



**School of Information Technology & Engineering**

**MTech Software Engineering**

**SOFTWARE TESTING (SWE2005)**

**WINTER SEMESTER 2020-2021**

**REVIEW 1**

**J Component- Student Information System**

**Submitted by:**

**19MIS0117-B.Jhansi Durga Devi**

**18MIS0156-M.K. Pranitha**

**19MIS0099-B.Thulasi**

**Faculty: Prof. RAMESH P.S**

**Slot:F2**

## **ABSTRACT:**

Student Information System is one of the key factor in academics. It is used to store, administer all the aspects of student information like student details, subjects, semester details, enrollment details, grades, cgpa of the students. The system which we are going to develop allows easy modification of students and other details like payment status, adding exam result and generate auto ranking. Student Information system is a system for the education establishments to manage the student data. It provides capabilities like tracking students attendance, checking the marks, sending sms of results and notices etc.

Information security is a concern, as universities house an array of sensitive personal information, making them potentially attractive targets for security breaches, such as those experienced by retail corporations or healthcare providers.

The main motive of the project is to test the particular software and getting the awareness about that project. How the testing is being done and so on... In our project we are taking the web application which is regarding the student information system. And we are suppose to test the student information system using the TEST COMPLETE tool. And we will come to know the major things in the future. Like how to work with the tool and so many other things.

## **SCOPE:**

The main scope of the project is to test the particular application and getting to know how testing is very important. Testing has become crucial for several application/product implementations and businesses have begun to realize the importance of structured testing of applications before their release.

## **DESCRIPTION:**

Student Information system is a system for the education establishments to manage the student data. It helps to manage the information like add student information, admit student in multiple programs, student payment system , student attendance system, institute report. Are the modules present in this student information system. Here we are going to test this application using the test cases and getting all the test cases gets satisfies all the conditions. so that the system is working perfectly with no errors.

## **MODULES PRESENT IN THE STUDENT INFORMATION SYSTEM**

- Admin Login
- Add Student Information
- Admit Student In Multiple Program
- Student Payment System
- Student Attendance System
- Institute Report (Payment , Expense, Income, Profit , Attendance)

## **TOOL : TEST COMPLETE**

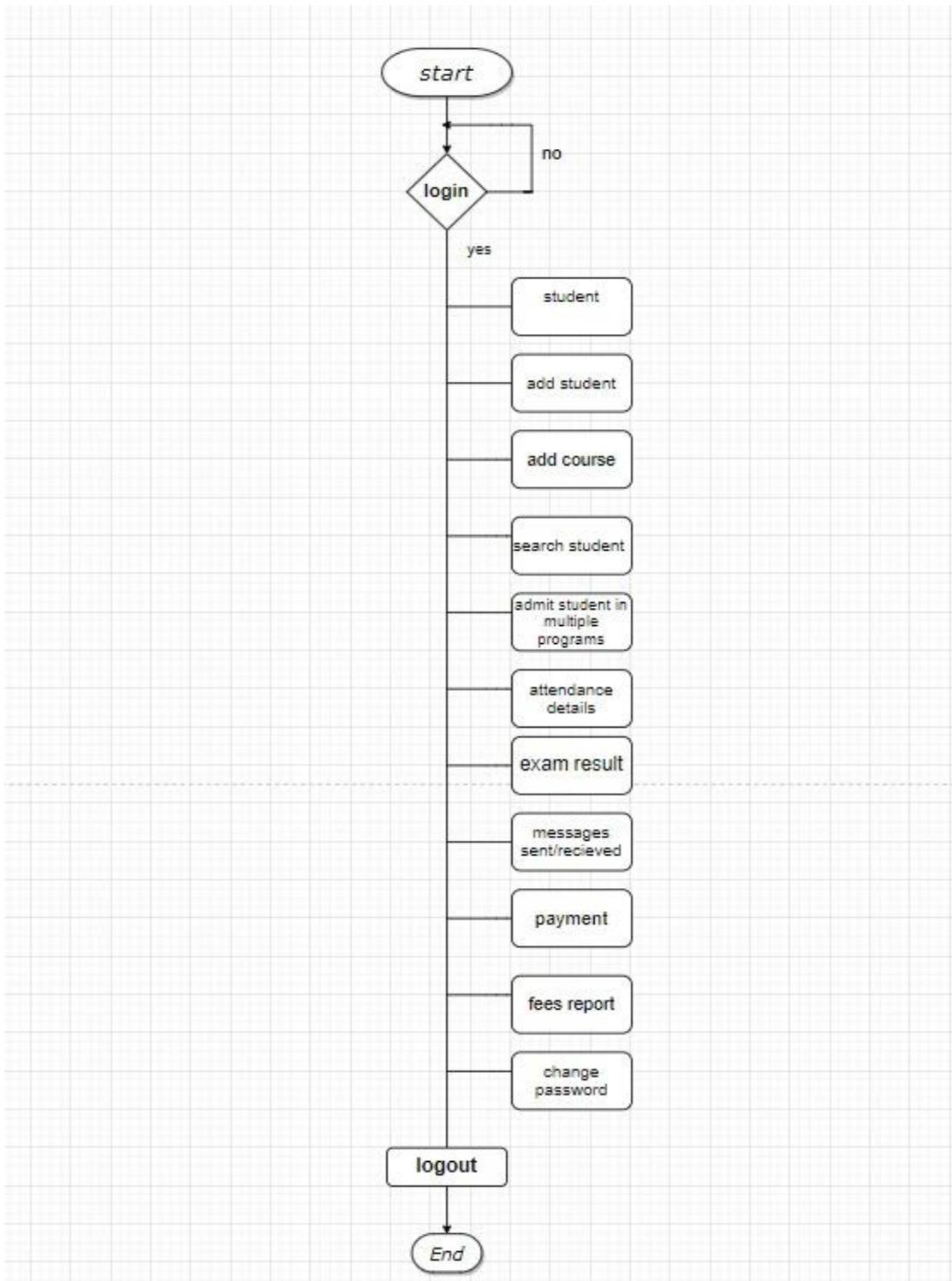
Test Complete is an automated testing tool that lets you create, manage and run tests for any Windows, Web or Rich Client software. It makes it easy for anyone to create automated tests. And automated tests run faster, increase test coverage and lower costs. Test Completes new script-free keyword testing, ease of use enhancements and centralized Start Page make learning automated testing a snap for new users. Test Complete's flexibility and extensive feature list ensure power users always have a solution to testing challenges. Test Complete is a must-have tool for QA teams that need to do more testing, keep up with rapid development schedules and still deliver software on time.

## **Testing Methodologies used in the project are:**

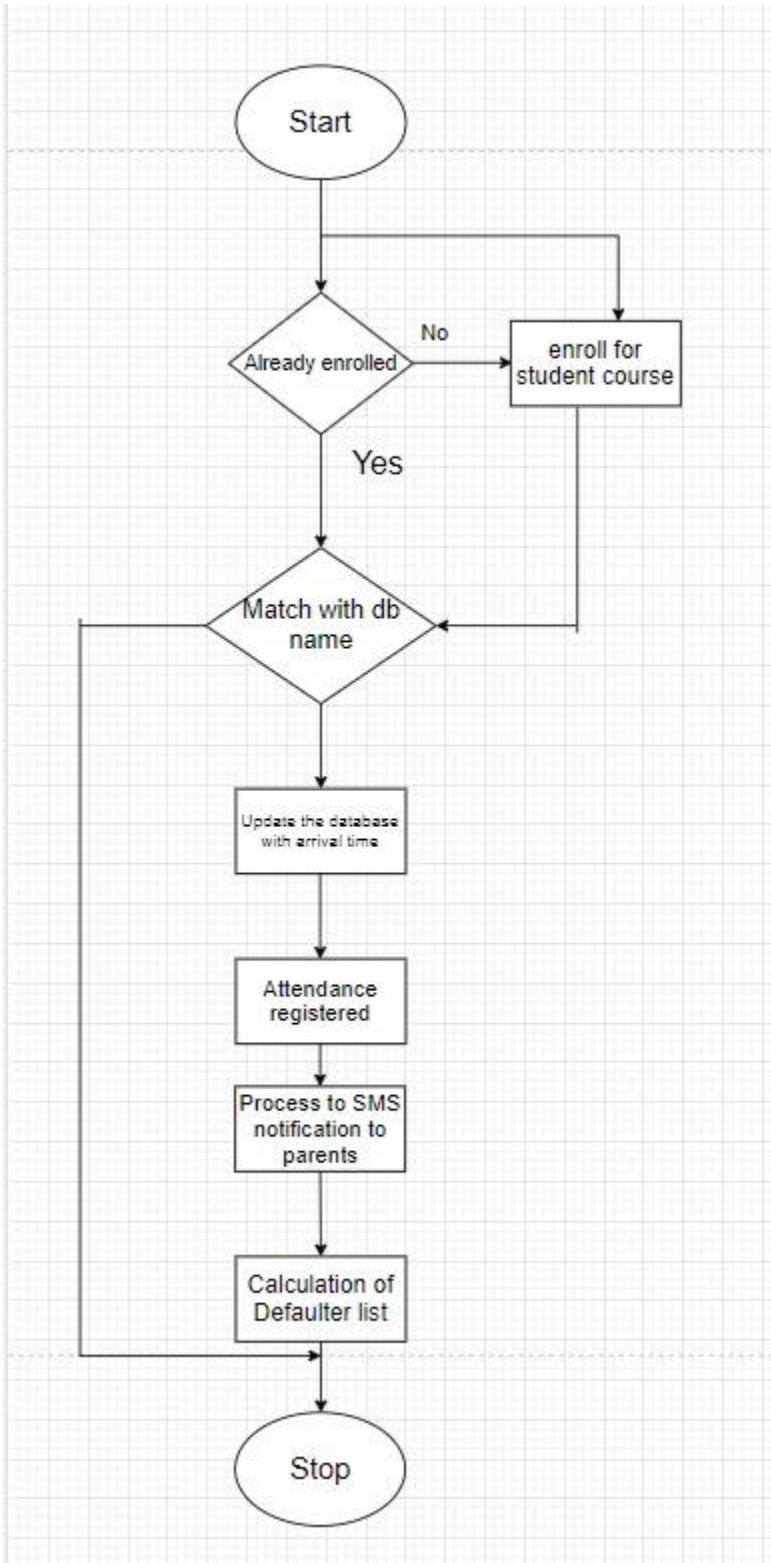
1. Automated UI testing
2. Check points testing
3. Cross browser Testing
4. Keyword testing
5. Data driven testing
6. Running Multiple tests

# Review-2

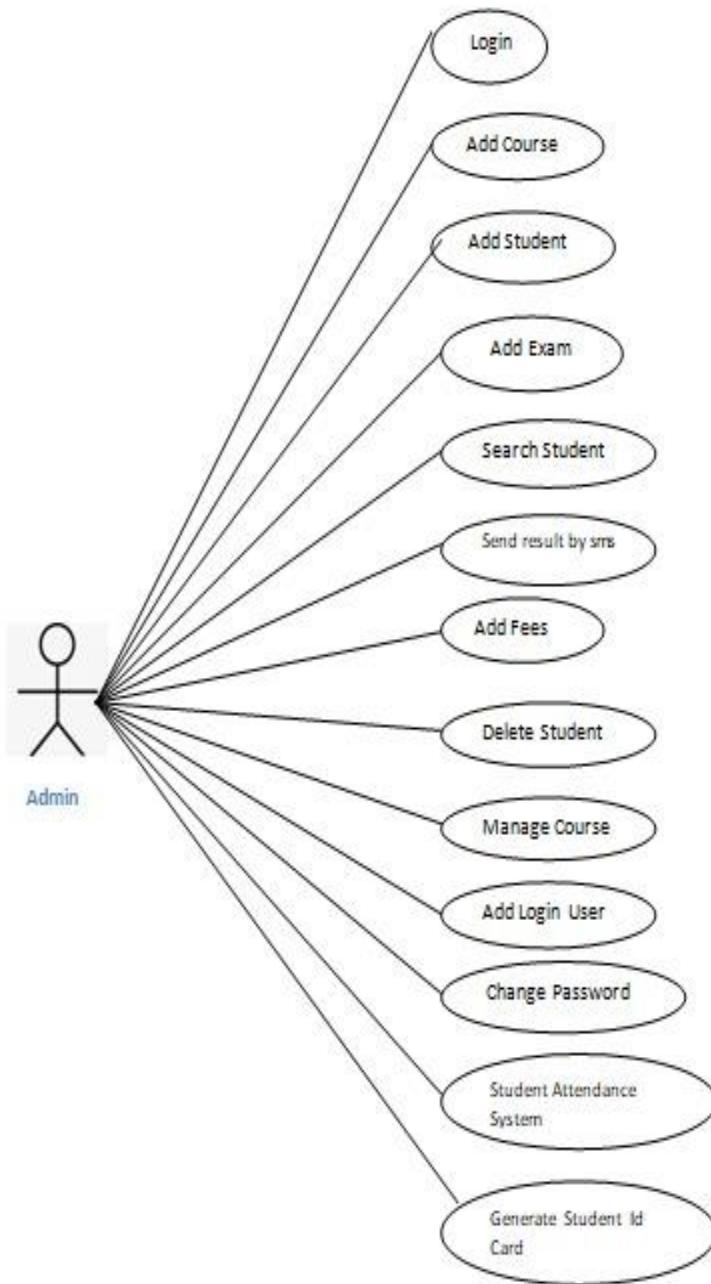
## Flowchart for Student Information System:



## Flowchart for Attendance System:



## Use case diagram:



## SAMPLE CODE:

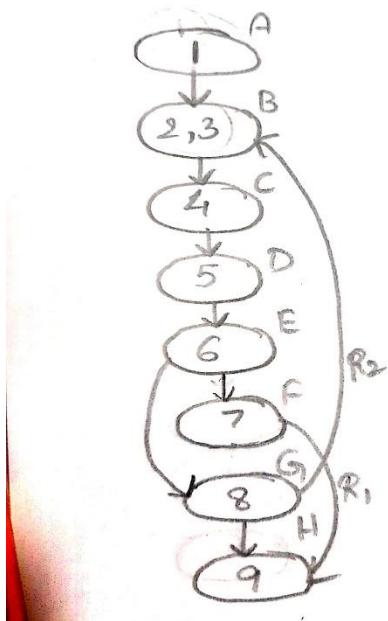
### Login:

```
function login(){
    uname=get_value('uname');
    pass=get_value('pass');
    document.getElementById("error_msg").style.display="none";
    login_area("none");
    $.ajax({
        type: 'POST',
        url: 'login_action.php',
        data: {
            login: uname,
            uname: uname,
            pass: pass
        },
        success: function(response) {
            response=JSON.parse(response);
            setTimeout(function () {
                fun_wait(response);
            }, 1000);
        }
    });
}

function fun_wait(data){
if(data.error==0)location.reload();
else{
    document.getElementById("error_msg").style.display="block";
    set_html("error_msg_text",data.error_msg);
    login_area("block");
}
}
```

1. Start the function login.
2. Get the value of username from the user.
3. Get the value of password from the user.
4. Assign it to a variable.
5. Check the values with the data base.
6. If the entered username and password is correct.
7. Then it will redirect to the dashboard page.
8. Else, Error message will display, user should check the details again and enter the correct details.
9. Exit.

### Flow graph for login Page.



### Cyclomatic Complexity:

Number of nodes in the flowgraph ( $n$ )=8

Number of edges in the flowgraph ( $e$ )=8

Number of graphs ( $p$ )=1

$$V(G) = e - n + 2(p) = 8 - 8 + 2(1) = 2$$

So, the cyclomatic complexity for the flowgraph =2.

### Independent Paths:

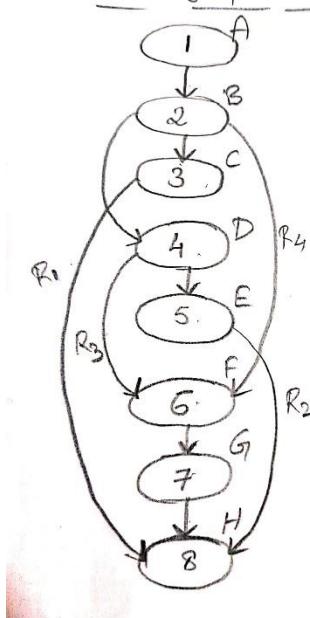
- I. A-B-C-D-E-F-H
- II. A-B-C-D-E-G-B-C-D-E-F-H

## Add student in multiple program:

```
function get_program_form(type, id = 0) {  
    if (type == "insert") {  
        header = "Add program";  
        data_key = "add_program_form";  
        val = 1;  
    } else if (type == "update") {  
        header = "Update program";  
        data_key = "update_program_form";  
        val = id;  
    } else {  
        header = "Delete program";  
        data_key = "delete_program_form";  
        val = id;  
    }  
}
```

1. Start the function program.
2. If insert into new program
3. Add program details
4. Else, if update
5. Update Program details
6. Else
7. Delete Program
8. Exit

Flow graph to add program



### Cyclomatic Complexity:

Number of nodes in the flowgraph (n)=8

Number of edges in the flowgraph (e)=10

Number of graphs (p)=1

$$V(G) = e - n + 2(p) = 10 - 8 + 2(1) = 4$$

So, the cyclomatic complexity for the flowgraph =4.

### Independent Paths:

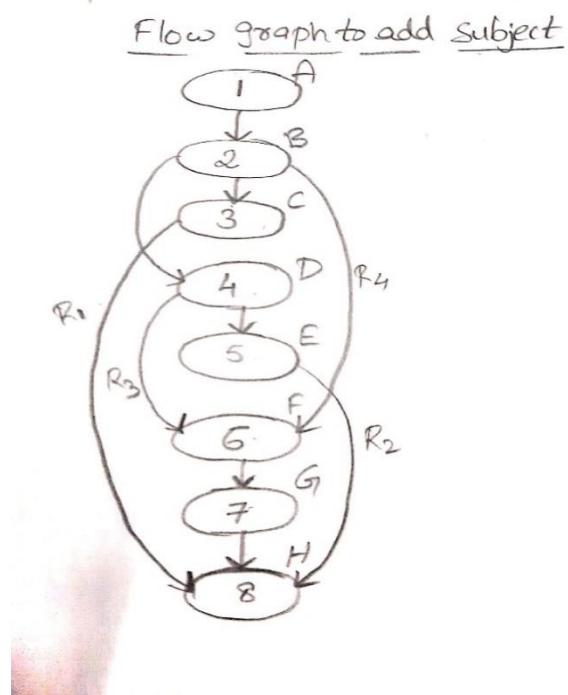
- I. A-B-C-H
- II. A-B-D-E-H
- III. A-B-D-F-G-H
- IV. A-B-F-G-H

### Add subject details:

```
function get_subject_form(type,id=0){  
    if(type=="insert"){  
        header="Add Subject";  
        data_key="get_subject_form";  
        val=1;  
    }  
    else if(type=="update"){  
        header="Update Subject";  
        data_key="update_subject_form";  
        val=id;  
    }  
    else{  
        header="Delete Subject";  
        data_key="delete_subject_form";  
        val=id;  
    }  
}
```

1. Start the function get subject.
2. If insert
3. Add new subject details
4. Else, if update

5. Update subject details
6. Else
7. Delete subject details
8. Exit



### Cyclomatic Complexity:

Number of nodes in the flowgraph ( $n$ )=8

Number of edges in the flowgraph ( $e$ )=10

Number of graphs ( $p$ )=1

$$V(G) = e - n + 2(p) = 10 - 8 + 2(1) = 4$$

So, the cyclomatic complexity for the flowgraph =4.

### Independent Paths:

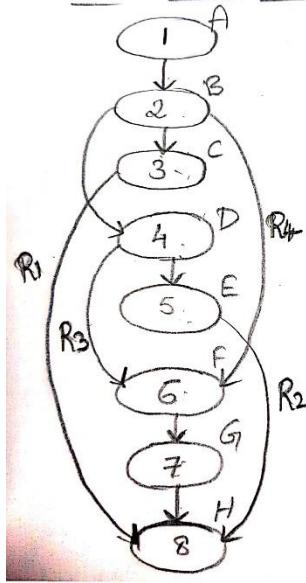
- I. A-B-C-H
- II. A-B-D-E-H
- III. A-B-D-F-G-H
- IV. A-B-F-G-H

## **Student Payment system:**

```
function select_payment_type(){
    type=get_value("payment_type");
    var div,data;
    if(type==1){
        div="payment_panel_body";
        payment_year=0;
        payment_month=0;
        payment_type=1;
        show_payment_panel();
        set_html("payment_month_area","");
    }
    else if(type==2){
        div="payment_month_area";
        var data={
            "select_monthly_program": program_id
        };
        set_html("payment_panel_body","");
    }
    else{
        set_html("payment_month_area","");
        set_html("payment_panel_body","");
        return;
    }
    loader(div);
    get_ajax(get_action_data(div), data);
}
```

1. Start the function payment type.
2. If payment type=1
3. Payment panel body
4. Else ,if payment type=2
5. Payment month area
6. Else
7. Return to payment type
8. Exit

Flow graph for Payment type



Cyclomatic Complexity:

Number of nodes in the flowgraph (n)=8

Number of edges in the flowgraph (e)=10

Number of graphs (p)=1

$$V(G) = e - n + 2(p) = 10 - 8 + 2(1) = 4$$

So, the cyclomatic complexity for the flowgraph =4.

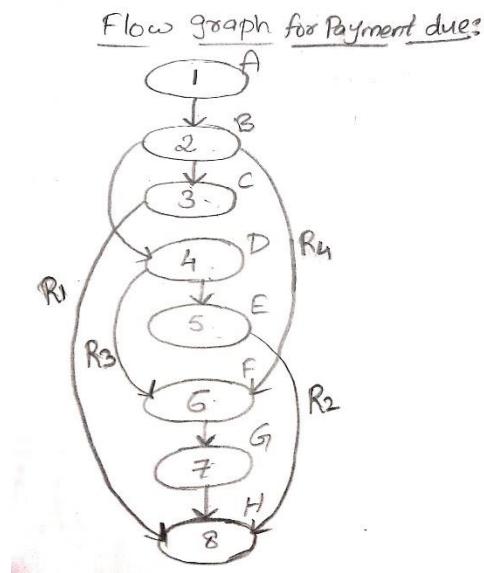
Independent Paths:

- I. A-B-C-H
- II. A-B-D-E-H
- III. A-B-D-F-G-H
- IV. A-B-F-G-H

## Student Payment Due:

```
function paid_fun(){
    total=document.getElementById('total').value;
    paid=document.getElementById("paid").value;
    due=total-paid;
    if(paid<0){
        alert("Paid Ammount Cannot Be negative");
        document.getElementById("due_ammount").value="";
    }
    else if(due<0){
        alert("Paid Ammount must Be smaller or equal then total ammount");
        document.getElementById("due_ammount").value="";
    }
    else{
        document.getElementById("due_ammount").value=due;
    }
}
```

1. Start the function paid due.
2. If paid<0
3. Alert message
4. Else ,if due<0
5. Alert message
6. Else
7. Accepts due amount
8. Exit



### Cyclomatic Complexity:

Number of nodes in the flowgraph (n)=8

Number of edges in the flowgraph (e)=10

Number of graphs (p)=1

$$V(G) = e - n + 2(p) = 10 - 8 + 2(1) = 4$$

So, the cyclomatic complexity for the flowgraph =4.

