Team: Bo Cao
Fernando Nobre
Zhongzhi Zhang

Title: Robot Path Learning

Project Summary:

The goal of this project is to allow users to navigate a robot through a series of random environments with obstacles. The user-data will be made available to a third-party system that will attempt to learn robotic path planning policies. These learned policies will be used to navigate robots in new environments without any user supervision.

All user interactions will happen through a web interface. The user should navigate the the robot to a predetermined location on the map, while avoiding obstacles. The inputs given by several users will then be used by the third-party system to train a convolutional neural network for path planning.

Project Requirements:

No Business Requirements

Functional Requirements				
ID	**Requirement**	**Topic Area**	**User**	**Priority**
FR-001	System shall only allow registered users to login.	Login	All	Critical
FR-002	System shall not allow regular users to perform administrative tasks.	Registration	All	Critical
FR-003	System shall only allow administrators to perform administrative tasks.	Designation	Admin	Medium
FR-004	System shall persist all user information when user is created or modified.	Management	Admin	Medium
FR-005	System shall present visitors with a login/signup page.	Management	Admin	Medium
FR-006	System shall direct user to map screen after	Control	All	High

	login.			
FR-007	System shall display map screen with a blank map.	Control	All	High
FR-008	Map screen shall always be of size 32 by 32 squares.	Management	All	High
FR-009	Map screen shall have fixed start position at (0,0)	Management	All	Medium
FR-010	Map screen shall have fixed end position at (31, 31)	Management	All	High
FR-011	Map screen shall have a side panel with "Edit", "Control" and "Cancel" buttons.	Management	All	High
FR-012	All screens shall have a navigation bar at the top.	Management	All	High
FR-013	Navigation bar shall have links to "Map", "Paths" and "Logout".	Management	All	High
FR-014	System shall attempt to move robot in a straight line from its starting position to where the user clicks on a map.	External System	All	Critical
FR-015	System shall create obstacles in the grid cells that a user clicks on in the Map.	External System	All	Critical
FR-016	System shall remove obstacles in the grid cells that the user right-clicks on, if that cell was occupied by an obstacle.	External System	All	Critical
FR-017	Users shall only be able to create obstacles that fill a 1x1 grid cell.	External System	All	Critical
FR-018	Users shall be able to create as many obstacles as there are free grid cells.	External System	All	Critical
FR-019	The system shall be configured with an initial administrator account with username 'admin' and password 'admin'.	Management	All	Critical
FR-020	User accounts shall be of two types: Regular user or Administrator	Management	All	Critical

FR-021	System shall require passwords that are at least 8 characters long.	Security	All	Critical
FR-022	System shall require passwords to contain only letters and/or numbers.	Security	All	Critical
FR-023	System shall ask visitor to confirm password when creating an account	Security	All	Critical

Non-Functional Requirements

ID	Requirement	Topic Area	User	Priority
NFR-001	Use of this system shall not require any knowledge of robotics or how robots work.	Usability	All	Critical
NFR-002	Pages in this system shall have a response time of up to 0.5s.	Usability	All	Medium
NFR-003	The system shall be usable without any specific documentation.	Usability	All	Low
NFR-004	System shall have same behaviour on the following browsers: Chrome, FireFox and Safari	Platform Constraints	All	High
NFR-005	When reloading the website, the system shall maintain the current state.	Reliability	All	High
NFR-006	The system shall not become unresponsive.	Reliability	All	High
NFR-007	The system shall be able to handle any exception without restarting.	Reliability	All	High
NFR-008	The system shall be able to support 1,000 concurrent users.	Performance	All	High
NFR-009	Users shall be directed to the main page within 3 seconds after validating login.	Performance	All	Critical
NFR-010	Robot shall respond to user command within 0.5 seconds.	Performance	All	High

NFR-011	The system will be hosted on a web hosting service.	Deployment	Developer	High
NFR-012	System shall present data to the user through a web interface.	Interface	Developer	High

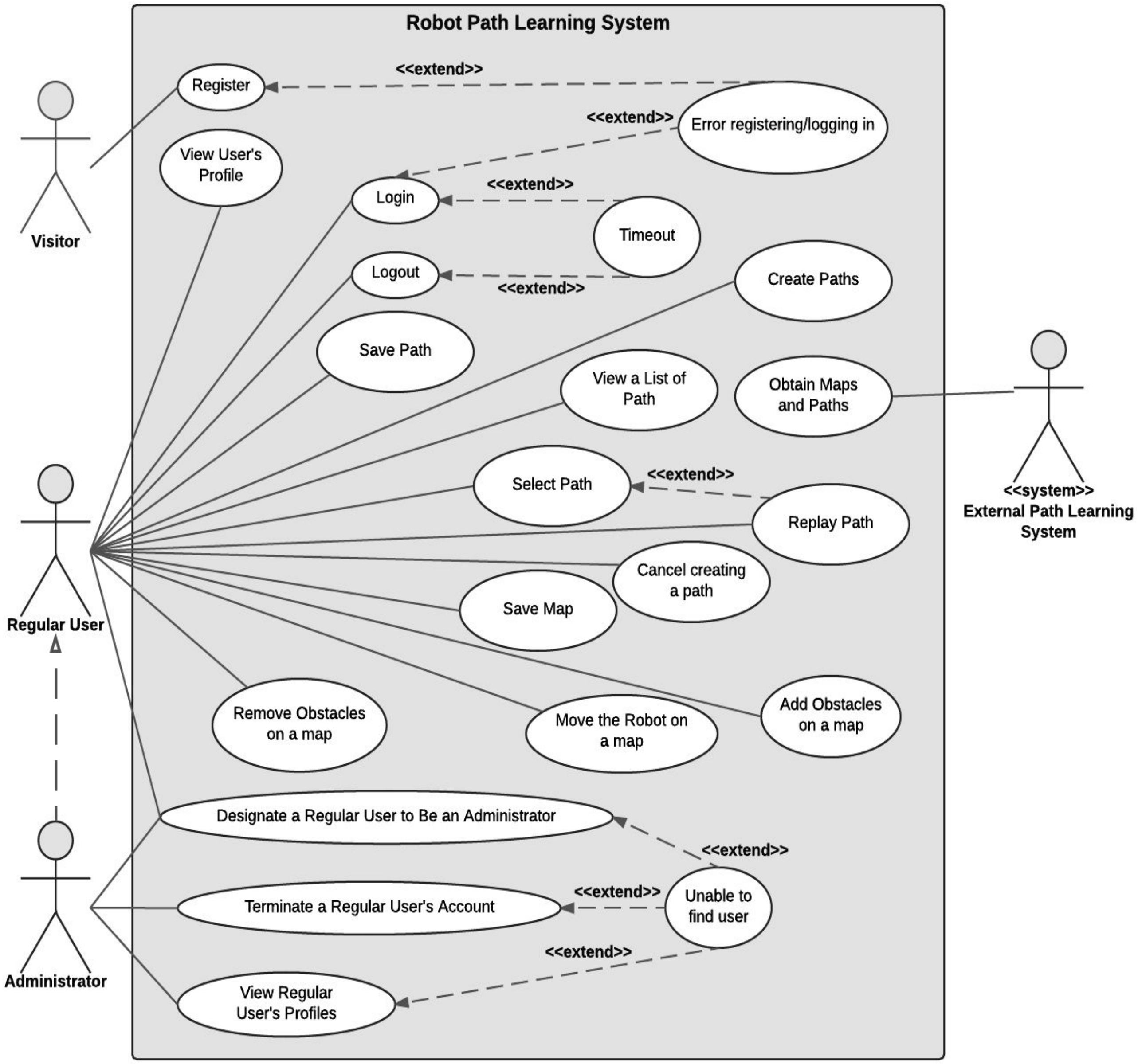
User Requirements

ID	**Requirement**	**Topic Area**	**User**	**Priority**
UR-001	As a visitor, I want to register to be a regular user.	Registration	Visitor	Critical
UR-002	As a user, I want to log in to the system.	Login	User	Critical
UR-003	As a user, I want to logout of the system.	Logout	User	Medium
UR-004	As a user, I want to view my profile.	Profile	User	High
UR-005	As a user, I want to be directed to the new map page when I login.	Navigation	User	Medium
UR-006	As a user, I want to see a list of the previous paths I've created.	Operation	User	Medium
UR-007	As a user I want to select one of the previous paths I created and see the details of the path.	Operation	User	Medium
UR-008	As a user I want to replay the paths I created before.	Operation	User	Medium
UR-009	As a user, I want to control a robot to move to the expected position where I click on the map.	Operation	User	High
UR-010	As a user, I want to add an obstacle in any available position on the map.	Operation	User	High
UR-011	As a user, I want to remove an obstacle in any occupied position on the map.	Operation	User	High
UR-012	As a user I want to be able to save paths I've created.	Operation	User	High
UR-013	As a user I want to be able to cancel	Operation	User	High

