**Project Report**

**on**

**“Studify”**

**Submitted to:**

**Faculty of Computer Science & Applications**

**Amrapali Institute of Applied Sciences, Haldwani**

**In the partial fulfillment for the award of degree of**

**Bachelor of Computer Application**

**Project Guide: Dr. Suresh Wariyal**

**Submitted By: Chandra Mohan Pandey**

****

**Amrapali Institute of Applied Sciences, Haldwani**

**Project Report**

**on**

**“Studify”**

**Submitted to:**

**Faculty of Computer Science & Applications**

**Amrapali Institute of Applied Sciences, Haldwani**

**In the partial fulfillment for the award of degree of**

**Bachelor of Computer Application**

**Project Guide: Dr. Suresh Wariyal**

**Submitted By: Chandra Mohan Pandey**

****

**Amrapali Institute of Applied Sciences, Haldwani**

**Declaration**

“I confirm that the work within the completed assignment is all my own work, and does not include any work completed by anyone other than myself unless referenced. I have completed the assignment in accordance with instructions given by the assigned mentor and within the time limits set by my mentor.

By signing my name below, I am declaring the authenticity of the work done.”

Name: Chandra Mohan Pandey

Place: Haldwani

Date: 14-06-2020

**Acknowledgement**

I would like to express my special thanks of gratitude to my mentor (Dr. Suresh Wariyal) as well as our director (Dr. M. K. Pandey) who gave me the golden opportunity to do this wonderful project on the topic (Studify), which also helped me in doing a lot of research and I came to know about so many new things I am really thankful to them.

Secondly, I would like to thank our computer department (Faculty of Computer Science & Applications) of my college Amrapali Institute of Applied Sciences, Haldwani.

At last, I would also like to thank my parents and friends who helped me a lot in finalizing this project within the limited time frame.

**Index**

|  |  |  |
| --- | --- | --- |
| **S. No.** | **Topic** | **Page No.** |
| 1 | Abstract of the project | 8 |
| 2 | System Requirements | 9-10 |
| 3 | System Requirement Specifications | 11-17 |
| 4 | Design | 18-23 |
| 5 | Test Plan | 24 |
| 6 | Conclusion | 25 |

**Abstract of the project**

Studify is a student utility app.

It is designed by keeping-in-mind the needs of the students of today’s generation.

It provides its users with many useful functionalities such as the following:

1. Keeping a track of their attendance.
2. Listing their current semester/ class subjects.
3. One-tap accessibility to the time table.
4. Adding the list of upcoming tasks.
5. Making a profile about themselves and indicating their current overall attendance.

In one line, “everything that a student need”.

**System Requirements**

1. **Product Definition**

Studify is a student utility app which provides its users with functionalities that improve their academic performance.

1. **Problem Statement**

A lot of students often struggle to keep up with the consistency in today’s world which is full of distractions all around. They have little idea regarding their current attendance or the upcoming tasks and assignments or their subjects or their timetable. Here’s when **Studify** come to the rescue!

1. **Functions to be provided**

* Attendance management
* Subjects listing
* Time Table accessibility
* To Do List
* Profile maintenance

1. **Processing Environment:**

Hardware recommendation:

* Installed Memory (RAM): 4GB DDR3 1333MHz
* Internal Storage: 512GB HDD
* Processor: Intel Core i3-3220 CPU @ 3.30 GHz
* Processor Type: x64-based Processor

Software recommendation:

* Code Editor: Visual Studio Code 1.46.0
* Runtime Environment: Node.js 12.16.1
* Package Manager: NPM 6.13.4
* Framework: React Native 0.62.2
* Operating System: Windows 10 Home
* OS Type: 64-bit Operating System

1. **Solution strategy**

The most optimized and accessible solution for the given problem came out to be a mobile application which works without any need of an internet connection or high device specifications.

1. **User Story & Acceptance Criteria**

User Story: As a student, the user wants evaluation of his progress and reminder of upcoming tasks to improve his academic performance.

Acceptance Criteria:

* Display student’s current attendance for each subject.
* Display list of the concerned subjects of current semester/ class.
* Display time table in just one tap.
* Display upcoming tasks/ assignments.
* Display profile and overall attendance.