### Independent Paths:

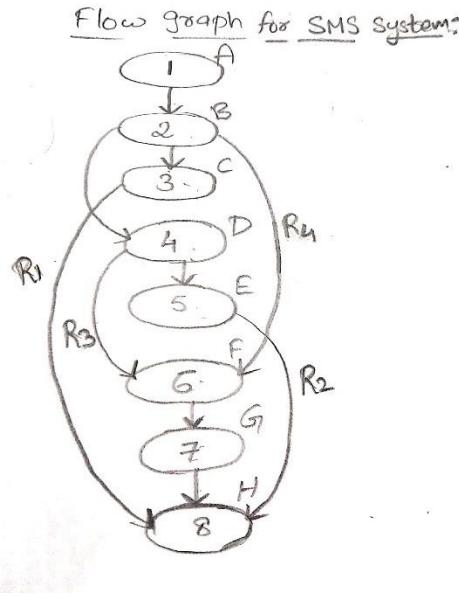
- I. A-B-C-H
- II. A-B-D-E-H
- III. A-B-D-F-G-H
- IV. A-B-F-G-H

### Send SMS:

```
function get_sms_form(type,id=0){  
    if(type=="insert"){  
        header="Add SMS";  
        data_key="get_sms_form";  
        val=1;  
    }  
    else if(type=="update"){  
        header="Update SMS";  
        data_key="update_sms_form";  
        val=id;  
    }  
    else{  
        header="Delete SMS";  
        data_key="delete_sms_form";  
        val=id;  
    }  
}
```

1. Start the function get sms.
2. If insert

3. Add new sms details
4. Else, if update
5. Update sms details
6. Else
7. Delete sms details
8. Exit



### Cyclomatic Complexity:

Number of nodes in the flowgraph ( $n$ )=8

Number of edges in the flowgraph ( $e$ )=10

Number of graphs ( $p$ )=1

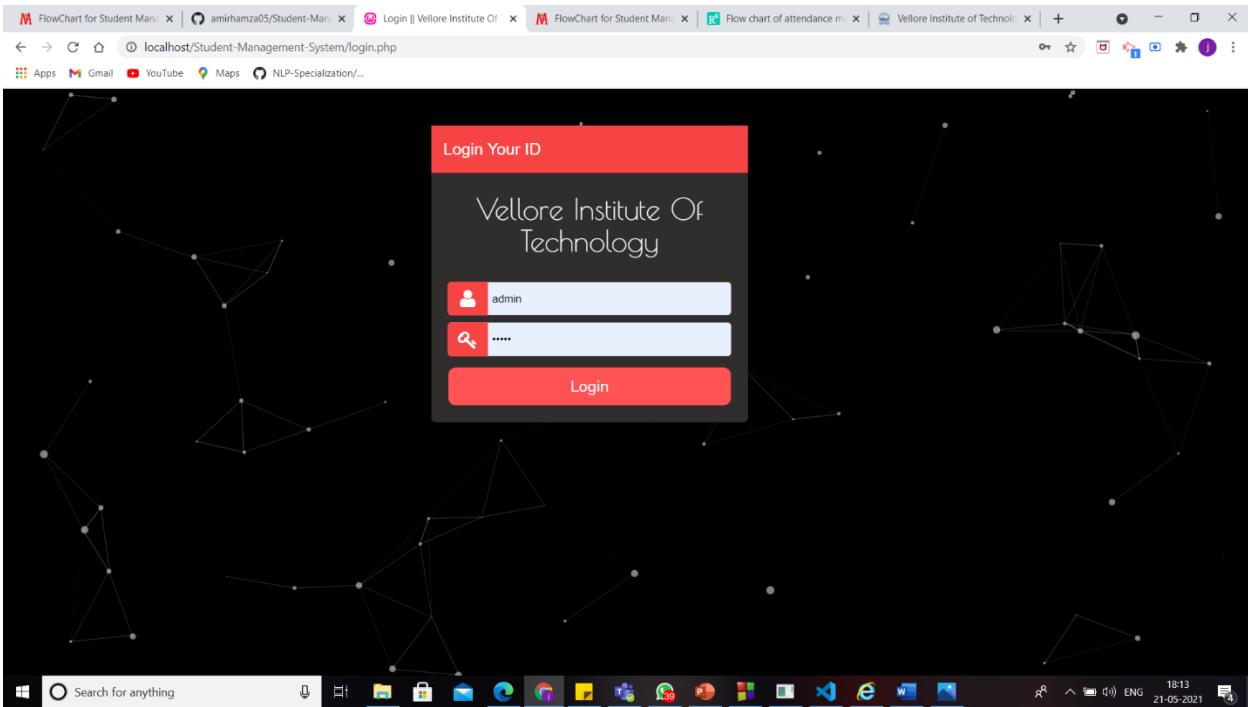
$$V(G) = e - n + 2(p) = 10 - 8 + 2(1) = 4$$

So, the cyclomatic complexity for the flowgraph =4.

### Independent Paths:

- I. A-B-C-H
- II. A-B-D-E-H
- III. A-B-D-F-G-H
- IV. A-B-F-G-H

## Admin Login page:



## Dashboard:

The screenshot shows a Windows desktop environment. A browser window is open at the URL `localhost/Student-Management-System/index.php`. The dashboard has a dark teal header with the "Vellore Institute Of Technology" logo and the text "Good Evening". On the left, there is a sidebar with navigation links: Dashboard, Students, Batch, Program, Subject, Exam, Notice, User, Account, Report, and SMS. The main content area displays the "Vellore Institute Of Technology" logo, address, and phone number. It also features four summary cards: "TOTAL STUDENTS" (2), "TOTAL TEACHER" (-), "SMS BALANCE" (100), and "TOTAL NOTICE" (1). Below these are two graphs: "Last 7 Days Site Activity Graph" and "Last 7 Days SMS Send Performance Graph". The browser's address bar shows the full URL. The taskbar at the bottom includes icons for various applications like File Explorer, Edge, and Control Panel.

## Student Profile:

The screenshot shows the 'Student Profile' section of the system. On the left, there is a 'Profile' card displaying a placeholder student photo and the following details:

Full Name:	B.Jhansi Durga Devi
Nick:	Jhansi
ID:	10003
ID Code:	

On the right, there is an 'Info Panel' containing the following information:

Full Name:	B.Jhansi Durga Devi
Nick Name:	Jhansi
Student Photo:	
Student ID:	10003
Student ID Barcode:	
Father Name:	B.Janardan Reddy
Mother Name:	Maha Lakshmi
Email:	jhansi@gmail.com

At the top of the page, there is a navigation bar with tabs: 'Student Quick Access', 'SMS Statistics', 'Live Chat', and 'SQL Editor'. Below the navigation bar, there are five buttons: 'Info', 'Program', 'Payment', 'Result', and 'Message'. A 'Update Student Information' button is located in the top right corner of the info panel.

## ID-card:

The screenshot shows the 'Student ID Card' section of the system. At the top, there are dropdown menus for 'MIS' and 'Afternoon ( 2:00PM - 7:00PM )', and a 'View Program' button. Below these are four navigation buttons: 'Program Overview', 'Student List', 'Student ID Card' (which is highlighted in blue), and 'Student Payment'.

Two student ID cards are displayed side-by-side:

**Left ID Card (Student Name: Hari):**

Student Name:	Hari
Student ID:	10001
Father Name:	sai
Mother Name:	yamini
Mobile:	.

**Right ID Card (Student Name: Mansi):**

Student Name:	Mansi
Student ID:	10002
Father Name:	Hari
Mother Name:	priya
Mobile:	.

Each ID card includes a placeholder photo, the Vellore Institute of Technology logo, and a barcode at the bottom. There are 'Print' and '+' buttons in the top right corner of each card area.

## Students details:

The screenshot shows a web browser window for the Vellore Institute Of Technology Student Management System. The URL is [localhost/Student-Management-System/student\\_list.php](http://localhost/Student-Management-System/student_list.php). The page displays student information for the MIS program, specifically for the Afternoon batch. The interface includes dropdown menus for 'MIS' and 'Afternoon ( 2:00PM - 7:00PM )'. Below these are tabs for 'Program Overview', 'Student List', 'Student ID Card', and 'Student Payment'. A table provides detailed information about the program:

Program Name	MIS
Batch Name	Afternoon
Subject	Software Testing
Total Student	2

On the right, another table shows the schedule for the Afternoon batch: Monday, Tuesday, Wednesday, Thursday, Friday, (2:00PM - 7:00PM). The Windows taskbar at the bottom shows various application icons.

## Program details:

The screenshot shows a web browser window for the Vellore Institute Of Technology Student Management System. The URL is [localhost/Student-Management-System/program\\_list.php](http://localhost/Student-Management-System/program_list.php). The page displays a 'Program List' table with the following data:

Program Name	Program Start	Program End	Subject	Batch	Type	Fee	Monthly Fee	Action	Set Payment
MIS	2019-06-12	2024-06-14	Software Testing	Afternoon	Package	95000	0		

Below the table, it says 'Showing 1 to 1 of 1 entries'. The Windows taskbar at the bottom shows various application icons.

## Fee Payment Receipt:

Print Payment Receipt

Vellore Institute Of Technology  
Vellore, Tamil Nadu.  
8765436789  
vit@gmail.com

Student ID: 10001  
Name: Hari  
Program Name: MIS  
Type: Admission Fee  
Year: -  
Month: -

Total Fee: 95000 tk  
Previous Total Pay: 0 tk  
Pay: 50000 tk  
Total Due: 45000 tk  
Status: Due

Signature

Print Money Receipt

## Attendance list:

MIS      Afternoon (2:00PM - 7:00PM)      21-05-2021      Attendance Panel

Take Attendance    Attendance Report    Send Attendance SMS

Present And Absent Report    Present Report    Absent Report

Print Report

Vellore Institute Of Technology  
Vellore, Tamil Nadu.  
Phone: 8765436789 | Email: vit@gmail.com

Daily Attendance Report  
Date: 21 May 2021

Program: MIS

Student ID	Student Name	Attendance
10001	Hari	Present
10002	Mansi	Absent

## Attendance Report:

The screenshot shows the Vellore Institute Of Technology Student Management System interface. The top navigation bar includes links for FlowChart for Student Manager, amirhamza05/Student-Manager, Vellore Institute Of Technology, and Flow chart of attendance manager. Below the header, there are tabs for MIS, Afternoon (2:00PM - 7:00PM), 2021, May, and View Report. The main content area displays the Vellore Institute Of Technology logo and contact information (Vellore, Tamil Nadu; Phone: 8765436789; Email: vit@gmail.com). A title for the Monthly Attendance Report for May 2021 is shown. A table lists student attendance for the MIS program. The table has columns for ID, Name, and dates from 1 to 31. It shows two students: Hari (ID 10001) and Mansi (ID 10002). Hari has a green cell for May 21st labeled 'P' and a red cell for May 22nd labeled 'A'. Mansi has a red cell for May 22nd labeled 'A'. The total present (T.P.) is 0 and total absent (T.A.) is 1. A 'Print Report' button is visible in the top right corner.

## Notice details:

The screenshot shows the Vellore Institute Of Technology Student Management System interface. The top navigation bar includes links for FlowChart for Student Manager, amirhamza05/Student-Manager, Notice || Vellore Institute Of Tech, FlowChart for Student Manager, and Flow chart of attendance manager. Below the header, there are tabs for MIS, Afternoon (2:00PM - 7:00PM), 2021, May, and View Report. The main content area shows a large digital clock displaying '21 May 2021'. To its right, a notice titled 'Cat-2 Exam' is displayed, stating 'Cat2 exam starts from may18th'. Below the notice are buttons for '+ Add Notice', 'Send SMS', 'Edit', and 'Delete'. A 'Print Report' button is also visible in the top right corner.

## SMS dashboard:

The screenshot displays the SMS Dashboard of the Vellore Institute Of Technology. The top navigation bar includes links for Student Quick Access, SMS Statistics, Live Chat, and SQL Editor. The dashboard features four main performance metrics in large circular icons:

- TOTAL SMS**: 150 (More Info)
- TOTAL SEND**: 0 (More Info)
- TOTAL EXPIRE**: 50 (More Info)
- TOTAL BALANCE**: 100 (More Info)

Below these are two tables:

- Buy SMS List**: Shows 2 entries. Row 1: #2, Total Buy SMS 50, Total Use 0, Buy Date 2021-05-01, Expire Date 2021-05-13, Add By admin, Status Expired. Row 2: #1, Total Buy SMS 100, Total Use 0, Buy Date 2021-05-14, Expire Date 2021-06-26, Add By admin, Status Active. Actions column contains edit and delete icons.
- Send SMS List**: Shows 0 to 0 of 0 entries. Includes a search bar, sorting by Phone Number, and a table header with columns: #, Phone Number, Message, Time, Sender. A large green '+' button is located at the bottom right of this section.

The taskbar at the bottom shows various application icons and the system clock indicating 23:02 on 21-05-2021.

#	Total Buy SMS	Total Use	Buy Date	Expire Date	Add By	Status	Action
2	50	0	2021-05-01	2021-05-13	admin	Expired	
1	100	0	2021-05-14	2021-06-26	admin	Active	

# Review-3

## **Test Complete Tool:**

TestComplete is a functional automated testing platform developed by SmartBear Software. TestComplete gives testers the ability to create automated tests for Microsoft Windows, Web, Android (operating system), and iOS applications. Tests can be recorded, scripted or manually created with keyword driven operations and used for automated playback and error logging.

## **Uses:**

TestComplete is used to create and automate many different software test types. Record and playback test creation records a tester performing a manual test and allows it to be played back and maintained over and over again as an automated test. Recorded tests can be modified later by testers to create new tests or enhance existing tests with more use cases.

## **Main Features:**

- **Keyword Testing:** TestComplete has a built-in keyword-driven test editor that consists of keyword operations that correspond to automated testing actions.
- **Scripted Testing:** TestComplete has a built-in code editor that helps testers write scripts manually. It also includes a set of special plug-ins that help.
- **Test Record and Playback:** TestComplete records the key actions necessary to replay the test and discards all unneeded actions.
- **Distributed Testing:** TestComplete can run several automated tests across separate workstations or virtual machines.
- **Data-driven testing:** Data-driven testing with TestComplete means using a single test to verify many different test cases by driving the test with input and expected values from an external data source instead of using the same hard-coded values each time the test runs.
- **Test Visualizer :** TestComplete automatically captures screenshots during test recording and playback. This enables quick comparisons between expected and actual screens during test.
- **Custom Checkpoints:** TestComplete uses comparison commands called checkpoints.

To perform various verifications, TestComplete allows creating checkpoints to compare various types of data: images, object properties, XML documents, PDF files, arbitrary files, web services, and so on.

## **PROCEDURE OF RECORD & REPLAY TESTING TECHNIQUE IN TEST COMPLETE:**

We have to follow these basic steps to get familiar with the Record & Playback features in test complete Test:

1. Click on the New Project button on the main toolbar. Provide a name for your project test case and click OK. An empty test case will be created.
2. Click on create new project from the main toolbar and fill in the relevant details.
3. Execute all your test steps on the browser. All of your actions performed on the browser will be recorded by test complete.
4. You can Stop recording if needed. When you are done with recording, click OK to save recorded actions into test complete.
5. You can verify the tests.
6. Recorded objects and actions will be shown.
7. You can run the project normally to check the actions of the test cases.
8. Click on the Run Project suite button on the main Toolbar to execute the script. You can specify the target browser to run by selecting it from the drop-down list.
9. So that we can get the graph of performance tests
10. So that we can get the list of execution steps where the test case passed and where it was failed.

## **Testing Methodologies used in the project are:**

7. Automated UI testing
8. Check points testing
9. Cross browser Testing
10. Keyword testing
11. Data driven testing
12. Running Multiple tests

- **Automated UI testing:**

An automated UI testing tool like TestComplete, which has a powerful object and visual recognition engine, will speed up and scale your testing efforts by enabling you to detect and test any UI element. Easily build and automate functional UI tests for Desktop, Mobile, and Web applications.

1. Structure, Organize, & Set Up Source Control
2. Familiarize Yourself with the Application
3. Determine Your Testing Environments & Gather Data
4. Set Up a Smoke Test Project
5. Create Utilities for On Screen Actions
6. Build and Manage Verifications
7. Set Up Your Logging & Reporting Mechanism

### **1. Automated UI testing for Login Module:**

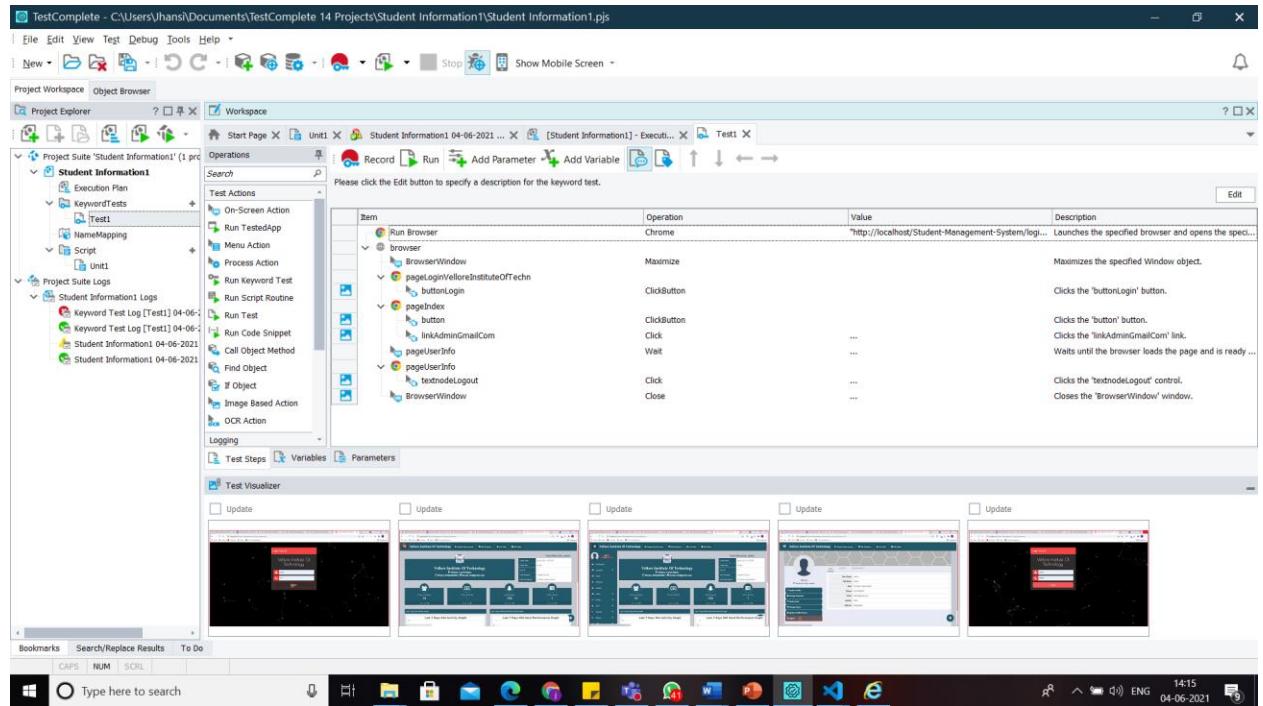
#### **Test Cases for login Moule:**

Test Case id	input	Expected Output	Actual Output	Status
1	Username: admin Password: admin	Navigate to dashboard page	Navigated to dashboard page	Pass
2	Username: admin Password: jhansi	Navigate to dashboard page	Enter correct password	Fail
3	Username: Jhansi Password: admin	Navigate to dashboard page	Check the user name	Fail
4	Username: @#### Password: admin	Check user name format	Invalid username or password	Fail

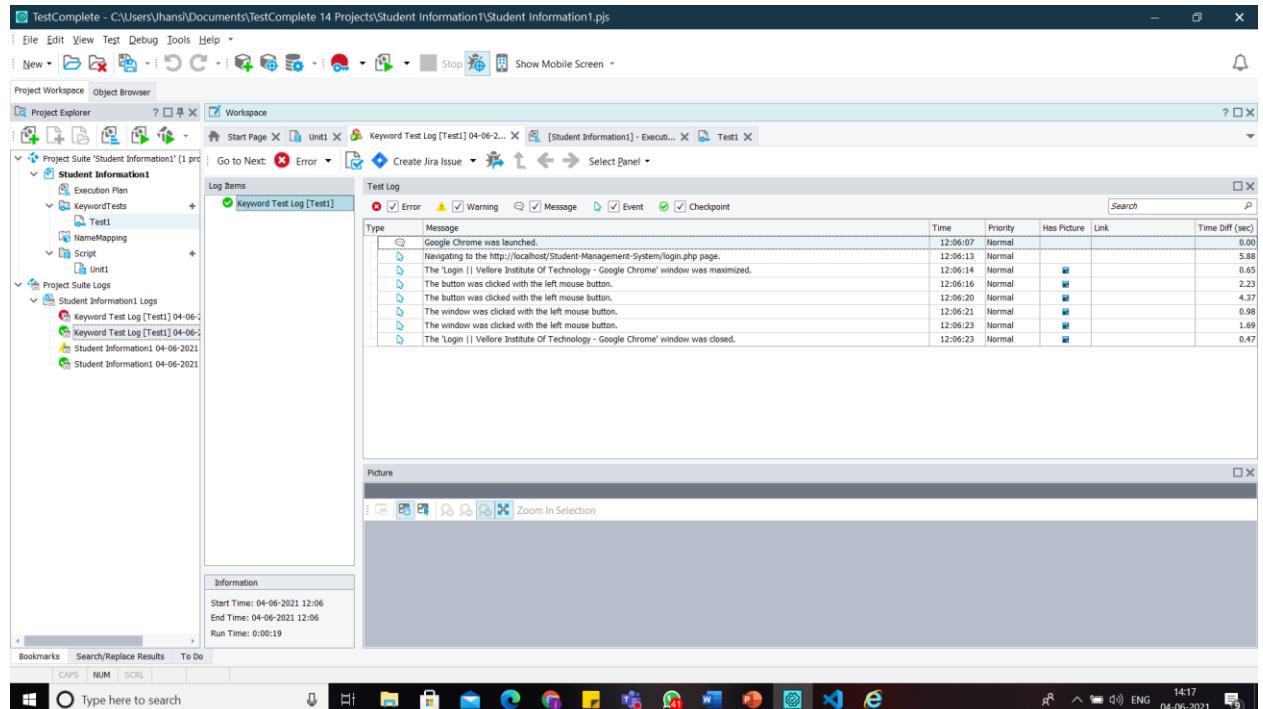
#### **Test complete tool:**

In test complete tool, it records all the actions of the login portal and displays the tests performed while validating.

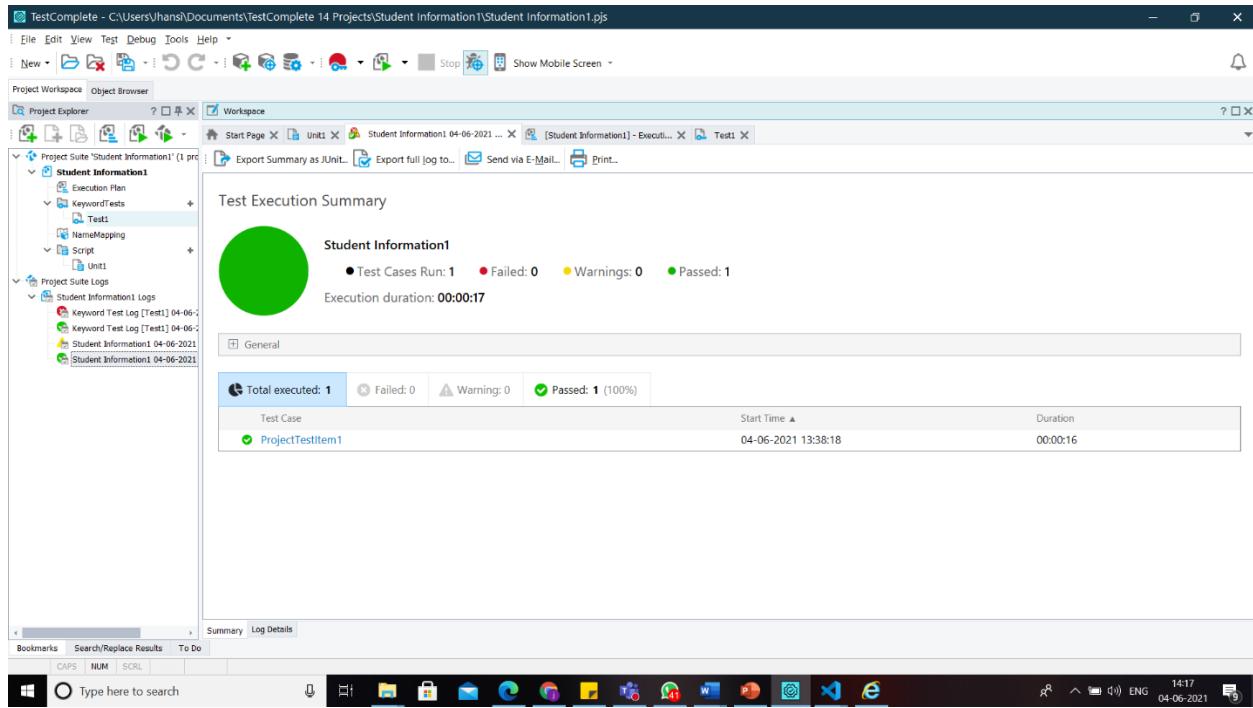
#### **Test case results:**



## Test success for login module:



## Test statistics:



## 2. Automated UI testing for Student Information Module:

### Test Cases to add student information module:

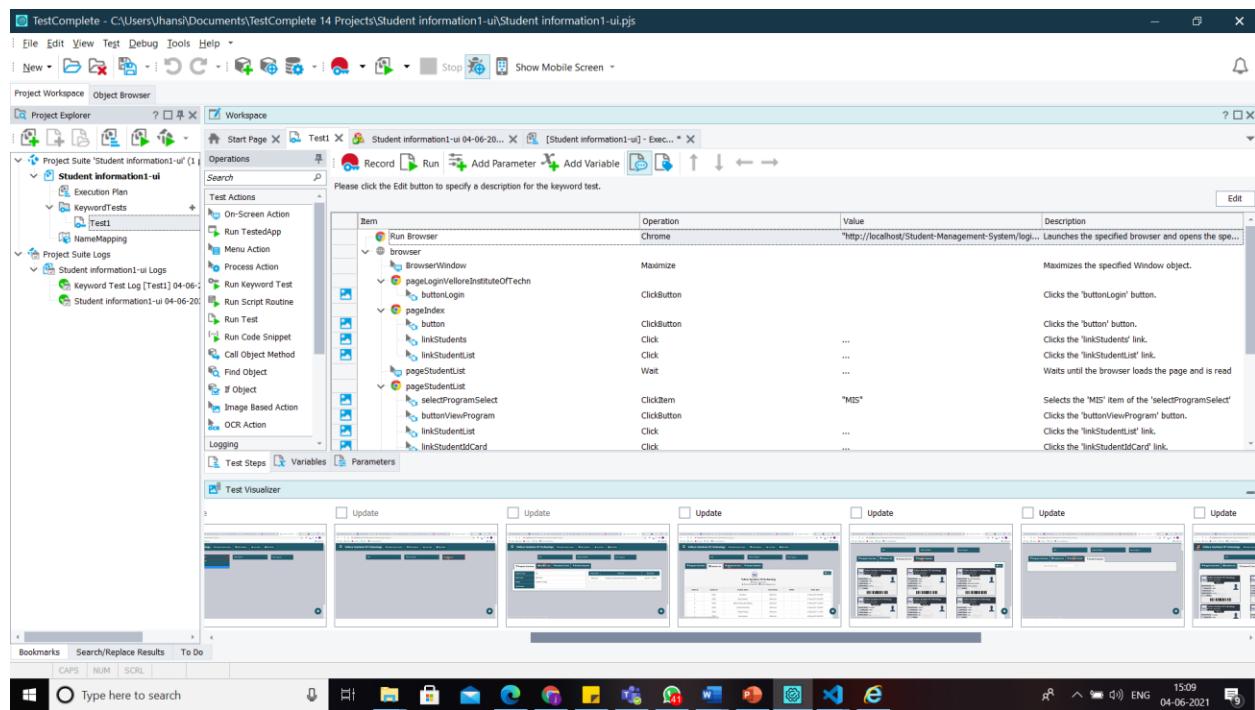
Test Case id	input	Expected Output	Actual Output	Status
1	To display list of students and id card: i. Select Students list button ii. Select program from drop down list iii. Select batch from dropdown list and view program iv. Select the option students list and student id card to get information of students	Student information details will be shown in table format from batch wise and id cards of each student will be displayed.	Student information details will be shown in table format from batch wise and id cards of each student will be displayed.	Pass
2	To display list of students and id card:	Student information details will be	Student information details will be	Pass

