

SOFTWARE REQUIREMENTS SPECIFICATION

FOR

GOVolunteer

Table of Contents

1. Introduction.....	3
1.1 Purpose	3
1.2 Scope	3
1.3 Definitions, Acronyms, and Abbreviations	3
1.4 References	4
1.5 Overview	4
2. The Overall Description.....	5
2.1 Product Perspective	5
2.2 Product Functions	5
2.3 User Characteristics	6
2.4 Constraints	6
2.5 Assumptions and Dependencies	7
3. External interface Requirements.....	7
3.1 User Interface Requirement	7
3.2 Software Interface Requirement	8
4. System Features.....	8
4.1 User Registration	8
4.2 Personalized Dashboard	9
4.3 Quick Access	9
4.4 Create Job Opportunity and Review Applicants	9
4.5 Search and apply for job	10
5. Other Non-functional Requirements.....	10
5.1 Performance Requirements	10
5.2 Safety Requirements	10
5.3 Security Requirements	11
5.4 Software Quality Attributes	11
5.5 Business Rules	12

1. Introduction

GOVolunteer is a user-friendly web platform simplifying the volunteer selection process. It connects organizations with a diverse pool of volunteers for various part-time opportunities. This Web application includes an algorithm which matches the unique skills of the volunteer to the organizations in need. GOVolunteer is a catalyst for a connected and engaged community, encouraging contributions to events or everyday acts of kindness.

1.1 Purpose

This SRS defines Performance and Software System Attributes requirements of GOVolunteer. This document is intended for the following group of people: -

- ✓ Developers for the purpose of maintenance and new releases of the software.
- ✓ Volunteer Management
- ✓ Documentation writers.
- ✓ Testers.

1.2 Scope

The scope of the GOVolunteer project encompasses the development and implementation of a web application that facilitates the identification, coordination, and engagement of volunteers in various part-time activities. The platform covers a wide range of opportunities, ranging from supporting important events like elections to everyday acts of kindness such as babysitting or assisting the elderly. The scope also includes the utilization of secure verification processes, efficient coordination tools, and real-time updates to ensure a reliable and trustworthy pool of volunteers.

The software takes as input the login Id and the personal details and interests of the user for login purposes. The outputs then comprise of an interactive display that lets the user select the desirable opportunities that who wants to apply.

The software is expected to complete in duration of three months and the estimated cost is Rs 2 Crore.

1.3 Definitions, Acronyms, and Abbreviations.

GOVolunteer	The name of the web application.
Frontend	The user interface of the application developed using the React framework.
Backend	The server-side architecture developed using Node.js.
Database	MySQL2, the database management system used for data storage.
SRS	Software Requirements Specification.

1.4 References

- i. IEEE. Software Requirements Specification Std. 830-1993.

1.5 Overview

Section 1.0 discusses the purpose and scope of the software.

Section 2.0 describes the overall functionalities and constraints of the software and user characteristics.

Section 3.0 details all the requirements needed to design the software.

2. The Overall Description

2.1 Product Perspective

- ✓ The primary function of this software is to enable users to remotely access and manage their volunteer activities without the need for direct assistance from an intermediary.
- ✓ Key software components include secure verification systems, coordination tools, real-time updates, and an intuitive user interface. These elements collectively contribute to the seamless operation of the platform.
- ✓ GOVolunteer establishes communication with its central server to ensure real-time updates and a reliable pool of volunteers, fostering a connected community.
- ✓ The application is optimized for efficiency, with a memory capacity of ___MB to handle various data processing requirements.

2.2 Product Functions

✓ **Login/Register:**

- User Registration:-Allow users to create a new account by providing necessary information. Validate and securely store user details.
- User Login:- Provide a secure login mechanism with username/email and password. Implement authentication to verify user identity.
- Forgot Password:- Allow users to reset their password through a secure email verification process.

✓ **Dashboard:**

- Personalized Overview:-Display a personalized dashboard for each user after login. Showcase relevant information, such as upcoming opportunities or recent activity.
- Quick Access:-Provide quick links to essential features like profile, opportunity search, and job postings.
- Notifications:-Notify users about new opportunities, application status, and system updates.

