

Testing Policy Document

Developed for Vinesh Naidu by Team HighTech

Theo Naidu

16148861

Alexandros Petrou

15291792

Janaki Patil

16006110

Tristan Sander-hughes

17071390

Zi Xin Zhang

15192556

18 May 2109

University of Pretoria, Hatfield
Engineering, Built environment and Information Technology



UNIVERSITEIT VAN PRETORIA
UNIVERSITY OF PRETORIA
YUNIBESITHI YA PRETORIA

Contents

Introduction	3
Test policy	4
Unit testing.....	4
Integration testing	4
System testing.....	4
Specifications regarding testing.....	4
Testing software	4
FireBase TestLab	4
PenTest Tools.....	5
Chai	6
GTMetrix	7

Introduction

Quality assurance of software can only be guaranteed by vigorous testing to ensure the smoothness of running software. Not only does it prevent future cost but also helps improve programming skills in general. The aim of this policy is to test and explain what why and how processes are being tested. As well to show level of outcome we are looking for. We want to show that the requirements set out to us as a team have been met as well. The following done test sections are unit test, integration tests and system tests. Through project life span regression testing has taken place mainly during integration testing.

1. Definition of Testing

The purpose of this testing document is to show potential users and clients that the necessary steps have been taken in order to show that the mobile application as well as the portal admin side. The main reason is ensuring the software fulfils its requirements. The testing techniques used include comparison testing to expected output. Using firebase's built-in apk integration testing.

2. Description of the test process

The test process will be done in the following manner. Each function will be performed as intended and the result of each function will be compared to what the expected output should be. Then a counter will count each correct and incorrect evaluations and display the results.

3. Test Evaluation:

For evaluation we will store the results chosen in their own fields for comparisons later.

4. Quality Level to be achieved:

The level of quality to be achieved has to be 100% at all times as no system can be used with failing functions. The quality criteria will be the results of the evaluations done.

5. Approach to Test Process Improvement

project review meetings to be held after project completion in order to see what can be improved and what functions and processes can be fixed.

Test policy

For the following areas of testing there must be done at certain times and by certain means in order to get full results.

Unit testing

Done by: developers during development

Areas of testing: classes and functional units.

Goal: Determine if all requirements are met.

Find any bugs or undesired errors

Integration testing

Done by: developers during development

Areas of testing: functionality and how units make use of each other, unit interoperability and compatibility, performance.

Goal: Determine if all separate entities can work together.

System testing

Done by: Static quality assurance applications

Areas of testing: Functionality, data quality, performance, reliability, usability, resource utilization, maintainability, installability, portability and interoperability

Goal: Test the system in ways that regular code cannot in ways that measure other features such as sustainability, security and others.

Specifications regarding testing

During testing the follow scenarios are always used

- Normal tests: test performed using normal valid input to get justified outcomes
- Follow up random tests: tests done using randomised values to see how modules handle unknown data after known data
- Independent random tests: full on random tests
- Stress tests: see how much the system can handle at a time within reasonable limit
- Integration test: see if most or all module can co exist together without crashing
- Invalid tests: feed the modules wrong data to see if it has enough counter measures

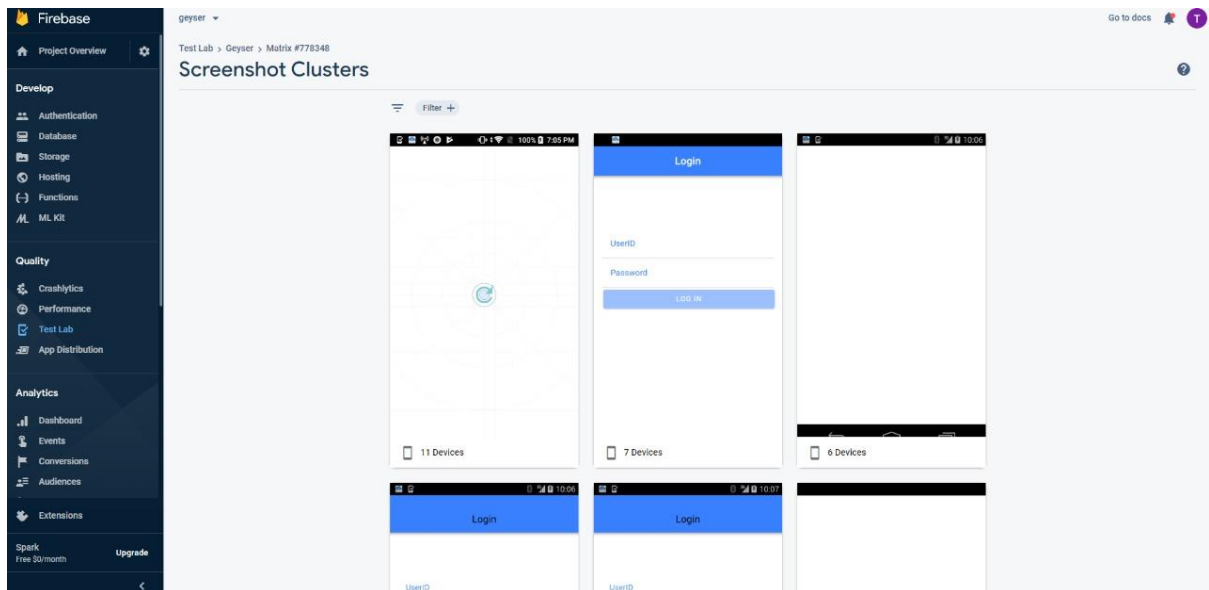
Testing software

Whilst testing functionality and modules other tools were used to test other features of our systems.

FireBase TestLab

FireBases TestLab was used in order to test our mobile application in order to determine if it can work on multiple platforms varying in requirements and can compile without issues. It allows us to detect any errors that may occur during the execution of the app so we may fix it

TestLab running the app and multiple virtual devices for compatibility



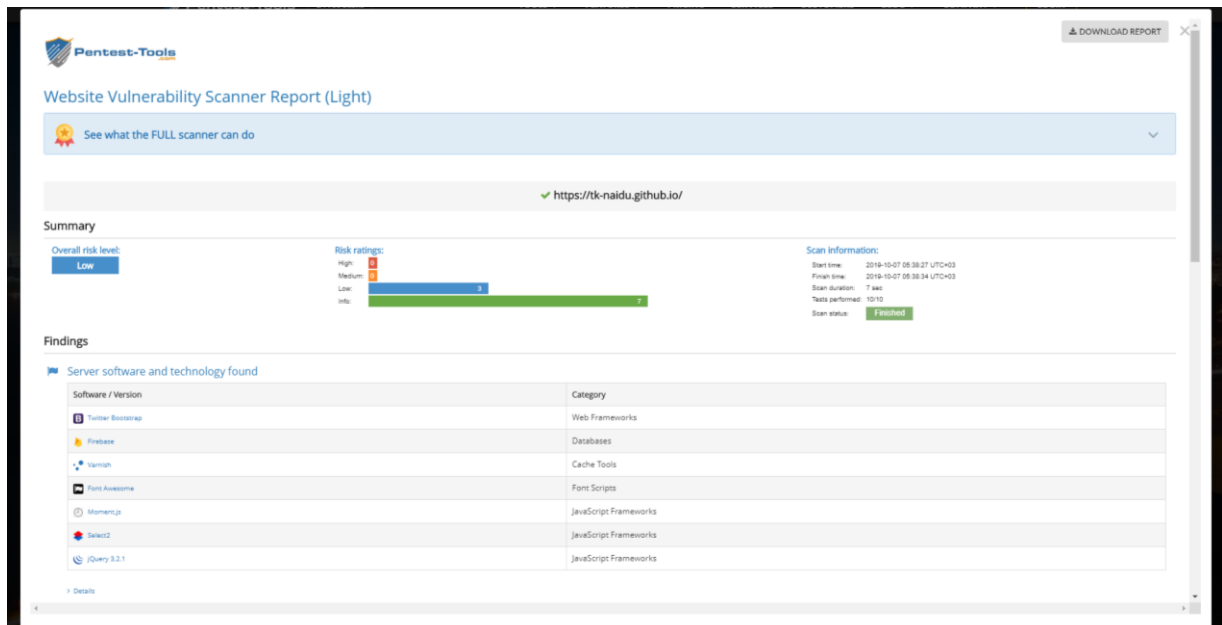
Link to our tests

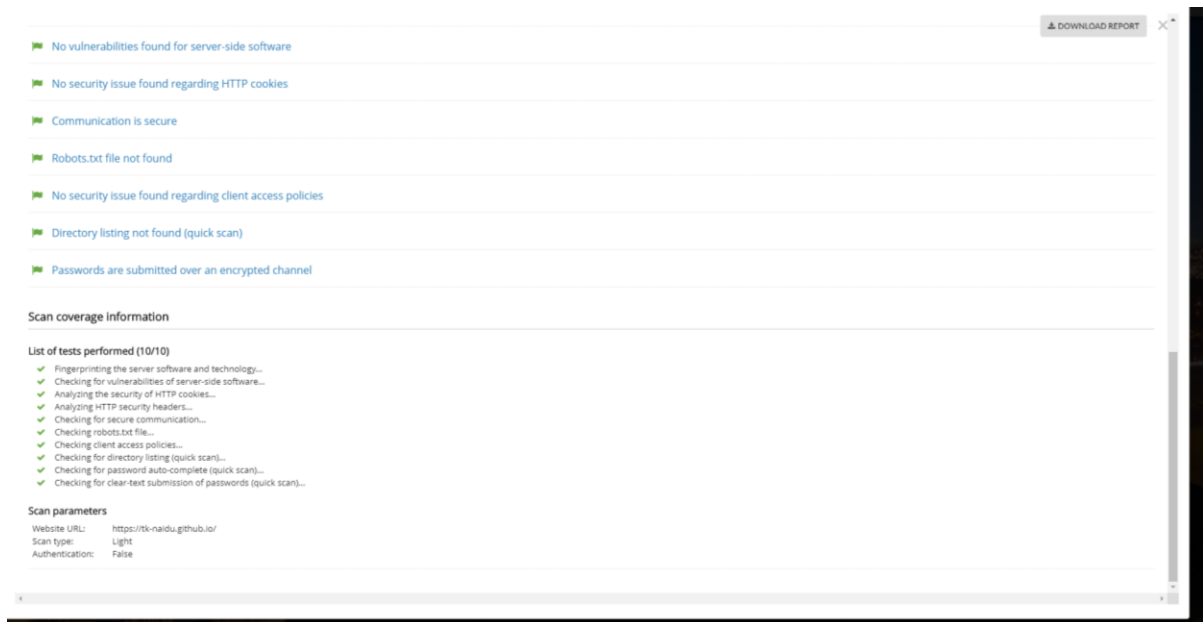
<https://console.firebase.google.com/project/geyser-74ddf/testlab/histories/bh.df84d700050654b2>

PenTest Tools

A website penetration software tool that allows us to see just how vulnerable our website currently is.

Summary of the test results ran.





Link to Pentest

<https://pentest-tools.com/website-vulnerability-scanning/website-scanner#>

Chai

A unit test js language framework that allows you to runs automated test and view easily which test pass and which fail

Test layout.

