



CORSO DI LAUREA MAGISTRALE IN INGEGNERIA INFORMATICA E DELL'AUTOMAZIONE



IoT FOR EMERGENCY MANAGEMENT

Progetto di Ingegneria del Software

RELAZIONE

Docente:

Prof. Luca Spalazzi

Studenti:

Corrente Mattia

Giampieri Gianoberto

Mascolo Martina

Anno Accademico 2015/2016

Indice

1. INTRODUZIONE

1.1 Ambito di Progetto

2. PIANIFICAZIONE

2.1 Diagramma di Gantt

2.2 Matrice delle responsabilità

3. RACCOLTA DEI REQUISITI

3.1 Definizione delle fonti

3.2.1 Glossario

4. ANALISI DEI REQUISITI

4.1 Story Card e Task Card

4.2 Analisi orientata alle interfacce

4.2.1 Analisi CRC

4.2.2 User e Task model, Modelli di Presentazione, di Navigazione e di Dialogo

4.3 Analisi orientata ai dati

4.3.1 Diagramma delle classi

4.3.2 Diagramma dei Package

4.4 Analisi orientata ai comportamenti

4.4.1 Diagramma delle sequenze

4.4.2 Diagramma degli stati

4.4.3 Diagramma delle attività

5. PROGETTAZIONE

5.1 Progettazione dell'Architettura

5.2 Progettazione dei Componenti

5.2.1 Modello delle classi di progettazione

5.2.2 Progettazione del DataBase

5.3 Metriche

5.3.1 Coesione Interna Package - CCRC

5.3.2 Stabilità Package - S

5.3.3 Astrattezza Package - A

5.3.4 Coesione Interna - LCOM4

5.3.5 Interdipendenza - CBO

5.3.6 Complessità - WMC

6. REALIZZAZIONE

7. TEST

7.1 Login utente

7.2 Invio e aggiornamento propria posizione sul server

7.3 Funzionamento in condizioni normali

7.4 Notifica di emergenza

7.5 Guida al punto di raccolta in modalità online

7.6 Guida al punto di raccolta in modalità offline

8. MANUALE D'USO

9. MANUALE D'INSTALLAZIONE

9.1 Installazione IDE Android Studio 2.0 e importazione del progetto

9.2 Compilazione e installazione sullo smartphone

9.3 Impostazione delle indicazioni su un indossabile Android Wear (Opzionale)

1. INTRODUZIONE

1.1 Ambito di Progetto

Questo progetto ha come finalità la realizzazione di un'applicativo mobile per smartphone o tablet che permetta di gestire e aiutare le persone a compiere le giuste scelte nel processo di evacuazione in caso di emergenze ambientali quali terremoti e incendi all'interno dell'edificio dell'Università Politecnica delle Marche. In generale sfruttando il famoso algoritmo di Dijkstra, l'applicazione offre agli utenti, oltre ad una guida in condizioni di emergenza in presenza di rete WIFI e connessione con il server, anche altri due tipi di livelli di assistenza: una guida al punto di raccolta più vicino e sicuro in caso di assenza di connettività e in aggiunta una guida in condizioni normali per raggiungere una destinazione desiderata. L'utente potrà fornire la propria posizione tramite diverse modalità di immissione: tap sulla mappa, scrittura di indirizzo interno o tramite QR codes all'ingresso della stanza.

In particolare in caso di emergenza ove sia fruibile la connettività con il server, il vantaggio più significativo offerto dall'utilizzo dell'applicativo mobile risulta essere l'offerta di due livelli di interazione:

- in pre-movimento attraverso una notifica che informi la persona su cosa sta accadendo;
- in movimento per monitorare la posizione della persona in modo da prevedere gli scenari di evacuazione e l'eventuale aggiornamento di questi ultimi in caso di cambiamenti ambientali all'interno dell'edificio.

2. PIANIFICAZIONE

2.1 Diagramma di Gantt

Di seguito sono riportati due diagrammi di Gantt, il primo relativo alla fase di analisi e progettazione del software e il secondo relativo all'implementazione vera e propria.

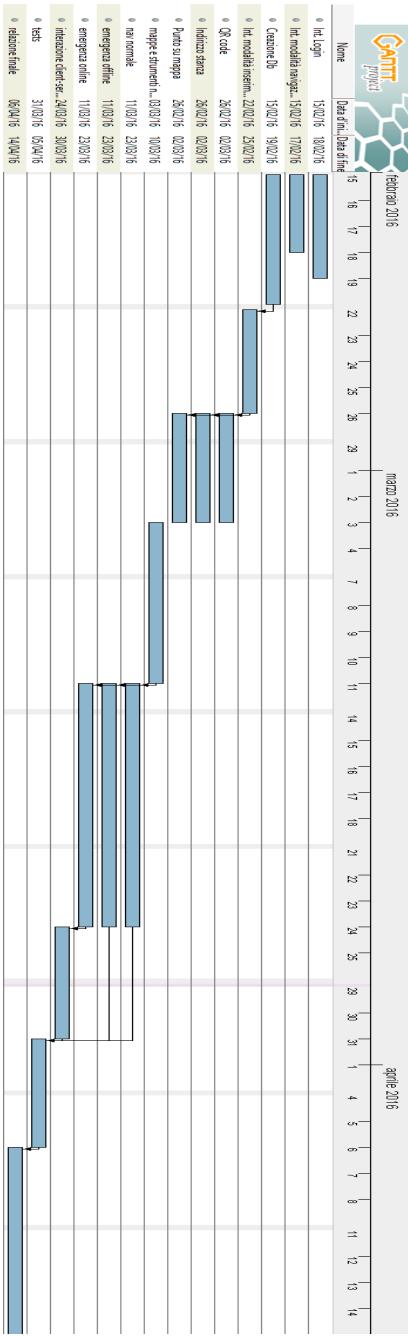


Diagramma di Gantt relativo alla codifica

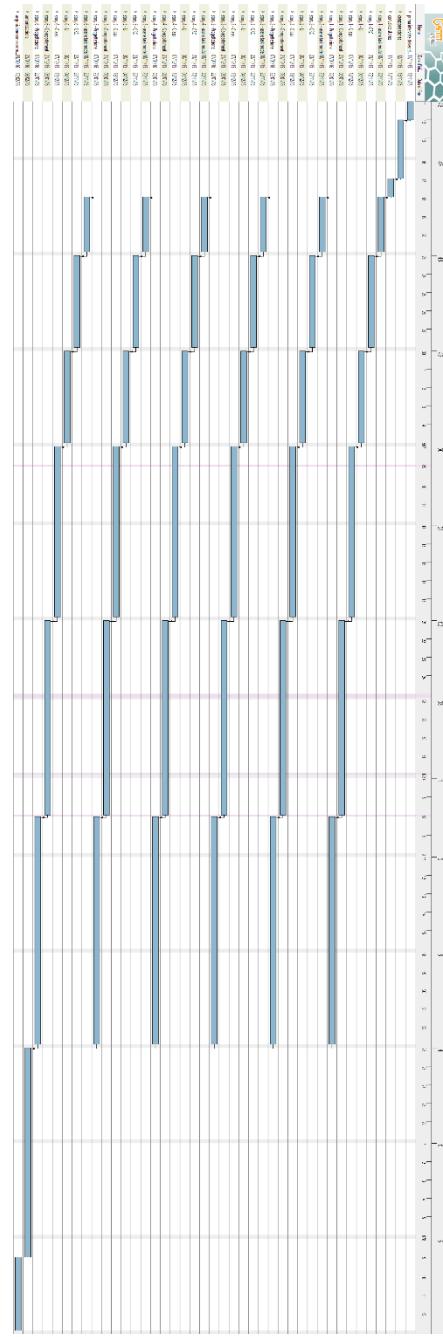


Diagramma di Gantt fasi di analisi e progettazione

2.2 Matrice delle responsabilità

La matrice delle responsabilità accompagna il diagramma di Gantt, se il secondo scansiona temporalmente le attività, la prima serve ad assegnare ciascuna attività ad uno sviluppatore. In questo modo si suddivide il carico di lavoro e si può avere una visione d'insieme del progetto.

Attività	Corrente Mattia	Giampieri Gianoberto	Mascolo Martina
prima intervista	X	X	X
creazione stories	X	X	X
convalida stories	X	X	X
Analisi e progettazione story 1		X	
Analisi e progettazione story 2			X
Analisi e progettazione story 3			X
Analisi e progettazione story 4	X		
Analisi e progettazione story 5		X	
Analisi e progettazione story 6	X		
Armonizzazione	X	X	X
Acquisizione conoscenze Android	X	X	X
Interfaccia login		X	
Modalità navigazione	X		

Creazione DB			X
Scelta modalità inserimento posizione	X		
QR Code		X	
Indirizzo stanza			X
Punto su mappa	X		
Navigazione Normale			X
Navigazione Emergenza Online	X		
Navigazione Emergenza Offline		X	
Scambi client-server		X	
Tests			X
Relazione Finale	X	X	X

3. RACCOLTA DEI REQUISITI

La raccolta dei requisiti viene suddivisa in due fasi principali: la definizione delle fonti e la raccolta dei requisiti vera e propria.

3.1 Definizione delle fonti

Le fonti principali che hanno permesso l'identificazione dei requisiti sono state:

- Gli stakeholders;
- Documento Word “*Progetti 15-16*”;
- Documento pdf “evacuazione d'emergenza: realizzazione di una app per il wayfinding all'interno di un edificio e della piattaforma server ad esso collegata”;

- Documento Word “*IoT FOR EMERGENCY MANAGEMENT*” , nel quale vengono approfonditi con maggior dettaglio l’ambito del progetto e le funzionalità richieste ;
- Documento “equazioni.pdf”, in cui si definiscono le modalità di calcolo dei percorsi alla base dell’algoritmo di Dijkstra ;
- Documento excel “UNIVPMmappa.xlsx” da cui si è successivamente costruito il file json per definire il posizionamento dei punti di controllo interni alla mappa;

In particolare gli stakeholders intervistati sono stati il prof. Luca Spalazzi e gli assistenti Gabriele Bernardini e Silvia Santarelli, che hanno fornito le linee guida e gli obiettivi per la realizzazione del progetto.

Tramite il web sono stati poi consultati, inoltre, in fase di realizzazione delle guide per l’apprendimento della programmazione Android.

Come suggerito da una metodologia agile quale XP, la raccolta dei requisiti (e successiva analisi) sono però proseguiti durante l’intero svolgimento del progetto attraverso incontri ogni due settimane (o quasi) con gli stakeholders per revisionare ciò che veniva prodotto.

Terminata la fase di raccolta dei requisiti, si è iniziata la fase di analisi.

3.2 Acquisizione e ed analisi dei requisiti

Una volta raccolti i requisiti questi sono stati classificati in due categorie: requisiti utente e requisiti sistema.

I requisiti utente si suddividono in:

- Utente: si individuano i servizi di cui ha bisogno e la modalità di utilizzo;
- Ambiente: si specificano i luoghi e le situazioni in cui l’utente utilizzerà il sistema;
- Obiettivi: si analizzano gli obiettivi e i compiti dell’utente.

I requisiti di Sistema specificano quali funzioni dovrebbe avere il sistema.

Si suddividono in :

- Dominio: si specifica il dominio operativo in cui si andrà a lavorare;

- Funzionali: si definiscono i servizi o gli effetti del sistema;
- Non funzionali: si rappresentano i vincoli sul sistema o sul processo di sviluppo;

In particolare i requisiti utente del progetto sono:

- Utente: l'utente, attraverso la mobile app, potrà fornire e aggiornare la propria posizione sulla mappa scegliendo tra diverse modalità di localizzazione tra le quali l'inserimento diretto del codice interno della stanza, la cattura del QR code e tramite tap sulla mappa. Inoltre potrà scegliere di seguire un percorso piuttosto che da un altro tra quelli consigliati per raggiungere le zone di emergenza. In caso di navigazione in condizioni ordinarie potrà preselezionare la destinazione desiderata.
- Ambiente: il sistema potrà essere utilizzato da qualsiasi utente all'interno dell'edificio universitario.
- Obiettivo: attraverso il login, la mobile app permetterà di identificare l'utente affinché il sistema possa conoscere la sua esatta posizione e inviare gli aggiornamenti. Nel caso in cui non si possa eseguire il login, l'utente avrà comunque la possibilità di accedere all'applicativo per sfruttarne tutte le funzionalità.

I Requisiti Sistema del progetto sono:

- Dominio: L'applicazione deve fornire una mappa da utilizzare per l'evacuazione di emergenza, e deve richiedere all'utente solo la sua posizione iniziale, per far sì che possa raggiungere il punto di raccolta più vicino con facilità e immediatezza.
- Requisiti funzionali:
 - La mobile app, ricevendo un alert da parte del sistema centrale (server) in presenza di connessione della rete, notifica all'utente la presenza di una situazione di emergenza. Il server essendo collegato ad un sistema di sensoristica permette di raccogliere dati quali la vulnerabilità , presenza di fumo o altri gas tossici, macerie , numero di persone, e di memorizzarle in un database; in tal modo la mobile app potrà offrire all'utente indicazioni migliori per l'evacuazione.

- L'applicazione gestisce differenti scenari di evacuazione, essa può essere usata sia online (collegando il proprio device alla rete interna o esterna) che offline; offre inoltre una modalità di navigazione in condizioni normali, nel caso l'utente voglia raggiungere una destinazione desiderata.
 - L'applicazione è in grado di calcolare il percorso migliore a costo minimo ed eventualmente un percorso alternativo , per raggiungere il luogo di raccolta in caso di emergenza o la destinazione desiderata in condizioni ordinarie, mediante l'algoritmo di Dijkstra.
 - La mobile app usa un database locale per memorizzare le informazioni relative alla mappa, inclusi nodi e archi.
 - La mobile app permette l'aggiornamento della posizione dell'utente a seguito di una richiesta dello stesso, che in caso di navigazione online verrà comunicata al server, il quale risponderà aggiornando le variabili di costo, relative agli archi della mappa, nel database locale.
 - La mobile app potrà permettere la visualizzazione dei percorsi calcolati tramite interfaccia grafica.
- Requisiti non funzionali:
- Il sistema è di tipo client-server
 - La mobile app invia dati al server
 - I dati inviati e ricevuti dai vari componenti del sistema sono in formato JSON
 - la mobile app deve essere sviluppata per piattaforma Android

3.2.1 Glossario

Termine	Descrizione	Sinonimi	Collegamenti
Utente	Chi utilizza l'app		Posizione, Server, Freccia, Indicazioni
Alert	Avviso che l'utente riceve dal server, grazie all'app	Notifica	Server, Utente, Emergenza online, Connessione, Credenziali
server	sistema informatico di elaborazione e gestione del traffico di informazioni (provenienti dal sistema sensoristico) che vengono fornite alla mobile app		Utente, Emergenza online, connessione WIFI interna
Database	insieme organizzato di dati riguardanti l'utente e la mappa		Server, utente, punti di controllo, credenziali, login,
login	procedura di accesso alla mobile app		Server, Utente, credenziali
Credenziali	Nome utente e password per l'accesso alle funzionalità dell'app	Dati di login	Utente, Server, Connessione WIFI interna, Emergenza, tasto FAB

Tasto FAB	Floating Action Button		Server, Utente, Credenziali, Bypass login
Bypass login	Possibilità di usare l'app, saltando la fase di login		Server, Utente, Tasto FAB
Indicazioni	Descrizione della direzione successiva da seguire	Notifiche su smartphone e Android Wear	Percorso migliore, percorso alternativo, Freccia, Posizione
Freccia	Indicatore della posizione dell'utente sulla mappa	Indicatore	Velocità media, Percorso, Posizione, Indicazioni
Condizioni normali	Situazione di non emergenza	condizioni ordinarie, funzionamento in condizioni normali	Destinazione, Indicazioni
destinazione	punto di destinazione scelto dall'utente in condizioni normali		utente, punti di controllo, indicazioni, percorso migliore, percorso alternativo
Guida al punto di raccolta	Situazione di emergenza		Punto di raccolta, punti di controllo, Indicazioni, freccia, Variabili di costo, Server
Punti di controllo	punti localizzati della mappa	nodo	Posizione, Utente, Percorso migliore, percorso alternativo

Punto di raccolta	Destinazione in caso di emergenza	Uscita	Posizione, Utente, Percorso migliore, percorso alternativo
Percorso migliore	primo percorso più breve (sicuro)		Utente, punti di controllo, punto di raccolta, destinazione, Percorso
Percorso alternativo	Secondo percorso più breve (sicuro)		Utente, Percorso migliore

4. ANALISI DEI REQUISITI

Il modello di processo adottato è stato l'Extreme Programming che prevede di rappresentare le situazioni in cui il sistema si può trovare all'interno di user stories. Esse rappresentano un modo per modellare i requisiti del sistema dal punto di vista dell'utente. Si è deciso di suddividere il progetto in 6 story cards, ognuna delle quali sarà costituita da diversi task (compiti) e relativi test.

4.1 Story Card e Task Card

STORY CARD 1

Release:1	Story id:1
Story tag: Autenticazione alla prima apertura	Priorità:3
Descrizione generale: l'utente deve essere in grado di autenticarsi alla prima apertura dell'applicazione.	Dettaglio: La mobile app deve permettere all'utente di accedere a tutte le funzionalità dell'applicazione dopo aver inserito le giuste credenziali. Le credenziali saranno salvate per

	<p>la successiva apertura in modo tale che l'utente rimanga loggato.</p> <p>Una volta effettuato l'accesso l'utente visualizza un'interfaccia dove poter selezionare la modalità di navigazione per successivo riconoscimento della posizione iniziale.</p> <p>E' inoltre presente un tasto FAB, che permette di bypassare l'inserimento delle credenziali nel caso in cui lo smartphone sia connesso alla rete WIFI di ateneo, ma il server sia offline.</p>
--	---

Task

Story id:1	Task id:1.1
Task tag: Schede CRC	Software Engineer: Gianoberto Giampieri
Description:	
1.Preparare tutte le schede CRC 2.Effettuare operazioni sulle schede CRC di selezione e disambiguazione	

Story id:1	Task id:1.2
Task tag: Analisi orientate alle UI	Software Engineer: Gianoberto Giampieri
Description:	
1. Selezionare schede CRC relative agli utenti 2. Vedere quali responsabilità necessitano di un'interfaccia 3. Preparare il mockup 4. Preparazione modelli di dialogo, navigazione e presentazione	

Story id:1	Task id:1.3
Task tag: Analisi orientata alle classi	Software Engineer: Gianoberto Giampieri
Description:	
1. Classificare le schede CRC in classi e attributi 2. Definire associazioni e altre relazioni 3. Disegnare diagramma delle classi	

Story id:1	Task id:1.4
Task tag: Analisi dei comportamenti	Software Engineer: Gianoberto Giampieri
Description:	
1. Preparazione diagramma delle sequenze, di stato e delle attività.	

Story id:1	Task id:1.5
Task tag: Verifica credenziali	Software Engineer: Gianoberto Giampieri
Description: L'utente deve inserire le proprie credenziali per poter fare il login. Le credenziali sono verificate e se corrette permettono l'accesso all'app e vengono salvate nel db locale se non presenti, per poter essere utilizzate al prossimo avvio. Non modellando il server, sono state stabilite le uniche credenziali "admin", "admin" tramite codice.	
Test	
Input:	
-Nome utente -Password	
Test:	
- test credenziali corrette - test credenziali errate - test credenziali salvate	
Output:	
- visualizzazione homepage dell'applicazione (sia per credenziali corrette che salvate) - messaggio autenticazione fallita	

Story id:1	Task id:1.6
Task tag: Verifica funzionamento FAB	Software Engineer: Gianoberto Giampieri
Description:	
L'utente può tappare sul tasto FAB e se è collegato alla rete WIFI di Ateneo, ma il server non fosse disponibile, allora si bypassa il controllo delle credenziali. Se l'utente è offline viene chiesto di attivare la connessione. Se il server è disponibile, allora viene chiesto di fare il login con le credenziali di Ateneo. Si è simulato il server disponibile, facendo un ping verso www.google.it.	

Test**Input:**

-Tap sul FAB

Output:

-Se connesso in WIFI, ma il server non è disponibile, si bypassa il controllo delle credenziali

-Se non connesso, si invita l'utente a collegarsi alla rete

-Se connesso, ed il server è disponibile, si invita l'utente a fare il login con le credenziali di Ateneo

STORY CARD 2

Release:1	Story id:2
Story tag: Invio e aggiornamento propria posizione sul server	Priorità:2
Descrizione generale: L'utente deve poter comunicare la propria posizione all'interno dell'edificio. La posizione verrà aggiornata sul server.	Dettaglio: L'utente può consentire alla mobile app di fargli riconoscere la propria posizione iniziale: <ul style="list-style-type: none">• fotografando un QR code all'interno della stanza• facendo un tap sulla propria posizione• scrivendo l'indirizzo interno della stanza Di conseguenza la posizione verrà aggiornata sul server. Successivamente durante la navigazione avrà la possibilità di fare tap su di un punto di controllo, all'interno del percorso selezionato o in quello alternativo, per aggiornare tempestivamente la propria posizione. Se si seleziona un nodo al di fuori dei suddetti percorsi, una finestra di dialogo chiederà all'utente se terminare la navigazione attuale o se ignorare la richiesta.

Task

Story id:2	Task id:2.1
Task tag: Schede CRC	Software Engineer: Martina Mascolo
Description: Preparare tutte le schede CRC Effettuare operazioni sulle schede CRC di selezione e disambiguazione	

Story id:2	Task id:2.2
Task tag: Analisi orientate alle UI	Software Engineer: Martina Mascolo
Description: 1. Selezionare schede CRC relative agli utenti 2. Vedere quali responsabilità necessitano di un'interfaccia 3. Preparare il mockup 4. Preparazione modelli di dialogo, navigazione e presentazione	

Story id:2	Task id:2.3
Task tag: Analisi orientata alle classi	Software Engineer: Martina Mascolo
Description: 1. Classificare le schede CRC in classi e attributi 2. Definire associazioni e altre relazioni 3. Disegnare diagramma delle classi	

Story id:2	Task id:2.4
Task tag: Analisi dei comportamenti	Software Engineer: Martina Mascolo
Description: 1. Preparazione diagramma delle sequenze, di stato e delle attività	

Story id:2	Task id:2.5
Task tag: Verifica funzionamento pulsanti per inserimento posizione	Software Engineer: Martina Mascolo

Description:

L'utente può stabilire la propria posizione attraverso tre modalità, che devono controllare se la posizione richiesta è effettivamente un punto di controllo del grafo.

Test

Input: Fotografa QR CODE

Output: Apertura fotocamera e corretta acquisizione qr code, o messaggio di errore se il nome della stanza non è presente nel db;

Input: Inserisci indirizzo interno della stanza;

Output: Apertura box di testo e test sulla presenza o meno del codice interno della stanza, altrimenti invita ad inserire un codice corretto.

Input: Tap sulla mappa

Output: Apertura carousel in cui selezionare il piano e apertura della mappa corrispondente. Successivamente test sulla corretta acquisizione della posizione tramite tap diretto sulla mappa.

Story id:2	Task id:2.6
Task tag: Verifica corretto aggiornamento della posizione sul Server	Software Engineer: Martina Mascolo
Description: L'utente, aggiornando la propria posizione con un tap sulla mappa, la comunica al Server, se disponibile. Il Server riceve la posizione iniziale, e le successive posizioni che l'utente decide di aggiornare. Per i test è stato simulato un Server all'indirizzo localhost:8080, per poter visualizzare i dati acquisiti.	
Test	
Input: Tap sulla mappa	
Output: Lettura sul Server della posizione aggiornata	

STORY CARD 3

Release:1	Story id:3
Story tag: Funzionamento in condizioni normali	Priorità:2
Descrizione generale: L'utente può utilizzare la mobile app	Dettaglio: Nota la posizione iniziale, l'utente seleziona o

anche in condizioni di non emergenza preselezionando la destinazione desiderata.	inserisce la destinazione desiderata. Nel caso la mobile app preveda due percorsi alternativi l'utente può visualizzarli entrambi sulla mappa. Dopo aver selezionato il percorso che ritiene migliore, verrà guidato mediante indicazioni direzionali, sulla base dell'orientamento dell'utente sul piano reale, e per quanto riguarda la presenza di scale, verso la destinazione impostata. Tali indicazioni sono richiamate all'utente anche sotto forma di notifica sullo smartphone tramite vibrazione dello stesso, e anche su un eventuale dispositivo Android Wear dell'utente. In aggiunta l'utente potrà sempre scegliere di selezionare il percorso alternativo. Inoltre potrà aggiornare la propria posizione durante il percorso scelto selezionando, al passaggio , punti di controllo intermedi (appartenenti al percorso) permettendo di ricevere informazioni aggiuntive. In tal caso non è contemplata la presenza di connessione e quindi l'eventuale aggiornamento delle variabili di costo.
--	--

Task

Story id:3	Task id:3.1
Task tag: Schede CRC	Software Engineer: Martina Mascolo
Description: Preparare tutte le schede CRC Effettuare operazioni sulle schede CRC di selezione e disambiguazione	

Story id:3	Task id:3.2
Task tag: Analisi orientate alle UI	Software Engineer: Martina Mascolo
Description: 1. Selezionare schede CRC relative agli utenti 2. Vedere quali responsabilità necessitano di un'interfaccia	

- | |
|---|
| 3. Preparare il mockup
4. Preparazione modelli di dialogo, navigazione e presentazione |
|---|

Story id:3	Task id:3.3
Task tag: Analisi orientata alle classi	Software Engineer: Martina Mascolo

Description:

- 1. Classificare le schede CRC in classi e attributi
- 2. Definire associazioni e altre relazioni
- 3. Disegnare diagramma delle classi

Story id:3	Task id:3.4
Task tag: Analisi dei comportamenti	Software Engineer: Martina Mascolo

Description:

- 1. Preparazione diagramma delle sequenze, di stato e delle attività.

Story id:3	Task id:3.5
Task tag: Controllo sulla destinazione	Software Engineer: Martina Mascolo

Description:

L'utente può scegliere la destinazione desiderata attraverso due modalità, che devono controllare oltre che l'esistenza nel grafo, anche la non coincidenza tra destinazione e partenza.

Test

Input: Inserimento indirizzo stanza per autocompletamento;

Output: Se l'indirizzo di destinazione è corretto, ed esso non coincide con la partenza, vengono tracciati, nel caso esistano, i due percorsi alternativi dalla posizione di partenza a quella desiderata, altrimenti si invita ad inserire un codice esatto;

Input: Inserimento tramite tap su di un punto di controllo nella mappa

Output:

- 1. Apertura del carousel in cui selezionare il piano e apertura della mappa corrispondente. Successivamente test sulla corretta acquisizione della posizione

tramite tap diretto sulla mappa, con conseguente visualizzazione dei percorsi, se il nodo di destinazione non coincide con quello di partenza.

2. Se l'indirizzo di destinazione non è corretto, viene richiesto ogni volta di riprovare a fare tap su di un punto di controllo interno alla mappa del piano precedentemente selezionato.

Story id:3	Task id:3.6
Task tag: Controllo avvio navigazione e indicazioni	Software Engineer: Martina Mascolo
Description: Una volta visualizzati i percorsi all'interno della mappa, l'utente ne può scegliere uno per iniziare la navigazione	
Test	
Input: Tap su un percorso Output: Compare una freccia verde che identifica l'utente, che inizia a muoversi nella direzione corretta. Compaiono inoltre le indicazioni da seguire, sia sulla mappa, sia sotto forma di notifica sullo smartphone e su Android Wear precedentemente configurato.	

Story id:3	Task id:3.7
Task tag: Controllo sui punti intermedi	Software Engineer: Martina Mascolo
Description: L'utente, durante la navigazione, può toccare in qualsiasi punto dello schermo. L'app deve riconoscere tale punto e stabilire l'operazione da effettuare.	
Test	
Input: -Selezione tramite tap sulla mappa di un punto di controllo, durante la navigazione. Output: <ol style="list-style-type: none"> 1. Se il punto di controllo selezionato appartiene al tracciato scelto, la freccia durante la navigazione si riposizionerà in quel punto e riproseglierà sulla base della velocità media presunta. Se il punto di controllo appartiene al percorso alternativo, l'indicatore verrà posizionato sul nodo scelto, e la navigazione si sposterà sul nuovo percorso. Verranno inoltre visualizzate informazioni ulteriori 	

- (direzioni) per aiutare la navigazione;
2. Se il punto selezionato non appartiene ai percorsi consigliati e non è un punto di controllo, non viene preso in considerazione e viene notificato di selezionare un nodo oppure un percorso;
 3. Se il punto selezionato è un punto di controllo ma non appartiene a nessuno dei due percorsi consigliati, una finestra di dialogo chiederà all'utente se ricalcolare un nuovo percorso partendo dal nuovo punto di controllo o se riprendere la navigazione.

Story id:3	Task id:3.8
Task tag: Controllo sull'arrivo a destinazione	Software Engineer: Martina Mascolo
Description:	
Una volta che l'utente ha raggiunto la sua destinazione, verrà aperta una finestra di dialogo in cui si può scegliere se aggiornare la propria posizione o se tornare alla homepage per una nuova navigazione	
Test	
Input: Freccia arriva sul nodo di destinazione Output: Compare una finestra di dialogo che invita l'utente a scegliere se aggiornare la propria posizione (e quindi viene visualizzata ancora la stessa mappa), o se tornare alla homepage per una nuova navigazione (e quindi viene visualizzata l'homepage).	

Story id:3	Task id:3.9
Task tag: Controllo sullo switch delle mappe tra i piani, in navigazione, e delle indicazioni relative	Software Engineer: Martina Mascolo
Description:	
Quando l'utente raggiunge un nodo che rappresenta una scala, riceverà l'indicazione sulla direzione di salita o discesa a seconda del piano di passaggio. La mappa dovrà aggiornarsi e switchare su quella corretta, con i rispettivi nodi.	
Test	
Input: Freccia arriva su un nodo che rappresenta una scala Output: La mappa cambia dinamicamente e l'utente riceve le indicazioni di salita o discesa della scala.	

STORY CARD 4

Release:1	Story id:4
Story tag: notifica di emergenza	Priorità:2
Descrizione generale: L'utente deve poter essere avvisato di una condizione di emergenza mediante un notifica.	Dettaglio: La condizione di emergenza viene comunicata dal server all'applicazione; quest'ultima avviserà l'utente attraverso una notifica. L'utente può selezionare la notifica, per entrare automaticamente in modalità di emergenza nella schermata di selezione della posizione iniziale.

Task

Story id:4	Task id:4.1
Task tag: Schede CRC	Software Engineer: Mattia Corrente
Description:	
Preparare tutte le schede CRC	
Effettuare operazioni sulle schede CRC di selezione e disambiguazione	

Story id:4	Task id:4.2
Task tag: Analisi orientate alle UI	Software Engineer: Mattia Corrente
Description:	
<ol style="list-style-type: none"> 1. Selezionare schede CRC relative agli utenti 2. Vedere quali responsabilità necessitano di un'interfaccia 3. Preparare il mockup 4. Preparazione modelli di dialogo, navigazione e presentazione 	

Story id:4	Task id:4.3
Task tag: Analisi orientata alle classi	Software Engineer: Mattia Corrente
Description:	
1. Classificare le schede CRC in classi e attributi	

- | |
|--|
| 2. Definire associazioni e altre relazioni |
| 3. Disegnare diagramma delle classi |

Story id:4	Task id:4.4
Task tag: Analisi dei comportamenti	Software Engineer: Mattia Corrente
Description:	
1. Preparazione diagramma delle sequenze, di stato e delle attività.	

Story id:4	Task id:4.5
Task tag: Funzionamento della notifica di emergenza	Software Engineer: Mattia Corrente
Description:	
L'utente, se il server è online e vi è una situazione di emergenza nell'edificio, deve poter essere avvisato mediante una notifica. Aprendo tale notifica, l'utente deve essere reindirizzato alla pagina di selezione della posizione, impostando automaticamente la modalità di navigazione in emergenza.	
Test	
Input:	
1. Ricezione di una notifica di emergenza spedita dal server all'applicazione mobile; 2. Tap dell'utente sulla notifica	
Output:	
1. Notifica all'utente della sopravvenuta emergenza. 2. Richiesta posizione bypassando la scelta della modalità	

STORY CARD 5

Release:1	Story id:5
Story tag: Guida al punto di raccolta in modalità online	Priorità:1
Descrizione generale: Nel caso di emergenza finché la connessione con il server rimarrà disponibile, l'applicazione dovrà guidare l'utente al punto di raccolta	Dettaglio: La mobile app deve, solo in un primo momento, richiedere all'utente di indicarle la propria posizione. Di conseguenza verrà calcolato il percorso ottimo che condurrà l'utente verso il

<p>più vicino a partire dal punto di partenza precedentemente comunicato dall'utente.</p>	<p>punto di raccolta più vicino e sicuro. L'app fornirà, inoltre, un percorso alternativo che l'utente potrà selezionare e seguire nel caso lo ritenga necessario. La pericolosità dei percorsi sarà stabilita sulla base di alcune variabili fornite al server dal sistema sensoristico. Entrambi i percorsi saranno tracciati sulla mappa. La posizione istantanea dell'utente, lungo il tracciato scelto, verrà approssimata desumendola dalla presunta velocità media di camminata. Sarà l'utente stesso, se lo desidera, ad aggiornare la sua posizione esatta facendo un tap sul punto di controllo del percorso scelto su cui sta transitando. Inseguendo la freccia l'utente potrà proseguire. Ad ogni aggiornamento della posizione, l'app memorizza le variabili di costo ricevute dal server e ad un eventuale richiesta di aggiornamento del percorso da parte dell'utente, lo aggiornerà sulla base del costo K per ogni arco.</p>
---	---

Task

Story id:5	Task id:5
Task tag: Schede CRC	Software Engineer: gianoberto giampieri
Description:	
Preparare tutte le schede CRC	
Effettuare operazioni sulle schede CRC di selezione e disambiguazione	

Story id:5	Task id:5.2
Task tag: Analisi orientate alle UI	Software Engineer: Gianoberto Giampieri
Description:	
<ol style="list-style-type: none"> 1. Selezionare schede CRC relative agli utenti 2. Vedere quali responsabilità necessitano di un'interfaccia 3. Preparare il mockup 	

4. Preparazione modelli di dialogo, navigazione e presentazione

Story id:5	Task id:5.3
Task tag: Analisi orientata alle classi	Software Engineer: Gianoberto Giampieri
Description:	
1. Classificare le schede CRC in classi e attributi 2. Definire associazioni e altre relazioni 3. Disegnare diagramma delle classi	

Story id:5	Task id:5.4
Task tag: Analisi dei comportamenti	Software Engineer: Gianoberto Giampieri
Description:	
1. Preparazione diagramma delle sequenze, di stato e delle attività.	

Story id:5	Task id:5.5
Task tag: Test di navigazione in emergenza	Software Engineer: Gianoberto Giampieri
Description:	
In caso di emergenza, all'utente è richiesto soltanto di indicare la propria posizione. La destinazione sarà successivamente selezionata automaticamente desumendola dal punto di raccolta più vicino al nodo di partenza, informazione contenuta all'interno del database, per ogni nodo.	
Test	
Input:	
1. Seleziona modalità di emergenza 2. Seleziona modalità di inserimento posizione iniziale 3. Seleziona posizione iniziale	
Output:	
1. Notifica all'utente se si trova già nei pressi di un punto di raccolta 2. Altrimenti, traccia percorso verso la destinazione stabilita automaticamente	

Story id:5	Task id:5.6
Task tag: Controllo per aggiornamento variabili di costo dal server	Software Engineer: Gianoberto Giampieri
Description: In caso di emergenza il server può aggiornare le variabili di costo relativi agli archi e di conseguenza, se l'utente richiede di aggiornare la propria posizione, otterrà un percorso pesato con le nuove variabili	
Test	
Input: Invio dal server dell'aggiornamento delle variabili di costo, e tap su un nodo esterno al percorso per richiedere l'aggiornamento della posizione	
Output: Vengono ricalcolati i percorsi e, se i pesi cambiano di molto, possono differire dai precedenti per la stessa tratta.	

STORY CARD 6

Release:1	Story id:6
Story tag: Guida al punto di raccolta in modalità offline	Priorità:1
Descrizione generale: Nel caso di emergenza e in assenza di connessione con il server, l'applicazione dovrà comunque guidare l'utente al punto di raccolta migliore.	Dettaglio: L'utente non potrà essere aggiornato della situazione di emergenza intorno a lui tramite la mobile app, non essendo possibile connettersi al server. Nel caso prenda visione della situazione di emergenza potrà comunque comunicare la propria posizione. Di conseguenza verrà calcolato il percorso ottimo, indicato tramite freccia, che condurrà l'utente verso il punto di raccolta più vicino e sicuro. L'app fornirà, inoltre, un percorso alternativo che l'utente potrà selezionare e seguire nel caso lo ritenga necessario. Non disponendo di

	connessione, il costo totale K relativo al percorso scelto sarà calcolato sulla base degli ultimi valori delle variabili di costo memorizzate nel database. Entrambi i percorsi saranno tracciati sulla mappa. La posizione istantanea dell'utente lungo il tracciato scelto, verrà approssimata desumendola dalla presunta velocità media di camminata. Sarà l'utente stesso, se lo desidera, ad aggiornare la sua posizione esatta facendo un tap sul punto di controllo del percorso scelto sul quale sta transitando. In tal caso l'app mobile non sarà però capace di offrirgli un percorso aggiornato.
--	--

Task

Story id:6	Task id:6.1
Task tag: Schede CRC	Software Engineer: mattia corrente
Description:	
Preparare tutte le schede CRC	
Effettuare operazioni sulle schede CRC di selezione e disambiguazione	

Story id:6	Task id:6.2
Task tag: Analisi orientate alle UI	Software Engineer: mattia corrente
Description:	
<ol style="list-style-type: none"> 1. Selezionare schede CRC relative agli utenti 2. Vedere quali responsabilità necessitano di un'interfaccia 3. Preparare il mockup 4. Preparazione modelli di dialogo, navigazione e presentazione 	

Story id:6	Task id:6.3
Task tag: Analisi orientata alle classi	Software Engineer: mattia corrente
Description:	
1. Classificare le schede CRC in classi e attributi	

- | |
|--|
| 2. Definire associazioni e altre relazioni |
| 3. Disegnare diagramma delle classi |

Story id:6

Task id:6.4

Task tag: Analisi dei comportamenti

Software Engineer: Mattia Corrente

Description:

2. Preparazione diagramma delle sequenze, di stato e delle attività.

Story id: 6

Task id: 6.5

Task tag: Test di navigazione in emergenza

Software Engineer: Mattia Corrente

Description:

In caso di emergenza, all'utente è richiesto soltanto di indicare la propria posizione. La destinazione sarà successivamente selezionata automaticamente desumendola dal punto di raccolta più vicino al nodo di partenza, informazione contenuta all'interno del database, per ogni nodo.

Test

Input:

1. Seleziona modalità di emergenza
2. Seleziona modalità di inserimento posizione iniziale
3. Seleziona posizione iniziale

Output:

3. Notifica all'utente se si trova già nei pressi di un punto di raccolta
4. Altrimenti, traccia percorso verso la destinazione stabilita automaticamente

4.2 Analisi orientata alle interfacce

In questa fase per ogni story card si analizzano i requisiti per comprendere come progettare le interfacce. Definite le interfacce, si analizzano le strutture dati che il software dovrà usare.

4.2.1 Analisi CRC

CRC sta per Classi-Responsabilità-Collaboratori. Ogni scheda CRC rappresenta una classe, se ha responsabilità e collaboratori, o un attributo, se non ha responsabilità e pochi collaboratori. Le responsabilità sono le operazioni incapsulate nelle classi mentre i collaboratori sono le classi o gli attributi con cui la scheda in esame collabora.

Autenticazione utente

Flusso 1: prima apertura con Utente connesso al WIFI, e server disponibile

- la mobile app visualizza la schermata di login alla prima apertura
- l'utente inserisce le proprie credenziali di ateneo e preme il tasto login
- se le credenziali sono corrette la mobile app visualizza la schermata di selezione della modalità di navigazione altrimenti visualizza di nuovo la schermata di login con l'errato inserimento delle credenziali

Flusso 2: aperture successive dopo salvataggio credenziali con Utente connesso al WIFI e server disponibile

- la mobile app visualizza la schermata di selezione della modalità di navigazione

Flusso 3: utente connesso al WIFI e server non disponibile

- la mobile app visualizza la schermata di login alla prima apertura
- l'utente preme il FAB button
- la mobile app visualizza la schermata di selezione della modalità di navigazione bypassando l'inserimento delle credenziali

Flusso3: utente non connesso al WIFI

- la mobile app visualizza la schermata di login alla prima apertura
- l'utente preme il FAB button
- la mobile app invita l'utente a connettersi

Schede CRC

Class: Utente	
Description: Utente dell'applicazione	
Responsibility	Collaborator
<ul style="list-style-type: none"> ● inserisce le proprie credenziali di ateneo ● premere il tasto “login” ● preme il FAB button 	<ul style="list-style-type: none"> ● credenziali ● mobile app ● tasto Fab

Class: Mobile app	
Description:	
Responsibility	Collaborator
<ul style="list-style-type: none"> ● visualizzare schermata di login ● visualizza la schermata di selezione della modalità di navigazione ● visualizza schermata di login evidenziando l'errato inserimento delle credenziali ● salvare credenziali nel database per le successive aperture se corrette ● controllare giusto inserimento credenziali ● permettere di bypassare 	<ul style="list-style-type: none"> ● Utente ● Tasto FAB ● Server ● credenziali

l'inserimento delle credenziali quando lo smartphone è connesso alla rete WIFI di ateneo, ma il server sia offline.

Invio e aggiornamento della posizione

Flusso: acquisire posizione utente tramite inserimento room code

- la mobile app visualizza la schermata di selezione modalità di navigazione
 - l'utente seleziona una delle due modalità di navigazione
- la mobile app visualizza schermata di scelta della modalità di riconoscimento posizione
 - l'utente seleziona “insert room code”
- la mobile app visualizza un box di testo
 - l'utente inserisce il codice della stanza da dove parte e clicca select
- la mobile app verifica la presenza o meno del codice interno della stanza altrimenti invita ad inserire un codice corretto

Flusso: acquisire posizione utente tramite cattura QR code

- la mobile app visualizza la schermata di selezione modalità di navigazione
 - l'utente seleziona una delle due modalità di navigazione
- la mobile app visualizza schermata di scelta della modalità di riconoscimento posizione
 - l'utente seleziona “shot a QR CODE”
- la mobile app visualizza schermata di cattura QR CODE
 - l'utente fa scansionare il QR CODE contenente il codice della stanza
- la mobile app acquisirà il QR CODE se presente altrimenti visualizza messaggio di errore se il nome della stanza non è presente nel database;

Flusso: acquisire posizione utente tramite tap sulla mappa

- la mobile app visualizza la schermata di selezione modalità di navigazione

- l'utente seleziona una delle due modalità di navigazione
- la mobile app visualizza schermata di scelta della modalità di riconoscimento posizione iniziale
- l'utente seleziona "tap on the map"
- la mobile app visualizza un corousel dove vi sono i piani
- l'utente seleziona un piano dal carousel
- la mobile app visualizza il piano della mappa corrispondente
- l'utente fa tap su uno dei punti di controllo presenti sul piano visualizzato

Flusso: acquisire nuova posizione utente

- la mobile app visualizza la mappa con i percorsi consigliati
- l'utente durante la navigazione seleziona un punto di controllo all'interno della mappa del piano per indicare la nuova posizione
- se il nuovo punto di controllo selezionato appartiene a uno dei due percorsi la mobile app visualizza la nuova posizione dell'utente sulla mappa altrimenti visualizza una finestra di dialogo chiedendo all'utente se ricalcolare un nuovo percorso partendo dal nuovo punto di controllo o se ignorare la richiesta.

Schede CRC

Class: utente	
Description: l'utente dell'applicazione	
Responsibility	Collaborator
<ul style="list-style-type: none"> • seleziona una delle due modalità di navigazione • seleziona inserimento codice stanza • inserisce il codice della stanza da dove parte e clicca select 	<ul style="list-style-type: none"> • posizione iniziale • server • mobile app • punti di controllo • percorso

<ul style="list-style-type: none"> ● seleziona scansione QR CODE ● scansiona il QR CODE contenente il codice della stanza per posizione iniziale ● seleziona “tap on the map” ● seleziona un piano dal carousel ● tap su uno dei punti di controllo presenti sul piano visualizzato per posizione iniziale ● seleziona un punto di controllo all'interno della mappa de piano per indicare la nuova posizione durante la navigazione 	
--	--

Class: mobile app	
Description: Applicazione di navigazione per smartphone e tablet	
Responsibility	Collaborator
<ul style="list-style-type: none"> ● visualizza schermata di scelta della modalità di riconoscimento posizione ● visualizza box di testo ● verifica la presenza o meno del codice interno della stanza ● visualizza richiesta di inserimento di un codice della stanza corretto ● visualizza schermata di cattura QR CODE ● acquisire QR CODE se presente ● visualizza messaggio di errore se il nome della stanza scansionato non è presente nel db; ● visualizza carousel dove vi sono i piani ● visualizza piano della mappa corrispondente ● visualizza la mappa con i percorsi consigliati 	<ul style="list-style-type: none"> ● Utente ● posizione ● percorso ● punti di controllo

- visualizza la nuova posizione dell'utente sulla mappa se il nuovo punto di controllo selezionato appartiene a uno dei due percorsi
- visualizza una finestra di dialogo chiedendo all'utente se ricalcolare un nuovo percorso partendo dal nuovo punto di controllo o se ignorare la richiesta.
- aggiornare posizione utente sul server

Funzionamento in condizioni normali

Flusso: con destinazione settata tramite inserimento room code

-la mobile app visualizza schermata di inserimento dello destinazione, una volta acquisita la posizione iniziale

- l'utente seleziona l'inserimento del codice della stanza per comunicare la destinazione desiderata

-la mobile app visualizza un box di testo

- l'utente inserisce il codice della stanza che desidera raggiungere e conferma l'inserimento

-la mobile app verifica la presenza o meno del codice della stanza e se esso non coincide con la posizione iniziale, visualizza i due percorsi alternativi calcolati (in blu il percorso migliore e in grigio quello alternativo) sulla base delle lunghezze degli archi tra la partenza e la destinazione, altrimenti invita ad inserire un codice corretto

- l'utente seleziona il percorso che ritiene migliore

-la mobile app visualizza la mappa con il relativo percorso evidenziato, all'interno del quale la posizione dell'utente è indicata da una freccia verde che avanza lungo il percorso nella direzione corretta sulla base della velocità media presunta;inoltre visualizza anche delle indicazioni direzionali , sia tramite interfaccia che sotto forma di notifica tramite vibrazione e anche su un eventuale dispositivo Android Wear dell'utente, al passaggio di ogni punto di controllo o scala.

- l'utente aggiorna la sua posizione potendo switchare tra un percorso e l'altro selezionando, durante la navigazione, punti di controllo intermedi
 - se il punto selezionato è un punto di controllo appartenente a uno dei percorsi la mobile app visualizza la nuova posizione dell'utente tramite freccia verde ricalcolata, che si riposizionerà in quel punto (che può appartenere ad uno o l'altro percorso) e riproseglierà fino al punto di controllo di destinazione
 - Se il punto selezionato non appartiene ai percorsi consigliati e non è un punto di controllo, la mobile app invita a selezionare un nodo oppure un percorso
 - Se il punto selezionato è un punto di controllo ma non appartenente a nessuno dei due percorsi consigliati la mobile app visualizza una finestra di dialogo chiedendo all'utente se ricalcolare un nuovo percorso, partendo dal nuovo punto di controllo, o se riprendere la navigazione.
- l'utente raggiunge la destinazione

-la mobile app visualizza una finestra di dialogo in cui si può scegliere se aggiornare la propria posizione o se tornare alla schermata di selezione modalità di navigazione per una nuova navigazione

Flusso: con destinazione settata tramite tap sulla mappa

-la mobile app visualizza schermata di inserimento dello destinazione, una volta acquisita la posizione iniziale

- l'utente seleziona “tap on the map”
- la mobile app visualizza un carousel dove vi sono i piani
- l'utente seleziona il piano corrispondente dal carousel
- la mobile app visualizza il piano della mappa corrispondente
- l'utente fa tap su uno dei punti di controllo presenti sul piano visualizzato

-la mobile app acquisendo il punto di controllo come destinazione ,se è diverso dal punto di controllo di partenza visualizza i due migliori percorsi calcolati sulla base delle lunghezze degli archi tra la partenza e la destinazione (in blu il percorso migliore e in grigio quello alternativo), altrimenti invita a fare tap su un punto di controllo diverso.

- l'utente seleziona il percorso che ritiene migliore

-la mobile app visualizza la mappa con il relativo percorso evidenziato, all'interno del quale la posizione dell'utente è indicata da una freccia verde che avanza lungo il percorso nella direzione corretta sulla base della velocità media presunta;inoltre visualizza anche delle indicazioni direzionali , sia tramite interfaccia che sotto forma di notifica tramite vibrazione e anche su un eventuale dispositivo Android Wear dell'utente, al passaggio di ogni punto di controllo o scala.

- l'utente aggiorna la sua posizione potendo switchare tra un percorso e l'altro selezionando, durante la navigazione, punti di controllo intermedi:

- se il punto selezionato è un punto di controllo appartenente a uno dei percorsi la mobile app visualizza la nuova posizione dell'utente tramite freccia verde ricalcolata, che si riposizionerà in quel punto (che può appartenere ad uno o l'altro percorso) e riproseglierà fino al punto di controllo di destinazione
- Se il punto selezionato non appartiene ai percorsi consigliati e non è un punto di controllo, la mobile app invita a selezionare un nodo oppure un percorso
- Se il punto selezionato è un punto di controllo ma non appartenente a nessuno dei due percorsi consigliati la mobile app visualizza una finestra di dialogo chiedendo all'utente se ricalcolare un nuovo percorso,partendo dal nuovo punto di controllo, o se riprendere la navigazione.

- l'utente raggiunge la destinazione

-la mobile app visualizza una finestra di dialogo in cui si può scegliere se aggiornare la propria posizione o se tornare alla schermata di selezione modalità di navigazione per una nuova navigazione

Schede CRC

Class: Utente	
Description: Utente dell'applicazione	
Responsibility	Collaborator
● seleziona inserimento "room code"	● Posizione iniziale

<ul style="list-style-type: none"> ● inserisce il codice della stanza come destinazione ● premere il tasto “select” ● seleziona “tap on the map” ● seleziona un piano dal carousel ● tap su uno dei punti di controllo presenti sul piano visualizzato, come destinazione ● seleziona percorso scelto ● aggiorna posizione switchando tra un percorso e l’altro o selezionando un altro punto di controllo 	<ul style="list-style-type: none"> ● mobile app ● Freccia ● indicazioni direzionali ● Mappa ● piano ● destinazione desiderata ● Velocità media ● Punti di controllo ● pesi ● L_arco ● variabili di costo
---	---

<p>Class: mobile app</p> <p>Description: Applicazione di navigazione per smartphone e tablet</p>	
<p>Responsibility</p> <ul style="list-style-type: none"> ● visualizzare schermata di inserimento posizione iniziale ● visualizza un box di testo ● visualizza un corousel dove vi sono i piani ● visualizza il piano della mappa corrispondente ● verifica la presenza del “room code” e la non coincidenza con la posizione iniziale ● visualizza il percorso più breve e quello alternativo secondo L_arco ● visualizza un invito ad inserire un codice della stanza corretto 	<p>Collaborator</p> <ul style="list-style-type: none"> ● Posizione iniziale ● Utente ● Freccia ● indicazioni direzionali ● Mappa ● piano ● destinazione desiderata ● Velocità media ● Punti di controllo ● scala ● pesi ● L_arco

- visualizza un invito a fare tap su un punto di controllo diverso.
- visualizza il piano della mappa con il relativo percorso scelto, la posizione utente (freccia verde) che avanza con una velocità media stimata lungo il percorso, e le indicazioni direzionali al passaggio di ogni punto di controllo o scala.
- visualizza indicazioni direzionali tramite notifica su smartphone e su un eventuale dispositivo Android Wear dell'utente, al passaggio di ogni punto di controllo o scala.
- visualizza la nuova posizione dell'utente a seguito di tap
- visualizza un invito a selezionare un punto di controllo oppure un percorso
- visualizza finestra di dialogo che chiede di ricalcolare un nuovo percorso, partendo dal nuovo punto di controllo, o se riprendere la navigazione.
- visualizza una finestra di dialogo in cui scegliere se aggiornare la propria posizione o se tornare alla schermata di selezione modalità di navigazione per una nuova navigazione
- visualizza una finestra di dialogo in cui scegliere se aggiornare la propria posizione o se tornare alla “schermata di selezione modalità di navigazione” per l' inizio di una nuova navigazione

- variabili di costo

Notifica di emergenza

Descrizione: notifica di una situazione di emergenza con utente connesso al WIFI, e server disponibile

Flusso:

-la mobile app riceve dal server la presenza di un'emergenza all'interno dell'edificio e la comunica all'utente tramite notifica “Attenzione emergenza!!” sullo smartphone

- l'utente seleziona la notifica di emergenza “Attenzione emergenza!!”

-la mobile app visualizza automaticamente modalità di emergenza nella schermata di selezione della posizione iniziale, bypassando la scelta della modalità di navigazione

Schede CRC

Class: Utente	
Description: Utente dell'applicazione	
Responsibility	Collaborator
<ul style="list-style-type: none">• selezione notifica di emergenza “Attenzione emergenza!!”• Accedere alla modalità di emergenza	<ul style="list-style-type: none">• Notifica di emergenza• connessione WIFI• server• mobile app• Modalità di emergenza

Class: mobile app	
Description: Applicazione di navigazione per smartphone e tablet	
Responsibility	Collaborator
<ul style="list-style-type: none">• ricevere una notifica di emergenza dal server• visualizzare schermata di comunicazione posizione iniziale in modalità di emergenza	<ul style="list-style-type: none">• Utente• connessione WIFI• server• mobile app• Notifica di emergenza• modalità di emergenza

Guida al punto di raccolta in modalità online

Flusso:

-la mobile app visualizza schermata di selezione modalità di navigazione dopo aver verificato il giusto inserimento delle credenziali (vd. story card 1 Autenticazione utente)

- l'utente seleziona la modalità di navigazione “Emergency mode”

-la mobile app visualizza schermata di inserimento della posizione iniziale

- l'utente seleziona la modalità di inserimento della posizione come spiegato nella story card 2

- la mobile app visualizza che si è già in un punto di raccolta se questo coincide con la posizione iniziale ricevuta e si ritorna alla schermata di selezione del punto iniziale

- se la posizione iniziale non coincide con un punto di raccolta, la mobile app visualizza i due percorsi alternativi(in blu il percorso migliore e in grigio quello alternativo) dalla posizione iniziale (precedentemente verificata) al punto di raccolta più vicino e stabilito per quel nodo di partenza, calcolati sulla base delle variabili di costo e quindi del costo_k, aggiornate dal server , relative a tutti gli archi tra la partenza e la destinazione

- l'utente seleziona il percorso che intende seguire

-la mobile app visualizza la mappa con il relativo percorso evidenziato, all'interno del quale la posizione istantanea dell'utente è indicata da una freccia verde che avanza lungo il percorso nella direzione corretta sulla base della velocità media di camminata; inoltre visualizza anche delle indicazioni direzionali , sia tramite interfaccia che sotto forma di notifica tramite vibrazione e anche su un eventuale dispositivo Android Wear dell'utente, al passaggio di ogni punto di controllo o scala.

- l'utente aggiorna la sua posizione durante la navigazione selezionando, punti di controllo intermedi appartenenti rispettivamente ai due percorsi o alternativamente anche punti di controllo esterni

- se il punto selezionato è un punto di controllo appartenente a uno dei percorsi la mobile app visualizza la nuova posizione dell'utente tramite freccia verde ricalcolata, che si riposizionerà in quel punto (che può appartenere ad uno o l'altro percorso); in caso di un aggiornamento dal server delle variabili di costo può aggiornare e visualizzare altri percorsi
 - Se il punto selezionato non appartiene ai percorsi consigliati e non è un punto di controllo, la mobile app invita a selezionare un nodo oppure un percorso
 - Se il punto selezionato è un punto di controllo ma non appartenente a nessuno dei due percorsi consigliati la mobile app visualizza una finestra di dialogo chiedendo all'utente se ricalcolare un nuovo percorso, partendo dal nuovo punto di controllo, o se riprendere la navigazione.
- l'utente raggiunge la destinazione

-la mobile app visualizza una finestra di dialogo in cui si può scegliere se aggiornare la propria posizione o se tornare alla schermata di selezione modalità di navigazione per una nuova navigazione

Schede CRC

Class: Utente	
Description: Utente dell'applicazione	
Responsibility	Collaborator
<ul style="list-style-type: none"> • seleziona la modalità di navigazione “Emergency mode” • seleziona la modalità di inserimento della posizione iniziale 	<ul style="list-style-type: none"> • posizione iniziale • punto di raccolta • freccia • indicazioni direzionali • mobile app • Mappa

<ul style="list-style-type: none"> ● inserisce posizione iniziale ● seleziona il percorso ● aggiorna posizione switchando tra un percorso e l'altro o selezionando un altro punto di controllo 	<ul style="list-style-type: none"> ● piano ● Velocità media ● Punti di controllo ● variabili di costo ● server ● connessione WIFI ● pesi ● costo_k ● variabili di costo
---	--

Class: mobile app	
Description: Applicazione di navigazione per smartphone e tablet	
Responsibility <ul style="list-style-type: none"> ● visualizza schermata di selezione modalità di navigazione ● visualizza schermata di inserimento della posizione iniziale ● visualizza che si è già in un punto di raccolta ● visualizza sulla mappa il percorso più breve e quello alternativo secondo il costo_k presente sul server, fino al punto di raccolta più vicino ● visualizza il piano della mappa con il relativo percorso scelto, la posizione utente (freccia verde) che avanza con una velocità media stimata lungo il 	Collaborator <ul style="list-style-type: none"> ● posizione iniziale ● punto di raccolta ● freccia ● indicazioni direzionali ● utente ● Mappa ● piano ● Velocità media ● Punti di controllo ● scala ● pesi ● variabili di costo ● costo_k ● server ● connessione WIFI

<p>percorso, e le indicazioni direzionali al passaggio di ogni punto di controllo o scala.</p> <ul style="list-style-type: none"> ● visualizza indicazioni direzionali tramite notifica su smartphone e su un eventuale dispositivo Android Wear dell'utente, al passaggio di ogni punto di controllo o scala. ● visualizza la nuova posizione dell'utente a seguito di tap ● memorizza variabili di costo aggiornate dal server e aggiorna percorso ● visualizza un invito a selezionare un punto di controllo oppure un percorso ● visualizza finestra di dialogo che chiede di ricalcolare un nuovo percorso, partendo dal nuovo punto di controllo, o riprendere la navigazione. ● visualizza una finestra di dialogo in cui scegliere se aggiornare la propria posizione o se tornare alla schermata di selezione modalità di navigazione per una nuova navigazione ● visualizza una finestra di dialogo in cui scegliere se aggiornare la propria posizione o se tornare alla “schermata di selezione modalità di navigazione” per l’ inizio di una nuova navigazione 	
--	--

Guida al punto di raccolta in modalità offline

Descrizione:

Flusso:

-la mobile app visualizza schermata di selezione modalità di navigazione

- l'utente seleziona la modalità di navigazione “Emergency mode”

-la mobile app visualizza schermata di inserimento della posizione iniziale

- l'utente seleziona la modalità di inserimento della posizione come spiegato nella story card 2

- la mobile app visualizza che si è già in un punto di raccolta se questo coincide con la posizione iniziale ricevuta e si ritorna alla schermata di selezione del punto iniziale

- se la posizione iniziale non coincide con un punto di raccolta la mobile app visualizza i due percorsi alternativi (in blu il percorso migliore e in grigio quello alternativo) dalla posizione iniziale (precedentemente verificata) al punto di raccolta più vicino e stabilito per quel nodo, sulla base del costo_k ottenuto dagli ultimi valori delle variabili di costo memorizzate (non aggiornate)

- l'utente seleziona il percorso che intende seguire

-la mobile app visualizza la mappa con il relativo percorso evidenziato, all'interno del quale la posizione istantanea dell'utente è indicata da una freccia verde che avanza lungo il percorso nella direzione corretta sulla base della velocità media di camminata; inoltre visualizza anche delle indicazioni direzionali, sia tramite interfaccia che sotto forma di notifica tramite vibrazione e anche su un eventuale dispositivo Android Wear dell'utente, al passaggio di ogni punto di controllo o scala.

