**ELDERLY SOCIAL APPLICATION**

**FINAL REPORT**

**CMSE 322**

**PROJECT NO :4**

**GROUP NO :10**

**PROJECT NAME :ELDERLY SOCIAL APPLICATION**

**PROJECT START DATE :** **15th May 2024**

**PROJECT END DATE :** **26th May 2024**

**SUPERVISOR : Prof.Dr. Duygu Celik Ertugrul**

**SEMESTER TERM :** **Spring 2023/2024**

GROUP MEMBERS

|  |  |  |
| --- | --- | --- |
| Edidiong Etuk | **ID No** | 21902156 |
| Emmanuel Trevor Magala | **ID No** | 21901595 |
| Dally Aita | **ID No** | 18701149 |
| Narin Sila Laleci | **ID No** | 19000092 |

**Computer Engineering Department**

**Eastern Mediterranean University**

**Term: Term 2 and Year 3**

# ABSTRACT

The Elderly Social Application is aimed at fostering connections among old people and combat loneliness. Our application is aimed at making it easy for our users to communicate and making it easy to use. The application is a simple implementation of a social application that enables users to send messages, images, search for other users as well as to create groups and communities depending on their preferences. We started this project with planning phase where we carried out meetings on how to handle different parts of the application, we then moved onto the design stage, where we used tools like modelio to draw diagrams such use case diagrams, activity diagrams etc. For the implementation stage we used different languages such as HTML, CSS, REACT, JAVASCRIPT and SQL for our database. We also did some tests on our website to see whether it could implement the functionalities. We also prepared several reports to show our progress i.e. SRS and SDS reports.

Finally, we obtained a web social Application, and we know the different stakeholders will benefit from it . With the time given to us we consider this a step forward to making communication easier for the elderly people/group in our community

**Keywords:** ELDERLY, LONILNESS, CONNECTIONS, HTML, CSS,SQL,REACT

# Table of Contents

[ABSTRACT II](#_Toc478723910)

[Table of Contents III](#_Toc478723911)

[LIST OF FIGURES V](#_Toc478723912)

[LIST OF TABLES VI](#_Toc478723913)

[1. INTRODUCTION 1](#_Toc478723914)

[2. PROJECT PLANNING AND MANAGEMENT 2](#_Toc478723915)

[3. REQUIREMENTS ANALYSIS 3](#_Toc478723916)

[3.1 Functional Requirements 3](#_Toc478723917)

[3.2 Non-Functional Requirements 3](#_Toc478723918)

[3.3 Realistic constraints 3](#_Toc478723919)

[3.4 Ethical issues 3](#_Toc478723920)

[4. DESIGN 4](#_Toc478723921)

[4.1 High level design (architectural) 4](#_Toc478723922)

[4.2 Software design 4](#_Toc478723923)

[5. IMPLEMENTATION 5](#_Toc478723924)

[5.1 Tools, technologies and platforms used 5](#_Toc478723925)

[5.2 Algorithms 5](#_Toc478723926)

[5.3 Standards 5](#_Toc478723927)

[5.4 Detailed description of the implementation (coding) 5](#_Toc478723928)

[6. TESTING 6](#_Toc478723929)

[7. USER GUIDE OF THE SYSTEM 7](#_Toc478723930)

[8. DISCUSSION 8](#_Toc478723931)

[9. CONCLUSION 9](#_Toc478723932)

[10. REFERENCES 10](#_Toc478723933)

[APPENDICES 11](#_Toc478723934)

[A. Instructions for installing the system 11](#_Toc478723935)

[B. Code for the system 11](#_Toc478723936)

[C. Other relevant material 11](#_Toc478723937)

# LIST OF FIGURES

[Figure 2 Network Diagram (w/ Critical Path) 9](#_Toc514793471)

[Figure 4 Architecture Overview 1 18](#_Toc514793473)

[Figure 6 Level 0 DFD 19](#_Toc514793475)

[Figure 7 Use Case Diagram 20](https://d.docs.live.net/deb243adb1f9c3c7/7-Documents/EMU/CMSE%20322/WIP%204/3-SOFTWARE_FINAL.docx#_Toc514793476)

[Figure 8 UML Class ER Diagram 21](#_Toc514793477)

[Figure 9 Activity Diagram 21](https://d.docs.live.net/deb243adb1f9c3c7/7-Documents/EMU/CMSE%20322/WIP%204/3-SOFTWARE_FINAL.docx#_Toc514793478)

[Figure 12 BPMN 24](#_Toc514793481)

[Figure 14 Sequence Diagram 26](#_Toc514793483)

# [Figure 16 User Guides 31](#_Toc514793485)

# LIST OF TABLES

[Table 1 Work Package 1 2](#_Toc514793462)

[Table 2 Work Package 2 3](#_Toc514793463)

[Table 3 Work Package 3 4](#_Toc514793464)

[Table 4 Work Package 4 5](#_Toc514793465)

[Table 5 Risk Table 6](#_Toc514793466)

[Table 6 Procurement Table 7](#_Toc514793467)

[Table 7 Activities/Tasks 8](#_Toc514793468)

[Table 8 Paths 8](#_Toc514793469)

# 1. INTRODUCTION

# The proposed project is, “Application: Fostering Social Connections for Elderly Well-being” this is an Android app that will be developed with an interface explitably designed for use by elders so as to provide a method through which the elderly in North Cyprus could be connected hence alleviating their loneliness. There has been worsening of loneliness as a result of the COVID 19 epidemic which causes such serious conditions as depression and deterioration of cognition. The application assists in the establishment of a social platform through which users may be engaged in interested tasks, converse with each other using custom avatars, and foster friendships. The low contrast and readability, timely content updates, and English/Turkish availability make it more friendly to first-time attendees with limited computer skills. Priority features are, of course, multiple language support, tasks that contribute to the improvement of the mind faculties, and using the user-centered approach in development with users involved in the test phases. Apart from enhancing contribution and increasing technological accessibility for users with limited digital literacy, it is vital to reduce elderly’s isolation and loneliness, enhance their mental health and enrich their lives for the meaning of human being is, indeed, the quality of relationships within their social network. It may also be useful for promoting more inventions and investments in this area in Northern Cyprus.2. PROJECT PLANNING AND MANAGEMENT

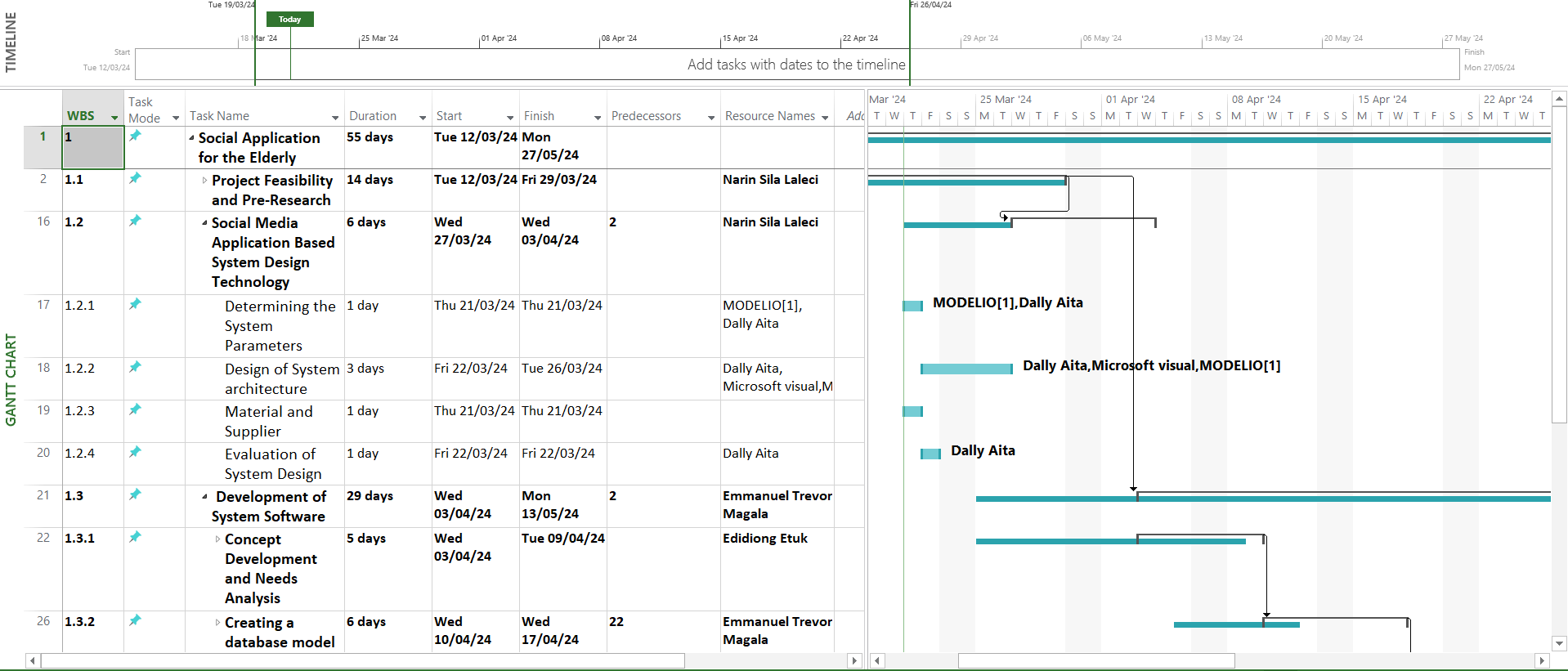
Project Definition, Aim, Scope, Target Users:

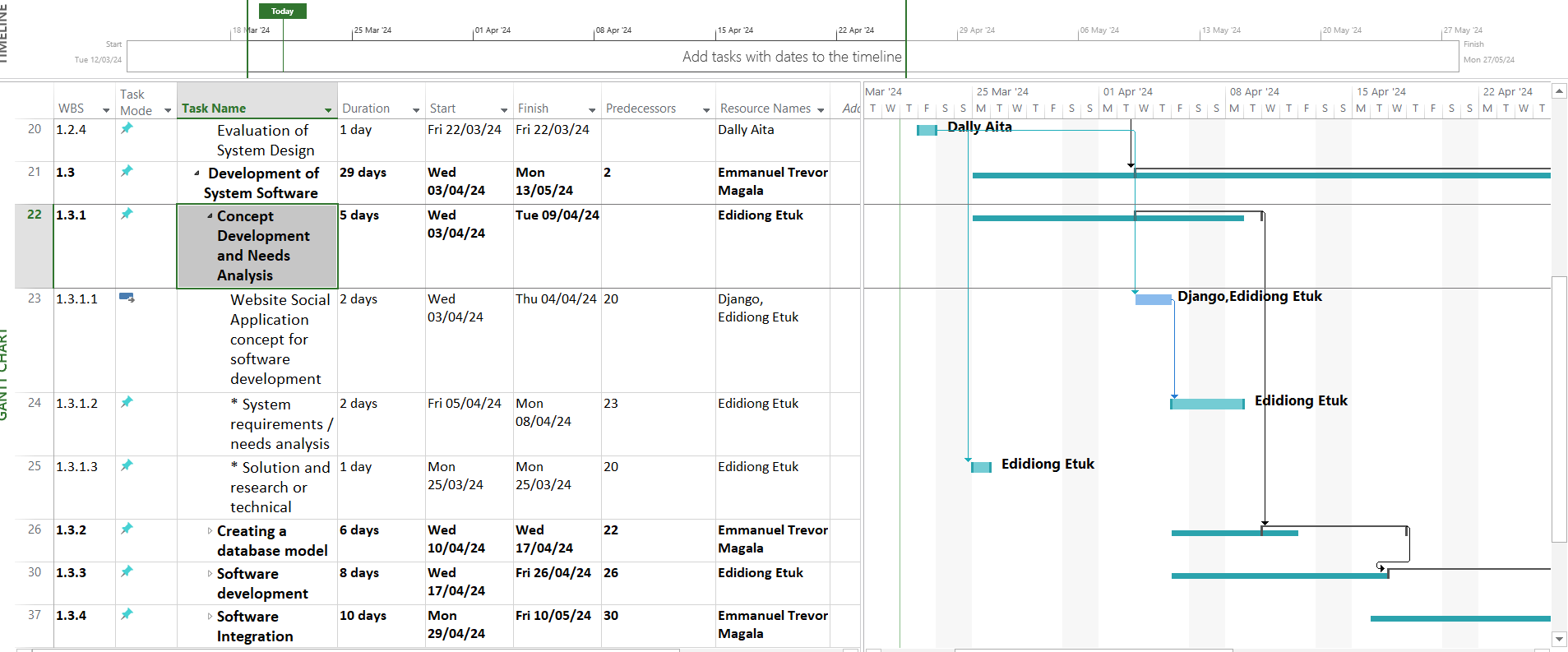
The idea behind the project “The application supporting elderly people in creation of their social networks for the purpose of their wellbeing enhancement” is to develop an intuitive Android application that would help elderly people improve their social interactions in order to reduce loneliness. It provides an online platform with avatar to enable the users to make new friends and it is targeted at only men and women above 60 years of age living in North Cyprus. This program aims at countering the impacts of loneliness in the community that has resulted from the outbreak of COVID-19. This also provides a means for achieving objectives set down by the World Health Organization and the AARP Foundation. The overall goal or purpose of the platform is to create an app based social network that fosters ones to ones so that senior citizens feel connected and engage with folks like them in a way that is both healthy and stimulating while making technology friendly and enjoyable. In more detail, the goals of the project encompass the development of an Android application with the following options: the possibility to have the application in both English and the country native language; the application’s possibility to have different font sizes to make it easy for the seniors to read; a user-friendly interface that will suit seniors since many of them are not very computer literate. The app will also have regular exercises that entail mental and cognitive health thus pushing the users to get active and live healthy lives. It is mainly for people over the age of 60 who reside in Northern Cyprus . It is comprehensive and efficient and does not render a segregation between the technologically inclined persons and those who are relatively new to computer usage.

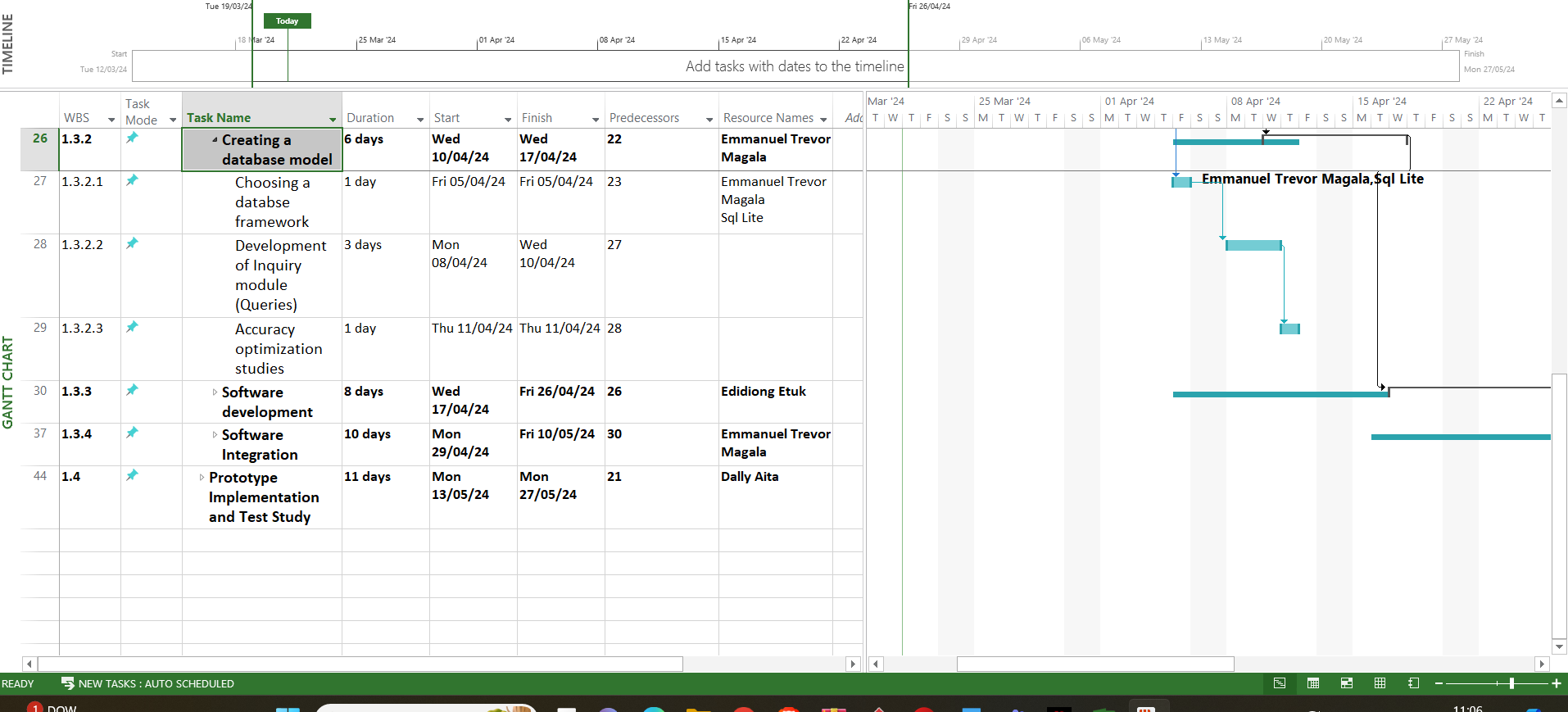
Work Breakdown Structure (WBS):

