

A TRAINING REPORT
ON
“Automation Testing with UFT Learning Path”

A Training Report is submitted to
Jaypee University of Information And Technology,
In the partial fulfillment of the requirements for the award of degree of

BACHELOR OF TECHNOLOGY
In
COMPUTER SCIENCE & ENGINEERING

Submitted By
Rahul Soni 171321

Under the guidance of
Mr. Surjeet Singh
Assistant Professor of C.S.E. Department



DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

JAYPEE UNIVERSITY OF INFORMATION AND TECHNOLOGY
(Established under H.P. Legislative Assembly Act No. 14 of 2002)

And Approved by UGC under section 2(f)
Waknaghat, P.O. Waknaghat, The Kandaghat, Distt.

Solan, H.P. - 173 234,
(2017-2021)

INDEX

| CONTENTS | page |
|--|-------------|
| LIST OF FIGURES | 4 |
| CERTIFICATE BY SUPERVISOR | 5 |
| PROJECT REPORT UNDERTAKING | 6 |
| DECLARATION BY CANDIDATE | 7 |
| CONFIDENTIAL INFORMATION | 8 |
| ACKNOWLEDGEMENT | 12 |
| LIST OF ACRONYMS AND ABBREVIATIONS | 13 |
| ABSTRACT | 14 |
| 1. CHAPTER 1 | 15 |
| 1.1. Module 1: Fundamental Testing | 15 |
| 1.2. Objective | 15 |
| 1.3. SDLC – Software Development Lifecycle | 16 |
| 1.4. What is SDLC? | 16 |
| 1.5. SDLC Models | 20 |
| 1.6. STLC | 25 |
| 1.7. Basic Concepts of Testing | 28 |
| 1.8. Terms Related to Testing | 28 |
| 1.8.1. Examples of Bugs in History | 29 |
| 1.9. Testing Types | 30 |
| 1.10. Test Scenario | 31 |
| 1.11. Test Cases | 32 |
| 1.12. Test Report | 32 |
| 1.13. Defect Report | 33 |
| 2. CHAPTER 2 | 34 |
| 2.1. Module 2: Data Source | 34 |
| 2.2. SQL | 34 |
| 2.3. Joins | 35 |
| 2.4. Aggregate Functions | 36 |

| | |
|-----------------------------------|-----------|
| 2.5. XML | 36 |
| 2.6. JSON | 37 |
| 3. CHAPTER 3 | 38 |
| 3.1. Module 3: VB Script | 38 |
| 3.2. Prerequisties | 38 |
| 3.3. Syntax | 38 |
| 3.4. White Spaces and Line Breaks | 39 |
| 4. CHAPTER 4 | 40 |
| 4.1. Module 4: UFT | 40 |
| 4.2. About UFT | 40 |
| 4.3. History of UFT | 41 |
| 4.4. Object Repository | 41 |
| 4.5. Editor View | 41 |
| 4.6. Keyword View | 42 |
| 4.7. Limitations and Drawbacks | 43 |
| 5. CHAPTER 5 | 44 |
| 5.1. Conclusions | 44 |
| 5.2. Future Scope | 44 |
| 6. REFERENCES | 45 |
| 7. APPENDIX | 65 |

LIST OF FIGURES

1. SDLC
2. SDLC
3. Waterfall Model
4. Spiral Model
5. Iterative Model
6. V-Model
7. Big Bang Model
8. Agile Model
9. Incremental Model
10. RAD Model
11. Typical RAD Model
12. Prototyping Models
13. STLC
14. STLC Phases
15. Test Levels
16. Failure
17. Test Scenario
18. Application UI
19. Test Cases
20. Test Report
21. Defect Report
22. Inner Join
23. Left Join
24. Right Join
25. Full Join
26. Self Join
27. Aggregate Functions
28. XML
29. JSON
30. VB Script Example 1
31. VB Script Example 2
32. UFT UI

33. Object repository
34. Editor View
35. Keyword View
36. Test Scenarios
37. Test Cases
38. Application of XML
39. Table for JSON
40. Application of JSON
41. Application of VB Script
42. Application of UFT
43. Output of UFT

JAYPEE UNIVERSITY OF INFORMATION AND TECHNOLOGY

**(Established under H.P. Legislative Assembly Act No. 14 of 2002 and Approved by UGC
under section 2(f))**

(An ISO 9001-2008 Certified Institution)

Department of Computer Science & Engineering



CERTIFICATE

This is to certify that the project work entitled as “**Automation Testing with UFT Learning Path**” is being Submitted by RAHUL SONI 171321 in the partial fulfillment for the award of the Degree of **Bachelor of Technology** in “**COMPUTER SCIENCE & ENGINEERING**” in the academic during 2017-2021.

Under the esteemed Guidance of

Mr. SURJEET SINGH

ASSISTANT PROFESSOR JUIT

SOWNDHARIYA RAVICHANDRAN

GENC IMPLEMENTATION COGNIZANT

PROJECT REPORT UNDERTAKING

I Mr. **RAHUL SONI** Roll No. **171321** Branch **COMPUTER SCIENCE AND ENGINEERING** is doing my internship with **COGNIZANT TECHNOLOGY SOLUTIONS** from **27TH FEB. 2021** to **11TH JUNE 2021**.

As per procedure I have to submit my project report to the university related to my work that I have done during this internship.

I have compiled my project report. But due to COVID-19 situation my project mentor in the company is not able to sign my project report.

So I hereby declare that the project report is fully designed/developed by me and no part of the work is borrowed or purchased from any agency. And I'll produce a certificate/document of my internship completion with the company to TnP Cell whenever COVID-19 situation gets normal.

Signature 

Name Rahul Soni

Roll No. 171321

Date 15/05/2021

External Examiner

Candidate's Declaration

I hereby declare that the work presented in this report entitled “ **Training in Automation Testing with UFT Learning Path**” in partial fulfilment of the requirements for the award of the degree of **Bachelor of Technology in Computer Science & Engineering** submitted in the Department of Computer Science & Engineering and Information Technology, Jaypee University of Information And Technology, Wagnaghat is an authentic record of my own work carried out over a period from January 2021 to May 2021 under the supervision of **Mr Surjeet Singh (JUIT)** and **Sowndhariya Ravichandran (Cognizant)**.

The work done embodied in the report has been appeased for the award of any other degree or diploma.



Student Signature

Rahul Soni

171321

This is to certify that the above affirmation made by the candidate is true to the best of my knowledge.

Mr Surjeet Singh



Assistant Professor

Department Of Computer Science and Engineering And Information Technology

Dated:

CONFIDENTIAL INFORMATION AND INVENTION ASSIGNMENT
AGREEMENT COGNIZANT TECHNOLOGY SOLUTIONS INDIA
PRIVATE LIMITED

The logo for Cognizant, featuring the word "Cognizant" in a white, sans-serif font centered on a solid blue rectangular background.

As a condition of my employment or my continuing employment with Cognizant Technology Solutions India Pvt. Ltd (together with its affiliates “Cognizant”);, and in consideration of my employment or continuing employment with Cognizant and my receipt of salary and other compensation, training, access to confidential information and relationships with vendors and customers, the benefit of Cognizant’s goodwill, and the other valuable consideration I am receiving and will receive in the future, I agree to the following provisions of this Cognizant Confidential Information and Invention Assignment Agreement (this “**Agreement**”):

1. INVENTIONS.

A. Definitions.

“**Invention**” means any invention, development, concept, know-how, improvement, technique, design, trade secret, idea, data, discovery, mask work, writing, photo, art item, audio or video recording, software item, source code, documentation, or other original work of authorship, and the like, and all copies, improvements, enhancements, derivative works and tangible embodiments of the foregoing, whether or not patentable, subject to trademark protection, or registerable under copyright or similar laws.

“**IP Rights**” means the following types of intellectual property rights under the laws of

(1) any jurisdiction in the world: (a) rights associated with works of authorship, including but not limited to copyrights, design rights, moral rights and mask work rights; (b) trademarks, service marks, trade name rights and similar rights; (c) trade secret rights; (d) patent and industrial property rights; (e) other proprietary rights in and to Inventions and intellectual property of every kind; and (f) rights in or relating to applications, registrations, renewals, extensions, combinations, divisions, amendments and reissuances of or for any of the foregoing rights.

A. My Obligations Relating to Cognizant Inventions. I agree to promptly disclose all Cognizant Inventions to the Cognizant IP Team at IP@cognizant.com. I will cooperate with Cognizant, both before and after my employment ends, in signing documents, providing information, and doing other things reasonably requested by Cognizant to secure, protect, enforce, and transfer to Cognizant all IP Rights in Cognizant Inventions. If Cognizant is unable for any reason to secure my signature to apply for or to pursue any application for any Indian or foreign patent or copyright or similar registrations covering any Cognizant Invention, I hereby appoint Cognizant and its duly authorized officers and agents as my agent and attorney-in-fact to act for and on my behalf to sign and file applications and other documents and to do all other lawfully permitted acts to further the prosecution and issuance of patent, copyright or similar registrations for Cognizant Inventions, which will have the same legal force and effect as if I had signed them.

B. Inventions Created Before My Employment with Cognizant. I understand that Cognizant is not seeking to own any Invention that I Created before I began my employment with Cognizant, whether it was Created while I was working alone or with

2

others (“**Personal Inventions**”), except for any Invention that I Created while I was working: (1) under a contract that gave ownership of the Invention to Cognizant; or (2) for an Acquired Company, if the Invention is owned by, or I was obligated to assign it to the Acquired Company, or if the Invention met any one or more of the four conditions listed in Section 1.B in relation to the Acquired Company. Inventions meeting either of these two exceptions are considered to be Cognizant Inventions.

To help avoid any future confusion or dispute as to what Inventions should be considered to be my Personal Inventions, I have provided below a non-confidential description of all of my Personal Inventions that I own as of the date I sign this Agreement, whether they are owned solely by me or jointly with others, and that could relate in any way to the actual or anticipated business, research or development activities of Cognizant. If I write “none” or leave the lines below blank then I am certifying that I do not have any ownership rights in any Personal Inventions or that my Personal Inventions do not relate in any way to the actual or anticipated business, research or development activities of Cognizant

I represent and warrant to Cognizant that none of my obligations relating to my Personal Inventions will materially affect my ability to perform my work obligations to Cognizant.

A. Restriction on Using My Personal Inventions in My Job. I agree not to incorporate any of My Personal Inventions into any Cognizant Invention or otherwise use any of My Personal Inventions in the course of my job responsibilities for Cognizant, unless I have obtained express prior written approval to do so from the Cognizant IP Team at IP@cognizant.com. I further agree not to incorporate any other Inventions into any Cognizant Invention, or otherwise use any other Inventions in the course of my job responsibilities for Cognizant, unless the Invention is owned by Cognizant or properly licensed to Cognizant for such purpose. If I do incorporate any of My Personal Inventions or any other Inventions that are not owned by Cognizant (and not licensed to Cognizant for such purpose) into any Cognizant Invention or otherwise use any of them in the course of my job responsibilities for Cognizant in such a way that Cognizant or any of its customers might be liable for infringement or misappropriation (which I acknowledge would be a violation of this Agreement), then I hereby grant Cognizant a nonexclusive, royalty-free, paid-up, irrevocable, perpetual, transferable, worldwide license (with rights to grant and authorize sublicenses through multiple tiers of sub-licensees) to make, have made, use, import, offer for sale, sell, reproduce, distribute, modify, adapt, prepare derivative works of, display, perform, and otherwise exploit such Inventions, without restriction, and to practice any method related thereto.

A. Disputes Regarding Ownership of Inventions. I understand that if there is any dispute about the ownership of an Invention, Cognizant will be deemed the rightful owner unless I can provide conclusive written evidence that the Invention qualifies as one of My Personal Inventions.

Disclosures made in Section 1.E or other verifiable records such as a publication date or a filing date with a patent office that precedes the commencement of my employment with Cognizant or an Acquired Company, for example, may serve as evidence of ownership of My Personal Inventions.

B. Customer Inventions. If I had been deputed to any specific project(s) of customers during the course of my employment with Cognizant (whether directly or indirectly), I agree that subsequent to my employment with Cognizant I will fully, punctually and to the satisfaction of Cognizant and such customers comply with the rules and requirements relating to Invention, non-disclosure, confidentiality, security checks, policies and procedures or any other related matters as may be prescribed by such customers.

My agreement, and my acknowledgement of a receipt of a copy of this Agreement, **I know that sharing copy of private information of Cognizant might lead to legal action against me or the Demanding authority, and I accept all the terms and agreement which are indicated by my signature below.**

A handwritten signature in black ink that reads "Rahul Soni". The signature is written in a cursive style with a small dot at the end of the name.

RAHUL SONI
(171321)

ACKNOWLEDGEMENT

I would like to thank you MR. SURJEET SINGH, Assistant Professor at Jaypee University of Information Technology, Solan H.P. He is a wonderful teacher as well as a wonderful Mentor. His guidance always help me to solve my academic problems.