1. **Feasibility Analysis**

Feasibility Type:

* Economic Feasibility: No need for any major capital.
* Technical Feasibility: Can be coded on any latest and working system.
* Operational Feasibility: Can be executed and tested on a smartphone emulator or a mobile device.
* Schedule Feasibility: Can be completed within the given timeline.

1. **Project Plan**

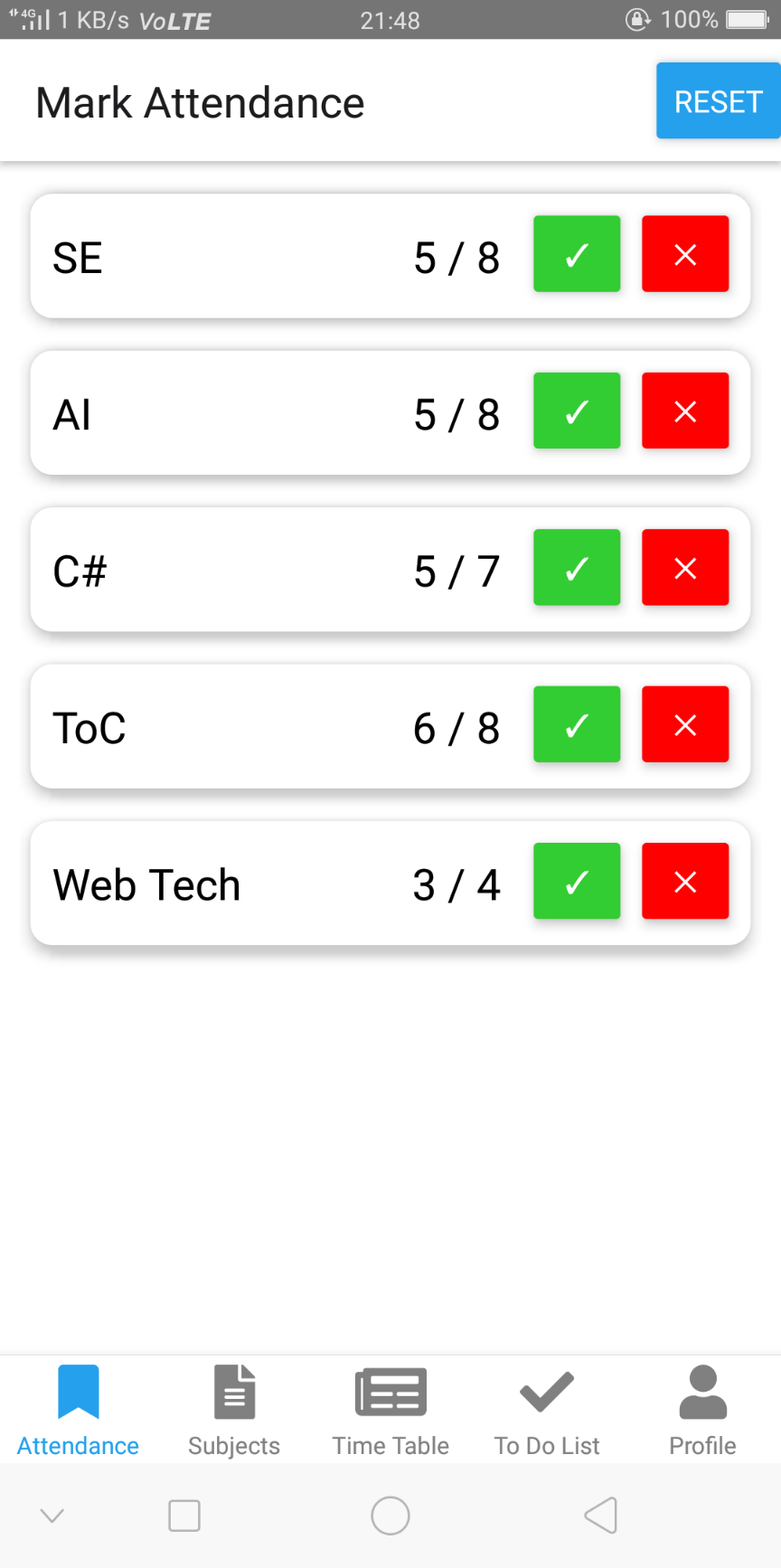
* Team structure: One developer.
* Development Schedule: Everyday 09:00 to 15:00.
* Programming Language: JavaScript
* Framework: React Native
* Runtime Environment: Node.js
* Development Tool: Visual Studio Code

