

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Національний університет «Запорізька політехніка»

Факультет комп'ютерних наук і технологій

(повне найменування інституту, факультету)

Кафедра програмних засобів

(повне найменування кафедри)

Пояснювальна записка

до дипломного проєкту (роботи)

бакалавр

(ступінь вищої освіти)

на тему РОЗРОБКА ВЕБЗАСТОСУНКУ ДЛЯ ВЕДЕННЯ ФОРУМУ
З АВТОМОБІЛЬНОЇ ТЕМАТИКИ
DEVELOPMENT OF A WEB APPLICATION FOR AN
AUTOMOTIVE FORUM

Виконав: студент 4 курсу, групи КНТ-139
Спеціальності 121 Інженерія програмного
забезпечення

(код і найменування спеціальності)

Освітня програма (спеціалізація)

Інженерія програмного забезпечення

ЧЕРНЯВСЬКИЙ К.Є.

(ПРІЗВИЩЕ та ініціали)

Керівник КАПЛІЄНКО Т.І

(ПРІЗВИЩЕ та ініціали)

Рецензент ШИТІКОВА О.В.

(ПРІЗВИЩЕ та ініціали)

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Національний університет «Запорізька політехніка»
 (повне найменування закладу вищої освіти)

Факультет ФКНТ
 Кафедра програмних засобів
 Ступінь вищої освіти бакалавр
 Спеціальність 121 Інженерія програмного забезпечення
 (код і найменування)
 Освітня програма (спеціалізація) Інженерія програмного забезпечення
 (назва освітньої програми (спеціалізації))

ЗАТВЕРДЖУЮ

Завідувач кафедри ПЗ, д.т.н, проф.
Сергій СУББОТІН

“ ” 2023 року

З А В Д А Н Н Я

НА ДИПЛОМНИЙ ПРОЄКТ (РОБОТУ) СТУДЕНТА(КИ)

ЧЕРНЯВСЬКОГО Кирила Євгеновича

(ПРИЗВИЩЕ, ім'я, по батькові)

1. Тема проєкту (роботи) Розробка вебзастосунку для ведення форуму з автомобільної тематики.

Development of a Web Application for an Automotive Forum.

керівник проєкту (роботи) КАПЛІЄНКО Т.І., к.т.н., доцент,
 (ПРИЗВИЩЕ, ім'я, по батькові, науковий ступінь, вчене звання)

затверджені наказом закладу вищої освіти від “04” квітня 2023 року №87

2. Строк подання студентом проєкту (роботи) 07 червня 2022 року

3. Вихідні дані до проєкту (роботи) рекомендована література, технічне завдання

4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити) 1. Аналіз проблеми та постановка завдань дослідження. 2. Матеріали і методи. 3. Опис програми. 4. Експлуатація, тестування та експериментальне дослідження програми.

5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень)
Слайди презентації

6. Консультанти розділів проєкту (роботи)

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	прийняв виконане завдання
1-5 Основна частина	КАПЛІЄНКО Т.І., доцент		
Нормоконтроль	КАМІНСЬКА Ж.К., асистент		

7. Дата видачі завдання « 06 » лютого 2023 року.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів дипломного проєкту (роботи)	Строк виконання етапів проєкту (роботи)	Примітка
1	Постановка завдання роботи.	1 тиждень	Завдання, ТЗ
2	Аналіз предметної області.	1 тиждень	Розділ 1
3	Вибір мови програмування та інших технологій розробки.	2 тиждень	Розділ 2
4	Розробка архітектури програми.	2 тиждень	Розділ 3
5	Розробка програми.	3-4 тижні	Розділ 3, 4
6	Тестування та експериментальне дослідження програмного забезпечення.	5 тиждень	Розділ 4
7	Оформлення пояснювальної записки та документів до неї.	6 тиждень	Додатки
8	Нормоконтроль та рецензування.	7 тиждень	
9	Захист роботи.	8 тиждень	

Студент(ка)

(підпис) Кирило ЧЕРНЯВСЬКИЙ
(ініціали та ПРІЗВИЩЕ)

Керівник проєкту (роботи)

(підпис) Тетяна КАПЛІЄНКО
(ініціали та ПРІЗВИЩЕ)

РЕФЕРАТ

Пояснювальна записка до дипломної кваліфікаційної роботи бакалавра: 67 с., 22 рис., 5 табл., 19 джерел., 2 додатки.

ВЕБЗАСТОСУНОК, ТЕМА, ВЕБФОРУМ, ПОВІДОМЛЕННЯ.

Об'єкт дослідження – методи та засоби побудови вебзастосунку для створення форуму.

Предмет дослідження – вебфорум з автомобільної тематики.

Мета роботи – розробка автомобільного вебфоруму для вирішення користувачами актуальних питань.

Матеріали, методи та технічні засоби: PHPStorm, Laravel, MariaDB, phpMyAdmin.

Створення цього вебфоруму є актуальним через високий попит на подібні рішення в Інтернеті, зокрема серед автомобілістів, що утворюють спільноту.

В ході виконання проєкту був створений вебфорум, спрямований на обговорення автомобільних тематик. В процесі розробки були виправлені більшість недоліків, які були присутні у подібних рішеннях, зберігаючи при цьому їх переваги.

ABSTRACT

Explanatory note to the bachelor's thesis: 67 pages, 22 figures, 5 tables, 2 appendices, 19 sources.

WEB APPLICATION, WEB FORUM, MESSAGE, SUBJECT.

The object of research is methods and means of building a web application for creating a forum.

The subject of the research is a web forum on automobile topics.

The purpose of the work is the development of an automotive web forum for users to solve current issues.

Materials, methods and technical means: PHPStorm, Laravel, MariaDB, phpMyAdmin.

The creation of this web forum is relevant due to the high demand for such solutions on the Internet, particularly among motorists who form a community.

During the implementation of the project, a web forum aimed at discussing automotive topics was created. During the development process, most of the shortcomings that were present in similar solutions were corrected, while preserving their advantages.

ЗМІСТ

ВСТУП	9
1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ.....	10
1.1 Вебфорум	10
1.2 Особливості та види форумів.....	11
2 ВИБІР ТА ОБҐРУНТУВАННЯ СИСТЕМИ, ЯКА ПРОЄКТУЄТЬСЯ	23
2.1 Вибір застосунків для розробки	23
2.1.1 Мова програмування.....	23
2.1.2 База даних	27
2.1.3 Середовище розробки.....	28
2.2 Висновки за розділом.....	29
3 ОСНОВНІ РІШЕННЯ ЩОДО РЕАЛІЗАЦІЇ КОМПОНЕНТІВ СИСТЕМИ	30
3.1 Використані моделі даних.....	30
3.2 Алгоритм введення даних у застосунок	32
3.3 Висновки за розділом 3	33
4 КЕРІВНИЦТВО ПРОГРАМІСТА.....	34
4.1 Використання та призначення застосунку	34
4.2 Характеристики програмного продукту	34
4.3 Вхідні та вихідні дані.....	36
4.4 Висновки за розділом.....	38
5 КЕРІВНИЦТВО ОПЕРАТОРА	39

5.1 Використання та призначення застосунку	39
5.2 Використання програмного продукту.....	39
5.3 Системні повідомлення	45
5.4 Висновки за розділом.....	45
ВИСНОВКИ.....	46
ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ	47
ДОДАТОК А	50
ДОДАТОК Б	60

ПЕРЕЛІК СКОРОЧЕНЬ ТА УМОВНИХ ПОЗНАК

API – Application Programming Interface;

БД – База даних;

ВЗ – вебзастосунок;

СКБД – системи керування базами даних;

ПЗ – програмний застосунок

ПП – програмний продукт;

ВСТУП

Сучасний світ пропонує безліч можливостей для спілкування та обміну інформацією, які майже всі доступні в Інтернеті. Один з найпопулярніших способів спілкування - вебфоруми, які знайшли своє застосування в багатьох сферах життя, зокрема в автомобільній тематиці. На таких форумах можуть обговорюватись питання будь-якого рівня складності, від новачків до професіоналів з великим досвідом.

З розвитком вебзастосунків програмістам доводиться працювати з багатьма технологіями, що робить форуми одним з найкращих способів знайти допомогу, вирішити будь-яку проблему та обмінятися досвідом. У зв'язку з попитом на подібні рішення в Інтернеті, включаючи україномовну спільноту, створення вебфоруму з автомобільної тематики є актуальною та важливою задачею.

В мережі Інтернет є багато старих форумів, які мають значну кількість обговорень та відповідей, але використовують застарілі технології та не оновлюються.

Для вирішення основних проблем, запозичуючи функціональні переваги старих форумів, було прийнято рішення розробити новий продукт.

1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ

У першому розділі дипломної роботи будуть проаналізовані особливості форумів в Інтернеті, зокрема тематичних форумів, що присвячені автомобільній тематиці.

Розділ буде присвячений дослідженню предметної області, розгляду особливостей використання форумів, огляду та порівнянню наявних рішень, визначенню їх переваг та недоліків, встановленню меж проєкту, опису функціональності системи, вимог до безпеки та інтерфейсу.

1.1 Вебфорум

Інтернет-мережа широко використовується для комунікації, обговорень та дискусій. Завдяки Інтернету значно легше знайти людину, яка розбирається в питанні, яке цікавить користувача, або знайти цікавого співрозмовника з подібними інтересами. Багато людей відчують себе більш розкріпленими під час онлайн-спілкування. Одним з видів ресурсів Інтернету, що використовуються для комунікації, є вебфорум.

Вебфорум - це сервіс Інтернету, який надає можливість проводити онлайн-дискусії, де люди спілкуються через створення тем для обговорення та подальше розміщення в них повідомлень [1]-[3].

Головна ідея форумів в Інтернеті полягає в обміні думками та досвідом, спілкуванні з однодумцями та наданні допомоги без будь-якої вигоди для себе. Форуми часто містять корисну інформацію, яка відсутня в Інтернеті або в автомобільних посібниках, оскільки користувачі діляться своїм реальним досвідом [1]-[3].

Форуми в Інтернеті відрізняються від чатів і месенджерів тим, що комунікація на них є асинхронною, що означає, що користувачі не очікують миттєвої відповіді на свої повідомлення.

Асинхронна комунікація дозволяє користувачам надавати детальніші та вичерпні відповіді на повідомлення.

Форуми різняться від звичайних вебсайтів тим, що контент на них створюється користувачами, а не власником сайту. Однак, для забезпечення належного функціонування, форуми повинні мати правила, які контролюють наповнення форуму.

Майже всі форуми мають адміністраторів, які контролюють якість контенту та технічний стан форуму. Адміністратори мають більше повноважень, ніж звичайні користувачі, що дозволяє їм втручатися, якщо правила форуму порушуються.

Форуми поділяються на розділи та підрозділи з різними тематиками, де користувачі можуть створювати теми та обговорювати їх шляхом публікації повідомлень. Зазвичай, ці повідомлення мають структуру з ієрархічною організацією, яка робить відповіді користувачів у форматі ланцюжків.

На форумах використовуються специфічні жаргонні вислови, які можуть змінюватися в залежності від направлення вебфоруму. Прикладом може слугувати тема з повідомленнями, яка має назву «тред», а «оффтопом» є повідомлення користувачів написані не за темою.

1.2 Особливості та види форумів

Зараз багато форумів які мають різні підходи до організації тем, повідомлень, розподілу прав доступу та функціональності. Щоб оптимізувати функціонування свого форуму, слід розглянути найпоширеніші підходи.

Вебфоруми мають декілька прав доступу. Деякі з них не вимагають реєстрації, дозволяють переглядати теми та повідомлення і навіть створювати власні гостям. Звичайно, ці форуми є анонімними і не зберігають інформацію про користувача, який залишив повідомлення на них.

Найпоширенішим видом являється форум, котрий вимагає реєстрацію перед додаванням теми, або написання відповідей, проте перегляд тем та рубрик дозволено без реєстрації. [1]-[3].

Вебфоруми доступні для більшості користувачів Інтернету, але є закриті форуми, які потребують запрошення від адміністратора. Також існують форуми з різним рівнем доступу, де деякі розділи можуть бути відкриті для всіх, а інші – тільки з дозволу адміністрації. Форуми можуть бути динамічними або статичними за способом визначення тем. На динамічних форумах теми визначаються користувачами, а на статичних – адміністрацією. Форуми можуть мати широкий спектр тем або бути фокусованими на конкретній галузі з обмеженим кількістю тем.

Textboard та Imageboard є поширеними типами вебфорумів.

Textboard - це простий вид форуму, де спілкування відбувається тільки за допомогою тексту і без необхідності реєстрації.

Imageboard, в свою чергу, дозволяє розміщувати зображення в тексті, що розширює можливості спілкування та забезпечує більш візуальну презентацію інформації.

У сучасних вебфорумах використовуються редактори тексту, які мають функціонал та можливості для додавання рубрик, тем та відповідей. Більшість таких редакторів називаються WYSIWYG-редакторами, які дозволяють формувати текст, додавати зображення, посилання, таблиці, списки та інші функції, щоб забезпечити кращу якість візуального відображення контенту на вебсторінці.

Створюваний форум буде спрямований на обговорення питань, пов'язаних з автомобільною галуззю. Адміністрація форуму визначатиме та редагуватиме розділи. Будь-хто зможе без обмежень дивитися рубрики і теми, проте для додавання нових відповідей, тем та рубрик, користувачу буде запропоновано зареєструватися [1]-[3].

