

“Software Engineering” Course a.a. 2019-2020

Lecturer: Prof. Henry Muccini (henry.muccini@univaq.it)

Progetto 5: Portami a destinazione

Date	05/12/2019
Deliverable	Milestone 1
Team (Name)	Team Banana

Team Members		
Name & Surname	Matriculation Number	E-mail address
Tagliente Gabriele	247295	gabriele.tagliente@student.univaq.it
Innamorati Giorgio	253380	giorgio.innamorati@student.univaq.it
Sborlini Valentina	247975	valentina.sborlini@student.univaq.it

Table of Contents of this deliverable

List of Challenging/Risky Requirements or Tasks	3
A. Stato dell'Arte	4
B. Raffinamento dei Requisiti	5
A.1 Servizi (con prioritizzazione)	5
A.2 Requisiti non Funzionali	7
A.3 Scenari d'uso dettagliati	8
A.4 Excluded Requirements	8
A.5 Assunzioni	9
A.6 Use Case Diagrams	10
C. Architettura Software	11
C.1 The static view of the system: Component Diagram	11
C.2 The dynamic view of the software architecture: Sequence Diagram	13
D. Dati e loro modellazione	15
E. Design Decisions	16
F. Design di Basso Livello	17
G. Explain how the FRs and the NFRs are satisfied by design	18
G. Effort Recording	20
Appendix. Prototype	22

List of Challenging/Risky Requirements or Tasks

Challenging Task	Date the task is identified	Date the challenge is resolved	Explanation on how the challenge has been managed
Manutenzione del database	22/11/2019	25/11/2019	Abbiamo deciso che il database una volta a settimana dovrà obbligatoriamente effettuare un backup su disco rigido. Così nel caso in cui il nostro database non funziona più sappiamo dove andare a riprendere i dati mancanti.
Segnalazione di nuovi POI	23/11/2019	24/11/2019	Abbiamo deciso che non diamo l'opportunità allo studente di segnalare nuovi POI in quanto non abbiamo personale che può andare a controllare se il POI segnalato è veritiero o fittizio. I POI verranno aggiornati solamente dopo segnalazione da parte dell'Università stessa.
Chat Supporto Studente-Università	23/11/2019	26/11/2019	Abbiamo deciso che non offriamo neanche una chat di supporto studente-supporto applicazione perchè non avendo personale che può seguire la pratica si rischierebbe che i messaggi resterebbero lì non letti. Abbiamo previsto che nel caso in cui lo studente ha problemi deve inviare una mail alla segreteria studenti che poi girerà la segnalazione ai gestori della piattaforma.

A. Stato dell'Arte

Stato dell'arte POI Team Banana

Prima di creare questo sistema ci siamo guardati intorno, e navigando in rete abbiamo trovato queste altre università italiane che possono offrire agli studenti dei sistemi innovativi come POI.

Il migliore che si può trovare in rete è quello della SAPIENZA DI ROMA, di cui lasciamo il link qui sotto.

<https://virtualtour.uniroma1.it/>

Altre università invece hanno semplicemente embeddato la mappa di Google My Maps con i loro punti di interesse all'interno del loro sito internet accompagnato da descrizioni testo HTML.

Lasciamo i link di queste altre università qui sotto.

<http://www.unife.it/studenti/en/mappa-di-ferrara>

<https://www.unibo.it/uniboWeb/unibomappe/default.aspx?site=bologna>

<http://old.unica.it/pub/9/show.jsp?id=5350&is=9&iso=1>

B. Raffinamento dei Requisiti

A.1 Servizi (con prioritizzazione)

- 1) Mappare tutti i POI di un Ateneo in una mappa. (Priorità **ALTA**)
- 2) Permettere allo studente di localizzare in modo semplice quanto di suo interesse (ovvero, risalire dal servizio necessario, al POI che fornisce tale servizio); (Priorità **BASSA**)
- 3) Se lo studente è registrato prioritizzare i POI del CdL e Dipartimento; (Priorità **MEDIA**)
- 4) Categorizzare i POI in base a CdL-specifico, Dipartimento-specifico, Università; (Priorità **ALTA**)
- 5) Mappare i POI in mappa con icone diverse a seconda del servizio offerto; (Priorità **BASSA**)
- 6) Utilizzare comandi vocali; (Priorità **BASSA**)