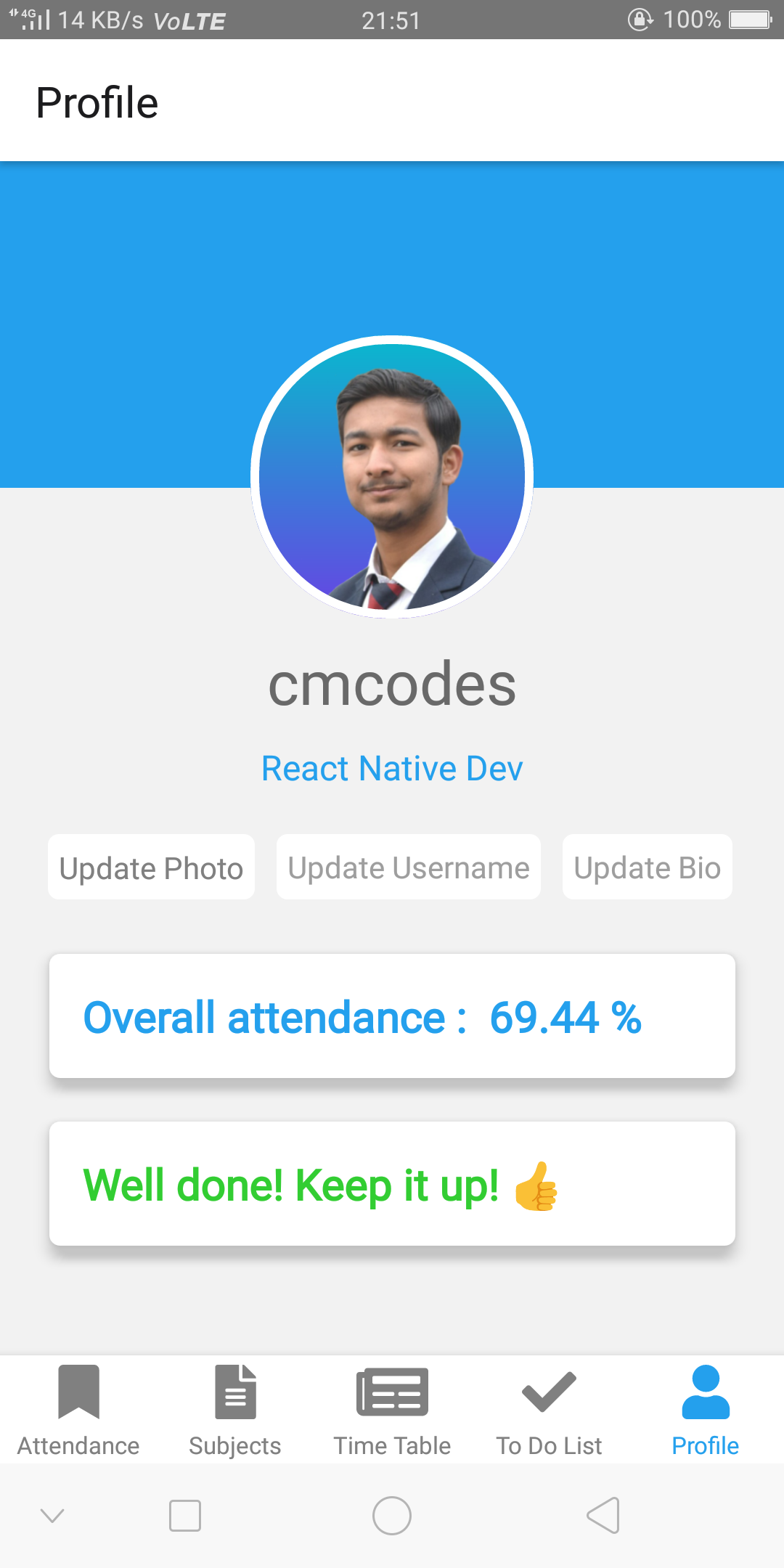