- l'utente aggiorna la sua posizione durante la navigazione selezionando, punti di controllo intermedi appartenenti rispettivamente ai due percorsi o alternativamente anche punti di controllo esterni

- se il punto selezionato è un punto di controllo appartenente a uno dei percorsi la mobile app visualizza la nuova posizione dell'utente tramite freccia verde ricalcolata, che si riposizionerà in quel punto (che può appartenere ad uno o l'altro percorso) e riproseggerà fino al punto di raccolta corrispondente

- Se il punto selezionato non appartiene ai percorsi consigliati e non è un punto di controllo, la mobile app invita a selezionare un nodo oppure un percorso
 - Se il punto selezionato è un punto di controllo ma non appartiene a nessuno dei due percorsi consigliati la mobile app visualizza una finestra di dialogo chiedendo all'utente se ricalcolare un nuovo percorso, partendo dal nuovo punto di controllo, o se riprendere la navigazione.
- I l'utente raggiunge la destinazione
- la mobile app visualizza una finestra di dialogo in cui si può scegliere se aggiornare la propria posizione o se tornare alla schermata di selezione modalità di navigazione per una nuova navigazione

Schede CRC

Class: Utente	
Description: Utente dell'applicazione	
Responsibility	Collaborator
<ul style="list-style-type: none"> ● seleziona la modalità di navigazione “Emergency mode” ● seleziona la modalità di inserimento della posizione iniziale ● inserisce posizione iniziale ● seleziona il percorso ● aggiorna posizione switchando tra un percorso e l'altro o selezionando un altro punto di controllo 	<ul style="list-style-type: none"> ● Posizione ● punto di raccolta ● punti di controllo ● piano ● mappa ● freccia ● indicazioni ● velocità media ● pesi ● costo_k ● variabili di costo ● mobile app

Class: mobile app	
Description: Applicazione di navigazione per smartphone e tablet	
Responsibility	Collaborator
<ul style="list-style-type: none"> ● visualizza schermata di selezione modalità di navigazione ● visualizza schermata di inserimento della posizione iniziale ● visualizza che si è già in un punto di raccolta ● visualizza sulla mappa il percorso più breve e quello alternativo, secondo l'ultimo costo_k non aggiornato, fino al punto di raccolta più vicino ● visualizza il piano della mappa con il relativo percorso scelto, la posizione utente (freccia verde) che avanza con una velocità media stimata lungo il percorso, e le indicazioni direzionali al passaggio di ogni punto di controllo o scala. ● visualizza indicazioni direzionali tramite notifica su smartphone e su un eventuale dispositivo Android Wear dell'utente, al passaggio di ogni punto di controllo o scala. ● visualizza la nuova posizione dell'utente a seguito di tap 	<ul style="list-style-type: none"> ● Posizione ● punto di raccolta ● punti di controllo ● scala ● piano ● mappa ● freccia ● indicazioni ● velocità media ● pesi ● costo_k ● variabili di costo ● utente

<ul style="list-style-type: none"> ● visualizza un invito a selezionare un punto di controllo oppure un percorso ● visualizza finestra di dialogo che chiede di ricalcolare un nuovo percorso, partendo dal nuovo punto di controllo, o riprendere la navigazione. ● visualizza una finestra di dialogo in cui scegliere se aggiornare la propria posizione o se tornare alla schermata di selezione modalità di navigazione per una nuova navigazione ● visualizza una finestra di dialogo in cui scegliere se aggiornare la propria posizione o se tornare alla “schermata di selezione modalità di navigazione” per l’ inizio di una nuova navigazione 	
---	--

4.2.2 User e Task model, Modelli di Presentazione, di Navigazione e di Dialogo

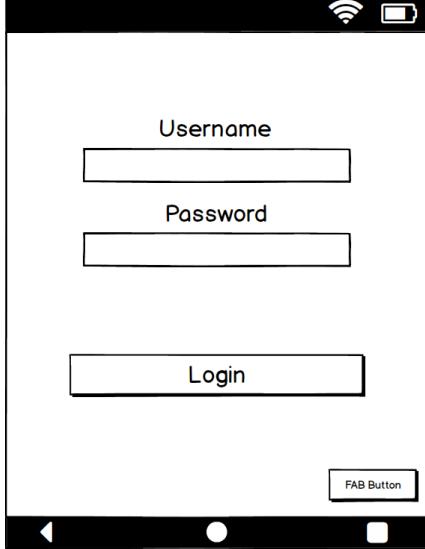
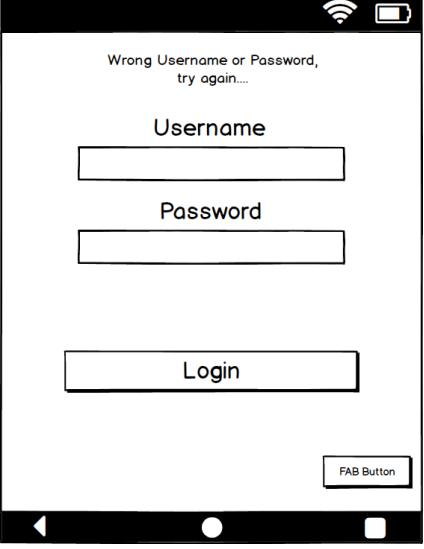
A questo punto, oltre ai modelli utente e dei task, per ogni story card si vogliono illustrare le interfacce che si andranno ad implementare, e le interazioni tra utente e app, attraverso i modelli di Presentazione (che racchiudono i mockup preliminari dell’interfaccia), i modelli di Navigazione (che illustrano il susseguirsi delle schermate) e quelli di Dialogo (che mettono in evidenza le interazioni)

Autenticazione utente

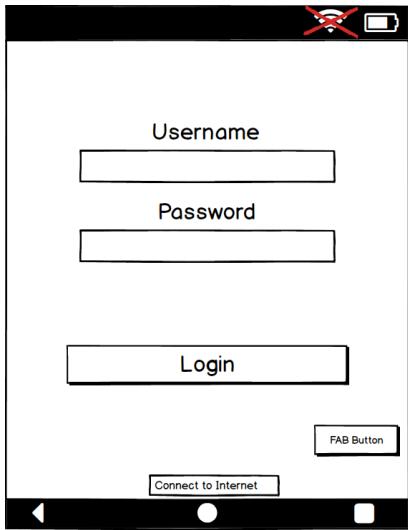
User and task Model

User Model	Utente
Task Model	
	<ul style="list-style-type: none">• inserisce le proprie credenziali di ateneo• premere il tasto “login”• preme il FAB button

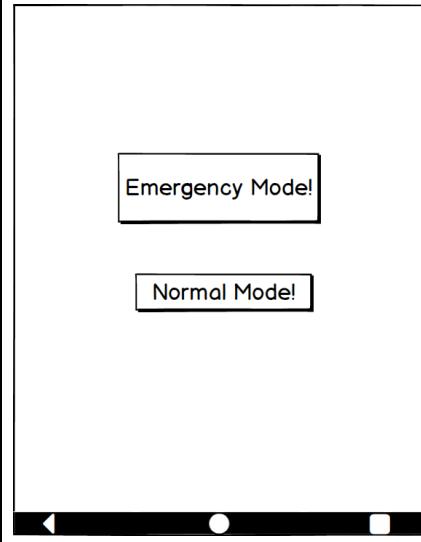
Modello di presentazione

<p>“Schermata di login”</p> 	<p>“Credenziali errate”</p> 
---	--

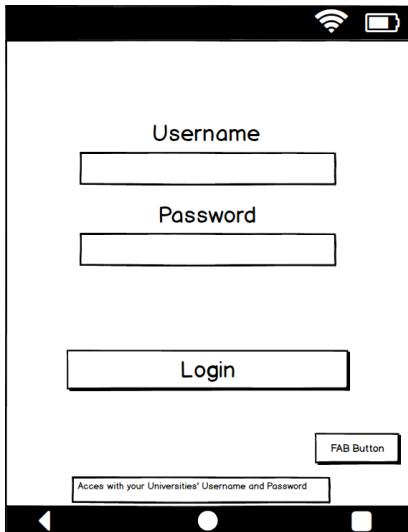
"Utente preme su FAB button, non collegato alla rete"



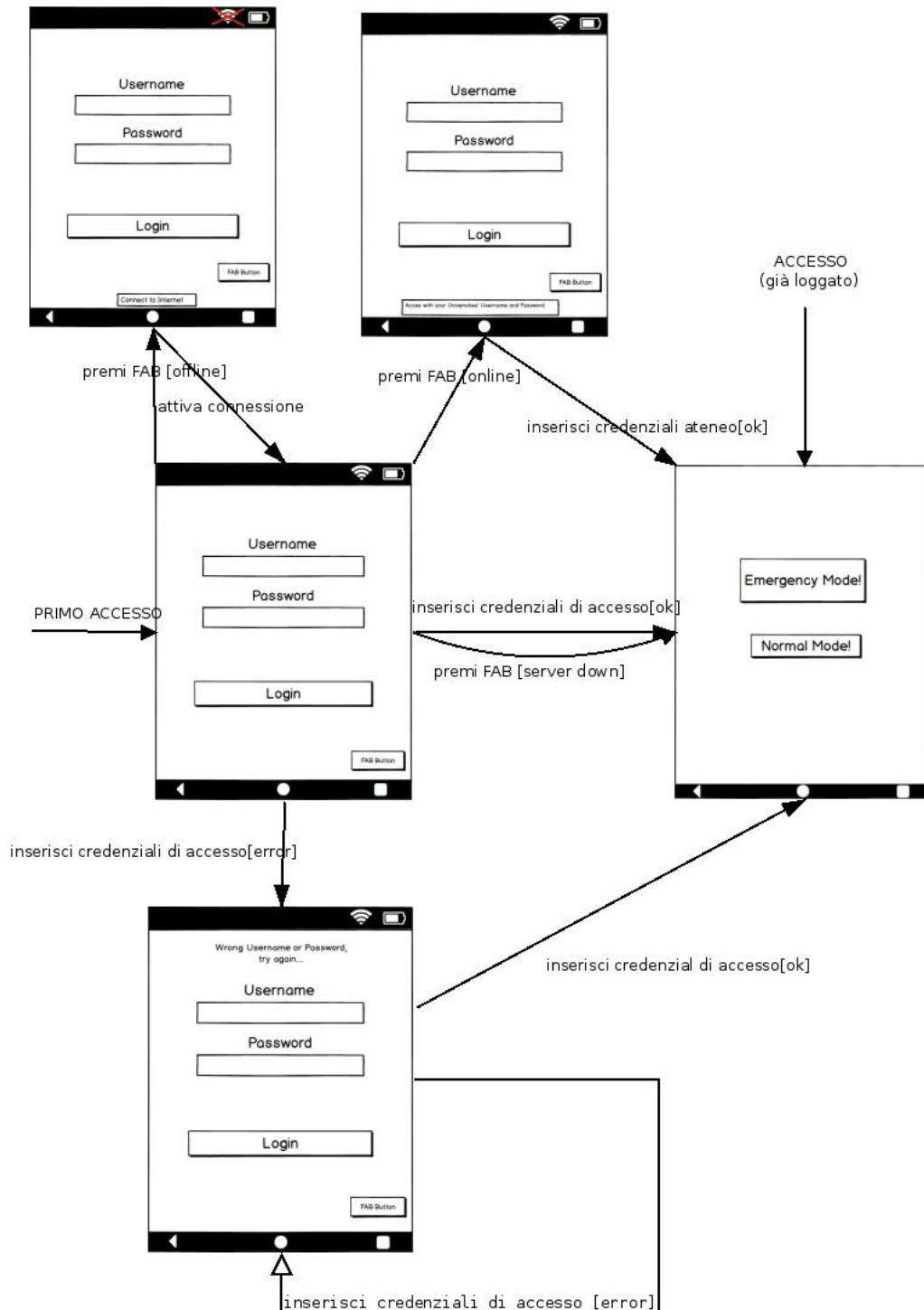
"Homepage"



"Utente preme su FAB button, server online"

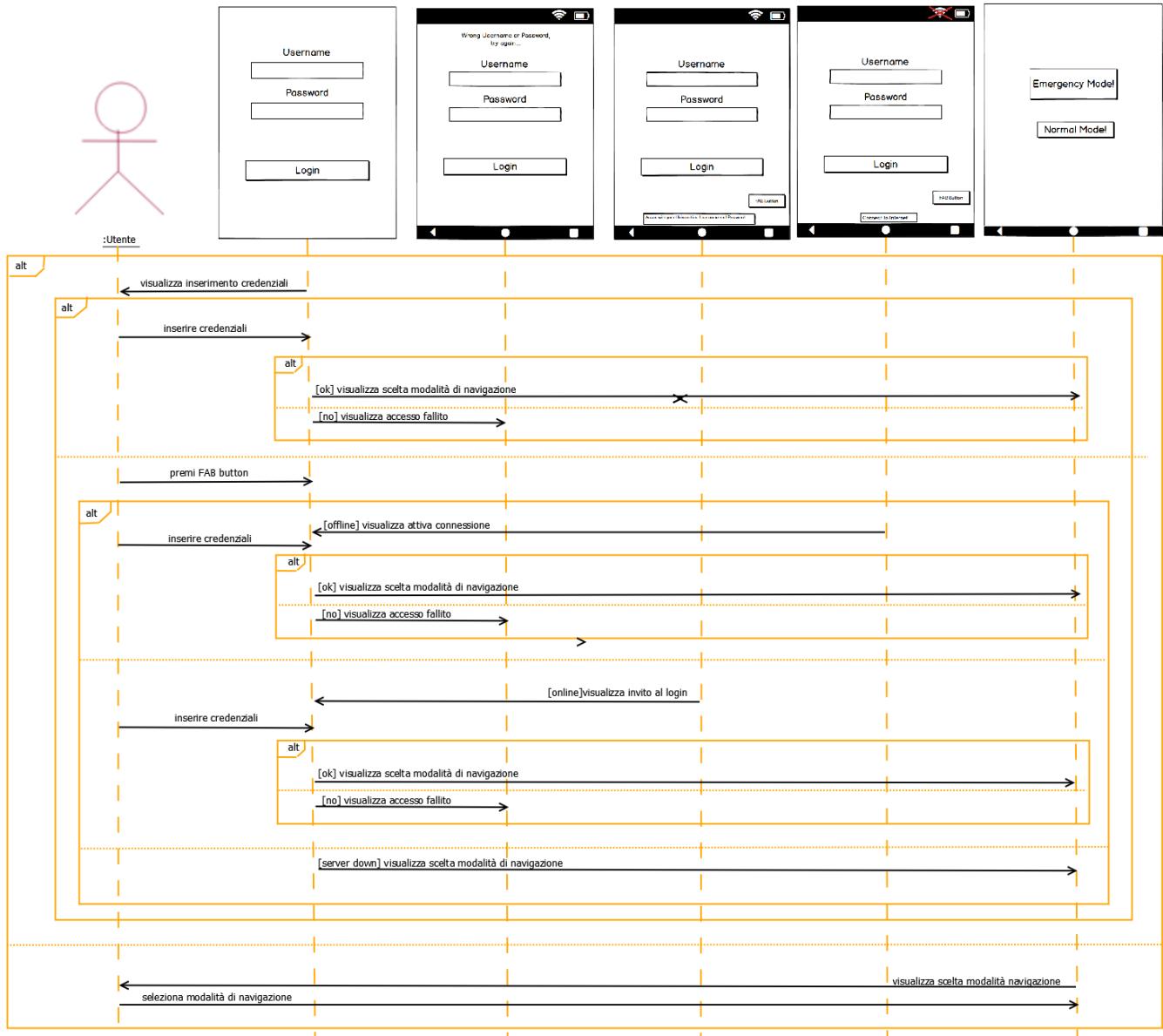


Modello di navigazione



Modello di dialogo

L'alternativa più esterna evidenzia l'interazione dell'applicazione con l'utente sia al primo accesso di quest'ultimo e sia agli accessi successivi.



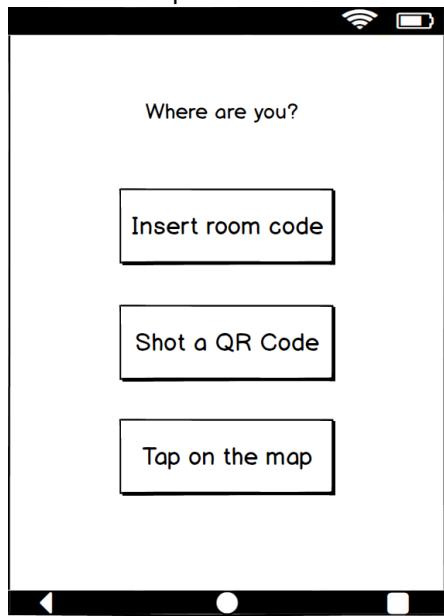
Invio e aggiornamento della posizione (caso per la sola emergenza)

User and task Model

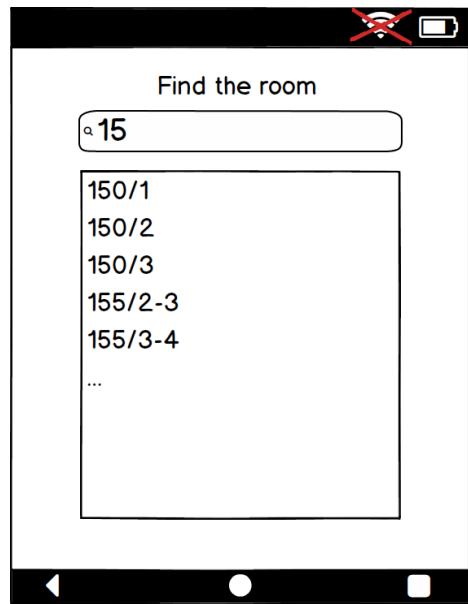
User Model	Utente
Task Model	<ul style="list-style-type: none">● seleziona una delle due modalità di navigazione● seleziona inserimento codice stanza● inserisce il codice della stanza da dove parte e clicca select● seleziona scansione QR CODE● scansiona il QR CODE contenente il codice della stanza per posizione iniziale● seleziona “tap on the map”● seleziona un piano dal carousel● tap su uno dei punti di controllo presenti sul piano visualizzato per posizione iniziale● seleziona un punto di controllo all'interno della mappa de piano per indicare la nuova posizione durante la navigazione

Modello di presentazione

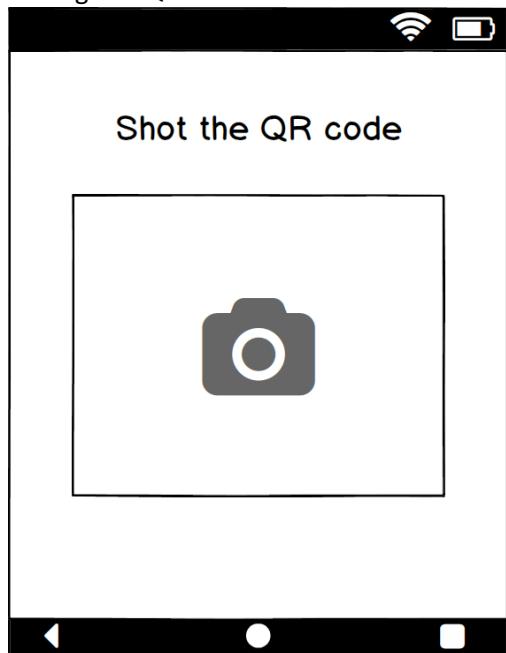
"Inserimento posizione"



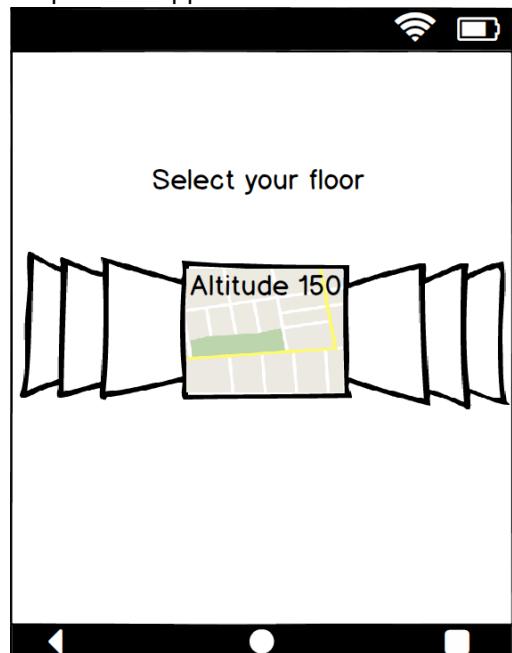
"Inserimento codice stanza"



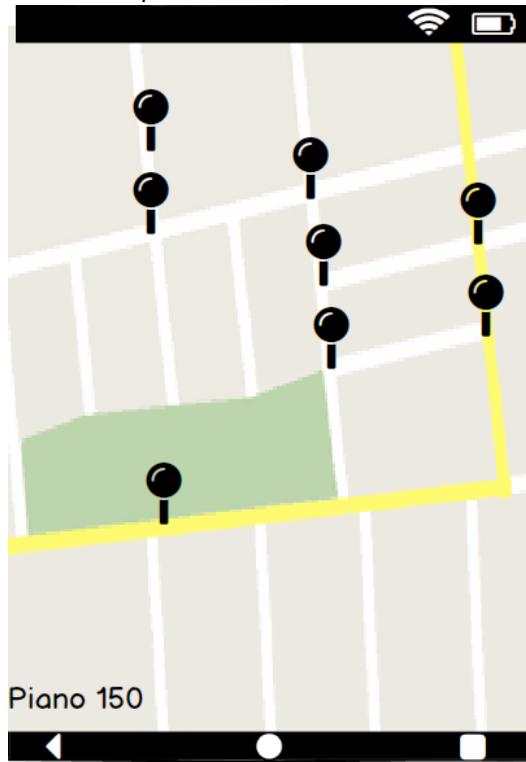
"Fotografa QR Code"



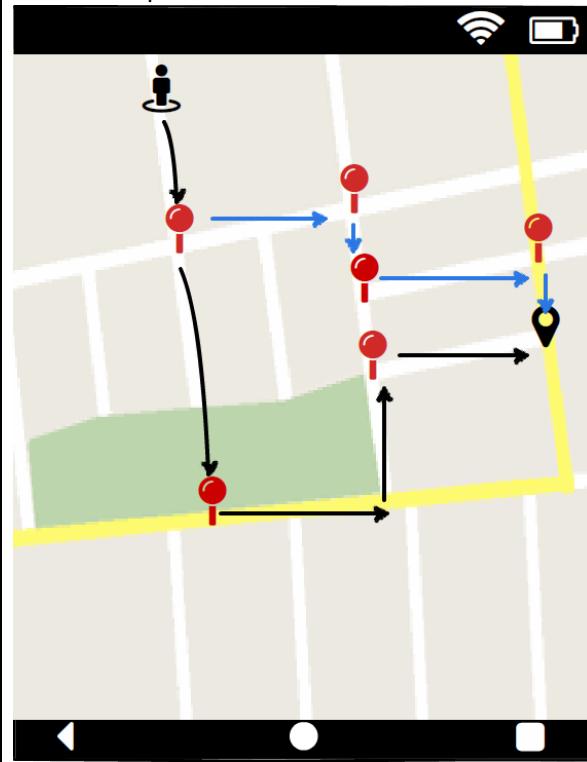
"Tap sulla mappa"



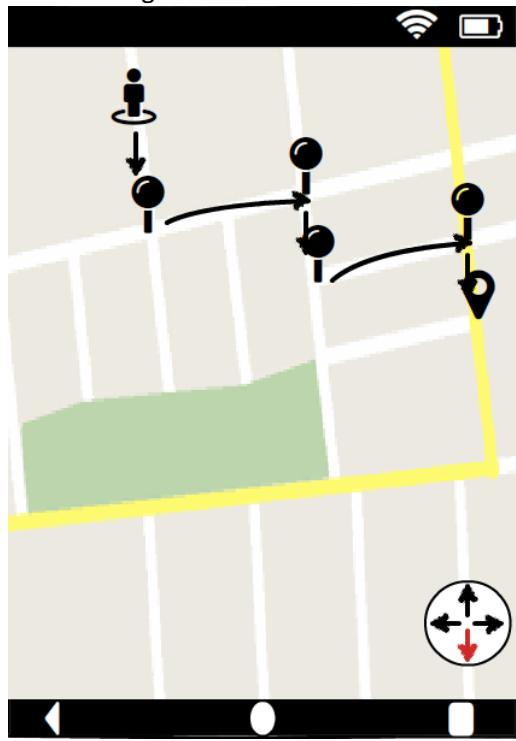
"Selezione punto di controllo"



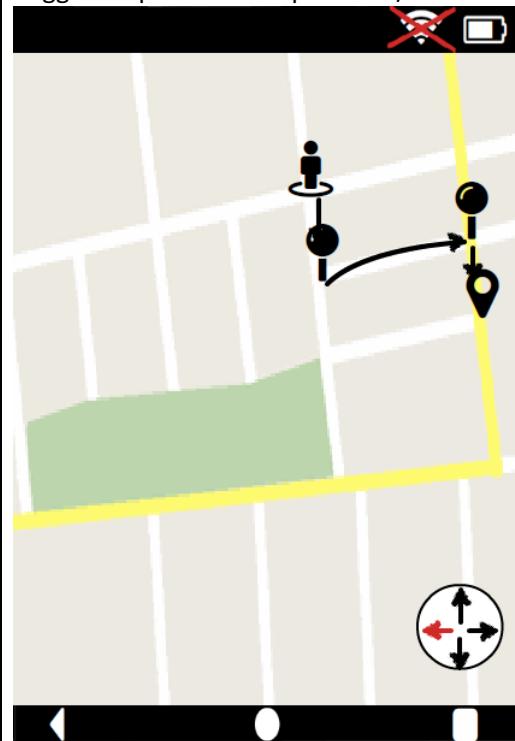
"Visualizza percorsi"



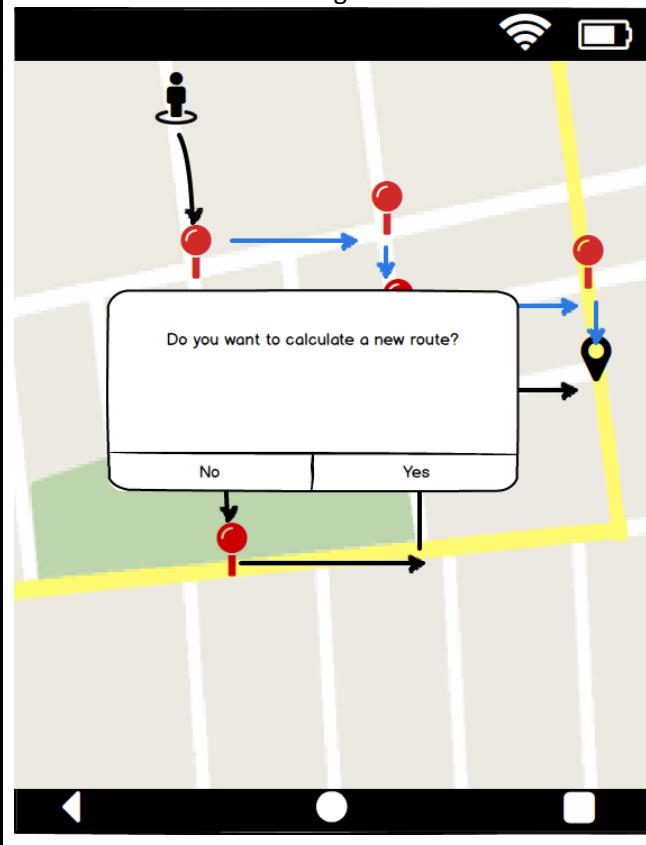
"Avvia navigazione"



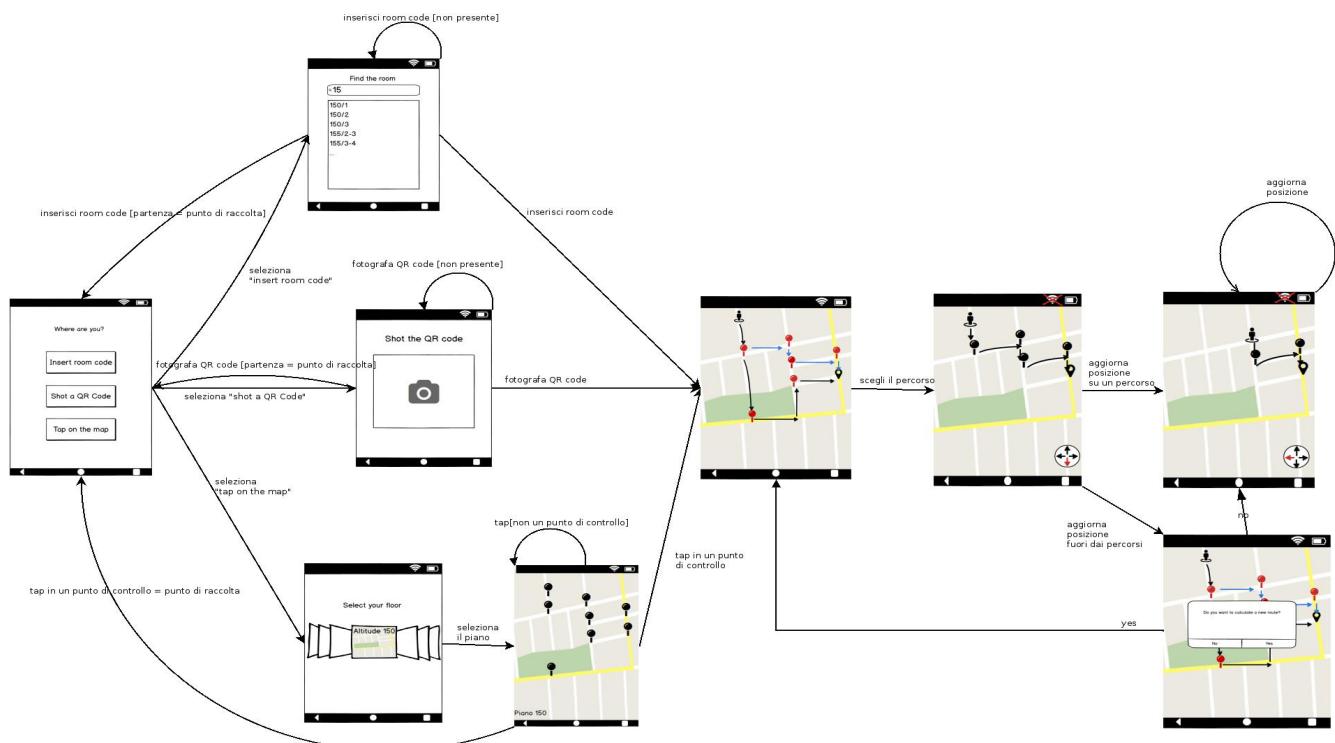
"Aggiorna posizione su percorso/i"



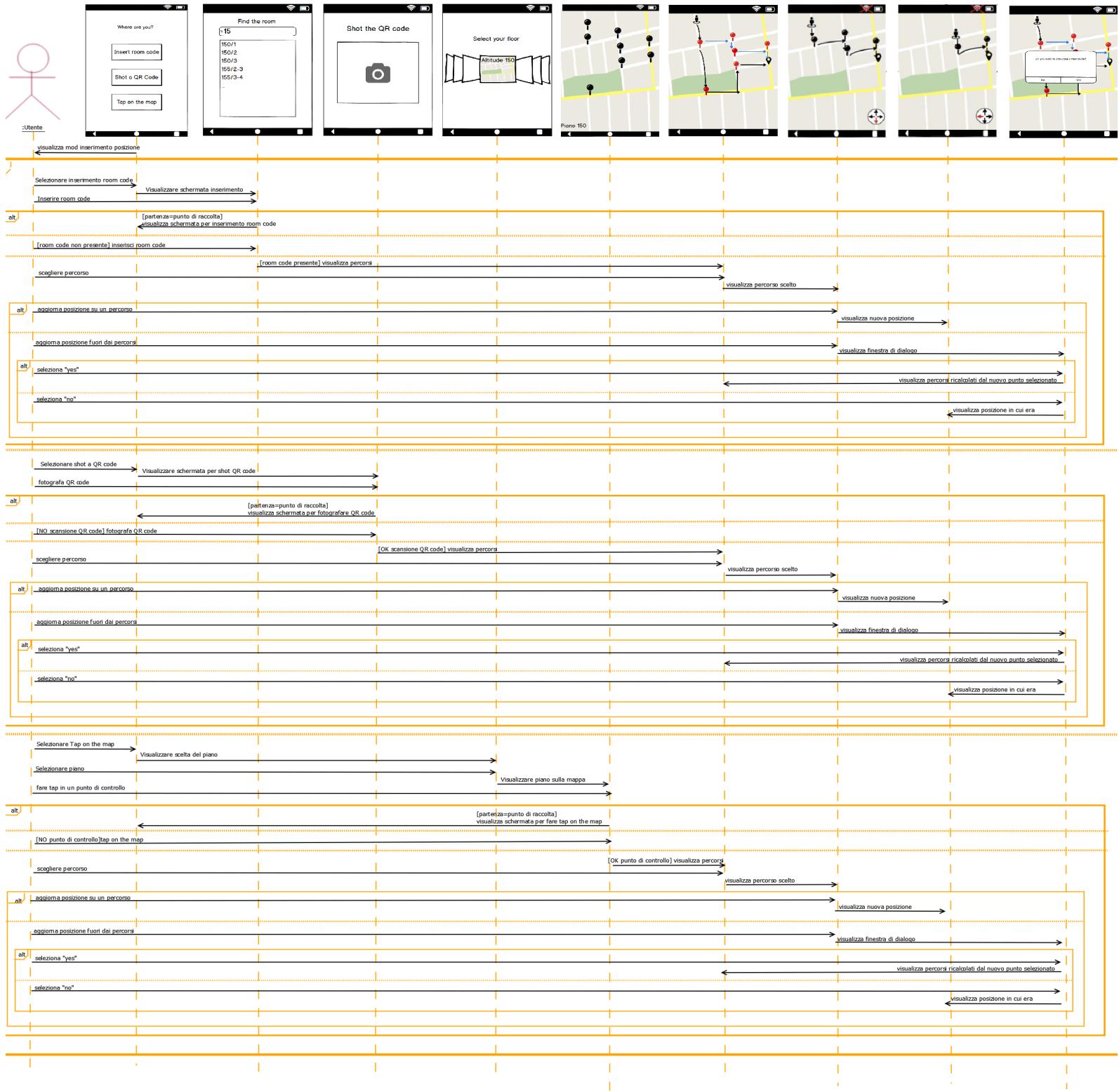
"Visualizza finestra di dialogo"



Modello di navigazione



Modello di dialogo

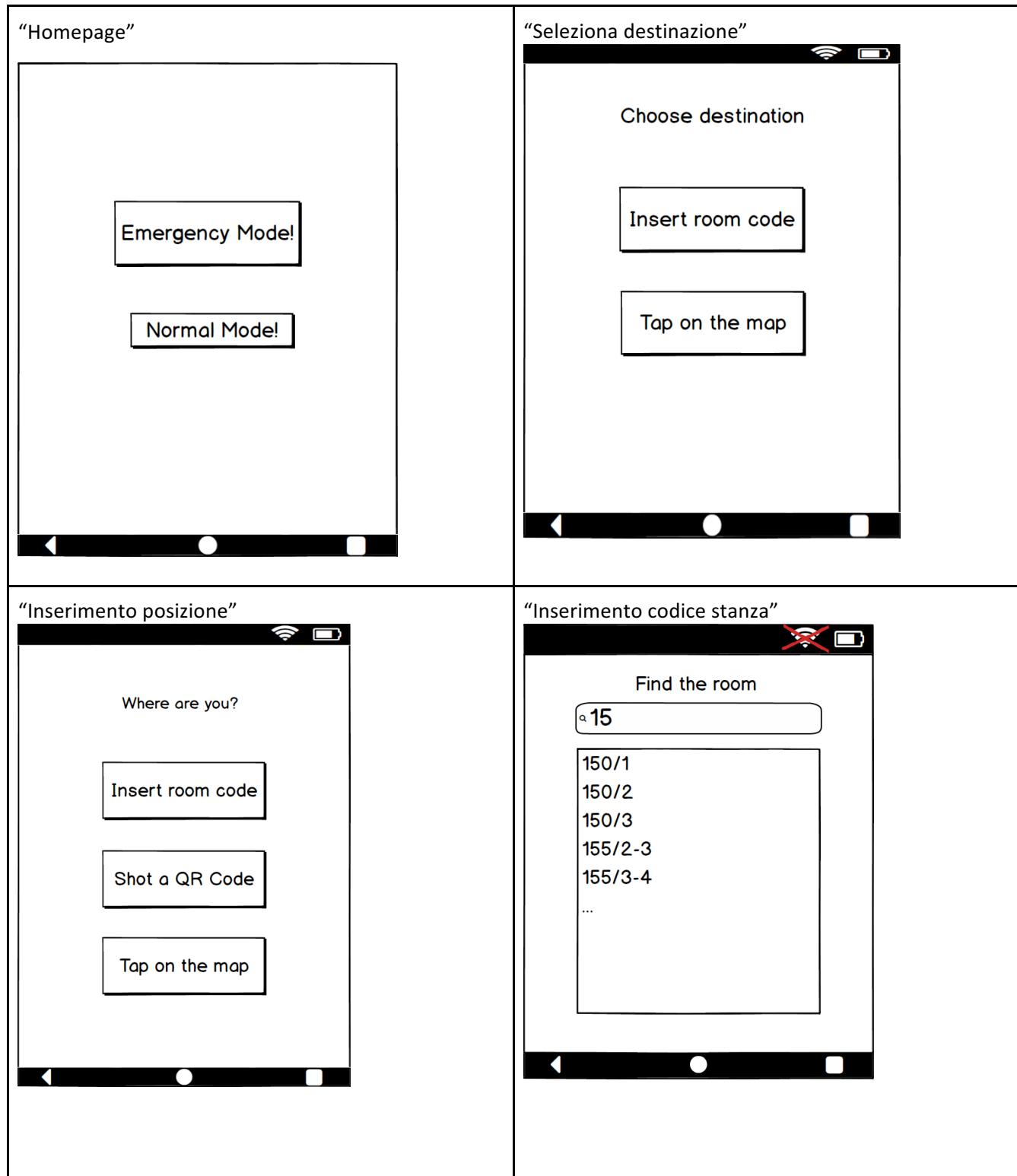


Funzionamento in condizioni normali

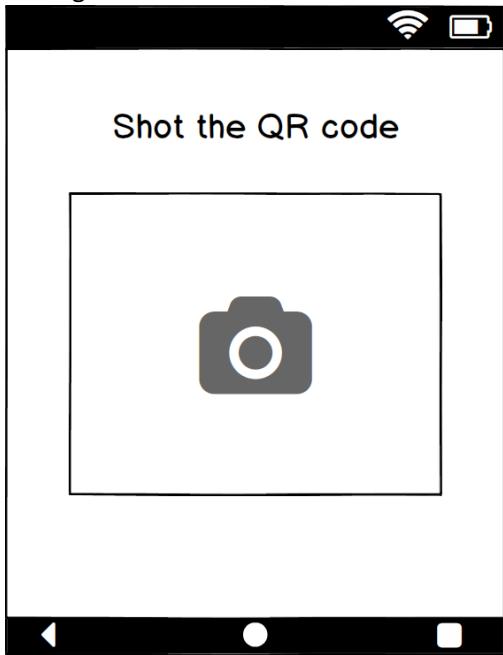
User and task Model

User Model	Utente
Task Model	<ul style="list-style-type: none">● seleziona inserimento “room code”● inserisce il codice della stanza come destinazione● premere il tasto “select”● seleziona “tap on the map”● seleziona un piano dal carousel● tap su uno dei punti di controllo presenti sul piano visualizzato, come destinazione● seleziona percorso scelto● aggiorna posizione switchando tra un percorso e l’altro o selezionando un altro punto di controllo

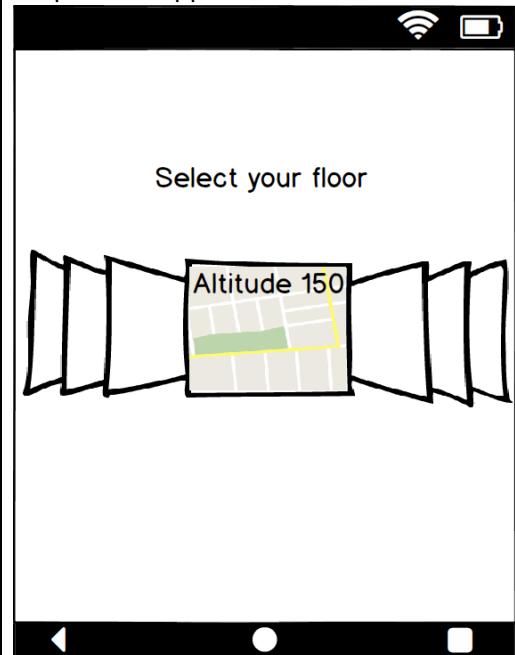
Modello di presentazione



"Fotografa QR Code"



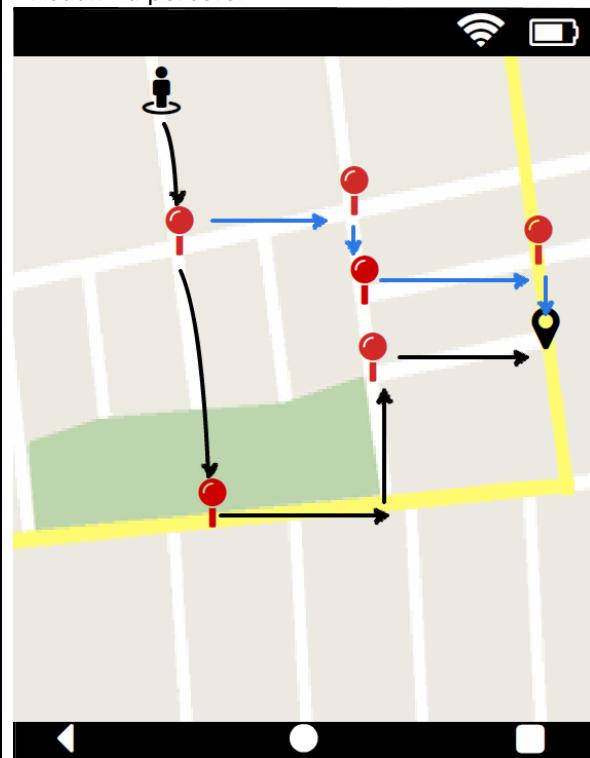
"Tap sulla mappa"



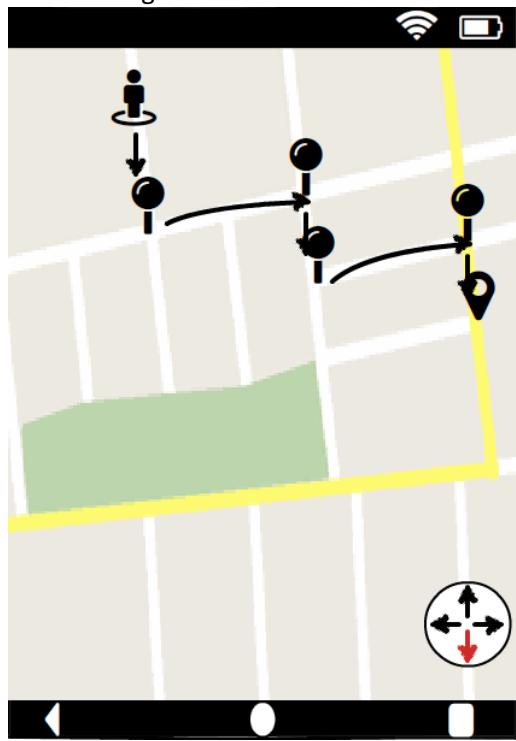
"Seleziona punto di controllo"



"Visualizza percorsi"



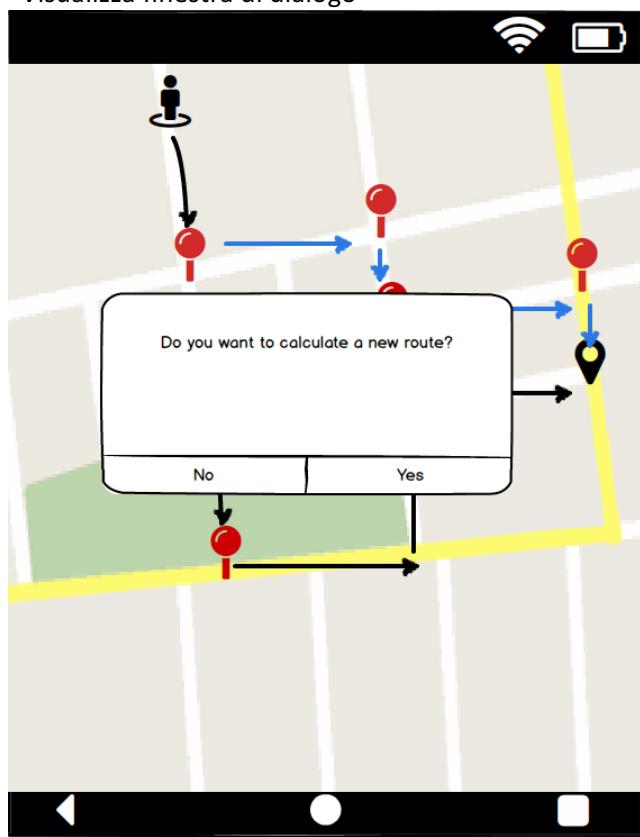
"Avvia navigazione"



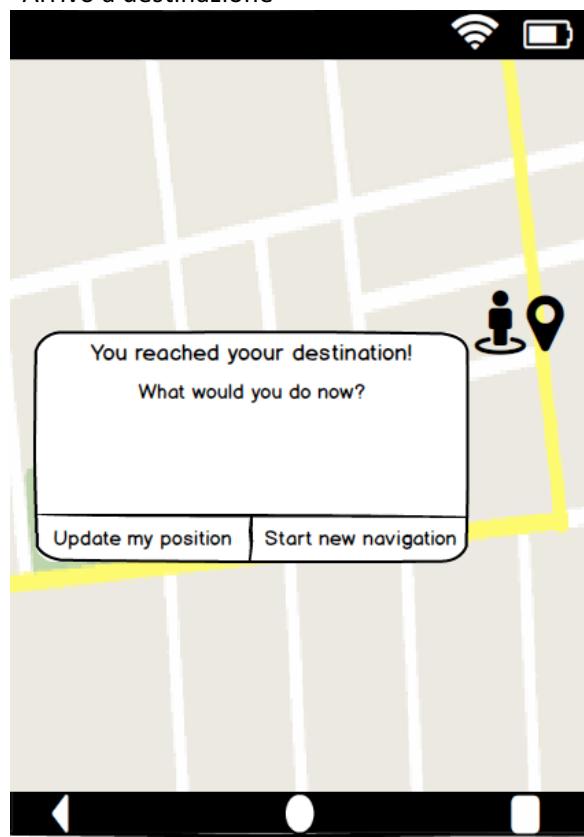
"Aggiorna posizione su percorso/i"



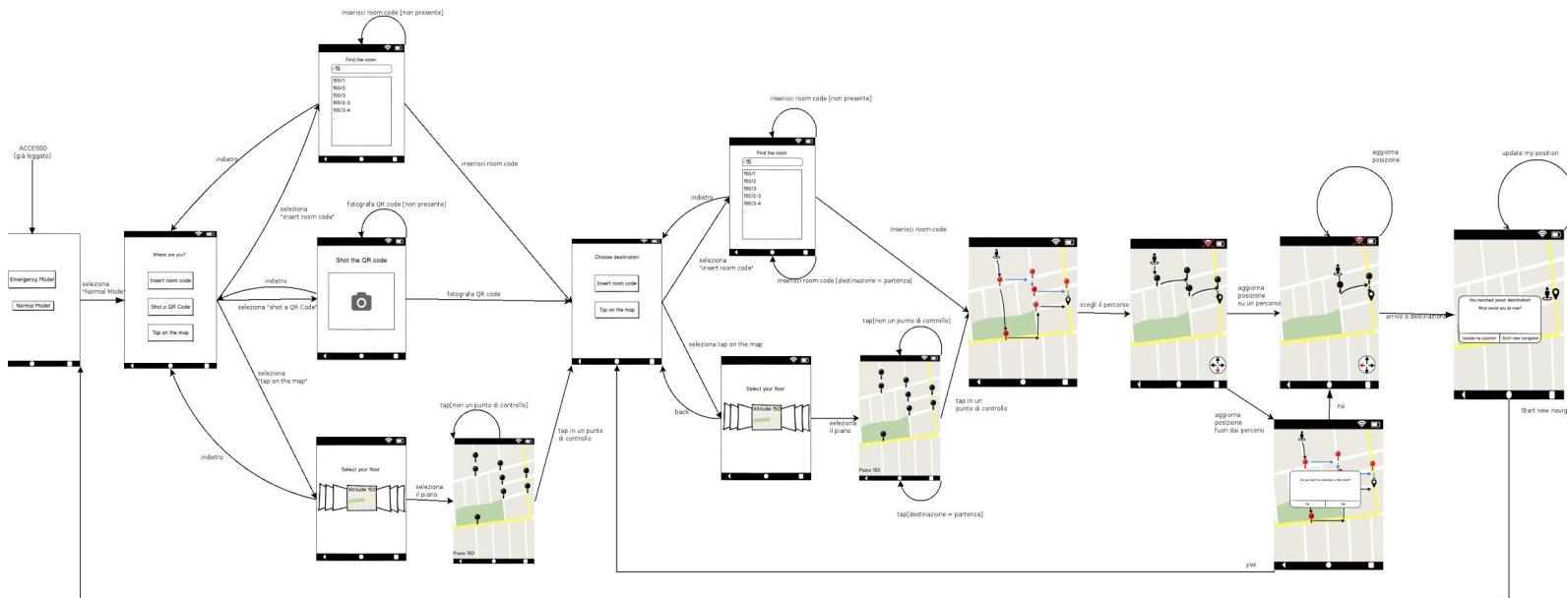
"Visualizza finestra di dialogo"



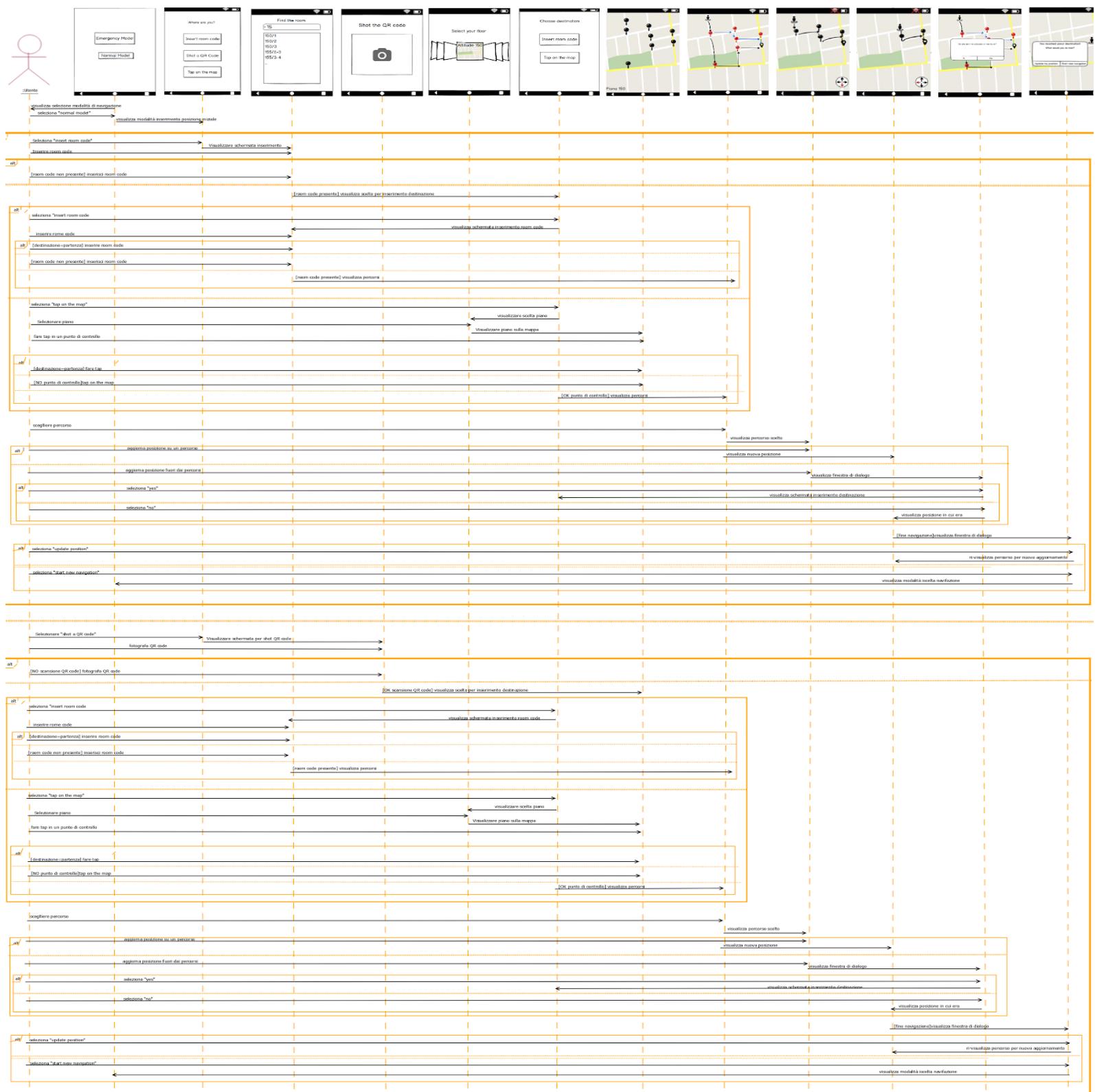
"Arrivo a destinazione"

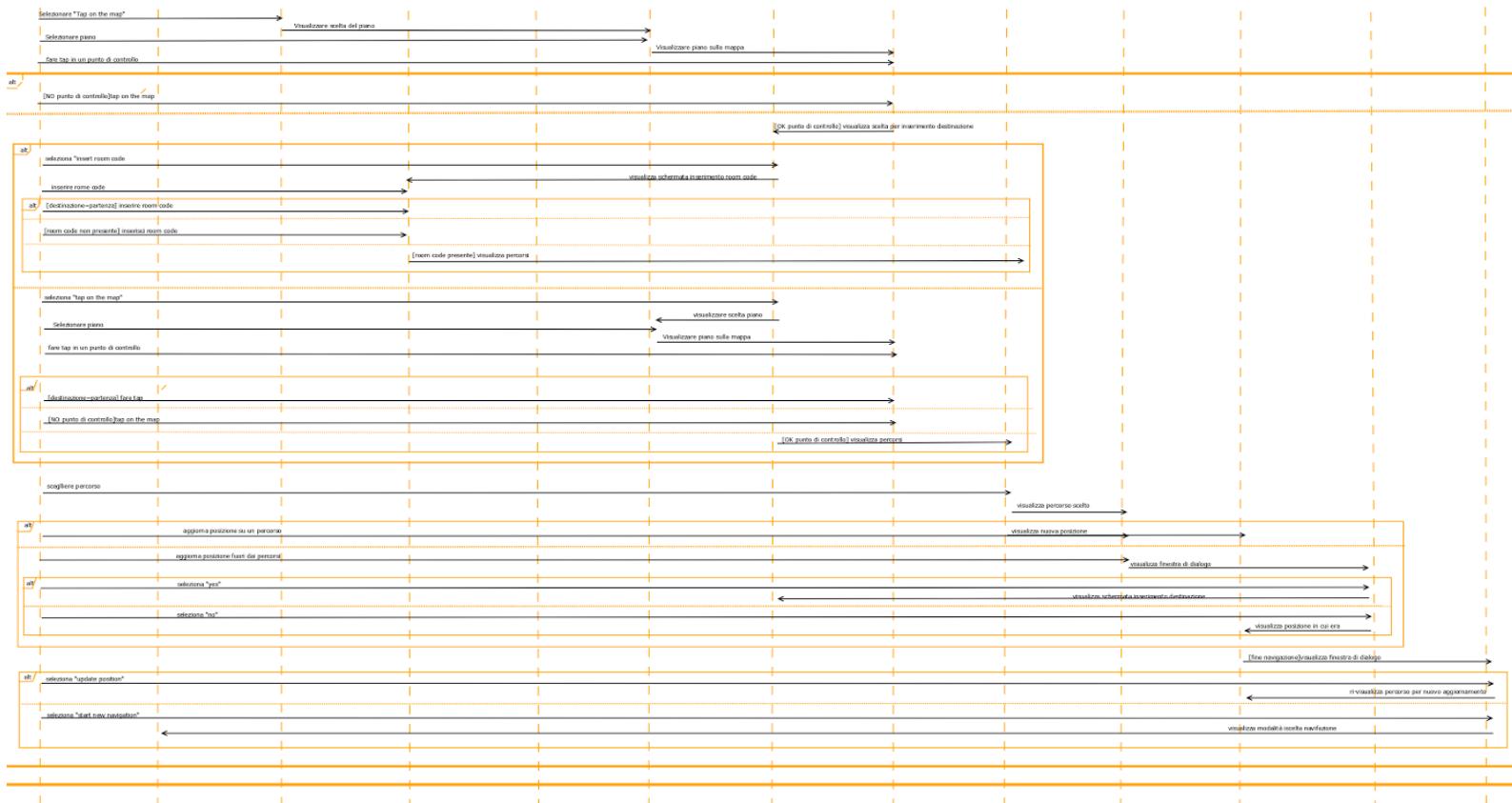


Modello di navigazione



Modello di dialogo





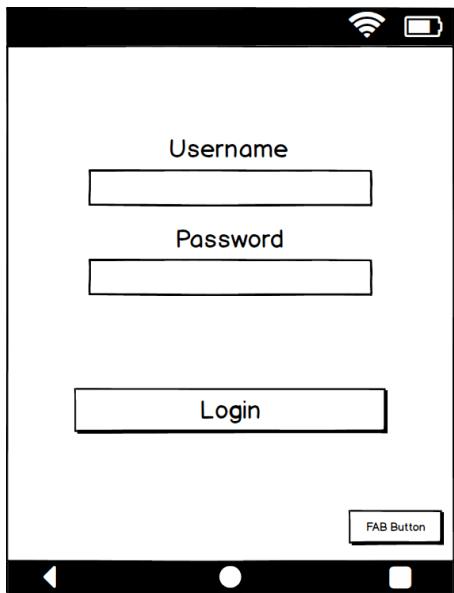
Notifica di emergenza

User and task model

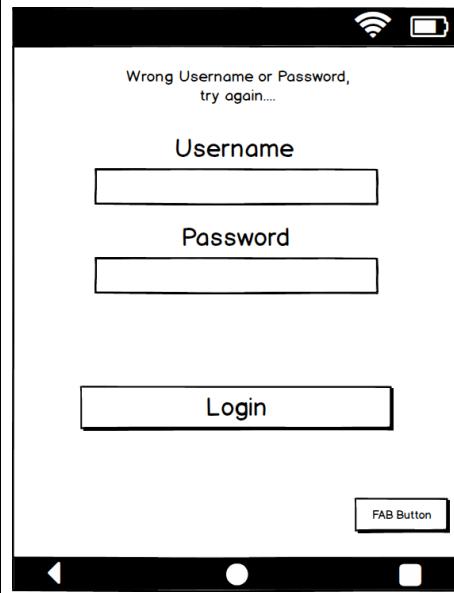
User Model	Utente
Task Model	<ul style="list-style-type: none"> selezione notifica di emergenza “Attenzione emergenza!!” Accedere alla modalità di emergenza

Modello di presentazione

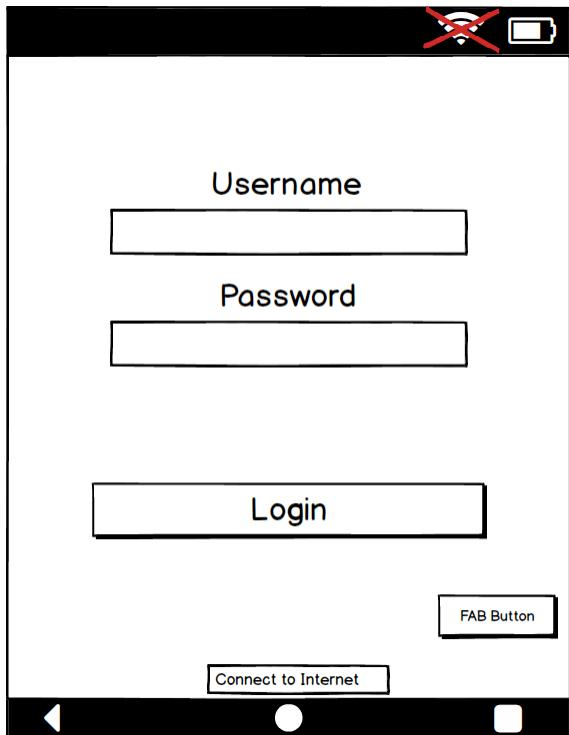
"Schermata di login"



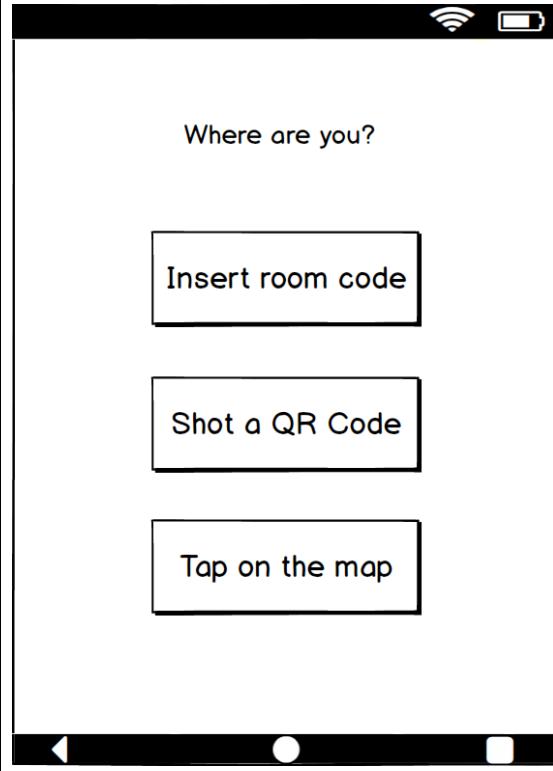
"Credenziali errate"



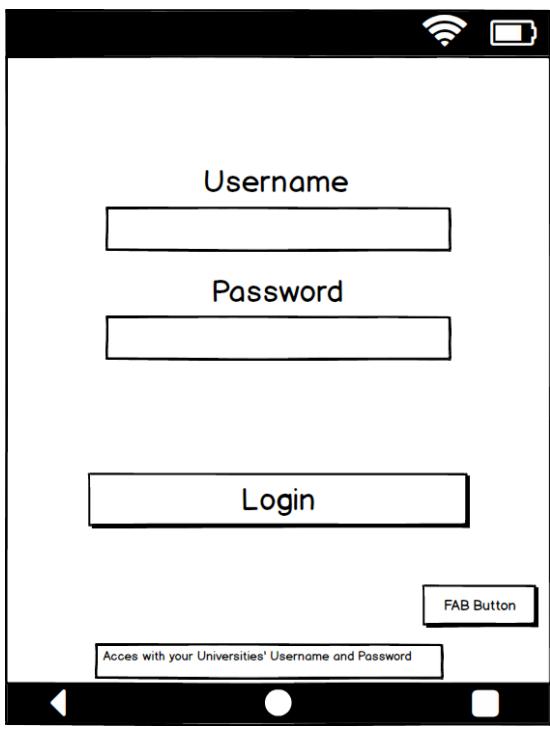
"Utente preme su FAB button, non collegato alla rete"



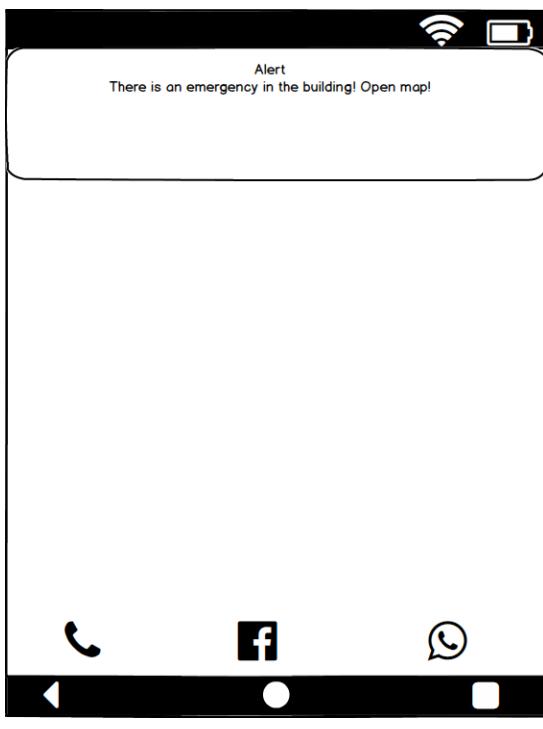
"Inserimento posizione"



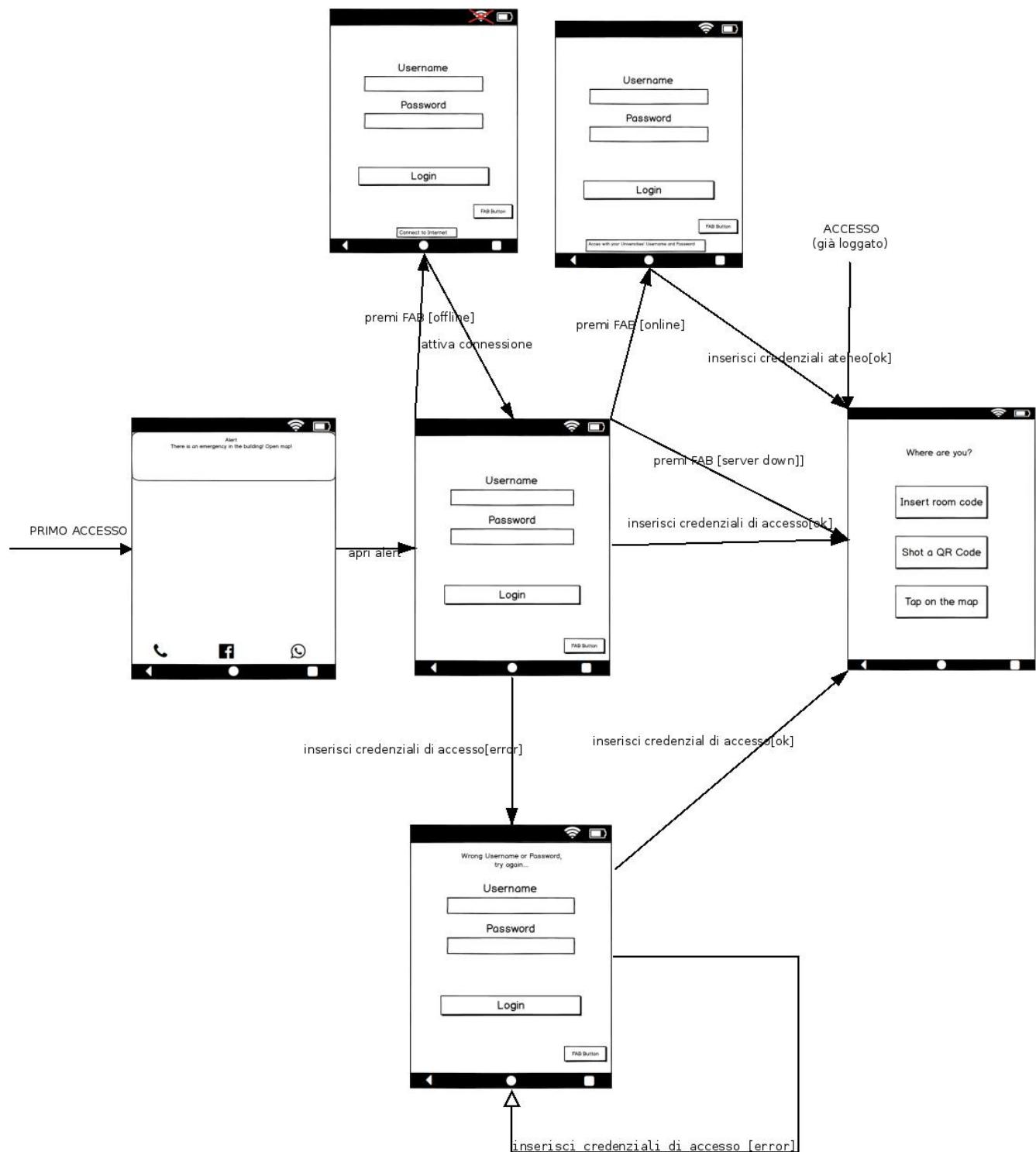
"Utente preme su FAB button, server online"



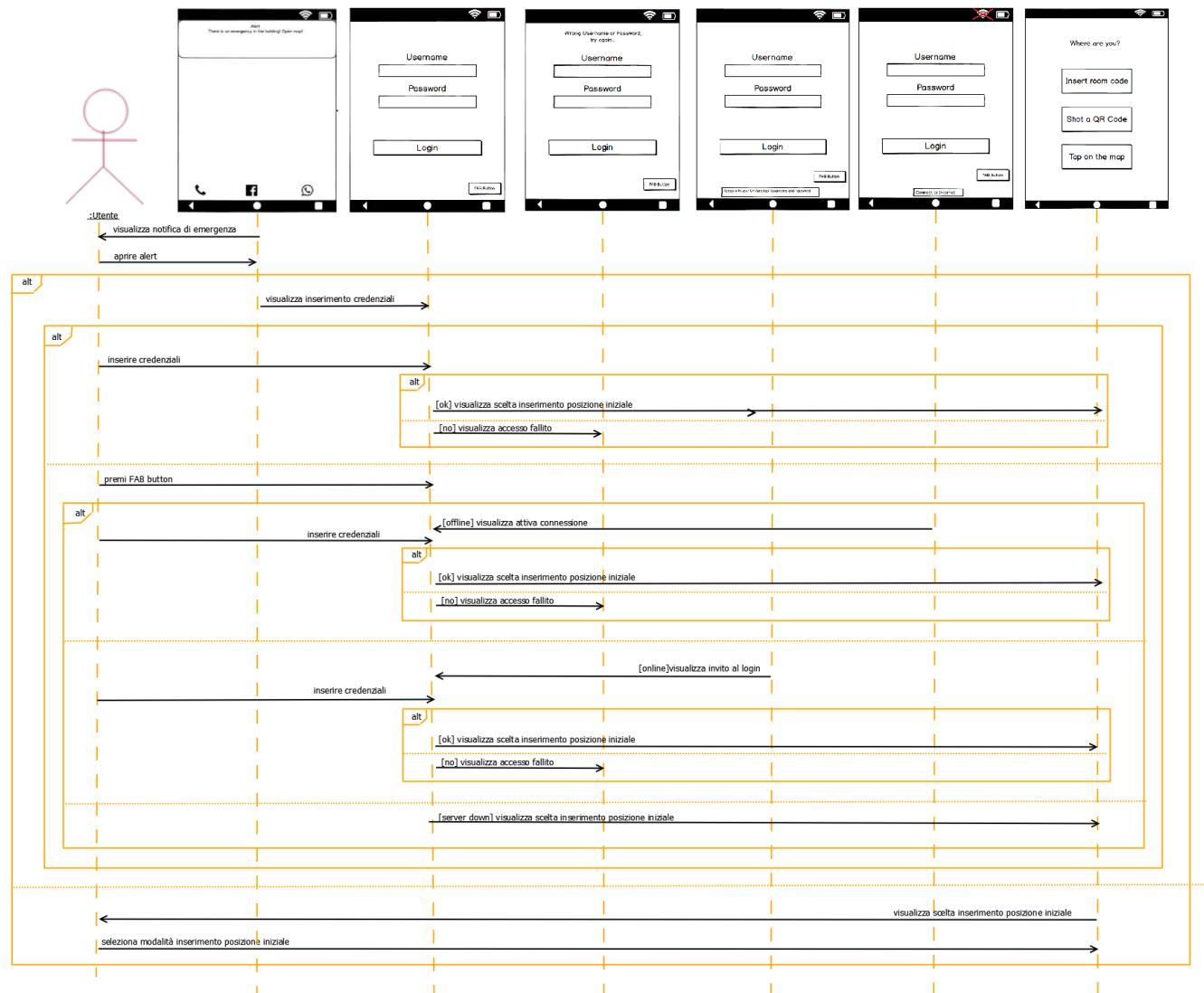
"Notifica di emergenza"



