**CI253 Assignment 2 Technical Documentation and Testing**

Table of Contents

[UML Diagrams 1](#_Toc512420654)

[Test 1 (Bug) 1](#_Toc512420655)

[Test 2 (Unexpected Behaviour) 2](#_Toc512420656)

[Test 3 (Unexpected Input) 2](#_Toc512420657)

[Test 4 (Unexpected Data Type) 3](#_Toc512420658)

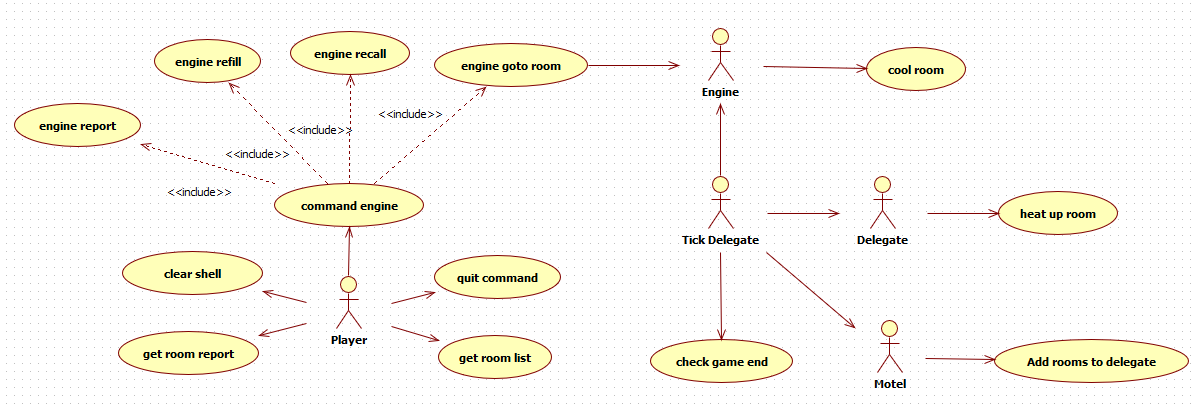
[Test 5 (Unexpected Data Format) 3](#_Toc512420659)

[Test 6 (Unexpected Result) 4](#_Toc512420660)

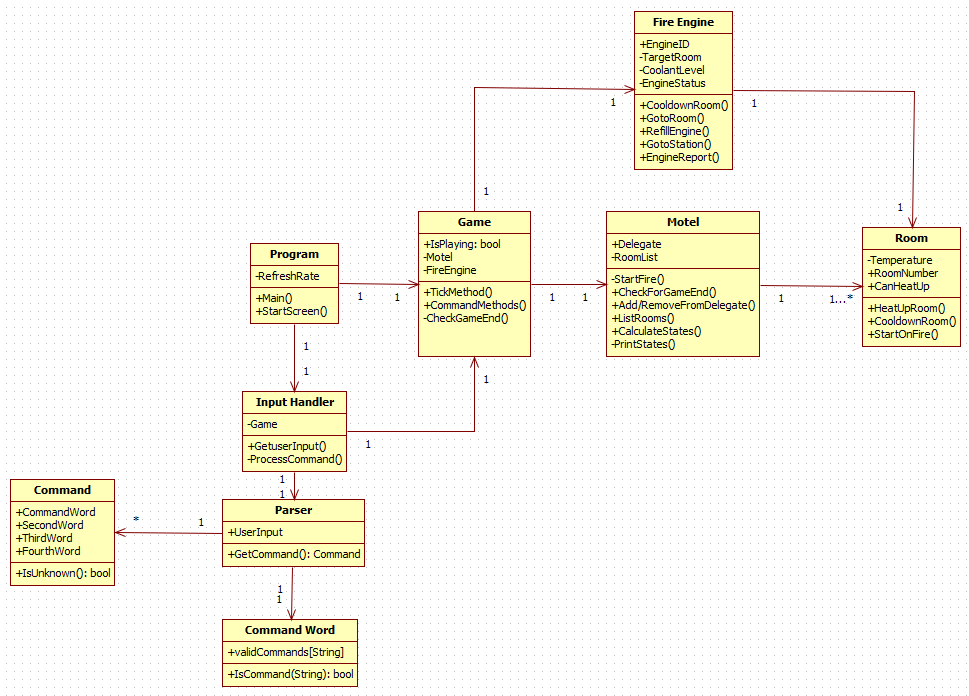
[Test 7 (Unexpected Method Call) 4](#_Toc512420661)

[Work Log 5](#_Toc512420662)

