PROJECT DESIGN PHASE – PART 2

Date	04 November 2023
TeamID	NM2023TMIDO2239
Project Name	Electronic Voting System
Maximum Mark	4 Mark

DETERMINE THE REQUIREMENTS

(CUSTOMER JOURNEY MAPS)

Creating a customer journey map for an electronic voting system involves understanding the experience and interactions of voters, election officials, and other stakeholders throughout the electoral process. Here are some key points to consider when determining the requirements for a customer journey map for an electronic voting system:

1. Define the Target Audience:

- Identify the primary stakeholders involved, including voters, election officials, and IT administrators responsible for the electronic voting system.

2. Establish Objectives:

- Clarify the goals and objectives for creating the customer journey map. What insights or improvements are you aiming to achieve in the context of electronic voting?

3. Gather Data:

- Collect relevant data, including feedback from voters, election officials, and historical election data. This may include surveys, interviews, and usability studies.

4. Identify Election Phases:

- Divide the electoral process into key phases such as voter registration, ballot casting, counting, and results reporting.

5. Create Voter Personas:

- Develop detailed voter personas that represent different demographics, needs, and preferences of the electorate.

6. Document Touchpoints:

- List all the touchpoints or interactions that occur during each phase, such as voter registration websites, polling stations, and electronic voting machines.

7. Capture User Emotions:

- Understand the emotional aspects associated with each touchpoint. For example, voters may experience trust, satisfaction, or anxiety.

8. Identify Pain Points and Opportunities:

- Highlight any pain points or challenges that voters, election officials, or IT administrators encounter during the electronic voting process.

9. Define Stakeholder Goals:

- Determine the specific goals and motivations of voters, election officials, and IT administrators at each phase of the process.

10. Specify Key Performance Metrics:

- Define the key metrics to measure the success of the electronic voting system, such as voter turnout, system uptime, and security measures.

11. Consider Multichannel Experiences:

- Acknowledge that voters may interact with the electronic voting system through various channels, including online registration, mobile apps, and in-person voting.

12. Involve Cross-Functional Teams:

- Collaborate with experts from various disciplines, including election officials, cybersecurity experts, and user experience designers, to create a comprehensive map.

13. Visualize the Customer Journey:

- Create a visual representation of the customer journey map that can be used to communicate findings and insights to stakeholders.

14. Ensure Security and Privacy:

- Prioritize security and privacy considerations at all stages of the electronic voting process and integrate them into the map.

15. Test and Validate:

- Validate the customer journey map by conducting usability testing and obtaining feedback from relevant stakeholders to ensure accuracy.

16. Continuously Improve:

- Recognize that the electoral process and technology can evolve, so regularly update the customer journey map to reflect changes and improvements.

Creating a customer journey map for an electronic voting system is crucial to improving the voter experience, enhancing transparency, and maintaining the integrity of the electoral process. These requirements will help you create a comprehensive map that addresses the needs and challenges associated with electronic voting.

REQUIREMENT ANALYSIS (FUNCTIONAL,OPERATIONAL,TECHNICAL)/FLOW CHARTS

Analyzing the requirements and creating flowcharts for an electronic voting system involves understanding the functionality, operations, and technical aspects of the system. Here's a breakdown of requirement analysis and the creation of flowcharts for an electronic voting system:

Functional Requirements:

1. Voter Registration:

- Define the process for voter registration, including online registration and in-person registration at designated locations.

- Specify the information to be collected, such as name, address, date of birth, and identification documents.

2. Ballot Preparation:

- Outline how candidates and measures are added to the ballot.

- Specify the process for verifying candidate eligibility and ensuring a fair representation of all eligible candidates and issues.

3. Voting Process:

- Define the steps involved in the voting process, including voter authentication, ballot selection, and casting a vote.

- Ensure the system can accommodate various voting methods, such as electronic voting machines or remote voting options.

4. Security Measures:

- Specify the security measures to protect against fraud, tampering, and unauthorized access.

