|  |  |
| --- | --- |
| D:\UKTC\UKTC tempatets\Logo\UKTC-logo-color.png | **Национална професионална гимназия по компютърни технологии и системи**  **гр.Правец при технически университет – СОФИЯ** |

**Дипломна работа**

ДЪРЖАВЕН ИЗПИТ ЗА ПРИДОБИВАНЕ НА ТРЕТА СТЕПЕН НА ПРОФЕСИОНАЛНА КВАЛИФИКАЦИЯ – ЧАСТ ПО ТЕОРИЯ НА ПРОФЕСИЯТА

на

Ученик: Николай Малинов Колибаров

ученик от 18125 (XII) клас

Тема: **Уеб сайт за експонатите в Лувъра**

**ПРОФЕСИОНАЛНО НАПРАВЛЕНИЕ:** 481 КОМПЮТЪРНИ НАУКИ

**ПРОФЕСИЯ:** 481020 системен програмист

**СПЕЦИАЛНОСТ:** 4810201 системно програмиране

Ученик: ………………….. Ръководител: ………………………

*Елена Първанова*

Правец, 2023

Съдържание

[**Увод** 2](#_Toc131177795)

[**Глава 1 Литературен обзор** 3](#_Toc131177796)

[**Frontend:** 3](#_Toc131177797)

[1. HTML5: 3](#_Toc131177798)

[2. JavaScript: 4](#_Toc131177799)

[3. CSS: 6](#_Toc131177800)

[4. Bootstrap: 7](#_Toc131177801)

[**Backend:** 8](#_Toc131177802)

[1. PHP: 8](#_Toc131177803)

[2. MySQL: 9](#_Toc131177804)

[3. XAMPP: 10](#_Toc131177805)

[**I D E:** 11](#_Toc131177806)

[1. Visual Studio Code: 11](#_Toc131177807)

[**Глава 2 Теоретична част** 13](#_Toc131177808)

[1. Структура на директориите 13](#_Toc131177809)

[на проекта: 13](#_Toc131177810)

[**Глава 3 Инженерно решение на поставената задача** 14](#_Toc131177811)

[1. Основни функции на кода 14](#_Toc131177812)

[2. Структура на базата данни 14](#_Toc131177813)

[и таблиците в нея: 14](#_Toc131177814)

[**3.** **SQL CREATE заявки за** 17](#_Toc131177815)

[**създаване и попълване на таблици** 17](#_Toc131177816)

[**Глава 4 Анализ на получените резултати, приложимост и изводи** 18](#_Toc131177817)

[**1.** **Работа на уебсайта** 18](#_Toc131177818)

[Заключение 21](#_Toc131177819)

[Използвана литература 22](#_Toc131177820)

[Списък с използваните означения и съкращения 23](#_Toc131177821)

# **Увод**

Настоящата дипломна работа има за цел създаването на сайт, което служи за информация на хора, които желаят да посетят музея Лувъра. Създадено е със средствата на маркиращият език ***HTML (HyperText Markup Language)*** и стилизиращият език ***CSS (Cascading Style Sheet).***

Сайтът съдържа най-важната информация свързана с Лувъра, както и някои популярни експонати. Системата е направена с помощта на ***Visual Studio Code*** и използва Бази Данни чрез ***MySQL-***поддръжка.

Налични са всички по-важни функции, които се изискват за такъв тип сайт, а именно:

1. Регистрация на ***потребители*** чрез потребителско име, имейл и парола.
2. Вписване на ***потребители*** чрез потребителско име и парола.
3. Вписване като обикновен ***потребител*** или ***администратор***, от което зависи и нивото на достъп и вида операции, които могат да се изпълняват.

Потребителският ***интерфейс*** е опростен, за да не затруднява работата на ***потребителите***.

# **Глава 1 Литературен обзор**

В настоящия проект са използвани следните технологии, езици, среда за разработка ***(I D E – Integrated Developement Enviroment)*** и работни рамки ***(FrameWorks):***

Frontend:

1. HTML5(HyperText Markup Language)
2. JavaScript
3. CSS(Cascading Style Sheet)
4. Bootstrap

Backend:

1. PHP(Hypertext Preprocessor)
2. MySQL
3. XAMPP(X-operating system, Apache, Mysql, Php, Perl)

I D E(Integrated Developement Enviroment):

1. Visual Studio Code

## **Frontend:**

### HTML5:

**Какво е HTML5?**

***HTML5*** е модерен и все по-използван и навлизащ маркиращ език за структуриране и представяне на съдържание в ***интернет***.

Езикът добавя много нови синтаксови функции, като ***video, audio и canvas***, които подобряват работата с мултимедийно и графично съдържание.

***HTML5*** се използва често в приложенията за мобилни устройства, мобилни сайтове и операционни системи, като ***Firefox OS, Ubuntu Touch, Tizen и други.***

Групата разработчици ***WHATWG(Web Hypertext Application Technology Working Group)*** започва работа по новия стандарт през 2004 година. По това време, ***HTML 4.01*** не е бил обновяван от 2000-та година и организацията ***W3C(World Wide Web Consortium)*** се фокусира в бъдещата разработка на ***XHTML 2.0***.HTML кодът се поставя във файл с разширение ***\*.html***.На **фиг .1.1** е показан примерен код за извеждане на текста “Hello World!” в уеб страница, чрез ***HTML5***.

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta http-equiv="X-UA-Compatible" content="IE=edge">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>Document</title>

</head>

<body>

    <p>Hello World!</p>

</body>

</html>

**Фиг.1.1 –** Примерен код на ***HTML5***

### JavaScript:

**Какво е JavaScript?**

***JavaScript*** е един от най-използваните ***скриптови езици*** за програмиране в ***интернет***. Разработен е през 1995 година от ***Брендан Айк*** и осигурява моментното и динамично взаимодействие между ***потребител и браузър***. Днес смело може да заявим, че почти няма нещо, което ***JavaScript*** не може да направи. Благодарение на него уеб страниците, които използваме в момента, изпълняват много повече функции от това просто да зареждат данни.

**За какво се използва JavaScript?**

Функциите на ***JavaScript*** са се разширили значително през годините. Първоначално езикът просто е регулирал начинa, по който потребителят си взаимодейства с уебсайта: онлайн въпросници и форми за контакт, подаване на обратна връзка под формата на известия и важни съобщения и др. Днес обаче езикът е еволюирал до такава степен, че с него може дори да управлявате дрон. ***JavaScript*** кодът се поставя във файл с разширение ***\*.js***.На **фиг.1.2** е показан примерен код за извеждане на текста “Hello World!” в конзолата на уеб страница, чрез ***JavaScript***.

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta http-equiv="X-UA-Compatible" content="IE=edge">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>Document</title>

</head>

<body>

    <script>

        console.log("Hello World!");

    </script>

</body>

</html>

**Фиг.1.2** – Примерен код на ***JavaScript***

### CSS:

**Какво е CSS?**

***CSS*** е език за описване на презентацията и стиловете на елементите в един ***HTML/XML документ***. ***CSS*** е една от основните технологии, използвани в уеб, редом с ***HTML*** и ***JavaScript***.

