**Youth Club**

**Requirement Analysis**

**Document  
Versione 2.0**

****

***DATA: 13/01/2019***

**Coordinatore del progetto:**

|  |  |
| --- | --- |
| Nome | Matricola |
| Francesco Truono | 0512104830 |

**Partecipanti:**

|  |  |
| --- | --- |
| Nome | Matricola |
| Vincenzo Liguorino | 0512104692 |
| Aniello Petrosino | 0512104680 |
| Cristiana Elena Lazar | 0512104662 |

|  |  |
| --- | --- |
| **Scritto da:** | Tutto il team |

**Revision History**

|  |  |  |  |
| --- | --- | --- | --- |
| Data | Versione | Descrizione | Autore |
| 27/12/2018 | 0.1 | Prima stesura | \* |
| 27/12/2018 | 0.2 | Revisione scenari | Liguorino Vincenzo |
| 10/01/2019 | 0.3 | Revisione conclusive Entity/Control/Object | Truono Francesco  Lazar Cristiana Elena |
| 11/01/2019 | 1.0 | Aggiunta Class Diagram e entità Geocode | Truono Francesco |
| 11/01/2019 | 1.1 | Aggiunta State Chart | Petrosino Aniello |
| 13/01/2019 | 1.2 | Aggiunta sequence diagram | Liguorino Vincenzo |
| 13/01/2019 | 2.0 | Stesura finale del documento | \* |

Sommario

[**1.** **Introduzione** 4](#_Toc536013097)

[1.1. Proposta del sistema 4](#_Toc536013098)

[1.2. Scopo del sistema 4](#_Toc536013099)

[1.3. Ambito del sistema 4](#_Toc536013100)

[**2.** **Sistema proposto** 5](#_Toc536013101)

[2.1. Overview 5](#_Toc536013102)

[2.2. Requisiti funzionali 5](#_Toc536013103)

[2.3. Requisiti non funzionali 5](#_Toc536013104)

[2.3.1. High Accurancy 5](#_Toc536013105)

[2.3.2. Adaptability 6](#_Toc536013106)

[2.3.3. Usability 6](#_Toc536013107)

[2.3.4. Reliability 6](#_Toc536013108)

[2.3.5. Performance 6](#_Toc536013109)

[2.3.6. Supportability 6](#_Toc536013110)

[2.4. System model 6](#_Toc536013111)

[2.4.1. Scenari 6](#_Toc536013112)

[2.4.2. Use case model 9](#_Toc536013113)

[2.4.3. Use case diagram 14](#_Toc536013114)

[2.4.4. Object model 15](#_Toc536013115)

[2.4.5. Class Diagram 16](#_Toc536013116)

[2.4.6. Sequence diagram 17](#_Toc536013117)

[2.4.7. Statechart diagram 20](#_Toc536013118)

[2.4.8. Mock-up 20](#_Toc536013119)

# **Introduzione**

## Proposta del sistema

Il sistema è stato progettato per dare supporto a utenti in cerca di qualche locale particolare o per utenti che semplicemente non conoscono la zona, inoltre permette anche di vedere la votazione del locale e le recensioni fatte da altri utenti. La ricerca del locale viene divisa in quattro tipologie: Bar, Pub, Enoteche e Discoteche in modo da facilitare ancora di più la ricerca del cliente.

A differenza degli altri competitors il nostro sistema permette anche la modifica della propria recensione.

## Scopo del sistema

Le funzionalità da sviluppare sono:

* Ricerca del locale:
  + Per nome
  + Per luogo
  + Per GPS
* Recensioni:
  + Inserimento
  + Modifica

## Ambito del sistema

Il sistema che andremo a realizzare sarà indirizzato ad un’utenza formata in particolare da giovani utenti, che, grazie all’ausilio di un’app mobile, potranno ricercare in maniera semplice locali nella zona in cui si trovano in quel momento oppure in uno specifico luogo di interesse.

Le operazioni e funzionalità generali proposte dal sistema sono le seguenti:

* Ricerca del locale per nome, luogo e tramite l’utilizzo del GPS
* Modifica e Inserimento di una recensione

# **Sistema proposto**

## Overview

Il Requirement Analysis Document si concentra su:

* + Requisiti funzionali
  + Requisiti non funzionali
  + Scenari
  + Use case model
  + Use case diagram
  + Object model
  + Class diagram
  + Sequence diagram
  + Statechart diagram

## Requisiti funzionali

**Utente**

**FR1:** l’utente deve poter usufruire del servizio di ricerca dei locali

**FR2:** l’utente deve poter usufruire del servizio di ricerca delle recensioni dei locali

**FR3:** l’utente deve poter effettuare una recensione su uno o più locali

**FR4:** l’utente deve poter modificare la propria recensione

**Servizi esterni**

**FR1:** Location Management: deve poter sfruttare il servizio GPS del dispositivo

**FR2:** YELP API: deve accertare la posizione e l’elenco dei locali

**FR3:** FourSquare: deve permettere la ricerca dei luoghi d’interesse

**FR4:** Google Place: deve permettere la ricerca dei luoghi d’interesse e visualizzare i dettagli riguardante i locali

**FR5:** Google Maps: deve contrassegnare i posti ricercati sull’applicativo

## Requisiti non funzionali

### High Accurancy

L’applicativo fornisce un’alta affidabilità della risposta data dal confronto dei risultati elaborati da YELP API, FourSquare e Google Place.

