

STUDENT SUVIDHA – A REACT NATIVE APPLICATION

Project report submitted in partial fulfillment of the requirement for the
degree of

BACHELOR OF TECHNOLOGY

in

INFORMATION TECHNOLOGY

By

Rahul Gupta (161465)



UNDER THE SUPERVISION OF

Dr. Aman Sharma

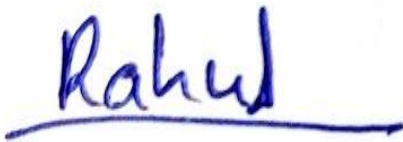
to

Department of Computer Science & Engineering and Information Technology
Jaypee University of Information Technology Waknaghat, Solan-173234,
Himachal Pradesh

CANDIDATE'S DECLARATION

I hereby declare that the work presented in this report entitled “**Student Suvidha – A React Native Application**” in partial fulfillment of the requirement for the award of the degree of **Bachelor of Technology in Information Technology** submitted in the Department of Computer Science & Engineering and Information Technology, Jaypee University of Information Technology, Waknaghat is an authentic record of my own work carried out over a period from January 2020 to June 2020 under the supervision of **Dr. Aman Sharma** (Assistant Professor (Grade-II) Department of Computer Science & Engineering and Information Technology).

The matter embodied in the report has not been submitted for the award of any other degree or diploma.

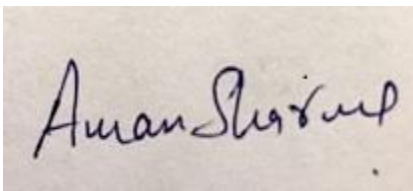


(Student Signature)

Rahul Gupta

161465

This is to certify that the above statement made by the candidate is true to the best of my knowledge.



(Supervisor Signature)

Aman Sharma

Assistant Professor Grade-II

Department of Computer Science & Engineering and Information Technology

ACKNOWLEDGEMENT

First of all, I would like to express my deep gratitude to my project guide **Dr. Aman Sharma** (*Assistant Professor (Grade II), Department of Computer Science Engineering & IT*) for providing me an opportunity to work under his supervision and guidance. His constant encouragement at every step was a precious asset during the project work.

I express my deep appreciation and sincere thanks to **Dr. Hemraj Saini** (*Department of Computer Science Engineering & IT*) for providing all kinds of possible help and encouragement during my project work.

I am thankful to the faculty and staff of *Department of Computer Science Engineering & Information Technology*, Jaypee University of Information Technology for providing me with all the facilities required for experimental work.

I would like to thank my family for their continuous support and motivation. Finally I would like to thank those who directly or indirectly helped me in completing this project.

Rahul Gupta

TABLE OF CONTENTS

CAPTION	PAGE NO.
CANDIDATE'S DECLARATION	I
ACKNOWLEDGEMENT	II
LIST OF TABLE	V
LIST OF FIGURES	VI
ABSTRACT	VII
CHAPTER-1: INTRODUCTION	1-7
1.1 General Introduction	
1.2 Problem Statement	
1.3 Objectives	
1.4 Methodology	
CHAPTER-2: LITERATURE SURVEY	8-19
2.1 React Native	
2.2 More On React Native	
2.3 Node.js	
2.4 NPM: Node Package Manager	

CHAPTER-3: SYSTEM DEVELOPMENT	20-28
3.1 Flowchart Of the Project	
3.2 Development Experience	
3.3 Approach to problem in terms of technology to be used	
3.4 Requirement Specification	
3.5 Design Diagrams	
3.6 The Requirement for Cross Platform	
CHAPTER-4: PERFORMANCE ANALYSIS	29-46
4.1 Complete Description	
4.2 Test plan	
CHAPTER-5: CONCLUSION AND FUTURE SCOPE	47-49
REFERENCES	50
PLAGIARSIM REPORT	51-53

LIST OF Table

S.No.	Description	Page.No.
1	Testing/Debugging	6

LIST OF FIGURES

Figure	DESCRIPTION	Page No.
Fig. 1	Rendering in React Native	10
Fig. 2	React Native Application Architecture	11
Fig. 3	Node.js Architecture	19
Fig. 4	Basic Flowchart Of System	21
Fig. 5	Basic Structure	24
Fig. 6	Use Case Diagram	25
Fig. 7	Welcome Screen	30
Fig. 8	Login Screen	31
Fig. 9	Sign-Up Screen For Student	32
Fig. 10	Sign-Up Screen For Warden	33
Fig. 11	Main Screen For Student	34
Fig. 12	Main Screen For Warden	35
Fig. 13	Lodge Complaint	36
Fig. 14	View Complaint Screen For Warden	37
Fig. 15	Hostel Complaints Screen For Warden	38
Fig. 16	Notices Screen For Student	39
Fig. 17	Wardens Details Screen For Student	40
Fig. 18	Mess Menu Screen For Wardens	41
Fig. 19	Date Sheet Screen For Wardens	42
Fig. 20	Hostel Complaints Screen For Student	43
Fig. 21	Health Complaits Screen For Student	44

ABSTRACT

This project aims to provide a complete framework for any university or college facing problem of communication between the students and wardens .This app aims to focus on solving the problem which are faced by the students and digitalize the whole process of contacting to any hostel worker or wardens . The framework also includes an app made for the students who can contact wardens or any hostel worker from the app. The key point of this project is the idea of react native which gives extra feasibility and attention to this project as it can run on ios as well as on android with no change inside the code.

CHAPTER 1

INTRODUCTION

1. INTRODUCTION

1.1 General Introduction:

Students have to face certain problems in hostels. The problems can be related to hostels like electricity problem, water coolers problem, cleanliness problem, etc., or may be a personal problem like illness, accidental injuries, etc.

Usually, the student has to lodge a complaint in the complaint register which takes a lot of effort and time. Sometimes the complaint gets ignored or takes a long time for it to get catered to.

This application will act as a link between students and college officials. All the complaints lodged by the students will be directly checked by the maintenance officials and the wardens, and action will be taken in the shortest possible time.

This web application will help the students to lodge complaints online, from any place, anytime, and being a web application, it is platform independent (can work on any OS).

Besides providing a facility to file complaints, this application will also be linked directly to the official website of the college. There will be a separate option to open the student's webkiosk directly.

All the official notices will also be displayed in the Notices section of the application.

The Dispensary doctors will also be linked to this application. Sometimes there arise some cases where a student is so unwell that he/she is not even able to leave his/her room and go all the way up to the dispensary. This application will provide a separate section for medical help in which all the contacts of university doctors will be provided and also a description box will be provided to explain the problem. The problems mentioned by the students will be directly seen by the doctor and an ambulance will be sent near the hostel of the student. This will greatly help the students.

Besides all this there will be some more significant features like mess menu, academic calendar, etc which makes student's life easier to some extent.

React native is used in the development of this framework.

REACT NATIVE:

React Native is an open source mobile application structure made by Facebook. It is utilized to create applications for Android, iOS, and Web by enabling designers to utilize React alongside local platform highlights.

React Native is a JavaScript system for composing genuine, natively rendered mobile applications for iOS and Android. It depends on React, the JavaScript library from Facebook for making UIs. Rather than focusing on the program, mobile platforms are focused on. As it were, web engineers would now be able to compose mobile applications that truly look and feel "native". This with the comfort of a JavaScript library that we definitely know and love. Since the vast majority of the code you compose can be shared over various platforms, React Native makes it simple to create it for Android and iOS simultaneously.

Facebook first created React Native to help iOS. With the ongoing help for Android, the library would now be able to render mobile UIs for the two platforms. Facebook has made React Native, its very own Ads Manager application, which incorporates both an iOS and an Android form. The two adaptations were made by a similar improvement group.

React Native spotlights solely on making a mobile interface. It looks like a JavaScript library than a system. The subsequent UI reacts rapidly and feels liquid gratitude to nonconcurrent JavaScript communications with the local condition. This makes sure that app has quicker burden times than an average cross breed application and feels smoother. The key element is the moment reload of the page just as the prompt execution of Android, rolling out the improvements quickly noticeable and the application can be grown quicker.

1.2 Problem Statement:

Problems faced by college students:

- Hostel related problems:-
 - Cleanliness of hostels
 - Electricity related problems

- Water supply related problems
- Nonoperating water coolers
- Repairment of broken furnitures
- Mattress related,broken doors/windows related,etc.

The maintenance department will be in direct contact with the students.The maintenance department officials will have to use this application and will check into the complaints lodged by the students.

Complaints related to all the above mentioned fields can be lodged by the students using this application and the maintenance department officials of the college will be notified directly via this application.The student will have to mention the problem in the description box and will also have to mention the time and date on which the student wants the repairment to be done.According to the availability of the maintenance men the problem will be resolved.

This method of filing complaints is somewhat more efficient and time saving.

- Personal problems:-
 - Illness
 - Injury
 - Lost couriers or posts

All the contact numbers and email addresses of the medical team of the college will be made available to the students and it will be made sure that fastest treatment is provided to the afflicted person.

In case some courier or post recieved by the college gets lost by the college management,the student will be able to lodge a complaint against this issue also.

- Other services:-
 - Mess menu
 - Academic calender
 - Notices
 - Datesheets

All the basic services like academic calender, notices, mess menu, datesheets, webkiosk, etc. will also be provided to the users.

Hence, major or minor, any kind of service needed by the students in the college, will be provided to them.

1.3 Objectives:

- **Study of problem statement:**

A careful study of the problem statement, to make the vision more clear. Noted down all the services that could be made available to the users. Understood the various requirements for the development of the web application.