✓ **Profile:**

- User Profile Management:-Enable users to update and manage their personal information, skills, and preferences.
- Upload Profile Picture:-Allow users to upload and manage their profile pictures.
- Activity History:-Display a history of user activity, such as applied opportunities and contributions.

✓ **Opportunity Search:**

- Search Filters:-Provide advanced search filters to help users find opportunities based on location, category, and date.
- Detailed Opportunity Information:-Display detailed information about each opportunity, including requirements and contact details.
- Application Process:-Allow users to easily apply for opportunities directly through the platform.

✓ **Job Posting:**

- Create Job Opportunity:-Allow registered users to post new volunteer opportunities or jobs.
- Edit/Delete Postings:-Enable users to edit or delete their posted opportunities.
- Application Review:-Provide a mechanism for organizations to review and manage applications received for their opportunities.

2.3 User Characteristics

The software is designed to serve three primary user roles, each with specific functions and responsibilities within the system:-

- ✓ **Job Seekers/Volunteers:** These users actively search for employment or volunteer opportunities, represent a varied group with different levels of experience and skills. Their system interaction involves accessing the opportunity search feature, applying for positions, and managing personal profiles to receive relevant notifications about new opportunities and application updates.
- ✓ **Recruiter:** These individuals or organizations, have diverse backgrounds and aims, posting job opportunities or volunteer positions. They have various options like creating, editing, or deleting posted opportunities and managing applications received.
- ✓ **Admins:** These users are responsible for system oversight and maintenance. They can manage user accounts, monitor system integrity, configure settings, address technical issues, and access maintenance and troubleshooting options.

2.4 Constraints

The major constraints of the project are as follows:

- ✓ How many can the server handle at a time?
- ✓ How many unsuccessful logins
- ✓ How many opportunities can be posted by a user at a time?
- ✓ How long can the job stay inactive?
- ✓ how many jobs can the volunteer apply for at a time?
- ✓ The software requires a minimum memory of
- ✓ The database used should be

2.5 Assumptions and Dependencies

The requirements stated in the SRS could be affected by the following factors:

- ✓ Dependency on external systems: Integration with external databases or verification services may require updates to maintain functionality and data integrity.
- ✓ Legal and regulatory changes: Evolving laws regarding volunteer engagement and data privacy may necessitate updates for compliance and risk mitigation.
- ✓ Internet connectivity and server uptime: Reliance on stable internet and server infrastructure is crucial for uninterrupted access to the platform.
- ✓ Scalability considerations: The system should be able to handle increased traffic and data demands as the user base grows.
- ✓ Resource availability: Adequate funding and resources are essential for ongoing maintenance and updates.
- ✓ Stakeholder collaboration: Effective communication between developers, organizations, and users is vital for gathering feedback and addressing issues.
- ✓ Scope expansion: The project may need to accommodate new features and integrations over time to meet evolving user needs and market trends.

3.External Interface Requirements

3.1 User Interface Requirements

The interface provided to the user should be a very user-friendly one and it should provide an optional interactive help for each of the service listed. **The** interface provided is a menu driven one and the following screens will be provided

1. Login Screen: A login page is provided with the username or mobile number with the password. After the user has successfully entered the username and password, The website navigates to the dashboard section
2. Forgot Password Screen: In case of forgot password, The user will navigate to the forgot password where they can change the password after confirmation of their email.
3. Signup Screen: When the user visits the websites for the first time and want to register themselves as the volunteer/user they are navigated to the signup page. Once the user had successfully filled those details, they can navigate to the login page.
4. Profile Picture Upload Screen: Users can upload and manage their profile pictures.
5. Authentication page: For the part of the authentication process in the signup module, the user must enter the OTP that was sent to their phone number. After the validation of the OTP, the user account is successfully created
6. Job Posting Screen: Registered users can post new volunteer opportunities or jobs, edit/delete their postings, and review/manage applications received for their opportunities.
7. Opportunity Search Screen: Users can search for volunteer opportunities using advanced filters such as location, category, and date.

8. **Profile Management Screen:** Users can update and manage their personal information, skills, and preferences.