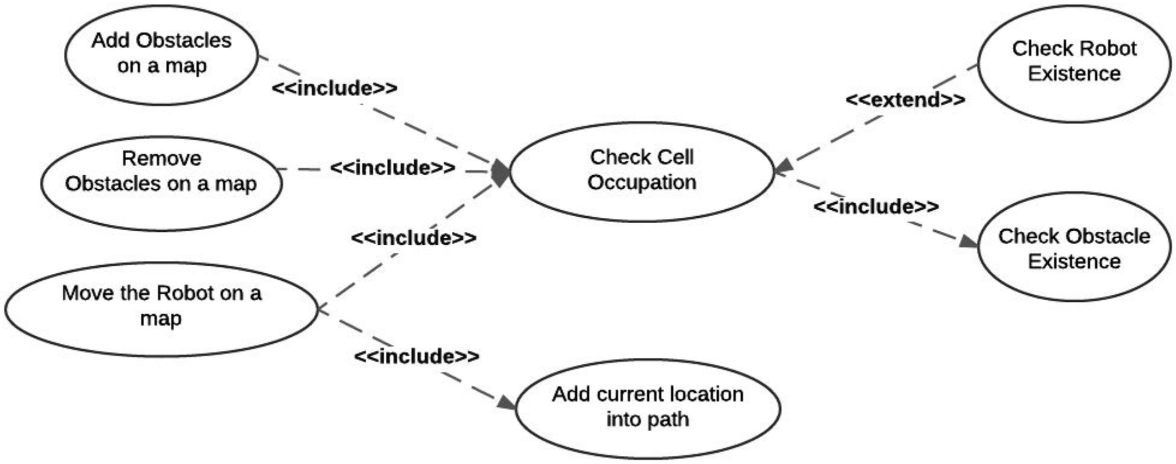
	creating a path.			
UR-014	As an administrator, I want to view all regular users’ profiles.	Management	Admin	High
UR-015	As an administrator, I want to designate a regular user as an administrator.	Management	Admin	Critical
UR-016	As an administrator, I want to terminate a regular user’s account.	Management	Admin	High
UR-017	As an external system, I want to be able to obtain all the maps and paths created by users.	Dataset	External System	Critical

Use Case Overview:

Use Case Diagram:



Use Case Sub-Diagram:



Use Case Documents:

User Case ID:	UR-001		
User Case Name:	Register to become a regular user.		
Description:	A visitor can register to become a regular user.		
Actors:	Visitor		
Pre-conditions:	Visitor can access the login webpage.		
Post-conditions:	Visitor will be notified whether his registration is successful or not. If yes, then he/she will be redirected to the new map screen.		
Frequency of Use:	Often by visitors		
Flow of Events:	**Actor Action**		**System Response**
	1	Fill out the registration form, including username and password.	Check duplicate username, and password strength.
	2	Click Register button.	Creates a new account for this new user and manipulate the

			profile’s data into the user database.
	3	View user profile.	The system will be redirected to his personal user profile page.
Variations	1. Re-enter the fields in the registration form.		
Notes and Issues:	1. Username has already been used. 2. The two passwords entered in the field of “Password” and “Confirm Password” does not match.		
Developer Notes			

User Case ID:	UR-002		
User Case Name:	Log in to the System		
Description:	User can log in to the system.		
Actors:	User		
Pre-conditions:	Visitor has registered as a user in the system. User can access login screen.		
Post-conditions:	User has successfully logged in to the system.		
Frequency of Use:	Frequently by users		
Flow of Events:	**Actor Action**		**System Response**
	1	Enter username and password.	Check field’s format and any blank fields.
	2	Click Login button.	Validate user by checking user’s username and password.
	3	View user profile.	Redirect to user’s profile page.

Variations	1. Re-enter username or email and password.
Notes and Issues:	1. User validation could fail.
Developer Notes	

User Case ID:	UR-003		
User Case Name:	Logout of the system		
Description:	User can logout of the system.		
Actors:	User		
Preconditions:	User has logged in to the system.		
Postconditions:	The session of the user will expire. Any unsaved paths will be stored.		
Frequency of Use:	Frequently by users		
Flow of Events:	Actor Action		System Response
	1	User click on the logout button on any page in the system	The webpage will be redirected to the login page and the current session will be cleared
Variations	The user who has already logout tries to logout again in other pages he previously opened.		
Notes and Issues:	If the user has opened multiple pages in the same browser before he logout, then if he is trying to do anything on other pages after the logout, the system will require him to login again at first		
Developer Notes			

User Case ID:	UR-004		
User Case Name:	View my profile		
Description:	A user views his profile.		
Actors:	User		
Preconditions:	User is logged in to the system.		
Postconditions:	The user's profile will be displayed.		
Frequency of Use:	Frequently throughout the day by users		
Flow of Events:	Actor Action		System Response
	1	User clicks on the the “My Profile” button or.	The webpage will be redirected to his personal profile page.
Variations			
Notes and Issues:	If the user is a regular user, his username and date created shall be displayed. If the user is an administrator, a list of all the active users will also be displayed.		
Developer Notes			

User Case ID:	UR-005
User Case Name:	Direct to the new map page when logged in.
Description:	A user will be directed to the new map page when he/she logs into the system.
Actors:	Users
Pre-conditions:	User has logged in successfully.
Post-conditions:	User will be directed to a webpage where a new map is created.

Frequency of Use:	Frequently throughout the day by users		
Flow of Events:	**Actor Action**		**System Response**
		None	The system will be redirected to a page where a map is created automatically as soon as the login use case is completed successfully.
Variations			
Notes and Issues:			
Developer Notes			

User Case ID:	UR-006		
User Case Name:	View created paths		
Description:	A User can see all the path's he has created.		
Actors:	User		
Pre-conditions:	User has successfully logged in.		
Post-conditions:	User can see all the path's he has created		
Frequency of Use:	Occasionally by Users		
Flow of Events:	**Actor Action**		**System Response**
	1	User clicks on the "My Paths" button in the page header.	System will retrieve all the paths the user has created and display a list of them to the user.
Variations			

Notes and Issues:	If no paths have yet been created, a message “no previous paths” will be presented
Developer Notes	

<table>
<tr><td><b>User Case ID:</b></td><td colspan="3">UR-007</td></tr>
<tr><td><b>User Case Name:</b></td><td colspan="3">Select and view one of the previous paths</td></tr>
<tr><td><b>Description:</b></td><td colspan="3">User can select and view one of the previous paths.</td></tr>
<tr><td><b>Actors:</b></td><td colspan="3">User</td></tr>
<tr><td><b>Pre-conditions:</b></td><td colspan="3">User has created at least one path before
Visitor was on the listing page of the previous paths he created.</td></tr>
<tr><td><b>Post-conditions:</b></td><td colspan="3">The user was on a new page with a map which show the path he created</td></tr>
<tr><td><b>Frequency of Use:</b></td><td colspan="3">Frequently by users</td></tr>
<tr><td rowspan="2"><b>Flow of Events:</b></td><td colspan="2"><b>Actor Action</b></td><td><b>System Response</b></td></tr>
<tr><td>1</td><td>User click on one of the paths in the list</td><td>The webpage will be redirected to a new one where shows the map which contains the path he created for that map</td></tr>
<tr><td><b>Variations</b></td><td colspan="3"></td></tr>
<tr><td><b>Notes and Issues:</b></td><td colspan="3"></td></tr>
<tr><td><b>Developer Notes</b></td><td colspan="3"></td></tr>
</table>

User Case ID:	UR-008

User Case Name:	Replay the path		
Description:	User can let the robot move along the path he created before and draw the trace behind it.		
Actors:	User		
Pre-conditions:	Visitor was on the page of which is showing one of the paths he created before on a map with edited before.		
Post-conditions:	The robot will be moving along the path on the map and leaving aa painted trajectory behind it.		
Frequency of Use:	Frequently by users		
Flow of Events:	**Actor Action**		**System Response**
	1	User click on the “Play” button on that page	The robot will move along the path on the map from Start to the End, leaving a painted trajectory behind it.
Variations			
Notes and Issues:			
Developer Notes			

User Case ID:	UR-009
User Case Name:	Control Robot
Description:	A User can move the robot to a position on the map by clicking on the map.
Actors:	User
Pre-conditions:	User has successfully logged in. User is in the new map page.

Post-conditions:	Robot moves in a straight line in the direction that the user has clicked.		
Frequency of Use:	Occasionally by Users		
Flow of Events:	**Actor Action**		**System Response**
	1	User clicks on the "Control" button in the map page.	System will go into "control" mode, where the user can click on the map to direct the robot.
	2	User clicks on a location on the map.	System attempts to move the robot from its current location to the coordinates on the map where the user has clicked. System will check if the click is in an unoccupied position in the map grid, if there is an obstacle between the robot and the desired location. If the robot's end position is different than it's start position, the updated position is stored in the path.
Variations			
Notes and Issues:			
Developer Notes			

User Case ID:	UR-010
User Case Name:	Add obstacle to map
Description:	A User can add an obstacle to the map by clicking on the map.
Actors:	User

Pre-conditions:	User has successfully logged in. User is in the new map page.		
Post-conditions:	Obstacle is added to map in desired location, if possible.		
Frequency of Use:	Occasionally by Users		
Flow of Events:	**Actor Action**		**System Response**
	1	User clicks on the "Edit" button in the map page.	System will go into "edit" mode, where the user can click on the map to direct the robot.
	2	User clicks on a location on the map.	System checks if position clicked by user is already occupied. If not, an obstacle is added. If it is already occupied (either by an obstacle or by the robot) then nothing happens.
Variations			
Notes and Issues:			
Developer Notes			

User Case ID:	UR-011	
User Case Name:	Remove obstacle from map	
Description:	A User can remove an obstacle to the map by clicking on the map.	
Actors:	User	
Pre-conditions:	User has successfully logged in. User is in the new map page.	
Post-conditions:	Obstacle is removed from the map in desired location, if possible.	
Frequency of Use:	Occasionally by Users	
Flow of Events:	**Actor Action**	**System Response**

	1	User clicks on the "Edit" button in the map page.	System will go into "edit" mode, where the user can click on the map to direct the robot.
	2	User right clicks on a location on the map.	System checks if position clicked by user is already occupied. If yes, an obstacle is removed. If it is not occupied then nothing happens.
Variations			
Notes and Issues:			
Developer Notes			

User Case ID:	UR-012		
User Case Name:	Save created paths		
Description:	A user can save path's he has created.		
Actors:	User		
Pre-conditions:	User is in map page and has moved the robot to a location different than the starting point.		
Post-conditions:	Path is saved.		
Frequency of Use:	Occasionally by User		
Flow of Events:	**Actor Action**		**System Response**
	1	User clicks on any button that will navigate away from the map page.	System will save the path created by the user. If the path is incomplete, it will mark it as such.

