The Mobile based Voting System

CRITICAL DESIGN REVIEW.

Faculty: Dr. Kavita Vemuri
Student mentors: Suhan Prabhu
Group Number: 6
Team Leader: Siddharth Bhat
Assistant Team Leader: Ganesh Vanahalli
Subgroup Leaders:

- ➔ Maintenance and Backend : Rohit Kumar Agarwal
- ➔ Statistics and Big Data Analysis : Samyak Agrawal
- ➔ User Interface and Architecture : Vashist Madiraju
- ➔ Security : Nitesh Gupta
- ➔ Awareness and Reaching Out : Kaushik Komallapalli

Table of Contents:

- System Organization

Iii.

1. Subsystem 1 - Maintenance and Backend
    1.1 Introduction
        1.1.1 Subsystem Scope
        1.1.2 Subsystem Purpose
        1.1.3 Subsystem Overview
        1.1.4 Subsystem Characteristics
    1.2 Requirements
    1.3 Objectives
    1.4 Implementation
    1.5 Challenges
    1.6 References
    1.7 Subsystem Use Case Diagram
    1.8 Interactions with other subsystems
    1.9 Workflow diagram

2. Subsystem 2 - Statistics and Big Data Analysis
    2.1 Introduction
        2.1.1 Subsystem Scope
        2.1.2 Subsystem Purpose
        2.1.3 Subsystem Overview
        2.1.4 Subsystem Characteristics
    2.2 Requirements
    2.3 Objectives
    2.4 Implementation
    2.5 Challenges
    2.6 References
    2.7 Subsystem Use Case Diagram
    2.8 Interactions with other subsystems

System Overview

- This project is to improve the Mobile Based Voting system proposed in 2009, integrating the system with new technological developments made so far which will make the system more efficient and user friendly.
- Citizens around the world recognize and embrace the benefits of e-Government services such as online tax filing, license renewal, and benefits claims. Now governments are initiating strategies that support e-democracy and in doing so, engaging more citizens in democratic processes. This brief addresses the highly formal processes of e-democracy; in particular e-voting to offer the government the infrastructures, applications, and services necessary to implement and manage manageable, scalable, reliable, secure e-voting systems.
- A majority of the country's youth does not vote as the election day generally clashes with the other commitments such as exams, college, work etc. In such cases, due to their laziness, they don't go and give excuses of long lines to cast vote, etc.. This mobile based system allows the people of the country to vote from the comfort of their homes, without the waiting and travelling period.
- The benefits of e-voting :
    - **Reduced costs** - E-voting systems reduce the materials required for printing and distributing ballots. Internet based voting, in particular, offers superior economies of scale in regard to the size of the electoral roll.
    - **Increased participation and voting options** - E-voting offers increased convenience to the voter, encourages more voters to cast their votes remotely, and increases the likelihood of participation for mobile voters. Additionally, it permits access to more information regarding voting options. Around 30% of the country's population possess or have access to a smartphone, most of these which have the usual Android or iOS operating systems. The people between 18 and 30 who are the youth of the country make up around 16-17% of the population, and most of them are not able to vote for various reasons (eg. Studying in a different state).
    - **Greater speed and accuracy placing and tallying votes** - E-voting's step-by-step processes help minimize the number of miscast votes.

The electronic gathering and counting of ballots reduces the amount of time spent tallying votes and delivering results.

- **Greater accessibility for the disabled and elderly** - Because the system supports a variety of interfaces and accessibility features, e-voting systems allow senior citizens who are unable to travel to cast their vote, and citizens with disabilities to vote independently and privately.
- **Awareness** - Using social media, awareness (benefits) of mobile based voting system are conveyed.
- **Flexibility** - E-voting can support multiple languages, and the flexible design allows up-to-the-minute ballot modifications.

System Objective

- Our belief in democracy, our values of tolerance, fairness and justice all compel us to uphold the basic right given by DEMOCRACY, RIGHT TO VOTE to choose our leaders on our own and this is an attempt to make it more comfortable and secure.
- In this paper we suggest a mobile phone voting protocol consisting of three phases: voter registration; vote casting and vote collecting and result phase. Proposed protocol provides secure and efficient online vote casting and can also be implemented parallel with paper ballot voting system.
- **The goal is to provide convenience and access to the electorate without the geographical restrictions.**
- To analyze the current election system and suggest modern development of an mobile based voting system which will allow voters to cast their votes in a more convenient way.
- To develop a mobile based voting system that is transparent, reliable, tamper proof and that will provide a platform for voting.
- To reduce the usage of paper, which indeed is a way to **save trees**.
- Effective usage of social media so as to provide maximal amount of awareness on the new system.
- Greater speed and accuracy placing and tallying votes.
- Minimize the number of miscast votes and ensuring that the voter's genuine choice is recorded.
- Reduce the entire effort of setting up polling booths.
- To avoid the malpractices such as booth capturing.

- Our system also enhances the security and provides more mobility and convenience to voters.
- To allow the youth of the nation that fail to vote due to not being in their own ward to vote.
- To let all those physically disabled people, who due to their physical challenges can't make it to the polling booth, vote.

Operational Objectives

- To analyse the current election system and suggest modern development of an electronic online voting mobile application which will allow voters to cast their votes in a more convenient way.
- To design a mobile application prototype of a voting system for people with access to smartphones.
- To develop a mobile application voting system that is transparent, reliable, tamper-proof, secure, convenient and user friendly.
- Validating the system so that only eligible voters are allowed to vote.
- Improve the registration process by allowing voters to check their registration status prior to voting and centralizing registrational databases.
- Ensure that nobody's vote is counted more than once and discard the duplicate votes, if any.
- The system should ensure the privacy of the voter is not breached via any digital method while he is in the process of casting his vote.
- To ensure that all computations during the election are done in a reasonable amount of time and voters are not required to wait on other voters to complete the process.
- Test and review the system for usability and functionality.

System Design

Scope

- The current scope of the system doesn't include the entire nation.
- This is only for the people who own smartphone and are technically capable to install an application and complete the authorisation.
- According to the current statistics, ~30% of the people in India own a smartphone.
- Assuming on an average 2 people vote from a smartphone, our system will cover on an average ~60% of population.
- The scope of the system is limited to the people who have network access and people without the same are out of the scope of this system.
- It is limited to the people who are eligible to vote.
- The scope of the system however will expand with people having a more ready access to the smartphones and with setting up of more network stations.

Design Requirements

- The system requires the voters to have Voter ID/Aadhar card.
- The voter must possess/have access to a smartphone which runs on Android/iOS operating system.
- Good graphic designers to provide good pictorial representation wherever possible and necessary.
- Aadhar Database/ Voters list
- Stable Internet connection (preferably 3G or higher.)
- The government will ensure that sufficient network stations are set up in order to handle the traffic during the voting period.
- The public should be aware of the means and minimum/bare requirements required of the them to be eligible to vote in via this system.
- Good security protocols to ensure that the public trust and can safely exercise their right to vote an take part in fair elections.

Stakeholders

- Government
- Citizens
- ISPs

Design Characteristics

- Exporting and storage of election data which can be used for future analysis.
- Relinquishing the server when it faces a downfall.
- Elections results of all the constituencies and statistics of the elections are made public which can be used to study general trends or to improve accountability of the whole election process.
- Application interface which can be used to communicate all the details of the elections to the users.
- It is important to create awareness among the people whenever a new system arrives in society.
- Data encryption services so that our voting data is not accessible to mobile service provider.

System Organization

- The whole system is divided into five subsystems based on their expertise

    - **Maintenance and Backend** :- this subsystem handles the maintenance of the proxy servers and designs the architecture of the voting process.

    - **Statistics and Big Data Analysis** :- this subsystem processes the data that is acquired in this whole voting period and analyzes to give more information on how to make the system more efficient.

    - **User Interface and Architecture** :- this subsystem builds a GUI that is easier to use and efficient to carry out the voting process.

- **Security** :- this subsystem handles the security issues regarding the system such as hacking, tampering of votes physically as well as on network.

- **Awareness and Reaching Out** :- this subsystem plays a major role on educating the people about the system's advantages over existing voting systems.

- The subsystems are in constant interaction with each other to exchange information and making their subsystems more and more efficient hence making the system as a whole more efficient.

- Subsystem maintenance and backend is in interaction with security subsystem to design its way of data encryption so as to be compatible with security measures taken by security subsystem, it is in interaction with ui and architecture directly as the vote data is directly passed on from UI to backend subsystem.

- Subsystem statistics and big data analysis is in interaction with awareness and reaching out to collect participation data, also its in interaction with maintenance subsystem to know how many servers are deployed and other information to provide efficient measures. It takes population data from reaching out subsystem and analyzes number of servers needed as per the information given by backend subsystem and gives that number back to backend subsystem.

Subsystem 1: Maintenance and Backend

1.1. Introduction

1.1.1. Scope

- Setup and Maintenance of server infrastructure.
- Setup logging systems to notify stakeholders about the current server status.
- Adding new users and maintaining the user database.
- Storage of the votes polled. Maintaining the database securely.
- Verifying and allowing only authorised access to the servers.
- Maintenance of all other subsystems mainly Security and UI Architecture.

1.1.2. Purpose

- To set up the servers across the nation keeping in mind the traffic
- Ensure that the systems are genuine and have not been tampered with
- Ensure that the votes casted are stored properly and that there is no bug in the system
- Maintain the servers and keep updating it with the latest available features in consultation with the relevant sub-system
- Foresee problems that may occur and plan effectively to counter them.
- Being ready with effective backup plans in case of system failures on the D Day.

1.1.3. Overview

System maintenance is an ongoing activity, which covers a wide variety of activities, including removing program and design errors, updating documentation

and test data and updating user support. For the purpose of convenience, maintenance may be categorized into three classes, namely:

- Corrective Maintenance: This type of maintenance implies removing errors in a program, which might have crept in the system due to faulty design or wrong assumptions. Thus, in corrective maintenance, processing or performance failures are repaired. For our mobile voting system, it is essential to evaluate the error log and fix these errors before the system could be exploited.

- Adaptive Maintenance: In adaptive maintenance, program functions are changed to enable the information system to satisfy the information needs of the user. This type of maintenance may become necessary because of organizational changes which may include:
    a) Change in the election commission's procedures,
    b) Change in election commission's objectives, goals, policies, etc.
    c) Change in forms,
    d) Change in information needs of election council.
    e) Change in system controls and security needs, etc.

- Perfective Maintenance: Perfective maintenance means adding new programs or modifying the existing programs to enhance the performance of the information system. This type of maintenance undertaken to respond to voter's additional needs. Outside changes are primarily environmental changes, which may in the absence of system maintenance, render the information system ineffective and inefficient. These environmental changes include:
    a) Changes in governmental policies, laws, etc.,
    b) Economic and competitive conditions, and
    c) New technology.

1.1.4. System Characteristics

1.1.4.1 Physical characteristics

- This system's physical characteristics consist of the required hardware that is necessary to run servers. In general, our system will be a hardware heavy

system. We will require access to powerful servers, with access to large network bandwidth at low latency. We require both of those characteristics to provide for a consistent experience for the voting system.

- The location of the system - Since this system will need to talk to the outside world: That is, the election committee's software and servers, it would be advantageous for this system to be co-located with the election servers. This will help for faster connections, as well as increased security for our system.
- Our system will need to be located at a center where there is access to uninterrupted electricity and internet. Otherwise, our servers will fail and will hamper the election process.
- It is advisable to make sure that the location is physically inaccessible to reduce any risk of terrorist attack.

1.1.4.2 Behavioural Characteristics

- Uptime: The system will guarantee a five nines uptime (a downtime of less that five minutes a year). Such reliability is essential for the voting process to be held.
- Bug-freeness: The system will ensure that all of its code is race condition free and crash free. We will ensure this is possible by rigorously checking our code using state-of-the-art model checkers and static analysis tools. We have considered using formally verified code, akin to the DeepSpec project, that aims to provide full correctness of hardware and software. This will enable us to fully trust the computations performed by our server.
- Fault tolerance: It is always possible that adversaries gain access to some of the network. Under such circumstances, our services will be written in a distributed style to make it harder for an adversary to control the entire system. Indeed, we shall use byzantine-resistant protocols to ensure that we minimise the chance of a hostile takeover.
- Data exchange: The system will allow for exporting and storage of data, to allow for analysis of the election to be performed later. This is necessary both to publish statistics and analyses about the elections. It also enable researchers to mine the data for information. It would also allow the detection of certain styles of voter fraud. Indeed, some research has gone into this, but it appears that high quality data is not available so far. We will be the first to publish such data and make it available for science.
- Openness: We will host all the code on an open source platform, to ensure community participation, as well as increased trust in our systems. We would like for our system to be adopted by other countries and trusted by our citizens. To

this end, we wish to host our source-code on public version control. We are also considering using a blockchain-based model for voting, to ensure that a public ledger is present that records all votes that have been cast. Due to us storing data on a public ledger (ala blockchain), it will be tamper-proof.