	<ul style="list-style-type: none"> <li>i. Select Students list button</li> <li>ii. Select program from drop down list</li> <li>iii. Select View program button</li> <li>iv. Select the option students list and student id card to get information of students</li> </ul>	<p>shown program wise from all batches in table format from and id cards of each student will be displayed.</p>	<p>shown program wise from all batches in table format from and id cards of each student will be displayed.</p>	
3	<ul style="list-style-type: none"> <li>i. Select Students list button</li> <li>ii. Select View program button</li> </ul>	All students lists from all programs and all batches	Please Select Program	Fail
4	<p>To add new student details:</p> <ul style="list-style-type: none"> <li>i. From the students list select, add student.</li> <li>ii. Fill all the details of the students.</li> </ul>	New Student added.	New Student added.	<b>Pass</b>
5	<p>To add new student details:</p> <ul style="list-style-type: none"> <li>i. From the students list select, add student.</li> <li>ii. Fill only relevant details.</li> </ul>	New student added.	Fill all the details to add new student.	Fail

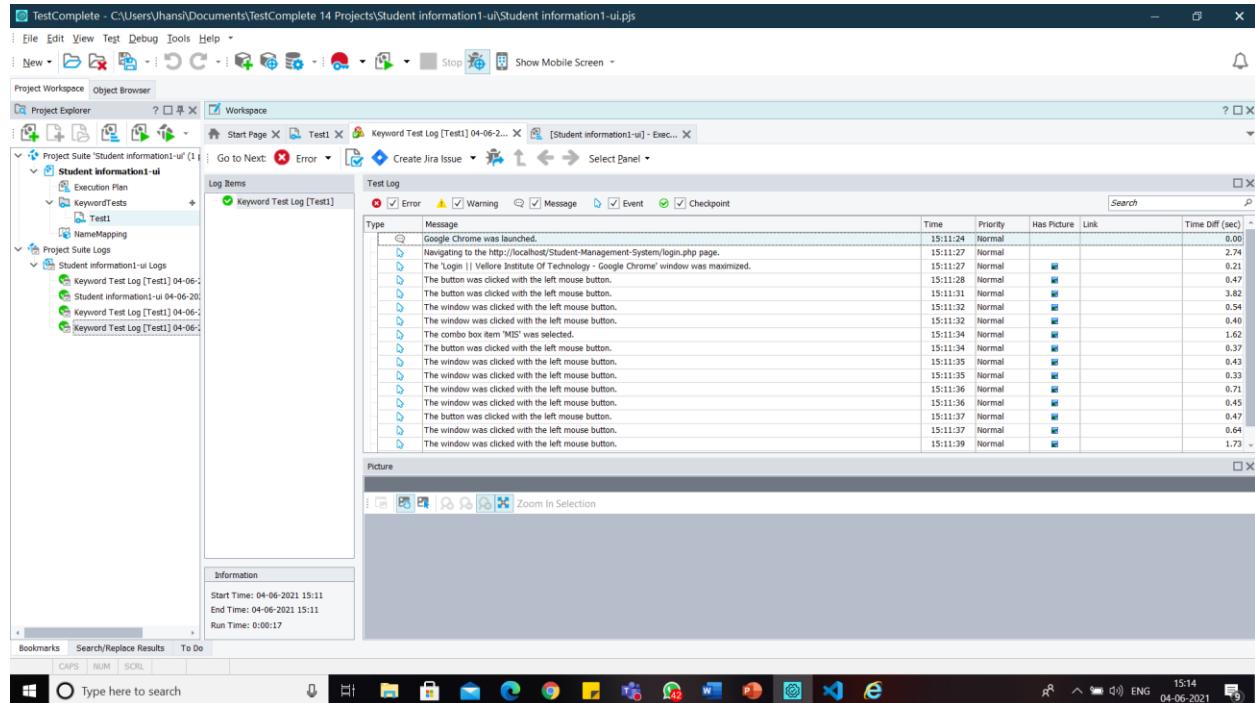
#### **Test complete tool:**

In test complete tool, it records all the actions of the student information and displays the tests performed while validating.

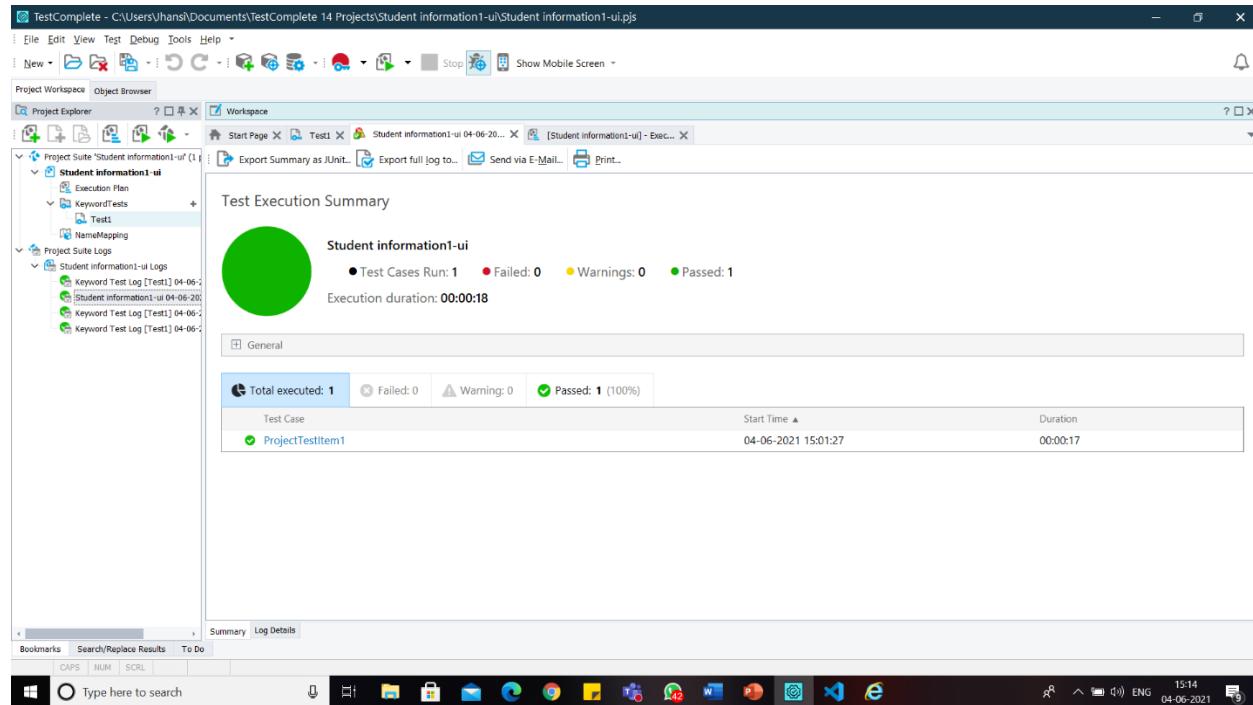
#### **Test case results:**



## Test success for Student Information module:



## Test statistics:



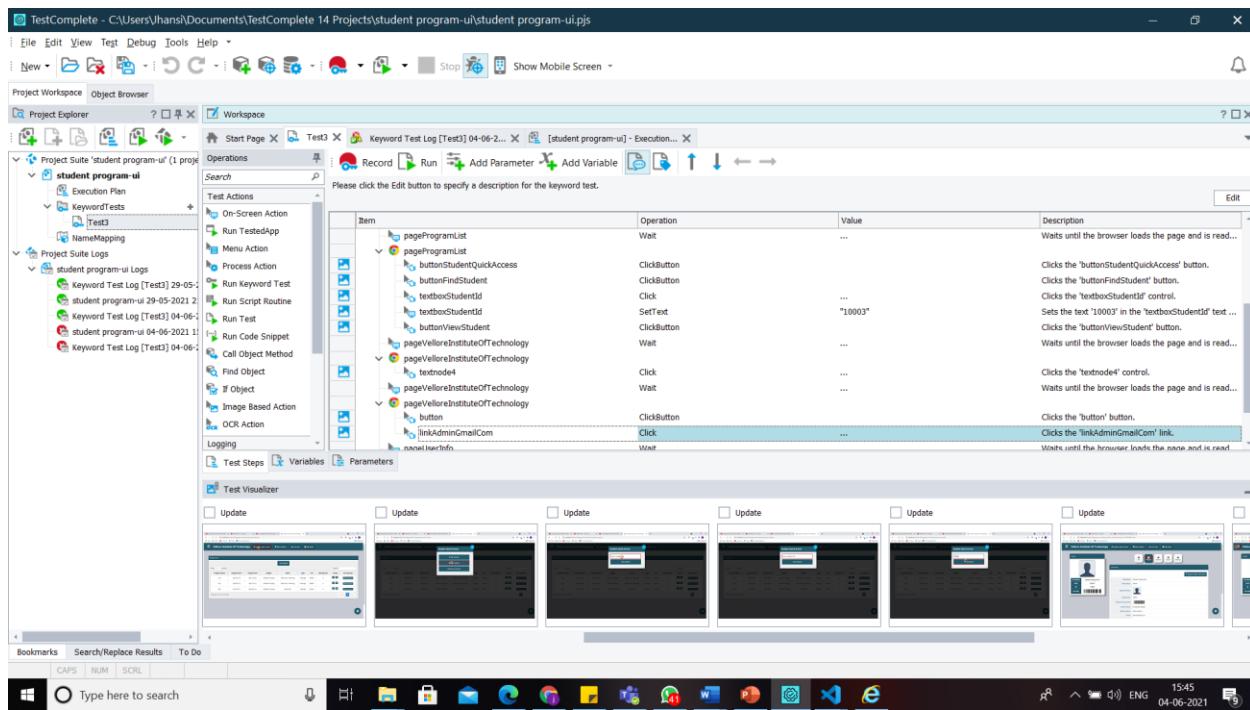
### 3. Automated UI testing for Student Program Module:

#### Test Cases to Admit Student in Multiple Program module:

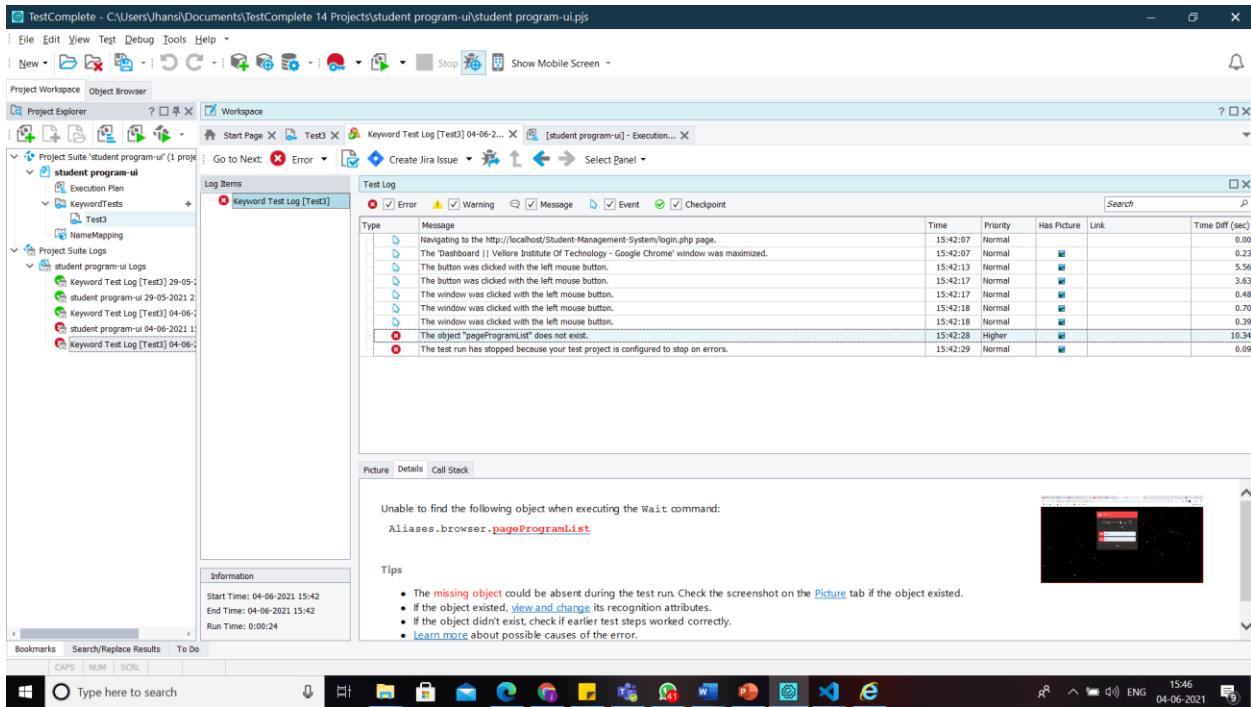
Test Case id	input	Expected Output	Actual Output	Status
1	To view all program details: Select Program Button in the left side of the dashboard page	All program details will be displayed.	All program details will be displayed.	Pass
2	To view all program details: To search by student id: Select student quick access Enter student id and select program option.	Student program details will be displayed for each student individually	Student program details will be displayed for each student individually	Pass

3	To view all program details: Select batch button in the left side of the dashboard page	All batch wise program details should display	Only batch details	Fail
4	To view students list from program wise: Select students list, Select program, batch and click on view program.	Program and batch wise list of students will be displayed.	Program and batch wise list of students will be displayed.	Pass
5	To view students list from program wise: Select students list, click on view program	List of all students list from all programs.	Select program	Fail

### Test Case results:



## Test Failure for Student Program module:



## Debugging:

We cannot display program details from batch option, to avoid this error we need to select program button which displays all program details and to check program allocation for each student we need to search from program id and select program option to display details.

## Test Success for Student Program module:

TestComplete - C:\Users\Jhans\Documents\TestComplete 14 Projects\student program-ui\student program-ui.pjs

File Edit View Test Debug Tools Help

New Stop Show Mobile Screen

Project Explorer Object Browser

Log Items

Test Log

Type Message Time Priority Has Picture Link Time Off (sec)

Navigation to URL: http://localhost/StudentManagementSystem/login.php page.	21:41:56	Normal		0.00
The window [1] 'Vollzeit-Berater der Technologie - Google Chrome' was maximized.	21:41:57	Normal		0.00
The button was clicked with the left mouse button.	21:41:57	Normal		0.53
The button was clicked with the left mouse button.	21:41:58	Normal		3.45
The window was clicked with the left mouse button.	21:41:59	Normal		0.71
The window was clicked with the left mouse button.	21:42:00	Normal		0.84
The window was clicked with the left mouse button.	21:42:02	Normal		0.36
The button was clicked with the left mouse button.	21:42:02	Normal		2.04
The window was clicked with the left mouse button.	21:42:03	Normal		0.54
The button was clicked with the left mouse button.	21:42:03	Normal		0.37
The window was clicked with the left mouse button.	21:42:08	Normal		0.17
The text '100%' was entered in the text editor.	21:42:08	Normal		0.35
The button was clicked with the left mouse button.	21:42:10	Normal		1.15
The window was clicked with the left mouse button.	21:42:10	Normal		0.50
The button was clicked with the left mouse button.	21:42:12	Normal		0.40
The window was clicked with the left mouse button.	21:42:12	Normal		1.04

Information

Start Time: 29-05-2021 21:43  
End Time: 29-05-2021 21:44  
Run Time: 0:00:18

Bookmarks Search/Replace Results To Do

Type here to search

15:49 04-06-2021

## Test Statistics:

TestComplete - C:\Users\Jhans\Documents\TestComplete 14 Projects\student program-ui\student program-ui.pjs

File Edit View Test Debug Tools Help

New Stop Show Mobile Screen

Project Explorer Object Browser

Test Execution Summary

student program-ui

● Test Cases Run: 1 ● Failed: 0 ● Warnings: 0 ● Passed: 1

Execution duration: 00:00:19

Total executed: 1 Failed: 0 Warning: 0 Passed: 1 (100%)

Test Case	Start Time	Duration
ProjectTestItem1	29-05-2021 21:44:29	00:00:17

Summary Log Details

Bookmarks Search/Replace Results To Do

Type here to search

15:49 04-06-2021

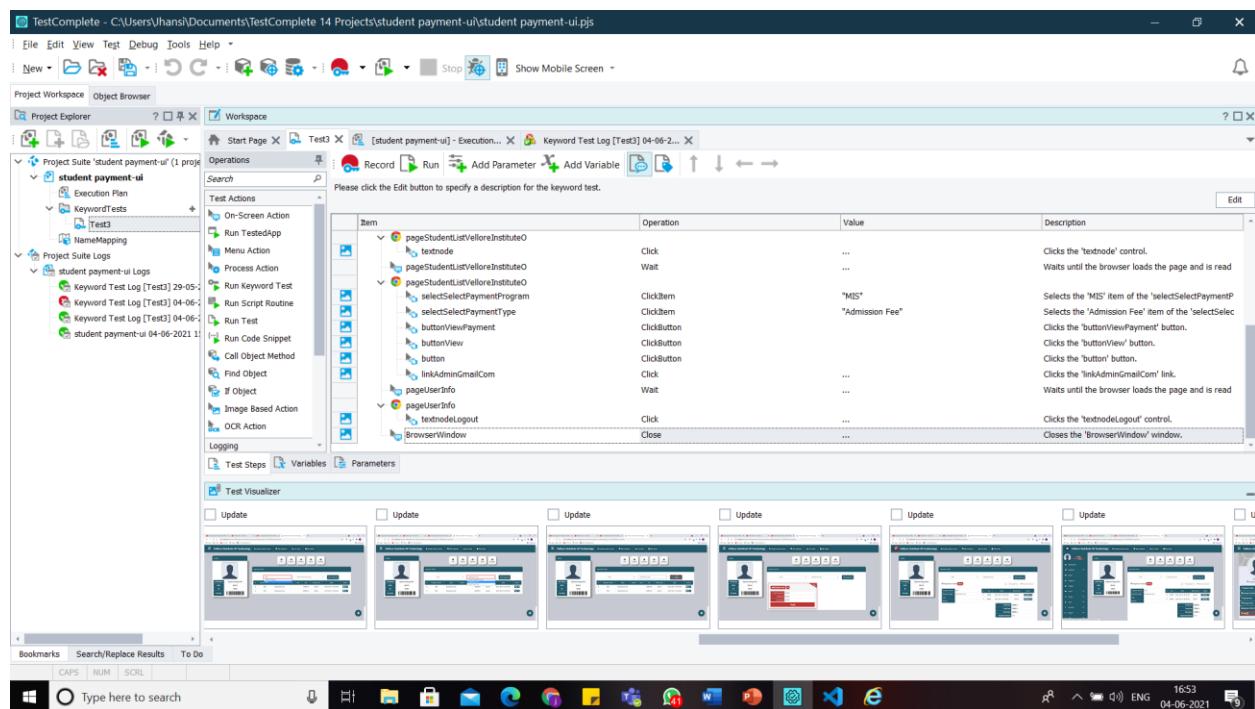
#### **4. Automated UI testing for Student Payment System Module:**

##### **Test Cases for student Payment system module:**

<b>Test Case id</b>	<b>input</b>	<b>Expected Output</b>	<b>Actual Output</b>	<b>Status</b>
1	To search by student id: i. Select student quick access Enter student id and select payment option. ii. In the payment panel, select program and select payment type and click on view payment option.	Payment details will be displayed in the form of amount paid, date details and received by and we can also print the payment receipt.	Payment details will be displayed in the form of amount paid, date details and received by and we can also print the payment receipt.	Pass
2	To search by student id: i. Select student quick access Enter student id and select payment option. ii. Click on view payment option	All payment details	Select program details	Fail
3	To search by student id: i. Select student quick access Enter student id and select payment option. ii. In the payment panel, select program and Click on view payment option	Payment details relation to program	Select payment type	Fail
4	To get payment statistics from program wise: i. Select students list, select program and batch details.	Payment overview, payment status, Paid and due students list and payment sms	Payment overview, payment status, Paid and due students list	Pass

	ii. Select student payment option and select payment type to get details.	system will be displayed.	and payment sms system will be displayed.	
5	To get payment statistics from program wise: i. Select students list, and select view details.	Payment status	Select program	Fail

## Test Case results:



## Test Failure:

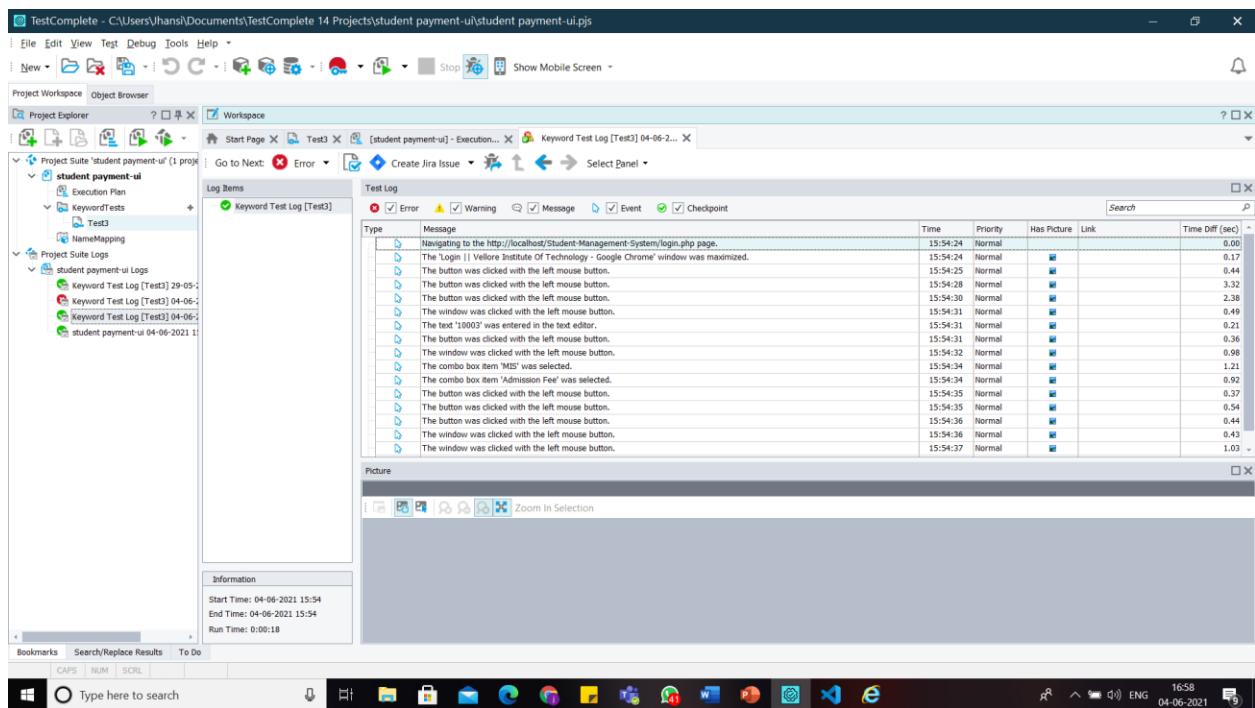
The screenshot shows the TestComplete interface with a project named 'student payment-ui'. The 'Test Log' window is open, displaying a list of log items. Most items are 'Message' type, indicating normal interactions like navigating to the login page and clicking buttons. One item is an 'Error' type, stating 'The combo box item 'Admission Fee' not found.' The log table has columns for Type, Message, Time, Priority, Has Picture, Link, and Time Diff (sec). The total run time is listed as 0:00:12.

