|  |
| --- |
|  |
| IMG_256 Documentation of e-store of Laptops project. |
| **Members of project:**  SULEYMAN Jumaniyazov EG6X6K  ALBERTO Balazs Sandor NKNHIK |
| **Subject**: Web Programmin II |
| **Academic Year: 2024-2025**  John Von Neumann University |
| Introduction with Thanks Our am immensely grateful to the divine presence that has granted me the fortitude, vitality, and determination to accomplish this endeavor.  Our heartfelt gratitude extends to my beloved family members whose unwavering support and various forms of assistance have propelled me to this juncture of our educational journey.  We extend oursincere appreciation to Dr. Subecz Zoltan , our esteemed supervisor, whose invaluable guidance, insightful feedback, and unwavering encouragement have been instrumental in bringing this project to fruition.  We express profound gratitude to the esteemed members of the evaluation committee for graciously agreeing to assess our work.  Lastly, I extend heartfelt thanks to all the educators who have imparted their knowledge and wisdom, as well as to all individuals who have supported and assisted me throughout my academic and professional pursuits. |

# Catalog

[THANKS 2](#_Toc1428902693)

[INSTRUCTION 4](#_Toc257936850)

[1) Description  : 5](#_Toc1522638304)

[2) Download 5](#_Toc1566837676)

[3) Then opening application after successful installation and creating out entity 6](#_Toc1396341018)

[Chapter II : Analysis and presentation of the subject 7](#_Toc602195110)

[1) Analysis of the project : 7](#_Toc2785459)

[CHAPTER III: conception 9](#_Toc1718052826)

[1) Introduction : 9](#_Toc248729020)

[2) UML : 9](#_Toc1385462078)

[3) Conclusion 13](#_Toc295960525)

[CHAPITRE IV : Implementation 13](#_Toc636417223)

[1) Introduction : 14](#_Toc1795704901)

[2) Application Presentation : 14](#_Toc1824579816)

[Conclusion 26](#_Toc1793971999)

[WEBOGRAPHIE 28](#_Toc616983313)

# INSTRUCTION

### Web Application Documentation

#### Task Overview:

Create a server-side web application in PHP using a chosen database. Adapt the website theme to the database and implement required functionalities.

#### Group Formation and Task Allocation:

1. Form groups of two.
2. If an individual is without a partner, they must still use two GitHub accounts to complete the task.
3. Choose a database from the provided Google Drive folder. Each group can select only one database.
4. Enter pair and database selection details in the "Pair and Database selection for the Homework.xlsx" file in the Teams group.

#### Project Requirements:

**Server-Side Web Application in PHP:**

* 1. **Menu Management:**
     1. Store menu item names in the database.
     2. Implement a multi-level menu system.
  2. **User Management:**
     1. Implement registration and login functionalities.
     2. Separate roles: "visitor," "registered visitor," and "admin."
  3. **Homepage:**
     1. Introduce the selected topic on the first page.

**Responsive Theme:**

* 1. Find and apply a free responsive theme for the pages.
  2. Describe the chosen theme in the documentation.

**SOAP Server:**

* 1. Create a RESTful web service using all tables in the database.

**SOAP Client:**

* 1. Create a client to test the SOAP web service.

**SOAP-MNB Integration:**

* 1. Use the Hungarian National Bank SOAP data service.
  2. Implement queries to fetch exchange rates for a given day and a given month.
  3. Display data in a table and graph (using Chart.js).

**RESTful Server:**

* 1. Create a RESTful web service for the database with GET, POST, PUT, DELETE functions.
  2. Test the web service with cURL and Postman.
  3. Document the testing steps.

**RESTful Client:**

* 1. Create a client to test the RESTful web service (GET, POST, PUT, DELETE functions).

**PDF Creation:**

* 1. Create a PDF creation service using TCPDF.
  2. User inputs data into three text fields or a drop-down list.
  3. System reads from the database and generates a downloadable PDF file using three tables.

**Deployment:**

* 1. Upload and implement the application on an internet hosting provider.
  2. Ensure the URL includes one member's name.
  3. Operation will be verified based on the deployment.

**GitHub Version Control:**

* 1. Use GitHub for version control.
  2. Upload the project in at least 5 steps per person.
  3. Clearly identify uploads by each group member.

