

GoldFolks

System Requirement Specifications (SRS)

# 1 Table of Contents

[1 Table of Contents 1](#_Toc82224938)

[1. Problem Statement 2](#_Toc82224939)

[2. Overview 2](#_Toc82224940)

[2.1. Background 2](#_Toc82224941)

[2.2. Overall Description 2](#_Toc82224942)

[3. Investigation & Analysis Methodology 2](#_Toc82224943)

[3.1. System Investigation 2](#_Toc82224944)

[3.2. Analysis Methodology 3](#_Toc82224945)

[3.2.1. Feasibility study and requirements elicitation 3](#_Toc82224946)

[3.2.2. System analysis and requirements specification 3](#_Toc82224947)

[3.2.3. Object-oriented design using UML 3](#_Toc82224948)

[3.2.4. Prototyping 5](#_Toc82224949)

[4. Constraints 6](#_Toc82224950)

[4.1. Scalability 6](#_Toc82224951)

[4.2. Latencies 6](#_Toc82224952)

[4.3. Network Issues 6](#_Toc82224953)

[4.4. Updates 6](#_Toc82224954)

[4.5. Memory Footprint 6](#_Toc82224955)

[4.6. Project Schedule 6](#_Toc82224956)

[5. Operational Requirements 7](#_Toc82224957)

[5.1. Help Desk Support 7](#_Toc82224958)

[5.2. Application Services and Technical support 7](#_Toc82224959)

[5.3. Administration Features 7](#_Toc82224960)

[5.4. System hardware fail over and routine back up 7](#_Toc82224961)

[5.5. Audit Trail 7](#_Toc82224962)

[6. Functional Requirements 7](#_Toc82224963)

[7. Input Requirements 12](#_Toc82224964)

[7.1. User Information 12](#_Toc82224965)

[7.2. Medication Details 12](#_Toc82224966)

[7.3. Game Answers 12](#_Toc82224967)

[8. Process Requirements 13](#_Toc82224968)

[8.1. Database transaction 13](#_Toc82224969)

[8.2. Data integrity 13](#_Toc82224970)

[8.3. Data validation 13](#_Toc82224971)

[8.4. Performance 13](#_Toc82224972)

[8.5. Data repository 13](#_Toc82224973)

[9. Output Requirements 13](#_Toc82224974)

[9.1. User Information 13](#_Toc82224975)

[When the user creates an account or changes their password, the information should be stored in the cloud database. 13](#_Toc82224976)

[9.2. Medication Details 13](#_Toc82224977)

[9.3. Mental Math & Simon Says Game Results 13](#_Toc82224978)

[10. Hardware Requirements 14](#_Toc82224979)

[10.1. Network 14](#_Toc82224980)

[10.2. Client Devices 14](#_Toc82224981)

[10.3. Server 14](#_Toc82224982)

[10.4. Production support systems 14](#_Toc82224983)

[11. Software Requirements 14](#_Toc82224984)

[11.1. Client Operating Systems 14](#_Toc82224985)

[11.2. Client Application 14](#_Toc82224986)

[11.3. Network System 14](#_Toc82224987)

[11.4. Mainframe System 14](#_Toc82224988)

[11.5. Licenses 15](#_Toc82224989)

|  |  |  |
| --- | --- | --- |
|  | | Client Application 8 |
|  | | Network system 8 |
|  | | Mainframe system 9 |
|  | | Licenses 9 |
| 13. |  | Deployment Requirements 9 |

# Problem Statement

Aging populations are a problem faced by many countries. In addition, the multitude of health problems coupled with the inability to adhere to medication instructions warrant constant care and support for the elderly, which costs time and money. Although there are mitigating measures such as nursing homes and family support, these measures often proved to be financially infeasible or insufficient. As such, there must be a way to help elderly people monitor their own health and take care of themselves better in the comfort of their own homes. This will reduce emotional and financial burden on the elderly as well as their loved ones.

# Overview

## Background

As the proportion of elderly people in Singapore rises over time, the strain on resources required to care for them increases too. This is mainly due to Singapore’s poor birth rate, which results in a lower population of young compared to old. Additionally, more young people are focusing on their careers; many of them lack time to care for their elderly parents/relatives. Also, many elderly people fail to take their medication properly due to reasons like memory loss, poor vision, and social isolation. This only exacerbates their health problems, which might strain resources even more as more care and support are warranted. Nursing/retirement homes are expensive solutions and may give rise to more problems like depression among residents.

