Software Requirements Specification

for

Dementia-App

**Version 1.0 approved**

**Prepared by**

**100052234 Chhaya Sok  
102079222 Alexey Maksimets**

**Swinburne**

**30.09.2019**

**Table of Contents**

**Table of Contents [ii](#_heading=h.gjdgxs)**

**Revision History [ii](#_heading=h.30j0zll)**

Part 1

**1. Introduction [1](#_heading=h.1fob9te)**

1.1 Purpose [1](#_heading=h.3znysh7)

1.2 Document Conventions [1](#_heading=h.1y810tw)

1.3 Intended Audience and Reading Suggestions [1](#_heading=h.4i7ojhp)

1.4 Product Scope [1](#_heading=h.tyjcwt)

1.5 References [1](#_heading=h.3dy6vkm)

**2. Overall Description [2](#_heading=h.1t3h5sf)**

2.1 Product Perspective [2](#_heading=h.4d34og8)

2.2 Product Functions [2](#_heading=h.2s8eyo1)

2.3 User Classes and Characteristics [2](#_heading=h.17dp8vu)

2.4 Operating Environment [2](#_heading=h.2xcytpi)

2.5 Design and Implementation Constraints [2](#_heading=h.3rdcrjn)

2.6 User Documentation [2](#_heading=h.1ci93xb)

2.7 Assumptions and Dependencies [3](#_heading=h.3whwml4)

Part 2

**2. External Interface Requirements [3](#_heading=h.26in1rg)**

3.1 User Interfaces [3](#_heading=h.lnxbz9)

3.2 Hardware Interfaces [3](#_heading=h.2bn6wsx)

3.3 Software Interfaces [3](#_heading=h.qsh70q)

3.4 Communications Interfaces [3](#_heading=h.3as4poj)

**4. System Features [4](#_heading=h.35nkun2)**

4.1 System Feature 1 [4](#_heading=h.1ksv4uv)

4.2 System Feature 2 (and so on) [4](#_heading=h.44sinio)

**5. Other Nonfunctional Requirements [4](#_heading=h.2jxsxqh)**

5.1 Performance Requirements [4](#_heading=h.1pxezwc)

5.2 Safety Requirements [5](#_heading=h.49x2ik5)

5.3 Security Requirements [5](#_heading=h.z337ya)

5.4 Software Quality Attributes [5](#_heading=h.2p2csry)

5.5 Business Rules [5](#_heading=h.147n2zr)

**6. Other Requirements [5](#_heading=h.3o7alnk)**

**Appendix A: Analysis Models [5](#_heading=h.3j2qqm3)**

**Revision History**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Date** | **Reason For Changes** | **Version** |
| Dementia-app | 30.09.19 | Initial file | 1.0 |

# Introduction

## Purpose

The purpose of this document is to present the Dementia-App software.   
Moreover, the repost will present the purpose and the feature of the software, the User Interface of the software. And how the software will do to help it user. This report work as a guide for it user and to make the user in better understanding on how the software works.

## Product Scope

Dementia Tool is a software that provide different and useful tool for its users. Its potential user can use the app to remind them of any task that may or may not remind them at a certain time or a specific location. The app also works as a location tracker on which will allow the user to know where they’re currently located.

## References

[IEEE Template for System Requirement Specification Documents](https://goo.gl/nsUFwy)

# Overall Description

## Product Perspective

Dementia app is specifically developed for everyone who has dementia diagnosis or for their caregiver. The sole purpose of this app is to make taking care of their patient easier. The app has many different tools such as keeping track of their everyday activity and reminding them of any tasks that may need to be done.

It is a phone app where user can download it from App Store (IOS) or Play Store (android).

## Product Functions

* Download App from specific phone store. Exp: App store or Play store.
* Click on the app where it has been installed
* Open Reminder: Check Reminder
* Open Reminder: Create Reminder
* Open Reminder: Choose Date/Time
* Close Reminder
* Open Activity: Check Activity
* Close Activity
* Open Location: Check Current location
* Close Location
* Open Contact: Check Contact
* Open Contact: Edit/Add Contact
* Close Contact
* Close App: Click home screen for iOS or back for Android

## User Classes and Characteristics

* PwD (patient with dementia) - use to remind them of certain tasks, contact emergency services or contact their caregiver.
* Caregiver/relative - can keep track of their patient, location and of their current activity

