CPS 218 – WEB DEVELOPMENT TECHNOLOGY II

MORNING

ACCRA TECHNICAL UNIVERSITY , GHANA.

FACULTY OF APPLIED SCIENCE

DEPARTMENT OF COMPUTER SCIENCE

HND COMPUTER SCIENCE – YEAR II (FULL- TIME )

END-OF-SECOND-SEMESTER EXAMINATION, 2024/2025



Development of a Full-Stack Web-Based Palm Oil Marketing Management System for Nanamon farms

Table of Contents pages

1. **Introduction2**
2. Problem Statement2
3. System Requirements Specification2 - 3
4. **System Design 3**
   * Use Case Diagram
   * Entity Relationship Diagram (ERD)
5. Implementations Details4 - 7

* Technologies Used
* Screenshots

1. Testing8
2. Validation………………………………………………………………………………………………………………………………8 - 9
3. Deployment Instructions………………………………………………………………………………………………………….8
4. Challenges………………………………………………………………………………………………………………………………9
5. Recommendations……………………………………………………………………………………………………………..9
6. Conclusions…………………………………………………………………………………………………………………………..10
7. References……………………………………………………………………………………………………………………………10

# Introduction

The increasing need for digitization in agriculture has driven businesses like Nananom Farms to adopt web-based solutions for palm oil sales and service coordination. This project simulates a real-world marketing system where customers can book services, make enquiries, and interact with the business through a dynamic and responsive website. It also provides administrative tools for managing services, bookings, and customer feedback securely and efficiently.

# 2.Problem Statement

Nananom Farms previously relied on manual or offline methods to manage sales, customer communication, and service scheduling, which caused delays, inefficiencies, and poor customer engagement. The lack of a centralized online platform created barriers for scaling operations, maintaining records, and reaching a broader audience.

This project solves the above by providing a responsive, database-driven web platform to improve operations, communication, and accessibility.

# 3.System Requirements Specifications

## Functional Requirements

- Customers can book services via a booking form.  
- Customers can submit enquiries.  
- Admins can log in to manage services, bookings, and enquiries.  
- Role-based access control (admin vs. support agent).  
- System provides dynamic feedback and confirmation.

## Non-Functional Requirements

- System should be responsive on all devices.  
- Booking form must support date/time input.  
- Forms must include input validation (client + server side).  
- Backend must securely handle sessions and data storage.