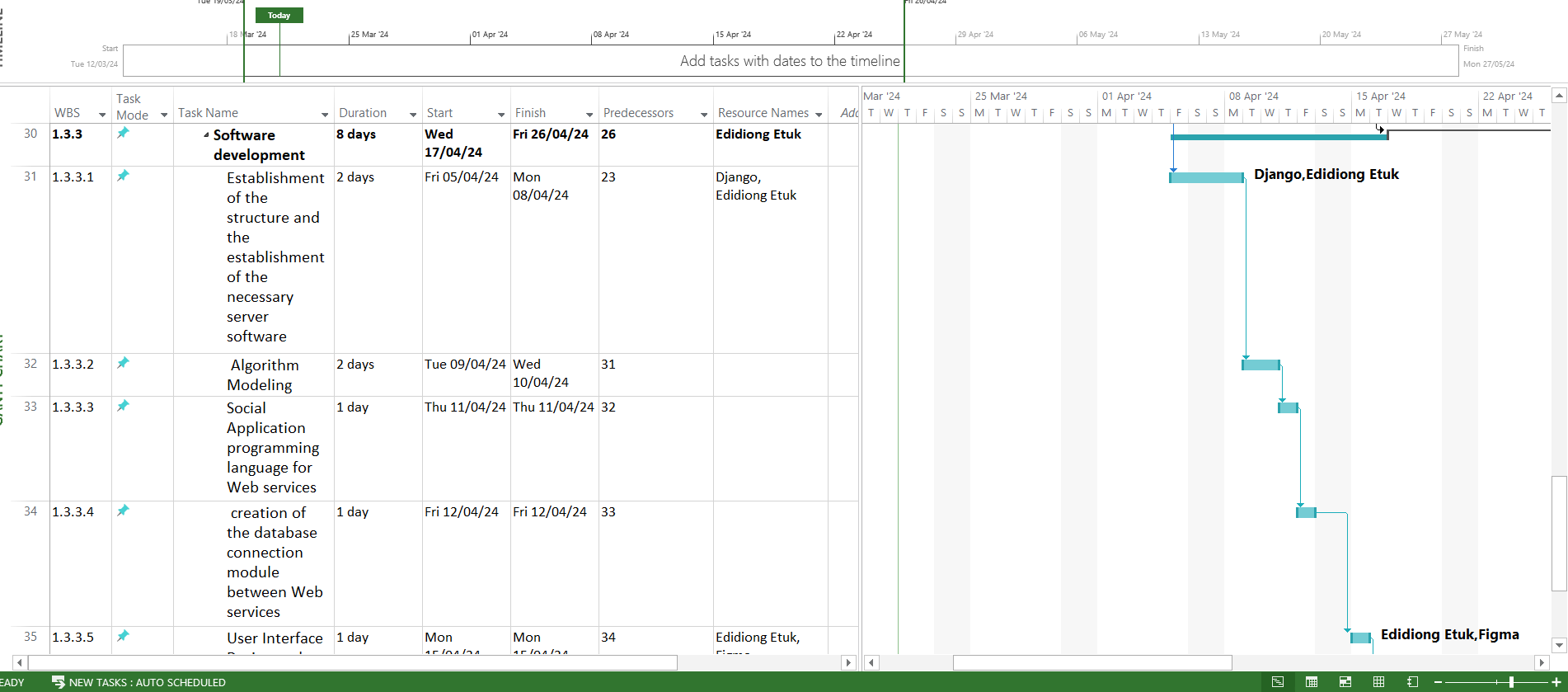
Project development phases include the Project initiation; The system design; Development; Testing & implementation; and Maintenance based on WBS. During project initiation phase plans and specifications are set, preliminary data about the market is collected, the division of work is channeled to the team that works on the project charter, risk and their control responses are noted, estimates of costs and profitability are made. Requirements definition, system structure, data organization and management, input and output design, data security and safety modeling, and constructing initial models and prototypes are accomplished in the system design phase. The development phase entails aspects such as developing frameworks, coding techniques, putting up of databases, designing interfaces, and employing the utilization of CASE tools. Testing and implementation involves checking security, real and non-functional requirements, examining user acceptability and compatibility of interfaces, checking compatibility with the database and servers as well as ensuring standarad compliance. The maintenance phase entails conducting bug fixes recurrently, making alterations to the program wherein improvements are needed according to users, and ensuring that the program adheres to the innovations in technology.

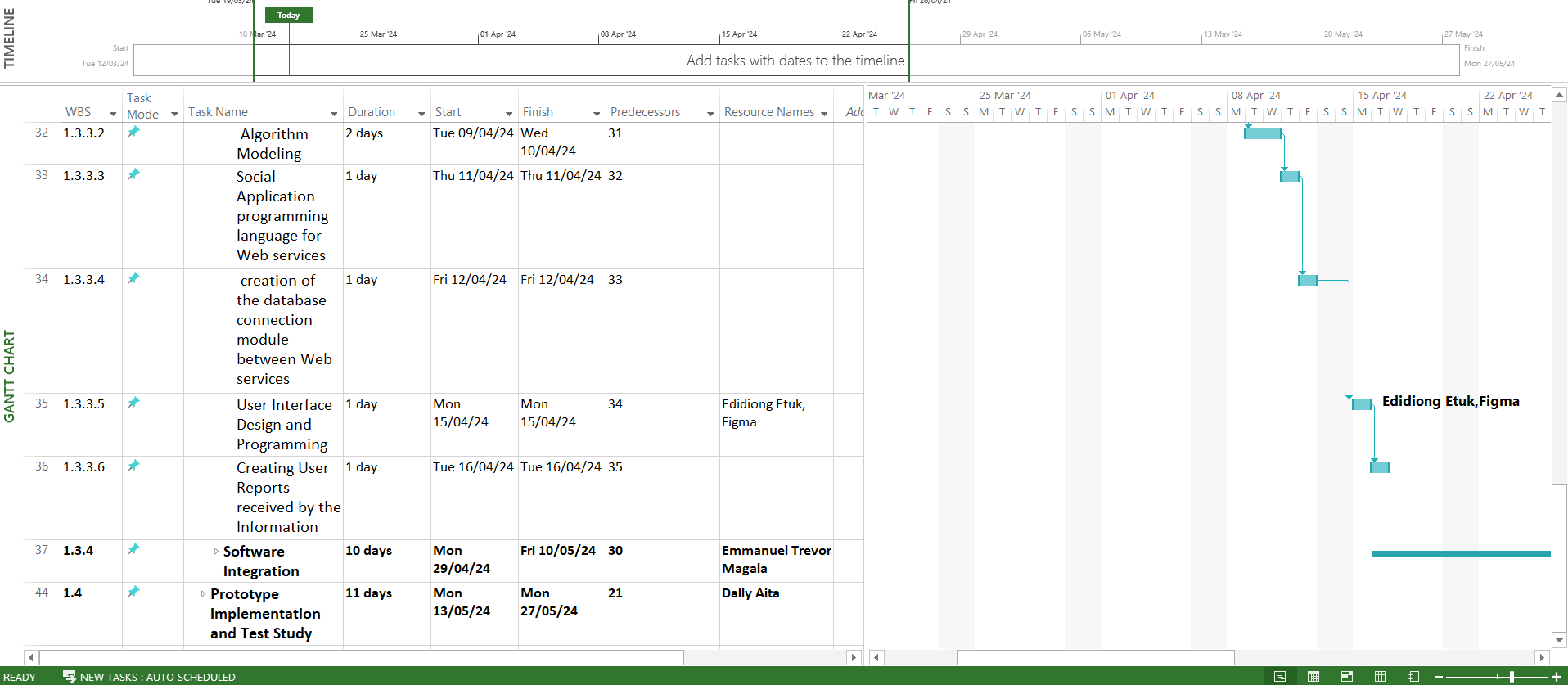
Gantt Chart:

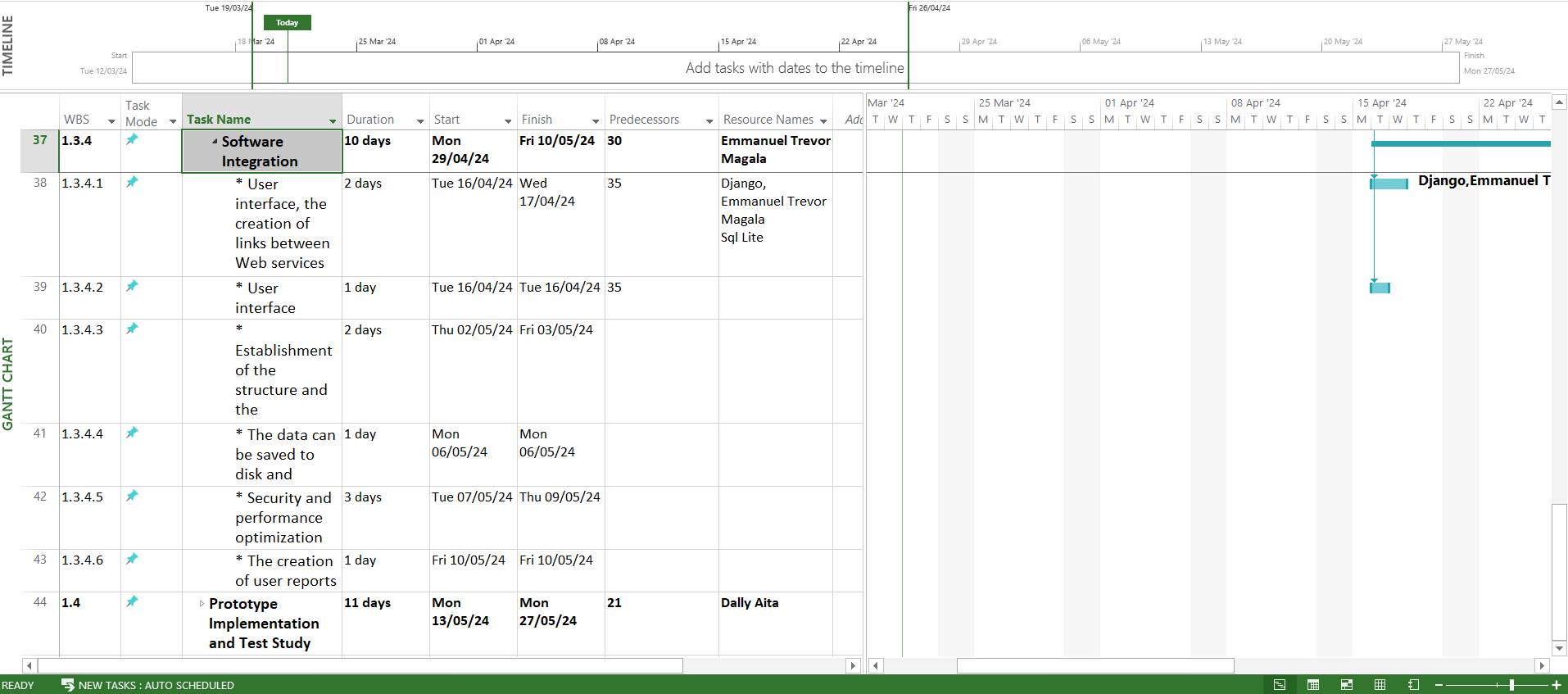


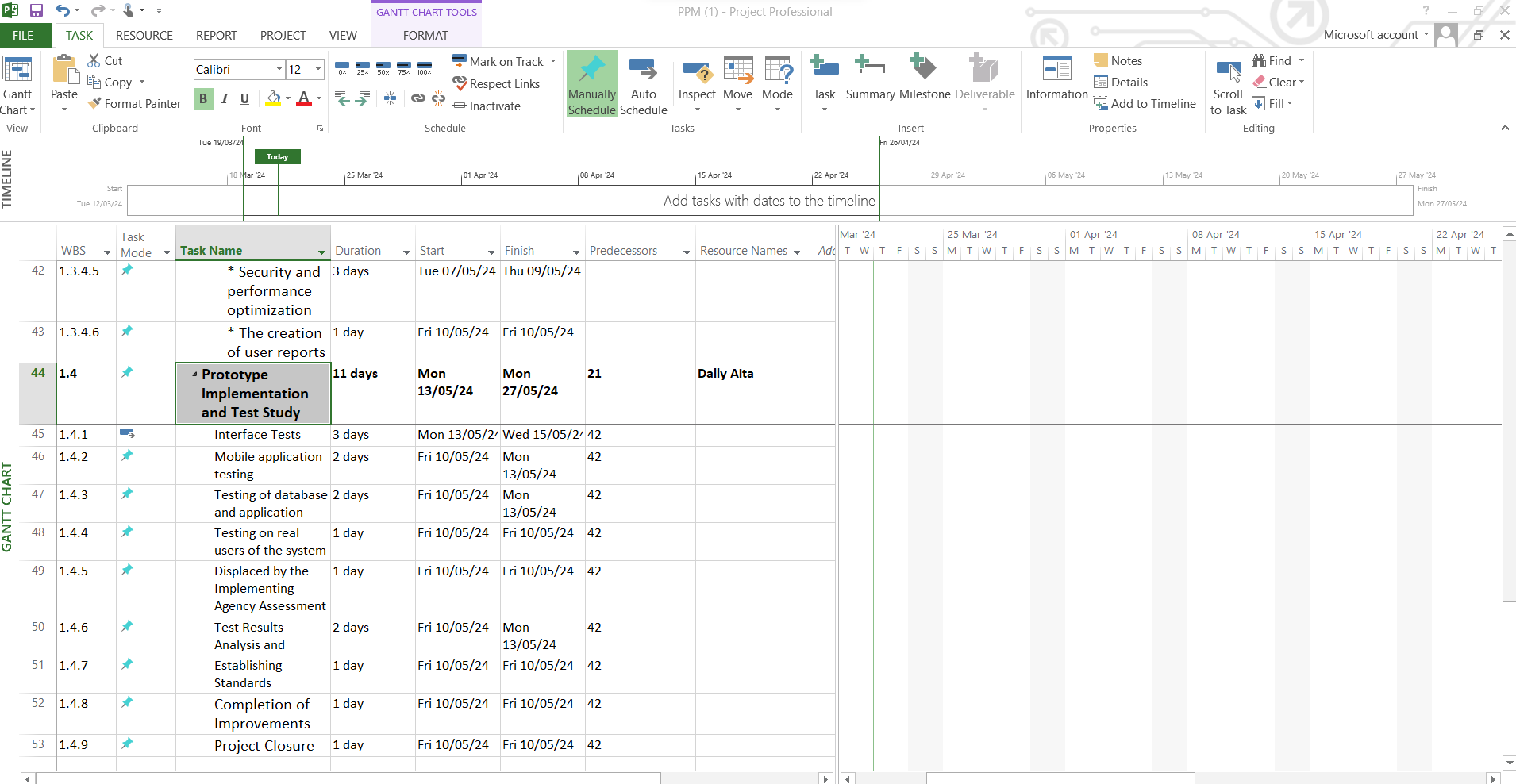












Project Package Tables:

|  |  |
| --- | --- |
| **Work Package No** | 1 |
| **Work Package Name** | **Project Feasibility and Pre-Research (Feasibility Analysis)** |
| **Start-End Date and Time** | 12.03.2024 - 29.03.2024 |
| **Related Organizations** |  |

|  |
| --- |
| **1- List the activities of work packages.** |
| **1.1 Project Process and Economic Feasibility:**  • Definition of project objectives and its scope  • Perform initial market research on the demand for communication apps for the elderly  • Formulate the project teams and assign roles to its members  • Development of a project charter that outlines the purpose, objectives and constraints of the project  • Identify potential risks and mitigation strategies  • Perform production cost and profitability projection  **1.2 Technological Feasibility:**  **•** Evaluate the technological requirement for the development of the android application  • Decide the hardware constraints for the end user |
| **2- Describe the methods and parameters that will be used for work package.** |
| The method that will be used for the feasibility study includes market research, controlled user(elderly) interviews, and technical analysis to assess the economic viability and user needs. Parameters such as development costs, revenue projections and ROI will be evaluated. |
| **3- List the experiments, tests and analysis in the work package.** |
| • Online research  • Cost-Benefit analysis  • Risk analysis  • Project specification clarity test |
| **4- List the output of work package and its success criterias.** |
| **Outputs:**  • Examining the practicality and sustainability of the project  • Assessment of the risk  • Analyzing the cost of production and the return of investment within a specified duration  **Success Criteria:**  **•** Clear definition of the objectives of the project  • Risks have been analyzed and carefully considered prior to advancing with the project  • Clear understanding of time, economic and technological constraints |
| **5- Explain the relation of output with other work packages** |
| This work package serves as the cornerstone for subsequent activities, providing essential information to support the project’s implementation |

|  |  |
| --- | --- |
| **Work Package No** | 2 |
| **Work Package Name** | **Based System Design Technology (Analysis & Design stage)** |
| **Start-End Date and Time** | 03.04.2024 - 13.04.2024 |
| **Related Organizations** |  |

|  |
| --- |
| **1- List the activities of work packages.** |
| **•** Requirement analysis  • System architecture design  • Database design  • User interface design  • Security design  • Modelling and prototyping |
| **2- Describe the methods and parameters that will be used for work package.** |
| • Use of UML modeling techniques  • Prototyping  • Compatibility testing  • Security assessment |
| **3- List the experiments, tests and analysis in the work package.** |
| • Develop requirement documents based on the information gathered  • Prototype creation  • Resource definition and allocation  • Research about similar communication application’s system architecture |
| **4- List the output of work package and its success criterias.** |
| **Outputs:**  **•** Requirements documents reflecting gathered information  • Prototypes of the android application  • Preliminary system design and development plans  • Defined and allocated resources  • Description of interdependence among system functions and capabilities  **Success Criterias:**  **•** Validation of system scope and structure  • Cost and risk refined  • Convenient models and methodologies |
| **5- Explain the relation of output with other work packages** |
| This phase evaluates the quality of pre-research, determining how collected information can be presented for implementation. It specifies requirements to ensure adherence, supporting progress towards optimal system functionality |

|  |  |
| --- | --- |
| **Work Package No** | 3 |
| **Work Package Name** | **Development of System Software (Development Stage)** |
| **Start-End Date and Time** | 03.04.2024 - 13.05.2024 |
| **Related Organizations** |  |

|  |
| --- |
| **1- List the activities of work packages.** |
| **•** Designing User Interfaces  • Database design  • Leveraging frameworks  • Algorithm development and coding  • Utilizing CASE tools |
| **2- Describe the methods and parameters that will be used for work package.** |
| • Utilize HTML for frontend structure  • Apply CSS for styling  • Implement ASP.NET MVC for server-side web application  • Establish the relational database using SQL |
| **3- List the experiments, tests and analysis in the work package.** |
| • Examine functional specifications  • Utilize CASE tools to aid system development  • Write codes for various modules  • Create database components  • Conduct developer testing |
| **4- List the output of work package and its success criterias.** |
| **Outputs:**  **•** Completed code module  • Codebase and database development  • Utilization report of CASE tools  • Developer testing report, focusing on unit testing results  **Success Criteria:**  **•** Adherence to Rational Unified Process  • Quality of code and Database  • Well-designed User interface |
| **5- Explain the relation of output with other work packages** |
| The result of this process is the functional prototype embodying all system requirements and user interface designs. Subsequently, it will undergo testing for validation, utilizing design and requirement packages to ensure accurate and comprehensive functionality. |

