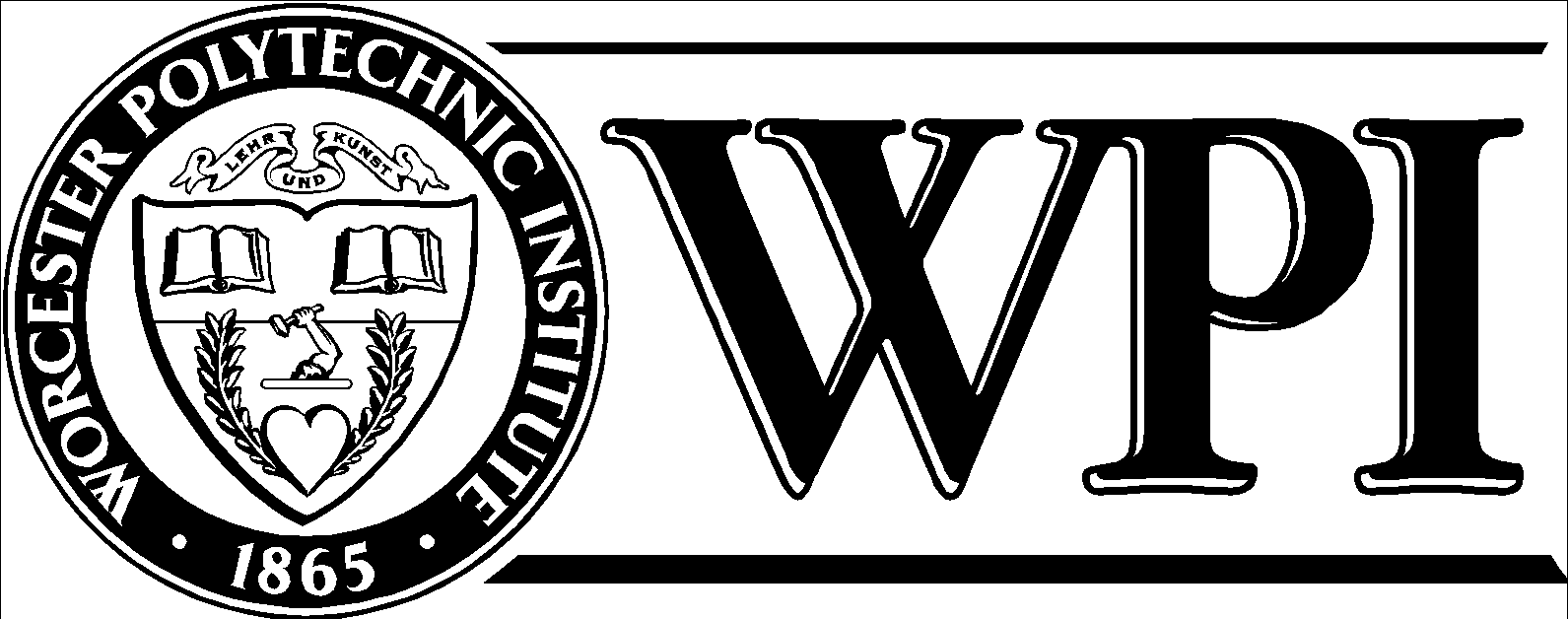
***Worcester Polytechnic Institute***

*CS3733 - Software Engineering*

**Get There**

**Deliverable - 2**

**Submitted By:**  
***Team 8***

**November 11, 2015**

**Professor Wilson Wong**

Introduction

This deliverable documents team 8’s progress through the first iteration of the navigation application Get There, which is exclusively written for the Worcester Polytechnic Institute Campus. This application is directed toward the WPI community; either students or faculty including the visitors. The first iteration will be the basic implementation of the application which will simply draw the desired path on a specified map. In future iterations this will be extended upon to allow for more advanced features. In order to allow for an efficient implementation of the application, the team is using Agile Development approach. The details and plans for the entire application including the first iteration will be elaborated on below.