- Define the authentication methods, encryption standards, and auditing procedures.

5. Vote Counting:

- Describe the procedures for counting votes, including tallying and reporting.

- Ensure transparency and accuracy in the counting process.

6. Results Reporting:

- Specify how election results are compiled and reported to the public.

- Ensure the system supports real-time reporting and result verification.

7. Accessibility:

- Ensure the system is accessible to all voters, including those with disabilities.

- Define accessibility features and accommodations, such as audio interfaces and tactile ballots.

Operational Requirements:

1. Training:

- Define the training requirements for election officials, poll workers, and IT administrators.

- Specify the resources and materials needed for training.

2. Polling Stations Setup:

- Outline the setup process for polling stations, including the placement of electronic voting machines and other equipment.

3. Voter Assistance:

- Specify procedures for providing assistance to voters who require it, such as language assistance or accessible voting options.

4. Emergency Procedures:

- Define protocols for handling technical failures, power outages, or other emergencies during the voting process.

Technical Requirements:

1. Hardware Specifications:

- Specify the hardware requirements for electronic voting machines, servers, and data storage.
- Ensure compatibility with existing infrastructure.

2. Software Development:

- Detail the software development requirements, including programming languages, security protocols, and user interfaces.

3. Database Management:

- Define the database structure and data management procedures, including voter registration databases and election result databases.

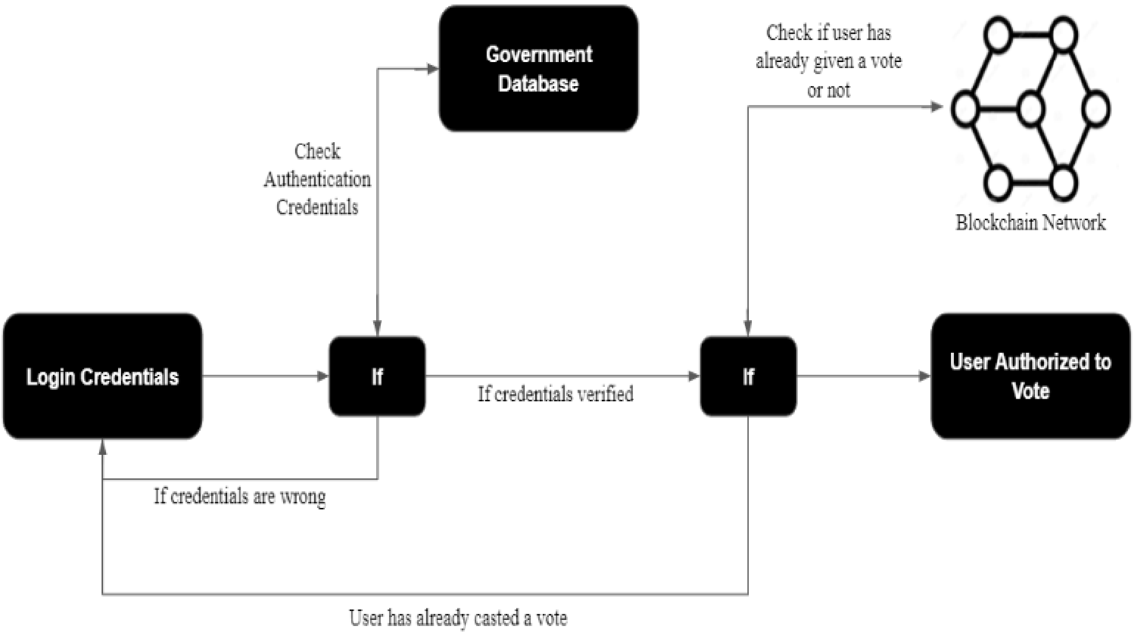
4. Security Measures:

- Outline technical security measures, including firewalls, encryption, and intrusion detection systems.

5. Scalability:

- Ensure the system can handle an increase in users and data during high-stakes elections.