# UML Diagrams

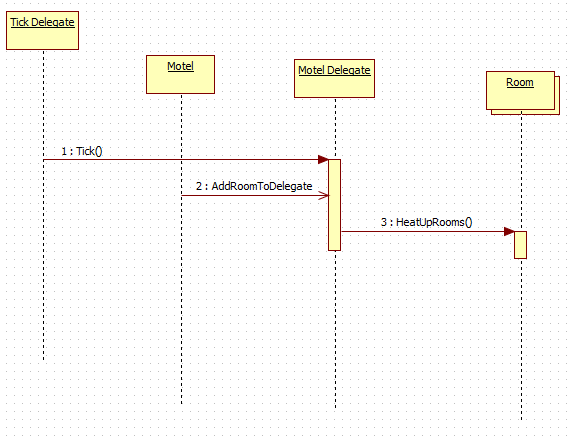
**Use Case Diagram**

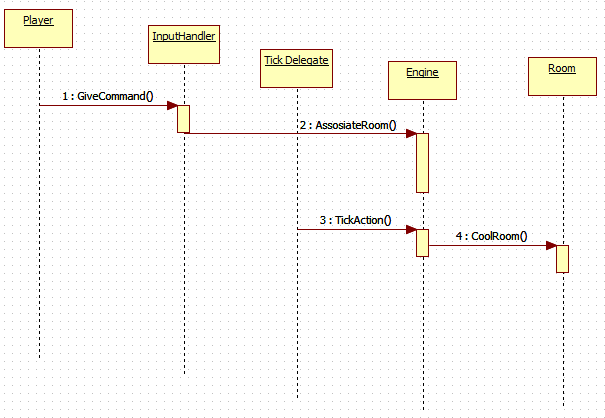
|  |  |  |  |
| --- | --- | --- | --- |
| **Use Case** | **Initiating Actor** | **Input** | **Result** |
| Command Engine | Player | report command | Engine reports its status, remaining coolant and if its cooling a room: the room’s status and temperature. |
| refill command | Engine will refill coolant to max if it is stationed. Will do nothing if it is not. |
| recall command | Set engine’s status as stationed. |
| go to room command | Engine status switched to on-call, the room given is associated with the engine. |
| Clear Shell | Player | clear command | Shell is cleared of previous written lines. |
| Get Room Report | Player | room report command | Lists the amount of rooms in the different states derived from their temperature. |
| Get Room List | Player | room list command | Rooms are listed with their respective numbers, temperatures and states. |
| Quit | Player | quit command | Game ends and end screen is displayed with the room report. |
| Cool Room | Tick Delegate | coolant amount | Room cools by an amount relative the amount of coolant released. Coolant does not go beyond 0. |
| Add Rooms To Delegate | Motel | temperature of neighbouring rooms | If the temperature of a room exceeds 149 degrees, the adjacent rooms heat up methods are added to the delegate. |
| Heat Up Room | Tick Delegate | heat room methods | All rooms added to the delegate are heated by the amount stated per tick. |
| Check Game End | Tick Delegate | Room’s can heat up bools, delegate members | If the delegate is empty and none of the rooms can heat up the game ends. |

