

PROJECT IMPLEMENTATION

of the mobile application

“STUPP”

For the class

CSCI5708 MOBILE COMPUTING

SUBMITTED BY

MC DEVELOPERS

LEKHA KEERTHI lk982761@dal.ca

SITHARA JAYACHANDRAN st962427@dal.ca

AGBOLA ISEOLUWATOBI is814889@dal.ca

Table of Contents

1. Abstract	3
2. Completion Report:.....	3
3. Functional Decomposition	4
4. High Level Organization.....	5
5. Click-Streams.....	6
6. Layout	8
7. Implementation.....	10
8. Future Work:	11
References	12

1. Abstract

After careful planning of the project, the team has accomplished on rounding off all the features that fall under minimum, maximum and bonus functionalities. The students can now check the weather, have access to the locations of different campuses, find out near nearby stores and cool places to hang out, complete Dalhousie specific tasks before attending the class, have a day to day to-do list, contact helpline staffs (e.g. counselling services, LGBTQ, Dal legal aid etc.) in case of emergencies. As promised, the app has been working up to all the expectations proposed under the project contract.

2. Completion Report:

The team has successfully pulled off all the milestones and completed minimum, maximum and bonus features on time. It was observed that most of the challenges were faced on the beginning phases of the project. Initially, the team faced a problem on accommodating the minimalistic heuristic as the number features seemed to be high. The team came with an idea of merging on or more related features to make the design more meaningful and appealing. Once the design challenges were solved, the team had a smooth flow on the rest of the milestones.

The team faced an issue on fetching data from the network in an efficient manner. To tackle this problem, the application used a special library named 'Volley'. This library includes multiple operations that fetched the data from the network in a quick and efficient way (developer android, n.d.).

Git repository- https://git.cs.dal.ca/agbola/mc_project.git

Milestone	Description	Percentage Completion	TimeLine
Project Planning	Technology Research, Low Fidelity Wireframes and Click, Sitemaps	100	Week 1
Application Theme Design	Color Scheme Selection, Design Pattern Selection	100	Week 1
Minimum Features UI Design	Design of Application Home Activity, Mobile Recommender Activity, Campus Maps and Help Line User Interface	100	Week 2

Minimum Features Functionality	Implementation of Home Activity Functionality, Mobile Recommender Activity Functionality, Campus Maps Functionality and Help Line Functionality	100	Week 3
Expected Features UI Design	Design of Events User Interface	100	Week 3
Expected Features Functionalities	Design of Events Functionalities	100	Week 4
Bonus Features UI	Design of Student Checklist, Weather information and to do list User Interface	100	Week 3
Bonus Features Functionality	Design of Student Checklist, Weather information and to do list Functionality	100	Week 5
Application Polish	Polish of the Application Features	100	Week 5

Figure 2: Project Status

3.Functional Decomposition



Figure 3: Functional decomposition

Campus: It helps the user navigate to different points on the campus.

Events: Helps the user attend or cancel events planned at Dalhousie. The student can also find the location of the events.

To-do List: It's like a virtual diary where the students can keep track of their pending tasks.

Check List: Check list gives a complete list of tasks to be accomplished on campus. It also guides the students to download Dalhousie apps, maps and transit.

Places: This feature helps the students find the list of stores, campus buildings and restaurants.

Helpline: This feature provides a list of services and emergency contacts that helps the student's call the helpline staffs.

