**Gurukula Test Automation Project**

**Purpose:**

The purpose of the document is to create the automation test strategy for Gurukula application. Gurukula application is a java-based application developed for education domain.

**Scope:**

1. **Automation Testing:**

To create a robust framework which can be easily maintainable for the regression testing of Gurukula application.

**Automation Testing Strategy:**

1. **Test Case Identification for Automation**

Exploratory test was performed on Gurukula application to understand the application functionality and created a regression test cases which are suitable for regression testing.

1. **Test data Analysis**

Based on the exploratory test, required test dataset was created for each functionality of the application. The test dataset will check both positive and negative flow application. These are created based on Boundary value and Equivalence partitioning.

1. **Check the tool feasibility**

Selenium tool with java technology is used to build automation framework

**Tools Used:**

* Selenium Webdriver 3.141.59
* TestNG 6.8
* JDK 1.8
* Maven for build management

1. **Test Script Creation**

* By following Page Object model for creation of object repository, created separate page class file for each web page. Parameterizing the test data was done using Testng framework.
* Every functionality is tested with multiple test datasets using Testng data provider function.
* Cross browser testing is implemented.

1. **Test Execution**

* Tests can be triggered in multiples ways
  + Using maven command: **mvn clean test**
  + Using IDE and testng.xml

1. **Results and Defect log creation**

Testng reporting functionality has been implemented for execution report.

Defects are maintained in the excel sheet.

**Deliverables**

**Source code Git hub link**: https://github.com/gopisrikanth1/GurukulaAutomation.git

**Test Cases:**



**Defect List:**

****

**Steps for Execution:**

**Perquisites:**

* Maven should be installed in the system
* Gurkula Application was started with command **“java -jar gurukula-0.0.1-SNAPSHOT.war”**

**Steps:**

1. Clone the project to the local directory from <https://github.com/gopisrikanth1/GurukulaAutomation.git>
2. Extract the archive
3. Open command prompt and navigate to the root folder of the project
4. Give **mvn test**