### **Robot Framework & Selenium E2E Testing Content Checklist**

Use this checklist to make sure you independently learn the needed concepts. During the project phase you will apply these concepts in a case I will provide you. For now, you should submit as an assignment a zip folder with your name and your tests within it (tests you done during your self-learning phase). These test files will be your “learning diary” equivalent. I will download this folder under my own pycharm setup and verify that your tests run properly, and what sort of tests you did develop.

1. **Setup Your Machine:**
   * Install Python
   * Install PyCharm and configure the Python interpreter.
   * Install required libraries:
     + Selenium
     + robotframework
     + robotframework-seleniumlibrary
     + Robot support for keywords (install to pycharm directly)
2. **Robot Basics:**
   * Create and activate a Python virtual environment.
   * Organize a clear folder structure (e.g., tests/, resources/, libraries/).
   * Understand the basic structure of a Robot test file:
     + \*\*\* Settings \*\*\*
     + \*\*\* Variables \*\*\*
     + \*\*\* Test Cases \*\*\*
     + \*\*\* Keywords \*\*\*
3. **Working with Input Elements:**
   * Practice interacting with input boxes using Input Text and Clear Element Text.
   * Learn to select radio buttons with the appropriate keywords.
   * Work with drop-downs and list boxes using keywords like Select From List By Value/Label.
4. **Synchronization and Browser Control:**
   * Apply wait strategies using Wait Until Element Is Visible and other timeout keywords.
   * Ensure proper browser closure with Close Browser. ????? heittelee voltteja
   * Handle alerts, frames, tabs, and browser windows using keywords such as Handle Alert, Select Frame, Switch Window, etc. maximize window
5. **Browser Interaction and Visual Testing:**
   * Master browser control keywords (Open Browser, Go To, Reload Page).
   * Capture screenshots with Capture Page Screenshot or Capture Element Screenshot.
   * Execute mouse operations (clicks, hover, right-click).
6. **Advanced Robot Keywords and Control Structures:**
   * Develop user-defined keywords and organize them in resource files.
   * Implement page scrolling and use loops (:FOR) in your tests.
   * Interact with hyperlinks and handle tables in test cases.
7. **Data-Driven Testing:**
   * Create script-based data-driven tests with variable iteration.
   * Integrate external data (Excel, CSV) to drive test scenarios.
8. **Database Testing:**
   * Connect to a database using appropriate libraries.
   * Execute SQL queries and validate the results within your tests.
9. **Test Structure Enhancements:**
   * Implement setup and teardown routines (Suite/Test Setup, Teardown).
   * Use tags to group tests and facilitate selective execution.
   * Apply the Page Object Model (POM) for scalable test design.
   * Explore parallel test execution and headless browser strategies.
10. **Integration and Continuous Testing:**
    * Integrate Robot Framework tests into a CI/CD pipeline (e.g., Jenkins).
    * Configure Jenkins jobs to run tests and generate detailed reports.
    * Understand how automated tests support continuous integration and deployment.