

РЕФЕРАТ

Пояснювальна записка до дипломного проекту (роботи) бакалавра: 73 с., 27 рис., 4 табл., 23 бібліографічних джерел посилань, 19 с. додатків.

Об'єкт розробки: веб-додаток для ведення тематичного блогу.

Мета роботи: розробка веб-додатку для ведення тематичного блогу.

В проекті виконано:

- 1 Аналіз існуючих аналогів
- 2 Вибір засобів для виконання дипломного проекту
- 3 Реалізація прототипів
- 4 Структура веб-додатку
- 5 Використання веб-додатку
- 6 Охорона праці та безпека в надзвичайних ситуаціях

Отримано наступні результати: краще ознайомилася з мовою PHP, поліпшила знання в області програмування та верстки сайтів, переглянула схожі веб-додатки, оцінила їх плюси та мінуси. Створила веб-додаток, який в подальшому легко можна модернізувати та покращити, додати потрібні функції, а бо видалити наявні.

Практичне значення, галузь застосування роботи: застосовувати у особистих цілях для вираження своєї думки та спілкування

Ключові слова: веб-додаток, розробка, блог, мова програмування, охорона праці.

Умови одержання дипломного проекту: СНУ ім. В. Даля, пр. Центральний 59-А, м. Сєвєродонецьк, 93400.

ЗМІСТ

ВСТУП.....	4
1 АНАЛІЗ ЗАДАЧІ РОЗРОБКИ.....	6
1.1 Аналіз існуючих аналогів тематичних блогів.....	6
1.2 Постановка задачі на розробку тематичного блогу.....	8
1.3 Функціонал веб-додатку.....	9
2 ВИБІР ЗАСОБІВ ДЛЯ РОЗРОБКИ ВЕБ-ДОДАТКУ ДЛЯ ВЕДЕННЯ ТЕМАТИЧНОГО БЛОГУ.....	10
2.1 Вибір мови програмування веб-додатку.....	10
2.2 Вибір додаткових засобів для створення веб-додатку.....	11
2.2.1 Мова розмітки HTML.....	11
2.2.2 Каскадні таблиці стилів.....	12
2.2.3 Мова для управління базами даних SQL.....	12
2.3 Вибір програмного забезпечення.....	13
2.3.1 Програмне середовище OpenServer.....	13
2.3.2 Текстовий редактор Atom.....	14
2.3.3 Додаток для створення прототипів Balsamiq mockups.....	14
2.4 Обґрунтування вибору засобів розробки.....	14
2.5 Висновки до розділу 2.....	14
3 РЕАЛІЗАЦІЯ ПРОТОТИПІВ ВЕБ-ДОДАТКУ ДЛЯ ВЕДЕННЯ ТЕМАТИЧНОГО БЛОГУ.....	15
4 РОЗРОБКА ВЕБ-ДОДАТКУ ДЛЯ ВЕДЕННЯ ТЕМАТИЧНОГО БЛОГУ.....	17
4.1 Створення бази даних веб-додатку.....	17
4.2 Структура коду веб-додатку.....	19
4.3 Висновки до розділу 4.....	23
5 РОБОТА ВЕБ-ДОДАТКУ ДЛЯ ВЕДЕННЯ ТЕМАТИЧНОГО БЛОГУ.....	24
5.1 Вигляд сторінок та їх функції.....	24

5.2 Реєстрація та додавання статті.....	26
6 ОХОРОНА ПРАЦІ ТА БЕЗПЕКА В НАДЗВИЧАЙНИХ СИТУАЦІЯХ.....	28
6.1 Загальні питання з охорони праці.....	29
6.1.1 Правові та організаційні основи охорони праці.....	29
6.1.2 Організаційно-технічні заходи з безпеки праці.....	32
6.2 Аналіз стану умов праці.....	33
6.2.1 Вимоги до приміщень.....	33
6.2.2 Вимоги до організації місця праці.....	34
6.2.3 Навантаження та напруженість процесу праці.....	36
6.3 Виробнича санітарія.....	37
6.3.1 Аналіз небезпечних та шкідливих факторів при виробництві (експлуатації) виробу.....	37
6.3.2 Пожежна безпека.....	39
6.3.3 Електробезпека.....	39
6.4 Гігієнічні вимоги до параметрів виробничого середовища.....	40
6.4.1 Мікроклімат.....	40
6.4.2 Освітлення.....	41
6.5 Шум та вібрація, електромагнітне випромінювання.....	42
6.6 Вентилювання.....	43
6.7 Заходи з організації виробничого середовища та попередження виникнення надзвичайних ситуацій.....	44
6.8 Висновки до розділу 6.....	48
ВИСНОВКИ.....	50
ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ.....	51
ДОДАТКИ.....	53
Додаток А. Лістинг програми.....	53
Додаток Б. Презентація.....	68

ВСТУП

В даний час велика кількість людей користуються Інтернетом не лише для того, щоб знайти цікаву для них інформацію, наприклад, свіжі новини, або відвідати інформаційні ресурси певної спрямованості, задовольнивши свою допитливість. Так само вони використовують Інтернет для спілкування з однодумцями і знайомими в соціальних мережах. Але є і такі користувачі, які спілкуються виключно через блоги, і таких користувачів налічується не мало, їх число зростає з кожним днем.

Блог - це веб-сайт, основний вміст якого - регулярно додавані записи, що містять текст, зображення або мультимедіа. Для блогів характерні недовгі записи тимчасової значущості, впорядковані в зворотному хронологічному порядку. Відмінності блогу від традиційного щоденника обумовлюються середовищем: блоги зазвичай публічні і передбачають сторонніх читачів.

За версією газети Вашингтон профайл (англ. Washington Profile), першим блогом вважають сторінку Тіма Бернерса-Лі, де він, починаючи з 1992 року, публікував новини. Широке використання блогів розпочалося з 1996 року.

У серпні 1999 року комп'ютерна компанія «Pyra Labs» із Сан-Франциско відкрила сайт Blogger. Це була перша безкоштовна блогова служба. Згодом Blogger був викуплений компанією Google.

2006 року після позову компанії «Apple» (2004 р.) на блогерів, було винесено рішення суду про те, що вони володіють тими ж правами по нерозголошенню джерел інформації, що й журналісти.

Відмінністю блогів від щоденникових записів є те, що перші зазвичай передбачають сторонніх читачів, які можуть вступити в публічну дискусію з автором.

В наш час, в епоху нових технологій, книги стали набагато доступніше, ніж, скажімо, кілька десятиліть тому. Книги можна купити в магазинах як на паперових носіях, так і в електронному вигляді.

Відомо, що читання продукує розумову діяльність, стимулює розвиток очних м'язів, що підвищує і концентрацію, і здатність до спостереження. Крім того, читання передбачає повну концентрацію читача на процесі, покращує комунікативні навички. При читанні ми знаходимо такі точні і влучні фрази, які дуже доречні в повсякденному житті - тобто розвиток словникового запасу.

У дипломній роботі планується розробити веб-додаток для ведення блогу, в якому читачі зможуть поділитися враженнями від прочитаної книги.

1 АНАЛІЗ ЗАДАЧІ РОЗРОБКИ

1.1 Аналіз існуючих аналогів тематичних блогів

Перед створенням веб-додатку для ведення тематичного блогу треба ознайомитися з існуючими аналогами для виділення наявних плюсів та недоліків.

1) Bigga Книга

Адреса сайту : <http://biggakniga.ru/>. Скриншоти блогу (див. рис. 1.1, 1.2, 1.3).



Рисунок 1.1 – Головна сторінка блогу



Рисунок 1.2 – Головна сторінка блогу

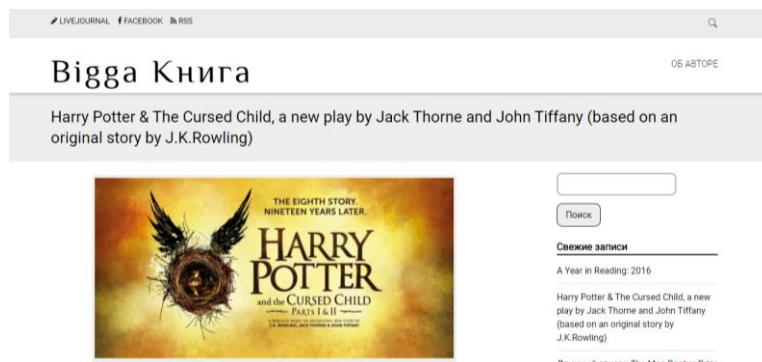


Рисунок 1.3 – Сторінка статті

Блог має світлий дизайн, більша частина сторінки має білий фон, у верхньому лівому куті розташована назва сайту, під нею зображення, праворуч мається сторінка про автора блогу. На головній сторінці знаходяться останні статті, які розташовані у трьох колонках, мають заголовки, які чітко читаються. Натиснувши на назву будь-якої статті, включається сторінка з повним текстом, зображенням та можливістю поділитися посиланням на цей запис у соціальних мережах.

2) Маленький читатель

Адреса сайту: <http://littlereader.ru/>

Скриншоти блогу (див. рис. 1.4, 1.5, 1.6).

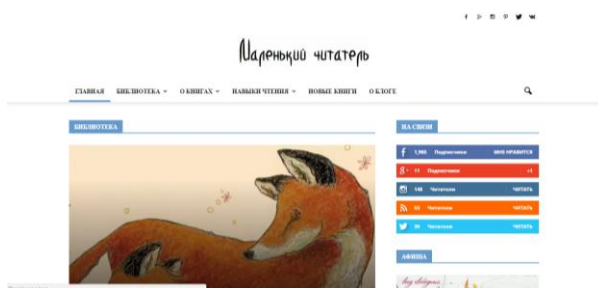


Рисунок 1.4 – Головна сторінка блогу Маленький читатель

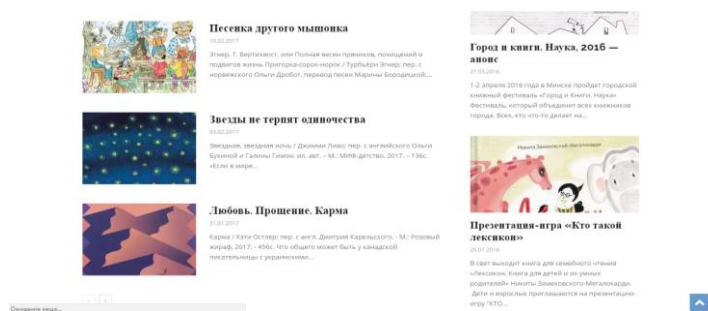


Рисунок 1.5 - Головна сторінка блогу Маленький читатель

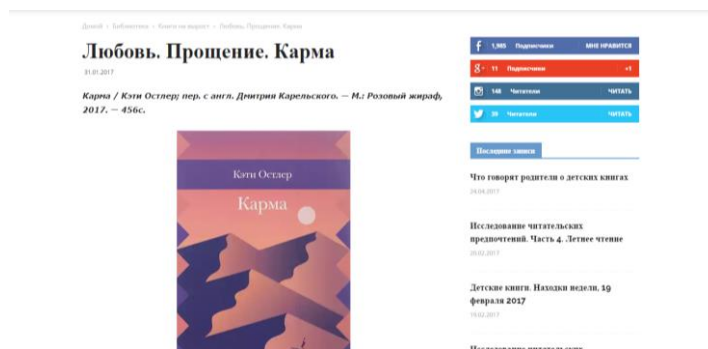


Рисунок 1.6 – Сторінка статті

Сайт має білий фон на всіх сторінках, зверху посередині розташована назва блогу, під ним меню з посиланнями на сторінки. У двох колонках на головній сторінці розташовані останні статті ліворуч, можливість поділитися посиланнями у соціальних мережах та афіша праворуч. Натиснувши на заголовок будь-якої статті, відкривається сторінка з повним текстом та зображенням.

1.2 Постановка задачі на розробку тематичного блогу

У даному дипломному проєкті (роботі) постало завдання створити веб-додаток для ведення тематичного блогу, а саме блогу, в якому бажаючі зареєстровані користувачі можуть поділитися своїми враженнями, тобто написати статтю, про прочитані ними книги.

Цей веб-додаток повинен мати зрозумілу структуру, зручно організовану систему навігації, простий, світлий інтерфейс та можливість легко додати потрібні функції до коду програми, аби урізноманітнити функціонал цього блогу.

Оформлення сайту має бути якомога більш спрощеним та лаконічним та мати зручний інтерфейс. Загальна кольорова гамма повинна бути в світлих тонах, загальна компоновка сторінки не повинна змінюватися та бути постійною на всіх інших сторінках.

Шрифт має бути чітким та легко читаним навіть для людей з ослабленим зором. Окрім цього повинна матися база даних, в якій будуть зберігатися данні про користувачів та статті.

Блог використовується для вирішення таких задач:

- зацікавлення відвідувачів блогу до читання книжок;
- навчання гарно та чітко висловлювати свою думку;
- знайомства та підтримання відносин з користувачами

блогу.

Веб-додаток повинен мати такі сторінки:

- авторизації на сайті;
- сторінка реєстрації;
- сторінка нових записів;
- сторінка додавання запису;
- сторінка зі статтями певного жанру;

1.3 Функціонал веб-додатку

Функціонал сайту - це ключові можливості сайту по накопиченню, обробці і видачі інформації людям, які зайшли на сайт. Для розробки сайту дуже важливо чітко уявляти, які функції повинен виконувати сайт.