Type	Message	Time	Priority	Has Picture	Link	Time Diff (sec)
Message	Navigating to the http://localhost/Student-Management-System/login.php page.	15:27:08	Normal			0.00
Message	The 'Login    Vellore Institute Of Technology - Google Chrome' window was maximized.	15:27:08	Normal			0.21
Message	The button was clicked with the left mouse button.	15:27:08	Normal			0.42
Message	The button was clicked with the left mouse button.	15:27:11	Normal			2.63
Message	The button was clicked with the left mouse button.	15:27:12	Normal			0.55
Message	The window was clicked with the left mouse button.	15:27:12	Normal			0.56
Message	The text '10000' was entered in the text editor.	15:27:12	Normal			0.21
Message	The button was clicked with the left mouse button.	15:27:13	Normal			0.50
Message	The window was clicked with the left mouse button.	15:27:14	Normal			0.85
Message	The combo box item 'MIS' was selected.	15:27:16	Normal			2.21
Error	The combo box item 'Admission Fee' not found.	15:27:16	Normal			0.30
Error	The test run has stopped because your test project is configured to stop on errors.	15:27:16	Normal			0.07

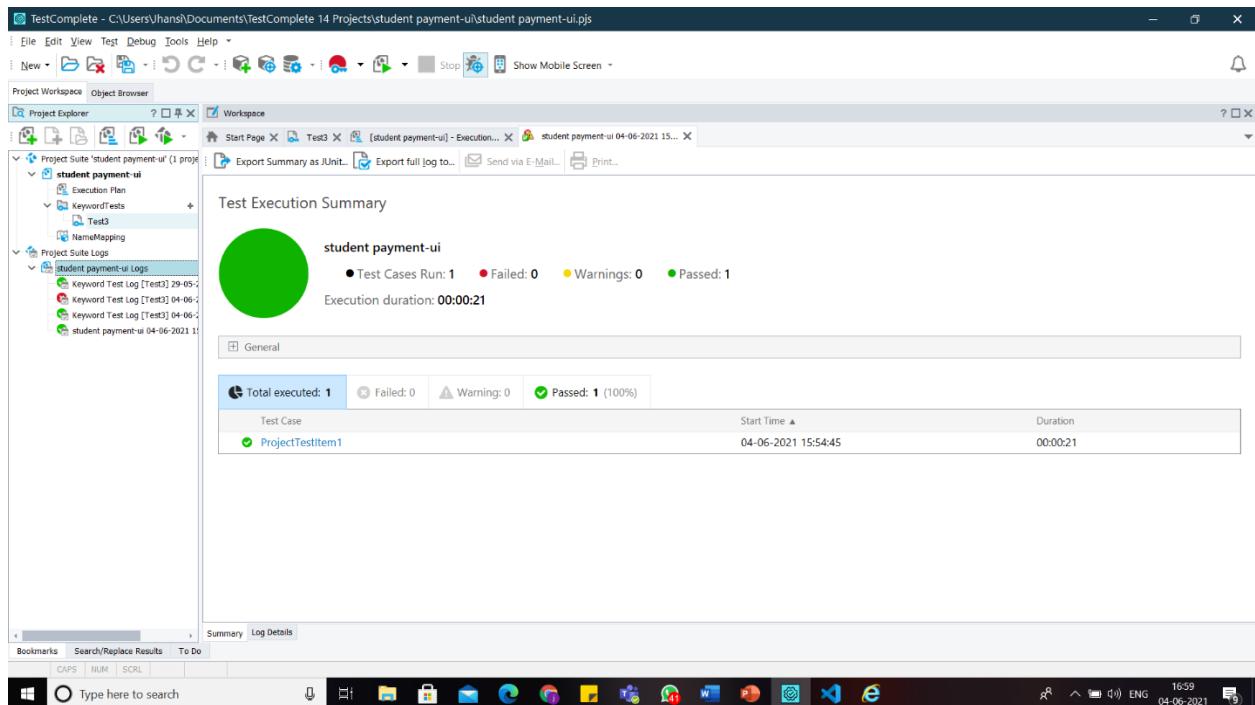
## Debugging:

To view the student payment details, we need to select the program wise and payment type and then go through the payment information and print the payment receipt.

## Test Success for Student Payment module:



## Test statistics:

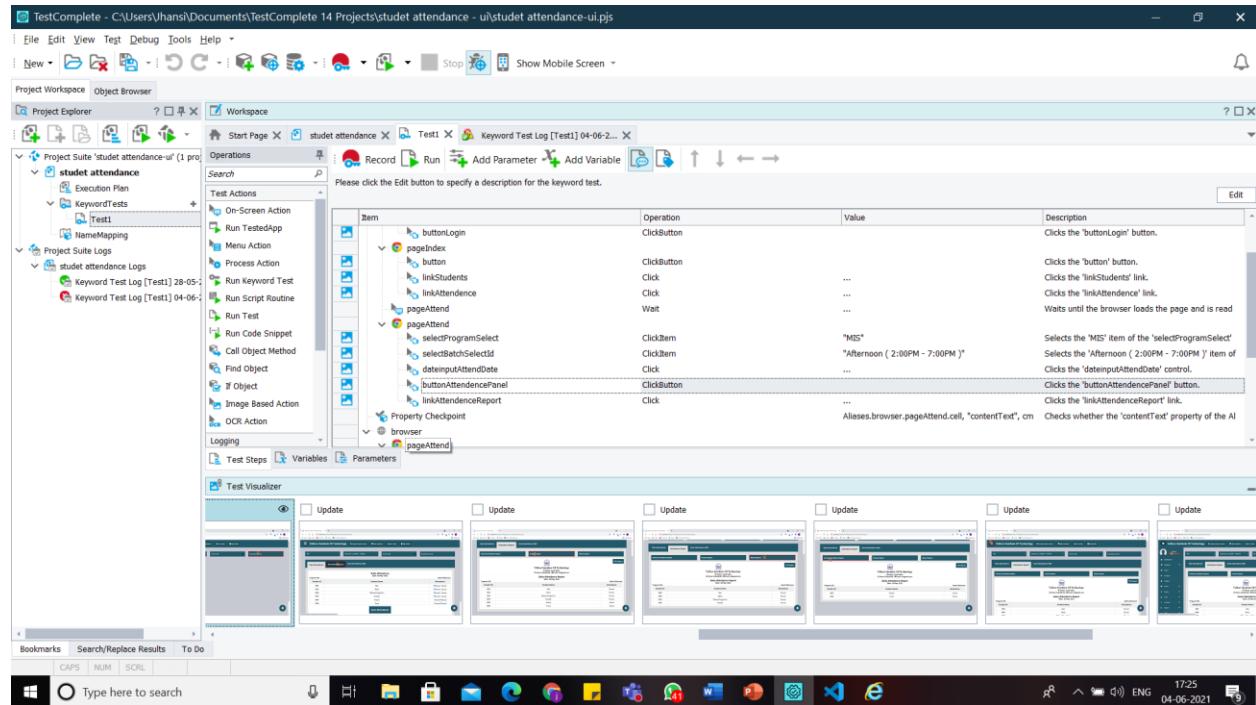


## **5. Automated UI testing for Student Attendance System Module:**

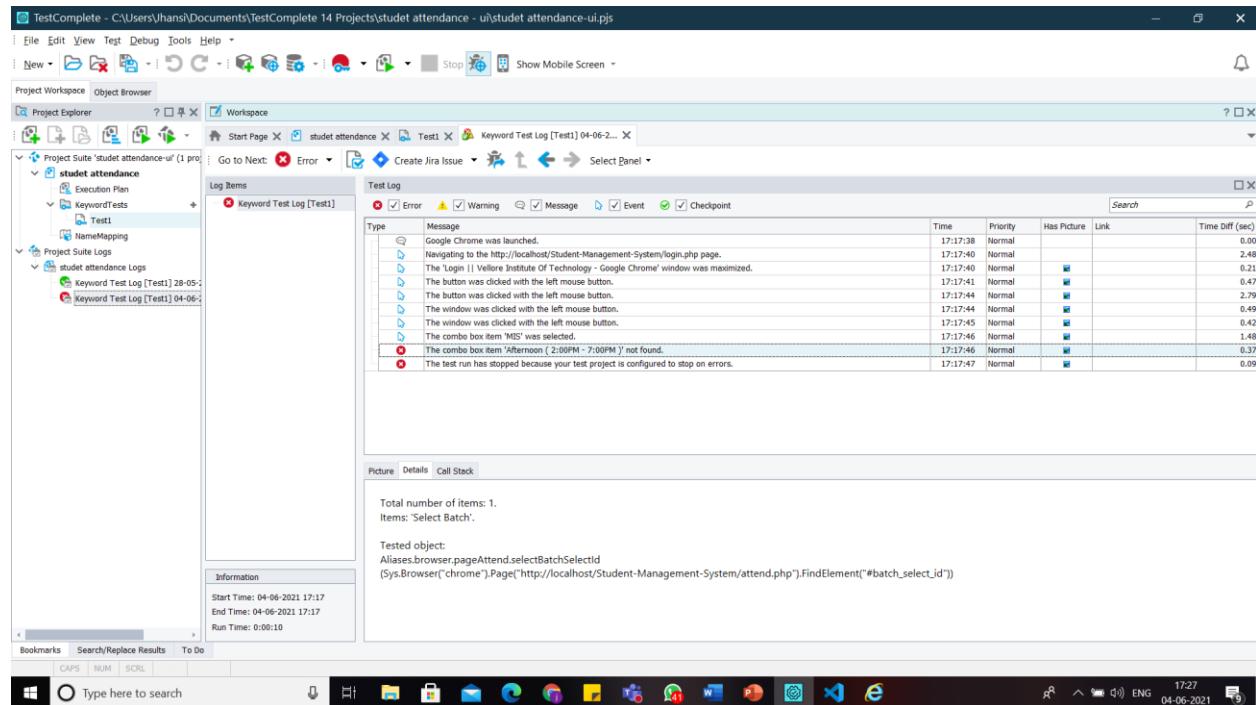
### **Test Cases to display student Attendance System details:**

<b>Test Case id</b>	<b>input</b>	<b>Expected Output</b>	<b>Actual Output</b>	<b>Status</b>
1	i. From students list, select attendance. ii. From the attendance panel, select program, batch, and give the date and select attendance panel.	Enter the attendance details and view details like presentees list and absentees list	Enter the attendance details and view details like presentees list and absentees list	Pass
2	i. From students list, select attendance. ii. Select attendance panel.	Attendance details	Select program details	Fail
3	i. From students list, select attendance. ii. From the attendance panel, select program, select attendance panel.	Attendance details related to program	Select batch details	Fail

## Test Case results:



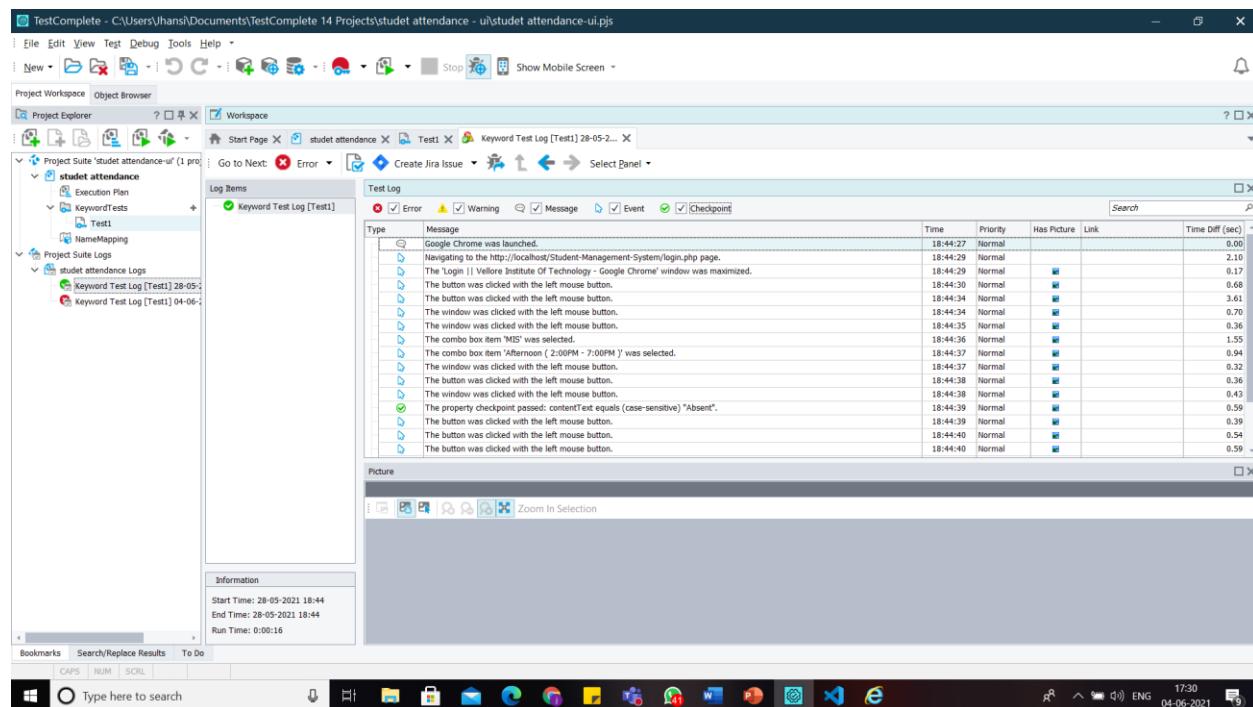
## Test Failure:



## Test Debugging:

To get the student information details, from the student list after selecting attendance option. From the Attendance panel ,we need to select program, batch details and give the date for the attendance details. So that we can view information in the form of all students attendance list and prsentees list and absenentes list.

## Test Success for Student Attendance module:



## 6. Automated UI testing for Institute Report Module:

### Test Cases to display Institute Report Module:

Test Case id	Input	Expected Output	Actual Output	Status
1	To generate payment report: i. Select report, from the report list select type of report to generate.	Payment report of students on the particular date will be displayed in the tabular format and report can be printed.	Payment report of students on the particular date will be displayed in the tabular format and	Pass

	<ul style="list-style-type: none"> <li>ii. Select payment report, select program, payment type and give the date details to get the report on the particular date</li> </ul>		report can be printed.	
2	<p>To generate payment report:</p> <ul style="list-style-type: none"> <li>i. Select report, from the report list select type of report to generate.</li> <li>ii. Select payment report, select program, payment type and select type.</li> </ul>	Payment report of students and we can also print the report	Please select date to generate report.	Fail
3	<p>To generate attendance report:</p> <ul style="list-style-type: none"> <li>i. Select report, from the report list select type of report to generate.</li> <li>ii. Select attendance report, select program, batch, month, year and click on view report option .</li> </ul>	Attendance report for the particular month of the year will be displayed and report can be printed.	Attendance report for the particular month of the year will be displayed and report can be printed.	Pass
4	<p>To generate attendance report:</p> <ul style="list-style-type: none"> <li>i. Select report, from the report list select type of report to generate.</li> <li>ii. Select attendance report, select program, batch, year and click on</li> </ul>	Attendance report need to generate and also can print the report.	Please select month to generate report	Fail

	view report option .			
--	----------------------	--	--	--

## Test Case results:

The screenshot shows the TestComplete 14 interface with a project named 'institute report-ui'. The 'Test Actions' table details the steps taken during the test, including selecting program items and entering dates. The 'Test Visualizer' pane displays the application's state at different points in the test run.

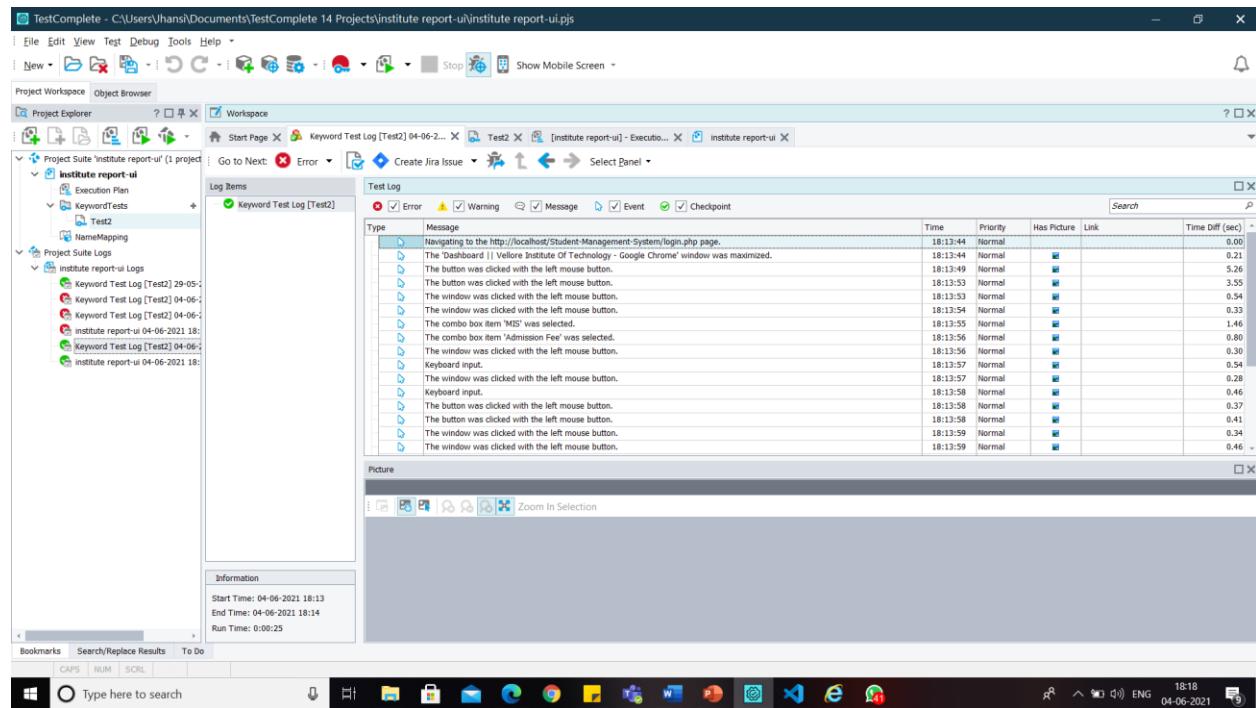
## Test Failure:

The screenshot shows the TestComplete 14 interface with a failed test log. The 'Log Items' table lists errors such as 'The combo box item 'MIS' was selected.' and 'The test run has stopped because your test project is configured to stop on errors.' The 'Information' pane provides details about the test duration.

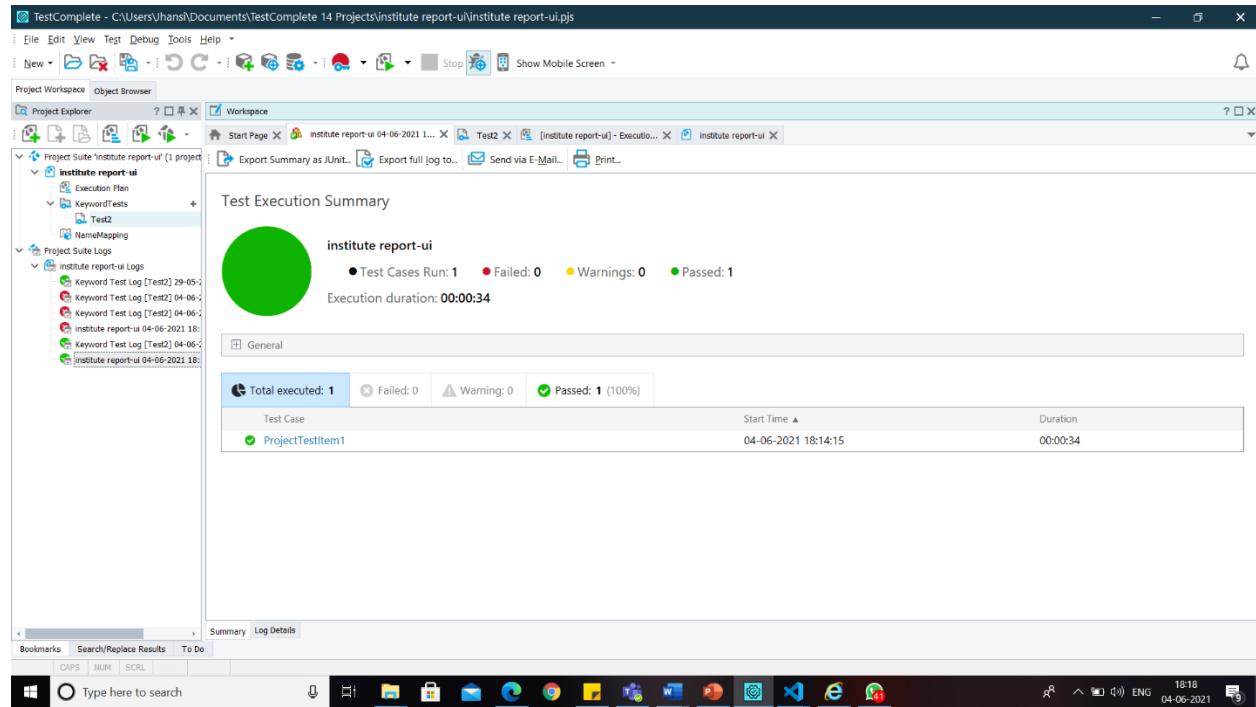
## Debugging:

To generate institute reports, we need to select type of report need to be generated. For payment report select program name, payment type and enter the details of date to generate payment report of students on particular date. To generate attendance report select program, batch, year, month to view the attendance report on particular date and reports can be printed.

## Test Success for Institute Report module:



## Test statistics:



### • Check Point testing:

In software development functional testing basically means checking if your software functions works well.

Functional testing steps:

1. List your application functions and features
2. Decide on input data.
3. Specify expected output.
4. Create test cases.
5. Compare actual and expected output.
6. Recording is easier in the test complete than the doing the testing manually. so we basically followed the recording method for the testing.
7. While testing is being done we can add the check points.
8. And we get the testing results based on the recording its done.
9. We will finally get whether testing is successful or not in the result.
10. Check points are credible aspects of software testing the work with the user actions

## 1. Check Point Testing for Student information Module:

### Test Complete:

New Student details will be added by the admin to the data base and while testing is done we can add checkpoints to validate the type of elements. And the testing methods will be recorded.

### Test Case Results:

The screenshot shows the TestComplete application interface. The main window displays a test log titled 'student information' with several test steps and their corresponding actions. The 'Operations' pane on the left lists various test actions like 'On-Screen Action', 'Run TestedApp', 'Process Action', etc. The 'Test Steps' pane on the right shows a detailed list of actions with their descriptions and values. The 'Test Visualizer' pane at the bottom shows a sequence of screenshots illustrating the test flow. The taskbar at the bottom of the screen shows various open applications.

Item	Operation	Value	Description
Navigate	Chrome	"http://localhost/Student-Management-System/login"	Opens the specified URL in a running instance of the browser.
browser	Maximize		Maximizes the specified Window object.
buttonLogin	Click@button		Clicks the 'buttonLogin' button.
pageIndex	Click@button		Clicks the 'button' button.
linkStudents	Click@link		Clicks the 'linkStudents' link.
linkAddStudent	Click@link		Clicks the 'linkAddStudent' link.
pageAddStudent	Wait		Waits until the browser loads the page and is ready.
textboxName	Click@text		Clicks the 'textboxName' control.
textboxName	SetText	"Swarna"	Sets the text 'Swarna' in the 'textboxName' text editor.
textboxClick	Click@text		Clicks the 'textboxClick' control.
textboxClick	SetText	"sapna"	Sets the text 'sapna' in the 'textboxClick' text editor.