REQUISITI FONDAMENTALI

- 1. Studente (Cliente Finale dell' Applicazione)
 - 1.1 Deve Poter Scaricare L' Applicazione
 - 1.1.1 Deve Poter Registrarsi
 - 1.1.2 In Registrazione deve Fornire Dati Anagrafici
 - 1.1.3 In Registrazione deve fornire anche dati Universitari Anagrafici (Matricola-Email di ateneo)...
 - 1.2 Deve Poter Loggare
 - 1.2.1 Deve Poter Accedere all'applicazione attraverso Social Network
 - 1.2.2 Deve Poter Accedere all'applicazione attraverso servizi Mail.
 - 1.3 Una volta loggato per la prima può accedere alla mappa con tutti i POI interni alla sua università presente nelle anagrafiche Universitarie.
 - 2.1 Lo studente può selezionare i propri interessi in una sezione dedicata nel proprio profilo personale.
 - 2.1.1 Lo studente in base ai propri interessi può filtrare tali POI.
 - 3.1 Lo studente in fase di registrazione deve selezionare il proprio CDL e dipartimento e la sede di appartenenza
 - 3.1.1 Lo studente in base ai proprio CLD e dipartimento e in base alla sede di appartenenza può prioritizzare nella mappa i propri POI.
 - 4.1 Deve poter visualizzare nella mappa i POI colorati/rappresentati diversamente in base al servizio offerto
 - 5.1 Deve poter usare i comandi vocali
 - 5.1.1 Deve poter usare i comandi vocali per raggiungere un determinato POI
 - 5.1.2 Deve poter usare i comandi vocali per catalogare i POI in base ad una nuova ad una nuova ricerca
 - 5.1.3 Deve poter usare i comandi vocali per registrarsi nell'applicazione (CASO NON VEDENTE)
 - 5.1.4 Deve poter usare i comandi vocali per loggare nell'applicazione (CASO NON VEDENTE)
-

A.2 Requisiti non Funzionali

- E' obbligatorio creare un' account interno all' applicazione.
 - Bisogna essere in possesso di credenziali Universitarie.
 - Il sistema deve rispettare il Regolamento Europeo in materia di Privacy UE 2016/679
 - Il sistema deve rispettare il Regolamento Europeo in materia dei cookie UE 2016/679
 - Il sistema è indipendente da ogni singola università
 - Il Sistema deve avere una mappa perennemente aggiornata della propria università
 - Il sistema si deve integrare con il calendario docenti, in modo tale per permettere di trovare il Docente (POI) nel proprio orario di ricevimento all'interno della mappa
 - Deve poter funzionare con molteplici sedi, una quantità importante di POI e numerosi studenti
 - Deve poter funzionare con molteplici dispositivi (IOS,Android...)
 - Il sistema deve garantire una garanzia di utilizzo 16 ore su 24 (sono escluse le ore notturne in cui gli studenti non possono essere in università, in quanto quest'ultima è chiusa).
 - Il sistema deve prevedere un backup dei dati in caso di guasti
 - Il sistema deve avere un database capace di elaborare la richiesta di ogni singolo studente in pochi secondi.
 - Il sistema deve assicurarsi che nel caso un POI non è disponibile al momento (RISTRUTTURAZIONE,...) non venga visualizzato nella mappa.
-

A.3 Scenari d'uso dettagliati

SCENARI D'USO

Aperta l'applicazione per la prima volta l'utente dovrà registrarsi oppure potrà eseguire il login o attraverso i social network ,oppure attraverso le credenziali fornite dalla propria università (ESSE3 nel caso dell' università degli studi dell'Aquila). **NB** Ogni Studente deve possedere la mail associata ai social network **UGUALE** a quella presente nello storico delle anagrafiche dell' università, altrimenti in caso contrario lo studente avrà più Username e più password; Riconosciute nel sistema come account diversi.

Il sistema deve tenere traccia di tutti i login effettuati dagli studenti ed una vista dello storico di tutti i POI recentemente visitati.

Una volta completato questo primo step, l'utente troverà una mappa con tutti i POI che può raggiungere che potranno inoltre essere filtrati in base al proprio CDL oppure alla sede universitaria di appartenenza.

Una volta arrivati nella schermata principale lo studente potrà scegliere il POI che vuole raggiungere. Una volta selezionato il database dopo alcuni secondi fornirà allo studente tutte le indicazioni necessarie a raggiungere il punto di interesse.

Nel caso di utente non vedente la registrazione all'applicazione potrà essere effettuata attraverso i comandi vocali, idem nel caso in cui un utente non vedente voglia raggiungere un determinato POI.

A.4 Excluded Requirements

Tra i **Servizi Esclusi**, possiamo trovare:

Recensioni dei POI.

Ogni Studente, una volta arrivato a destinazione non potrà effettuare recensioni sull' ultimo POI recentemente visitato, in quanto sennò bisognerebbe creare una sezione apposita.

Magari in un'altra versione dell' applicazione.

Segnalazione nuovi POI

Ogni Studente che vuole segnalare un nuovo POI non potrà in alcun modo segnalare nuovi POI attraverso form interne al sistema in quanto, potrebbero arrivare numerose segnalazioni fittizie e non avendo personale a disposizione sarebbe abbastanza complicato gestirle.

