
Bilkent University

Department of Computer Engineering

# CS353 TERM PROJECT

## Design Report

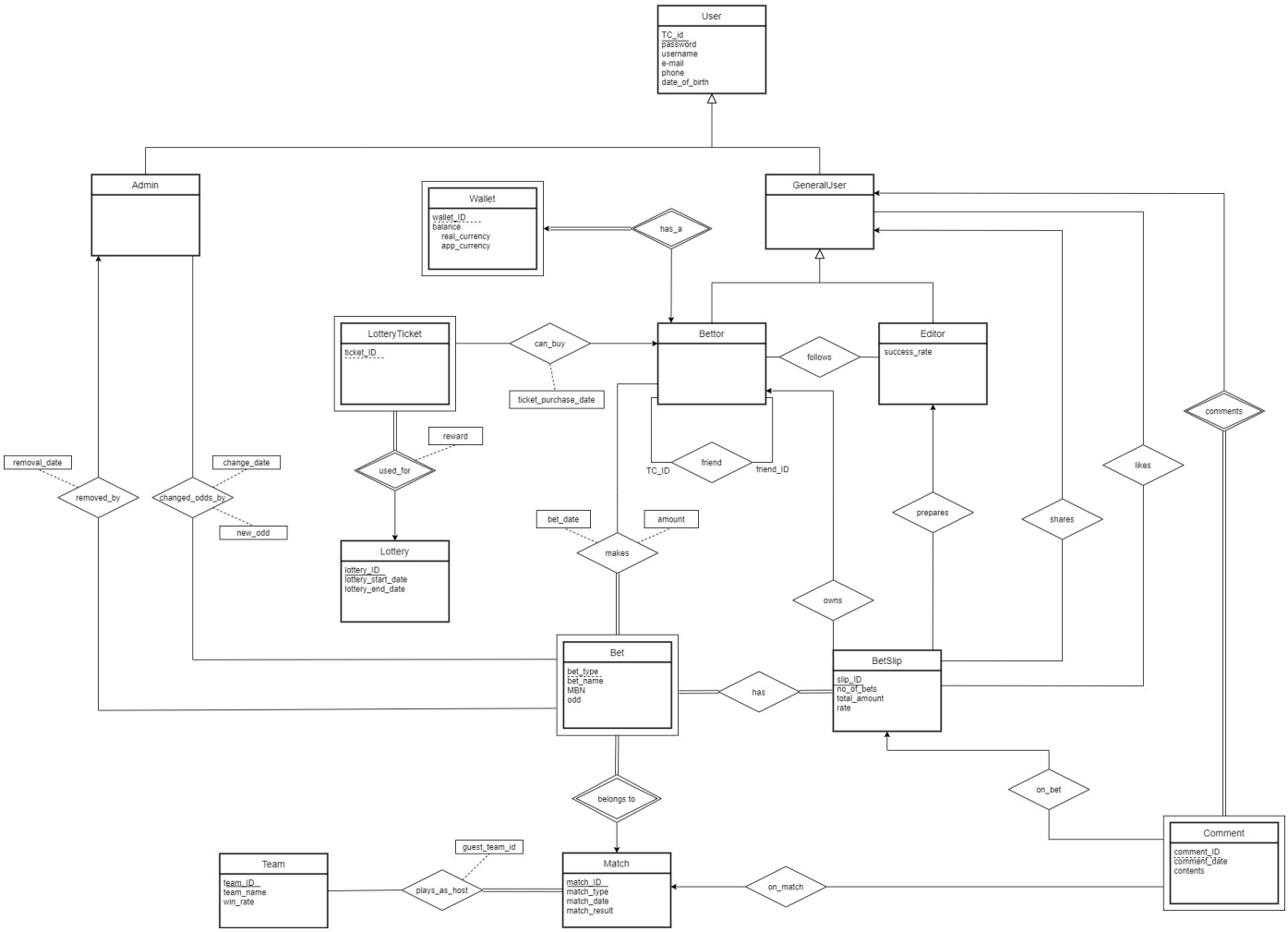
Zeynepnur Cavcar, Ece Teker, Yaren Durgun, Selen Görgün

Instructor: Hamdi Dibeklioğlu
Teaching Assistant: Mustafa Can Çavdar

April 8, 2022

# Table of Contents

## 1. Revised ER Diagram




## 2. Relations

All relations are in 3NF format.

- User (<u>TC_id</u>, password, username, email, phone, date_of_birth)
    - *Candidate Keys:* TC_id, username, email, phone
    - *Primary Key:* TC_id
    - Dependencies:
        - TC_id → password, username, email, phone, date_of_birth
        - username → TC_id, password, email, phone, date_of_birth
        - email → TC_id, password, username, phone, date_of_birth
        - phone → TC_id, password, username, email, date_of_birth
    - Table Creation:

**CREATE TABLE** User (
TC_id **INT NOT NULL** ,
password **VARCHAR(20) NOT NULL,**
username **VARCHAR(30) NOT NULL,**
email **VARCHAR(100) NOT NULL,**
phone **INT NOT NULL,**

date_of_birth **DATE NOT NULL**
**PRIMARY KEY (**TC_id**))**

- GeneralUser (TC_id)
  - *Candidate Keys:* TC_id
  - *Primary Key:* TC_id
  - *Foreign Keys:* TC_id to User Relation

**CREATE TABLE** GeneralUser (
TC_id **INT NOT NULL,**
**PRIMARY KEY (**TC_id**),**
**FOREIGN KEY** (TC_id) **REFERENCES** User (TC_id))

- Admin (TC_id)
  - *Candidate Keys:* TC_id
  - *Primary Key:* TC_id
  - *Foreign Keys:* TC_id to User Relation

**CREATE TABLE** Admin (
TC_id **INT NOT NULL,**
**PRIMARY KEY (**TC_id**),**
**FOREIGN KEY** (TC_id) **REFERENCES** User (TC_id))

- Bettor (TC_id)
  - *Candidate Keys:* TC_id
  - *Primary Key:* TC_id
  - *Foreign Keys:* TC_id to GeneralUser Relation

**CREATE TABLE** Bettor (
TC_id **INT NOT NULL,**
**PRIMARY KEY (**TC_id**),**
**FOREIGN KEY** (TC_id) **REFERENCES** GeneralUser (TC_id))

- Editor (TC_id, success_rate)
  - *Candidate Keys:* TC_id
  - *Primary Key:* TC_id
  - *Foreign Keys:* TC_id to GeneralUser Relation
  - Dependencies:
    - TC_id → success_rate

