CS353 Database Systems

Project Proposal

Local Events Application

Group 1

Alper Mumcular - 21902740
Vesile İrem Aydın - 21902914
Ravan Aliyev - 21500405
Ece Kahraman - 21801879

Table of Contents

1. Introduction

This is a proposal report for the Local Events Application. In this report firstly the system will be described, then functional and non-functional requirements along with the limitations of the system will be explained. Lastly, the detailed E/R model of the database will be given.

The proposal starts with a description of the Local Events Application system by explaining its scope and properties. In the following two sections, why and how the database is going to be used as a part of the system will be explained. In the requirements section, the functional and non-functional requirements of the system will be described.

The non-functional requirements describe the concerns about usability, security, performance, privacy, reliability, and concurrency issues. It will be followed by the limitations that are taken into account while designing the Entity-Relationship model which is given in the last section.

2. Project Description

Local Events Application is a web application where the organizers can organize events and other users can join those events. Organizers can generate the events, modify their date, description, location, and the type and restrict those events according to age. Organizers have two types: verified and unverified. Verified organizers can create paid events while unverified organizers can only create free events. The application does not allow unverified organizers to create paid events to increase the security of the system. All organizers have a popularity index to show the popularity of their events. Participants can search events and filter search results according to age restrictions, location and type. They can join the desired event if it is free or can purchase a ticket if that event is paid for. Participants can save their credit card information in the application and easily pay for the events. Participants also collect points from paid events they attended and when their points exceed a limit, they can get free wallet balance. Admin is responsible for verifying the organizers, can ban users, cancel events and create system reports like “most popular events” according to the number of the participants.

3. Why We Use a Database

As it is mentioned above, the system has many functionalities that require a database. All the information of users and events creates lots of data and it needs to be stored in the application. Also, this information needs to be updated, filtered, deleted and listed when needed. That’s why the application needs a database to function properly.

4. How We Use a Database

The database of the system will store all the information of users and events. The information about users such as email, password, and location will be stored in the database. Also, events and information related to them will be stored in the database. Events can also be updated, listed, and deleted and the database will be used for all these operations.

5. Requirements

5.1. Functional Requirements

5.1.1. Web Administrator

- The person who is able to manage the website itself.
- Admins are able to verify organizers so they can host paid events.
- Admins are able to ban user accounts.
- Admins are able to cancel events.
- Admins can create system reports (finding the most popular events and the most expensive/cheapest events).
- Admins are able to sign in with a nickname and password.

5.1.2. User

- A user can sign up with a valid email address and password.
- A user is able to change their password.
- A user is able to change their contacts (email, address, phone number).
- A user is able to switch their site view between participant and organizer.

5.1.2.1. Participant

- A user is considered as a participant once they sign up.
- A participant is able to search for an event by its title.
- A participant is able to search for events by category.
- A participant is able to search for events by locations.
- A participant can join an event.
- A participant is able to purchase tickets.
- A participant can perform cancellation if they are given the chance.
- A participant can view their participation and tickets.
- A participant can save his/her credit card information to purchase tickets easily.
- A participant is able to collect points by joining paid events. With these points, a participant can get an on-site wallet balance.
- A participant is able to become an organizer.

5.1.2.2. Organizer

- The user who creates an event.
- An organizer can create an event.
- An organizer can set the details of an event, such as event title, location, date and time, description, type, and category.
- An organizer can cancel an event.
- An organizer can view the participant list.
- An organizer can view the lists of all events (upcoming, ongoing, and past).
- An organizer can cancel a participant’s participation.
- An organizer can gain popularity for each person attending his/her/its event.
- An organizer can become a verified organizer by verification of a website admin.
- An organizer can become a participant.
- An organizer has a popularity index determined by the number of users participating in its events. This index is visible to all users.

5.1.2.3. Verified Organizer

- A verified organizer can do all the things an organizer can do.
- A verified organizer can create paid events.
- A verified organizer can set ticket prices for the paid event.

- A verified organizer can set the maximum number of tickets per participant for a paid event.
- A verified organizer can change his/her name to an organization name.
- A verified organizer can enter his/her/its International Bank Account Number (IBAN) to get a payment for a paid event.

5.1.3. Event

- An event has a specific date and location.
- An event has a category related to its content such as a sports event or concert.
- An event has a participation quota.
- An event may have an age limit and users under that limit cannot participate in the event.

5.1.3.1. Paid Event

- A paid event has all attributes of an event.
- A paid event has a price determined by its organizer.
- A paid event may have more than one ticket price.
- A paid event has a maximum ticket limitation per participation.

5.1.4. Wallet

- A wallet keeps the balance of its user.
- A wallet is used to pay for the paid events and receive balance from the points of the participant.
- A wallet keeps the credit card information of the user.

5.1.5. Ticket

- A ticket is given for the paid events.
- A ticket has a price.
- A ticket may be refundable or non-refundable.

5.2. Non-Functional Requirements

5.2.1. Usability

- Users should be able to understand the basics of a website easily.
- Users should be able to filter and find events in accordance with their interests.
- Each event should be in a category that fits its features.

5.2.2. Security

- All users must verify their accounts via email and phone number.
- Purchasing tickets for an event will be provided by using an online payment service.
- Any organizer who wishes to organize a paid event must have already been verified by an admin to avoid any fraudulent events.

5.2.3. Performance

- The system’s response to any human interactions such as I/O should not exceed 1 second.

5.2.4. Privacy

- Participants in events are only visible to the organizers of those events.
- Contact information (email and phone number) of participants in events is only visible to organizers of those events.

5.2.5. Reliability

- User data must be preserved when the platform shuts down.