Chat Supporto Studente-Gestore App

Nel caso in cui uno studente non riesce ad accedere all'Applicazione, Non gli vengono aggiornati i POI in base al suo CDL o sede universitaria, oppure semplicemente non trova i POI di suo interesse, potrà soltanto inviare una mail di assistenza alla segreteria studenti e non potrà ricevere una live chat di supporto, sempre per il motivo di mancanza di personale a disposizione.

Si riparerà il guasto non appena arriverà segnalazione dalla segreteria studenti.

No Criptazione dei dati

Abbiamo deciso in gruppo che non verrà effettuata nessuna criptazione dei dati, in quanto il sistema non presenta dati che possano essere definiti SENSIBILI.

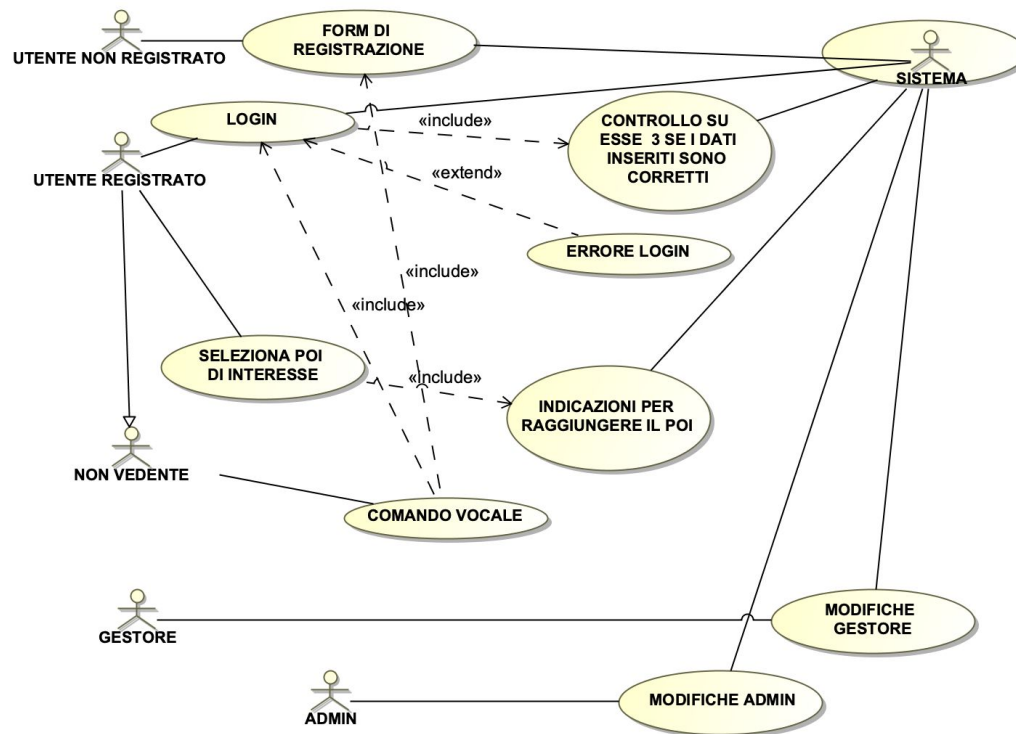
Ad eccezione dei dati di login che per **LEGGE** devono essere nascosti ai proprietari dell'applicazione.

A.5 Assunzioni

ASSUNZIONI PRINCIPALI

L'Assunzione Principale nel Progetto commissionatoci dal Prof. Henry Muccini, è quello di scegliere quale mappa usare (**Satellitare oppure non**) inoltre, altra principale assunzione interna al progetto è scegliere se l'applicativo può funzionare OFFLINE e quindi usare una mappa **OFFLINE** oppure debba funzionare solo **ONLINE** e quindi è possibile usare le mappe Google (My Maps) utilizzando anche noi **EMBED** oppure **IFRAME** come hanno fatto le altre principali università.

A.6 Use Case Diagrams



C. Architettura Software

C.1 The static view of the system: Component Diagram

Il component Diagram viene utilizzato per rappresentare la struttura interna dell'applicazione, modellando i componenti principali e le loro relazioni.

I componenti principali del nostro software sono i seguenti:

1. Login/Registrazione:

I dati che arrivano dall'autenticazione tramite Social Network oppure tramite credenziali ESSE3 o simili per le università convenzionate vengono processate dal SISTEMA (Database) per controllare la veridicità dei dati inseriti nel form.

2. Mappa:

Viene intesa come quella parte software che visualizza i POI di ogni studente sulla mappa in base al proprio CDL, alla sede universitaria di appartenenza oppure semplicemente in base ai propri interessi.

