**Walking Navigation App**

**Project Management Plan**

**Version 2.0.0**

**Christopher, Rishi, Ryan, Yang**

**TEAM 55**

**Funny Fliers**

**10th May 2018**

# 

Table of Contents

[Introduction and Purpose 2](#_Toc514010470)

[Project Information 3](#_Toc514010471)

[Background and intended use 3](#_Toc514010472)

[Scope 3](#_Toc514010473)

[Deliverables 3](#_Toc514010474)

[Timeline 4](#_Toc514010475)

[Personnel/HR Management 5](#_Toc514010476)

[Project Leaders 5](#_Toc514010477)

[Team Members 5](#_Toc514010478)

[Member Responsibilities 5](#_Toc514010479)

[Decision on Processes 6](#_Toc514010480)

[Communications Management 7](#_Toc514010481)

[Risk Management 10](#_Toc514010482)

# Introduction and Purpose

The Navigation app as the name suggests is basically an app to navigate from one place to other with the help of smart devices. The developed app is useful in directing the user along paths comprising of ordered lists of geographical locations. The app will monitor the user’s location and give them directions to the next location on the path, noting the distance to the next waypoint and directing them to the subsequent waypoint until they reach the destination.

The Contents of this document gives a proper introduction and a proper Background information on how things were managed and all the necessary background information for a new member to read through and get all the information necessary. This Document also addresses and covers all aspects of the project for external use by the GetOverHere Organisation and all the staff members.

# Project Information

Background and intended use

The organisation behind outsourcing the navigation app is GetOverHere specializing in directing vision impaired people around cities. The company previously employed aides to walk with their customers in order to show them the direction but one of the company’s executives, Ian Scorp got in touch with Toposcope, an app in Mogul Pay™ store and showed keen interest in developing similar app to replace the current employed aides with the app which will be beneficial for the company. Mr. Scorp witnessing the main developers of the code as Team 55, contacted us and asked to provide him with the prototype of the navigation app and further the app will be modified for visually impaired customers.

Our overall responsibility was to create a prototype version of the app and hand in all the necessary information in terms of documentation and Project Management Plan.

Scope

The App in its complete edition should be able to direct blind and visually impaired people around the city but the task given to the developers of team 55 was to make a prototype of the app for directing the user around the two areas of Clayton Campus and Sunway Campus. The application should monitor the position of the user and direct him to the listed locations while showing the path at every step. The direction of the movement should also be shown to the user in the context of the location.

The Limitations of this app include the number of Locations. The Number of Location for this app to work are fixed and as of now, the user can only get directions and travel between these locations which are in Clayton and Sunway campuses. The Other Limitation of this app is that this app is just a Prototype and not the final App so it doesn’t contain the functionality of directing the visually impaired as of now but that responsibility will be assigned to the group of developers who are selected to further develop the app.

Deliverables

|  |  |  |
| --- | --- | --- |
| Task | Due date | Client |
| The Walking Navigation App | May 13th 2018 | GetOverHere Organization |
| User Guide | May 13th 2018 | GetOverHere Organization |

Timeline

|  |  |  |
| --- | --- | --- |
| Task | Description | Start date |
| Getting prepared | Setting up GitKraken, Bitbucket, and Asana | April 9th 2018 |
| Planning | Reading the project outline and lay out all the task | April 10th 2018 |
| Start working | Start making progress on the project | April 12th 2018 |
| Release of the app | App is finished and fully functioning | May 12th 2018 |
| User guide | Write a manual for the app | May 13th 2018 |

# Personnel/HR Management

Project Leaders

|  |  |
| --- | --- |
| **Project Leader** | Christopher |
| **Vice Project Leader** | Rishi Joshi |

Team Members

|  |  |
| --- | --- |
| Member | Contact Info |
| **Christopher** | [chri0005@student.monash.edu](mailto:chri0005@student.monash.edu) |
| **Rishi Joshi** | [rjos0006@student.monash.edu](mailto:rjos0006@student.monash.edu) |
| **Ryan** | [yliu0045@student.monash.edu](mailto:yliu0045@student.monash.edu) |
| **Yang Fei** | [yfei0002@student.monash.edu](mailto:yfei0002@student.monash.edu) |

Member Responsibilities

|  |  |  |
| --- | --- | --- |
| Member | Duty | Description |
| **Christopher** | Code Quality | Ensures the final product is up to the client’s standards and requests |
| **Rishi Joshi** | Documentation | Notes and files a report of the current progress |
| **Ryan** | Monitoring | Makes sure the current progress is not behind the timeline |
| **Yang Fei** | Risk Management | Identifies any error that might cause the app to break |

# Decision on Processes

As this app was to designed by a group of developers, communication was important. The main software for writing out the JavaScript was the Brackets application. The code was saved in the local copy of the author and to have better communication of the code and for the access of every developer, the software called GitKraken was used. GitKraken uses an online repository using BitBucket and gives access to all the team members of that repository. The main reason for the preference of GitKraken was the functionality of Version control. GitKraken has a built in function to track all the previous versions of the code that is being uploaded which can be highly useful if there is some error in the new version that is uploaded and it is easy to revert back to the previous version.