Flowcharts:



TECHNICAL ARCHITECTURE

Designing the technical architecture for an electronic voting system is a critical step to ensure the system's security, reliability, and scalability. Here's an overview of the components and considerations that should be included in the technical architecture for an electronic voting system:

1. Front-End User Interface:

- Voter-facing interfaces for registration, ballot selection, and voting.
- Accessibility features to accommodate a diverse range of voters.

2. Voter Registration Module:

- A module for collecting and verifying voter information.
- Integration with government databases for identity verification.

3. Ballot Preparation Module:

- A module for defining and creating the electronic ballot.
- Features for adding, verifying, and certifying candidates and measures.

4. Authentication and Authorization:

- Strong authentication mechanisms to ensure that only eligible voters can cast their ballots.
- Authorization to determine the specific ballot options available to each voter based on their eligibility.

5. Voting Module:
   - Secure voting mechanisms to capture and encrypt votes.
   - Options for different voting methods (e.g., in-person voting, remote voting, vote-by-mail).

6. Security Measures:
   - Encryption of all data in transit and at rest.
   - Robust security protocols to protect against hacking, tampering, and fraud.
   - Intrusion detection systems and firewalls to monitor and secure the system.

7. Database Management:
   - Voter registration database for storing and managing voter information.
   - Election results database for securely storing and reporting results.
   - Backup and disaster recovery procedures to ensure data integrity.

8. Back-End Processing:
   - Ballot counting and validation algorithms.
   - Real-time result compilation and reporting mechanisms.

9. Servers and Infrastructure:
   - High-availability servers with redundancy to ensure system uptime.
   - Load balancing for even distribution of voter traffic.
   - Scalability to accommodate increased voter turnout during elections.

10. Mobile Voting Support:
    - Mobile application support for voters using smartphones or tablets.
    - Ensuring the security and integrity of mobile voting processes.

11. Auditing and Logging:

- Comprehensive audit logs to track all system activities.

- Tools for real-time monitoring and reporting of system health.

12. Integration with External Systems:

- Integration with government databases for voter verification.

- Integration with geographic information systems (GIS) for polling station locations.

13. Accessibility Features:

- Support for voters with disabilities, including audio interfaces, tactile ballots, and other accommodations.

14. User Support and Helpdesk:

- Mechanisms for voter and election official support, including a helpdesk and user documentation.

15. Testing and Quality Assurance:

- Rigorous testing processes, including penetration testing, to identify and rectify vulnerabilities.

- Ensuring that the system complies with legal and security standards.

16. Continuous Updates and Maintenance:

- Regular software updates to address security vulnerabilities and improve system performance.

- Monitoring and maintenance procedures to ensure ongoing system health.

17. Disaster Recovery and Contingency Planning:

- Procedures for handling technical failures, power outages, or other emergencies during elections.

- Redundant systems to ensure the continuity of the voting process.

18. Legal Compliance and Compliance Monitoring:

- Adherence to legal requirements, including data protection and privacy laws.
- Regular compliance monitoring and reporting.

19. User Education and Training:

- Voter and election official training to ensure the correct and secure use of the system.

The technical architecture for an electronic voting system must prioritize security, accessibility, and transparency to maintain public trust in the electoral process. It should also be designed to adapt to evolving technology and security standards. Regular security audits and testing are crucial to identify and mitigate potential vulnerabilities. Collaboration with experts in cybersecurity, election administration, and IT is essential to create a robust technical architecture for an electronic voting system.

OPEN SOURCE FRAMEWORKS

Open-source frameworks for electronic voting systems provide a foundation for building transparent and secure voting solutions. These frameworks are often customizable and can be tailored to meet specific election requirements. Here are some open-source frameworks commonly used for electronic voting systems:

1. DemocracyOS:

- An open-source platform that focuses on citizen participation in decision-making processes.
- Allows for online voting on various issues and policies.
- Encourages transparency and engagement in democratic processes.

2. Scytl:

- Offers open-source election management software.
- Provides tools for online voting, election administration, and secure result reporting.
- Used by governments and organizations worldwide for e-voting.