1.3 Форуми з автомобільної тематики

Користувачі автомобілів постійно зіштовхуються з проблемами, пов'язаними з електронікою, обслуговуванням та доглядом за авто, а також з поломками, що можуть бути специфічними для певних марок автомобілів. Часто вони шукають вирішення цих проблем в Інтернеті за допомогою пошукових систем, або тематичних вебфорумах [4]-[5].

На вебфорумах можна знайти багато відповідей на питання, а вдосконалені технології пошуку дозволяють швидко знайти вирішення проблеми або звернутися за порадою до інших користувачів. Більшість проблем, з якими стикаються користувачі форуму, вже були вирішені, тому її вирішення не займе багато часу[4]-[5].

Однак, недоліком використання вебфорумів є те, що відповіді можуть містити помилки через те, що на форумах знаходяться користувачі не тільки з великим технічним досвідом, а й нові користувачі. Багато користувачів, особливо новачки, можуть бездумно дотримуватися порад, що може створити більше проблем у майбутньому [4]-[5].

Форуми для автомобілістів мають свої особливості, такі як можливість публікувати кілька фотографій у темі для кращого розуміння проблеми.

Розділи та теми на таких форумах визначаються та редагуються адміністрацією, і гості форуму мають змогу дивитися усі рубрики та належні теми, проте для додавання нових відповідей або тем, буде запропоновано зареєструватися.

1.4 Огляд форумів-конкурентів

Форум, присвячений автомобільній тематиці, створений для обговорення питань, пошуку допомоги вирішення проблем, збільшення знань у цій галузі. Існує багато таких форумів, які надають користувачам різноманітні можливості.

Кожен накопичує недоліки та переваги, тому перед створенням конкурентоспроможного форуму необхідно проаналізувати наявні аналоги, виявити сильні сторони, які можна використати, і покращити слабкі сторони.

Для цього можна скористатися такими сервісами, як «Vagdrive.com», «Forum.autoua.net» та «forum.uavto.com.ua».

1.4.1 «Vagdrive.com»

«Vagdrive» - один з небагатьох форумів для автомобілістів з вузькою направленістю на конкретну марку, функціонує з 2010 року. Форум має вузьку направленість, рубрики мають багатий ланцюжок основних проблем [6]. Головна сторінка має пошук, розділи, теми, та останні дії на форумі(рис. 1.1).

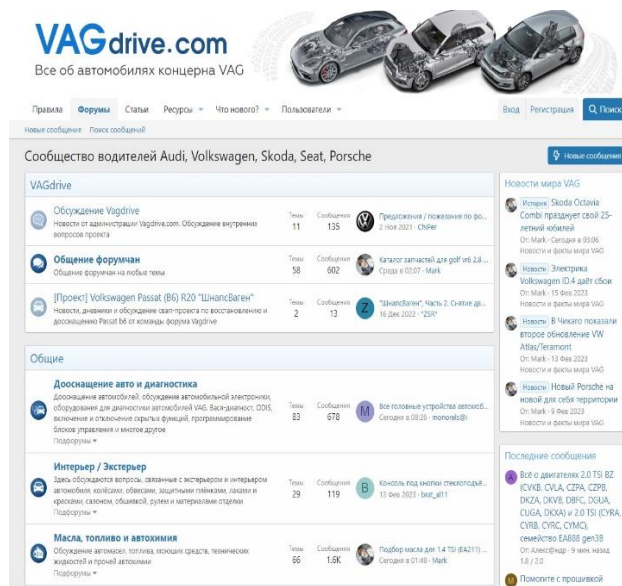


Рисунок 1.1 – Головна сторінка вебфоруму «Vagdrive»

Вигляд вебфоруму є сучасним, присутня навігаційна панель з основними посиланнями. Пошукове вікно має зручне розташування, а сам пошук працює коректно, проте для результатів необхідно перейти на додаткову сторінку. Авторизаційне та реєстраційне вікно зручне, має

додаткове підтвердження через пошту. Відсутня можливість оцінювання тем, та надання відповідям оцінок(рис. 1.2.).

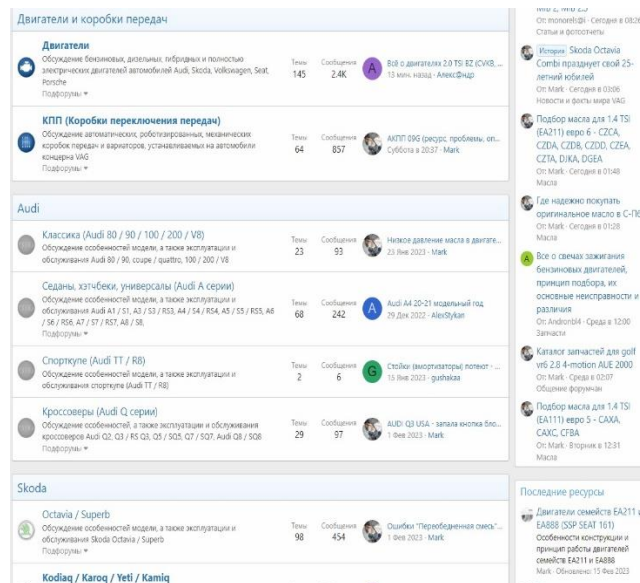


Рисунок 1.2 – Теми форуму «Vagdrive»

Для додавання відповідей, або тем до розділів, є можливість редагування тексту завдяки редактору. Відсутня можливість оцінювання тем та відповідей, а також додавання зображень до коментарів.

Вебфорум має актуальний дизайн, навігацію за сайтом реалізовано на високому рівні, проте для користування зараз – бракує функціоналу. Так як форум створено 13 років назад, він має багато обговорень популярних тем, проте на сайті замало користувачів.

1.4.2 «Forum.autoua.net»

«Forum.autoua.net» — популярний україномовний вебфорум для автомобілістів, працює з 1998 року, що дозволяє користувачам знаходити відповіді до багатьох питань.

Дизайн форуму застарілий, що може ускладнити користувачам орієнтуватися в функціоналі та навігації сайту. Використовується виключно українська мова для спілкування, без можливості зміни на інші мови (рис. 1.3).

The screenshot displays the main page of the Forum.auto.ua website. At the top, there is a navigation bar with the site logo and links for 'Запитати Автоua', 'Задать вопрос', and 'Вход / Регистрация'. Below this is a secondary navigation bar with links to 'Форум', 'Маркет', 'Товари', 'Новости', 'Базар', 'Объявления', 'Страхование on-line', and 'Еще'. The main content area is a list of forum topics, organized into sub-forums. The first sub-forum is 'Український автомобільний форум', which includes topics like 'Автоua форум (і не тільки)', 'Запитати Автоua (і не тільки)', 'Автоua.net Клуб (і не тільки)', 'Питання клубу (і не тільки)', 'Питання та відповіді на питання', 'Будів автомобіліста', 'Проблеми на дорогах (і не тільки)', 'Відомі машини відомі експерти (і не тільки)', 'Вибір автомобіля (і не тільки)', 'Обговорення технічних автомобілів', 'Автоua.net Страхування', 'Закон, страхування та кредити (і не тільки)', 'Шина та диски (і не тільки)', 'Паливо, ГММ та двигуни', 'Купівля та продаж автомобілів', 'Електрика', 'Безпечність (і не тільки)', 'Вибір автомобіля (і не тільки)', 'Марки автомобілів', 'Audi форум (і не тільки)', 'BMW форум (і не тільки)', and 'Спеціальні форуми (і не тільки)'. Each topic is listed with its title, a brief description, the number of replies, and the date of the last post. On the right side of the page, there are advertisements for Silyo and Bacardi, along with a section for 'Надмірне споживання алкоголю шкідливе для вашого здоров'я'.

Рисунок 1.3 – Головна сторінка форуму «Forum.auto.ua.net»

Форум має великий набір підфорумів, що охоплюють широкий спектр питань з механічної тематики[7].

На форумі існує система рейтингу тем і відповідей, яка в свою чергу допомагає показати найбільш корисні теми та відповіді. Пошукове поле знаходиться на панелі навігації, результати пошуку демонструються на новій вкладці. Присутня неприбрана реклама, яку неможливо відключити. Щоб додати нову тему або відповідь необхідно пройти реєстрацію, яка є зручною і не вимагає підтвердження через електронну пошту. При додаванні тем або відповідей є функціонал, який дозволяє прикріпляти зображення до відповідей, що покращує пояснення проблематики питань[7].

В цілому, форум насичений багатьма темами та популярними питаннями і відповідями. Проте застарілий дизайн, та маленька кількість активних користувачів роблять цей вебфорум непривабливим.

1.4.3 «Forum.uavto.com.ua».

«Forum.uavto.com.ua» –вебфорум для автомобілістів, працює з 2005 року, має безліч тем та рубрик. Дизайн сайту досить застарілий, що може викликати певні труднощі для користувачів у розумінні функціоналу. Крім того, на форумі присутня реклама, яку неможливо відключити[8]. При додаванні рубрик, тем та повідомлень необхідно підтвердження що ви авторизований користувач. Реєстрація без підтвердження через електронну пошту, досить зручна(рис. 1.4).

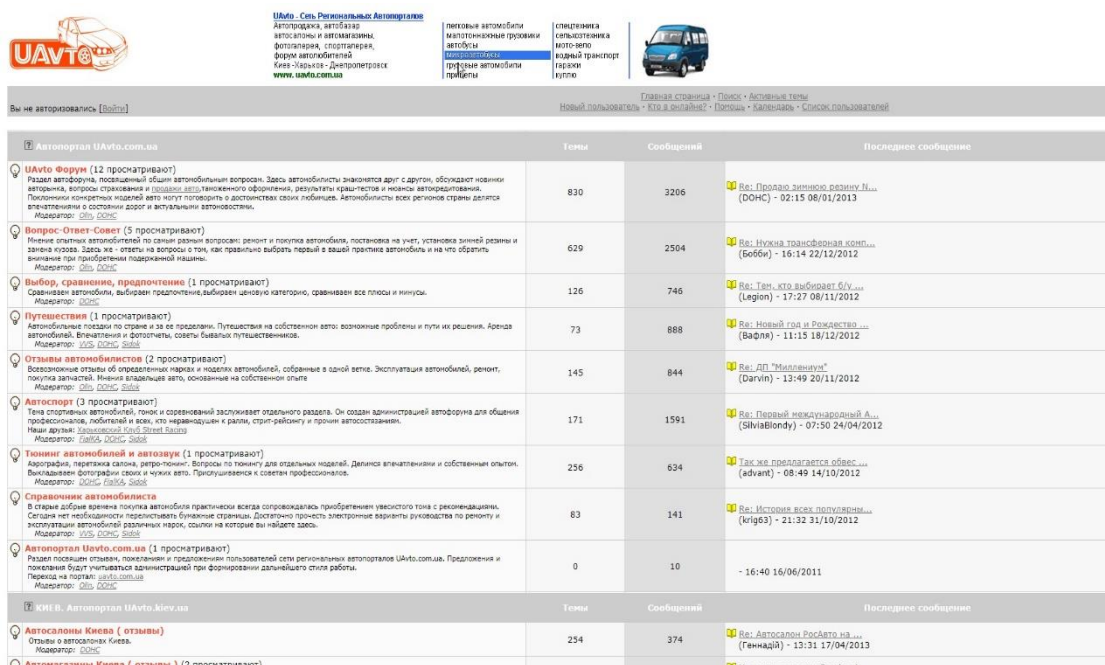


Рисунок 1.4 – Головна сторінка форуму «Forum.uavto.com.ua»

Форум надає можливість додавання зображень до відповідей, завдяки редактору тексту[8].

Створення тем та відповідей здійснюється за допомогою редактора.

Функція пошуку на форумі зручна для користування, розміщується на навігаційній панелі, але результати на іншій сторінці.

Отже, вебсайт має застарілий дизайн, не дуже зручну навігацію та брак

функціоналу. Проте через те, що форуму 20 років, він має перевагу у вигляді великого обсягу інформації, котра корисна зараз.

1.4.4 Порівняння аналогів

Щоб порівняти існуючі аналоги – виділимо такі умови:

- система рейтингу відповідей та тем;
- адаптивний дизайн;
- актуальний дизайн та зручна навігація;
- можливість додавання файлів;
- відсутність старих та проблемних рішень;

Порівняння форумів буде наведено у вигляді таблиці (табл. 1.1).

Таблиця 1.1 – Порівняння форумів

Критерій порівняння	Vagdrive.com	Forum.autoua.net	Forum.uavto.com.ua	Розроблюваний застосунок
Рейтинг відповідей та тем	Присутня	Відсутня	Відсутня	Присутня
Актуальний дизайн та навігація	Присутній	Відсутній	Відсутній	Присутній
Можливість додавання файлів	Присутній	Присутній	Присутній	Присутній
Адаптивний дизайн	Присутній	Відсутній	Відсутній	Присутній

Зараз існує безліч вебзастосунків, кожен з яких має власні особливості, переваги та недоліки функціоналу. Аналізуючи їх, можна прийти до висновку, що більша частина автомобільних форумів мають дуже застарілий дизайн, та обмежений функціонал.

Тому вебзастосунок(ВЗ), що розроблюється, має включати у себе переваги існуючих вебфорумів, та виправлені основні недоліки. Застосунок

буде мати можливість завантаження зображень, систему оцінювання тем, можливість перегляду тем без попередньої реєстрації на ресурсі вебфоруму, та розв'яже проблеми з застарілим дизайном.

1.5 Постановка завдання

Основною метою цього проєкту є розробка та створення вебзастосунку, присвяченого автомобільній тематиці. Цей форум надає можливість користувачам обмінюватися своїм досвідом та знаходити відповіді на питання, що виникають у них щодо автомобілів.