Modello di navigazione



Modello di dialogo

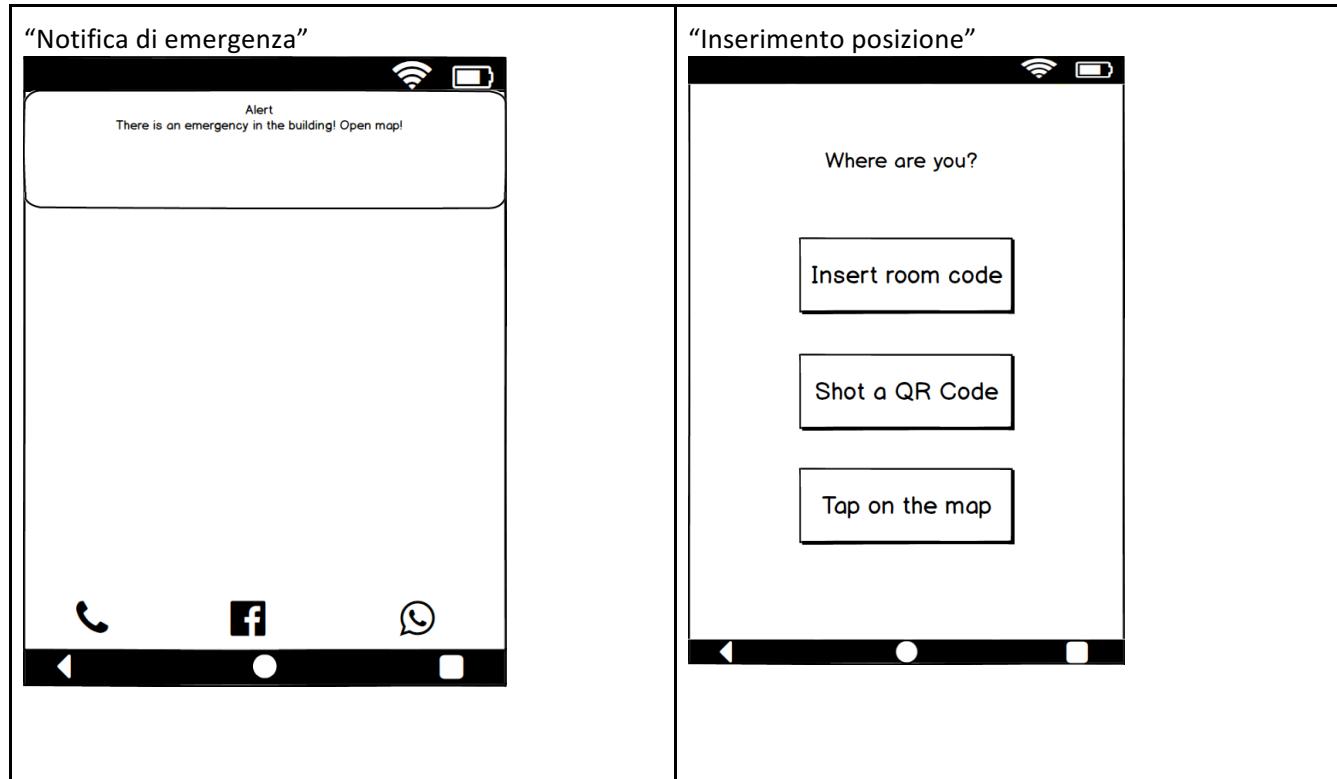


Guida al punto di raccolta in modalità online

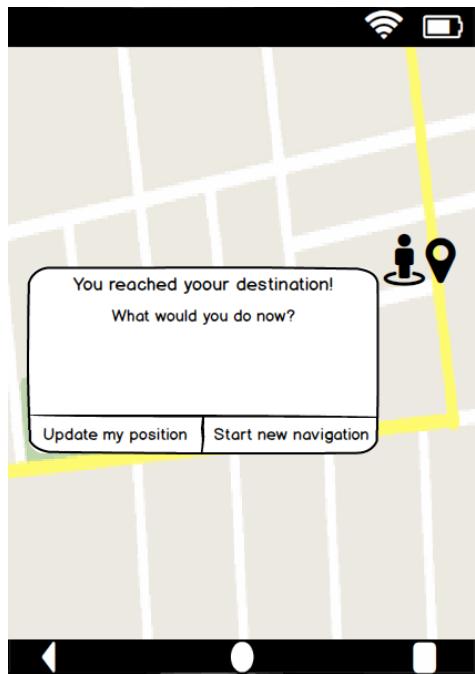
User and task Model

User Model	Utente
Task Model	
	<ul style="list-style-type: none">• seleziona la modalità di navigazione “Emergency mode”• seleziona la modalità di inserimento della posizione iniziale• inserisce posizione iniziale• seleziona il percorso• aggiorna posizione switchando tra un percorso e l’altro o selezionando un altro punto di controllo

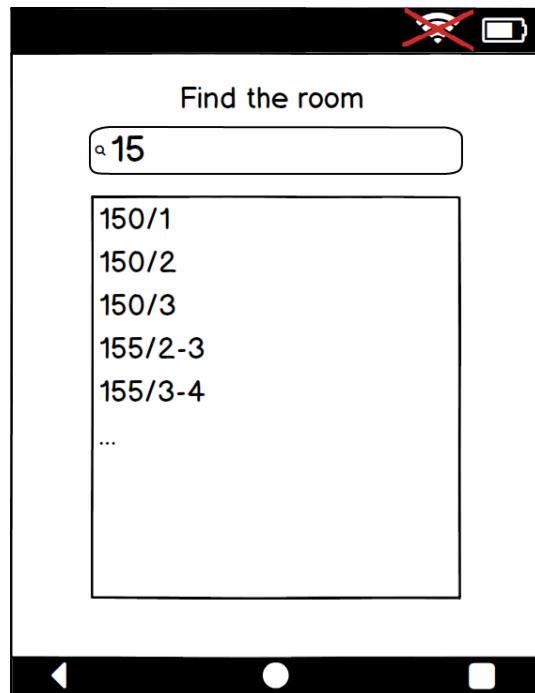
Modello di presentazione



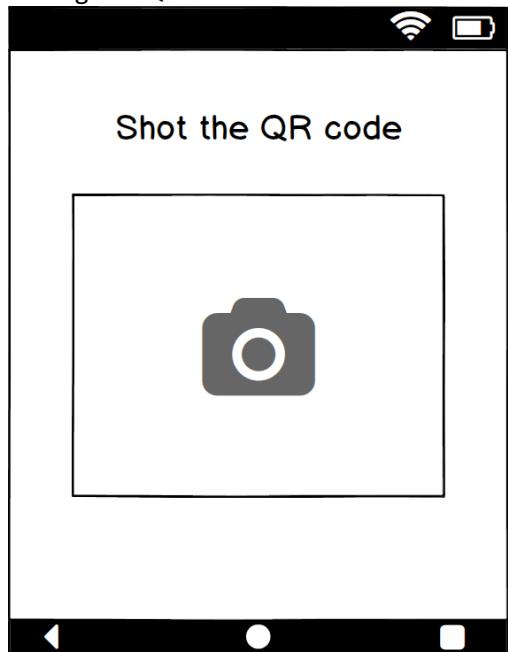
"Arrivo a destinazione"



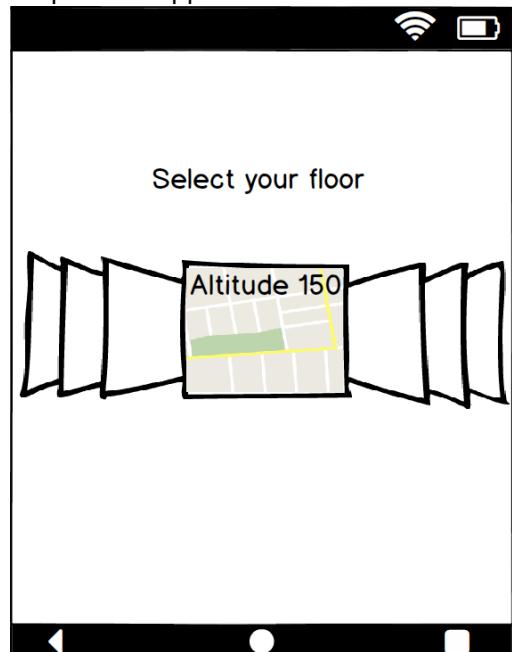
"Inserimento codice stanza"



"Fotografa QR Code"



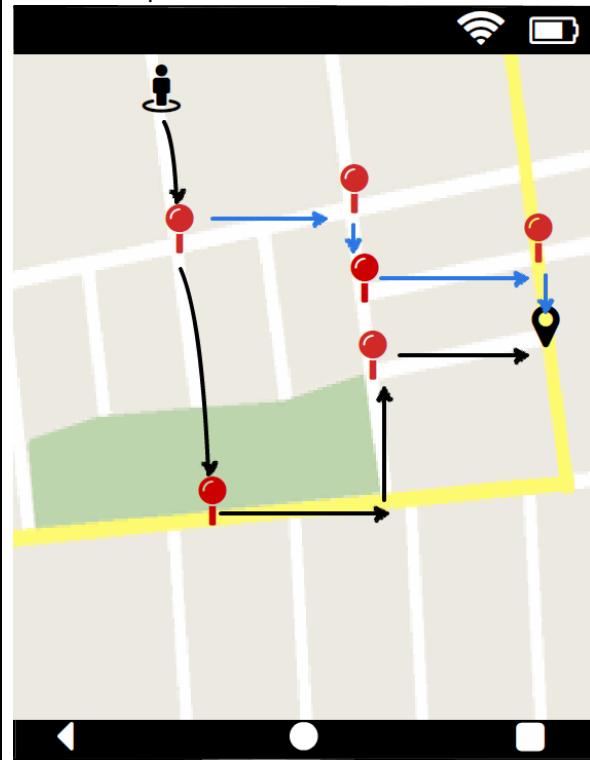
"Tap sulla mappa"



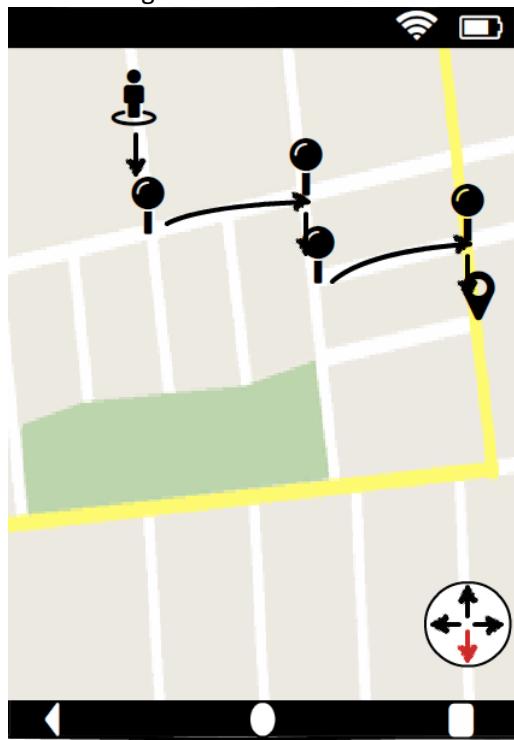
"Selezione punto di controllo"



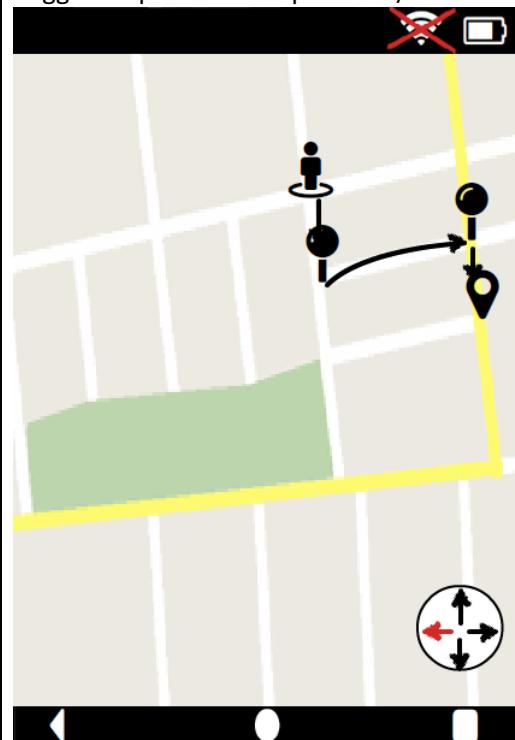
"Visualizza percorsi"



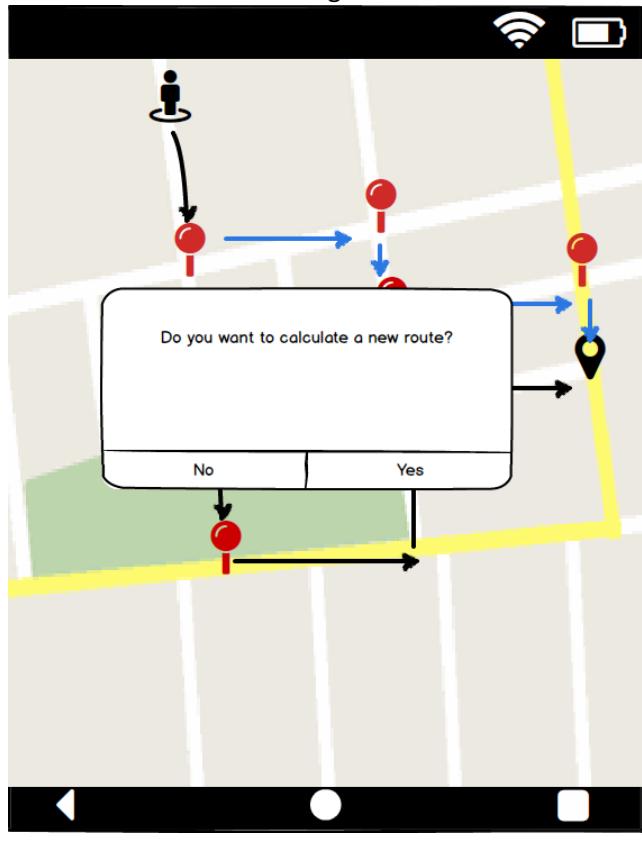
"Avvia navigazione"



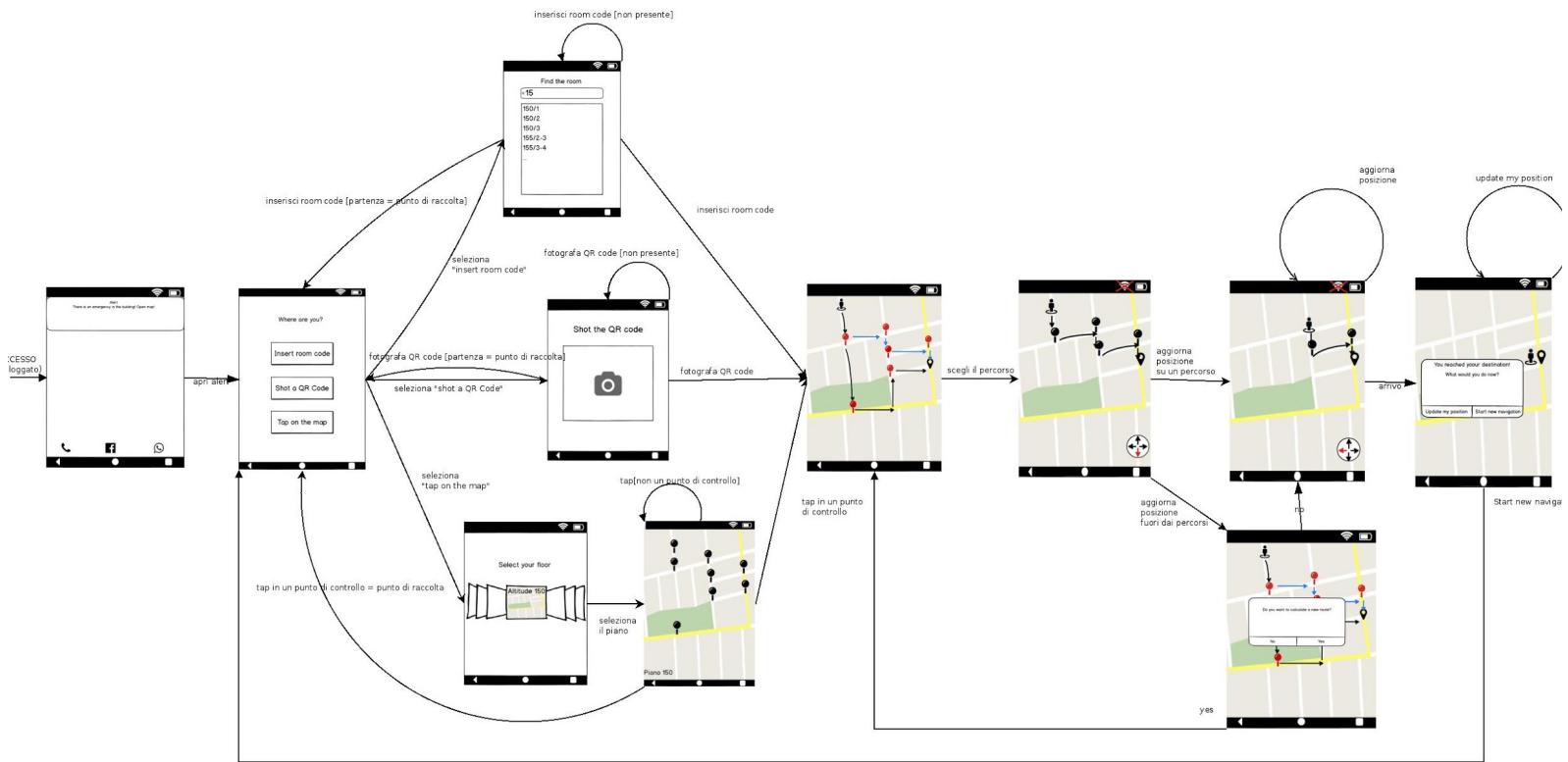
"Aggiorna posizione su percorso/i"



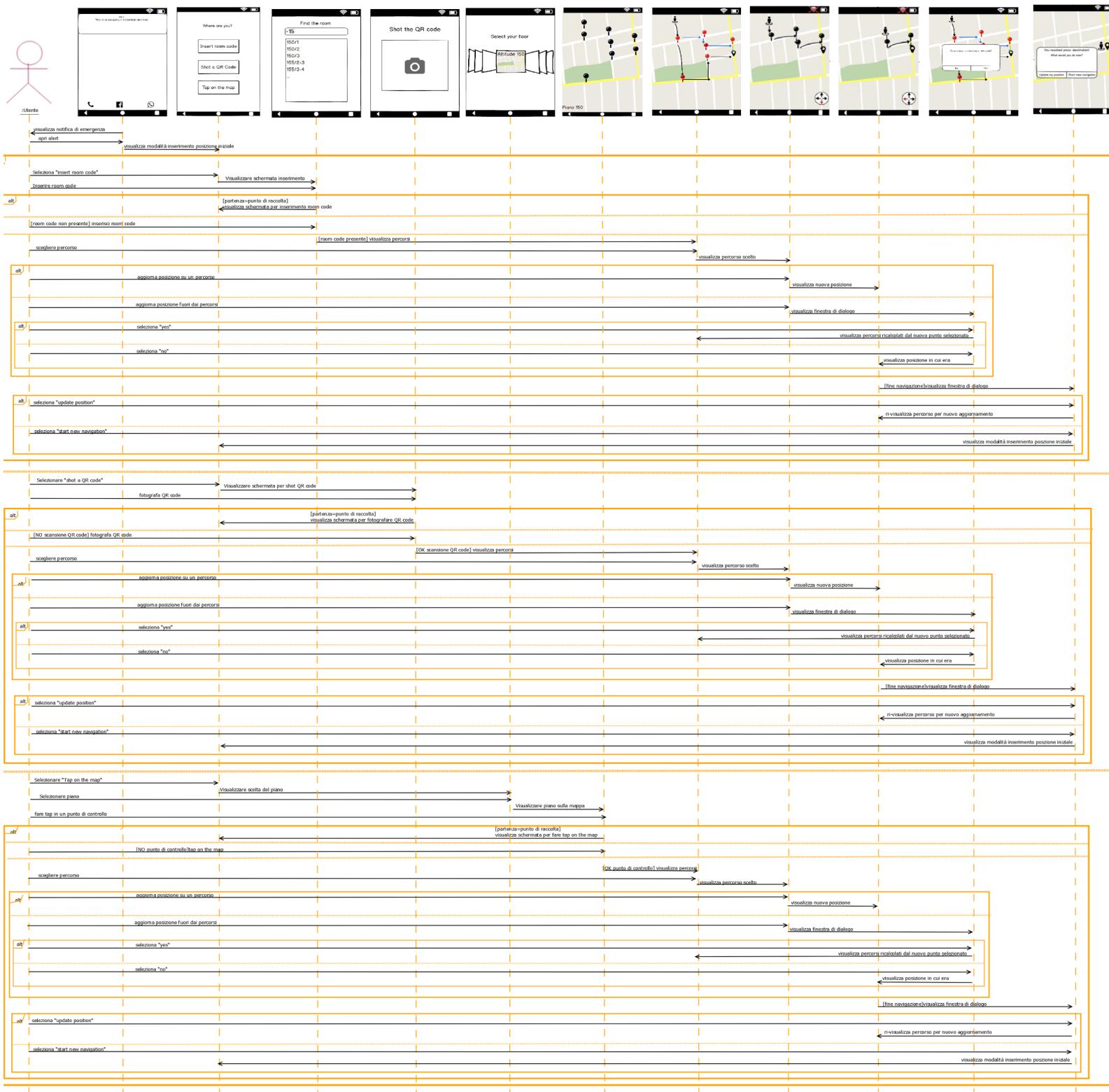
"Visualizza finestra di dialogo"



Modello di navigazione



Modello di dialogo

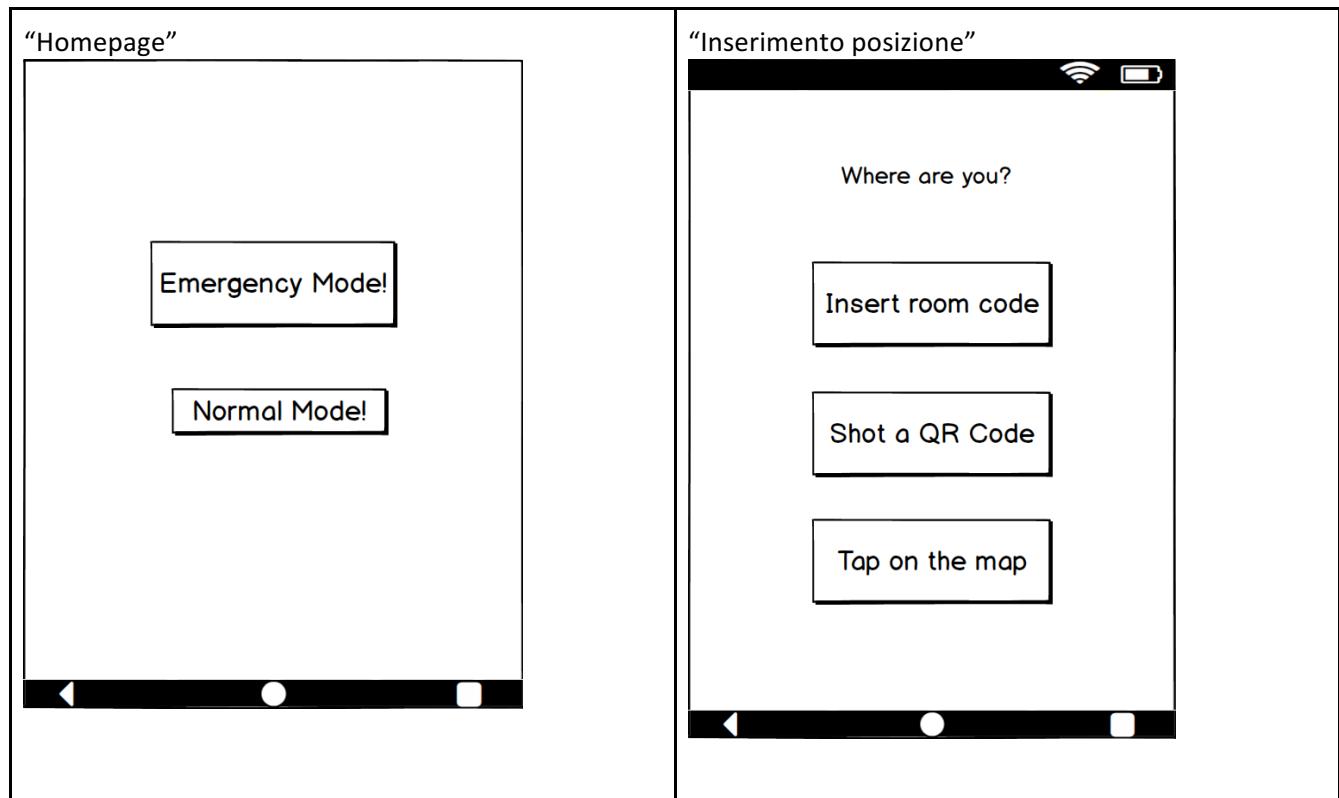


Guida al punto di raccolta in modalità offline

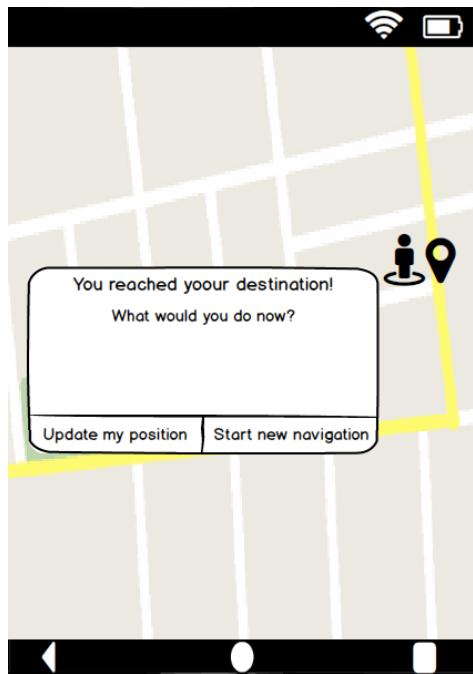
User and task model

User Model	Utente
Task Model	
	<ul style="list-style-type: none">• seleziona la modalità di navigazione “Emergency mode”• seleziona la modalità di inserimento della posizione iniziale• inserisce posizione iniziale• seleziona il percorso• aggiorna posizione switchando tra un percorso e l’altro o selezionando un altro punto di controllo

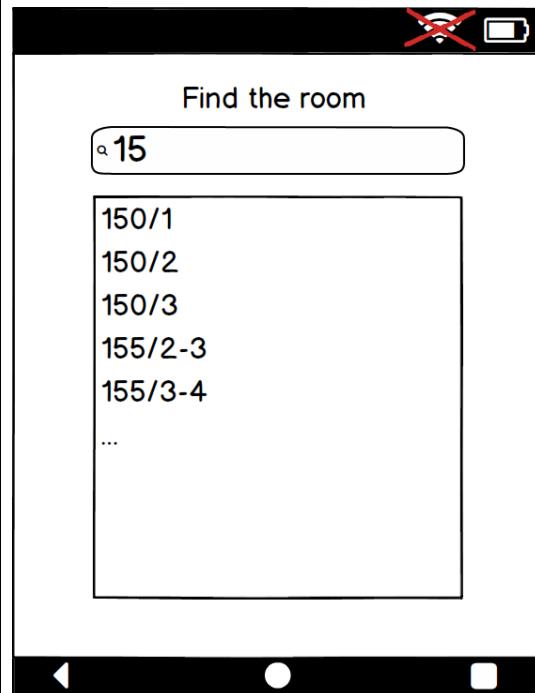
Modello di presentazione



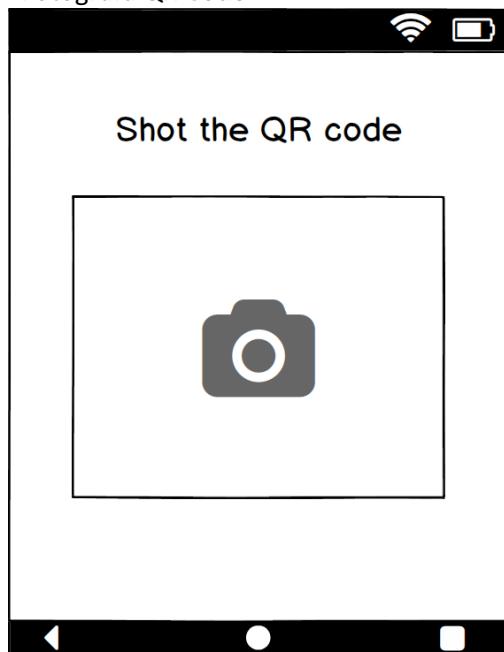
"Arrivo a destinazione"



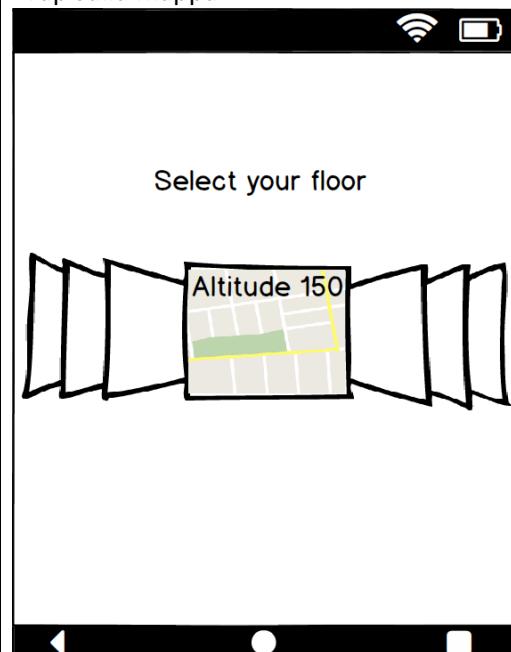
"Inserimento codice stanza"



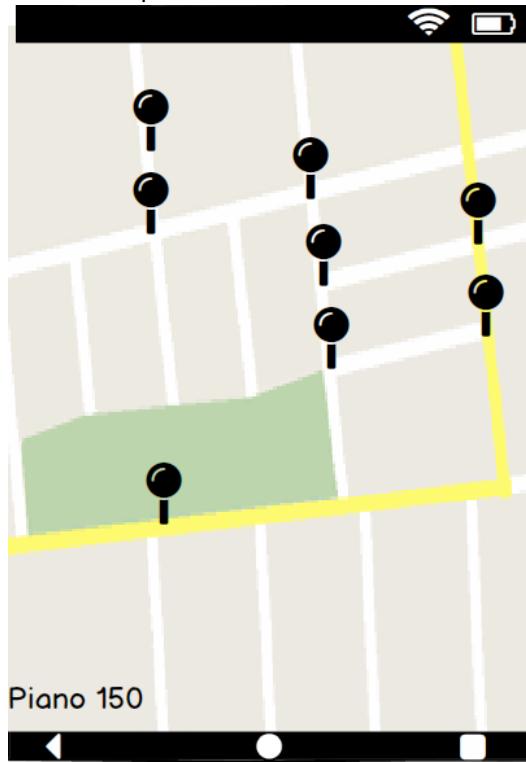
"Fotografa QR Code"



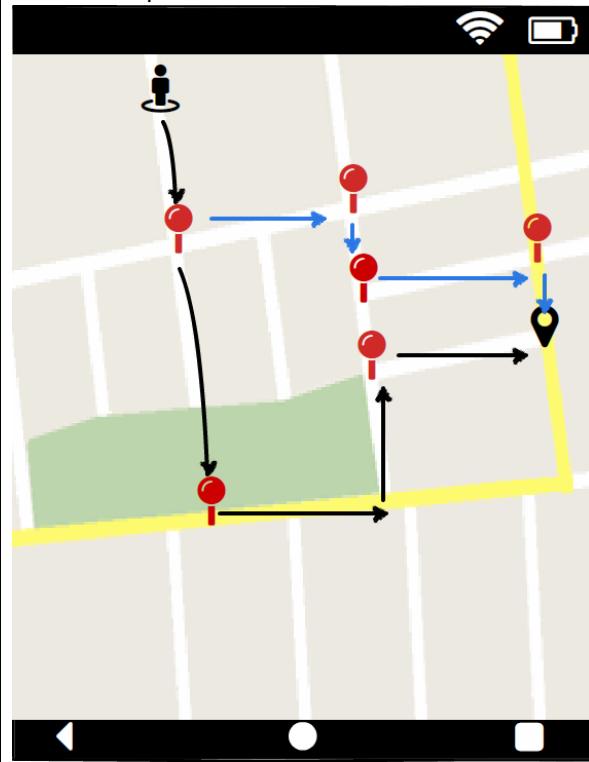
"Tap sulla mappa"



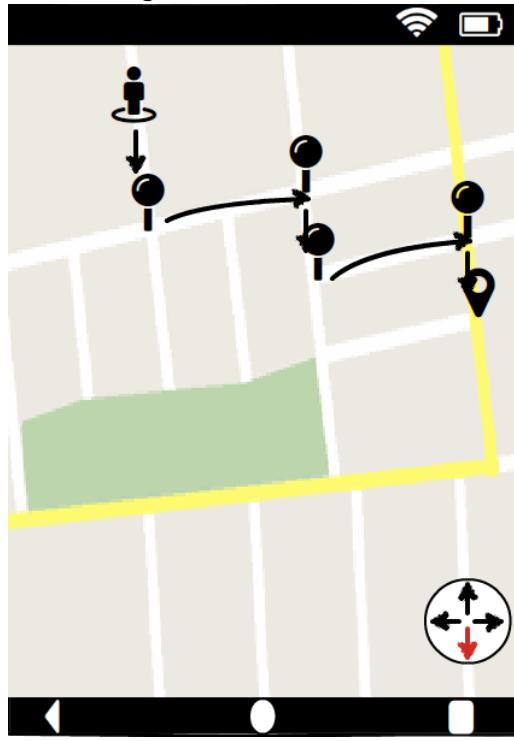
"Selezione punto di controllo"



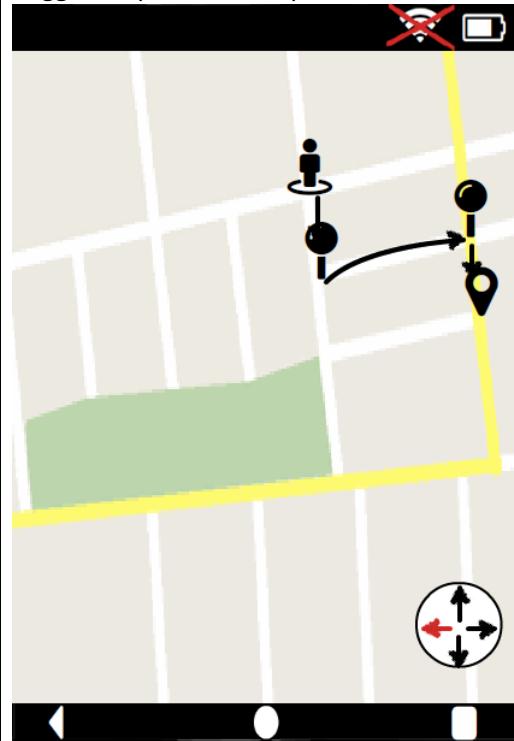
"Visualizza percorsi"



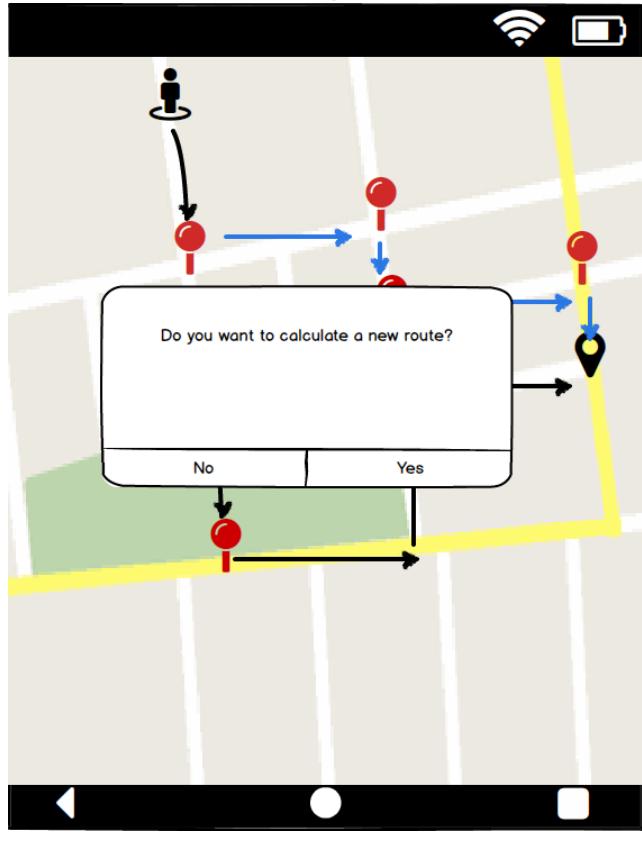
"Avvia navigazione"



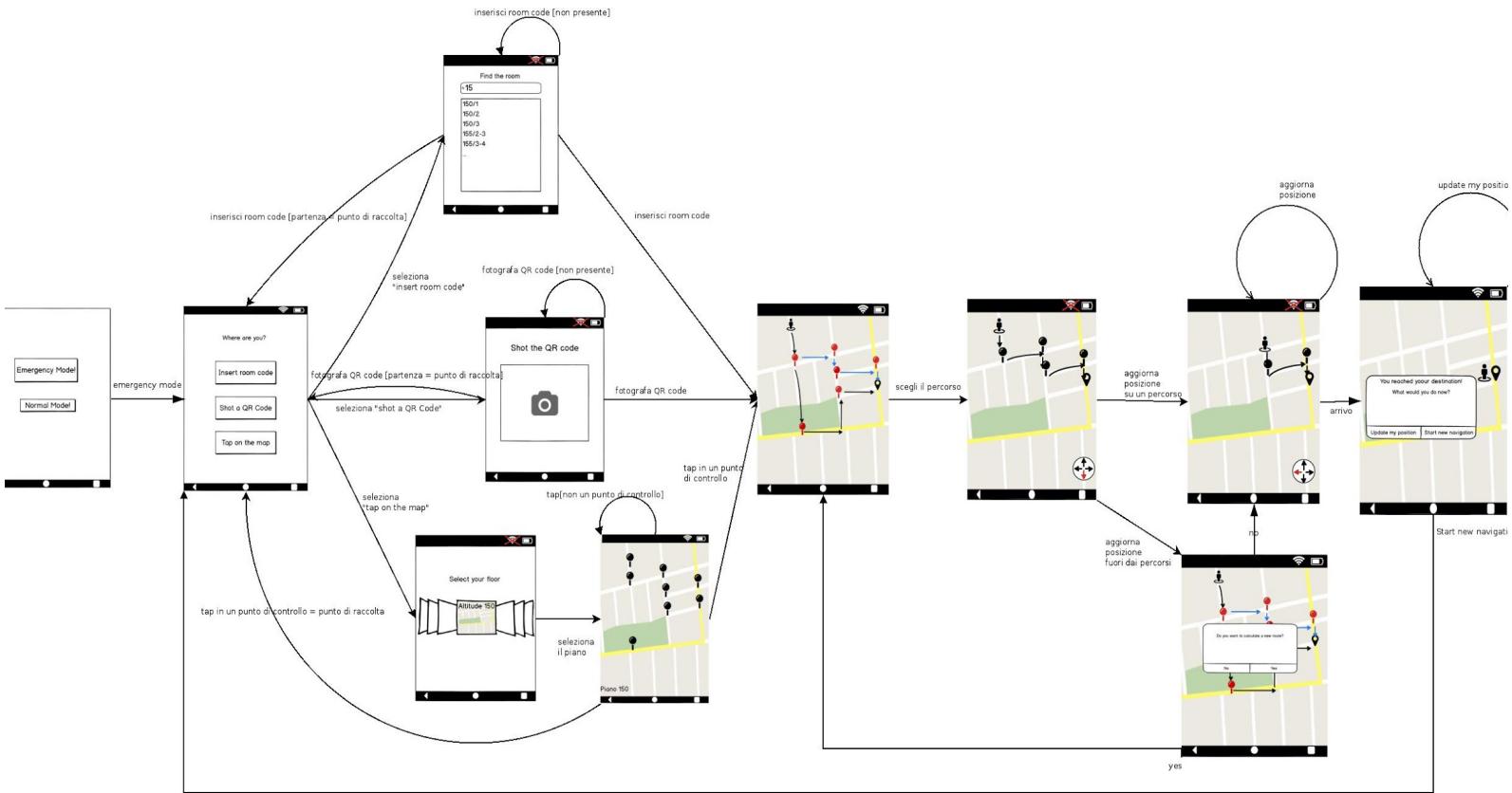
"Aggiorna posizione su percorso/i"



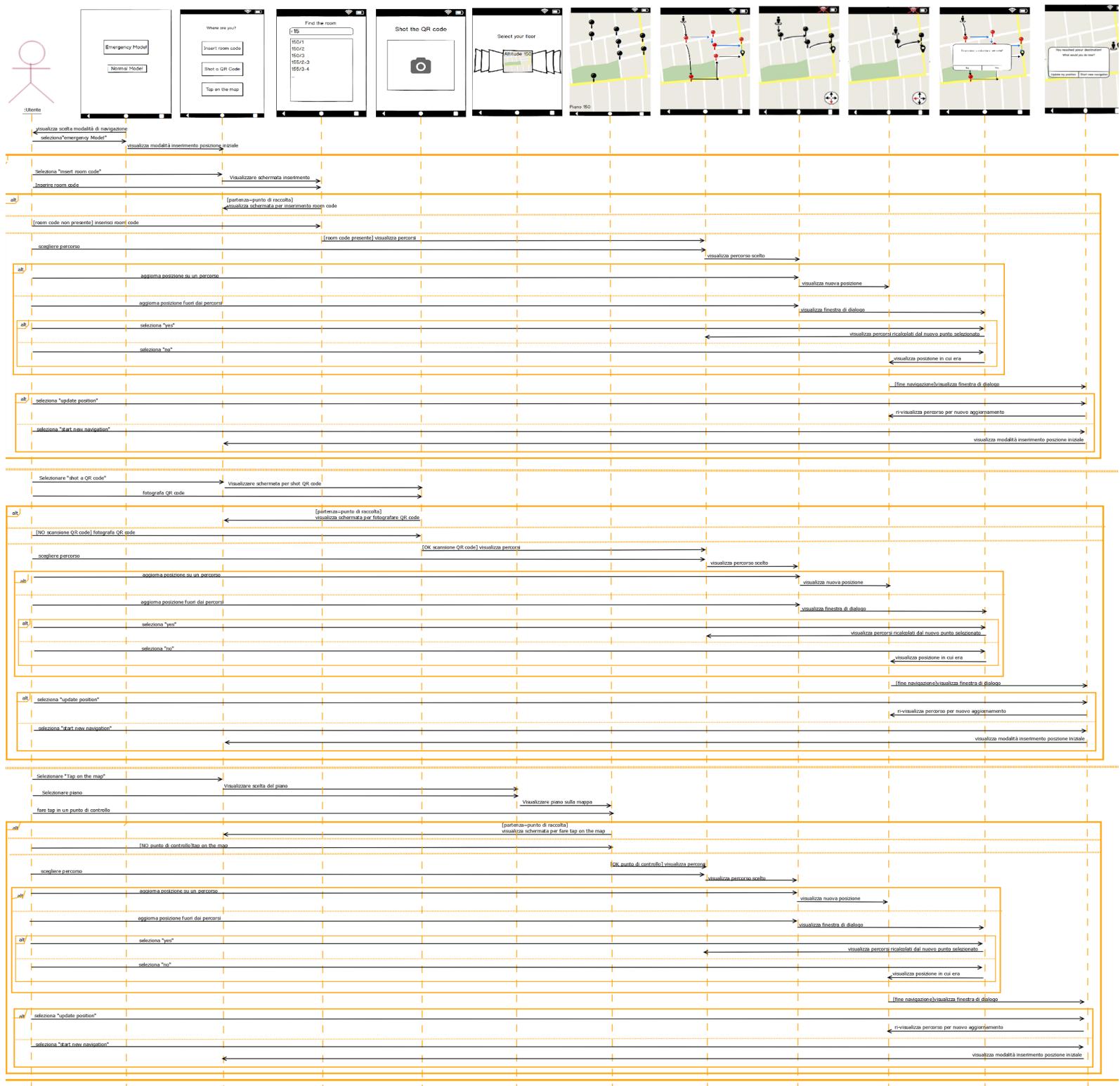
"Visualizza finestra di dialogo"



Modello di navigazione



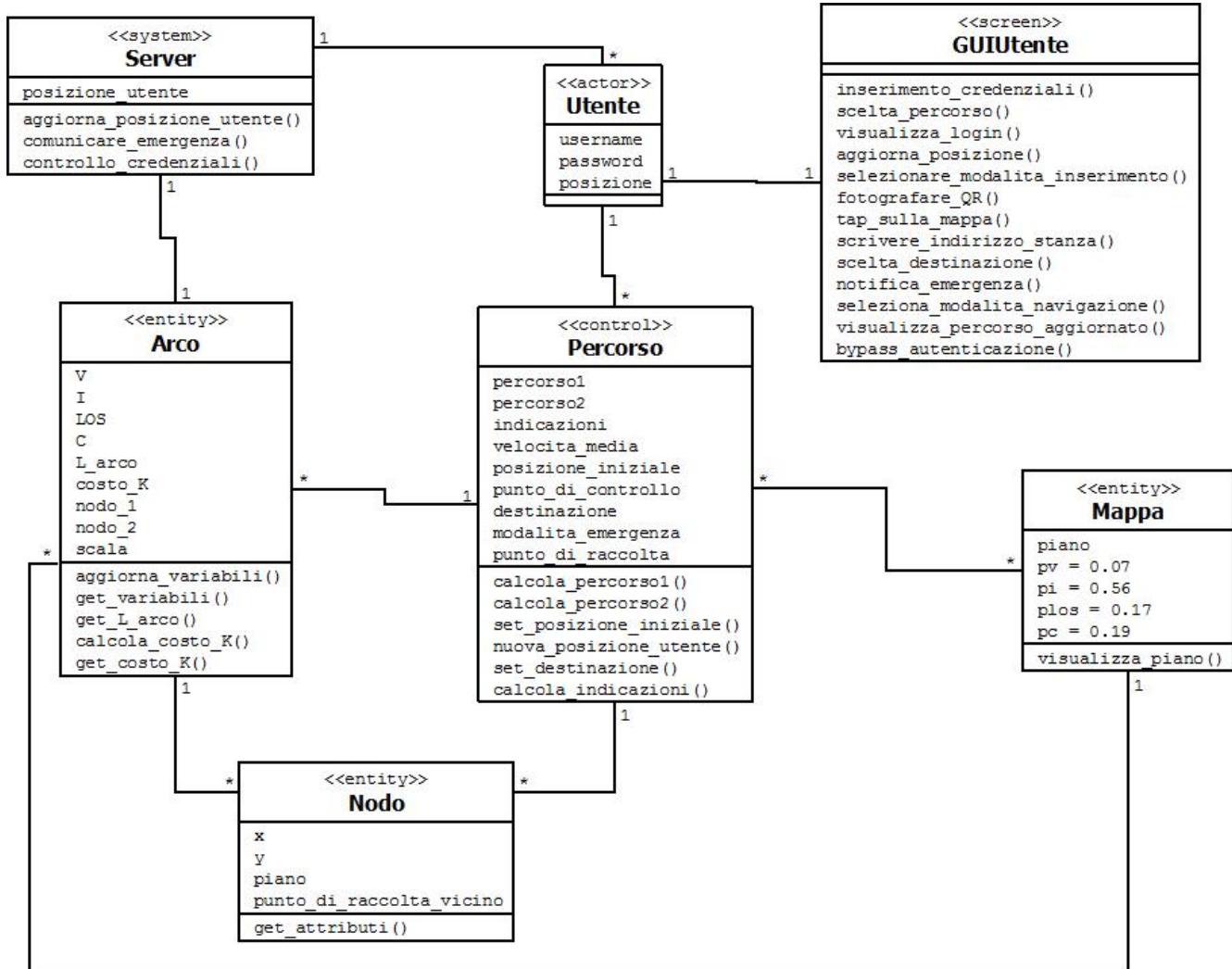
Modello di dialogo



4.3 Analisi orientata ai dati

4.3.1 Diagramma delle classi

Per costruire il diagramma delle classi di analisi si sono analizzate nuovamente le schede CRC prodotte, cercando di individuare tutte le componenti del modello. Seguendo la teoria, per ogni scheda e per ogni collaboratore si è stabilito se questi fossero da interpretare come classi o come attributi, questa classificazione è stata dedotta confrontando il numero e le tipologie di responsabilità che l'oggetto portava in dote assieme al concetto che esso modellava. Lo schema prodotto da questi ragionamenti è stato il seguente.

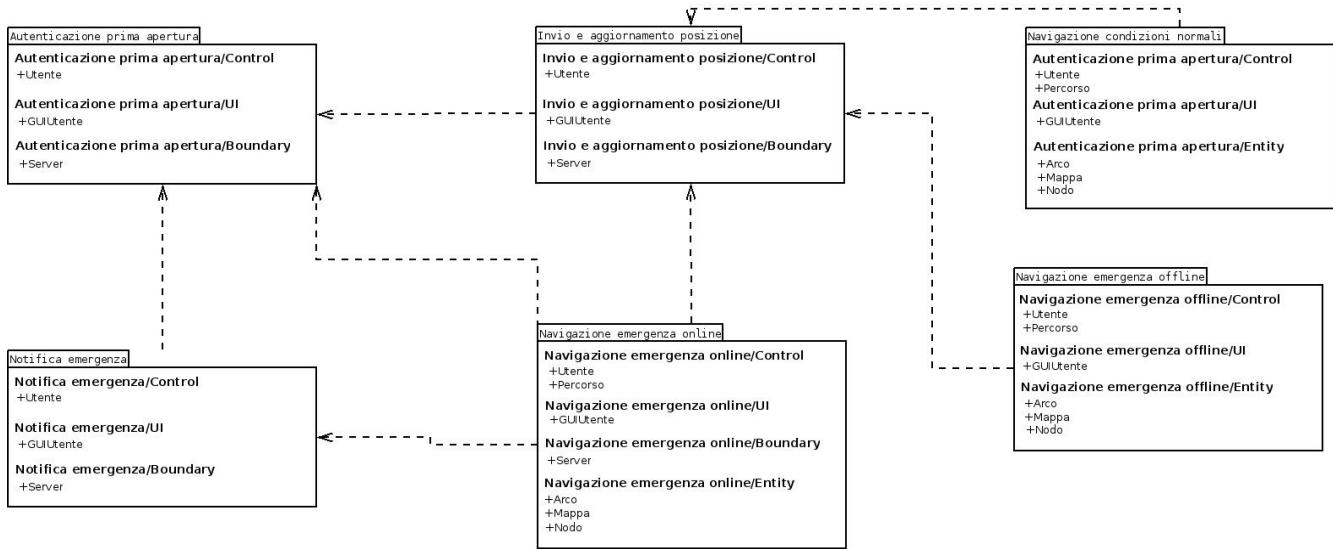


Come si nota a partire da tutte le user stories e da tutte le schede CRC si sono individuate 7 classi principali che, a nostro avviso, modellano altrettanti concetti fondamentali per l'applicazione. Anzitutto l'**Utente** viene definito da delle credenziali di accesso e da una posizione nello spazio, i cui metodi sono stati assegnati alla relativa GUI. Di seguito la classe **GUIUtente** raccoglie tutti i metodi che presuppongono un'interazione, essa offre all'utente gli strumenti per interagire e portare a compimento i vari task: partendo dalla prima autenticazione e alla possibilità di bypassarla, passando per la scelta della modalità di navigazione e di selezione della propria posizione, per arrivare alla visualizzazione e scelta dei percorsi migliori calcolati. Per completare il quadro delle classi che modellano un'interazione esterna rimane la classe **Server**, che possiede come attributo la posizione dell'utente, la quale sarà aggiornata ogni volta che questo la indicherà in maniera esplicita all'applicazione. Inoltre vi sono tre metodi che servono rispettivamente ad aggiornare l'attributo della posizione utente, a comunicare lo stato di emergenza e infine quello deputato al riconoscimento dell'utente. Per quanto riguarda la meccanica interna dell'applicazione, la classe principale risulta sicuramente essere **Percorso**. Al suo interno i diversi attributi rivestono un ruolo chiave per tutte le procedure di navigazione: percorso1 e percorso2 rappresentano il percorso più breve e il secondo più breve; indicazioni rappresenta le indicazioni di navigazione per l'utente che andranno ricalcolate ad ogni passaggio; la velocità_media racchiude la velocità presunta dell'utente; modalità_emergenza identifica se i percorsi vengono calcolati per la navigazione normale o per quella in stato di emergenza; tutti gli altri sono attributi di tipo Nodo che serviranno per descrivere la posizione iniziale, la destinazione e il punto di transito. Per quanto riguarda i metodi, si intuisce immediatamente dai nomi che servono tutti a calcolare o settare particolari valori degli attributi della classe. Menzione particolare va riservata a nuova_posizione_utente() che di fatto andrà ad aggiornare il valore dell'attributo punto_di_controllo, che nel caso in esame modellerà la posizione esatta dell'utente in uno dei nodi. Infine le classi che costituiscono i tipi di dato utilizzati dal software ovvero: **Arco**, **Nodo**, **Mappa**. **Nodo** rappresenta un nodo qualsiasi sul grafo, dunque possiede gli attributi che lo identificano in maniera univoca nel sistema di riferimento tridimensionale, più il punto di raccolta più vicino, e un metodo per estrarre i valori di tali attributi. La classe **Mappa** raccoglie tutte le costanti nel calcolo del costo totale K di un arco più un indicatore per il piano. Questa scelta è stata fatta per limitare gli accessi in lettura e scrittura a tali costanti che rimarranno invariate per qualsiasi arco. La classe **Arco** raccoglie tutti gli attributi che definiscono un arco del grafo, quindi tutte le variabili nel calcolo del K (che saranno periodicamente aggiornate dal server), il valore

del costo K una volta calcolato, i nodi di riferimento che identificano l'arco e un identificatore per capire se un particolare arco rappresenti una scala.

4.3.2 Diagramma dei Packages

Una volta stilato il diagramma delle classi di analisi il nostro lavoro si è concentrato sullo sviluppo del modello dei packages di analisi. Ad ogni storia utente è stato fatto coincidere un package, riconoscendo all'interno di ciascun flusso le diverse classi e suddividendole a seconda del proprio stereotipo.



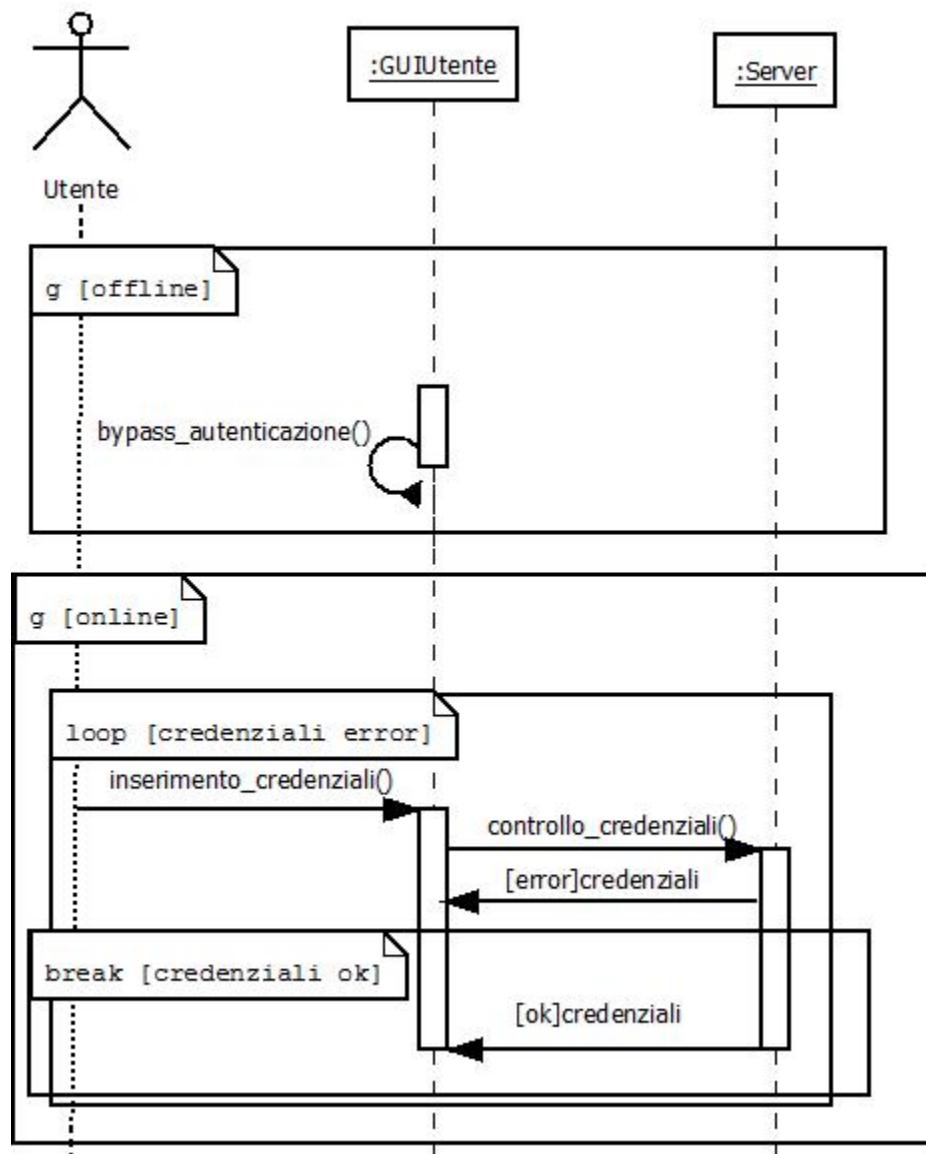
Successivamente sono state individuate le dipendenze tra packages che sono state rappresentate attraverso le opportune frecce. Allora si vede come quelli di navigazione, sia in condizioni normali che di emergenza offline, siano dipendenti dall'invio e aggiornamento della posizione dell'utente, che a sua volta dipende dal package di autenticazione. Al contrario la navigazione in emergenza online risulta dipendente sia dall'invio della posizione, che dall'autenticazione, che dalla notifica di emergenza. Ricordiamo infatti che una delle possibilità per far partire la navigazione di emergenza è quella della segnalazione da parte del server. Infine la notifica di emergenza è risultata dipendente solamente dalla autenticazione alla prima apertura.

4.4 Analisi orientata ai comportamenti

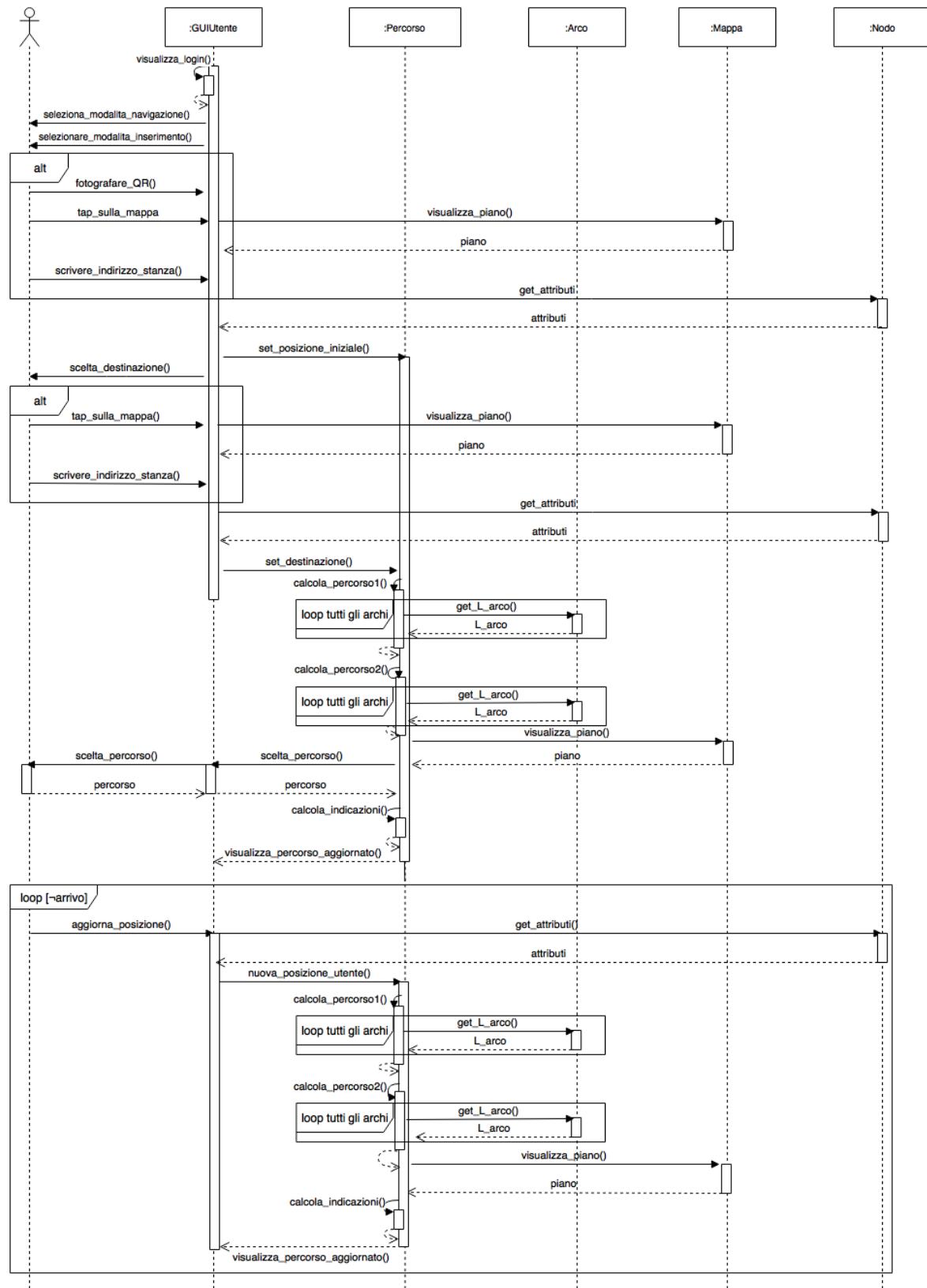
4.4.1 Diagramma delle sequenze

Si procede ora riportando i diagrammi delle sequenze di 4 user stories, questo in virtù del fatto che quello relativo alla notifica di emergenza è considerato troppo ridotto per una modellazione di questo tipo, e quello sull'aggiornamento e invio della posizione sul server, verrà comunque integrato in tutti quelli considerati.

