Traffic Simulator

Violetta Avkhukova
Yakubu Aliyu Doma
Rawan Mohammed Alrahili
Felix Santiago Anda Basabe
Jia Liu

March 31, 2016

7CCSMGPR

Aims

- **Must**
  - Cellular Automaton
  - Vehicle Entry and Exit
  - Free Movement and Turning of vehicles
  - Default Map
  - Display Simple Animation of Vehicle Movement
- **Should**
  - Create, Save/Load Maps
  - Traffic Policies - lane disabling, light durations
  - Priorities for Emergency Services
- **Could**
  - Statistics - time spent at traffic e.t.c
  - Curved Roads
  - External Map Sources, e.g OpenStreetMap

Design

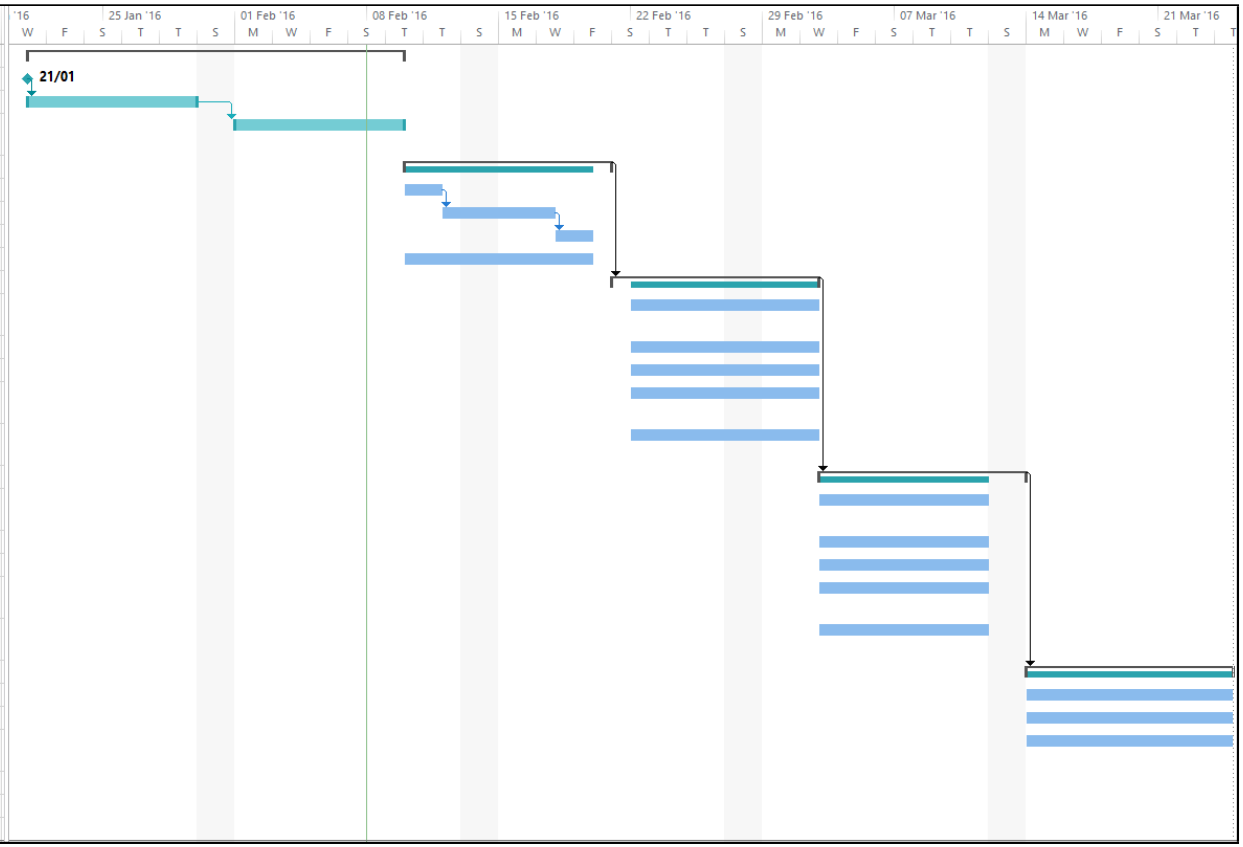
- **Cellular Automata**
  - Simple and efficient
  - Realistic simulation
- **MVC**
- **Design Patterns**

Design cont....