1.5.1 Межі проєкту

Найголовнішою можливістю вебфорумів являється додавання рубрик, тем та відповідей в них, саме тому головною задачею буде реалізація цього функціоналу.

Також проблемою інших форумів є застарілий дизайн, котрий у деяких випадках тільки заважає користуванню форумом.

1.5.2 Системний функціонал

Розроблюваний вебзастосунок має наступні функціональні можливості:

- зручна навігація рубриками, навігація по темам для зручного переходу між ними;
- можливість додавання нових тем, та їх видалення;
- можливість створення відповідей на теми або на відповіді інших користувачів;
- можливість редагування користувацького профілю та зміни фото профілю користувача;

- пошук по всім рубрикам для швидкого знаходження потрібної інформації;
- відображення дати створення тем для визначення їх актуальності;
- система реєстрації та авторизації на сайті для забезпечення безпеки та ідентифікації користувачів;
- можливість додавання зображень;
- розширений функціонал для адміністраторів форуму;
- можливість створення, редагування та видалення розділів та підрозділів адміністратором;

1.5.3 Інтерфейс користувача

Користувацький інтерфейс має бути зручним і легким у сприйнятті, та інтуїтивно-зрозумілим. Користувач має легко орієнтуватися на сайті і мати зрозумілі шляхи переходу між розділами.

Крім того, система повинна надавати чіткі повідомлення про помилки, але не перевантажувати користувача надлишковими повідомленнями на кожному кроці.

Перегляд тексту і візуальної інформації повинен бути зручним і не викликати незручностей для користувачів. Також повинен бути чіткий розподіл на рубрики та теми.

1.5.4 Безпека форуму

Користувачі форуму, які пройшли реєстрацію, мають доступ до додавання нових тем, написання коментарів та подальше видалення їх. Також зареєстровані користувачі мають змогу до оцінювання тем та відповідей.

Не авторизовані користувачі мають змогу лише передивлятися створені рубрики, теми та відповіді.

Доступ до додавання нових рубрик, тем, видалення усіх коментарів, видалення акаунтів користувачів, а також наповнення форуму мають тільки користувачі зі статусом адміністратора.

1.5.5 Вимоги до системи

Основні вимоги до системи для нашого вебзастосунку (ВЗ) включають:

- Процесор: Intel Core i3;
- Оперативна пам'ять: 1 ГБ;
- Жорсткий диск: Мінімум 1 ГБ, але для зберігання бази даних форуму може знадобитися більше простору при зростанні обсягу користувачів, тем та повідомлень;
- База даних: MySQL, MariaDB;
- Порт: 8001 (HTTP);
- Програмне забезпечення: JetBrains IDE, зокрема PhpStorm.

Щодо клієнтської частини ВЗ, для стабільного функціонування інтерфейсу сайту та його доопрацювання, користувачеві потрібно мати наступне апаратне та програмне забезпечення згідно мінімальних вимог популярного браузера Google Chrome[9].

Мінімальні вимоги для комп'ютерів з ОС Windows:

- Операційна система: Windows 7, 8, 8.1, 10 або новіше;
- Процесор: Intel Core i3 або новіший з підтримкою SSE 3.

Мінімальні вимоги для комп'ютерів з ОС MacOS:

- Операційна система: OS X El Capitan (10.11) або новіша;
- Комп'ютер: Macbook Pro 2011, Macbook Air 2012 або новіше.

1.6 Висновки за розділом

Перший розділ дипломної роботи був присвячений аналізу предметної області, зокрема вебфорумів, їх типів та особливостей функціонування, з фокусом на автомобільну тематику.

Після ознайомлення з існуючими рішеннями і їх аналізу, були встановлені вимоги до створюваного вебзастосунку з метою забезпечення його актуальності та конкурентоспроможності.

Також були визначені функціональні, системні та нефункціональні вимоги до вебзастосунку, що розроблюється.

2 ВИБІР ТА ОБҐРУНТУВАННЯ СИСТЕМИ, ЯКА ПРОЄКТУЄТЬСЯ

Попередній розділ був присвячений вебфоруму загалом і зокрема тематичному форуму про автомобілі та їх особливості.

Були проаналізовані переваги та недоліки наявних аналогів, на основі чого були визначені функціональні та нефункціональні вимоги до майбутнього застосунку, а також безпека та інтерфейс користувача.

Будуть розглянуті технічні питання, пов'язані з вибором мови програмування, середовища розробки та тестування майбутнього застосунку, а також будуть розглянуті основні аспекти структури цього застосунку.

2.1 Вибір застосунків для розробки

Для успішного проєктування програмного продукту необхідно ретельно обрати технології, які дозволять реалізувати потрібний функціонал без обмежень та зайвого баласту.

Так як вебфоруми включають в себе обробку та обмін даними, важливо обрати відповідну базу даних (СКБД), а також створити ефективну архітектуру застосунку та визначити методи взаємодії між його компонентами. Крім того, необхідно обрати середовище для подальшої розробки, та мову програмування застосунку.

2.1.1 Мова програмування

Зараз є безліч мов програмування для створення вебзастосунків, включаючи Python, Ruby, C#, JavaScript, PHP, та ін. Жодна з них не є найкращою загалом, а вибір мови залежить від задачі, складності та масштабу проєкту. Також важливо враховувати наявність готових рішень та фреймворків для цієї мови.

Фреймворк - це набір програмних рішень, які визначають структуру проєкту та допомагають створити складні системи.

PHP - це скриптова мова програмування, яка використовується на стороні сервера для створення динамічних вебсторінок. Ця мова серверна, тому процес генерації сторінки відбувається на сервері, а результат передається клієнту у вигляді готової вебсторінки.

Мова є об'єктно-орієнтованою, дозволяє повторно використовувати код та швидко масштабувати проєкти. Мова PHP є однією з найпопулярніших мов програмування для веб-розробки. Ось кілька коротких даних про мову PHP та її переваги та недоліки:

- PHP є серверною мовою програмування, що дозволяє генерувати динамічні вебсторінки на стороні сервера;
- вона має простий синтаксис та легко вивчається, що робить її доступною для багатьох розробників;
- PHP має велику спільноту розробників, що забезпечує наявність багатьох ресурсів, документації та підтримки;
- вона підтримує багато розширень і бібліотек, що спрощують розробку та розширення функціональності веб-додатків;
- PHP сумісна з багатьма базами даних, такими як MySQL та PostgreSQL, що дозволяє легко працювати з даними;
- PHP має широке застосування та використовується для розробки різноманітних веб-додатків, від невеликих сайтів до складних систем;
- недоліками PHP є деякі проблеми з безпекою, які виникають при неправильному використанні та недостатня ефективність в роботі з деякими типами даних;
- в порівнянні з деякими іншими мовами програмування, PHP може бути менш швидкою та масштабованою в роботі з великими проєктами[10];

Мова JavaScript є однією з найпоширеніших мов програмування для веб-розробки[11]. Ось короткий огляд даних про мову JavaScript та її переваги і недоліки:

- JavaScript є мовою програмування на стороні клієнта, що дозволяє додавати інтерактивність та динаміку до вебсторінок;
- вона має простий синтаксис, що сприяє швидкому вивченню та розвитку веб-додатків;
- JavaScript є стандартом веб-розробки і підтримується всіма сучасними браузерами, що забезпечує високу переносимість коду;
- вона дозволяє взаємодіяти з HTML-елементами, обробляти події, валідувати форми та змінювати стиль вебсторінок;
- JavaScript має широкий вибір бібліотек і фреймворків, таких як React, Angular та Vue.js, що спрощують розробку складних веб-додатків;
- вона підтримує асинхронне програмування, що дозволяє виконувати операції без блокування інтерфейсу користувача;
- JavaScript може використовуватись для розробки як фронтенду, так і бекенду веб-додатків, за допомогою платформи Node.js;
- недоліками JavaScript є деякі проблеми з безпекою, які виникають при неправильному використанні та можливість конфліктів імен при розробці великих проектів;
- завдяки виконанню на стороні клієнта, JavaScript залежить від швидкості та потужності пристрою користувача, що може вплинути на продуктивність додатків[11];

Laravel - це відкритий PHP-фреймворк, який допомагає розробникам створювати складні сайти та вебзастосунки. Він надає зручні інструменти для автентифікації, маршрутизації, роботи з сесіями, кешування та базами даних, спрощуючи розробку. Laravel використовує архітектуру MVC і допомагає визначити структуру вебзастосунку, розширюючи можливості PHP[12].

Призначення Laravel – створення вебзастосунків та сайтів на основі MVC. Фреймворк написано на PHP що і розширює його можливості.

До можливостей фреймворка можливо віднести:

- консоль Artisan дозволяє генерувати моделі, нові тести, контролери, повідомлення з командного рядка;
- Eloquent ORM технологія програмування для зв'язку бази даних та мови програмування. Використання ORM дозволяє прискорити розробку;
- шаблонізатор Blade використовується для перетворення HTML-шаблонів на готові сторінки, він не має обмежень на чистий PHP у шаблонах зручніше для розробника backend;
- система контролю версій баз даних дозволяє проводити міграції структури бази даних;
- юніт-тестування – базовий рівень проведення тестів. На працездатність тестуються окремі модулі коду;
- аутентифікація через пакети Laravel Passport та Laravel Socialite дозволяють входити через API інших програм за допомогою стандарту OAuth. Це протокол авторизації, який дає можливість авторизуватись за допомогою облікового запису іншого сайту;
- Laravel підтримує noSQL бази даних. Вони відрізняються вищою швидкістю роботи, ніж звичні БД. Інформація зберігається в оперативній пам'яті сервера, тому можна швидко отримати доступ до неї;
- механізми захисту від SQL-ін'єкцій та XSS-атак. SQL-ін'єкції не дає провести власну ORM: вона не дозволяє обробляти сторонні SQL-запити[12].

Проте до недоліків можна віднести:

- відсутність базових класів для моделей, як і для контролерів, запитів, подій. А ось моделі при генерації успадковуються від моделі з вендора, що створює труднощі з розширенням, доводиться правити ланцюжок успадкування після кодогенерації;
- між версіями великі зміни, що призводить часто до несумісності складових коду;
- міграції не збігається з ім'ям файлу, через що виникають колізії.

Щоб порівняти мови програмування, буде представлено таблицю 2.1.

Таблиця 2.1 – Порівняння розглянутих мов програмування

Критерії	PHP	JavaScript	Laravel
Виконання на клієнті	Ні	Так	Так
Зручне налагодження	Так	Ні	Так
Кросплатформовість	Так	Так	Так
Зрозуміла та повна документація	Так	Ні	Так
Готові рішення та фреймворки	Ні	Так	Так

Кожна мова має свої переваги і недоліки, тому не можна визначити найкращу.

Проте було вирішено використати мову PHP з фреймворком Laravel, для розробки проєкту, оскільки вона дозволяє значно полегшити розробку завдяки зрозумілій та докладній документації, а також великого обсягу фреймворків, які дозволяють швидко побудувати архітектуру за допомогою готових рішень.

2.1.2 База даних

Для ефективного функціонування вебфоруму потрібна швидкість оновлення даних у базі даних та швидкість їх отримання й відображення на сторінці, тому було обрано реляційний підхід SQL.

Для функціонування вебфоруму потрібний значний об'єм пам'яті для зберігання інформації, так як наш застосунок завжди додає нові дані. Після аналізу особливостей доступних БД, було вирішено використовувати реляційний підхід, а саме СКБД MariaDB з системою адміністративного керування myAdmin[13]-[14].

MariaDB запитує лише ті таблиці, які потрібні для запиту, а також надає новий тип таблиць, що зберігає дані у формі колонок замість рядків, що дозволяє покращити продуктивність.

Завдяки використанню адміністративної частини БД буде зручно отримувати потрібні дані, наприклад логіни останніх користувачів, за потреби[13]-[14].

2.1.3 Середовище розробки

Сьогодні існує багато різних середовищ та редакторів коду, які можливо використовувати для різних напрямів, зокрема у розробці вебфоруму. Далі розглянемо декілька середовищ, а саме PHPStorm та Visual Studio Code (VS Code).

Visual Studio є інтегрованою середовищем розробки (IDE), розробленим компанією Microsoft. Ось короткий огляд Visual Studio. Visual Studio надає повнофункціональне середовище для розробки програмного забезпечення, включаючи десктопні, веб- та мобільні додатки.

Воно підтримує різні мови програмування, такі як C#, C++, Visual Basic, Python, JavaScript та інші.

Visual Studio надає широкий набір інструментів для програмістів, включаючи редактор коду з функціями автодоповнення, налагоджувач, систему контролю версій та інструменти для тестування та розгортання додатків.

Хоча базова версія VS Code не містить такий широкий функціонал, як повноцінні IDE, необхідні можливості можна легко додати, встановивши потрібний модуль. Завдяки довголітній розробці та великій спільноті ентузіастів, яка створює розширення, необхідні функції можна знайти й додати дуже просто. Редактор має вбудований функціонал для налагодження

коду за допомогою точок зупинки та стеків виклику, а також інтелектуального автодоповнення.

VS Code також має зручний інструментарій для роботи з Git, що дозволяє переглядати здійснені зміни, виявляти конфлікти при злитті гілок та виконувати всі необхідні операції з Git-репозиторієм за допомогою візуального інтерфейсу[15].

Одним з недоліків VS Code є необхідність налаштування багатьох розширень для додавання функціоналу. Це може бути проблемою, оскільки розширення можуть містити помилки та працювати некоректно.