### Test Success for Student Information module:

TestComplete - C:\Users\Jhansi\Documents\TestComplete 14 Projects\student information\student information.pjs

File Edit View Test Debug Tools Help

New Stop Show Mobile Screen

Project Explorer Object Browser

Project Suite 'student information' (1 project)

- student information
  - Execution Plan
  - KeywordTests
    - Test1
  - NameMapping
  - Script
    - Unit1
- Project Suite Logs
- student information Logs
  - Keyword Test Log [Test1] 27-05-2021 08:48
  - Keyword Test Log [Test1] 27-05-2021 08:49
  - Keyword Test Log [Test1] 28-05-2021 08:49
  - Keyword Test Log [Test1] 29-05-2021 08:49
  - Keyword Test Log [Test1] 29-05-2021 08:49
  - student information 29-05-2021 08:49
  - student information 29-05-2021 08:49
  - Keyword Test Log [Test1] 04-06-2021 08:49

Log Items Test Log

Keyword Test Log [Test1]

Type Message Time Priority Has Picture Link Time Diff (sec)

Information	The window was clicked with the left mouse button.	8:49:10	Normal		0.29		
Information	The text "VIT" was entered in the text editor.	8:49:11	Normal		0.22		
Information	The window was clicked with the left mouse button.	8:49:11	Normal		0.32		
Information	Keyboard input.	8:49:11	Normal		0.23		
Information	The window was clicked with the left mouse button.	8:49:12	Normal		0.30		
Information	Keyboard input.	8:49:12	Normal		0.24		
Information	The window was clicked with the left mouse button.	8:49:12	Normal		0.22		
Information	The test 'abc' was entered in the text editor.	8:49:12	Normal		0.16		
Information	The window was clicked with the left mouse button.	8:49:13	Normal		0.31		
Information	Keyboard input.	8:49:13	Normal		0.24		
Information	The button was clicked with the left mouse button.	8:49:13	Normal		0.23		
Information	The property checkpoint passed: contentText equals (case-sensitive) "Swapna".	8:49:14	Normal		1.20		
Information	The button was clicked with the left mouse button.	8:49:15	Normal		0.37		
Information	The window was clicked with the left mouse button.	8:49:15	Normal		0.35		
Information	The window was clicked with the left mouse button.	8:49:16	Normal		0.95		
Information	The 'Login		Vellore Institute Of Technology - Google Chrome' window was closed.	8:49:17	Normal		1.20

Picture

Information

Start Time: 29-05-2021 08:48  
End Time: 29-05-2021 08:49  
Run Time: 0:00:27

CAPS NUM SCR To Do

18:42 ENG 04-06-2021

## Test Statistics:

TestComplete - C:\Users\Jhansi\Documents\TestComplete 14 Projects\student information\student information.pjs

File Edit View Test Debug Tools Help

New Stop Show Mobile Screen

Project Explorer Object Browser

Project Suite 'student information' (1 project)

- student information
  - Execution Plan
  - KeywordTests
    - Test1
  - NameMapping
  - Script
    - Unit1
- Project Suite Logs
- student information Logs
  - Keyword Test Log [Test1] 27-05-2021 08:48
  - Keyword Test Log [Test1] 27-05-2021 08:49
  - Keyword Test Log [Test1] 28-05-2021 08:49
  - Keyword Test Log [Test1] 29-05-2021 08:49
  - Keyword Test Log [Test1] 29-05-2021 08:49
  - student information 29-05-2021 08:49
  - student information 29-05-2021 08:49
  - Keyword Test Log [Test1] 04-06-2021 08:49

Test Execution Summary

student information

Total executed: 1 Failed: 0 Warnings: 0 Passed: 1

Execution duration: 00:00:32

General

Total executed: 1	Failed: 0	Warning: 0	Passed: 1 (100%)
Test Case	Start Time	Duration	
ProjectTestItem1	29-05-2021 08:49:26	00:00:31	

Summary Log Details

CAPS NUM SCR To Do

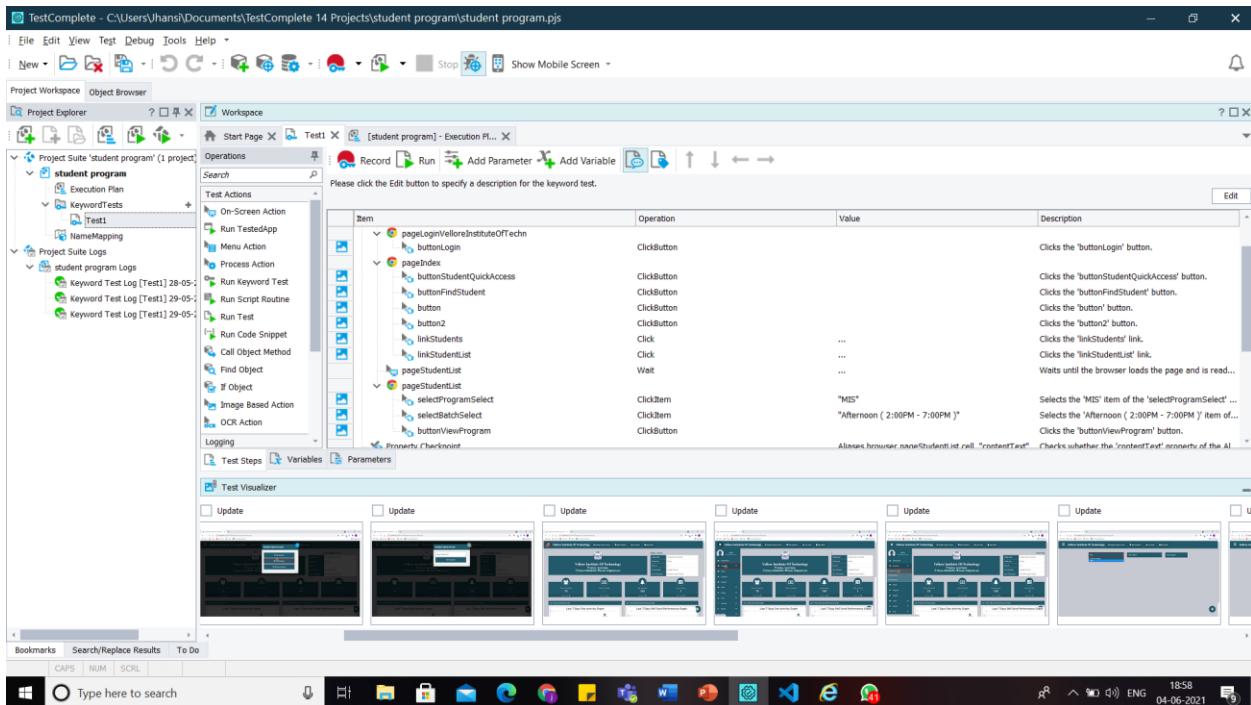
18:43 ENG 04-06-2021

## 2. Check Point Testing for Admit Student in multiple program Module:

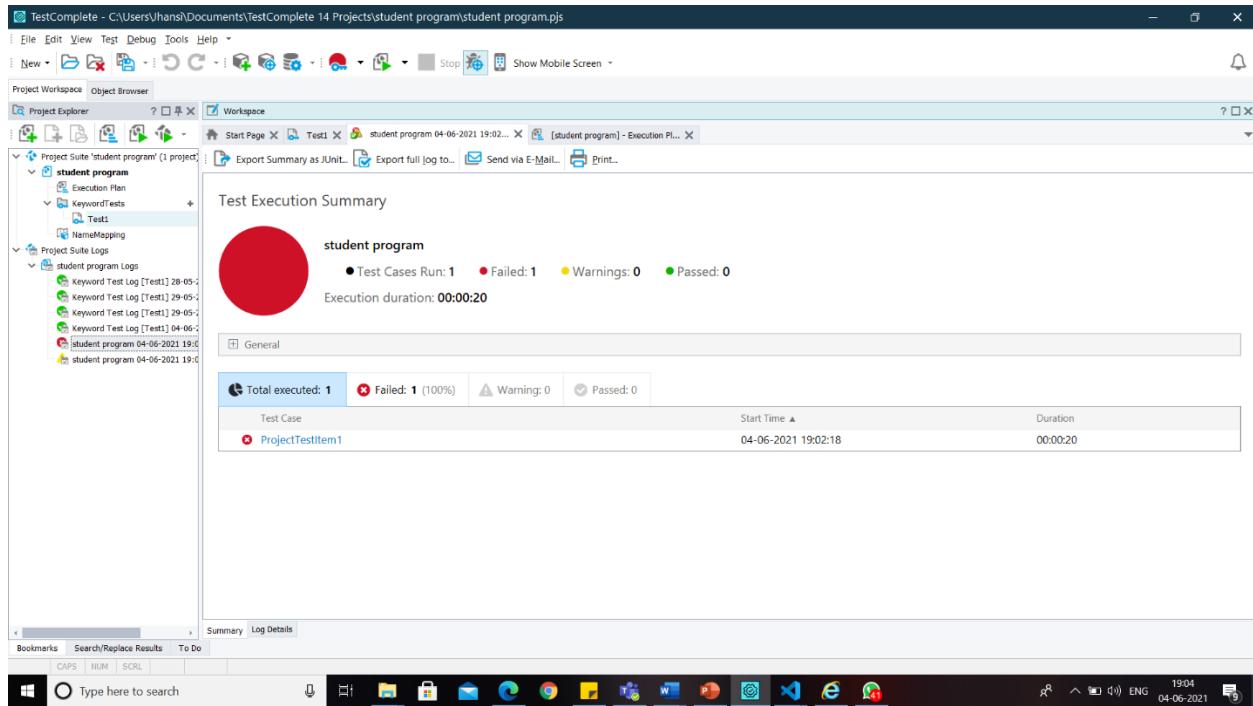
### Test Complete:

To display students list with program allocation, need to select program and batch details to view list of students allocated to particular program.

### Test Case Results:



### Test Failure:



## Debugging:

To get the student details allocated to particular program, from the student list select program and batch details from the dropdown list and click on view. Students details allocated to particular program on batch wise will be displayed.

## Test Success for Admit Student in multiple program Module:

TestComplete - C:\Users\Jhansi\Documents\TestComplete 14 Projects\student program\student program.pjs

File Edit View Test Debug Tools Help

New Stop Show Mobile Screen

Project Workspace Object Browser

Project Explorer Go to Next: Error Create Jira Issue Select Panel

Log Items Test Log Keyword Test Log [Test1]

Type Message Time Priority Has Picture Link Time Diff (sec)

- Google Chrome was launched. 18:59:13 Normal 0.00
- Navigating to the http://localhost/Student-Management-System/login.php page. 18:59:16 Normal 3.18
- The 'Login !! Veluru Institute Of Technology - Google Chrome' window was maximized. 18:59:16 Normal 0.24
- The button was clicked with the left mouse button. 18:59:17 Normal 0.52
- The button was clicked with the left mouse button. 18:59:19 Normal 0.23
- The button was clicked with the left mouse button. 18:59:20 Normal 0.60
- The button was clicked with the left mouse button. 18:59:21 Normal 0.51
- The window was clicked with the left mouse button. 18:59:21 Normal 0.56
- The window was clicked with the left mouse button. 18:59:22 Normal 0.41
- The combo box item 'MIS' was selected. 18:59:23 Normal 1.39
- The combo box item 'Afternoon ( 2:00PM - 7:00PM )' was selected. 18:59:24 Normal 0.84
- The button was clicked with the left mouse button. 18:59:24 Normal 0.37
- The property checkpoint passed: contentText equals (case-sensitive) "MIS".** 18:59:25 Normal 0.52
- The window was clicked with the left mouse button. 18:59:25 Normal 0.28
- The window was clicked with the left mouse button. 18:59:26 Normal 0.40

Information Start Time: 04-06-2021 18:59 End Time: 04-06-2021 18:59 Run Time: 0:00:18

Bookmarks Search/Replace Results To Do

CAPS NUM SCR Type here to search 18:59 ENG 04-06-2021

## Test Statistics:

TestComplete - C:\Users\Jhansi\Documents\TestComplete 14 Projects\student program\student program.pjs

File Edit View Test Debug Tools Help

New Stop Show Mobile Screen

Project Workspace Object Browser

Project Explorer Export Summary as JUnit Export full log to... Send via E-Mail Print...

Test Execution Summary

student program

● Test Cases Run: 1 ● Failed: 0 ● Warnings: 0 ● Passed: 1

Execution duration: 00:00:20

General

Total executed: 1	Failed: 0	Warning: 0	Passed: 1 (100%)
Test Case			Start Time Duration
ProjectTestItem1			04-06-2021 19:07:30 00:00:20

Summary Log Details

CAPS NUM SCR Type here to search 19:07 ENG 04-06-2021

### 3. Check Point Testing for Student Payment System Module:

#### Test Complete:

To get payment statistics from program wise:

Select students list, select program and batch details, select student payment option and select payment type to get details. Payment overview, payment status, Paid and due students list and payment sms system will be displayed.

#### Test Case Results:

The screenshot shows the TestComplete application interface. The title bar reads "TestComplete - C:\Users\Jhansi\Documents\TestComplete 14 Projects\student payment\student payment.pjs". The main window displays a "Project Explorer" tree on the left containing "student payment" (Execution Plan, KeywordTests, NameMapping), "Project Suite Logs" (student payment Log [Test1] 28-05-2021, student payment Log [Test1] 29-05-2021), and "Test1" (Execution Plan). The "Test Actions" pane on the right lists various operations such as Run Browser, Click@button, Click, Drag, etc., with their descriptions and values. Below the actions is a "Test Visualizer" pane showing a sequence of screenshots of a web application interface. The taskbar at the bottom shows several open applications, including Microsoft Edge, File Explorer, and other development tools.

#### Test failure:

**Test Log - Keyword Test Log [Test1]**

Type	Message	Time	Priority	Has Picture	Link	Time Diff (sec)
Information	The browser is already running.	19:24:21	Normal			0.00
Information	Navigating to the http://localhost/Student-Management-System/login.php page.	19:24:23	Normal			2.76
Information	The 'Dashboard    Vellore Institute Of Technology - Google Chrome' window was maximized.	19:24:24	Normal			0.25
Information	The button was clicked with the left mouse button.	19:24:25	Normal			0.74
Information	The button was clicked with the left mouse button.	19:24:33	Normal			3.32
Information	The window was clicked with the left mouse button.	19:24:34	Normal			0.48
Information	The window was clicked with the left mouse button.	19:24:34	Normal			0.39
Information	The drag operation was performed with the left mouse button.	19:24:35	Normal			0.99
Information	The combo box item 'MIS' was selected.	19:24:36	Normal			1.58
Error	The combo box item 'Afternoon ( 2:00PM - 7:00PM )' not found.	19:24:37	Normal			0.96
Error	The test run has stopped because your test project is configured to stop on errors.	19:24:37	Normal			0.10

Total number of items: 1.  
Items: 'Select Batch'.  
Tested object:  
Aliases.browser.pageStudentList.selectBatchSelect  
(Sys.Browser("chrome").Page("http://localhost/Student-Management-System/student\_list.php").FindElement("#batch\_select"))

Information

Start Time: 04-06-2021 19:24  
End Time: 04-06-2021 19:24  
Run Time: 0:00:17

## Debugging:

We need to select all the details correctly, to view payment status program wise. For that program and batch need to be selected correctly.

## Test Success for Student Payment System Module:

**Test Log - Keyword Test Log [Test1]**

Type	Message	Time	Priority	Has Picture	Link	Time Diff (sec)
Information	The window was clicked with the left mouse button.	8:39:33	Normal			0.41
Information	The drag operation was performed with the left mouse button.	8:39:34	Normal			1.09
Information	The combo box item 'MIS' was selected.	8:39:37	Normal			2.58
Information	The combo box item 'Afternoon ( 2:00PM - 7:00PM )' was selected.	8:39:38	Normal			0.86
Information	The button was clicked with the left mouse button.	8:39:38	Normal			0.38
Information	The window was clicked with the left mouse button.	8:39:38	Normal			0.38
Information	The combo box item 'Admission Fee' was selected.	8:39:40	Normal			1.09
Information	The button was clicked with the left mouse button.	8:39:40	Normal			0.52
Information	The button was clicked with the left mouse button.	8:39:40	Normal			0.37
Information	The button was clicked with the left mouse button.	8:39:41	Normal			0.47
Information	The button was clicked with the left mouse button.	8:39:41	Normal			0.38
Success	The property checkpoint passed: contentText equals (case-sensitive) "25000".	8:39:42	Normal			0.33
Information	The button was clicked with the left mouse button.	8:39:42	Normal			0.43
Information	The window was clicked with the left mouse button.	8:39:42	Normal			0.37
Information	The window was clicked with the left mouse button.	8:39:43	Normal			0.85
Information	The 'Login    Vellore Institute Of Technology - Google Chrome' window was closed.	8:39:44	Normal			0.49

Picture

Zoom In Selection

25000

Information

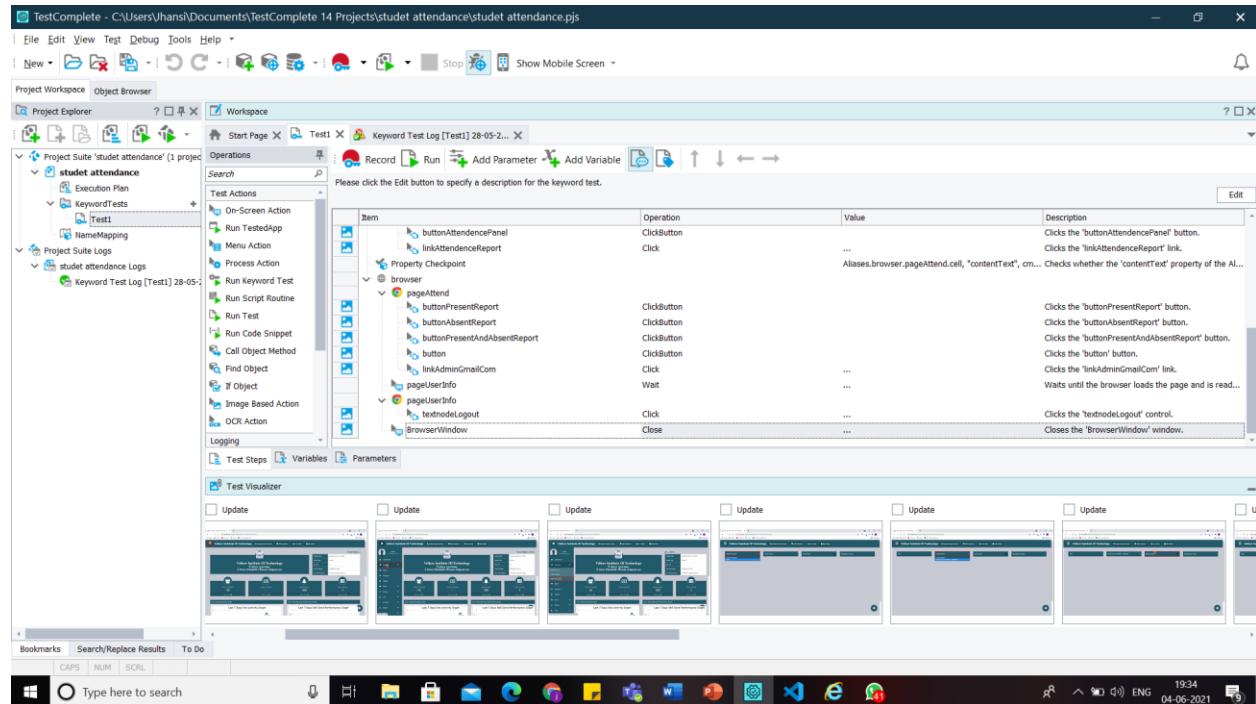
Start Time: 29-05-2021 08:39  
End Time: 29-05-2021 08:39  
Run Time: 0:00:19

#### 4. Check Point Testing for Student Attendance System Module:

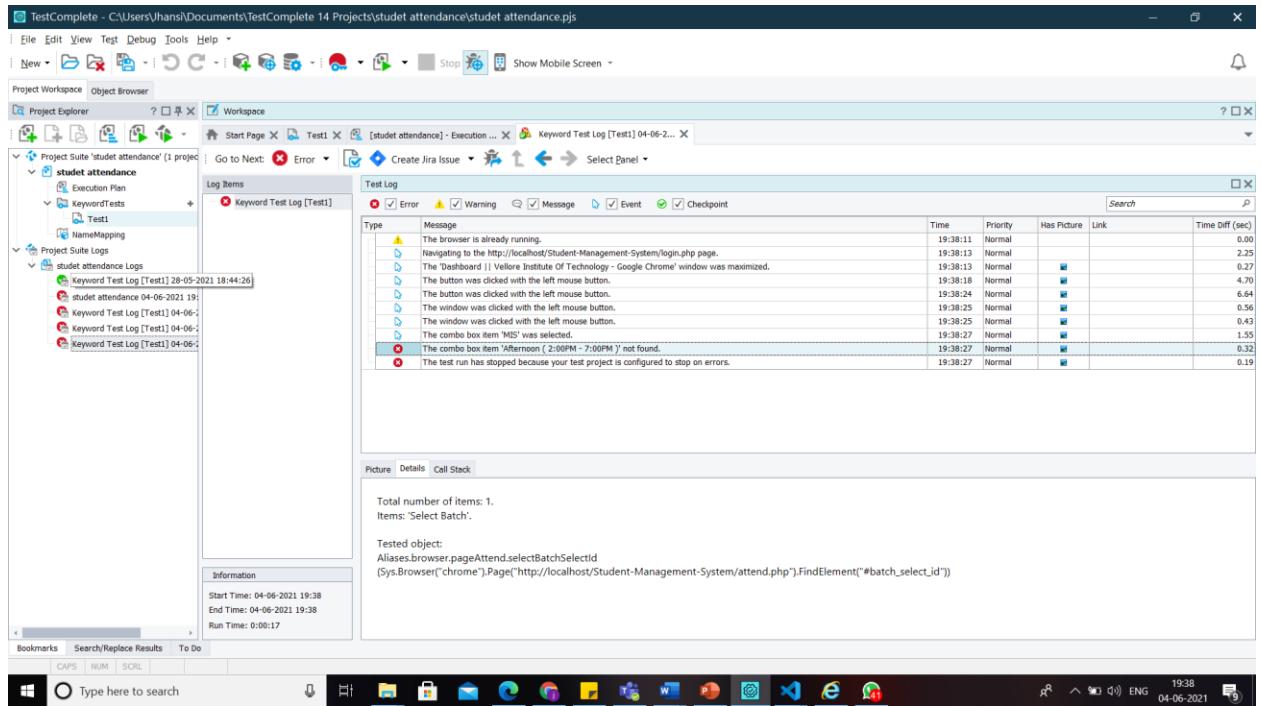
##### Test Complete:

To view Student attendance details, from students list, select attendance, from the attendance panel, select program, batch, and give the date and select attendance panel. Enter the attendance details and view details like presentees list and absentees list.

##### Test Case Results:



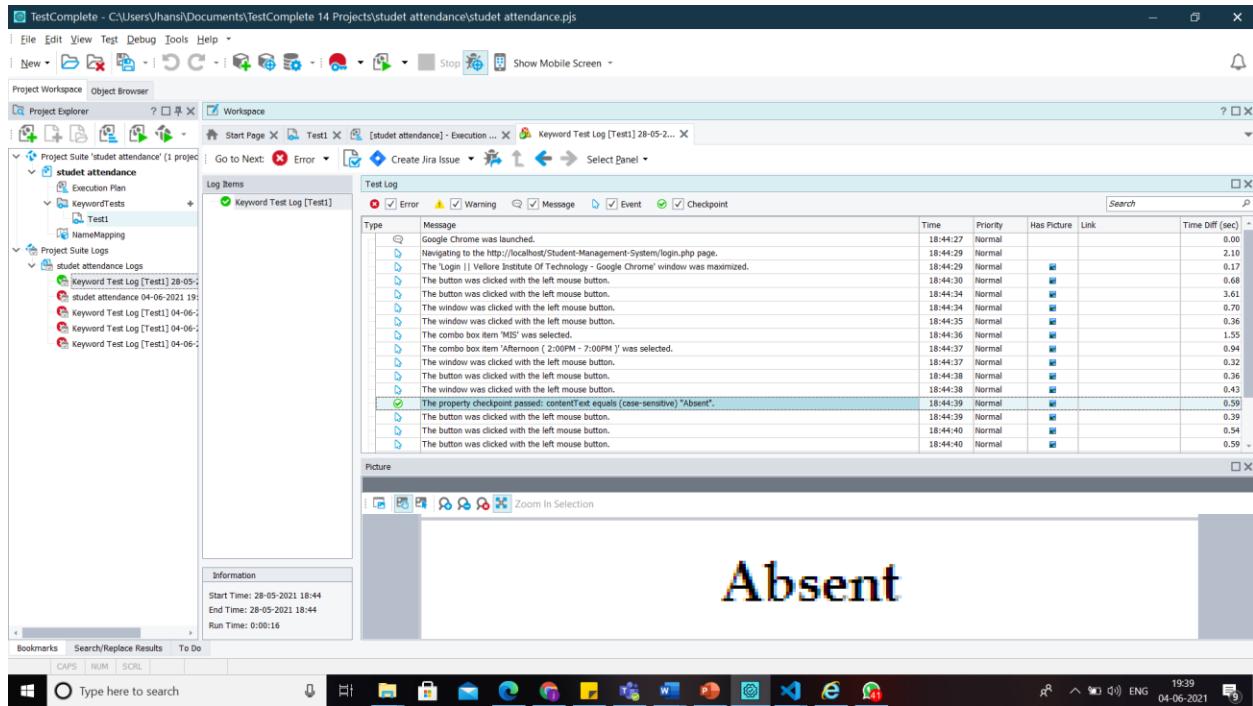
##### Test Failure:



## Test Debugging:

To get the student information details, from the student list after selecting attendance option. From the Attendance panel ,we need to select program, batch details and give the date for the attendance details. So that we can view information in the form of all students attendance list and prsentees list and absenentes list.