- **Study of technologies used in the development process of the application:**

This application could have been developed simply in Android, but this would have made the application platform dependent. Since, this application is to be used by users having different OS's, hence it has to be made platform independent.

Using React Native, this can be made possible. React Native provides the flexibility to run the application on any operating system.

Table 1:Testing/Debugging

S. No.	List of various modules that require testing	Types of Testing Required	Technique for writing test cases
1.	Sign Up Address Fetching Mechanism	Unit testing Performance Testing Acceptance Testing	White box testing
2.	Login	Unit testing Acceptance Testing	White box testing
3.	Edit Student Details Sending of details to firebase	Unit testing Performance Testing	White box testing
4.	Edit Complaint Addition of categories to firebase	Unit testing	White box testing

▪ **Deployment:**

After successful testing,the application will be deployed to users.

This application focuses on providing significant services quickly to its users,that are students of the college,hence the name “Student Suvidha”.

This application will provide Suvidha i.e. facility to the users,be it the minor ones like checking various notices and mess menu,or the major ones like lodging complaints easily at anytime and from anywhere.

The aim is to reduce the effort and time invested by the students in such activities.

1.4 Methodology:

Technology has been improving at a very fast pace since last 10 years and new technologies are entering the technical market every year. One such technology has been developed by the facebook organisation which is react native. The power of this language is it can compete with the native languages like swift for ios and android in terms of speed and usability. And above that it uses components just like the web application. It follows the virtual DOM structure which is the most powerful functionality of this language that enables the application to load only relevant data as it changes.

It computes the state of its component whenever they change and updates them accordingly.

The framework that has been built in this project i.e. Student Suvidha Application and it's corresponding user's app uses this framework and function absolutely like that of any android or ios application.

Firebase has been used as a database in this project as it functions most suitably as required by the framework, all the database related changes are updated to the firebase in real time.

So react-native and firebase together make this project stand out of the other projects made on android or ios separately.

CHAPTER 2

LITERATURE SURVEY

2.LITERATURE SURVEY

2.1 React Native:

As this is a new technology, not many research surveys are conducted. However, there are several comparisons that were made by many experts during an online survey. Therefore, the entire React-Native literature is only available online. The key findings of the technology are as follows:

1) - Facebook initially created React Native to help iOS. With the ongoing help for Android, the library would now be able to render versatile UIs for the two stages. Facebook has made React Native, its own Ads Manager application, which incorporates both an iOS and an Android adaptation. The two variants were made by a similar advancement group.

2) - WebView segments of other half and half versatile applications are presently gone. This is conceivable in light of the fact that React Native's structure squares are reusable "local parts" that are arranged legitimately into local. Segments that you use in iOS or Android have reciprocals legitimately in React, giving them a predictable look.

3) - This component-based structure uses a agile approach to build apps.This approach is also more web based.This approach uses no web.The functionality and the speed of the application built with this approach is similar to the native mobile application.

4)-React Native spotlights solely on making a mobile interface. It looks like a JavaScript library than a system. The subsequent UI reacts rapidly and feels liquid gratitude to nonconcurrent JavaScript communications with the local condition. This makes sure that app has quicker burden times than an average cross breed application and feels smoother. The key element is the moment reload of the page just as the prompt execution of Android, rolling out the improvements quickly noticeable and the application can be grown quicker.

5) - The main feature is that the page is reloaded on the fly, as well as the immediate execution of Android, making the changes immediately visible and the app can be developed faster.

2.2 More on react:

Since the arrival of the main advanced mobile phone, the utilization and request of mobile applications has expanded quickly. Numerous organizations have been set up by giving a help through an application however a few organizations need an application so as to demonstrate that they are contemporary or in light of the fact that their rivals have one. This has prompted an immense measure of uses being made yet there is one huge issue, the application should be bolstered by both Android and iOS.

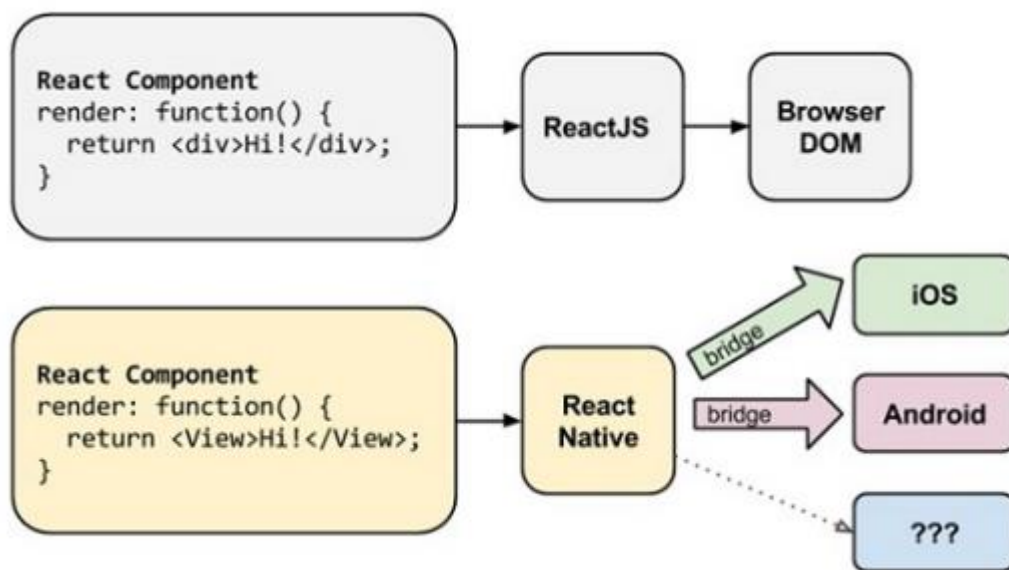


Figure 1: Rendering in React and React Native

Rather than running React in the browser and render divs and writings, React Native runs in an implanted example of JavaScriptCore (iOS) or V8 (Android) inside the applications and render to more significant level stage explicit segments. JavaScript segments are pronounced by utilizing a lot of implicit primitives bolstered by iOS or Android segments.

Features:

There are a few highlights of React Native which makes it much progressively alluring to the advancement network.

React Native handles screen association by actualizing a framework which will handle touches in a complex view with elevated level highlights. Since the gesture recognition is further developed on a cell phone than on the web, there are various activities which can be translated by the touch, for example, looking over, tapping, sliding, etc. In addition, there can be numerous touches being performed simultaneously. The clients see the distinction between a web application and a native since each touch should show what will happen when its discharged and the client ought to have the option to drop the activity by hauling their finger away.

In conclusion, React Native is motivated by React where parts are the building blocks. The comparable methodology applies for React Native as the application is developed by native modules and UI segments. React Native has support for the regular platform APIs

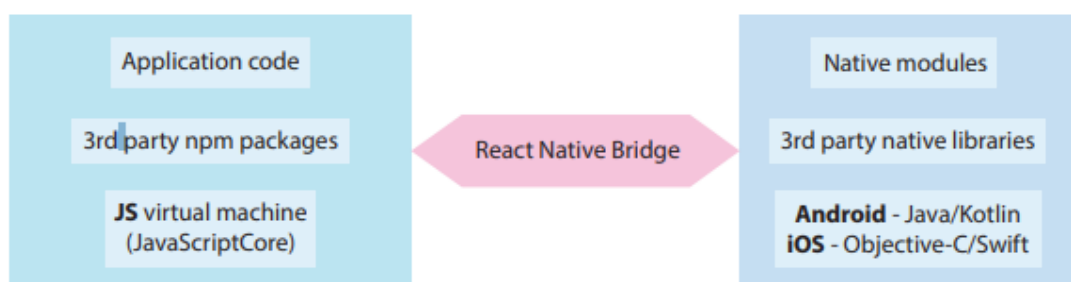


Figure 2:React native application architecture

and native UI segments yet in addition the help to make custom local modules and perspectives on the off chance that it is required.

This permits the utilization of JavaScript to build up the application while as yet using the local UI parts and highlights of the stages.

React Navigation offers various kinds of navigation, for example, stack navigation or tab navigation, for various circumstances.

Native Features:

The vast majority of the basic native highlights are as of now actualized in React Native and can be called from the JavaScript code straightforwardly. Some of the time however the designers need to execute a more mind boggling highlight than the React Native APIs can give, for example, accelerometer following in a foundation procedure.

Cross-Platform Development:

In the course of the most recent years, various strategies and structures have ascended so as to give an answer for making a cross-stage application. The objective is to just create one help which can either be gotten to or sent to various working frameworks and give a homogeneous and native inclination. One mainstream method is making a site, a web application accessible by cell phones using their program. The UI is responsive as the measurements and design of segments are styled in connection to the stature and width of the getting to gadget's screen. This outcomes in a web application with a UI as a standard site when gotten to by a PC however is progressively like an application when gotten to by a portable device. An elective is to utilize a half and half structure so as to make an application that can be utilized by any working framework.

React.js is a JavaScript structure created by Facebook and discharged as open source in 2013 so as to help the advancement network to construct interfaces. React's genuine power lies by they way it makes you to compose code. A decisive programming style portrays what to do yet not how it ought to be done, bringing about less code while a basic programming style anyway depicts how to do it.

2.3 Node JS

What is Node.js?

Node.js is a server-side platform dependent on the Google Chrome JavaScript Engine (V8 Engine). Node.js was created in 2009 by Ryan Dahl and the most recent form is v0.10.36.

Node.js is a cross-platform open source runtime condition for the improvement of server-side applications and system applications.

Node.js likewise gives a rich library of different JavaScript modules that enormously streamlines the improvement of Web applications with Node.js.

