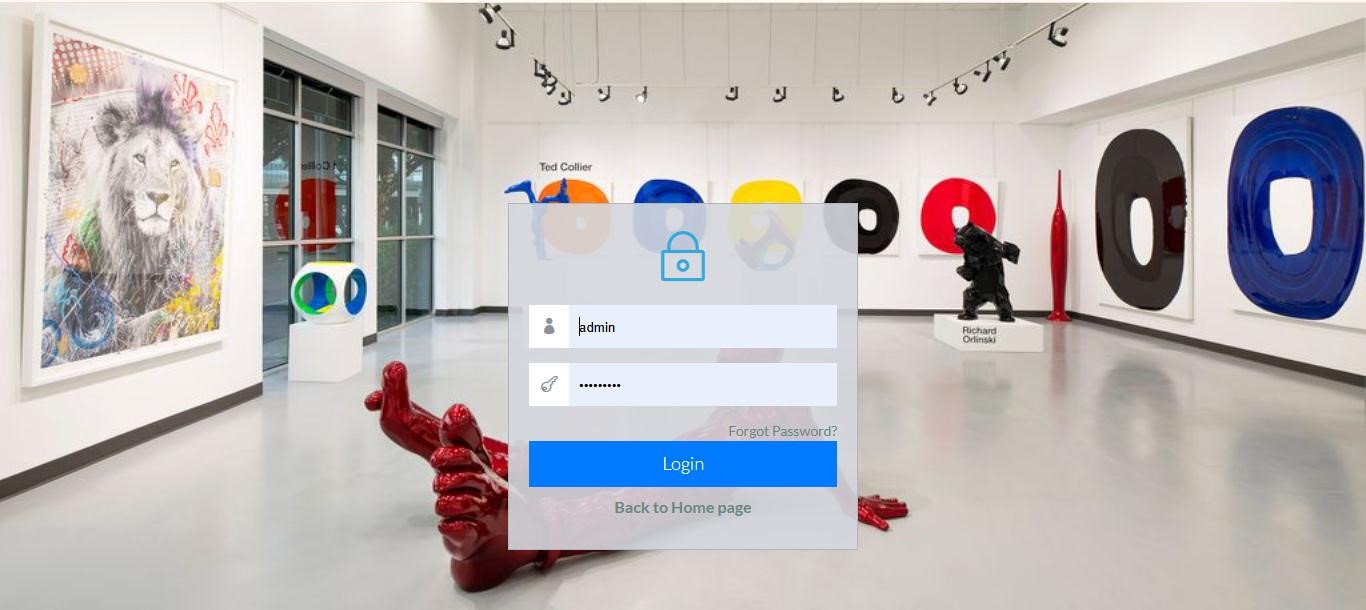
### IMPLEMENTATION

### 1 SYSTEM IMPLEMENTATION

#### 6.1.1 INTRODUCTION

In this process of testing the whole system and integrated software. The purpose is to evaluate the system compliance with the specified requirement. These are several types of testing.

### Login page



In the above screenshot the admin opens the admin log in page and enters the username and password so that they can gain access into the account.

### Type of Test

This is the process of testing complete and integrated software. The purpose is to evaluate the systems compliance with the specified requirements. There are several types of testing that can be done. The following tests were done:

**Unit testing** This is a software verification and validation method in which a programmer tests if individual units of source code are fit for use.

|  |  |  |
| --- | --- | --- |
| Test area | Expected results | Actual results |
| Logging in | A user with an account  should be able to log in | Logging in |
| Checking the Artwork entry module | To ensure it correctly stores and retrieves artwork details | Arwork details are captured |
| Verifying functions | So as to calculate pricing for artwork sales | .pricing for artwork sales |

Integration testing

This refers to testing the interaction module between different modules ad ensure they work together as expected

|  |  |  |
| --- | --- | --- |
| Test area | Expected results | Actual results |
| Artwork catalogue and search feature | To confirm that users can filter artworks | Users were able to filter and search for artworks |
| Artists profile | To confirm when a new artist is added the artist list and artwork assignment features update correctly | Artwork assignment features were updated when the new artist profile was added |

System testing

Refers to the process of testing an integrated hardware and software system to verify it meets its specified requirements.