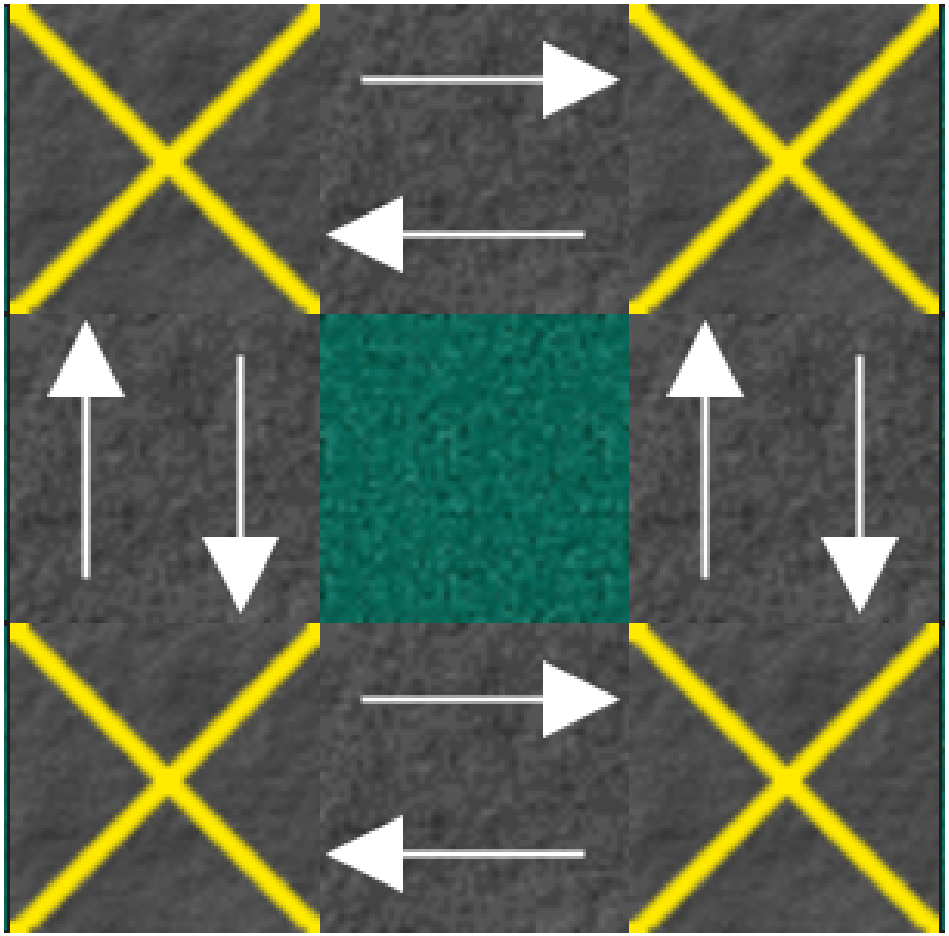

User
Start simulation
Load pre-made maps
Create new maps
Set ticker duration
Traffic policies:
Enable & disable traffic light
Set traffic light duration
Disable lane

