University of Southampton

COMP2211: Software Engineering Group Project

Deliverable 2:
Increment 1

Group 30: Benjamin Lellouch (brl1u18), Daniel Best (db5n17),
Deniz Matur (dm1n17), Iman Dzulhadi (ids1g18) Rajpal Dhillon (rsd1n18)

September 28, 2020

Contents

1 Design

1.1 Design Artifacts

1.1.1 UML

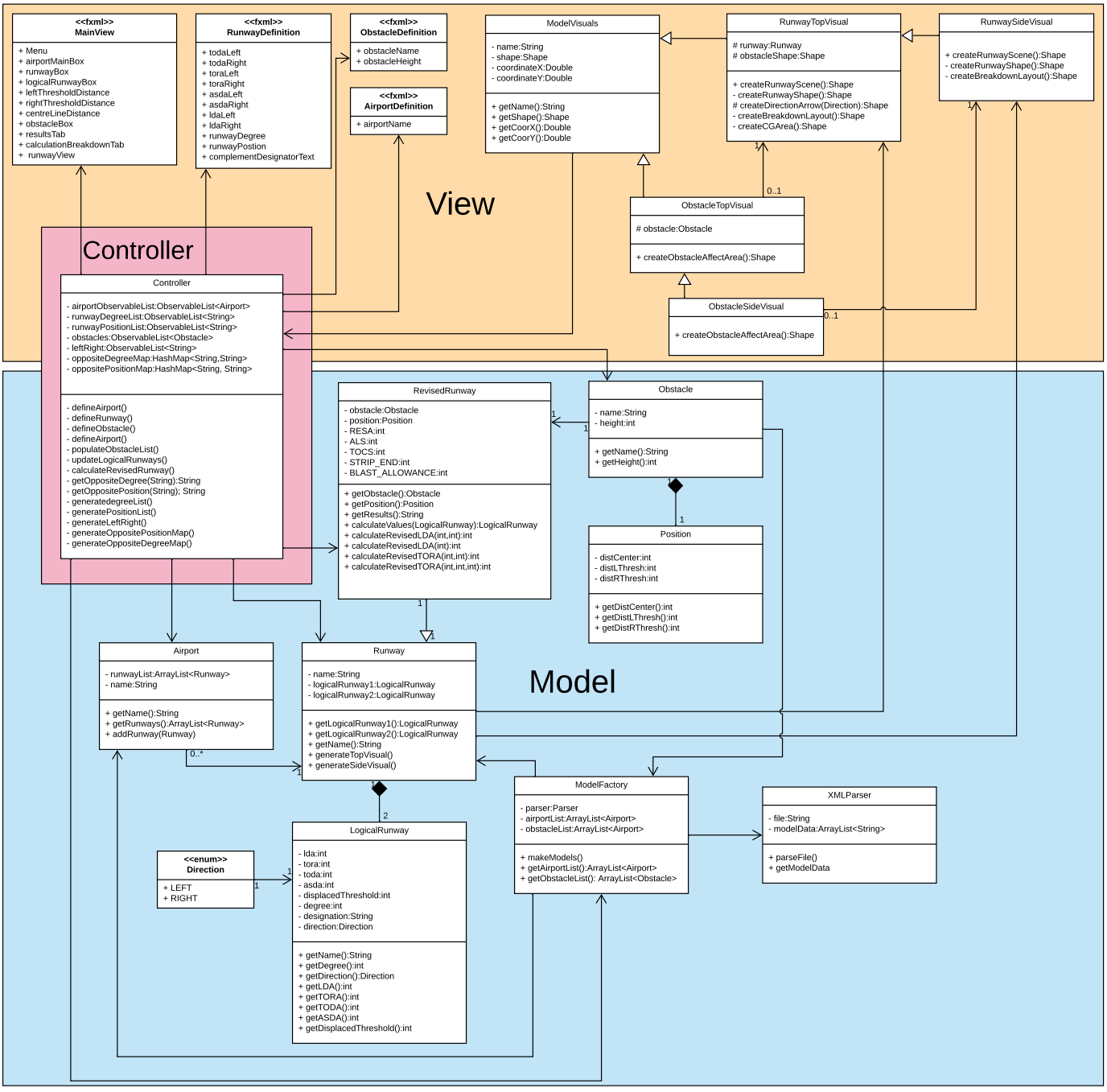


Figure 1: Class Diagram of System