|  |  |  |
| --- | --- | --- |
| **Test area** | **Expected results** | **Actual reports** |
| Full application | Verifying the system ability to handle simultaneous | Artists' profiles were updated artworks were managed and |
|  | gallery management tasks like updating artists' profiles managing artwork and processing visitor  transactions | visitor transactions were processed. |
| Full application | Ensure the system can handle various user roles and permissions such as the gallery admin, curators and visitors | Admin was granted permission to create and add artists |

Performance testing

To assess how the system performs under various conditions, including heavy load and stess

|  |  |  |
| --- | --- | --- |
| **Test area** | **Expected results** | **Actual results** |
| Load testing | To ensure the system can handle multiple simultaneous users without crashing | The system handled multiple users without crashing |

Acceptance testing

|  |  |
| --- | --- |
| **Objective** | **Achievement** |
| Gathering feedback from gallery administrators, artists, and visitors to confirm the system meets and their needs | admin was able to add artists profile without any difficulties |

Formal testing conducted to determine whether or not a system satisfies its acceptance criteria and to enable the customer the system.

# CHAPTER 6: SYSTEM IMPLEMENTATION

### Introduction

In this chapter, the newly developed system is addressed before it is deployed into the operations of the business. As a result, I am therefore going to examine the tools used for coding the system as well as testing, the system test plan, actual testing and finally propose a suitable change over method that the business should employ in order to bring the system into operation.

Tools used for coding and testing

During the coding process of the entire system, the following tools were of great importance for the project.

**Coding tools Editing:** During the coding process, I used the atom software as the tool for editing the code using the various languages as discussed below. Programming languages: During the coding process, I used the following web scripting languages:

* PHP: PHP is an abbreviation for Hypertext Pre-Processor. It is a very powerful programming language as it can be used in in web development as well as developing mobile device applications. I used PHP as my main programming language to implement most of functional requirements of the system ranging from variable declarations to query declarations.
* Html5: Html stands for Hypertext Mark-up Language. I used Html mainly to display text codes as well as formatting these texts.
* CSS: CSS stands for Cascading Style sheets. This is a very powerful language for formatting the web pages and has been of great help in my project. I used it to format the user interface in order to make it more appealing to the users.
* Java scripts: Java scripts played a very crucial role in adding some functionalities to my system. These included sliding images, a feature which is much clear in the system’s homepage.
* MYSQL: This is an abbreviation which stands for MY Structured Query language. I used MYSQL as the language to connect the PHP code to the database as well as executing the various queries.

Testing tools performance test

This test evaluates the working of the system that has been developed to establish whether it is solving the intended problem. Below are the tests that will be used for this system.

**Unit testing**:

This requires that testing be done on individual units constituting the entire system. This testing approach was to help identify errors since each unit was examined independently.

**Stress testing:**

This is a testing method that always tests the behavior of a system when subjected to unusual conditions. I tested the system with invalid input data such as unfilled input fields and no execution could continue. Actual system testing: This is done to the entire system to test the general working of the system after it has been fully developed. This test will be done on this system to test whether the objectives stated earlier have been achieved or not.

**Functional testing**:

This involves testing the functions of the program by providing an input data and observing the output. This will be done to test the working of the various functions of the program and any unexpected behavior will be identified and corrected accordingly.

System test plan

The system was tested in all aspects of functionality whereby various types of data inputs such as integers (INT), variable characters (VARCHAR), DATETIME and others were used and the results were observed.

User acceptance testing

During the testing process, any invalid data input altered the expected results and the system validation functions could alert the user of these invalid inputs. The system was also subjected to potential users for feedback and acceptance tests and got a positive response from these users whereby they accepted the system as a solution to inefficient manual operations in herb farmers market system. Acceptance testing was done after the completion of development process where the system was delivered to the users for their views and once they accepted the system, then the system is said to have met the user requirement. User acceptance for this system was be done at later stages of development to give potential users/clients an opportunity to give views about it.

Proposed change-over techniques Generally, there are four approaches for the implementation of the system in an organization. These are: Direct changeover, phased approach, pilot approach and the parallel approach. I greatly analyzed the four approaches to the system implementation and chose the phased operation

# CHAPTER 7: CONCLUSIONS AND RECOMMENDATIONS

##### **Recommendations**

In order to reverse the risks/problems involved in the project and realize improvements in succeeding developments, I would like to make the following recommendations.