## Technologies Used

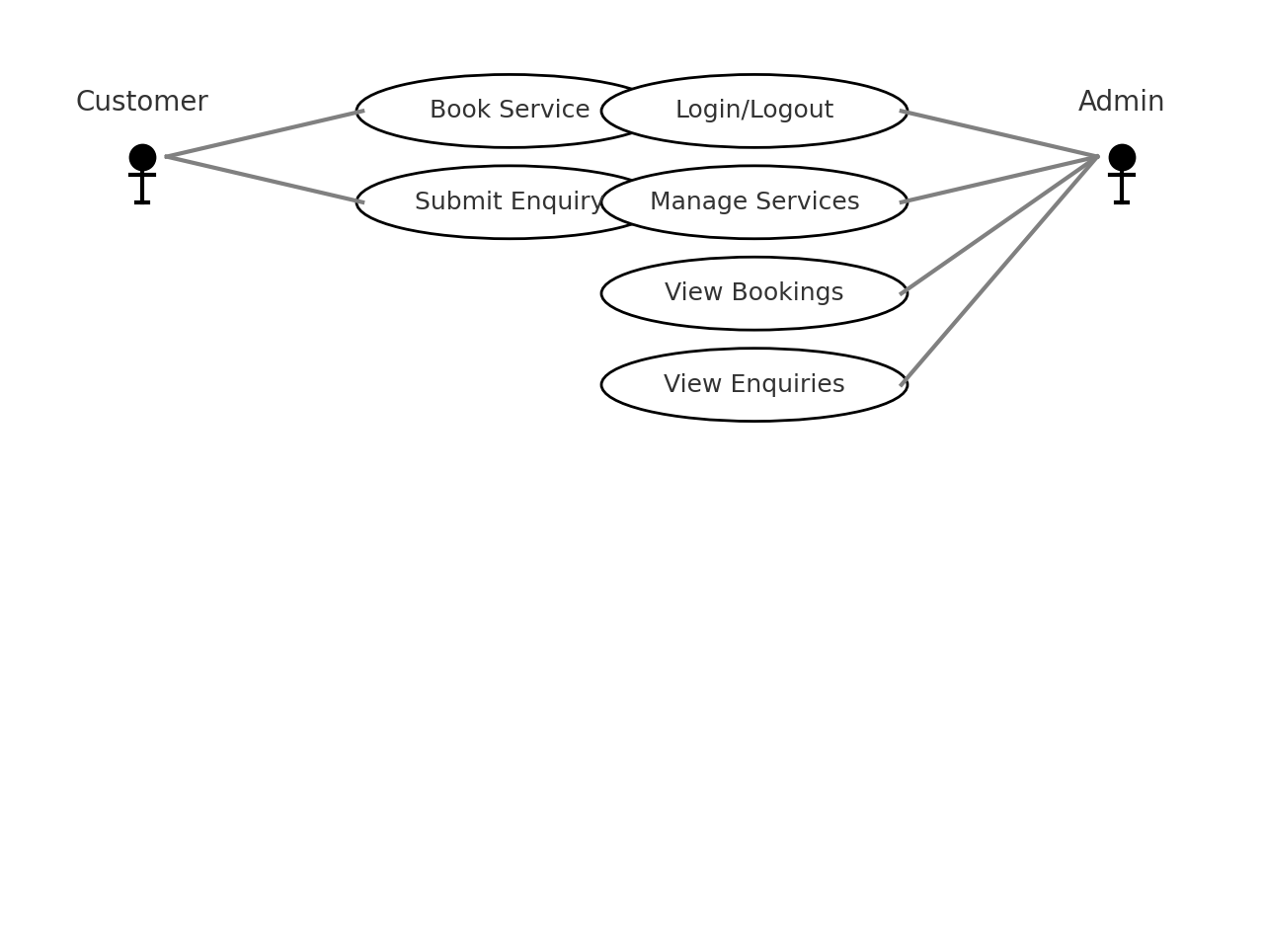
- Frontend: HTML, CSS, JavaScript, Bootstrap  
- Backend: PHP  
- Database: MySQL (phpMyAdmin)

- Server: XAMPP (local deployment)  
- Tools: VS Code, Chrome, phpMyAdmin, PHP-Mailer.

# 4. System Design

## Use Case Diagram

Below is the Use Case Diagram representing the interactions between system actors and their respective functionalities.