Current System

Current students are able to find campus maps of the WPI campus in paper handouts and in online 2D and 3D versions. While this is helpful for navigating from building to building, it is not helpful when students are trying to navigate to classrooms they have never been to.  Many students have to talk to their classmates and faculty for assistance in finding these classrooms and during the first week of classes many students struggle to arrive on time to their lectures and labs. There are currently no pathfinding solutions for the WPI campus.

Proposed System

Overview:

The system our team will produce is PC software that allows users to obtain directions from one WPI campus location to another. It will display the recommended path to follow on maps of the campus and its buildings on a graphical user interface (GUI). Any individual can select a starting location and destination and our system will produce the shortest walkable route between them including navigation instructions. Additional route options will be produced to address user preference and convenience.

Functional Requirements:

These functional requirements were determined after evaluating the survey results and the open ended questions answered by 48 people. Unpopular features in the survey results were removed or reprioritized.

|  |  |
| --- | --- |
| **Priority** | **User Stories** |
| 1 | Turn by Turn instructions |
| 2 | ETA/Distance |
| 3 | Exportable to:   * Print * Image * Email |
| 4 | Map displays important landmarks such as:   * Bathrooms * Emergency blue towers * Elevators/Stairs * Vending machines, food courts * Emergency exits |
| 5 | Access to transportation Schedule   * SNAP * Gateway Shuttle |
| 6 | Disability access Routes |
| 7 | Search Location of Professors |
| 8 | Multiple destinations |
| 9 | Possible Mobile Application - Android |

Non-Functional Requirements

|  |  |
| --- | --- |
| **Priority** | **User Stories** |
| 1 | Efficient Execution – less than 4 seconds |
| 2 | Scalability of the User Interface |

Constraints (“Pseudo requirements”):

|  |
| --- |
| **User Stories** |
| The implementation language must be Java. |
| The application must work on desktop computers running Linux, Windows OS, or OS X. |
| The application must be made available for free download online. (Unless instructed otherwise) |
| The application must be efficient, and return results as fast as possible. |

System Model for Complete Application

Prioritized Product Backlog:

|  |  |
| --- | --- |
| **Priority** | **User Stories** |
| 1 | As an admissions employee, I want to enter paths on the campus map so the application is able to generate routes. |
| 2 | As a visitor, I want to know how to get to the dining hall. |
| 3 | As a new student, I want to know how to get to my physics lab. |
| 4 | As a user, I want turn by turn instructions on how to get where I need to go. |
| 5 | As a professor, I want know the path distance between Riley and AK for math problem. |
| 6 | As a student, I want to be able to print the route I plan to take. |
| 7 | As a student, I want to know how to get to my professor's office. |
| 8 | As a visitor, I want to know where the nearest bathroom is and how to get there. |
| 9 | As a freshman, I want to know when the next shuttle is, where it stops, and how to get there. |
| 10 | As a professor, I want to get a route from the parking lot to the campus center and then to my classroom. |
| 11 | As a student, if I don’t like my route I want to see another option. |

System Model for first Map Tool

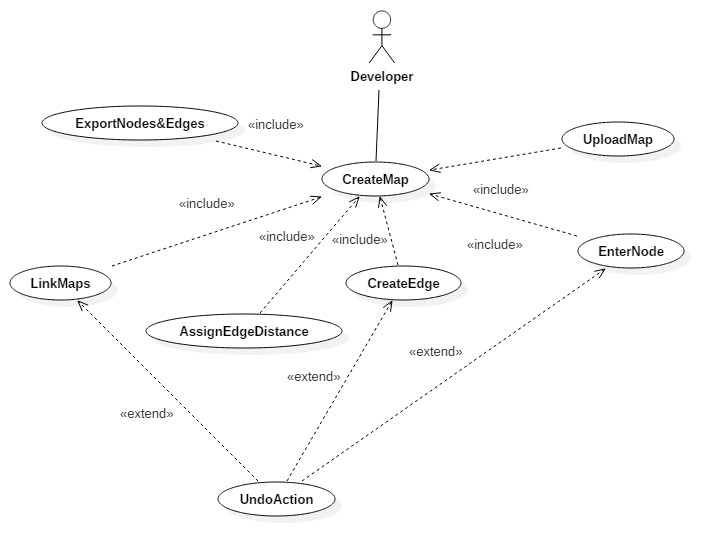
1. Epic Story: As a developer I want to easily construct a network of paths on a Map.
2. Map Tool Backlog:

|  |  |
| --- | --- |
| **Priority** | **User Stories** |
| 1 | As a developer, I want to be able to upload maps into the map tool so that I can enter nodes and paths. |
| 2 | As a developer, I want to visually enter nodes onto a map so that the routing application will have reference points. |
| 3 | As a developer, I want to create edges between nodes so that the program can plan routes between nodes. |
| 4 | As a developer, I want to create edges between nodes on different maps so that the program can plan routes between maps. |
| 5 | As a developer, I want to assign distance to edges so the program can find the shortest path. |
| 6 | As a developer, I want to export the nodes and edges so that I don’t have to hardcode nodes and edges for the path finding program. |

1. Scenario:

Jim is a developer at Pathfinding Studios, and he needs to set up a new map so that the Studio’s proprietary software Get There can find efficient paths on it.  He opens up their specialized development tool designed to do exactly that. He uses built in buttons to load an image of the new map. Then Jim clicks on the map in each place a pathfinding node is needed, and a circle is created on those spots to represent the nodes. After he places all of the required nodes, using other buttons to delete and move nodes as necessary, he needs to place paths to connect the nodes. He clicks on the ‘Pathways’ button and begins to click on different nodes. These nodes are then visually connected by lines on the map to represent pathways. Once he is done he clicks another button which creates a new Map class in the Get There program, and the new map is ready to be implemented.

1. User Case Diagram:



1. Textual Description:

**Name**: CreateMap

**Participating actor**: Developer

**Entry condition**: Developer executes the map tool

**Exit condition**: Nodes and edges are exported into a text file

**Flow of events**:

* Developer presses on upload map, include (uploadMap).
* Developer creates nodes, include (EnterNode) and edges, include(CreateEdge) and assigns edge distances, include(AssignEdgeDistance).
* Developer links a node on one map to a corresponding node on another map, if any, include(LinkMaps).
* Developer exports nodes and edges to a text file, include(ExportNodes&Edges).

**Alternative flow of events:**

[UndoAction]

* Developer presses on upload map, include (uploadMap).
* Developer creates node, include (EnterNode).
* Developer deletes node by hitting “Undo Action” button.

**Alternative flow of events:**

[UndoAction]

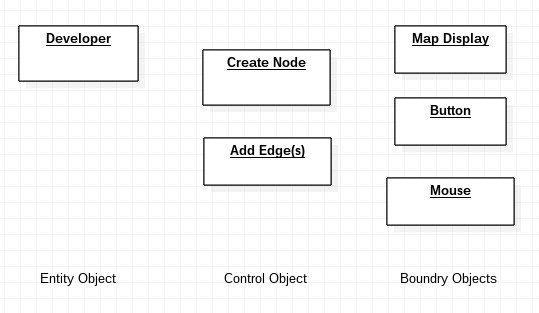
* Developer presses on upload map, include (uploadMap).
* Developer creates nodes, include (EnterNode) and an edge between them, and include (CreateEdge).
* Developer deletes edge by hitting “Undo Action” button.