The most significant thing about a web server is the capacity to proficiently deal with numerous clients. This has a great deal to do with the programming language with which the content was composed. Along these lines, the presentation of server-side scripting dialects, for example, PHP and Python has been considered contrasted with Node.js.

Node.js is a product platform that gives you a chance to construct your very own web server and fabricate web applications on it. Node.js is neither a web server nor a language. This will at last give you better command over how your web server functions, yet it will build the multifaceted nature of commissioning, particularly in a live situation. For instance, with PHP, you can without much of a stretch locate a mutual server web have running Apache, send a few records by means of FTP, and good, that your webpage is running. This works on the grounds that the webhost has just arranged Apache for you and different clients. This isn't the situation with Node.js on the grounds that you arrange the Node.js server when you assemble the application.

Node.js is a great tool on the off chance that you need a sort of live communication and continuous outcomes. It can convey information to or from a server quick. It was tackled with multithreading quite a while back, which enabled us to have various strings on a solitary CPU. It switches between them constantly and in spite of the fact that it's entirely quick, switching has a great deal of overhead. To maintain a strategic distance from this overhead, node.js takes care of this. Rather than having another thread made on each request, there is a thread for each solicitation. At the point when another one shows up, an

event is raised that executes code. For instance, in the event that you call a database and don't block it until it is returned, essentially pursue a callback function after the call finishes.

Your application code will never run simultaneously. The most significant thing about a web server is that numerous clients can be prepared proficiently. This has a ton to do with the programming language with which the content was composed. Subsequently, the exhibition of server-side scripting dialects, for example, PHP and Python has been considered contrasted with Node.js.

Node.js is an incredible apparatus tool in the event that you need some sort of live association, constant outcomes. It can transmit information rapidly to/from a web server.

Customarily, there has consistently been a major issue with PCs where the processor can just do each thing in turn. This was comprehended quite a while prior with multi-threading, enabling us to have numerous "threads" on a solitary processor. It switches constantly among them, and despite the fact that it is sufficiently quick, the switchover produces an over-burden of time. To stay away from this over-burden, node.js takes care of this issue by running in a solitary occasion driven thread. Instead of making another thread with each request, there is one thread for each request. At the point when another one shows up, it triggers an event that executes code.

At the point when you call a database, for instance, as opposed to obstructing until it is returned, you essentially play out a callback function after the call is done. Everything your program can do, for example, sitting tight for information from a record or an approaching HTTP demand, is dealt with by Node, in parallel, out of sight. Your application code will never be executed simultaneously as whatever else. He will consistently have all the consideration of the JavaScript motor of Node during its activity.

Overwhelming I/O applications profit by this, not at all like CPU-serious applications. Be that as it may, an essential database server requires more I/O, so it's a proficient trade off. Numerous middleware undertakings are I/O related, as are customer side contents and databases. These server-side projects frequently need to hang tight for data, for example, database results, remarks from an outsider web administration, or approaching connection requests. Node.js is intended for precisely this sort of utilization.

Contrasted with numerous clients, the performance of Node.js is better. The performance of Node.js is twice that of PHP and six to multiple times that of Python. In this manner, making it exceptionally synchronous and ongoing.

The decision of the database is significant when thinking about a web server. Instead of a segment characterizing the substance of the line, each line is a record. This archive characterizes and contains the information itself. MongoDB stores reports as BSON, which is a double JSON record. To put it plainly, JSON is a JavaScript approach to store information, which is the reason MongoDB fits so a ways into our JavaScript-driven system.

Advantages of Node JS:

1. Node.js provides easy scalability

One of the key advantages of Node.js is that it's simple for designers to resize applications both evenly and vertically. The size of uses can be changed on a level plane by adding extra nodes to the current framework.

What's more, Node.js gives the capacity to add extra assets to singular nodes as you resize the application vertically. It's exceptionally adaptable and offers a superior choice than other JavaScript servers.

2. Easy to learn

Since JavaScript is one of the most prevalent programming dialects, most front-end designers are great at it.

It will be a lot simpler for you to utilize Node.js in the backend. It's simpler to learn Node.js and invest less energy working.

3. Node.js is used as a programming language

Node.js gives designers the advantage of composing server-side JavaScript applications. Along these lines, Node.js designers can compose both the front-end and essential web applications in JavaScript utilizing a run-time condition.

4. The benefits of Fullstack JS

Node.js was viewed as a full-stack JavaScript for taking care of both customer side and server-side applications.

So the favorable position is that you don't need to set up isolated designers for back-end and front-end improvement. This sets aside important cash and time.

5. The support of a large and active community

Node.js can point to an enormous and dynamic network of engineers who are constantly adding to the improvement and upgrade of Node.js.

Truth be told, the designer bunches are very much bolstered by the JavaScript developers who give instant and basic arrangements and codes in GitHub. It is normal that the engineers will start a lot more designers later on.

6. Offers the freedom to develop applications

Another preferred position that Node.js offers engineers is the opportunity to create applications and programming.

This is a key component that doesn't exist in Ruby on Rails and requires explicit guidelines. When creating applications, you can start from the very beginning once more.

7. Backing for ordinarily utilized devices

With Node.js, engineers can get expanded help for the different usually utilized tools. We should take a model.

In the event that you need to distinguish and introduce venture conditions, you can likewise utilize npm, a ground-breaking bundle supervisor. You can utilize snort to perform venture undertakings.

8. Procedure demands simultaneously

Since Node.js offers the choice of not blocking I/O frameworks, it is moderately simple to deal with various demands at the same time.

The framework can deal with concurrent solicitation preparing more productively than others, for example, Ruby or Python. Approaching requests are adjusted and executed rapidly and reliably.

9. Node.js is very extensible

The Node.js are known to be exceptionally extensible. This implies you can redo and stretch out the Node.js to your needs.

You can likewise utilize JSON to empower the trading of information between the Web server and the customer.

The Node.js is progressively worthwhile for the designers contrasted with its impediments.

After some time, an ever increasing number of corporate associations have presented Node.js and are never again accomplishing positive outcomes.

2.4 NPM: The Node Package Manager

While talking about Node.js, there ought to be no absence of built-in support for package management with NPM, a tool included as standard with each Node.js establishment. The thought behind NPM modules is like that of Ruby Gems: a lot of effectively reachable, openly accessible reusable segments accessible in an online vault with forming and reliance the executives.

For a total rundown of gave modules, see the npm site or the npm CLI device, which is naturally introduced with Node.js. The module biological system is available to anybody and anybody can distribute their own module recorded in the npm storehouse.

Probably the most helpful npm modules today are:

- `express` - Express.js - or essentially Express - a Web advancement system for Node.js dependent on Sinatra and the accepted standard for most Node.js applications at present available.
- `hapi` - an exceptionally particular, simple to-utilize, setup driven system for making web and administration applications.
- `Connect` - Connect is an extensible HTTP server foundation for Node.js that gives an assortment of elite modules called middleware. fills in as a base for Express.
- `socket.io` and `sockjs` - server-side segment of the two most generally utilized Websocket segments today.
- `Pug` (once in the past `Jade`): one of the most well known layout engine, propelled by `HAML` and utilized as a matter of course in `Express.js`.
- `lodash` (`underscore`, `lazy.js`): JavaScript utility. `Underscore` began the game, yet was toppled by one of its two partners, chiefly for better execution and a particular usage.
- `forever` - Probably the most ordinarily utilized utility to guarantee the nonstop execution of a particular node script. Keeps the Node.js procedure running regardless of sudden creation mistakes.
- `bluebird` - A full usage of guarantees/A + with remarkable exhibition.
- `moment` - A JavaScript date library to break down, survey, alter, and design dates.

Examples of Where Node.js Should Be Used

CHAT

Chat is the most ordinary ongoing multi-client application. From IRC (some time ago), through numerous exclusive and open conventions that sudden spike in demand for non-standard ports, to the capacity to execute everything today on Node.js with websockets running on standard 80 ports.

The chat application is a sweet-spot model for Node.js: this application is lightweight, high-traffic, information concentrated (however preparing/processing low) that sudden spikes in demand for circulated gadgets. This is likewise a decent use case for adapting as well, since it is basic, yet covers the greater part of the ideal models you have utilized in a run of the typical Node.js application.

In the most straightforward model, we have a chat room on our site where individuals come and can trade messages in one-to-numerous mode (in reality all). For instance, we have three individuals on the site who are altogether associated with our message sheets.