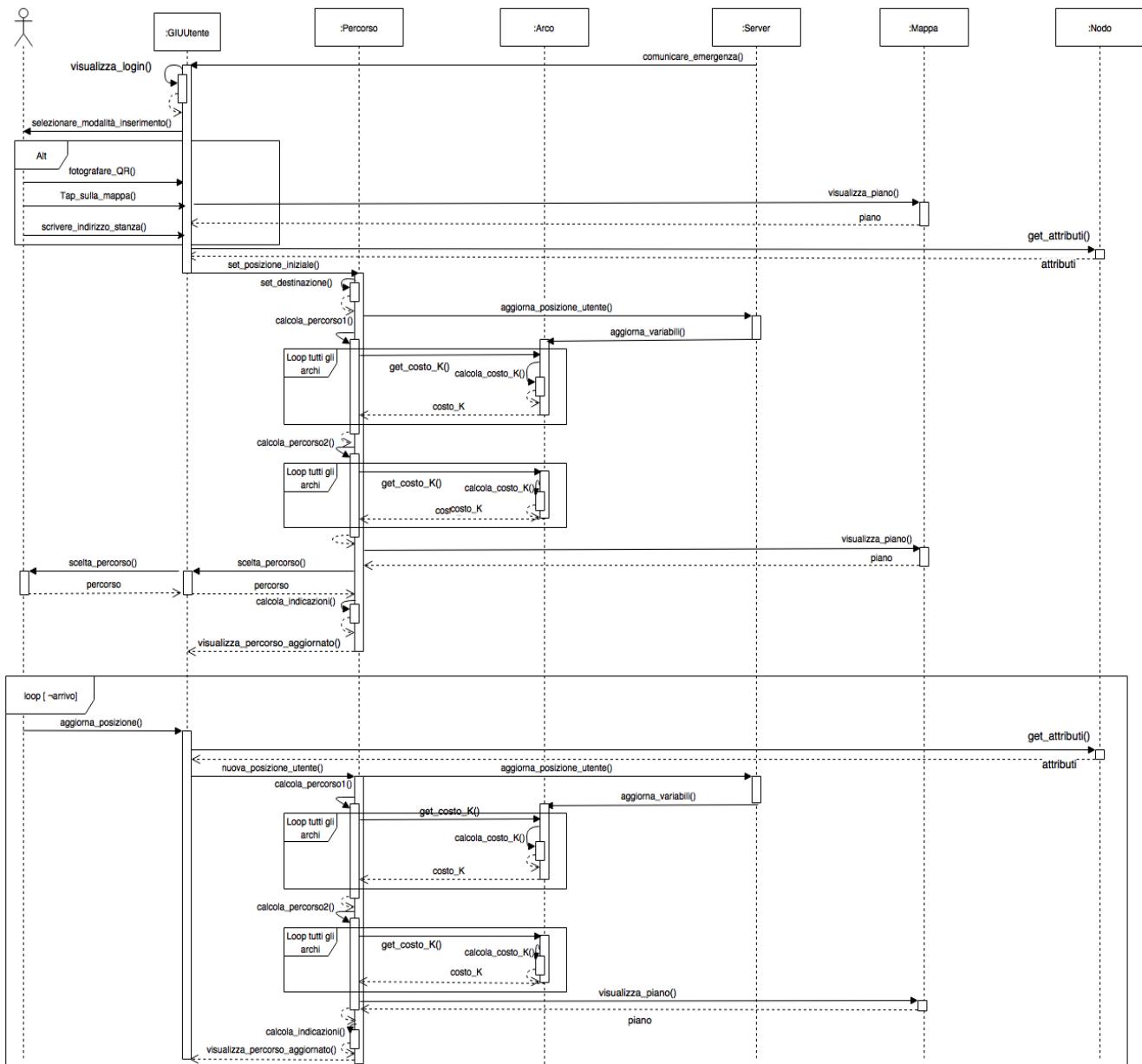
User Story 1 : Inserimento credenziali



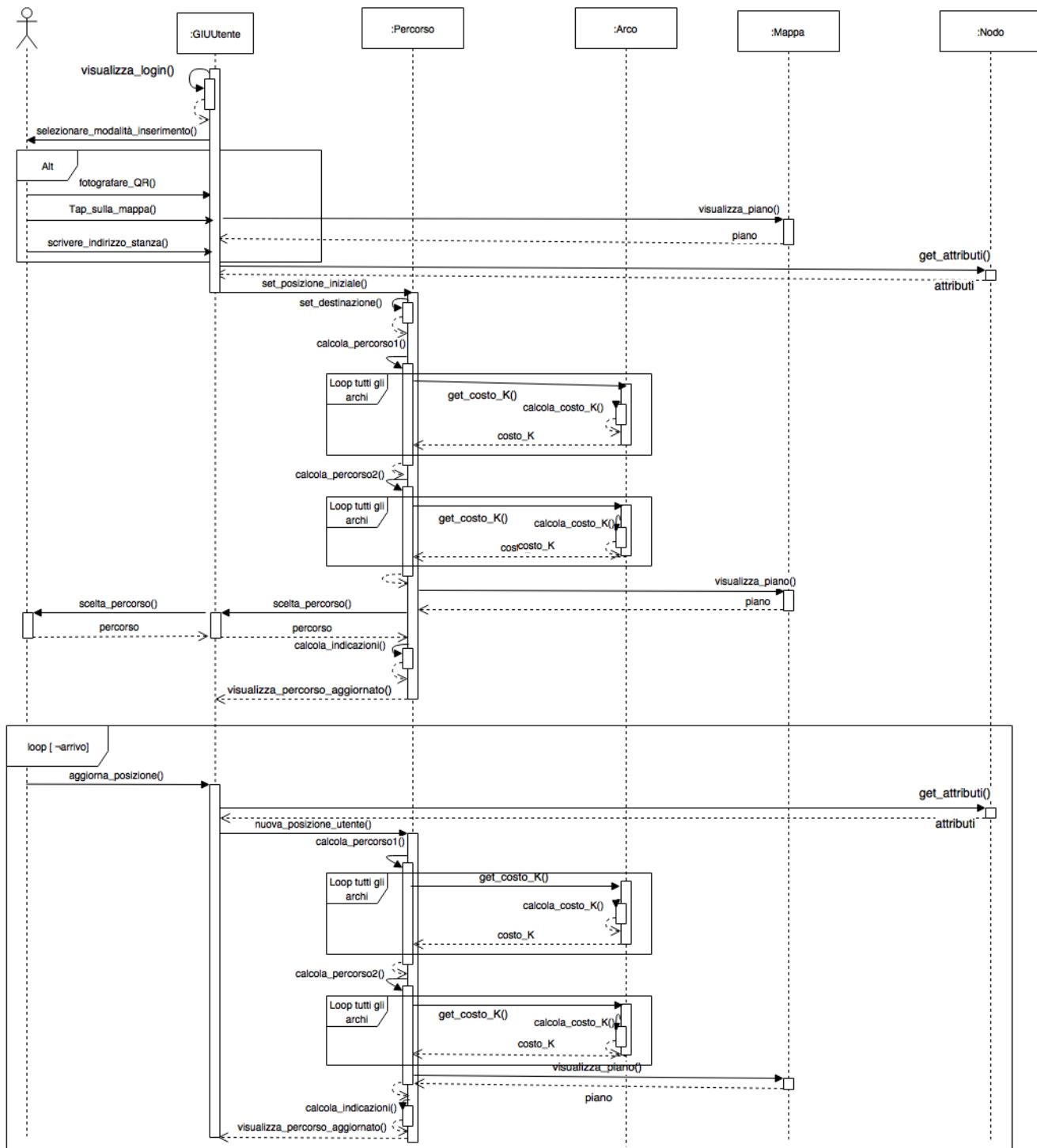
User Story 3 : Funzionamento in condizioni normali



User Story 5: Guida al punto di raccolta in modalità online



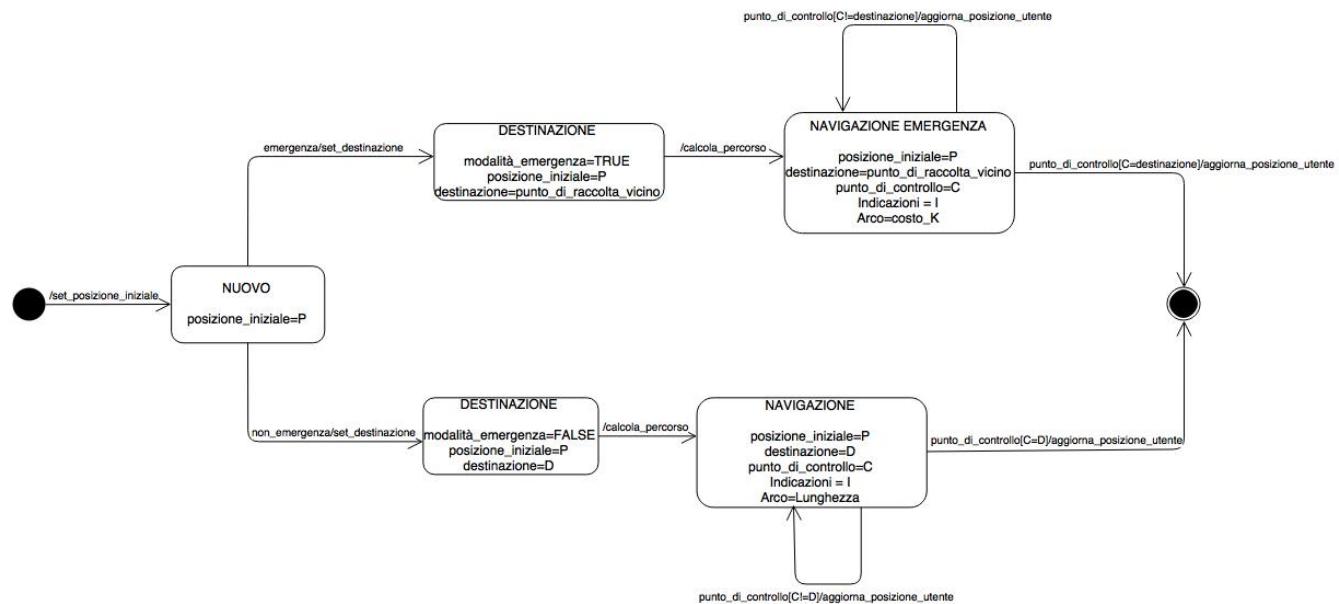
User Story 6: Guida al punto di raccolta in modalità offline



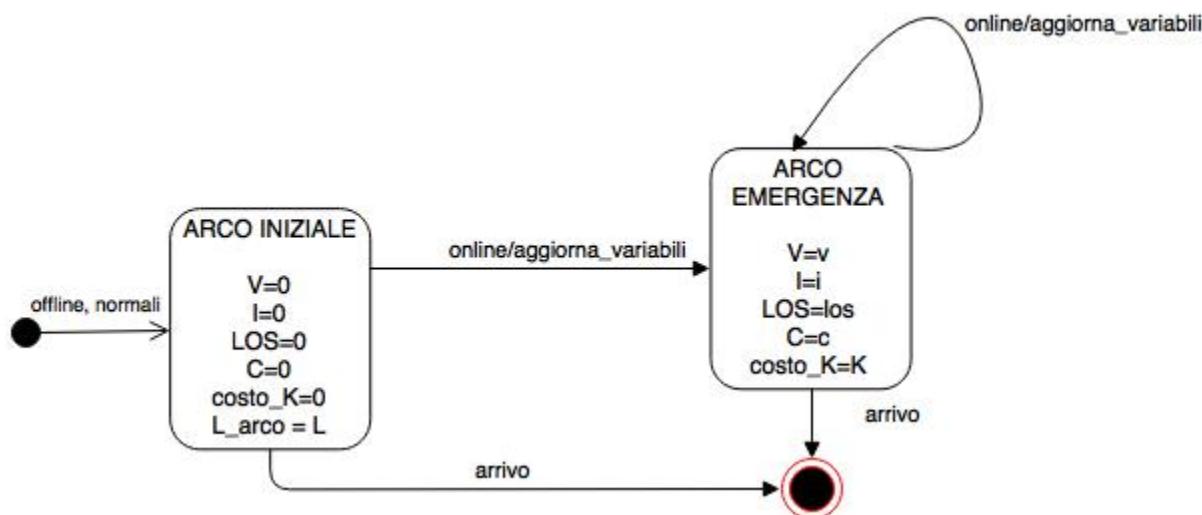
4.4.2 Diagramma degli stati

Si riportano, a continuazione dell'analisi sui comportamenti, i diagrammi di stato che descrivono i cambiamenti di un oggetto durante tutto il suo ciclo di vita.

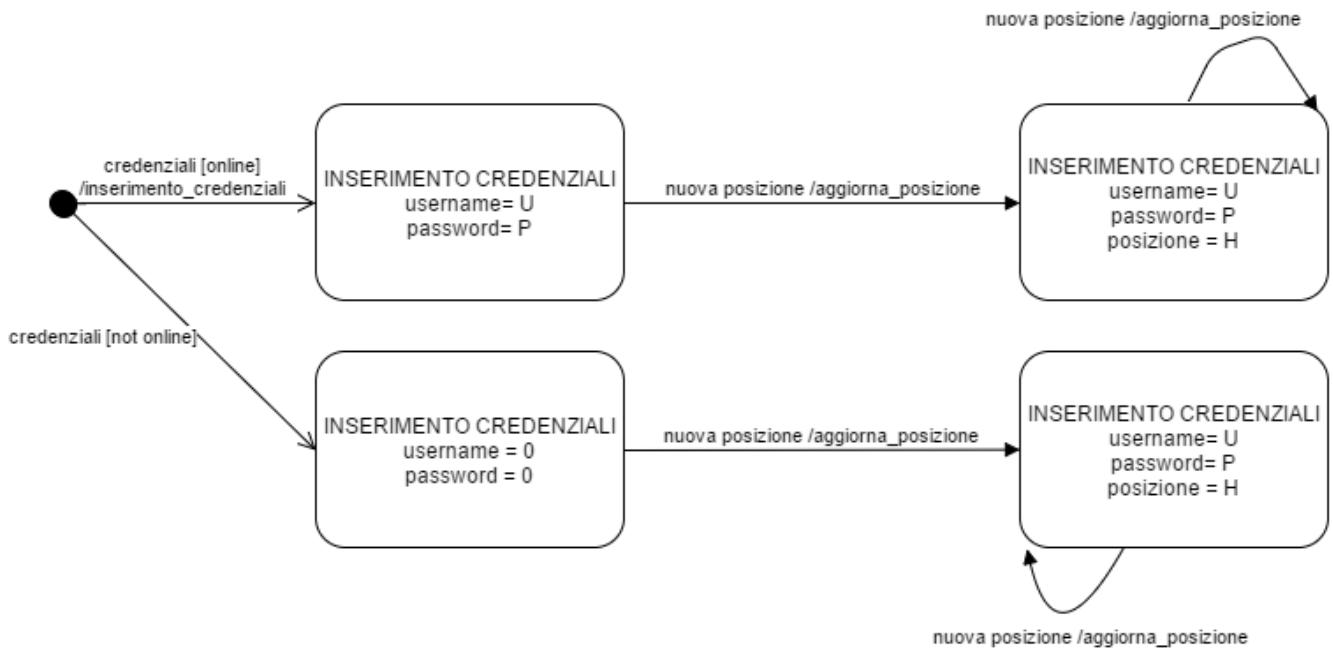
Classe "Percorso"



Classe "Arco"



Classe “Utente”

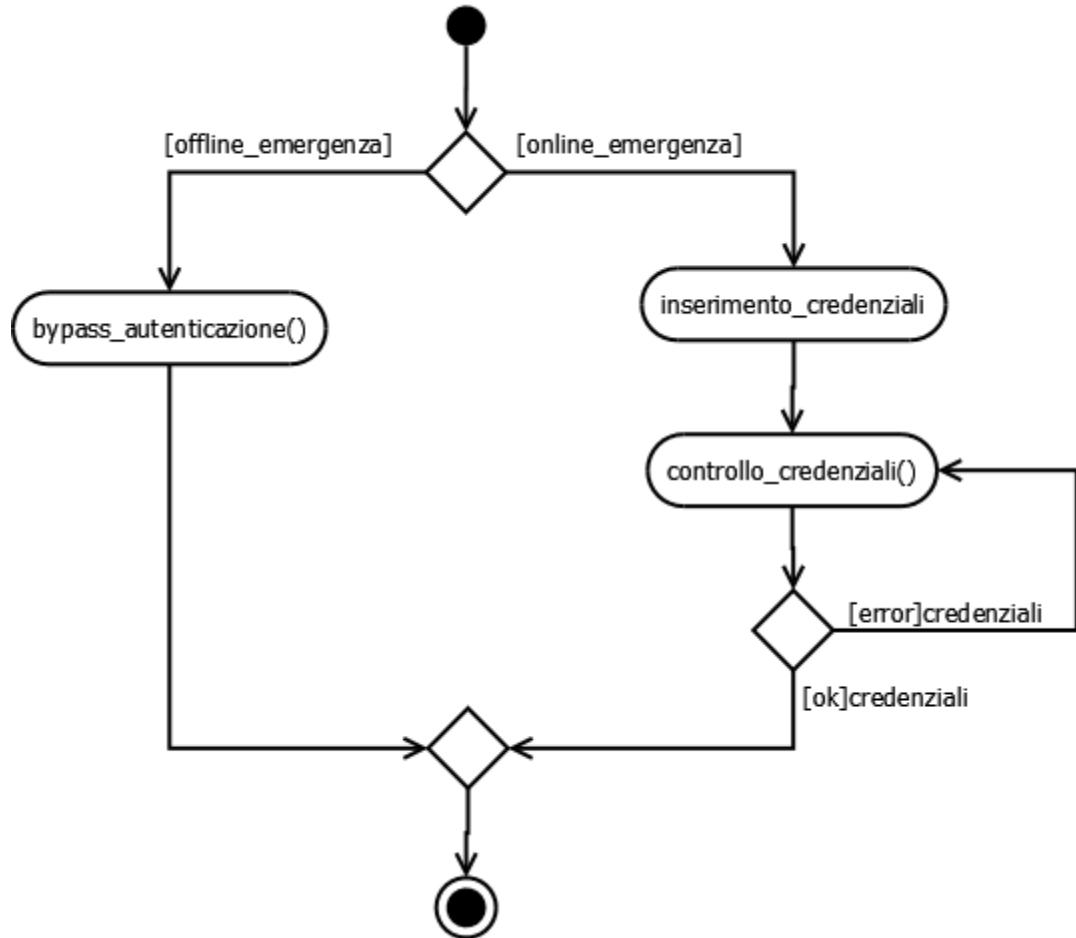


4.4.3 Diagramma delle attività

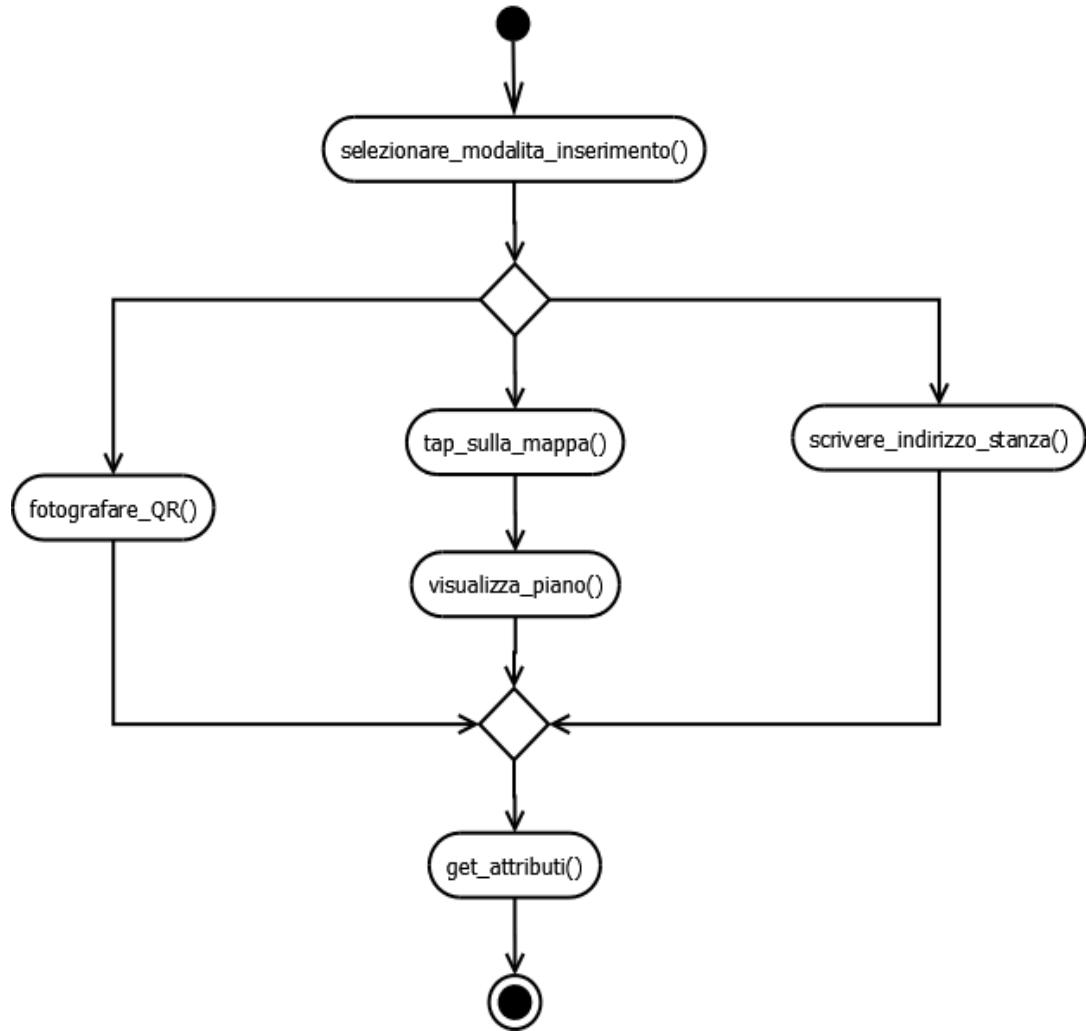
Come conclusione della fase di analisi orientata ai comportamenti, vengono mostrati nel seguito i diagrammi delle attività utilizzati per descrivere il flusso di elaborazione dei principali metodi interni alle classi analizzate. Sulla base dei diagrammi delle sequenze visti in precedenza, per ogni focus di controllo che riguarda un metodo di una classe, si compone un grafo in cui le attività sono i nodi e le transizioni sono gli archi. In particolare, abbiamo concentrato la nostra attenzione sui metodi riguardanti le classi “GUiutente” e “percorso” in interazione tra loro e con le restanti classi, accorpando in un unico diagramma quelle operazioni che vengono svolte negli scenari di emergenza online, offline e in condizioni normali.

- Classe GULutente

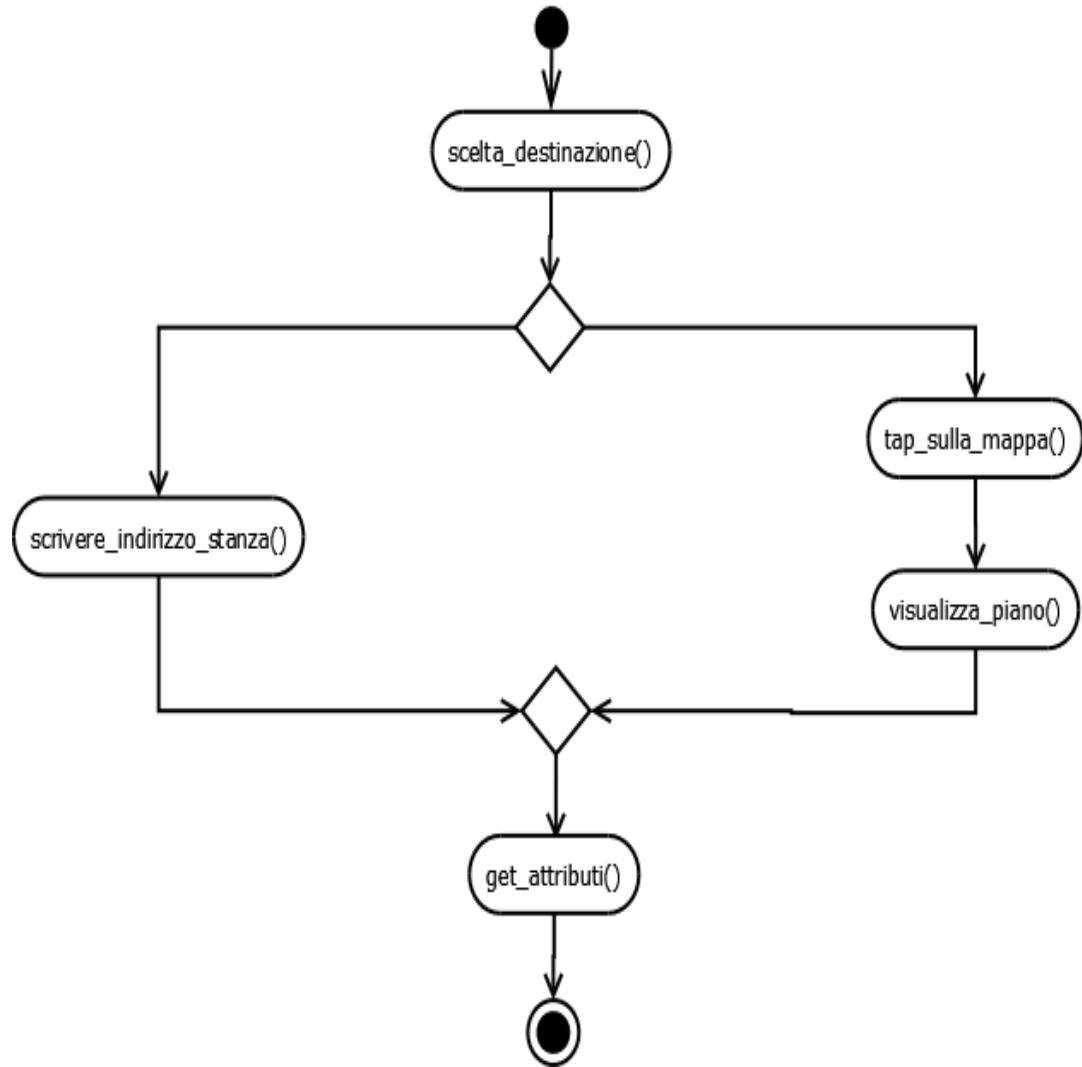
-Metodo: inserimento_credenziali()



-Metodo: Selezione_modalità_inserimento()

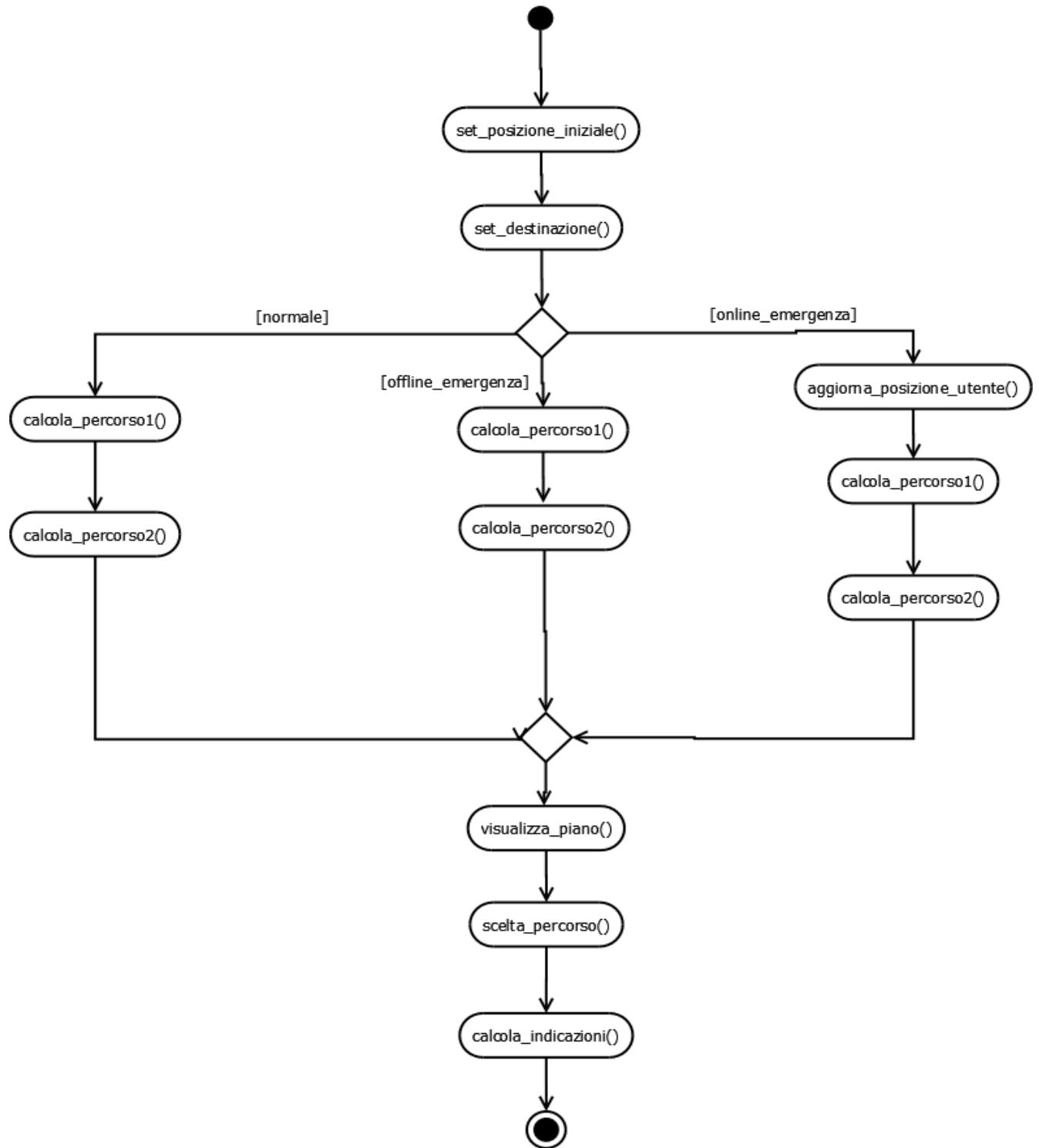


-Metodo: scelta_destinazione()

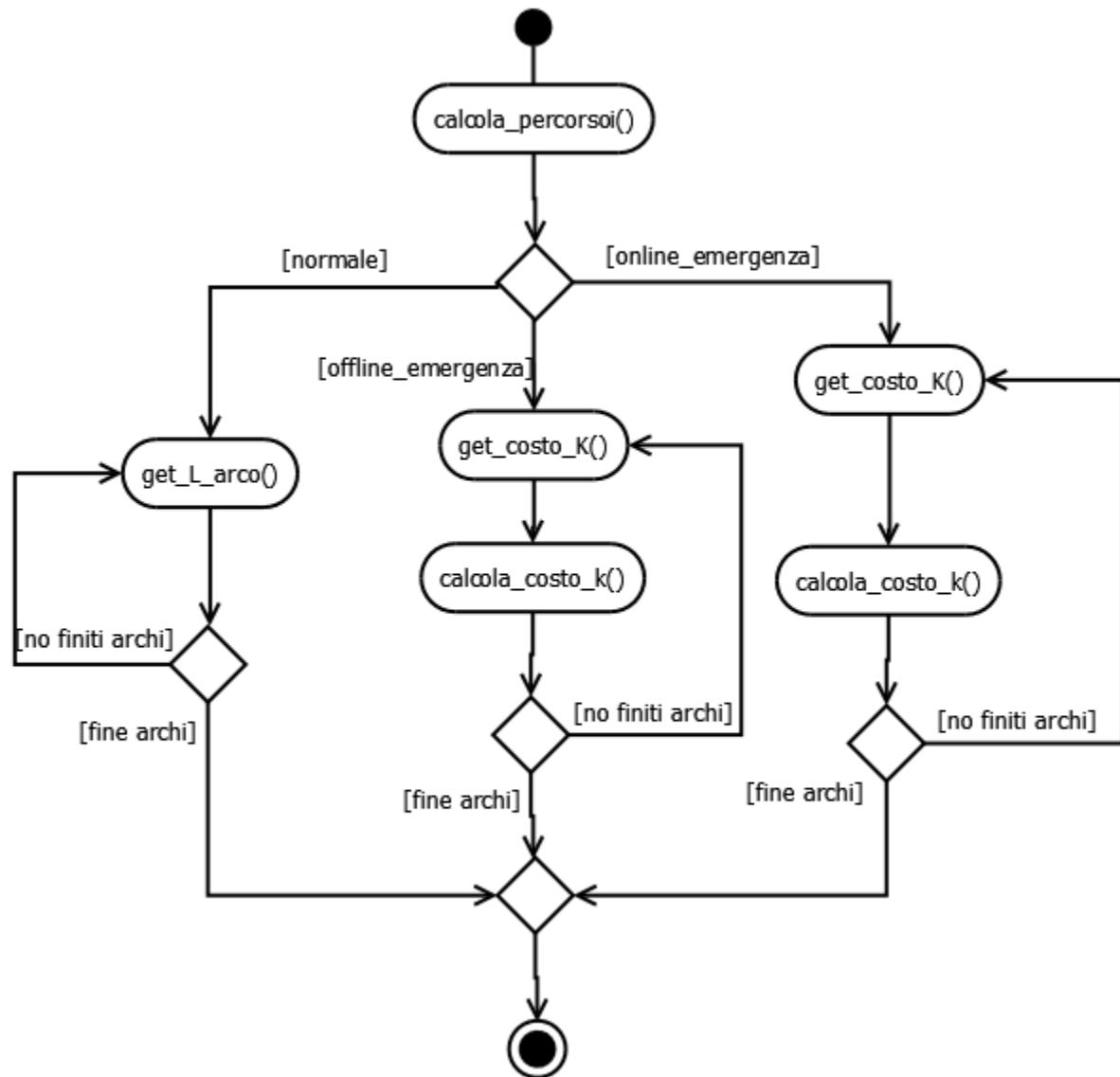


- Classe Percorso

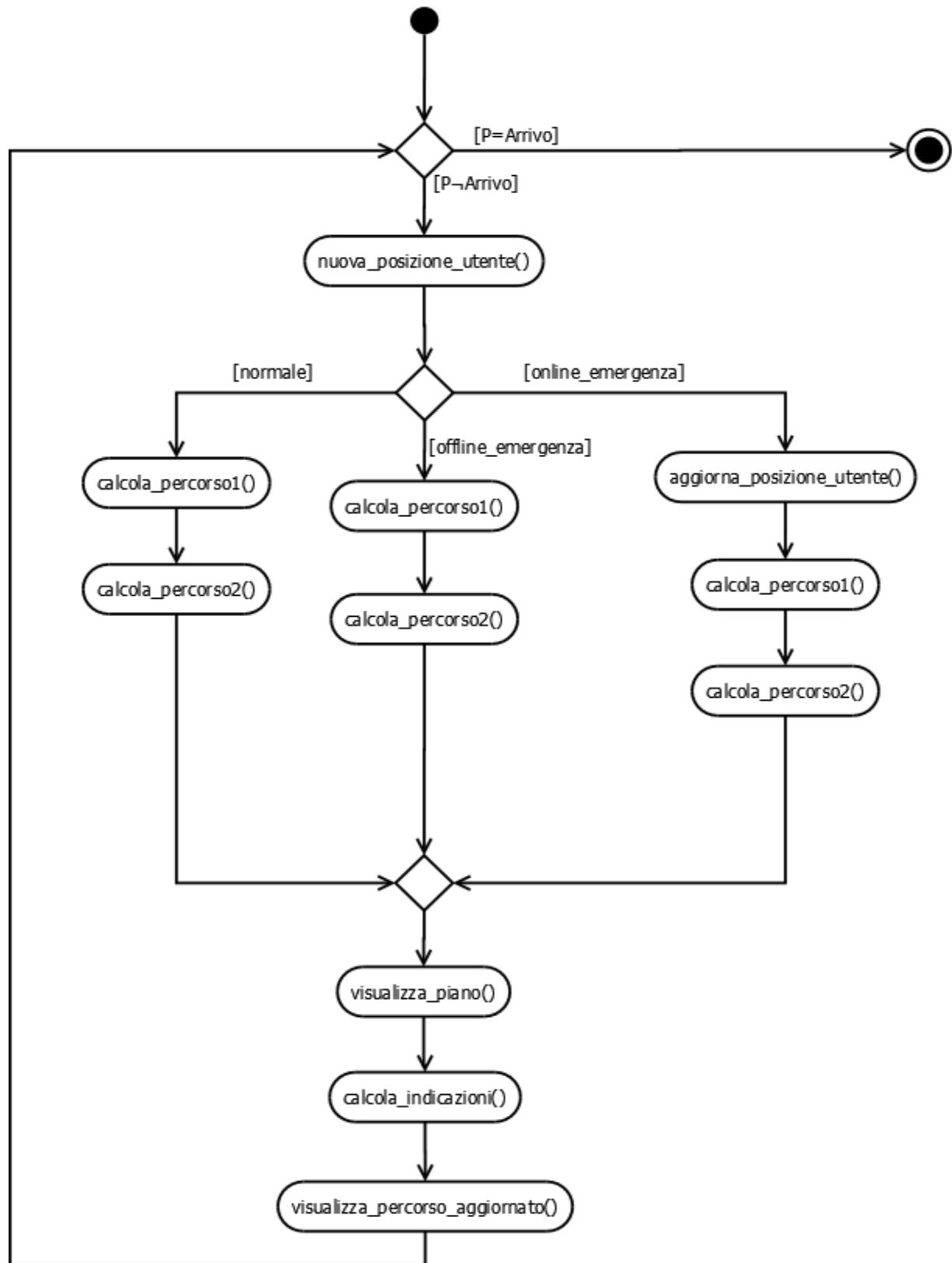
-Metodo: set_posizione_iniziale()



-Metodo: calcolo_percorso1() e/o calcolo_percorso2()



-Metodo: nuova_posizione_utente()

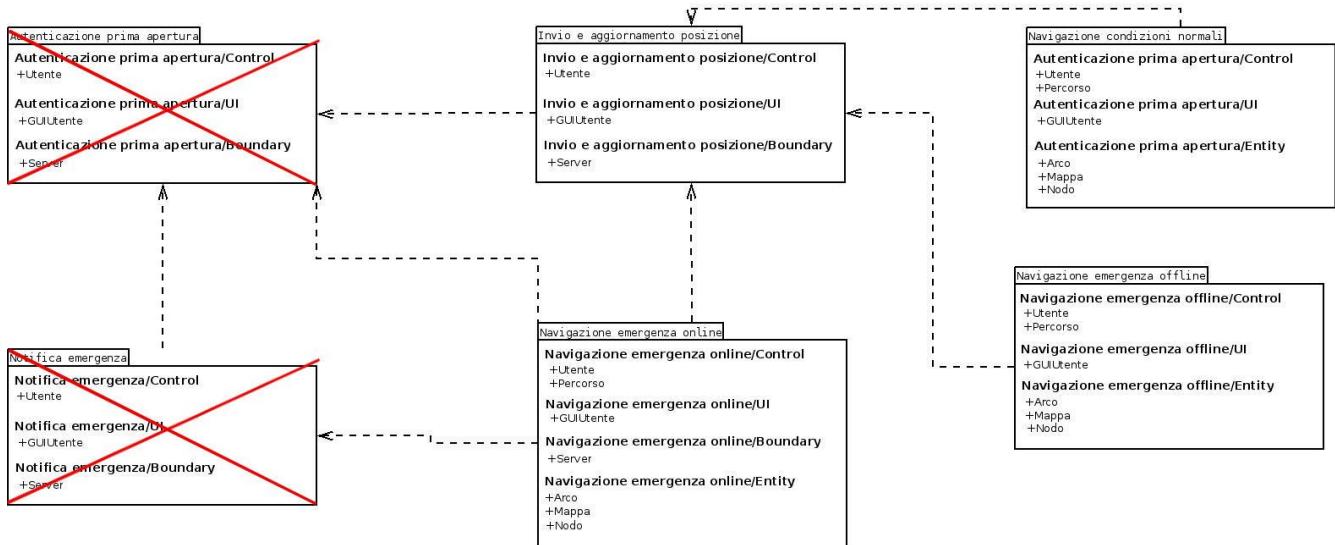


5. PROGETTAZIONE

Terminata la fase di analisi si passa a quella di progettazione, i prodotti di questa fase saranno i diagrammi dell'architettura e delle classi di progettazione che, per loro natura, avvicineranno il modello teorico all'implementazione del software permettendo di analizzare con maggiore cognizione di causa il lavoro svolto finora.

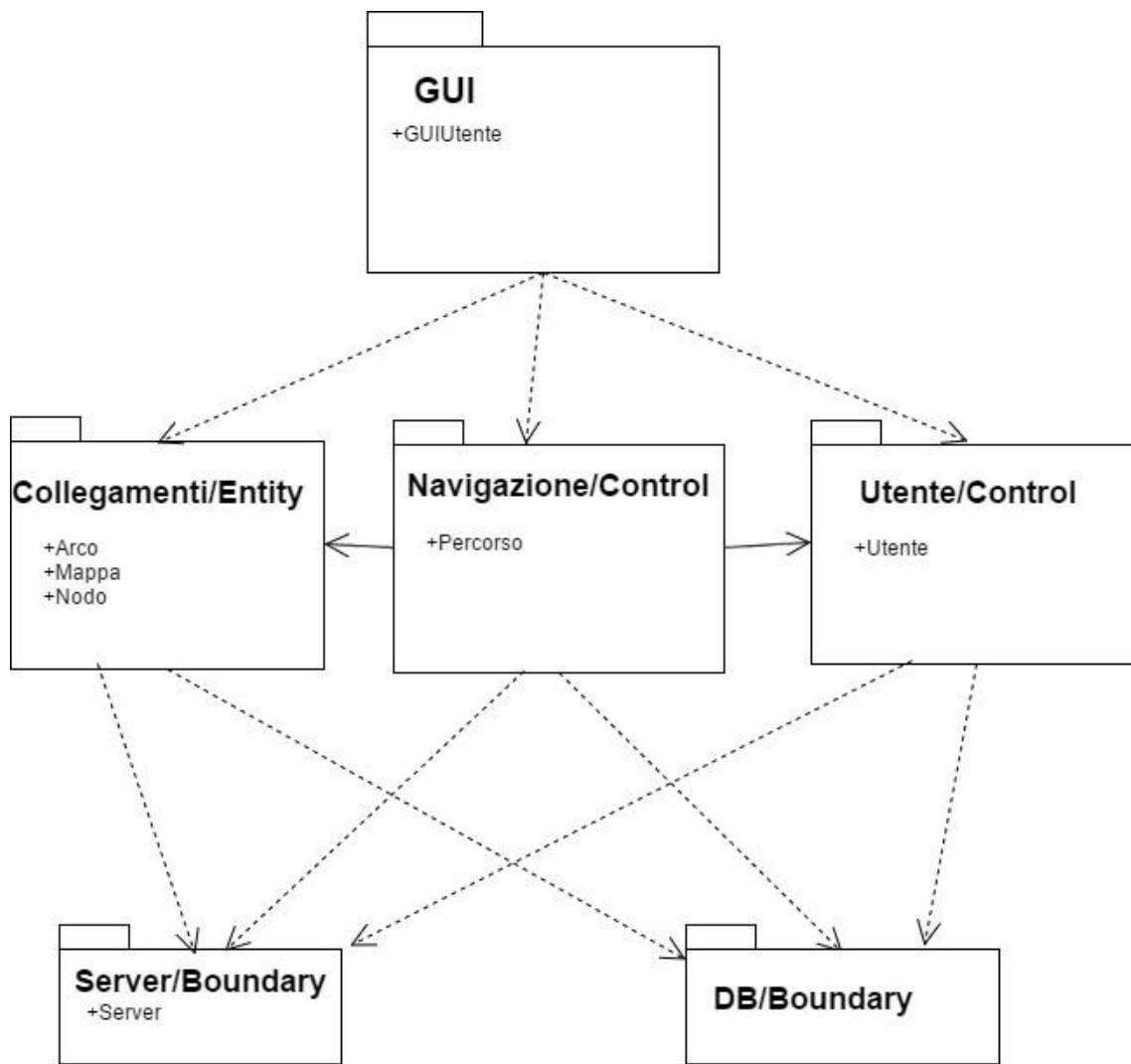
5.1 Progettazione dell'Architettura

Per prima cosa si definisce il diagramma dei componenti, che per la programmazione ad oggetti spesso aderisce al modello astratto dell'architettura stratificata. Per la produzione del diagramma è necessario riprendere il modello dei packages di analisi, contenente tutti i packages provenienti da tutte le user stories, ed effettuare un raffinamento. Tale raffinamento è stato effettuato, da un lato, cercando di massimizzare la coesione interna e minimizzare l'accoppiamento, e dall'altro cercando di fare in modo che il diagramma risultante non rendesse di difficile gestione il prodotto finale.



Partendo dal diagramma dei packages ottenuto ai passi precedenti si è deciso di eliminarne due: Autenticazione prima apertura, Notifica di emergenza.

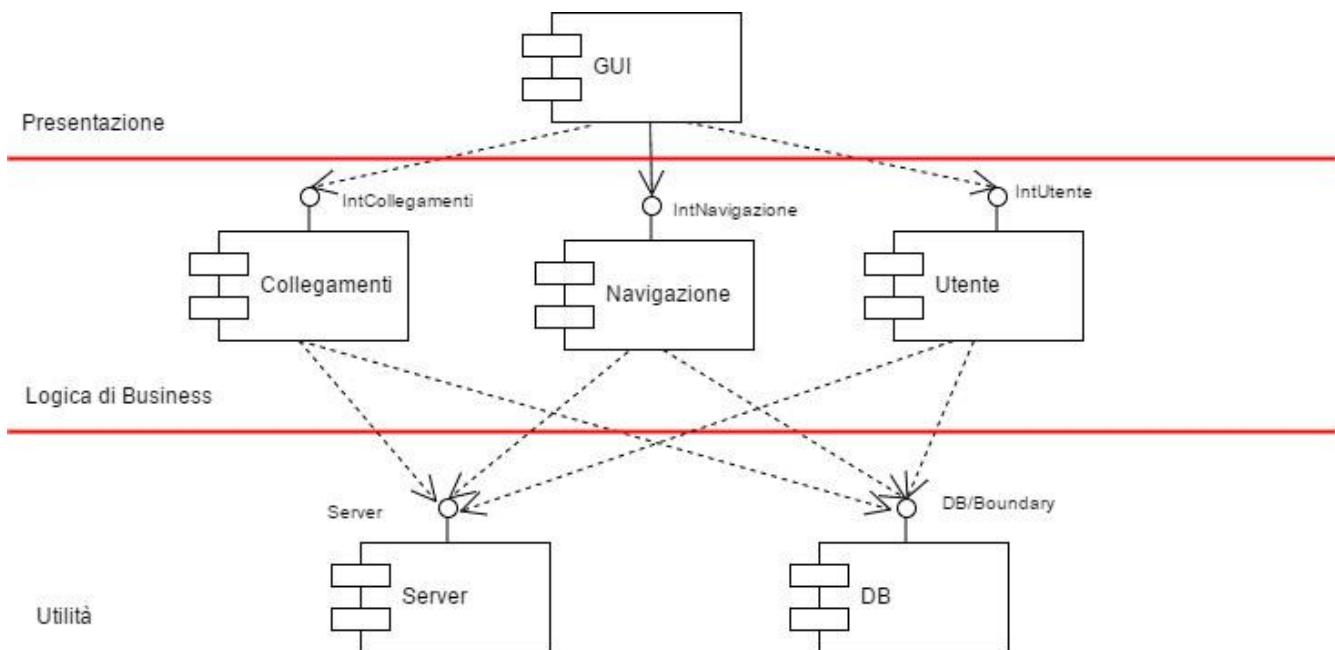
Questo perché le classi contenute in quei packages erano già tutte comprese in altri più complessi, dunque per minimizzare l'accoppiamento si è ritenuto di poterli fondere. Successivamente a questo primo intervento, si è proceduto a verificare che non vi fossero dipendenze circolari ed a un ricollocamento delle classi all'interno di package che si avvicinassero maggiormente alla struttura stratificata che si voleva raggiungere.



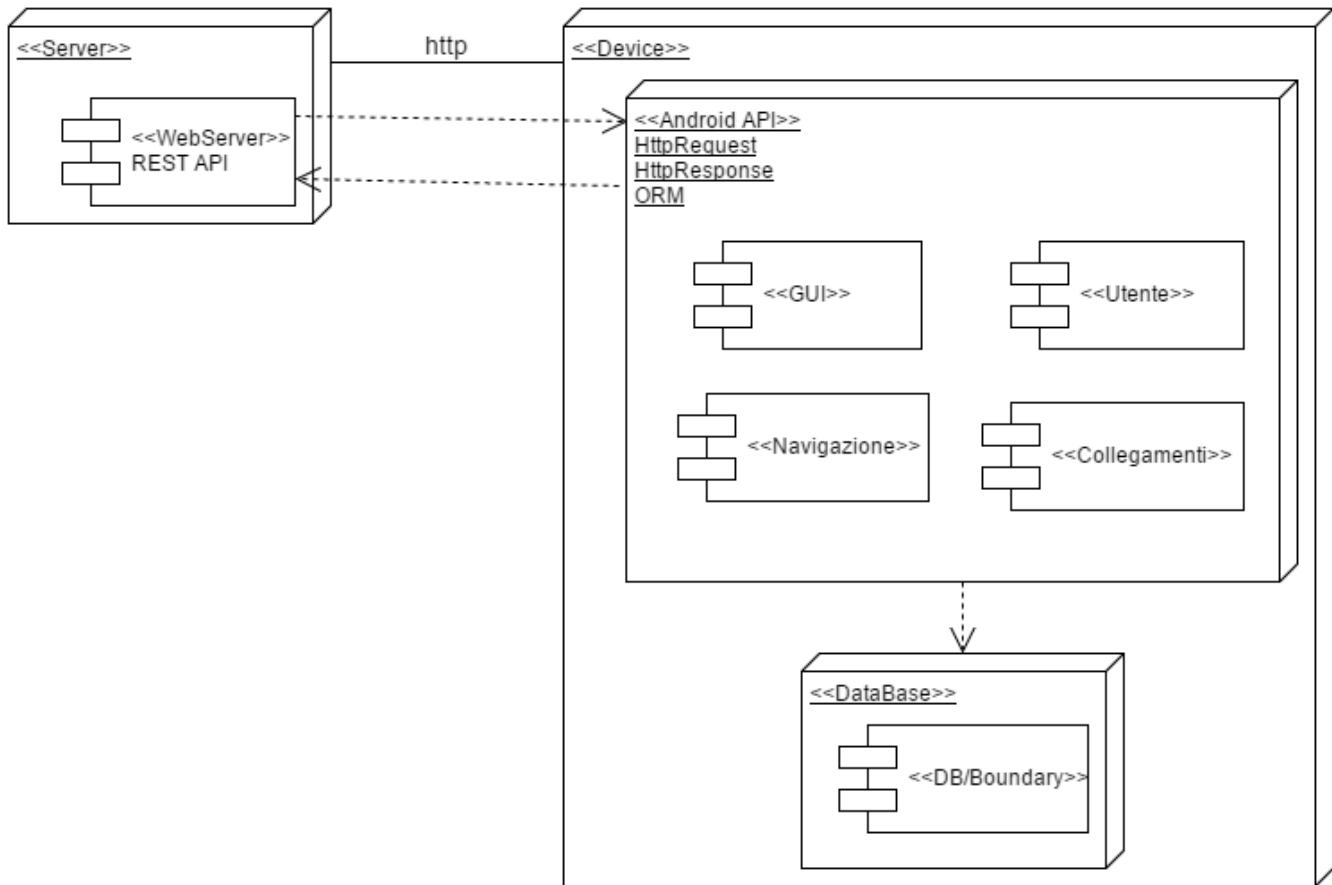
Come si evince dal grafico i packages sono stati riorganizzati tenendo conto delle modalità del loro utilizzo, cercando di fare in modo che modellassero singoli componenti il cui uso rispecchiasse la descrizione delle tipologie. In questo senso sono

stati accorpati i concetti di Arco, Nodo e Mappa in un unico componente di tipo entity chiamato Collegamenti; mentre i concetti di Navigazione ed Utente sono stati scissi in due packages di tipo Control. Si nota infine il package destinato a modellare le interazioni attraverso la GUI e i quelli di tipo Boundary che modellano risorse esterne con le quali ci si dovrà interfacciare, come appunto Server e DataBase. Sulla base di questa nuova organizzazione, si sono aggiornate anche le relazioni che legano i nuovi componenti del grafo, tenendo conto di tutte le possibili operazioni che il software dovrà essere in grado di eseguire.

Continuando verso la creazione del diagramma dei componenti, sulla base del diagramma dei packages raffinato, si sono individuate le interfacce tra i diversi componenti e li si è disposti sui tre diversi livelli prescritti dal paradigma architetturale stratificato. Il risultato di quest'ultima fase di raffinamento è il seguente diagramma finale dei componenti.



Infine, ad eccezione di Server, tutti i componenti risiederanno sullo stesso device fisico. Questo ospiterà anche il DB sul quale verranno salvati tutti i dati persistenti che il software dovrà essere in grado di consultare anche in modalità offline. Perciò il diagramma di dislocamento (o di deployment) diviene estremamente semplice e risulta essere il seguente.



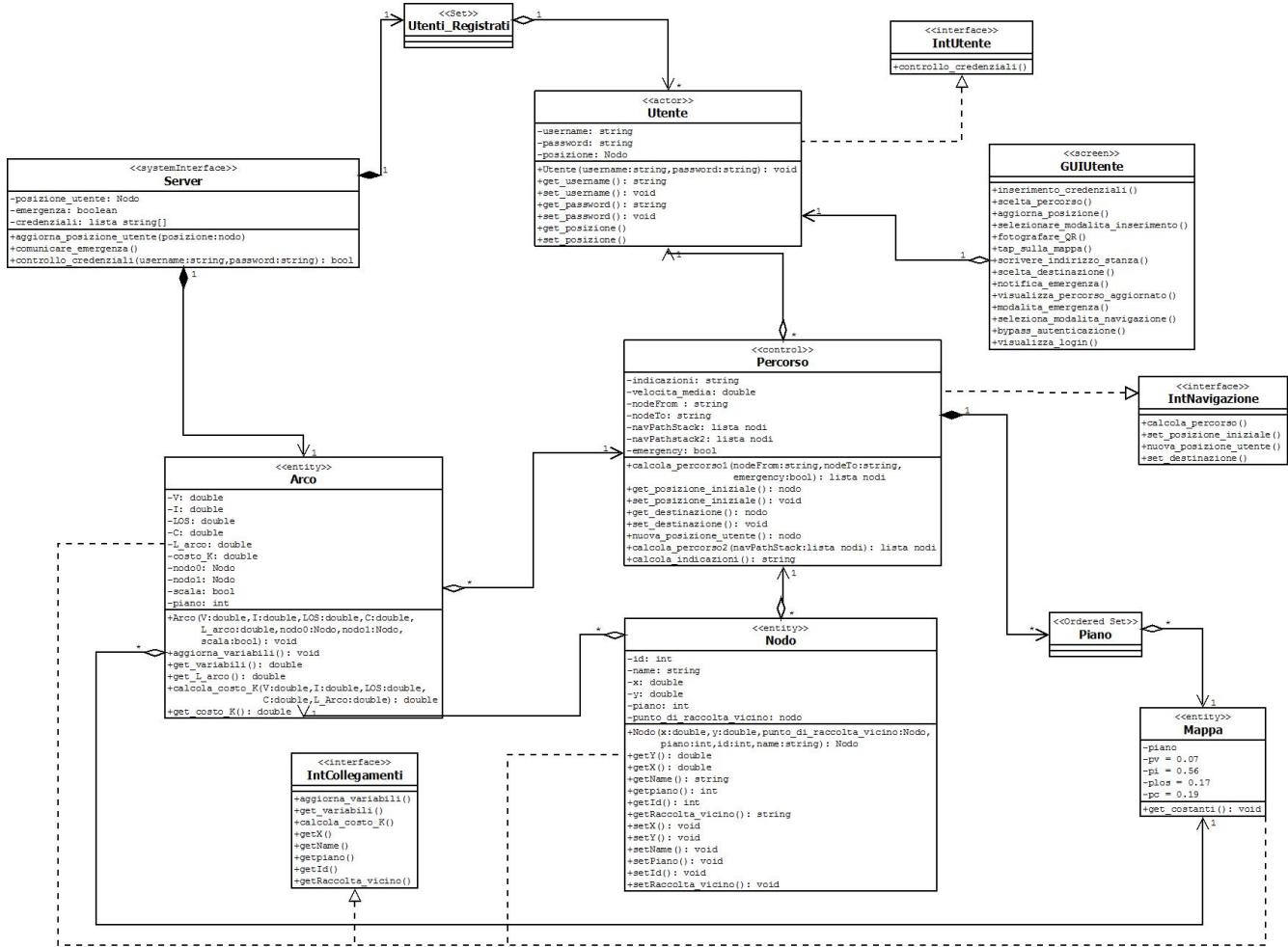
5.2 Progettazione dei Componenti

5.2.1 Modello delle classi di progettazione

Finita la fase di progettazione dell’architettura ed individuati i componenti, si passa alla progettazione interna dei componenti stessi. Per effettuare questo passaggio si dovranno trasformare tutte le classi e le relazioni di analisi in classi e relazioni di progettazione.

Seguendo le linee guida per ogni classe di analisi si è creata una classe di progettazione, inserendo una interfaccia nel caso in cui la classe dipendesse da un’altra di un componente esterno. Inoltre per ciascuna classe si è provveduto a rivedere l’insieme degli attributi e dei metodi, in alcuni casi aggiungendo elementi e in altri eliminando

possibili duplicati. In seconda battuta si sono raffinate le relazioni, facendo attenzione a stabilire un verso di navigazione e raffinandole come prescritto nelle linee guida.



Come si può vedere dal nuovo diagramma, sono stati introdotti molti nuovi metodi pubblici che consentiranno di accedere agli attributi delle singole classi. Inoltre si nota come oltre alle sopracitate interfacce, siano state aggiunte anche classi intermedie per modellare le relazioni di tipo uno a molti.

5.2.2 Progettazione del Database

Dopo aver individuato le classi di progettazione con i relativi stereotipi si è proceduto ad individuare tutte le classi di tipo **<<entity>>** ed **<<actor>>** e le relative associazioni per comporre lo schema logico della base di dati.

Utente(username, password, posizione)

Nodo(id, x, y, piano, punto_di_raccolta vicino)

Arco(V, I, LOS, C, L_Arco, costo_K, nodo_0, nodo_1, scala, **piano**)

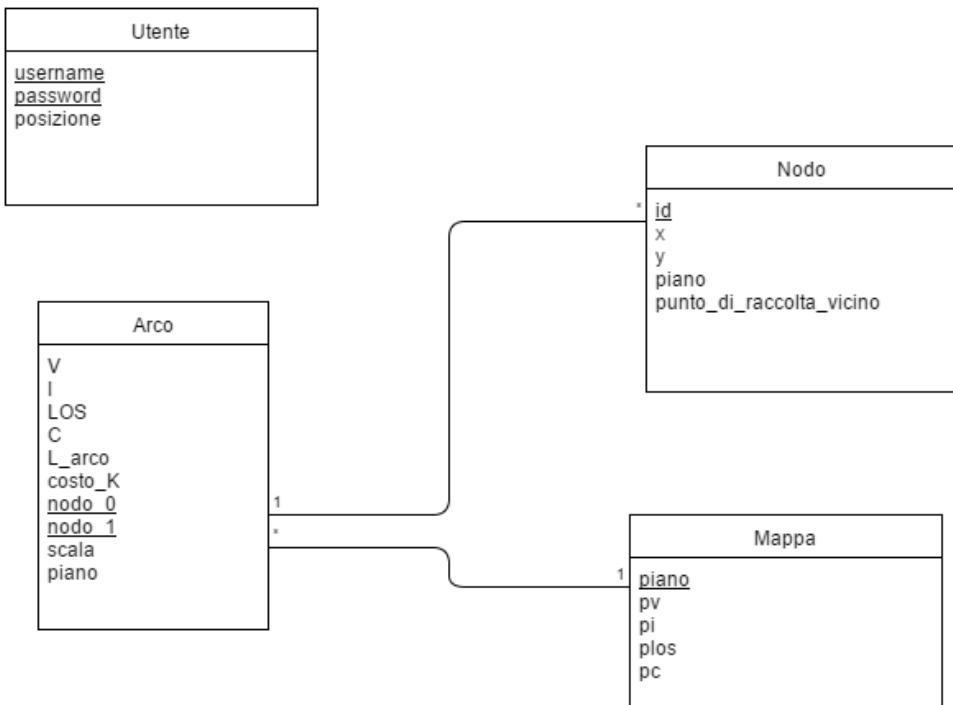
Mappa(**piano**, pv, pi, plos, pc)

.. = vincolo di integrità referenziale

.. = vincolo di integrità referenziale

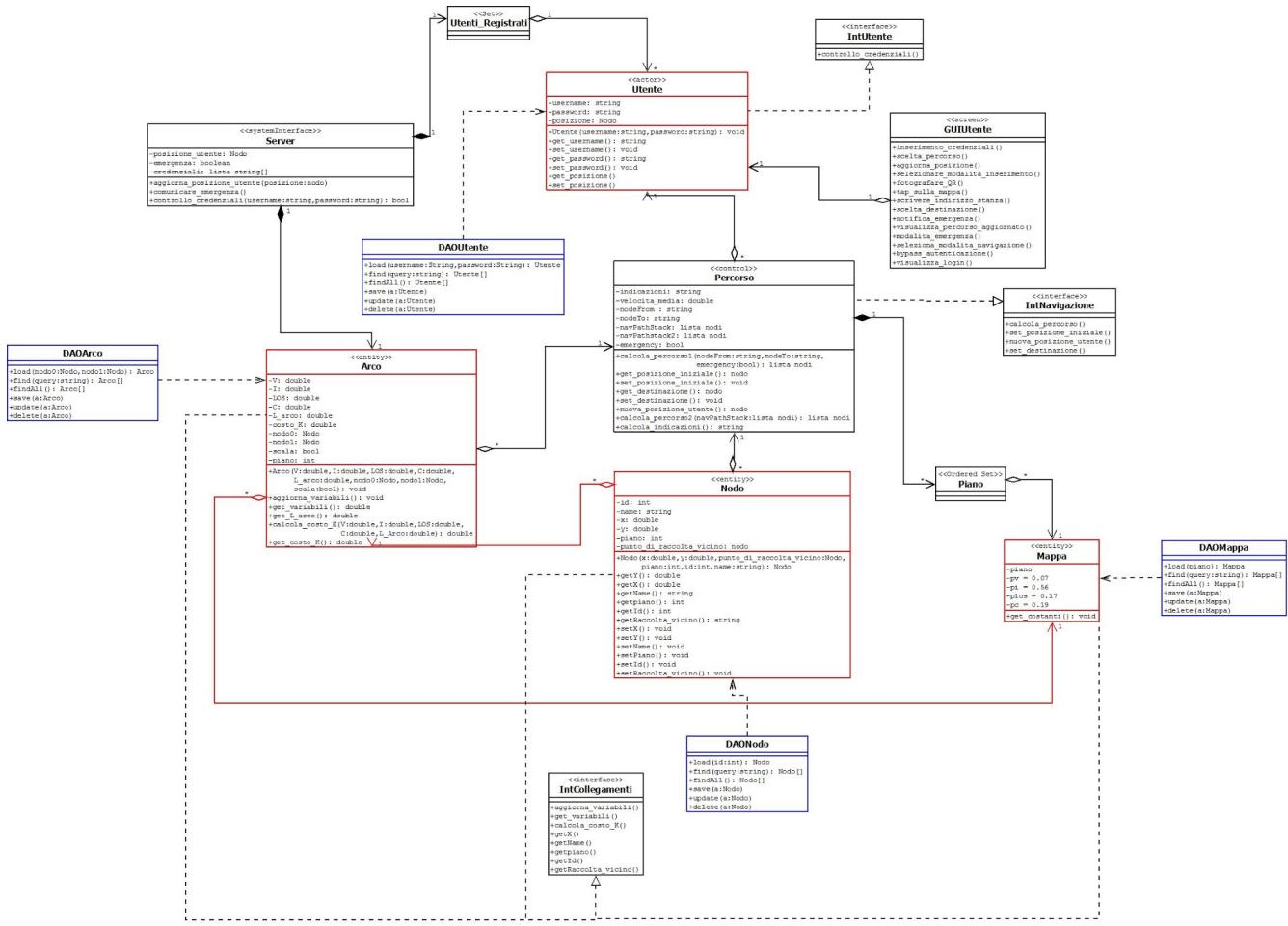
__ = chiave primaria

Attraverso i criteri sopra esposti sono state selezionate le classi Utente, Nodo, Arco e Mappa. Le relazioni riportate nel modello delle classi di progettazione sono state tradotte in associazioni del modello relazionale, producendo dei vincoli di integrità referenziale sia tra i nodi che individuano un arco e la classe Nodo, che tra l'attributo piano di Arco e la classe Mappa. Inoltre per ciascuna entità relazionale sono stati individuati gli attributi che ne costituiscono la chiave primaria, resi riconoscibili attraverso la sottolineatura. Utilizzando una rappresentazione più schematica e visuale, lo schema finale del DB è risultato il seguente.



Come si vede la relazione relativa alla classe Utente rimane disgiunta dalle altre classi poiché non possiede nessun vincolo di integrità referenziale. Tutto il resto invece rispecchia fedelmente lo schema logico. Infine sono state aggiunte le cardinalità relative a ciascun lato di ogni associazione presente.

Individuati i componenti del database, per modellare la persistenza utilizzando gli strumenti messi a disposizione dal paradigma, sono state inserite le classi DAO che modellano le interazioni CRUD con la base di dati.



Nello schema in figura sono state colorate in rosso tutte le classi e le relazioni facenti parte del database, inoltre a queste sono state aggiunte le classi DAO in blu che rappresentano un'interfaccia per svolgere le operazioni sulla base di dati.

5.3 Metriche

Per aiutarsi durante le varie fasi della progettazione ci si è avvalsi dell'ausilio di metriche. In particolare per verificare alcune proprietà dei package di analisi per

produrre il successivo diagramma dei componenti e per verificare la bontà delle classi di progettazione prodotte. Rispetto ai package si sono usate le metriche per verificare: la coesione interna (metrica CCRC), la stabilità (metrica S) e l'astrattezza (metrica A). Per quanto riguarda le classi di progettazione, sono state utilizzate le metriche relative alla coesione interna (metrica LCOM4), alla interdipendenza (metrica CBO) e infine alla complessità (metrica WMC).

