# Software Test Plan

## Pokémon

Tharaa Abu Saleh

# CONTENTS

## 1. Introduction

### a. Document overview

The Software Test Plan (STP) presented here delineates the testing strategies, goals, and methodologies for Quality Assurance (QA) automation testing of the Pokémon API. The Pokémon API serves as a vital platform for retrieving information about Pokémon species, abilities, moves, and other related data for developers and enthusiasts alike.

### b. Project overview

The Pokémon API plays a crucial role in providing accurate and up-to-date information about Pokémon, enabling developers to integrate this data into their applications seamlessly. This STP aims to ensure the effective functioning of the Pokémon API, guaranteeing a reliable experience for users. It outlines the testing scope, defines objectives, and establishes the testing approach to be adhered to throughout the QA automation testing process.

### c. project references

The purpose of this STP is to:

Define the scope and objectives of QA automation testing for the Pokémon API.

Identify the roles and responsibilities of the testing team involved in API testing.

Outline the testing approach and methodologies, focusing on API testing practices.

Specify the test environment, encompassing hardware, software, and network configurations relevant to API testing.

Detail the test deliverables, including API test cases, test scripts, and defect reports.

Establish a timeline for testing activities, ensuring a systematic and efficient testing process.

Define metrics for measuring the effectiveness and success of the Pokémon API testing process.

## 2. Test Strategy

Verify the functionality of key Pokémon API features, including Pokémon search, details retrieval, abilities, moves, evolution chains, and adherence to API standards.

### a. Test Objectives

i. Confirm Pokémon Search: Validate the API's ability to search for Pokémon based on various parameters (e.g., name, type).

ii. Retrieve Pokémon Details: Ensure accurate retrieval of comprehensive information for individual Pokémon, covering attributes like abilities, moves, and evolution details.

iii. Validate Abilities and Moves: Verify the correctness and completeness of data related to Pokémon abilities and moves provided by the API.

iv. Compliance with API Standards: Ensure the Pokémon API adheres to industry standards, including proper response formats, status codes, and error handling.

v. Usability Check: Assess the usability of the Pokémon API, focusing on developer-friendly documentation and ease of integration.

vi. Data Consistency: Ensure data consistency across various Pokémon API endpoints, avoiding discrepancies in information representation.

### b. Test Assumption

i. Data Accuracy: Pokémon details, abilities, moves, and other information provided by the API are assumed to be accurate and up-to-date.

ii. Search Relevance: The API's search functionality is assumed to yield relevant and accurate results for Pokémon based on parameters like name and type.

iii. Timely Data Updates: Pokémon API content, including Pokémon details and attributes, is assumed to be regularly updated to maintain relevance.

iv. Data Consistency: Consistency is assumed across various Pokémon API endpoints, avoiding discrepancies in information representation.

v. Usability: The Pokémon API is assumed to have a developer-friendly design and documentation, facilitating ease of integration.

## c. Test Types

i. Functional Testing:

- Positive Testing:Verify that the Pokémon API functions correctly under normal, expected conditions.
- Negative Testing: Assess how well the API handles invalid inputs, unexpected scenarios, and error conditions.

This comprehensive set of test types ensures thorough examination of the Pokémon API, covering functional correctness, performance, security, usability, compatibility, and other critical aspects of API testing.

## d. Test Approach

The testing approach for the Pokémon API will be a balanced combination of manual and automated testing methodologies. Manual testing will be employed for exploratory testing, evaluating usability, and scenarios where human judgement is crucial. Automated testing will be utilized for regression testing, performance testing, and repetitive tasks to enhance efficiency and reliability.

## e. Data Approach

Test data for Pokémon API testing will be meticulously chosen to cover a wide range of scenarios, including edge cases and boundary conditions. Where applicable, mock data will be utilized to facilitate comprehensive testing of specific functionalities, ensuring a thorough examination of the API's capabilities.

## f. Levels of Testing

Conduct unit testing to verify individual API endpoints, followed by integration testing to ensure seamless component interactions. System testing will assess the Pokémon API as a whole, and user acceptance testing will involve real users to validate its usability and meeting expectations.

