

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНИЙ АВІАЦІЙНИЙ УНІВЕРСИТЕТ
ФАКУЛЬТЕТ КІБЕРБЕЗПЕКИ, КОМП'ЮТЕРНОЇ
ТА ПРОГРАМНОЇ ІНЖЕНЕРІЇ
КАФЕДРА ПРИКЛАДНОЇ ІНФОРМАТИКИ

ДОПУСТИТИ ДО ЗАХИСТУ

Завідувач кафедри

_____ Гамаюн В.П.
(підпис) (ПІБ)

“ _____ ” _____ 2021р.

ДИПЛОМНИЙ ПРОЕКТ
(ПОЯСНЮВАЛЬНА ЗАПИСКА)

ВИПУСКНИКА ОСВІТНЬОГО СТУПЕНЯ “БАКАЛАВР”

Тема: Web-сайт тюнінг ательє_____

Виконавець: _____ Блейчик Дмитро Олександрович
(підпис) (ПІБ)

Керівник: _____ Сінько Юрій Іванович
(підпис) (ПІБ)

Нормоконтролер: _____ Боровик Володимир Миколайович
(підпис) (ПІБ)

Київ 2021

НАЦІОНАЛЬНИЙ АВІАЦІЙНИЙ УНІВЕРСИТЕТ

Факультет кібербезпеки, комп'ютерної та програмної інженерії

Кафедра прикладної інформатики

Освітній ступень Бакалавр

Спеціальність 122 “Комп'ютерні науки”

ЗАТВЕРДЖУЮ

Завідувач кафедри

_____ Гамаюн В.П.
(підпис) (ПІБ)
“ _____ ” _____ 2021р.

ЗАВДАННЯ

на виконання дипломного проекту

Блейчика Дмитра Олександровича

(прізвище, ім'я, по-батькові випускника в родовому відмінку)

1. Тема дипломного проекту: Web-сайт тюнінг ательє

затверджена додатком до наказу ректора від “ _____ ” _____ 2021 р., № _____ /ст _____

2. Термін виконання проекту: з “ _____ ” _____ 2021 р. по “ _____ ” _____ 2021 р.

3. Вихідні дані до проекту: 1. Формулювання актуальності, об'єкту, мети, задачі, новизни завдання. 2. Провести аналіз web-сайтів тюнінг ательє. 3. Етапи створення web – сайтів. 4. Обґрунтування вибору середовища розробки. 5. Розробка web – сайту. 6. Формулювання висновка.

4. Зміст пояснювальної записки: 1. Огляд сучасних технологій та принцип створення web – сайтів. 1.1. Сучасні технології для створення web – сайтів. 1.2. Етапи створення web – сайтів. 1.3. Види середовищ розробки. 1.4. Обґрунтування вибору середовища розробки PhpStorm. 2. Проектування web – сайту тюнінг ательє. 2.1. Аналіз існуючих web – сайтів тюнінг ательє. 2.2. Створення концепту web – сайту. 2.3. Дизайн web - сайту. 3. Розробка web – сайту тюнінг ательє. 3.1. Верстка web – сайту. 3.2. Розробка інтерфейсу. Текст програми.

5. Перелік обов'язкового графічного (ілюстративного) матеріалу: 1. Види структури web – сайтів. 2. Скріншоти редакторів. 3. Приклади web – сайтів

існуючих компаній. 4. Ілюстративний прототип web – сайту. 5. Макет дизайну головної сторінки web – сайту. 6. Етапи верстки.

6. Календарний план-графік

№ пор.	Завдання	Термін виконання	Відмітка про виконання
1.	Розглянути сучасних технологій та принцип створення web – сайтів	10.04.21- 15.04.21	
2.	Спроекувати web – сайт тюнінг ательє “TuninGuru”	16.04.21- 27.04.21	
3.	Розробити web – сайт тюнінг ательє “TuninGuru”	28.04.21- 18.05.21	
4.	Оформити текстові і графічні матеріали дипломного проекту	19.05.21- 29.05.21	
5.	Підготувати презентацію по дипломному проекту	30.05.21- 31.05.21	

8. Дата видачі завдання: " ____ " _____ 2021 р.

Керівник дипломного проекту _____ Сінько Ю. І.
(підпис керівника) (ПІБ.)

Завдання прийняв до виконання _____ Блейчик Д.О.
(підпис випускника) (ПІБ.)

