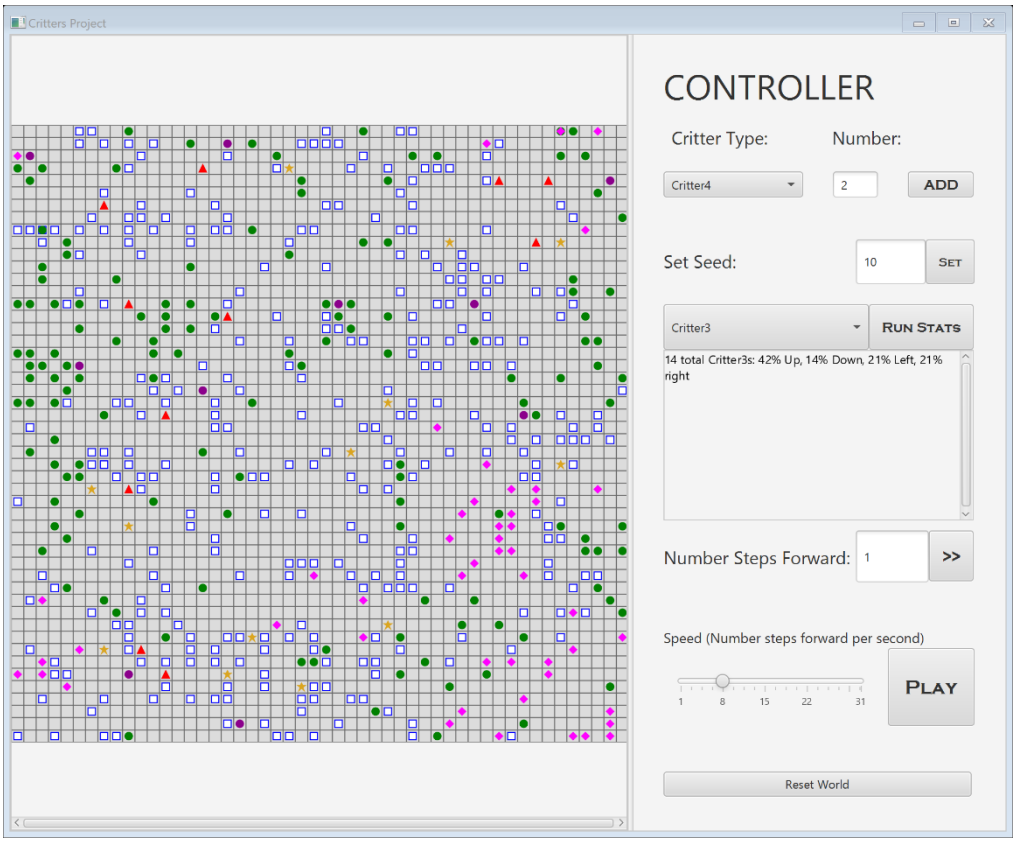
Critters Part 2: GUI Implementation



VIEW

- Displays Critter world
- Size is dictated by Params.java
- **Grid is SCALABLE**
  - There is a min and max box size
  - If canvas space is less than what is needed to fit a grid with min box size, it will turn scrollable
  - If canvas space is more than what is needed to fit a grid with max box size, the graph will center and not scale larger
  - Otherwise canvas scales depending on world size

CONTROLLER

- **ADD CRITTER**
  - Adds <number> of <critter type>
  - List of Critter type depends on files in directory that can be casted to Critter
- **SET SEED**
  - Sets seed for random number generator
- **RUN STATS**
  - Prints out stats for the chosen critter
  - List of Critter type similar to Add Critter list
- **STEP FORWARD**
  - Move world forward <number> of steps
- **ANIMATION**
  - Continually steps through the world at speed indicated by slider in steps indicated by <number>
  - All other controls are disabled during animation
- **RESET**
  - Clears Critter population

Custom Classes

WorldTimeStepTimer

- Extends AnimationTimer
- Contains the code needed for animation
- Is activated by the play button in the controller at the speed indicated by the speed slider in increments of the value input in the number steps forward text box.

CritterWorldView

- Extends Canvas
- Contains all aspects of the grid, including how to make it scalable and all the paint functions
- Passed as a parameter to Critter.displayWorld, which links the view to the model
- Animation/Step implementation
    - Empty grid is redrawn at beginning of each step
    - Critter.displayWorld adds each Critter in the population to the world by calling paintCritter function in CritterWorldView

Controller

- Linked to the view through the FXMLLoader
    - View elements are accessed through @FXML directives
    - Functions are linked through FXML file
- Contains the code to handle all events

AppScene.fxml

- GUI was implemented through SceneBuilder, which generates an FXML file that dictates the layout of the GUI. The FXMLLoader is then called in the Main to link the GUI with the Controller code.