

AUTOMATION TESTING – HC FACETS WITH UFT

Project report submitted in partial fulfillment of the requirement for the degree of

BACHELOR OF TECHNOLOGY

IN

ELECTRONICS AND COMMUNICATION ENGINEERING

By

SHRUTI SHARMA (171051)

UNDER THE GUIDANCE OF

SOWNDHARIYA RAVICHANDRAN



JAYPEE UNIVERSITY OF INFORMATION TECHNOLOGY, WAKNAGHAT

May, 2021

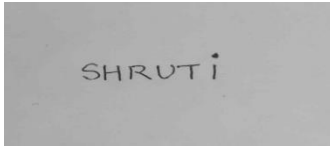
TABLE OF CONTENTS

CAPTION	PAGE NO.
DECLARATION	i
PROJECT REPORT UNDERTAKING	ii
ACKNOWLEDGEMENT	iii
LIST OF ACRONYMS AND ABBREVIATIONS	iv
LIST OF FIGURES	v
LIST OF TABLES	vii
ABSTRACT	viii
CHAPTER-1: SOFTWARE TESTING	1
1.1 Introduction	1
1.2 Need of Software Testing	2
1.3 Principles of Software Testing	3
1.4 Software Development Lifecycle	5
1.5 Objectives of Testing	5
1.6 Testing Process	6
1.7 Testing Types	9
1.8 Testing Levels	11
1.9 Agile	12
1.9.1 Agile Principles	12
1.9.2 Agile Software Development	13
1.9.3 Agile Methodologies	14
CHAPTER-2: DATASOURCE	15
2.1 Database	15
2.2 Database Management System	15
2.3 Types of Database	16
2.4 Database Components	17
2.5 SQL	18
2.5.1 Use of SQL	19
2.5.2 How SQL Works	20
2.6 XML	21
2.6.1 XML Usage	21
2.6.2 Difference between XML and HTML	22
2.7 JSON	22

2.7.1 Uses of JSON	23
2.7.2 Characteristics of JSON	23
CHAPTER 3 - VBscript	24
3.1 Introduction	24
3.2 Features of VBscript	24
3.3 History and Uses	24
3.4 Work Done	25
CHAPTER 4 - UFT	34
4.1 Introduction	34
4.2 Why UFT?	34
4.3 Advantages	35
4.4 Work Done	35
CONCLUSION	40
REFERENCES	41
APPENDIX	
PUBLICATIONS	
PLAGIARISM REPORT	

DECLARATION

I hereby declare that the work reported in the B. Tech Project Report entitled “**Automation Testing- HC facets with UFT**” submitted at **Jaypee University of Information Technology, Wagnaghat, India** is an authentic record of my work carried out under the supervision of **Sowndhariya Ravichandran (Cognizant)**. I have not submitted this work elsewhere for any other degree or diploma.

A rectangular box containing the handwritten name 'SHRUTI' in capital letters.

SHRUTI SHARMA
171051

This is to certify that the above statement made by the candidates is correct to the best of my knowledge.

A handwritten signature in cursive script, appearing to read 'Arunya'.

Sowndhariya Ravichandran
Date: 20th May, 2021

Head of the Department/Project Coordinator

Project Report Undertaking

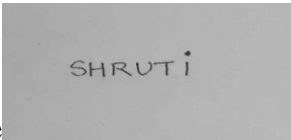
I Ms. SHRUTI SHARMA, Roll No. 171051, Branch Electronics and Communications is doing my internship with Cognizant from 27th February, 2021 to 11th June 2021.

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



SHRUTI

Name SHRUTI SHARMA

Roll No. 171051

ACKNOWLEDGEMENT

Knowledge, energy and time are the resources in the completion of this project but the most requisite is the proper guidance. I would like to express my deepest appreciation to all those who have been helping me throughout the project and without whom this project would have been a very difficult task. I am highly indebted to my respected project guide, Sowndhariya Ravichandran for his valuable suggestions, ideas and constant supervision as well as for providing necessary information during the course of the project. Also, I wish to express my heartfelt gratitude to my college mentor Mr. Anuj Kumar Maurya for guiding me whenever I needed help.

I would like to thank Dr. Rajeev Kumar, Head of the Department, Department of ECE, Jaypee University of Information Technology, Wagnaghat for giving us this opportunity to carry out our project.

No thanks can counter my indebtedness to my parents who have been with me throughout. I thank them from the core of my hearts.

LIST OF ACRONYMS AND ABBREVIATIONS

SDLC	Software Development Lifecycle
UAT	User Acceptance Testing
DBMS	Database Management System
SQL	Structured Query language
GUI	Graphical User Interface
XML	Extension Markup Language
VBA	Visual Basic for Applications
IIS	Internet Information Services
WSH	Windows Scripting Host
UFT	Unified Functional Testing

LIST OF FIGURES

Figure No.	Caption	Page Number
Figure 1.1	Software Testing	1
Figure 1.2	Software Development Lifecycle	4
Figure 1.3	Types of Software Testing	9
Figure 1.4	Agile Principles	12
Figure 2.1	Database	15
Figure 2.2	Types of Database	16
Figure 2.3	Database Components	17
Figure 2.4	SQL Process	19
Figure 2.5	SQL Working	20
Figure 3.1	Problem Statement 1	25
Figure 3.2	Code 1	25
Figure 3.3	Output 1.1	25
Figure 3.4	Output 1.2	26
Figure 3.5	Problem Statement 2	26
Figure 3.6	Code 2	26
Figure 3.7	Output 2.1	27
Figure 3.8	Output 2.2	27
Figure 3.9	Problem Statement 3	27
Figure 3.10	Code 3	28
Figure 3.11	Output 3.1	28
Figure 3.12	Output 3.2	28
Figure 3.13	Output 3.3	28
Figure 3.14	Output 3.4	29
Figure 3.15	Output 3.5	29
Figure 3.16	Output 3.6	29
Figure 3.17	Output 3.7	29
Figure 3.18	Problem Statement 4	30
Figure 3.19	Code 4	30
Figure 3.20	Output 4.1	30
Figure 3.21	Output 4.2	31
Figure 3.22	Problem Statement 5	31
Figure 3.23	Code 5	31
Figure 3.24	Output 5	32
Figure 3.25	Problem Statement 6	32
Figure 3.26	Code 6	32
Figure 3.27	Output	33
Figure 4.1	UFT	34
Figure 4.2	Problem Statement 1	36
Figure 4.3	Select Web	36

Figure No.	Caption	Page Number
Figure 4.4	Record	36
Figure 4.5	Web Objects	37
Figure 4.6	Deselect Web	37
Figure 4.7	Record and Run	38
Figure 4.8	Disable Mobile Settings	38
Figure 4.9	Object Identification	39

LIST OF TABLES

Table No.	Caption	Page Number
Table 1.1	Difference between HTML and XML	22

ABSTRACT

Automation frameworks are regarded crucial in the software testing area, especially when you are interested in automation testing. We'll look at the many sorts of automation frameworks in this report. In technical terms, an automation framework is a collection of principles that includes coding standards, app code and data folder organization, test results storage, external resource access, and shared object repositories, among other things.

Standard Automation Framework Requirements:

- Code should be re-usable, understandable, and easy to maintain.
- The automation framework must include data parameterization.
- Test data should be kept separate from the code, and the code should be reusable for various input data sets.
- Test results should be recorded in a separate log file and be simple to comprehend.
- The automation framework must have an appropriate error handling method.
- Additionally, the use of recovery scenarios should be reduced to a bare minimum, as they take additional processing time and resources.
- The automation framework should be developed in such a way that if one test script fails, Unified Functional Testing will continue to run.

1.2 NEED OF SOFTWARE TESTING

Software's efficiency, accuracy and quality determines its success in the market and business. Hence it is crucial to always ensure the performance of the software.

We perform software testing for our own benefit. Some of them are:

1. **Cost-Effective:** One of the most important advantages of testing is its cost-effectiveness. In the long run, testing our project on time can save money. There are several stages to software development, and catching flaws early so as to save a lot of money in the long run.
2. **Security:** While testing of software, most sensitive and vulnerable aspect is security. Users are constantly on the lookout for reliable items. It aids in the prevention of problems and risks.
3. **Reliability:** Software errors can be costly or even deadly, therefore testing is critical. Software flaws have the ability to cause monetary and human losses, and there are numerous examples throughout history.
4. **Productivity:** The software needs to be performed in an efficient and productive manner. Therefore, to bring product vision into life, it should be planned. We need to keep in mind the product requirements to acquire the desired results.
5. **Customer Satisfaction:** A product owner's ultimate goal is to provide the highest level of customer satisfaction. Software should be thoroughly tested in order to provide the greatest possible user experience. Being the best product in a crowded market will help you gain loyal customers, which will have long-term benefits. [1]

1.3 PRINCIPLES OF SOFTWARE TESTING

Software testing requires a lot of creativity and intelligence on the part of testers. There are some basic rules that are vital when testing software or applications.