РЕФЕРАТ

Пояснювальна записка до дипломного проекту «Web – сайт тюнінг ательє», загальним обсягом на 46 сторінок, 35 рисунків, 7 джерел літератури.

ГРАФІЧНИЙ ІНТЕРФЕЙ, ПЛАНУВАННЯ, РОЗРОБКА, СТВОРЕННЯ, ТЕСТУВАННЯ, РОЗМІЩЕННЯ, ПІДТРИМКА, ІНФОРМАЦІЙНА СИСТЕМА, ВЕБ РЕДАКТОР.

Мета дипломного проекту – розглянути існуючі технології та принципи створення web – сайтів, проаналізувати існуючі web – сторінки тюнінг ательє, розробити сучасний, зручний та привабливий web – сайт тюнінг ательє.

Завдання дипломного проектування:

- Проаналізувати існуючі web – сайти в мережі Інтернет;
- Підкреслити недоліки, які слід усунути в розроблені web – сайту;
- На основі аналізу проведених робіт розробити макет та дизайн web - сайту.

Об'єкт проектування – web - сайт тюнінг ательє в мережі Інтернет.

На основі проведеного аналізу принципів розробки web – сайтів, обрана структура та загальна схема web – сайту, що спроектовано, розглянуть аналоги існуючих web – сторінок тюнінг ательє в Інтернеті. Для забезпечення зручного функціонування web – сайту було розроблено та створенно навігаційне меню, блок порівняння зміненого та стокового зовнішнього вигляду автомобіля, розміщено карту, галерею із світлин автомобілів клієнтів.

Результати проекту можуть бути використані будь-якими компаніями зі схожою структурою позиціонування в Інтернеті, з подальшими допрацюваннями та зміною існуючого бренду.

ЗМІСТ

ПЕРЕЛІК УМОВНИХ СКОРОЧЕНЬ.....	6
ВСТУП.....	7
РОЗДІЛ 1. ОГЛЯД СУЧАСНИХ ТЕХНОЛОГІЙ ТА ПРИНЦИП СТВОРЕННЯ WEB – САЙТІВ.....	8
1.1 Сучасні технології для створення web – сайтів.....	8
1.2 Етапи створення web – сайтів.....	14
1.3 Види середовищ розробки.....	17
1.3.1 Обґрунтування вибору середовища розробки PhpStorm.....	24
Висновки до розділу.....	25
РОЗДІЛ 2. ПРОЕКТУВАННЯ WEB – САЙТУ ТЮНІНГ АТЕЛЬЄ.....	26
2.1 Аналіз існуючих web – сайтів тюнінг ательє.....	26
2.2 Створення концепту web – сайту.....	30
2.2.1 Опис вибраної структури сайту.....	30
2.3 Дизайн web - сайту.....	34
Висновки до розділу.....	35
РОЗДІЛ 3. РОЗРОБКА WEB – САЙТУ ТЮНІНГ АТЕЛЬЄ.....	36
3.1 Верстка web – сайту.....	36
3.2 Розробка інтерфейсу.....	39
Висновки до розділу.....	44
ВИСНОВКИ.....	45
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	46

Кафедра ПІ				НАУ 21 01 64 – 000 ПЗ			
<i>Виконав</i>	<i>Блейчик Д.О.</i>			Web-сайт тюнінг ательє	<i>Літ.</i>	<i>Лист</i>	<i>Листів</i>
<i>Керівник</i>	<i>Сінько Ю.І.</i>					5	46
<i>Консульт.</i>					122 ТП-415Б		
<i>Н. Контр.</i>	<i>Боровик В.М.</i>						
<i>Зав. каф.</i>	<i>Гамаюн В.П.</i>						

ПЕРЕЛІК УМОВНИХ СКОРОЧЕНЬ

HTML — мова розмітки гіпертексту (HyperText Markup Language)

CSS — каскадні таблиці стилів (Cascading Style Sheets)

CMS — система керування вмістом (Content Managenent System)

ТЗ — технічне завдання

UI — графічний інтерфейс (User Interface)

ІТ — інформаційні технології

ПЗ — програмне забезпечення

ВСТУП

Web – сайт - сукупність вебсторінок та залежного вмісту, доступних у мережі Інтернет, які об'єднані як за змістом, так і за навігацією під єдиним доменним ім'ям. Фізично сайт може розміщуватися як на одному, так і на кількох серверах. [1] Перший у світі сайт info.cern.ch з'явився в 1990 році. Його створив Тім Бернерс-Лі — батько сучасного інтернету. Автор опублікував на своєму сайті опис нової технології WWW (World Wide Web), заснований на протоколі передачі даних HTTP, системі адресації URI і мові розмітки HTML. [2]

На даний час кількість нових сайтів з великими можливостями швидко збільшується, саме тому їх потрібно зробити зручними та логічними в навігації, адже в мережі ми проводимо значну частину свого життя. Через велику конкуренцію сайти розробляються з підвищеним функціоналом та привабливим дизайном.