# 5. Implementations Details

## **Technologies Used**

## Component Technology

Frontend HTML, CSS, JavaScript, Bootstrap

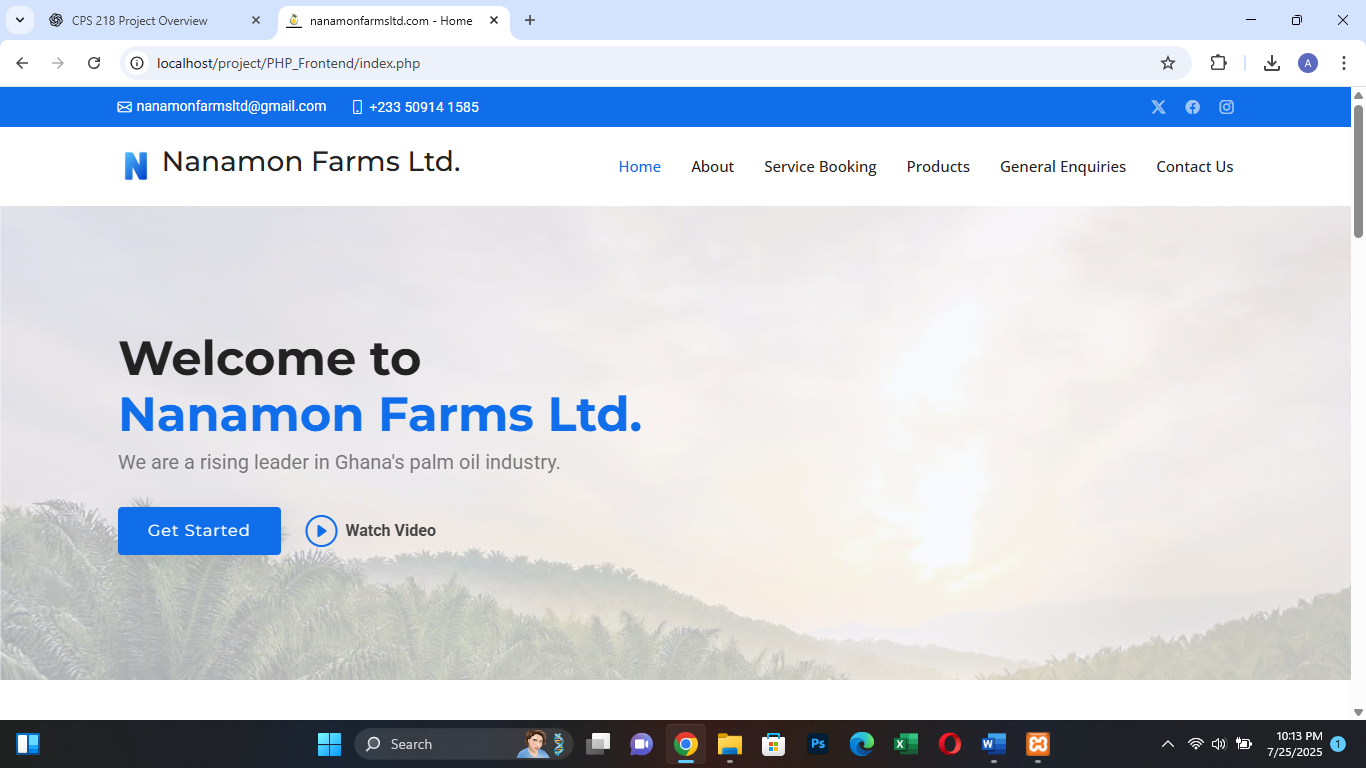
