**Test Plan Template:**

site Retro Games Store

**Prepared by:**

Budnik Serhii

18.03.24

**TABLE OF CONTENTS** 1.0 INTRODUCTION

2.0 OBJECTIVES AND TASKS 2.1 Objectives

2.2 Tasks

3.0 SCOPE

4.0 Testing Strategy

4.1 Alpha Testing (Unit Testing) 4.2 System and Integration Testing 4.3 Performance and Stress Testing 4.4 User Acceptance Testing 4.5 Batch Testing

4.6 Automated Regression Testing

4.7 Beta Testing

5.0 Hardware Requirements

6.0 Environment Requirements

6.1 Main Frame

6.2 Workstation

7.0 Test Schedule

8.0 Control Procedures

9.0 Features to Be Tested

10.0 Features Not to Be Tested

11.0 Resources/Roles & Responsibilities

12.0 Schedules

13.0 Significantly Impacted Departments (SIDs)

14.0 Dependencies

15.0 Risks/Assumptions

16.0 Tools

17.0 Approvals

**1.0 INTRODUCTION**

This document outlines the testing approach, objectives, tasks, and strategies for ensuring the quality and functionality of the online retro games store website. The primary goal is to guarantee a seamless user experience, adherence to specifications, and compliance with industry standards.

**2.0 OBJECTIVES AND TASKS**

**2.1 Objectives**

Ensure all website functionalities meet specified requirements.

Confirm the user interface is intuitive and user-friendly.

Validate performance and responsiveness under various conditions.

Assess security measures effectiveness.

Ensure compatibility across different devices and browsers.

Ensure compliance with industry standards and regulations.

**2.2 Tasks**

Develop comprehensive test cases covering all aspects of the website.

Execute tests based on predefined test cases.

Report and track defects throughout the testing phase.

Collaborate with developers and stakeholders to address issues.

Validate fixes and perform retests as necessary.

**3.0 SCOPE**

Testing will cover all features and functionalities of the online retro games store website, including but not limited to:

User registration and authentication

Game browsing and searching

Shopping cart functionality

Checkout process

Payment processing

Account management

Performance across devices and browsers

**4.0 TESTING STRATEGY**

**4.1 Alpha Testing (Unit Testing)**

Conducted by developers to test individual components.

Focus on code-level testing to ensure each unit functions correctly.

**4.2 System and Integration Testing**

Verify interactions between different modules and systems.

Ensure seamless integration and communication between components.

**4.3 Performance and Stress Testing**

Assess website performance under normal and peak loads.

Identify and optimize performance bottlenecks for scalability.

**4.4 User Acceptance Testing**

Involve end-users to validate the website meets their requirements.

Focus on usability, accessibility, and overall satisfaction.

**4.5 Batch Testing**

Validate bulk operations such as batch processing orders.

**4.6 Automated Regression Testing**

Implement automated tests to ensure new updates do not affect existing functionality negatively. **4.7 Beta Testing**

Release a beta version to a limited audience for real-world feedback.

**5.0 HARDWARE REQUIREMENTS**

Standard server hardware configurations for hosting.

Devices representative of target audience for client-side testing.

**6.0 ENVIRONMENT REQUIREMENTS 6.1 Main Frame**

Web servers (e.g., Apache, Nginx)

Database servers (e.g., MySQL, PostgreSQL) **6.2 Workstation**

Desktop and mobile devices for testing

Supported browsers (e.g., Chrome, Firefox, Safari)

**7.0 TEST SCHEDULE**

Week 1: Develop test cases

Week 2-3: Alpha testing

Week 4-5: System and integration testing

Week 6: Performance and stress testing

Week 7-8: User acceptance testing

Ongoing: Automated regression testing

**8.0 CONTROL PROCEDURES**

Version control for test cases and documentation. Change management process for handling updates.

**9.0 FEATURES TO BE TESTED**

All features outlined in the project requirements document. **10.0 FEATURES NOT TO BE TESTED**

Features explicitly excluded from the project scope.

**11.0 RESOURCES/ROLES & RESPONSIBILITIES** Testers: Execute tests and report defects.

Developers: Fix reported defects and implement changes.

Project Manager: Oversee testing process and coordinate with stakeholders.

**12.0 SCHEDULES**

Test execution schedule outlined in section 7.0.

**13.0 SIGNIFICANTLY IMPACTED DEPARTMENTS (SIDs)** Development team

Customer support

**14.0 DEPENDENCIES**

Availability of development resources for fixing defects.

Timely feedback from stakeholders during user acceptance testing.

**15.0 RISKS/ASSUMPTIONS**

Risks: Potential delays due to technical issues or changes in requirements. Assumptions: Availability of required resources.

**16.0 TOOLS**

Test management tools (e.g., Jira, TestRail)

Automation testing tools (e.g., Selenium, Cypress)

**17.0 APPROVALS**

Specify the names and titles of all persons who must approve this plan. Provide space for the signatures and dates.

Vasylchuk Oleksandr + 18.04.24

Bukevych Illia + 18.04.24

Budnik Serhii + 18.04.24

Yednak Ivan + 18.04.24