5.2.6. Concurrency

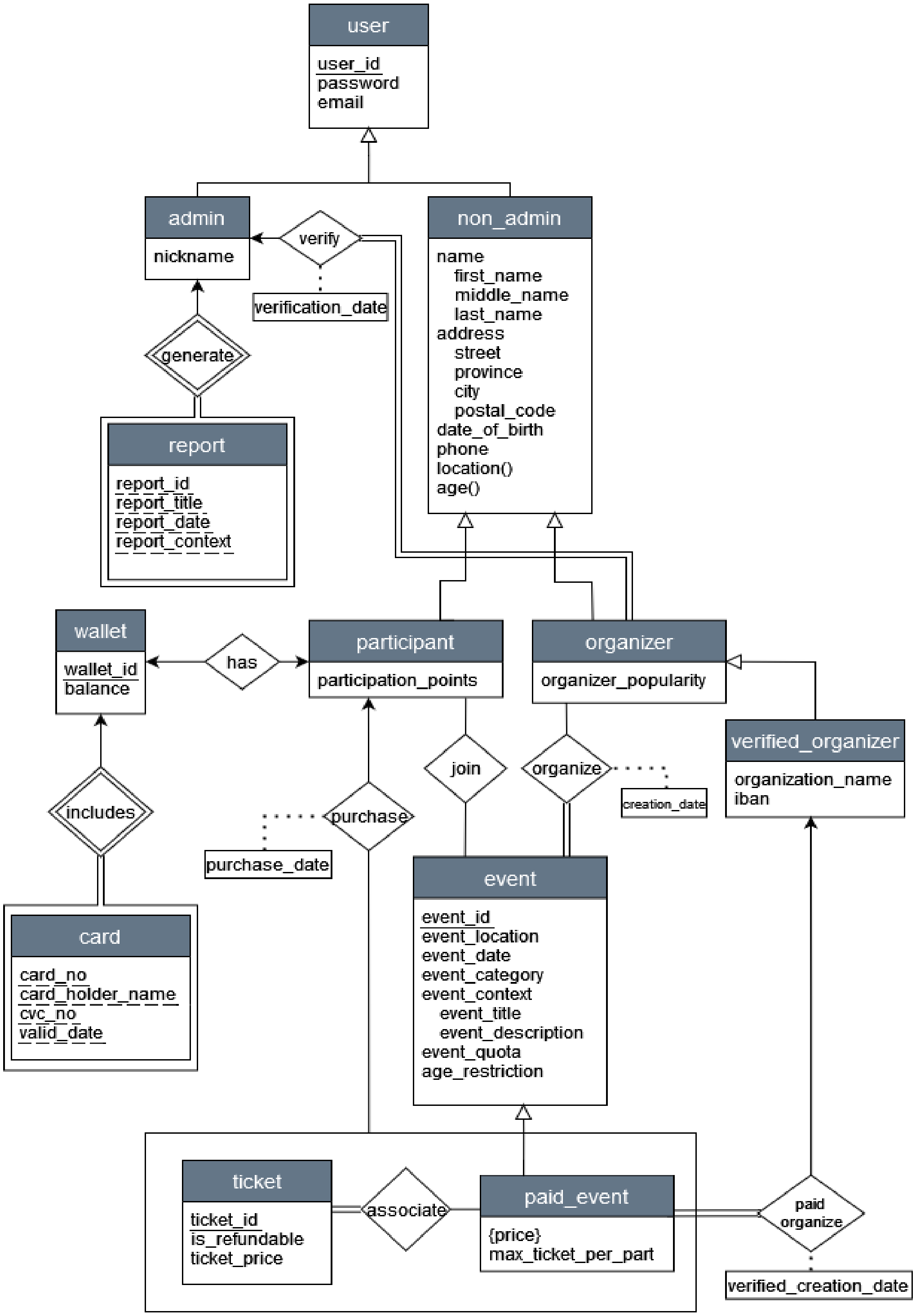
- The server of the platform must be able to handle user requests concurrently.
- A user does not have to wait for another user’s requests to be handled first.

6. Limitations

The system constraints are listed as follows:

- Each event must have at least one organizer.
- Each event must have a location.
- Each event must have an upcoming date.
- Each event must have a category.
- Each event must have an event type.
- Each event must have a title and description.
- Each event can only be edited by its organizers.
- Each event must have a specified number of participants.
- Each paid event must have a verified organizer.
- Each paid event must have a maximum number of tickets per participant to buy.
- Each verified organizer must be verified by an admin.
- Each verified organizer must have an IBAN to get payment from paid events.
- Each participant can not buy more tickets than the maximum number of tickets per participant for a specific event.
- Each user must have an address.
- Each non-admin user must have a phone number.
- Each user must have a valid email and password.
- Each participant must have a unique wallet.
- Password length must have at least 8 characters and contain at least one uppercase and lowercase character.
- Users who do not meet the restrictions cannot participate in the specific event.
- Each report must be generated by an admin.
- Each card must be associated with a wallet.

7. Conceptual Design


user
user_id
password
email
admin
nickname
verify
verification_date
non_admin
name
first_name
middle_name
last_name
address
street
province
city
postal_code
date_of_birth
phone
location()
age()
generate
report
report_id
report_title
report_date
report_context
wallet
wallet_id
balance
has
participant
participation_points
organizer
organizer_popularity
verified_organizer
organization_name
iban
join
organize
creation_date
includes
purchase
purchase_date
card
card_no
card_holder_name
cvc_no
valid_date
event
event_id
event_location
event_date
event_category
event_context
event_title
event_description
event_quota
age_restriction
ticket
ticket_id
is_refundable
ticket_price
associate
paid_event
{price}
max_ticket_per_part
paid organize
verified_creation_date