I would like to thank you my coach SOWNDHARIYA, Genc Implement at Cognizant Technology Solutions. She is a wonderful coach as well as a wonderful Mentor. Her guidance always help me to solve my training problems.

To wrap things up, I would Like thank whole Computer Science Department, all My Cognizant colleagues, my friends and my Family for their marvelous support.

LIST OF ACRONYMS AND ABBREVIATIONS

| Abbreviation | Meaning |
|--------------|---------------------------------|
| VBSCRIPT | Visual Basic Script |
| FUNC | Function |
| STRREVERSE | String reverse |
| DISP | Display |
| UFT | Unified Functional Testing |
| QTP | Quick test Professional |
| DBMS | Database Management System |
| XML | Extensible Markup Language |
| SDLC | Software Development Life Cycle |
| STLC | Software Testing Life Cycle |

Automation Testing with UFT Learning Path

ABSTRACT

The main goal of the Automation Testing is to perform testing using special automated testing softwares. In Cognizant during we have worked on the UFT – Unified Functional Testing software and FACETs. I have mainly worked on the UFT which is also known as QTP and used as software testing tools for testing on Web and Mobiles. In the automation testing we write scripts for the functionalities and compare the expected and actual results and according to that we generate test reports. Like Manual testing, In automation testing we are writing test cases and comparing those values with actual values only difference is here we are writing scripts to software tools. In this training we have worked and learned about Manual Testing and its types, Databases, XML, JSON, VBScript and UFT. We use VBScript in QTP/UFT. UFT only support VBScript Language So we can't write Test Script as in any other Language. But UFT is user friendly and provide great technical support. Till Now we have just worked on UFT we have not worked on FACETs yet so I have no idea about FACETs.

KEYWORDS: Automation Testing, UFT, FACETs, Unified Functional Testing, Databases, XML, JSON, VBScript, Test Cases, Manual Testing.

CHAPTER-1

This Report is about my Internship at Cognizant Technology Solutions Corporation. My domain in Cognizant was Automation Testing with UFT. We have 5 Modules for learning and 3 projects:

- Module 1: Fundamental Testing
- Module 2: Data Source
- Module 3: VB Script
- Module 4: UFT Automation

1.1. Module 1: Fundamental Testing

Software Testing is the process in which we identify and check the Quality of the software under several test cases. In Software testing we identify the accuracy of the software. The main purpose of testing is to check that the product fulfills all the requirements and desires of the end user or clients. The goal of the testing is to point out the bugs in the software and help the developers to fix that issue.

1.2. Objective

The main objectives of software testing are as follows:

- To find out as many as bugs and errors in the given application or product as possible.
- Always test that the given application or software fulfills its requirements.
- Always try to validate the quality of a software using the minimum resources, costs and efforts.
- Always try to perform effective tests and always generate high quality of test cases and generate helpful problem reports.

1.3.SDLC – Software Development Lifecycle

SDLC stands for Software Development Lifecycle. It is a process which is used by the companies. In SDLC companies follow the process like designing then development then testing. The main aim of the SDLC is to develop a system or application which is of high quality, fulfils all the client requirement and free of bugs and failure and this application should meet the expectation of the end users with less cost.

- SDLC is known as Software Development Life Cycle.
- It is also called as Software Development Process.
- SDLC is a framework which tells the developers and testers which step they should follow.
- International standard for SDLC or Software lifecycle processes is ISO/IEC 12207. The main aim of this international standard is that it defines all the steps or tasks required for developing, testing and maintaining software or application.

1.4. What is SDLC?

In software projects, SDLC is a process which is followed. SDLC consists of a well maintained and detailed plan which describes how to develop, test, maintain, repair, replace, alter and increase efficiency of specific software and the overall development process.

SDLC contain various stages as shown in the figure below:



Fig. 1

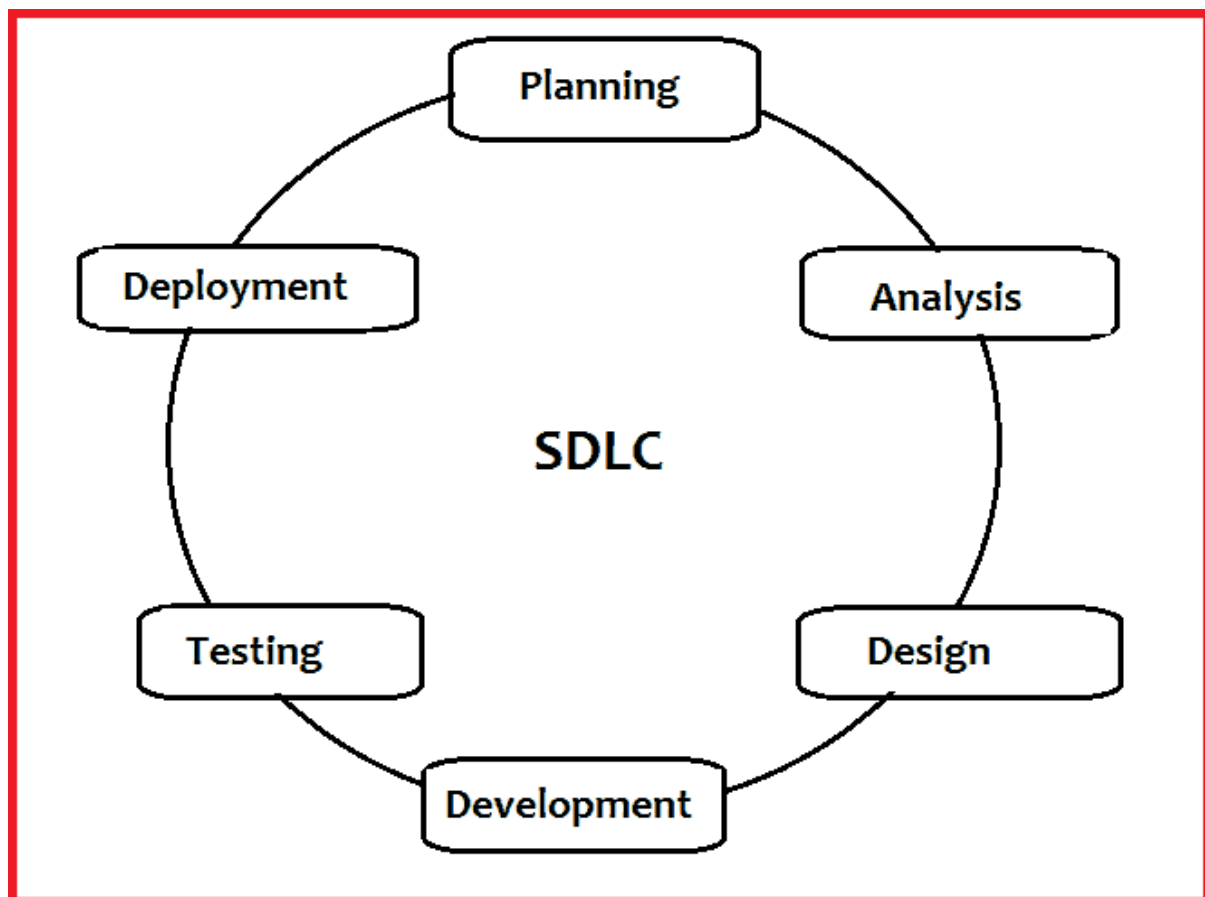


Fig. 2

- **Planning and Requirement Analysis:**

In SDLC, the first and most important section is Planning and requirements analysis stage. This stage is the fundamental and the basic stage of SDLC in which we discuss with our clients and note down all of their requirements. This stage is performed and handled by the Senior members of the team in which they take input and overall requirements of the project or software from the customer. These requirements are then used to plan the basic structure of the project or we can say the basic plan of the project and it also helps in other stages of SDLC.

After requirements, Planning also plays an important role in this stage. In this steps we discuss about the risks associated with the project and this is very important to discuss all the risks related to the project in future. This will help the team to solve and identify the issues in early stage and solve as soon as possible and team can implement a system successfully with minimum error or risks.

- **Defining Requirements:**

As soon as planning and requirement analysis stage is completed the team will move to the next stage which is defining requirements. In this stage we clearly define and document all the specified product requirements and the documented requirements are then sent to the customers or market analysts for the approval. This is done by an Software requirement Specification or SRS document which contains all the requirements of the product which are to designed and developed during the life cycle of the project.

- **Designing the Product Architecture**

As soon as the defining requirement stage is completed, the team will move to the next stage which is designing the product architecture. In this stage team takes the reference of the SRS document for the designing of the product. After this all the design approaches for the product are documented in a Design Document Specification which is also known as DDS. This DDS is reviewed by

all the stakeholders where they take decision and finalize one design based on various parameters such as time constraints and risk assessment.

- **Building or Developing the Product**

As soon as designing is done and reviewed by the stakeholders. The DDS is handed to the team and team will start developing the project according to the requirements and the Design selected by the stakeholders.

Developers will write the code which must follow the organization's guidelines. They have to write code according to the tools they are using to generate the code like the compilers, debuggers and interpreters etc. They will choose the high level programming language according to the software they developing. Developers can work any programming language like C, C++, Java, Python etc according to the software requirements.

- **Testing the Software**

As soon as development part is completed the developer send the product or software to the testers. There are two type of testers mainly: Manual Testers and Automation Testers.

Manual Tester write test cases manually he/she will cover all the possible test cases to test the system and generate the report. If they found some error or bug the product will resend to the developer with the proper report so that developer can easily fix that error.

Second one are Automation testers. They write scripts for the test cases which are out of range from the manual testers. There are lot of automation testing tools in market like UFT, QTP, Seleium. Then the tester execute the test cases and generates a report . If everything is fine the product is ready to deliver but if there will be an error than the product is send back to the developer. The developer will again review that code and will correct the error and will send that code to the tester again for retesting. If the code runs properly after retesting then the product is ready to deliver. The tester will continue the testing of the product unlit and the requirements are meet and errors are left in the product.

- **Deployment in the Market and Maintenance**

When the testing stage is completed the product become eligible for the deployment. The product is released in the open market. But sometimes the product will be released in part its totally depends upon the organizations. If the company follows waterfall model then they release whole product together. But it they follow agile model then the organization will release the product in stages.

In agile model the product is released in stages where the organization take reviews from the users and according to their demands organization update the product and release the newer version of the product.

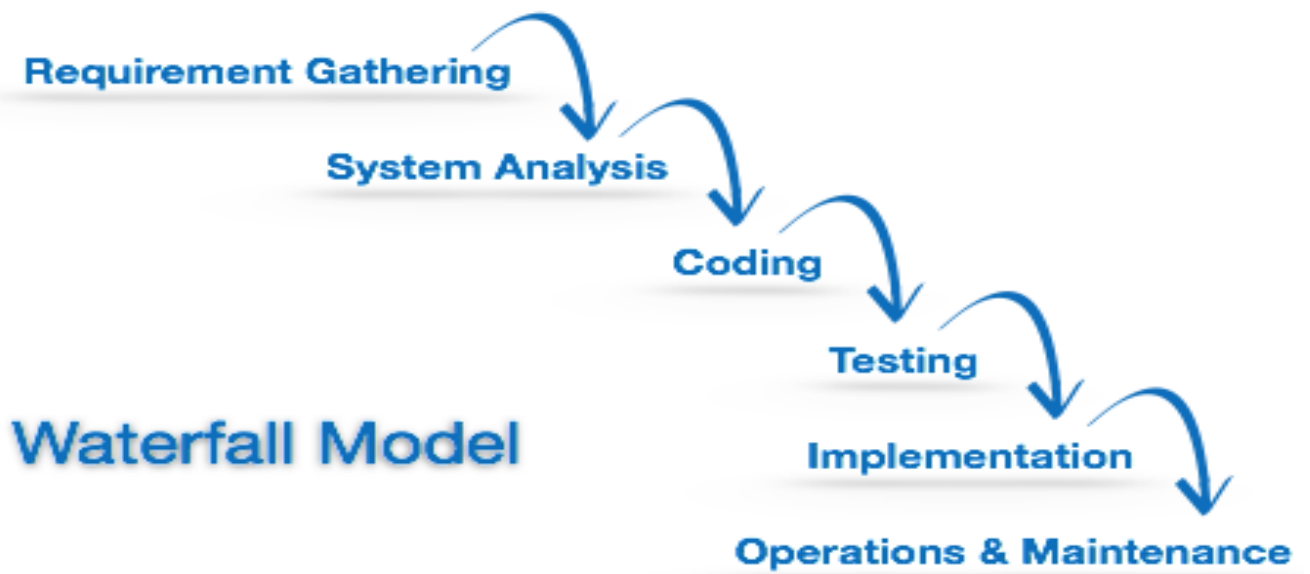
1.5. SDLC Models

In organizations, we use various type of SDLC or Software Development Life Cycle models. There are lot of SDLC models which are designed and defined and they are followed during the development process of any software. These models are also known as Software Development Life Cycle Models. Where each model is different from other model both in working and stages.

