

RELAX REMINDER

Software Requirement Specification

**Project Code: 2R**

**Document Code: RR-DOC – v1.1**

**<Hoa Lac, 02/03/2014>**

Record of change

\*A - Added M - Modified D - Deleted

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Effective Date | Changed Items | A\* M, D | Change Description | New Version |
| 24/02/2014 | The whole document | A | Initiate version | 1.0 |
| 02/03/2014 | Appendix | A | Some user interface for easy implementation | 1.1 |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

TABLE OF CONTENTS

[1 Introduction 4](#_Toc381490726)

[1.1 Purpose 4](#_Toc381490727)

[1.2 Scope 4](#_Toc381490728)

[1.3 Definitions, Acronyms, and Abbreviations 4](#_Toc381490729)

[1.4 References 5](#_Toc381490730)

[1.5 Overview 5](#_Toc381490731)

[2 Overall Description 6](#_Toc381490732)

[2.1 Product Perspective: 6](#_Toc381490733)

[2.2 Product Functions: 6](#_Toc381490734)

[2.3 User characteristics 7](#_Toc381490735)

[2.4 General Constraints 8](#_Toc381490736)

[2.5 Requirements subsets 8](#_Toc381490737)

[3 Specific Requirements 9](#_Toc381490738)

[3.1 Functionality 9](#_Toc381490739)

[3.2 Reliability 22](#_Toc381490740)

[3.3 Performance 22](#_Toc381490741)

[3.4 Supportability 23](#_Toc381490742)

[3.5 Design Constraints 23](#_Toc381490743)

[3.6 Interfaces 23](#_Toc381490744)

[4 Appendix 25](#_Toc381490745)

[4.1 Appendix A: Relaxing Mode 25](#_Toc381490746)

[4.2 Appendix B: Dialog show when Study period end. 26](#_Toc381490747)

[4.3 Appendix C: Main Menu of Relax Reminder 27](#_Toc381490748)

[4.4 Appendix D: Main menu of CMS 28](#_Toc381490749)

# Introduction

## Purpose

The purpose of this document is to provide a consistent and complete description of the requirements for the software: Relax Reminder. Which identify existing project information and the software components that should be tested.

## Scope

We describe what features are in the scope of the software and what are not in the scope of the software to be developed.

*In Scope:*

1. Study function which help user to improve efficiency and focus by music. Reminder user when to stop learning and take a break.
2. Relax function will entertain the user by several type of content, help user relieve stress so he or she can continue learning.
3. For content manager system: implement functions: Manager Authentic, Modify, Add, Delete content

Out of scope:

1. License of Music and Content.
2. What music and content software use. It will be done by content Manager and 3rd party licensing services.

## Definitions, Acronyms, and Abbreviations

|  |  |  |
| --- | --- | --- |
| ID | Acronym | Definition |
| **1** | SRS | Software Requirement Specification |
| **2** | App | Application |
| **3** | RR | Relax Reminder (Software’s name) |
| **4** | OS | Operating System |
| **5** | GUI | Graphic User Interface |
| **6** | CMS | Content Manager System |
| **7** | DB | Database |
| **8** | CM | Content Manager |

## References

* Software requirement specification form
* SRS of Cafeteria Ordering System (by Karl Wiegers)
* Software engineering – Ninth Edition (by Ian Sommerville)

## Overview

We intend to develop the application with the following main part:

1. Client side: this part contains two main functions:
   1. Study function include features to help user study such as: set study interval (minute), how application remind user to take a break, play music help focus and gain efficiency, etc…
   2. Relax function include features to help user relax: application randomly select a content in database and display into screen which entertain user.
2. CMS side: this part contains all module involving manager the content of the “Relax function” such as: view, modify, add, delete, etc…

# Overall Description

## Product Perspective:

* RR is aimed toward student who:
  + Feel difficult to concentrate on learning
  + Want to optimize learning effectiveness (by manager time between study and relax combine with study music)
  + Have difficulty to find ways to relax
  + Waste a lot of time relaxing and can’t back to work
* RR is an Android Application. Which means it only works on Android OS. So the user need to have a device with Android OS installed on. RR is design for mainly use on Android Smart Phone and Android Tablet. The GUI should be as friendly as possible, easy to use and attractive to pupils and students.
* CMS for RR is a web-based app. It should run on any OS with a web browser and Internet supported. The Manager can easily update content, review feedbacks of user, etc….
* RR is intended to be a stand-alone product and should not depend on the availability of other software.