**Alternative flow of events:**

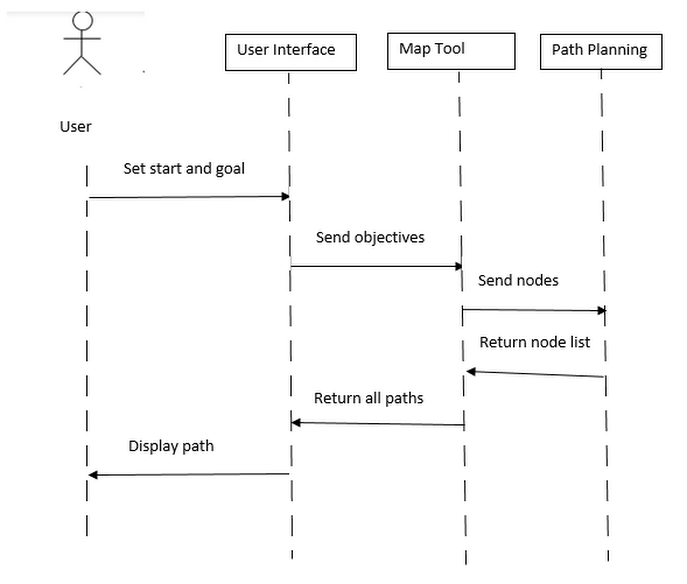
[UndoAction]

* Developer presses on upload map, include (uploadMap).
* Developer creates nodes, include (EnterNode) and edges, include(CreateEdge) and assigns edge distances, include(AssignEdgeDistance).
* Developer links a node on one map to a corresponding node on another map, include(LinkMaps).
* Developer deletes the link between maps

Object Diagram for Map Tool



Sequence Diagram for Map Tool



System Model for first Iteration

1. Epic Story
   1. To successfully navigate around WPI including interior buildings
2. Iteration Backlog:

|  |  |
| --- | --- |
| **Priority** | **User Stories** |
| 1 | As a student, I want to get to a classroom to attend my class. |
| 2 | As a student I want to reach Kaven from the campus center to attend my Civil Engineering Class. |
| 3 | As a student I want to reach the gym to attend my basketball class. |
| 4 | As a non-WPI student I want to find the Admissions Department to find campus tours. |

1. Scenarios:

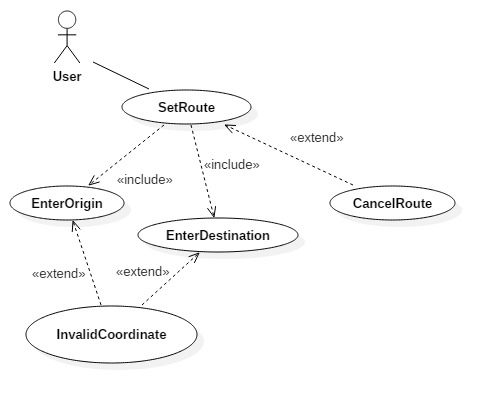
**Building to building**

Professor Frank is new to WPI. His lecture ended in Atwater Kent. He now needs to find the Campus Centre to buy his lunch.

Professor Frank pulls up the application to navigate through WPI on his laptop. He searches “Campus Centre” and sets it as his destinations. The application displays the campus map with a drawn path from Atwater Kent to the Campus Centre. He takes a picture of the map with his mobile phone and uses it to navigate to the Campus Center.

**Room to room**Bob, sitting in his physics class looks at his schedule and sees that he has a lab on the same floor following class. Bob is a freshman and does not know where anything is.  
Bob, launched the WPI navigation application his computer and enters in his current room. He then searches for the lab room number and selects “route”. The application then displays a map with the route from his class to his lab.

1. User Cases



1. Textual Description

**Name**: SetRoute

**Participating Actor**: Application user

**Entry Condition**:

* User executes the application on a desktop computer
* User enters the origin, i.e. the starting point of the trip
* User enters the destination, i.e. the ending point of the trip

**Exit Condition:** Passenger sees a map on the screen, with the shortest route from the origin to the destination. Passenger also receives turn-to-turn instructions, with how far to travel during each turn.

**Flow of Events:**

User inputs origin, and destination, using a keyboard, in the corresponding boxes

User presses the go button

Application generates the map, with a clear path from origin to destination

Application generates turn-to-turn instruction.