Following are the most common, important and popular Software Development Life Cycle Models followed by the organizations:

- **Waterfall Model**

Fig. 3



- Spiral Model

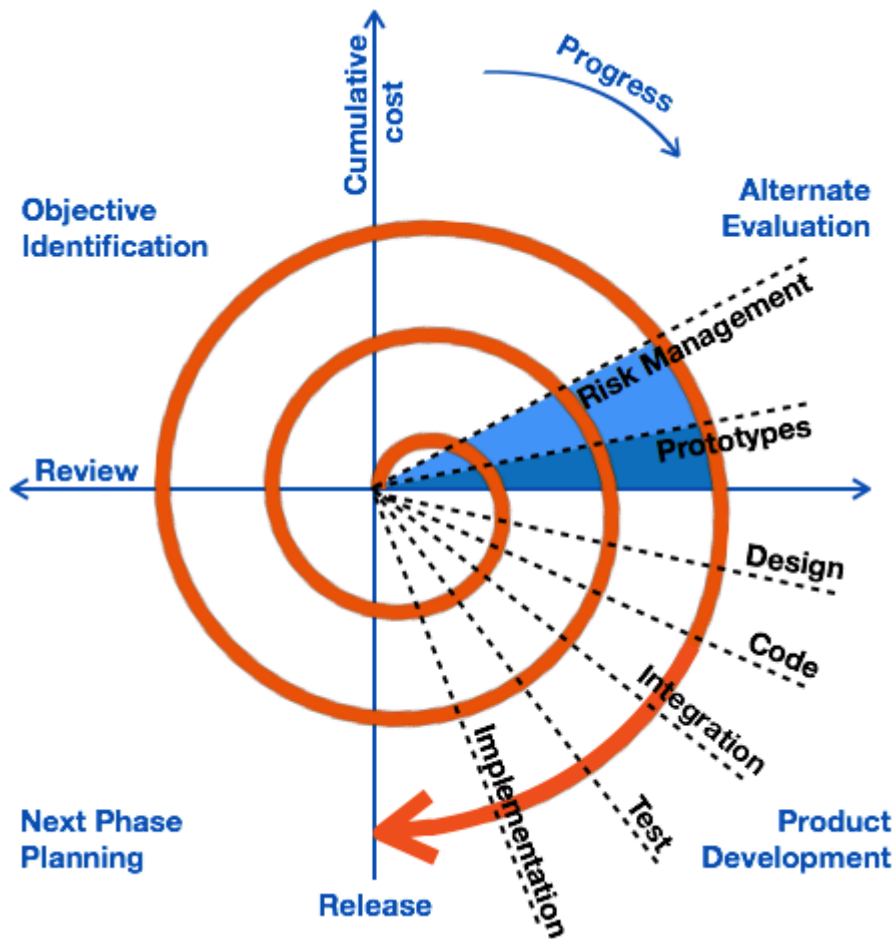


Fig. 4

- Iterative Model

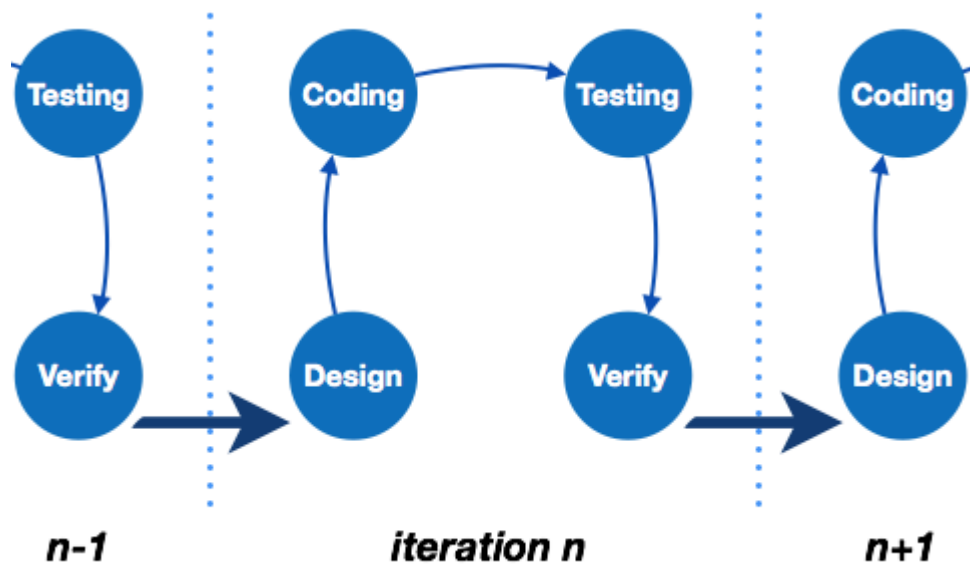


Fig. 5

- V-Model

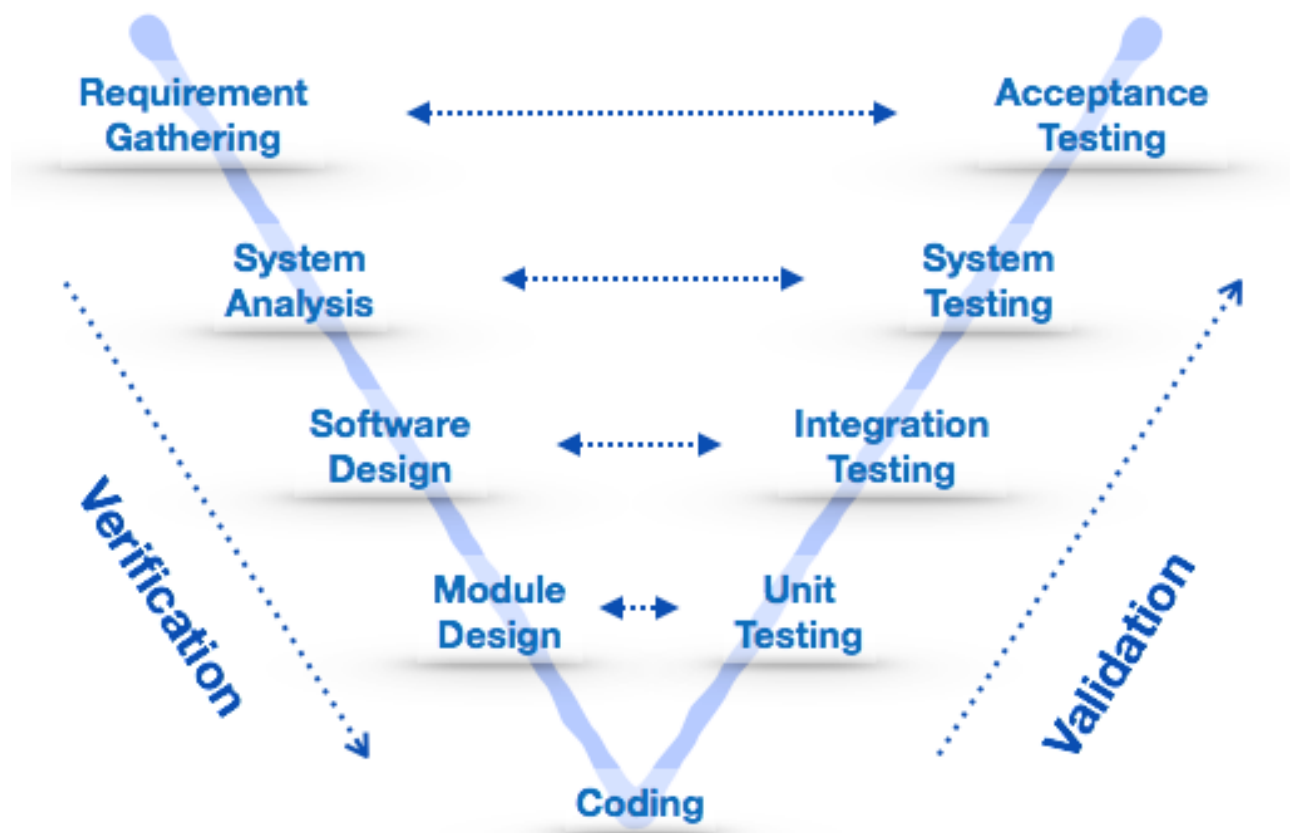


Fig. 6

- Big Bang Model

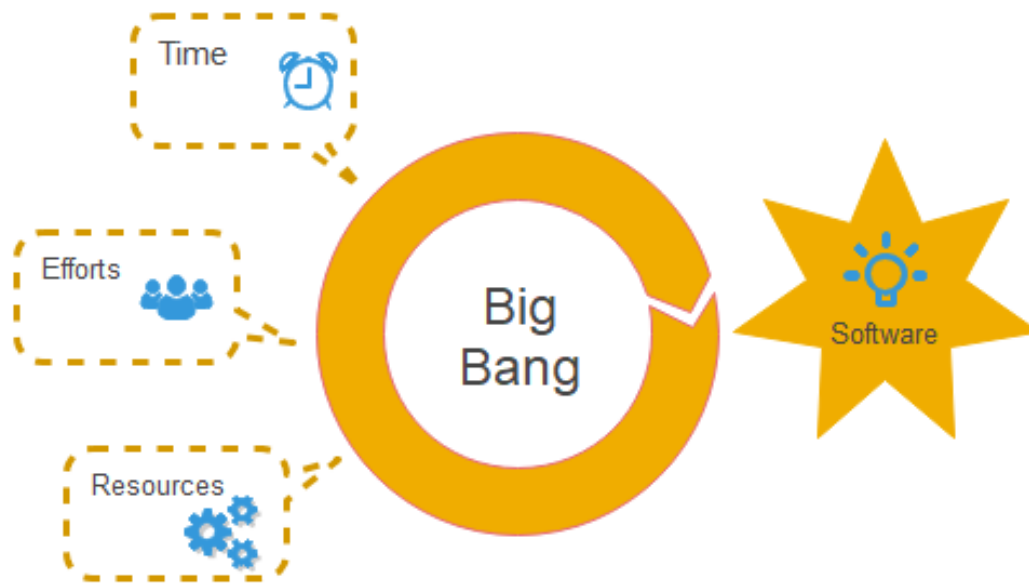


Fig. 7

- **Agile Model**

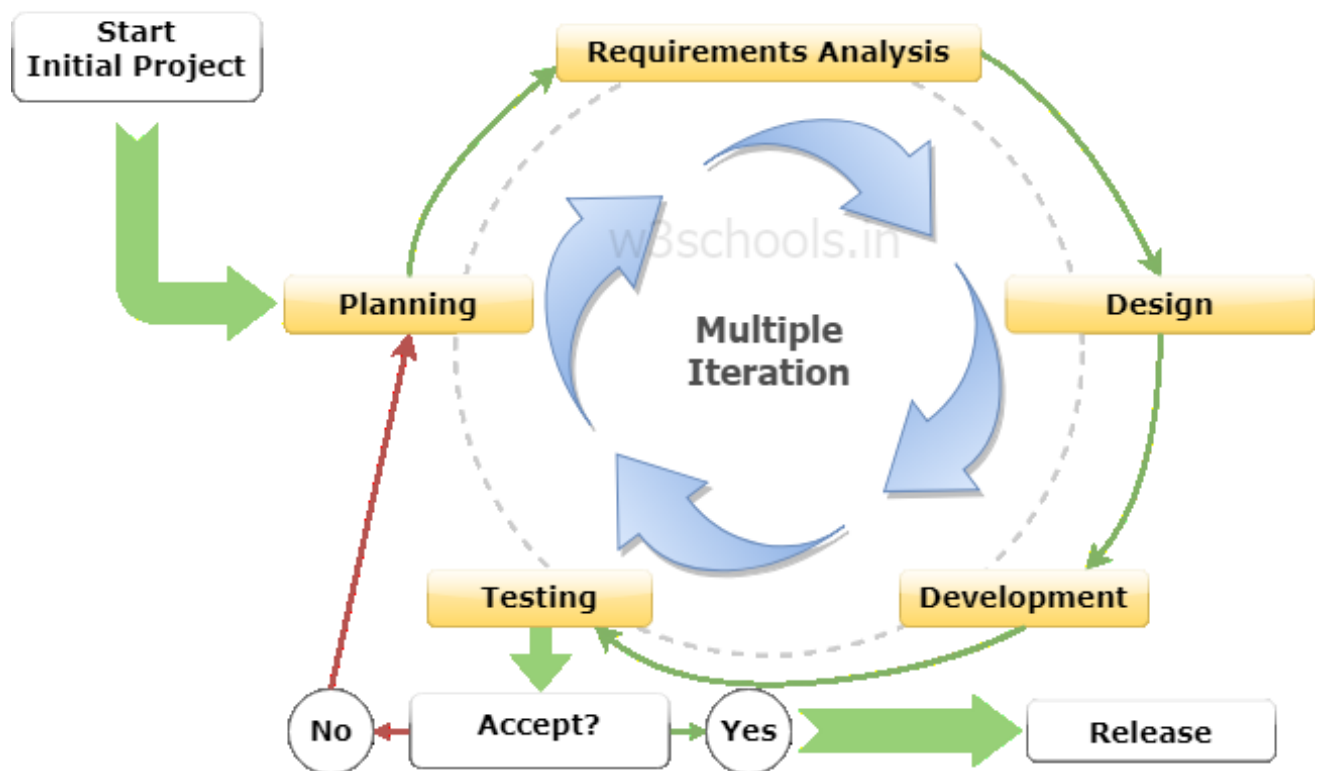


Fig: SDLC Agile Software Development Model

Fig. 8

- **Incremental Model**

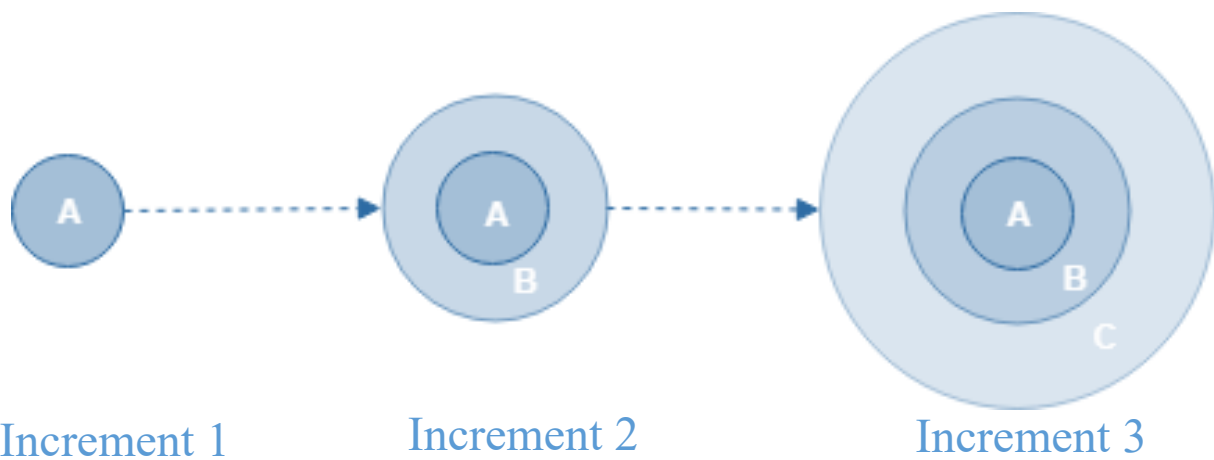
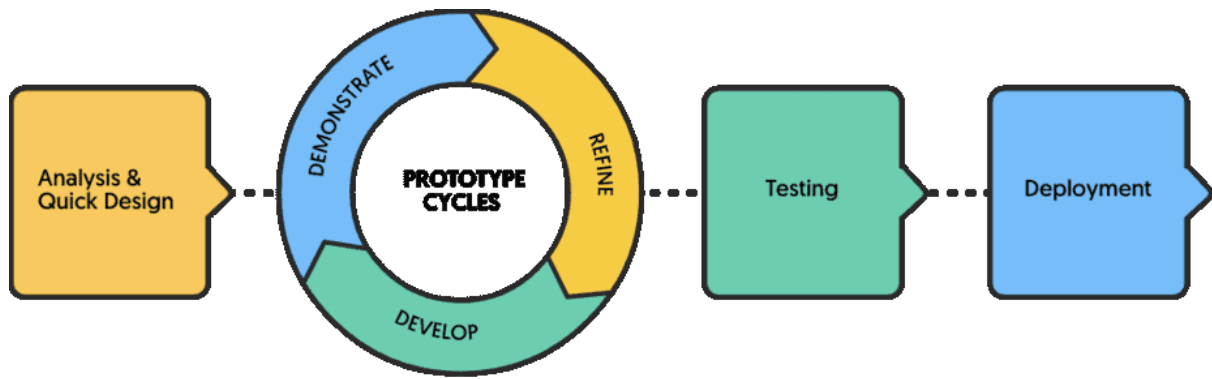
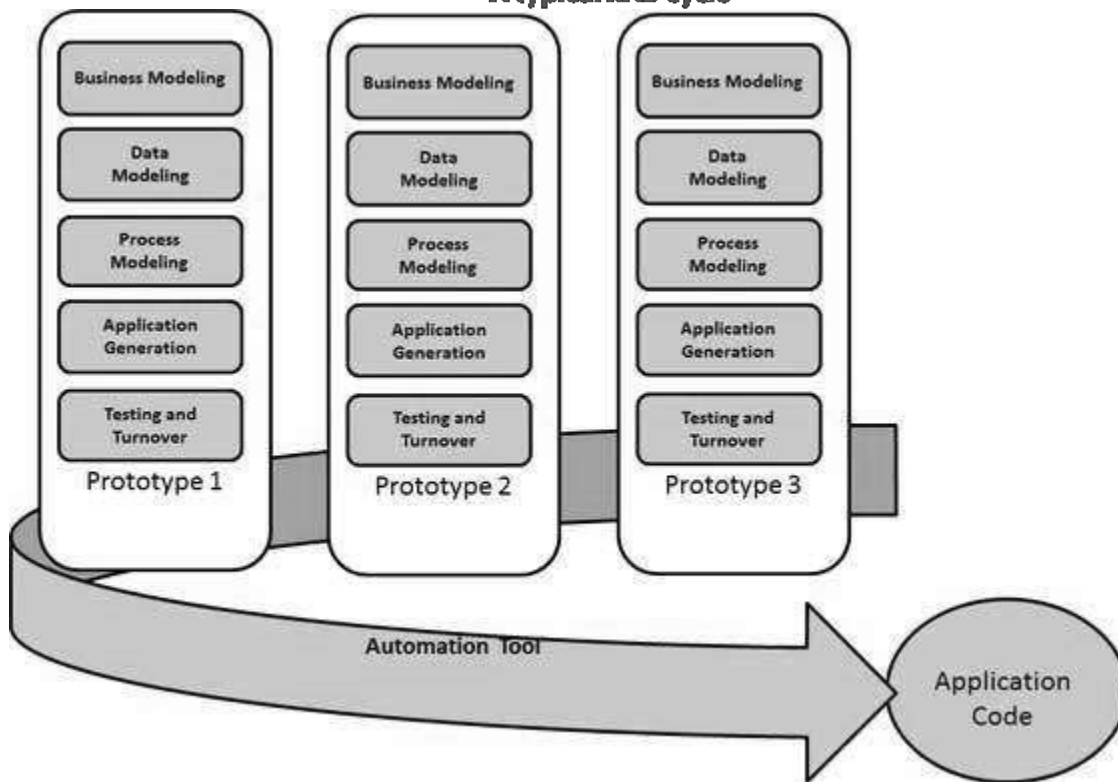


Fig. 9

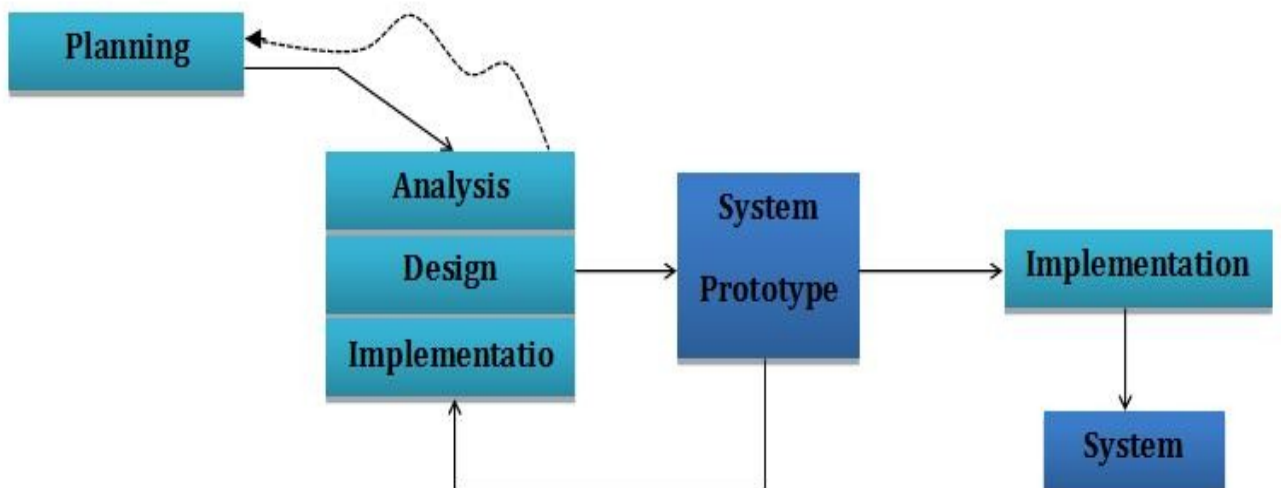
- **RAD Model**



A typical RAD cycle



- **Prototyping Models**



1.6. STLC – Software Testing Life Cycle

