**HTML5 GAM-Test Plan**

**Change history**

|  |  |  |
| --- | --- | --- |
| **Version** | **Date** | **Summary of Change** |
| V1.1 | Jul 2014 | Initial Draft |
|  |  |  |

**Content**

This document provides guidance for Complete Testing Solution for HTML5 GAM Project.

**Distribution**

This document is for use by Ordnance Survey staff. The document, or any part of it, **must** **not** be supplied or communicated to any other individual or organization without the prior written permission of the owner.

**Trademarks**

Ordnance Survey and the OS Symbol are registered trademarks of Ordnance Survey, the national mapping agency of Great Britain.

Contents

[1 Introduction 3](#_Toc393267909)

[1.1 Purpose of this document 3](#_Toc393267910)

[1.2 Project Background 3](#_Toc393267911)

[1.3 Test Objectives 3](#_Toc393267912)

[1.4 Reference 3](#_Toc393267913)

[2 Scope 3](#_Toc393267914)

[2.1 Test Items 3](#_Toc393267915)

[2.2 Features To Be Tested 4](#_Toc393267916)

[2.3 Features not to be tested (for beta release) 4](#_Toc393267917)

[3 RISKS, Impact and Mitigations 5](#_Toc393267918)

[4 Approach 6](#_Toc393267919)

[4.1 Sprint Testing 6](#_Toc393267920)

[4.2 Regression TESTING (Through Automation) 6](#_Toc393267921)

[4.3 Automation Testing 6](#_Toc393267922)

[4.4 Component Testing 7](#_Toc393267923)

[4.5 Integration Testing (Through Automation testing) 7](#_Toc393267924)

[4.6 Compatibility Testing (Through Automation Testing) 7](#_Toc393267925)

[4.7 Usability Testing: 7](#_Toc393267926)

[4.8 Performance Testing 7](#_Toc393267927)

[4.9 Security Testing: 8](#_Toc393267928)

[5 Entry,Exit,Suspension and Resumption 8](#_Toc393267929)

[5.1 Entry Criteria: 8](#_Toc393267930)

[5.2 Exit Criteria 8](#_Toc393267931)

[5.3 Suspension Criteria 8](#_Toc393267932)

[5.4 Resumption Criteria 8](#_Toc393267933)

[5.5 Testing Process 8](#_Toc393267934)

[5.6 Test Deliverables 8](#_Toc393267935)

[5.7 Testing Tasks 9](#_Toc393267936)

[5.8 Test Data 9](#_Toc393267937)

[5.9 Responsibilities (Inherited from Test Strategy) 9](#_Toc393267938)

[5.10 Resources 10](#_Toc393267939)

[6 Schedule 11](#_Toc393267940)

[7 Environmental Requirements 12](#_Toc393267941)

[7.1 Hardware 12](#_Toc393267942)

[7.2 Software and Tooling 12](#_Toc393267943)

[8 Plan Approvals 12](#_Toc393267944)

# Introduction

## Purpose of this document

Demonstrate Test plan to be carried through the project until go Live

Demonstrates the extent to which the Test team has understood the nature and scope of the project

## Project Background

The goals of the HTML5 Getamap project are to replace the current Getamap application with an HTML5 version that can be viewed by Desktop and mobile browser based client. This project be being built purely based agile best practices.

Further aiming the benefits of Agile best practices, Testing function is integrated and team is expected to implement testing observing agile methodologies.

Test strategy is defined and agreed around agile implementation. Please refer to Test Strategy document for more details.

## Test Objectives

Test Objective is to deliver MVP for a Beta launch in Autumn (Oct 2014) .

This would include fully functional and non-functional tested. Testing shall include sprint level functional, full regression testing, Compatibility testing, performance testing and Security testing.

It’s been agreed that all must requirements is to be fully tested and triage all issues which will be prioritized by stake holders. Depending on the business decisions and known issues, decision may be made to go for beta release or extend bug fixing time.

## Reference

* *Project plan*
* *Test Strategy* [*https://confluence.ordsvy.gov.uk/display/IGMAP/HTML5GetaMap-TestStrategy*](https://confluence.ordsvy.gov.uk/display/IGMAP/HTML5GetaMap-TestStrategy)
* *Confluence(wiki)* [*https://confluence.ordsvy.gov.uk/display/IGMAP/Intelliscan+Get-a- Map+Home*](https://confluence.ordsvy.gov.uk/display/IGMAP/Intelliscan+Get-a-%20%20Map+Home)
* *Application Architecture* [*https://confluence.ordsvy.gov.uk/display/IGMAP/Application+Architecture*](https://confluence.ordsvy.gov.uk/display/IGMAP/Application+Architecture)