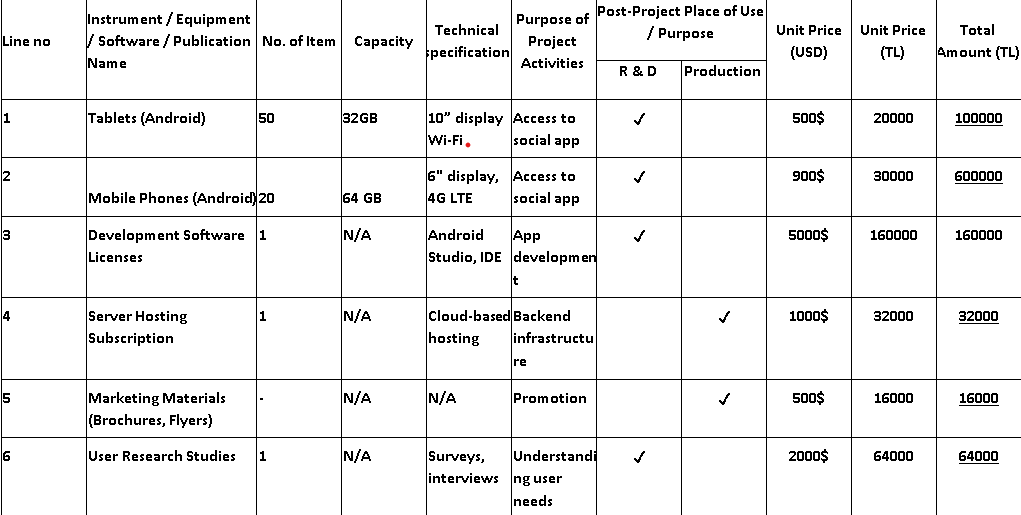
|  |  |
| --- | --- |
| **Work Package No** | 4 |
| **Work Package Name** | **Prototype Implementation and Test Study and Maintenance (Test & Maintenance stage)** |
| **Start-End Date and Time** | 13.05.2024 - 27.05.2024 |
| **Related Organizations** |  |

|  |
| --- |
| **1- List the activities of work packages.** |
| **•** Security assessment  • Functional and non-functional validation  • User acceptance verification  • Interface examination  • Compatibility check  • Database and server testing  • Standard compliance confirmation  • Project conclusion |
| **2- Describe the methods and parameters that will be used for work package.** |
| • Conduct performance evaluations  • Perform usability assessments  • Execute unit and integration tests  • Validate system functionality  • Test web load capacity  • Validate HTML/CSS  • Perform maintenance checks |
| **3- List the experiments, tests and analysis in the work package.** |
| • Performance benchmarking on different device  • A/B testing for interface variations  • Automated regression testing for code changes  • Load testing to assess system scalability  • Bug triage to prioritize fixes based on impact |
| **4- List the output of work package and its success criterias.** |
| **Outputs:**  **•** Test result and reports  • Incident reports  • Release testing documentation  • Project readiness for release  **Success Criterias:**  **•** The application is fully developed and operational  • Comprehensive test suite ensuring functionality |
| **5- Explain the relation of output with other work packages** |
| As the concluding work package, the flawless execution ensures project completion, signaling readiness for release. |

**Risk Analysis Table**

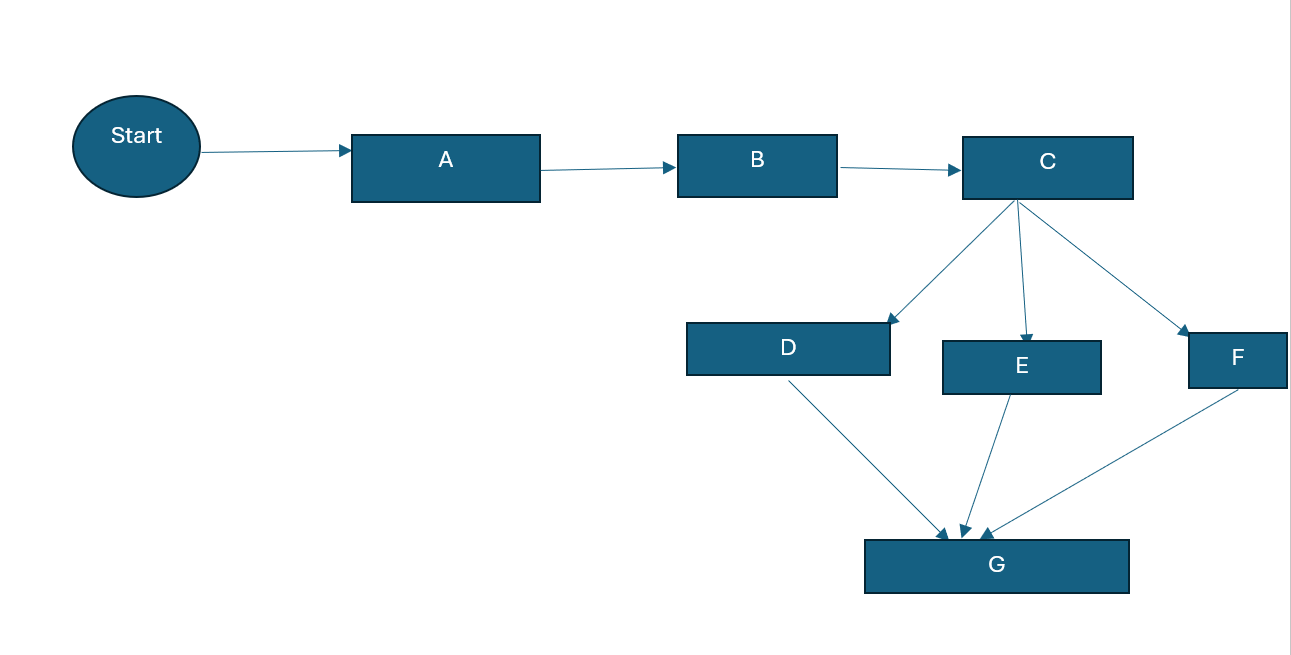
|  |  |  |  |
| --- | --- | --- | --- |
| Risk | Probability | Effects | Your Strategy |
| The time required to develop the software is underestimated. | High | Serious | Frequent progress reports and adaptable timeframe modifications |
| Software tools cannot work together in an integrated way. | High | Tolerable | Early compatibility testing and integration testing |
| Customers fail to understand the impact of requirements changes. | Moderate | Tolerable | Thoroughly documented requirements, frequent customer feedback meetings |
| The rate of defect repair is underestimated. | Moderate | Tolerable | Replace potentially defective components with more reliable bought-in components. |
| The size of the software is underestimated. | High | Serious | Investigate buying sw components;  Investigate use of a program generator. |
| Code generated by code generation tools is inefficient. | Moderate | Insignificant | Handle code optimization and choose effective techniques. |
| Key staff are ill at critical times in the project. | Moderate | Serious | Reorganize team so that there is more overlap of work and people therefore understand each other’s jobs. |
| The database used in the system cannot process as many transactions per second as expected. | Moderate | Serious | Investigate the possibility of buying a higher-performance database. |

**Procurement Table:**





**Network Diagram:**



# 3. REQUIREMENTS ANALYSIS

## 3.1 Functional Requirements

The system's goal is to help senior users build social ties in order to reduce loneliness and enhance their overall wellbeing. This program will be especially easy to use for senior citizens and offer a virtual world in which they may communicate with each other through avatars. The main participants in the system are:

1.Elderly Users: An easy-to-use and accessible user interface will be the main method of interaction for these users with the system. They'll be capable of:

* Create and manage personal profiles.
* Search for and interact with other users.
* Send and receive messages.
* Join and create groups.

2. System Administrators: In responsible for handling system settings, content, and user accounts. They are going to:

* Oversee user account creation.
* Delete account.
* Update content.
* Manage the overall integrity of the system.
* Provide technical support and ensure the system runs smoothly.

**Functional Requirements:**

**Account Creation/Management**

***Description and Priority:***

These features will be provided to the user to create an account and to also access their account at any time. They are the standard user and system interactions.

