Software Requirements Specification

for

ExeC

Version 1.0

Prepared by: UNiX Name

Group #: 5   **Group Name: UNiX NAME**

Aryan Sharma	200203	aryans20@iitk.ac.in
Kunwar Preet Singh	200536	kunwarps20@iitk.ac.in
Parinay Chauhan	200667	parinayc20@iitk.ac.in
Pratyush Gupta	200717	gpratyush20@iitk.ac.in
Harshit Bansal	200428	harshitb20@iitk.ac.in
Jaya Gupta	200471	jayagupta20@iitk.ac.in
Suket Raj	201013	suketraj20@iitk.ac.in
Soham Samaddar	200990	sohams20@iitk.ac.in
Aditya Tanwar	200057	tanwar20@iitk.ac.in
Akhil Agrawal	200076	akhilag20@iitk.ac.in

Course: CS253

Mentor TA: *Mr. Pinaki Chakraborty*

Date: 30-01-2022

Contents

Revisions

Version	Primary Author(s)	Description of Version	Date Completed
Draft Type and Number	Full Name	Information about the revision. This table does not need to be filled in whenever a document is touched, only when the version is being upgraded.	00/00/00

1 Introduction

1.1 Product Scope

The purpose of the web app is to ease and streamline the elections held by the Election Commission of IIT-K. The app will help digitalize the process by automating the different stages in an election. This application will feature user validation and login, candidate registration, information regarding the prospective candidates such as contact information, proposers, seconders, manifestos, etc. The app will also facilitate anonymous communication and reporting during the elections. The aim is to help the general body members get to know their candidates better and aid them in making an informed choice. The app will also help the Election Commission by including a common portal for reporting, tracking, and dealing with penalties.

1.2 Intended Audience and Document Overview

This document is intended for everyone involved in the development process including but not limited to the course instructor, our TA, our team and the team that would conduct the beta-testing of our application. The intended audience of the finished product is everybody involved in the elections conducted at IITK, from roles ranging from EC, GBM, Candidate to the general public audience who wishes to scrutinize the candidates.

Section 1:
This has some basic information that would be useful in reading the SRS such as document conventions, abbreviations, etc. The reader may choose to skip the section if they are familiar with some basic terminology. In any case, this section will serve as a useful collection of information to clarify any confusion that may occur while reading the document.

Section 2:
This section offers a bird's eye view of the software system and its functionalities, assumptions and dependencies. This will be a useful read for those seeking to familiarize themselves with the system at a quick glance. A reader is encouraged to read this part as it provides a good basis for understanding the next section of the SRS.

Section 3:
This section contains detailed information about the software and explains the functions in detail through the use of multiple diagrams. This is essential for end-users, clients and developers as it will serve as a guide in the development process and also an instruction manual for end-users.

Section 4:
Important non functional requirements are expounded here. This is of special importance to the developers of the software.

1.3 Definitions, Acronyms and Abbreviations

- **EC: (Election Commision of IIT Kanpur)** - The body responsible for the conduction of elections in IIT Kanpur. The members of the Election Commission would act as the admin in our application.
- **GBM: (General Body Member)** - All the people in IIT Kanpur who are eligible to vote in the elections held here constitute the General Body Members
- **OARS: (Online Academic Registration System)** - One of the websites managed by IIT Kanpur. It contains the information of all the students in IIT Kanpur and we think of extracting the details of all students to feed in the application database.
- **COC**: Code of Conduct
- **API:** Application Programming Interface
- **DBMS**: Database Management System
- **HTTPS**: HyperText Transfer Protocol Secure
- **JSON**: JavaScript Object Notation
- **MTTF:** Mean Time To Failure
- **SSL**: Secure Sockets Layer
- **POV**: Point of View
- **SRS:** Software Requirements Specification
- **UCM**: Use Case Model
- **UI**: User Interface

1.4 Document Conventions

While preparing this document, to make readability user-friendly and important parts clear, we have used following conventions:

- The headings of all the sections are written in **bold** and underlined using Arial font size 14.
- The headings of subsections are **bold** and written in the same font as that of content using font size 13.
- The content of the section is written in Arial font size 11.
- Any important term or short form is written in **bold**.
- The alignment of the whole content is **justify**.
- Text has been indented wherever required to highlight the hierarchy of the content.
- The document follows the IEEE formatting, indenting and numbering conventions. Any deviations from the same will be explicitly specified

1.5 References and Acknowledgments

Style guide (for this document): IEEE Editorial Style Manual (Online)
OARS IITK: https://iitk.ac.in/new/index.php/oars
Cryptographic Security for Anonymous Communication:
https://link.springer.com/chapter/10.1007%2F3-540-45682-1_32

2 Overall Description

2.1 Product Overview

It is a new and self contained product which is proposed as a solution to the issues prevalent in the institute in relation to the election time and procedures by the campus population. Our product provides a level playing field for each participating candidate irrespective of reach since the sharing of manifestos, videos, posters, etc. can be regulated by EC allowing everybody to contest solely on the basis of the promises made.