1.Reduction in strictness of the Time deadlines

Since some of the issues in this system cover new concepts, I would recommend that the students be allowed to begin the project development at a quite early time to build up on their ideas and to complete early and meet the set deadlines by the requirements.

2. Provision of project finances to the students

Due to the fact that some of the students are unable to meet the threshold required for data and requirements capture, I would recommend that some special finances be provided to act as the support for the students who face difficulties in the development and research process.

3. Compelling some institutions to pave way for the students to develop

Some institutions have been a major bottleneck in the development of the projects and the higher-level institutions should compel them to release and loosen the restrictions they have over their intellectual property such API (Application Programming Interface).

4.Future improvements

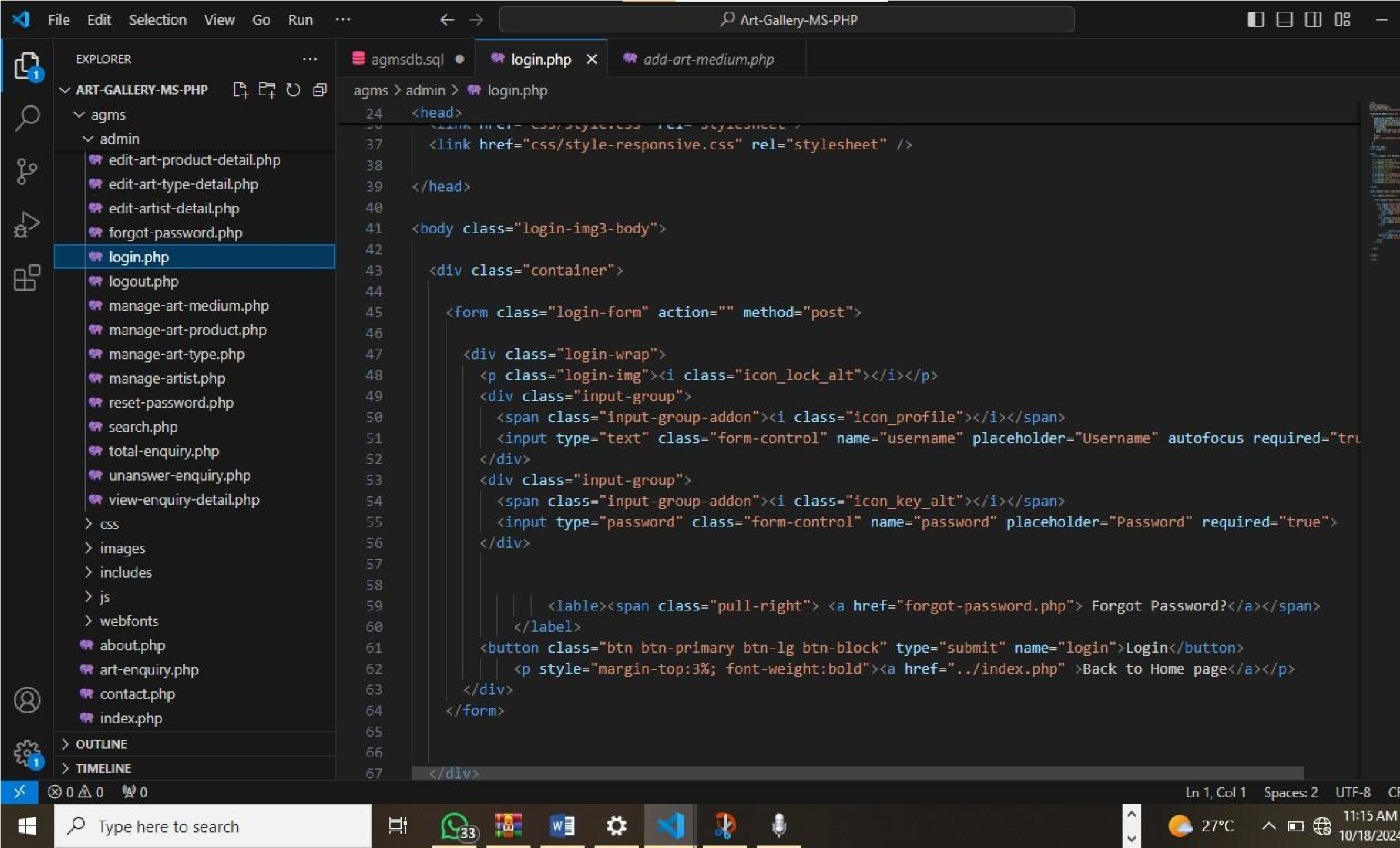
I would like to say that my system did not capture everything that would be required and would therefore recommend for future improvements

