



# **T. Y. B. Sc.**

## **(Computer Science)**

### **SECC III**

#### **Software Testing Tools - CS 3610**

### **Semester VI**

#### **(From Academic Year 2021)**

Name

Roll No. \_\_\_\_\_

College

Division \_\_\_\_\_

Academic Year \_\_\_\_\_

## **PREPARED BY:**

**PROF. Dr. MANISHA M. PATIL**

(Indira College of Commerce and Science, Pune.)

**PROF. RAJESH V. NIKAM**

(MVP's K.K.W. Arts, Science & Commerce College, Pimpalgaon (B), Dist Nashik.)

## ABOUT THE WORKBOOK

### □ OBJECTIVES OF THIS BOOK

This lab-book is intended to be used by T.Y.B.Sc(Computer Science) students for SECC-III Software Testing Tools, CS-306 , Semester VI.

The objectives of this book are

- a. Covers the complete scope of the syllabus.
- b. Bringing uniformity in the way course is conducted across different colleges.
- c. Continuous assessment of the students.
- d. Providing ready references for students while working in the lab.

### • How to use this book?

This book is mandatory for the completion of the laboratory course. It is a measure of the performance of the student in the laboratory for the entire duration of the course.

### • Instructions to the students

1. Students should carry this book during practical sessions.
2. Print outs of source code and outputs is optional
3. Student should read the topics mentioned in Reading section of this book before coming for the practical session.
4. Students should solve those exercises which are selected by Practical in-charge as a part of journal activity. However, students are free to solve additional exercises for more practice.
5. Each assignment will be assessed on a scale of 0 to 5 as indicated below.
  - Not done 0
  - Incomplete 1
  - Late Complete 2
  - Needs improvement 3
  - Complete 4
  - Well Done 5

### • Difficulty Levels

Self-Activity: Students should solve these exercises for practice only.

SET A - Easy: All exercises are compulsory.

SET B - Medium: All exercises are compulsory.

### • Instruction to the Instructors

- 1) Make sure that students follow the instruction as given above.
- 2) After a student completes a specific set, the instructor must verify the outputs and sign in the space provided after the activity.
- 3) Evaluate each assignment on a scale of 5 as specified above by ticking the appropriate box.
- 4) The value should also be entered on the assignment completion page of the respective Lab course.
- 5) Students should be encouraged to use Word, Excel Tool, and Selenium IDE for their assignments.
- 6) Case Study based teaching approaches like desktop, mobile and web applications
  - 1) Simple Calculator, Notepad, Word pad program – Desktop apps
  - 2) Calculator, Air Ticket Booking, Online shopping – Mobile apps
  - 3) Railway reservation, College Website – Admission, exam, results, events etc. Web apps.

**Roll No :** \_\_\_\_\_

**Name :** \_\_\_\_\_

### **Assignment Completion Sheet**

Sr. No	Assignment Name	Marks
1	Introduction to Test case design: Excel Tool	
2	Test cases for simple programs	
3	Test cases and Test plan (Word)	
4	Defect Report (Word)	
5	Testing Tools – Selenium IDE	
	Total out of 20	
	Total out of 5	

Signature of Incharge:

Examiner I:

Examiner II:

Date:

## Assignment 1: Introduction to Test case design: Excel Tool

### Objectives

Design entry and exit criteria for test case, design Manual test cases in excel.

### Reading

#### **Example : Registration and Login page – Manual Test Cases**

Writing test cases (which is an important task for a tester in application testing) requires experience and a keen eye for detail to build scenarios. This means that an application tester needs to focus on creating a set of variables or conditions which will help him to determine whether the software meets requirements and functions lawlessly. To reduce that hassle, we have compiled a list of use cases that you need to test a signup or login page. We hope they will help you to create more error-free Register/Login pages.

One should gather all the customer requirements you can. The list of requirements which can be useful are:

1. FirstName
2. Confirm Password
3. LastName
4. Address
5. UserName
6. PhoneNumber
7. Email ID
8. Gender
9. Password

Required fields are:

1. Email ID
2. UserName
3. Password
4. Confirm Password

Validation for fields:

1. Email
2. Password: It should have alphanumeric; Length should be 8 to 32.
3. PhoneNumber: Phone Number should have only Numbers, Country code is required

After successful Verification email should send to the user On Required fields \* should show.

Now, let's begin with our test cases. Although most testers use Bugzilla or other test management tools to maintain test cases, you can also use Excel or spreadsheets.

Ready Reference

### Test Cases for Registration Page

Signup and login page by assuming some client requirements, such as:

- Username and password are mandatory fields
- There is cancel and reset button at the bottom of the form
- Radio buttons and checkboxes are placed correctly
- The limit of the Password should be 8-13 characters (alphanumeric).

Sr No.	Test Cases	Feature	Description	Steps To Execute	Test Data / Input	Expected Results
1	TC-001	User Interface	Check all the text boxes, radio buttons, buttons, etc	1. Click on Radio buttons and dropdowns	N/a	UI should be perfect
2	TC-002	Required fields	Check the required fields by not filling any data	1. Do not enter any value in the field. 2. Click on the Register	N/a	It should show a mandatory symbol (*) on mandatory fields.

Sr No.	Test Cases	Feature	Description	Steps To Execute	Test Data / Input	Expected Results
				button.		
3	TC-003	Required fields	Check user should Register by filling all the required fields	1. Enter valid values in the required fields. 2. Click the Register button.	N/a	1. Users should be registered successfully. 2. A successful registration message should show. 3. Mail should send to the user
4	TC-004	Optional Fields	Check all the optional fields when do not fill data	1. Do not enter any detail in optional fields 2. Enter valid data in required fields 3. Click on the Signup button	N/a	1. It should not ask to fill the optional fields 2. User should be registered successfully 3. A successful registration message should show 4. Mail should send to the user

Sr No.	Test Cases	Feature	Description	Steps To Execute	Test Data / Input	Expected Results
						user
5	TC-005	Optional Fields	Check all the optional fields when filling data	1. Enter valid data in optional fields 2. Enter valid data in required fields 3. Click on the Register button	N/a	1. User should be registered successfully 2. A successful registration message should show 3. Mail should send to the user



6	TC-006	Email validation	<ul style="list-style-type: none"> <li>Check the Email text field that has an Email address without @ symbol.</li> <li>Check the Email text field that has a random string instead of a real email.</li> <li>Check the Email text field that has @ symbol written in words.</li> <li>Check the Email text field that has a</li> </ul>	1. Enter Invalid Emails 2. Click the on Register Button.	1.testAtgmail.com 2.test@gmailcom 3.test@gmail 4.@gmail	It should show the validation message for valid email
---	--------	------------------	---	---	--	---