Функціональна частина блогу складається з:

- Функціонал незареєстрованого користувача;
- Функціонал звичайного зареєстрованого користувача;
- Функціонал адміністратора.

Незареєстрований користувач не повинен відчувати якихось труднощів в читанні статей в блозі, може передивлятися записи в блозі.

Зареєстрований користувач має легко додавати відгуки про книги, мати можливість додавати зображення до своєї статті, та переглядати останні новини на сайті.

Функціонал адміністратора:

Адміністратор повинен мати можливість додати статтю на сайт. Вхід в адміністраторську частину сайту здійснюється за допомогою спеціального адміністраторського паролю.

2 ВИБІР ЗАСОБІВ ДЛЯ РОЗРОБКИ ВЕБ-ДОДАТКУ ДЛЯ ВЕДЕННЯ ТЕМАТИЧНОГО БЛОГУ

На передпроектному етапі слід провести детальний аналіз існуючих програмних засобів, за допомогою котрих можливе виконання поставленої задачі.

2.1 Вибір мови програмування веб-додатку

PHP - це поширена мова програмування загального призначення з відкритим вихідним кодом. PHP сконструйований спеціально для ведення Web-розробок і його код може впроваджуватися безпосередньо в HTML.

Замість рутинного виведення HTML-коду командами мови, скрипт PHP містить HTML з вкрапленнями коду. Код PHP відділяється спеціальними початковим і кінцевим тегами `<? php і ?>`, Які дозволяють "перемикатися" в "PHP-режим" і виходити з нього.

PHP відрізняється від JavaScript тим, що PHP-скрипти виконуються на сервері і генерують HTML, який надсилається клієнту.

PHP вкрай простий для освоєння, але разом з тим здатний задовольнити запити професійних програмістів.

Плюси мови PHP складається в тому, що вона є традиційною, простою в використанні, гнучкою та безпечною.

PHP Data Objects - розширення для PHP, що надає розробнику простий і універсальний інтерфейс для доступу до різних баз даних.

PDO пропонує єдині методи для роботи з різними базами даних, хоча текст запитів може трохи відрізнитися. Так як багато СУБД реалізують свій діалект SQL, який в тій чи іншій мірі підтримує стандарти ANSI і ISO, то при використанні простих запитів можна домогтися сумісності між різними мовами. На практиці це означає, що можна досить легко перейти на іншу

СУБД, при цьому не змінюючи або частково змінюючи код програми.

Швидкість роботи і масштабованість: PDO не використовує абстрактних шарів для підключення до БД, на зразок ODBC, а використовує для різних БД їх «рідні» драйвери, що дозволяє добитися високої продуктивності. В даний час для PDO існують драйвери практично до всіх загальновідомим СУБД і інтерфейсів. Так само дозволяє працювати відразу з декількома базами даних одночасно.

Захист: З методів боротьби з sql-ін'єкціями з'явилося prepared. Prepared statement - це заздалегідь скомпільований SQL-вираз, яке може бути багаторазово виконано шляхом відправки сервера лише різних наборів даних. Додатковою перевагою є неможливість провести SQL-ін'єкцію через дані, які використовуються в placeholder'ах.

2.2 Вибір додаткових засобів для створення веб-додатку

2.2.1 Мова розмітки HTML

HTML - стандартизована мова розмітки документів у Всесвітній павутині. Більшість веб-сторінок містять опис розмітки на мові HTML (або XHTML). Мова HTML інтерпретується браузером; отриманий в результаті інтерпретації форматований текст відображається на екрані монітора комп'ютера або мобільного пристрою.

У всесвітній павутині HTML-сторінки, як правило, передаються браузером від сервера по протоколах HTTP або HTTPS, у вигляді простого тексту або з використанням шифрування.

HTML5 - мова для структурування та подання вмісту всесвітньої павутини. Це п'ята версія HTML. Хоча стандарт був завершений тільки в 2014 році (попередня, четверта, версія опублікована в 1999 році), ще з 2013 року браузерами оперативно здійснювалася підтримка, а розробниками - використання робочого стандарту. Мета розробки HTML5 - поліпшення

рівня підтримки мультимедіа-технологій з одночасним збереженням зворотної сумісності, комфортне читання коду для людини і простоти аналізу для парсерів.

2.2.2 Каскадні таблиці стилів

CSS (Cascading Style Sheets - каскадні таблиці стилів) - одна з базових технологій у сучасності Інтернету. Нечасто можна зустріти сайт, зверстаний без примінення CSS. Інструкції CSS зручно зберігати у вигляді окремого текстового файлу з розширенням .css, або у вигляді окремого текстового фрагмента на початку XHTML/HTML- документа. Основна ідея CSS в тому, щоб відокремити дизайн документа від його вмісту. CSS відповідає за оформлення і зовнішній вигляд, а XHTML/HTML - за зміст і логічну структуру документа.

2.2.3 Мова для управління базами даних SQL

SQL - формальна непроцедурна мова програмування, застосовується для створення, модифікації та управління даними в довільній реляційній базі даних, керованої відповідною системою управління базами даних.

SQL є перш за все інформаційно-логічним мовою, призначеним для опису, зміни і вилучення даних, що зберігаються в реляційних базах даних. SQL можна назвати мовою програмування, при цьому він не є Тьюринг-повним, але разом з тим стандарт мови специфікацією SQL / PSM передбачає можливість його процедурних розширень.

Кожна пропозиція SQL - це або запит даних з бази, або звернення до бази даних, яке призводить до зміни даних в базі. Відповідно до того, які

зміни відбуваються в базі даних, розрізняють такі типи запитів:

- запити на створення або зміна в базі даних нових або існуючих об'єктів (при цьому в запиті описується тип і структура створюваного або змінюваного об'єкта);
- запити на отримання даних;
- запити на додавання нових даних (записів);
- запити на видалення даних;
- звернення до СУБД.

2.3 Вибір програмного забезпечення

2.3.1 Програмне середовище OpenServer

Open Server Panel - це портативна серверна платформа і програмне середовище, створена спеціально для веб-розробників з урахуванням їх рекомендацій і побажань.

Програмний комплекс має багатий набір серверного програмного забезпечення, зручний, багатофункціональний продуманий інтерфейс, має потужні можливості з адміністрування та налаштування компонентів. Платформа широко використовується з метою розробки, налагодження і тестування веб-проектів, а так само для надання веб-сервісів в локальних мережах.

Хоча спочатку програмні продукти, що входять до складу комплексу, не розроблялись спеціально для роботи один з одним, така зв'язка стала дуже популярною серед користувачів Windows, в першу чергу через те, що вони отримували безкоштовний комплекс програм з надійністю на рівні Linux серверів.

Зручність і простота управління безумовно не залишать вас байдужими, за час свого існування Open Server зарекомендував себе як першокласний і надійний інструмент необхідний кожному веб-майстру.

2.3.2 Текстовий редактор Atom

Atom - безкоштовний текстовий редактор з відкритим вихідним кодом для macOS, Linux, Windows з підтримкою плагінів, написаних на Node.js, і вбудованих під керуванням Git Control. Більшість плагінів мають статус вільного програмного забезпечення, розробляються і підтримуються спільнотою.

2.3.3 Додаток для створення прототипів Balsamiq mockups

Balsamiq Mockups - графічний макет призначеного для користувача інтерфейсу і додаток для побудови каркасного сайту. Це дозволяє дизайнеру впорядковувати готові віджети за допомогою редактора WYSIWYG перетягуванням мишею. Додаток пропонується як в настільній версії, так і в якості плагіна для Google Drive, Confluence і JIRA.

2.4 Обґрунтування вибору засобів розробки

Проаналізувавши усі якості розглянутих засобів для створення тематичного блогу, оптимальним вибором для його розробки є мова програмування PHP та програмне забезпечення OpenServer, а також усі вищезазначені програми.

2.5 Висновки до розділу 2

У даному розділі проаналізувала засоби та програмне забезпечення для розробки тематичного блогу. Надала опис кожного з них.

3 РЕАЛІЗАЦІЯ ПРОТОТИПІВ ВЕБ-ДОДАТКУ ДЛЯ ВЕДЕННЯ ТЕМАТИЧНОГО БЛОГУ

Каркас сайту являється одним з найголовніших етапів у розробці веб-додатків та використовується для подальшого створення дизайну сторінок сайту. Для створення прототипів використовую Balsamiq Mockups, це проста в використанні та зі зрозумілим інтерфейсом програма. Прототип головної сторінки веб-додатку (рис. 3.1).

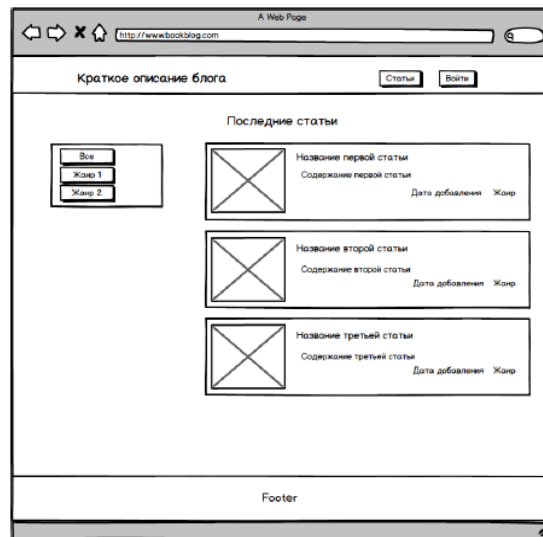


Рисунок 3.1 – Прототип головної сторінки

Прототип сторінки входу на сайт (рис. 3.2).

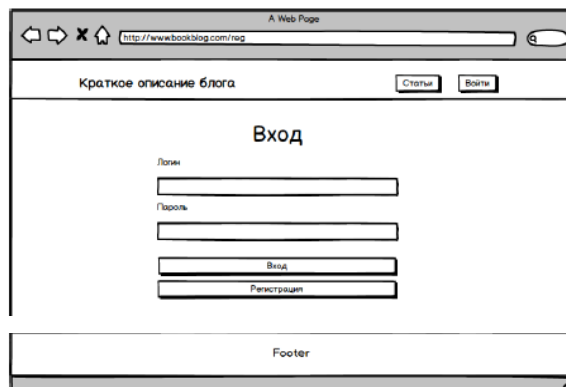


Рисунок 3.2 – Прототип сторінки входу на сайт

Прототип сторінки реєстрації на сайті (рис. 3.3).

The image shows a wireframe of a registration page within a browser window. The browser's address bar contains the URL `http://www.bookblog.com/auth`. The page header includes the text "Краткое описание блога" and two buttons: "Статьи" and "Выйти". The main content area is titled "Регистрация" and contains the following form elements:

- Two input fields for "Имя" (Name) and "Фамилия" (Surname).
- A single input field for "Логин" (Login).
- A single input field for "Пароль" (Password).
- A single input field for "E-mail".
- A "Зарегистрироваться" (Register) button.

The footer of the page contains the word "Footer".

Рисунок 3.3 – Прототип сторінки реєстрації

Сторінка додавання статті у блог (див. рис. 3.4).

The image shows a wireframe of an article submission page within a browser window. The browser's address bar contains the URL `http://www.bookblog.com/news.add`. The page header includes the text "Краткое описание блога" and two buttons: "Статьи" and "Выйти". The main content area is titled "Добавить статью" and contains the following form elements:

- An input field for "Название статьи" (Article Name).
- A large text area for "Текст статьи" (Article Text).
- A dropdown menu for "Жанр" (Genre).
- A "Добавить фото" (Add Photo) button.
- A "Опубликовать" (Publish) button.

The footer of the page contains the word "Footer".

Рисунок 3.4 – Прототип сторінки додавання статті

Прототипи створені у чорно-білому кольорі та вміщують в себе найбільш важливі елементи сторінок.

4 РОЗРОБКА ВЕБ-ДОДАТКУ ДЛЯ ВЕДЕННЯ ТЕМАТИЧНОГО БЛОГУ

4.1 Створення бази даних веб-додатку

Завдяки базам даних можна легко та зручно управляти даними, які в них занесені.

База даних має назву `news_site` та вміщує в себе такі таблиці (рис. 4.1):

- `category_news`;
- `news`;
- `page`;
- `user`.

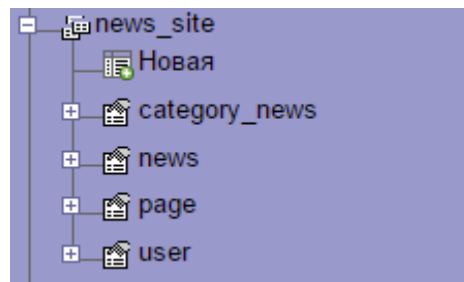


Рисунок 4.1 – Структура бази даних

Таблиця `category_news` (рис. 4.2) має поля `id` та `name`, в якому записується назва категорії статей, а саме жанр книги.

#	Имя	Тип	Сравнение	Атрибуты	Null	По умолчанию	Комментарии	Дополнительно
1	id	int(10)		UNSIGNED	Нет	Нет		AUTO_INCREMENT
2	name	text	utf8_general_ci		Нет	Нет		

Рисунок 4.2 – Структура таблиці `category_news`

Для додавання жанру до таблиці бази даних використовую SQL-запит, наприклад:

```
INSERT INTO `category_news`(`id`, `name`) VALUES ('7', 'Фантастика');
```

Таблиця `news` (рис. 4.3) має поля:

- `id`;
- `title`, в якому записується назва статті;

- article, в якому записується текст статті;
- date для запису дати;
- news_category_id, до якого жанру відноситься книга, про яку пишеться відгук;
- status;
- user_id, в який записується автор статті.