Until now, most of the campaigning has been done via social media, with candidate manifestos and campaigning materials spammed across groups. This makes it very difficult to assimilate all available information, and requires concerted effort. Additionally, spread of misinformation and rumours are also possible. Our product seeks to solve this problem with its regulated environment. It also provides the Campus Junta with a centralized information resource regarding all the candidates in an election, allowing them to take a more informed decision when voting.
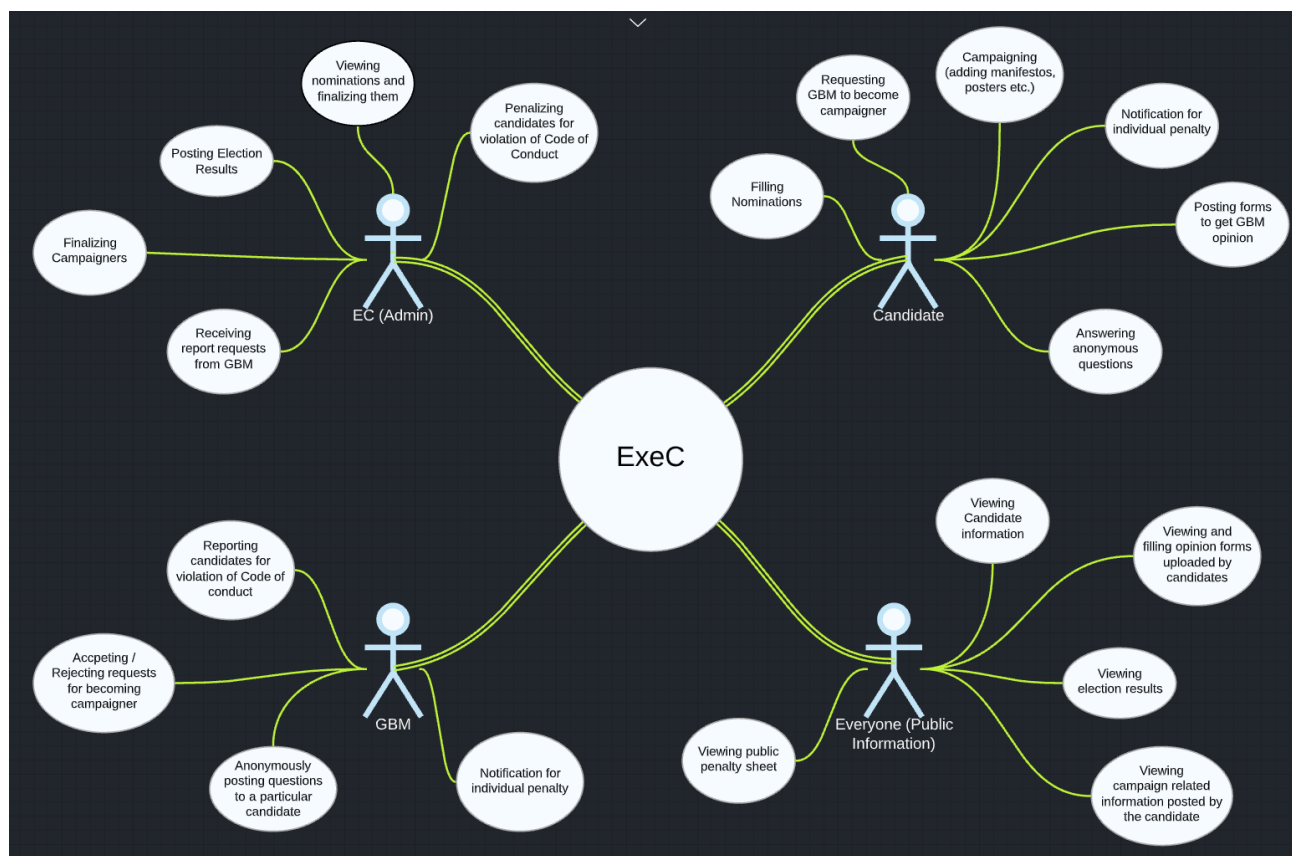
2.2 Product Functionality

- Login and Sign-up for all the involved parties using Roll number and password.
- Anonymous communication: Anyone can contact a candidate to ask questions while maintaining his/ her anonymity.
- Every candidate will receive a notification when a penalty is imposed on him/ her which will be displayed on his/ her dashboard post login.
- Anonymous reporting: Any person can anonymously report any activity against the spirit of free and fair election.
- Anyone can approach a candidate with a request of campaigning for him/her and vice-versa.
- Portal for signing up as a candidate.
- Candidates can send their manifestos for approval to the admin post login.
- Admin can send out penalty notifications and approve (or cancel) candidature.

2.3 Design and Implementation Constraints

- Student database forms an integral part of the relevance of the software, the responsibility of timely updation via approving or rejecting the update requests lies solely with the system admin.
- Server should have sufficient memory and resources available in order to accommodate and serve all the requests and the data requested by the users concurrently.
- The OARS server must be available at the time of creating an account as ExeC would make a request to it in order to retrieve information of the signee.