**CREATE TABLE** Editor (
TC_id **INT NOT NULL,**
success_rate **FLOAT,**
**PRIMARY KEY (**TC_id**),**
**FOREIGN KEY** (TC_id) **REFERENCES** GeneralUser (TC_id))

- Wallet (TC_id, wallet_id, real_currency, app_currency)
  - *Candidate Keys:* TC_id, wallet_id
  - *Primary Key:* TC_id
  - *Foreign Keys:* TC_id to Bettor Relation
  - Dependencies:

        - TC_id, wallet_id → wallet_id, real_currency, app_currency

**CREATE TABLE** Wallet (
TC_id **INT NOT NULL,**
wallet_id **INT NOT NULL**,
real_currency **FLOAT,**
app_currency **FLOAT, PRIMARY KEY (**TC_id**),**
**FOREIGN KEY** (TC_id) **REFERENCES** Bettor (TC_id))

- Match (match_id, match_type, match_date, match_result)
    - *Candidate Keys:* match_id
    - *Primary Key:* match_id
    - Dependencies:
        - match_id → match_type, match_date, match_result

**CREATE TABLE** Match (
match_id **INT NOT NULL**,
match_type **VARCHAR(20) NOT NULL**,
match_date **DATE NOT NULL,**
match_result **VARCHAR(20)**
**PRIMARY KEY (**match _id**))**

- Team (team_id, team_name, win_rate)
    - *Candidate Keys:* team_id, team_name
    - *Primary Key:* team_id
    - Dependencies:
        - team_id → team_name, win_rate
        - team_name → team_id, win_rate

**CREATE TABLE** Team (
team_id **INT NOT NULL**,
team_name **VARCHAR(30 NOT NULL)**,
win_rate **FLOAT,**
**PRIMARY KEY (**team _id**))**

- TeamsPlaying (host_id, guest_id, match_id)
    - *Candidate Keys:* match_id
    - *Primary Key:* match_id
    - *Foreign Keys:* match_id to Match, host_id to team_id in Team, guest_id to team_id in Team
    - Dependencies:
        - match_id → host_id, guest_id

**CREATE TABLE** TeamsPlaying (
host_id **INT NOT NULL**,
guest_id **INT NOT NULL**,
match_id **INT NOT NULL,**
**PRIMARY KEY (**match_id**)**
**FOREIGN KEY** (host _id) **REFERENCES** Team (team _id)
**FOREIGN KEY** (guest _id) **REFERENCES** Team (team _id)

**FOREIGN KEY** (match _id) **REFERENCES** Match (match _id))

- BetSlip (<u>slip_id</u>, no_of_bets, total_amount, rate)
    - *Candidate Keys:* slip_id
    - *Primary Key:* slip_id
    - *Foreign Keys:* editor_id to TC_id in Editor, bettor_id to TC_id in Bettor
    - Dependencies:
        - slip_id → no_of_bets, total_amount, rate

**CREATE TABLE** BetSlip (
slip_id **INT NOT NULL**,
no_of_bets **INT,**
rate **INT,**
**PRIMARY KEY (**slip_id**)**)

- EditorPreparesSlip (<u>slip_id</u>, editor_id)
    - *Candidate Keys:* slip_id
    - *Primary Key:* slip_id
    - *Foreign Keys:* slip_id to BetSlip, editor_id to TC_id in Editor
    - Dependencies:
        - slip_id → editor_id

**CREATE TABLE** EditorPreparesSlip (
slip_id **INT NOT NULL**,
editor_id **INT NOT NULL,**
**PRIMARY KEY (**slip_id**),**
**FOREIGN KEY** (slip _id) **REFERENCES** Slip (slip _id)
**FOREIGN KEY** (editor _id) **REFERENCES** Editor (TC _id))

- Bet (<u>match_id, bet_type,</u> bet_name, MBN, odds)
    - *Candidate Keys:* (match_id, bet_type), (match_id, MBN, odds)
    - *Primary Key:* (match_id, bet_type)
    - *Foreign Keys:* match_id to Match
    - Dependencies:
        - match_id, bet_type → bet_name, MBN, odds
        - match_id, MBN, odds → bet_type, bet_name

**CREATE TABLE** Bet(
match_id **INT NOT NULL**,
bet_type **VARCHAR(20) NOT NULL**,
bet_name **VARCHAR(20)**,
MBN **INT NOT NULL**,
odds **FLOAT NOT NULL,**
**PRIMARY KEY (**match_id, bet_type**),**
**FOREIGN KEY** (match _id) **REFERENCES** Match (match _id))

- SlipHasBet (<u>slip_id, match_id, bet_type</u>)
    - *Candidate Keys:* (slip_id, match_id, bet_type)
    - *Primary Key:* (slip_id, match_id, bet_type)

- *Foreign Keys:* slip_id to BetSlip, (match_id, bet_type) to Bet

**CREATE TABLE** SlipHasBet (
slip_id **INT NOT NULL**,
match_id **INT NOT NULL**,
bet_type **VARCHAR(20) NOT NULL,**
**PRIMARY KEY (**slip_id, match_id, bet_type**),**
**FOREIGN KEY** (match_id, bet_type) **REFERENCES** Bet (match_id, bet_type)
**FOREIGN KEY** (slip _id) **REFERENCES** Slip (slip _id)
**FOREIGN KEY** (match _id) **REFERENCES** Match (match _id))

- BettorOwnsSlip (bettor_id, slip_id)
  - *Candidate Keys:* slip_id
  - *Primary Key:* slip_id
  - *Foreign Keys:* slip_id to BetSlip, bettor_id to TC_id in Bettor
  - Dependencies:
    - slip_id → bettor_id

**CREATE TABLE** BettorOwnsSlip (
bettor_id **INT NOT NULL**,
slip_id **INT NOT NULL,**
**PRIMARY KEY (**slip_id**),**
**FOREIGN KEY** (slip _id) **REFERENCES** Slip (slip _id)
**FOREIGN KEY** (bettor _id) **REFERENCES** Bettor (TC _id))

- BettorMakesBet (bettor_id, match_id, bet_type, bet_date, amount)
  - *Candidate Keys:* (bettor_id, match_id, bet_type)
  - *Primary Key:* (bettor_id, match_id, bet_type)
  - *Foreign Keys:* bettor_id to TC_id in Bettor, (match_id, bet_type) to Bet
  - Dependencies:
    - bettor_id, match_id, bet_type → bet_date, amount