При създаване на ***HTML*** страница, съдържанието й се описва с ***HTML*** код, а презентацията на това съдържание, тоест как ще изглежда то в браузъра, се описва с ***CSS*** код.

***CSS*** кодът се поставя във файл с разширение ***\* .css***. След това този файл може да се използва във всяка една страница от уеб сайта. Така че стилът ще се намира на едно място и при промяна ще се отразява на всички уеб страници. На **фиг.1.3** е показан примерен код за стилизация със ***CSS*** на параграф, в уеб страница, с текст “Hello World!”.

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta http-equiv="X-UA-Compatible" content="IE=edge">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>Document</title>

    <style>

        p{

            font-family: 'Times New Roman', Times, serif;

            color: red;

            font-size: 30px;

        }

    </style>

</head>

<body>

    <p>Hello World!</p>

</body>

</html>

**Фиг.1.3** – Примерен код на ***CSS***

### Bootstrap:

**Какво е Bootstrap?**

***Bootstrap*** е платформа, която е комбинация от ***HTML***, ***CSS*** и ***JavaScript*** код. ***Bootstrap*** напълно поддържа ***CSS3*** и ***HTML5***. Последеният факт е много ценен, тъй като ***CSS3*** и ***HTML5*** са стандарт за уеб дизайн през последните години. В интерес на истината – ***Bootstrap*** се превръща в стандарт за уеб разработки. Той е съвместим с всички основни ***уеб браузъри***. ***Bootstrap*** сайтовете се адаптират много лесно за всякакви устройства. Използвайки комбинация от ***JavaScript***, ***CSS*** и ***интелигентни мрежи***, ***Bootstrap*** позволява да се изпълни почти всеки елемент от дизайна, който може да се изиска. На **фиг.1.4** е показано как да вградим ***Bootstrap*** в дадена уеб страница чрез ***CDN (Content Delivery Network)***, което ни позволява да го използваме без да е необходимо сваляне.

<!-- Последно компилиран и минимизиран CSS -->  
<link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.4.1/css/bootstrap.min.css">  
  
<!-- jQuery библиотека -->  
<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.6.1/jquery.min.js"></script>  
  
<!-- Последно компилиран JavaScript -->  
<script src="https://maxcdn.bootstrapcdn.com/bootstrap/3.4.1/js/bootstrap.min.js"></script>

**Фиг.1.4 –** Вграждане на ***Bootstrap*** в уеб страница чрез

***CSS***, ***JQuery*** и ***JavaScript***

## **Backend:**

### PHP:

**Какво е PHP?**

***PHP*** е скриптов език. Използва се предимно в ***интернет*** среда за изпълнение на широк кръг от услуги.

***PHP*** се разпространява под отворен лиценз ***(PHP License),*** който позволява безплатно разпространяване на програмния код на интерпретатора на езика, както и създаването на производни интерпретатори под други лицензи с уговорката, че тези интерпретатори не могат да включват ***PHP*** в името си. Фактът, че ***PHP*** се разпространява свободно, го прави удачен избор за изграждане на уеб сървър, базиран изцяло на свободни продукти ***– GNU/Linux, Apache, MySQL/PostgreSQL*** и др.

***PHP*** файловете могат да съдържат текст, ***HTML***, ***CSS***, ***JavaScript*** и ***PHP*** код и имат разширение ***\*.php***.На **фиг.1.5** е показан примерен код на ***PHP*** за извеждането на текста “Hello World!” в уеб страница.

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta http-equiv="X-UA-Compatible" content="IE=edge">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>Document</title>

</head>

<body>

    <h1>

        <?php

        echo 'Hello, World!';

        ?>

    </h1>

</body>

</html>

**Фиг.1.5** – Примерен код на ***PHP***

### MySQL:

**Какво е MySQL?**

***MySQL***, система за управление на релационни ***бази данни*** с отворен код ***(RDBMS),*** съществува от 1995 г. Създадена е от ***MySQL AB***, която по-късно става ***Oracle*** ***Corporation***. Софтуерът използва ***SQL*** като основен език за данни и съхранява ***данни*** в таблици на дисковото устройство на сървъра. Данните могат да се съхраняват свободно в определени граници или обвързани със схема, която определя как трябва да бъдат структурирани.

***MySQL*** е софтуер с отворен код и може да бъде изтеглен безплатно. Няма ограничения за броя ***сървъри***, ***потребители*** или ***бази данни***, които можете да създадете с него, докато абонаментът ви остава активен.

***MySQL*** е популярна ***RDBMS*** за уеб приложения, защото е лесна за използване, бърза и надеждна. Освен това софтуерът има голяма потребителска общност и са налични много опции за търговска поддръжка. ***MySQL*** също така предлага голяма гъвкавост, като ви позволява да съхранявате данни в различни формати и да дефинирате свои собствени схеми.На **фиг.1.6** са показани примерни заявки за създаване на таблица и съответно добавяне на стойности към нея в ***MySQL***.

CREATE TABLE table\_name (  
    column1 datatype,  
    column2 datatype,  
    column3 datatype,  
   ....  
);

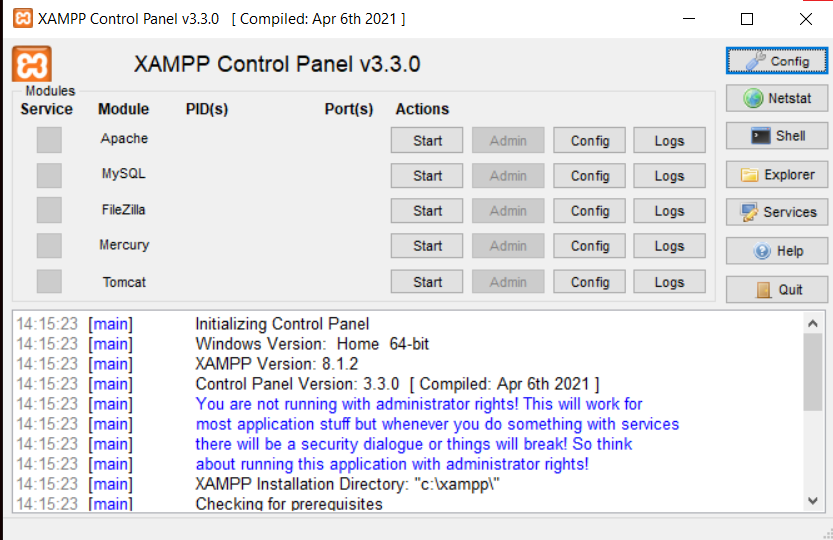
INSERT INTO table\_name (column1, column2, column3, ...)  
VALUES (value1, value2, value3, ...);

**Фиг.1.6** – Примерни заявки в ***MySQL***

### XAMPP:

**Какво е XAMPP?**

***XAMPP*** е съкращение за междуплатформени, ***Apache, MySQL, PHP*** и ***Perl*** и ви позволява да създавате сайт офлайн, на локален уеб сървър на вашия компютър. Това просто и леко решение работи на ***Windows, Linux*** и ***Mac*** – оттук и ***„кросплатформената“*** част. Може да бъде срещнато и под наименование ***WEB-Stack***, което не е синоним на конкретното използвания набор, а по-скоро обобщаващо наименование за всички пакети от подобен вид.На **фиг.1.7** е представен интерфейсът на ***XAMPP*** контрол панелът.