1.2. Requirements

- The initial requirements for maintenance are listed, although the requirements would continue to evolve during the development and operation of the system. Considerations include:
- Maximizing system availability to meet the operational requirements. This has to take into account the designed-in reliability and maintainability of the system and resources available.
- Preserving system operating potential through proper planning of system scheduled maintenance. This requires a reliability-centered maintenance strategy that incorporates preventive maintenance in order to preempt failures, thereby extending the mean time between corrective maintenance, as well as enhancing the availability of the system.
- Segmentation of maintenance activities for potential outsourcing of non-critical activities to approved maintenance subcontractors as to optimize scarce technical manpower resources and maintenance/repair turn-around times.
- Harnessing IT technology for maintenance management. This involves rigorous and systematic capturing and tracking of operating and maintenance activities to facilitate analysis and planning.

1.3. Objectives

- To ensure that the servers are up and running before the elections
- To ensure the integrity of the elections by having tamper-proof systems
- To ensure that only authentic users are able to use the app
- To maintain and store the election and votes database securely.
- To ensure that only authorised people get root access to the system. To put in place necessary checks for that.

- To maintain all the subsystems and keep them ready

1.4. Implementation

- **Disaster management :** Our primary servers will be managed by us, in a physically secure location. However, it is always possible that an attack or a disaster takes place which destroys or corrupts our primary servers. In such cases, we will have backup services that can be launched on demand, by using SaaS (software as a service) provides such as Amazon AWS, or google cloud computing, where we would have backup services available and ready.

- **Data management:** We wish to ensure that the election data cannot be compromised, while also ensuring openness of our data. This may seem paradoxical, but is not so. Indeed, bitcoin has managed to do the same for cryptocurrency.

  We will leverage the blockchain technology to associate each person with a digital anonymous globally unique ID (GUID). This GUID is encrypted and is only stored on the user's phone. This makes it less susceptible to large scale data-loss as would be the case if the data was stored in a centralised server. Indeed, apple phones provide for secure enclaves to store cryptographically secure information. We will ensure that we use the best cryptography technology possible.

  We will make each vote add a node into the distributed ledger (Blockchain) of votes, thereby ensuring that an adversary is unable to tamper with the election data. This will be using the exact same principles as that of bitcoin / other blockchain providers.

- **Efficient data encoding :** Compressing the data size as much as possible to reduce the data usage so that even at very low net speed voting can be done. We will use standard techniques from coding theory to ensure that our data communication sizes are as small as possible. Most web servers now-a-days communicate with JSON, which is text based. While convenient, they are much

larger than what a densely packed binary encoding of the same data would be. We will use Google's Protocol Buffers, a technology for serialising data which is available on all platforms for this purpose.

- **Congestion control:** We will ensure that our servers are fault and congestion tolerant. We shall naturally follow best practises and have caches and load balancers as the frontline to our servers. We shall also make use of content delivery networks such CloudFlare, which as a bonus also defend us from naive DDoS attacks.

- **Vote uniqueness:** We will need to ensure that each user is only able to cast one vote. In light of this, we will ensure that users that have registered for the e-voting service will not be allowed to vote at EVM's - their name will not be printed on the list of eligible voters at the voting booth. This will ensure that voters cannot "double vote" and abuse our system.
- **Anti vote-manipulation:** We will use machine learning to try and learn patterns of vote manipulation and collusion, to alert the authorities about such possibilities as the votes are being cast. There is some past literature on the topic, but no one has really tried such a system on a scale such as India (the world's *largest* democracy). Therefore, this would be a prime research project.
- **Software:** As mentioned before, we will first design the control flow of our server in a model checker such as TLA+, which is a specification tool to ensure that concurrent systems behave as expected.

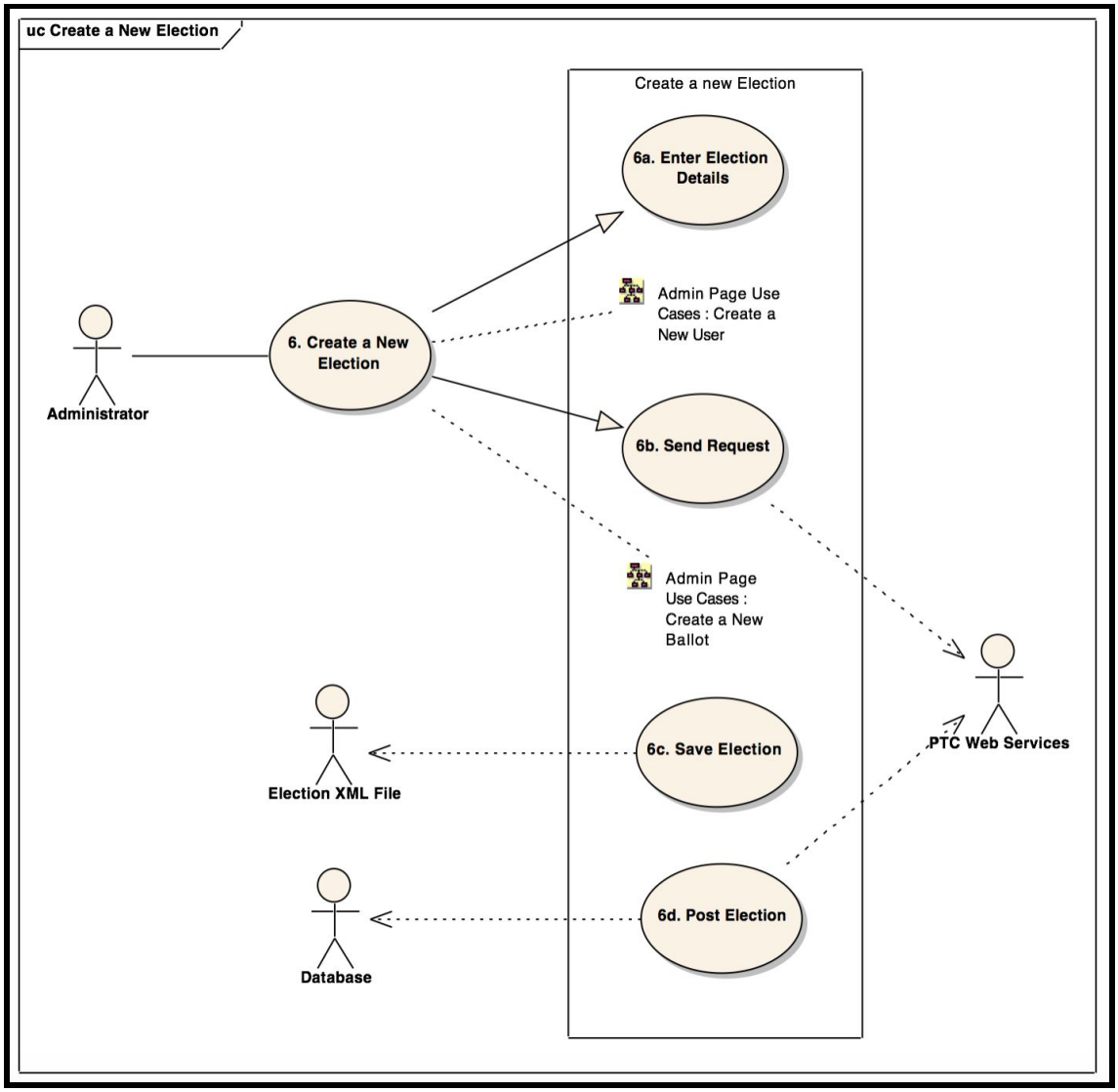
  Next, we will ensure that our system behaves as expected by having a rigorous testing environment with unit and integration tests. However, this is definitely not enough. We will make sure to run our code through static analysis tools which help identify errors in programs by deeply analysing the source code. We will ensure that we run our code through a gauntlet of static analysis tools.

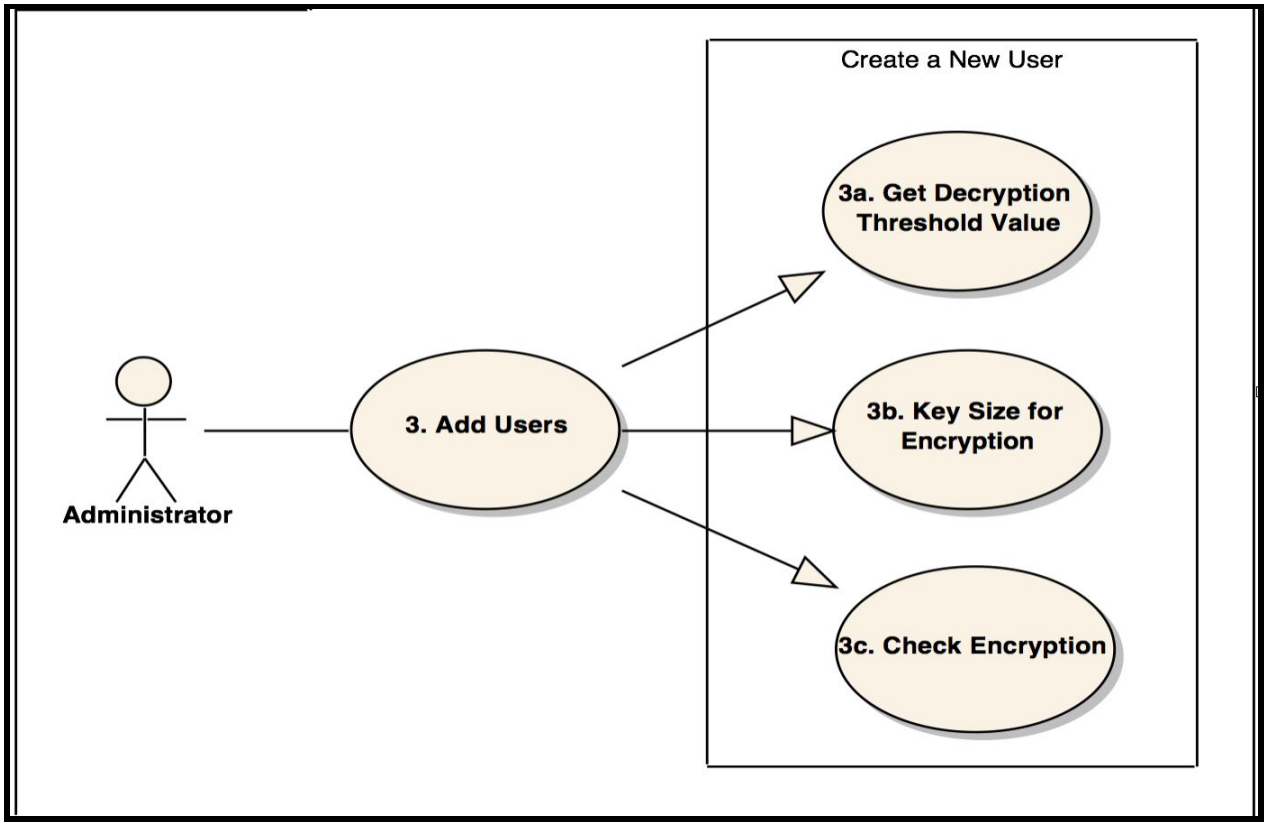
  In the end, we will run fuzz testing tools such as american fuzzy lop against the code that trace the assembly at runtime and try to abuse the code that it is running against.
- **Distributed system testing:** We will make use of industry best-practices when it comes to testing our distributed systems. Particularly, we will run Netflix's services called ChaosMonkey, which will arbitrarily always shut down some proportion of our servers during testing, to ensure that we can handle challenges like servers randomly failing during deployment. Si

1.5. Challenges

Challenges	Manageability	Reason
Unpredicted traffic	Manageable	Collecting the information from subsystem statistics and big-data analysis and posting the acquired number of proxy servers to handle traffic.
Loss of data while transmission	Manageable	Using recent error correction codes, data retrieval techniques to reconstruct the data, if not possible in worst case conducting the elections again.
Implementing security and prevent system breaches	Manageable	Monitoring server health, making sure no attacks have taken place

1.6. Use-Case Diagrams


uc Create a New Election
Create a new Election
6a. Enter Election Details
Admin Page Use Cases : Create a New User
6. Create a New Election
Administrator
6b. Send Request
Admin Page Use Cases : Create a New Ballot
PTC Web Services
Election XML File
6c. Save Election
Database
6d. Post Election



1.7. Interactions with other subsystems

- **Analysis** : Data analysed from the previous elections or from the feedback given by the Awareness team(to the Analysis team) will help us in deploying necessary proxy servers for the given region and preparing the system for sudden bursts in traffic and other problems.