Iterations

Task Mode	Task Name	Duration	Start	Finish	Predecessor
	Iteration 0	**14 days**	**Thu 21/01/16**	**Tue 09/02/16**	
	Team Formation	0 days	Thu 21/01/16	Thu 21/01/16	
	Research & Brainstorming	7 days	Thu 21/01/16	Fri 29/01/16	2
	Refined Development Plan & Documentation	7 days	Mon 01/02/16	Tue 09/02/16	3
	Iteration 1	**9 days**	**Wed 10/02/16**	**Sat 20/02/16**	
	One road, one car	2 days	Wed 10/02/16	Thu 11/02/16	
	One road, multiple cars	4 days	Fri 12/02/16	Wed 17/02/16	6
	Simple GUI	2 days	Thu 18/02/16	Fri 19/02/16	7
	Research	8 days	Wed 10/02/16	Fri 19/02/16	
	Iteration 2	**9 days**	**Sun 21/02/16**	**Wed 02/03/16**	**5**
	Roads, using intersections	8 days	Mon 22/02/16	Wed 02/03/16	
	Cars do not turn yet	8 days	Mon 22/02/16	Wed 02/03/16	
	Traffic lights should work	8 days	Mon 22/02/16	Wed 02/03/16	
	GUI can display the bigger map	8 days	Mon 22/02/16	Wed 02/03/16	
	Create foundations for map making components	8 days	Mon 22/02/16	Wed 02/03/16	
	Iteration 3	**8 days**	**Thu 03/03/16**	**Sun 13/03/16**	**10**
	Cars can now turn, randomly	7 days	Thu 03/03/16	Fri 11/03/16	
	GUI shows cars turning	7 days	Thu 03/03/16	Fri 11/03/16	
	Traffic light management	7 days	Thu 03/03/16	Fri 11/03/16	
	Rules of the road management	7 days	Thu 03/03/16	Fri 11/03/16	
	Make a simple map with some components	7 days	Thu 03/03/16	Fri 11/03/16	
	Iteration 4	**9 days**	**Mon 14/03/16**	**Thu 24/03/16**	**16**
	Cars more clever	9 days	Mon 14/03/16	Thu 24/03/16	
	Sliders	9 days	Mon 14/03/16	Thu 24/03/16	
	Load maps and have cars run on them	9 days	Mon 14/03/16	Thu 24/03/16	



Map Structure



Testing

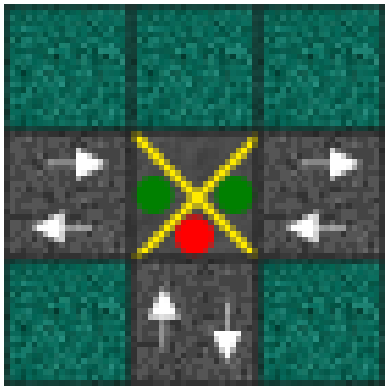
- Integration Testing
- JUNIT

```
1 test passed - 49ms

R R R R R R - -
- - - - - - - -
- - - - - - - I
- - - I - - - -
- - - - - - - -

R R R R R R - -
- - - - - - - -
- - - - - - - I
R - - I - - - -
R - - - - - - -

Process finished with exit code 0
```

Ticker

- Keeps track of time in the system
- Tick Interval
- **Two implementations:**
  - First implementation:
    - Java Timer
    - Ran in its own thread
    - Issues with JavaFX
  - Second implementation:
    - Relies on RxJava, RxJavaFX
    - Ticker = Observable
    - Classes become Subscribers, perform operation onNext(Long l)
- Tick interval set before simulation start

Traffic Lights



- 2 states: red, green
- Duration
- Subscribe to ticker
- onNext(): change colour when needed

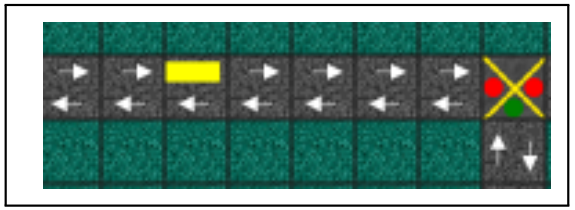
Vehicle Class

- Vehicle is an abstract class that all vehicles implement.
- Vehicles have a lot of attributes that are either global or specific.
- There are two types of vehicles:
  - Cars.
  - Ambulances.