2.4 Assumptions and Dependencies

- The number of users who use this application shall never be greater than 10,000.
- The number of candidates for all posts combined shall never be greater than 500.
- At any point of time, there shall be at least a single account with admin privileges for regulation of the platform.
- Upcoming new technology shall not affect the execution of the application.
- The OARS shall not make significant changes to the current method of storing student information as its database shall be used as the source for student data.
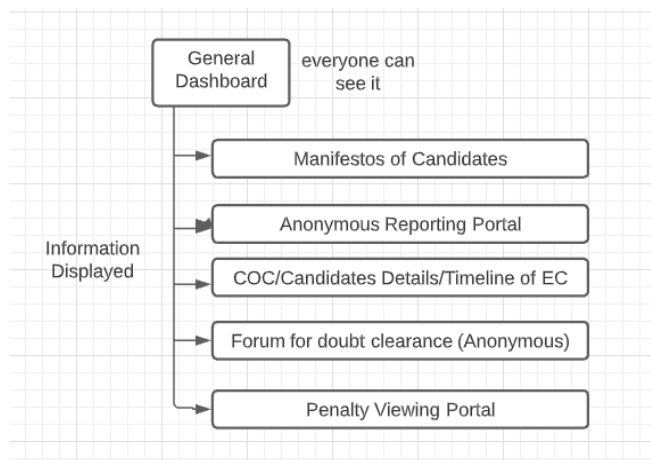
3 Specific Requirements

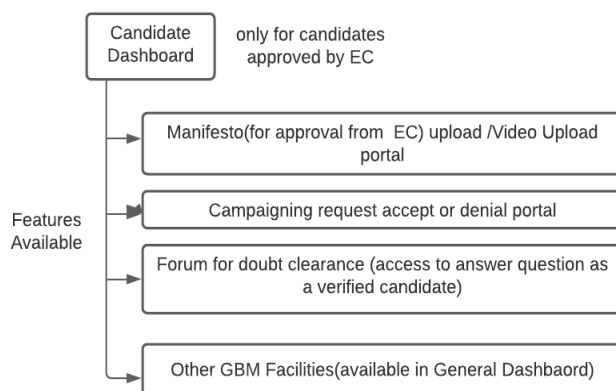## 3.1 External Interface Requirements

3.1.1 User Interfaces
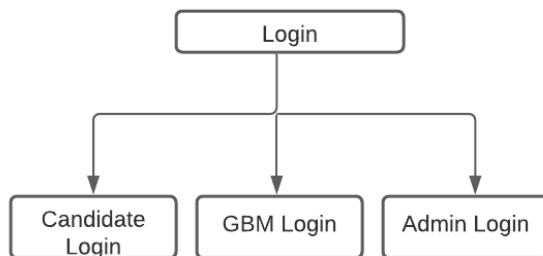
Three different login pages; one each for **Candidate**, **GBM** and **Admin**.

General Dashboard will be publicly accessible. It will contain links to other pages, and display manifestos of candidates.

Candidate Dashboard is restricted to authorized candidates, contesting for the elections.

It will allow candidates to upload their manifestos for approval, accept/reject campaigning requests and other general facilities for better user experience.
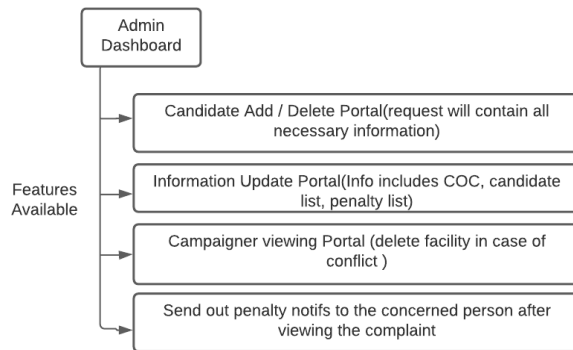
Admin Dashboard is the most protected part of the app which will be accessible only by the EC admin.

It will feature superuser actions like approving candidates and their details and uploading verified content like manifestos, election timeline.
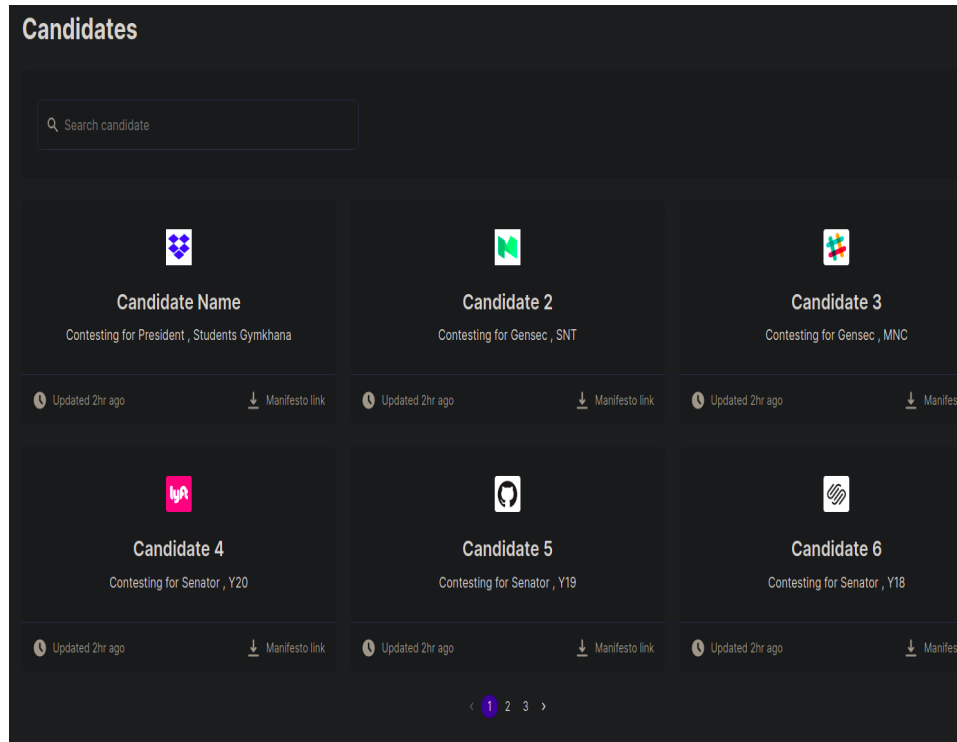
1. UI DESIGNS

