



UNIVERSITEIT VAN PRETORIA  
UNIVERSITY OF PRETORIA  
YUNIBESITHI YA PRETORIA

## TESTING POLICY DOCUMENT

COS 301

### Brogrammers

Marcus Bornman *u15024522*

Vernon Francis *u15185703*

Leonardo Ianigro *u16036094*

Myron Ouyang *u16008368*

Keagan Seddon *u14017823*

Michael Yatrakos *u15015531*

By submitting this document we confirm that we have read and are aware of the University of Pretoria's policy on academic dishonesty and plagiarism and we declare that the work submitted in this assignment is our own as delimited by the mentioned policies. We explicitly declare that no parts of this assignment have been copied from current or previous students' work or any other sources (including the internet), whether copyrighted or not. We understand that we will be subjected to disciplinary actions should it be found that the work we submit here does not comply with the said policies.

October 16, 2018

# Contents

|          |                                     |          |
|----------|-------------------------------------|----------|
| <b>1</b> | <b>Introduction</b>                 | <b>2</b> |
| 1.1      | Purpose of This Document . . . . .  | 2        |
| 1.2      | What is Testing? . . . . .          | 2        |
| 1.3      | Why is Testing Required? . . . . .  | 2        |
| <b>2</b> | <b>The Testing Process</b>          | <b>2</b> |
| 2.1      | Unit Testing . . . . .              | 2        |
| 2.2      | Integration Testing . . . . .       | 2        |
| <b>3</b> | <b>Test environment</b>             | <b>3</b> |
| 3.1      | Travis CI . . . . .                 | 3        |
| 3.2      | Jest . . . . .                      | 3        |
| <b>4</b> | <b>Test evaluation</b>              | <b>3</b> |
| <b>5</b> | <b>Approach to test improvement</b> | <b>3</b> |

# 1 Introduction

## 1.1 Purpose of This Document

The purpose of this document is to provide a comprehensive description of how and why testing will be done on the system being developed. It is to ensure that as the system progresses it maintains a desired level of functionality with minimal errors.

The requirements stated in this document will be applied system-wide throughout the development process. This ensures that each component of the system will be thoroughly tested before adding on to it.

## 1.2 What is Testing?

Testing refers to the process of verifying and validating that a software program:

- Fulfills all requirements set out in the description of the document.
- Works as expected, without error
- Has been implemented with the required non-functional software characteristics

## 1.3 Why is Testing Required?

Testing is essential to ensure the successful construction of a system. Due to complexity it is near impossible to implement it correctly first time around. It is therefore a necessity to verify each part of the system by testing it.

Potential problems and risks need to be identified early on in systems development in order to ensure they won't be a problem later on in the development process.

Along with detecting defects that exist, testing provides a way to prove to the client that the system is in fact in proper working order and fulfills its requirements.

# 2 The Testing Process

Unit testing and end-to-end integration testing is performed to ensure that all components and containers function as intended.

## 2.1 Unit Testing

Unit tests refer to the testing of individual components. This may be components that exist as user-visible components and the layouts of such, or to the individual testable modules and methods. Tests are also performed on the methods contained within the stores, which act as controllers for the components. Jest, our chosen testing platform, supports snapshot testing in which a shallow render of a component is captured. This shallow render, or snapshot, is used as a template for future tests where the component being tested is compared to the snapshot to ensure that it renders as expected.

## 2.2 Integration Testing

All of the components of the system need to work efficiently together to ensure that the system as a whole operates as expected. Individual components are rendered inside container components. These container components are thoroughly tested to ensure that the integration of the individual components is performed correctly. The system is implemented as one single software application that will carry out all the tasks as described by the requirements document. It is a necessity that all components interact in a way that is predictable and expected.

## **3 Test environment**

### **3.1 Travis CI**

Travis CI is a hosted, distributed continuous integration service used to build and test software projects hosted at GitHub. Travis CI automates testing by evoking the npm test script each time a new commit is pushed to GitHub. The results of these tests are readily available and are emailed to the user which issued the push to GitHub.

### **3.2 Jest**

The tests Travis CI will run will be designed using Jest. Jest is a zero configuration testing platform which is used to test React applications. Jest in conjunction with the node module, Enzyme, allows for the testing of individual components as well as container components. Jest will test all of the parts of the website such as whether or not the layout of a page is correct as well as whether or not the functional requirements are met, this includes the proper operation of all methods in the program.

## **4 Test evaluation**

This refers to how tests are evaluated to ensure the quality of the system is maintained. The system will have to succeed at all tests on Travis CI in order to be deemed as correct. This is because all tests using Jest that are run by Travis CI will be vital to the proper running of the system.

If the system for some reason does not pass the test it will need to be evaluated in order to determine the effect it will have on operation of the system as a whole and the effect it will have on the business.

## **5 Approach to test improvement**

During the regular meetings, the current testing methods will be evaluated and a way forward will be discussed.