3. Navigatore:

Viene definito Tragitto tutta quella parte software chiamata tramite API a **TOOL ESTERNO** che porta il nostro Cliente (Studente Universitario) dalla propria posizione corrente a destinazione (POI selezionato).

4. Comando Vocale:

Questa componente invece, si occupa dell'utilizzo dell'applicazione da parte degli utenti NON VEDENTI, e li aiuta in fase di registrazione oppure di login. Il comando vocale non è altro che un traduttore che trasforma voce in testo e invia i dati inseriti dall'utente con il database.

5. Interfaccia Gestore:

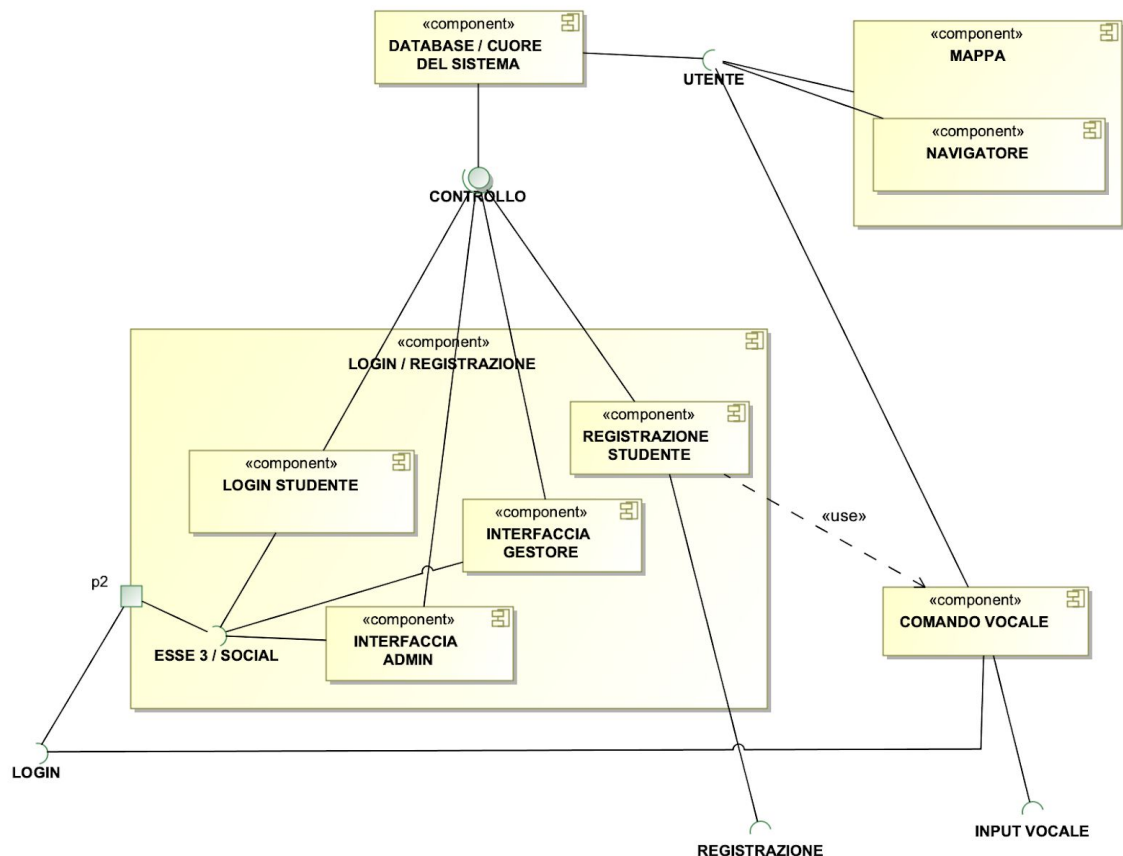
E' la parte del sistema accessibile dai gestori, essi dopo aver superato la fase di login, potranno andare a modificare la parte dei dati relativa ai corsi di studi degli studenti, oppure potranno direttamente agire sui POI (cancellarli, aggiungerli, segnalarli temporaneamente OFFLINE).