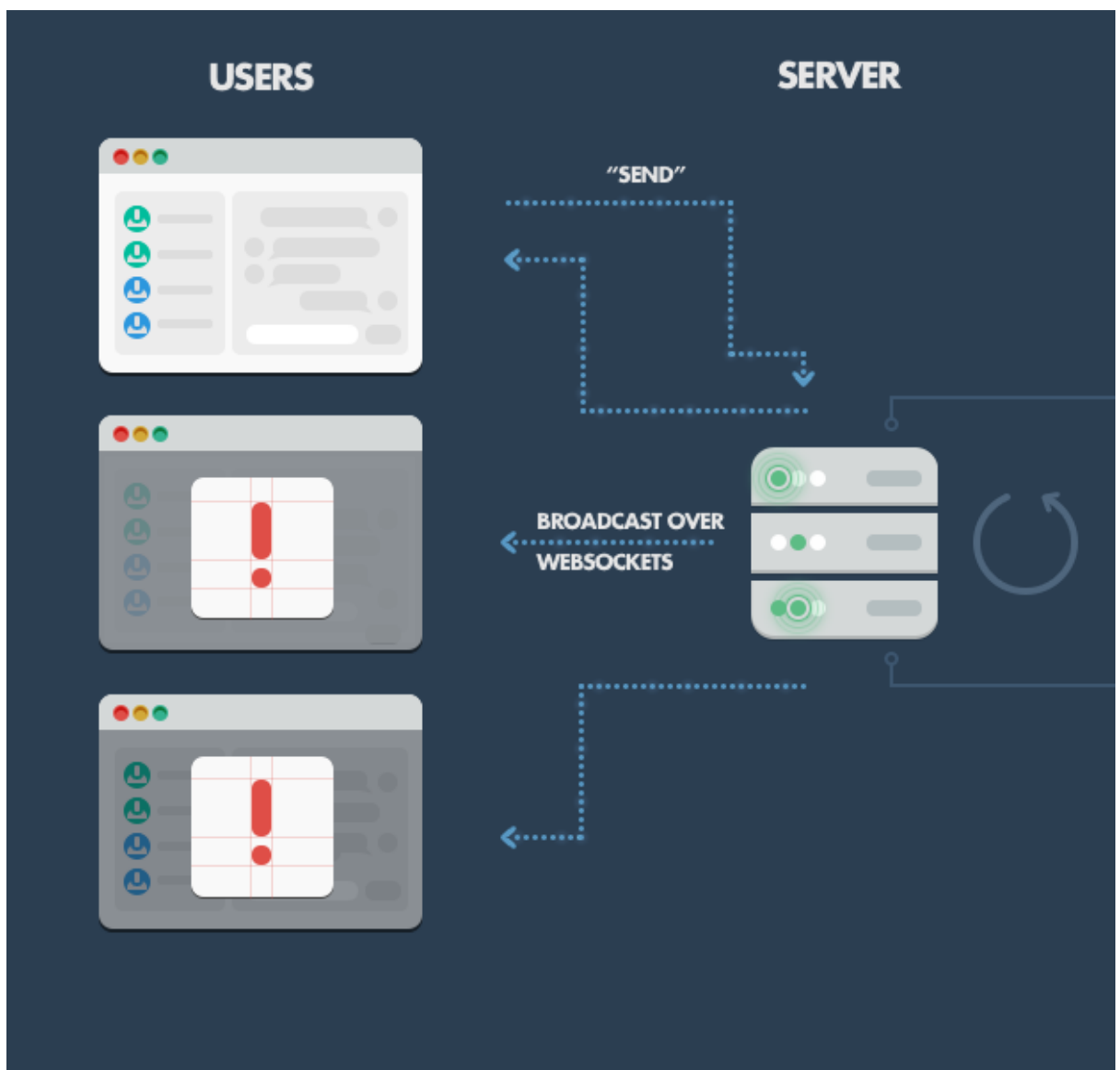


Figure 3: Node.js Architecture

CHAPTER 3

SYSTEM DEVELOPMENT

3. Development

3.1 FLOW CHART:

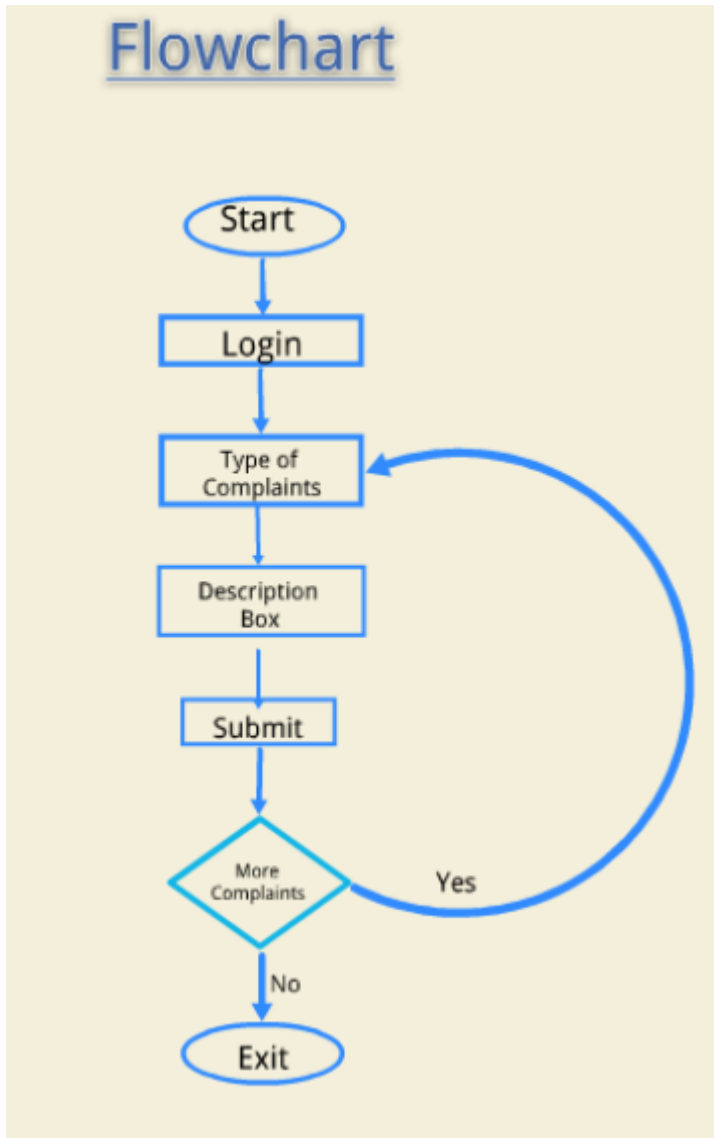


Figure 4: Flowchart Of the System

On starting the application, the user has to first log in using the log in ID and a password.

After logging in, the user will be able to see various options: Complaint lodging, notices, webkiosk, mess menu, datesheet, warden details, etc.

If the user selects the Complaint Lodging option, then next he has to select the type of complaint to be filed. After this a description box will be displayed, where the user will be able to lodge his complaint.

By pressing the submit button the complaint will be lodged and the related authority will be notified regarding the same.

After lodging a complaint, the user will be given an option to file another complaint or go back to the home screen.

3.2 Development Experience:

It is anything but difficult to set up the improvement condition utilizing Expo's devices, and it is similarly simple to get the advancement form of the application to a test gadget or an emulator. Getting the underlying "Hi world" application running is a fast procedure yet introducing the entirety of the supporting npm libraries and getting them to work in the application requires substantially more exertion.

Building up an application with React Native is portrayed by quick prototyping. It appears as though a significant number of the upsides of the system are revolved around making the testing of a refreshed application simpler and quicker. One of the significant selling purposes of React Native and numerous other local scripting structures is live reloading, otherwise called hot reloading. This diminishes the measure of time spent on trusting that a form will finish so as to test another change, yet it likewise supports the engineer towards utilizing an experimentation improvement style where the application is revamped after each little change. This can be a decent or an awful impact of the component.

Facebook and numerous different cross-platform innovation designers showcase their structures with a motto that can be abridged as "adapt once, compose all over". This implies the designer should just need the information on one programming language to compose a mobile application. This case is by all accounts valid at any rate on account of the test application and different utilizations of comparable multifaceted nature. Except if native segments should be created, React Native applications can be composed simply in

JavaScript which may make this innovation speaking to web engineers who are hoping to get into mobile advancement.

Different issues that surface when utilizing outsider libraries are practicality and life expectancy. Apple and Google consistently keep their Source Development Kits (SDKs) fully informed regarding the most recent working framework adaptations, and they discharge engineer sees before a significant working framework update with the goal that the local application designers can stay up with the latest. With cross-platform advances we don't just anticipate that Apple and Google should do this, however we additionally trust the cross-platform creators to stay up with the latest, and we confide in the outsiders to stay up with the latest

3.3 Approach to problem in terms of technology to be used:

First of all there are various different frameworks available in the market beside android and ios which are ionic and phonegap which are also cross-platform frameworks but the approach to working in react-native is because there are not well developed libraries available in other frameworks as they are all third party frameworks but react-native is completely owned by facebook and this makes the framework completely stand out as facebook continuously monitors the issues on many social platforms like stack overflow and github also they keep resolving the issue and making it useful day by day. That makes up choose this framework.

3.4 Requirement Specifications:

Interface – This application associates with the user through Mobile devices. Interface ought to be straightforward, simple to deal with and obvious. Once opened, client will come into stream with application and effectively utilizes every one of these interfaces legitimately.

Equipment Interface –

1. 2 GB Ram
2. Andorid or ios Smartphone
3. Internet connection.

Program interface -

1. Sublime is a word processor and source code editorial manager and provides an environment for creating HTML, CSS, node.js, JavaScript, jQuery and many different editing purposes.
2. Working framework, (for example, windows 7, 8, 10, Mac OS or Linux).
3. Node.js - It is a step based on the JavaScript Chrome rhythm to create fast and adaptable framework applications. Node.js uses an I / O show that is not blocked by an event that makes it light and talented, suitable for information of serious ongoing applications that keep running over each other over scattered gadgets.
4. Firebase-Firebase is used as a database in this application. Through this we can save and update the data in real time.

Functional Requirements:

Remote Client: - The Remote Client will be an android or IOS mobile device through which user can lodge a complaint.

Server: Server authorizes the user to check for its and account and successively takes an account of the complaint of the user.

Components Used:

We are using react native which is used as a UI framework in this project and with the help of node js we can connect to the firebase database and perform the READ, WRITE, DELETE and UPDATE functionalities on the database.