**CREATE TABLE** BettorMakesBet (
bettor_id **INT NOT NULL**,
match_id **INT NOT NULL**,
bet_type **VARCHAR(20) NOT NULL**,
bet_date **DATE NOT NULL**,
amount **FLOAT,**
**PRIMARY KEY (**bettor_id, match_id, bet_type**),**
**FOREIGN KEY** (bettor _id) **REFERENCES** Bettor (TC _id),
**FOREIGN KEY** (match _id) **REFERENCES** Match (match _id),
**FOREIGN KEY** (match_id, bet_type) **REFERENCES** Bet (match_id, bet_type))

- BettorFollowsEditor (bettor_id, editor_id)
  - *Candidate Keys:* (bettor_id, editor_id)
  - *Primary Key:* (bettor_id, editor_id)
  - *Foreign Keys:* bettor_id to TC_id in Bettor, editor_id to TC_id in Editor

**CREATE TABLE** BettorFollowsEditor (
bettor_id **INT NOT NULL**,
editor_id **INT NOT NULL,**
**PRIMARY KEY(**bettor_id, editor_id**),**
**FOREIGN KEY** (bettor _id) **REFERENCES** Bettor (TC _id),
**FOREIGN KEY** (editor _id) **REFERENCES** Editor (editor _id))

- Lottery (lottery_id, lottery_start_date, lottery_end_date)
  - *Candidate Keys:* lottery_id
  - *Primary Key:* lottery_id
  - Dependencies:
    - lottery_id → lottery_start_date, lottery_end_date
    - lottery_start_date → lottery_id, lottery_end_date
    - lottery_end_date → lottery_id, lottery_start_date

**CREATE TABLE** Lottery (
lottery_id **INT NOT NULL**,
lottery_start_date **DATE NOT NULL**,
lottery_end_date **DATE NOT NULL,**
**PRIMARY KEY(**lottery_id**))**

- LotteryTicket (ticket_id, lottery_id, reward)
  - *Candidate Keys:* ticket_id
  - *Primary Key:* ticket_id
  - *Foreign Keys:* lottery _id to Lottery
  - Dependencies:
    - ticket_id → lottery_id, reward

**CREATE TABLE** LotteryTicket (
ticket_id **INT NOT NULL**,
lottery_id **INT NOT NULL**,
reward **FLOAT,**
**PRIMARY KEY(**ticket_id**)**
**FOREIGN KEY** (lottery _id) **REFERENCES** Lottery (lottery _id))

- BettorBoughtTicket (ticket_id, bettor_id, ticket_purchase_date)
  - *Candidate Keys:* ticket_id
  - *Primary Key:* ticket_id
  - *Foreign Keys:* bettor _id to TC_id in Bettor
  - Dependencies:
    - ticket_id → bettor_id, ticket_purchase_date

**CREATE TABLE** BettorBoughtTicket (
ticket_id **INT NOT NULL**,
bettor_id **INT NOT NULL**,
ticket_purchase_date **DATE,**
**PRIMARY KEY(**ticket_id**)**
**FOREIGN KEY** (ticket _id) **REFERENCES** LotteryTicket (ticket _id),

**FOREIGN KEY** (bettor _id) **REFERENCES** Bettor (TC _id))

- Comment (<u>comment_id,</u> TC_id, comment_date, contents)
  - *Candidate Keys:* comment_id
  - *Primary Key:* comment_id
  - Dependencies:
    - comment_id → TC_id, comment_date, contents

**CREATE TABLE** Comment (
comment_id **INT NOT NULL**,
TC_id **INT NOT NULL**,
comment_date **DATE**,
contents **VARCHAR(500),**
**PRIMARY KEY(** comment_id **)**
**FOREIGN KEY** (TC_id) **REFERENCES** GeneralUser (TC _id))

- CommentOnSlip (<u>comment_id, slip_id)</u>
  - *Candidate Keys:* (comment_id, slip_id)
  - *Primary Key:* (comment_id, slip_id)
  - *Foreign Keys:* comment_id to Comment, slip_id to BetSlip

**CREATE TABLE** CommentOnSlip (
comment_id **INT NOT NULL**,
slip_id **INT NOT NULL,**
**PRIMARY KEY(**comment_id, slip_id**),**
**FOREIGN KEY** (comment_id) **REFERENCES** Comment (comment _id),
**FOREIGN KEY** (slip_id) **REFERENCES** BetSlip (slip_id))

- CommentOnMatch (<u>comment_id, match_id)</u>
  - *Candidate Keys:* (comment_id, match_id)
  - *Primary Key:* (comment_id, match_id)
  - *Foreign Keys:* comment_id to Comment, match_id to Match

**CREATE TABLE** CommentOnSlip (
comment_id **INT NOT NULL**,
match_id **INT NOT NULL,**
**PRIMARY KEY(**comment_id, match_id**),**
**FOREIGN KEY** (comment_id) **REFERENCES** Comment (comment _id),
**FOREIGN KEY** (match_id) **REFERENCES** Match (match_id))

- Friend (<u>TC_id, friend_id</u>)
  - *Candidate Keys:* (TC_id, friend_id)
  - *Primary Key:* (TC_id, friend_id)
  - *Foreign Keys:* TC_id to Bettor, friend_id to TC_id in Bettor

**CREATE TABLE** Friend (
TC_id **INT NOT NULL**,
friend_id **INT NOT NULL,**
**PRIMARY KEY(**TC_id, friend_id**),**
**FOREIGN KEY** (TC_id) **REFERENCES** Bettor (TC _id),

**FOREIGN KEY** (friend _id) **REFERENCES** Bettor (TC _id))

- UserLikesSlip (TC_id, slip_id)
    - *Candidate Keys:* (TC_id, slip_id)
    - *Primary Key:* (TC_id, slip_id)
    - *Foreign Keys:* TC_id to GeneralUser, slip_id to BetSlip

**CREATE TABLE** UserLikesSlip (
TC_id **INT NOT NULL**,
slip_id **INT NOT NULL**,
**PRIMARY KEY(**TC_id, slip_id**),**
**FOREIGN KEY** (TC_id) **REFERENCES** GeneralUser (TC _id),
**FOREIGN KEY** (slip_id) **REFERENCES** BetSlip (slip_id))

- UserSharesSlip (TC_id, slip_id)
    - *Candidate Keys:* (TC_id, slip_id)
    - *Primary Key:* (TC_id, slip_id)
    - *Foreign Keys:* TC_id to GeneralUser, slip_id to BetSlip

**CREATE TABLE** UserSharesSlip (
TC_id **INT NOT NULL**,
slip_id **INT NOT NULL**,
**PRIMARY KEY(**TC_id, slip_id**),**
**FOREIGN KEY** (TC_id) **REFERENCES** GeneralUser (TC _id),
**FOREIGN KEY** (slip_id) **REFERENCES** BetSlip (slip_id))