**Project Documentation:**

* 1. Create a detailed documentation of at least 15 pages with screenshots.
  2. Present the application and describe the implementation of each task.
  3. Include the URL of the deployed page, GitHub project link, and login details for an admin and another user.

**Chapter 1: Project Overview and Database Management**

* 1. Project Introduction

The web application is a server-side PHP project designed to manage and interact with a laptop database. The project focuses on creating a comprehensive web platform with multi-level functionality, user management, and various data interaction methods.

* 1. Database Selection and Structure

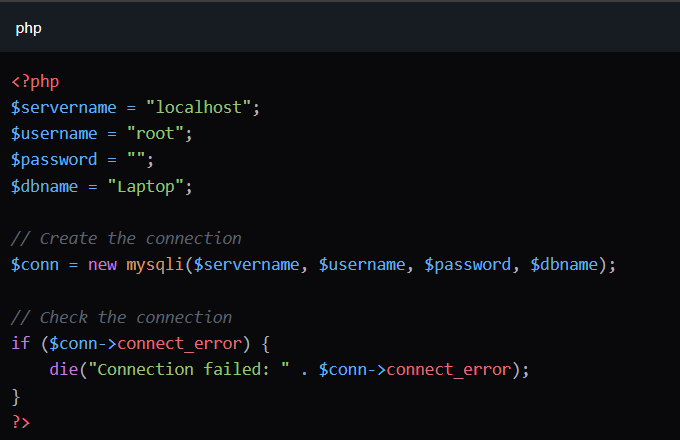
Database Name: Laptop Primary Tables:

1.processor

2.opsystem

3.notebook

1.3 Database Connection Strategy



* 1. Database Import Mechanism

A custom DatabaseImporter class was developed to handle data importing from text files:

Key Features of Import Process:

* Supports UTF-8 encoding
* Removes Byte Order Mark (BOM)
* Handles potential import errors
* Imports data in a specific order to maintain referential integrity

Import Methods:

importProcessors()

* Imports processor data
* Clears existing table before import
* Validates and sanitizes input

importOperatingSystems()

* Imports operating system data
* Ensures data consistency

importNotebooks():

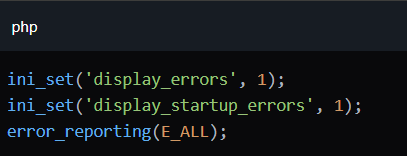
* Imports notebook details
* Maps data to appropriate columns
* Handles type conversions

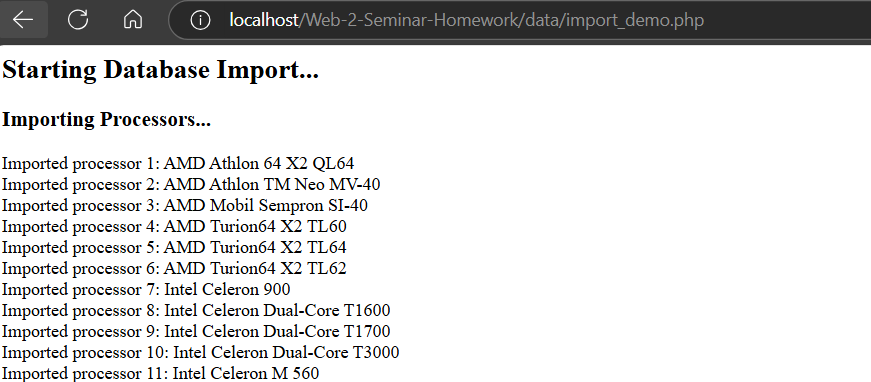
Sample Import Code Snippet

# 

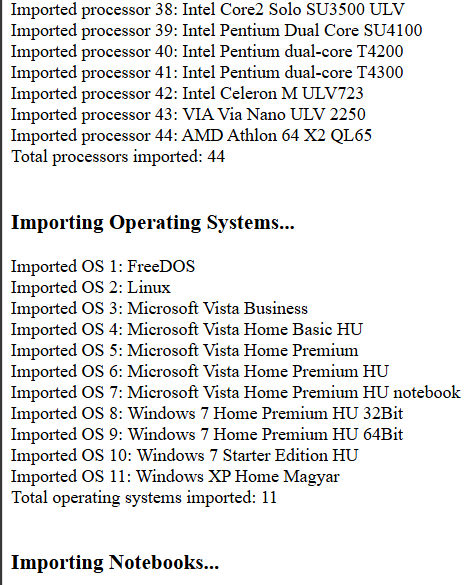
1.5 Error Handling and Logging

* Custom error reporting enabled
* Comprehensive error messages
* Detailed logging of import process





