



Bilkent University

Department of Computer Engineering

---

# CS353 TERM PROJECT

## Design Report

Zeynepnur Cavcar, Ece Teker, Yaren Durgun, Selen Görgün

Instructor: Hamdi Dibeklioglu

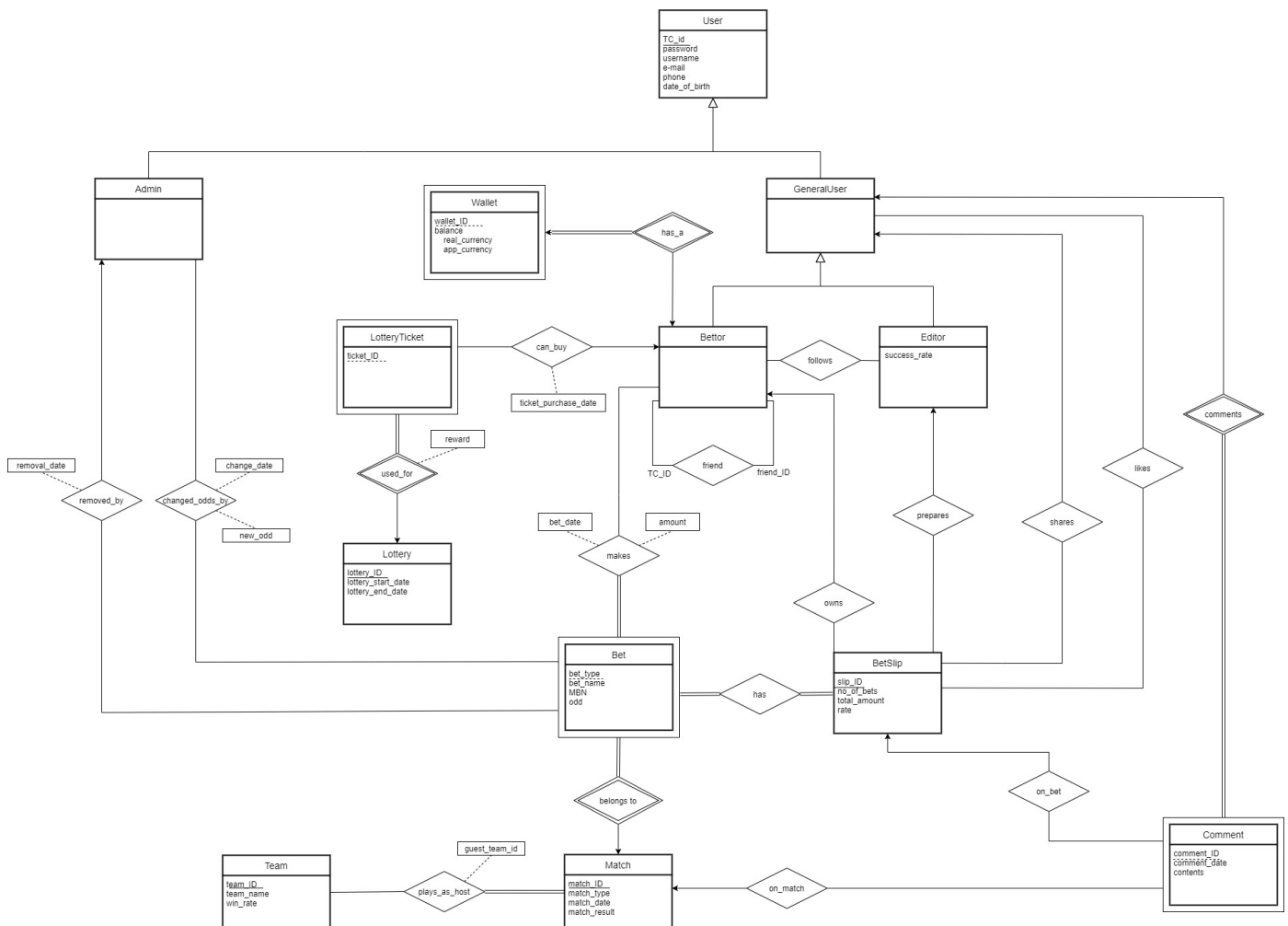
Teaching Assistant: Mustafa Can Cavdar

April 8, 2022

## Table of Contents

1. Revised ER Diagram	3
2. Relations	3
3. UI Design	11
3.1 Registration Page	11
3.2 Login Page	12
3.3 Match List Page	13
3.4 Bet List Page	13
3.5 Bet Slip List	16
3.6 Bet Slip	17
3.7 Bet Slip Info with Comments	19
3.8 Match Slip Info with Comments	20
3.9 Wallet Page of User	20
3.10 Main Feed	21
3.11 Bet List from Admin's Point of View	22
3.12 Lottery Ticket List	23
4. Implementation	23

## 1. Revised ER Diagram



## 2. Relations

All relations are in 3NF format.

- User (TC\_id, 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 (slip\_id, 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 (slip\_id, 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 (match\_id, bet\_type, 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 (slip\_id, match\_id, bet\_type)
  - *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 (comment\_id, 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 (comment\_id, slip\_id)
  - *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 (comment\_id, match\_id)
  - *Candidate Keys:* (comment\_id, match\_id)
  - *Primary Key:* (comment\_id, match\_id)
  - *Foreign Keys:* comment\_id to Comment, match\_id to Match

```
CREATE TABLE CommentOnMatch (  
    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 (TC\_id, friend\_id)
  - *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 DECIMAL(10,2) NOT NULL,  
    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))
```