- BetRemovedByAdmin (TC_id, match_id, bet_type, removal_date)
    - *Candidate Keys:* (TC_id, match_id, bet_type)
    - *Primary Key:* (bettor_id, match_id, bet_type)
    - *Foreign Keys:* TC_id to Admin, (match_id, bet_type) to Bet
    - Dependencies:
        - TC_id, match_id, bet_type → removal_date

**CREATE TABLE** BetRemovedByAdmin (
TC_id **INT NOT NULL**,
match_id **INT NOT NULL**,
bet_type **VARCHAR(20) NOT NULL**,
removal_date **DATE**,
**PRIMARY KEY(**TC_id, match_id, bet_type,
**FOREIGN KEY** (TC_id) **REFERENCES** Admin (TC _id),
**FOREIGN KEY** (match_id, bet_type) **REFERENCES** Bet (match_id, bet_type))

- BetChangedByAdmin (TC_id, match_id, bet_type, change_date, new_odd)
    - *Candidate Keys:* (TC_id, match_id, bet_type)
    - *Primary Key:* (bettor_id, match_id, bet_type)
    - *Foreign Keys:* TC_id to Admin, (match_id, bet_type) to Bet
    - Dependencies:
        - TC_id, match_id, bet_type → change_date, new_odd

**CREATE TABLE** BetRemovedByAdmin (

TC_id **INT NOT NULL**,
match_id **INT NOT NULL**,
bet_type **VARCHAR(20) NOT NULL**,
change_date **DATE**,
new_odd **FLOAT**,
**PRIMARY KEY(**TC_id, match_id, bet_type,
**FOREIGN KEY** (TC_id) **REFERENCES** Admin (TC _id),
**FOREIGN KEY** (match_id, bet_type) **REFERENCES** Bet (match_id, bet_type))

# 3. UI Design

## 3.1 Registration Page


*Figure 1: Registration Page as Bettor*

- *Registration as Bettor*

**INSERT INTO** User (TC_id, password, username, email, phone, date_of_birth)
**VALUES(**'id', 'pw', 'un', 'email_address', 'pn', 'dob'**);**

**INSERT INTO** Bettor (TC_id)
**VALUES(**'id'**);**

### 3.2 Login Page


*Figure 2: Login Page*

- *Login Info Check:* Returns NULL if there is no user with the given information.
  chosen_username = username that is given by the user.
  chosen_password = password that is given by the user
  **SELECT** * **FROM** User **WHERE** username = 'chosen_username' **AND** password = 'chosen_password'

- *Find the User Type:* Loops through all the user types with the given user information and finds the user type of the user with the given information. The extra information of the user will be stored in the session.

  **SELECT** * **FROM** Bettor **WHERE** username = 'chosen_username'
  **SELECT** * **FROM** Editor **WHERE** username = 'chosen_username'
  **SELECT** * **FROM** Admin **WHERE** username = 'chosen_username'

### 3.3 Match List Page

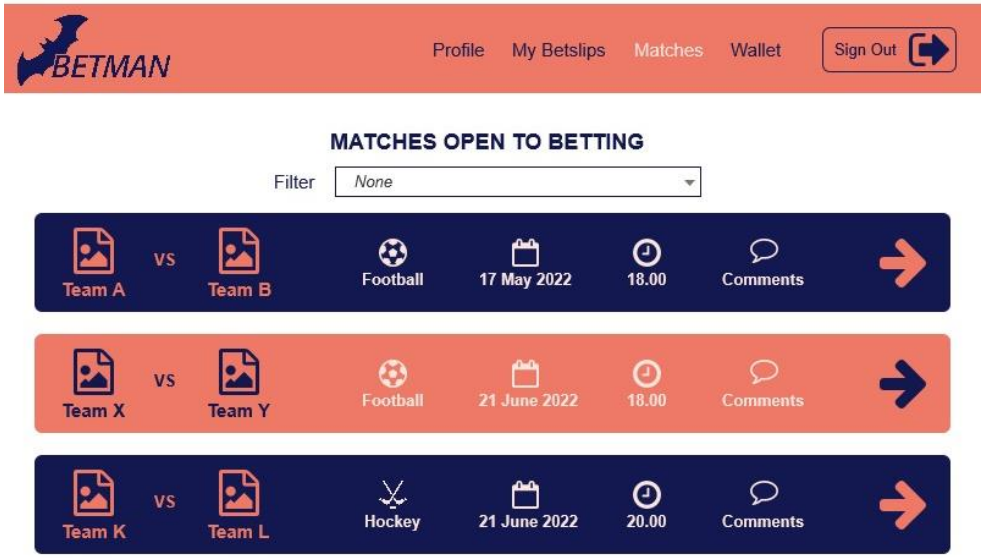



*Figure 3: Match List*

- *Match List:* It is assumed that there are only 2 teams involved in a match who are called host and guest.

**SELECT** M.match_id, M.match_type, M.match_date, T1.team_name, T2.team_name
**FROM** Plays **AS** P, Match **AS** M, Team **AS** T1, Team **AS** T2
**WHERE** P.match_id = M.match_id **AND** P.host_id = T1.team_id
**AND** P.guest_id = T2.team_id

