
COMPLIANCE REPORT

FOR

ONLINE CLASS MANAGEMENT

SYSTEM

Prepared by:

- 1. Sumit Kumar (22CS30056)**
- 2. Aviral Singh (22CS30015)**
- 3. Sagnibha Roy (22CS10064)**

April 7, 2024

Contents

1	Introduction and Important Points	3
2	Compliance Report for Student access through the Interface	3
2.1	Login Page	3
2.2	Signup Page	3
2.3	Dashboard.....	3
2.4	Classes.....	3
2.5	Join Classes.....	4
2.6	Discussion or Chat.....	4
2.7	Submit Assignments.....	4
2.8	View Grades.....	4
2.9	View Calendar.....	5
2.10	Join Online Live Class.....	5
3	Compliance Report for Teacher access through the Frontend	5
3.1.	Login Page	5
3.2	Signup Page	5
3.3	Dashboard.....	5
3.4	Classes.....	5
3.5	Create a Class.....	6
3.6	Discussion or Chat.....	6
3.7	Create and Grade Assignments.....	6
3.8	View Attendees.....	6
3.9	Set Class TimeTable.....	6
3.10	Organise Online Live Class.....	7
4	Automated Testing	8
4.1.	Login Page	8
4.2	Signup Page	9
4.3	Create class.....	10
4.4	Join Class.....	11
5	Manual Testing	12
5.1.	Dashboard	12
5.2	Viewing Attendees	12
5.3	Discussion or chat.....	13
5.4	Create Assignments.....	14
5.5	Submit Assignments	15
5.6	Grade Assignments.....	16
5.7	View Grades.....	17
5.8	Create a Live class.....	17
5.9	Joining Live Class.....	18
5.10	Set Class Timings.....	18
5.11	View Class Calendar.....	19
5.12	Selenium IDE Testing.....	20

Introduction and Important Points

This is the Compliance Report document for Online Classroom Management System. Here, we list down the test cases that were mentioned and planned out in the test suite along with the PASS/FAIL results that were obtained after testing the software. The exact test cases and expected outputs can be found in the test suite document.

Note: If the result is as expected for a test case, it is marked as **PASSED**, else it is marked as **FAILED**.

2 Compliance Report for the Frontend Interface

2.1 Login Page

1. Both Username and Password are correct

PASSED

2. Username is correct but Password is incorrect

PASSED

3. Username is incorrect but Password is correct

PASSED

4. Both Username and Password are incorrect

PASSED

2.2 Signup Page

Working Properly

2.3 Dashboard

1. All buttons can be clicked

PASSED

2.4 Classes

1. Buttons to join class can be clicked

PASSED

2.5 Join Classes

1. Entering class code to join class

PASSED

2. Opening one class out of many classes joined

PASSED

2.6 Discussion or Chat

1. Student can send text through chat which is visible to all

PASSED

2. Student can send Files of any type through chat which is visible to all

PASSED

2.7 Submit Assignments

1. Assignment in the form of pdf can be submitted by the student

PASSED

2. Assignment pdf can be downloaded

PASSED

2.8 View Grades

1. Students can view Grades received in their Assignments

PASSED

2.9 View Calendar

1. Student can see his/her time table of online classes

PASSED

2.10 Join Live Online Classes

1. *Students can Join online Live Classes by clicking on the video link send by the teacher in the discussion box*