**Alternate Flow of Events:**

**InvalidCoordinate:**

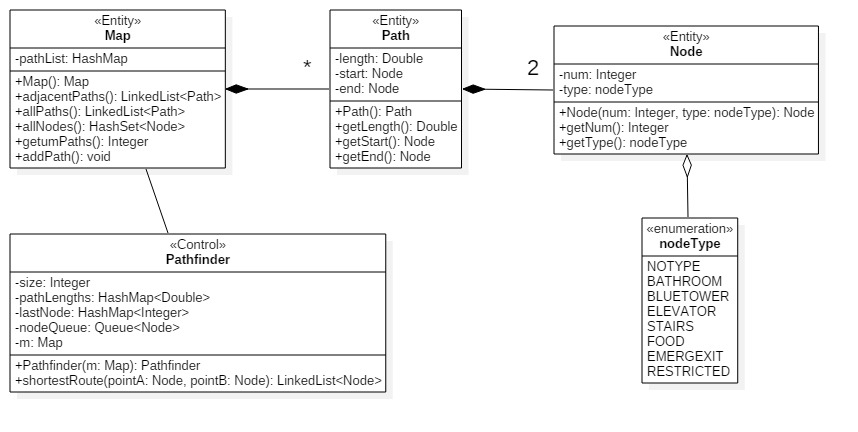
User inputs invalid origin or invalid destination or both

Requests the User to check the invalid input and re-enter it

**CancelRoute:**

The User cancels the route selection process and closes the application.

Object Model for first Iteration

Class Diagram:

Data Dictionary:

1. **Class Name:** Node

**Description:** A class used to help identify paths for the pathfinding software

**Attributes:**

Attribute Data Type\* Description

num int an identifier for the node

type nodeType an identifier for the type of node

**Methods:**

Method Description

Node(num: Integer, type: nodeType): Node Creation method

getNum(): Integer a getter for the num attribute

getType(): nodeType a getter for the type attribute

**Relationships**

Other Class Type of Relationship Description

Path Aggregate Two nodes are used by a path to define a start and end point.