### 3.4 Bet List Page

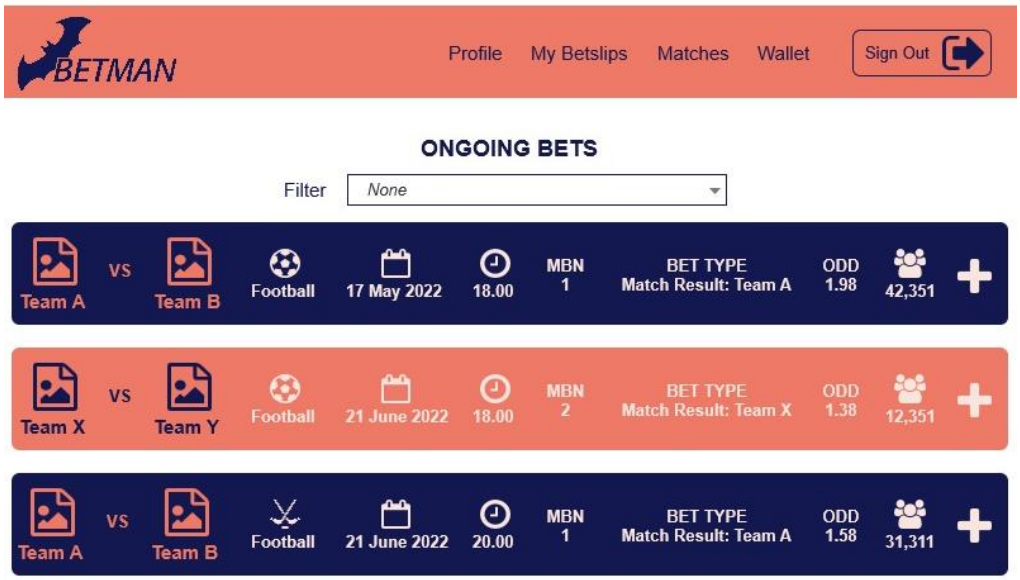



*Figure 4: Bet List*

- *Bet List*

**WITH**
bet_total_bettor_no (match_id, bet_type, bettor_no) **AS**
**(SELECT** match_id, bet_type, **COUNT (**bettor_id**)**
**FROM** BettorMakesBet
**GROUP BY** match_id, bet_type**),**
bet_ match_bettor_no (match_id, bet_type, bettor_no) **AS**
**(SELECT** match_id, bet_type, **COUNT (**bettor_id**)**
**FROM** BettorMakesBet
**GROUP BY** match_id**),**
selection_ratio (match_id, bet_type, ratio) **AS**
**(SELECT** B1. match_id, B1.bet_type, (B1.better_no / B2.better_no)
**FROM** bet_total_bettor_no **AS** B1, bet_ match_bettor_no **AS** B2
**WHERE** B1.match_id = B2.match_id**)**
**SELECT** match_id, bet_type, bet_name, MBN, odd, ratio
**FROM** Bet **NATURAL JOIN** selection_ratio
**WHERE (**match_id, bet_type) **NOT IN (SELECT** match_id, bet_type **FROM** BetRemovedByAdmin**)**

- *List of Bets According to the Bet Type:*
  chosen = type of the bet that was chosen from the bet type list by the user.

**WITH**
bet_total_bettor_no (match_id, bet_type, bettor_no) **AS**
**(SELECT** match_id, bet_type, **COUNT (**bettor_id**)**
**FROM** BettorMakesBet
**GROUP BY** match_id, bet_type**),**
bet_ match_bettor_no (match_id, bet_type, bettor_no) **AS**
**(SELECT** match_id, bet_type, **COUNT (**bettor_id**)**
**FROM** BettorMakesBet
**GROUP BY** match_id**),**
selection_ratio (match_id, bet_type, ratio) **AS**
**(SELECT** B1. match_id, B1.bet_type, (B1.better_no / B2.better_no)
**FROM** bet_total_bettor_no **AS** B1, bet_ match_bettor_no **AS** B2
**WHERE** B1.match_id = B2.match_id**)**
**SELECT** match_id, bet_type, bet_name, MBN, odd, ratio
**FROM** Bet **NATURAL JOIN** selection_ratio
**WHERE** bet_type = 'chosen' **AND (**match_id, bet_type) **NOT IN (SELECT** match_id, bet_type **FROM** BetRemovedByAdmin**)**

- *List of Bets According to the Match Date:*
  chosen = date of the match that the bet is based on.

**WITH**
bet_total_bettor_no (match_id, bet_type, bettor_no) **AS**
**(SELECT** match_id, bet_type, **COUNT (**bettor_id**)**

```
                    FROM BettorMakesBet
                    GROUP BY match_id, bet_type),
    bet_ match_bettor_no (match_id, bet_type, bettor_no) AS
                    (SELECT match_id, bet_type, COUNT (bettor_id)
                    FROM BettorMakesBet
                    GROUP BY match_id),
    selection_ratio (match_id, bet_type, ratio) AS
                    (SELECT B1. match_id, B1.bet_type, (B1.better_no / B2.better_no)
                    FROM bet_total_bettor_no AS B1, bet_ match_bettor_no AS B2
                    WHERE B1.match_id = B2.match_id)
SELECT match_id, bet_type, bet_name, MBN, odd, ratio
FROM Bet NATURAL JOIN selection_ratio
WHERE match_date = 'chosen' AND (match_id, bet_type) NOT IN (SELECT match_id,
bet_type FROM BetRemovedByAdmin)
```

- *List of Bets According to the Match Type:*

  chosen = Type of the match that the bet is based on.

```
WITH
    bet_total_bettor_no (match_id, bet_type, bettor_no) AS
                    (SELECT match_id, bet_type, COUNT (bettor_id)
                    FROM BettorMakesBet
                    GROUP BY match_id, bet_type),
    bet_ match_bettor_no (match_id, bet_type, bettor_no) AS
                    (SELECT match_id, bet_type, COUNT (bettor_id)
                    FROM BettorMakesBet
                    GROUP BY match_id),
    selection_ratio (match_id, bet_type, ratio) AS
                    (SELECT B1. match_id, B1.bet_type, (B1.better_no / B2.better_no)
                    FROM bet_total_bettor_no AS B1, bet_ match_bettor_no AS B2
                    WHERE B1.match_id = B2.match_id)
SELECT match_id, bet_type, bet_name, MBN, odd, ratio
FROM Bet NATURAL JOIN selection_ratio
WHERE match_type = 'chosen' AND (match_id, bet_type) NOT IN (SELECT match_id,
bet_type FROM BetRemovedByAdmin)
```