The main application used for assigning the tasks and having conversations regarding the assignment of code was Asana. Asana has this amazing feature to assign certain tasks to certain people which can be highly useful in collaborative tasks. Asana also has the in-built function of creating the number of tasks and marking them as complete if they are completed and hence Asana shows up the current standing of the project with the tasks completed and remaining.

Google Drive and Google Docs were the main facilities used for writing out the entire documentation of the project because everyone in the team had the access to the google docs and hence editing can become so much easier.

Other applications for casual communication includes Whatsapp, Text messaging and Phone Calls as in case of emergency or some important decision, if the person is unable to connect to the web service then he can communicate through Phone Calls and text messaging.

Communications Management

The main communication among the team members was done by Asana and all the code was communicated using the Bitbucket repository using GitKraken as the interface. The documentation was saved in Google Docs for free editing between all the group members.

The response time which was expected from the team members was 3-4 hours and if the response time is exceeded then in case of emergency, Christopher is the one to decide or if he is not approachable then, the most appropriate decision according to the person is chosen as the best one.

Now in case of some mistake or misunderstanding, with the use of version control, Christopher chooses the previous best version to continue working on it.

For better and efficient communication, email notification is set to active to receive messages on time and to reduce response time.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Communication name (e.g. Report, minutes, Asana entries)** | **Communication type:**  **external/internal and document/message** | **Audience of the communication** | **Purposes of the communication** | **Requirements specific to the type of communication** | **Strategies to make the communication effective** | **Frequency** |
| Project Management Plan | Internal document | The new members of the team. | Communicate procedures and information to team members | “Closed” and “context independent” | Needs to fully explain how to use the application, including addressing any known issues. | Once |
| Client presentation | External message | The client | To make the client aware about the functionality of application | “Open" and “context-dependent” | Every team member must present his part with a certain time limit. | Once |
| Group Chats through Whatsapp | Internal Message | New or existing team members | Real time discussion with team members | “Closed” and “context independent” | every team member must actively participate in group chat | Ad Hoc |
| Asana discussion | Internal Message | New or existing team members | Assigning tasks and other conversation | “Closed” and “context independent” | Every team member must finish the work appointed to them on time | Ad Hoc |
| Team meeting | Internal Documentation | New or existing team members | Record meeting outcomes, and planning | “Closed” and “context independent” | Every team member must come to the meeting on time | Once a week |
| GitKraken | Internal Documentation | New or existing team members | Record every modification to the code | “Closed” and “context independent” | Every team member needs to work together on the code and upload the code to GitKraken | Ad Hoc |
| User guide | External documentation | End users | A guide on how to use the application | “Open” and “context-dependent” | Needs to fully explain how to use the application, the requirements, and how to fix problems | Once |
| Google Docs | Internal Documentation | New or existing team members | Writing and saving the PMP and User Guide with other various Documentation | “Closed” and “context independent” | Those who are assigned the tasks should write the User Guide and PMP | Ad Hoc |
| Phone calls | Internal message | New or existing team members | Urgent matters | “Closed” and “context independent” | Every member should make phone calls if needed | Ad Hoc |

Risk Management

The main Risk factor is the code being replaced by erroneous code. This is one of the most common factor that can affect the teams progress and can lead to massive losses so, our team used GitKraken for saving all the previous versions of the code and reverting back to the original one when needed.

Another risk factor eliminated by GitKraken is the deletion of the Local Copy of code which can lead to huge loss. So, the code is primarily saved in BitBucket’s repository which is the probably the safest option and also is easy to access by any member. In this way, if the code is deleted, we have a safe online copy of the entire code and anyone can access it anywhere.

Other one of the major risk was the Member Absenteeism which can lead to huge losses especially when there is some kind information which only that member knows. This was a major aspect we faced and were mostly able to sort it out with the help of other texting applications which helped retrieving the useful information from the member but in case the member does not responds, this can be a big deal to overcome.

One other aspect that a team can face is the Team member’s conflict. We were Lucky as nothing of that sort was the case with our team but it is the area of concern that needs to be addressed by any team and any team should pre-decide upon it as what is to be done in that case. Ideally the Team Leader should be capable of solving the conflict but if that's not the case, then the matter should be raised to higher authority for solving up the dispute.