

**Social Bike Keeper**

****

Ingegneria del Software

**Caruso Sara 399000113 Prof M. Di Penta**

**Fredella Michele 399000104**

**Fucci Marianna 399000112**

Sommario

[1 Introduzione 4](#_Toc499033185)

[1.1 Problem Statement 4](#_Toc499033186)

[1.2 Scopo 4](#_Toc499033187)

[1.2.1 Definizioni, acronimi e abbreviazioni 4](#_Toc499033188)

[2 Descrizione generale 5](#_Toc499033189)

[2.1 Prospettive del prodotto 5](#_Toc499033190)

[3 Requisiti speciali e supplementari 6](#_Toc499033191)

[3.1 Interfaccia utente 6](#_Toc499033192)

[Registrazione 6](#_Toc499033193)

[Login 6](#_Toc499033194)

[Sessione d’allenamento 7](#_Toc499033195)

[Registrazione risultati allenamento 7](#_Toc499033196)

[Visualizzazione storico allenamenti 8](#_Toc499033197)

[Ricerca ciclisti da sfidare 8](#_Toc499033198)

[La sfida 8](#_Toc499033199)

[3.1.1 Caratteristiche degli utenti 9](#_Toc499033200)

[3.2 Interfaccia con le altre applicazioni 9](#_Toc499033201)

[3.3 Business Rules 9](#_Toc499033202)

[4 Requisiti specifici 10](#_Toc499033203)

[4.1 Specifiche dei casi d’uso 10](#_Toc499033204)

[5 Design architetturale 16](#_Toc499033205)

[5.1 Deployment Diagram 16](#_Toc499033206)

[5.2 Class Diagram (Client) 17](#_Toc499033207)

[5.3 Class Diagram (Server) 18](#_Toc499033208)

[5.4 Sequence Diagram 19](#_Toc499033209)

[Accesso al sistema 19](#_Toc499033210)

[Registrazione 19](#_Toc499033211)

[Acquisizione geoposizione 20](#_Toc499033212)

[Sessione d’allenamento 20](#_Toc499033213)

[La sfida – lato sfidante 21](#_Toc499033214)

[La sfida – lato sfidato 22](#_Toc499033215)

[5.5 StateMachine Diagram 23](#_Toc499033216)

[6 Testing 24](#_Toc499033217)

[6.1 Black box testing 24](#_Toc499033218)

[Registrazione 24](#_Toc499033219)

[Accesso al sistema 24](#_Toc499033220)

[Sessione d’allenamento 25](#_Toc499033221)

[Lancio di una sfida 25](#_Toc499033222)

[Challenge 26](#_Toc499033223)

[6.2 Coverage test 26](#_Toc499033224)

**Analisi e specifica dei requisiti software**

1 Introduzione

1.1 Problem Statement

L’obiettivo prefissato è quello di dare supporto agli sportivi, in particolar modo ai ciclisti amatoriali. Tale supporto verrà fornito attraverso un’applicazione mobile per il sistema operativo Android.

Gli utenti dell’applicazione sono:

* *Ciclisti amatoriali*: coloro che interagiscono con l’applicazione per:
  + Visualizzare la propria posizione su una mappa
  + Iniziare una sessione di allenamento
  + Fare analisi post-allenamento
  + Visualizzare lo storico degli allenamenti terminati
  + Visualizzare altri ciclisti sulla mappa
  + Lanciare una sfida
  + Accettare (o rifiutare) una sfida e visualizzarne il risultato

Il documento è rivolto a utenti finali, sviluppatori e software testers.

1.2 Scopo

L’applicazione mobile **Social Bike Keeper** deve fornire una molteplicità di servizi, che vanno dal semplice accesso alla rilevazione della geoposizione, alla verifica dei vincoli di svolgimento delle sfide e alla definizione e aggiornamento dello storico degli allenamenti.

1.2.1 Definizioni, acronimi e abbreviazioni

* *Social Bike Keeper*: nome dell’applicazione
* *Ciclista:* utente iscritto all’applicazione
* *Storico:* raccolta dei risultati degli allenamenti terminati
* *Sfida:* competizione sportiva tra ciclisti
* *Sfidante:* ciclista che ha lanciato una sfida
* *Sfidato:* ciclista a cui è destinata la sfida
* *Archivio:* termine generico per indicare la persistenza di informazioni su ciclisti, sfide, ecc.
* *Vicino:* altri ciclisti individuati sulla mappa
* *Credenziali:* dati inseriti nella fase di Login per accedere all’applicazione
* *Form:* interfaccia che consente all’utente di inserire i dati
* *Geoposizione:* posizione geografica del ciclista sulla mappa
* *Marker:* icona che rappresenta la posizione di un ciclista sulla mappa

2 Descrizione generale

**Social Bike Keeper** deve consentire al ciclista di:

* Accedere all’applicazione
* Consultare lo storico dei propri allenamenti
* Mostrare la propria posizione sulla mappa
* Effettuare un allenamento
* Lanciare una sfida
* Accettare o rifiutare una sfida
* Effettuare il logout dall’applicazione

2.1 Prospettive del prodotto

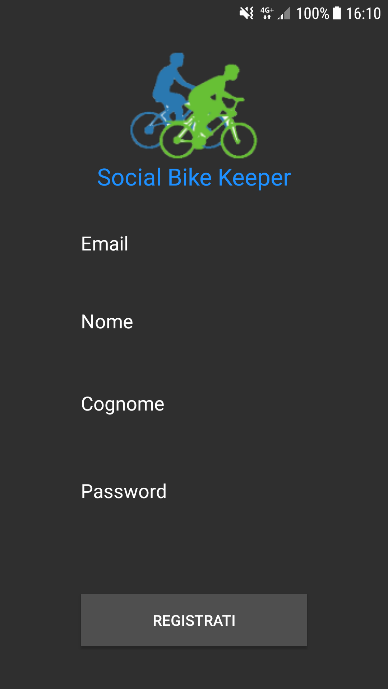
**Social Bike Keeper** è costituito da:

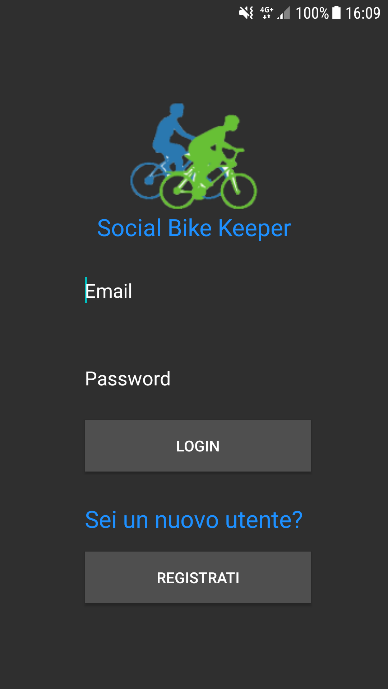
* Un dispositivo con sistema operativo Android
* Un’interfaccia grafica che mette a disposizione degli utenti le varie funzionalità dell’applicazione
* Un database dove sono memorizzate le informazioni relative ai ciclisti e alle sfide

3 Requisiti speciali e supplementari

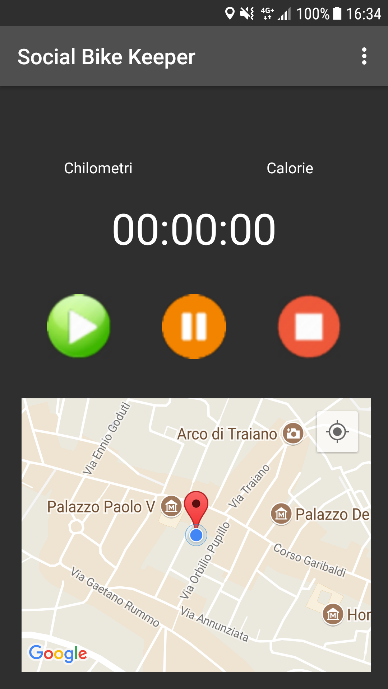
3.1 Interfaccia utente

Avviata l’applicazione, l’utente già registrato potrà effettuare il login, altrimenti potrà procedere alla registrazione.

****Registrazione: l’interfaccia consente l’inserimento di informazione personali: nome, cognome, email e password. Dopo aver confermato i dati, comparirà un messaggio di avvenuta registrazione.

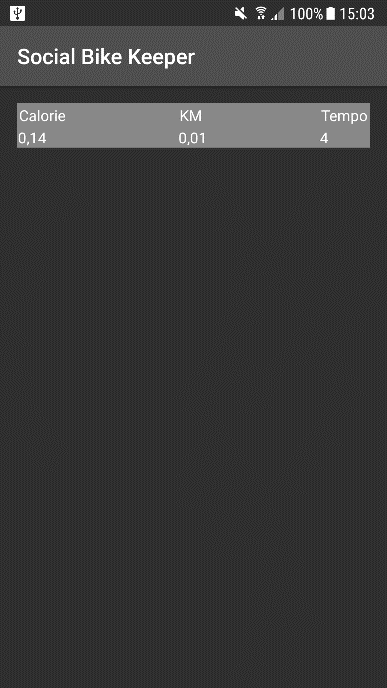
****

Login: l’interfaccia consente l’inserimento di email e password. Se le credenziali sono errate comparirà il messaggio “Login Errato”, altrimenti l’utente potrà accedere alle funzionalità previste dall’applicazione.