## Design and Implementation Constraints

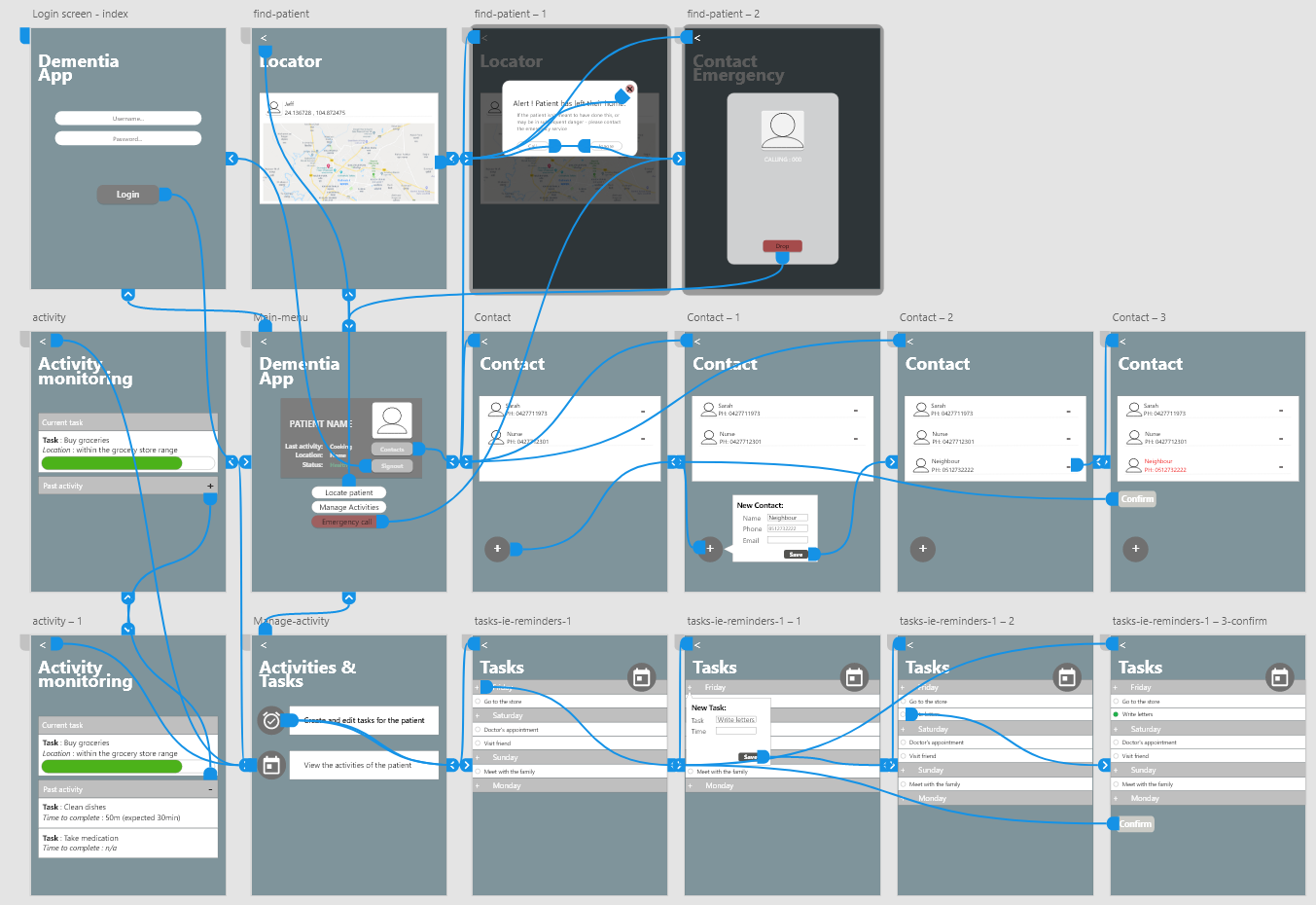
The app is developed using a certain programming software called Visual Studio. Some feature may not be available to user as it may following under a regulatory policy that may restraint the app to fully functional.