### Adaptability

Il software è in grado di memorizzare i risultati delle ricerche dei vari utenti riducendo in tal modo i tempi di risposta per le successive ricerche.

### Usability

Gli utenti per utilizzare l’applicativo possono registrarsi senza conoscenza preliminare del funzionamento di quest’ultimo.

### Reliability

Quando un’azione viene interrotta a causa di un crash, il sistema dovrebbe essere in grado di riportare l’utente all’azione precedente al crash. Al massimo una ricerca potrebbe essere persa.

### Performance

Il sistema deve essere in grado di parallelizzare le ricerche su tre diversi sistemi e memorizzarle per apprendere dalle precedenti ricerche degli utenti.

L’applicazione deve avere tempi di risposta brevi.

### Supportability

Il sistema deve essere manutenibile e adattabile.

## System model

### Scenari

Scenario di ricerca locale per nome

Vincenzo sta cercando un locale e decide di aprire l’applicazione. L’applicazione una volta aperta mostra a Vincenzo una schermata dove può selezionare la ricerca per nome e per località, sceglie quella per nome. Una volta selezionata la ricerca per nome, l’applicazione mostra una schermata che contiene un’area di testo per inserire il nome del locale e un pulsante per cercare. Vincenzo inserisce nell’area di testo “Panta Rei” e subito dopo clicca sul pulsante di ricerca. Una volta che Vincenzo ha premuto il tasto di ricerca, l’applicazione mostra lo schermo diviso in due, dove nella parte superiore mostra la posizione del locale con l’ausilio di un marker e nella parte inferiore il locale cercato, eventualmente se sono presenti più locali con lo stesso nome mostra la lista e la rispettiva posizione sulla mappa. Vincenzo analizza l’intera lista e seleziona il locale che dalla mappa è situato all’uscita dell’autostrada di Fisciano, perché quello più vicino a lui; una volta individuato il locale Vincenzo ci clicca sopra. Dopo aver cliccato vengono mostrate in un’altra schermata le informazioni relative a quel locale come ad esempio numero di telefono, sito web, orari di apertura e chiusura, una foto, una breve descrizione e l’elenco delle recensioni fatte dai clienti precedenti. Il locale selezionato presenta recensioni che convincono Vincenzo allora decide di andarci.

Scenario Ricerca Locale per località con l’uso del GPS

Vincenzo sta cercando un locale e decide di aprire l’applicazione. L’applicazione appena aperta mostra a Vincenzo una schermata dove può selezione la ricerca per località o nome e sceglie quella per località. Una volta che Vincenzo ha selezionato la ricerca per località, l’applicazione gli mostra una schermata dove trova una barra di ricerca, un pulsante che permette all’utente di geo localizzarsi e al di sotto di essa i checkBox relativi alle categorie di locali che intende trovare nelle vicinanze. Dopo aver cliccato sul pulsante di geo localizzazione sulla barra di ricerca comparirà latitudine e longitudine effettivi. Vincenzo seleziona le categorie di suo interesse e clicca sul bottone di ricerca al di sotto dei checkBox che lo trasferisce alla schermata successiva. L’applicazione nella seguente pagina mostra a Vincenzo una schermata divisa in due parti: la parte superiore occupata dalla mappa sulla quale vengono visualizzati i marker dei locali situati intorno a Vincenzo, mentre nella parte inferiore compare una lista dei locali ordinati in base alla loro valutazione. Vincenzo analizza l’intera lista e trovandosi a Fisciano seleziona il bar “Panta Rei” che si trova a 600 metri dalla sua posizione attuale, cliccandoci sopra. Dopo aver cliccato vengono mostrate in un’altra schermata le informazioni relative a quel locale come ad esempio numero di telefono, sito web, orari di apertura e chiusura, una foto, una breve descrizione e l’elenco delle recensioni fatte dai clienti precedenti. Il locale selezionato presenta recensioni che convincono Vincenzo allora decide di andarci**.**

Scenario Ricerca Locale per località con inserimento del luogo specifico

Vincenzo sta cercando un locale e decide di aprire l’applicazione. L’applicazione appena aperta mostra a Vincenzo una schermata dove può selezione la ricerca per località o nome e sceglie quella per località. Una volta che Vincenzo ha selezionato la ricerca per località, l’applicazione gli mostrerà una schermata dove trova una barra di ricerca, in cui inserire il testo, un pulsante che permette all’utente di geo localizzarsi e al di sotto di essa i checkBox relativi alle categorie di locali che intende trovare. Dopo aver scritto il luogo che potrà essere una strada, una città o un comune Vincenzo seleziona le categorie di suo interesse e clicca sul bottone di ricerca al di sotto dei checkBox che lo trasferirà alla schermata successiva. Vincenzo decide di cercare i bar che si trovano a Fisciano. L’applicazione nella seguente pagina mostra a Vincenzo una schermata divisa in due parti: la parte superiore occupata dalla mappa sulla quale verranno visualizzati i marker dei locali che si trovano nella località selezionata che nel suo caso è Fisciano, mentre nella parte inferiore compare una lista dei locali ordinati in base alla loro valutazione. Vincenzo analizza l’intera lista e seleziona il bar “Panta Rei” cliccandoci sopra. Dopo aver cliccato vengono mostrate in un’altra schermata le informazioni relative a quel locale come ad esempio numero di telefono, sito web, orari di apertura e chiusura, una foto, una breve descrizione e l’elenco delle recensioni fatte dai clienti precedenti. Il locale selezionato presenta recensioni che convincono Vincenzo allora decide di andarci

