MRKJ Team

CS 347 - Professor Peyrovian
Authors: Kailie J, Martyna Z, Jack N, Renny V

Table of Contents

1. Goals

2. Uses & Values

3. Code Demo

4. Software Development Process

5. Challenges

6. Successes

7. Improvements

Goals

- Learn how the Software Development Process works
- Develop knowledge on UML diagram construction
- Completion of a self driving car project using IoT
    - Safety
        - Collision avoidance
    - Reliability
        - Code stability
    - Quick implementation of improvements

Uses and Value

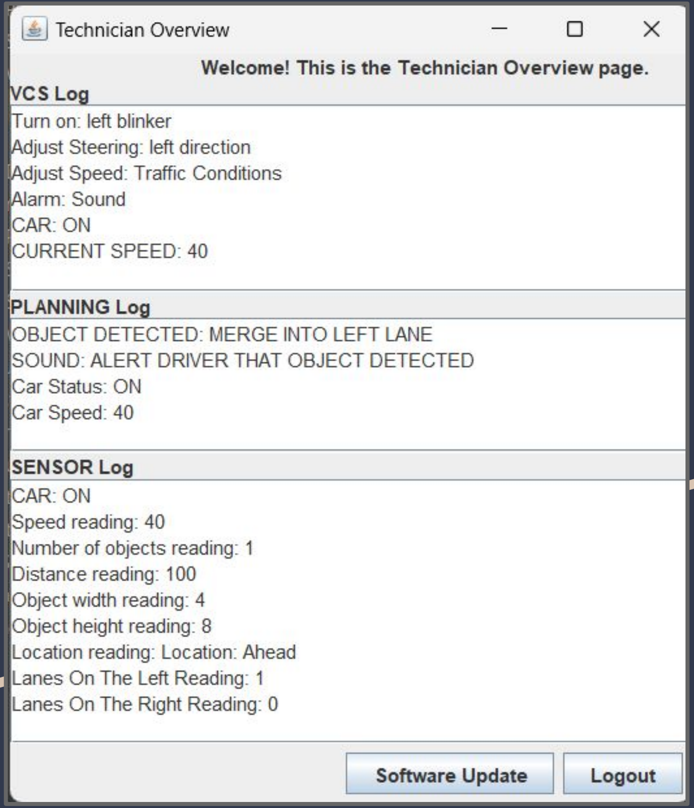
USES

- Virtual car environment
- Testing data outside a physical car
- Features can be used in a multitude of ways
  - Technician UI
  - Prioritization algorithm
- IoT edge incorporation for data capturing

VALUES

- Foundation for future autonomous vehicle software
- Can be built upon
- Ideas on what type of data the car captures

Code Demo



Welcome! This is the Technician Overview page.

VCS Log

```
Turn on: left blinker
Adjust Steering: left direction
Adjust Speed: Traffic Conditions
Alarm: Sound
CAR: ON
CURRENT SPEED: 40
```

PLANNING Log

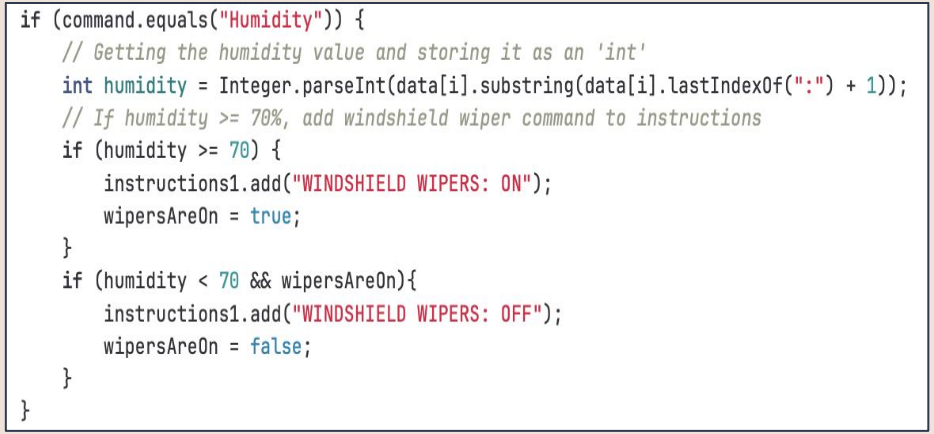
```
OBJECT DETECTED: MERGE INTO LEFT LANE
SOUND: ALERT DRIVER THAT OBJECT DETECTED
Car Status: ON
Car Speed: 40
```

SENSOR Log

```
CAR: ON
Speed reading: 40
Number of objects reading: 1
Distance reading: 100
Object width reading: 4
Object height reading: 8
Location reading: Location: Ahead
Lanes On The Left Reading: 1
Lanes On The Right Reading: 0
```

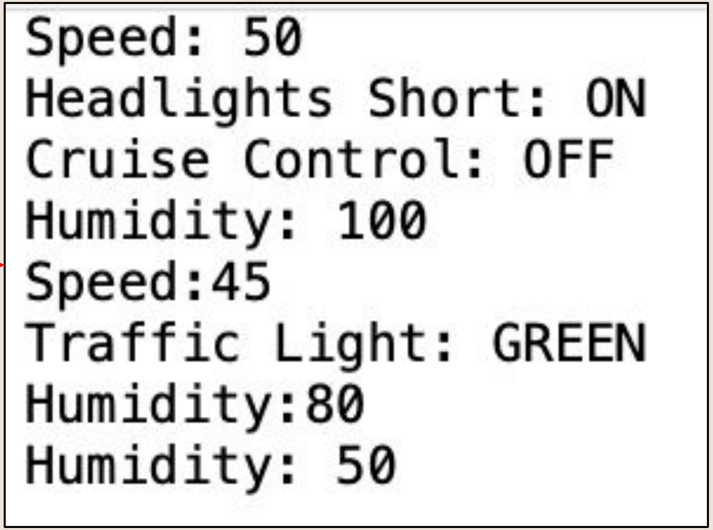
- Sensor Test
  - Object Avoidance?
- Driver Input Test
  - Headlights?
- Technician Interface Test
  - Software Update
  - Login
  - Log Analysis

