# Log Book

25/10/2018

Worked on data structure to store information about building layouts.

* Created “Wall” class
* Created “Building” class

Encountered issues with using JavaFX to display the GUI for the desktop application- the JDK is not set up correctly to work with JavaFX. In version 11 of the JDK JavaFX has been moved outside of the code Java libraries.

26/10/2018

Continued to look into the issue with JavaFX.

Investigating using Maven to manage dependencies.

28/10/2018

Figured out how to use JavaFX version 11 with version 11 of the JDK.

Updated the project to use Maven, and applied the fix for JavaFX to the project:

* Add JavaFX as a dependency to Maven pom.xml
* Edit the way IntelliJ builds the project to make sure it compiles the pom.xml- otherwise it will cannot find the correct runtime

Video on setting up Maven and JavaFX: <https://www.youtube.com/watch?v=r_tdK8zWr_w>

IntelliJ Java Version Setup: [https://stackoverflow.com/questions/21006136/intellij-idea-13-uses-java-1-5-despite-setting-to-1-7#](https://stackoverflow.com/questions/21006136/intellij-idea-13-uses-java-1-5-despite-setting-to-1-7)

JavaFX Scene Graph: <https://docs.oracle.com/javafx/2/scenegraph/jfxpub-scenegraph.htm>

29/10/2018

Created a product backlog. Added user stories for the prototype to the backlog.

30/10/2018

Researched using a BorderPane for the layout of the GUI.

<https://examples.javacodegeeks.com/desktop-java/javafx/javafx-borderpane-example/>

<https://docs.oracle.com/javafx/2/layout/builtin_layouts.htm>

<https://docs.oracle.com/javase/8/javafx/api/javafx/scene/Scene.html>

Worked on rendering the layout of the building to the screen:

* Created a Canvas and a Graphics Context which can be used to draw to the canvas.
* Created a static method on the Building class that creates mock building layout for testing and prototyping purposes.

01/11/2018

Created an AnimationTimer which allows the application to display moving images.

Fixed an issue where the animation was very slow, and became slower the longer the application was open; the problem was that the method drawWalls() contained a memory leak and would continue to add the same data to the data structure over and over, rather than clear it in between method calls. The fix was to use beginPath() on the graphics context to clear the data structure.

02/11/2018

Created an array to store information about the positions of the people, and a mock testing method which will allow the GUI to be tested that it is reading the data structure properly. Wrote a method to draw the locations of the people to the GUI; this will be extended in the future. Currently I am planning the next steps to be:

1. Add a schedule to people that they will follow, now that they can be observed on the GUI
2. Write unit tests using Junit for the current code that I have, to check that the methods return what is expected in a variety of test cases.

07/11/2018

Started work on implementing an A\* pathfinding algorithm to allow the people to move around the building. Created a new class to store information about coordinates which is used in the pathfinding algorithm.

08/11/2018

Finished work on implementing the pathfinding algorithm. The algorithm can currently find a path if one exists; if no path exists, the person will simply not move- this behaviour may need to be adjusted in future.

14/11/2018

Added an edit mode to the program. The program uses a pane to display rectangles and circles in the edit mode, which will be made to be clickable in the future if the user wants to edit their position.

15/11/2018

Added a toolbar with pause/play button, speed up and slow down buttons. Added the ability for the user to speed up or slow down the animation- this currently has some bugs which occur when the animation plays too fast.

16/11/2018

Fixed bugs which occur by running the animation too fast. A limit was imposed on the speed the animation could be run at to prevent the bugs that occur when it runs too fast. A possible solution if the user wants to run a very fast simulation without watching it would be to add functionality that instantly simulates a day (without animating it) and reports statistics about the day to the user- for example, how much certain facilities are used.

22/11/2018

Started work on a menu to add a person. The menu currently does not feature all the necessary GUI components to work properly. I am currently investigating how to best display the person’s schedule to the user- tables in JavaFX are not editable by default.

23/11/2018

Started work on the loading/saving files functionality. The file format used will be .xml as this is a standard format that has been proven to work for storing a variety of data, and Java has good functionality for reading xml files.

24/11/2018

Finished off the loading/saving files. The application features a window where the user can choose which file to open and where to save their file. The application features both a save and save as method. There are some minor bugs with the functionality, for example the file extension does not always set correctly as .xml.

