1.FUNCTIONAL AND NON-FUNCTIONAL REQUIREMENTS FOR ONLINE VOTING SYSTEM

Functional Requirements:

1 . Voter Registration:

- Users should be able to create accounts for voter registration.
- The system must validate and store voter information securely.

2. Authentication and Authorization:

- Implement a secure login system for verified users.
- Ensure proper authorization to access different sections based on user roles.

3. Ballot Creation:

- Allow administrators to create electronic ballots for different elections.
- Support various types of elections, such as general, primary, or referendum.

4. Candidate Nomination:

- Enable candidates to submit their nominations online.
- Include a verification process for candidate eligibility.

5. Voting Interface:

- Provide a user-friendly interface for casting votes.
- Ensure the confidentiality and integrity of each vote.

6. Vote Counting:

- Implement an automated and accurate vote counting mechanism.
- Display real-time or timely updates on the vote count.

7. Results Publication:

- Publish election results securely and accurately.
- Ensure transparency in the results presentation.

8. Audit Trail:

- Maintain an audit trail for all system activities, ensuring accountability.
- Log all user interactions and system events.

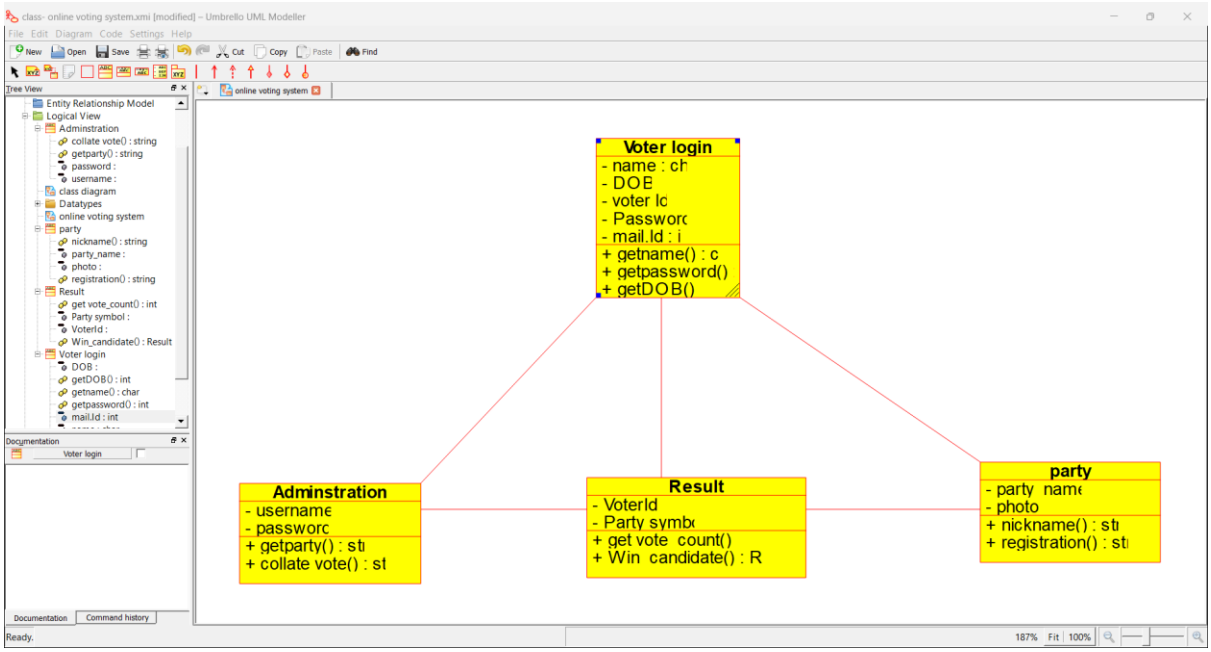
Non-Functional Requirements:

1. Security:

- Employ strong encryption methods to protect voter data.

- Implement measures to prevent unauthorized access and tampering.

