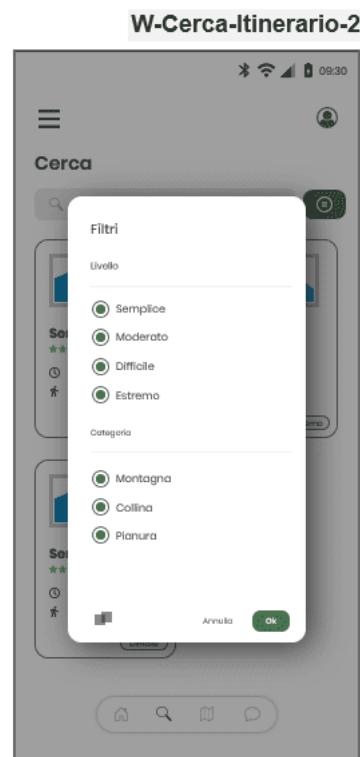


### 3.6.10 W-Cerca-Itinerario-2

Il wireframe in figura mostra la disposizione degli elementi all'interno della schermata dedicata alla selezione dei filtri di ricerca.

Gli elementi presenti sono:

- 1 finestra di dialogo con messaggi di sistema.
- 3 tasti. Rispettivamente *conferma*, *annulla*, *seleziona / deseleziona tutti*
- Una lista di radio button rappresentati i filtri di ricerca.



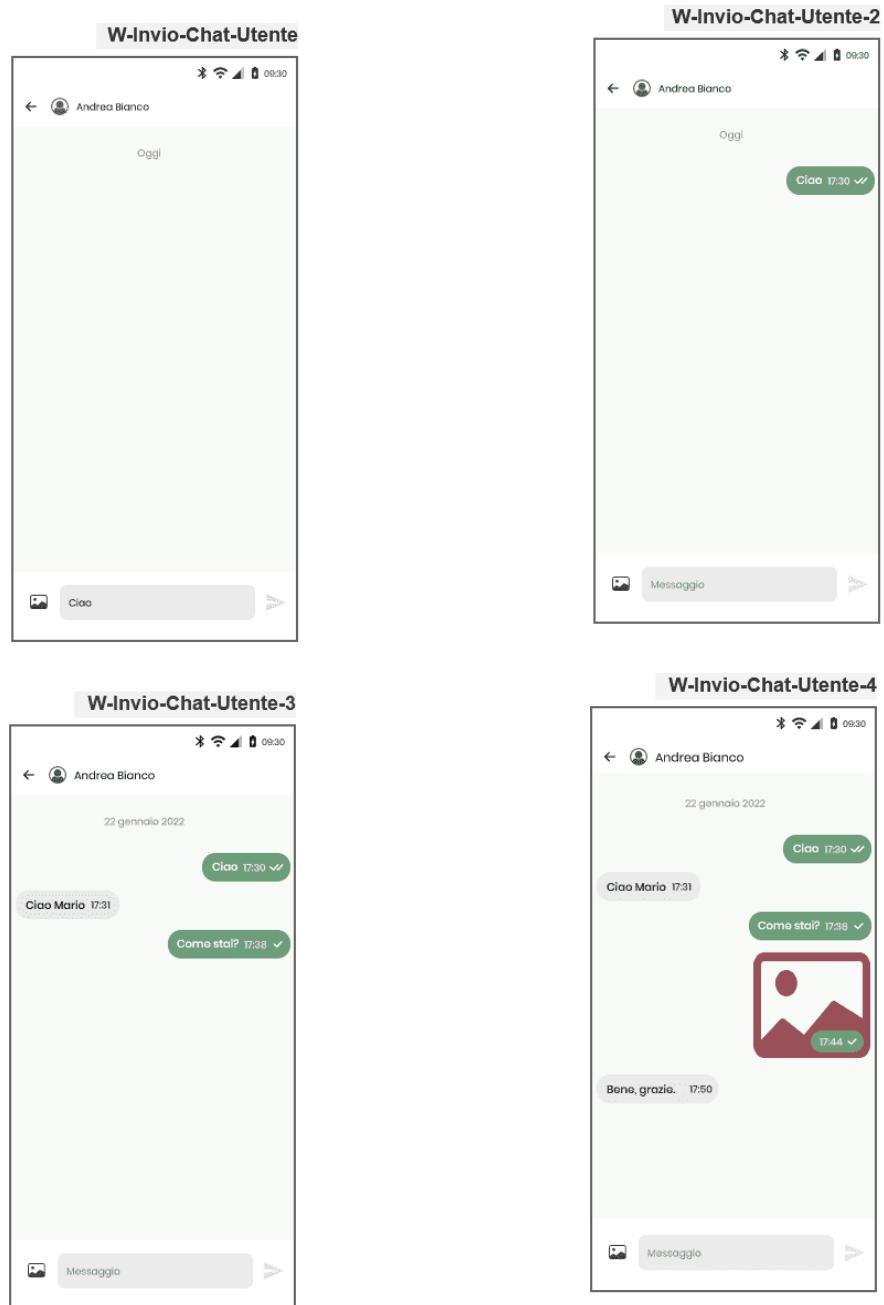
### 3.6.11 W-Itinerario

Il wireframe in figura mostra la disposizione degli elementi all'interno della schermata dedicata alla visualizzazione del singolo itinerario. Gli elementi presenti sono:

- 4 tasti funzionali. Rispettivamente *salva dettagli tracciato*, *chat* che porterà l'utente alla schermata W-Invio-Chat-Utente, *torna alla schermata precedente*, *modifica punti di interesse*.



### 3.6.12 W-Invio-Chat-Utente



Il wireframe in figura mostra la disposizione degli elementi all'interno della schermata dedicata all'invio dei messaggi con un utente. Gli elementi presenti sono:

- Un'area di testo utile a digitare il testo del messaggio.
- 3 tasti funzionali. Rispettivamente *invia messaggio*, *invia immagine*, *torna alla schermata precedente*.
- Una lista contenente i messaggi inviati e ricevuti dall'utente.

### 3.6.13 W-Invio-Chat

Il wireframe in figura mostra la disposizione degli elementi all'interno della schermata dedicata alla lista delle chat attive. Gli elementi presenti sono:

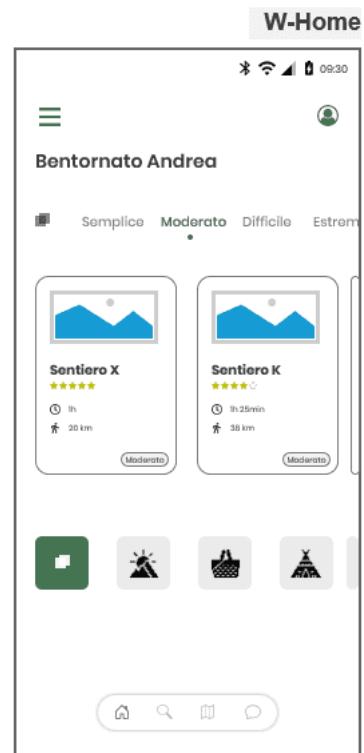
- Una lista contenente tutte le chat aperte dall'utente
- 3 tasti. Rispettivamente *apri il menù, il mio profilo, cerca nelle chat*
- Una barra fluttuante composta da tasti, con la finalità di navigare tra le schermate di inserimento itinerario.



### 3.6.14 W-Home

Il wireframe in figura mostra la disposizione degli elementi all'interno della schermata dedicata alla home page dell'app. Gli elementi presenti sono:

- 5 tasti funzionali. Rispettivamente *apri menu*, *il mio profilo*, *seleziona difficoltà*, *deseleziona difficoltà*, *seleziona punti di interesse*.
- Una lista di card cliccabili che da l'accesso alla visualizzazione dell'itinerario.
- Una barra fluttuante composta da tasti, con la finalità di navigare tra le schermate.



### 3.6.15 W-Riceve-Chat

Il wireframe in figura mostra la disposizione degli elementi all'interno della schermata dedicata alla lista di tutte le chat aperte. Gli elementi presenti sono:

- Una lista contenente tutte le chat aperte dall'utente
- 3 tasti. Rispettivamente *apri il menù, il mio profilo, cerca nelle chat*
- Una barra fluttuante composta da tasti, con la finalità di navigare tra le schermate di inserimento itinerario.



### 3.6.16 W-Ricevete-Chat-Utente

Il wireframe in figura mostra la disposizione degli elementi all'interno della schermata dedicata all'invio dei messaggi con un utente.

Gli elementi presenti sono:

- Un'area di testo utile a digitare il testo del messaggio.
- 3 tasti funzionali. Rispettivamente *invia messaggio*, *invia immagine*, *torna alla schermata precedente*.
- Una lista contenente i messaggi inviati e ricevuti dall'utente.

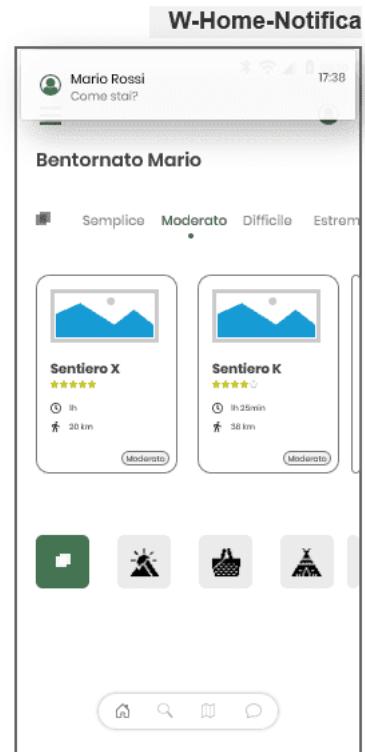


### 3.6.17 W-Home-Notifica

Il wireframe in figura mostra la disposizione degli elementi all'interno della schermata dedicata alla ricezione di una notifica di chat.

Gli elementi presenti sono:

- Una notifica a comparsa contenente il Mitente e il contenuto del messaggio.



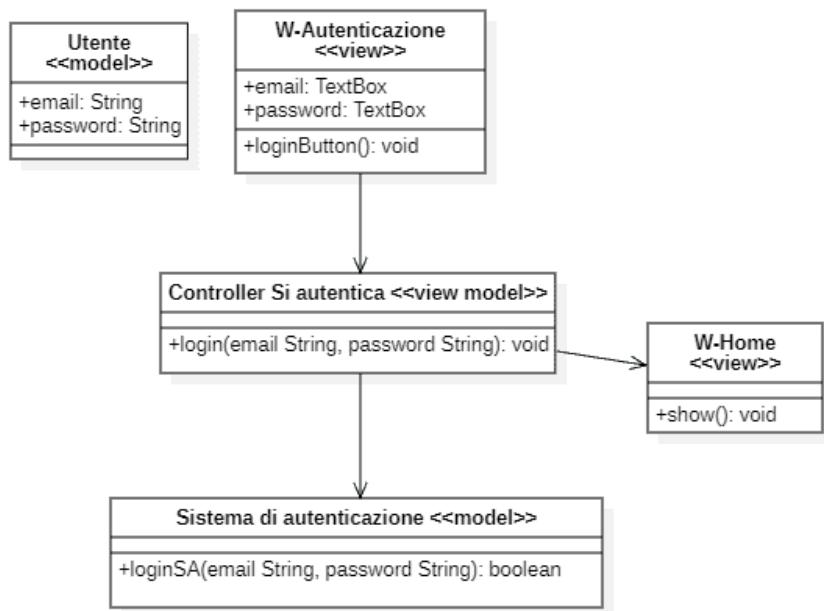
# 4. Modelli di dominio

Di seguito vengono presentati i modelli di dominio dell'applicazione.

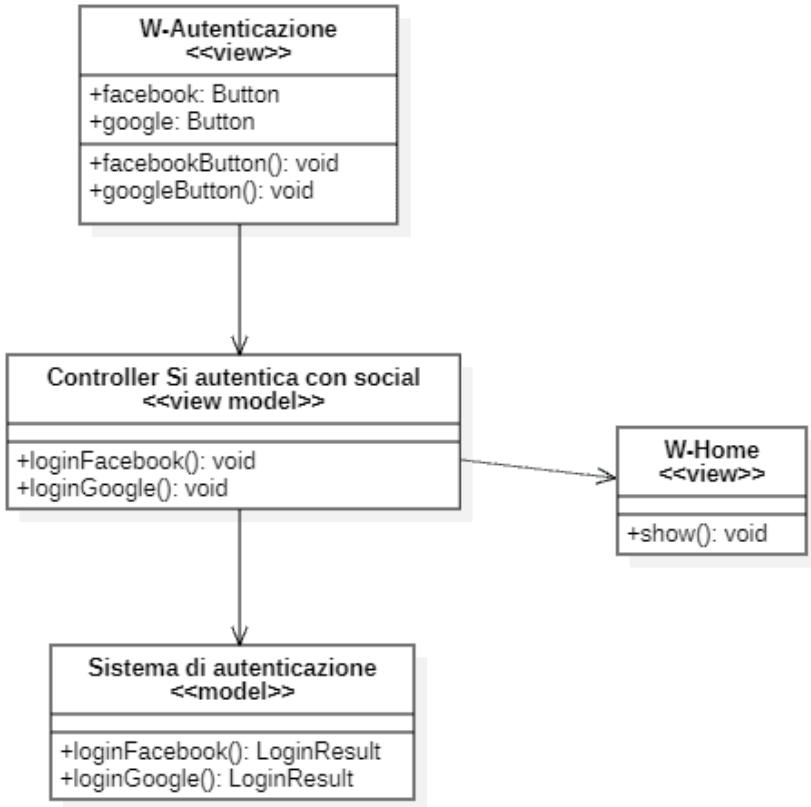
I modelli di dominio saranno divisi in tre sottosezioni: *Diagrammi di classi di analisi*, *Diagrammi di sequenza di analisi* e *Diagrammi di attività*.

## 4.1 Diagrammi di classi di analisi

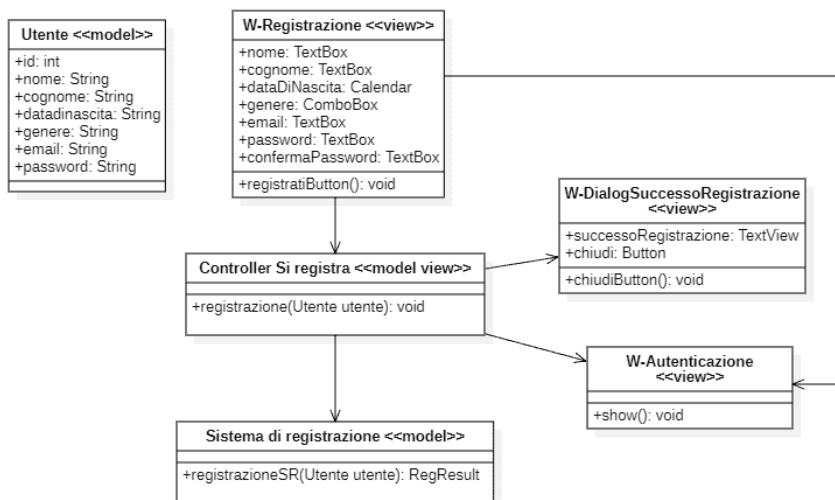
### 4.1.1 Si autentica



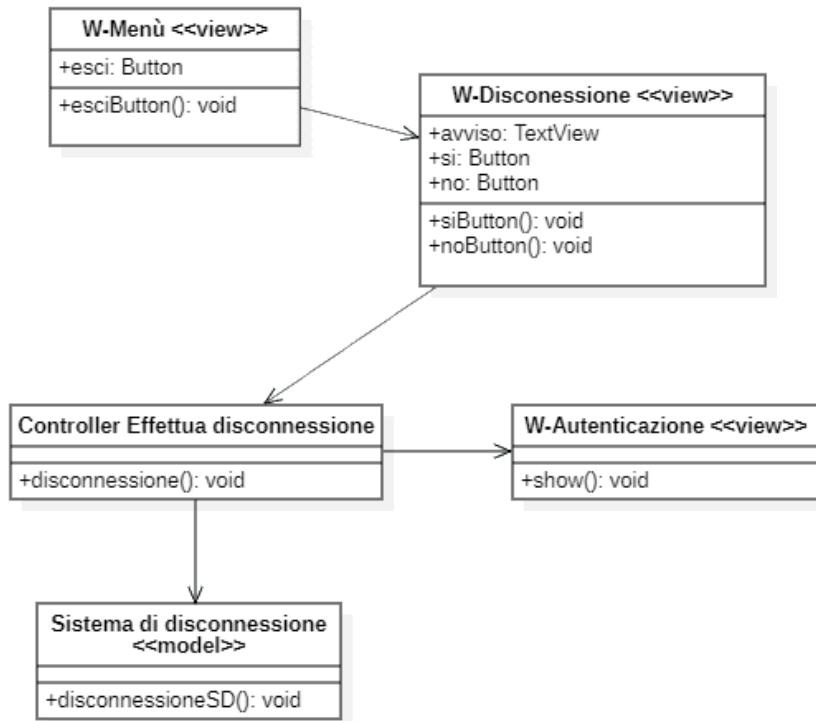
#### 4.1.2 Si autentica con social



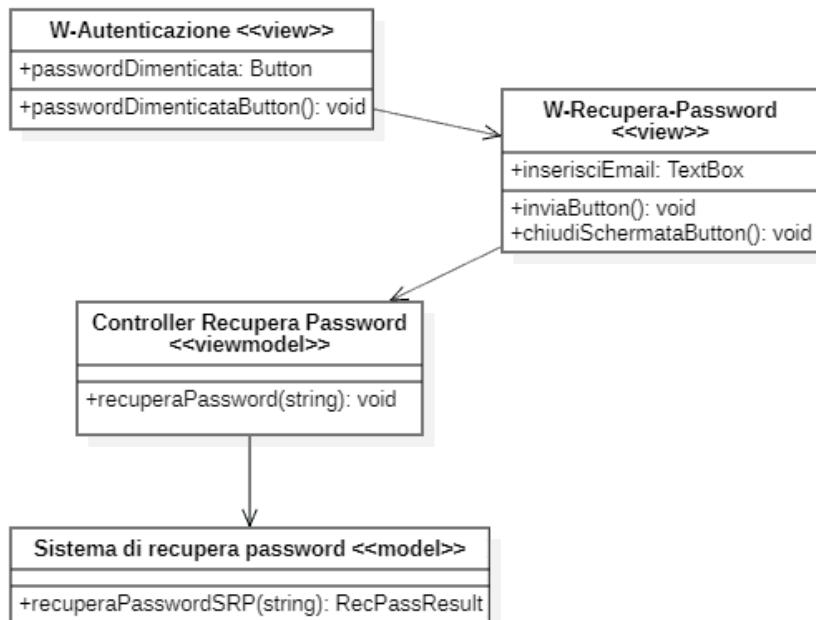
#### 4.1.3 Si registra



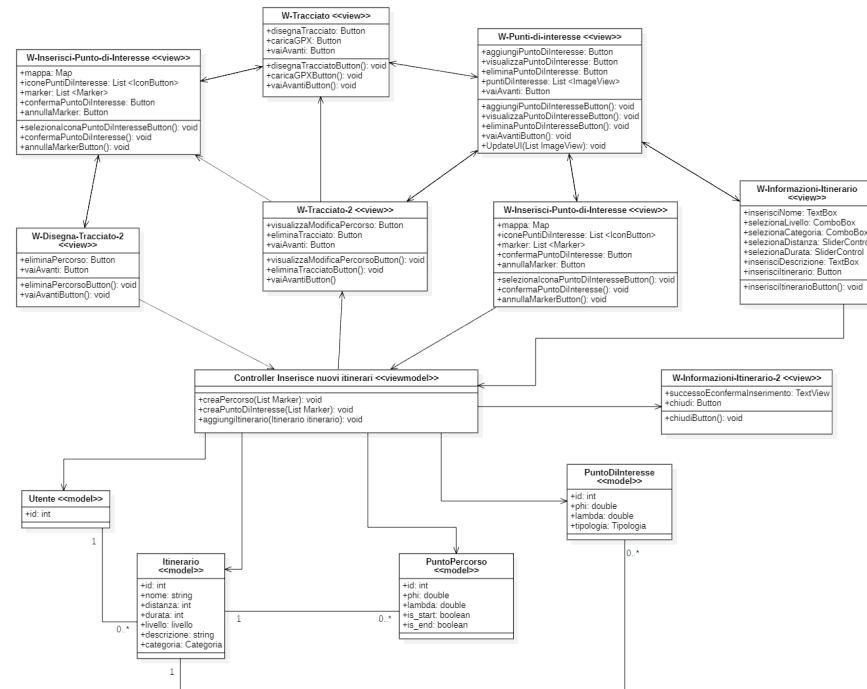
#### 4.1.4 Effettua disconnessione



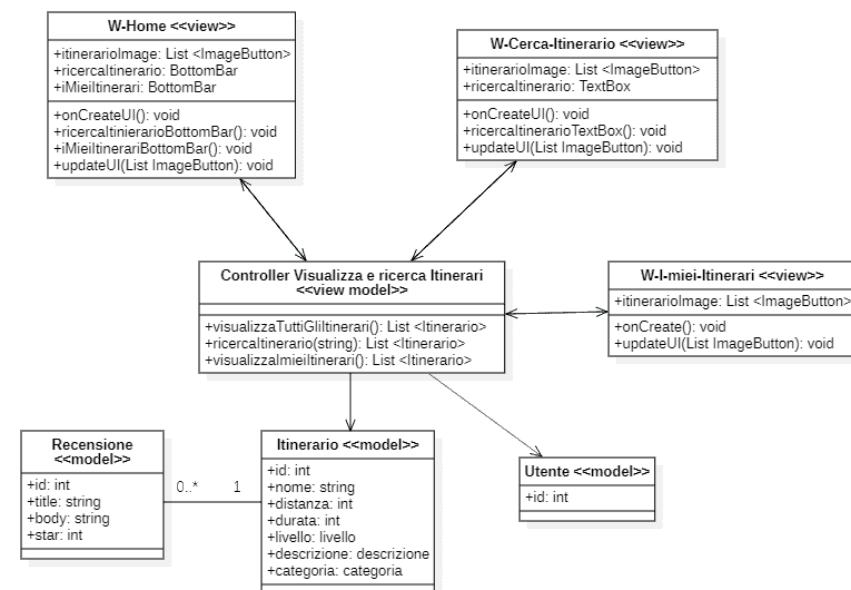
#### 4.1.5 Recupera Password



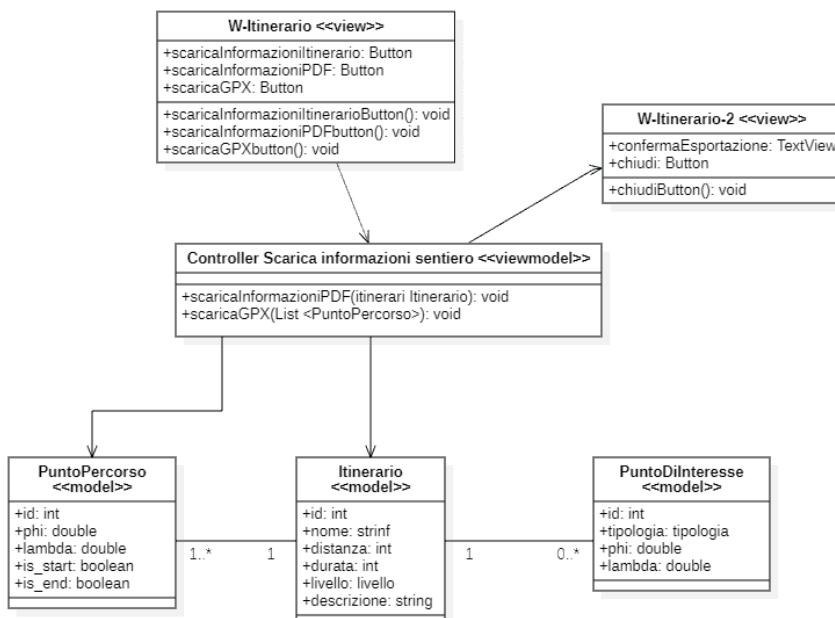
## 4.1.6 Inserisce nuovi itinerari



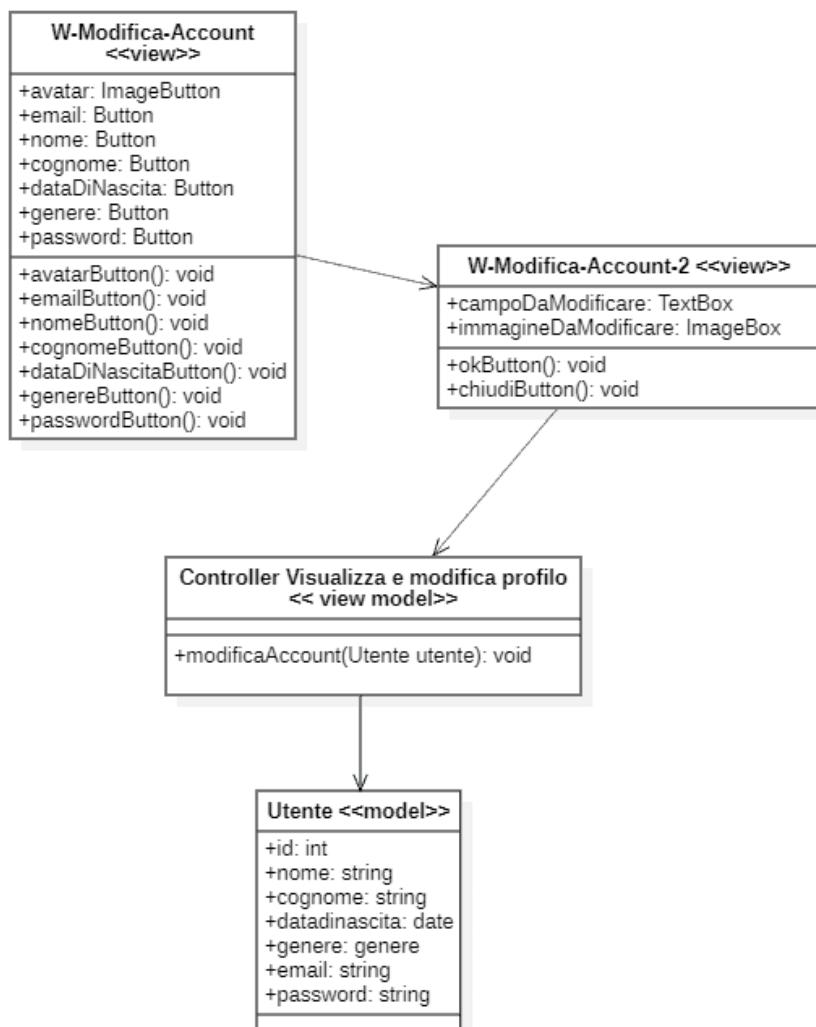
## 4.1.7 Visualizza e ricerca Itinerari



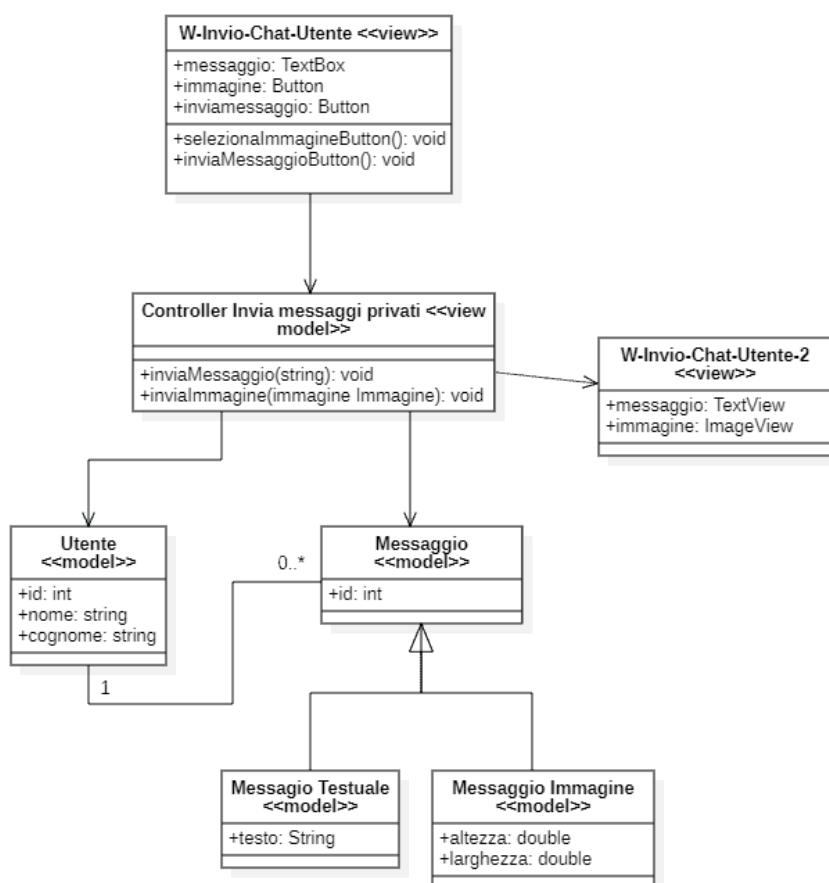
#### 4.1.8 Scarica informazioni sentiero