With the growing need for more resources to care for the elderly and the current issues faced by the elderly, their loved ones, and society overall, there is an urgent need to develop a means to allow the elderly to take charge of their own health.

## Overall Description

GoldFolks is a mobile application that will be developed using Flutter. The targeted users are elderly individuals, although the general public can use the app as well. By using GoldFolks, users can take charge of their own health at home.

The main system comprises of three core functionalities: Medication Reminders, Cognitive Games to improve and maintain cognitive ability, and Exercise Videos to help the elderly stay in good health.

# Investigation & Analysis Methodology

## System Investigation

The mobile application allows users to log in by matching the entered username and password combination with an existing record in the database that stores account information. When first-time users create an account, a new record will be created and added to that database. Upon successful entry of login details, appropriate feedback will be sent to inform users of successful login, subsequently allowing them access to the features of the app.

Exercise videos, game high scores, and medication reminders will also be stored on the cloud database. Whenever users achieve new high scores or create new medication reminders, a corresponding record will be created in the database.

## Analysis Methodology

### Feasibility study and requirements elicitation

Organise a development and implementation team comprising people who are knowledgeable about the development of mobile applications and have had experiences caring for the elderly or are at least familiar with the problems they face. Regular meetings will be held with this group of individuals. Interviews will also be conducted with elderly persons and nursing/retirement home staff to further understand user requirements. Developers of similar health-monitoring applications will be contacted and consulted to help define the system requirements and potential challenges that might arise during development. A Feasibility and Risk Assessment study will be conducted to determine the appropriateness of the solution(s).

### System analysis and requirements specification

* + - 1. Perform an analysis of the problem using object-oriented techniques

An external view of the application will be developed using Unified Modeling Language (UML). This System Requirement Specifications documents will form part of the documentation for the project. Some desired features of the application include:

* + - * 1. Allow users to login and register to track reminders and game scores between devices/installations.
        2. Allow users to reset their passwords to regain access to their accounts.
        3. Allow users to set reminders to take medication.
        4. Provide a list of simple physical exercises and accompanying instructional videos.
        5. Allow users to play games that strengthen their cognitive abilities and track their high scores.
      1. Scope and Limitations

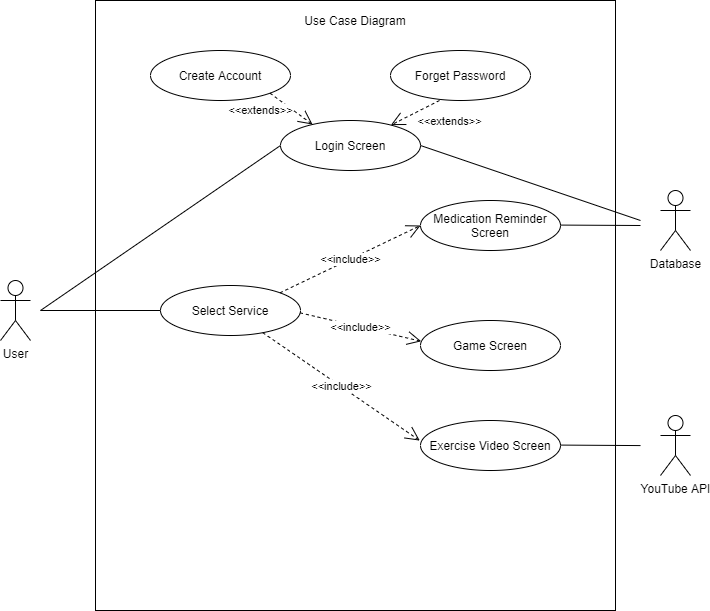
Analysis methodology will involve business analysis, requirement analysis, data analysis, process analysis, and application architecture:

* + - * 1. Business analysis – State the business rules, business system interfaces, business function, business ownership, sponsorship and associated project budget requirement
        2. Requirement analysis – System I/O description, user requirement definition, functional and security requirement
        3. Data analysis Involve data collection process, data validation, data storage, manipulation and retrieval
        4. Process analysis – Data/process flow analysis, process decomposition and system interfaces
        5. Application architecture – Analyze application information structure, usability, user interface design, interaction and application implementation.

