

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ**  
**Сумський державний університет**  
Факультет електроніки та інформаційних технологій  
Кафедра комп'ютерних наук

«До захисту допущено»

В.о. завідувача кафедри

Ігор ШЕЛЕХОВ

\_\_\_\_\_ (підпис)

червня 2024 р.

---

**КВАЛІФІКАЦІЙНА РОБОТА**  
**на здобуття освітнього ступеня бакалавр**

зі спеціальності 122 - Комп'ютерні науки,  
освітньо-професійної програми «Інформатика»  
на тему: «Електронна бібліотека з можливістю реєстрації та завантаженням  
книг»  
здобувача групи ІН - 06 Бережного Владислава Олександровича

Кваліфікаційна робота містить результати власних досліджень.  
Використання ідей, результатів і текстів інших авторів мають посилання на  
відповідне джерело.

Владислав БЕРЕЖНИЙ

\_\_\_\_\_ (підпис)

Керівник,  
старший викладач кафедри  
комп'ютерних наук, к.ф.-м.н.

Анна БАДАЛЯН

\_\_\_\_\_ (підпис)

**Суми – 2024**

**Сумський державний університет**  
Факультет електроніки та інформаційних технологій  
Кафедра комп'ютерних наук

«Затверджую»

В.о. завідувача кафедри

Ігор ШЕЛЕХОВ

(підпис)

**ІНДИВІДУАЛЬНЕ ЗАВДАННЯ НА КВАЛІФІКАЦІЙНУ РОБОТУ**  
**на здобуття освітнього ступеня бакалавра**

зі спеціальності 122 - Комп'ютерних наук, освітньо-професійної програми «Інформатика»  
здобувача групи ІН-06 Бережного Владислава Олександровича

1. Тема роботи: «Інформаційна технологія прогнозування курсу валют»  
затверджую наказом по СумДУ від «22» квітня 2024 р. № 0414-VI
2. Термін здачі здобувачем кваліфікаційної роботи до
3. Вхідні дані до кваліфікаційної роботи \_\_\_\_\_
4. Зміст розрахунково-пояснювальної записки (перелік питань, що їх належить розробити)  
1) Аналіз проблеми предметної області, постановка й формування завдань дослідження.  
2) Огляд технологій, що використовуються для прогнозування курсу валют. 3) Розробка інтелектуальної системи з прогнозування курсу валют. 4) Аналіз результатів.
5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень) \_\_\_\_\_
6. Консультанти до проекту (роботи), із значенням розділів проекту, що стосується їх

Розділ	Консультант	Підпис, дата	
		Завдання видав	Завдання прийняв

7. Дата видачі завдання «06» травня 2024 р.

Завдання прийняв до виконання \_\_\_\_\_  
(підпис)

Керівник \_\_\_\_\_  
(підпис)

**КАЛЕНДАРНИЙ ПЛАН**

№ п/п	Назва етапів кваліфікаційної роботи	Термін виконання	Примітка
1	<i>Аналіз предметної області та аналогів, постановка й формування завдань.</i>	<b>06.05.24-08.05.24</b>	
2	<i>Вибір технологій для виконання поставлених завдань</i>	<b>09.05.24-11.05.24</b>	
3	<i>Розробка прототипу веб-сторінки</i>	<b>12.05.24-17.05.24</b>	
4	<i>Виклад вибраного програмного засобу рішення поставленої задачі</i>	<b>18.05.24- 31.05.24</b>	

Здобувач вищої освіти \_\_\_\_\_  
(підпис)

Керівник \_\_\_\_\_  
(підпис)

## АНОТАЦІЯ

**Записка:** 71 стр., 18 рис., 2 додаток, 10 використаних джерел.

**Обґрунтування актуальності теми роботи** – Тема кваліфікаційної роботи є актуальною, оскільки вона надає швидкий доступ до різних інформаційних ресурсів, має ефективний пошук цих ресурсів, фільтри по жанрам та привертає увагу суспільства до різних витворів літератури, які можна переглянути у будь-якому місці.

**Об’єкт дослідження** — електронні бібліотеки.

**Мета роботи** — розробка електронної бібліотеки та надання доступу на завантаження книг авторизованим користувачам.

**Методи дослідження** — аналіз та огляд літератури, вивчення наявних досліджень, статей та книг з теми електронних бібліотек для виявлення поточних тенденцій, проблем та досягнень.

**Результати** — розроблено веб-сайт як інтернет-бібліотека книг. За допомогою технологій: php, javascript, html та css. Насамперед призначений для користувачів зацікавлених в книгах. За допомогою навігаційної панелі та гіперпосилань можна дізнатися важливу інформацію про книгу, наприклад, автора, якого року відбулась публікація, жанр і т.д. Була створена база даних з якої можна дістати та записати дані. Реалізовано систему реєстрації та авторизації, після якої користувач зможе завантажити на свій комп’ютер книгу в форматі pdf.

ЕЛЕКТРОННІ БІБЛІОТЕКИ, БАЗА ДАНИХ, ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ,  
КНИГИ, MYSQL.

## Зміст

Вступ.....	5
1. Інформаційний огляд.....	6
1.1 Дослідження предметної галузі.....	6
1.2 Дослідження аналогів.....	7
1.3 Постановка задачі.....	9
2. Вибір методів розв’язання задачі.....	10
2.1 Огляд технологій для розробки.....	10
3. Програмна реалізація.....	12
3.1 Розробка інтерфейсу.....	12
3.2 Розробка бази даних.....	22
Висновок.....	26
Список використаних джерел.....	27
Додаток А.....	28
Додаток Б.....	57

## Вступ

**Обґрунтування вибору теми роботи.** Тема роботи була обрана тому що, електронна бібліотека має значні переваги над традиційною версією. Користувач зможе зайти з любого місця і в любий час за умови доступу до інтернету, час на пошук необхідної літератури значно зменшується. Електронна бібліотека дозволяє завантажити файл та передивитися його в умовах відсутності інтернету.

**Актуальність.** Тема роботи є актуальною, оскільки вона надає швидкий доступ до різних інформаційних ресурсів, має ефективний пошук цих ресурсів, фільтри по жанрам та привертає увагу суспільства до різних витворів літератури, які можна переглянути у будь-якому місці.

**Об'єкт дослідження.** Електронні бібліотеки.

**Предмет дослідження.** Інформація в електронному вигляді.

**Гіпотеза.** Я вважаю, що електронні бібліотеки значно покращать доступ до знань і інформації, підвищать ефективність навчання та дослідницької діяльності, а також сприяють збереженню культурної спадщини, при цьому знизять вплив на довкілля традиційних бібліотек.

**Новизна.** У моїй роботі використовується простий и зручний інтерфейс, адаптація під більшість пристроїв, наявна панель новин на головній сторінці. Є інструменти для збору інформації щодо вподобань користувачів.

**Структура.** Дане робота складається зі вступу, інформаційного огляду, вибір методів розв'язання задачі, програмної реалізації, висновків, списку використаних джерел та додатків.

## 1. Інформаційний огляд

### 1.1 Дослідження предметної галузі

Сучасний період розвитку суспільства переживає великий перелом через сильний вплив інформаційних технологій. Інформаційний простір стає все більш глобальним, а інтернет стає не тільки засобом комунікації, але й потужним інструментом доступу до знань та інформації. З появою інтернет-бібліотек, доступ до книг став набагато простішим та зручнішим. Тепер читачі можуть переглядати різноманітні твори із зручності власного дому та навіть ознайомлюватися з книжками у будь-який час доби. Це особливо важливо для тих, хто має обмежений доступ до традиційних бібліотек або мешкає в віддалених регіонах.

Електронні бібліотеки - це онлайн-ресурси, які містять велику кількість інформації в електронному форматі, доступну для користувачів через Інтернет. Вони включають в себе книги, наукові статті, журнали, відео та інші види контенту. Електронні бібліотеки стали важливим інструментом для зберігання, пошуку та поширення знань у сучасному світі.

Інформатизація книг також дозволяє швидше знаходити необхідну літературу завдяки зручним інструментам сортування та пошуку. Користувачам залишається лише клацнути мишею або ввести відповідне запитання, і система швидко видає результати, що відповідають їхнім потребам[1]. Це заощадить багато часу та зусиль у пошуку інформації. Окрім того, інтернет-бібліотеки допомагають розширити асортимент доступних книжок. Багато таких бібліотек мають широкий вибір творів різних авторів та жанрів, включаючи класику, сучасну літературу, наукові дослідження, і багато іншого. Це дозволяє користувачам знаходити цікаву літературу навіть у ситуаціях, коли традиційні бібліотеки не мають необхідного видання на полицях. Необхідно зазначити, що з усім цим прогресом важливо також забезпечити збереження культурної

спадщини та прав авторів. Захист авторських прав та стимулювання творчості є ключовими завданнями для розвитку інтернет-бібліотек.

Електронні бібліотеки також забезпечують можливість зберігання та збереження інформації. Вони вимагають менше простору, ніж паперові книги та журнали, і дозволяють зберігати велику кількість даних на серверах. Це допомагає у збереженні ресурсів та зменшенні впливу на довкілля. Крім того, електронні бібліотеки забезпечують можливість спільного використання інформації. Користувачі можуть легко ділитися ресурсами з іншими користувачами, надаючи доступ до певних матеріалів або створюючи спільні проекти та колекції. Це сприяє обміну знаннями та співпраці. Однак разом з численними перевагами, електронні бібліотеки також стикаються з рядом викликів і проблем. Однією з найбільших проблем є доступність контенту. Не всі ресурси доступні безкоштовно, і багато електронних книг та журналів можуть бути доступні лише за плату. Це може створити нерівність у доступі до інформації. Крім того, існує питання щодо якості і достовірності інформації в електронних бібліотеках. Завдяки відсутності редакційного контролю, може бути важко визначити, наскільки надійні деякі джерела. Користувачі повинні бути критичними до інформації, яку вони знаходять у мережі.

Загалом, електронні бібліотеки є важливим інструментом у сучасному інформаційному суспільстві. Вони забезпечують доступ до великої кількості інформації та дозволяють користувачам зберігати, пошукувати і спільно використовувати ресурси. Однак їх використання вимагає уваги до питань доступності, якості та правових аспектів. Дослідження предметної галузі електронних бібліотек може сприяти розвитку цієї важливої галузі та покращенню доступу до інформації для всіх користувачів.

