# Test plan

(https://vwo.com/)

**Version:** 1.0

**Created:** 04/03/2025

**Last Updated: 04/03/2025**

**Status:** DRAFT

**Document History**

| Version | Date | Author | Description of Change |
|---|---|---|---|
| V00.01 | 04/03/2025 | Mercy Swamy | Draft |

# 1. INTRODUCTION

## 1.1. Purpose

This test plan describes the testing approach and overall framework that will drive the testing of the VWO (https://vwo.com/)

The document introduces:

- **Test Strategy**: The rules the test will be based on, including the project's details (e.g., start/end dates, objectives, assumptions). It also describes the process to set up a valid test (e.g., entry/exit criteria, creation of test cases, specific tasks to perform, scheduling, data strategy).
- **Execution Strategy**: How the test will be performed, the process to identify and report defects, and how to fix and implement fixes.
- **Test Management**: The process to handle the logistics of the test and all events that arise during execution (e.g., communications, escalation procedures, risk and mitigation, team roster).

Project team members perform tasks specified in this document and provide input and recommendations on it.

- **Project Manager**: Plans the testing activities in the overall project schedule, reviews the document, tracks the performance of the test according to the tasks specified herein, approves the document, and is accountable for the results.
- **Stakeholders' Representatives and Participants**: Individuals identified by the PMO Leads may take part in the UAT test to ensure the business is aligned with the test results.
- **Technical Team**: Ensures that the test plan and deliverables align with the design, provides the environment for testing, and follows procedures related to fixing defects.
- **Business Analysts**: Provide their input on functional changes.

## 1.2. Project Overview

VWO is an enhanced SmartStats web application. The Bayesian-powered sequential testing engine gives users full control over the statistical parameters of their experiments to ensure precise, statistically robust results for confident decision-making. Users can create metrics using any event, customize their statistical settings, and save them for reuse in future campaigns.

VWO helps users obtain error-free, real-time reports with its stats engine, which auto-adjusts for peeking and multiple comparisons. It reduces visitor loss by quickly disabling underperforming variations, enabling faster evaluation of others. VWO monitors experiment

health, performs background checks, and notifies users of issues, ensuring a seamless experience.

## 1.3. Audience

Project team members perform tasks specified in this document and provide input and recommendations.

- **Project Manager**: Plans the testing activities in the overall project schedule, reviews the document, tracks the performance of the test according to the tasks specified herein, approves the document, and is accountable for the results.
- **Stakeholders' Representatives and Participants**: Individuals identified by the PMO Leads may take part in the UAT test to ensure the business is aligned with the test results.
- **Technical Team**: Ensures that the test plan and deliverables align with the design, provides the environment for testing, and follows procedures related to fixing defects.
- **Business Analysts**: Provide their input on functional changes.

# 2. TEST STRATEGY

## 2.1. Test Objectives

The objective of the test is to verify the functionality of VWO according to the requirement specifications.

The test team will:

- Verify and execute the test scripts.
- Identify and retest all high and medium severity defects per the acceptance criteria.
- Prioritize lower severity defects for future fixing.

The final product of the test is threefold:

- A production-ready quality product.
- A set of stable test scripts that can be reused for Functional and UAT test execution.
- A dedicated and knowledgeable QA team for the product, capable of addressing any future feedback from clients

## 2.2. Test Assumptions

**Key Assumptions**

- A sprint is created and started for a specific time frame, focusing on implementing the defined scope for the given project.
- Project receivables exist and are available for the test team to understand the functionality to be tested.
- Production-like data is required and should be available in the system prior to the start of functional testing.
- Each testing environment (Dev, QC, and Staging) is stable enough to carry out test execution.
- The development lifecycle will follow an Agile model, where the product will be built in sprints.
- Whenever there are enhancements or new features in the requirement specification, end-to-end testing will be carried out to ensure builds are stable and continue to meet the defined product and business requirements.
- Bugs and defects detected will be analyzed to determine their severity, and regression testing across the app will be performed on subsequent builds upon fixes made available by the DEV team.
- All defect fixes and retests will be performed within the sprint timeline.