Scenario inserimento recensioni

Vincenzo dopo aver effettuato una ricerca utilizzando une delle tecniche descritte in precedenza, seleziona il locale che vuole recensire. Dopo aver selezionato il locale, l’applicazione mostra in un’altra schermata le informazioni relative ad esso come via, numero di telefono, sito web, orari di apertura e chiusura (se disponibili), una foto, l’elenco delle recensioni fatte dai clienti precedenti e un pulsante per inserire una recensione. Vincenzo dopo aver cliccato il pulsante viene trasferito su un’altra schermata dove comparirà un’apposita form compilabile dove inserirà il titolo e il testo della recensione negli appositi campi, un giudizio complessivo del locale espresso numericamente con valori che vanno da 1 (pessimo) a 5 (ottimo), e una valutazione del servizio, qualità/prezzo e cibo. Una volta aver completato la recensione Vincenzo clicca su un bottone situato al di sotto della form che invierà la recensione al sistema, che l’accetterà e la mostrerà insieme a quelle già presenti.

Scenario modifica recensione

Vincenzo apre l’applicazione e accede alla sua area utente. Dopo che si è posizionato sulla sua area utente l’applicazione mostra tutte le informazioni relative a Vincenzo ed una lista di tutte le recensioni già effettuate in precedenza. Vincenzo poiché nella stessa settimana è andato due volte nello stesso locale ha ricevuto due servizi diversi e decide quindi di modificare la recensione che aveva inserito precedentemente. Vincenzo allora clicca sul pulsante di modifica e viene trasferito sulla schermata relativa all’inserimento delle recensioni dove potrà modifica i parametri che lui ritiene migliorati o peggiorati. Una volta aver completato la modifica della recensione Vincenzo clicca su un bottone situato al di sotto della form che invierà la recensione al sistema, che l’accetterà e la mostrerà cambiata sul suo profilo

### Use case model

|  |  |
| --- | --- |
| **Nome Caso d’uso** | Seleziona tipo di ricerca |
| **Attori** | Utente |
| **Flusso di eventi** | * L’utente apre l’applicazione * All’utente si presenta una pagina contenente dei pulsanti che permettono di selezionare il tipo di ricerca che vuole effettuare * L’utente seleziona il tipo di ricerca |
| **Entry Condition** | * L’utente avvia l’applicazione |
| **Exit Conditions** | * L’utente ha selezionato il tipo di ricerca |

|  |  |
| --- | --- |
| **Nome Caso d’uso** | Stesura recensione |
| **Attori** | Utente |
| **Flusso di eventi** | 1. Il sistema mostra all’utente oltre alle informazioni del locale la possibilità di scrivere una recensione; 2. L’utente tramite la schermata di recensione compila la form con titolo e testo della recensione;    * Inserimento valutazione locale (1 pessimo, 5 ottimo);    * Valutazione qualità del servizio;    * Valutazione qualità del cibo/bevanda; 3. Il sistema valida la recensione e si accerta che non ci sono parole offensive\*; |
| **Entry Condition** | * L’utente ha selezionato un locale |
| **Exit Conditions** | * Il sistema accetta la recensione |
| **Exception** | * Nel punto 3 del flusso di eventi il sistema controlla se ogni campo della form è compilato correttamente, nel caso in cui questo non accade mostra all’utente i campi sbagliati marcandoli in rosso. |

|  |  |
| --- | --- |
| **Nome Caso d’uso** | Ricerca per nome |
| **Attori** | Utente |
| **Flusso di eventi** | 1. L’utente scrive il nome del locale nell’apposita area di testo 2. L’applicazione verifica se è stato rispettato il formato: *[Nome locale, Città]* 3. Il sistema verifica l’esistenza della città sul database. 4. Recupera la lista dei locali associati a quel nome 5. Restituisce il locale (potrebbero essere presenti più locali con lo stesso nome e si trovano nella stessa città |
| **Entry Condition** | * L’utente ha selezionato il tipo di ricerca per nome |
| **Exit Conditions** | * Il sistema ritorna il risultato della ricerca |
| **Eccezione** | * Nel punto 3 in caso di riscontro negativo, esegue caso d’uso “Apprendimento” |

