Group Project : Online Voting System

Project plan: Create a secure platform for user registration, candidate information, and casting votes. Includes key features of Joins for candidates, voters, and voting records; enforced one-time voting per user;secure login for voting; transactions for vote submission.

Entities

```
CREATE TABLE users (
    id INT AUTO_INCREMENT PRIMARY KEY,
    username VARCHAR(50) UNIQUE,
    password VARCHAR(255) NOT NULL,
    email VARCHAR(100) UNIQUE,
    registration_date TIMESTAMP DEFAULT CURRENT_TIMESTAMP
);
```

This table stores information about registered users

`id`: A unique identifier for each user.

`username`: A unique username for the user.

`password`: A hashed password for the user.

`email`: A unique email address for the user.

`registration_date`: The timestamp when the user registered.

```
CREATE TABLE candidates (
    id INT AUTO_INCREMENT PRIMARY KEY,
    name VARCHAR(100) NOT NULL,
    description TEXT,
    party VARCHAR(50),
    election_date TIMESTAMP DEFAULT CURRENT_TIMESTAMP
);
```

This table stores information about the candidates running in the election.

`id`: A unique identifier for each candidate.

`name`: The name of the candidate.

`description`: A brief description of the candidate.

`party`: The political party the candidate represents.

`election_date`: The timestamp when the candidate entered the election.

```
CREATE TABLE votes (
    id INT AUTO_INCREMENT PRIMARY KEY,
    user_id INT,
    candidate_id INT,
    vote_time TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
    FOREIGN KEY (user_id) REFERENCES users(id) ON DELETE CASCADE,
    FOREIGN KEY (candidate_id) REFERENCES candidates(id) ON DELETE CASCADE,
    UNIQUE(user_id)
);
```

This table stores vote information, linking users to the candidates they voted for.

`id`: A unique identifier for each vote.

`user_id`: A reference to the user who cast the vote.

`candidate_id`: A reference to the candidate who received the vote.

`vote_time`: The timestamp when the vote was cast.

```
CREATE TABLE logs (
    id INT AUTO_INCREMENT PRIMARY KEY,
    user_id INT,
    action VARCHAR(255),
    ip_address VARCHAR(45),
    user_agent VARCHAR(255),
    timestamp TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
    FOREIGN KEY (user_id) REFERENCES users(id) ON DELETE SET NULL
);
```

This table tracks user actions, such as logging in or voting.
`id`: A unique identifier for each log entry.

`user_id`: A reference to the user who performed the action.

`action`: The action performed by the user (e.g., logged in, voted).

`ip_address`: The IP address of the user when the action was performed.

`user_agent`: The user agent string identifying the browser or device.

`timestamp`: The timestamp when the action occurred.

Technical Details / Database design

```
SELECT c.name, COUNT(v.id) AS total_votes
FROM candidates c
LEFT JOIN votes v ON c.id = v.candidate_id
GROUP BY c.id;
SELECT v.id AS vote_id, u.username, c.name AS candidate_name, v.vote_time
FROM votes v
JOIN users u ON v.user_id = u.id
JOIN candidates c ON v.candidate_id = c.id;
```

Left Join on votes and candidates to count the amount of votes casted for each candidate
Join on users an voter id to ensure one-time voting
Join attaching candidate to vote

We can use the database to:

- Register users.

- Track votes for candidates.

- Log user actions such as voting.

Source Files

Onlinevotingsystem2.sql : Creation of database,joins and inputs of candidates
db.php : Established database connection
Login.php: Page to login with registered credentials
Register.php : Page to register creating a username,password and providing an email
Vote.php : Page for registered users to cast their vote and enforces one time voting rule
Results.php: Displays vote count for each candidate (only visible to admin)