
Software Requirements Specification

for

NU APP

Version 1.0 approved

Prepared by VISHAL MOHANKUMAR

NIIT UNIVERSITY

18-09-2016

Table of Contents

Table of Contents	ii
Revision History	iii
1. Introduction.....	1
1.1 Purpose	1
1.2 Document Conventions.....	1
1.3 Intended Audience and Reading Suggestions.....	1
1.4 Product Scope	1
1.5 References.....	1
2. Overall Description	2
2.1 Product Perspective	2
2.2 Product Functions	2
2.3 User Classes and Characteristics	2
2.4 Operating Environment.....	2
2.5 Design and Implementation Constraints.....	2
2.6 User Documentation	3
2.7 Assumptions and Dependencies	3
3. External Interface Requirements	3
3.1 User Interfaces	3
3.2 Hardware Interfaces	3
3.3 Software Interfaces	3
3.4 Communications Interfaces	3
4. System Features.....	4
4.1 Authentication.....	4
4.2 Gate pass	Error! Bookmark not defined.
4.3 Library.....	4
4.4 Nucleus	5
4.5 Apply Gate pass	5
4.6 Fixed Timing.....	6
4.7 Flexible	
Timing.....	6
4.8 In-out Timing.....	7
4.9 Choose Warden.....	7
4.10	
Submit.....	8
4.11 Apply outstation Gate pass.....	8
4.12 Form Fill.....	9
4.13 StatusBar.....	9
4.14 Notification of issued book or due	
date.....	10
4.15 Search Book.....	10
4.16	
Timetable.....	11
4.17 Class Schedule	
Notification.....	11
4.18	
Attendance.....	12
4.19 Present	
day.....	12
4.20 Last Week.....	12
5. Other Nonfunctional Requirements	13

5.1 Performance Requirements	13
5.2 Safety Requirements	13
5.3 Security Requirements	13
5.4 Software Quality Attributes	13
5.5 Business Rules	13
6. Other Requirements	13
Appendix A: Glossary	13
Appendix B: Analysis Models	14

Revision History

Name	Date	Reason for Changes	Version
VISHAL	18-09-2016	THE FIRST DRAFT	1.0

1. Introduction

1.1 Purpose

This Application is developed for the students of NIIT University, so as to integrate the Gate pass system, Library system and Nucleus under a single mobile application that will be convenient for the users and reliable.

1.2 Document Conventions

The Document uses Times font in Bold for heading and Arial font in italics for explanations. Spacing as 1 unit between lines. The SRS document is arranged as per the index given.

1.3 Intended Audience and Reading Suggestions

This document is intended for developers, project managers, marketing staff, users, testers, and documentation writers.

1.4 Product Scope

The product is a Hybrid Mobile Application for the students of NIIT University to improve the current system by making it more simple, reliable and handy. The App incorporated the most frequently used features of a larger system.

Benefits:

- *The Current system will be made very handy; the users can check their daily required features in a mobile Application.*
- *Reliable system. Can help in last minute Gate pass.*
- *Check attendance in a click*
- *Help in avoiding penalty for late book submission.*
- *Will function in all platforms.*

What the Application is not:

- *This app is not a new system or replacement. It is intended to simplify the current process.*

1.5 References

The Application include features of the following websites:

- <https://moodle.niituniversity.in/moodle/>
- <https://nucleus.niituniversity.in/>

2. Overall Description

2.1 Product Perspective

This App is a new developed mobile application that is a simplified model of the current system that is developed for the students so as to make the current system more approachable and handy. The App is a sub product of a larger system that has been simplified as per the priority needs of the students on the daily basis. The App aims only to project those features that are applicable on daily basis. The App does not replicate the UI/UX of the current system, but to produce a simpler application to access the features that are being repeatedly in use.

2.2 Product Functions

The App opens up with the home page of displaying Time Table that is connected to Nucleus, students can view the previous, current and the next class. Next page will have the Gate pass system where the students can view the requested Gate pass status, request for new Gate pass and see all the applied Gate pass in time order. Library system shows the borrowed Books, Due dates and availability of certain book.