29/11/2018

Added a color picker to the add person menu. The button to add the person to the data structure now works, but it will probably have some bugs when this menu is used to edit the person- the id of the person who is being edited may not be saved properly so the program won’t know if this is an existing or new person. Added a unique UI field to the person class, which is currently initialised when any new person object is created.

30/11/2018

Fixed a bug with the .xml file where it could not read the people stored in it. Fixed another bug- when loading a file with 0 walls and 1 person the method would throw an exception due to trying to iterate through the (wrong) empty array. This was fixed changing the method to iterate through the correct array that is populated.

09/12/2018

Added a re-sizeable table class for the GUI. This can be used by the user to schedule their person’s tasks. They can add as many rows as they want to the table and delete rows as necessary. An improvement to this table would be to allow the user to click and drag rows to re-order them, however this is an advanced optional functionality that there probably won’t be time to implement.

12/12/2018

Fixed a bug where the .xml file would not save the schedule correctly due to the method not adding the array of activity elements to the .xml file. Added the ability for the player to set a user-friendly “name” for each person so that they can remember who they are. Added this name field to the .xml file so that it would be saved.

14/12/2018

Fixed a bug where, upon adding a new person to the data structure, they would not be shown in the building until the animation was played. Fixed a bug where the dot representing a person was twice the size in edit mode as in animation mode. Discovered a bug where the displayed x and y position of a person varies in edit mode and animation mode (this is a GUI layer bug not a back-end logic bug). Started investigating how to click on a person’s circle to edit their information- this will most likely require an object wrapper which associates each circle with a person’s id. This could also be achieved by checking the mouse position against which people are closest in the data structure. Added an edit menu to the top menu, added an “add person” functionality to this top menu.

Current plans for future development:

* Prototype version is now complete. Write unit tests for back-end logic
* Add in instrumentation. Before doing this a “room” object that stores information about rooms should be created- then measuring who is inside a room’s bounds becomes a simple task. The rooms should have information about the walls that make up the room. Rooms could also have different coloured flooring that represents the type of room they are.
* Add the ability for users to click on people and walls to edit them.

07/01/2019

Features:

* Added the ability to click on a person and edit them.

Bug Fixes:

* Fixed an issue where a person’s location would be rendered differently if the program was in edit mode or animation mode. The person’s location is now rendered the same on both modes.
* Fixed an issue with the person edit menu where deleting a row of the table could delete the table header instead, or an incorrect row from the table
* Fixed a bug where rows would be added to the top of the table instead of the bottom.
* Fixed a bug where leaving a field blank will cause a crash.

10/01/2019

Features:

* Added the ability to click on a wall and edit it, and delete it if necessary.

Project Management:

* Re-ordered backlog and added some new items to backlog.
* Revised the product roadmap.

15/01/2019

Features:

* Added the ability for the user to add an entire room. This work includes implementing the business logic and user interface for this functionality.

Bug Fixes:

* Discovered a bug where the edit wall functionality does not work correctly for walls that share x and y coordinates.
* Discovered a bug where the edit wall dialog box will always default to horizontal, even if the wall is vertical.
* Found and fixed a bug where the room floors wouldn’t render in the correct location in the editing mode.

17/01/2019

Features:

* Added the option for users to right click on a location to add new walls to the building.

Bug Fixes:

* Fixed a bug where editing some walls would lead to a copy of the wall being added
* Fixed the bug where the edit wall combo box will always default to horizontal, even if the wall is vertical. The combo box now defaults to the correct orientation when the wall is edited.
* Discovered a bug where deleting an activity from a person can sometimes cause an array out of bounds error
* Discovered a bug where walls created by right clicking on the edit pane are not created with the correct coordinates.

18/01/2019

Features:

* The Color Picker for creating a new person now defaults to red instead of white. A white dot was very difficult to see if the user forgot to manually set the colour.
* New people and new rooms can be created by right clicking on the edit view.
* Rooms can now be created without walls. This allows certain areas of buildings to be assigned separate uses without needing to put walls in between them.
* Started work on adding doorways into the program. They will allow users to easily insert traversable pathways through the walls without having to delete walls and change them to create a gap.