Backend PHP, PHP-Mailer

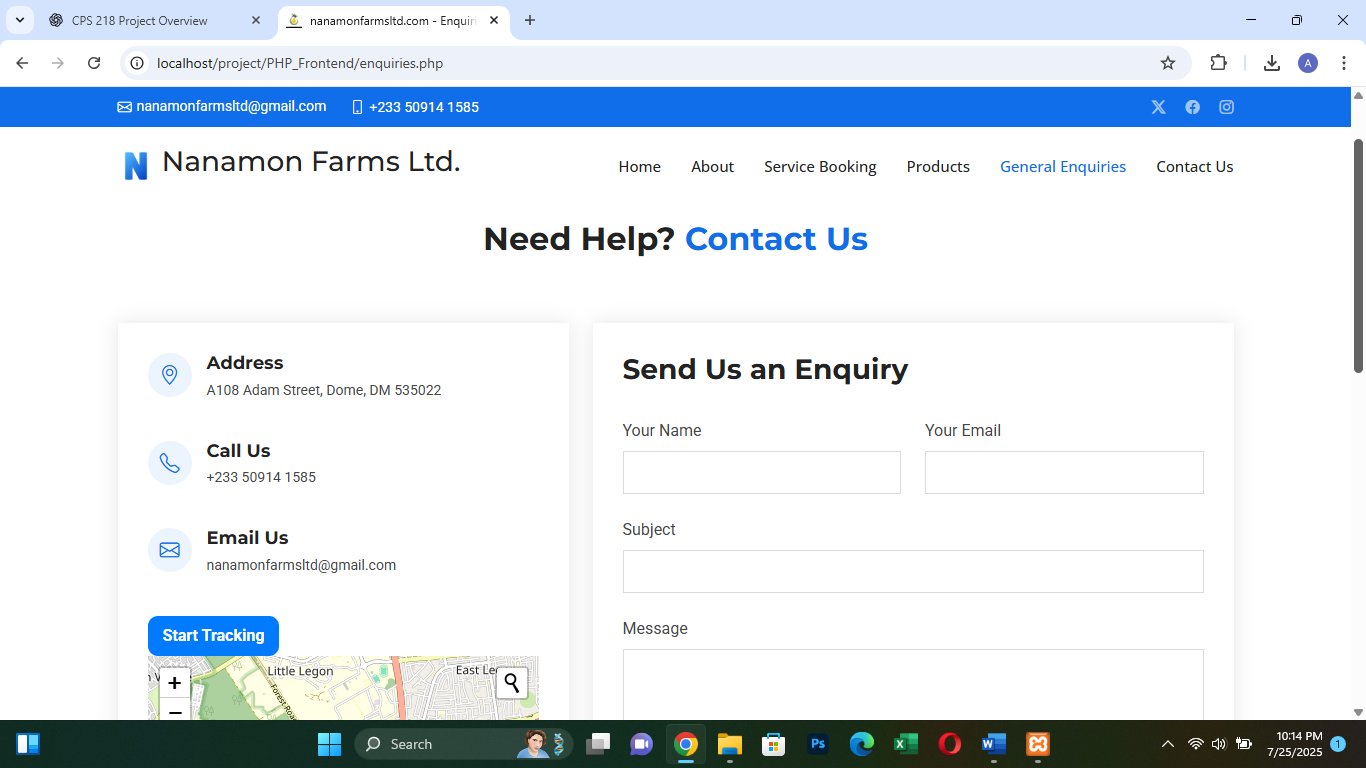
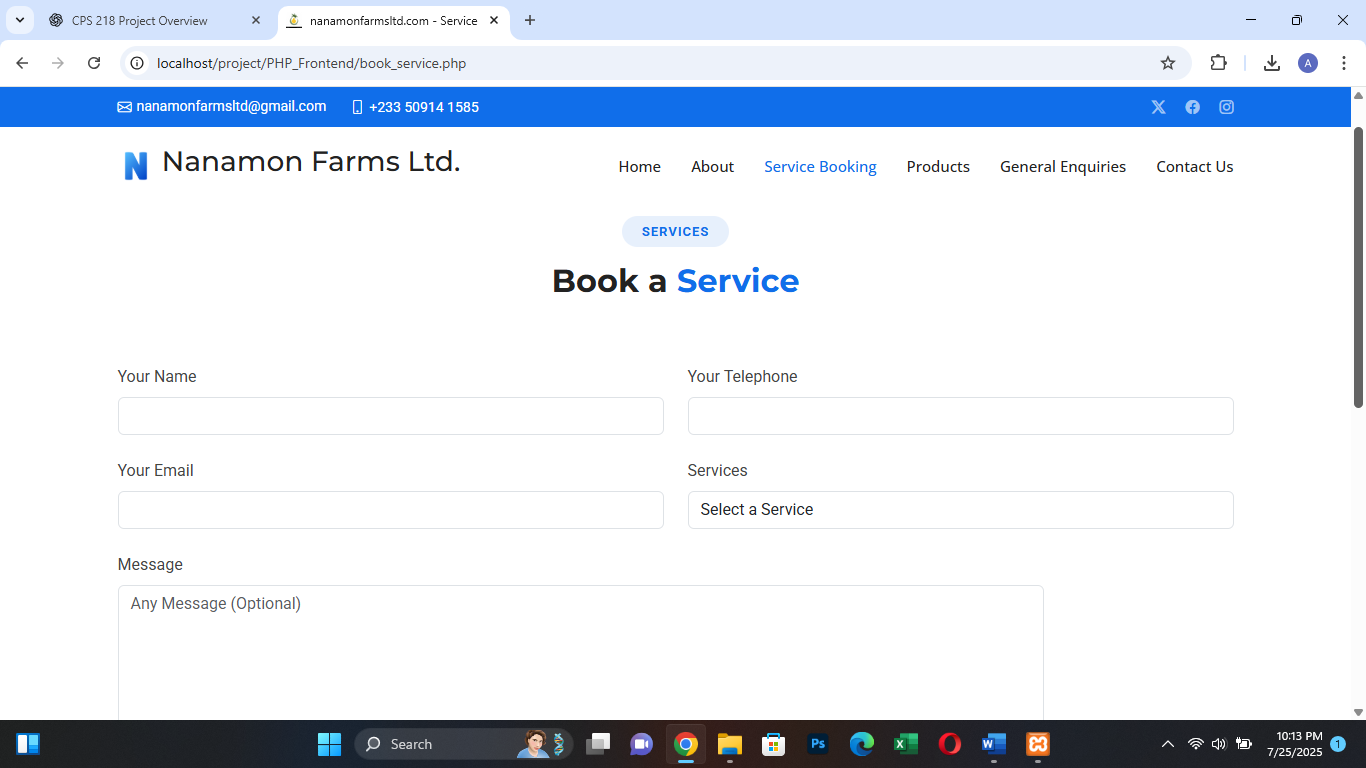
Database MYSQL ( phpMyAdmin )