#### 4.1.9 Visualizza e modifica profilo

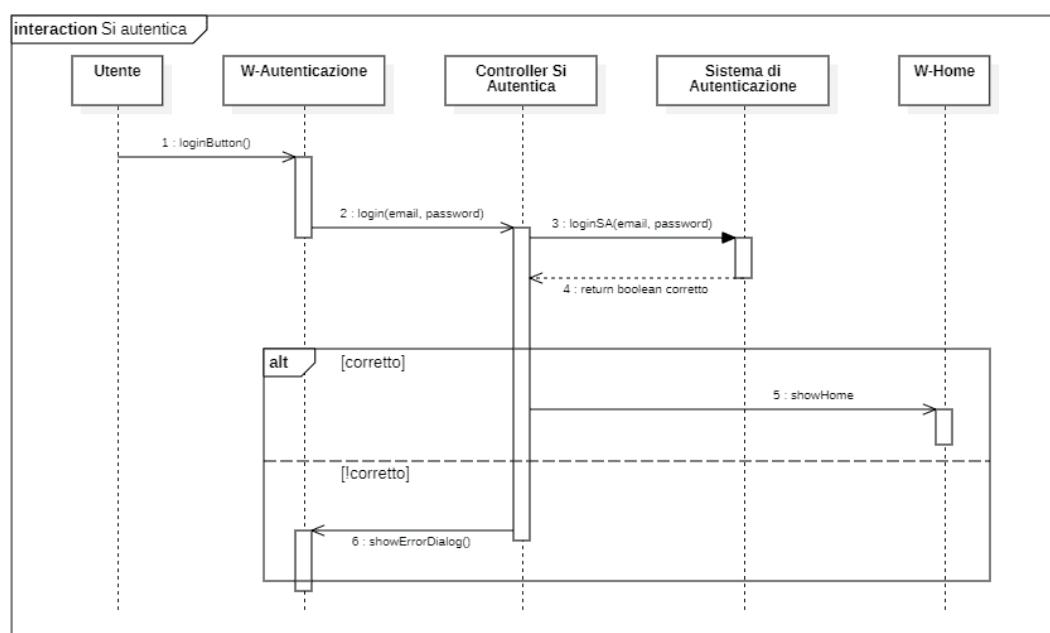


#### 4.1.10 Invia messaggi privati

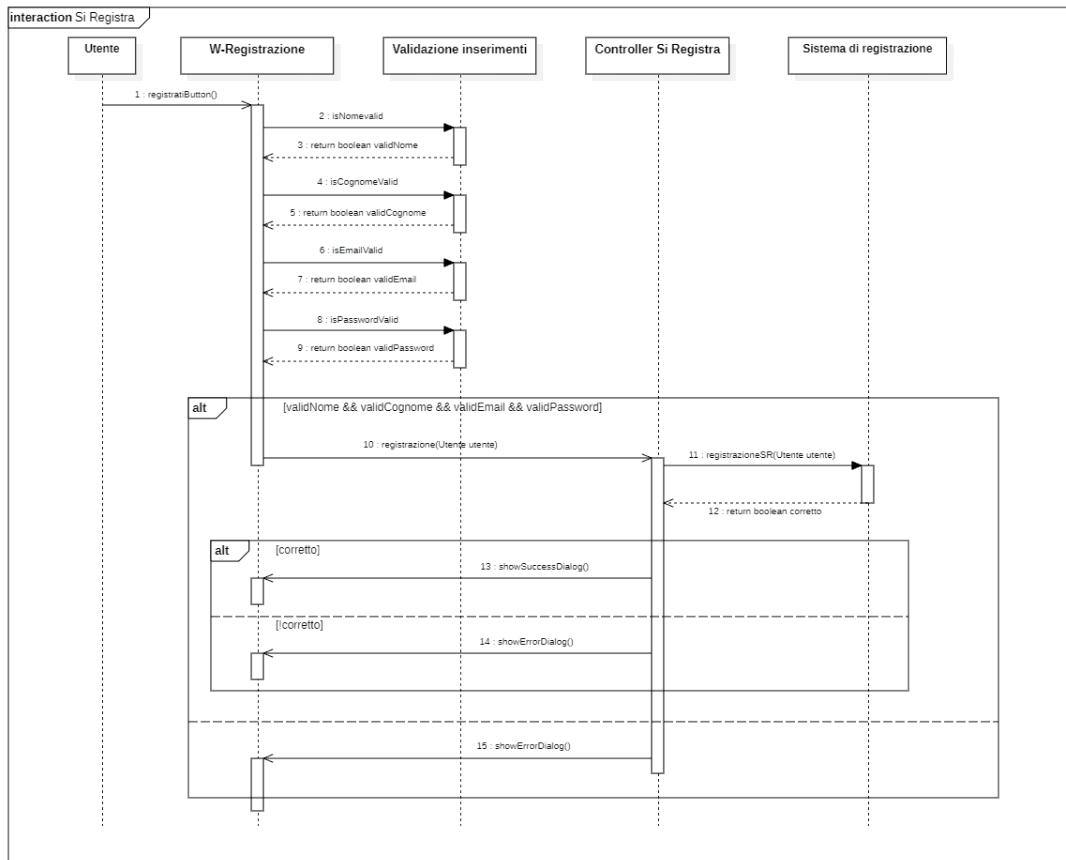


## 4.2 Diagrammi di sequenza di analisi

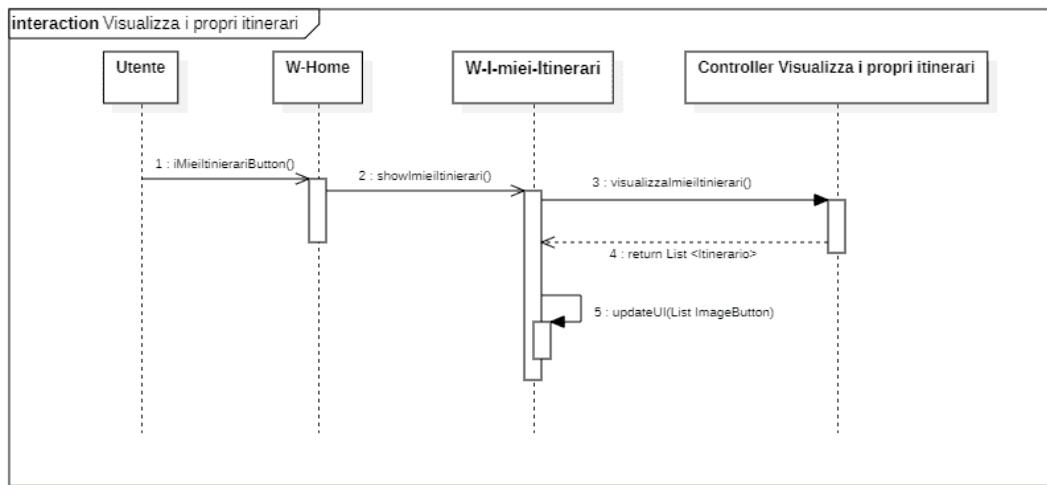
### 4.2.1 Si Autentica



## 4.2.2 Si Registra

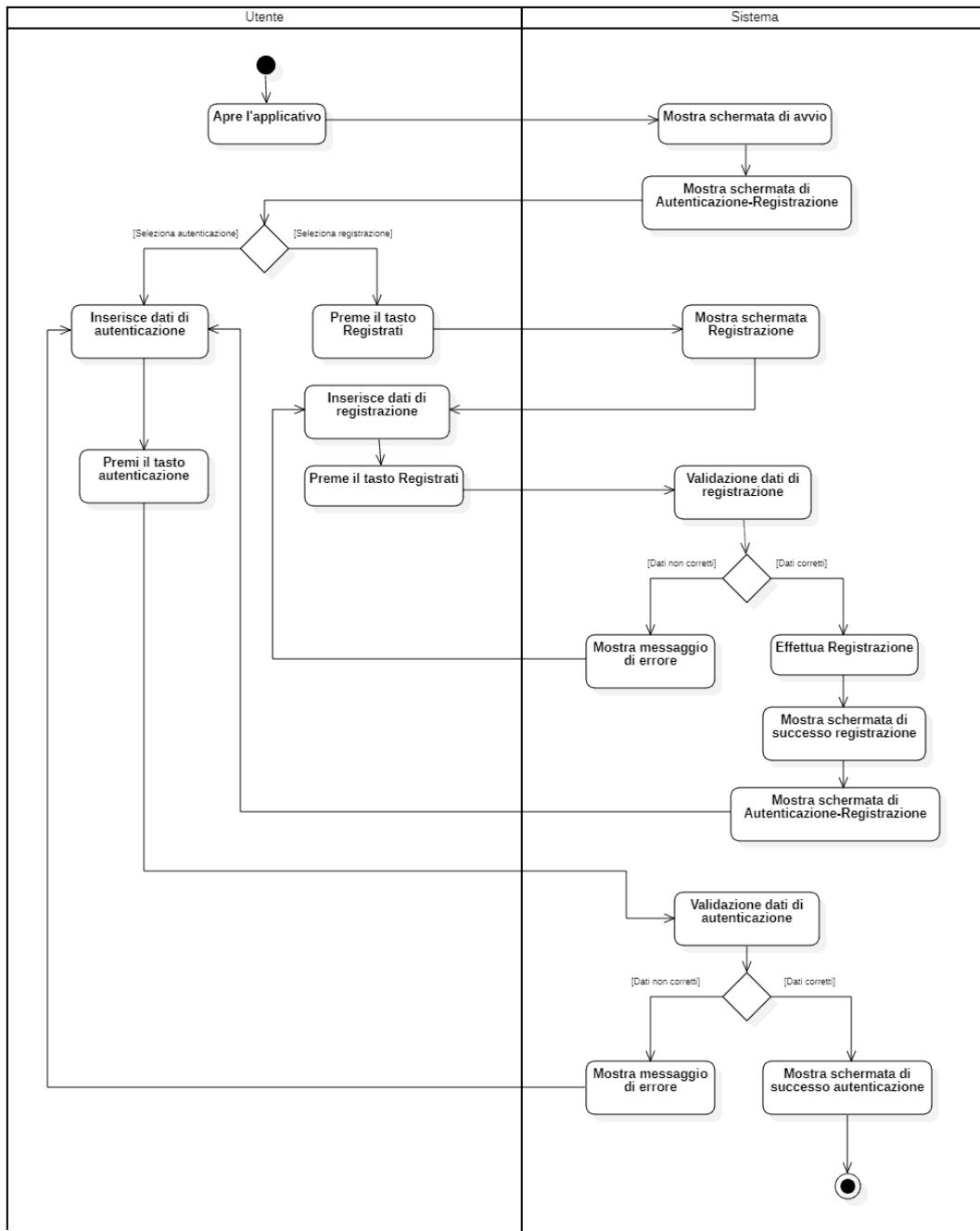


### 4.2.3 Visualizza I Proprio Itinerari

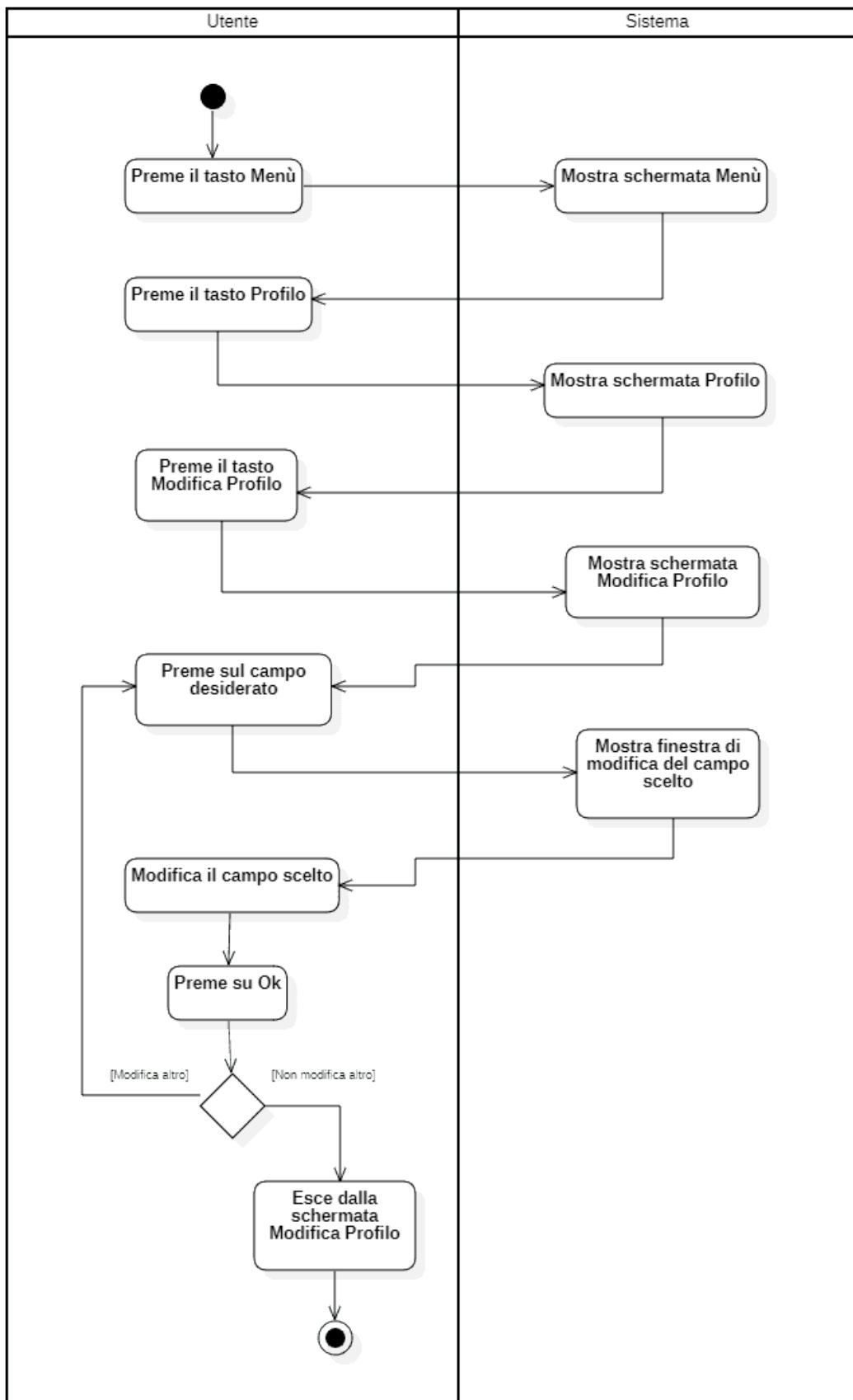


## 4.3 Diagrammi di attività

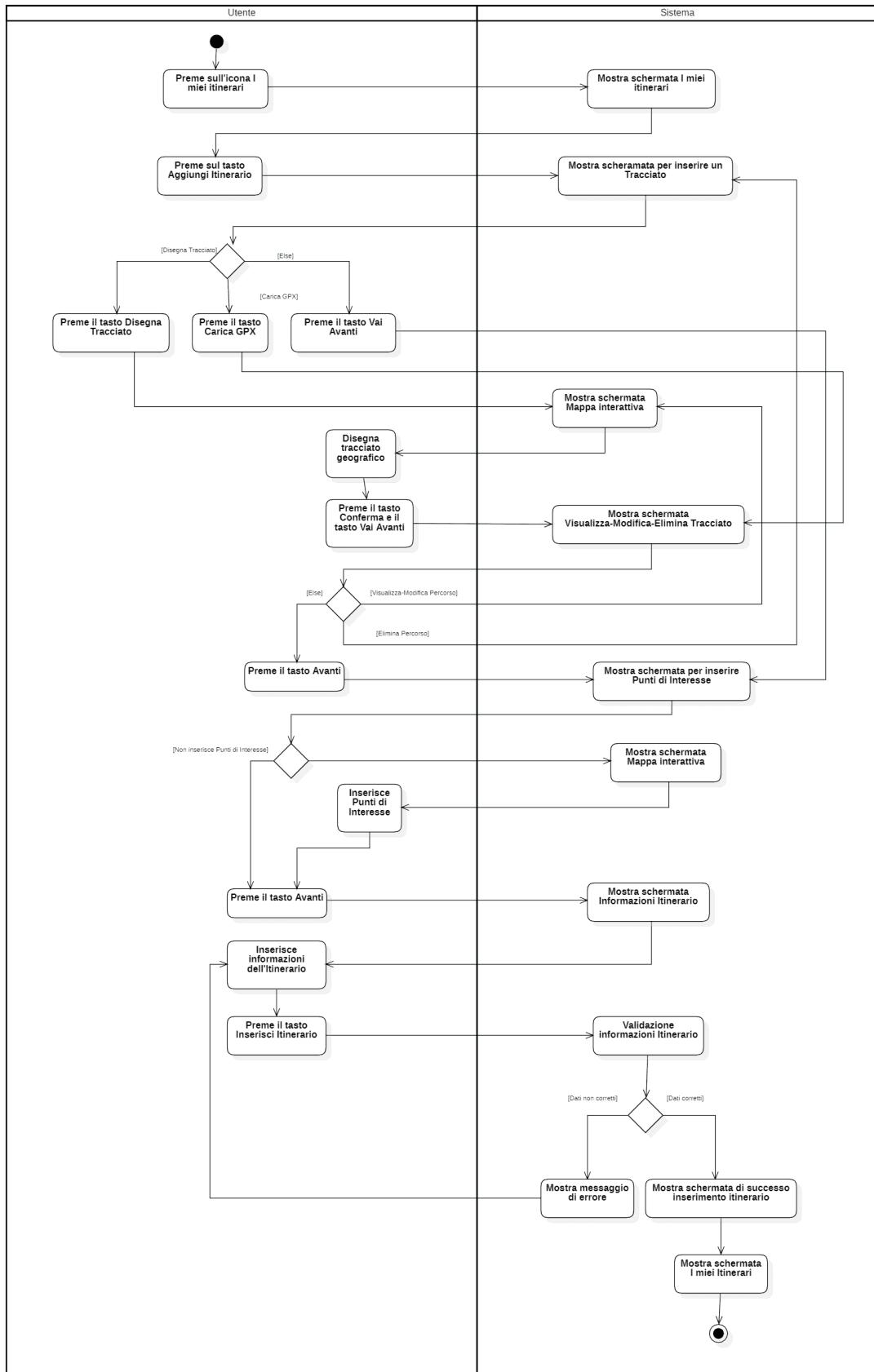
### 4.3.1 Autenticazione e Registrazione



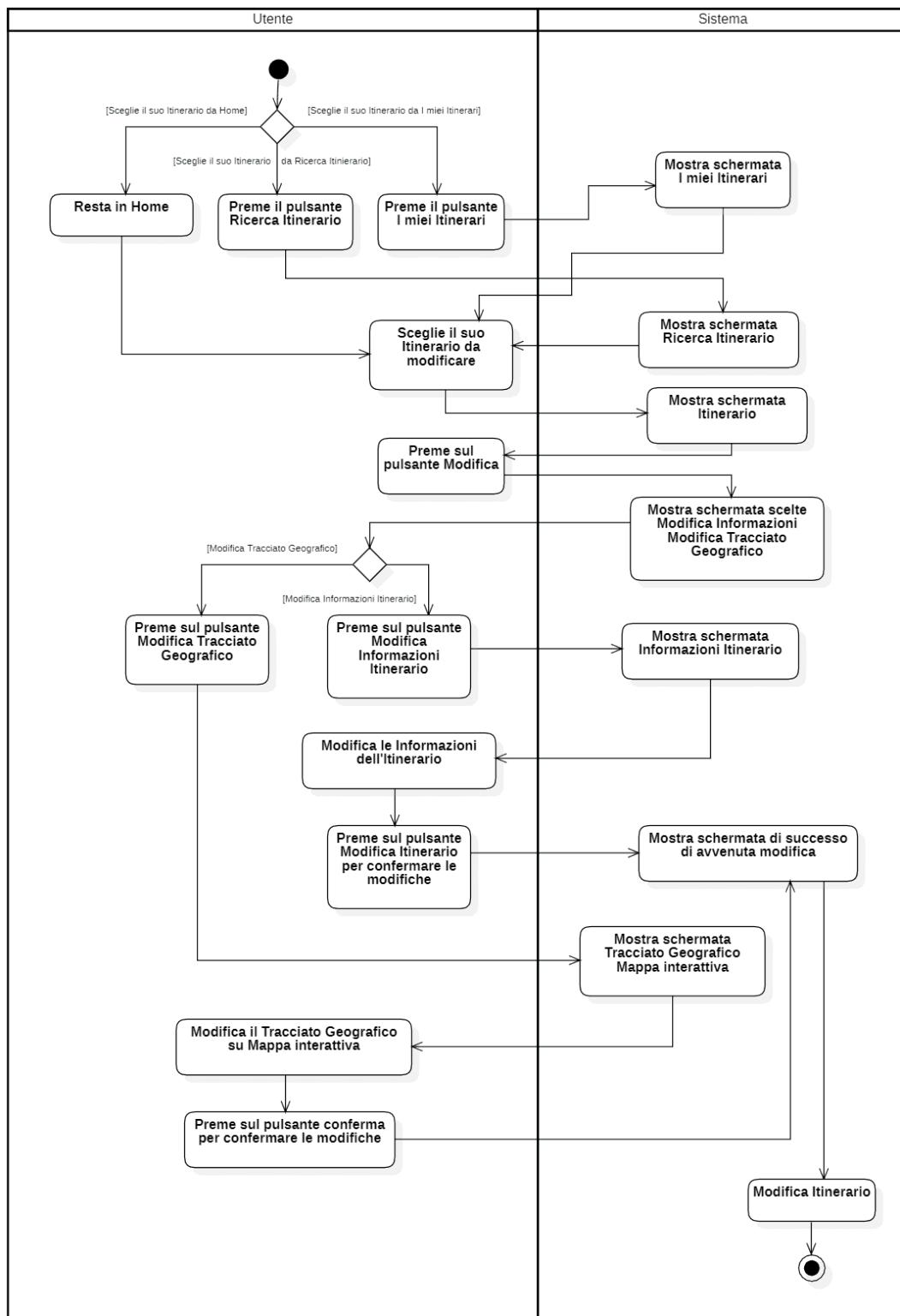
#### 4.3.2 Modifica Profilo



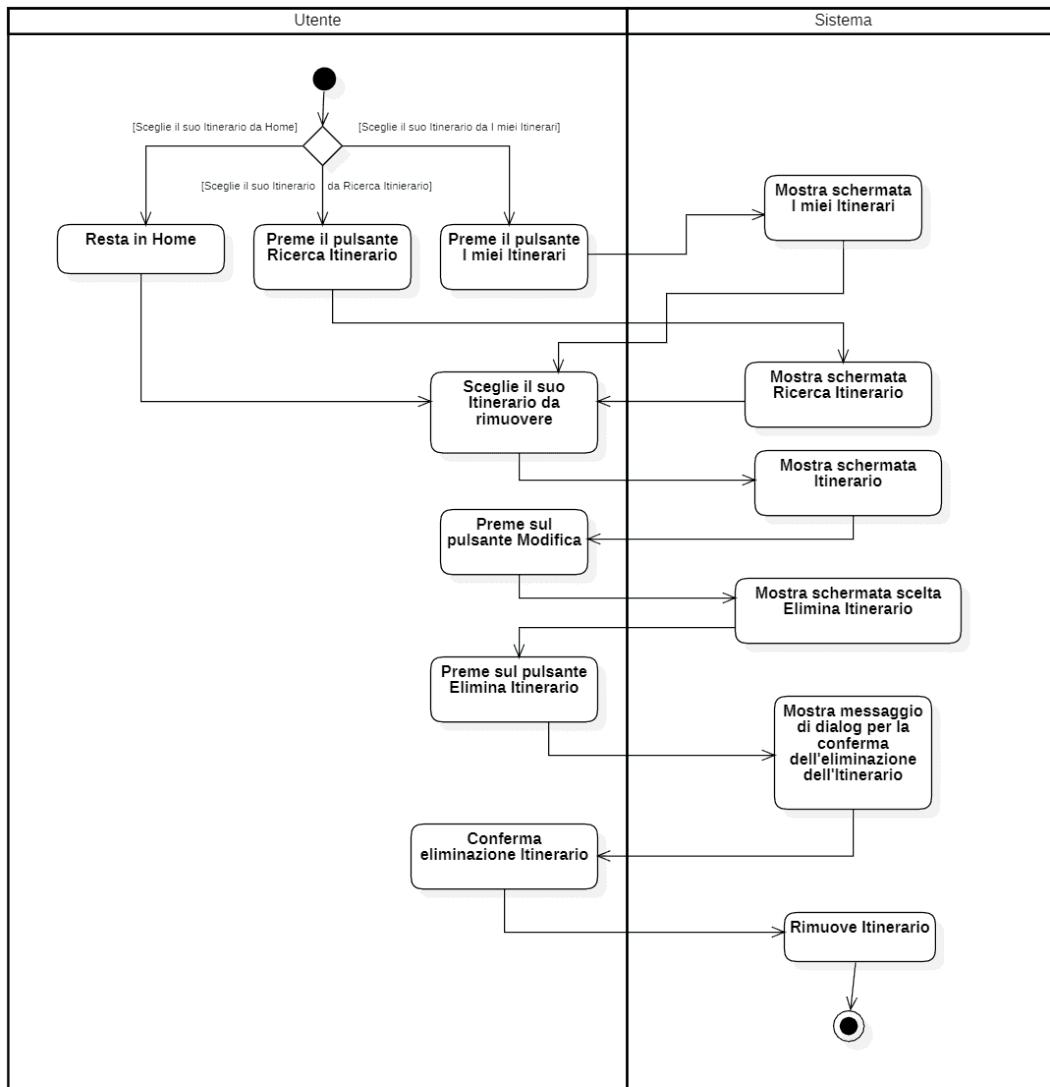
### 4.3.3 Inserimento Itinerario e Punti di Interesse