Bug Fixes:

* Fixed the bug discovered on 17/01/2019 where walls created by right clicking on the edit pane were not created with the correct coordinates; the correct coordinates are now automatically filled in.
* Fixed a bug where editing a person would cause the wrong coordinates to be saved if the user pressed the save button. The correct coordinates are now saved.
* Fixed a bug where the user interface wouldn’t render changes until the animation was resumed and paused again.

19/01/2019

Bug Fixes:

* Fixed an issue where doors could not be added to certain walls.
* Fixed an issue where walls would be different thicknesses. Walls are now of uniform thicknesses when created.
* Discovered a bug where sometimes the pathfinding algorithm will not account for newly added walls in its calculations, sometimes returning a path which crosses through a wall. The bug only occurs at faster simulation speeds, and is much more common when there are less traversable paths to a given location.

25/01/2019

Bug Fixes:

* Fixed the bug discovered in 15/01/2019 where attempting to edit some walls would result in the wrong wall being edited.
* Fixed a bug where doors added would not be displayed correctly in edit mode.
* Fixed a bug where the room would not render properly in animation mode if the animation had not been run at least once yet.

Features:

* Added the ability to edit previously created rooms by clicking on them.

28/01/2019

* Discovered a bug where rooms did not save correctly. Began work on a fix

29/01/2019

Bug Fixes:

* Fixed the bug where rooms did not save correctly, and refactored some of the code used for saving the data to improve code reusability.

01/02/2019

Features:

* Added application logic for instrumenting people’s behaviours
* Added a bar chart for displaying information about how people use the rooms

Bugs:

* Fixed a bug where doors would not be saved to the file.
* Discovered a bug where some walls tied to rooms could not be deleted.

04/02/2019

Features:

* Added the ability to delete a room
* Added cancel buttons to all edit windows for walls, rooms and people.
* Added a feature where the program will automatically align rooms and walls for the user, using a simple grid system.

Bug Fixes:

* Fixed a bug where the user interface would not update correctly after a person was deleted.
* Fixed a bug where trying to create a new person could cause a null pointer exception to be thrown.
* Fixed an issue where the “has walls” checkbox when editing a room would always be checked when opening the dialog box.
* Discovered a bug where if a wall or room was edited, the doors placed on that wall or room would not move with that wall or room.
* Fixed a bug where walls would be duplicated when a file was opened.
* Fixed a bug where walls created as part of a room could not be deleted.
* Fixed a bug where editing a room would cause it to generate new walls, even if the room wasn’t supposed to have walls.

Future plans:

After performing some manual user testing of the building edit functionality, I discovered that adding a grid system to the edit mode would allow users to place their rooms and walls much more easily and make it easier to understand how the auto-alignment feature works.

05/02/2019

Features:

* Added the option for users to delete doors from their custom building.
* Used inheritance to create an Entrance object that is a subclass of a door. An entrance will be a location where people can enter and exit the building.
* Added the ability to give Entrances a name so that the user can select where a person will enter the building from.

07/02/2019

Features:

* Added validation for entrance names. Entrance names should be unique so that the user can identify which entrance they would like a person to enter the building from. Also fixed an issue where the data for entrances would not be updated properly when the user presses the save button.
* People will now appear at the entrance when they are created.

08/02/2019

Refactored code to improve use of inheritance and reduce redundant and duplicated code.

Created a simple plan for the AI of people.

10/02/2019

Features:

* Changed the UI for creating a new person- the user can now specify rooms to visit and what time to visit them, instead of specifying x and y coordinates which was confusing.
* Improved the person object so that they can find their way to a room instead of a specific coordinate.
* When a timetable is saved, the activities will be saved in chronological order.

Bug Fixes:

* Fixed the bug where an out-of-bounds exception would be thrown when trying to delete a row from the person’s timetable.
* Fixed a bug where newly added activities would appear in the wrong order.

11/02/2019

Features:

* People will now follow the updated timetables.
* They will start their timetable at the building’s entrance and leave through an entrance at the end of their day.
* People will now move to a room and stay there for the time stipulated on their timetable, instead of simply moving between coordinates.

12/02/2019

Bug Fixes:

* Fixed the bug where building entrances would not be loaded from a saved file.
* Fixed a bug where building entrance names would not be loaded from a saved file.
* Fixed a bug where the entrances would not be displayed in the correct colour during animation mode.