Bilkent University
Department of Computer Engineering

CS 353-Database Management Systems

I-Venue
Group #27

Project Proposal

February 26, 2018

Seda Gülkesen - 21302403
İlhami Kayacan Kaya -21502875
Göktuğ Özdoğan -21301042
İlhami Özer -21300828

Course Instructor: Çağrı Toraman

This report is submitted to the GitHub in partial fulfilment of the requirements of the Database Management Systems Project, course CS353.

1.0 Introduction

This report gives the basic explanation of the application and functionalities of the Social Network for check-in project. By means of this report, the aim of the project, the requirements and functionalities of the system, the ways that the database has been integrated into the system and the limitations are clearly indicated.

The report begins with Project Definition part where the aim and basic functionalities of the system are explained. The following section explains why a database is needed, and basically, it discusses the significance of the database system for this project. Afterwards, how the database structure is going to be integrated into the project is discussed. In the Requirements part, functional requirements, nonfunctional requirements and pseudo requirements are explained in detail. Defining functional requirements has significant importance in order to obtain the system that we propose. There are two distinct end-users in the Functional Requirements part: user and venue owner. According to the permissions of actions of the users, functional requirements are listed separately. The reason why we separate to different users in the functional requirements part is we want to encapsulate the users according to their relation to the system.

In Nonfunctional Requirements, part, authentication, user-friendliness, quick response time, reliability, supportability and modifiability of the system are provided. In Pseudo Requirements, implementation technologies for back-end and front-end development are explained. In Limitations part, fundamentally, constraints of the system is discussed in detail. The scope of the project, user roles and constraints on the system are clearly indicated in this part.

After this part, E/R diagram showing the basis of the database system that will be used in our system is provided. The diagram includes the entities and the relations that represent the basis of the database system and in addition to those, it includes all the functional parts of the system that are proposed.

2.0 Project Description

I-venue is a web-based social platform for being in communication between friends and sharing online check in to a venue with those. There are two types of users in this platform that are distinguished with their roles in the system. The first type of users on the system is the venue owners who are able to create a venue profile (or multiple numbers of venues) in which the other type of the users, clients (we will simply call them users), are going to be able to check in to that venue. Therefore, it is crucial for the

system that the venue owners should create their venue's profiles at first in order to make them checkable by the other users. With this system, the user is going to be able to check-in to a place so that their friends will be able to see his/her check-in, to comment on that place or check-in. Thus, our application allows you to see your friend's activities in general. Another function is that our application provides the users to see a venue's profile by means of making a search with the venue's name or, with querying the venues regarding their types such as breakfast, dinner, fast food etc.

Our purpose of implementing this application is to establish an online platform in which the users will be able to share their activities, see their friends' activities, search for a particular venue to see its description and comments that have been shared by other users. This social platform will be a place for social people to share their locations with others to express their feeling about that venue and to get accompanies.

2.1 Why We Need Database for Social Network ?

I-venue is a social check-in platform that has a bunch of information that are related to the users, venues, check-in that need to be stored somewhere so that whenever a particular user interface is activated(called) the information that is related to that particular user interface can be derived from that somewhere where they are stored. The "somewhere" that is mentioned in the database that is going to hold the information of the users for their registration, signing in, user profile interface (homepage), the information of venues such as their names, addresses, the comments on a particular venue or friend's activities, the check-in information such as check in date and time, who-with whom-where information. As there is a lot of information that needs to be kept somewhere in order to show them to the related particular user, database comes into the action and plays its role as that storage.

2.2 How We Use Database as a Part of the Project?

In our system, database is used to store all the necessary information that will be required for the functional requirements of the system. It will at first enable users to log in with their particular credentials to the system by means of some queries to compare with their must information required for the logging in. In addition to that, for searching the venues, filtering the venues via the demands of the users, searching for friends or non-friend users the queries of the database system are used. Other than that, the profile information regarding the users and the venues will be queried according to the input

request (interrupt) that the user of the system had, and the corresponding information which is derived from the result of the query will be used to show them on user interface.

3.0 Requirements

3.1 Functional Requirements

I-Venue supports two main end-users: user and venue owner.