PASSED

3 Compliance Report for Admin access through the Frontend

3.1 Login

1. *Can login with correct credentials*

PASSED

2. *Cannot log in with correct ID but wrong password*

PASSED

3. *Cannot log in with wrong ID but correct password*

PASSED

4. *Cannot log in with wrong ID and wrong password*

PASSED

3.2 Signup Page

Working Properly

3.3 Dashboard

1. All buttons can be clicked

PASSED

3.4 Classes

1. *Buttons to create and join class can be clicked*

PASSED

3.5 Create a Class

1. Able to create a class by entering all the class credentials

PASSED

2. Able to open the class created by clicking on its button

PASSED

3.6 Discussion

1. Teacher can send text through chat which is visible to all

PASSED

2. Teacher can send Files of any type through chat which is visible to all

PASSED

3.7 Create And Grade Assignments

1. Teacher can set grades for the student

PASSED

2. Teacher can Download the file submitted by the student

PASSED

3. Teacher can create an assignment

PASSED

3.8 View Attendees

1. Teachers can view all the students enrolled in the class by clicking on the attendees.

PASSED

3.9 Set TimeTable

1. Teachers can set a time table for the class .

PASSED

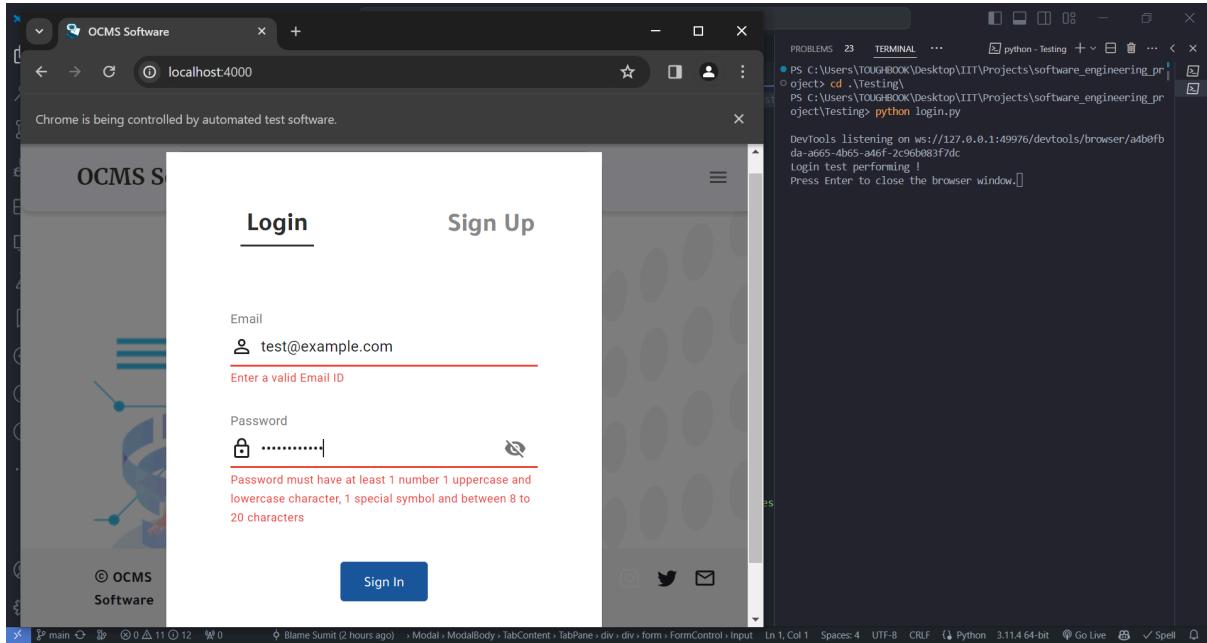
3.10 Organize Live Online Classes

1. *Teachers can start a live class and share its link.*

PASSED

AUTOMATED TESTING

LOGIN PAGE-



CODE FOR AUTOMATED TESTING OF LOGIN PAGE-

```
import os
from selenium import webdriver
from selenium.webdriver.chrome.service import Service
from selenium.webdriver.support.ui import WebDriverWait
from selenium.webdriver.support import expected_conditions as EC

# Start the Chrome WebDriver
driver = webdriver.Chrome()

# Open the web page
driver.get("http://localhost:4000")

# Wait for the page to load
driver.implicitly_wait(10)

