DSA Hackathon

TEAM NUMBER:17

TEAM MEMBERS:

SHRIYA KONDURU: PES2UG22CS546
SHRIYANS GANDHI: PES2UG22CS547
SINCHANA C: PES2UG22CS555

Abstract:

The Online Voting System is a comprehensive solution designed to address the critical challenges of security, accessibility, and usability in the domain of online voting. The project focuses on implementing data structures to safeguard voter information, candidate details, and vote tallies, ensuring the integrity and confidentiality of the electoral process.

Key Features:

- **Two-Factor Authentication**: Integrating a two-factor authentication mechanism to enhance user verification, reducing the risk of unauthorized access.
- **Checking for duplicates:** Ensuring proper voting takes place by eliminating the possibility of duplicate voter ids and thus multiple votes under the same name.
- **Password Checking:** Users can put their own password to secure their credentials. The password is checked when the voter wants to vote.
- **Secure Password Handling**: Implementing secure password storage techniques like hashing, to protect user credentials from potential breaches.

- **Candidate Details:** Implementing secure data structures to store candidate information, preventing tampering and maintaining the accuracy of candidate profiles.
- **Efficient Vote Counting System**: Using seperate datafields to store raw vote tallies to ensure fast displaying of the votes

Problem Solving Approach

- **Understand the requirements:** We read the project requirements carefully and understood the features that need to be implemented. We identified the data structures that need to be used to store voter information, candidate details, and vote tallies.
- **Design the system:** We created a high-level design of the system, including the user interface, data structures, and algorithms. We decided to use linked lists to store structures of the candidates and voters. Each structure's data fields store information regarding the voter or candidate. We also decided to implement various features to enhance the security and reliability of the voting system, such as two-factor authentication, checking for duplicates, password checking, secure password storage techniques like hashing, and an efficient vote counting system.
- **Implement the system:** We wrote the code for the system, following the design. We used linked lists to store structures of the candidates and voters. Each structure's data fields store information regarding the voter or candidate. We implemented various features to enhance the security and reliability of the voting system, such as two-factor authentication, checking for duplicates, password checking, secure password storage techniques like hashing, and an efficient vote counting system.
- **Test the system:** We tested the system thoroughly to ensure that it works as expected. We tested all the features and edge cases to identify and fix any bugs or issues.

Data Structures Used:

Created two structures.
Voter structure holds essential voter information and tracks whether the voter has already cast a vote.
Candidate structure stores candidate details along with the number of votes they received.

These structures are stored in their own independent singly linked lists for easy traversal in the information.

Problem-Solving Approach:

1. Data Structures (DS) and Justification:
    - Linked List for Voters:
      Justification: Voter data is efficiently stored via a linked list. It enables simple navigation and dynamic voter registration.

    - Linked List of Candidates:
      Justification: A linked list is utilised for candidates, just like it is for voters, enabling dynamic candidate registration and result computation.

Abstract Data Type(ADT):

Voter ADT:

- Properties:
    - Voter username, password, randomPasscode, and voted status.
- Operations:

    - registerVoter: Register a new voter and add them to the linked list.
    - authenticateVoter: Authenticate a voter based on username, password, and randomPasscode.
    - castVote: Allow a registered voter to cast a vote.