Weather: This helps the students to keep track of the current weather of Halifax

4.High Level Organization

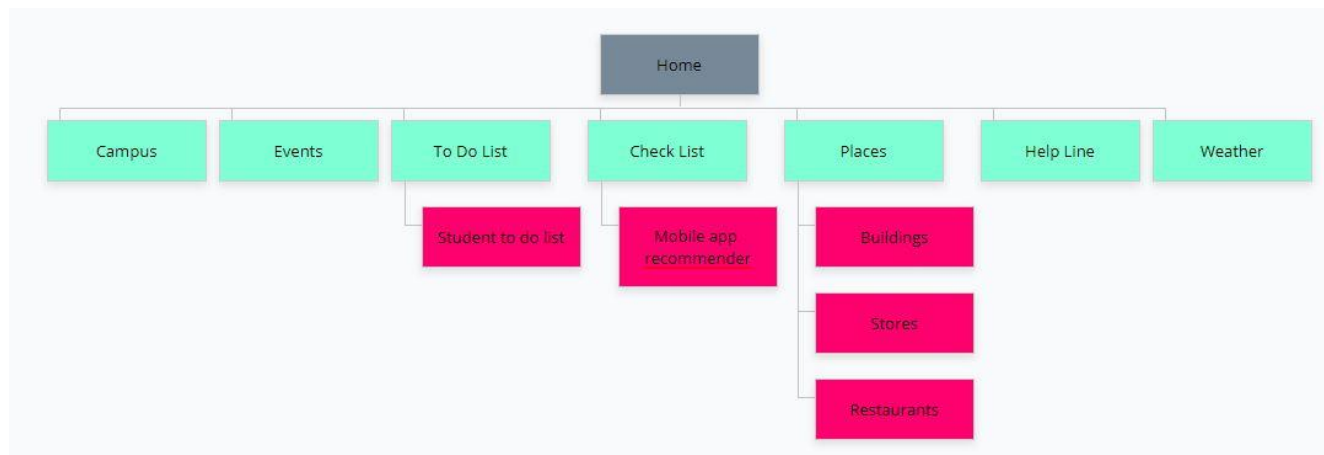


Figure 4: Site map

5.Click-Streams

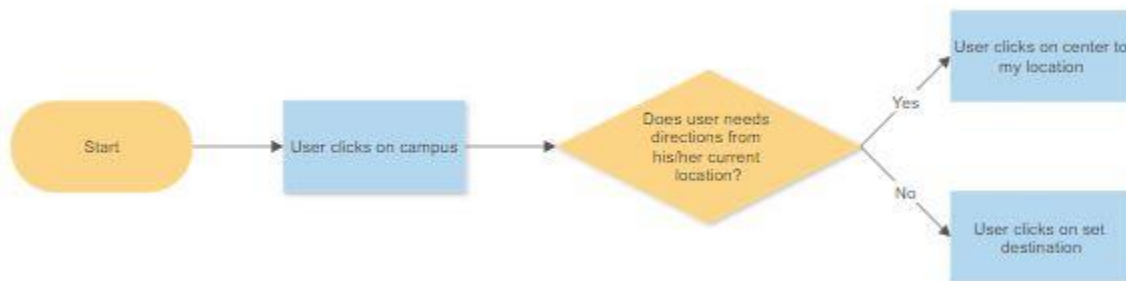


Figure 5.1: User flow for campus navigation

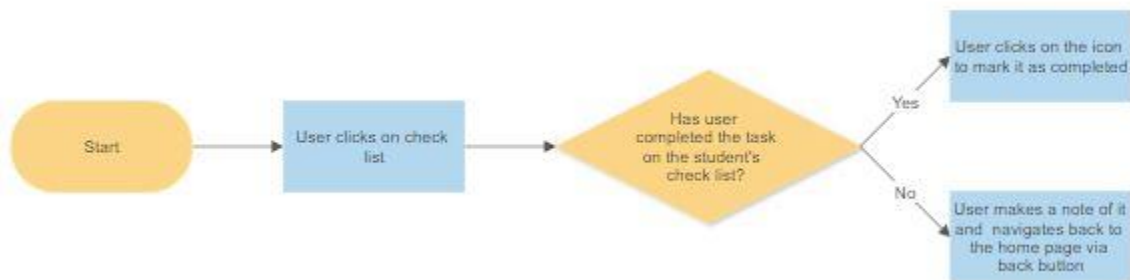


Figure 5.2: User flow for student's check list

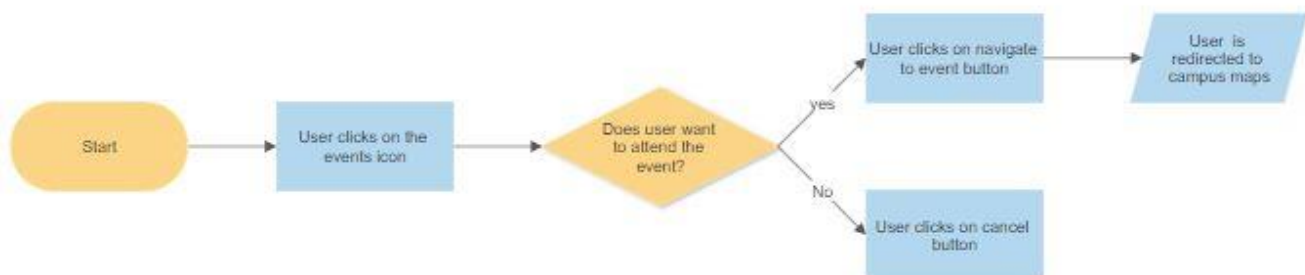


Figure 5.3: User flow for events icon