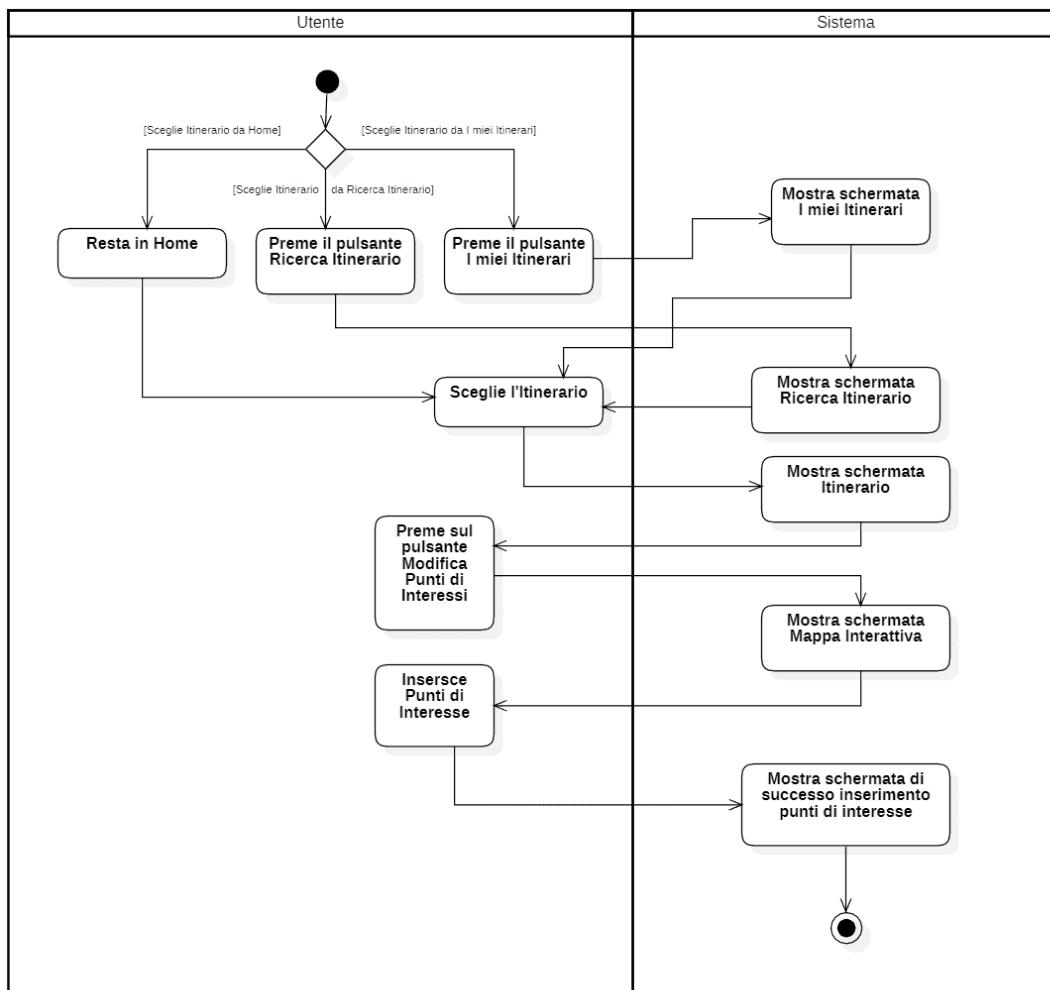
#### 4.3.4 Modifica Itinerario



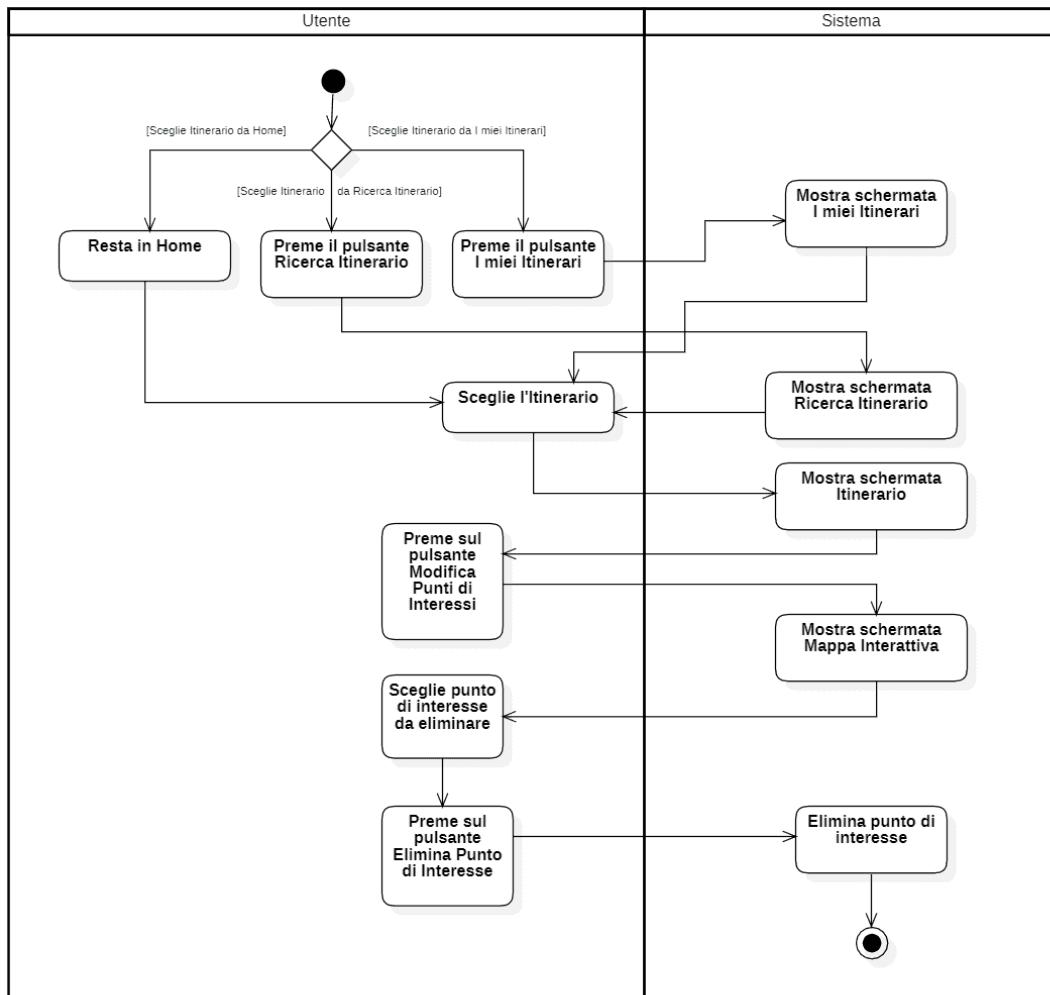
#### 4.3.5 Rimozione Itinerario



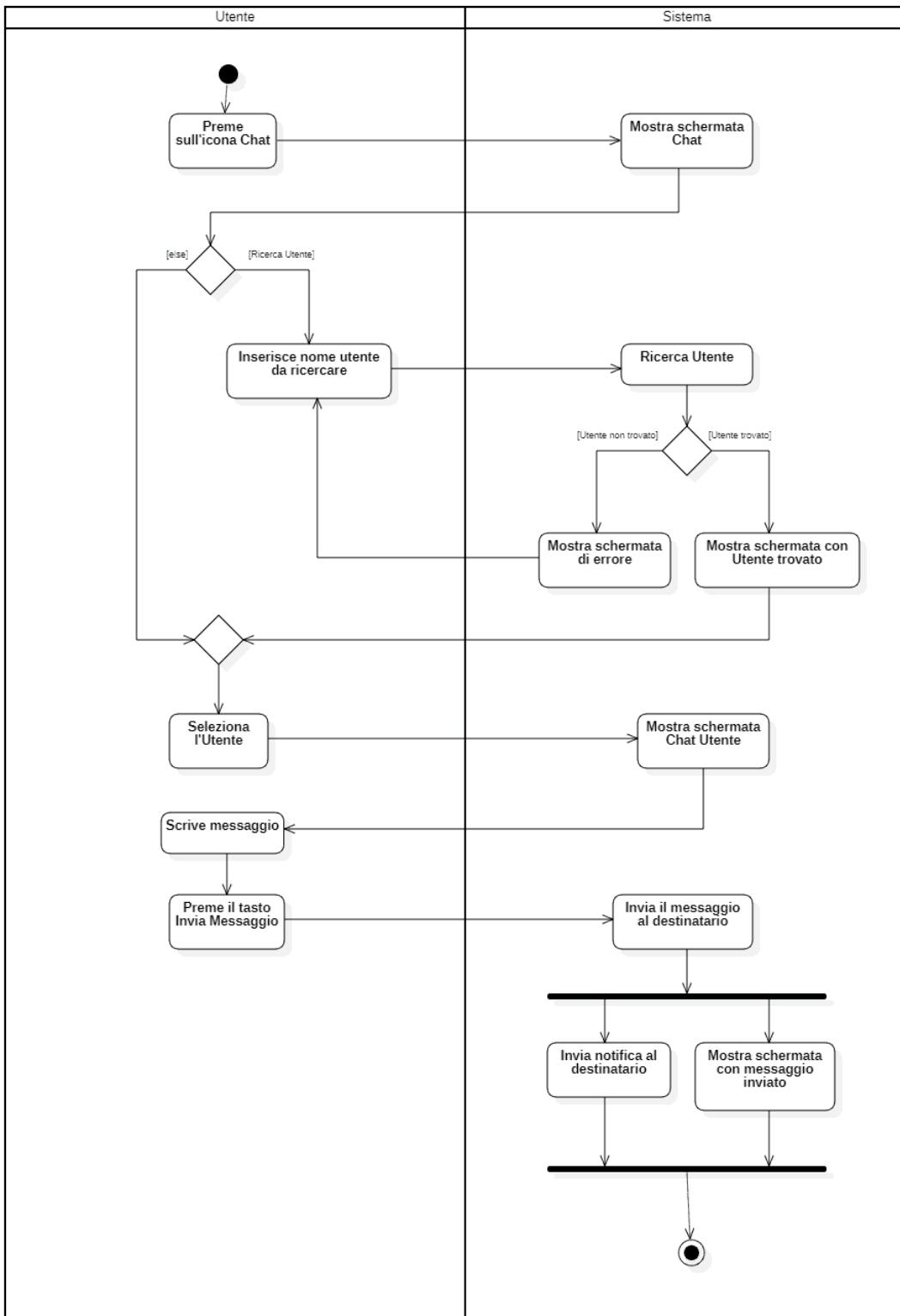
#### 4.3.6 Inserimento Punti di Interesse



### 4.3.7 Rimozone Punti Di Interesse



#### 4.3.8 Chat



## Parte II

# Documento di design del sistema

# 5. Analisi dell'architettura

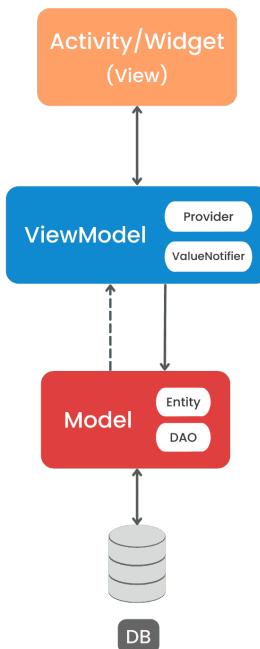
## 5.1 Architettura Client

Il client mobile si basa su un applicativo per dispositivi con sistema operativo Android e iOS sviluppato con il framework Flutter su base Dart.

L'architettura client è stata progettata seguendo le regole del pattern Model View View Model  
La struttura MVVM è composta:

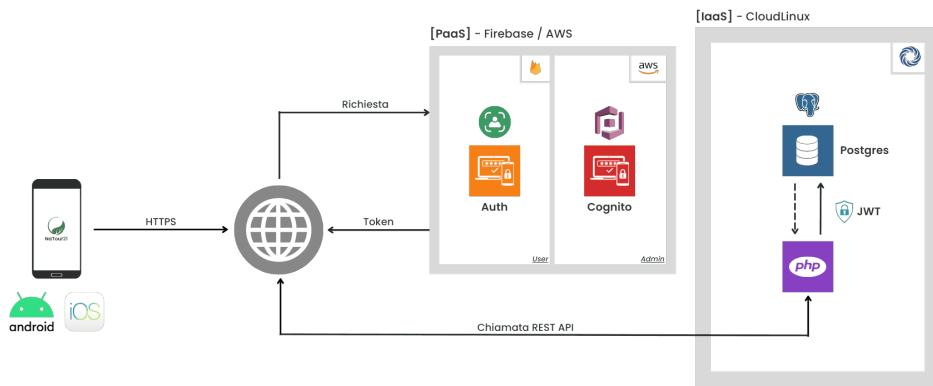
- Model, rappresenta il punto di accesso ai dati, Trattasi di una o più classi che leggono dati da un Database.
- View, rappresenta la vista dell'applicazione, ossia l'interfaccia grafica che mostrerà i dati.
- ViewModel, rappresenta il punto d'incontro tra la View e il Model, i dati ricevuti da quest'ultimo sono elaborati per essere presentati e passati alla view.

La rappresentazione del componente ViewModel è avvenuta mediante due implementazioni, Provider e ValueNotifier. Le due implementazioni si basano sul pattern Observer, entrambi inviano una notifica all'ascoltatore (View) nel momento in cui c'è un cambiamento di dati proveniente dal Model. La differenza principale tra Provider e ValueNotifier è caratterizzata da come entrambi le implementazioni inviano la notifica all'osservatore. ValueNotifier a differenza di Provider invia sempre una notifica all'osservatore quando c'è un cambiamento di dati, mentre in Provider l'invio della notifica non è un'operazione automatica ma decisionale.



## 5.2 Architettura Server

L'architettura Server progettata per l'applicativo prevede l'uso di un Cloud Server su base CloudLinux come SO. Per lo storage dei dati è stato utilizzato un server PostgreSQL integrato in CloudLinux. La comunicazione client-server avviene tramite Server NGINX (HTTPS) la cui accessibilità è data dalle REST API messe in sicurezza dallo standard JWT (JSON Web Token), l'interpretazione della REST API è a carico di PHP che comunica la richiesta al Database di Postgres la cui risposta sarà restituita in formato JSON.



### 5.2.1 Servizi cloud

- **Firebase Auth** : è uno strumento messo a disposizione dalla piattaforma Firebase. Ci consente di creare accessi e registrazioni degli utenti in tutta sicurezza. Con possibilità di sfruttura il login tramite servizi terzi come Facebook, google ed etc.
- **Firebase Firestore** : è un servizio messo a disposizione dalla piattaforma Firebase. Ci Consente di creare una base di dati non relazionale che sfrutti l'aggiornamento in tempo reale dei dati.
- **Firebase Cloud Storage** : è un servizio messo a disposizione dalla piattaforma di Firebase. Ci consente di salvare ogni tipo di file e creare dei criteri di accesso ad essi.
- **Firebase Analytics** : è la piattaforma di analisi sviluppata da Google che ci consente di individuare il comportamento dell'utente nel nostro applicativo tramite registrazione degli eventi.
- **One Signal** : è un servizio multi piattaforma che ci da la possibilità di poter collegare tramite ad un ecosistema interno, tutti gli utenti che si registrano all'app dando la possibilità di inviare notifiche fluttuanti e notifiche in-app. E' compatibile con tutti i moderni sistemi operativi mobile e desktop in circolazione e con tutti i browser più famosi.
- **Cognito** : è uno strumento messo a disposizione da Amazon Web Service (AWS). Ci consente di creare accessi e registrazioni degli utenti in tutta sicurezza. Con possibilità di sfruttura il login tramite servizi terzi come Facebook, google ed etc.

## 5.2.2 API

Le seguenti API rispettano l'architettura *Representational State Transfer* (REST). Il funzionamento prevede l'invocazione della stessa API mediante protocollo HTTP/HTTPS. A operazione terminata viene restituito un codice di errore 200 se l'azione prevista dall'API è andata a buon fine oppure 404 se tale operazione non è stata soddisfatta. Per le richieste di tipo GET i dati esistenti vengono restituiti in formato JSON.

Le API sono state sviluppate in PHP e sono protette mediante il JWT onde evitare di avere richieste esterne dal client.

### 5.2.2.1 Utenti

<b>GET</b>	/api/user/{id_key}	▼
<b>POST</b>	/api/user/	▼

### 5.2.2.2 Itinerari

<b>GET</b>	/api/itinerary/{id_key}	▼
<b>PUT</b>	/api/itinerary/{id_itinerary}	▼
<b>DELETE</b>	/api/itinerary/{id_itinerary}	▼
<b>GET</b>	/api/itinerary	▼
<b>POST</b>	/api/itinerary	▼

### 5.2.2.3 Token di Notifica

<b>DELETE</b>	/api/token_notify/{token_id}	▼
<b>POST</b>	/api/token_notify	▼

### 5.2.2.4 Statistiche di Ricerca

<b>GET</b>	/api/search_statistics/{keyword}	▼
<b>POST</b>	/api/search_statistics	▼

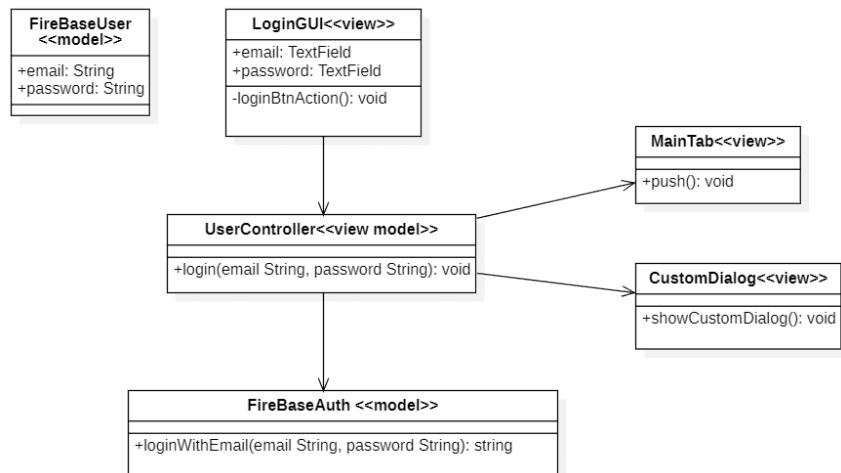
# 6. Modelli di Design

Di seguito vengono presentati i modelli di design dell'applicazione.

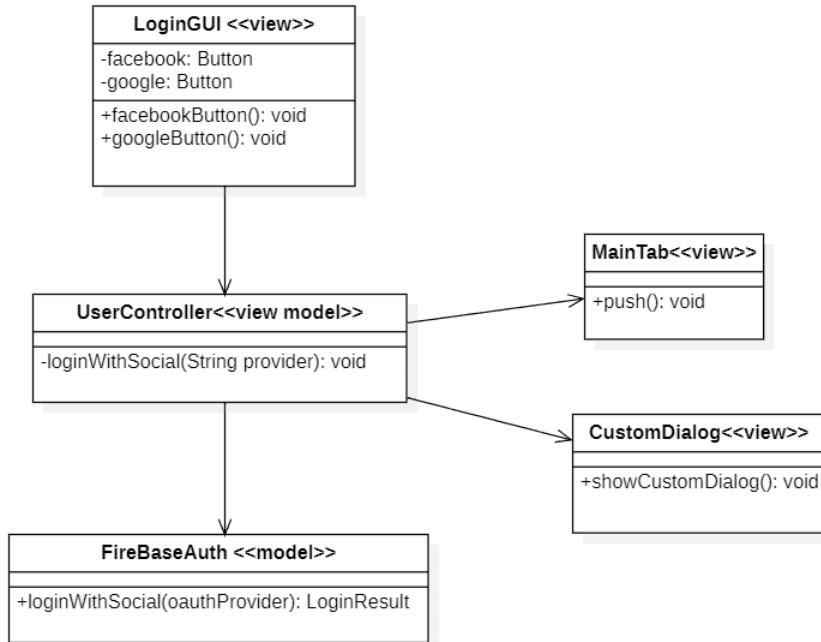
I modelli di design saranno divisi in due sottosezioni: *Diagrammi di classi di design* e *Diagrammi di sequenza di design*.