6. Database / Cuore del sistema

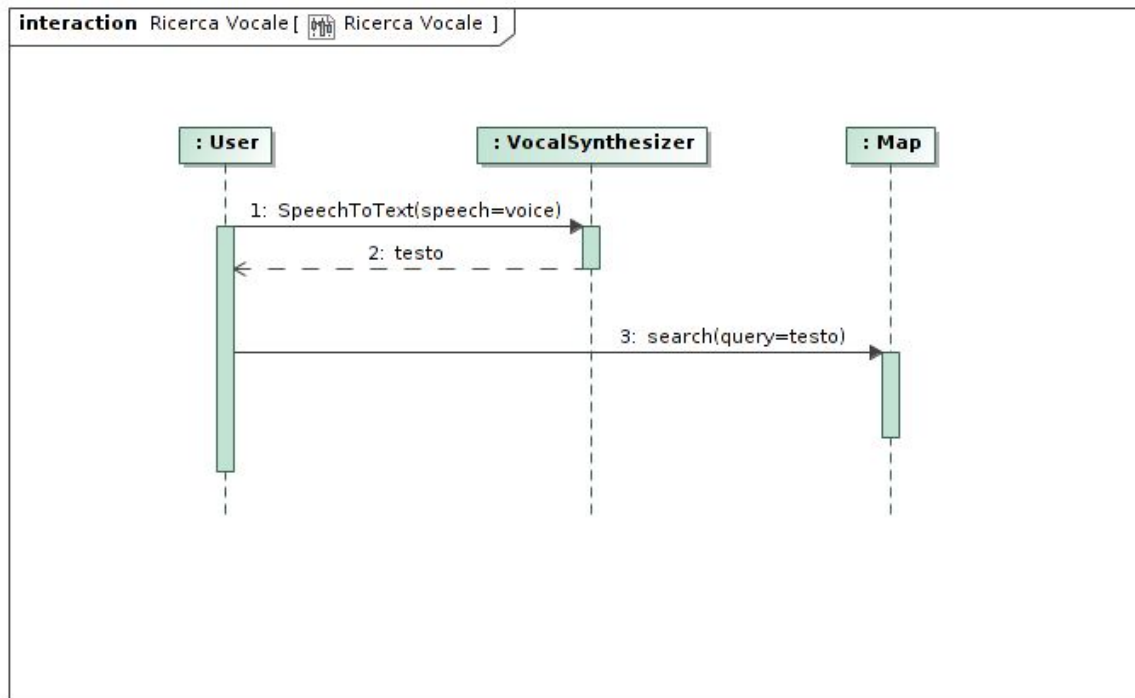
Componente che si occupa degli accessi degli studenti all'interno dell'applicazione e dell'assegnazione del tragitto più breve dalla posizione corrente dello studente al POI selezionato.

7. Interfaccia Admin

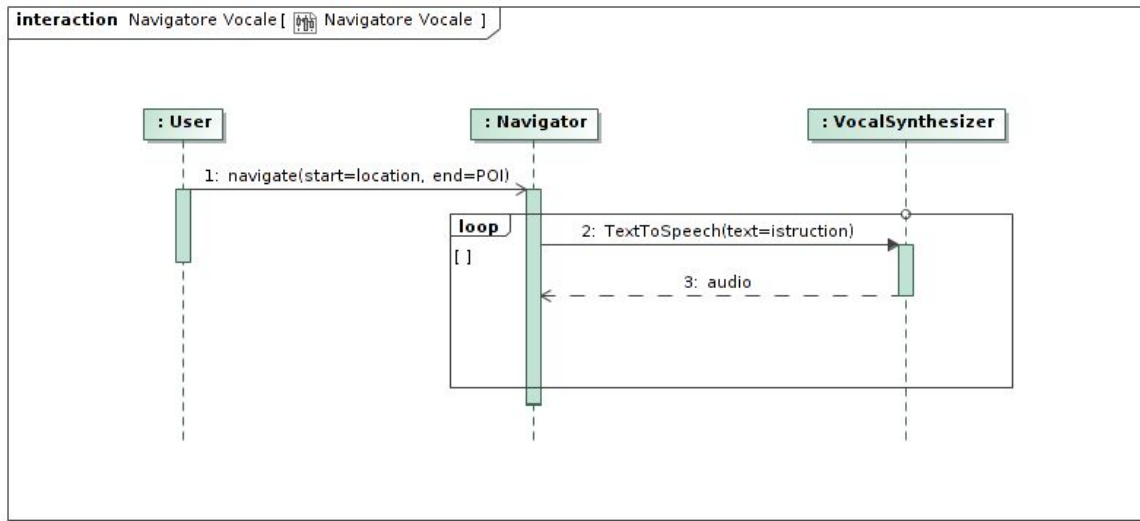
Come per l'interfaccia gestore è la parte accessibile solo dagli admin che possono svolgere tutti i compiti dei gestori e oltre a tali servizi possono aggiungere nuovi gestori.