|  |  |
| --- | --- |
| **Nome Caso d’uso** | Ricerca per posizione |
| **Attori** | Utente, GPS |
| **Flusso di eventi** | 1. L’utente chiede la posizione al Gps che gli ritorna la latitudine e longitudine, che verranno scritte nell’apposita area di testo. 1.1 L’utente seleziona una o più categorie: Enoteche, Discoteche, Bar o Pub 2. Il sistema verifica se il luogo è già presente sul database 3. Restituisce la lista dei locali in quella zona |
| **Entry Condition** | * L’utente ha selezionato il tipo di ricerca per posizione |
| **Exit Conditions** | * Il sistema ritorna il risultato della ricerca |
| **Exception** | * Nel punto 1 del flusso di eventi se il GPS è disattivato, il sistema chiede all’utente di attivarlo * Nel punto 2 in caso di riscontro negativo, esegue caso d’uso “Apprendimento” |

|  |  |
| --- | --- |
| **Nome Caso d’uso** | Apprendimento |
| **Attori** | Google API  Foursquare API  Yelp API |
| **Flusso di eventi** | 1. In caso di riscontro negativo, interroga Yelp, Google e Foursquare 2. Esegue il merge dei tre risultati ed esclude i duplicati 3. Salva le informazioni sul database |
| **Entry Condition** | * Il sistema verifica se il luogo è presente nel database |
| **Exit Conditions** | * Aggiornamento del database e restituzione dati |

|  |  |
| --- | --- |
| **Nome Caso d’uso** | Modifica della recensione dalla pagina del locale |
| **Attori** | Utente |
| **Flusso di eventi** | 1. L’utente accede alla pagina del locale utilizzando il caso d’uso di ricerca 2. Il sistema ordina le recensioni in modo tale da mostrare come prima recensione quella dell’utente 3. L’utente chiede al sistema di modificare la recensione desiderata. 4. L’utente apporta le modifiche desiderate. 5. Il sistema valida la recensione e si accerta che non ci siano parole offensive. |
| **Entry Condition** | * L’utente ha selezionato una recensione |
| **Exit Condition** | * Il sistema aggiorna la recensione |
| **Exception** | * il sistema può rifiutare la recensione o mantenere la precedente |

|  |  |
| --- | --- |
| **Nome Caso d’uso** | Modifica della recensione dalla lista delle recensioni |
| **Attori** | Utente |
| **Flusso di eventi** | 1. L’utente accede alla recensione dall’apposita area della lista di recensioni. 2. L’utente chiede al Sistema di modificare la recensione desiderata. 3. L’utente apporta le modifiche desiderate. 4. Il sistema valida la recensione e si accerta che non ci sono parole offensive |
| **Entry Condition** | * L’utente ha selezionato la recensione dalla lista delle recensioni |
| **Exit Condition** | * Il sistema aggiorna la recensione |
| **Exception** | * Il sistema può rifiutare la recensione o mantenere la precedente |

|  |  |
| --- | --- |
| **Nome Caso d’uso** | Ricerca per luogo |
| **Attori** | Utente, GPS |
| **Flusso di eventi** | 1. L’utente inserisce nell’apposita area di testo il luogo interessato. 1.1 L’utente seleziona una o più categorie: Enoteche, Discoteche, Bar o Pub 2. Il sistema verifica se il luogo esiste 3. Restituisce la lista dei locali in quella zona |
| **Entry Condition** | * L’utente ha selezionato il tipo di ricerca per luogo |
| **Exit Conditions** | * Il sistema ritorna il risultato della ricerca |
| **Exception** | * Nel punto 2 in caso di riscontro positivo che il luogo è già stato ricercato, esegue caso d’uso “Verifica luogo (Presente)” * Nel punto 2 in caso di riscontro negativo che il luogo inserito è sbagliato, esegue caso d’uso “Verifica luogo (Sbagliato)” * Nel punto 2 in caso di riscontro positivo che il luogo ancora non è stato ricercato, esegue caso d’uso “Verifica luogo (Non Presente)” |

|  |  |
| --- | --- |
| **Nome Caso d’uso** | Verifica luogo (Presente) |
| **Attori** | Geocoder API |
| **Flusso di eventi** | 1. L’utente in caso di riscontro positivo, interroga Geocoder API 2. Memorizza il luogo |
| **Entry Condition** | * Presa in input di un luogo |
| **Exit Condition** | * Il sistema scrive il luogo sul database |

|  |  |
| --- | --- |
| **Nome Caso d’uso** | Verifica luogo (Sbagliato) |
| **Attori** | Geocoder API |
| **Flusso di eventi** | 1. L’utente in caso di riscontro negativo, interroga Geocoder API 2. Restituisce che il luogo inserito è sbagliato |
| **Entry Condition** | * Presa in input di un luogo |
| **Exit Condition** | * Il sistema restituisce che è scorretto |

|  |  |
| --- | --- |
| **Nome Caso d’uso** | Verifica luogo (Presente) |
| **Attori** | Geocoder API |
| **Flusso di eventi** | 1. L’utente in caso di riscontro positivo, interroga Geocoder API 2. Memorizza il luogo |
| **Entry Condition** | * Presa in input di un luogo |
| **Exit Condition** | * Il sistema scrive il luogo sul database |

|  |  |
| --- | --- |
| **Nome Caso d’uso** | Verifica luogo (Sbagliato) |
| **Attori** | Geocoder API |
| **Flusso di eventi** | 1. L’utente in caso di riscontro negativo, interroga Geocoder API 2. Restituisce che il luogo inserito è sbagliato |
| **Entry Condition** | * Presa in input di un luogo |
| **Exit Condition** | * Il sistema restituisce che è scorretto |

|  |  |
| --- | --- |
| **Nome Caso d’uso** | Verifica luogo (Non Presente) |
| **Attori** | Geocoder API |
| **Flusso di eventi** | 1. L’utente in caso di riscontro negativo, interroga Geocoder API 2. Trasforma il luogo in latitudine e longitudine 3. Salva informazioni sul database |
| **Entry Condition** | * Presa in input di un luogo |
| **Exit Condition** | * Il sistema memorizza i dati |