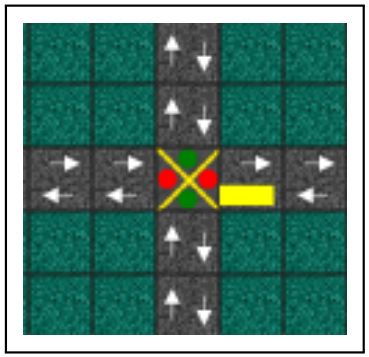
Vehicle Movement and Ticker Interaction

Vehicle movement can be divided into two main categories:

1. Moving in a lane:
   - Each lane is an array list of vehicles.
   - Vehicles are items in lanes.



2. Turning to a new lane
   - Reading traffic light. `void readTrafficLight()`
   - Choosing lane to move to. `Lane chooseLane()`
   - Ability to turn. boolean `vehicleTurnFirst (ArrayList<Vehicle> vehicles)`



Vehicle turns: `int vehicleTurn(Lane l)`

Log

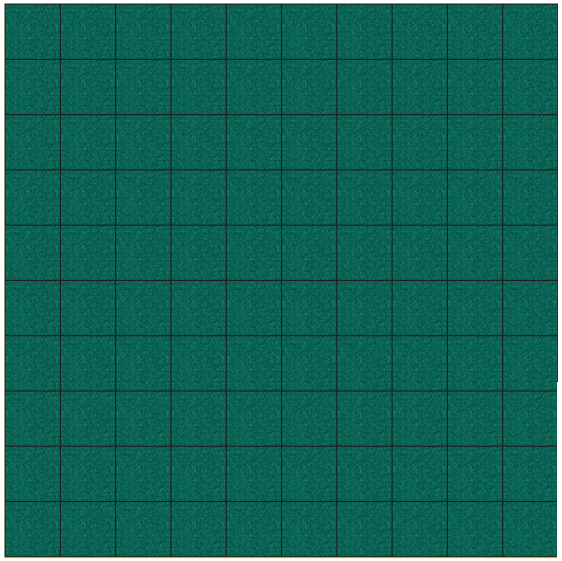
- Format Log Year-Month-Day-Hour- Minute-Second
- Suscribes to ticker
- Records attributes of each Subscriber
- Useful for debugging processes, audits, and statistics.

```
------TRAFFIC LIGHT------
ID: 0
DURATION: 3000
STATE: RED
------TRAFFIC LIGHT------
ID: 24
DURATION: 3000
STATE: GREEN
---------VEHICLE---------
ID: 1
CURRENT LANE ID: 48
CURRENT COORDINATES: 20,18
PREVIOUS LANE ID: 48
PREVIOUS LANE COORDINATES: 20,18
CURRENT CELL: 2
BEHAVIOUR: CAUTIOUS
PRIORITY: 1
STATE: 1
---------VEHICLE---------
ID: 2
CURRENT LANE ID: 40
CURRENT COORDINATES: 8,14
PREVIOUS LANE ID: 40
PREVIOUS LANE COORDINATES: 8,14
CURRENT CELL: 3
BEHAVIOUR: AVERAGE
PRIORITY: 1
STATE: 0
```

GUI

Map GUI

- Grid of rows and columns
- Each cell is a StackPane
- Gimp for drawings
- Dynamic size: resize factor