**General**

- The test team assumes all necessary inputs required during test design and execution will be appropriately supported by the development team and business analysts.
- Test environment creation and preparation activities will be owned by the development team.
- Business analysts will review and sign off on all test cases prepared by the test team prior to the start of test execution.
- The QA team will provide test planning, test design, and test execution support.
- Test case design activities will be performed by the QA group.
- Exploratory testing will be carried out once the build meets all the functional specifications.
- All defects will include a snapshot in JPEG format.
- The development team will provide defect fix plans based on the defect meetings during each cycle. The same will be communicated to the test team prior to the start of defect fix cycles.
- Defects will be tracked through JIRA only. Any defect fixes planned will be shared with the test team before applying the fixes in the test environment.
- The test team will be provided with access to the test environment via WiFi and mobile networks.
- The project manager/business analyst will review and sign off on all test deliverables.
- The test team will manage the testing effort in close coordination with the project manager/business analyst.

- The project team has the necessary knowledge and experience, or has received adequate training in the system, the project, and the testing processes.

**Functional Testing**

- The QA team will carry out end-to-end functional and non-functional testing on the DEV, Test, and Staging environments.
- If the QA team identifies any improvements in the functionality, usability, or performance of the product, these will be tracked through JIRA for future enhancement by the client and will be treated as value-add elements.

**UAT**

- UAT test execution will be performed by both the QA team and end users. The QA group will provide support in creating UAT scripts.
- UAT will be performed only on the Staging Build (Production Environment).

## 2.3. Test Principles

- Testing will focus on meeting business objectives, cost efficiency, and quality.
- There will be common, consistent procedures for all teams supporting testing activities.
- Testing processes will be well-defined, yet flexible, with the ability to change as needed.
- The test environment and data will emulate a production environment as much as possible.
- Testing will be a repeatable, quantifiable, and measurable activity.
- Testing will be divided into distinct phases, each with clearly defined objectives and goals.

## 2.4. Data Approach

In functional testing, testers will feed the website with input test data by applying their experience and knowledge of the application and the business processes involved. There is no dependency on third-party (TP) input test data.

## 2.5. Scope and Levels of Testing

### 2.5.1. Functional Test

**Purpose**: Functional testing will be performed to check the functions of the application. Functional testing is carried out by feeding the input and validating the output from the application.

### SCOPE:

### VWO Web based application :

VWO (Visual Website Optimizer) offers a comprehensive suite of tools for optimizing digital experiences and maximizing conversions. Here are some key features and products provided by VWO:

1. **VWO Testing**:
   - **A/B Testing**: Create and run tests to compare different versions of a webpage or app to determine which performs better.
   - **Multivariate Testing**: Test multiple combinations of elements to find the best-performing combination.
   - **Split URL Testing**: Compare different URLs to see which one performs better.
2. **VWO Insights**:
   - **Heatmaps**: Visualize where users click, scroll, and hover on your website.
   - **Session Recordings**: Record and replay user sessions to understand their behavior.
   - **On-Page Surveys**: Collect feedback directly from users while they are on your website.
   - **Form Analytics**: Analyze how users interact with forms on your website.
3. **VWO FullStack**:
   - **Server-Side Testing**: Perform enterprise-grade server-side testing for complex scenarios.
   - **Mobile App Testing**: Test and optimize mobile applications.
4. **VWO Engage**:
   - **Web Push Notifications**: Send messages directly to users' browsers to keep them engaged.
   - **Cart Abandonment Campaigns**: Reach out to users who have abandoned their shopping carts.
5. **VWO Personalize**:
   - **Personalized Experiences**: Create tailored experiences for different user segments based on their behavior and preferences.
6. **VWO Plan**:
   - **Test Management**: Plan, manage, and track all your testing activities in one place.