STLC stands for Software Testing Life Cycle. It is the part of the SDLC – Software Development Life Cycle. Only difference is that STLC focus only on the Testing part of the SDLC. It is a Sequence in which testing team perform various activities on the software to ensure the quality of that software or the application.

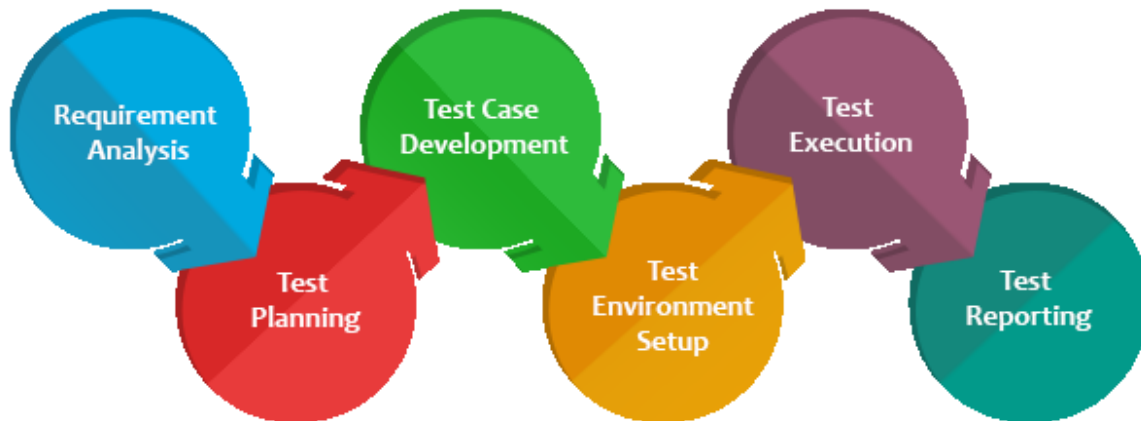


Fig. 13

- As soon as stakeholders share SRD- Software Requirement document the team starts STLC.
- A step by step process is provided by STLC to ensure quality software.
- A tester can analyze and define test case and scope of testing while the developer is developing the software or the product.
- The testers are ready to test the software and perform test cases as soon as development phase get completed. This helps the team to find the bugs and errors at initial phase.

- **STLC Phases:**

Like SDLC, the STLC also have some phases but it is not mandatory that the tester should flow all the phases. According to the nature of the software, resources and time the phases are selected by the testers.

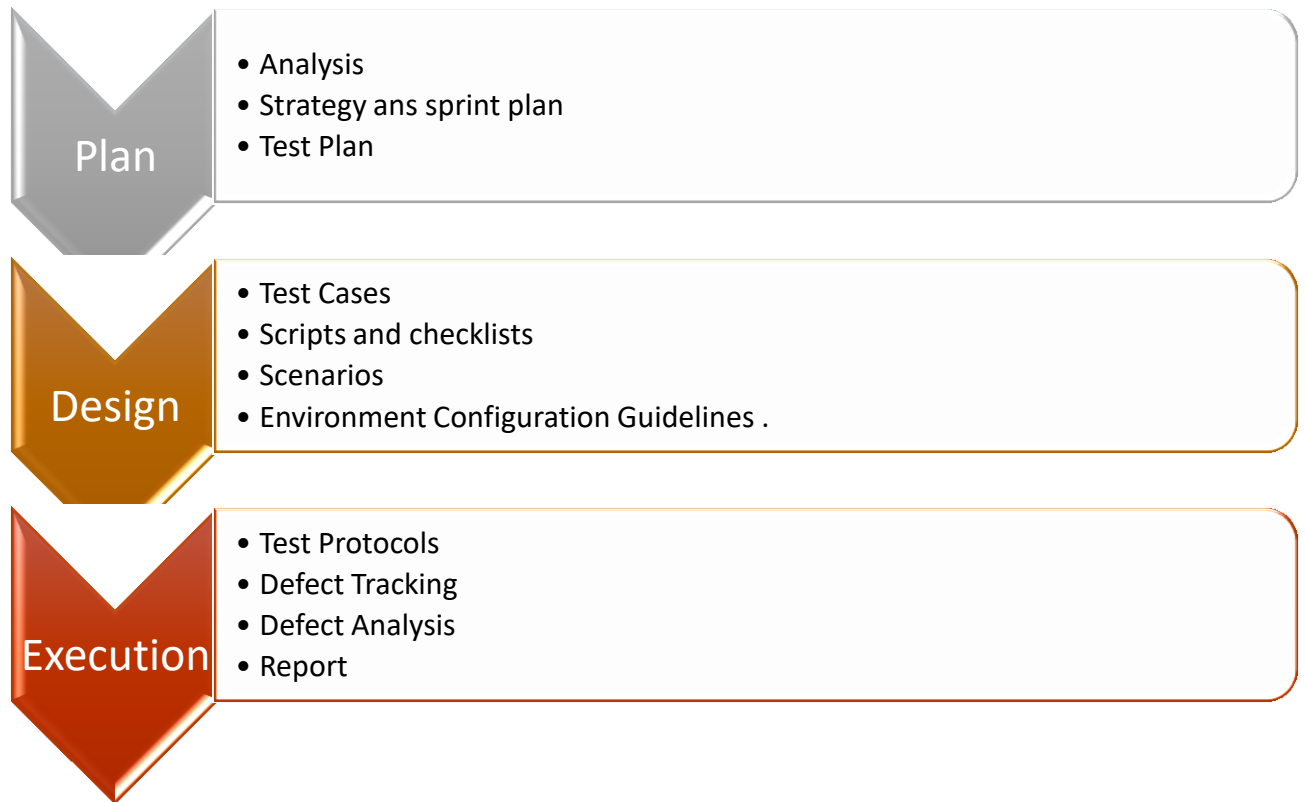


Fig. 14

Mainly there are 6 major phases of Software Testing Life Cycle which testers commonly follow:

1. Requirement Analysis: Testing team analyze the SRD.
2. Test Planning: Testing Team start planning their strategy and approach.
3. Test Case Designing: Designing the test cases based on criteria and scope.
4. Test Environment setup: Using systems validate the softwares.
5. Test Execution: Validating the product in real time and finding the bugs.
6. Test Closure: Generating reports when testing is complete.

- **Test Levels:**

Testing levels are fundamentally to distinguish missing regions. It is used to test different parts of the software or different modules of the software and test the software as combination of those components.

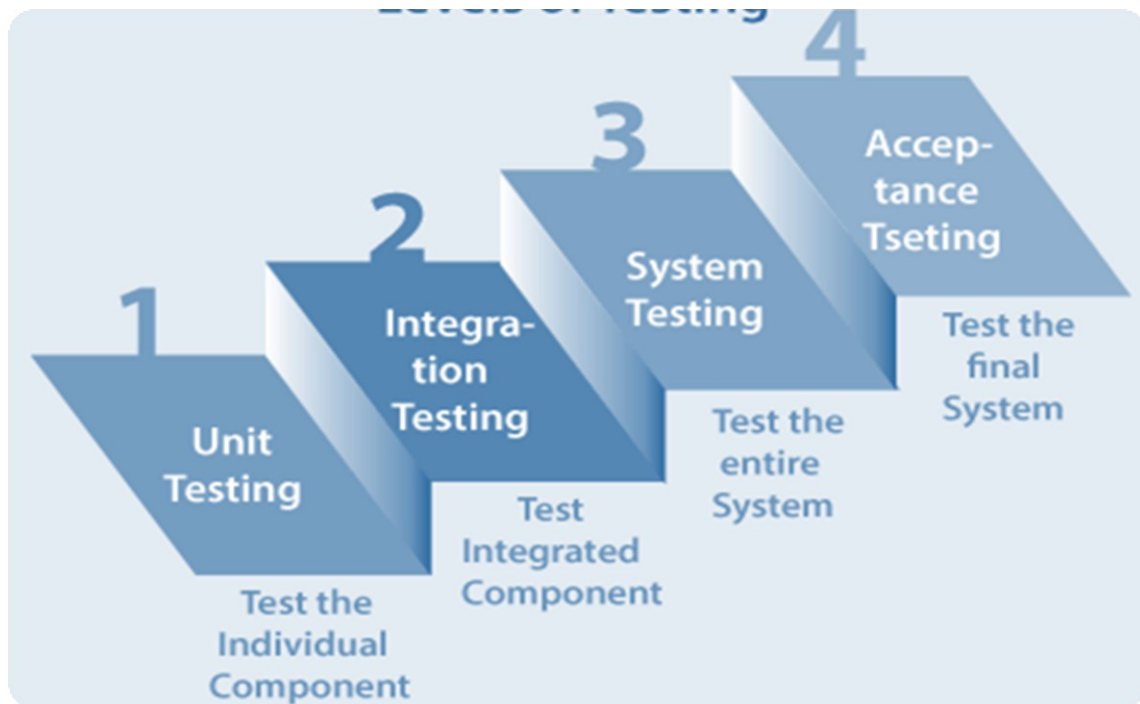


Fig. 15

1. **Unit Testing** – Unit testing is term used for the testing of signal units or modules of the software. In unit testing testers test the modules individually.
2. **Integration Testing** – Integration testing is further divided into two parts first one is Component Integration Testing and second one is System Integration Testing. Component integration testing is done between 2 components it is done to check the integration between two components. It is usually done by the developer. On the other hand, System Integration is done in bigger projects by the testers in which testers check the integration between two systems like frontend and backend.
3. **System Testing** – System testing is the most important type of testing in which we test our software in real environment with the real time test cases. This testing is done to check the integration and stability between different components.

4. **Acceptance Testing** – Acceptance testing is done by the clients or stakeholders. In this they check that the software met their requirements and no failure present in the software.

1.7. Basic Concept of Testing

Software testing is divided into two main processes:

- **Verification:**

Verification is a process of software testing in which we as a tester check that the system or the application we have is aligned according to the requirements of the documentation. We need to verify the requirement of the software or application from the documentation.

- **Validation:**

Validation is a process of software testing in which we as a tester check or verify the accuracy and efficiency of the software or application. In this we actually think from the user point of view. We will verify that what users want from the application and what has been done.

1.8. Terms related to Testing:

- **Error:**

In Simple words an error is a difference between the expected result and the actual result. It is mainly occur due to the mistakes made by the people.

- **Bug:**

In Simple words a bug is just an error which is found in the application or software before goes into the production. It can be a programming error. It produce some incorrect results and make the program to work poorly or sometimes it crash the program or application

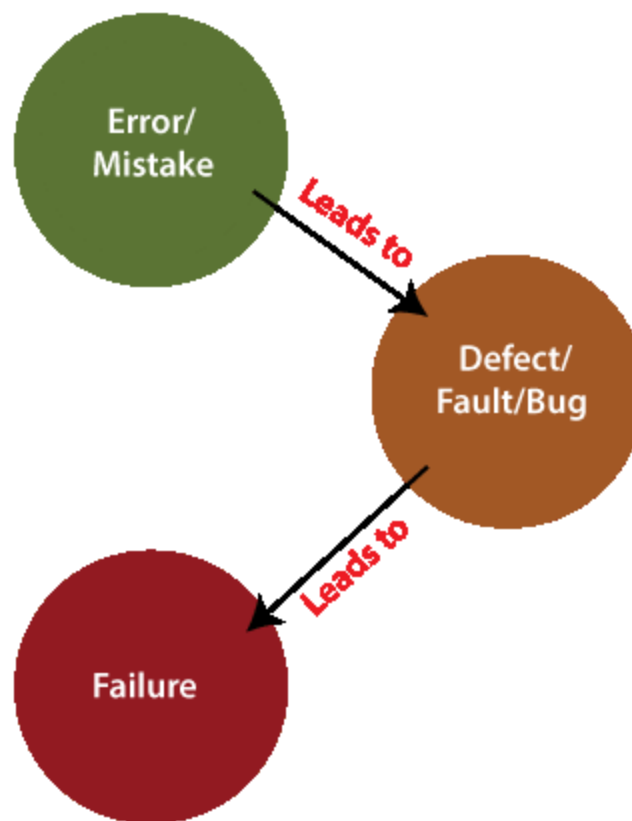
- **Defect:**

In tracking system the error is logged as a defect when it is identified during testing.

- **Failure:**

In Simple words a failure occurs when a system is incapable to conduct its required functions. It is literally a letdown or sometimes a disappointment and no client wants to do work with a failure or no one wants to buy fail product.

Fig. 16



1.8.1. Examples of Bugs in History

History is full of such examples where software bugs caused monetary and human loss. Some examples are:

- In April 2015, Due to a system glitch Bloomberg terminal in London crashed. More than 300,000 traders are affected on financial markets. Because of this crash a 3bn pound debt sale has been postponed by the government.
- Over 1 million cars are recalled by the Nissan Cars from the market due to a software failure in its POS systems.

- In USA and Canada 60% stores of Starbucks were forced to close because of some software failure in its POS system. Their system was unable to process the transactions.
- Vulnerability in Windows 10. This bug enables the Windows users to escape them from the security sandboxes in the Win32k system through a flaw.

1.9. Testing Types

There can be a common of testing. Not many significant ones are recorded underneath:

- **Functional Testing** – Functional testing is an example of black box testing in which we write test cases for the software components and test different functionalities of those components. Functional testing is done by the testing team.
- **Non – Functional Testing** – Non-Functional testing is the testing technique in which we test the non-functional requirements of the software. Non-functional testing can be done by manual tester or by performance testers.
- **Black box Testing** – Black box testing is a method of testing in which we write test cases to verify the functionalities of the software. In black box testing the tester does not have any specific knowledge about the inner structure or code of the software. Tester only enters the input and verifies the output.
- **White box Testing** – White box testing is a method of testing in which we write test cases to verify the functionalities of the software. In white box testing the tester has specific knowledge about the inner structure or code of the software.
- **Dynamic Testing** – Dynamic testing is the testing technique in which the execution of the framework is done.
- **Static Testing** – Static testing is the testing technique in which no execution of the framework is done.

- **Retesting** – Resting is the testing technique in which tester again test the bug after it by the developer.
- **Regression Testing** – Regression testing is the testing technique in which tester test the other components after fixing a bug or error in one component. This testing is done just to check the impact of fixed bug on other components.
- **Smoke Testing** - Smoke Testing is the testing technique in which tester test the basic components of that software to check that the component is working properly with basic requirements or not.

1.10. Test Scenario

| | A | B | C |
|----|--------------------------------|-------------------|------------------------------------|
| 1 | | | |
| 2 | | | Test Scenario Template |
| 3 | | | |
| 4 | * Test Cases field is Optional | | |
| 5 | | | |
| 6 | Test Scenario # | Requiremen | Test Scenario Description |
| 7 | 1 | S1.1 | Check the Login Functionality |
| 8 | 2 | S1.2 | Check the Search Functionality |
| 9 | 3 | S1.3 | Check the Product Description Page |
| 10 | 4 | S1.4 | Check the Payments Functionality |
| 11 | 5 | S1.5 | Check the Order History |

Fig. 17

- **To check the Login functionality:**

Sign in

Email (phone for mobile accounts)

Password [Forgot your password?](#)

Sign in

Keep me signed in. [Details](#) ▾

[New to Amazon?](#)

Create your Amazon account

Fig. 18

1.11. Test Cases

| Project Name | Bank Webiste Functionality | | | | | | | | |
|---|---|-----------|------------------------|------------------|----------------------------------|----------------------------|--------------------------|--------|----------|
| Module Name | Login Functionality | | | | | | | | |
| Created By | Archana | | | | | | | | |
| Created Date | yyyy-mm-xx | | | | | | | | |
| Executed By | XYZ | | | | | | | | |
| Executed Date | YYYY-MM-DD | | | | | | | | |
| Test Case ID | Test Case Description | Pre Steps | Test Step | Preconditions | Test Data | Expected Result | Actual Result | Status | Comments |
| Test the Login Functionality in Banking | Verify login functionality with valid username & password | | Navigate to login page | | | Able to see the login page | As expected | Pass | |
| | | | Enter valid username | Valid Username | username: choudaryac97@gmail.com | Credential can be entered | As expected | Pass | |
| | | | Enter valid password | Valid password | password: XXXXXXX@1 | Credential can be entered | As expected | Pass | |
| | | | Click on login button | | | User logged | User logged successfully | Pass | |
| Test the Login Functionality in Banking | Verify login functionality with valid username & invalid password | | Navigate to login page | | | Able to see the login page | As expected | Pass | |
| | | | Enter valid username | Valid Username | username: choudaryac97@gmail.com | Credential can be entered | As expected | Pass | |
| | | | Enter valid password | Invalid password | password: XXXXXXX@2 | Credential can be entered | As expected | Pass | |
| | | | Click on login button | | | User logged | Unsuccessful login | Fail | |