## **1.2 Дослідження аналогів**

Однією з аналогів електронної бібліотеки є веб-сайт «Лібрук», який надає користувачам доступ до різної літератури для читання, має систему пошуку,

реєстрації та фільтрації. Всю літературу на сайті можливо завантажити, є зручне меню та інструменти статистики[3] (лічильник переглядів та лічильник вподобань). Головну сторінку зображено на рисунку 1.1.

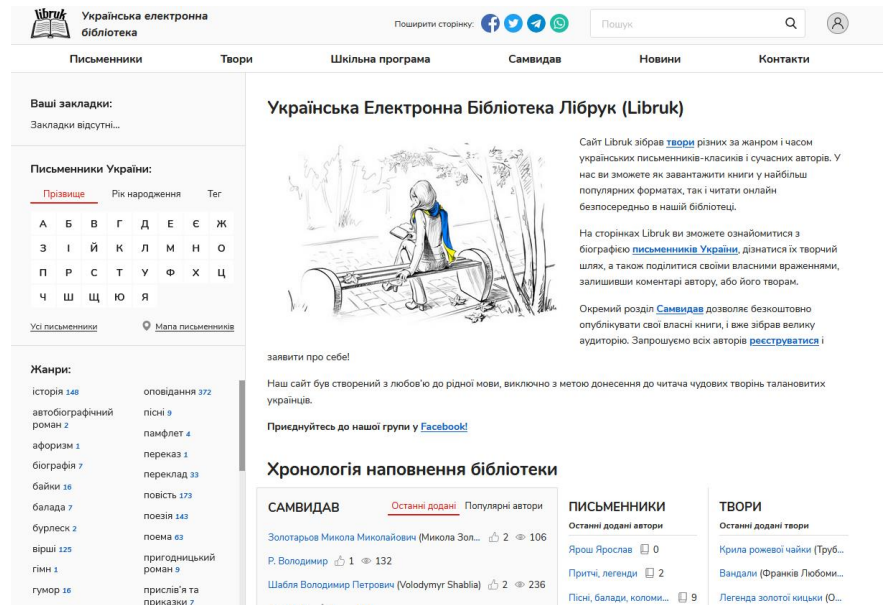


Рисунок 1.1 – Головна сторінка сайту «Лібрук»

Наступним аналогом слугуватиме електронна бібліотека «Читака». На цьому веб-сайті вже відсутня реєстрація та можливість перегляду з завантаженням книг[4]. Функції пошуку та лічильників присутні. Головна сторінка відображена на рисунку 1.2.

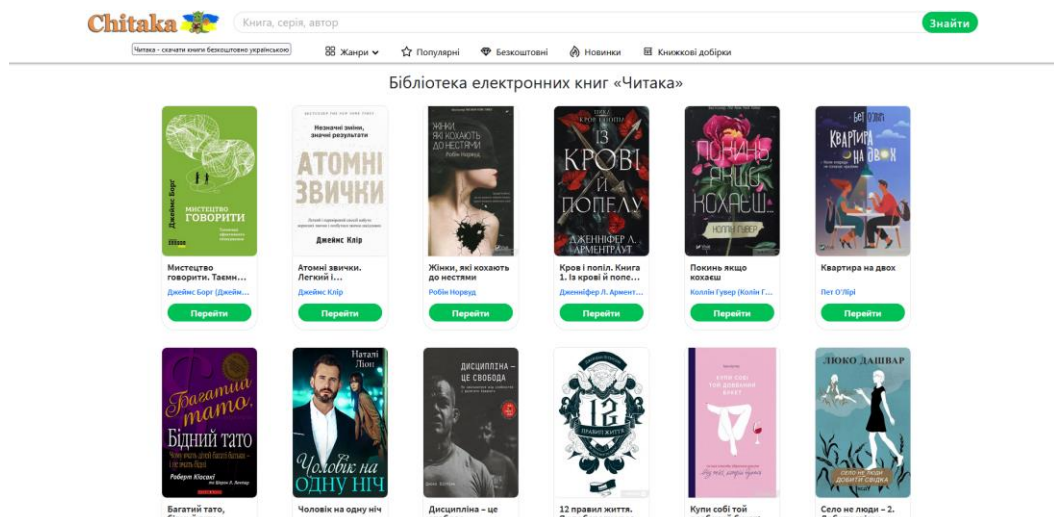


Рисунок 1.2 – Головна сторінка сайту «Читака»



### 1.3 Постановка задачі

Метою роботи є розробка та впровадження електронної бібліотеки з можливістю завантаження книг, яка забезпечує зручний доступ до різноманітних ресурсів, підтримує сучасні технології захисту авторських прав та інтегрується з іншими інформаційними системами.

Для досягнення поставленої мети необхідно вирішити наступні задачі:

- 1) Аналіз існуючих рішень та визначення вимог до електронної бібліотеки.
  - Провести огляд сучасних електронних бібліотек та визначити їх сильні та слабкі сторони.
  - Сформулювати вимоги до функціоналу, інтерфейсу та безпеки майбутньої електронної бібліотеки на основі аналізу користувацьких потреб.
- 2) Розробка інтерфейсу користувача та архітектури системи:
  - Створити інтуїтивно зрозумілий та адаптивний дизайн інтерфейсу, який забезпечує зручний доступ до функцій завантаження книг після реєстрації.
- 3) Реалізація функціоналу електронної бібліотеки:
  - Розробити та впровадити функції пошуку, сортування та фільтрації книг за жанрами.
  - Створити базу даних для зберігання інформації.
- 4) Забезпечення безпеки даних:
  - Розробити політику конфіденційності та безпеки, що гарантує захист особистих даних користувачів та безпеку інформаційних ресурсів.
- 5) Тестування та оптимізація системи:
  - Провести всебічне тестування системи для виявлення та усунення можливих помилок і недоліків.

## 2. Вибір методів розв'язання задачі

### 2.1 Огляд технологій для розробки

Для розробки інформаційної системи я обрав такі технології як: HTML, PHP, CSS, MySql та Java Script.

По-перше, інформаційна система буде реалізована у вигляді веб-сторінки, тому використання HTML буде обов'язковим.

По-друге, зовнішній вигляд буде зроблено за допомогою стилів CSS, бо я зможу відділити основний код від оформлення та розробити різне оформлення під користувачів мобільних пристроїв.

Ще однією важливою технологією при створенні інформаційної системи є мова PHP, тому що вона має такий ряд ознак як:

- висока швидкість роботи;
- легкість в освоєнні та простий синтаксис;
- відмінна сумісність та переносимість;
- має відкритий вихідний код;
- дуже актуальний в наш час;
- щоб створити просту веб-сторінку достатньо звичайного редактору;
- висока гнучкість, ємність та функціональність. Програма PHP може бути окремо від розробки веб-сторінки, без зв'язування, після чого поєднується. Це значно спрощує життя програмістів;

- багатозадачні та широкі можливості – створення будь-яких веб-додатків, блогів, гостей, інтернет-магазинів, сайтів, роботи з переспрямуванням, заголовками, PDF-документами, базами, електронною поштою тощо.

Останньою технологією є JavaScript, вона популярна завдяки своїм безсумнівним перевагам:

- незамінність для веб-розробки. Підтримка скриптів від всіх популярних браузерів; Повна інтеграція з макета сторінки (HTML + CSS) і серверної частини;

- швидкість роботи і продуктивність. JavaScript дозволяє частково обробляти веб-сторінки на комп'ютері користувачів без запитів до сервера. Це економить час і трафік, знижує навантаження на сервер;

- потужна інфраструктура (екосистема);

- простота і раціональність застосування. Проста задача може бути вирішена протягом 5 хвилин, не треба робити додаткову роботу. Для складних завдань є різні варіанти вирішення, ви можете підібрати кращий або адаптувати інший;

- зручність для користувацьких інтерфейсів. Заповнення форм, вибір дій, кнопка активації, вхідні перевірки, що реагують на наведення / клік миші і т. д.

Як і будь-яка мова програмування, JavaScript має деякі недоліки:

- немає ніякого способу читання і завантаження файлів. Це обмеження функціональності на стороні клієнта;

- вільна інтерпретація, мова ігнорує очевидні невідповідності, існує інша інтерпретація даних. Немає можливості раннього виявлення помилок. Всі недоліки виявляються вже на стадії роботи;

- немає підтримки для віддаленого доступу. Таким чином, мова не може бути використана для мережових додатків. Слід зазначити, що підприємницькі кола користувачів JavaScript активно покращують мову, усуваючи багато недоліків.

### 3. Програмна реалізація

#### 3.1 Розробка інтерфейсу

Шапка сайту включає: навігаційне меню, картинку в якості фону, форму входу та пошук (див. рисунок 3.1). Вона ідентична для всіх сторінок окрім адмін-панелі. Детальніше розписано у додатку Б.

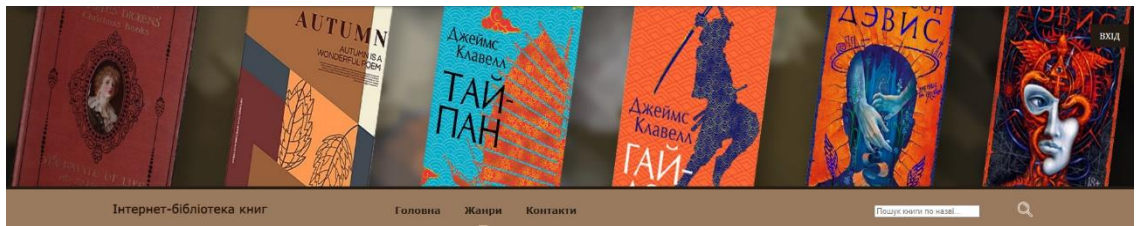


Рисунок 3.1 – Шапка сайту

При створення напису «Інтернет-бібліотека книг» було використано такі стилі:

```
.site_name{
    padding: 30px 0px 30px 0px;
    text-decoration: none;
    color: white;
    width: 330px;
    -webkit-text-stroke: 0.7px #44BAE5;
    font-family: Verdana;
    font-size: 30px;
}
.site_name:hover{
    text-decoration: underline;
    text-decoration-color: #44BAE5;
}
```

Інтернет-бібліотека книг

Рисунок 3.2 – Результат роботи стилів при наведенні на напис

Пояснення щодо стилів:

- padding - створює відступи;
- text-decoration - створює підкреслення, закреслення або прибирає його;
- webkit-text-stroke - створює обведення тексту;
- site\_name:hover - відображує стилі при наведенні на об'єкт.