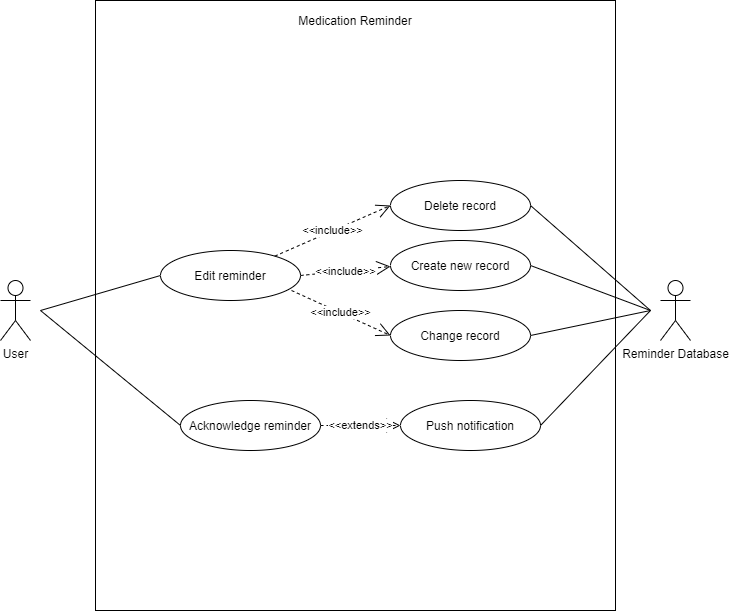
### Object-oriented design using UML

A detailed object-oriented design for the registration system will be developed. UML will be used again for the graphical representation and documentation of the design. The system will primarily concern itself with the storage of reminders and high scores from the games. A user will open the application and be allowed to set medication reminders, play games, or watch instructional exercise videos.

