TECHNISCHE
UNIVERSITÄT
DARMSTADT

LOCATION AWARE MOBILE APPLICATION

Internet Praktikum TK

Team: XRAY

Abdul Razzak (2932464)

TripBuddy

Table of Contents

1. Introduction

1.1. Overview

Tripbuddy is a location aware mobile application and it uses the capabilities of your smartphone to locate you and then provide the option to “check-in” to places around you based on your location. Tripbuddy lets the user search for restaurants for breakfast, lunch, dinner, tea/coffee and other places of interest in their surrounding area. It is also possible to search other areas by entering the name of a remote location. The user gets details of the searched venue such as address and contact number. The user can check-in to the venue and also leave a tip about his experience at the venue. The application also shows you the venues surrounding your current location as marker on Google maps.

1.2. Purpose

Tripbuddy allows users to quickly find the best places in their current location. Being location-based, Tripbuddy defaults to your current location to send you the best recommendations for coffee, lunch, dinner, etc. If you've just arrived in a new city's main train station, it will point out the best lunch spot next to you. If you enjoy food or drink and spend time in urban areas, Tripbuddy is the best way to explore. It allows you to check-in to a specific place and also rate and leave comments so that other users can review your recommendation about the venue. Also Google maps show the marker location of venues surrounding your current location.
Tripbuddy encourages the writing of "Leave a Tip" - short messages about a location which let other users know what is good (or bad) there.

2. Architecture

2.1. Introduction

Tripbuddy provide functionality to users to quickly find the best places in their current location.

Below are the tasks lists for user role.

- Role: User
  - Register to the system (with email, password)
  - Login (with registered email, password) and logout
  - User's profile management
  - Search venues for breakfast, lunch, dinner, etc from radio button
  - Search other categories using search bar
  - Search nearby venues of current location using Google maps
  - Check-in to a venue
  - Maintain history of all the check-ins in history tab
  - Leave a tip about experience at the venue
  - View other users comments about the their experience at venue

2.2. Components

Components involved are:

- Cloud based MongoDB
- Node JS
- Android studio
- Foursquare (Third party tool)
- Amazon ec2

2.2.1. MongoDB

MongoDB is an open-source document database that provides high performance, high availability, and automatic scaling. A record in MongoDB is a document, which is a data structure composed of field and value pairs. MongoDB documents are similar to JSON objects. The values of fields may include other documents, arrays, and arrays of documents.

A cloud based version of MongoDB also known as a Database as a service (DAAS) is used for persisting the values.

2.2.2. Node.js

Node.js is used for creating the server using its express library. It listens to port 3000 for external communication. Additionally, it exposes several endpoints to serve the get and post requests to the clients. All the persistence of data to the Cloud based MongoDB explained in 2.2.1 is implemented using the Mongoose library. It deals with the persistence and transfer of data back to the node server. Bcrypt library is used to encrypt the secured data like password of the users.

2.2.3. Android Studio

Android client is developed using Android studio that is the official IDE (Integrated Development Environment) for developing application exclusively for Android platform. The screens are configured using the .xml files for defining the layouts and its components. Business logic is implemented using the Java classes consisting of Fragments, Activity and Data Models. Within the client application, the client to server communication is developed using the Retrofit library.