**Фиг.1.7** – Интерфейсът на ***XAMPP*** контрол панела

Тъй като не е самостоятелно приложение, ***XAMPP*** предоставя два основни компонента за неговото инсталиране – ***Apache***, който се използва за създаване на локалния сървър, и ***MySQL***, който можете да използвате като база данни за вашия уебсайт.

Може би се чудите защо и как разработчиците използват локален сървър . Отговорът е прост – позволява им да създадат локално копие на сайта, в което могат да изпробват нови актуализации на плъгини, преди да ги внедрят в неговата версия на живо. По този начин те могат да предотвратят и открият потенциални грешки и проблеми, които могат да възникнат.

**Какво е Apache?**

***Apache*** е най-широко използваният софтуер за уеб сървър. Разработен и поддържан от ***Apache Software Foundation***, ***Apache*** е софтуер с отворен код, достъпен безплатно. Той работи на 67% от всички уеб сървъри в света. Той е бърз, надежден и сигурен. Той може да бъде силно персонализиран, за да отговори на нуждите на много различни среди чрез използване на разширения и модули.

Така че основно уеб сървърът е софтуерът, който получава вашата заявка за достъп до уеб страница. Той изпълнява няколко проверки за сигурност на вашата HTTP заявка и ви отвежда до уеб страницата. В зависимост от страницата, която сте заявили, страницата може да поиска от сървъра да стартира няколко допълнителни модула, докато генерира документа, който да Ви служи. След това Ви връчва искания от вас документ.

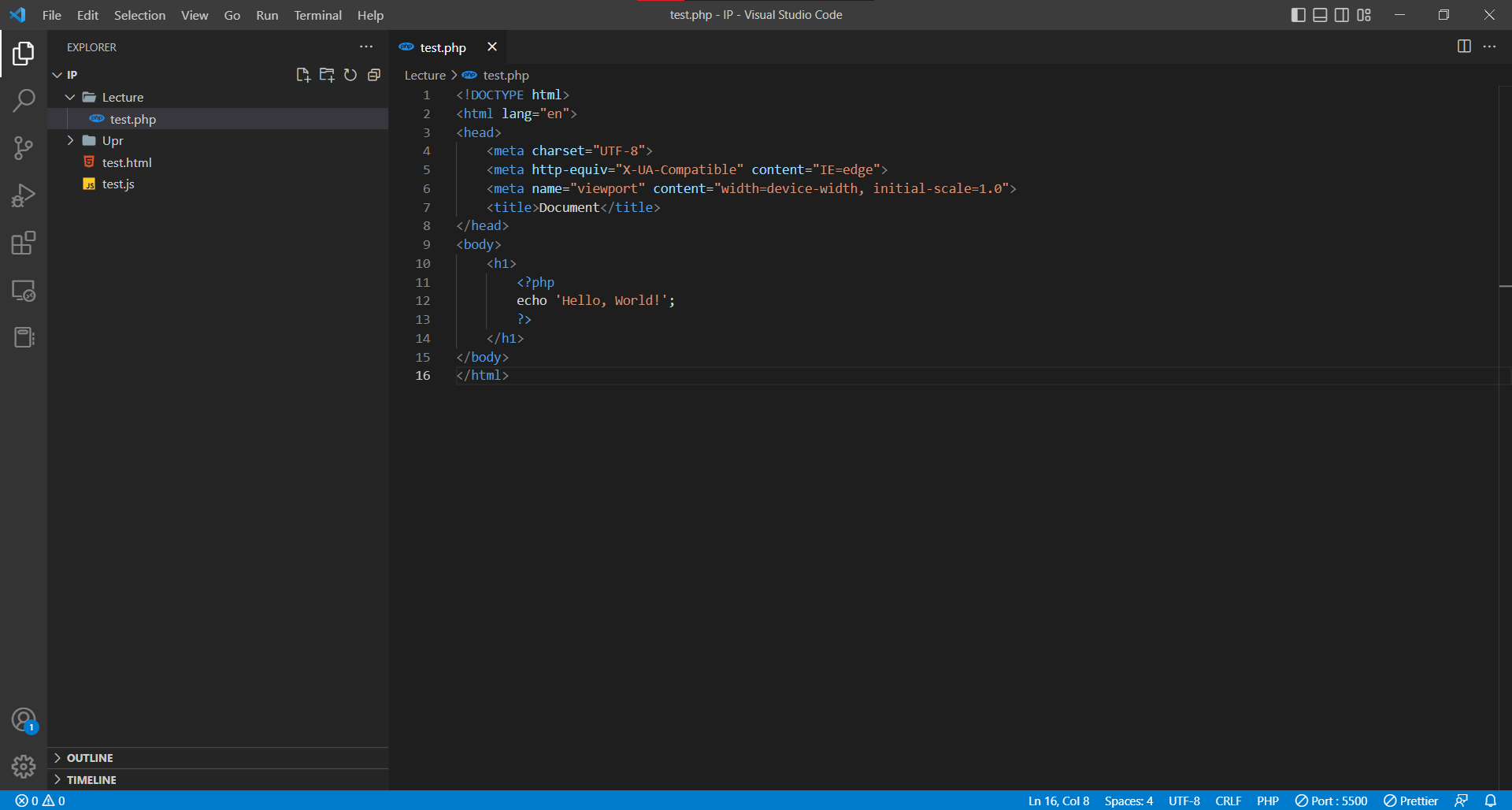
## **I D E:**

### Visual Studio Code:

**Какво е Visual Studio Code?**

***Visual Studio Code*** е редактор на програмен код за ***Windows, Linux*** и ***OS X***.Това е първият редактор на ***Microsoft***, който може да се ползва под ***Linux*** и ***macOS***. Поддържа богат набор от инструменти за разработване като ***дебъгване***, вграден ***Git Control, IntelliSense,*** и др. Той също така дава възможност за персонализиране, което означава, че потребителите могат да променят темата на редактора, клавишните комбинации, настройките и др. Редакторът е е безплатен и публично достъпен за преглед.

