



Dhaka Polytechnic Institute

Tejgaon, Dhaka- 1208

Software Testing & Quality Assurance

Manual & Automation Testing Project

Presented by

Name : Md Yeamin Hossain Fuhad
Roll : 641367
Department : Computer Science & Technology
Semester : 8th
Shift : 1st
Section : A
Session : 2021-22



সফটওয়্যার টেস্টিং এবং কোয়ালিটি অ্যাসুরেন্স (QA) হলো সফটওয়্যার ডেভেলপমেন্ট লাইফ সাইকেলের (SDLC) একটি অপরিহার্য অংশ। এর প্রাথমিক ভূমিকা হল:

- সফটওয়্যারটির কার্যকারিতা, নির্ভরযোগ্যতা, কর্মদক্ষতা এবং ব্যবহারযোগ্যতা যাচাই করা।
- ডেভেলপমেন্ট প্রক্রিয়ার শুরুতেই ত্রুটি বা বাগ শনাক্ত করে উন্নয়ন খরচ ও সময় কমানো।
- এই প্রক্রিয়া নিশ্চিত করে যে চূড়ান্ত পণ্যটি ব্যবহারকারীর চাহিদা এবং নির্ধারিত মানদণ্ড (Specifications) পূরণ করে।
- টেস্টিং ও QA-র মাধ্যমে একটি উচ্চ-মানের, নিরাপদ এবং সুরক্ষিত সফটওয়্যার ব্যবহারকারীর কাছে পৌঁছে দেওয়া সম্ভব হয়।



টেস্টিং ও QA-র মৌলিক ধারণাগুলো বোঝা জরুরি:

- **টেস্টিংয়ের স্তর (Testing Levels):** ইউনিট টেস্টিং, ইন্টিগ্রেশন টেস্টিং, সিস্টেম টেস্টিং এবং ইউজার অ্যাকসেপট্যান্স টেস্টিং (UAT)।
- **টেস্টিংয়ের ধরন (Testing Types):** ফাংশনাল টেস্টিং, নন-ফাংশনাল টেস্টিং (পারফরম্যান্স, লোড, সিকিউরিটি টেস্টিং ইত্যাদি), রিগ্রেশন টেস্টিং এবং স্মোক টেস্টিং।
- **টেস্ট কেস ও টেস্ট স্যুট (Test Case & Test Suite):** টেস্ট কেসে ইনপুট, এক্সিকিউশনের শর্ত এবং প্রত্যাশিত আউটপুট বর্ণনা করা থাকে। একই উদ্দেশ্যে একত্রিত কেসগুলোর গ্রুপই টেস্ট স্যুট।
- **ডিফেক্ট লাইফ সাইকেল (Defect Life Cycle):** একটি বাগ বা ত্রুটি শনাক্ত হওয়া থেকে শুরু করে রিপোর্ট, ফিক্স, যাচাই এবং শেষ পর্যন্ত বন্ধ হওয়া পর্যন্ত যে ধাপগুলো পার হয়।
- **QA এবং QC (Quality Assurance vs. Quality Control):** QA একটি প্রক্রিয়াভিত্তিক (Process-Oriented) প্রতিরোধমূলক (Preventive) কার্যক্রম, যা নিশ্চিত করে ভালো পণ্য তৈরি হবে। অন্যদিকে QC একটি পণ্য-ভিত্তিক (Product-Oriented) কার্যক্রম, যা তৈরি হওয়া পণ্যের ত্রুটি শনাক্ত করে (Detective)।



বিভিন্ন ধরনের টেস্টিংয়ের জন্য বিভিন্ন টুল ব্যবহার করা হয়:

- **টেস্ট ম্যানেজমেন্ট টুল:** যেমন Jira, TestRail, qTest। এগুলো টেস্ট কেস, প্ল্যান, রান এবং বাগ ট্র্যাক করতে সাহায্য করে।
- **অটোমেশন টেস্টিং টুল:** যেমন Selenium (ওয়েব অ্যাপের জন্য), Appium (মোবাইল অ্যাপের জন্য), Cypress, Playwright, Katalon Studio।
- **পারফরম্যান্স টেস্টিং টুল:** যেমন JMeter, LoadRunner, Gatling।
- **API টেস্টিং টুল:** যেমন Postman, SoapUI, Rest-Assured।
- **কন্টিনিউয়াস ইন্টিগ্রেশন/ডেলিভারি (CI/CD) টুল:** যেমন Jenkins, GitLab CI, CircleCI এগুলোতে অটোমেটেড টেস্ট স্ক্রিপ্ট ইন্টিগ্রেট করা যায়।
- **AI-পাওয়ার্ড টুল :** Deepseek, ChatGPT, Claude etc.



