

МІНІСТЕРСТВО ОСВІТИ ТА НАУКИ УКРАЇНИ
ПрАТ «ПРИВАТНИЙ ВИЩИЙ НАВЧАЛЬНИЙ ЗАКЛАД
«ЗАПОРІЗЬКИЙ ІНСТИТУТ ЕКОНОМІКИ
ТА ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ»

Предметно-циклова комісія інформаційних технологій

ДО ЗАХИСТУ ДОПУЩЕНА

Голова ПЦК _____
спеціаліст в/к Сабанов С.О.

ВИПУСКНА РОБОТА МОЛОДШОГО СПЕЦІАЛІСТА

**РОЗРОБКА ВЕБЗАСТОСУНКУ ПОРТАЛУ НОВИН
НА БАЗІ IDE VISUAL STUDIO**

Виконав
ст. гр. ПЗ-119К9 _____ Віноградов Д.В.

Керівник
викладач _____ Суха К.С.

Запоріжжя
2023

СТРУКТУРНИЙ ПІДРОЗДІЛ «ФАХОВИЙ КОЛЕДЖ ЕКОНОМІКИ ТА
ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ ПРАТ «ПВНЗ «ЗІЕІТ»

Предметно-циклова комісія інформаційних технологій

ЗАТВЕРДЖУЮ

Голова ПЦК спеціаліст в/к

Сабанов С.О. _____

“21” лютого 2023 року

ЗАВДАННЯ

НА ВИПУСКНУ РОБОТУ МОЛОДШОГО СПЕЦІАЛІСТА

студенту гр. ІПЗ-119К9

спеціальності: 121 – Інженерія програмного забезпечення

Віноградову Данилу Володимировичу

(прізвище, ім'я, по батькові)

1. Тема: «Розробка вебзастосунку порталу новин на базі IDE Visual Studio»

затверджена наказом по інституту: № 09.2-14 від 20 лютого 2023 року

2. Термін здачі студентом закінченої роботи: 20 червня 2023 року

3. Перелік питань, що підлягають розробці:

1. Провести огляд літератури та інтернет-джерел, присвячених
тематиці випускної роботи.

2. Провести огляд різних порталів новин та навести основні
характеристики.

3. Розглянути інструментальні засоби створення вебзастосунку з
метою вибрати оптимальні інструменти розробки.

4. Визначити основні функціональні можливості вебзастосунку.

5. Реалізувати алгоритм роботи вебзастосунку.

6. Провести тестувати розробки та доопрацювати недоліки.

7. Створити наповнення для бази даних.

8. Оформити результати роботи у вигляді пояснювальної записки, що
відповідає стандартам підприємства щодо оформлення випускних
робіт.

4. Календарний графік

№ етапу	Зміст	Термін виконання	Готовність (%), підпис керівника	Підпис керівника про повну готовність етапу, дата
1	Збір практичного матеріалу за темою	16.01.23-17.02.23		
2	I атестація. I розділ випускної роботи	27.03.23-01.04.23		
3	II атестація. II розділ випускної роботи	01.05.23-06.05.23		
4	III атестація. III розділ випускної роботи, висновки та рекомендації, додатки, реферат.	29.05.23-03.06.23		
5	Перевірка випускної роботи на оригінальність	15.05.23-12.06.23		
6	Доопрацювання випускної роботи, підготовка презентації, отримання відгуку керівника та рецензії	05.06.23-10.06.23		
7	Попередній захист випускної роботи	12.06.23-18.06.23		
8	Подача випускної роботи на кафедру	за 3 дні до захисту		
9	Захист випускної роботи	19.06.23-24.06.23		

Дата видачі завдання: 22 лютого 2023 р.

Керівник випускної роботи _____ Суха К.С.
(підпис)

Завдання прийняв до виконання _____ Віноградов Д.В.
(підпис студента)

РЕФЕРАТ

Випускна робота молодшого спеціаліста містить 46 сторінок, 18 рисунків, 8 лістингів, 15 бібліографічних посилань.

Об'єкт дослідження – застосування технологій прогресивних вебдодатків для створення сучасних застосунків.

Предмет – застосування технологій прогресивних вебдодатків для створення сучасних застосунків.

Мета роботи – розробка вебзастосунку порталу новин на базі IDE Visual Studio.

Здійснено детальний огляд предметної області та сучасних аналогів.

Проект реалізовано за допомогою таких засобів, як HTML, CSS, PHP, MySQL, JS.

Програмний продукт є легким у використанні, має лаконічний дизайн та зрозумілий для користувача інтерфейс. У створеному вебсайті було реалізовано адаптивну верстку, що дозволяє зручно користуватися ним за допомогою будь-якого девайсу та завжди мати можливість переглянути актуальні новини.

ВЕБЗАСТОСУНОК, ПОРТАЛ НОВИН, HTML, CSS, PHP

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ, СКОРОЧЕНЬ І ТЕРМІНІВ	7
ВСТУП	8
РОЗДІЛ 1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ Й ПОСТАНОВКА ЗАДАЧІ.....	9
1.1 Історія виникнення вебсайтів	9
1.2 Основні види сайтів	10
1.2.1 Сайт блогу.....	10
1.2.2 Бізнес сайт.....	10
1.2.3 Інформаційний сайт	10
1.2.4 Сайт форум	11
1.2.5 Інтернет-магазин	11
1.2.6 Персональний сайт.....	12
1.3 Способи створення вебсайту	12
1.3.1 Конструктор.....	12
1.3.2 CMS	13
1.3.3 Самостійна розробка.....	15
1.4 Аналоги вебзастосунку порталу новин	16
1.5 Висновки за розділом	18
РОЗДІЛ 2 ВИБІР ТЕХНОЛОГІЙ ПРОЕКТУВАННЯ.....	19
2.1 Мова розмітки HTML	19
2.2 Огляд технології Cascade Style Sheets (CSS).....	20
2.3 JavaScript	21
2.4 Мова програмування PHP	23

2.5 MySQL.....	25
2.6 Visual Studio Code	26
2.7 Висновки за розділом	27
РОЗДІЛ 3 ПРАКТИЧНА РЕАЛІЗАЦІЯ СИСТЕМИ.....	28
3.1 Архітектура програми	28
3.2 Властивості коду	29
3.3 Користувацькі властивості.....	38
3.4 Висновки за розділом	43
ВИСНОВКИ	44
ПЕРЕЛІК ПОСИЛАНЬ.....	45

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ,
СКОРОЧЕНЬ І ТЕРМІНІВ

Скорочення	Повна назва	Пояснення/переклад
БД	База даних	
ООП	Об'єктно-орієнтоване програмування	
СУБД	Система управління базами даних	
ЗМІ	Засоби масової інформації	
CERN	Conseil Européen pour la Recherche Nucléaire	Європейська організація з ядерних досліджень
CMS	Content Management System	Каскадні таблиці стилів
HTML	HyperText Markup Language	Мова розмітки гіпертексту
NeXT	Near End Crosstalk	Перехідне згасання, виміряне на ближньому кінці сусідньої пари
PHP	Hypertext Preprocessor	Препроцесор гіпертексту
SEO	Search Engine Optimization	Система керування вмістом сайту
WWW	World Wide Web	Всесвітня мережа

ВСТУП

Метою даної випускної роботи є збір теоретичних відомостей про сучасні інструментальні засоби для побудови сайту, огляд та аналіз популярних допоміжних бібліотек та розробка сайту за їх допомогою, визначення оптимальних сфер використання інструментальних засобів, спираючись на аналіз процесу розробки сайту.

Сайти новин у наші дні дуже популярні серед користувачів. Розвиток таких ресурсів просувається стрімкими темпами. Вони повністю витіснили традиційні ЗМІ – газети, журнали, телебачення. Ці послуги зазвичай мають чітку географічну спрямованість – кожне місто чи регіон створюють свій сайт.

Для повноцінного додатку потрібно розробити серверну та клієнтську частину, обрати базу даних для зберігання інформації, налаштувати взаємодію та потік даних між клієнтською та серверною частинами. Для цього потрібно обрати мову програмування, середовище розробки, допоміжні бібліотеки, тощо.

У ході розробки будуть вивчені різноманітні бібліотеки та реалізовані наступні кроки:

- обдумати та створити архітектуру додатку;
- створити моделі або схеми об'єктів додатку;
- реалізувати збереження інформації у базі даних;
- оглянути додаток та зробити висновки.

РОЗДІЛ 1

АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ Й ПОСТАНОВКА ЗАДАЧІ

1.1 Історія виникнення вебсайтів

У 1989 році британський учений Тім Бернерс-Лі винайшов Всесвітню павутину (WWW), працюючи в CERN. Вона була розроблена з метою задовольнити потреби в автоматизованому обміні інформацією між вченими в університетах і дослідницьких установах по всьому світу. CERN — це не ізольована лабораторія, а центр великої спільноти, що налічує понад 17 000 вчених із понад 100 країн. Науковці зазвичай працюють в університетах і національних лабораторіях своєї країни. Основна ідея Всесвітньої павутини полягає в тому, щоб об'єднати новітні комп'ютерні, мережеві технології передачі даних і гіпертекстові технології в одну потужну та просту у використанні глобальну інформаційну систему. Першу пропозицію щодо Всесвітньої павутини Тім Бернерс-Лі написав у березні 1989 року, а другу — у травні 1990 року.

Наприкінці 1990 року Тім Бернерс-Лі представив свою ідею в CERN, що дозволило підключити вебсервер і браузер. Він розробив код вебсервера на комп'ютерах NeXT. «Ця машина є сервером» написали на комп'ютері, щоб люди випадково не вимкнули його[1].

WWW створено, щоб полегшити доступ до існуючої інформації та перших вебсайтів, пов'язаних з інформацією, корисною для вчених CERN, таких як телефонний довідник CERN та посібник користувача сервера CERN. Інструменти пошуку поклалися на ключові слова — спочатку пошукових систем не було. Оригінальний браузер Бернерса-Лі, який працював на комп'ютері NeXT, продемонстрував його бачення та мав багато функцій сучасного браузера. Він також включав можливість змінювати сторінки безпосередньо в браузері, що ознаменувало першу функцію редагування в Інтернеті[1].