Sample Code



```
if (command.equals("Humidity")) {
    // Getting the humidity value and storing it as an 'int'
    int humidity = Integer.parseInt(data[i].substring(data[i].lastIndexOf(":") + 1));
    // If humidity >= 70%, add windshield wiper command to instructions
    if (humidity >= 70) {
        instructions1.add("WINDSHIELD WIPERS: ON");
        wipersAreOn = true;
    }
    if (humidity < 70 && wipersAreOn){
        instructions1.add("WINDSHIELD WIPERS: OFF");
        wipersAreOn = false;
    }
}
```

Sample Test Case



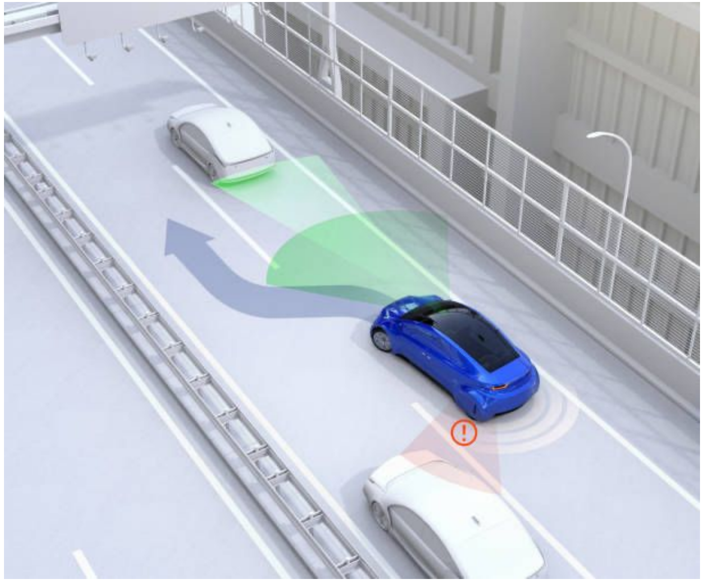
```
Speed: 50
Headlights Short: ON
Cruise Control: OFF
Humidity: 100
Speed:45
Traffic Light: GREEN
Humidity:80
Humidity: 50
```

WINDSHIELD WIPERS: OFF

WINDSHIELD WIPERS: ON

Guaranteed Safety

- Various testing
- System updates
- Technician log-in & support
- Cruise Control
- Assisted parking and reversing
- Feature automation
  - Windshield wipers
  - Braking
  - Object avoidance



Software Development Process

REQUIREMENT UNDERSTANDING→

DOCUMENTATION→

ANALYSIS→

TESTING

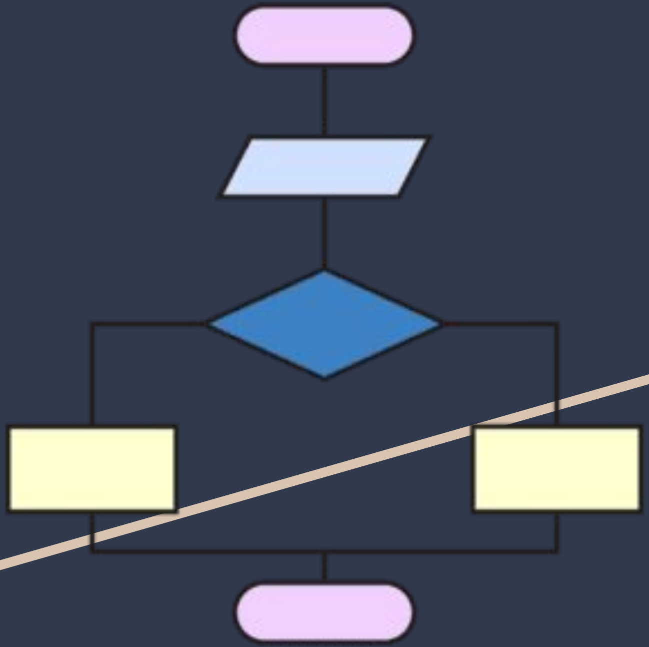
Iterative Waterfall Process:

- Allows for consistent software development and for an easy editing process.
- Make improvements easily
- Communicating regularly with our customers to gain valuable feedback
- Easy to add in more features

Our Implementation:

- Updates conducted throughout the process
  - Removed a requirement, etc
- Built in safety features
  - Priority Algorithm, alerts, etc

Design



Object Oriented

- Modular Design
- Made it easy to delegate tasks
- Multiple uses for subsections of code
- Easy to coordinate

Challenges

- Managing data types
  - When using multiple java classes, it is important to record what each data type result will be
- Update Synchronization
  - Updates only synchronize when updates are pushed to Git
- Scheduling Issues
  - Very different schedules so hard to coordinate times to meet

Successes

- Ease of development of code
  - The layout from the software development process made it easy to program
- Coordination between team members
  - Everybody did their part well
- Communication
  - Took feedback and implemented it
- Submission of deliverables on time (all semester)

Improvements

- Cloud Services
    - Cloud integration with Technician UI for more protected data
- Be very Specific with Data Types
    - Need this when using Object Oriented Programming
- Setting up a specific time to meet each week earlier in the semester
- Further implementation of individual features
    - Ex. Software Update

Thanks for listening!

Questions?