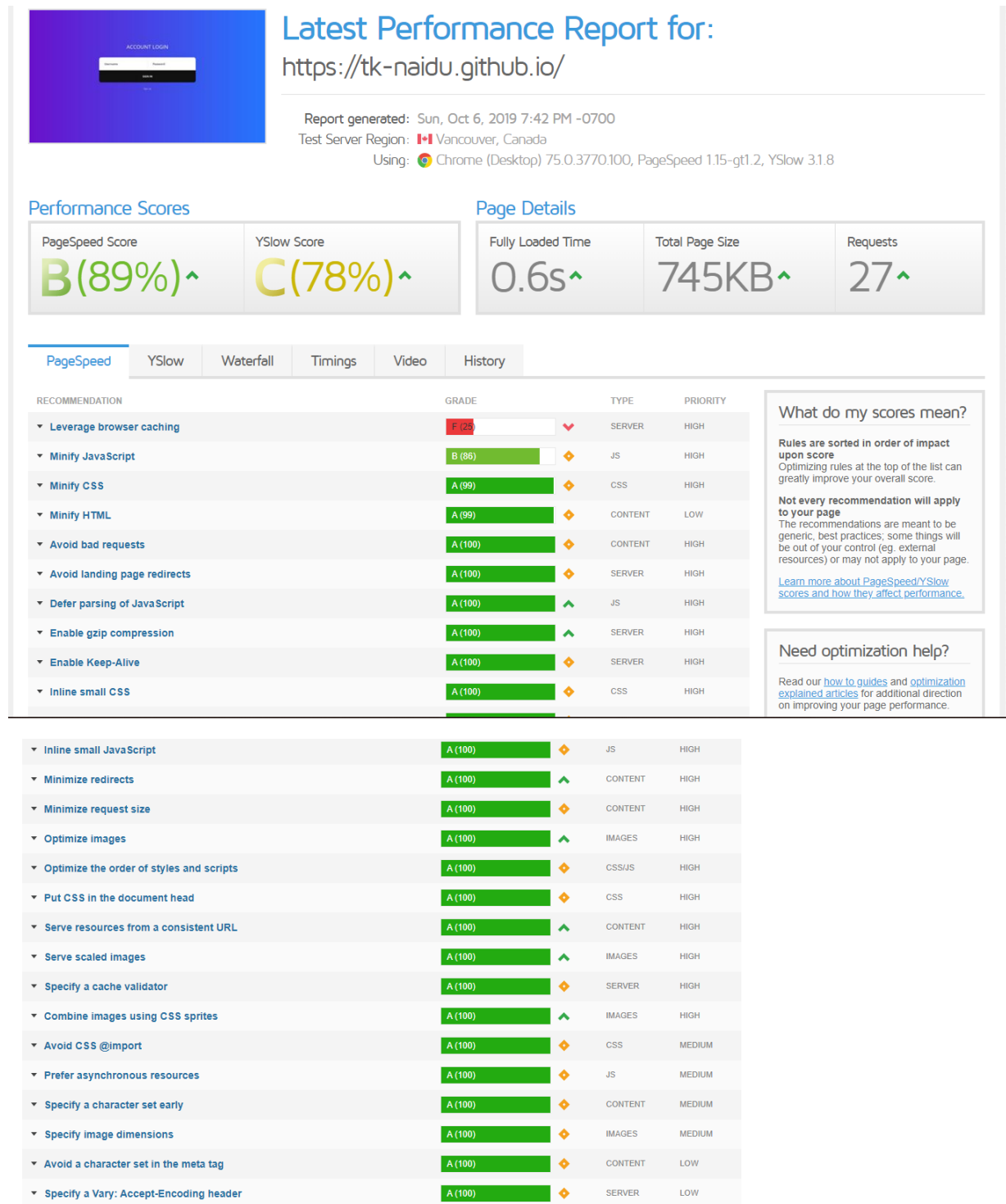
```

1  var expect = require('chai').expect;
2  var fs = require("fs");
3  var request = require("request");
4  var rp = require('request-promise');
5
6
7  var capacity1 = "20K1";
8  var modell = "1321";
9  var barcode1 = 1132564;
10 var insurance1 = "yes";
11 var manufacturer1 = "kwikot";
12
13 // owner details
14 var name1 = "Bob";
15 var surname1 = "McLain";
16 var cellNumber1 = 1326549870;
17 var address1 = "1 up street Pretoria";
18 var firebaseConfig = {
19   apiKey: "AIzaSyD1bg-oYRpVXU9HqCA0LdxIpYXcZUPpAGM",
20   authDomain: "geyser-74ddf.firebaseio.com",
21   databaseURL: "https://geyser-74ddf.firebaseio.com",
22   projectId: "geyser-74ddf",
23   storageBucket: "geyser-74ddf.appspot.com",
24   messagingSenderId: "1001903322970",
25   appId: "1:1001903322970:web:a904b91e99d67b22"
26 };
27 // Initialize Firebase
28 firebase.initializeApp(firebaseConfig);
29 var db = firebase.firestore();
30
31
32 describe('Unit testing', function () {
33   console.log("Testing Upload of details unit");
34   it('Should allow the upload of a new entry to the data base ', function () {
35     var res = storeData();
36     console.log("Uploaded success");
37     expect(res).to.equal("success");
38     barcode1="adf1214121";
39     barcode1 = 1132564;
40
41   });
42   it('Should attempt to upload new details but should fail due to incorrect format of a field', function () {
43     var res = storeData();
44     console.log("Upload failed due to incorrect barcode format");
45     expect(res).to.equal(false);
46     insurance1="";
47
48   });
49   it('Should attempt to upload new details but should fail due to missing field value ', function () {
50     var res = storeData();
51     console.log("Correct format");
52     expect(res).to.equal(false);
53
54   });
55 });

```

GTMetrix

A website evaluation tool that checks various features of a site including use of css, speed, stability and availability



Link

<https://gtmetrix.com/reports/tk-naidu.github.io/SE8xVJGs>