Project Requirements

This document details the initial Functional, non-Functional and Domain requirements that we have identified from the extensive research that we have carried out. These requirements are likely to change over the course of the project as we look to constantly engage our target audience to ensure that we are effectively implementing these. As we do this, we predict that the customers will identify new requirements and feel that some requirements are not as important as they first thought.

Below the list of Functional, Non-Functional and domain requirements we have prioritised the requirements using the MoSCoW method. This will allow us to easily identify the criticality of requirements and will also clearly indicate to us whether the project has been successful.

# Functional Requirements

These are requirements that determine how the system should behave and how it should react to particular inputs.

# Non-Functional Requirements

These are the requirements that specify how the web and mobile applications should perform and the qualities that it should adhere too.

*e.g. Speed, Size, Ease of Use, Reliability, Robustness, Portability*

# Domain Requirements

These requirements reflect the constraints of the environment that the application will be developed in.

*e.g. legal, php, html, java, hardware limitations*

Terms and conditions accepting (registration)

Explicit statement accepting (registration)

Tick box to acknowledge that data provided is accurate and that it is the responsibility of the user to keep this information up-to-date and is not the responsibility of JustHealth (registration)

Total deletion of all user details from database/tick to agree to keep details in case of reactivation (account delete)

Any modifications to the database need to be added to an audit table, who, what, when? (All)

# User Requirements

## MoSCoW Definitions

### MUST

Requirements that are put into the MUST section of the table are requirements that are necessary to fulfil in order to be able to label the project a success. If even one of the MUST requirements are not satisfied the then project would have been a failure.

### SHOULD

Requirements that are put into the SHOULD section of the table are requirements that are important to the success of the project although, if they are not fulfilled the project is not necessarily a failure. For the purpose of this project, requirements that are placed in the ‘should’ category are as important as those in the ‘must’ although, if they are not met we may be able to implement another feature or functionality that would satisfy the requirement in a different way.

### COULD

Requirements that are put into the COULD section of the table are those requirements that are not necessary to the success of the project although, will be included if time permits us to.

### WON’T

These are requirements that we have identified but feel that they are either not important and would not add value to the final product or it may be, that they are requirement that would not realistically be able to be implemented; this may be due to time, cost or resource constraints.

## User Requirements in the MoSCoW Table

|  |  |
| --- | --- |
| **MUST**   1. Login Screen 2. Carer Interface 3. Patient Interface 4. Personal Profile 5. Ability to update personal profile at any time after account creation 6. Ability to add medication incl. frequency, dose, name, criticality (determine the alert to the carer) 7. Access Control (Security, Privacy etc) 8. Accessibility Features (per WCAG 2.0)    1. Larger Fonts    2. Bold Fonts    3. Change colour scheme 9. Ability to de-activate the user account | **SHOULD**   1. Record and track medication that has/has not been taken 2. Reminder to take medication 3. Alert carer to abnormalities i.e. if medication hasn’t been taken 4. Ability to record appointments with the hospital/GP etc. 5. Emergency Alarm to carer, this feature may to be turned off 6. Interfacing with NHS Direct website 7. Database of drug names |
| **COULD**   1. Remind carer if they are running low on medication 2. Alert Pharmacy if they are low on medication (by email) 3. Track the movements of patients using google maps 4. Give directions to the doctors or hospital when the patient has an appointment 5. Live chat – patient is able call or instant message the carer 6. Heart Rate monitor    1. Ability to record and track this over time    2. Ability to alert carer of abnormalities 7. Blood Pressure monitor    1. Ability to record and track this over time    2. Ability to alert carer of abnormalities 8. API to support accessories 9. Wheelchair request/Wheelchair accessibility alerting (Integrate with TFL) 10. Links from emails to put appointments directly into the app 11. Facility to allow carers to notify patients easily if they are running late 12. Primary/Secondary Carers to account for holidays etc. 13. Text to speech functionality 14. Interface for relatives to be able to track patient medication/appointments etc. | **WON’T**   1. Call 999 if no response from carer/patient after emergency alarm is activated. 2. Support smart watches |