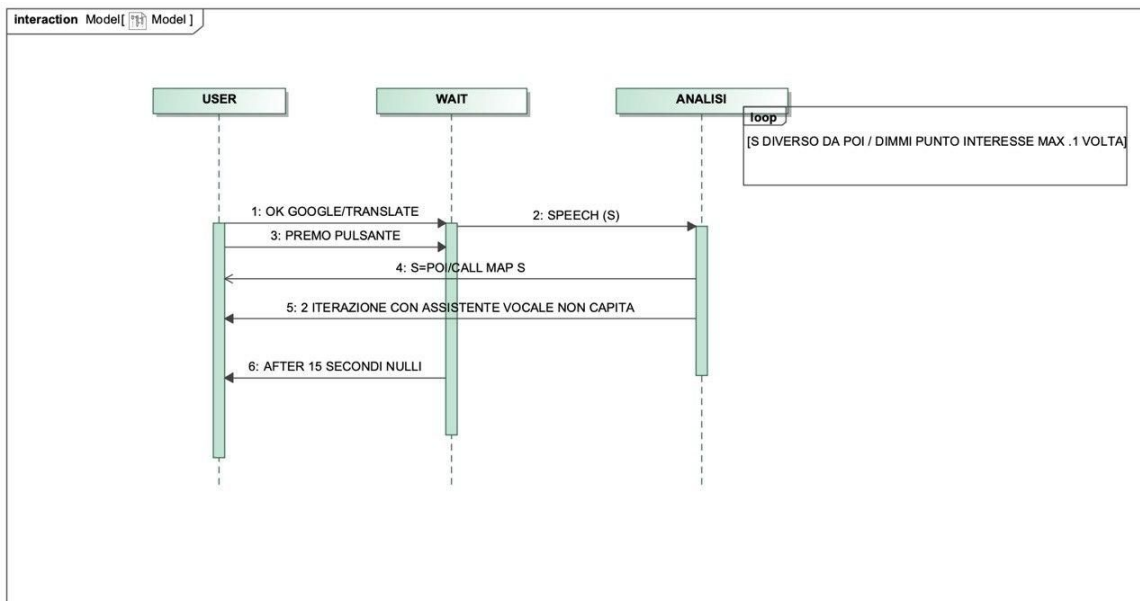
C.2 The dynamic view of the software architecture: Sequence Diagram



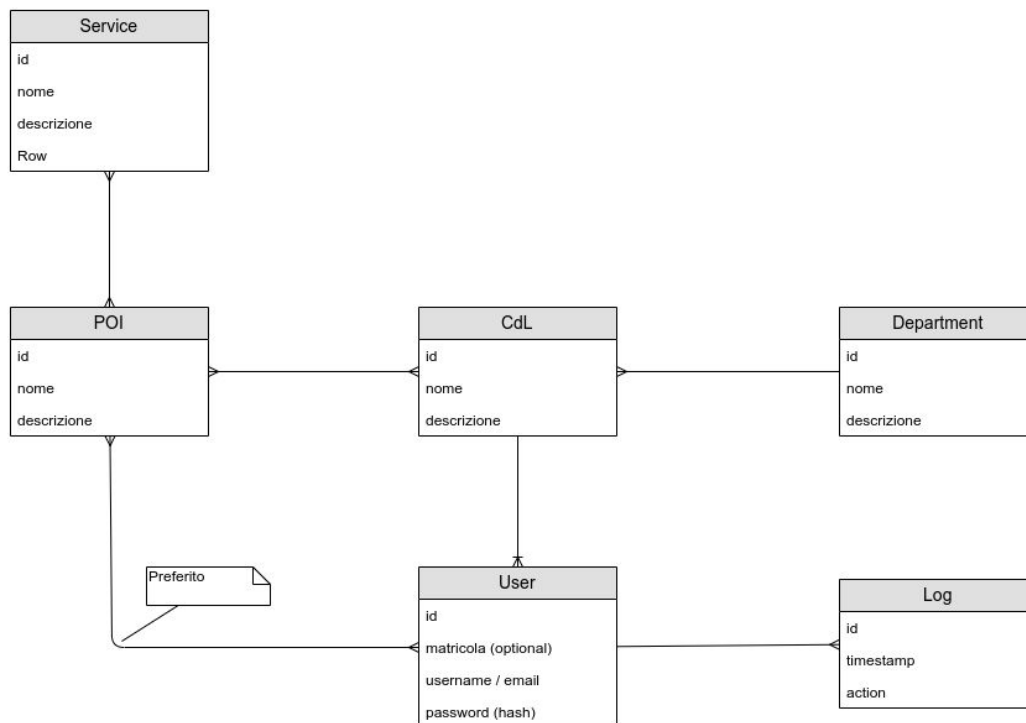
Ricerca di POI tramite input vocale: l'utente richiede al sintetizzatore vocale di convertire la voce in testo, il quale sarà utilizzato come query di ricerca sulla mappa.