The following reports will be generated after each session dealt with in the Web App:-

- **User Activity Report:** The login time, logout time and username are recorded for each session. This information is stored in the platform's database for auditing and tracking purposes.
- **Account Changes Report:** The change includes updates to personal information, skills, preferences, etc. All changes are logged in the database to maintain a history of account modifications for reference and security purposes.

3.2 Software Interface Requirements

To perform various functions, this software needs to interact with various other software's. So, there are certain software interface requirements that need to be fulfilled which are listed as follows: -

1. **Frontend Interface:** Developed using React framework. Responsive design principles ensure compatibility across various devices.
2. **Backend Interface:** Utilizes Firebase as the backend platform. Handles requests from the frontend, processes data, and interacts with Firebase services. Utilizes Firebase's APIs and SDKs for authentication, database interactions, and cloud functions.
3. **Database Interface:** Uses Firebase Realtime Database or Firestore for data storage. Interacts with Firebase's database services to store and retrieve user data, volunteer opportunities, job postings, and other relevant information.
4. **Authentication Interface:** Interfaces with Firebase Authentication to verify user identity during login and signup processes.

4. System Features

4.1 User Registration

4.1.1 Description and Priority

User Registration facilitates the creation of new accounts within the system by collecting necessary user information. This feature is of utmost priority as it serves as the fundamental step for users to access the platform's functionalities.

4.1.2 Action/Result

Upon accessing the registration page, users provide essential details such as name, email, and password. After submission, the system validates the information for accuracy and completeness. Upon successful validation, a new user account is created, granting access to the platform.

4.1.3 Functional Requirements

- ✓ Registration form with fields for user details.
- ✓ Validation checks to ensure the accuracy and completeness of provided information.

- ✓ Secure storage of user credentials.
- ✓ Confirmation mechanism to verify the creation of a new user account.

4.2 Personalized Dashboard

4.2.1 Description and Priority

Personalized Dashboard offers users a customized view of relevant information and activities based on their preferences and interactions with the platform. This feature holds a high priority as it enhances user engagement and satisfaction.

4.2.2 Action/Result

Upon logging in, users are directed to their personalized dashboard, which showcases tailored content such as upcoming opportunities, recent activity, and recommended jobs. The dashboard dynamically updates based on user interactions and preferences.

4.2.3 Functional Requirements

- ✓ Dashboard interface with customizable widgets or sections.
- ✓ Data retrieval and filtering based on user preferences and activity history.
- ✓ Real-time updates for new opportunities or changes in application status.
- ✓ Customization options for users to adjust their dashboard settings.

4.3 Quick Access (Profile Management)

4.3.1 Description and Priority

Quick Access (Profile Management) provides users with swift navigation to essential profile management features. This feature holds a high priority as it streamlines user interaction with their profile settings.

4.3.2 Action/Result

Users can access profile management functions such as editing personal information, updating skills, and managing preferences through quick links available on the platform. This enables efficient navigation and management of user profiles.

4.3.3 Functional Requirements

- ✓ Navigation menu or toolbar with quick links to profile management features.
- ✓ Accessible interface for seamless interaction with profile settings.
- ✓ Integration with user preferences to customize profile-related options.
- ✓ Mobile-responsive design for accessibility across devices.

4.4 Create Job Opportunity and Review Applicants

4.4.1 Description and Priority

Create Job Opportunity and Review Applicants enables registered users to post new job opportunities and manage applicant submissions efficiently. This feature holds high priority as it facilitates recruitment processes within the platform.

4.4.2 Action/Result

Registered users can create job opportunities by providing details such as job title, description, requirements, and application instructions. Additionally, they can review applicant submissions, evaluate qualifications, and make informed decisions regarding recruitment.

4.4.3 Functional Requirements

- ✓ Job posting form with fields for opportunity details and application instructions.
- ✓ Mechanism to review and manage applicant submissions associated with posted job opportunities.
- ✓ Communication tools for contacting applicants or scheduling interviews.
- ✓ Status update mechanisms for tracking the progress of applicant reviews.

4.5 Search and Apply for Job

4.5.1 Description and Priority

Search and Apply for Job allows users to explore available job opportunities based on their preferences and apply for positions of interest. This feature is of high priority as it empowers users in their job search endeavours.