- *Users should login with the university domain email.*
- *Users can apply for Gate pass.*
- *Users can keep a track of the books borrowed.*
- *Users can check their timetable with one click*
- *User can check the availability of the books.*
- *Users can check their attendance with ease.*

2.3 User Classes and Characteristics

The App 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 Application is a Web App, that will be compatible for all the Operating systems and Platforms; i.e. Android, IOS, Microsoft Windows, etc. This app is built on IONIC-2 Framework with Angular-2 modules in typescript mode with installed Cordova Plugins. Firebase will be used as a BaaS for authenticating and for push notifications. VSCode editor is used.

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 be in English language and 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 is 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

A set of visual instructions will be made available in the application itself. This application doesn't require an explicit User manual because the students are already familiar with the system and the application main motive is to simplify the process, which will surely be reflected in the simplicity of the app.

2.7 Assumptions and Dependencies

This app will use API to retrieve the data from the NU database and also need the access to the NU servers so that any changes made must be reflected. Security will be a concern while accessing the systems. The Firebase cloud service is used for storage and hosting.

3. External Interface Requirements

3.1 User Interfaces

The user interface is kept very simple for the ease of the students and is not overloaded with features and buttons. The Buttons are made of cards. The pages are to be moved in a sliding format. The side menu bar will be available at every page. Page navigation is through sliding. Due date is displayed at the homepage itself with red color label to signify the importance.

3.2 Hardware Interfaces

The hardware to run this Application is very generic, as this is a web App this App can be deployed in any platform, i.e. Android, IOS, Microsoft Windows. Another type of hardware is access to NU database, NU servers, cloud server. API call are used to retrieve the data from database and servers.

3.3 Software Interfaces

This Application is built on Linux operating system (Ubuntu). Ionic 2 frameworks are used to improve the GUI of the Web App; Angular 2 modules are used for library including Cordova plugins. For Cloud services, Firebase is used as a Backend as a service which will allow authentication and push notifications. API's are developed to retrieve data from the NU Database.

3.4 Communications Interfaces

Email notifications and push notification are enabled. Connection with servers. All these connections with firebase and servers need a reliable communication that can be achieved through network protocol HTTP and sometimes FTP Database exchange of information exchange will be made in HTTP with MD-5 encryption to make it secure.

4. System Features

4.1 Authentication

4.1.1 Description and Priority

Any user who need to access the app must be signed in by google NIIT account
Priority – High

4.1.2 Stimulation/Response Design

The system responds when the user fills in the login details.

4.1.3 Functional Requirement

*RQ-1 : Working App
RQ-2 : Stable internet connection
RQ-3 : Valid student account.*

4.2 Gate pass

4.2.1 Description and Priority

Users can apply gate pass
Priority – Medium

4.2.2 Stimulation/Response Design

Gate pass request is generated and accept or reject is the result

Functional Requirement

*RQ-1 : Working App
RQ-2 : Stable internet connection
RQ-3 : Valid student account.*

4.3 Library

4.3.1 Description and Priority

Users can check book availability

Priority – Medium

4.3.2 **Stimulation/Response Design**

Users can search or a book by requesting the KOHA server

4.3.3 **Functional Requirement**

RQ-1 : Working App

RQ-2 : Stable internet connection

RQ-3 : Valid student account.

4.4 **Nucleus**

4.4.1 **Description and Priority**

Students attendance and timetable can be checked

Priority - Medium

4.4.2 **Stimulation/Response Design**

Request is given to NU server and response is in the form of parsing format.

4.4.3 **Functional Requirement**

RQ-1 : Working App

RQ-2 : Stable internet connection

RQ-3 : Valid student account.

4.5 **Apply Gate pass**

4.5.1 **Description and Priority**

User can apply for gatepass

4.5.2 **Stimulation/Response Design**

Request is pushed to NU server and the result is pushed with the help of firebase.

4.5.3 **Functional Requirement**

RQ-1 : Working App

RQ-2 : Stable internet connection

RQ-3 : Valid student account.

Local Gate pass

4.5.4 Description and Priority

User can apply for a local gate pass.

Priority - High

4.5.5 Stimulation/Response Design

Timings need to be specified and gate pass is requested via NU server.

4.5.6 Functional Requirement

RQ-1 : Working App

RQ-2 : Stable internet connection

RQ-3 : Valid student account.

4.6 Fixed Timing

4.6.1 Description and Priority

C

Priority - High

4.6.2 Stimulation/Response Design

Auto approval of this kind of Gate pass.

4.6.3 Functional Requirement

RQ-1 : Working App

RQ-2 : Stable internet connection

RQ-3 : Valid student account.