Актуальність: Автомобільний транспорт розвивається якісно і кількісно бурхливими темпами. Паралельно з розвитком автомобілебудування розвивається галузь технічного обслуговування автомобілів, і як вид технічного обслуговування – тюнінг автомобілів. Цей сайт розроблено для представлення тюнінг ательє в мережі Інтернет.

Новизною проекту є створення сучасного та прогресуючого web – сайту для тюнінг ательє, який може скласти конкуренцію існуючим web – сайтам в мережі Інтернет.

Практична цінність результатів: Результати проекту можуть бути використані тюнінг ательє з подальшими допрацюваннями та зміною бренду.

РОЗДІЛ 1. ОГЛЯД СУЧАСНИХ ТЕХНОЛОГІЙ ТА ПРИНЦИП СТВОРЕННЯ WEB – САЙТІВ

1.1 Сучасні технології для створення web – сайтів

Всесвітня павутина складається з веб-сторінок, які створено у форматі HTML (HyperText Markup Language, «мова розмітки гіпертексту»). HTML - це фундаментальна, базова технологія Інтернету.

HTML не є мовою програмування, це мова розмітки тексту, що використовує спеціальні оператори – теги (tag) чи інша назва дескриптори (descriptor) для розмітки текстового документа. Ці позначки вказують в якому вигляді буде виведено текстовий чи інший елемент у вікні браузера.

HTML дозволяє формувати на сторінці сайту текстові блоки, додавати до них зображення, організовувати таблиці, керувати відтворенням кольору, додавати до дизайну сайту звуковий супровід, організовувати гіперпосилання з переходом до інших розділів сервера або звертатися до інших ресурсів Інтернету і компоувати всі ці елементи між собою. Документи, що створено лише засобами HTML мають розширення .htm або .html.

Однією з основних функціональних особливостей мови HTML, завдяки якій вона і отримала свою назву, є гіперпосилання.

Гіперпосилання (Hyperlink) — це базовий функціональний елемент HTML-документу, який реалізовує зв'язок певного об'єкту веб-сторінки з іншим об'єктом. Для гіперпосилання може використовуватися як фрагмент тексту, так і графічний об'єкт, а сам гіперзв'язок можна встановлювати як між об'єктами одного сайту, так і між об'єктами, що розміщені на різних сайтах Інтернету.

HTML є мовою, що лише інтерпретується, тому, для виконання коду, його не потрібно компілювати. Інтерпретатор мови втілено в браузер, і він

Кафедра ПІ				НАУ 21 01 64 – 000 ПЗ			
<i>Виконав</i>	<i>Блейчик Д.О.</i>			Огляд сучасних технологій та принципів створення web-сайтів	<i>Літ</i>	<i>Лист</i>	<i>Листів</i>
<i>Керівник</i>	<i>Сінько Ю.І.</i>					8	46
<i>Консульт.</i>					122 ТП-415Б		
<i>Н. Контр</i>	<i>Боровик В.М.</i>						
<i>Зав. каф</i>	<i>Гамаюн В.П.</i>						

«компілює» код безпосередньо під час відкривання документа. Якщо в кодї сторінки виявлено помилку, інтерпретатор, зазвичай, не видає відповідного попередження, а просто ігнорує весь «помилковий» рядок, що може зіпсувати зовнішній вигляд завантаженої сторінки. Це є важливим для розробників, тому слід бути уважним під час складання HTML-коду і ретельно тестувати результати своєї роботи.

CSS – каскадна таблиця стилів

CSS (Cascading Style Sheets) — це технологія опису зовнішнього вигляду документа, що створено засобами HTML, XML і XHTML.