Sessione d’allenamento

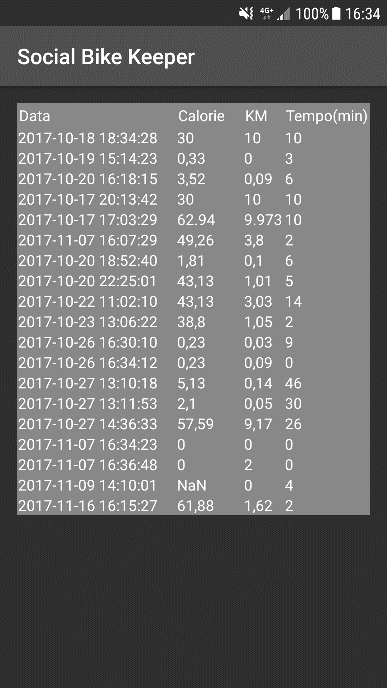
L’interfaccia consente:

* di visualizzare la propria geoposizione sulla mappa
* di iniziare una sessione di allenamento tramite la selezione del tasto “Start”
* di terminare la sessione di allenamento selezionando il tasto “Stop”
* di sospendere momentaneamente la sessione di allenamento attraverso il tasto “Pausa”
* di visualizzare in real time i chilometri percorsi e le calorie bruciate

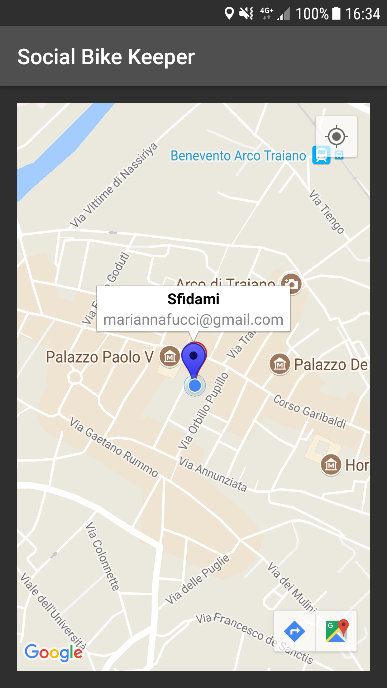


Registrazione risultati allenamento

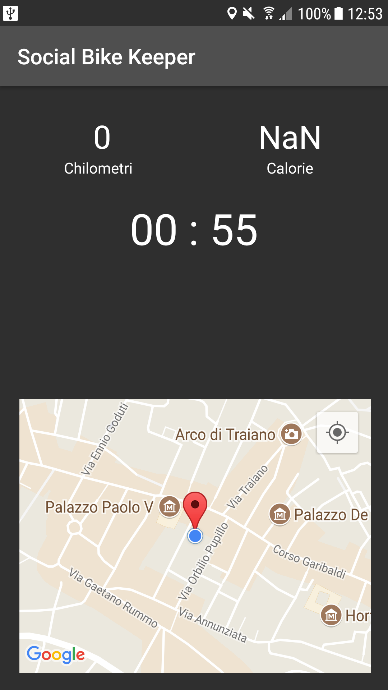
L’interfaccia consente la visualizzazione dei risultati dell’allenamento appena terminato.

Visualizzazione storico allenamenti

L’interfaccia consente la visualizzazione dei risultati di tutti gli allenamenti terminati dall’utente.

Ricerca ciclisti da sfidare

L’interfaccia mostra una mappa con i marker dei ciclisti loggati in quel momento e dà la possibilità di lanciare una sfida al ciclista scelto selezionando il marker corrispondente.

La sfida

L’interfaccia consente di visualizzare un countdown della durata scelta in fase di lancio della sfida, i chilometri percorsi, le calorie bruciate, il percorso sulla mappa e un messaggio che informa il ciclista quando la sfida è stata completata.

3.1.1 Caratteristiche degli utenti

Gli utenti destinati all’uso del prodotto Social Bike Keeper sono:

* Ciclisti dotati di un dispositivo Android che vogliono monitorare i propri allenamenti e testare le proprie abilità mettendosi alla prova con altri ciclisti.

3.2 Interfaccia con le altre applicazioni

Social Bike Keeper si interfaccia con:

* Un DBMS MySQL per gestire la persistenza

3.3 Business Rules

* Se un utente non è registrato all’applicazione non può accedere alle funzionalità fornite dall’applicazione stessa.
* Se un utente in fase di registrazione non compila tutti i campi del form o inserisce informazioni non valide come un indirizzo email senza “*@”* non potrà registrarsi all’applicazione e dovrà ripetere l’operazione.
* Se un utente prova ad accedere all’applicazione senza prima essersi registrato, non accederà ad essa.
* Se un utente registrato inserisce credenziali errate non potrà accedere all’applicazione.
* Le sfide possono durare o 60 o 90 minuti.
* Ogni sfida comprende non più di due partecipanti.

4 Requisiti specifici

4.1 Specifiche dei casi d’uso

|  |  |
| --- | --- |
| **Caso d’uso: UC-01** | **Registrazione** |
| **ATTORI** | * Ciclista |
| **INPUT** | * Email * Nome * Cognome * Password |
| **PRECONDIZIONI** | * Il ciclista non deve essere già presente nell’archivio * La connessione a Internet sia attiva |
| **OUTPUT** | * Il ciclista viene notificato della corretta registrazione |
| **POSTCONDIZIONI** | * Il ciclista è memorizzato all’interno dell’archivio |
| **SCENARIO PRIMARIO** | 1. Il ciclista inserisce le proprie credenziali: 2. Email 3. Nome 4. Cognome 5. Password 6. Il ciclista legge il messaggio di avvenuta registrazione 7. Il caso d’uso termina |
| **SCENARIO ALTERNATIVO** | **Il form non viene compilato con tutte le informazioni richieste**   1. Il ciclista legge un messaggio di errore 2. Il caso d’uso riparte dal punto 1 |
| **SCENARIO ALTERNATIVO** | **L’email è già presente in archivio**   1. Il ciclista legge un messaggio di errore 2. Il caso d’uso riparte dal punto 1 del UC-02 |
| **SCENARIO ALTERNATIVO** | **Uno o più campi del form non sono compilati correttamente**   1. Il ciclista legge un messaggio di errore 2. Il caso d’uso riparte dal punto 1 |
| **Caso d’uso: UC-02** | **Accesso al sistema** |
| **ATTORI** | * Ciclista |
| **INPUT** | * Email * Password |
| **PRECONDIZIONI** | * L’email sia presente nell’archivio dei ciclisti * La password sia associata all’email * La connessione a Internet sia attiva |
| **OUTPUT** | * Il ciclista viene notificato del corretto accesso * Il ciclista è abilitato all’utilizzo dell’applicazione |
| **POSTCONDIZIONI** | * Il ciclista è abilitato all’utilizzo del sistema |
| **SCENARIO PRIMARIO** | 1. Il ciclista inserisce le proprie credenziali: 2. Email 3. Password 4. Il ciclista accede alle funzionalità dell’applicazione 5. Il caso d’uso termina |
| **SCENARIO ALTERNATIVO** | **Il form non viene compilato con tutte le informazioni richieste**   1. Il ciclista legge un messaggio di errore 2. Il caso d’uso riparte dal punto 1. |
| **SCENARIO ALTERNATIVO** | **La password non corrisponde all’email inserita**   1. Il ciclista legge un messaggio di errore 2. Il caso d’uso riparte dal punto 1. |

|  |  |
| --- | --- |
| **Caso d’uso: UC-03** | **Acquisizione della geoposizione** |
| **ATTORI** | * Ciclista |
| **INPUT** | * Google Maps |
| **PRECONDIZIONI** | * Il ciclista abbia effettuato l’accesso all’applicazione * Il GPS sia attivo * La connessione a Internet sia attiva |
| **OUTPUT** | * Il ciclista viene notificato della sua posizione sulla mappa |
| **POSTCONDIZIONI** | * Registrazione di una nuova geoposizione all’interno dell’archivio |
| **SCENARIO PRIMARIO** | 1. Il ciclista accede all’applicazione 2. Il ciclista vede un marker relativo alla sua posizione sulla mappa 3. Il caso d’uso termina |
| **SCENARIO ALTERNATIVO** | **Il GPS non è stato attivato**   1. Il ciclista non ha attivato il GPS 2. Il ciclista viene informato che il GPS non è attivo 3. Il caso d’uso termina |