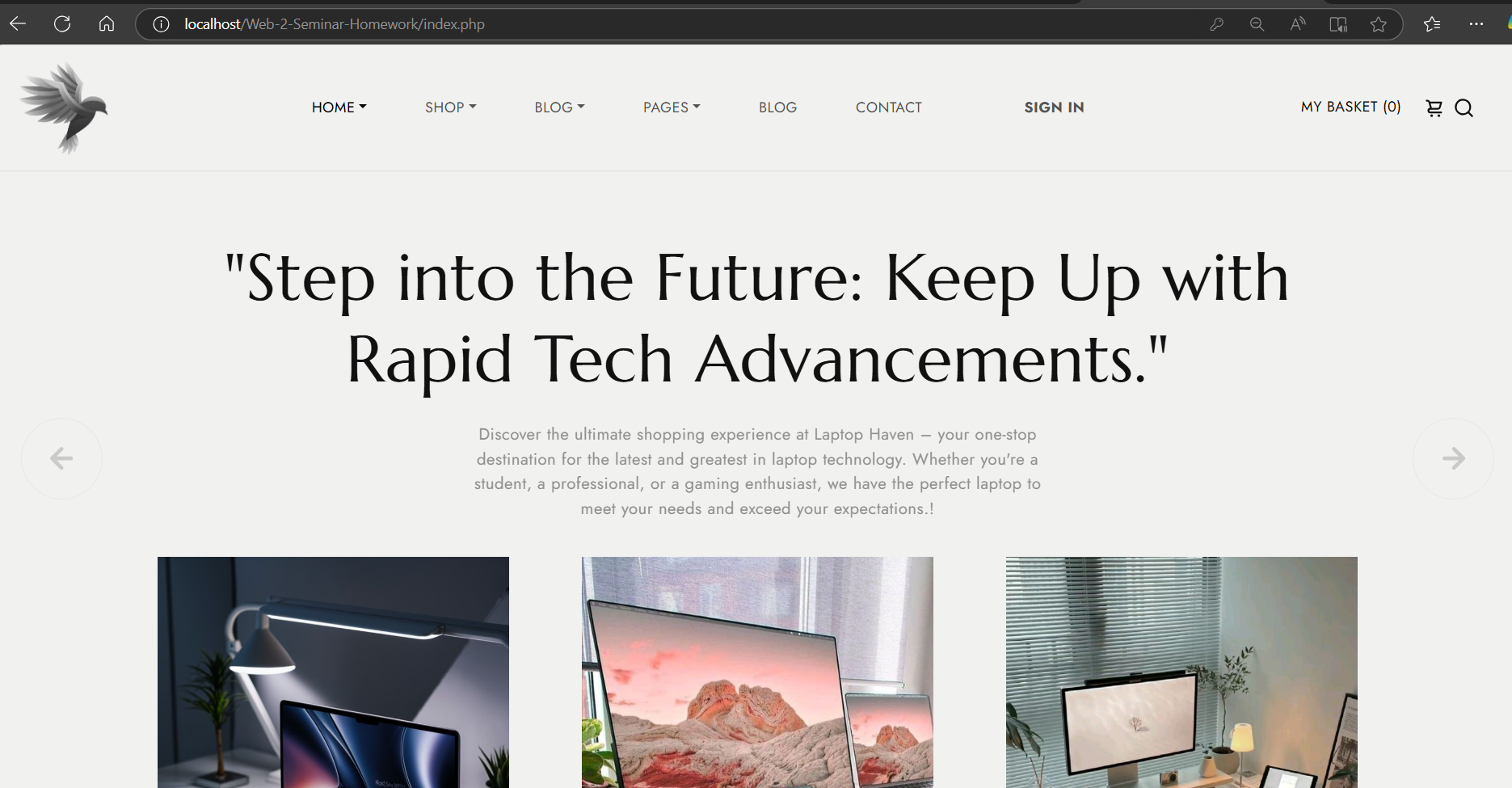
…



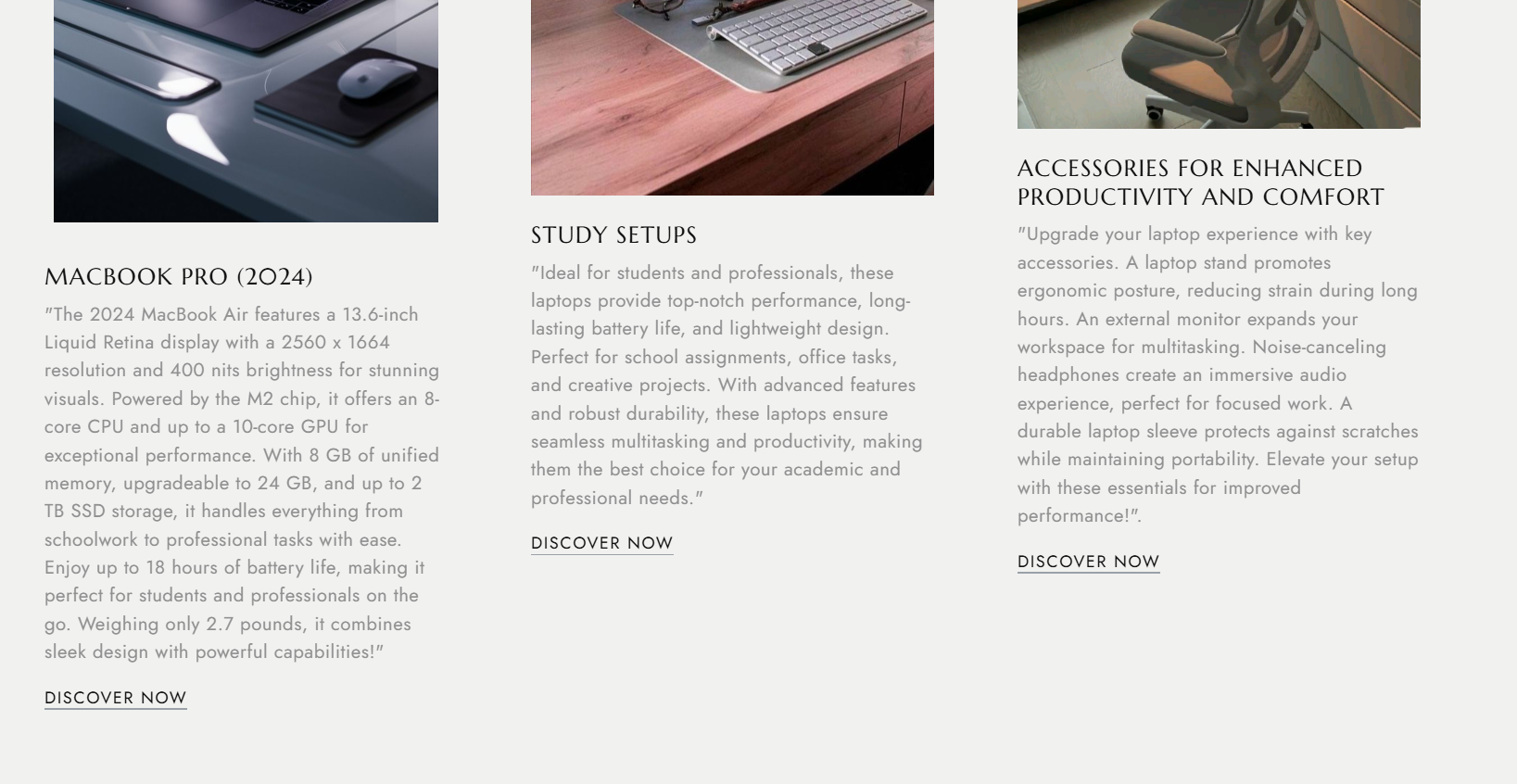
… like this in front page.

# **Responsive design**

We used Theme Wagon and implemented that template design to out page.