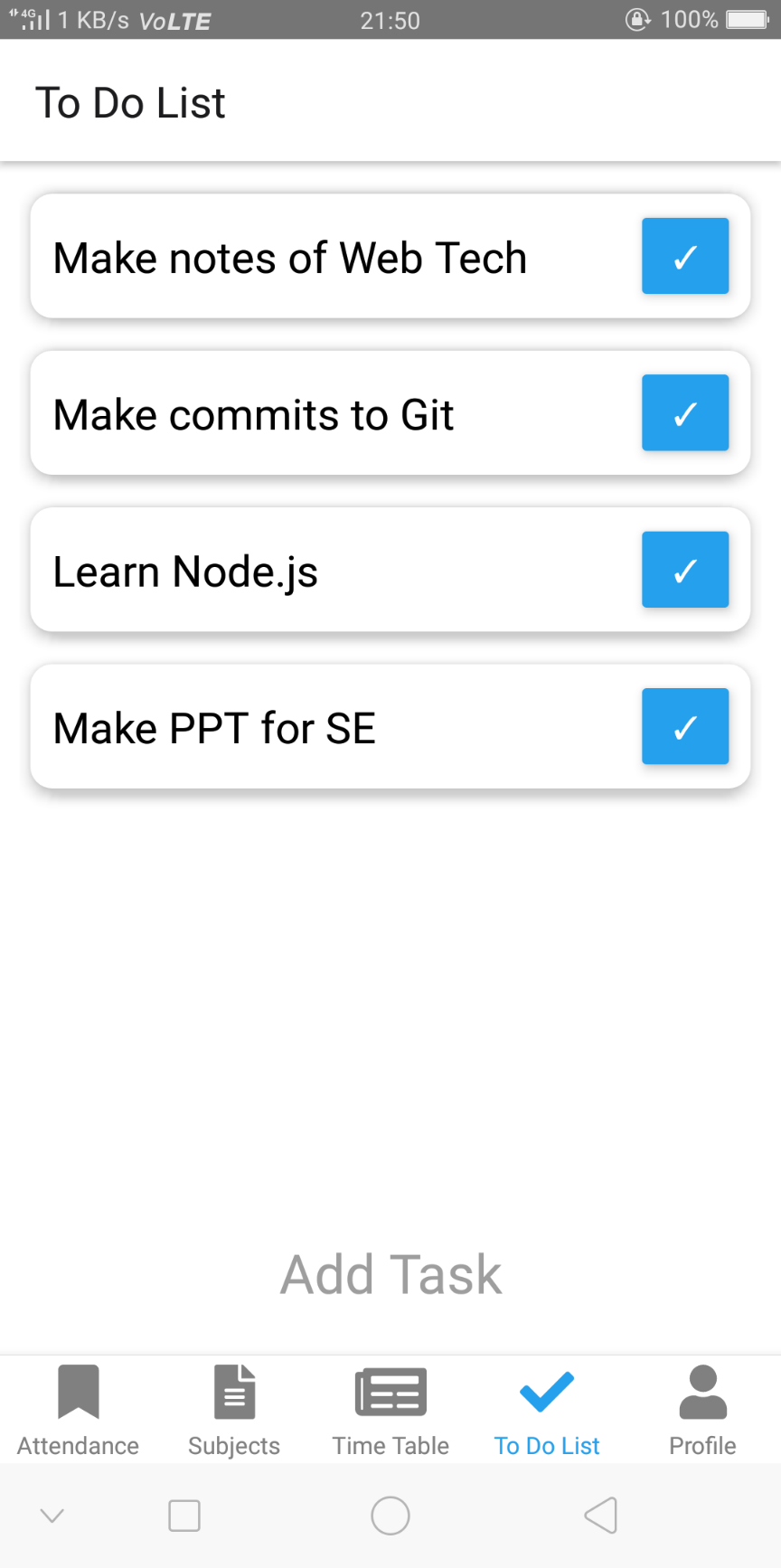