|  |  |
| --- | --- |
| **Caso d’uso: UC-04** | **Sessione di allenamento** |
| **ATTORI** | * Ciclista |
| **INPUT** |  |
| **PRECONDIZIONI** | * Il ciclista abbia effettuato l’accesso all’applicazione * Il ciclista sia stato geolocalizzato * La connessione a Internet sia attiva |
| **OUTPUT** | * Il ciclista prende visione dei chilometri percorsi e delle calorie bruciate durante l’allenamento |
| **POSTCONDIZIONI** | * Memorizzazione del risultato dell’allenamento nell’archivio |
| **SCENARIO PRIMARIO** | 1. Il ciclista accede all’applicazione (vedi **UC-02**) 2. Il ciclista viene geolocalizzato (vedi **UC-03**) 3. Il ciclista seleziona il tasto Start per iniziare l’allenamento 4. Il ciclista vede cambiare in real time i chilometri percorsi e le calorie bruciate 5. Il ciclista seleziona il tasto stop per terminare l’allenamento 6. Il ciclista vede i risultati ottenuti 7. Il caso d’uso termina |
| **SCENARIO ALTERNATIVO** | **Selezione prematura del tasto Stop**   1. Il ciclista seleziona il tasto Stop prima di selezionare lo Start 2. Il ciclista legge un messaggio di errore 3. Il caso d’uso riparte dal punto 3 |

|  |  |
| --- | --- |
| **Caso d’uso: UC-05** | **Visualizzazione storico allenamenti** |
| **ATTORI** | * Ciclista |
| **INPUT** |  |
| **PRECONDIZIONI** | * Il ciclista abbia effettuato l’accesso all’applicazione * Il ciclista sia stato geolocalizzato * La connessione a Internet sia attiva |
| **OUTPUT** | * Il ciclista prende visione di tutti i risultati degli allenamenti da lui completati |
| **POSTCONDIZIONI** |  |
| **SCENARIO PRIMARIO** | 1. Il ciclista accede all’applicazione (vedi **UC-02**) 2. Il ciclista viene geolocalizzato (vedi **UC-03**) 3. Il ciclista seleziona la voce “Storico” dal menù 4. Il ciclista prende visione di una tabella con tutti i risultati degli allenamenti svolti 5. Il caso d’uso termina |

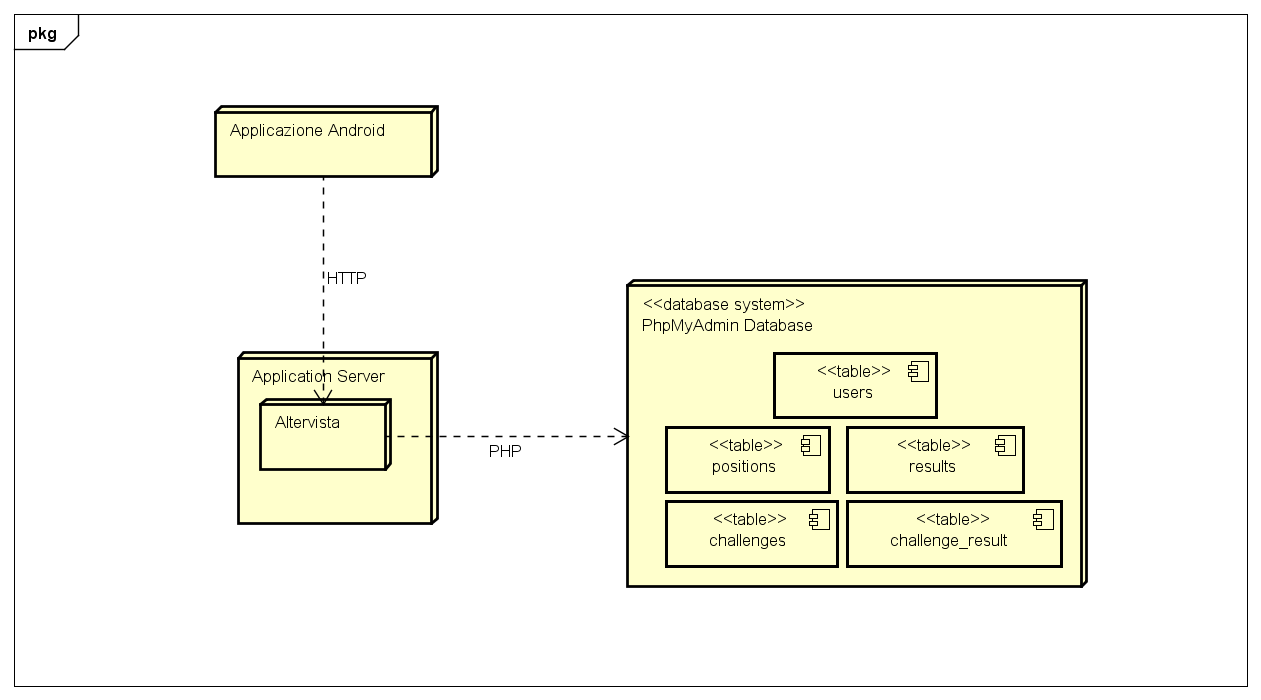
|  |  |
| --- | --- |
| **Caso d’uso: UC-06** | **Lancio di una sfida** |
| **ATTORI** | * Ciclista |
| **INPUT** |  |
| **PRECONDIZIONI** | * Il ciclista abbia effettuato l’accesso all’applicazione * Il ciclista sia stato geolocalizzato * La connessione a Internet sia attiva * È presente almeno un altro ciclista loggato |
| **OUTPUT** | * Il ciclista lancia una sfida ad un altro ciclista |
| **POSTCONDIZIONI** | * Il ciclista che lancia la sfida diventa *sfidante* * Il ciclista che accetta la sfida diventa *sfidato* * Memorizzazione della sfida nell’archivio |
| **SCENARIO PRIMARIO** | 1. Il ciclista accede all’applicazione (vedi **UC-02**) 2. Il ciclista viene geolocalizzato (vedi **UC-03**) 3. Il ciclista seleziona la voce “Sfida” dal menù 4. Il ciclista prende visione dei marker sulla mappa corrispondenti ai ciclisti loggati 5. Il ciclista sceglie un marker e seleziona “Sfidami” 6. Il ciclista sceglie la durata della sfida 7. Il ciclista riceve una notifica di accettazione 8. Il caso d’uso termina |
| **SCENARIO ALTERNATIVO** | **La sfida viene rifiutata**   1. Il ciclista riceve una notifica di rifiuto della sfida 2. Il caso d’uso termina |

|  |  |
| --- | --- |
| **Caso d’uso: UC-07** | **Accettazione di una sfida** |
| **ATTORI** | * Sfidato |
| **INPUT** |  |
| **PRECONDIZIONI** | * Lo sfidato abbia effettuato l’accesso all’applicazione * Lo sfidato sia stato geolocalizzato * La connessione a Internet sia attiva |
| **OUTPUT** | * Lo sfidato inizia la sfida |
| **POSTCONDIZIONI** | * Lo stato della sfida in archivio viene modificato |
| **SCENARIO PRIMARIO** | 1. Lo sfidato accede all’applicazione (vedi **UC-02**) 2. Lo sfidato viene geolocalizzato (vedi **UC-03**) 3. Lo sfidato riceve una notifica di una sfida lanciatagli 4. Lo sfidato accetta la sfida 5. Il caso d’uso termina |
| **SCENARIO ALTERNATIVO** | **Lo sfidato rifiuta la sfida**   1. Lo sfidato rifiuta la sfida lanciatagli 2. Il caso d’uso termina |