The Principles of Software Testing are as follows:

1. Program testing may aid in the detection of bugs: Testing of project or a software may aid in the discovery of many or a few problems that developers might never be aware of. However, software testing does not guarantee that our software or project has no error. As a result, it is critical to create test scenarios and uncover all flaws.
2. Effective testing is impossible: It will be impossible to verify and test all potential sets of data, modules, and situations until your project or application under test has a simple structure with minimal input.
3. Testing Early: The sooner we start testing our project and product, the better we'll be able to make the most of our time.
4. Clustering defect: During testing, you may see that the bulk of the reported defects or faults are caused by a limited number of phases within the planning system.
5. Software testing is context-dependent: Different methods, techniques, and types of testing exist, each of which determines the application's kind and characteristics. A game or small piece of software, for example, requires more testing and doctor-based feedback than a health-related app.
6. Software that is free of error or bug is no less than a myth: Simply because a tester found no problems in an application does not mean or concludes that software is complete.[1]

As a result, when testing modules or the software's operation, a tester must determine whether the software meets all of the user or client criteria and whether any flaws discovered during testing have been fixed. Before shipping or releasing software to the market, these numerous aspects must be taken into account.

1.4 SOFTWARE DEVELOPMENT LIFECYCLE (SDLC)

The Software Development Life Cycle (SDLC) is known as a series of procedures carried out by the team of testers to guarantee that the product or software is of high quality. It refers to a set of procedures for performing testing of the software. It also specifies which activities of testing should be performed and when they should be completed. Each action in the SDLC process is done in a schematic and efficient manner, and every activity has its own set of goals and deliverables.



Figure 1.2Software development cycle

The different activities of life cycle of testing of a software are:

1. Analysis of requirements — The basic and foremost step in the Life Cycle of software testing is Requirement Analysis. The testers team and team that ensure quality decipher and determine the requirements that needed to be tested in this step.
2. Test Planning – The most significant step of the Life Cycle of software testing is Test Planning, when all testing strategies are created. This phase is also known as the stage of test strategy. The Manager of testing is active in this phase to determine the project's effort and cost estimations. It establishes the project's goal and scope.

3. Development of test case - After the phase of planning step is performed, the development of test case and scenarios step starts. This is the stage of the SDLC when the testing team takes thorough notes on the test cases. In addition to test cases, the team of testers prepare data for testing. Once the test scenarios are complete, they are consulted by fellow men.
4. Setup of test environment – A crucial aspect of the Life Cycle of software testing is setting up the test environment. A testing environment includes both hardware and software configuration that allows teams of testers to run cases of test. It enables execution of test with properly configured software, hardware and network.
5. Execution of test – Test Execution is the followed step in the Life Cycle of software testing. The procedure of implementing the code and comparison of actual and expected results is known as test execution. When test execution phase starts, the analysts proceed implementing the test scripts in accordance with the project's strategy of testing.
6. Test Cycle Closure – Test Cycle Closure is the last component of the Software Testing Life Cycle. It entails identifying and analyzing completion of cycle criteria based on time, cost, quality, test coverage, and objectives of business, and efficiency of software for each member of the testing team.[1]

There are two types of SDLC:

1. Sequential Model: It describes development of software process as a sequential and linear flow of activities. Any phase in the development process should begin when the previous phase is completed.
2. Iterative and Incremental Model: Iterative and incremental development is a combination of both iterative design or iterative method and incremental build model for development.

1.5 OBJECTIVES OF TESTING

1. Work product evaluation: This includes everything created while working on the project, including requirements, code, test cases, test plans, and designs.

2. Requirement fulfilment: Software must complete the requirements of the customers.
3. Building confidence: It is made sure that the components and systems are good enough in order to build confidence among the customers.
4. Preventing defects: By early testing, the defects can be found and corrected.
5. Providing information to the stakeholders: To provide the team leader, manager, client about the level of quality used for the software.
6. Compliance with the laws: The contract is honored and customer satisfaction is kept number one priority.

1.6 TESTING PROCESS

Testing process refers to the steps we go through in order to test our software. It consists of 3 basic parts:

1. TEST STRATEGY AND TEST PLAN: A test strategy and plan are required for every project.

These artefacts define the project's testing scop

- The systems that need to be examined, as well as special setups
- The focus of the project are features and functions
- A demand that is Non-functional
- Methodology for testing
- Processess to follow
- Instruments to detect flaws and errors
- Setup and pre requisites for test environment
- Threats, repercussions and contingencies
- Test Timetable [2]

2. TEST ANALYSIS AND DESIGN: After test planning, we analyze and design the test cases and test scenarios

Design of test has the following main tasks:

- • To go over the test basis again. (The test basis is the data required to begin the analysis and develop our own cases of test.) It's essentially a set of documents that serve as the foundation for test cases, such as design specifications, quality, efficiency and risk analysis, including user interface. The baseline of test documents can help us figure out what the system should do after it's finished.)
 - To figure out the conditions of test.
 - To create the tests.
 - To assess the needs and system's testability.
 - To plan the setup of test environment and determine the infrastructure and methods that are required.
3. IMPLEMENTATION AND EXECUTION OF TEST: During execution and implementation, we acquire the conditions of test and procedures and, the environment of test and other test requirements.

Test implementation has the following major task:

- i. To use approaches to construct and prioritize our test cases, as well as to generate data to conduct those tests. In order to test a software, we must first enter some information to test the majority of its capabilities. Test data is any such properly specified data that is utilized in tests. We also write test protocols, which are instructions for carrying out the tests.

A test suite is a collection of test cases intended to demonstrate that a software programme has a certain set of procedures. A test suite often includes specific information and instructions for every set of cases that will be utilized during testing. the suites of test are used to bundle together comparable cases of test.

- ii. To set up and test the environment.

Execution of test has the main task such as:

- i. Follow test procedures to execute cases of test.

In order to confirm a fix, re-run the tests that previously failed. Confirmation testing or re-testing is the term for this.

- ii. To record the versions and identities of the programme under test, as well as the results of the test execution.
- iii. To Compare actual results with projected results.
- iv. It reports disparities as Incidents when there are disparities between actual and expected results.

4. EVALUATING EXIT CRITERIA AND REPORTING:

We need to create the criteria for every level of test against which we will conduct "enough testing" based on the project's risk assessment. Exit criteria are a type of criterion that varies from project to project.

When exit criteria are used, they are referred to as "exit criteria.":

- The maximum number of test cases are run with a specific pass rate.
- The bug rate drops down below threshold level.
- When the deadlines are met.

The following are the primary tasks involved in evaluating exit criteria:

- i. To compare logs of test to the exit criteria set forth in the planning of tests.
- ii. Find whether additional required tests re needed or whether the given exit criteria must be altered.
- iii. To prepare a stakeholder summary report on the test results.

5. TEST CLOSURE ACTIVITIES: When software is delivered, test closure tasks are completed.

Testing might be halted for a variety of reasons, including:

- When you've gathered all of the information you'll need for the test.
- When a project is postponed or discontinued.
- When a specific goal is met.
- When there is a maintenance release or an update.

The following are the key tasks of test closure activities:

- i. Verify that all planned deliverables have been completed, as well as that all reports of incident are handled.
- ii. To complete and archive the testware, such as test environments and scripts, and so on, in order to reuse it later.
- iii. To submit the testware to the organization of maintenance.
- iv. To find how the errors.[3]

1.7 TESTING TYPES

Testing is an essential component of any successful software development project. Project needs, money, timeframe, experience, and suitability all influence the type of testing done.



Figure 1.3Types of testing

Software testing is a huge domain but can be broadly categorized into:

- 1) **FUNCTIONAL TESTING:** Functional testing focuses on an application's business needs. They just inspect the output of an activity and do not inspect the system's intermediate stages while doing it. Only the most important features of software are tested.
- 2) **NON- FUNCTIONAL TESTING:** We examine how the system functions (response time, accuracy etc.). As an example, to assess performance.

- 3) **BLACK BOX TESTING:** This method of testing ignores internal system architecture. The requirements and functionality are used to guide the testing process. We test without looking at the software's internal structure.
- 4) **WHITE BOX TESTING:** The technique of White Box Testing is based on an understanding of an application's core logic.

Glass box testing is another name for it. Internal software and code should be well-known for carrying out this type of testing. These tests rely on the coverage of code statements, branches, paths, and conditions, among other things.

- 5) **DYNAMIC TESTING:** In dynamic testing we execute the software.
- 6) **STATIC TESTING:** Static testing is a sort of testing that takes place without the use of any code. During the testing step, the execution is carried out on the documentation.

It entails reviewing, walking through, and inspecting the project's deliverables. Instead of executing the code, static testing checks the naming conventions.

Static testing can be used on test cases, test plans, and design documents. Static testing by the testing team is required since the problems discovered during this sort of testing are cost-effective from a project standpoint.

