# Testing Policy
# Retro Rabbits Claim System

Tlou Lebelo

August 24, 2019

# Contents

# 1 Introduction

## 1.1 Purpose

Testing serves as the basic measure of progress within the software development process. Therefore without testing software throughout the development cycle there is no form of progress which proves that developers have made an attempt to solving the problems specified by the customer, user or encountered during the process.
Also in situations where products need to ensure compliance with regulatory requirements, software testing can safeguard the organization from legal liabilities by verifying compliance (i.e.) Such as protecting user data.

## 1.2 Goal

The main goal in testing is ensuring that Reformed Rabbit Requisition system software fulfills its requirements as stipulated in the systems requirement documentation. The requirements to be meet include both functional and non-functional requirements which might way down the performance, security or any other system qualities.

# 2 Testing Process

## 2.1 Unit Test

Unit testing is performed on a regular basis for any component under development. Unit test are performed on the following use cases:

- REST API for DB
- PDF Report Generator
- Local Data Persistence on mobile
- User Authentication
- eMailing Component

### 2.2 Integration Test

Integration tests are performed on subsystems before integrating the rest of the system. Integration points identified for testing are:

- REST API for DB
- PDF Report Generator
- Local Data Persistence on mobile
- User Authentication
- eMailing Component

### 2.3 Performance Test

Performance tests are performed after each integration test. The following points were chosen for test on speed and effectiveness:

- OCR overall performace
- API communication transmission rate

## 3 Tools

This section will discuss the tools and frameworks used to test the system; it will also describe how to configure them and reasons for the specified tools.

### 3.1 React Native Jest

- Jest is a delightful JavaScript Testing Framework with a focus on simplicity.
- It works with projects using: Babel, TypeScript, Node, React, Angular, Vue and more!
- Configuration: https://jestjs.io/docs/en/tutorial-react-native

### 3.2 Nodejs Chai

- Chai is a BDD / TDD assertion library for node and the browser that can be delightfully paired with any javascript testing framework.
- Configuration: https://www.chaijs.com