Figure 5:Basic Structure

3.5 Design Diagrams:

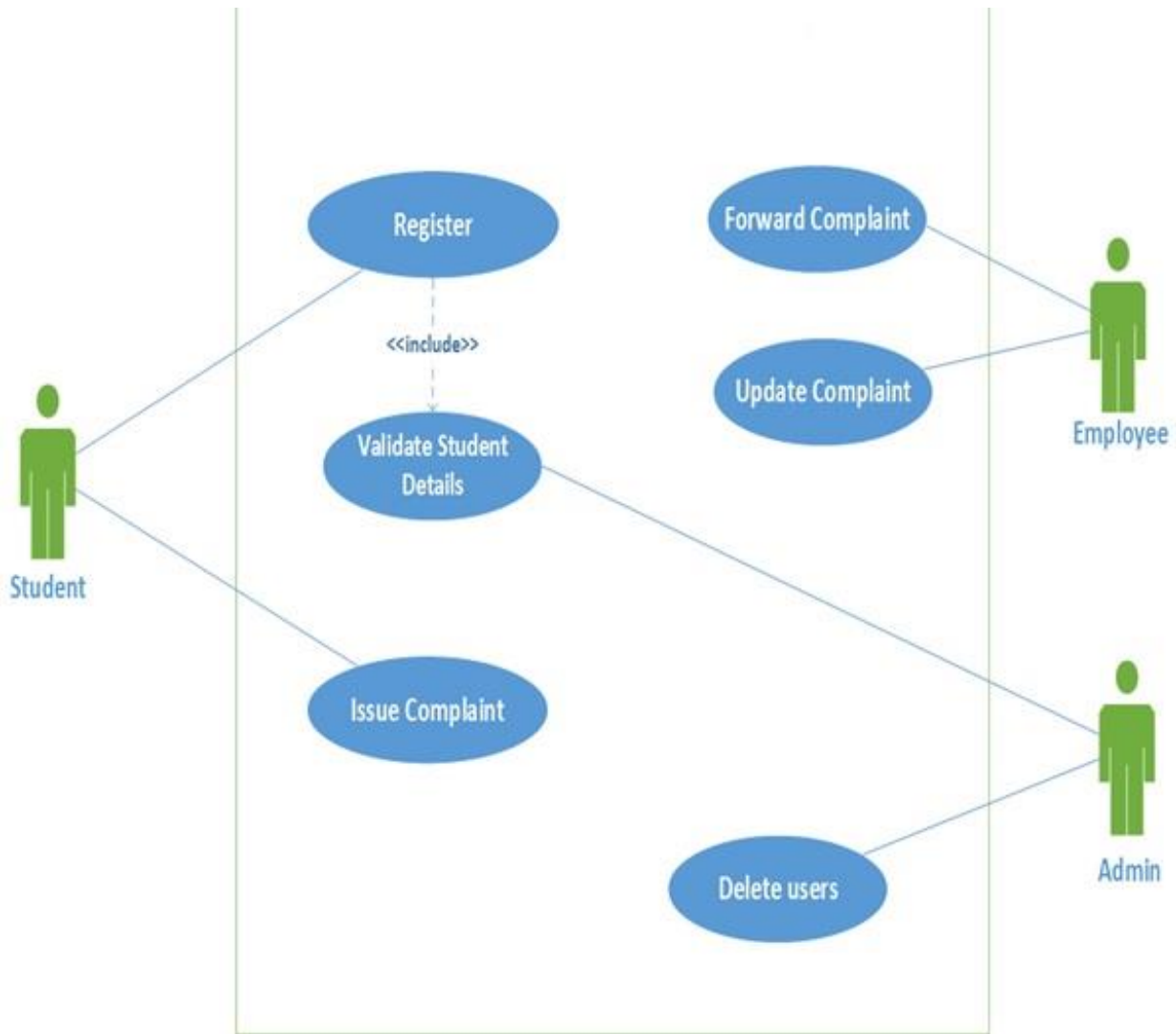
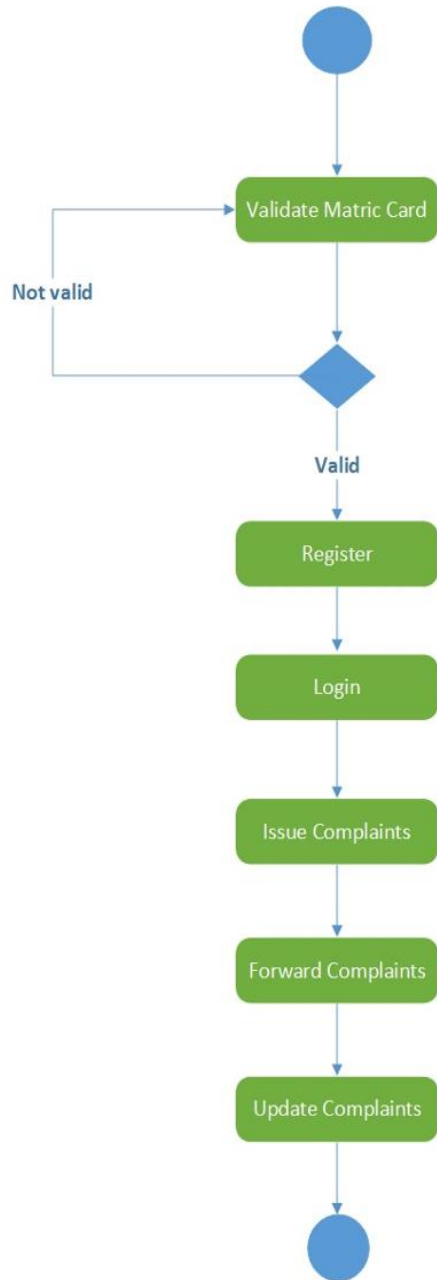


Figure 6:Use Case Diagram



3.6 The Requirement for Cross-Platforms

The primary motivation behind cross-platform is to create mobile applications for multi-platform, however it is critical to consider the prerequisites expected to deliver and keep up superior mobile applications. In light of our own writing and investigation, we have recognized the ideal necessities of each cross-platform innovation introduced as follows:

A. Scalability and Application Maintenance

The cross platform must give the likelihood to keep up and improve the application. For instance, if another choice should be executed in an application, change should consistently be at a cross platform level and after that it is utilized in an alternate platform.

B. Access to Device Features

Applications made utilizing cross platform must have all access to gadget highlights. It is favored that this access is immediate without a transitional layer.

C. Resource Consumption

Cross-platform must consider the utilization of assets including CPU, control utilization and memory use. It is significant for designers to consider the utilization of account resources, yet cross-platform must offer greatest programmed enhancement for mobile resources in deployment for every platform. Various examinations have been completed about this.

D. Security

Security issues are one of the most significant criteria in the improvement of mobile phones. A portion of these issues can be comprehended by the engineer. Designers must focus on which information is put away on the gadget, on the grounds that the gadget can be taken or fall into unapproved hands. Be that as it may, cross-platform must have the ability to regard the security arrangements set by the designer on every single distinctive OSs.

E. Environmental Developement

The advancement condition is a significant certainty in cross platform. This is the reason we need to incorporate all the required debuggers, compilers, and wise computerized consummation frameworks. Be that as it may, in particular, we have to have a few test systems with various frameworks (Android, iOS, Windows Phone and so forth ...) to get great perceivability on the application before organization.

CHAPTER 4

PERFORMANCE ANALYSIS

4.1 Complete Description:

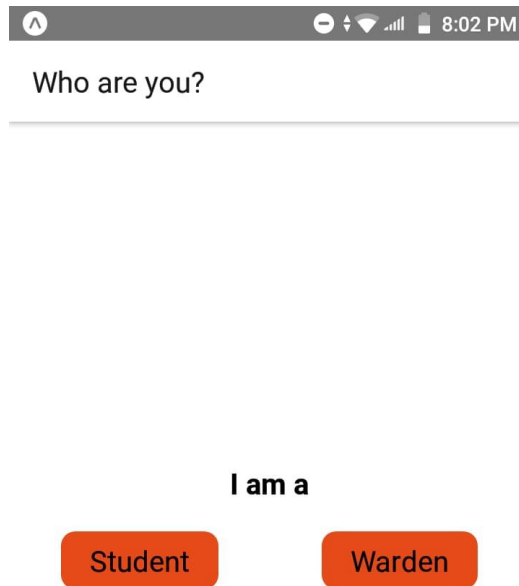


Figure 8:Welcome Screen

- This will be the first screen that would be displayed in front of our user.
- Here, the user has to choose between the type of user i.e. student or warden, then he/she will directly log in to the application using Log in ID(Roll No.) and password.

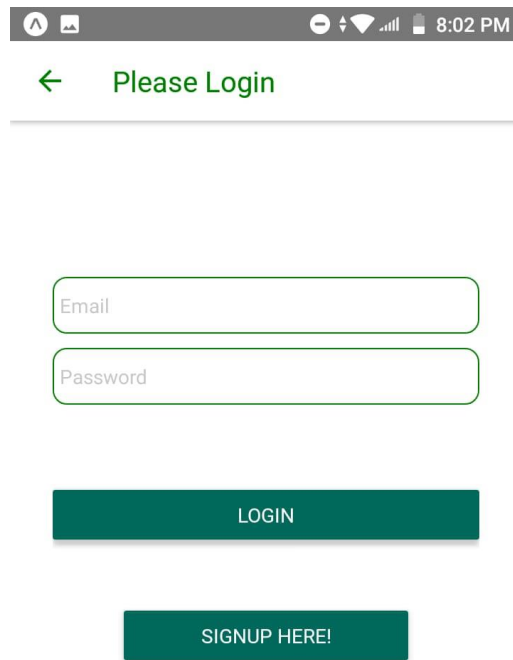


Figure 9:Login Screen

- This will be the second screen that would be displayed in front of our user.
- The Login screen is the same for both the users.
- Here, if the user is already registered to this app, then he/she will directly log in to the application using Log in ID (Roll No.) and password.
- If the user is not already registered to this application then he/she will make a new account using the sign up option.
- While logging in, if the user types a wrong password or ID then an error will occur, and an error message will be popped up.

