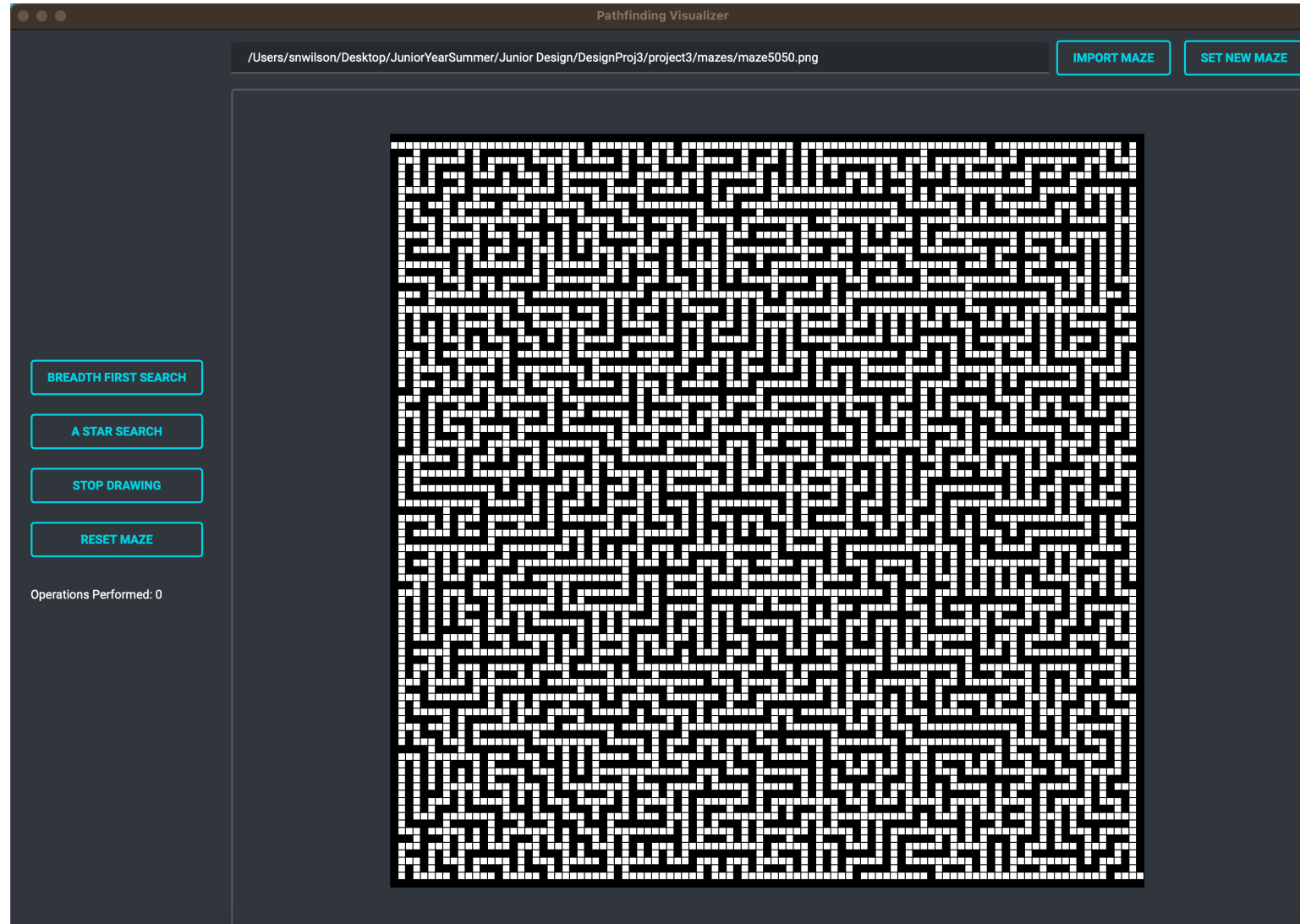# Design Project 3 – Algorithm Visualizer

By Shelby Wilson

Brief Overview

- Pathfinding visualizer
  - Breadth-first search
  - A* Search w/ shortest path shown
- Upload maze pictures using maze generator or preset mazes
  - https://keesiemeijer.github.io/maze-generator/
- Size limit: ~60*60 maze
- Stop drawing
- Reset maze
- Number of operations performed