Navigazione con comandi vocali: quando l’utente inizia la navigazione verso il POI desiderato, il navigatore oltre a fornire le indicazioni sullo schermo le “legge” con l’ausilio del sintetizzatore vocale. In particolare cicla su tutte le istruzioni per convertire le stringhe di testo in audio da poter riprodurre.



D. Dati e loro modellazione



E. Design Decisions

Per quanto riguarda il Design Decision, avevamo pensato di proporre:

1. Un'interfaccia amichevole, giovanile e con caratteri ben leggibili, così da poter essere utilizzata da tutti gli utenti senza fatica. Abbiamo Pensato di Posizionare all'apertura dell'applicazione una schermata di Login che presenta 3 opzioni (Login By Facebook, Login By Google, Login By Esse3)
2. Nella fase della registrazione/login saranno richiesti pochi elementi in quanto deve essere un'applicazione di utilizzo veloce, cliccando nelle login dei social network si verrà reindirizzati alle classiche schermate di login.
3. La schermata principale (dopo l'accesso) presenterà il logo dell'università nel momento in cui si sta caricando la mappa. (BUFFERING)
4. Una volta caricata la Mappa L'app presenterà subito la mappa con i POI
5. A sinistra della mappa sarà presente un menù ad hamburger che permette la disconnessione dell'account, la ricerca dei POI e la possibilità di inoltrare una richiesta di assistenza (presente anche nella schermata di Login)


```

classDiagram
    package Model
    class VocalSynthesizer {
        +SpeechToText( speech : Audio ) : String
        +TextToSpeech( text : String ) : Audio
    }
    class Map {
        +search( query : String ) : void
        +zoomIn( value : Integer ) : void
        +prioritize() : void
        +addFavourite( user_id : int, poi : POI ) : Boolean
    }
    class Navigator {
        +navigate( start : coordinates, end : coordinates )
    }
    class POI {
        +name : String
        -coordinates : coordinates
    }
    class Service {
        +name : String
    }
    class coordinates {
        +coordinateX : Real
        +coordinateY : Real
    }
    class Managing {
        +addPOI( poi : POI )
        +addService( service : Service )
    }
    class User {
        +login()
        +signup()
    }
    class Registered {
        +id : int
        -username : String
    }
    class Student {
        +CdL
    }
    class Admin {
    }
    VocalSynthesizer ..> Map : «use»
    Map ..> Navigator : «use»
    Map ..> POI : «use»
    Map ..> Service : «use»
    Map ..> Managing : «use»
    Map ..> User : «use»
    Map ..> Registered : «use»
    Map ..> Student : «use»
    Map ..> Admin : «use»
    Navigator ..> coordinates : «use»
    POI --> Service : offre 0..*
    Managing --> POI : «use»
    Managing --> Service : «use»
    Managing --> User : «use»
    Managing --> Registered : «use»
    Managing --> Student : «use»
    Managing --> Admin : «use»
    User --> Registered : «use»
    Registered --> Student : «use»
    Student --> Admin : «use»
    
```

Un utente generico (non registrato) è in grado di accedere alla mappa e ad alcune sue funzionalità. Alcune di queste funzionalità sono gestite da una classe separata, come il navigatore o il sintetizzatore vocale (per permettere all'utente di interagire tramite comandi vocali). Uno studente potrà inoltre prioritizzare i POI in base al suo CdL e aggiungere un POI ai suoi preferiti.

17

G. Explain how the FRs and the NFRs are satisfied by design

QUALITÀ DEL PRODOTTO SISTEMA/SOFTWARE

- Efficienza nella prestazioni

Per quanto riguarda il time behaviour, deve essere breve in quanto lo studente deve ricevere una risposta in poco tempo da parte del sistema. Quindi la richiesta dell'utente viene mandata al sistema, che deve elaborare la richiesta in pochi secondi, ed inviare la risposta sul display.

Il database e il sistema devono essere in grado di gestire una quantità abbastanza elevata di dati nel più breve tempo possibile.

- Usabilità

Il progetto presenta un'interfaccia basilare e semplice così da poter essere utilizzata da tutti gli studenti.

E' veloce, in quanto vengono salvate nel database le preferenze e gli storici dei singoli utilizzatori.

- Affidabilità

E' la capacità di un sistema o componente di eseguire le funzioni richieste in base a quanto indicato per un periodo di tempo specificato, per questo, il progetto presenta un tempo medio accettabile per guasti e la capacità di rilevare guasti specifici o di resistere ad 'attacchi' esterni.

- Sicurezza

Non è uno dei punti fondamentali e forti di questo progetto, in quanto non vengono trattati dati sensibili, ma solo dati anagrafici che vengono 'protetti' in base alle norme presenti sulla privacy.

