# Final Report

Group 5:

Arman Kocharyan – 100888381

Katherine Nelson – 101012786

Andrew Dodge - 100938015

Moh Gahelrasoul – 101007118

Roman Kishinevsky – 101009733

Dr. Gregory Franks

SYSC 3303A – Real-Time Concurrent Systems

Wednesday, April 10th, 2019

# **Table of Contents**

[Title Page 1](#_Toc5559279)

[Table of Contents 2](#_Toc5559280)

[Breakdown of Responsibilities 3](#_Toc5559281)

Iteration 1…………………..……………………………………………………………………………#

Iteration 2…………………..………………………………………………………………..…….……#

Iteration 3…………………..…………………………………………………………………………...#

Iteration 4…………………..……………………………………………………………………………#

Iteration 5………..………………………………………………………………………………………#

[Diagrams 5](#_Toc5559282)

UML Class Diagram – Floor……………………………………………………………………………#

UML Class Diagram – Scheduler………………………………………………………………….……#

UML Class Diagram – Elevator………………………………………………………………………...#

State Machine Diagram – Scheduler……………………………………………………………………#

Sequence Diagram………………………………………………………………………………………#

Timing Diagram – Scheduler……………………………………………………………………………#

[Instructions 6](#_Toc5559283)

Set Up……………………..……………………………………………………………………………#

Testing……………………..………………………………………………………………..…….……#

[Measurement Results 7](#_Toc5559284)

[System Analysis 8](#_Toc5559285)

[Reflection 9](#_Toc5559286)

# **Breakdown of Responsibilities**

*Iteration 1*

|  |  |
| --- | --- |
| **Group Member** | **Responsibilities** |
| Arman Kocharyan |  |
| Katie Nelson |  |
| Andrew Dodge | * UML class diagram |
| Mohamed Gahelrasoul |  |
| Roman Kishinevsky |  |

*Iteration 2*

|  |  |
| --- | --- |
| **Group Member** | **Responsibilities** |
| Arman Kocharyan | * Test cases * Timing of elevator carts * Created cucumber tests and the logger * Elevator message (the default message class for all UDP messages) |
| Katie Nelson | * Floor subsystem arrival * Floor controller * Sensors * Implemented pickUpPerson, rideToFloor, and the event listener which opens a socket on a given port and listens until message is received |
| Andrew Dodge | * UML class diagram * Implemented startListen which starts a new thread/daemon that blocks and waits call |
| Mohamed Gahelrasoul | * Documentation * Timing diagrams * Worked on resetting the buttons, and elevator arrival |
| Roman Kishinevsky | * Requesting elevator up and down, with the up down buttons * Small interactions with the elevator |

*Iteration 3*

|  |  |
| --- | --- |
| **Group Member** | **Responsibilities** |
| Arman Kocharyan | * Worked on exceptions * Worked on testing |
| Katie Nelson | * Worked on exceptions * Worked on testing |
| Andrew Dodge | * Worked on timing * Worked on diagrams |
| Mohamed Gahelrasoul | * Worked on timing * Worked on diagrams |
| Roman Kishinevsky | * Worked on timing * Worked on diagrams |

*Iteration 4*

|  |  |
| --- | --- |
| **Group Member** | **Responsibilities** |
| Arman Kocharyan | * Worked on exceptions * Worked on testing |
| Katie Nelson | * Worked on exceptions * Worked on testing |
| Andrew Dodge | * Worked on timing * Worked on diagrams |
| Mohamed Gahelrasoul | * Worked on timing * Worked on diagrams |
| Roman Kishinevsky | * Worked on timing * Worked on diagrams |

*Iteration 5*

|  |  |
| --- | --- |
| **Group Member** | **Responsibilities** |
| Arman Kocharyan | * GUI |
| Katie Nelson | * GUI * Refactoring code to accept multiple people |
| Andrew Dodge | * GUI * Created, wrote, and formatted the report |
| Mohamed Gahelrasoul | * GUI |
| Roman Kishinevsky | * GUI |

# **Diagrams**

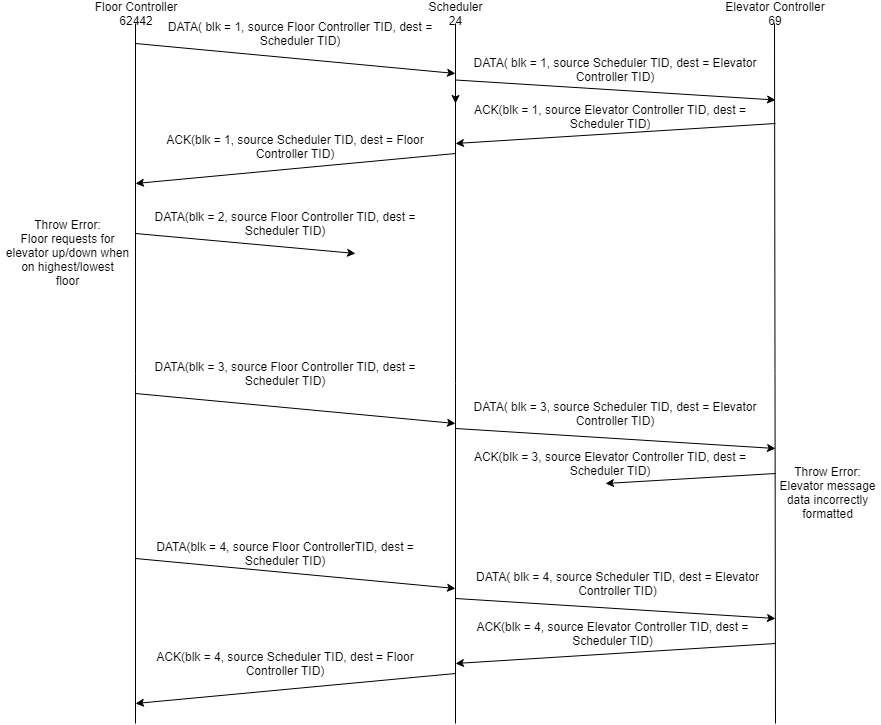
*UML Class Diagram – Floor*

*UML Class Diagram – Scheduler*

*UML Class Diagram – Elevator*

*State Machine Diagram – Scheduler*

*Sequence Diagram*

*Timing Diagram – Scheduler*

# **Instructions**

*Set Up*

To run the code:

* Open eclipse ide and import project as a maven project
* Open the subsystems directory

1) Right click elevator and run as java program

2) Right click scheduler and click run as a java program

3) Right click floor and click run as a java program

To run on multiple computers:

* Open eclipse ide and import project as a maven project on both machines
* Open the subsystems directory

1. In the core package run the IPGetter.java file on both machines
2. In the scheduler package open the Scheduler.java file on machine 1, set the public static final String ADDRESS = ""; to the IP address of machine 2, add it inside the ""
3. open the Floor.java file from floor package and Elevator.java file from the elevator package, and add machine 1's IP address to the public static final String ADDRESS = ""; variable in those files
4. Run Scheduler.java on machine 1
5. Run ElevatorController.java followed by FloorController.java on machine 2

*Testing*

* Open eclipse ide and import project as a maven project
* Open the subsystems directory

1. Run CucmonberRunner.java as a junit test

# **Measurement Results**

# **System Analysis**

# **Reflection**