Test strategy

1. **Introduction**

This file is a content of test strategy for a web server.

The web server implemented by a Linux executable file and it has data for players, their names and their ids, and it has a simple API request as a gate for this data.

The web server has one way to get the players data, this way is by use HTTP REST API on local port, This API allows quiring database of players, and it composed of some parts:

* Page index
* Username
* Password

And the response is a JSON array, each object composed of a simple JSON object - tuples values:

* player name
* player ID

**1.1. Scope**

**1.1.1. in Scope**

The features that we get validate in this scope are the REST API request and its response.

**1.1.2. out of scope**

Tests that NOT contained in this scope

- Interface testing

- Compatibility testing

**1.2. Quality Objective**

Our objectives in these tests:

* Ensure the Application Under Test conforms to functional and non-functional requirements
* Ensure the AUT meets the quality specifications defined by the client
* Bugs/issues are identified and fixed before go live

**1.3. Roles and Responsibilities**

QA Automation group is responsible on the quality of the web server tests.

1. **Test Methodology**
   1. **Overview**

The selected test Methodology is Waterfall, DEVs has completely developed this web server, and after they finishing QA Automation group verify it.

* 1. **Test levels**

Test levels affected by the:

* + time that allocated for this task because the QA Automation programmer has basic work and this is the secondary.
  + Budget, no budget for it.

The test types that contained in this test strategy are:

* + functional tests
  + security tests
  + performance tests
  + usability tests
  1. **Bug Triage**

The goal of the triage is to:

* + To define the type of resolution for each bug
  + To prioritize bugs and determine a schedule for all “To Be Fixed Bugs’.
  1. **Test Completeness**

Test completeness should do those points

* + 100% test coverage
  + All Manual & Automated test cases executed
  + All open bugs are fixed or will be fixed in next release

1. **Test Deliverables**
   1. **Test cases**
   * Look at test ‘case.doc’ file
   1. **Test planes**
   * Look at ‘test plan.doc’ file
2. **Resource & Environment Needs**
   1. **Testing Tools**
   * Automation Tools
   * Bug Tracking
   1. **Test Environment**

Following software’s are required in addition to client-specific software

* + Docker
  + Pytest 7.2.0 and above
  + Pytest-html 3.2.0 and above

1. **Terms/Acronyms**

|  |  |
| --- | --- |
| API | Application Program Interface |
| AUT | Application Under TEST |
|  |  |