- **Security** : The backend will communicate with the security team in two phases.
  When the user is being authenticated, a unique One Time Password(OTP) will be sent to the user as a second layer of authentication. The OTP will have to be securely sent and received by the backend.
  When a user casts his vote, the security team should ensure its proper encryption to maintain the integrity of the vote.
  Additionally, the security team must also ensure that any

communication taking place between the backend and UI is secure and not being tampered with.

- **Frontend** : The backend will communicate with the UI continuously, right from the login and authentication till the confirmation of the vote.
  The voter will have to be authenticated from the data stored in our database and his/her password securely matched.
  The backend will also send the list of candidates and their manifestos, and other relevant information.
  The vote casted will have to be checked for integrity and a confirmation message will be sent to the front end, completing the voting process for the user.

Subsystem 2: Statistics and big data analysis

1. Introduction

1.1. Scope

- The sub-system will work towards analysing the elections in terms of statistics, numbers and data.
- Sub-system is such that it can be used to better the whole process of digital voting via feedback on various components of the system.
- This sub-system mainly deals with collecting relevant election statistics and deriving important conclusions from them.
- The system also contributes towards enhancing the democratic setup of the country by releasing pre/post election data so that the people can make rational decisions.

1.2. Overview

- Big data is data sets that are so voluminous and complex that traditional data-processing application software are inadequate to deal with them. Some challenges in handling big data involve capturing data, data

storage, data analysis, search, sharing, transfer, visualization, querying, updating, information privacy and data source.

- Some examples of data statistics that we could exploit to make interesting observations/inferences could be:
    - The number of voters who actually cast their vote.
    - Percentage of people who voted for the candidate of a particular religion only.
    - Region-wise ethnic and communal population distribution and the distribution of votes amongst candidates from various communal and ethnic backgrounds.

1.3. Goal

- The goal of statistics and big data analytics is to help the Election Commission find out the actual percentage of the people who actually casted their votes in a particular area and in which area people are not interested in voting so that the Government can spread awareness among the people in that area.
- The feature can also be used by different political parties to analyse in which area they have support and in which area they need to concentrate in the upcoming elections.

2. Requirements

2.1. System Entities

- Ideation for proposing methods for conventional voting system with mobile voting.

- NGO's and government survey bodies.
- Voters of the country.

2.2. Subsystem Mission

- Find new trends in elections by looking at data from different domains and different sections of the same domain.
- Find out the sections of the society and regions of the country where people are not using the application and provide that data to the advertising subsystem.
- To make sure that the proposed solutions are consistent and compatible with other subsystems .
- To propose a maintainable and efficient Data collection system and optimize on the current methods of data collection .

3. Objectives

3.1. Pre-Mission Phase Objectives:

- The pre-mission phase involves gathering relevant data from surveys conducted over the years by government as well as independent NGO's.
- Build up the required software to support and handle the huge amount of data and different types of data.

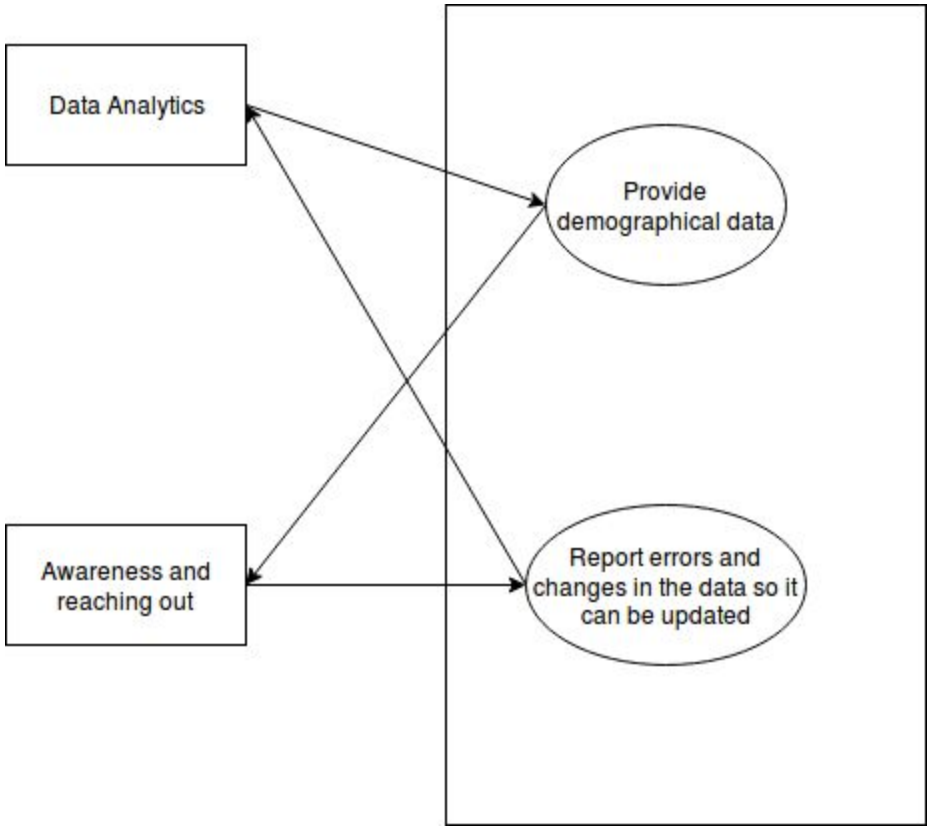
3.2. Mission Phase Objectives:

- Live update of data as soon as its available.
- Changing statistics based on the inflowing data during the time of elections .

3.3. Post-Mission Phase Objectives:

- Analyse the collected data and help other subsystems handle it and obtain useful conclusions from it .
- Identifying the shortcomings of the subsystems and working on them , also optimize the current processes and useful data generation methods.

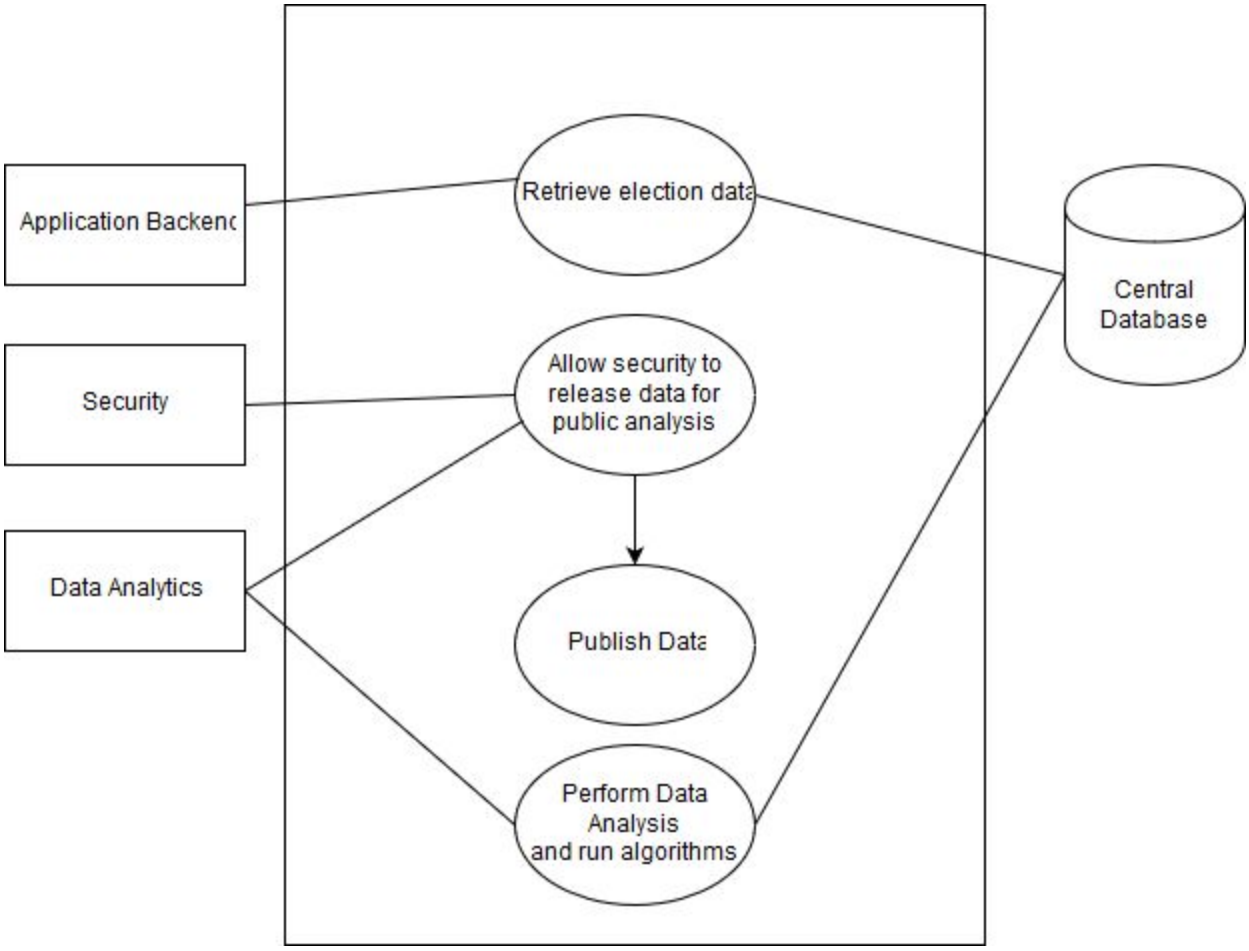
- Conduct surveys around the country to collect more data and target important areas based on the current data.
- Release the data which can be made public so that media and people can utilize it in decision making and discussions.
- Provide the demographics data to awareness and reaching out subsystem to help them prioritize the regions to visit based on the voter turnout and number of people using smartphones and also the economic strata to which people belong to .



4. Mission phase

4.1. Gathering data from backend:

Get live data from the central database of the developed application in order to process and analyse the data.



4.2. Process data using data analysis techniques:

Once we have the data from central database, we can run various techniques and algorithms to analyze and arrange the data and finally publish data for public viewing. Some of the techniques which can be put to use are:

- Associative Rule Learning:
  Are people who voted via traditional ballot paper voting more or less likely to vote through online voting system?
  Association rule learning is a method for discovering interesting correlations between variables in large databases.
- Machine Learning:
  To find patterns of different parameters based on previous voting trends. For example, which party is most likely to win in a particular constituency given it's previous history in elections.

Machine learning includes software that can learn from data. It gives computers the ability to learn without being explicitly programmed, and is focused on making predictions based on known properties learned from sets of "training data."

- Regression Analysis:
  How does your caste affect the party you vote for?
  At a basic level, regression analysis involves manipulating some independent variable (eg. caste, religion, financial status) to see how it influences a dependent variable (i.e. votes received by different parties). It describes how the value of a dependent variable changes when the independent variable is varied. It works best with continuous quantitative data like voter turnout, number of votes received by a party, etc.
- Sentiment Analysis:
  How well is the new e-voting system being received?
  Sentiment analysis helps researchers determine the sentiments of public with respect to a topic.

5. Stakeholders

- Election Commision Of India
- Political Parties
- Government of India
- Media(Both Electronic and Printed)

6. Challenges

Challenges	Manageability	Reasons and Proposed Solutions
Data Storage	Manageable	- Reason : Dynamic Nature of the electoral politics leads to intake and production of newer data. - Solution : Reducing excessive intake and ignoring minor news affecting the political

		world. Inconsistent and redundant data must be dropped.
Data Integration and Validation	Manageable	- Reason : Data intake is done via various resources such as news, NGOs, surveys, etc. leading to the problem of integrating and consolidating the data provided by the various sources. - Solution : Consider data from certain reputable sources and news channels. Ignore off-topic, unnecessary data. Tie-ups with unbiased sources for data provision and restrict the usage on the data provided by them.
Data Usage Insight	Manageable	- Reason : With data from multiple sources and with high intake of data, the segregation of minor unaffecting data from major game changing data that may turn the elections need to be identified and respective calculations must be carried out. - Solution : Upcoming technologies like machine learning and sentiment analysis, etc can be performed over big data to identify the

		cruciality of the data.
Data Quality	Manageable	- Reason : Multiple sources of data can lead to dirty data or false data, which might lead to incorrect statistical calculations and hence incorrect results. - Solution : Reliance with only reputable sources. Dropping data based on rumours.

