



FUNCTIONAL SPECIFICATION

Project Code:	EVS-01
Project Name:	Electronic Voting System

FUNCTIONAL SPECIFICATION



EVS-01/FS-1.00

Revision History

Version (x.yy)	Date of Revision	Description of Change	Reason for Change	Affected Sections	Approved By
1.00	16-Aug-2011	Initial Draft			

Affected Groups

Development Engineering
Quality Assurance
XYZ Automations Ltd

List of Reference Documents

Name	Version No.
1. Request For Proposal	1.2
2.	
3.	
4.	

FUNCTIONAL SPECIFICATION



EVS-01/FS-1.00

Table of Contents

1. INTRODUCTION	4
2. SYSTEMS OVERVIEW	4
3. SUB-SYSTEM DETAILS	5
4. DATA ORGANIZATION	8
5. ASSUMPTIONS	10
6. EXPECTATIONS	10
7. ACCEPTANCE CRITERIA	10
8. TRACEABILITY TO REQUIREMENTS	10
9. ACRONYMS AND GLOSSARY	10

FUNCTIONAL SPECIFICATION



EVS-01/FS-1.00

1. Introduction

1.1 Background

XYZ Automations Ltd focuses on automating various systems that have been working manually since years

1.2 Purpose

XYZ Automations Ltd has recently planned to develop "Electronic Voting System" - a web application to automate the process of planning and managing the voting system, as well as the actual voting activity.

1.3 Scope

The scope of the Electronic Voting System (EVS) will be to provide the functionality as described in Functional Requirements below. The system will be developed on a Windows XP machine using J2EE, JSP/HTML, and JDBC.

2. Systems Overview

2.1 System Description

The Electronic Voting System should support the following users.

- Administrator (A)
- Electoral officer (E)
- User (voter) (U)

The common functional requirements are explained below.

2.1.1 Authentication

Any end-user should be authenticated using a unique login ID and password.

2.1.2 Authorization

The operations supported and allowed would be based on the user type. For example, Administrator has the rights to add or delete election details.

Search for ...

0 of 0

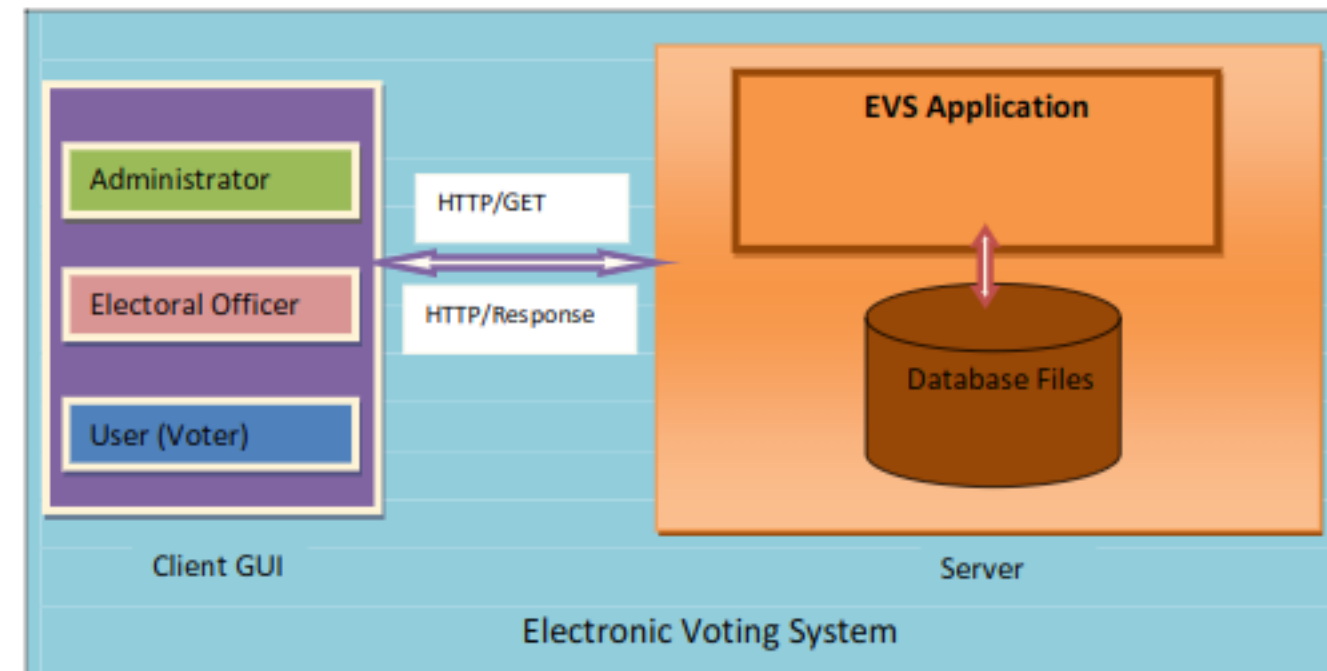


4

of 11



The functional flow of the messages across different application components is shown below.