3. TrustTheVote:

- An initiative by the Open Source Election Technology (OSET) Foundation.
- A collection of open-source software for election administration and electronic voting.
- Aimed at improving the transparency and integrity of elections.

4. OpenSTV:

- An open-source framework for counting single transferable vote (STV) elections.
- Provides a transparent and auditable method for counting votes in multi-winner elections.

5. Follow My Vote:

- An open-source end-to-end verifiable online voting platform.
- Focuses on security and transparency.
- Provides cryptographic proofs that allow voters to verify their votes were counted accurately.

6. ElectionGuard:

- Developed by Microsoft, ElectionGuard is an open-source project for secure and verifiable electronic voting.
- It incorporates encryption, zero-knowledge proofs, and advanced cryptographic techniques to ensure the integrity of the voting process.

7. OpaVote:

- An open-source ranked-choice voting (RCV) and preferential voting system.
- Used for conducting online elections, including elections with ranked choices.

8. CIVS (Condorcet Internet Voting Service):

- An open-source online voting system that focuses on conducting ranked-choice voting elections.
- Provides a web-based platform for easy administration and voting.

9. Free & Fair's Suite of Voting Systems:

- Offers a set of open-source voting systems, including election management tools and voting machine software.
- Designed to enhance the security, transparency, and auditability of elections.

10. Belgium's eID Middleware:

- Belgium's electronic identity card middleware can be used as a reference for secure e-voting systems.

- It offers authentication and digital signature capabilities that can be integrated into electronic voting applications.

It's essential to note that building and implementing an electronic voting system, even with open-source frameworks, requires careful consideration of security, accessibility, and compliance with legal and electoral regulations. Additionally, customization and rigorous testing are necessary to meet the specific needs of each election.

When considering open-source frameworks for an electronic voting system, collaboration with experts in cybersecurity, election administration, and software development is vital to ensure the system's integrity and reliability.

THIRD – PARTY API'S

Integrating third-party APIs (Application Programming Interfaces) into an electronic voting system can enhance its functionality and provide additional features or services. However, it's crucial to choose third-party APIs carefully, ensuring they meet security, reliability, and compliance standards. Here are some categories of third-party APIs that can be considered for integration into an electronic voting system:

1. Identity Verification APIs:

- APIs that provide identity verification services, such as government-issued ID validation and biometric authentication. These can help ensure that voters are eligible and have a valid identity.

2. Geolocation APIs:

- Geolocation services can assist in verifying the physical location of voters, ensuring they are within the appropriate voting jurisdiction.

3. Messaging and Notification APIs:

- These APIs enable the system to send reminders, notifications, and updates to voters, election officials, and other stakeholders via email, SMS, or push notifications.

4. Communication APIs:

- Integrating communication APIs like video conferencing or chat services can facilitate remote voter assistance and support during the voting process.

5. Payment Gateway APIs:

- For online voting systems that require payment of voting fees or other transactions, payment gateway APIs can be used to securely process payments.

6. Blockchain APIs:

- If your voting system uses blockchain technology for added security and transparency, you may integrate blockchain APIs for transactions, verification, and auditing.

7. Security APIs:

- Third-party security APIs can provide features such as intrusion detection, vulnerability scanning, and advanced encryption for added protection against cyber threats.

8. Document Verification APIs:

- These APIs can assist in verifying the authenticity of documents, such as voter registration forms and identification.

9. Accessibility APIs:

- For ensuring that the electronic voting system is accessible to all, accessibility APIs can be integrated to provide features like screen readers and assistive technologies for voters with disabilities.

10. Translation APIs:

- If your electronic voting system serves multilingual communities, translation APIs can offer real-time translation services for voters who prefer to vote in their native language.

11. Result Reporting APIs:

- APIs for generating and publishing election results in various formats, making them accessible to the public and relevant authorities.

12. Social Media APIs:

- Integrating social media APIs can enhance voter engagement and communication, facilitating information dissemination and voter education.

