MAD LAB PROJECT

SLcM APP

Submitted by:~

Name: Nishant Sahoo, Harshini N
Reg. No.: 150953244, 150953250

Lab Section: CCE A2

Instructor: Prof. Arjun CV
Course: Mobile Application Development Lab (ICT 3263)
Date: 12th April, 2018

ABSTRACT

Manipal Academy of Higher Education uses SLcM (Student Life Cycle Management) system which aims to digitize the Student-Academic section interaction. Few of the features include broadcast messages to students in real time for notices, day-to-day attendance, internal marks, grades, check financials and online applications for Open Electives and Program Electives. These are just few of the features of SLcM. The site is not mobile friendly and it gets difficult to use it on a mobile browser.

A mobile application for the same is necessary because:

1. The broadcast messages sent to students can be viewed on SLcM only after logging in and the student does not get to know when a new message has arrived.

2. Attendance is updated on a daily basis. If a student wants to skip a class but wants to make sure he/she does not have attendance shortage then the student will have to login into SLcM portal and click on a number of buttons to navigate to the attendance page. The mobile app will only require one swipe or click.

3. The UI and UX of SLcM is not very good. A mobile app can be designed with a lot of elaborate functions based on advanced gestures like ‘tap’, ‘swipe’, ‘drag’, ‘hold’ and more which will enhance UX. And a well-designed mobile app can perform actions much quicker than a website.

We plan to build the mobile application by using a web scraper. A web scraper is an Application programming Interface (API) to extract data from a website. The web scraper will extract data or copy specific data from SLcM and using JSON the data will be sent to the mobile application where the data will be displayed. JSON is used as a transport storage mechanism between the client and the web server.

Using the app, a student will be able to view attendance status and view personal details. All the other features of the website are not included in the mobile app because the above listed ones are the only frequently used ones. A mobile is used for faster and easier access to important and necessary features.

Technologies used: Flask, React Native

INTRODUCTION

The **Manipal Institute of Technology** is a constituent institution of Manipal Academy of Higher Education that specializes in the fields of engineering and technology. MIT contains 16 academic departments and awards undergraduate, graduate and post graduate degrees. Being a technical institute, MIT uses technology to simplify work. MIT with the aim of digitizing the Student-Administration interaction, uses SLcM whose features include broadcasting messages/notices to students in real time, day-to-day updation of attendance, internal marks upload, grades calculation, results check, financial transactions and online applications for Open Electives and Program Electives, Certificates required from college. These are just few of the features of SLcM.

The site is not mobile friendly and it gets difficult to use it on a mobile browser because it is responsive. A mobile application substitute for the web portal is required because firstly, broadcast messages sent to students can be viewed on SLcM only after logging in and the student does not get to know when a new message has arrived. Secondly, attendance is updated on a daily basis. If a student wants to skip a class but wants to make sure he/she does not have attendance shortage then the student will have to login into SLcM portal and click on a number of buttons to navigate to the attendance page. The mobile app will only require one swipe or click. Thirdly, the UI and UX of SLcM is not very good. A mobile app can be designed with a lot of elaborate functions based on advanced gestures like ‘tap’, ‘swipe’, ‘drag’, ‘hold’ and more which will enhance UX. And a well-designed mobile app can perform actions much quicker than a website.

We plan to build the mobile application by using a **Web Scraper** which will scrape data from the SLcM main website. A web scraper is an Application programming Interface (API) to extract data from a website. Web scraping is data scraping used for extracting data from websites. Web scraping software may access the World Wide Web directly using the Hypertext Transfer Protocol, or through a web browser. While web scraping can be done manually by a software user, the term typically refers to automated processes implemented using a bot or web crawler. It is a form of copying, in which specific data is gathered and copied from the web, typically into a central local database or spreadsheet, for later retrieval or analysis.

The web scraper will extract data or copy specific data from SLcM and using JSON the data will be sent to the mobile application where the data will be displayed. JSON is used as a transport storage mechanism between the client and the web server. **JSON** (JavaScript Object Notation) is a lightweight data-interchange format. It is easy for humans to read and write. It is easy for machines to parse and generate. JSON is a text format that is completely language independent but uses conventions that are familiar to programmers of the C-family of languages, including C, C++, C#, Java, JavaScript, Perl, Python, and many others. These properties make JSON an ideal data-interchange language. JSON is built on two structures:

- A collection of name/value pairs. In various languages, this is realized as an *object*, record, struct, dictionary, hash table, keyed list, or associative array.
- An ordered list of values. In most languages, this is realized as an *array*, vector, list, or sequence.