## Product Functions:

1. **Client side (Android app):**

|  |  |  |
| --- | --- | --- |
| Class of use cases | Functions need to implement | Description of function |
| Use cases related to setting | Set Learning duration | Set the amount of time for each learning phase |
| Set snooze duration | Set the amount of time for each Snooze |
| Set Relaxing duration | Set the amount of time for each relaxing phase |
| Set Alarm Type | Choose how App will notify use when it’s time to relax |
| Set Alarm Sound | Choose sound from sound collection of RR. This sound will play when alarm fired. |
| Use cases related to Study | Enable play music | Choose to play music while learning or not |
| Set music song | Choose music in App’s database to play while learning |
| Force stop learning | Force app to quit study mode even the user doesn’t learn enough time |
| Pause learning | Pause the timer until use resume it |
| Use cases related to Relax | Display another relaxing content | Change to another relaxing content |
| Stop relaxing | Stop relaxing mode and return user to home of the application. |
| Use cases related to notification, alarm | Notify when user learn enough | Notify use to relax |
| Snooze notify to continue learning | Snooze notify so user can continue learning. |
| Notify when user reach relaxing length | Notify user to return to work |
| Snooze to relax more | App give more time for use to relax. |
| Use cases related to feedback system | Send feedback or suggestion | Send feedback or app suggestion |

1. **CMS Side:**

|  |  |  |
| --- | --- | --- |
| Class of use cases | Use cases | Description of use cases |
| Use cases related to system authorization | Login | Login into CMS |
| Change Password | Change manager’s password |
| Use cases related to modify data | View existing content | View all existing content on database |
| Add content | Add new content to the database |
| Edit content | Edit an existing content |
| Remove content | Remove content form database |
| View feedback of user | View all user feedback which sent via Android App |

## User characteristics

Software doesn’t require any special characteristics of user. User just to have Android-based device.

## General Constraints

1. To update content database (Client Side), use need to have Internet connection
2. RR currently only support Android-based devices
3. RR is single-user software
4. CMS need Internet connection and web browser.

## Requirements subsets

1. RR will request following permissions on device:
   * Internet Access (to update content)
   * Alarm Service (to set alarm and reminder)

# Specific Requirements

## Functionality

|  |  |
| --- | --- |
| Primary Actor | Use Cases |
| User | 1. Configure Application Setting 2. Enter Study Mode 3. Pause Study Mode 4. Force stop Study mode 5. Snooze for study mode. 6. Enter Relaxing Mode 7. Update content 8. Send feedback and suggestion |
| Content Manager | 1. Login 2. Change Password 3. Add new content 4. View content database 5. Edit content 6. Remove content 7. View feedback and suggestion |