5.3.1 Coesione Interna Package - CCRC

Attraverso la metrica CCRC (Class Category Relational Cohesion) si cerca di stabilire quanto un package sia coeso, poiché da una maggiore coesione si ottiene maggiore modularità e manutenibilità.

Package	Valore CCRC
ID-1 Autorizzazione prima apertura	$2/3 = 0,666$
ID-2 Invio e aggiornamento posizione sul server	$2/3 = 0,666$
ID-3 Navigazione in condizioni normali	$6/7 = 0,857$
ID-4 Notifica emergenza	$2/3 = 0,666$
ID-5 Navigazione emergenza online	$9/7 = 1,286$
ID-6 Navigazione emergenza offline	$7/6 = 1,166$

Come si vede dai valori riportati in tabella i packages sono risultati debolmente coesi, soprattutto per quanto riguarda le operazioni più elementari. Questo è stato uno degli indicatori che ha portato il gruppo a riflettere su alcuni possibili accorpamenti, cercando di non ammassare assieme concetti diversi.

5.3.2 Stabilità Package - S

Un package risulta essere stabile quando risulta avere più dipendenze entranti che uscenti, questo significa che a una sua modifica ne dovranno seguire molte altre in tutti

i packages che sono dipendenti da questo. Di seguito la tabella con i valori per la metrica.

Package	Valore S
ID-1 Autorizzazione prima apertura	1
ID-2 Invio e aggiornamento posizione sul server	$3/4 = 0,75$
ID-3 Navigazione in condizioni normali	0
ID-4 Notifica emergenza	$1/2 = 0,5$
ID-5 Navigazione emergenza online	0
ID-6 Navigazione emergenza offline	0

Dai valori si evince come molti packages non risultino stabili, alcuni presentano addirittura valori nulli, questo a causa del fatto che presentano poche o nessuna dipendenze entranti.

5.3.3 Astrattezza Package - A

Da quanto appreso dalla teoria una classe astratta è una classe genitore i cui metodi hanno una descrizione solo nelle classi figlio; nel caso in esame non si hanno generalizzazioni e dunque nessuna classe genitore. Per questo motivo il valore di A sarà uguale a 0 per ogni package.

5.3.4 Coesione Interna - LCOM4

Di seguito le tabelle con i valori della metrica LCOM4 (Lack of Cohesion in Methods 4) per le classi di progettazione.

Classe: Percorso

Metodi	Attributi acceduti
calcola_percorso1	navPathStack, indicazioni, velocita_media, nodeFrom, nodeTo, emergency

get_posizione_iniziale	nodeFrom
get_destinazione	nodeTo
set_posizione_iniziale	nodeFrom
set_destinazione	nodeTo
nuova_posizione_utente	nodeFrom
calcola_percorso2	navPathStack2, indicazioni, velocita_media, nodeFrom, nodeTo, emergency
Valore LCOM4 = 1	

Classe: **Nodo**

Metodi	Attributi acceduti
Nodo	x, y, punto_di_raccolta vicino, piano, id, name
getY	y
getX	x
getName	name
getPiano	piano
getId	id
getRaccolta_vicino	punto_di_raccolta_vicino
setX	x
setY	y
setName	name

setPiano	piano
setId	id
setRaccolta_vicino	punto_di_raccolta_vicino
Valore LCOM4 = 1	

Classe : Arco

Metodi	Attributi acceduti
Arco	V, I , LOS, C, L_Arco, nodo0, nodo1, scala, piano
aggiorna_variabili	V, I, LOS, C
get_variabili	V, I, LOS, C
get_L_arco	L_Arco
calcola_costo_K	V, I, LOS, C, L_Arco, costo_K
get_costo_K	costo_K
Valore LCOM4 = 1	

Classe : Mappa

Metodi	Attributi acceduti
get_costanti	piano, pv, pi, plos, pc
Valore LCOM4 = 1	

Classe : Utente

Metodi	Attributi acceduti
Utente	username, password
get_username	username
get_password	password
set_username	username
set_password	password
set_posizione	posizione
get_posizione	posizione
Valore LCOM4 = 2	

Classe : Server

Metodi	Attributi acceduti
aggiorna_posizione_utente	posizione_utente
comunicare_emergenza	emergenza
controllo_credenziali	username, password
Valore LCOM4 = 3	

Come si può notare quasi tutte le classi, ad eccezione di Server ed Utente, possiedono un valore di LCOM4 uguale a 1. Questo significa che le classi godono in generale di una buona coesione interna, cioè che modellano un unico concetto astratto e che risultano più maneggevoli, manutenibili e testabili minimizzando il rischio di commettere errori. Ad eccezione delle classi server, che è tenuta in considerazione per l'interazione con

l'applicazione ma che non verrà modellata direttamente in questo studio, ed Utente, per il quale non è possibile istanziare una posizione prima dell'avvenuta autenticazione, tutti i valori della metrica risultano essere minimi e di conseguenza non risulta necessario intraprendere alcuna azione correttiva rispetto alla coesione interna.

5.3.5 Interdipendenza - CBO

Di seguito la Tabella con i valori di CBO (Coupling Between Objects) per ciascuna classe presente nel modello di progettazione.

Classe	Dipendenze in Uscita	valore CBO
Utente	IntUtente	1
GuiUtente	Utente	1
Percorso	Utente, Mappa, IntNavigazione	3
Mappa	IntCollegamenti	1
Nodo	Percorso, Arco, IntCollegamenti	3
Arco	Percorso, Mappa, IntCollegamenti	3
Server	Arco, Utente	2

Dai valori riportati sopra, si evince come in generale il livello di accoppiamento sia molto basso, arrivando addirittura a valori nulli. Valori bassi di CBO modellano classi con un basso valore di interdipendenza (o accoppiamento) e di conseguenza più semplici da testare e manutenere, poiché dipendono da un numero minore di oggetti. Si ritiene, dunque, che i valori trovati per le classi di progettazione siano buoni e che non siano necessari aggiustamenti ulteriori.

5.3.6 Complessità - WMC

Attraverso la metrica WMC (Weighted Method per Class) è possibile avere una stima della complessità di ciascuna classe e quindi della sua manutenibilità e della possibile presenza di errori. Studi empirici hanno infatti dimostrato che più aumenta la complessità di una classe più si è inclini a commettere errori. La stima per la sopracitata metrica può essere calcolata attribuendo ad ogni metodo di ogni classe un peso e sommando i pesi dei metodi presenti in ogni classe.

Il calcolo dei pesi può essere effettuato attraverso approssimazione, assegnando ad ogni metodo un peso uguale ad 1, o calcolato in maniera più esaustiva attraverso la complessità ciclomatica. A questo stadio non si è ancora prodotto codice, quindi risulta impossibile costruire il grafo relativo ad ogni metodo su cui calcolare la complessità. Per tale motivo, come esposto anche nel paper NASA¹ sullo studio di metriche per la programmazione ad oggetti, si è deciso di assegnare un peso standard di 1 ad ogni metodo e procedere con i calcoli.

Classe	Metodi	WMC
Utente	Utente, get_username, get_password, set_username, set_password, get_posizione, set_posizione	7
GuiUtente	inserimento_credenziali, scelta_percorso, aggiorna_posizione, seleziona_modalita_inserimento, forografare_QR, tap_sulla_mappa, scrivere_indirizzo_stanza, scelta_destinazione, notifica_emergenza, visualizza_percorso_aggiornato,	14

¹ Laing, Victor, and Charles Coleman. "Principal Components of Orthogonal Object-Oriented Metrics." *White Paper Analyzing Results of NASA Object-Oriented Data (323-08-14)* (2001).

	modalita_emergenza, seleziona_modalita_navigazione, bypass_autenticazione, visualizza_login	
Percorso	calcola_percorso1, get_posizione_iniziale, set_posizione_iniziale, get_destinazione, set_destinazione, nuova_posizione_utente, calcola_percorso2, calcola_indicazioni	8
Mappa	get_costanti	1
Nodo	Nodo, getY, getX, getName, getPiano, getId, getRaccolta vicino, setX, setY, setName, setPiano, setId, setRaccolta vicino	13
Arco	Arco, aggiorna_variabili, get_variabili, get_L_arco, calcola_costo_K, get_costo_K	6
Server	aggiorna_posizione_utente , comunicare_emergenza, controllo_credenziali	3

Generalmente si considera il valore di WMC=10 come soglia da non superare. Nel caso in esame si nota come tutte le classi risultino molto al di sotto di tale valore, eccezione fatta per la classe che modella le operazioni della GUI e per la classe Nodo. Per quanto riguarda la prima, essendo questa una trasposizione teorica di operazioni che sarà l'utente a richiedere ed a effettuare sul dispositivo, si è ritenuto che non avrebbe avuto molto senso partizionarla. Rispetto alla classe Nodo invece, si nota come la maggior parte dei metodi presenti sia di get e set, quindi metodi estremamente semplici che

servono a reperire o settare i valori degli attributi di un oggetto. Per tali motivi si è deciso di lasciare inalterato lo schema di progettazione e proseguire con l'implementazione dell'applicazione.

6. REALIZZAZIONE

Una volta conclusa la fase di analisi e progettazione, ci siamo dedicati alla realizzazione vera e propria dell'applicazione. Come già detto ampiamente, dai requisiti risulta che l'applicazione debba funzionare su sistema operativo mobile Android. Per questo ci siamo documentati su quale fosse il miglior IDE da utilizzare per la programmazione. La nostra scelta è ricaduta sull'IDE ufficiale Android Studio, che vanta anche la possibilità di utilizzare un emulatore di vari modelli di smartphone per testare ampiamente l'app. Si è utilizzata la versione 2.0 (da poco rilasciata in versione stabile) per poter sfruttare le potenzialità espresse dal changelog di questa versione.

Android Studio

The Official IDE for Android

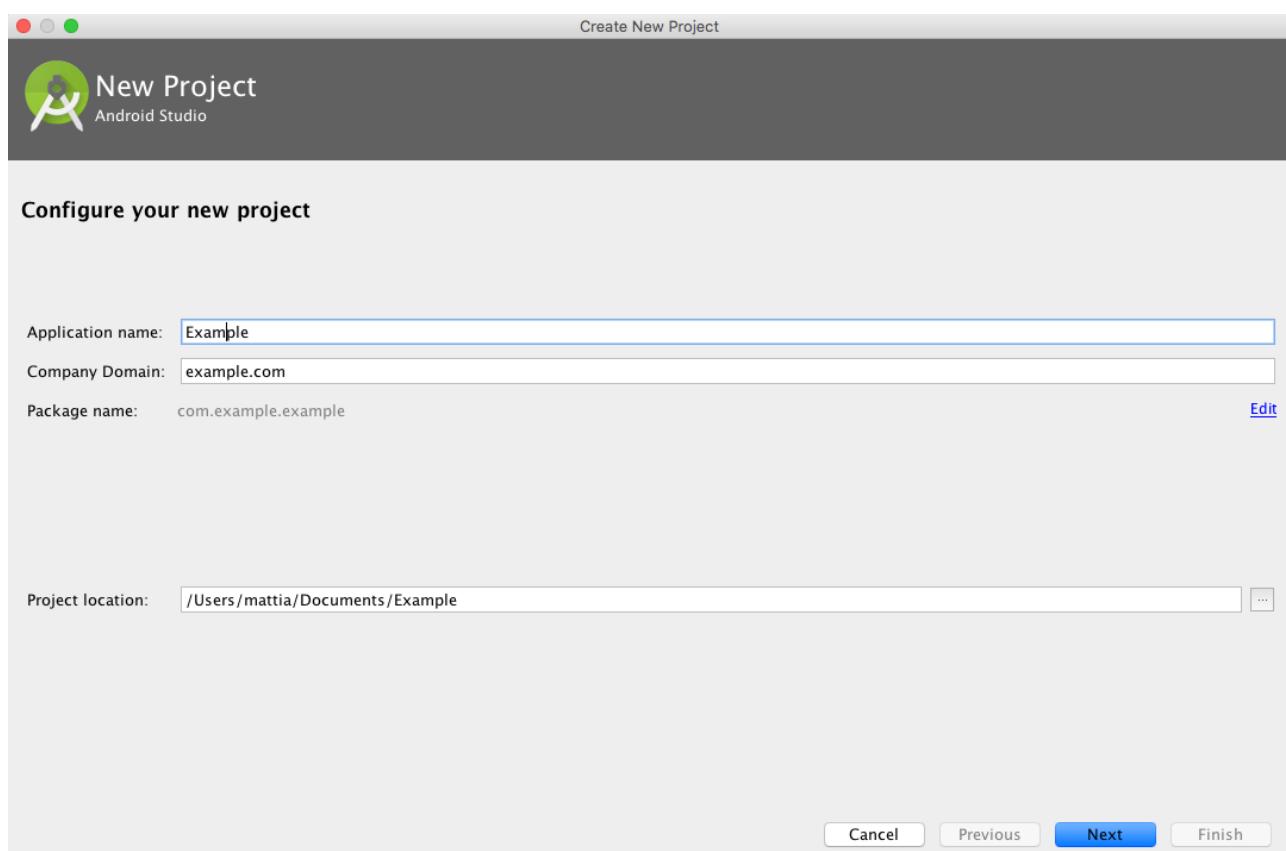
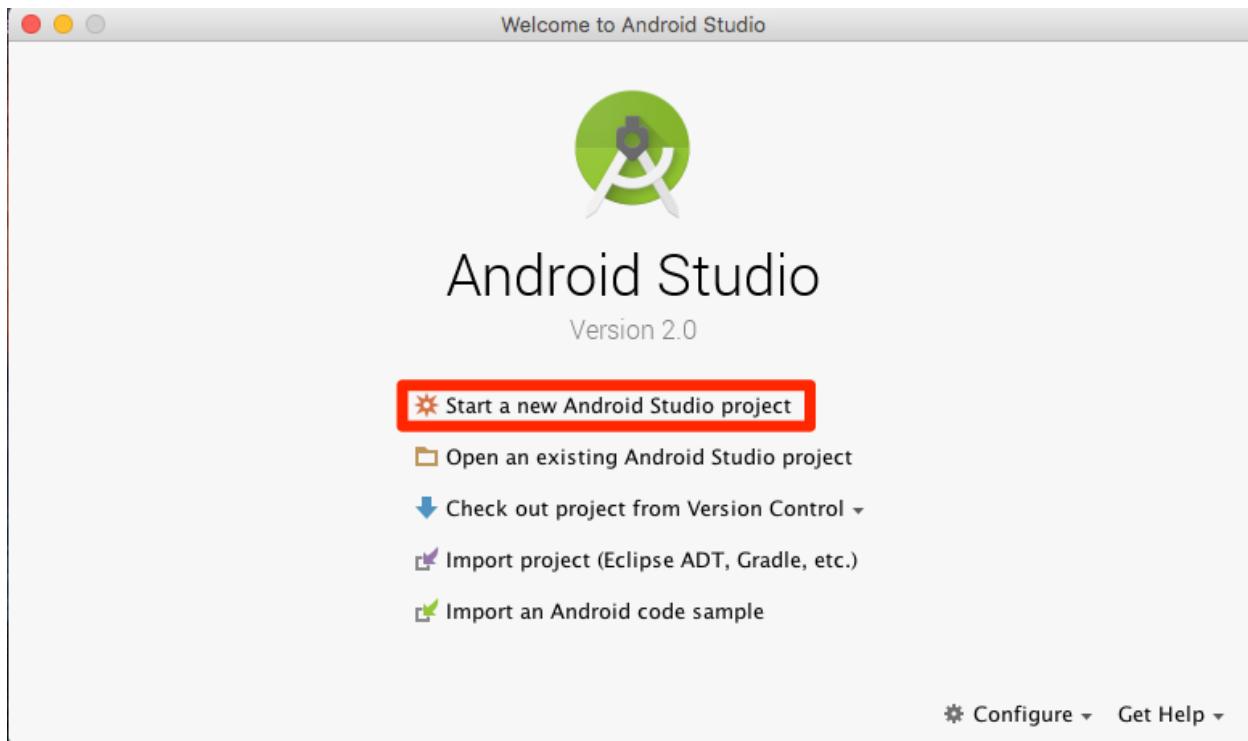
Android Studio provides the fastest tools for building apps on every type of Android device.

World-class code editing, debugging, performance tooling, a flexible build system, and an instant build/deploy system all allow you to focus on building unique and high quality apps.

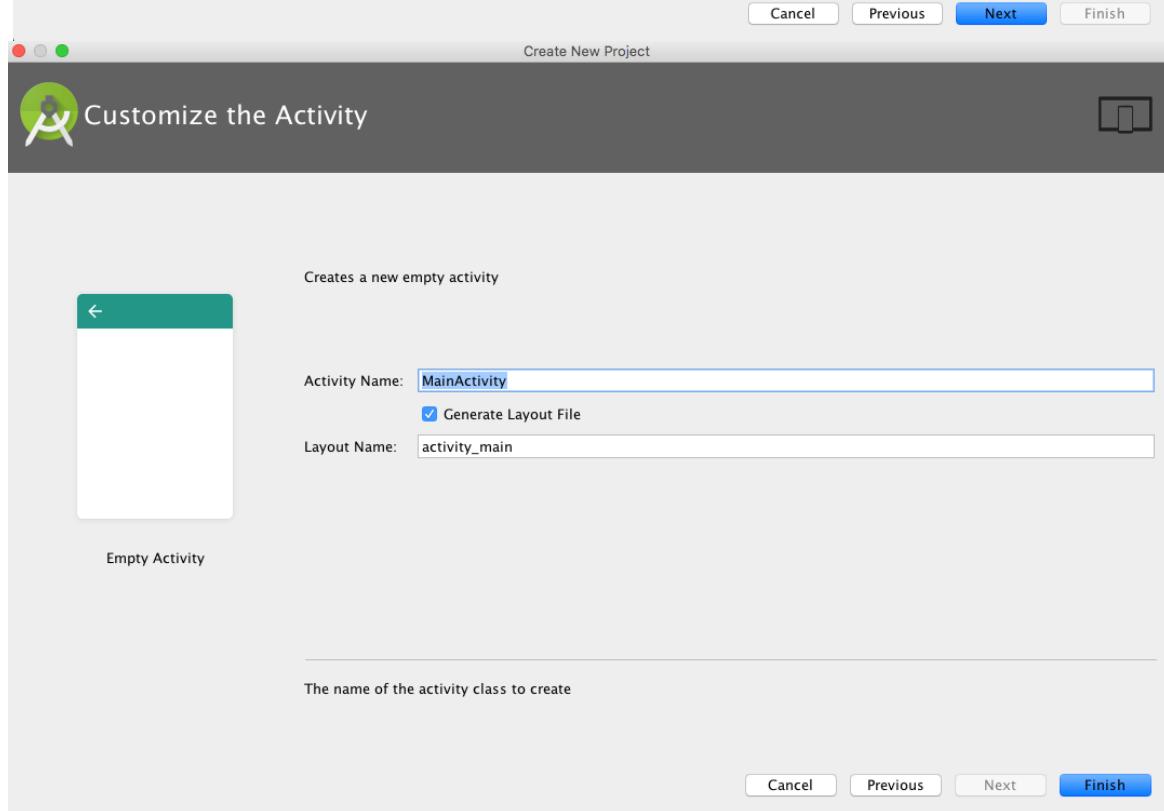
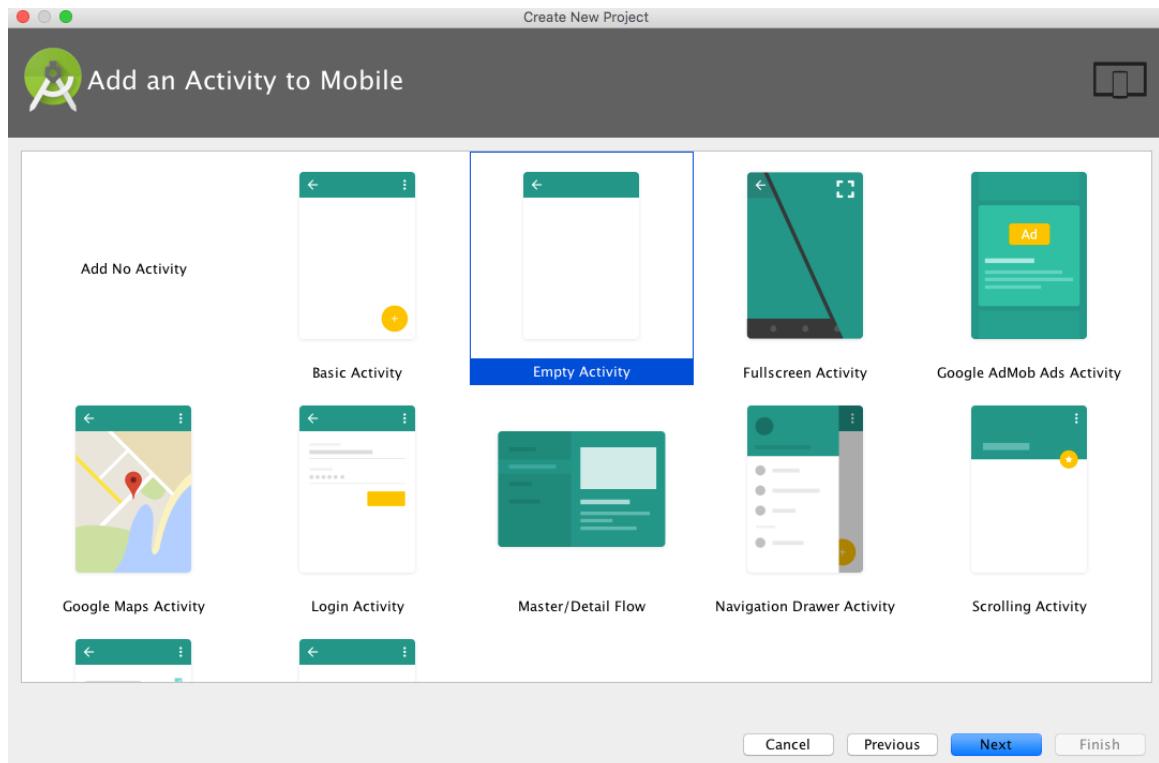
[DOWNLOAD ANDROID STUDIO 2.0 FOR WINDOWS \(1166 MB\)](#)



Una volta scaricato e installato come riportato sul manuale d'installazione si è creato un nuovo progetto.

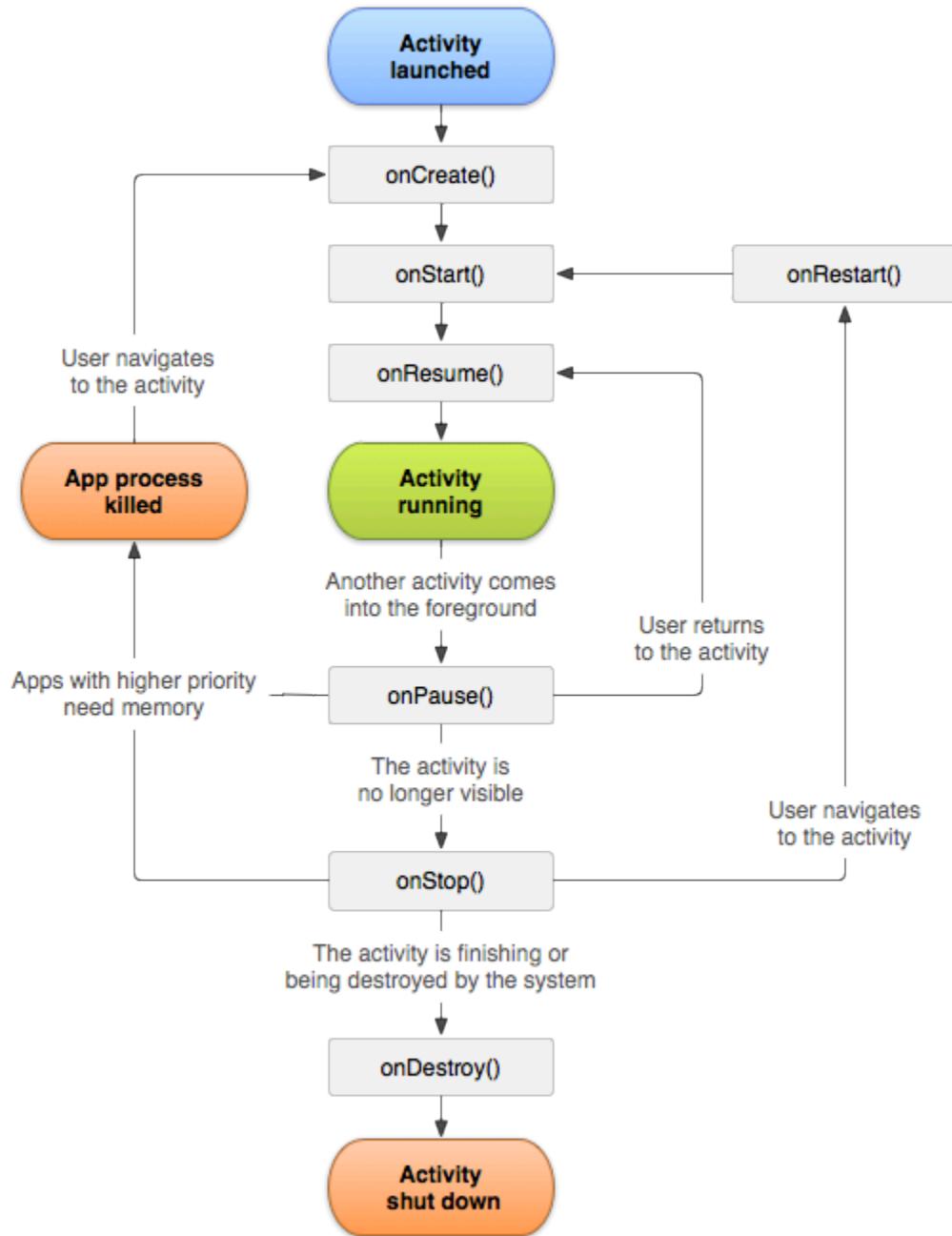


Una volta assegnato il nome al nuovo progetto, si passa a creare il fulcro di un programma Android: le activity.



Esse rappresentano proprio quello che l'utente può fare tramite l'interazione con l'applicazione, e inglobano una cosiddetta View (la UI) che può essere di diverse tipologie.

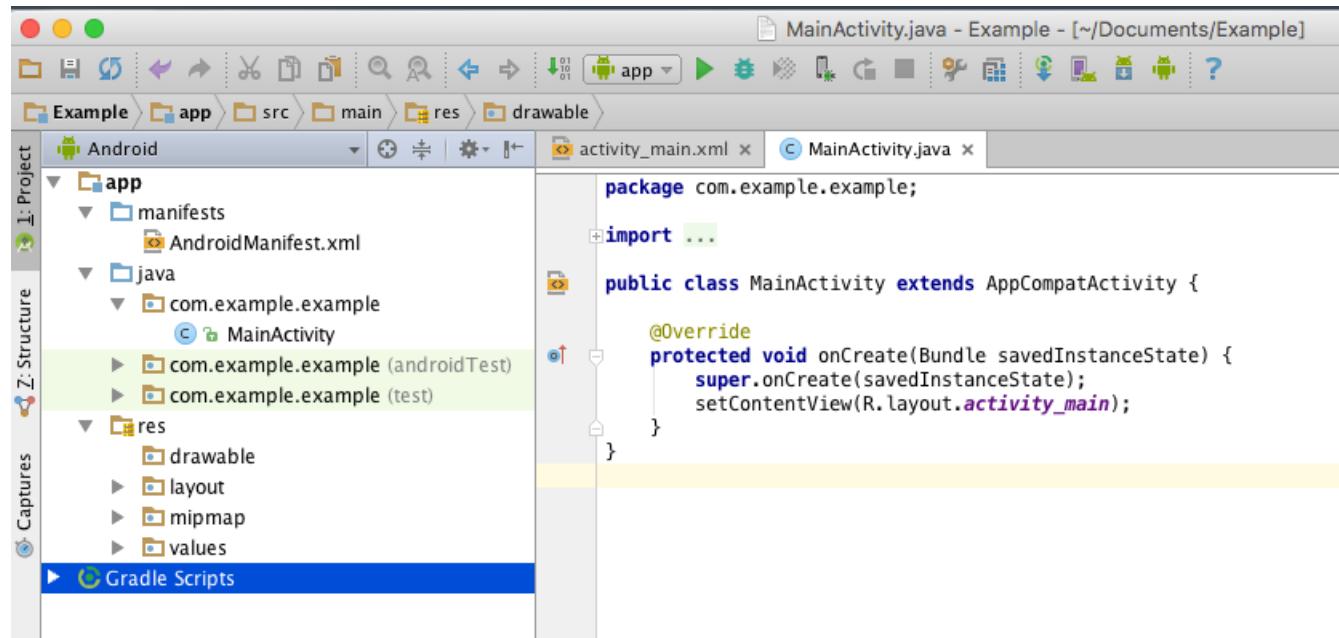
Le varie activity sono gestite dal sistema come uno Stack. Una volta avviata un'activity viene quindi messa in cima allo Stack e diventa quella in esecuzione. Il ciclo di vita completo di un'activity viene riportato di seguito, includendo i metodi che possono essere utilizzati in Override per modificarne il comportamento di default.



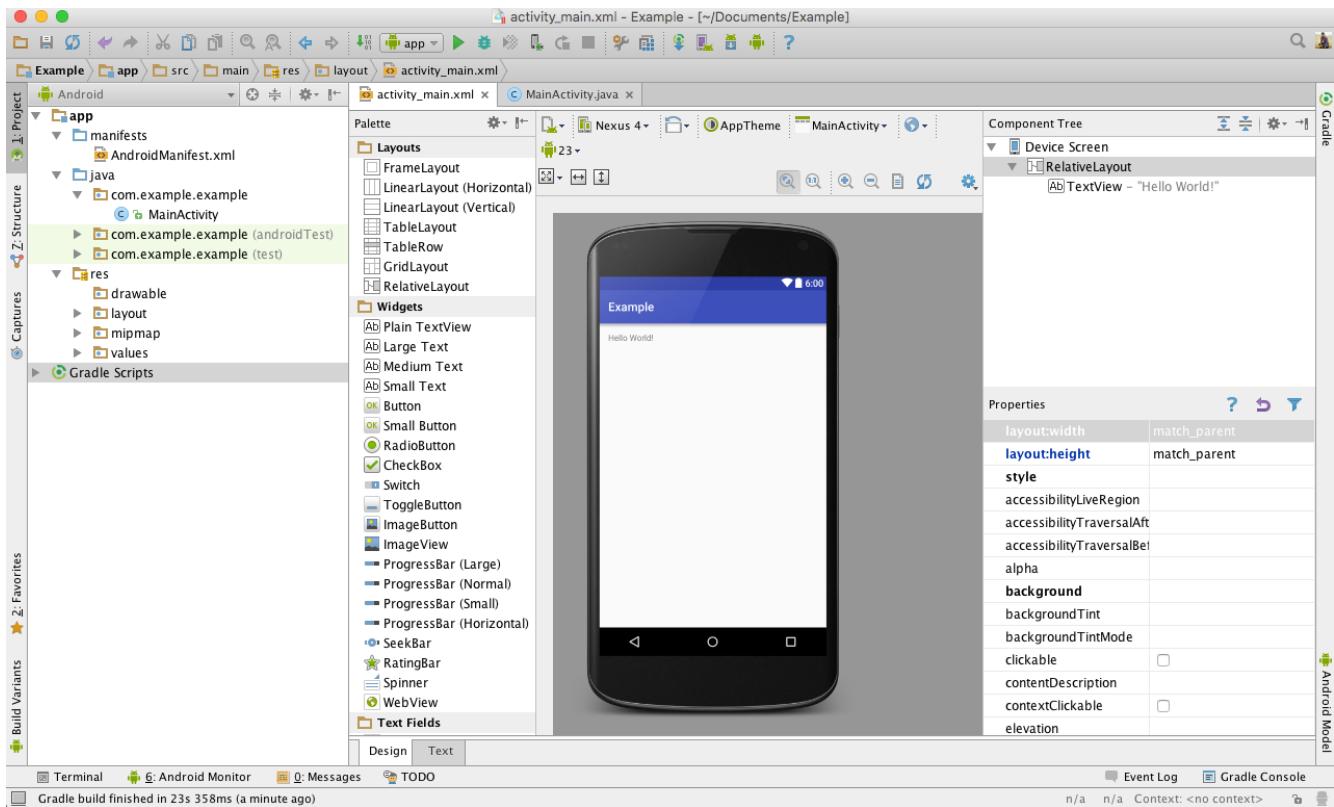
La struttura di un progetto Android può essere analizzata nella figura seguente:

-Troviamo i moduli (in questo caso solo l'app e gli script Gradle), tra cui si possono aggiungere altri relativi ad esempio ad Android Wear, Android TV, ecc..

-All'interno dei singoli moduli, si ritrovano i file necessari all'esecuzione dell'app sulla piattaforma desiderata. Tra questi vi è il Manifest, in cui si dichiarano i permessi, le activity che si utilizzeranno, ecc..; i Package che includeranno le classi di progettazione; e i file adibiti ai test. Inoltre è presente una cartella "res", che contiene al suo interno tutti i file di appoggio alle varie classi, strutturati a seconda del loro utilizzo.



L'activity di esempio include una View importata da un file di Layout, che grazie all'IDE di Android, può essere modificato o tramite un file xml, o grazie ad un'interfaccia di Design molto intuitiva e ricca di funzionalità.



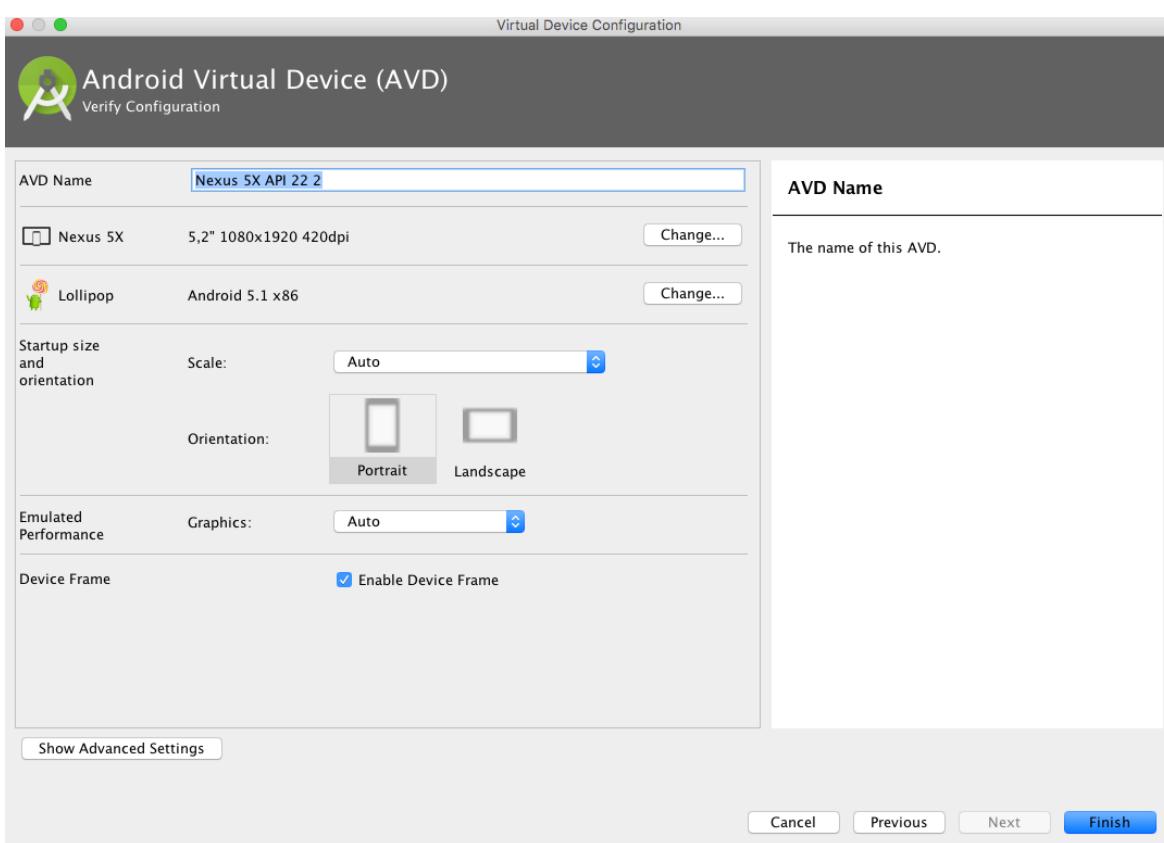
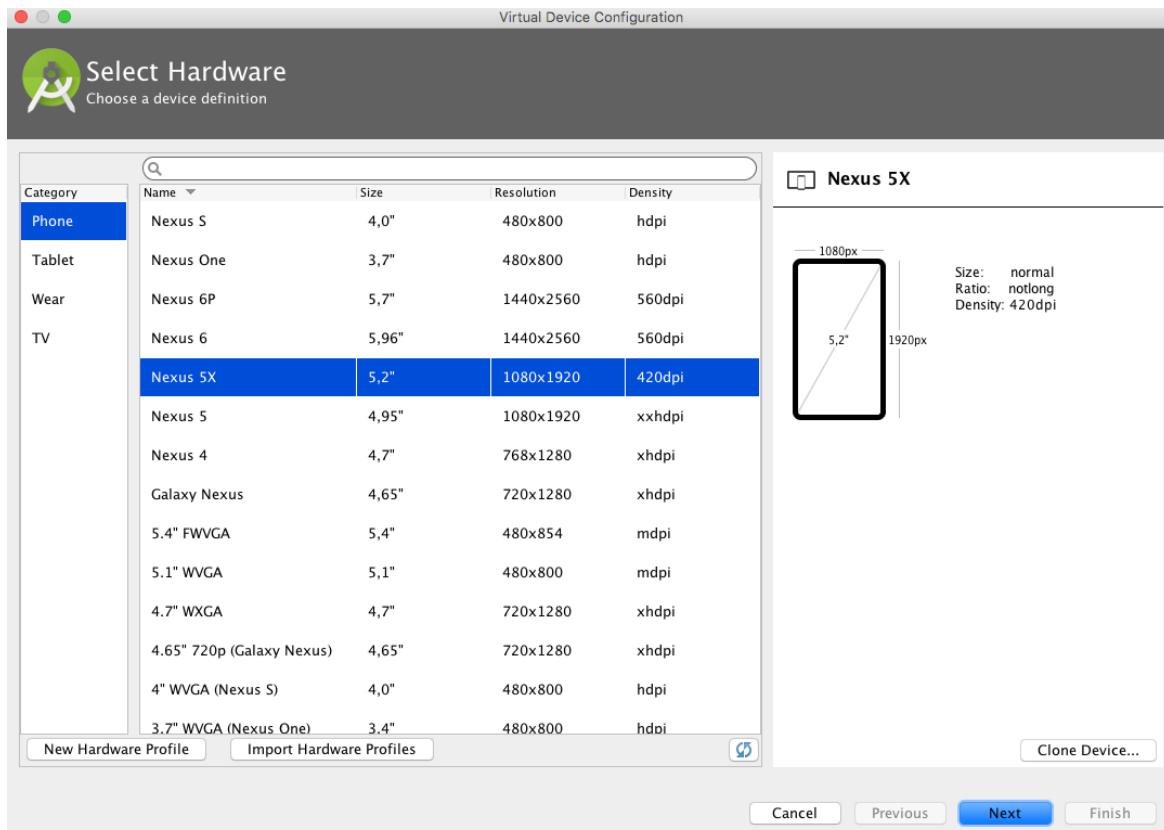
Per poter poi testare l'applicazione durante il suo sviluppo, abbiamo utilizzato due modalità:

- Tramite debug USB sui nostri smartphone,
- Tramite emulatore fornito da Android Studio.

Questa seconda modalità è risultata più versatile, perché dà la possibilità di testare l'applicazione su diversi device, pur non possedendoli, e allo stesso tempo di lavorare e simulare sullo stesso PC utilizzato.

Per poterla configurare bisogna cliccare sull'icona AVD Manager, presente sulla barra in alto dell'IDE.

Quindi, si procede con lo scegliere lo smartphone da emulare e le sue caratteristiche, compresa la versione di Android utilizzata.



Una volta impostato l'emulatore (o collegato il proprio smartphone in modalità debug), si può lanciare l'applicazione col tasto apposito (), e vedere subito il risultato della programmazione.



Per quanto riguarda le indicazioni su Android Wear, è stato utilizzato, per testarne la funzionalità, un Sony Smartwatch 3 con Android 5.1.1, accoppiato con uno smartphone come descritto nel manuale d'installazione.

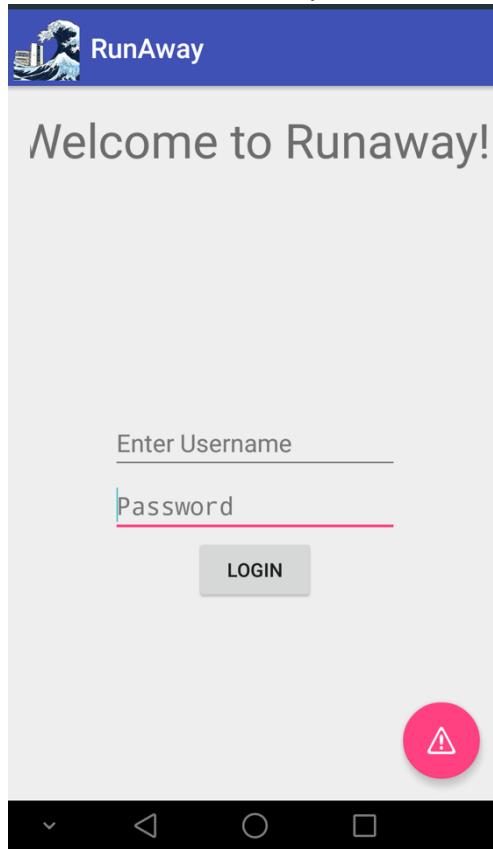
7. TEST

Nel seguito verranno effettuati i test programmati in fase di analisi, basati sullo schema input-output descritto nelle task card.

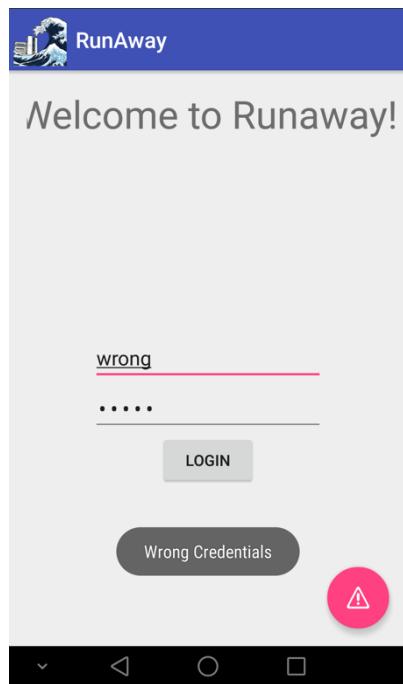
7.1 Login utente

Task ID: 1.5 Verifica Credenziali

In questo test si vuole far vedere come l'app si comporta all'inserimento di credenziali per il login, sia nel caso siano errate, sia nel caso siano giuste, ed in un questo secondo scenario si testa il corretto salvataggio delle stesse per la successiva apertura dell'app. Questa è la schermata che l'utente ottiene all'apertura dell'app per la prima volta.



Inserendo delle credenziali sbagliate, e cliccando sul pulsante LOGIN, non si riesce ad accedere alle altre funzionalità e viene restituito un messaggio di errore.

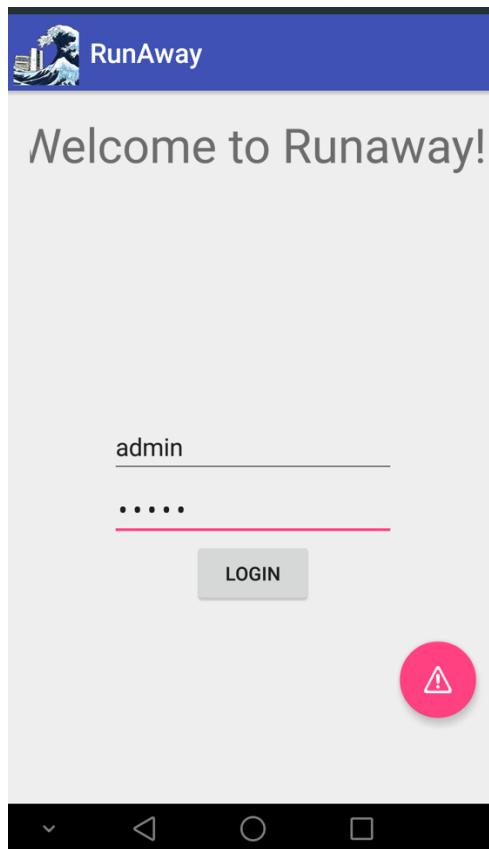


Andando invece ad inserire le credenziali corrette, che nel nostro caso sono:

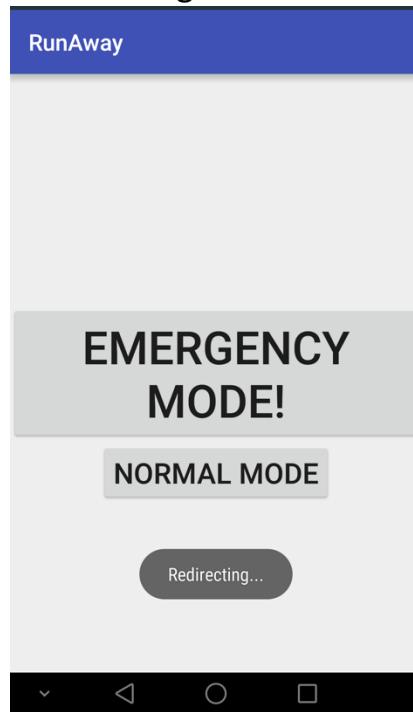
Username: admin

Password: admin

e selezionando il tasto LOGIN



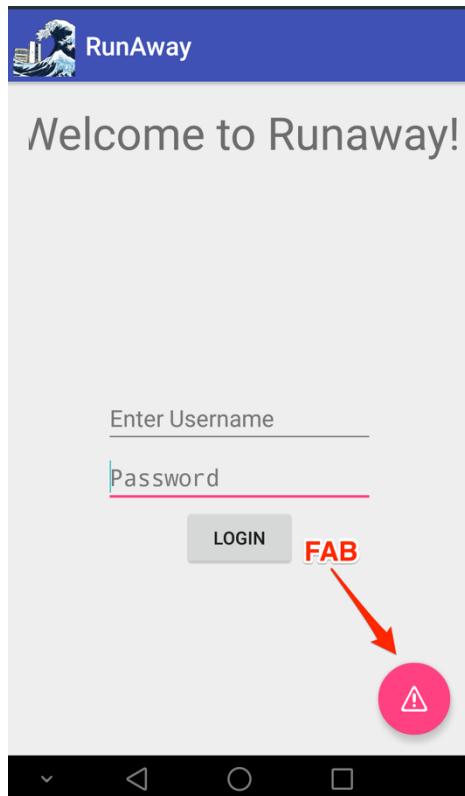
Si riesce quindi ad accedere alle piene funzionalità dell'app, venendo reindirizzati alla pagina per la scelta della modalità di navigazione.



A questo punto, chiudendo l'app e riaprendola successivamente, si verrà autenticati automaticamente e comparirà direttamente la schermata di scelta della modalità di navigazione.

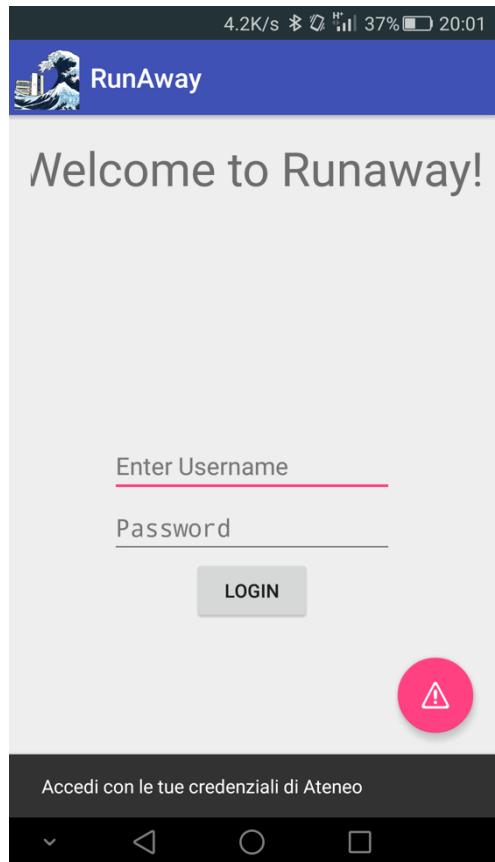
Task ID: 1.6 Verifica funzionamento FAB

All'utente, in caso di emergenza, viene data la possibilità di bypassare il login solo nel caso in cui il server non abbia la connessione disponibile. Questo attraverso il tasto FAB presente in basso a destra della schermata di login.

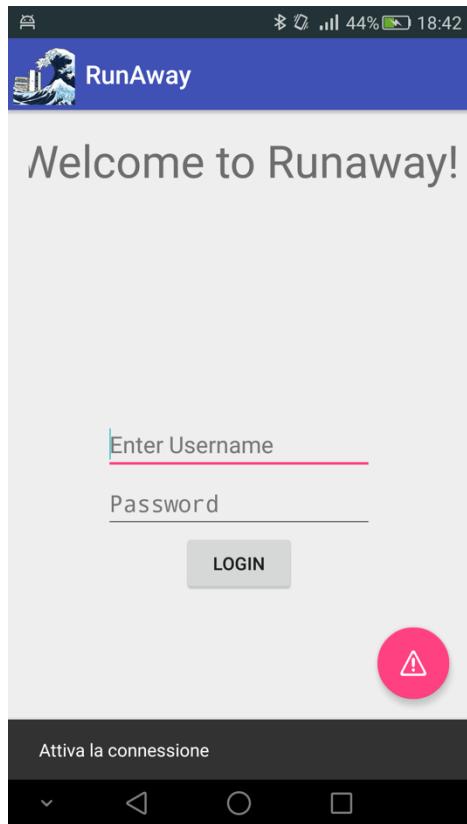


Tappando su questo pulsante possono accadere sostanzialmente tre cose:

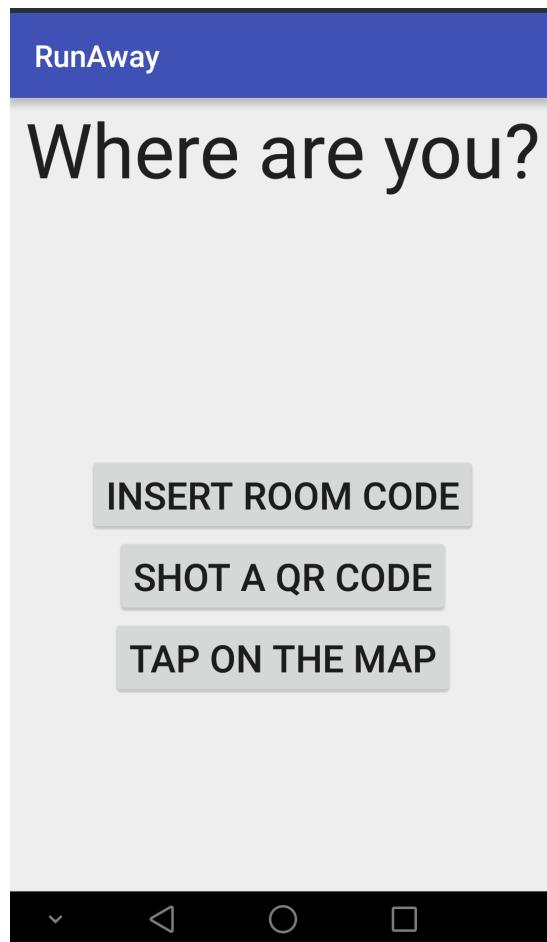
-Se il server è disponibile (lo smartphone è collegato alla rete WIFI di Ateneo), allora viene notificata la possibilità di effettuare il login con le credenziali di Ateneo. Per questo test abbiamo inserito un ping nel codice verso www.google.it, per simulare la disponibilità del Server.



-Se lo smartphone non ha alcuna connessione attiva, viene richiesto di attivare la connessione.



-Altrimenti, se il Server non è disponibile ma lo smartphone è collegato alla rete locale, allora si può bypassare il login e l'utente può usufruire della navigazione verso il punto di raccolta più vicino, accedendo direttamente alla modalità di emergenza.



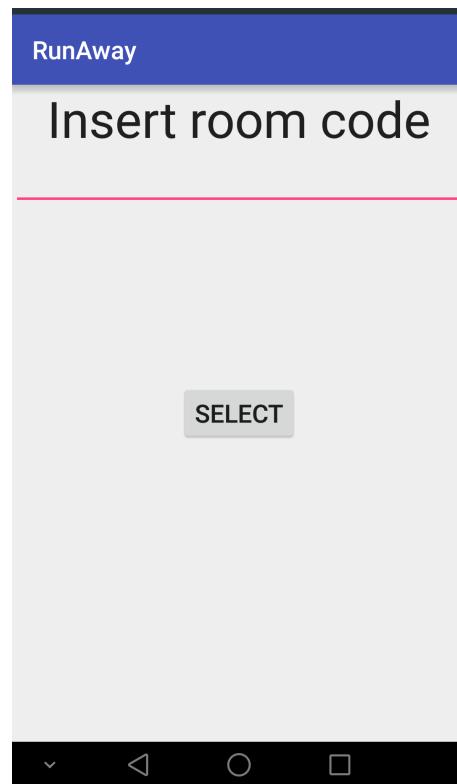
7.2 Invio e aggiornamento propria posizione sul server

Task ID: 2.5 Verifica funzionamento pulsanti per inserimento posizione

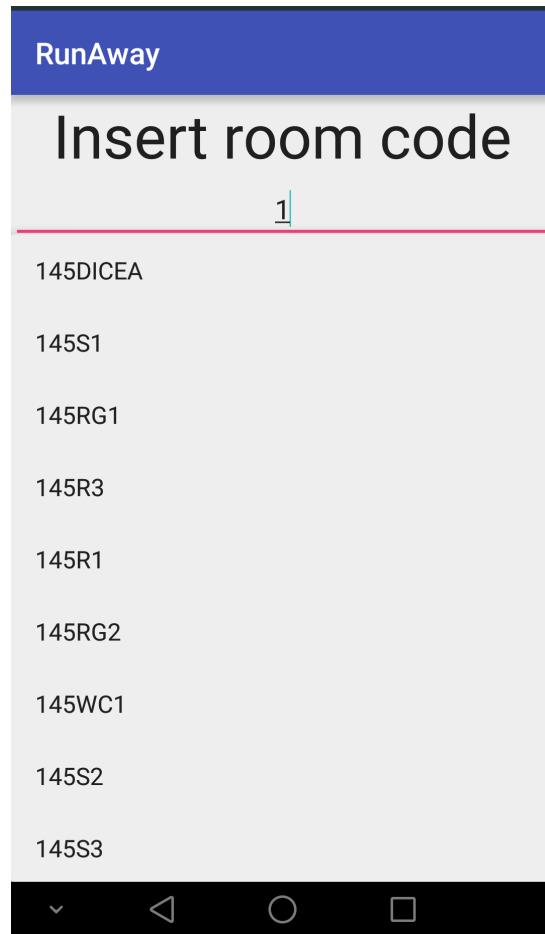
In questo task si esaminerà la corretta gestione dei pulsanti per stabilire la posizione iniziale dell'utente, visibili nella schermata precedente, subito dopo aver selezionato la modalità di navigazione. La possibilità di selezione offerta all'utente è la lista dei nodi presenti all'interno del database. I codici definiscono inoltre il tipo di nodo considerato:

- Con EM s'intende un'uscita di emergenza;
- Con U s'intende un'uscita generica (quindi anche di emergenza);
- Con A s'identificano i nodi inclusi per le scale;
- Con R vengono rappresentate le aulee;
- Sono poi presenti alcuni codici interni alla facoltà (dipartimenti o sigle).

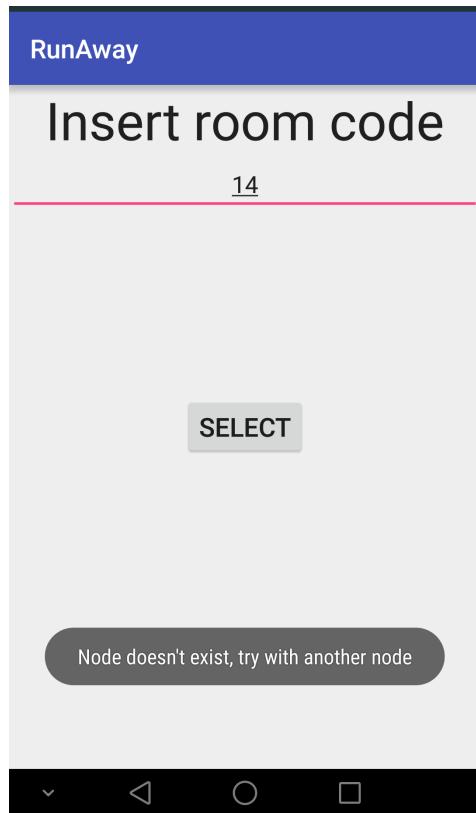
- Insert room code: permette all'utente di inserire il codice di riferimento della stanza.



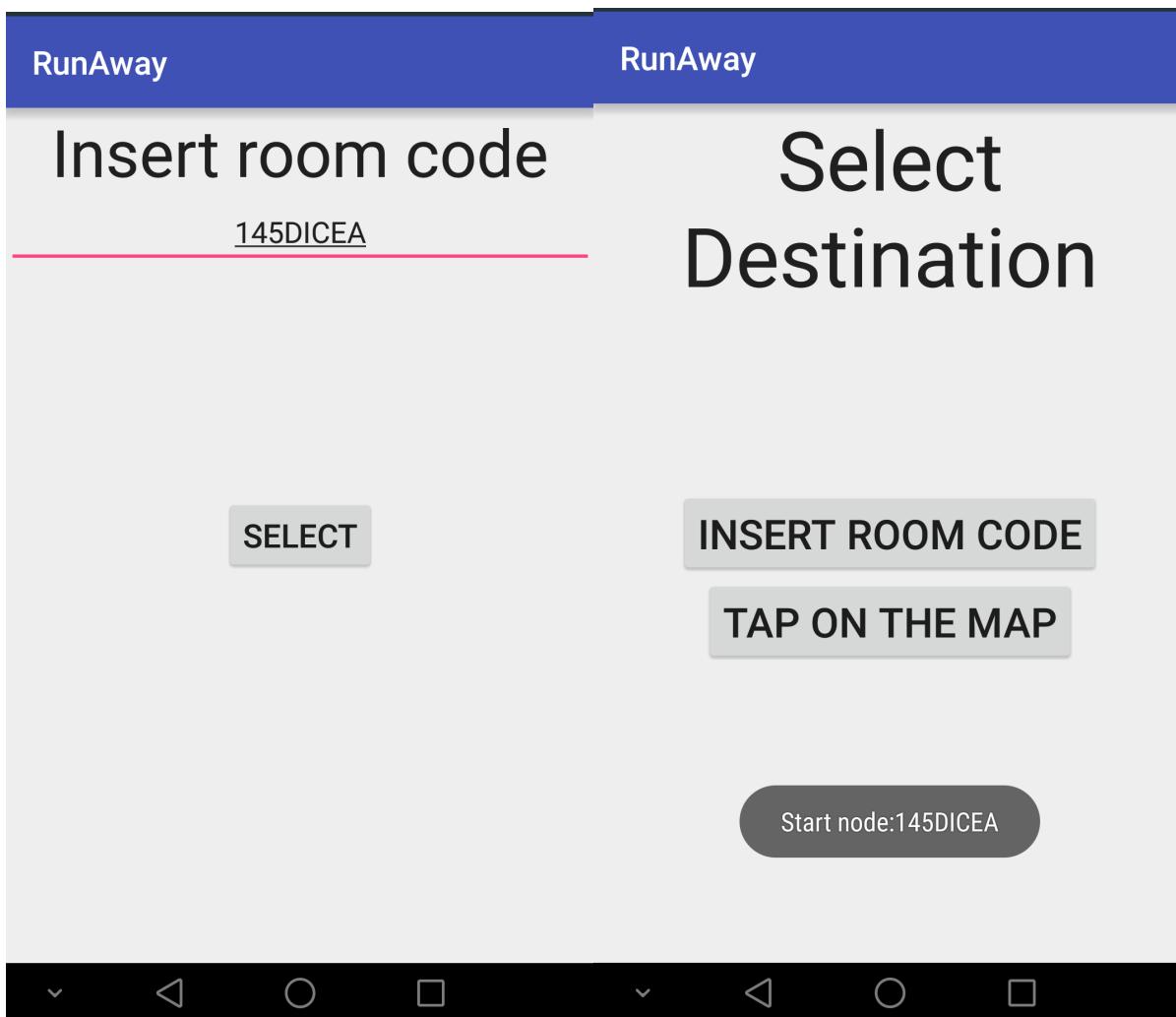
E' presente una casella di testo con auto riempimento, in cui viene suggerita una lista di nodi concordi con quello che si sta scrivendo.



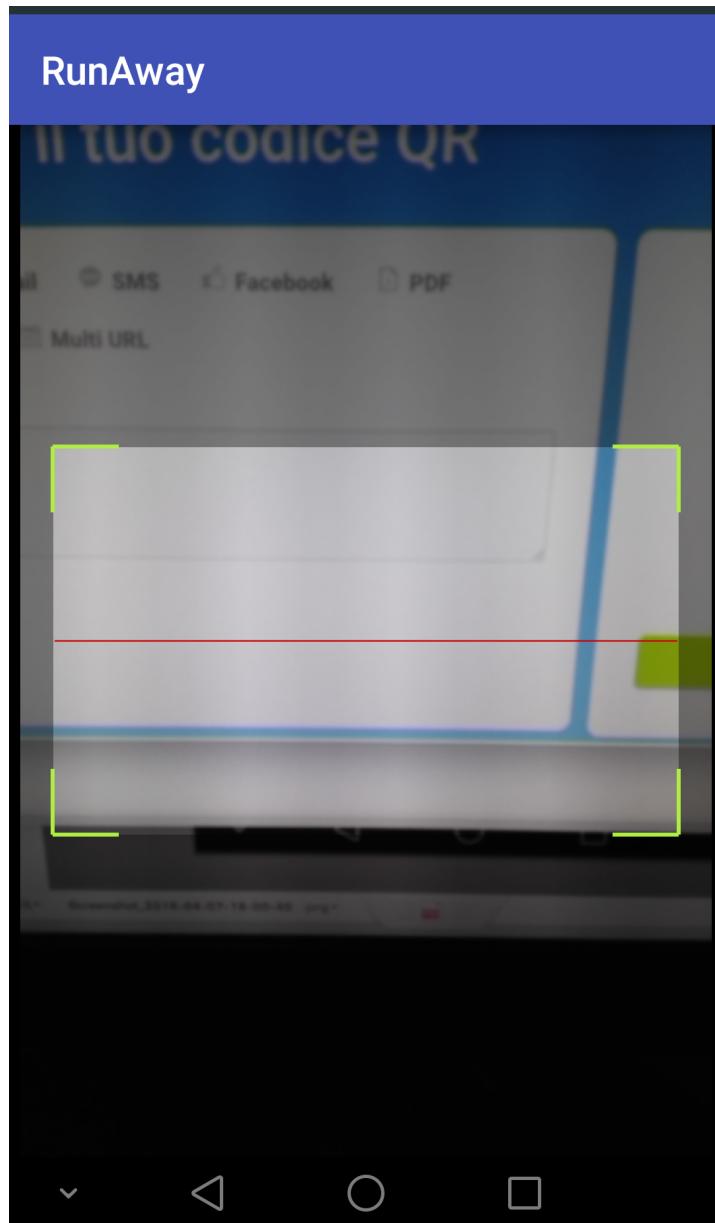
Se si tenta ora di cliccare sul tasto SELECT, avendo inserito un codice non presente all'interno del database (in questo caso si è scelto di inserire solo il numero 14), comparirà il seguente messaggio di errore:



Se invece si seleziona un nodo dalla lista o si scrive il codice corretto, testiamo che venga selezionato come nodo di partenza, giungendo quindi alla schermata per selezionare il nodo di destinazione.



- All'utente viene data anche la possibilità di stabilire la propria posizione attraverso la scannerizzazione di un QR code (supposto che ne sia presente uno di solo testo con il codice di riferimento su ogni nodo). Tappando quindi su questa opzione si aprirà una schermata con la fotocamera del dispositivo abilitata. Per il test sono stati generati online due QR code: uno con il solo testo 145S (codice errato), ed uno con 145S1 (codice esatto), per verificare che l'applicazione si comporti correttamente.



Generando quindi il QR code errato ed inquadrandolo, l'applicazione dovrebbe rispondere con una notifica di errore, ma consentendo ancora di poter inquadrare il QR code giusto.

