

Aircraft Landing Simulation System Test Report

1. Compute Simulation State method

a. Source code of ComputeSimulationState.java with code coverage

The screenshot shows the Eclipse IDE interface with the Java - STProject/src/inputs/ComputeSimulationState.java - Eclipse window open. The left side displays the Package Explorer with various Java files and resources. The main editor area shows the source code of ComputeSimulationState.java. The code is annotated with color-coded highlights indicating code coverage: green for executed code, red for unexecuted code, and yellow for partially executed code. The code itself is as follows:

```

1 package inputs;
2
3 import java.io.BufferedReader;
4 import java.io.FileNotFoundException;
5 import java.io.FileReader;
6 import java.io.IOException;
7
8 public class ComputeSimulationState {
9
10    private static int timeUntilLanding;
11    private static int speed;
12    private static int speedIncrement;
13    private static int altitude;
14    private static int altitudeIncrement;
15    private static Position gearPosition;
16    private static Position selectedGearPosition;
17    private static String throttleCmd;
18    private static String elevonCmd;
19    private static boolean airBrakeWarningOn;
20    private static boolean gearOverrideWarningOn;
21    private static boolean gearNotDownAlarmOn;
22    private static boolean gearAirSpeedAlarmOn;
23    private static boolean gearUpCommand;
24    private static boolean silenceAlarms;
25    private static int pass=0;
26    private static int fail=0;
27    private static int i=1;
28
29    public static void computeSimulationState(int altitude,int speed,int timeUntilLanding,Position currentGearPosition,
30                                              String throttleCmd,String elevonCmd){
31        if (throttleCmd == "+")
32            ComputeSimulationState.speed=speed+10;
33        else
34        {
35            if (throttleCmd=="-")
36                ComputeSimulationState.speed=speed-10;
37            else
38                ComputeSimulationState.speed=speed;
39        }
40    }
41}

```

The status bar at the bottom indicates the code is Writable and the current time is 13:33.

Java - STProject/src/inputs/ComputeSimulationState.java - Eclipse

```

File Edit Source Refactor Navigate Search Project Run Window Help
Quick Access Java PyDev Debug Team Synchronizing Java EE

Package Explorer X
P2PFileShare2 [NikhilnvjshareCN2.m
STProject [STProject master]
src
controller
DisplayController.java
DisplayControllerTest.java
inputs
Altitude.java
ComputeSimulationState.java
DataSensors.java
ElevonCmd.java
GearPosition.java
Position.java
SelectedGearPositionSwitch.java
SilenceAlarmSwitch.java
Speed.java
StatusValues.java
ThrottleCmd.java
Timers.java
TimeToLand.java
TwoPositionSwitch.java
Referenced Libraries
JRE System Library [JavaSE-1.8]
JRE System Library [jre1.8.0_31]
JUnit 4
1.csv
forms-1.0.3.jar

ComputeSimulationState.java X
35     if (throttleCmd=="-")
36         ComputeSimulationState.speed=speed-10;
37     else
38         ComputeSimulationState.speed = speed;
39 }
40     if (elevonCmd=="+")
41         ComputeSimulationState.altitude=altitude+20;
42     else
43     {
44         if (elevonCmd =="-")
45             ComputeSimulationState.altitude=altitude-20;
46         else
47             ComputeSimulationState.altitude=altitude;
48 }
49 ComputeSimulationState.airBrakeWarningOn = (ComputeSimulationState.speed >= 250) && (timeUntilLanding < 60);
50 ComputeSimulationState.gearOverrideWarningOn = (currentGearPosition == Position.Down) && (ComputeSimulationState.speed>400);
51 ComputeSimulationState.gearNotDownAlarmOn = (currentGearPosition == Position.Up) && ((timeUntilLanding <120) || (ComputeSimulationState.speed>300));
52 ComputeSimulationState.gearAirspeedAlarmOn = (currentGearPosition== Position.Down) && (ComputeSimulationState.speed>300);
53 if (!gearOverrideWarningOn)
54     currentGearPosition = selectedGearPosition;
55 }

public static void main(String[] args) {
56     String csvFile = "1.csv";
57     BufferedReader br = null;
58     String line = "";
59     String cvsSplitBy = ",";
60     try {
61         br = new BufferedReader(new FileReader(csvFile));
62         line = br.readLine();
63         while ((line = br.readLine()) != null) {
64             // use comma as separator
65             String[] inputs = line.split(cvsSplitBy);
66
67             speed = Integer.parseInt(inputs[0]);
68             if(inputs[1].equals("Y"))
69                 inputs[1] = "Down";
70             else
71                 inputs[1] = "Up";
72             gearPosition = Position.valueOf(inputs[1]);
73             altitude = Integer.parseInt(inputs[2]);
74             timeUntilLanding = Integer.parseInt(inputs[3]);
75             ComputeSimulationState.computeSimulationState(altitude, speed, timeUntilLanding, gearPosition,null,null);
76             airBrakeWarningOn = ComputeSimulationState.isAirBrakeWarningOn();
77             gearOverrideWarningOn = ComputeSimulationState.isGearOverrideWarningOn();
78             gearNotDownAlarmOn = ComputeSimulationState.isGearNotDownAlarmOn();
79             gearAirspeedAlarmOn = ComputeSimulationState.isGearAirspeedAlarmOn();
80             gearUpCommand = ComputeSimulationState.isGearOverrideWarningOn();
81             if(ComputeSimulationState.gearAirspeedAlarmOn || ComputeSimulationState.gearNotDownAlarmOn)
82                 silenceAlarms = true;
83             else
84                 silenceAlarms = false;
85             if(gearNotDownAlarmOn == Boolean.valueOf(inputs[4]) && gearAirspeedAlarmOn == Boolean.valueOf(inputs[5]) &&
86                 airBrakeWarningOn == Boolean.valueOf(inputs[6]) && gearOverrideWarningOn == Boolean.valueOf(inputs[7]) &&
87                 gearUpCommand == Boolean.valueOf(inputs[8])&& silenceAlarms == Boolean.valueOf(inputs[9]))
88                 pass++;
89             else
90                 fail++;
91             System.out.println("Test Case No: "+i+ " passed.");
92         }
93     }
94     catch (FileNotFoundException e)
95     {
96         e.printStackTrace();
97     }
98     i++;
99 }
100 }
101 catch (IOException ei)
102 {
103     ei.printStackTrace();
104 } catch (Exception e)
105 {
106 }
```

Java - STProject/src/inputs/ComputeSimulationState.java - Eclipse

```

File Edit Source Refactor Navigate Search Project Run Window Help
Quick Access Java PyDev Debug Team Synchronizing Java EE

Package Explorer X
P2PFileShare2 [NikhilnvjshareCN2.m
STProject [STProject master]
src
controller
DisplayController.java
DisplayControllerTest.java
inputs
Altitude.java
ComputeSimulationState.java
DataSensors.java
ElevonCmd.java
GearPosition.java
Position.java
SelectedGearPositionSwitch.java
SilenceAlarmSwitch.java
Speed.java
StatusValues.java
ThrottleCmd.java
Timers.java
TimeToLand.java
TwoPositionSwitch.java
Referenced Libraries
JRE System Library [JavaSE-1.8]
JRE System Library [jre1.8.0_31]
JUnit 4
1.csv
forms-1.0.3.jar

ComputeSimulationState.java X
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104

speed = Integer.parseInt(inputs[0]);
if(inputs[1].equals("Y"))
    inputs[1] = "Down";
else
    inputs[1] = "Up";
gearPosition = Position.valueOf(inputs[1]);
altitude = Integer.parseInt(inputs[2]);
timeUntilLanding = Integer.parseInt(inputs[3]);
ComputeSimulationState.computeSimulationState(altitude, speed, timeUntilLanding, gearPosition,null,null);
airBrakeWarningOn = ComputeSimulationState.isAirBrakeWarningOn();
gearOverrideWarningOn = ComputeSimulationState.isGearOverrideWarningOn();
gearNotDownAlarmOn = ComputeSimulationState.isGearNotDownAlarmOn();
gearAirspeedAlarmOn = ComputeSimulationState.isGearAirspeedAlarmOn();
gearUpCommand = ComputeSimulationState.isGearOverrideWarningOn();
if(ComputeSimulationState.gearAirspeedAlarmOn || ComputeSimulationState.gearNotDownAlarmOn)
    silenceAlarms = true;
else
    silenceAlarms = false;
if(gearNotDownAlarmOn == Boolean.valueOf(inputs[4]) && gearAirspeedAlarmOn == Boolean.valueOf(inputs[5]) &&
    airBrakeWarningOn == Boolean.valueOf(inputs[6]) && gearOverrideWarningOn == Boolean.valueOf(inputs[7]) &&
    gearUpCommand == Boolean.valueOf(inputs[8])&& silenceAlarms == Boolean.valueOf(inputs[9]))
    pass++;
else
    fail++;
System.out.println("Test Case No: "+i+ " passed.");
}
else
{
    fail++;
    System.out.println("Test Case No: "+i+ " failed.");
}
i++;
}
catch (FileNotFoundException e)
{
    e.printStackTrace();
}
} catch (IOException ei)
{
    ei.printStackTrace();
} catch (Exception e)
{
    e.printStackTrace();
}
```

Java - STProject/src/inputs/ComputeSimulationState.java - Eclipse

```

File Edit Source Refactor Navigate Search Project Run Window Help
Quick Access Java PyDev Debug Team Synchronizing Java EE

Package Explorer X ComputeSimulationState.java X
99     i++;
100    }
101    catch (FileNotFoundException e) {
102        e.printStackTrace();
103    } catch (IOException e1) {
104        e1.printStackTrace();
105    }
106    System.out.println("Total No. of Test Cases passed: "+pass+" and failed: "+fail+".");
107
108    if (br != null) {
109        try {
110            br.close();
111        } catch (IOException e) {
112            e.printStackTrace();
113        }
114    }
115}
116
117 public static int getTimeUntilLanding() {
118     return timeUntilLanding;
119 }
120
121 public static void setTimeUntilLanding(int timeUntilLanding) {
122     ComputeSimulationState.timeUntilLanding = timeUntilLanding;
123 }
124
125 public static int getSpeed() {
126     return speed;
127 }
128
129 public static void setSpeed(int speed) {
130     ComputeSimulationState.speed = speed;
131 }
132
133 public static int getSpeedIncrement() {
134     return speedIncrement;
135 }

```

Java - STProject/src/inputs/ComputeSimulationState.java - Eclipse

```

File Edit Source Refactor Navigate Search Project Run Window Help
Quick Access Java PyDev Debug Team Synchronizing Java EE

Package Explorer X ComputeSimulationState.java X
128
129     public static void setSpeed(int speed) {
130         ComputeSimulationState.speed = speed;
131     }
132
133     public static int getSpeedIncrement() {
134         return speedIncrement;
135     }
136
137     public static void setSpeedIncrement(int speedIncrement) {
138         ComputeSimulationState.speedIncrement = speedIncrement;
139     }
140
141     public static int getAltitude() {
142         return altitude;
143     }
144
145     public static void setAltitude(int altitude) {
146         ComputeSimulationState.altitude = altitude;
147     }
148
149     public static int getAltitudeIncrement() {
150         return altitudeIncrement;
151     }
152
153     public static void setAltitudeIncrement(int altitudeIncrement) {
154         ComputeSimulationState.altitudeIncrement = altitudeIncrement;
155     }
156
157     public static Position getGearPosition() {
158         return gearPosition;
159     }
160
161     public static void setGearPosition(Position gearPosition) {
162         ComputeSimulationState.gearPosition = gearPosition;
163     }
164

```

Java - STProject/src/inputs/ComputeSimulationState.java - Eclipse

```
159     }
160
161     public static void setGearPosition(Position gearPosition) {
162         ComputeSimulationState.gearPosition = gearPosition;
163     }
164
165     public static Position getSelectedGearPosition() {
166         return selectedGearPosition;
167     }
168
169     public static void setSelectedGearPosition(Position selectedGearPosition) {
170         ComputeSimulationState.selectedGearPosition = selectedGearPosition;
171     }
172
173     public static String getThrottleCmd() {
174         return throttleCmd;
175     }
176
177     public static void setThrottleCmd(String throttleCmd) {
178         ComputeSimulationState.throttleCmd = throttleCmd;
179     }
180
181     public static String getElevonCmd() {
182         return elevonCmd;
183     }
184
185     public static void setElevonCmd(String elevonCmd) {
186         ComputeSimulationState.elevonCmd = elevonCmd;
187     }
188
189     public static boolean isAirBrakeWarningOn() {
190         return airBrakeWarningOn;
191     }
192
193     public static void setAirBrakeWarningOn(boolean airBrakeWarningOn) {
194         ComputeSimulationState.airBrakeWarningOn = airBrakeWarningOn;
195     }
196 }
```