CSS використовується для завдання кольорів, шрифтів, розташування елементів сторінки тощо. До появи CSS оформлення веб-сторінок вказувалося безпосередньо в HTML-кодї сторінки. Проте, з появою CSS стало можливим принципове розділення змісту і представлення документа. За рахунок такого нововведення стало можливим легке застосування єдиного стилю оформлення для кількох сторінок сайту, а також швидка зміна цього оформлення.

Переваги:

- Застосування кількох варіантів дизайну сторінки для різних пристроїв перегляду. Наприклад, для відображення на екрані монітора - дизайн буде розраховано на велику ширину. У разі друкування документу не буде роздруковане меню сайту, а у разі перегляду у мобільному комп'ютері чи телефоні, меню буде виведено після вмісту сторінки.
- Зменшення часу завантаження сторінок сайту за рахунок перенесення правил представлення даних до окремого CSS-файлу. В цьому випадку браузер завантажує лише структуру документа і дані, що містяться на сторінці. CSS-файл з правилами представлення цих даних завантажується браузером лише один раз і зберігається в кеші браузера.
- Простота подальшої зміни дизайну. Не потрібно виправляти кожен сторінку, достатньо лише змінити кілька правил у CSS-файлі.
- Додаткові можливості оформлення. Наприклад, за допомогою CSS-правил можна застосувати обтікання певного блоку текстом або зробити

так, щоб меню фіксовано знаходилося в певному місці при перегортанні сторінки.

Недоліки:

- Різні браузеры можуть в різний спосіб інтерпретувати правила CSS, і відповідно, по різному відображати одні і ті ж фрагменти сторінки.

DHTML – динамічна мова розмітки тексту

DHTML (Dynamic HTML) – це набір засобів, які реалізують інтерактивність веб-сторінки без звертання до серверу. Тобто, певні дії відвідувача можуть спричинити зміну зовнішнього вигляду і вмісту сторінки.

DHTML побудовано на об'єктній моделі документа DOM (Document Object Model), яка розширює традиційний статичний HTML-документ. DOM забезпечує динамічний доступ до вмісту документа, його структури і стилів. В DOM кожен елемент веб-сторінки є об'єктом, який надається до змін. DOM не визначає нових тегів чи атрибутів, а лише забезпечує можливість програмного управління всіма тегами, атрибутами і каскадними листами стилів CSS.

JavaScript – мова сценаріїв

JavaScript – це мова, що призначена для написання сценаріїв для інтерактивних HTML-сторінок. Мова JavaScript не має жодного відношення до мови Java. Java розроблено фірмою SUN, а JavaScript – фірмою Netscape Communication Corporation. Первинною назвою було «LiveScript», але, з огляду на велику популярність мови Java, назву «LiveScript» з комерційних міркувань було змінено на «JavaScript».

JavaScript не призначено для створення автономних застосунків. Програмний код на JavaScript вписується безпосередньо в текст HTML-документа і інтерпретується браузером в міру завантаження цього документа. За допомогою JavaScript можна динамічно змінювати текст завантаженого HTML-документу і реагувати на події, які пов'язані з діями відвідувача або змінами стану документа чи вікна.

Важливою особливістю JavaScript є об'єктна орієнтованість. Програмісту є доступними численні об'єкти, такі, як документи, гіперпосилання, форми,

фрейми тощо. Об'єкти характеризуються описовою інформацією (властивостями) і можливими діями (методами).

PHP - мова створення сценаріїв

PHP (Personal Home Page) - мова створення сценаріїв, яка давно переросла свою назву. Перша версія PHP була створена Расмусом Лердорфом в 1994 р. і була набором інструментів для відстеження поведінки відвідувачів сайту. З часом PHP з набору інструментів перетворилася на повноцінну мову програмування, а її назву було змінено як PHP HyperText Preprocessor (препроцесор гіпертексту PHP).

PHP - це серверна мова. Конструкції PHP, що вставлено в HTML-текст, виконуються сервером при кожному відвідуванні сторінки. Результат обробки конструкцій разом із звичайним HTML-текстом передається браузеру.

Основними конкурентами PHP є ASP (Active Server Pages) від компанії Microsoft і ColdFusion від компанії Allaire.