3.1.1 User

- Users should be able to register the system.
- User should be able to login with using their username and password or log out the system.
- Users should be able to add profile picture to their profiles.
- Users should be able to add their friends.
- Users should be able to see their friends' profile and see their activities.
- Users should be able to search a place (cafe, restaurants etc.)
- Users should be able to like or dislike a venue.
- Users should be able to comment on a venue.
- Users should be able to reply to the comments.
- Users should be able to check-in to the venue.
- Users should be able to check-in with their friends. (tagging)
- Users should be able to see who else is in the same spot.
- Users should be able to see who is the mayor of the venue.
- Users should be able to see their rank on the leaderboard.
- Users should be able to review a place.
- Users should be able to add pictures to added places.

3.1.2 Venue owner

- Veneu owner should be able to add a place.
- Veneu owner should be able to add or edit the description of the venue.
- Veneu owner should be able to reply to user’s comment.
- Veneu owner should be able to add pictures to the corresponding venue.

3.1.3 System Requirements

- The system should be able to give mayorship when the user has visited somewhere more often than anyone else, s/he can become the mayor of that place.

- The system should be able to give coins to the users. Each check-in earns users coins. (The user who has more coins than his/her friends, s/he will be on the leaderboard)

3.2 Nonfunctional requirements

3.2.1 Authentication

As mentioned before there are two types of end-users: user and venue owner. The permissions of the actions of the users should be explicitly specified and each of them must not be able to perform any extra action except the ones that they are given access to.

3.2.1 Ease of Use / User Friendliness

I-Venue is a web-based social network application. Most of the web-based user interfaces have similar layout which is easy to understand and use. We will try to build an interface which looks like popular social platforms. First time visitors should not have any unfamiliarity to use I-Venue. The application must be easily understandable by the users from different age groups. In order to achieve these goals, our main concerns will be:

- It must be easy to navigate. There is no need to make four or five layer of menu options. User should be able to access desired menus effortless. The most important point will be to properly categorize the required places.

- It must have attractive layout. Images and text should be arranged in a fashion that user can see what they looking for.

- Main feature, check-in, must not be complicated to do. We are planning to add a simple button to check-in. It will ask necessary information and perform the check-in very quickly.

3.2.2 Quick Response Time

In our system, there is a big storage of information corresponding to the users, venues, venue owners, check-in etc. Therefore, the system needs to be very fast in order to

respond to the request of the users, make the requested database queries faster and faster algorithms that will minimize the time cost. As a result, the system should be designed in a way that it will provide better performance for database queries, front-end and back-end of the system.

3.2.3 Reliability

This application is aimed to be bug-free because facing with unexpected terminations could be annoying for the user. To be able to avoid such experiences, the tests will be done for each function of the application.

Since the application is a social network application, its database holds highly valuable social data which gives information about the user's lifestyle. Thus, reliability is considered one of our main concern should be taken into consideration.

The users should not be able to change any information about their previous check-ins.Also, the application should be continue without any failure when the user tries invalid actions.

3.2.4 Supportability

Since it is web-based social network application, the system should be platform independent. It can be capable of being run on a computer from any operating system. It should be accessible anywhere, anytime by computers, smart phones or tablets with internet connection.

3.2.5 Modifiability

The system should be modifiable because social network applications need quick and rapid changes. Whenever the developer wants to add new features to the application, an update will be easily implemented by the developer. In this manner, object oriented design is chosen in order to add new classes and objects into existing code properly.

3.3 Pseudo Requirements

MySQL will be used for the database.

Java and JSP will be used for backend.

Website will be created with JavaScript, HTML and CSS.

4.0 Limitations

Users cannot add the new venue. Only venue owner can add new venue.

If the user deletes his/her account, his/her check in will be displayed in venues' profile. However, if venue is deleted by venue owner, the check in will be deleted from user's profile.

The user cannot do check in, if he/she does not log into the application.