13. Analytics and Reporting APIs:

- These APIs can provide insights into voter behavior, system performance, and other important metrics related to the electronic voting process.

14. Legal Compliance and Audit APIs:

- To ensure adherence to legal requirements and regulations, APIs for compliance monitoring and auditing can be integrated to track and report on the election process.

When integrating third-party APIs into an electronic voting system, it's essential to consider data privacy, security, and the API provider's terms of service. Thorough testing and validation should be conducted to ensure the compatibility and reliability of these APIs within the voting system. Additionally, ongoing monitoring and maintenance are crucial to address any issues and maintain the system's integrity.

CLOUD DEPLOYMENT

Deploying an electronic voting system on the cloud can offer scalability, security, and accessibility benefits. However, it's crucial to follow best practices and security measures to ensure the integrity of the electoral process. Here's a step-by-step guide for cloud deployment of electronic voting machines:

1. Define Your Cloud Strategy:

- Determine which cloud service provider (e.g., Amazon Web Services, Microsoft Azure, Google Cloud) is most suitable for your needs based on factors like infrastructure, geographic availability, and compliance with regulations.

2. System Architecture Design:

- Design the cloud-based architecture of the electronic voting system. Consider factors like scalability, redundancy, and security.

3. Data Security and Privacy:

- Implement robust security measures to protect voter data, including encryption of data in transit and at rest.

- Ensure compliance with data protection and privacy laws applicable in your region.

4. Access Control and Authentication:

- Use access control mechanisms to limit system access to authorized personnel.

- Implement multi-factor authentication (MFA) for added security.

5. Redundancy and Failover:

- Deploy your system in multiple availability zones or regions to ensure high availability.

- Implement automatic failover mechanisms to minimize downtime in case of server failures.

6. Backups and Disaster Recovery:

- Regularly back up data and system configurations to prevent data loss.

- Develop a disaster recovery plan and conduct drills to ensure data can be restored in the event of a catastrophic failure.

7. Compliance and Auditing:

- Ensure your cloud deployment complies with relevant regulations and standards for electronic voting systems.

- Enable auditing and logging features to monitor system activities for security and compliance purposes.

8. Network Security:

- Implement firewalls and network security groups to control inbound and outbound traffic.

- Use virtual private clouds (VPCs) or virtual networks to isolate different components of the system.

9. Monitoring and Alerting:

- Set up monitoring tools to track system performance, resource utilization, and security events.

- Configure alerts to notify administrators of any anomalies or security incidents.

10. Load Testing and Scalability:

- Conduct load testing to determine the system's capacity to handle a high volume of concurrent voters.

- Configure auto-scaling policies to dynamically allocate resources based on demand.

11. Secure Software Development:

- Develop and deploy secure code, following best practices for web application security.

- Regularly update and patch software components to address vulnerabilities.

12. Data Center Selection:

- Choose the cloud data center locations strategically to comply with local and international regulations and reduce latency.

13. Voter Education:

- Educate voters about the new cloud-based voting system to ensure they are comfortable with the changes and the security measures in place.

14. Disaster Preparedness:

- Develop a comprehensive plan for handling technical failures, security breaches, and other emergency situations.

15. Continuous Testing and Monitoring:

- Regularly test the system for security vulnerabilities, conduct penetration testing, and update your security measures based on findings.

- Continuously monitor the system's performance and security status.

16. Voter Authentication and Access:

- Ensure that voters have secure and user-friendly methods for authentication and access to the electronic voting system.

17. Public Communication:

- Communicate to the public, candidates, and relevant authorities about the security and integrity measures in place for the cloud-based voting system.

18. Legal and Regulatory Compliance:

- Ensure the system complies with all applicable election laws, regulations, and standards.

Remember that deploying an electronic voting system on the cloud involves a significant responsibility to safeguard the democratic process. Collaboration with experts in cybersecurity, election administration, and cloud technology is essential to ensure the system's security and integrity. Additionally, independent auditing and transparency measures can help build public trust in the electoral process.