The mobile application is built using **React Native,** a hybrid mobile-app development framework for iOS and Android**.** The tool is based on the very same principles as ReactJS. **ReactJS** is a JavaScript library that combines the speed of JavaScript and uses a new way of rendering webpages, making them highly dynamic and responsive to user input. React Native combines native application development with JavaScript UI development. While computationally heavy features can be implemented with native modules for iOS and Android, the rest of the code can be written with JavaScript and shared across platforms. React Native doesn't use WebView, a mobile engine that wraps code. Instead, it provides access to native mobile controllers which allows for achieving native look and feel to apps. React Native uses native APIs as a bridge to render components on mobile. For example, for Android components, it uses Java APIs and it invokes Objective-C API to render to iOS. React Native framework is one of the fastest and most efficient environments for mobile app development.

METHODOLOGY

Web scraping a web page involves fetching it and extracting from it. Fetching is the downloading of a page (which a browser does when you view the page). Therefore, web crawling is a main component of web scraping, to fetch pages for later processing. Once fetched, then extraction can take place. The content of a page may be parsed, searched, reformatted, its data copied into a spreadsheet, and so on. Web scrapers typically take something out of a page, to make use of it for another purpose somewhere else. An example would be to find and copy names and phone numbers, or companies and their URLs, to a list (contact scraping).

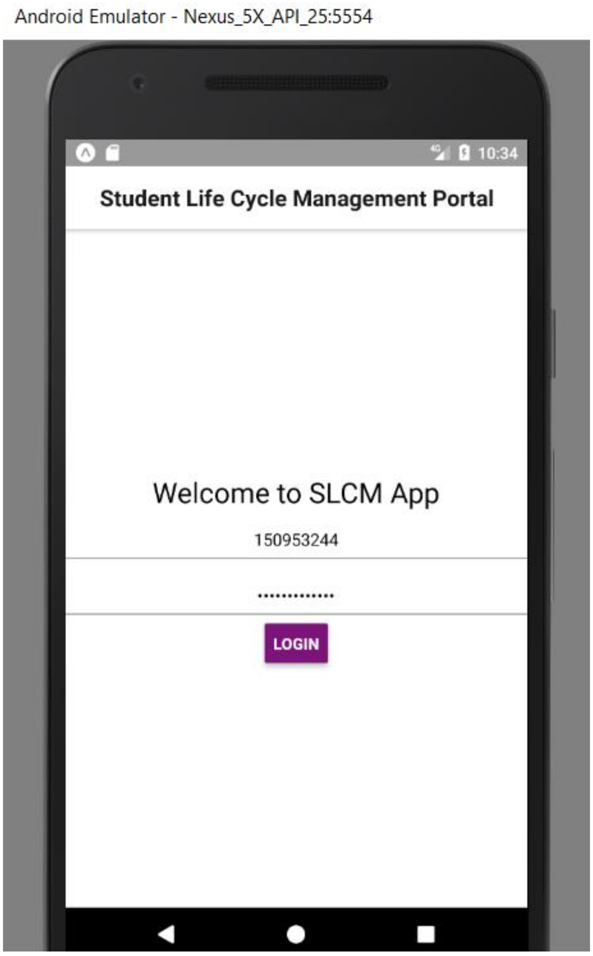
When the user first enters his/her login credentials, the data is stored in JSON format and using POST request method the login credentials are validated. To get the data from SLcM, get method is used. All the data scraped are converted to JSON format and sent to the mobile application to display. The mobile application is built using React Native.

RESULTS

The final app built has the following features:

1. Improvised UI
2. Faster Access
3. Can check attendance
4. Can check personal details
5. Does not locally store data anywhere (keeping data privacy in mind)