- ● SIGN-UP/SIGN-IN PAGE

Portal to register and verify GBMs and candidates, after which a dashboard will be created for every registered person.

- DASHBOARD

General Dashboard which every GBM can access.

Will be able to view the Candidate list, their manifestos and related information.

- ANONYMOUS QUESTIONING (to any candidate)

Accessible to all GBM members. For asking questions to any candidate anonymously

- FORUM

Discussion Forum where questions asked to different candidates will be visible

3.1.2 Hardware Interfaces

- Support for modem and Ethernet card – that is, appropriate drivers of compatible modem and Ethernet card are installed for accessing web pages over the internet or intranet.
- The server side of the system must be connected to the IITK intranet.

3.1.3 Software Interfaces

The server-side components of the software system must operate within a Linux operating system environment.

The client-side components of the software system must operate within common web browser environments using Secure Sockets Layer (SSL) / Transport Layer Security (TLS) cryptographic protocols at a minimum encryption level of 128 bits. The minimum set of browsers that must be supported is:

- Apple Safari 7+
- Google Chrome 44+
- Microsoft Internet Explorer 10+
- Mozilla Firefox 40+

The server will be integrated with two separate databases, using MongoDB Atlas as the DBMS, one for storing the user (both GBM and candidate) related information, campaign related information and penalty sheet along with others. The other database would be used to store the authenticity related information, including the username, encrypted password and the temporary codes required to sign in.

The database and its management systems would be deployed only on the server side, the final client application delivered would not be independent of this.

3.2 <u>Functional Requirements</u>

3.2.1 The system shall provide an easy interface for the election commission for regulation of elections and information dispensing.

The EC shall be able to view and receive nomination applications filled out by candidates. The EC shall reserve the sole right to approve or reject these candidatures as it deems fit. The EC shall also be able to approve or reject candidate posters, manifestos and other material such as videos.

The EC shall also be able to publish information for both candidates and GBM as it deems appropriate. This information shall be visible to the targeted audience on their own dashboards on the application.

Information about penalties and fines imposed shall also be sent out to the affected parties. It shall also be possible for everyone to access the list on their dashboards.

If there are any clashes such as a person signing up for campaigning for two competing candidates, the EC shall be notified and an attempt such as this shall be resolved/ blocked.

3.2.2 The system will streamline the process by providing a single platform for the candidates.

The candidates are supposed to fill out the form floated by the EC for their nomination. Once their nomination is approved by the EC, they will be provided the candidate level access to the application and the required authentication credentials. They will not be able to login as a normal user.

The candidates can prepare and upload their manifestoes and posters on the app. Following which, the poster and manifesto will be sent to the EC for approval. The status of the procedure will be displayed to the candidate. Once the manifesto has been approved it will be visible to all the GBMs. The same applies to other material such as videos. Final result of the Election will also be displayed on the app.

Candidates will have an option of putting up forms for collecting the grievances of the GBMs. The responses received on the aforementioned forms will be forwarded to the candidate who floated the form.

The candidate will be able to see the questions posed by the GBM and other candidates and can choose to answer these questions at his own will.

3.2.3 The application will provide a convenient platform for GBMs.

Every GBM will be able to login by default and will be able to change the authentication credentials when logged in.

Every GBM will be able to ask questions to a candidate contesting for any post that the GBM will be voting for. The question can also be asked anonymously and will be forwarded to the candidate in question. All details of the contesting candidates such as their manifestos, posters and other material will be visible to every GBM.

Notifications about penalties imposed on the person will be displayed on their dashboard. An anonymous reporting portal would also be available. Anyone who feels that there has been a violation of the CoC can make an anonymous complaint along with the evidence to the EC on the portal itself.

Any person can make a request to a candidate to become a campaigner for them. Any attempt to campaign for multiple people competing for the same office will not be entertained by the system.