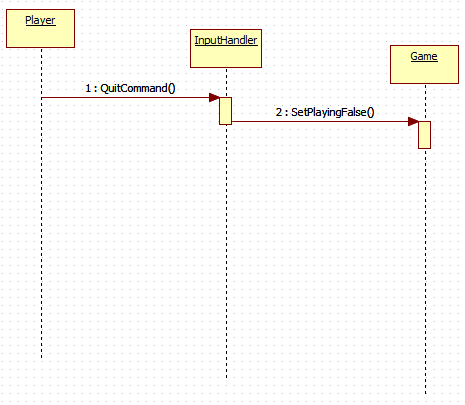
**Class Diagram**

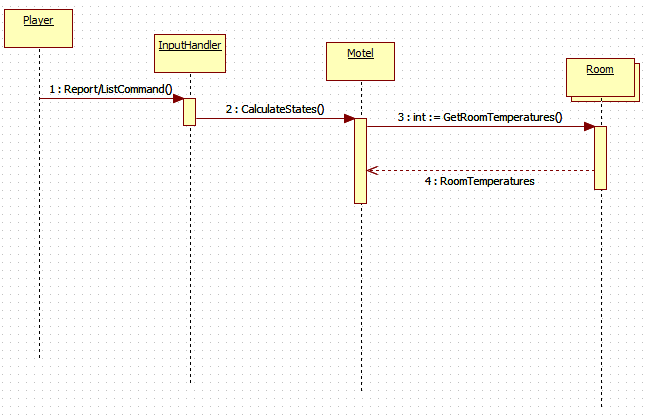
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Class** | **Attributes** |  | **Opperations** |  | **Purpose** |
| Command | +CommandWord | First word routing the command (room, engine etc). | +IsUnknown() | Returns a Boolean to validate the command word is entered. | This Class is used to wrap the command given by the player in an object. |
| +SecondWord | Second word of command (list, goto etc). |
| + ThirdWord | Third word of command (ie room #). |
| +FourthWord | Fourth word of command. |
| Command Words | +validCommands[String] | List of valid command words. | +IsCommand(String):bool | Checks player entered command word. | User to check the initial command entered by the player. |
| Fire Engine | +EngineID | Used to identify the fire engine. | +CooldownRoom(): | Reduce the temperature of a room. | The fire engine is the object used to prevent the rooms from heating. |
| -TargetRoom | Room to cool down. | +GotoRoom(room) | Change status to on-call and room reference. |
| -CoolantLevel | Amount of coolant remaining. | +RefillEngine() | Sets coolant to max if status is stationed. |
| -EngineStatus | e.g on-call, stationed. Can be used to drive logic via states. | +GotoStation() | Set the engines status to stationed. |
| +EngineReport() | Lists the engine’s coolant levels, status and the target room’s: status and temperature. |
| Game | +IsPlaying: bool | Boolean resolved by motel that notates if the game is playing. | +TickMethod | Tick method calls motel room delegate operations and the fire engine cooling. | Contains the game objects motel and fire engine, as well as the main tick body. |
| -Motel | Reference to motel | +CommandMethods | Routes command messages to objects. |
| -FireEngine | Reference to FireEngine | -CheckGameEnd | Calls motel to check for game end conditions. |
| Input Handler | -GameRef | Game passes a reference of self. | +GetUserInput() | While game is playing wait for user input. | Used to interface command objects with the game. |
| -Parser | Generates Command Objects. | -ProcessCommand() | Change command into program action. |
| Motel | +Delegate | Delegate used to call heat-up method on rooms. | -StartFire() | Initial Room to start on fire. | This class contains the list of rooms and can perform various actions on those rooms. |
| +CheckForGameEnd() | Check for game end conditions. |
| +Add/RemoveFromDelegate() | First establish if room needs to be added, then call an add to the delegate. |
| -RoomList | Contains all the rooms in the game. | +ListRooms() | List all the rooms with their temperatures and status. |
| +CalculateStates | Calculate the amount of rooms in each state. |
| -PrintStates | Print a list of states with the amount of rooms in those states. |
| Parser | -UserInput | String entered by the user. | GetCommand():Command | Creates a new command object from the player’s input. | Creates command object for use with input. |
| Program | -RefreshRate | The rate at which the rooms will tick | Main() | Main | Entry point for the program. |
| StartScreen() | Show player instructions. |
| StartTimer() | Start the timer delegate. |
| Room | -Temperature | Temperature of the room | +HeatUpRoom() | Heats up the room, called by delegate. | Has a temperature and can heat up and cooldown. |
| +RoomNumber | Room number as resolved by list creation. | +CooldownRoom(Coolant) | Room is cool down by the amount of. |
| +CanHeatUp | Can the room heat up | +StartOnFire() | Chooses a room at random to start on fire. |