***Stimulus/Response Sequences:***

Register a New Account:

Stimulus: Upon accessing the application home page, a user clicks on create new account

Response: The system displays an interface to create a new account

Log in:

Stimulus: User enters username and password

Response: System will display user page if information is valid

Forgot Password:

Stimulus: User clicks on forgot password

Response: The system displays an interface to change password where it verifies user information.

***Functional Requirements***

* REQ-1: New users will be able to register with their emails, username and preferred password.
* REQ-2: Returning users will login with their username and password.
* REQ-3: Users that have forgotten their passwords will be able to reset password through a verification page.
* REQ-4: Users will be able to change to language settings of their preference.
* REQ-5: The system will be able to detect and notify users about invalid data fields entered.
* REQ-6: Users shall be able to log out when they are leaving the page.

**User Page/Interaction**

***Description and Priority:***

Users will be able to access a wide range of features such viewing other user profiles, sending messages to other users, joining groups, creating groups, searching for other users, and seeing whether users are online or offline.

***Stimulus/Response Sequences:***

* View another user's profile:
* Stimulus: User clicks on another user's profile
* Response: The system displays profile details of another user

* Searching for other users:
* Stimulus: User clicks on search bar and types of names/usernames of user they are searching for.
* Response: System displays a list of possible users and user selects who they are searching for.

* Joining groups:
* Stimulus: People who have created groups will be able to add anyone they want to communicate with.
* Response: Users will be able to see other members in the group and be able to send messages to the group

* Creating Groups:
* Stimulus: Users will click on an icon to create a new group
* Response: After group is created users can add other users

* Sending Messages:
* Stimulus: User clicks on another user profile.
* Response: User can send messages and pictures.

* Seeing weather other users are online or offline:
* Stimulus: Users will click on other user's profiles
* Response: Users will see whether other users are offline or online.

***Functional Requirements***

REQ-1: Users shall be able to view profiles

* The system shall provide an option to view profiles of other users.
* The system shall display information such as name bio, profile, picture and show whether user is online or offline.

REQ-2: The system shall display autocomplete suggestions for usernames/names as the user types in the search bar.

REQ-3: The system shall allow group creators to add members and facilitate group communication.

REQ-4: The system shall display online/offline status for users and enable messaging and media sharing between users.

REQ-5: The system shall provide group creation functionality with the ability to add and manage members.

REQ-6: The system shall support direct messaging and media sharing between users and display their online/offline status.

**Admin Account Management:**

***Description and Priority:***

The administrators need to be able to have an account so that they can access the admin features and make updates to the application.

***Stimulus/Response Sequences:***

The administrator will login using their password and username and then be able to access the admin panel.

***Functional Requirements:***

REQ-1: The administrators shall be able to register into the system as admin.

REQ-2: The administrators shall be able to login and logout.

REQ-3: The administrators shall be able to verify their login credentials.

REQ-4: The administrators shall be able to access the admin panel.

**User Account Management**

***Description and Priority:***

Administrators will be able to create and delete user accounts and groups. They however will not be able to view user's messages for privacy reasons

***Stimulus/Response Sequences:***

* Delete Account:
* Stimulus: Admin clicks on delete user account
* Response: Account deleted
* Create Account:
* Stimulus: Create user account with username email and password
* Response: Account created
* Create/Delete group:
* Stimulus: Admin clicks on Create/delete user group
* Response: Group created/deleted
* ***Functional Requirements:***
* REQ-1: The system shall allow the admin to delete user accounts upon request.
* REQ-2: The system shall provide functionality for creating user accounts with username, email, and password.

REQ-3: The system shall enable the admin to create and delete user groups as needed.

**User Settings**

***Description and Priority:***

These settings are essential for enhancing the user experience and managing preferences within the chatting app.

***4Stimulus/Response Sequences:***

* **Change Profile Picture**

Stimulus: User selects the option to change their profile picture.

Response: User can upload a new image from their device to set as their profile picture.

* **Notification Preferences**

Stimulus: User accesses the settings menu to manage notification preferences.

Response: User can toggle notifications for messages, group activity, mentions, etc., based on their preference.

**Username Change:**

**Stimulus:** User accesses the user settings menu and selects the option to change their username.

* **Response:** The system prompts the user to enter their desired new username.

**Stimulus:** User enters the new username.

* **Response:** The system validates the username for uniqueness and checks if it meets any criteria (e.g., length, allowed characters).

**Stimulus:** User submits the new username.

* **Response:** If the new username is unique and meets the criteria, the system updates and displays the new username across the app.

**Password Change:**

**Stimulus:** User navigates to the user settings menu and chooses the option to change their password.

* **Response:** The system prompts the user to enter their current password for verification.

**Stimulus:** User enters their current password.

* **Response:** The system verifies the current password before proceeding.

**Stimulus:** User inputs a new password and confirms it.

* **Response:** The system enforces password strength criteria (e.g., minimum length, special characters) and securely updates and stores the user's new password.

**Language Change:**

**Stimulus:** User accesses the user settings menu and selects the option to change the app's language.

**Response:** The system displays a list of supported languages for the user to choose from.

**Stimulus:** User selects a new language from the list.

**Response:** The system applies the selected language settings to the app's interface and notifications.

**REQ-1: Change Profile Picture**

1.The system shall provide an option for users to change their profile picture.

2.The system shall allow users to upload an image from their device to set as their profile picture.

3.The system shall display the updated profile picture across the app after the user makes changes.

**REQ-2: Notification Preferences**

1.The system shall offer notification settings in the user settings menu.

2.The system shall allow users to customize notification preferences for different types of activities such as messages, group activity, mentions, etc.

3.The system shall implement toggles or checkboxes for users to enable or disable specific types of notifications based on their preferences.

The system shall save and apply the user's notification preferences across app sessions.

**REQ-3: Language Change**

1.The system shall provide an option for users to change the app's language in the user settings menu.

2.The system shall offer a list of supported languages for users to choose from.

3.The system shall apply the selected language settings to the app's interface and notifications.

**REQ-4: Password Change**

1.The system shall allow users to change their password in the user settings menu.

2.The system shall require users to input their current password for verification before changing to a new password.

3.The system shall enforce password strength criteria (e.g., minimum length, special characters) for new passwords.

4.The system shall securely update and store the user's new password.

**REQ-5: Username Change**

1.The system shall enable users to change their username in the user settings menu.

2.The system shall validate that the new username is unique and not already in use by another user. The system shall display an error message if the chosen username is already taken.

3. The system shall update and display the new username across the app after the change is made. give the stimulus and response sequences for the username, password change, language change

**Nonfunctional Requirements**

**A. Performance Requirements**

1. The system should be optimized to remain high in its response time during heavy traffic.

 2. The transitions between application states like populating the next chat room or going to some other part of application should feel smooth and they should occur in less than a second.

 3. The implementation must be efficient to provide the feeling of getting a response from the system almost instantly.

 4. The content must also take less time to load, preferably within two seconds for images and media.

 5. This rule implies that as the chat continues, every new message that is inputted should appear on the other person’s screen almost immediately with little or no lag time.

 6. The intended user system must be able to support multiple group chats at one time, each of which is expected to support up to 50 users.

 7. The system should be able to manage and present information from active group chats as well as the capability of handling more than one chat at a time.

 8. Speed: A query run should, at most, take five seconds even where the query is for many records.

 9. Old message retrieving should take not more than three seconds, which will enhance the scrolling through in chats.

 10. The system should answer this by being accurate each time data operations are performed on the system.

 11. In addition, this should be in a way that efficiency compromises are not made in that the system should never degrade in performance even if it is handling many user requests at a time.

 12. It must be efficient so that when it is working on large data, there should not be any errors that occur.

**Safety Requirements**

Due to the fact that clients may use the system from remote locations, all communications between the data server and any external client application must be encrypted to avoid vulnerability to hackers.

 2. To ensure private access to the subsystems in the system, there will be a core user login that will need the user details such as (Email, Name, Password).

 3. Data must be backed up on each system at least once in 24 hours and stored preferably in a different site from the system and can be stored in hard disks, CD, and flash disks.

 4. It should also have proper backup copies of the data that will prevent leakage of data and easiness when it comes to data restoration in case of any mishaps or accidents.

 4. Using the forgotten password link, the account should first check and ensure that the user is the rightful owner of the account before resetting the password through a process such as email confirmation or answering several security questions.

 5. Passwords should not be stored in plain text, instead they should be encrypted by hashing them with the help of a method of salted encryption just in case the database is penetrated by an attacker.

 6. Security in the system should be checked regularly to ensure that the system has no security loopholes whereby an outsider or an unauthorized user can sneak through and corrupt it.

 7. The information within the domain must be moderated to prevent exercising the impact of hatred or perverted minds within the community.

 8. There should be options to flag any abuse or violation of community standards and policies to identify and remove abusive users or content.

 9. Information provided by the users should be handled in accordance to the Data Protection Act to protect the rights of the users by enhancing the way information is collated and used.

 10. Some of the suggested measures included a user password must meet certain complexity level that enhances the level of security against password guessing attacks.

 11. It is also important that data is backed up often, and stored away from the original data, physically, to ensure that should an update go bad, or the system crash, there is a viable back up.