3.3 Use Case Model
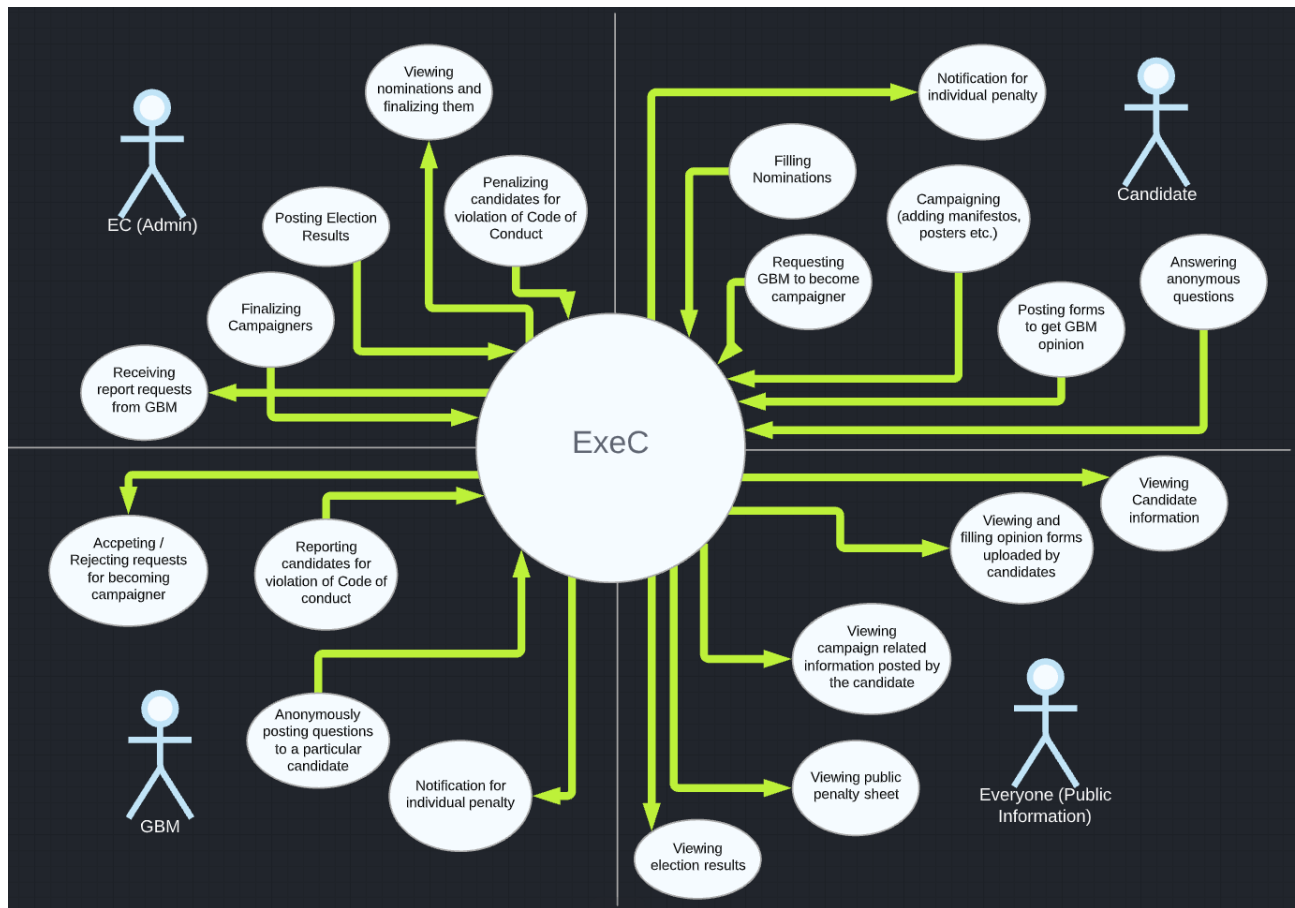
The above use case diagram provides an overall view of the processes and actors that our application will be dealing with. The main actors specified here are:

- **Election Commision** - The main governing authority of the elections as well as this application (i.e. will have the admin access)
- **Candidate -** The candidates aspiring to contest the elections and are successfully nominated for a particular post
- **General Body Members** - All people of IIT Kanpur eligible to vote in the elections of student gymkhana
- **General Public -** All people visiting the home page of this application. There is no authentication required to view the information under this domain.

3.3.1 Use Case #1 (U1)

Author – This use case was written by Aditya Tanwar

Purpose - To display the various actors and requests involved in the process of filing nomination by a candidate.

Requirements Traceability – The candidate must have with them all the details required in the filling of the nominations, asked by the Election Commission

Priority - The priority of this use case is high as it is one of the main features of this application and any problems in this use case will defeat one of the main purposes of this application.