4.7 Flexible Timing

4.7.1 Description and Priority

Features involves the permission of wardens.

Priority - High

4.7.2 Stimulation/Response Design

*Request is pushed to NU server and the result is pushed with the help of firebase.
Warden needs to give permission.*

Functional Requirement

RQ-1 : Working App

RQ-2 : Stable internet connection

RQ-3 : Valid student account.

4.8 In-out time

4.8.1 Description and Priority

Timing needs to be specified for Gate pass

4.8.2 Stimulation/Response Design

This feature is accessed through drop down of timing clause.

4.8.3 Functional Requirement

RQ-1 : Working App

RQ-2 : Stable internet connection

RQ-3 : Valid student account.

4.9 Choose Warden

4.9.1 Description and Priority

Warden's name can be selected at the time of requesting Gate pass.

4.9.2 Stimulation/Response Design

This feature is accessed through drop down of timing clause.

4.9.3 Functional Requirement

RQ-1 : Working App

RQ-2 : Stable internet connection

RQ-3 : Valid student account.

4.10 Submit

4.10.1 Description and Priority

This is a button via which the generated form is pushed to NU server

Priority - Medium

4.10.2 Stimulation/Response Design

Status is updated as in NU server

4.10.3 Functional Requirement

RQ-1 : Working App

RQ-2 : Stable internet connection

RQ-3 : Valid student account.

4.11 Apply Outstation Gatepass

4.11.1 Description and Priority

This enables the user to apply the outstation Gatepass.

4.11.2 Stimulation/Response Design

Users can use outstation gate pass if they are leaving the locality.

Outstation option needs to be checked so that the request is pushed to NU servers.

Functional Requirement

RQ-1 : Working App

RQ-2 : Stable internet connection

RQ-3 : Valid student account.

4.12 Form fill

4.12.1 Description and Priority

*The Gate pass request can be created with the help of form.
Priority - High*

4.12.2 Stimulation/Response Design

The above mentioned option of Submit is used to push the form to the NU server.

4.12.3 Functional Requirement

*RQ-1 : Working App
RQ-2 : Stable internet connection
RQ-3 : Valid student account.*

4.13 Status bar

4.13.1 Description and Priority

Gives the accept or rejected status in the notification bar.

4.13.1 Stimulation/Response Design

Pop up to give the status.

4.13.2 Functional Requirement

*RQ-1 : Working App
RQ-2 : Stable internet connection
RQ-3 : Valid student account.*

Cancel request

4.13.3 Description and Priority

This feature enables the user to cancel pending gatepass request.

4.13.4 Stimulation/Response Design

The cancel gate pass is updated in the NU server which is pushed to notification.

4.13.5 Functional Requirement

RQ-1 : Working App

RQ-2 : Stable internet connection

RQ-3 : Valid student account.

4.14 Notification of issued book or due date

4.14.1 Description and Priority

This feature gives the details about the due date regarding the books issued.

4.14.2 Stimulation/Response Design

The system will give the push notifications to the user about the due dates of book issued.

4.14.3 Functional Requirement

RQ-1 : Working App

RQ-2 : Stable internet connection

RQ-3 : Valid student account.

4.15 Search Book

4.15.1 Description and Priority

This feature enables the user to search for a book in library to check.

4.15.2 Stimulation/Response Design

The system will respond with a query result whether the book is available or not.

4.15.3 Functional Requirement

RQ-1 : Working App

RQ-2 : Stable internet connection

RQ-3 : Valid student account.

4.16 Timetable

4.16.1 Description and Priority

This feature gives the details of the timetable.

4.16.2 **Stimulation/Response Design**

The system is requested for time table and the data is returned in parsing format.

4.16.3 **Functional Requirement**

*RQ-1 : Working App
RQ-2 : Stable internet connection
RQ-3 : Valid student account.*

4.17 **Class schedule notification**

4.17.1 **Description and Priority**

The feature notificatifies about the venue, time and date of the class next to current time.

4.17.2 **Stimulation/Response Design**

The system will return a notification to the user giving all the details of the class going to be the next.

4.17.3 **Functional Requirement**

*RQ-1 : Working App
RQ-2 : Stable internet connection
RQ-3 : Valid student account.*

4.18 **Attendance**

4.18.1 **Description and Priority**

The feature displays the attendance of the user.

4.18.2 **Stimulation/Response Design**

The user requests the NU server for attendance and the response is pushed in parsing format.