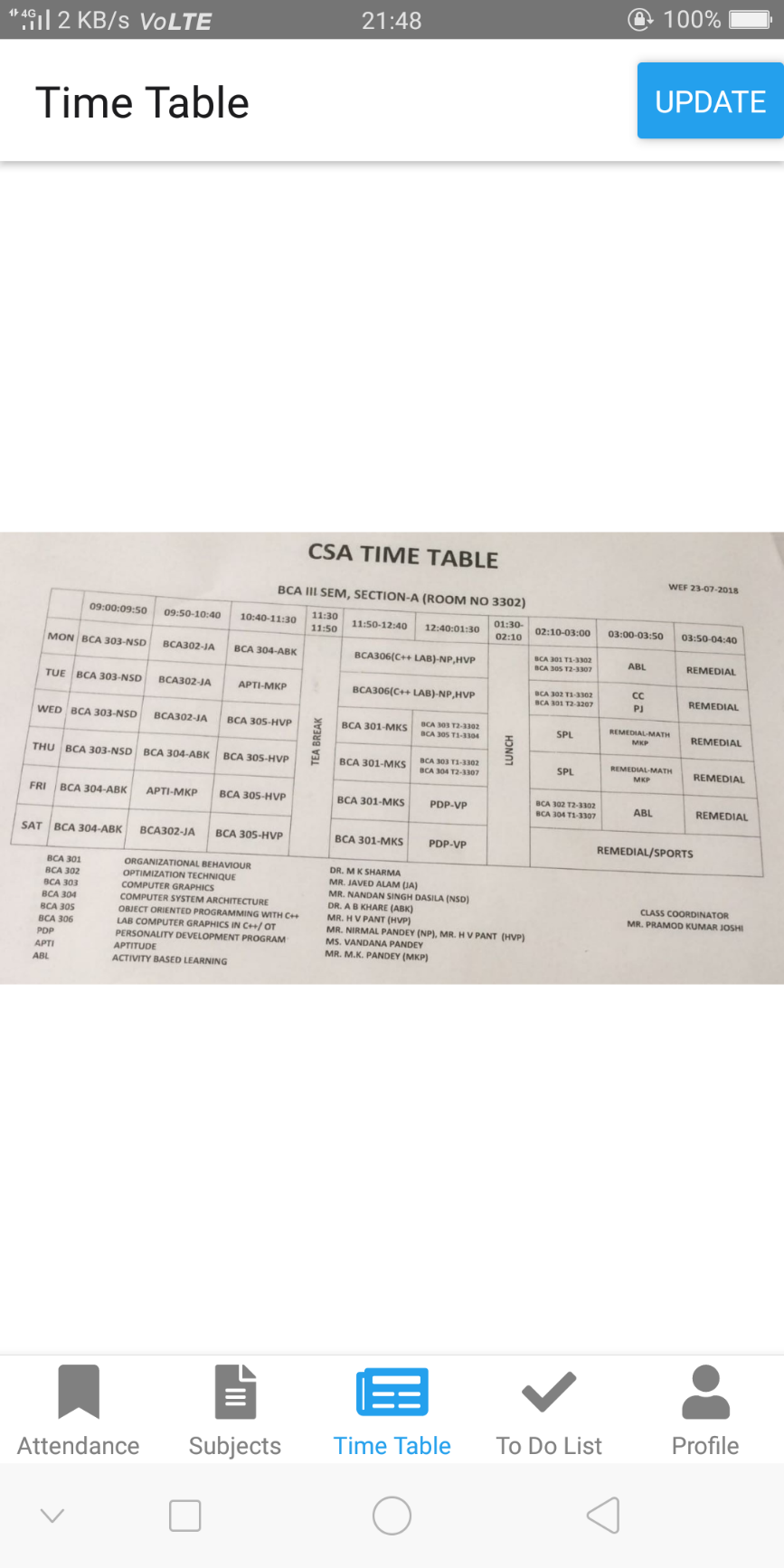
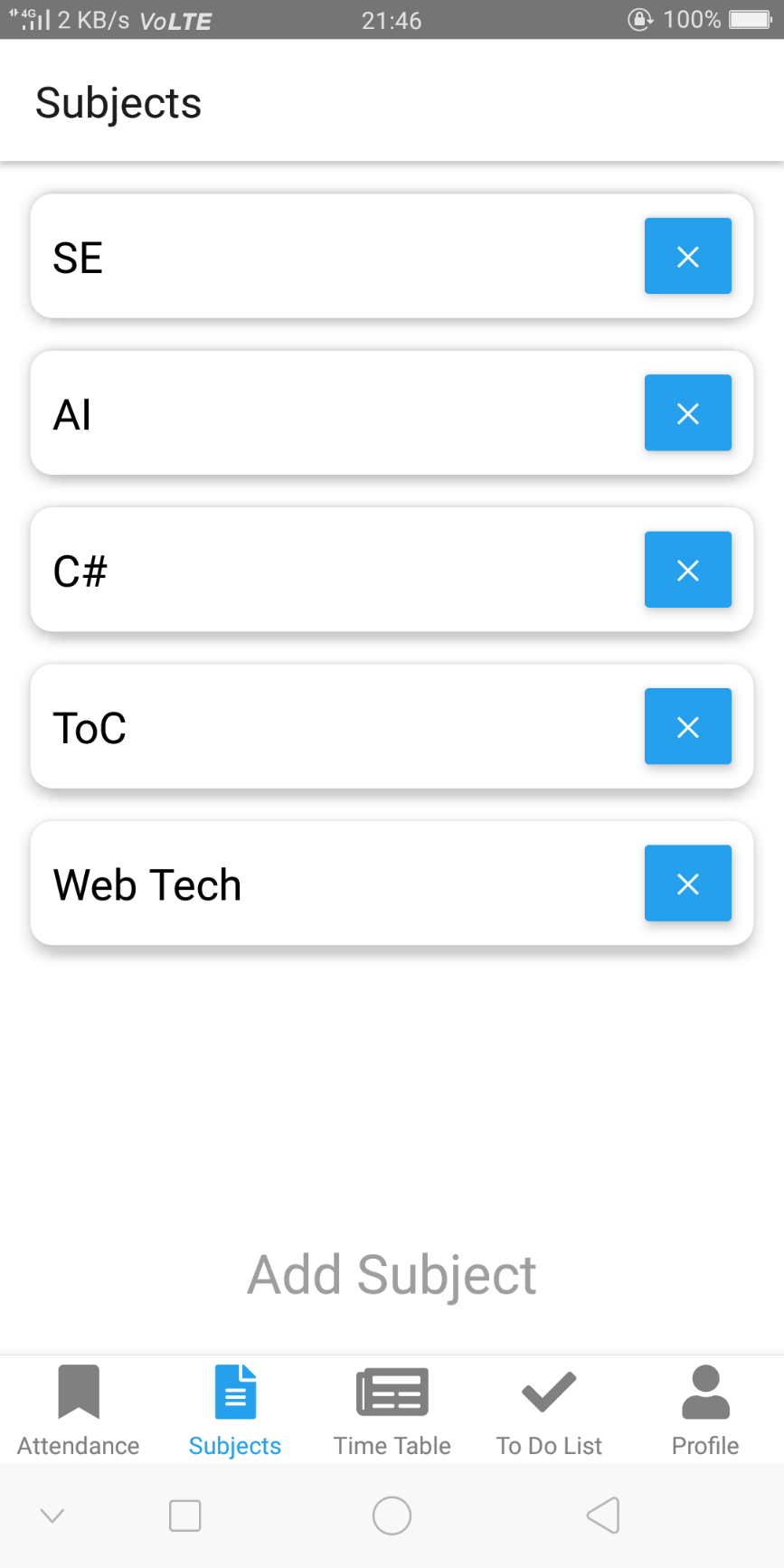
**System Requirement Specifications**