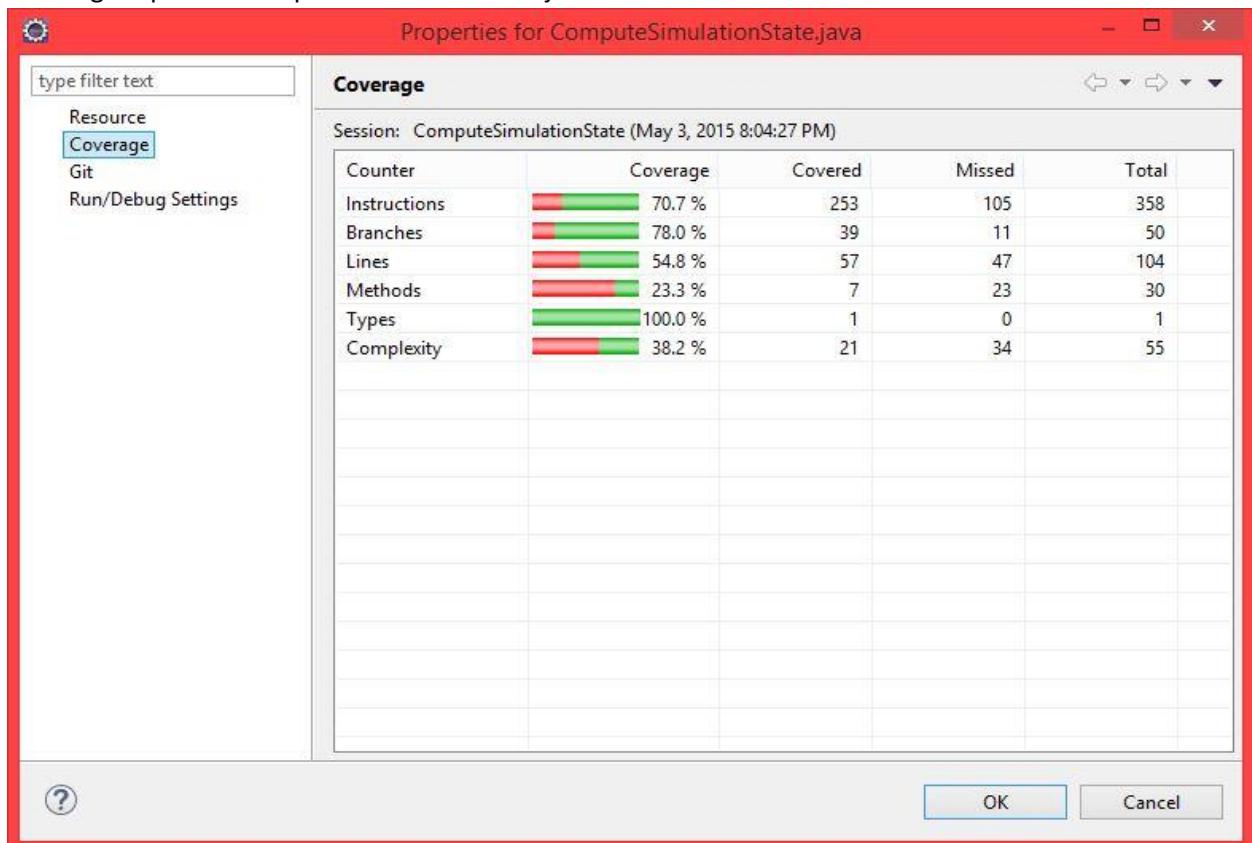
Writable | Smart Insert | 13:33

The screenshot shows the Eclipse IDE interface with the following details:

- Title Bar:** Java - STProject/src/inputs/ComputeSimulationState.java - Eclipse
- Menu Bar:** File, Edit, Source, Refactor, Navigate, Search, Project, Run, Window, Help
- Toolbar:** Includes icons for New, Open, Save, Cut, Copy, Paste, Find, Replace, and others.
- Quick Access:** Shows Java, PyDev, Debug, Team Synchronizing, and Java EE.
- Package Explorer:** Displays the project structure:
 - P2PFileShare2 [NikhilnvjshareCN2.m]
 - STProject [STProject master]
 - src
 - controller
 - DisplayController.java
 - DisplayControllerTest.java
 - inputs
 - Altitude.java
 - ComputeSimulationState.java
 - DataSensors.java
 - ElevonCmd.java
 - GearPosition.java
 - Position.java
 - SelectedGearPositionSwitch.java
 - SilenceAlarmSwitch.java
 - Speed.java
 - StatusValues.java
 - ThrottleCmd.java
 - Timers.java
 - TimeToLand.java
 - TwoPositionSwitch.java
 - Referenced Libraries
 - JRE System Library [JavaSE-1.8]
 - JRE System Library [jre1.8.0_31]
 - JUnit 4
 - 1.csv
 - forms-1.0.3.jar
 - Code Editor:** The file ComputeSimulationState.java is open, showing the following code:

```
184
185     public static void setElevonCmd(String elevonCmd) {
186         ComputeSimulationState.elevonCmd = elevonCmd;
187     }
188
189     public static boolean isAirBrakeWarningOn() {
190         return airBrakeWarningOn;
191     }
192
193     public static void setAirBrakeWarningOn(boolean airBrakeWarningOn) {
194         ComputeSimulationState.airBrakeWarningOn = airBrakeWarningOn;
195     }
196
197     public static boolean isGearOverrideWarningOn() {
198         return gearOverrideWarningOn;
199     }
200
201     public static void setGearOverrideWarningOn(boolean gearOverrideWarningOn) {
202         ComputeSimulationState.gearOverrideWarningOn = gearOverrideWarningOn;
203     }
204
205     public static boolean isGearNotDownAlarmOn() {
206         return gearNotDownAlarmOn;
207     }
208
209     public static void setGearNotDownAlarmOn(boolean gearNotDownAlarmOn) {
210         ComputeSimulationState.gearNotDownAlarmOn = gearNotDownAlarmOn;
211     }
212
213     public static boolean isGearAirSpeedAlarmOn() {
214         return gearAirSpeedAlarmOn;
215     }
216
217     public static void setGearAirSpeedAlarmOn(boolean gearAirSpeedAlarmOn) {
218         ComputeSimulationState.gearAirSpeedAlarmOn = gearAirSpeedAlarmOn;
219     }
220 }
```
 - Bottom Status Bar:** Writable, Smart Insert, 13:33

b. Coverage report of ComputeSimulationState.java



c. Test result output on the console as we run the ComputeSimulationState method

```

1 package inputs;
2
3 import java.io.BufferedReader;
4
5 public class ComputeSimulationState {
6
7     private static int timeUntilLanding;
8     private static int speed;
9     private static int speedIncrement;
10    private static int altitude;
11    private static int altitudeIncrement;
12
13    public void compute() {
14        System.out.println("Time until landing: " + timeUntilLanding);
15        System.out.println("Current speed: " + speed);
16        System.out.println("Speed increment: " + speedIncrement);
17        System.out.println("Current altitude: " + altitude);
18        System.out.println("Altitude increment: " + altitudeIncrement);
19    }
20
21    public static void main(String[] args) {
22        BufferedReader reader = new BufferedReader(new InputStreamReader(System.in));
23        String input;
24
25        try {
26            while ((input = reader.readLine()) != null) {
27                if ("start".equals(input)) {
28                    ComputeSimulationState state = new ComputeSimulationState();
29                    state.compute();
30                } else {
31                    System.out.println("Unknown command: " + input);
32                }
33            }
34        } catch (IOException e) {
35            e.printStackTrace();
36        }
37    }
38}

```

Console output:

```

Test Case No: 1 passed.
Test Case No: 2 passed.
Test Case No: 3 passed.
Test Case No: 4 passed.
Test Case No: 5 passed.
Test Case No: 6 passed.
Test Case No: 7 passed.
Test Case No: 8 passed.
Test Case No: 9 passed.
Test Case No: 10 passed.
Test Case No: 11 passed.
Test Case No: 12 passed.
Test Case No: 13 passed.
Test Case No: 14 passed.
Test Case No: 15 passed.
Test Case No: 16 passed.
Test Case No: 17 passed.
Test Case No: 18 passed.
Test Case No: 19 passed.
Test Case No: 20 passed.
Test Case No: 21 passed.
Test Case No: 22 passed.

```

The screenshot shows the Eclipse IDE interface. The left pane displays the Package Explorer with several Java files under the 'src' directory, including Controller, DisplayController, DisplayControllerTest, Inputs, Altitude, ComputeSimulationState, DataSensors, ElevonCmd, GearPosition, Position, SelectedGearPositionSwitch, SilenceAlarmSwitch, Speed, StatusValues, ThrottleCmd, Timers, TimeToLand, and TwoPositionSwitch. The right pane shows the Java editor with the code for ComputeSimulationState.java:

```

1 package inputs;
2
3 import java.io.BufferedReader;
4
5 public class ComputeSimulationState {
6
7     private static int timeUntilLanding;
8     private static int speed;
9     private static int speedIncrement;
10    private static int altitude;
11    private static int altitudeIncrement;
12
13}
14

```

Below the editor is a terminal window showing the output of a JUnit test run:

```

<terminated> ComputeSimulationState [Java Application] C:\Program Files (x86)\Java\jre1.8.0_31\bin\javaw.exe (May 3, 2015, 8:04:27 PM)
Test Case No: 18 passed.
Test Case No: 19 passed.
Test Case No: 20 passed.
Test Case No: 21 passed.
Test Case No: 22 passed.
Test Case No: 23 passed.
Test Case No: 24 passed.
Test Case No: 25 passed.
Test Case No: 26 passed.
Test Case No: 27 passed.
Test Case No: 28 passed.
Test Case No: 29 passed.
Test Case No: 30 passed.
Test Case No: 31 passed.
Test Case No: 32 passed.
Test Case No: 33 passed.
Test Case No: 34 passed.
Test Case No: 35 passed.
Test Case No: 36 passed.
Test Case No: 37 passed.
Total No. of Test Cases passed: 37 and failed: 0.

```

d. Test summary:

- I. We have tested the *computeSimulationState* method by reading the inputs from the 'TestCases.csv' file and comparing the outputs with the expected outputs provided in the csv file.

- Test case Inputs: Speed, Gear Down, Altitude, Time.
- Test Case Outputs: Gear Not Down alarm, Gear Air Speed alarm, Air Brake warning, Gear Override warning, Gear-up command, Silence Alarm.



- II. In the above file, we have 37 test cases, which were run by the *computeSimulationState* method of the *ComputeSimulationState* class which resulted in 100% test success i.e. all the 37 test cases passed successfully and 0 failed. The same can be seen in console output provided above.
- III. As shown in the coverage properties above, we have achieved overall 70.7% code coverage. The shortfall of 29.3% is observed because in addition to the *computeSimulationState* method, the *ComputeSimulationState* class also has some code which is used only by the *DisplayController* class to get and set input variables for computing alarms and warnings. Also the code that hasn't been covered in this test-case run is the one handling the button click events (+, -, Up, Down) of the *DisplayController* JFrame UI. Also as all the test cases passed successfully, the exception handling statements were not covered in the test run.

Element		Coverage	Covered Instructions	Missed Instructions
inputs		64.2 %	303	
ComputeSimulationState.java	J	70.7 %	253	
DataSensors.java	J	0.0 %	0	
StatusValues.java	J	0.0 %	0	
Timers.java	J	0.0 %	0	
TwoPositionSwitch.java	J	0.0 %	0	
Altitude.java	J	0.0 %	0	
ElevonCmd.java	J	0.0 %	0	
GearPosition.java	J	0.0 %	0	
SelectedGearPositionSwitch.java	J	0.0 %	0	
SilenceAlarmSwitch.java	J	0.0 %	0	
Speed.java	J	0.0 %	0	
ThrottleCmd.java	J	0.0 %	0	
TimeToLand.java	J	0.0 %	0	
Position.java	J	100.0 %	50	

- IV. The test cases helped us in uncovering some issues like uninitialized variables, missing boundary values and some minor logical errors. After resolving them, all the test cases passed successfully.
- V. Here, we can observe that we are only testing specific alarms and warnings but we haven't checked the conditions such as no alarms/warnings etc i.e. we are only testing the positive test cases.

2. DisplayController Class JUnit Test Run:

To test and display the 8 scenarios, we comment the main method of the ComputeSimulationState class, that is used to test the above-mentioned 37 test cases, and the main method of the DisplayController class, that is used to run real-time simulation of the aircraft, as these are not required while running the Junit test class DisplayControllerTest.java. If left uncommented, they would decrease the code coverage as those statements would never be executed during the test run.