1.2 Основні види сайтів

1.2.1 Сайт блогу

Блоги надають платформу для обміну письмовим і візуальним вмістом про інтереси користувача. Деякі користувачі, які активно ведуть свої блоги, згодом відкривають для себе прибуткові можливості, як-от афілійований маркетинг і продаж рекламного простору. Приклад із реального блогу: Есті Кесслер перетворила свою пристрасть до написання стилю життя на добре оплачуваний бізнес за допомогою свого сімейного блогу «Abode by Estie». У своїх публікаціях про дизайн інтер'єру, подорожі та стиль вона ділиться партнерськими посиланнями на свої улюблені магазини товарів для дому та продає пакети читачам, які хочуть працювати з її дизайнерською фірмою[2].

1.2.2 Бізнес сайт

Створення бізнес-сайтів зосереджується на потребах клієнтів та пояснень, яким чином компанія вирішуватиме їхні унікальні побажання. Приклад реального бізнес-сайту: На вебсайті «Puffin Packaging» пояснюється, що її флісова упаковка є доступним і надійним рішенням для коробок з полістиролу. На сайті використовуються чіткі лінії, барвисті зображення та багато білого простору, щоб зацікавити читачів на комп'ютерах і мобільних пристроях. «Animal Music Studios» надає послуги з композиції музики, звукового оформлення та зведення. Сайт демонструє попередні проекти команди для брендів, зокрема «Comcast», «Infinity» та «Poreye's»[2].

1.2.3 Інформаційний сайт

Інформаційні вебсайти є цінним ресурсом для тих, хто хоче дізнатися

більше про певну тему, послугу чи продукт. Орієнтовані на надання поглибленої інформації, ці сайти пропонують багатий вміст, який допомагає відповісти на найактуальніші запитання читачів.

Інформаційний вебсайт також може допомогти організаціям стати експертом у сфері інтересів[2]. Приклад справжнього сайту новин: енциклопедія Wix надає користувачам вичерпний глосарій термінів, які використовуються в бізнесі, маркетингу та вебдизайні.

1.2.4 Сайт форум

Онлайн-форуми об'єднують людей для обговорення та обміну інформацією на спільні теми, як-от технології чи фітнес. Форуми часто включають різноманітні пости, упорядковані за темами, що полегшує користувачам пошук тематик, які вони хочуть обговорити.

Приклад справжнього онлайн-форуму: «BE RADICAL» прагне надати своїм членам можливість бути найкращою версією себе. Цей вебсайт містить радикальну підтримку, серію онлайн-груп, обговорень і завдань, де учасники можуть ділитися своїми знаннями[2].

1.2.5 Інтернет-магазин

Багато компаній вже давно використовують інтернет-магазини для переміщення продажів товарів в онлайн-простір і отримання доходу через Інтернет. Інтернет-магазини відрізняються від інших сайтів в першу чергу тим, що тут представлений не тільки каталог товарів, а й можливість придбати їх прямо на сайті. Популярність такого типу сайтів пояснюється тим, що відкривати свої інтернет-магазини можуть як великі компанії, так і представники малого бізнесу.

Навіть якщо не планувати нічого продавати на платформі інтернет-магазину, вебсайт все одно може допомогти створити онлайн-присутність і

створити впізнаваність бренду компанії[2].

1.2.6 Персональний сайт

У той час як вебсайт портфоліо чудово підходить для демонстрації професійних якостей, особистий вебсайт – це місце, де користувач може бути собою. Створення облікового запису з метою ділитися досягненнями, досліджувати власні інтереси або навіть просування особистого бренду. Те, що користувач може розмістити на особистому вебсайті, залежить лише від конкретних цілей, що він ставить перед собою. У випадку, якщо персональний сайт використовується для просування професійної кар'єри, доречно буде розмістити резюме, кліп або біографію. Якщо ж первинною метою створення сайту було самовираження, то формат блогу з посиланнями на інші соціальні мережі є найкращим варіантом. Конкретний приклад особистого сайту: художниця Лаура Баросс з Нью-Йорку використовує свій особистий вебсайт, щоб поділитися своїм ентузіазмом щодо екологічного способу життя та дизайну без відходів. На вебсайті є можливість подивитись блог про автопортрети та добірку її минулих проектів і фотогалерею[2].

1.3 Способи створення вебсайту

1.3.1 Конструктор

Конструктор – це програмне рішення (найчастіше онлайн), яке дозволяє побудувати вебсайт за принципом модульності, оскільки розробник збирає всю конструкцію за допомогою готових «кубиків», наданих конструктором. Цей підхід дозволяє створити вебсайт без будь-яких знань веброзробки та відповідних навичок. Він ідеально підходить для тих, хто хоче створити простий вебсайт про себе чи свій бізнес, не скаржачись на якість програмного коду та швидкість вебсайту.

Крім того, дизайнери часто створюють прості посадкові сторінки для

невеликих рекламних кампаній. Використовуючи генератор, неможливо приймати серйозні бізнес-рішення або робити покупки в Інтернеті[3].

Переваги:

- Низька ціна. Майже всі оригінальні генератори є безкоштовними, а вартість підключених модулів дуже низька;
- Простий у використанні. Щоб створити свій вебсайт, просто потрібно знати, як користуватися комп'ютерним браузером і мати гарний смак;
- Весь процес виконує дизайнер. Великі сценарії програмного забезпечення, підключення модулів, зберігання та інші приховані процеси запускаються через зрозумілу панель керування лише кількома натисканнями кнопок.

Недоліки:

- Приховані витрати. Дешевий первинний ринок часто приховує іноді значні додаткові витрати: хостинг, домени другого рівня, електронну пошту з доменним іменем і багато іншого.

1.3.2 CMS

В даному випадку буде розглянуто найпопулярнішу CMS — WordPress. Має багато аналогів, які не настільки популярні, але мають схожий набір інструментів і можливостей: Joomla, Drupal і т.д. CMS не буде враховуватися, тому що лише перша покупка ліцензії на таку CMS може коштувати так само дорого, як розробка невеликого сайту в вебстудії [3]. CMS — це набір програмних засобів для керування вебконтентом. Простіше кажучи, це базова структура та набір інструментів і доповнень, які не тільки створюють вебсайти чи вебдодатки, але й підтримують і оновлюють вміст і взаємодіють з користувачами. Усі CMS мають панель керування з відносно зручним інтерфейсом. Основною мовою програмування виступає PHP. Будь-яка CMS дозволяє створювати навіть дуже складні

рішення, такі як інтернет-магазин або великий корпоративний сайт з глибоко вкладеними сторінками, але вони мають свої особливості[3].

Переваги:

- Вільний доступ. Багато CMS спочатку безкоштовні, крім того, в інтернеті є багато готових шаблонів сайтів за будь-якою тематикою (особливо для WordPress);
- Зручне керування контентом. Користуватися інформаційною панеллю для керування сайтом легко та зрозуміло. При використанні CMS не потрібні спеціальні навички для управління контентом;
- Багато готових рішень. Існує безліч модулів, плагінів, надбудов для різних завдань (від слайдерів зображень до SEO оптимізації та систем підтримки користувачів).

Недоліки:

- Вразливість сайту. Найсуттєвіше обмеження будь-якої популярної CMS. Навіть сайти, створені на генераторах, іноді краще захищені від піратства, ніж сайти, керовані WordPress та іншими CMS. Навіть платний варіант не визнаний самим надійним і захищеним від атак і вторгнень;
- Потрібні знання. Створення вебсайтів на CMS вимагало від клієнтів базових знань з верстки та програмування (переважно PHP), що ускладнювало створення вебсайтів. Однак слід зазначити, що знання можуть знадобитися у тому випадку, якщо клієнт хоче додати нову функціональність до існуючого шаблону або створити власний проект з нуля;
- Труднощі з перенесенням. Більшість популярних CMS мають інструменти автоматичного встановлення на хостинг, але якщо виникне необхідність перенесення сайту або керування його розташуванням, користувач може зіткнутися з труднощами, оскільки доведеться проходити всю процедуру встановлення з самого початку;
- Витрати на додатковий контент. Як і у випадку з конструкторами, додаткові модулі та розширення для CMS коштують грошей,

але в данному випадку навіть більше, ніж у при виборі конструктору. Більшість модулів, які, ймовірно, будуть використовуватися постійно, вимагають скромних початкових інвестицій або платної щомісячної підписки;

– Великий сайт — це великі затрати на підтримання його у робочому стані. Ця розробка є більш дорожчою та важкою. Помилка полягає в тому, що за допомогою CMS можна створити сайт дешевше, ніж замовляти розробку в ІТ-компанії, але найчастіше при подальшому використанні даної системи вартість все одно буде збільшуватися.

1.3.3 Самостійна розробка

Найбільш творчий і звільняючий процес, але і найдовший, порівнянний з пошиттям вбрання по фігурі. Написання вебсайту або вебдодатку потребує глибоких знань не лише самих мов програмування, а й розуміння клієнтської архітектури, бізнес-процесів тощо. Також створити вебсайт з нуля, клієнти отримають унікальний і персоналізований продукт, який вирішить їх завдання і не витрачаючи вільний час на непотрібні процеси. Самостійна розробка дозволяє створювати проекти будь-якої складності та відповідно до побажань замовника.

Переваги:

– Свобода вибору. можна розмістити все що потрібно для вирішення бізнес-завдань. При цьому всі функції будуть написані під потреби замовника, без адаптації з шаблону;

– Широкі можливості просування. На відміну від CMS і конструкторів, розкрутити сайт самостійної розробки в звичайному пошуку набагато легше;

– Індивідуальний дизайн. Тільки чистий код дозволяє створити продукт, який повною мірою задовольнятиме бажанням клієнта. Окремо можна відзначити, що ефективний підхід до UI/UX можна використовувати

тільки з чистим кодом, оскільки при використанні CMS клієнти змушені переходити на готові рішення, які можуть не задовольнити всі їхні потреби.

Недоліки:

– Ціна. Суперечливий, але менш чистий код. Хоча у випадку великих проектів розробка без конструктора і без CMS є найефективнішим, а часом і найдешевшим рішенням, тому що кожне подальше поліпшення та доопрацювання не вимагатимуть витрат у великій кількості[3].

1.4 Аналоги вебзастосунку порталу новин

Для розробки сайту у першу чергу потрібно проаналізувати аналоги в області порталу новин, щоб розуміти які методології, дизайнерські рішення та технології для цього використовують.