У порівнянні з ними PHP має ряд переваг, зокрема:

- Висока продуктивність. PHP-програми працюють швидше, ніж ASP.
- Функціональність. Розробку PHP-програми можна відокремити від власне розробки веб-сторінки, що спрощує працю і для програміста і для дизайнера.
- Ціна. Мова PHP є абсолютно безкоштовною.
- Простота у використанні. Програмісти, що мають досвід програмування на поширених мовах швидко зрозуміють синтаксис PHP.
- Переносимість. Один і той же PHP-код можна використовувати як в середовищі Windows, так і на платформах UNIX.

ASP - активні сторінки сервера

ASP (Active Server Pages) —технологія, що є аналогічною до JavaScript і PHP. Для того, щоб зробити інтерактивну веб-сторінку із застосуванням макромови ASP, необхідно вбудувати в її код відповідний скрипт, що віддалено нагадує Java і C#. Скрипт інтерпретується і виконується безпосередньо на сервері, після чого до браузера відправляється вже готовий HTML-документ з результатами роботи сценарію ASP. Для сторінок, що

містять конструкції ASP, не має значення, яке програмне забезпечення встановлено на комп'ютері користувача, але принципове значення має тип сервера, який має підтримувати дану технологію.

XML - розширена мова розмітки.

Основну увагу в мові XML зосереджено на даних. Тут, структурна розмітка даних і представлення даних є строго розділеними.

Основні причини створення XML:

- Спроба надати могутні засоби форматування і структуризації даних для широкого кола розробників.
- Необхідність в стабільній реалізації мови структуризації документів, де можна легко створювати допоміжні інструменти, що є доступними для звичайних користувачів.

XML є метамовою - спеціальною мовою, якою можна скласти повний опис класу інших мов, якими створено документи. XML містить набір правил, що дозволяють створювати унікальні застосування і підмножини даних.

В багатьох розробників виникають певні труднощі у зв'язку з абстрактністю XML і довільним використанням його методів. Насправді, XML - є досить логічною і добре організованою структурою, вона має чіткий синтаксис, що змушує строго дотримуватися певних правил. Починати вивчення XML слід із застосування вже отриманих знань про HTML. XML, як і HTML, використовує теги та атрибути.

Як мова розмітки веб-документів XML має свої особливості:

- Гнучкість. XML дозволяє обробляти унікальні дані, і незалежно від їх характеру надає адекватні методи для їх зберігання і обробки.
- Можливість налаштування. Гнучкість XML безпосередньо пов'язана з можливістю визначати власні дескриптори, необхідність в яких виникає в процесі рішення задачі.
- Узгодженість. XML відрізняється синтаксичною цілісністю і строгою структурою.

Практично всі сучасні браузері підтримують XML. Вона здатна цілком замінити HTML, як засіб розмітки веб-сторінок, хоча вивчення XML є складнішим за вивчення HTML.

XSLT - розширена мова перетворення листів стилів.

Мова XSLT (eXtensible Stylesheet Language Transformations) є транслятором, за допомогою якого можна вільно модифікувати початковий текст. XSLT грає вирішальну роль в затвердженні XML як універсальної мови збереження і передачі даних. Область застосування XSLT є широкою - від електронної комерції до безпроводного Веб.

Фактична збірка результуючого документа відбувається, коли початковий документ і лист стилів XSLT передаються до синтаксичного аналізатору XSLT (XSLT-процесора).

При використанні XSLT в середовищі Веб синтаксичний аналіз може відбуватися або з боку користувача (в браузері), або з боку сервера.

Перетворення XSLT засновано на шаблонах. XSLT-процесор аналізує початковий документ і намагається знайти відповідний XSL-шаблон. Якщо такий шаблон знайдено, то виконуються інструкції, що містяться всередині нього.

Ajax - технологія для взаємодії з сервером без перевантаження сторінок

Ajax (Asynchronous Javascript And XML) це підхід до створення веб-застосувань за допомогою наступних технологій:

- Стандартизоване представлення засобами XHTML і CSS.
- Динамічне відображення і взаємодія з користувачем за допомогою DOM.
- Обмін і обробка даних у вигляді XML и XSLT.
- Широке застосування мови сценаріїв JavaScript.
- Асинхронні запити за допомогою об'єкту XMLHttpRequest.

В стандартному веб-застосуванні обробкою всієї інформації займається сервер, браузер відповідає лише за взаємодію з користувачем, передачу запитів і виведення отриманих даних у форматі HTML.

В Ajax-застосуванні між користувачем і сервером з'являється ще один посередник – програмний механізм (рушій) Ajax. Він визначає, які запити можна обробити з боку клієнта, а які необхідно виконувати на сервері.