Fig. 19

1.12. Test report

Test Report

Test Cycle **System Test**

| | | | | | |
|--------------------|---|--|--|-----|-----|
| EXECUTED | PASSED | | | 130 | |
| | FAILED | | | 0 | |
| | <i>(Total) TESTS EXECUTED (PASSED + FAILED)</i> | | | | 130 |
| PENDING | | | | | 0 |
| IN PROGRESS | | | | | 0 |
| BLOCKED | | | | | 0 |
| | <i>(Sub-Total) TEST PLANNED (PENDING + IN PROGRESS + BLOCKED + TEST EXECUTED)</i> | | | | 130 |

| Functions | Description | % TCs Executed | % TCs Passed | TCs pending | Priority | Remarks |
|----------------------|---|----------------|--------------|-------------|----------|---------|
| New Customer | Check new Customer is created | 100% | 100% | 0 | High | |
| Edit Customer | Check Customer can be edited | 100% | 100% | 0 | High | |
| New Account | Check New account is added | 100% | 100% | 0 | High | |
| Edit Account | Check Account is edit | 100% | 100% | 0 | High | |
| Delete Account | Verify Account is delete | 100% | 100% | 0 | High | |
| Delete customer | Verify Customer is Deleted | 100% | 100% | 0 | High | |
| Mini Statement | Verify Ministatement is generated | 100% | 100% | 0 | High | |
| Customized Statement | Check Customized Statement is generated | 100% | 100% | 0 | High | |
| | | | | | | |
| | | | | | | |

Fig. 20

1.13. Defect Report

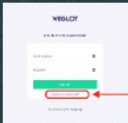
| | |
|--------------------|--|
| Title | Login -> Forgot password button isn't working |
| Steps to reproduce | <ol style="list-style-type: none"> 1. Click on Login 2. Click on Forgot password |
| Expected Result | The button can be clicked and user should be redirected to a page to enter his email |
| Actual result | Clicking on the button doesn't have any impact |
| Test Environment | Samsung Galaxy Note 10 - Android 10 - 4G Network |
| Priority | High |
| Type | Functional |
| Screenshot |  |

Fig. 21

CHAPTER-2

2.1 Module 2 : Data Source

In data source module, we have studied about various concepts of databases. Mainly, we have studied about RDBMS – Relational Database Management system, SQL – Structured Query Language, XML- Extensible Markup Language and JSON- JavaScript Object Notation.

2.2 SQL

SQL stands for Structure Query Language. SQL is used for interaction between data, tables and Databases in RDBMS.

The SQL commands are divided into four categories as mentioned below:

1) DDL – Data Definition Language

Examples of DDL Commands:

- a) Create
- b) Drop
- c) Alter

- d) Truncate
- e) Comment
- f) Rename

2) DQL – Data Query Language

Examples of DQL Commands:

- a) Select

3) DML – Data Manipulation Language

Examples of DML Commands:

- a) Insert
- b) Update
- c) Delete

4) DCL – Data Control Language

Examples of DCL Commands:

- a) Grant
- b) Revoke

5) TCL – Transaction Control Language

Examples of TCL Commands:

- a) Commit
- b) Rollback
- c) Savepoint
- d) Settransaction

2.3 Joins

Joins are the clause which are used to connect or combine two or more tables and combine rows from 2 or more tables. It is done on the basis of column related to them.

Different type of SQL Joins:

- a) Inner Join:**

INNER JOIN

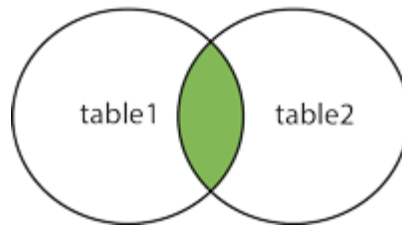


Fig. 22

b) Left Join

LEFT JOIN

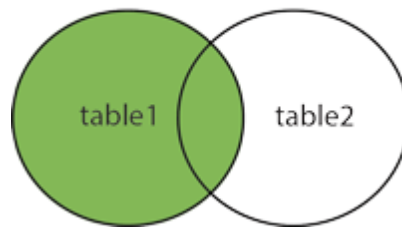


Fig. 23

c) Right Join

RIGHT JOIN

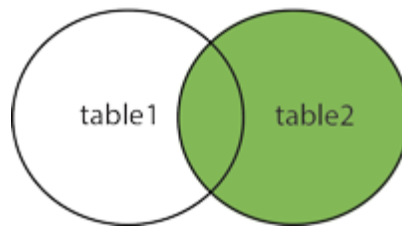


Fig. 24

d) Full Join

FULL OUTER JOIN

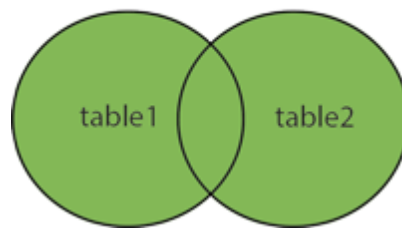


Fig. 25

e) Self Join

SELF JOIN

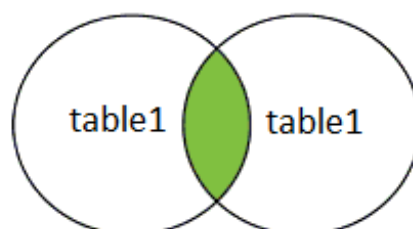


Fig. 26

2.4 Aggregate functions

These functions are used to perform sum ,min ,max on a given column etc

Fig. 27

SQL Functions

- **Aggregate Functions**
 - SQL aggregate functions **return a single value**, calculated from **all the values in a column**.

```
Tbl_stock Select SUM(int_price) from tbl_stock;
```

| Pk_int_id | Vchr_product | int_price |
|-----------|--------------|-----------|
| 1 | Pen | 10 |
| 2 | Book | 15 |
| 3 | Eraser | 2 |
| 4 | Pencil | 6 |

→ Returns a single value

- Select **AVG(int_price)** from tbl_stock; // returns 8
- Select **SUM(int_price)** from tbl_stock; // returns 33
- Select **MIN(int_price)** from tbl_stock; // returns 2

2.5 XML

XML stand for eXtensible Markup Language. It is software- and hardware-independent tool which is used for storing and transporting data. It is markup language which is similar to the HTML – Hypertext Markup Language. XML is case sensitive where as HTLM is not case sensitive.

Example:

It contains a note in which date, hour, sender, receiver and message send.

```
<note>  
  <date>2017-11-08</date>  
  <hour>08:30</hour>  
  <to>Raj</to>  
  <from>Ravi</from>  
  <body>Meeting at 8am.</body>  
</note>
```

Fig. 28

2.6 JSON

JSON stands for Java Script Object Notation. It is used to store and transport the data. When data is sent from server to web page that time JSON is used.

Example:

It contains a name, age, school name, athlete or not, and in classes it contains math, English, French, physics.

```
1 {  
2     "name": "Michael",  
3     "age": 16,  
4     "school": "JSON High School",  
5     "athlete": true,  
6     "Classes": [  
7         "math",  
8         "english",  
9         "french",  
10        "physics"  
11    ],  
12    "GPA": null  
13 }
```

Fig. 29

CHAPTER-3

3.1 Module 3: VB Script

In this module we have studied about the basics of VB Script, Objects, User Input, Output, Loops, If-Else statements, File Handling, Data handling in excel sheets using VB Script and FSO.

VB Script is the fundamental requirement for the UFT. In UFT we write scripts in VB Script. That's why VB Script is the most important module of our training.

We can use notepad, notepad++, Visual Code studio for run the VB Script.

VB Script is is very light and easy to learn and understand scripting language and It is the language which is developed by the Microsoft. VB Script is a great option for the new programmers and also for those who want to do automation testing.

VB Script has lots of option and FSO is one of them. In FSO we deal with the file, folders and subfolders of the system drives just using scripting.

3.2 Prerequisites

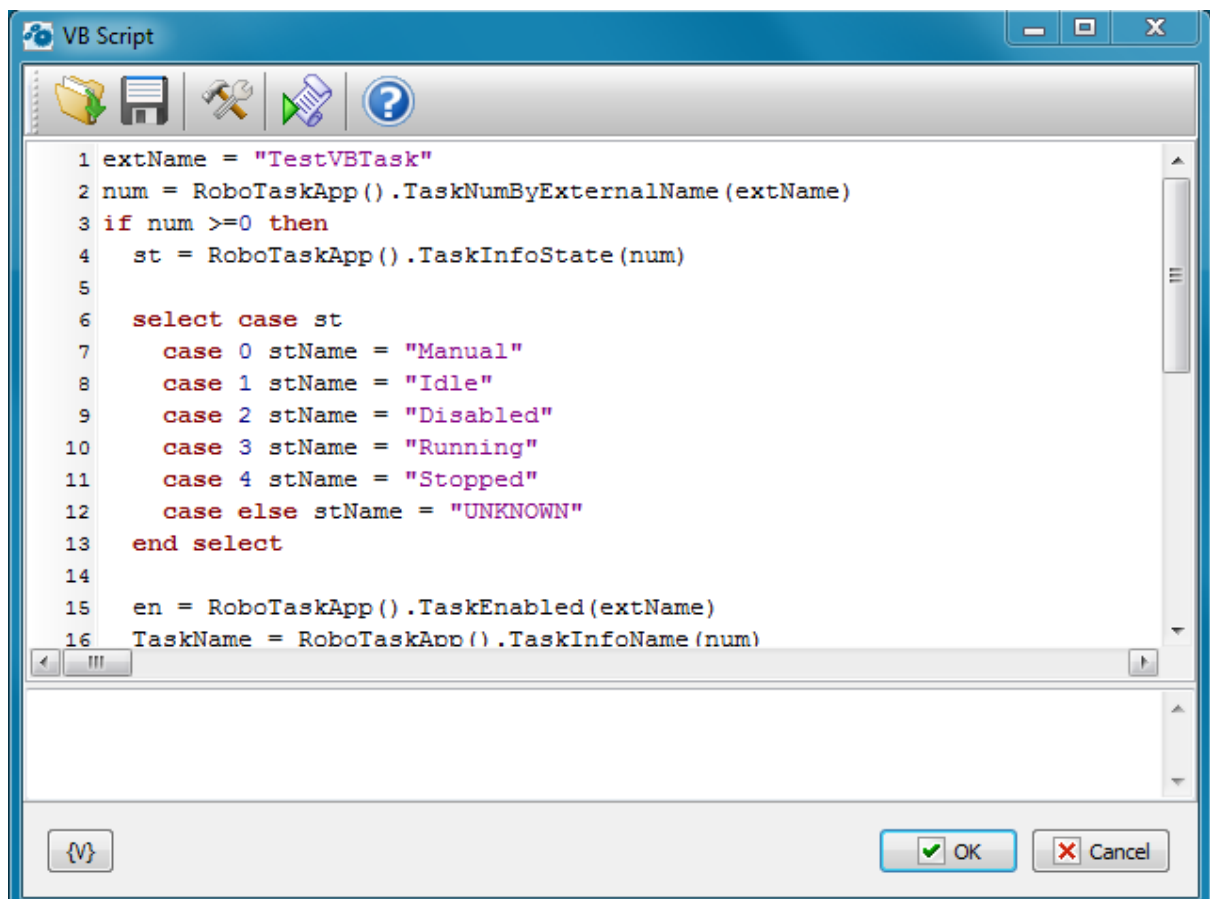
If you have basic knowledge of programming languages then you can learn VB Script very easily. You can learn VB Script from Udemy, Youtube, Guru99 website or you can learn from some other websites which you like most.

3.3 Syntax

In the below fig., We have created a VB Script. In this script we have defined the variable “extName” and then assigned some value “TestVBTask” to it.

Then we have used a variable “num” to define the function. After that we have used if else statement to verify the value to num greater than and equal to 0 or not.

Then we have used the select statement. Select statement in VB Script is similar to the Switch statement in C++ language. In select statement we have defined 6 cases. At last we have ended the select statement using “End” command.



```
1 extName = "TestVBTask"
2 num = RoboTaskApp().TaskNumByExternalName(extName)
3 if num >=0 then
4     st = RoboTaskApp().TaskInfoState(num)
5
6     select case st
7         case 0 stName = "Manual"
8         case 1 stName = "Idle"
9         case 2 stName = "Disabled"
10        case 3 stName = "Running"
11        case 4 stName = "Stopped"
12        case else stName = "UNKNOWN"
13    end select
14
15    en = RoboTaskApp().TaskEnabled(extName)
16    TaskName = RoboTaskApp().TaskInfoName(num)
```

Fig. 30

3.4 White Spaces and Line Breaks

In VB Script we can use spaces, newlines, tabs, freely. There is no issue of spaces, tabs and newlines in VBScript. Because VB Script ignores extra spaces, tabs and empty new lines.