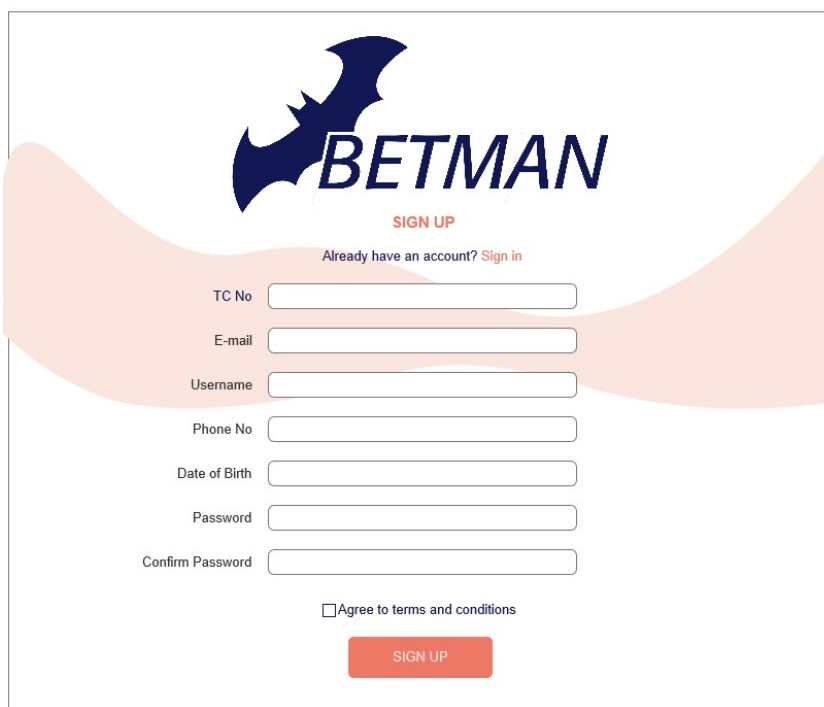
```

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



The image shows a registration page for a website called "BETMAN". The page has a white background with a blue bat logo and the word "BETMAN" in blue. Below the logo, there is a "SIGN UP" link in red. A link "Already have an account? Sign in" is also present. The registration form consists of several input fields: "TC No", "E-mail", "Username", "Phone No", "Date of Birth", "Password", and "Confirm Password". Below these fields is a checkbox labeled "Agree to terms and conditions". At the bottom of the form is a red "SIGN UP" button.

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



**BETMAN**

**SIGN IN**

Don't have an account yet? [Sign Up](#)

E-mail

Password

☐ Remember Me

**SIGN IN**

[Forgot your password?](#)

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.bettor_no / B2.bettor_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.bettor_no / B2.bettor_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

**BETMAN** Profile My Betslips Matches Wallet Sign Out

**MY BETSLIPS**

Filter None

 Betslip ID: 21000000	 17 May 2022	 18.00	
 Betslip ID: 21000001	 21 June 2022	 18.00	




Figure 5: Bet Slip List



### 3.6 Bet Slip

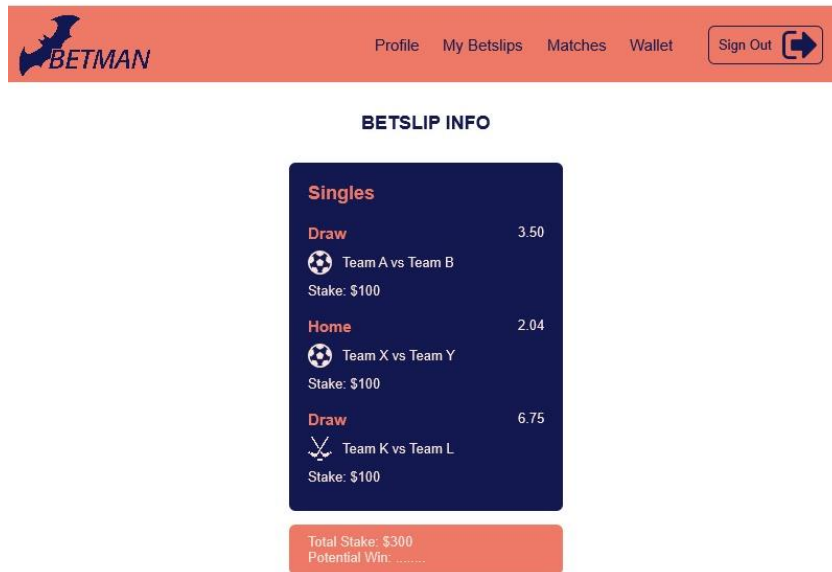


Figure 6: Bet Slip Info