একটি সাধারণ সফটওয়্যার টেস্টিং প্রক্রিয়া নিচের ধাপগুলো অনুসরণ করে:

1. **প্রয়োজন বিশ্লেষণ (Requirement Analysis):** টেস্ট করার আগে সফটওয়্যার রিকোয়ারমেন্ট বা ইউজার স্টোরি ভালোভাবে বোঝা।
2. **টেস্ট পরিকল্পনা (Test Planning):** টেস্টের সুযোগ (Scope), উদ্দেশ্য, পদ্ধতি, রিসোর্স এবং সময়সীমা নির্ধারণ। টেস্ট প্ল্যান ডকুমেন্ট তৈরি।
3. **টেস্ট কেস ডিজাইন (Test Case Design):** রিকোয়ারমেন্টের উপর ভিত্তি করে বিস্তারিত টেস্ট কেস তৈরি করা। প্রত্যাশিত ফলাফল (Expected Result) উল্লেখ করা।
4. **টেস্ট এনভায়রনমেন্ট সেটআপ (Test Environment Setup):** টেস্টিংয়ের জন্য আলাদা হার্ডওয়্যার, সফটওয়্যার, নেটওয়ার্ক কনফিগারেশন ইত্যাদি প্রস্তুত করা।
5. **টেস্ট এক্সিকিউশন (Test Execution):** ডিজাইনকৃত টেস্ট কেস অনুযায়ী ম্যানুয়ালি বা অটোমেশন টুলের মাধ্যমে টেস্ট চালানো এবং প্রকৃত ফলাফল (Actual Result) রেকর্ড করা।
6. **বাগ রিপোর্টিং (Defect Reporting):** প্রত্যাশিত এবং প্রকৃত ফলাফলের মধ্যে অমিল পাওয়া গেলে, একটি বাগ বা ডিফেক্ট রিপোর্ট তৈরি করে সংশ্লিষ্ট ডেভেলপার টিমকে জানানো।
7. **রিগ্রেশন ও রিটেস্টিং (Regression & Retesting):** ডেভেলপাররা বাগ ঠিক করার পর, সংশ্লিষ্ট ফাংশন রিটেস্ট করা এবং অন্য কোন ফাংশন ক্ষতিগ্রস্ত হয়েছে কিনা তা দেখতে রিগ্রেশন টেস্ট করা।
8. **টেস্ট ক্লোজার (Test Closure):** সমস্ত প্ল্যান করা টেস্ট সম্পন্ন এবং প্রি-ডিফাইন্ড এক্সিট ক্রাইটেরিয়া (যেমন- কোন ক্রিটিক্যাল বাগ নেই) পূরণ হলে, টেস্টিং সাইকেল শেষ করে রিপোর্ট তৈরি করা।



প্রোগ্রাম/কোড উদাহরণ

```
1 from selenium import webdriver
2 from selenium.webdriver.common.by import By
3 import time
4
5 def test_signup():
6     driver = webdriver.Chrome() # 1. Open the Google Chrome
7     driver.get("#")
8
9     # input first name
10    elem_first_name = driver.find_element("xpath", "#")
11    elem_first_name.send_keys("#")
12
13    # input last name
14    elem_last_name = driver.find_element("xpath", "#")
15    elem_last_name.send_keys("#")
16
17    # input email
18    elem_email = driver.find_element("xpath", "#")
19    elem_email.send_keys("#")
20
21    # input password
22    elem_pass = driver.find_element("xpath", "#")
23    elem_pass.send_keys("#")
24
25    # input confirm password
26    elem_confirm_pass = driver.find_element("xpath", "#")
27    elem_confirm_pass.send_keys("#")
28
29    # click button
30    elem_button = driver.find_element("xpath", "#")
31    elem_button.click()
32
33    time.sleep(10)
```



সফটওয়্যার টেস্টিং এবং কোয়ালিটি অ্যাসুরেন্স কোনো এককালীন কাজ নয়; এটি SDLC-র সাথে সমন্বিত একটি চলমান প্রক্রিয়া। এর সফল বাস্তবায়ন একটি নির্ভুল, দক্ষ এবং ব্যবহারকারী-বান্ধব সফটওয়্যার তৈরি করতে অত্যন্ত গুরুত্বপূর্ণ। ম্যানুয়াল টেস্টিংয়ের পাশাপাশি সঠিক টুল নির্বাচন করে অটোমেশন টেস্টিং প্রক্রিয়াকে আরও দ্রুত ও কার্যকর করা যায়। প্রযুক্তির বিকাশের সাথে সাথে AI ও মেশিন লার্নিং এর সাহায্যে টেস্ট কেস জেনারেশন, অ্যানালিটিক্স এবং প্রেডিক্টিভ টেস্টিংয়ের মতো নতুন দিগন্তও উন্মোচিত হচ্ছে। সর্বোপরি, টেস্টিং এবং QA দলের লক্ষ্য কেবল বাগ খুঁজে বের করা নয়, বরং উন্নয়ন প্রক্রিয়ায় মূল্য যোগ করে চূড়ান্তভাবে একটি বিশ্বস্ত ডিজিটাল পণ্য নিশ্চিত করা।



ধন্যবাদান্তে

Thank You!

Let's Build Quality Software Together

GitHub Link : https://github.com/yeaminhossainfuhad-cloud/SQA_DPI_8th_Project.git