### 3.5 Bet Slip List

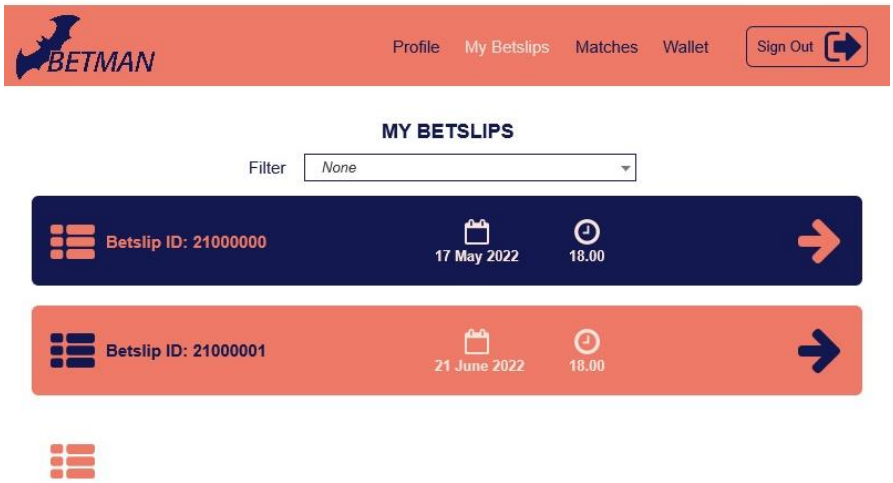



*Figure 5: Bet Slip List*

### 3.6 Bet Slip

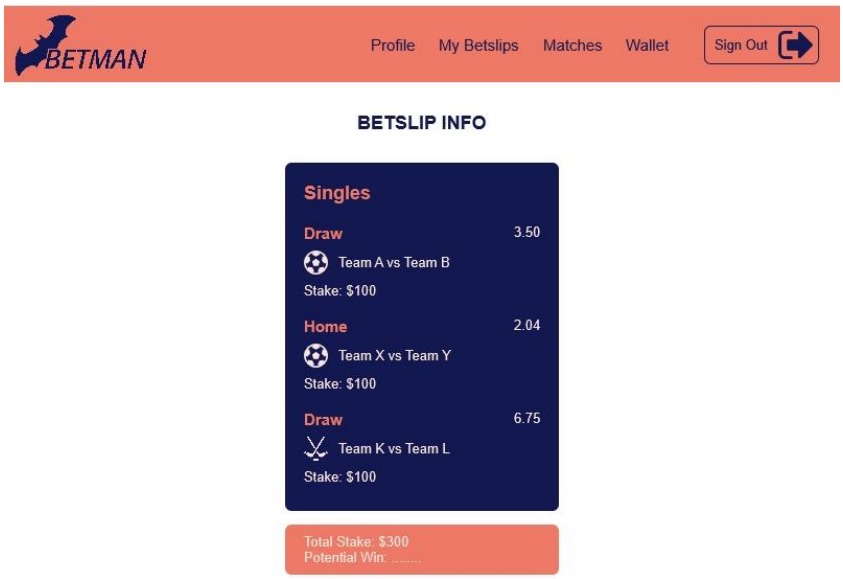



*Figure 6: Bet Slip Info*

- *Add Bet to Bet Slip of the User:* The user information would have been saved previously as session. No money will be entered initally

  chosen_type = ID of the bet that was chosen from the Bet List which will be retrieved from the query above.

  chosen_match = ID of the match that was chosen from the Match List.

  current_slip = id of the slip that will have the new bet.

**INSERT INTO** SlipHasBet
**VALUES (‘**current_slip’, ‘chosen_match’, ‘chosen_type’**);**

**INSERT INTO** BettorMakesBet
**VALUES (‘**user_id’, ‘chosen_match’, ‘chosen_type’, ‘current_date’, ‘0’**);**

**UPDATE** BetSlip
**SET** no_of_bets = no_of_bets + ‘1’
**WHERE** slip_id = ‘current_slip’;

**SELECT** slip_id, bet_name, MBN, bet_type, odd, bet_date
**FROM** Bet **NATURAL JOIN** SlipHasBet
**WHERE** bet_id = ‘chosen’ **AND** slip_id = ‘current_slip’;

- *Place bet amount on the Bet Slip:* User specifies the money that will be placed on the specified bet.

  chosen_bettor = ID of the bettor

chosen_type = type of the bet
chosen_match = ID of the match the bet is made on
chosen_slip = ID of the current betslip of the bettor

**UPDATE** BettorMakesBet
**SET** amount = ‘chosen_amount’
**WHERE** bet_type = ‘chosen_type’ **AND** match_id = ‘chosen_match’

**UPDATE** BetSlip
**SET** totalAmount = totalAmount + ‘chosen_amount’
**WHERE** slip_id = ‘chosen_slip’

- *Check if the bettor has enough money and MBN is reached. Then, update the wallet after making the bet:*
  chosen_bettor = ID of the bettor that is making the bet.
  chosen_slip = ID of the bet slip

**UPDATE** Wallet
**SET** real_currency = real_currency –
(**SELECT** totalAmount
**FROM** BetSlip
**WHERE** slip_id = ‘chosen_slip’)
**WHERE** TC_id = chosen_bettor
**AND** real_currency >=
(**SELECT** totalAmount
**FROM** BetSlip
**WHERE** slip_id = chosen_slip)
**AND** no_of_bets <=
(**SELECT MAX**(MBN)
**FROM** SlipHasBet **NATURAL JOIN** Bet
**WHERE** slip_id = ‘chosen_slip’
**GROUP BY** MBN)

### 3.7 Bet Slip Info with Comments

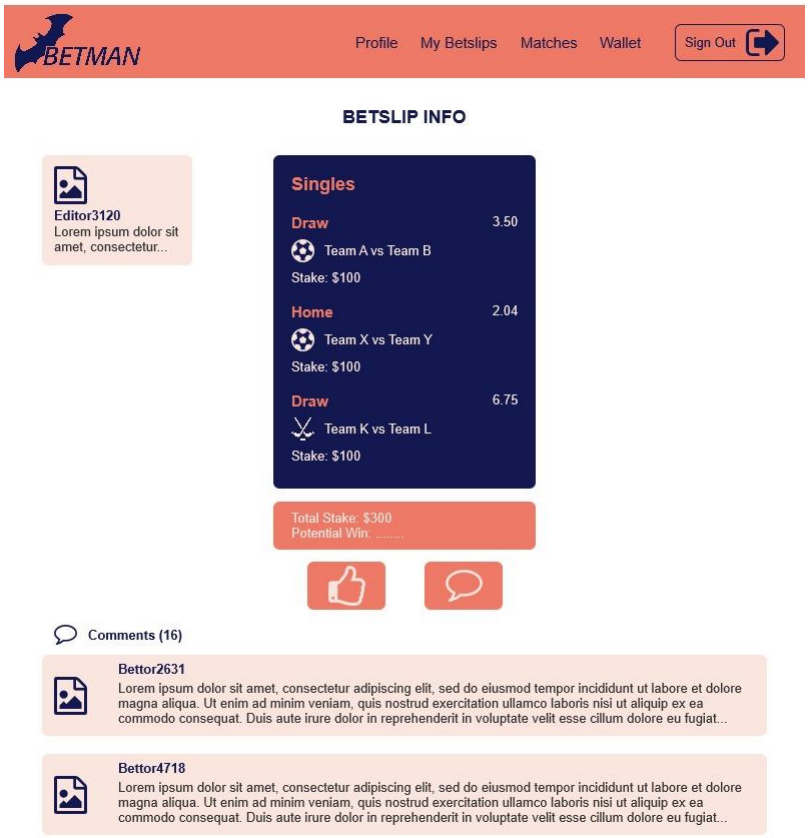



*Figure 7: Bet Slip with Comments*

- *Users comment on and likes their friends' bet slips:* The user can only see the slips of their friends and the editors they are following so there is no need for checking while commenting.

  cid = Randomly generated comment id.

  random_content = Content of the comment done by user to be retrieved from textbox.

  user_id = id of the user in current session.

  friend_slip = id of the slip owned by friend.