***Visual Studio Code*** е базиран на ***Electron***, който е базиран на ***Chromium***, използван да разгръща ***io.js*** приложения за десктопа. ***Visual Studio Code*** използва ***Blink layout engine***, за интерфейс. На **фиг.1.8** е представен потребителският интерфейс на ***Visual Studio Code*.**



**Фиг.1.8** – Потребителският интерфейс

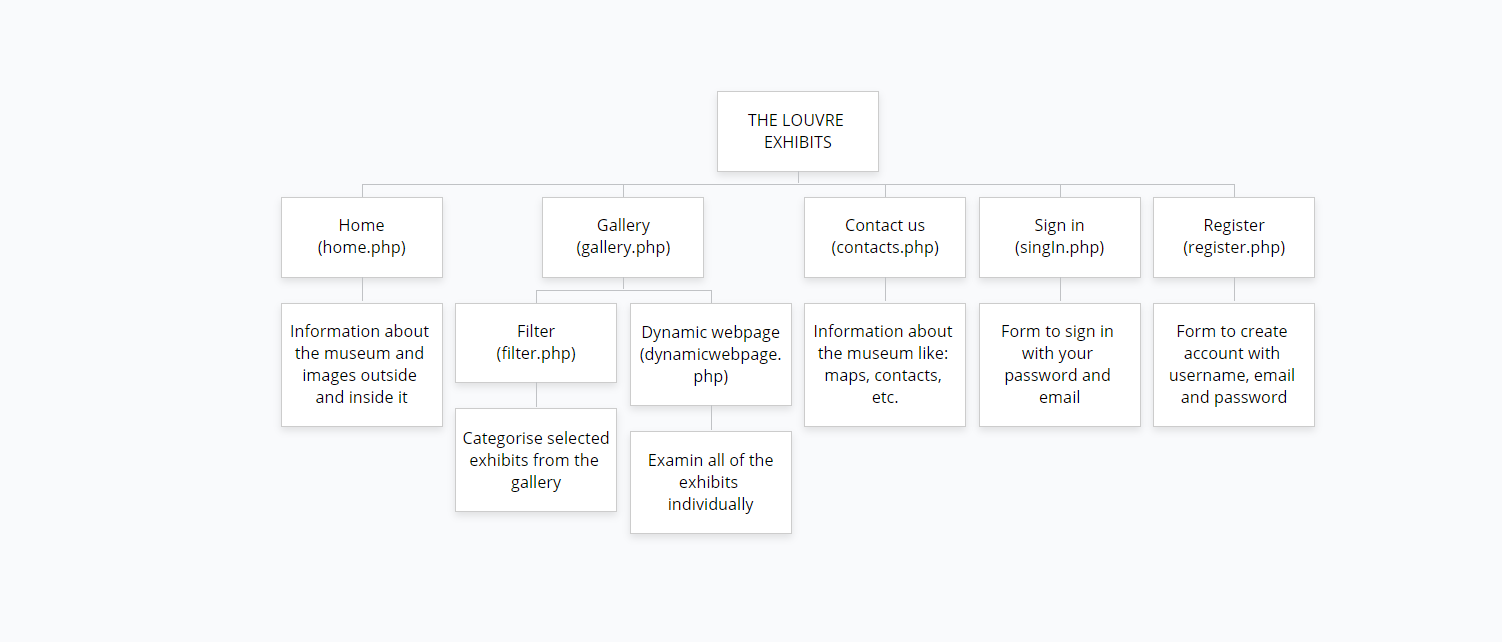
на ***Visual Studio Code***

# **Глава 2 Теоретична част**

1. Структура на директориите

на проекта:

Проектът е разположен в директория, като в нея се съдържа отделните файлове отговарящи за неговата работа, както следва :

В избрана от потребителя папка /директория/ на компютъра се намира всяка една страница от менюто на уебсайта. На **фиг.2.1** е представен ***Sidemap*** на проекта.

**Фиг.2.1** –***Sidemap*** на проекта

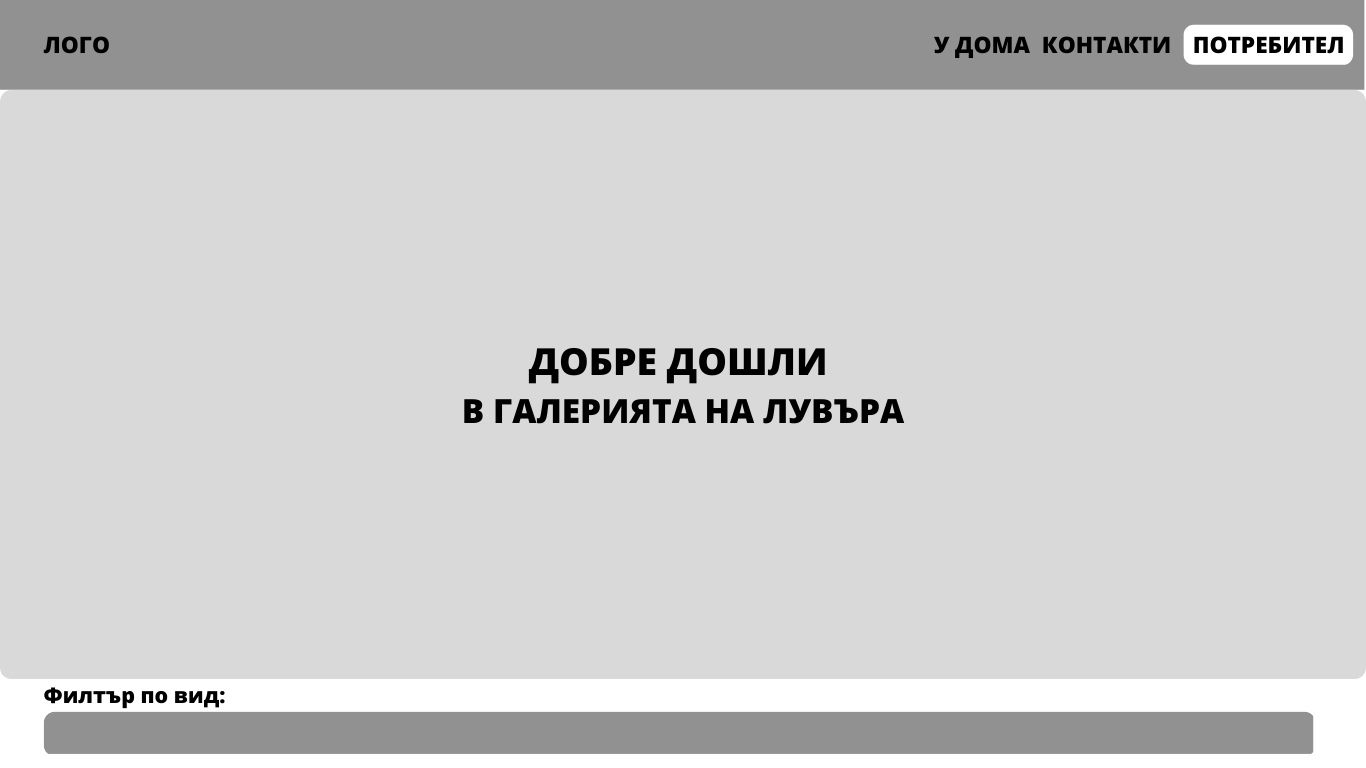
1. Структура на директориите

на проекта:



**Фиг.2.2** – ***Home page(home.php)***

Първата страница на уебсайта – ***home.php* (фиг.2.2).** Съдържа най-различни снимки на Лувъра,както и информация за него. В полето отгоре има бутони за отвеждане до галерията.

