



CS353 DATABASE SYSTEMS

Proposal

Social Network for Check-Ins

Group : 26

Furkan Hüseyin - 21302659

Irmak Turan - 21305099

Berfu Anıl - 21401502

Onur Kocahan - 21402013

- 1) Introduction
- 2) Description
 - a) Usage of DBMS for Social Network for Check-Ins (Why)
 - b) Approach (How)
- 3) Requirements
 - a) Functional Requirements
 - b) Non-functional Requirements
 - c) Pseudo Requirements (Constraints)
- 4) Limitations
- 5) E-R Diagram
- 6) Conclusion
- 7) Website

1) Introduction

The project will be a website where people can have an account to check-in at, comment on, and rate a place. Users will be able to check-in at a place to express that they are currently at that place. A user will sign-up at the system either as an owner of a place or as a visitor. So the website aims to help both visitors to express their feelings about a place that they have been and also aims to help owners to see the reputation of her place.

Users will be able to add other users as their friends, check-in at a place and see where her friends and she have been. Users will see other friends of her who are currently at the same location with her. Users will be able to leave comments about a place and one user can add multiple comments on a place. A user can recommend a place to a friend of her. A user may be either a Visitor or an Owner in this website.

Users will rate the place according to three features: price, ambience and service quality. A place will have type and a place can have multiple types. The popularity of the place will be calculated according to its type and three features listed above. The calculation will be done in the following way. Since a place has multiple types such as a place can be both dining and wine, every type will add a weight to the corresponding feature (price, ambience or service quality). For example, a wine place needs more ambience than price. However a dining place needs to be cheaper than being fancy. So having both types for a place will add weights in respective areas while calculating the popularity. So dining and wine place will be more popular because of its price and ambience rather than service. So even a lower service might end up in good popularity according to type. Rating and commenting will be separate to allow users to rate the place without leaving a comment or leaving a comment without rating the place. Users will be able to like a comment.

2) Description

a) Usage of DBMS for Social Network for Check-Ins (Why)

Social network for check-ins is a DBMS problem for two main reasons. First one is that the user sign-up and log in system requires to keep data of users. Along with that, the places and comments should also be stored to display later on and they should be stored in a decent way so that the large amount of data (thousands of comments) can be retrieved quickly. This is where DBMS comes into play to help storing

this data (user info, place info, comments etc). The second reason why we need DBMS is relations. Relations among data is also the main reason why this social network is a DBMS problem. With the help of DBMS, a user can see all places that her friends go to or a user can find another place just like the ones that she likes. These kind of queries and relations between data is easier with DBMS.

b) Approach (How)

User will have a social feed that she can see where her friends have checked-in. DBMS will be used to query her friends and the check-ins of her friends and sorting according to the date of check-in. User will also see a place that the user might like according to her friends' favorites and the type of places the user added favorite. DBMS will be used to query her friends' and her favorite places and types of those places to find the place to recommend.

3) Requirements

a) Functional Requirements

Place

Each place can have multiple type.

Places could have multiple images.

Popularity of the place is calculated according to its type.

User

Users have profile with information, image and check-in records.

User can block or delete a friend.

Users can see basic information of other users.

Users can see profile of their friends.

Users can see check-in records of friends.

Users can see mutual friends of other friends and users.

User can rate places according to 3 category which are ambience, price, service quality.

Users gets check-in notifications of their friends.

When users checked-in, notification is sent to their friends.

User can see people checked-in at a particular place.

User can add other users as a friend.

User can comment on places.

User can see popularity of the places.

User can see comment of the places.

User can edit their old comments.

User can check-in at the place.

Users can delete their friends.
Users can search places according to its name, type or address.
Users can recommend places to their friends.
Users can update their profile.
Users can search near-by places (places sharing commonality in address) according to popularity and location.

Owner

Owner can update the information of the place.
Owners are notified when their places get rated or commented.
Owner can see the number of check-ins at her place.

Comment

Comments could be liked.
Comments can be edited by the comment owner.

b) Non-functional Requirements

User-friendly interface

UI is easy to use for people from every age.
UI provides a neat view.
UI is easy-to-learn and easy-to-use.
UI is engaging, encouraging, and supportive in use.

Authetentication

Users have a password that has length requirements.
Users can deactivate their accounts.
Users can retrieve the missing password.

Reliability

Database will be backed up for expected failures.

Consistency

The database design meets the need of the system.

Platform independent

Since the system is a web-based application, the support range will be wide.