1)Новинний сайт «ВотТак»

Це проект, вигаданий та реалізований журналістами з Росії, Білорусі, України та інших країн регіону, які хочуть робити якісні ЗМІ, інформувати читача, пояснювати складні процеси простою мовою. Нижче приведено характеристики сайту:

- сайт новин, що реалізовано мовою JS;
- на сайті присутня функція пошуку новин, що реалізовано методом сортування;
- сайт має простий та зрозумілий для користувача інтерфейс;
- присутня динамічна стрічка новин.

На рисунку 1.1 представлено інтерфейс даного сайту новин.

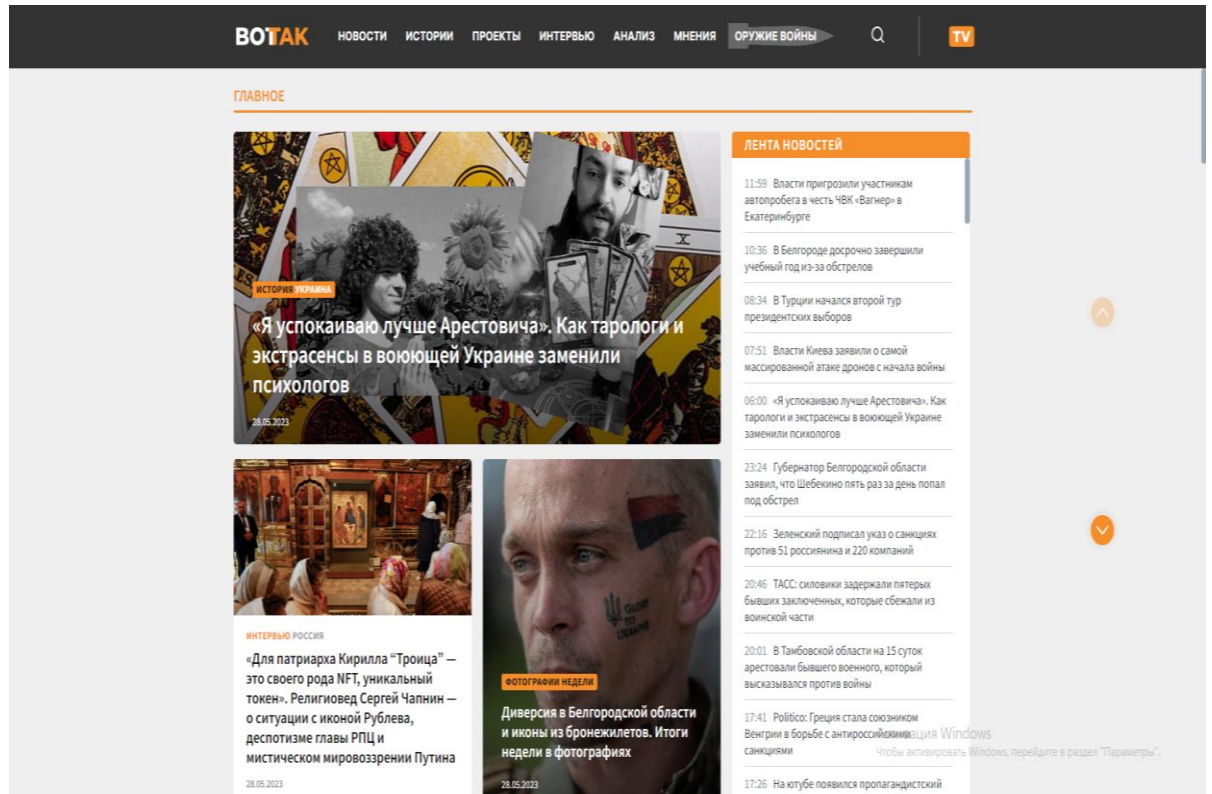


Рис 1.1 — Вебсторінка новин ВотТак

Недоліком цього сайту є відсутність на кнопках анімацій, які б покращили візуальний ефект.

2) Вебсторінка Hromadske

Сайт був створений у 2013 році незалежними журналістами, щоб розповідати суспільству важливу інформацію за допомогою різних каналів зв'язку. За допомогою сайту журналісти пояснюють важливу інформацію та створюють дискусії на різних майданчиках. Нижче приведено характеристики сайту:

- сайт новин, що реалізовано мовою JS;
- на сайті присутня функція пошуку новин;
- сайт має простий та зрозумілий для користувача інтерфейс;
- присутня автоматична стрічка новин;
- в наявності безліч різних анімацій.

На рисунку 1.2 представлено інтерфейс даного сайту новин.

The screenshot shows the Hromadske website interface. At the top, there is a navigation bar with the logo and menu items: «Новини», «Тексти», «Думки», «Відео», «Спецпроекти», «Можливості». There are also search and language selection icons. Below the navigation bar, there are two main news banners. The left banner features a photo of an elderly man and the headline: «У мамі стався мікроінсульт, бо їй не дозволили подивитися на рідний дім — історія вигнання та повернення кримських татар». The right banner has the headline: «росіяни запустили рекордну кількість «Шахедів» по Україні. Що відомо про нічну атаку?». Below these banners are several smaller article thumbnails with titles like «Невже їм заліза шкода? Хлопці життя віддають, а вони не можуть склепати. У чому проблема з відзнаками для військових» and «Рука не працює, а з армії не відпускають». On the right side, there is a sidebar titled «Останні новини» containing a list of recent news items. At the bottom right, there is a small advertisement for «Активация».

Рис 1.2 — Вебсторінка Hromadske

Недоліком цього сайту виявляється те, що при переході на новини не працюють посилання такі як «Текст», «Думки», «Новини» з меню.

1.5 Висновки за розділом

Була розглянута історія інтернет-сайтів, яка дала змогу зрозуміти хто і для яких цілей створив мережу, якою нині користується увесь світ. Вже тоді інтернет-сайти набули популярності завдяки чому нині ці технології безперервно удосконалюються для простоти використання та розуміння.

Також було проаналізовано найпопулярніші тематики сайтів та основні технології їх створення. Для більш глибокого аналізу було розглянуто декілька новинних сайтів, аналогів майбутньої розробки. Із наведених способів створення сайтів було обрано використання мов програмування, що дозволить у простий спосіб створити вебсайт з інформаційних сторінок.

РОЗДІЛ 2

ВИБІР ТЕХНОЛОГІЙ ПРОЕКТУВАННЯ

2.1 Мова розмітки HTML

HTML (мова розмітки гіпертексту) — це код, який використовується для структурування та відображення вебсторінки та її вмісту. Наприклад, вміст можна структурувати в абзаци, марковані списки або використовувати зображення та таблиці даних.

HTML не є мовою програмування, це мова розмітки, яка використовується для вказівки обраному браузеру, яким чином відобразити вебсторінки, що відвідує користувач. Цей процес може бути складним або простим, залежно від того, як цього хоче вебдизайнер. HTML складається з ряду елементів, які користувач використовує для вкладення або обгортання різних елементів вмісту, щоб він виглядав або поведився певним чином. Теги захисту можуть зробити посилання слова на зображення чи на інше слово, виділяти слова курсивом, збільшувати або зменшувати шрифт тощо[4].

У 1989 році Тім Бернерс-Лі, програміст Міжнародного центру високих енергій у Женеві, запропонував проект Всесвітньої павутини. Ідея полягала у створенні взаємопов'язаної системи гіпертекстових документів, що зберігаються на комп'ютерах по всьому світу. Для цього Тім і його помічники розробили технології, без яких сучасний Інтернет був би немислимий: HTTP — протокол передачі документів від клієнта до сервера, URL — унікальна адреса, що визначає місце знаходження ресурсу в Інтернеті, а HTML — гіпертекстова адреса для створення вебдокументів. «Батько інтернету» взяв за основу свою відому SGML. Візитівка HTML походить від неї: є документ «ТегиHTML», опублікований Бернерсом-Лі в 1991 році. HTML почав підтримуватися гіпертекстом. Із зростанням популярності Інтернету стає зрозуміло, що мови розмітки гіпертексту потребують розвитку. Але інновації повинні відповідати всьому розумному, потрібно було щоб з

вебсайтом була легка взаємодія, наприклад відкриття в будь-якому браузері. Консорціум World Wide Web, або W3C, заснований Тімом Бернерсом-Лі, має створювати такі стандарти. Ця всесвітня спільнота продовжує працювати над розробкою вебстандартів. Сьогодні актуальна їх HTML-версія з порядковим номером 5.3[14].

2.2 Огляд технології Cascade Style Sheets (CSS)

CSS (каскадні таблиці стилів) — це код, який користувач використовує для оформлення вебсторінки[5].

CSS є мовою програмування для побудови вебсайту. Це не мова розмітки – це мова стилів які покращують візуальну частину. Це означає, що він дозволяє розробникам вибірково застосовувати стилі до елементів у документі HTML.

Таблиці стилів дозволяють спростити процес створення сторінок і покращити їх зовнішній вигляд. Концепція стилів схожа на ідею стилів, реалізовану в сучасних текстових редакторах — текст спочатку вводиться, а потім форматується за допомогою стилів. Застосування стилів дозволяє виводити необхідний текст та інші елементи на сторінці, не турбуючись про їх зовнішній вигляд і розміщення. Програмісти часто створюють таблиці стилів окремо від файлу html. Під час створення html-файлу він фокусується на вмісті сторінки, а не на її зовнішньому вигляді, а при створенні таблиці стилів все навпаки. Таким чином, стилі допомагають розрізняти етапи створення файлу html і покращують зовнішній вигляд сторінки. Припустимо, що таблицю стилів уже створено, тепер необхідно змусити таблицю взаємодіяти з конкретним файлом html, для цього існують наступні методи взаємодії: зв'язування, імпортування, вбудовування[15].

2.3 JavaScript

JavaScript — це мова програмування, яку розробники використовують для створення інтерактивних вебсторінок. Функції JavaScript можуть покращити досвід роботи користувача з вебсайтом, від оновлення стрічок новин соціальних мереж до відображення інтерактивних карт та анімації. JavaScript — це мова сценаріїв на стороні клієнта, що робить його однією з основних технологій Всесвітньої павутини. Наприклад, карусель зображень, яка з'являється після клацання в меню, і кольори елементів вебсторінки, які користувач бачитиме, що динамічно змінюються під час перегляду вебсторінок, створюються за допомогою JavaScript. [8].