2. Scalability:
   - Design the system to handle a scalable number of users and votes.
   - Ensure performance under varying load conditions.
3. Reliability:
   - Minimize downtime and ensure high system availability during elections.
   - Implement backup and recovery mechanisms to safeguard against data loss.
4. Usability:
   - Create an intuitive and easy-to-use interface for both voters and administrators.
   - Ensure accessibility for users with diverse abilities.
5. Compliance:
   - Ensure compliance with electoral laws and regulations.
   - Implement measures to prevent fraudulent activities.
6. Auditability:
   - Facilitate auditing processes to verify the integrity of the election.
   - Provide tools for election authorities to conduct post-election audits.
7. Performance:
   - Optimize system performance to handle concurrent user activities.
   - Minimize latency in processing votes and displaying results.
8. Data Privacy:
   - Adhere to data protection laws and regulations.
   - Implement measures to safeguard voter privacy and anonymity.



2.FUNCTIONAL AND NON-FUNCTIONAL REQUIREMENTS FOR LIBRARY MANAGEMENT SYSTEM

Functional Requirements:

1. User Authentication: The system should provide secure login functionality for librarians, administrators, and users.

2. Book Catalog: The system must allow librarians to add, edit, and delete book records. Users should be able to search and view the available books.

3. Check-in and Check-out: Librarians should be able to check in and check out books for users. The system must update the book availability status accordingly.

4. Reservations: Users should be able to reserve books that are currently checked out.

5. User Management: Administrators should be able to add, edit, and delete user accounts. Users should have the ability to update their profiles.

6. Notifications: The system should notify users about due dates, overdue books, and reservation status.

7. Reporting: Generate reports on book availability, overdue books, and user activities.

Non-functio

nal Req uire me nts:

1. Performance: The system should respond to user requests within 2 seconds under normal load conditions.

2. Scalability: The system should be scalable to accommodate a growing number of books, users, and transactions.

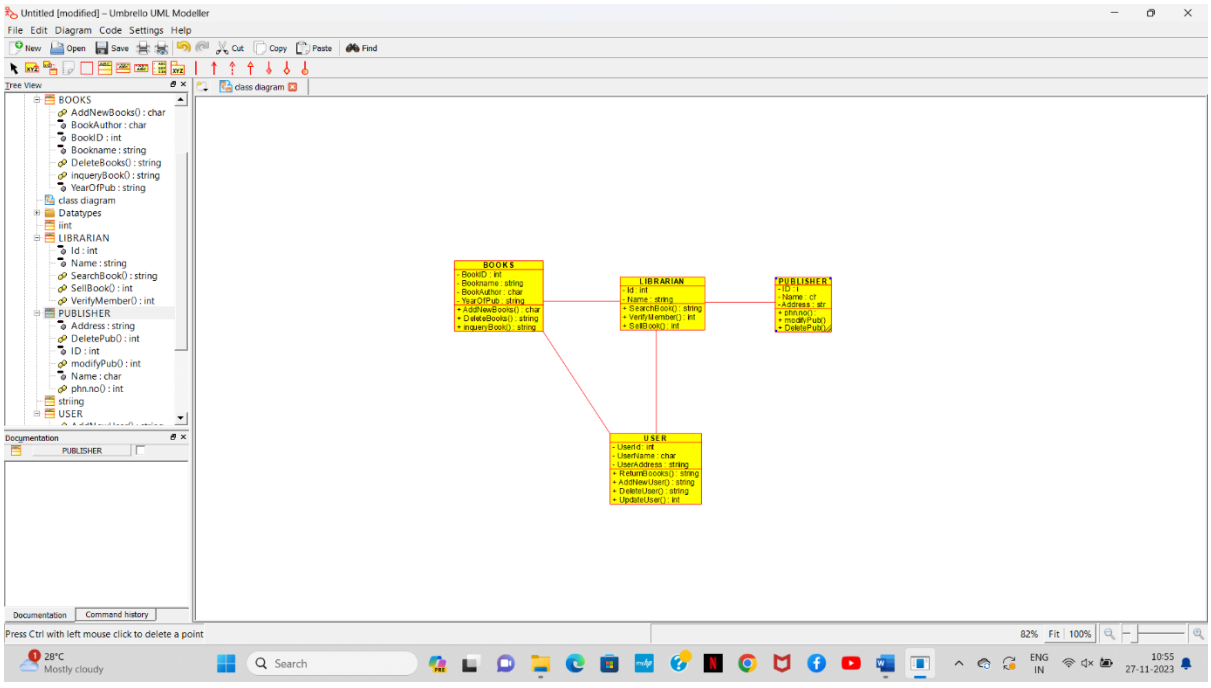
3. Reliability: The system should be available 99.9% of the time and should be able to recover gracefully from failures.