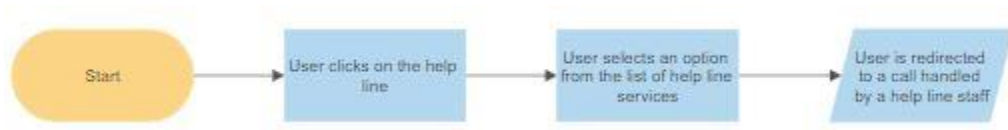


Figure 5.4: User flow for navigating through help line options

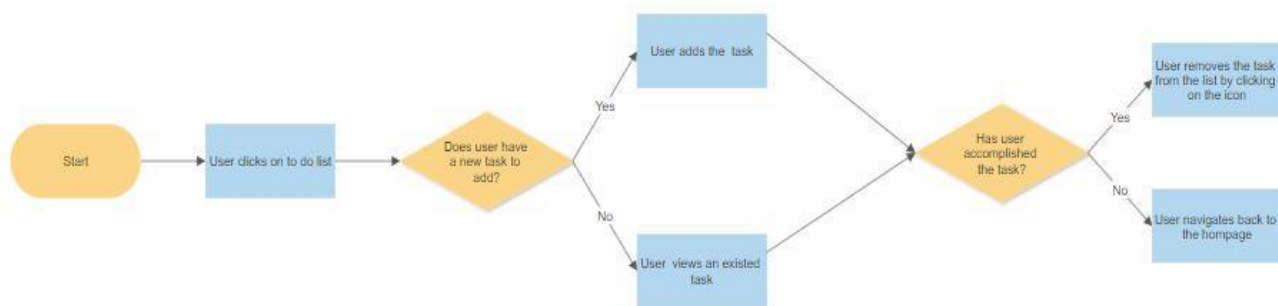


Figure 5.5: User flow for to do list

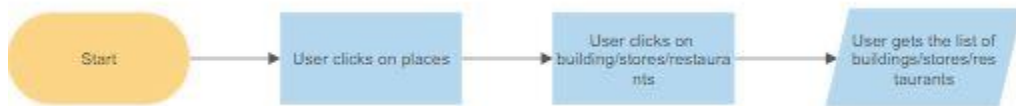


Figure 5.6: User flow for navigating through places

6.Layout

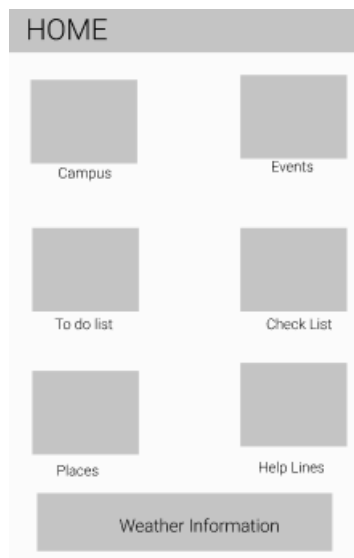


Figure 6.1 describes the home page of the application. The page includes 7 key features.