Web Server Apache ( via XAMPP )

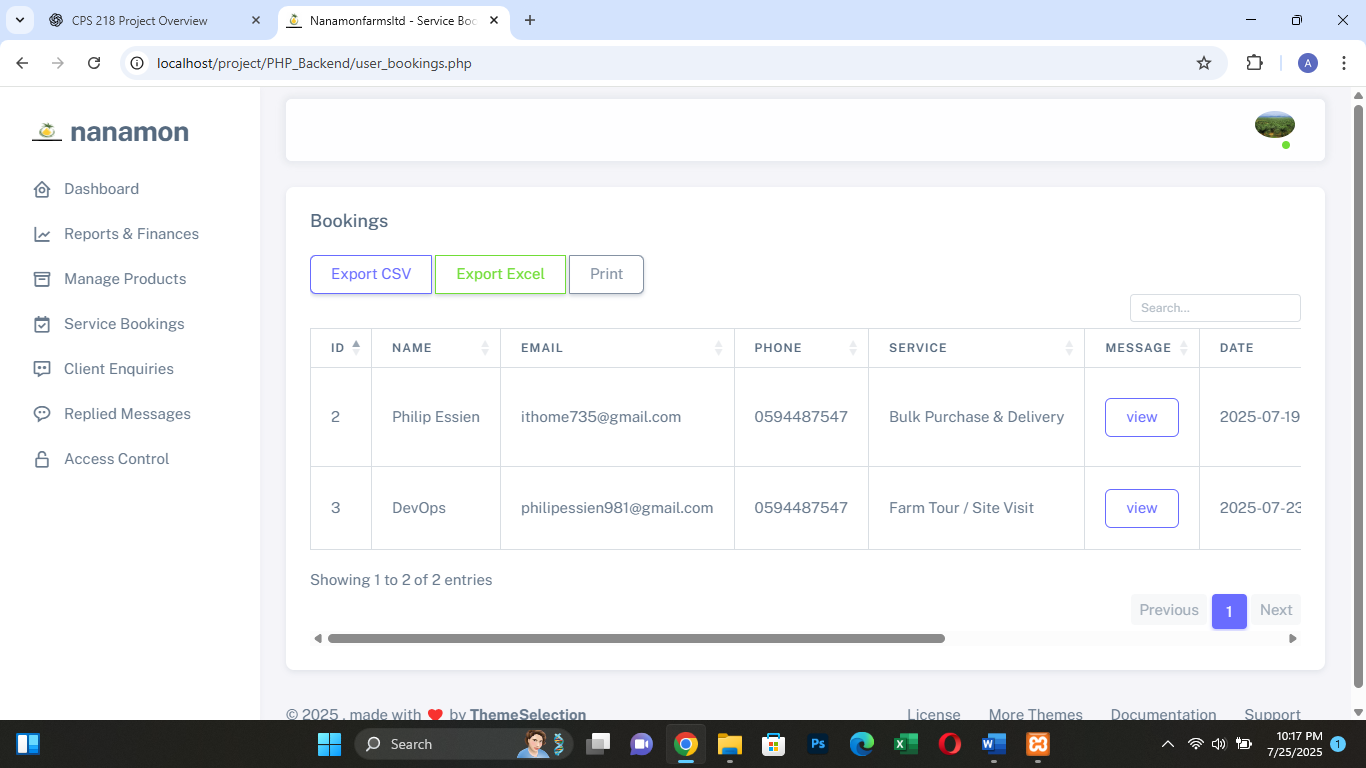