PhpStorm, що розробляється компанією JetBrains, є інтелектуальним редактором для PHP, HTML і JavaScript. Він має вбудовані засоби аналізу коду та автоматизовані засоби рефакторинга для PHP і JavaScript, а також повноцінний SQL-редактор. PhpStorm може бути розширено за рахунок установки плагінів для платформи IntelliJ IDEA[16].

Після аналізування особливостей цих двох середовищ для розробки, було обрано PhpStorm, зокрема через можливість безкоштовного використання, легкість та різноманітність розширень, що можуть бути встановлені.

2.2 Висновки за розділом

У другому розділі було проведено аналіз технічних аспектів розробки вебзастосунків.

В результаті цього аналізу було обрано фреймворк Laravel, який був обраний за його універсальність та зручність у розробці вебзастосунків.

Також, для зберігання інформації було використано СКБД MariaDB, який був обраний за швидкість пошуку даних та можливості керування і адміністрування.

У ролі середовища розробки був використаний редактор JBPHPStorm.

3 ОСНОВНІ РІШЕННЯ ЩОДО РЕАЛІЗАЦІЇ КОМПОНЕНТІВ СИСТЕМИ

Основна мета проекту - створення вебфоруму, який буде забезпечувати функції створення, пошуку та перегляду тем та повідомлень.

Для реалізації проекту необхідно проєктувати моделі даних, алгоритми обробки даних та приділяти увагу взаємодії з користувачами.

3.1 Використані моделі даних

Інформація, яку користувачі створюють, повинна бути доступною і зберігатися на сервісі в найкоротший термін. Тому важливо створити ефективну модель бази даних, яка забезпечить швидке отримання інформації.

Склад таблиць БД можна побачити на рисунку 3.1.

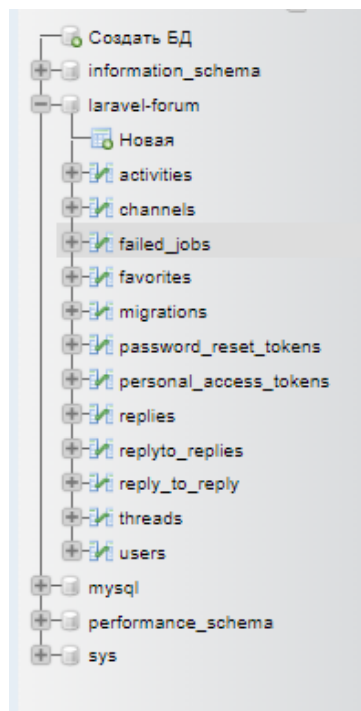


Рисунок 3.1 – База даних вебзастосунку

Застосунок, який розробляється, для зберігання даних використовується система керування реляційними базами даних MariaDB. Для

роботи з базою даних буде використовуватися PhpMyAdmin. У цій програмі будуть відображені таблиці, що складають структуру ВЗ.

MariaDB є гібридом між реляційною та нереляційною СКБД, яка пропонує високу продуктивність, масштабованість та стабільність[17]-[19].

PhpMyAdmin є вебінтерфейсом для управління базами даних MariaDB, який дозволяє адміністраторам та користувачам взаємодіяти з базою даних за допомогою веббраузера[17]-[19].

У таблиці 3.1 буде наведено призначення кожної з наступних таблиць.

Таблиця 3.1 – Таблиці у БД

Назва таблиці	Опис таблиці
activities	Зберігає активність користувачів у окремому сторінку вебзастосунку. Там зберігається кількість повідомлень, тем та рубрик відтворених користувачем.
channels	Зберігає усі створені рубрики на вебфорумі.
favorites	Зберігає усі повідомлення, на яких є позначка «кращий».
migrations	Усі міграційні процеси таблиць.
password_reset_tokens	refresh-токени зберігаються для забезпечення автоматичного оновлення авторизації користувачів після закінчення терміну дії access-токену.
personal_access_tokens	Refresh-токени користувачів зберігаються з метою прив'язки унікального ідентифікатораю.
replies	Зберігає інформацію про повідомлення користувачів у темах та рубриках.
replyto_replies	Зберігає інформацію про відповіді на повідомлення користувачів у темах та рубриках.
users	Усі користувачі які зареєстровані на вебфорумі

Продовження таблиці 3.1

Назва таблиці	Опис таблиці
reply_to_reply	Зберігає інформацію про відповіді на повідомленнях у темах.
threads	Усі створені на вебфорумі теми за рубриками

3.2 Алгоритм введення даних у застосунок

Основна мета застосунку полягає у збереженні та управлінні даними, тому критично важливо мати ефективний алгоритм обробки та перевірки валідності[17].

Для уникнення серйозних помилок, та збереження цілісності даних, перевірка виконується на клієнтській, і на серверній стороні[17]-[19].

Всього повинно бути три види користувачів: Адміністратор, зареєстрований користувач, та гість – незареєстрований користувач.

Кожен з зазначених матиме свою роль, від якої залежать правила користування форумом. Адміністратор буде мати більше прав ніж усі, а зареєстрований користувач матиме змогу додавання нових тем та відповідей. Для розуміння наведена структурна схема вебзастосунку на рисунку 3.2.



Рисунок 3.2 – Схема вебзастосунку

На клієнтській стороні валідація форм відбувається перед відправкою на серверну частину, і на момент заповнення полів.

На серверній стороні використовуються спеціальні проміжні обробники для валідації даних перед їх передачею до контролерів. Якщо потрібно, перевірку можна додатково здійснювати у самому контролері.

У якості останнього заходу перевірки правильності передачі даних виступає СКБД MariaDB, яка перевіряє вміст запитів до бази даних відповідно до правил, що були встановлені під час опису схем таблиць.

3.3 Висновки за розділом 3

У даному розділі було проведено розробку моделі бази даних, на основі якої були описані схеми для роботи з таблицями.

Окрім того, були розглянуті основні таблиці, які були створені в БД MariaDB, а також надано їх детальний опис.

4 КЕРІВНІЦТВО ПРОГРАМІСТА

4.1 Використання та призначення застосунку

Розроблено клієнт-серверний вебзастосунок, призначений для обговорення тем, пов'язаних з автомобільною тематикою та пошуку рішень проблем.

Для розгортання та налаштування серверної частини вебзастосунку потрібні мінімальні системні вимоги:

- CPU: Intel Core i5;
- RAM: 4 Gb;
- HDD/SSD: 1Gb;
- СКБД MariaDB;
- порти: 80 (HTTP), 8001 (стандартний порт MariaDB);
- програмне забезпечення: середовище PhpStorm, Laravel фреймворк.

Мінімальні системні вимоги до середовища клієнтської частини ВЗ:

- CPU: двоядерний процесор;
- RAM: 4 Gb;
- HDD/SSD: 1Gb;
- операційна система Windows 7;
- програмне забезпечення: браузер.

4.2 Характеристики програмного продукту

Розроблений вебзастосунок складається з двох основних частин: клієнтської та серверної. Клієнтська частина відповідає за створення інтерфейсу та взаємодію з користувачем, тоді як серверна частина забезпечує маніпуляції з даними в базі даних.

Клієнтська частина представлена як звичайний застосунок на декілька сторінок з використанням CSS-стилей, використано бібліотеки SweetAlert для повідомлень та спливаючих помилок, використовується вбудований у Laravel

авторизаційний застосунок. Використано Laravel фреймворк з міграційними таблицями до БД, та використанням MVC-структури, яка зображена на рисунку 4.1.

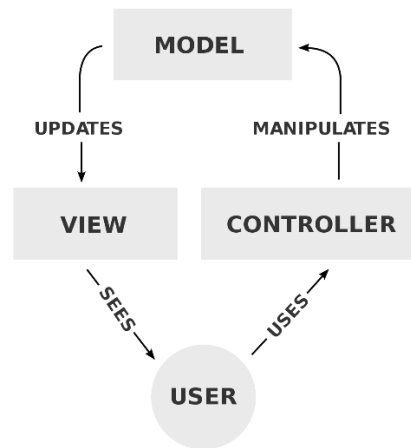


Рисунок 4.1 – MVC-структура

Ця структура дозволяє систематизувати компоненти, ресурси та інші елементи, які використовують у клієнтській додатку за MVC-структурою, що зображено на рисунку 4.2

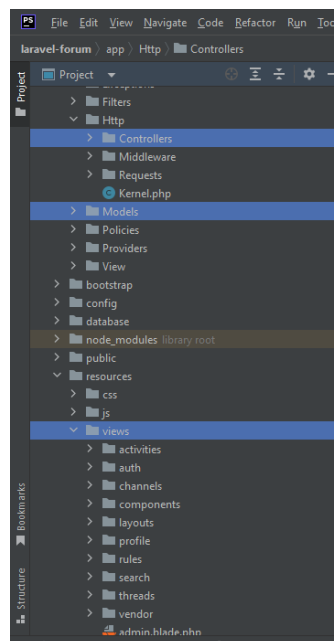


Рисунок 4.2 - Структура клієнтської частини

Серверна частина додатку функціонує завдяки основним файлам у бібліотеках з міграціями та посиланнями.

Серверна частина зображена на малюнку 4.3.

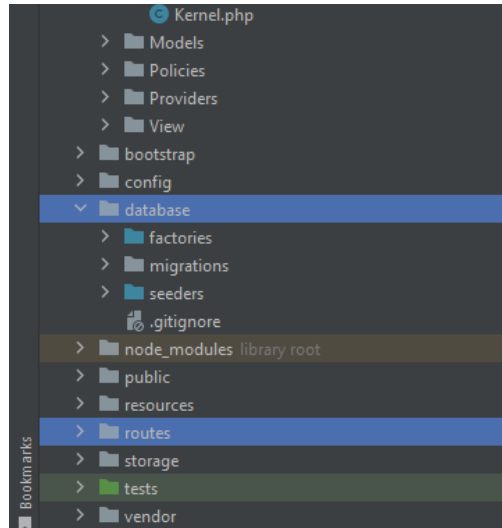


Рисунок 4.3 – Серверна структура та БД

У Laravel міграції використовуються для керування базою даних застосунку. Вони представляють собою файли, що містять визначення таблиць, індексів, зв'язків та інших структурних елементів бази даних.

Міграції дозволяють легко і контрольовано змінювати схему бази даних застосунку, реалізуючи принцип версіонування бази даних. Кожна міграція в Laravel представляє собою окрему операцію зміни бази даних, яку можна виконати та скасувати.

4.3 Вхідні та вихідні дані

Для створення тем, повідомлень та відповідей у розробленому вебзастосунку користувачі вводять текстові дані у форми. Для цього використовується WYSIWYG-редактор тексту.

На рисунку 4.4 продемонстровано додавання нової теми з використанням редактору.

 Рубрики Темі форуму

Пошук... Знайти kirill

Додати нову Тему

Рубрики:

VAG

Заголовок:

Заголовок до теми!

Опис:

File Edit View Insert Format Tools

↩ ↪ Paragraph B I U

Тут опис теми на 10+ знаків

POWERED BY TINY

Створити

forum@gmail.com +380660762002 Документація Форумні документи

Рисунок 4.4 – Створення нової теми

Результатом роботи розробленого вебзастосунку є відображення текстової інформації, яка була збережена у базі даних.

Інформація відображається на вебсторінках у вигляді рубрик, тем і повідомлень.

 Рубрики Темі форуму

Пошук... Знайти kirill

Масло для двигунів

Тут будуть вказані назви масел, які потрібно заливати у **бензинові мотори** лінійки VAG. Найпоширенішим маслами є [Motul 5100](#).

Для двигунів 1.6 MPI BFQ, BGU, BSE, BSF заводом встановлений фіксований інтервал заміни масла - раз на 15 000 км або через щороку експлуатації, залежно від того, що настане раніше. Варто зазначити, що на території Російської Федерації - фіксований інтервал заміни масла встановлений для всіх автомобілів концерну VAG. Плюс, не можна забувати про такий показник як мотогодин, для РФ термін заміни масла також прив'язується до 357 мотогодин роботи двигуна.

Двигуни 1.6 MPI BFQ, BGU, BSE, BSF потужністю 102 к.с., відповідно, не є високонавантаженими або форсованими та призначені для повсякденної та довгої експлуатації. Такі двигуни ще називають ресурсними, оскільки здебільшого вони працюють у різних конторах та автопарках, виконуючи звичайну роз'їзну роботу для співробітників цих компаній. Тобто, простіше кажучи, двигун повинен бути надійний, невибагливий і дешевий в експлуатації. Докладніше про особливості, недоліки та проблеми цих двигунів можна прочитати у довідці по моторах 1.6 MPI сімейства EA113.



Автор: Admin.

Повідомлень: 3.

Тема була створена 28-04-2023 о 18:00:58

Рисунок 4.5 – Сторінка зі створеною темою

4.4 Висновки за розділом

В даному розділі було визначено призначення розробленої системи. Був сформований список мінімальних системних вимог, необхідних для роботи та доопрацювання серверної та клієнтської частин окремо.

Описано структуру файлів вебзастосунку , продемонстровано вхідні та вихідні дані.

5 КЕРІВНИЦТВО ОПЕРАТОРА

5.1 Використання та призначення застосунку

Розроблений ВЗ, який призначений для обговорення автомобільної тематики, пошуку рішень проблем та вирішення питань, що можуть виникнути при користуванні будь-яким видом автомобільної техніки.

Для користування цим вебзастосунком клієнтам потрібен пристрій з новою версією одного з популярних веббраузерів та стабільним підключенням до Інтернету.