#	Имя	Тип	Сравнение	Атрибуты	Null	По умолчанию	Комментарии	Дополнительно
<input type="checkbox"/>	1 id	int(10)		UNSIGNED	Нет	Нет		AUTO_INCREMENT
<input type="checkbox"/>	2 title	text	utf8_general_ci		Нет	Нет		
<input type="checkbox"/>	3 article	text	utf8_general_ci		Нет	Нет		
<input type="checkbox"/>	4 date	timestamp			Нет	CURRENT_TIMESTAMP		
<input type="checkbox"/>	5 news_category_id	int(255)			Нет	Нет		
<input type="checkbox"/>	6 status	text	utf8_general_ci		Нет	Нет		
<input type="checkbox"/>	7 user_id	int(11)			Нет	1		

Рисунок 4.3 – Структура таблиці news

Таблиця page (рис. 4.4) має поле id.

#	Имя	Тип	Сравнение	Атрибуты	Null	По умолчанию	Комментарии	Дополнительно
<input type="checkbox"/>	1 id	int(10)		UNSIGNED	Нет	Нет		AUTO_INCREMENT

Рисунок 4.4 – Структура таблиці page

Таблиця user (рис. 4.5) має поля:

- id;
- first_name для запису ім'я користувача;
- last_name для запису прізвища;
- login для запису нікнейму, під яким користувач буде входити до сайту;
- password, в який вписується пароль користувача;
- email користувача;
- is_admin, поле в якому відмічається адміністратор сайту.

#	Имя	Тип	Сравнение	Атрибуты	Null	По умолчанию	Комментарии	Дополнительно
<input type="checkbox"/>	1 id	int(10)		UNSIGNED	Нет	Нет		AUTO_INCREMENT
<input type="checkbox"/>	2 first_name	text	utf8_general_ci		Нет	Нет		
<input type="checkbox"/>	3 last_name	text	utf8_general_ci		Нет	Нет		
<input type="checkbox"/>	4 login	text	utf8_general_ci		Нет	Нет		
<input type="checkbox"/>	5 password	text	utf8_general_ci		Нет	Нет		
<input type="checkbox"/>	6 email	text	utf8_general_ci		Нет	Нет		
<input type="checkbox"/>	7 is_admin	tinyint(1)			Нет	0		

Рисунок 4.5 – Структура таблиці user

4.2 Структура коду веб-додатку

Програма складається з моделей, контролерів та роутів, які виконують функції.

Наприклад, у моделі `Category.php` пописані функції, які вибирають усі, або по одній категорії з бази даних.

```
class Category
{
    public static function getAll(){
        $db = Db::getConnection();
        $result = $db->query('SELECT * FROM category_news');
        $i = 0;
        if($result){
            while($row=$result->fetch()){
                $categoryItems[$i]["id"] = $row["id"];
                $categoryItems[$i]["name"] = $row["name"];
                $i++;
            }
            return $categoryItems;
        }
    }

    public static function getById($id){
        $db = Db::getConnection();
        $result = $db->query('SELECT name FROM category_news
WHERE id='.$id);
        $result = $result->fetch(PDO::FETCH_ASSOC);
        $name = $result["name"];
        return $name;
    }
}
```

У моделі `News.php` виконуються такі функції, як додавання статті до бази даних, виборка та підрахунок статей у базі даних.

```

public static function addNews($title, $article, $category, $idAuthor){
    $db = Db::getConnection();
    $sql = "INSERT INTO news(title, article, news_category_id, user_id)
        VALUES (:title, :article, :category, :user_id)";
    $result=$db->prepare($sql);
    $result->bindValue(":title", $title);
    $result->bindValue(":article", $article);
    $result->bindValue(":category", $category);
    $result->bindValue(":user_id", $idAuthor);
    if($result->execute()){
        return $db->lastInsertId();
    }
    return 0;
}

```

У моделі User.php виконуються функції реєстрації користувачів, авторизації та виходу з блогу, функція перевірки чи є користувач адміністратором.

```

public static function register($firstName, $lastName, $login, $password,
$email){
    $db = Db::getConnection();
    $sql = 'INSERT INTO user(first_name, last_name, login, password,
email)
        VALUES (:first_name, :last_name, :login, :password, :email)';
    $result = $db->prepare($sql);
    $result->bindParam(':first_name', $firstName, PDO::PARAM_STR);
    $result->bindParam(':last_name', $lastName, PDO::PARAM_STR);
    $result->bindParam(':login', $login, PDO::PARAM_STR);
    $result->bindParam(':password', $password, PDO::PARAM_STR);
    $result->bindParam(':email', $email, PDO::PARAM_STR);
    return $result->execute(); }
public static function isAdmin($idUser){
    $db = Db::getConnection();
    $sql = 'SELECT is_admin FROM user WHERE id=:id';

```

```

$result = $db->prepare($sql);
$result->bindParam(":id", $idUser, PDO::PARAM_INT);
$result->execute();
if($result->fetchColumn()){
    return true;
}
return false;
}

```

У контролерах проводиться перевірка на заповнення полей у формах вводу, формування пагінації та інш.

Наприклад, у файлі `UserController.php` проводиться перевірка на правильне заповнення полей при реєстрації та вході на сайт.

```

public function actionRegister(){
    $view = new View();
    $firstName = "";
    $lastName = "";
    $login = "";
    $password = "";
    $email = "";
    if(isset($_POST["action"])){
        $firstName = $_POST["first_name"];
        $lastName = $_POST["last_name"];
        $login = $_POST["login"];
        $password = $_POST["password"];
        $email = $_POST["email"];
        $errors = false;
        if(!User::checkName($firstName)){
            $errors[] = "Имя пользователя слишком короткое";
        }
        if(!User::checkName($lastName)){
            $errors[] = "Фамилия пользователя слишком короткая";
        }
        if(!User::checkName($login)){
            $errors[] = "Логин слишком короткий";
        }
    }
}

```

```

}
if(!User::checkEmail($email)){
    $errors[] = "E-mail введен не коректно";
}
if(!User::checkPassword($password)){
    $errors[] = "Пароль слишком короткий";
}
if(!User::checkLoginExist($login)){
    $errors[] = "Такой логин уже существует";
}
if(!User::checkEmailExist($login)){
    $errors[] = "Такой e-mail уже существует";
}
$view->assign("errors", $errors);
if($errors == false){
    $result = User::register($firstName, $lastName, $login,
$password, $email);
    if($result){
        $view2 = new View();
        $view2->assign("register", true);
        $view2->display("auth/auth.php");
        return true;
    }
}

```

Також наявні файли, в яких написана на мові HTML розмітка сторінок (рис 4.6).

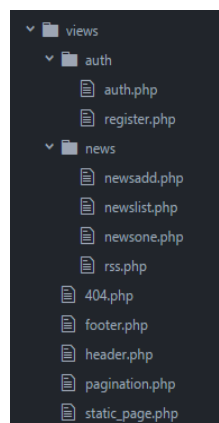


Рисунок 4.6 – Структура папки views

4.3 Висновки до розділу 4

В даному розділі приведена інформація щодо виконаної розробки веб-додатку. Створена база даних, в яку додаються дані з блогу, розроблено код програми на мові PHP з використанням HTML та CSS для надання сайту гарного вигляду.

5 РОБОТА ВЕБ-ДОДАТКУ ДЛЯ ВЕДЕННЯ ТЕМАТИЧНОГО БЛОГУ

5.1 Вигляд сторінок та їх функції

В даному розділі наводяться приклади використання веб-додатку.

1) Головна сторінка блогу (рис. 5.1), як і весь веб-додаток, виконано у світлих тонах, а саме він має білий колір фону та зелені елементи. Зверху розміщується назва сайту ліворуч, кнопки статей та входу праворуч. Ця сторінка ділиться на дві частини: з однієї сторони розташовані можливі жанри, з іншої самі статті.

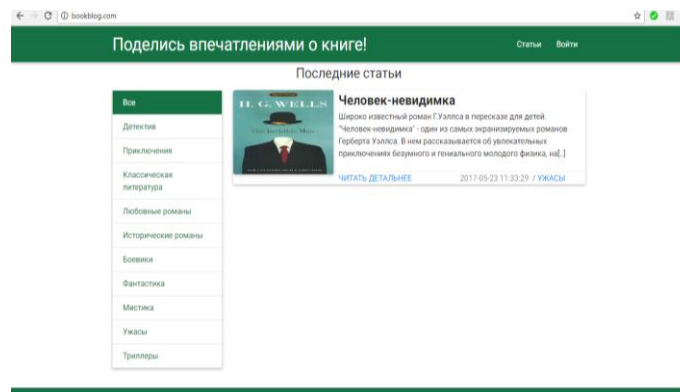


Рисунок 5.1 – Скриншот головної сторінки блогу

2) Сторінка входу на сайт (рис. 5.2) виконана в такому ж дизайні, посередині розташована форма входу з двома поля вводу та кнопки увійти та реєстрація:

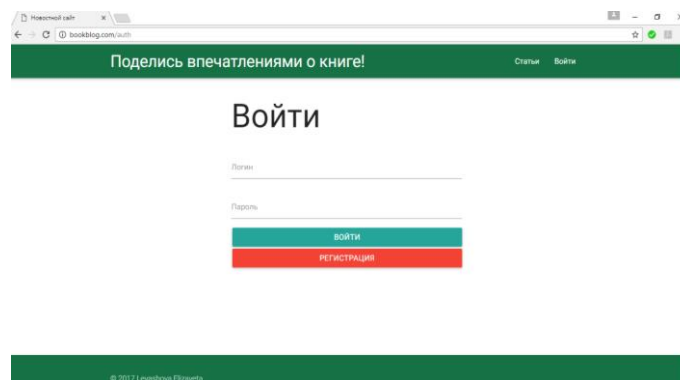


Рисунок 5.2 – Сторінка входу на сайт

3) На сторінці реєстрації (рис. 5.3) розташована форма для вводу даних: ім'я, прізвище, логін, пароль та email. Під формою розміщується кнопка зареєструватися.

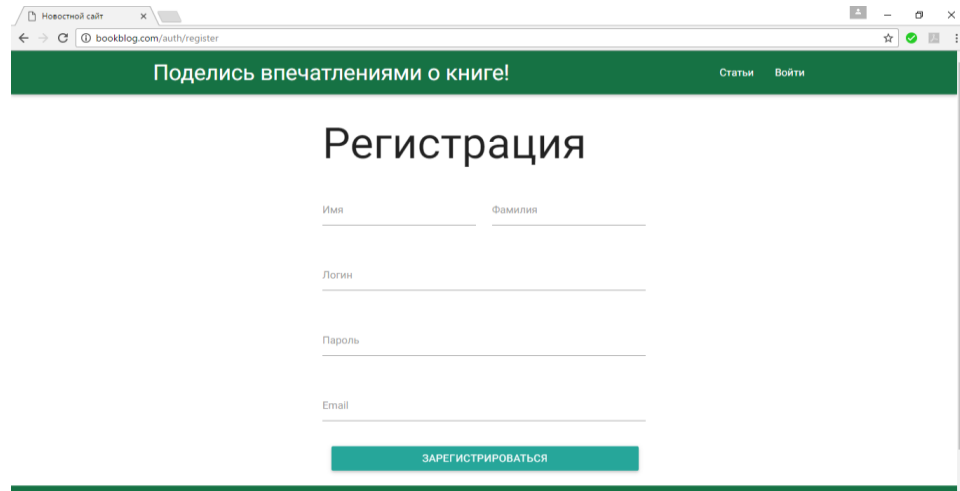


Рисунок 5.3 – Сторінка реєстрації на сайті

4) Після входу на сайт, на головній сторінці з'являється кнопка для додавання статті (рис. 5.4).

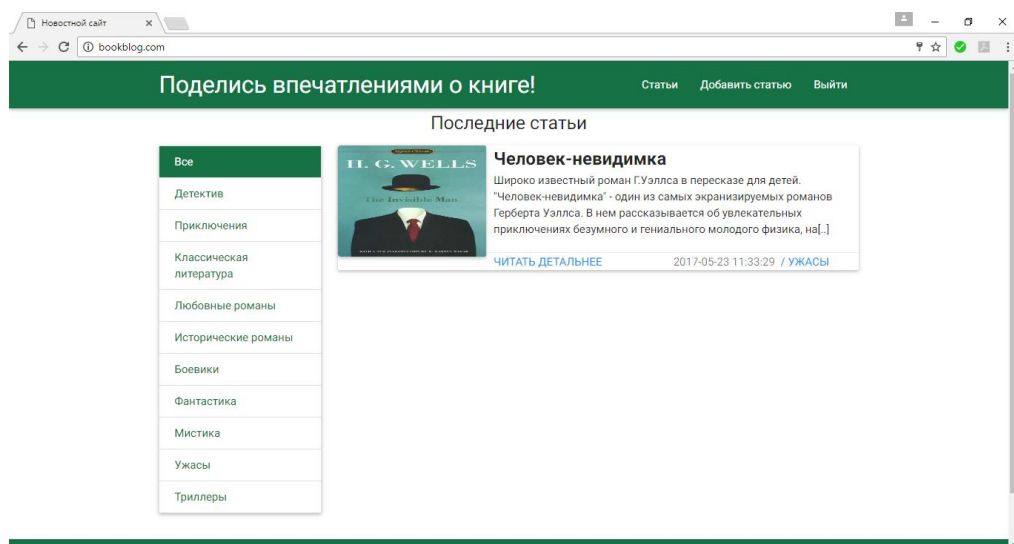


Рисунок 5.4 – Головна сторінка блогу

5) На сторінці додавання статті (див. рис. 5.5, 5.6) є форма для вводу назви статті, тексту, вибору жанру та можливість загрузити зображення.