8:54 PM

← Please SignUp

ID

Name

Contact Email

Password

Hostel/Room

SIGNUP

LOGIN HERE!

Figure 10:Sign-Up Screen For student

- This is the sign up screen,where the student user can make a new account.
- Necessary details need to be filled up here.
- These details will be saved in the database:Firebase.
- Once sign up button is clicked,the new account will be created successfully

8:02 PM

← Please SignUp

Name

Contact Email

Mobile

Password

Hostel

SIGNUP

LOGIN HERE!

Figure 11: Sign-Up Screen For Wardens

- This is the sign up screen, where the Warden user can make a new account.
- Necessary details need to be filled up here.
- These details will be saved in the database: Firebase.
- Once sign up button is clicked, the new account will be created successfully.

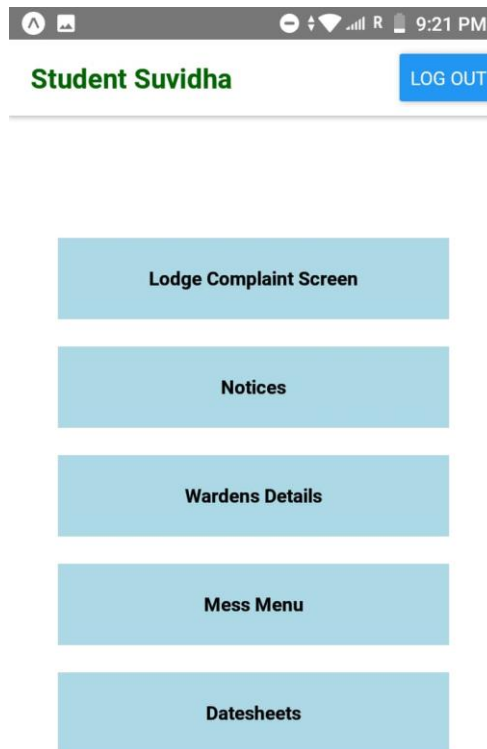


Figure 12:Main Screen For Student

- This will be the main screen after the user is logged in to the application successfully.
- There is the Logout Button on the top right where the user can logout.
- Here,in this screen,the user can see several services options:
 - Lodge Complaints
 - Notices
 - Warden Details
 - Mess Menu
 - Datesheets

Any new notices circulated by the University will be displayed in the notices section.

The contact numbers and emails of all the wardens will be displayed in the Warden Details section.Mess Menu and Datesheets can also be seen using this application.

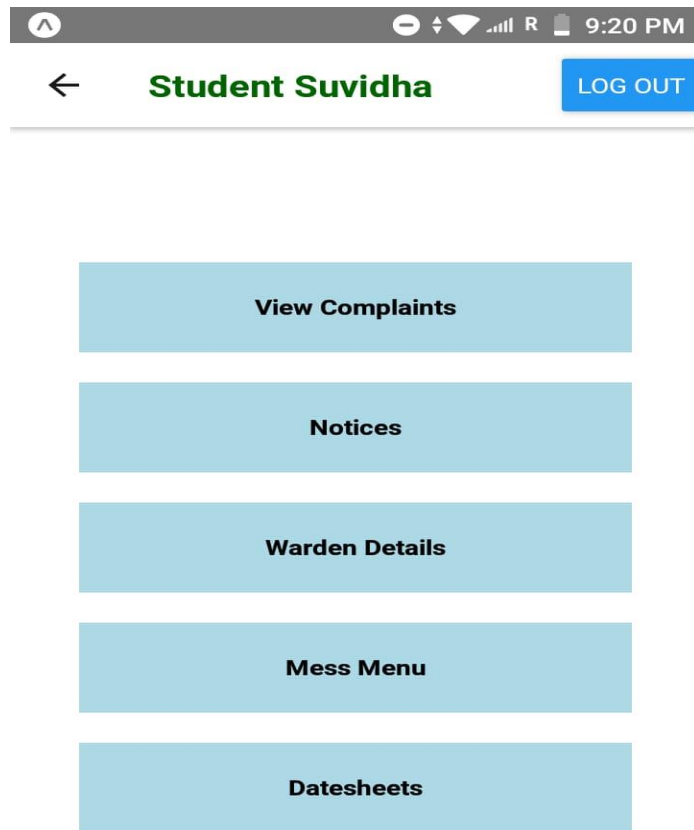
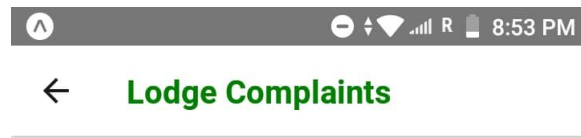


Figure 13:Home Screen For Warden

- This is the home screen after the warden is logged in to the application successfully.
- There is the Logout Button on the top right where the user can logout.
- Here,in this screen,the user can see several services options:
 - Viwe Complaints
 - Notices
 - Warden Details
 - Mess Menu
 - Datesheets

Any new notice circulated by the University will be updated updated by the wardens.

The contact numbers and emails of all the wardens will be displayed in the Warden Details section.Mess Menu and Datesheets can also be updated using this application.



Select the type of Complaint



Figure 14:Lodge Complaint

- If the student wants to lodge any complaint,then he/she will select the Lodge Complaints option.
- On selecting the Lodge Complaint option,the above pasted screen will appear in front of the user.
- Here the user has to select which type of complaint he/she wants to lodge.
- Hostel related complaints will be notified to the maintenance department of the college and health related will be notified to the medical department.

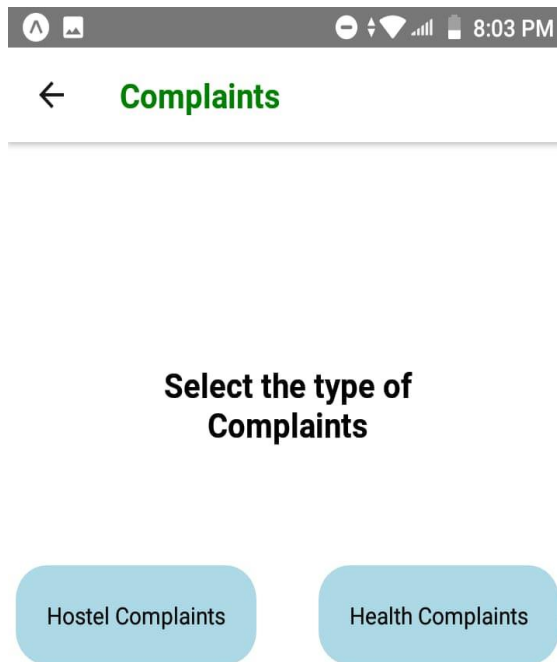


Figure 15:View Complaint screen for Wardens

- If the warden wants to view any complaint,then he/she will select the View Complaints option.
- On selecting the View Complaint option,the above pasted screen will appear in front of the user.
- Here the user has to select which type of complaint he/she wants to View.
- Hostel related complaints will be notified to the maintenance department of the college and health related will be notified to the medical department.

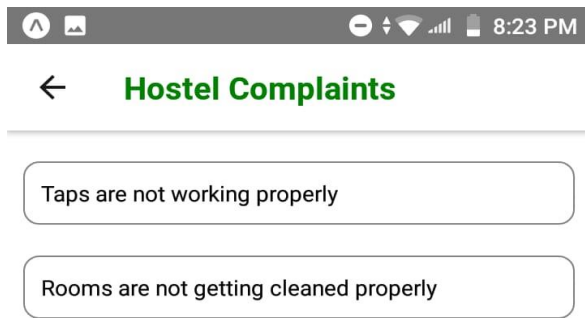


Figure 16: Hostel Complaint screen for Wardens

- Here ,if the warden wants to view any complaint,then he/she will select the View Hostel Complaints
- On selecting the Hostel Complaint option,the above pasted screen will appear in front of the user.
- Here the warden can watch all the complaints by the students.
- Similarly there is the screen for all the health complaints.

← Notices

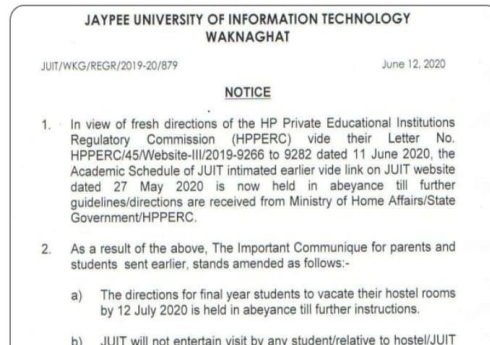
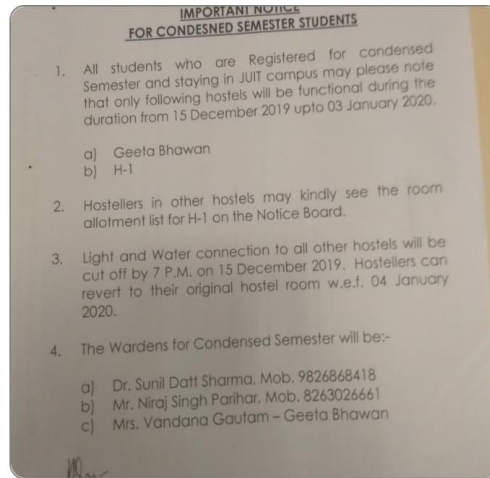


Figure 17: Notices screen for Students

- Here the students can view all the notices that are updated by the University.
- They can click on the particular notice and view it on the full screen.