5.2 Використання програмного продукту

При завантаженні головної сторінки, користувач може бачити інтерфейс головної сторінки, на якій розміщено список рубрик та тем, які належать до кожної з них. Також на головній сторінці (рис. 5.1) присутній глобальний пошук, налаштування профілю користувача, останні дії користувачів, та статистика форуму за повідомленнями та створених тем.

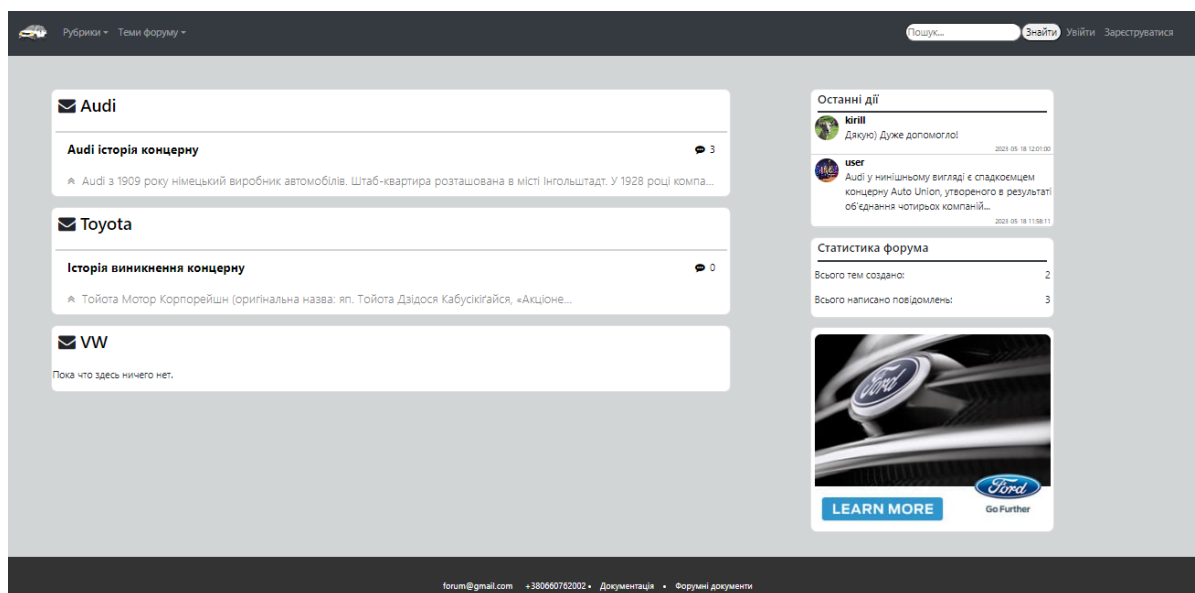


Рисунок 5.1 – Головна сторінка форуму

При натисканні на назву теми у рубриці, відбудеться перехід на сторінку з обраної теми. Щоб побачити усі теми які входять до належної рубрики, потрібно натиснути на рубрику, де буде відображено список усіх тем у рубриці, з подальшою можливістю переходу на них.

У лівій частині присутня назва рубрики, та інші теми за цією рубрикою.

У основному блоці (рис. 5.2) відображено усі теми у обраній рубриці, та короткий опис за кожною.

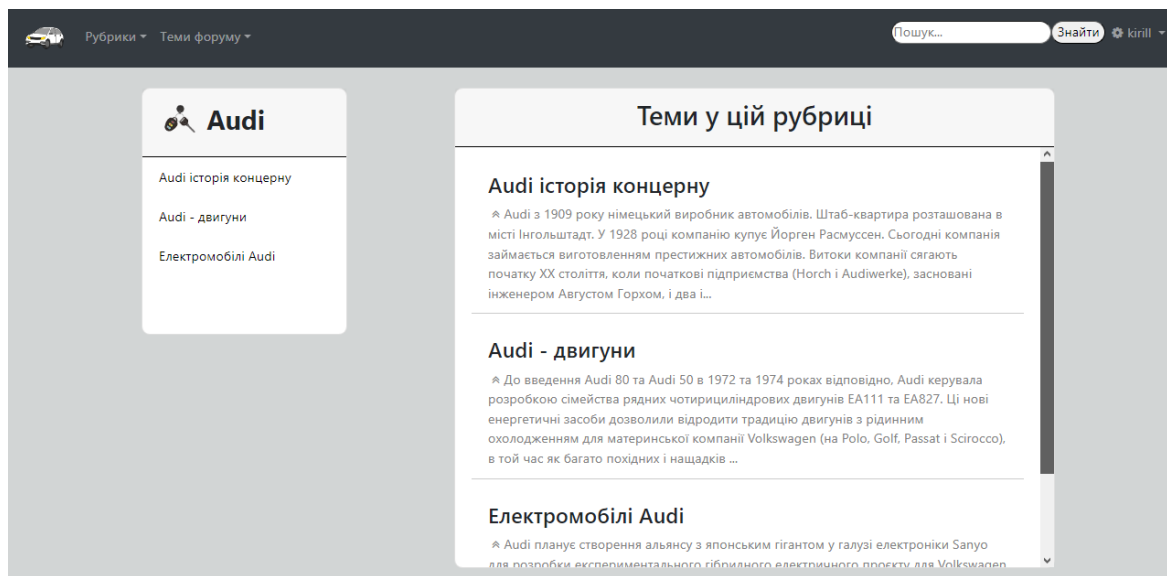


Рисунок 5.2 – Сторінка з рубриками

При відкритті теми із списку можливо побачити ідентифікатор теми, нік автора теми, дату та час додавання теми, опис, заголовки та кількість відповідей або повідомлень у темі.

Коли тема створена авторизованим користувачем, або адміністратором, то додаються до форми кнопки для її видалення. Основна форма теми зображена на малюнку 5.3.

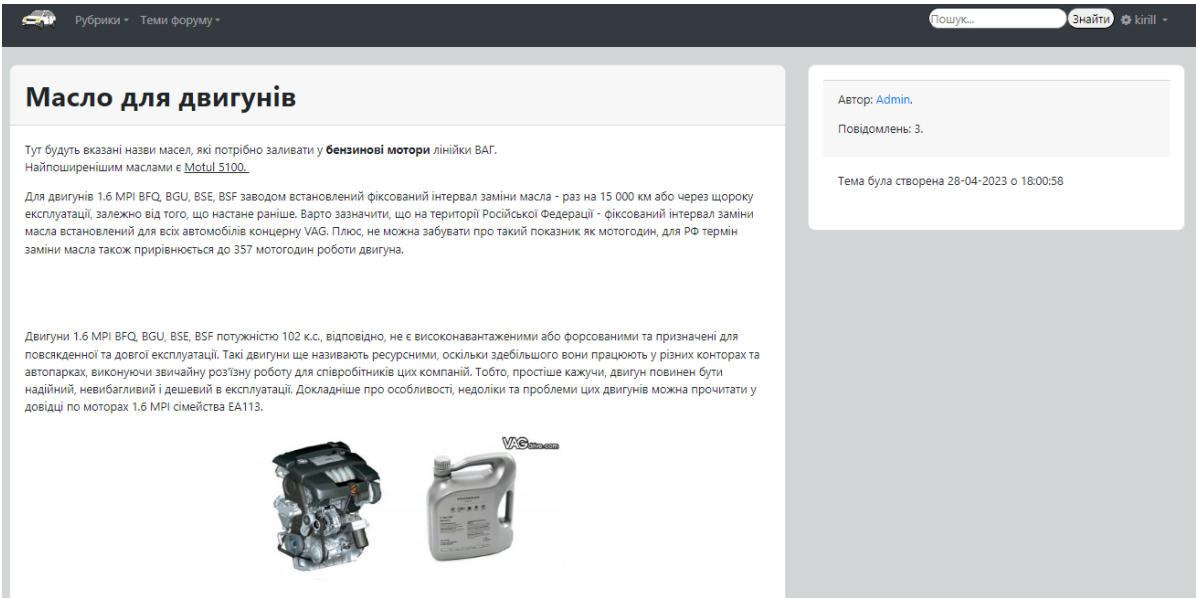


Рисунок 5.3 – Форма теми

Кожен зареєстрований користувач має можливість відповідати на тему, спілкуватися з іншими користувачами, виділяти кращий коментар або просто видалити свої коментарі (рис. 5.4).

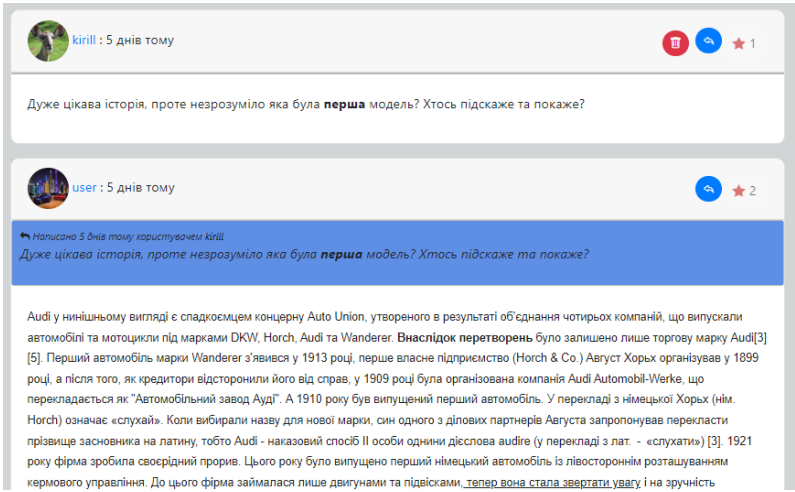


Рисунок 5.4 – Відповіді користувачів у темі

Щоб додати нову тему необхідно зареєструватися на форумі. Для цього потрібно натиснути на відповідні кнопки у панелі навігації (рис. 5.5).

Рисунок 5.5 – Реєстраційна та авторизаційна форма

Щоб додати нову тему необхідно на навігаційній панелі натиснути кнопку «додати тему» , після чого відкриється нова сторінка з додаванням теми (рис. 5.6).

Форми додавання рубрик та тем однакові, окрім того що додавання рубрики можливе тільки для користувача з правами адміністратора. Форма додавання теми має ще функціонал для додавання зображень.

Рисунок 5.6 – Додавання нової теми

Глобальний пошук (рис. 5.7) за словом по усім рубрикам та темам, або повідомленням наявний на кожній сторінці, яка відповідає за теми або рубрики. Для пошуку треба в навігаційному полі написати запит, натиснути

на кнопку та перейти на нову сторінку з результатами пошуку. Якщо результат пошуку має не один збіг, то видаються усі варіанти.

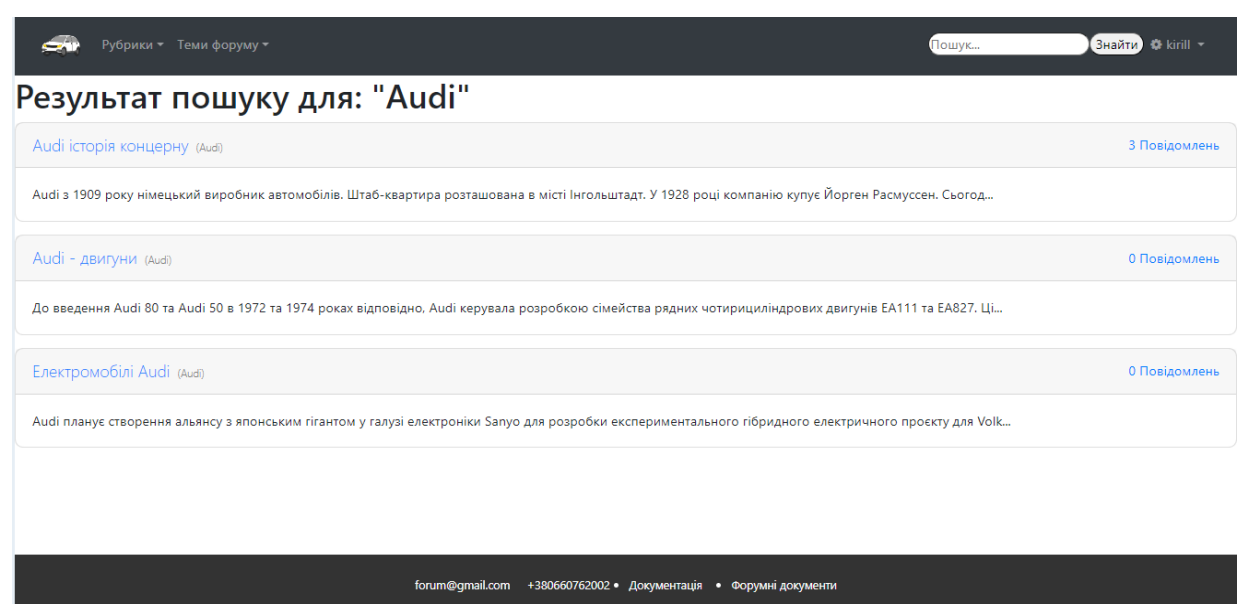


Рисунок 5.7 – Пошук за темами

Сторінка профілю користувача, де є можливість зміни імені, зображення акаунту, зміни паролю та видалення свого профілю зображено на рисунку 5.8.