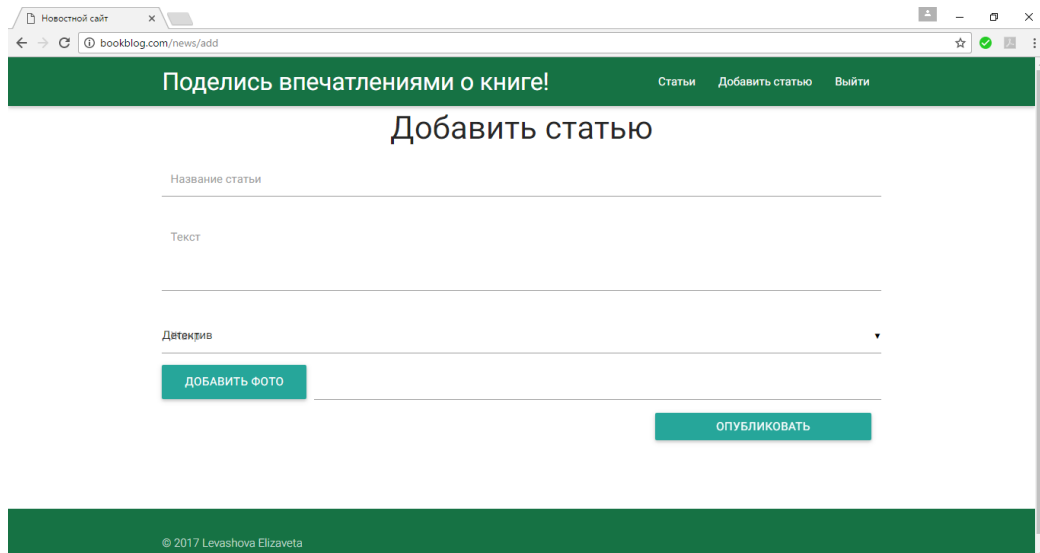


Рисунок 5.5 – Скриншот сторінки додавання статті

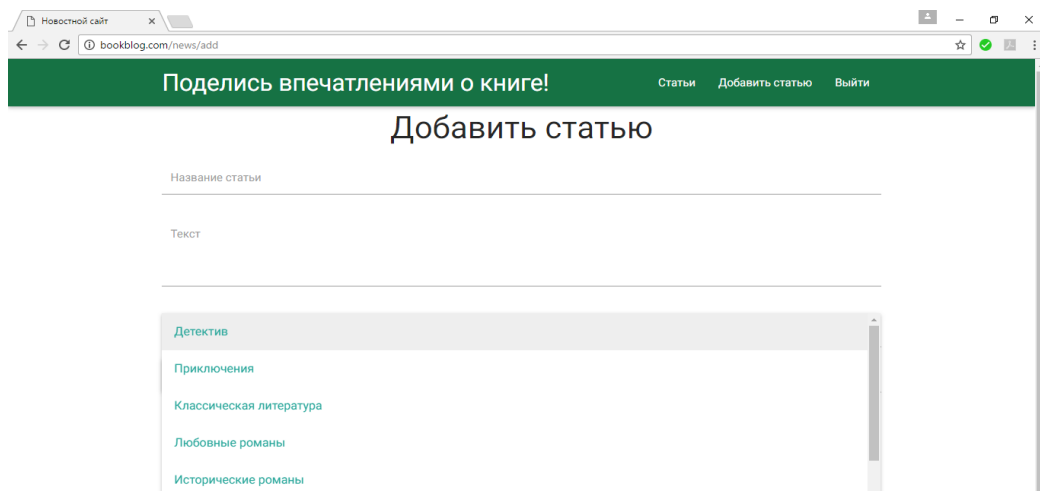


Рисунок 5.6 - Скриншот сторінки додавання статті

5.2 Реєстрація та додавання статті

1) Реєстрація у веб-додатку (рис. 5.7) проходить швидко та зрозуміло, після введення даних у форми, натиснувши кнопку зареєструватися, дані вносяться до таблиці user бази даних (рис. 5.8):

Новостной сайт x

bookblog.com/auth/register

Поделись впечатлениями о книге! Статьи Войти

Регистрация

Имя: Николай

Фамилия: Тарасов

Логин: tarasov1

Пароль:

Email:

ЗАРЕГИСТРИРОВАТЬСЯ

Рисунок 5.7 – Пример заполнения страницы регистрации

+ Параметры

		id	first_name	last_name	login
<input type="checkbox"/>		1	Elizaveta	Levashova	admin
<input type="checkbox"/>		4	Николай	Тарасов	tarasov1

Рисунок 5.8 – Добавлены записи в таблицу после регистрации

2) После заполнения формы добавления статьи (рис. 5.9), вся введенная информация записывается в таблицу news базы данных (рис. 5.10) и появляется на главной странице блога (рис. 5.10):

Новостной сайт x

bookblog.com/news/add

Поделись впечатлениями о книге! Статьи Добавить статью Войти

Добавить статью

Название статьи: Десять негритят

Текст:

Детективный роман английской писательницы Агаты Кристи, написанный в 1939 году. Изначально публиковался в США в семи частях в журнале «The Saturday Evening Post» с 20 мая по 1 июля 1939 года и параллельно точно так же выходил в Великобритании в «Daily Express» в 23 частях с 6 июня по 1 июля. В Великобритании отдельным изданием роман был выпущен в тот же год 6 ноября, в США – в январе 1940 года. Британский выпуск продавался в розницу за семь шиллингов и семь пенсов, американский – за два доллара. Из соображений политкорректности в США роман был издан под названием «И никого не стало» (англ. And Then There Were None), а все слова «негритята» в тексте были заменены на «маленькие индейцы»

Детектив

ДОБАВИТЬ ФОТО 119051.png

ОПУБЛИКОВАТЬ

© 2017 Levashova Elizaveta

Рисунок 5.9 – Пример заполнения полей

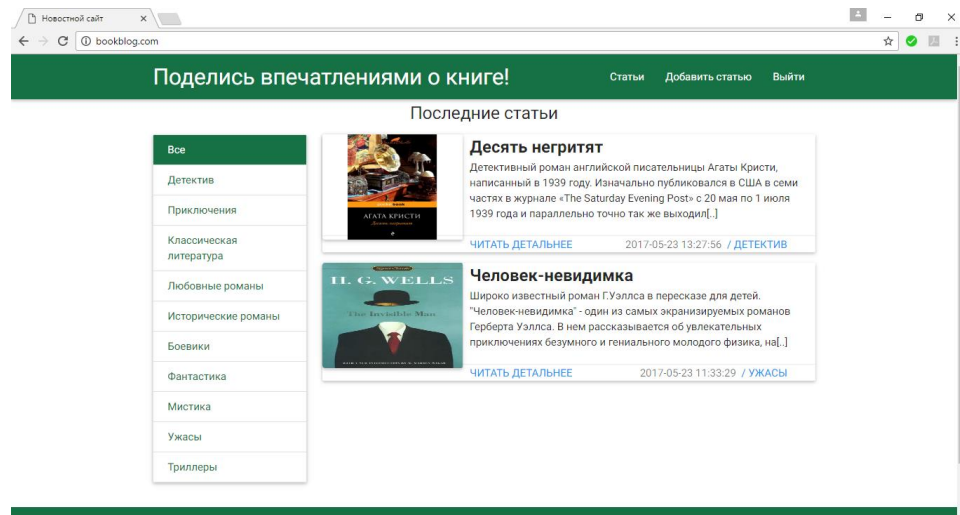


Рисунок 5.10 – Головна сторінка після додавання статті

id	title	article	date	news_category_id	status	user_id
10	Человек-невидимка	Широко известный роман Г.Уэллса в пересказе для де...	2017-05-23 11:33:29	9		1
11	Десять негритят	Детективный роман английской писательницы Агаты Кр...	2017-05-23 13:27:56	1		4

Рисунок 5.11 – Таблица після додавання статті

Висновок до розділу 5

У даному розділі навела приклади по використанню веб-додатку для ведення тематичного блогу. Описала як проводиться реєстрація та заповнення форм вводу при додаванні статті. Наводиться опис вигляду сторінок веб-додатку.

6 ОХОРОНА ПРАЦІ ТА БЕЗПЕКА В НАДЗВИЧАЙНИХ СИТУАЦІЯХ

В даному розділі проведено аналіз потенційних небезпечних та шкідливих виробничих факторів, причин пожеж. Розглянуті заходи, які дозволяють забезпечити гігієну праці і виробничу санітарію. На підставі аналізу розроблені заходи з техніки безпеки та рекомендації з пожежної профілактики.

Завданням даної роботи бакалавра було «Створення веб-додатку для ведення тематичного блогу», і як результат було створено веб-додаток для ведення тематичного блогу. За цим веб-додатком в подальшому розроблятиметься реальна система, яка значно полегшить процес ведення блогу. Так як в процесі проектування використовувався персональний комп'ютер, то аналіз потенційно небезпечних і шкідливих виробничих чинників виконується для персонального комп'ютера на якому буде використовуватися розроблений веб-додаток.

Системи безпеки праці та життєзабезпечення є одними з найголовніших, які використовуються на підприємствах, адже вони забезпечують умови для безпечної праці людини, правильне функціонування об'єктів. Під час виникнення надзвичайних ситуацій, забезпечення захисту життя людини та майна є одним із основних завдань. Вирішити ці питання неможливо без використання інформаційних технологій та розвитку автоматизації систем безпеки праці та життєзабезпечення.

Охорона праці - це система правових, соціально-економічних, організаційно-технічних, санітарно-гігієнічних і лікувально-профілактичних заходів та засобів, спрямованих на збереження життя, здоров'я і працездатності людини у процесі трудової діяльності.

Охорона праці тісно пов'язана з широким спектром дисциплін та сфер діяльності, серед яких медицина, психологія, епідеміологія, фізіотерапія, реабілітація, ергономіка, юриспруденція, людський фактор і багато іншого.

Охорона життя і здоров'я людини є пріоритетним напрямком соціальної політики держави. В Україні прийнято закон прямої дії «Про охорону праці», який регламентує захист конституційного права працівників на безпечні умови праці. Створена законодавча база захищеності життя і здоров'я працівників.

Дія Закону України «Про охорону праці» поширюється на всіх юридичних та фізичних осіб, які відповідно до законодавства використовують найману працю, та на всіх працюючих.

Законодавство України про охорону праці складається із загальних законів України та спеціальних законодавчих актів. Загальними законами України, що визначають основні положення з охорони праці є Конституція України, Закон України «Про охорону праці», Кодекс законів про працю (КЗпП), Закон України «Про загальнообов'язкове державне соціальне страхування від нещасного випадку на виробництві та професійного захворювання, які спричинили втрату працездатності».

Тема охорони праці є дуже актуальною в наш час, бо вирішення цього питання в ході проектної розробки має на меті зменшити виробничі травми та професійні захворювання, які виникають в результаті дії небезпечних та шкідливих факторів, таких як вплив шкідливих речовин, неналежні умови праці, погана освітленість робочого місця, шум та вібрація, оптимізувати метеорологічні умови на робочому місці працівників.

6.1 Загальні питання з охорони праці

6.1.1 Правові та організаційні основи охорони праці

Законодавство України про охорону праці являє собою систему взаємозв'язаних нормативно-правових актів, що регулюють відносини у галузі охорони праці. Воно складається з Закону України «Про охорону праці», Кодексу законів про працю України, Закону України "Про загальнообов'язкове державне соціальне страхування від нещасного випадку

на виробництві та професійного захворювання, які спричинили втрату працездатності" та прийнятих відповідно до них нормативно-правових актів.

Базується законодавство України про охорону праці на конституційному праві всіх громадян України на належні, безпечні і здорові умови праці, гарантовані статтею 43 Конституції України. Ця ж стаття встановлює також заборону використання праці жінок і неповнолітніх на небезпечних для їхнього здоров'я роботах. Ст. 45 Конституції гарантує право всіх працюючих на щотижневий відпочинок та щорічну оплачувану відпустку, а також встановлення скороченого робочого дня щодо окремих професій і виробництв, скороченої тривалості роботи у нічний час.

Інші статті Конституції встановлюють право громадян на соціальний захист, що включає право забезпечення їх у разі повної, часткової або тимчасової втрати працездатності (ст. 46); охорону здоров'я, медичну допомогу та медичне страхування (ст. 49); право знати свої права та обов'язки (ст. 57) та інші загальні права громадян, в тому числі, право на охорону праці.