- 7) **CONFIRMATION TESTING:** Confirmation testing, often known as retesting, is testing done after debugging to ensure that the issues have been solved.
- 8) **REGRESSION TESTING:** Regression Testing is the process of testing a programme as a whole for changes to any module or functionality. Because it is impossible to test every aspect of a system in Automation Testing and regression testing Tools are commonly utilized for such types of tests.

- 9) **SMOKE TESTING:** Smoke tests are simple tests that verify the application's basic operation. They're designed to be quick to implement, and their objective is to give you peace of mind that your system's essential features are functioning properly.

Smoke tests are simple tests that verify the application's basic operation. They're designed to be quick to implement, and their objective is to give you peace of mind that your system's essential features are functioning properly.

1.8 TEST LEVELS

Test levels are referred to as activities that are managed together.

- 1) **UNIT OR COMPONENT TESTING:** Unit testing refers to the testing of a single software component or module. It is usually done by developers rather than team of testers because it requires an efficient knowledge of the internal programme design and code. It could also require the creation of test driver modules or test harnesses. We test everything that can be tested separately.
- 2) **INTEGRATION TESTING:** Integration testing refers to the process of testing all integrated modules to ensure that all the functions run correctly and efficiently as a whole. Then we put these code modules together and concentrate on how components and systems interact.

Code modules, standalone apps, client and server apps on a network, and so on are examples of modules.

- 3) **SYSTEM TESTING:** The complete system is tested as per to the requirements using the System Testing approach. It's one of the forms of Black-box testing that's based on overall requirement.
- 4) **ACCEPTANCE TESTING:** n Acceptance Test is conducted by the client to gather information if the systems application meets his requirements.

It focuses on the needs and requirements of user so as to provide product with functionalities and performance desired by the client.

It is the last stage of testing before the software is put into market for production. User Acceptance Testing is another name for this.

Types of acceptance testing:

- i. Alpha testing: Alpha testing takes place on the developer's property. For this form of testing, an in-house virtual user environment can be established.
- ii. Beta testing: Beta testing is a type of formal software testing performed by the customer. Before releasing the product to the market for actual end-users, it is tested in a real-world setting.[4]

1.9 AGILE

Agile is a term that consists a set of principles that are implemented during development of any software and management of project. Agile encourages the developers and testers to work together in a fair, efficient and systematic manner keeping in mind the requirements of clients. The design, plans and test cases, and overall results analyzed. This enhances the team's performance to respond as fast as possible to changes.

1.9.1 AGILE PRINCIPLES



Figure 1.4Agile Principles

1. **SATISFACTION OF CUSTOMERS:** The customer must be satisfied with the product's prompt delivery.
2. **WELCOME CHANGE:** Changing demands must be handled even late in the development phase.
3. **DELIVERY OF PRODUCT IN TIME:** Focus on a shorter timeframe and make sure things are provided on a regular basis.
4. **COLLABORATE.** Throughout the project, the business and development teams must collaborate
5. **MOTIVATED TEAM:** To execute the project successfully and on schedule, team members must be motivated and trustworthy.
6. **ONE TO ONE COMMUNICATION:** One of the most successful kinds of communication is face-to-face connection.
7. **EFFICIENT SOFTWARE:** Having working and correct output indicates that you've made progress toward your goal.
8. **CONSTANT PACE:** Agile encourages long-term growth.
9. **GOOD DESIGN:** Focus on good design and technical excellence to improve agility.
10. **SIMPLICITY:** The amount of time spent on non-productive activities must be reduced.
11. **SELF ORGANIZATION:** These teams produce the most innovative ideas, specifications, and structures.
12. **ADJUST AND REFLECT:** The team's performance can be increased by reflecting and improving on their work on a frequent basis.[5]

1.9.2 AGILE SOFTWARE DEVELOPMENT

One of the simplest and most successful ways to turn a vision for a company need into software solutions is to use the Agile software development methodology. Continuous learning, designing planning, improving and development as a whole, and early delivery are all terms used to describe development of agile methodologies.

The four essential values of agile software development are highlighted.

1. Interactions between customers and teams over software.

2. Efficient software trumps thorough documentation
3. Collaboration more important than negotiations
4. Adapting to change as per requirement [6]

1.9.3 AGILE METHODOLOGIES

1. **EXTREME PROGRAMMING:** This is a framework that allows teams to produce good performance and quality of software that brings help to everyone. It allows for software development in conjunction with good engineering principles.
2. **KANBAN:** This is a system flow design, management, and improvement strategy. Organizations can use Kanban to visualize their work flow and limit the quantity of work in process. It's employed in circumstances where work arises unexpectedly and needs to be corrected at present.
3. **LEAN:** This is a set of ideas and tools aimed at identifying and eliminating waste in order to accelerate the development of processes. The maximum amount of value is obtained.
4. **SCRUM:** This is a framework that teams use to create a hypothesis, test it, reflect on the results, and make changes. Depending on the requirements, it allows teams to apply practices from various frameworks. Cross-functional teams working on product development use it, and the work is divided into many 2–4-week iterations.
5. **CRYSTAL:** Rather than focusing on tools and procedures, it focuses on people and their connections. Crystal operates on the idea that each project is unique and dynamic in order to streamline operations and increase optimization. It's employed when the goal is to improve team communication and integration over time.[5]

CHAPTER 2

DATASOURCE

2.1 DATABASE

Database is a logically organized collection of informative and structured data kept electronically in a computer system. A database management system is usually in charge of a database. The data or information along with the DBMS, and the applications that go with them are referred to as a database system, which is commonly abbreviated to just database.



Figure 2.1 Database

A database is saved as a collection of files. The data in these files can be divided into records, each of which has one or more fields. Each field normally provides information relevant to one feature or feature of the entity defined by the database. Tables are also used to organize records, which contain information about the relationships between the various fields. Although the term "database" is loosely applied to any collection of data stored in computer files, a database in the technical meaning allows for cross-referencing. Users may quickly search, reorganize, categorize, and select items using keywords and other sorting methods.[7]

2.2 DATABASE MANAGEMENT SYSTEM

Database Management System (DBMS) is a collection of programs that enable its users to add, edit manipulate and access data.

Database management system (DBMS) is a varied and comprehensive database-based program that is required by a database. A DBMS acts as a bridge between the database and its end users or programs, and allows its users the retrieval, updating, and management about organization and optimization of information.

2.3 TYPES OF DATABASE

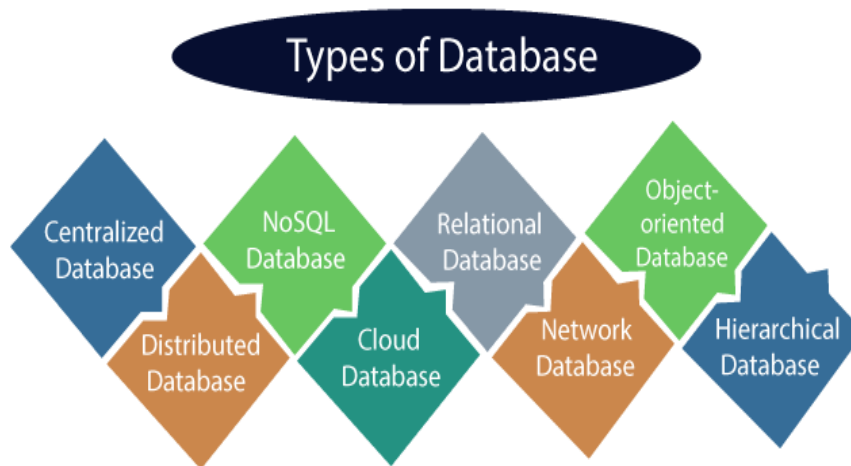


Figure 2.2Types of database

There are many types of database. Some of them are:

Relational databases:

Under this database tables are used for defining relationships in databases. Also called Relational DBMS, it is the most extensively sought-after database management system across the global marketplace. Some examples of RDBMS systems are Oracle, and Microsoft SQL Server databases.

Object-oriented databases:

In this type of computer database, data of all kinds can be stored. The information inside it is held in the form of objects. To define what to do with the data, characteristics as well as methods of the objects stored in database are responsible.

Centralized database:

It is a centralized site where data may be accessed by people from various backgrounds. Application processes that allow users to access data from a remote place are stored in this form of computer database.

Cloud databases:

Cloud databases are databases that have been customized or created for specific use in a virtual surrounding. Use of a cloud database comes with a variety of benefits, including covering the bandwidth and storage cost. On-demand scalability and excellent availability are some of its other features.

NoSQL databases:

For massive sets of distributed data, a NoSQL database is employed. There are a few issues with large data performance that can be solved by relational databases in an efficient manner. Evaluation of huge amounts of unstructured data can be done using this database.

Hierarchical:

This form of DBMS stores data in a "parent-child" connection. It has a tree-like structure, where nodes signify records and branches signify to fields. The Windows XP registry is an example of a hierarchical database.