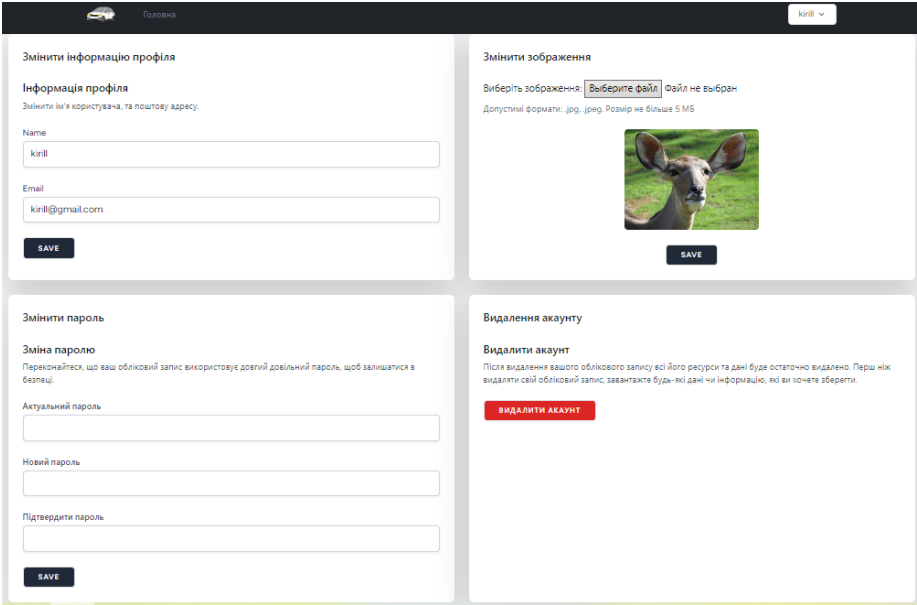


Рисунок 5.8 – Сторінка редагування профілю користувача

Сторінка перегляду активності користувача доступна при натисканні на ім'я користувача. На цій сторінці (рис. 5.9) є можливість бачити активність користувача, його створені теми, лайки на повідомлення, відповіді та коментарі.

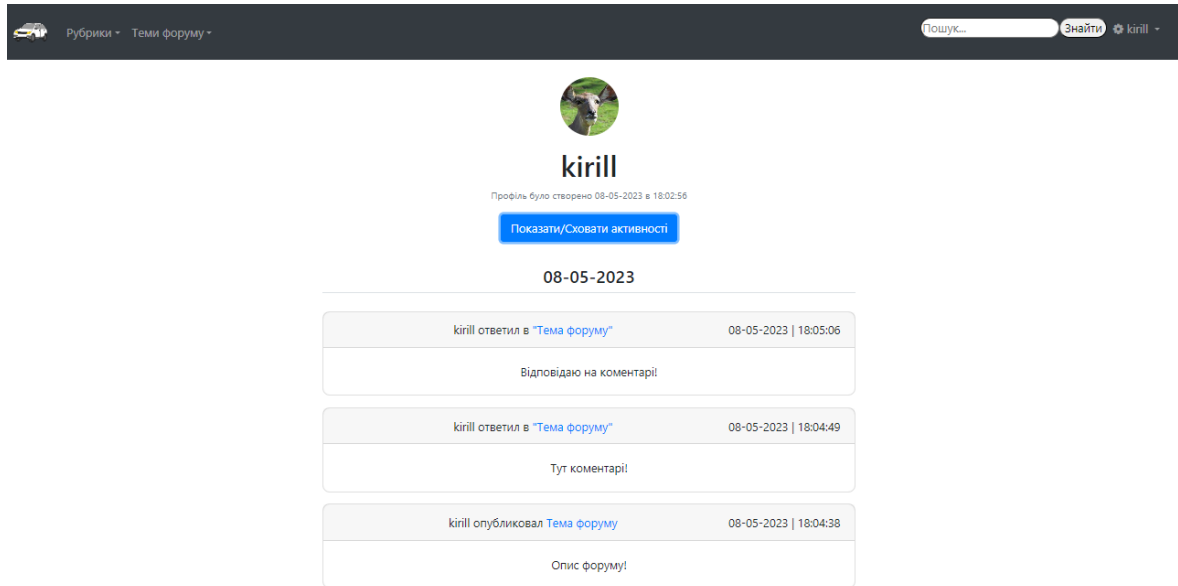


Рисунок 5.9 – Активність користувача

Якщо маєте права доступу адміністратора, то майже на усіх сторінках вебфоруму додаються додаткові можливості. Користувач з правами адміністратора має доступ до усіх рубрик та тем, можливість видалення тем, рубрик та повідомлень користувачів. Кнопки та можливості адміністратора показано на рисунку 5.10.

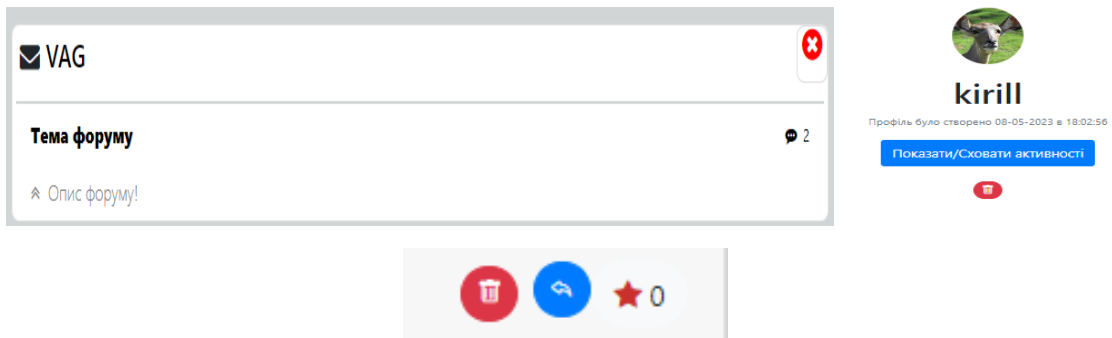


Рисунок 5.10 – Можливості адміністратора

5.3 Системні повідомлення

При різних діях як користувача так і адміністратора, ВЗ інформує про виникнення помилок, або при підтвердженні дії, яка може понести за собою негативні незмінні наслідки для форуму. Це зображено на малюнку 5.11.

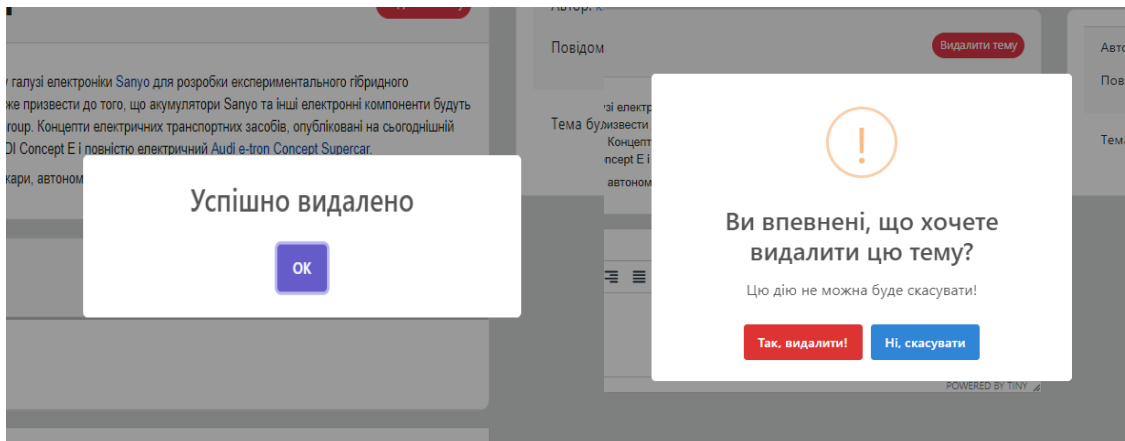


Рисунок 5.11 – Повідомлення користувачу

5.4 Висновки за розділом

В даному розділі було описано сторінки та елементи управління, які використовуються у вебзастосунку.

Були охарактеризовані взаємодії користувача та системи, наведені приклади типових повідомлень від системи.

ВИСНОВКИ

Під час реалізації дипломної кваліфікаційної роботи бакалавра було проведено аналіз сфери вебфорумів, зокрема автомобільних, з фокусом на їх типи та функціональність. Після огляду наявних рішень і їх переваг і недоліків були встановлені вимоги до розробки конкурентоздатного додатку. Також були досліджені технічні аспекти розробки, включаючи вибір мови програмування, системи керування базами даних та розроблювального середовища.

Була створена структурна схема застосунку, визначені права доступу та методи обміну між сервером та клієнтом даними. Для реалізації було обрано PHP через його універсальність та можливість розробки вебзастосунків використанням CSS стилів. В якості системи керування базами даних було обрано MariaDB з використанням phpMyAdmin для адміністрування.

Для розробки був використаний фреймворк Laravel. Редактор PhpStorm був обраний як середовище розробки. Був реалізований механізм розподілу прав доступу користувачів та методи обміну даними між клієнтом і сервером. Були використані застосунки SweetAlert та авторизаційний застосунок Laravel.auth.

В результаті був розроблений клієнт-серверний застосунок, який надає функціональність вебфоруму для обговорення автомобільних тематик і пошуку рішень проблем.

Розроблений продукт відповідає всім вимогам, що були визначені на етапі планування проєкту.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Iuniverse Inc, PHP Manual, Volume 1. // Iuniverse Inc, - 2000. - 480с..
2. CoderLessons [Електрон. ресурс]. – Режим доступу:
<https://coderlessons.com/articles/veb-razrabotka-articles/kak-sozdat-forum-s-podderzhkoi-php-mysql-s-nulia>
3. Холявкіна Т. В Створення вебфоруму при використанні сучасних технологій веброзробки. Т. В. Холявкіна, Я. О. Резаєв, О. О. Харченко. // Проблеми інформатизації та управління. – 2020.
4. Smrithi M. Understanding the usage of online forums as learning platforms/ Rekha, V. Smrithi, S. Venkatapathy. // Procedia Computer Science. – 2015.
5. LARAVEL PHP FRAMEWORK AND PROJECT CREATED USING LARAVEL // International Journal of Progressive Research in Engineering Management and Science. 2023. – Access mode:
<https://doi.org/10.58257/ijprems30976>
6. Vagdrive.com [Електрон. ресурс]. – Режим доступу:
<https://vagdrive.com/threads/pravila-foruma-vagdrive-com.1286/>
7. Forum.autoua.net [Електрон. ресурс]. – Режим доступу:
<http://forum.autoua.net/>
8. Forum.uavto.com.ua [Електрон. ресурс]. – Режим доступу:
<http://forum.uavto.com.ua/faq.php?Cat=0>
9. Браузер Chrome: Вимоги до обладнання. [Електрон. ресурс]. – Режим доступу:
<https://support.google.com/chrome/a/answer/7100626>
10. PHP Manual [Electronic resource]. – Access mode:
<https://www.php.net/manual/en/index.php> 69
11. JavaScript [Electronic resource]. – Access mode:
<https://developer.mozilla.org/en-US/docs/Web/JavaScript>

12. Laravel Beerbohm M. Laravel: Laravel for Beginners // Independently Published, 2019.
13. GOYALA. MariaDB Tutorial for Beginners // - Learn MariaDB from Scratch: Learn MariaDB Step by Step. Independently Published, 2018.
14. NoSQL vs SQL Databases [Electronic resource]. – Access mode: <https://www.mongodb.com/nosql-explained/nosql-vs-sql>
15. Visual Studio Code – Code Editing. Redefined [Electronic resource]. – Access mode: <https://code.visualstudio.com>
16. Chaudhary M., Kumar A. PhpStorm Cookbook // - Packt Publishing, 2014. 254 p.
17. Wood W. Database and Application Code. // - Migrating to MariaDB. Berkeley, CA, 2018. [Electronic resource]. – Access mode: https://doi.org/10.1007/978-1-4842-3997-1_3.
18. Vue.js The Progressive JavaScript Framework [Electronic resource]. – Access mode: <https://vuejs.org/>
19. JSON Web Token Introduction [Electronic resource]. – Access mode: <https://jwt.io/introduction>

ДОДАТОК А

Лістинги файлів роботи

A1. Файл CreateChannel

```

<div class="container py-4">
  <div class="row justify-content-md-center">
    <div class="col-md-8 col-md-offset-2">

      <div class="card mb-3">
        <div class="card-header">
          Додати нову Рубрику
        </div>

        <div class="card-body">
          <form method="POST" action="/channels">
            {{ csrf_field() }}

            <div class="form mb-3">
              <label for="title">Назва:</label>
              <input type="text" class="form-control" id="name" name="name" value="{{
old('name') }}" required>
            </div>

            <div class="form-group mb-3">
              <label for="title">Ter:</label>
              <input type="text" class="form-control" id="slug" name="slug" value="{{
old('slug') }}" required>
            </div>

            <div class="form-group mb-3">
              <button type="submit" class="btn btn-primary">Зберегти</button>
            </div>

            @if (count($errors))
              <ul class="alert alert-danger">
                @foreach ($errors->all() as $error)
                  <li>{{ $error }}</li>
                @endforeach
              </ul>
            @endif

          </form>

        </div>
      </div>
    </div>
  </div>