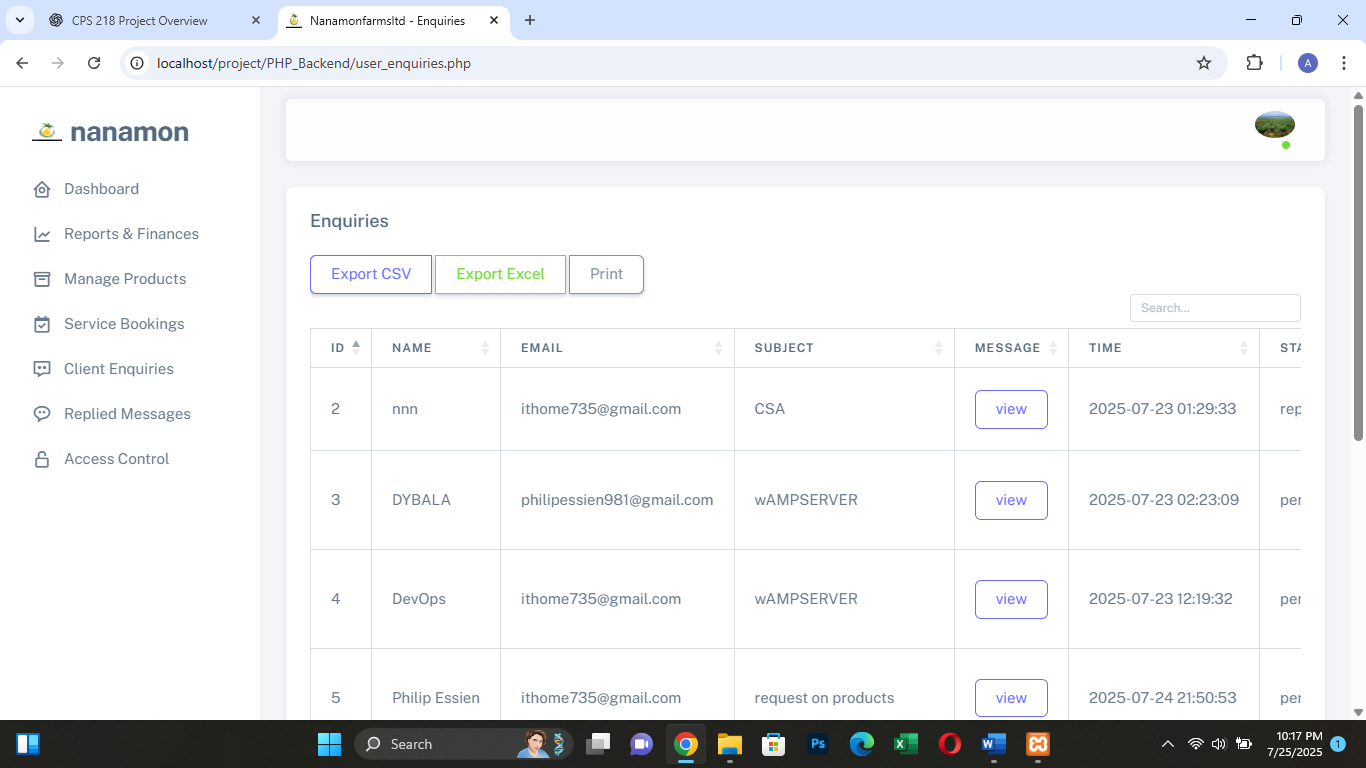
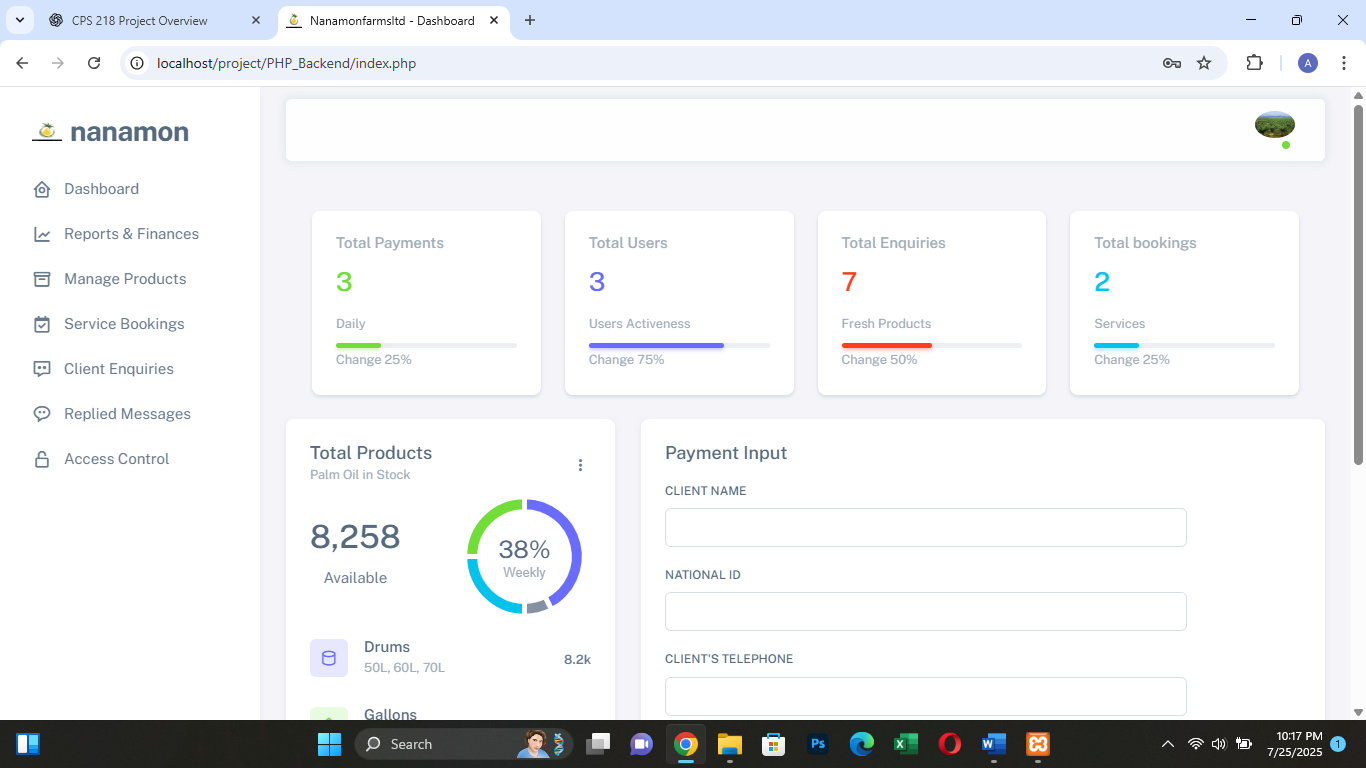