## 6.1 Diagrammi di classi di design

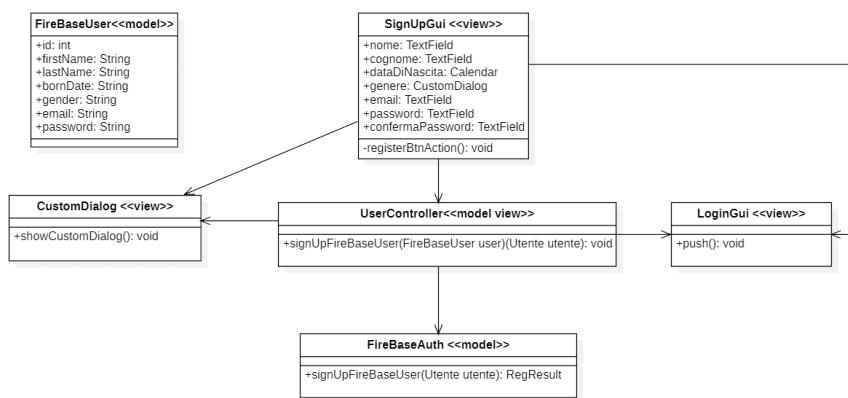
### 6.1.1 Si autentica



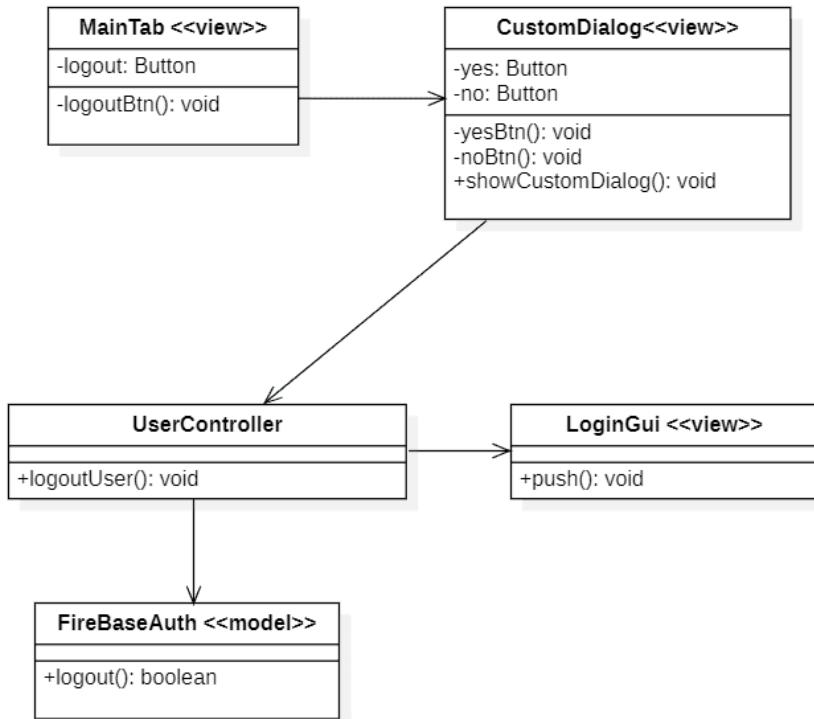
## 6.1.2 Si autentica con social



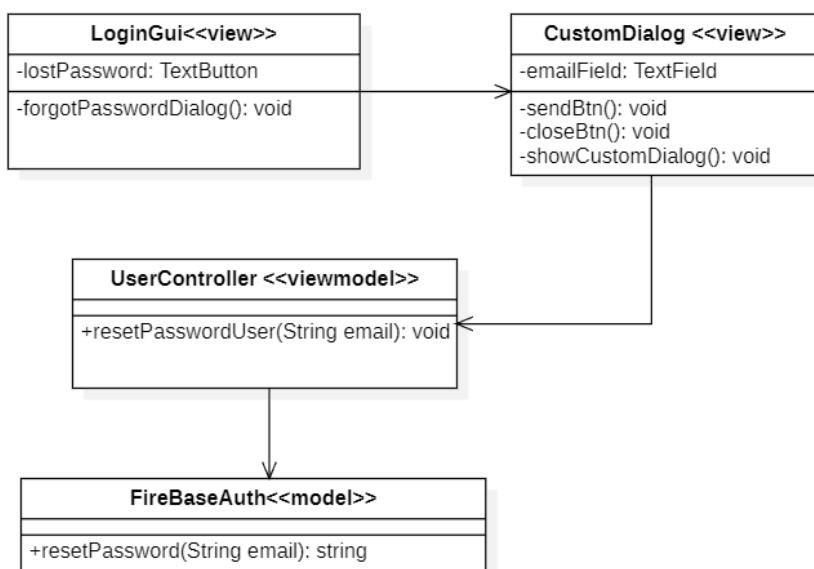
## 6.1.3 Si registra



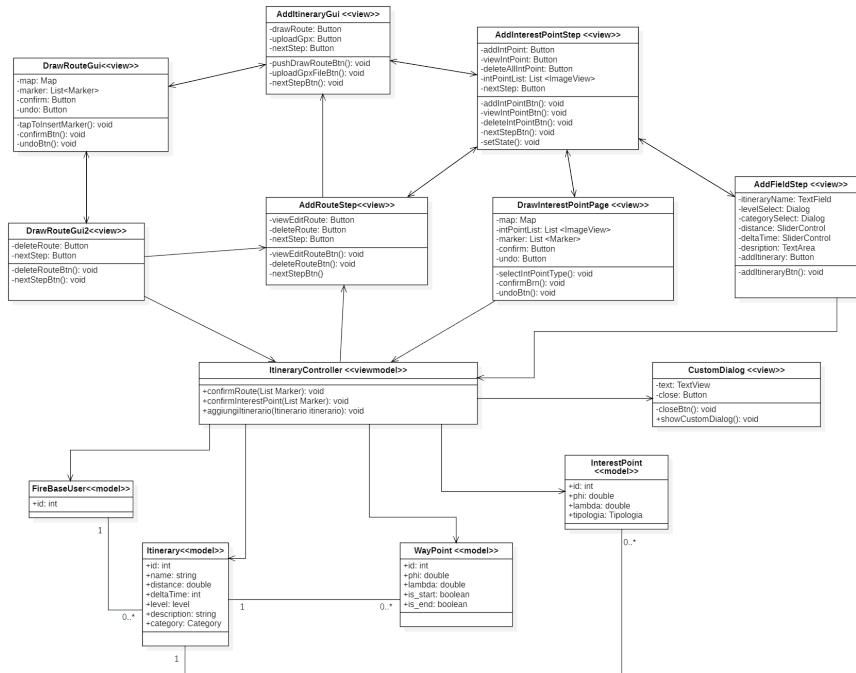
### 6.1.4 Effettua disconnessione



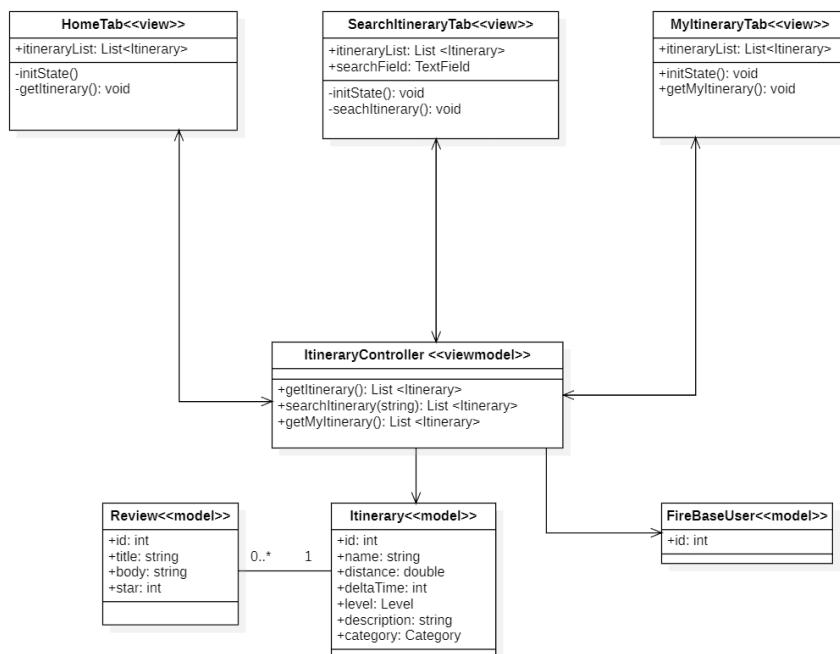
### 6.1.5 Recupera Password



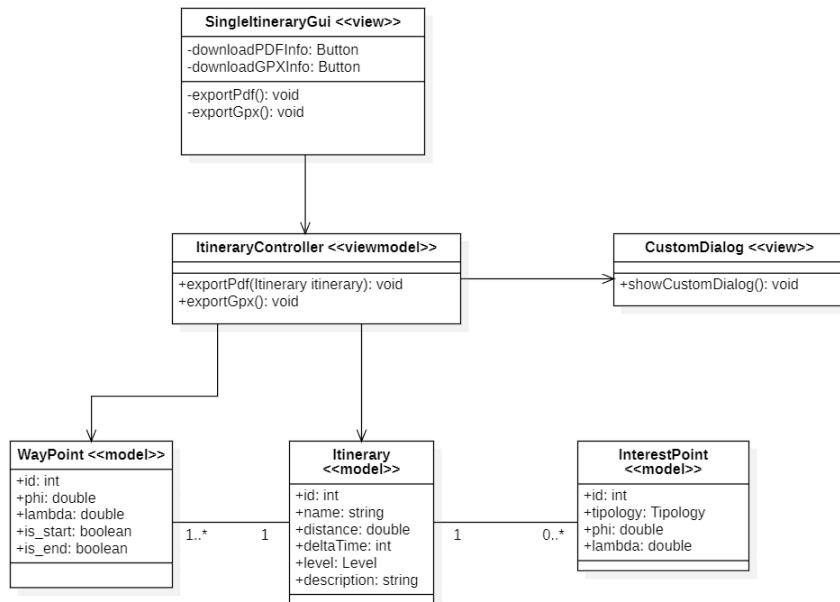
## 6.1.6 Inserisce nuovi itinerari



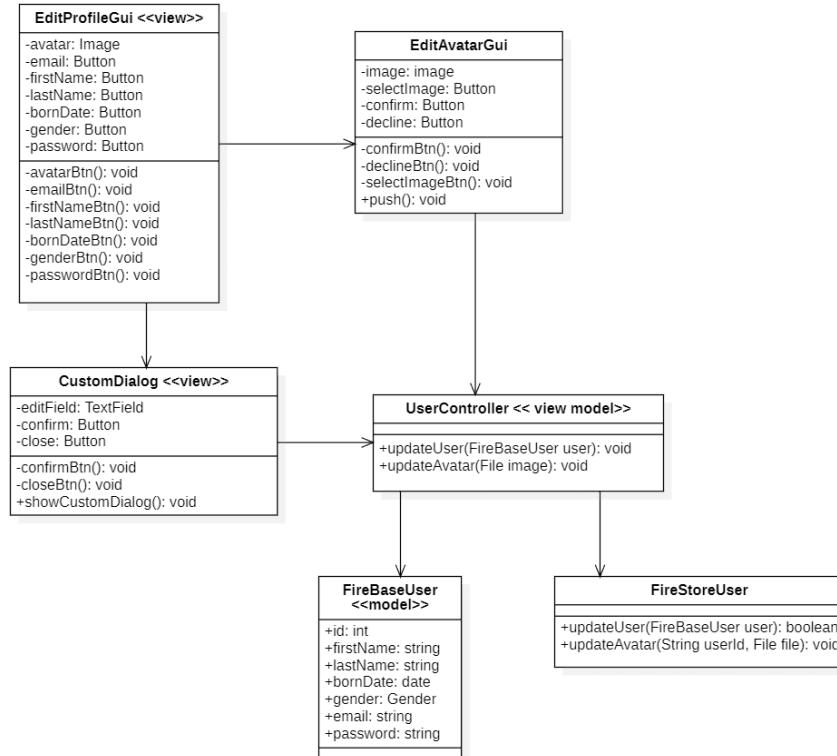
## 6.1.7 Visualizza e ricerca Itinerari



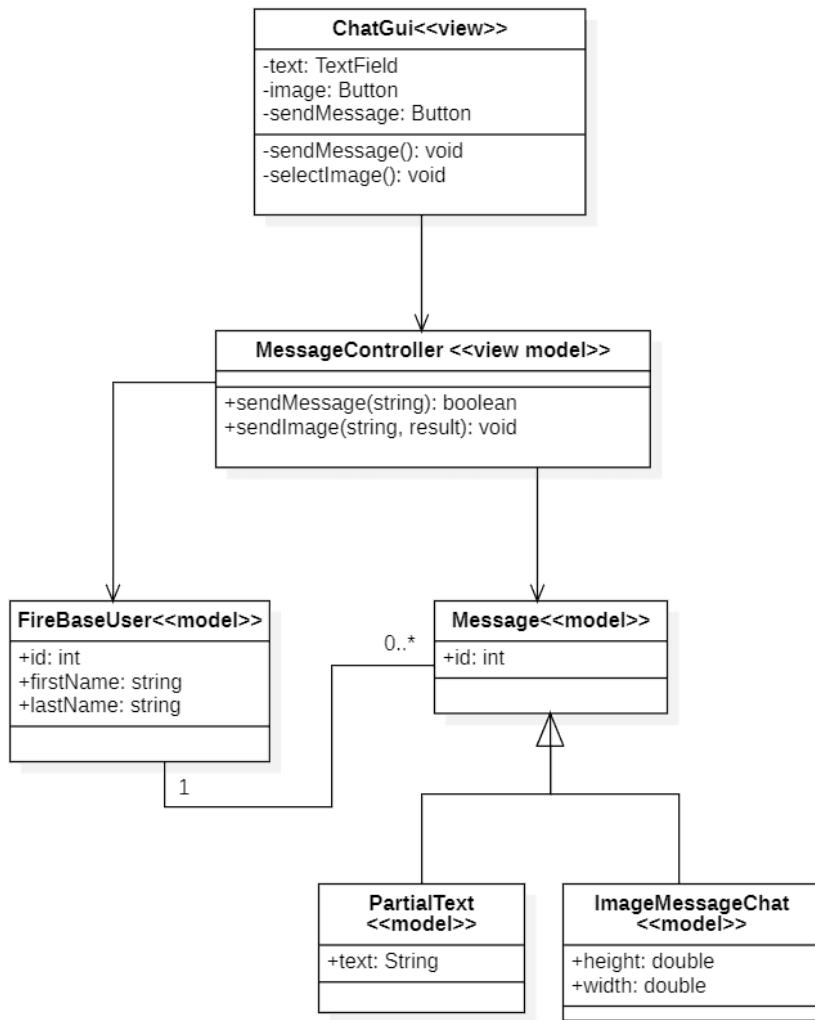
## 6.1.8 Scarica informazioni sentiero



## 6.1.9 Visualizza e modifica profilo

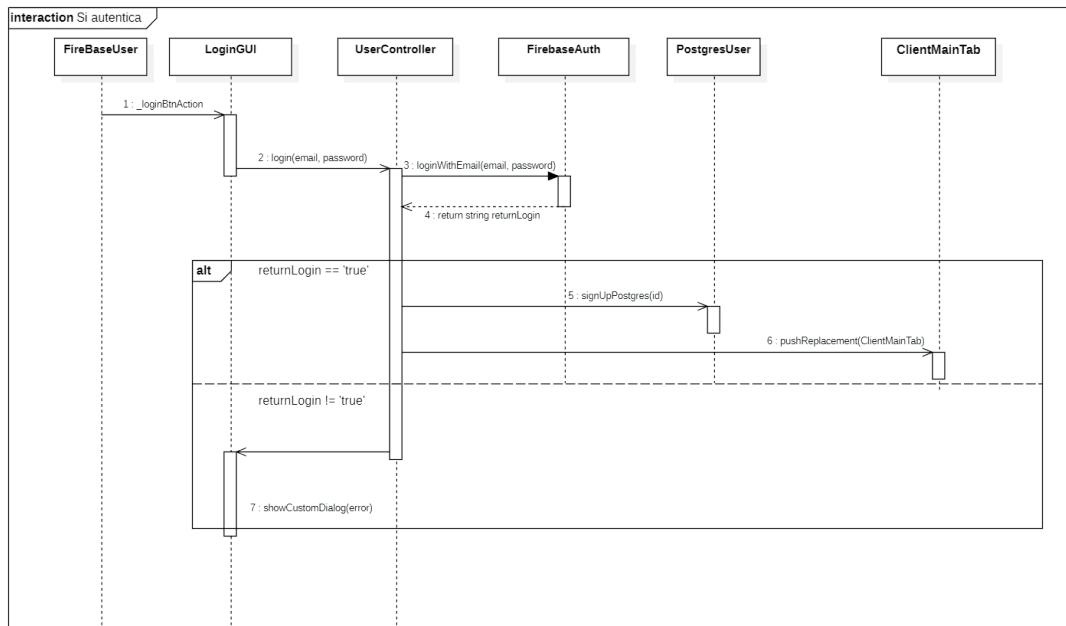


### 6.1.10 Invia messaggi privati

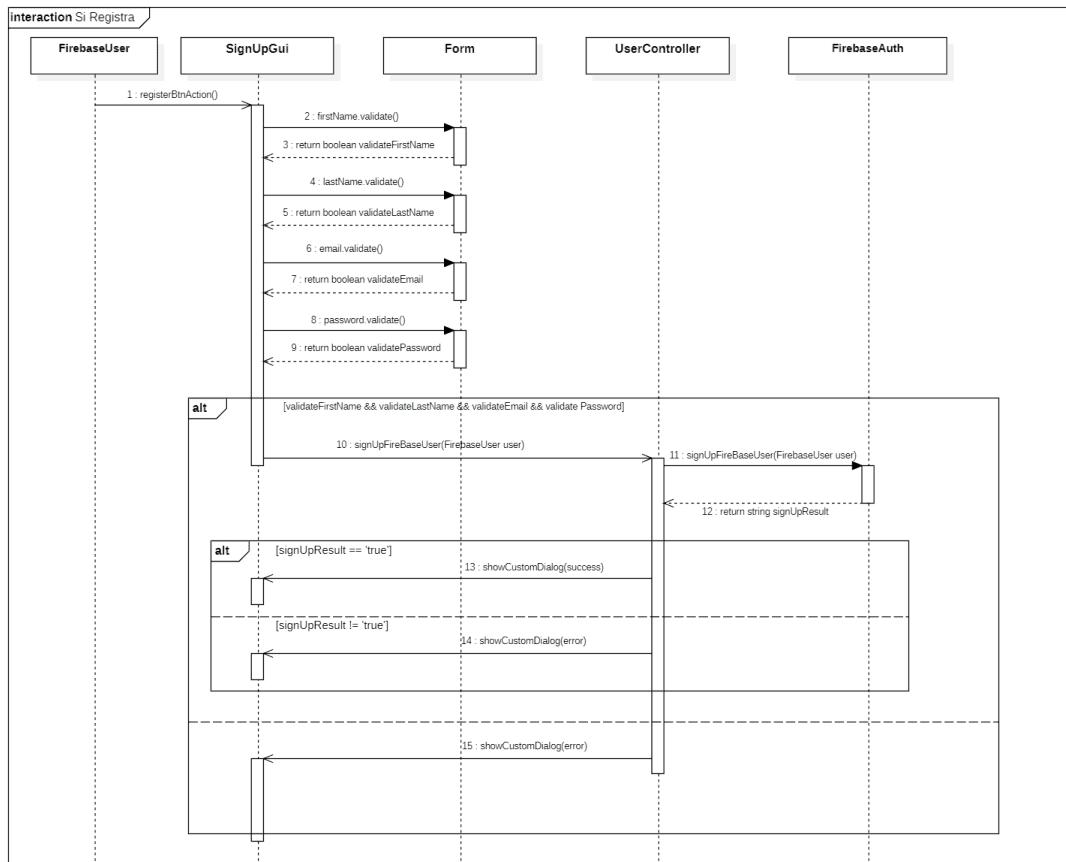


## 6.2 Diagrammi di sequenza di design

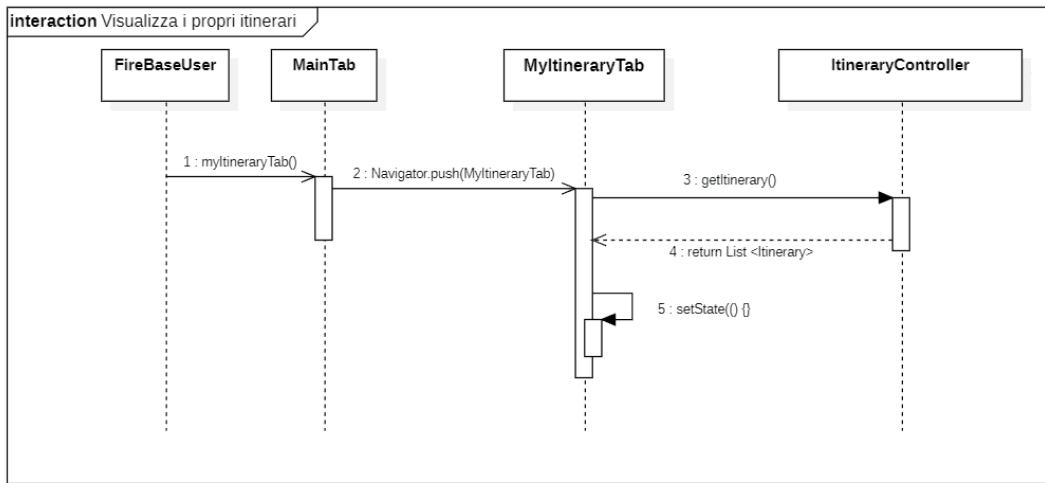
### 6.2.1 Si Autentica



## 6.2.2 Si Registra



### 6.2.3 Visualizza I Proprio Itinerari



# Parte III

## Testing del sistema

# 7. Unit testing

## 7.1 Strategia Black Box

Per il testing dei metodi black box è stata adottata la strategia SECT. Sono state individuate alcune classi di equivalenza in base ai parametri di ingresso dei metodi.

### 7.1.1 Testing metodo *convertGpxIntoClass*

Il metodo `convertGpxIntoClass` è un metodo che converte un file di tipo GPX in un percorso geografico rappresentato sull'applicazione.

Parametro in ingresso: `file` (file di tipo gpx importato dall'utente).

Di seguito il codice del metodo da testare:

---

```
Future<bool> convertGpxIntoClass(File? file) async {
    Gpx gpx;

    if (file == null) {
        return false;
    }
    final fileString = await file.readAsString();

    try {
        gpx = GpxReader().fromString(fileString);
    } catch (_) {
        return false;
    }

    int rCount = 0;
    int tCount = 0;
    int wCount = 0;

    if (gpx.rtes.isNotEmpty) {
        for (int i = 0; i < gpx.rtes.length; i++) {
            for (int j = 0; j < gpx.rtes[i].rtepts.length; j++) {
                rCount = 1;
                Global().itinerary.route!.add(
                    WayPointRoot(
                        phi: gpx.rtes[i].rtepts[j].lat,
                        lambda: gpx.rtes[i].rtepts[j].lon,
                        isStart: j == 0 ? true : false,
                        isEnd: j == gpx.rtes[i].rtepts.length - 1 ? true : false),
                );
            }
        }
    }
}
```

```

        else if (gpx.trks.isNotEmpty) {
            for (int i = 0; i < gpx.trks.length; i++) {
                for (int j = 0; j < gpx.trks[i].trksegs.length; j++) {
                    for (int z = 0; z < gpx.trks[i].trksegs[j].trkpts.length; z++) {
                        tCount = 1;
                        Global().itinerary.route!.add(WayPointRoot(
                            phi: gpx.trks[i].trksegs[j].trkpts[z].lat,
                            lambda: gpx.trks[i].trksegs[j].trkpts[z].lon,
                            isStart: z == 0 ? true : false,
                            isEnd: z == gpx.trks[i].trksegs[j].trkpts.length - 1
                                ? true
                                : false));
                    }
                }
            }
        } else {
            for (int i = 0; i < gpx.wpts.length; i++) {
                wCount = 1;
                Global().itinerary.route!.add(WayPointRoot(
                    phi: gpx.wpts[i].lat,
                    lambda: gpx.wpts[i].lon,
                    isStart: i == 0 ? true : false,
                    isEnd: i == gpx.wpts.length - 1 ? true : false));
            }
        }
        if (rCount == 0 && tCount == 0 && wCount == 0) {
            return false;
        }
        return true;
    }

```

---

Il metodo richiamato in modalità sincrona, ritorna un valore booleano a seconda dei casi:

- **True** se l'importazione e la conversione dei file gpx va a buon fine.
- **False** se non viene importato nessun file.  
Se il file è formattato in un formato diverso da gpx oppure all'interno è vuoto.  
Se il file non contiene un numero necessario di waypoint da poter generare un percorso

Sono state individuate 4 classi di equivalenza:

- CE1: file associato a un file non caricato
- CE2: file associato a un file vuoto o non formattato in formato gpx
- CE3: file associato a un file che non contiene almeno 2 waypoint validi per generare un percorso
- CE4: file associato a un file valido correttamente importato

Di seguito l'implementazione dei casi di test che coprono tutte le classi di equivalenza in base al parametro di ingresso:

---

```
Future<void> importGpxFileCE1() async {
    var result = await ItineraryController().convertGpxIntoClass(null);

    assert(result == false);
}

Future<void> importGpxFileCE2() async {
    final data = await rootBundle.load('assets/test/ce2.gpx');

    final directory = (await getTemporaryDirectory()).path;
    var temp =
        await File('$directory/ce2.gpx').writeAsBytes(data.buffer.asUint8List(
            data.offsetInBytes,
            data.lengthInBytes,
        ));
    bool result = await ItineraryController().convertGpxIntoClass(temp);

    assert(result == false);
}

Future<void> importGpxFileCE3() async {
    final data = await rootBundle.load('assets/test/ce3.gpx');

    final directory = (await getTemporaryDirectory()).path;
    var temp =
        await File('$directory/ce3.gpx').writeAsBytes(data.buffer.asUint8List(
            data.offsetInBytes,
            data.lengthInBytes,
        ));
    bool result = await ItineraryController().convertGpxIntoClass(temp);

    assert(result == false);
}

Future<void> importGpxFileCE4() async {
    final data = await rootBundle.load('assets/test/ce4.gpx');

    final directory = (await getTemporaryDirectory()).path;
    var temp =
        await File('$directory/ce4.gpx').writeAsBytes(data.buffer.asUint8List(
            data.offsetInBytes,
            data.lengthInBytes,
        ));
    bool result = await ItineraryController().convertGpxIntoClass(temp);

    assert(result == true);
}
```

---

## 7.1.2 Testing metodo *updateAvatar*

Il metodo `updateAvatar` è un metodo utili ad aggiornare il proprio avatar.

Parametri in ingresso: **userID** (id dell'utente a cui si vuole modificare l'avatar) e **avatarFile** (il nuovo avatar in formato immagine).

Di seguito il codice del metodo da testare:

---

```
Future<bool> updateAvatar(String? userID, File? avatarFile) async {
    if (userID == null) {
        return false;
    }

    if (avatarFile == null) {
        await _mock.collection('users').doc(userID).update({
            'first_login': false,
        });
        return false;
    }

    String path = 'avatar/' + userID + getRandomString(20) + '.png';

    try {
        var collectionRef;
        var doc;

        if (_mock.runtimeType.toString() == 'FakeFirebaseFirestore') {
            collectionRef = await _mock.collection('users').get();
            doc = collectionRef.docs.first.get('username') == userID ? userID : '';
            if (!doc.isNotEmpty) {
                return false;
            } else {
                return true;
            }
        } else {
            collectionRef = _mock.collection('users');
            doc = await collectionRef.doc(userID).get();

            if (!doc.exists) {
                return false;
            } else {
                await FirebaseStorage.instance.ref(path).putFile(avatarFile);
                await _mock.collection('users').doc(userID).update({
                    'first_login': false,
                    'avatar': (await (FirebaseStorage.instance.ref().child(path)).getDownloadURL()).toString(),
                });
            }
        }

        await _mock.collection('users').doc(FirebaseAuth.instance.currentUser!.uid)
            .get()
            .then((value) {
                Global().myUser.value = FireBaseUser.fromJson(value.data());
                Global().myUser.value.setId = FirebaseAuth.instance.currentUser!.uid;
                Global().myUser.value.setEmail =
                    FirebaseAuth.instance.currentUser!.email!;
            });
        return true;
    }
} catch (e) {
    return false;
}
```

---

Il metodo richiamato in modalità sincrona, ritorna un valore booleano a seconda dei casi:

- **True** l'userID esiste e l'avatarFile è stato caricato in modo corretto.
- **False** in tutti i casi contrari del valore true.

Sono state individuate 3 classi di equivalenza per il parametro userID e 2 per il parametro avatarFile:

- CE1: userID il parametro inserito è null
- CE2: userID il parametro inserito non esiste nella base di dati
- CE3: userID il parametro inserito esiste nella base di dati
- CE1: avatarFile il parametro inserito è null
- CE2: avatarFile il parametro inserito è corretto (diverso da null)

Di seguito l'implementazione dei casi di test che coprono tutte le classi di equivalenza in base ai parametri di ingresso:

---

```
Future<void> updateAvatarCE1(FakeFirebaseFirestore mock) async {
    var result = await FireStoreUser(mock: mock).updateAvatar(null, null);
    assert(result == false);
}

Future<void> updateAvatarCE2(FakeFirebaseFirestore mock) async {
    final data = await rootBundle.load('assets/test/avatar.png');

    final directory = (await getTemporaryDirectory()).path;
    var temp = await File('$directory/avatar.png')
        .writeAsBytes(data.buffer.asUint8List(
            data.offsetInBytes,
            data.lengthInBytes,
        ));

    var result = await FireStoreUser(mock: mock).updateAvatar(null, temp);
    assert(result == false);
}

Future<void> updateAvatarCE3(FakeFirebaseFirestore mock) async {
    var result = await FireStoreUser(mock: mock).updateAvatar('fabio', null);
    assert(result == false);
}

Future<void> updateAvatarCE4(FakeFirebaseFirestore mock) async {
    final data = await rootBundle.load('assets/test/avatar.png');

    final directory = (await getTemporaryDirectory()).path;
    var temp = await File('$directory/avatar.png')
        .writeAsBytes(data.buffer.asUint8List(
            data.offsetInBytes, data.lengthInBytes,
        ));

    var result = await FireStoreUser(mock: mock).updateAvatar('fabio', temp);
    assert(result == false);
}
```

```

Future<void> updateAvatarCE5(FakeFirebaseFirestore mock) async {
    var result =
        await FireStoreUser(mock: mock).updateAvatar('andrea', null);
    assert(result == false);
}

Future<void> updateAvatarCE6(FakeFirebaseFirestore mock) async {
    final data = await rootBundle.load('assets/test/avatar.png');

    final directory = (await getTemporaryDirectory()).path;
    var temp = await File('$directory/avatar.png')
        .writeAsBytes(data.buffer.asUint8List(
            data.offsetInBytes,
            data.lengthInBytes,
        ));

    var result =
        await FireStoreUser(mock: mock).updateAvatar('andrea', temp);

    assert(result == true);
}

```

---

## 7.2 Strategia White Box

Per il testing del seguente metodo è stata adottata la strategia White Box, rappresentata attraverso dei grafi.

**Nota:** Nel grafo le diramazioni verso destra che implicano la presenza di un if-statement indicano che la condizione è verificata.

### 7.2.1 Testing metodo *sendChatNotify*

Il metodo `sendChatNotify` è un metodo utile ad inviare un notifica.

Parametri in ingresso: **title** (titolo della notifica), **body** (corpo della notifica) e **userToken** l'identificativo dell'utente.

Alla pagina successiva il codice del metodo:

---

```

Future<bool> sendChatNotify(
    String title, String body, String userToken) async {
  if (userToken != await Global().notify.getPlayerId()) {
    return false;
  }

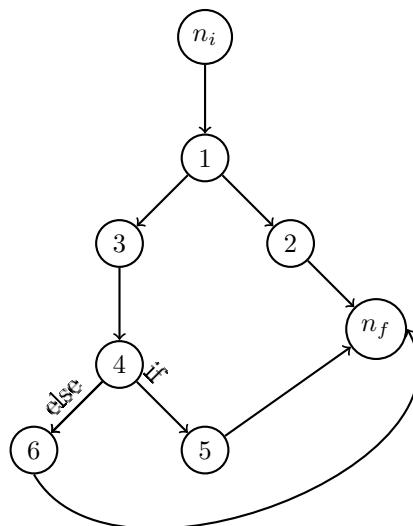
  var doc = (await _mock.collection('user_token').get())
      .docs
      ?.first
      ?.get('token') ==
      userToken
    ? userToken
    : (await _mock.collection('user_token').get()).docs.last.get('token') ==
      userToken
    ? userToken
    : '';

  if (doc.isNotEmpty) {
    return true;
  } else {
    return false;
  }
}

```

---

Di seguito è riportato il grafo del metodo:



I percorsi da coprire sono rispettivamente 3:

- `sendChatNotify_1_2()`.
- `sendChatNotify_1_3_4_5()`.
- `sendChatNotify_1_3_4_6()`.

Di seguito l'implementazione dei casi di test che vanno a coprire tutti e 3 i percorsi:

---

```
Future<void> sendChatNotify_1_2(FakeFirebaseFirestore mock) async {
    var result = await OneSignal().sendChatNotify('title', 'body', 'example');
    assert(result == false);
}

Future<void> sendChatNotify_1_3_4_5(FakeFirebaseFirestore mock) async {
    await mock.collection('user_token').add({'token': '43tqgwrfaV3'});
    var result =
        await OneSignal().sendChatNotify('title', 'body', '9rrWD2K5Kr');
    assert(result == false);
}

Future<void> sendChatNotify_1_3_4_6(FakeFirebaseFirestore mock) async {
    await mock.collection('user_token').add({'token': '9rrWD2K5Kr'});
    var result =
        await OneSignal().sendChatNotify('title', 'body', '9rrWD2K5Kr');
    assert(result == true);
}
```

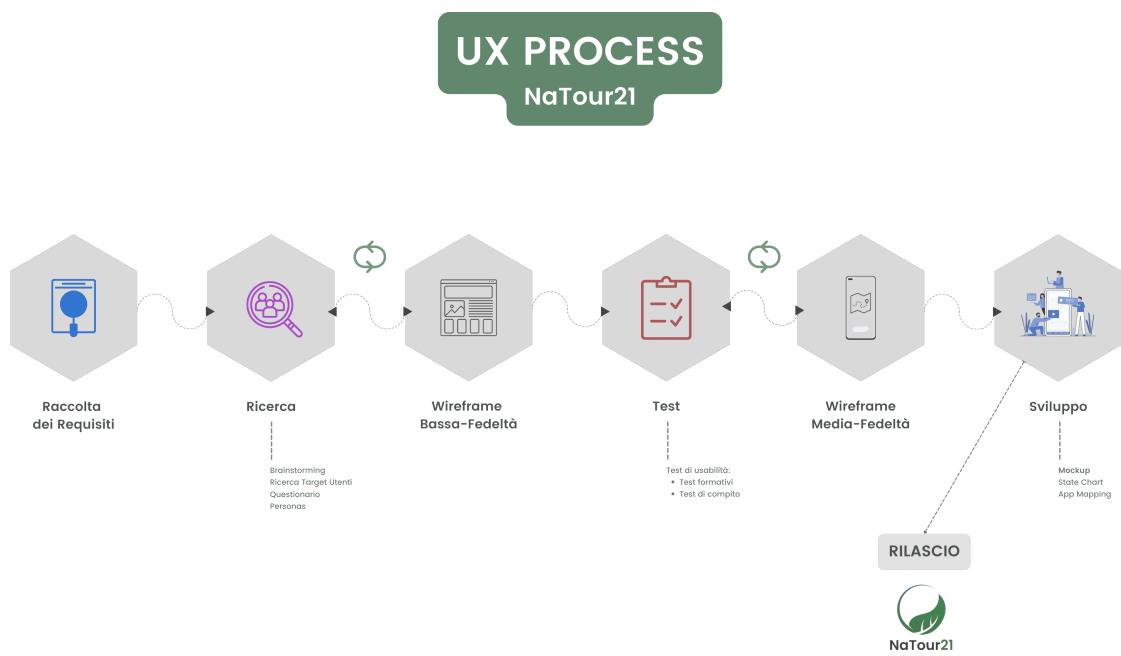
---

## Parte IV

# Relazione di usabilità

# 8. UX Process

Di seguito si riporta il processo di UX Design di NaTour21.



# 9. Target degli utenti

Gli utenti giocano un ruolo chiave nella progettazione di un applicativo software, dato che risultano essere gli utilizzatori principali. Per questo è stato fondamentale fare un'analisi accurata del target degli utenti svolta in fase di analisi dei requisiti, categorizzando quelli che possono e potranno essere ragionevolmente gli attuali e i futuri utilizzatori, determinando le caratteristiche e le esigenze.

L'architettura dell'applicativo software in particolare dell'interfaccia grafica è stata progettata e resa il più possibile compatibile con i profili utenti tracciati nella fase di analisi e, in particolare, le informazioni e le funzioni di maggiore interesse sono state rese accessibili nella maniera più semplice e intuitiva.

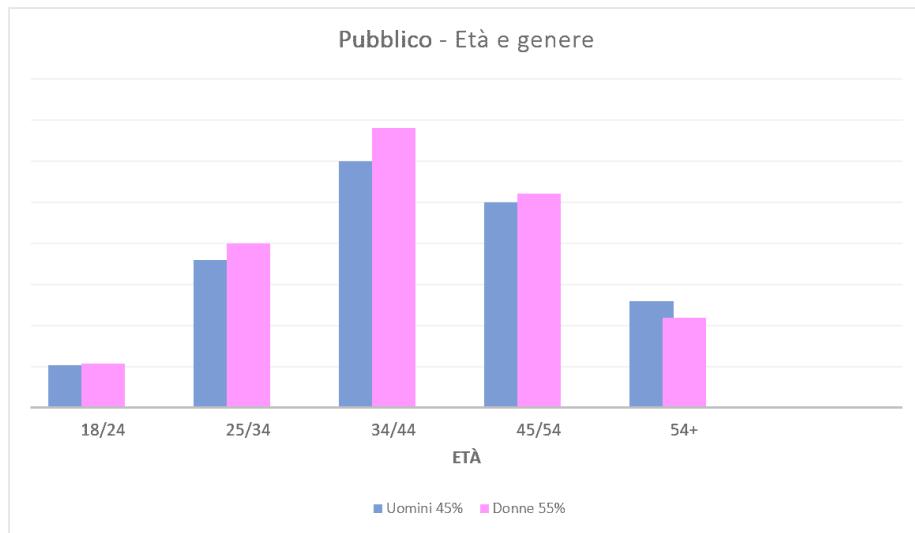
## 9.1 Studio

Lo studio del target degli utenti è stato svolto in due fasi: una prima fase di ricerca più generica che ci ha permesso di identificare un insieme ristretto di utenti più propensi a frequentare costantemente attività di escursionismo. Una seconda fase, conseguenza della prima è stata svolta sulla base di un questionario sottoposto ad alcuni utenti che rientrano in quell'insieme individuato nella prima fase di ricerca.

Attraverso alcuni dati e statistiche reperiti in rete, l'attività di escursionismo è provocata dalla diffusione della cultura del benessere e da un ritrovato interesse per i temi ambientali oltre che dall'esperienza sportiva, sia agonistica che amatoriale, che tende a coinvolgere un target di riferimento sempre più ampio e diversificato.

Grazie a questi dati raccolti abbiamo creato un grafico di riferimento che sta a dimostrare quel pubblico di destinazione più vicino a tale attività, notando l'esigenza di avere a disposizione in questo caso un sistema dotato di buona apprendibilità e buona memorabilità.

Il grafico seguente rappresenta in percentuale, l'età e il genere in cui si fa maggiore attività di escursionismo:



Possiamo notare dal precedente grafico, la partecipazione maggiore di un pubblico femminile rispetto a quello maschile, inoltre osserviamo una maggiore partecipazione nel range di età che va dai 34 ai 44 anni. Dai dati reperiti in reteabbiamo riscontrato che negli ultimi anni c'è stato un netto aumento del pubblico femminile propenso a partecipare a questa tipologia di attività, in ogni caso gli over 35 sono il target di riferimento da prendere in considerazione per tale scopo.

## 9.2 Questionario

Grazie ai dati raccolti durante la prima fase di studio che ci ha permesso di filtrare un target di riferimento più dettagliato, nella fase successiva si è deciso di sottoporre un semplice questionario ad un gruppo di persone con lo scopo di ottenere informazioni più dettagliate sui loro interessi verso l'attività di escursionismo.

Segue il questionario con domande e risposte:

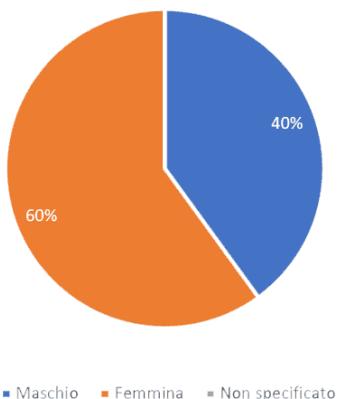
Numero utenti scelti: 5

Domande:

- 1) Il tuo sesso.
- 2) Seleziona la tua fascia d'età.
- 3) Cosa fai nella vita?
- 4) Viaggi spesso?
- 5) Fai uso di social network?
- 6) Sei iscritto a qualche blog di escursionismo?
- 7) Ti piace conoscere nuove persone?
- 8) Ti piacerebbe usare un applicativo dedicato all'attività di escursionismo e poter parlare con altre persone?
- 9) In media quante escursioni fai in un anno?
- 10) Solitamente dove scopri nuovi itinerari?