```

```

        </div>
    </div>
</div>
@endsection

```

A.2 Файл ShowSearch

```

<h1>Результат пошуку для: "{{ $query }}"</h1>

@if($threads->count())
    @foreach($threads as $thread)
        <div class="card mb-3">
            <div class="card-header">
                <div class="level">
                    <div class="flex lead">
                        <a href="{{ $thread->path() }}">
                            @if (auth()->check() && $thread->hasUpdatesFor(auth()-
>user()))
                                <strong>{{ $thread->title }}</strong>
                            @else
                                {{ $thread->title }}
                            @endif
                        </a>

                        @if (isset($thread->channel))
                            <span style="color: gray; font-size: small;">&nbsp;{{ $thread-
>channel->name }}</span>
                        @endif
                    </div>

                    <a href="{{ $thread->path() }}">
                        {{ $thread->replies_count }} {{ $thread->replies_count == 1 ?
'Повідомлення' : 'Повідомлень' }}
                    </a>
                </div>
            </div>

            <div class="card-body">
                {{ mb_strlen(str_replace(['&nbsp;', '&mdash;'], ' ', strip_tags($thread-
>body))) > 150 ? mb_substr(str_replace(['&nbsp;', '&mdash;'], ' ',
strip_tags($thread->body)), 0, 150) . '...' : str_replace(['&nbsp;', '&mdash;'], ' ',
strip_tags($thread->body)) }}
            </div>
        </div>
    @endforeach
@endif

```

```

        </div>
    </div>
@endforeach

    {{ $threads->appends(request()->query())->links() }}
@else
    <p>Нічого не знайдено</p>
@endif
@endsection

```

A.3 Файл CreateThread

```

<div class="container py-4">
    <div class="row justify-content-md-center">
        <div class="col-md-8 col-md-offset-2">
            <div class="card mb-3">
                <div class="card-header" style="text-align: center; font-weight: bold;
font-size: 24px;">Додати нову Тему</div>

                <div class="card-body">
                    <form method="POST" action="/threads">
                        {{ csrf_field() }}

                        <div class="form-group mb-3">
                            <label for="channel_id">Рубрики:</label>
                            <select name="channel_id" id="channel_id" class="form-control"
required>

                                <option value="">Виберіть одну з наступних...</option>

                                @foreach ($channels as $channel)
                                    <option value="{{ $channel->id }}" {{ old('channel_id') ==
$channel->id ? 'selected' : '' }}>
                                        {{ $channel->name }}
                                    </option>
                                @endforeach
                            </select>
                        </div>

                        <div class="form-group mb-3">
                            <label for="title">Заголовок:</label>
                            <input type="text" class="form-control" id="title" name="title" value="{{
old('title') }}" required>

```

```

</div>
        value="{{ old('title') }}" required>
    </div>

    <div class="form-group mb-3">
        <label for="body">Опис:</label>
        <textarea name="body" id="body" class="form-control"
rows="8" required>
            {{ old('body') }}
        </textarea>

    </div>

    <div class="form-group mb-3">
        <button type="submit" class="btn btn-
primary">Створити</button>
    </div>

    @if (count($errors))
        <ul class="alert alert-danger">
            @foreach ($errors->all() as $error)
                <li>{{ $error }}</li>
            @endforeach
        </ul>
    @endif
</form>
</div>
</div>
</div>
</div>
</div>
@endsection

```

A.4 Файл ActivityUser

```

<div class="container py-4">
    <div class="row justify-content-md-center">

        <div class="card mb-3 border-0">
            <div class="d-flex justify-content-center align-items-center mb-3">
                
</div>
<h1 class="text-center">{{ $profileUser->name }}</h1>
<small class="text-muted d-block text-center">Профіль було створено {{
$profileUser->created_at->format('d-m-Y') }} в {{ $profileUser->created_at-
>format('H:i:s') }}</small>
</div>

<div class="col-md-8 col-md-offset-2 text-center">
    <a href="#collapse-activities" class="btn btn-primary mb-3" data-
toggle="collapse" role="button" aria-expanded="false" aria-controls="collapse-
activities">
        Показати/Сховати активності
    </a>
    <div class="collapse" id="collapse-activities">
        @foreach ($activities as $date => $activity)
            <div class="pt-3 mb-4 border-bottom">
                <h4>{{ $date }}</h4>
            </div>
            @foreach ($activity as $record)
                @if (view()->exists("activities." . $record->type))
                    @include("activities." . $record->type, ['activity' => $record])
                @endif
            @endforeach
        @endforeach
    </div>

    @if(Auth::check() && Auth::user()->role == "admin")
    <div>
        <form method="POST" action="/user/{{ $profileUser->id }}">
            {{ csrf_field() }}
            {{ method_field('DELETE') }}
            <button type="submit" class="btn btn-danger btn-sm">
                <i class="bi bi-trash"></i>
            </button>
        </form>
    </div>
    @endif

</div>
</div>

```

```

</div>
@endsection

```

A.5 Файл ActivityUserShow

```

@component('activities.activity')
    @slot('heading')
        <span class="flex">
            {{ $profileUser->name }} проголосував за
            <a href="<?php echo e($activity->subject->favorited->path()); ?>">
повідомлення </a>
        </span>
        <span><?php echo e($activity->subject->created_at->format('d-m-Y | H:i:s'));
?></span>
        <?php endif; ?>

    <?php if($component): ?>
        <?php echo e(strip_tags($activity->subject->favorited->body)); ?>
    <?php endif; ?>
<?php endif; ?>

@component('activities.activity')
    @slot('heading')
        <span class="flex">

            {{ $profileUser->name }} ответил в
            @if ($activity->subject && $activity->subject->thread)
                <a href="{{ $activity->subject->thread->path() }}">"{{ $activity->subject-
>thread->title }}"</a>
            @endif

        </span>
        <span><?php echo e($activity->subject ? $activity->subject->created_at-
>format('d-m-Y | H:i:s') : 'Unknown'); ?></span>
        <?php endif; ?>

    <?php if($component): ?>
        <?php echo $activity->subject ? preg_replace('/<img[^>]+>/',
'[Зображення]',html_entity_decode($activity->subject->body)) : 'Тема видалена';
?>

```

```

<?php endif; ?> @endslot

@endcomponent

@component('activities.activity')
    @slot('heading')
        <span class="flex">
            {{ $profileUser->name }} опубликовал
            @if ($sactivity->subject)
                <a href="{{ $sactivity->subject->path() }}">{{ $sactivity->subject->title }}</a>
            @endif

            </span>
            <?php if($sactivity->subject): ?>
                <span><?php echo e($sactivity->subject->created_at->format('d-m-Y | H:i:s'));
            ?></span>
            <?php endif; ?>

        @endslot

    @slot('body')
        @if(!empty($sactivity->subject))
            {{ strip_tags($sactivity->subject->body) }}
        @endif

    @endslot
@endcomponent

```

A.6 Файл ReplyUser

```

<div class="card mb-3" style="width: auto;">
    <div class="card-header">
        <div class="level">
            <div class="flex">
                
                <a href="{{ route('user', $reply->owner) }}">
                    {{ $reply->owner->name }}
                </a> : {{ $reply->created_at->diffForHumans() }}
            </div>

            <div class="d-flex justify-content-end align-items-center">

```

```

<div class="d-flex justify-content-end align-items-center ml-3">

    @if(Auth::check() && (Auth::user()->role == "admin" || Auth::user()->id
== $reply->user_id))
        <form method="POST" action="/replies/{{ $reply->id }}" id="delete-
form">
            {{ csrf_field() }}
            {{ method_field('DELETE') }}
            <button type="submit" class="btn btn-danger btn-sm"
onclick="showSuccessMessage()">
                <i class="bi bi-trash"></i>
            </button>
        </form>
    @endif

    <button class="btn btn-primary btn-sm ml-2 reply-button"
style="border-radius: 20px;">
        data-toggle="modal" data-target="#reply-modal" data-parent-
id="{{ $reply->id }}">
            <i class="bi bi-reply"></i>
        </button>
</div>

<script>
    function showSuccessMessage() {
        // Показуємо повідомлення "Успішно видалено" за допомогою
alert або Swal.fire
        Swal.fire('Успішно видалено');
    }
</script>

<form method="POST" action="/replies/{{ $reply->id }}/favorites">
    {{ csrf_field() }}
    <button type="submit" class="btn btn-light" {{ $reply->isFavorited() ?
'disabled' : '' }}>
        <i class="bi bi-star-fill" style="color: #c9302c"></i>
        {{ $reply->favorites_count }}
    </button>
</form>

```

```

        </div>
    </div>
</div>
<div style="background-color: #d2d5d5; font-style: italic;" id="reply-{{ $reply->id
}}" class="card mb-1">
    @if($reply->reply()!="")
        <div style="background-color: #5f8fe5; padding: 10px;">
            <div style="font-size: 12px;">
                <i class="fa fa-reply"></i> Написано {{ $reply->reply()->created_at-
>diffForHumans() }} користувачем
                <a href="{{ route('user', $reply->reply()->owner) }}" style="color:
black">{{ $reply->reply()->owner->name }}</a>
            </div>

            <div style="font-style: italic;">
                {!! nl2br(html_entity_decode($reply->reply()->body)) !!}
            </div>
        </div>
    @endif
</div>

<div class="card-body" data-id={{$reply->id}}>
    {!! html_entity_decode($reply->body) !!}
</div>
</div>

```

ДОДАТОК Б
Презентація

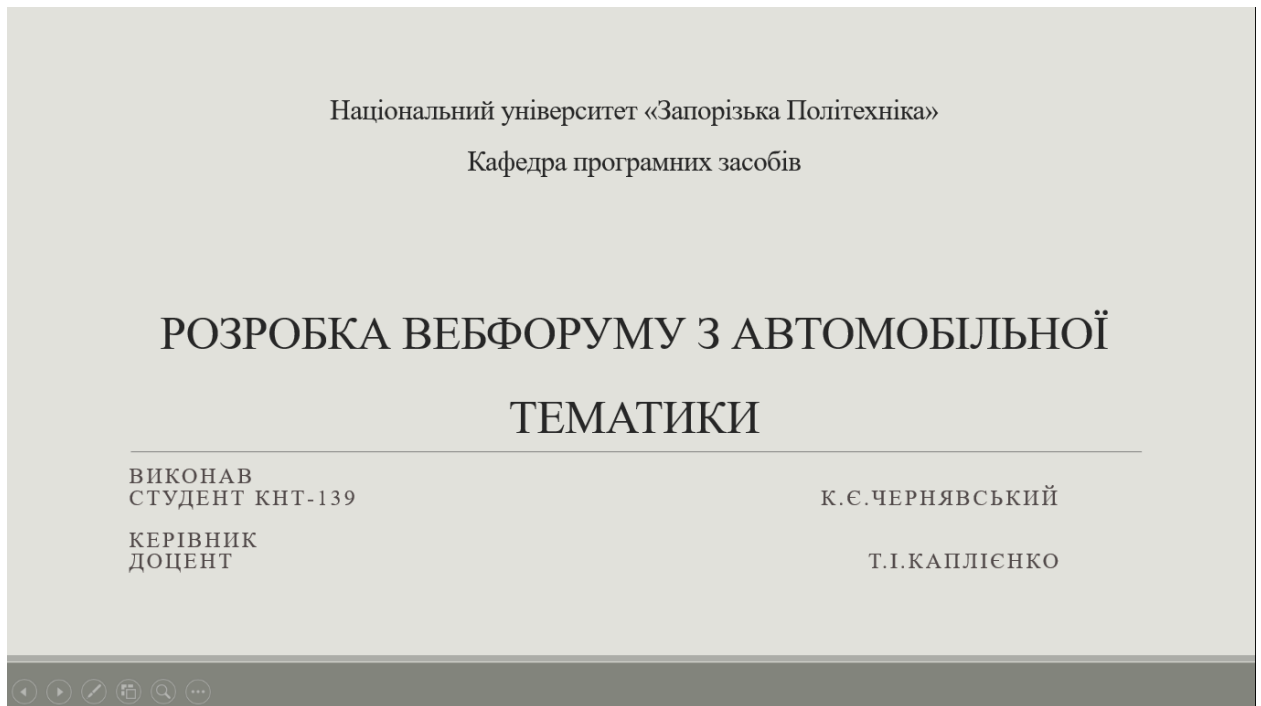


Рисунок Б.1 – Розробка вебфоруму

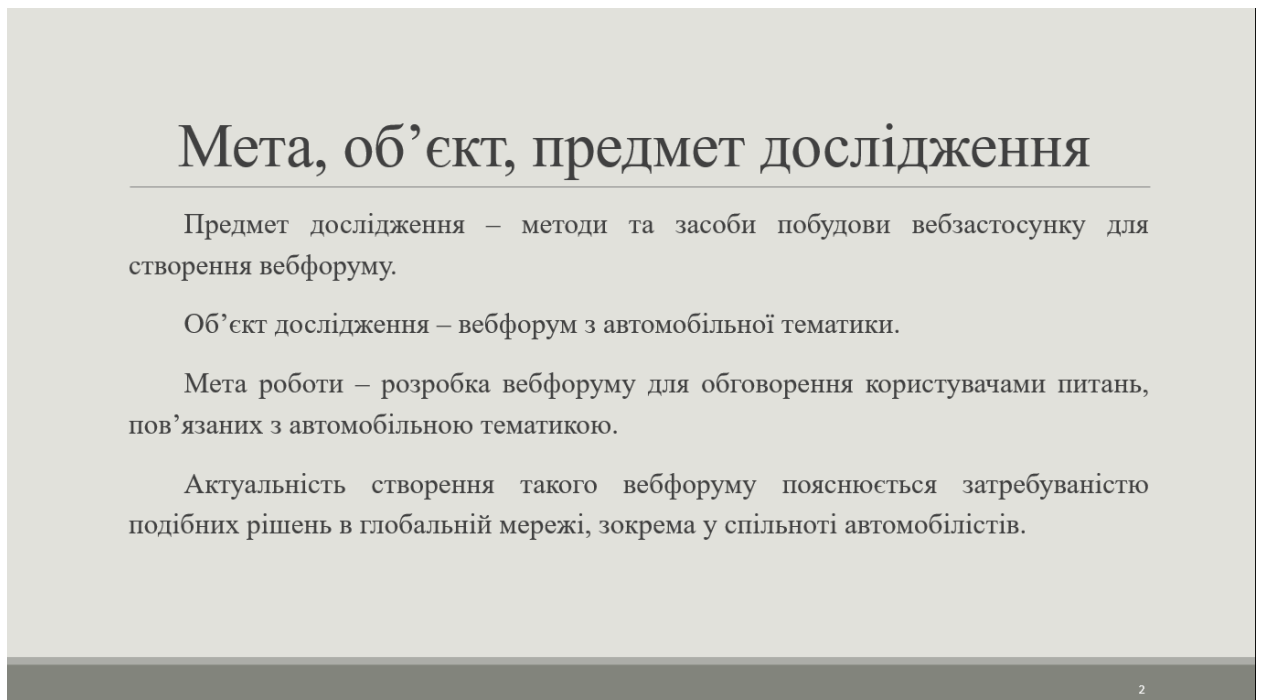


Рисунок Б.2 – Мета, предмет дослідження

Постановка завдання до роботи

Метою роботи є розробка вебфоруму на автомобільну тематику, призначеного для обговорення різних питань технічного характеру, а також пошуку рішення проблем, що трапляються при користуванні автомобільною технікою.