Історично вебсторінки були статичними, подібно до книжкових сторінок. Статична сторінка здебільшого відображає інформацію у фіксованій формі та не виконує все, що користувач очікує від сучасного вебсайту. JavaScript спочатку був технологією на стороні браузера, щоб зробити вебдодатки більш динамічними. Використовуючи його, браузери можуть реагувати на взаємодії користувача та змінювати розташування вмісту на вебсторінці. У міру розвитку мови розробники JavaScript створили бібліотеки, фреймворки та методи програмування та почали використовувати його поза браузерами. Сьогодні JavaScript можна використовувати як для розробки на стороні клієнта, так і на стороні сервера. У наступних підрозділах буде наведено кілька прикладів використання. Усі мови програмування працюють шляхом перекладу англійського синтаксису в машинний код, що виконується операційною системою. JavaScript у широкому сенсі можна класифікувати як мову сценаріїв або інтерпретовану мову. Інтерпретований код JavaScript, тобто код машинної мови, який безпосередньо перекладається механізмом JavaScript. В інших мовах програмування компілятор обробляє весь машинний код на окремому етапі. Отже, усі мови програмування є мовами програмування, але не всі мови програмування є мовами сценаріїв[8].

Двигун JavaScript — це комп'ютерна програма, яка виконує код JavaScript. Ранні двигуни JavaScript були просто інтерпретаторами, але всі сучасні двигуни використовують JIT або компіляцію під час виконання для підвищення продуктивності. JavaScript на стороні сервера JavaScript на стороні сервера відноситься до мови кодування у внутрішній логіці сервера. У цьому випадку механізм JavaScript знаходиться безпосередньо на сервері. Функція JavaScript на стороні сервера може отримувати доступ до бази даних, виконувати різні логічні операції та реагувати на різні події, ініційовані операційною системою сервера. Основна перевага сценаріїв на стороні сервера полягає в тому, що користувач може суттєво налаштувати реакцію сайту на потреби користувача, доступ і запити інформації.

Динамічні слова описують як клієнтський, так і серверний JavaScript. Динамічна поведінка — це можливість оновлювати відображення вебсторінки для створення нового вмісту за потреби. Різниця між клієнтським і серверним JavaScript полягає у створенні нового вмісту. Код на стороні сервера автоматично створює новий вміст за допомогою логіки програми та змінює дані бази даних. З іншого боку, JavaScript на стороні клієнта автоматично створює новий вміст у браузері за допомогою логіки інтерфейсу користувача та змінює наявний вміст вебсторінки на стороні клієнта. Значення дещо відрізняються в цих двох контекстах, але вони пов'язані, і два підходи працюють разом, щоб покращити взаємодію з користувачем.

Ще одна відмінність між двома використаннями JavaScript, окрім реалізації динамічних функцій, полягає в ресурсах, до яких код може отримати доступ. На стороні клієнта браузер керує середовищем JavaScript. Цей код може отримати доступ лише до ресурсів, авторизованих браузером. Наприклад, він не може записати вміст на жорсткий диск, доки користувач не натисне кнопку завантаження. З іншого боку, функції на стороні сервера можуть отримати доступ до всіх ресурсів сервера за потреби.

2.4 Мова програмування PHP

PHP є популярною мовою програмування, особливо серед веброзробників. Автором першої версії є Расмус Лердорф, якому прийшла в голову ідея розробити набір інструментів для спрощення процесу створення динамічних вебсторінок. Хоча сучасний PHP є мовою загального призначення, він часто використовується як серверний інструмент для створення HTML-коду, який потім інтерпретується браузером. Це мова програмування з відкритим кодом, розроблена програмістами-ентузіастами з усього світу. Він має простий синтаксис, дещо схожий на Java та C. Це проект, який постійно розвивається, наразі використовується 7-ма версія цієї мови.

За статистикою, кожен шостий програмний продукт створений на PHP. Сьогодні існує дві основні сфери використання PHP: написати сценарії та завершити вебдодатки, які працюють на стороні сервера.

Це найпопулярніша область застосування, оскільки мова спочатку була створена спеціально для веброзробки. Сервер, парсер (програма CGI) і клієнтське програмне забезпечення (браузер), що відображає результати виконання коду, необхідні для належного функціонування вебпрограми, написаної на PHP. Для створення сценаріїв, які запускаються в командному рядку. Ці невеликі програми можуть працювати на будь-якому ПК. Для їх запуску потрібен лише один парсер.

Оскільки PHP містить потужні інструменти для роботи з рядками, ці сценарії часто створюють для обробки текстових даних. Написати графічні інтерфейси. PHP має багато гілок, створених для виконання різних завдань. Одним з таких форків є PHP-GTK. Він часто використовується програмістами, знайомими з синтаксисом PHP.

Популярність PHP пояснюється такими його перевагами:

- синтаксис простий та інтуїтивно зрозумілий. PHP швидко

освоюють навіть початківці програмісти. Він поєднує в собі всі найкращі функції популярних мов, таких як C, Java і Perl. Код PHP легко читається в будь-якому режимі використання (для написання невеликих сценаріїв або створення потужних додатків з використанням об'єктно-орієнтованих методів для реалізації програм);

- кросплатформенність і гнучкість. PHP сумісний з усіма популярними платформами (Linux, Windows, MacOS). Програми, написані на ньому, успішно працюють на різному серверному програмному забезпеченні (IIS, Nginx, Apache та бага-то інших);

- чудова масштабованість. PHP дозволяє досягти максимальної продуктивності написаних на ньому додатків із зростанням апаратних ресурсів. Вебдодатки розподіляються на кількох серверах, здатних витримувати великі навантаження (високий трафік).

- портування в HTML-документ. Користувач може легко додати динамічний вміст до HTML-сторінки, просто вставивши блоки коду PHP. Вони додаються як теги HTML та не порушують структуру документа, активно розвиваються та вдосконалюються.

Спільнота розробників постійно працює над впровадженням додаткових функцій, які розширюють можливості мови, спрощують синтаксис і покращують захист від можливих атак.

На офіційному сайті проекту представлена повна інформація про функціональні одиниці з прикладами вживання. Використання мови PHP пропонує простий пошук рішень проблем, що виникають під час розробки. В Інтернеті є багато форумів, присвячених програмуванню на PHP. Експерти відзначають великі перспективи подальшого розвитку даної мови програмування. Також треба зауважити, що більшість CMS побудовано на чистому PHP і фреймворках. Це пов'язано з популярністю та затребуваністю PHP-програмістів[9].

2.5 MySQL

MySQL — безкоштовна система управління реляційними базами даних, розроблена компанією TsX для прискорення обробки великих баз даних. Ця система управління базами даних (СУБД) з відкритим кодом була створена як альтернатива комерційним системам. На початку MySQL була дуже схожа на mSQL, але з часом вона еволюціонувала, і на даний момент MySQL є однією з найпопулярніших систем керування базами даних. Він в основному використовується для створення динамічних вебсторінок, оскільки має чудову підтримку багатьох мов програмування. MySQL з'явився в результаті спроби застосувати mSQL до власних розробок компанії: таблиць, які використовують ISAM, підпрограму низького рівня для доступу до індексованих даних. У результаті був розроблений новий інтерфейс SQL, але інтерфейс API був успадкований від mSQL.

Походження назви "MySQL" не є однозначним після того, як корпорація Oracle придбала Sun Microsystems у 2009р.

Під час розробки під керівництвом Oracle MySQL все більше ізолювалася від спільноти та робила процес розробки все менш і менш прозорим. Наприклад, відновлено практику надання розширених пропрієтарних функцій у корпоративній версії MySQL, інформацію про приховані вразливості, набори тестів видалено з компонентів, доступ до більшості систем. Система відстеження помилок закрита та публікує журнали групових змін, що дозволяє оцінка асоційованих виправлень із конкретними змінами[10].

MySQL — це безкоштовний відкритий для усіх бажаючих сервер. Серед його основних переваг слід виділити високу швидкість, стабільність і простоту використання. MySQL вважається гарним рішенням для малих і середніх за обсягом баз даних. Вихідний код сервера скомпільовано на кількох платформах. Вся функціона-

льність сервера міститься в системах UNIX, у яких підтримується багатопотоковість, що підвищує продуктивність усієї системи. [10].

2.6 Visual Studio Code

Visual Studio Code — дуже популярний безкоштовний редактор коду від Microsoft. Редактор допомагає веброзробникам і дизайнерам будь-якого рівня кваліфікації. З одного боку, він підходить новачкам через простий та інтуїтивно зрозумілий інтерфейс. З іншого боку, VS Code має багато вбудованих функцій, які цікаві досвідченим розробникам. При запуску редактора вперше відкривається стартова сторінка, де описані всі можливості редактора. Усі посилання є інтерактивними та відкривають потрібну частину інтерфейсу або сторінку в браузері. Редактор підтримує кілька мов програмування і його легко налаштувати. Користувач може встановити різні комбінації клавіш і колірні схеми. Основну частину екрана займає вікно з текстовим вмістом файлу. У лівій частині екрана знаходяться вкладки активного меню, де розташовані основні функції редактора. Вкладка Провідник відкривається за замовчуванням під час запуску програми. Він відображає список відкритих файлів і каталог відкритої папки. VS Code підтримує систему контролю версій Git. Користувач може виконувати основні операції та дивитися, як змінилися файли з часу останнього запуску. Конфігурація Git у Visual Studio Code детально описана в офіційній документації. VS Code відрізняється від своїх конкурентів наявністю вбудованого відладчика. Після невеликої конфігурації користувач можете знайти помилки в коді безпосередньо з редактора, наприклад, встановити точки зупину та спостерігати за виконанням певного фрагмента коду. Крім того, редактор має вбудовану панель управління, де можуть відобразитися результати роботи або повідомлення про помилки у разі виникнення проблеми. Відладчик можна налаштувати для різних мов і різних завдань[13].

2.7 Висновки за розділом

В розділі досліджено та описано інструменти розробки. Підведено підсумки, які саме інструменти буде обрано при розробці практичної частини випускної роботи, а саме: вибрано мови програмування HTML, CSS, JS, PHP.

У кожній з представлених мов є свої сильні сторони, за якими вони були обрані а саме:

- HTML було обрано за структурне відображення сторінки на вмісту;
- CSS, щоб покращити візуальну частину сайту;
- JS для створення інтерактивних карт та анімації;
- PHP для спрощення створення динамічних вебсторінок.

Для збереження всієї інформації було обрано MySQL, тому що це безкоштовний, а саме тому доступний та зручний у використанні.