# Find and click on the login button
login_button = driver.find_element('id','loginTesting')
login_button.click()

# Find email and password fields
email_field = driver.find_element('id','email')
password_field = driver.find_element('id','password')

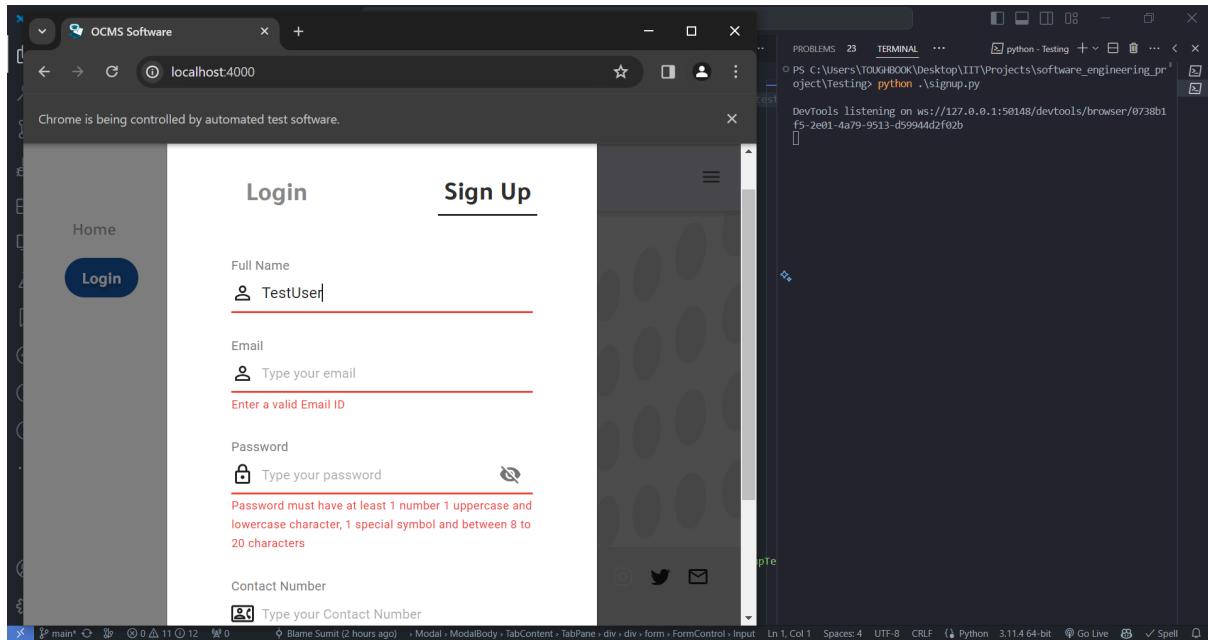
# Fill in email and password fields
email_field.send_keys("test@example.com")
password_field.send_keys("testpassword")

WebDriverWait(driver, 100).until(EC.element_to_be_clickable(('id', 'loginTesting')))

print("Login test performing !")

input("Press Enter to close the browser window.")
```

SIGNUP PAGE -



CODE FOR AUTOMATED TESTING OF SIGNUP PAGE-

```
import os
from selenium import webdriver
from selenium.webdriver.chrome.service import Service
from selenium.webdriver.support.ui import WebDriverWait
from selenium.webdriver.support import expected_conditions as EC

# Start the Chrome WebDriver
driver = webdriver.Chrome()

# Open the web page
driver.get("http://localhost:4000")

# Wait for the page to load
driver.implicitly_wait(10)

# Click on the signup tab
signup_tab = driver.find_element("id", "signupTesting")
signup_tab.click()

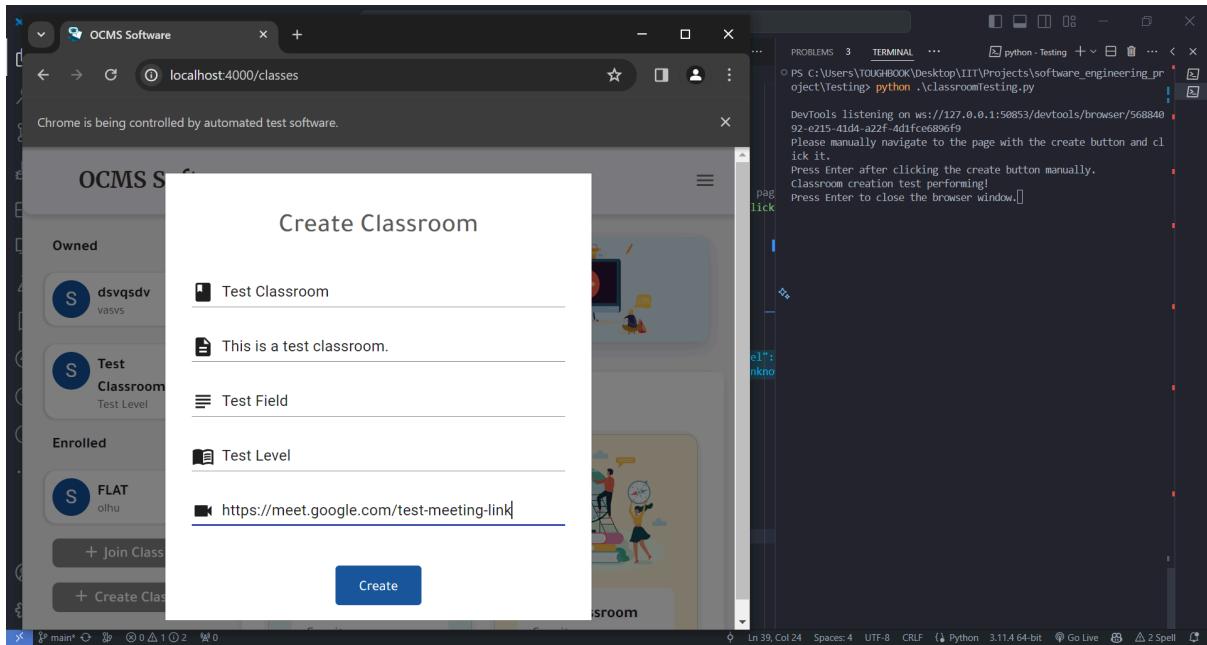
# Find input fields
name_field = driver.find_element("id", "name")
email_field = driver.find_element("id", "email")
password_field = driver.find_element("id", "password")
contact_field = driver.find_element("id", "contact")

# Fill in signup form fields
name_field.send_keys("TestUser")
email_field.send_keys("test@gmail.com")
password_field.send_keys("AsDr1234!")
contact_field.send_keys("1234567890")

WebDriverWait(driver, 100).until(EC.element_to_be_clickable(("id", 'signupTesting')))

print("Signup test performing!")
input("Press Enter to close the browser window.")
```

