Software Requirements Specification

For

Hybrid App of NU-Gatepass and NU-Library

Version 1.0 approved

Prepared by PRATEEK AGARWAL

NIIT UNIVERSITY

September 18, 2016

Table of Contents

1.Introduction:

1.1 Purpose:

The purpose of this app is to ease the process of applying the gatepass for students and getting approval from warden. With this student will be notified as soon as the warden approves/rejects the gatepass which otherwise student has to check manually by logging to Moodle and so on. In Library section student can search for a particular book and see if its already issued or not. And will be notified about the due date of book to return.

1.2 Scope:

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

In Scope:

- The app will allow students with register college id to access the features of app.
- App will allow students to
    - Apply local fixed, local flexible, outstation gatepass, view recent gatepass history.
    - Notify for due date, search through books available in college library and get their status like issued/not issued.
    - Timetable and attendance for current and previous week only.

Out Scope;

- The app will not allow wardens to login and accept requests.
- Will not allow students to book books from library through app

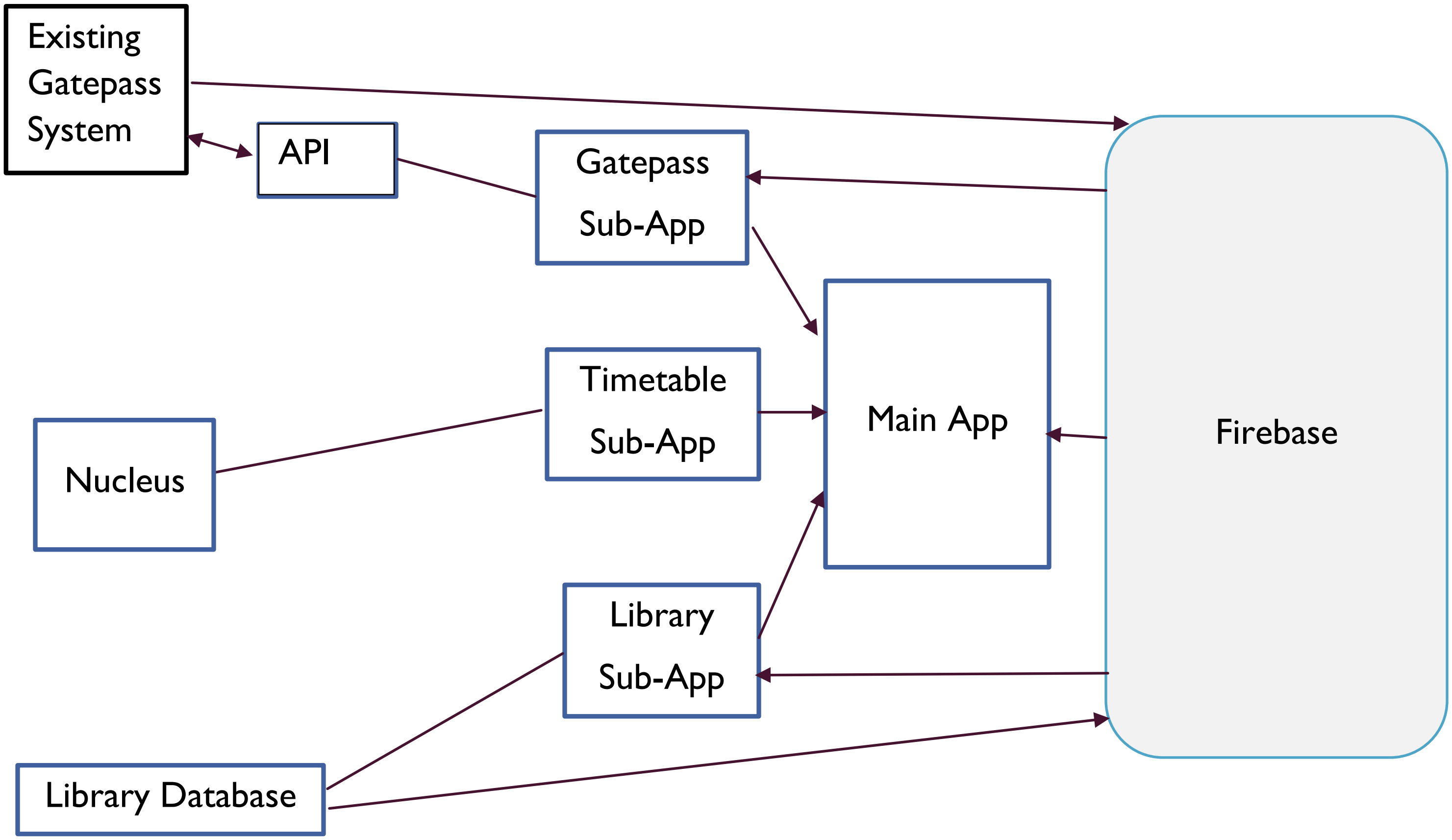
1.3 Intended Audience and Reading Suggestions:

The document is intended to anyone who is interested to know how the system was developed, what all framework used in the app. The rest of the SRS is organized as follows:

Section 2 gives an overall description of the app.