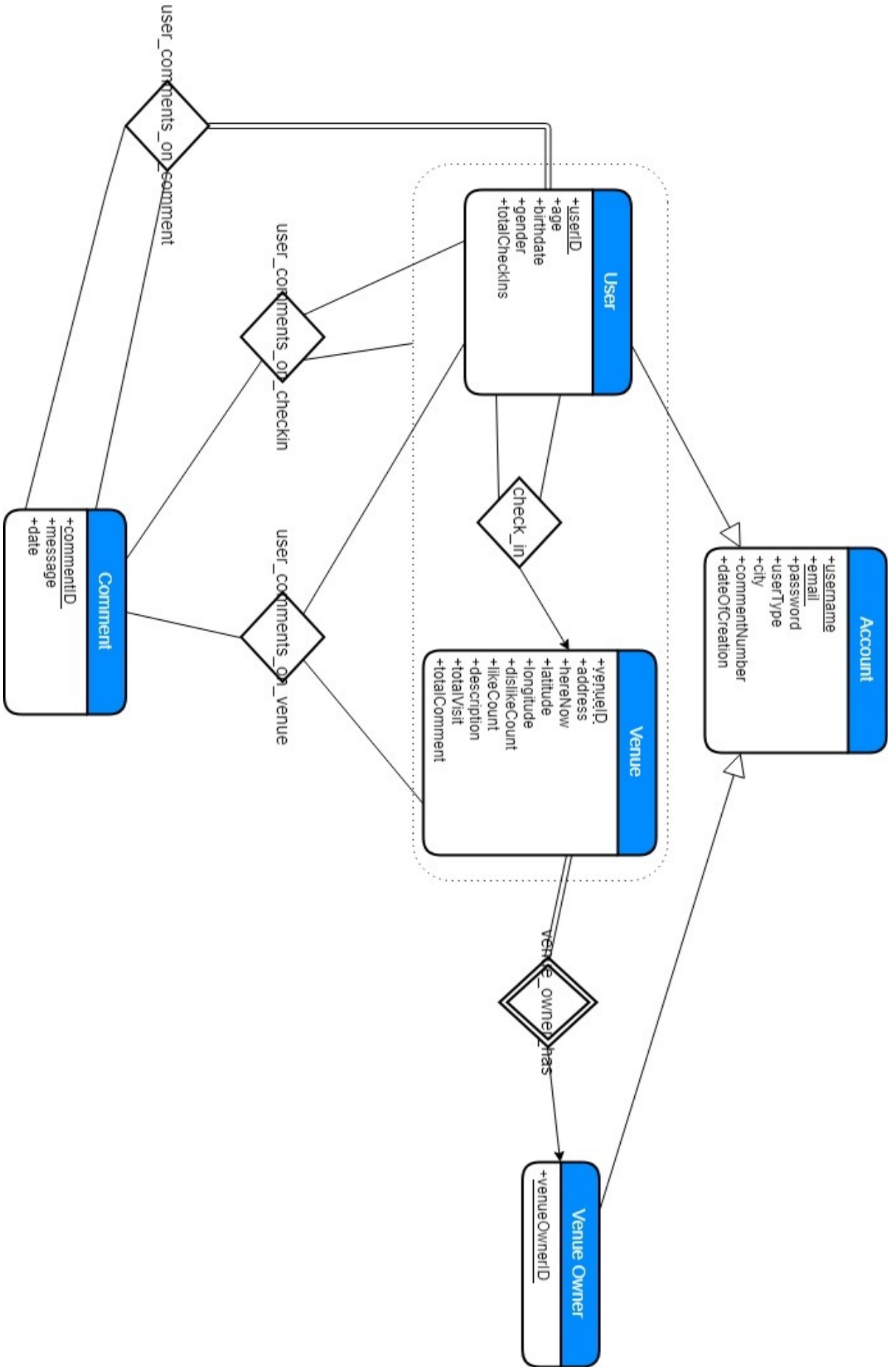
The venue owner cannot add new venue, if he/she doesn’t log into the application.

The user cannot have the nickname which is taken earlier by other user.

The venue owner cannot have the nickname which is taken earlier by other venue owner.

The user cannot make a comment, if he/she doesn’t log into the application.

5.0 E/R Diagram



6.0 Conclusion

The Airway is a web based application for letting your friends to know where you are and also to you where they are. It provides much functionality to its user such as evaluating venue if they do check in there.In this report, we have detailed description of the project and we explained the importance of Database Management System as a part of project and how it used in our project. The requirements of project are clearly indicated. To do it, we explained the system requirement and nonfunctional requirements of the project. Limitations of the system are also provided in the report.As a basis of our database design, E/R Diagram (Entity Relation Diagram) is provided.

7.0 WebSite

The following is the link to our CS353 Database Systems Project:

https://github.com/sedagulkesen/I-Venue-CS353.git