7. Resources

- The data analysed in the previous years is helpful to make comparison between the past and present outcomes.
- Various big data storage and mining techniques such as Apache Hadoop, NoSql, Hive is used.
- Different Machine learning algorithms are applied to analyse the data in an efficient way.

Subsystem 3: Application Development and UI Architecture

1. Introduction

1.1. Scope

- This system targets the people who own smartphone and are technically capable to install and use the application.
- This subsystem will look at the User Interface (hence forward referred to as UI) architecture of the system.
- This subsystem is not fully independent, it has to coordinate with the other subsystem Backend and Maintenance.

1.2. Purpose

- UI is an essential component for building an app for the general public.
- Focus of this subsystem is to make UI easy to use, attractive, efficient and responsive.

1.3. Overview

- The Application is used after the voter goes to a nearby election office before the elections (over a period of 6 months), and have to link their Voter ID and Aadhar details to the applications database.(ADDED)
- After biometric verification, the person is asked to set a password, and this along with their Voter ID number will act as their LOG IN credentials on the application. (ADDED).
- The User Interface (UI) design of a mobile app is associated with its look and feel that means that how your app is going to be interacting with its users.The design of the app should be made by comprehending the perspective of end users.
- As the system is based on smartphones, it'll encourage the "Tech Savvy" youth of the country to come out and vote to voice their opinion, which has been missing in recent elections.

1.4. System Characteristics

- **Intuitive and consistent design**

  For an interface to be easily useable and navigable, the controls and information must be laid out in an intuitive and consistent fashion. Your users are probably well acquainted with many other interfaces, and you should be too if you want to achieve a level of familiarity for your users. Coming out with an entirely new layout might sound like a highly rewarding, paradigm-breaking project, but for all practical purposes, if you want users to feel at home then follow the path of your predecessors! Logic of usability should play a big part in the design process: features that are the most frequently used should be the most prominent in the UI and controls should be consistent so that users know how to repeat their actions.

- **Clarity**

  If a user is not able to understand his or her way around your interface, all the time you spent perfecting the software's functionality is rendered useless. Both in terms of visual hierarchy and content, there should be absolutely no ambiguity over the way your interface operates.

- **High responsivity**

  For a user to enjoy using your interface, it cannot feel as if the interface is lagging to keep up with their mouse clicking and keyboard tapping. If the interface fails keep up with the demands of the user, this will significantly diminish their experience and can result in frustration, particularly when trying to perform basic tasks.

- **Maintainability**

  Call it flexibility if you wish - a UI should have the capacity for updates to be installed and changes to be integrated without causing a conflict of interest. For instance, you may need to add an additional feature to the software, if your interface is so convoluted that there is no space to draw attention to this feature without compromising something else or appearing unaesthetic, then this signifies a flaw in design.

- **Attractiveness**

  A pretty look cannot make up for a poor design. But pleasant colour scheme can go a long way in making the user feel more at home when using your interface. Again, the aesthetics you choose for your interface

must be appropriate for the particular user - so perhaps some market research is required to determine exactly what your users are looking for. Good aesthetics give the product character and soul.

- **Conciseness**

  Clarity in a user interface is great, however, the design should not be over-clarifying. It is easy to add definitions and explanations, but every time you do that you add mass. Your interface grows. Add too many explanations and your users will have to spend too much time reading through them, it would be tough for the common person who is using the app. For example to use a particular feature instead of explaining in one sentence you are explaining in 3 , that makes your UI not concise.

- **Forgiving**

  A forgiving interface is one that can save your users from costly mistakes. For example, if someone deletes an important piece of information, can they easily retrieve it or undo this action? If by chance someone navigates to a broken or nonexistent page on the app,then they shouldn't be greeted with a cryptic error instead they should get a helpful list of alternative destinations.

2. Requirements

2.1. Every user must have Aadhar card or Voter ID card for voting.

2.2. Android or iOS (Most common Mobile Operating system in Smartphones) phone should be available to each user i.e . at least one within every household to ensure easy access to app. Android and iOS provide variety of pre-built UI components such as structured layout objects and UI controls that allow you to build the graphical user interface for your app.

2.3. Aadhaar and Voter ID Database.

2.4. State/Central government (i.e the governing body) will create servers for voting process (proxy).

2.5. Stable internet connection, preferably 3G or higher.

2.6. Some market research required to determine exactly what your users are looking for and what type of things are users familiar with.

2.7. Good UI designers.

2.8. Good UI specifications are needed which takes into account the data and context of the user within the application. This will be done while consulting the Awareness and Statistics subsystems.

3. Objectives

3.1. To ensure that the UI is friendly enough for people with all kinds of expertise so that everyone is able to vote easily.

3.2. To ensure that users want to vote and have ease of access to vote.

3.3. To ensure that the UI is light enough to access easily in poor connections.

4. Implementation

4.1. Design

- This system is designed to ensure that all the citizens of the country with access to a smartphone are able to access the application and are able to correctly identify and vote for their chosen party/candidate.
- This is done while consulting the Security subsystem and is done with complete coordination to avoid any loopholes and glitches in the Application.
- It is a light application which should be able to run on almost all smartphones regardless of the specification of the phone.

4.2. Accessing the Application

- The person will need to open the application.
- Then the person is displayed with a screen from which they will get to choose the language they want to access the app in.

4.3. Verification and Authentication

- The person then is required to enter their Aadhar credentials/Voter ID credentials(current system allows for either) for basic authentication of person.

- The user is verified from the relevant database. The verification includes checking if the person is eligible to vote and whether that person has already voted or not.
- In case a user is not valid or the user has already given the vote and still he is getting login to the app for voting relevant message is displayed on the app for which the response status can be received by backend.
- After user and password validation, they are sent an OTP (One-Time Password) for confirmation to their registered phone number.
- The OTP will be auto-detected by the Application, and incase it isn't due to special circumstances (Like voter not in possession of registered phone number), the voter will have the option to fill it in manually.
- The auto detection is to increase the ease of usage of the app, but mainly also to help the blind users who are unable to fill in the OTP.
- If the OTP is identified incorrectly or if the user has wrongly typed the OTP then he is given another chance to enter OTP and after certain time limit a new OTP is sent to the person's mobile.
- After confirmation of OTP the voter then proceeds to the voting portal.

4.4. Voting

- The voter is sent a list of all the candidates, in their constituency along with their respective party names and symbols.
- The ID's of respective candidates are randomly generated which is managed by backend. A random number generating device generates a random ordering of IDs of the candidates which is only known to the server and every time different IDs are passed to the backend and in the backend side they are decrypted to the original IDs. It prevents hacking of candidate IDs.
- So a user selects a candidate from the list by the symbols or the party name and the vote is sent for confirmation.
- If the vote is not confirmed then the user is given the appropriate message of some problem in giving vote and the screen goes back to Candidate selection screen and again the user selects the candidate for voting.

4.5. Confirmation

- After the person has chosen their candidate they are asked to confirm their choice, in case of a mistake. If it is confirmed then the vote is confirmed and sent to a secure outbox, which uploads the vote when the server is active, else holds it till server is active. This prevents the situation of internet connection getting lost in between voting or so.
- After confirmation of the vote the voter is then automatically logged out and the Application settings are then refreshed after changing the status of the previous voter to DONE, preventing duplication of votes.

5. Interactions with other subsystems

5.1. Awareness

The awareness subsystem will gather information about the different problems faced by users while voting. We will use their data as a feedback to continuously improve the UI to make it more user-friendly for the subsequent elections.

5.2. Security

The UI will communicate with the security team in two phases.

- When the user is being authenticated, a unique One Time Password(OTP) will be sent to the user as a second layer of authentication.
- When a user casts his vote, the security team must ensure that the vote is not intercepted by a third-party. The vote must reach the datacenter unaltered and unseen.

5.3. Backend

The UI will communicate with the back-end continuously, right from the login and authentication till the confirmation of the vote.

- Initially, the UI will send the user details to backend for all kinds of checks and verifications to be performed.
- Then, the backend will respond with a relevant error message, if any, which will be displayed as a popup and the user will be asked to login again.

- Once the user is authenticated, the UI will receive a randomly and uniquely indexed list of candidates.
- After the user casts his vote, the vote will be sent to the backend.

6. Challenges

The main challenges faced are listed with possible solutions in the table below:

- Physically Disabled Voters
- Illiterate Voters
- Mentally Disabled Voters
- Loss of connection to server

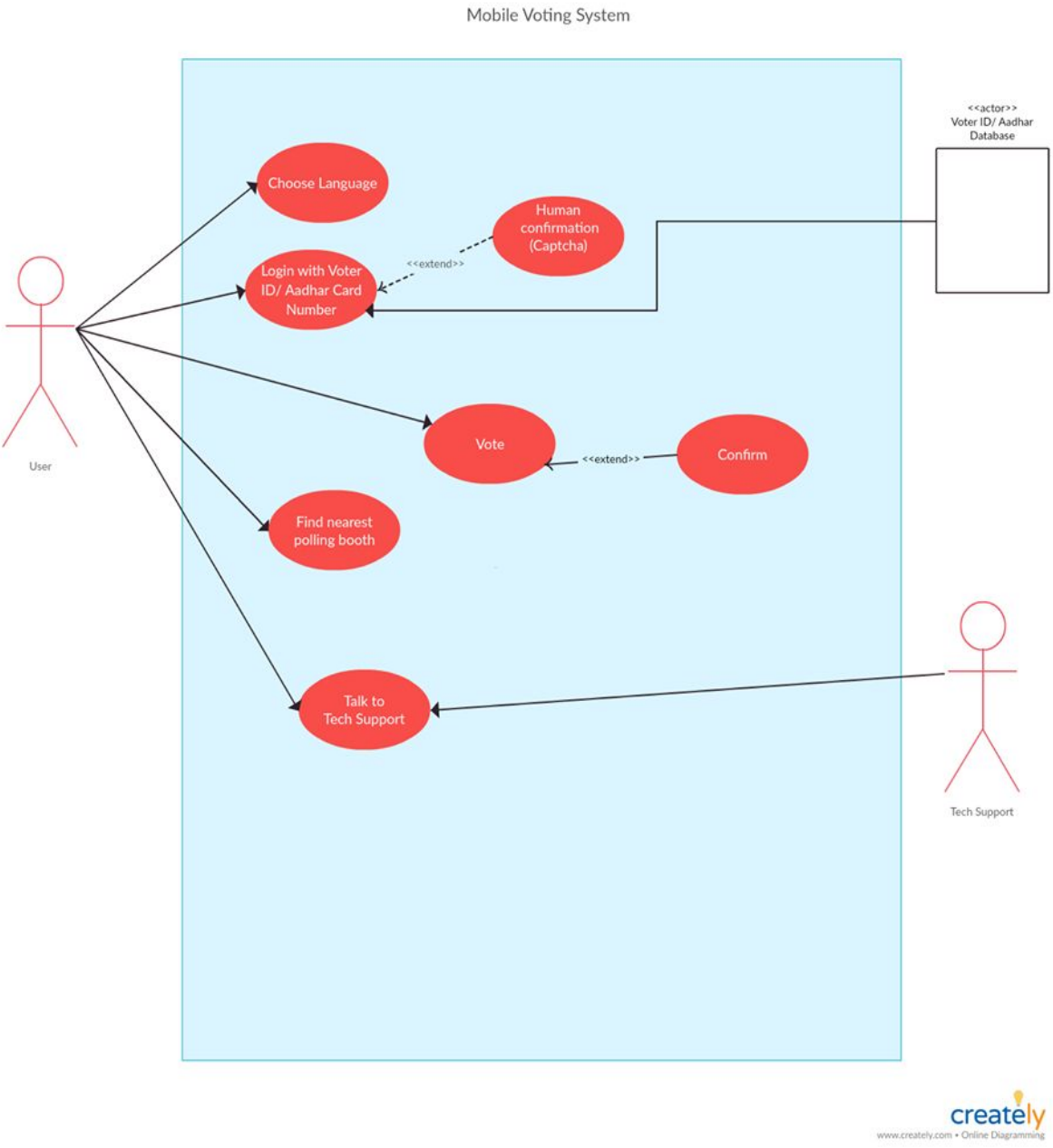
Table representing challenges:

Challenges	**Manageability**	**Proposed Solution**
Physically Disabled Voters: - Amputees	• Yes	• Through Vocal guidance/ can access app normally
- Blind/ Visually challenged	• Yes (With some compromises)	• Through Vocal guidance and help of a second person
- Deaf	• Yes	• Do not face any problems due to disability in this regard.
- Blind & Deaf	• No	• Not possible to make generalised app which will allow them complete freedom.
Illiterate Voters	• No	• Through pictorial representation we can overcome it to an extent, but they will need help to fill in the login credentials. This can be achieved with the help of good graphical advisory team, and is done regularly in many places like plane

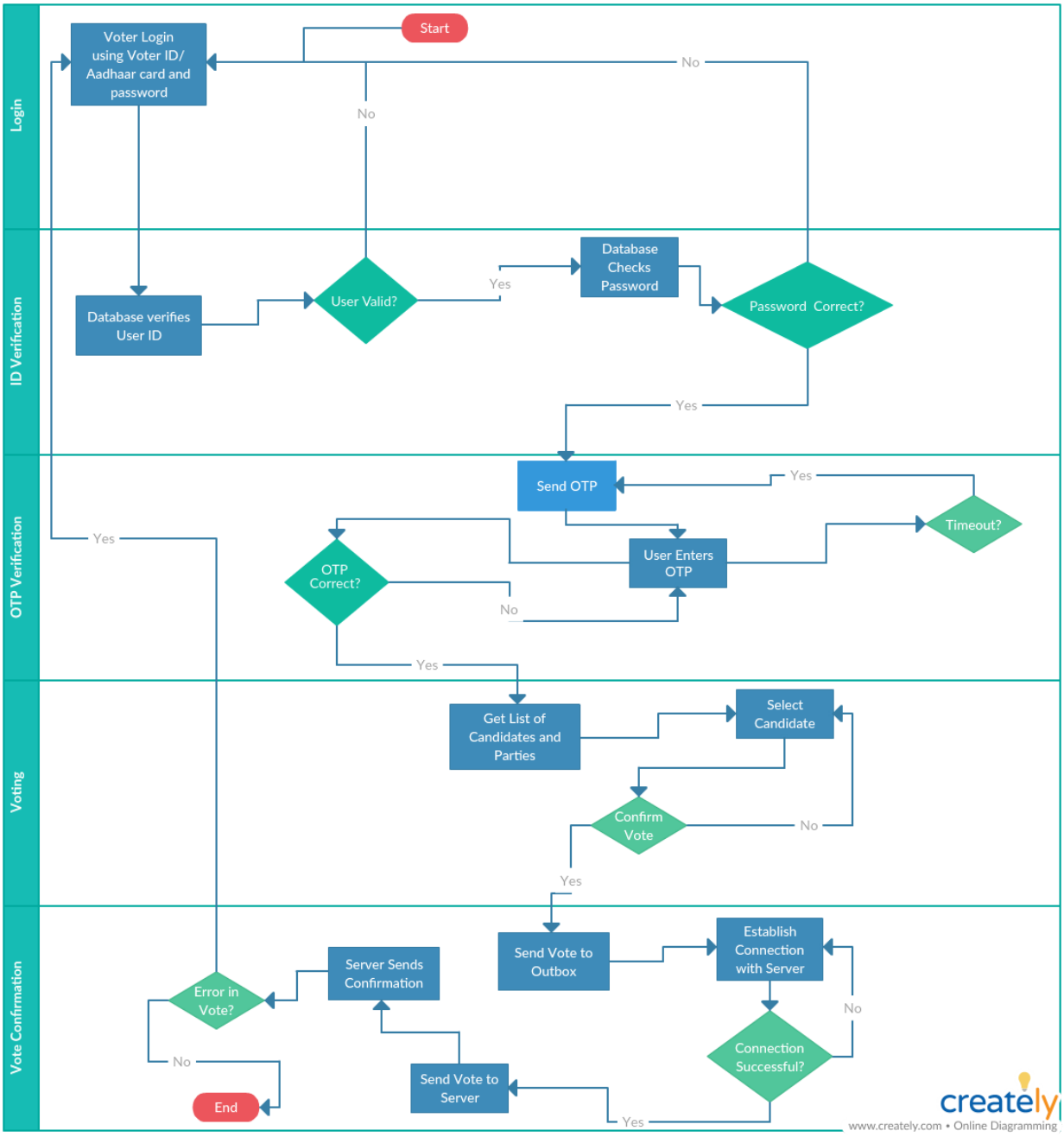
		safety manual etc.
Mentally Disabled Voters	• No	• Can not help the voter if they are not able to make proper choice and not mentally well.
Loss of connection to server	• Yes	• Solved by sending vote to secure outbox for storage until secure connection to server is established

7. Subsystem Diagrams

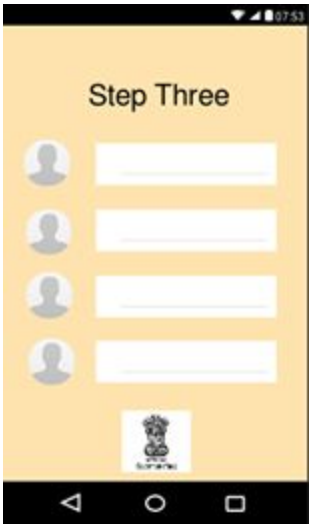
7.1. Use Case Diagram



7.2. Workflow/ State Diagram



7.3. Wireframes/ Mockups



Subsystem 4: Security

1. Introduction

1.1 Scope

- This subsystem will look at the major areas which are most prone to the attacks.
- The data collected will be within the past 1-2 year.
- The system will take care of all the major problems/attacks that have happened on already deployed Electronic Voting systems.
- This subsystem is limited to the people who are eligible to vote and have access to smartphones and are technically capable of using the application.
- With constant and active efforts of the Awareness and the Reaching out team, the scope of the system will expand as people who're passionate about voting and are unable to do so because they're not in their voting ward will get an incentive to buy a cheap smart phone and download the application as it'll let them vote.

1.2 Purpose

- The system is not foolproof and vulnerable to security and software problems , it can be hacked as compared to the manual voting done. This subsystem handles all such problems.
- It focuses on providing Network security as well as social security.
- Network security : Network security aims to provide security to avoid loss/manipulation of the votes along the network.
- Social security : Social security means avoiding social threats that might lead to vote tampering like people bullying other people to take their phone and cast vote or people stealing/borrowing phones to cast votes.

1.3 Overview

- Cyber threats are evolving everyday thus increasing the need to develop and tighten security measures to ensure the protection of everyone using the cyberspace. Increasing use of cyber space throughout the world coupled with globalization has increased the

complexity of cyber threats. Cyber security threats continue to escalate in frequency and variation.

- This has led to major security assaults all over the world some of which has been documented below.
- Attacks (threats):
    - Denial of Service in internet voting system:It can affect the internet voting system in two different ways of denial of service attacks
        - Hacker may able to change the network connection of the targeted web server with junk data that clogs the network and prevents the user or voter by accessing the web server and casting his vote.
        - In the second attack the hacker may put irrelevant resources with useless task on the election web server so that the server is busy. As the server is busy it may be unable to respond to the legitimate voters. This type of attack may mostly be done on the last day of the election, as most of the electorate may plan to cast their vote on the last day.

    - Domain name service (DNS) attack:
        - Attacks against the Domain name service route traffic to an attacker instead of to the legitimate vote service.

    - Viruses or Malicious Software:
        - There is a threat of introducing any malicious software onto the internet voting server before or on the Election Day by any communication link or email and also if huge number of PCs (personal computers) connected to the internet voting server, then any PC which is infected with virus, there might be more number of chances to spread the viruses to the voting server. This may change the confidentiality and integrity of the vote or even the vote may be changed without the knowledge of the voter before it is transmitted to the voting server.

    - Hacking
        - If the links of the voting system is changed or hacked then the voter may face difficulty in casting his/her vote, which may change the confidentiality and integrity of the vote.

    - Authentication Threats
        - One of the main feature of our voting system is to link with Aadhar. This gives rise to many Cross site attacks such as CSRF attacks, XSS attacks, Cross site timing attacks.
    - Documented problems with electronic voting across the world:
        - In Australia, during the New South Wales state election in 2015, there were several reports that over 66,000 electronic votes could have been compromised. Although the iVote (remote electronic voting system in New South Wales) website is secure, security specialist believe that a third party website was able to attack the system. This was the first time a major vulnerability was discovered in the middle of an ongoing poll.
        - In Netherlands, on October 30, 2006, the Dutch Minister of the Interior withdrew the license of 1187 voting machines from manufacturer Sdu NV, about 10% of the total number to be used, because it was proven by the General Intelligence and Security Service that one could eavesdrop on voting from up to 40 meters.
        - In United States, in 2010, graduate students from the University of Michigan hacked into the District of Columbia online voting systems during an online voting mock test run and changed all the cast ballots to cater to their preferred candidates. This voting system was being tested for military voters and overseas citizens, allowing them to vote on the Web, and was scheduled to run later that year. It only took the hackers, a team of computer scientists, thirty-six hours to find the list of the government's passwords and break into the system.
- Thus, there is a need for installation of better security measures to prevent the occurrence of these threats and for systems like the mobile based voting system which decides the future of the country, decides which party come to power it becomes essentially important to have preventive measures relative to the security of the system to prevent hackers or competing parties from tampering with the votes caste by the voters and thus altering the results of the overall elections.

1.4 System Characteristics

- Scalability: With the growth of population use of internet based voting system continues to increase so expansion in terms of coverage and capacity as needed- Should provide protection against huge traffic (Denial of Service attacks)
- Data Loss Prevention **:** Stops sensitive information from being revealed lets you stay in control what information is sent out.
- Application Control: Impedes unauthorized executables from functioning on your protected application (used for voting).
- Prevention as much as on real time protection against well known and new threats.
- To establish rules and measures to use against attacks over the Internet.

Requirements

Aadhar Card for each voter.
Phone (preferable android, ios, etc as operating system)
Aadhar Database (Used by government for linking bank accounts)
State government will create servers for voting process(proxy)
Stable internet connection
Ballot Server for counting

Objectives

3.1 Network security

- Network security consists of the policies and practices adopted to prevent and monitor unauthorized access, misuse, modification, or denial of a computer network and network-accessible resources.
- It contains the workflow of how the user will use the API to vote
- It also explains the protocol required for interaction between different parties
- It involves the authorization of access to data in a network, which is controlled by the network administrator.

3.2 Social security

- Social security consists of the practices adopted to avoid social threats to fair voting process like social bullying, phone theft etc.
- It contains our proposed solutions to overcome these problems.
- These problems however can’t be solved completely and this solutions only avoid them to some extent.
- To ensure accuracy and precision in voting to report the will of voters, to make every vote count.

Implementation

4.1 Network security

- Workflow
    - In this proposed system
        - Initially the voters should have to provide their Aadhar number to authenticate themselves and establish their user-ids. This constraint is imposed to ensure that only the genuine person is allowed to vote in the elections.
        - Our main focus is on the session OTP that is provided to the voter during the process of casting a vote. The OTP is nothing but a One-Time Password that is required to continue with the process of voting and this makes the system secure and free from misuse by an external party.
        - Election committee or administrator can authenticate and validate the voter as well as candidate.
        - Voter will caste his/her vote to the appropriate candidate according to their background information provided through the registered account.
        - Result will be displayed on the prescribed date and time

- Protocol
    - This system provides for three roles, namely, the signer (aadhar database), the recipient (the voter), the proxies(the state government), voting center and the bulletin(central government). The operation proceeds through four phases, namely, initiation, verification, voting and counting.

- Initialization
    - Step 1. The system first generates the required parameters(public key and private keys using the aadhar number for OTP) for each voter using blind signature. In a blind signature scheme, a signee could get a message's digital signature signed by a signer without revealing any information about the message.
    - Step 2. The public keys are passed on to various proxies(State government).

- Verification:
    - Step 3a. The voter gets authenticated with the Aadhar database using his Aadhar number. Then the proxy examines whether the registrant is a legal voter; if so, he or she distributes a certificate to the voter.(Check workflow pt 1,2,3)
    - Step 3b. The proxy publishes the certificate of the voter to the bulletin.
    - Step 3c. Meanwhile the voter checks whether he or she has registered successfully via the bulletin.
- Voting:
    - Step 4. The voter chooses a candidate.