Sr No.	Test Cases	Feature	Description	Steps To Execute	Test Data / Input	Expected Results
			missing dot in the email address.			
7	TC-007	Email validation	Check all the valid emails	1. Enter valid Emails 2. Click on the Register Button.	1.test.22@gmail.com 2.test@gmail.com	It should not show any validation message

8	TC-008	Phone Number validation	Check the phone number when passing alphanumeric data	1. Enter alphanumeric data in phone field 2. Click on Register button	1. dada5\$7567#7	It should show the validation message 8 for Phone Number
9	TC-009	Phone Number validation	Check the phone number when not pass country code	1. Enter valid phone number without country code 2. Click on Register button	1. 9012078654	It should show the validation message for country code is required
10	TC-010	Phone Number validation	Check the phone number when passing country code	1. Enter valid phone number with country code 2. Click on	1. +9190112244	It should not show any validation message
Sr No.	Test Cases	Feature	Description	Steps To Execute	Test Data / Input	Expected Results
				Register button		

11	TC-011	Password Validation	Check the password limit when enter value less than min	1. Enter value which is alphanumeric but less than 8. 2. Click on Register button	1. Password	It should show validation message
12	TC-012	Password Validation	Check the password limit when enter value greater than max	1. Enter alphanumeric value but more than 32. 2. Click on Register button	Any Random string with numbers	It should show validation message
13	TC-013	Password Validation	Check the password when passing only numbers	1. Enter a value in numbers which is in between 8-32 2. Click on Register button	1. 12345678	It should show validation message
14	TC-	Password	Check the	1. Enter value	1. Pass123456	It should not

Sr No.	Test Cases	Feature	Description	Steps To Execute	Test Data / Input	Expected Results
	014	Validation	password when passing valid data	in alphanumeric which is in between 8-32 2.Click on Register button		show any validation message
15	TC-015	Required Fields	Verify if blank spaces are passed in required fields.	1. Go to the Site. 2. Passed blank spaces in required fields. 3. Click on the Register button	N/a	Those Blank spaces should trim and Validation error message for required fields should visible.
16	TC-016	Required Fields	Verify user can verify its Email ID	1. Go to the Email. 2. Click on the verification link.	test22@gmail.com	User should get a verification link and able to verify his/her Email ID.

17	TC-017	Phone Number Validation	Verify if the length of the phone number is incorrect i.e. less than 10.	1. Enter phone number less than 10 digits.	91901122	It should show the validation error message for phone
<b>Sr No.</b>	<b>Test Cases</b>	<b>Feature</b>	<b>Description</b>	<b>Steps To Execute</b>	<b>Test Data / Input</b>	<b>Expected Results</b>
			than 10.	2.Enter all required fields. 3.Click on Register Button		number length.
18	TC-018	Phone Number Validation	Verify if the length of the phone number is incorrect i.e. more than 10	1. Enter phone number less than 10 digits. 2.Enter all required fields. 3.Click on Register Button	91901122445566	It should show the validation error message for phone number length.

19	TC-019	Password Validation	Verify if the password required rules are not satisfied in the password	1. Enter the password which not satisfies the required rule. 2. Click on Register button	passw	It should display error with required rules for password value (like it should contain a special character, a small case, a number)
----	--------	---------------------	---	---	-------	---

Self-Activity ( using Excel Tool)

#### Test Cases for Login Page

Sr No.	Test Cases	Feature	Description	Steps To Execute	Expected Results
1	TC-01	User Interface	Check all the text boxes and buttons	Check Page	<ul style="list-style-type: none"> <li>• UI should be perfect</li> <li>• Text boxes and button should be aligned</li> </ul>

2	TC-02	Required Fields	Check the required fields by not filling any data.	1. Enter invalid username 2. Enter correct password 3. Click on Login Button	User should not login and should show proper error message
3	TC-03	User Login	Check When passing a correct username and invalid password	1. Enter valid username 2. Enter incorrect password 3. Click on Login Button	User should not login and should show proper error message
4	TC-04	User Interface	Check Keeping Password	1. Enter valid username 2. Do not enter password 3. Click on Login Button	User should not login and should show proper error message

Sr No.	Test Cases	Feature	Description	Steps To Execute	Expected Results
--------	------------	---------	-------------	------------------	------------------

5	TC-05	User Login	Check when pass correct email and password	1. Enter valid username 2. Enter valid password 3. Click on Login Button	User should log in
6	TC-06	User Login	Check if the password is entered in encrypted	1. Enter valid username 2. Enter password 3. Click on Login Button	Password is entered in encrypted form



7	TC-07	Signup Option for new users	Check whether the signup link for the new user is working	Click Signup link	Clicking signup link takes the user to signup page successfully
8	TC-08	Forgot Password	Verify user should get an error message when he/she enters not registered email id.	1. Click on the Forgot password link. 2. Enter unregistered email id and click on the send button.	User should get an error message.
9	TC-09	Reset Password	Verify user should get an error message when he/she enters the previous password.	1. Go to the reset password link. 2. Enter the previous password.	User should get an error message.
<b>Sr No.</b>	<b>Test Cases</b>	<b>Feature</b>	<b>Description</b>	<b>Steps To Execute</b>	<b>Expected Results</b>
				3. Click on the Reset Password button.	

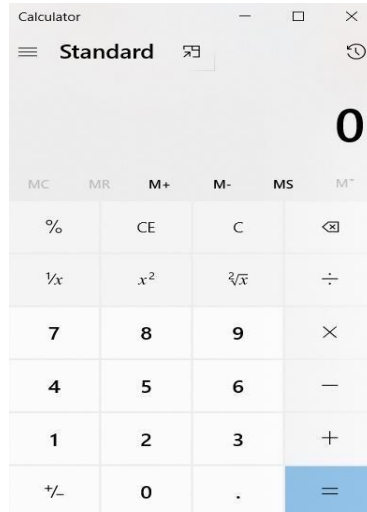
10	TC-10	Reset Password	Verify user able to reset his/her password	1. Go to the reset password link. 2. Enter a new password and a confirm password. 3. Click on the Reset Password button.	Users should get the success message and the password should get reset.
11	TC-11	Reset Password	Verify user should get an error message when password and confirm password not matches	1. Go to the reset password link. 2. Enter a different new password and a confirm password. 3. Click on the Reset Password button.	Users should get an error message.
12	TC-12	Reset Password	Verify user should able to login with a new password.	1. Go to the reset password link. 2. Enter a new password and a confirm password. 3. Click on the Reset Password	User should able to login

Sr No.	Test Cases	Feature	Description	Steps To Execute	Expected Results
				button. 4. Log in by using the new password.	
13	TC-13	Reset Password	Verify if the user enters a new password that does not cover the basic requirements of password then the user should be displayed error message	1. Go to the reset password link. 2. Enter a new password that does not cover the basic requirements. 3. Click on the Reset Password.	Users should get an error message.
14	TC-14	Required Fields	Verify if blank spaces are passed in required fields.	1. Go to the Site. 2. Passed blank spaces in required fields. 3. Click on the Login button	Those Blank spaces should trim and Validation error message for required fields should visible.
15	TC-15	Welcome Email	Verify new users should get the welcome email once after the login.	1. Go to the Email. 2. Enter Login Email.	Users should get a welcome email on his/her email id.

