### **Test Plan for app.vwo.com**

#### **1. Objective**

The goal of this test plan is to ensure that **app.vwo.com** functions correctly, delivers a seamless user experience, and meets the requirements for creating, managing, and analyzing A/B tests, heatmaps, and other optimization features.

#### **2. Scope**

The test plan will cover the following areas:

* **User Authentication**: Login, logout, and account management.
* **Dashboard**: Navigation, widgets, and overview metrics.
* **Test Creation**: A/B tests, split URL tests, and multivariate tests.
* **Editor Functionality**: Visual editor, code editor, and targeting rules.
* **Reporting**: Test results, analytics, and data accuracy.
* **Integrations**: Third-party tools (e.g., Google Analytics, CRMs).
* **Performance**: Load times, responsiveness, and scalability.
* **Security**: Data protection, access controls, and compliance.

#### **3. Test Types**

* **Functional Testing**: Verify all features work as intended.
* **Usability Testing**: Ensure the platform is intuitive and user-friendly.
* **Performance Testing**: Test load times and responsiveness under different conditions.
* **Security Testing**: Validate data security and access controls.
* **Compatibility Testing**: Ensure the platform works across browsers, devices, and operating systems.

#### **4. Test Scenarios**

##### **4.1 User Authentication**

* Verify login with valid credentials.
* Test login with invalid credentials (error handling).
* Test password reset functionality.
* Verify logout functionality.

##### **4.2 Dashboard**

* Check if the dashboard loads correctly.
* Verify that widgets display accurate data.
* Test navigation to different sections (e.g., Tests, Reports, Settings).

##### **4.3 Test Creation**

* Create an A/B test using the visual editor.
* Create a split URL test.
* Test targeting rules (e.g., geography, device type).
* Verify saving and publishing of tests.

##### **4.4 Editor Functionality**

* Test the visual editor for modifying web pages.
* Verify the code editor for custom changes.
* Test undo/redo functionality.
* Check preview mode for changes.

##### **4.5 Reporting**

* Verify that test results are displayed accurately.
* Test filtering options in reports (e.g., date range, segments).
* Check export functionality for reports.

##### **4.6 Integrations**

* Test integration with Google Analytics.
* Verify CRM integration (if applicable).
* Check data synchronization between VWO and third-party tools.

##### **4.7 Performance**

* Test page load times under normal and peak traffic.
* Verify responsiveness on different devices (desktop, tablet, mobile).
* Test scalability with multiple simultaneous users.

##### **4.8 Security**

* Verify SSL encryption for data transmission.
* Test access controls (e.g., admin vs. user roles).
* Check for vulnerabilities (e.g., SQL injection, XSS).

#### **5. Test Data**

* Use real-world data for test creation and reporting.
* Include edge cases (e.g., empty fields, invalid inputs).
* Simulate high traffic for performance testing.

#### **6. Tools and Resources**

* **Testing Tools**: Selenium, JMeter, Postman, BrowserStack.
* **Bug Tracking**: Jira, Trello, or any issue-tracking tool.
* **Test Environment**: Staging environment mirroring production.

#### **7. Timeline**

* **Week 1**: Test planning and environment setup.
* **Week 2-3**: Functional and usability testing.
* **Week 4**: Performance and security testing.
* **Week 5**: Compatibility testing and bug fixing.
* **Week 6**: Final review and reporting.

#### **8. Success Criteria**

* All critical functionalities work as expected.
* No major bugs or performance issues.
* The platform meets usability and security standards.
* Test results are documented and shared with stakeholders.

#### **9. Risks and Mitigation**

* **Risk**: Delays in testing due to incomplete test data.
  + **Mitigation**: Prepare test data in advance.
* **Risk**: Performance issues under high traffic.
  + **Mitigation**: Optimize server configuration and code.
* **Risk**: Security vulnerabilities.
  + **Mitigation**: Conduct regular security audits.

#### **10. Reporting**

* Document all test cases and results.
* Share a summary report with stakeholders.
* Include recommendations for improvements.