**Team QQ Requirements Document**

**Cody Malone, Corey Massey, Kelsey St. Clair**

Executive Summary

Imagine bringing an object of fantasy and mysticism into this world and making it a reality. If the objects you hear about in fantasy or science fiction stories could be achievable through modern technology and properly integrated in a current setting, would that not be an accomplishment toward which to strive? Introducing the Neumont Marauder’s Map. Our goal with the NU-rauder’s Map is to capture the mysticism of everyone’s favorite fantasy series (Harry Potter) while making it practical in a modern environment at the same time. Never again will anyone who attends this university have a question as to the whereabouts of their instructors, faculty members, or other students. Everyone’s location coinciding with their schedule will be displayed on the NU-rauder’s Map for all to see.

While we do not have the legal right to track student or faculty members movements, our map will still function through information which is already public university knowledge. Based on everyone’s schedule for the quarter, their virtual figure will move around the building accordingly. As these icons move throughout their day weaving through the corridors of Neumont, the user will be able to interact with the icons to locate those individuals amongst the crowds. If the user decides to search for a particular individual, then the person in question’s icon will transform into a new unique icon which advertises their presence amongst the chaos of the school. Imagine there is a student using this program. He notices an icon representing a professor move throughout his day, and the user who is viewing this wants to know where he can locate said instructor when he is not preoccupied with classes. Through the process of searching for the sought after teacher or adjusting the time to a later moment, the user will determine his location and discover when the instructor has free time and then his need to obtain this information elsewhere will be satiated.

An interactive map keeping track of student and faculty schedules along with a plethora of useful, ease-of-access information about said individuals is a truly monumental feat that few would attempt. The potential uses for this program are numerous and would only serve to aid in productivity and management for the university. While the task set before our team may be monumental, our strength and resolve will see it to completion and implementation. Through our unwavering resolve and dedication, we will construct this piece of software which will change the way establishment are able to keep track of and manage the time of their constituents. The NU-rauder’s map will contain something of value to all parties who seek to use it and will represent the future that technology has the capability to build.

Scope and Purpose

When a product is created, there is always a reason behind its existence. There is always a group of people for which this product is specifically created and a group which benefits immensely from its actions. The markings of a high quality product are directly tied with its success in providing a useful resource to those in need. Being such, the NU-rauder’s map is no different in its quest to provide an entire body of scholars and instructors with a quick and easy means of locating fellow academics within their halls of education.

Being that this program is an interactive map of Neumont University, it would make the most sense that those who would utilize it often would be those who either attend or are employed at the university. It is highly unlikely that some entity outside of Neumont would find a use for, let alone care about the virtual map. However, inside the school, all members of the staff and student body would be able to find some sort of use for this software, even if it was only used to probe their curiosity. The map could also be used as a tool of encouragement and intrigue before curious visitors to the school. If any parent or benefactor were to have doubts as to the standards and quality of Neumont’s education, a simple glance at the map could sway their minds into a state of unyielding positivity. The sheer complexity of the map would most likely leave curious students or parents in awe of the skill that flows throughout the halls of the university. Even if the scope and influence of the program is rather self-contained, the effect that the map would have in just this space would undoubtedly be positive.

Since this map will most likely reach out to a high number of individuals within Neumont University, its reason for existence and its purpose for use must be well defined. Simply put, the NU-rauder’s map will provide a simple and efficient method for locating students and staff at any location within the building throughout the day. It can be used to keep track of the schedules of students and teachers and help correct errors in scheduling. The visual nature of the program will also be more appealing to most than the current way of keeping track of information. As humans, we tend to possess a higher appreciation for something that is visually appealing. Would the average person rather gaze at a list of names and schedules on a collection of scattered documents or behold a map constantly in motion and precisely displaying the location of those people listed on those documents? Aside from this, the map is also useful for demonstrating Neumont’s modern technological prowess to all those who look upon it. Nothing says 21st century like an interactive, ever-expanding digital map of your school. It is an attractive proposition to visitors and would definitely leave a positive lasting impression upon them. However, most of all it is an interesting demonstration of the types of marvels that can be constructed through programming. It stands as a testament to the fantastic applications that students will be able to craft for themselves if they choose to accept the responsibilities which come with it and purpose themselves to achieve greater.