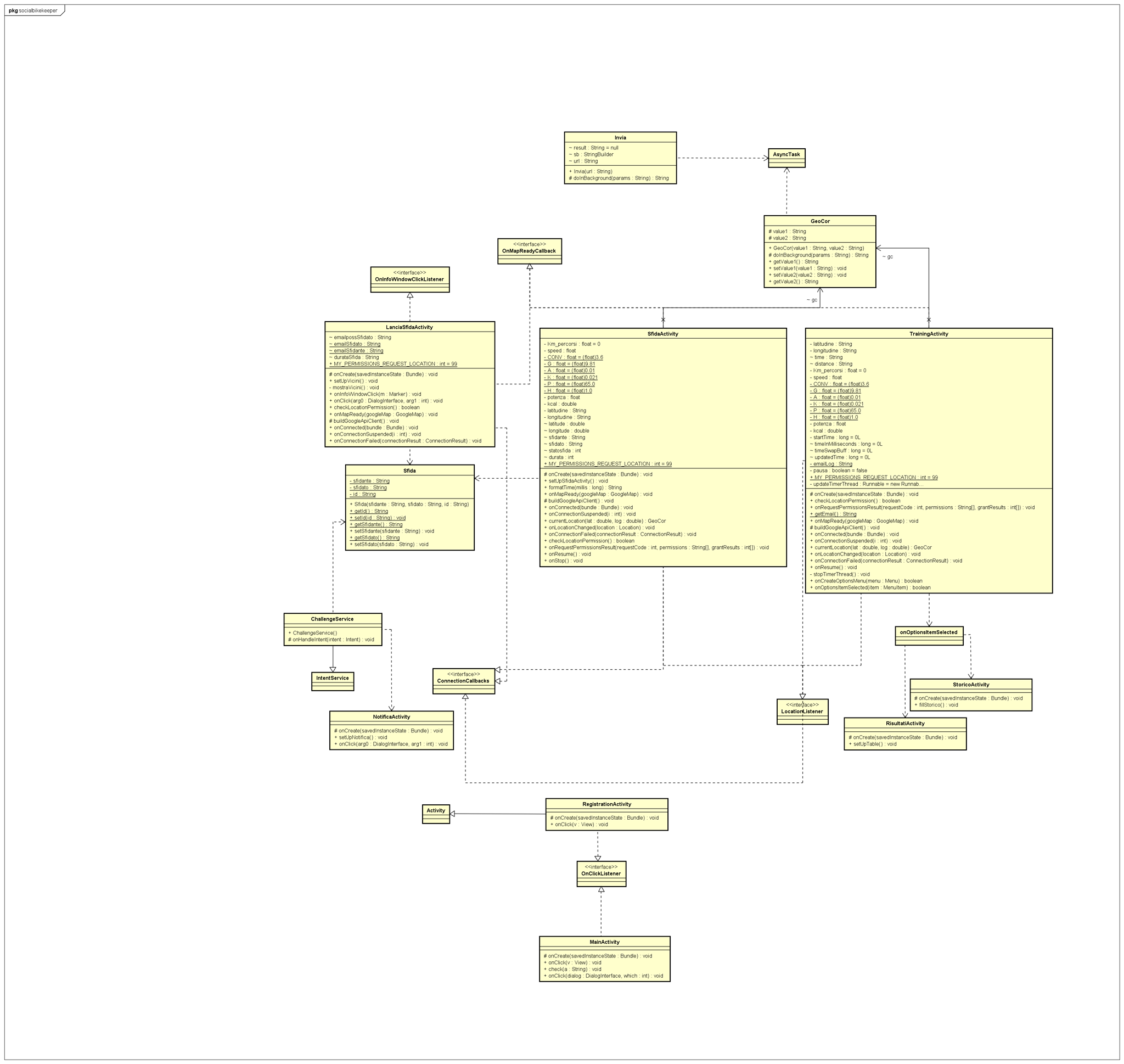
|  |  |
| --- | --- |
| **Caso d’uso: UC-08** | **La sfida** |
| **ATTORI** | * Sfidante * Sfidato |
| **INPUT** | * Lo sfidato deve accettare la sfida * Lo sfidante deve cliccare sulla notifica |
| **PRECONDIZIONI** | * Lo sfidante e lo sfidato abbiano effettuato l’accesso all’applicazione * Lo sfidante e lo sfidato siano stati geolocalizzati * La connessione a Internet sia attiva |
| **OUTPUT** | * Lo sfidante e lo sfidato ricevono il risultato della sfida |
| **POSTCONDIZIONI** | * Il risultato della sfida viene memorizzato nell’archivio |
| **SCENARIO PRIMARIO** | 1. Lo sfidante e lo sfidato accedono all’applicazione (vedi **UC-02**) 2. Lo sfidante e lo sfidato vengono geolocalizzati (vedi **UC-03**) 3. Lo sfidato inizia la sfida 4. Lo sfidante clicca sulla notifica e inizia la sfida 5. Lo sfidato termina la sfida 6. Lo sfidante termina la sfida 7. Lo sfidante e lo sfidato ricevono una notifica con l’esito della sfida 8. Il caso d’uso termina |
| **SCENARIO ALTERNATIVO** | **Lo sfidato non completa la sfida**   1. Lo sfidato non completa la sfida 2. Lo sfidante riceve una notifica di vittoria a tavolino 3. Il caso d’uso termina |
| **SCENARIO ALTERNATIVO** | **Lo sfidante non completa la sfida**   1. Lo sfidante non completa la sfida 2. Lo sfidato riceve una notifica di vittoria a tavolino 3. Il caso d’uso termina |