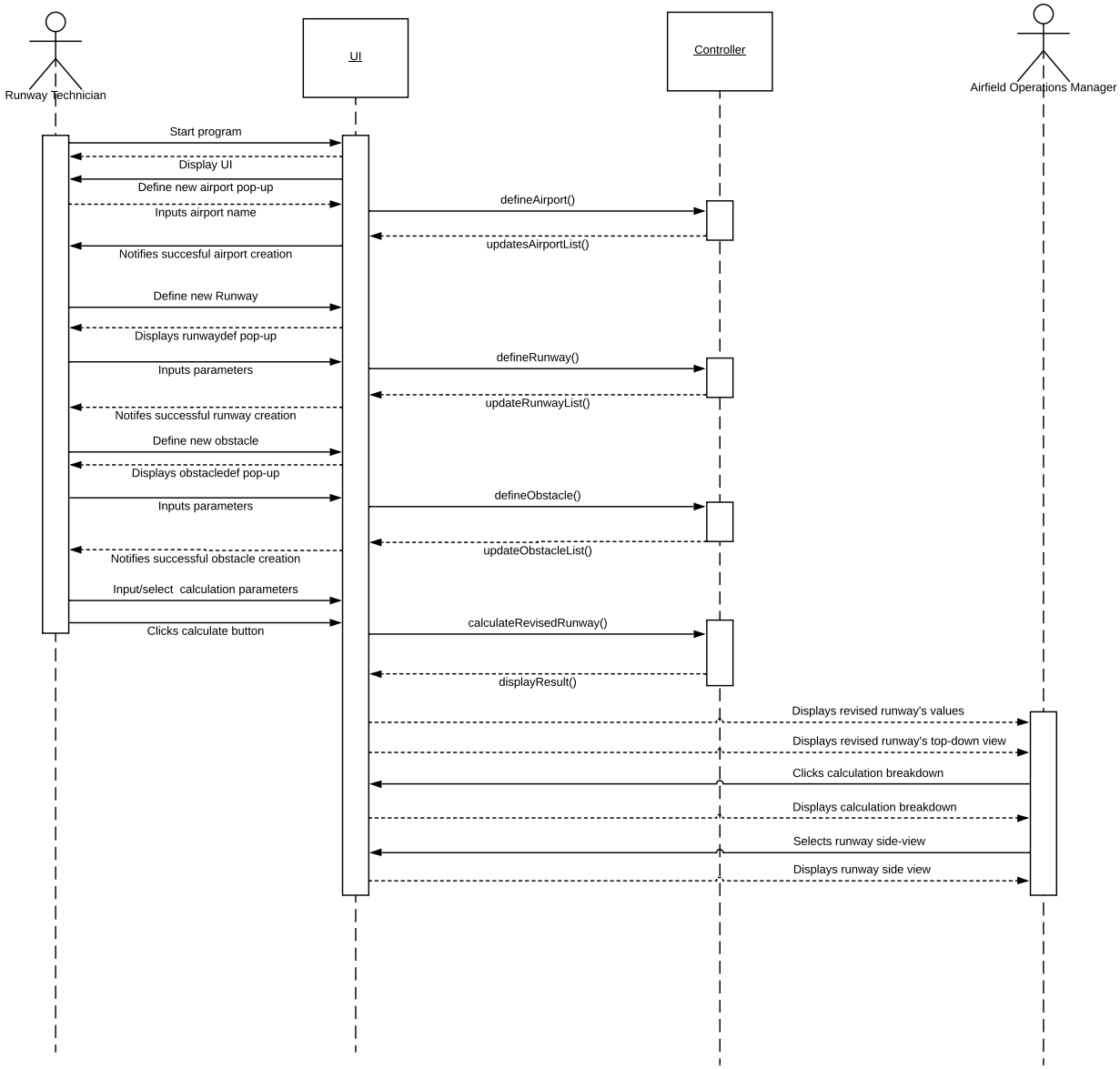
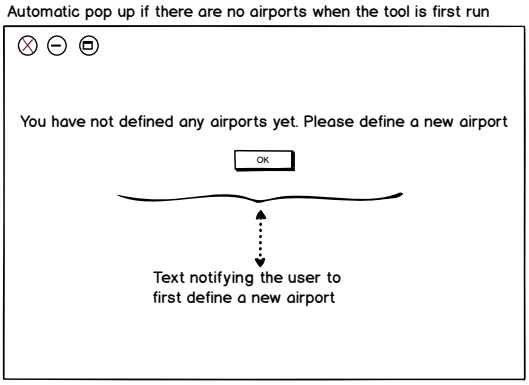
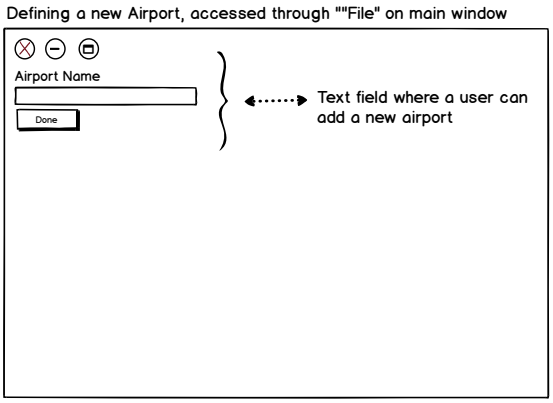


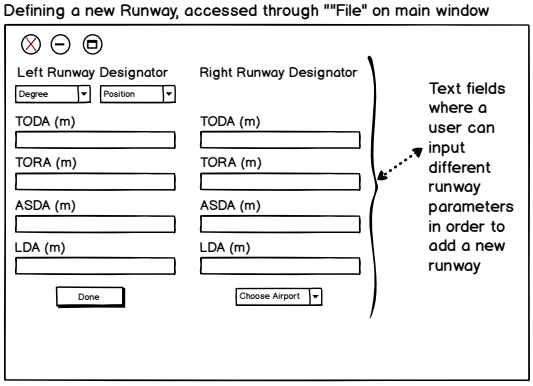
Figure 2: Sequence Diagram

1.1.2 Storyboards

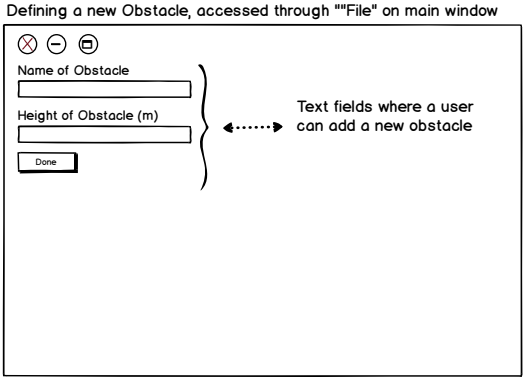




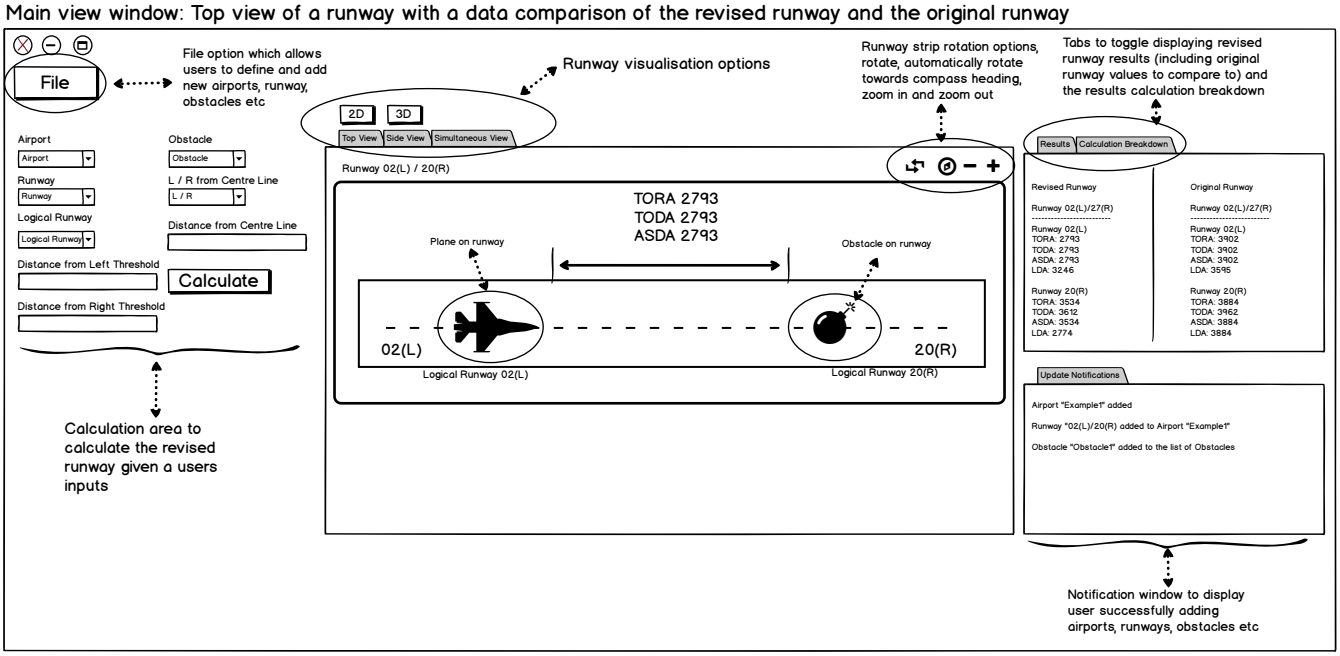
Defining a new Runway, accessed through ""File" on main window



