**Software Requirements Specification**

**for**

**Online Voting System(OVT)**

**Proposed by:**

**Sonali Agrawal**

**114CS0634**

**8th February, 2018**

**INDEX**

1. **Introduction**
   1. **Purpose**
   2. **Scope**
   3. **Definitions, Acronyms, and Abbreviations**
   4. **Overview**
2. **Overall Description**

**2.1Product Perspective  
2.2 Product Functions  
2.3 User Characteristics  
2.4 Principal Characters  
2.5 General Constraints  
2.6 Assumptions and Dependencies**

1. **Specific Requirements**

**3.1Functional Requirements  
3.2 Performance Requirements  
3.3 Design Constraints  
3.4 Future Extensions**

**1. Introduction**

1.1 **Purpose**:

Online Voting System (OVS) is intended to provide an easier, more convenient alternativeto the tedious costly offline voting process of an election.It provides a platform for voters to cast vote online, election commission to approve candidates based on their documents, and candidates to share their profiles and accomplishments with their constituency.With automated vote counting, results declaration, and statistics generation, this software aims to completely replace the traditional election procedure.

This document is meant to delineate the features of OVS, so as to serve asa guide to the developers on one hand and a software validation documentfor the prospective client on the other

**1.2 Scope:**

We describe what features are in the scope of the software and what arenot in the scope of the software to be developed.

In Scope:

a) Authentication of users (candidate/voter/admin).

b) Registration of electoral candidates and voters.

c) Verification of candidate profiles.

d)Profile management of electoral candidates, which includes uploadingpersonal details and milestones.

e)Provision for elections and automated vote counting.

f) Managing elections and finding winners from each area.

g) Cancelling or pausing elections in case of security breaches or violation of other laws as decided by the EC.

h) Generation of statistics after election.

Out of Scope:

a) Any election related prediction.

b) Evaluation of candidate profiles.

c)Automatic verification of uploaded documents.’

d) Registration of candidates in the election. Registration to an election will be done offline through the official process as stated by the EC.

**1.3 Definitions, Acronyms, and Abbreviations:**

*Acronyms and Abbreviations:*

a) OVS: Online Voting System.

b)EC: Election Commission.

c)SRS: Software Requirements Specification.

*Definitions:*

1. Election: A formal and organized choice by vote of a person for apolitical office or other position.
2. Election Commission: An independent body set up by the Indian Parliament for the purpose of organizing and monitoring elections.
3. Voter: An Indian citizen above the age of eighteen, who has registered himself/herself with the Electoral Commission for the purpose of voting.
4. Candidate: An Indian citizen above the age of thirty, who has voted before, and wishes to represent his constituency in an election.

**1.4 Overview:**

The rest of this SRS is organized as follows: Section 2 gives an overalldescription of the software. It gives what level of proficiency is expectedof the user, some general constraints while making the software and someassumptions and dependencies that are assumed. Section 3 gives specificrequirements which the software is expected to deliver. Functional requirements are given by various use cases. Some performance requirements anddesign constraints are also given

**2. Overall Description:**

**2.1 Product Perspective:**

OVS is aimed towards providing an online platform for conducting elections in an efficient and cost-effective way. OVS should be user-friendly,easy to use, and reliable for the mentioned purpose.

OVS is intended to be a stand-alone web-based application and shouldnot depend on the availability of other software other than a modern webbrowser. It should run on UNIX, Windows and IOS based platforms.