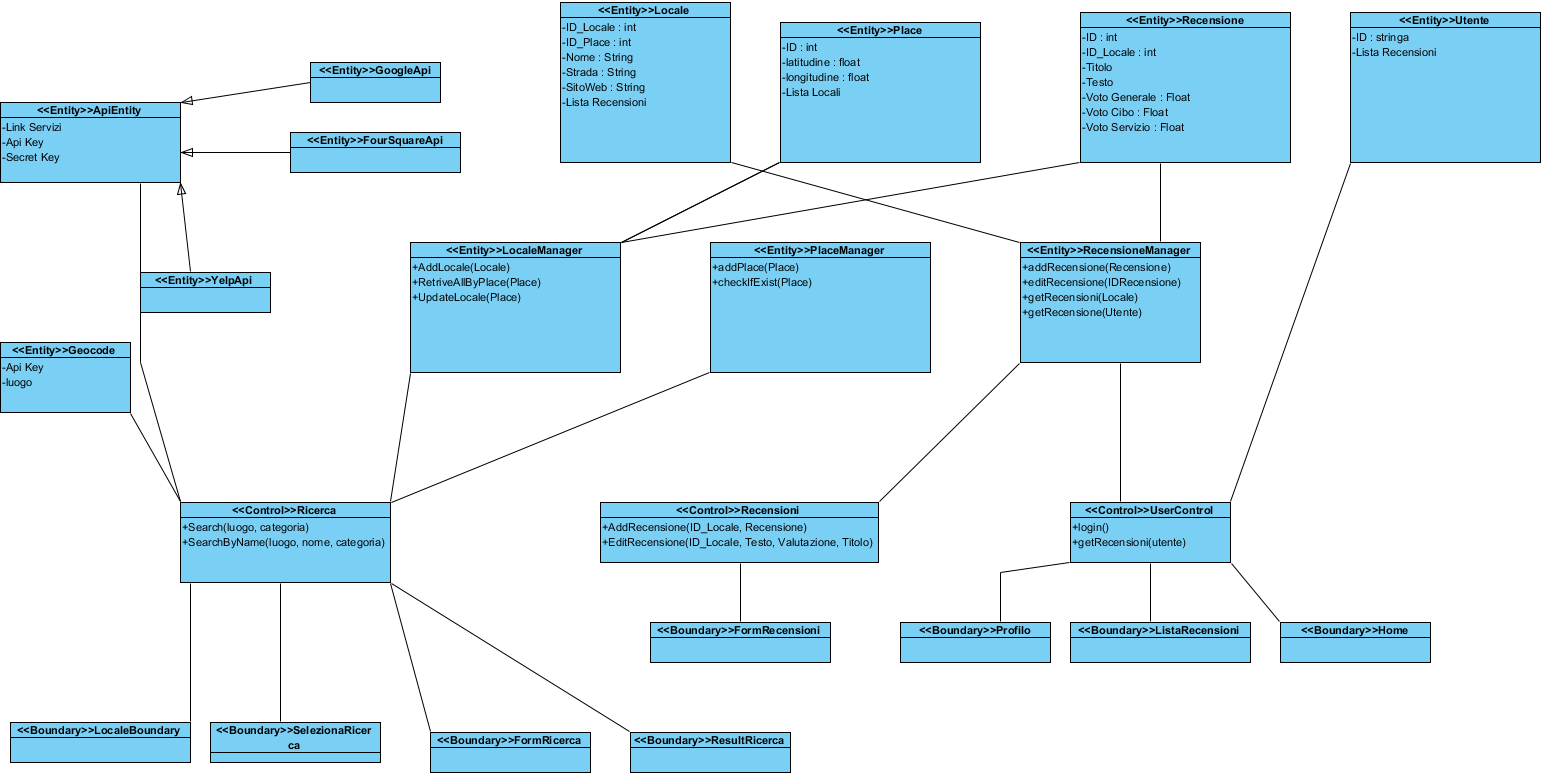
### Use case diagram



### Object model

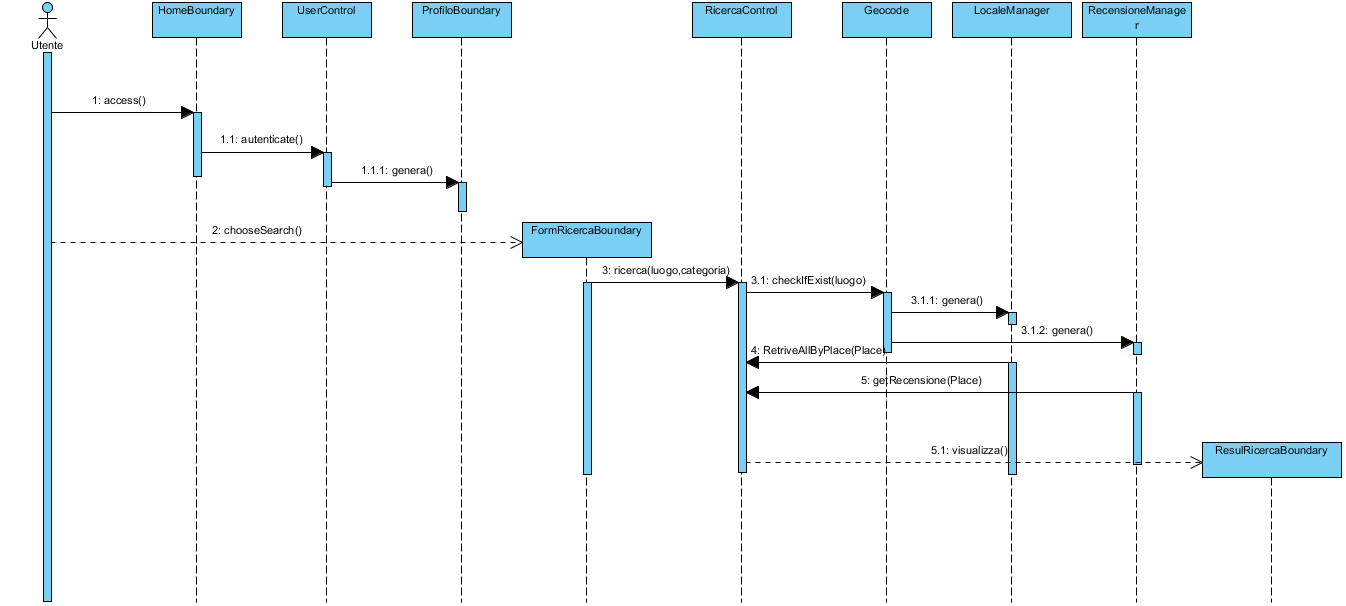
|  |  |  |
| --- | --- | --- |
| **TIPO** | **NOME** | **DESCRIZIONE** |
| Entity object | UtenteEntity  RecensioneEntity  PlaceEntity  LocaleEntity  GoogleApiEntity  YelpApiEntity  FoursquareApiEntity  GeocodeEntity | Rappresenta l’utente  La recensionerappresenta la valutazione espressa da un utente per un determinato locale  Identifica un luogo rappresentato dalle coordinate geografiche e con un raggio di circa 15km  Rappresenta un locale  Google Api è un servizio esterno utile ad ottenere la lista dei locali  Yelp Api è un servizio esterno utile ad ottenere la lista dei locali  Foursquare Api è un servizio esterno utile ad ottenere la lista dei locali  Traduce un luogo, scritto in linguaggio naturale in latitudine e longitudine |
| Boundary | HomeBoundary  SelezionaRicercaBoundary  FormRicercaBoundary  ResultRicercaBoundary  LocaleBoundary  FromRecensioniBoundary  ProfiloBoundary  ListaRecensioniBoundary | Visualizza la schermata iniziale  Visualizza la schermata per la selezione della ricerca  Visualizza il form di ricerca con la presenza o meno di alcuni campi (GPS e categorie)  Visualizza la mappa dei locali e la lista , dopo aver effettuato la ricerca  Visualizza i dettagli del locale  Visualizza il form per scrivere o modificare una recensione  Visualizza il profilo dell’utente  Visualizza la lista delle recensioni fatte dall’utente |
| Control object | UserControl  RicercaControlo  RecensioniControl | Si occupa delle operazioni consentite all’utente  Esegue la ricerca in base alla modalità selezionata dall’utente (Gps, nome, luogo)  Si occupa della gestione delle recensioni |