- a. Source code of DisplayControllerTest.java with code coverage

Java - STProject/src/controller/DisplayControllerTest.java - Eclipse

```

File Edit Source Refactor Search Project Run Window Help
Quick Access Java PyDev Debug Team Synchronizing Java EE

Package Explorer JUnit
Finished after 80.602 seconds
Runs: 1/1 Errors: 0 Failures: 0
controller.DisplayControllerTest [Run]
test (80.524 s)

Failure Trace

DisplayController.java DisplayControllerTest.java ComputeSimulationState.java
1 package controller;
2
3 import inputs.ComputeSimulationState;
4 import inputs.Position;
5
6 import org.junit.After;
7 import org.junit.AfterClass;
8 import org.junit.Before;
9 import org.junit.BeforeClass;
10 import org.junit.Test;
11
12 public class DisplayControllerTest {
13     private static int timeUntilLanding;
14     private static int speed;
15     private static int altitude;
16     private static Position gearPosition;
17     @BeforeClass
18     public static void setUpBeforeClass() throws Exception {
19     }
20
21     @AfterClass
22     public static void tearDownAfterClass() throws Exception {
23     }
24
25     @Before
26     public void setUp() throws Exception {
27     }
28
29     @After
30     public void tearDown() throws Exception {
31     }
32
33     @Test
34     public void test() {
35         DisplayController pilot = new DisplayController();
36         pilot.run();
37         String[][] inputs = new String[][]{

```

Java - STProject/src/controller/DisplayControllerTest.java - Eclipse

```

File Edit Source Refactor Search Project Run Window Help
Quick Access Java PyDev Debug Team Synchronizing Java EE

Package Explorer JUnit
Finished after 80.602 seconds
Runs: 1/1 Errors: 0 Failures: 0
controller.DisplayControllerTest [Run]
test (80.524 s)

Failure Trace

DisplayController.java DisplayControllerTest.java ComputeSimulationState.java
35     DisplayController pilot = new DisplayController();
36     pilot.run();
37     String[][] inputs = new String[][]{
38         {"249", "false", "1500", "59"}, 
39         {"401", "false", "1500", "59"}, 
40         {"401", "true", "1500", "59"}, 
41         {"401", "true", "1500", "121"}, 
42         {"301", "true", "1500", "59"}, 
43         {"301", "true", "1500", "121"}, 
44         {"258", "true", "1500", "59"}, 
45         {"415", "false", "1025", "130"}, 
46     };
47
48
49     for (int i=0;i<8;i++){
50         System.out.println("Display Scenario:"+(i+1));
51         speed = Integer.parseInt(inputs[i][0]);
52         altitude = Integer.parseInt(inputs[i][2]);
53         timeUntilLanding = Integer.parseInt(inputs[i][3]);
54         if(inputs[i][1].equals("true"))
55             inputs[i][1] = "Down";
56         else
57             inputs[i][1] = "Up";
58         gearPosition = Position.valueOf(inputs[i][1]);
59         ComputeSimulationState.setAltitude(altitude);
60         ComputeSimulationState.setGearPosition(gearPosition);
61         ComputeSimulationState.setSpeed(speed);
62         ComputeSimulationState.setTimeUntilLanding(timeUntilLanding);
63         ComputeSimulationState.computeSimulationState(altitude, speed, timeUntilLanding, gearPosition, null, null);
64         DisplayController.lblAltitude.setText(String.valueOf(ComputeSimulationState.getAltitude()));
65         DisplayController.lblSpeed.setText(String.valueOf(ComputeSimulationState.getSpeed()));
66         if(String.valueOf(ComputeSimulationState.getGearPosition()).equals("Up"))
67         {
68             DisplayController.lblDown.setText("");
69             DisplayController.lblUp.setText(String.valueOf(ComputeSimulationState.getGearPosition()));
70         }
71     }

```

Java - STProject/src/controller/DisplayControllerTest.java - Eclipse

```

File Edit Source Refactor Navigate Search Project Run Window Help
Quick Access Java PyDev Debug Team Synchronizing Java EE

Package Explorer JUnit
Finished after 80.602 seconds
Runs: 1/1 Errors: 0 Failures: 0
controller.DisplayControllerTest [Run]
test (80.524 s)

Failure Trace

DisplayController.java DisplayControllerTest.java ComputeSimulationState.java

52     altitude = Integer.parseInt(inputs[i][2]);
53     timeUntilLanding = Integer.parseInt(inputs[i][3]);
54     if(inputs[i][1].equals("true"))
55         inputs[i][1] = "Down";
56     else
57         inputs[i][1] = "Up";
58     gearPosition = Position.valueOf(inputs[i][1]);
59     ComputeSimulationState.setAltitude(altitude);
60     ComputeSimulationState.setGearPosition(gearPosition);
61     ComputeSimulationState.setSpeed(speed);
62     ComputeSimulationState.setTimeUntilLanding(timeUntilLanding);
63     ComputeSimulationState.computeSimulationState(altitude, speed, timeUntilLanding, gearPosition, null, null);
64     DisplayController.lblAltitude.setText(String.valueOf(ComputeSimulationState.getAltitude()));
65     DisplayController.lblSpeed.setText(String.valueOf(ComputeSimulationState.getSpeed()));
66     if(String.valueOf(ComputeSimulationState.getGearPosition()).equals("Up"))
67     {
68         DisplayController.lblDown.setText("");
69         DisplayController.lblUp.setText(String.valueOf(ComputeSimulationState.getGearPosition()));
70     }
71     else
72     {
73         DisplayController.lblUp.setText("");
74         DisplayController.lblDown.setText(String.valueOf(ComputeSimulationState.getGearPosition()));
75     }
76     DisplayController.lblLandingTimeCountDown.setText(String.valueOf(ComputeSimulationState.getTimeUntilLanding()));
77     ComputeSimulationState.computeSimulationState(altitude, speed, timeUntilLanding, gearPosition, null, null);
78     pilot.setWarningLabels();
79     try {
80         Thread.sleep(10000);
81     } catch (InterruptedException e) {
82         e.printStackTrace();
83     }
84 }
85
86
87 }
88

```

Writable Smart Insert 33:10

Source code of DisplayController.java with code coverage,

Java - STProject/src/controller/DisplayController.java - Eclipse

```

File Edit Source Refactor Navigate Search Project Run Window Help
Quick Access Java PyDev Debug Team Synchronizing Java EE

Package Explorer JUnit
Finished after 80.602 seconds
Runs: 1/1 Errors: 0 Failures: 0
controller.DisplayControllerTest [Run]
test (80.524 s)

Failure Trace

DisplayController.java DisplayControllerTest.java ComputeSimulationState.java

1 package controller;
2 import inputs.ComputeSimulationState;
3
4 public class DisplayController extends JFrame {
5
6     private static final long serialVersionUID = 1L;
7     public static JLabel lblAltitude;
8     public static JLabel lblSpeed;
9     public static JLabel lblUp;
10    public static JLabel lblDown;
11    public static JLabel lblAirResistance;
12    public static JLabel lblAltitudeLoss;
13    public static JLabel lblLandingTimeCountDown;
14    public static JPanel contentPane;
15    public static JLabel lblLandingGearOverride;
16    public static JLabel lbGearAirSpeed;
17    public static JLabel lbGearNotDown;
18    public static JLabel lbBreakingOverride;
19    public static JLabel lbTimeUnit;
20
21    public DisplayController() {
22        setTitle("Pilot Display Scenario:");
23        setDefaultCloseOperation(3);
24        setBounds(100, 100, 494, 368);
25        this.contentPane = new JPanel();
26        this.contentPane.setBorder(new EmptyBorder(5, 5, 5, 5));
27        setContentPane(this.contentPane);
28        this.contentPane.setLayout(null);
29
30        JLabel lblAltitude = new JLabel("Altitude");
31        lblAltitude.setHorizontalAlignment(0);
32        lblAltitude.setBounds(25, 116, 63, 14);
33        this.contentPane.add(lblAltitude);
34
35        JLabel lblLandingGear = new JLabel("Landing Gear");
36        lblLandingGear.setHorizontalAlignment(0);
37        lblLandingGear.setBounds(25, 170, 89, 14);
38
39    }
40
41
42
43
44
45
46
47
48
49
50
51

```

Writable Smart Insert 73:24

Java - STProject/src/controller/DisplayController.java - Eclipse

```

File Edit Source Refactor Navigate Search Project Run Window Help
Quick Access Java PyDev Debug Team Synchronizing Java EE

Package Explorer JUnit
Finished after 80.602 seconds
Runs: 1/1 Errors: 0 Failures: 0
controller.DisplayControllerTest [Runn...
test (80.524 s)

Failure Trace

```

```

1 DisplayController.java 2 DisplayControllerTest.java 3 ComputeSimulationState.java
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88

```

Writable Smart Insert 73:24

Java - STProject/src/controller/DisplayController.java - Eclipse

```

File Edit Source Refactor Navigate Search Project Run Window Help
Quick Access Java PyDev Debug Team Synchronizing Java EE

Package Explorer JUnit
Finished after 80.602 seconds
Runs: 1/1 Errors: 0 Failures: 0
controller.DisplayControllerTest [Runn...
test (80.524 s)

Failure Trace

```

```

1 DisplayController.java 2 DisplayControllerTest.java 3 ComputeSimulationState.java
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124

```

Writable Smart Insert 73:24

Java - STProject/src/controller/DisplayController.java - Eclipse

```

123     Position currentGearPosition=(lbUp.getText()!="")?Position.valueOf(lbUp.getText()):Position.valueOf(lbDown.getText());
124     ComputeSimulationState.computeSimulationState(Integer.parseInt(lbAltitude.getText()), Integer.parseInt(lbSpeed.getText()), Int
125     currentGearPosition, null);
126     lbSpeed.setText(String.valueOf(ComputeSimulationState.getSpeed()));
127     setWarningLabels();
128   });
129 
130   JButton btIncreaseAltitude = new JButton("+");
131   btIncreaseAltitude.setBounds(188, 100, 51, 23);
132   this.contentPane.add(btIncreaseAltitude);
133   btIncreaseAltitude.addActionListener(new ActionListener() {
134     @Override
135     public void actionPerformed(ActionEvent e) {
136       Position currentGearPosition=(lbUp.getText()!="")?Position.valueOf(lbUp.getText()):Position.valueOf(lbDown.getText());
137       ComputeSimulationState.computeSimulationState(Integer.parseInt(lbAltitude.getText()), Integer.parseInt(lbSpeed.getText()), Int
138       currentGearPosition, null,"+");
139       lbAltitude.setText(String.valueOf(ComputeSimulationState.getAltitude()));
140       setWarningLabels();
141     }
142   });
143   JButton btDecreaseAltitude = new JButton("-");
144   btDecreaseAltitude.setBounds(188, 128, 51, 23);
145   this.contentPane.add(btDecreaseAltitude);
146   btDecreaseAltitude.addActionListener(new ActionListener() {
147     @Override
148     public void actionPerformed(ActionEvent e) {
149       Position currentGearPosition=(lbUp.getText()!="")?Position.valueOf(lbUp.getText()):Position.valueOf(lbDown.getText());
150       ComputeSimulationState.computeSimulationState(Integer.parseInt(lbAltitude.getText()), Integer.parseInt(lbSpeed.getText()), Int
151       currentGearPosition, null,"-");
152       lbAltitude.setText(String.valueOf(ComputeSimulationState.getAltitude()));
153       setWarningLabels();
154     }
155   });
156 
157   JButton btUp = new JButton("Up");
158   btUp.setBounds(177, 152, 74, 23);

```

Java - STProject/src/controller/DisplayController.java - Eclipse

```

159   btUp.setBounds(177, 152, 74, 23);
160   this.contentPane.add(btUp);
161   btUp.addActionListener(new ActionListener() {
162     @Override
163     public void actionPerformed(ActionEvent e) {
164       Position currentGearPosition=(lbUp.getText()!="")?Position.valueOf(lbUp.getText()):Position.valueOf(lbDown.getText());
165       ComputeSimulationState.computeSimulationState(Integer.parseInt(lbAltitude.getText()), Integer.parseInt(lbSpeed.getText()), Int
166       currentGearPosition, null,null);
167       ComputeSimulationState.setSelectedGearPosition(Position.Up);
168       lbDown.setText("");
169       lbUp.setText(String.valueOf(ComputeSimulationState.getSelectedGearPosition()));
170       setWarningLabels();
171     }
172   });
173 
174   JButton btDown = new JButton("Down");
175   btDown.setBounds(177, 183, 74, 23);
176   this.contentPane.add(btDown);
177   btDown.addActionListener(new ActionListener() {
178     @Override
179     public void actionPerformed(ActionEvent e) {
180       Position currentGearPosition=(lbUp.getText()!="")?Position.valueOf(lbUp.getText()):Position.valueOf(lbDown.getText());
181       ComputeSimulationState.computeSimulationState(Integer.parseInt(lbAltitude.getText()), Integer.parseInt(lbSpeed.getText()), Int
182       currentGearPosition, null,null);
183       ComputeSimulationState.setSelectedGearPosition(Position.Down);
184       lbDown.setText(String.valueOf(ComputeSimulationState.getSelectedGearPosition()));
185       lbUp.setText("");
186       setWarningLabels();
187     }
188   });
189 
190   lbUAirResistance = new JLabel("5 mph/sec");
191   lbUAirResistance.setBounds(378, 59, 63, 14);
192   this.contentPane.add(lbUAirResistance);
193 
194   lbAltitudeLoss = new JLabel("20 ft/sec ");
195   lbAltitudeLoss.setBounds(378, 116, 63, 14);