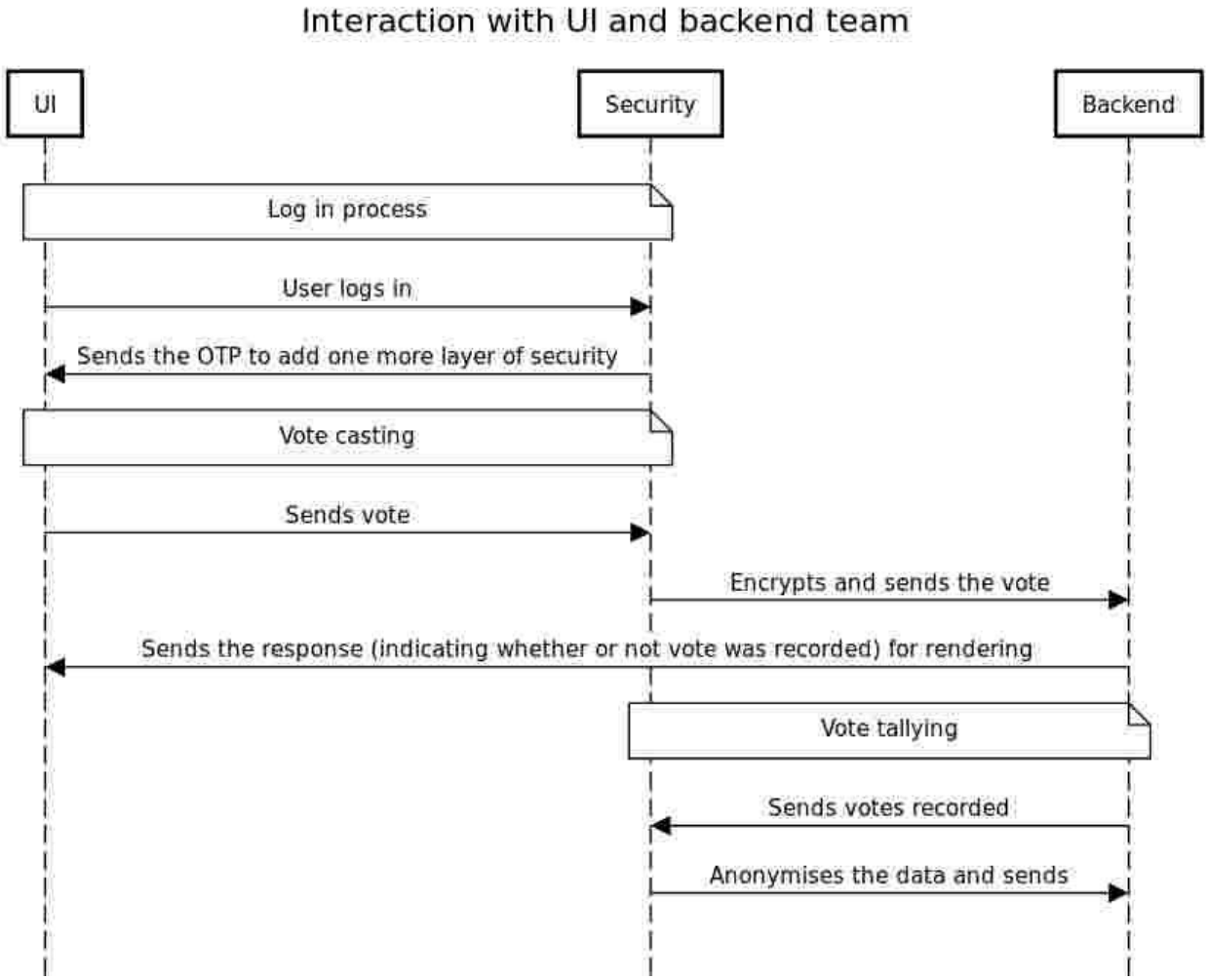
    - Step 5a. The voter casts his or her ballot and sends it to the voting center.
    - Step 5b. The voting center publishes a message about the ballot from the voter to the bulletin.
    - Step 5c. Every voter can confirm whether his or her ballot has been received by the voting center; if not, he or she can resend the ballot.
    -

- Counting
    - Step 6a. At the end of the voting period, the proxy creator forwards the decrypting key to the voting canter, and the voting center starts to verify and count the ballots.
    - Step 6b. The voting center publishes the voting result to the bulletin, where everyone can verify and count all ballots.

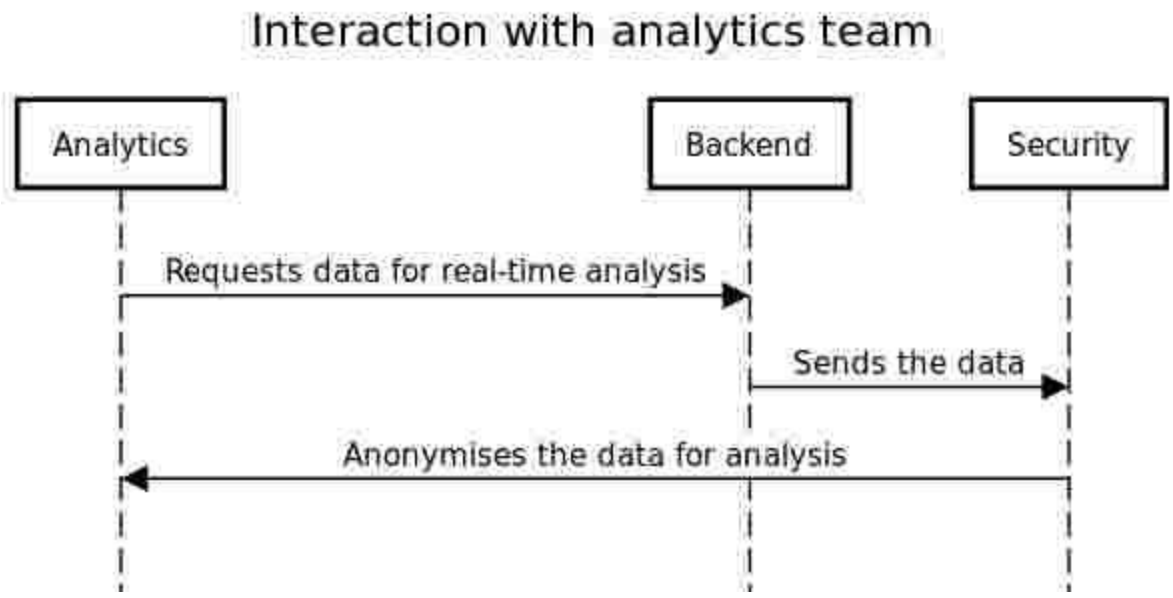
- **Social security**
    - Phone theft - To avoid this problem of phone thievery, what we aim to provide is a theft report portal, where you can enter your aadhar card number and voting for that aadhar card will be disabled. This will ensure that people can't just go about stealing phones in order to cast multiple votes. (How feasible is alternate phone number?)
    - Social bullying - How far this problem can be solved still remains a question. However, to avoid as much bullying as possible, in addition to the above solutions proposed to provide network security, we will ensure that each session expires in a given time which then results in a user requesting to log in again, and hence to some extent might reduce a person bullying a person to log in and then hand him the phone.

Interaction with other subsystems

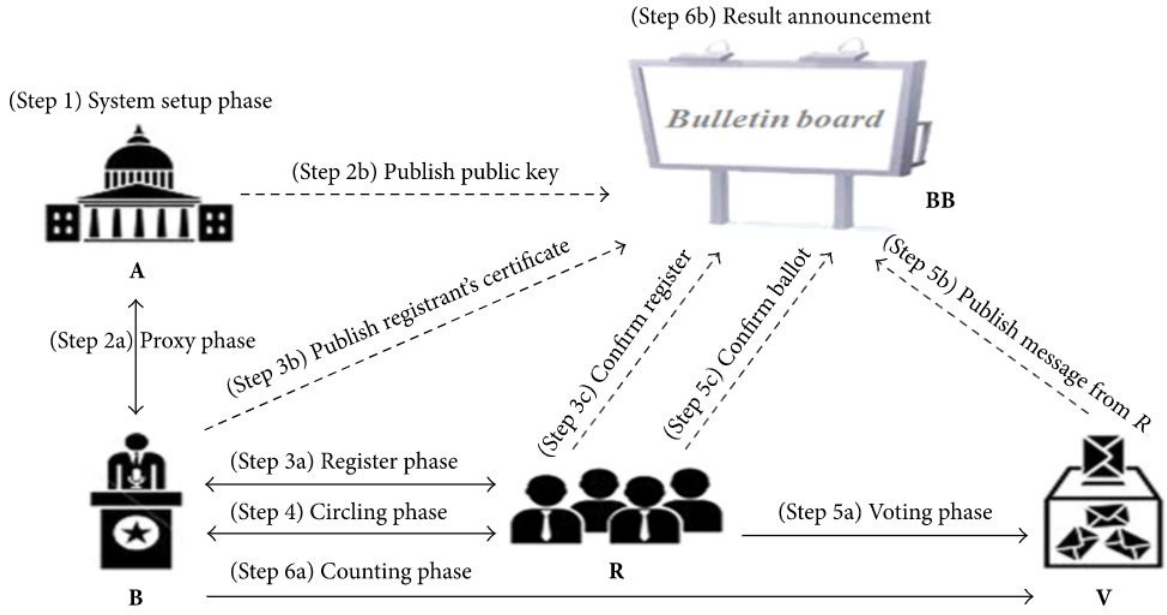
5.1 UI and backend:



5.2 Analytics:



Workflow Diagram:



Case Study

E-Voting in UK

The UK Government tested various E-Voting's systems in the years from 2000 to 2006 , precisely in May 2000, May 2002, May 2003, June 2004 and May 2006. They tried following methods:

- Interactive Voice-Responder
- Digital television Voting
- Mobile telephone (SMS)
- Kiosk Voting
- smart cards for voter identification purposes
- e-counting of paper ballots
- Polling station operated e-register of voters

The security-strategies and control-methods adopted by the UK were:

- To exclude multiple voting, each voter was identified by a unique vote reference number (VNR).
- If a voter used the postal vote, than his e-credentials would be disabled.
- Staff and technicians were given some access to rectify problems at small level.
- In case of a temporary unavailable problem the polling station's staff had to keep paper records of voters who have voted to insert them later in the system and call the election office to verify each voter's eligibility.
- In case of a permanently down system, they had to convert to traditional voting.
- In 2003 the UK Government introduced the smart cards containing the voter's ID to speed up the process of retrieving voter's details.

Problems faced:

- Time and resources needed to set up polling stations underestimated.
- Not always possible to establish a channel of communication from polling stations and election office.
- Insufficient number of back up hardware.

Conclusion:

Following things should be kept in mind while designing any E-Voting system:

1. Only registered voters should be able to vote.
2. Each voter should have one and only one vote.
3. Votes cannot be intercepted nor modified.
4. Votes cannot be known before the official ballot reading.

5. Vote secrecy is guaranteed, it should not be possible to link a voter to his/her vote.
6. The voter should be protected against identity theft.
7. The voting website should resist any denial of service attack.
8. It should be possible to prove that a given citizen has voted.
9. The number of cast votes should be equal to the number of received votes.
10. The system should be audible.
11. The system should not accept votes outside the ballot opening period.

Challenges

8.1. Digital divide

- e-Voting system leads to a financial divide in the sense, that rich people who can afford and buy the technologies will be able to vote while the comparatively less affluent voters who may not be able to afford the technical devices won't be able to vote. Apart from that people who involve more in net banking facilities etc. are more used to using apps for every function. On the other hand, some financially strong people who even though are able to afford the device might not be very comfortable with this system if they haven't used the app based systems for rest of their jobs.

8.2. Privacy

- This challenge is currently being legislated worldwide. We need to balance privacy with the need to gather information that can help address security breaches or fraud, while complying with associated legislation.

8.3. State-sponsored espionage

- This challenge highlights the need to protect critical data from politically or financially motivated threats. Critical data includes the information needed to run network attached infrastructure and drive innovative solutions.