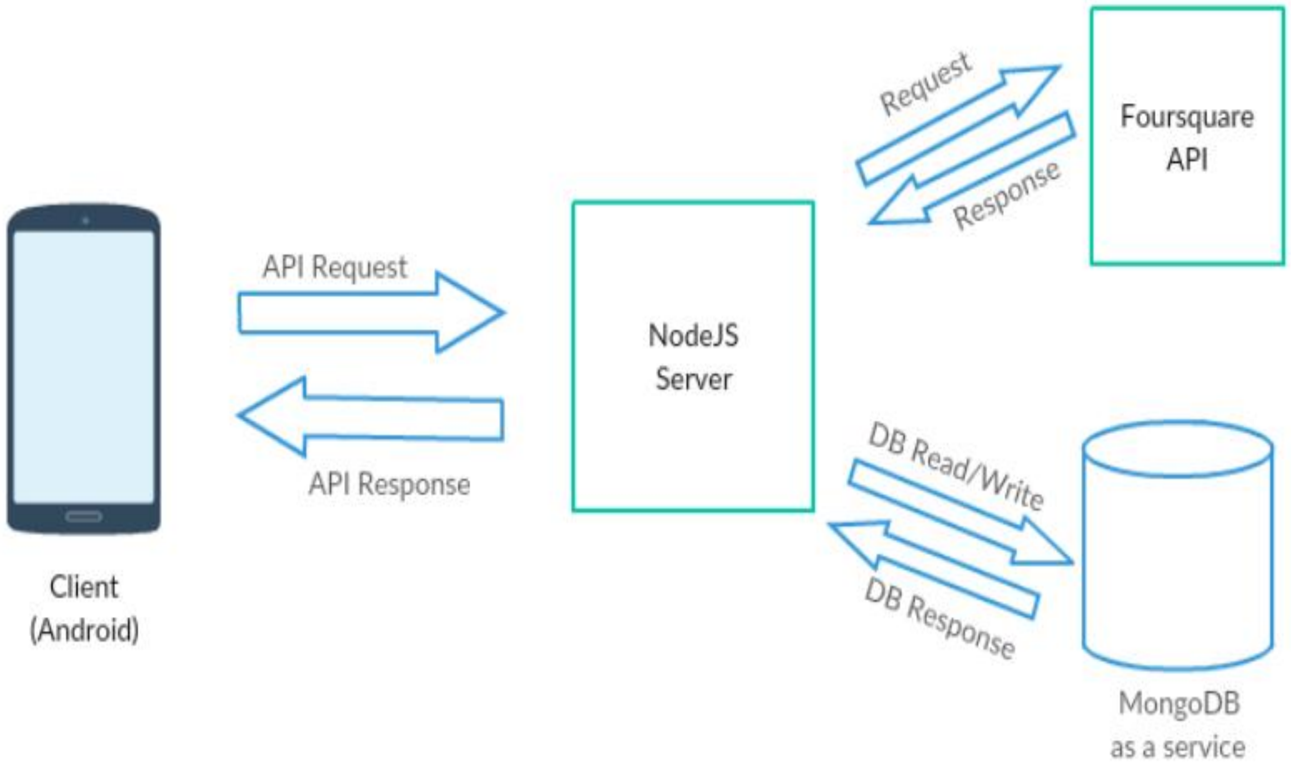
2.2.4. Foursquare API

Foursquare API provides several endpoints to its users to provide the data of the venues, users, comments, images etc. I am using the venues endpoint for requesting the venues around the user. It provides the user with client id and client secret key to the users for accessing its API. If the request made is valid with the right credentials then Foursquare provides the user with an authentication token that can be used by the user for making future requests. In this context, the user means the node server that makes the request to Foursquare API on behalf of the clients.

2.2.5. Amazon EC2

Amazon EC2 is used to host the node server on which the request can be made by external android clients.