Network DBMS:

Many-to-many relationships are supported by this database management system. It frequently leads to the creation of complicated database structures. The network model is implemented by RDM Server, a database management system.

Distributed databases:

This type of database includes information gathered by local computers as well as contributions from common database. The data in this database system isn't all in one place.

2.4 DATABASE COMPONENTS

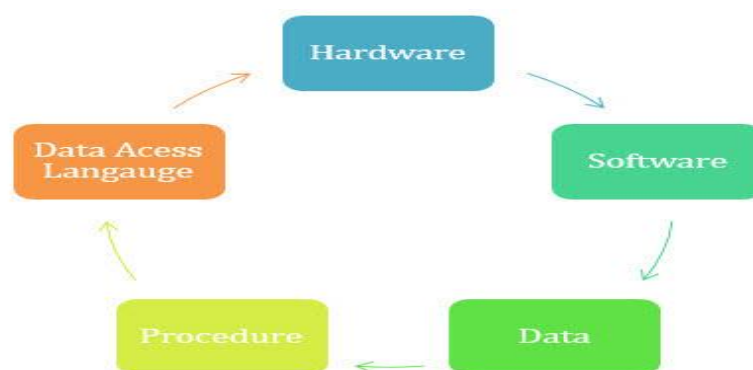


Figure 2.3 Database components

A database has below mentioned five main components:

Hardware:

The tangible, electronic devices like computers, I/O devices, devices for storage, and many more. Computers and our real-world system is connected together by use of hardware.

Software:

A collection of programmes used for managing and governing the database as a unit is called software. This includes the database software, the operating system, the network software allowing its users to exchange data, and the application programmes that allow users to access data in the database.

Data:

A fact that is not yet structured and processed and for which processing is required to make it significant is called Data. Data tends to stay simple and disorganized at the same time until organization. Some examples of data are facts, observations, perceptions, and numeric data.

Procedure:

A set of instructions that help you in usage of the database management system is called procedure. It includes development and execution of a database through the usage of predefined techniques so that the users can get help regarding its operation and administration.

Database Access Language:

A set of instructions that help you in usage of the database management system is called procedure. It includes development and execution of a database through the usage of predefined techniques so that the users can get help regarding its operation and administration.

2.5 SQL

MySQL is an open-source relational database management system. It was created with web apps in mind and can operate on any platform. As new and variable requirements were found with the internet, this slowly became the most chosen platform by most web developers and applications based on web. E commerce companies choose MySQL as they deal with huge number of money transfers and it can be used to deal with an enormous amount of queries and handle lots of transactions. MySQL's main feature is its on-demand flexibility.

To deal with relational databases, when it comes to the standard, nothing is at par with SQL language. SQL enables you to insert, search, update, and delete database records. Optimizing and maintaining database are some of the many tasks done by SQL.

2.5.1 USE OF SQL

Here are important reasons for using SQL

- It provides accessibility to the data in an RDBMS system.
- It allows to description of the data.
- It helps in defining and managing data inside the database.
- Creation, modification and deletion of databases and tables can be done.
- SQL enables usage of a database function, constructing a view, and executing a predefined procedure.
- Permissions related with tables, processes, and views can be modified.

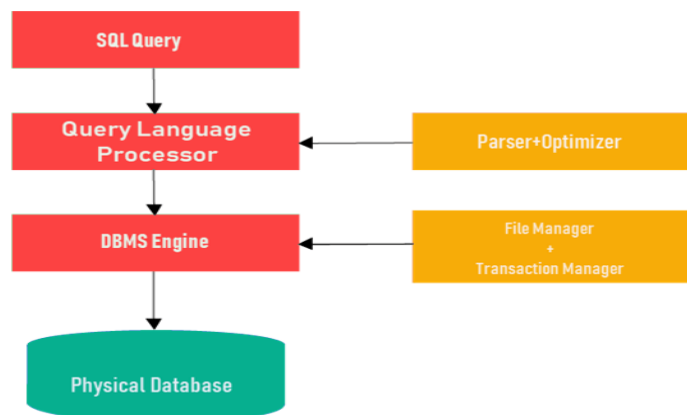


Figure 2.4 SQL Process

SQL process contains following components:

- SQL Query Engine
- Optimization Engines
- Query Dispatcher
- Classic Query Engine
- A classic query engine allows you to manage all the non-SQL queries.

2.5.2 HOW SQL WORKS?

The Client-Server Architecture is the basis of SQL. It allows the usage of network services in order to gather resources from the server for the user. Upon receiving a request from the user, the server quickly matches the instructions and provides the desirable results back to the user. The same client-server model is followed by SQL as the client-server model.

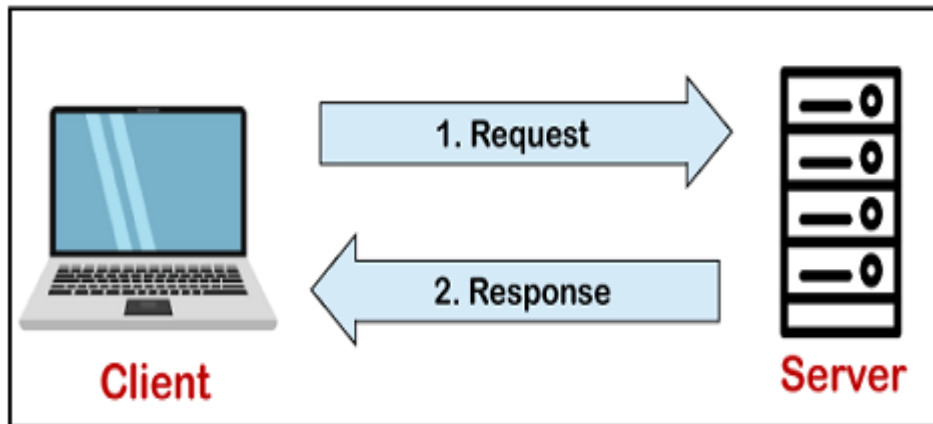


Figure 2.5 SQL Working

The MySQL Server is the heart of the MySQL database. This server is a stand-alone software that is in charge of managing all database instructions, statements, and commands. The following is how MySQL database and MySQL Server work together:

1. MySQL produces a database that allows you to create several tables for storing and manipulating data, as well as establishing the relationships between them.
2. Clients use specific SQL expressions on MySQL to perform requests via the GUI panel or command prompt.
3. Finally, the server programme will respond with the requested expressions and the intended client-side outcome.

Any MySQL GUI can be used by a client. However, to make your data management tasks faster and easier, make sure your GUI is lightweight and user-friendly.

MySQL Workbench, SequelPro, DB Visualizer, and the Navicat DB Admin Tool are some of the most popular MySQL GUIs. [9]

2.6 XML

XML refers to Extensible Markup Language. It is a text-based markup language derived from Standard Generalized Markup Language.

Unlike HTML tags, which indicate how to display the data, XML tags identify the data and are used to store and organize it, rather than determining how to show it. XML will not replace HTML in the near future, but it will open up new possibilities by incorporating many of HTML's great features.

XML has three key qualities that make it helpful in a wide range of systems and solutions—

- It is extensible, which means we can construct language and our own self-descriptive tags, as per our requirement.
- It transports data, not presents it. It allows to collect and store data not considering the way it will appear.
- It is a publicly available standard. It was formulated by the World Wide Web Consortium and is available online easily.

2.6.1 XML USAGE

A quick rundown of XML applications tells us—

- XML can simplify the generation of HTML documents for huge web sites by being the one that works continuously at the back end.
- XML is a format for exchanging data between companies and systems.
- We can be offload and reload database using XML.
- It is also used to collect, manage and organize data, enabling us to edit data as per our requirements.
- It may readily be combined with style sheets to produce nearly any output.[10]

2.6.2 DIFFERENCE BETWEEN XML AND HTML

Table 1.1 Difference between HTML and XML

HTML	XML
<ul style="list-style-type: none">• It is used for data displaying and its primary objective is focusing on the looks of data.• It is itself a markup interface.• It is not case sensitive.• It is a presentation language.• It makes use of its own set of predefined tags.• Using a closing tag is not required in HTML.	<ul style="list-style-type: none">• It is a tool free from any kind of hardware or software dependency and is used for transportation and storage of data.• It provides a working interface to the markup languages.• It is case sensitive.• It is neither a presentation nor a programming language.• Here tags are not predefined and can be changed accordingly.• Usage of a closing tag becomes necessary in XML.

2.7JSON

JSONS, or JavaScript Object Notation, is a simple text-based open standard for transferring human-readable data. Programmers are familiar with JSON conventions, which include C++, C, Java, and others.

- JSON is an acronym for JavaScript Object Notation.
- Douglas Crockford specified the format.

- It was created to exchange data in a human-readable format.
- The JavaScript scripting language has been extended.
- The .json file extension is used.
- JSON is a type of data format. Application/json is the Internet Media Type.
- The Uniform Type Identifier is open to the public.

