Software Requirements Specification

For

NUForAll

Version 1.0 approved

Prepared by Siddharth Shashikar

NIIT University

17/9/2016

Table of Contents

Revision History

Name	Date	Reason For Changes	Version

1. Introduction

1.1 Purpose

The purpose of the project is to integrate the three services that are Gate-pass, Library and the Nucleus of NIIT University in one place so that the user can access them more easily and conveniently.

1.2 Document Conventions

- Font family used in the entire document is: Arial.
- Font size used in the entire document is: 11.
- Headings are bold and the remaining text is normal.
- Line spacing is 1.15 in the entire document.

1.3 Intended Audience and Reading Suggestions

This SRS document targets our client i.e. NIIT University, Developers, Product Owner. For Clients and the Product Owner they should follow the index for reading this document and for the Developers they skip the initial part that is the “Introduction” of this SRS document and straight go to the “Overall Description” that is the second part of the document.

1.4 Product Scope

The objective of this project is that we are integrating all the three services that is provided by the NIIT University that are Gate-pass, Nucleus and Library. We will be developing hybrid applications (runs on all the three platforms Android, ios and windows) that will bring together the readily used functionalities by the students of NU.Some of the readily used functionalities are:-
Attendance, Time Table is the most used frequently used features of Nucleus, Applying Gate-pass both Local and Outstation is the most frequently used and due date and issued date notifications, searching books availability are the most frequently used feature of Library System.

1.5 References

N/A

2. Overall Description

2.1 Product Perspective

Our product is basically a new self contained product that aims on bringing the readily used functionalities of the Moodle, Nucleus and Library system.

2.2 Product Functions

List of some functionality in brief that will be provided by our product:-

- Login
    - Students will login to the application through their Gmail account having domain name as "*St.niituniversity.in*".

- Nucleus
    - Checking Attendance
    - Time table

- Gate-pass
    - Apply gate-pass both local and outstation
    - Checking request status
    - Cancelling gate pass

- Library
    - Issued book details
        - Issued date notifications
        - Due date notifications
        - Fine details
    - Searching books

2.3 User Classes and Characteristics

Our Application has a single user class, NU Students who will use the product functions defined above for the ease of using the current system. The students are the potential customers. The App aims to simplify the above mentioned three systems and allow the students to access them with ease and reliability.

2.4 Operating Environment

The product will work equally well on all the three platforms i.e. Android, ios, Windows. Android version 4.4.4 (Kitkat) or above is required, similarly for ios version 7.1.2 or above is required, windows version 8.10.12359.845 or above is highly recommended. The application development shall be performed under the following environment:

- Operating System: Ubuntu 16.04 (not a must but is used by the team for resonance in installation of software, and frameworks)
- Text Editor: VSCode and Atom.
- Framework: Ionic framework includes Angular and Cordova framework.
- Programming or development language: Typescript, HTML, CSS, SASS, JS.

2.5 Design and Implementation Constraints

This application is explicitly developed for the students of NIIT University, which restricts the usage outside the University students. The app will work on older versions of the platform as well, but it is recommended to upgrade. Memory requirements are minimal. Time constraints to develop the App are a challenge. The main issue will be on the constraints that the university poses on the access of the NU Server and Database access

2.6 User Documentation

There is no such user manual required for operating this product as the operations will be very intuitive in nature although the user will be provided with a small demonstrations for the first time after installing the application on his/her device.

2.7 Assumptions and Dependencies

The project does have few dependencies on the NIIT's Moodle, Nucleus, and Library System. We also assume that the NU will provide internet connection on mobile. The project will be affected if these any of these dependencies or the assumption is incorrect or not shared.

3. External Interface Requirements

3.1 User Interfaces

Below is the prototype of the application, although this was made for getting an idea about the application and is not final.

3.2 Hardware Interfaces

The application supports mobile devices running on Android, ios and Windows

3.3 Software Interfaces

The application connects with the following interfaces:

1. Shall connect to the Gate pass Database through a REST API which would also be developed by the developers of the project.
2. Shall connect to the Nucleus API (developed by the TCO team of NIIT University) to get Timetable and attendance from the database.
3. Shall connect to the Koha Database of the Library to get details of library related operations.
4. Shall connect to the Google Firebase, for authentication and notification to student about library and gate pass details.
5. Shall have local storage to store the timetable and other details of the app.