2. Overall Description

2.1 Product Perspective:



2.2 Product function:

- Login student with college id with google Oauth2.0
- Can apply 3 different type of gatepass and view previous gatepass history
- Due date of the library book issued by the student
- Total Due amount till date of the student
- Search a particular book from college library database for availability
- Attendance of current and previous week only
- Timetable of the student

2.3 Operating Environment:

The app we are building is a hybrid cross-platform app which open the app using the web browser present on the smartphone so it is expected to work on all platform but we are focusing it to develop for android above KitKat devices, in apple for IOS >= 6 and windows platform with windows10 phones.

2.4 Assumptions and Dependencies:

The API needed to communicate with our app for the gatepass system is not yet prepared. And moreover the database we are using for this application in gatepass system is not currently used. Besides this On commercial components we may plan to use paid cloud service in Firebase to enable storage and hosting component for our app deployment.

2.5 Design and Implementation constraints:

There can be hardly any constraint on timing and memory requirements. This app will be made available in English language. Further defined constraints include the constraints on DATABASE access on Firebase and NU server access. Our team will be responsible for maintaining the delivered software. Our team will be responsible for maintain the delivered software.

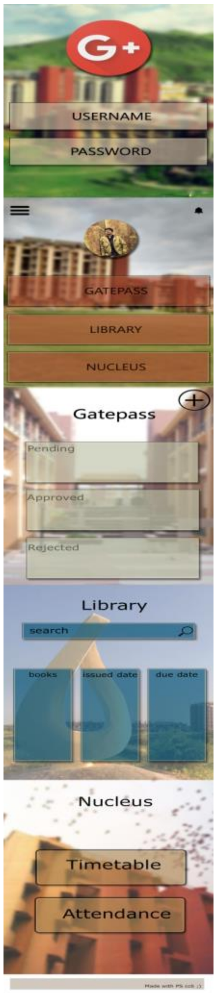
3.External Interface Requirements: -

3.1 Communication Interface:

In this app we are planning to notify student as soon as his gatepass gets approved/rejected for which we will use push notification service provided by the google firebase service(Baas). The API is not yet prepared for the gatepass system therefore we can't confirm like on which protocol it will talk. And that for the Library we will use PDO connection with the database for querying.

3.2 User Interface requirements:

The GUI prepared by our team is just a prototype which can change over time. The sample is attached below: -



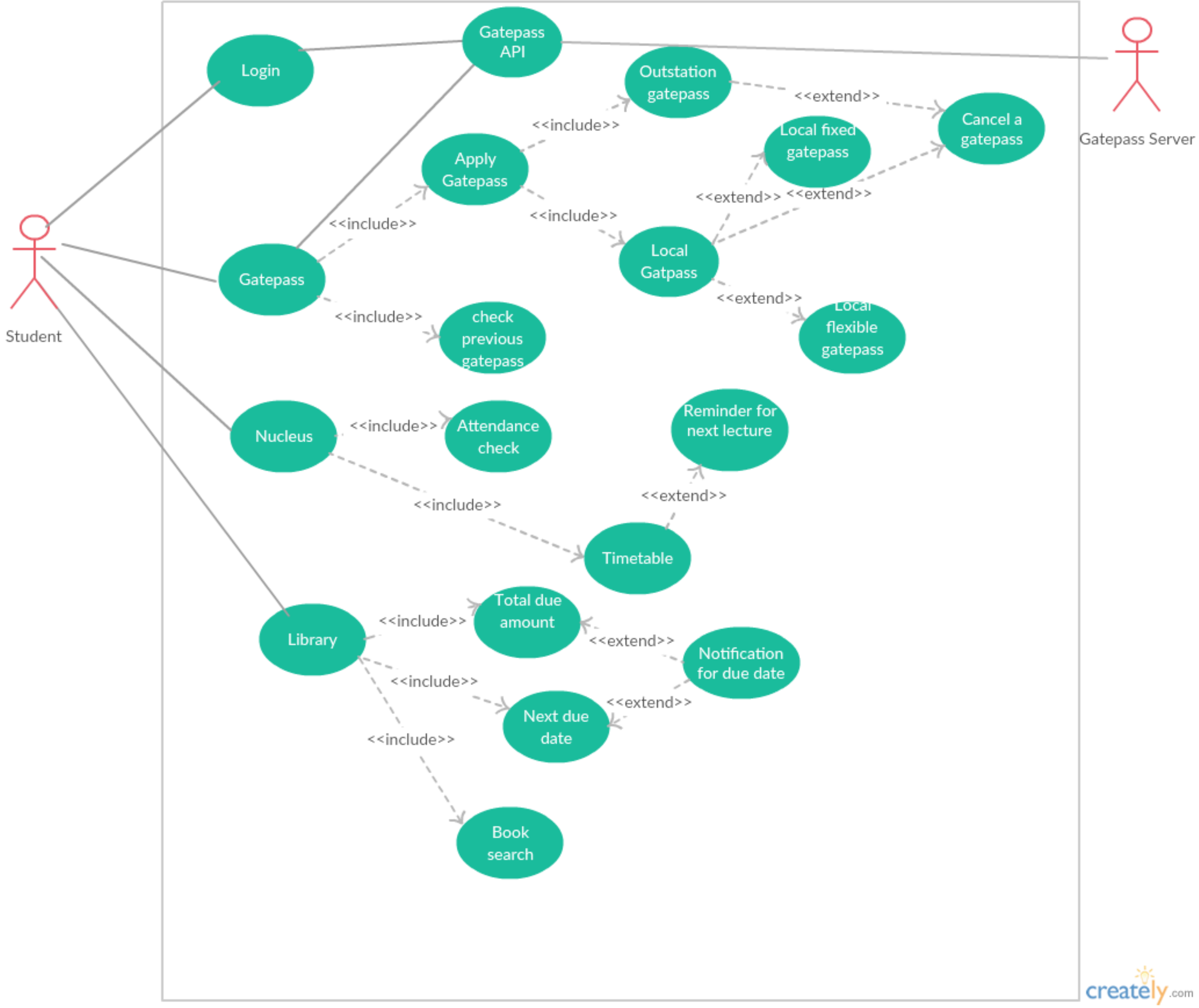
3.3 Software Interface:

As our library we need to import Angular-2 modules and various Cordova plugins for local notifications and all. Data that will come in will be the result of database queries made by API calls and data that moves out includes the API calls to know about the gatepass status, library module and ERP module features. As our BaaS we included Google Firebase to host, store and deploy our web application. As our support we need to fetch data from Moodle and ERP servers.

4. System Features: -

REQUIREMENT NUMBER	SHORT NAME	SHORT DESCRIPTION
1)	Login	Will use google oauth2.0 for logging in the app.
2)	Check previous gatepass	See the history of previous gatepasses applied by the student
3)	Local Gatepass	Enables the user to apply the local Gate pass, which is divided in fixed timing and flexible timing
4)	Local fixed	Enables the user to apply the fixed timing for the local gate pass, which comes under Auto-Approval.
5)	Local flexible	Enables the user to apply the Variable Timing Gate pass, which has to be accepted by the warden
6)	Outstation gatepass	Enables the user to apply for out station gate pass.
7)	Attendance Check	Student can check attendance of current and previous week only.
8)	Timetable	Current semester timetable will be fetched using nucleus api
9)	Cancel gatepass	The user can cancel the order if they are not willing to receive it after payment and give the appropriate reasons for cancelling and proceed to a portal for refund.

10)	Notification	Gives the notification to the user regarding the due date for book issue and also the book check-out.



User Case List:

Use case ID	Use case name
UC1	Login
UC2	Gatepass
UC3	Nucleus
UC4	Library
UC5	Gatepass API
UC6	Apply Gatepass
UC7	Check previous gatepass
UC8	Attendance check
UC9	Timetable
UC10	Notification
UC11	Local Gatepass
UC12	Outstation gatepass
UC13	Local fixed
UC14	Local flexible
UC15	Cancel gatepass
UC16	Book search

5. Other Non-functional Requirements: -

5.1 Performance Requirement:

App should load within 1-2 seconds on data pack. And on college connection data should load in 0.5 seconds.

5.2 Safety Requirements:

As we will directly connect every app with the library database therefore the connection must be secure and proper log of every activity of the student must logged and any suspicious activity should be notified to the admin. We do provide invalid login as safety moves if anyone with his/her entry not present in the database of st. niituniversity.in account, tries to login in our app.

5.3 Security Requirement:

To do so we used Firebase as our login authentication with IONIC-2 framework. This provides securities which makes sure that only authorized users will login successfully and not the other way around.

Appendix A: Glossary

HTTP	Hyper Text Transfer Protocol
FTP	File Transfer Protocol
Firebase	A Google cloud integration as BaaS
CRUD	Create, Retrieve, Update, Delete
CLI	Command Line Interface
Cordova	Apache plugins for IONIC framework
User	The person who operate the software product.
VScode	Visual studio code editor
CSS	Cascading style sheet
TCO	Telecommunication Organization
Koha	Backend on which library system currently works
API	Application protocol interface
Sass	Syntactically awesome style sheet
NU	Niit University

Appendix B: Analysis Models

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