1. **Class Name:** Path

**Description:** A class that is a traversable path on a map

**Attributes**

Attribute Data Type\* Description

length Double the length of the path

start Node the startpoint of the path

end Node the endpoint of the path

**Methods**

Method Description

Path(): Path Creation method

getLength(): Double A getter for the length attribute

getStart(): Node A getter for the start attribute

getEnd(): Node A getter for the end attribute

**Relationships**

Other Class Type Description

Node Aggregate Two nodes are used by a path to define a start and endpoint

Map Composite Many paths make up a traversable map of paths

1. **Class Name:** Map

**Description:** This class is used to create a map with the respective nodes

**Attributes**

Attribute Data Type\* Description

pathList HashMap Contains a hashmap of nodes

**Methods**

Method Description

Map(): Map Creates a map to hold the nodes

adjacentPaths(): LinkedList<Path> Creates a linked list of neighboring Paths

allPaths():LinkedList<Path> Creates a linked list of all Paths

allNodes(): HashSet<Node> Creates a Has set of Nodes

addPath(): void Nothing returned, adds a Path to a linked list

**Relationships**

Other Class Type Description

Path Composite A map contains many Paths stored in a Linked List

PathFinder Aggregate The pathfinder is used to find a path within a given map

1. **Class Name:** PathFinder

**Description:** Used to find a path between the given nodes

**Attributes**

Attribute Data Type\* Description

Size Integer length of the current path

pathLengths HashMap(double) Stores the various lengths of paths

lastNode HashMap<int> Stores the last node

nodeQueue Queue<Node> Stores a queue of Nodes

m Map The map in which the paths are found

**Methods**

Method Description

PathFinder(m:Map) Used to find a certain path within the

given map

shortestRoute(pointA: Node, pointB: Node) Used to find the shortest distance

between the give two points

**Relationships**

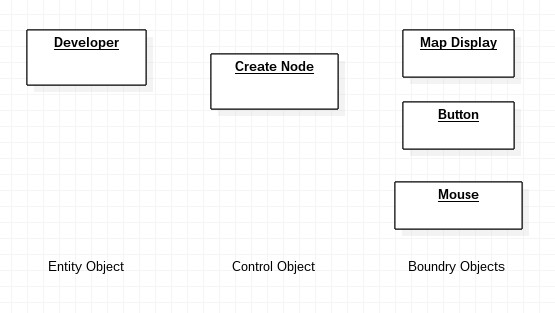
Other Class Type Description

Map Composite finds the paths within the map object

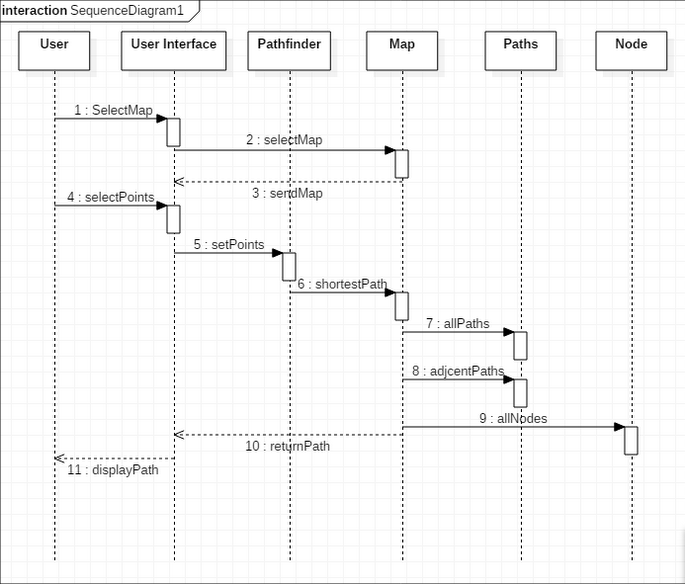
User Interface Mockup:



Boundary Object Diagram:



Dynamic Model for first Iteration



Interview Question Responses

What would you use our application for?

|  |
| --- |
| Interior Building Layouts for route planning |
| Maybe location of bathrooms near destination? |
| Integration into Google Maps |
| Classroom locations/ floors of buildings |
| Population density by time of day, based on where classes are scheduled and when they get out |
| Open parking spaces, current events |
| Push to Google maps on android |
| Mobile version |

Would you have any suggestions in addition to the features already mentioned?

|  |
| --- |
| I would imagine that an application on the phone would be the most widely accepted and used. if you could code it on android and is it would get the most possible traffic and use in my opinion. |
| Our campus is so small, we really don't need this |
| only really useful as mobile app |
| a phone app would be way more convenient! |
| A mobile app would be significantly more convenient to use (and thus I would use it more often). |
| this seems pointless? |
| Sounds amazing! |
| Pedometer to track how far you've traveled throughout the day |
| Will not use if it's on the desktop, but might use if it's on mobile. |
| Having your current location based on phone GPS would be helpful |
| This app should also provide snap schedules and be topographically accurate. I want to get around having to go uphill as little as humanly possible |
| I'm actually an alum, I would find the app useful if it told me what's happening on campus and where |

Glossary

**Node**: A point on a map.

**Edge**: The connection between two neighboring nodes.

**Hardcode**: Physically writing code instead of using a program to generate it.

Salary Distribution

|  |  |  |
| --- | --- | --- |
| **Developer** | **Salary (Total 90W)** | **Contribution for the week** |
| Chaves, Jefferey J. | 13.125W |  |
| Griffin, Christopher S | 13.125W |  |
| Kaiser, Joseph T | 13.125W |  |
| Parnas, Lumbini | 13.125W |  |
| Peterson, Paul W | 13.125W |  |
| Ruggiero, Alexander N | 13.125W |  |
| Sullivan, William | 13.125W |  |
| Touma, Jean Marc A | 13.125W |  |