Moods – App for Champlain Students

By

Amir Osman (CEO)

Étienne Bérubé (Secretary)

William Perron-Lafleur (Janitor)

Presented to

Mr. Amin Ranj Bar

420-204-RE, Integrative Project in Computer Science and Mathematics, section 00871

Date due: Wednesday, February 22nd, 2017

Champlain College Saint-Lambert

**Table of Contents**

**Introduction**…………………………………………………………………………………………………………………..1

**Design**…………………………………………………………………………………………………………………………….4

Analysis……………………………………………………………………………………………………………….4

Algorithm……………………………………………………………………………………………………………4

UML Diagrams…………………………………………………………………………………………………….5

Graphical User Interface……………………………………………………………………………………10

Object Design……………………………………………………………………………………………………11

Timeline……………………………………………………………………………………………………………12

Software……………………………………………………………………………………………………………13

**Features**……………………………………………………………..………………………………..……………………..14

**Conclusion**……………………………………………………………………………………………………………………15

**Introduction**

As our final project we want to create a social network on android phones for Champlain students. We want students to be able to meet their friends and new people in the College more easily by using our application. Each user is going to be able to reach out to other students within a click.

**Design**

Storyline

As students of Champlain College, we have noticed a need to enhance students' life at school by helping them connect quickly with their friends and classmates. It is often difficult for people to know where their friends are, what they are doing, who they are with, etc. Thus, an application that would be able to provide all relevant information about their presence at school would be extremely useful for most students, not only in Champlain, but also for all other schools.

Analyze, define, and understand the problem

The major problem is that students' are having trouble connecting quickly with their friends within their college. The main reason why is that they have no way of consulting their peers schedule, thus it is extremely inconvenient as students don't know when they have common breaks with their friends. This prohibits the students from doing things they would prefer doing together, such as studying, eating, learning, relaxing, and the list goes on.

Our application would be the perfect solution to that problem as one of its feature involves storing every students' schedule in database. Using an efficient mathematical algorithm, our tool will be able to perfectly match which students have common breaks. In other words, our users will be able to know who's available when you are to do anything you want, easily. In order to display what is that thing you want, you will be provided a tool to update your mood, thus your friends will know what you are up to.

Additionally, colleges are important in size, thus locating your best friends can most of the times be time-consuming. Therefore, our application would remedy to that problem by providing an option to share your location with your friends, so that they know exactly where you are when they need your presence, in the matter of seconds.

Again, there is an obvious need to simplify the students' social life at college by helping them connect together easily and quickly.

The Mathematic Algorithm

*If you use a specific algorithm or formula (physics, math or chemistry), write the algorithm or formula* for *solving the problem, along with a complete description and find the Efficiency of the algorithm. (Keep in mind the report has to be complete for anyone with any background).*

*Explaining your design and the reason behind it. The explanation should include at least: UML (complete description of every class, every method, and relationship between them). You should include explanations similar to* [*http://docs.oracle.com/javase/8/docs/api/*](http://docs.oracle.com/javase/8/docs/api/)

*GUI (design of every single page in your project. For drawing your GUI, you can use any software). Design of every character or object in your project*

*TimeLine, including all steps from first week of classes until the day of delivery.*

|  |  |  |  |
| --- | --- | --- | --- |
| Task | Plan Date | Assigned Person | Notes |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

* Specify the software will be used and explain the reason behind it, for example:*

*o IDE: NetBeans, Eclipse, etc.*

*o Platform: windows, Android, Unix, etc.*

*o Software or packages: JavaFX, 3D designer, SQL, Photoshop, etc.*

*Features*

* Explaining all the features your project will have.*

* Explaining all the limitations, constraints, etc.*

* Optional features: features that you will do your best to add.*

* Here, don’t only put titles. You have to explain each feature in details.*

Conclusion

**Design**

Analyze, define and understand the problem.

Algorithm

Explaining your design and the reason behind it

UML diagrams

GUI (design of every single page)

Design of every character or object in your project

TimeLine, including all steps from first week of classes until the day of delivery

|  |  |  |  |
| --- | --- | --- | --- |
| Task | Plan Date | Assigned Person | Notes |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Specify the software will be used and explain the reason behind it

IDE: Netbeans & Android Studio

Platform: Android Java

Software or packages: Android Studio XML

**Features**

\*means optional (will be implemented if we have enough time)

* Location of users (sends to selected contacts)
* Uses GPS location from device (API integrated in Android Studio )
* Schedule scanning (each account will have its schedule shared with contacts)\*
* Mood & Statuses (Looking for someone to Eat, Study, Tutor, Lift/Go Home, Chill)
* Helps you communicate with friends
* Stored in database (resets every 24h)
* Find friend by number (import contacts)\*
* Map of Champlain
* Drawn by us (JPG format)
* Ask school for plans\*
* May use GoogleMap SDK
* Notifications to cellphones
* Send if the friends is near or accepted a “poke request”
* Access to vibrations or other phone components
* Access to contacts\*
* Access to GPS
* Access to Vibrations
* Access to pictures \*
* Networking
* Use SQL databases (SQLite implemented in Android Studio)
* Store user information on server
* User name
* Email
* Phone number
* Schedule
* Store user information on phone
* ID
* Temporary mood
* Picture

**Conclusion**