****

**Фиг.2.3 – *Gallery(gallery.php)***

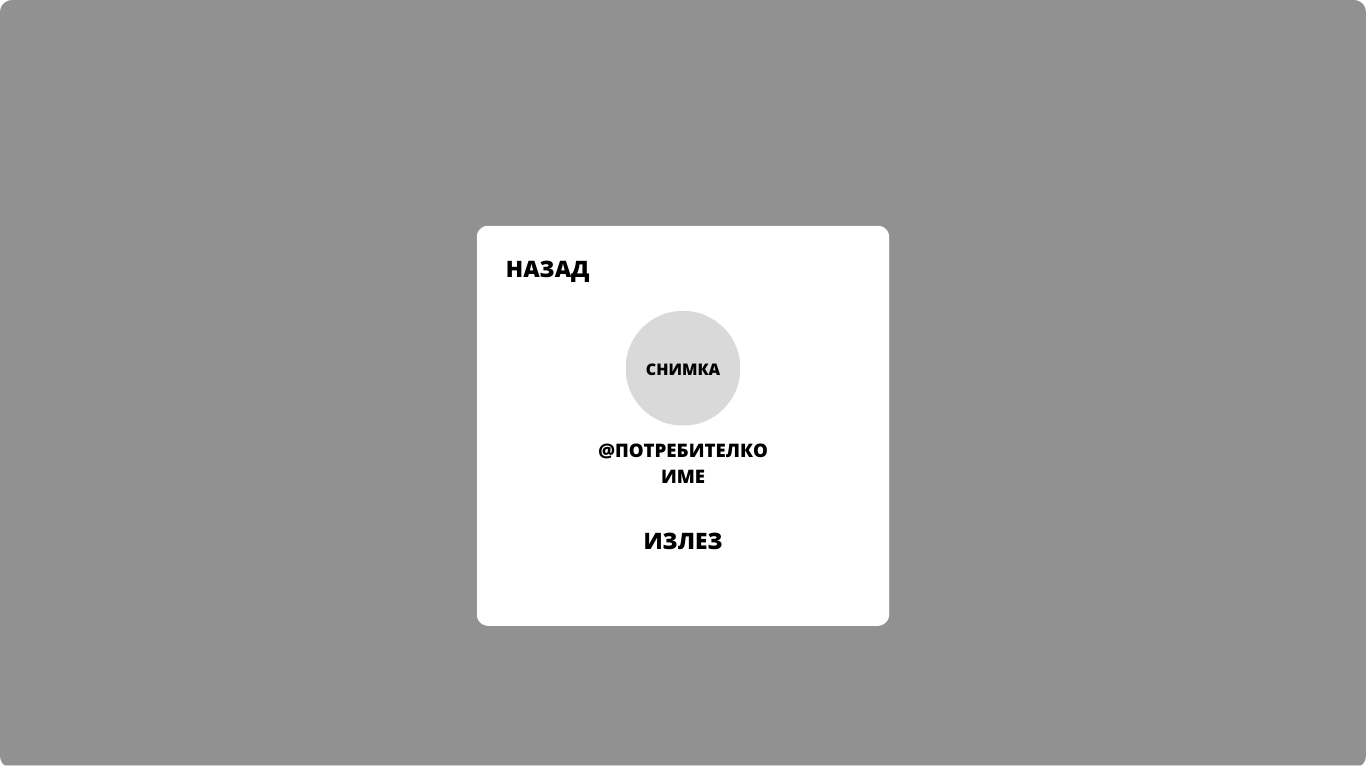
Втората страница на проекта е ***gallery.php* (фиг.2.3).** По-надолу тя съдържа снимки на експонати от Лувъра, върху които потребителят може да кликне, за да види повече информация за експоната.

****

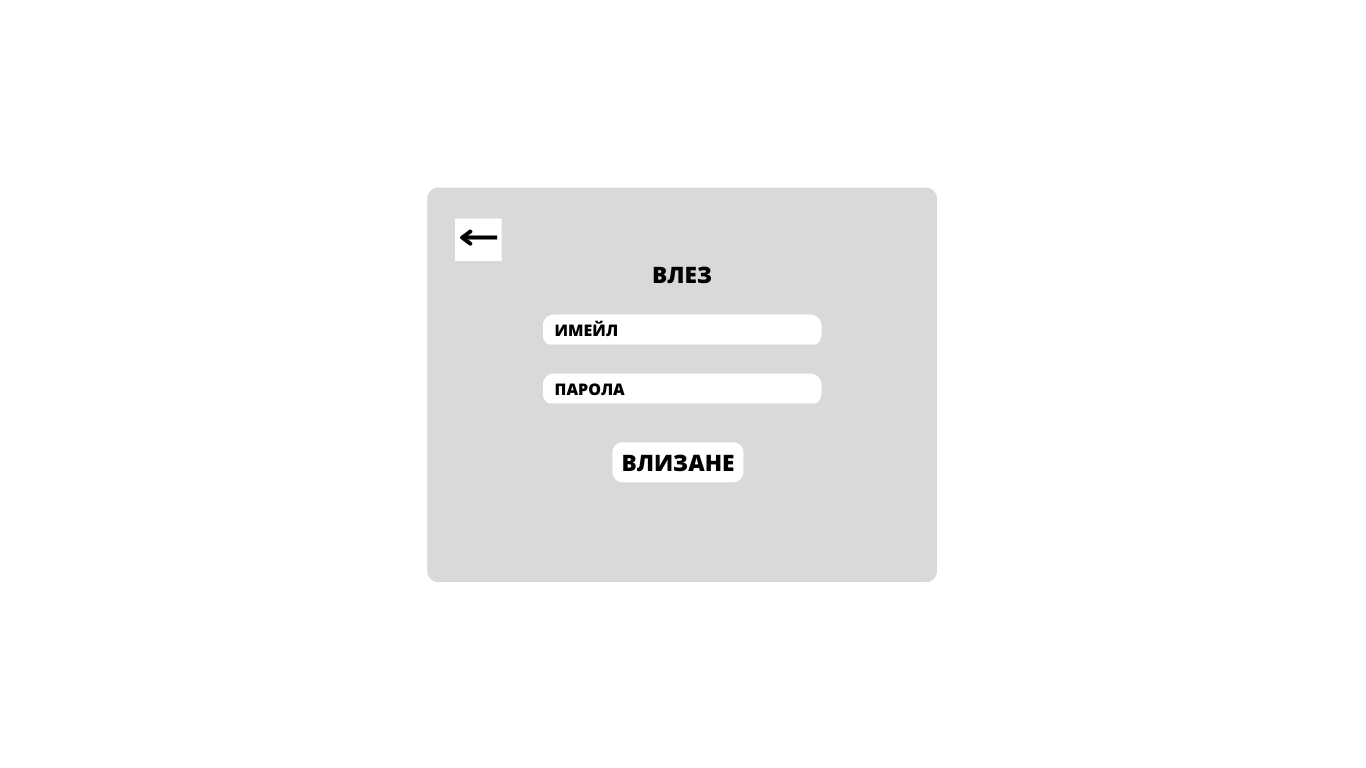
**Фиг.2.4 – *Contacts(contacts.php)***

Третата страница от проекта е ***contacts.php* (фиг.2.4).**Тя съдържа допълнителна информация като контакти за връзка на музея и т.н.