Generatore di codici QR - C X

it.qr-code-generator.com

GET QR CODE GENERATOR PRO

ACCEDE REGISTRAZIONE

Crea gratuitamente il tuo codice QR

URL VCard Testo E-mail SMS Facebook PDF

MP3 Appstore Immagini Multi URL

Testo libero

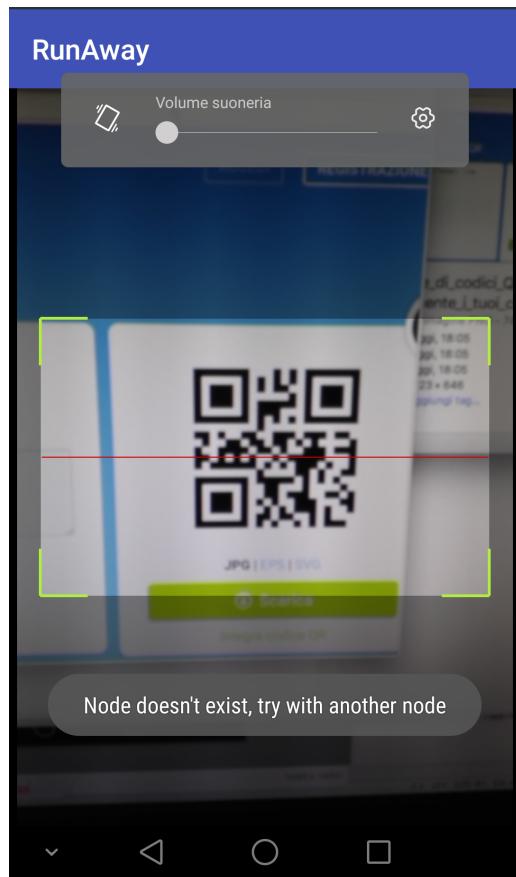
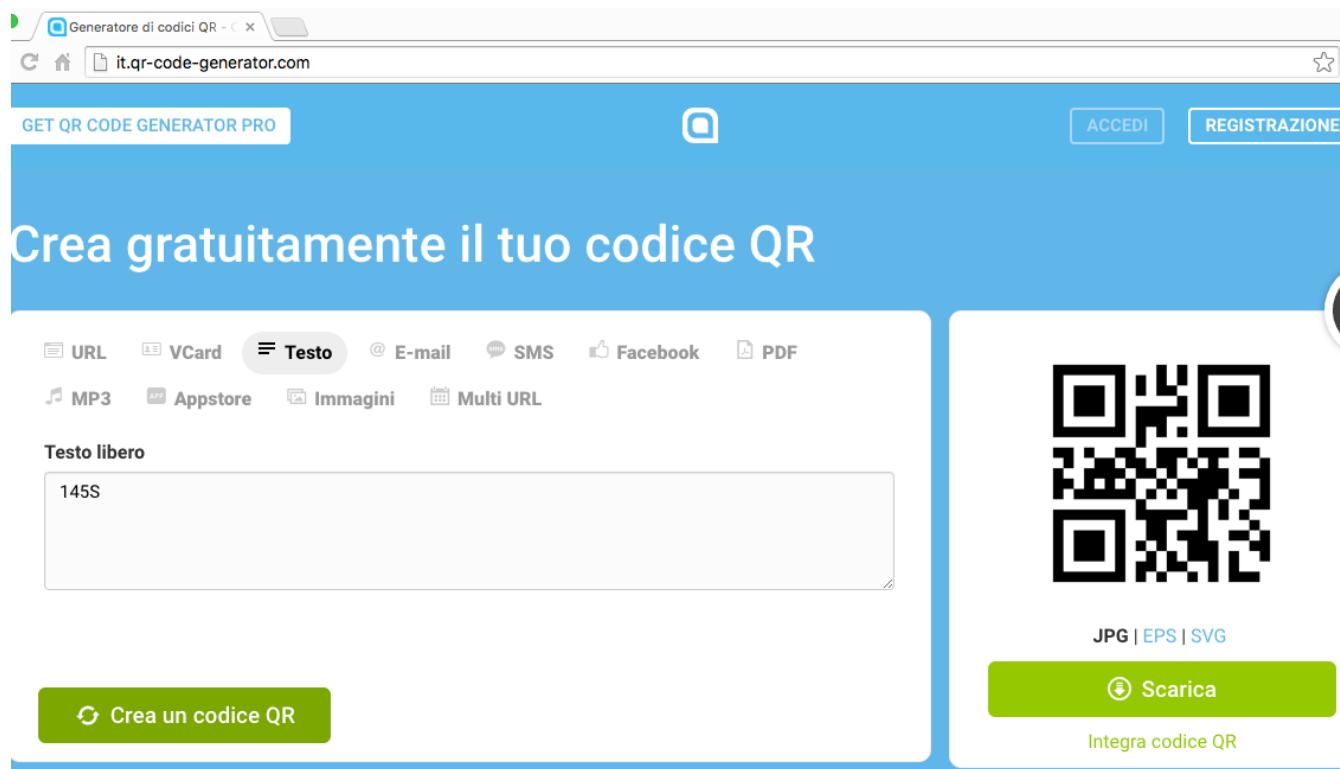
145S

Crea un codice QR

JPG | EPS | SVG

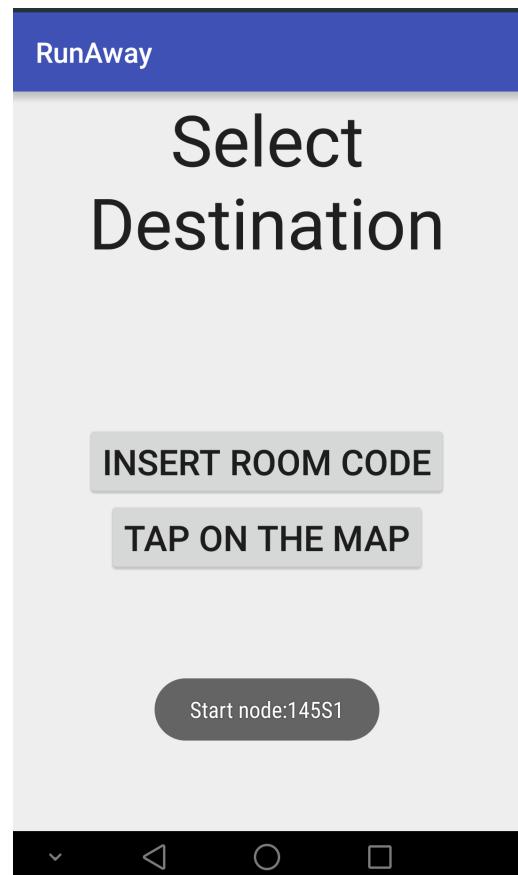
Scarica

Integra codice QR

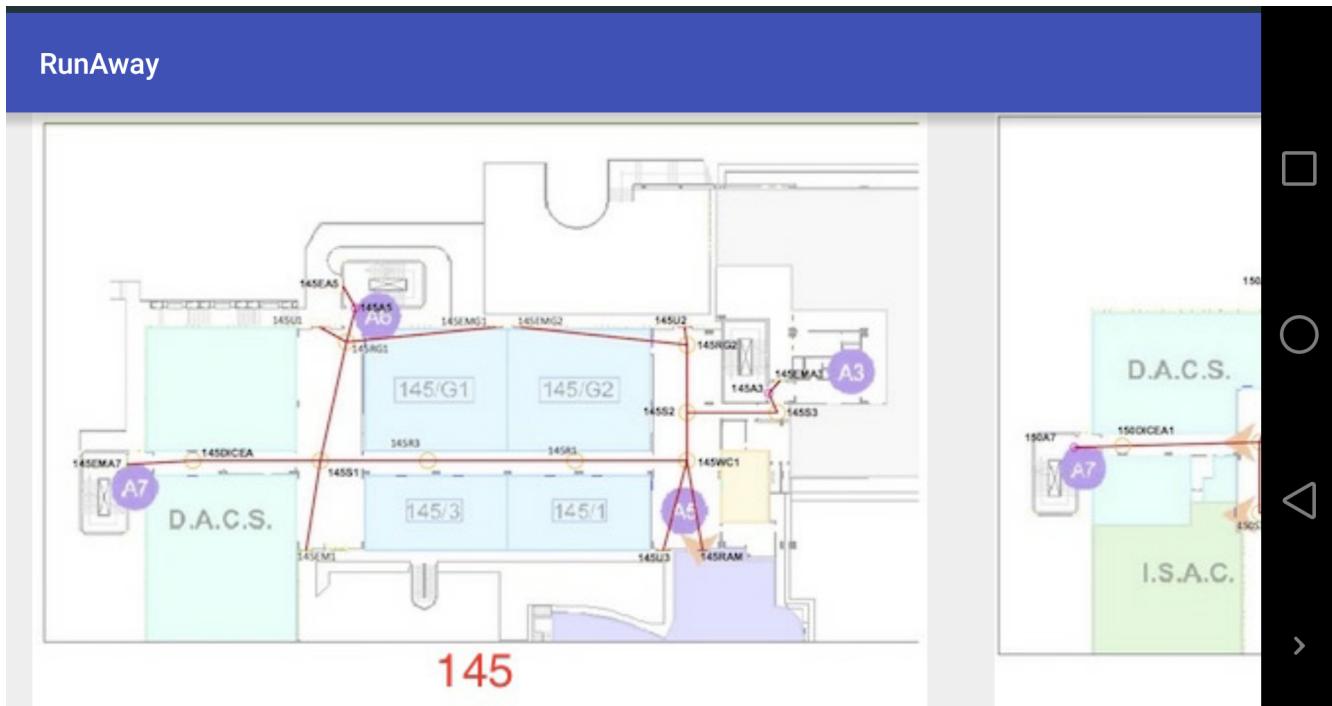


Generando ed inquadrando il QR code corretto invece, quello che ci si aspetta è il settaggio del nodo di partenza, e la comparsa quindi della schermata per la selezione del nodo di destinazione.

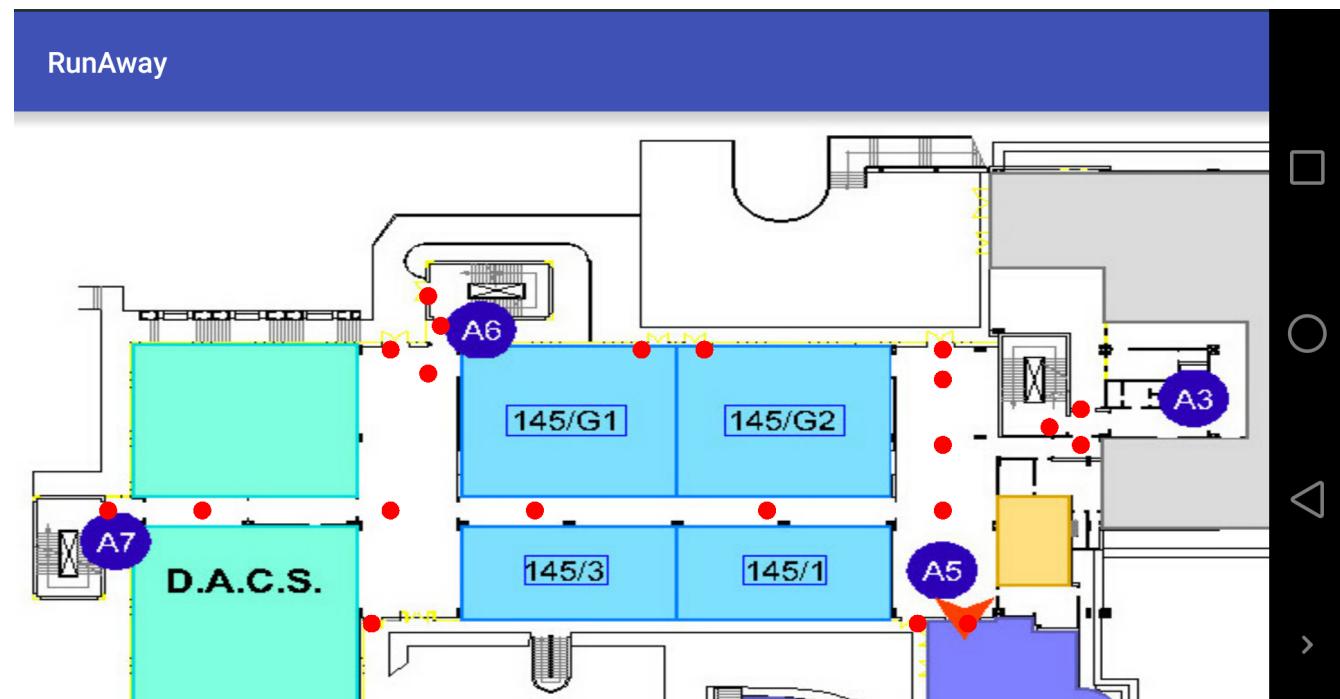




- L'ultima possibilità per stabilire la posizione iniziale è quella di tappare la posizione del nodo desiderato direttamente sulla mappa del piano di riferimento. A tal proposito, alla scelta di questa modalità, verrà aperto un carosello, con scorrimento orizzontale, contenente i piani a disposizione, con un anteprima dei nodi e dei possibili percorsi, per poter avere una visione d'insieme.

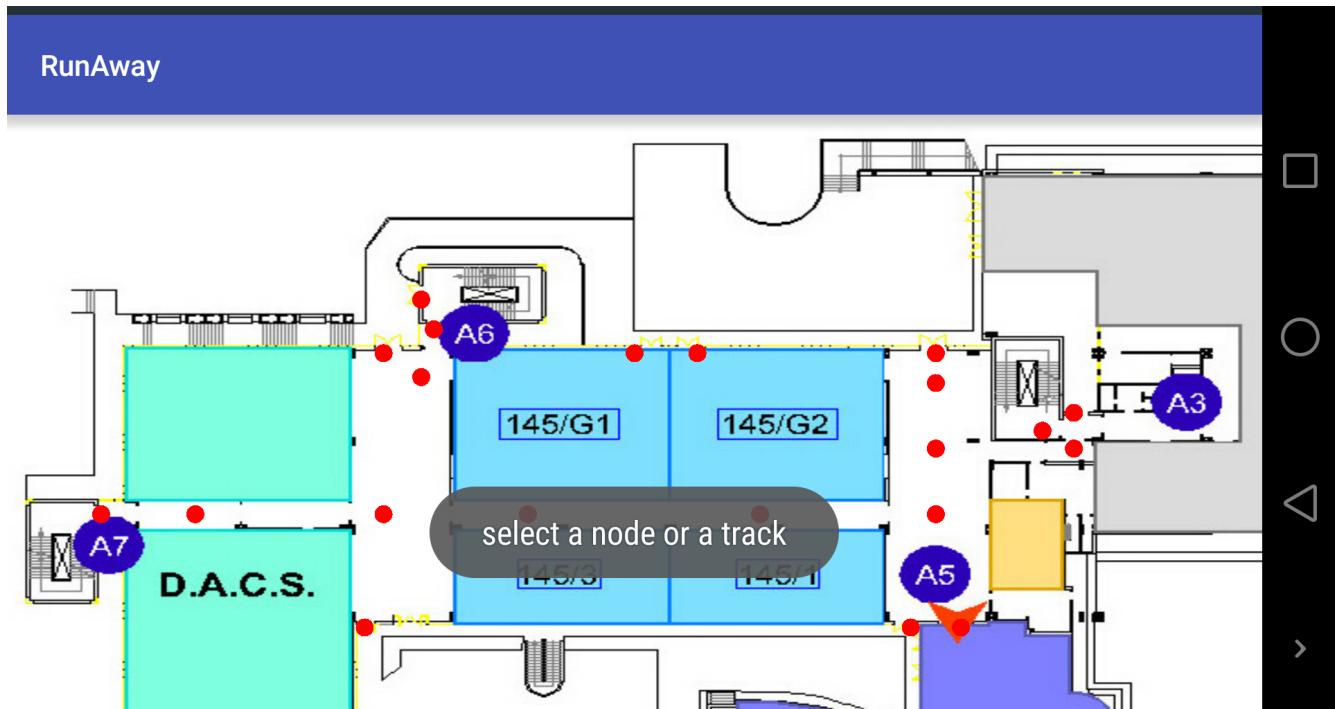


Selezionando ad esempio il piano 145, viene visualizzata solo la mappa relativa a tale piano, con i nodi in rosso collocati nella giusta posizione.

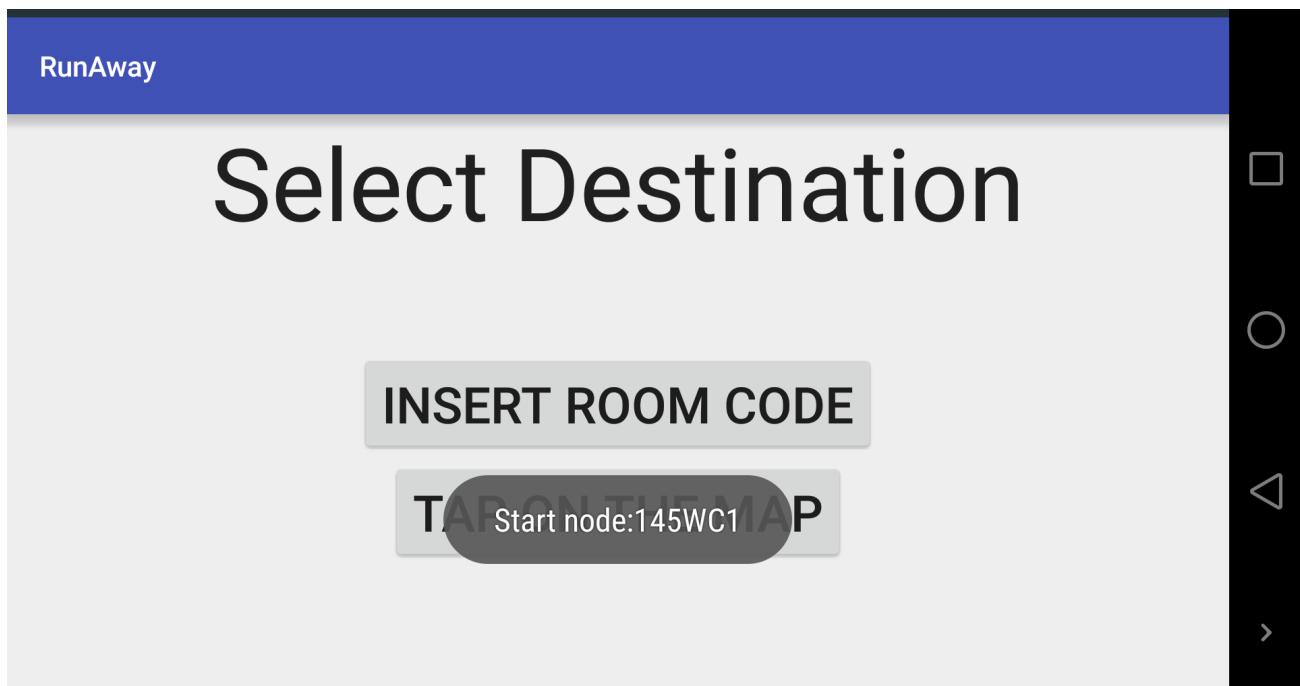


A questo punto si può direttamente scegliere il nodo facendo un tap su di esso, ma può anche capitare che si tocchi nello spazio non contenente nodi. L'app allora deve gestire correttamente entrambe le situazioni.

Tap errato



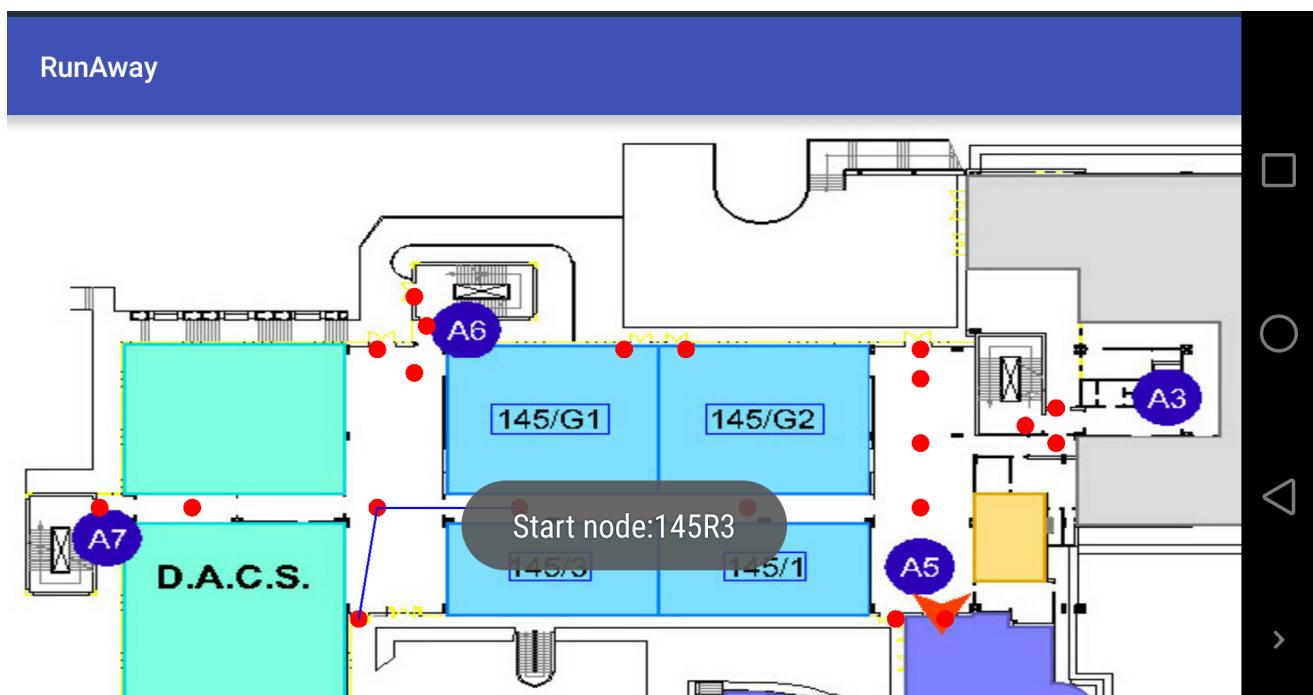
Tap sul nodo 145WC1

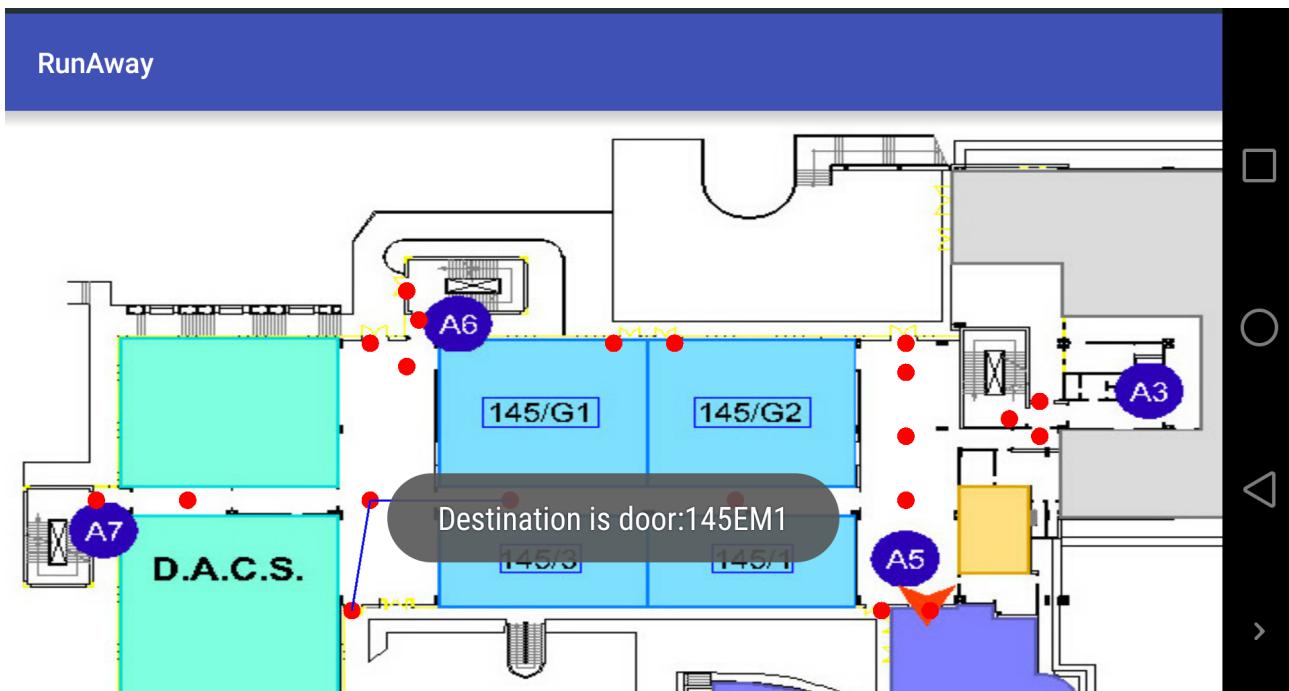


Task ID: 2.6 Verifica corretto aggiornamento della posizione sul Server

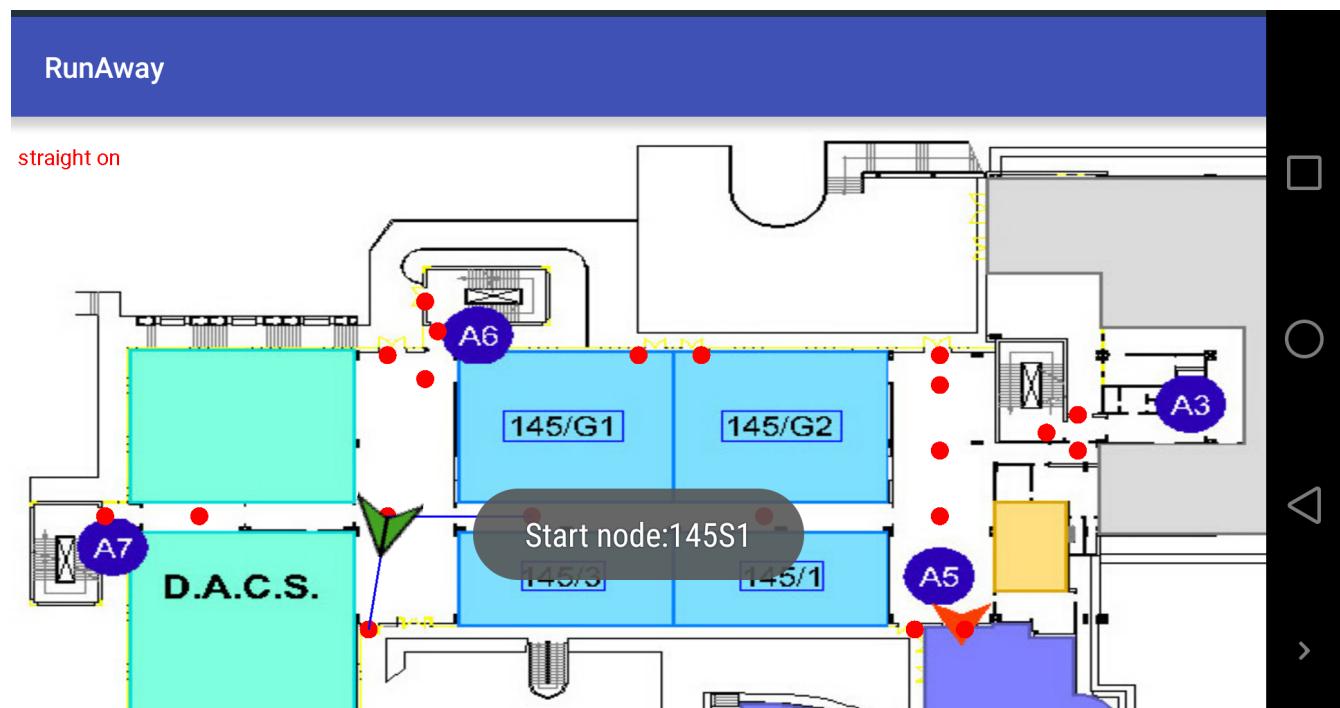
Non modellando un server vero e proprio, è stato creato un Server locale all'interno dell'app, accessibile all'indirizzo localhost:8080 una volta aperta l'app. Questo server ha il compito di creare un log delle posizioni che l'utente decide di comunicare, tappando sulla mappa per aggiornare la propria posizione, e della posizione iniziale.

Partendo quindi dalla situazione in cui si sia già in navigazione di emergenza da 145R3 a 145EM1, si supponga di aggiornare la propria posizione impostandola sul nodo 145S1.

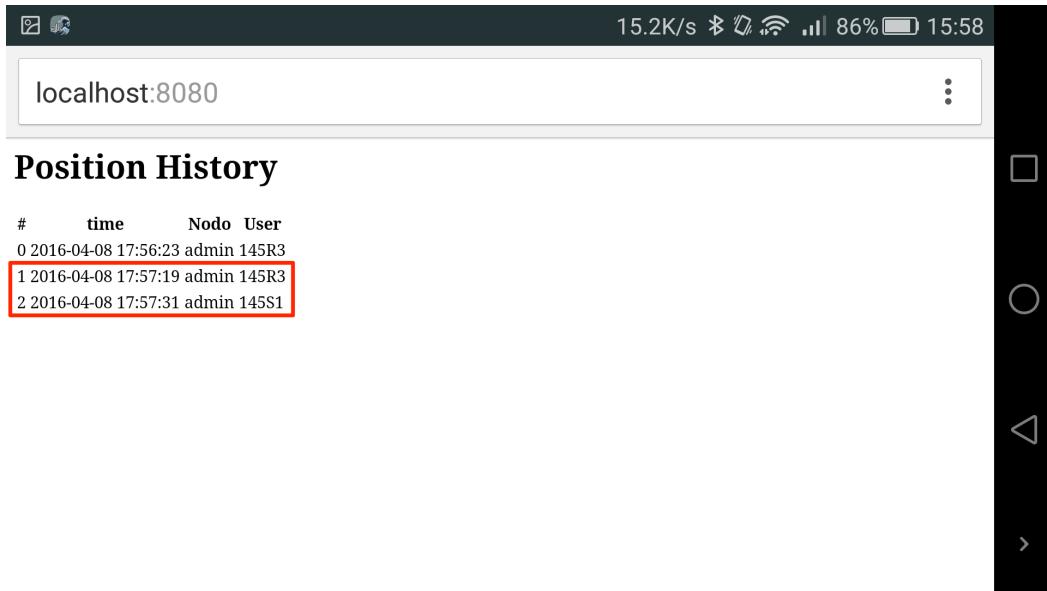




Una volta fatto il tap su 145S1, il nodo di partenza viene aggiornato, e l'indicatore dell'utente viene posizionato sopra di esso.



Ora, per vedere cosa abbia ricevuto il Server, apriamo il browser e digitiamo localhost:8080.

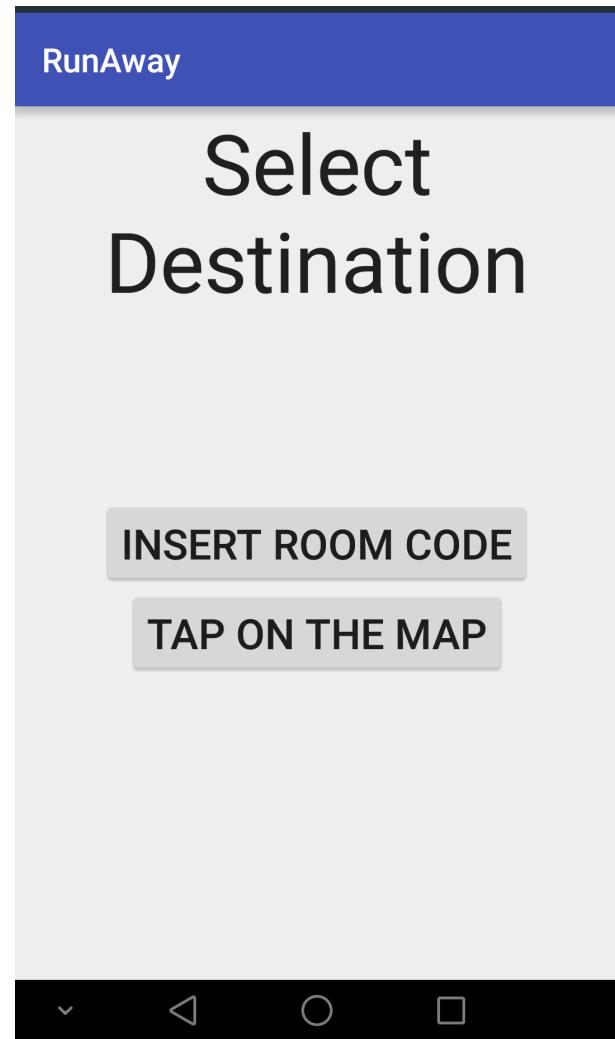


La posizione aggiornata e la posizione iniziale sono state acquisite correttamente. Il primo log è dovuto ad un tap accidentale sullo stesso nodo di partenza.

7.3 Funzionamento in condizioni normali

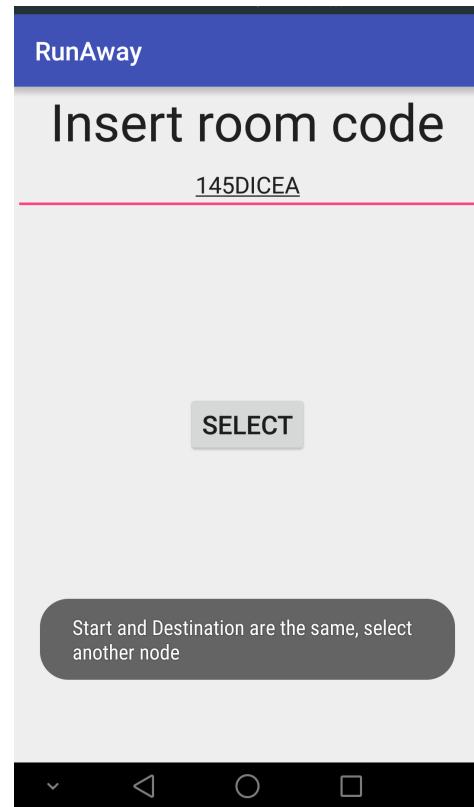
Task ID: 3.5 Controllo sulla destinazione

Una volta selezionata la partenza, e in condizioni non di emergenza, si dà la possibilità all'utente di poter scegliere la propria destinazione attraverso due modalità già viste: inserimento del codice della stanza, ed il tap sulla mappa. In questo scenario, oltre al controllo sulla correttezza del codice inserito e del tap sulla mappa, bisogna anche notificare se la destinazione che si sta selezionando coincide con la partenza.

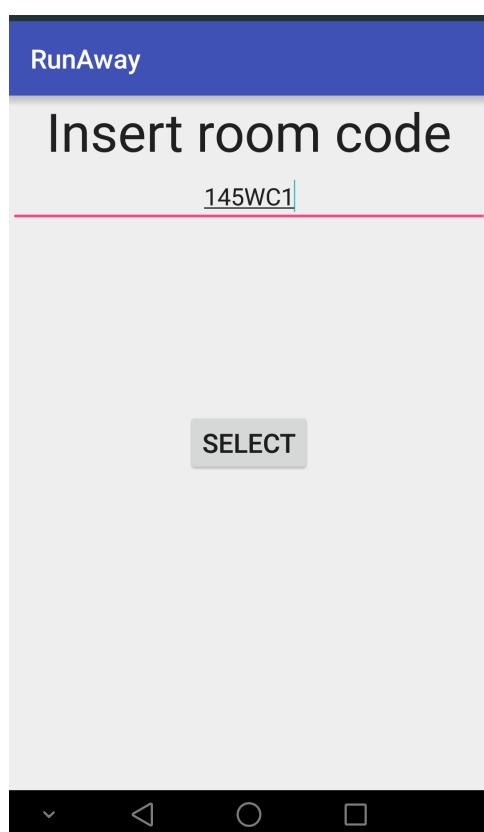


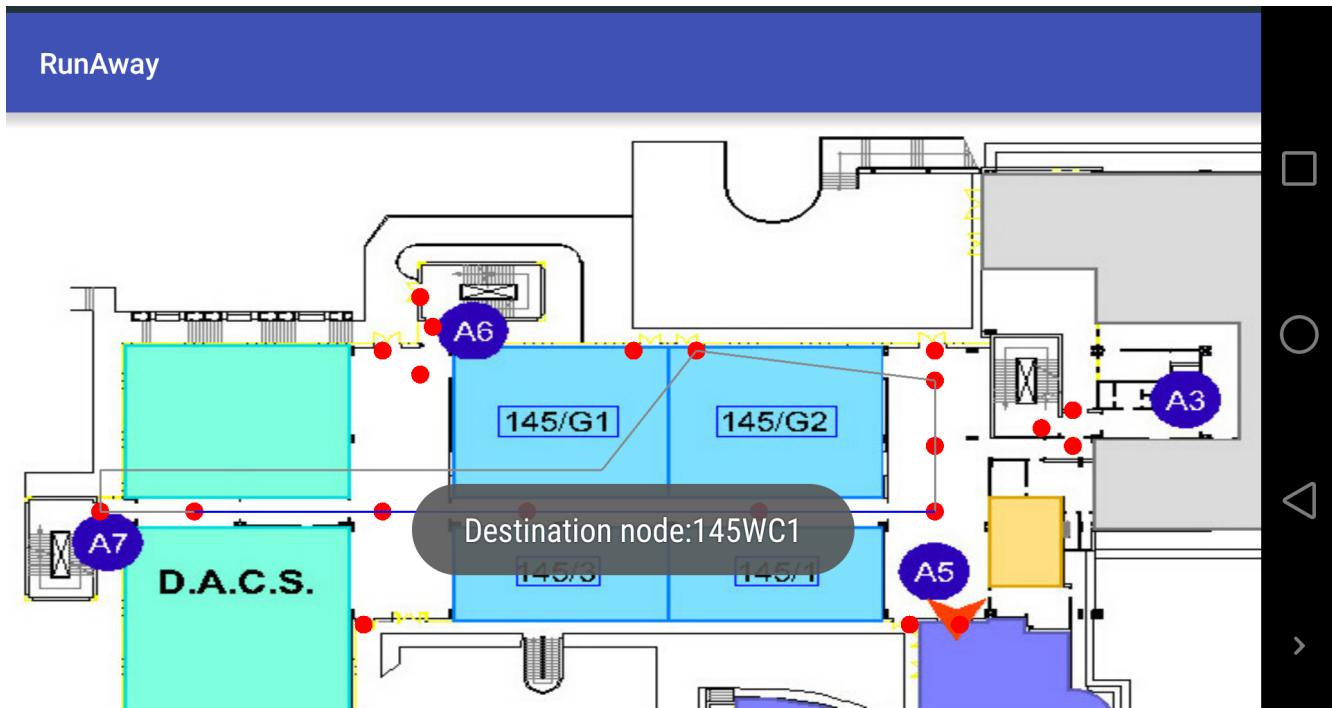
In questo caso l'app dovrebbe solo notificare l'errore ed invitare l'utente a specificare un'altra destinazione, se non si trova già in quel punto.

- Inserimento codice stanza, supponendo sia stata impostato come partenza il nodo 145DICEA:



Selezionando invece il nodo 145WC1

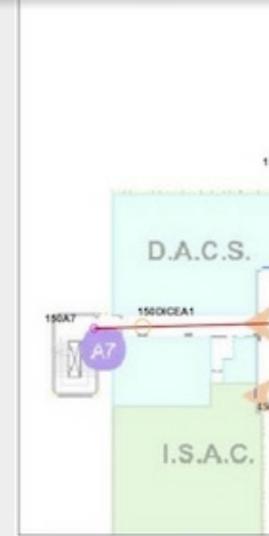




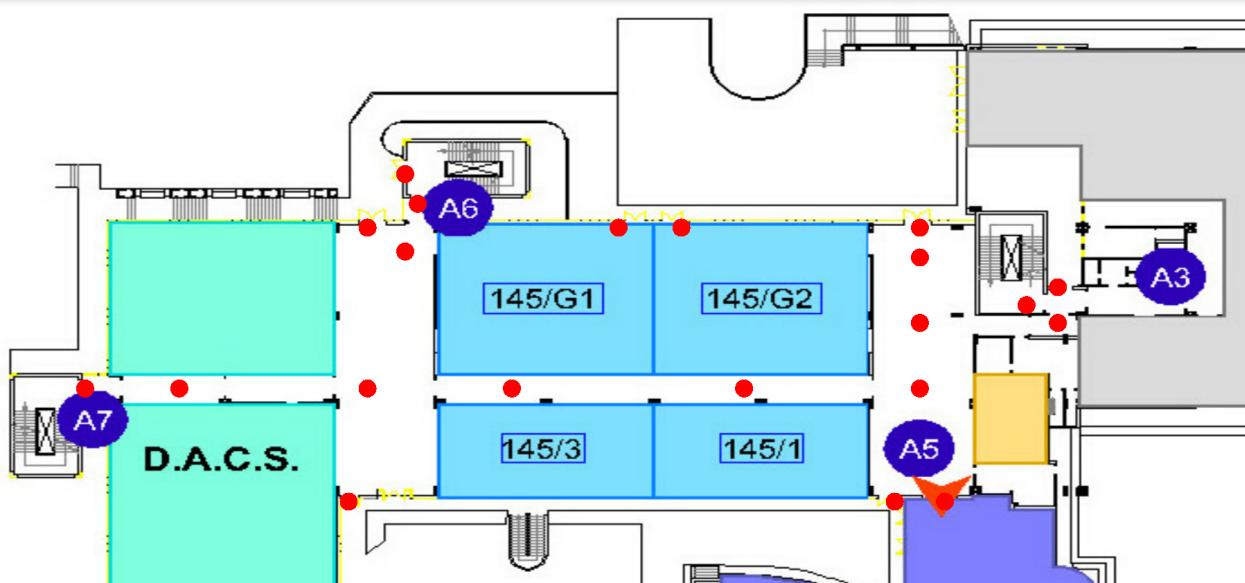
La destinazione viene impostata correttamente e i percorsi tracciati sulla mappa, in blu quello con i pesi minori (in questo caso di non emergenza la lunghezza degli archi), in grigio il percorso alternativo.

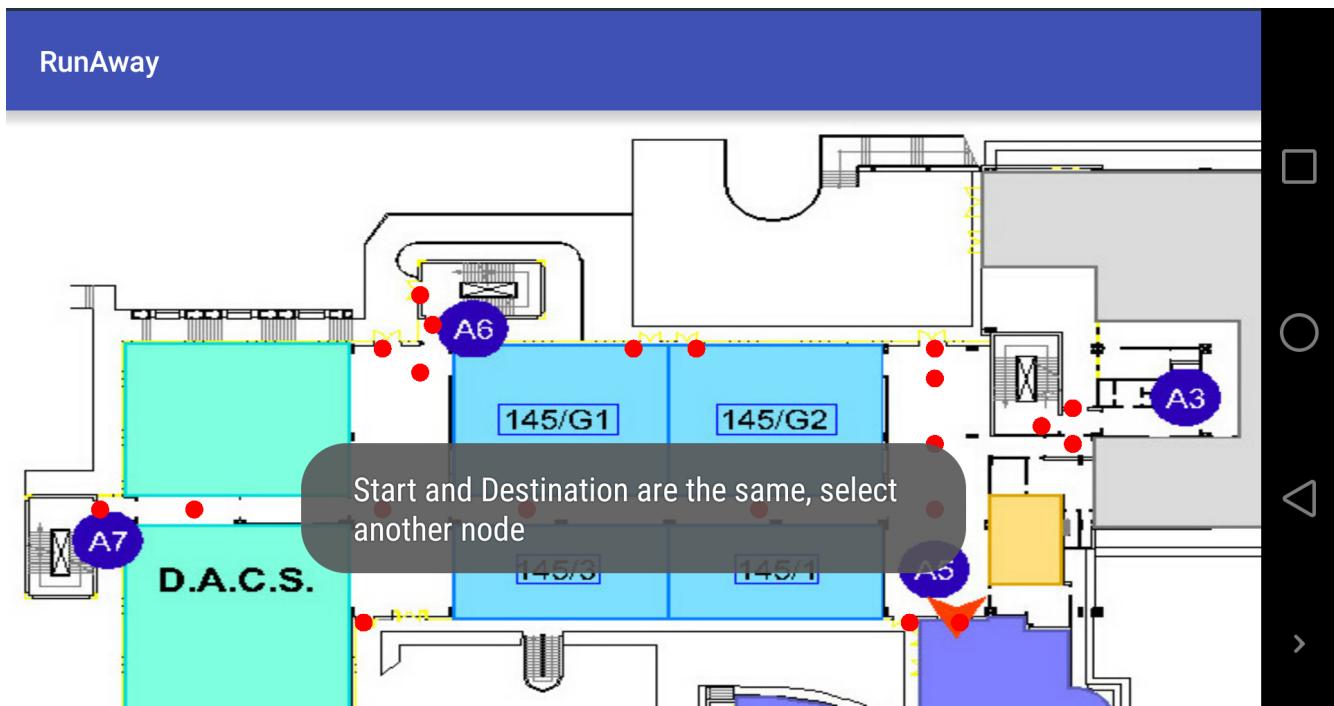
- Tap sulla mappa: anche in questo caso l'app deve evitare che si selezioni un nodo già selezionato come nodo di partenza, e comunicarlo all'utente. Deve inoltre impostare correttamente la destinazione e visualizzare i percorsi calcolati. Avendo impostato ad esempio il nodo 145WC1 come partenza, si suppone di fare un tap sullo stesso nodo per quanto riguarda la destinazione, dopo aver selezionato il piano 145 dal carosello.

RunAway

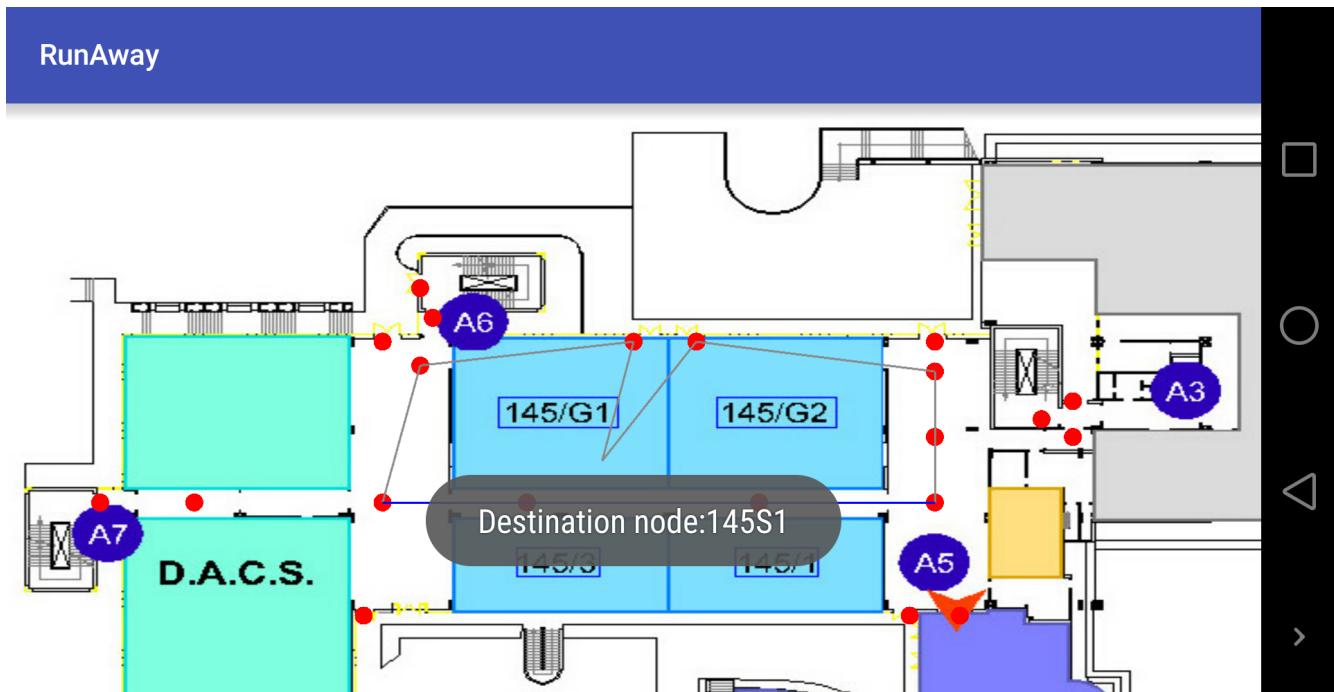


RunAway





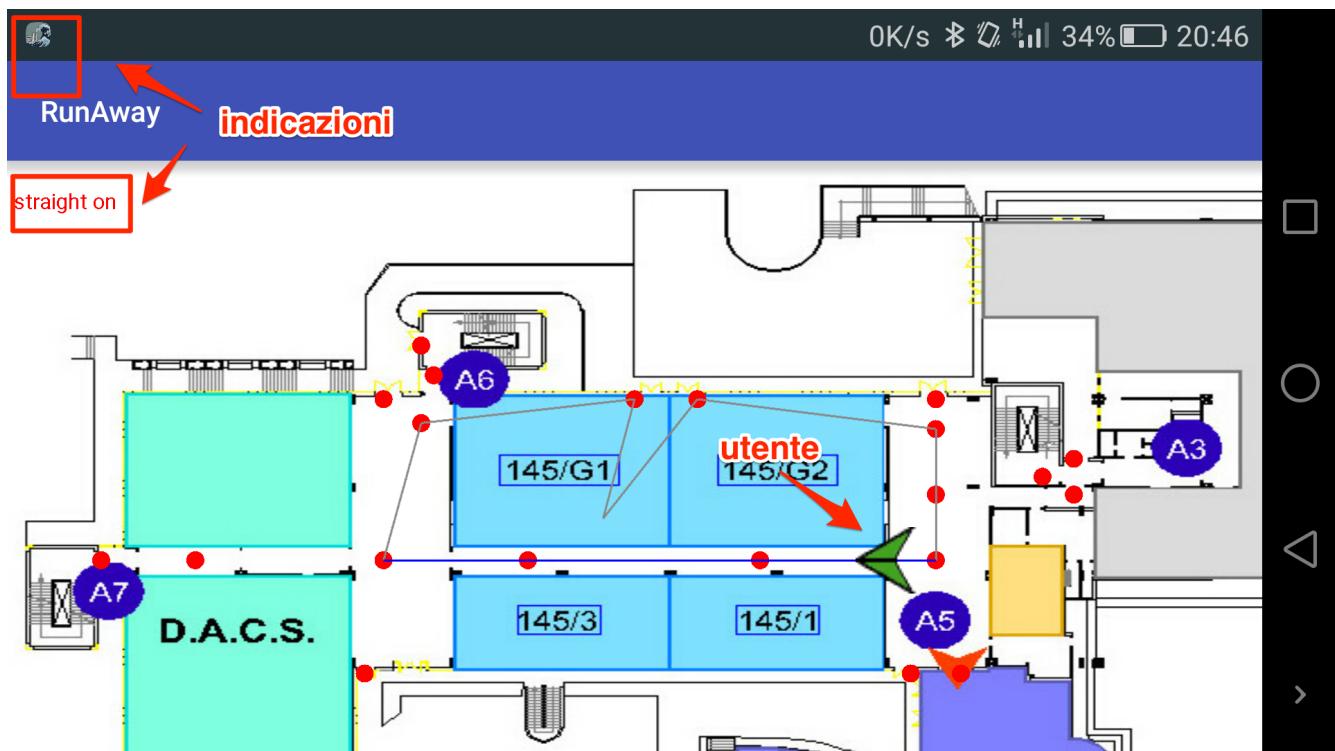
Selezionando un nodo diverso come destinazione, ad esempio il nodo 145S1, si ottengono i percorsi calcolati.



Task ID: 3.6 Controllo avvio navigazione e indicazioni

Nel momento in cui l'utente abbia a disposizione sulla mappa i percorsi calcolati, può fare un tap sull'uno e sull'altro percorso per iniziale la navigazione. Questa viene assistita attraverso delle notifiche vibrazionali dello smartphone e, se l'utente ne è impossesso, vengono replicate su un dispositivo indossabile dotato di Android Wear, precedentemente configurato per il mirroring delle notifiche dallo smartphone.

Partendo quindi dall'interfaccia che include i due percorsi calcolati da 145WC1 a 145S1, si testa il corretto avvio della navigazione verificando la comparsa dell'indicatore utente sul nodo di partenza, con l'orientamento giusto, e la presenza delle indicazioni a schermo, e di quelle sull'indossabile. Si suppone di scegliere il percorso più breve tra i due. A questo punto la freccia si sposterà solo se l'utente si fosse girato nella direzione giusta, in modo da seguire esattamente le indicazioni a schermo. Se non lo fosse, le indicazioni gli suggeriranno come girarsi per rimettersi nella giusta direzione.



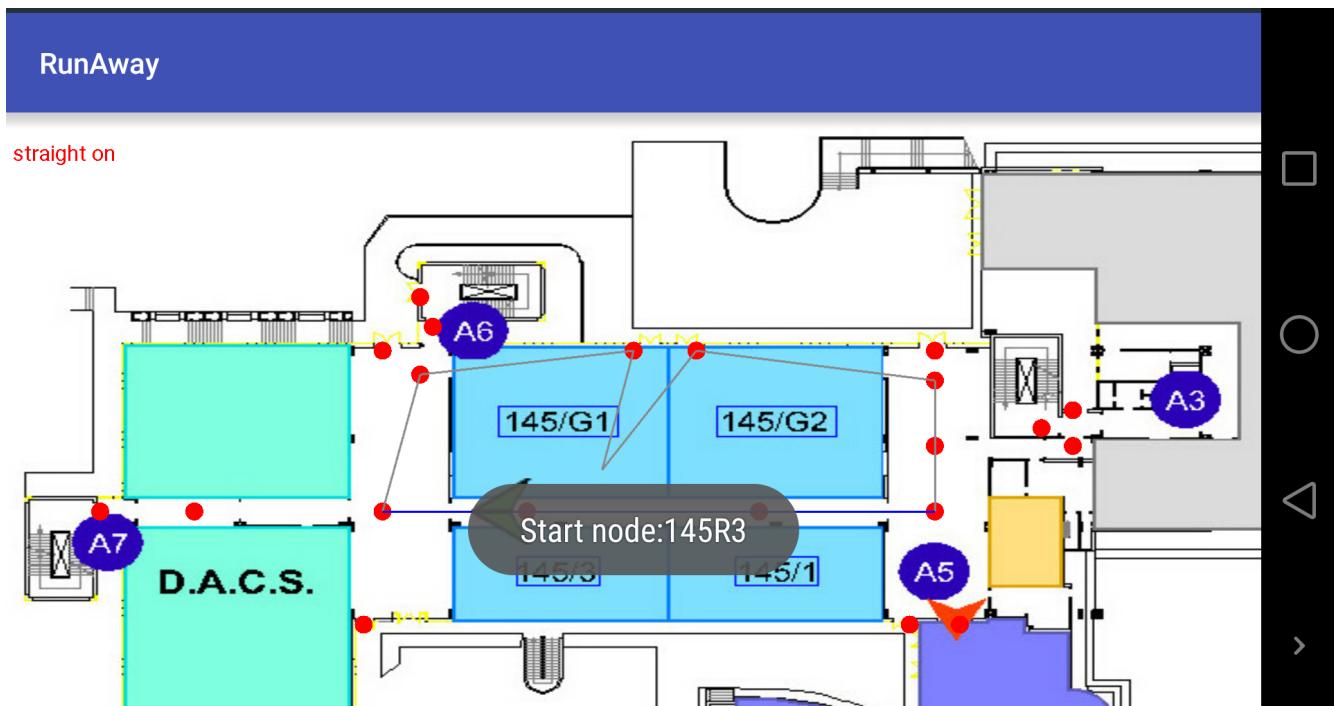
E sull'indossabile, si ottiene la stessa indicazione.



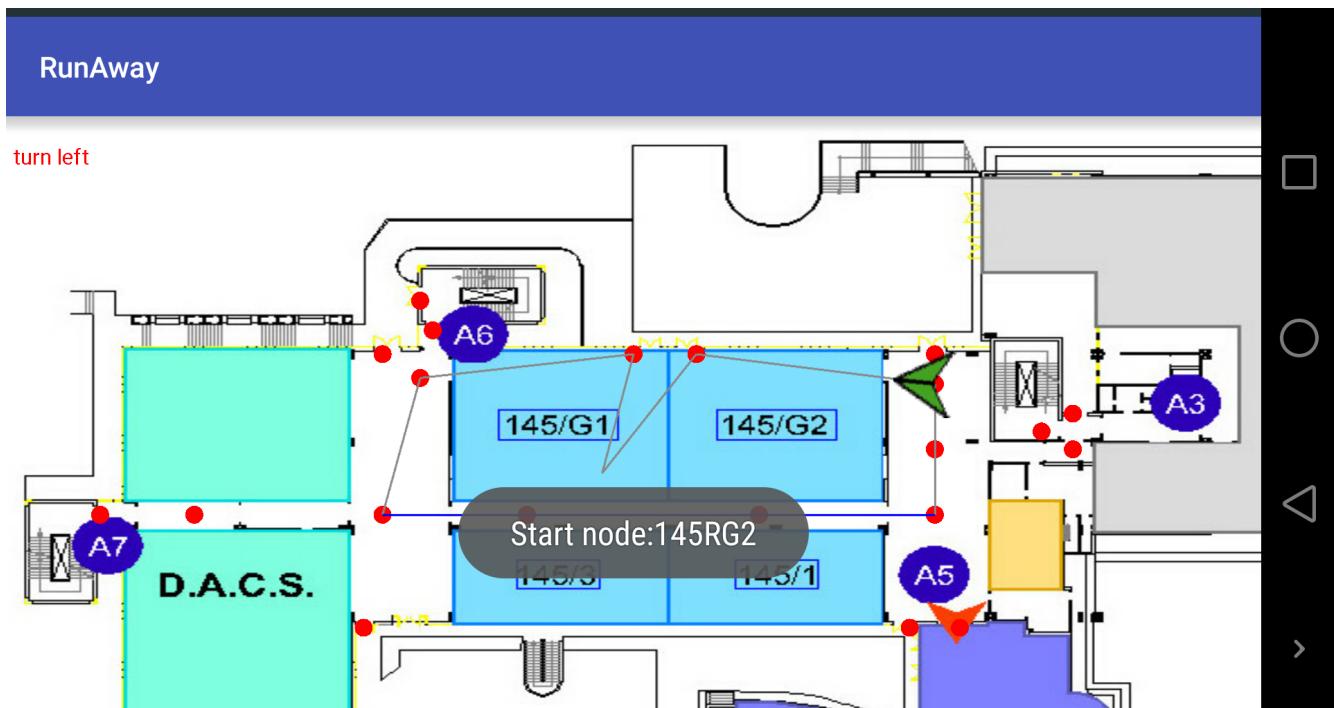
Task ID: 3.7 Controllo sui punti intermedi

Durante la navigazione, l'utente può aggiornare sulla mappa la propria posizione, se si rende conto di essere in un punto di controllo diverso da quello visualizzato sull'app, o se ritiene di voler passare per un nodo diverso da quello stabilito lungo il percorso, sia esso incluso nel percorso alternativo, sia al di fuori (ed in questo caso verrà chiesto se si vuole partire con una nuova navigazione).

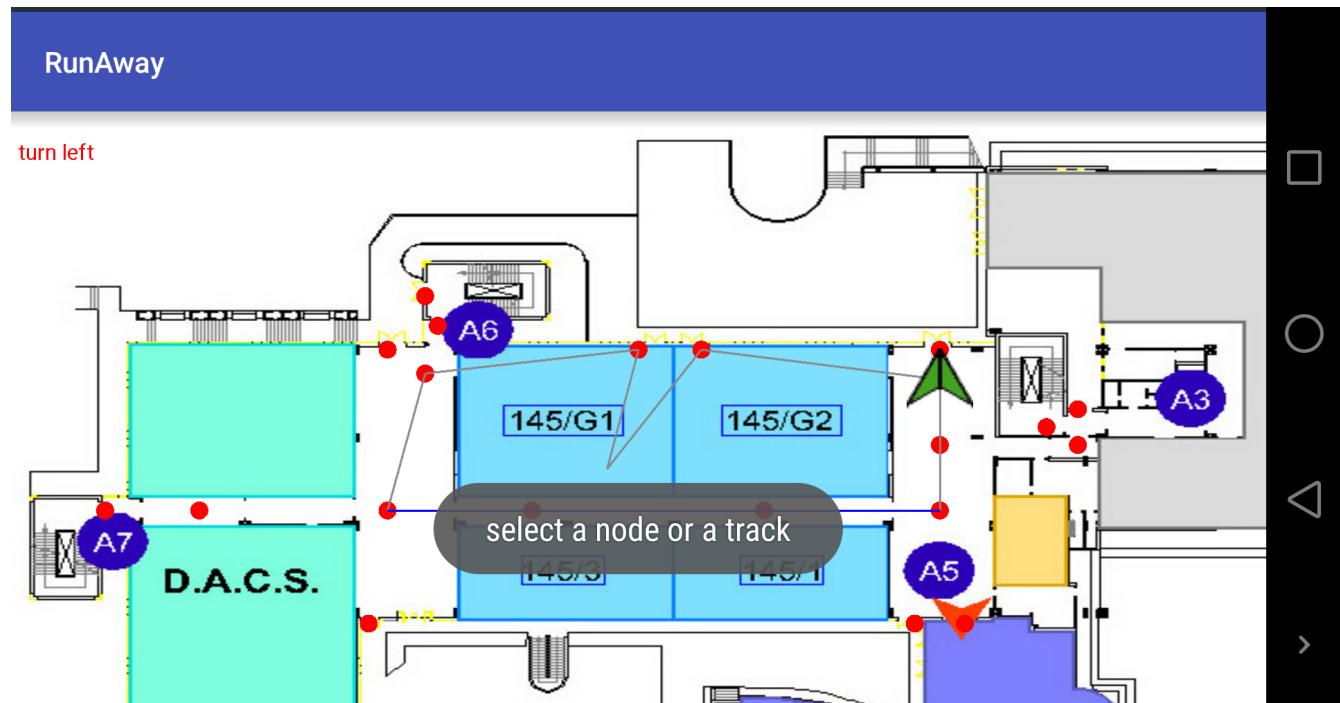
Supponendo di utilizzare la navigazione del Test precedente, e volendo selezionare il nodo 145R3 come nuova posizione utente, si verifica il corretto posizionamento dell'indicatore, sempre sullo stesso percorso perché il nodo vi è incluso.



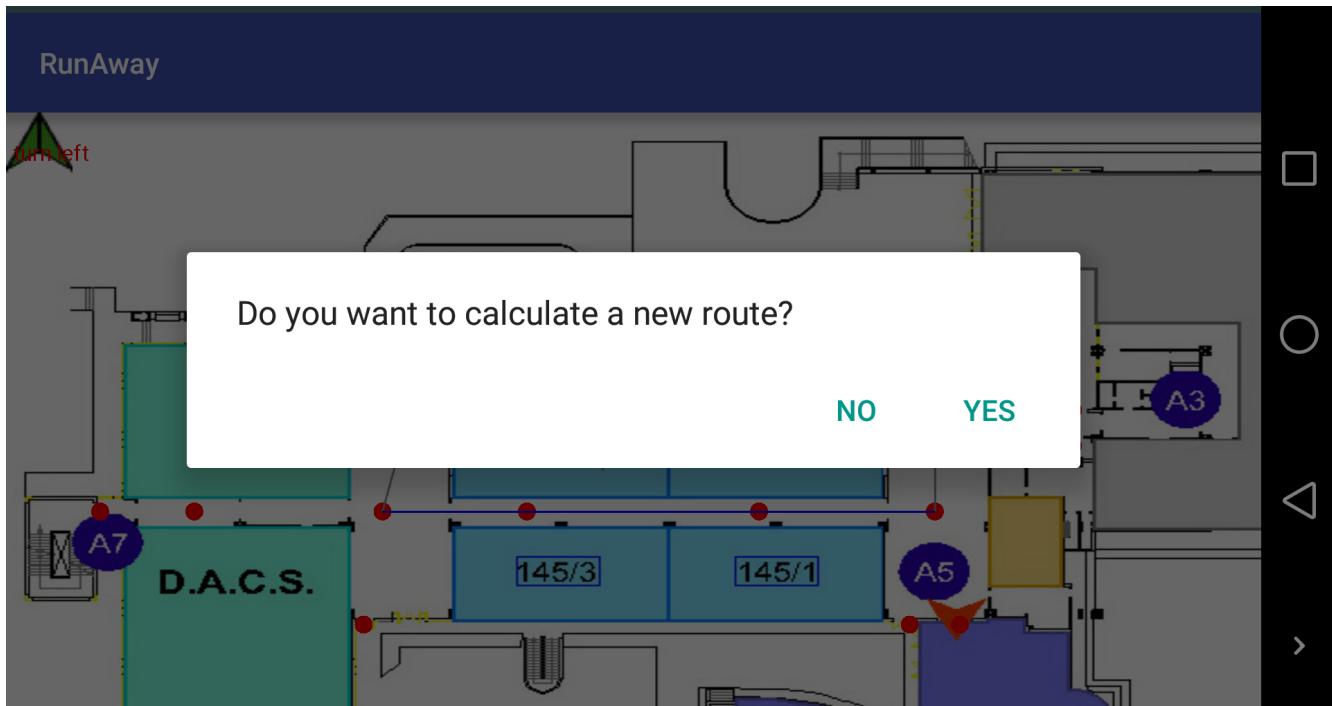
Se invece si vuole cambiare percorso, ed in particolare si vuole stabilire come posizione dell'utente il nodo 145RG2, incluso nel percorso alternativo, si può notare come si aggiornino correttamente anche le indicazioni:



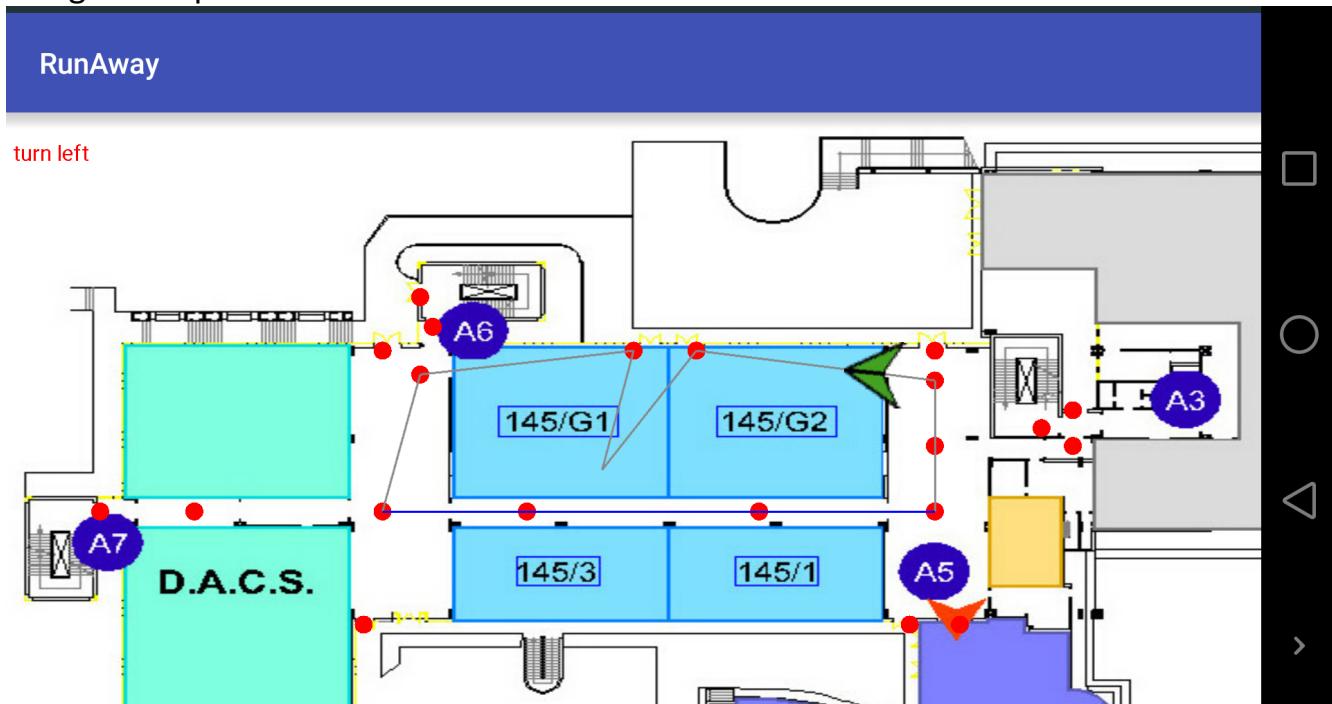
Nel caso in cui, durante la navigazione si selezioni un'area della mappa che non include nè percorsi, nè nodi, allora viene notificato all'utente di selezionare uno di essi e la navigazione non viene influenzata in alcun modo.