CREATE CLASSROOM-



CODE FOR AUTOMATED TESTING OF CREATE CLASSROOM-

```
import os
from selenium import webdriver
from selenium.webdriver.chrome.service import Service
from selenium.webdriver.support.ui import WebDriverWait
from selenium.webdriver.support import expected_conditions as EC

# Start the Chrome WebDriver
driver = webdriver.Chrome()

# try:
# Open the web page
driver.get("http://localhost:4000")

# Wait for the page to load
# driver.implicitly_wait(1000)
# Print a message prompting the user to manually navigate to the desired page
print("Please manually navigate to the page with the create button and click it.")

# Wait for the user to confirm that they have clicked the button
input("Press Enter after clicking the create button manually.")

# Find classroom creation form fields
class_name_field = driver.find_element('id', 'classname')
description_field = driver.find_element('id', 'description')
field_name_field = driver.find_element('id', 'field')
class_level_field = driver.find_element('id', 'classlevel')
meet_link_field = driver.find_element('id', 'meetlink')

# Fill in classroom creation form fields
class_name_field.send_keys("Test Classroom")
description_field.send_keys("This is a test classroom.")
field_name_field.send_keys("Test Field")
class_level_field.send_keys("Test Level")
meet_link_field.send_keys("https://meet.google.com/test-meeting-link")

# Submit the form
create_button = driver.find_element('id', 'createClassroomButton')
# create_button.click()

print("Classroom creation test performing!")

input ("Press Enter to close the browser window.")
```

```

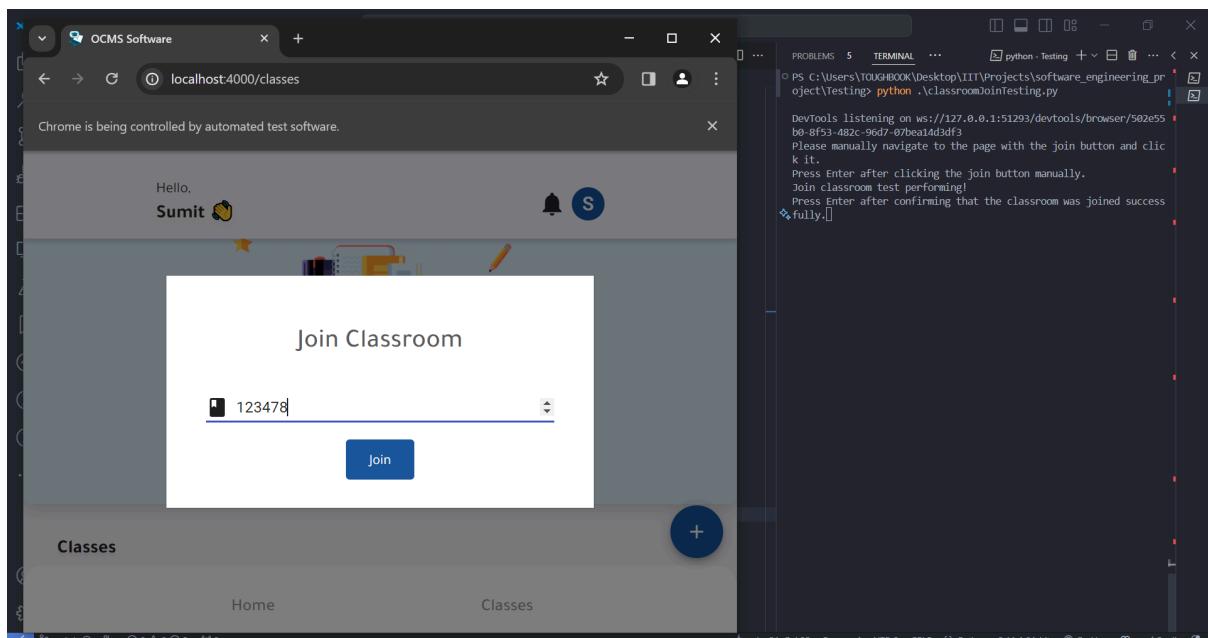
driver.quit()
# You can add assertions here to verify if the classroom was successfully created

# except Exception as e:
#     print("An error occurred during testing:", e)

# finally:
#     # Close the browser session
#     driver.quit()