РОЗДІЛ 3

ПРАКТИЧНА РЕАЛІЗАЦІЯ СИСТЕМИ

3.1 Архітектура програми

Щоб краще розуміти, що повинно бути імплементовано в додатку і як це повинно взаємодіяти, були створені UML діаграми прецедентів та станів. Види UML-діаграм відображені на рисунках 3.1 – 3.2



Рис. 3.1 — Діаграма прецедентів

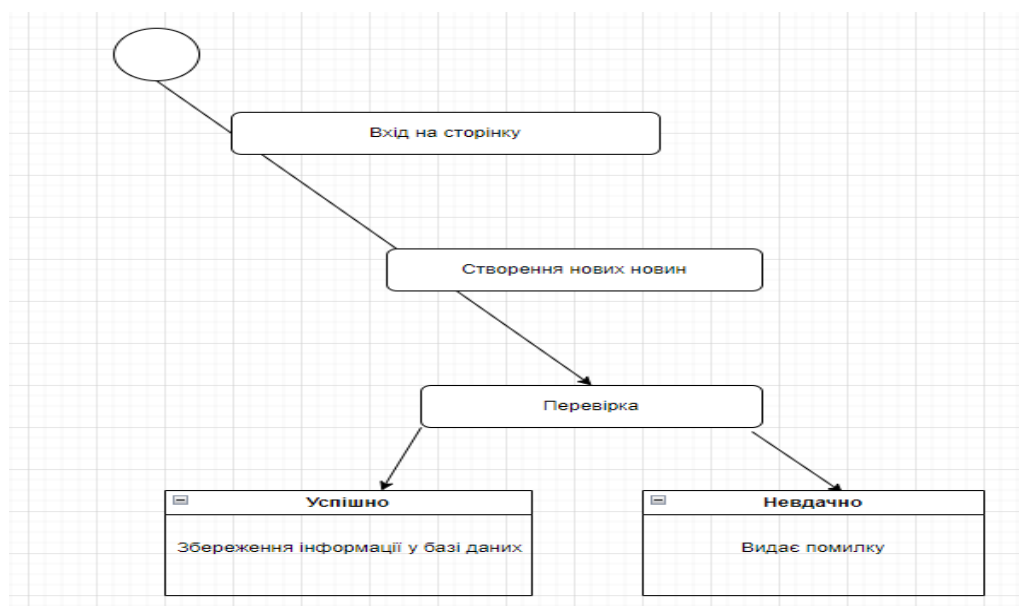


Рис. 3.2 — Діаграма станів

3.2 Властивості коду

Спочатку було створено базу даних у MySQL, для збереження певної інформації. Огляд заповненої бази даних представлено рисунку 3.3.

The screenshot shows the phpMyAdmin interface for a database named 'dr'. The left sidebar displays a tree view of the database structure, including tables 'fact1' through 'fact10', 'history1' through 'history10', 'news', and 'text1' through 'text10'. The main window shows a table structure view with columns: Таблица, Действие, Строки, Тип, Сравнение, Размер, and Фрагментировано. The table list includes:

Таблица	Действие	Строки	Тип	Сравнение	Размер	Фрагментировано
fact1		1	InnoDB	utf8_general_ci	16.0 КиБ	-
fact2		1	InnoDB	utf8_general_ci	16.0 КиБ	-
fact3		1	InnoDB	utf8_general_ci	16.0 КиБ	-
fact4		1	InnoDB	utf8_general_ci	16.0 КиБ	-
fact5		1	InnoDB	utf8_general_ci	16.0 КиБ	-
fact6		1	InnoDB	utf8_general_ci	16.0 КиБ	-
fact7		1	InnoDB	utf8_general_ci	16.0 КиБ	-
fact8		1	InnoDB	utf8_general_ci	16.0 КиБ	-
fact9		1	InnoDB	utf8_general_ci	16.0 КиБ	-
fact10		1	InnoDB	utf8_general_ci	16.0 КиБ	-
history1		1	InnoDB	utf8_general_ci	16.0 КиБ	-
history2		1	InnoDB	utf8_general_ci	16.0 КиБ	-
history3		1	InnoDB	utf8_general_ci	16.0 КиБ	-
history4		1	InnoDB	utf8_general_ci	16.0 КиБ	-
history5		1	InnoDB	utf8_general_ci	16.0 КиБ	-
history6		1	InnoDB	utf8_general_ci	16.0 КиБ	-
history7		1	InnoDB	utf8_general_ci	16.0 КиБ	-
history8		1	InnoDB	utf8_general_ci	16.0 КиБ	-
history9		1	InnoDB	utf8_general_ci	16.0 КиБ	-
history10		1	InnoDB	utf8_general_ci	16.0 КиБ	-
news		1	InnoDB	utf8_general_ci	16.0 КиБ	-
text1		1	InnoDB	utf8_general_ci	16.0 КиБ	-
text2		1	InnoDB	utf8_general_ci	16.0 КиБ	-
text3		1	InnoDB	utf8_general_ci	16.0 КиБ	-
text4		1	InnoDB	utf8_general_ci	16.0 КиБ	-
text5		1	InnoDB	utf8_general_ci	16.0 КиБ	-
text6		2	InnoDB	utf8_general_ci	16.0 КиБ	-
text7		1	InnoDB	utf8_general_ci	16.0 КиБ	-
text8		1	InnoDB	utf8_general_ci	16.0 КиБ	-
text9		1	InnoDB	utf8_general_ci	16.0 КиБ	-
text10		1	InnoDB	utf8_general_ci	16.0 КиБ	-

Рис. 3.3 — Створена база даних

Для того що б виводити інформацію, потрібно виконати підключення до самої бази даних, що реалізовано за допомогою класу Database.

Лістинг 3.1 — Підключення бази даних

```
<?php
```

```
$database = new Database();
```

```
$db = $database->getConnection();
```

```
class Database {
```

```
    private $host = "localhost";
```

```
    private $db_name = "creat";
```

```
    private $username = "root";
```

```
    private $password = "root";
```

```
    public $conn;
```

```

public function getConnection() {
    $this->conn = null;

    try {
        $this->conn = new PDO("mysql:host=" . $this->host .
";dbname=" . $this->db_name, $this->username, $this->password);
    } catch(PDOException $exception) {
        echo "Connection error: " . $exception->getMessage();
    }

    return $this->conn;
}
}
?>

```

Після того як було підключено базу даних, потрібно виводити звітти інформацію з допомогою методу `fetch_all`, виведення інформації реалізується змінними, такими як `$Fact1`.

Лістинг 3.2 — Вивід інформації на сторінки

```

$result = mysqli_query($link, "SELECT * FROM `Fact1`");
$result = mysqli_fetch_all($result);
foreach ($result as $Fact1) {
}

```

Структуру таблиці для збереження даних в якій знаходиться заголовок, інформація та картинки зображено на рисунку 3.4.

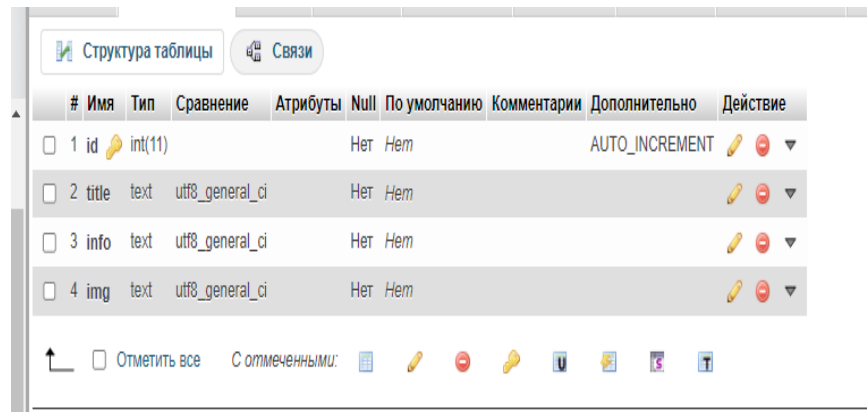


Рис. 3.4 — Структура таблиці

Для зручного виведення даних було створено таблицю для збереження інформації. Вихідну таблицю бази даних представлено на рисунку 3.5.

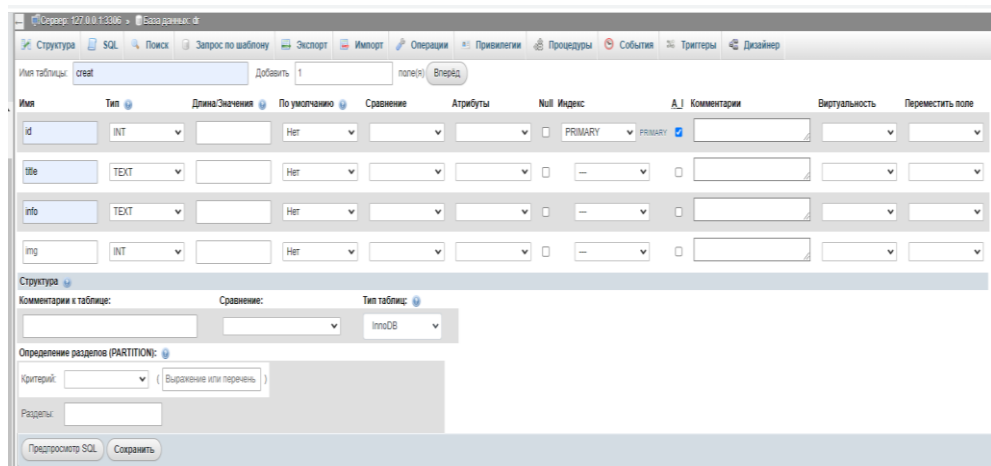


Рис. 3.5 — Видяг створеної таблиці

На рисунку 3.6 можна побачити вигляд заповненої таблиці, дані з якої будуть виводитися на вебсторінку сайту новин.