3. **Usability**:

* Large text and icons should be used in a straightforward and user-friendly design.
* The navigation should be simple enough for users with little technological knowledge.

• There should be a help area with FAQs and lessons.

4. **Accessibility**:

* The application needs according to WCAG guidelines.
* Text-to-speech and voice commands are two features that should be available to people with disabilities.

5. **Evolution Requirements**:

* **Testability**: Different modules should be able for automated testing within the system.
* **Maintainability**: The codebase needs to be thoroughly documented and conform to best practices.
* **Extensibility**: The architecture of the system should make it simple to add new features.
* **Scalability**: The program needs to be able to support extra users without experiencing any performance problems.

**3.3 Realistic Constraints**

1. **Economic**:

* Users shouldn't need to make a substantial financial commitment in order for the system to be affordable.
* Budgetary restrictions should be observed when it comes to development and maintenance charges.

2. **Environmental**:

* The program needs to be designed to use the least amount of electricity possible.
* It should promote sustainable habits and not add to the garbage generated by electronics.

3. **Social**:

* The program needs to be inclusive and suitable for a wide range of senior citizens.
* Restrictions based on age can be required to protect the community of users.

4. **Political**:

* The application must abide by national and international laws pertaining to privacy and data protection.

5. **Ethical**:

* Unattributed third-party code should not be used in the development process.
* Handling user data ethically is necessary to protect consent and privacy.

8. **Sustainability**:

* Regular updates and support should be provided, and the system should be built for long-term usage.

**3.4 Ethical Issue**

1. **Privacy**:

* Guarantee the security of user data and prevent unwanted access.
* Establish clear guidelines for data usage.

2. **Safety**:

* Keep an eye on and stop the platform from being abused for illegal activities.
* Put in place systems for reporting abusive behavior.

3. **Equity**:

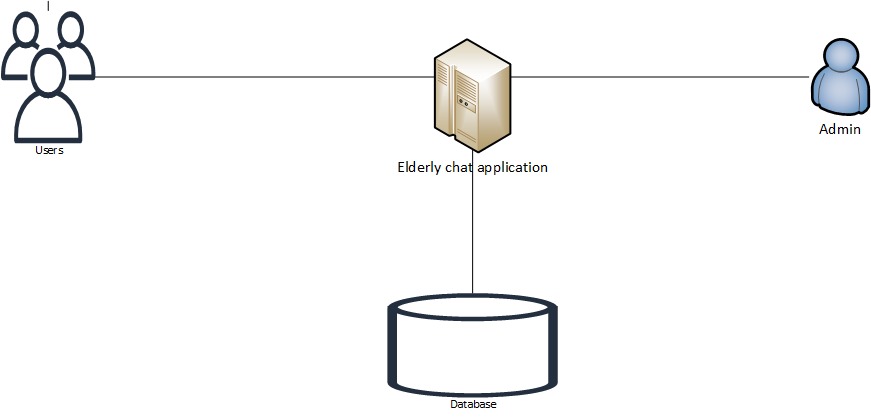
* Guarantee equitable access for all users, no matter their financial situation or level of technological proficiency.

5. **User Consent**:

* Before collecting or using user data, get their express consent.

# 4. DESIGN

## 4.1 High level design (architectural)



Explanation:

This is the design of the system containing users in the front end and there is an admin panel for the system manager . Then the backend contains the database where we are going to store and retrieve our user data

CONTEXT DIAGRAM

A diagram of an older chat application

Description automatically generated

Explanation:

The context diagram above shows the main system i.e. Elderly Chat Application how the different actors in the system interact with each other

For example:

Users: User profile when logging in, notifications to users from system, sending and receiving messages,

Admin: Announcements ,getting feedback from user, different user reports, banning users, creation of users.