1. **Client Side**

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case ID: | RR01 | | |
| **Use Case Name:** | Configure Application Setting | | |
| **Created By:** | LongNH | **Last Updated By:** | LongNH |
| **Date Created:** | February 24, 2014 | **Date Last Updated:** | February 24, 2014 |
| **Actors:** | User | | |
| **Description:** | User enter Setting Activity to configure: Learning duration, Relaxing duration, Alarm method, Alarm sound | | |
| **Pre-conditions:** | User opened application | | |
| **Post-conditions:** | Changes to Application Setting made by user are updated | | |
| **Normal Flow:** | **1.0 Setting for Learning duration and Relaxing duration**   1. User click on [Setting] button (See [Appendix C](#_Appendix_C:_Main)) 2. App provide a screen where user can choose what setting to change, include learning duration, relaxing duration, alarm method, Restore default 3. User click on [Learning duration] or [Relaxing duration] 4. App asks the user for value of that setting 5. System saved setting | | |
| **Alternative Flows:** | **1.1 Setting for Alarm method** (branch after step 2)   1. User click on [Alarm method] 2. App asks the user to choose one of these option: [Sound Only], [Sound + Vibrate], [Vibrate Only] 3. Return to step 5   **1.2 Choose alarm sound** (brand after step 2)  **1.** User click on [Alarm sound]  2. App asks the user to choose sound from Alarm collection of RR  3. Return to step 5  **1.3 Restore to default setting** (brand after step 2)  1. User click on [Restore default]  2. App restored default setting  3. App inform user that all setting has been restore to default value. | | |
| **Exceptions:** | **1.0.E.1 Out of range value allowed for Learning duration**  1. In step 3, if you user set Learning duration > 1h30’ or < 25’ an warning message will appear tell user that the time which optimize for learning is between 45’ and 1 hour but setting still be saved. If user set Learning duration > 3 hours, an error message will appear to tell that time was too much and setting shouldn’t be saved | | |
| **Priority:** | High | | |
| **Special Requirements:** | **Value of Learning duration and Relaxing duration should satisfy the following conditions:**   * Can only be an Integer number. Don’t contain [space]. Cannot be null. * The unit of value is minute   **Default setting (when user click on [Restore default]:**   * Learning duration: 45 (minutes) * Relaxing duration: 5 (minutes) * Alarm method: Sound Only | | |
| **Notes and Issues:** | * User can discard value entered by click [Cancel] button and [Back] instead of [OK] button. * User can change any setting he/she want and in any order. No need to change all setting. | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case ID: | RR02 | | |
| **Use Case Name:** | Enter Study Mode | | |
| **Created By:** | LongNH | **Last Updated By:** | LongNH |
| **Date Created:** | February 24, 2014 | **Date Last Updated:** | February 24, 2014 |
| **Actors:** | User | | |
| **Description:** | User enter “Study mode”. This function support the learning process by notify user when to have a break by alarm. User can choose to play learning music while learning or don’t play anything. After that, the application will wait for amount time which user specified in [Learning duration] in Setting Activity. After this time, RR notify user to take a break. | | |
| **Pre-conditions:** | User opened application | | |
| **Post-conditions:** | * + 1. RR entered Study mode and run in the background     2. RR will notify user after finish a period time of learning | | |
| **Normal Flow:** | **1.0 User choose to play music while learning**   1. User click on [Study] button (see [Appendix C](#_Appendix_C:_Main)) 2. App ask user whether he/she want to play learning music while learning 3. User click on [YES] (play music while learning) 4. App ask user to choose learning music song(s) from App’s database. 5. User choose songs in the list. 6. App display a message that recommend user use headphone. 7. App enter Study mode and ready to fire alarm | | |
| **Alternative Flows:** | **1.1 User choose not to play music while learning** (branch after step 2)   1. User click on [NO] (don’t play music) 2. Return to step 7 | | |
| **Exceptions:** | **No Exception** | | |
| **Priority:** | High | | |