Variations	
Notes and Issues:	
Developer Notes	

User Case ID:	UR-013		
User Case Name:	Cancel Path		
Description:	A user can cancel path's he is creating.		
Actors:	User		
Pre-conditions:	User is in map page.		
Post-conditions:	Path is not saved. New map is shown		
Frequency of Use:	Occasionally by User		
Flow of Events:	**Actor Action**		**System Response**
	1	User clicks on "Cancel" button.	System will discard the current path and reload the map page with a blank map.
Variations			
Notes and Issues:			
Developer Notes			

User Case ID:	UR-014
User Case Name:	View Regular Users’ Profiles
Description:	An administrator can view regular users’ profiles.

<table>
<tr><td><b>Actors:</b></td><td colspan="3">Administrator</td></tr>
<tr><td><b>Pre-conditions:</b></td><td colspan="3">Administrator has successfully logged in.</td></tr>
<tr><td><b>Post-conditions:</b></td><td colspan="3">All the users profiles are displayed.</td></tr>
<tr><td><b>Frequency of Use:</b></td><td colspan="3">Occasionally by Administrators</td></tr>
<tr><td rowspan="2"><b>Flow of Events:</b></td><td colspan="2"><b>Actor Action</b></td><td><b>System Response</b></td></tr>
<tr><td>1</td><td>Administrator clicks on the "My Profile" button in the map page.</td><td>System will open administrator's profile and display a list of all users in the system.</td></tr>
<tr><td><b>Variations</b></td><td colspan="3"></td></tr>
<tr><td><b>Notes and Issues:</b></td><td colspan="3">If no users are registered an empty list shall be displayed.</td></tr>
<tr><td><b>Developer Notes</b></td><td colspan="3"></td></tr>
</table>

<table>
<tr><td><b>User Case ID:</b></td><td colspan="3">UR-015</td></tr>
<tr><td><b>User Case Name:</b></td><td colspan="3">Designate regular user as administrator</td></tr>
<tr><td><b>Description:</b></td><td colspan="3">An administrator can designate a regular user to be an administrator.</td></tr>
<tr><td><b>Actors:</b></td><td colspan="3">Administrator</td></tr>
<tr><td><b>Pre-conditions:</b></td><td colspan="3">Administrator has successfully logged in. Administrator is in "my profile" page.</td></tr>
<tr><td><b>Post-conditions:</b></td><td colspan="3">User is promoted to administrator. Status message is displayed.</td></tr>
<tr><td><b>Frequency of Use:</b></td><td colspan="3">Occasionally by Administrators</td></tr>
<tr><td rowspan="2"><b>Flow of Events:</b></td><td colspan="2"><b>Actor Action</b></td><td><b>System Response</b></td></tr>
<tr><td>1</td><td>Administrator clicks on the desired user</td><td>System will open user's profile page.</td></tr>
</table>

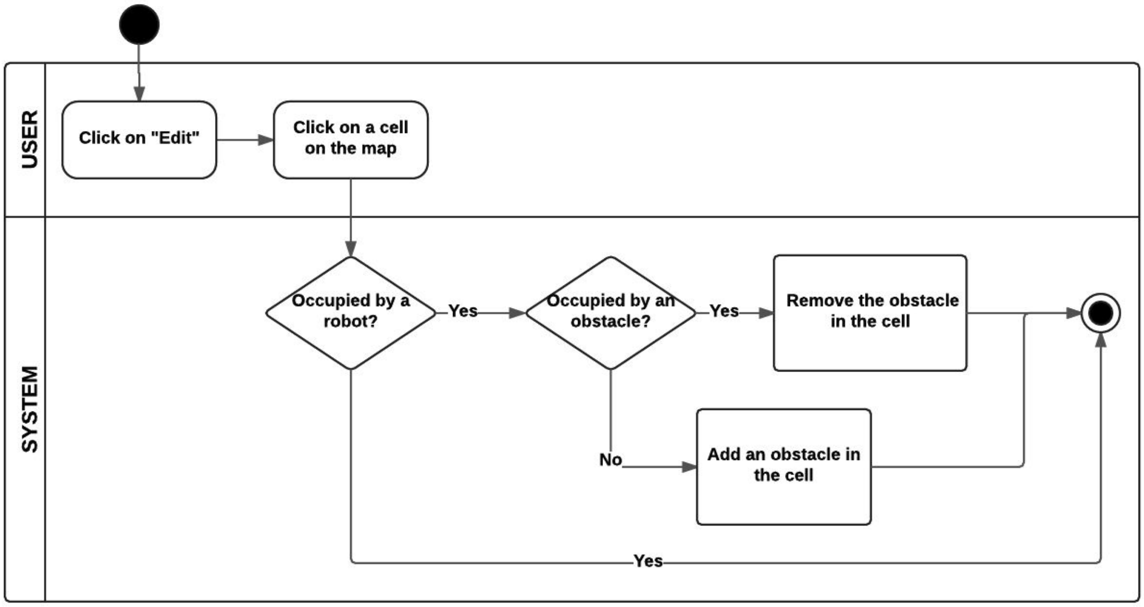
	2	Administrator clicks on "promote user"	System will attempt to change user to administrator status.
Variations			
Notes and Issues:	If the system is unable to find the user or edit the user an error message will be displayed		
Developer Notes			

User Case ID:	UR-016		
User Case Name:	Terminate user's account		
Description:	An administrator can terminate a regular user's account.		
Actors:	Administrator		
Pre-conditions:	Administrator has successfully logged in. Administrator is in "my profile" page.		
Post-conditions:	User is deleted. Status message is displayed.		
Frequency of Use:	Occasionally by Administrators		
Flow of Events:	**Actor Action**		**System Response**
	1	Administrator clicks on the desired user	System will open user's profile.
	2	Administrator clicks on "delete user" button	System will attempt to delete the user.
Variations			
Notes and Issues:	If there is an error accessing the user or deleting an error message will be displayed.		
Developer Notes			

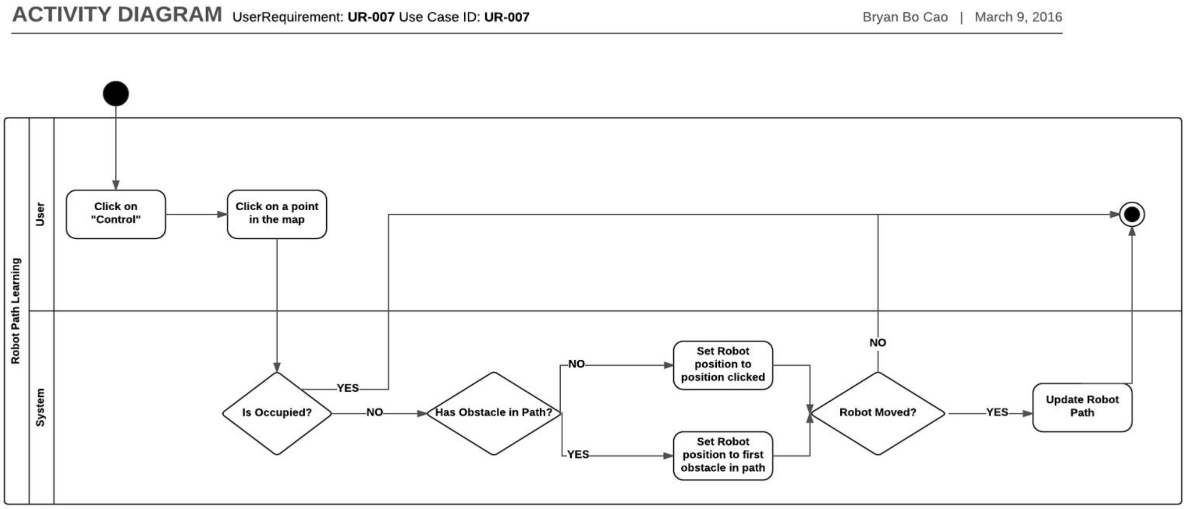
User Case ID:	UR-017		
User Case Name:	Retrieve maps and paths		
Description:	An external system can retrieve all the maps and paths created by all the users.		
Actors:	External system		
Pre-conditions:	System is operational and accessible.		
Post-conditions:	Maps and paths are provided to external system		
Frequency of Use:	Occasionally by external system		
Flow of Events:	**Actor Action**		**System Response**
	1	External System makes an API call to request maps and paths created by users	System will obtain all the maps and paths and return a list in serialized form to the requesting system.
Variations			
Notes and Issues:	If there are no maps and/or paths created, an empty list is returned.		
Developer Notes			

Activity Diagram:

Team member: Bo Cao
UR-010 A User can add an obstacle to the map by clicking on the map.
UR-011 A User can add an obstacle to the map by clicking on the map.