Основоположним документом в галузі охорони праці є Закон України «Про охорону праці», який визначає основні положення щодо реалізації конституційного права працівників на охорону їх життя і здоров'я у процесі трудової діяльності, на належні, безпечні і здорові умови праці, регулює за участю відповідних державних органів відносини між роботодавцем і працівником з питань безпеки, гігієни праці та виробничого середовища і встановлює єдиний порядок організації охорони праці в Україні. Інші нормативні акти мають відповідати не тільки Конституції та іншим законам України, але, насамперед, цьому Законові.

Закон України "Про охорону праці", прийнятий у 1992 році, вперше став таким правовим актом, який орієнтує законодавство на захист інтересів громадянина, віддаючи перевагу в цій важливій сфері правовому регулюванню на відміну від адміністративного, що існувало раніше. У листопаді 2002 р. Верховна Рада України прийняла нову редакцію цього

закону. Закон “Про охорону праці” відповідає діючим конвенціям і рекомендаціям Міжнародної організації праці, іншим міжнародним правовим нормам у цій галузі. Згідно ст. 1 цього Закону “охорона праці – це система правових, соціально-економічних, організаційно-технічних, санітарно-гігієнічних і лікувально-профілактичних заходів та засобів, спрямованих на збереження життя, здоров'я і працездатності людини у процесі трудової діяльності”.

Закон визначає основні засади державної політики в галузі охорони праці, які базуються на принципах:

- пріоритету життя і здоров'я працівників, повної відповідальності роботодавця за створення належних, безпечних і здорових умов праці;
- підвищення рівня промислової безпеки шляхом забезпечення суцільного технічного контролю за станом виробництв, технологій та продукції, а також сприяння підприємствам у створенні безпечних та нешкідливих умов праці;
- комплексного розв'язання завдань охорони праці на основі загальнодержавної, галузевих, регіональних програм з цього питання та з урахуванням інших напрямів економічної і соціальної політики, досягнень в галузі науки і техніки та охорони довкілля;
- соціального захисту працівників, повного відшкодування шкоди особам, які потерпіли від нещасних випадків на виробництві та професійних захворювань;
- встановлення єдиних вимог з охорони праці для всіх підприємств та суб'єктів підприємницької діяльності незалежно від форм власності та видів діяльності;
- адаптації трудових процесів до можливостей працівника з урахуванням його здоров'я та психологічного стану;
- використання економічних методів управління охороною праці, участі держави у фінансуванні заходів щодо охорони праці, залучення добровільних внесків та інших надходжень на ці цілі, отримання яких не

суперечить законодавству;

- інформування населення, проведення навчання, професійної підготовки і підвищення кваліфікації працівників з питань охорони праці;

- забезпечення координації діяльності органів державної влади, установ, організацій, об'єднань громадян, що розв'язують проблеми охорони здоров'я, гігієни та безпеки праці, а також співробітництва і проведення консультацій між роботодавцями та працівниками (їх представниками), між усіма соціальними групами під час прийняття рішень з охорони праці на місцевому та державному рівнях;

- використання світового досвіду організації роботи щодо поліпшення умов і підвищення безпеки праці на основі міжнародного співробітництва.

До нормативно-правових актів з охорони праці відносяться, згідно ст. 27 Закону, правила, норми, регламенти, положення, стандарти, інструкції та інші документи, обов'язкові до виконання.

6.1.2 Організаційно-технічні заходи з безпеки праці

Технічні заходи - технічні засоби, що забезпечують безпечні і нешкідливі умови праці, та пов'язані з впровадженням нового обладнання, пристроїв і приладів безпеки і безпечною експлуатацією засобів виробництва.

Нормативно-методичні заходи:

- розробка посібників і рекомендацій;
- розробка нормативно-правової бази з охорони праці на підприємстві;

- забезпечення необхідною нормативно-правовою документацією функціональних служб, окремих структурних підрозділів та робочих місць;

- забезпечення програм і розробка методик навчання з питань охорони праці;

- розробка розділів охорони праці в посадових інструкціях,

інструкціях за професіями;

- перегляд НПАОП підприємства.

Організаційні заходи:

- контроль за технічним станом обладнання, інструментів, будівель і споруд;
- контроль за дотриманням вимог нормативних документів з охорони праці;
- нагляд за обладнанням підвищеної небезпеки;
- організація навчання, перевірка знань з питань охорони праці і інструктажів робітників підприємства;
- контроль за виконанням технологічного процесу відповідно до вимог охорони праці;
- організація належних умов до проїздів і проходів відповідно до вимог охорони праці;
- забезпечення працівників засобами індивідуального та колективного захисту;
- забезпечення відповідними знаками безпеки, плакатами.

6.2 Аналіз стану умов праці

6.2.1 Вимоги до приміщень

Проаналізувала стан приміщення, в якому виконувалася сама робота та її оформлення (див. табл. 6.1).

Таблиця 6.1 – Стан приміщення

Найменування	Значення
Довжина, м	4
Ширина, м	3
Висота, м	3
Площа, м ²	12
Об'єм, м ³	36

1. Місце праці знаходиться на відстані 1 м від стіни з вікном;
2. Кількість робочих місць з комп'ютерами – одне.

Згідно з [15] розмір площі для одного робочого місця оператора персонального комп'ютера має бути не менше 6 кв. м, а об'єм — не менше 20 куб. м. Отже, дане приміщення цілком відповідає зазначеним нормам.

Для забезпечення потрібного рівного освітленості кімната має вікно та систему загального рівномірного освітлення, що встановлена на стелі.

6.2.2 Вимоги до організації місця праці

Місце праці – це зона професійних дій працівника, обладнана для виконання певних операцій виробничого процесу, де взаємодіють три головні елементи праці — предмет, засоби і суб'єкт праці. На одному місці праці можуть працювати два або кілька працівників, які виконують спільне завдання.

При порівнянні відповідності характеристик робочого місця нормативним основні вимоги до організації робочого місця за [14] і відповідними фактичними значеннями для робочого місця, констатуємо повну відповідність.

Таблиця 6.2 - Характеристика робочого місця

Найменування параметра	Фактичне значення	Нормативне значення
Висота робочої поверхні, мм	730	680 ÷ 800
Висота простору для ніг, мм	610	не менше 600
Ширина простору для ніг, мм	600	не менше 500
Глибина простору для ніг, мм	700	не менше 650
Висота поверхні сидіння, мм	430	400 ÷ 500
Ширина сидіння, мм	420	не менше 400

Подовження таблиці 6.2

Глибина сидіння, мм	450	не менше 400
Висота поверхні спинки, мм	360	не менше 300
Ширина опорної поверхні спинки, мм	380	не менше 380
Радіус кривини спинки в горизонтальній площині, мм	400	400
Відстань від очей до екрану дисплея, мм	800	700 ÷ 800

Положення тіла при роботі:

- ступні ніг — на підлозі;
- стегна — в горизонтальній площині;
- передпліччя — вертикально;
- лікті знаходяться на столі;
- нахил голови — 15–20° відносно вертикальної площини.

Так як користування ПК є головним видом діяльності, то ПК і його периферійні пристрої розміщують на одному столі з лівого боку.

Для зниження статичного напруження м'язів рук застосовуються знімні підлокітники довжиною 250 мм, шириною 48 мм.

Клавіатура розміщується на поверхні столу.

Отже, за всіма параметрами робоче місце відповідає нормативним вимогам.

Температура в приміщенні протягом року коливається у межах 18–22°C, відносна вологість — близько 55%. Швидкість руху повітря не перевищує 0,1 м/с. Шум знаходиться на рівні 50 дБА. Система вентилявання приміщення — природна неорганізована, а опалення — централізоване.

Наявна електрична мережа з напругою 220 В, яка створює небезпеку ураження електричним струмом. ПК та периферійні пристрої можуть бути джерелами електромагнітних випромінювань, аерозолів та шкідливих речовин (часток тонеру, оксидів нітрогену та озону).

Наявна аптечка для надання долікарської допомоги, а також у кабінеті роблять вологе прибирання та щоденно провітрюють приміщення.

6.2.3 Навантаження та напруженість процесу праці

Напруженість праці – характеристика трудового процесу, що відображає навантаження переважно на центральну нервову систему, органи чуттів, емоційну сферу працівника. До факторів, що характеризують напруженість праці, відносяться: інтелектуальні, сенсорні, емоційні навантаження, ступінь монотонності навантажень, режим роботи.

Робота з комп'ютером відноситься до легкої категорії робіт, а саме Іа, тому енерговитрати складають до 120 ккал за годину. Роботу виконують сидячи з періодичним ходінням. Щодо характеру організування виконання дипломної роботи, то він підпадає під нав'язаний режим, оскільки певні розділи роботи необхідно виконати у встановлені конкретні терміни. За ступенем нервово-психічної напруги виконання роботи можна віднести до II – III ступеня і кваліфікувати як помірно напружений – напружений за умови успішного виконання поставлених завдань.

Найбільшому ризику виникнення різноманітних порушень піддаються: органи зору, м'язово скелетна система, нервово-психічна діяльність, репродуктивна функція у жінок.

Тобто наявне психофізіологічні небезпечні та шкідливі фактори:

а) фізичного перевантаження:

- статичного;
- динамічного;

б) нервово-психічного перевантаження:

- розумового перенапруження;
- монотонності праці;
- перенапруження аналізаторів;

- емоційних перевантажень.

Робота за дипломним проектом визнано займає 50% часу робочого дня, тому перерва між роботою складає 15 хв через кожен годину роботи.

6.3 Виробнича санітарія

6.3.1 Аналіз небезпечних та шкідливих факторів при виробництві (експлуатації) виробу

Аналіз небезпечних та шкідливих виробничих факторів виконується у табличній формі (табл. 6.3).

Робочими характеристиками даного персонального комп'ютера є:

- робоча напруга $U=+220\text{В} \pm 5\%$;
- робочий струм $I=2\text{А}$;
- споживана потужність $P=350\text{ Вт}$.

Таблиця 6.3 - Аналіз небезпечних та шкідливих виробничих факторів

<i>Небезпечні і шкідливі виробничі фактори</i>	<i>Джерела факторів (види робіт)</i>	<i>Кількіс на оцінка</i>	<i>Нормативні документи</i>
1	2	3	4
фізичні			
- підвищений рівень шуму на робочому місці		2	[16]
- підвищений рівень іонізуючого випромінювання в робочій зоні	Робота персонального комп'ютеру	2	[15] [17]
- недостатнє освітлення робочої зони	порушення гігієнічних параметрів виробничого середовища	3	[18]

Подовження таблиці 6.3

- недостатність природного світла	порушення умов праці (вимог до приміщень)	2	[18]
- підвищений рівень електромагнітного випромінювання	Робота персонального комп'ютеру	2	[17]
- підвищений рівень напруги електричної мережі, замикання якої може відбутися через тіло людини	Експлуатація електричних приборів	4	[19] [20]
хімічні:			
- загазованість повітря робочої зони, яка впливає на організм людини через органи дихання та надає токсичну і канцерогенну дію	оплавлення електричних і комутаційних кабелів, резисторів - CO, CO ₂ , SO ₂ , P ₂ O ₅ , H ₂ S, HCl, H, NH ₃ , ClF ₃ , F ₂ O ₂ , F ₂ O ₃ , SeO ₂ . SeF ₆ , TeF ₆ , COCl ₂ , SO ₂ F ₂ , інш.	3	[13] [21] [22] [23]
психофізіологічні:			
- нервово-психічна перевантаження (розумове, перенапруження аналізаторів-зорових)	- пошук інформації для постановки теми; - пошук та аналіз аналогів і літератури; - пошук наявних технологій, моделювання та аналіз алгоритмів; - виконання роботи за темою диплома, тестування; - оформлення роботи	4	[10] [14]
- фізичні (статичне – сидіння)	організації робочого часу - безпервна робота)	2	[10] [14]

Робоче місце відповідає вимогам Державних санітарних правил і нормам роботи з візуальними дисплейними терміналами електронно-обчислювальних машин [14].

6.3.2 Пожежна безпека

Небезпека загоряння пов'язана з особливістю комп'ютерів - із значною кількістю щільно розташованих на монтажній платі і блоках електронних вузлів і схем, електричних і комутаційних кабелів, резисторів, конденсаторів, напівпровідникових діодів і транзисторів. Надійна робота окремих елементів і мікросхем в цілому забезпечується тільки в певних інтервалах температури, вологості і при заданих електричних параметрах. При відхиленні реальних умов експлуатації від розрахункових можуть виникнути пожежонебезпечні ситуації.

Тому передбачаються способи і засоби запобігання утворення пожежі, як швидке відключення можливих джерел запалення.

Потенційними джерелами запалювання можуть бути:

- 1) іскри і дуги короткого замикання;
- 2) перегріву від тривалого перевантаження.

Причинами можливого загоряння і пожежі можуть бути:

- 1) несправність електроустановки;
- 2) коротке замикання в електричних мережах.

6.3.3 Електробезпека

На робочому місці виконуються наступні вимоги електробезпеки: ПК, периферійні пристрої та устаткування для обслуговування, електропроводи і кабелі за виконанням та ступенем захисту відповідають класу зони за ПУЕ

(правила улаштування електроустановок), мають апаратуру захисту від струму короткого замикання та інших аварійних режимів. Лінія електромережі для живлення ПК, периферійних пристроїв і устаткування для обслуговування, виконана як окрема групова три-провідна мережа, шляхом прокладання фазового, нульового робочого та нульового захисного провідників. Нульовий захисний провідник використовується для заземлення (занулення) електроприймачів. Захисне заземлення включає в себе заземлюючих пристроїв і провідник, який з'єднує заземлюючий пристрій з обладнанням, яке заземлюється - заземлюючий провідник.