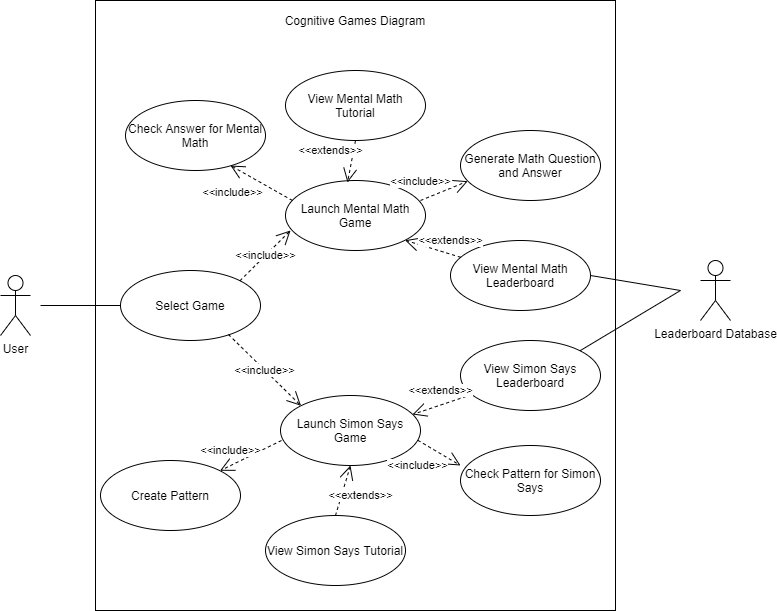
* + - 1. Overall Use Case Diagram



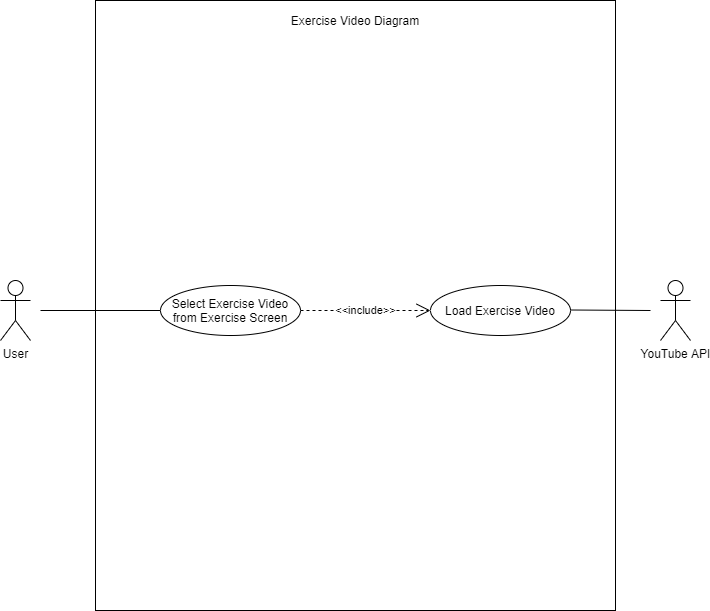
* + - 1. Medication Reminder Use Case Diagram



* + - 1. Cognitive Game Use Case Diagram



* + - 1. Exercise Video Use Case Diagram



### Prototyping

Object-Oriented Rapid Prototyping (OORP) method will be used to implement a limited and functional prototype of the application. The prototype will be a working example of part of the system for demonstration and proof of concept purposes only. Specifically, it will be a skeletal version of the final product, mainly used to demonstrate how users would interact with the user interface. The prototype will be presented to the implementation team.

# Constraints

## Scalability

Mobile devices have varying screen sizes. This may affect the UI appearance. Hence, the application needs to be designed for use on a range of different mobile devices.

Devices may also have differing specifications. The application must be able to run on both low-end and high-end devices.

## Latencies

In all mobile applications, there are load times. When an app is used for the first time, there is typically a lot of background setup that needs to be done. The application should thus be able to responsive to user input to reduce frustration with the app.

## Network Issues

Access to the Internet is necessary for loading videos or updating/viewing game high scores. Users will also need to log in to their account on a new device to import saved preferences for medication reminders into their local database.

## Updates

Users may need to update the mobile application manually. The primary target audience is elderly people. They most likely are not familiar as to how to update the application. A solution could be to prompt users, upon opening the app, with a message to update the app. In the message, there should be a button that takes users automatically to the app store to update the application.

## Memory Footprint

Memory is not unlimited. When memory usage is high, the operating system of the mobile device may shut down the application or sacrifice cached data, slowing program execution. Thus, the application should aim to minimise memory usage and cache size.

## Project Schedule

There is a 3-month deadline for the planning, development, implementation and quality assurance of the project.

# Operational Requirements

## Help Desk Support

Users have access to email assistance for questions that are technical in nature, such as, slow, or sluggish system response time, incompatible mobile phone features, application errors, system downtime inquiries, account lock out assistance etc. The email channel can also be used for other non-related enquires such as feature requests and application support. Email tickets will be responded within 2-3 business days.

## Application Services and Technical support

Our help desk support system will send email notifications to our application developers or quality assurance team depending on the nature of the new tickets in the help desk support system. The respective team will then follow up accordingly with the required actions to resolve the tickets.

## Administration Features

System security and access levels are provided in the online system. There are varying levels of system access and functional authority. Each user’s access is limited to his/her own medication reminders and games records. Only the authorized system administrator(s) has access to all user records. Only the admin has access to update or modify the exercise videos and games system.

## System hardware fail over and routine back up

Computer operations center will handle system hardware tasks such as data tape back-up, hardware maintenance, fail over, scheduled system patches and maintenance.

## Audit Trail

System audit trails are inherent part of all user’s registrations. In addition, as the application may contain sensitive health information of our users due to their medication, all transactions will be tracked and transaction records will capture what action was taken, when (time-stamp) the transaction occurred and who made the transaction.

# Functional Requirements

GoldFolks is “self-service style” system that provides an all-encompassing solution to healthy aging. The system shall provide an intuitive and user-friendly planner for the elderly using the app. The functional requirements for the application are grouped by the system features. Each major feature is broken down into smaller features for better atomicity, clarity, and verifiability.

* 1. User Authentication
     1. **Description and Priority**

The first time the user opens the app, the system should prompt the user to sign up or log in. Subsequent uses of the app will remember the user’s login details.

