**Password Hashing Application Test Plan**

**Project Name:** Password Hashing Application

**Test Environment:** OS. Windows 10

**Tester:** Valeria Rivas

**Estimate time of execution:** 2 hours

**Manual Testing; Test Technique:** black box; **Test Type:** Functional testing

**Out of Scope:** Stress testing, load testing, performance testing

**Scenarios to Cover:**

|  |  |  |
| --- | --- | --- |
| **No** | **Attempt to:** | **Verify:** |
| 1 | When launched, the application should wait for http connections | http Connection successful. |
| 2 | It should answer on the PORT specified in the PORT environment variable | Specified port working as expected |
| Not specified port, app doesn't work |
| 3 | It should support three endpoints: |  |
|  | A POST to /hash should accept a password. It should return a job identifier immediately. It should then wait 5 seconds and compute the password hash. The hashing algorithm should be SHA512. | Job identifier is returned correctly |
| create a new hash with empty psw should return a new different hash |
| create a new hash with new psw should return a new different hash |
| Create a new hash with a long character psw should return a new different hash |
| create new hash with similar psw showld return a new different hash |
| create new hash with same psw should return same hash |
| create new hash case sensitive psw should return a different hash |
| create new hash with blank space psw should return a different hash |
| create new hash with number psw should return a different hash |
| create new hash with special characters psw should return a different hash |
| create new hash with 1 character psw should return a different hash |
| Use an unsupported endpoint DELETE for a given hash |
|  |  | Use an unsupported endpoint PUT for a given hash |
| 4 | A GET to /hash should accept a job identifier. It should return the base64 encoded password hash for the corresponding POST request. | send the corresponding POST request and get the corresponding hash |
| send a Request hasn't been created and get answer that doesn't exists |
| 5 | A GET to /stats should accept no data. It should return a JSON data structure for the total hash requests since the server started and the average time of a hash request in milliseconds. | stats returns total hash requests correctly |
| send stats with data should return error |
| 6 | The software should be able to process multiple connections simultaneously. | multiple connections simultaneously processed correctly |
| 7 | The software should support a graceful shutdown request. Meaning, it should allow any in-flight password hashing to complete, reject any new requests, respond with a 200 and shutdown. | shutdown application closes correctly |
| create a hash then shutdown application, hash is created correctly and then application closes |
| Close application window, any in-flight password can’t be completed |
| 8 | No additional password requests should be allowed when shutdown is pending. | create a hash then shutdown application and send a new request to create a hash |

**Test cases:**

link

**Quality Gates:**

* 100% Tests were run
* 95% test passed
* 0 high priority bug found

**Bugs Found:**

|  |  |  |
| --- | --- | --- |
|  | **Bug Description** | **Priority** |
| **1** | Unvalid Key Generates Password Hash | low |
| **2** | SHA512 hash is Returned Instead of base64 Encoded Password | high |
| **3** | Average Time in stats is not Accurate | low |
| **4** |  |  |

**Reviewers:**

@TechLead, @QA Lead, @PO

**Author:**

Valeria Rivas