Usability

Non-experienced users can easily adapt the system.

Portability

The web application can run on numerous platforms such as Windows, MacOS.

Extensibility

New feature (such as storing edited comments) can be easily added to database system without changing the overall structure of the database.

Response Time

The interface between the UI and system shall have a maximum response time of three seconds.

Capacity

The system can handle transactional volumes.

c) Pseudo Requirements (Constraints)

MySQL will be used for database design.

PHP will be used for implementation of website.

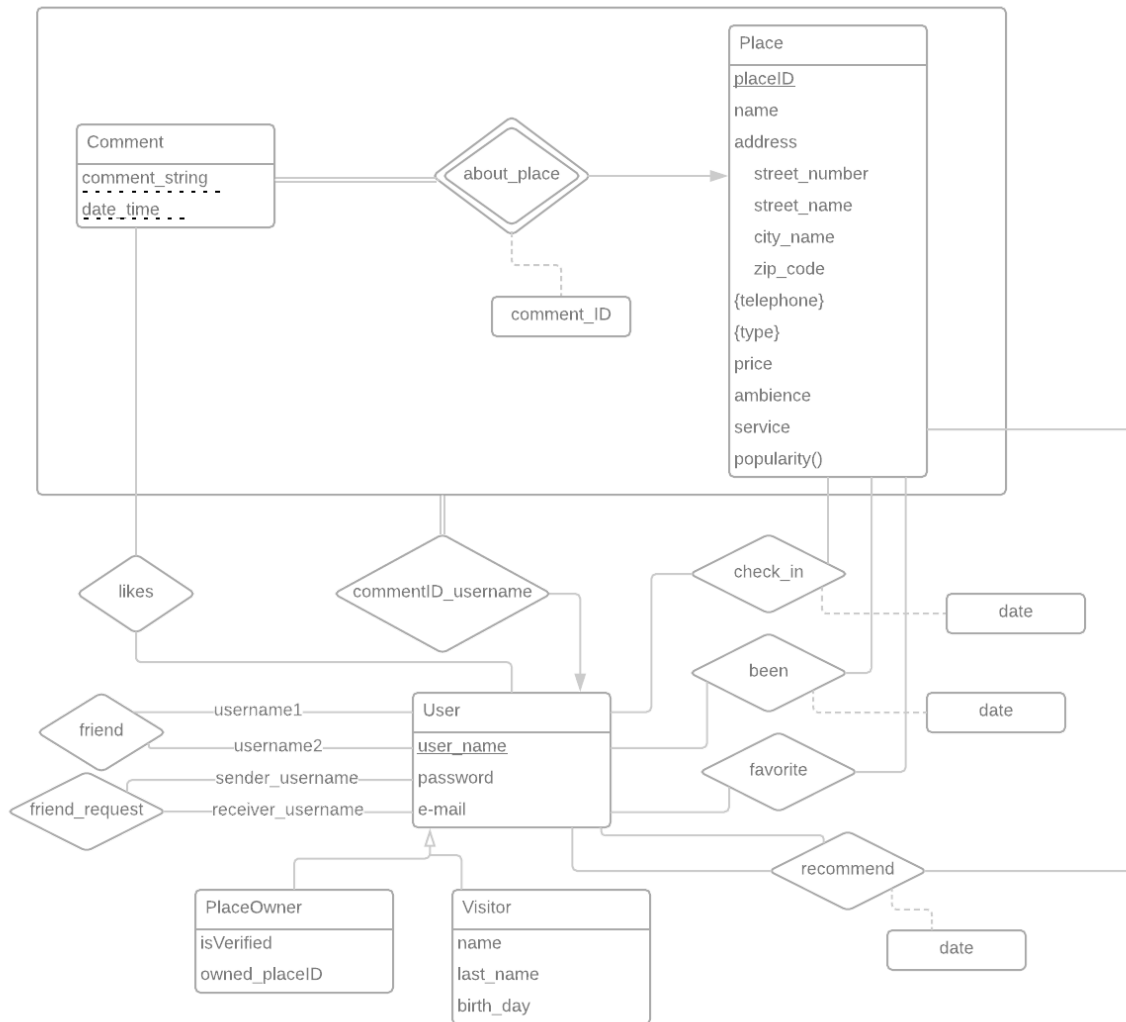
4) Limitations

Users cannot see full profile of other users unless they are friends.

Users cannot update information of places unless they are owners.

Owners cannot rate its place.

5) E-R Diagram



6) Conclusion

7) Website

Our project information website link: