Test plan

Each test specifies: what is tested, action performed, expected outcome, actual outcome and comments.

## Unit test

* Done by the assigned programmer for their module: if you’re programming Model, you write unit tests for Model.
* Unit tests for model and view but primarily model to test the calculator with the data provided in the spec.
* Test data are the data from the spec and boundary cases:
  + Model: No obstacle
  + Model: Obstacle which is out of range of the runway (length > TORA).
  + View: Not entering data in text fields into the GUI and click OK.
  + View: Entering wrong type of data into text fields into the GUI and click OK.
  + Etc.

## Integration test

* Done for the controller.
* Tests if the user interactions in View maps to the correct corresponding function in model.
* Test cases are sample user action with the user interface and checking for if they carry out their designated function.
* Test cases include
  + New > Airport > [data entry] > Edit > Airport
  + Edit > Airport > [data entry] > Edit > Place obstacle
  + Etc.

## System test

* Deploying the application on different operating systems:
  + Windows
  + Linux
  + Mac OS
* Run a small portion of test cases from unit test for view to check for UI consistency across different operating systems.
  + Set of test cases will trigger all the dialog boxes so all user interactions can be checked.

## Acceptance test

* Determine if application has met all spec points
* Run black box test from the GUI
* Action performed will be actions the user will take to achieve each requirement.
* There will be one test case for each requirement from the spec.