# Scope

## Test Items

1. Data transformation & loading - migration of all live data which currently exists within GAM.(Not for Beta)
2. Verification of current GAM functionality implemented into the new solution
3. Verification of any new functionality
4. Account creation & registration of new customers
5. Account access for existing registered users(Not for Beta Release)
6. Customer preferences/Settting
7. Verifying user profiles & restrictions for new and existing customers. i.e. Guest users/Registered users/Subscribers
8. Receive the payments for subsribers on embedded in getamap(on shops)
9. Expiry of user subscriptions and renewal emails from shops.
10. Standard UI functionality
11. Creating & editing routes (new & old)
12. Exporting & importing of routes
13. Purchases(subscription through getamap)
14. Integration with Magento including registration, user authentication, payments
15. Printing
16. Browser compatibility for desktop and mobile devices
17. System interfaces - internal /external API's
18. Verification of performance related to NF requirements(TBC with JT)
19. Security

## Features To Be Tested

1. Road Routing (A-B)
2. Road Navigation
3. Route planning
4. Create groups
5. Find a route
6. Search UGC
7. Viewing OS Maps.
8. Routes
9. *Printing*
10. *Sharing(Social media and email)*
11. Nicknames(Registration )
12. Mobile responsive
13. Weather(live data rom met office)
14. Favourites(add as favourite)
15. POI
16. *Registration*
17. *3D Map(Sprint 6)*
18. Flythrough(once route is plotted,want to preview route)
19. Import / Export routes(GPX functionality, import/export to garmin device if possible)
20. Manage routes
21. Manage groups
22. Analytics(Google analytics)
23. Other Pages(legal customer services)
24. Terrain 50(sprint 6 , replaces landform, required for elevation)
25. Admin – upload(upload POIs) schema in db into which data is added.
26. Admin – moderation. (Admin manage)

## Features not to be tested (for beta release)

Below list of features will be not tested as part of Beta release

However these will be tested for go live release.

1. Offroad routing
2. Crowdsourcing – implicit
3. Crowdsourcing – Explicit
4. White labelling
5. User Generated Content (UGC) – POI
6. UGC – Challenges
7. UGC – Day out
8. Gamification – scoring
9. Gamification – rewarding
10. Vector mapping
11. Imagery(OS Ariel)

# RISKS, Impact and Mitigations

|  |  |  |  |
| --- | --- | --- | --- |
| Risk | Description | Impact | Mitigation |
| **Environment Risk** | OS may provide integration environment for testing which may not be stable. | High impact on delivery of quality tested feature | Will be given dedicated environment –Testing environment |
| **Product Risk** | No documents on functional level implementation | High | Test analyst must approach Requirement analyst very frequently to confirm functional behaviour. |
| **Remote Development** | Lack of communication with developers on daily basis to check with the progress | Medium | Communication on regular basis in terms of bugs raised, replication steps, build in which its testable, test blockers etc.. |
| **Testing Debt** | Development is ahead of testing, this mean testing is required to cover up old sprints with high level functionality. | Low | Will be Covered by QA. |
| **Automation Testing** |  | Low( may impact on regression release cycle time) | Will be covered by QA iteratively adding more regression test scenarios in regression pack. |
| **Resource Risks** | Only one resource is allocated for this project | High, may impact some areas of testing (cross Browser testing) | Mobile QA can cover up this risk as and when needed. |

# Approach

As this project is developed in agile environment where new features are being developed iteratively, Testing will be performed at various stages.

## Sprint Testing

With BDD approach,

A thorough story level testing will be performed at this stage and bugs will be raised at sprint level or regression level.

**Its agreed that P1 and P2 bugs which are related to current sprint story will be actioned immediately by development team and other issues will be guided by PO**.

Test Deliverables in Sprint Testing

* Test feature files (Test Cases)
* Delivery of Story which is satisfying all must criteria which is again debated by PO.
* Regression Defects (which will be triaged)
* Automated Test scripts(for automatable scenarios)
* Test report(Sprint level)

## Regression TESTING (Through Automation)

As development is following agile process, all new features are built iteratively in Sprints. With this approach it is quite important to makes sure new features doesn’t impact or break existing ones. To verify this we always need to perform regression testing.

This is the most critical testing required to be performed frequently to make sure existing functionality is still retained and not broken. A formal time period is allocated to perform regression testing before making any major release. It is agreed that test analyst will create **Automation Regression** pack to reduce the regression cycle time and also support cross browser testing very efficiently through Automated testing.