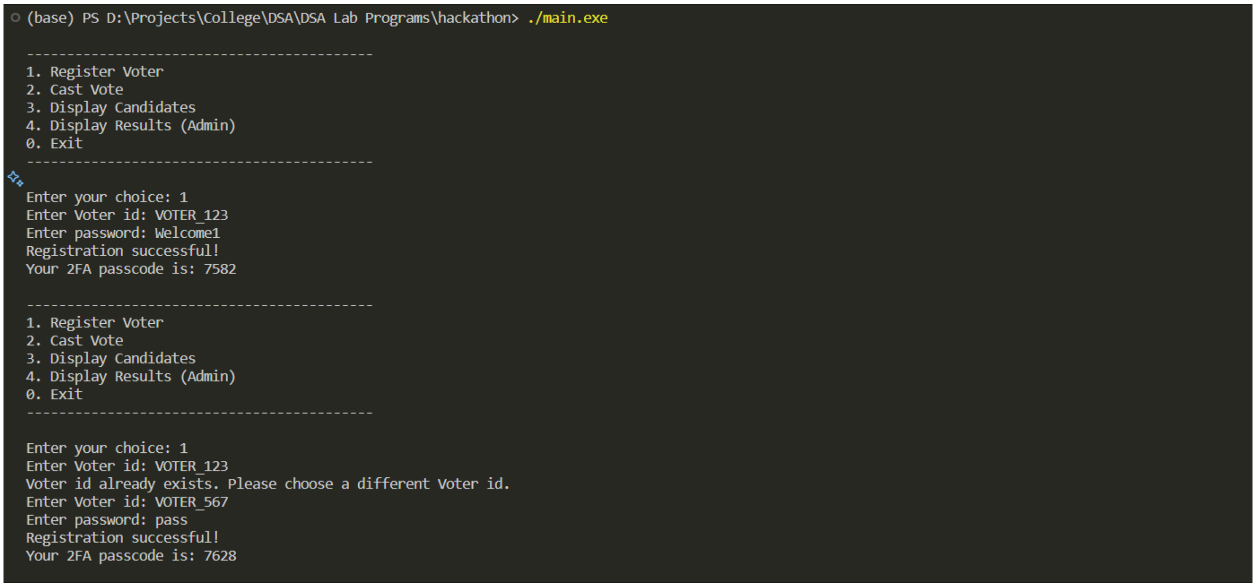
Candidate ADT:

- Properties:
    - Candidate name and votes received.
- Operations:
    - intialiseCandidates: Initialize the list of candidates.
    - displayCandidates: Display the list of candidates.
    - displayResults: Display the election results.

Assumptions made:

We assumed candidates as Candidates 1-5. Users were also allowed to create an voter id which was assumed to be valid.

Images:



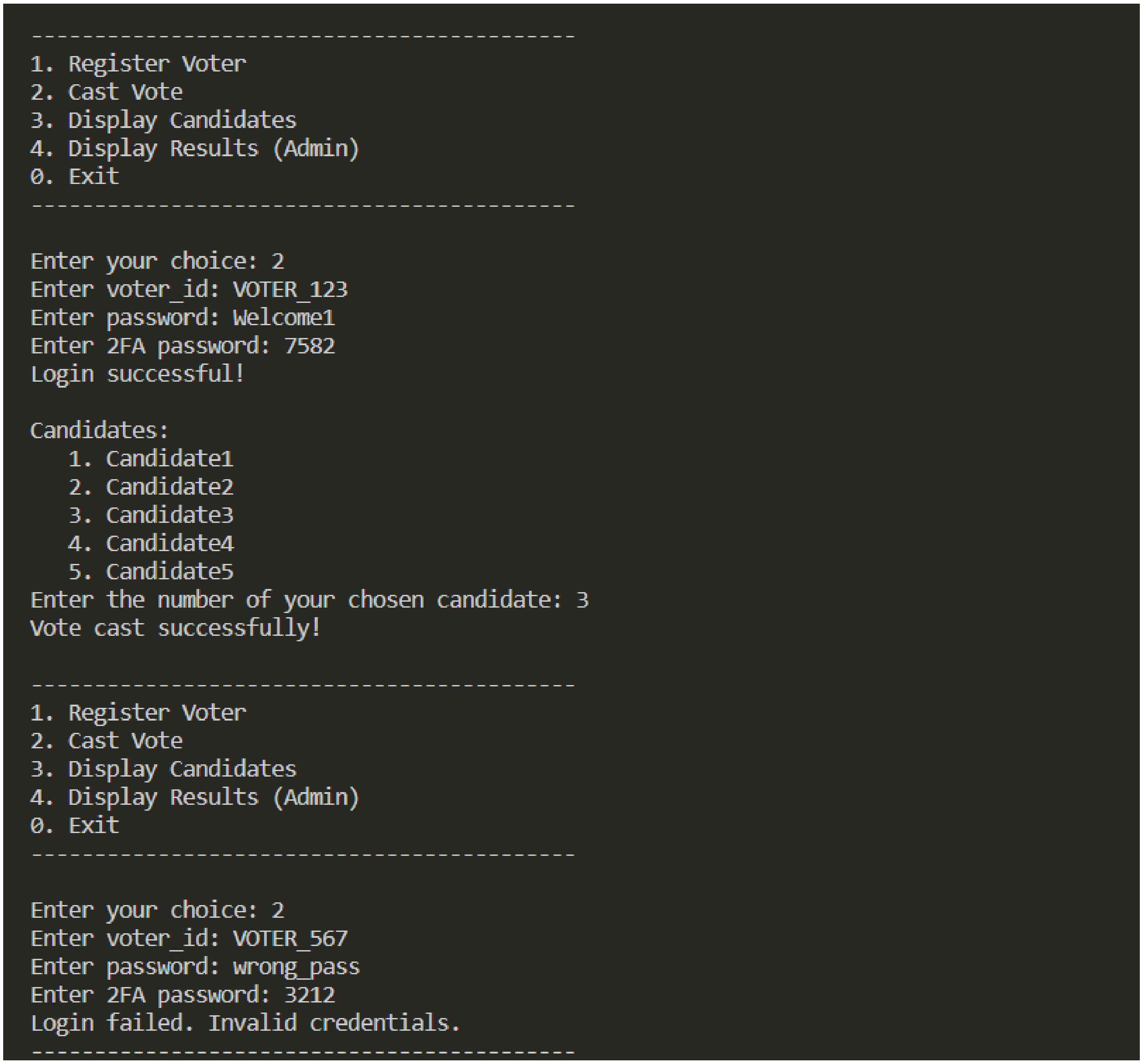
```
(base) PS D:\Projects\College\DSA\DSA Lab Programs\hackathon> ./main.exe

  -------------------------------------------
  1. Register Voter
  2. Cast Vote
  3. Display Candidates
  4. Display Results (Admin)
  0. Exit
  -------------------------------------------

  Enter your choice: 1
  Enter Voter id: VOTER_123
  Enter password: Welcome1
  Registration successful!
  Your 2FA passcode is: 7582

  -------------------------------------------
  1. Register Voter
  2. Cast Vote
  3. Display Candidates
  4. Display Results (Admin)
  0. Exit
  -------------------------------------------

  Enter your choice: 1
  Enter Voter id: VOTER_123
  Voter id already exists. Please choose a different Voter id.
  Enter Voter id: VOTER_567
  Enter password: pass
  Registration successful!
  Your 2FA passcode is: 7628
```