Figure 6.1

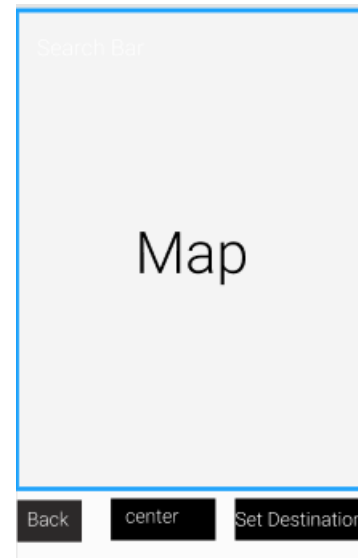


Figure 6.2 describes the 'Campus' feature.

Figure 6.2

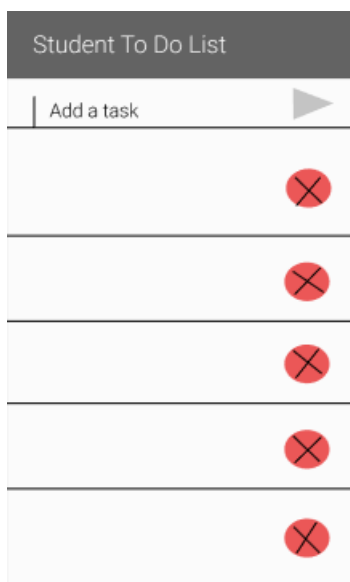


Figure 6.3 describes the 'To do list' feature. Allows users to add new tasks or delete completed tasks by clicking the red icons.

Figure 6.3

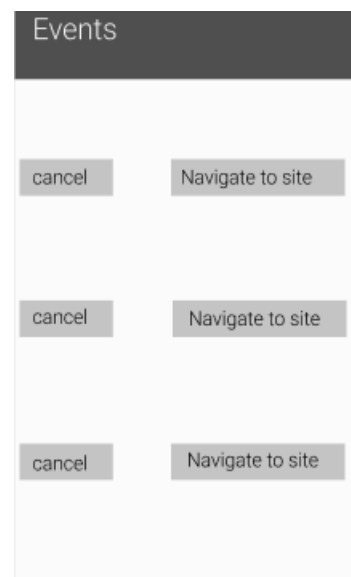


Figure 6.4 describes the 'Events' feature.

Figure 6.4



Figure 6.5

Figure 6.5 describes the 'Check List' feature. Red icons represent incomplete tasks while the green icons represent completed ones.

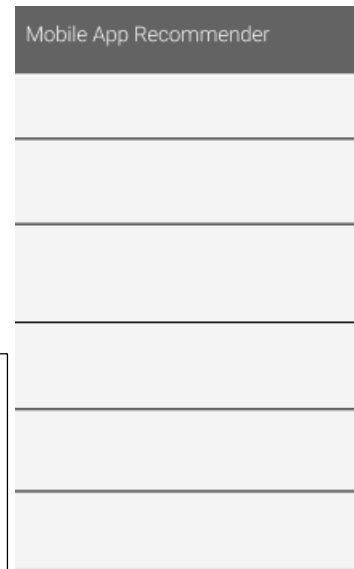


Figure 6.6

Figure 6.6 describes the 'Mobile app recommender' feature.