**Фиг.2.5 – *User Profile(userProfile.php)***

****На **фиг.2.5** може да видите страницата ***userProfile.php***. В нея потребителят може да види своя профил и да излезе от акаунта си.

**Фиг.2.6 – *Register(register.php)***

****На **фиг.2.6** може да видите страницата ***register.php.*** В нея потребителят може да влезе в профила си с имейл и парола. След което да натисне бутона за влизане, за да влезе в акаунта си.

**Фиг.2.7 – *Sign In(signIn.php)***

****На **фиг.2.7** може да видите страницата ***signIn.php.*** В него потребителят може да се регистрира с потребителско име, имейл и парола. След което да натисне бутона за регистриране, за да се запази информацията за него в базата данни.

# **Глава 3 Инженерно решение на поставената задача**

1. Основни функции на кода

В главната папка на проекта се намират основните файлове, отговорни за работата на уебсайта, а именно:

* home.php;
* gallery.php
* dynamicwebpage.php
* signin.php
* register.php
* logout.php
* signInProcess.php
* navigation.php
* footer.php
* userProfile.php

1. Структура на базата данни

и таблиците в нея:

Базата данни на проекта се наричa **`thelouvre`** и съдържа следните таблици и полета във всяка от тях, както следва :

Име на таблицата: user

|  |  |  |
| --- | --- | --- |
| Име: | Тип: | Забележка: |
| userId | INT | Id на потребителя |
| name | VARCHAR | Име на потребителя |
| email | VARCHAR | Имейл на потребителя |
| pass | VARCHAR | Парола на потребителя |
| userType | VARCHAR | Роля на потребителя |

Име на таблицата: place

|  |  |  |
| --- | --- | --- |
| Име: | Тип: | Забележка: |
| Id | INT | Id на мястото |
| Room | VARCHAR | Стая на експонат |
| Floor | VARCHAR | Етаж на експонат |
| ExhibitsId | INT | Id на експонат |

Име на таблицата: exhibits

|  |  |  |
| --- | --- | --- |
| Име: | Тип: | Забележка: |
| Id | INT | Id на експонат |
| Name | VARCHAR | Име на експонат |
| Category | VARCHAR | Категория на експонатите |
| Place | VARCHAR | Id на мястото |
| MadePlace | VARCHAR | Страна на създаване |
| FilePath | VARCHAR | Пътят към снимка на експоната |

След като сте инсталирали XAMPP, следваме стъпките по долу :

1. Отворете браузър и отидете на URL <http://localhost/phpmyadmin/>.
2. След това щракнете върху раздела бази данни.
3. Създайте база данни с име “thelouvre” и след това щракнете върху раздела за импортиране.
4. Щракнете върху преглед на файла и изберете файл”thelouvre.sql”.
5. Щракнете върху Go.

## **SQL CREATE заявки за**

## **създаване и попълване на таблици**

Заявка за създаването на таблицата ***user***:

**CREATE** **TABLE** **user**(

userId **INT** **PRIMARY** **KEY** **AUTO\_INCREMENT**,

**name** **VARCHAR**(255),

email **VARCHAR**(255),

pass **VARCHAR**(255),

userType **VARCHAR**(255)

);

Примерна заявка за въвеждане на потребител в таблицата user:

**INSERT** **INTO** `user` (`name`, `email`, `pass`, `userType`) **VALUES**

('Потребител', 'user@mail.bg', '1234567890', 'user');

Заявка за създаването на таблицата ***exhibits***:

**CREATE** **TABLE** **user**(

id **INT** **PRIMARY** **KEY** **AUTO\_INCREMENT**,

**Name** **VARCHAR**(255),

Category **VARCHAR**(255),

Place **VARCHAR**(255),

madePlace **VARCHAR**(255),

FilePath **VARCHAR**(255)

);

Примерна заявка за въвеждане на потребител в таблицата ***exhibits***:

**INSERT** **INTO** `exhibits ` (`Name`, `Category `, `Place `, `MadePlace`, `FilePath`) **VALUES** ('The Mona Lisa', 'Portrait', '1', 'Italy', '../images/mona\_lisa.jpg');

Заявка за създаването на таблицата ***exhibits***:

**CREATE** **TABLE** **user**(

id **INT** **PRIMARY** **KEY** **AUTO\_INCREMENT**,

**Name** **VARCHAR**(255),

Category **VARCHAR**(255),

Place **VARCHAR**(255),

madePlace **VARCHAR**(255),

FilePath **VARCHAR**(255)

);

Примерна заявка за въвеждане на потребител в таблицата ***exhibits***:

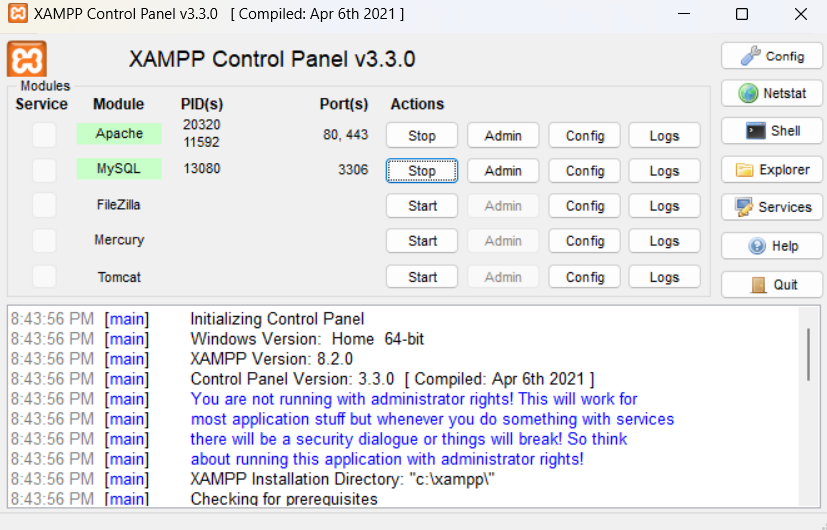
**INSERT** **INTO** `place ` (`Id`, `Room`, `Floor `, `ExhibitsId `) **VALUES** ('1', '1', '1', '1');

# **Глава 4 Анализ на получените резултати, приложимост и изводи**

## 

## **Работа на уебсайта**

За стартиране на уебсайта първоначално трябва да се стартират всички нужни сървъри през интерфейса на ***XAMPP***, а те са **фиг.4.1** : ***Apache Web Server*** и ***MySQL Database***.