Pre conditions - The nomination filing (election season) must have started from the side of the admin, here Election Commission

Post conditions - The candidate is nominated for a particular post after the confirmation from the Election Commission

Actors – Actors in this use case are the aspiring candidate and the EC members acting as the admin.
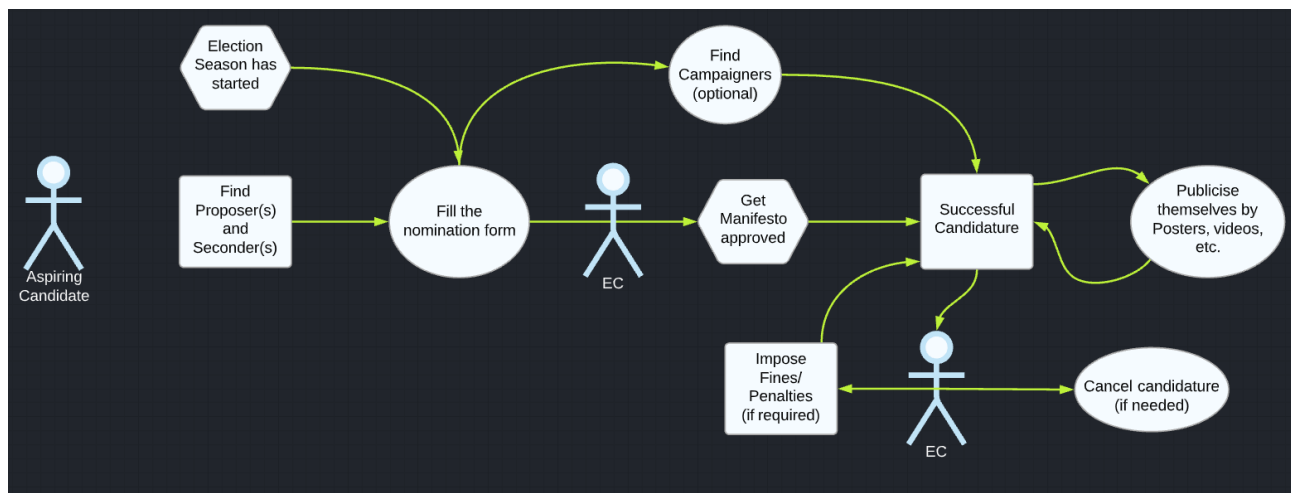
3.3.2 Use Case #2 (U2)

Author – This use case was written by Aditya Tanwar

Purpose - To explain the interactions between the system and the various category of users involved in the registration of the a GBM as the campaigner of a candidate

Requirements Traceability – The candidate and the general body member to whom the request is sent must have signed up in their respective portals.

Priority - Priority of this use case is medium, as if this system fails, it will affect a group of information but won't bring down the whole application.

Preconditions - The GBM to whom the request is sent must not have accepted the campaigning request from any other candidates

Post conditions - The GBM would be added to the list of campaigners of the candidate

Actors – The actors involved in the use case are the candidate, the GBM and the EC member, acting as the admin to finalize the request.
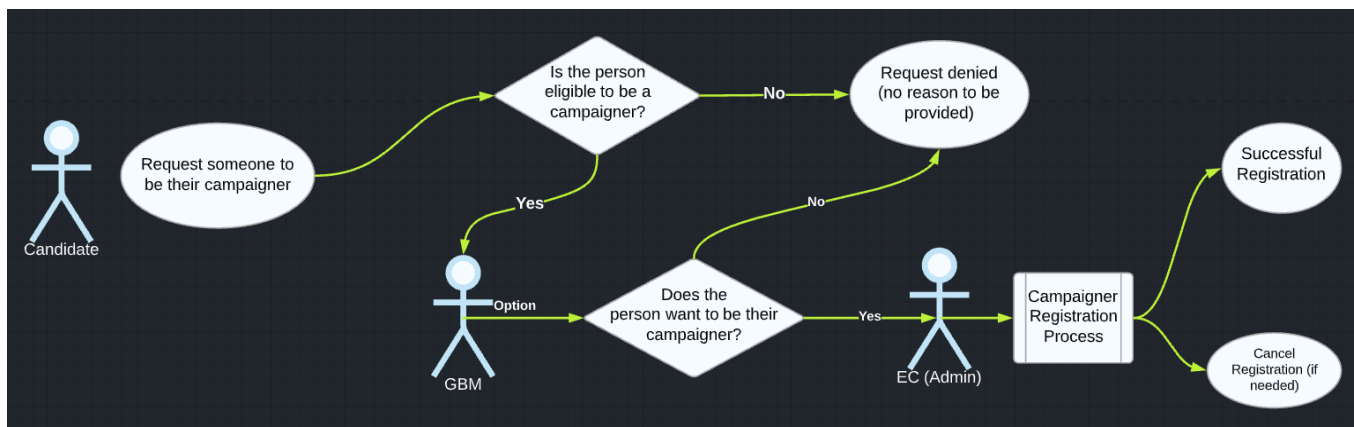
3.3.3 Use Case #3 (U3)

Author – This use case was written by Parinay Chauhan

Purpose - The given use case describes the process of reporting a person to the EC against the violation of the prescribed code of conduct.

Requirements Traceability – The person reporting must have signed up for his/her general body member (GBM) portal

Priority - The priority of this use case can be considered as low

Preconditions - The person who wants to report must login into their respective GBM portal.

Post conditions - The report is sent to the Election commission (Admin) dashboard along with the information in the report and the details of the sender.