Figure 6.7

Figure 6.7 describes the 'Help Line' feature

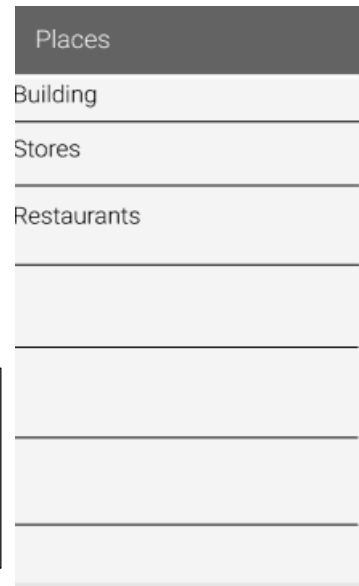


Figure 6.8

Figure 6.8 describes the 'Places feature'

7. Implementation

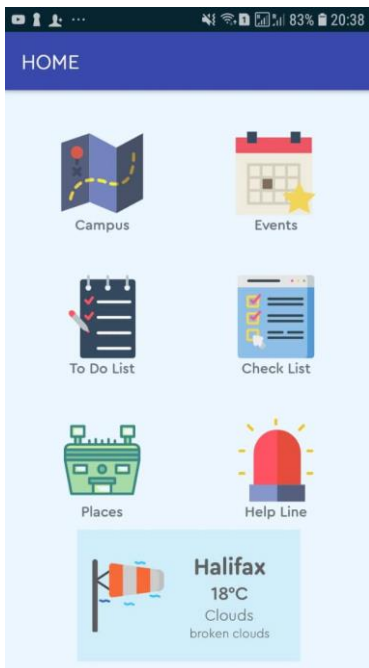


Figure 7.1 Home Page



Figure 7.2 Campus page

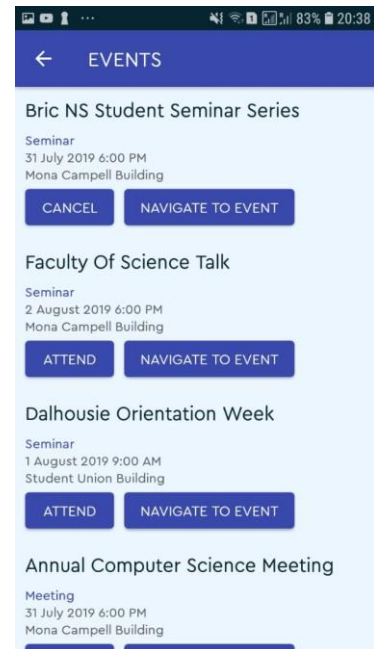


Figure 7.3 Events page

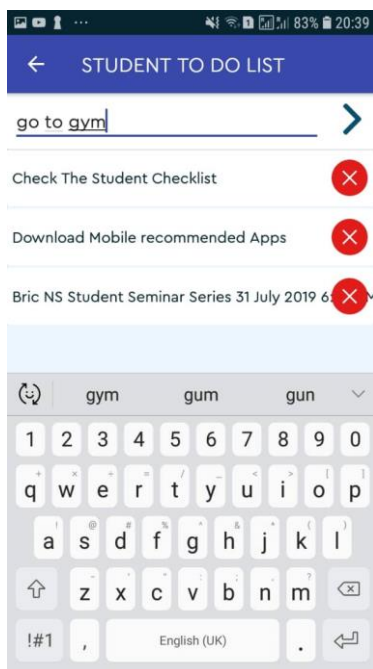


Figure 7.4 Student to do list page

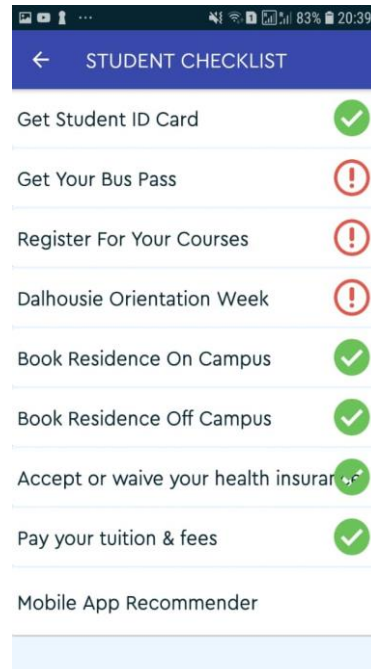


Figure 7.5 Student Checklist page

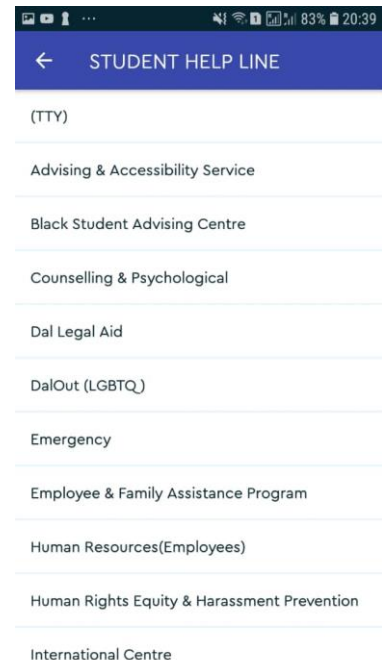


Figure 7.6 Help Line page

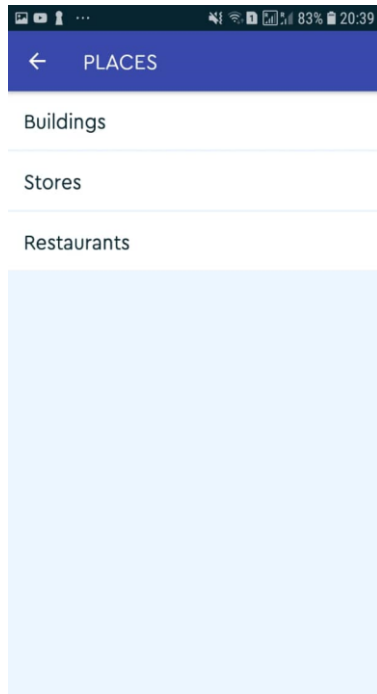


Figure 7.7 Places page

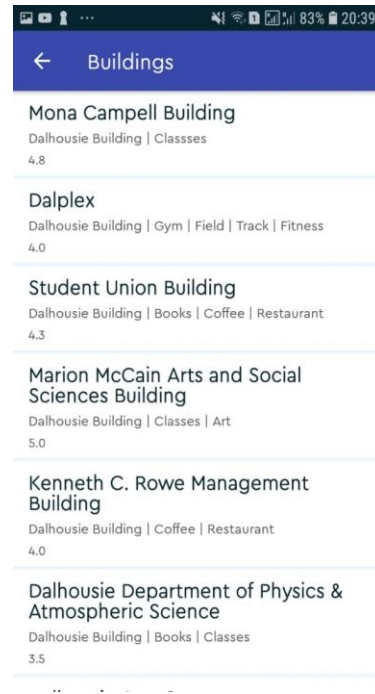


Figure 7.8 Buildings page

8. Future Work:

There is a huge scope on introducing some simple but useful functionalities on the existing features. For example, as the students are always trying to save money or spend less, the app could help the students find out the latest deals or discounts on the nearby stores or restaurants (under places). The student app could also be used on guiding students to their classrooms. Other functionalities like getting notifications on the weather or setting up reminders on the pending tasks could be also be implemented.

References

add maps. (n.d.). Retrieved from developer android: <https://developer.android.com/training/maps>

developer android. (n.d.). Retrieved from <https://developer.android.com/training/volley>

mathew, G. (2019, 07 20). *route directions*. Retrieved from wingssquare: <https://www.wingsquare.com/blog/drawing-driving-route-directions-between-two-locations-using-google-directions-in-google-map-android/>

weather api. (n.d.). Retrieved from weather: <https://openweathermap.org/>