# External Interface Requirements

## User Interfaces

*<Describe the logical characteristics of each interface between the software product and the users. This may include sample screen images, any GUI standards or product family style guides that are to be followed, screen layout constraints, standard buttons and functions (e.g., help) that will appear on every screen, keyboard shortcuts, error message display standards, and so on. Define the software components for which a user interface is needed. Details of the user interface design should be documented in a separate user interface specification.>*

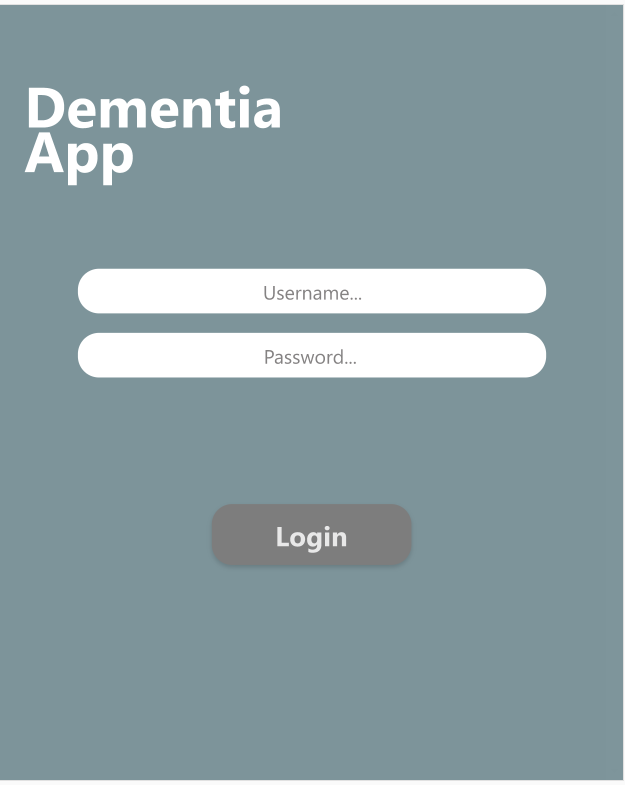
Connections between individual screens (software)



GUI Standards:

* Notifications – The user will be reminded of any updates regarding the status/whereabout of the PwD.

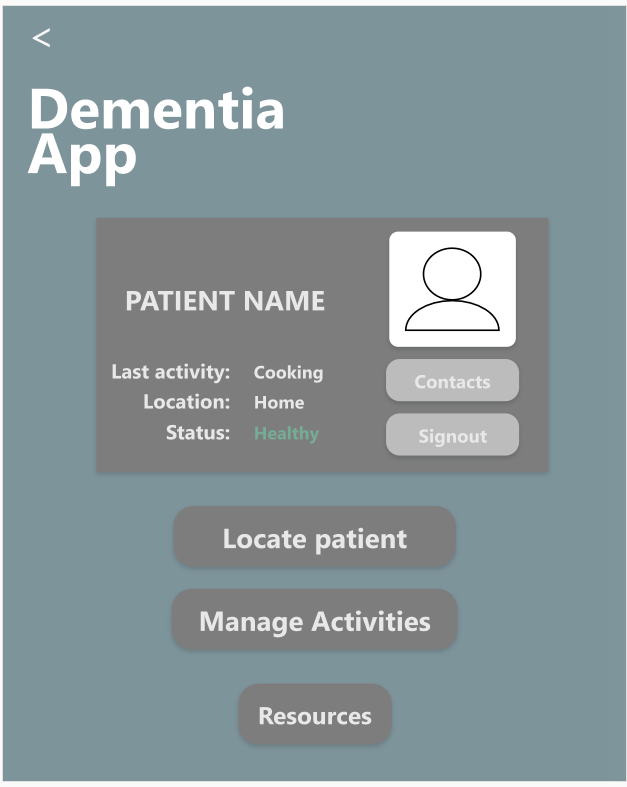
1. Dementia-App’s login screen



1. Home-page overview (for Dementia-App).  
   Top right button is the navigation allowing the user to quickly sign out or navigate to the previously selected window.

Main content of the page [from top, left to bottom] is the general overview of the patient’s status [including] : the current task, location and the status. The right-most side shows placeholder (where the user image would be), and two (non functional requirements) buttons which (in the case of the top button), redirect the user to a page for managing the contacts (which the PwD has access to incase of emergency), aswell as the signout button.

Bellow , are the main (hence they are drawn larger then other elemnts (buttons) to show their importance) buttons, which provide the user with the functionality required to achiev their task.