2.7.1 USES OF JSON

- It's utilized in the development of JavaScript-based applications, such as browser extensions and webpages.
- JSON is a serialization and transmission format for structured data over a network connection.
- It's mostly used to send data from a server to web apps.
- JSON is the format used by web services and APIs to expose public data.
- It's compatible with today's programming languages.

2.7.2 CHARACTERISTICS OF JSON

1. It is simple to understand and write.
2. JSON is a text-based exchange format that is lightweight.
3. JSON is language agnostic.[11]

CHAPTER 3

VB Script

3.1 INTRODUCTION

Visual Basic Scripting, also known as VBScript, is a subset of Visual Basic for Applications (VBA). VBA is a Microsoft product that can be found in Microsoft products like MS Project and MS Office and in some other programmes as well.

3.2 FEATURES OF VBSCRIPT

- VBScript is known as a simple and efficient scripting language.
- VBScript is case insensitive. It is easy to implement and has the syntax is also easy and simple to learn and use.
- VBScript is an object-based scripting language, not an Object-Oriented Programming language, like C++ or Java.
- So as to access the elements of the environment in which it is running, it employs the Component Object Model.
- Only in a Host Environment, such as Internet Information Services (IIS), Internet explorer or Windows Scripting Host, can VBScript be successfully executed (WSH).

3.3 HISTORY AND USES

Microsoft released the first version of VBScript in 1996, and it was version 1.0. Now the recent stable version is 5.8, that is included with Internet Explorer 8 and Windows 7. The places where VBScript can be used are numerous and are not limited to the ones listed below.

- Quick Test Professional, abbreviated as QTP, is a popular automation testing application that uses VBScript as a scripting language.
- Windows Scripting Host is a tool for automating the Windows desktop that is mostly used by Windows System Administrators.

- Active Server Pages (ASP) is a server-side scripting environment that employs VBScript or Java Script to create dynamic webpages.
- In Microsoft Internet Explorer, VBScript is utilized for client-side scripting.
- Microsoft Outlook Forms usually runs on VBScript; however, the application-level programming relies on VBA.[12]

3.4 WORK DONE

Example 1:

Problem Statement:

Write a procedure, which will accept an integer value as input and displays the factorial for the given number.

Figure 3.1 Problem statement 1

Solution1: CODE:

```
option explicit

dim n,f,m
n= InputBox("Enter the number")
m=n
f=1

Do while n>0
f=f*n
n=n-1
Loop
msgbox "Factorial of" & m & " is "& f
```

Figure3.2Code 1

OUTPUT:



Figure3.3 Output 1.1

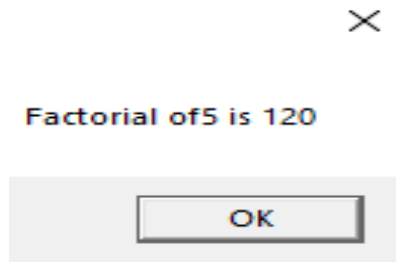


Figure3.4 Output1.2

Example 2:

Problem Statement:

Write a procedure which will accept a number from 1 to 12 as input parameter and output should be the month name. For example if the input is 1, then the output should be "January", if the input is 2 then the output should be "February" and so on.

Figure3.5 Problem statement 2

Solution 2: CODE

```
dim month
month=InputBox("Enter month number")
Select case month
case 1
MsgBox "January"
case 2
MsgBox "February"
case 3
MsgBox "March"
case 4
MsgBox "April"
case 5
MsgBox "May"
case 6
MsgBox "June"
case 7
MsgBox "July"
case 8
MsgBox "August"
case 9
MsgBox "September"
case 10
MsgBox "October"
case 11
MsgBox "November"
case 12
MsgBox "December"
End select
```

Figure 3.6 Code 2

OUTPUT :



Figure 3.7 Output 2.1

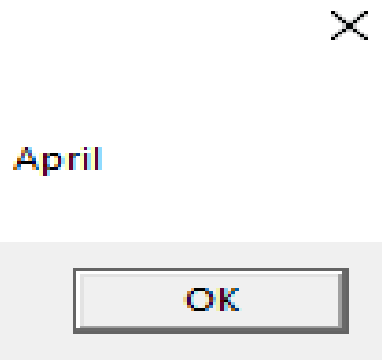


Figure3.8 Output 2.2

Example 3:

Problem Statement:

Write a program to handle Array out of Index error using error handling techniques

Figure 3.9 Problem statement 3

Solution 3: CODE


```

a=Array("a","b","c","d","e")
for each element in a
  msgbox element
next
On Error Resume Next
MsgBox a(5)="f"
Err.Raise 101
MsgBox "Error " & err.number
err.clear
msgbox "Error " & Err.Number & " END "

```

Figure 3.10 Code 3

OUTPUT:

×

a

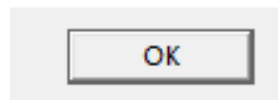


Figure 3.11 Output 3.1

×

b

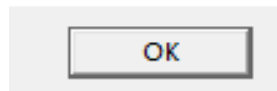


Figure 3.12 Output 3.2

×

c

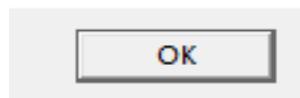


Figure 3.13 Output 3.3

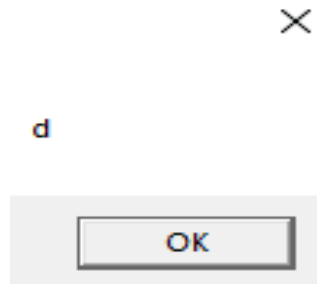


Figure 3.14 Output 3.4

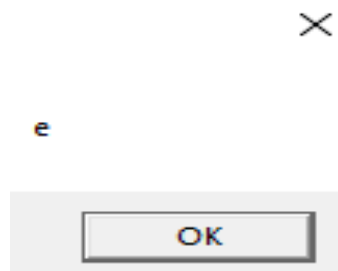


Figure 3.15 Output 3.5

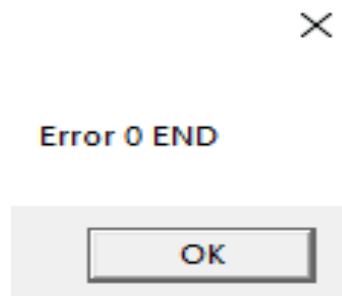


Figure 3.16Output 3.6

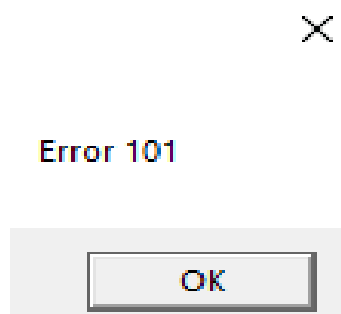


Figure 3.17 Output 3.7

Example 4:

Problem Statement:

Write a Program to replace all the numbers with „*“ in a given string using regular expression Examples:

Input String : " Hello 123 , How are you 1" Output String : " Hello *** , How are you *"

Figure 3.18 Problem statement 4

Solution 4: CODE

```
Dim oRE, oMatches
Set oRE = New RegExp
oRE.Pattern = "Inbox\([0-9]\)"
oRE.Global = True
oRE.IgnoreCase = True
Set oMatches = oRE.Execute("two times Inbox(0) three equals Inbox(3) three times two")
For Each oMatch In oMatches
    If (oMatch="Inbox(0)") Then
        MsgBox("Inbox")
    Else
        MsgBox "Match: " & oMatch.Value & " At: " & CStr(oMatch.FirstIndex + 1)
    End If
Next
```

Figure 3.19 Code 4

OUTPUT:

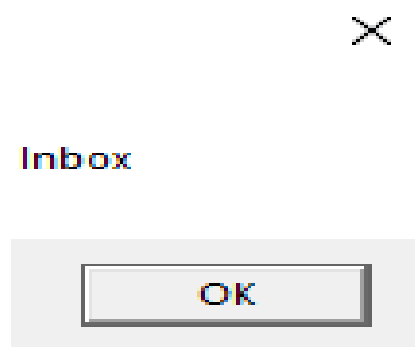


Figure 3.20 Output 4.1



Match: Inbox(3) At: 33



Figure 3.21 Output 4.2

Example 5:

Problem Statement:

Write a function which will create a directory in a particular path if the directory does not exist.

Eg: C:\test.

Program should check whether "Test" exists and if it does not exist it should create a folder.