**TESTERS:** Test Team Members.

**METHOD:** The test will be performed according to Functional scripts, which are stored in Test Rail.

Below are the different types of functional testing that will be carried out:

1) Risk Based Testing

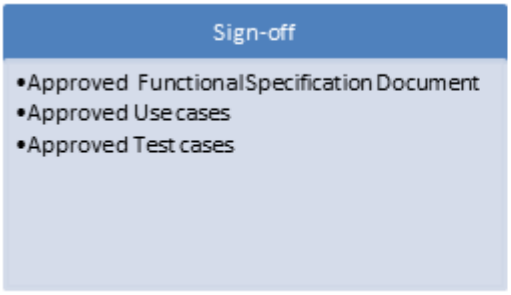
2) Priority Based Testing

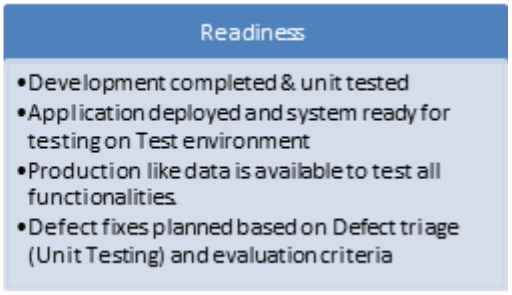
3) Smoke Testing

TIMING: Functional Testing begins during the DEV_INT cycle with focus to identify as many bugs as possible both for web frontend and backend functionality and continues into QC cycle where both regression and End to End QC is performed on each cycle to verify the sanctity of the builds.

**TEST ACCEPTANCE CRITERIA**

- Approved Functional Specification document and Use Case documents must be available prior to the start of the test design phase.
- Test cases must be approved and signed off prior to the start of test execution.
- Development must be completed, unit tested with a pass status, and results shared with the testing team to avoid duplicate defects.
- The test environment must have the application installed, configured, and in a ready-to-use state.




**TEST DELIVERABLES**

| S.No. | Deliverable Name | Author | Reviewer |
|---|---|---|---|
| 1. | Test Plan | Test Lead | Project Manager/ Business Analyst's |
| 2. | Functional Test Cases | Test Team | Business Analyst's Sign off |

| | | | |
|---|---|---|---|
| 3. | Logging Defects in JIRA | Test Team | Test Lead/ Programming Lead |
| 4. | Daily/weekly status report | Test Team/ Test Lead | Test Lead/ Project Manager |
| 5. | Test Closure report | Test Lead | Project Manager |
| 6. | Regression Test Suite | Test Team | Test Lead |
| 7. | UAT Test Suite | Test Team | Test Lead |

### 2.5.2. Non-Functional Test

**Smoke Testing**: Smoke testing will be carried out prior to full-blown testing to ensure that the test environment/test build is stable.

**Performance Testing**: Performance testing will be carried out to validate the system's response time, data loss, or application crash.

| S.No. | Deliverable Name | Author | Reviewer |
|---|---|---|---|
| 1. | Performance Test Cases | Test team | Test Lead |
| 2. | Smoke Test Cases | Test team | Test Lead |
| | | | |

### 2.5.3. User Acceptance Test (UAT)

**PURPOSE:** This test focuses on validating the business logic. It allows end users to complete one final review of the system prior to deployment.

**TESTERS:** The UAT is performed by the end users.

**METHOD:** Since the business users are the most qualified to provide input around business needs and how the system adapts to them, they may perform some validation not contained in the scripts. The test team writes the UAT test cases based on the inputs from end users and business analysts for performing UAT testing.

**TIMING:** UAT is conducted after all other levels of testing (functional and adhoc) are completed. The product can only be released to production upon successful completion of UAT

**TEST DELIVERABLES**

| S.No. | Deliverable Name | Author | Reviewer |
|---|---|---|---|
| 1. | UAT Test Cases | Test Team | Business Analyst's Sign off |

# 3. EXECUTION STRATEGY

## 3.1. Entry and Exit Criteria

The entry criteria refer to the desirable conditions required to start test execution. Only the migration of the code and fixes need to be assessed at the end of each cycle.

The exit criteria are the desirable conditions that need to be met to proceed with the implementation.

Entry and exit criteria are flexible benchmarks. If they are not met, the test team will:

- Assess the risk,
- Identify mitigation actions, and provide a recommendation.

All this information is input to the project manager for a final "go/no-go" decision.