| **Special Requirements:** | **User can choose more than one songs, the application will play all of them.**  **When App entered Study mode:**   * RR will open “Learning screen” that provide   + Time left of the current period of learning   + [Pause] button   + [Stop learning] button * RR can be running in the background when device is locked or user switch to another application | | |
| **Notes and Issues:** | * User can cancel and go back to home of application whenever he/she on step 2 to step 5 (on 1.0) | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case ID: | RR03 | | |
| **Use Case Name:** | Pause Study mode | | |
| **Created By:** | LongNH | **Last Updated By:** | LongNH |
| **Date Created:** | February 24, 2014 | **Date Last Updated:** | February 24, 2014 |
| **Actors:** | User | | |
| **Description:** | While in study mode, user can pause the timer and resume it later. This may happen when user have something to do in short time such as: have a phone call | | |
| **Pre-conditions:** | 1. RR entered Study mode 2. RR run in background | | |
| **Post-conditions:** | 1. Reminder will be paused (timer is pause) until user resume it later 2. Music (if apply) will be paused | | |
| **Normal Flow:** | **1.0 User want to pause Study mode**   1. User click on [Pause] button 2. Timer will be paused, until user press [Resume] button later. 3. The [Pause] button will turn into [Resume] button | | |
| **Alternative Flows:** | **N/A** | | |
| **Exceptions:** | While Study mode is in pause state, user still can click on [Stop study] or [Back] button to stop Study mode. | | |
| **Priority:** | Low | | |
| **Special Requirements:** | * The timer and music play will be paused until user click on [Resume] button | | |
| **Notes and Issues:** | N/A | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case ID: | RR04 | | |
| **Use Case Name:** | Force stop study mode | | |
| **Created By:** | LongNH | **Last Updated By:** | LongNH |
| **Date Created:** | February 24, 2014 | **Date Last Updated:** | February 24, 2014 |
| **Actors:** | User | | |
| **Description:** | While in study mode, user can [Study mode] and go to the main menu. | | |
| **Pre-conditions:** | 1. RR entered Study mode 2. RR is running in background | | |
| **Post-conditions:** | Reminder and music will be stopped. App go to main menu. | | |
| **Normal Flow:** | **1.0 User want to stop Study mode**   1. User click on [Stop study] button 2. App ask you to confirm. 3. User press [Yes] 4. Timer and music will stop. 5. App return to main menu. | | |
| **Alternative Flows:** | **User can stop Study mode by click on [Back] button while in Study mode.**   1. User click on [Back] button 2. App ask for confirmation 3. User click [Yes]. 4. Timer and music will stop. 5. App return to main menu. | | |
| **Exceptions:** | No Exception | | |
| **Priority:** | Low | | |
| **Special Requirements:** | * The timer and music must be stop after user press [Yes] button. * Release any resources using by Music player and Timer | | |
| **Notes and Issues:** | N/A | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case ID: | RR05 | | |
| **Use Case Name:** | Snooze for study mode | | |
| **Created By:** | LongNH | **Last Updated By:** | LongNH |
| **Date Created:** | February 24, 2014 | **Date Last Updated:** | February 24, 2014 |
| **Actors:** | User | | |
| **Description:** | If the study period ended and alarm fire, but user still want to learn more, he/she can choose to learn more with desire time. | | |
| **Pre-conditions:** | 1. Study period ended. 2. Alarm fired with pop-up notification on screen. | | |
| **Post-conditions:** | Keep in study mode for xxx minutes (user choose), then notify user again. | | |
| **Normal Flow:** | **1.0 User want to snooze Study mode**   1. User click on [Study more] button 2. App ask user amount of time he/she want to snooze (by pop-up, with option: 5, 10, 15, 20, 25, ……, 60) 3. User press [OK] 4. Music keep playing (if apply) and app keep waiting for the time user chosen. | | |
| **Alternative Flows:** | N/A | | |
| **Exceptions:** | If user click [Cancel] button or [Back] button, App will stop Study mode and return to Main menu. | | |
| **Priority:** | High | | |
| **Special Requirements:** | * User can only choose amount of time which already in option: 5, 10, 15….60. | | |