### Class Diagram

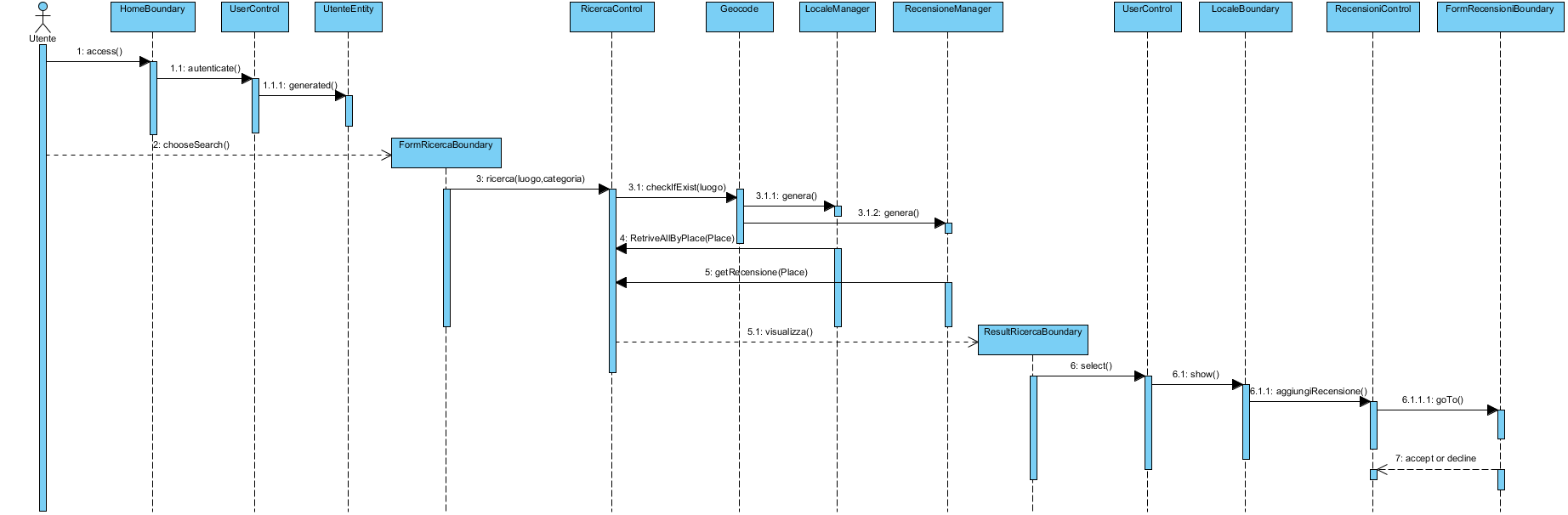


### Sequence diagram

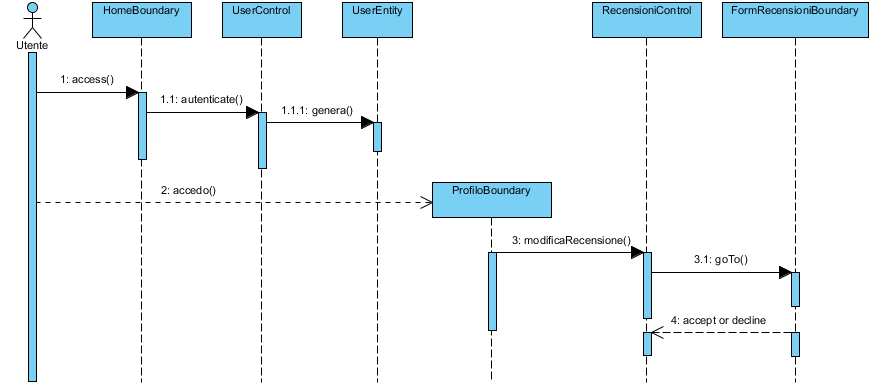
**Sequence diagram: Ricerca**

****

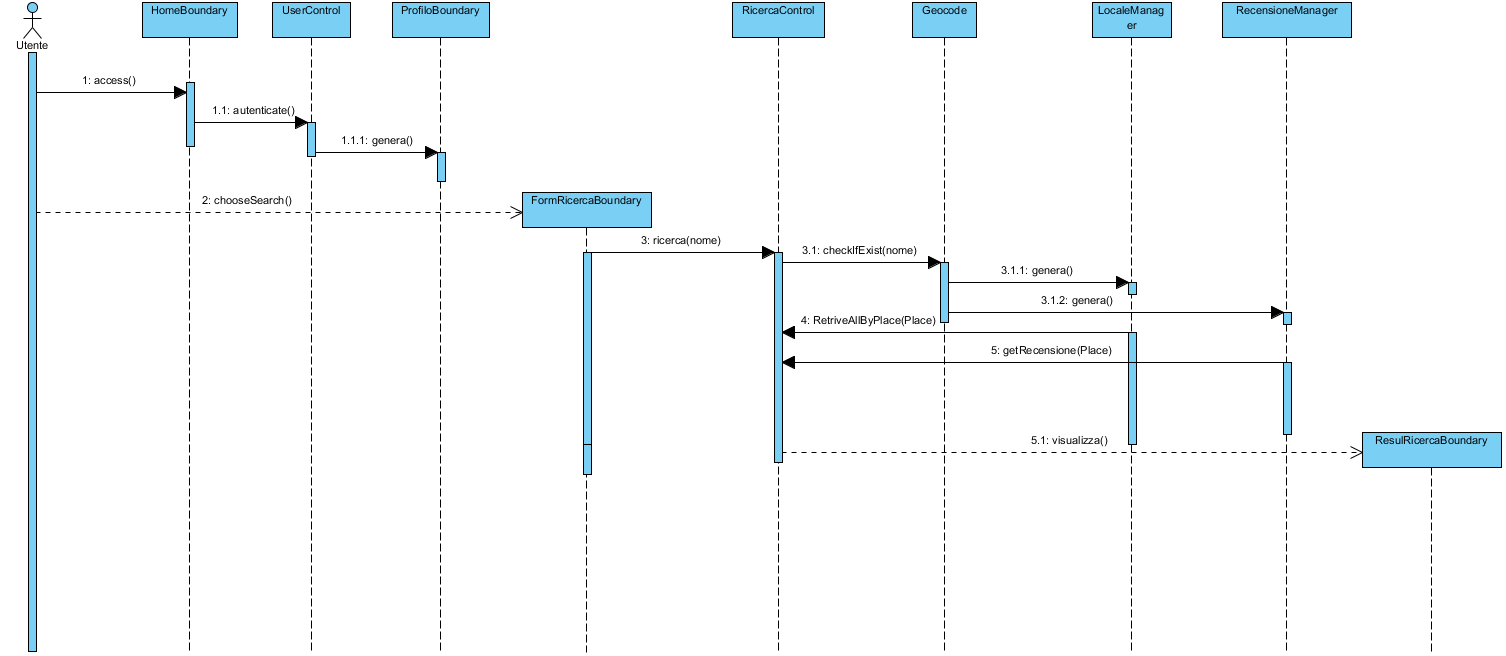
**Sequence diagram: Inserisci recensione**