Regression testing is expected to commence

* On new build deployment on dev,CI, QA(testing) Environments.
* Before end of sprint.
* On request on CI, ITE environments.
* Before making any major releases. (formal allocated time period)

**Test Deliverables**

* Regression test Scenarios(feature files)
* Automated Tests scripts (Automated Regression pack)
* Regression bugs(which will be triaged)

## Automation Testing

Test analyst will implement Automation testing as part of Qa process. All regression test scenario which are automatable will be automated in this project. A framework approach is already built which will be extended to accommodate regression suite. This Automated regression pack will be run every now and then when there is a need of regression testing in the project.

Test Deliverables:

* Automation scripts for new features
* Regression pack for the project.

## Component Testing

As the name defines, each component will be tested to ensure that the component works as designed.

Purpose of component level testing is to give good test coverage at individual component level to make sure it works accurately when integrated with other components. Story requirements and all possible edge cases are dealt at component level.

Also there might be requirement for component level performance test which can be done at this stage.

Developers might need to help testers to make components individually tested by providing dummy consumer or stub. This will be mostly in api level testing.

**Test Deliverables**

* Test cases at component level
* Defects (regression or sprint bugs)
* Automated test scripts(to go in Automated regression pack)

## Integration Testing (Through Automation testing)

A thorough integration testing will be performed through manual and automated testing.

Purpose of this testing is to ensure all other systems/services interacting with getamap would work as expected.

A detail end to end user journey test scenarios will be tested as part of Integration testing. This will be part of sprint testing itself.

**Test Deliverables:**

* End to End user journey Test Scenarios(connecting to external services, eg., Magento,Bin,Met)
* Defects around Integration.(Sprint level,Regression bugs)
* Automated test scripts for end to end Test Scenarios

## Compatibility Testing (Through Automation Testing)

Only functional test scenarios will be run on different OS/Browsers list to verify functionality works as expected.

A list of supported OS and browsers is given on wiki <https://confluence.ordsvy.gov.uk/display/IGMAP/Supported+Browsers>

**Automated regression pack** will be customized to run test on agreed supported browsers. This will performed at sprint level testing and also Regression level testing.

A dedicated machine Selenium Grid will be used to run Automation tests on different browsers.

A sub test plan may be need to written to perform compatibility testing to make sure a thorough testing is done.

## Usability Testing:

It is been suggested/agreed that for GAM, Usability testing will be performed officially by group of stake holders, customer service team. A sub test plan will be created to perform this testing and recording valid user experiences and make decisions accordingly.

## Performance Testing

A high level performance testing on system will be performed by test analyst using **Jmeter tool** to understand the system performance.

A performance test plan will be created prior performing this activity.

Tester will require inputs in terms of response time, through put, number of concurrent users etc., from Technical coordinator of this project.

## Security Testing:

As agreed security testing will be performed using Web Inspect tool. A security scan will be performed on the application urls and provide output to project Architect.

Training or resource may be required to carry out this testing activity in this project.

# Entry,Exit,Suspension and Resumption

## Entry Criteria:

Test analyst is expected to run through entry criteria testing can be commenced,

Make sure when new build is ready for test

1. List of features/Bugs are marked as ready for test with build number.
2. Smoke test is run and passed without any major application errors
3. Test Data (to be supplied by Supplier is ready)
4. Test Environment readiness.

## Exit Criteria

Test Analyst responsibility to make sure all possible test deliverables are provided to PO and stake holders in making decision for project sign off.

* Test execution evidence
* Test Coverage Report indicating the requirement coverage.
* Jenkins test execution report for Automated test execution.
* Manual Execution test report (Excel or any other format)
* Defect Summary report
* Reports indicating Test phase completeness

## Suspension Criteria

Testing will be suspended on following situations

* When the requirement changes after implementation.
* Incomplete UX design to test(development prototype won’t be tested)
* On Environmental issues(not stable environment to test)
* Features not in current build.
* Insufficient test data to validate certain requirements.

## Resumption Criteria

In order to resume testing follow criteria should be met

* All suspension criteria are resolved.
* Should re-run the smoke test if it was suspended due to environmental issues.
* Will resume when features made available in current build for testing.

## Testing Process

The core test process for HTML5 GAM should include planning and control, Analysis and Design, Implementation and execution, Evaluate exit criteria and Reporting and finally Test closure Activity

## Test Deliverables

*1. Sub Test plans for Performance, Security and Compatibility test*

*To include following key elements*

* *Objectives*
* *Risks*
* *Test Scope*

*2. Test Cases (feature files)*

*3. Automated Test Scripts(Test Automation)*

*4. Test Execution Report*

*5. Defect Summary Report*

## Testing Tasks

*In general testing tasks should include*

*At story level*

* *Story analysis and identify test Scenarios*
* *Test case prep*
* *Cross Browser testing*
* *Prepare performance test plan*
* *Test Case Execution*
* *Regression Testing*
* *Test Closure activities*

*Release level*

* *Run Automated Regression testing*
* *Manual regression Testing*
* *Report Defect Summary report.*

## Test Data

Following data is always required in Testing Environment (ITE)

**POIs Data**

This data is usually supplied by third party in different file formats which should be consumed in our system. This is considered to be manual process.