5 Design architetturale

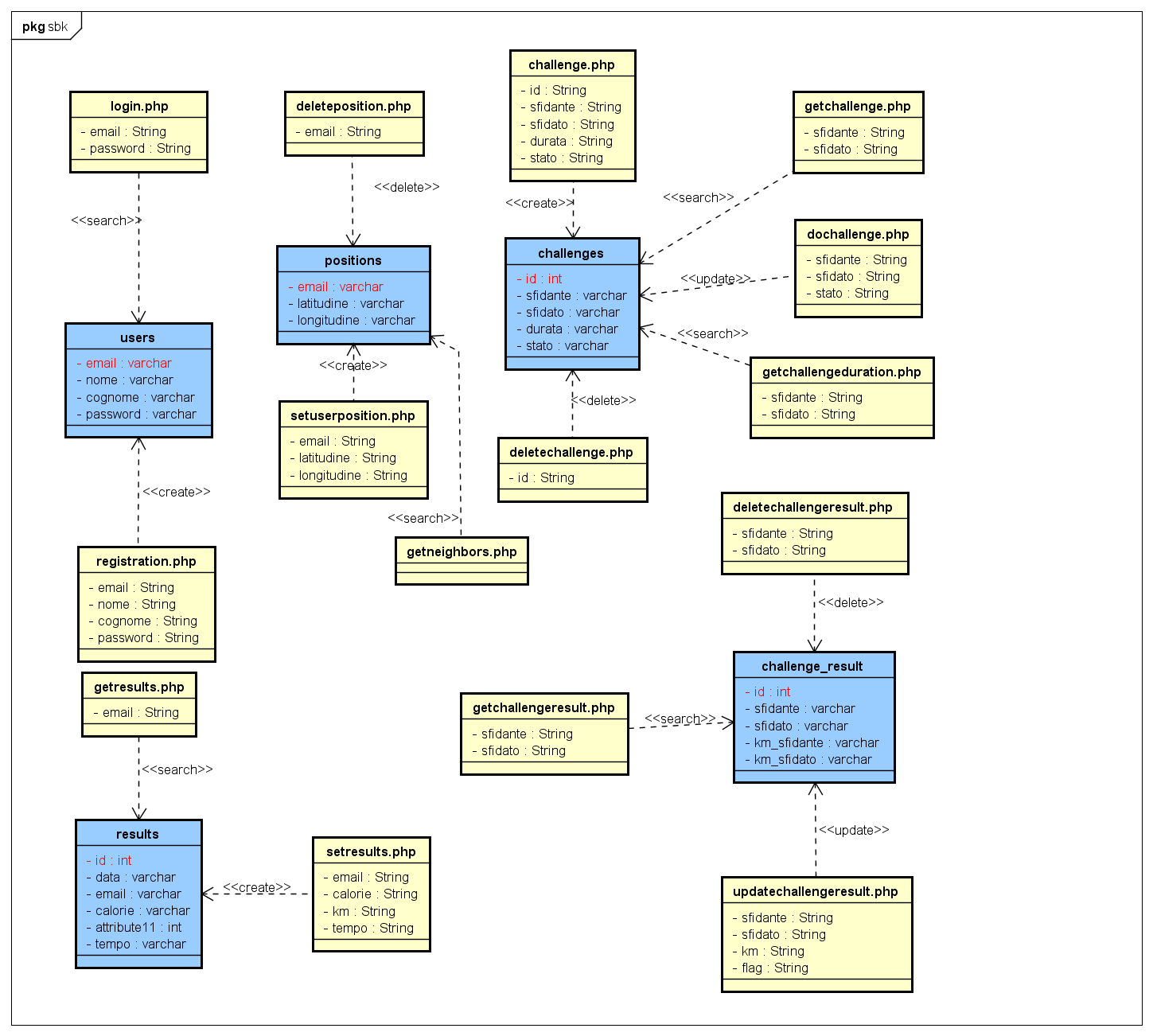
5.1 Deployment Diagram



5.2 Class Diagram (Client)

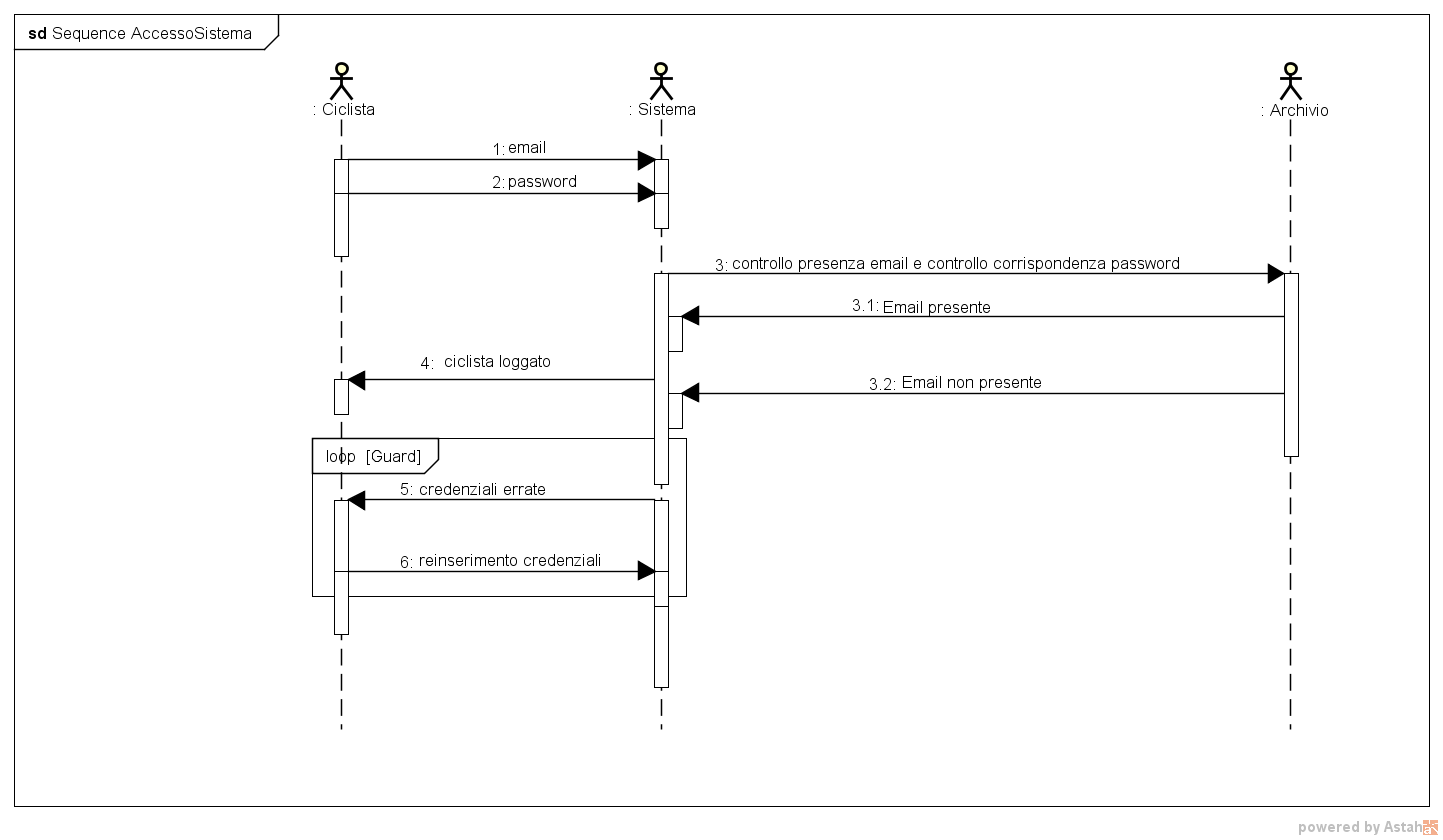


5.3 Class Diagram (Server)

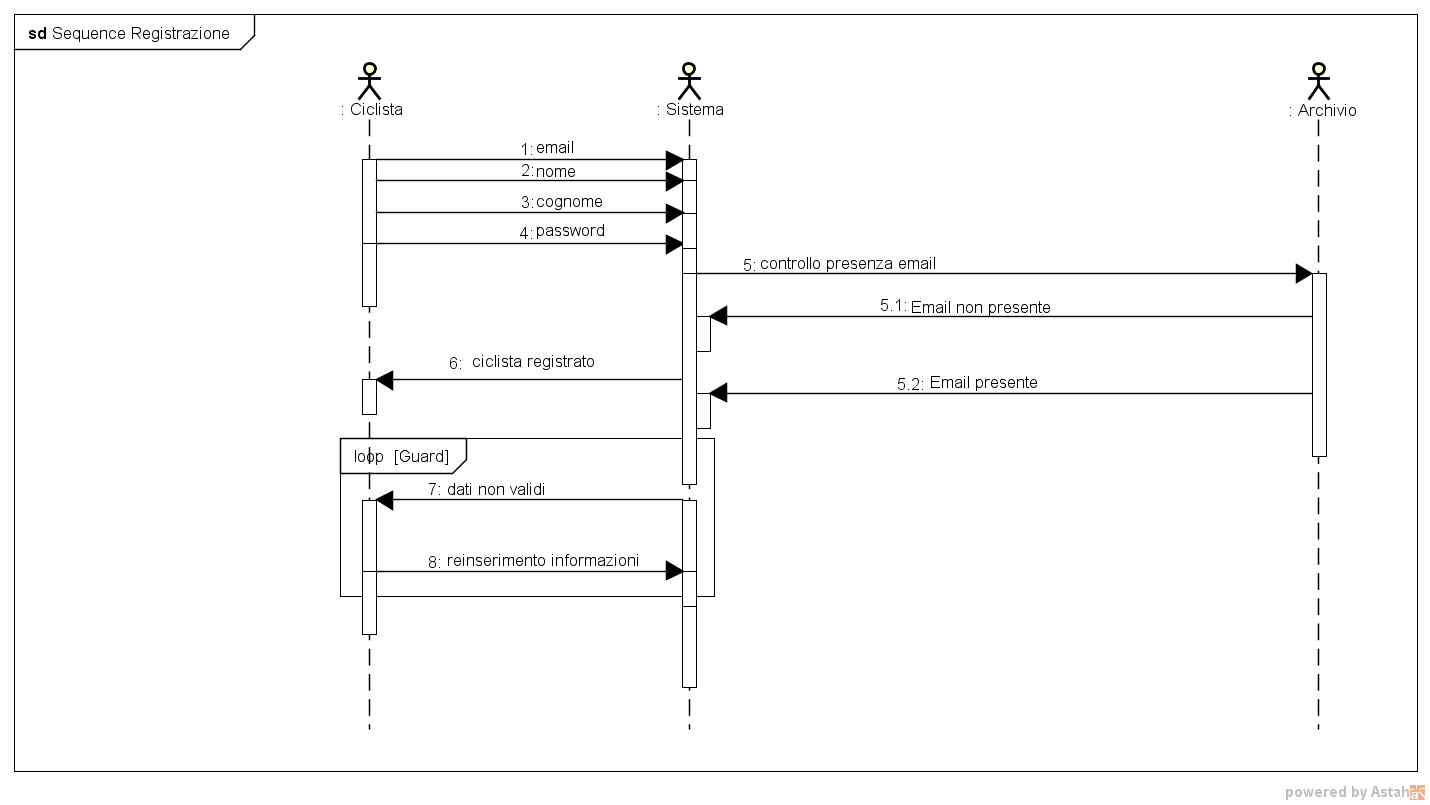


5.4 Sequence Diagram

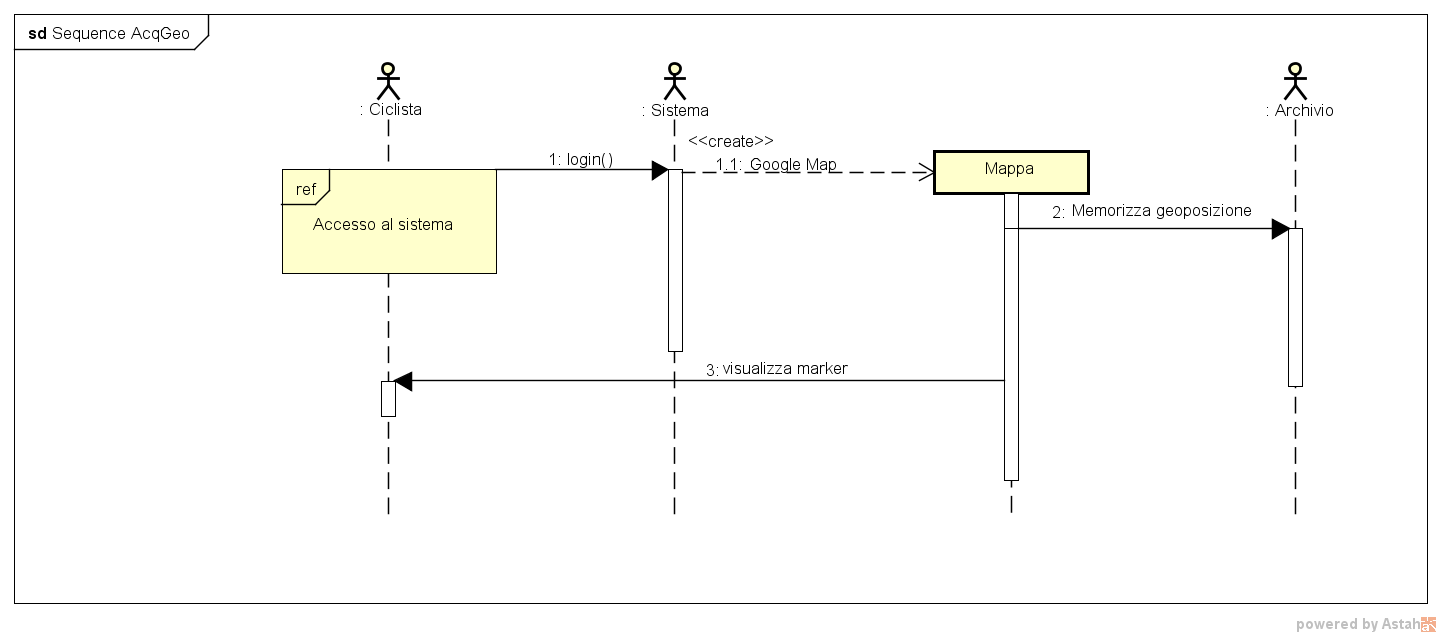
Accesso al sistema

****

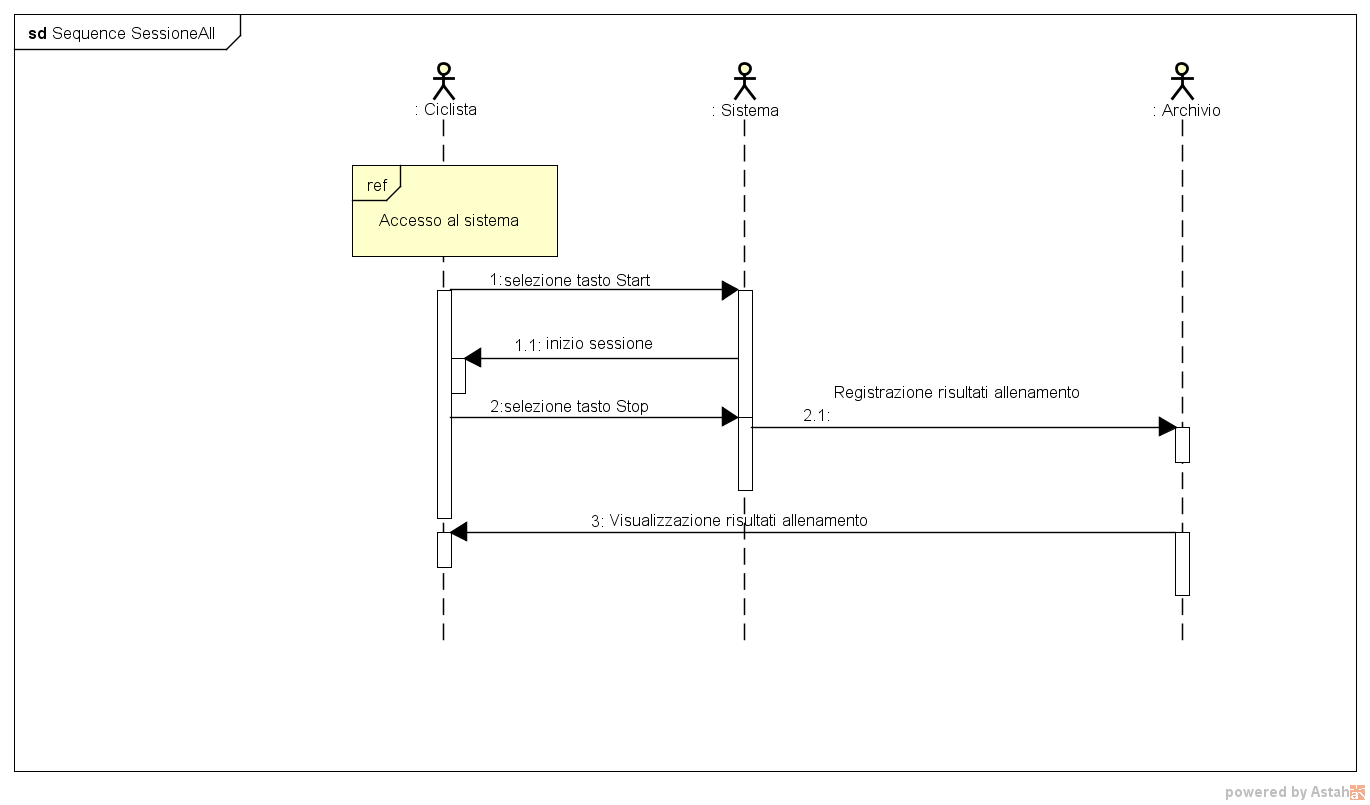
Registrazione

****

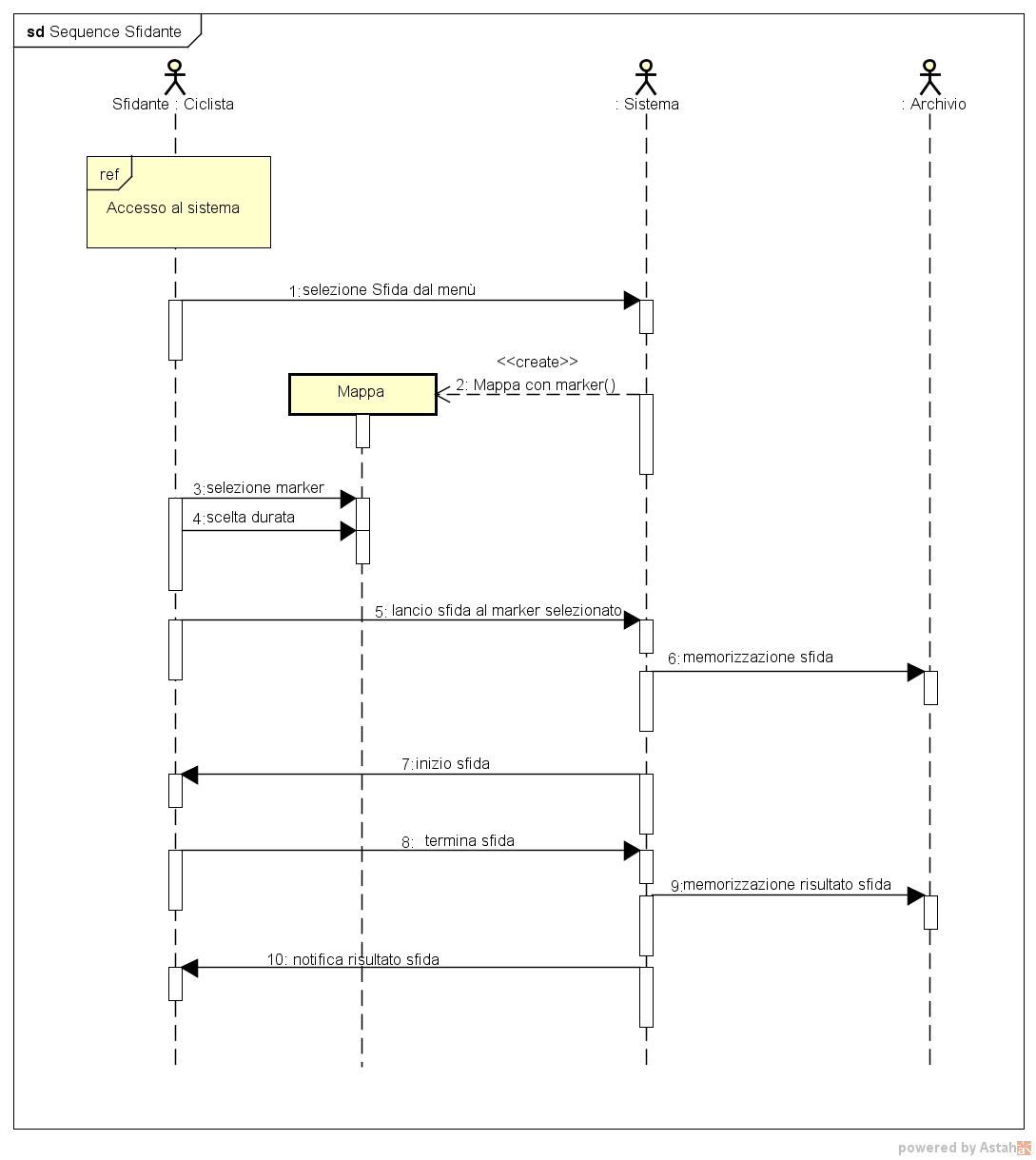
Acquisizione geoposizione

****

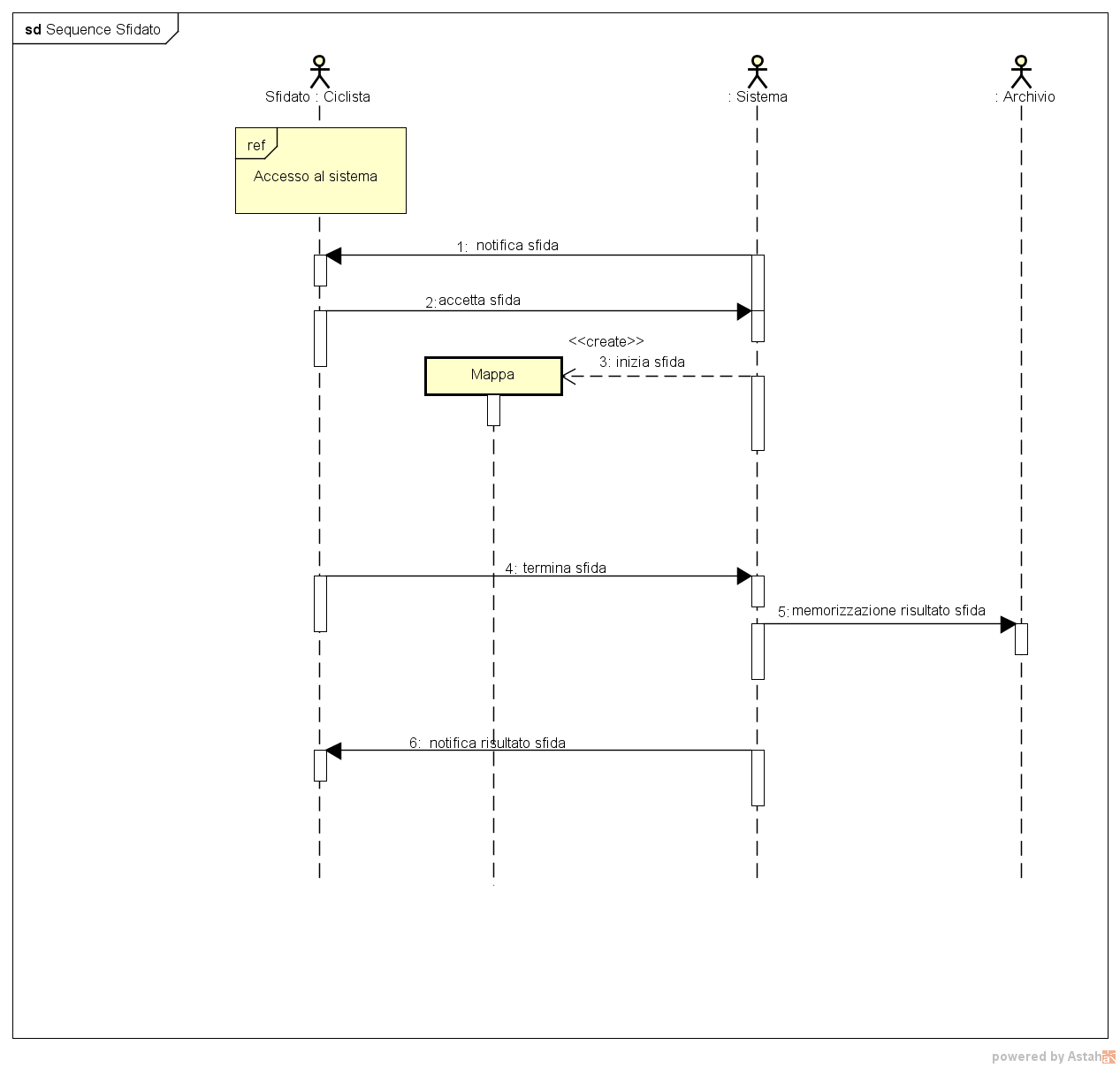
Sessione d’allenamento

****

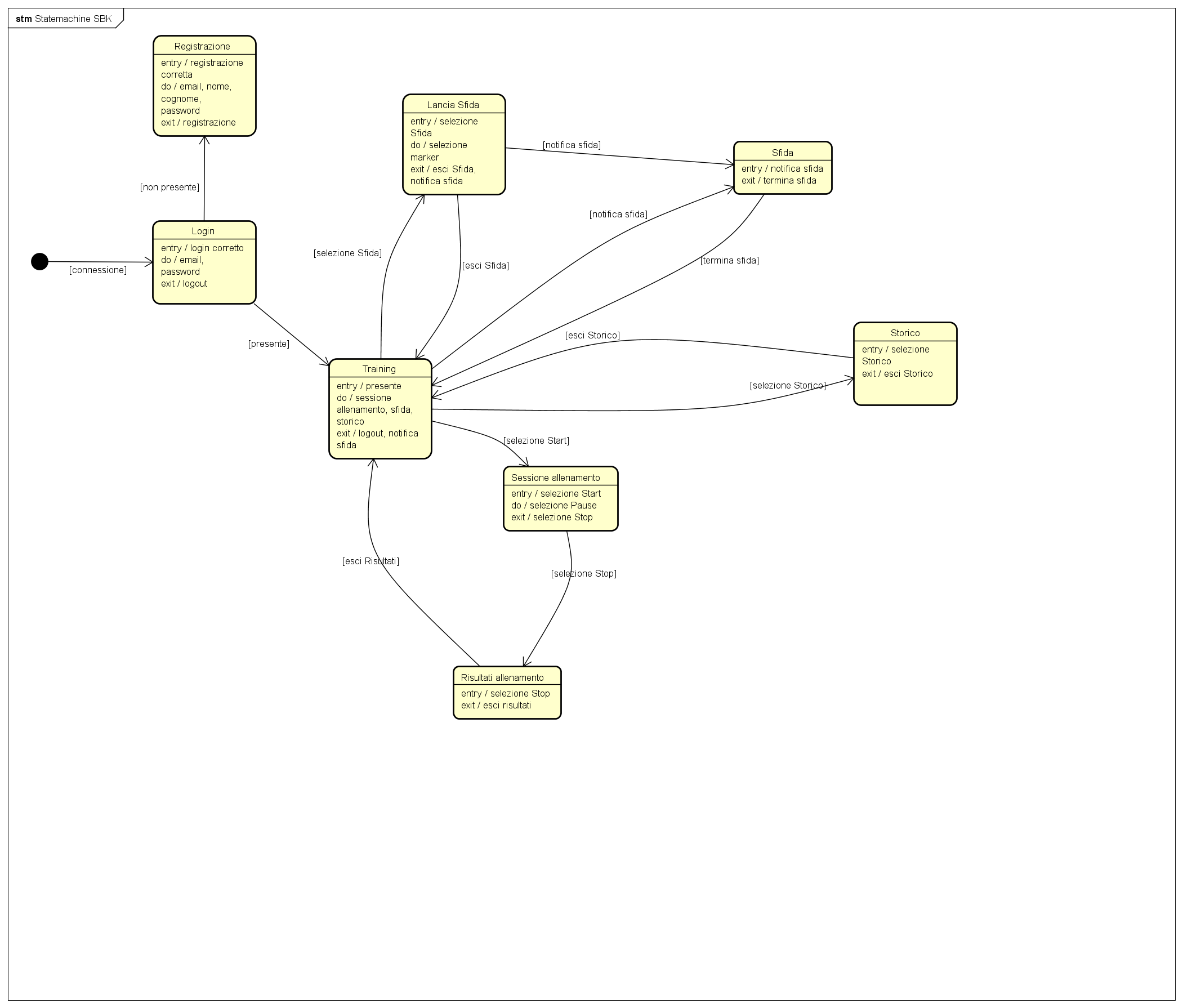
La sfida – lato sfidante



La sfida – lato sfidato



5.5 StateMachine Diagram



6 Testing

6.1 Black box testing

Il testing black box è stato effettuato creando per ogni Activity un caso di Test e simulando il comportamento dell’applicazione in risposta agli input dell’utente tramite apposite librerie quali: Espresso, UiAnimator e ActivityInstrumentationTestCase2.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Registrazione | | | | | | | |
| **Choice** | **Input parameters** | | | | **Enviroment characteristics** |  | |
|  | **Email** | **Nome** | **Cognome** | **Password** | **GPS/Network** | **Valore atteso** | **Output** |
| #1 | “[sara@gmail.com](mailto:sara@gmail.com)” | “sara” | “caruso” | “sara” | Attivo/Attivo | “Utente esistente” | “Utente esistente” |
| #2 | “[jim@gmail.com](mailto:jim@gmail.com)” | “jim” | “jim” | “jim” | Attivo/Attivo | “Registrazione avvenuta” | “Registrazione avvenuta” |