Test Deliverables

| Deliverable Name | Author | Review |
|---|---|---|
| Test Plan | Test Lead | Project Manager |
| Functional and Non-Functional Test  Cases and STD | Test Team | Test Lead |

## 3. Execution Strategy

### a. Entry and Exit Criteria

Entry Criteria for Pokémon API Testing:

1. Understanding Requirements: Clear understanding of Pokémon API project requirements and functionalities.
2. Environment Ready: Testing environment set up, including servers, databases, and required configurations for API testing.
3. Test Data Prepared: Relevant and diverse test data, including Pokémon species, abilities, and moves, is prepared and available for testing scenarios.
4. Test Documentation Ready: Test cases and scripts are documented, outlining various test scenarios for Pokémon API testing.
5. Testing Tools Installed: API testing tools, including Postman or Insomnia, are installed and configured.
6. Team Availability: Testing team, developers, and necessary stakeholders are available for collaboration during Pokémon API testing.

Exit Criteria for Pokémon API Testing:

1. Test Execution Completed: All planned tests, covering positive and negative cases, are executed for Pokémon API testing.

2. Test Summary Report Generated: A comprehensive test summary report is generated, summarising Pokémon API testing activities and results.
3. Performance Goals Met: Performance testing confirms that the Pokémon API meets predefined criteria.
4. Usability and Accessibility Confirmed: Usability and accessibility testing verify that the Pokémon API meets user-friendly and accessibility standards.
5. Acceptance Criteria Satisfied: The Pokémon API aligns with the acceptance criteria defined by stakeholders.
6. Release Ready: The Pokémon API is deemed ready for release based on testing outcomes and stakeholder approval.

### b. Validation and Defect Management

Defects discovered during Pokémon API testing will be meticulously logged in a defect tracking system, prioritized based on their severity and impact on the API's functionality. Comprehensive defect tracking and reporting, including test cases, execution status, and defect details, will be documented and regularly communicated to stakeholders to maintain transparency and facilitate timely resolution.

### c. Defect tracking & Reporting

Test results, including test cases, test execution status, and defects, will be documented and reported regularly to stakeholders.

## 4. SCHEDULE & ESTIMATION

| Task | Objective | Expected Completion Date |
|---|---|---|
| Test Plan | Thorough test plan for validation. | 09/03/2024 |
| Test DesignTest Case and Development | Structured design, case creation, development. | 09/03/2024 |

|  |  |  |
|---|---|---|
| Test Execution phase 1: Get random card | Retrieve accurate information random card. | 09/03/2024 |
| Test Execution phase 2: Open image for random card | Verify opening image for card. | 09/03/2024 |
| Test Execution phase 3: Get the text and card name | Retrieve and verify Pokémon card text. | 09/03/2024 |

## 5. TEST MANAGEMENT PROCESS

### a. Test Management Tools

Test automation will be used to automate repetitive and time-consuming tasks, such as regression testing and performance testing. Selenium WebDriver will be used for automated testing, along with other tools and frameworks as needed.

### b. Role Expectations

| Roles | Name |
|---|---|
| Project Manager | tzahi anidgar |
| Test Lead | tzahi anidgar |
| Testing Team | Thara'a Abu Saleh |

### c. Project Management

- responsible for reviewing and approving the content of the Test Plan, Test Strategy, and Test Estimates. They provide their sign-off to indicate their endorsement and acceptance of these documents.

### d. Test Planning (Test Lead)

- Before beginning the execution, verify that entrance criteria are employed as input.
- Develop test plans and the guidelines to create test conditions, test cases, expected results and execution scripts.

- Inform the testing team about any necessary modifications to the test deliverables or application and specify the expected completion timelines.
- Provide approval to commence the subsequent phase of testing.

e. Test Team

- Develop test conditions, outline test cases, specify expected results, and generate execution scripts.
- Recognize, document, and prioritize defects in accordance with the instructions given by the Test Lead.

## 6. Test Environment

Replicate production environment, including hardware, software, and network configurations. Utilize API testing tools, sample data, and automation frameworks for efficient testing execution.

## 7. Approvals

| Name | Role | Signature |
|---|---|---|
| tzahi anidgar | Project Management | |
| tzahi anidgar | Test Lead | |
| - | Business Analyst | |