- **Entry criteria to start the execution phase of the test:** The activities listed in the Test Planning section of the schedule are 100% completed.
- **Entry criteria to start each cycle:** The activities listed in the Test Execution section of the schedule are 100% completed at each cycle.

| **Entry Criteria** | **Test Team** | **Technical Team** | **Notes** |
|---|---|---|---|
| Test Bed is ready with required test data | Yes | | |
| Test Environment is stable | | Yes | |
| Test Build is unit tested & ready | | Yes | |

The exit criteria are the desirable conditions that need to be met in order to proceed with the implementation.

| Exit Criteria | Test Team | Technical Team | Notes |
|---|---|---|---|
| 100% Test Scripts executed | | | |
| 95% pass rate of Test Scripts | | | |
| No open Critical and High severity defects | | | |
| 95% of Medium severity defects have been closed | | | |
| All remaining defects are either cancelled or documented as Change Requests for a future release | | | |
| All expected and actual results are captured and documented with the test script | | | |
| All defects logged in JIRA | | | |
| Test environment cleanup completed and a new back up of the environment | | | |

## 3.2. Validation and Defect Management

It is the responsibility of the tester to:

- Open the defects.
- Create under the specific story or link them to the corresponding Story.
- Assign an initial severity and status.
- Retest and close the defect.

It is the responsibility of the Defect Manager to:

- Review the severity of the defects.
- Facilitate the fix and its implementation with the technical team.

- Communicate with testers when the test can continue or should be halted.
- Request the tester to retest.
- Modify the status as the defect progresses through the cycle.

It is the responsibility of the technical team to:

- Review JIRA on a daily basis.
- Ask for details if necessary.
- Fix the defect.
- Communicate to the Defect Manager once the fix is done.
- Implement the solution per the Defect Manager's request.

Defects found during testing will be categorized according to the bug-reporting tool “JIRA.” The categories are:

| Severity | Impact |
|---|---|
| 1 (Critical) | • This bug is critical enough to crash the system, cause file corruption, or cause potential data loss<br>• It causes an abnormal return to the operating system (crash or a system failure message appears).<br>• It causes the application to hang and requires re-booting the system |
| 2 (Blocker) | • It causes a lack of vital program functionality with workaround.<br>• This bug prevents other areas of the product from being tested further. |
| 3 (Major) | • This Bug will degrade the quality of the System. However there is an intelligent workaround for achieving the desired functionality - for example through another screen.<br>• This bug prevents other areas of the product from being tested. However other areas can be independently tested. |
| 4 (Minor) | • There is an insufficient or unclear error message, which has minimum impact on product use. |
| 5 (Trivial) | • There is an insufficient or unclear error message that has no impact on product use. |

# 4. TEST MANAGEMENT PROCESS

## 4.1. Test Management Tool

Test Rail will be used as the Test Management tool. Here, the tester will be able to:

- Prepare the test cases.
- Run the test cases on different environments.
- Assign the test results to the test cases.
- Prepare graphs and reports to track the testing status.

## 4.2. Test Design Process


## 4.3. Test Execution Process

## 4.4. Test Risks and Mitigation Factors

Test Risks and Mitigation Factors Risk Prob. Impact Mitigation Plan Absence of people for more than 2-3 days Test Schedule will be affected in the agile framework Non-adherence to schedule & quality not upto the mark Proper back up plan Test Environment not stable Testing will not be started on time & test coverage will not be 100% Affected quality Smoke Testing prior to full blown testing

Your section on Test Risks and Mitigation Factors is clear and thorough. Here are a few improvements for clarity and readability:

| Risk | Prob. | Impact | Mitigation Plan |
|---|---|---|---|
| Absence of people for more than 2-3 days | Test Schedule will be affected in the agile framework | Non-adherence to schedule & quality not upto the mark | Proper back up plan |
| Test Environment not stable | Testing will not be started on time & test coverage will not be 100% | Affected quality | Smoke Testing prior to full blown testing |

# 5. Communications Plan and Team Roster

## 5.1. Role Expectations

The following list defines in general terms the expectations related to the roles directly involved in the management, planning or execution of the test for the project.