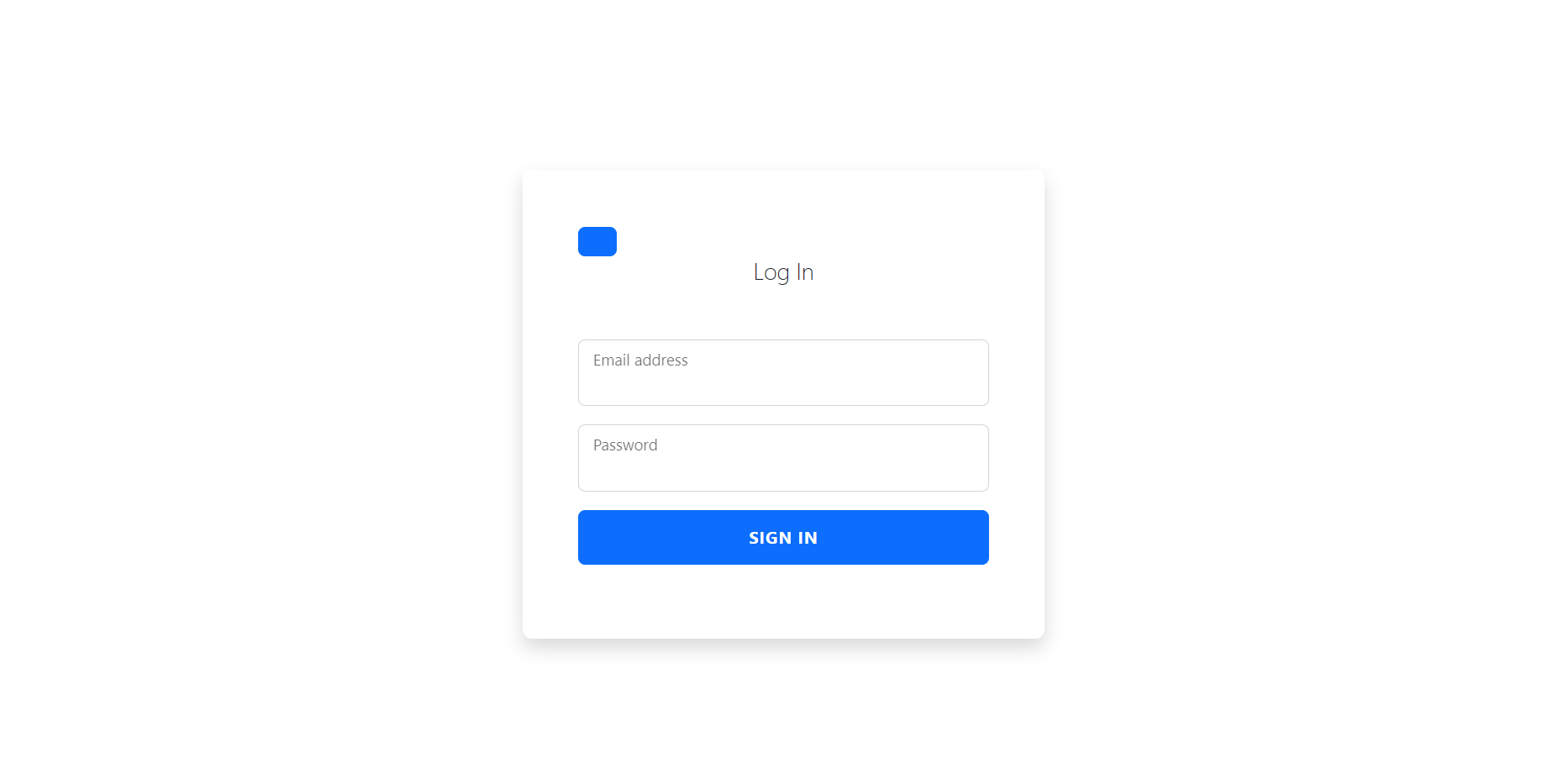
**Фиг.4.1** – Интерфейс на***XAMPP Control Panel***

След стартиране на уебсайта се показва следния екран:

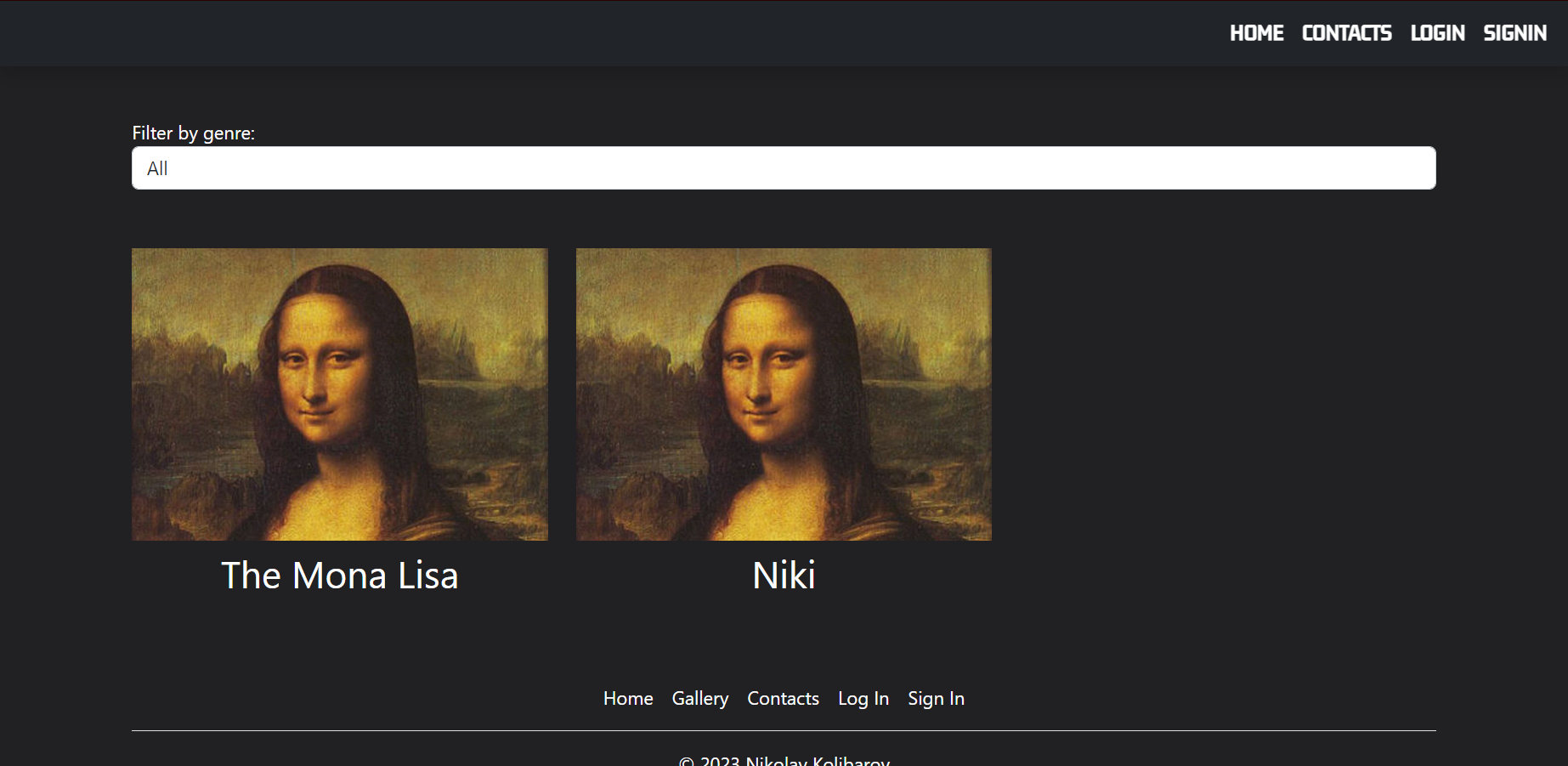


Това е началния екран. Отгоре има възможност за (***LOGIN*** и ***SIGNIN***). За вписване се влиза в ***LOGIN***. Ако потребителят е нов ***SINGIN*** бутонът му дава възможност за създаване на акаунт, а след това трябва да се впише.