3.4 Communications Interfaces

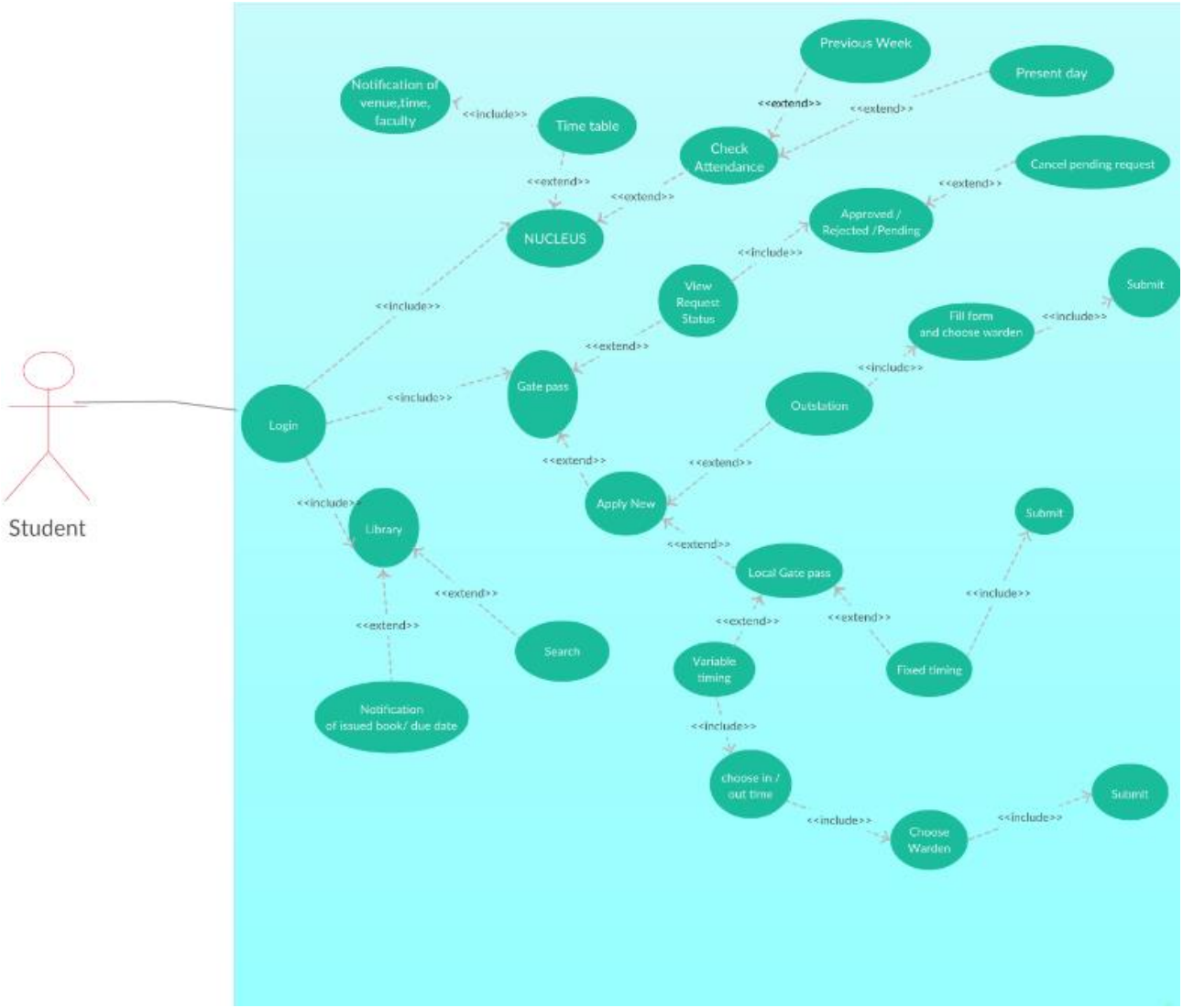
The application for now just expects to use FCM (previously GCM) to send notification to the client side application.