Actors – The actors involved in this use case are the reporter (GBM) and the Election Commission members, acting as the admin.
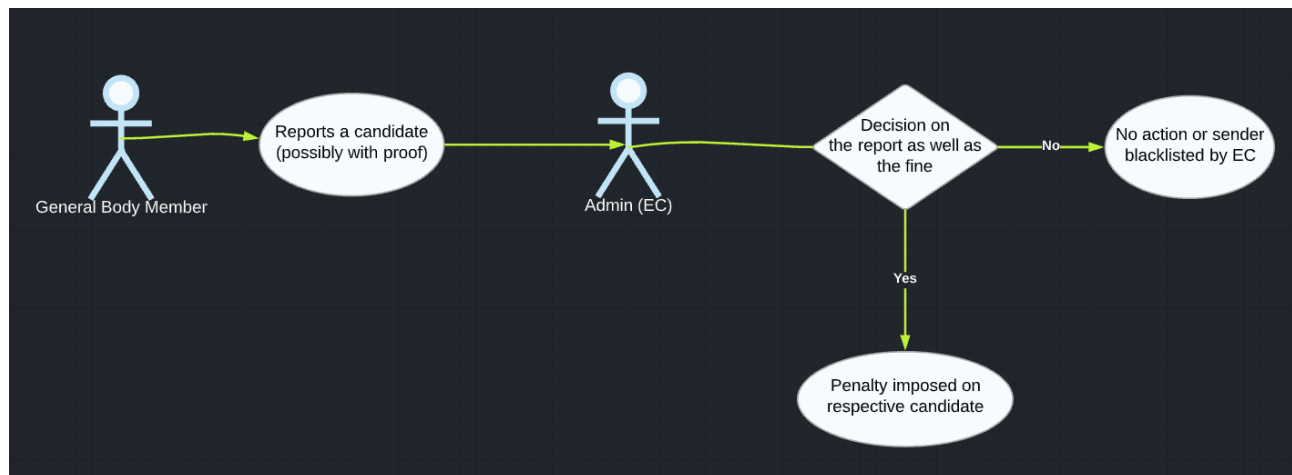
3.3.4 Use Case #4 (U4)

Author – This use case was written by Parinay Chauhan

Purpose - This use case is meant to describe the process of penalization of a candidate from the Election Commission

Requirements Traceability – The GBM must have signed up in their respective GBM portal in order to receive the dashboard notification, nevertheless the information could still be visible to them via the public penalty sheet

Priority - The priority of this use case can be considered as medium.

Post conditions - The information of the penalization gets updated on the public penalty sheet and individual notifications are sent to the person being penalized.

Actors – The actors involved in this use case are the EC members acting as the admin, the penalized person and the general public.
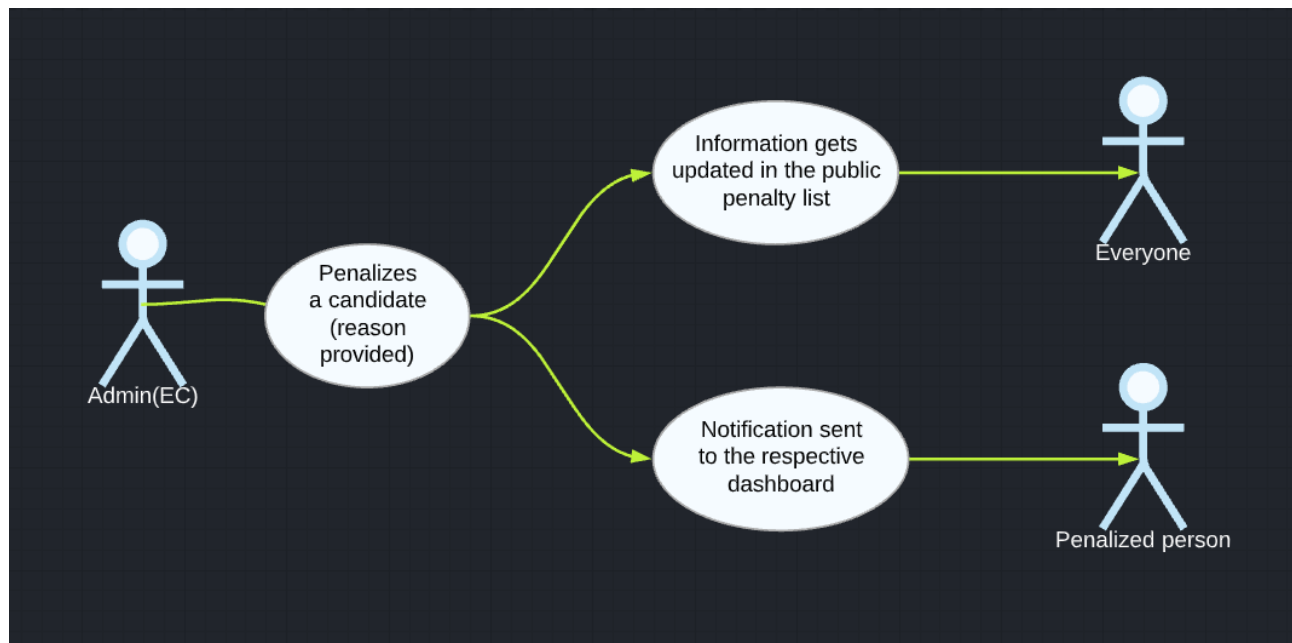
4 Other Non-functional Requirements

4.1 Performance Requirements

- The ExeC must provide available fresh data to the requesting client at a rate of no less than 1 Hz.
- The response-time for the Exec should be minimized using industry recommended practices.
- All other performance related to storage, memory, and processing should follow industry recommended practices to ensure resource requirements are minimized.