Limitations and Exclusions

No matter how technologically advanced the NU-rauder’s map may be, it obviously has limitations on its ability to fulfill the needs of everyone who would utilize it. As a result of our team’s lack of experience and time limitations, there will be features missing from the map that would have made useful additions. While none of these missing features directly impact the final product or limit its usefulness, they would have made the program easier to manage.

First and foremost, the most obvious and glaring limitation is that the map will be unable to provide the definitive location of students and faculty members. If, under the supervision of the law, we were permitted to track others’ locations at all times within the building, then we would definitely have implemented this feature into our application. It would have been fascinating to see the icons on the map constantly moving and adjusting positions with fluid human motion as they traverse the school. The experience provided would have mirrored the effects of the Marauder’s Map almost precisely and would have provided a true Harry Potter like experience. Unfortunately, we are unable to track a person’s movements due to legal issues; thus, what we currently have implemented will have to suffice.

Due to the methods behind our software’s pathing system, our application will be unable to display the current number of students in each classroom. We originally intended for each icon represented on the map to disperse into the corners of the room which it entered. This method would have displayed the icons stationed side by side within a room during class time. However, due to technical limitations, our team was only able to display the icons stacked upon each other inside a room. The icons file into a room once class has started and join the same position where they wait until the class has ended. After which they will begin filing out of the room until it is emptied. This will still get the idea across and display the process as we originally intended, minus some extra flare.

Another area of limitation for our program relates to its data management. In its current form, users are required to manually submit data for each individual person they create on the map. While it gets the job done, it can easily become extremely tedious or burdensome when attempting to create hundreds of people for representation on the map. There is no way for the user to automatically sync data with an external list of names and class times. The implementation of such a feature would have saved the time and patience of many who will eventually use this program. However, even without this feature, our application will still be able to accomplish the goals which our team has set for it. We set out to create an ease-of-access tool for the user; however, it will not completely remove all the input of the user. Even with the existence of these limitations, our application is not held back but instead exceeds its negatives by establishing a fully functioning, useful application which provides a worthwhile experience to the user.

Requirements

|  |  |  |
| --- | --- | --- |
| ID | Name | Description |
| **#1** | Creating a Person | When the user first creates a person, the user will navigate the menu to find the appropriate JMenuItem for creating a new person. Once clicked, a new JOptionPane will open in which there will be text input fields. The user will then be prompted to enter information about the new person, specifically their name, status at school (teacher, student…etc.), how long they have been at the university, and their class schedules. The classes to choose will be displayed in JCheckBoxes of which the user will select the appropriate ones for the new person. The new person object is then sent to the controller where it adds a new person to the person library and saves the updated list. |
| **#2** | Deleting a Person | In order to delete a person from the list, the user will open the menu and locate the appropriate JMenuItem for deleting an already existing person. Upon choosing this option, a JOptionPane will open which displays a current list of active person objects and prompts the user to enter a name for removal. The chosen person object is then passed to the controller which removes the object from the person library and resaves the list for updated use. |
| **#3** | Change the Current Floor | If the user wished to view different floors on the map, then he would look toward a feature located in the main user interface. Next to the active map, there will be two JButtons which will allow the user to either view a higher or lower floor until the limit is reached on both ends. Once the appropriate button is clicked, the current floor is removed from the user map and the new floor is redrawn by the UI. The exception to this process would be if the user’s map was already viewing the top or bottom floor. If the user attempted to view a higher floor than the top floor, then the program would be unable to complete said action and the floor would remain the same. |
| **#4** | Change Time | If the user wished to adjust the current time of the map, there would be a clock located on the left side of the map. The clock is contained within a JTextField that is open to editing from the user. Once the time is adjusted, it is passed into a method that will update the person object positions on the map. The icons will then warp to their set positions for that time. The area to which they are warped if at all will be determined by the amount of time skipped. If time is only incremented by a minute or two, then the icon’s positions will most likely not adjust at all. Their positions will be based entirely around their set schedules. |
| **#5** | Search for a Person | If the user desired to search for a specific person on the map, he would utilize the onscreen JButton for the act. On the UI, there will be a labeled search button which upon being clicked will open a JOptionPane for user access. The user will be prompted to enter a person’s name. If the system can determine that said person exists, then the person will be located. The current floor of the map will be adjusted to that person’s current floor and the color of his icon will change to signal to the user that the unique icon is the person for which he was searching. Alongside this, an arrow might display above the icon to better signal to the user the location of their sought after person. |

References and Appendices

1. http://stackoverflow.com/
2. http://docs.oracle.com/javase/7/docs/api/