Brief Overview

- Sorting Visualizer
  - Quick Sort
  - Selection Sort
- Slider for selecting number of elements
- Stop drawing
- Number of comparisons performed
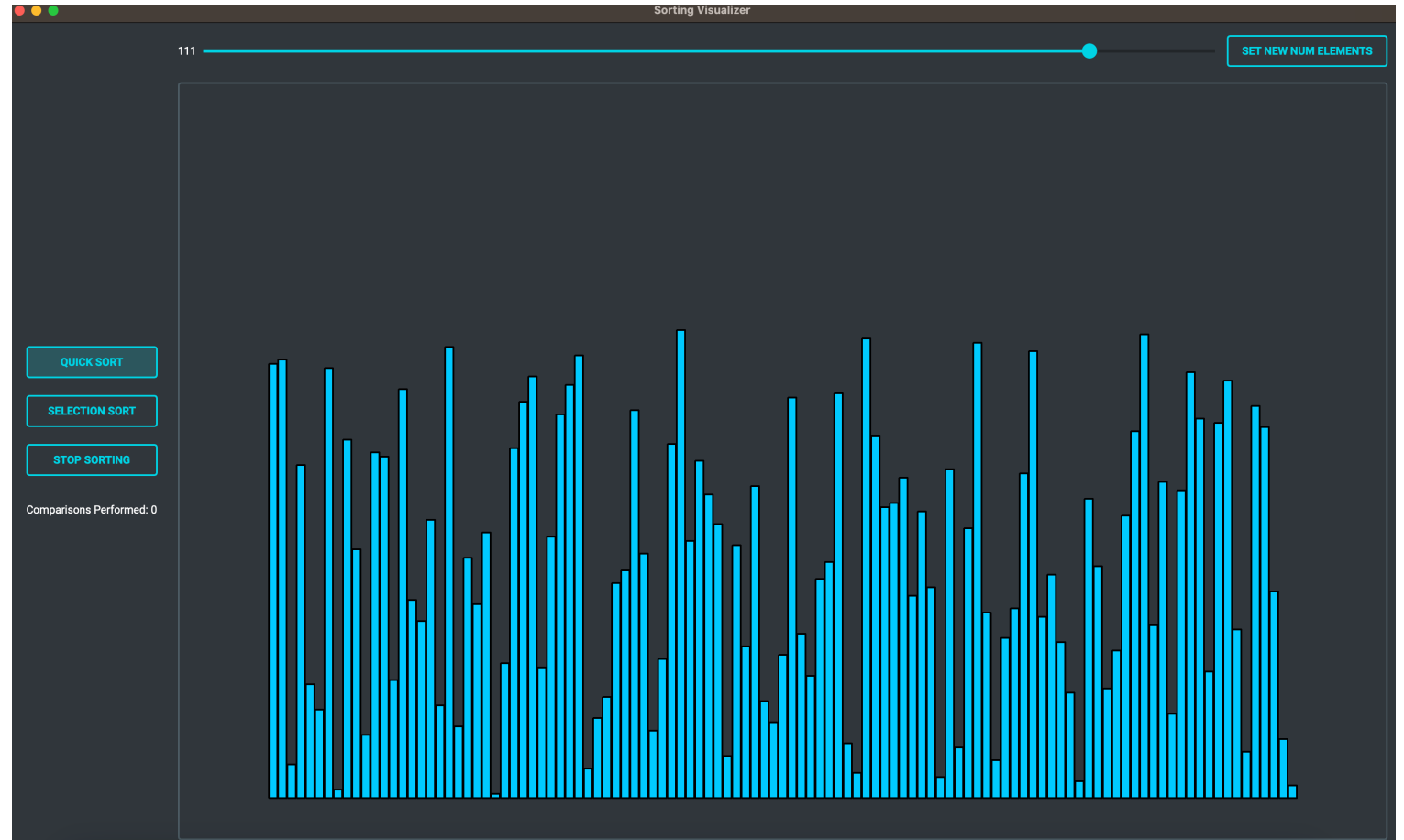
Brief Overview

