RPA Testing

RPA testing is an integral part of RPA Software development, which aims to identify and resolve issues as early as possible in the development stages(Therefore, proactive maintenance is what we need to do, and reactive maintenance is what we must avoid.)

There are 3 Categories when RPA to Fail:

1. **Changes in the Application:**

(New Interfaces, New Functionalities, New Buttons etc)

1. **Environment issues:**

(For Example, in the Virtual Machine where the automation runs in production:1.

Browser got updated VM performance issues System updates in the VM etc)

1. **Automation Issues itself:** any kind of issues in your RPA workflow itself. Unable to detect buttons in prod environment, Selectors not working, No Error handling logic etc.

RPA Testing aims in reducing the **Maintenance Cost** (When RPA Project Fails, We need to Fix it)

Allowing us to creates sustainable and scalable RPA

Levels of RPA Testing

RPA Testing in UiPath follows standard software testing practices, which includes unit, integration, system, and acceptance testing levels within the software development lifecycle

1. **Unit Testing:**

Testing workflows individually

1. **Integration Testing:**

Testing the interactions between two sequences in the same automation project

1. **System Testing:**

Developers or testers assessing the behavior of the entire automation

1. **Acceptance Testing:** End-users or UAT(User Acceptance Testing)group interacting with the RPA Project and checking if it meets the Business Needs or Not?

**What is BDD**

* **BDD** stands for **Begavioral- Driven Development**
* BDD has a standard template which follows the **Given-When-Then** structure to describe the behavior of a system.

**Given:** there are items in the shopping cart,

**When:** the user clicks the ‘Checkout’ button

**Then:** they should be directed to the payment page

**NOTE :** here we checking the behavior of the ‘Checkout’ button that how it behaving if we don’t have the items in the shopping cart then it should not go to the payment page.

So, here we checking the behavior of the ‘Checkout’ button

Why You Need Random Test Data?

* The real problem is in collecting and testing data for exception scenario or rarely encountered scenarios during software testing
* Testing teams often encounter privacy requirements, including GDPR(General Data Protection Regulation) and PCI (Payment Card Industry Data Security Standard) compliance.
* Hence, using real production data is not recommended
* The latest trend is to generate synthetic test data
* **Random number generate from an Array :**

NumArray={1,3,5,10}

If we want 1st number we can use , NumArray(0) it returns 1

NumArray(New Random().Next(0,3))