6.4 Гігієнічні вимоги до параметрів виробничого середовища

6.4.1 Мікроклімат

Мікроклімат виробничих приміщень характеризують температурою, вологістю та швидкістю руху повітря, а також інтенсивністю радіації, переважно в інфрачервоній та ультра-фіолетовій областях спектру електромагнітних випромінювань (табл. 6.4).

Таблиця 6.4 – Мікроклімат приміщення

Категорія робіт	Температура		Вологість повітря		Швидкість руху повітря	
	Наявна	Оптимальна	Наявна	Оптимальна	Наявна	Оптимальна
Ia	20 ⁰ C	23–25 ⁰ C	55 %	40-60 %	0,1 м/с	0,1 м/с

Дане приміщення обладнане системами опалення та кондиціонування повітря. У приміщенні на робочому місці забезпечуються оптимальні значення параметрів мікроклімату: температури, відносної вологості й рухливості повітря у відповідності до [15].

6.4.2 Освітлення

Освітлення відіграє важливу роль у житті людини. Біля 90% інформації сприймається через зоровий канал, тому правильно виконане раціональне освітлення має важливе значення для виконання всіх видів робіт. Світло є не тільки важливою умовою роботи зорового аналізатора, але й біологічним фактором розвитку організму людини в цілому. Стан освітлення виробничих приміщень відіграє важливу роль і для попередження виробничого травматизму. Багато нещасних випадків на виробництві стається через погане освітлення.

Залежно від джерела світла виробниче освітлення може бути трьох видів:

1. Природне - це пряме або відбите світло сонця (небосхилу), що освітлює приміщення через світлові прорізи в зовнішніх огорожувальних конструкціях.
2. Штучне - здійснюється штучними джерелами світла (лампами розжарювання або газорозрядними) і призначене для освітлення приміщень у темні години доби, або таких приміщень, які не мають природного освітлення.
3. Сполучене (суміщене) - одночасне поєднання природного і штучного освітлення.

Розрахунок природного освітлення:

$$S_b = \left(\frac{1}{5} \div \frac{1}{10} \right) \cdot S_n, \quad (6.1)$$

де S_b – площа віконних прорізів, m^2 ;

S_n – площа підлоги, m^2 ;

$$S_b = \left(\frac{1}{8}\right) * 12 = 1,5 \text{ м}^2$$

Розрахунок штучного освітлення:

$$n = \frac{E \cdot S \cdot Z \cdot K}{F \cdot U \cdot M}, \quad (6.2)$$

де E – мінімально допустима освітленість робочої поверхні, визначається нормами

S – освітлювана площа, m^2 ;

F – світловий потік однієї лампи;

K – коефіцієнт запасу, для люмінесцентних ламп приймаємо $K=1,5$ (з малим виділенням забруднення);

Z – поправочний коефіцієнт ($Z = 1,15$ для ламп розжарювання та ДРЛ; $Z = 1,1$ для люмінесцентних ламп);

U – коефіцієнт використання освітленості установки, що залежить від конструкції світильника, коефіцієнтів відбиття стелі і стін, а також показника i , для люмінесцентних ламп $U=0,55-0,6$;

M – кількість люмінесцентних ламп в світильнику.

$E = 300$ лк;

$S = 12 \text{ м}^2$;

$F = 1000$ лм для ЛД – 20;

$$n = \left(\frac{300 * 12 * 1,1 * 1,5}{1000 * 0,55 * 2}\right) \sim 5$$

6.5 Шум та вібрація, електромагнітне випромінювання

Шкідливий та небезпечний вплив шуму на організм людини встановлено тепер з повною достовірністю. Ступінь такого впливу, переважно, залежить від рівня та характеру шуму, форми та тривалості впливу, а також індивідуальних особливостей людини. Численні дослідження підтвердили той факт, що шум належить до загальнофізіологічних подразників, які за певних обставин можуть впливати на більшість органів та систем організму людини. Так, за даними медиків дія шуму може спричинити нервові, серцево-судинні захворювання, виразкову хворобу, порушення обмінних процесів та функціонування органів слуху тощо.

Рівень шуму, що супроводжує роботу користувачів персональних комп'ютерів (зумовлений як роботою системних блоків, клавіатури, так і друкуванням на принтерах, а також зовнішніми чинниками), коливається у межах 50–65 дБА.

Дослідження вчених за останні 20 років показали, що електромагнітні поля, створені технічними системами, навіть у сотні разів слабші природного поля Землі, можуть бути небезпечними для здоров'я людини. Джерелами електромагнітних випромінювань є мережі живлення (частота 50 Гц), система рядкового розгорнення (2-400 кГц), блок модуляції променя (5-10 МГц).

6.6 Вентилювання

Для забезпечення у виробничих приміщеннях оптимальних параметрів мікроклімату та належної чистоти повітря (граничнодопустимих концентрацій шкідливих речовин) передбачають вентиляцію приміщень,

яке здійснюють за допомогою сукупності пристроїв для транспортування, подання та видалення повітря.

Здійснюється провітрювання приміщення, в залежності від погодних умов, тривалістю 30 хв. Найкращий обмін повітря здійснюється при наскрізному провітрюванні.

6.7 Заходи з організації виробничого середовища та попередження виникнення надзвичайних ситуацій

Розмір робочого місця, стіл та параметри стільця відповідають встановленим нормам.

Для зменшення негативного впливу на стан здоров'я працівників різних факторів ризику, пов'язаних з роботою на ПК, робляться перерви через кожний час безперервної роботи – 15 хвилин.

Так як об'єм приміщення становить 36 м³, то потрібно подати не менш як 20 м³/год повітря.

Вимоги безпеки при надзвичайних ситуаціях:

- 1) При раптовому припиненні подачі електричної енергії вимкнути всі пристрої ПК в такій послідовності: периферійні пристрої, системний блок, стабілізатор (або блок безперервного живлення). Витягнути вилки з розеток. При наявності ознак горіння (дим, запах горілого) необхідно вимкнути всі пристрої ПК, знайти місце загоряння і виконати всі можливі заходи для його ліквідації, попередивши терміново про це керівництво. У випадку виникнення пожежі негайно попередити про це пожежну частину.
- 2) При замиканні, перевантаженні електричного струму на електричному обладнанні, внаслідок ураження грозової блискавки та ймовірної небезпеки ураженням електричним струмом,

приймають наступне:

- попередження замикання здійснюється правильним вибором, монтажем експлуатації мереж;
- застосування захисту схем у вигляді швидкодіючих реле, а також вимикачів, плавких запобіжників, автоматичних вимикачів.

Розрахунок захисного заземлення

Згідно з класифікацією приміщень за ступенем небезпеки ураження електричним струмом [12], приміщення в якому проводяться всі роботи відноситься до першого класу (без підвищеної небезпеки). Під час роботи використовуються електроустановки з напругою живлення 220 В. Опір контура заземлення повинен мати не більше 4 Ом.

Послідовність розрахунку.

- 1) Визначається необхідний опір штучних заземлювачів $R_{шт.з.}$:

$$R_{шт.з.} = \frac{R_d \cdot R_{пр.з.}}{R_{пр.з.} - R_d}, \quad (6.3)$$

де $R_{пр.з.}$ – опір природних заземлювачів; R_d – допустимий опір заземлення.

Якщо природні заземлювачі відсутні, то $R_{шт.з.} = R_d$.

Так як природні заземлювачі відсутні, то опір штучних заземлювачів дорівнює допустимому опору заземлення, а саме 10 Ом.

- 2) Опір заземлення в значній мірі залежить від питомого опору ґрунту ρ , Ом·м. Питомий опір ґрунту залежить від характеру ґрунту, а також від пори року. Найбільшу величину він має в холодний період у північних районах при промерзанні ґрунту або в теплий період в південних районах, коли ґрунт найбільш сухий.

Наявна садова земля, тому питомий опір ґрунту дорівнює $50 \text{ Ом}\cdot\text{м}$.

1. У випадку малого відсотка вмісту вологи в ґрунті можливі більші значення опорів.

2. Питомі опори ґрунтів коливаються протягом року, що враховують при розрахунках введенням так званих сезонних коефіцієнтів опору ґрунту.

3) Розрахунковий питомий опір ґрунту, $\rho_{\text{розр.}}$, $\text{Ом}\cdot\text{м}$, визначається за формулою:

$$\rho_{\text{розр.}} = \psi \cdot \rho, \quad (6.4)$$

де ψ – коефіцієнт сезонності; ρ – табличне значення питомого опору ґрунту, $\text{Ом}\cdot\text{м}$.

Кліматична зона I, це значить, що $\psi_{\text{в}}$ для вертикального електрода довжиною $l_{\text{в}}=3 \text{ м}$ при нормальній вологості землі дорівнює 1,7, а $\psi_{\text{г}}$ для горизонтального електрода довжиною $l_{\text{г}}=10 \text{ м}$ дорівнює 5,5, тому:

$$\rho_{\text{розр. в}} = 1,7 \cdot 50 = 85 \text{ Ом}\cdot\text{м}$$

$$\rho_{\text{розр. г}} = 5,5 \cdot 50 = 275 \text{ Ом}\cdot\text{м}$$

4) Розраховується опір розтікання струму вертикального заземлювача $R_{\text{в}}$, Ом .

$$R_{\text{в}} = \frac{\rho_{\text{розр.}}}{2 \cdot \pi \cdot l_{\text{в}}} \cdot \left(\ln \frac{2 \cdot l_{\text{в}}}{d} + \frac{1}{2} \cdot \ln \frac{4 \cdot t + l_{\text{в}}}{4 \cdot t - l_{\text{в}}} \right), \quad (6.5)$$

де $l_{\text{в}}$ – довжина вертикального заземлювача; d – діаметр стержня, $d = 0,05 \text{ м}$; t – відстань від поверхні землі до середини заземлювача:

$$t = h_{\text{в}} + \frac{l_{\text{в}}}{2},$$

де $h_{\text{в}}$ – глибина закладання вертикальних заземлювачів (прийняти за $0,8 \text{ м}$).

$$t = 0,8 + \frac{3}{2} = 2,3$$

$$R_B = \left(\frac{85}{2} * 3,14 * 3 \right) * \left(\ln \left(\frac{2 * 3}{0,05} \right) + \left(\frac{1}{2} \right) * \ln \left(\frac{4 * 2,3 + 3}{4 * 2,3 - 3} \right) \right) = 23,130 \text{ м}$$

5) Визначається теоретична кількість вертикальних заземлювачів n штук, без урахування коефіцієнта використання η_B :

$$n = \frac{2 \cdot R_B}{R_D}. \quad (6.6)$$

Відповідно до розрахованого значення n визначається коефіцієнт використання вертикальних заземлювачів η_B .

$$n = \frac{(2 * 23,13)}{10} = 4,62$$

Прийmemo кількість заземлювачів за 4, тому η_B для розташованих в ряд буде дорівнювати 0,65.

б) Визначається необхідна кількість вертикальних заземлювачів з урахуванням коефіцієнта використання n_B , шт:

$$n_B = \frac{2 \cdot R_B}{R_D \cdot \eta_B}. \quad (6.7)$$

$$n_B = \frac{(2 * 23,13)}{10 * 0,65} = 7,11$$

7) Визначається довжина з'єднувальної стрічки горизонтального заземлювача l_C , м:

$$l_C = 1,05 \cdot L_B \cdot (n_B - 1), \quad (6.8)$$

де L_B – відстань між вертикальними заземлювачами, (прийняти за $L_B = 3\text{ м}$);

n_B – необхідна кількість вертикальних заземлювачів, з урахуванням коефіцієнта використання η_B .

$$l_c = 1,05 * 3 * (7 - 1) = 18,9 \text{ м}$$

8) Визначається опір розтіканню струму горизонтального заземлювача (з'єднувальної стрічки) R_Γ , Ом:

$$R_\Gamma = \frac{\rho_{\text{розр.}\Gamma}}{2 \cdot \pi \cdot l_c} \cdot \ln \frac{2 \cdot l_c^2}{d_{\text{см}} \cdot h_\Gamma},$$

(6.9)

де d – еквівалентний діаметр смуги шириною b , $d = 0,95b$, $b = 0,15 \text{ м}$;

h_Γ – глибина закладання горизонтальних заземлювачів (прийняти за $0,5 \text{ м}$).

$$R_\Gamma = \left(275 \frac{\square}{\square} (2 * 3,14 * 18,9) \right) * \ln \left(\frac{(2 * (18,9)^2)}{\square} (0,95 * 0,15 * 0,5) \right) = 21,3$$

9) Визначається коефіцієнт використання горизонтального заземлювача η_c відповідно до необхідної кількості вертикальних заземлювачів n_B .

Так як кількість заземлювачів дорівнює 7, прийmemo за 10, заземлювачі розташовані в ряд, тому коефіцієнт використання горизонтального заземлювача 0,62.