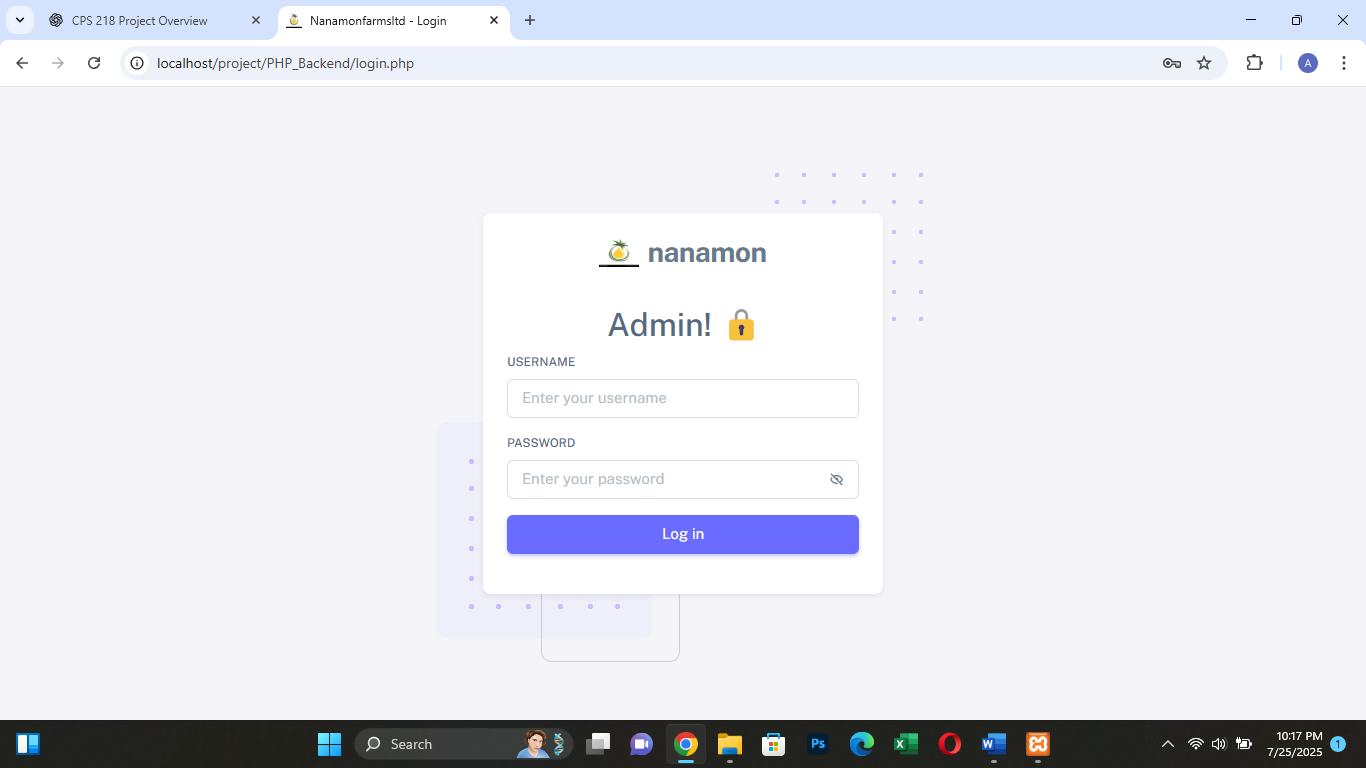
Screenshots