| SN0. | Roles | Name | Contact Info | Email |
|---|---|---|---|---|
| 1. | Project Manager | Xxxxx Xxxxx | xxx xxx xxxx | xxxxx@xxxx.xxx |
| 2. | Test Lead | Xxxxx Xxxxx | xxx xxx xxxx | xxxxx@xxxx.xxx |
| 3. | Business Analyst | Xxxxx Xxxxx | xxx xxx xxxx | xxxxx@xxxx.xxx |
| 4. | Development Lead | Xxxxx Xxxxx | xxx xxx xxxx | xxxxx@xxxx.xxx |

| | | | | |
|---|---|---|---|---|
| 5. | Testing Team | Xxxxx Xxxxx | xxx xxx xxxx | xxxxx@xxxx.xxx |
| 6. | Development Team | Xxxxx Xxxxx | xxx xxx xxxx | xxxxx@xxxx.xxx |
| 7. | Technical Lead | Xxxxx Xxxxx | xxx xxx xxxx | xxxxx@xxxx.xxx |

### 5.1.1. Project Management

**Project Manager:**

- Reviews the content of the Test Plan, Test Strategy, and Test Estimates and signs off on them.
- Ensures entrance criteria are used as input before starting the execution.
- Develops the test plan and the guidelines to create test conditions, test cases, expected results, and execution scripts.
- Provides guidelines on how to manage defects.
- Attends status meetings in person or via the conference call line.
- Communicates to the test team any changes that need to be made to the test deliverables or application and when they will be completed.

### 5.1.2. Test Planning (Test Lead)

- Ensure entrance criteria are used as input before starting the execution.
- Develop the test plan and guidelines to create test conditions, test cases, expected results, and execution scripts.
- Provide guidelines on how to manage defects.
- Attend status meetings in person or via the conference call line.
- Communicate to the test team any changes that need to be made to the test deliverables or application and when they will be completed.
- Provide on-premise or telecommute support.
- Develop test conditions, test cases, expected results, and execution scripts.
- Perform execution and validation.
- Identify, document, and prioritize defects according to the guidance provided by the Test Lead.
- Re-test after software modifications have been made according to the schedule.
- Prepare testing metrics for each QC Cycle and provide regular status updates.
- Acknowledge the completion of a section within a cycle.
- Give the OK to start the next level of testing.
- Facilitate defect communications between the testing team and the technical/development team.

- Review testing deliverables (test plan, cases, scripts, expected results, etc.) and provide timely feedback.
- Assist in the validation of results (if requested).
- Support the development and testing processes being used to support the project.
- Certify that correct components have been delivered to the test environment at the points specified in the testing schedule.
- Keep the project team and leadership informed of potential software delivery date slips based on the current schedule.
- Define processes/tools to facilitate the initial and ongoing migration of components.
- Conduct the first line of investigation into execution discrepancies and assist test executors in creating accurate defects.
- Implement fixes to defects according to the schedule.

#### 5.1.3. Test Team

- Develop test conditions, test cases, expected results, and execution scripts.
- Perform execution and validation.
- Identify, document, and prioritize defects according to the guidance provided by the Test Lead.
- Re-test after software modifications have been made according to the schedule.
- Prepare testing metrics for each QC Cycle and provide regular status updates.

#### 5.1.4. Test Lead

- Acknowledge the completion of a section within a cycle.
- Give the OK to start the next level of testing.

#### 5.1.5. Development Team

- Review testing deliverables (test plan, cases, scripts, expected results, etc.) and provide timely feedback.
- Assist in the validation of results (if requested).
- Support the development and testing processes being used to support the project.
- Certify correct components have been delivered to the test environment at the points specified in the testing schedule.
- Keep the project team and leadership informed of potential software delivery date slips based on the current schedule.
- Define processes/tools to facilitate the initial and ongoing migration of components.
- Conduct the first line of investigation into execution discrepancies and assist test executors in the creation of accurate defects.
- Implement fixes to defects according to the schedule.

# 6. TEST ENVIRONMENT

## 6.1. Devices to be validated for the Website: -

Latest Chrome, Firefox, Safari and Edge

## 6.2. Network to test with:

WIFI, Internet

## 6.3. Defect Tracking Tool:

JIRA