```

JOIN CLASSROOM-



CODE FOR AUTOMATED TESTING OF JOIN CLASS FEATURE-

```

import os
from selenium import webdriver
from selenium.webdriver.chrome.service import Service
from selenium.webdriver.support.ui import WebDriverWait
from selenium.webdriver.support import expected_conditions as EC

# Start the Chrome WebDriver
driver = webdriver.Chrome()

# Open the web page
driver.get("http://localhost:4000")

# Print a message prompting the user to manually navigate to the page with the join button
print("Please manually navigate to the page with the join button and click it.")

# Wait for the user to confirm that they have clicked the join button
input("Press Enter after clicking the join button manually.")

# Find the class code input field
class_code_field = driver.find_element('id', 'classCode')

# Fill in the class code input field
class_code_field.send_keys("123478") # Assuming the class code is "123456"

# Find and click the join button
join_button = driver.find_element('id', 'JoinClassButton')
# join_button.click()

print("Join classroom test performing!")

```

```
input("Press Enter after confirming that the classroom was joined successfully.")

# You can add assertions here to verify if the classroom was successfully joined

# Close the browser session
driver.quit()
```

MANUAL TESTING

Dashboard- passed

The screenshot shows the OCMS Software dashboard. On the left, there are sections for 'Owned' (Chemistry) and 'Enrolled' (empty). In the center, a large box says 'Hello, Sagnibha Roy!' with the subtext 'Learn anytime, anywhere. Boost your skills flexible, online courses'. Below this is a 'Classes' section with a card for 'Chemistry' by Sagnibha Roy. At the bottom, the taskbar shows various icons and the date 08-04-2024.

Viewing Attendees- passed

The screenshot shows the attendees page for the 'Chemistry' class. It lists 'Admin' (Sagnibha Roy) and 'Members' (Aviral Singh). A reminder box says 'No work due!'. At the bottom, the taskbar shows various icons and the date 08-04-2024.

Discussion or Chat- passed

The screenshot shows the OCMS Software interface for a class named "Chemistry". The top navigation bar includes "Home", "Classes", and a user profile icon. The main area displays a list of messages in a chat-like format:

- Sagnibha Roy | 8 | chemistry
Class Code - 123485
Set Class Timings
- S Start a discussion, share class materials, etc....
- Sagnibha Roy Admin 8 April 2024 9:11 PM
yes, sending
- A Aviral Singh 8 April 2024 9:09 PM
sir , can you please share the important notes

Below the messages, there are buttons for "Reminders" (No work due!), "+ Create Assignment", and "Start a Live Class". The system status bar at the bottom shows weather (28°C, Mostly sunny), system icons, and the date/time (08-04-2024, 21:04).

