**Table of Contents**

[**1. Introduction 1**](#_1fob9te)

[○ Analogues 2](#_3znysh7)

[○ Used technologies 2](#_2et92p0)

[○ Documents 2](#_tyjcwt)

[○ Project team 2](#_3dy6vkm)

[○ Project deadlines 2](#_1t3h5sf)

[○ Communications approach 2](#_4d34og8)

[**2. Test Approach 3**](#_2s8eyo1)

[○ Types of Testing 4](#_3rdcrjn)

[○ Priorities for test activities 4](#_fbqlr8roa8w1)

[○ Weekly objectives for QA team 4](#_t3yx1ylgz2ma)

[○ Using templates 4](#_28xjbgu9ggk)

[3. Acceptance criteria 4](#_i7z4e67m8zn)

[**3. Test Environment 4**](#_lnxbz9)

[**4. Testing Tools and Artifacts 5**](#_35nkun2)

[● Testing artifacts for the Project: 5](#_1ksv4uv)

[● Defect Life Cycle: 5](#_44sinio)

[● Automated testing: 6](#_2jxsxqh)

[**5. Release Control 7**](#_z337ya)

[○ Testing entry criteria: 7](#_1y810tw)

[○ Testing exit criteria: 8](#_4i7ojhp)

[**6. Additional Procedures 8**](#_qsh70q)

[1. Meetings: 8](#_3as4poj)

[2. Work review: 8](#_1pxezwc)

[3. Testing team: 9](#_49x2ik5)

[**7. Common risks 9**](#_2eo1akz3d35q)

[**8. Test Plan (link to document) 9**](#_z4r34j607qmy)

# 

# Introduction

The purpose of the test strategy for the **SimpleLand** is to:

* Provide a central artifact to govern the strategic approach of the test effort; it defines the general approach to be employed when testing the software and when evaluating the results of that testing. Planning artifacts will refer to the test strategy regarding the governing of detailed testing work.
* Provide visible confirmation to test-effort stakeholders that adequate consideration has been given to governing the test effort and, where appropriate, to have those stakeholders approve the strategy.

## Analogues

* [durable.co](https://durable.co/)
* [Editorx.com](https://manage.editorx.com/account/sites?referralAdditionalInfo=Route)

## Used technologies

* Common
  + [TypeScript](https://www.typescriptlang.org/)
* FrontEnd
  + [React](https://reactjs.org/)
  + [ReduxToolkit](https://redux-toolkit.js.org/)
  + [SCSS](https://sass-lang.com/)
* BackEnd
  + [NodeJS](https://nodejs.org/en/)
  + [Fastify](https://www.fastify.io/)
  + [Knex](https://knexjs.org/) & [Objection](https://vincit.github.io/objection.js/)
  + [WS](https://developer.mozilla.org/en-US/docs/Web/API/WebSockets_API)
* DB
  + [PostgreSQL](https://www.postgresql.org/)
* Host
  + [AWS](https://aws.amazon.com/ru/)

The project is deployed to https://simpleland-development.herokuapp.com/my-projects.

Github actions are used for CI and CD.

Scrum model of SDLC.

## Documents

[Design](https://www.figma.com/file/KpnOr1khE6DW8ykObHg5Iz/AI-website-builder-BSA)

[Specification](https://docs.google.com/document/d/1_3hI47xHqFSN0C5VJ5PNdqVzeZargTCn7HQW3iL2gzo/edit?pli=1#)

## Project team

The team that will work on the project:

* developer team;
* QA team;
* 3 coaches of the developer team;
* 1 mentor of the QA team;
* Product Owner

## Project deadlines

10/03/2023 - project starts;

15/04/2023 - release.

All works must be completed before 14/04/2023.

## Communications approach

Testers can communicate:

* with questions that arise during the development to mentor and coaches;
* with questions about the product to the product-owner during the demo;
* with any technical questions to the development team.

# Test Approach

A **proactive** approach will be used in the testing - an approach in which the test design process is initiated as early as possible in order to find and fix the defects before the build is created.

* Manual testing
* Automated testing

Automated testing will cover all high-level features (only API tests should be automated).

## Types of Testing

The following types of testing will be used:

* Functional testing
* User interface testing
* Smoke testing
* Regression testing
* Acceptance testing

## Priorities for test activities

* Critical bugs to be tested.
* Features to be tested.
* Testing approach, criteria & deliverables.
* Duties & tasks assigned to each tester.