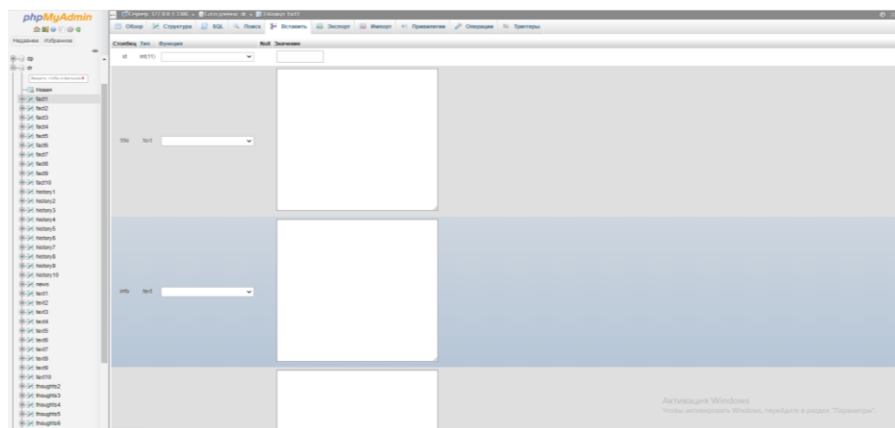


Рис. 3.6 — Приклад заповнення таблиці

Слайдер-меню було реалізовано з допомогою бібліотеки `tns`, яка виконує автоматичний показ слайдів.

Лістинг 3.3 — Реалізація слайдеру

```
const slider = tns({
  container: '.my-slider' ,
  loop: true,
  items: 1,
  slideBy: 'page',
  nav: false,
  autoplay: true,
  speed: 400,
  autoplayButtonOutput: false,
  mouseDrag: true,
  lazyload: true,
  controlsContainer: "#customize-controls",
  responsive: {
    640: {
      items: 2,
    },

    768: {
      items: 3,
    }
  }
});
```

У даному лістингу реалізовано випадające меню, яке спрацьовує при натисканні на список та закривається при натисканні або на інший список або довільно по екрану, завдяки методу `addEventListener`, який приймає повідомлення о подіях (натисканні) на сторінці.

Лістинг 3.4 — Меню кнопок

```

document.querySelectorAll('.dropdown-toggle').forEach(e => {
  e.addEventListener('click', e => {
    const menu = e.currentTarget.dataset.path;
    document.querySelectorAll('.dropdown-menu').forEach(e
=> {
      if (!document.querySelector(`[data-
target=${menu}]`).classList.contains('open')) {
        e.classList.remove('menu-active');
        e.classList.remove('open');
        document.querySelector(`[data-
target=${menu}]`).classList.add('menu-active');
        intervalId = setTimeout(() => {
          document.querySelector(`[data-
target=${menu}]`).classList.add('open');
        }, 0);
      }

      if (document.querySelector(`[data-
target=${menu}]`).classList.contains('open')) {
        clearTimeout(intervalId);
        document.querySelector(`[data-
target=${menu}]`).classList.remove('menu-active');
        intervalId = setTimeout(() => {
          document.querySelector(`[data-
target=${menu}]`).classList.remove('open');
        }, 0);
      }

      window.onclick = e => {
        if (e.target == document.querySelector(`[data-
target=${menu}]`) || e.target == document.querySelector(`[data-
path=${menu}]`)) {
          return;
        } else {

```

```

        document.querySelector(`[data-
target=${menu}]`).classList.remove('menu-active');
        document.querySelector(`[data-
target=${menu}]`).classList.remove('open');
    }
}
});
});
});

```

Було розроблено гамбургер-меню для зручності користувача, для цього використовується метод `addEventListener` після якого відбуваються такі події як відкриття меню по натисненню на кнопки та закриття.

Лістинг 3.5 — Реалізація меню

```

document.addEventListener("DOMContentLoaded", function() {

document.getElementById("burger").addEventListener("click",
function() {

document.querySelector(".header").classList.toggle("open")
    })
})
window.addEventListener('keydown', (e) => {
    if (e.key === "Escape") {

document.querySelector(".header").classList.remove("open")
    }
});

document.getElementById("menu").addEventListener('click',
event => {
    event._isClickWithInMenu = true;

```

```

});
document.getElementById("burger").addEventListener('click',
event => {
    event._isClickWithInMenu = true;
});
document.body.addEventListener('click', event => {
    if (event._isClickWithInMenu) return;

    document.querySelector(".header").classList.remove("open")
});

```

Було розроблено media-запити, щоб адаптувати вебсайт для усіх пристроїв, що дозволить зручно переглядати сайт з використанням будь-якого розширення екрану.

Лістинг 3.6 — Медіа-запит

```

@media only screen and (max-width: 768px) {
    .container{
        width: 100%;
    }

    .hilen{
        width: none;
        height: none;
    }

    .header{
        background-color: saddlebrown;
    }

    .center{
        background-color: rebeccapurple;
    }
}

```

```

.header__list-link.header__list-link--active{
  font-size: 15px;
}

.header__nav{
  margin-left: 0%;
}

.header__logo-link{
  font-size: 20px;
}

.header__logo{
  margin-left: 7%;
}

.box{
  height: 400px;
}

```

Було розроблено фільтр для новин, які працюють за допомоги `targetId`. Було задано `Id` новинам наприклад: `old`, `new` за якими відбувається сортування.

Лістинг 3.7 — Фільтр новин

```

const list = document.querySelector('.filter'),
  item = document.querySelectorAll('.hilen')

function filter(){
  list.addEventListener('click', event => {
    const targetId = event.target.dataset.id
    console.log(targetId)

    switch(targetId){

```

```

        case 'all':
            getItem('hilen')
            break
        case 'new':
            getItem(targetId)
            break
        case 'old':
            getItem(targetId)
            break
    }
})
}
filter()

function getItem(className) {
    item.forEach(item => {
        if (item.classList.contains(className)) {
            item.style.display = 'block'
        } else {
            item.style.display = 'none'
        }
    })
}
}

```

Спочатку відбувається підключення до бази даних, після цього задаються типи полів, наприклад `type = title`, щоб у подальшому виводити з їх допомогою дані до БД.

Лістинг 3.8 — Перенесення даних

```

<?php
$link = mysqli_connect("127.0.0.1","root","","dr");

$title = $_POST['title'];
$info = $_POST['info'];

```

```

$img = $_POST['img'];

mysqli_query($link,"INSERT INTO `creat` (`id`, `title`,
`info`, `img`) VALUES (NULL, '$title', '$info', '$img')");

header('Location: index.html');
?>

```

3.3 Користувацькі властивості

При вході на вебсторінку користувача зустрічає головне меню, у якому є слайдер з новинами, випадаюче меню та кнопки, завдяки яким можна подивитися як усі новини так і застосувати фільтр для перегляду нових новин або ж архіву усіх новин, наприкінці сторінки розміщенні посилання. Вигляд головної сторінки представлено на рисунку 3.7.

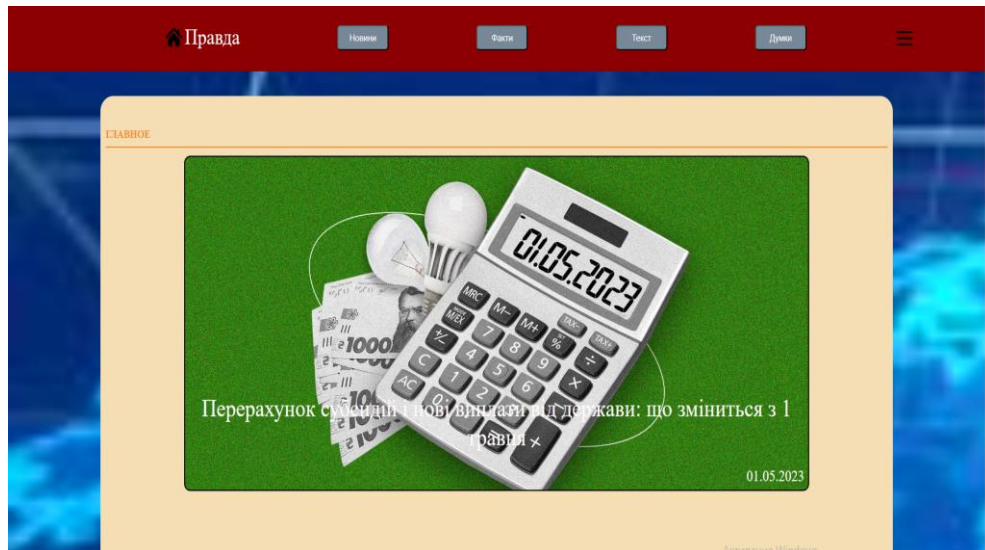


Рис. 3.7 — Головне меню

Також на головній сторінці присутні автоматичні слайдери з новинами, реалізовані за допомогою бібліотеки tns, яке дає змогу розмістити n-кількість новин, що продемонстровано на рисунку 3.8.

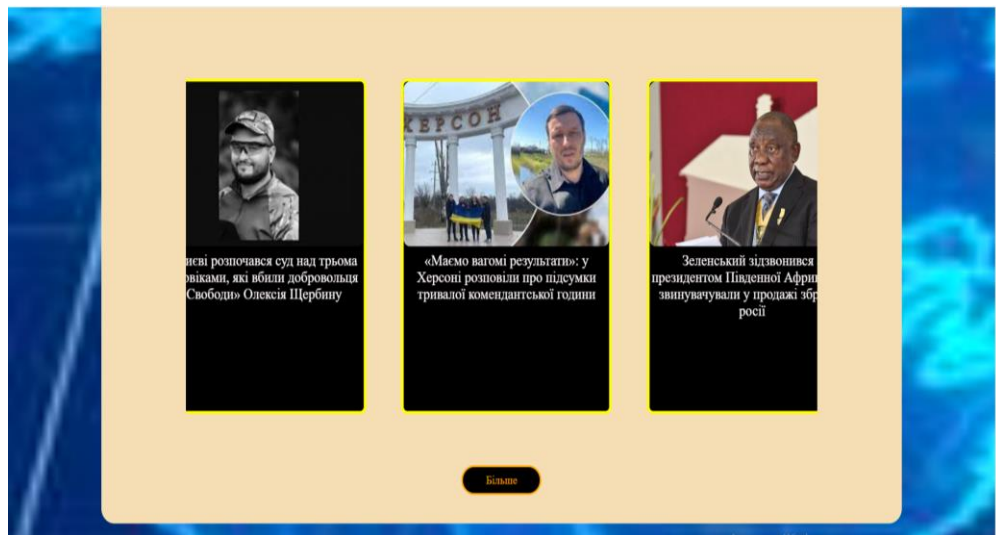


Рис. 3.8 — Слайдер на сторінці головного меню

При натисканні кнопки Новини => Усі, користувача перемістить до усіх сучасних новин, такий самий результат користувач отримує якщо обере шлях Новини => Сучасні. За таким самим сценарієм користувач може перейти до архіву з застарілими новинами, якщо обере параметр Старі замість Сучасні. Результат цих двох варіантів дій відображено на рисунках 3.9-3.10 відповідно.