2.2 Environment

The system will be developed on a Windows XP machine using J2EE, JSP/HTML, and JDBC.

- Intel hardware machine (PC P4-2.26 GHz, 512 MB RAM, 40 GB HDD)
- Server – Apache Tomcat 6 or higher
- Database – Oracle 9i or higher
- JRE
- Eclipse IDE

3. Sub-system Details

3.1 Administrator

The Administrative User should be able to do the following operations once he has logged in with his unique user id and password

- Add Election Details
- Add Party Details
- Add Candidate Details
- View Voter's request for voter_id
- View Election Details
- View Party Details.
- View Candidate Details
- View Election Results

FUNCTIONAL SPECIFICATION



EVS-01/FS-1.00

Add Election Details: As part of this operation, the admin should have the ability to add election details into the system and get confirmation on successful entry.

Election Details include:

- Name
- ElectionDate
- Voting Time (8.00 am to 5.00 pm)
- District
- Constituency

(Refer to section 4. Data Organization → Table 3: ElectionDetails)

Add Party Details: As part of this operation, the admin should have the ability to add Party details into the system and get confirmation on successful entry.

Party Details include:

- Party Name
- Party Leader
- Party Symbol

Add Candidate Details: As part of this operation, the admin should have the ability to add Candidate details into the system and get confirmation on successful entry.

Candidate Details include:

- Candidate Id
- Candidate Name
- DOB
- Election Name
- Party Name
- District
- Constituency
- Address
- Phone No
- Email Id

View Voter's request for voterId: As part of this operation, after admin logs in, he should be able to see all requests for voter ids'.

View Election Details: As part of this operation, the admin should be able to view election details.

View Party Details: As part of this operation, the admin should be able to view party details.

View Candidate Details: As part of this operation, the admin should be able to view candidate details.

View Election Results: As part of this operation, the admin should be able to view election results

Quit Session: As part of this operation, admin should have the ability to quit the session completely.

3.2 Electoral Officer

The Electoral Officer should be able to do the following operations once he has logged in with his unique UserID and password.

- View voter Id request
- Generate voter Id's

View voter Id request: As part of this operation, after electoral officer logins, he should be able to see all voters requests got from admin.

(Refer to section 4. Data Organization → Table 4: VoterIdRequestDetails for further details)

Generate Voter ID: As part of this operation, the Electoral Officer should have the ability to view the list of voters in his respective constituency with the following details:

- UserID
- First Name
- Last Name
- DOB
- Age
- Gender
- Password
- Address
- Phone No
- Constituency

and an option to either "Accept ☒ " or "Reject ☐ " for each user.

On **accept**, the voter-Id gets auto generated and stored in the respective voters table.

(Refer to section 4. Data Organization → Table 1: UserDetails for further details)

Quit Session: As part of this operation, the Electoral Officer should have the ability to quit the session completely, with all the sessions inactivated.

3.3 User (Voter)

The Voter should be able to do the following operations

- User Registration
- Request for voter ID
- View voter ID
- View Election Schedules
- Cast Vote
- View Election Results.

User Registration: As part of this operation, an unregistered user should be able to register into the system with the following details.

Registration details include (all mandatory):

- User ID
- First Name
- Last Name
- DOB
- Age
- Gender
- Password
- Address

- WIPKU**
Applying Thought

(Refer to section 4. Data Organization → Table 1: UserDetails for further details)

(Refer to section 4. Data Organization → Table 4: VoterIdRequestDetails)

View Election schedule: As part of this operation, the user (voter) should be able to view election schedules.

The user (voter) should have the ability to cast their vote for an election declared on a date, to one of the contesting parties in their constituency.

View Election Results: As part of this operation, the user (voter) should have the ability to view the election results, with the following details by picking an ElectionDate or ConstituencyName from the respective lists.