Для навігаційного меню було створено такі стилі:

```
.menu{
    list-style: none;
    display: flex;
    margin: 0px;
}
.menu li{
    transition-property: background, color;
    transition-duration: 0.3s;
    transition-timing-function: ease-in-out;
    text-align: center;
}
.menu a{
    color: white;
    font-size: 13pt;
    text-decoration: none;
    display: block;
    padding: 35px 30px 35px 30px;
}
```

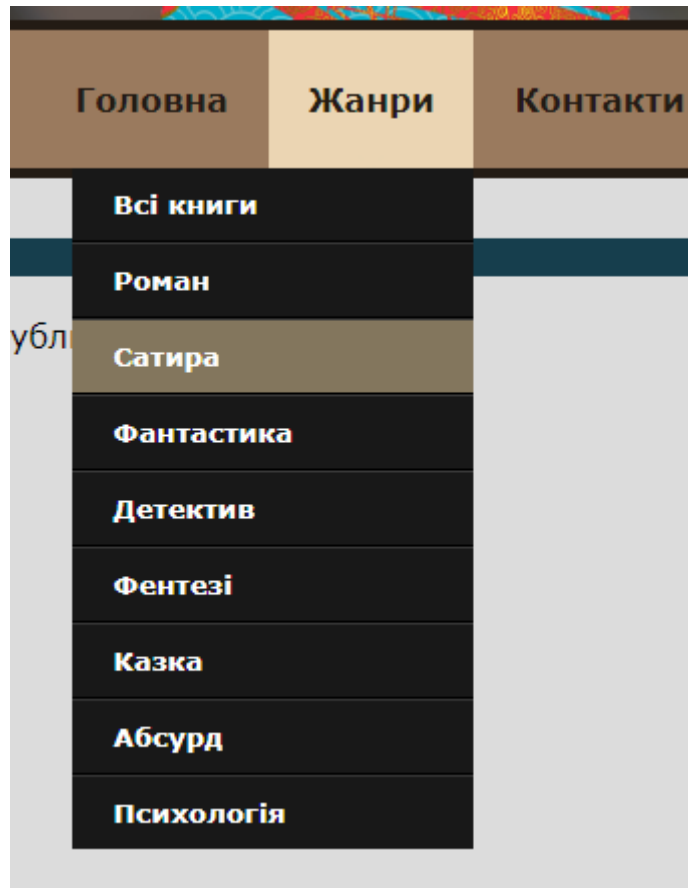


Рисунок 3.3 – Результат роботи стилів навігаційної панелі

Пояснення щодо стилів:

- `list-style` - встановлює тип маркеру;
- `display:flex` - використовується для дуже гнучкого вирівнювання об'єктів горизонтально та вертикально;
- `transition` - створює плавний перехід із одних стилів в інші;
- `display:block` - перетворює елемент в блок та заповнює простір цим блоком.

Контент на сторінці `index.php` показано на рисунку 3.4.



Рисунок 3.4 – Центральний блок на сторінці index.php

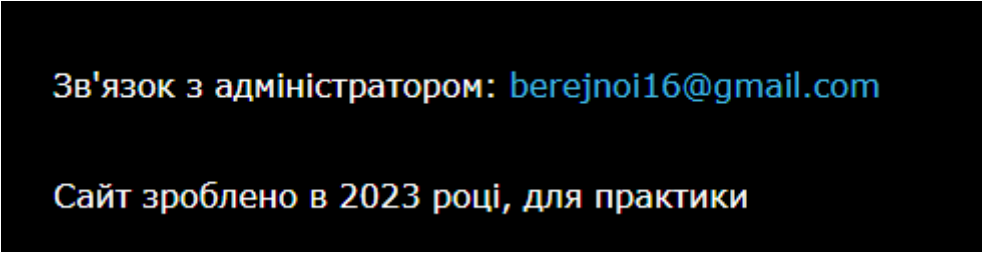
Використані стилі для цього блоку показані далі.

```
.first_chapter{
    min-height: 800px;
}
.first_chapter .date{
    background-color: #163e4d;
    color: white;
    font-family: Verdana;
    margin: 0px;
    padding-left: 40px;
}
.first_chapter .text{
    font-family: Verdana;
    padding-left: 10px;
}
```

Для підвалу сайту були використані такі стилі. Підвал ідентичний для кожної сторінки окрім адмін-панелі (див. рисунок 3.5).

```
.last_chapter{
    background-color: black;
    color: white;
    width: 100%;
    height: 130px;
}
```

```
.last_chapter p{
    padding-left: 150px;
    padding-right: 150px;
    padding-top: 30px;
    font-family: Verdana;
}
.last_chapter a{
    text-decoration: none;
    color: #44BAE5;
}
.last_chapter a:hover{
    text-decoration: underline;
}
```



Зв'язок з адміністратором: [berejnoi16@gmail.com](mailto:berejnoi16@gmail.com)

Сайт зроблено в 2023 році, для практики

Рисунок 3.5 – Підвал сайту

Розташування об'єктів й саму сторінку з книгами показано на рисунку 3.6.



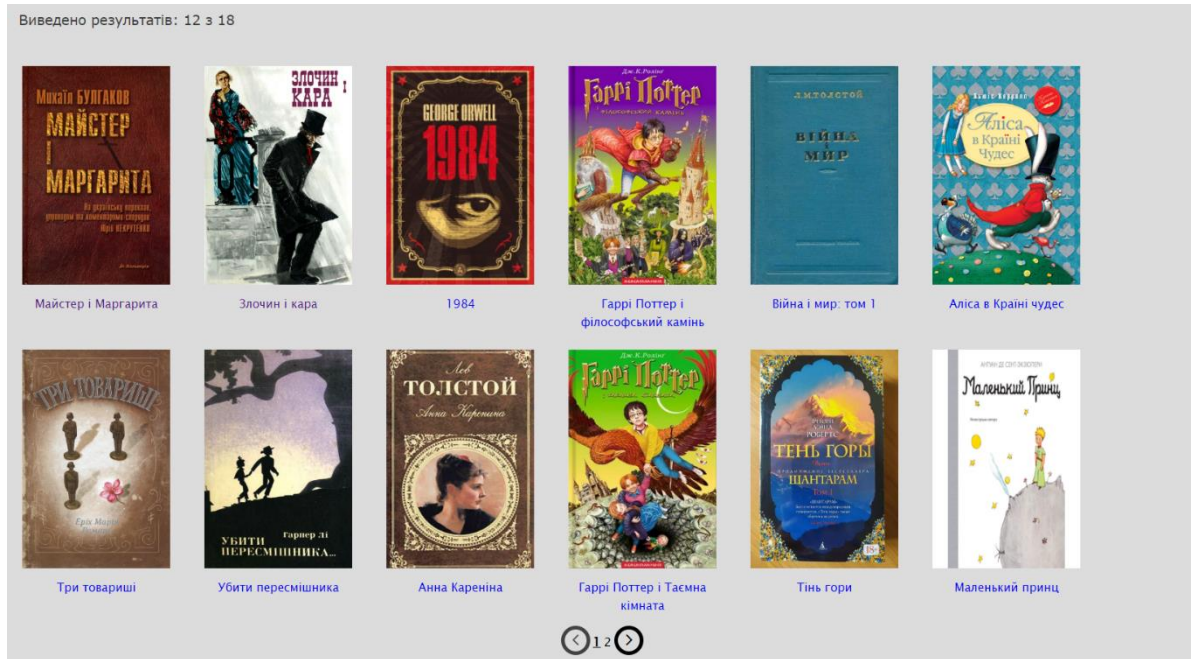


Рисунок 3.6 – Сторінка з книгами

```
.second_chapter{
    min-height: 900px;
}
.second_chapter img{
    width: auto;
    height: 300px;
    width: 303.383px;
    display: inline-block;
}
.images{
    margin-top: 40px;
}
.imagename{
    word-break: break-word;
    text-align: center;
    width: inherit;
    margin-top: 10px;
    margin-bottom: 5px;
    display: grid;
    width: 303.383px;
```

```
}
```

Пояснення щодо стилів:

- `display: inline-block` - використовується для розміщення елементів в лінію и перетворення їх в блокові елементи;
- `display: grid` - використовується для розміщення елементів за сіткою;
- `word-break` - використовується для переносу слів, які не помістилися однією лінією в блок.

```
.buttons span{
    font-size: 13pt;
    font-family:'Lucida Sans', 'Lucida Sans Regular', 'Lucida
Grande', 'Lucida Sans Unicode', Geneva, Verdana, sans-serif;
    cursor: pointer;
}
.not_active{
opacity: 70% !important;
cursor: default !important;
}
.not_active:hover{
    background: none !important;
    color: black !important;
    cursor: default !important;
}
.format{
    font-weight: bold;
    text-decoration: underline;
}
```

Пояснення щодо стилів:

- `cursor` - встановлює тип курсору при наведенні на об'єкт;
- `opacity` - встановлює прозорість об'єкту;
- `font-weight` - встановлює насиченість шрифту.



Рисунок 3.7 – Робота стилів для кнопок

Наступна сторінка – це сторінка з інформацією про книгу на рисунку 3.8.



Рисунок 3.8 – Сторінка з інформацією про книгу

```
.Third_chapter{
    display: inline-block;
}
.book_name{
    text-align: center;
    font-size: 3vw;
}
.only_image img{
    height: 400px;
}
.only_image{
    width: 300px;
    float: left;
}
.info_about_book{
    width: 800px;
```

```

word-wrap: break-word;
float: left;
}

```

Адмін панель також має свої стилі (див. рисунок 3.9).

Вхід успішний!