| **Notes and Issues:** | N/A | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case ID: | RR06 | | |
| **Use Case Name:** | Enter Relaxing Mode | | |
| **Created By:** | LongNH | **Last Updated By:** | LongNH |
| **Date Created:** | February 24, 2014 | **Date Last Updated:** | February 24, 2014 |
| **Actors:** | User | | |
| **Description:** | User enter Relaxing Mode to relax, relief stress… | | |
| **Pre-conditions:** | User is in Main Menu **OR** after study period has ended | | |
| **Post-conditions:** | App entered Relaxing Mode | | |
| **Normal Flow:** | **1.0 When user is on Main Menu**   1. User click on [Relax] button 2. App enter Relaxing Mode   **1.1 When user have just finished study**  1. App ask whether user want to relax or not  2. User click on [Yes] button  3. App enter Relaxing Mode | | |
| **Alternative Flows:** | N/A | | |
| **Exceptions:** | If user press [Back] button or [Quit] button when App ask for entering Relaxing Mode (See [Appendix B](#_Appendix_B:_Dialog)) 🡪 Stop Study Mode and return to Main Menu | | |
| **Priority:** | High | | |
| **Special Requirements:** | * Relaxing Mode will have several contents and choose randomly by App (see [Appendix A](#_Appendix_A:_Relaxing)) * User can choose to see another Relax Content by click on [Next content] button (see [Appendix A](#_Appendix_A:_Relaxing)) | | |
| **Notes and Issues:** | N/A | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case ID: | RR07 | | |
| **Use Case Name:** | Update content | | |
| **Created By:** | LongNH | **Last Updated By:** | LongNH |
| **Date Created:** | February 24, 2014 | **Date Last Updated:** | February 24, 2014 |
| **Actors:** | User | | |
| **Description:** | This use case describe how Application update the content’s DB | | |
| **Pre-conditions:** | User is in Main Menu  Device have to internet connection | | |
| **Post-conditions:** | Content’s database of RR is updated. | | |
| **Normal Flow:** | 1. User click on [Update] button 2. App will run new Thread to update its database. 3. After update completed, App will notify user via Notification Bar that updated completed | | |
| **Alternative Flows:** | N/A | | |
| **Exceptions:** | In step 2, if the device suffers from lost network connect and error message will appear in Notification Bar | | |
| **Priority:** | High | | |
| **Special Requirements:** | * Updating database should run as a separate thread and not affect the user experience (he/she still can go to Study Mode or Relaxing Mode). The status of updating will update in Notification Bar | | |
| **Notes and Issues:** | N/A | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case ID: | RR08 | | |
| **Use Case Name:** | Send feedback and suggestion | | |
| **Created By:** | LongNH | **Last Updated By:** | LongNH |
| **Date Created:** | February 24, 2014 | **Date Last Updated:** | February 24, 2014 |
| **Actors:** | User | | |
| **Description:** | In this use case, user can send a simple text to server as a feedback or suggestion. Later, Content Manager can read it in CMS. | | |
| **Pre-conditions:** | User is in Main Menu | | |
| **Post-conditions:** | Feedback sent to CMS Server | | |
| **Normal Flow:** | 1. User click on [Send feedback & suggestion] button 2. New windows will show up 3. User provides information and message then click on [Send] button. 4. App will notify user whether message has sent or not (by Toast in Android) 5. App return to Main Menu | | |
| **Alternative Flows:** | N/A | | |
| **Exceptions:** | In step 2, if the device suffers from lost network connect and error message will display on screen by Toast service in Android and the send feedback windows still there. If user still want to send feedback, they should click on [Send] button again.  Or if user don’t want to send feedback anymore, he/she can click on [Back] button to return to Main Menu. | | |
| **Priority:** | High | | |
| **Special Requirements:** | * Information and message should satisfy the following conditions:   + Email (optional): a string from 1 to 256 characters in length, can contain any kind of characters.   + Name: a string from 1 to 256 characters in length, can contain any kind of characters. Cannot be null   + Message: a text box input, accept string from 1 to 1024 characters in length. Cannot be null | | |
| **Notes and Issues:** | N/A | | |