- Created using PySide6, a Python binding for the C++ Qt6 framework.
  - QRunnable, Signal, Slot, QGraphicsScene, QGraphicsView, QThreadpool
- QtWidgets, QtGui frameworks:
 - QPushButton, QLineEdit, QSlider

Brief Overview

- UI Theme:
  - Dark cyan Qt-Material configuration
  - Link: https://github.com/UN-GCPDS/qt-material

Implementation

- UI Base
  - **LaunchWindow:**
    - pick which app to run
  - **PathfindingVisualizer:**
    - Main Window for pathfinding app
    - Uses GridCell, GridScene, PIL to render images for visualization
  - **SortingVisualizer:**
    - Main Window for sorting app
    - Uses SortingScene to create bars used in visualization

Algorithm Visualier: Launch Window

SHOW PATHFINDING
VISUALIZER

SHOW SORTING
VISUALIZER

EXIT
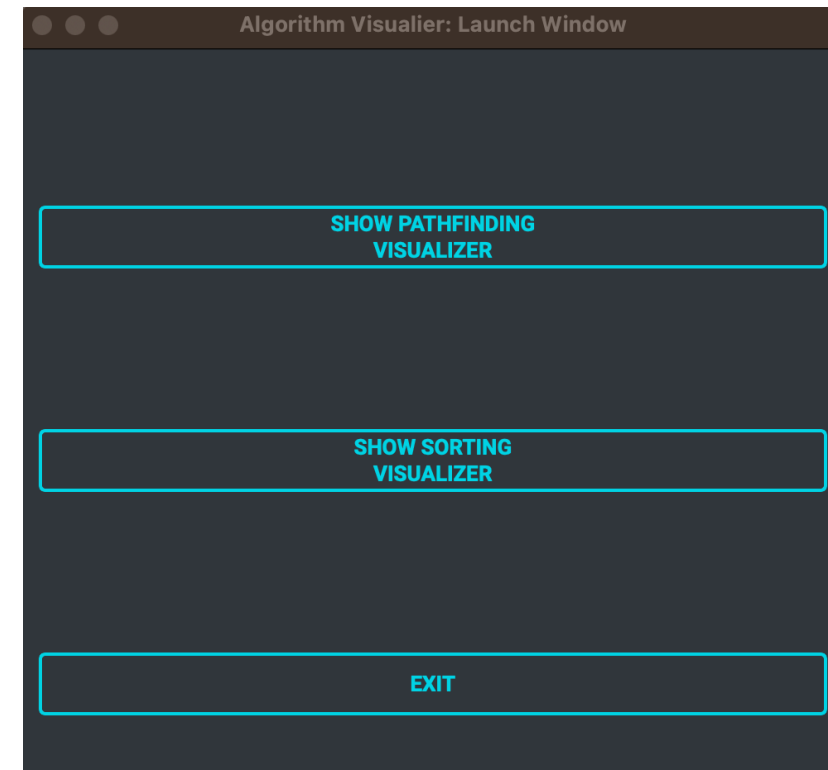

```
You, 6 hours ago | 1 author (You)
> class LaunchWindow(QMainWindow): ...

You, 35 minutes ago | 1 author (You)
> class PathfindingVisualizer(QMainWindow): ...

You, last week | 1 author (You)
> class GridCell(QGraphicsRectItem): ...

You, 2 days ago | 1 author (You)
> class GridScene(QGraphicsScene): ...

You, 35 minutes ago | 1 author (You)
> class SortingVisualizer(QMainWindow): ...

You, 35 minutes ago | 1 author (You)
> class SortingScene(QGraphicsScene): ...
```

Implementation

- Algorithms Backend
  - Holds QRunnable and Signal containers for each algorithm
  - When UI button pressed, respective Runner is added to QThreadPool and started
  - Stop button sets stop flag of current algorithm running
  - Uses python **time** module to create delays during visualization

Node is used inside A runner

Demonstration

Future Work & Improvements

- Add more algorithms

- Add more customization

- Add more algorithm information (time complexity, timer to show how long the algorithm runs, etc.)

- Find way to scale QGraphicsScene and QGraphicsView so maze & sort size is not limited

- Create downloadable executable so I can share the app easily