10) Розраховується результуючий опір заземлювального електроду з урахуванням з'єднувальної смуги:

$$R_{\text{заг}} = \frac{R_{\text{в}} \cdot R_{\text{г}}}{R_{\text{в}} \cdot \eta_{\text{с}} + R_{\text{г}} \cdot n_{\text{в}} \cdot \eta_{\text{в}}} \leq R_{\text{д}}. \quad (6.10)$$

$$R_{\text{заг}} = \frac{(25,03 * 18,98)}{\square} (25,03 * 0,62 + 18,98 * 0,65 * 8) \leq 10$$

$$4,37 \leq 10$$

6.8 Висновки до розділу 6

В результаті проведеної роботи у розділі «Охорона праці» було зроблено аналіз умов праці, шкідливих та небезпечних чинників. Було визначено параметри і певні характеристики приміщення для роботи над проектом, описано, які заходи потрібно зробити для того, щоб дане приміщення відповідало необхідним нормам і було комфортним і безпечним.

Приведені рекомендації щодо організації робочого місця, а також важливу інформацію щодо пожежної та електробезпеки. Були наведені розміри приміщення та наведено значення температури, вологості й рухливості повітря, необхідна кількість і потужність ламп та інші параметри, значення яких впливає на умови праці, а також – наведені інструкції з охорони праці, техніки безпеки при роботі на комп'ютері.

ВИСНОВКИ

У даній дипломній роботі було створено веб-додаток для ведення блогу, в якому читачі можуть поділитися враженнями від прочитаної книги. Під час роботи над веб-додатком були використані такі програмні продукти, як:

- Open Server для розробки, налагодження і тестування веб-проектів, а так само для надання веб-сервісів в локальних мережах;
- Atom – це текстовий редактор, який дуже зручно використовувати для написання коду;
- Balsamiq mockups використовувався для створення прототипів деяких сторінок веб-додатку.

Написання коду здійснювалося на мові програмування PHP, найбільш популярній мові для створення веб-проектів. Плюси мови PHP складається в тому, що вона є традиційною, простою в використанні, гнучкою та безпечною.

Створення сторінок блогу здійснювалося за допомогою мови розмітки документів HTML та каскадних таблиць стилів.

Створила веб-додаток, який в подальшому легко можна модернізувати та покращити, додати потрібні функції, а бо видалити наявні.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Кузнецов М. В. PHP на примерах. – 2-е изд., перераб. и доп. – СПб.: БХВ-Петербург, 2012. – 400 с.
2. Бабаев А. Б., Евдокимов Н. В., Боде М. М. Создание сайтов. – СПб.: Питер, 2014. – 304 с.
3. Steve Prettyman Learn PHP 7: Object-Oriented Modular Programming using HTML5, CSS, JavaScript, XML, JSON, and MySQL. – Stone Mountain, Georgia USA, 2016.
4. Дунаев В. В. Базы данных. Язык SQL для студента: 2-е изд. доп. и перераб. – СПб.: БХВ-Петербург, 2007. – 320 с.
5. Безопасность жизнедеятельности. Под ред. С.В.Белова, М.:Высшая школа, 1999 г.
6. ГОСТ 12.1.004-91 ССБТ. Пожарная безопасность. Общие требования
7. ГОСТ 12.1.007-76 ССБТ. Вредные вещества. Классификация и общие требования безопасности
8. Закон України «Про охорону праці»
9. Кодекс законів про працю України
- 10.НПАОП 0.00-1.28-10 Правила охорони праці під час експлуатації електронно- обчислювальних машин
- 11.НПАОП 0.00-8.24-05 Перелік робіт з підвищеною небезпекою
- 12.НПАОП 40.1-1.01-97 Правила безопасной эксплуатации электроустановок
- 13.НПАОП 40.1-1.21-98 Правила безпечної експлуатації електроустановок споживачів
- 14.ДСанПіН 3.3.2.007-98 «Правила і норми роботи з візуальними дисплейними терміналами електронно-обчислювальних машин»
- 15.ДСН 3.3.6.042-99 «Санітарні норми мікроклімату виробничих приміщень»

- 16.ДСН 3.3.6.037-99 Санітарні норми виробничого шуму, ультразвуку та інфразвуку
- 17.ГОСТ 12.1.006-84 ССБТ. Электромагнитные поля радиочастот. Общие требования безопасности. Допустимые уровни на рабочих местах и требования к проведению контроля
- 18.ДБН В.2.5-28:2015 Природне і штучне освітлення
- 19.ГОСТ 12.1.030-81 ССБТ. Электробезопасность.Защитное заземление. Зануление
- 20.ГОСТ 13109-97 „Электрическая энергия. Совместимость технических средств электромагнитных. Нормы качества электроэнергоснабжения общего назначения”
- 21.ДБН В.2.5-67:2013 Опалення, вентиляція та кондиціонування
- 22.ГОСТ 12.1.005-88 ССБТ. Общие санитарно-гигиенические требования к воздуху рабочей зоны
- 23.ГОСТ 12.1.044-89 ССБТ. Пожаровзрывоопасность веществ и материалов. Номенклатура показателей и методы их определения

ДОДАТКИ

Додаток А.

Лістинг програми

NewsController.php

```

<?php
include_once $_SERVER["DOCUMENT_ROOT"]."/models/News.php";
include_once $_SERVER["DOCUMENT_ROOT"]."/models/User.php";
class NewsController
{
    public function actionList($page = 1){
        $newsItems = News::getAll($page);
        $categoryItems = Category::getAll();
        //получаю данные для пагинации
        $numberItems = News::SHOW_BY_DEFAULT;
        $countItems = News::getCountNews();
        $numberPages = ceil($countItems/intval($numberItems)); //округляю количество
страниц
        //получаю данные про пользователя
        //отправляю данные на вьюху

        $view = new View();
        View::userControl($view);
        $view->assign("page", $page);
        $view->assign("items", $newsItems);
        $view->assign("categories", $categoryItems);
        $view->assign("activeCategory", 0);
        $view->assign("numberPages", $numberPages);
        $view->display("news/newslis.php");
        return true;
    }

    public function actionCategory($numberCategory, $page = 1){

```

```

$newsItems = News::getAllByCategory($numberCategory, $page);
$categoryItems = Category::getAll();

$numberItems = News::SHOW_BY_DEFAULT;
$countItems = News::getCountNews($numberCategory);
$numberPages = ceil($countItems/intval($numberItems));

$view = new View();
View::userControl($view);
$view->assign("page", $page);
$view->assign("items", $newsItems);
$view->assign("categories", $categoryItems);
$view->assign("activeCategory", $numberCategory);
$view->assign("numberPages", $numberPages);
$view->display("news/newslist.php");

return true;
}

public function actionView($numberNews){
    $newsItem = News::getOneById($numberNews);
    $view = new View();
    $view->assign("news", $newsItem);
    View::userControl($view);
    $view->display("news/newsone.php");
    return true;
}

public function actionAdd(){
    $article = "";
    $title = "";
    $categoryNumber = 1;
    $errors = false;

    //проверка на аутентификацию пользователя
    $userId = User::checkUserLogged();
    $isAdmin = User::isAdmin($userId);

```

```

//if($isAdmin == false) {
    // header("Location: /");
// }
$categoryItems = Category::getAll();
if(isset($_POST["action"])){

    $article = $_POST["article"];
    if(!isset($article) || News::articleValid($article) == false){
        $errors[] = "Ошибка ввода записи";
    }
    $title = $_POST["title"];

    if(!isset($title) || News::titleValid($title) == false){
        $errors[] = "Ошибка ввода названия";
    }

    $categoryNumber = $_POST["category"];

    //валидация данных
    //отправляем данные в базу данных

    if($errors==false){
        $result = News::addNews($title, $article, $categoryNumber, $userId);
        if($result != 0){

            if(is_uploaded_file($_FILES["image"]["tmp_name"])) {
                $namePathToLoad = $_SERVER['DOCUMENT_ROOT']
                .
                "/static/img/{ $result }.jpg";
                if(move_uploaded_file($_FILES["image"]["tmp_name"], $namePathToLoad)){
                    header("Location: /");
                }else{
                    $errors[] = "Ошибка записи файла!";
                }
            }else{

```



```

        header("Location: /");
    }

    }else{
        $errors[] = "Ошибка записи в Базу Данных";
    }
}
}
$view = new View();
View::userControl($view);
$view->assign("errors", $errors);
$view->assign("categories", $categoryItems);
$view->display("news/newsadd.php");
    //считываем данные из формы
    return true;
}

public function actionAlljson(){
    header("HTTP/1.1 200 OK");
    header("Access-Control-Allow-Origin: *");
    header("Access-Control-Allow-Methods: GET, POST");
    header("Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
Accept");
    $newsItems = News::getAll(1);
    echo json_encode($newsItems);
    return true;
}

public function actionOnejson($numberNews){
    header("HTTP/1.1 200 OK");
    header("Access-Control-Allow-Origin: *");
    header("Access-Control-Allow-Methods: GET, POST");
    header("Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
Accept");
    $newsItem = News::getOneById($numberNews);
    echo json_encode($newsItem);
    return true;
}

```

```

}
public function actionRss(){
    $newsItems = News::getAll(1);
    $categoryItems = Category::getAll();
    $view = new View();
    $view->assign("items", $newsItems);
    $view->assign("categories", $categoryItems);
    $view->display("news/rss.php");
    return true;
}
public function actionRsscategory($numberCategory){
    $newsItems = News::getAllByCategory($numberCategory, 1);
    $categoryItems = Category::getAll();
    $view = new View();
    $view->assign("items", $newsItems);
    $view->assign("categories", $categoryItems);
    $view->display("news/rss.php");
    return true;
}
}
}

```

PageController.php

```

<?php
include_once $_SERVER["DOCUMENT_ROOT"]."/models/Page.php";
include_once $_SERVER["DOCUMENT_ROOT"]."/models/User.php";
class PageController
{
    public function actionOne($id=1){
        $page = Page::getOneById($id);
        $view = new View();
        View::userControl($view);
        $view->assign("page", $page);
        $view->display("static_page.php");
    }
}

```

```
        return true;
    }
}
```

UserController.php

```
<?php
include_once $_SERVER["DOCUMENT_ROOT"]."/models/User.php";
class UserController
{
    public function actionRegister(){
        $view = new View();
        $firstName = "";
        $lastName = "";
        $login = "";
        $password = "";
        $email = "";

        if(isset($_POST["action"])){
            $firstName = $_POST["first_name"];
            $lastName = $_POST["last_name"];
            $login = $_POST["login"];
            $password = $_POST["password"];
            $email = $_POST["email"];
            $errors = false;
            if(!User::checkName($firstName)){
                $errors[] = "Имя пользователя слишком короткое";
            }
            if(!User::checkName($lastName)){
                $errors[] = "Фамилия пользователя слишком короткая";
            }
            if(!User::checkName($login)){
                $errors[] = "Логин слишком короткий";
            }
        }
    }
}
```

```

    }
    if(!User::checkEmail($email)){
        $errors[] = "E-mail введен не корректно";
    }
    if(!User::checkPassword($password)){
        $errors[] = "Пароль слишком короткий";
    }
    if(!User::checkLoginExist($login)){
        $errors[] = "Такой логин уже существует";
    }
    if(!User::checkEmailExist($login)){
        $errors[] = "Такой e-mail уже существует";
    }
    $view->assign("errors", $errors);
    if($errors == false){
        $result = User::register($firstName, $lastName, $login, $password, $email);
        if($result){
            $view2 = new View();
            $view2->assign("register", true);
            $view2->display("auth/auth.php");
            return true;
        }
    }
}
$view->assign("password", $password);
$view->assign("login", $login);
$view->assign("last_name", $lastName);
$view->assign("first_name", $firstName);
$view->assign("email", $email);
$view->display("auth/register.php");
return true;
}
public function actionAuth(){
    $login = "";
    $password = "";

```

```
$errors = false;
if(isset($_POST["action"])){
    $login = $_POST["login"];
    $password = $_POST["password"];

    if(!User::checkPassword($password)){
        $errors[] = "Пароль слишком короткий";
    }
    if(!User::checkName($login)){
        $errors[] = "ЛОГИН слишком короткий";
    }
    $userId = User::checkUserExist($login, $password);
    if($userId==false){
        $errors[] = "Неверный логин или пароль";
    }else {
        User::auth($userId);
        header("Location: /");
    }
}
$view = new View();
$view->assign("login", $login);
$view->assign("password", $password);
$view->assign("errors", $errors);
$view->display("auth/auth.php");
return true;
}
public function actionLogout(){
    User::logout();
}
}
```

Category.php

```
<?php
class Category
{
    public static function getAll(){
        $db = Db::getConnection();
        $result = $db->query('SELECT * FROM category_news');
        $i = 0;
        if($result){
            while($row=$result->fetch()){
                $categoryItems[$i]["id"] = $row["id"];
                $categoryItems[$i]["name"] = $row["name"];
                $i++;
            }
            return $categoryItems;
        }
    }

    public static function getById($id){
        $db = Db::getConnection();
        $result = $db->query('SELECT name FROM category_news WHERE id='.$id);
        $result = $result->fetch(PDO::FETCH_ASSOC);
        $name = $result["name"];
        return $name;
    }
}
```