Figure 3.22 Problem statement 5

Solution 5: CODE

```
Dim ofso, startfolder
startfolder="D:\abcd"
set ofso=CreateObject("Scripting.FileSystemObject")
if ofso.FolderExists(startfolder) then
msgbox startfolder &"already exists",0,"Result."
else
ofso.CreateFolder(startfolder)
MsgBox strfolder &"created.",1,"Status"
end if
```

Figure 3.23 Code 5

OUTPUT:

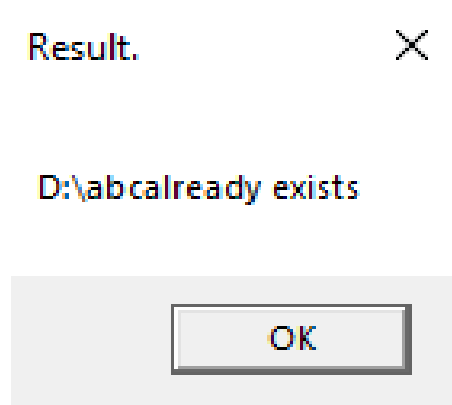


Figure 3.24 Output 5

Example 6:

Problem Statement:

Write a function to generate a unique number. The number should be the combination of : YYYYMMDDHHMMSS

Figure 3.25 Problem statement 6

Solution 6: CODE

```
option Explicit
dim a,curr_date,curr_time,y,d,m,hr,min,sec
a=Unique

Function Unique()
curr_date=Date
curr_time=Time
y=Year(curr_date)
m=Right("00"&Month(curr_date),2)
d=Right("00"&Day(curr_date),2)
hr=Right("0"&Hour(curr_time),2)
min=Right("0"&Minute(curr_time),2)
sec=Right("0"&Second(curr_time),2)
MsgBox y&m&d&hr&min&sec

End Function
```

Figure 3.26 Code 6

OUTPUT:

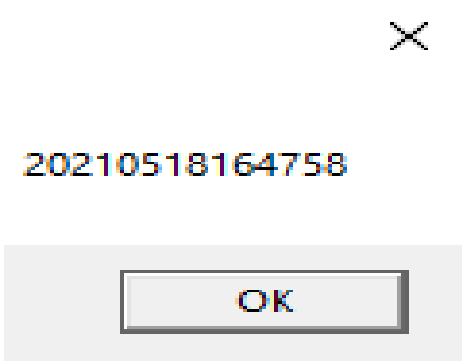


Figure 3.27 Output 6

CHAPTER 4

UFT AUTOMATION

4.1 INTRODUCTION



Figure 4.1 UFT

UFT is a term used for an automated functional Testing tool that comes in aid of testers to execute automated tests in order to detect any issues, defects or errors in comparison to the expected results of the product. Full form of UFT is Unified Functional Testing and it is also known as QTP which means Quick Test Professional.

4.2 WHY UFT?

- It's an icon-based tool that automates an application's regression and functional testing.
- Micro Focus QTP can be used by both technical and non-technical testers.
- We can test both desktop and web-based applications.
- It has both recording and playback capabilities
- It enables Business Process Testing (BPT)
- Unified automation Testing is based on the scripting language VB script
- VBScript is used by Micro Focus' UFT to automate programs.
- It supports the largest pool of software development environments like SAP, Oracle etc.
- The UFT tool helps the testers to perform an uninterrupted automated functional testing.

4.3 ADVANTAGES

- It supports features such as playback and record
- It features an effective object identification method or mechanism. • It employs a feature that enables to record an active screen.
- It has excellent mechanism and identification process of objects
- It works with a variety of add-ins, including Java, NET, Oracle SAP, PeopleSoft, and webform etc.
- Using an active screen, you can improve existing tests even if you don't have the AUT
- It has an integrated development environment (IDE).
- It's compatible with test management software such as Winrunner, Quality Center and Test Director
- Different types of test suites like Regression, Smoke and Sanity can be easily looked after.
- Easy to run and maintain [13]

4.4 WORK DONE

Example 1:

Problem Statement:

Verify availability of add-ins. How to select and de-select an add-in. Try to unselect Web add-in and identify web-based objects. Compare the observations between Web add-in is selected and not selected while identifying objects in web page.

Also Use Generate script to understand configuration of UFT

Figure 4.2 Problem statement 1

Solution 1:

1. Start UFT one
2. The Unified Functional Testing Add-in Manager Dialog Box opens.

3. Select the web add in as shown below

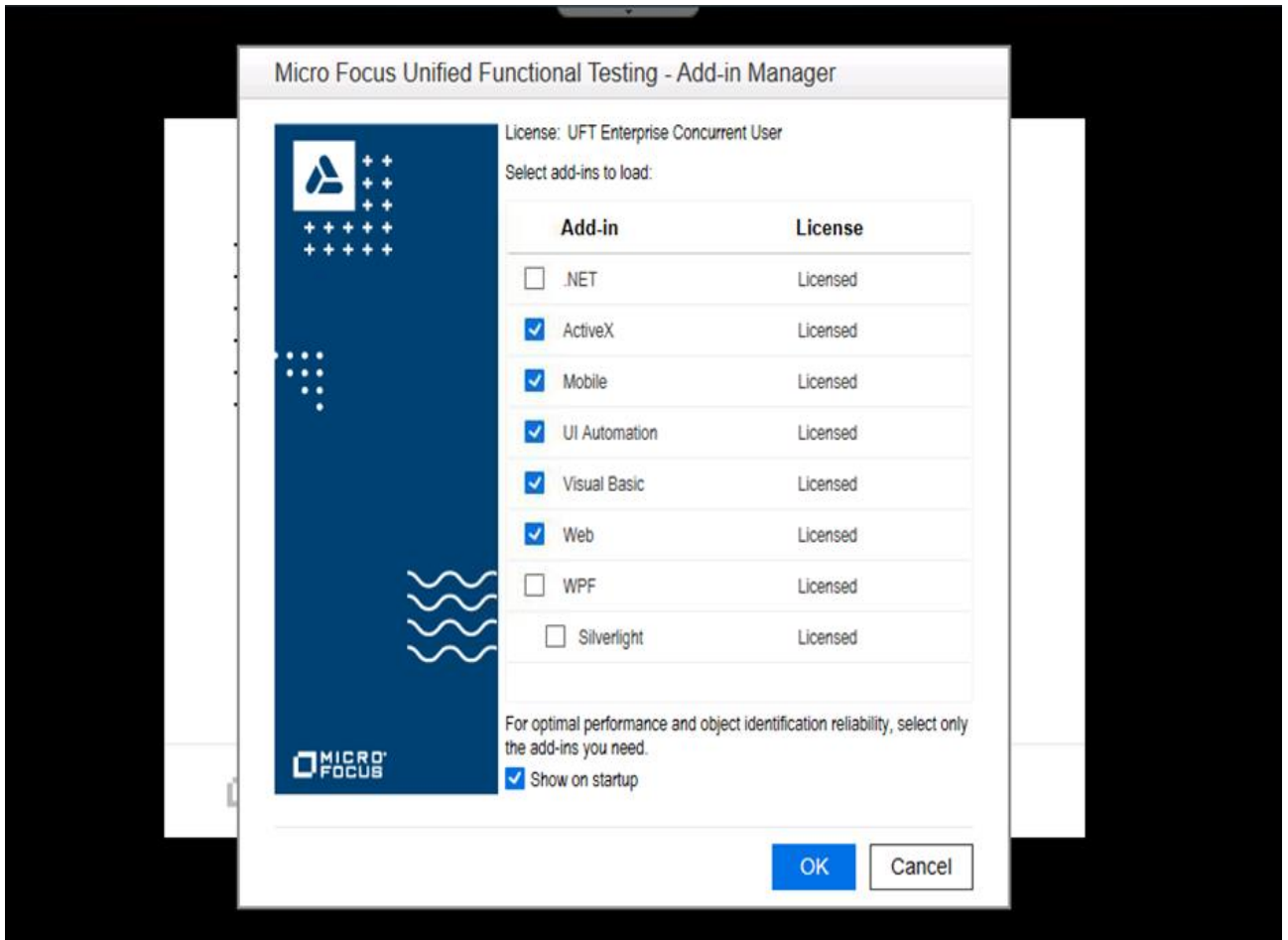


Figure 4.3 Select web

4. Now go to record and run settings

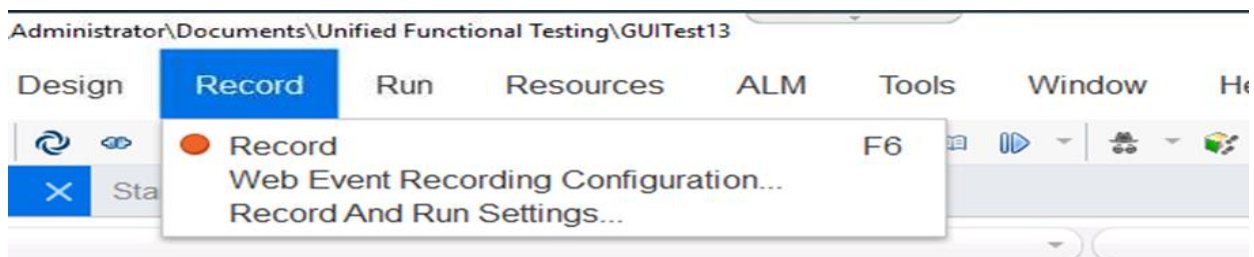


Figure 4.4 Record

5. It can be seen that the web objects are shown in the image below

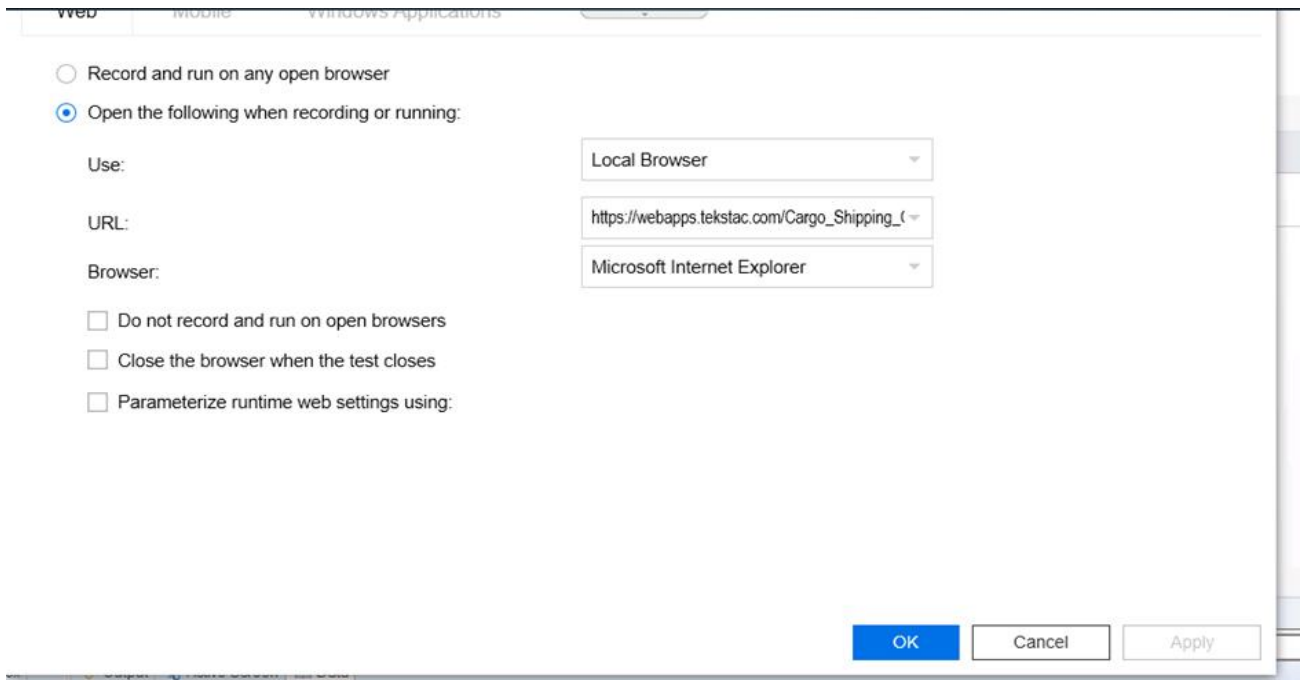


Figure 4.5 Web objects

6. Now close the application and open it again and deselect the web add in as shown below

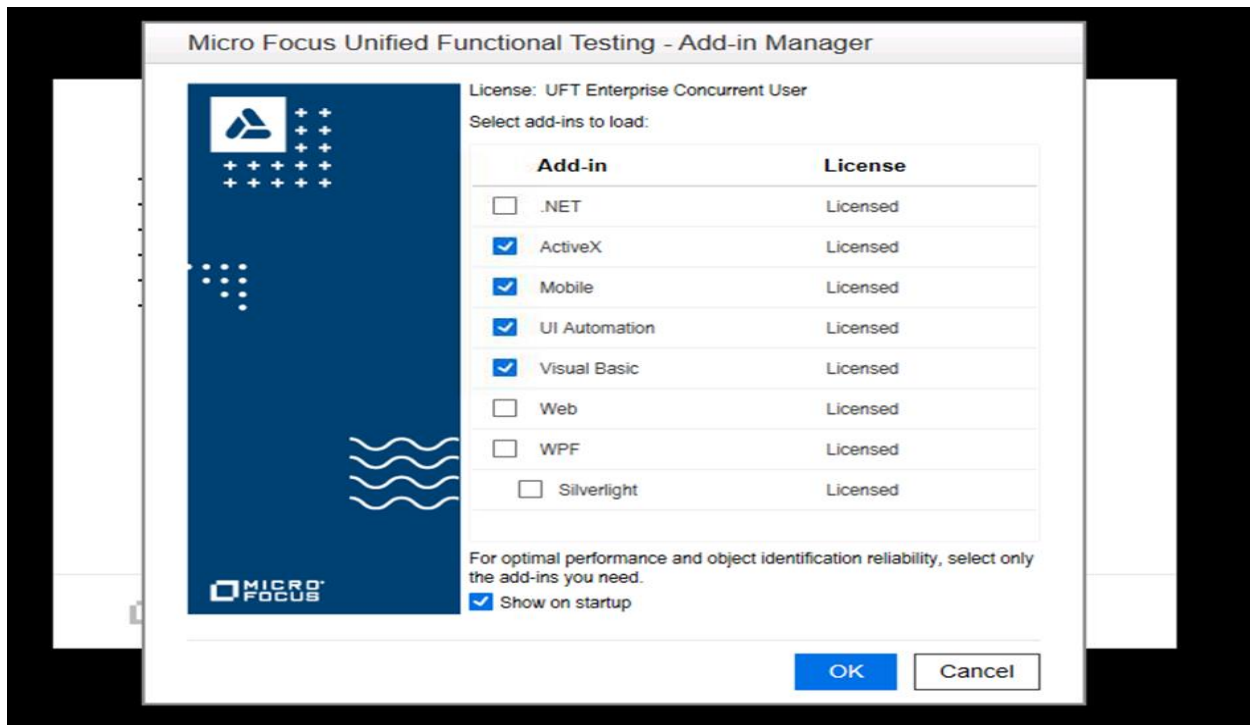


Figure 4.6 Deselect Web

7. Go to record and run settings and we can see that the web objects are not present

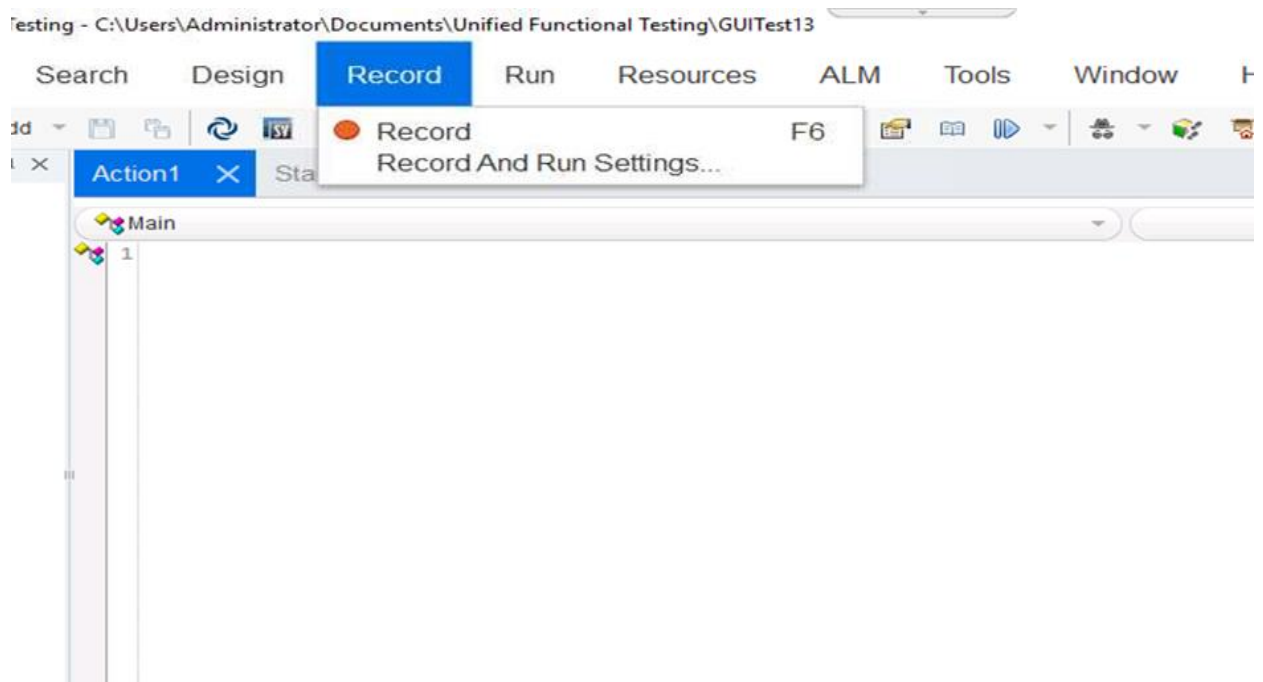


Figure 4.7 Record and run

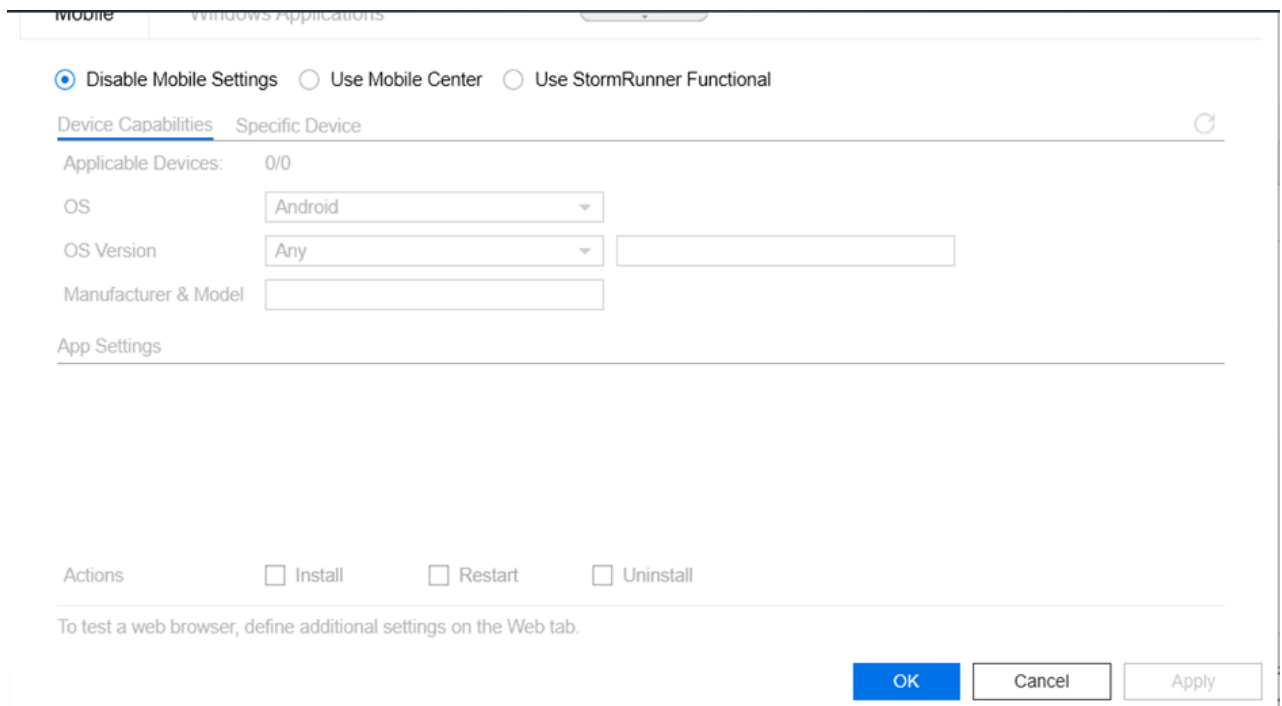


Figure 4.8 Disable mobile settings

8. For object identification go to tools>object identification

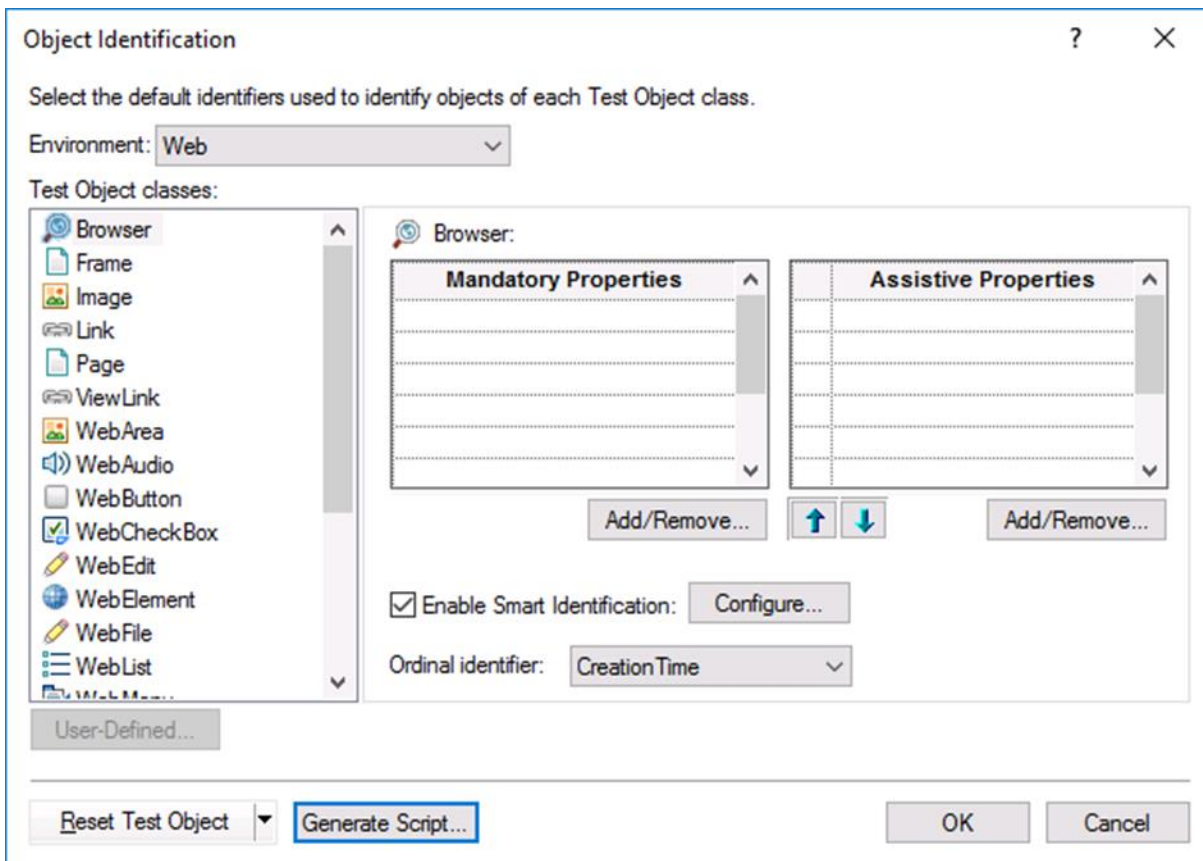


Figure 4.9 Object identification

CONCLUSION

This learning experience has been very enlightening. UFT instructional movies and UFT fundamentals have made simple via QA Eversity HP Unified Functional Testing (UFT) does break the bank however it is nonetheless a marketplace chief within the automation checking out business. Nowadays, many open supply and coffee-value automation equipment got here into play however QTP and UFT are nonetheless widespread for his or her complex options and controls.