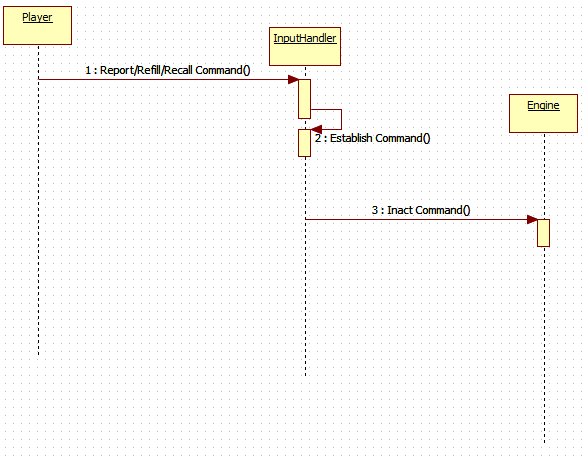
**Sequence Diagrams**

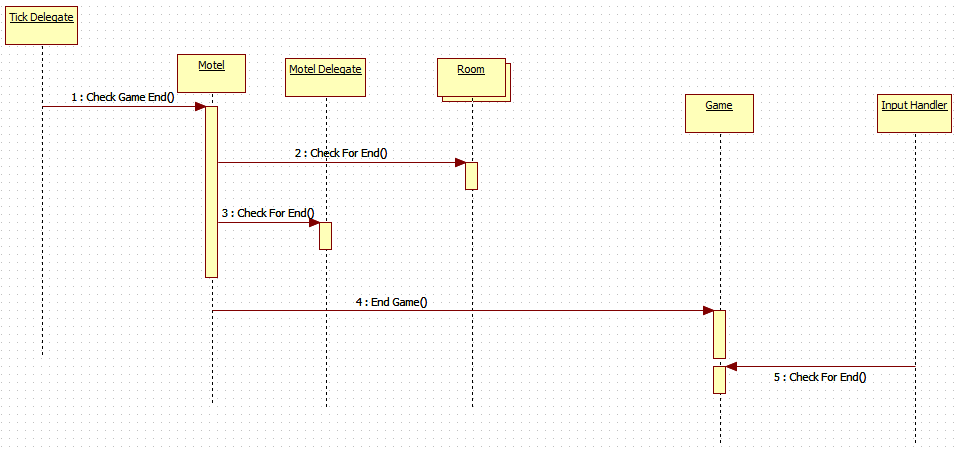
HeatUpRoom

CoolRoom

Quit command

RoomReport/List

Engine report/refill/recall

CheckGameEnd

# Test 1 (Unexpected Behaviour)

**Error/Issue**

**Test Strategy**

**Test Data**

|  |  |  |
| --- | --- | --- |
| Error/Issue | Test | Expected Outcome |
|  |  |  |

**Test Execution**

**Test Results**

**Interpretation of Results**

|  |  |  |
| --- | --- | --- |
| Expected Outcome | Actual Outcome | Discussion |
|  |  |  |

# Test 2 (Unexpected Input)

**Error/Issue**

**Test Strategy**

**Test Data**

|  |  |  |
| --- | --- | --- |
| Error/Issue | Test | Expected Outcome |
|  |  |  |

**Test Execution**

**Test Results**

**Interpretation of Results**

|  |  |  |
| --- | --- | --- |
| Expected Outcome | Actual Outcome | Discussion |
|  |  |  |

# Test 3 (Unexpected Data Type)

**Error/Issue**

**Test Strategy**

**Test Data**

|  |  |  |
| --- | --- | --- |
| Error/Issue | Test | Expected Outcome |
|  |  |  |

**Test Execution**

**Test Results**

**Interpretation of Results**

|  |  |  |
| --- | --- | --- |
| Expected Outcome | Actual Outcome | Discussion |
|  |  |  |

# Test 4 (Unexpected Data Format)

**Error/Issue**

**Test Strategy**

**Test Data**

|  |  |  |
| --- | --- | --- |
| Error/Issue | Test | Expected Outcome |
|  |  |  |

**Test Execution**

**Test Results**

**Interpretation of Results**

|  |  |  |
| --- | --- | --- |
| Expected Outcome | Actual Outcome | Discussion |
|  |  |  |

# Test 5 (Unexpected Result)

**Error/Issue**

**Test Strategy**

**Test Data**

|  |  |  |
| --- | --- | --- |
| Error/Issue | Test | Expected Outcome |
|  |  |  |

**Test Execution**

**Test Results**

**Interpretation of Results**

|  |  |  |
| --- | --- | --- |
| Expected Outcome | Actual Outcome | Discussion |
|  |  |  |

# Test 6 (Unexpected Method Call)

**Error/Issue**

**Test Strategy**

**Test Data**

|  |  |  |
| --- | --- | --- |
| Error/Issue | Test | Expected Outcome |
|  |  |  |

**Test Execution**

**Test Results**

**Interpretation of Results**

|  |  |  |
| --- | --- | --- |
| Expected Outcome | Actual Outcome | Discussion |
|  |  |  |

# Work Log

|  |  |  |
| --- | --- | --- |
| Student | Activity | Period |
| Harry | Constants | February 2018 |
| Harry | Setting up Commands | February 2018 |
| Harry | Room Temperature States | February/March 2018 |
| Jordan | Room Temperature Checks | March 2018 |
| Jordan | Room Temperature Tick | March 2018 |
| Harry | Input Handling | March 2018 |
| Jordan/Harry | FireEngine Coolant and Room temperature reduction | March 2018 |
| Jordan/Harry | Adding/Removing rooms from Delegate | March 2018 |
| Jordan | End Game Check | March 2018 |
| Jordan | Refining Code (Memory Usage/Reduction of cluttered, inefficient code) | March 2018 |
| Harry | Intro/Instructions Dialogue | March 2018 |
| Harry | Code Comments | March 2018 |
| Jordan | Use Case/Class Diagrams | April 2018 |
| Jordan/Harry | Use Case/Class Catalogues | April 2018 |
| Harry | Sequence Diagrams | April 2018 |
| Harry | Testing Documentation | April 2018 |