4.18.3 **Functional Requirement**

*RQ-1 : Working App
RQ-2 : Stable internet connection
RQ-3 : Valid student account.*

4.19 Present day**4.19.1 Description and Priority**

Attendance feature returns the current day attendance.

4.19.2 Stimulation/Response Design

Response is pushed to the app with detailed report.

4.19.3 Functional Requirement

RQ-1 : Working App

RQ-2 : Stable internet connection

RQ-3 : Valid student account.

4.20 Last week**4.20.1 Description and Priority**

Attendance feature includes last week attendance of the user,

4.20.2 Stimulation/Response Design

Response is pushed to the app with detailed report.

4.20.3 Functional Requirement

RQ-1 : Working App

RQ-2 : Stable internet connection

RQ-3 : Valid student account.

Identifier for Req.	Short Name	Short Description
RQ1	Authentication	<i>Any user who need to access the app must be signed in by google NIIT</i>

		<i>account</i>
RQ2	Gate pass	<i>Users can apply gate pass</i>
RQ3	Library	<i>Users can check book availability</i>
RQ4	Nucleus	<i>Students attendance and timetable can be checked</i>
RQ5	Apply Gate pass – Local Gate pass	<i>User can apply for gatepass</i>
RQ6	Fixed Timing	<i>User can apply for gatepass</i>
RQ7	Flexible Timing	<i>Features involves the permission of wardens.</i>
RQ8	In-out time	<i>Timing needs to be specified for Gate pass</i>
RQ9	Choose Warden	<i>Warden's name can be seleted at the time of requesting Gate pass.</i>
RQ10	Submit	<i>This is a button via which the generated form is pushed to NU</i>

		server
RQ11	Apply Outstation Gatepass	<i>This enables the user to apply the outstation Gatepass.</i>
RQ12	Form fill	<i>The Gate pass request can be created with the help of form.</i>
RQ13	Status bar	<i>Gives the accept r rejected status in the notification bar.</i>
RQ14	Notification of issued book or due date	<i>This feature gives the details about the due date regarding the books issued.</i>
RQ15	Search Book	<i>This feature enables the user to search for a book in library to check.</i>
RQ16	Timetable	<i>This feature gives the details of the timetable.</i>
RQ17	Class schedule notification	<i>The feature notificatifies about the venue, time and date of the class next to current time.</i>
RQ18	Attendance	<i>The feature displays the attendance of the user.</i>
RQ19	Present day	<i>Attendance feature returns the current day attendance.</i>

RQ20	Last week	<i>Attendance feature includes last week attendance of the user,</i>

5. Other Nonfunctional Requirements

5.1 Performance Requirements

For the App to work fast and reliably, Ionic 2 framework along with Angular 2 have been implemented. This gives the UI/UX better look and the performance of the App is enhanced. As the tools are the current version of the market, it is more stable with lesser security issues. The App responds to the user and works with the minimum time delay.

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

The authentication of users is being done by Google Firebase. The at most concern is the Data retrieval and update to the NU database and servers. MD-5 Encryption can be used to overcome the issue of someone tapping the information.

5.4 Software Quality Attributes

The App aims to deliver adaptability, availability, correctness, flexibility, interoperability, maintainability, portability, reliability, reusability, robustness, testability, and usability. The App will provide the users with the most robust and simple App that can help improving the current system. Many features like burger Menu, Offline synchronization, barcode scanner is used to further enhance the quality of the App

5.5 Business Rules

The developer team will be responsible for the correctness of the App. But the users will be frequently querying the NU database and server which comes under the observation of the NU

TCO team. So, it will be a combined effort to minimize any distress and will be the responsibility of both the parties to ensure quality of the App.

6. Other Requirements

All the requirement is mostly covered in the SRS. Many legal policies and requirements of the NU has to be fulfilled to get access to the NU database servers.

Appendix A: Glossary

STUDENTS/USERS : Both refer to the students using the App.

App/Web App: Both refer to the Mobile Application

MD-5 : Encryption

CLI : Command Line Interface

IONIC-2 : Framework to build Web Applications.

HTTP : Hyper Text Transfer Protocol

FTP : File Transfer Protocol

Cordova : Apache plugins for IONIC framework

Firebase : A Google cloud integration (Backend As A Service)

Appendix B: Analysis Models

Analysis model to be explained with the attached use-case diagram