

Project Proposal

Sanil Shah | UNI: ss4924 | ss4924@columbia.edu

Topic: Analysis and evaluation of JavaScript testing frameworks

Overview

I plan to work on my project independently and build on the topic I explored while researching and writing my midterm paper. The project will involve understanding several JavaScript testing frameworks in greater detail. The project will first implement a JavaScript application that will interact with DOM elements so as to mimic a real world application. Unit tests for this application will then be implemented using three different test frameworks. The project will aim to analyze the developer effort and test coverage obtained from implementing unit tests in each of the frameworks to better understand the most effective framework or tool to use. The primary research question being answered is: which JavaScript unit test framework is easiest to use for first time users or beginners? The project is targeted at beginner JavaScript developers who wish to test their code but have a hard time choosing from the various test frameworks available.

Project Details

I will start by writing a simple tic-tac-toe application where a two human players can play against each other on the same computer. This implies that there will be no computer AI involved but both players are at the same computer passing the mouse back and forth. This is to ensure that the program being tested stays relatively simple and that most of the emphasis of this project is on testing instead of creating a strong AI or computer player. Once the application is written along with the UI and DOM interactions to allow users to properly play the game, I will add unit tests for the application using three different test frameworks. The project will concentrate on Mocha, Jasmine and Jest since they seem to be the most commonly adopted test frameworks today. While implementing these test frameworks, I will measure the time taken to write the unit tests, setup the test framework and how long the test suite takes to build and run from scratch. Based on the features available in each framework I will also add integration, functional or snapshot tests to ensure that the UI looks correct and the application behaves correctly when actually being used. This will be done without integrating with Selenium if possible in the interest of time and brevity.

The project report will compare the differences in productivity and developer effort from implementing tests in each of these test frameworks. It will also measure code coverage to understand how effective testing techniques were across test frameworks. Most new JavaScript developers shy away from testing because of the setup time needed to get these frameworks up and running and properly integrated with their code, so this project will measure and evaluate this metric as well. As a first time user for each of these test frameworks, I believe that my implementing them from scratch for a new application will provide an unbiased benchmark.

Deliverables

The following code will be submitted using GitHub. They will be submitted under a new project here: <https://github.com/sanils>

- Tic-tac-toe application written in JavaScript
- Tic-tac-toe front end and UI in HTML and CSS that interacts with the JavaScript
- Unit tests implemented using Mocha
- Unit tests implemented using Jasmine
- Unit tests implemented using Jest
- Integration or end-to-end tests where applicable
- Snapshot tests implemented in Jest

Further, as part of the deliverables, the project will present a detailed comparison of the metrics discussed above in the project details section. I expect to be able to demo the actual tic tac toe game as well as running the unit tests from the command line to show them passing using all three test frameworks.