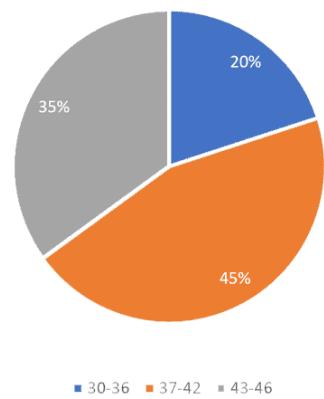
Risposta 1)

### Il tuo sesso

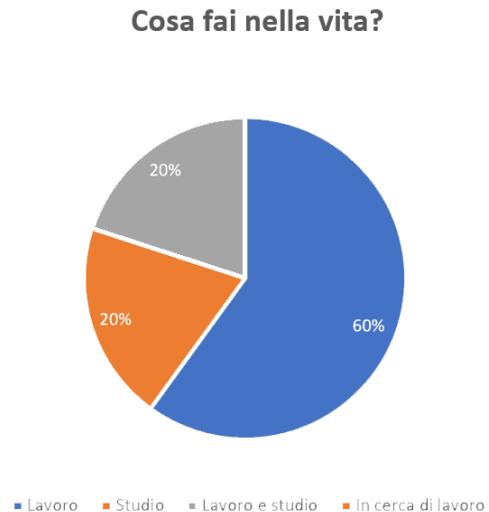


Risposta 2)

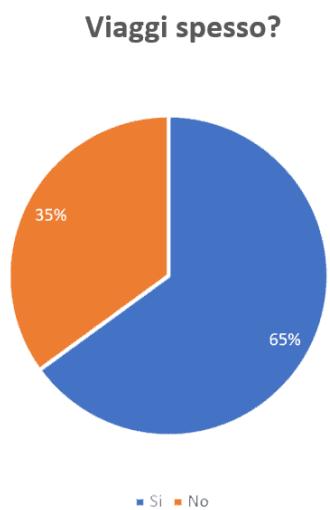
### Seleziona la tua fascia d'età



Risposta 3)

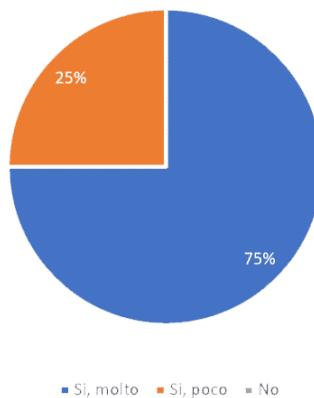


Risposta 4)



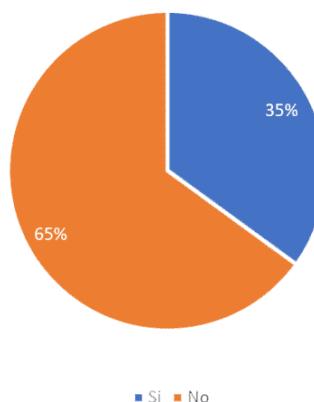
Risposta 5)

**Fai uso di social network?**



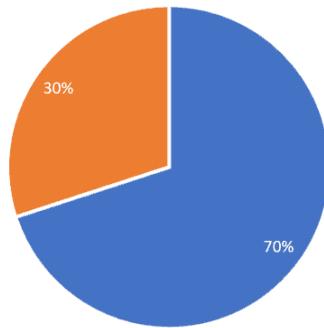
Risposta 6)

**Sei iscritto a qualche blog/gruppi di escursionismo?**



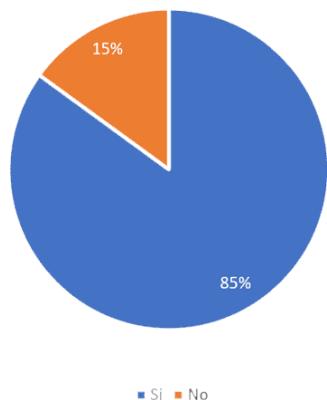
Risposta 7)

**Ti piace conoscere nuove persone?**



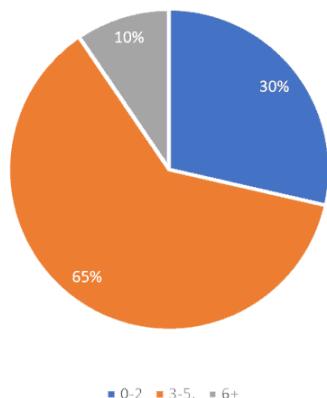
Risposta 8)

**Ti piacerebbe usare un applicativo dedicato all'attività di escursionismo e poter parlare con altre persone?**



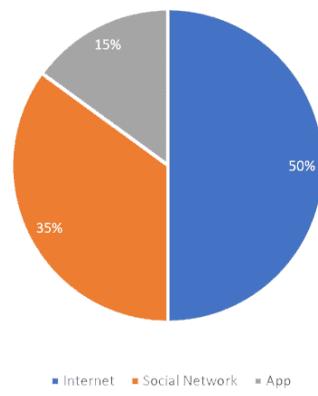
Risposta 9)

**In media quante escursioni fai in un anno?**



Risposta 10)

**Solitamente dove scopri nuovi itinerari?**



Osservazione studio del target di seconda fase:

Dai risultati emersi dal questionario, si conferma che gli utenti coinvolti in questo studio sono persone che praticano l'attività di escursionismo con costanza e grande passione.

Ciò è confermato anche dalla maggioranza di quelle persone che viaggiano spesso, attitudine a conoscere nuove persone, quindi questo comporta ad avere una certa condivisione di quello che si sta facendo verso altre persone.

Gli utenti coinvolti sono persone che fanno un rilevante uso di social network e circa il 65% è iscritto in blog o gruppi di escursionismo. Inoltre è stato confermato che la maggior parte degli utenti, circa il 85% scopre nuovi itinerari attraverso Internet e Social Network e solo il 15% fa uso di App non prettamente dedicate alle escursioni.

Il questionario dimostra un pubblico predisposto a utilizzare una moderna piattaforma social per appassionati di escursionismo, il che rende possibile la consultazione degli itinerari su un unico portale, condividere i propri itinerari con tutti gli altri appassionati utilizzatori della piattaforma, leggere e rilasciare recensioni, scrivere e ricevere messaggi da e verso altri utenti pertinenti un certo itinerario.

Una piattaforma concentrata e prettamente dedicata, facilita la ricerca, la consultazione e la comunicazione con persone che condividono la stessa passione e ciò diventa possibile in un unico strumento chiamato NaTour21.

## 9.3 Personas

Sulla base del precedente questionario, la creazione delle Personas ci ha permesso di rappresentare quella classe di utenti che potrebbero utilizzare l'applicazione. Questa rappresentazione ci ha fatto ben comprendere quali sono le esperienze, le esigenze e gli obiettivi di utenti chiave.



**Marco  
Bavagnoli**  
Età 50  
Programmatore Informatico

**GOALS**

- esplorare nuovi posti
- migliorare il proprio benessere
- condividere ricordi con i propri amici

**PAIN POINTS**

- uso troppe risorse per ricercare nuove avventure
- perdo parecchie informazioni importanti che renderebbero il mio viaggio migliore

**Brands**





**Giovanni  
Aiello**  
Età 38  
Docente

**GOALS**

- viaggiare e scoprire
- imparare bene la lingua inglese
- praticare escursioni impegnative

**PAIN POINTS**

- socializzo con pochissime persone
- spesso pratico itinerari in solitudine

**Brands**





## Elizabeth Luongo

Età 29  
Studente

### GOALS

- raggiungere i miei obiettivi formativi
- conoscere nuove culture
- imparare più lingue

### PAIN POINTS

- non ho tanti amici
- difficoltà a programmare un itinerario
- non riesco mai a concludere un percorso impegnativo

#### Brands

**patagonia**



## Antonia Russo

Età 34  
Ingegnere Biomedico

### GOALS

- rendermi migliore
- viaggiare in gruppo
- conoscere nuove persone

### PAIN POINTS

- sono permalosa
- mi risulta difficile organizzare escursioni in gruppi di più persone

#### Brands



**patagonia**



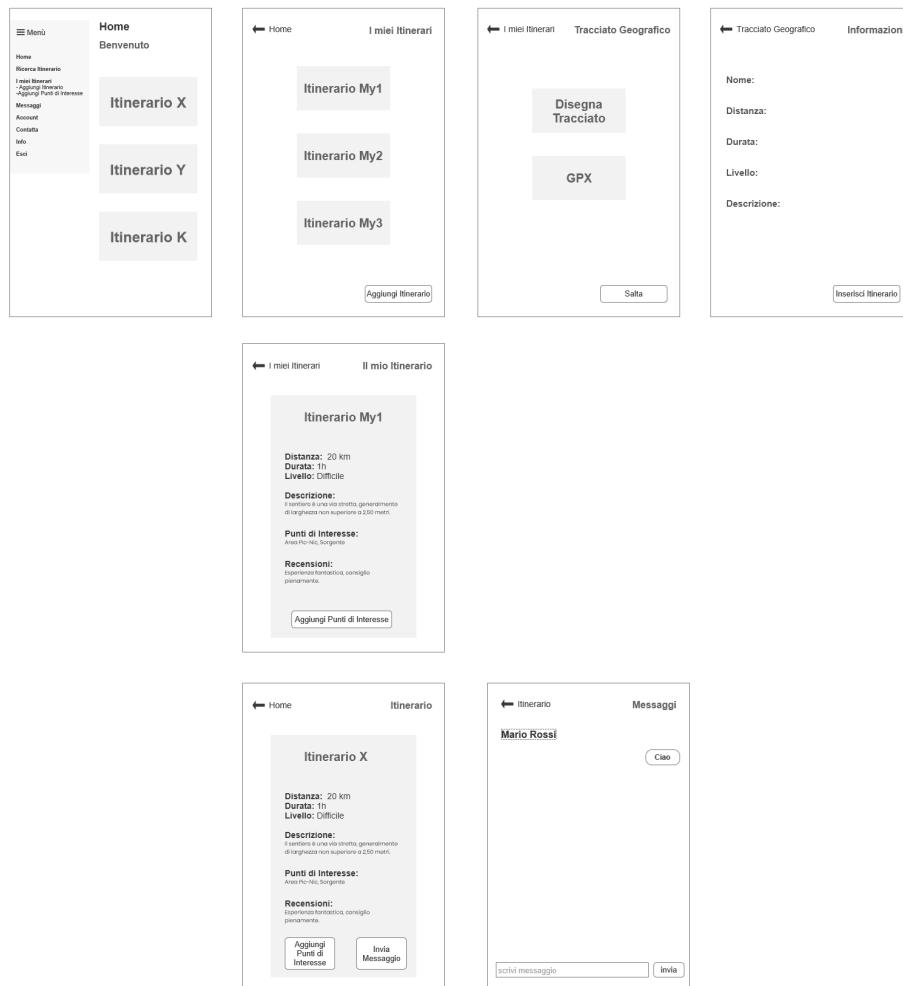
# 10. Valutazione e Test Di Usabilità

La valutazione dell'usabilità è stata eseguita utilizzando alcuni test di usabilità per lo specifico prodotto, in particolare di test formativi su prototipi iniziali di bassa fedeltà e ulteriori test di usabilità su prototipi di media fedeltà.

I primi sono stati eseguiti durante il ciclo iterativo di progettazione iniziale, tali test sono basati su bozze di prototipi molto semplici, disegnati digitalmente e poi stampati per poi essere stati sottoposti a prove d'uso ad un gruppo di utenti-tester ristretto facendo simulare l'interazione tra le varie schermate.

Questa fase è stata utile per migliorare l'usabilità del prodotto specie la gestione delle diverse interfacce, dove è stato possibile evidenziare in tempi rapidi, grazie anche al costo minimo di realizzazione dei prototipi tutti i difetti macroscopici e quelli di minore entità.

Di seguito si riportano lo sviluppo dei primi prototipi di bassa fedeltà e le relative osservazioni sul primo set di test formativi:

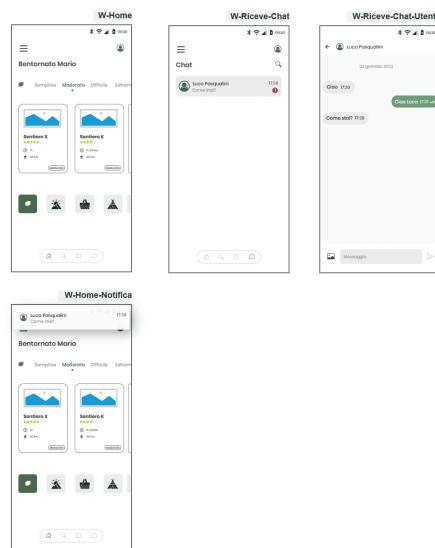
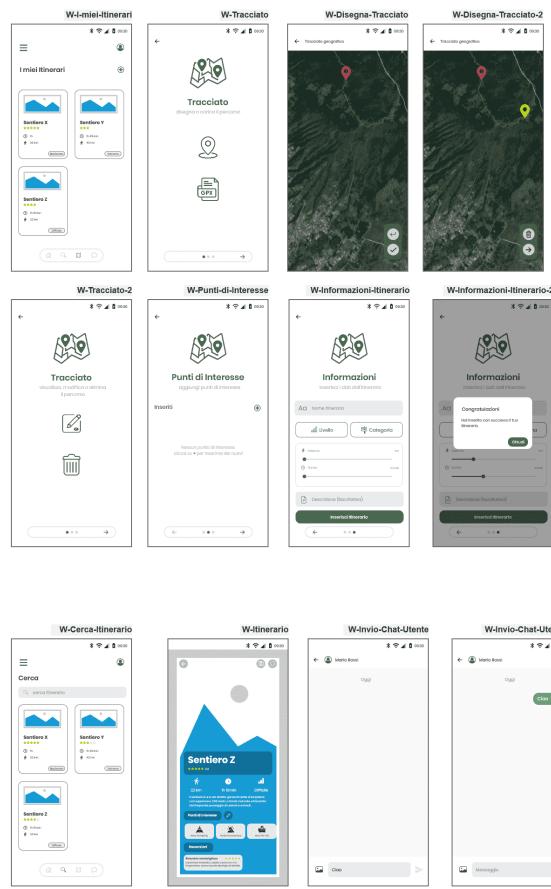


Durante la prima fase di test, abbiamo notato che la gestione tra le varie schermate risultava troppo lunga, in particolare durante l'aggiunta di un nuovo itinerario. Per un compito specifico si otteneva comunque un flusso di operazioni lineare non condizionato da altre scelte, ma se l'utente durante l'aggiunta di un nuovo itinerario, voleva inserire alcuni Punti di Interesse pertinenti a quell'itinerario, poteva farlo ma solamente dopo l'inserimento. Questo comportava a ritornare nella schermata W-I-Miei-Itinerari scelta dal menù, scegliere l'ultimo itinerario inserito e aggiungere Punti di Interesse per quell'itinerario.

Sulla base di queste osservazioni, abbiamo dato la possibilità diretta all'utente di inserire eventuali Punti di Interesse durante l'aggiunta di un nuovo itinerario. Conseguenzialmente abbiamo ulteriormente migliorato l'interazione tra le diverse schermate, togliendo dal menù alcune voci, tra cui: Ricerca Itinerario, I miei Itinerari, Messaggi (Chat) rese poi disponibili su un bottom bar, persistente durante la navigazione da schermata a schermata. Quest'ultima modifica permette all'utente di spostarsi tra le varie schermate con facilità e comodità, senza eseguire troppi passi per arrivare all'obiettivo finale.

Questi ultimi sono stati classificati come Interventi necessari che permettevano comunque l'utilizzo del sistema ma spesso con percorsi alternativi e abbastanza lunghi.

Grazie alle osservazioni ottenute dalla prima fase di test formativi, la fase successiva ha portato alla creazione di prototipi più avanzati, mostrando un design quasi definitivo e vicino a quello finale. È risultato utile riproporre dei prototipi più strutturati il che ha permesso di ottenere dagli utenti dei feedback più chiari che riguardavano punti di forza e di debolezza del sistema, gli aspetti che dovevano essere migliorati, e quelli graditi maggiormente.



Durante questi test di usabilità, è stato richiesto al gruppo di utenti-tester una serie di compiti da eseguire e da questi test di compiti sono state raccolte delle misure oggettive, in particolare, il tempo impiegato (in sec) da ogni tester per l'esecuzione di ciascun compito e il tasso di successo dei compiti che ciascun tester è riuscito a portare a termine.

Di seguito i cinque compiti richiesti ai tester che ci hanno permesso di valutarne l'usabilità del prodotto in base ai risultati ottenuti:

1. Registrazione a NaTour21
2. Inserimento di un nuovo itinerario
3. Invio messaggio ad un utente
4. Visualizzare i miei itinerari
5. Ricerca itinerario

Rappresentazione del tempo impiegato:

	<b>Compito 1</b>	<b>Compito 2</b>	<b>Compito 3</b>	<b>Compito 4</b>	<b>Compito 5</b>
<b>Tester 1</b>	~40s	~80s	~40s	~10s	~40s
<b>Tester 2</b>	~60s	~90s	~20s	~10s	~80s
<b>Tester 3</b>	~50s	~40s	~50s	~10s	~90s
<b>Tester 4</b>	~30s	~50s	~40s	~5s	~30s

Rappresentazione del tasso di successo:

	<b>Compito 1</b>	<b>Compito 2</b>	<b>Compito 3</b>	<b>Compito 4</b>	<b>Compito 5</b>
<b>Tester 1</b>	S	S	P	S	S
<b>Tester 2</b>	S	S	S	S	P
<b>Tester 3</b>	S	F	P	S	F
<b>Tester 4</b>	S	P	S	S	S

S=successo, F=fallimento, P=successo parziale TS= tasso di successo

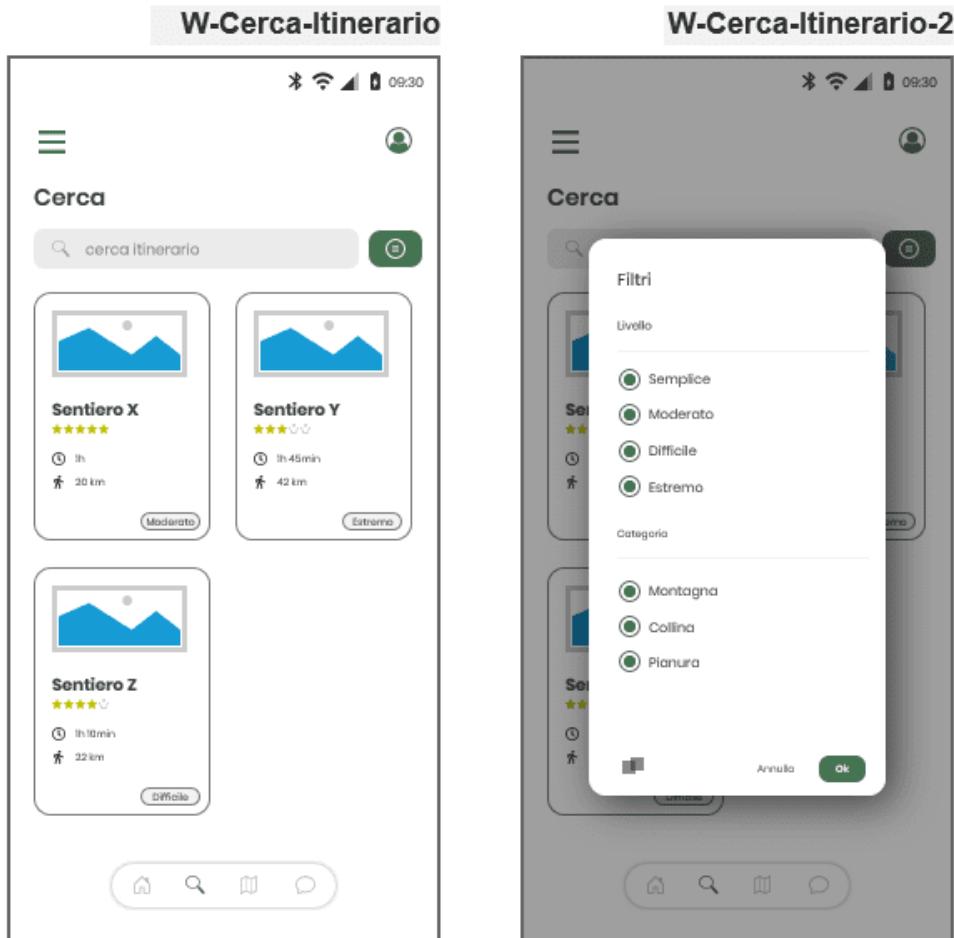
TS= 84%

I primi test sono stati di grande importanza, avendo avuto un ruolo fondamentale per lo sviluppo di prototipi successivi. I test successivi basati sui prototipi di media fedeltà hanno avuto un risultato più che soddisfacente.

Per i test di compito si è ottenuto un tasso di successo pari all'84% dove ciascun utente-tester è riuscito quasi sempre a portare a termine il compito assegnato, inoltre con tempi di esecuzione di ciascun compito sufficientemente bassi, eccetto il compito numero 5 che a differenza degli altri compiti ha messo in evidenza una leggera difficoltà in più nel ricercare uno specifico itinerario. Infatti da come possiamo notare dalla tabella il compito 5 produce un tempo complessivo di circa 4 minuti, nettamente superiore a tutti gli altri tempi complessivi di altri compiti.

Sulla base di questo si è deciso di apportare ulteriori interventi nella tab Cerca Itinerario, aggiungendo al campo di ricerca un filtraggio di ricerca più affinato che permette agli utenti utilizzatori di ricercare un itinerario più velocemente, filtrando per livello di difficoltà e categoria di appartenenza.

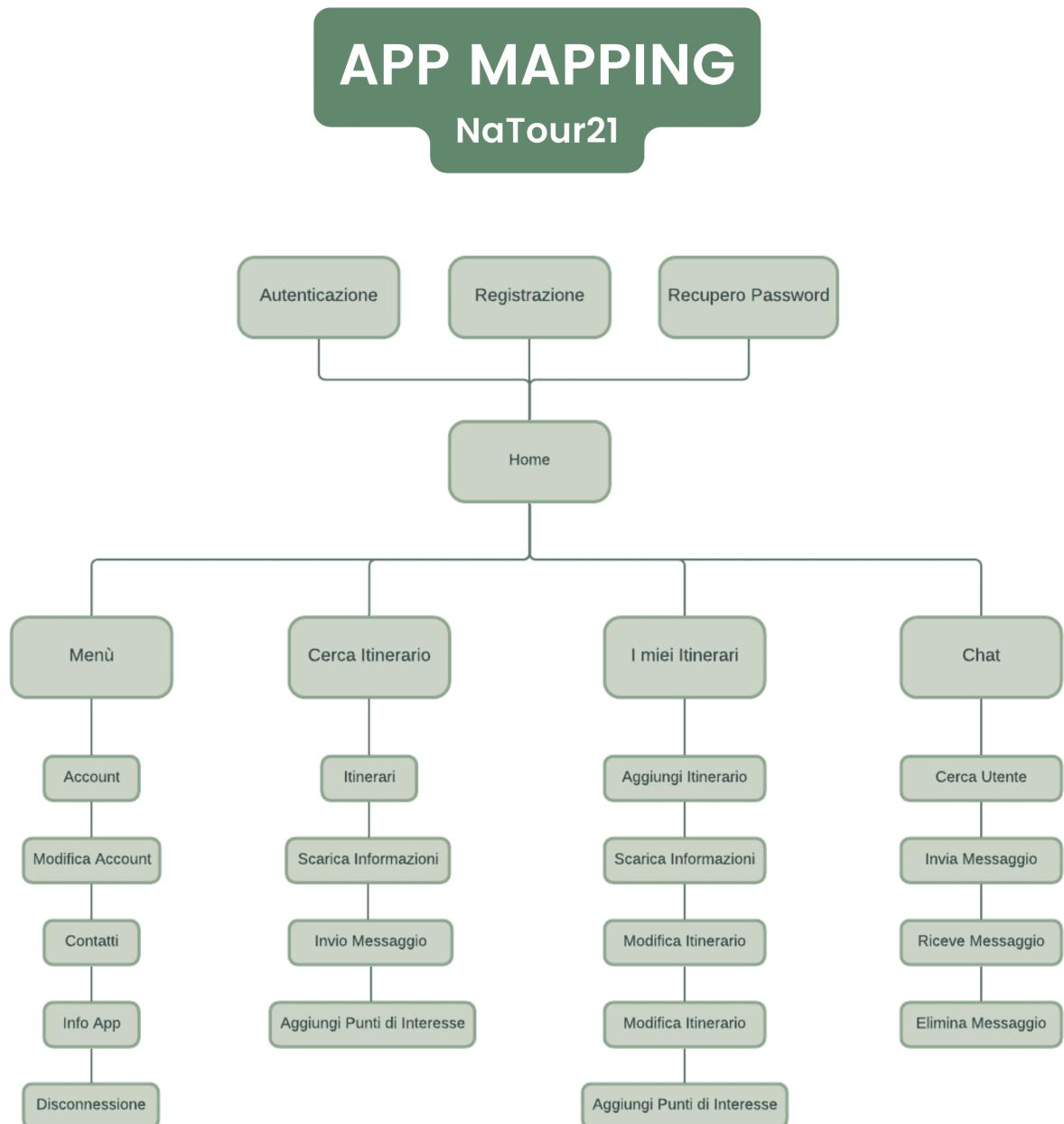
Di seguito si riporta l'intervento apportato alla tab *Cerca Itinerario*, la nuova versione include l'aggiunta di un filtraggio all'interno della barra di ricerca come mostrati dai wireframe seguenti:



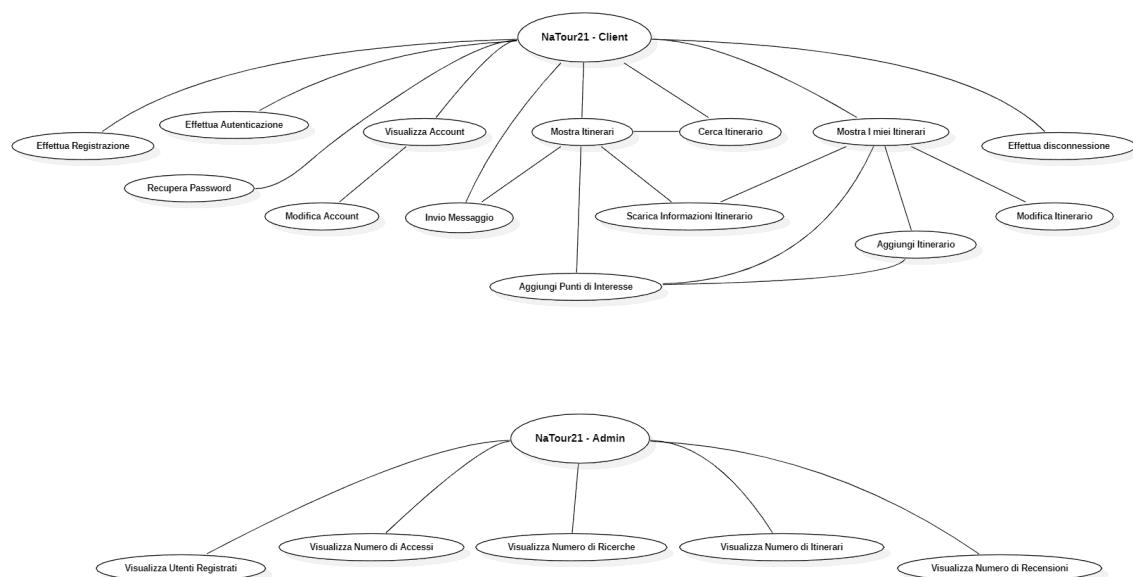
Di seguito si riporta l'intervento apportato alla schermata *Inserisci Informazioni Itinerario*, la nuova versione include l'aggiunta di una barra fluttuante al fine di migliorare la navigabilità tra le schermate, inoltre è stato rimosso il tasto *Aggiungi itinerario* (wireframe pg 79) aggiungendolo alla barra fluttuante, con consueta miglioria della leggibilità della schermata come mostrato nel wireframe seguente:



# 11. App Mapping



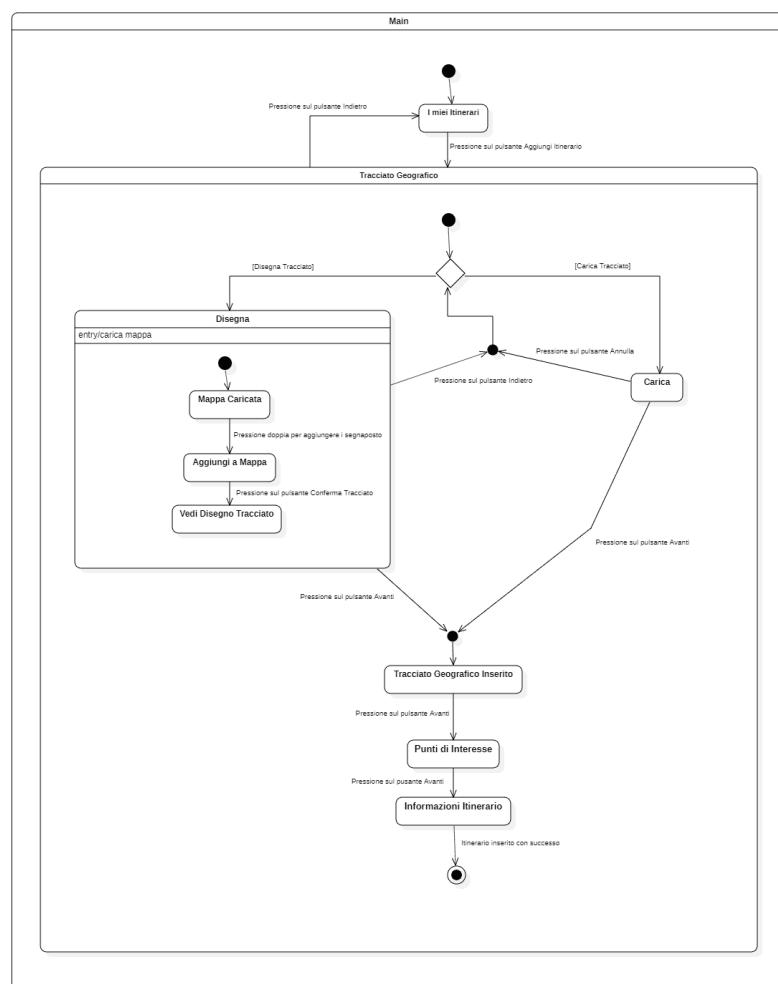
# 12. Gerarchie Funzionali



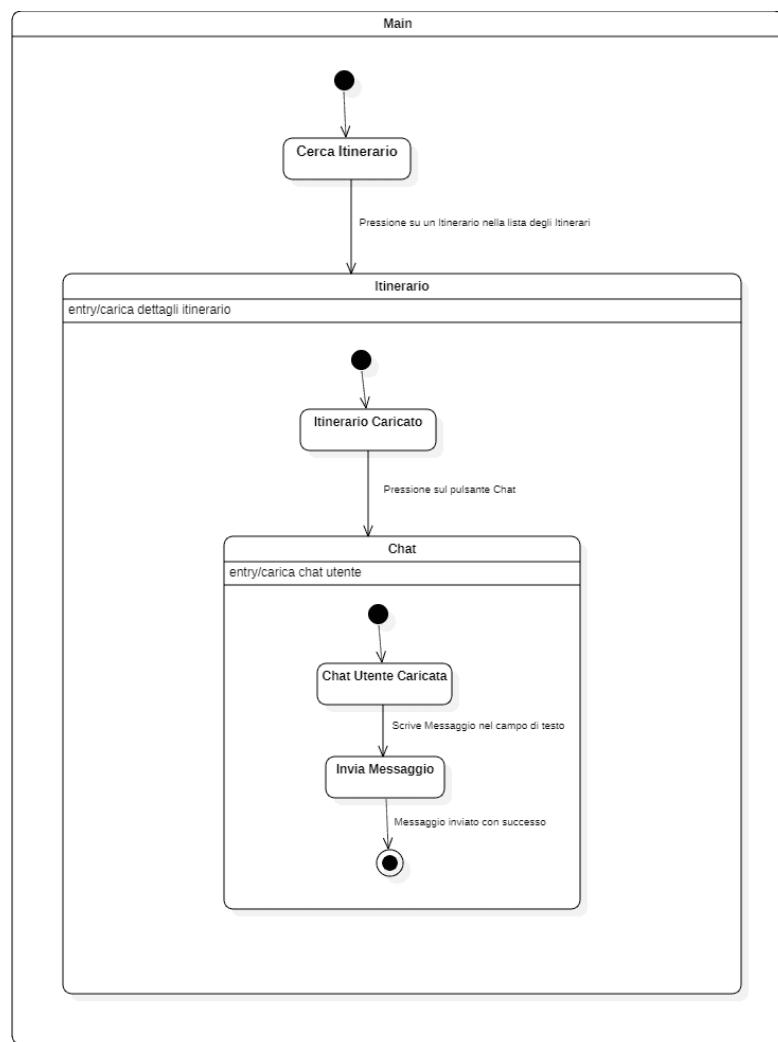
# 13. State Chart

Di seguito vengono mostrate alcune funzionalità dell'interfaccia grafica mediante state chart.