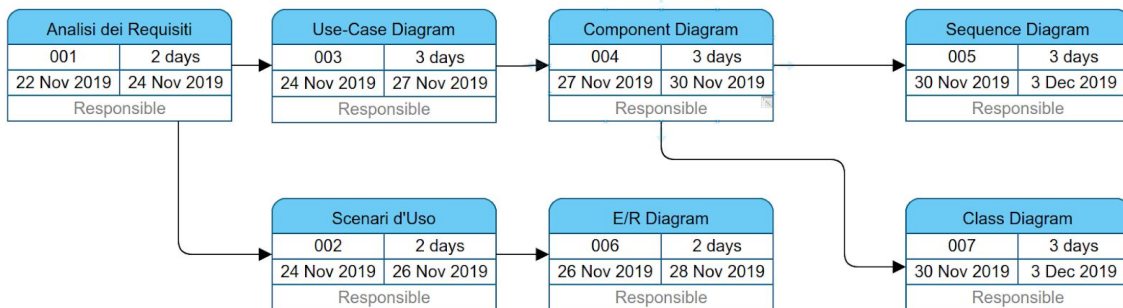
- Portabilità

Presenta un'alta portabilità, in quanto può essere usata su diverse piattaforme come ad esempio IOS e Android.

QUALITA' IN USO

- Efficacia ed efficienza
Vogliamo avere il massimo risultato con il minimo sforzo, per questo vogliamo offrire un servizio che sia di gradimento all'utilizzatore finale. Deve avere quantità e qualità.
- Soddisfazione
Altro punto fondamentale, lo studente deve essere soddisfatto del servizio offerto, deve sentirsi sicuro nel immettere i propri dati e fornire la sua posizione. Per l'utente deve poter essere un piacere utilizzare il sistema e non essere un peso.
- Libero da rischi
L'utilizzatore dell'app, non va incontro a dei particolari rischi, ad esempio economici in quanto non inserisce dati relativi ai propri conti in bancari, di salute psico-fisica in quanto non sono previsti pop-up o pubblicità che possono nuocere ad uno studente più sensibile.
- Copertura del contesto
Il progetto, copre unicamente il contesto universitario, quindi come già detto si può accedere all'app solamente nelle ore in cui l'ateneo è aperto al pubblico.

PERT



Logging

[illegible]

Personal Journal						
Team (number and name): team banana						
Student name: Gabriele Tagliente						
Student number: 247295						
Email: gabriele.tagliente@student.univaq.it						
When (Month/Day)	Time spent	Partners (please report how many people have been working)	Brief Description of the performed task	Category	Sub-Category	
11 22	1:00	1	sviluppo del grafico PERT	Doing	Effort Recording	
11 22	1:30	2: Giorgio	elenco dei requisiti funzionali e non funzionali	Doing	Analisi dei Requisiti	
11 25	2:00	1	Modellazione e UML	Learning	Modeling	
11 26	1:00	1	ricerca e studio di applicativi esistenti	Doing	Stato dell'Arte	
11 26	1:30	2: Valentina	desrizione degli scenari d'uso	Doing	Scenari D'Uso	
11 27	3:00	1	Use-Case, Component e Class Diagrams	Learning	Diagrammi	
11 28	2:30	2: Giorgio	sviluppo Component Diagram	Doing	Component Diagram	
11 1	2:30	1	sviluppo Class Diagram	Doing	Design di Basso Livello	

Personal Journal						
Team (number and name): team banana						
Student name: Valentina Sborlini						
Student number: 247975						
Email: valentina.sborlini@student.univaq.it						
When (Month/Day)	Time spent	Partners (please report how many people have been working)	Brief Description of the performed task	Category	Sub-Category	
11 23	1:10		Modellazione casi d'uso	learning	Use Case Diagram	
11 26	1:30	Tagliente	Scenari d'uso dettagliati	doing	Scenari D'Uso	
11 26	1:15		Descrizione principali assunzioni	doing		
11 27	2:00		Modellazione class diagram	learning	Class Diagram	
11 28	1:40		Dati e loro modellazione	doing	ER Diagram	
11 29	2:00		Design Decisions	doing		
11 30	1:30	Innamorati	Design di Basso Livello	doing	Class Diagram	
12 3	1:00		Spiegazione di come FRs e NFRs sono stati soddisfatti	doing	FR e NFR soddisfatti	

Summary Statistics

Total time spent: 43h 20m
 Total time Learning: 13h 55m
 Total time Doing: 29h 25m

~~Appendix. Prototype~~

<Provide a brief report on your prototype, and especially: information on what you have implemented, how the implementation covers the FR and NFR, how the prototypes demonstrates your project correctness with respect to the FR and NFR. You may add some screenshots to describe what required above. Be ready to show your prototype during the oral examination>