**2. CMS Side**

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case ID: | CMS01 | | |
| **Use Case Name:** | Login | | |
| **Created By:** | LongNH | **Last Updated By:** | LongNH |
| **Date Created:** | February 24, 2014 | **Date Last Updated:** | February 24, 2014 |
| **Actors:** | Content Manager | | |
| **Description:** | Content Manager provide their access credentials in order to use the CMS. | | |
| **Pre-conditions:** | To be pre-registered in the CMS.  To have internet access and a web browser. | | |
| **Post-conditions:** | A connection to database is established. | | |
| **Normal Flow:** | 1. Retrieve [Login] screen for administrators (Content Manager) 2. CM provides access information (User name and password) 3. CMS does authentication. 4. Main screen is displayed. | | |
| **Alternative Flows:** | 4(a). Authorization fails  4(a)1. Prompt the user that he/she typed the wrong access information.  4(a)2. Allow him/her to re-enter the access information. Give he/she 3 chances. | | |
| **Exceptions:** | N/A | | |
| **Priority:** | High | | |
| **Special Requirements:** | N/A | | |
| **Notes and Issues:** | The content of the error messages will be decided by developers. | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case ID: | CMS02 | | |
| **Use Case Name:** | Change password | | |
| **Created By:** | LongNH | **Last Updated By:** | LongNH |
| **Date Created:** | February 24, 2014 | **Date Last Updated:** | February 24, 2014 |
| **Actors:** | Content Manager | | |
| **Description:** | CM want to change their account’s password | | |
| **Pre-conditions:** | CM logged in | | |
| **Post-conditions:** | Password changed. | | |
| **Normal Flow:** | 1. CM is prompted for old password, new password and confirm new password 2. CM provide above information 3. CMS does authentication. 4. New password is registered with the system | | |
| **Alternative Flows:** | 4(a). Authorization fails  4(a)1. Prompt the user that he/she typed the wrong access information.  4(a)2. Allow him/her to re-enter the access information. Give he/she 3 chances.  4(b). New password and confirm new password do not match.  4(b)1. Allow him/she to re-enter the attributes. | | |
| **Exceptions:** | N/A | | |
| **Priority:** | High | | |
| **Special Requirements:** | N/A | | |
| **Notes and Issues:** | The content of the error messages will be decided by developers. | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case ID: | CMS02 | | |
| **Use Case Name:** | Add new content | | |
| **Created By:** | LongNH | **Last Updated By:** | LongNH |
| **Date Created:** | February 24, 2014 | **Date Last Updated:** | February 24, 2014 |
| **Actors:** | Content Manager | | |
| **Description:** | CM want to add new content to database. | | |
| **Pre-conditions:** | CM logged in. | | |
| **Post-conditions:** | New content added. | | |
| **Normal Flow:** | 1. User click on [Add new content] 2. CMS asks CM for: Content Name, Content Type, Content Data 3. CM provide provides all related information about new content then click [Save] button. 4. CMS queries the database 5. CMS accepts or rejects [Add content] request. | | |
| **Alternative Flows:** | N/A | | |
| **Exceptions:** | * In step 2, if CM provides insufficient data a message will appear and all required fields will be mark red. | | |
| **Priority:** | High | | |
| **Special Requirements:** | Information of the content should satisfy the following conditions:   * Content Name: a string from 1 to 256 characters in length, can contain any kind of characters. Cannot be null. * Content Type: CM choose a type from a specified list * Content Data: a string, unlimited characters, but this field will depend on Content Type. Different Content Type will have different Content Data input. Cannot be null. | | |
| **Notes and Issues:** | The content of the error messages will be decided by developers. | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case ID: | CMS03 | | |
| **Use Case Name:** | View content database | | |
| **Created By:** | LongNH | **Last Updated By:** | LongNH |
| **Date Created:** | February 24, 2014 | **Date Last Updated:** | February 24, 2014 |
| **Actors:** | Content Manager | | |
| **Description:** | CM want to add new content to database. | | |
| **Pre-conditions:** | CM logged in. | | |
| **Post-conditions:** | N/A | | |
| **Normal Flow:** | 1. User click on [View all content] button 2. CMS list all content available in database with: content name, content type 3. CM click on Content name he/she wants to see more details. 4. CMS show a pop-up with all data for that content | | |