**2.2 Product Functions:**

|  |  |  |
| --- | --- | --- |
| **Class of use cases** | **Use cases** | **Description of use cases** |
| Use cases related to system authorization | 1.Login User | 1. Log into OVS. |
| Use cases related to registration | 1. Verify candidate’s documents 2. Register Candidate 3. Register Voter | 1. Admin verifies documents uploaded by candidates. 2. Registers a candidate. 3. Registers a voter. |
| Use cases related to candidate information | 1. View Candidate’s data 2. Upload Candidate Information | 1. System allows voters to view candidate’s data. 2. Provision for logged in candidates to upload details and milestones. |
| Use cases related to election | 1. Create Election 2. Vote for Candidate 3. Calculate Result 4. Cancel Election 5. Pause Election 6. Resume Election | 1. Create a new election 2. Provision for voters to vote for candidates 3. Calculation of results. 4. Cancels ongoing election 5. Pauses an ongoing election 6. Resumes a paused election. |

**2.3 User Characteristics:**

a) User should have sufficient expertise to use a web-based application.

b) All voters and candidates must possess a valid Voter ID card.

**2.4 Principal Actors:**

The principal actors in OVS are “admin”, “voter”, “candidate”, and the“system”

**2.5 General Constraints:**

a) For working, OVS requires an internet connection.

b) OVS is a web-based software.

**2.6 Assumptions and Dependencies:**

a) Full working of OVS is dependent on the availability of an internetconnection.

b) OVS can only be used to vote in selected areas.

**3.Specific Requirements:**

**3.1 Functional Requirements:**

We describe the functional requirements by giving various use cases.

***Use cases related to system authorization***

**Use Case 1**: **Login User**

*Primary Actor*: Candidate, Admin, Voter

*Pre-Condition*: Good Internet connection

*Main Scenario*:

1. System prompts the user for login and password.

2. User gives the login and password.

3. System does authentication.

4. The user dashboard is displayed according to the category of user(admin/voter/candidate).

*Alternate Scenario*:

4(a). Authorization fails

4(a)1. Prompt the user that he typed the wrong password

4(a)2. Allow him to re-enter the password. Give him 3 chances.

***Use cases related to registration***

**Use Case 2: Verify candidate’s documents**

*Primary Actor*: Admin

*Pre-Condition*: Admin logged in.

*Main Scenario*:

1. Admin clicks a candidate name marked with the status “Pending Approval”.
2. Admin checks the documents of the candidate.
3. If the document is correct, he clicks Approve button. If it is faulty and admin wants to remove it, admin clicks on Reject button.
4. Admin clicks on “approve Candidate”.

*Alternate Scenario*:

4(a). All documents present do not have the status “Approved”

4(a)1. System asks the admin to delete the faulty document or wait for the candidate to upload the correct one.

4(b) One or more documents has not been marked as Approve/Reject.

4(b)1. System prompts the admin to mark it.

**Use Case 4: Register Candidate**

*Primary Actor*:Candidate

*Pre-Condition*:Good Internet connection

*Main Scenario*:

1. System asks candidate to enter basic details like Name, Age, Party, Constitution etc.

2. Candidate enters the details.

3. System asks candidate to upload the requisite documents.

4. Candidates browses to the directory containing the documents and uploads them.

5. Candidate clicks on Register button.

6. System notifies that the user has to wait for verification of the documents.

*Alternate Scenario*:

3(a) Candidate left a field blank

3(a)1. System displays error message and allows candidate to type the information

5(a) File limit exceeded

5(a)1. System displays an apology message and allows user to delete uploaded documents.

**Use Case 5: Register Voter**

*Primary Actor*:Voter

*Pre-Condition*:Good Internet connection.

*Main Scenario*:

1. System asks candidate to enter basic details like Name, Age, Voter Card ID, Email ID etc.

2. Voter enters the details.

3. Voter clicks on Register button.

4. System registers the voter.

*Alternate Scenario*:

5(a) Voter left a field blank

5(a)1. System displays error message and allows voter to type the information

***Use cases related to candidate information***

**Use Case 6**: **View Candidate’s data**

*Primary Actor*: Voter

*Pre-Condition*:Voter logged in.

*Main Scenario*:

1. Voter clicks on view candidates tab

2. System retrieves database and accesses requisite information.

3. System displays the documents and detailed information about the background of candidate.

*Alternate Scenario*:

2(a). No candidates in database.

2(a)1. System displays a blank table.