…

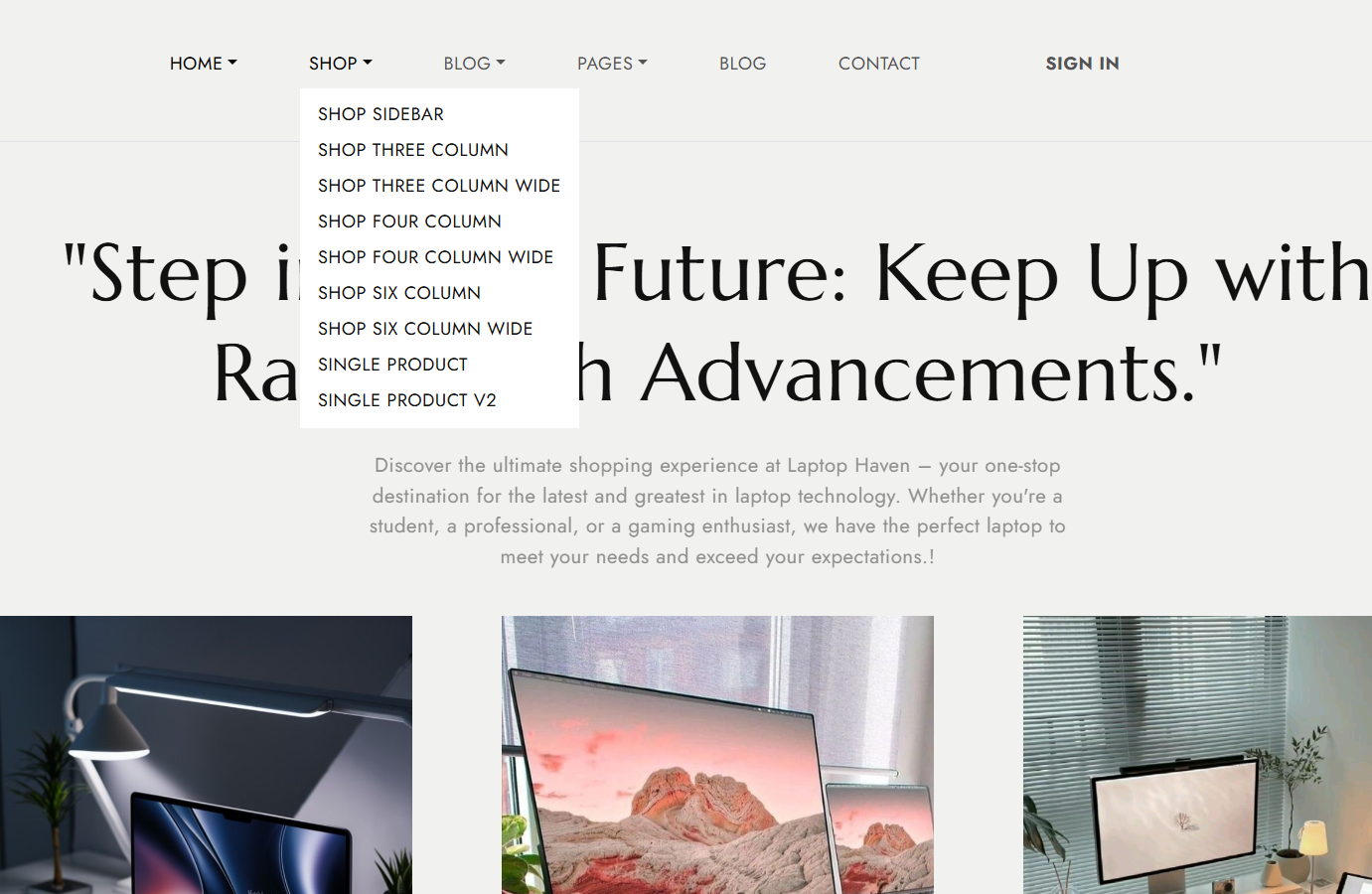


… and so on, its very long and has a lot of interesting animations, and static menus.

# **Implement a multi-level menu system**

# 

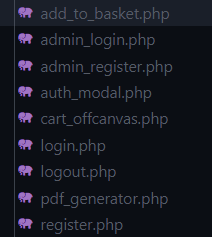
Navigation menu is responsive according to size of page, and space available



Registration, Login: at least "visitor", "registered visitor" and "admin" roles are separated.

# **Registration, Login: at default "visitor", after registration "user" and "admin" roles are separated.**

# 



# Conclusion

# …

# WEBOGRAPHIE

Online References :

* <https://github.com/abmagarin/Web-2-Seminar-Homework>