Create Assignment- passed

The screenshot shows the OCMS Software interface for a class named "Chemistry". A modal window titled "Create Assignment" is open in the center. It contains fields for the assignment title ("chemistry") and description ("chemistry 1"), a date and time selector ("10-04-2024 23:40"), and a file upload field ("Choose File") containing "assignment1.pdf". A "Create" button is visible at the bottom of the modal.

The background shows the same class discussion list as the previous screenshot. The system status bar at the bottom shows weather (28°C, Mostly sunny), system icons, and the date/time (08-04-2024, 21:38).

The screenshot shows the OCMS Software interface for creating an assignment. At the top, there's a navigation bar with tabs for Home, Classes, and a user icon. Below the navigation is a class summary for "Chemistry" by "Sagnibha Roy | 8 | chemistry". It includes a "Class Code - 123485" button and a "Set Class Timings" button. The main area has tabs for Discussion, Assignments (which is selected), Attendees, and Calendar. Under the Assignments tab, a card for "17 April 2024" at "9:10 PM" shows a list of assignments: "chemistry" and "chemistry 1". To the right, a "Reminders" section says "No work due!" with buttons for "+ Create Assignment" and "Start a Live Class". The bottom of the screen shows a taskbar with various icons and system status.

ASSIGNMENT CREATED

Submit Assignment- passed

The screenshot shows the OCMS Software interface for submitting an assignment. At the top, there's a navigation bar with tabs for Home, Classes, and a user icon. Below the navigation is a class summary for "Chemistry". The main area shows a submission card for "17/04/2024" at "9:10 PM" from "Sagnibha Roy Admin". The submission image is a photograph of a document titled "Assignment". To the right, there's a "Your Submission" section with an "Upload" button. The bottom of the screen shows a taskbar with various icons and system status.

Upload Assignment- passed

The screenshot shows a web browser window for the OCMS Software platform. The URL is <https://ocms-frontend.vercel.app/classes/123485/assignment/661411be8c70eb59c504aa0>. The page displays a submission for an assignment titled "chemistry". The submission was made by "Sagnibha Roy" (Admin) on "17/04/2024" at "9:10 PM". The file uploaded is "assignmentsolution.pdf". There are "Submit" and "ReUpload" buttons. The status bar at the bottom shows the date as 08-04-2024 and the time as 21:16.

After uploading assignments Students can submit it.

Grade Assignment- passed

The screenshot shows a web browser window for the OCMS Software platform. The URL is <https://ocms-frontend.vercel.app/classes/123485/assignment/661411be8c70eb59c504aa0/admin>. The page displays a grade breakdown for the "chemistry" assignment. It shows 1 submission, with 1 OnTime Submission and 0 Late Submissions. The total score is 10/10. A vertical slider on the right indicates the current view is at position 7. The status bar at the bottom shows the date as 08-04-2024 and the time as 21:18.

Teachers can also see late submissions and on time submissions.