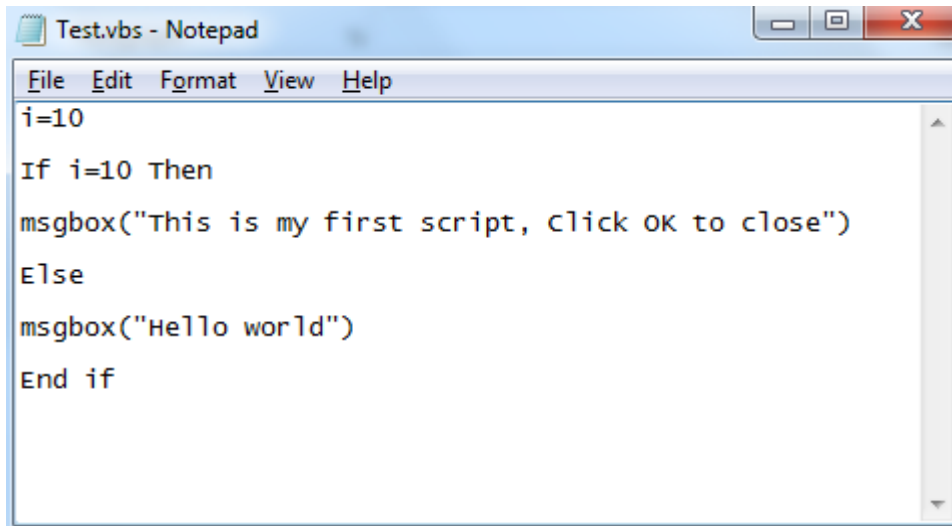


Fig. 31

CHAPTER-4

PERFORMANCE ANALYSIS

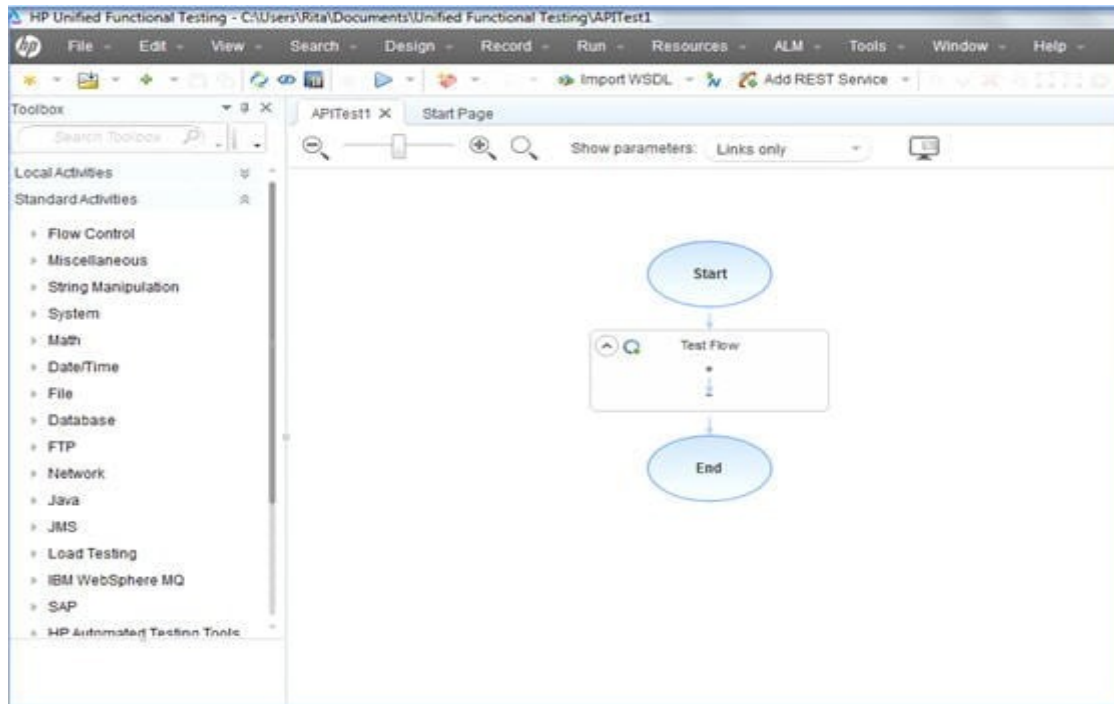
4.1 Module 4: UFT – Unified Functional Testing

In this module we have studied about the Micro Focus UFT. It is the one the most important module of our training. In this module we learn about the actual meaning of Automation testing. We learn a lot in the module, and we solve lot of hands on and we have created 3 projects also in this module.

4.2 About UFT

UFT stands for Unified Functional Testing. This software is now owned by Micro focus. UFT is a automation Testing tool. In this automation testing tool we can test on mobile, web, window applications. We can also use different browsers like Google Chrome, Microsoft Edge, Internet Explorer etc for testing in UFT. It supports lot of applications such as Windowws,

Web, Mobile, .Net, Java, WPF and Visual Basics and to use these applications we just have to check these add-ins when we launch UFT. It supports VB Script as scripting language. It also support databases, XML, JSON etc.



Fi
g.
32

4.3 History of UFT

Some year backs the Name of UFT was QTP which stands for Quick Test Professional. It is first owned by the Mercury and then UFT was bought by HP. They named it as HP UFT from QTP. Then HP handed over the UFT to the Micro Focus and they named it as Micro Focus UFT. Current version of UFT is 15.

4.4 Object Repository

UFT has a object repository in which we can store the objects. We can record objects in object repository using record button are using object spy tool.

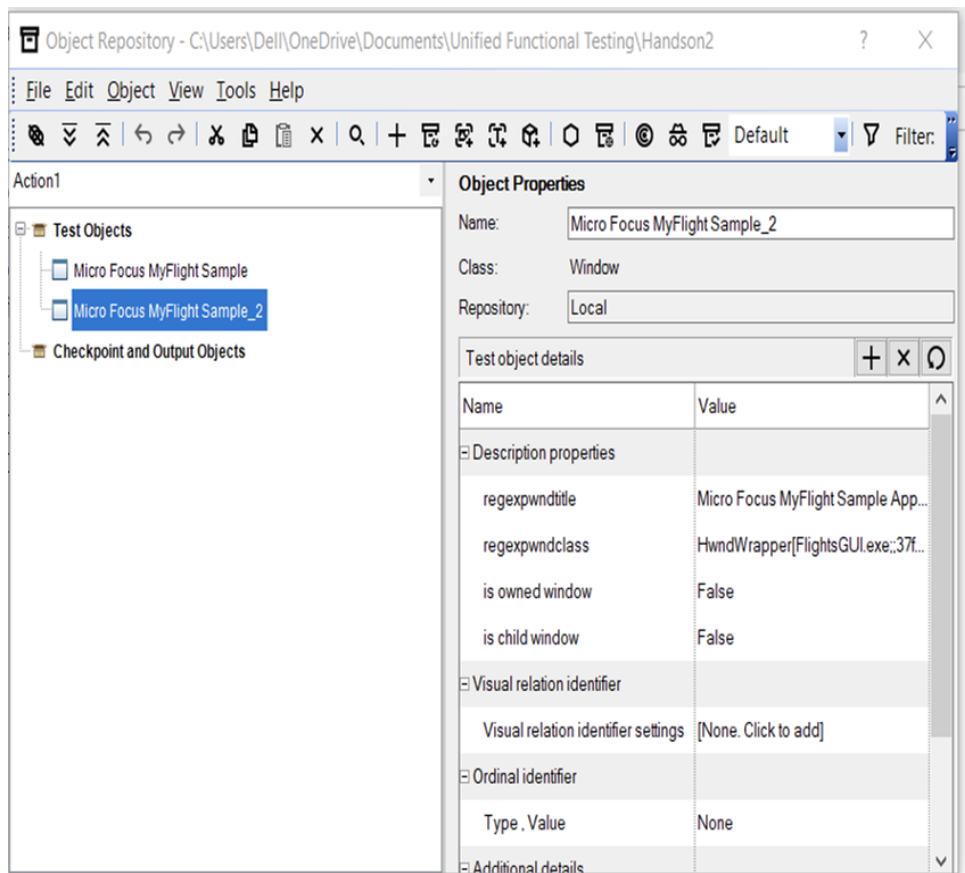


Fig. 33

4.5 Editor View

UFT has Editor view. It is the window in which we write the script. Everything done in UFT is scripted here.

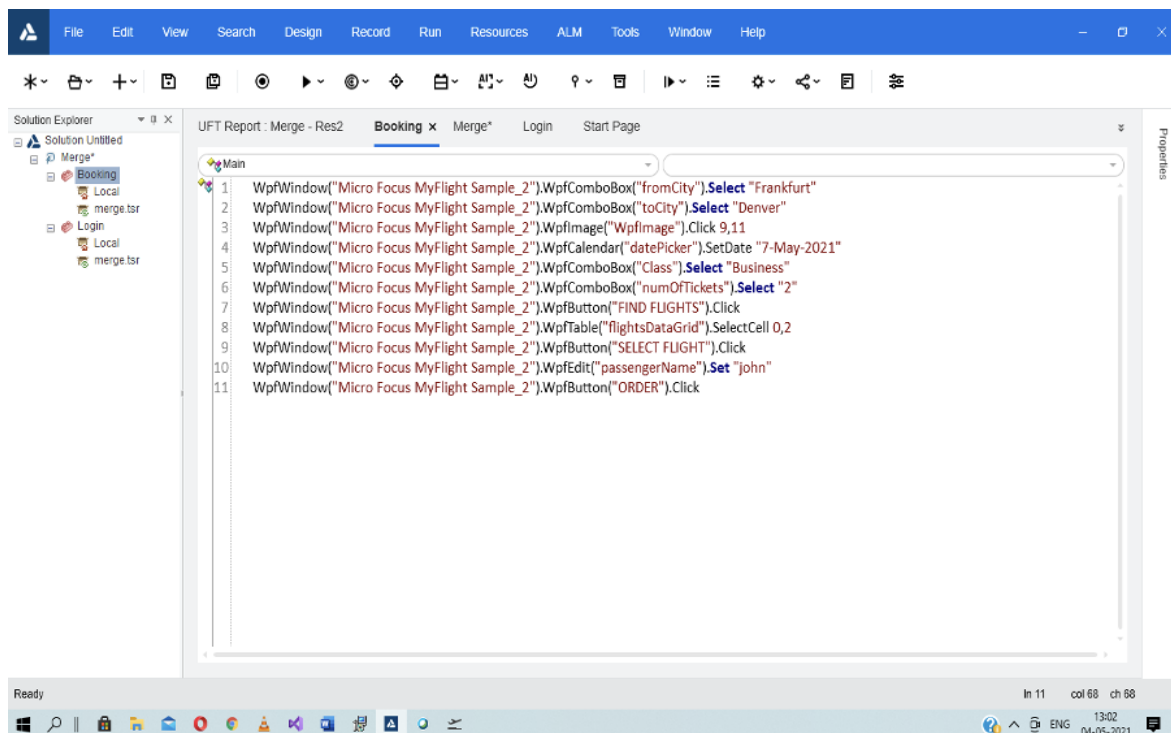
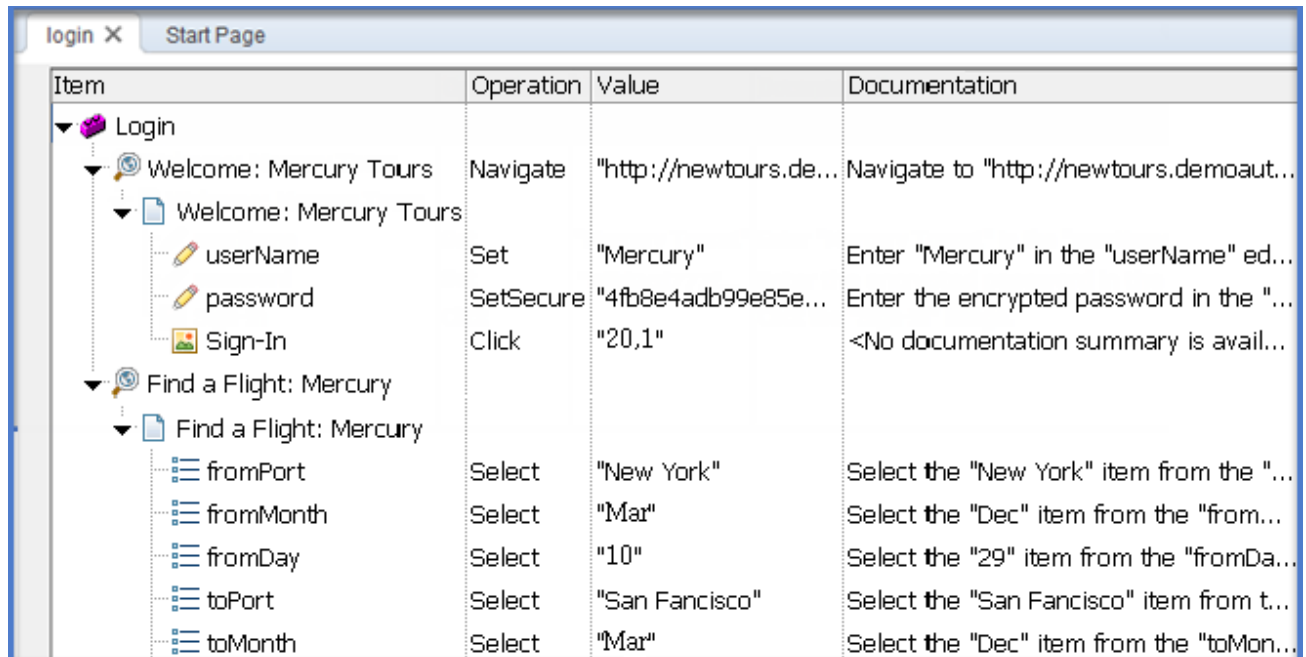


Fig. 34

4.6 Keyword View

UFT offers keyword view along with editor view. Tester can work on any of the view. It is in form of tables containing rows and columns we just have to add variables and their values in it. In keyword view user can parametrize the values, can add checkpoints and can also insert breakpoints to help the tester in debugging.