- NOTE: 1. The ElectionDate list should have max 10 (recent) election dates for which, counting has already being done.

- Quit Session: As part of this operation, the user (voter) should have the ability to quit the session completely.

Note: Details included here are indicative. Please add more details wherever required.

This section explains the data storage requirements of the Electronic Voting System application and **indicative** table (database) structure. The following sections explain few of the tables required for the application and the other tables will have to be designed accordingly.

4.1 Table: User Information

The user specific details such as name, address, authentication and authorization / privileges should be kept in one or more tables, as necessary and applicable.

Table 1: UserDetails (Voter)

Field Name	Data Type	Description
<i>UserID</i>	<i>VARCHAR(9)</i>	It will be a numeric value sequentially generated by the system and need not be passed from the client. It will be added by the system whenever a new user gets registered.
<i>VoterID</i>	<i>NUMBER(5)</i>	Auto generated and added to the system as soon as an Electoral Officer accepts the credentials entered by the user/or null
<i>Password</i>	<i>VARCHAR(9)</i>	Encrypted Password
<i>First Name</i>	<i>VARCHAR(15)</i>	First Name of user
<i>Last Name</i>	<i>VARCHAR(15)</i>	Last Name of user
<i>DOB</i>	<i>DATE</i>	DOB of user
<i>Age</i>	<i>NUMBER</i>	Age of user
<i>Gender</i>	<i>VARCHAR(7)</i>	Gender of user
<i>Address</i>	<i>VARCHAR(30)</i>	Address of user
<i>PhoneNo</i>	<i>VARCHAR(10)</i>	Phone No of user
<i>Constituency</i>	<i>VARCHAR(15)</i>	Constituency the user belongs to
<i>District</i>	<i>VARCHAR(15)</i>	District the user belongs to

4.2 Table: Authentication Information

The table contains Authentication Information for Administrator and Electoral Officer

Table 2: LoginCredentials

Field Name	Data Type	Description
<i>UserType</i>	<i>VARCHAR(1)</i>	A-Administrator E-Electoral Officer
<i>UserID</i>	<i>VARCHAR(9)</i>	User Identification
<i>Password</i>	<i>VARCHAR(9)</i>	Password

4.3 Table: Election Information

This table contains Election specific information, kept in one or more tables, as necessary and applicable.

Table 3: ElectionDetails

Field Name	Data Type	Description
<i>ElectionName</i>	<i>VARCHAR(50)</i>	The unique election should be given from the end user
<i>ElectionDate</i>	<i>DATE</i>	Date of Election
<i>VotingTime</i>	<i>VARCHAR(50)</i>	Start and End time of voting .Ex: (8.00 am to 5.00 pm)

Table 4: VoterIdRequestDetails

Field Name	Data Type	Description
<i>UserID</i>	<i>VARCHAR(9)</i>	Unique userID of the registered user in section 4.1 Table 1

5. Assumptions

- User Interface: The type of client interface (front-end) to be supported is – GUI based
- Each user will hold only one voter-ID and cast vote only once

6. Expectations

- The application should be designed using client server architecture
- The server should be a concurrent server servicing multiple clients
- Database can be implemented using Oracle 9i or above
- To begin with, the application should support at least 1 admin and 5 electoral officers)
- Compilation and Build should be done using Eclipse IDE
- Source-code and all documents must be maintained (checked-in) in configuration management system (subversion)
- Wipro's coding standards (for Java) should be followed
- Deliverables should include use-case diagrams, design document, compiled and tested source code, test-plans, test-cases documents, test-results and release note

7. Acceptance Criteria

All P1 requirements have to be mandatorily implemented

8. Traceability to Requirements

Document Reference ID & Description: (Doc ID from which this document is derived)		
Sl. No.	Reference document: RS Requirement/Feature (Section ID/Name)	Current document: FS Location (Section ID/Name)
1.	Requirements as mentioned in Section 2.1	Section 3.1, 4.3
2.	Requirements as mentioned in Section 2.2	Section 3.2, 4.3
3.	Requirements as mentioned in Section 2.3	Section 3.3, 4.1, 4.3
4.	Requirements as mentioned in Section 2.4	Section 2.1.1, 2.1.2, 4.2
5.	Requirements as mentioned in Section 2.5	Section 3.1, 3.2, 3.3

9. Acronyms and Glossary

Abbreviation	Remark
EVS	Electronic Voting System
RS	Requirement Specification
FS	Functional Specification