2.3. System Architecture

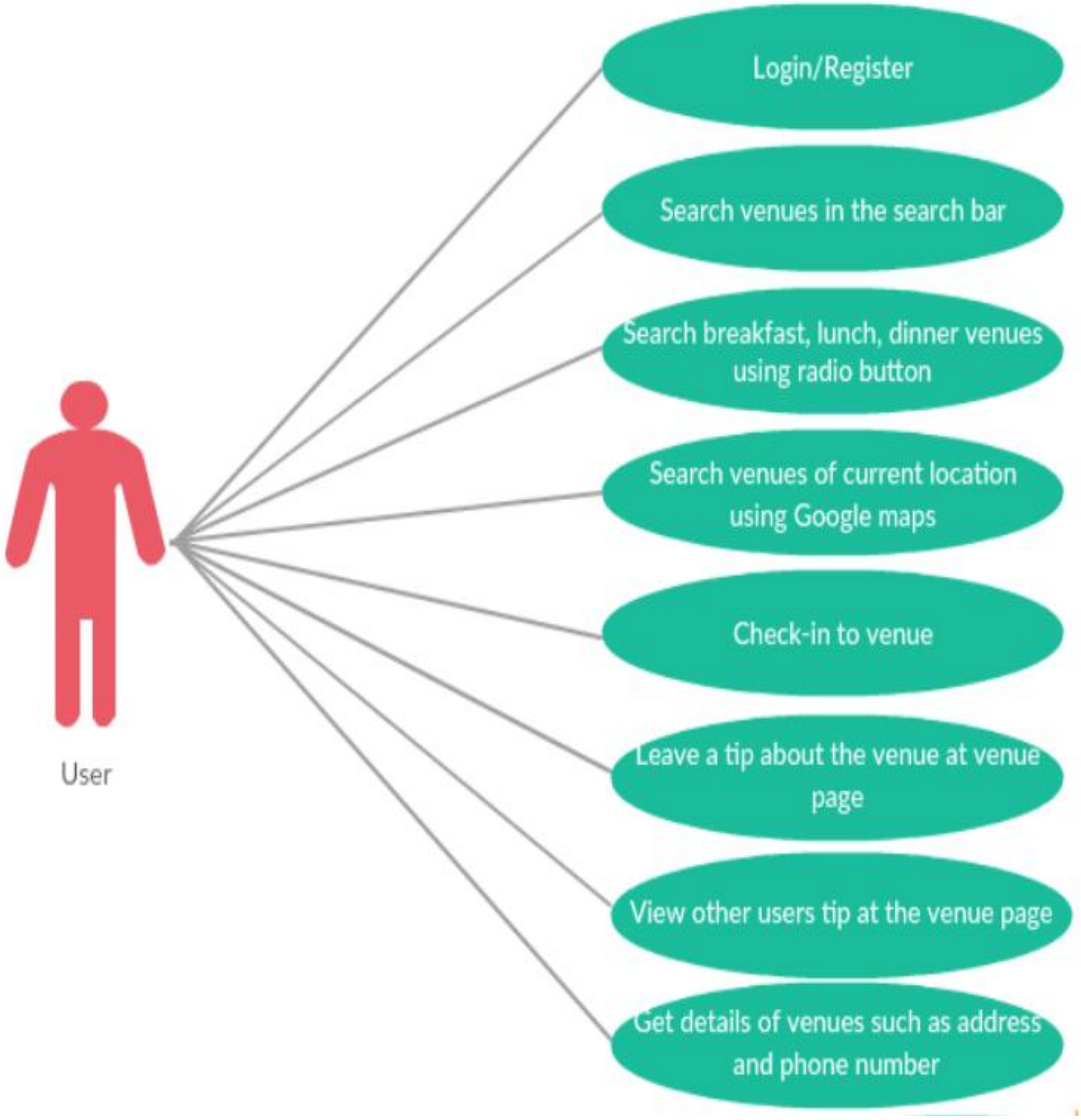


Explanation:

Client makes API call to the endpoints provided by Node.js. Node.js makes call to API behind the scenes to foursquare API to get the venue data based on the user location and the category defined by the user. Foursquare API returns the result to Node.js server. Node server returns back the response to the requesting Android client. Additionally Node.js makes call to MongoDB for fetching local requests such as authentication of user during login/registration, updating the user profile and also comments on venues photos.

2.4 Use Diagram

Below is the description of the roles of user in the application using use diagram


Login/Register
Search venues in the search bar
Search breakfast, lunch, dinner venues using radio button
Search venues of current location using Google maps
Check-in to venue
Leave a tip about the venue at venue page
View other users tip at the venue page
Get details of venues such as address and phone number
User

3. Implementation

3.1 MongoDB Schema

As explained in the previous section, MongoDB is implemented in cloud. We have implemented the following schema definitions for:

User

lastName	firstName	email	Password	username
string	string	string (unique field)	String (asterisk mark)	string

Comments

venue_id	email	Comment	created_at
string	string (unique field)	String	string

checkin_details

venue_id	email	created_at
string	string (unique field)	string

3.2 Foursquare API

The following endpoints are used to fetch data from the Foursquare into the implemented application

Endpoints	Parameters required	Results in Json Format
/	clientId, clientSecretKey, CallbackURL	Authentication token
/venues/search	lat, lng, searchText	Array of Venue Objects consisting of Venue Id, Venue Name, Rating, Location, Address, Phone number, Comments, Images
/venue	venue_id	Returns data of a single venue