UFT supplies a greater regulate over our automation procedure and it got here up with many complex and helpful options QTP didn't have. It is appropriate for each other folks with non-programming and programming backgrounds.

UFT helps VB script for personalization and higher regulate which is straightforward to be informed.

REFERENCES

- [1] [HTTPS://WWW.EDUREKA.CO/BLOG/WHAT-IS-SOFTWARE-TESTING/](https://www.edureka.co/blog/what-is-software-testing/)
- [2] [HTTPS://REQTEST.COM/TESTING-BLOG/THE-A-TO-Z-GUIDE-TO-THE-SOFTWARE-TESTING-PROCESS/](https://reqtest.com/testing-blog/the-a-to-z-guide-to-the-software-testing-process/)
- [3] [HTTPS://WWW.TOOLSQA.COM/SOFTWARE-TESTING/TEST-PROCESS-IN-SOFTWARE-TESTING/](https://www.toolsqa.com/software-testing/test-process-in-software-testing/)
- [4] [HTTPS://WWW.SOFTWARETESTINGHELP.COM/TYPES-OF-SOFTWARE-TESTING/](https://www.softwaretestinghelp.com/types-of-software-testing/)
- [5] <https://www.simplilearn.com/tutorials/agile-scrum-tutorial/what-is-agile>
- [6] <https://www.guru99.com/agile-scrum-extreme-testing.html>
- [7] <https://www.britannica.com/technology/database>
- [8] <https://www.guru99.com/introduction-to-mysql-workbench.html>
- [9] <https://www.javatpoint.com/mysql-tutorial>
- [10] https://www.tutorialspoint.com/xml/xml_overview.htm
- [11] https://www.tutorialspoint.com/json/json_overview.htm
- [12] <https://www.tutorialspoint.com/vbscript/index.htm>
- [13] <https://www.guru99.com/uft-qtp-automation-testing.html>

PUBLICATIONS