| #3 | “null@null” | “null” | “null” | “null” | Attivo/Attivo | “Dati inseriti non validi” | “Dati inseriti non validi” |
| #4 | “jim” | “” | “” | “” | Attivo/Attivo | “Dati inseriti non validi” | “Dati inseriti non validi” |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Accesso al sistema | | | | | |
| **Choice** | **Input parameters** | | **Enviroment characteristics** |  | |
|  | Email | Password | **GPS/Network** | Valore atteso | Output |
| #1 | “[sara@gmail.com](mailto:sara@gmail.com)” | “sara” | Attivo/Attivo | “Benvenuto” | “Benvenuto” |
| #2 | “” | “” | Attivo/Attivo | “ATTENZIONE login errato” | “ATTENZIONE login errato” |
| #3 | “[sara@gmail.com](mailto:sara@gmail.com)” | “aras” | Attivo/Attivo | “ATTENZIONE login errato” | “ATTENZIONE login errato” |
| #4 | “[aras@gmail.com](mailto:aras@gmail.com)” | “sara” | Attivo/Attivo | “ATTENZIONE login errato” | “ATTENZIONE login errato” |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Sessione d’allenamento | | | | | |
| **Choice** | **Input parameters** | | **Enviroment characteristics** |  | |
|  | **Operazione #1** | **Operazione #2** | **GPS/Network** | **Valore atteso** | **Output** |
| #1 | Selezione tasto Start |  | Attivo/Attivo | Parte il cronometro Valore Chilometri e Valore Calorie sono visibili | Parte il cronometro Valore Chilometri e Valore Calorie sono visibili |
| #2 | Selezione tasto Start | Selezione tasto Stop | Attivo/Attivo | Visualizza i risultati | Visualizza i risultati |