Sr No.	Test Cases	Feature	Description	Steps To Execute	Expected Results
16	TC-16	User Login	Verify when passing incorrect Email and correct password	<ol style="list-style-type: none"> <li>1. Enter incorrect Email.</li> <li>2. Enter the correct password.</li> <li>3. Click on the Login Button.</li> </ol>	User should not be able to log in and the error message should be displayed.
17	TC-17	User Login	Verify when passing both incorrect Email and password	<ol style="list-style-type: none"> <li>1. Enter incorrect Email.</li> <li>2. Enter the correct password.</li> <li>3. Click on the Login Button</li> </ol>	User should not be able to log in and the error message should be displayed.
18	TC-18	User Forgot Password.	Verify Forgot Password sends a forgot password link.	<ol style="list-style-type: none"> <li>1. Click on the Forgot Password link.</li> <li>2. Enter Email and click on the send button.</li> <li>3. Now go to mail7.io and enter the email id.</li> </ol>	User should get the forgot password link on his/her email id.

## SET A

Q1. Design entry and exit criteria for test case, design Manual test cases in excel for Simple Calculator Application.



Q.2 Design entry and exit criteria for test case, design Manual test cases in excel for Online Air Ticket Booking / Railway Reservation Form

PNR STATUS

CHARTS / VACANCY

### BOOK TICKET

✈ From

⇅

📍 To

DD/MM/YYYY \*

20/08/2021

📅 All Classes

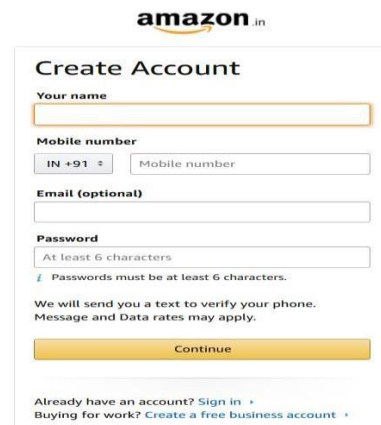
🏠 GENERAL

☐ Divyaang Concession
 ☐ Flexible With Date
 ☐ Train with Available Berth
 ☐ Railway Pass Concession

Search

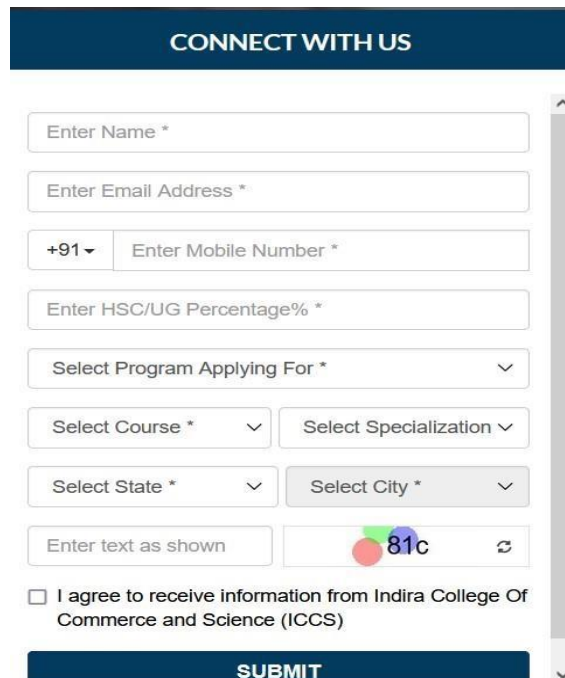
## SET B

Q1. Design entry and exit criteria for test case, design Manual test cases in excel for E-Commerce shopping portal's Login form (like Flipkart, Amazon)



The screenshot shows the 'Create Account' form on the Amazon.in website. It includes input fields for 'Your name', 'Mobile number' (with a country code dropdown set to 'IN +91'), and 'Email (optional)'. There is a 'Password' field with a note 'At least 6 characters' and a hint 'Passwords must be at least 6 characters.' Below the fields, a message states 'We will send you a text to verify your phone. Message and Data rates may apply.' A yellow 'Continue' button is at the bottom. At the very bottom, there are links for 'Already have an account? Sign in' and 'Buying for work? Create a free business account'.

Q.2 Design entry and exit criteria for test case, design Manual test cases in excel for Web pages of any website / College / University website



The screenshot shows a 'CONNECT WITH US' form on a college website. It features several input fields: 'Enter Name \*', 'Enter Email Address \*', and 'Enter Mobile Number \*' (with a '+91' dropdown). There are also fields for 'Enter HSC/UG Percentage% \*', 'Select Program Applying For \*' (dropdown), 'Select Course \*' (dropdown), 'Select Specialization \*' (dropdown), 'Select State \*' (dropdown), and 'Select City \*' (dropdown). A CAPTCHA field with the text 'Enter text as shown' and a visual of the number '81c' is present. Below the CAPTCHA is a checkbox labeled 'I agree to receive information from Indira College Of Commerce and Science (ICCS)'. A blue 'SUBMIT' button is at the bottom.

**Signature of the Instructor:**

**Date:**

**Assignment Evaluation:**

{Not done 0 , Incomplete 1 , Late Complete 2 , Needs improvement 3 , Complete 4 , Well Done 5 }

## Assignment 2 : Test cases for simple programs

### Objectives

Write programs and Design test cases for following programming statements.

For, while Do... While Loops, If ...Else , Switch ...Case

### Reading

Sample Code:

```
<html>
<head>
<script type="text/javascript">
function display()
{
console.log("Hello");
var n=Number(document.getElementById("num").value);

if(n%2==0)
{
document.writeln("Number is EVEN");
}
else
{
document.writeln("Number is ODD");
}

document.writeln("Number 1 to "+n+"is:");

for(i=1;i<=n;i++)
{
document.writeln(i);
}
document.writeln("

Displaying Number in Words:
");
switch(n)
{
case1: document.writeln("One");
break;

case2:
document.writeln("Two");
break;
case3:
document.writeln("Three");
break;
```

```

        case4: document.writeln("Four");
break;

        case5:
document.writeln("Five");
        break;
        default: document.writeln(" To display number in
words enter less than 5 ");
        break;
    }
}
</script>
</head>

<body bgcolor="grad">
Enter any Number:
    <input type="text" id="num"/>
    <button onclick="display()"> Display
</button> </body> </html>

```

Ready Reference and Self-Activity ( using Excel Tool)

Sample

Test Cases:

TC_NAME	STEPS	INPUT DATA	EXPECTED RESULT	ACTUAL RESULT	STATUS
IF	1) Enter even Number	6	It should go to the IF Part	Going to IF Part and Displaying "Number is EVEN"	Pass
ELSE	1) Enter odd Number	7	It should go to the ELSE Part	Going to ELSE Part and Displaying "Number is ODD"	Pass
IF	1) Enter even Number	3	It should go to IF Part	Going to ELSE Part and Displaying "Number is ODD"	Fail
ELSE	1) Enter odd Number	4	It should go to the ELSE Part	Going to IF Part and Displaying "Number is EVEN"	Fail
FOR	1) Enter Number as 10	10	It should display 1 to 10 Number	Going to FOR Loop and Displaying 1 to 10 Numbers	Pass
Switch	1) Enter Number less than 5	4	It should go to case 4	Going to Case 4 and Displaying FOUR	Pass
Switch	1) Enter Number less than 5	6	It should go to default part	Going to default part	Pass

## Lab Assignments

### Set A

Q1. Write a Java Program to calculate Factorial of a given Number. Write Test cases for this code.

Q.2 Write a Java Program to calculate maximum of given 2 numbers. Design test cases for this code.

### Set B

Q.1 Write a Java Program to calculate Fibonacci Series for given input number. Write Test cases for this code.

Q.3 Write a java Program to calculate Grades for students as per marks scored . Write Test cases for this code.

**Signature of the Instructor:**

**Date:**

**Assignment Evaluation:**

{Not done 0 , Incomplete 1 , Late Complete 2 , Needs improvement 3 , Complete 4 , Well Done 5}



## Assignment 3 : Test cases and Test plan (Word)

### Objectives

### Design Test Cases and Test Plan

### Reading

#### Test Cases :

We can see that often the experienced and the fresher tester is asked to write the test cases for the calculator. Many times, the type of the calculator is not mentioned. Test cases for the calculator depends on the type – scientific, simple calculator, financial calculator, or some other specific calculator like programmable calculator.

For the sake of simplicity, we are choosing simple calculator for the test cases. You can come up with plenty of test cases to test calculator based on the mathematical computation. However, we are here testing the calculator to see if it is fit for consumer use. So, we must make sure that it fits for the most obvious tests. If those tests are passed, then we can safely assume that product is fit for use.

We can use **calculator** as an example for writing test cases.

#### Basic Operational Tests

Write the test cases based on the following functions and scenarios.

- Check the calculator if it starts by on button. If it is software based calculator then check if it starts via specific means like from searching for calculator in search bar and then executing application, or by accessing menu item in the Windows.
- Check if the calculator window maximizes to certain window size.
- Check the if the calculator closes when the close button is pressed or if the exit menu is clicked from file > exit option.
- Check if the help document is accessed from Help > Documentation.
- Check if the calculator allows copy and paste functionality.
- Check if the calculator has any specific preferences.
- Check if all the numbers are working ( 0 to 9)
- Check if the arithmetic keys ( +, -, \*, %, /) are working.
- Check if operator is pressed twice, it will either show latest operator removing earlier or give an appropriate message.
- Check if the clear key is working.
- Check if the brackets keys are working.
- Check if the sum or equal key is working.
- Check if the square and square root key is working.

## Functionality Test Cases

- Check the addition of two integer numbers.
- Check the addition of two negative numbers.
- Check the addition of one positive and one negative number.
- Check the subtraction of two integer numbers.
- Check the subtraction of two negative numbers.
- Check the subtraction of one negative and one positive number.
- Check the multiplication of two integer numbers.
- Check the multiplication of two negative numbers.
- Check the multiplication of one negative and one positive number.
- Check the division of two integer numbers.
- Check the division of two negative numbers.
- Check the division of one positive number and one integer number.
- Check the division of a number by zero.
- Check the division of a number by negative number. □ Check the division of zero by any number.

We can use **Notepad** as an example for writing test cases.

We can write a greater number of test case on notepad application. The basic concept is to validate menu

- 1) File
- 2) Edit
- 3) Format
- 4) View
- 5) Help

Each of these menu has submenu ,i can give you brief description how to start writing test cases.

### For Menu

Validate the click of each menu corresponding to relevant result. Say if you open file menu and hit on new, the notepad should open blank file and cursor blinking on the top righten corner

### Eg: test case

Test case No: TXXXX

Test scenario :Validating menu option -- > if you open file menu and hit on new, the notepad should open blank file and cursor blinking on the top righten corner

Expected Result : What you get as a output

Actual Result : Notepad successfully opened new file

Similarly you can validate for submenu and other menu

### **For Shortcut**

For this you need to know the shortcut for each function say if you want to open new file you can hit Ctrl+N. Do the same for other shortcuts.

Unless and until you understand each function on notepad you can't write test cases. Please navigate and find the functionality to build test cases in similar way.

### **Functional Test Cases :**

- 1) new page should be blank and cursor on beginning of first line.
- 2) the application allows typing
- 3) the application allows saving
- 4) the application allows opening a saved instance
- 5) validate menus and submenus
- 6) type of data that can be saved (numeric, alphanumeric, special chars)
- 7) editing the saved data(changing the font size, type, deleting, adding) and saving the changes.
- 8) saving a blank file.
- 9) validate file name : renaming file/keeping the default name/duplicate file name/sp chars in filename

### **Non-functional Test Cases :**

- 1) do performance testing by opening, say, 100 instances of notepad (possibly by using an automated tool) and determining the response time. 2) check max length of data

Test Plan Template : Download [Here](#)

{<https://drive.google.com/uc?export=download&id=0ByI5-ZLwpo25LThJNUZzUzdkQXM>}

# TEST PLAN FOR

## <<PROJECT NAME>>

### ChangeLog

Version	Change Date	By	Description
version number	Date of Change	Name of person who made changes	Description of the changes made

<b>1</b>	<b>INTRODUCTION .....</b>	<b>26</b>
1.1	SCOPE .....	26 1.1.1
	<i>In Scope</i> .....	26
	1.1.2 <i>Out of Scope</i> .....	26
1.2	QUALITY OBJECTIVE.....	26
1.3	ROLES AND RESPONSIBILITIES .....	26
<b>2</b>	<b>TEST METHODOLOGY .....</b>	<b>26</b>
2.1	OVERVIEW .....	26
2.2	TEST LEVELS .....	27 2.3 BUG TRIAGE .....
	.....	27
2.4	SUSPENSION CRITERIA AND RESUMPTION REQUIREMENTS.....	27
2.5	TEST COMPLETENESS .....	27
<b>3</b>	<b>TEST DELIVERABLES .....</b>	<b>27</b>