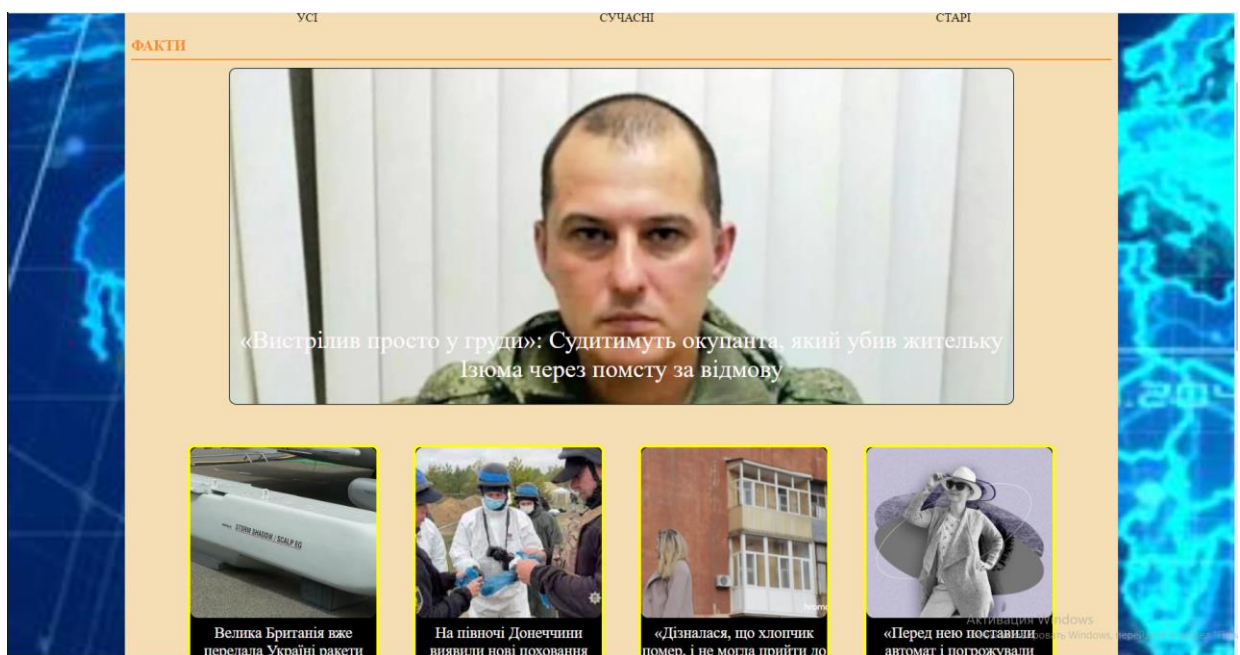


Рис. 3.9 — Сучасні новини

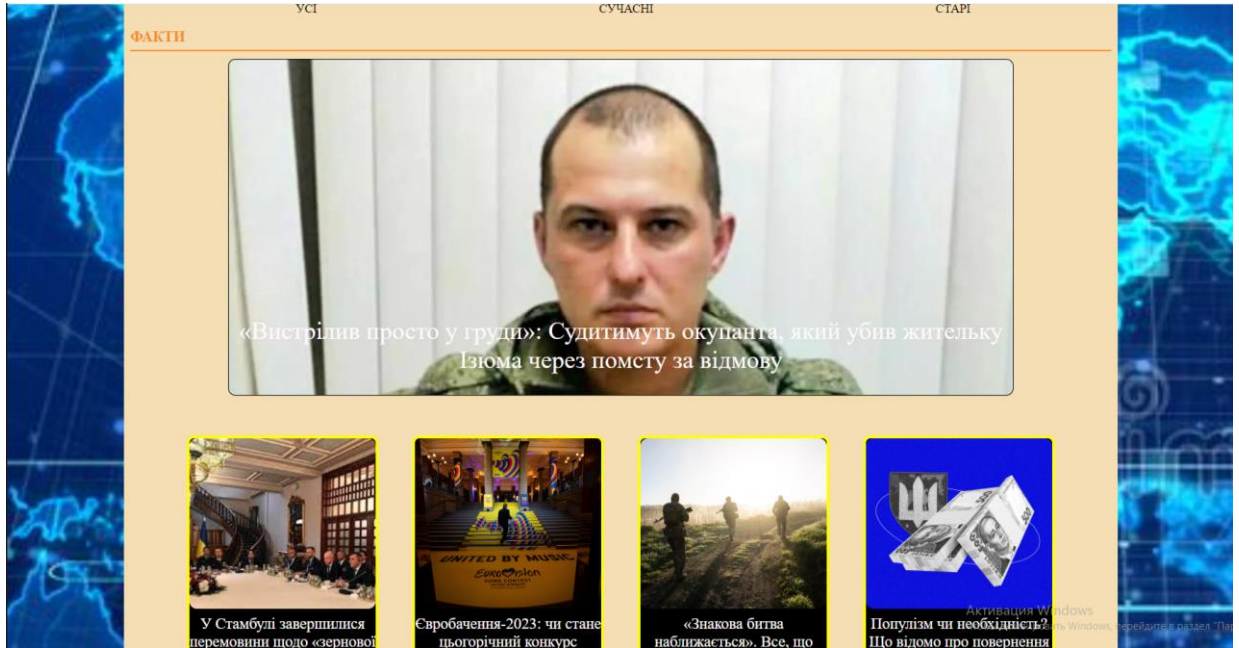


Рис 3.10 — Старі новини

При переході на будь-які новини, на сторінці буде розміщена певна інформація, яка зберігається у базі даних. Приклади зображено на рисунках 3.11-3.12.

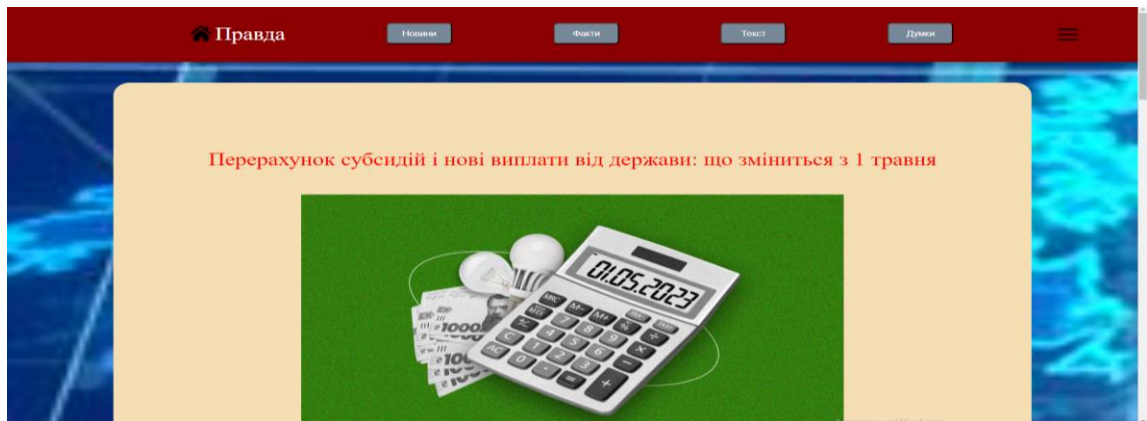


Рис. 3.11 — Сторінка з інформацією

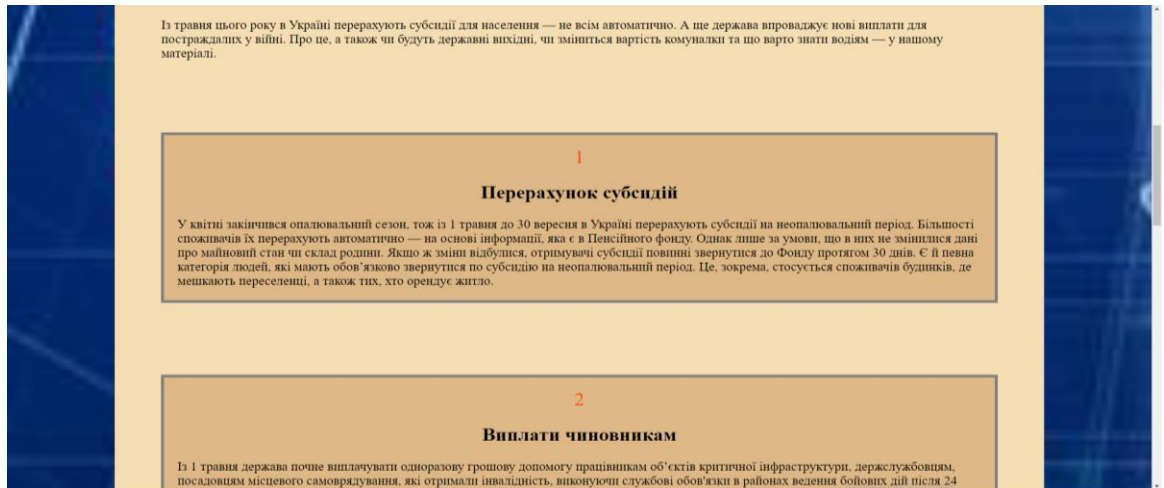


Рис. 3.12 — Сторінка з інформацією

Нижче всіх наведених сторінок є посилання з інформацією про сайт, політику сайту та на новини, вигляд якої представлено на рисунку 3.13.