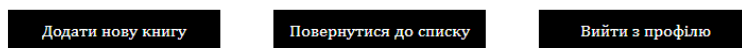


Рисунок 3.9 – Адмін-панель

```

.add_new_book
{
    margin-top: 50px;
    margin-left: 50px;
    width: 300px;
    height: 40px;
    background: black;
    color: white;
    text-align: center;
    cursor: pointer;
    transition: all 500ms ease-in-out;
    float: left;
    margin-bottom: 50px;
}
.add_new_book:hover{
    transform: scale(1.1);
}
.add_new_book p{
    padding-top: 10px;
}

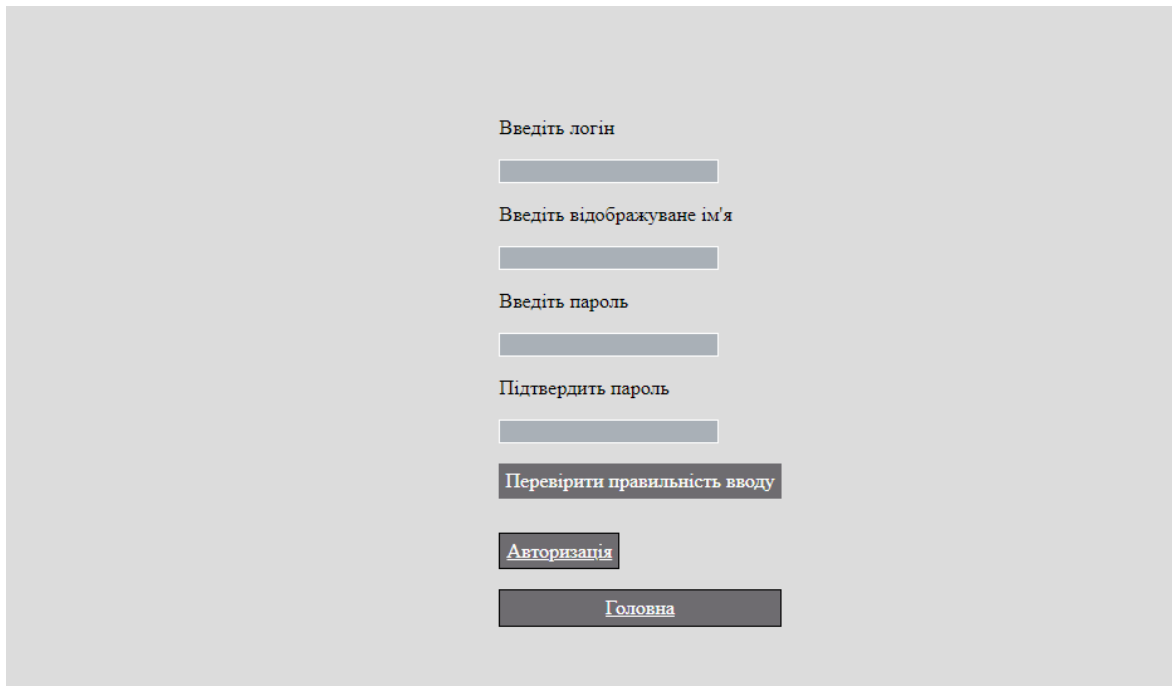
```

Пояснення щодо стилів:

- `transform` - дозволяє повертати, масштабувати, нахилити або зрушувати елемент.

Сторінка з реєстрацією була створена для користувачів, які мають бажання завантажити та прочитати книгу, без належної реєстрації, на сайті існують обмеження.

Роботу стилів зображено на рисунку 3.10.



The image shows a registration form on a light gray background. The form consists of the following elements from top to bottom:

- Text label: "Введіть логін" (Enter login)
- Text input field
- Text label: "Введіть відображуване ім'я" (Enter display name)
- Text input field
- Text label: "Введіть пароль" (Enter password)
- Text input field
- Text label: "Підтвердіть пароль" (Confirm password)
- Text input field
- Button: "Перевірити правильність вводу" (Check input correctness)
- Button: "Авторизація" (Authorization)
- Button: "Головна" (Home)

Рисунок 3.10 – Сторінка реєстрації

```
.registration_button a{
    padding: 5px 5px 5px 5px;
    color: white;
    background: rgb(110, 108, 113);
    border: 1px solid black;
    cursor: pointer;
    font-size: 13pt;
    position: relative;
    top: 5px;
}
.registration_button a:hover{
    filter: brightness(130%);
}
.login_form form div{
    margin-top: 10px;
```

```

}
#test_pass p{
    margin-bottom: 0px;
    background: rgb(110, 108, 113);
    padding: 5px 5px 5px 5px;
    color: white;
    cursor: pointer;
}

```

Пояснення щодо стилів:

- Filter: brightness – регулює яскравість елементу.

### 3.2 Розробка бази даних

Для веб-сайту було створено базу даних, концептуальну модель та структуру таблиць якої можна побачити на рисунках 3.11 – 3.16.

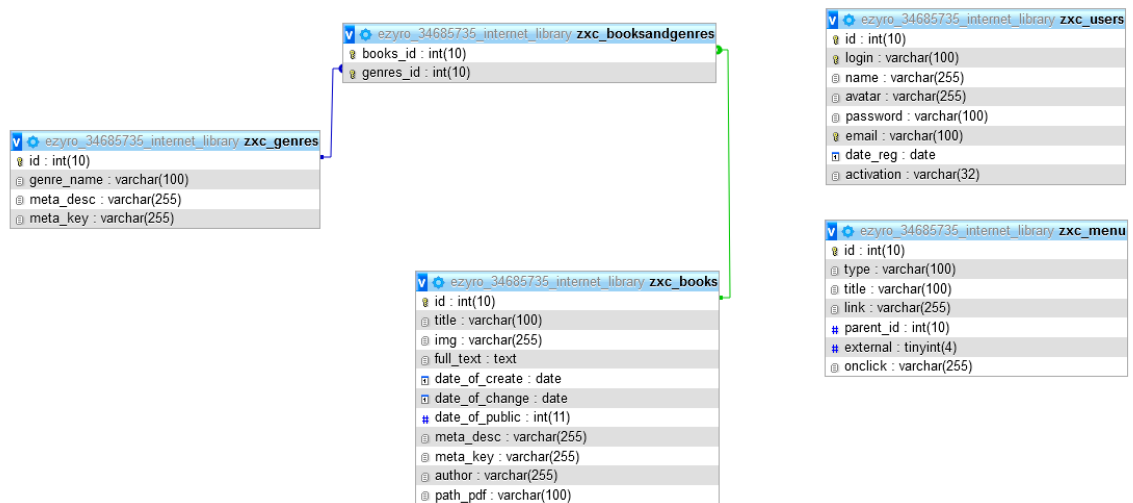


Рисунок 3.11 – Концептуальна модель бази даних книг

#	Имя	Тип	Сравнение	Атрибуты	Null	По умолчанию	Комментарии	Дополнительно	Действие
<input type="checkbox"/>	1 <b>id</b> 🔑	int(10)			Нет	Нет		AUTO_INCREMENT	Изменить  Удалить  Ещё
<input type="checkbox"/>	2 <b>title</b>	varchar(100)	utf8_general_ci		Да	NULL			Изменить  Удалить  Ещё
<input type="checkbox"/>	3 <b>img</b>	varchar(255)	utf8_general_ci		Да	NULL			Изменить  Удалить  Ещё
<input type="checkbox"/>	4 <b>full_text</b>	text	utf8_general_ci		Да	NULL			Изменить  Удалить  Ещё
<input type="checkbox"/>	5 <b>date_of_create</b>	date			Да	NULL			Изменить  Удалить  Ещё
<input type="checkbox"/>	6 <b>date_of_change</b>	date			Да	NULL			Изменить  Удалить  Ещё
<input type="checkbox"/>	7 <b>date_of_public</b>	int(11)			Да	NULL			Изменить  Удалить  Ещё
<input type="checkbox"/>	8 <b>meta_desc</b>	varchar(255)	utf8_general_ci		Да	NULL			Изменить  Удалить  Ещё
<input type="checkbox"/>	9 <b>meta_key</b>	varchar(255)	utf8_general_ci		Да	NULL			Изменить  Удалить  Ещё
<input type="checkbox"/>	10 <b>author</b>	varchar(255)	utf8_general_ci		Да	NULL			Изменить  Удалить  Ещё
<input type="checkbox"/>	11 <b>path_pdf</b>	varchar(100)	utf8_general_ci		Да	NULL			Изменить  Удалить  Ещё

Рисунок 3.12 – структура таблиці zxc\_books

#	Имя	Тип	Сравнение	Атрибуты	Null	По умолчанию	Комментарии	Дополнительно	Действие
<input type="checkbox"/>	1 <b>id</b> 🔑	int(10)			Нет	Нет		AUTO_INCREMENT	Изменить  Удалить  Ещё
<input type="checkbox"/>	2 <b>genre_name</b>	varchar(100)	utf8_general_ci		Да	NULL			Изменить  Удалить  Ещё
<input type="checkbox"/>	3 <b>meta_desc</b>	varchar(255)	utf8_general_ci		Да	NULL			Изменить  Удалить  Ещё
<input type="checkbox"/>	4 <b>meta_key</b>	varchar(255)	utf8_general_ci		Да	NULL			Изменить  Удалить  Ещё

Рисунок 3.13 – структура таблиці zxc\_genres

#	Имя	Тип	Сравнение	Атрибуты	Null	По умолчанию	Комментарии	Дополнительно	Действие
<input type="checkbox"/>	1 <b>books_id</b> 🔑	int(10)			Нет	Нет			Изменить  Удалить  Ещё
<input type="checkbox"/>	2 <b>genres_id</b> 🔑	int(10)			Нет	Нет			Изменить  Удалить  Ещё

Рисунок 3.14 – структура таблиці zxc\_booksandgenres

#	Имя	Тип	Сравнение	Атрибуты	Null	По умолчанию	Комментарии	Дополнительно	Действие
<input type="checkbox"/>	1 <b>id</b> 🔑	int(10)			Нет	Нет		AUTO_INCREMENT	Изменить  Удалить  Ещё
<input type="checkbox"/>	2 <b>type</b>	varchar(100)	utf8_general_ci		Да	NULL			Изменить  Удалить  Ещё
<input type="checkbox"/>	3 <b>title</b>	varchar(100)	utf8_general_ci		Да	NULL			Изменить  Удалить  Ещё
<input type="checkbox"/>	4 <b>link</b>	varchar(255)	utf8_general_ci		Да	NULL			Изменить  Удалить  Ещё
<input type="checkbox"/>	5 <b>parent_id</b>	int(10)			Да	NULL			Изменить  Удалить  Ещё
<input type="checkbox"/>	6 <b>external</b>	tinyint(4)			Да	NULL			Изменить  Удалить  Ещё
<input type="checkbox"/>	7 <b>onclick</b>	varchar(255)	utf8_general_ci		Да	NULL			Изменить  Удалить  Ещё