<b>4</b>	<b>RESOURCE &amp; ENVIRONMENT NEEDS .....</b>	<b>28</b>
4.1	TESTING TOOLS .....	28
4.2	TEST ENVIRONMENT .....	28
<b>5</b>	<b>TERMS/ACRONYMS.....</b>	<b>28</b>

## Introduction

Brief introduction of the test strategies, process, workflow and methodologies used for the project

## Scope

### In Scope

Scope defines the features, functional or non-functional requirements of the software that **will be** tested

### Out of Scope

Out Of Scope defines the features, functional or non-functional requirements of the software that **will NOT be** tested

## Quality Objective

Here make a mention of the overall objective that you plan to achieve without your testing

Some objectives of your testing project could be

- Ensure the Application Under Test conforms to functional and non-functional requirements
- Ensure the AUT meets the quality specifications defined by the client
- Bugs/issues are identified and fixed before go live

## Roles and Responsibilities

Detail description of the Roles and responsibilities of different team members like

- QA Analyst
- Test Manager
- Configuration Manager
- Developers
- Installation Team Amongst others

## Test Methodology

## Overview

Mention the reason of adopting a particular test methodology for the project. The test methodology selected for the project could be

- WaterFall
- Iterative
- Agile
- Extreme Programming

The methodology selected depends on multiple factors. You can read about Test Methodology [here](#)

## Test Levels

**Test Levels define the Types of Testing to be executed on the Application Under Test (AUT).** The Testing Levels primarily depends on the scope of the project, time and budget constraints.

## Bug Triage

The goal of the triage is to

To define the type of resolution for each bug

- To prioritize bugs and determine a schedule for all “To Be Fixed Bugs”.

## Suspension Criteria and Resumption Requirements

Suspension criteria define the criteria to be used to suspend all or part of the testing procedure while Resumption criteria determine when testing can resume after it has been suspended

## Test Completeness

Here you define the criterias that will deem your testing complete.

For instance, a few criteria to check Test Completeness would be

- 100% test coverage
- All Manual & Automated Test cases executed
- All open bugs are fixed or will be fixed in next release

## Test Deliverables

Here mention all the Test Artifacts that will be delivered during different phases of the testing lifecycle.

Here are the sample deliverables

- 
- Test Plan
  - Test Cases

- Requirement Traceability Matrix
- Bug Reports
- Test Strategy
- Test Metrics
- Customer Sign Off

---

## Resource & Environment Needs

### Testing Tools

Make a list of Tools like

- Requirements Tracking Tool
- Bug Tracking Tool
- Automation Tools Required to test the project

### Test Environment

It mentions the minimum **hardware** requirements that will be used to test the Application.

Following **software's** are required in addition to client-specific software.

☐ Windows 8 and ☐  
above ☐  
Office 2013 and above MS  
Exchange, etc.

### Terms/Acronyms

Make a mention of any terms or acronyms used in the project

TERM/ACRONYM	DEFINITION
API	Application Program Interface
AUT	Application Under Test

Self-Activity ( using WordTool)

## Sample Test Plan Document Banking Web Application

### Example [Refer](https://www.guru99.com/test-plan-for-project.html)

*{<https://www.guru99.com/test-plan-for-project.html> }*

## 1 Introduction

The Test Plan is designed to prescribe the scope, approach, resources, and schedule of all testing activities of the project **Banking Web App**.

The plan identify the items to be tested, the features to be tested, the types of testing to be performed, the personnel responsible for testing, the resources and schedule required to complete testing, and the risks associated with the plan.

### 1.1 Scope

#### 1.1.1 In Scope

All the feature of websiteGuru99 Bank which were defined in software requirement [specs](#) are need to be tested

Module Name	Applicable Roles	Description
Balance Enquiry	Manager Customer	<b>Customer:</b> A customer can have multiple bank accounts. He can view balance of his accounts only <b>Manager:</b> A manager can view balance of all the customers who come under his supervision
Fund Transfer	Manager Customer	<b>Customer:</b> A customer can have transfer funds from his “own” account to any destination account. <b>Manager:</b> A manager can transfer funds from any source bank account to destination account
Mini Statement	Manager Customer	A Mini statement will show last 5 transactions of an account <b>Customer:</b> A customer can see mini-statement of only his “own” accounts <b>Manager:</b> A manager can see mini-statement of any account



Customized Statement	Manager Customer	<p>A customized statement allows you to filter and display transactions in an account based on date, transaction value</p> <p><b>Customer:</b> A customer can see Customized- statement of only his “own” accounts</p> <p><b>Manager:</b> A manager can see Customized -statement of any account</p>
Change Password	Manager Customer	<p><b>Customer:</b> A customer can change password of only his account.</p> <p><b>Manager:</b> A manager can change password of only his account. He cannot change passwords of his customers</p>
New Customer	Manager	<b>Manager:</b> A manager can add a new customer.
	Manager	<b>Manager:</b> A manager can edit details like address, email, telephone of a customer.
New Account	Manager	<p>Currently system provides 2 types of accounts</p> <ul style="list-style-type: none"> <li>• Saving</li> <li>• Current</li> </ul> <p>A customer can have multiple saving accounts (one in his name, other in a joint name etc).</p> <p>He can have multiple current accounts for different companies he owns.</p> <p>Or he can have a multiple current and saving accounts.</p> <p><b>Manager:</b> A manager can add a new account for an existing customer.</p>
Edit Account	Manager	<b>Manager:</b> A manager can add a edit account details for an existing account
Delete Account	Manager	<b>Manager:</b> A manager can add a delete an account for a customer.
Delete Customer	Manager	<p>A customer can be deleted only if he/she has no active current or saving accounts</p> <p><b>Manager:</b> A manager can delete a customer.</p>

Deposit	Manager	<b>Manager:</b> A manager can deposit money into any account. Usually done when cash is deposited at a bank branch.
Withdrawal	Manager	<b>Manager:</b> A manager can withdraw money from any account. Usually done when cash is withdrawn at a bank branch.

### 1.1.2 Out of Scope

These feature are not be tested because they are not included in the software requirement specs

- User Interfaces
- Hardware Interfaces
- Software Interfaces
- Database logical
- Communications Interfaces
- Website Security and Performance

### 1.2 Quality Objective

The test objectives are to **verify** the Functionality of website Guru99 Bank, the project should focus on testing the **banking operation** such as Account Management, Withdrawal, and Balance...etc. to **guarantee** all these operation can work **normally** in real business environment.

### 1.3 Roles and Responsibilities

The project should use **outsourcing** members as the tester to save the project cost.

No.	Member	Tasks
1.	Test Manager	Manage the whole project Define project directions Acquire appropriate resources
2.	Test	Identifying and describing appropriate test techniques/tools/automation architecture Verify and assess the Test Approach Execute the tests, Log results, Report the defects. Outsourced members
3.	Developer in Test	Implement the test cases, test program, test suite etc.