```

Java - STProject/src/controller/DisplayController.java - Eclipse

```

File Edit Source Refactor Navigate Search Project Run Window Help
Quick Access Java PyDev Debug Team Synchronizing Java EE

Package Explorer JUnit
Finished after 80.602 seconds
Runs: 1/1 Errors: 0 Failures: 0
controller.DisplayControllerTest [Run]
test (80.524 s)

Failure Trace

DisplayController.java DisplayControllerTest.java ComputeSimulationState.java

191     lbUAirResistance.setBounds(378, 59, 63, 14);
192     this.contentPane.add(lbUAirResistance);
193
194     lbUAltitudeLoss = new JLabel("20 ft/sec ");
195     lbUAltitudeLoss.setBounds(378, 116, 63, 14);
196     this.contentPane.add(lbUAltitudeLoss);
197
198     lbLandingGearOverride = new JLabel("Landing Gear Override");
199     lbLandingGearOverride.setBounds(10, 286, 132, 13);
200     this.contentPane.add(lbLandingGearOverride);
201
202     lbBrakingOverride = new JLabel("Braking Override");
203     lbBrakingOverride.setBounds(142, 285, 97, 14);
204     this.contentPane.add(lbBrakingOverride);
205
206     lbGearNotDown = new JLabel("Gear Not Down");
207     lbGearNotDown.setBounds(270, 285, 96, 14);
208     this.contentPane.add(lbGearNotDown);
209
210     lbGearAirSpeed = new JLabel("Gear Air Speed");
211     lbGearAirSpeed.setBounds(366, 285, 89, 14);
212     this.contentPane.add(lbGearAirSpeed);
213
214     JCheckBox cbSilenceAuralAlarm = new JCheckBox("Silence Aural Alarm");
215     cbSilenceAuralAlarm.setBounds(309, 245, 146, 23);
216     this.contentPane.add(cbSilenceAuralAlarm);
217     cbSilenceAuralAlarm.addActionListener(new ActionListener() {
218         @Override
219         public void actionPerformed(ActionEvent e) {
220             AbstractButton abstractButton = (AbstractButton) e.getSource();
221             boolean selected = abstractButton.getModel().isSelected();
222             if(selected)
223             {
224                 setWarningLabels();
225                 if(lbGearNotDown.getForeground() == Color.RED)
226                 {
227                     lbGearNotDown.setForeground(Color.YELLOW);
228                 }
229             }
230         }
231     });
232
233 });
234 });
235 });

JLabel lbCountDown = new JLabel("Count Down");
lbCountDown.setBounds(270, 178, 91, 14);
this.contentPane.add(lbCountDown);

JPanel panel = new JPanel();
panel.setBorder(new TitledBorder("Warnings"));
panel.setBounds(4, 227, 242, 91);
this.contentPane.add(panel);

JPanel panel_1 = new JPanel();
panel_1.setBorder(new TitledBorder("Alarms"));
panel_1.setBounds(243, 227, 225, 91);
this.contentPane.add(panel_1);

JPanel panel_2 = new JPanel();
panel_2.setBorder(new TitledBorder("Pilot Display"));
panel_2.setBounds(4, 11, 464, 209);
this.contentPane.add(panel_2);

}

public static void main(String[] args) {
    DisplayController frame = new DisplayController();
    frame.setVisible(true);
    int timeUntilLanding = Integer.parseInt(lbUAltitudeCountDown.getText());
    int currentSpeed = Integer.parseInt(lbUSpeed.getText());
    int currentAltitude = Integer.parseInt(lbUAltitude.getText());
}

```

Java - STProject/src/controller/DisplayController.java - Eclipse

```

File Edit Source Refactor Navigate Search Project Run Window Help
Quick Access Java PyDev Debug Team Synchronizing Java EE

Package Explorer JUnit
Finished after 80.602 seconds
Runs: 1/1 Errors: 0 Failures: 0
controller.DisplayControllerTest [Run]
test (80.524 s)

Failure Trace

DisplayController.java DisplayControllerTest.java ComputeSimulationState.java

191     lbUAirResistance.setBounds(378, 59, 63, 14);
192     this.contentPane.add(lbUAirResistance);
193
194     lbUAltitudeLoss = new JLabel("20 ft/sec ");
195     lbUAltitudeLoss.setBounds(378, 116, 63, 14);
196     this.contentPane.add(lbUAltitudeLoss);
197
198     lbLandingGearOverride = new JLabel("Landing Gear Override");
199     lbLandingGearOverride.setBounds(10, 286, 132, 13);
200     this.contentPane.add(lbLandingGearOverride);
201
202     lbBrakingOverride = new JLabel("Braking Override");
203     lbBrakingOverride.setBounds(142, 285, 97, 14);
204     this.contentPane.add(lbBrakingOverride);
205
206     lbGearNotDown = new JLabel("Gear Not Down");
207     lbGearNotDown.setBounds(270, 285, 96, 14);
208     this.contentPane.add(lbGearNotDown);
209
210     lbGearAirSpeed = new JLabel("Gear Air Speed");
211     lbGearAirSpeed.setBounds(366, 285, 89, 14);
212     this.contentPane.add(lbGearAirSpeed);
213
214     JCheckBox cbSilenceAuralAlarm = new JCheckBox("Silence Aural Alarm");
215     cbSilenceAuralAlarm.setBounds(309, 245, 146, 23);
216     this.contentPane.add(cbSilenceAuralAlarm);
217     cbSilenceAuralAlarm.addActionListener(new ActionListener() {
218         @Override
219         public void actionPerformed(ActionEvent e) {
220             AbstractButton abstractButton = (AbstractButton) e.getSource();
221             boolean selected = abstractButton.getModel().isSelected();
222             if(selected)
223             {
224                 setWarningLabels();
225                 if(lbGearNotDown.getForeground() == Color.RED)
226                 {
227                     lbGearNotDown.setForeground(Color.YELLOW);
228                 }
229             }
230         }
231     });
232
233 });
234 });
235 });

JLabel lbCountDown = new JLabel("Count Down");
lbCountDown.setBounds(270, 178, 91, 14);
this.contentPane.add(lbCountDown);

JPanel panel = new JPanel();
panel.setBorder(new TitledBorder("Warnings"));
panel.setBounds(4, 227, 242, 91);
this.contentPane.add(panel);

JPanel panel_1 = new JPanel();
panel_1.setBorder(new TitledBorder("Alarms"));
panel_1.setBounds(243, 227, 225, 91);
this.contentPane.add(panel_1);

JPanel panel_2 = new JPanel();
panel_2.setBorder(new TitledBorder("Pilot Display"));
panel_2.setBounds(4, 11, 464, 209);
this.contentPane.add(panel_2);

}

public static void main(String[] args) {
    DisplayController frame = new DisplayController();
    frame.setVisible(true);
    int timeUntilLanding = Integer.parseInt(lbUAltitudeCountDown.getText());
    int currentSpeed = Integer.parseInt(lbUSpeed.getText());
    int currentAltitude = Integer.parseInt(lbUAltitude.getText());
}

```

Java - STProject/src/controller/DisplayController.java - Eclipse

```

File Edit Source Refactor Navigate Search Project Run Window Help
Quick Access Java PyDev Debug Team Synchronizing Java EE

Package Explorer JUnit
Finished after 80.602 seconds
Runs: 1/1 Errors: 0 Failures: 0
controller.DisplayControllerTest [Run]
  test (80.524 s)

Failure Trace

DisplayController.java DisplayControllerTest.java ComputeSimulationState.java
309             ComputeSimulationState.setSpeed(currentSpeed);
310         }
311         try {
312             Thread.sleep(1000);
313         } catch (InterruptedException e) {
314             e.printStackTrace();
315         }
316     }
317     else
318         System.out.println("Please check the values of initialization of Speed [0:500], Altitude [0:5000] and TimeUntillLanding [0:250] in i");
319     */
320     public void run(){
321         DisplayController frame = new DisplayController();
322         frame.setVisible(true);
323     }
324
325     public void setWarningLabels(){
326         if(ComputeSimulationState.isGearOverrideWarningOn())
327             DisplayController.lbLandingGearOverride.setForeground(Color.RED);
328         else
329             DisplayController.lbLandingGearOverride.setForeground(Color.black);
330         if(ComputeSimulationState.isAirBrakeWarningOn())
331             DisplayController.lbBrakingOverride.setForeground(Color.RED);
332         else
333             DisplayController.lbBrakingOverride.setForeground(Color.black);
334         if(ComputeSimulationState.isGearNotDownAlarmOn())
335             DisplayController.lbGearNotDown.setForeground(Color.RED);
336         else
337             DisplayController.lbGearNotDown.setForeground(Color.black);
338         if(ComputeSimulationState.isGearAirSpeedAlarmOn())
339             DisplayController.lbGearAirSpeed.setForeground(Color.RED);
340         else
341             DisplayController.lbGearAirSpeed.setForeground(Color.black);
342     }
343 }
344
345 }

```

Writable Smart Insert 73 : 24

Source code of ComputeSimulationState.java with code coverage,

Java - STProject/src/inputs/ComputeSimulationState.java - Eclipse

```

File Edit Source Refactor Navigate Search Project Run Window Help
Quick Access Java PyDev Debug Team Synchronizing Java EE

Package Explorer JUnit
Finished after 80.602 seconds
Runs: 1/1 Errors: 0 Failures: 0
controller.DisplayControllerTest [Run]
  test (80.524 s)

Failure Trace

DisplayController.java DisplayControllerTest.java ComputeSimulationState.java
1 package inputs;
2
3 import java.io.BufferedReader;
4 import java.io.FileNotFoundException;
5 import java.io.FileReader;
6 import java.io.IOException;
7
8 public class ComputeSimulationState {
9
10     private static int timeUntilLanding;
11     private static int speed;
12     private static int speedIncrement;
13     private static int altitude;
14     private static int altitudeIncrement;
15     private static Position gearPosition;
16     private static Position selectedGearPosition;
17     private static String throttleCmd;
18     private static String elevonCmd;
19     private static boolean airBrakeWarningOn;
20     private static boolean gearOverrideWarningOn;
21     private static boolean gearNotDownAlarmOn;
22     private static boolean gearAirSpeedAlarmOn;
23     private static boolean gearUpCommand;
24     private static boolean silenceAlarms;
25     private static int pass=0;
26     private static int fail=0;
27     private static int i=1;
28
29     public static void computeSimulationState(int altitude,int speed,int timeUntilLanding,Position currentGearPosition,
30         String throttleCmd,String elevonCmd){
31         if (throttleCmd == "+")
32             ComputeSimulationState.speed=speed+10;
33         else
34         {
35             if (throttleCmd== "-")
36                 ComputeSimulationState.speed=speed-10;
37             else
38
39         }
40     }
41 }

```

Writable Smart Insert 61 : 33

Java - STProject/src/inputs/ComputeSimulationState.java - Eclipse

```

File Edit Source Refactor Navigate Search Project Run Window Help
Quick Access Java PyDev Debug Team Synchronizing Java EE

Package Explorer JUnit
Finished after 80.602 seconds
Runs: 1/1 Errors: 0 Failures: 0
controller.DisplayControllerTest [Run]
test (80.524 s)

DisplayController.java DisplayControllerTest.java ComputeSimulationState.java
36     ComputeSimulationState.speed=speed-10;
37     else
38         ComputeSimulationState.speed = speed;
39     }
40     if (elevonCmd== "r")
41         ComputeSimulationState.altitude=altitude+20;
42     else
43     {
44         if (elevonCmd =="-")
45             ComputeSimulationState.altitude=altitude-20;
46         else
47             ComputeSimulationState.altitude=altitude;
48     }
49     ComputeSimulationState.airBrakeWarningOn = (ComputeSimulationState.speed >= 250) && (timeUntilLanding < 60);
50     ComputeSimulationState.gearOverrideWarningOn = (currentGearPosition == Position.Down) && (ComputeSimulationState.speed>400);
51     ComputeSimulationState.gearNotDownAlarmOn = (currentGearPosition == Position.Up) && ((timeUntilLanding <=120) || (ComputeSimulationState.speed>300));
52     ComputeSimulationState.gearAirSpeedAlarmOn = (currentGearPosition== Position.Down ) && (ComputeSimulationState.speed>300);
53     if (gearOverrideWarningOn)
54         currentGearPosition = selectedGearPosition;
55 }

56
57 /**
58  * public static void main(String[] args) {
59  *     String csvFile = "1.csv";
60  *     BufferedReader br = null;
61  *     String line = "";
62  *     String cvsSplitBy = ",";
63  *     try {
64  *         br = new BufferedReader(new FileReader(csvFile));
65  *         line = br.readLine();
66  *         while ((line = br.readLine()) != null) {
67  *             // use comma as separator
68  *             String[] inputs = line.split(cvsSplitBy);
69  *
70  *             speed = Integer.parseInt(inputs[0]);
71  *             if(inputs[1].equals("V"))
72  *                 inputs[1] = "Down";
    else
    }
    }
    }
    }
    }
    */
    public static int getTimeUntilLanding() {
        return timeUntilLanding;
    }

    public static void setTimeUntilLanding(int timeUntilLanding) {
        ComputeSimulationState.timeUntilLanding = timeUntilLanding;
    }

    public static int getSpeed() {
        return speed;
    }

    public static void setSpeed(int speed) {
        ComputeSimulationState.speed = speed;
    }

    public static int getSpeedIncrement() {
        return speedIncrement;
    }

    public static void setSpeedIncrement(int speedIncrement) {
        ComputeSimulationState.speedIncrement = speedIncrement;
    }

    public static int getAltitude() {
        return altitude;
    }
}