1. **Developing / Operating / Maintenance Environment**

* Installed Memory (RAM): 4GB DDR3 1333MHz
* Internal Storage: 512GB HDD
* Processor: Intel Core i3-3220 CPU @ 3.30 GHz
* Processor Type: x64-based Processor
* Code Editor: Visual Studio Code 1.46.0
* Runtime Environment: Node.js 12.16.1
* Package Manager: NPM 6.13.4
* Framework: React Native 0.62.2
* Operating System: Windows 10 Home
* OS Type: 64-bit Operating System

1. **User Display Format**



****

**Data Dictionary**

|  |  |  |  |
| --- | --- | --- | --- |
| **Field Name** | **Data Type** | **Description** | **Example** |
| present\_count | Array | Stores the count of individual classes attended of the respective subjects sequentially. | [2, 3, 5, 7, 5] |
| total\_count | Array | Stores the count of individual classes attended of the respective subjects sequentially. | [4, 7, 6, 7, 8] |
| present | Integer | Stores the total number of classes attended of all the subjects. | 22 |
| absent | Integer | Stores the total number of classes held of all the subjects. | 33 |
| subjects | Array | Stores the name of all the subjects. | [“SE”, “AI”, C#”, “TOC”, “Web Tech”] |
| text | String | Stores the current subject. | “Web Tech” |
| refreshing | Boolean | Stores the current refresh state. | false |
| filePath | String | Stores the path on the device of the saved time table photo. | “val” |
| tasks | Array | Stores the name of all the tasks. | [“Submit assignment”, “Complete the app dev”, “Read book “] |
| dpPath | Array | Stores the path on the device of the saved profile photo. | “path” |
| username | String | Stores the name of the user. | “cmcodes” |
| bio | String | Stores the bio of the user. | “Developer” |

**Design**

1. **Use Case Diagram**

Student

1. **Data Flow Diagrams**

0-level DFD:

Attendance

Student

Profile

Subjects

Time Table

To Do

1-level DFD:

Subjects List

Overall Attendance

Enter Subjects

Enter Profile Details

Select Time Table Photo

Enter Tasks

Student

2-level DFD:

Enter Subjects

Student

Save Subjects

subjects

present

Retrieve Subjects

subjects

total\_count

total

present\_count

Student

Mark Attendance

Select Time Table Photo

Student

filePath

Enter Tasks

Student

Save Tasks

tasks

Retrieve

total classes

total

present

Retrieve

total presence

bio

username

total

Student

present

Enter Profile Details

filePath

1. **Entity Relationship Diagram:**

Subjects List

Attendance

Subjects

attendance

Profile

To Do List

Time Table

**Test Plan**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Test Plan** | **Test Type** | **Screen** | **Test Name** | **Purpose of Test** | **Test Data/ Situation** | **Expected Result** | **Actual Result** | **Outcomes and Actions Required** |
| 1 | App | Attendance | Rendering of subjects FlatList with buttons. | Test that screen renders as expected. | 15-06-2020, 21:03  Android 7.1 | FlatList containing all the subjects with respective buttons to mark attendance. | As expected. | Screen rendered as expected. No action required. |
| 2 | App | Subjects | Rendering of subjects FlatList. | Test that screen renders as expected. | 15-06-2020, 21:03  Android 7.1 | FlatList containing all the subjects with one delete button. | As expected. | Screen rendered as expected. No action required. |
| 3 | App | Time Table | Rendering of time table photo. | Test that screen renders as expected. | 15-06-2020, 21:03  Android 7.1 | Photo retrieval from AsyncStorage. | As expected. | Screen rendered as expected. No action required. |
| 4 | App | To Do List | Rendering of tasks FlatList. | Test that screen renders as expected. | 15-06-2020, 21:04  Android 7.1 | FlatList containing all the tasks with one remove button. | As expected. | Screen rendered as expected. No action required. |
| 5 | App | Profile | Rendering of profile details and overall attendance. | Test that screen renders as expected. | 15-06-2020, 21:04  Android 7.1 | Profile photo with username, bio and overall attendance. | As expected. | Screen rendered as expected. No action required. |

**Conclusion**

**Current Status of Project:**

Report name: 15th June 2020, Studify v0.0.1

Project status: On Track.

Milestone 1: Rendering of subjects FlatList successful.

Milestone 2: Rendering of subjects FlatList with buttons successful.

Milestone 3: Rendering of time table photo successful.

Milestone 4: Rendering of tasks FlatList successful.

Milestone 5: Rendering of profile details and overall attendance successful.

Issues/challenges: All issues have been taken care of as of now. No challenges.

**Bibliography:**

ReactJS official docs: <https://reactjs.org/docs/getting-started.html>

React Native official docs: <https://reactjs.org/docs/getting-started.html>

React Navigation official docs: <https://reactnavigation.org/docs/getting-started/>