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

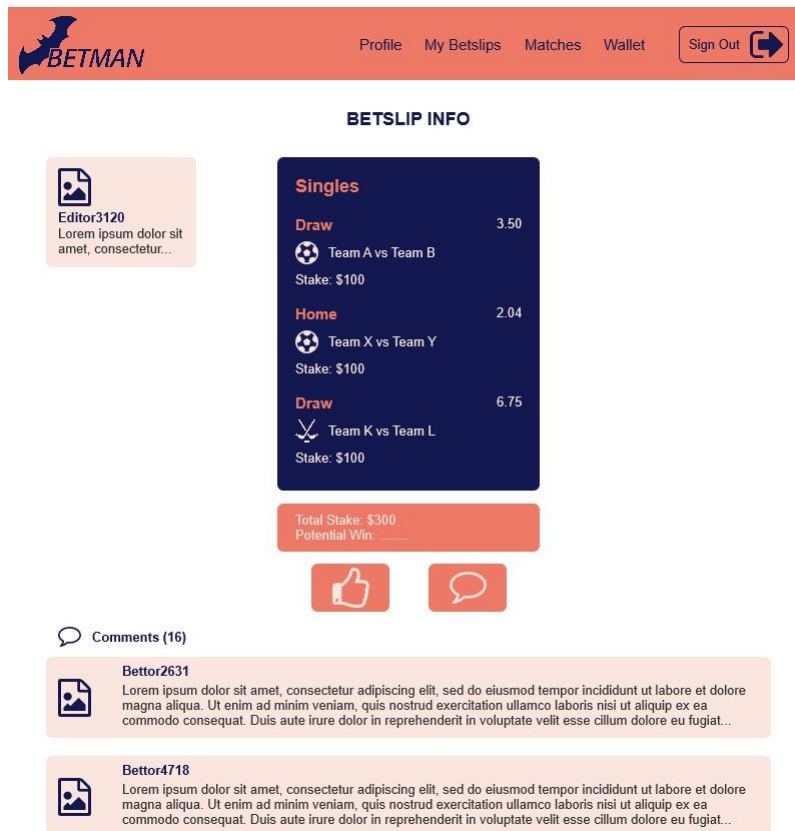


Figure 7: Bet Slip with Comments

- *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


ProfileMy BetslipsMatchesWalletSign Out


#### MATCH INFO

  
Team X  
Win Rate: 37.3%

  
Team Y  
Win Rate: 41.2%

Football21 June 202218.00

Comments (16)

Bettor2631  
Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat...



Bettor4718  
Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat...


Figure 8: Match Info


### 3.9 Wallet Page of User


ProfileMy BetslipsMatchesWalletSign Out


#### MY WALLET


**\$1,025.50**  
Current Betman Wallet Balance

ADD MONEY TO WALLET

**25**  
Current BetCoin Balance

BUY EVENT TICKET


Transaction History

**Money added to your wallet**  
+\$75  
Old Balance: \$950.50  
New Balance: \$1,025.50

21.04.2022  
13.45 AM

Figure 9: Wallet

### 3.10 Main Feed



[Main](#) [Profile](#) [My Betslips](#) [Matches](#) [Wallet](#) [Sign Out](#)


WELCOME TO BETMAN!

VIEW ONGOING BETS


Bettor1982 has shared a new betslip!  
Bettor1982: Lorem ipsum...

Singles


Draw3.50

 Team A vs Team B  
Stake: \$100


Home2.04


 Team X vs Team Y  
Stake: \$100

Draw6.75

 Team K vs Team L  
Stake: \$100

Total Stake: \$300  
Potential Win: .....







Editor2831 has shared a new betslip!  
Editor2831: Lorem ipsum...

Singles