4. System Features

The product will have three major subdivisions which are discussed below:-

Identifier for Req.	Short Name	Short Description
RQ1	Login	Enables the user to login by the Gmail Google account to get authenticated.
RQ2	Gate pass	Contains all the frequently used functionalities involved in Gate pass, treated as a module of the app.
RQ3	Library	Contains the basic functionalities of Library, treated as a module of the app.
RQ4	Nucleus	Contains the functionalities of ERP : Timetable and Attendance, treated as a module of the app.
RQ5	Apply Gate pass	Enables the user to apply the Gate pass of his/her choice with ease.
RQ6	Local Gate pass	Enables the user to apply the local Gate pass, which is divided in fixed timing and flexible timing.
RQ7	Fixed Timing	Enables the user to apply the fixed timing for the local gate pass, which comes under Auto-Approval..
RQ8	Flexible Timing	Enables the user to apply the Variable timing Gate pass, which has to be accepted by the warden.
RQ9	In-Out time	An extension of Flexible gate passes which make the user to include making the in-out time fill it and then submit.
RQ10	Choose Warden	An extension of flexible local gate pass which makes the user to make the request to his/her selected warden and can note down his/her number in case of any dispute.
RQ11	Submit	This functionality comes under local as well as outstation gate pass which enables the user to successfully request for his/her gate pass.
RQ12	Apply outstation gate pass	Enables the user to apply for out station gate pass.
RQ13	Form fill	As a stimulus of the outstation gate pass, form will be generated to be filled by the user which contains the destination address and date and time of arrival and departure.

RQ14	Status Bar	Contains the current status of the requested gate pass by the user, showing whether it is approved, pending or rejected.
RQ15	Cancel Gate pass	Shows the functionality to cancel the pending gate pass by the user.
RQ16	Cancel Order	The user can cancel the order if they are not willing to receive it after payment and give the appropriate reasons for canceling and proceed to a portal for refund.
RQ17	Notification of issued book or due date.	Gives the notification to the user regarding the due date for book issue and also the book check-out.
RQ18	Search book	Enables the user to search for a book in the library whether it is available, reserved or not present.
RQ19	Timetable	Enables the user to look for his timetable schedule.
RQ20	Class schedule notification	This enables the user to get the information regarding the next class such as its venue and time.



5. Other Nonfunctional Requirements

5.1 Performance Requirements

For better performance it's must to have an Android version *4.4.4 (Kitkat)* or above, similarly for *ios version 7.1.2* or above is required, *windows version 8.10.12359.845* or above is highly recommended. Other performance requirement for running this application on your mobile is that the user must have a good internet connection accessing the functionalities provided by the product.

5.2 Safety Requirements

As the App needs to constantly communicate with the NU Database and server, it is very important that safety features are taken into consideration. The App will use MD-3 Encryption to retrieve and fetch data. Bugs in the code is ensured to be kept as minimum as possible. One advantage is that this App is limited only to the students of NIIT so it will be comparatively easier to monitor the safety issues.

5.3 Security Requirements

TBD

5.4 Software Quality Attributes

The product should have the following quality attributes:

- Adaptability – The user should be able to easily able to figure out the process flow.
- Reliability and Correctness – The product should provide correct and accurate results.
- Robustness – The application should notify the respective user if something is not correct for example his login session has expired or the user has a bas internet connection.

5.5 Business Rules

TBD

6. Other Requirements

Here is a list of few other requirements in this project:

1. API to interact with the NIIT University's Moodle DB.
2. API to interact with the NIIT University's Nucleus DB.
3. API to interact with Koha (Library management software).

Appendix A: Glossary

TBD – To be decided
N/A – Not Applicable
NU– NIIT UNIVERISTY

API – Application Programmer Interface
DB – Data Base
i.e. – that is
FCM – Firebase Cloud Messaging
GCM – Google Cloud Messaging

Appendix B: Analysis Models

For knowing about the analysis models, refer to the attached Use-Case diagram.

Appendix C: To Be Determined List

1. *Document convention*
2. *References*
3. *Design and Implementation constraints*
4. *Safety Requirements*
5. *Security requirements*
6. *Business Rules*