4.2 Safety and Security Requirements

- The client-side components of the software system must operate within common web browser environments using Secure Sockets Layer (SSL) / Transport Layer Security (TLS) cryptographic protocols at a minimum encryption level of 128 bits. The minimum set of browsers that must be supported is: • Apple Safari 7+ • Google Chrome 44+ • Microsoft Internet Explorer 10+ • Mozilla Firefox 40+
- Secure storage of passwords will be done after salting and hashing of the passwords. In other words, the system admin can only verify if the password is correct, but won't know what the password submitted by the user was.
- There will be a CAPTCHA system to avoid DDOS attacks.
- Periodic Backups of the entire data of the platform will be taken.
- Anonymity in anonymous communications will be cryptographically ensured using ring signatures and open APIs.
- Request Handling by server will be rate-limited so avoid illegitimate requests from flooding the request queue.

4.3 Software Quality Attributes

4.3.1 Maintainability

- The architecture, design, implementation, and documentation of the software must be such that they make the system reduce the maintenance overhead as much as possible.
- Fixing a security defect, including updating of the documentation and testing, must not take more than two person days.
- The average work time required to add a minor feature, including documentation update and unit testing, should be less than one person a week.

4.3.2 Availability

- The system will only see utilisation during the months of the election cycle. During this time, it will see extensive usage. Hence, the system availability needs to be enough for this purpose.
- In case of a server crash, the system state must be restored within two hours.

4.3.3 Reliability

- The MTTF shall be more than one week. In addition the system must undergo extensive feature testing, load testing, and regression testing prior to release and/or deployment.

5 Other Requirements

- Permission will be required for utilising OTPs for the authentication process as this will require sending automated emails to the student;'s IITK email id.
- A software system in use by the Election Commission of IITK may be subject to confidentiality requirements. Making such a software open source will need permission for the administration.
- Permission of the OARS will be required in order to fetch student data for creation of accounts on our portal.
- Two databases would be required as explained above, one for storing the authenticity related information like username, passwords, the one temporary tokens required for logging in etc. and another for managing the user related information, the campaign related information and the penalty sheet among others.

Appendix A – Data Dictionary

S. No.	Type	Name	Description
1)	Actor	Election Commission	The main governing authority of the elections as well as this application (i.e. will have the admin access)
2)	Actor	Candidate	The candidates aspiring to contest the elections and are successfully nominated for a particular post
3)	Actor	General Body Members	All people of IIT Kanpur eligible to vote in the elections of student gymkhana
4)	Actor	General Public	All people visiting the home page of this application. There is no authentication required to view the information under this domain
5)	State	CoC inactive	No activities regarding elections except reporting
6)	State	Blackout period	Candidate info is online, campaigning not allowed officially, reporting allowed
7)	State	Nomination period	Aspirants can apply for nomination for the position they're eligible
8)	State	Campaigning period	ExeC is fully functioning

Appendix B - Group Log

Date	Timings	Duration	Minutes
10/01	10:00PM-12:00AM	2hr	- Idea suggestions by everyone. - Two main ideas were presented. - One idea was a software system for building teams for courses. - The second idea was developing an application for streamlining the election process at IITK.
13/01	6:30PM - 9:30PM	3 hr	- Preliminary Discussion on infeasibility and usefulness of ideas - It was commonly felt that a great number of issues were faced during elections. - A software solution for this problem was seen as a very pressing need - Finalisation of the idea of app for EC.
19/01	7:00PM - 12:00AM	5 hr	- Brainstorming on our final idea and discussion on possible use cases, features, data flows. - Exploration of possible implementations, challenges, tech stacks - It was decided to use Java for building the backend for our web application
24/01	9:00PM - 11:00 PM	2hr	- The system will have three access levels. - The levels will be, GBM, candidate, and admin each with their separate functionalities.
28/01	7:00PM-11:30PM	4.5 hr	- The functionalities for each access level were decided upon. - The UI/UX interface was finalized for each level.
29/01	7:00 PM - 12:00AM	5 hr	- Section 1 and section 4 of the SRS were nearly finalized. - Functional requirements were formally stated - Construction of use case models was distributed and started. - A UI UX model was distributed and started as well.

30/01	6:00 PM-12:00AM	6 hr	- Construction of Use case Models was completed and agreed upon. - Construction of UI/UX design was completed and presented to the rest of the members. - The incomplete Section 1 and Section 4 were completed. - Finalisation of the SRS