**Route Data**

Following route data is required in ITE dB for testing

Public, Authored, national trials

Authored routes are normally provided by third party in different file formats which is consumed in Gam app database.

Seems OS has already supplied above data to Intelliscan for them to process in HTML5 Gam app.

## Responsibilities (Inherited from Test Strategy)

|  |  |  |
| --- | --- | --- |
| **Responsibility** | **Sub-actions (if split)** | **Owner** |
| Test Strategy |  | OS |
| Test Plan |  | OS & Intelliscan |
| Unit tests | creation & maintenance | Intelliscan |
| execution | OS |
| Code Quality tests | configuration | Intelliscan |
| execution | OS |
| Functional tests | creation & maintenance | Intelliscan |
| execution | OS |
| Integration tests | creation & maintenance | Intelliscan |
| execution | OS |
| Story Acceptance tests | creation & maintenance(TDD) | Intelliscan |
| execution(through Jenkins) | OS |
| User Acceptance tests |  | OS & OSL |
| Compatability tests |  | OS |
| Usability tests |  | OS |
| Performance tests |  | OS |
| Data Quality tests |  | Intelliscan |
| Security tests |  | OS |
| Test data | supply | OS & OSL |
| preparation | Intelliscan and OS |
| Test reporting |  | OS |
| Test environments | setup & maintenance | OS |
| Continuous Integration |  | OS |
| Defect tracking | tooling setup & maintenance | OS |
| defect reporting | all |

## Resources

|  |  |  |
| --- | --- | --- |
| **Name** | **Capacity** | **Responsibilities** |
| Nasir Khan | Test Lead | * Manage Testing Solution for the project * Define Test Strategy and Test Plans * Report to Project Management for Test progress. * Test Impediments, * Liaise with Intelliscan team on day to day basis in regards to testing blockers defects, issues. * Schedule and control Non-functional testing. * Liaise with Project Management in conducting Usability Testing. |
| ?????(take support from mobile project) | Tester | Support Testing Activity in the Project such as  Write Test Cases  Execute Test cases  Raise Defects |

# Schedule

Below table defines high level testing milestones with end date. This may vary depending on the testing dependencies and Impediments.

This is inherited from project plan.

|  |  |  |
| --- | --- | --- |
| **Testing Milestones** | **End Date** | **Comments** |
| Define Test Plan | 18-Jul-2014 |  |
| Initial Security Testing | 17-Jul-2104 |  |
| Sprint 4 Testing |  |  |
| Automation Regression Pack(Test debt for sprint 1,2&3) | ?? | TBC-Depending on the backlog priority. |
| System Regression (Manual + Automation) | (1-2 weeks provided no dependencies) |  |
| Defect Re-Testing | 2 days |  |
| Compatibility Testing across the system(regression test on supported browsers) |  |  |
| Performance Testing on dedicated Environment | 01-Oct-2014 |  |
| Vulnerability Security Scan on GAM | 01-Oct-2014 |  |
| Test Summary Report | 03-Oct-2014 |  |
| Usability Testing |  | TBC |

# Environmental Requirements

As part of Test environment requirements, it is always expected that test environment is up and running with the build which is intended for testing.

Intelliscan must and should provide Test team with proper build details such as what features ,bug fixes being delivered in the release.

It is always recommended for Intelliscan to send release notes or mark Jira ticket with release numbers to notify testers to re-test in proper build.

Internal Hosting team should give testers enough access and control to deploy latest code on designated test environment.

## Hardware

Dev Box(VM) for Automation Script Development

*New Selenium Grids (VM) Instance for Automation test execution*

## Software and Tooling

*Required software tool for testing in this project*

***IDE***

* *Eclipse IDE(Luna)*

***Version control***

* *SVN repo*

***Automation tools***

* *Selenium Webdriver*
* *Cucumber-jvm*
* *Sublime note pad(for writing feature files)*
* *Jmeter(for performance testing)*

# Plan Approvals

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Stake Holders | Role | **Approved Yes/NO** | **Dated** | **Comments** |
| Claire Loon | Project Manger |  |  |  |
| Lee Newton | Product Owner |  |  |  |
| John Tidmus | Solution Architect |  |  |  |
| Jonathan Scott | Technical Manager |  |  |  |
| Nasir Khan | Test Lead |  |  |  |