| **Alternative Flows:** | After step 2, user can click con [Search] button to search content base on its name. Or sort content by Name or by Type. | | |
| **Exceptions:** | N/A | | |
| **Priority:** | Average | | |
| **Special Requirements:** | * CM should enter in Search box a string. | | |
| **Notes and Issues:** | Developers must ensure that CM can’t exploit any security on this search box. | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case ID: | CMS04 | | |
| **Use Case Name:** | Edit content | | |
| **Created By:** | LongNH | **Last Updated By:** | LongNH |
| **Date Created:** | February 24, 2014 | **Date Last Updated:** | February 24, 2014 |
| **Actors:** | Content Manager | | |
| **Description:** | CM want to add new content to database. | | |
| **Pre-conditions:** | CM logged in. | | |
| **Post-conditions:** | N/A | | |
| **Normal Flow:** | 1. User click on [Edit content] button (See Appendix D) 2. CMS redirect user to [View content mode] 3. In the list of content after filtering by search function or sort. 4. CM click on [Edit] icon next to Content he/she want to edit 5. CMS show a pop-up with editable fields 6. CM edit data he/she wants to change 7. CM click [Save] button 8. CMS queries to database to accepts or rejects the request | | |
| **Alternative Flows:** | After step 1, if user click on Content name to see more details, he/she also can click on [Edit] button on that pop-up screen.  Return to step 3. | | |
| **Exceptions:** | N/A | | |
| **Priority:** | Average | | |
| **Special Requirements:** | * Content name shouldn’t be duplicate with other existing content. | | |
| **Notes and Issues:** | N/A | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case ID: | CMS05 | | |
| **Use Case Name:** | Remove content | | |
| **Created By:** | LongNH | **Last Updated By:** | LongNH |
| **Date Created:** | February 24, 2014 | **Date Last Updated:** | February 24, 2014 |
| **Actors:** | Content Manager | | |
| **Description:** | CM want to delete a content from database | | |
| **Pre-conditions:** | CM logged in. | | |
| **Post-conditions:** | N/A | | |
| **Normal Flow:** | 1. User click on [Delete content] button (See Appendix D) 2. CMS redirect user to [View content mode] 3. In the list of content after filtering by search function or sort. 4. CM click on [Delete] icon next to Content he/she want to Remove 5. CMS show a pop-up ask for confirmation. 6. CMS delete the content. | | |
| **Alternative Flows:** | N/A | | |
| **Exceptions:** | N/A | | |
| **Priority:** | Average | | |
| **Special Requirements:** | N/A | | |
| **Notes and Issues:** | N/A | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case ID: | CMS06 | | |
| **Use Case Name:** | Remove content | | |
| **Created By:** | LongNH | **Last Updated By:** | LongNH |
| **Date Created:** | February 24, 2014 | **Date Last Updated:** | February 24, 2014 |
| **Actors:** | Content Manager | | |
| **Description:** | CM want to delete a content from database | | |
| **Pre-conditions:** | CM logged in.  CM entered “View content” function | | |
| **Post-conditions:** | N/A | | |
| **Normal Flow:** | 1. User click on [Edit content] button (See Appendix D) 2. CMS redirect user to [View content mode] 3. In the list of content after filtering by search function or sort. 4. CM click on [Delete] icon next to Content he/she want to Remove 5. CMS show a pop-up ask for confirmation. 6. CMS delete the content. | | |
| **Alternative Flows:** | N/A | | |
| **Exceptions:** | N/A | | |
| **Priority:** | Average | | |
| **Special Requirements:** | N/A | | |
| **Notes and Issues:** | N/A | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case ID: | CMS07 | | |
| **Use Case Name:** | View feedback and suggestion | | |
| **Created By:** | LongNH | **Last Updated By:** | LongNH |
| **Date Created:** | February 24, 2014 | **Date Last Updated:** | February 24, 2014 |
| **Actors:** | Content Manager | | |
| **Description:** | CM want to view Feedback from user | | |
| **Pre-conditions:** | CM logged in. | | |
| **Post-conditions:** | N/A | | |
| **Normal Flow:** | 1. User click on [View F&S] button (See Appendix D) 2. CMS show a list of feedback by user and sort it by Time. | | |
| **Alternative Flows:** | After step 2, CM can click on [Mark as read] button to mark that feedback as read. | | |
| **Exceptions:** | N/A | | |
| **Priority:** | Average | | |
| **Special Requirements:** | * Feedbacks which marked as read, will be blurred so other item not read will easily to notice. | | |
| **Notes and Issues:** | N/A | | |