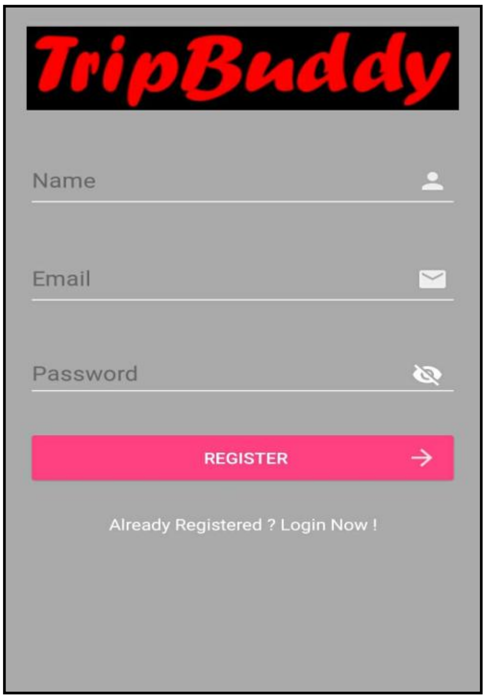
3.3 Node.js Endpoints

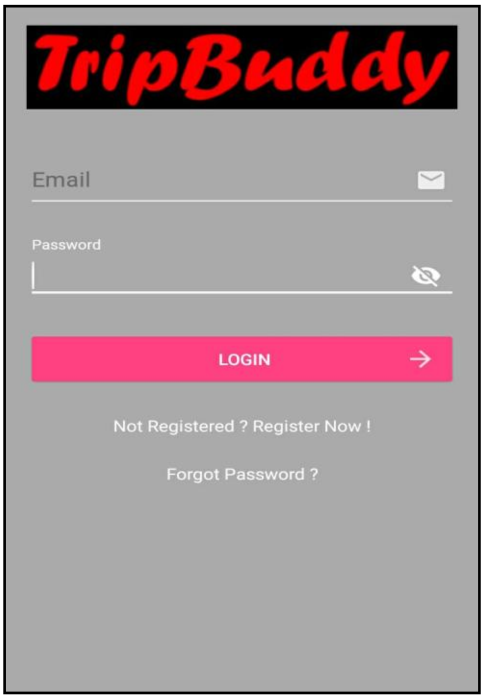
The following endpoints are implemented in Node.js to be requested by the client application.

Feature	Endpoints	Request Type	Parameters	Results in Json format
Registration	/users	post	username, lastName, firstName, email, password	Success or error message
Get profile data	/users	get	id	user data
Change password	/users	put	id	success or error message
Authenticate user	/authenticate	post	username, password	Authentication token
Search venues	/places	get	lat, lng, query	Array of Venue Objects consisting of Venue Id, Venue Name, Location,
Load Venue	/venue	get	venueId	Returns single venue Object
Checkin	/checkinVenue	post	userName, venueId	Returns success or error message
Get checkin Info	/getCheckinInfo	get	userName, venueId	Returns checkin object or error
Add comments	/addComment	post	venueId, email, tips	Returns success or error message
Get comments on venue	/getComments	get	venueId	Returns comments array

