Name: Varlene

Surname: Sibanda

Student Number: ST10276007

Module: SOEN6222 – software requirements specification

Introduction

The Helping Hands Software Solution is a comprehensive web-based platform designed to streamline the operations of Helping Hands, a non-profit organization dedicated to providing support, resources, and essential services to individuals and families in need. The organization offers various programs, including food distribution, shelter provision, clothing donations, and educational resources. However, reliance on manual and paper-based processes has created inefficiencies, inaccuracies, and challenges in scaling operations and tracking the impact of their initiatives.

The purpose of this software is to automate and digitize these processes, enabling the organization to better manage its resources, monitor program success, and communicate effectively with volunteers and beneficiaries. This project's scope includes developing a user-friendly website that supports program management, resource distribution tracking, volunteer coordination, and reporting.

Agile development methodology will ensure efficient software development that aligns with Helping Hands’ evolving needs. By breaking down the project into iterative sprints, we can deliver key features incrementally, focusing on modules such as Volunteer Management, Resource Allocation, and Program Reporting. Each sprint is validated through continuous integration and testing to maintain high quality while reducing risks. This approach not only optimizes current processes but also supports Helping Hands in scaling its operations to better serve the community. (prach, 2024)

Functional Requirements for Helping Hands Software Solution (janar, 2024)

1. User Registration and Authentication:

Registration Process: Users can register as volunteers, beneficiaries, or administrators. The system requires unique email addresses, secure passwords, and two-factor authentication (optional).

Role-Based Access Control (RBAC):

* + Volunteers: Access to event sign-up, task notifications, and hours logged.
  + Beneficiaries: Access to program schedules, resource requests, and feedback submission.
  + Administrators: Full access to all modules, including user management, resource allocation, and reporting.

Password Recovery: Users can reset forgotten passwords using secure email links. (nuclino, 2024)

1. Volunteer Management System:
   * + Volunteer Registration: A form collects contact details, availability, skills, and preferences.
     + Opportunity Listings: Administrators post opportunities with descriptions, dates, and required skills. Volunteers can browse and sign up.
     + Task Assignment: Admins assign tasks based on volunteer availability and skills, with task reminders sent via email/SMS.
     + Hours Tracking: Volunteers log hours worked, which admins verify. Reports summarize volunteer contributions over time.
2. Resource Distribution Tracking:
   * + Distribution Records: The system tracks distributed resources by item type, quantity, recipient details, and date.
     + Beneficiary Allocation: Admins allocate resources to beneficiaries, maintaining records for audits.
     + Historical Data: Generate reports detailing distributions over specific timeframes or for specific programs.
3. Inventory Management:

* The system must have a module for managing inventory, including food, clothing, and other essential resources
* Item Database: Store details for each inventory item, including type, quantity, source, and expiration.
* Stock Updates: Admins can update inventory through batch uploads or manual entries.
* Alerts: Low-stock and expiration alerts help prevent shortages or waste.

1. Online Donations Module:
   * + Donation Methods: Supports one-time and recurring donations via secure gateways (PayPal, Stripe).
     + Customizable Amounts: Donors choose predefined amounts or enter custom contributions.
     + Receipts: Automated receipts with tax-deductible information sent via email.
     + Donation History: Donors view and download their contributions for personal records or tax filing.
2. Mobile-Responsive Design:
   * + Responsive Layout: The platform dynamically adjusts to different screen sizes for optimal usability.
     + Mobile-Specific Features: Optimized forms, navigation menus, and faster loading times for mobile users.
3. Secure Data Storage and Access Control:

* All data (user information, program details, resource tracking, etc.) must be stored securely in compliance with data protection laws.
* Access Control: Sensitive information, such as beneficiary details and donation records, is accessible only to authorized personnel.
* Audit Logs: Maintain logs of all system activities to monitor data access and changes.

1. Event Management System:
   * + Event Creation: Admins set up events, specifying dates, locations, required resources, and tasks.
     + Event Sign-Up: Volunteers and beneficiaries sign up, with confirmation emails sent automatically.
     + Resource Allocation: Track resources needed for each event, ensuring availability.
     + Event Feedback: Post-event surveys allow participants to provide insights for improvement.
2. Beneficiary Feedback System:

* The software must provide a mechanism for beneficiaries to submit feedback on the programs they participate in.
* This feedback should be stored and accessible to administrators for review and program improvement purposes.
* The feedback form must be easy to use and include both rating scales and open-ended responses.

1. Multi-Language Support:

* The system must support multiple languages to cater to a diverse user base.
* The user interface, notifications, and relevant documentation should be available in at least English and other commonly spoken languages in the region.

External Interface: Payment Gateway Integration (e.g., PayPal or Stripe) (jelvix, 2024)

The payment gateway is a critical external system that the Helping Hands software must integrate with to facilitate secure online donations from individuals and organizations. This system will enable donors to make contributions directly through the website using a variety of payment methods, including credit/debit cards, bank transfers, and digital wallets (e.g., Apple Pay, Google Pay). Below is a detailed breakdown of how this integration will work and the required functionality (stripe, 2023):

* Integration Scope and Purpose: Facilitates secure online donations, automates financial tracking, and ensures transparency through receipts and reports.
* Transaction Flow: initiate payments, enter details, and the system securely transfers data to the gateway. It handles payment confirmation or failure and sends transaction receipts.