USER
SYSTEM
Click on "Edit"
Click on a cell on the map
Occupied by a robot?
Yes
Occupied by an obstacle?
Yes
Remove the obstacle in the cell
No
Add an obstacle in the cell
Yes

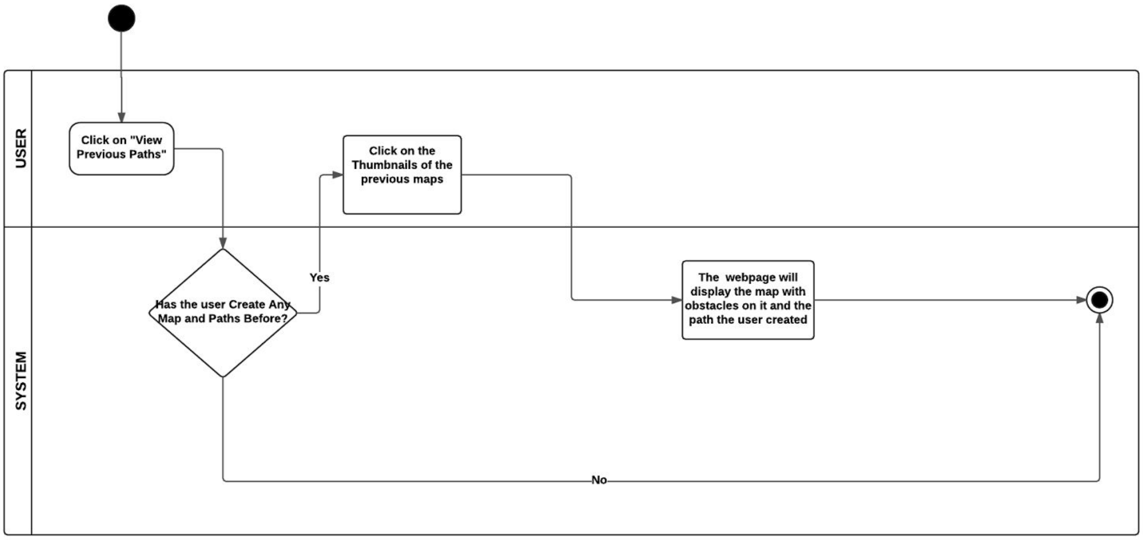
Team member: Fernando
UR-009 A User can move the robot to a position on the map by clicking on the map.


ACTIVITY DIAGRAM UserRequirement: UR-007 Use Case ID: UR-007
Bryan Bo Cao | March 9, 2016
Robot Path Learning
User
System
Click on "Control"
Click on a point in the map
Is Occupied?
YES
NO
Has Obstacle in Path?
NO
YES
Set Robot position to position clicked
Set Robot position to first obstacle in path
Robot Moved?
NO
YES
Update Robot Path

Team member: Zhongzhi Zhang

UR-006 A User can see all the path's he has created.

UR-007 User can select and view one of the previous paths.


USER
SYSTEM
Click on "View Previous Paths"
Has the user Create Any Map and Paths Before?
Yes
Click on the Thumbnails of the previous maps
The webpage will display the map with obstacles on it and the path the user created
No

Data Storage:

We will be using **MySQL** to store all the data, and **Hibernate** for mapping classes into database tables.

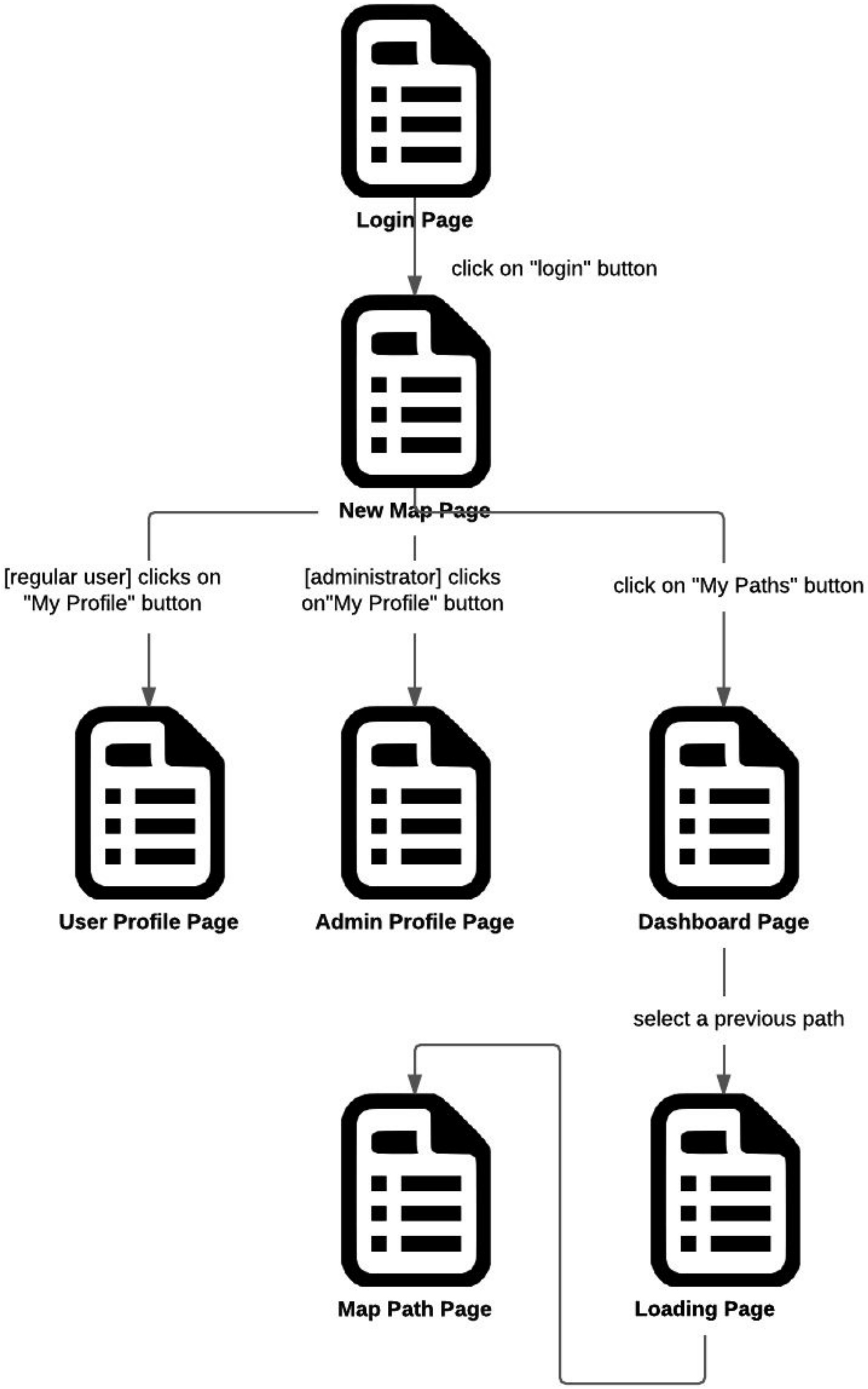
Data Storage: MySQL

Classes:

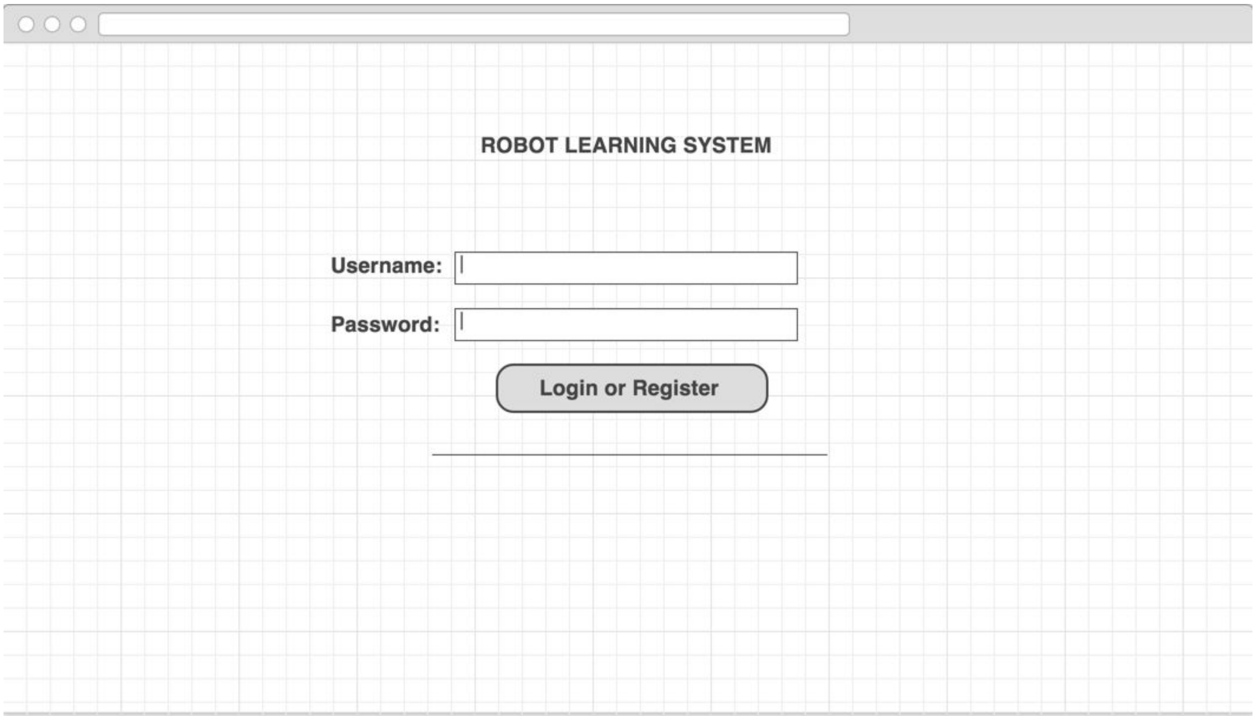
- UserModel class to store information about users.
- AdministratorModel class to store administrator data.
- MapModel class to store information about the map.
- GridModel class to store information about the grid.
- RobotModel class to store information about the robot.
- ObstacleModel class to store information about obstacles.
- PathModel class to store information about paths.
- CellModel class to store information about cells.
- RobotCurrentLocationModel class to store information about the current location of the robot.

