### Test Plan for Cross-Platform E-commerce App

#### Testing Objectives

* Verify the correctness of core functionalities, such as user registration, login, product search, adding items to the cart, and checkout.
* Ensure the UI is consistent and functions correctly across different platforms (web, Android, iOS).
* Test compatibility across various devices and browsers.
* Assess the app’s performance under high load scenarios, such as loading multiple products and high user traffic.

#### Testing Scope

* Functional testing on Android and iOS devices with varying screen sizes.
* Testing on Chrome, Firefox, and Safari for the web version.
* Core functionalities of the e-commerce app will be the primary focus of testing.

#### Tools and Approaches

**Automated testing**: Use Flutter’s integration testing tools or automated UI testing tools like Appium for cross-platform functionality.

**Manual testing**: Conduct manual testing to ensure that critical business flows function as expected, especially for UI verification.

### Test Cases for E-commerce App

#### ****User Authentication (Login/Signup)****

**Test Case**: Verify user can register with valid email, password, and other required fields.

**- Expected Result**: User receives a success message, and an account is created.

**Test Case**: Verify login functionality with valid credentials.

**- Expected Result**: User successfully logs in and is redirected to the home page.

**Test Case**: Verify login with invalid credentials.

**- Expected Result**: User receives an error message indicating incorrect credentials.

#### ****Product Search and Filtering****

**Test Case**: Verify product search functionality using a valid product name.

**- Expected Result**: Relevant search results are displayed based on the search term.

**Test Case**: Verify that the product filter by category works correctly.

**- Expected Result**: Products are displayed based on selected category, and the filter works consistently.

#### ****Product Details****

**Test Case**: Verify that clicking on a product opens the product detail page.

**- Expected Result**: The product detail page opens, displaying the correct product information, including name, price, description, and images.

**Test Case**: Verify that product details are displayed consistently across mobile and web platforms.

**- Expected Result**: Product information is fully visible, and the layout is consistent across platforms.

#### ****Shopping Cart****

**Test Case**: Verify the user can add a product to the cart.

**- Expected Result**: Product is added to the cart, and the cart count increases.

**Test Case**: Verify the user can update product quantity in the cart.

**- Expected Result**: Quantity updates in real-time, and total price reflects the update.

**Test Case**: Verify that removing a product from the cart works correctly.

**- Expected Result**: Product is removed from the cart, and the total price updates.

#### ****Checkout Process****

**Test Case**: Verify the user can complete a purchase with valid payment details.

**- Expected Result**: Purchase confirmation is displayed, and order details are saved.

**Test Case**: Verify that invalid payment details are rejected.

**- Expected Result**: User receives an error message about invalid payment information.

**Test Case**: Verify that the checkout process is responsive and functions correctly across devices.

**- Expected Result**: The checkout flow is smooth, without errors, on both mobile and web platforms.

#### ****Responsive Design and Layout Testing****

**Test Case**: Verify the layout adapts correctly to different screen sizes (mobile, tablet, desktop).

**- Expected Result**: All UI elements are properly aligned and displayed without overlapping or cut-off issues.

**Test Case**: Verify the header and footer remain consistent across pages.

**- Expected Result**: Header and footer display correctly on all platforms and devices.

#### ****Performance Testing****

**Test Case**: Verify that product pages load within 2 seconds when there are 100+ products.

**- Expected Result**: Product pages load quickly, maintaining a smooth user experience.

**Test Case**: Test app performance under simulated high user load.

**- Expected Result**: The app remains stable and responsive without crashes or slowdowns.

### Additional Notes

**Test execution**: Prioritize automated testing for repetitive test cases, such as login and product searches. Manual testing can focus on unique scenarios and UI consistency.

**Cross-platform considerations**: Ensure that all critical flows are tested on both mobile and web to confirm uniformity and functionality across platforms.