## Test Success for Student Attendance module:



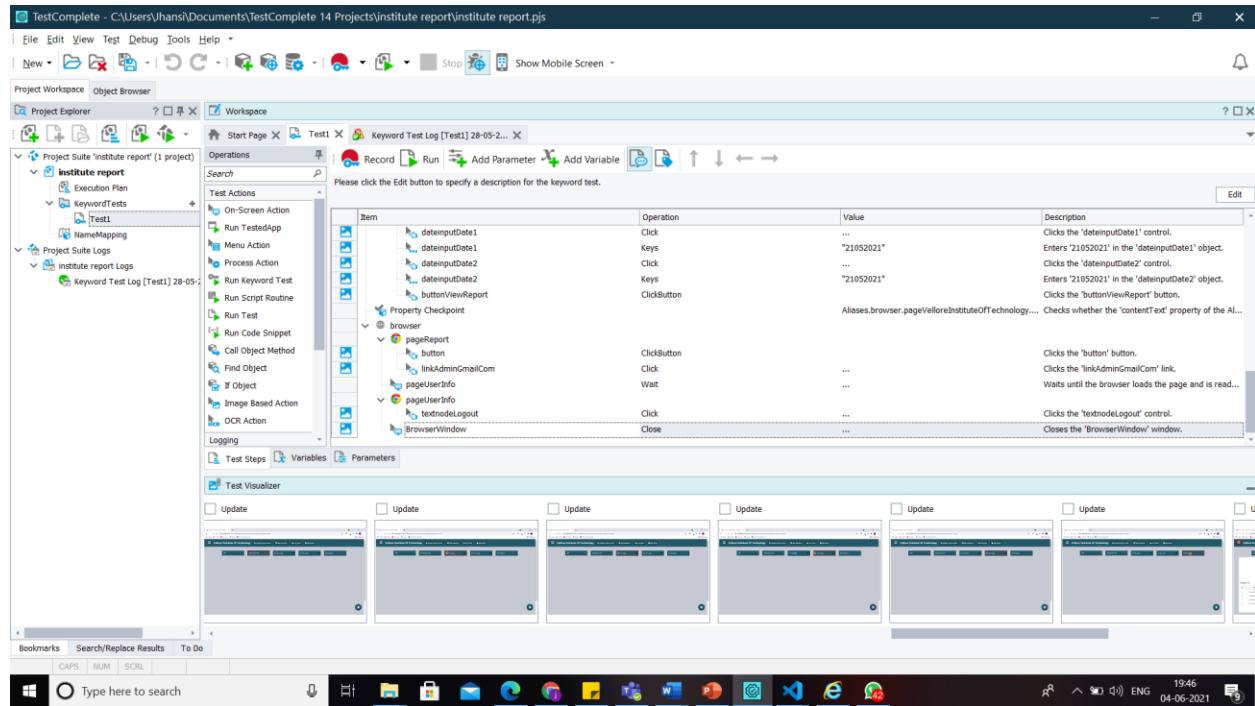
## 5. Check Point Testing for Institute Report Module:

### Test Complete:

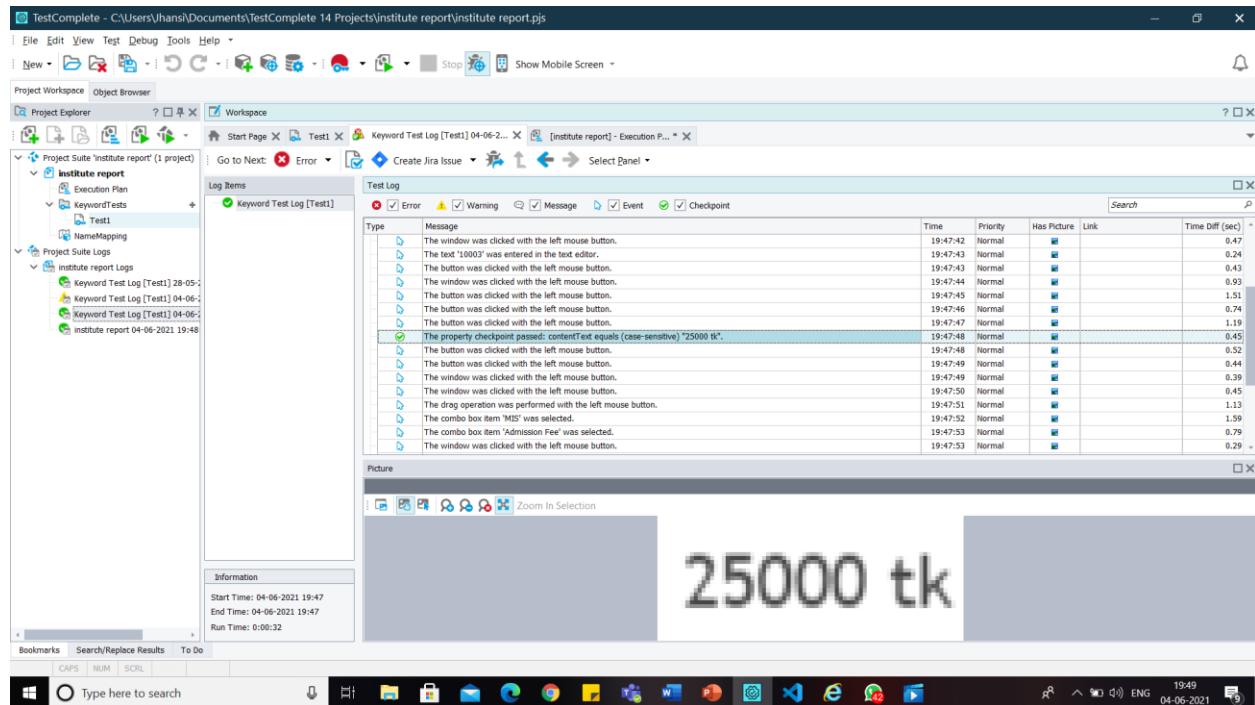
To generate payment report, select report, from the report list select type of report to generate, Select payment report, select program, payment type and give the date details to get the report on the particular date.

To generate report student wise, search for student and select payment option payment details and select generate report.

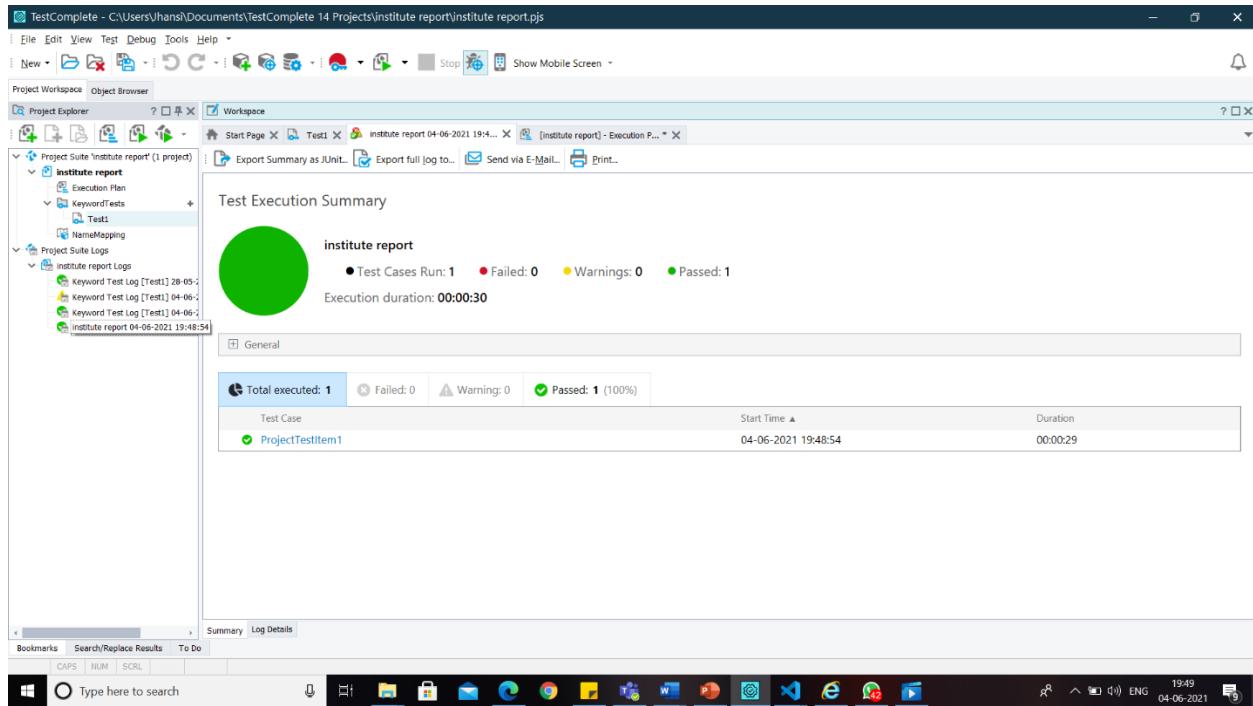
### Test Case Results:



## Test Success for Institute Report module:



## Test statistics:



- **Cross Browser testing:**

Cross-browser testing helps you ensure that your web site or web application functions correctly in various web browsers.

TestComplete includes a number of features that make creating cross-browser tests easier. You can record or create a test for one of supported web browsers and then run the test with no or little modifications in other supported web browsers.

So to run this in other browser we select the particular browser and what are the actions performed will be tested if we select it.

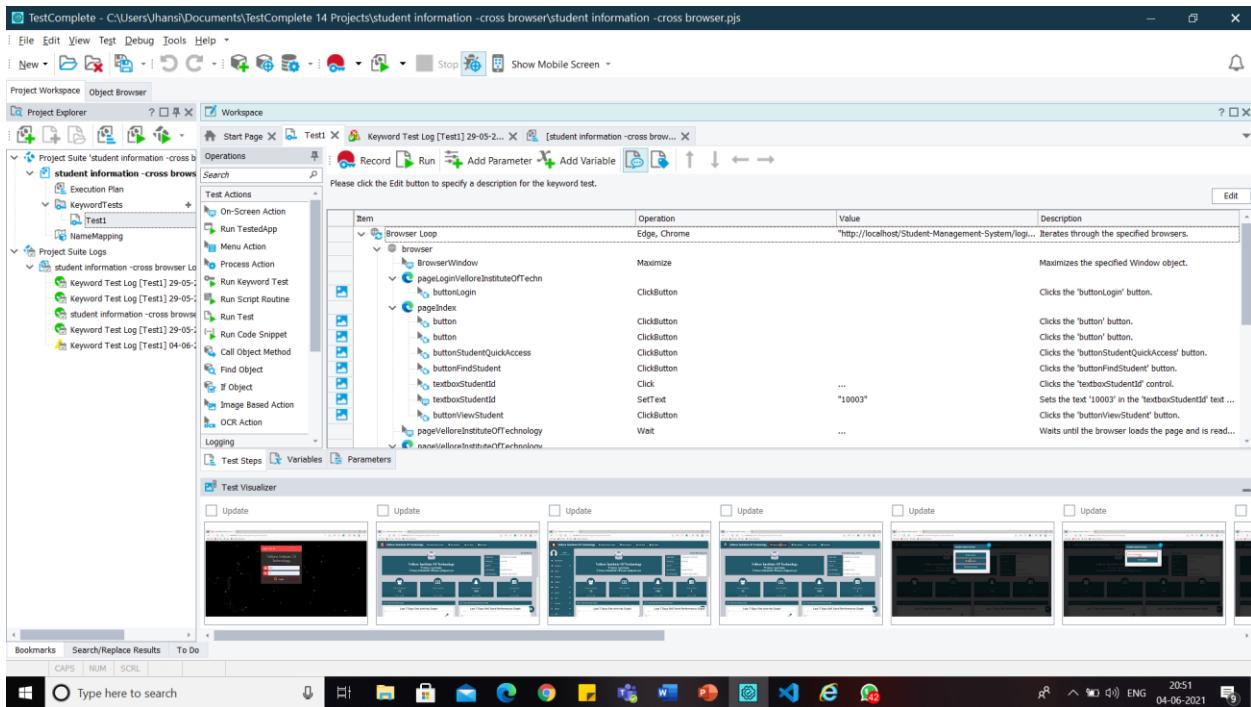
If we select all the browser available on the system. It automatically performs all the testings and we get the results that is the cross browser testing.

#### **Test Complete:**

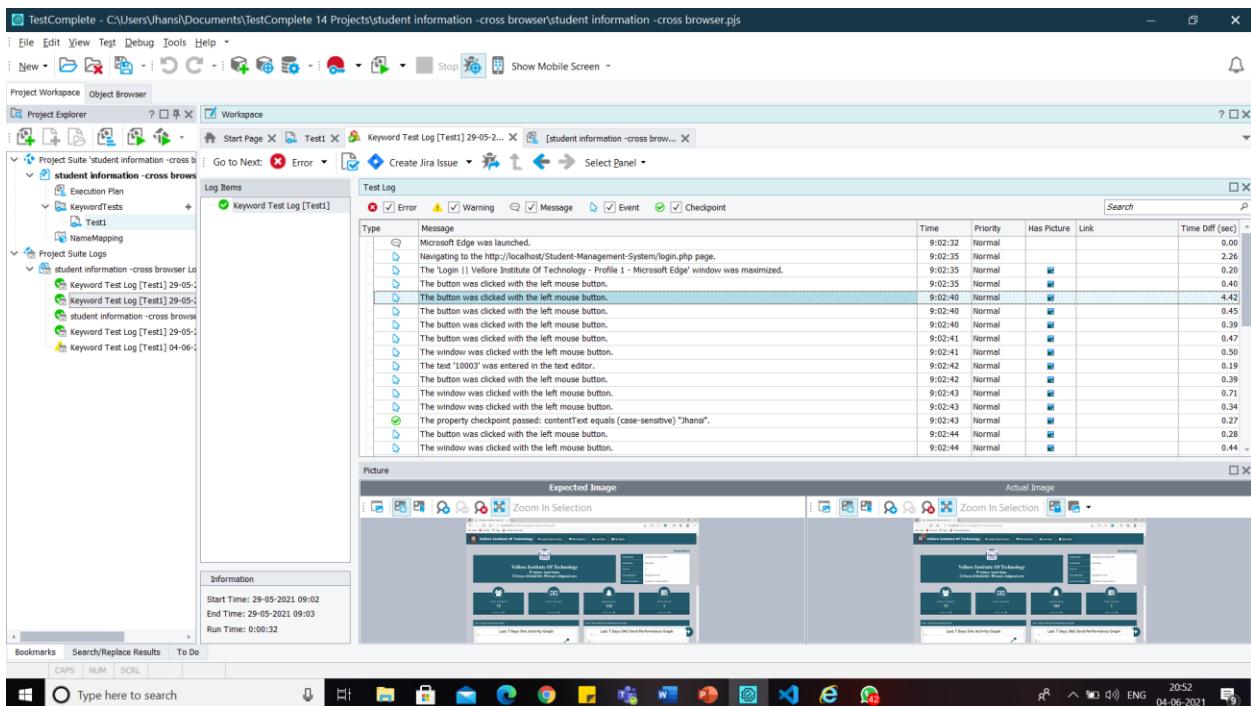
To verify each module with Cross Browser Methodology, we need to create a browser loop and select the browser types to verify the testing with cross browser.

## 1. Cross browser Testing for Student information Module:

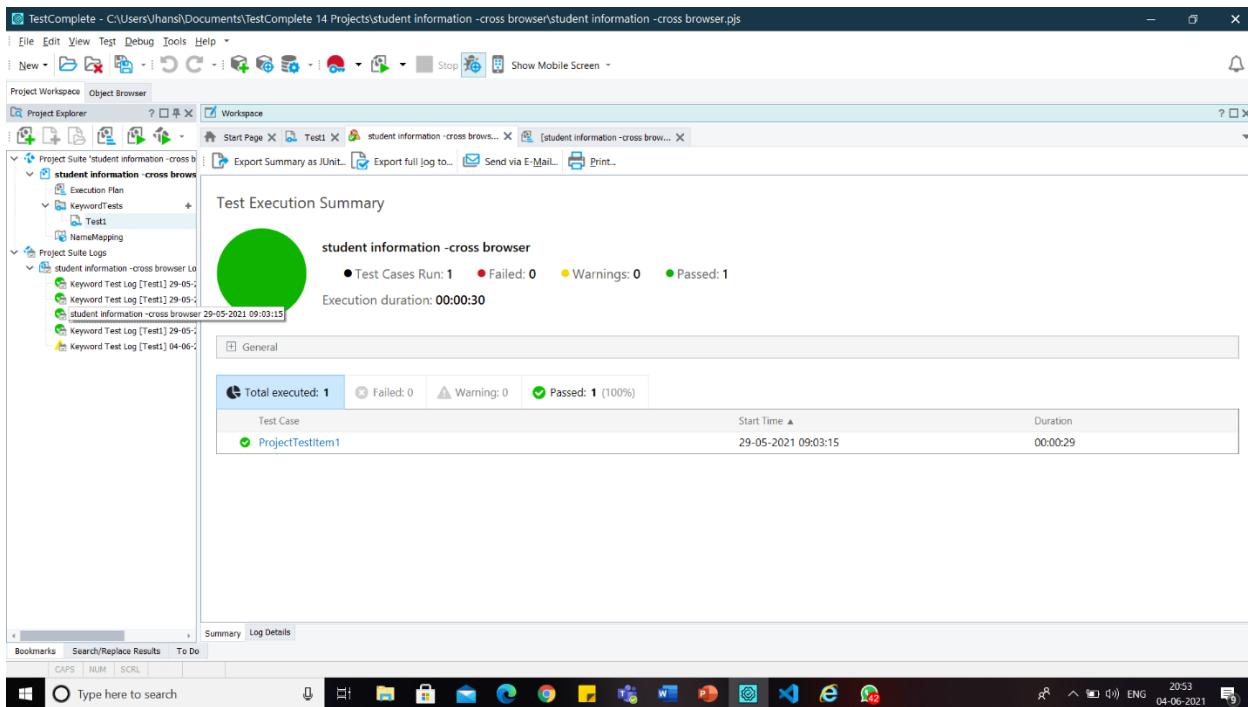
### Test Case Results:



### Test Success for Student information Module:

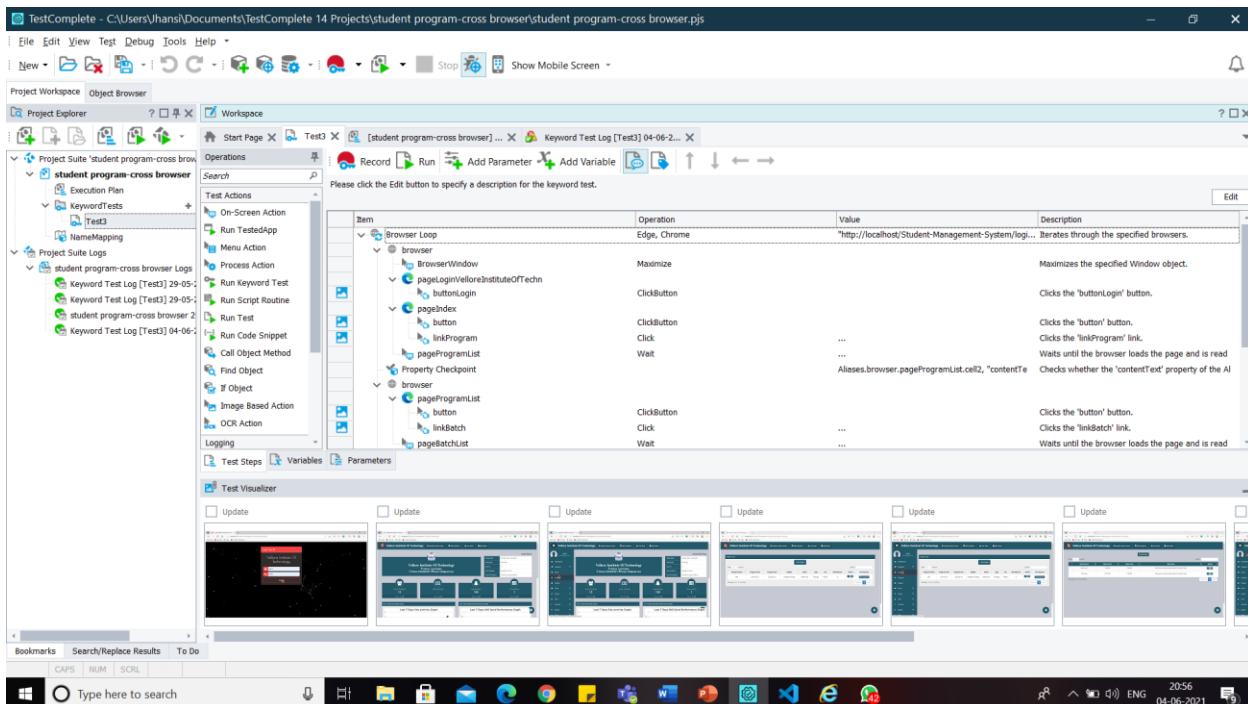


## Test Statistics:



## 2. Cross Browser Testing for Admit Student in multiple program Module:

### Test Case Results:



## Test Success for Admit Student in multiple program Module:

The screenshot shows the TestComplete interface with the following details:

- Project Explorer:** Shows a project named "student program-cross browser" containing "KeywordTests" and "NameMapping".
- Workspace:** Displays a "Test Log" titled "Keyword Test Log [Test3] 29-05-2...".
- Log Items:** A table of log entries with columns: Type, Message, Time, Priority, Has Picture, Link, and Time Diff (sec). The log contains 20 entries related to navigating and interacting with a browser window.
- Picture:** Two screenshots labeled "Expected Image" and "Actual Image" showing a browser window with a login form.
- Information:** Shows start time (29-05-2021 09:35), end time (29-05-2021 09:35), and run time (0:00:29).
- Taskbar:** Shows various application icons and the date/time (04-06-2021 20:57).

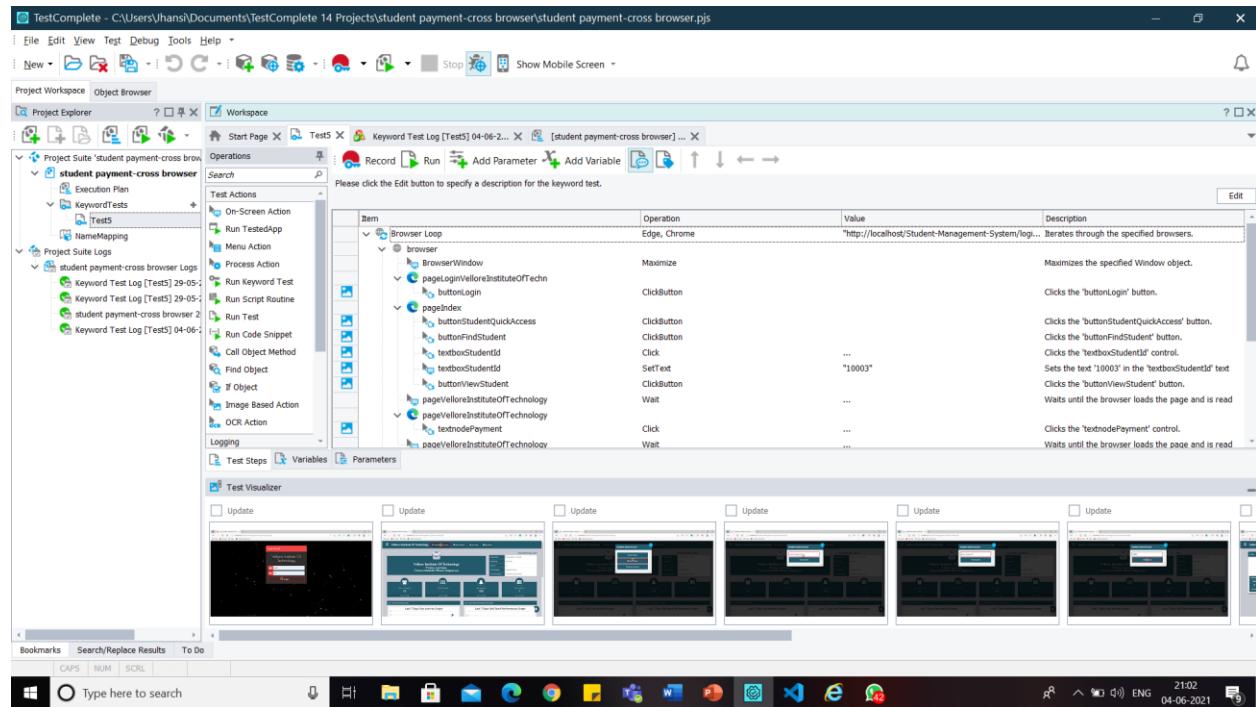
## Test Statistics:

The screenshot shows the TestComplete interface with the following details:

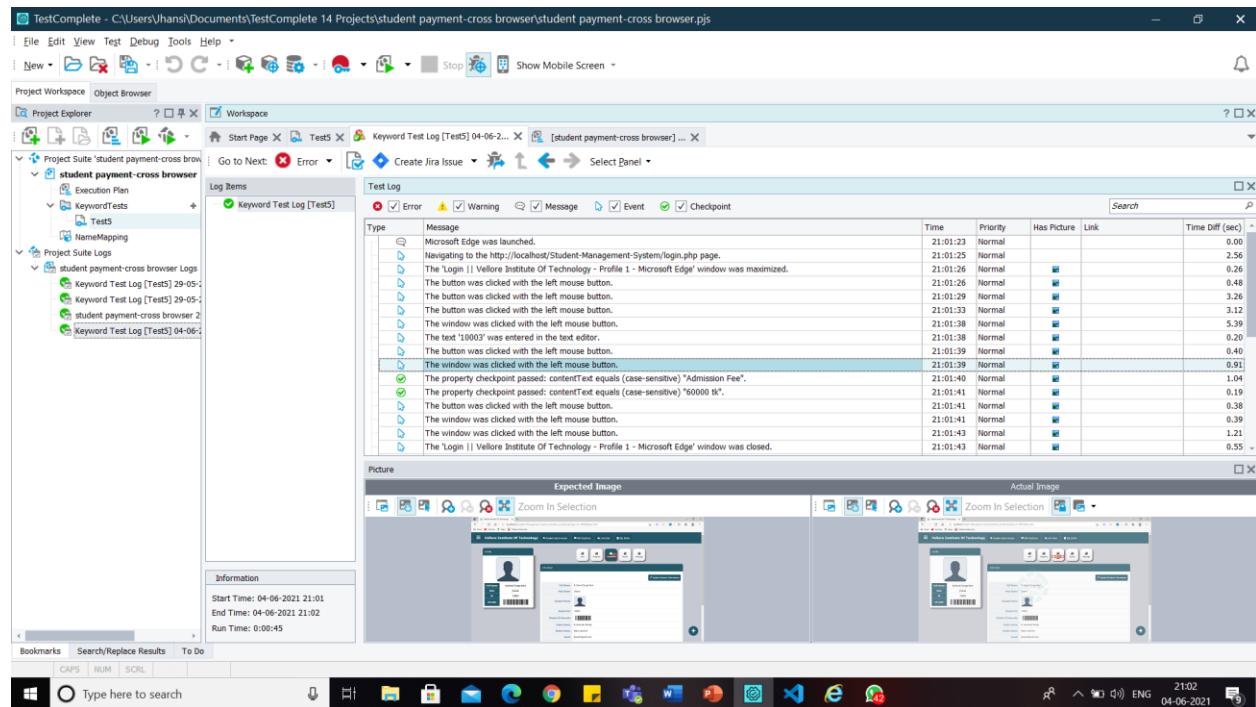
- Project Explorer:** Shows a project named "student program-cross browser" containing "Execution Plan", "KeywordTests", and "NameMapping".
- Workspace:** Displays a "Test Execution Summary" for "student program-cross browser".
- Summary:** A green circle indicates 100% success. Statistics: Total executed: 1, Failed: 0, Warnings: 0, Passed: 1 (100%). Execution duration: 00:00:35.
- Table:** A table showing test case details: ProjectTestItem, Start Time (29-05-2021 09:36:41), and Duration (00:00:34).
- Taskbar:** Shows various application icons and the date/time (04-06-2021 20:57).

