# CS353-Database Systems

## Project Proposal Report

## Social Network for Check-ins

**Group 31**

Emre Yiğit Kuzhan – 21502056
Ferhat Karaca – 21400921
Serdar Erkal – 21400626
Halil İbrahim Azak 21501821

# Contents

**Contents** **2**

**1. Introduction** **3**

**2. Project Description** **4**

**3. Requirements** **5**

3.1 Functional Requirements 5

3.1.1 User 5

3.1.2 Location Manager 5

3.2 Non-Functional Requirements 5

3.2.1 Security 5

3.2.2 Performance 5

3.2.3 Usability 5

3.2.4 Scalability 5

3.3. Constraints (Pseudo Requirements) 6

**4. Limitations** **6**

**5. Entity Relation Model** **7**

**6. Website** **8**

# 1. Introduction

This Project proposal is for a Social Network for Check-ins. It will contain a brief description of the application we make, show the non-functional and functional requirements, discuss limitations and restrictions of our database and how system is going to be integrated will be shown with the  E/R model of our database.

The proposal starts with Project Description which contains brief information about the Project and we will try to Show how database will be used in this Project. It will continue with an explanation of why we need a database for Social Network for Check-ins and how we can use database for this project.

The proposal continues with requirements. In this part, functional and nonfunctional requirements will be examined and there will be an explanation for our pseudo requirements. Functional requirements are a must for setting up the functionalities and properties of project, which will be acquire by analyzing the users.  The nonfunctional requirements will be about the performance, reliability, security and usability goals for our system. In pseudo requirements , we will examine the technologies we will use in our system. In the limitations section, we will discuss the boundaries of our system.

At the end of the proposal, we will have our E/R diagram which is a basis for our database system.

# 2. Project Description

This application will be a web based system for making check-ins in different places, make comments about places and rating the places by their judgement, which is similar to FourSquare. The system will be used by regular people and owners of the places. System will include different ID's for different users which they will login and add friends with those ID's. There will be a password for each user which will be required in the login page. There will be also information about the names, comments and rates about the places in our database.

Regular users will be able to see those data in the application and change them according to their want by making a comment or rating the place they go. They will be able to search for places and friends and see the profiles of their friends, add friends by sending a friend request and see friend requests. In addition to this, application will have a news feed page, which will contains regular users' friends' latest check-ins, comments and ratings about the place chronologically. Regular users will also be able to make comments to their friends' check-ins.

Our system will be also used by the owners of the places. Owners will have a permission to add their places in our database, so that regular users can search the name of the place and check-in.

All users will be able to change settings in the application's setting menu, which will have options for security and privacy. Users also be able to change their ppassword, user name, and e-mail address in this menu.

Also, when users make a check-in in a location, application will notify the user if one of their friends' ever visited the place before. So that user can see the rating and comment of his or her friend.

# 3. Requirements

## 3.1 Functional Requirements

### 3.1.1 User

- Users can arrange their profile
- Users can change their privacy settings
- Users can make check-in
- Users can search for friends and locations
- Users can add and remove friends
- Users can comment on the check-ins and locations
- Users can change their password and username
- Users can like or dislike a location
- Users can identify their favorite location
- Users can login or sign up
- Users can see their friends' previous check-ins on specific location
- Users can report comments on their locations

### 3.1.2 Location Manager

- Managers can create a location
- Managers can arrange location's page
- Managers can see people who made checkin in their location
- Managers can comment on a check-in
- Managers can put a photo to their location page

## 3.2 Non-Functional Requirements

### 3.2.1 Security

- System gives opportunity users  to prevent their locations from non-friend users.
- Users can report comments for security
- Username and password will be unique therefore everybody has their own accounts

### 3.2.2 Performance

- System will return each operation immediately

### 3.2.3 Usability

- User friendly UI
- Users can easily find their friends with their names(Easy search)
- Different search bars with different purposes(Locations and users)
- Different modes for users and managers