Priority: **High**

This functionality will be upon every app installation and without authentication the user is unable to access the rest of the application functions

* + 1. **Stimulus/Responses Sequence**

6.1.2.1. The user will first open the application.

6.1.2.2. The user will then proceed to login or sign-up using email.

* + 1. Functional Requirement
       1. The user must be able to access the registration page and the forget password page from the login page (default page).
       2. The system must prompt the user for his/her credentials during login
          1. The system must prompt the user for his/her email address.
          2. The system must prompt the user for his/her password.
       3. The system must validate that all fields are filled in.
          1. If any of the fields are not filled, a corresponding error message will be displayed for an error field.
          2. The system should be able check the validity of the fields entered by the user.
          3. The system must validate the user’s password and email address exists and is valid.
          4. The system must display the corresponding error message when any of above validation fails.
          5. The system must redirect the user to the home screen upon successful login.
       4. The user must tap on the ‘Create Account’ button to sign up for a new account.
       5. The user must be able to register for an account with the system if they have not before.
       6. The system must prompt the user for his/her credentials during registration.
          1. The system must prompt the user for his/her name.
          2. The system must prompt the user for his/her email.
          3. The system must prompt the user for his/her password.

The system must ensure that the password is complex enough to deter brute force attacks (e.g., 8-characters alphanumeric password)

* + - 1. The system must be able to check the validity of the user account.
         1. The system must display an error message when students try to register using a previously registered account.
         2. The system shall display an error message if the password does not meet the password criterion set.
         3. The system must notify the user upon successful account creation.

The system must display a message on the application to inform the user of successful account registration.

* + - 1. Once an account has been created, the system should re-direct the user back to the login page.
      2. The user must be able to logout from the system if they have logged in.
         1. The system shall redirect the user to the login page upon successful log out.
      3. User’s must be able to tap on the ‘Forget Password’ button to get a new password via email.
         1. The system must prompt the user to enter his/her email address.
         2. The system shall display a corresponding error message if the text entered does not match a valid email format.
         3. The system must validate that the email address is valid.
         4. The system shall send an email providing a link to change the password.
         5. The user must enter the new password through the link.
         6. The system must save the newly created password to the database.
         7. The user can continue to sign in with the new password.
  1. Home Screen
     1. **Description and Priority**

This is the first screen that the user will see every time he/she is logged into the app. The application must have 3 main features, Medication Reminder, Cognitive Games and Exercise video on the main menu.

Priority: **High**

The user will use this feature every time the app is launched. It allows users to gain access to the main feature of the app

* + 1. **Stimulus/Response sequence**

1. The user will have gone through feature 7.1, the user authentication and is logged in.
2. Once the user finishes feature 7.1, this feature would be triggered.
   * 1. **Functional Requirement**
        1. The system must display the 3 main functions/buttons – Medication Reminder, Cognitive Games and Exercise Video in the home screen in a list view.
        2. The system must allow the user to navigate to the respective screens which clicking of the respective buttons.
           1. In the first function, which is the ‘Medication Reminder’, the system shall allow users to view and edit their reminders.

The Medication Reminder page shall be loaded when the user taps the Medication Reminder button.

* + - * 1. In the second function, which is the ‘Cognitive Games’, the system shall allow users to select the game they would like to play.

The Cognitive Game page shall be loaded when the user taps the Cognitive Games button.

* + - * 1. In the third function, which is the ‘Exercise videos’, the system shall allow users to select and watch the selected videos.

The Exercise Game page shall be loaded when the user taps the Exercise video button.

* 1. **Medication Reminder Mode**
     1. **Description and Priority**

The user can tap on the edit reminder button to edit the timing and frequency of the reminder. The user can tap on the create reminders button to create new reminders. Lastly, the user taps the delete reminder button to delete existing reminders. The user will receive notification on the reminder time set by the user.

Priority: **High**

The user will use this feature only when he/she wants to edit his/her reminder.

* + 1. **Stimulus/Response sequence**

1. The user will have gone through feature 7.2, the home screen.
2. The user must be on the home screen.
3. The user taps on the Medication Reminder button to trigger this feature.
   * 1. **Functional Requirement**
        1. The system shall allow the user to create a new medication reminder.
           1. The app must display a create reminder button.