News.php

```
<?php
include $_SERVER["DOCUMENT_ROOT"]."/models/Category.php";
class News
{
    const SHOW_BY_DEFAULT = 6;
```

```

public static function getOneById($id){
    $id = intval($id);
    if($id){
        $db = Db::getConnection();
        $result = $db->query("SELECT * FROM news WHERE id=".$id);
        $result->setFetchMode(PDO::FETCH_ASSOC);
        $newsItem = $result->fetch();
        $newsItem["category"] = Category::getById($newsItem["news_category_id"]);
        return $newsItem;
    }
}

public static function getAllByCategory($numberCategory, $page = 1)
{
    $page = intval($page);
    $numberCategory = intval($numberCategory);

    if ($numberCategory) {
        $db = Db::getConnection();
        $offset = ($page-1)*self::SHOW_BY_DEFAULT;
        $result = $db->query("SELECT * FROM news
            WHERE news_category_id=" . $numberCategory.' ORDER BY date DESC LIMIT
            '.self::SHOW_BY_DEFAULT.' OFFSET '.$offset);
        $i = 0;
        while ($row = $result->fetch()) {
            $newsItems[$i]["id"] = $row["id"];
            $newsItems[$i]["title"] = $row["title"];
            $newsItems[$i]["article"] = $row["article"];
            $newsItems[$i]["date"] = $row["date"];
            $newsItems[$i]["category"] = Category::getById($numberCategory);
            $newsItems[$i]["status"] = $row["status"];
            $i++;
        }
        return $newsItems;
    }
}

public static function getCountNews($idcategory = 0)

```

```

{
    $db = Db::getConnection();
    if ($idcategory == 0) {
        $result = $db->query("SELECT COUNT(*) FROM news");
    }else{
        $result = $db->query("SELECT COUNT(*) FROM news WHERE
news_category_id=".$idcategory);
        var_dump($result);
    }
    $count = $result->fetch(PDO::FETCH_NUM);
    return intval($count[0]);
}

public static function getAll($page = 1){
    $page = intval($page);
    $offset = ($page-1)*self::SHOW_BY_DEFAULT;
    $db = Db::getConnection();
    $result = $db->query("SELECT id, title, article, date, news_category_id, status
FROM news ORDER BY date DESC LIMIT '.self::SHOW_BY_DEFAULT.' OFFSET
'.$offset);
    $i = 0;
    while($row=$result->fetch()){
        $newsItems[$i]["id"] = $row["id"];
        $newsItems[$i]["title"] = $row["title"];
        $newsItems[$i]["article"] = $row["article"];
        $newsItems[$i]["date"] = $row["date"];
        $newsItems[$i]["category"] = Category::getById($row["news_category_id"]);
        $newsItems[$i]["status"] = $row["status"];
        $i++;
    }
    return $newsItems;
}

public static function addNews($title, $article, $category, $idAuthor){
    $db = Db::getConnection();
    $sql = "INSERT INTO news(title, article, news_category_id, user_id)
VALUES (:title, :article, :category, :user_id)";

```



```

$result=$db->prepare($sql);
$result->bindValue(":title", $title);
$result->bindValue(":article", $article);
$result->bindValue(":category", $category);
$result->bindValue(":user_id", $idAuthor);
if($result->execute()){
    return $db->lastInsertId();
}
return 0;
}
public static function articleValid($text){
    if($text != "" and $text != false){
        return true;
    }
    return false;
}
public static function titleValid($text){
    if($text != "" and $text != false){
        return true;
    }
    return false;
}
}
}

```

User.php

```

<?php
class User
{
    public static function register($firstName, $lastName, $login, $password, $email){
        $db = Db::getConnection();
        $sql = 'INSERT INTO user(first_name, last_name, login, password, email)
            VALUES (:first_name, :last_name, :login, :password, :email)';
        $result = $db->prepare($sql);
        $result->bindParam(':first_name', $firstName, PDO::PARAM_STR);
    }
}

```

```

$result->bindParam(':last_name', $lastName, PDO::PARAM_STR);
$result->bindParam(':login', $login, PDO::PARAM_STR);
$result->bindParam(':password', $password, PDO::PARAM_STR);
$result->bindParam(':email', $email, PDO::PARAM_STR);
return $result->execute();
}
public static function auth($userId){
    $_SESSION["user"] = $userId;
}
public static function logout(){
    unset($_SESSION["user"]);
    header("Location: /");
}
public static function checkName($name){
    if(strlen($name) >= 2){
        return true;
    }
    return false;
}
public static function checkEmail($email){
    if(filter_var($email, FILTER_VALIDATE_EMAIL)){
        return true;
    }
    return false;
}
public static function checkLoginExist($login){
    $db = Db::getConnection();
    $sql = 'SELECT COUNT(*) FROM user WHERE login=:login';
    $result = $db->prepare($sql);
    $result->bindParam(':login', $login, PDO::PARAM_STR);
    $result->execute();
    if($result->fetchColumn())
        return false;
    return true;
}

```

```

public static function checkEmailExist($email){
    $db = Db::getConnection();
    $sql = 'SELECT COUNT(*) FROM user WHERE email=:email';
    $result = $db->prepare($sql);
    $result->bindParam(':email', $email, PDO::PARAM_STR);
    $result->execute();
    if($result->fetchColumn())
        return false;
    return true;
}

public static function checkPassword($password){
    if(strlen($password) >= 6){
        return true;
    }
    return false;
}

public static function checkUserExist($login, $password){
    $db = Db::getConnection();
    $sql = 'SELECT * FROM user WHERE login=:login AND password=:password';
    $result = $db->prepare($sql);
    $result->bindParam(':login', $login, PDO::PARAM_STR);
    $result->bindParam(':password', $password, PDO::PARAM_STR);
    $result->execute();
    $user = $result->fetch();
    if($user)
        return $user["id"];
    return false;
}

public static function checkUserLogged(){

    if(isset($_SESSION["user"])){
        return $_SESSION["user"];
    }
    return false;
}

```

```
public static function isAdmin($idUser){
    $db = Db::getConnection();
    $sql = 'SELECT is_admin FROM user WHERE id=:id';
    $result = $db->prepare($sql);
    $result->bindParam(":id", $idUser, PDO::PARAM_INT);
    $result->execute();
    if($result->fetchColumn()){
        return true;
    }
    return false;
}

public static function isGuest(){
    if (isset($_SESSION["user"])){
        return false;
    }
    return true;
}
}
```

Додаток Б. Презентація

Веб-додаток для ведення тематичного блогу

АВТОР: СТУДЕНТ ГРУПИ КН-13Д ЛЕВАШОВА Є. Н.

КЕРІВНИК: ЩЕРБАКОВА М. Є.

1

Актуальність теми

Блог - це веб-сайт, основний зміст якого - регулярно додавані записи, що містять текст, зображення або мультимедіа.

Для блогів характерні недовгі записи тимчасової значущості, впорядковані в зворотному хронологічному порядку. Відмінності блогу від традиційного щоденника обумовлюються середовищем: блоги зазвичай публічні і передбачають сторонніх читачів.

За версією газети [Вашингтон профайл](#) (англ. Washington Profile), першим блогом вважають сторінку [Тіма Бернерса-Лі](#), де він, починаючи з 1992 року, публікував новини. Широке використання блогів розпочалося з 1996 року.

В наш час, в епоху нових технологій, книги стали набагато доступніше, ніж, скажімо, кілька десятиліть тому. Книги можна купити в магазинах як на паперових носіях, так і в електронному вигляді.

У дипломній роботі планується розробити веб-додаток для ведення блогу, в якому читачі зможуть поділитися враженнями від прочитаної книги.



2

ВИБІР ЗАСОБІВ ДЛЯ ВИКОНАННЯ ДИПЛОМНОГО ПРОЕКТУ

PHP - найбільш популярна мова для створення веб-проектів. Плюси мови PHP складається в тому, що вона є традиційною, простою в використанні, гнучкою та безпечною.

HTML - стандартизована мова розмітки документів у Всесвітній павутині. Більшість веб-сторінок містять опис розмітки на мові HTML (або XHTML). Мова HTML інтерпретується браузерами; отриманий в результаті інтерпретації форматований текст відображається на екрані монітора комп'ютера або мобільного пристрою.

CSS-код - це список інструкцій для браузера, - як і де відобразити елементи веб-сторінки, написаний особливим чином. Під «елементами» зазвичай маю на увазі теги XHTML / HTML і їх вміст.

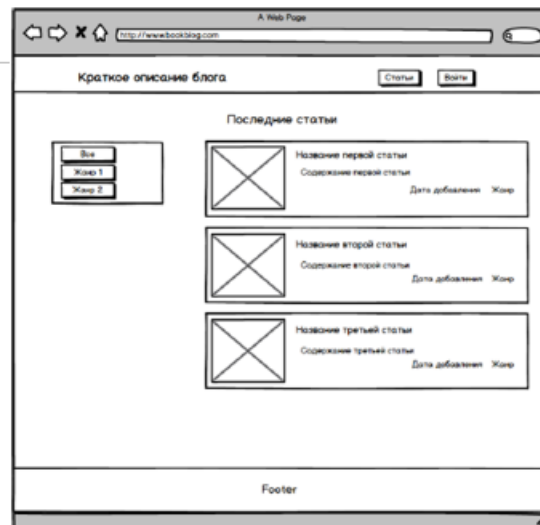


3

Реалізація прототипів

Для створення прототипів використовую Balsamiq Mockups, це проста в використанні та зі зрозумілим інтерфейсом програма.

Прототип головної сторінки веб-додатку для ведення тематичного блогу



4

Реалізація прототипів

Прототип сторінки реєстрації на сайті

5

Структура веб-додатку

Структура бази даних.

Таблиця `category_news` містить дані про жанри, їх назва.

Таблиця `news` містить данні про статті так, як назва, зміст статті, дата публікації та ішн.

У таблиці `page` знаходиться id сторінок.

Таблиця `user` містить дані про зареєстрованих користувачів, ім'я, прізвище, логін, пароль.



6

Структура коду

Програма складається з моделей, контролерів та роутів, які виконують функції.

Наприклад, у моделі Category.php пописані функції, які вибирають усі, або по одній категорії з бази даних.

У моделі News.php виконуються такі функції, як додавання статті до бази даних, виборка та підрахунок статей у базі даних.

У моделі User.php виконуються функції реєстрації користувачів, авторизації та виходу з блогу, функція перевірки чи є користувач адміністратором.

У контролерах проводиться перевірка на заповнення полів у формах вводу, формування пагінації та інш.

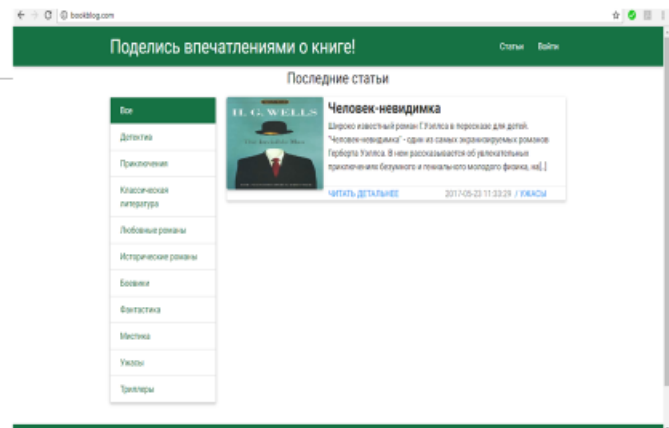
Наприклад, у файлі UserController.php проводиться перевірка на правильне заповнення полів при реєстрації та вході на сайт.



7

Використання блогу

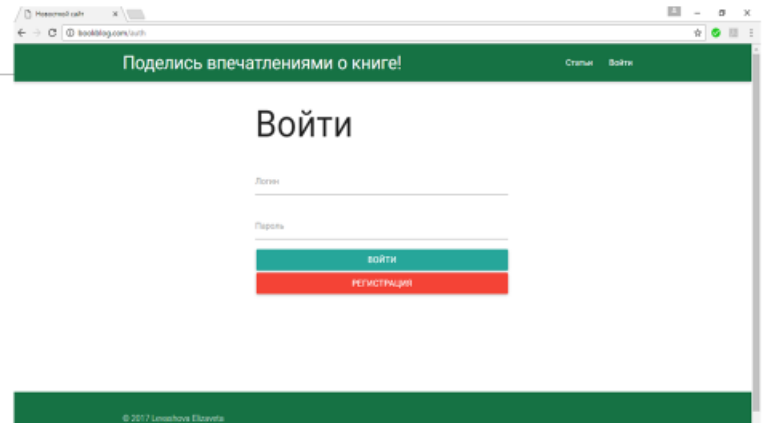
Головна сторінка блогу, як і весь веб-додаток, виконано у світлих тонах, а саме він має білий колір фону та зелені елементи. Зверху розміщується назва сайту ліворуч, кнопки статей та входу праворуч. Ця сторінка ділиться на дві частини: з однієї сторони розташовані можливі жанри, з іншої самі статті.



8

Використання блогу

Сторінка входу на сайт виконана в такому ж дизайні, посередині розташована форма входу з двома поля вводу та кнопки увійти та реєстрація.



Висновки

У даній дипломній роботі було створено веб-додаток для ведення блогу, в якому читачі можуть поділитися враженнями від прочитаної книги. Під час роботи над веб-додатком були використані такі програмні продукти, як:

Open Server для розробки, налагодження і тестування веб-проектів, а так само для надання веб-сервісів в локальних мережах;

Atom – це текстовий редактор, який дуже зручно використовувати для написання коду;

Balsamiq mockups використовувався для створення прототипів деяких сторінок веб-додатку.

Створила веб-додаток, який в подальшому легко можна модернізувати та покращити, додати потрібні функції, а бо видалити наявні.