<b>4.</b>	Test Administrator	Builds up and ensures test environment and assets are managed and maintained Support Tester to use the test environment for test execution
<b>5.</b>	SQA members	Take in charge of quality assurance Check to confirm whether the testing process is meeting specified requirements

## 2 Test Methodology

### 2.1 Overview 2.2 Test Levels

### 2.3 Bug Triage 2.4 Suspension Criteria and Resumption Requirements

If the team members report that there are **40%** of test cases **failed**, suspend testing until the development team fixes all the failed cases.

### 2.5 Test Completeness

- Specifies the criteria that denote a **successful** completion of a test phase
- **Run** rate is mandatory to be **100%** unless a clear reason is given. •
- Pass** rate is **80%**, achieving the pass rate is **mandatory**

### 2.6 Project task and estimation and schedule

Task	Members	Estimate effort
<b>Create the test specification</b>	Test Designer	170 man-hour
<b>Perform Test Execution</b>	Tester, Test Administrator	80 man-hour
<b>Test Report</b>	Tester	10 man-hour
<b>Test Delivery</b>		20 man-hour
<b>Total</b>		280 man-hour

**Schedule to complete these tasks**

## 3 Test Deliverables • Test plans document.

- Test cases documents □ Test Design specifications.

## 4 Resource & Environment Needs

### Lab Assignments

#### Set A

Q.1 Define Test cases and Test Plan for simple application like Mobile app - calculator

Q.2 Define Test cases and Test Plan for simple application like Desktop app - Notepad

**Signature of the Instructor:**

**Date:**

**Assignment Evaluation:**

{Not done 0 , Incomplete 1 , Late Complete 2 , Needs improvement 3 , Complete 4 , Well Done 5 }

## Assignment 4 : Defect Report(Word)

### Objectives

### Design and Prepare Defect/ Bug Report

### Reading

## What is a defect?

A defect or a bug is an error in a program that causes the application to perform in an unintended manner, deviating from its requirements. Based on the urgency of fixing the defect, we can classify them on a scale of P0 to P3, with P0 defect having the most urgency to fix.

Also, the defects can be classified based on their criticality or the impact on the functionality. Depending on the organization, we can have different levels of defect severity ranging from minor to critical or showstopper.

To report a bug, we have different Defect Management Tools like – Jira, Mantis, Bugzilla etc. Next, we will see the different components of a Defect Report.

### Defect Reporting Template

**DefectId** – A unique identifier of the defect.

**Summary** – A one-line summary of the defect, more like a defect title.

**Description** – A detailed description of the defect.

**Build Version** – Version of the build or release in which defect is found.

**Steps to reproduce** – The steps to reproduce the defect.

**Expected Behavior** – The expected behavior from which the application is deviating because of the defect.

**Actual Behavior** – The current erroneous state of the application w.r.t. the defect.

**Priority** – Based on the urgency of the defect, this field can be set on a scale of P0 to P3. **Severity** – Based on the criticality of the defect, this field can be set to minor, medium, major or show stopper.

**Reported By** – Name of the QA, reporting the defect.

**Reported On** – The date on which the defect was raised.

<p><b>Assigned To</b> – The person to whom the defect is assigned in its current state. It can be the developer fixing the defect, the QA for verification of the fixed defect or the manager approving the defect.</p> <p><b>Current Status</b> – The current status of the defect (one of the states of the defect life cycle).</p> <p><b>Environment</b> – The environment in which the defect is found – release, staging, production, etc.</p>
---

### Here is the Example scenario that caused a Bug:

Let's assume in your application under test you want to create a new user with user information, for that you need to login into the application and navigate to USERS menu > New User, then enter all the details in the 'User form' like, First Name, Last Name, Age, Address, Phone etc.

Once you enter all this information, you need to click on the 'SAVE' button in order to save the user. Now you can see a success message saying, "New User has been created successfully".

But when you entered your application by logging in and navigated to USERS menu > New user, entered all the required information to create the new user and clicked on SAVE button. BANG! The application crashed and you got one error page on the screen. (Capture this error message window and save as a Microsoft paint file)

Now, this is the **Bug scenario** and you would like to report this as a BUG in your BugTracking Tool.

Ready Reference

### SAMPLE BUG REPORT

**Bug Name:** Application crash on clicking the SAVE button while creating a new user.

**Bug ID:** (It will be automatically created by the BUG Tracking tool once you save this bug)

**Area Path:** USERS menu > New Users

**Build Number:** Version Number 5.0.1

**Severity:** HIGH (High/Medium/Low) or 1

**Priority:** HIGH (High/Medium/Low) or 1

**Assigned to:** Developer-X **Reported**

**By:** Your Name

**Reported On:** Date **Reason:** Defect

**Status:** New/Open/Active (Depends on the Tool you are using)

**Environment:** Windows 2003/SQL Server 2005 **Description:**

Application crash on clicking the SAVE button while creating a new the user, hence unable to create a new user in the application.

**Steps To Reproduce:**

- 1) Login into the Application
- 2) Navigate to the Users Menu > New User 3) Filled all the user information fields
- 4) Clicked on the 'Save' button
- 5) Seen an error page "ORA1090 Exception: Insert values Error..."
- 6) See the attached logs for more information (Attach more logs related to bug..IF any) 7)

And also see the attached screenshot of the error page.

**Expected Result:** On clicking SAVE button, should be prompted to a success message "New User has been created successfully".

(Attach 'application crash' screenshot. IF any)

**Save** the Defect/Bug in the BUG TRACKING TOOL. You will get a Bug id, which you can use for further bug reference.

Default 'New Bug' mail will go to the respective developer and the default module owner (Team leader or manager) for further action.

**Bug report sample 2:Web Project bug report**

**Summary:** In CTR (Click through ratio) 'Total' row calculation is wrong

**Product:** Example product

**Version:** 1.0 **Platform:** PC

**URL:** (Provide url of page where bug occurs)

**OS/Version:** Windows 2000

**Status:** NEW **Severity:** Major **Priority:** P1

**Component:** Publisher stats

**Assigned To:** developer@example.com

**Reported By:** tester@example.com

**CC:** manager@example.com **Bug Description:**

**Reproduce steps:**

- 1) Go to page: (Provide URL of page where bug occurs)
- 2) Click on 'Publisher stats' link to view publisher's revenue detail stats date wise.
- 3) On page (Provide URL of page where bug occurs) check CTR value in 'Total' row of CTR stats table.

Actual result: Calculation of 'Total' row in CTR table is wrong. Also, Individual row CTR for each publisher is not truncated to 2 digits after decimal point. It's showing CTR like 0.042556767.

Expected result: Total CTR= (Total clicks/Total searches)\*100

**[Attach bug screenshot if any]** Please

fix the bug.