LEVEL 0 DATA FLOW DIAGRAM

A diagram of a network

Description automatically generated with medium confidence

LEVEL 1 DFD

A screenshot of a computer screen

Description automatically generated

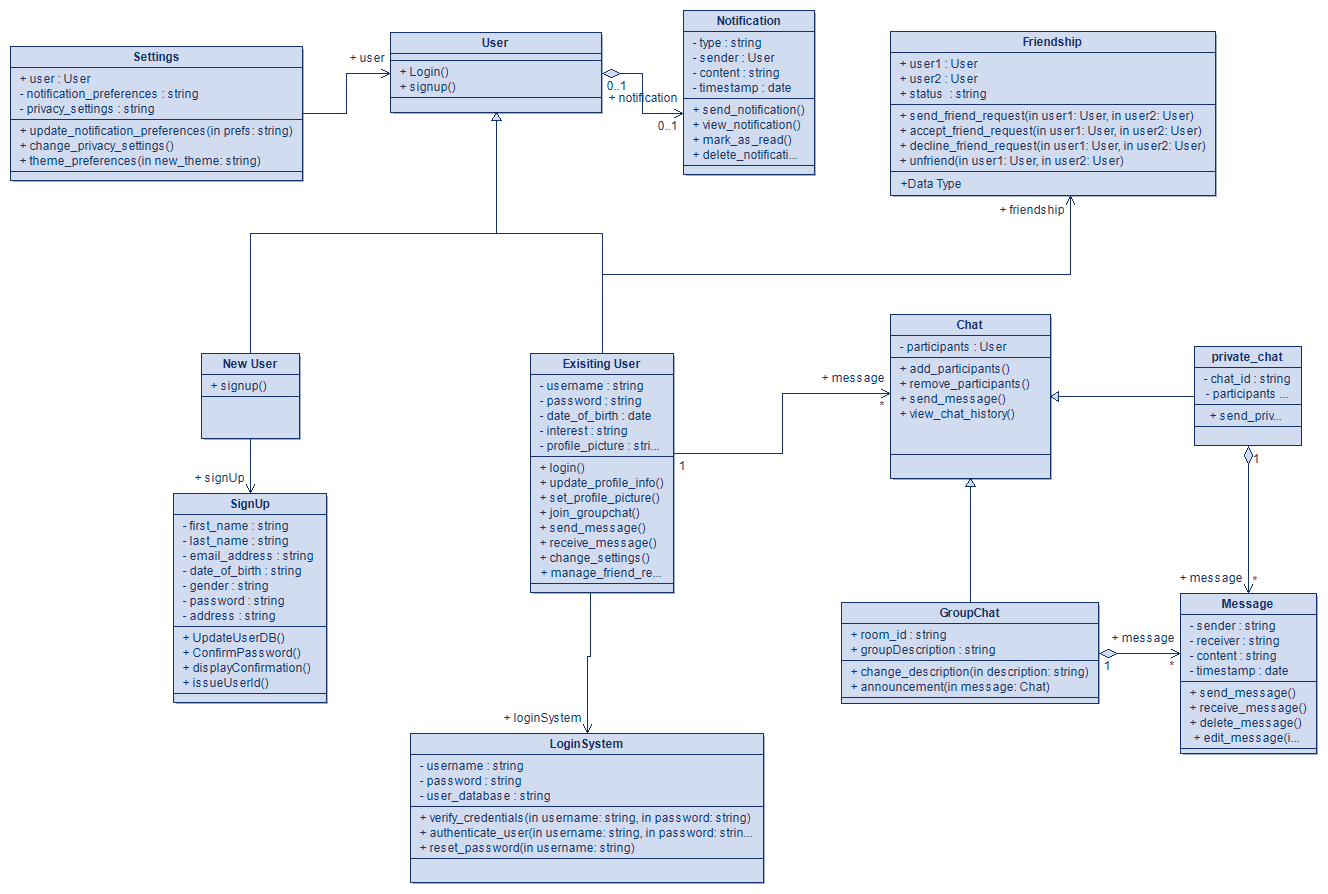
## 4.2 Software design

Use Case Diagram

A diagram of a network

Description automatically generated

# UML Modelling:



UML Class Diagram

**Activity Diagram**

**A screenshot of a computer game

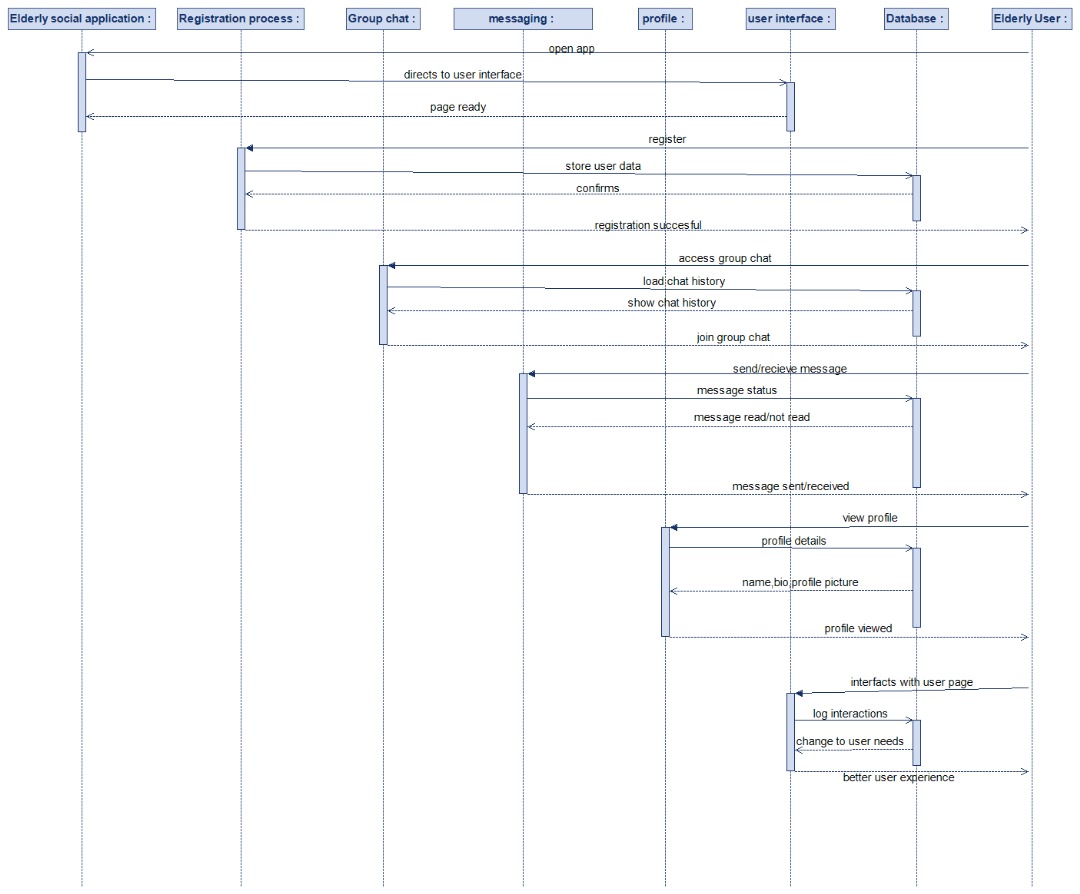
Description automatically generated**

Bpmn diagram

A diagram of a company

Description automatically generated

# Sequence diagram:



# E-R Diagram:

# 

CREATE TABLE Users (

user\_id INT AUTO\_INCREMENT PRIMARY KEY,

username VARCHAR(50) NOT NULL UNIQUE,

email VARCHAR(100) NOT NULL UNIQUE,

password\_hash VARCHAR(255) NOT NULL,

avatar\_url VARCHAR(255),

created\_at TIMESTAMP DEFAULT CURRENT\_TIMESTAMP,

updated\_at TIMESTAMP DEFAULT CURRENT\_TIMESTAMP ON UPDATE CURRENT\_TIMESTAMP

);

CREATE TABLE Admins (

admin\_id INT AUTO\_INCREMENT PRIMARY KEY,

username VARCHAR(50) NOT NULL UNIQUE,

email VARCHAR(100) NOT NULL UNIQUE,

password\_hash VARCHAR(255) NOT NULL,

created\_at TIMESTAMP DEFAULT CURRENT\_TIMESTAMP,

updated\_at TIMESTAMP DEFAULT CURRENT\_TIMESTAMP ON UPDATE CURRENT\_TIMESTAMP

);

CREATE TABLE Groups (

group\_id INT AUTO\_INCREMENT PRIMARY KEY,

group\_name VARCHAR(100) NOT NULL UNIQUE,

description TEXT,

created\_at TIMESTAMP DEFAULT CURRENT\_TIMESTAMP,

updated\_at TIMESTAMP DEFAULT CURRENT\_TIMESTAMP ON UPDATE CURRENT\_TIMESTAMP

);

CREATE TABLE Messages (

message\_id INT AUTO\_INCREMENT PRIMARY KEY,

group\_id INT NOT NULL,

user\_id INT NOT NULL,

content TEXT NOT NULL,

created\_at TIMESTAMP DEFAULT CURRENT\_TIMESTAMP,

FOREIGN KEY (group\_id) REFERENCES Groups(group\_id) ON DELETE CASCADE,

FOREIGN KEY (user\_id) REFERENCES Users(user\_id) ON DELETE CASCADE

);

CREATE TABLE Group\_Members (

group\_id INT NOT NULL,

user\_id INT NOT NULL,

joined\_at TIMESTAMP DEFAULT CURRENT\_TIMESTAMP,

PRIMARY KEY (group\_id, user\_id),

FOREIGN KEY (group\_id) REFERENCES Groups(group\_id) ON DELETE CASCADE,

FOREIGN KEY (user\_id) REFERENCES Users(user\_id) ON DELETE CASCADE

);

CREATE TABLE Admin\_Groups (

admin\_id INT NOT NULL,

group\_id INT NOT NULL,

assigned\_at TIMESTAMP DEFAULT CURRENT\_TIMESTAMP,

PRIMARY KEY (admin\_id, group\_id),

FOREIGN KEY (admin\_id) REFERENCES Admins(admin\_id) ON DELETE CASCADE,

FOREIGN KEY (group\_id) REFERENCES Groups(group\_id) ON DELETE CASCADE

);

# 5. IMPLEMENTATION

## 5.1 Tools, technologies and platforms used

We used HTML(with CSS,REACT and JavaScript) to make the user interface and add styles to make the system user friendly.

We will connect to our database using php and use MySQL as our database

We will create tables for each role of our users i.e. elderly people, groups, administrators we will also makes sure that updating inserting and selecting is much easier in our tables

We will not use an external database for this project

## 5.2 Algorithms

A screenshot of a computer program

Description automatically generated

A screen shot of a computer

Description automatically generated

## 5.3 Standards

For Requirements Analysis: IEEE Standard was adhered to.

For Web Development: We used World Wide Web Consortium which is the international standard for the World Wide Web.

For Security: Advanced Encryption Standard was taken into consideration.

## 5.4 Detailed description of the implementation (coding)

In this section, describe the system you have implemented in detail, with illustrative diagrams, tables, scenarios, etc. Give representative samples of the code you wrote, explaining how it works. Supplement the code with flow diagrams of modules, so that the context in which the code is used becomes obvious.

# 6. TESTING

* Method of testing:
* Functional testing: we verified all features and functions of website i.e. checking links, forms, buttons
* Usability testing : We observed the interactions among with website
* Compatibility We tested weather the website worked on various websites such as google safari
* Unit testing: We tested weather individual components worked

# 7. USER GUIDE OF THE SYSTEM

This is how the user sees the application and then they can create an account and proceed to login after.

A screenshot of a login form

Description automatically generated

Below is a sample user Emmanuel who is creating an account and a success message is shown after

A screenshot of a computer

Description automatically generated

This is what happens after logging into the account :Note still in development

A screenshot of a computer

Description automatically generated

This is how we expect our application to look after its done

A screenshot of a computer

Description automatically generated

Users can search for other users in the application

A screenshot of a computer

Description automatically generated

Sending messages from one user to another

Screens screenshot of a computer

Description automatically generated

# 8. DISCUSSION

The creation of a web application specially designed for elderly people and in particular, the ones who are over 60 years old can be considered as a major improvement as such people often face various issues. Due to aging people and the rise in life span, it becomes important to develop interventions to improve the quality of life of these elderly people, so as to ease their required undertakings. The following aspects are discussed as a part of this discussion: the advantages of the web application proposed in the class, the economic value, the contribution to the environment, the possible social influence, the integration with healthcare systems, and risks.

Benefits and Features Ease of use is an important factor while selecting any product having an online presence, and the WebPT website promises to be highly accessible, if the features mentioned above are anything to go by. User-Friendly Interface: These features are; Basic interface – the application should have a simple user interface to fit the needs of those with little or no exposure to informative technologies. Some of them are; some of the key features that will be implemented include; the provision of larger text, clear icons, as well as use of voice commands. Personalized Assistance: Personalization options like alarms to take medicine or adhere to appointments, as well as other daily tasks, will therefore benefit the elderly to live more independently and happily. Healthcare Integration Remote Monitoring: Heath services interconnection will facilitate the continuous oversight of the figures of health, thus facilitating timely attendance by medical practitioners.

Environmental Contributions such as Reduced Pollution Less Travel: Since the COVID-19 pandemic broke out, the need to travel for treatment, shopping, or attending events was limited, thus making the application to lead to less emissions of gaseous pollutants in the atmosphere, and noise pollution. Energy Conservation: E-services will play a significant role in the conservation of energy since a lot of priorities will not have to be built physically or necessitate a physical journey. Paper Reduction Digital Records: Digitized records will eliminate or lessen the dependent on paper thus enhancing environmental conservation as fewer trees are chopped down to make papers.

Social Interaction: There are ideas of working with social groups to prevent users loneliness and isolate them they will have notification areas for that.

# 9. CONCLUSION

Advantages of the developed web application for elderly users, those are above 60, are the significant in various aspects like economic profit, environmental impact, social interaction, and healthcare sector. It is safe to conclude that addressing the issues of elderly users should improve their quality of life with convenience, security and increased access to service delivery via the application. However, it is pertinent to also evaluate the possible risks or factors that may hinder or complicate the use of the application in fulfilling its intended utility.We implemented a frontend a database and also did some tests on the application

# 10. APPENDIES

Installation of system:

You to install react, and run it in a webserver

# APPENDICES

## B. Code for the system

All code will be in one file