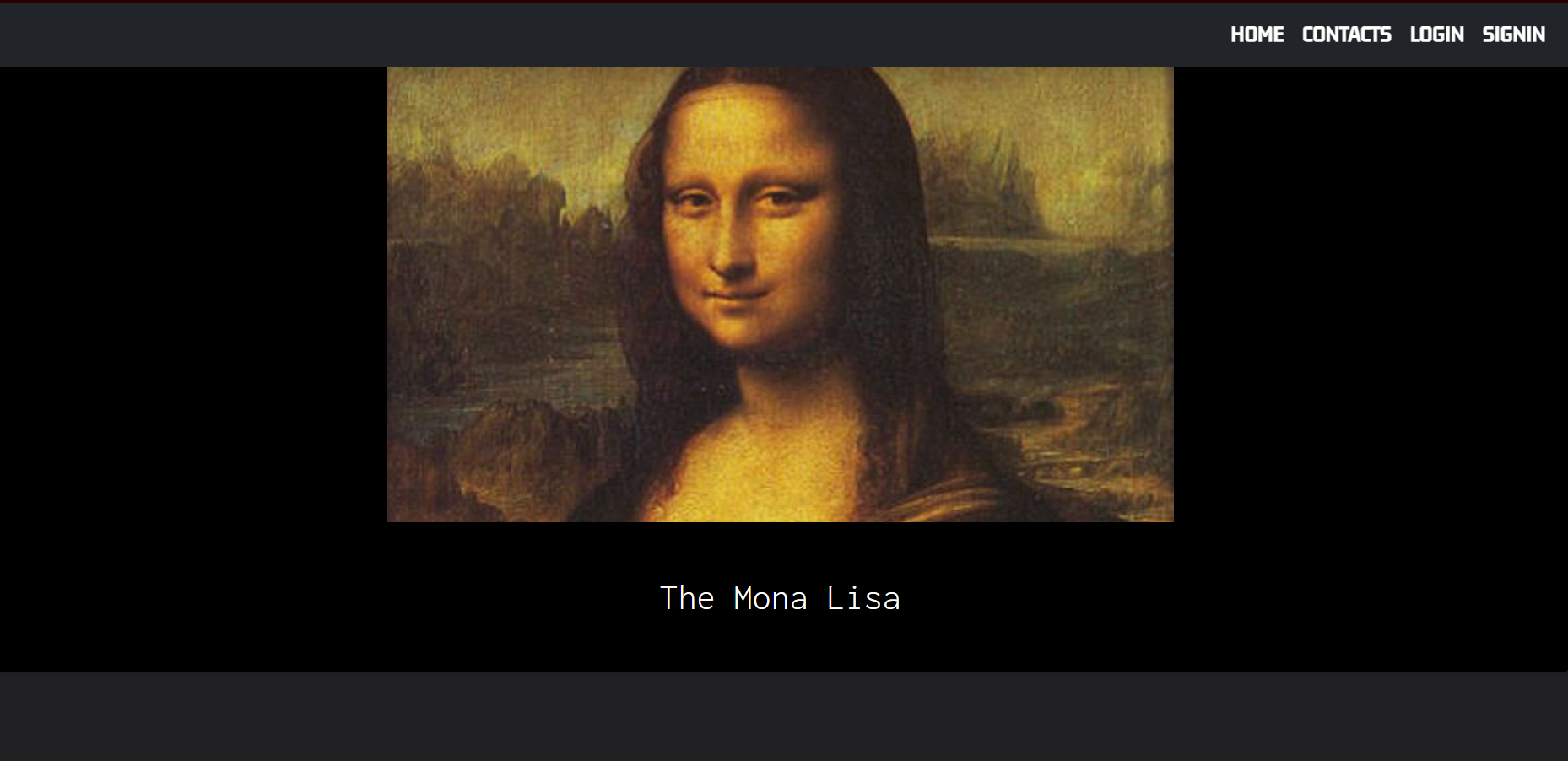
Така изглежда LOG IN формата:



След влизане в акаунта отново се показва началния екран, като потребителят може да влезе в галерията и да разгледа експонатите.



При натискането на снимката на някой от експонатите се отваря ***dynamic web page***, чрез които се отваря снимката в нов прозорец.



## **Заключение**

Работата с приложението е изключително улеснена и интуитивна. Подходящо са избрани всички необходими технологии, за да се получи едно стабилно и бързо работещо проложение.

## Използвана литература

(1 страница)

Книга, статия, документация и линк. Тук е дадено примерно описание. Номерът на референцията да се свързва с текста. В Word това се прави от References > Citation and Bibliography.

[1] Bennett, C.J. (**2011**). [Privacy advocacy from the inside and the outside: implications for the politics of personal data protection in networked societies](http://www.tandfonline.com/doi/abs/10.1080/13876988.2011.555996). *Journal of Comparative Policy Analysis*, vol. 13, No.2, pp.125-141.

[2] Rana, S., Pr. Kumar Joshi (**2012**). Risk analysis in web applications by using cloud computing. *International Journal of Multidisciplinary Research*, vol. 2, No.1, pp. 386-394.

[3] Morris F., B. Harrison (**2013**). Digital Signatures Beyond Security. In: *Digital Signatures*, McGraw Hill Publishing, Sydney, pp. 203-217.

[4] EC (**2010**). *Europe 2020: a new economic strategy* (*available at*: <http://ec.europa.eu/eu2020/>)

[5] Friedewald, K., D. E. Moriani (**2012**). Privacy and data protection in the global network. *Proceedings of the International Conference on Globalization*, 23-25 June, France, vol. 1, pp.61-67.

[6] Subramanian, A., M. Kessler (**2013**). The hyperglobalization of trade and its future. *Global Citizen Foundation*, June (*available at:* <http://www.gcf.ch/wp-content/uploads/>...).

[7] Olden, E. (**2011**). Architecting a cloud-scale identity fabric. *Computer*, March, pp.52-59

[8] Song, D., E. Shi, I. Fisher, V. Shankar (**2012**). Cloud data protection for the masses. *Computer*, January, pp.39-45.

## Списък с използваните означения и съкращения

(1 страница)

HTML – Hyper Text Markup Language

CSS - Cascading Style Sheets

PHP – Personal Home Page

SQL - Structured Query Language

HTTPS - Hypertext Transfer Protocol Secure

URL - Uniform Resource Locator

OOP - Object-oriented programming

MVC – model view controller

При форматирането на текста се спазват следните параметри:

* Обем на дипломния проект – минимум 30 страници
* Формат: А4; Брой редове в стр.: 30; Брой на знаците: 60 знака в ред
* Общ брой на знаците в 1 стр.: 1800 – 2000 знака
* Шрифт: Times New Roman, 12, разстояние между редовете 1,5