* Security and Compliance: Ensures PCI DSS compliance, handles payment data via tokenization, and incorporates fraud detection mechanisms to protect sensitive information.
* Donation Management Features: Supports one-time and recurring donations, automates receipts, and allows administrators to view and export donation history for audits and financial tracking.
* API Integration: Provides APIs for transaction processing, refunds, and managing recurring payments. Webhooks enable real-time notifications of transaction events.
* User Experience: Streamlines the donation process with user-friendly forms and mobile responsiveness, ensuring donors can easily contribute from any device.

Non-Functional Requirements for Helping Hands Software Solution (Ashm, 2024)

1. Performance:  
   The Helping Hands system must support up to 1,000 concurrent users during peak times, such as major fundraising events or disaster relief efforts, without slowing down. Pages like the donation portal, event signup, and resource management must load within 2 seconds. The system should be optimized for efficient data retrieval and processing, particularly for resource distribution and volunteer management, ensuring no delays in critical operations.
2. Scalability:  
   As the Helping Hands organization grows, the system must scale easily to support additional programs, users, and resource management needs. This could include new features such as adding more categories of donations, more locations for resource distribution, or increasing the number of beneficiaries. The infrastructure should support adding more servers (horizontal scaling) or upgrading existing hardware (vertical scaling) to handle increased traffic, especially during global crises where donations or volunteer signups may spike rapidly.
3. Security:  
   Given the sensitive nature of donor information, volunteer profiles, and beneficiary data, the system must prioritize security. All interactions between users and the platform, especially for donations, must use SSL/TLS encryption. Role-based access control (RBAC) will ensure that only authorized users (e.g., administrators or volunteers) can access sensitive data. Additionally, the system must comply with GDPR to safeguard personal information, ensuring beneficiaries and donors can request data deletion or access at any time.
4. Maintainability:  
   The Helping Hands platform must be designed for easy updates, allowing developers to introduce new features, make bug fixes, or apply security patches with minimal disruption to users. A clean, well-documented codebase will facilitate quick troubleshooting and feature expansion. Regular system maintenance should not require significant downtime, and ideally, updates should be deployed during off-peak hours or with zero-downtime strategies, ensuring the platform’s continuous operation.
5. Compatibility:  
   The system must be compatible across multiple platforms and browsers, including Chrome, Firefox, Safari, and Edge, ensuring that users can access the site regardless of their device. The platform should also be fully responsive, allowing beneficiaries, volunteers, and donors to use the website seamlessly on both desktop and mobile devices (Android and iOS). This is especially important for users in remote or disadvantaged areas who may rely on mobile devices to access resources or sign up for programs.

Conclusion

The Helping Hands Software Solution represents a crucial step in modernizing the organization's operations and improving efficiency. By transitioning from manual processes to a digital, web-based platform, Helping Hands will be able to manage its resources more effectively, track program outcomes, and streamline communication with volunteers, beneficiaries, and donors. This project, developed using an Agile methodology, ensures that the system evolves to meet the organization's needs, with continuous feedback and iterative improvements leading to a flexible and adaptable solution.

The software solution will provide vital functionality, such as resource distribution tracking, volunteer management, secure online donations, event coordination, and multi-language support. Its design prioritizes performance, scalability, security, and user experience, ensuring it can accommodate the growing needs of Helping Hands while remaining secure and reliable.

Through this comprehensive system, Helping Hands will be better equipped to fulfill its mission of providing aid and support to individuals and families in need, ultimately expanding its reach and impact in the community. This platform will not only streamline internal operations but also foster greater transparency, accountability, and engagement among volunteers and donors, making it a powerful tool for the organization's continued success. (relevant.software, 2024)

Appendix: Updates Addressed

The updates focused on refining clarity and detail while addressing scalability, security, and maintainability. The functional requirements now include beneficiary feedback, multi-language support, and event management to better align with Helping Hands' objectives. The non-functional requirements were enhanced to emphasize scalability and performance during critical periods. Key updates include:

1. Expanded Volunteer Management System to cover task assignment and hours tracking.
2. Enhanced Resource Distribution and Inventory Management with low-stock alerts and historical data.
3. Added detailed Beneficiary Feedback and Multi-Language Support for inclusivity.
4. Clarified security mechanisms like audit logs, GDPR compliance, and encrypted backups.
5. Improved Online Donations by specifying recurring options, receipt automation, and tax details.

These updates align with modern standards for non-profit platforms, ensuring Helping Hands has a scalable, secure, and efficient solution tailored to its mission.

# References

Ashm. (2024, july 03). *geeksforgeeks*. Retrieved from geeksforgeeks: https://www.geeksforgeeks.org/non-functional-requirements-in-software-engineering/

janar. (2024, february 15). *geeksforgeeks*. Retrieved from geeksforgeeks: https://www.geeksforgeeks.org/what-are-functional-requirements-in-system-design-examples-definition/

jelvix. (2024, september). *jelvix*. Retrieved from jelvix: https://jelvix.com/blog/software-requirements-specification

nuclino. (2024, september 27). *nuclino*. Retrieved from nuclino: https://www.nuclino.com/articles/functional-requirements

prach. (2024, september 23). *geeksforgeeks*. Retrieved from geeksforgeeks: https://www.geeksforgeeks.org/what-is-agile-methodology/

relevant.software. (2024, June 17). *relevant.software*. Retrieved from relevant.software: https://relevant.software/blog/software-requirements-specification-srs-document/

stripe. (2023, august 7). *stripe*. Retrieved from stripe: https://stripe.com/resources/more/how-to-integrate-a-payment-gateway-into-a-website#:~:text=Integrating%20a%20payment%20gateway%20into%20your