GCISLFullStackApp

Requirements and Specifications

WSU Granger Cobb Institute for Senior Living (GCISL)

Justin Keanini,

Teni Olugboyega,

Naomi Dion-Gokan

[09.30.2024]

Table of Contents

I. Introduction Justin

The GCISL Full Stack App is being developed for the WSU Granger Cobb Institute for Senior Living (GCISL), to streamline the management and data integration processes within senior living facilities. The application aims to support functionalities such as user management, resident data tracking, and stakeholder communication, providing a user-friendly interface to simplify operations.

The primary objectives of this project are to build a secure, scalable, and efficient full-stack application that addresses the needs of GCISL's staff and stakeholders. The system will enable data-driven decisions, improve overall facility management, and enhance communication within the organization.

II. System Requirements Specification

II.1. Use Cases Teni

Use Case 1: Managing User Accounts

Actors: Administrator

Description: An administrator manages user accounts by creating, updating, and deleting profiles. This ensures that only authorized individuals can access the system, and their information is always up to date.

Preconditions: The user is logged in as an administrator and has access to the User Management Module.

Main Flow:

The administrator selects the “Manage Users” option from the admin dashboard.

The administrator selects “Add New User” and inputs required details such as name, contact information, and role.

The administrator can also select an existing user, update their information, or delete the user from the system.

The system saves the changes and updates the user database.

Postconditions: The system updates the user list, reflecting the newly added, updated, or deleted profiles.

Exceptions: If the user does not have the required admin access, the system denies access to user management functionality.

Related Requirements:

User Management Module, Requirement 1: User profile management (creation, updating, deletion).

User Management Module, Requirement 2: Role-based access control.

Use Case 2: Generating Resident Health and Activity Reports

Actors: Administrator, Staff

Description: An administrator or staff member generates detailed reports of resident health and activity based on data collected over time. This helps in tracking trends and making informed decisions about care.

Preconditions: The administrator or staff has access to the Data Analysis and Reporting Module.

Main Flow:

The user selects the “Generate Report” option from the reporting dashboard.

The user selects a specific resident or group of residents, and specifies the date range and type of data (health or activity).

The system compiles the data and generates the report.

The user can view the report on-screen or choose to export it as a CSV or PDF file.

Postconditions: The report is displayed and optionally exported in the selected format.

Exceptions: If no data is available for the selected parameters, the system displays an error message.

Related Requirements:

Data Analysis and Reporting Module, Requirement 1: Generate resident health and activity reports.

Data Analysis and Reporting Module, Requirement 2: Export reports in CSV or PDF.

Use Case 3: In-App Messaging for Communication

Actors: Staff, Volunteers, Residents' Families

Description: Staff members communicate with external stakeholders such as volunteers or residents' families through the in-app messaging system, improving coordination without relying on external platforms.

Preconditions: The user has access to the Communication Module and appropriate permissions to send and receive messages.

Main Flow:

The user opens the messaging feature within the system.

The user selects a recipient or group (staff, volunteers, or residents' families).

The user types a message and sends it through the system.

The recipient receives and responds to the message within the system.

Postconditions: Messages are sent and received, enabling real-time communication between stakeholders.

Exceptions: If the recipient is not available, the system notifies the user that the message was not delivered.

Related Requirements:

Communication Module, Requirement 1: In-app messaging.

Use Case 4: Volunteer Registration

Actors: Volunteers, Administrator

Description: Volunteers input their personal information through a form on the website, enabling GCISL to maintain an updated volunteer database.

Preconditions: The volunteer has access to the Volunteer Management Module.

Main Flow:

The volunteer accesses the registration form from the system's website.

The volunteer enters their name, contact details, availability, and other requested information.

The system validates the input and saves the volunteer's information to the database.

Administrators can access the information through the volunteer management dashboard.

Postconditions: The volunteer's information is stored in the system and accessible to administrators.

Exceptions: If the form is incomplete or contains errors, the system prompts the user to correct the input.

Related Requirements:

Volunteer Management Module, Requirement 1: Input volunteer information.

Use Case 5: Exporting Volunteer Data

Actors: Administrator

Description: The administrator exports the volunteer database into an Excel file to analyze the data or share it with external parties.

Preconditions: The administrator has access to the Volunteer Management Module.

Main Flow:

The administrator selects “Export Volunteer Data” from the admin dashboard.

The system generates an Excel file with the details of all registered volunteers.

The administrator downloads the file for further use.

Postconditions: The Excel file is downloaded, containing the latest volunteer data.

Exceptions: If there is a system error during export, the system displays an error message and logs the issue.

Related Requirements:

Volunteer Management Module, Requirement 2: Export volunteer information in Excel format.

II.2. Functional Requirements Justin

1. User Management Module

Requirement 1: The system must allow for the creation, updating, and deletion of user profiles.

Description	The system will provide an interface for administrators to manage user accounts. This includes adding new users, editing their information (such as name, contact details, and role), and deleting users when they are no longer part of the system. This functionality is essential to ensure that only authorized users have access to the system and that user information remains accurate.
Source	GCISL
Priority	Priority Level 0: Essential and required functionality

Requirement 2: The system must support different user roles with corresponding access levels.