Se il punto selezionato è un punto di controllo ma non appartiene a nessuno dei due percorsi consigliati, una finestra di dialogo chiederà all'utente se ricalcolare un nuovo percorso, partendo dal nuovo punto di controllo, o se riprendere la navigazione.



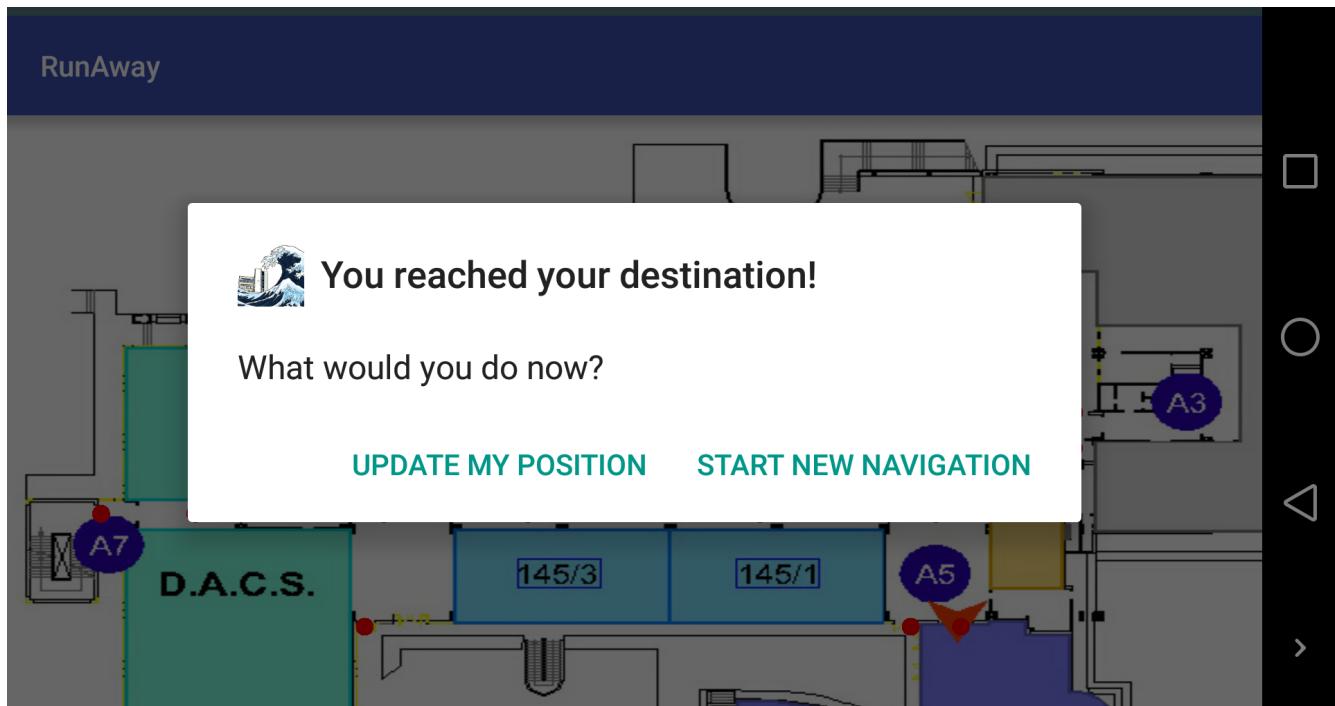
Nel caso in cui l'utente scelga di premere NO sulla finestra di dialogo, verrà ripresa la navigazione precedente.



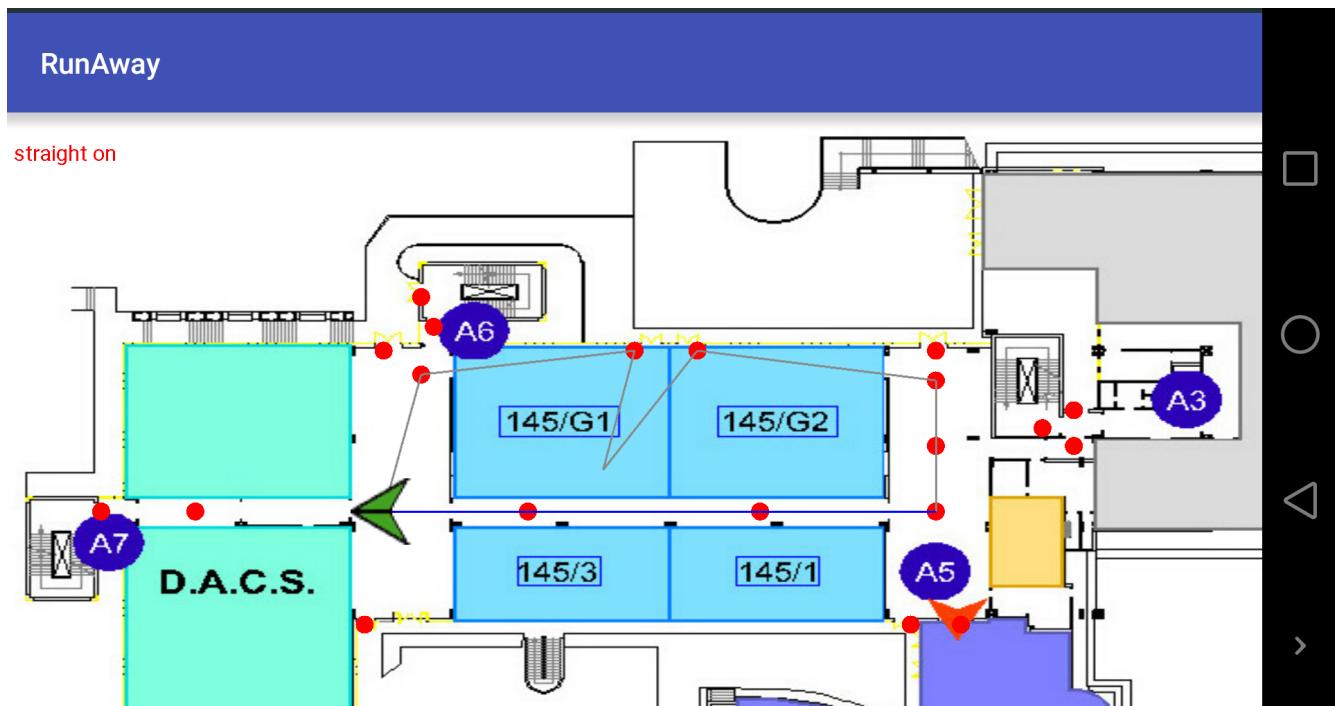
Se invece l'utente sceglie di calcolare un nuovo percorso, allora verrà riportato nella pagina per la scelta della destinazione, mantenendo l'informazione sulla modalità di navigazione precedente, e con il nodo iniziale impostato sul nodo scelto. Le varie modalità di navigazione verranno testate nei successivi test.

Task ID: 3.8 Controllo sull'arrivo a destinazione

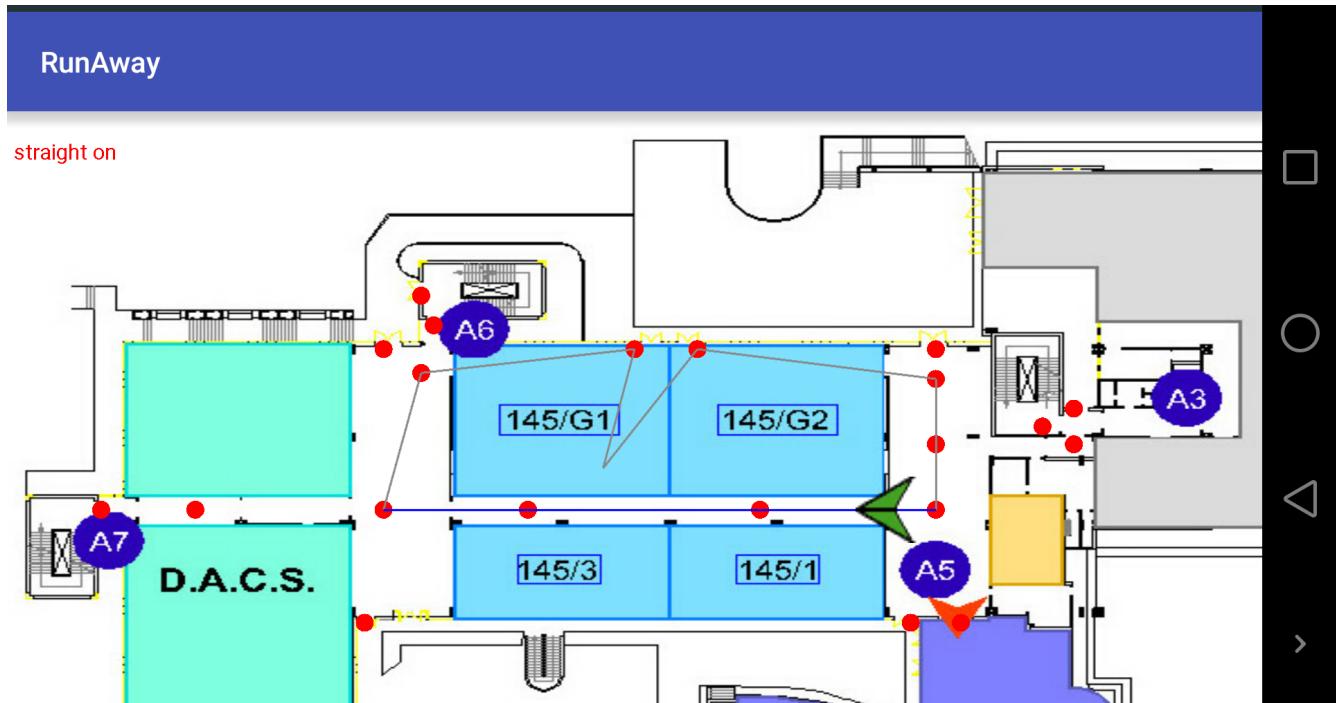
Una volta arrivato sul punto di destinazione, all'utente viene notificata la possibilità di aggiornare la propria posizione (nel caso abbia ancora bisogno di spostarsi), o di tornare alla homepage per una nuova navigazione.



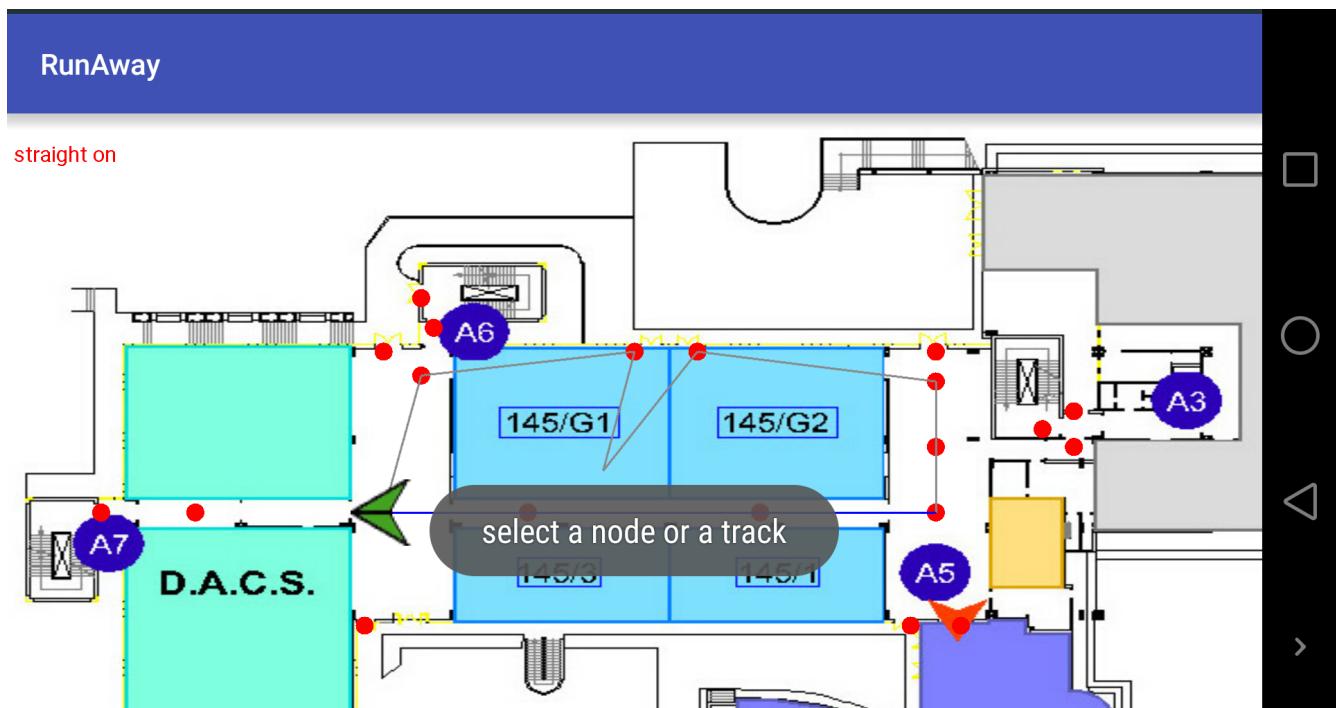
Se l'utente sceglie di aggiornare la propria posizione, verrà visualizzata la mappa corrente.



Dopodiché, se l'utente sceglie un percorso già tracciato, la navigazione riprenderà dall'inizio.

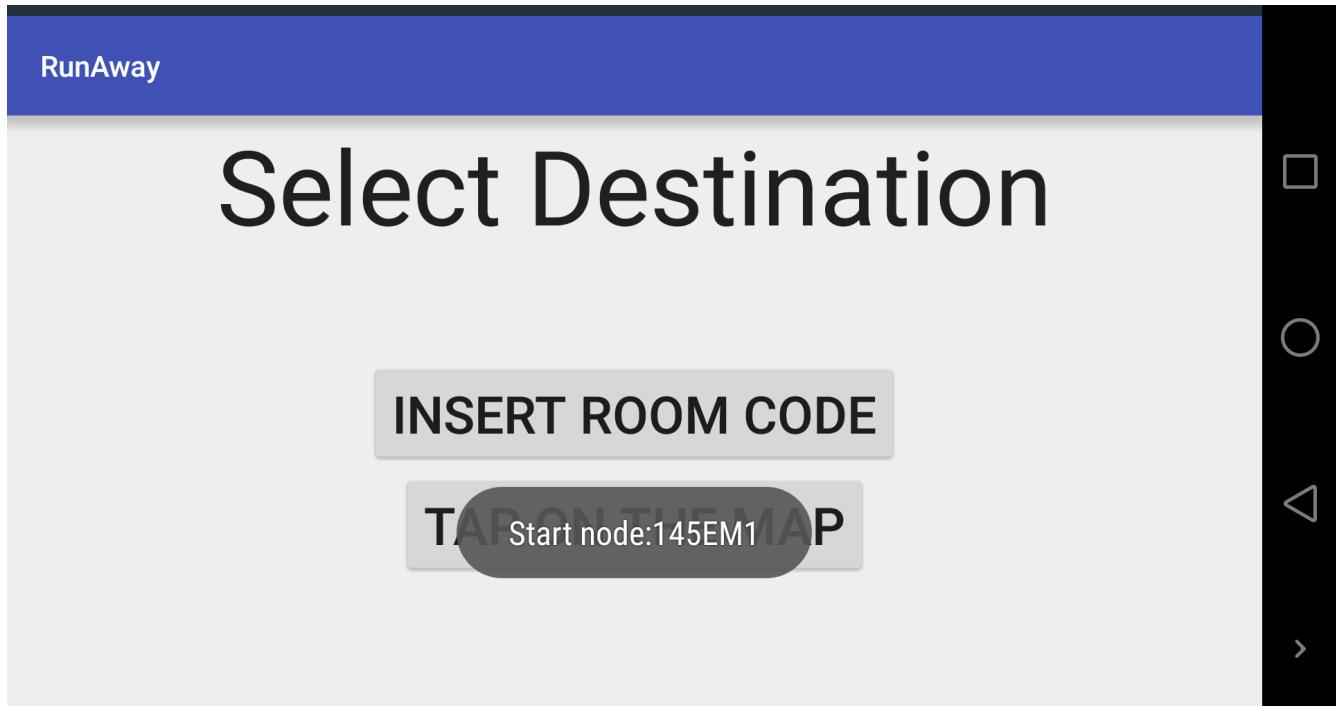


Se, l'utente seleziona un'area vuota riceve un invito a selezionare un nodo o un tracciato.

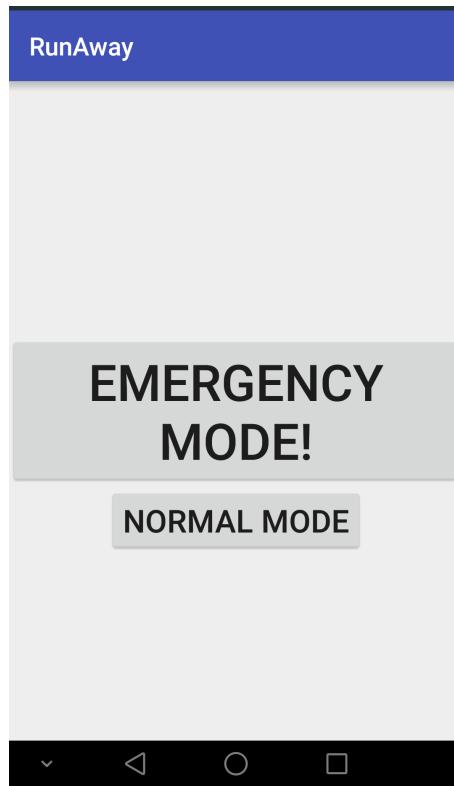


Se invece decide di scegliere un nuovo nodo, questo verrà preso come nodo di partenza per una nuova navigazione e si verrà riportati alla schermata per la selezione della destinazione.

Supponendo di selezionare il nodo 145EM1 come nuova posizione di partenza, si ottiene la schermata seguente.

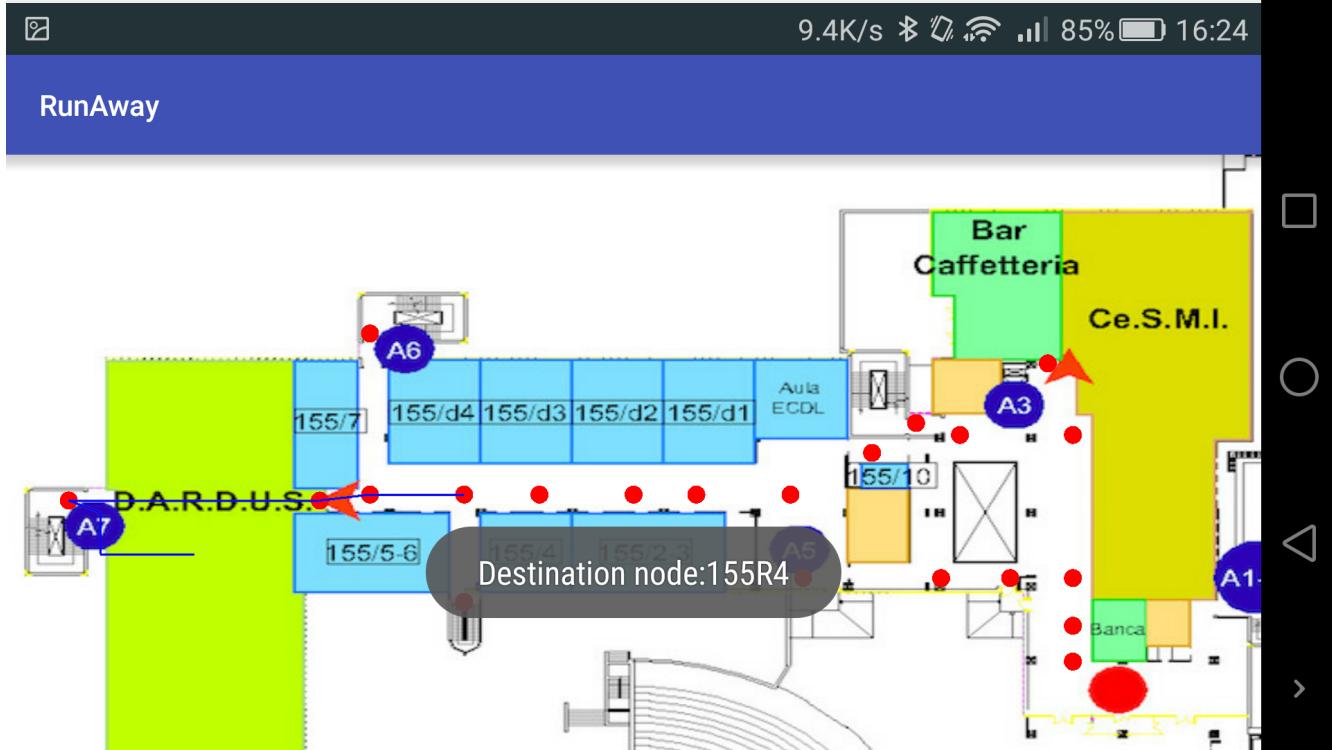
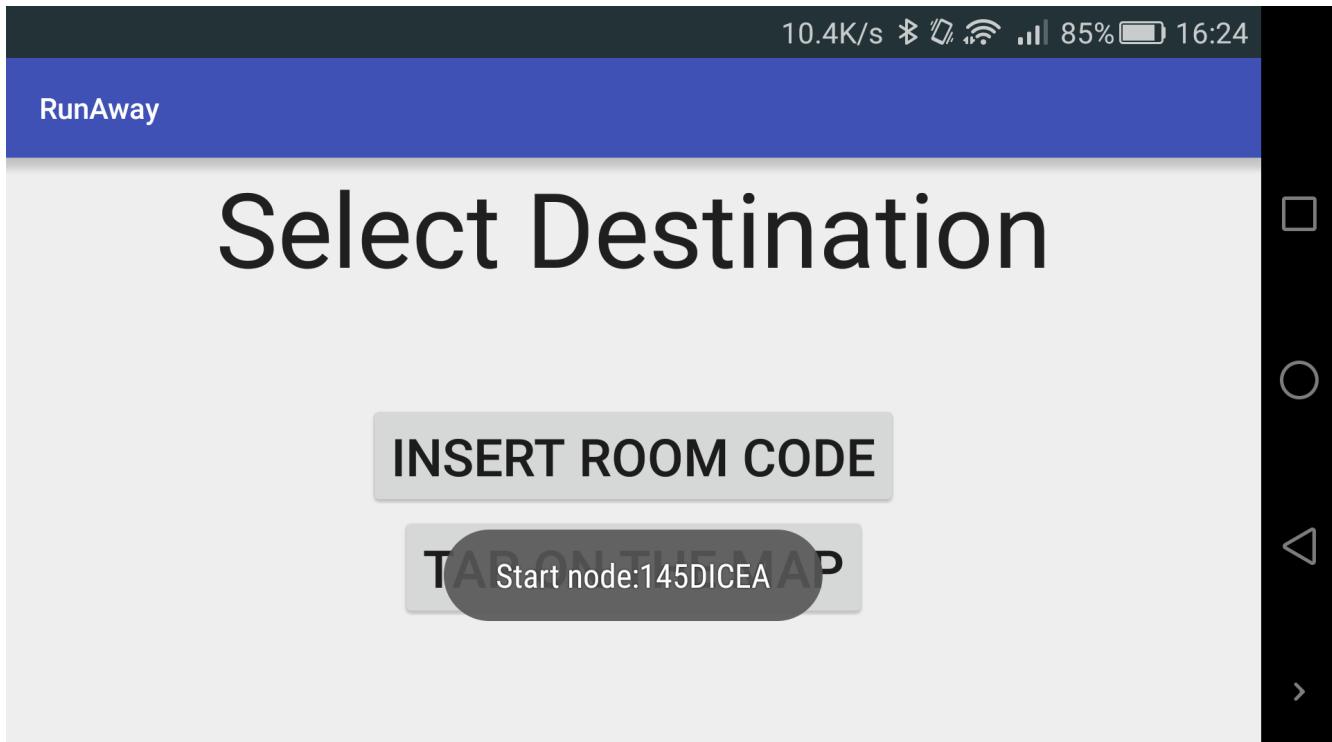


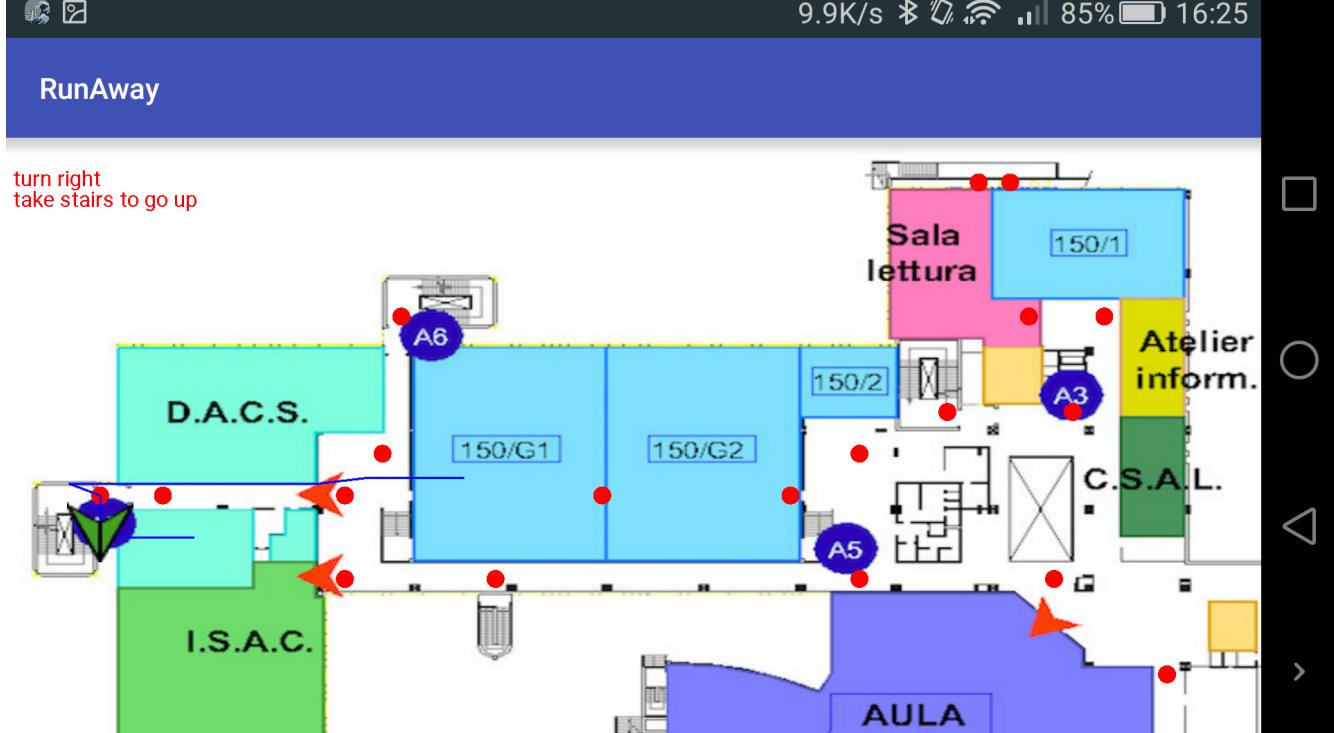
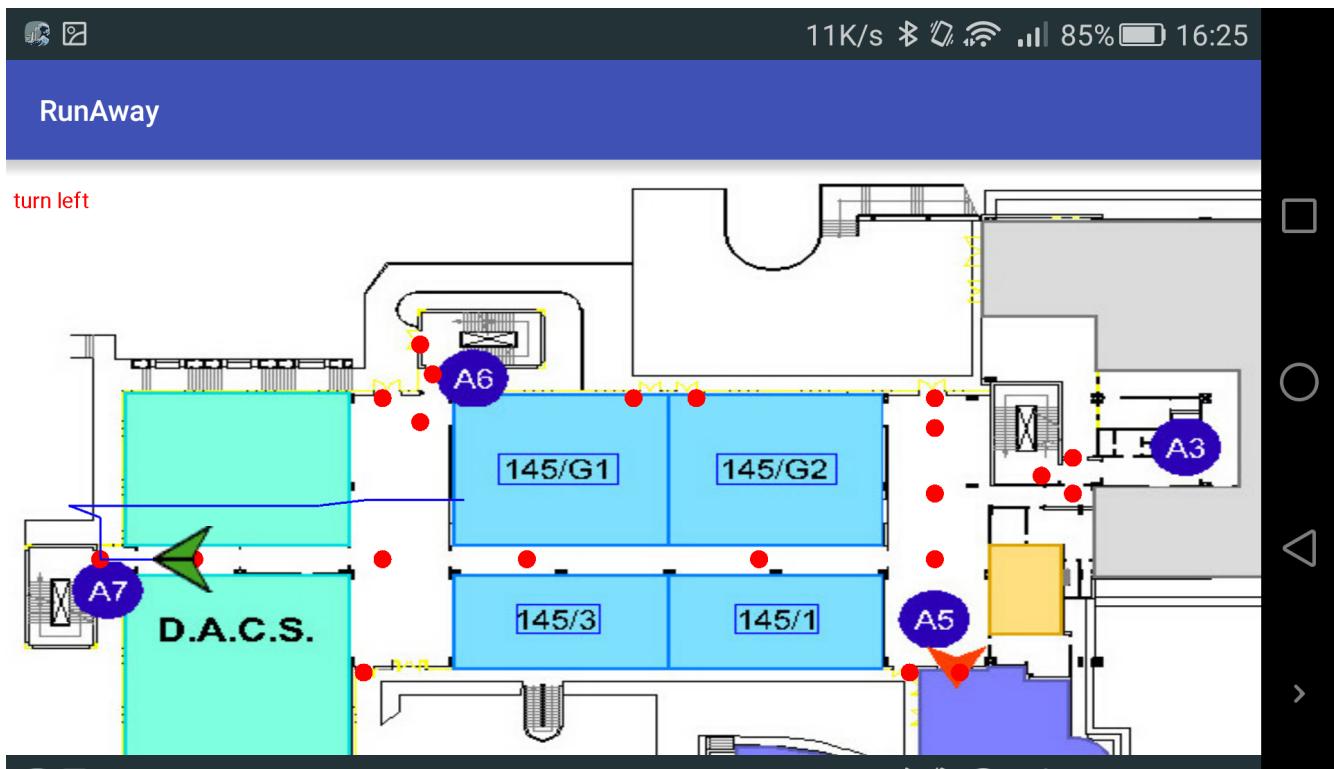
Se invece l'utente, una volta arrivato a destinazione, sceglie di iniziare una nuova navigazione, verrà riportato sulla schermata per selezionare la modalità della stessa.

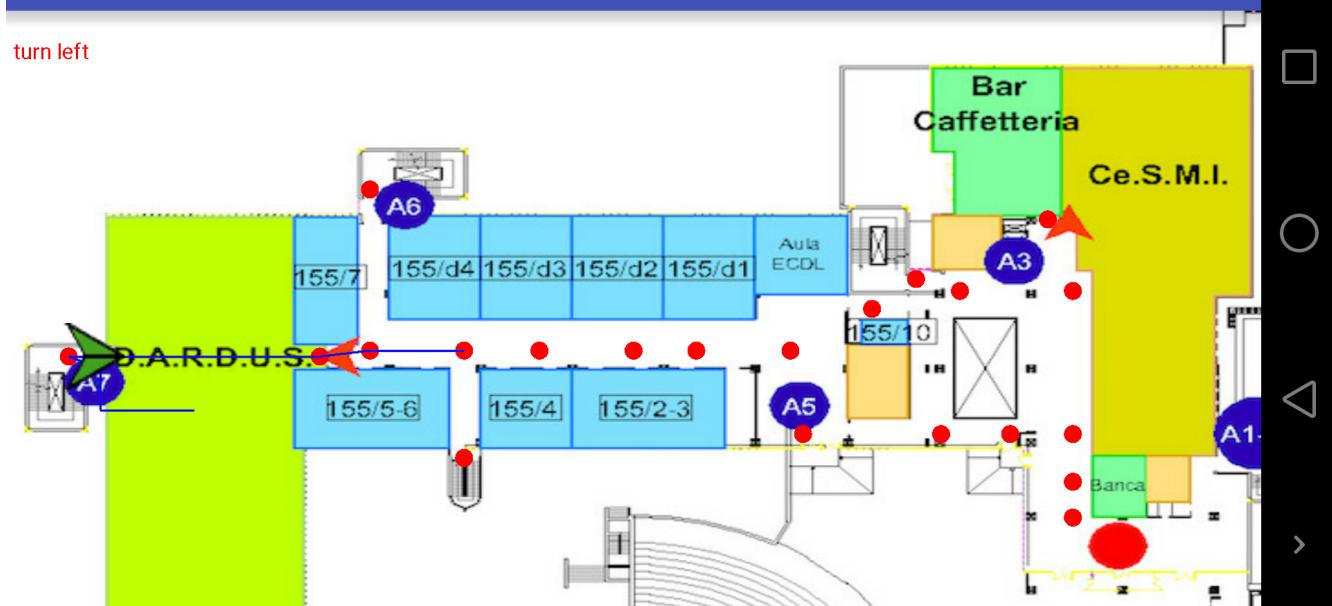
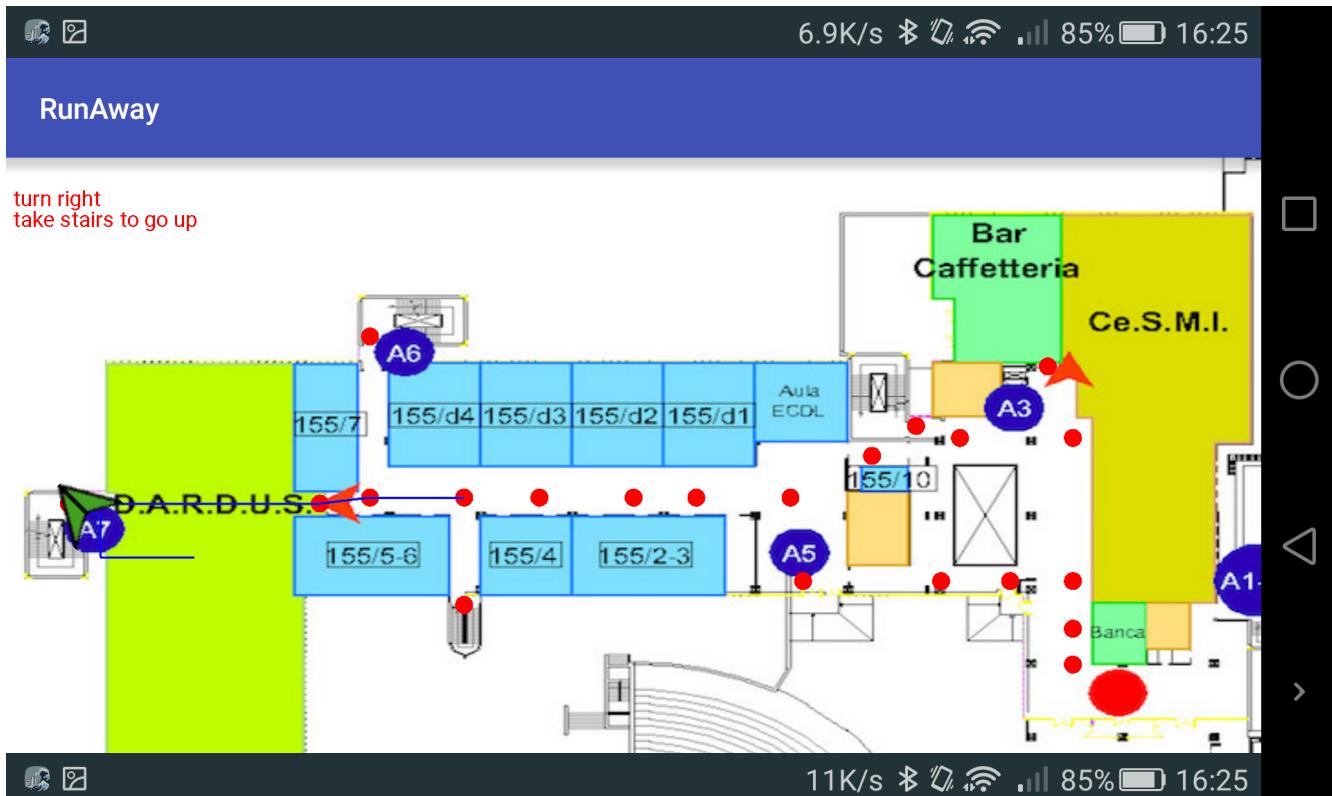


Task ID: 3.9 Controllo sullo switch delle mappe tra i piani, in navigazione, e delle indicazioni relative

Per testare la corretta navigazione tra i piani, si supponga di voler partire da 145DICEA e raggiungere il nodo 155R4. Si dovranno quindi risalire due piani, e la mappa dovrebbe cambiare due volte. Si illustrano quindi in seguito le schermate che si susseguono.





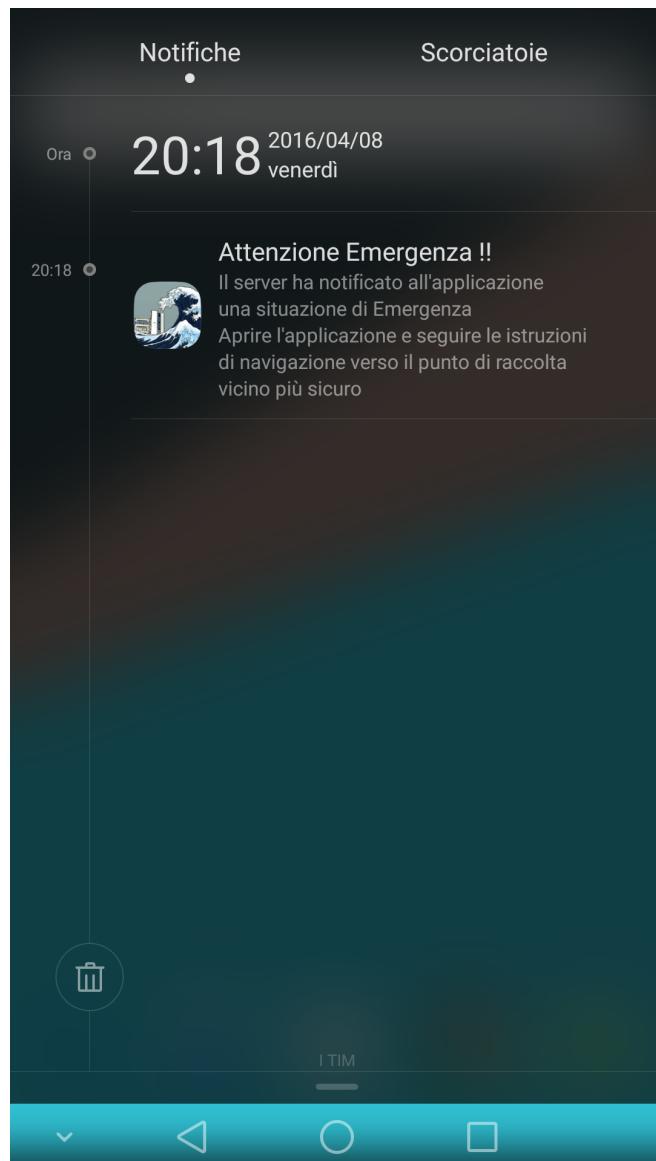


Come si può notare, l'applicazione funziona bene e le indicazioni sulle scale sono fornite in maniera corretta.

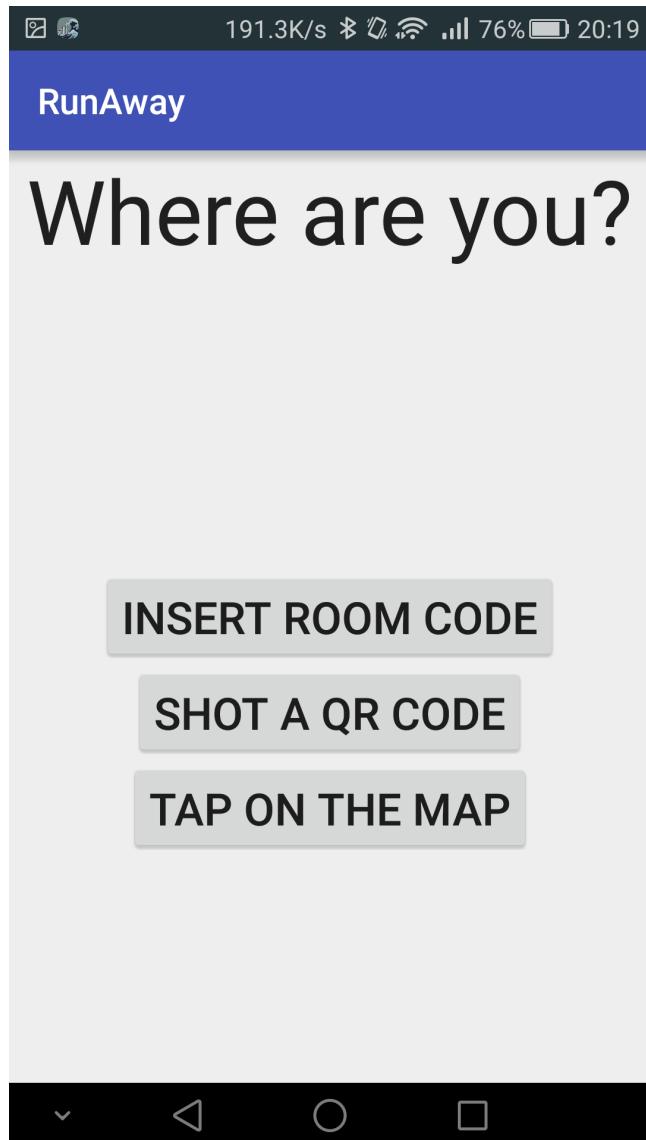
7.4 Notifica di emergenza

Task ID: 4.5 Funzionamento della notifica di emergenza

L'utente deve poter essere avvisato nel momento in cui vi è un'emergenza all'interno dell'edificio, se il server è disponibile per poterlo fare, ed iniziare la navigazione automaticamente in emergenza alla selezione della suddetta notifica. Per il nostro test, non modellando il server, si è programmata una notifica temporizzata tramite codice, che simulerà un avviso di situazione di emergenza all'interno dell'edificio.



E quindi selezionando la notifica:



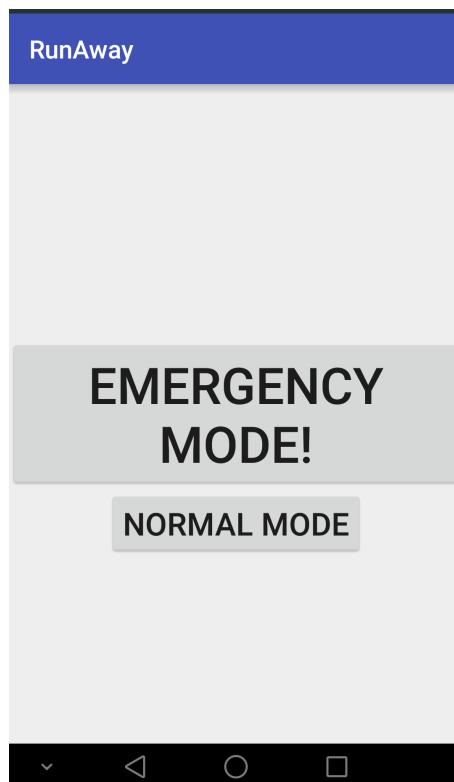
viene automaticamente impostata la modalità di emergenza e si chiede all'utente di stabilire la propria posizione, per guidarlo verso il punto di raccolta più vicino.

7.5 Guida al punto di raccolta in modalità online

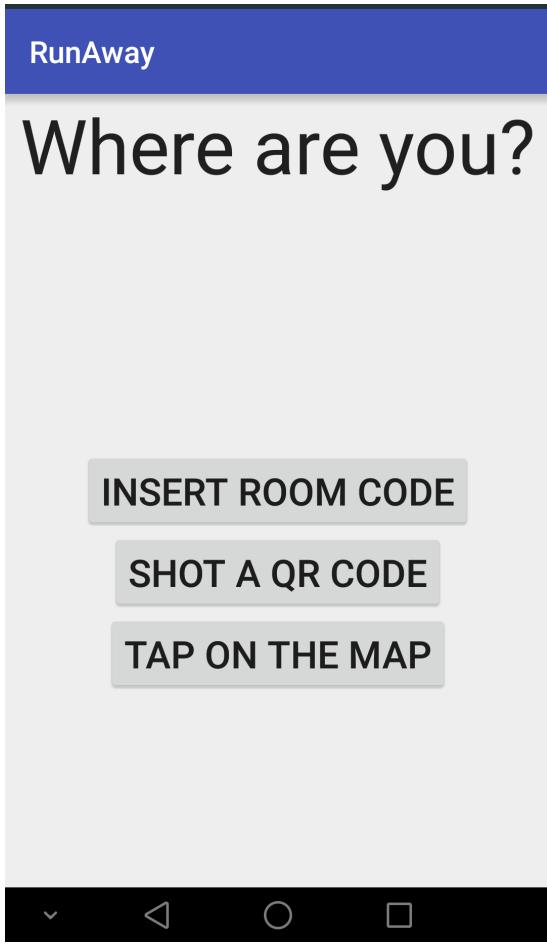
Dopo aver testato la navigazione in condizioni normali (ovvero di non emergenza), si vuole ora testare il caso in cui l'utente abbia scelto di navigare in modalità di emergenza (o abbia aperto una notifica inviata dal server). In questo caso la differenza sta nei valori che l'algoritmo prende come variabili di costo, da cui calcolarsi il nuovo peso degli archi. Essendo il server disponibile, possono essere aggiornate prima di un eventuale ricalcolo del percorso. Inoltre, una volta stabilito il nodo di partenza, la destinazione viene settata automaticamente al punto di raccolta più vicino (informazione contenuta nel database, come descritto nella Story Card 5).

Task ID: 5.5 Test di navigazione in emergenza

Si testeranno ora i passaggi per ottenere un percorso in emergenza per raggiungere il punto di controllo più vicino. All'apertura dell'app, supponendo che il login sia già stato memorizzato:



Selezionando quindi il tasto corrispondente alla modalità di emergenza (messo in evidenza per permettere un rapido feedback verso l'utente), si passa alla scelta della posizione iniziale.



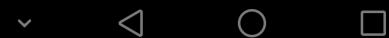
RunAway

Where are you?

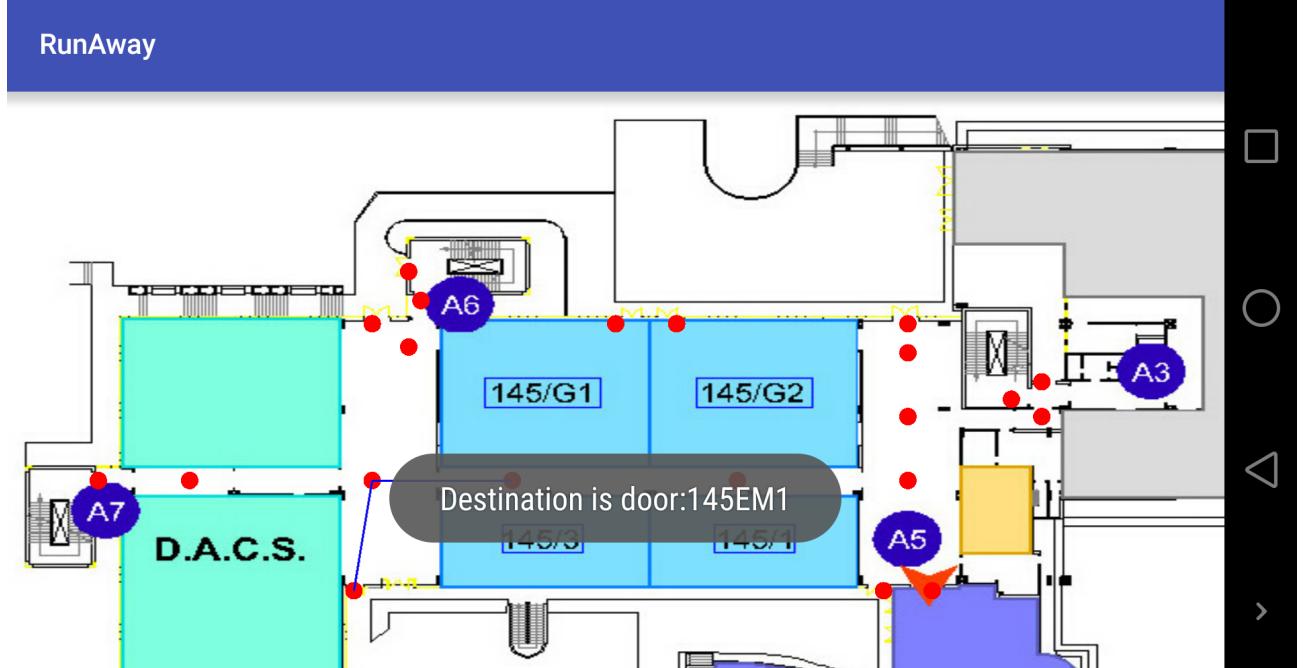
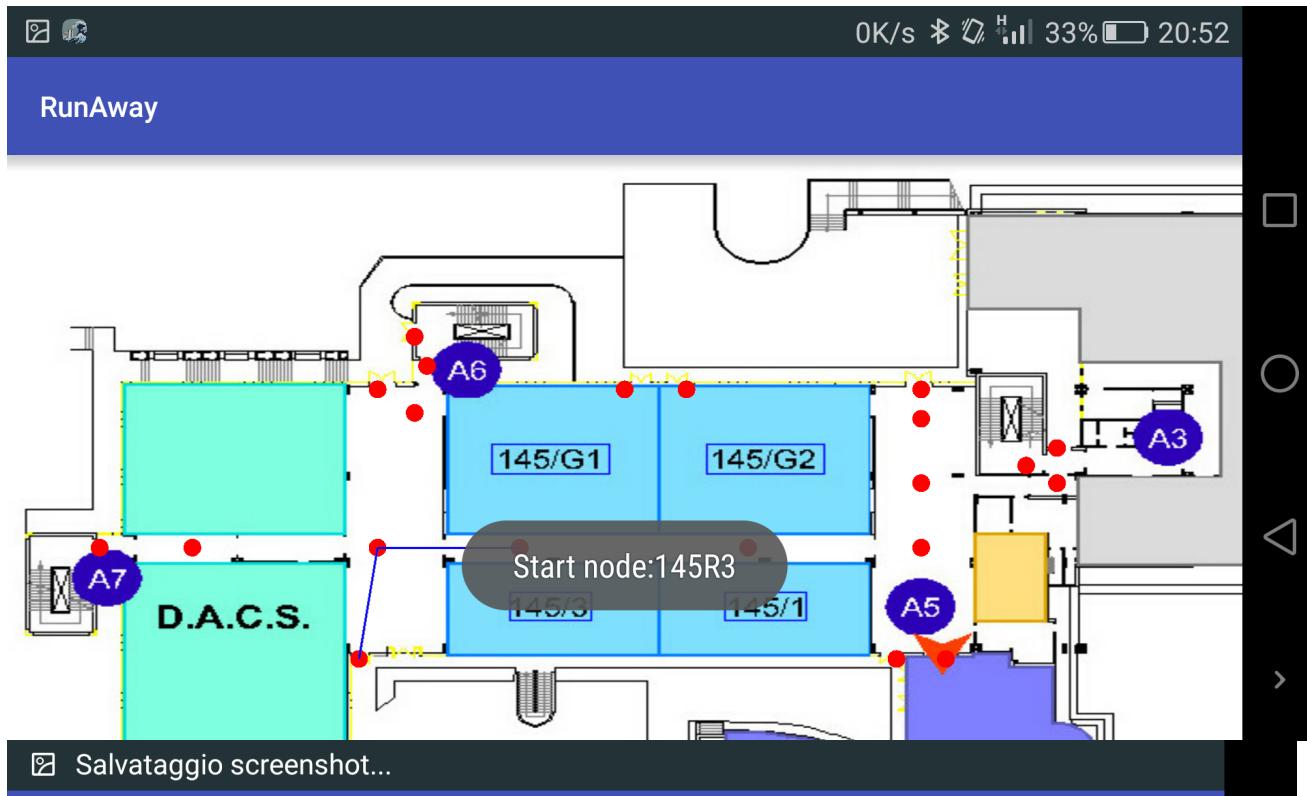
INSERT ROOM CODE

SHOT A QR CODE

TAP ON THE MAP

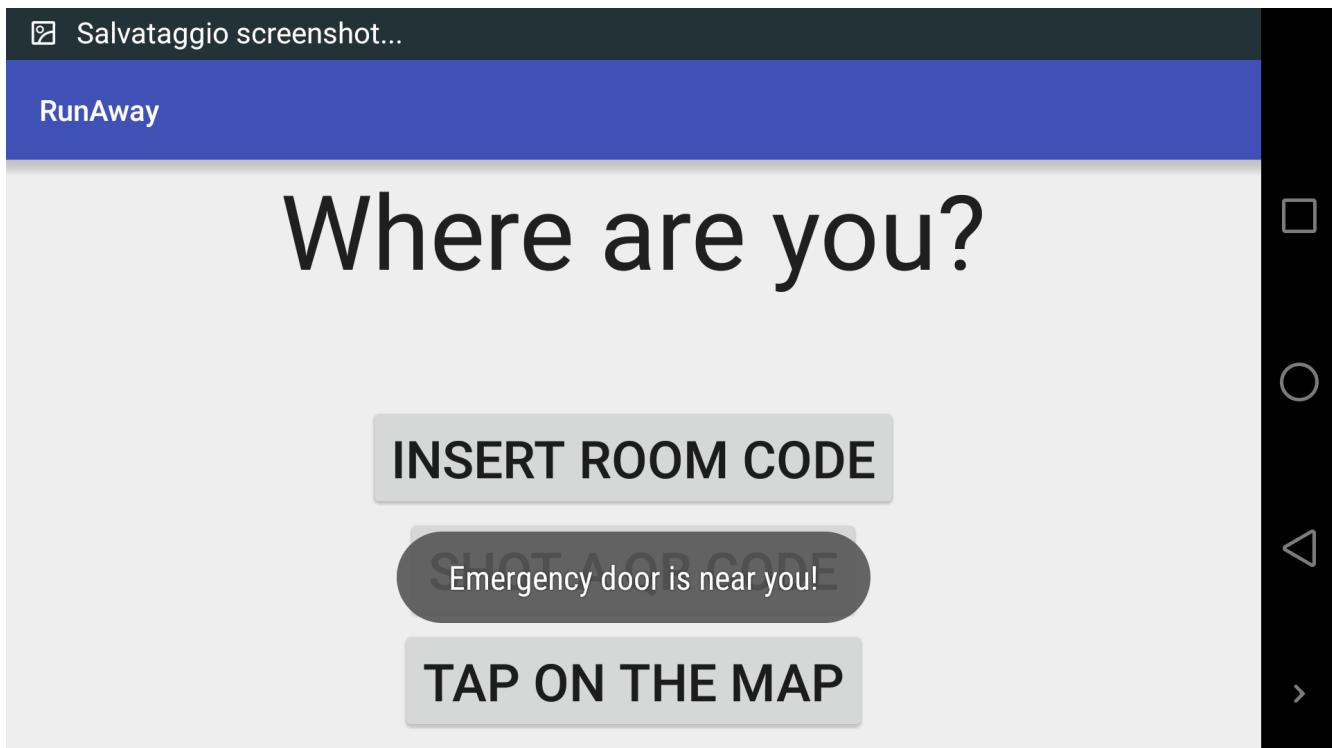
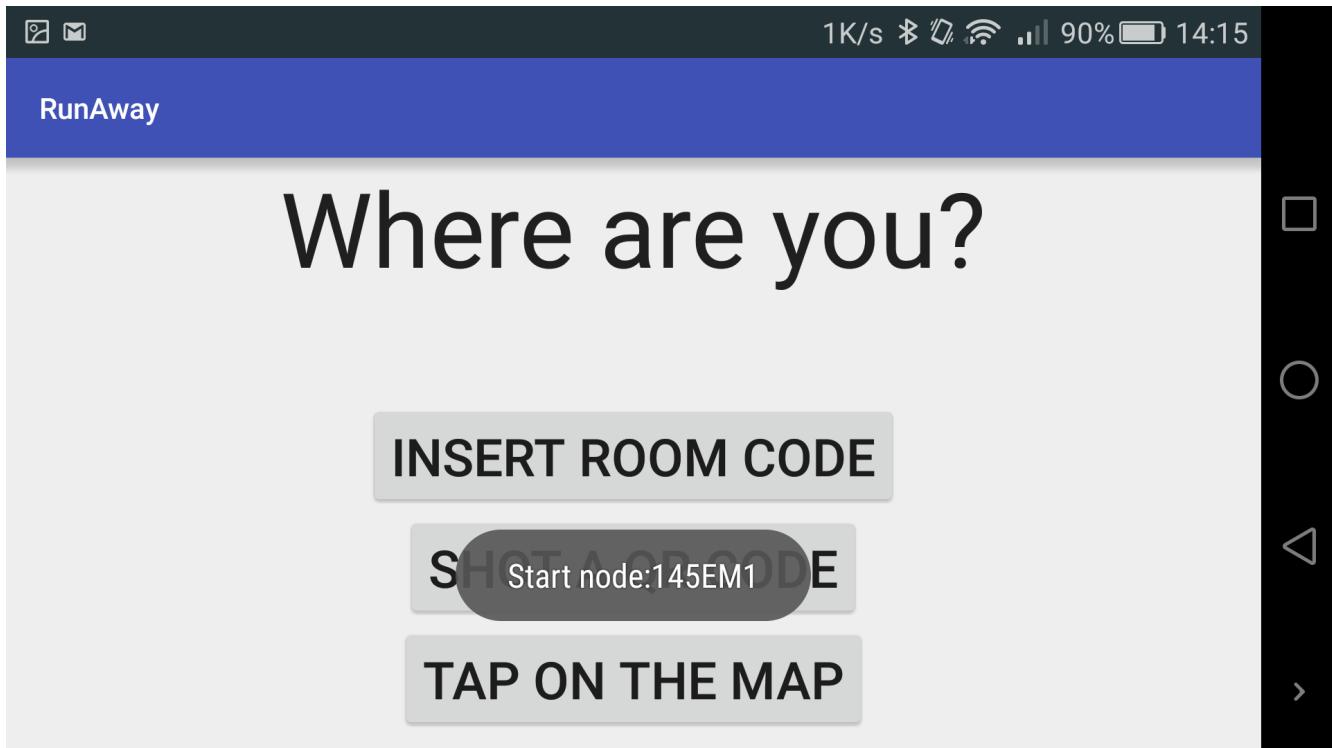


A questo punto, essendo già testate le varie modalità d'inserimento nel Task 2.5, si suppone di scegliere con un tap sulla mappa (dopo aver scelto il piano corrispondente) il nodo 145R3 come posizione iniziale.



Viene quindi impostato automaticamente come destinazione il nodo 145EM1 (porta di emergenza), e tracciato subito l'unico percorso possibile in questo caso.

Se invece venisse scelto come nodo iniziale la stessa porta di emergenza 145EM1, verrà notificato all'utente il fatto che si trovi già vicino alla destinazione, e si ritornerà alla schermata di selezione del punto iniziale (con l'informazione sulla modalità di emergenza attiva già impostata).