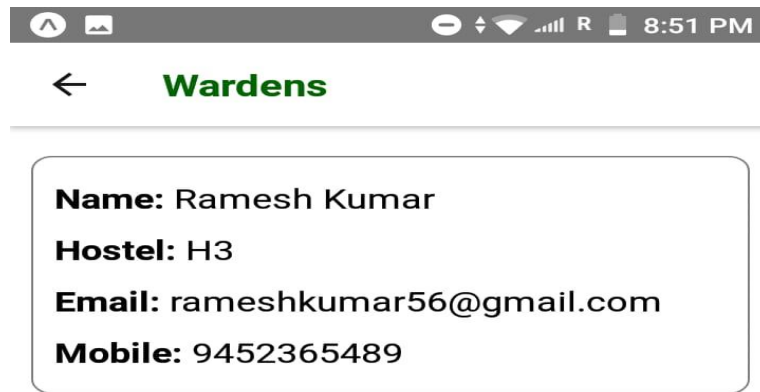
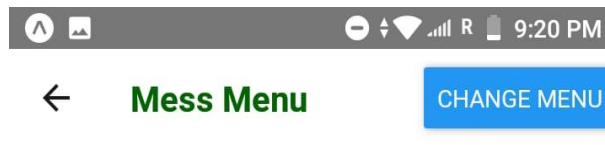


Figure 18:Warden details screen for Students

- Here the students can view all the wardens related information.
- They can click on the particular Warden detail screen and view it on the full screen.
- The Wardens can update their Information which will simultaneously be reflected to the students screen.
- In this way students will be having an access to all the details of the wardens which includes their contact no. , email and hostels.



JUIT-WAKNAGHAT
ANNAPURNA (A) WEEKLY MENU FOR JANUARY-2020

DAYS	BREAK FAST	LUNCH	DINNER
Monday	BREAD, BUTTER, JAM, UPMA & DOORUT CHUTNEY, BREAD ROLL / ALDO BANDA, HOT COFFEE	ALDO DMBH, KAGI, RICE, ROTI, TANDURI / PLAIN, SALAD (ONION, KHEERA) KIKKI	ALDO MATAR, MAH SAGUT, DAL, RICE, ROTI TANDURI / PLAIN, SALAD, GREEN CHUTNEY, SWEET DISH - GAJAR KA HALWA
Tuesday	BREAD, BUTTER, JAM, SPROUT SOPHRE BIKES WITH MILK, DAL, PRAKTHA & TOMATO CHUTNEY, TEA	ALDO GAJAR MATAR, RAJMA, RICE, ROTI TANDURI / PLAIN, PLAIN CURRY IN CUP, SALAD MULLI, GAJAR, APPLE	ALDO TOMATO GRAVY, CHOLLY MATAR, VEG BIFYAN, ROTI, TANDURI / PLAIN, MIXED CORN, SALAD, PAPAD SWEET DISH - FRUIT CUSTARD, HOT MILK
Wednesday	BREAD, BUTTER, JAM, POHA, OUTLET, BOILED EGG, TEA, BANANA	ALDO PALAK BHUNIA, CHOLLEY MATAR, PULAO, ROTI, DAHI, MIRA WITH DENTH, CHUTNEY, KACHUMBER SALAD.	PANKEER, BHURSI, DAL URAD CHANA, RICE, ROTI, TANDURI / PLAIN, SALAD, ONION, KHEERA, TOMATO, PICKLE SWEET DISH - KICORIG DAL, HALWA, HOT MILK
Thursday	BREAD, BUTTER, JAM, VEG, SANDWICH, PASTA, HOT MILK	ALDO BHUNIA, MIRCH, RAJMA, RICE, ROTI, TANDURI / PLAIN, BUNDI, RAITA, SALAD, ONION, MOOJI, TOMATO, ORANGER	ALDO MATAR, DAL, MASUR, SABUT, RICE, ROTI, TANDURI / PLAIN, SALAD (MOOJI, GAUR, ONION) GREEN CHUTNEY, SWEET DISH - RICE, KHEER, HOT MILK
Friday	BREAD, BUTTER, JAM, SPROUT, STUFFED PRAKTHA & CURD, PICKLE, CHALLETTE, TEA, BANANA	MITHI ALDO DAL, MOONG, MUKKA, VEG, PULAO, ROTI, TANDURI / PLAIN, CURRY IN CUP, MIXED SALAD (GAJAR, MOOJI, ONION)	PALAK PANKEER, KALA CHANA, RICE, ROTI, TANDURI / PLAIN, SALAD, SWEET DISH - SOOJI, HALWA, HOT MILK
Saturday	BREAD, BUTTER, JAM, DALIA, MAGGI, KACHORI, ALDO SUBI, HOT COFFEE	LAUKI CHANA, DAL, MOONG, SABUT, BUNDI, BAITA, RICE, ROTI, TANDURI / PLAIN, SALAD, KACHUMBER, PICKLE, KIKKI	SARISO, SARISO, DAL, ATJHAR, RICE, MISSI, ROTI, PLAIN / TANDURI, SALAD, SWEET DISH - GULAB JAMUN, HOT MILK
Sunday	BREAD, BUTTER, JAM, SPROUT, SANDWICH, BURGER, HOT COFFEE, BANANA	NOODLE, FRIED RICE, VEG, MANCHURIAN, HOT AND SOUR SOUP, PAPAD	MIX VEG, EGG CURRY, URAD CHANA, DAL, RICE, ROTI, TANDURI / PLAIN, SALAD (ONION, GAJAR, MOOJI, GREEN CHUTNEY) SWEET DISH - SEMIYA KHEER, HOT MILK

Milk Distribution

Figure 19: Mess Menu screen for Wardens

- Here the Wardens can Upload the mess menu for the students .
- When the Mess menu is uploaded to the server , at the same time the students Can view the newly updated mess menu .
- The Mess menu will only be accessible to those students whosoever is using the Student Suvidha Application.
- The students cannot change the Mess Menu , the change menu functionality is only for wardens to change the menu.

Jaypee University of Information Technology, Waknaghat
Test 3 (19-06-2020 to 21-06-2020)
B.Tech VII Semester

Date	Time	ECE	CSE / IT	Civ	Biotechnology/Bioinformatics	Dist Degree Biotechnology	
19-06-2020	9.30 a.m -10.30 a.m	Network Embedded Control Systems 18B19WCE01 Fundamentals of Neural Computation Communication System 18B19WCE04 Cognitive Radio Networks 18B19WCE03	Information: Pattern and Data Mining 18B19WCE02 Deep Learning 18B19WCE05 Facial Image System Design 18B19WCE02 Software Reliability 18B19WCE04Advanced Operating System 18B19WCE02	Digital Engineering 18B19WCE1202+2004+4+4 Design of Structures 18B19WCE1301+302+303+304 Structural Engineering 18B19WCE11	Virtual Disease 18B19WCE0400 BIOBIOINFORMATICS BIOBIOINFORMATICS BIOBIOINFORMATICS	Virtual Disease 18B19WCE0400 BIOBIOINFORMATICS BIOBIOINFORMATICS	
19-06-2020	9.30 a.m -10.30 a.m	Art Computing Techniques 18B19WCE01	Network Management 18B19WCE03 Advanced Software Engineering 18B19WCE02 Formal of Temp Architecture and Design 18B19WCE02 Formal Temp Architecture, Protocols and Applications 18B19WCE01 Computer Case 18B19WCE04	Process Control Structure 18B19WCE01 Industrial Waste Management 18B19WCE02 Waste Management 18B19WCE02	Genetic Computing 18B19WCE04 Molecular Evolution 18B19WCE04	Genetic Computing 18B19WCE04 Molecular Evolution 18B19WCE04	
20-06-2020	9.30 a.m -10.30 a.m	UNDERSTANDING INDIA: LITERARY REPRESENTATIONS 18B19WCE01 / CONTEMPORARY INDIA IN GLOBALISED ERA: CHALLENGES OF DEMOCRACY AND DEVELOPMENT 18B19WCE01 / RHYTHM, BUSINESS AND INDUSTRY 18B19WCE02 / INTERNATIONAL HUMAN RESOURCE MANAGEMENT 18B19WCE02 / INTERNATIONAL BUSINESS MANAGEMENT 18B19WCE02					
21-06-2020	9.30 a.m -10.30 a.m	Modern Antennas 18B19WCE02 Optical Communication Systems 18B19WCE04QUANTUM EFFECTS IN SEMICONDUCTOR PHYSICS 18B19WCE04	Cloud Computing 18B19WCE01 Advanced Algorithms 18B19WCE01 Design & Analysis of Multidimensional Data Structures 18B19WCE01 Advanced Computer Architecture 18B19WCE02	Highway Construction, Maintenance and Management 18B19WCE01 Waste Quality Management 18B19WCE01 Process Design in Environmental Engineering 18B19WCE04	Bi-resources & Industrial Products 18B19WCE04 Case Analysis & Applications 18B19WCE04 Biotechnology 18B19WCE01	Bi-resources & Industrial Products 18B19WCE04 Case Analysis & Applications 18B19WCE04 Biotechnology 18B19WCE01	

In case of any discrepancy contact the office of Controller of Examination (181901794)

CONTROLLER OF EXAMINATION

Figure 20:DateSheet screen for Wardens

- Here the Wardens can Upload the datesheet for the students .
- When the Date Sheet is uploaded to the server , at the same time the students Can view the newly updated date sheet .
- The Date Sheet will only be accessible to those students whosoever is using the Student Suvidha Application.
- The students cannot change the date sheet , the change menu functionality is only for wardens to change the date sheet.