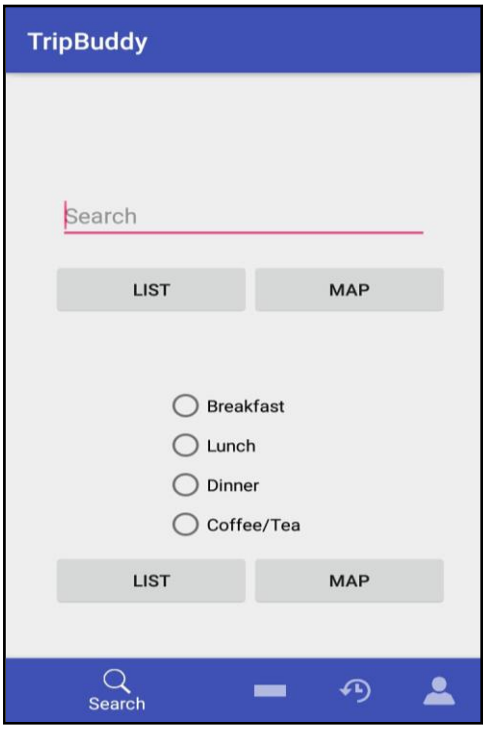
4. Application flow

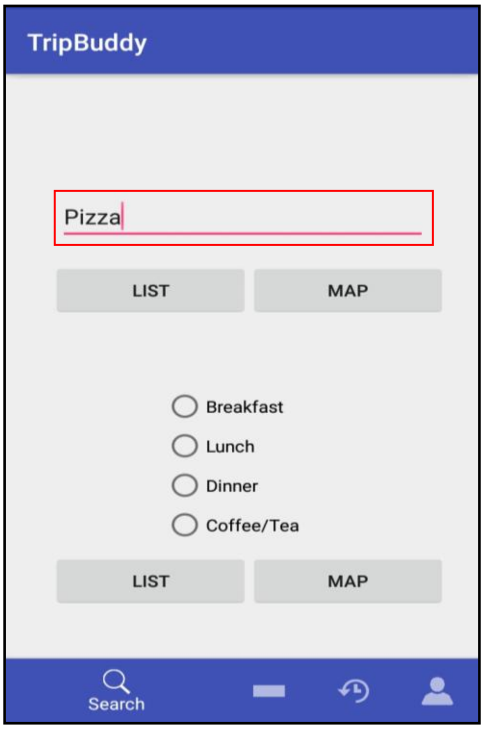
Implemented the apk file on android mobile and extracted the snapshots to describe the functions of the application.





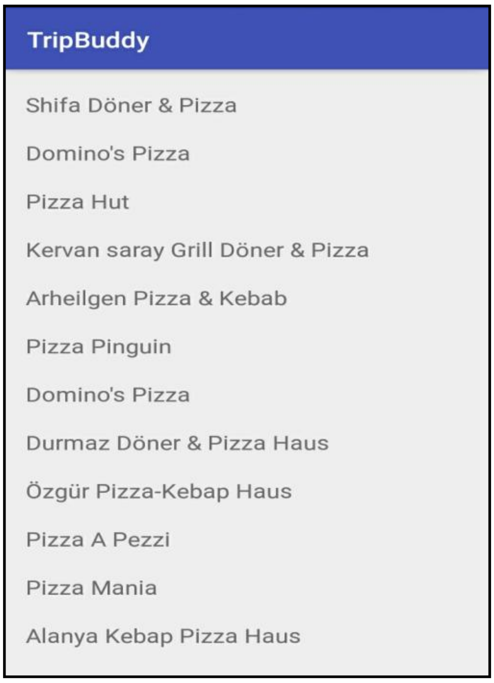
After successful Register/Login, the user can search specific venue using the search bar or can make use of radio button.




Pizza

When a user uses search button (For eg. Pizza), he has two option as mentioned below