#### Self-Activity

1)How to use Bugzilla Tool ? [Refer](#)

{<https://www.guru99.com/bugzilla-tutorial-for-beginners.html#3>}

2)How to use jiiraTool ? [Refer](#)

{<https://www.softwaretestinghelp.com/jira-bug-tracking/>}

#### Lab Assignments

##### Set A

Q.1 Prepare a defect report after executing Test cases forLogin form

Q.2 Prepare a defect report after executing Test cases forwithdraw Amount from ATM

**Signature of the Instructor:**

**Date:**

**Assignment Evaluation:**

{Not done 0 , Incomplete 1 , Late Complete 2 , Needs improvement 3 , Complete 4 , Well Done 5 }



## Assignment 5 : Testing Tool – Selenium

### Objectives

#### Design Test case using Selenium IDE (Browser based)

### Reading

#### What is Selenium?

**Selenium** is an open-source tool and portable framework that is used for automating the tests administered on web browsers. It is only used for testing web applications such as Shopping Carts, Email Programs like Gmail, Yahoo.

Testing done using **Selenium** is often referred to as **Selenium** Testing.

**Selenium** is not just a single **tool**, as it is a collection of softwares, and each tool has different cases of testing.

Now, you might be wondering, who created Selenium?

**Selenium was introduced by Jason Huggins in 2004.** Jason Huggins an Engineer at **Thoughtworks**. He was doing his work on some web application and he suddenly required testing.

He realized that mundane **Manual Testing** (used to identify bugs, issues, and defects in software) of their application was turning to be more and more ineffective.

He created a JavaScript Program that would instinctively control the browser's action. And he named the program "**JavaScriptTestRunner**" and later he made JavaScriptTestRunner as an Open Source that was later renamed as **Selenium core**.

#### Automation Testing tools that are pre-owned to Functional operations:- □

Auto It, open Source.

- Unified **Functional Testing** (UFT)
- Selenium, it's Open Source
- ParasoftSOAtest

#### Automation Testing tools which are pre-owned for nonfunctioning automation:

- JMeter, by Apache
- Loadster,
- Webserver Stress **Tool**
- Acunetix, provided by Acunetix

(\*Pre-requisites for Selenium like OS, Versions should be specified)

Ready Reference

## Selenium IDE Installation

As you all know, Selenium IDE is only in Mozilla Firefox and Chrome plug-in. So I assume that you had pre-installed firefox. If not, you can go to the link below and simply download it from [here](https://www.mozilla.org/en-US/firefox/new/). { <https://www.mozilla.org/en-US/firefox/new/> } **Steps For Downloading Selenium**

**IDE and Installing it:-** □ Launch your Installed Firefox browser.

- Click [Here](#) and You Will be Redirected to the page of firefox where you have to download it.
- Click add to Firefox  
{ <https://addons.mozilla.org/en-US/firefox/addon/selenium-ide/> }
- A Popup Will Come then allow that and your Selenium IDE Has Installed
- Restart your browser
- Top right corner of your firefox browser and look for the Selenium IDE □ Click on That Icon and Your Selenium has been launched.

## Selenium IDE Features

Selenium IDE has various components. Each component has its features for what they are designed. Given below is the diagram by categorizing its components.

**1. Menu bar Component 2. Test Case Pane 3. Toolbar 4. Address bar 5. Test Script Editor box 6. Log, Reference Pane 7. Stop and Start Button**

Self-Activity ( using Selenium Tool)

## Selenium-IDE Login test

In this part, we are going to do a Login Test in Selenium IDE. For educational purposes, we will use the Joodle Website for Testing. You can find that particular URL:-

<https://www.jdoodle.com/> . Go to this site.

The default look of this website is something like this:

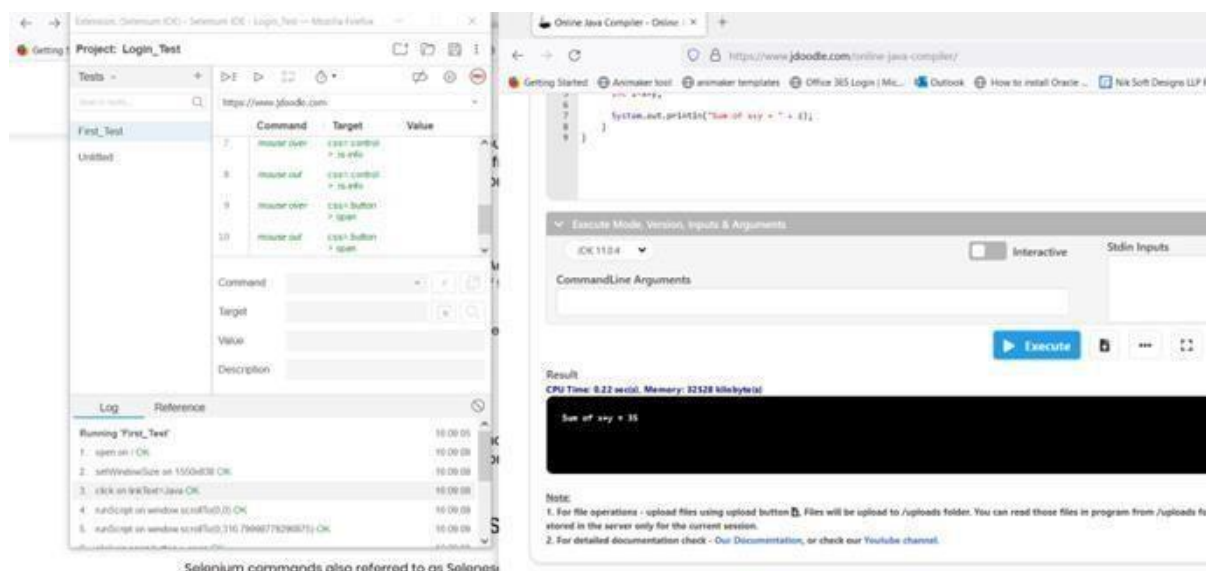
This is the default look. Now, let us start learning.

Firstly, we have to register here, and then we will sign up. Now we are going to create a test script for login/sign-up in Selenium IDE.

The steps to be followed are given below: **Recording**

1. Open your Firefox Browser.
2. At the top right corner, you will find the Selenium icon, hit on that. Now your Selenium IDE is Launched.
3. Go to the Firefox browser and then go this URL:- <https://www.jdoodle.com/>
4. Enter the project name as –**Login\_Test**
5. Enter the Test Event name as- **First\_Test**
6. Now click on the start recording button to keep track of all action done by the user.
7. Go to your firefox browser and right-click and select the Selenium IDE Option.