| #3 | Selezione tasto Stop |  | Attivo/Attivo | “Devi premere Start prima di iniziare l’allenamento” | “Devi premere Start prima di iniziare l’allenamento” |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Lancio di una sfida | | | | |
| **Choice** | **Input parameters** | **Enviroments characteristics** |  | |
|  | **Operazione #1** | **GPS/Network** | **Valore atteso** | **Output** |
| #1 | Selezione marker sfidante | Attivo/Attivo | Visualizza “Sei qui” | Visualizza “Sei qui” |
| #2 | Selezione marker possibile sfidato | Attivo/Attivo | Visualizza “Sfidami” | Visualizza “Sfidami” |

|  |  |  |  |
| --- | --- | --- | --- |
| Challenge | | | |
| **Choice** | **Input parameters** |  | |
|  | **Operazione #1** | **Valore atteso** | **Output** |
| #1 | Visualizza notifica | “Hai una nuova sfida di 60 minuti da:” | “Hai una nuova sfida di 60 minuti da:” |
| #2 | Selezione notifica | “Accetti la Sfida?” | “Accetti la Sfida?” |
| #3 | Seleziona “NO” (Rifiuta sfida) | Visualizza schermata per sessione d’allenamento | Visualizza schermata per sessione d’allenamento |
| #4 | Seleziona “SI” (Accetta sfida) | Visualizza schermata per la sfida | Visualizza schermata per la sfida |

6.2 Coverage test

Al fine di disporre di una valutazione quantitativa del livello di copertura dei test, è stato utilizzato il tool *Jacoco*. Esso ci ha fornito un report con i dettagli sulla percentuale delle righe di codice coperte dai test e, più precisamente, evidenzia in verde e in rosso rispettivamente le righe di codice coperte o meno.