Рисунок 3.15 – структура таблиці zxc\_menu

#	Имя	Тип	Сравнение	Атрибуты	Null	По умолчанию	Комментарии	Дополнительно	Действие
<input type="checkbox"/>	1 <b>id</b> 🔑	int(10)			Нет	Нет		AUTO_INCREMENT	Изменить  Удалить  Ещё
<input type="checkbox"/>	2 <b>login</b> 🔑	varchar(100)	utf8_general_ci		Да	NULL			Изменить  Удалить  Ещё
<input type="checkbox"/>	3 <b>name</b>	varchar(255)	utf8_general_ci		Да	NULL			Изменить  Удалить  Ещё
<input type="checkbox"/>	4 <b>avatar</b>	varchar(255)	utf8_general_ci		Да	NULL			Изменить  Удалить  Ещё
<input type="checkbox"/>	5 <b>password</b>	varchar(100)	utf8_general_ci		Да	NULL			Изменить  Удалить  Ещё
<input type="checkbox"/>	6 <b>email</b> 🔑	varchar(100)	utf8_general_ci		Да	NULL			Изменить  Удалить  Ещё
<input type="checkbox"/>	7 <b>date_reg</b>	date			Да	NULL			Изменить  Удалить  Ещё
<input type="checkbox"/>	8 <b>activation</b>	varchar(32)	utf8_general_ci		Да	NULL			Изменить  Удалить  Ещё

Рисунок 3.16 – структура таблиці zxc\_users

Прикладом взаємодії бази даних та php слугує наступний код. Детальніше розписано у додатку А.

```

function get_all_books()
{
    global $servername;
    global $username;
    global $password;
    global $dbname;
    $search = $_GET['search'];
    $genre = $_GET['genre_name'];
    $decode_genre = urldecode($genre);
    $decode_search = urldecode($search);
    $spec_page = $_GET['page'] - 1;
    $curr_page = $spec_page * 13;
    $conn = new mysqli($servername, $username, $password,
$dbname);
    if ($conn->connect_error) {
        die("Connection failed: " . $conn->connect_error);
    }
    $conn->set_charset("utf8");
    $where_s = "";
    $join = "";
    if ($decode_search != NULL) {
        $where_s = "where title like '%$decode_search%'";
    }
    else if ($decode_genre != NULL)
    {
        $join = " left join zxc_booksandgenres bg on
bg.books_id = b.id left join zxc_genres g on g.id = bg.genres_id
";
        $where_s = "where genre_name = '$decode_genre'";
    }
}

```

Прикладом виконання запитів та використання даних з бази даних слугує

КОД:

```

$sql = "SELECT b.id as book_id, b.title as book_name, b.img
as img from zxc_books b $join $where_s limit $curr_page,13";

```



```

if ($result = $conn->query($sql)) {
    while ($row = $result->fetch_array()) {
        $book_name_curr = $row["book_name"];
        $book_id_curr = $row["book_id"];
        // $image = sprintf("%.03d", $book_id_curr);
        // $url = "images/" . $image . ".jpg";
        $book_img = $row["img"];
        echo "<div class=\"imageplusname\">" . "<a
href=\"about_book.php?book_id=" . $book_id_curr . "\"><img
width=\"303.383px\" height=\"300px\" src=\"$book_img\"></a>" .
"<div class=\"imagename\">" . "<a href=\"about_book.php?book_id="
. $book_id_curr . "\">" . $book_name_curr . "</a>" . "</div>" .
"</div>";
    }
}
$conn->close();
}

```

## Висновок

Веб-сайт створений як інтернет-бібліотека книг. За допомогою таких технологій, як PHP, JavaScript, HTML та CSS, веб-сайт був створений з використанням PHP для серверної частини, що дозволяє динамічно обробляти запити та працювати з базами даних; JavaScript, який забезпечує інтерактивність. HTML, який формує структуру веб-сторінок і дозволяє розміщувати на них текст, зображення та інші медіа; та CSS, який відповідає за оформлення і стиль, надаючи сайту привабливий та зручний для користувачів вигляд. Насамперед призначений для користувачів зацікавлених в книгах. За допомогою навігаційної панелі та гіперпосилань можна дізнатися важливу інформацію про книгу, наприклад, автора, якого року відбулась публікація, жанр і т.д. Була створена база даних з якої можна дістати та записати дані. Реалізовано систему реєстрації та авторизації, після якої користувач зможе завантажити на свій комп'ютер книгу в форматі pdf.

На відміну від інших аналогів, дана бібліотека дозволяє користувачам не лише отримувати інформацію про книги, але й завантажувати їх на свій комп'ютер у форматі PDF для зручного читання. Це робить її більш привабливою для тих, хто бажає мати доступ до книжок навіть без Інтернет-з'єднання. Крім того, бібліотека зосереджена на забезпеченні зручного та ефективного користування за допомогою інтуїтивно зрозумілого інтерфейсу та навігаційної панелі.

### Список використаних джерел

1. Сайт «HTML BOOK» URL: <https://htmlbook.online/> (Дата звернення 14.07.2023).
2. Сайт о документації для розробників «MDN Web Docs» URL: <https://developer.mozilla.org/ru/> (Дата звернення 14.07.2023).
3. Українська електронна бібліотека Лібрук «Libruk» URL: <https://libruk.com.ua/> (Дата звернення 20.07.2023).
4. Електронна бібліотека «Chitaka» URL: <https://chitaka.com.ua/> (Дата звернення 20.07.2023).
5. Сайт «PHP» URL: <https://www.php.net/> (дата звернення: 21.07.2023).
6. Сайт «W3schoolsUA» українською URL: <https://w3schoolsua.github.io/index.html#gsc.tab=0> (Дата звернення 20.07.2023).
7. Сайт «Javascript.Info» URL: <https://uk.javascript.info/> (Дата звернення 17.07.2023).
8. Сайт «Web Library» URL: <http://weblib.com.ua/> (Дата звернення 15.07.2023).
9. Сайт «JavaScript довідка українською» URL: <http://яваскрипт.укр/> (Дата звернення 16.07.2023).
10. Сайт «Php українською» URL: <https://php.org.ua> (Дата звернення 14.07.2023).

## Додаток А

Файл admin\_panel\_edit.php

```

<?php
include "config.php";
global $servername;
    global $username;
    global $password;
    global $dbname;
    global $admin_login;
    global $admin_pass;

$book_id_curr = $_GET["book_id"];

$conn = new mysqli($servername, $username, $password,
$dbname);

if ($conn->connect_error) {
    die("Connection failed: " . $conn->connect_error);
}

$conn->set_charset("utf8");

    $book_genres = "";

    $sql3 = "SELECT GROUP_CONCAT(g.genre_name SEPARATOR
', ') as genre_name from zxc_booksandgenres bg left join
zxc_genres g on bg.genres_id=g.id where books_id = $book_id_curr";

    if ($result2 = $conn->query($sql3)) {
        while ($row2 = $result2->fetch_array()) {
            $book_genres = $row2["genre_name"];
        }
    }

    $sql1 = "SELECT * from zxc_books where id =
$book_id_curr";

```

```

if ($result = $conn->query($sql1)) {
    while ($row = $result->fetch_array()) {
        $book_name = $row["title"];
        $book_desc = $row["full_text"];
        $book_image = $row["img"];
        $book_pdf = $row["path_pdf"];
        $book_author = $row["author"];
        $book_year = $row["date_of_public"];
        $book_meta_key = $row["meta_key"];
        $book_meta_desc = $row["meta_desc"];
    }
}

echo "
<!DOCTYPE html>
<html>
<head>
    <meta charset=\"UTF-8\">
    <title>Админ-панель</title>
    <style>
        *{
            box-sizing: border-box;
            margin: 0px;
            font-family: Cambria, Cochin, Georgia, Times,
'Times New Roman', serif;
        }
        h1{
            text-align: center;
            padding-top: 40px;

```

```
        font-weight: lighter;
    }
    .add_new_book
    {
        margin-top: 50px;
        margin-left: 50px;
        width: 200px;
        height: 40px;
        background: black;
        color: white;
        text-align: center;
        cursor: pointer;
        transition: all 500ms ease-in-out;
        float: left;
        margin-bottom: 50px;
    }
    .add_new_book:hover{
        transform: scale(1.1);
    }
    .add_new_book p{
        padding-top: 10px;
    }
    .edit_book
    {
        margin-top: 50px;
        margin-left: 50px;
        width: 200px;
```

```
    height: 40px;
    float: left;
    background: black;
    color: white;
    text-align: center;
    cursor: pointer;
    margin-bottom: 50px;
    transition: all 500ms ease-in-out;
}
.edit_book:hover{
    transform: scale(1.1);
}
.edit_book p{
    padding-top: 10px;
}
form{
    border: 5px black solid;
}
.admin{
    float: left;
    width: 80%;
    margin-left: 200px;
}
form p{
    text-align: center;
    padding-top: 40px;
```

```
        border-bottom: 4px black solid;
        padding-bottom: 20px;
    }
    .last_p{
        border: none;
    }
    input{
        margin-left: 20px;
        margin-top: 20px;
    }
    textarea{
        resize: none;
        margin-left: 20px;
        margin-top: 20px;
    }
</style>
</head>
<body>
    <h1>Редагування книги!</h1>
    <div id=\"return_to_books\" class=\"add_new_book\">
        <p>Повернутися до списку книг</p>
    </div>
    <div id=\"exit\" class=\"add_new_book\">
        <p>Вийти з профілю</p>
    </div>
    <div class=\"admin\" id=\"admin\"></div>
    <script>
```



```

let login = localStorage.getItem('login');
let pass = localStorage.getItem('pass');
if(login !== "."\"$admin_login\""."&& pass
!=".\"$admin_pass\""."")
{
    location.href = \"index.php\";
}

let add = document.getElementById(\"add\");
let main_div = document.getElementById(\"admin\");
let form = document.createElement(\"form\");
let input1 = document.createElement(\"input\");
input1.setAttribute(\"name\", \"book_name\");
input1.setAttribute(\"required\", \"\");
input1.value = \"$book_name\";
let input2 = document.createElement(\"input\");
input2.setAttribute(\"name\", \"meta_key\");
input2.setAttribute(\"required\", \"\");
input2.value = \"$book_meta_key\";
let input3 = document.createElement(\"input\");
input3.setAttribute(\"name\", \"book_author\");
input3.setAttribute(\"required\", \"\");
input3.value = \"$book_author\";
let input4 = document.createElement(\"input\");
input4.setAttribute(\"name\", \"book_year\");
input4.setAttribute(\"required\", \"\");
input4.value = \"$book_year\";
let input5 = document.createElement(\"input\");
input5.setAttribute(\"name\", \"meta_desc\");

