



CS 353 Database Systems

Project Proposal

Local Events Application

Tuna Öğüt - 21803492
Erengazi Mutlu - 21803676
Yiğit E. Dinç - 21704275
Arda Güven Çiftçi - 21801763

<https://cs353ocp.github.io/cs353ocp/>

Table Of Contents

Table Of Contents	2
1.Introduction	3
2. Project Description	4
3. Requirements	4
3.1 Functional Requirements	4
3.2 Non-Functional Requirements	5
3.3 Constraints (Pseudo Requirements)	6
4. Limitations	6
5. Entity Relationship Model	7

1.Introduction

This project proposal is about a local events application. This proposal will include a brief description about the project, functional and non-functional requirements, limitations of our application's database and finally the entity relationship model of our database.

Project description will include information about the project, why the database is an essential part of the project and how we will use it.

In the requirements part, functional, non-functional and pseudo requirements will be introduced. Functional requirements part plays an essential role to the project because it includes functionalities and properties of the project. In the non-functional part we will be discussing the performance, security, reliability and usability goals for our application. Lastly, in the pseudo requirements, we will introduce which technologies are going to be used in our project.

Limitations part will include the boundaries and constraints that our application will have. At the end, we will add our E/R diagram, which is the fundamental of our project.

2. Project Description

This project is aimed to be a local event application that is used by people who want to know what is going on around their location. This project's main purpose is to create a multi-functional social platform that can deliver various types of events, such as theater, concerts, sports, and so on. People will see a list of events that are happening according to their location and also, and they will be able to choose these events related to their types as we mentioned above. Users can follow each other and also their favorite organizer. They will be able to see the events that their friends are attending. If the users meet the needs of certain requirements, decided by the admin, they can be the organizer for an event. Events can have a ticket feature which is decided by the organizer and, if there is a ticket requirement for a certain event, sales will be made

through our social platform. Since some events can have special requirements, such as age restriction or dress code, the organizer assigns these requests over the system and the people who will participate must comply with these requests. Lastly, admins will be able to create reports such as leaderboards, accomplishments, and most popular events, etc. Admins will be able to create statistical data about the events.

3. Requirements

3.1 Functional Requirements

3.1.1 User

- Each user has to sign in to the program, using email and password.
- Users have to enter their birthday.
- Users can use discount coupons.
- Users can follow other users and see their activity.
- Users can be applicants and/or organizers.
- Users will be able to see organizers and events rating.

3.1.2 Applicant

- Applicants can purchase tickets and join events.

3.1.3 Organizer

- Organizers can create events.
- Organizers can determine the rules of the event.

3.1.4 Admin

- Admins can see the event's statistics.
- Admins can create reports about the events.
- Admins can change/edit the requirements of events.

- Admins can ban users.
- Admins can change the status of organizer. They can revoke their organizer role.

3.2 Non-Functional Requirements

3.2.1 Security

Unique email and password must be needed in order to sign in to the system. Since our application has a feature that allows people to see what their friends are attending, there might be certain situations that a person might not want to be seen by their friends. The payment system will be implemented according to the current security standards. Also, admins will have authority to ban people which means they can use their authority to malicious people.

3.2.2 Performance

Our system will return each response immediately. Thanks to this, there will be no noticeable delay in our system.

3.2.3 Usability

The UI should be advanced enough to meet the needs of the users and also it should be easy enough to use. Therefore, there will be a balance between those particular trade-offs.

3.2.4 Scalability

This application needs to be modular and modifiable enough to grow in case of data size. So that in the future, the project can handle the data size.

3.2.5 Maintainability

The system should be developed in a way that if a problem or error occurs, it must be easy enough to handle. Also, if there is an update needed, it should be easy enough to deploy to the production.

3.3 Constraints (Pseudo Requirements)

- NodeJS, ReactJS, Javascript, HTML and CSS will be used in backend and frontend.
- PostgreSQL will be used for Structured Query Language

4. Limitations

- Each user must sign up with a password and unique email.
- Passwords can not exceed 32 characters.
- Users can not see who is attending which event unless they are allowed to.
- Applicants can not change or modify events unless they are the organizer for an event.
- Applicants can not organize an event unless they meet the needs.
- Applicants can not buy tickets unless they meet the needs of a certain event.
- Users can not see the statistics of the event until the admin creates it.

5. Entity Relationship Model