Frontend





Backend



# 6. Testing

Test Cases

|  |  |  |  |
| --- | --- | --- | --- |
| Feature | Test Input | Expected Result | Status |
| Booking Form | Valid name/date/service | Booking success message | |  | | --- | | ✅ |  |  | | --- | |  | |
| Booking Form | Missing fields | Validation error | |  | | --- | | ✅ |  |  | | --- | |  | |
| Login | Correct credentials | Admin redirected to dashboard | |  | | --- | | ✅ |  |  | | --- | |  | |
| Login | Wrong credentials | |  | | --- | | Error message shown |  |  | | --- | |  | | |  | | --- | | ✅ |  |  | | --- | |  | |
| |  | | --- | | CRUD (Services) |  |  | | --- | |  | | |  | | --- | | Add/Edit/Delete |  |  | | --- | |  | | Reflects in database | ✅ |
| CRUD (Enquiries) | Add/Delete enquiry | Reflects in database | ✅ |
| Email (Enquiry) | |  |  |  | | --- | --- | --- | | |  | | --- | | Valid name, email, subject, message |  |  | | --- | |  | |  |  | | --- | |  | | |  | | --- | | User receives email, and sees success message |  |  | | --- | |  | | ✅ |
| Email(booking) | |  | | --- | | Valid booking data |  |  | | --- | |  | | |  | | --- | | Confirmation email sent to user. |  |  | | --- | |  | | ✅ |
| |  | | --- | | Email (Failure) |  |  | | --- | |  | | |  | | --- | | Invalid SMTP configuration |  |  | | --- | |  | | |  | | --- | | PHPMailer error handled/logged securely |  |  | | --- | |  | | ✅ |

# 7. Validation

Client-side (HTML/JS):

* `required` on form inputs.
* `type="email"` for email fields.
* Use of input patterns to restrict malformed data.

Server-side (PHP + PHPMailer):

* Use `filter\_var()` to validate emails.
* -Use `htmlspecialchars()` and `trim()` to sanitize text.
* Use `mysqli\_prepare()` and `bind\_param()` to prevent SQL injection.

# 8. Deployment Instructions

**🖥️ Local Setup (XAMPP)**

1. Install XAMPP
2. Place your project in htdocs/
3. Import the SQL file into phpMyAdmin
4. Update database credentials in db.php
5. Access via <http://localhost/project-root/>

# 9. Challenges

**Challenges:**

* Difficulty implementing secure login/logout.
* Handling form validation on both frontend and backend.
* Learning PHP mail integration.
* Managing file uploads and image previews.

# 10. Recommendations

* Use frameworks like Laravel in future versions
* Add real payment integration (MTN MoMo, API, Stripe)
* Improve dashboard with graphs and charts

# 11. Conclusion

This project provided a practical experience in developing a real-world full-stack web application. It empowered our team to integrate front-end and back-end technologies using HTML, CSS, JavaScript, PHP, PHP-Mailer and MySQL. The system is modular, responsive, and solves a real business challenge for Nananom Farms, contributing to digitizing agricultural operations in Ghana.

# 12. References

* W3Schools – PHP and MySQL Tutorials
* MDN Web Docs – HTML/CSS/JS
* Stack Overflow – Problem-solving community
* XAMPP – Local development environment