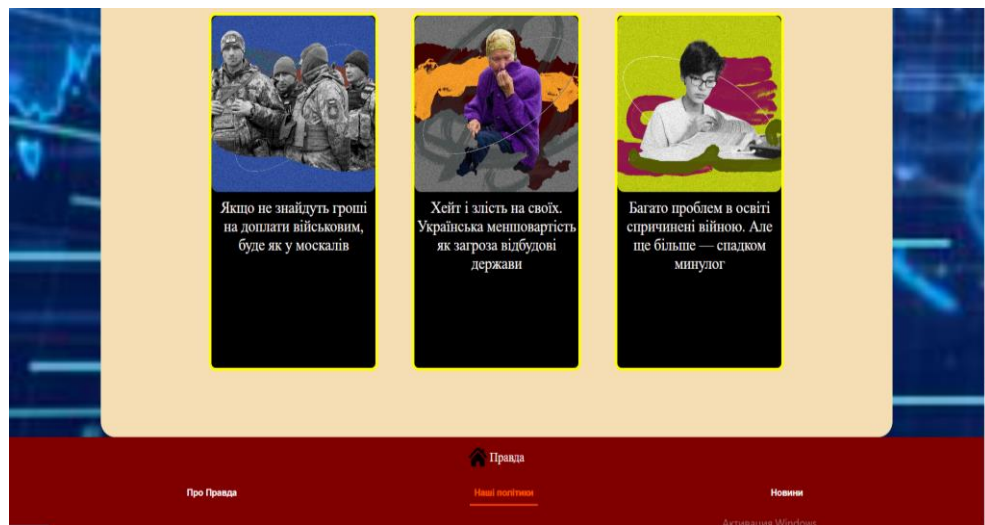


Рис. 3.13 — Посилання з інформацією про сайт

Приклад відображення інформації, що отримає користувач при натисканні кнопки «Наша політика» представлено на рисунку 3.14.

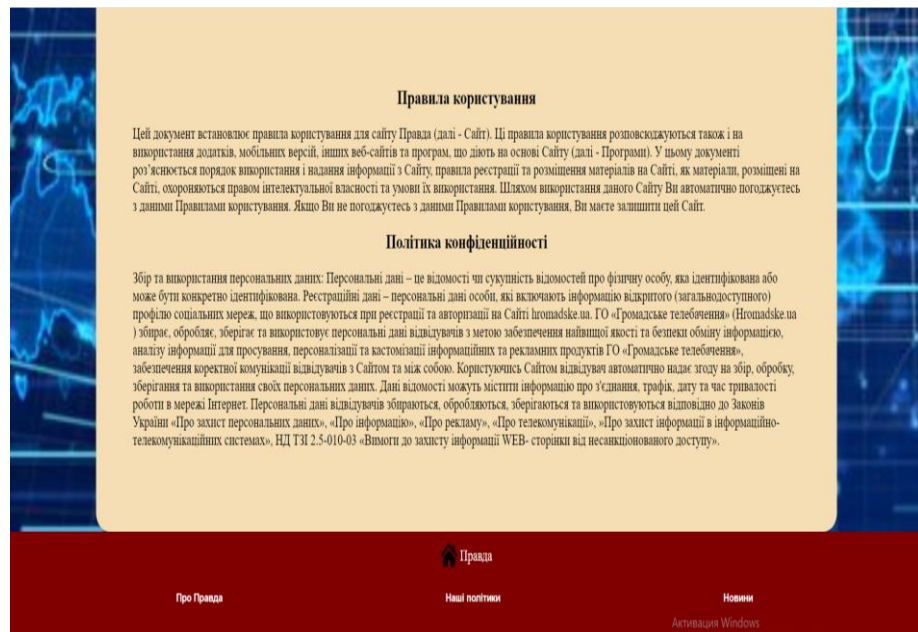


Рис. 3.14 — Політика сайту новин

При зменшенні екрану спрацьовує media-форма, у якій розташовані стилі, призначені під різні показники розширення екрану, адаптивність надає змогу сайту бути універсальним та пристосовуватися під різних користувачів. Версія сайту для екранів мобільних телефонів представлена рисунку 3.15.

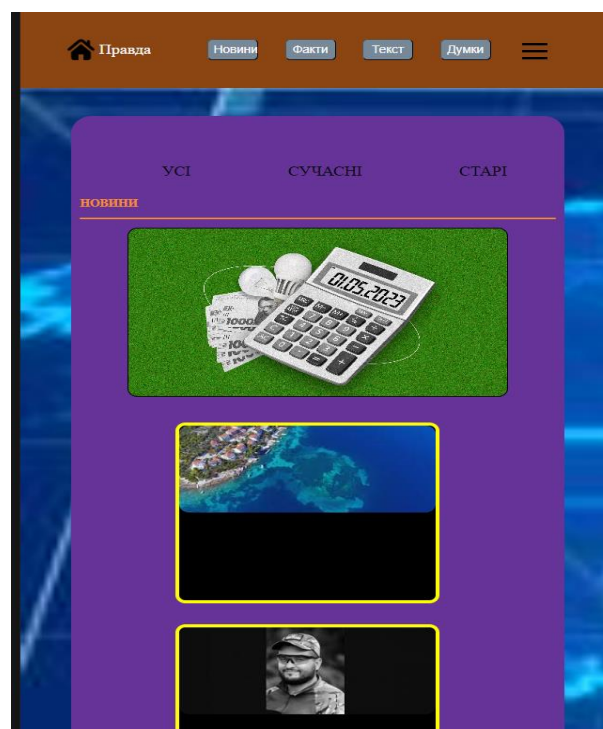


Рис. 3.15 — Адаптивна версія сайту

Для зручного та швидкого додавання новин до сайту було впроваджено форму, інформація з якої переноситься до бази даних для створених новин з сайту (Create). Загальний вигляд застосовуваної форми представлено на рисунку 3.16.

The image shows a web browser window with a dark red header. The header contains the logo 'Правда' and four navigation buttons: 'Новини', 'Факти', 'Текст', and 'Думки'. The main content area has a light blue background with a globe graphic. In the center, there is a white form titled 'Створення новин'. The form has three input fields: 'Заголовок', 'Інформація', and 'Посилання на картинку'. Below the form is a 'Додати' button. The bottom of the browser window shows a Windows activation watermark.

Рис. 3.16 — Форма заповнення новин

3.4 Висновки за розділом

Було побудовано дві UML діаграми: діаграму прецедентів та діаграму станів, де зображено, що може робити користувач на вебсторінці та що відбувається у базі даних під час заповнення формі новин.

Написано код для вебсторінки мовами програмування, що було обрано у другому розділі, розроблено фундамент, після цього були задані стилі та медіа-запити, для побудованого макету було розроблено анімації (слайдер, бюргер-меню, випадаючі списки) та виведено інформацію з БД на вебсторінку.

Після етапу програмування було представлено результати роботи у вигляді рисунків, що демонструють працездатність сайту новин.

ВИСНОВКИ

Огляд з історії розвитку інтернет-сайтів установив, що дана область інформаційних технологій активно розвивається та залишається актуальною. Були розібрані аналоги порталів новин, що дало змогу проаналізувати як сильні сторони так і недоліки.

Проведена класифікація встановила доцільність розробки сайту інформаційного типу для предметної галузі новин. Із наведених способів створення сайтів було обрано HTML, CSS, JS, PHP, що дозволить у простий спосіб створити вебсайт з інформаційних сторінок, контент яких буде зберігатись у базі даних та виводитися на сторінки з новинами.

У ході розробки були вивчені різноманітні бібліотеки та реалізовані наступні кроки:

- було обдумано та створено архітектуру додатку;
- створено схеми об'єктів додатку;
- реалізовано збереження інформації у базі даних;
- оглянуто додатки та зроблені висновки.

ПЕРЕЛІК ПОСИЛАНЬ

- 1 Історія виникнення вебсайтів - Вікіпедія. Інтернет-енциклопедія [Електронний ресурс]. Режим доступу: <https://home.cern/science/computing/birth-web/short-history-web> -30.04.2023
- 2 Основні види сайтів - Вікіпедія. Інтернет-енциклопедія [Електронний ресурс]. Режим доступу: <https://avada-media.ua/tipy-sajtov/>-30.04.2023
- 3 Способи створення вебсайту - Інтернет-ресурс [Електронний ресурс]. Режим доступу: [www. URL= https://vc.ru/dev/78714-3-sposoba-razrabotki-veb-sayta](https://vc.ru/dev/78714-3-sposoba-razrabotki-veb-sayta)- 30.04.2023.
- 4 HTML - web_doxs. Інтернет-енциклопедія [Електронний ресурс]. Режим доступу: [www.URL=https://developer.mozilla.org/ru/docs/Learn/Getting_started_with_the_web/HTML_basics](https://developer.mozilla.org/ru/docs/Learn/Getting_started_with_the_web/HTML_basics) - 30.04.2023.
- 5 CSS - Вікіпедія. Інтернет-енциклопедія [Електронний ресурс]. Режим доступу: [www.URL=https://developer.mozilla.org/ru/docs/Learn/Getting_started_with_the_web/CSS_basics](https://developer.mozilla.org/ru/docs/Learn/Getting_started_with_the_web/CSS_basics)- 30.04.2023.
- 7 Інтернет-ресурс [Електронний ресурс]. Режим доступу: [www. URL= https://loftblog.ru/material/osnovy-css-urok-2/](https://loftblog.ru/material/osnovy-css-urok-2/)-30.04.2023.
- 8 JS - Вікіпедія. Інтернет-енциклопедія [Електронний ресурс]. Режим доступу: [www. URL= https://aws.amazon.com/ru/what-is/javascript/](https://aws.amazon.com/ru/what-is/javascript/)- 30.04.2023.
- 9 PHP - Вікіпедія. Інтернет-енциклопедія [Електронний ресурс]. Режим доступу: [www. URL= https://freehost.com.ua/ukr/faq/wiki/chto-takoe-php/](https://freehost.com.ua/ukr/faq/wiki/chto-takoe-php/) - 30.04.2023.
- 10 MySQL - Вікіпедія. Інтернет-енциклопедія [Електронний ресурс]. Режим доступу: [www. URL= https://uk.wikipedia.org/wiki/MySQL](https://uk.wikipedia.org/wiki/MySQL)- 30.04.2023.
- 11 ВотТак Вебсторінка новин – Інтернет-ресурс [Електронний ресурс]. Режим доступу: <https://vot-tak.tv/>-30.04.2023
- 12 Hromadske Вебсторінка новин – Інтернет-ресурс [Електронний ресурс]. Режим доступу: <https://hromadske.ua/>-30.04.2023

13 Visual Studio Code – Інтернет енциклопедія [Електронний ресурс].Режим доступу: <https://htmlacademy.ru/blog/soft/vscode>-30.04.2023

14 HTML Вебсторінка новин – Інтернет-ресурс [Електронний ресурс].
Режим доступу: <https://mc.today/что-такое-html/>-30.04.2023

15 CSS Вебсторінка новин – Інтернет-ресурс [Електронний ресурс].
Режим доступу: <https://uk.wikibooks.org/wiki/CSS>-30.04.2023