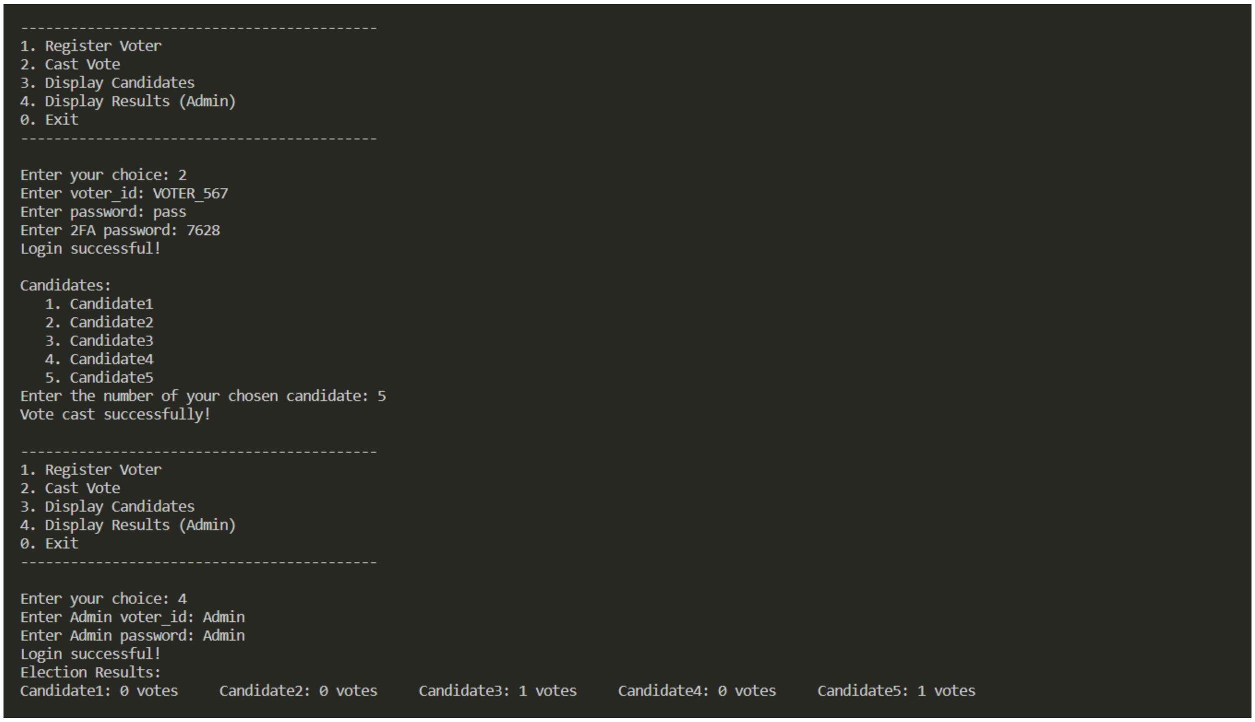
```
-------------------------------------------
1. Register Voter
2. Cast Vote
3. Display Candidates
4. Display Results (Admin)
0. Exit
-------------------------------------------

Enter your choice: 2
Enter voter_id: VOTER_123
Enter password: Welcome1
Enter 2FA password: 7582
Login successful!

Candidates:
   1. Candidate1
   2. Candidate2
   3. Candidate3
   4. Candidate4
   5. Candidate5
Enter the number of your chosen candidate: 3
Vote cast successfully!

-------------------------------------------
1. Register Voter
2. Cast Vote
3. Display Candidates
4. Display Results (Admin)
0. Exit
-------------------------------------------

Enter your choice: 2
Enter voter_id: VOTER_567
Enter password: wrong_pass
Enter 2FA password: 3212
Login failed. Invalid credentials.
-------------------------------------------
```



```
-------------------------------------------
1. Register Voter
2. Cast Vote
3. Display Candidates
4. Display Results (Admin)
0. Exit
-------------------------------------------

Enter your choice: 2
Enter voter_id: VOTER_567
Enter password: pass
Enter 2FA password: 7628
Login successful!

Candidates:
   1. Candidate1
   2. Candidate2
   3. Candidate3
   4. Candidate4
   5. Candidate5
Enter the number of your chosen candidate: 5
Vote cast successfully!

-------------------------------------------
1. Register Voter
2. Cast Vote
3. Display Candidates
4. Display Results (Admin)
0. Exit
-------------------------------------------

Enter your choice: 4
Enter Admin voter_id: Admin
Enter Admin password: Admin
Login successful!
Election Results:
Candidate1: 0 votes     Candidate2: 0 votes     Candidate3: 1 votes     Candidate4: 0 votes     Candidate5: 1 votes
```