```

Java - STProject/src/inputs/ComputeSimulationState.java - Eclipse

```

File Edit Source Refactor Navigate Search Project Run Window Help
Quick Access Java PyDev Debug Team Synchronizing Java EE

Package Explorer JUnit
Finished after 80.602 seconds
Runs: 1/1 Errors: 0 Failures: 0
controller.DisplayControllerTest [Run]
test (80.524 s)

DisplayController.java DisplayControllerTest.java ComputeSimulationState.java
106     }
107     System.out.println("Total No. of Test Cases passed: "+pass+" and failed: "+fail+".");
108
109     if (br != null) {
110         try {
111             br.close();
112         } catch (IOException e) {
113             e.printStackTrace();
114         }
115     }
116 }
117 /**
118 * public static int getTimeUntilLanding() {
119     return timeUntilLanding;
120 }
121
122 public static void setTimeUntilLanding(int timeUntilLanding) {
123     ComputeSimulationState.timeUntilLanding = timeUntilLanding;
124 }

125 public static int getSpeed() {
126     return speed;
127 }
128
129 public static void setSpeed(int speed) {
130     ComputeSimulationState.speed = speed;
131 }
132
133 public static int getSpeedIncrement() {
134     return speedIncrement;
135 }
136
137 public static void setSpeedIncrement(int speedIncrement) {
138     ComputeSimulationState.speedIncrement = speedIncrement;
139 }
140
141 public static int getAltitude() {
142     return altitude;
143 }
}

```

Java - STProject/src/inputs/ComputeSimulationState.java - Eclipse

```

File Edit Source Refactor Search Project Run Window Help
Quick Access Java PyDev Debug Team Synchronizing Java EE

Package Explorer JUnit
Finished after 80.602 seconds
Runs: 1/1 Errors: 0 Failures: 0
controller.DisplayControllerTest [Run]
test (80.524 s)

Failure Trace

138 public static void setSpeedIncrement(int speedIncrement) {
139     ComputeSimulationState.speedIncrement = speedIncrement;
140 }
141
142 public static int getAltitude() {
143     return altitude;
144 }
145
146 public static void setAltitude(int altitude) {
147     ComputeSimulationState.altitude = altitude;
148 }
149
150 public static int getAltitudeIncrement() {
151     return altitudeIncrement;
152 }
153
154 public static void setAltitudeIncrement(int altitudeIncrement) {
155     ComputeSimulationState.altitudeIncrement = altitudeIncrement;
156 }
157
158 public static Position getGearPosition() {
159     return gearPosition;
160 }
161
162 public static void setGearPosition(Position gearPosition) {
163     ComputeSimulationState.gearPosition = gearPosition;
164 }
165
166 public static Position getSelectedGearPosition() {
167     return selectedGearPosition;
168 }
169
170 public static void setSelectedGearPosition(Position selectedGearPosition) {
171     ComputeSimulationState.selectedGearPosition = selectedGearPosition;
172 }
173
174 public static String getThrottleCmd() {

```

Java - STProject/src/inputs/ComputeSimulationState.java - Eclipse

```

File Edit Source Refactor Search Project Run Window Help
Quick Access Java PyDev Debug Team Synchronizing Java EE

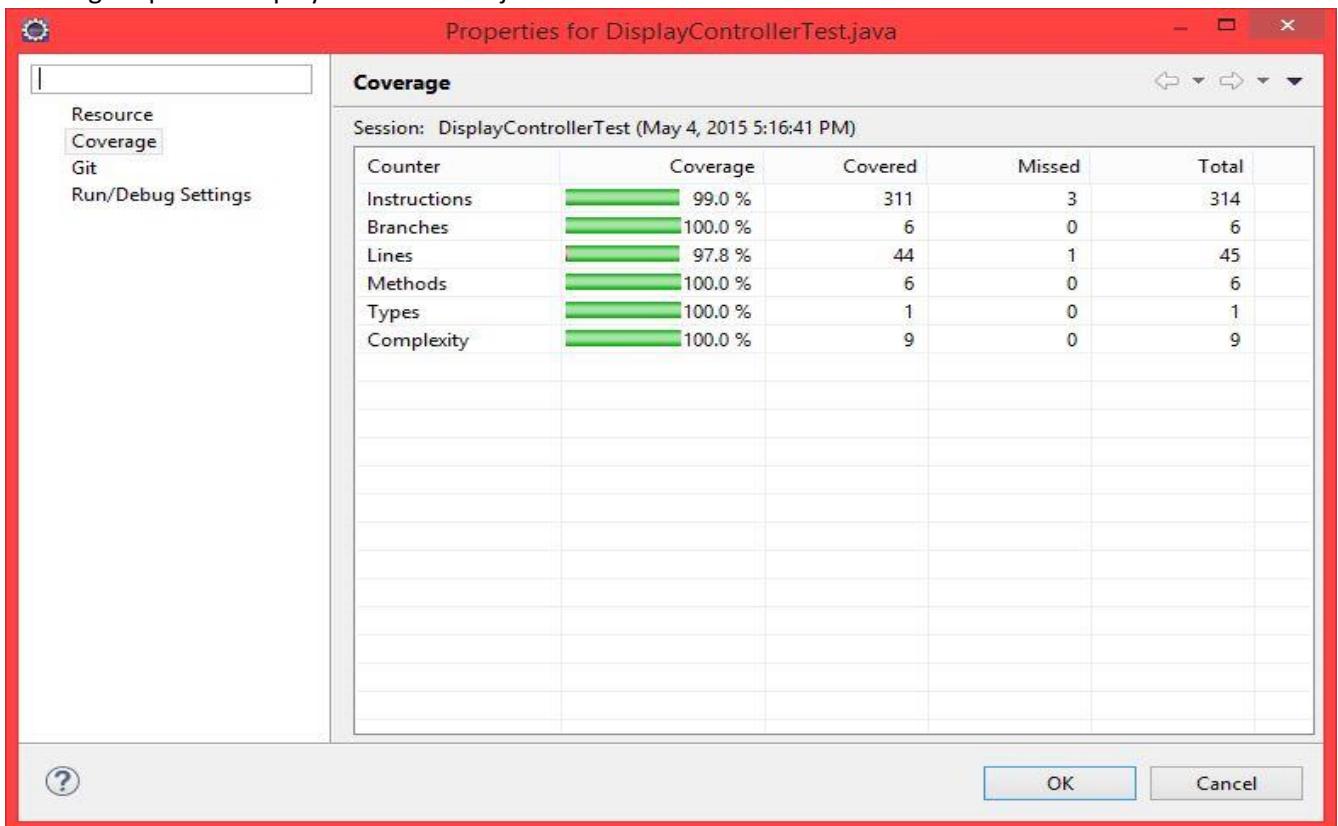
Package Explorer JUnit
Finished after 80.602 seconds
Runs: 1/1 Errors: 0 Failures: 0
controller.DisplayControllerTest [Run]
test (80.524 s)

Failure Trace

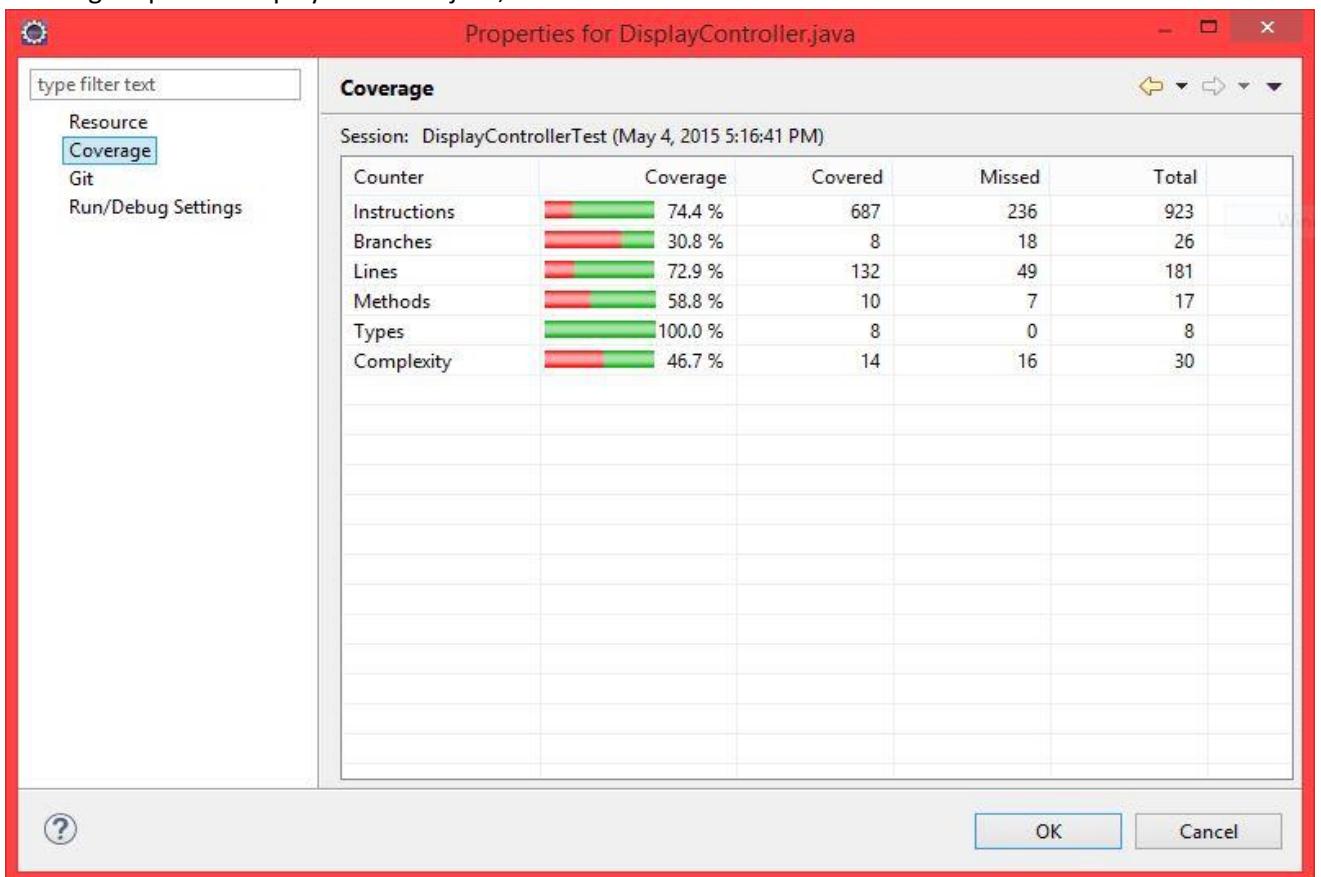
185 public static void setElevonCmd(String elevonCmd) {
186     ComputeSimulationState.elevonCmd = elevonCmd;
187 }
188
189 public static boolean isAirBrakeWarningOn() {
190     return airBrakeWarningOn;
191 }
192
193 public static void setAirBrakeWarningOn(boolean airBrakeWarningOn) {
194     ComputeSimulationState.airBrakeWarningOn = airBrakeWarningOn;
195 }
196
197 public static boolean isGearOverrideWarningOn() {
198     return gearOverrideWarningOn;
199 }
200
201 public static void setGearOverrideWarningOn(boolean gearOverrideWarningOn) {
202     ComputeSimulationState.gearOverrideWarningOn = gearOverrideWarningOn;
203 }
204
205 public static boolean isGearNotDownAlarmOn() {
206     return gearNotDownAlarmOn;
207 }
208
209 public static void setGearNotDownAlarmOn(boolean gearNotDownAlarmOn) {
210     ComputeSimulationState.gearNotDownAlarmOn = gearNotDownAlarmOn;
211 }
212
213 public static boolean isGearAirSpeedAlarmOn() {
214     return gearAirSpeedAlarmOn;
215 }
216
217 public static void setGearAirSpeedAlarmOn(boolean gearAirSpeedAlarmOn) {
218     ComputeSimulationState.gearAirSpeedAlarmOn = gearAirSpeedAlarmOn;
219 }
220 }
221