3

Рисунок Б.3 – Завдання до роботи

Огляд існуючих аналогів

Критерій порівняння	Vagdrive.com	Forum.autoua.net	Forum.uavto.com.ua	Вебзастосунок, що розробляється
Система оцінювання тем та повідомлень	Присутня	Відсутня	Відсутня	Присутня
Зручний дизайн та навігація	Присутній	Відсутній	Відсутній	Присутній
Функціонал обміну файлами	Присутній	Присутній	Присутній	Присутній
Адаптивний дизайн	Присутній	Відсутній	Відсутній	Присутній

4

Рисунок Б.4 – Огляд існуючих аналогів

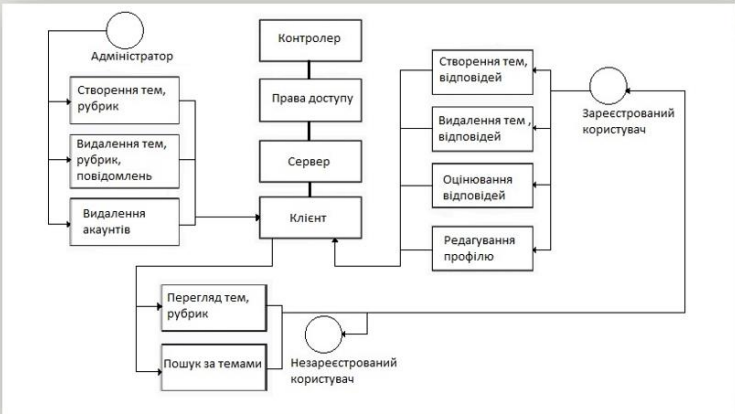
Огляд мов програмування

Критерій порівняння	PHP	JavaScript	Laravel
Можливість виконання на клієнті	Ні	Так	Так
Зручне налагодження	Так	Ні	Так
Кросплатформовість	Так	Так	Так
Зрозуміла та повна документація	Так	Ні	Так
Готові рішення та фреймворки	Ні	Так	Так

5

Рисунок Б.5 – Огляд мов програмування

Структурна схема застосунку



Всього повинно бути три види користувачів: Адміністратор, зареєстрований користувач, та гість. Кожен матиме свою роль, від якої залежать правила користування форумом.

6

Рисунок Б.6 – Схема застосунку

Структура збереження даних

Вирішено СКБД MariaDB з системою адміністративного керування myAdmin, вона запитує лише ті таблиці, які потрібні для запиту, а також надає новий тип таблиць, що зберігає дані у формі колонок замість рядків, що дозволяє покращити продуктивність.

Завдяки використанню адміністративної частини БД буде зручно отримувати потрібні дані, наприклад логіни останніх користувачів.

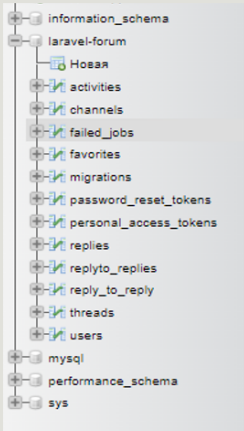


Рисунок Б.7 – Структура збереження даних

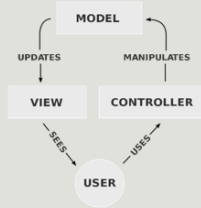
Опис таблиць у базі даних

Назва таблиці	Опис таблиці
activities	Зберігає активність користувачів у окрему сторінку вебзастосунку. Там зберігається кількість повідомлень, тем та рубрик відтворених користувачем.
channels	Зберігає усі створені рубрики на вебфорумі.
favorites	Зберігає усі повідомлення, на яких є позначка «кращий».
migrations	Усі міграційні процеси таблиць.
password_reset_tokens	Зберігає refresh-токени користувачів для поновлення авторизації при закінченні терміну життя access-токену
personal_access_tokens	Зберігає refresh-токени користувачів для присвоєння id.
replies	Зберігає інформацію про повідомлення користувачів у темах та рубриках.
replyto_replies	Зберігає інформацію про відповіді на повідомлення користувачів у темах та рубриках.
reply_to_reply	Зберігає інформацію про відповіді на повідомленнях у темах.
threads	Усі створені на вебфорумі теми за рубриками
users	Усі користувачі які зареєстровані на вебфорумі

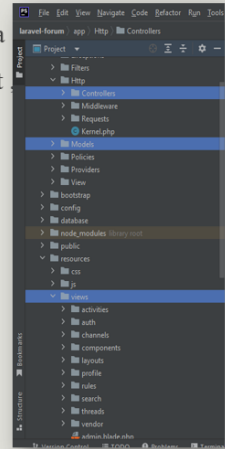
Рисунок Б.8 – Опис таблиць у базі даних

Клієнтська частина

Клієнтська частина представлена як звичайний застосунок на декілька сторінок з використанням CSS-стилей, використано бібліотеки SweetAlert вбудований у Laravel авторизаційний застосунок.



Клієнтська частина відповідає за створення інтерфейсу та взаємодію з користувачем.



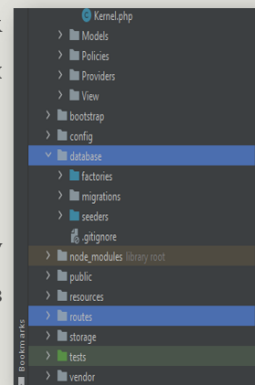
9

Рисунок Б.9 – Клієнтська частина

Серверна частина

У Laravel міграції використовуються для керування базою даних застосунку. Вони представляють собою файли, що містять визначення таблиць, індексів, зв'язків та інших структурних елементів бази даних.

Серверна частина додатку функціонує завдяки основним файлам у бібліотеках з міграціями та посиланнями, які здійснюють маніпуляції з даними в СКБД MariaDB.



10

Рисунок Б.10 – Серверна частина

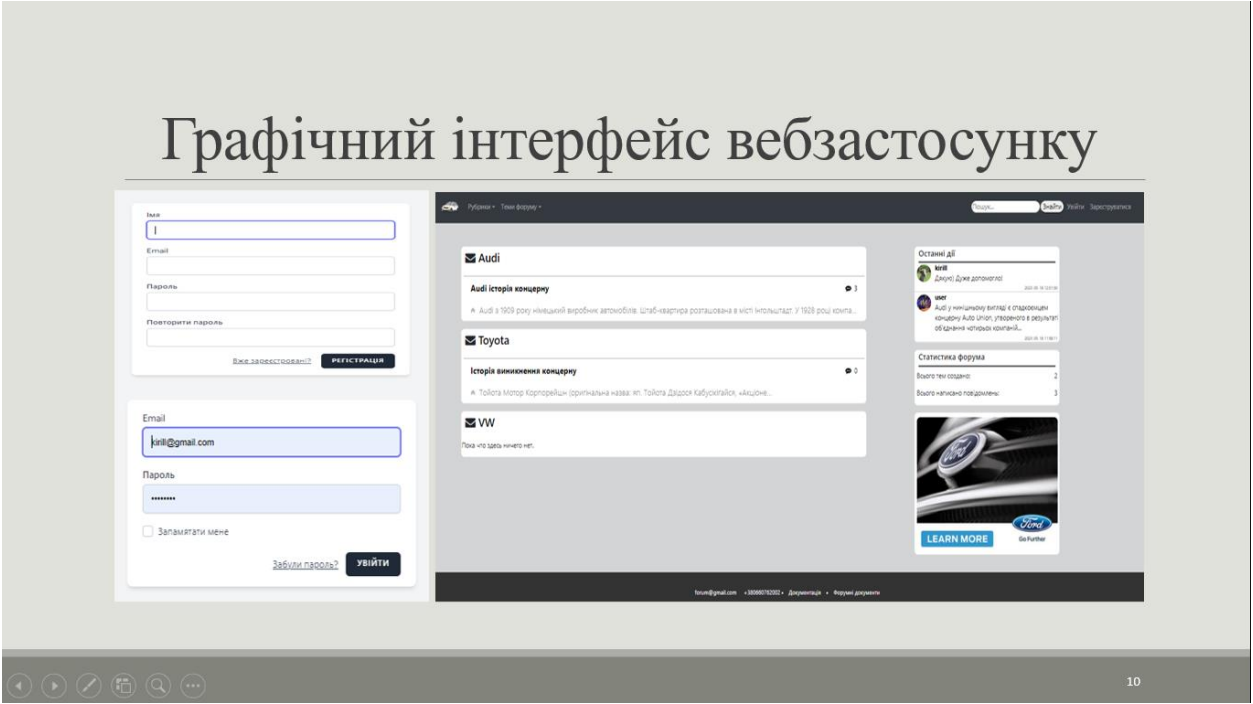


Рисунок Б.11 – Графічний інтерфейс

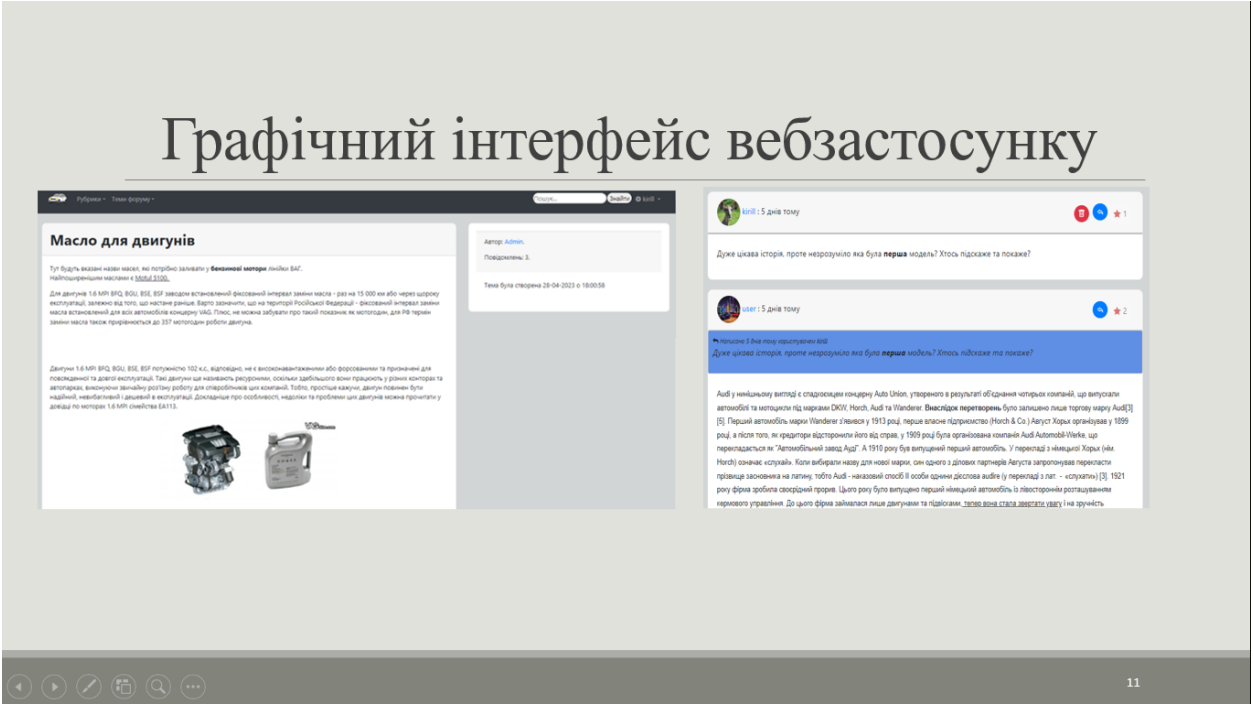


Рисунок Б.12 – Графічний інтерфейс

Графічний інтерфейс вебзастосунку

Рисунок Б.14 – Графічний інтерфейс

Висновки

Під час реалізації проєкту було проведено аналіз сфери вебфорумів, зокрема автомобільних, з фокусом на їх типи та функціональність.

Була створена структурна схема застосунку, визначені права доступу та методи обміну даними між клієнтом і сервером. В результаті був розроблений клієнт-серверний застосунок, який надає функціональність вебфоруму для обговорення автомобільних тематик і пошуку рішень проблем.

Розроблений продукт відповідає всім вимогам, що були визначені на етапі планування проєкту та описані у технічному завданні.

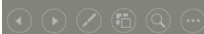


Рисунок Б.15 - Висновки