

Test Case Design

Unit Testing:

When creating components in Angular, test files are generated alongside the functional files. This allows for Test Driven Development at the unit level.

The base class files in the Rails applications will require unit tests.

Integration Testing:

As the Rails portion of the application will be API-based, the Integration testing will be testing the Rails back-end.

System Testing:

The entire system needs to be fully functional for system testing. Once ready, the system can be tested thoroughly. System tests will be through the front-end of the application, and the operations will carry through the entire system.

Testing Tools:

[Karma](#) – Angular Unit tests during application development.

[RSpec-Rails](#) – Ruby on Rails testing framework useful for Unit & Integration tests alike.

[Protractor](#) – End-to-end (system) testing for Angular applications. Simulate user behavior and assert application responses.

Test Cases:

Functionality Tested	Inputs	Expected Output	Actual Output
Submit Report	Title	Error: Description needed	
Submit Report	Description	Error: Title needed	
Submit Report		Error: Title & Description	
Submit Report	Title, Description	Success: Report generated	
Submit Report	Code injected inputs	Success: Sanitized input, report generated	
View Statistics	0 reports	Error: No information given	
View Statistics	All reports	Success: Stats generated	
Hide Graph Item	Click on data item in graph	Success: Hidden data item	
Unhide Graph Item	Click on data item in graph legend	Success: Unhidden data item	
'Agree' on Issue	Click 'Agree'	Issue vote increases, displayed graph updates	
'Disagree' on Issue	Click 'Disagree'	Issue vote increases, displayed graph updates	

Additional test cases added.