**Use Case 7: Upload Candidate Information or Milestones**

*Primary Actor*: Candidate

*Pre-Condition*:Candidate logged in.

*Main Scenario*:

1. Candidate clicks on add milestones tab.

2. System displays current milestones and gives space to add to them.

3. Candidate updates them.

*Alternate Scenario*:

*2(a)* No current milestones

2(a)1. System displays empty space to add milestones.

***Use cases related to election***

**Use Case 8: Create Election**

*Primary Actor*: Admin.

*Pre-Condition*: Admin logged in.

*Main Scenario*:

1. Admin clicks on “Create Election”.
2. System displays form for election creation.
3. Admin fills up the required details including start time and end time of election.
4. Admin clicks on “Create”
5. System displays a success message.

*Alternate Scenario:*

**Use Case 9: Vote for Candidate**

*Primary Actor*: Voter

*Pre-Condition*: Voter logged in.

*Main Scenario*:

1. User selects Voting option.
2. System displays the ongoing elections.
3. User selects the desired election.
4. System displays the different candidates where voter can view their details.
5. Voter clicks on the vote button beside the desired candidate.
6. System displays a success message and disables the voter from voting further in that election.

*Alternate Scenario:*

2(a) No ongoing election

2(a) 1. System displays an apology message or an empty table.

**Use Case 10: Calculate Result**

*Primary Actor*:Candidate, Admin, Voter

*Pre-Condition*:User(Candidate/ Admin/Voter) logged in.

*Main Scenario*:

1. User clicks on Elections
2. User clicks on “Results” beside the election.
3. System calculates the result and election statistics.
4. System displays candidate vs its number of votes.

*Alternate Scenario:*

3(a). No finished election

3(a)1. System displays empty table or apology message.

**Use Case 11: Cancel Election**

*Primary Actor*: Admin.

*Pre-Condition*: Admin logged in.

*Main Scenario*:

1. Admin clicks on Elections.
2. System displays the elections with their status “ongoing” or “paused” or “finished”.
3. Admin clicks on “Cancel Election” beside the appropriate election.
4. System asks a confirmation message.
5. Admin confirms and cancels election.

*Alternate Scenario:*

3(a) Cancel button is disabled as election is not ongoing.

3(a) 1.System allows user to cancel election if paused.

**Use Case 11: Finish Election**

*Primary Actor*: Admin.

*Pre-Condition*: Admin logged in.

*Main Scenario*:

1. Admin clicks on Elections.
2. System displays the elections with their status “ongoing” or “paused” or “finished”.
3. Admin clicks on “Finish Election” beside the appropriate election.
4. System asks a confirmation message.
5. Admin confirms and finishes election.
6. System displays success message

*Alternate Scenario:*

3(a) Election is not ongoing.

3(a) 1. System doesn’t allows user to finish election if paused.

**3.2 Performance Requirements:**

(a) The software should run on at least 500 MHz, 64 MB machine.

(b) The software is expected to serve a maximum of up to 500 voters at any point of time.

(c) OVT should be able to log in and feed the voter with new pages on request with a response time of the order of a few seconds.

**3.3 Design Constraints:**

1. *Security against power failure:*In order to prevent data loss in case of power failure, the result of votes that were polled till then have to be saved in the database, for the system to resume the counting process on reboot.

2.*Handling security breaches:*In case the EC detects any security breaches in the system, they should able to pause or cancel the election immediately while preserving the already polled votes.

3.*Security against crash:*The software should be capable of recovering from crashes and continuing the voting process.

4. *Authentication:*The system should provide basic security features like password authentication and encrypted transactions.

5. *Security:* Protection from malicious users should be maintained by limitingthe number of invalid log-in attempts.

**3.4Future Extensions:**

The functionality can be extended so as to accommodate all the functionalities of the elections online rather than just voting. All the processes conducted by the Election Commission, prior to the voting process, including registration of candidates in a particular election can be incorporated in the software.