### 3. Cross Browser Testing for Student Payment System Module:

#### Test Case Results:



#### Test Case Success for Student Payment System Module:



## Test Statistics:

The screenshot shows the TestComplete interface with the following details:

- Project Explorer:** Shows a project named "student payment-cross browser" with sub-folders like "Execution Plan", "KeywordTests", and "Test5".
- Test Execution Summary:** A large green circle indicates success. Summary: Test Cases Run: 1, Failed: 0, Warnings: 0, Passed: 1. Execution duration: 00:00:42.
- Test Case Details:** A table shows one test case: "ProjectTestItem" (Passed: 100%, Duration: 00:00:41).
- Bottom Bar:** Includes standard Windows icons for search, taskbar, and system status.

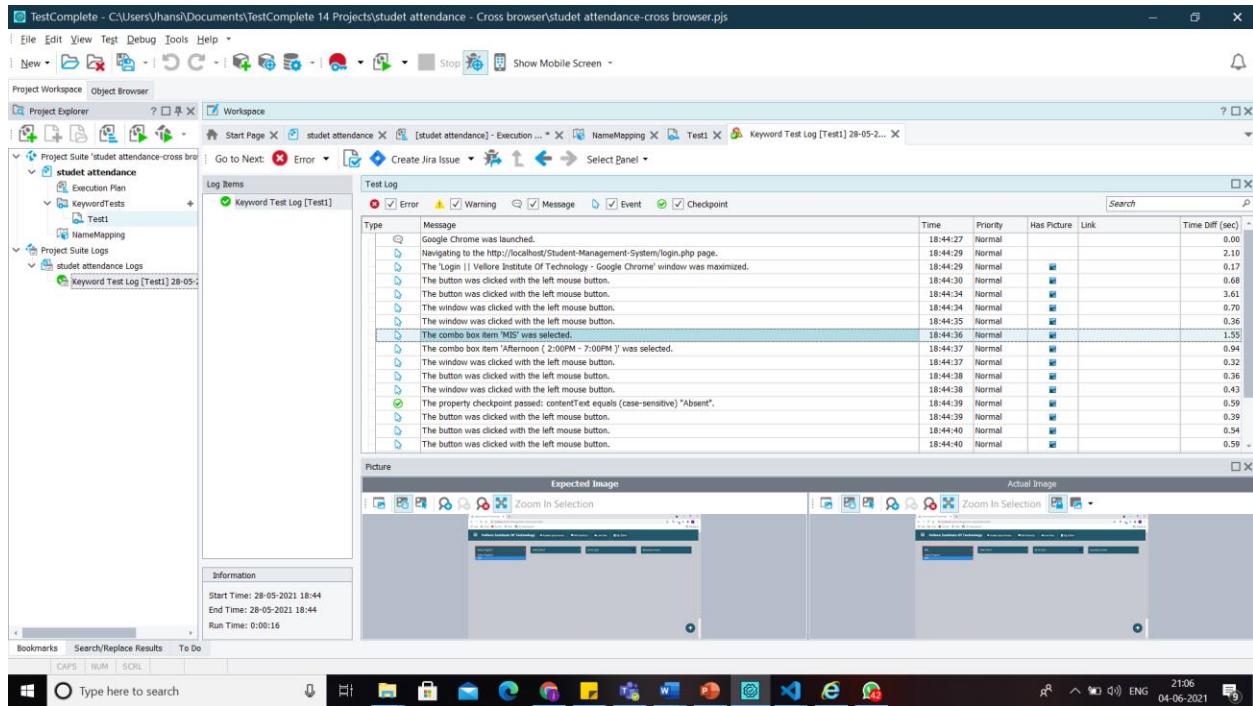
## 4. Cross Browser Testing for Student Attendance System Module:

### Test Case Results:

The screenshot shows the TestComplete interface with the following details:

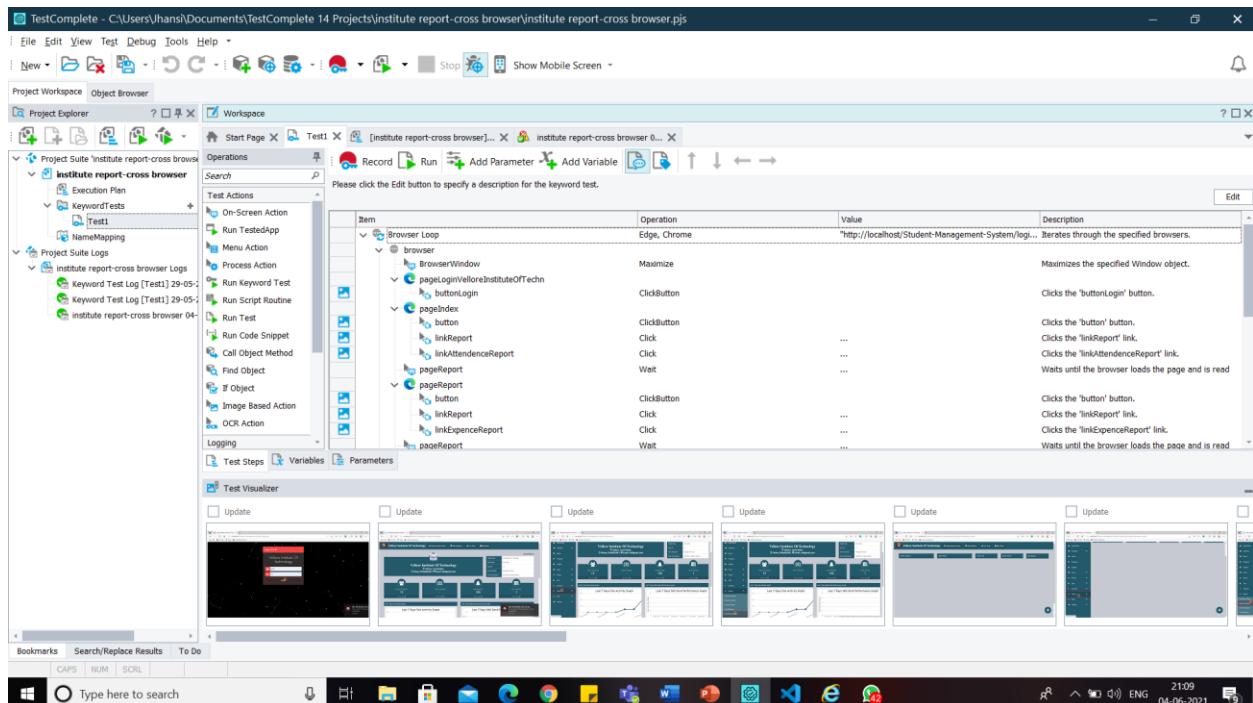
- Project Explorer:** Shows a project named "student attendance" with sub-folders like "Execution Plan", "KeywordTests", and "Test1".
- Operations View:** Displays a list of test steps:
  - On-Screen Action: Run TestedApp, Maximize browser window.
  - Process Action: Run Keyword Test, Click buttonLogin.
  - Run Script Routine: Run Test, Click button.
  - Find Object: If Object, Click linkStudents.
  - Image Based Action: Click linkAttendance.
  - OCR Action: Wait until pageAttends.
  - Test Step: pageAttend, Select dropdowns for program and session, Click buttonAttendancePanel.
- Test Visualizer:** Shows a series of screenshots illustrating the execution of the test steps.
- Bottom Bar:** Includes standard Windows icons for search, taskbar, and system status.

### Test Success for Student Attendance System Module:

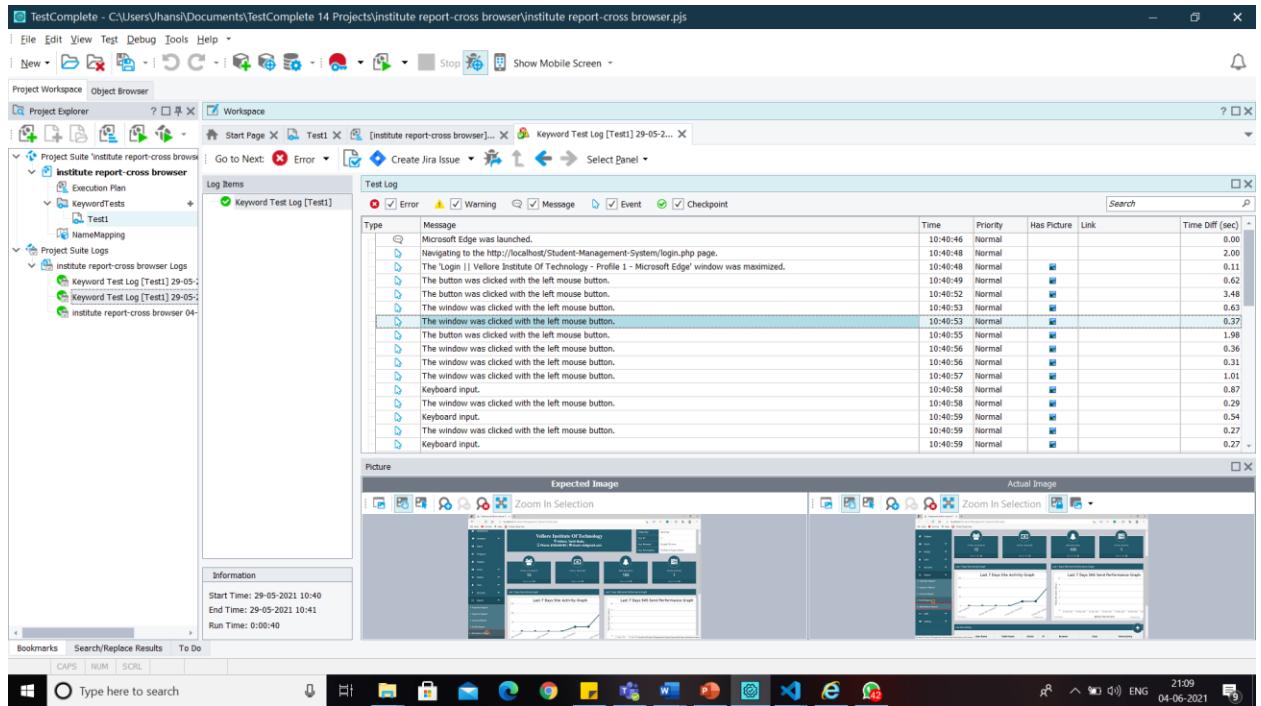


## 5. Cross Browser Testing for Institute Report Module:

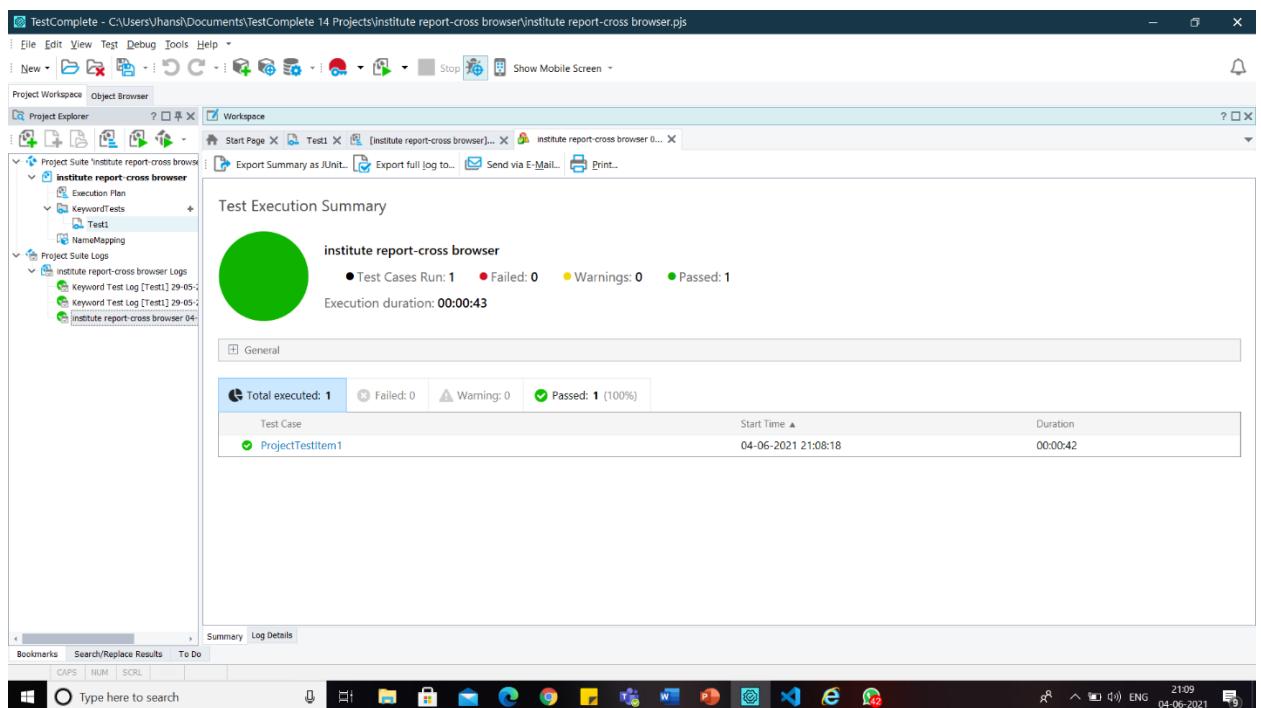
### Test Case Results:



### Test Success for Institute Report Module:



## Test Statistics:



- **Key Word Testing:**

1. Stores the test data like example username, email, password.
2. Access external test data.(excel.csv,databases)
3. Exchange data.
4. Helper tasks(creating loops)
5. It performs the keyword tests, scripts, projects, project suites.

### **Test Complete:**

In this we go to the keyword tests in the keyword tests we will be having like variables moving to that we will be having the temporary variables(stores the default values in the different variable types) and persistent variables(it stores the values which are being performed

during the testing being done. In the test complete it offers the data generator.

We can create new variable in the item with type of the variable this stores the value in the encrypted form.

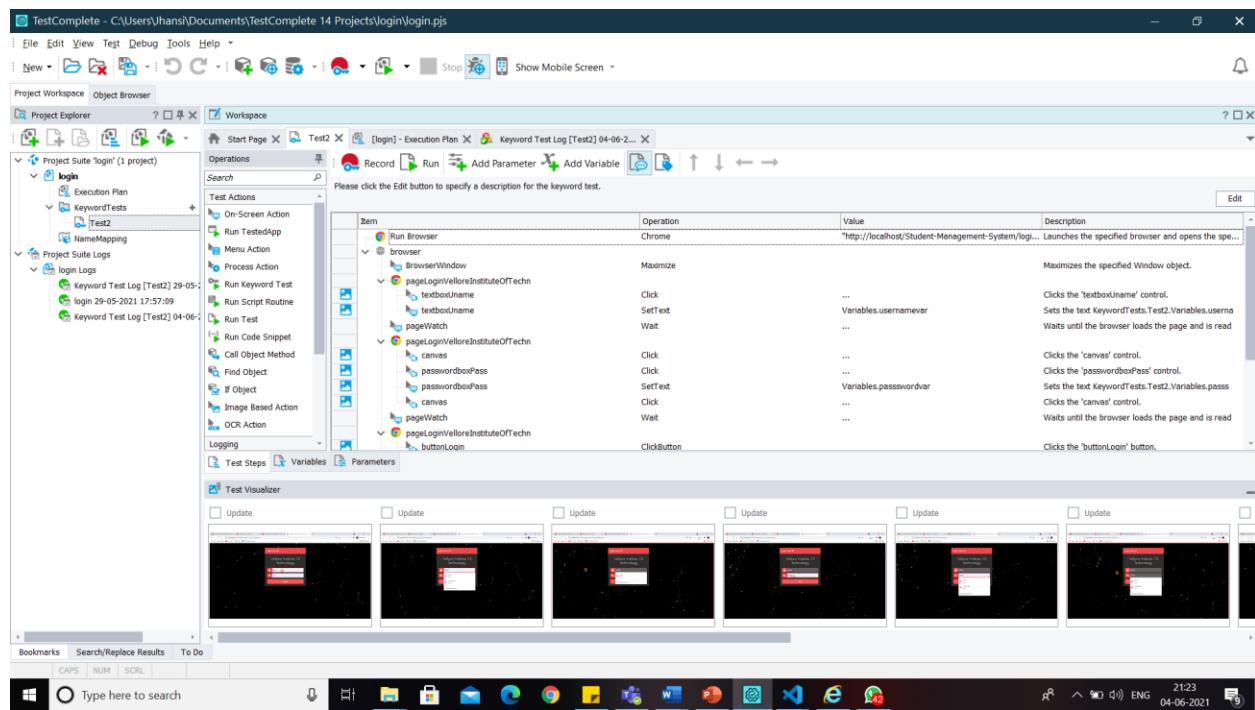
If add the variables in the test complete tool based on that it performs the tests.

We can also create the script variables in this test complete.

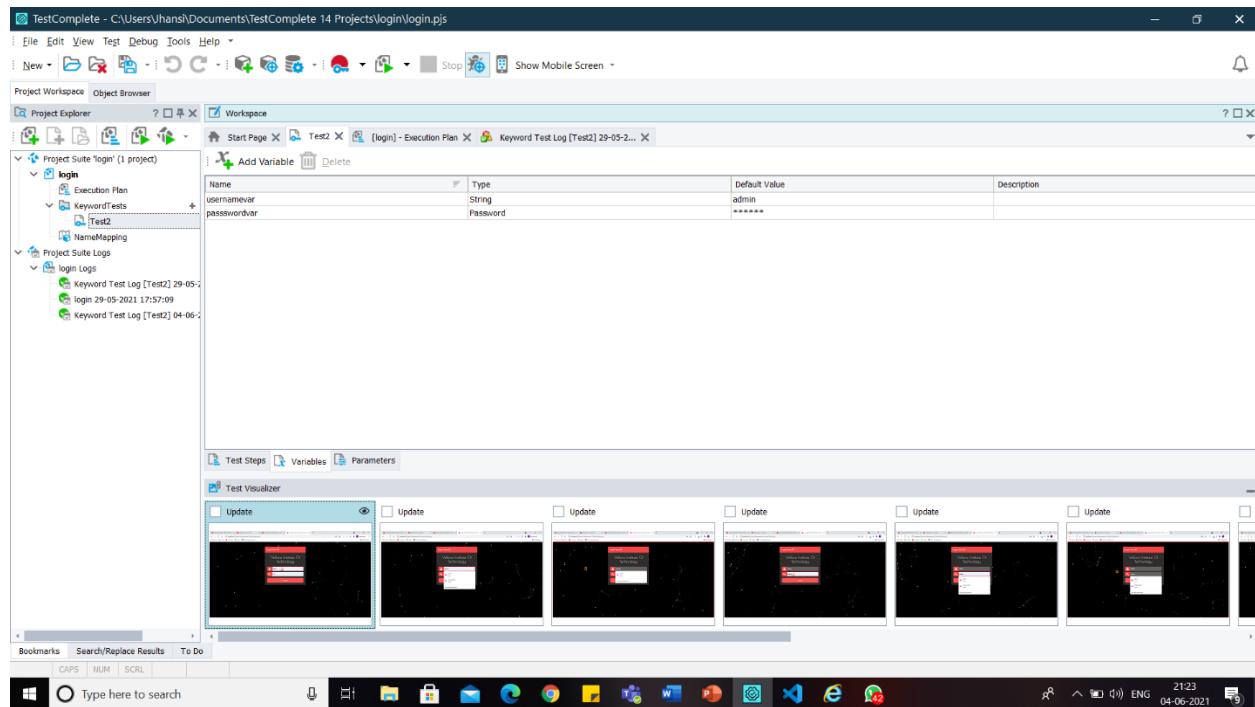
### **For login Module:**

We created two variables named username and password, which verifies with login details.

### **Test Case Results:**



## Variables created:



## Test Success:

TestComplete - C:\Users\Jhansi\Documents\TestComplete 14 Projects\login\login.pjs

File Edit View Test Debug Tools Help

New Stop Show Mobile Screen

Project Explorer Object Browser

Project Suite "login" (1 project)

- login
  - Execution Plan
  - KeywordTests
  - Ted2
  - NameMapping
- Project Suite Logs
- login Logs
  - Keyword Test Log [Test2] 29-05-2021 17:57:09
  - Keyword Test Log [Test2] 04-06-2021

Start Page X Test2 X [login] - Execution Plan X Keyword Test Log [Test2] 29-05-2021 17:57:09 X

Go to Next: Error Create Jira Issue Select Panel

Log Items

Test Log

Keyword Test Log [Test2]

Type Message Time Priority Has Picture Link Time Diff (sec)

Google Chrome was launched.	17:23:57	Normal		0.00
Navigating to the http://localhost/Student-Management-System/login.php page.	17:23:59	Normal		0.32
The 'Login    Vellore Institute Of Technology - Google Chrome' window was maximized.	17:24:00	Normal		0.17
The window was clicked with the left mouse button.	17:24:00	Normal		0.63
The text 'admin' was entered in the text editor.	17:24:01	Normal		0.02
The window was clicked with the left mouse button.	17:24:42	Normal		41.25
The window was clicked with the left mouse button.	17:24:42	Normal		0.39
The value of the login.login.KeywordTests.Ted2.passwordvar variable was entered in the text editor.	17:24:43	Normal		0.19
The window was clicked with the left mouse button.	17:24:43	Normal		0.44
The button was clicked with the left mouse button.	17:25:24	Normal		41.34
The button was clicked with the left mouse button.	17:25:28	Normal		3.57
The window was clicked with the left mouse button.	17:25:30	Normal		0.45
The window was clicked with the left mouse button.	17:25:39	Normal		1.95
The 'Login    Vellore Institute Of Technology - Google Chrome' window was closed.	17:25:31	Normal		0.77

Picture

Expected Image Actual Image

Information

Start Time: 29-05-2021 17:23  
End Time: 29-05-2021 17:25  
Run Time: 0:01:37

Bookmarks Search/Replace Results To Do

CAPS NUM SCR 2125 ENG 04-06-2021

## Test Statistics:

TestComplete - C:\Users\Jhansi\Documents\TestComplete 14 Projects\login\login.pjs

File Edit View Test Debug Tools Help

New Stop Show Mobile Screen

Project Explorer Object Browser

Project Suite "login" (1 project)

- login
  - Execution Plan
  - KeywordTests
  - Ted2
  - NameMapping
- Project Suite Logs
- login Logs
  - Keyword Test Log [Test2] 29-05-2021 17:57:09
  - Keyword Test Log [Test2] 04-06-2021

Start Page X Test2 X [login] - Execution Plan X login 29-05-2021 17:57:09 X

Export Summary as JUnit Export full log to... Send via E-Mail Print...