The screenshot shows the UFT Keyword View interface. It features a tree view on the left and a table on the right. The tree view shows a test suite named 'login' with two main test cases: 'Welcome: Mercury Tours' and 'Find a Flight: Mercury'. The 'Welcome: Mercury Tours' test case is expanded, showing a 'Sign-In' step. The 'Find a Flight: Mercury' test case is also expanded, showing several 'Select' steps for flight details. The table on the right provides a detailed view of the test steps, including the operation performed, the value being set or selected, and any associated documentation.

| Item | Operation | Value | Documentation |
|--------------------------|-----------|-------------------------|---|
| ▼ Login | | | |
| ▼ Welcome: Mercury Tours | Navigate | "http://newtours.de..." | Navigate to "http://newtours.demoaut..." |
| ▼ Welcome: Mercury Tours | | | |
| userNAme | Set | "Mercury" | Enter "Mercury" in the "userNAme" ed... |
| password | SetSecure | "4fb8e4adb99e85e..." | Enter the encrypted password in the "... |
| Sign-In | Click | "20,1" | <No documentation summary is avail... |
| ▼ Find a Flight: Mercury | | | |
| ▼ Find a Flight: Mercury | | | |
| fromPort | Select | "New York" | Select the "New York" item from the "... |
| fromMonth | Select | "Mar" | Select the "Dec" item from the "from..." |
| fromDay | Select | "10" | Select the "29" item from the "fromDa..." |
| toPort | Select | "San Fancisco" | Select the "San Fancisco" item from t... |
| toMonth | Select | "Mar" | Select the "Dec" item from the "toMon..." |

Fig. 35

4.7 Limitations or Drawbacks

Micro focus UFT has some limitations and drawbacks. Micro focus UFT is primarily runs on windows operating system. As VB Script is a Windows scripting language.

Second drawback is that we have to purchase a Licence for UFT.

CHAPTER - 5

CONCLUSIONS

5.1. Conclusions

In this internship, we have learned a lot of things like Testing and its types, Need of testing, Manual Testing and its type, SQL and different concepts of databases, XML and JSON, VBScript, Automation Testing and its type, UFT and FACETs. The concepts to this training really made my base strong in Testing, Manual Testing, SQL and different concepts of databases, XML, JSON, VBScript, Automation Testing, UFT and FACETs. and the learning rate will only increase after this.

5.2. Future Scope

In future we can learn and work on more efficient frameworks and tools of Automation Testing. We can also apply for the ISTQB certification which is world wide recognizable certification in testing We can use automation testing techniques to test in the following areas in future:

- Test Window and its new Applications.
- Test different webpages on different browsers like:
 - Google Chrome
 - Firefox
 - Internet Explorer
 - Safari
 - Microsoft Edge
- Test New Mobile Applications.
- Test new IOS and MAC applications.
- Test new Ubuntu and Linux Applications.

REFERENCES

- [1] <https://www.microfocus.com/en-us/home>
- [2] https://www.w3schools.com/sql/sql_intro.asp
- [3] <https://www.tutorialspoint.com/vbscript/index.html>
- [4] <https://www.softwaretestinghelp.com/automation-testing-tutorial-1/>
- [5] <https://cognizant.udemy.com/course/hp-uft-120-automation/learn>
- [6] <https://www.learnqtp.com/what-is-qtp/>
- [7] Everything you need to learn about software testing(Udemy) by Tarek Roshdy.
- [8] Understand SQL using the MySQL database. Learn Database Design and Data Analysis with Normalization and Relationships(Udemy) by Jon Avis.

- [9] API Documentation 1: Teaches technical writers how to document structured data. No programming experience required(Udemy) by Peter Gruenbaum.
- [10] Simplest Form of Automation(Udemy) by Coding Gears.
- [11] Learn the fundamentals of HP-UFT Functional Automation(Udemy) by Coding Gears.

APPENDIX

- Test Scenarios:

| Scenario ID | Scenario Name | Scenario Description | Requirement id |
|-----------------|--|---|----------------|
| TestScenario_01 | Raj Travels & Hotel Booking - Valid Search | To check if the user is able to Search hotel by entering valid details | R1 |
| TestScenario_02 | Raj Travels & Hotel Booking - Invalid Search | To check if the user is able to Search hotel by entering invalid details | R1 |
| TestScenario_03 | Raj Travels & Hotel Booking - Alternate Search | To check if the user is able to Search hotel by providing partial details | R1 |
| TestScenario_04 | Raj Travels & Hotel Booking - Field Validation | Validating fields given by the user interface | R2 |

Fig. 36

- Test Cases:


```

Employee.xml
1 <Department>
2   <Employee>
3     <empid>1001</empid>
4     <name>Tom</name>
5     <salary>20000</salary>
6     <email>tom@gmail.com</email>
7     <phoneno>9874563210</phoneno>
8   </Employee>
9   <Employee>
10    <empid>1002</empid>
11    <name>Sam</name>
12    <salary>25000</salary>
13    <email>sam@gmail.com</email>
14    <phoneno>7876545676</phoneno>
15  </Employee>
16  <Employee>
17    <empid>1003</empid>
18    <name>Shiny</name>
19    <salary>20000</salary>
20    <email>shiny@gmail.com</email>
21    <phoneno>9876543210</phoneno>
22  </Employee>
23 </Department>

```

Fig.39

- Application of VB Script:

```

2 Dim n
3 Dim f
4 f=1
5 n= InputBox("Enter the number to find factorial:")
6 For i= 1 to n Step 1
7   f=f*i
8 Next
9 MsgBox("Factorial of the given number is :") &f
10
11

```

Fig. 40

```

2 Dim a
3 Dim b
4 Dim c
5 a= InputBox("Enter the first Number:")
6 b= InputBox("Enter the Second Number:")
7 c= InputBox("Enter the operator character:")
8 Select case c
9   case "A"
10    MsgBox("Addition of both numbers:") & (CInt(a) + CInt(b))
11   case "S"
12    MsgBox("Subtraction of both numbers:") & (CInt(a) - CInt(b))
13   case "M"
14    MsgBox("Multiplication of both numbers:") & (CInt(a) * CInt(b))
15   case "D"
16     if (b=0) then
17       MsgBox("Division is not possible by 0")
18     else
19       MsgBox("Division of both numbers:") & (CInt(a) / CInt(b))
20     End if
21   case else
22     MsgBox("Unknown Operator")
23 End Select
24

```

Fig. 41

- Application of UFT Automation:

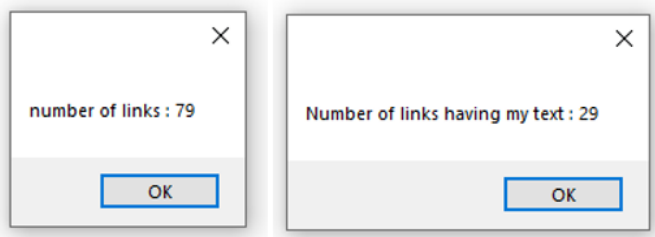
```

Main
1  Set desc=description.Create()
2
3  desc("micclass").value="Link"
4
5  Set objLink =Browser("creationtime:=0").Page("title:=.*").childobjects(desc)
6
7  c=0
8  msgbox "number of links : " & objLink.count
9  For i = 1 To objLink.count - 1 Step 1
10 if instr(objLink(i).getROProperty("name"),"my") then
11 c=c+1
12 end if
13 Next
14 msgbox "Number of links having my text : " & c

```

Fig. 42

OUTPUT:



TEST REPORT:

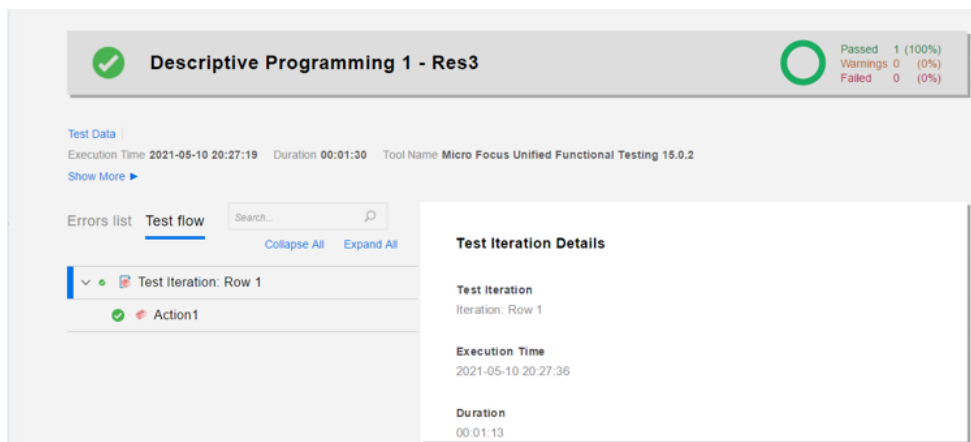


Fig. 43

final project report

by Rahul Soni

Submission date: 24-May-2021 01:52PM (UTC+0530)

Submission ID: 1592979377

File name: sem_8_Final_Report.docx (4.1M)

Word count: 5038

Character count: 25356

final project report

ORIGINALITY REPORT

19%

SIMILARITY INDEX

14%

INTERNET SOURCES

0%

PUBLICATIONS

13%

STUDENT PAPERS

PRIMARY SOURCES

| | | |
|---|---|-----|
| 1 | Submitted to Jaypee University of Information Technology Student Paper | 5% |
| 2 | Submitted to Sogang University Student Paper | 2% |
| 3 | investors.forescout.com Internet Source | 1% |
| 4 | gmsenthi.wordpress.com Internet Source | 1% |
| 5 | www.sec.gov Internet Source | 1% |
| 6 | Submitted to Colorado Technical University Online Student Paper | 1% |
| 7 | www.partnershipsbc.ca Internet Source | 1% |
| 8 | Submitted to UNITEC Institute of Technology Student Paper | <1% |
| 9 | www.guru99.com | |

JAYPEE UNIVERSITY OF INFORMATION TECHNOLOGY, WAKNAGHAT

PLAGIARISM VERIFICATION REPORT

Date: 17th June 2021



Type of Document (Tick): PhD Thesis M.Tech Dissertation/ Report B.Tech Project Report Paper

Name: RAHUL SONI Department: CSE Enrolment No: 171321

Contact No: 8219654799 E-mail: rahulsoni442000@gmail.com

Name of the Supervisor: MR. SURJEET SINGH

Title of the Thesis/Dissertation/Project Report/Paper (In Capital letters): AUTOMATION TESTING WITH UFT LEARNING PATH

UNDERTAKING

I undertake that I am aware of the plagiarism related norms/ regulations, if I found guilty of any plagiarism and copyright violations in the above thesis/report even after award of degree, the University reserves the rights to withdraw/ revoke my degree/report. Kindly allow me to avail Plagiarism verification report for the document mentioned above.

Complete Thesis/Report Pages Detail:

- Total No. of Pages = 51
- Total No. of Preliminary pages = 14
- Total No. of pages accommodate bibliography/references = 1

(Signature of Student)

FOR DEPARTMENT USE

We have checked the thesis/report as per norms and found **Similarity Index** at 19 (%). Therefore, we are forwarding the complete thesis/report for final plagiarism check. The plagiarism verification report may be handed over to the candidate.

(Signature of Guide/Supervisor)

Signature of HOD

FOR LRC USE

The above document was scanned for plagiarism check. The outcome of the same is reported below:

| Copy Received on | Excluded | Similarity Index (%) | Generated Plagiarism Report Details (Title, Abstract & Chapters) | |
|----------------------------|--|----------------------|--|-------|
| 24 rd May 2021 | <ul style="list-style-type: none">• All Preliminary Pages• Bibliography/Images/Quotes• 14 Words String | 19 | Word Counts | 5038 |
| Report Generated on | | | Character Counts | 25356 |
| 24 rd May 2021 | | Submission ID | Total Pages Scanned | 51 |
| | | 1592979377 | File Size | 4.1 M |

Checked by
Name & Signature

Librarian

Please send your complete thesis/report in (PDF) with Title Page, Abstract and Chapters in (Word File) through the supervisor at plagcheck.juit@gmail.com