4. Security: Data should be encrypted during transmission, and user authentication should be secure. Only authorized personnel should have access to administrative functions.

5. Usability: The user interface should be intuitive, and librarians/users should be able to perform common tasks with minimal training.

6. Compatibility: The system should be compatible with common web browsers and accessible on different devices (desktop, tablet, mobile).

7. Maintainability: The system should be easy to maintain, and updates should not disrupt regular operations.



3.FUNCTIONAL AND NON-FUNCTIONAL FOR ATM MACHIINE

Functional Requirements for an ATM Machine:

1. Cash Withdrawal: Allow users to withdraw cash from their accounts.

2. Balance Inquiry: Provide the option to check the account balance.

3. Cash Deposit: Allow users to deposit cash into their accounts.

4. Fund Transfer: Enable transferring funds between accounts.

5. Statement Generation: Generate and print or display account statements.

6. Pin Change: Allow users to change their PIN for security.

7. Receipt Printing: Provide a printed or digital receipt for transactions.

8. Card Issuance/Replacement: Support issuing new cards or replacing lost/stolen cards.

9. Language Selection: Allow users to choose their preferred language for interactions.

Non-Functional Requirements for an ATM Machine:

1. Security: Implement robust security measures to protect user data and transactions.

2. Availability: Ensure the ATM is available for use 24/7 with minimal downtime.

3. Performance: Provide fast transaction processing to minimize user wait times.

4. Reliability: Ensure the ATM operates reliably without errors or system failures.

5. User Interface: Design an intuitive and user-friendly interface for easy navigation.

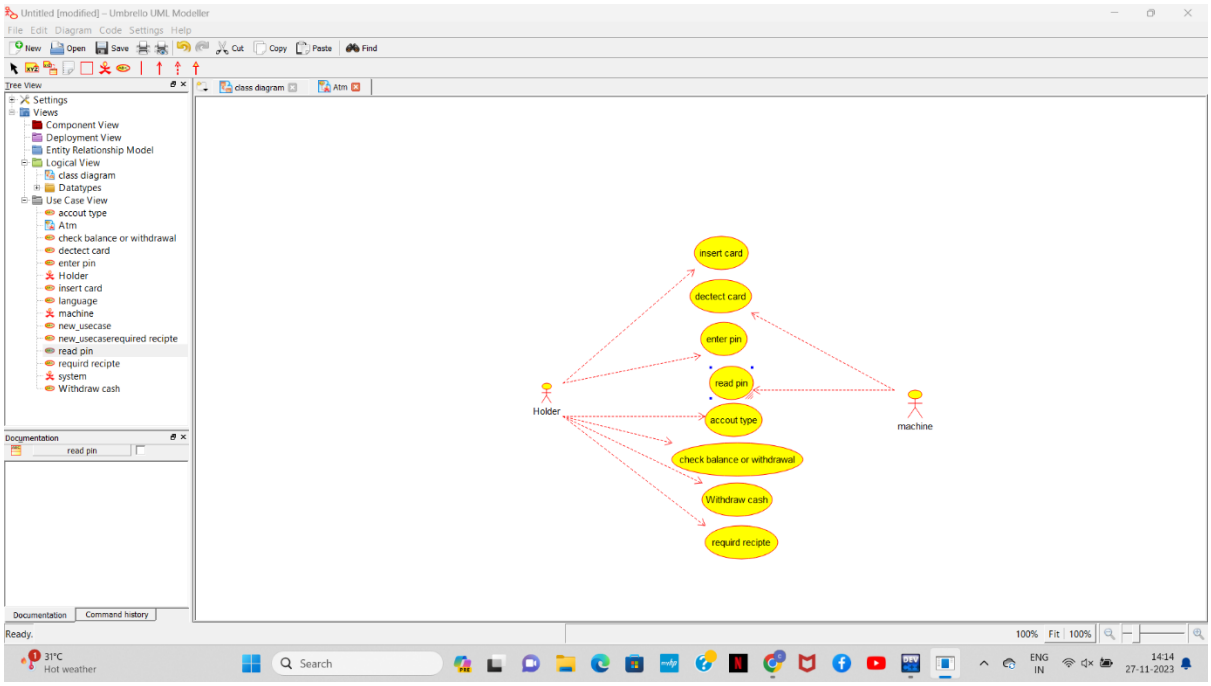
6. Accessibility: Ensure the ATM is accessible to users with disabilities.