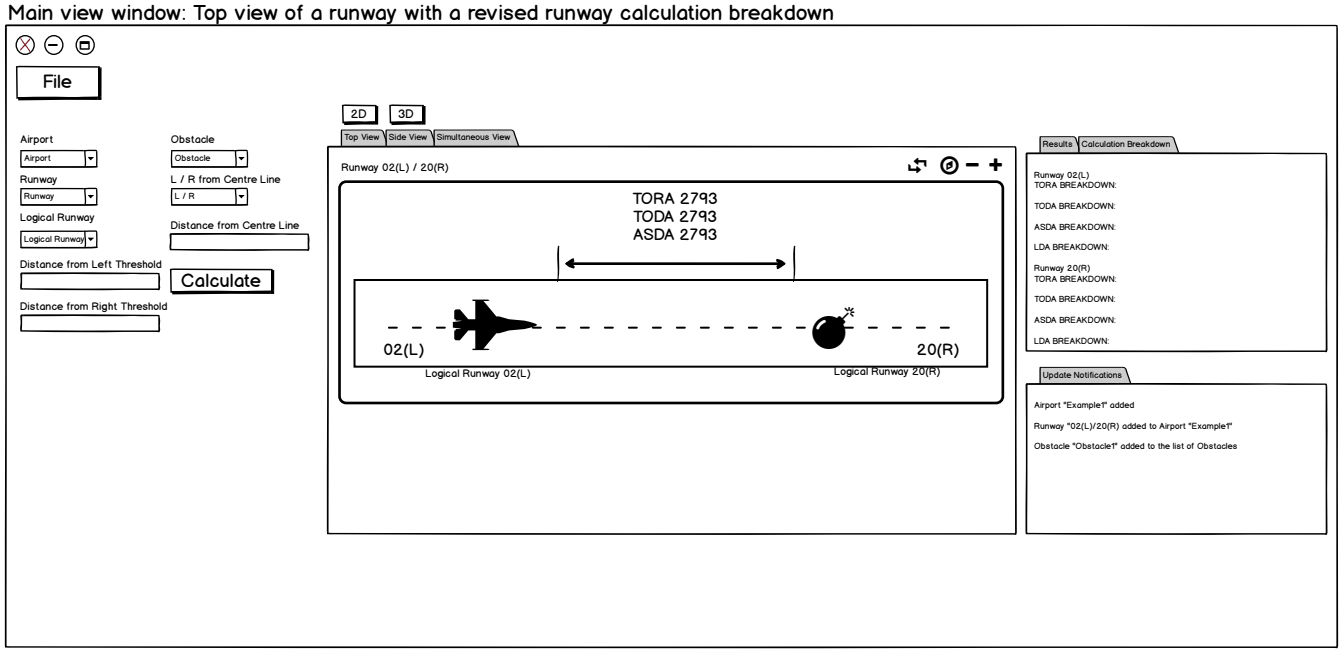
Defining a new Obstacle, accessed through ""File" on main window



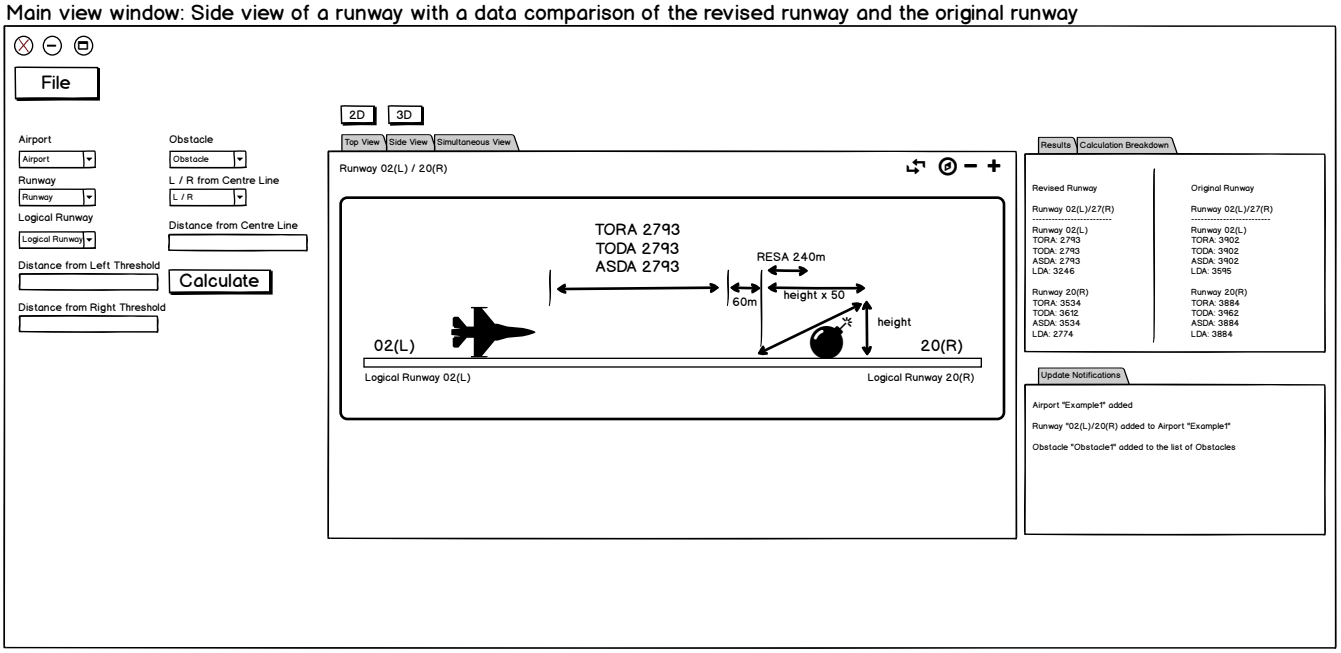
Main view window: Top view of a runway with a data comparison of the revised runway and the original runway