****

**Sequence diagram: Modifica recensione**

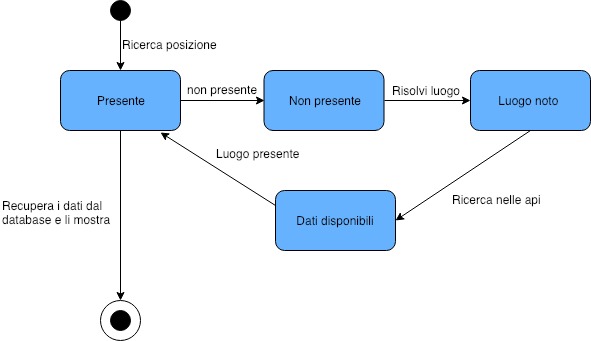
****

**Sequence diagram: Ricerca per nome**

****

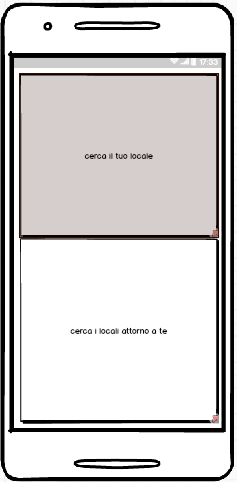
### Statechart diagram

**Statechart diagram: Ricerca**

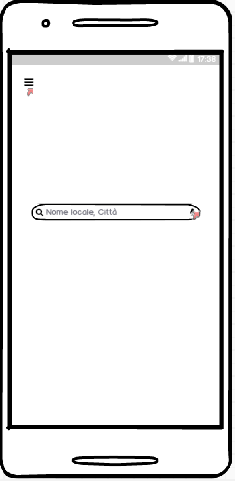
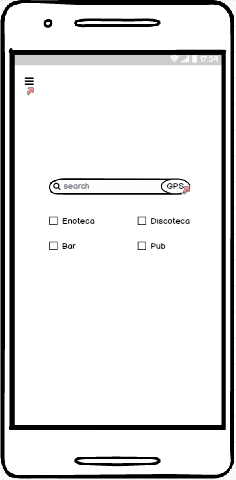
****

### Mock-up

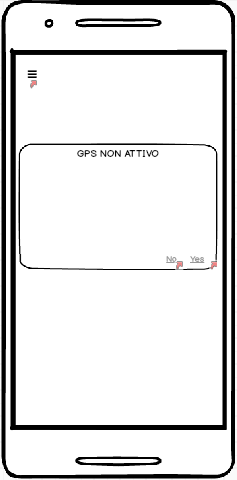
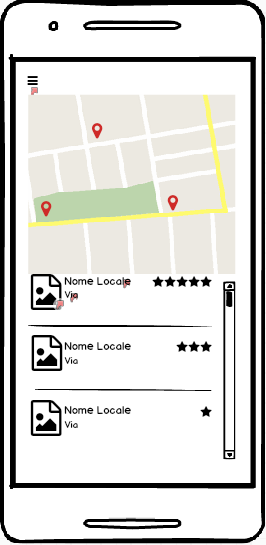
**Visualizzazione Pagina iniziale**

****

**Visualizzazione Ricerca per nome locale Ricerca per posizione**

** **

**Visualizzazione Alert GPS Risultati per ricerca**

** **

**Visualizzazione Profilo Informazioni per locale**

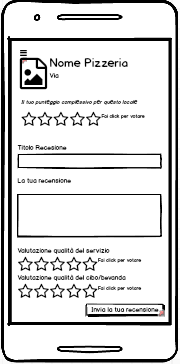
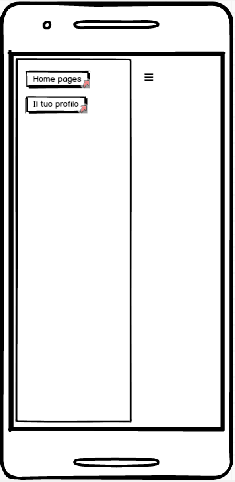
** **

**Visualizzazione Modifica da pagina**

**del locale Modifica dalla lista delle recensioni**

** **

**Visualizzazione Stesura recensione Menù laterale**

** **