```

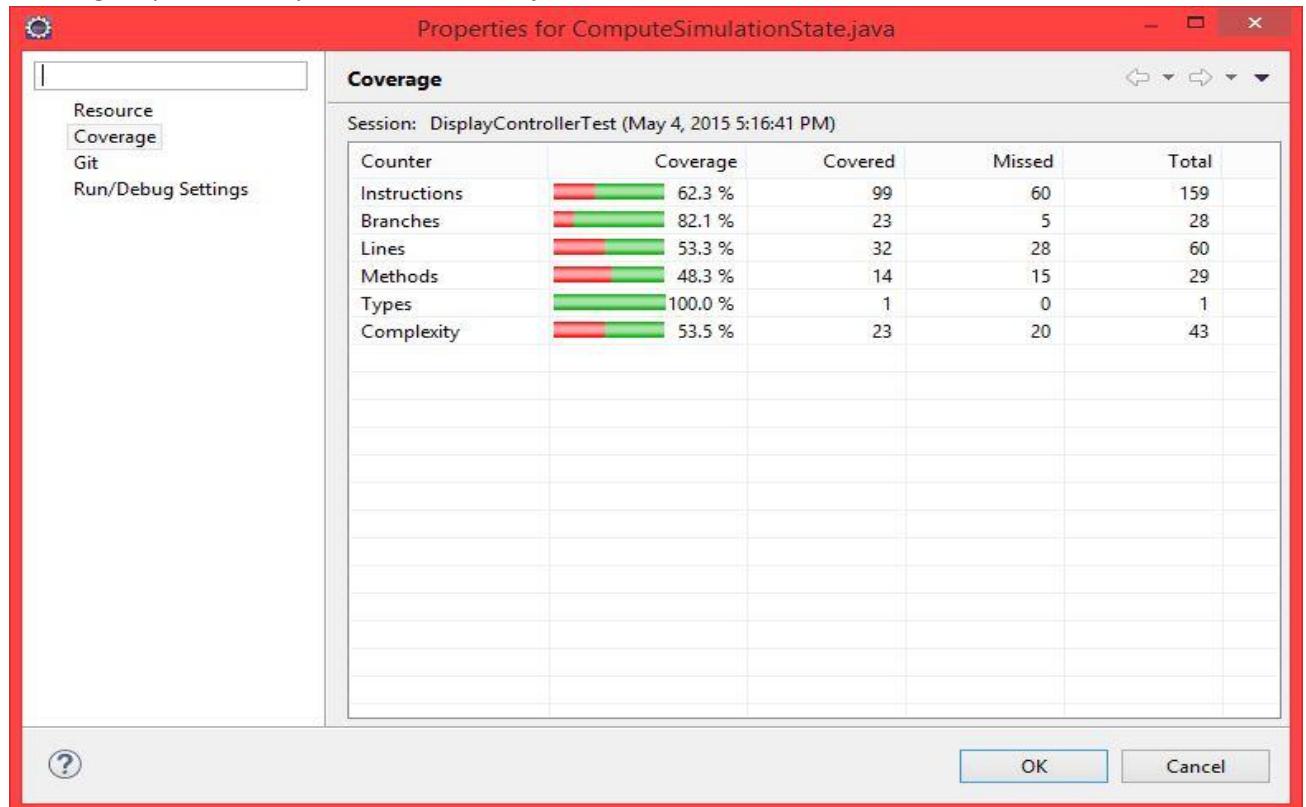
b. Coverage report of DisplayControllerTest.java



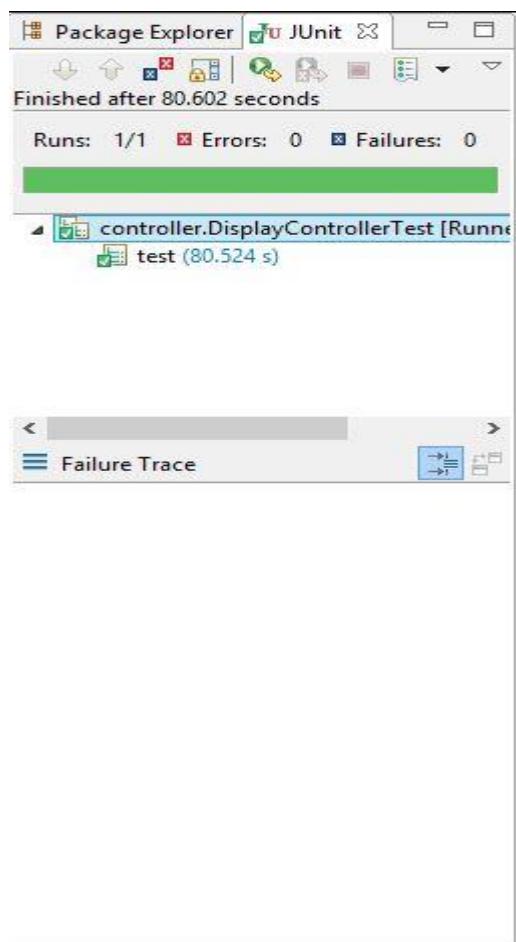
Coverage report of DisplayController.java,



Coverage report of ComputeSimulationState.java,



- c. JUnit test result successful notification of DisplayControllerTest.java.



d. Test Summary

- I. Here, we are testing 8 Display Scenarios comparing Display Scenarios implemented in PART 2 of project to display the scenarios generated by code.

Test case Inputs: Speed, Gear Down, Altitude, Time

Test Case Outputs: Gear Not Down alarm, Gear Air Speed alarm, Air Brake warning, Gear Override warning, Gear-up command.

Display Scenarios Test Oracle Table

Display Scenario	Speed	Gear Down	Altitude	Time	Gear Not Down alarm	Gear Air Speed alarm	Air Brake warning	Gear Override warning	Gear-up command
1	249	N	1500	59	X				
2	401	N	1500	59	X		X		
3	401	Y	1500	59		X	X	X	X
4	401	Y	1500	121		X		X	X
5	301	Y	1500	59		X	X		
6	301	Y	1500	121		X			
7	250	Y	1500	59			X		
8	415	N	1025	130					

II. Test Results

PART 2 Outputs

Speed	249 mph	+	Air Resistance	5 mph/sec
Altitude	1500 ft	-	Altitude Loss	20 ft/sec
Landing Gear	Up	△	Landing time	59 sec
	Down		Count down	
Warnings				
Landing Gear Override	Braking Override	Alarms Aural Alarm Switch OFF		
		Gear Not Down	Gear Air Speed	

Program Output

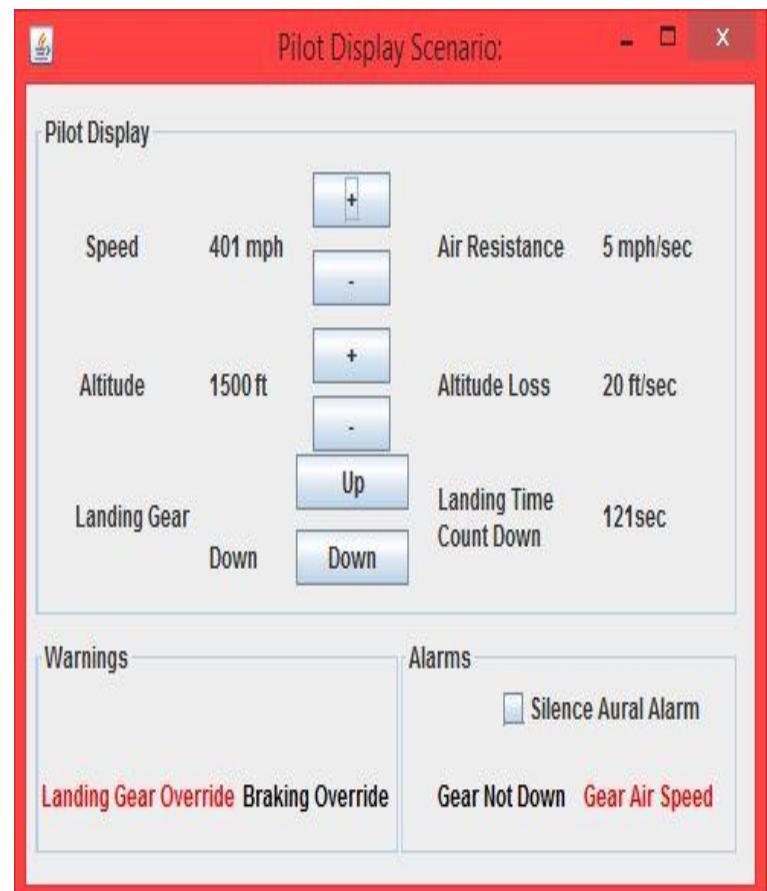
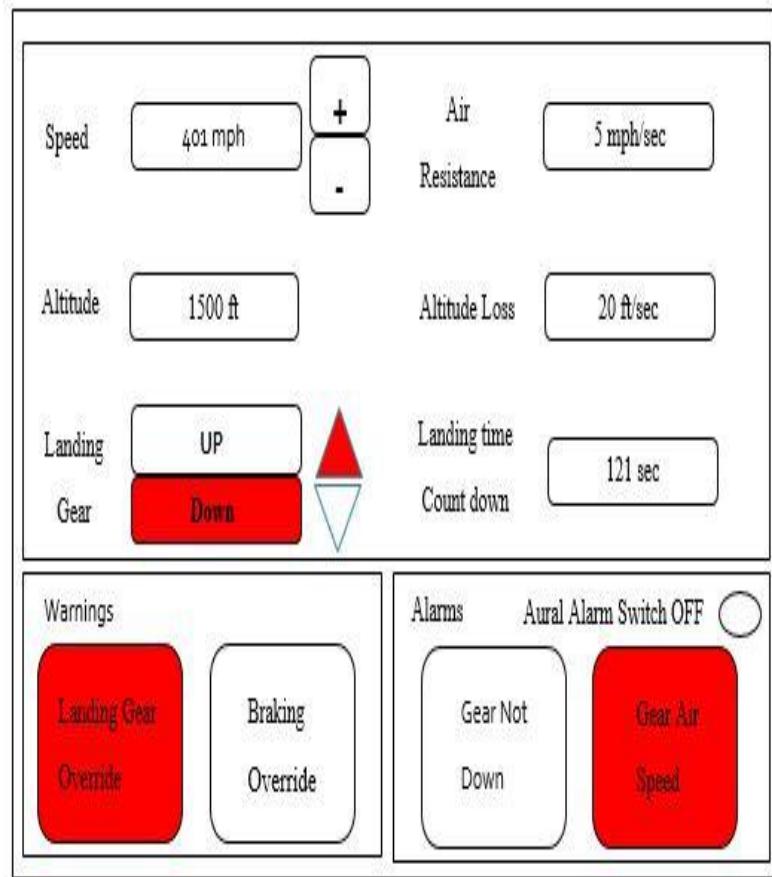
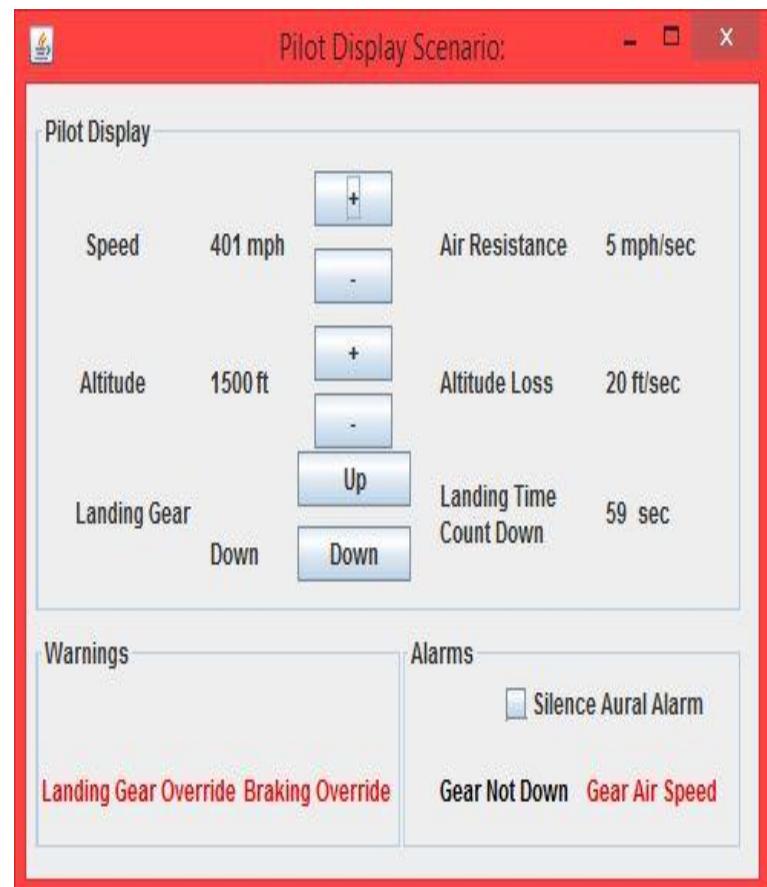
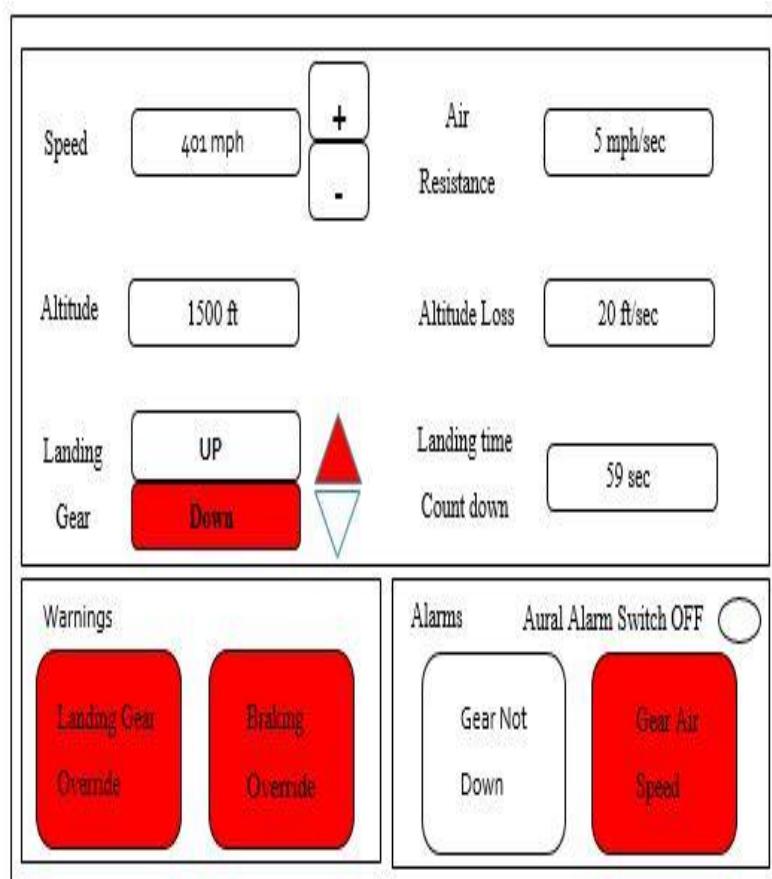
Pilot Display Scenario:

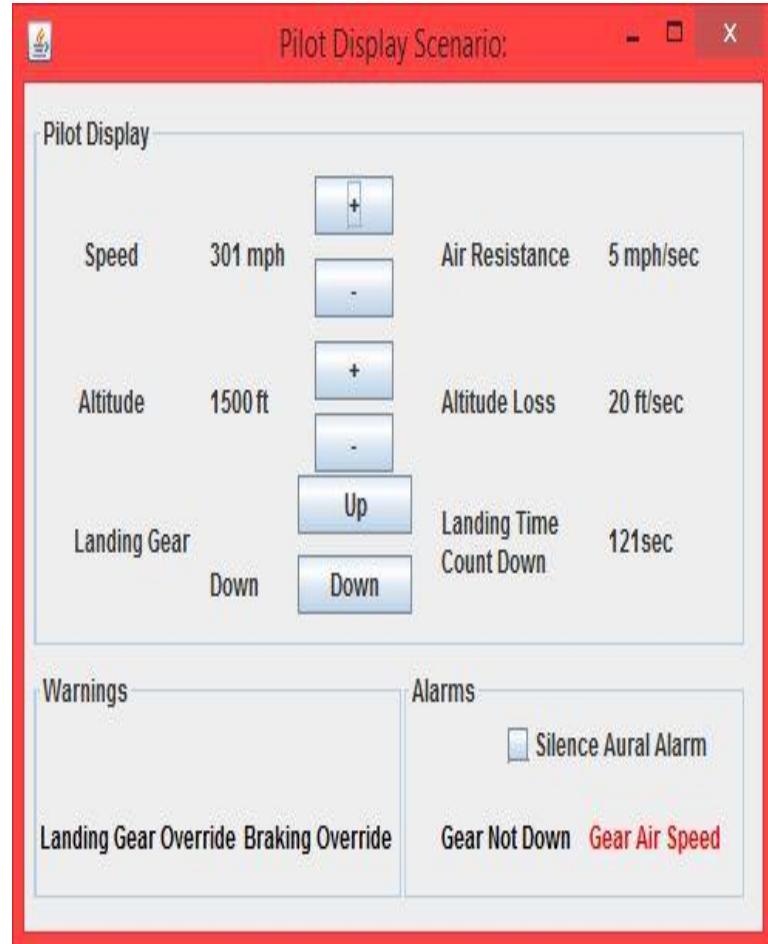
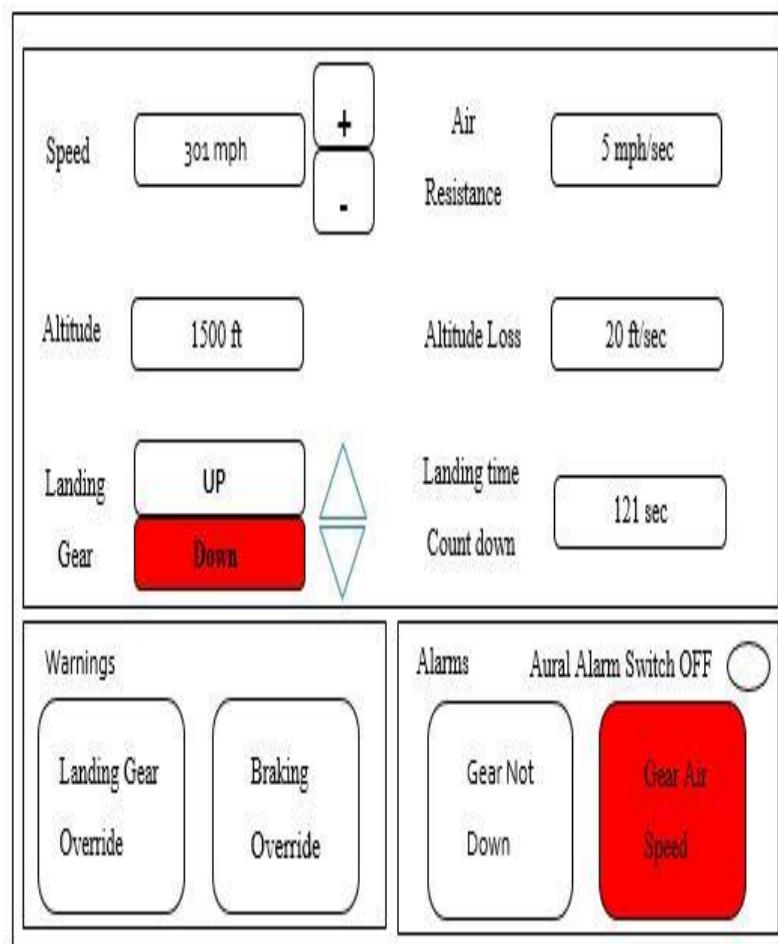
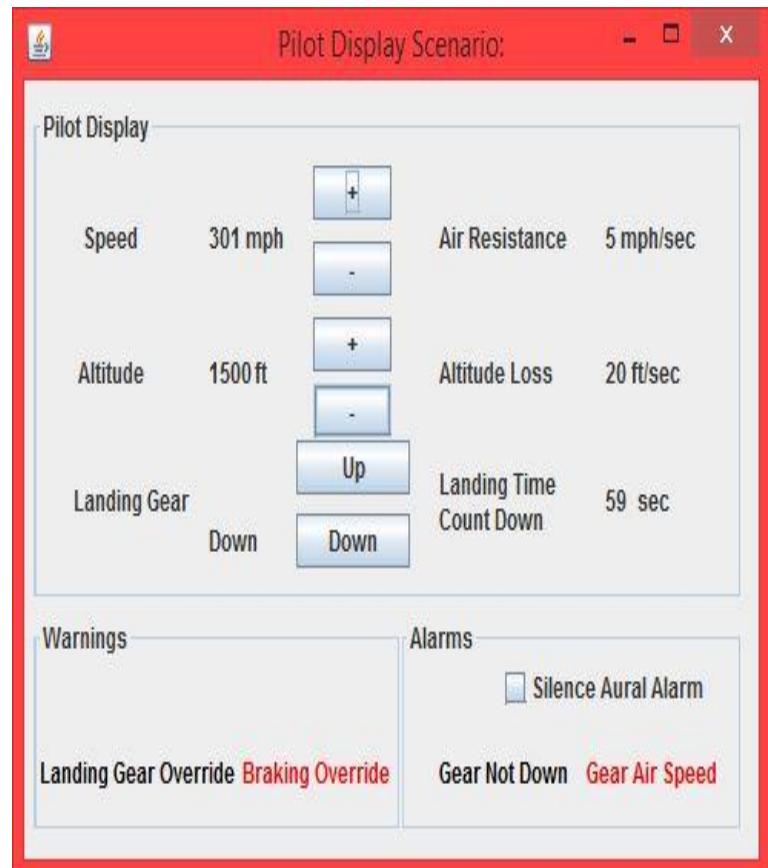
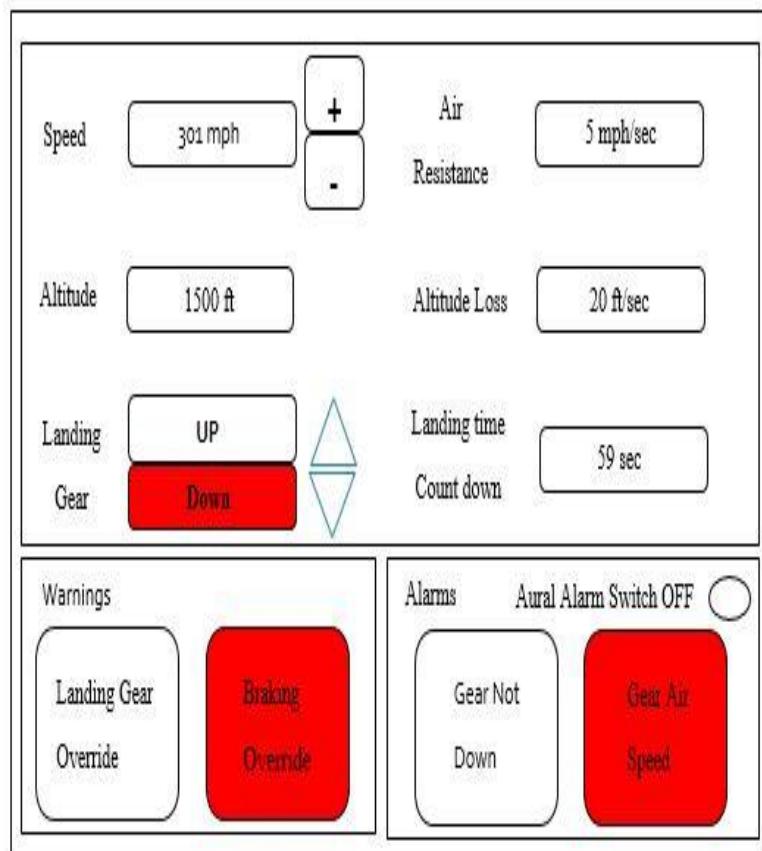
Pilot Display			
Speed	249 mph	Air Resistance	5 mph/sec
Altitude	1500 ft	Altitude Loss	20 ft/sec
Landing Gear	Up	Up	Landing Time Count Down
	Down	Down	59 sec
Warnings			
Alarms		<input type="checkbox"/> Silence Aural Alarm	
Landing Gear Override Braking Override		Gear Not Down Gear Air Speed	