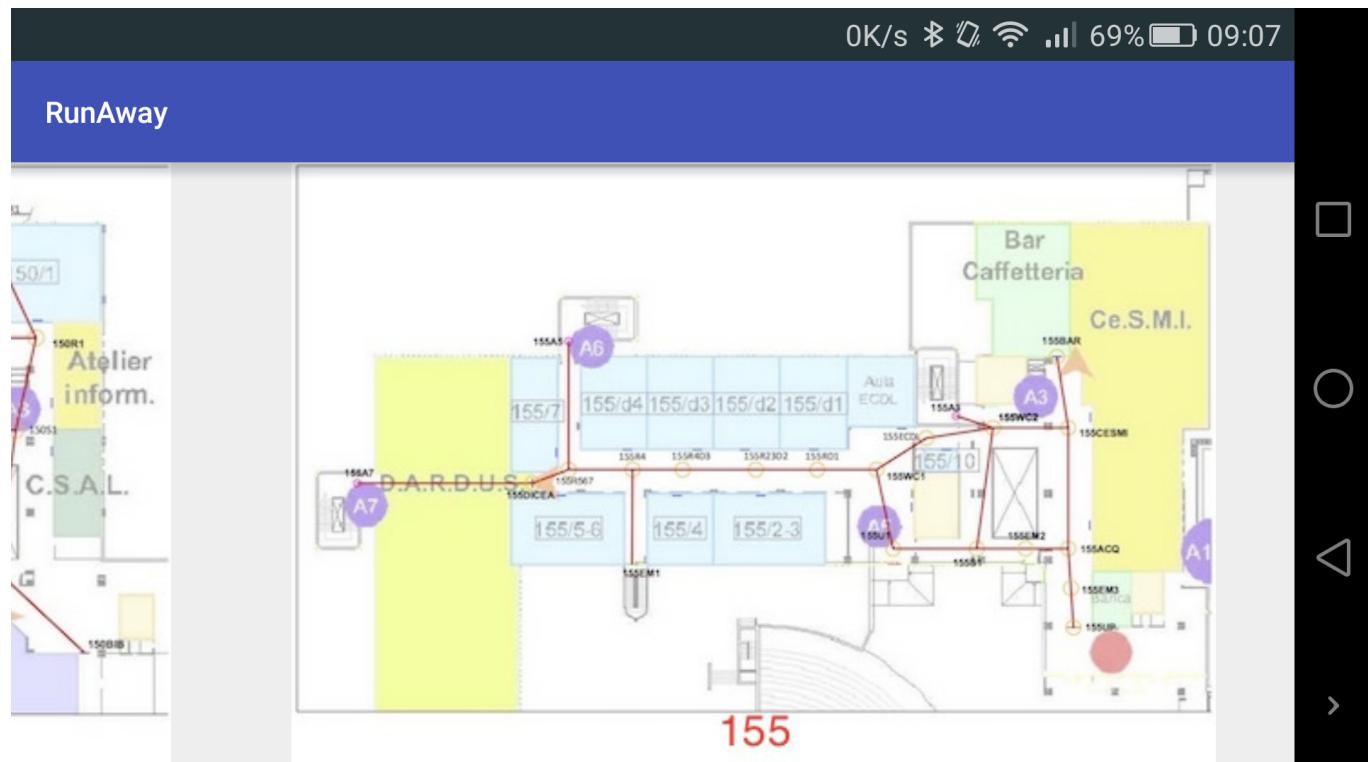


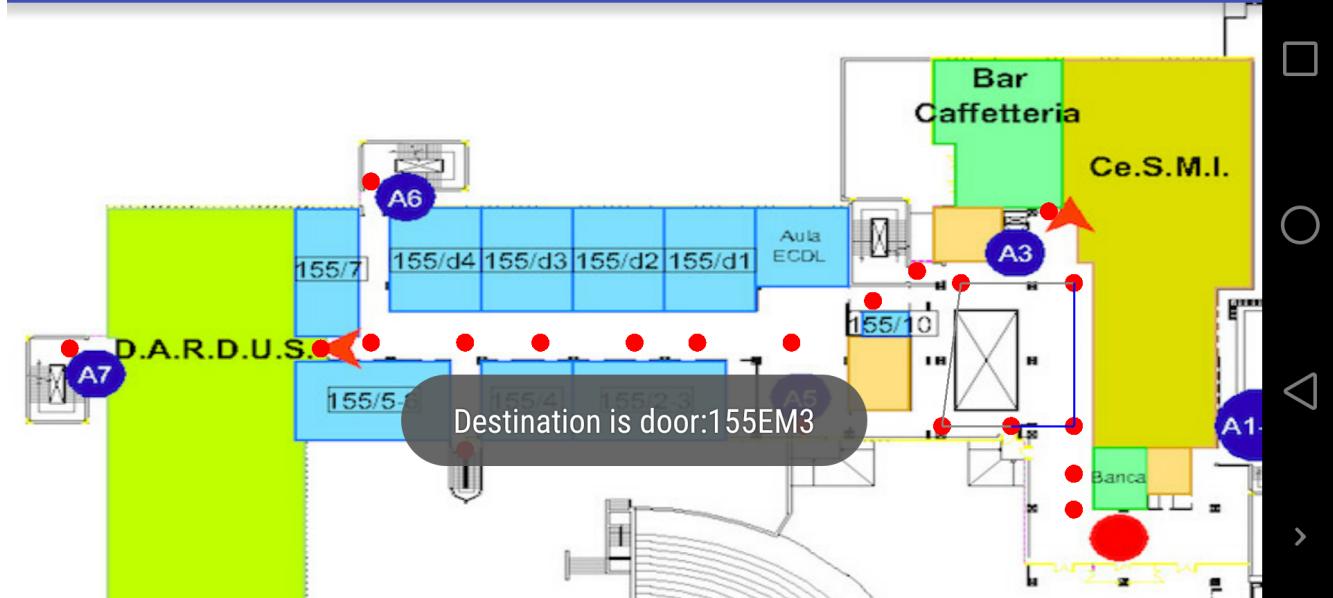
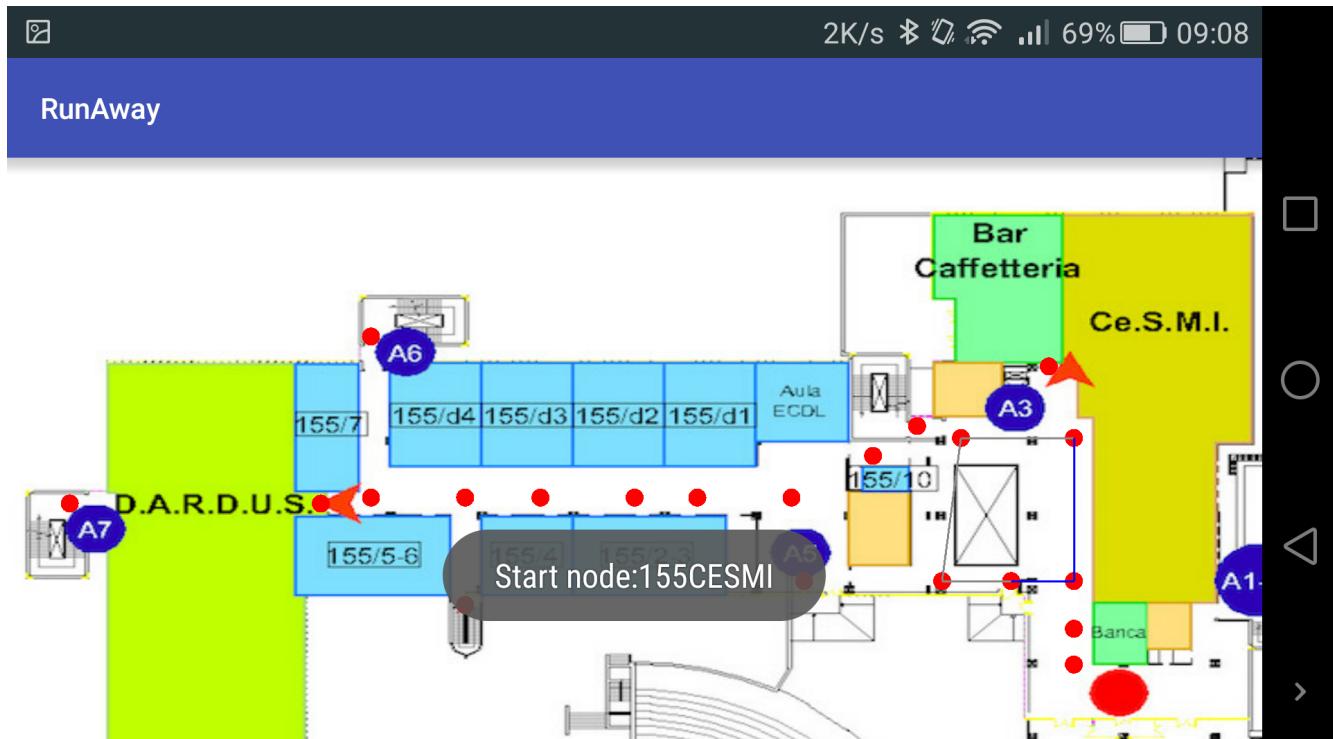
Task ID: 5.6 Controllo per aggiornamento variabili di costo dal server

Per controllare come l'app si comporti all'aggiornamento di eventuali variabili di costo da parte del server (ad esempio per tratti divenuti impraticabili durante l'emergenza), si è scelto di procedere nel seguente modo:

1. Creazione di un file “serverJson.json”, all'interno della cartella “assets” del progetto, nel quale inserire le modifiche alle variabili di costo relative all'ID specificato.
2. Aggiunta nel metodo che aggiorna la posizione iniziale sul Server, di una funzione di risposta diretta che vada ad aggiornare il database con le nuove variabili.
3. Prova di un percorso in emergenza senza il precedente aggiornamento.
4. Prova dello stesso percorso in emergenza con l'aggiornamento stabilito.

Supponendo quindi di trovarsi sul nodo 155CESMI, si mostra il percorso ottenuto in emergenza con le variabili di costo di default (numeri casuali compresi tra 0 e 1):





Si ottengono quindi due possibili percorsi.

Si supponga ora l'impraticabilità del tratto 155CESMI->155ACQ (secondo nodo del percorso più breve in blu).

Ricaviamo dal database l'ID dell'arco in questione e aggiungiamo in "serverJson.json" delle variabili di costo molto alte, che rappresentano quasi una probabilità nulla di poter scegliere l'arco.

The image shows two side-by-side JSON editors. The left editor is titled 'db.json' and shows a single edge entry under the 'edges' key:

```

edges
{
  "id": 45,
  "nodo1": "155ACQ",
  "nodo2": "155CESMI",
  "l_arco": 16,
  "v": 0.933709822466,
  "i": 0.0846823639478,
  "los": 0.872405057034,
  "c": 0.0725581491721,
  "scala": false
}
  
```

The right editor is titled 'serverJson.json' and shows the same edge entry with modified values:

```

edges
{
  "id": 45,
  "v": 100,
  "i": 100,
  "los": 100,
  "c": 100
}
  
```

Reinstalliamo l'applicazione, e ripetiamo i passaggi precedenti per iniziare la stessa navigazione in emergenza a partire dal nodo 155CESMI.



Possiamo quindi vedere come dai due percorsi possibili ci si sia ridotti ad uno solo, avendo escluso la possibilità di procedere per il precedente percorso più breve. Per riportare l'applicazione alle condizioni di default, ricordarsi di sostituire nel file "serverJson.json" le variabili di costo originarie per l'arco ignorato.

7.6 Guida al punto di raccolta in modalità offline

Task ID: 6.5 Test di navigazione in emergenza

Come ampiamente spiegato, la navigazione verso il punto di raccolta in modalità offline funziona sostanzialmente come la navigazione in condizioni normali, solo che utilizza gli ultimi pesi per gli archi eventualmente ricevuti dal server l'ultima volta che la connessione era presente. Per il test vero e proprio si rimanda al Task ID 5.5.

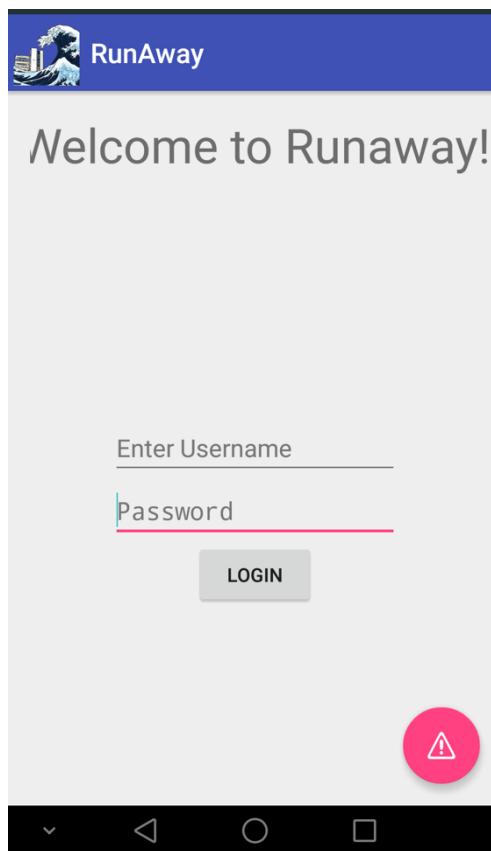
In questo caso, non avviene quindi l'aggiornamento dei pesi relativi agli archi.

8. MANUALE D'USO

Nel seguito vengono riportate le informazioni necessarie per poter utilizzare correttamente l'applicazione "RunAway".

L'applicazione potrà funzionare sia in modalità offline, che online, se è disponibile un server che si interfacci con l'app stessa. Nel presente manuale viene tralasciata la parte relativa alle funzionalità fornite da un eventuale Server, il cui comportamento è stato simulato in fase di test.

All'avvio dell'applicazione ci si trova davanti la schermata di login.



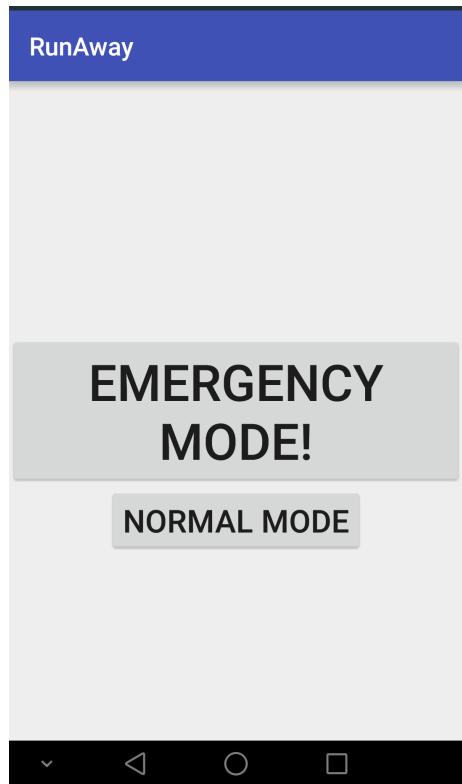
Questa include sia i campi per inserire le credenziali, sia il tasto FAB che permette di bypassarle nel caso in cui il server non sia disponibile per motivi di emergenza.

Per proseguire inserire i seguenti dati:

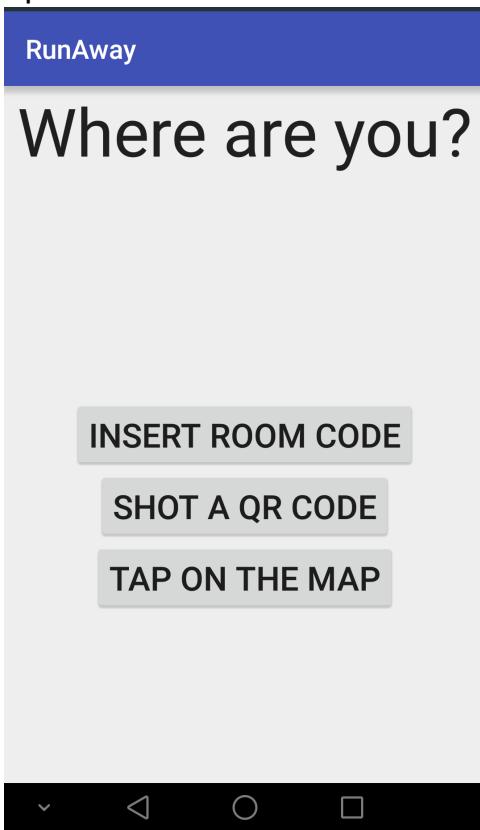
Username: admin

Password: admin

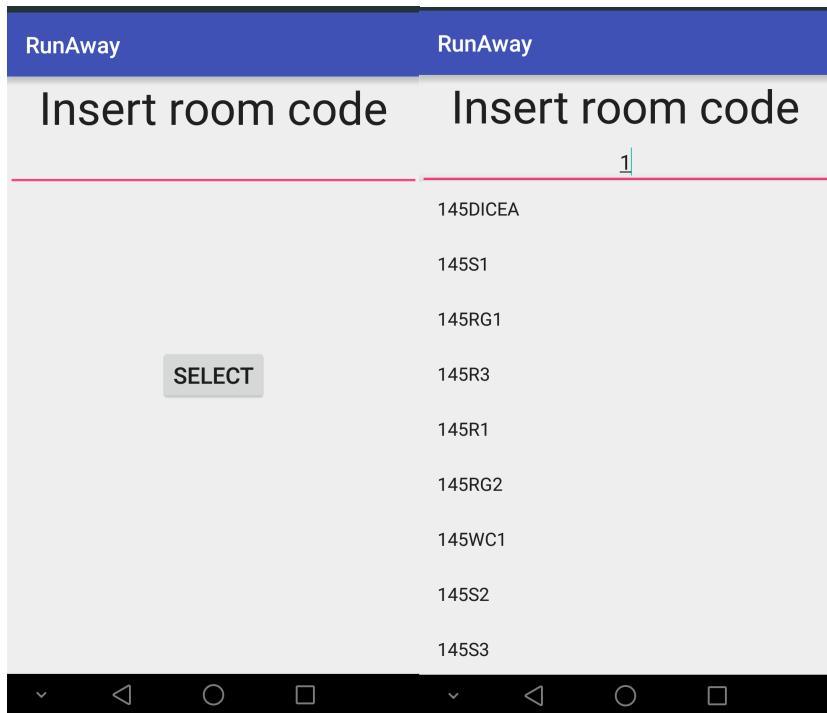
A questo punto, compare la homepage vera e propria, che permette di selezionare la modalità di navigazione.



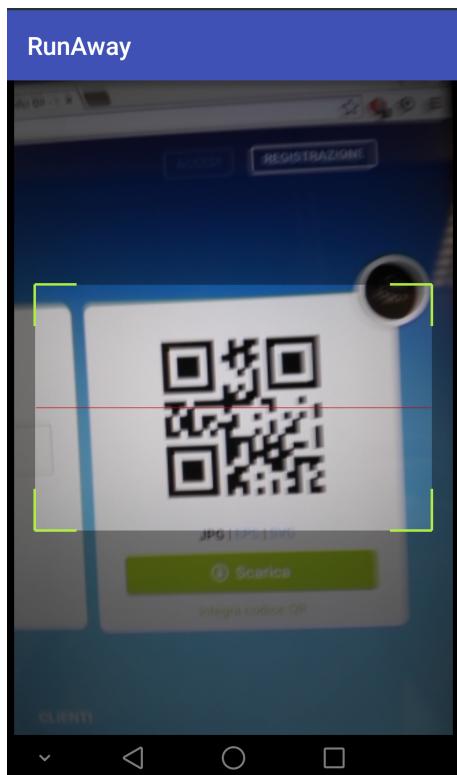
La differenza tra le due modalità sta nel fatto che in emergenza non compariranno le schermate di selezione della destinazione (che verrà impostata automaticamente).
Schermata per la scelta della posizione iniziale:



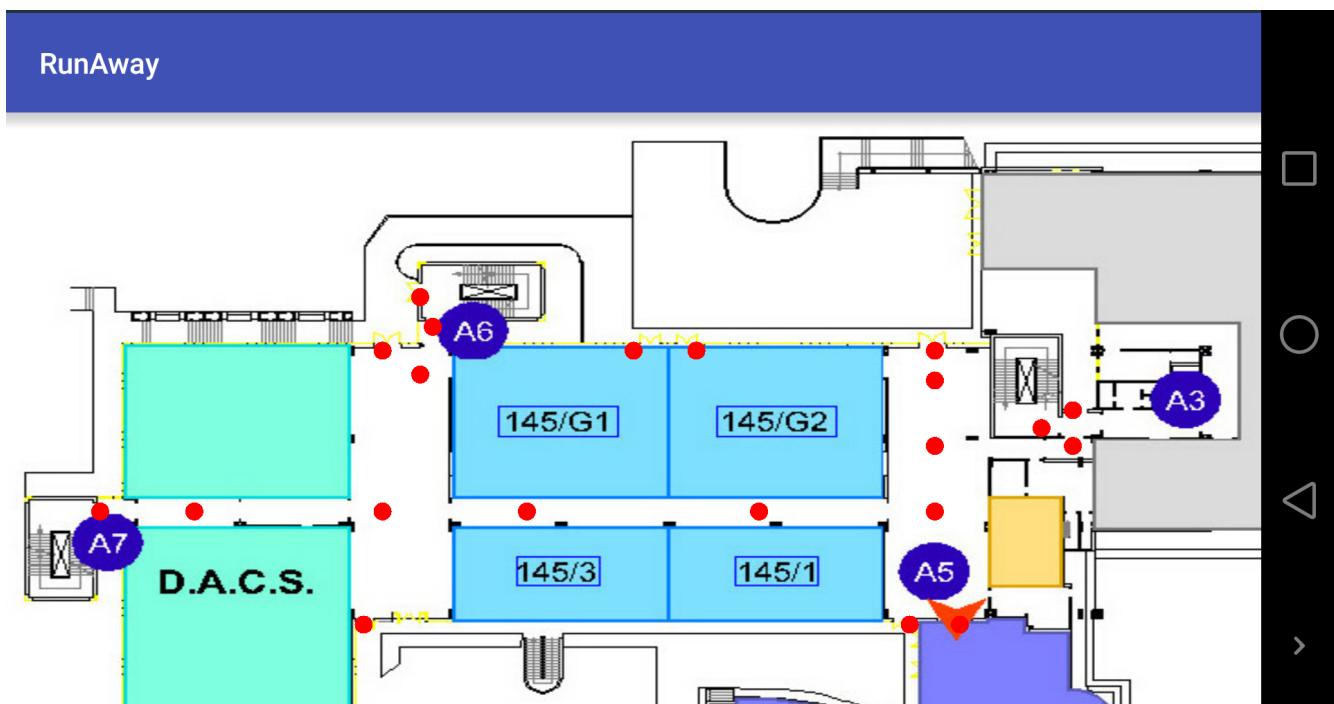
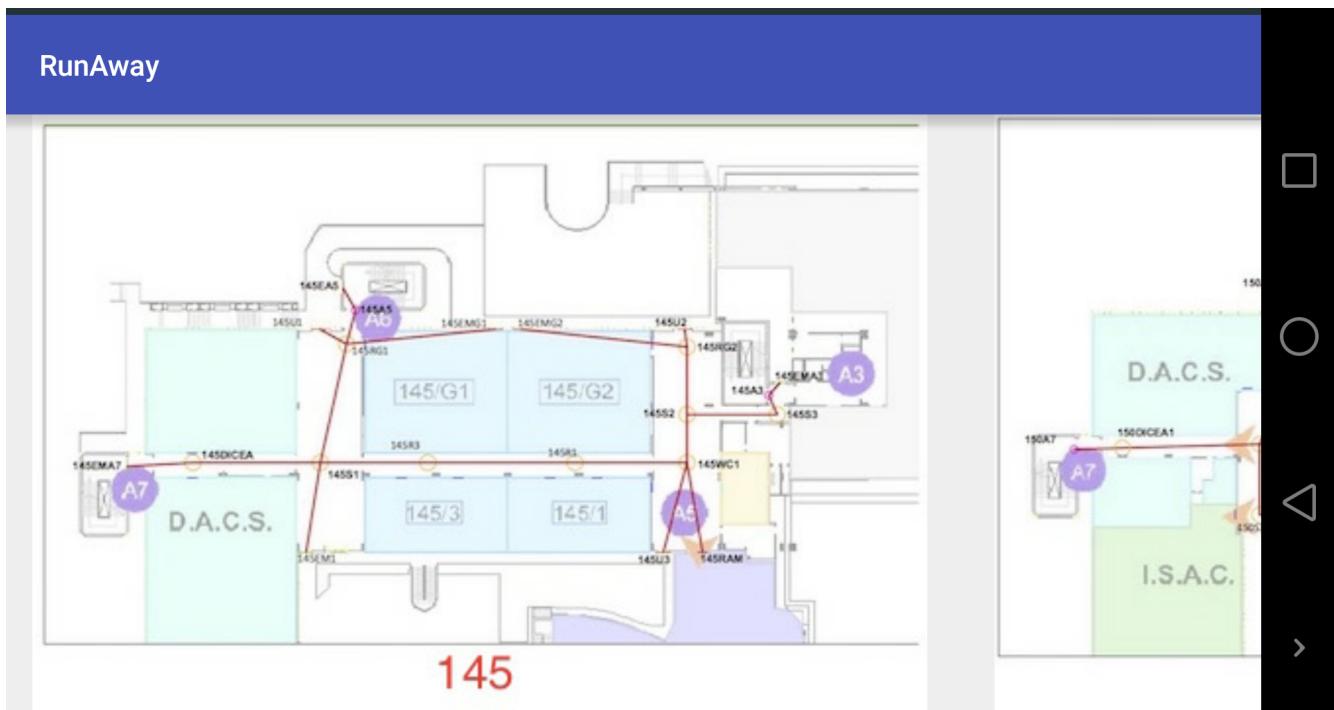
“Insert room code”



“Shot a QR Code”

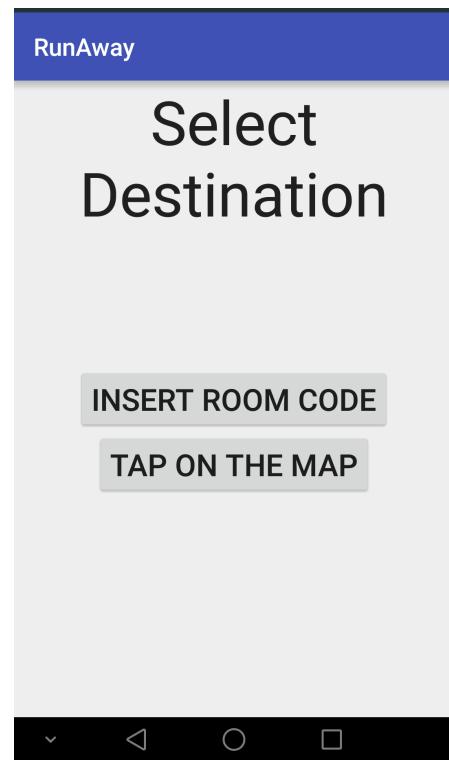


“Tap on the map”

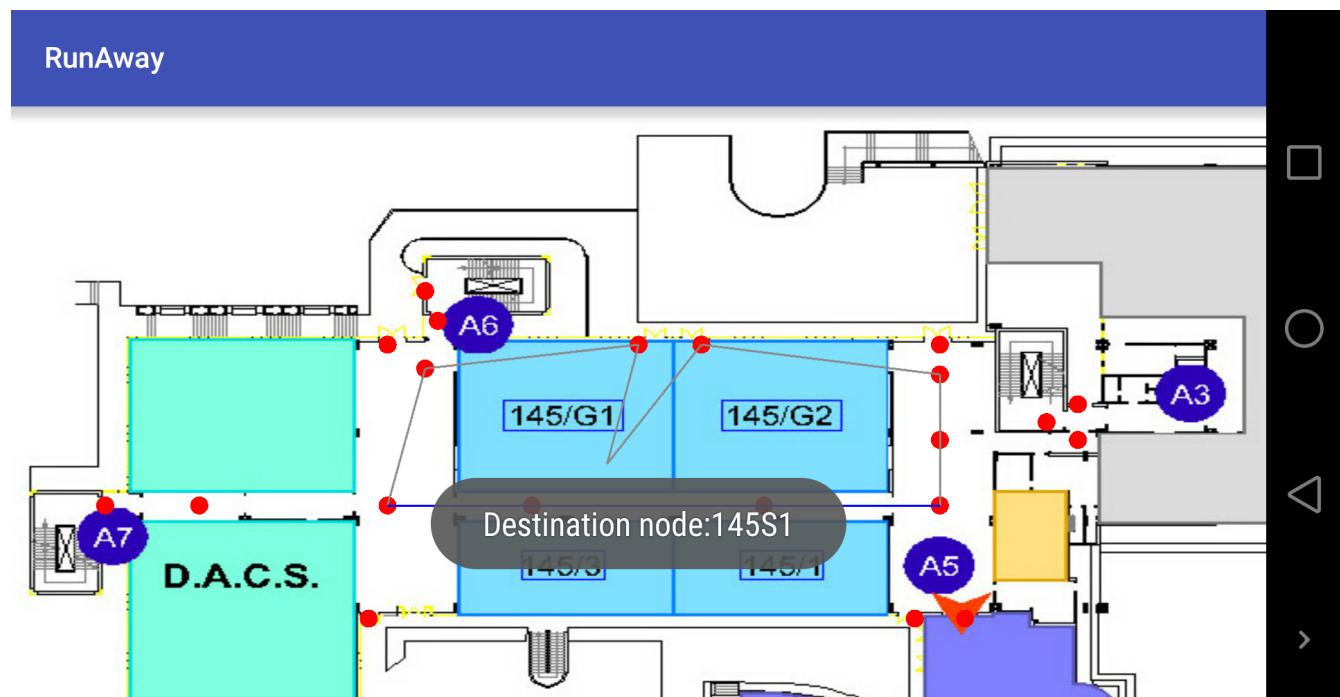


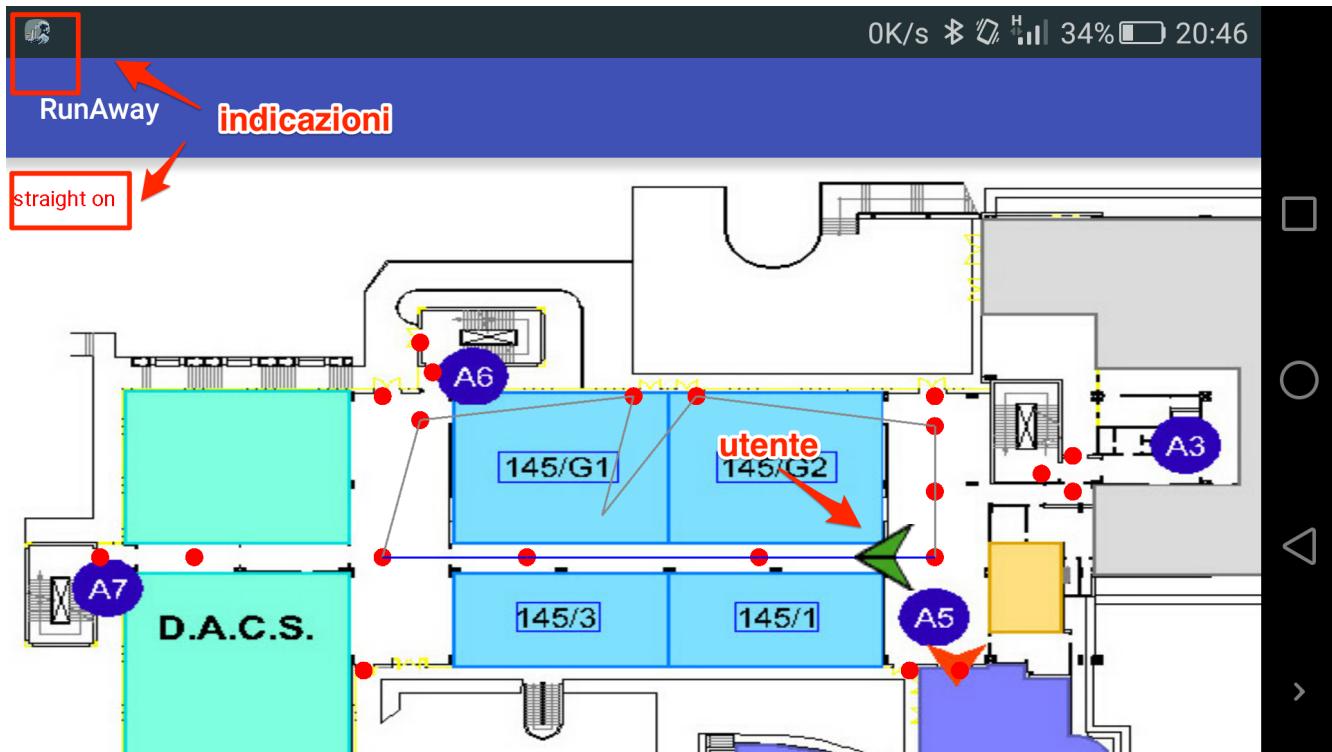
Si esegue quindi il tap su un nodo per impostare il nodo di partenza.

Per impostare la destinazione (non in modalità di emergenza), si avranno a disposizione le stesse modalità descritte in precedenza, a parte la cattura del QR Code.

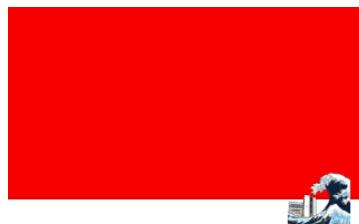


Una volta svolte queste operazioni verrà tracciato il percorso (o i percorsi possibili), per iniziare la navigazione si dovrà selezionare uno dei due.





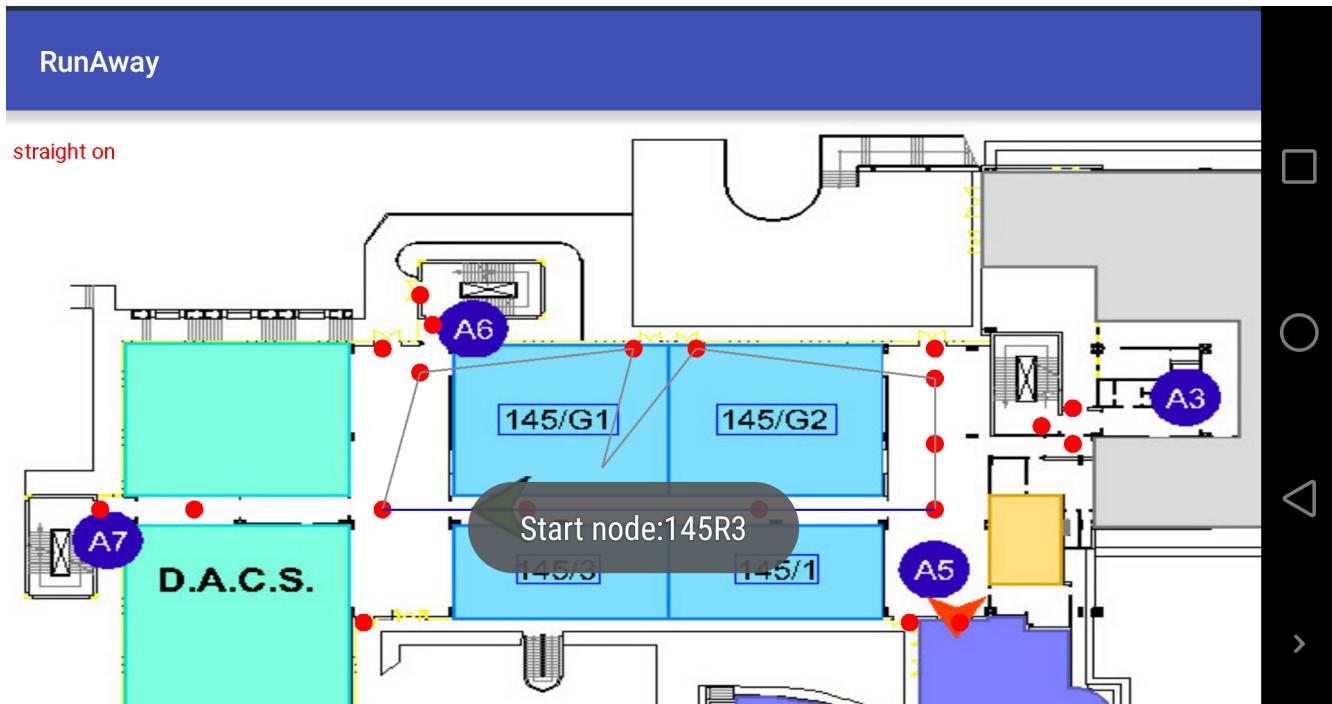
Indicazioni anche su Android Wear:



straight on

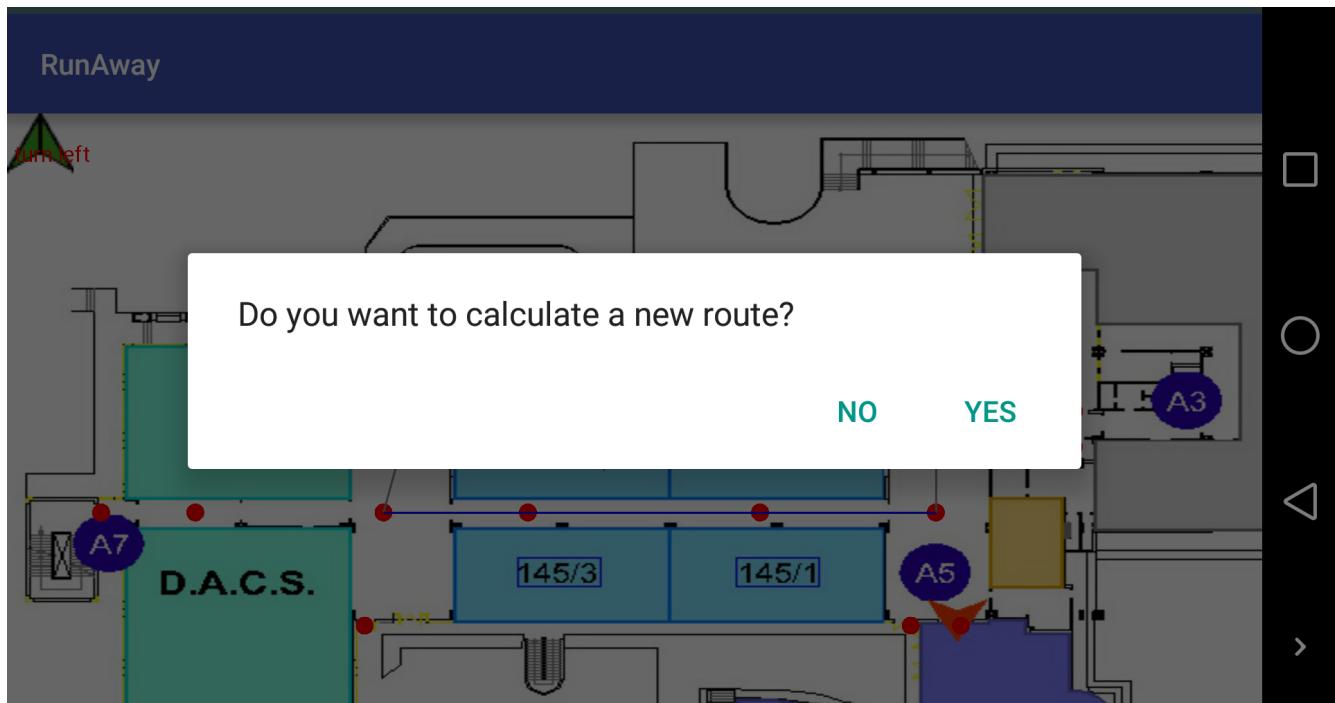


Per aggiornare la posizione lungo il percorso, basta toccare un nuovo nodo appartenente all'uno o all'altro tracciato.

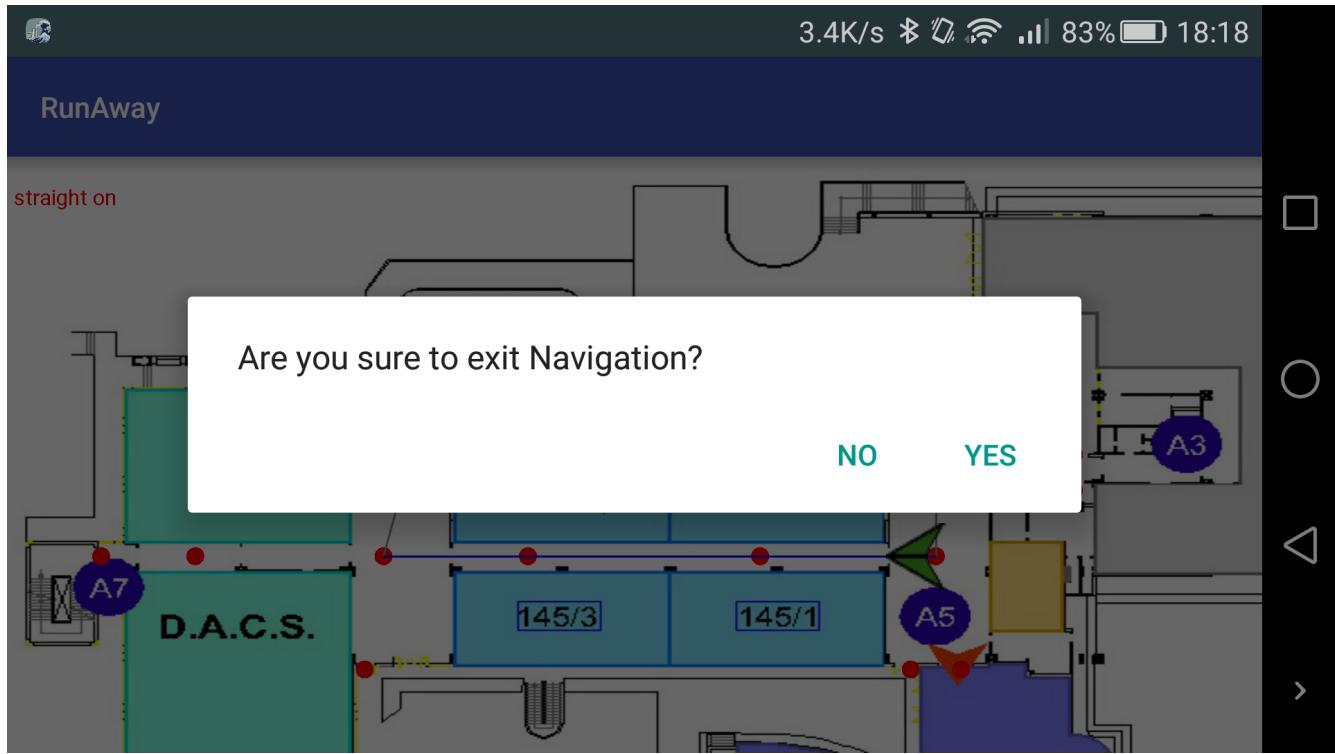


Per iniziare una nuova navigazione si può fare in due modi:

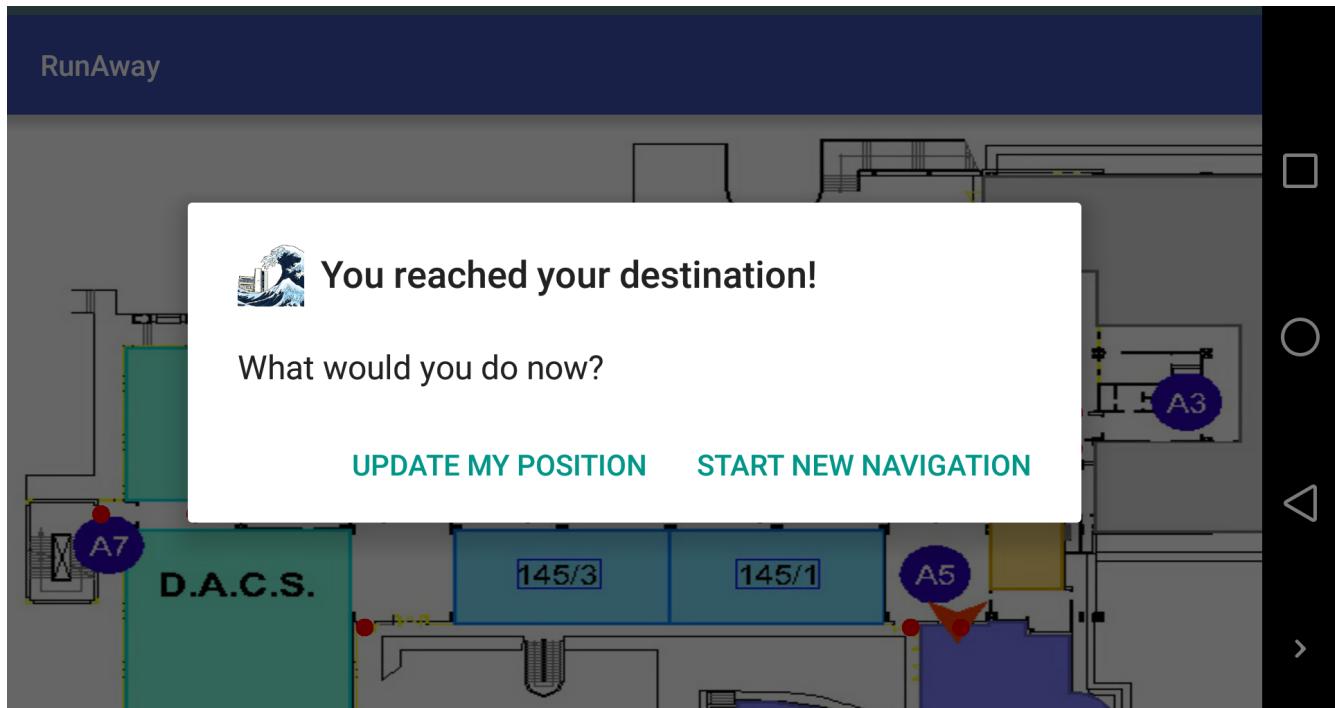
“Tap su un nodo non presente nei percorsi calcolati” - conserva in memoria la modalità di navigazione precedentemente impostata, ed imposta il nodo di partenza sul nodo selezionato.



“Tasto back” - Se viene premuto Yes, si ritorna alla selezione della modalità in homepage.



All’arrivo a destinazione comparirà la seguente finestra di dialogo.



“Update my position” - riapri la mappa e seleziona un nuovo nodo di partenza (mantiene la modalità di navigazione attuale)

“Start new navigation” - ritorna alla homepage per la selezione di una nuova modalità.

9. MANUALE D'INSTALLAZIONE

La presente applicazione è stata sviluppata per Android 4.2+ Jellybean (min SDK 17), utilizzando l’IDE Android Studio 2.0, e testata su dispositivi che montano la versione 5.1 Lollipop. Per la corretta installazione si illustreranno i passi da seguire per impostare tutto il necessario a partire dalla cartella del progetto.

9.1 Installazione IDE Android Studio 2.0 e importazione del progetto

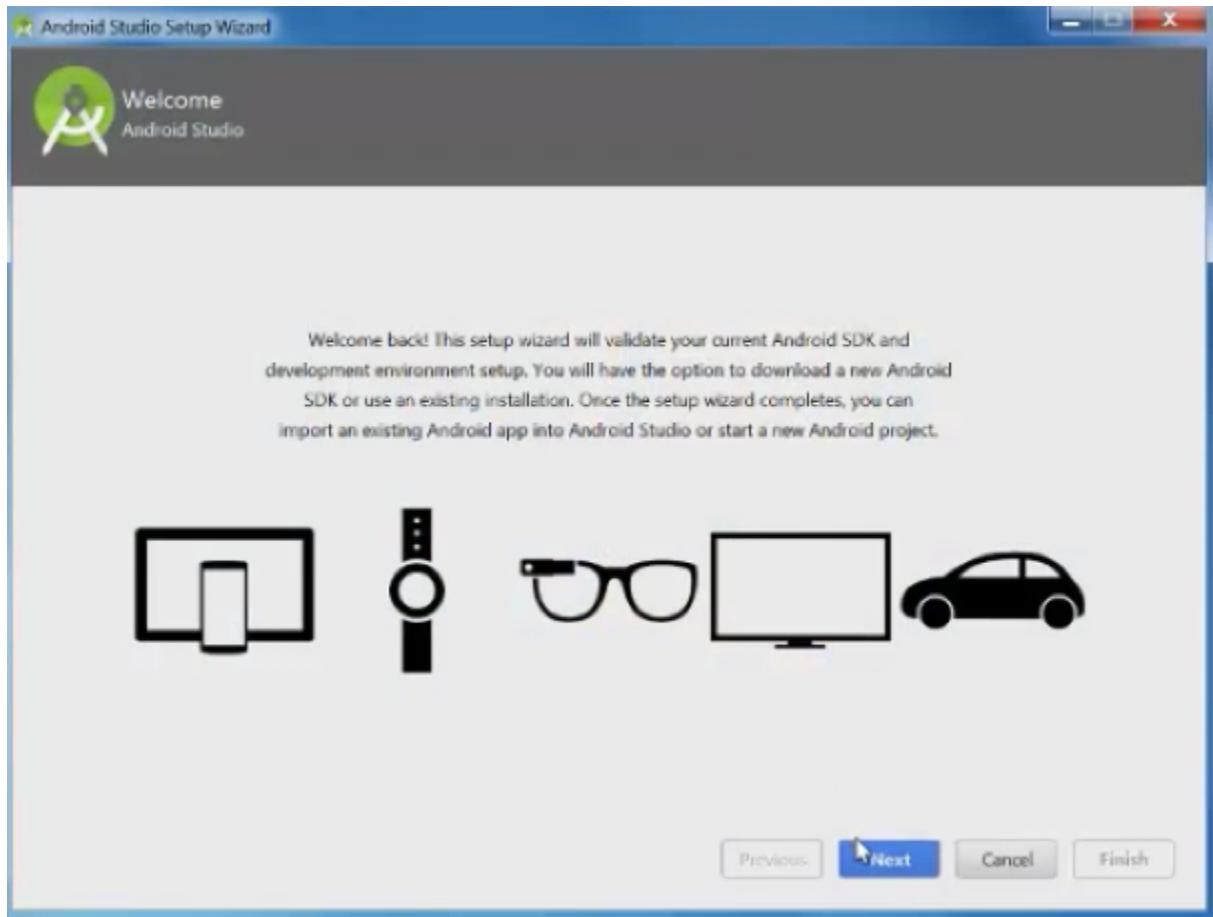
1. Scaricare la versione di Android Studio adatta per il sistema operativo in uso, dal sito degli sviluppatori: <https://developer.android.com/sdk>

The screenshot shows the official website for Android Studio. At the top, it says "Android Studio" and "The Official IDE for Android". Below that, it states: "Android Studio provides the fastest tools for building apps on every type of Android device. World-class code editing, debugging, performance tooling, a flexible build system, and an instant build/deploy system all allow you to focus on building unique and high quality apps." At the bottom left, there is a green button with white text that reads "DOWNLOAD ANDROID STUDIO 2.0 FOR WINDOWS (1166 MB)".

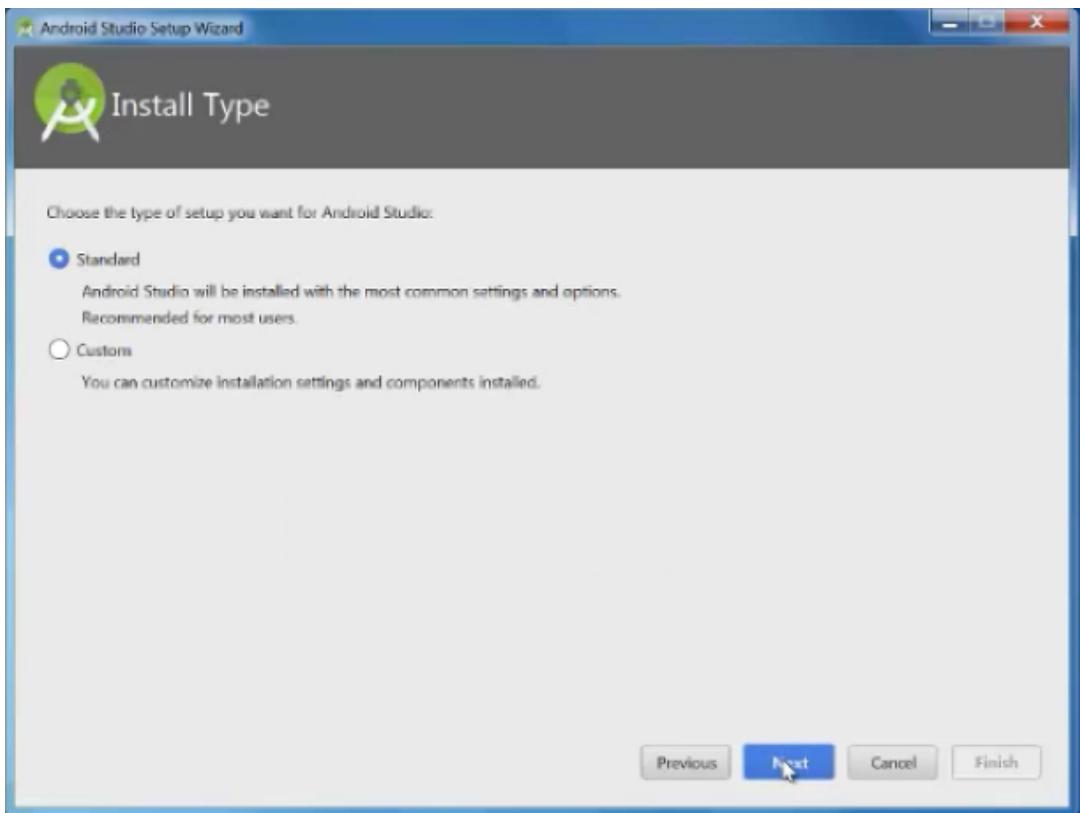
2. Aprire l’installazione, una volta che il download è terminato, e procedere lasciando la spunta su Android SDK.



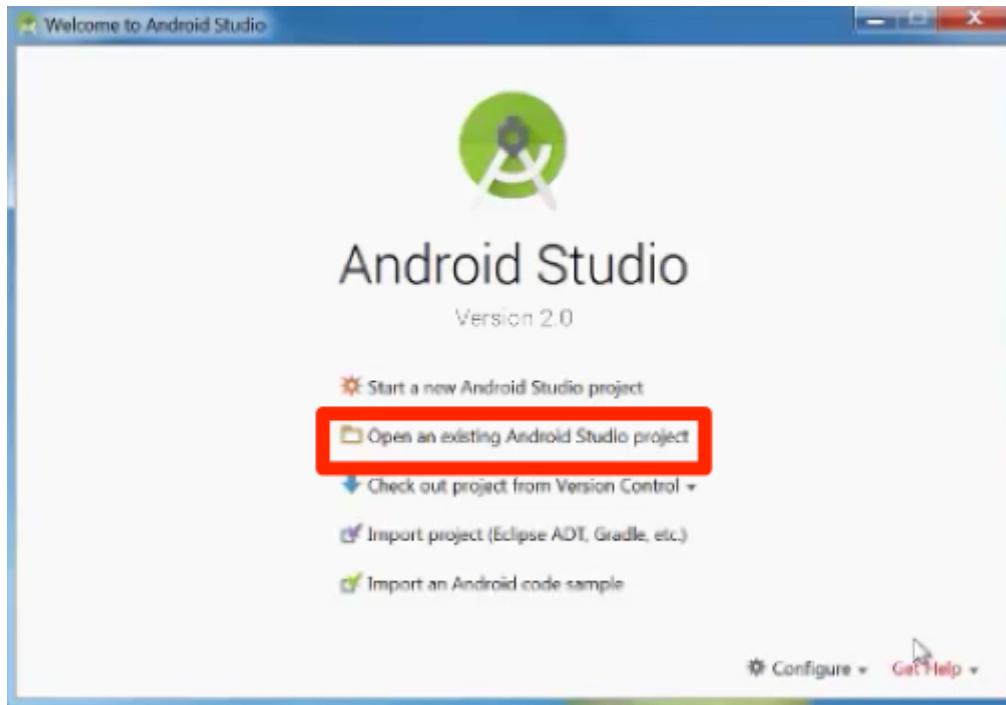
3. Una volta terminata, aprire Android Studio, e procedere sulla pagina di benvenuto.



4. Scegliere quindi un installazione Standard, e attendere il download dei componenti necessari all'applicazione.



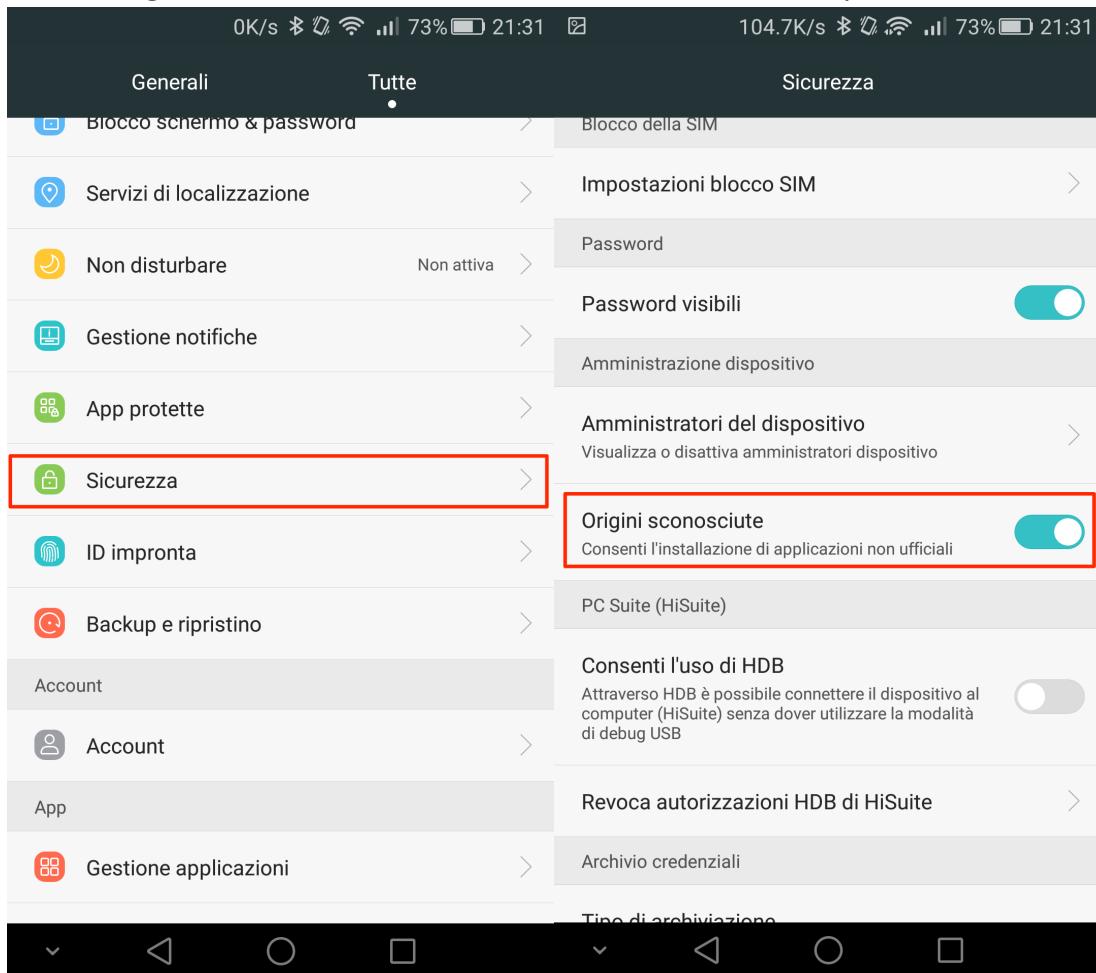
5. A questo punto, una volta aperta l'applicazione, selezionare l'opzione per importare un progetto Android Studio già esistente.



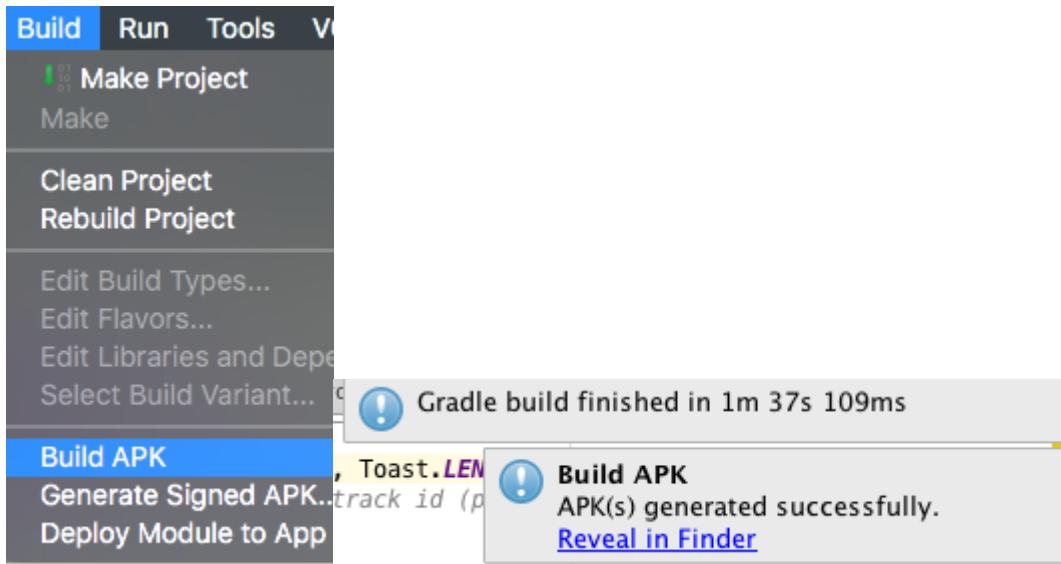
6. Selezionare l'intera cartella del progetto da importare, e attendere il processo di importazione.

9.2 Compilazione e installazione sullo smartphone

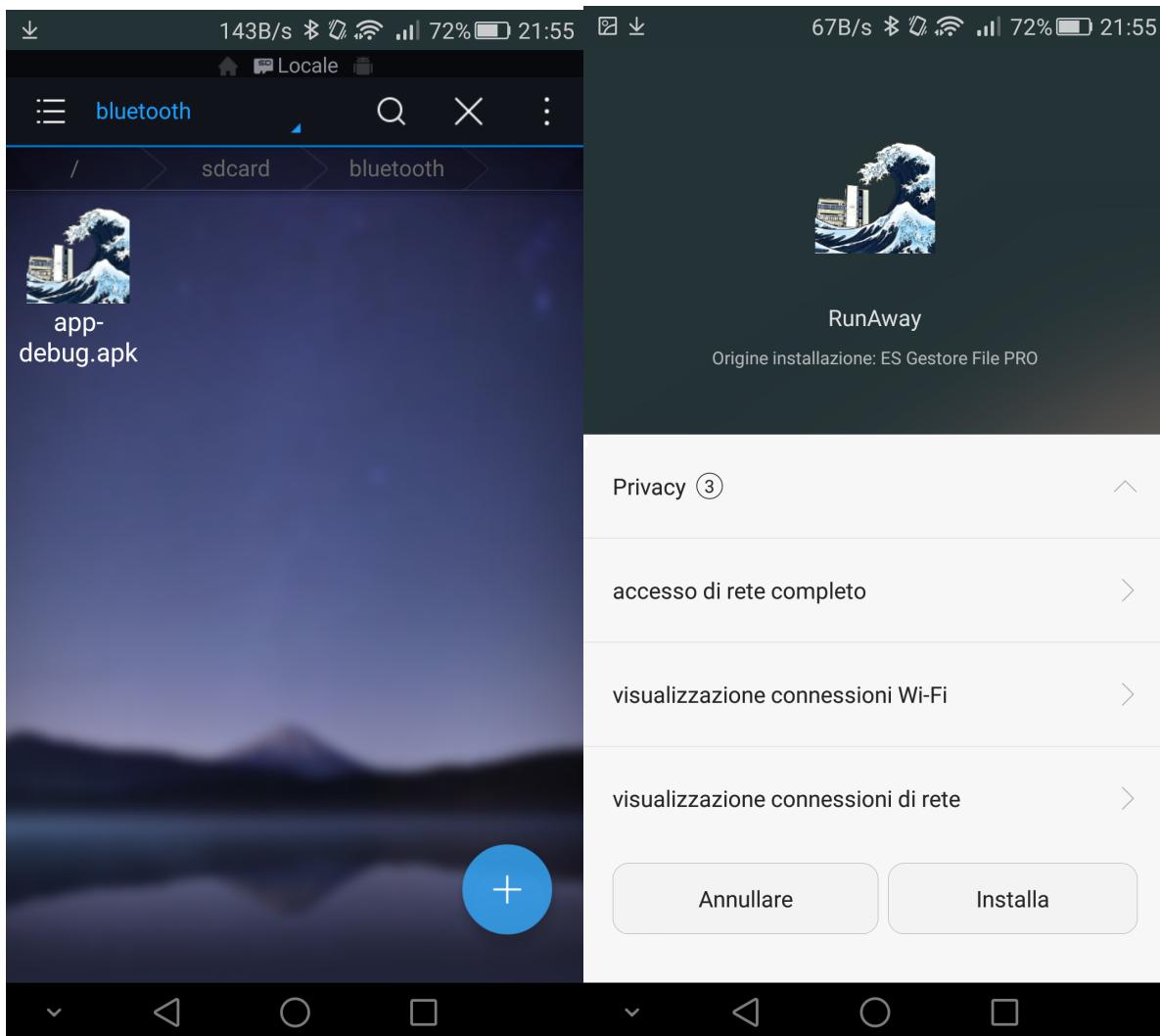
1. Procedere ad abilitare sullo smartphone la possibilità di installare applicazioni da origini sconosciute, nel menu ‘Sicurezza’ delle impostazioni.



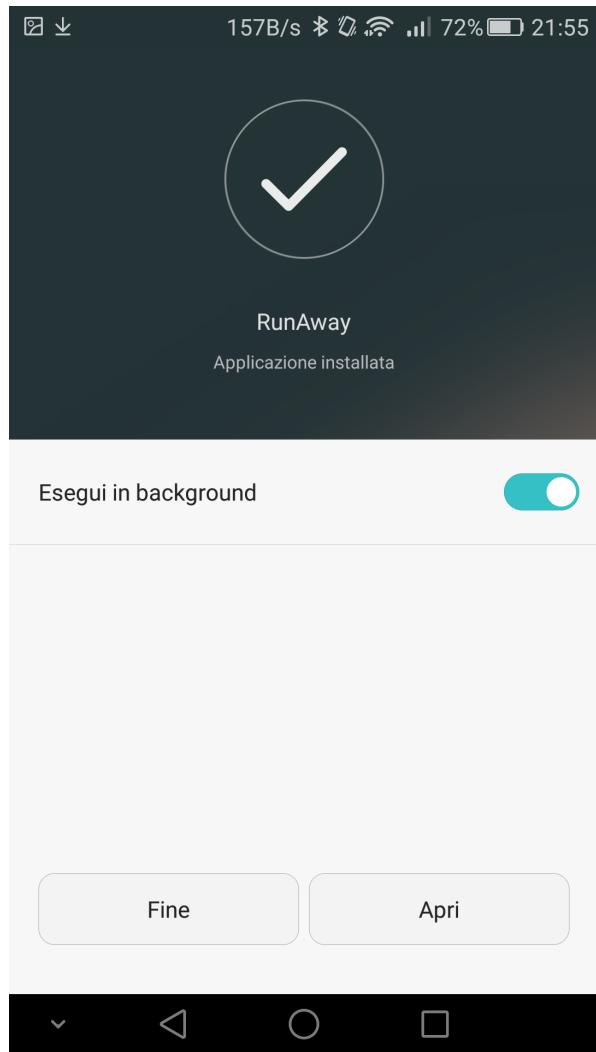
2. A questo punto, avendo aperto il progetto in Android Studio, procediamo con la creazione del file Apk, dal menu ‘Build’->‘Build APK’, e attendiamo il completamento.



3. Copiare il file APK generato nella memoria interna del cellulare (via USB ad esempio).
4. Attraverso un file manager, aprire il file APK sullo smartphone ed eseguire l'installazione.



5. Se richiesto, accettare la possibilità data all'app di rimanere in esecuzione in background, in modo che non venga interrotta la navigazione e le notifiche se si blocca lo schermo.



6. Cliccare su apri o aprire successivamente l'applicazione dal menu. Ora l'app è pronta per essere utilizzata.

9.3 Impostazione delle indicazioni su un indossabile Android Wear (Opzionale)

1. Installare l'applicazione Android Wear sullo smartphone, dal play store <https://play.google.com/store/apps/details?id=com.google.android.wearable.app>
2. Apri l'app Android Wear sul telefono e segui le istruzioni sullo schermo.
3. Sul telefono viene mostrato un elenco di dispositivi nelle vicinanze. Puoi controllare il nome dell'orologio sul suo schermo.
4. Tocca il nome del tuo orologio.

5. Sul telefono e sull'orologio viene mostrato un codice di accoppiamento. Assicurati che i codici corrispondano.
6. Tocca Accoppia sul telefono. Una volta accoppiato l'orologio, riceverai un messaggio di conferma.
7. Tocca Attiva notifiche sul telefono.
8. Tocca la casella di controllo "Android Wear" per attivare la visualizzazione sull'orologio delle notifiche di diverse app del telefono.
9. Assicurarsi che le notifiche per l'applicazione "RunAway" siano attive in Android Wear.