Description	The system will differentiate between user roles, such as administrators, staff, and volunteers. Each role will have specific permissions regarding what they can view and modify in the system. For example, administrators will have full access, while volunteers may only be able to update their own information. This ensures secure access control based on user responsibility.
Source	GCISL
Priority	Priority Level 0: Essential and required functionality

2. Data Analysis and Reporting Module

Requirement 1: The system must generate reports based on resident date, including health and activity information.

Description	The system will collect and organize resident data (e.g., health reports, activity logs) and allow administrators to generate detailed reports. These reports will help staff monitor resident health trends and daily activities, enabling better decision-making and care planning.
Source	GCISL
Priority	Priority Level 1: Desirable functionality

Requirement 2: The system must allow exporting of reports in CSV and PDF formats.

Description	The system will include a feature for exporting generated reports into widely used file formats, such as CSV and PDF. This will allow administrators to easily share data with external parties, such as healthcare providers, or for offline record-keeping.
Source	GCISL
Priority	Priority Level 0: Essential and required functionality

3. Communication Module

Requirement 1: The system must support in-app messaging between staff and stakeholders.

Description	The system will provide a messaging feature that enables real-time communication between staff members and external stakeholders (such as volunteers or residents' families). This will streamline internal communication and reduce the need for external messaging platforms, improving the efficiency of communication workflows.
Source	Internal requirements elicitation among members of the team
Priority	Priority Level 2: Extra features or stretch goals

4. Volunteer Management Module

Requirement 1: The system must provide a form on the website for volunteers to input their information, including name, email, and contact details.

Description	A dedicated web form will be available for volunteers to register their details, including personal contact information and their availability for volunteering. This will allow GCISL to maintain an organized database of volunteers, ensuring that they can be contacted and scheduled for tasks easily.

Source	GCISL
Priority	Priority Level 0: Essential and required functionality

Requirement 2: The system must allow administrators to export the collected volunteer information into an Excel file.

Description	Administrators will have the ability to export the volunteer database into an Excel file, providing flexibility for data analysis, reporting, and manual editing. This feature ensures that volunteer data can be easily shared with other departments or used for event planning and coordination.
Source	GCISL
Priority	Priority Level 0: Essential and required functionality

II.2.2. [The name of the next module/component/part] Teni

System Reliability Improvements

Automated Testing: Implement tests to check for errors and ensure continuous functionality during updates.

- **Source**: Developer best practices
- **Priority**: 0 (Essential)

Error Logging: Add error tracking and logging for rapid debugging by administrators.

- **Source**: GCISL administrators
- **Priority**: 0 (Essential)

II.3. Non-Functional Requirements Naomi

Reliability

The system shall be reliable and operational most of the time. It must handle user requests without crashing or becoming unresponsive. This should be GCISL's requirement for a stable system to support ongoing research and volunteer tracking.

Response Time

The system shall process and respond to user actions such as login, task assignments within a few seconds. For larger operations, like generating reports, the system shall complete the task within less than 10 seconds. Usability needs for staff and volunteers.

Scalability

The system shall support scalability, accommodating a large number of concurrent users without degradation in performance. This ensures that future expansions to other institutions or larger datasets will be manageable. It would be GCISL's potential expansion to other senior living institutions.

Usability

The system shall be intuitive and easy to use, requiring no more than a few hours of training for an administrator or volunteer to perform basic tasks such as assigning tasks, submitting volunteer reports. The need for accessibility and ease of use for staff, volunteers, and senior residents.

Maintainability

The system shall be modular and well-documented to ensure ease of updates, troubleshooting, and adding new features. New developers should be able to understand the system within a few days of reading the documentation. Requirement for future enhancements and system evolution.

Data Storage and Retention

The system shall store data efficiently, allowing for a minimum of few years of resident and volunteer reports while maintaining optimal performance. Compliance with data retention

III. System Evolution Naomi

The project is based on many assumptions:

- **Flexible and extensible design**

The system will be designed with flexibility and adaptability, allowing for easy future updates, refinements, and extensions. Given that GCISL's needs may evolve in the future, the architecture must support adding new features (example: additional reporting tools) without requiring a complete rebuild of the system. Future research initiatives may introduce new data requirements, necessitating extensions to the data analysis or reporting modules. However, without proper modular design, future changes could become costly and difficult to implement, leading to technical debt.

- **Several and consistent user testing and feedback integration as well evolving user needs**

The system will go through regular user testing, especially with staff, volunteers, and senior residents. Based on feedback, the interface, features, and overall usability will be refined to meet evolving user expectations. User feedback may highlight usability or feature gaps that were previously overlooked, requiring new iterations to improve the system. However, failing to incorporate feedback quickly enough could result in poor user adoption or require costly adjustments post-deployment.

- **Adaptability for broader use**

The system is primarily created for use within GCISL, but with future iterations, we might need to adapt it to other senior living institutions or expand functionality for different research projects. This means the system should be scalable in terms of both user management and data processing capabilities. The system might be adopted by other institutions, necessitating changes in the database architecture to handle a higher volume of users and reports. However, if the system is not adaptable, it may require significant redevelopment efforts to accommodate a larger user base, delaying deployment in new environments.

IV.Glossary Everyone

Database: An organized collection of structured information or data, typically stored electronically in a computer system, that can be easily accessed, managed, and updated.

Scalability: The capability of a system to handle an increasing number of users or transactions without performance degradation.

System Reliability: The ability of a system to consistently perform its intended functions without failure over time.

V. References Everyone