**INSERT INTO** comment
**VALUES (**'cid', 'user_id', 'current_date', 'random_content'**);**

**INSERT INTO** CommentOnSlip
**SELECT** comment_id, 'friend_slip'
**FROM** comment
**WHERE** comment_id = 'cid'

**INSERT INTO** UserLikesSlip
**VALUES (**'user_id', 'friend_slip')

### 3.8 Match Slip Info with Comments

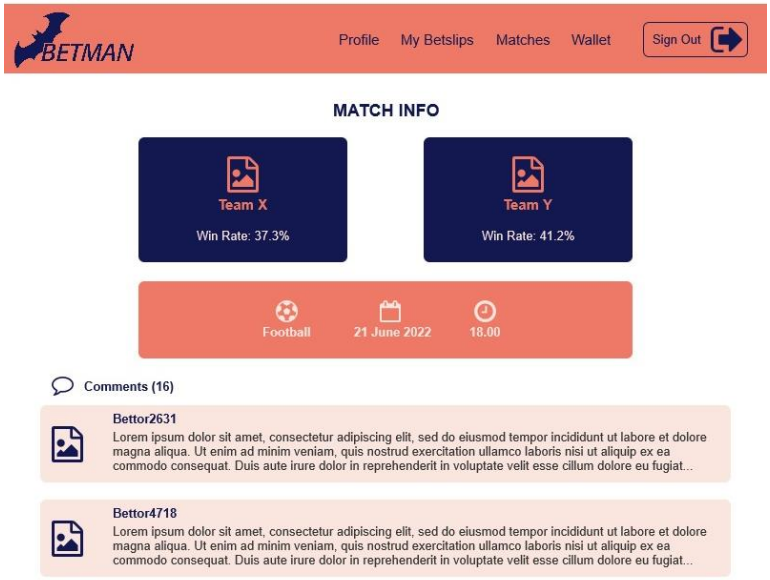



*Figure 8: Match Info*

### 3.9 Wallet Page of User

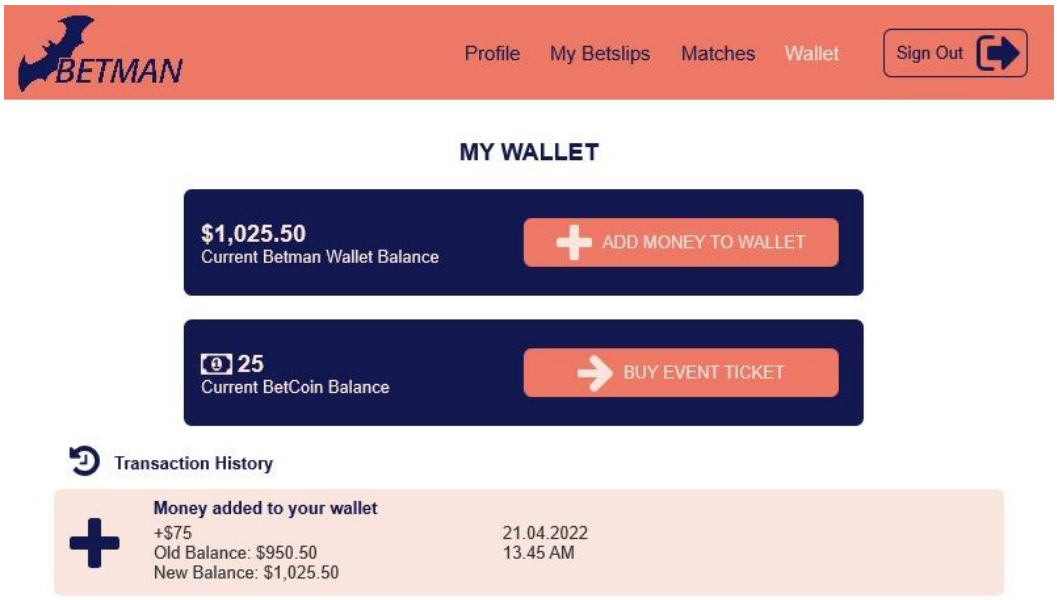



*Figure 9: Wallet*

### 3.10 Main Feed

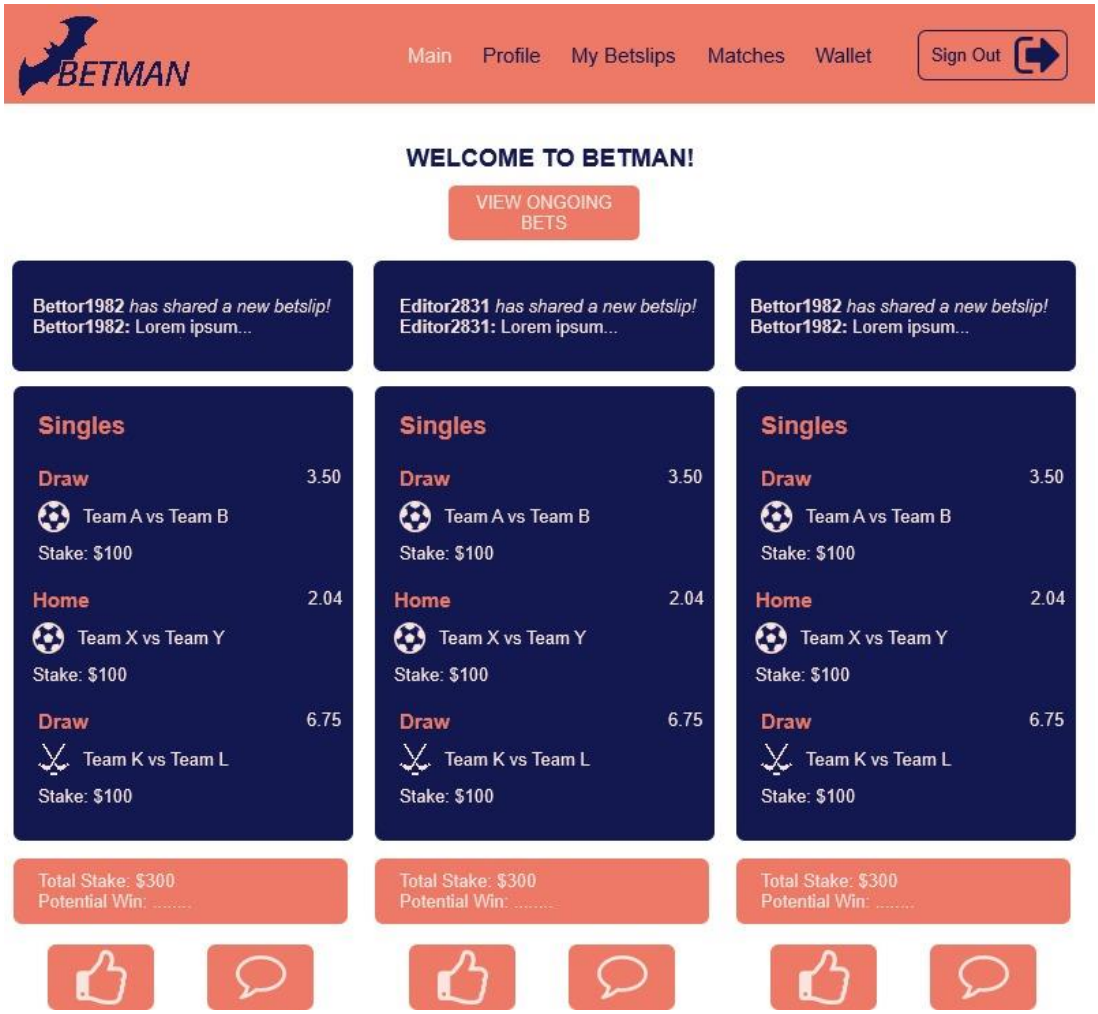



*Figure 10: Main Feed*

### 3.11 Bet List from Admin's Point of View

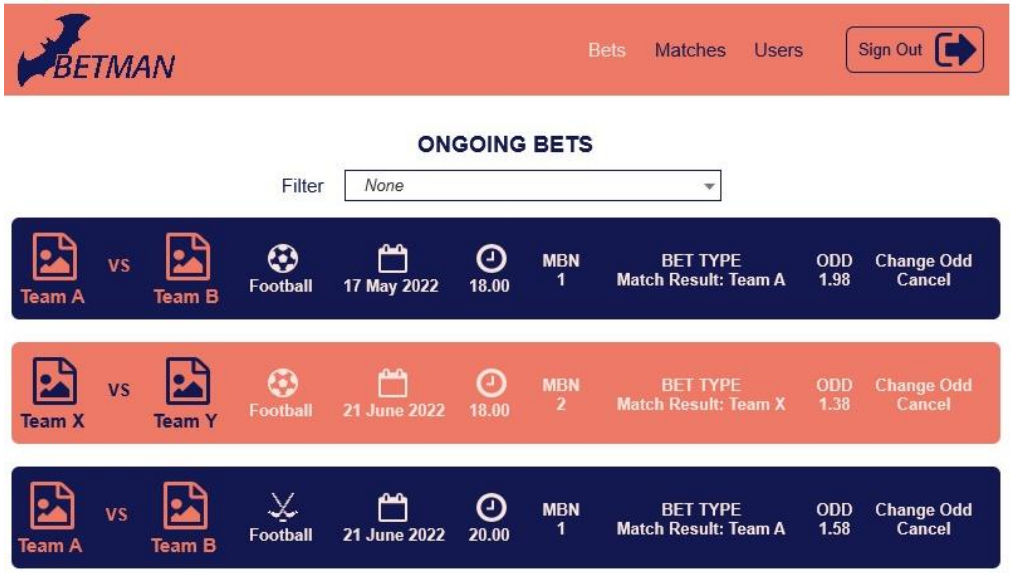



*Figure 11: Admin's Bet List*

- *Change the Odd of a Specific Bet:*

  chosen_type = type of the chosen bet.
  chosen_match = ID of the match of the chosen bet.
  chosen_odd = the new odd value of the chosen bet.

**UPDATE** Bet
**SET** odd = 'chosen_odd'
**WHERE** bet_type = 'chosen_type' **AND** match_id = 'chosen_match'

- *Cancel a Specific Bet:*

  chosen_type = type of the chosen bet.
  chosen_match = ID of the match of the chosen bet.

**INSERT INTO** BetRemovedByAdmin
**VALUES (**'user_id', 'chosen_match', 'chosen_type', 'current_date'**)**

### 3.12 Lottery Ticket List

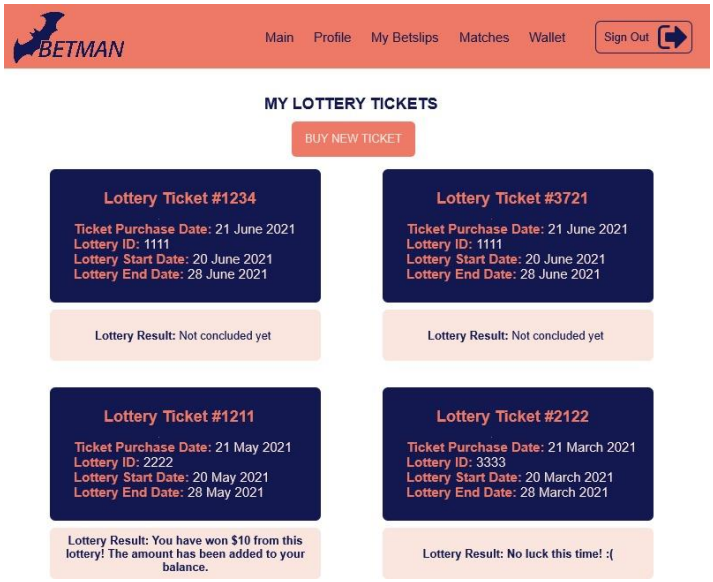



*Figure 12: Lottery Tickets*

- *Buy Lottery Ticket:*

  chosen_lottery = ID of the lottery of the lottery ticket.
  chosen_bettor = ID of the bettor that is buying the ticket.

**INSERT INTO** BettorBoughtTicket
**VALUES** (‘chosen_bettor’, ‘chosen_lottery’)

**UPDATE** Wallet
**SET** app_currency = app_currency - ‘ticket_price’
**WHERE** TC_id = ‘chosen_bettor’

- *Update the Wallet According to the Lottery Result:*

  reward = earned reward from the lottery ticket.
  chosen_bettor = ID of the bettor that has bought the ticket.

**UPDATE** Wallet
**SET** real_currency = real_currency + ‘reward’
**WHERE** TC_id = ‘chosen_bettor’

## 4. Implementation

We are going to use MySQL for database management and PHP for backend development. The application will be tested on Dijkstra servers of Bilkent University and on localhost through XAMPP. The UI was be designed with the mockup tool Axure. After the design phase, the GUI will be implemented using React.

## 5. Website

In the following, the website where we will be sharing our project documents can be found: https://znurcavcar.github.io/ProjectCS353/