In **figura 1** è mostrata la percentuale complessiva di codice coperto dai test relativa al package *sbk.unisannio.com.socialbikekeeper*. In **figura 2** tale percentuale viene mostrata in dettaglio per ognuna delle *Activity* implementate.

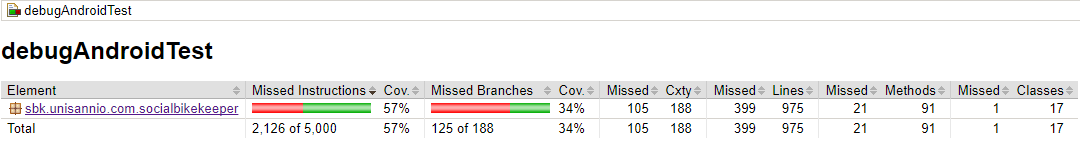


Figura 1Coverage del package

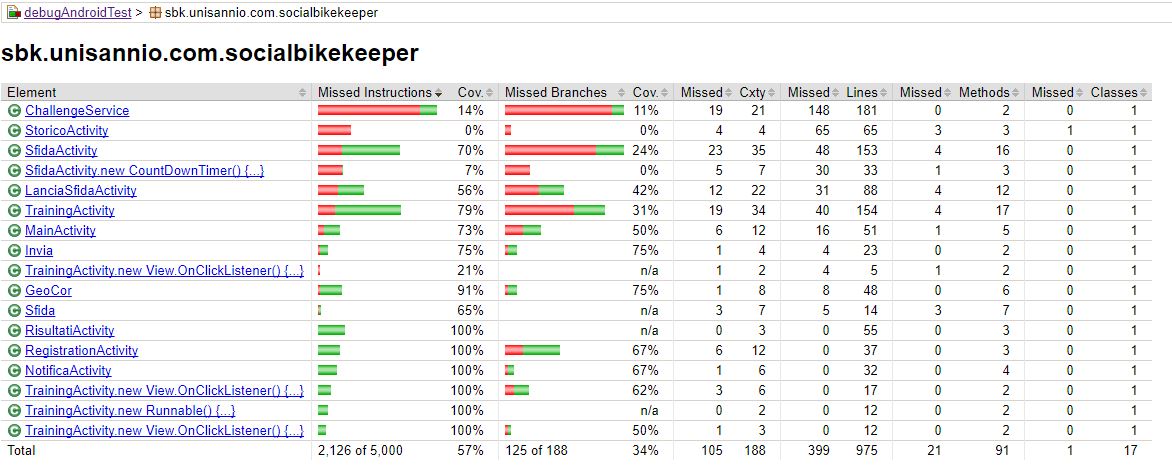


Figura 2Coverage delle activity

6.3 Analisi statica

Al fine di trovare eventuali errori all’interno del codice è stata effettuata un’analisi statica attraverso tre tool:

1. Checkstyle: trova errori di stile (vedi Figura 3)
2. Findbugs: scansiona il bytecode generato dal compilatore al fine di trovare potenziali bug (vedi Figura 4)
3. Pmd: trova imperfezioni (vedi Figura 5)

Tali strumenti hanno prodotto tre report:

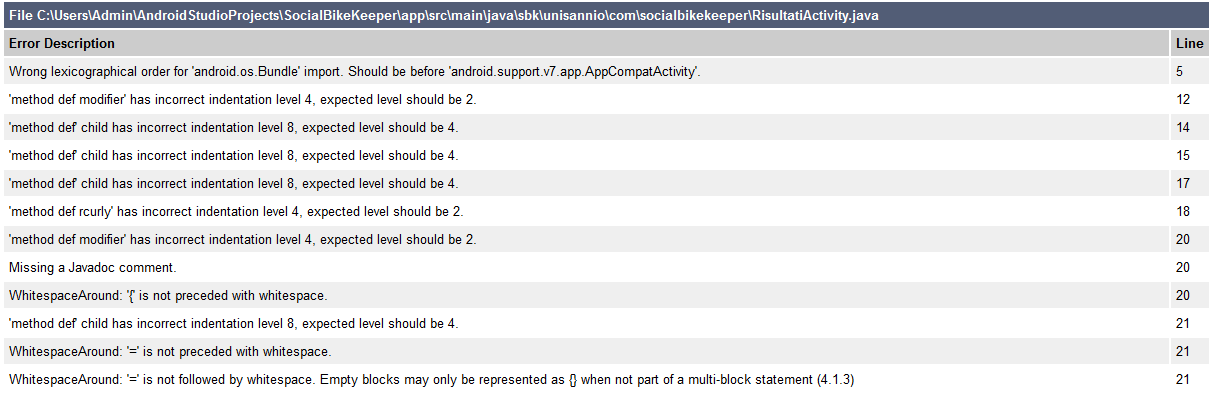


Figura 3Report Checkstyle

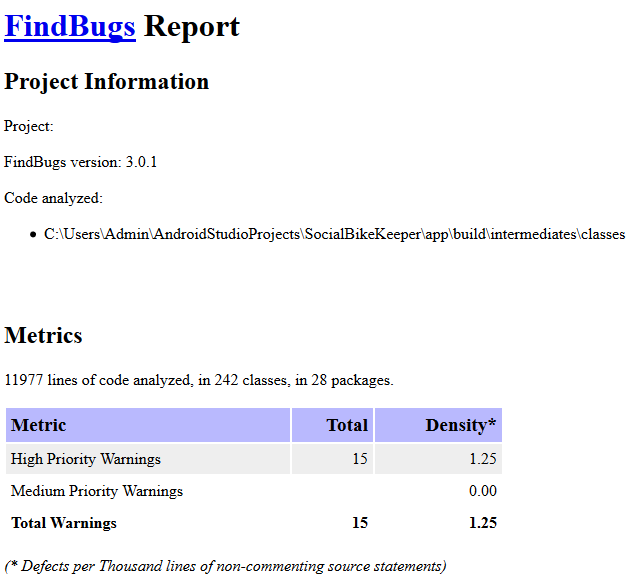


Figura 4Findbugs

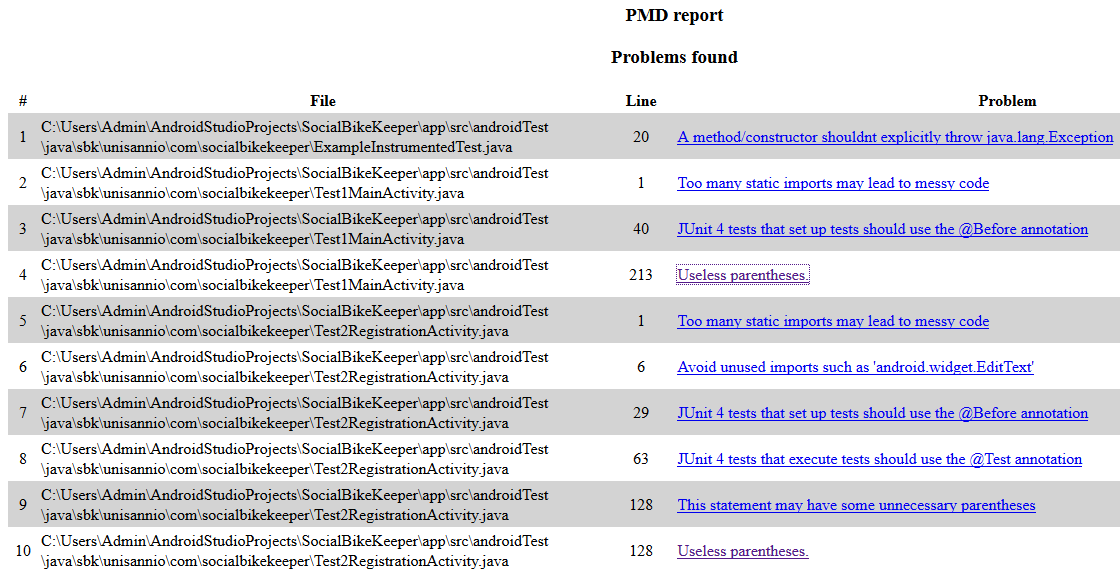


Figura 5Pmd

7 Sviluppi futuri

L’applicazione realizzata potrebbe essere ulteriormente migliorata inserendo nuove funzionalità:

* Ricerca di amici con cui pedalare insieme
* Chat per permettere ai ciclisti di comunicare tra loro
* Classifica, su base settimanale, per designare il ciclista migliore
* Group challenge: sfida tra più di due ciclisti