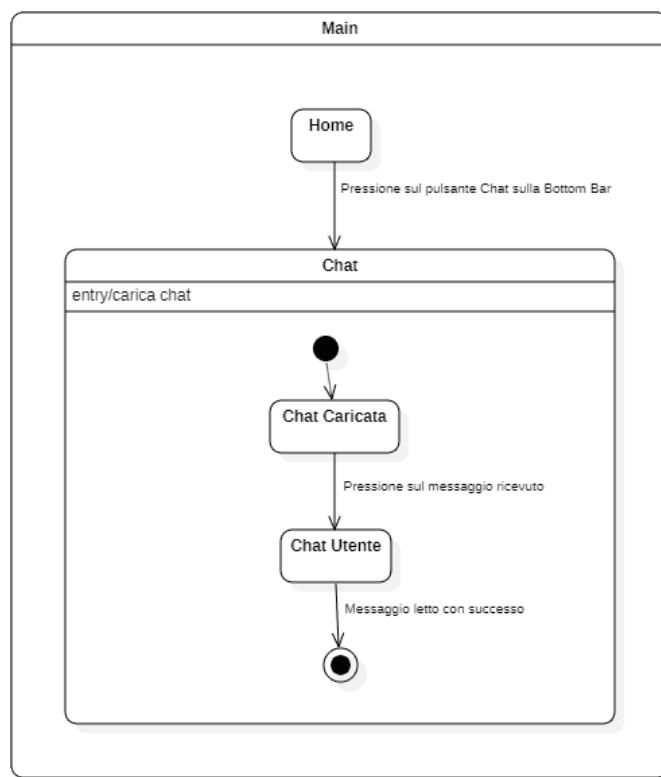
## 13.1 Inserimento Tracciato Geografico



## 13.2 Invia Messaggio Privato Utente



### 13.3 Lettura Messaggio Privato



# 14. Mockup

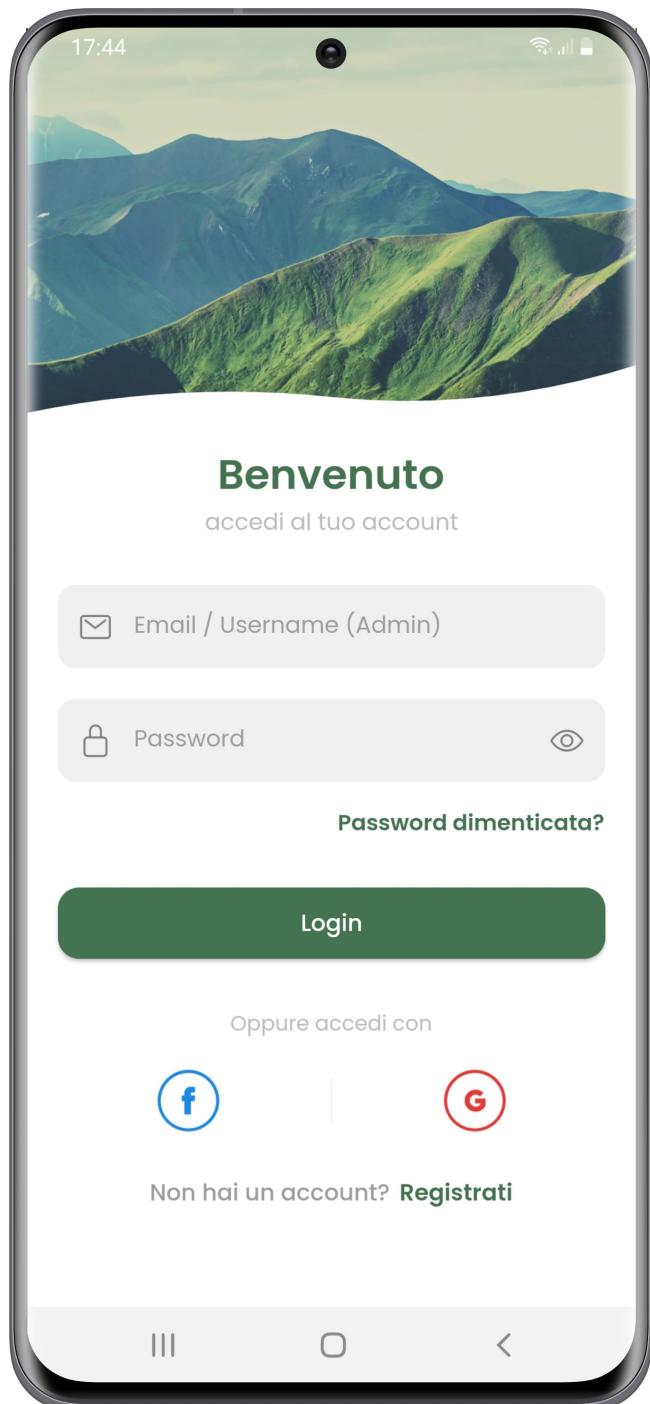
## 14.1 Rappresentazione mockups

Di seguito vengono mostrati tutti i Mockup relativi all'applicazione, sia lato client che admin. I mockup rappresentano l'interfaccia grafica dell'applicazione sviluppata secondo l'UX Procces, paragrafo 9 dell'attuale capitolo.

### 14.1.1 MU-SchermataDiAvvio



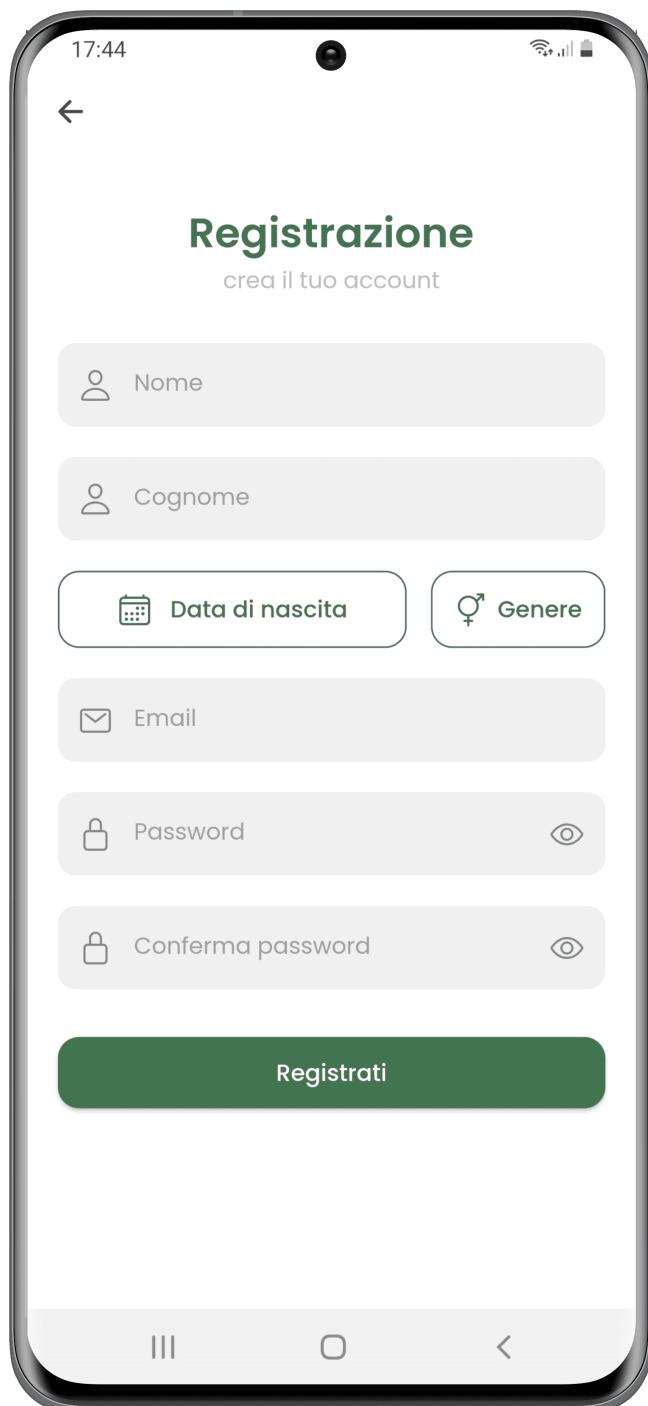
## 14.1.2 MU-Registrazione-Autenticazione