1. Search by list
2. Search by map



TripBuddy

Shifa Döner & Pizza

Domino's Pizza

Pizza Hut

Kervan saray Grill Döner & Pizza

Arheilgen Pizza & Kebab

Pizza Pinguin

Domino's Pizza

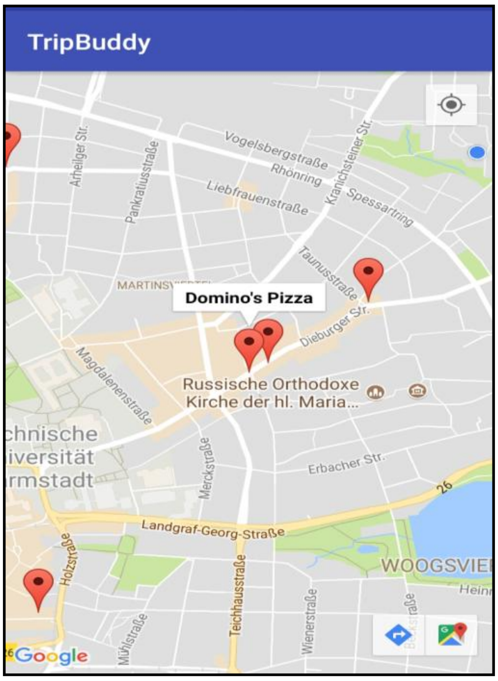
Durmaz Döner & Pizza Haus

Özgür Pizza-Kebap Haus

Pizza A Pezzi

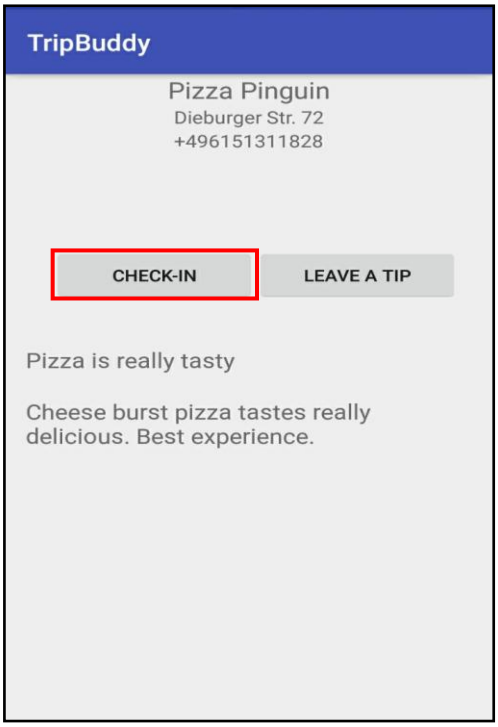
Pizza Mania

Alanya Kebap Pizza Haus



TripBuddy

When the user clicks on one of the list, he gets the details about that particular Venue and the user also gets the option of checking in and leaving a tip about the Venue. Once the user clicks on check in button, later when he checks the venue again, he can see that the venue is already visited.



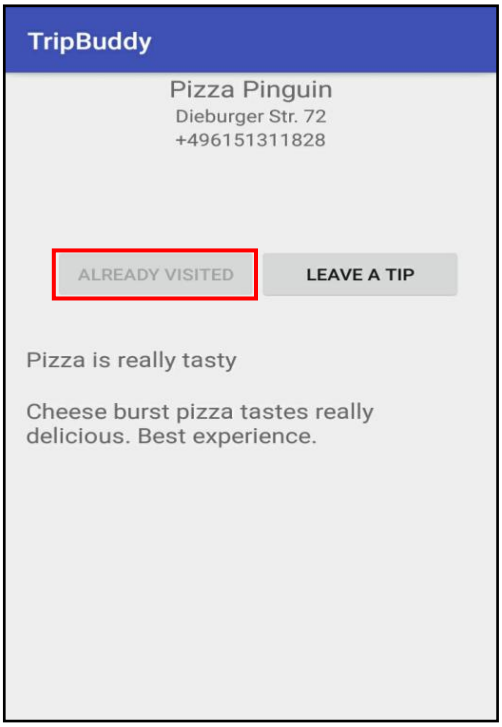
TripBuddy

Pizza Pinguin
Dieburger Str. 72
+496151311828

CHECK-IN | LEAVE A TIP

Pizza is really tasty

Cheese burst pizza tastes really delicious. Best experience.



TripBuddy

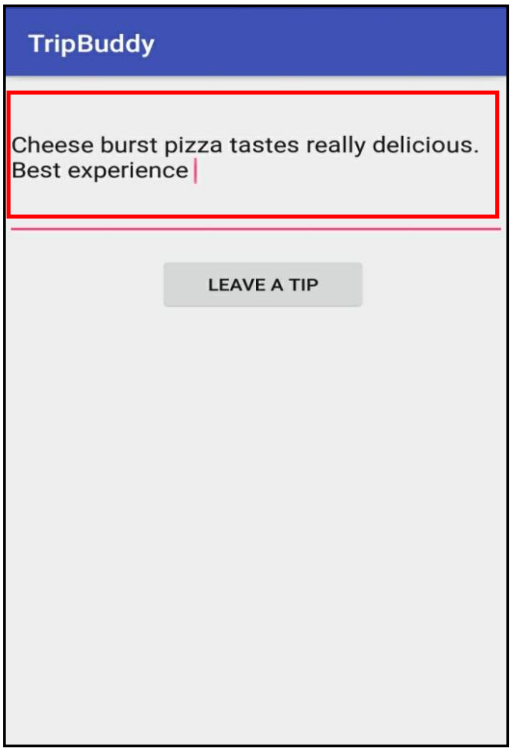
Pizza Pinguin
Dieburger Str. 72
+496151311828