7. Compliance: Adhere to relevant banking regulations and standards.

8. Scalability: Design the system to handle a scalable number of transactions as user demand grows.

9. Maintainability: Facilitate easy maintenance and updates to the ATM system.

10. Auditability: Keep logs of transactions for auditing purposes.



3.Online shopping system:

Functional requirements for online shopping system:

1.User Registration and Authentication:

Allow users to register securely.

Authenticate users to ensure secure access.

2.Product Catalog:

Display a comprehensive catalog of products.

3.Shopping Cart:

Enable users to add and remove items from their shopping cart.

4.Checkout Process:

Provide a secure and user-friendly checkout process.

5.Order Management:

Generate order confirmations for users.

6.User Reviews and Ratings:

Enable users to leave reviews and ratings for products.

7.User Account Management:

Allow users to manage their profiles, addresses, and payment methods.

Non-functional requirements for online shopping system:

1.Security:

Implement secure data transmission and storage.

2.Reliability and Availability:

Maintain high system reliability to prevent service disruptions.

3.Performance:

Provide quick response times for page loading and transaction processing.

4.Scalability:

Design the system to handle increased traffic during peak times.

5.User Interface:

Design an intuitive and visually appealing user interface.

6.Mobile Responsiveness:

Ensure the online shopping platform is responsive and accessible on various devices.

7.Data Backup and Recovery:

Regularly back up transaction and user data.

8.Compliance:

Comply with data protection and privacy regulations.

4. Blood donor system:

Functional requirements for blood donor system:

1.User Registration and Authentication:

Allow donors to register securely.

Authenticate users to ensure the validity of donor information.

2.Donor Profile Management:

Enable donors to manage their profiles, including contact details and blood type.

3.Blood Bank Inventory:

Maintain an inventory of available blood types.

4.Donation Scheduling:

Provide a system for donors to schedule blood donation appointments.

5.Donor Notifications:

Notify donors about blood donation drives or urgent needs.

6.Blood Donation Process:

Facilitate a smooth and efficient blood donation process.

7.Search and Request System:

Allow hospitals or organizations to search for specific blood types.

Non-functional requirement for blood donor system:

1.Security:

Implement secure data storage and transmission.

2.Reliability and Availability:

Maintain high system reliability to ensure availability during emergencies.

3.Performance:

Provide quick response times for donor registrations and inventory updates.

4.Scalability:

Design the system to handle an increasing number of donors and blood bank locations.

5.User Interface:

Design an intuitive and user-friendly interface for both donors and blood bank administrators.

6.Mobile Responsiveness:

Ensure the system is accessible and functional on mobile devices.