### 3.2.4 Scalability

- Ability to grow in size of data without losing performance

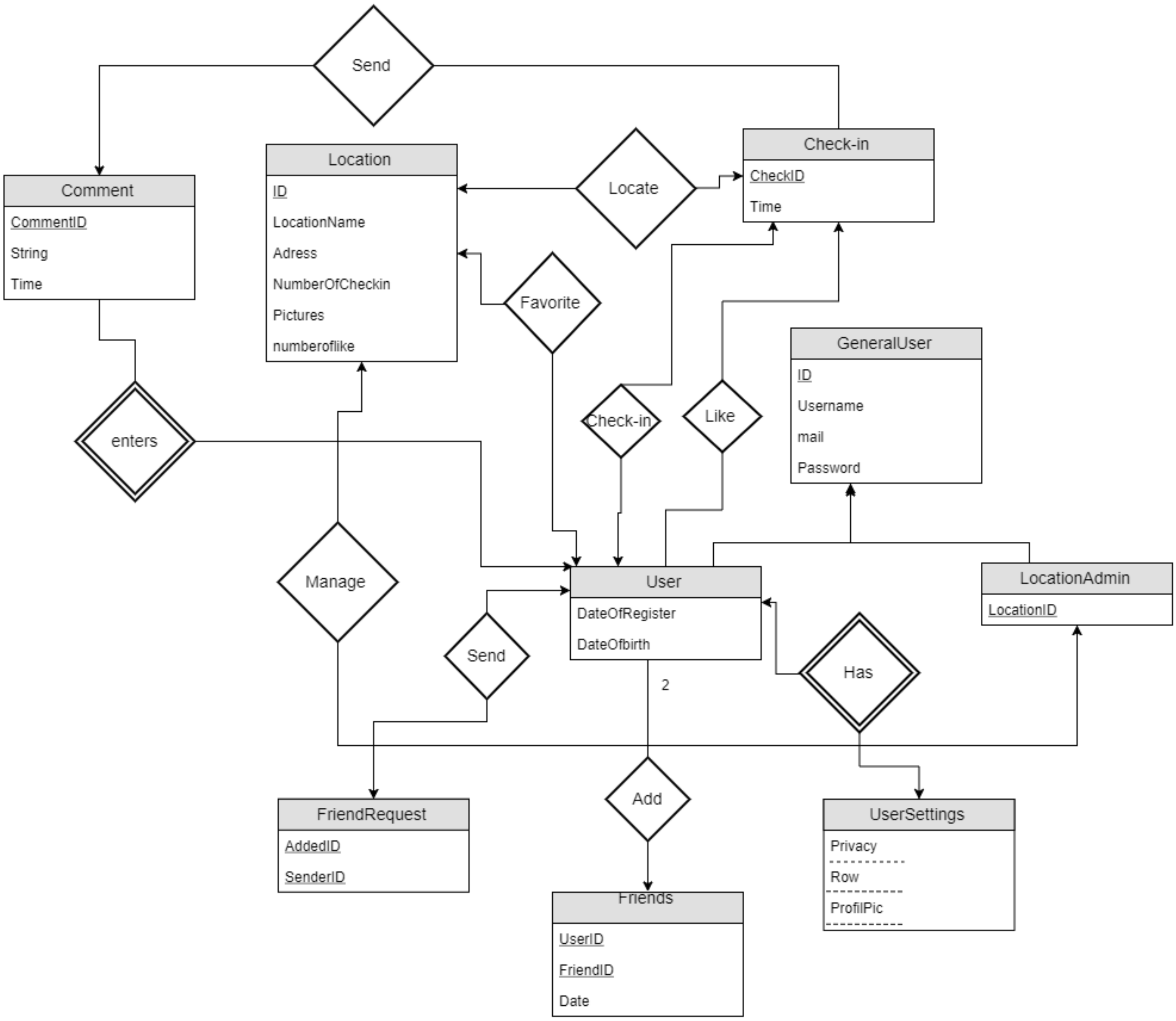
## 3.3. Constraints (Pseudo Requirements)

- MySQL will be used as the Structured Query Language.
- PHP, javascript, html will be used for software architecture.

# 4. Limitations

- Location Managers cannot add friends
- A user cannot change location specification
- A user cannot see a private account’s check-ins
- A command comment cannot be more than 400 character
- A Location can have at most 5 image
- A check-in cannot have more than 100 command
- Username and password cannot be longer than 32 character and also they cannot be less than 6

# 5. Entity Relation Model



# 6. Website

All the changes about the project will be on the following link.

https://cs353dbgroup.github.io/