### 14.1.3 MU-RecuperoPassword



#### 14.1.4 MU-Registrazione



## 14.1.5 MU-Registrazione2



## 14.1.6 MU-Registrazione3



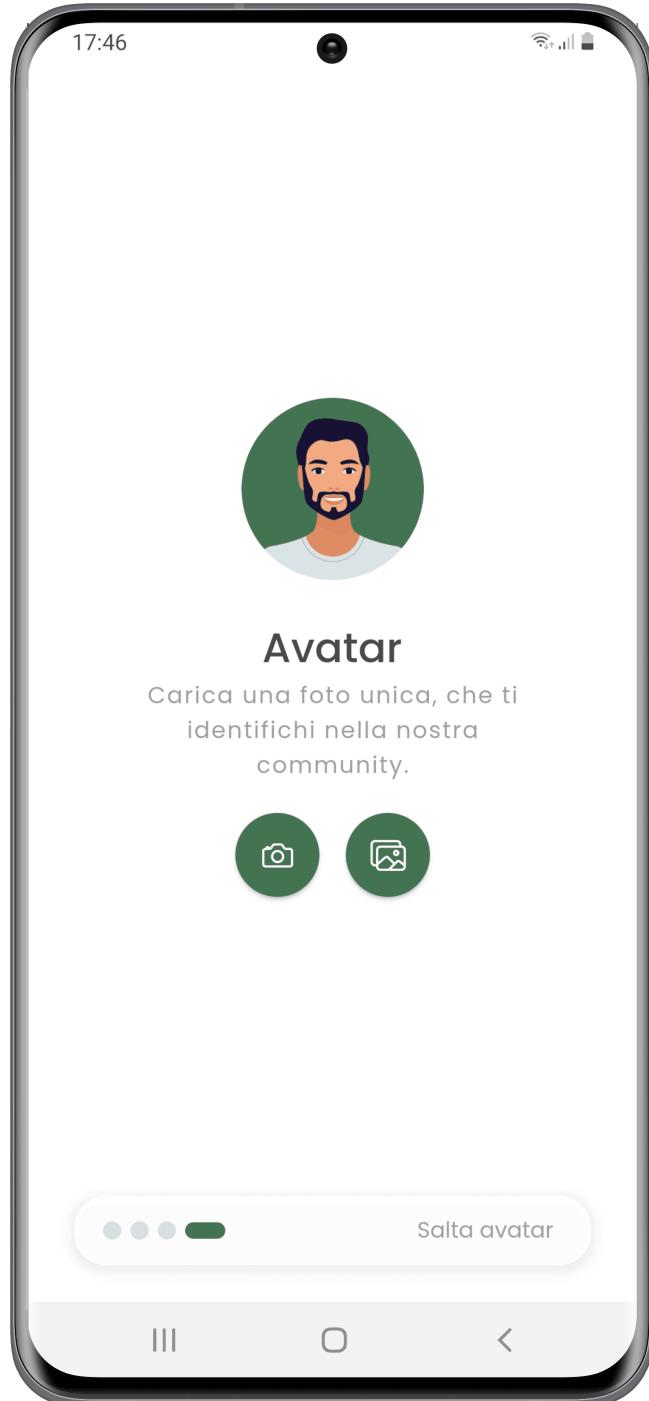
## 14.1.7 MU-Registrazione4



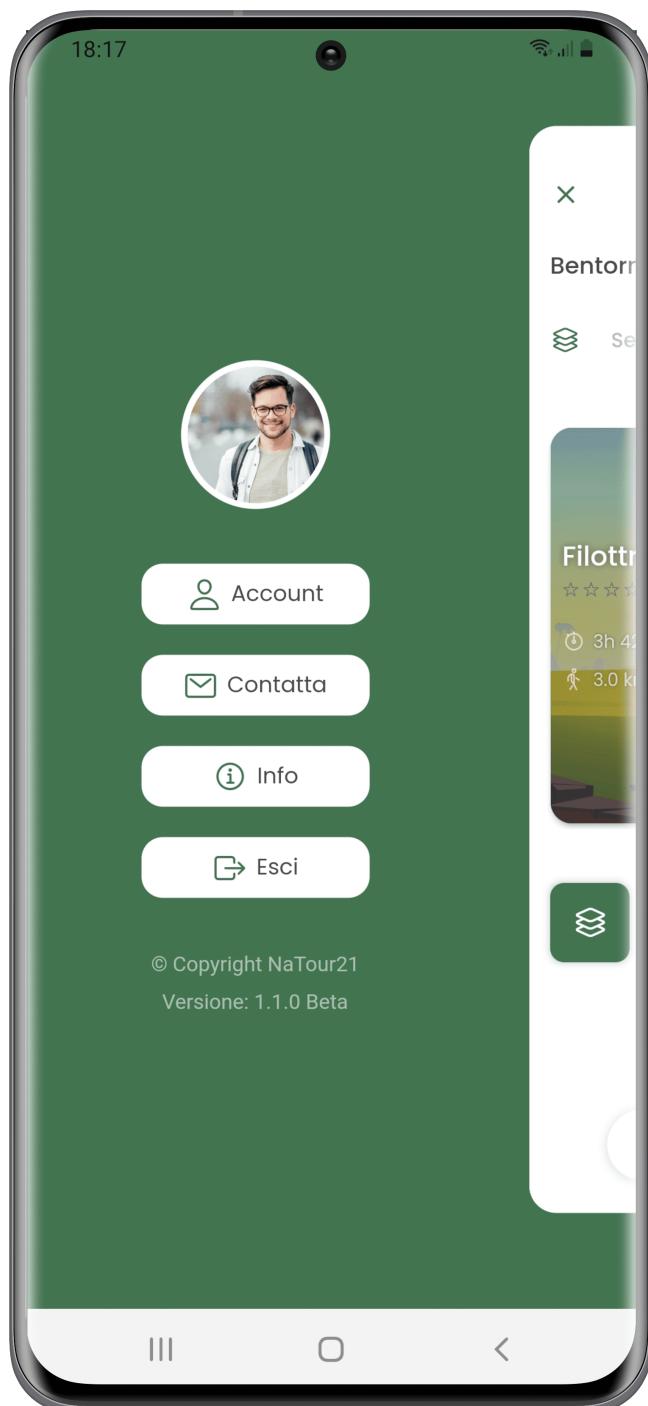
## 14.1.8 MU-Registrazione5



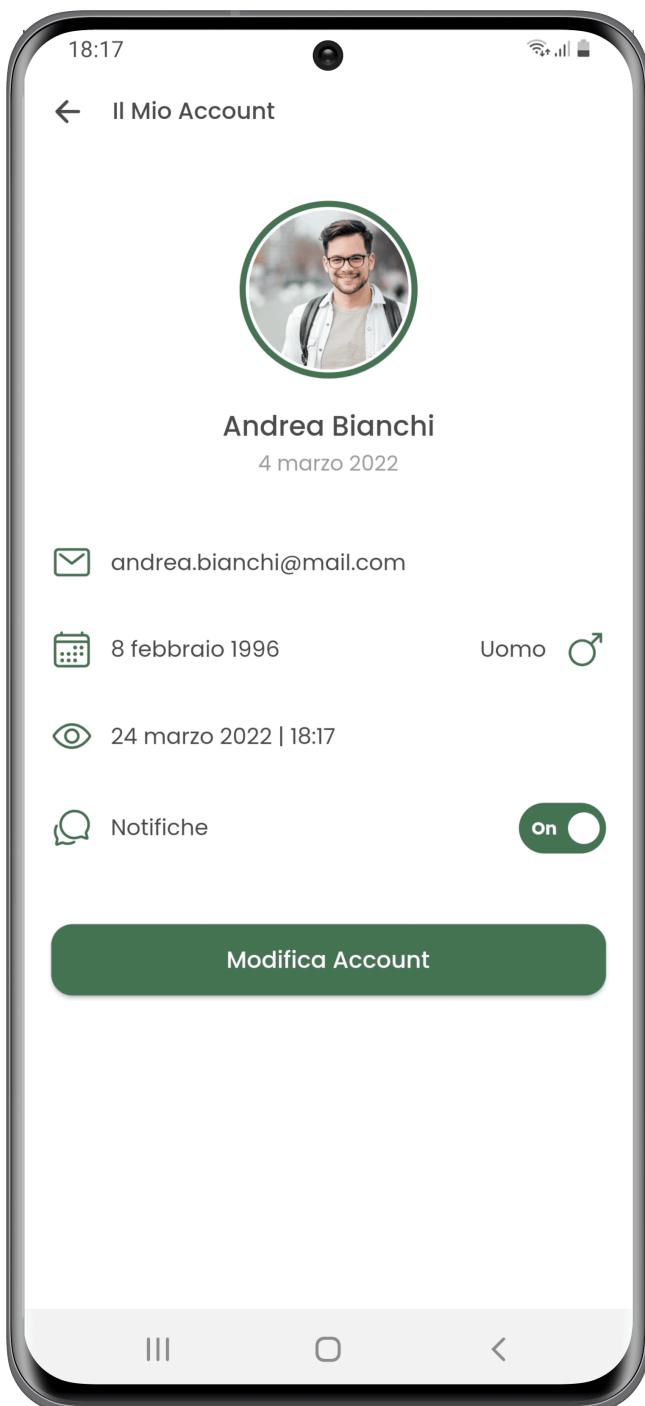
## 14.1.9 MU-Registrazione6



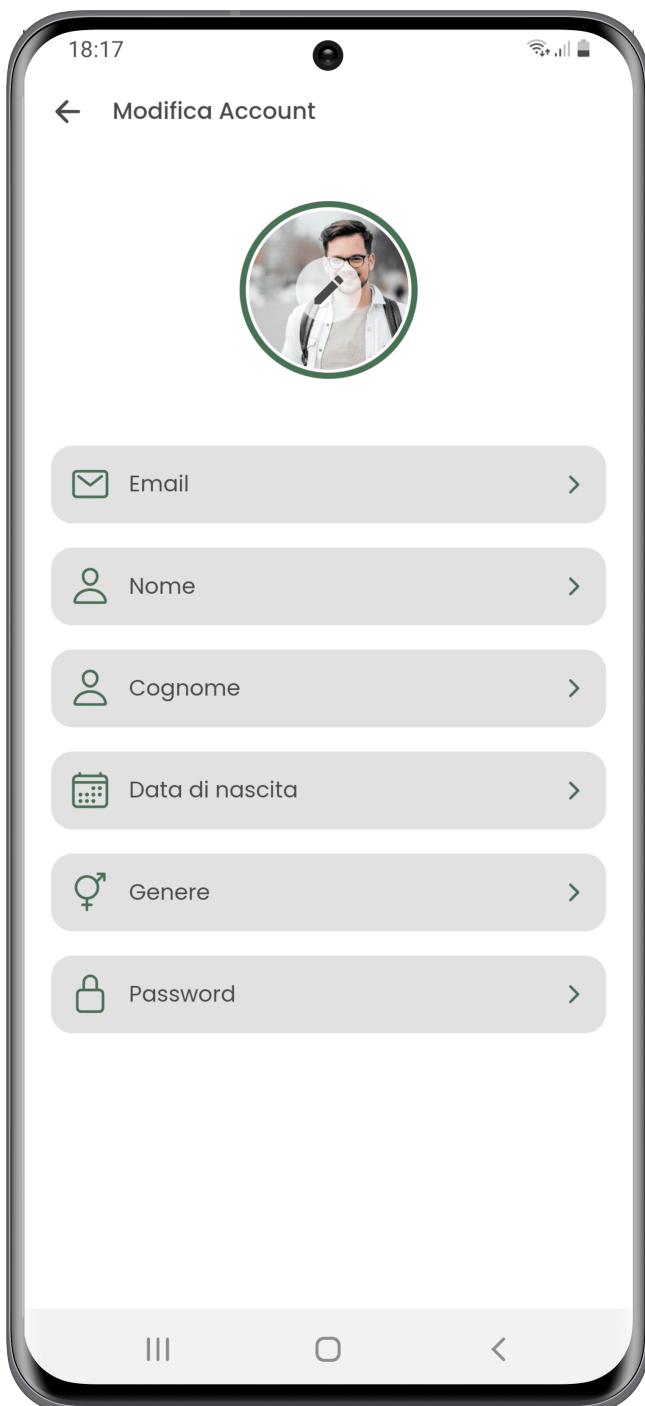
## 14.1.10 MU-Drawer



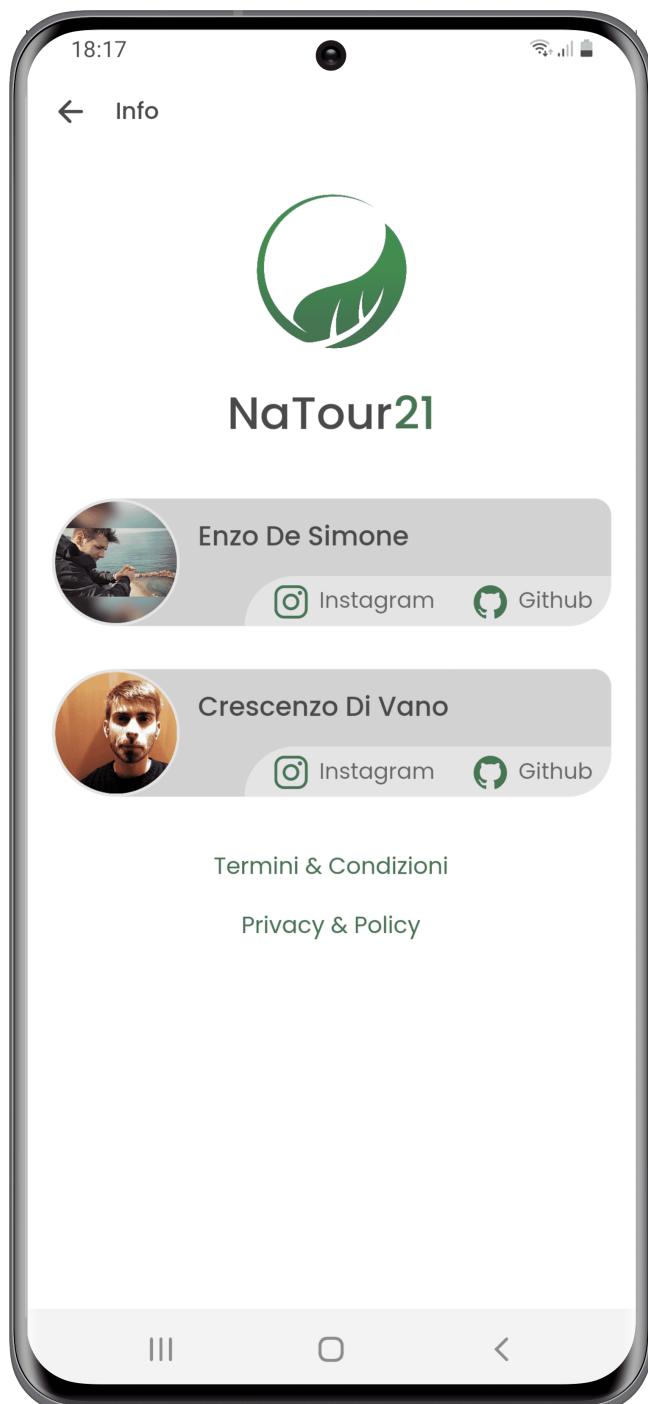
## 14.1.11 MU-Account



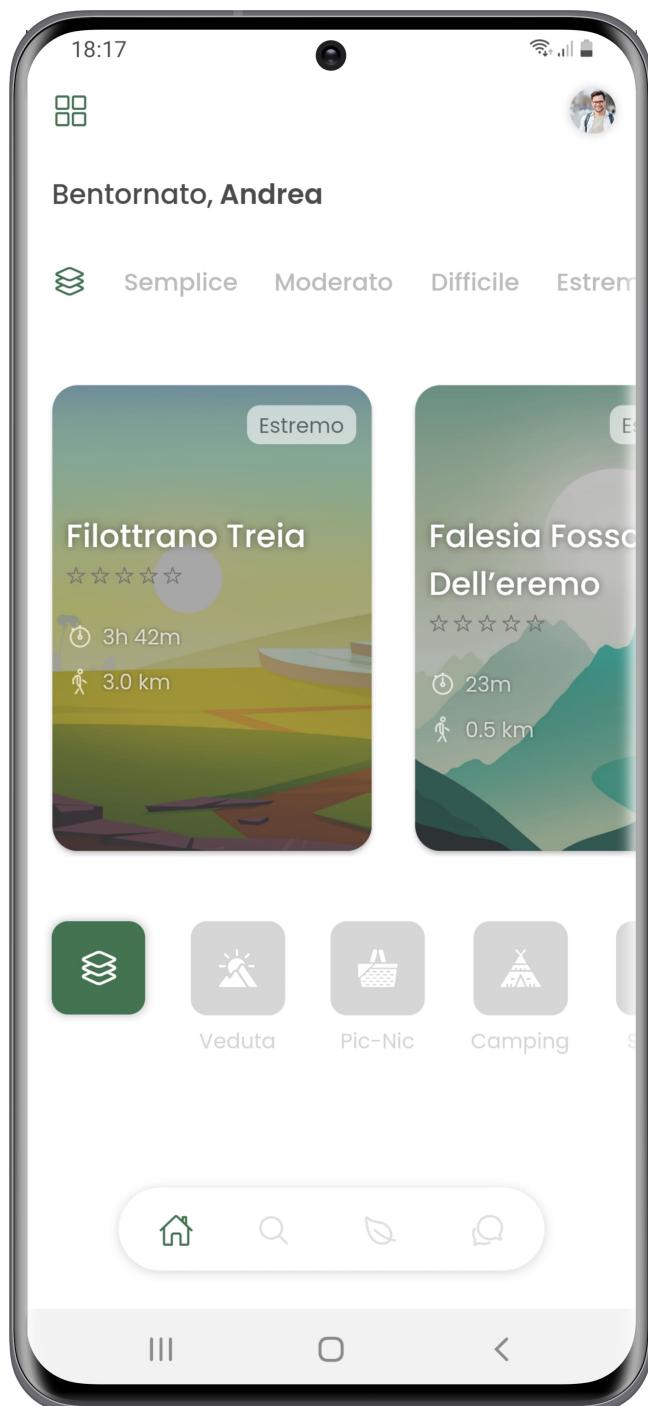
## 14.1.12 MU-ModificaAccount



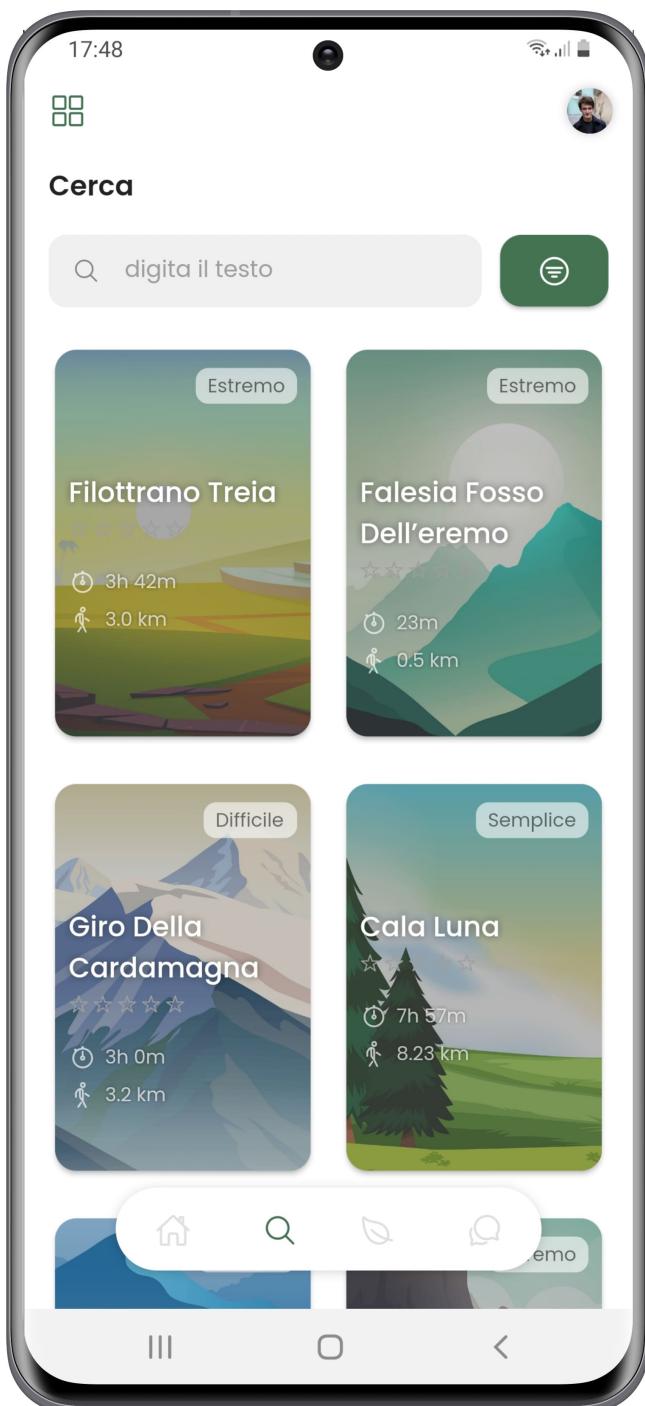
## 14.1.13 MU-Info



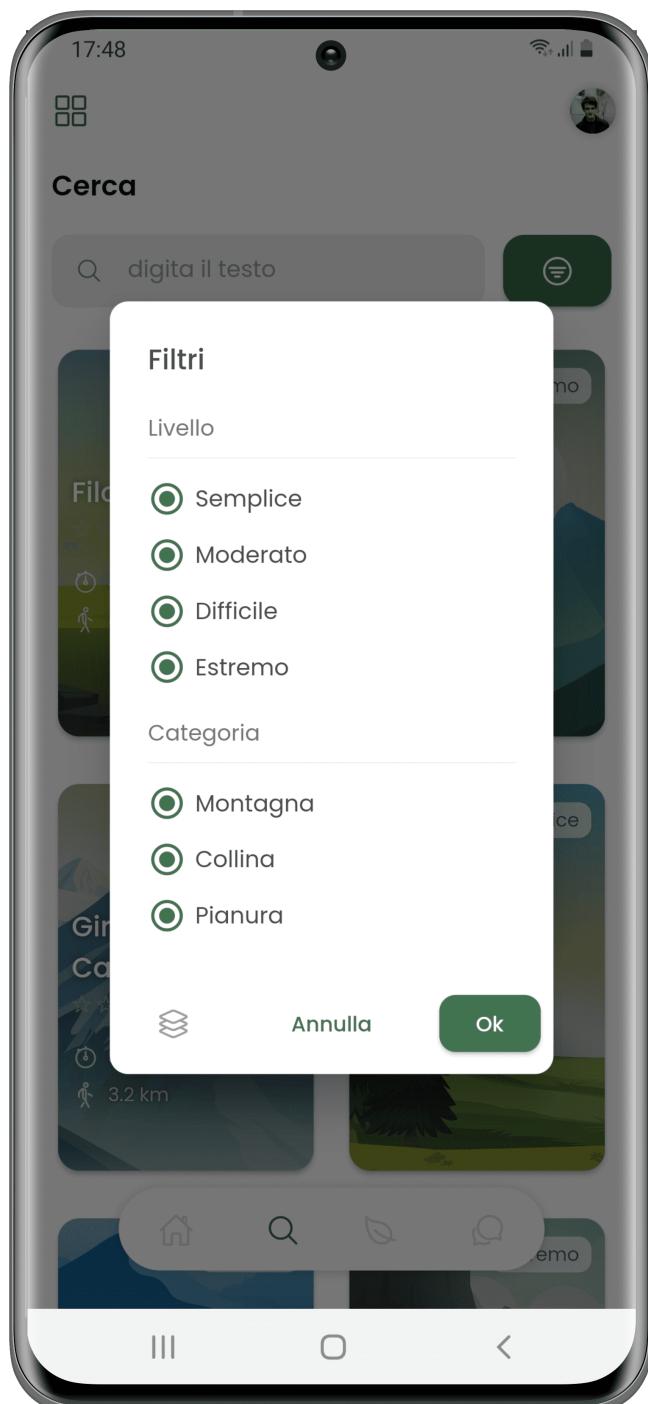
## 14.1.14 MU-Home



## 14.1.15 MU-RicercaItinerario



## 14.1.16 MU-RicercaItinerario2



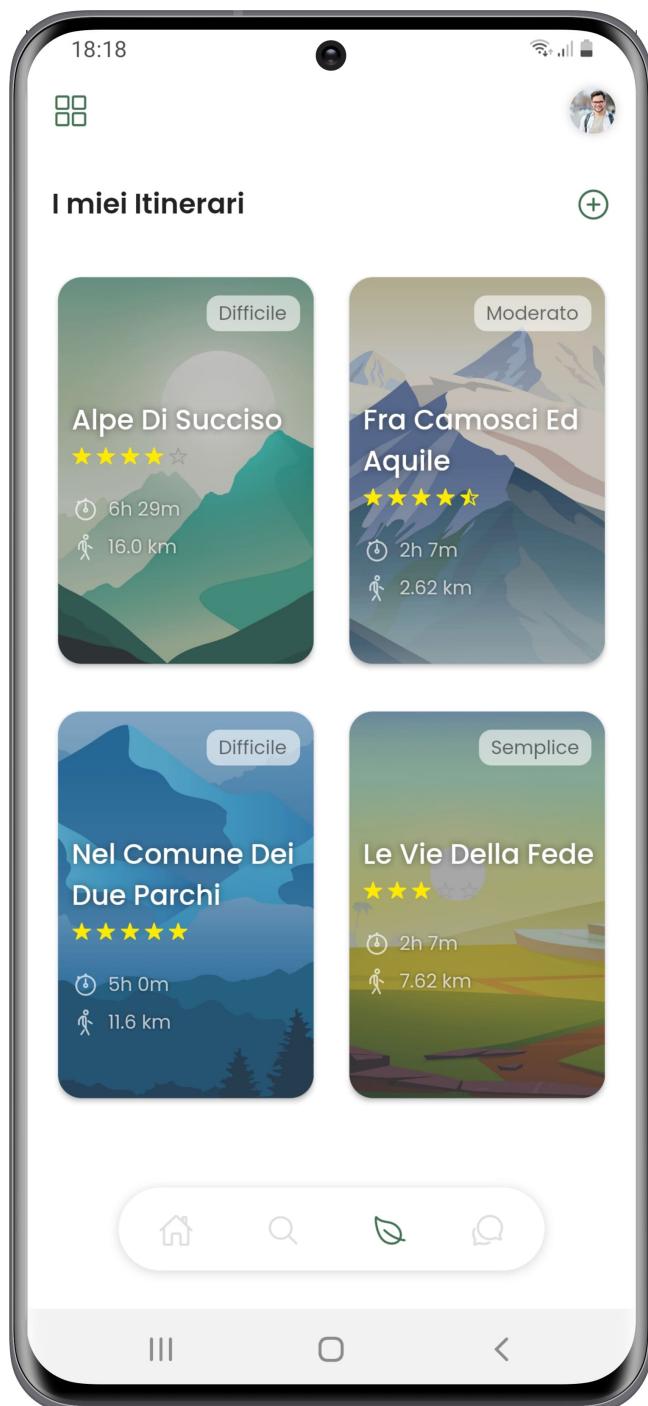
## 14.1.17 MU-Itinerario



## 14.1.18 MU-Itinerario2



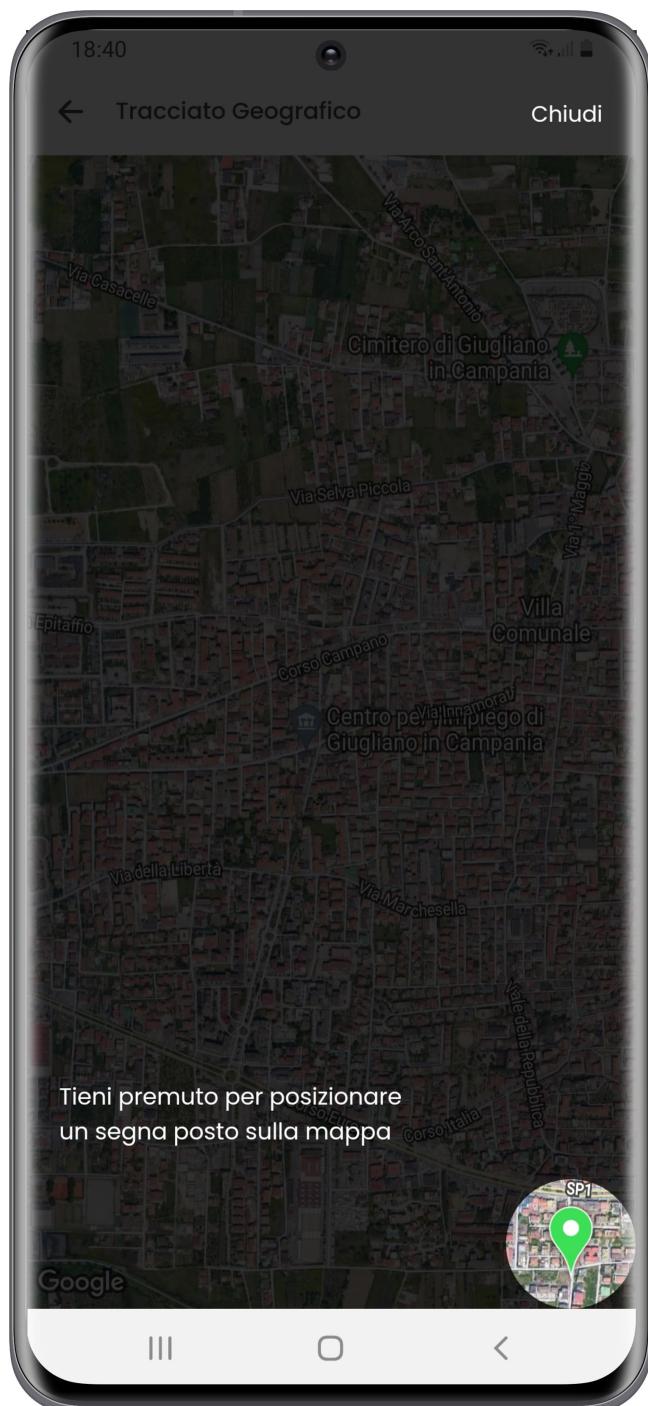
## 14.1.19 MU-ImieiItinerari



## 14.1.20 MU-TracciatoGeografico



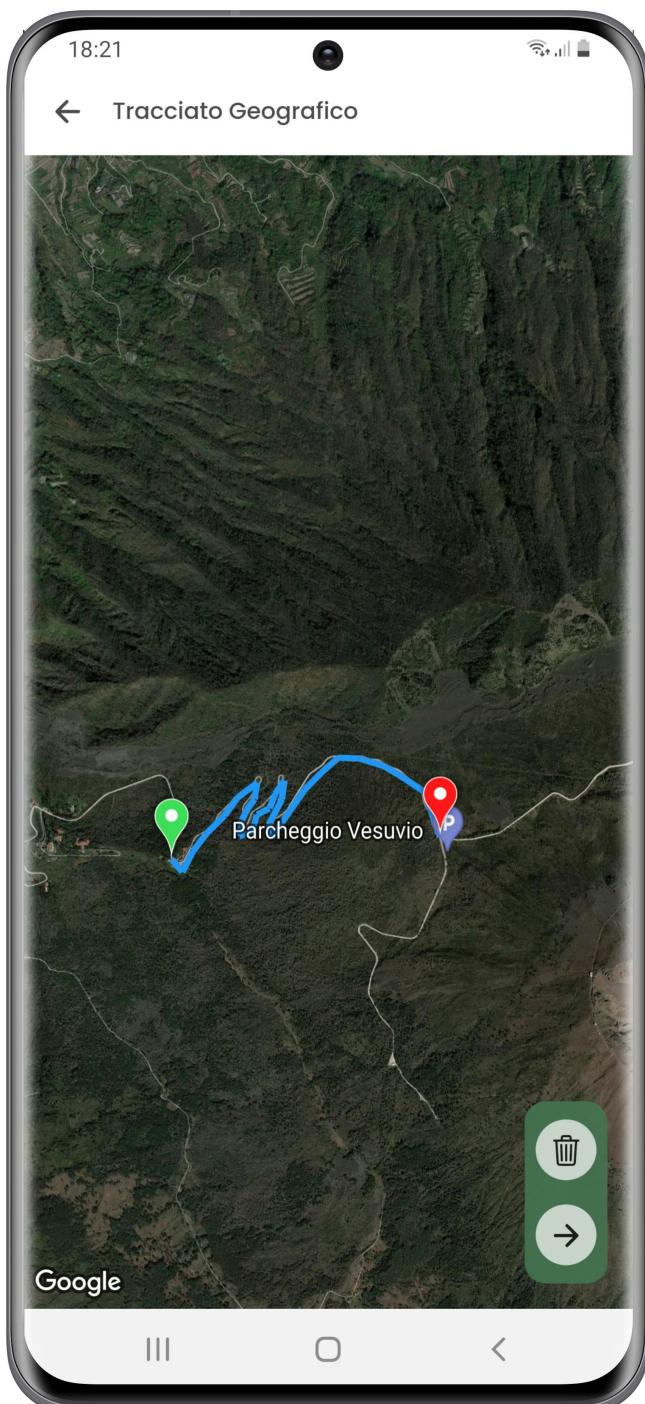
## 14.1.21 MU-TutorialTracciatoGeografico



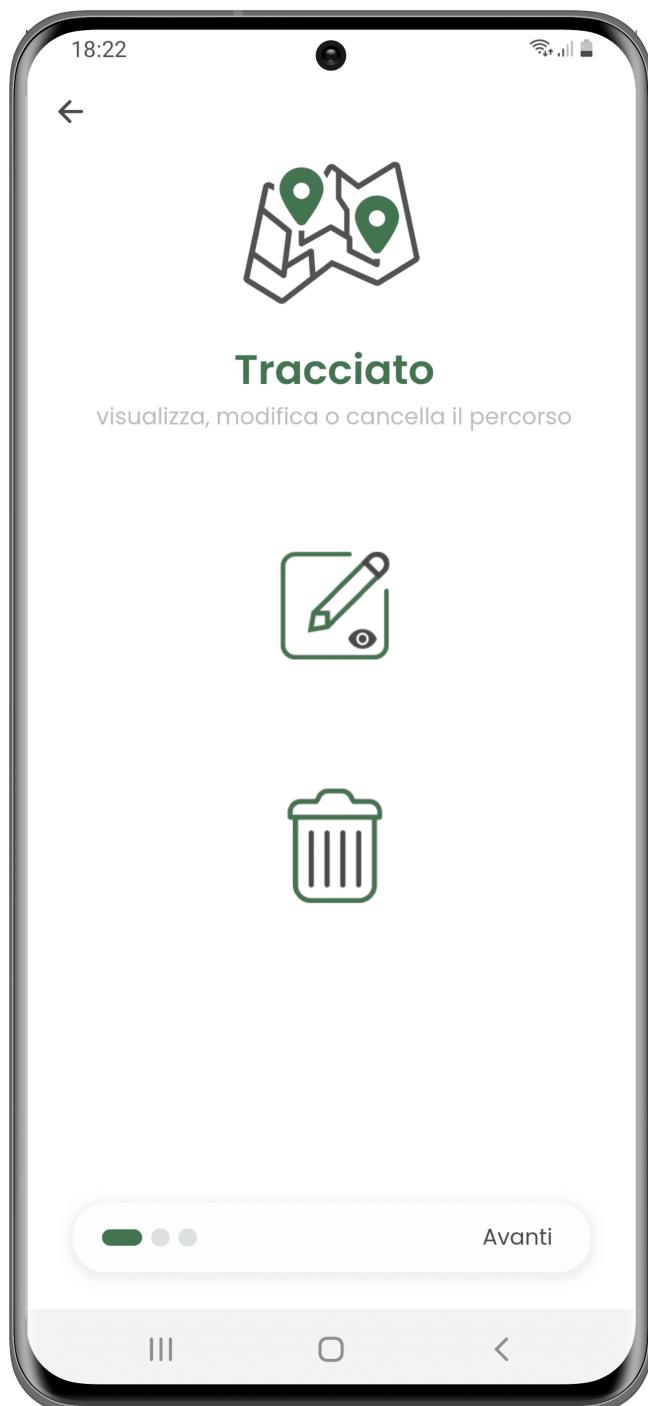
## 14.1.22 MU-DisegnaTracciatoGeografico