SNAPSHOTS



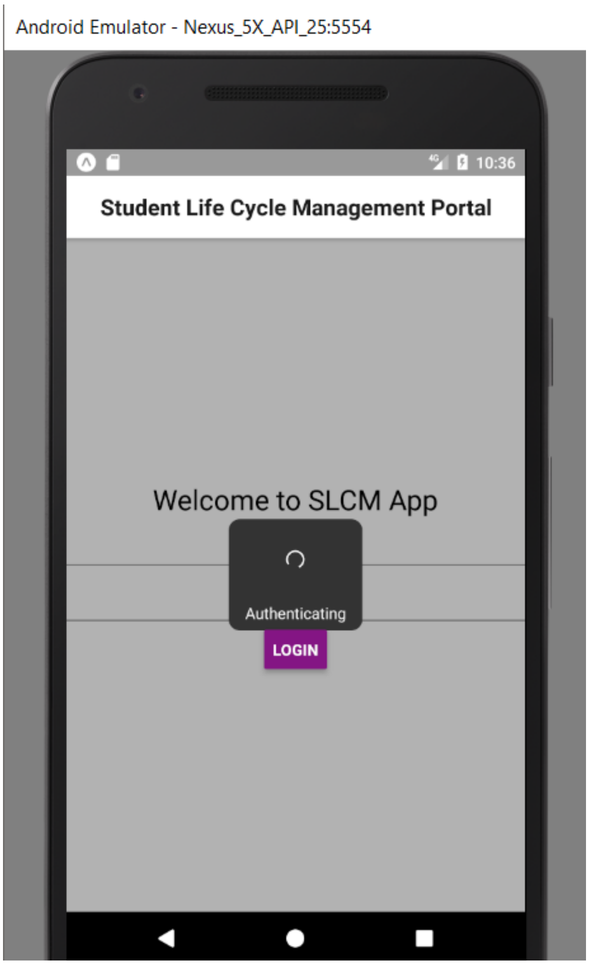
Android Emulator - Nexus_5X_API_25:5554

Student Life Cycle Management Portal

Welcome to SLCM App

150953244

....