The user must be allowed to input details about when the reminder should be.

The user must be allowed to input details about what medication should be taken.

The user must be able to save the new medication reminder.

The system shall update the database with the newly created medication reminder.

* + - 1. The system shall display a list of medication reminders that have been created.
         1. The system shall allow the user to edit an existing medication reminder.

The user shall be able to click into one of the existing reminders among the list of medication reminders.

The user shall be able to click the edit button upon entering the existing medication reminder.

The user shall be able to change the description of the medication to be taken.

The user shall be able to change the timings of the existing reminder.

The system shall save the changes to the database when the user taps the save button.

* + - * 1. The system shall allow the user to delete an existing medication reminder.

The user shall be able to delete the reminder by pressing on the delete button.

The system must remove the existing medication reminder from the database upon tapping the delete button.

The user will be notified when the current time is the same as the medication reminder set time.

The notification will appear in the form of local notification on the top bar of the user’s device.

* 1. **Cognitive Games Mode**
     1. **Description and Priority**

The user must be able to view the games available to be played in a compact list view.

Priority: Medium

* + 1. **Stimulus/Response Sequence**
       1. The user is in Feature 7.2.
       2. The user taps on the “Cognitive Games” button.
       3. The user triggers this feature.
    2. **Functional Requirement**
       1. The system must have 2 main games- Mental Math and Simon Says.
          1. The game selection screen should list the two games button in a vertical list view.
       2. The system shall allow the user to choose the game they would like to play.
          1. When the user clicks on the Mental Math game button, the system must bring the user to the Mental Math game screen to trigger Feature 7.5.
          2. When the user clicks on the Simon Says game button, the system must bring the user to the Simon Says game screen to trigger Feature 7.6.
  1. **Mental Math Game Screen**
     1. The Mental Math Game Screen must display the 3 main buttons – Play, Tutorial and View leaderboard.
     2. User must be able to leave the page before starting the game.
        1. The system shall bring the user back to the cognitive games page when the ‘back’ button is triggered by the user.
     3. User must be able to tap the ‘Play’ button to trigger the start of the game.
        1. Upon tapping the play button, the math question should flash on the screen.
        2. The system must display a timer of 1 minute.
        3. User must be given the appropriate UI to answer the question (e.g., MCQ).
        4. User must be able to select one of the MCQ choice.
        5. User must be able to submit the question by tapping the ‘Next’ button.
        6. The system must mark the user’s answers and generate new questions to be displayed when the user presses the ‘Next’ button.
        7. The system must end the game when 1 minute has elapsed.
        8. The payer must be able to see how many questions they have gotten right at the end of the game.
        9. The player must be able to leave the game halfway through by tapping the ‘back’ button.
           1. The user must be able to confirm they would like to exit the game.
     4. User must be able to tap the ‘Tutorial’ button to view the rules the game.
        1. The system shall display a series of steps detailing how to play the game.
     5. User must be able to tap the ‘View Leaderboard’ button to view the view the leaderboard.
        1. The system shall display the top 10 players for the game.
        2. The system shall display the user’s current rank for the game.
  2. **Simon Says Game Screen**
     1. **Description and Priority**

This feature describes the gameplay of the Simon Says Game. Upon entering this screen, the user is able to tap on the tutorial button to learn about how the game is played and/or tap the play to start the game. The user can also view the leaderboard.

Priority: **Medium**

This feature is only used when the user wants to play the Simon Says game.

* + 1. **Stimulus/Response sequence**
       1. The user is in Feature 7.4.
       2. The user taps on the “Simon Says” game button.
       3. The user triggers this feature.
    2. **Functional Requirement**
       1. The Simon Says Game Screen must display the 3 main buttons – Play, Tutorial and View leaderboard.
       2. User must be able to leave the page before starting the game.
          1. The system shall bring the user back to the cognitive games page when the ‘back’ button is triggered by the user.
       3. User must be able to tap the ‘Play’ button to trigger the start of the game.
          1. Upon tapping the ‘play’ button, the system must display a series of clicked boxes as a pattern.
          2. User must be given the appropriate UI (e.g 3x3 grids) to tap on the boxes that follows the pattern that was shown.
          3. The system must reuse the same pattern and add one more pressed box to the pattern every time the user gets the pattern correctly.
          4. The system must end the game once the user presses a box in the wrong order.
          5. The player must be able to leave the game halfway through by tapping the back button.