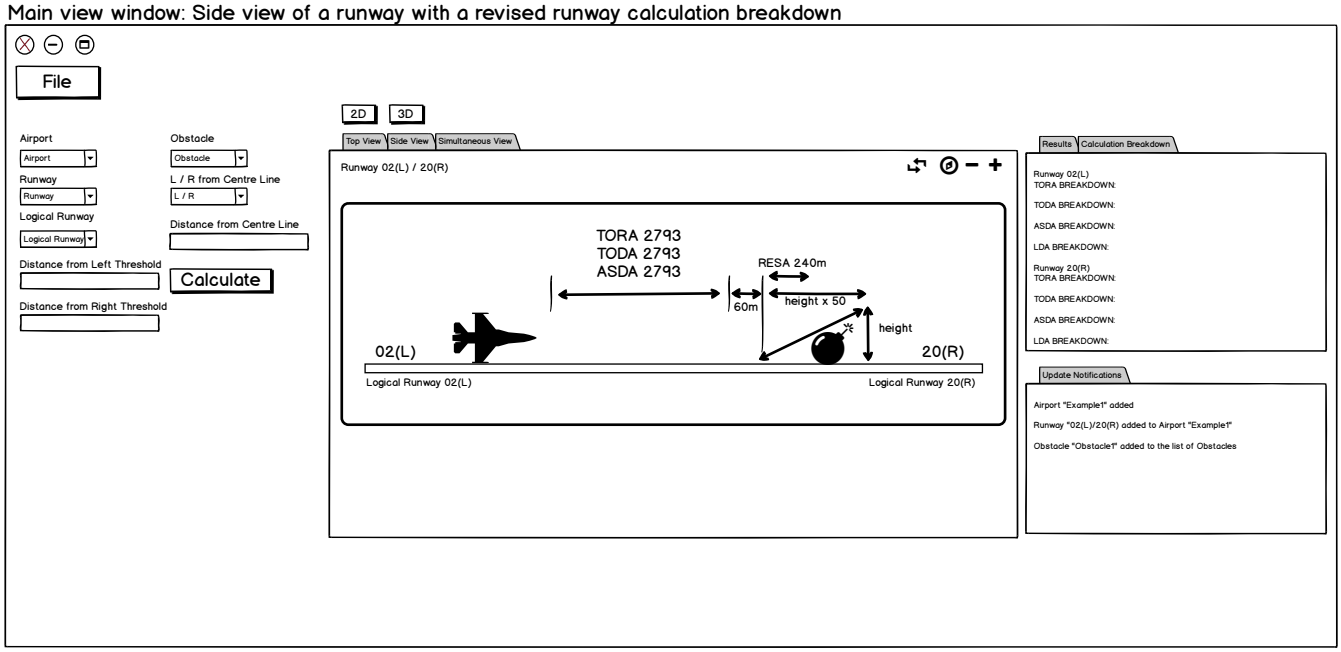
Main view window: Top view of a runway with a revised runway calculation breakdown



Main view window: Side view of a runway with a data comparison of the revised runway and the original runway



Main view window: Side view of a runway with a revised runway calculation breakdown



1.1.3 Scenarios

Scenario 1: Lauren (Runway Technician)

1. Lauren opens the runway redeclaration software, and an alert message appears informing her that she hasn't defined an airport yet.
2. She presses "OK" and a "Define New Airport" dialog appears.
3. Lauren attempts to close this dialog, but is unable to.
4. She then inserts the name of the Airport she works at into the text field, and presses "Done".
5. The main calculation dialog is displayed.
6. Lauren selects the option from the Airport combo box that she just input into the Define Airport dialog.
7. She navigates to "File -> Define New Runway" where a new dialog appears. Lauren selects a designator by defining the degree and position.
8. Lauren inputs values into all of the TODA/TORA/ASDA/LDA values, but accidentally puts the value of "5000;" and "-7500" into two of those fields, selects the appropriate airport, and presses "Done".
9. An appropriate error message appears informing Lauren of the problematic fields, and she is able to modify these values to "5000" and "7500" as originally intended.
10. Lauren then begins the process to define a new obstacle by navigating to "File -> Define New Obstacle", prompting a new pop-up to appear.
11. She inputs the value of "Jumbo Jet" into the obstacle name field, but the value of "tall" into the high field, and presses "Done".
12. An appropriate error message appears that informs Lauren of the mistake she has made, and she is able to modify the value to an appropriate numerical value.
13. Lauren is then able to select this new Runway (and one of the two Logical Runways) and the new Obstacle from the appropriate combo boxes, which she can then pass on to her manager, Charles.

Scenario 2: Charles (Airfield Operations Manager)

1. Charles takes control of the application from where Lauren left off, and is presented with the main menu in the state where the Airport, Runway, Logical Runway and Obstacle have been defined and selected.
2. Charles inputs values into the three "Distance from..." data fields, with at least one of them being negative.
3. He presses the calculate button, and the results panes are populated with the old runway values and the revised runway values.
4. The side view runway visualisation is also populated, and Charles looks at it to assess the impact of the obstacle based on his input values.
5. He then takes a cursory look at the top-down view runway visualisation by pressing the tab to switch to that perspective; the angle of the runway in the visualisation depends on the designation of the runway.
6. Charles then clicks the "Calculation Breakdown" and assesses the calculations made to ensure that they are correct.
7. After having looked at these values, he decides to analyse the visualisation further by switching to the simultaneous view in order to compare the two perspectives alongside one another to make a more informed decision as to how to proceed.

Scenario 3: Aaron (Runway Technician)

1. Aaron opens the runway redeclaration application, and an alert message appears informing him that he hasn't yet defined an airport.
2. He presses "OK" and a "Define New Airport" dialog appears.
3. He then inserts the name of the Airport he works at into the text field, and presses "Done".
4. The main calculation dialog is displayed.
5. Aaron imports a list of Airport and Runways via the XML import dialog.
6. Aaron selects one of the predefined obstacles without the need to explicitly create them.
7. Having completed his part in the process, Aaron passes off the application to the AOM on duty - James.