UI Mockups:

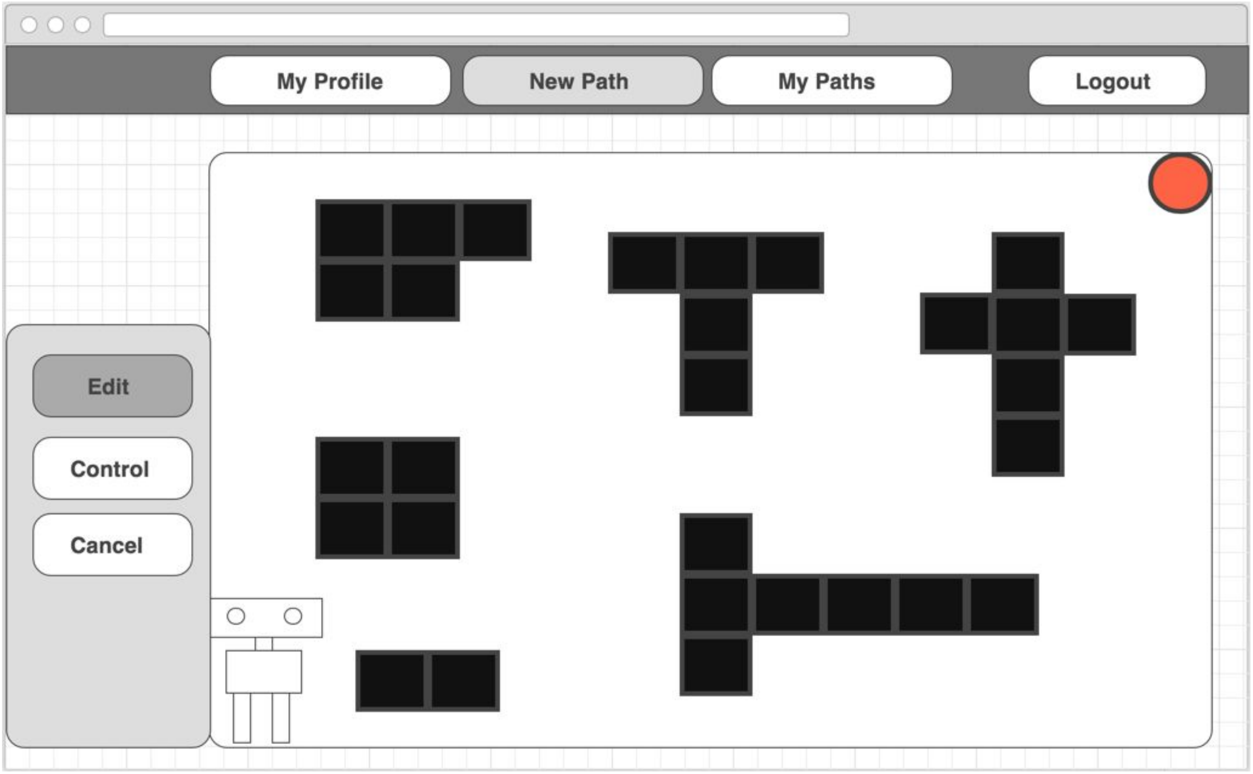
Page Overview:



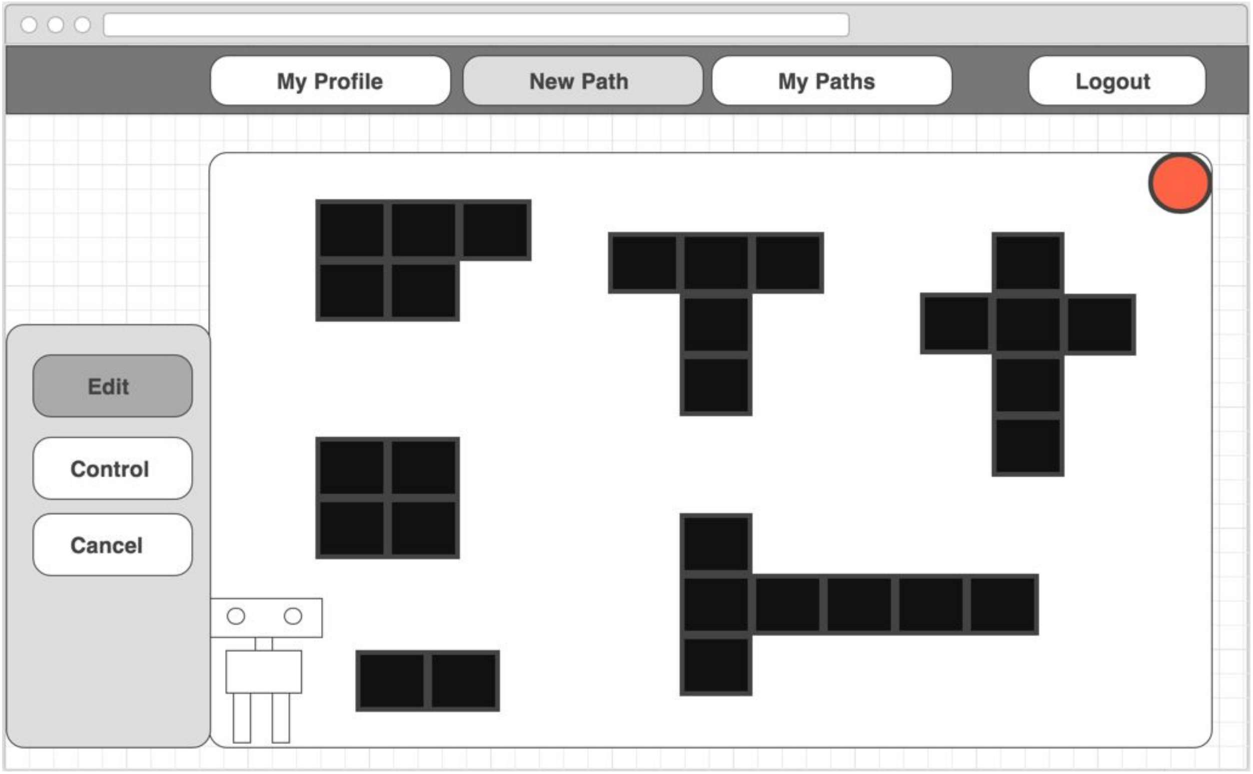
Login Page:



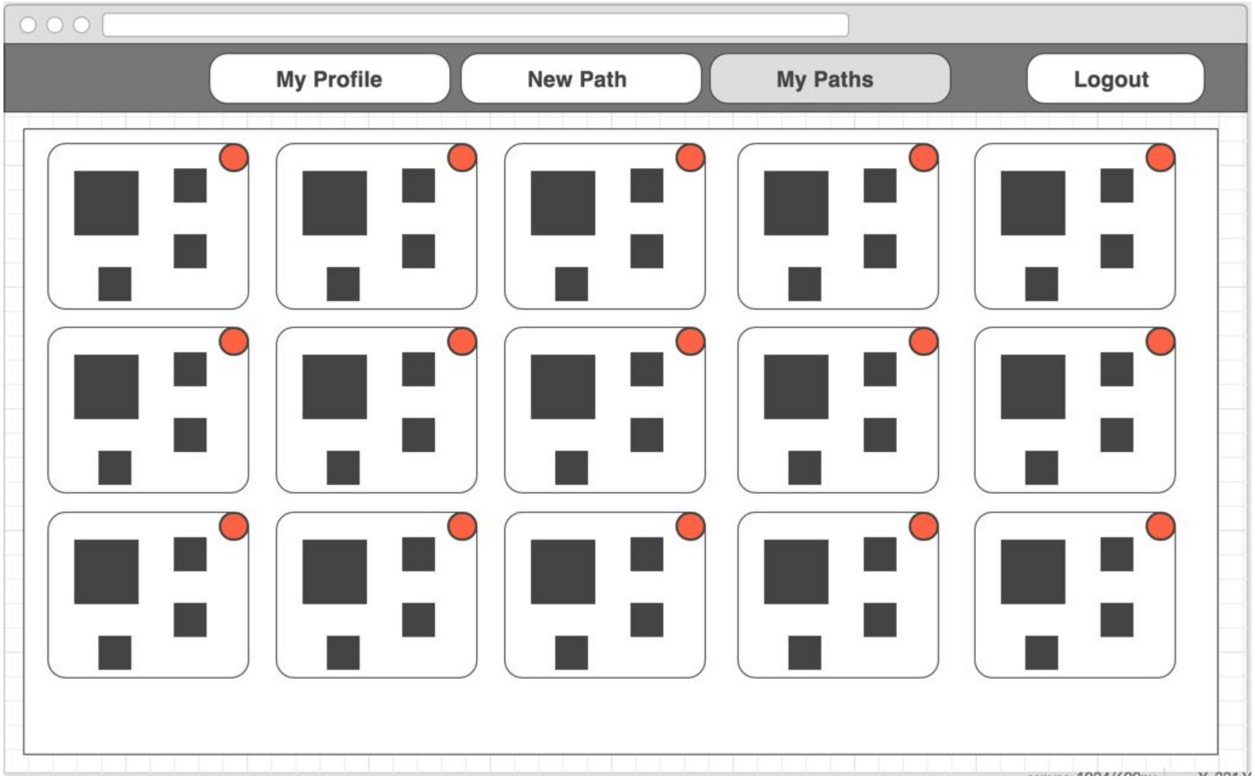
New Map Page:



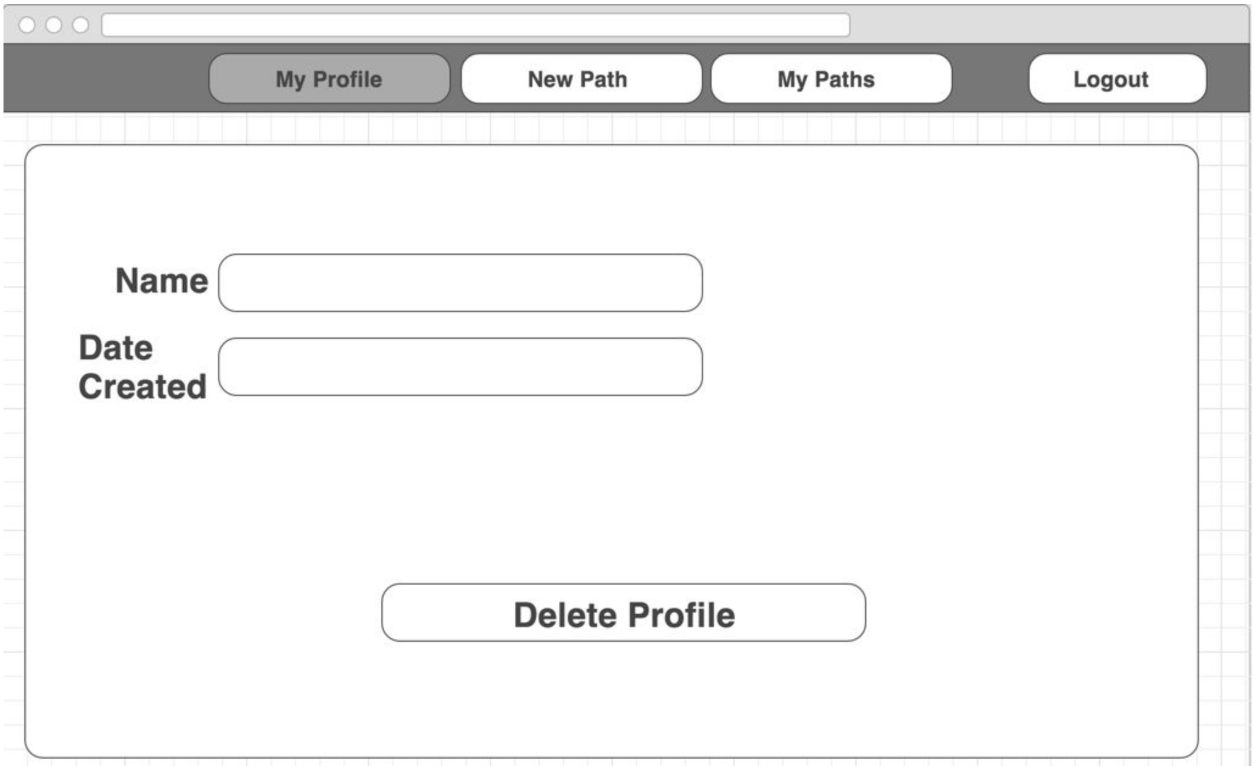
Map Path Page:



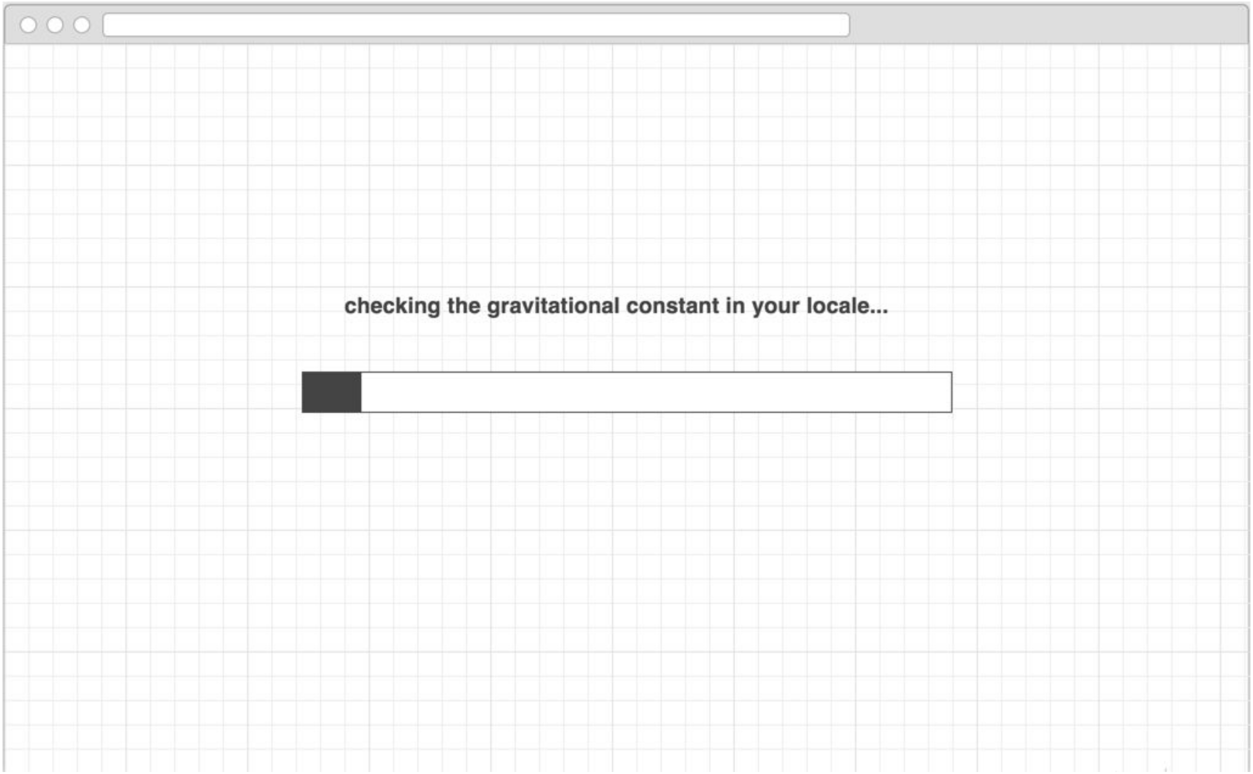
Dashboard Page:



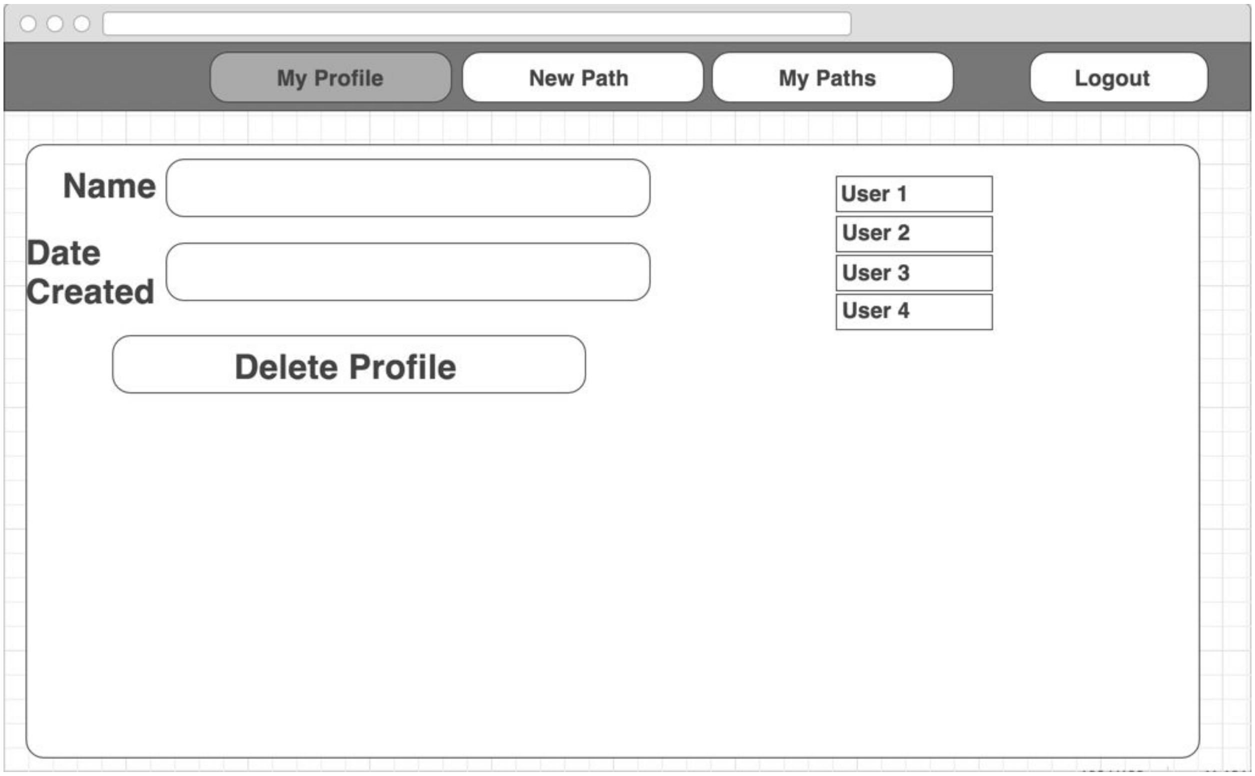
User Profile Page:



Loading Page:


checking the gravitational constant in your locale...