# System Features

*This section is dedicated to the features of the Dementia-app, their explanation and how they are being used to achieve the target user requirement.*

## Remote location monitoring

### Description and Priority

REQ-1: remote server connection

Users can directly monitor and report on the location of the patient over an internet connection. Since this is the ultimate point of the product it of HIGH priority, cost associated with hosting a p2p (peer to peer) connection server is marginal as well as the cost of GPS tracker and or a mobile application for a smartphone.  
This feature is also the more challenging feature to implement and carries a medium to high risk ; 6-7

4.1.2 Functional Requirements

Software needs a connection to the remote device via a WIFI connection, this connection, is established via a p2p server. The server is being sent a feed from the GPS sensor that declared the location (coordinates) of the PwD and this information is relayed to the caregiver. [Primary] anticipated errors/breaking points are the points at which the PwD cannot connect to the remote server, this is due to either a roaming/WIFI connection issue or to an issue with the server itself – the client side connection dropping out due to 3G connection shouldn’t be an issue since any action that resembles the PwD being lost or in danger (i.e. the device losing battery or connection) then the medical services will be contacted and their last known location used as a point of reference when finding said patient.

Appendix

For further testing we must evaluate the response time including the time needed for said signal to be sent (and received) via the server.

## Remote activity monitoring

REQ-2: remote activity monitoring

4.2.1 Description and Priority

Remote activity is a function that will allow the user to view and record the PwD’s activities, these activities will be recorded via server connection (view 4.1 – remote server connection) which receives sensor data in updates from the sensors attached to the PwD’s self, all the while they are in a remote location (in relation to the primary user).   
Cost is marginal since the implementation of a remote monitoring server already entails the capability to record and store data on it, meaning the server could be used for multiple things while the only significant investment being the initial server/server side solution purchase. Risk of establishing a server and device software that records and stores distinctively different types of data is high, since this is very error prone and could require several iterations before the data recording is accurate, rate the risk at 8.

4.2.2 Functional Requirements

Activity is the instance of a sensor recording –

*<Each requirement should be uniquely identified with a sequence number or a meaningful tag of some kind.>*

## Task management

REQ-3: creation and management of ‘remote-user’ tasks

4.3.1 Description and Priority

*<Provide a short description of the feature and indicate whether it is of High, Medium, or Low priority. You could also include specific priority component ratings, such as benefit, penalty, cost, and risk (each rated on a relative scale from a low of 1 to a high of 9).>*

4.3.2 Functional Requirements

*<Itemize the detailed functional requirements associated with this feature. These are the software capabilities that must be present in order for the user to carry out the services provided by the feature, or to execute the use case. Include how the product should respond to anticipated error conditions or invalid inputs. Requirements should be concise, complete, unambiguous, verifiable, and necessary. Use “TBD” as a placeholder to indicate when necessary information is not yet available.>*

*<Each requirement should be uniquely identified with a sequence number or a meaningful tag of some kind.>*

## Location notifications

REQ-3: creation and management of ‘remote-user’ tasks,

upon PwD reaching an unknown area, leaving home or due to an emergency

4.3.1 Description and Priority

*<Provide a short description of the feature and indicate whether it is of High, Medium, or Low priority. You could also include specific priority component ratings, such as benefit, penalty, cost, and risk (each rated on a relative scale from a low of 1 to a high of 9).>*

4.3.2 Functional Requirements

*<Itemize the detailed functional requirements associated with this feature. These are the software capabilities that must be present in order for the user to carry out the services provided by the feature, or to execute the use case. Include how the product should respond to anticipated error conditions or invalid inputs. Requirements should be concise, complete, unambiguous, verifiable, and necessary. Use “TBD” as a placeholder to indicate when necessary information is not yet available.>*

*<Each requirement should be uniquely identified with a sequence number or a meaningful tag of some kind.>*

# Other Nonfunctional Requirements

## Security Requirements

To ensure the users of Dementia app have secure connection to the patient - the app provides the users with a set of privacy measurements to make sure that no unauthorized person can access these services (in the form of server side user authentication).

## User-authentication

REQ-4: primary-user-side server connection and authentication

4.4.1 Description and Priority

High priority feature

sensitive data being transferred and the remote-access within the application the solution must provide a secure solution to authenticate users and prevent any unauthorized 3rd party to access data meant for the caregiver. Logical restriction to help secure the access to the server could be the tracking of IPs and allowing users to block any connections to the server made outside the IP (since the location of client’s device is static)

4.4.2 Functional Requirements

The software must connect via internet to the server but not before authenticating the client side user as being the caregiver - this is established with two factor authentication (2FA) which holds a private and public key (former being known to the user only) to establish a connection between the server and client side application.

**Appendix A: Analysis Models**

*<Optionally, include any pertinent analysis models, such as data flow diagrams, class diagrams, state-transition diagrams, or entity-relationship diagrams*.> **[refer to previous tasks]**