Scenario 4: James (Airfield Operations Manager)

1. James takes control of the application from where Aaron left off, and is presented with the main menu in the state where the Airport and Obstacle have been defined and selected.
2. Not realising that the runway has already been created since Aaron forgot to select it, he repeats the process of defining a runway with the same designation via the Define Runway dialog.
3. An appropriate error message appears informing James that he has attempted to create a runway with an identical designator, returning him to the Define Runway input form.
4. Realising what has happened, James closes the dialog and selects the runway (and one of the logical runways) that Aaron initially added.
5. James inputs values into the three "Distance from..." data fields, with at least one of them being negative.
6. He presses the calculate button, and the results panes are populated with the old runway values and the revised runway values.
7. The side view runway visualisation is also populated, and James looks at it to assess the impact of the obstacle based on his input values.
8. James then clicks the "Calculation Breakdown" and assesses the calculations made to ensure that they are correct.

1.2 Architecture

Our main architectural artifact is based on the MVC (Model View Controller) paradigm which enables simultaneous development, so that some of the team members can focus on the back-end whilst others develop the middle-tier and front-end. Also, in MVC, one can implement multiple views for a model. For example the side-view and bird's-eye view of the runway are just different visualisations of the same model. We also decided to used constructor injection from our RevisedRunway class in order to increase code re-usability notably when it comes to unit testing.

2 Testing

2.1 Unit testing

We decided to use unit testing for the calculations in order to make sure that they work as intended. We tested our revised runway calculator against the scenarios provided in the Example calculations at Heathrow airport (`https://secure.ecs.soton.ac.uk/noteswiki/images/Heathrow_Scenarios_New.pdf`). We used JUnit in order to achieve this.

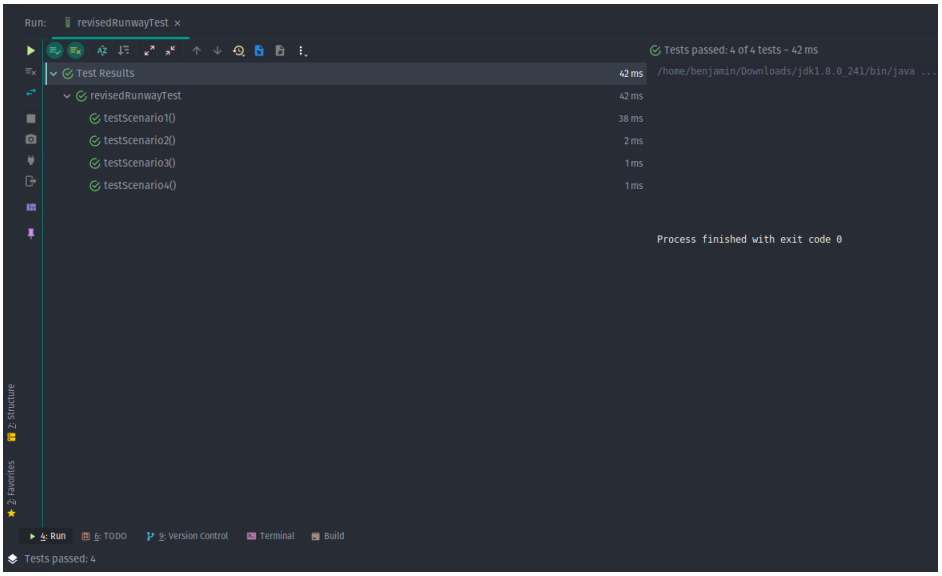


Figure 3: Revised runway calculation tests

2.2 Scenario Testing

We used a testing framework called TestFx to implement the scenarios into test cases. Once the test is executed, the framework will launch the application and take control of the mouse pointer and keyboard and will cycle through the different scenarios. The framework will then check for the expected output in specific part of the UI, a test passes when the expected output is equal to the actual output otherwise it fails. We also ran the scenario tests manually so that we could detect abnormal behaviour that the framework wouldn't be able to detect. We used these methods to ensure the application provided the behaviour that was expected from our scenarios, and also so that we could actually see the application working rather than just testing the underlying functionality.