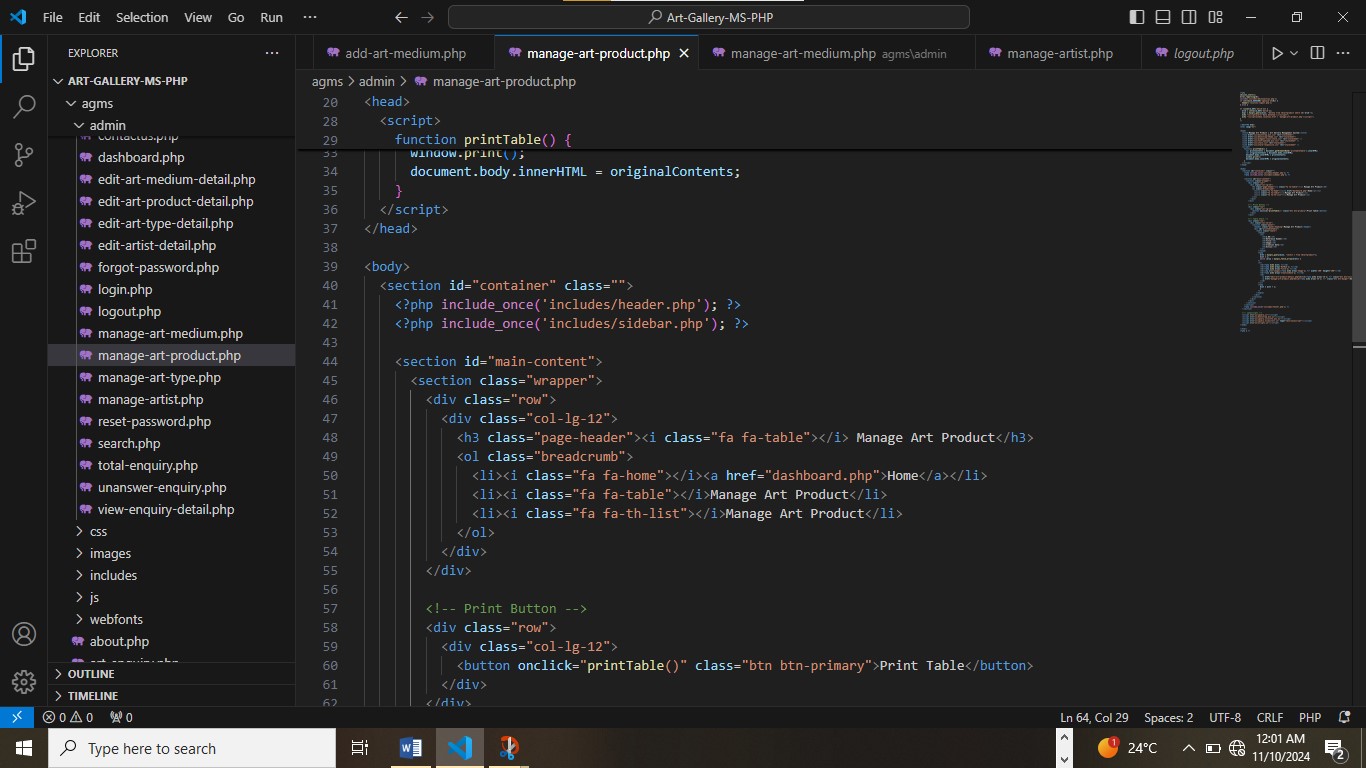
##### **CONCLUSION**

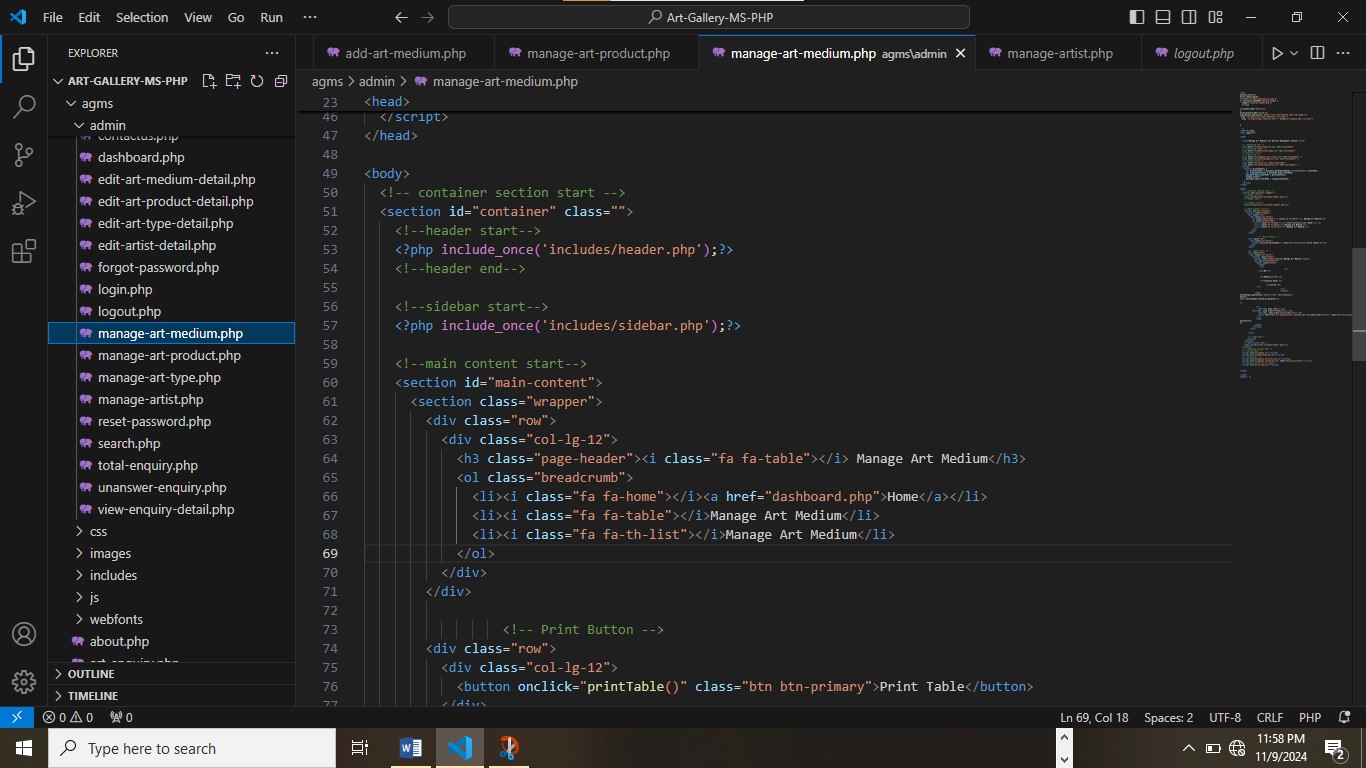
In conclusion, the proposed computerized system should be in a position to solve the identified problems and improve security of user’s and admin data in the system. The new computerized system has been designed to provide fast, easy and user friendly reliable system. The objectives of the system have been met in relation to having a good secured monitory system of every user of the system, payments.