Admin Profile Page:



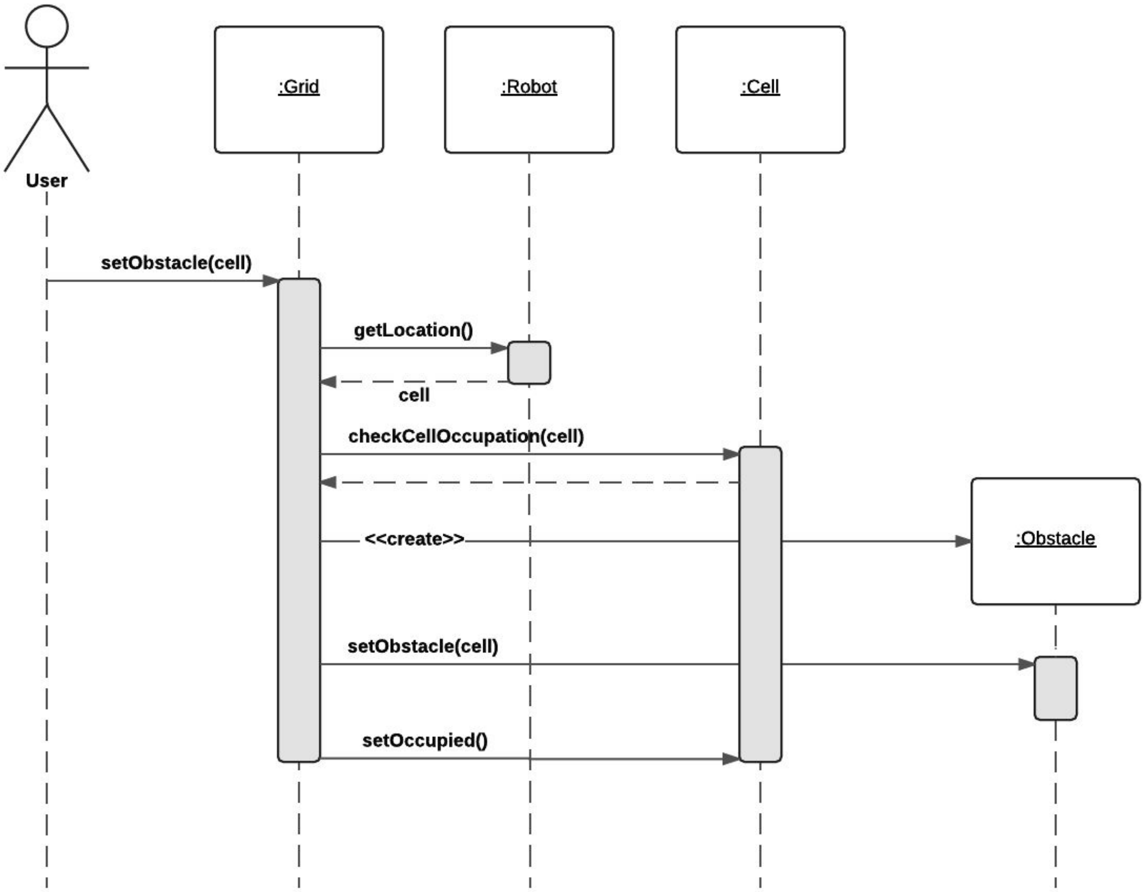
User Interactions:

Sequence Diagram:

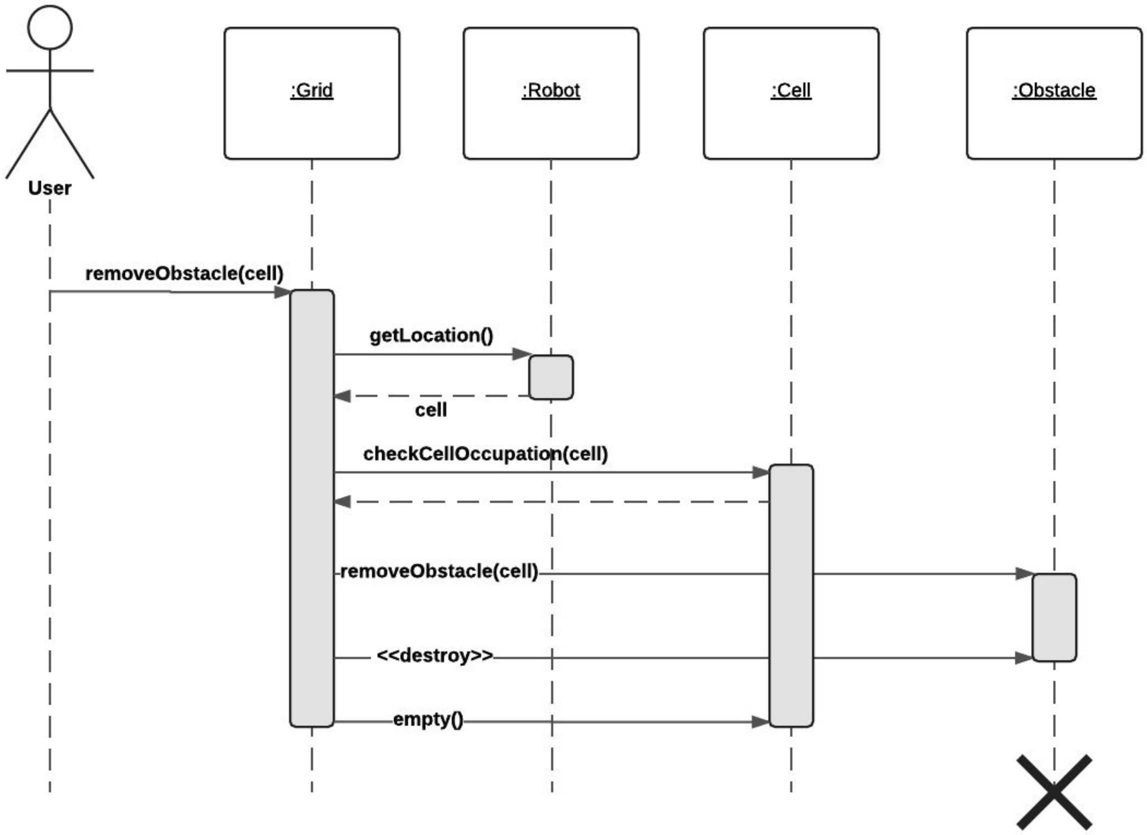
Team member: Bo Cao

UR-010 A User can add an obstacle to the map by clicking on the map.
UR-011 A User can add an obstacle to the map by clicking on the map.

Sequence Diagram #1:
condition: Cell occupied = false (not occupied by the robot, not occupied by the obstacle)