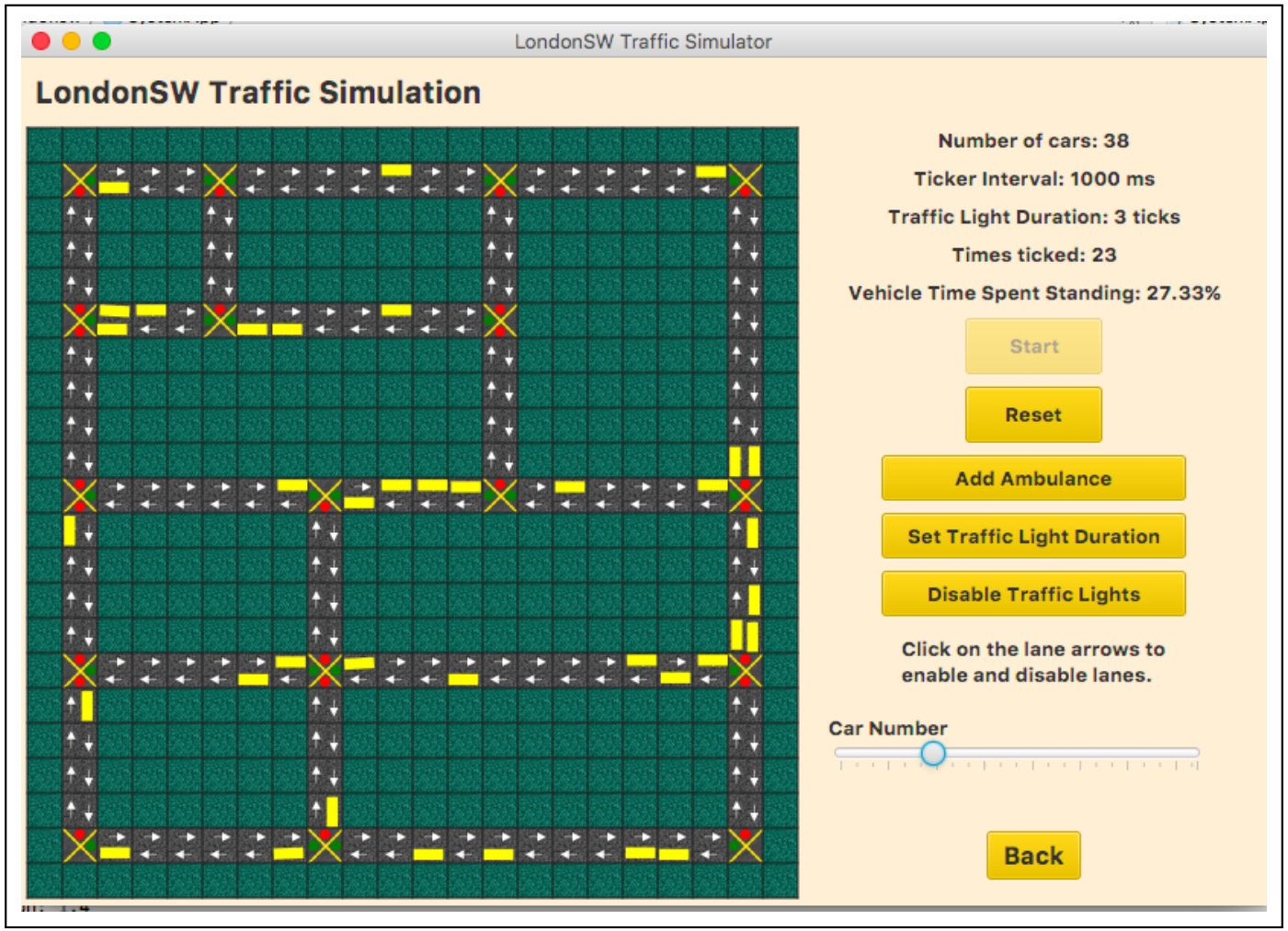
GUI

Decorators

- **MapGridGUIDecorator**
  - Manages the drawing of the entire map
- **RoadGUIDecorator**
  - Draws a Road and programatically the lanes
- **IntersectionDecorator**
  - Extends the Intersection Functionality
- **TrafficLightDecorator**
  - Circles are drawn programatically
- **VehicleGUIDecorator**
  - State [0 to 3]