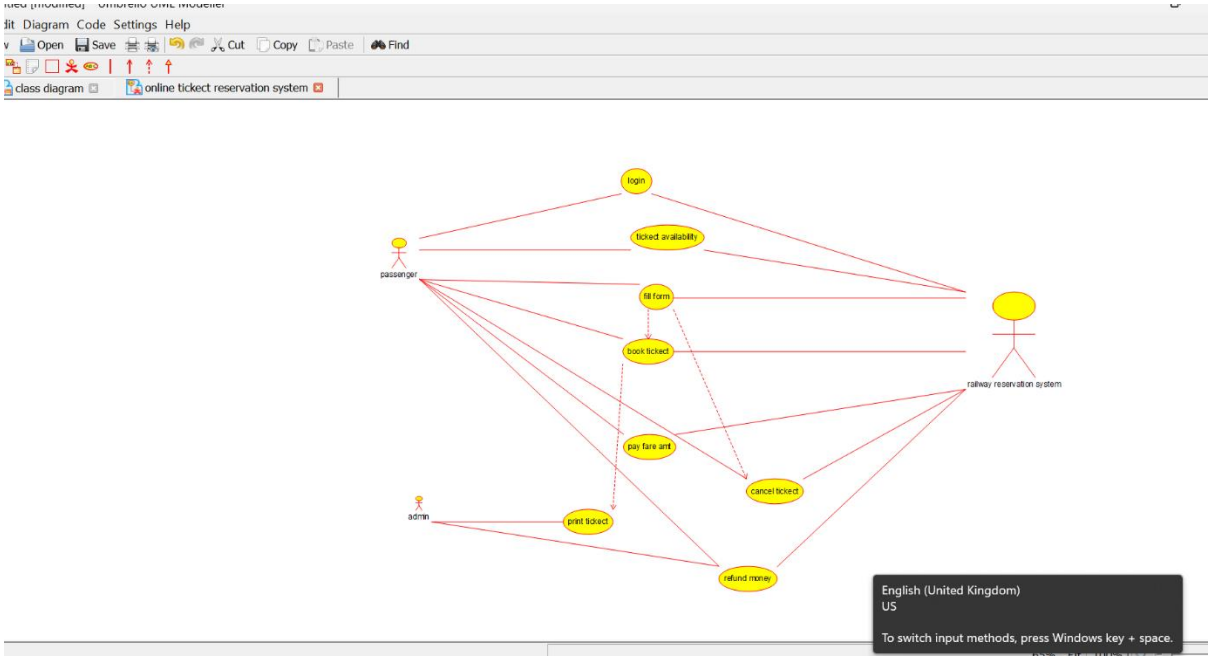
7.Data Backup and Recovery:

Regularly back up donor and inventory data.

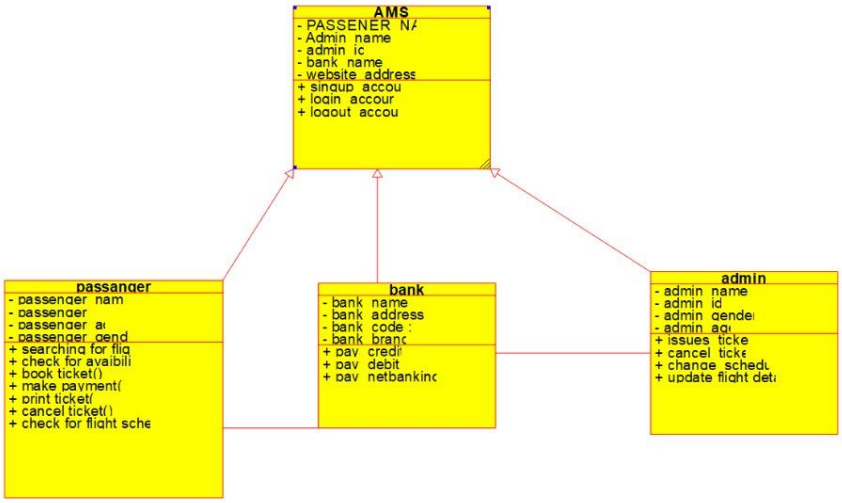
8.Compliance:

Comply with data protection and privacy regulations.

5. Use case diagram for online ticket reservation system:



6. Class diagram for airline ticket reservation system:



7. Cyclomatic complexity code:

```
#include<stdio.h>

int main() {

    int E, N, P, CC;

    printf("\n Program to find Cyclomatic Complexity:") ;

    printf("\n Enter the number of Edges in the flow graph:");

    scanf("%d", &E);

    printf("\n Enter the number of Nodes in the flow graph:");

    scanf("%d", &N);

    printf("\n Enter the number of Predicate Nodes in the flow graph:");

    scanf("%d", &P);

    CC = E - N + (2 * P);

    printf("\n The Cyclomatic Complexity of the flow graph is: %d", CC);

    return 0;

}
```