The user must be able to confirm they would like to exit the game.

The system shall redirect the user back to the Simon Says Game Screen upon successful confirmation.

* + - 1. User must be able to tap the ‘Tutorial’ button to view the rules the game.
         1. The system shall display a series of steps detailing how to play the game.
      2. User must be able to tap the ‘View Leaderboard’ button to view the view the leaderboard.
         1. The system shall display the top 10 players for the game.
         2. The system shall display the user’s current rank for the game.
  1. **Exercise Video Screen**
     1. **Description and Priority**

This feature allows the user to select the exercise video that they would like to watch and watch the selected video in the app.

* + 1. **Stimulus/Response sequence**
       1. The user is in Feature 7.2.
       2. The user taps on the “Exercise Video” button.
       3. The user triggers this feature.
    2. **Functional Requirement**
       1. The system must display the exercise videos in a grid view.
       2. The system must load the video once the user taps on the video they would like to play.
       3. The system must allow the user to stop, pause and restart the video. User must be able to tap the ‘Back’ button to exit the video player, which will bring them back to the grid view of exercise videos.

# Input Requirements

## User Information

During account registration, users must input his/her information like name, email address and password. The email address

and password will help facilitate the user to login to the system with the full functionality – Medication reminders, Cognitive Games and Exercise video. A unique identifier will be issued to each user upon registration of the system.

## Medication Details

For returning users, upon logging in, the users will not need to input their medication reminders details again unless they wish to modify their reminders. For first time users, upon logging in, they will be required to key in their medication reminders details if they require the use of this feature.

## Game Answers

While playing Simon Says or Mental Math, the user’s selection will be recorded and matched with the corresponding correct answer in the database. These records will be used for the final tabulation of scores.

# Process Requirements

The following are among the inherent requirements that the online registration system must be able to handle.

## Database transaction

The system must be able to send, receive and trigger transactions to the local database and online Firebase system.

## Data integrity

All processes must function as expected by the user without errors and the data processed and stored must be accurate. Avoiding committing to the database corrupted or incomplete records.

## Data validation

For the user to access the system functionality, user’s login credentials must be validated before allowing access to the system.

## Performance

Must resolve locking issues and handle concurrent use of the system on a 24x7 basis. Send, receive and display user messages to assist the over-all user experience. Must be able to add/update/delete data within 5 seconds.

## Data repository

The GoldFolks system will maintain the existing Firebase database as the main repository of data.

# Output Requirements

## User Information

## When the user creates an account or changes their password, the information should be stored in the cloud database.

## Medication Details

For the medication reminders, the user will be presented with a button to press if they have taken the medication as stated.

## Mental Math & Simon Says Game Results

After playing the game each time, the user will be presented with their score for the game, overall high score and the leaderboard of other players in the game. They will be able to compare their score with the leaderboard to determine where they stand if they are of a competitive nature.

# Hardware Requirements

## Network

Mobile users will need to have an internet connection in order to log in. Thus, either cellular data or Wi-Fi connection is required for login.

## Client Devices

Client devices must be either running on Android or iOS systems.

## Server

24/7 online server on Google Play Store and Apple App Store, with a backend database.

## Production support systems

Android and Apple mobile phones for testing. Redundant drives for storage of application as back up.

# Software Requirements

# Client Operating Systems

* iOS (excluding PC Macbook)
* Android

# Client Application

Mobile phones/tablets with compatible OS version:

* iOS (excluding PC Macbook)
* Windows

Alternatively, an emulator for Windows/Mac PC.

# Network System

Network software and protocols in order for systems to communicate:

* TCP/IP
* HTTP
* HTTPS

# Mainframe System

* Firebase

# Licenses

Valid licenses are required to run software from third party vendors:

1. To use application development tools (Google Play Developer and Apple Developer).
2. To use web server, application server and database software in development, test and production mode.
3. Deployment Requirements