GUI

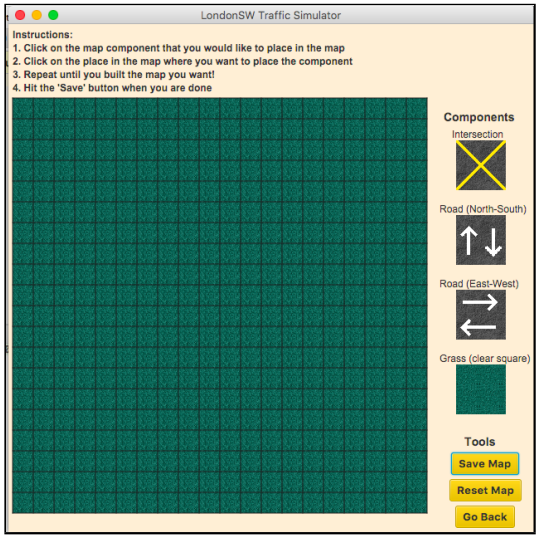
Map Simulation



Map Maker Mode

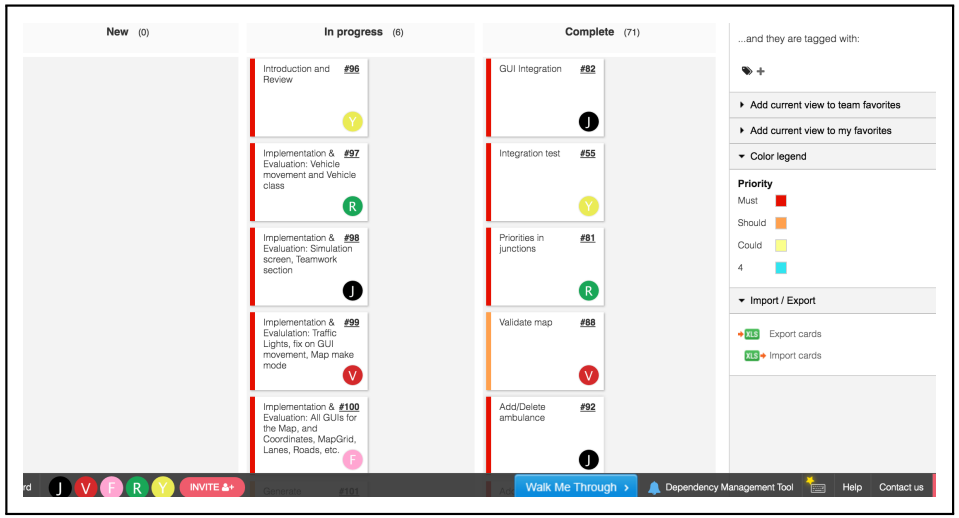
Users create their own maps

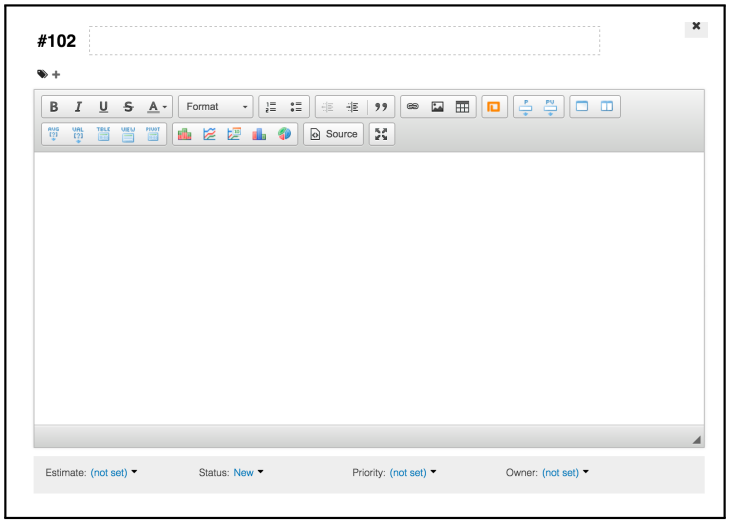
- Ask for width and height
- 4 types of map component:
  - intersection
  - road (north-south)
  - road (east-west)
  - grass
- Click on component,
  click in map any number of times
- Saving the map



Teamwork

- Group meeting
- Agile development with Mingle
  - Mingle's Planner feature





Simulation Screen

- Scene Builder
- Pure JavaFX
  - Architecturec: BorderPane
  - Simulaton monitor: Labels
  - Simulation Control: Buttons, slider, dialogs

General evaluation

- Things that went well:
  - project structure
  - teamwork
  - achieved all our goals
- Things that did not go well and what we did:
  - Ticker, we re-implemented
  - Vehicle movement in maps, we fixed it
  - Loaded maps and traffic lights didn't work, fixed it
- Possiblilities for future:
  - More types of vehicles, roads
  - Curved roads
  - More statistics

Simulation

DEMONSTRATION

Simulation

QUESTIONS AND ANSWERS