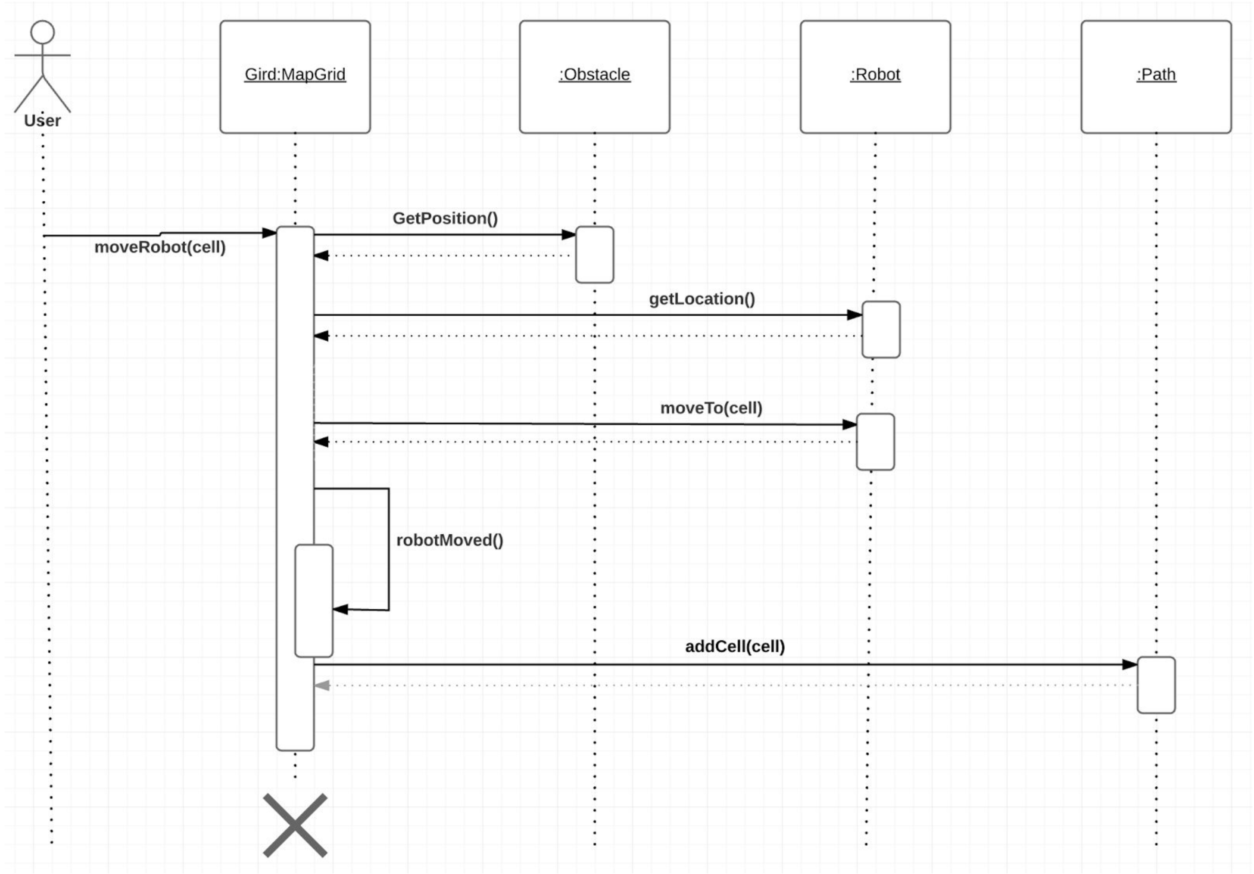


Sequence Diagram #2:
condition: Cell occupied = true (not occupied by the robot, occupied by the obstacle)



Team member: Fernando

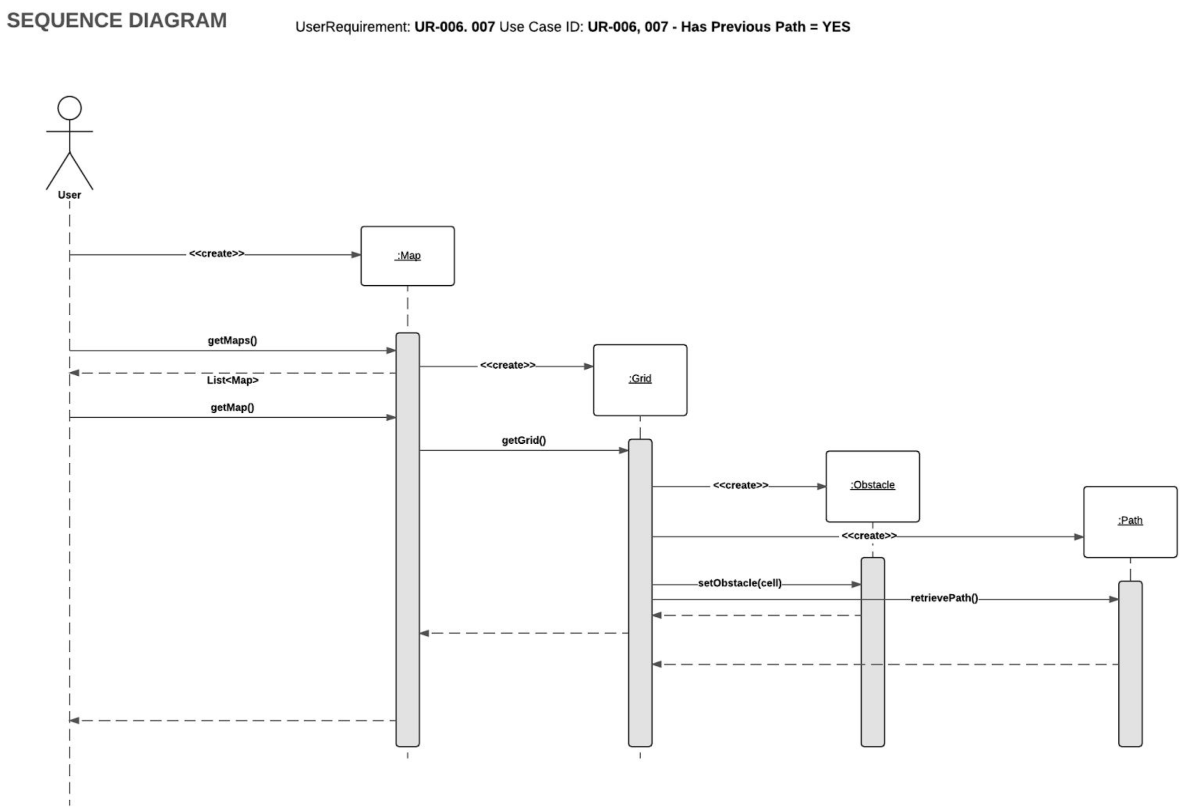
UR-009 A User can move the robot to a position on the map by clicking on the map.



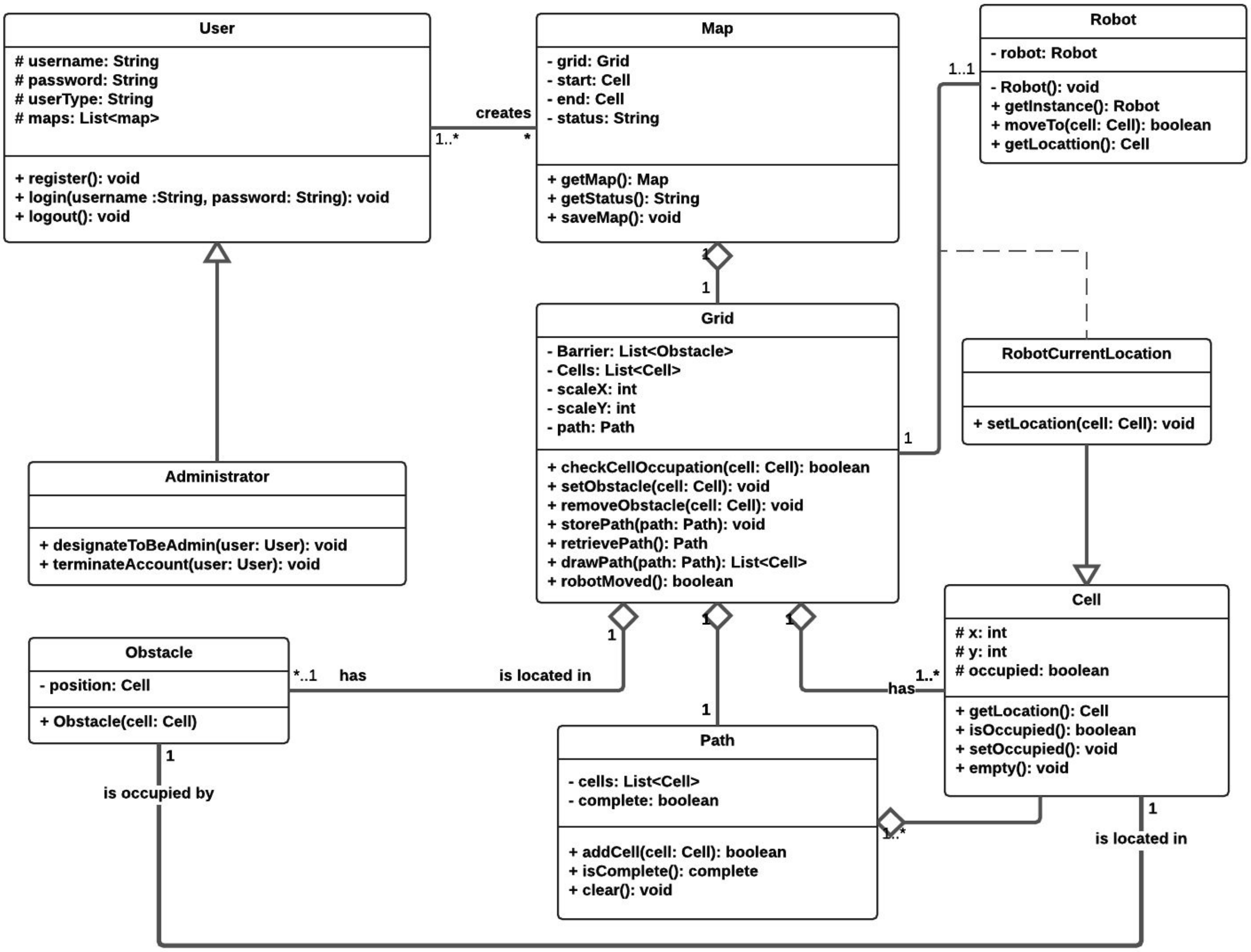
Team member: Zhongzhi Zhang

UR-006 A User can see all the path's he has created.

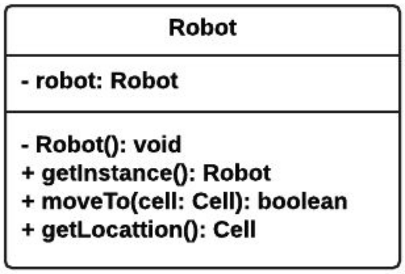
UR-007 User can select and view one of the previous paths.



Class Diagram:



We use **Singleton** Design Pattern, we don't allow more than one robots to be instantiated. The Robot Class is depicted as below:


Robot
- robot: Robot
- Robot(): void
+ getInstance(): Robot
+ moveTo(cell: Cell): boolean
+ getLocattion(): Cell