← **Hostel Complaint**

Type your complaint in the description box below :

Description

Submit

Figure 21: Hostel Related Complaints

- If the student chooses to lodge the hostel related complaint, then the above pasted screen will be displayed.
- Here, in the description box the user will be able to type the complaint.
- By pressing the 'Send' button the complaint will be lodged.
- The maintenance department of the college will be notified as soon as the complaint is lodged.

The screenshot shows a mobile application interface for logging a health complaint. At the top, there is a status bar with icons for signal strength, Wi-Fi, and battery, along with the time 8:53 PM. Below the status bar is a navigation bar with a back arrow and the title "Health Complaint" in green. The main content area contains the instruction "Type your complaint in the description box below :". Below this is a large, empty text input field with a light green border and the placeholder text "Description". Underneath the input field is a toggle switch labeled "Do you want ambulance to pick you up?" in red text. At the bottom right of the form is a blue "Submit" button.

Figure 22:Health Related problem

- If the student chooses to lodge the Health related complaint, then the above pasted screen will be displayed.
- Here, in the description box the user will be able to type the complaint.
- By pressing the 'Send' button the complaint will be lodged.
- The medical department of the college will be notified as soon as the complaint is lodged.
- The user is also given the option to choose if he/she wants the ambulance to pick him/her up from the hostel or not.

4.2 Test Plan:

Unit testing is a major order that can prompt a 40-80% decrease in the density of creation bugs. The unit test additionally has a few other significant points of interest:

- Improve the design and practicality of your applications.
- Improves APIs and similarity by concentrating the designer on the engineer's understanding (API) before the subtleties of the usage.
- Provide snappy criticism on record stockpiling to fill you in as to whether your progressions were effective or not.
- Provides an amazing wellbeing net that can expand your self-assurance when it's a great opportunity to include highlights or upgrade existing highlights.

Be that as it may, a few things are simpler than others for unit testing. Specifically, unit tests work very well for unadulterated capacities: works that give a similar info, consistently return a similar yield and have no side effects.

Frequently, UI segments are not part of the simple to-utilize thing classification for unit testing, which makes adherence to the TDD discipline increasingly troublesome: first compose the tests.

Composed tests are expected to accomplish a few advantages: improved engineering, better designer experience, and quicker input when you create applications. It takes teach and practice to work on utilizing TDD. Numerous engineers want to tinker before composing tests, however on the off chance that you don't compose the main test, you strip the best highlights of the unit test.

Be that as it may, this merits the training and control. TDD with unit tests can instruct you to compose a lot less difficult UI segments that are simpler to oversee and sort out and reuse with different segments.

Despite which system you use, the accompanying tips will assist you with composing better, testable, clear, and composable UI segments:

- Prefer unadulterated segments for the UI code: Always render a similar segment with similar props. On the off chance that you need a status from the application, you can wrap these unadulterated parts with a compartment segment that deals with the status and reactions.
- Isolate application rationale/business administers in unadulterated decrease capacities. Isolate reactions with compartment segments

CHAPTER 5

CONCLUSIONS

5. Conclusion

Through this project we have successfully checked the benchmarks of react-native, react-native works perfectly well on both platforms android as well as ios. Though there are several changes going on in this language as there are several issues emerging out everyday on github and stack overflow, react-native developers are continuously monitoring the issues and keep updating the libraries. We faced many challenges while building the app as one major library name react-navigation updated completely on which our project was wholly dependent. But eventually we were able to track the updates and reflected the same in our project.

Future Perspective:

This cross platform application saves the effort and time of the students. This web application will help in keeping things organised and simple, not only for the students but for the college maintenance authorities also. Students will be able to lodge complaints anytime and from anywhere. They need not to go to the maintenance department and file the complaints manually. Hence this application makes this process simpler.

Besides enabling the students to lodge complaints, this application provides many other services also.

Usually the students have to regularly check the notice boards in the college. There are times when the students miss out some important notices. But this application gives a separate section for notices. Everytime a new notice is issued by the administration, the students linked with this application will get notified and the chances of missing out any notice will be reduced.

The Dispensary doctors will also be linked to this application. Sometimes there arises some cases where student is so unwell that he/she is not even able to leave his/her room and go all the way up to the dispensary. This application will provide a separate section for medical help in which all the contacts of university doctors will be provided and also a description box will be provided to explain the problem. The problems mentioned by the students will be directly seen by the doctor and an ambulance will be sent near the hostel of the student. This will greatly help the students.

References

1. React Native, "This page will help you install and build your first React Native app", <https://facebook.github.io/react-native/docs/getting-started.html> , 2019
2. Expo, "Mobile App Development Tool" <https://docs.expo.io/versions/latest/sdk/location> , 2019
3. React JS, "Introduction To JSX Properties" <https://reactjs.org/docs/introducing-jsx.html> , 2019
4. React Navigation, "React Navigation Stack" <https://github.com/react-navigation/react-navigation/issues/288>, 2019
5. Firebase, "Iterating Through Multiple Children In Firebase", <https://stackoverflow.com/questions/40824060/iterating-through-multiple-children-infirebase>, 2019
6. Nodejs , "Nodejs Package Manager " <https://www.npmjs.com/package/react-native-simple-toast> , 2019
7. React Native, "State and Lifecycle " <https://reactjs.org/docs/state-and-lifecycle.html> , 2019
8. React JS , "Thinking In Reactjs Library" <https://reactjs.org/docs/thinking-in-react.html> , 2019

Plagiarism

ORIGINALITY REPORT

7%

SIMILARITY INDEX

1%

INTERNET SOURCES

0%

PUBLICATIONS

6%

STUDENT PAPERS

PRIMARY SOURCES

1

Submitted to Federation University

Student Paper

2%

2

Submitted to Asia Pacific University College of
Technology and Innovation (UCTI)

Student Paper

<1%

3

Submitted to CSU, San Jose State University

Student Paper

<1%

4

Submitted to Melbourne Institute of Technology

Student Paper

<1%

5

Submitted to University of Wales Institute,
Cardiff

Student Paper

<1%

6

www.tutorialspoint.com

Internet Source

<1%

7

Submitted to University of Wolverhampton

Student Paper

<1%

8

Submitted to Thames Valley University

Student Paper

<1%

9	etd.lib.metu.edu.tr Internet Source	<1 %
10	Submitted to University of Surrey Student Paper	<1 %
11	Submitted to Higher Education Commission Pakistan Student Paper	<1 %
12	Submitted to University of Bedfordshire Student Paper	<1 %
13	Submitted to University of London External System Student Paper	<1 %
14	Submitted to Glasgow Caledonian University Student Paper	<1 %
15	theses.gla.ac.uk Internet Source	<1 %
16	Submitted to University of Warwick Student Paper	<1 %
17	sharkoonlive-en.netapps.de Internet Source	<1 %
18	Submitted to Academy of Information Technology Student Paper	<1 %
19	Submitted to University of Sheffield	

Student Paper

<1%

20

Submitted to Heriot-Watt University

Student Paper

<1%

21

Submitted to University of Greenwich

Student Paper

<1%

Exclude quotes On

Exclude matches Off

Exclude bibliography On

JAYPEE UNIVERSITY OF INFORMATION TECHNOLOGY, WAKNAGHAT

PLAGIARISM VERIFICATION REPORT

Date: 16-07-2020.

Type of Document (Tick): PhD Thesis M.Tech Dissertation/ Report B.Tech Project Report Paper

Name: _Rahul Gupta_ Department: _CSE_ Enrolment No _161465_

Contact No. __8219232142__ E-mail. _dengre4@gmail.com__

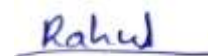
Name of the Supervisor: _Aman Sharma_____

Title of the Thesis/Dissertation/Project Report/Paper (In Capital letters): __ STUDENT SUVIDHA – A REACT NATIVE APPLICATION __

UNDERTAKING

I undertake that I am aware of the plagiarism related norms/ regulations, if I found guilty of any plagiarism and copyright violations in the above thesis/report even after award of degree, the University reserves the rights to withdraw/revoke my degree/report. Kindly allow me to avail Plagiarism verification report for the document mentioned above.

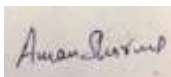
- Total No. of Pages = 58 (Excluding the Plag Report)
- Total No. of Preliminary pages = 8
- Total No. of pages accommodate bibliography/references = 1



(Signature of Student)

FOR DEPARTMENT USE

We have checked the thesis/report as per norms and found **Similarity Index** at 7(%). Therefore, we are forwarding the complete thesis/report for final plagiarism check. The plagiarism verification report may be handed over to the candidate.



(Signature of Guide/Supervisor)

Signature of HOD

FOR LRC USE

The above document was scanned for plagiarism check. The outcome of the same is reported below:

Copy Received on	Excluded	Similarity Index (%)	Abstract & Chapters Details	
Report Generated on	<ul style="list-style-type: none">• All Preliminary Pages• Bibliography/ Images/Quotes• 14 Words String		Word Counts	
			Character Counts	
		Submission ID	Page counts	
			File Size	

Checked by

Name & Signature

.....

Librarian

Please send your complete Thesis/Report in (PDF) & DOC (Word File) through your Supervisor/Guide at plagcheck.juit@gmail.com