View Grades- passed

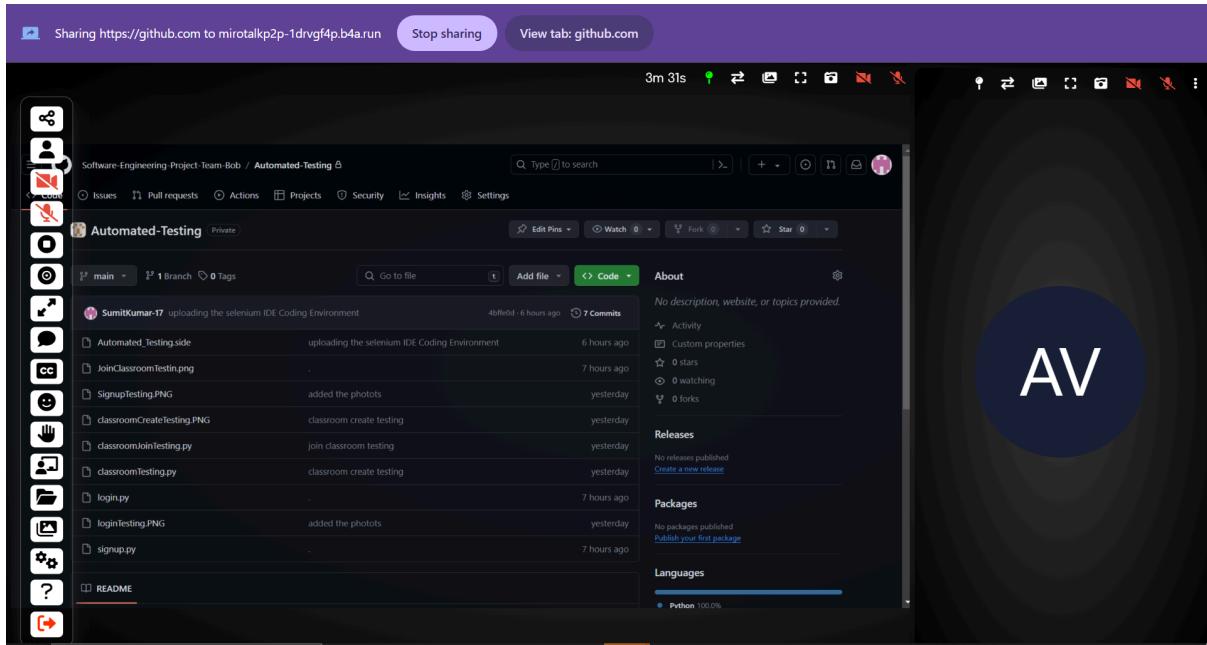
The screenshot shows a web browser window for 'OCMS Software'. The main content area displays a submission for a Chemistry assignment. On the left, there's a thumbnail image of a document titled 'assignment.pdf' with a timestamp of '17/04/2024' and a date/time of '9:10 PM'. The thumbnail also shows a profile picture of 'Sagnibha Roy Admin' and the text 'chemistry' and 'chemistry 1'. To the right, the submission status is shown as 'Submitted' with a green checkmark and 'Marks Secured :5/10'. Below this is a preview of the uploaded PDF file, which has a blue watermark reading 'assignmentsolution.pdf'. At the bottom right of the submission card is a red-bordered 'Unsubmit' button. The browser tab bar includes links for 'Automated-Testing/classes...', 'Compliance_Report_Sor...', 'Downloads', 'OCMS Software', 'Test_Plan_Sorrow - Google Sheets', and 'Inbox (5,717) - sagnibha...'. The system tray at the bottom shows weather (28°C, mostly sunny), system icons, and a date/time of '08-04-2024 21:19'.

Students can view their grades .