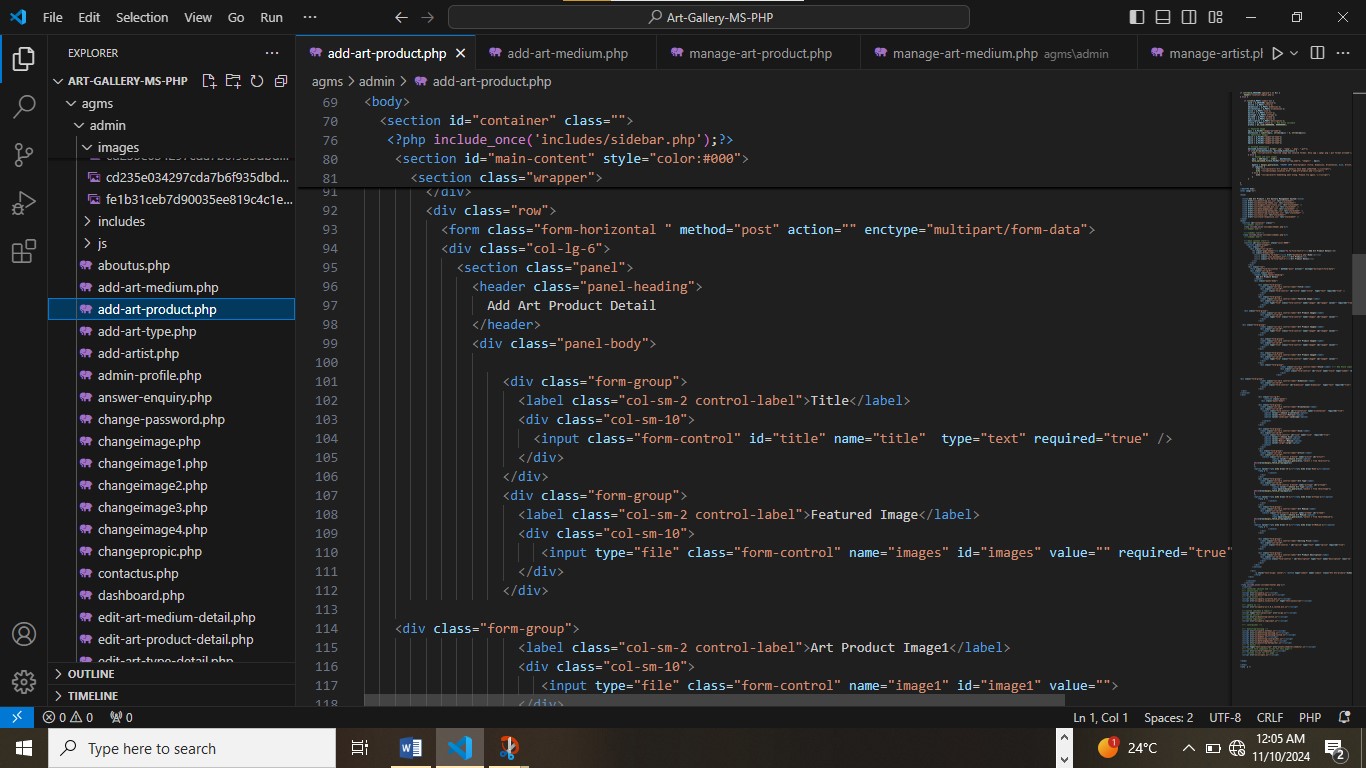
This project study has given me an opportunity to practice and improve on my development and design skills that were acquired during my learning period. The study of the project has also gave me a chance to utilize research skills, planning skills and time management skills. These skills and experiences will be useful in the course of my career development.

# Code sample









References

* "Flask Web Development: Developing Web Applications with Python" by Miguel Grinberg – Useful for understanding the Flask framework used in web applications.
* "Database System Concepts" by Abraham Silberschatz, Henry F. Korth, and S. Sudarshan – Provides foundational knowledge about databases, including design and management.
* "Clean Code: A Handbook of Agile Software Craftsmanship" by Robert C. Martin – Helps with writing clean, maintainable code, which is essential in any software project.
* “Principles of Good Database Design” on platforms like Medium or towardsdatascience.com for best practices in database design.
* “Restful API Design” on sites like Dev.to or DigitalOcean for understanding Restful architecture in APIs.
* Flask Documentation: https://flask.palletsprojects.com/ – Official Flask documentation for building web applications.
* SQLAlchemy Documentation: https://docs.sqlalchemy.org/ – For information on handling database operations within Python.
* Bootstrap Documentation: https://getbootstrap.com/docs/ – Useful for designing responsive, user-friendly front-end components for the application.
* "Don’t Make Me Think: A Common Sense Approach to Web Usability" by Steve Krug – Great for understanding user experience principles, which is key to creating a userfriendly art gallery management system.
* "The Design of Everyday Things" by Don Norman – Provides insights into intuitive design, relevant when designing interfaces for users with varying technical skills.