8.4. Social bullying

- Even though we tried to provide a solution for the problem of social bullying, the extent to which that solution solves the problem is definitely not 100%. Implementation of time

Challenges	Manageable/Unmanageable	Solution/Reason
Digital Divide	Unmanageable	The fact that there exists a difference in the price of a smartphone and a normal phone in the market which leads to the digital divide is beyond the scope of the our subsystem and in fact the entire system. Hence, this challenge remains a challenge.
Privacy	Manageable	Secured connections, encrypted votes to prevent hackers from eavesdropping and tampering with the connection
State-sponsored espionage	Manageable (to most extent)	Having a strong security infrastructure to protect the system from such threats.
Social Bullying	Manageable (to some extent)	● Provide network security ● Limited window for voting

Subsystem 5: Awareness and Reaching Out

1. Introduction

1.1 Scope

- The subsystem is only limited to the people who are eligible to vote
- The subsystem makes sure that the percentage of population voting is maximized after using this system.

1.2 Purpose

- As the name of the subsystem itself justifies, the system strives to provide awareness to the people to get adapted and habituated to the current digital world's mobile based voting system.
- To achieve this, the subsystem uses different mediums which enhances the way to reach required target

- This also operates on the time and cost constraints which is an important factor in the current days busy world.

1.3 Overview

- To understand the importance of voting using mobile and convenience of voting
- Make them believe in the security of their vote.

2. Characteristics

- The subsystem will have the following characteristics:
- Proper motivation and volunteering for people to attract more users to use the new system.
- More people to get adopted to the current digital world by proper campaigning on how to use the app and enable users to vote efficiently.
- Time efficiency : Saving a lot of time by not standing in long queues, going to polling booths and getting stuck in traffic. Also fast way to get election results.
- Cost efficiency : No manpower needed and no maintenance costs of the polling booth.
- Increase in the percentage of votes so as to form an efficient democratic government. (As more people vote, the accurate the government formed is)
- More people coverage. Examples:-
    - Many lazy people can't go to polling booths and give their vote. In spite of the vast effort involved in transport, they do ignore.
    - Not all would be free enough to go to their home town and give vote. Not being free include need for job leave, 18+ aged students having examinations at that time etc.
    - In that way, this system can also cover such people.

3. Requirements

- A supportive and engaged community to create awareness among the people.
- Good interactions among different communities
- Finding the person towards whom people will be receptive(Socially influential people).
- Conducting campaigns in the form of advertising and radio.
- The preparation and distribution of communication materials containing key messages on Elections (posters, brochures, …
- Campaigning people should be able to properly communicate with the local people in their respective regional languages.
- Present the information in the languages of each state.

4. Objectives

- Picking the effective social media to spread about the system.
- To sensitize the voters about the importance of participation in an election process to ensure responsive,accountable and democratically elected government.
- To persuade minorities, the homeless, disabled persons, and many others who lack access to the vote for a variety of reasons including poverty, illiteracy, intimidation, or unfair election processes to participate in the election process
- To ensure that people understand their right as voters and exercise that right with full knowledge and responsibility.
- To impart knowledge on voting procedure.
- To ensure that climate or location does not hinder casting of the vote.
- To make the system comfortable from user's side by reducing the effort needed by users. Examples:-
    - Huge rush involved in the public transport which users need to reach the election venues.
    - Elections generally take place in May, the duration of peak summer. In such unfavourable climatic situations also people have to adjust with the environment standing in the long queues. As a result of long exposure, it

may lead to sun-stroke for some delicate people. (Make the system keeping user's health in mind).

5. Implementation

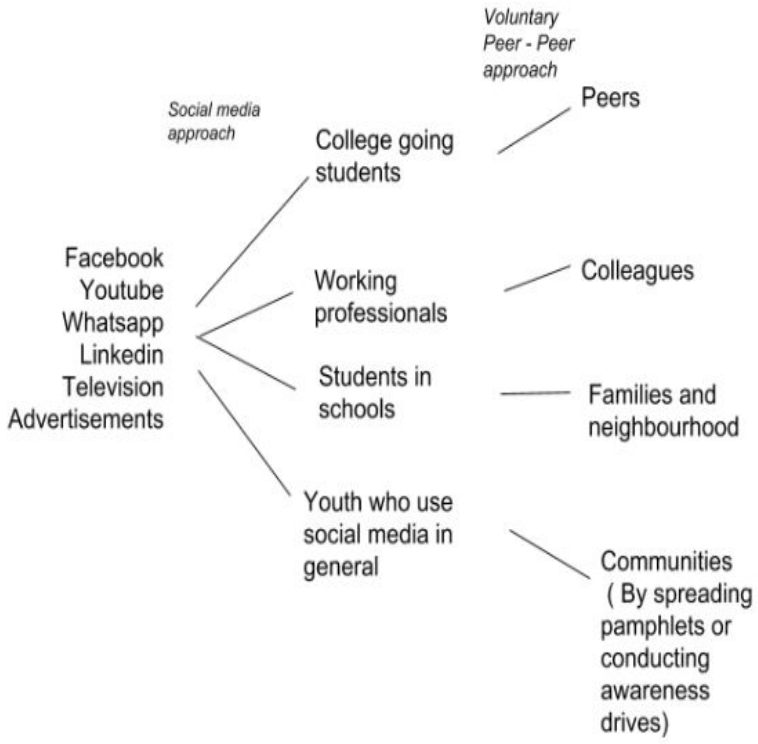
5.1 Sublevel-1:

- Man-Man-approach : There are two ways:
    - One way is via distributing pamphlets using volunteers in public places on the new system keeping the educated in view.
    - Peer-to-peer spreading of information. People knowing about this should be able to spread it to their neighbours. This, just being a conservation would be useful independent of being educated or uneducated.

5.2 Sublevel-2:

- Social-Media-approach :
    - Living in a digital world, people can get proper awareness via social media. Awareness could be provided via advertisements, radio, TV news channels headlines/scrollings or displaying them in news feed online. As most of the people are deeply immersed in social media, this would be a great idea to get the matter spread faster. Alternatively, radios help the uneducated to get awareness about the new system.
    - Most famous social network sites worldwide as of January 2018. Facebook(2167 Million users) , Youtube(1500 Million user),whatsapp(1300 million users).

Block Diagram:



6. Table of Challenges

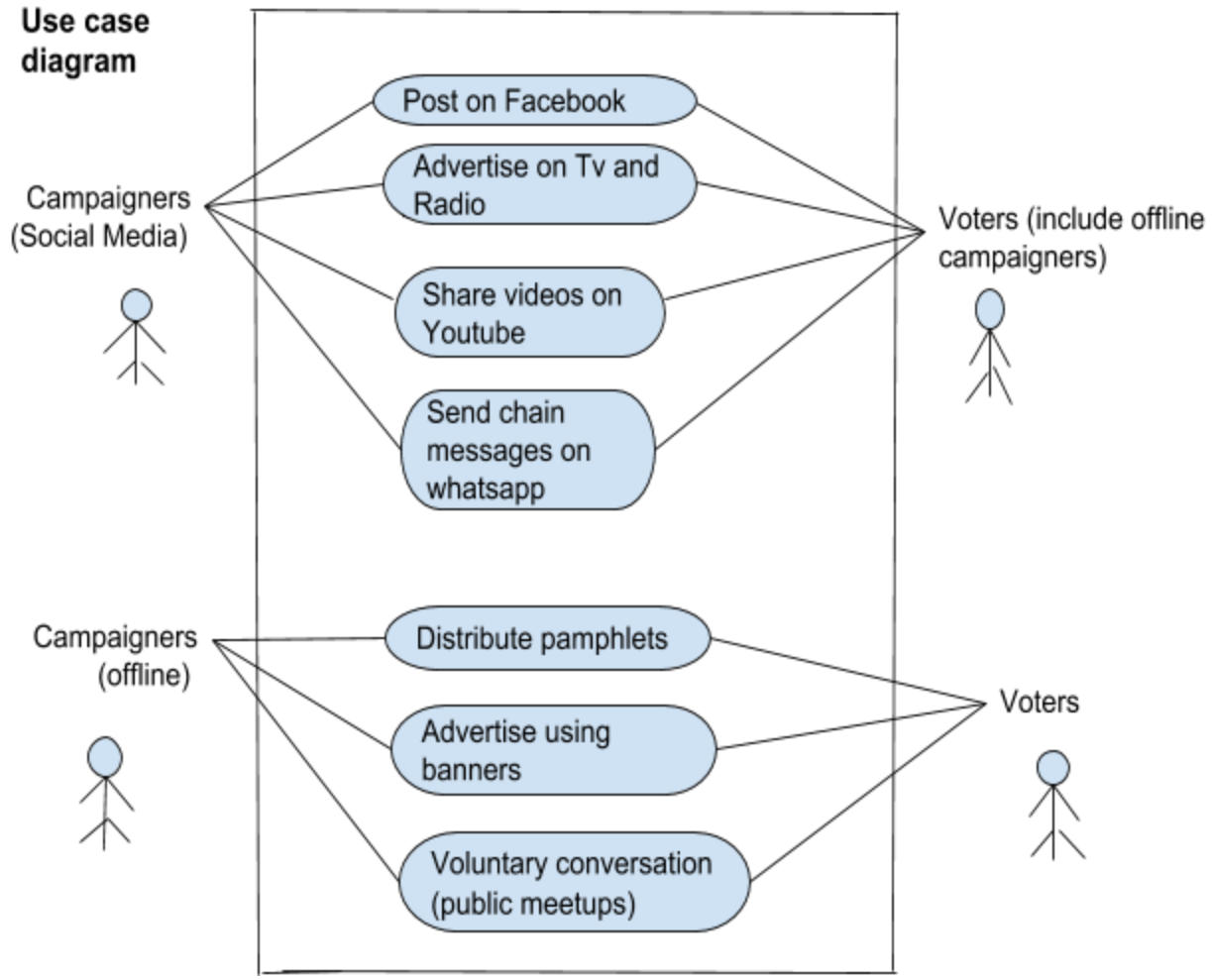
- **Challenges that still remain**
  - Those who can't afford to buy a mobile.
  - Unavailability of proper network in rural areas.
  - The public perception would need to be changed and use of mobile voting system should be encouraged.
  - Takes a lot of time for this system to get implemented.

Table representing challenges:

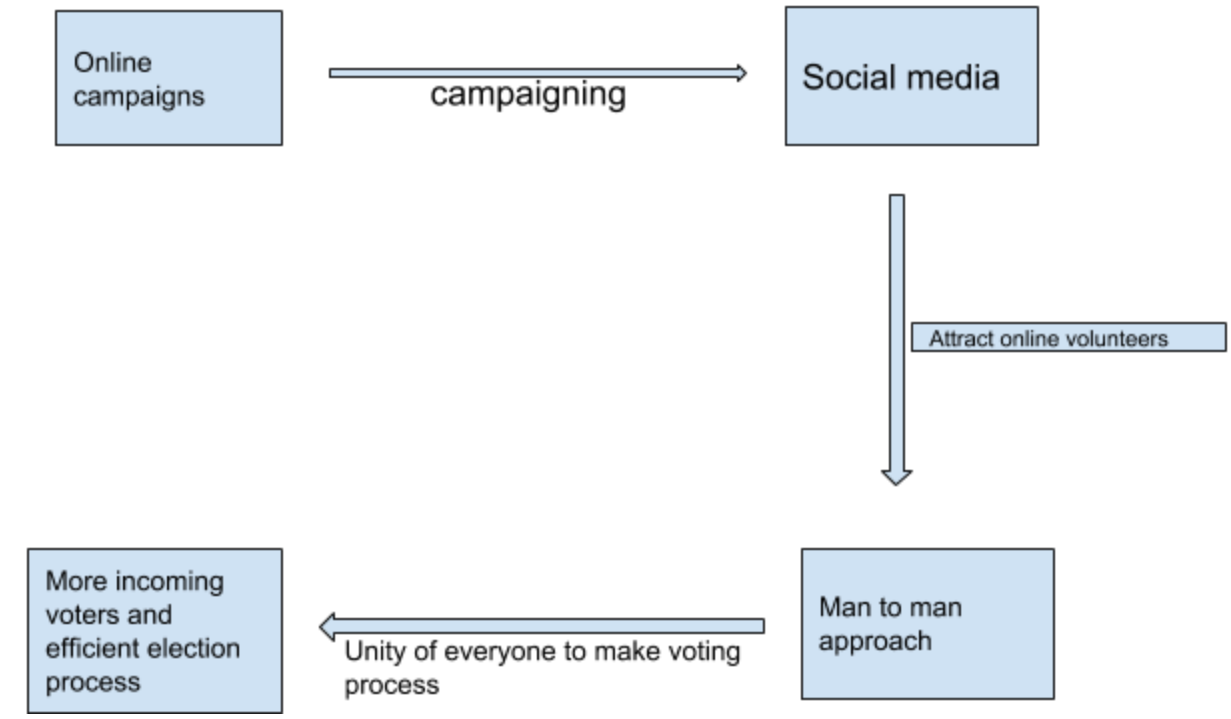
Challenges	Manageable / non manageable	Reason

Inability to buy a mobile	Non manageable	Extreme Poverty
Networking Issues	manageable	Increasing more ISP's
Time Constraint	Non manageable	Takes time to know the importance of the system
Public perception	manageable	Could be achieved by more campaigning

8. Use-Case Diagram



Flow diagram



System Workflow

The general system workflow has been described per-system, in terms of the way one system looks at the dependences on all other systems. Here, we will describe the entire interactions of all systems with each other, for a global perspective. Note that creating a “workflow diagram” for this is quite counterproductive, since such a diagram will necessarily not be able to handle the complexity of interactions that are present between different systems.