```

```

input5.setAttribute("required", "");
input5.value = "$book_meta_desc";
let input2_2 = document.createElement("input");
input2_2.setAttribute("name", "book_image");
input2_2.setAttribute("required", "");
let img = document.createElement("img");
img.setAttribute("src", "$book_image");
img.setAttribute("width", "200");
let input3_3 = document.createElement("input");
input3_3.setAttribute("name", "book_pdf");
input3_3.setAttribute("required", "");
let
                                textarea
document.createElement("textarea");
                                =
                                textarea.value = "$book_desc";
let
                                textarea_for_genres
document.createElement("textarea");
                                =
                                textarea_for_genres.setAttribute("required",
"\");
                                textarea.setAttribute("name", "book_desc");
                                textarea.setAttribute("required", "");
                                textarea_for_genres.value = "$book_genres";
let p1 = document.createElement("p");
let p2 = document.createElement("p");
let p2_2 = document.createElement("p");
let p3_3 = document.createElement("p");
let p3 = document.createElement("p");
let p4 = document.createElement("p");
let p5 = document.createElement("p");
let p6 = document.createElement("p");

```

```

let p10 = document.createElement("p");
let div1 = document.createElement("div");
let div2 = document.createElement("div");
let div2_2 = document.createElement("div");
let div3_3 = document.createElement("div");
let div3 = document.createElement("div");
let div4 = document.createElement("div");
let div5 = document.createElement("div");
let div6 = document.createElement("div");
let div10 = document.createElement("div");
let last_p = p10.classList.add("last_p");
    form.setAttribute("action",
"db_edit_book.php");
    form.setAttribute("method" , "post");
    form.setAttribute("enctype", "multipart/form-
data");
    input1.setAttribute("type", "text");
    textarea.setAttribute("id", "meta_desc");
    textarea.setAttribute("rows", "5");
    textarea.setAttribute("cols", "33");
    textarea_for_genres.setAttribute("rows", "5");
    textarea_for_genres.setAttribute("cols",
"33");
    textarea_for_genres.setAttribute("name",
"book_genres");
    input2.setAttribute("type", "text");
    input2.setAttribute("id", "meta_key");
    input3.setAttribute("type", "text");
    input3.setAttribute("id", "author");

```

```
input4.setAttribute(\"type\", \"number\");
input4.setAttribute(\"id\", \"year\");
input5.setAttribute(\"type\", \"text\");
input5.setAttribute(\"id\", \"meta_desc\");
input2_2.setAttribute(\"type\", \"file\");
input2_2.setAttribute(\"accept\", \".jpg\");
input3_3.setAttribute(\"type\", \"file\");
input3_3.setAttribute(\"accept\", \".pdf\");
main_div.appendChild(form);
form.appendChild(div1);
form.appendChild(div2);
form.appendChild(div2_2);
form.appendChild(div3_3);
form.appendChild(div3);
form.appendChild(div4);
form.appendChild(div5);
form.appendChild(div6);
form.appendChild(div10);
div1.appendChild(input1);
div2.appendChild(textarea);
div2_2.appendChild(p2_2);
div2_2.appendChild(input2_2);
div2_2.appendChild(img);
div3_3.appendChild(p3_3);
div3_3.appendChild(input3_3);
div3.appendChild(input3);
div4.appendChild(input4);
```

```

div5.appendChild(input2);
div6.appendChild(input5);
div10.appendChild(textarea_for_genres);
div10.appendChild(p10);
p1.textContent="\Введіть назву книги:";
div1.appendChild(p1);
p2.textContent="\Введіть опис книги:";
div2.appendChild(p2);
p2_2.textContent="\Додайте зображення:";
div2_2.appendChild(p2_2);
p3_3.textContent="\Додайте pdf файл книги:";
div3_3.appendChild(p3_3);
p3.textContent="\Введіть автора:";
div3.appendChild(p3);
p4.textContent="\Введіть год публікації:";
div4.appendChild(p4);
p5.textContent="\Введіть мета-опис:";
div5.appendChild(p6);
p6.textContent="\Введіть мета-теги:";
div6.appendChild(p5);
p10.textContent="\Введіть жанр книги, ЧЕРЕЗ КОМУ
ТА З ВЕЛИКОЇ ЛІТЕРИ:";
let submit = document.createElement("button");
submit.setAttribute("type", "submit");
submit.textContent="\Відправити";
div10.appendChild(submit);
function return_books()
{

```

```

        location.href
        \"books.php?page=1&genre_name=&search=\";

    }

    return_to_books.addEventListener(\"click\",
return_books);

    function exit_from_profile()
    {
        location.href = \"index.php\";
        localStorage.clear();
    }

    exit.addEventListener(\"click\",
exit_from_profile);

</script>

</div>

</body>

</html>\";

$conn->close();

?>

```

### Файл allbooks.php

```

<?php

include 'config.php';

function get_all_books()
{
    global $servername;
    global $username;
    global $password;
    global $dbname;

```

```

$search = $_GET['search'];
$genre = $_GET['genre_name'];
$decode_genre = urldecode($genre);
$decode_search = urldecode($search);
$spec_page = $_GET['page'] - 1;
$curr_page = $spec_page * 12;

$conn = new mysqli($servername, $username, $password,
$dbname);

if ($conn->connect_error) {
    die("Connection failed: " . $conn->connect_error);
}

$conn->set_charset("utf8");

$where_s = "";
$join = "";

if ($decode_search != NULL) {
    $where_s = "where title like '%$decode_search%'";
}

else if ($decode_genre != NULL)
{
    $join = " left join zxc_booksandgenres bg on bg.books_id =
b.id left join zxc_genres g on g.id = bg.genres_id ";
    $where_s = "where genre_name = '$decode_genre'";
}

$sql = "SELECT b.id as book_id, b.title as book_name, b.img as
img from zxc_books b $join $where_s limit $curr_page,12";

if ($result = $conn->query($sql)) {
    while ($row = $result->fetch_array()) {
        $book_name_curr = $row["book_name"];
        $book_id_curr = $row["book_id"];
    }
}

```

```

        // $image = sprintf("%'.03d", $book_id_curr);
        // $url = "images/" . $image . ".jpg";

        $book_img = $row["img"];

        echo      "<div      class=\"imageplusname\">"      .      "<a
href=\"about_book.php?book_id="      .      $book_id_curr      .      "\"><img
width=\"203.383px\"      height=\"300px\"      src=\"\"$book_img\"></a>"      .
"<div class=\"imagename\">"      .      "<a href=\"about_book.php?book_id="
.      $book_id_curr      .      "\">"      .      $book_name_curr      .      "</a>"      .      "</div>"      .
"</div>";

    }

}

$conn->close();
}

```

### Файл books\_on\_page.php

```

<?php

include 'config.php';

$count = 0;

$procc = 0;

$expand = 0;

$curr_count = 0;

$page = 0;

$spec_page = 0;

global $servername;

global $username;

global $password;

global $dbname;

$search = $_GET['search'];

$genre = $_GET['genre_name'];

$decode_search = urldecode($search);

```



```

$decode_genre = urldecode($genre);
$spec_page = $_GET['page'] - 1;
$curr_page = $spec_page * 12;
$conn = new mysqli($servername, $username, $password, $dbname);
if ($conn->connect_error) {
    die("Connection failed: " . $conn->connect_error);
}
$conn->set_charset("utf8");
$where_s = "";
$join = "";
if ($decode_search != NULL) {
    $where_s = "where title like '%$decode_search%'";
}
else if ($decode_genre != NULL)
{
    $join = " left join zxc_booksandgenres bg on bg.books_id =
b.id left join zxc_genres g on g.id = bg.genres_id ";
    $where_s = "where genre_name = '$decode_genre'";
}
$sql = "SELECT count(*) as count from zxc_books b $join $where_s";
if ($result = $conn->query($sql)) {
    while ($row = $result->fetch_array()) {
        $count = $row["count"];
        $procc = $count / 12;
        $expand = ceil($procc);
    }
}
$spec_page = $_GET['page'] - 1;

```

```

$page = $_GET['page'];

$curr_page = $spec_page * 12;

$sql = "SELECT COUNT(*) as curr_count FROM (Select b.id from
zxc_books b $join $where_s LIMIT $curr_page, 12) AS array";

if ($result = $conn->query($sql)) {
    while ($row = $result->fetch_array()) {
        $curr_count = $row["curr_count"];
    }
}

$conn->close();

```

### Файл db\_add\_new\_book.php

```

<?php

include "config.php";

global $servername;

global $username;

global $password;

global $dbname;

$conn = new mysqli($servername, $username, $password, $dbname);

if ($conn->connect_error) {
    die("Connection failed: " . $conn->connect_error);
}

$conn->set_charset("utf8");

$book_genres = "";

$book_desc = "";

$book_image = "";

$book_pdf = "";

$book_metakey = "";

$book_metadesc = "";

```

```
$book_name = "";
$book_author = "";
$book_year = "";

if (isset($_POST["book_name"])) {

    $book_name = $_POST["book_name"];
}

if (isset($_POST["book_desc"])) {

    $book_desc = $_POST["book_desc"];
}

if (isset($_POST["meta_key"])) {

    $book_metakey = $_POST["meta_key"];
}

if (isset($_POST["meta_desc"])) {

    $book_metadesc = $_POST["meta_desc"];
}

if (isset($_POST["book_author"])) {

    $book_author = $_POST["book_author"];
}

if (isset($_POST["book_year"])) {

    $book_year = $_POST["book_year"];
}
```

```

}

if (isset($_POST["book_genres"]) && $_POST["book_genres"] != NULL) {
    $book_genres = $_POST["book_genres"];
    $genres = explode(",", $book_genres);
    $count = count($genres);
    $genres_id = array();
    for ($i = 0; $i < $count; $i++) {
        $genres[$i] = trim($genres[$i]);
    }
    for ($i = 0; $i < $count; $i++) {
        $sql = "select count(*) as count from zxc_genres where
upper(genre_name) = upper('$genres[$i]')";
        if ($result = $conn->query($sql)) {
            while ($row = $result->fetch_array()) {
                $genres_diff = $row["count"];
            }
        }
        if ($genres_diff == 0) {
            $sql = "SHOW TABLE STATUS FROM `internet_library`
LIKE 'zxc_genres'";
            if ($result = $conn->query($sql)) {
                while ($row = $result->fetch_array()) {
                    $AI = $row["Auto_increment"];
                }
            }
            $sql = "insert into zxc_genres (genre_name) values
('$genres[$i]')";
            if ($result = $conn->query($sql)) {
            }
        }
    }
}