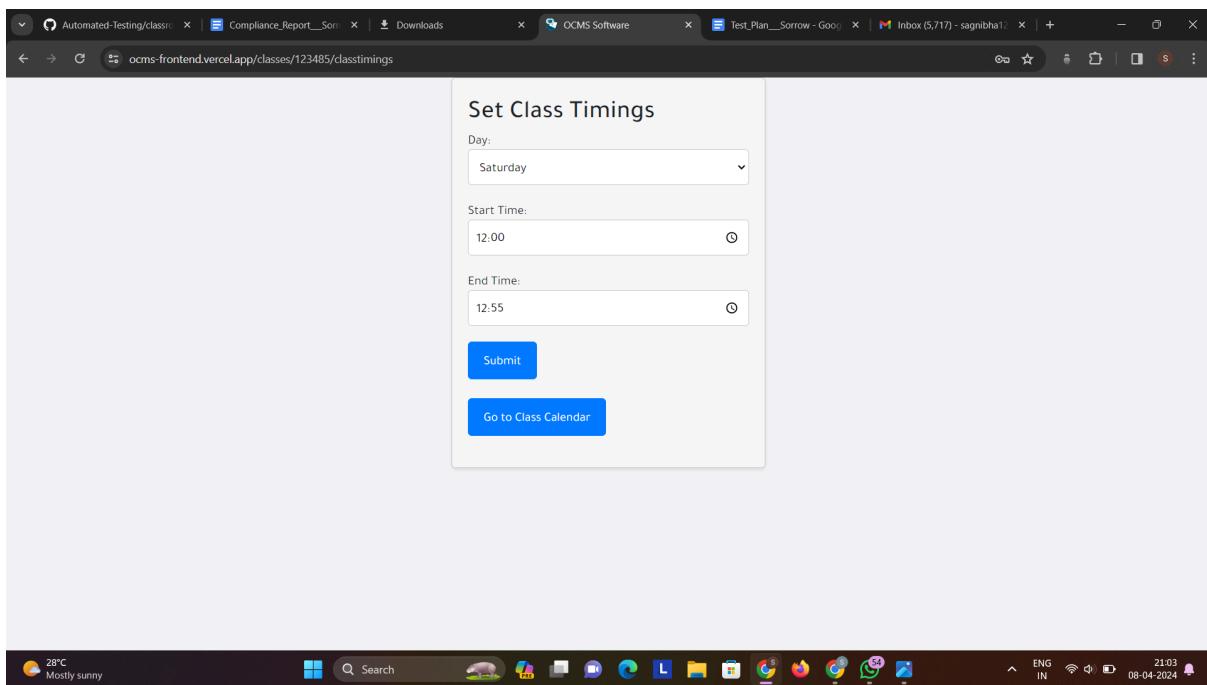
Create Live class- passed

The screenshot shows a web browser window for 'MiroTalk'. The main heading reads 'MiroTalk' and 'Free browser based Real-time video calls. Simple, Secure, Fast.' Below this, a sub-copy states: 'Start your next video call with a single click. No download, plug-in, or login is required. Just get straight to talking, messaging, and sharing your screen.' A large blue call-to-action button in the center says 'Pick a room name. How about this one?'. To its right is a text input field containing '2b0f62a4-0437-4773-88ef-4227890991da' and a 'JOIN ROOM' button. Below the input field is the text 'Your recent room: 2b0f62a4-0437-4773-88ef-4227890991da'. The browser tab bar includes links for 'Automated-Testing...', 'Compliance_Report...', 'Downloads', 'OCMS Software', 'MiroTalk a Free Secu...', 'Test_Plan_Sorrow - Google Sheets', and 'Inbox (5,717) - sagnibha...'. The system tray at the bottom shows weather (28°C, mostly sunny), system icons, and a date/time of '08-04-2024 21:14'.

Joining Live class- passed



Setting Class Timings - passed



Viewing Calendar- passed

Class Calendar for Chemistry

monday	wednesday	friday	saturday
13:00 - 15:55	10:00 - 11:45	08:00 - 08:55	12:00 - 12:55
Number of classes on this day: 1			

Automated-Testing/class... | Compliance_Report_Sorr... | Downloads | OCMS Software | Test_Plan__Sorrow - Google Sheets | Inbox (5,717) - sagnibha1...

ocms-frontend.vercel.app/classes/123485/calendar

28°C Mostly sunny Search ENG 22:13 08-04-2024

Selenium IDE Testing

Automated testing using Selenium IDE plays a pivotal role in ensuring the robustness and reliability of key functionalities within our application. By leveraging Selenium IDE, we systematically validate critical user interactions such as login, signup, classroom management (creation and joining), assignment submission, creation, and grading. This approach offers numerous benefits, including enhanced efficiency, repeatability, and accuracy in the testing process.

For login and signup functionalities, Selenium IDE meticulously simulates user actions, verifying that users can securely access the platform and create accounts without encountering any glitches. Similarly, the creation and joining of classrooms are rigorously tested to guarantee seamless navigation and functionality within these collaborative spaces.

Assignment management is another cornerstone of our application, and Selenium IDE ensures its smooth operation by validating the submission, creation, and grading processes. Through automated testing, we confirm that users can submit assignments successfully, instructors can effortlessly create assignments with specified parameters, and grading functionalities accurately evaluate student submissions.

Furthermore, Selenium IDE aids in identifying and addressing any potential issues or bugs early in the development cycle, thereby significantly reducing the likelihood of encountering them in production. This proactive approach not only enhances the overall quality of our application but also fosters user satisfaction by delivering a seamless and reliable user experience.

In summary, Selenium IDE serves as an indispensable tool in our testing arsenal, enabling comprehensive validation of critical functionalities and contributing to the overall stability and performance of our application. Its automation capabilities empower our team to iterate quickly, maintain code integrity, and deliver a superior product to our users.