We will describe our system workflow in the way the user would perceive it: From the user having to vote, to the user casting a vote, to the chain of events that occurs when the user chooses to caste a vote, and any actions that happen after.

Voting registration

When the user goes to register themselves for a vote, their data is stored in a global database that is maintained by the backend team, and the security is consulted by the security team. We require this pre-voting step to ensure that we know that our binding of person-to-mobile-number is unique.

This could be criticised as the weakest link in the plan. After all, if the youth are reluctant to stand in lines to cast a vote, why would they be ready to perform this?

Our answer is twofold. First, registration for this process is much easier than registering to cast a vote. To cast a vote, one must visit their constituency to perform this action. Also, it is on a particular day at a particular time. Elections being held in the sweltering heat of the summer also diminishes the chances of showing up.

In our system, people can go to *any* center that is close to them to register to vote. Indeed, this will also allow Indians overseas to register to vote, who cannot possibly be expected to return back to their constituency to vote.

The second part of our answer is that as biometric hardware gets cheaper, more accurate, and ubiquitous, it is quite possible that we can skip the registration step and directly allow people to use their biometric data from Aadhar. However, we have not considered this as part of our design currently since it is a very small user base who have phones that have hardware that would enable this. Moreover, this integration is not complicated: Indeed, it would require few changes to the current system to add this feature.

When a person has been registered, they are assigned a unique private ID. The ID is published on a ballot list. No one knows which private ID belongs to whom, thus the system is anonymous. However, this way, a person can still check that they belong to the voting list.

The fact that a person belongs to the voting list can be checked by the voter.

Once a person registers to the online voting list, their name is removed from the regular voting list. This way, we prevent people from double-voting, both electronically and physically.

Vote casting

When the user wishes to cast a vote, they make a request for vote casting to the backend. The backend provides them with an OTP, that is designed to expire in a given time. The user then picks the person they wish to vote for. This data is sent, once again encrypted to the backend, where it is stored. The encryption will use standard public-key cryptography to encrypt the data of the user. The details of the protocols have been described in the sections of the CDR. We can make sure that a person is not double voting by use of the public ledger (more about the ledger will be described below). In short, the ledger maintains a public, anonymised list of votes that can be used by the backend to check that double voting has not taken place.

Once the vote has been cast and frozen, the vote is once again anonymously released on a bulletin. This way, everyone can check that the vote has been cast.

This system is quite reminiscent of a public ledger system such as blockchain. Indeed, we have contemplated using a blockchain like system to maintain the bulletin, to ensure that the bulletin cannot be forged feasibly. However, this was deemed too experimental for the time being. Once the blockchain community has stabilised around a collection of technologies and is shown to be stable, it makes sense to migrate the bulletin system to blockchain to ensure permanence of election records. Such a ledger would still be centralized instead of decentralised. However, the ledger helps prove authenticity, since it is extremely hard to forge a blockchain.

Once the data is stored at the backend, the analytics team has the ability to query the data for near-real-time analytics. The security team helps to anonymise the data from the backend that will be sent to the analytics team. They can then perform data analysis to provide statistics to people. It can also be used to try to detect voter fraud. There have been several such schemes that have been proposed, but none of them have been tested out in real life.

During the vote casting process, we will ensure that psychological effects such as the order of the candidates do not affect the voting process, by randomly shuffling the order of the candidates for different users. This is another unique feature of our system: We can notice patterns from psychology, and we can use these to improve the quality and fairness of our voting system for everyone.

As the election process is going on, the data analytics team will provide the backend team about expected load on the voting systems. This way, the backend team can scale their servers up and down dynamically based on load. Not only is this cheaper in the long run, but also helps in mitigating against DDoS attacks.

Vote tallying

At the end of the voting process, the votes can be tallied accurately using the backend system. Once again, during this process, we will take the help of the security team to ensure that the votes which are cast are anonymised of all personal information.

After this stage, we have also considered making the data open to the public, since such a large dataset (indeed, from the world's largest democracy!) would definitely be interesting to sociologists, game theorists, and political experts, to name a few.

At this stage, the tallied vote can be authenticated by other parties since the vote ledger is public. This helps increase transparency of the election process.

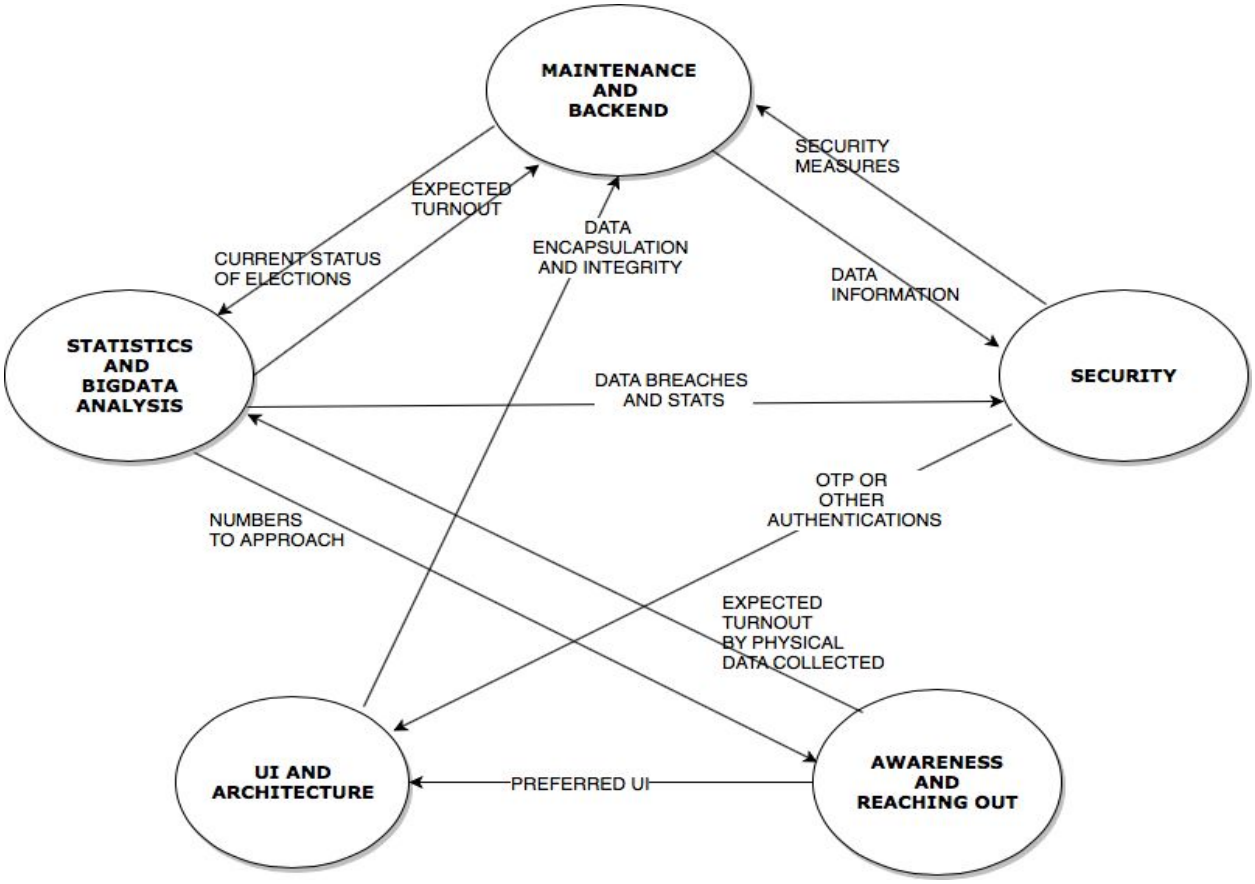
Vote analysis

Once the votes have been tallied and the elections have been completed, the analytics team will perform a "deep analysis" on the data. This can provide valuable information about voting trends within India, as well as help correct for many phenomena.

Outreach and Awareness

Clearly, for this system to succeed, we need to ensure that a critical mass of people start using it. For this purpose, we have an awareness team. They will help spread the word using social media, as well as peer-to-peer interactions within places where youth density is high, such as colleges and companies whose crowd skews young. We hope to attract more people to use the system. For this purpose, we have online campaigns that will take place on social media to attract more voters. We will also have TV ads, chain whatsapp messages that will inform people about this new system to ensure that people are aware of the existence of such a system.

They will also help the UI team to iterate and improve on their UI by helping to perform and interpret the results of constant A/B testing.



Workflow diagram

Self-Assessment of the Design in meeting the System Objectives

- We have implemented security features to maintain the integrity of the election process. This was critiqued heavily by the team and went through multiple rounds of review.
- The UI design has been kept simple and informative.
- We have ensured voter comfort by allowing them to vote from anywhere without having to stand in long queues.
- Paper wastage has been reduced as all our processes are digital. This was something that was not an initial goal, but emerged from discussions.

- Our awareness team has been reaching out to people for their feedback and for spreading word not only about our system but also about the need to exercise their right to vote.
- The effort needed to set up polling booth in every district and especially in remote areas can be utilised in other areas by using our system. The need to have scores of companies of police protection has also been eliminated.
- Our aim of having a greater turnout in the elections by including physically handicapped people and people not in their own wards on the election day will also be achieved by the system.

Learnings from the System Design

- Proper interactions between the subsystems should be there.
- Subsystems should not depend too much on each other. Otherwise, if one subsystem fails, other will also fail in their functionalities.
- Analyze the needs of the end users or customers.
- Produce one or more models of what the system eventually looks like and select the best out of all.
- Solve not only the existing problems, but also come up with acceptable solutions to the problem that may arise in the future.
- Understand the difference between problems that can be solved and interdependent issues which can only be managed.
- Make sure the system succeeds in achieving its primary aims without fail.
- Take regular feedback from the end users and make suitable changes to the system according to their needs.
- Subsystem boundaries must be clearly and rigidly defined, so as to make the system more modular.
- System should be designed keeping in mind its deployment environment as ultimately the system has to perform in the environment where its deployed.
- Primary aim should be to provide a good user experience.
- Make sure the system works well in all possible scenarios. An exhaustive analysis of the system is required to make this happen.

References

- HomeAlone: Co-residency Detection in the Cloud via Side-Channel Analysis.
- This is probably require separate negotiations with Amazon / Google etc to provide machines themselves. Talk about possible problems, solutions.
- Hiding Routing information - David M. Goldschlag, Michael G. Reed, Paul F. Syverson.
- Protecting SIP Proxy Servers from Ringing-Based Denial-of-Service Attacks - IEEE.
- Electronic voting (wikipedia) https://en.wikipedia.org/wiki/Electronic_voting#Documented_problems_with_electronic_voting
- Sandeep Mudana (University of Auckland) https://www.cs.auckland.ac.nz/courses/compsci725s2c/archive/termpapers/sr.pdf
- http://diuf.unifr.ch/main/is/sites/diuf.unifr.ch.main.is/files/file/studentprojects/reports/eGov_HS07_Case_Studies_on_E-Voting_(DanieleCavadini_LorenzoCimasoni).pdf
- http://www.socialfish.org/2012/08/online-voting-advantages-and-challenges-for-associations/

- http://www.elections.in/political-corner/voters-awareness-and-its-importance/
- https://docs.google.com/forms/d/1XBczw_ysJalQBCNGlSP3eW2mbhDJoBooTi89oQpxL1U/edit?ts=5accc696
- http://usabilitypost.com/2009/04/15/8-characteristics-of-successful-user-interfaces/
- http://www.it.bton.ac.uk/staff/lp22/CS133/usergroups.html