Test Execution Summary

login

● Test Cases Run: 1 ● Failed: 0 ● Warnings: 0 ● Passed: 1

Execution duration: 00:00:32

General

Total executed: 1 Failed: 0 Warning: 0 Passed: 1 (100%)

Test Case	Start Time	Duration
ProjectTestItem1	29-05-2021 17:57:10	00:00:32

Summary Log Details

Bookmarks Search/Replace Results To Do

CAPS NUM SCR 2125 ENG 04-06-2021

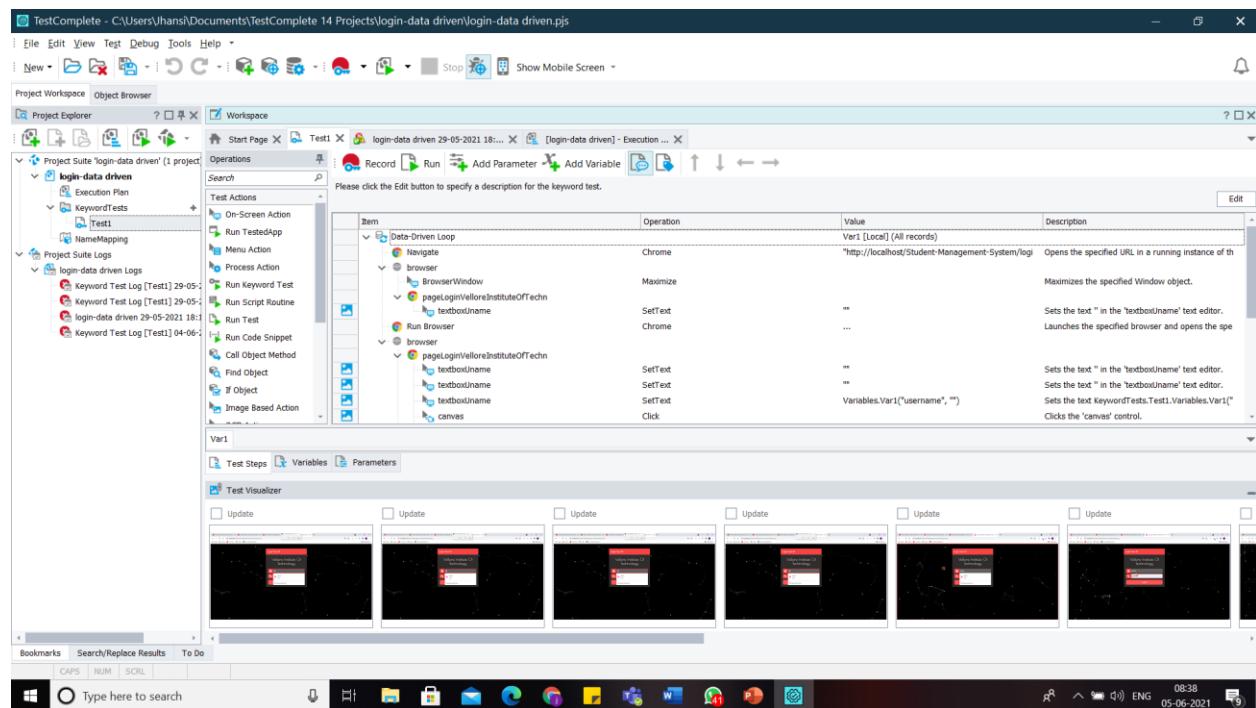
- **Data Driven Testing:**

Data-driven testing means running automated tests that simulate user actions over an application with different input data. This way you provide larger coverage and ensure that the application works properly with different data rather than with just one set of values.

### For Login Module:

We have created a data loop, and inserted variables named user name and password with string types to verify the login details. The data loop with details executes and tests for the login page.

### Test Case Results:



### Test case failure:

The screenshot shows the TestComplete interface with the 'Log items' panel open. The table displays the following log entries:

Type	Message	Time	Priority	Has Picture	Link	Time Diff (sec)
Message	Navigating to the http://localhost/Student-Management-System/login.php page.	18:13:57	Normal			0.00
Message	The 'Login    Vellore Institute Of Technology - Google Chrome' window was maximized.	18:13:57	Normal			0.10
Message	The text " was entered in the text editor.	18:13:57	Normal			0.25
Message	The browser is already running.	18:13:57	Normal			0.16
Message	The text " was entered in the text editor.	18:13:57	Normal			0.14
Message	The text " was entered in the text editor.	18:13:58	Normal			0.15
Exception	The test execution was interrupted.	18:13:58	Normal			0.23
Exception	The test execution was interrupted.	18:13:58	Normal			0.22

A message at the bottom states: 'A row index must be an integer value'.

## Test Statistics:

The screenshot shows the TestComplete interface with the 'Test Execution Summary' panel open. The summary indicates:

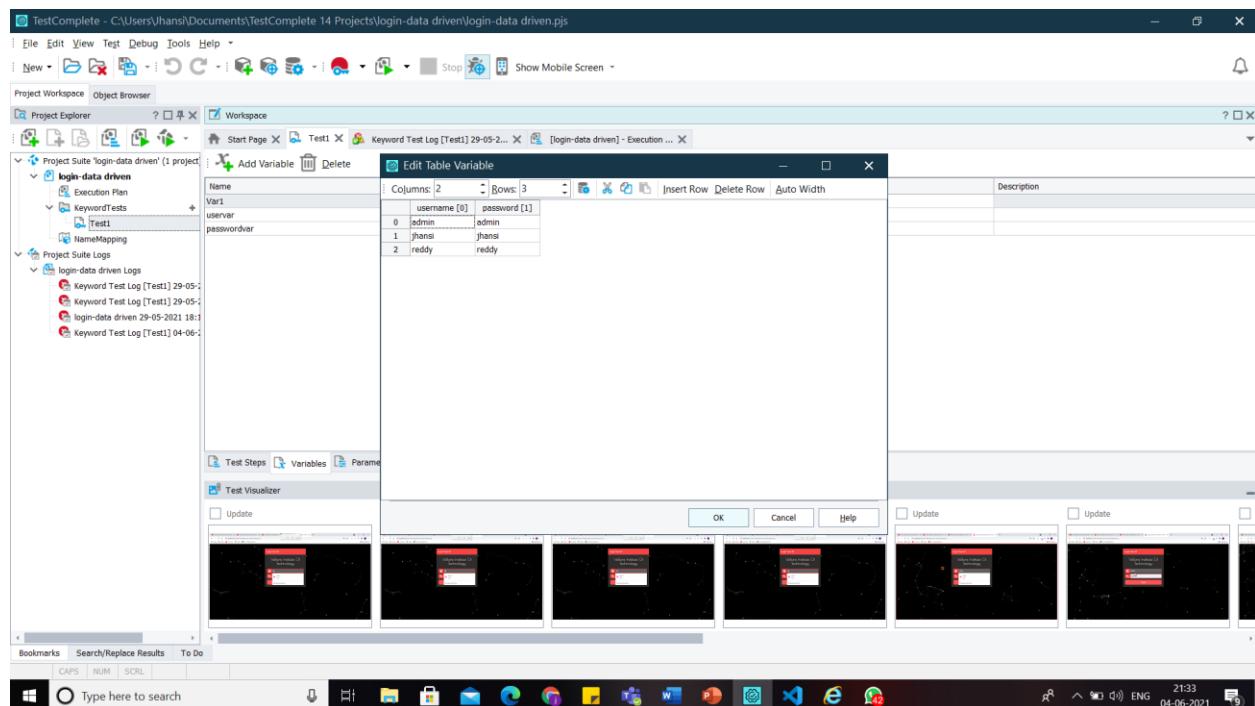
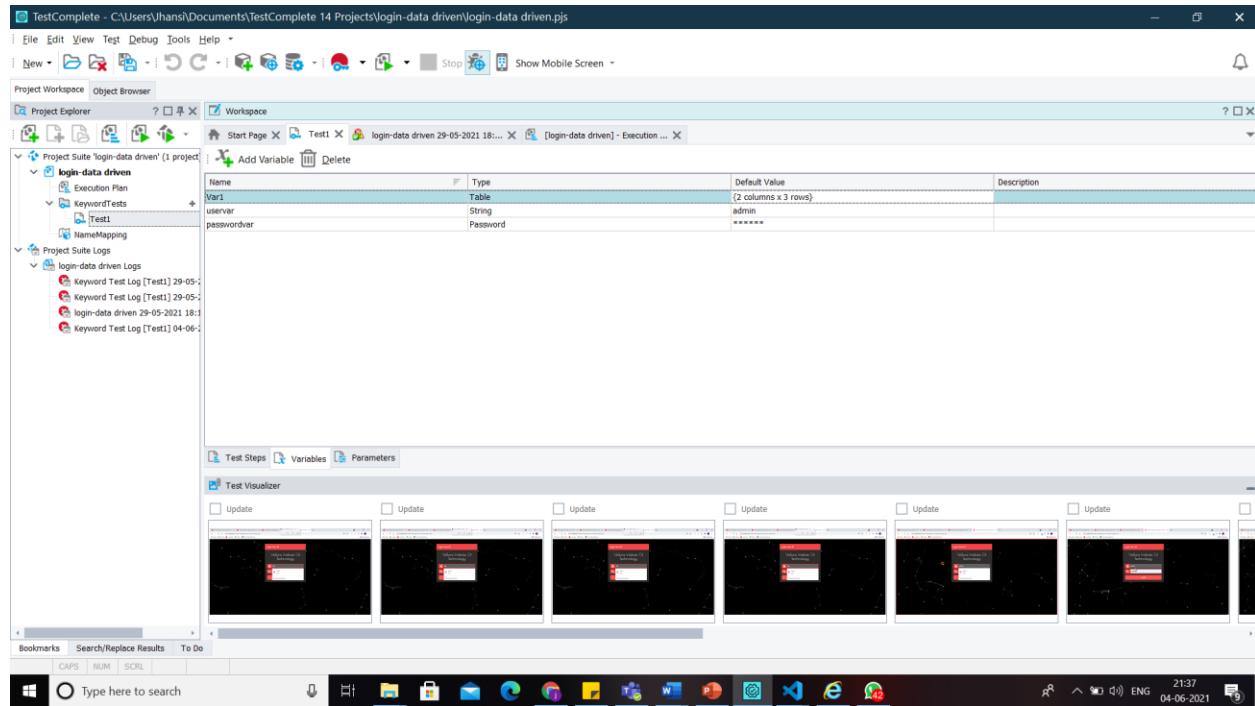
- Total test cases run: 1
- Failed: 1 (100%)
- Warnings: 0
- Passed: 0

Execution duration: 00:00:04

Test Case details:

Test Case	Start Time	Duration
ProjectTestItem1	29-05-2021 18:14:34	00:00:04

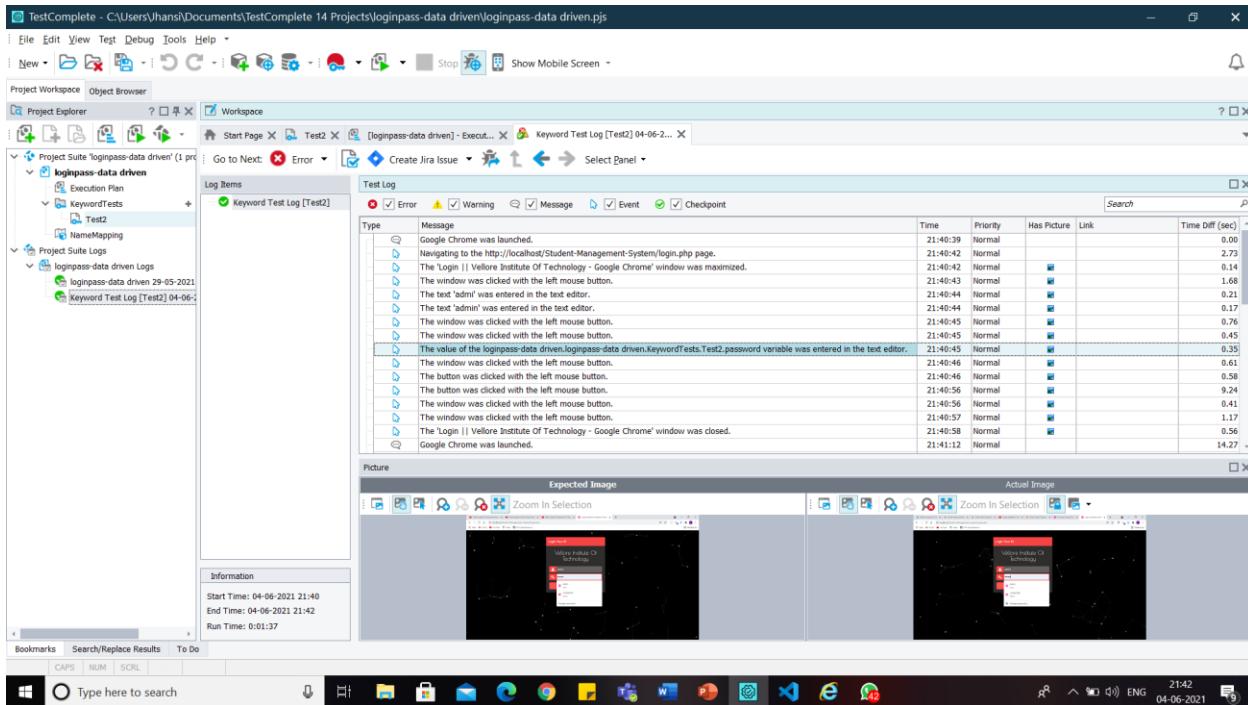
## Variables created:



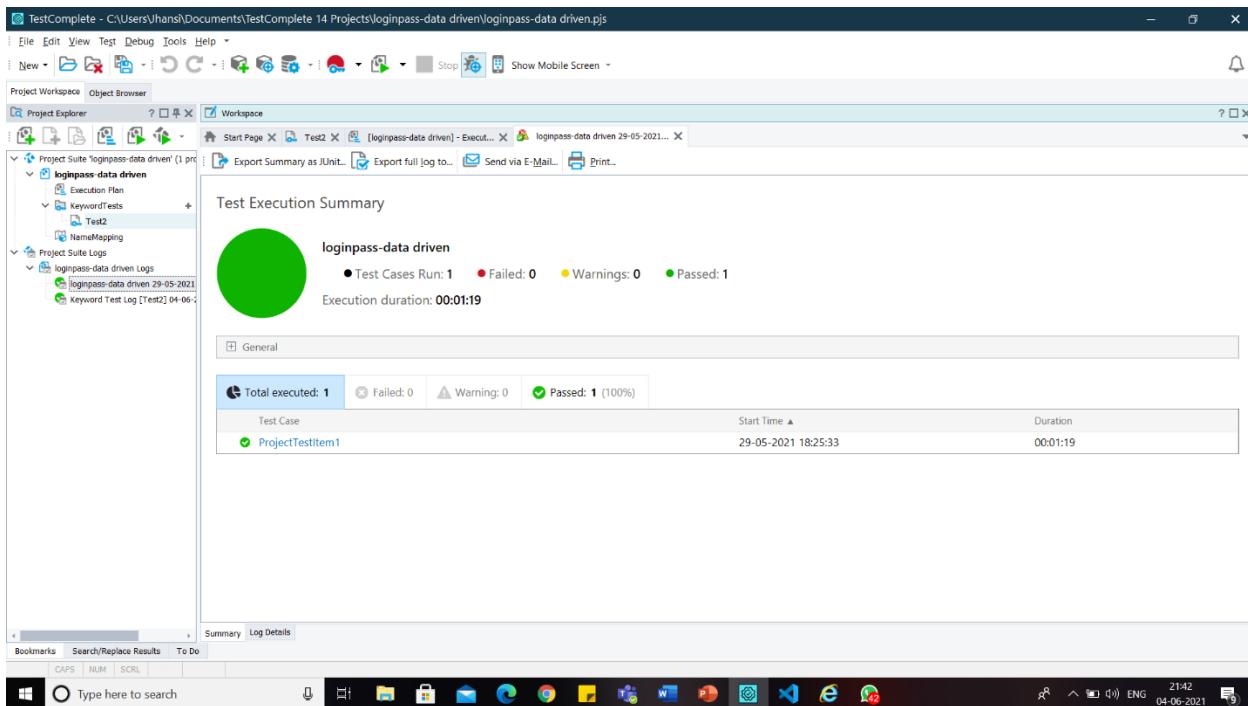
## Debugging:

As we have inserted wrong details for login information which are not stored in database the test cases are failed.

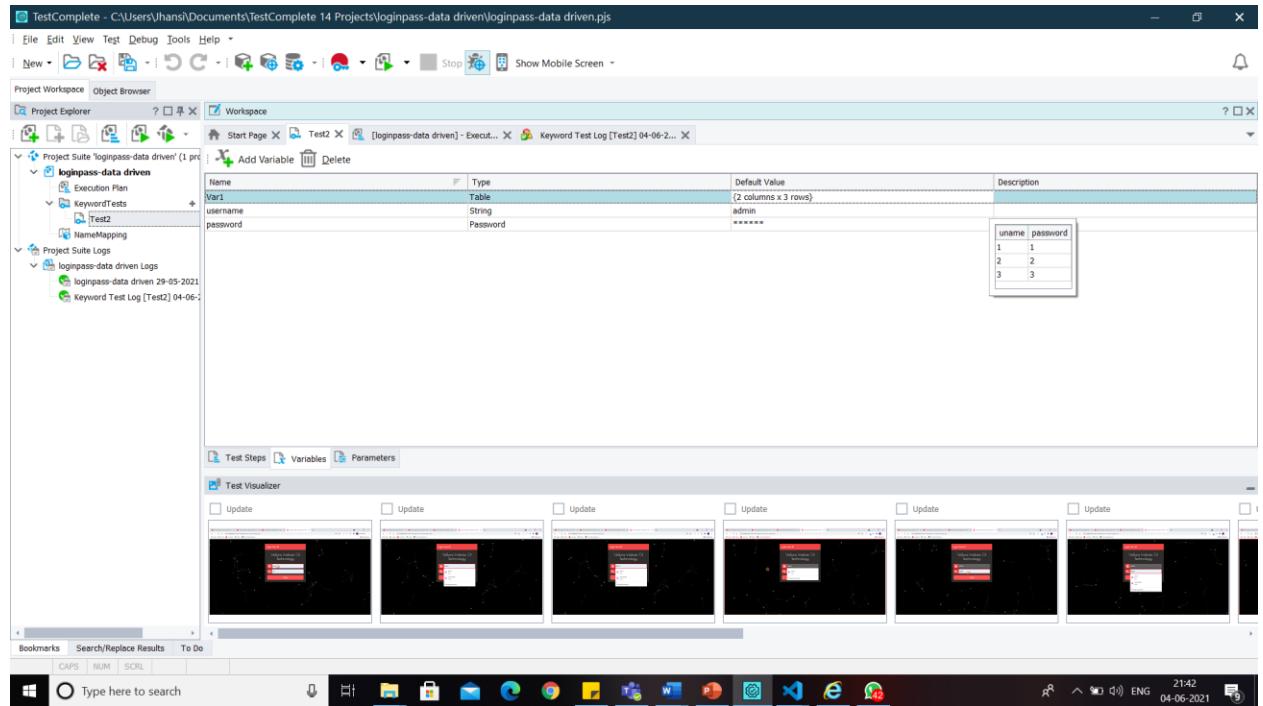
## Test Success:



## Test Statistics:



## Variables created:



- **Running Multiple tests:**

1. Group of tests in test complete.
2. Here in this we can perform many tests at a time.
3. Using the keyword tests we can add multiple tests at a time.
4. We can perform the particular tests which are needed.
5. By specifying what testing to be performed and what type of external tests to be linked we can link and run the tests.

### Test Case Results:

**TestComplete - C:\Users\Jhansi\Documents\TestComplete 14 Projects\multiple tests\multiple tests.pjs**

File Edit View Test Debug Tools Help

New Project Workspace Object Browser

Project Explorer Workspace

multiple tests (1 project)

- multiple tests
  - Execution Plan
  - KeywordTests
    - one
    - two
  - NameMapping
- multiple tests Logs
  - multiple tests 29-05-2021 12:59:20
  - Keyword Test Log [one] 04-06-20
  - multiple tests 04-06-2021 21:47:44
  - multiple tests 29-05-2021 13:02:34

Operations

Test Actions

Please click the Edit button to specify a description for the keyword test.

Item	Operation	Value	Description
pageStudentList	ClickButton	...	Clicks the 'button' button.
linkStudents	Click	...	Clicks the 'linkStudents' link.
linkBatch	Click	...	Clicks the 'linkBatch' link.
pageBatchList	Wait	...	Waits until the browser loads the page and is ready...
button	ClickButton	...	Clicks the 'button' button.
linkProgram	Click	...	Clicks the 'linkProgram' link.
pageProgramList	Wait	...	Waits until the browser loads the page and is ready...
programList	ClickButton	...	Clicks the 'button' button.
linkSubject	Click	...	Clicks the 'linkSubject' link.
pageSubjectList	Wait	...	Waits until the browser loads the page and is ready...
subjectList	ClickButton	...	Clicks the 'button' button.

Test Visualizer

Bookmarks Search/Replace Results To Do

CAPS NUM SCR 21:48 04-06-2021 ENG

**TestComplete - C:\Users\Jhansi\Documents\TestComplete 14 Projects\multiple tests\multiple tests.pjs**

File Edit View Test Debug Tools Help

New Project Workspace Object Browser

Project Explorer Workspace

multiple tests (1 project)

- multiple tests
  - Execution Plan
  - KeywordTests
    - one
    - two
  - NameMapping
- multiple tests Logs
  - multiple tests 29-05-2021 12:59:20
  - Keyword Test Log [one] 04-06-20
  - multiple tests 04-06-2021 21:47:44
  - multiple tests 29-05-2021 13:02:34

Operations

Test Actions

Please click the Edit button to specify a description for the keyword test.

Item	Operation	Value	Description
Run Browser	Chrome	...	Launches the specified browser and opens the spe...
browser	Maximize	...	Maximizes the specified Window object.
buttonLogin	Click	...	Clicks the 'buttonLogin' button.
pageLoginVonorInstituteOfTech	Click	...	Clicks the 'button1msStatistics' button.
button1msStatistics	Click	...	Clicks the 'button2' button.
pageIndex	Click	...	Clicks the 'buttonIndex' button.
button1	Click	...	Clicks the 'button1' button.
button2	Click	...	Clicks the 'button2' button.
button3	Click	...	Clicks the 'button3' button.
buttonLiveChat	Click	...	Clicks the 'buttonLiveChat' button.
button1LiveChat	Click	...	Clicks the 'button1LiveChat' button.
button2LiveChat	Click	...	Clicks the 'button2LiveChat' button.
button3LiveChat	Click	...	Clicks the 'button3LiveChat' button.
buttonMessage	SetText	"hello"	Sets the text 'hello' in the 'textboxMessage' text edi...
buttonSend	Click	...	Clicks the 'buttonSend' button.
button1Y	Click	...	Clicks the 'button1Y' button.

Test Visualizer

Bookmarks Search/Replace Results To Do

CAPS NUM SCR 21:48 04-06-2021 ENG

## Test Success statistics of multiple testing methods:

TestComplete - C:\Users\Jhans\Documents\TestComplete 14 Projects\multiple tests\multiple tests.pjs

File Edit View Test Debug Tools Help

New - Stop Show Mobile Screen

Project Workspace Object Browser

Project Explorer ? □ ×

multiple tests

- Execution Plan
- KeywordTests
  - one
  - two
- NameMapping

Project Suite Logs

- multiple tests 29-05-2021 12:59:0
- multiple tests 04-06-2021 21:47:4
- multiple tests 29-05-2021 13:02:34

multiple tests

- Test Cases Run: 2 Failed: 0 Warnings: 0 Passed: 2

Execution duration: 00:01:14

Total executed: 2 Failed: 0 Warning: 0 Passed: 2 (100%)

Test Case	Start Time	Duration
one	29-05-2021 13:02:34	00:00:31
two	29-05-2021 13:03:04	00:00:43

Summary Log Details

Bookmarks Search/Replace Results To Do

CAPS NUM SCRLL

Type here to search

21:49 ENG 04-06-2021

The screenshot shows the TestComplete 14 interface with a project named "multiple tests". The "Execution Summary" pane displays a green circle indicating success. It shows 2 test cases run, 0 failed, 0 warnings, and 2 passed (100%). The execution duration was 00:01:14. Below this, a detailed table lists the two test cases: "one" and "two", along with their start times (29-05-2021 13:02:34 and 29-05-2021 13:03:04) and execution durations (00:00:31 and 00:00:43). The bottom status bar shows the date (04-06-2021), time (21:49), and language (ENG).