ALREADY VISITED | LEAVE A TIP

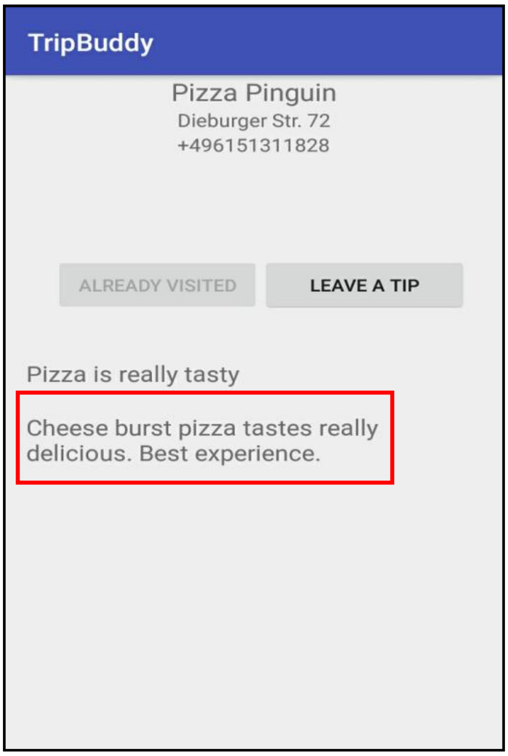
Pizza is really tasty

Cheese burst pizza tastes really delicious. Best experience.

Once the user leaves the comment for Venue then user can see the comments as shown below.



Cheese burst pizza tastes really delicious. Best experience

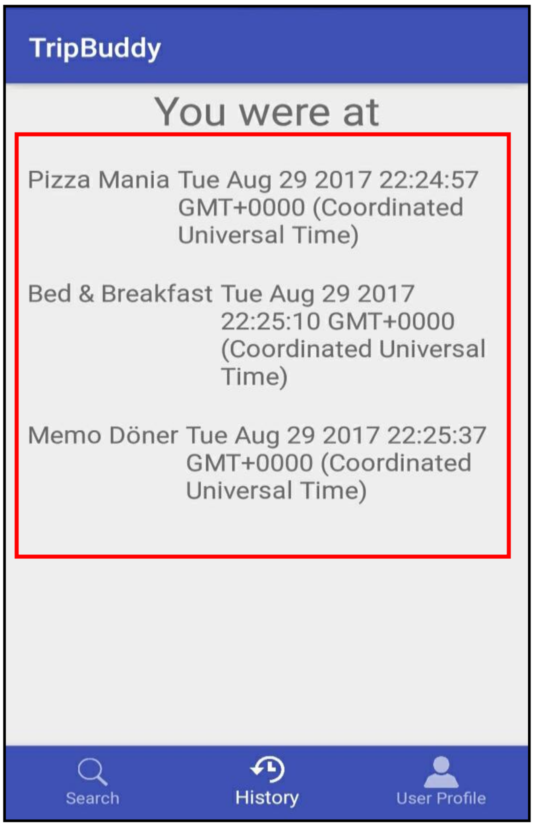


Pizza Pinguin
Dieburger Str. 72
+496151311828

Pizza is really tasty

Cheese burst pizza tastes really delicious. Best experience.

User can check all of his check-ins in the history tab.



You were at

Pizza Mania Tue Aug 29 2017 22:24:57 GMT+0000 (Coordinated Universal Time)

Bed & Breakfast Tue Aug 29 2017 22:25:10 GMT+0000 (Coordinated Universal Time)

Memo Döner Tue Aug 29 2017 22:25:37 GMT+0000 (Coordinated Universal Time)

5. List of features

- Registration/Login
- Get venues of other categories (for eg: bar, shisha, pizza, etc) from search bar
- Get venues of breakfast, lunch, dinner and Coffee/Tea by radio buttons
- Getting the results in list view as well as Google maps
- Get details of the searched venue in a new page such as Venue name, address and contact number
- Check-in to the venue
- Leave a tip about your experience at the venue
- User can check other users review about the venue
- All check-ins of a user is displayed in history tab
- Reset password
- Logout