Test Plan

Ivanti’s Website

3-10-2021

Version 1.0

Contents

[Introduction 2](#_gjdgxs)

[Test methodology 2](#_30j0zll)

[Used approaches 2](#_1fob9te)

[Unit tests 2](#_3znysh7)

[Integration tests 2](#_2et92p0)

[Acceptance tests 2](#_tyjcwt)

[SonarQube 3](#_3dy6vkm)

[Conclusion 3](#_1t3h5sf)

# Introduction

The Test Plan is designed to test the main functionality of Ivanti’s website project. The tests are broad and not specific as we are still at the start of the project. But they are going to give the main idea of whether the units and API work fine.

# Test methodology

## Used approaches

### Unit tests

We have made several unit tests so we can make sure whenever something in a certain unit (class) is changed it won’t break the whole application. Also, to do that we first run the tests and make sure they pass and afterwards we run the application.

### Integration tests

As we recently studied about CI/CD we will try to actively include it in our project because this feature is pretty useful for big projects where we have many tests on pieces of code and it will be easier to have a person developing some new feature instead of testing something which can be tested by a computer. Also, unit tests can be run using CI/CD. It is easy to update and maintain the existing pipelines.

### Acceptance tests

User acceptance tests are one of the most expensive as we can say. Thus, it is because we have to gather people who have no idea how the application works and give them instructions on how to use it or just let them explore it on their own. So it is time-consuming but this is also the most efficient way to see whether their unusual/unexpected actions are going to break the app. There we have some steps and expected results.  
 --- table with Acceptance tests

### SonarQube

Using SonarQube we can see easily how maintainable and usable is our code. We can see security vulnerabilities, warnings of probable bugs and more.  
 --- picture after including it

# Conclusion

In this Test plan we are going to show how the application was tested. This way we can make sure that everything is working according to the wanted behavior, without any bugs and problems.   
We will see whether we need Tests Results document, or we can use this one.