4.5.2 Action/Result

Users can utilize search filters to refine job listings based on criteria such as location, category, and date. Upon finding suitable opportunities, users can initiate the application process directly through the platform, submitting required documents and information.

4.5.3 Functional Requirements

- ✓ Search interface with advanced filtering options for job listings.
- ✓ Integration with job database for real-time retrieval of job opportunities.
- ✓ Application form/interface with fields for user details and submission of required documents.

Confirmation mechanism to acknowledge successful application submissions.

5. Other Nonfunctional Requirements

5.1 Performance Requirements

- The web application should load within 3 seconds on average, ensuring a smooth user experience with the help of lazy loading.
- Response times for user actions such as login, searching for opportunities, and updating profiles should be less than 1 second.
- Database queries should be optimized to minimize latency and ensure efficient data retrieval.

5.2 Safety Requirements:

- User data should be encrypted both in transit and at rest to ensure confidentiality and integrity.
- The system should have measures in place to prevent unauthorized access to user accounts and sensitive information.
- Error handling mechanisms should be implemented to gracefully handle unexpected errors and prevent system crashes.
- Users information is properly sanitized before storing them in the database.

5.3 Security Requirements:

- The web application should enforce strong password policies, including minimum length and complexity requirements.
- User authentication and authorization should be implemented securely to prevent unauthorized access to system resources.
- The system should protect against common security threats such as cross-site scripting (XSS), SQL injection, and cross-site request forgery (CSRF).
- Access to sensitive data and administrative functionalities should be restricted to authorized personnel only, using role-based access control (RBAC) mechanisms.

5.4 Software Quality Attributes:

5.4.1 Reliability:

- The software should consistently perform its intended functions accurately and predictably under specified conditions, without failure or errors. Users should trust its stability and consistency.

5.4.2 Performance:

- The software should execute tasks efficiently and respond promptly to user inputs, ensuring optimal speed, throughput, and resource utilization. It should meet performance requirements under varying workloads.

5.4.3 Scalability:

- The software should be able to handle increasing workloads or data volumes without sacrificing performance or functionality. It should scale effectively to accommodate growing user bases or system demands.

5.4.4 Maintainability:

- The software should be designed and implemented in a way that facilitates easy maintenance, updates, and modifications. It should have clear, well-structured code, documentation, and support for debugging and troubleshooting.

5.4.5 Usability:

- The software should be intuitive and user-friendly, requiring minimal training or assistance for users to understand and operate it effectively. It should have clear interfaces, informative feedback, and logical workflows.

5.4.6 Compatibility:

- The software should be compatible with various operating systems, platforms, devices, and software environments, ensuring seamless integration and interoperability with existing systems and technologies.

5.4.7 Flexibility:

- The software should be adaptable to changing requirements, environments, and user needs. It should support customization, configuration, and extensibility to accommodate diverse use cases and preferences.

5.4.8 Testability:

- The software should be designed and implemented in a way that facilitates thorough testing and validation of its functionality, performance, and quality. It should support automated testing and provide clear feedback to testers.

5.5 Business Rules

User Registration and Authentication:

- ✓ Users must register with valid personal information to create an account.
- ✓ Each user account must have a unique username and email address.
- ✓ Passwords must meet specified complexity requirements, including length and character types.
- ✓ Users must authenticate their identity via email verification or other secure mechanisms during registration and login.

Opportunity Posting and Management:

- ✓ Organizations or recruiters must provide accurate information when posting volunteer opportunities or part-time jobs.
- ✓ Opportunities must include details such as location, category, requirements, and contact information.
- ✓ Recruiters are responsible for reviewing and managing applications received for their posted opportunities.
- ✓ Recruiters have the authority to accept or reject volunteer applications based on their discretion and organizational needs.

Data Security and Privacy:

- ✓ The app must comply with data protection regulations and ensure the security and privacy of user information.
- ✓ User data, including personal details and application history, should be stored securely and accessible only to authorized personnel.
- ✓ User consent must be obtained for any data sharing or processing activities beyond the app's primary functionality.