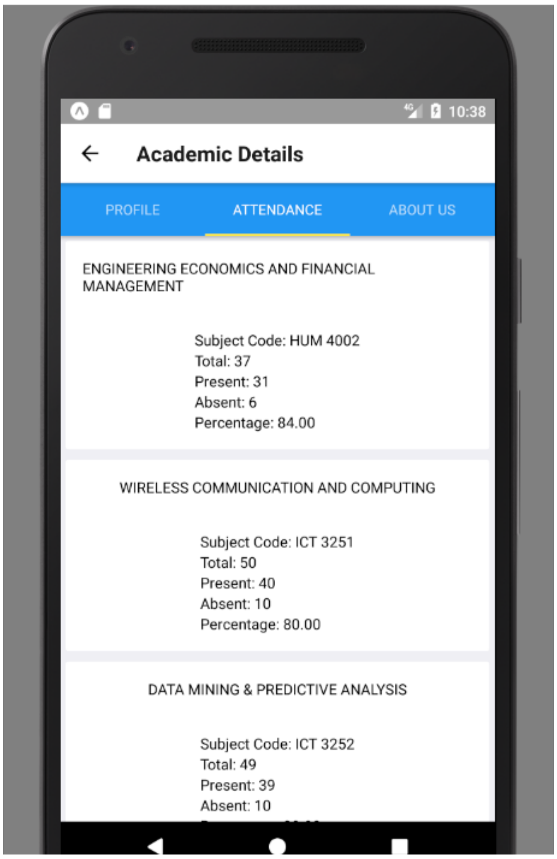
Android Emulator - Nexus_5X_API_25:5554

Student Life Cycle Management Portal

Welcome to SLCM App

Authenticating

Android Emulator - Nexus_5X_API_25:5554



Academic Details

ENGINEERING ECONOMICS AND FINANCIAL MANAGEMENT

Subject Code: HUM 4002
Total: 37
Present: 31
Absent: 6
Percentage: 84.00

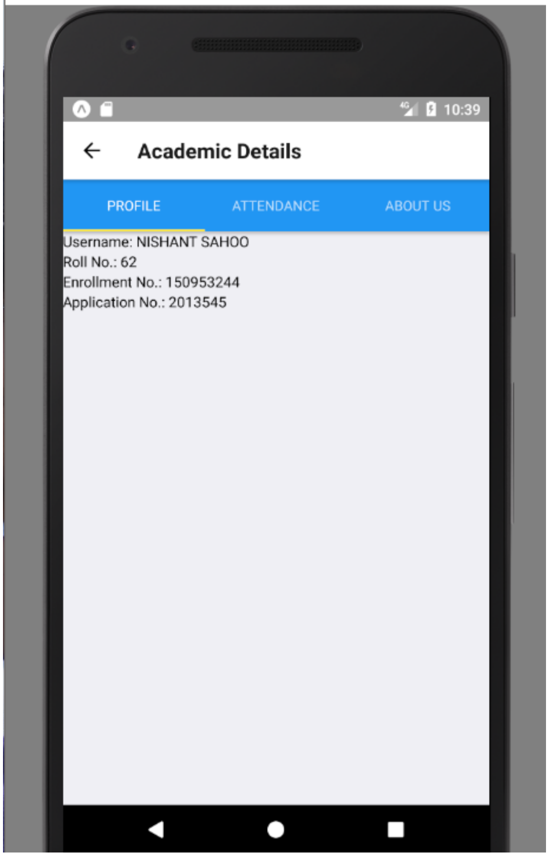
WIRELESS COMMUNICATION AND COMPUTING

Subject Code: ICT 3251
Total: 50
Present: 40
Absent: 10
Percentage: 80.00

DATA MINING & PREDICTIVE ANALYSIS

Subject Code: ICT 3252
Total: 49
Present: 39
Absent: 10

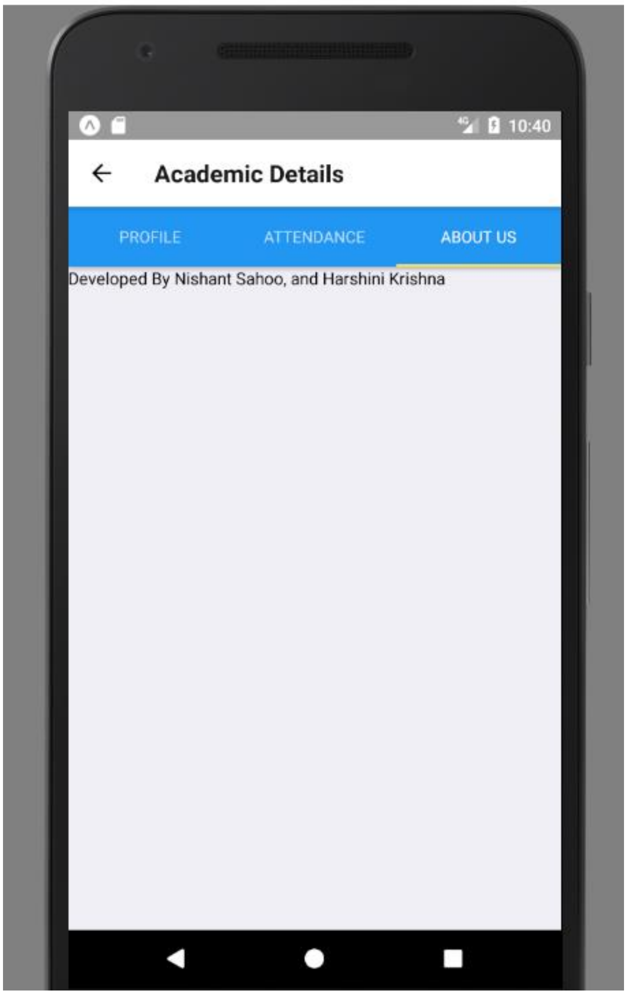
Android Emulator - Nexus_5X_API_25:5554



Academic Details

Username: NISHANT SAHOO
Roll No.: 62
Enrollment No.: 150953244
Application No.: 2013545

Android Emulator - Nexus_5X_API_25:5554



Academic Details

Developed By Nishant Sahoo, and Harshini Krishna

CONCLUSION

After building this app, we can conclude that:

1. Native controls and native modules in React Native improve performance of the app.
2. React Native uses JavaScript – fast and popular programming language. So app runs pretty fast.
3. React Native contains all ReactJS features, aimed at improving UI. UI is good.
4. Unfortunately, React Native inherits the main ReactJS disadvantage. The community is young so the available documentation is poor, especially for integration with additional tools.
5. Expertise for native modules is also unavailable.
6. Lagging SDK Updates is another disadvantage.

FUTURE WORK

We plan to work on including all features of the SLcM website in the mobile app and making it an official SLcM App for the college. We also plan on making this app work offline (parts of it) and integrate it with reminders for regular classes, labs, sessionals, end semester exams and extra classes.

REFERENCES

1. https://slcm.manipal.edu/loginForm.aspx

2. https://devcenter.heroku.com/

3. https://reactjs.org/

4. https://www.w3schools.com/js/js_json_intro.asp

5. https://www.analyticsvidhya.com/blog/2015/10/beginner-guide-web-scraping-beautiful-soup-python/

6. http://www.pythonforbeginners.com/python-on-the-web/web-scraping-with-beautifulsoup/