## 14.1.23 MU-DisegnaTracciatoGeografico2



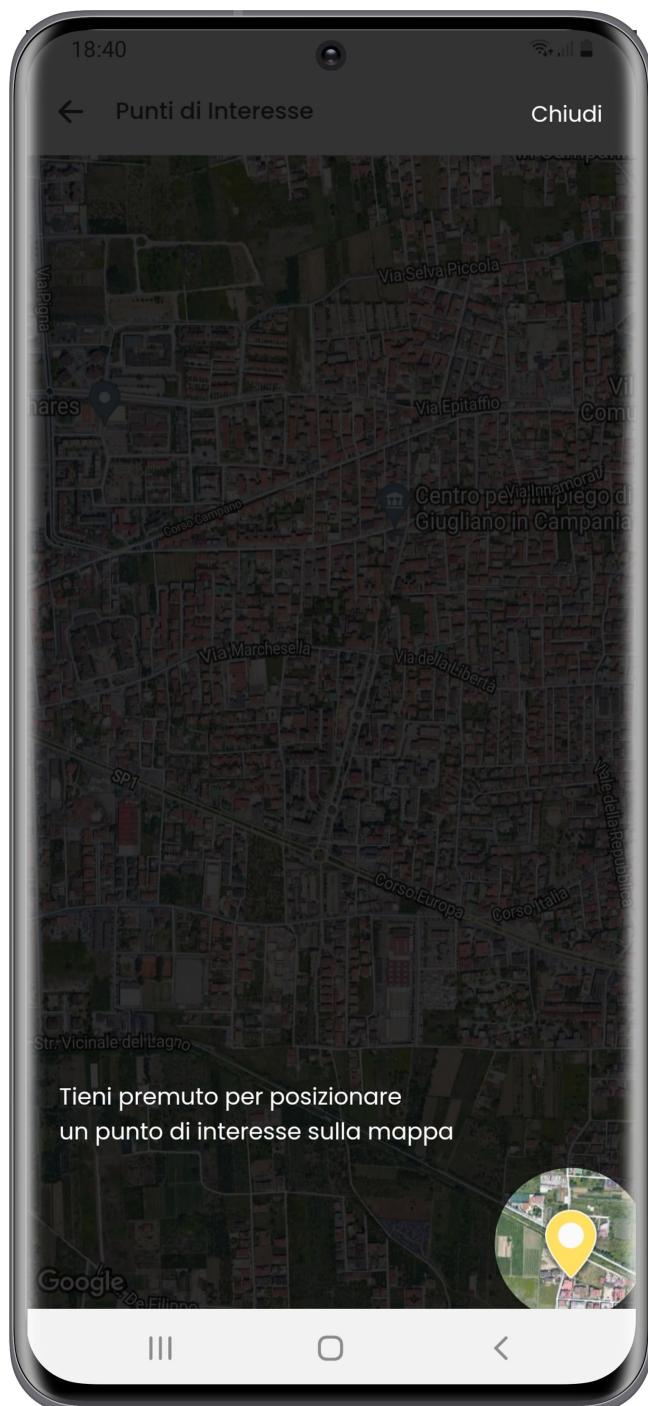
## 14.1.24 MU-TracciatoGeografico2



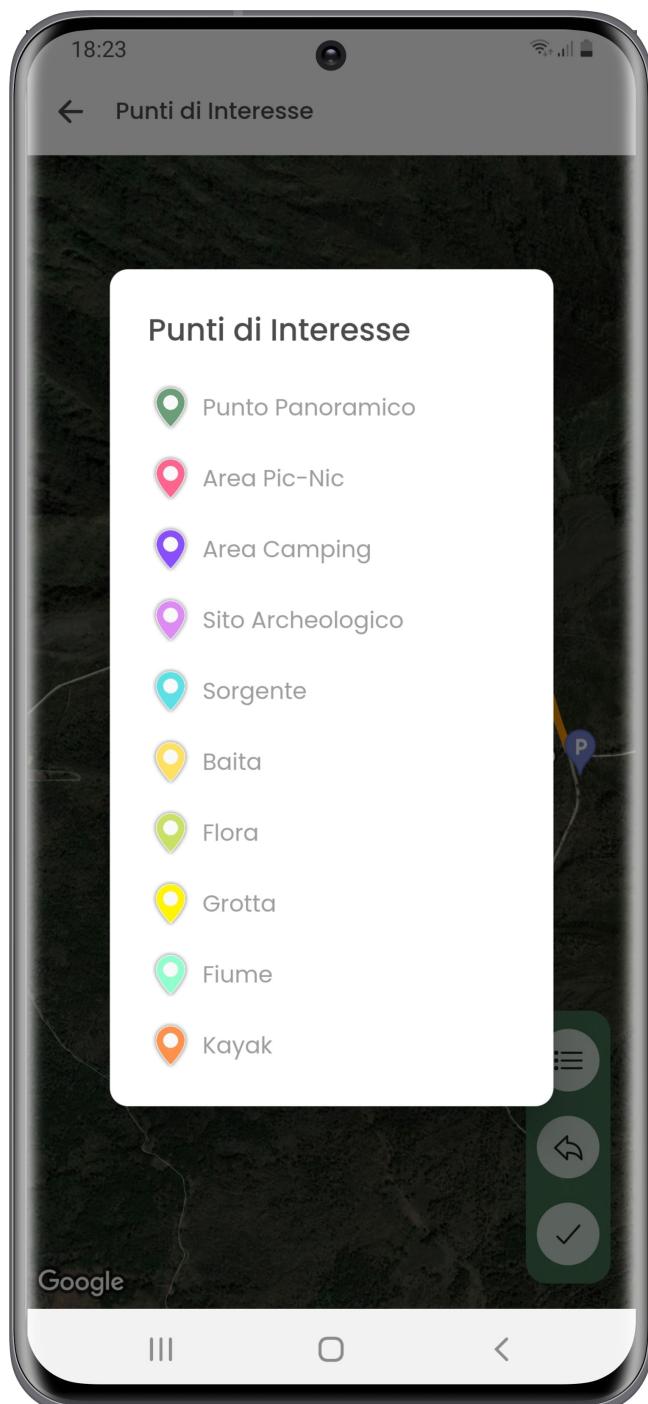
## 14.1.25 MU-PuntiDiInteresse



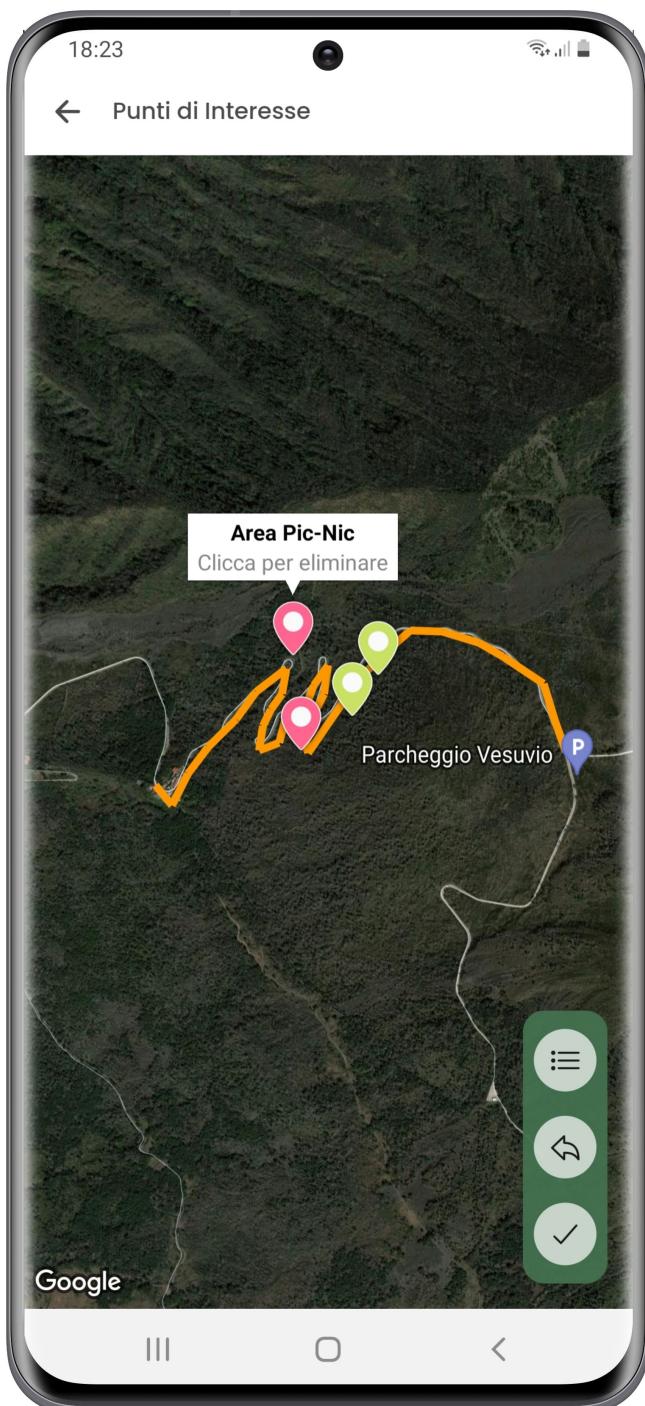
## 14.1.26 MU-TutorialPuntiDiInteresse



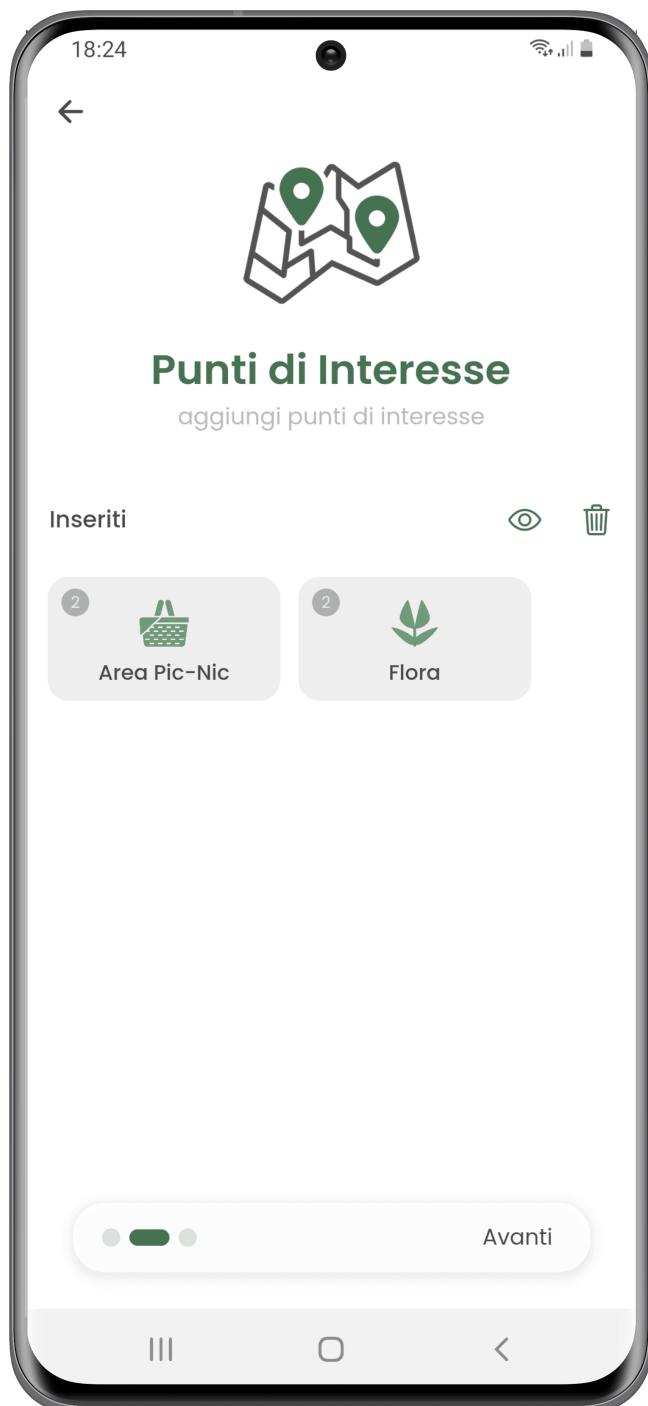
## 14.1.27 MU-SceltaPuntiDiInteresse



## 14.1.28 MU-InserisciPuntiDiInteresse



## 14.1.29 MU-PuntiDiInteresse2



## 14.1.30 MU-InformazioniItinerario



### 14.1.31 MU-InformazioniItinerario2



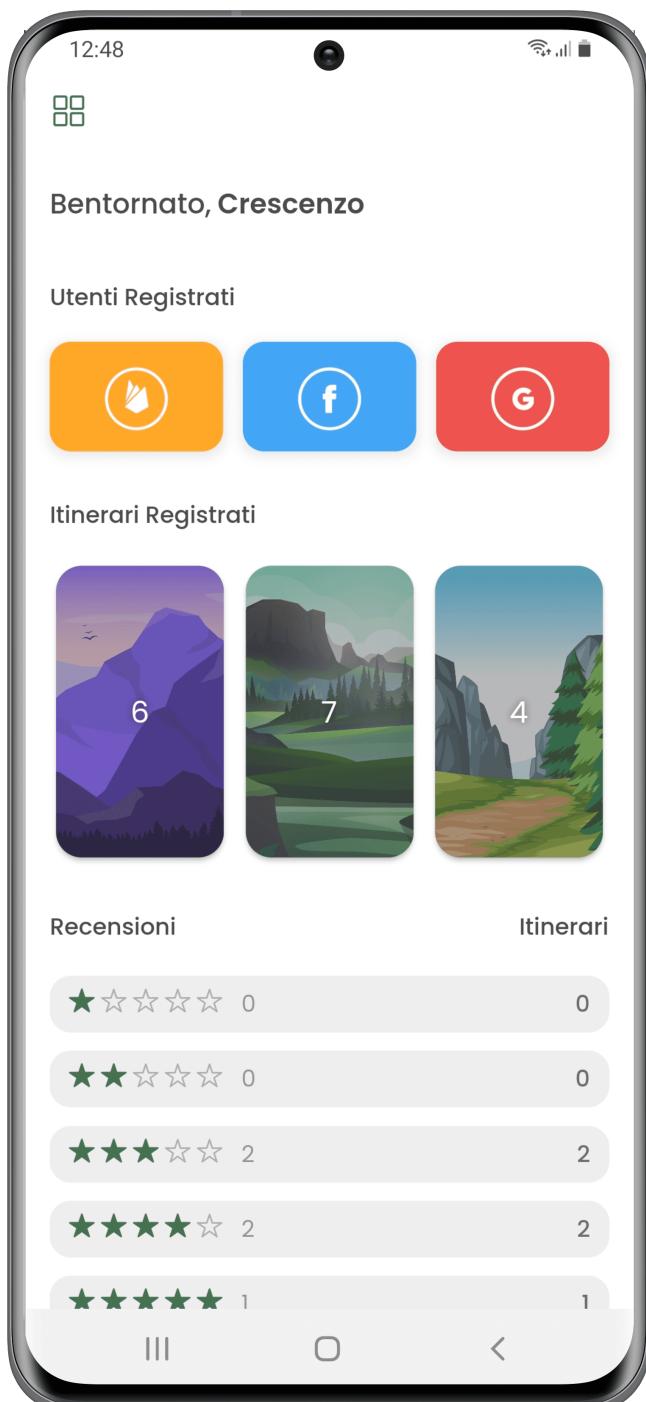
## 14.1.32 MU-Chat



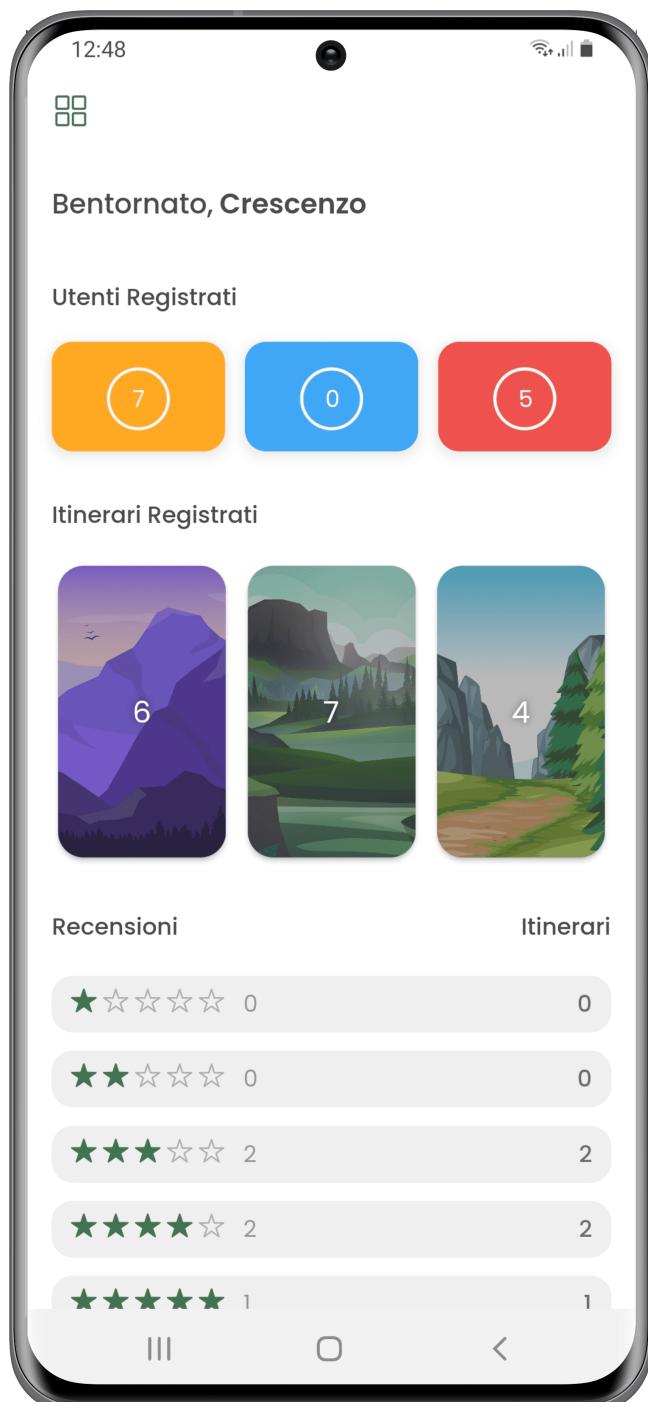
### 14.1.33 MU-Chat2



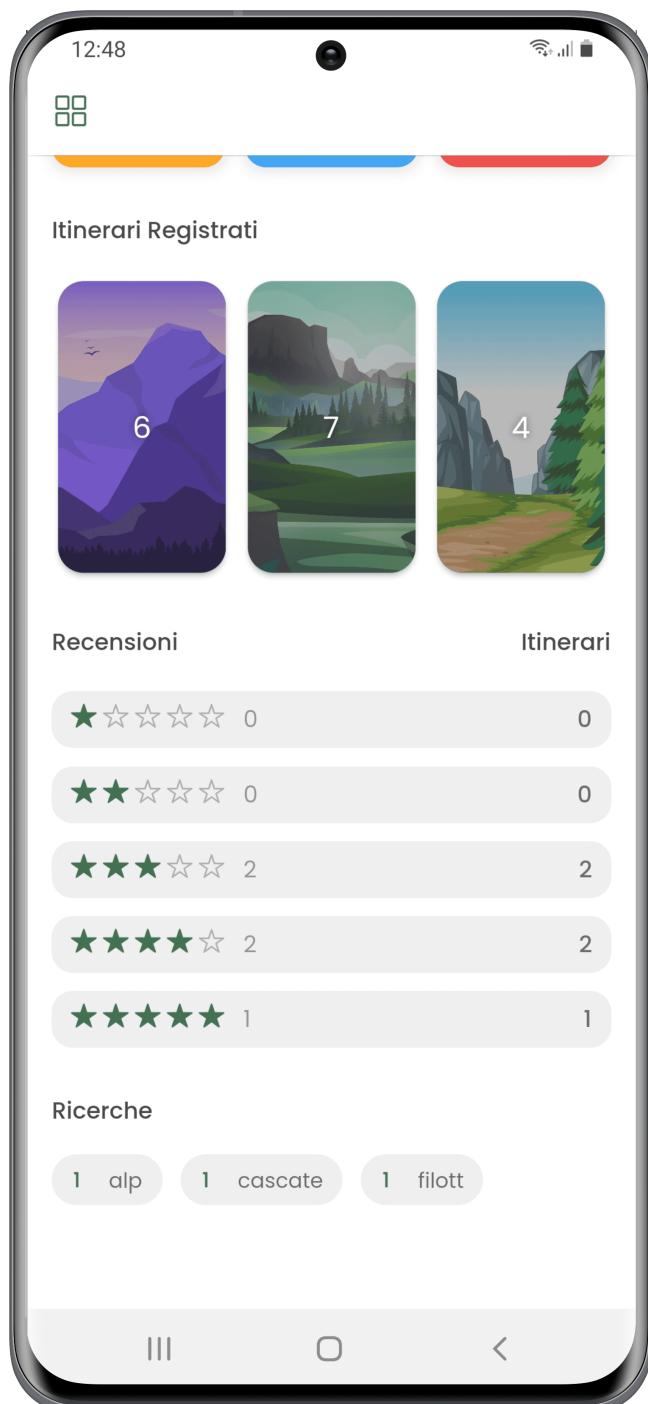
## 14.1.34 MU-A-Home



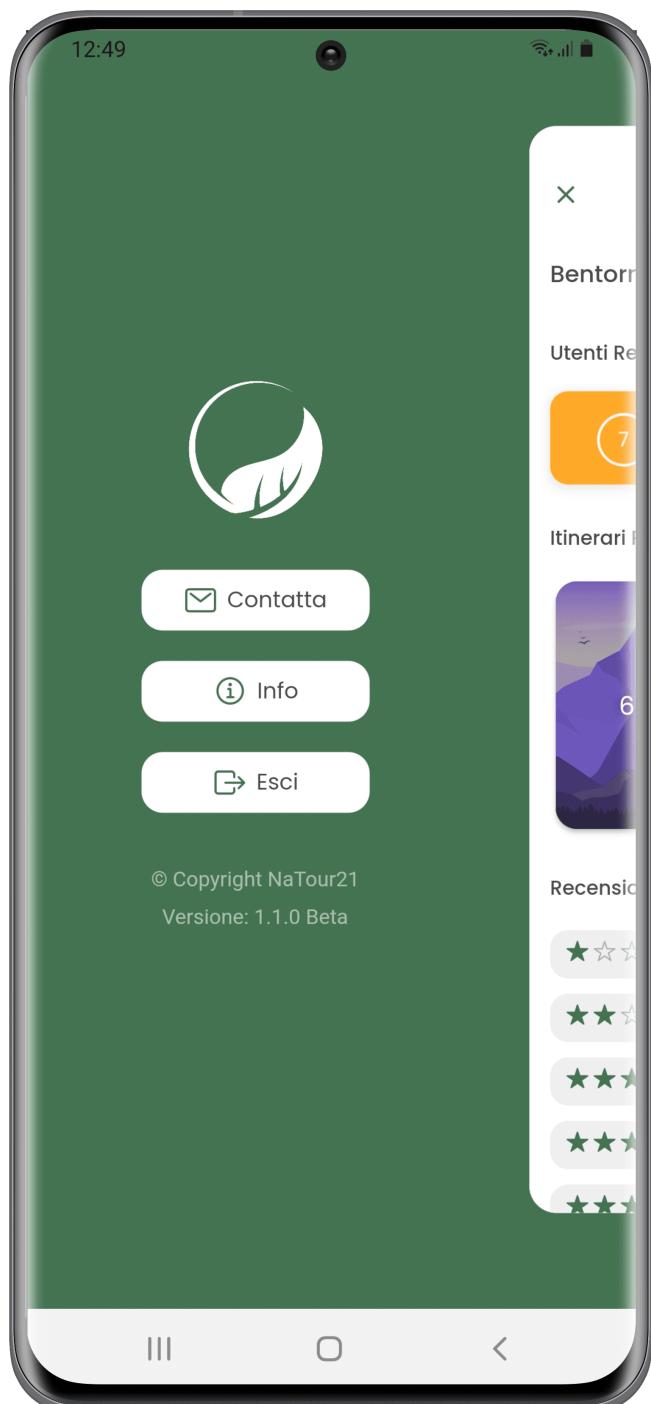
## 14.1.35 MU-A-Home2



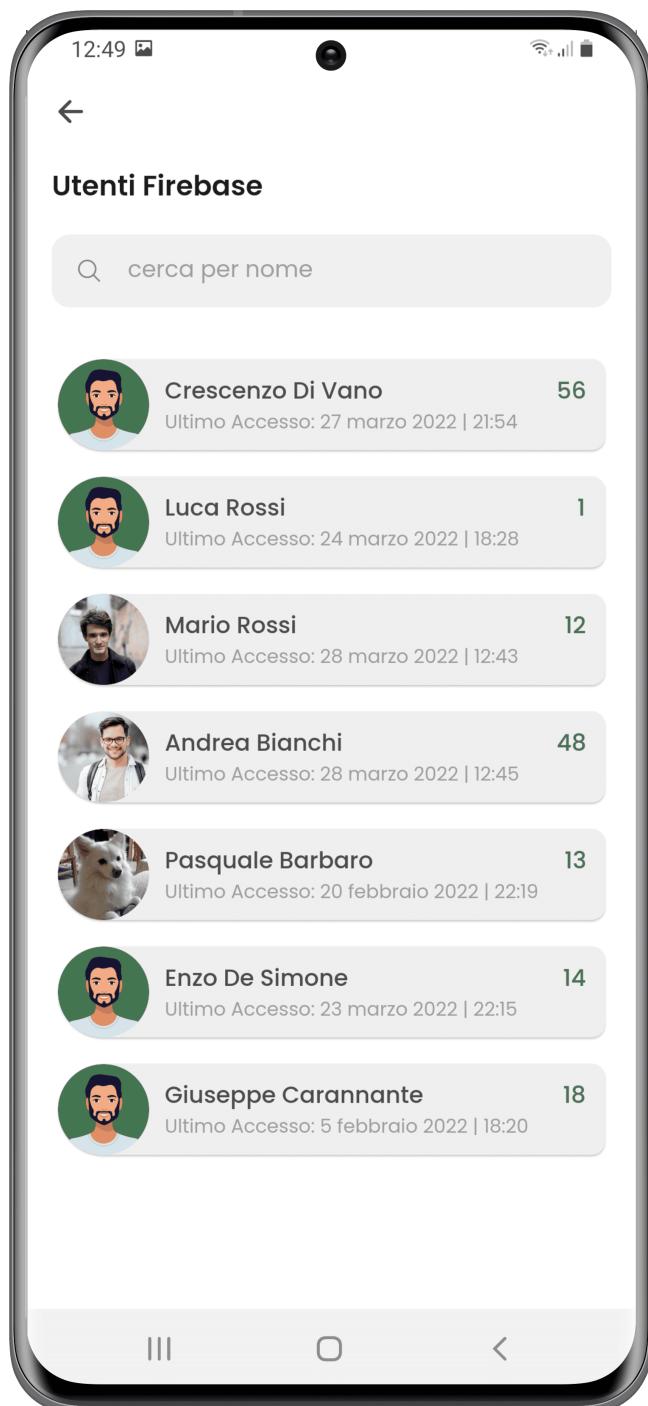
## 14.1.36 MU-A-Home3



## 14.1.37 MU-A-Drawer

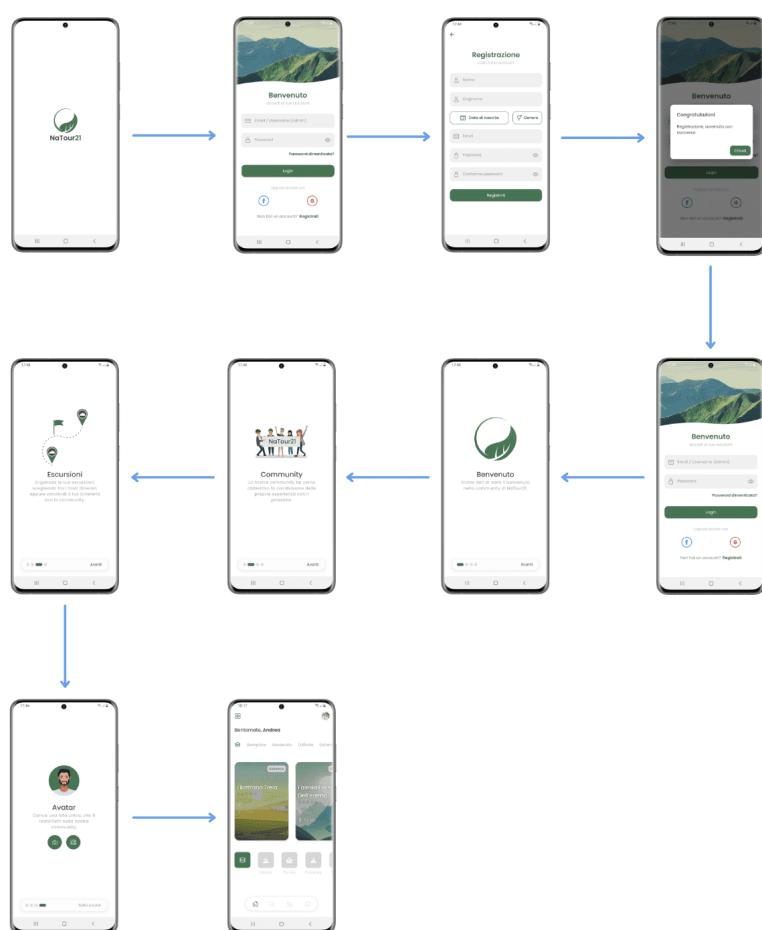


## 14.1.38 MU-A-UtentiRegistrati

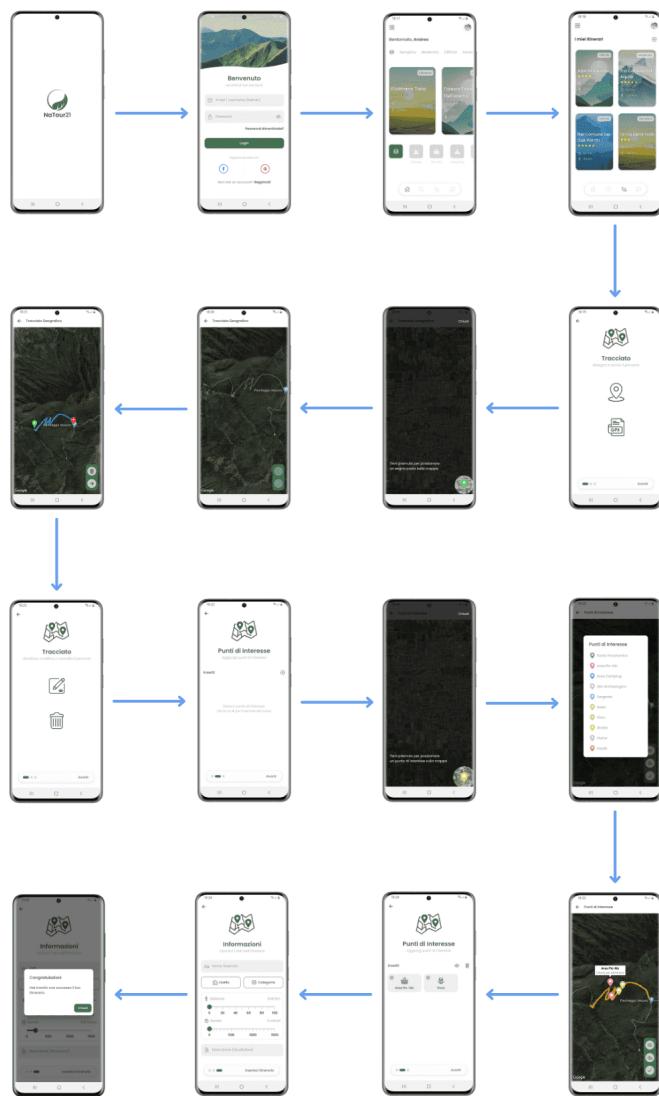


## 14.2 Interazione Schermate

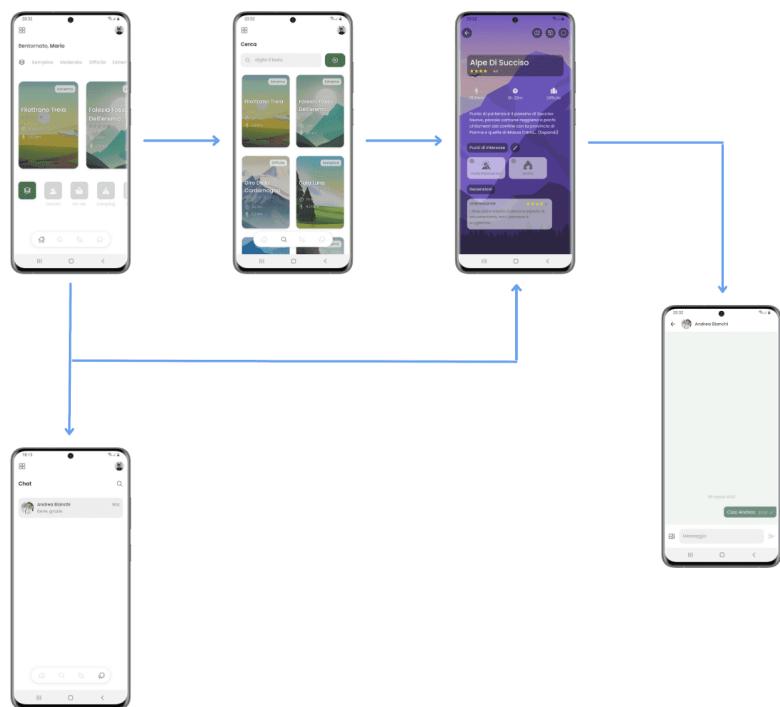
### 14.2.1 Registrazione



## 14.2.2 Inserimento Itinerario e Punti di Interesse



### 14.2.3 Invio e Lettura Messaggio



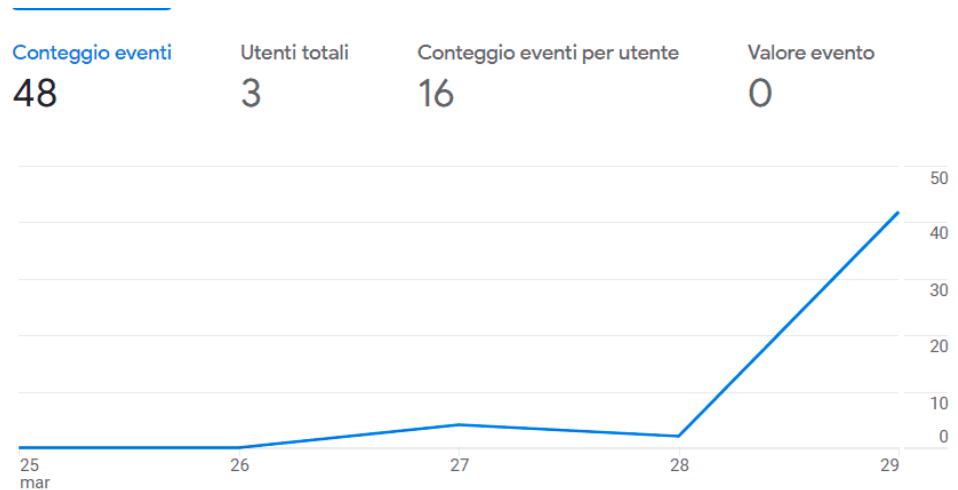
# 15. Valutazione usabilità sul campo

Succesivamente al rilascio dell'applicazione, sono state raccolte alcune statistiche relative ad alcune funzionalità dell'app, al fine di capire il comportamento dell'utente. Tutte le statistiche raccolte fanno riferimento a un periodo di tempo costante.

Di seguito riportate i grafici di Google Analytics di ogni singola statistica:

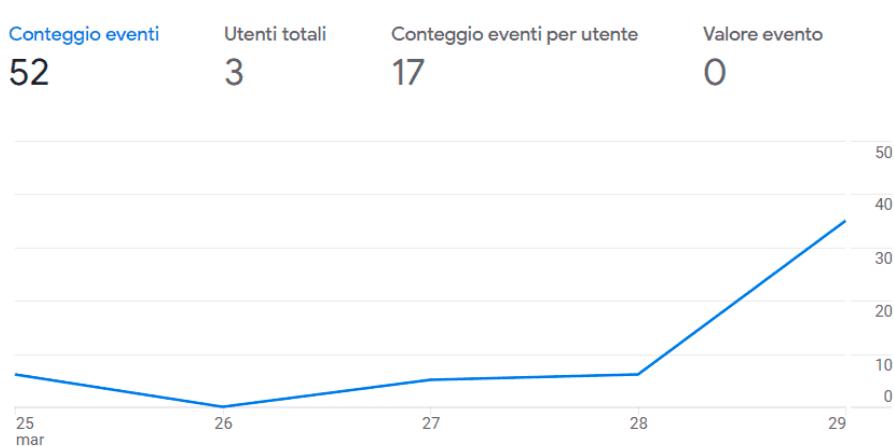
## 15.1 Inserisci Itinerario

La statistica in questione ci ha permesso di capire quante volte sono state richiamate funzioni di inserimento dell'itinerario.



## 15.2 Modifica Itinerario

La statistica in questione ci ha permesso di capire quante volte uno e più itinerario sono stati modificati.



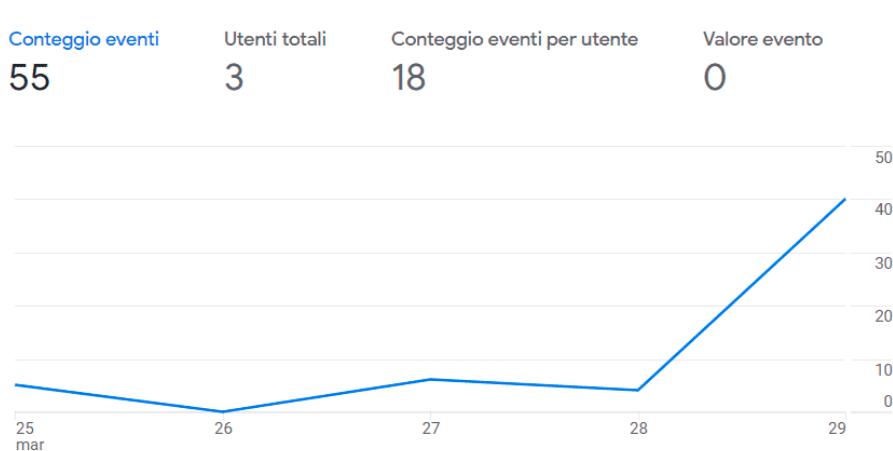
## 15.3 Modifica Punti Di Interesse

La statistica in questione ci ha permesso di capire il numero di modifiche dei punti di interesse quindi sia in aggiunta che in rimozione.



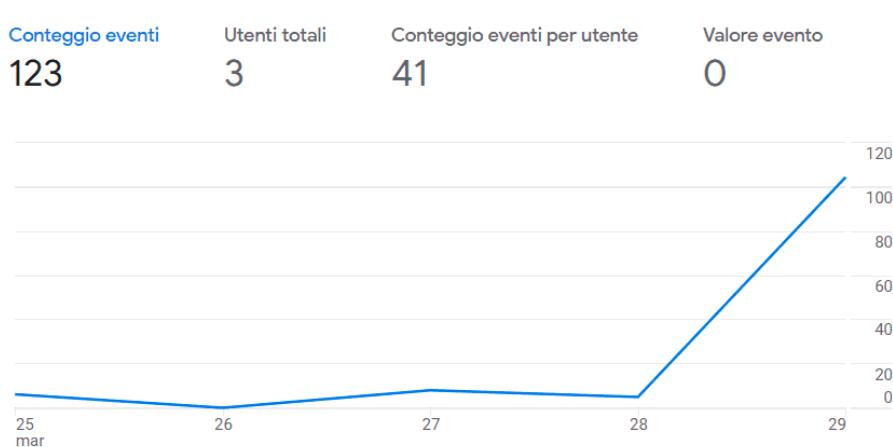
## 15.4 Esporta Dettagli Itinerario

La statistica in questione ci ha permesso di capire quante volte è stata richiesta la funzionalità di esportare i dettagli di un singolo itinerario.



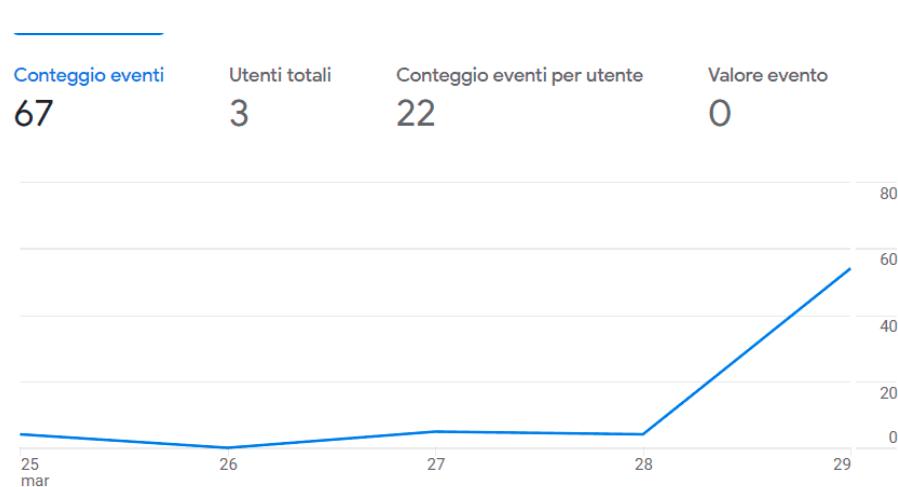
## 15.5 Esporta GPX

La statistica in questione ci ha permesso di capire quante volte è stata richiesta la funzionalità di esportare il percorso geografico tramite formato GPX.



## 15.6 Richiedi Info Itinerario

La statistica in questione ci ha permesso di capire quante volte l'utente ha cercato di avviare una conversazione col proprietario di un itinerario a fine di chiedergli delle informazioni.



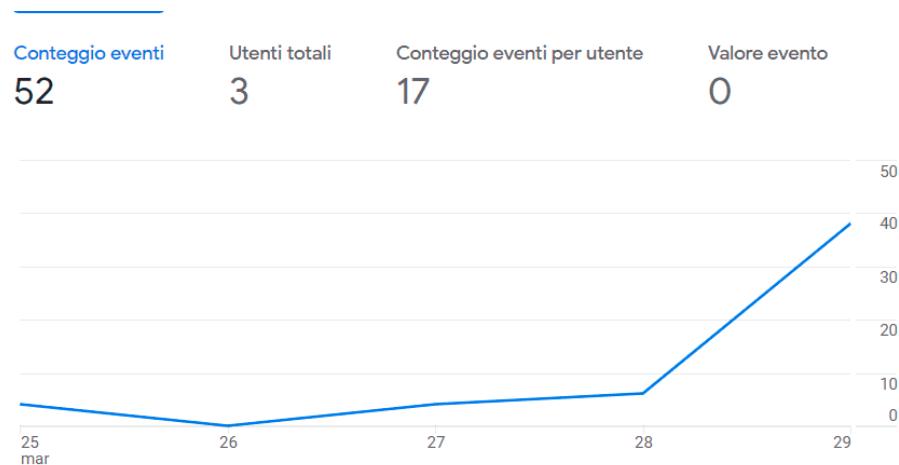
## 15.7 Invia Messaggio

La statistica in questione ci ha permesso di capire quante volte un utente che avvia una conversazione ha realmente inviato un messaggio al proprietario dell'itinerario.



## 15.8 Modifica Profilo

La statistica in questione ci ha permesso di capire quante volte un utente ha modificato i dettagli del proprio profilo utente.



## 15.9 Cerca

La statistica in questione ci ha permesso di capire quante volte un utente ha cercato qualcosa nella barra di ricerca.