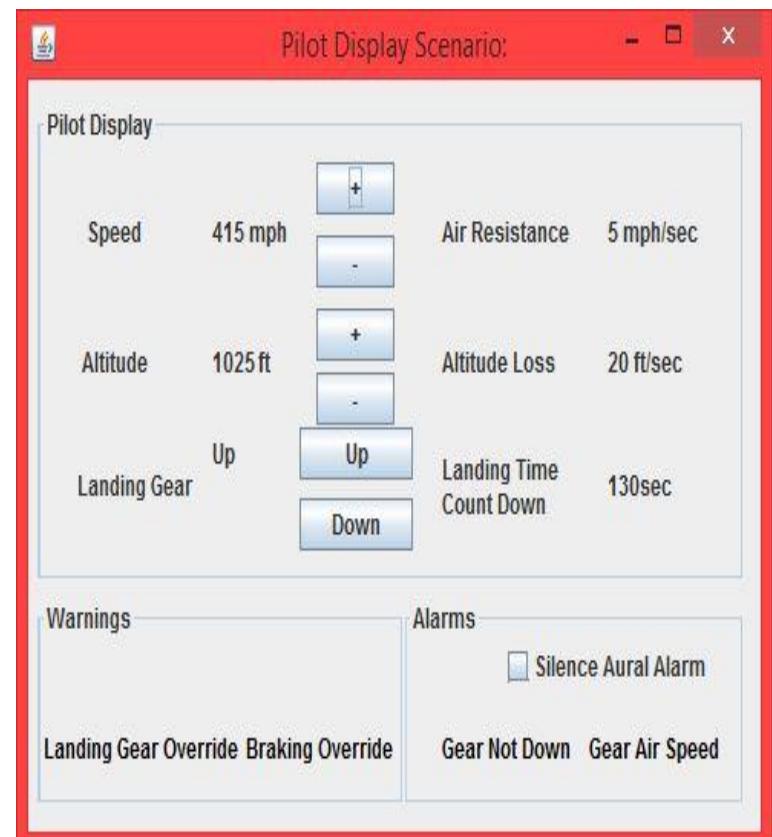
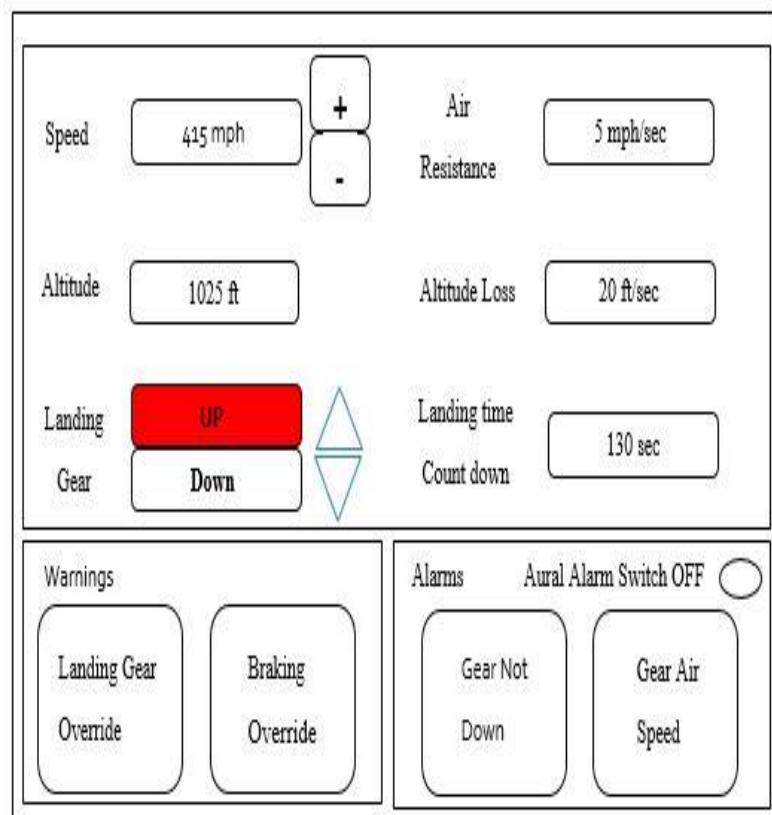
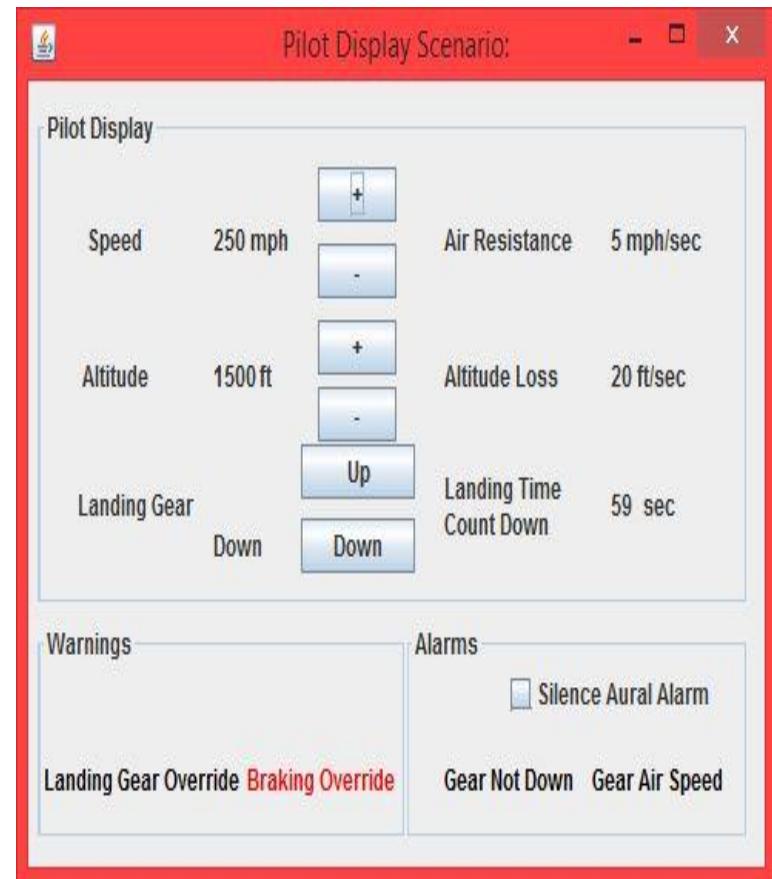
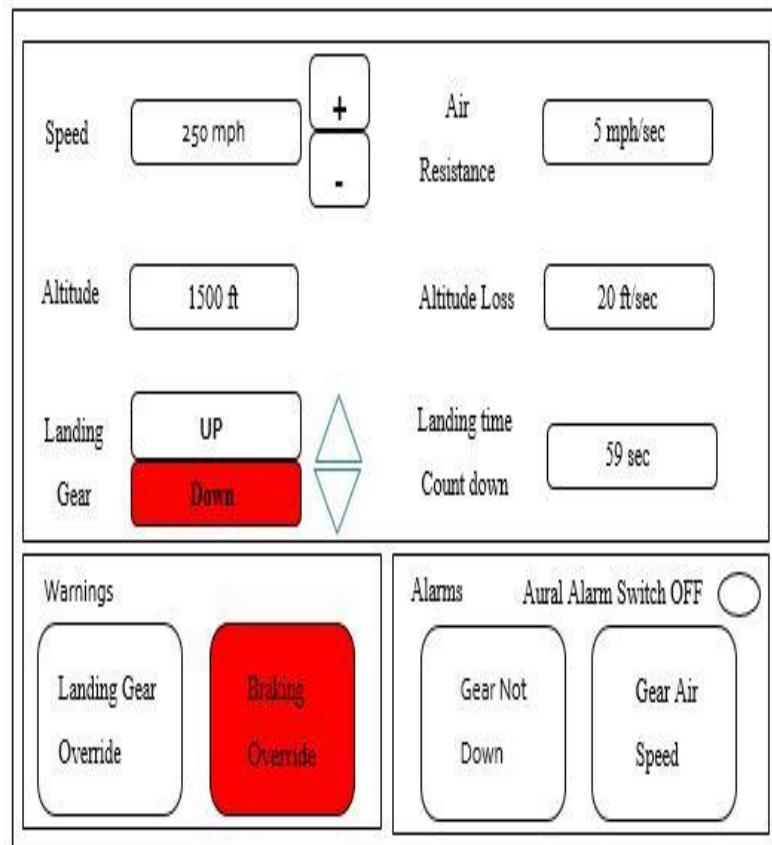
Speed	401 mph	+	Air Resistance	5 mph/sec
Altitude	1500 ft	-	Altitude Loss	20 ft/sec
Landing Gear	Up	△	Landing time	59 sec
	Down		Count down	
Warnings				
Landing Gear Override	Braking Override	Alarms Aural Alarm Switch OFF		
		Gear Not Down	Gear Air Speed	

Pilot Display Scenario:

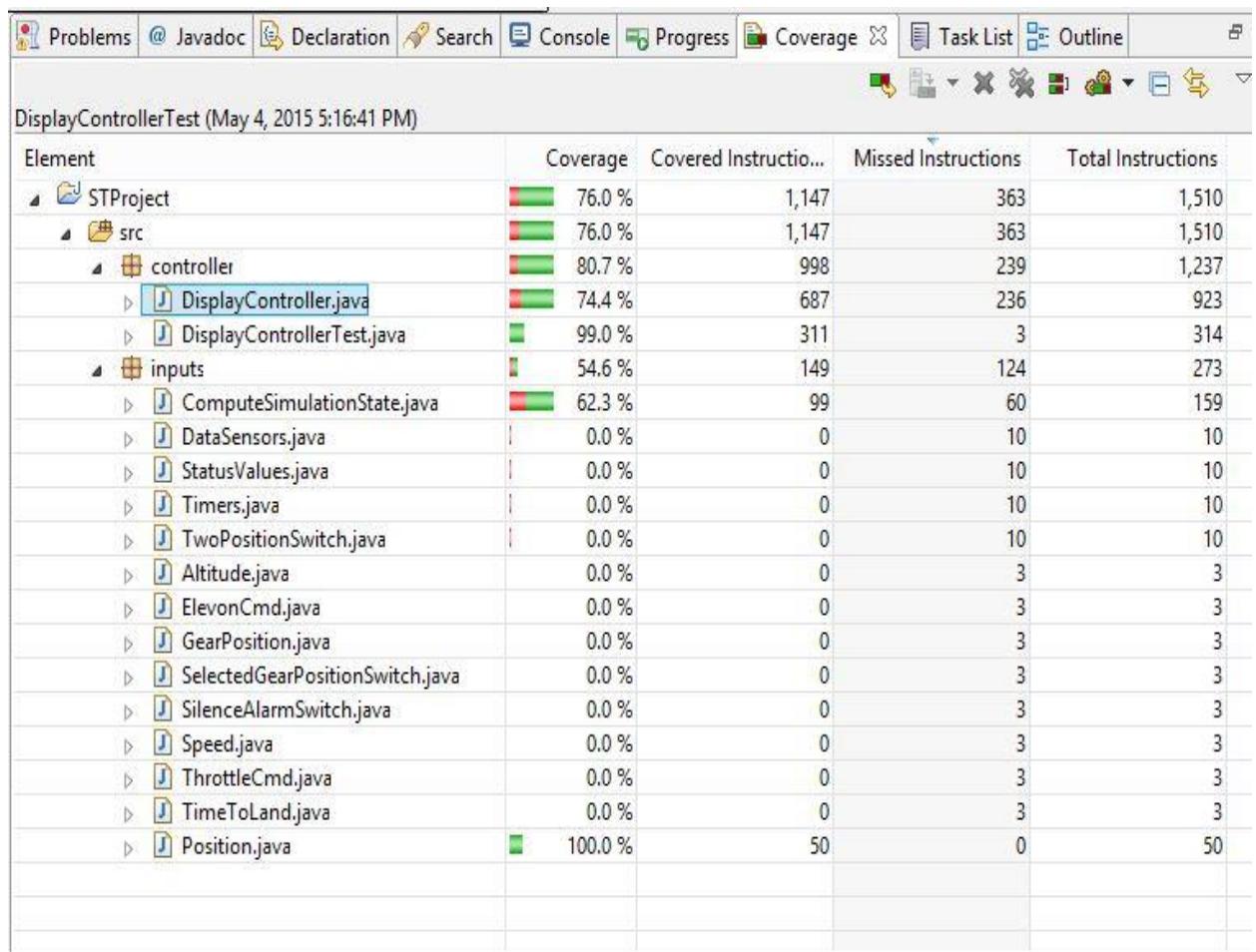
Pilot Display			
Speed	401 mph	Air Resistance	5 mph/sec
Altitude	1500 ft	Altitude Loss	20 ft/sec
Landing Gear	Up	Up	Landing Time Count Down
	Down	Down	59 sec
Warnings			
Alarms		<input type="checkbox"/> Silence Aural Alarm	
Landing Gear Override Braking Override		Gear Not Down Gear Air Speed	







- III. In our code, we are running all 8 display scenarios with a time lag (Thread.sleep) of 10sec each in a single run. We have all display scenarios matched with Test oracle. 100%Pass and 0% failed.
- IV. We have 74.4% overall coverage, with 25.6% shortfall, because in DisplayController class there is code related to UI button click actions that aren't performed while simulating the 8 display scenarios. Hence that part of the code won't be covered. The same applies for the code in the ComputeSimulationState class as mentioned earlier above and in part-1.



- V. In Scenarios 3 and 4 the 'Landing Gear Override warning' is turned on which results in automatically raising the landing gear position. The scenario is implemented in the simulation, but this change in the gear position from down to up isn't shown in display scenarios above as it is a snapshot of the scenario when the landing gear override warning is displayed. To view the same, we can run the simulation and modify the values in the UI to simulate the condition.
- VI. We have added an additional functionality of controlling the altitude of the plane. The pilot can increase/decrease the altitude by pressing the '+' and '-' buttons beside the Altitude label in the UI. This is an additional feature, hence we haven't included the same in the snapshot of the slides as that doesn't have any impact on the outcome of the display scenarios.

3. Other items of discussion:

- a. We have implemented full working landing simulation of aircraft which displays the altitude, speed, time until landing as well as different alarms/warnings, and performs actions, if any are needed. It also checks for valid landing conditions i.e. TimeUntilLanding = 0 and speed =0 and altitude = 0 and displays 'Landed' else displays 'Failed' on the UI.
- b. We had some initial difficulties with understanding the overall structure and flow while beginning the development. Also it involved an initial learning curve for developing UI using JFrames.
- c. N/A
- d. The project would be useful for the future classes too as it has helped gain practical experience w.r.t. JUnit and Jacoco that are widely used in industry. A suggestion for the next project would be building a simulator for a driverless car. That could have various challenges and testing requirements.