```

```

        $genres_id[$i] = $AI;
    } else {
        $sql = "select id from zxc_genres where
upper(genre_name) = upper('$genres[$i]')";
        if ($result = $conn->query($sql)) {
            while ($row = $result->fetch_array()) {
                $genre_id = $row["id"];
            }
        }
        $genres_id[$i] = $genre_id;
    }
}

$sql = "SHOW TABLE STATUS FROM `internet_library` LIKE
'zxc_books' ";
if ($result = $conn->query($sql)) {
    while ($row = $result->fetch_array()) {
        $AI = $row["Auto_increment"];
    }
}

if ($_FILES && $_FILES["book_image"]["error"]== UPLOAD_ERR_OK)
{
    $name = sprintf("%.03d", $AI);
    $path = "images/". $name. ".jpg";
    move_uploaded_file($_FILES["book_image"]["tmp_name"], $path);
}

if ($_FILES && $_FILES["book_pdf"]["error"]== UPLOAD_ERR_OK)
{

```

```

$name2 = sprintf("%.03d", $AI);

$path2 = "pdf/".$name2.".pdf";

move_uploaded_file($_FILES["book_pdf"]["tmp_name"], $path2);

}

$sql = "SHOW TABLE STATUS FROM `internet_library` LIKE
'zxc_books'";

    if ($result = $conn->query($sql)) {
        while ($row = $result->fetch_array()) {
            $book_id_curr = $row["Auto_increment"];
        }
    }

$sql = "insert into zxc_books(title, img, full_text,
date_of_create, date_of_change, date_of_public, meta_desc,
meta_key, author, path_pdf) values

('$book_name','$path', '$book_desc', SYSDATE(), NULL,
$book_year, '$book_metadesc', '$book_metakey', '$book_author',
'$path2')";

if ($result = $conn->query($sql)) {

}

for($i=0;$i<count($genres_id);$i++){

    $sql = "insert into zxc_booksandgenres (books_id, genres_id)
values ($book_id_curr,$genres_id[$i])";

    if ($result = $conn->query($sql)) {

        }

    }

$conn->close();

echo "<script>

location.href = \"admin_panel.php?book_id=$book_id_curr\"

```

```
</script>";
```

```
?>
```

### Файл db\_add\_user.php

```
<?php
```

```
include 'config.php';
```

```
    $conn = new mysqli($servername, $username, $password,  
$dbname);
```

```
    if ($conn->connect_error) {
```

```
        die("Connection failed: " . $conn->connect_error);
```

```
    }
```

```
    $conn->set_charset("utf8");
```

```
    if (isset($_POST["login"])) {
```

```
        $login = $_POST["login"];
```

```
    }
```

```
    if (isset($_POST["password1"])) {
```

```
        $password1 = $_POST["password1"];
```

```
    }
```

```
    if (isset($_POST["name"])) {
```

```
        $name = $_POST["name"];
```

```
    }
```

```
    if ((isset($_POST["login"])&& $_POST["login"]!= NULL) &&  
(isset($_POST["password1"])&& $_POST["password1"]!= NULL) &&  
(isset($_POST["name"])&& $_POST["name"]!= NULL)) {
```

```
        $login = $_POST["login"];
```

```
        $password1 = $_POST["password1"];
```

```
        $name = $_POST["name"];
```

```

        $sql = "select count(*) as count from zxc_users where
upper(login) = upper('$login)";

        if ($result = $conn->query($sql)) {
            while ($row = $result->fetch_array()) {
                $login_diff = $row["count"];
            }
        }

        if ($login_diff == 0) {
            $hashed_password = md5($password1);

            $sql = "insert into zxc_users (login, password,
date_reg, name) values ('$login', '$hashed_password', SYSDATE(),
'$name)";

            if ($result = $conn->query($sql)) {
                echo "location.href =
\"index.php?login=$login&name=$name\"; </script>";
            }
        }
        else {
            echo "<script>
location.href = \"registration_page.php\";
alert(\"Такий логін вже існує\");
</script>";
        }
    }

    $conn->close();
?>

```

### Файл db\_auth.php

```

<?php
include 'config.php';

```



```

global $servername;

    global $username;
    global $password;
    global $dbname;
    global $admin_login;
    global $admin_pass;

    if (isset($_POST["login"])) {

        $login = $_POST["login"];
    }

    if (isset($_POST["password1"])) {

        $password1 = $_POST["password1"];
        $hashed_password = md5($password1);
    }

    $conn = new mysqli($servername, $username, $password,
    $dbname);

    if ($conn->connect_error) {

        die("Connection failed: " . $conn->connect_error);
    }

    $conn->set_charset("utf8");

    if($login == $admin_login && $hashed_password ==
    md5($admin_pass))

    {

        echo "<script>

        myStorage = window.localStorage;

        localStorage.setItem('login', '$login');

```

```

        localStorage.setItem('password', '$admin_pass');

        location.href="index.php?\";

    </script>\";

}

else{

    $sql = \"SELECT * from zxc_users where login = '$login'\";

    if ((isset($_POST[\"login\"])&& $_POST[\"login\"]!= NULL) &&
(isset($_POST[\"password1\"])&& $_POST[\"password1\"]!= NULL)) {

        if ($result = $conn->query($sql)) {

            while ($row = $result->fetch_array()) {

                $user_login = $row[\"login\"];

                $user_password = $row[\"password\"];

                $user_name = $row[\"name\"];

                if($login == $user_login && $hashed_password ==
$user_password)

                {

                    echo \"<script>

                    myStorage = window.localStorage;

                    localStorage.setItem('login', '$user_login');

                    localStorage.setItem('password',
'$user_password');

                    localStorage.setItem('name', '$user_name');

                    location.href=\\\"index.php\\\";

                    </script>\";

                }

            }

        }

    }

    echo \"<script>

location.href = \\\"auth_page.php\\\";

```

```
    alert(\"Неправильно введені данні\");  
</script>\";  
    }  
  
}
```

```
    $conn->close();  
?>
```

### Файл db\_book.php

```
<?php  
include 'config.php';  
function get_info()  
{  
    global $servername;  
    global $username;  
    global $password;  
    global $dbname;  
    global $admin_login;  
    global $admin_pass;  
    $book_id_curr = $_GET["book_id"];  
    $conn = new mysqli($servername, $username, $password,  
$dbname);  
    if ($conn->connect_error) {  
        die("Connection failed: " . $conn->connect_error);  
    }  
    $conn->set_charset("utf8");  
    $sql = "SELECT * FROM zxc_books WHERE id = $book_id_curr";  
    if ($result = $conn->query($sql)) {
```

```

while ($row = $result->fetch_array()) {
    $book_id_row = $row["id"];
    $book_title_row = $row["title"];
    $book_desc_row = $row["full_text"];
    $book_public_row = $row["date_of_public"];
    $book_author_row = $row["author"];
    $book_pdf_row = $row["path_pdf"];
    $book_img = $row["img"];
    // $image = sprintf("%'.03d", $book_id_row);
    // $url = "images/".$image.".jpg";
    $book_genres = "";
    $sql2 = "SELECT GROUP_CONCAT(g.genre_name SEPARATOR ',
') as genre_name from zxc_booksandgenres bg left join zxc_genres g
on bg.genres_id=g.id left join zxc_books b on b.id = bg.books_id
where b.id = $book_id_curr";
    if ($result = $conn->query($sql2)) {
        while ($row = $result->fetch_array()) {
            $book_genres = $row["genre_name"];
        }
    }
    echo "
<p class=\"film_name\">".$book_title_row."</p>
<div class=\"only_image\">
    <img          width=\"271.183\"          height=\"400px\"
src=\"\".$book_img.\">
</div>
<div class=\"info_about_film\" id=\"info_about_film\">
    <p>Назва:
    ".$book_title_row."

```

```

</p>
<p>Жанр:
"$.book_genres."
</p>
<p>Опис:
"$.book_desc_row."
</p>
<p>Дата публікації:
"$.book_public_row." г."."
</p>
<p>Автор:
"$.book_author_row."
</p>
</div>
<script>
    let info_div =
document.getElementById("info_about_film");
    if(localStorage.getItem('login') != null &&
localStorage.getItem('password') != null)
    {
        let div = document.createElement("div");
        div.setAttribute("class",
"download_button");
        let a = document.createElement("a");
        a.setAttribute("href", "$book_pdf_row");
        a.setAttribute("download",
"$book_title_row");
        a.textContent = "Завантажити книгу";
        info_div.appendChild(div);
    }
</script>

```

```

        div.appendChild(a);
    }
    else{
        let p = document.createElement('p');
        p.style.color = '#c5530e';
        p.textContent = "\"Щоб завантажити книгу, вам
потрібно авторизуватися\"";
        info_div.appendChild(p);
    }
</script>
";
    }
}
$conn->close();
}

```

### Файл db\_manager.php

```

<?php
include 'config.php';
function get_date_list()
{
    global $servername;
    global $username;
    global $password;
    global $dbname;

    $conn = new mysqli($servername, $username, $password,
$dbname);

    if ($conn->connect_error) {
        die("Connection failed: " . $conn->connect_error);
    }
}

```

```

}

$conn->set_charset("utf8");

$sql_1 = "SELECT distinct cast(date_of_create as date) as
created FROM `zxc_books` order by created desc";

if ($result = $conn->query($sql_1)) {
    while ($row = $result->fetch_array()) {
        $date_curr = $row["created"];

        echo "<p class=\"date\">Були додані такі книги на сайт
в " . $date_curr . "</p>";

        $sql_2 = "SELECT id, title, author, date_of_public,
date_of_change, img from zxc_books where cast(date_of_create as
date) = '$date_curr'";

        if ($result_2 = $conn->query($sql_2)) {
            while ($row_2 = $result_2->fetch_array()) {
                $book_id = $row_2["id"];
                $book_name = $row_2["title"];
                $book_year = $row_2["date_of_public"];
                $author = $row_2["author"];
                $img_path = $row_2["img"];
                // $image = sprintf("%'.03d", $book_id);
                // $url = "images/".$image.".jpg";

                echo "<div><a
href=\"about_book.php?book_id=".$book_id.\"\"
class=\"text_link\">\"".$book_name.\"</a><span class=\"text\">,
Автор: ".$author.", Год публікації: ".$book_year.</span>";

                echo "<br><a
href=\"about_book.php?book_id=".$book_id.\"\"><img width=\"210px\"
height=\"260px\" src=\"".$img_path.\"\"></div></a>";

            }
        }
    }
}
}