## Reliability

* CMS server can be available 24/7 and can support a large number of queries for updating new content and receive feedback from user.
* When system failed it would took 10 minutes to completely restart and fully operate again.
* The maximum defect rate is about 20-25 bugs per 1000 lines of code.

## Performance

1. RR should be able to run on Android API 10 and higher API.
2. 90% of responses should be within 1 sec, except for Update content for which more time is acceptable

## Supportability

* The software development phase will follow the Project Coding Convention which is designed to help the team create explicit source code.
* Besides, the development will follow the Object-Oriented Paradigm. This will lead to easily maintenance and upgrading of the application in the future.

## Design Constraints

* Security: The database on CMS server should be secured against malicious deformations.
* Fault Tolerance: Data should not become corrupted in case of system crash or power failure.
* Backup: The database should be backup once 1 week.

## Interfaces

### User Interfaces

A graphic user interface will be available in all workflow scenarios to allow easy to access to all features of the system. The navigation options in every screen will be similar in order to help user can reduce the time learning and using system. All error occurring and exception handling will be caught and displayed for user with friendly messages.

GUI should follow Google Android Pattern.

### Hardware Interfaces

The prototype software (RR) will be implemented on Android Device:

* Google Galaxy Nexus
* LG LTE 2
* LG G Pro
* Google Nexus 7 (tablet)

CMS will be implemented on remote server (real server).

### Software Interfaces

Android device: Android API 10 to Android API 19

Computer: Any OS version with internet connection and internet browser.

### Communications Interfaces

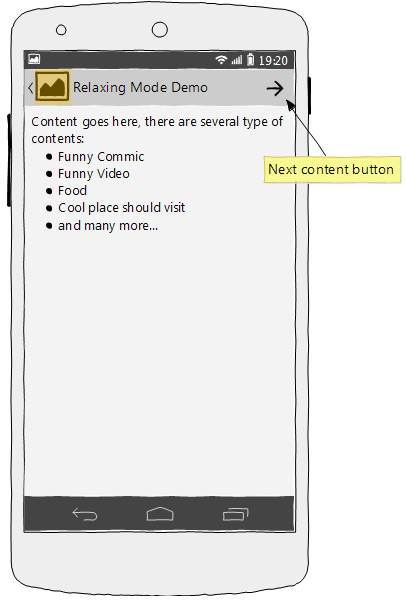
RR app use HTTP1.0 (or over) connection to communicate with server over the Internet

On CMS side will use:

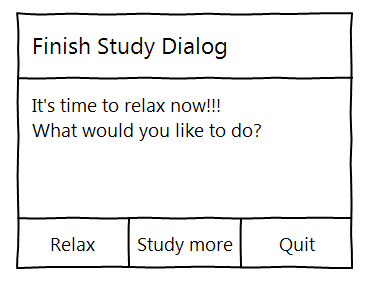
* PHP System (for all web interface and communication with MySQL database)
* MySQL (for storing database)
* Apache service (for HTTP protocol and PHP)

# Appendix

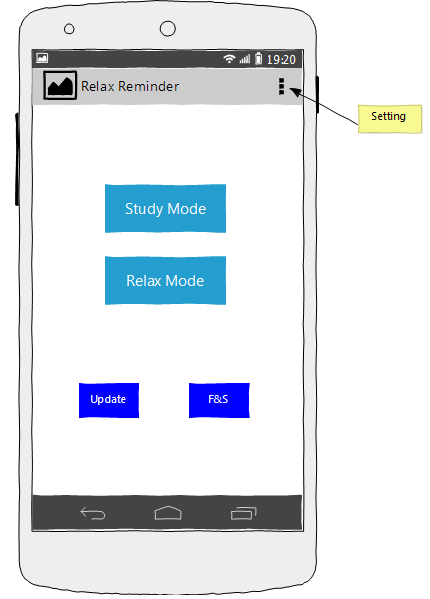
## Appendix A: Relaxing Mode



## Appendix B: Dialog show when Study period end.



## Appendix C: Main Menu of Relax Reminder



## Appendix D: Main menu of CMS