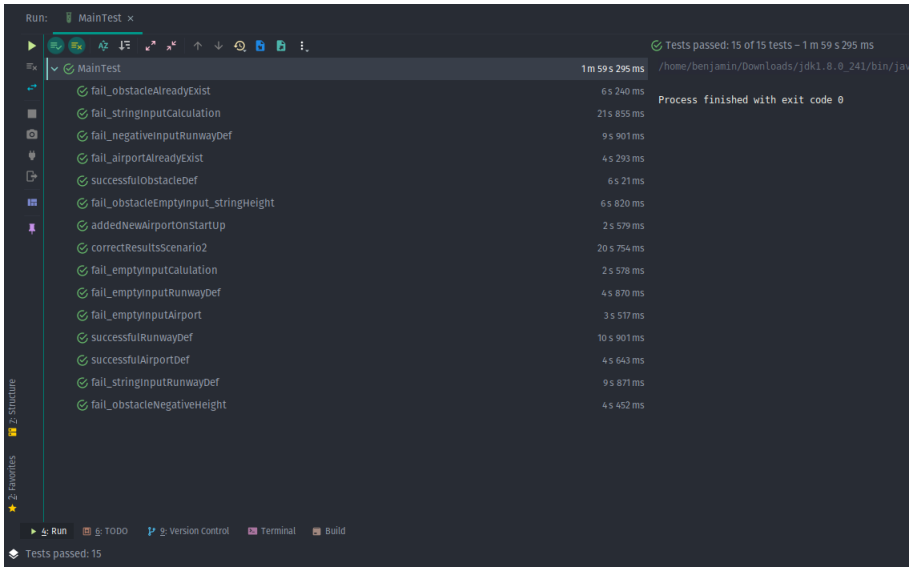
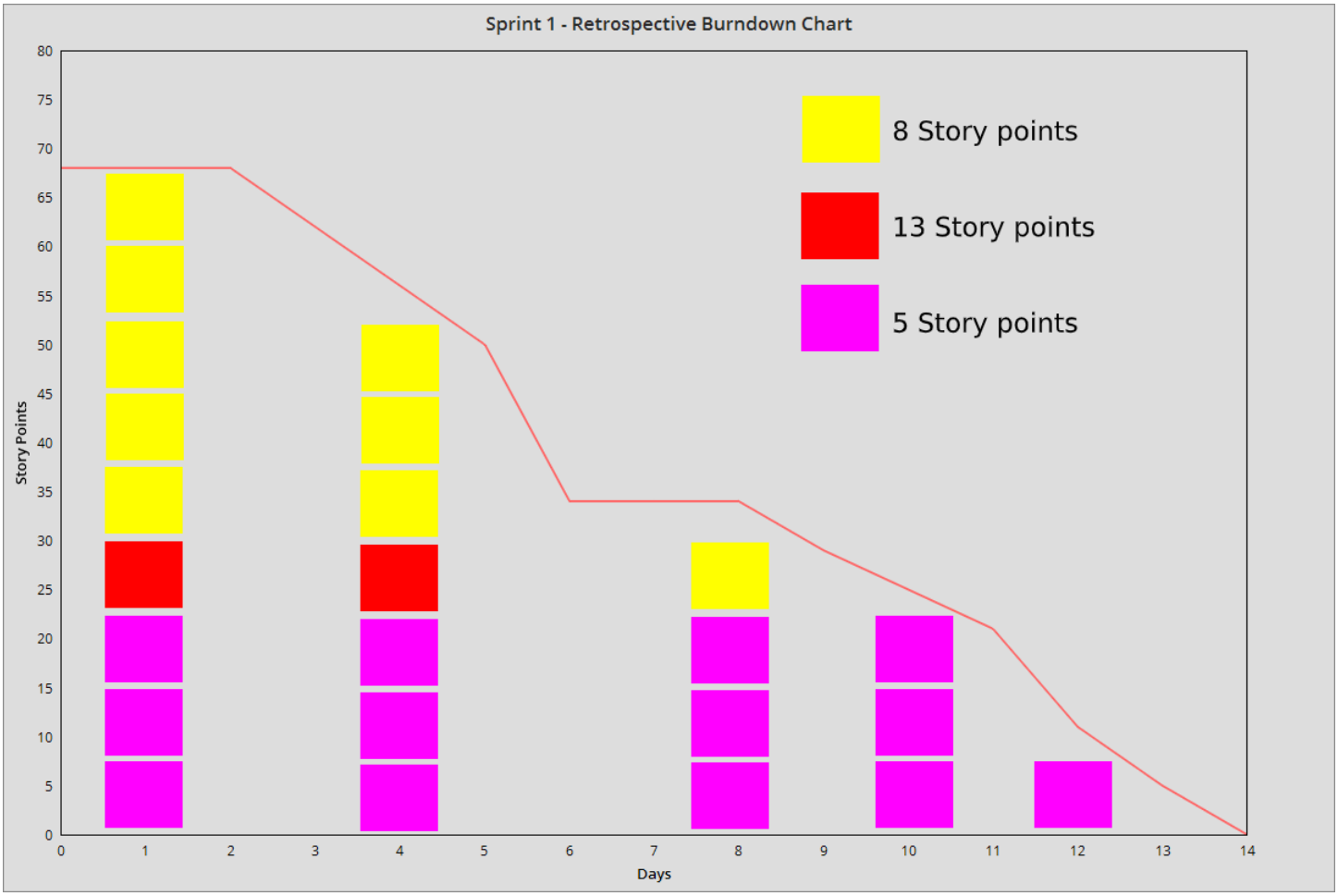


Figure 4: Scenario tests

3 Planning

3.1 Sprint 1: Review

Progress of our sprint went according to plan, we were able to deliver all of the agreed user stories within the allotted time. Every rectangle in the chart below represents a user story as there were too many tasks to be able to reasonably display on the chart.



3.2 Sprint 2: Plan

3.2.1 Backlog

- Benjamin Lellouch (B)
- Rajpal Dhillon (R)
- Iman Dzulhadi (I)
- Deniz Matur (DM)
- Daniel Best (DB)

The number at the beginning of each task represents the number of story points of that task.

Story: Sideways View [SEG-35]

Tasks:

- (5)(DB) Create back-end which generates the canvas for the UI to display
- (5)(DB) Implement UI which displays the sideway view
- (2)(R) Add interactions between UI and back-end in middle-tier
- (1)(R) Testing & debugging

Total story points:

13

Story: Bird's-eye View [SEG-36]

Tasks:

- (5)(B) Create back-end which generates the canvas for the UI to display
- (5)(B) Implement UI which displays the bird's-eye view
- (2)(R) Add interactions between UI and back-end in middle-tier
- (1)(I) Testing & debugging

Total story points: 13

Story: Simultaneous Views [SEG-50]

Tasks:

- (3)(B) Implement UI which displays both views
- (3)(B) Add interactions between UI and back-end in middle-tier
- (2)(DB) Testing & debugging

Total story points: 8

Story: Runway Revision Notification [SEG-7]

Tasks:

- (2)(DM) Code middle-tier which loads notification UI
- (2)(DM) Implement notification UI
- (1)(DB) Testing & debugging

Total story points: 5

Story: Values Changed Notification [SEG-25]

Tasks:

- (2)(DM) Code middle-tier which loads notification UI
- (2)(DM) Implement notification UI
- (1)(I) Testing & debugging

Total story points: 5

Story: Obstacle Added Notification [SEG-24]

Tasks:

- (2)(DM) Code middle-tier which loads notification UI
- (2)(DM) Implement notification UI
- (1)(DB) Testing & debugging

Total story points: 5

Story: Heading View Rotation [SEG-38]

Tasks:

- (2)(I)Code middle-tier which calculates how the canvas should be rotated
- (2)(I)Implement UI which rotates canvas
- (2)(I)Implement UI element which allows the user to rotate the canvas
- (2)(DB)Testing & debugging

Total story points: 8

Story: Import Airport [SEG-27]

Tasks:

- (3)(B)Create XML parser for airport in back-end
- (2)(R)Add XML parser instance to middle-tier
- (2)(R)Implement UI which allows to select XML file to import
- (1)(B)Testing & debugging

Total story points:

8

Story: Import Obstacle [SEG-28]

Tasks:

- (3)(B)Create XML parser for obstacle in back-end
- (2)(R)Add XML parser instance to middle-tier
- (2)(R)Implement UI which allows to select XML file to import
- (1)(DB)Testing & debugging

Total story points:

8

Story: Clear and Graded Area [SEG-51]

Tasks:

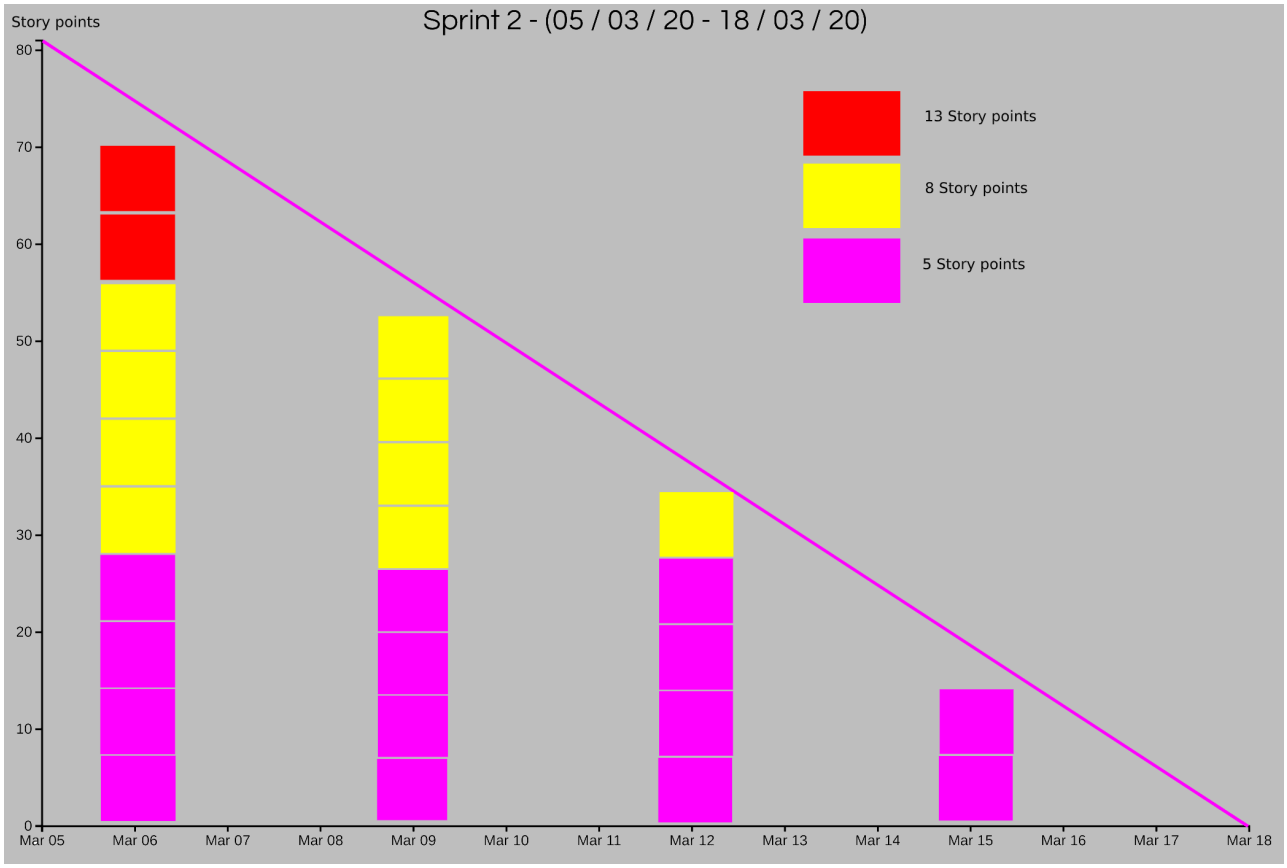
- (3)(I) add clear and graded area constants to visualisation back-end
- (4)(I) Implement UI such that those areas are proportionate to runway
- (1)(DM) Testing & debugging

Total story points:

8

3.2.2 Sprint Burndown

Every rectangle in the chart below represents a user story as there were too many tasks to be able to reasonably display on the chart.



4 Response to Envisioning Feedback

In response to the feedback we were given for the Envisioning part of this project, we made the following changes:

4.1 Sprint Burndown and Backlog

As seen above, we added visual representation of the story points to the Sprint Burndown chart as well as added more detail to the sprint backlog (story points per task and preliminary assignments).

4.2 Product Backlog

We changed the product backlog to make all of the provided requirements a **MUST** in MoSCoW prioritisation. We then changed all of the extension tasks to either **SHOULD** or **COULD** dependent on the perceived usefulness of the request. This left us with no tasks currently set to **WON'T**, with this prioritisation only being utilised at a later point should the team and customer agree that a task will not be completed.

As well as this, we added the Jira generated IDs to the backlog in order to more clearly associate them with the equivalent issues that exist on there. The updated product backlog can be seen below:

ID	Name	Description	Priority
SEG-22	Airport Definition	As a Runway Technician I want to be able to declare a new Airport so that I can store its runways within it.	MUST
SEG-6	Predefined Obstacle List	As a Runway Technician I want to have access to a list of predefined obstacles so that I don't spend as much time defining the size of the obstacle on the runway.	MUST
SEG-1	Obstacle Placement	As a Runway Technician I want to be able to declare the exact position of the obstacle on the runway so that the revised runway is as accurate as possible.	MUST
SEG-24	Obstacle Notification	As a Runway Technician I want to receive a notification from the system so that I know that I have added an obstacle.	MUST
SEG-7	Successful Runway Revision Notification	As a Runway Technician I want to receive a notification from the system so that I know that I have successfully a revised a runway.	MUST
SEG-25	Runway Update Notification	As a Runway Technician I want to receive a notification from the system so that I know that runway values have changed.	MUST
SEG-2	Obstacle Definition	As a Runway Technician I want to be able to define a new obstacle so that it can be placed on the runway for use in the calculations.	MUST
SEG-3	Runway Definition	As a Runway Technician I want to be able to define a new runway so that I can use it in calculations that may be required to determine if a runway should be re-declared or closed.	MUST
SEG-4	Revised Runway Calculations	As a Runway Technician I want to be able, given a runway and an obstacle, to calculate revised runway dimensions so that my manager can decide whether we should proceed with official calculations.	MUST
SEG-26	Obstacle Saving	As a Runway Technician I want to be able to save obstacles so that I can reuse it and save time in the future.	MUST
SEG-8	Input Error Checking	As a Runway Technician I want the system to check my input errors so that I don't end up producing incorrect results.	MUST
SEG-27	XML Airport Import	As a Runway Technician I want to be able to import details of the airport via an XML file so that I do not have to manually define it every time I use the system.	MUST
SEG-28	XML Obstacle Import	As a Runway Technician I want to be able to import obstacles via an XML file so that I do not have to manually define them every time I use the system.	MUST
SEG-29	XML Data Import	As a Runway Technician I want to be able to import other data via an XML file so that I do not have to manually define it every time I use the system.	MUST
SEG-30	XML Obstacle Export	As a Runway Technician I want to be able to export details of obstacles in an XML format so that I can use that data on other systems.	MUST
SEG-31	XML Airport Export	As a Runway Technician I want to be able to export details of airports in an XML format so that I can use that data on other systems.	MUST
SEG-32	XML Data Export	As a Runway Technician I want to be able to export other data in an XML format so that I can use that data on other systems.	MUST
SEG-33	System Accessibility - Screen Reader	As a Runway Technician I want to be able to use a screen reader so that I can use the system properly if I have impaired vision.	SHOULD
SEG-34	System Accessibility - Colour Scheme	As a Runway Technician I want to be able to change the colour scheme of the visual representation of the runway so that if I were colour-blind, there is no mistake when viewing the visualisation.	SHOULD
SEG-5	Data Comparison	As an Airfield Operations Manager I want to be able to view the re-calculated values and the originals so that I can more easily make a decision about runway re-declaration.	MUST
SEG-9	Calculation Breakdown	As an Airfield Operations Manager I want to be able to view a breakdown of the calculations so that I can compare them with the calculations made by my qualified personnel.	MUST
SEG-35	Runway Sideways	As an Airfield Operations Manager I want to be able to visualise the runway with the obstacle from a sideways perspective so that I can decide whether official calculations are necessary or the runway should be closed.	MUST
SEG-36	Runway Bird's-eye	As an Airfield Operations Manager I want to be able to visualise the runway with the obstacle from a bird's-eye perspective so that I can decide whether official calculations are necessary or the runway should be closed.	MUST
SEG-50	Simultaneous View Runway	As an Airfield Operations Manager I want to be able to visualise the runway from both a sideways and bird's-eye view simultaneously so that I can compare both perspectives to help determine whether to close the runway or re-declare it.	MUST
SEG-37	Change Thresholds	As an Airfield Operations Manager I want to be able to select different runways and thresholds so that I can see how the obstacle may obstruct taking-off and landing operations.	MUST
SEG-38	Runway Rotation	As an Airfield Operations Manager I want the system to automatically rotate the top-down view to the appropriate angle based on the compass heading so that I am able to easily visualise the runway re-declaration.	MUST
SEG-51	Clear and Graded Area	As an Airfield Operations Manager I want to be able to see the clear and graded area on the bird's-eye view of the runway so that I can determine if official calculations are required dependent on if the obstacle is located in that area.	MUST
SEG-43	JPEG Runway	As an Airfield Operations Manager I want to be able to export the displays in a JPEG format so that I can use the generated visualisation outside of the system.	SHOULD
SEG-44	PNG Runway	As an Airfield Operations Manager I want to be able to export the displays in a PNG format so that I can use the generated visualisation outside of the system.	SHOULD
SEG-45	GIF Runway	As an Airfield Operations Manager I want to be able to export the displays in a GIF format so that I can use the generated visualisation outside of the system.	SHOULD
SEG-46	Runway Colour Scheme	As an Airfield Operations Manager I want to be able to change the colour scheme of the visual representation of the runway so that if I were colour-blind, there is no mistake when viewing the visualisation.	SHOULD
SEG-47	Screen Reader	As an Airfield Operations Manager I want to be able to use a screen reader so that I can use the system properly if I have impaired vision.	SHOULD
SEG-39	3D Runway View	As an Airfield Operations Manager I want to be able to view the airfield in 3D so that I can more easily judge the severity of an obstruction caused by an obstacle.	COULD
SEG-40	Print Visual Representation	As an Airfield Operations Manager I want to be able to print out visual representations of redeclared runways so that information that is more understandable to most can be transferred around more quickly.	COULD
SEG-41	Real-World Overlay	As an Airfield Operations Manager I want to be able see a map view that overlays the runway diagram over a real-world image of it so that I am more easily able to visualise the obstacle(s) and proposed re-declaration.	COULD
SEG-42	Extra Visual Control	As an Airfield Operations Manager I want to have the ability to zoom and pan in any of the views so that I can examine certain details of the scenario more closely.	COULD
SEG-43	Print Result	As an Airfield Operations Manager I want to be able to print out the results of the current simulation so that I can easily physically show or share this information with other people.	COULD

4.3 Risk Analysis

We added a bit more detail to the mitigation strategies for the risk analysis, which can be seen in the updated version below:

Risk	Probability (1, low - 5, high)	Severity (1, low - 5, high)	Risk Exposure (P x S)	Mitigation
Commitments to other modules/tasks	5	3	15	The team will ensure that specific time, both individual and as a group, is allocated to the project. A scrum master will also be put into place to help with scheduling the workload.
Unable to access Git repository	1	5	5	The team will try to keep their source code as up-to-date as possible on their own PCs to ensure that should this happen, work can still continue for the time being.
Assigned task taking longer than expected	4	2	8	The team will provide additional support to the person who is performing the task in question in order to achieve the deadline, or shift user stories to future sprints to make the workload more manageable.
Individual with an appointed task becomes ill or otherwise unavailable	3	4	12	The team will cover for any such circumstance by splitting up any tasks that were assigned to the individual amongst all other team members.
University strike action	5	2	10	The team will use any support still available to them during the strike action should module staff be unavailable, notably requesting additional aid from their supervisor should problems arise.
Damage to university facilities (e.g. fire)	1	3	3	The team will find another facility to use for meetings. They may also organise meetings over some VoIP service in order to accommodate for the lack of physical meeting space.
Difficulties learning JavaFX and other new libraries	4	2	8	Experienced members of the team will assist others with learning these libraries, and/or the team will make good use of online support and documentation on the libraries/JavaFX.
Animosity between group members	2	3	6	Members of the team not involved in the dispute will attempt to de-escalate the situation, or will contact their supervisor or module staff to assist with resolving the issue(s).
Other technical problems	4	2	8	The team will make sure that they support each other in the event of any technical problems. As a team of 5 computer scientists, these issues should be resolved fairly swiftly.