8. Then select the assert title. This prescribes that your title is correct.
9. Click on the Sign-up Button to sign up with your email.
10. After Signing-up, you will get an email to activate click on activate. Now you are done.
11. Now go to the login section. You will have to fill in your provided credentials.
12. This website provides an online text editor and compiler for multiple languages.
13. Click on any programming language you want.
14. Go back to your profile and then logout from there.
15. Go to your Selenium IDE Interface, click on stop recording. **Playing back**
  1. Click on The “Run Current Test Tab”. The Actions by browsers are executed and then it gives the overall summary of the test scripts.
  2. The Execution process of all the execution tests are displayed on the log pane. **Save Project:-**
    1. Click on the save project button.
    2. Save the project whatever you want to name it.
    3. In the future, if you want to work on this project you can go to the open project option.



## Selenium IDE- Commands (Selenese)

Selenium commands also referred to as Selenese is the place of commands implemented in Selenium IDE that run your tests. A sequence of Selenium commands (Selenese) together is understood as a test script. Using Selenese, one can perform activities like:

Testing the existence of UI elements supported their HTML tags.

Testing is done for specific content.

Testing is done for broken links. Assessing the window size, mouse options, alerts, Ajax functionality, pop-up windows, event handling, and much more.

Note: Test Scripts are the sequence of Selenium Commands or Selenese.

Simplicity commands are classified into different types, let's see. **Types of Selenium Commands:**

Primarily there are three types of commands that are classified in Selenium IDE:

1. Actions
2. Accessors
3. Assertions

Now, in these three types, there are several commands that we are going to see.

#### 1. **Actions**

The state of application is generally manipulated from the Actions Commands in Selenium IDE.

Some of the most frequently used Actions Commands are given below:

Commands	Description	Arguments
	The coordinates are correlative to the target element and	Locators coord
click at	it clicks on the target element and they are used to check effects.	string
open	In This Case, it opens the URL and the URL is accepted is both relative and absolute URLs	URLs
type	It Sets the value of an input field, as you typed it in. Also Used to set values for combo boxes.	locator, value
typeKeys	Keystroke events are simulated on the specified element.	locator, value
doubleClick	dual Click on an element.	locator
focus	moves the focus to the specific element	locator
highlight	background colors are changed to yellow of a particular element.	locator
close	It Closes one window, no need to close the initial window.	
store	Easy reuse, it saves the target string	text, variable name

waitForCondition	It Evaluates to “true” until and unless the specified javascript executes.	script, timeout
------------------	--	-----------------

Consequently, Selenium Provides several locators to accurately locate GUI components.

Different types of locating strategies are given below:

- Locating by ID
- Locating by Name
- Locating by Identifier
- Locating by XPath Expressions
- Locating by CSS Selectors
- Locating by DOM (Document Object Model)

### 1. Locating by ID

Locating by ID is the most frequent method of locating elements. But IDs are supposed to be unique.

Let us consider an example related to locating by identifier.

As we had done before, we will create a login test to understand it better.

#### Steps to be followed below:

- Open Firefox Browser
- Click on the particular selenium icon on the top right of most of your browser.
- It will launch your Selenium IDE
- Click on the command tab present in the Text Editor.
- Recast it to the possession of the first command as:-

1. Command: **open**
2. Target: <https://mail.rediff.com/cgi-bin/login.cgi>
3. Execute It.

After Execution, your program will look like:-

To enter the second command, we need some identification for the username text box which would serve the IDE to recognize the target location.

- Right-click on your mouse with hand over to the username and select inspect element.
- The Codes will be launched contained in the production of username text box.
- Select the Component with the matching name feature that contains the precise name for the “Username” text box.

- Recast it to the possession of the first command as:-

1. Command: click at
2. Target: id=login
3. Execute it.

□ Recast it to the possession of the first command as:-

1. Command: type
2. Target: id=login1
3. Value: username
4. Execute it.

Now we will go through with password

- Right-click on your password text box and click on inspect element.
- Again containing codes at the bottom.
- Select the component with the matching name feature that contains the precise name for the “Password” text box.
- Recast it to the possession of the first command as:-

1. Command: click at 2.

Target: id=password

3. Execute it.

□ Recast it to the possession of the first command as:-

1. Command: type
2. Target : id=password
3. Value : \*\*\*\*\*

- Now, Right Click on “sign-in” and select inspect element.
- Recast it to the possession of the first command as:-

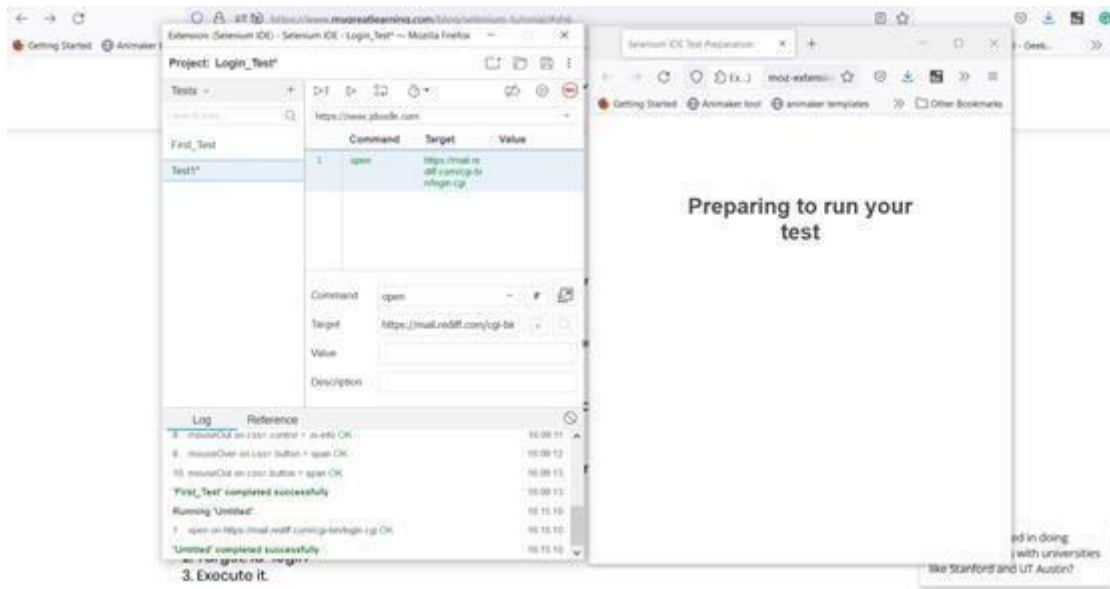
1. Command: click at 2.

Target : id=proceed 3.

Execute it.

□ Now, at last Click on the run current test tab which will execute your scripts. □

The log pane summarizes it.



## Lab Assignments

### Set A:

Q.1 Design and run Test cases using automated testing Tools for Login

Q.2 Design and run Test cases using automated testing Tools for Command.

**Signature of the Instructor:**

**Date:**

**Assignment Evaluation:**

{ Not done 0 , Incomplete 1 , Late Complete 2 , Needs improvement 3 , Complete 4 , Well Done 5 }