## [Weekly objectives for QA team](https://docs.google.com/spreadsheets/d/1PBBmrcs9_28AhS9-TKfMsMcXTsvGSwM8G_a6S5msGuE/edit#gid=0)

## Using templates

**Bug Report Template:**

1. Summary
2. Environment
3. Steps to reproduce
4. Actual Result
5. Expected Result
6. Visual Proof (screenshots, videos, text)
7. Severity

**User Story template:**

1. Background
2. Idea

As a (ROLE), I want to (ACTION) so that I can (VALUE/OUTCOME)

#### Acceptance criteria

# Test Environment

* **Setup required for the test environment:**
* Browser: Google Chrome 110.0.5481.177
* Display resolution: 1680x1050
* Display orientation: horizontal
* Localization of the project: English
* Number of environments are defined by coaches on each project

<https://simpleland-development.herokuapp.com/>

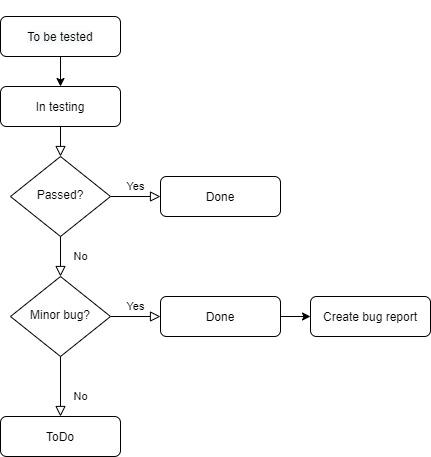
# Testing Tools and Artifacts

## Testing artifacts for the Project:

All artifacts should be published on the Project’s GitHub board

* Test strategy,
* [Project Test plan](https://docs.google.com/document/u/0/d/1m6IthJe9MdiDe0SToLG60NpWeW59dM-yNeAn0a3lfM0/edit),
* [Mind map](https://docs.google.com/spreadsheets/u/0/d/1zX2qiR1JXV2VwlIOOm_4UobCfoMgXz505_QcQajWZ40/edit),
* [Check-list](https://docs.google.com/spreadsheets/u/0/d/1AM8JsN5V9hZvbl9KTLNknzL2h3Jrq_SbxilSy_zIEm8/edit),
* [Bug reports](https://docs.google.com/spreadsheets/u/0/d/1KLO8tCuE5fW6YfBkm3iyHeT9ioCoga2quwvfSLPoBzo/edit),
* Traceability matrix (optional),

## Defect Life Cycle:

****

Severity:

Critical – Critical Impact/System Down. Complete system outage.

Major – Significant Impact/Severe downgrade of services.

Minor – Minor impact/Most of the system is functioning properly.

Trivial – Low Impact/Informational.

## Automated testing:

Automated testing should be implemented for previous sprints.

We will use Postman (TS, mocha, chai, got)

We will use GIT hub actions for CI/CD

# Release Control

The project workspace is organized on the GitHub project board:

GitHub is used as a code hosting platform for version control and collaboration for this project.

## Testing entry criteria:

* the testing process begins at the requirements analysis stage,
* control of the design process at the stage of developing a system project,
* fulfillment of acceptance criteria of the tickets.

## Testing exit criteria:

* all works must be completed not later than: 14/04/2023,
* fixed 95% of found defects, with no blocking, critical and high severity,
* management decision.

# Additional Procedures

**To improve the testing process next additional procedures are needed:**

## Meetings:

* Daily SCRUM team meeting;
* QA mentor meeting;
* QA team meetings (optional);
* Demo with PO.

## Work review:

* Weekly review of test artifacts with QA mentor.
* Sprint retrospective with QA team.

## Testing team:

Responsibilities of each QA team member:

* writing test documentation;
* specification analysis;
* manual testing;
* API testing (manual and automated);
* Report bugs and errors to development teams.

The master branch must be frozen at least in one day before demo each sprint.

# Common risks

What should someone do about it? Define responsible person and action point when this happens:

| Risk | Action point | Responsible |
| --- | --- | --- |
| Absence of team member | redistribute tasks between members; |  |
| Faulty test results | provide testing according to documentation; cover functionality as much as possible by tests; |  |
| Test environment not available | install all needed tools;  provide good internet connection; |  |
| Not enough time for testing | prioritize tasks and features; first cover main functionality by tests; |  |

# [Test Plan](https://docs.google.com/document/d/1m6IthJe9MdiDe0SToLG60NpWeW59dM-yNeAn0a3lfM0/edit)

Test plan document should be composed by QA team

It must contain:

1. Feature priorities
2. Team availability hours
3. Test approach (backend/frontend decomposition)
4. What will we automate? On what level?
5. Features planned
6. Features implemented
7. Demo date / final scope