```

```
$conn->close();  
}  
?>
```



## Додаток Б

### Файл fin\_styles.css

```
body{
    margin: 0px;
    padding: 0px;
    background-color: #DCDCDC;
    min-height: 100vh;
    display: flex;
    flex-direction: column;
}
* {
    box-sizing: border-box;
}
.login
{
    margin-right: 10px;
    margin-top: 15px;
}
.admin{
    margin-right: 20px;
    font-size: 23px;
    color: black;
    -webkit-text-stroke: 0.5px #231b14;
}
#enter_button
{
    visibility: hidden;
}
#register_button{
    visibility: hidden;
}
.main_img
{
```

```
    height: 300px;
    background: url(-12.png) no-repeat;
    background-size: 100% 300px;
}
.user{
    float: right;
    margin-top: 40px;
    text-align: right;
    height: 100px;
}
.main_img a{
    text-decoration: none;
    color: white;
    padding: 10px 15px 10px 15px;
    font-family: Cambria, Cochin, Georgia, Times, 'Times New
Roman', serif;
    text-transform: uppercase;
}
.a2{
    margin-right: 20px;
    background: #231b14;
    cursor: pointer;
}
.background{
    padding-left: 150px;
    padding-right: 150px;
    display: grid;
    margin-bottom: 20px;
}
.nav{
    display: flex;
    background: #9a7a5e;
    border-top: 5px solid #231b14;
    border-bottom: 5px solid #231b14;
```

```
padding-left: 150px;
padding-right: 150px;
justify-content: space-around;
font-family: Verdana;
margin-bottom: 30px;
}
.site_name{
padding: 20px 0px 20px 0px;
text-decoration: none;
color: rgb(0, 0, 0);
width: 350px;
-webkit-text-stroke: 0.7px #563111;
font-family: Verdana;
font-size: 20px;
}
.site_name:hover{
text-decoration: underline;
text-decoration-color: #513d2c;
}
.right_menu{
float: right;
width: 800px;
}
.menu{
list-style: none;
display: flex;
margin: 0px;
}
.menu li{
transition-property: background, color;
transition-duration: 0.3s;
transition-timing-function: ease-in-out;
text-align: center;
}
```

```

.menu a{
    color: #231b14;
font-size: 12pt;
text-decoration: none;
font-weight: bold;
display: block;
padding: 25px 20px 25px 20px;
}
.menu li:hover{
background:
#EBD5B3;
}
#button_search
{
    cursor: pointer;
}
.search{
padding-top: 25px;
height: 10px;
}
.search input:focus{
outline: 3px solid #EBD5B3;
}
.button_search_div {
display: block;
margin-top: auto;
margin-bottom: auto;
}
.nav ul li{
position: relative;
}
.icon::before{
content: '';
position: absolute;

```

```
display: block;
margin: auto;
height: 0;
width: 0;
left: 0;
right: 0;
bottom: -1px;
border-left: 8px solid transparent;
border-right: 8px solid transparent;
border-top: 10px solid #EBD5B3;
}
.additional_menu{
    list-style: none;
    display: none;
}
.additional_menu li{
    text-align: left;
    border-top: 2px groove #383838;
}
.additional_menu li:hover{
    background-color: #83765d;
}
.menu li:hover .additional_menu{
    display: block;
    position: absolute;
    top: 100%;
    right: 0;
    width: 200px;
    background: #181818;
    padding: 0px;
}
.additional_menu li a{
    color: white;
    font-size: 10pt;
```

```
        text-decoration: none;
        padding: 10px 20px 10px 20px;
    }
    .additional_menu li:first-child{
        border: 0px;
    }
    .first_chapter{
        min-height: 800px;
    }
    .first_chapter .date{
        background-color: #163e4d;
        color: white;
        font-family: Verdana;
        margin: 0px;
        padding-left: 40px;
    }
    .first_chapter .text{
        font-family: Verdana;
        padding-left: 10px;
    }
    .first_chapter div{
        padding-top: 20px;
        padding-bottom: 20px;
    }
    .first_chapter div a{
        text-decoration: none;
        font-family: Verdana;
        font-size: 12pt;
        color: rgb(65, 81, 124);
    }
    .first_chapter div a:hover{
        font-weight: bold;
        color: black;
        text-decoration: underline;
    }
```

```
}  
.first_chapter img{  
    padding-top: 10px;  
    transition: all 255ms ease-in-out;  
}  
.first_chapter img:hover{  
    transform: scale(1.05);  
}  
.last_chapter{  
    background-color: black;  
    color: white;  
    width: 100%;  
    height: 130px;  
    margin-top: auto;  
}  
.last_chapter p{  
    padding-left: 150px;  
    padding-right: 150px;  
    padding-top: 20px;  
    font-family: Verdana;  
}  
.last_chapter a{  
    text-decoration: none;  
    color: #44BAE5;  
}  
.last_chapter a:hover{  
    text-decoration: underline;  
}  
  
/* Конец html файла страницы новости */  
  
.second_chapter{  
    min-height: 900px;  
}
```

```
.second_chapter img{
    width: auto;
    height: 300px;
    width: 203.383px;
    display: inline-block;
}
.images{
    margin-top: 40px;
}
.imageplusname{
    max-width: 250px;
    display: inline-block;
    padding: 10px 100px 10px 5px;
    transition: all 255ms ease-in-out;
}
.imagename a{
    text-decoration: none;
    width: inherit;
    font-family: 'Lucida Sans', 'Lucida Sans Regular',
'Lucida Grande', 'Lucida Sans Unicode', Geneva, Verdana, sans-
serif;
}
.imageplusname img:hover{
    transform: scale(1.05);
}
.imagename{
    word-break: break-word;
    text-align: center;
    width: inherit;
    margin-top: 10px;
    margin-bottom: 5px;
    display: grid;
    width: 203.383px;
}
```



```
.sort{
    min-height: 25px;
    font-family: Verdana;
    font-size: 14pt;
    color: #383838;
}
.sort a{
    text-decoration: none;
    color: black;
}
.buttons{
    text-align: center;
}
.fa{
    font-size: 27px !important;
}
.fa-angle-left{
    border: 5px solid black;
    height: 40px;
    width: 40px;
    border-radius: 100%;
    text-align: center;
    transition: 0.3s ease-in-out;
    transition-property: background, color;
}
.fa-angle-right{
    border: 5px solid black;
    height: 40px;
    width: 40px;
    border-radius: 100%;
    text-align: center;
    transition: 0.3s ease-in-out;
    transition-property: background, color;
```

```
}
.fa-angle-left:hover{
    background: black;
    color: white;
    cursor: pointer;
}
.fa-angle-right:hover{
    background: black;
    color: white;
    cursor: pointer;
}
.buttons span{
    font-size: 13pt;
    font-family:'Lucida Sans', 'Lucida Sans Regular', 'Lucida
Grande', 'Lucida Sans Unicode', Geneva, Verdana, sans-serif;
    cursor: pointer;
}
.not_active{
opacity: 70% !important;
cursor: default !important;
}
.not_active:hover{
    background: none !important;
    color: black !important;
    cursor: default !important;
}
.format{
    font-weight: bold;
    text-decoration: underline;
}

/* Конец php файла страницы с книгами */

.Third_chapter{
```

```
        display: inline-block;
    }
    .film_name{
        text-align: center;
        font-size: 2vw;
    }
    .only_image img{
        height: 400px;
    }
    .only_image{
        width: 300px;
        float: left;
    }
    .info_about_film{
        width: 800px;
        word-wrap: break-word;
        float: left;
    }
    .info_about_film p{
        font-family: 'Avenir Next', 'BlinkMacSystemFont', '-apple-
system', 'Segoe UI', 'Roboto', 'Helvetica', 'Arial', 'sans-serif';
        font-size: 1vw;
    }
    #edit_button
    {
        width: 200px;
        float: left;
        cursor: pointer;
    }
    #edit_button p{
        transition: all 475ms ease-in-out;
    }
    #edit_button p:hover{
        text-decoration: underline;
    }
```

```
        color: #3b9cc0;
    }
    .download_button{
        width: 150px;
        height: 50px;
        cursor: pointer;
        text-align: center;
        display: table-cell;
        vertical-align: middle;
        border: #3290b3 0.5px solid;
        font-size: 1vw;
        transition: all 475ms ease-in-out;
        border-radius: 5px;
        background-color: #44BAE5;
    }
    .download_button a{
        text-decoration: none;
        color: #ece5df;
    }
    .download_button:hover{
        background-color: #3290b3;
    }
    .login_form_wrapper{
        display: grid;
        margin-bottom: 50px;
    }
    .login_form{
        margin-left: 400px;
        margin-right: 400px;
        display: grid;
        justify-content: center;
        margin-top: 170px;
        margin-bottom: 300px;
    }
}
```

```
.login_form input{
    background: rgb(169 176 183);
    border: 1px solid ;
    outline: none;
    color: white;
}
.login_form input:hover{
    color: white;
    background: rgb(141, 147, 153);
    cursor: pointer;
}
.login_button_login_page{
    float: right;
    padding: 5px 5px 5px 5px;
    color: white;
    background: rgb(110, 108, 112);
    border: 1px solid black;
    cursor: pointer;
    font-size: 12pt;
}
.login_button_login_page:hover{
    filter: brightness(120%);
}
.registration_button a{
    padding: 5px 5px 5px 5px;
    color: white;
    background: rgb(110, 108, 112);
    border: 1px solid black;
    cursor: pointer;
    font-size: 12pt;
    position: relative;
    top: 5px;
}
.registration_button a:hover{
```

```
        filter: brightness(120%);
    }
    .login_form form div{
        margin-top: 10px;
    }
    #test_pass p{
        margin-bottom: 0px;
        background: rgb(110, 108, 112);
        padding: 5px 5px 5px 5px;
        color: white;
        cursor: pointer;
    }
    /* Для экранов шириной до 767px (Телефоны) */
    @media (max-width: 767px) {
        .nav .right_menu{
            font-size: 2vw;
        }
        .site_name {
            font-size: 16px;
        }
        .date {
            font-size: 1vw;
        }
        .first_chapter div{
            font-size: 1vw;
        }
        .first_chapter div a{
            font-size: 1vw;
        }
        .download_button{
            height: 150px;
            width: 300px;
        }
    }
}
```

```
/* Для экранов шириной 768px и до 1023px (Планшеты) */
@media (min-width: 768px) and (max-width: 1023px) {
    .nav .right_menu{
        font-size: 2vw;
    }
    .site_name {
        font-size: 16px;
    }
    .date {
        font-size: 1vw;
    }
    .first_chapter div{
        font-size: 1vw;
    }
    .first_chapter div a{
        font-size: 1vw;
    }
    .info_about_film p{
        font-size: 2vw;
    }
    .download_button{
        height: 70px;
        width: 170px;
    }
    .download_button a{
        font-size: 3vw;
    }
}
```