Draw3.50

 Team A vs Team B  
Stake: \$100


Home2.04


 Team X vs Team Y  
Stake: \$100

Draw6.75

 Team K vs Team L  
Stake: \$100

Total Stake: \$300  
Potential Win: .....







Bettor1982 has shared a new betslip!  
Bettor1982: Lorem ipsum...

Singles


Draw3.50

 Team A vs Team B  
Stake: \$100


Home2.04

 Team X vs Team Y  
Stake: \$100

Draw6.75

 Team K vs Team L  
Stake: \$100

Total Stake: \$300  
Potential Win: .....






Figure 10: Main Feed

21

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


















									
<a href="#">Bets</a> <a href="#">Matches</a> <a href="#">Users</a> <a href="#">Sign Out</a> 									
ONGOING BETS									
Filter <span>None</span>									
	vs					MBN 1	BET TYPE Match Result: Team A	ODD 1.98	Change Odd Cancel
Team A		Team B	Football	17 May 2022	18.00				
	vs					MBN 2	BET TYPE Match Result: Team X	ODD 1.38	Change Odd Cancel
Team X		Team Y	Football	21 June 2022	18.00				
	vs					MBN 1	BET TYPE Match Result: Team A	ODD 1.58	Change Odd Cancel
Team A		Team B	Football	21 June 2022	20.00				

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

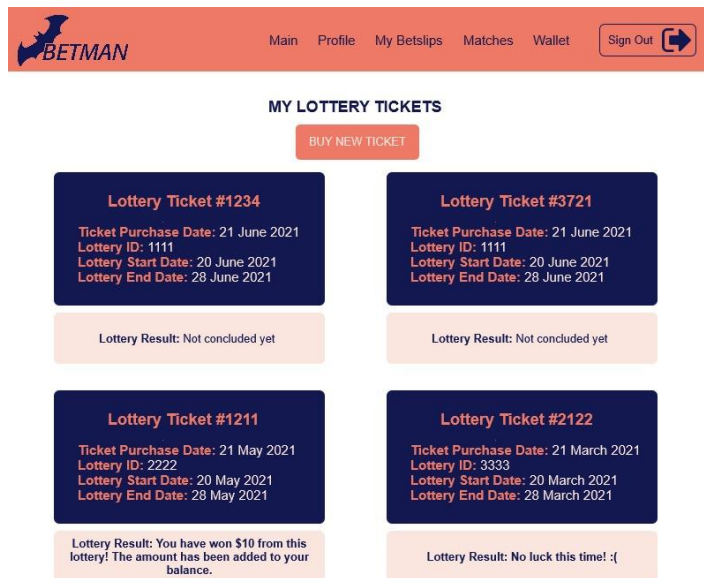


Figure 12: Lottery Tickets

- *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 designed with the mockup tool Axure. After the design phase, the GUI will be implemented using React.