Поведінка сервера теж змінилася. Якщо раніше на кожен запит сервер видавав нову сторінку, то тепер він надсилає лише ті дані, які потрібні клієнту, а рушій Ajax в браузері формує з них HTML-код.

Асинхронність виявляється в тому, що далеко не кожен запит користувача скеровується до сервера, причому зворотне теж справедливо - далеко не кожна реакція сервера обумовлена запитом користувача. Велику частину запитів формує рушій Ajax, причому є можливим, щоб він передбачав запити користувача.

Зрозуміло, що за такою схемою роботи міняється якісне навантаження на сервер - якщо раніше запитів було мало, але кожен з них вимагав значних ресурсів (серверу потрібно витягнути інформацію з бази даних, сформувати з неї веб-сторінку і відправити до браузера), то тепер завдання сервера спрощується (формувати веб-сторінки не потрібно, та і об'єм даних, що передаються є значно меншим), але доводиться більше обробляти запитів.

Створювати застосування на Ajax є трудомістким і складним завданням. Потрібно написати і відлагодити на JavaScript рушій з десятих чи двадцятьох тисяч рядків коду плюс реалізувати серверну частину. [3]

1.2 Етапи створення web – сайтів

Веб-сайти створюються у кілька етапів:

1. Планування — визначення тематики і призначення майбутнього сайту.
2. Розробка — розробка структури сайту, добір матеріалів, вибір програмних засобів для його створення.
3. Створення окремих сторінок відповідно до структури, включення до них гіперпосилань.
4. Тестування — перевірка та редагування веб-сайту.
5. Розміщення — розміщення сайту в Інтернеті.
6. Підтримка — оновлення вмісту сайту. [4]

Планування:

На етапі планування, перш за все, слід визначити призначення майбутнього сайту: це буде персональний сайт або сайт організації, електронна енциклопедія чи сайт бібліотеки, сайт для дистанційного навчання тощо. Тут же доцільно визначити, буде сайт тематичним чи різні його сторінки будуть присвячені різним темам і яким саме.

Розробка:

Наступний крок — розробка структури сайту. При розробці структури сайту слід визначитися з необхідною кількістю сторінок та встановити зв'язки між ними. Розрізняють лінійну, ієрархічну та довільну структури сайту (рис. 1.2.1).

Лінійну структуру веб-сайту доцільно використовувати у разі послідовного представлення інформації, наприклад, про товари та послуги або матеріали навчального посібника. Перегляд таких сайтів здійснюється послідовно: від початкової (головної) до останньої сторінки. Кожна сторінка має посилання тільки на одну, наступну сторінку сайту. Інколи, для зручності навігації по сайту до сторінки також додається посилання на попередню сторінку.

При ієрархічній структурі створюється одна сторінка (головна), яка не має попередніх, решта сторінок мають лише одну попередню сторінку. При ієрархічній структурі кожна сторінка може містити посилання на довільну кількість сторінок сайту. Така структура найкраще підходить для сайтів, що містять різну за тематикою інформацію: каталогів, збірань статей з різних тем або добірок послань.

Найчастіше для створення сайтів використовують довільну структуру. При такій структурі сайту його сторінки пов'язані між собою довільним чином. У сайтах довільної структури можна виділити фрагменти, які є лінійними або ієрархічними.

Прикладом довільної побудови структури сайту може бути Інтернет-енциклопедія Вікіпедія. Цей проект стартував у 2001 році. Нині статті до цієї енциклопедії пишуться 250 мовами народів світу. Характерною особливістю Вікіпедії є те, що її статті відкриті для доповнення і змін будь-яким користувачем.

У Вікіпедії на даний час розміщено більше 6 млн. статей. Орієнтовна кількість авторів та редакторів — 50 тис. В україномовному розділі енциклопедії налічується 124 тис. статей, працюють 15 тис. авторів та редакторів.

Вікіпедія знаходиться в десятці найбільш відвідуваних веб-ресурсів світу за кількістю запитів сторінок щомісяця. Так кожний 200-й запит в Інтернеті направляється у Вікіпедію. Вікіпедія складається з окремих розділів. Працюючи з матеріалами цієї енциклопедії, ви можете здійснювати різноманітні переходи та самостійно визначати послідовність перегляду окремих сторінок.

На етапі розробки необхідно також здійснити добір матеріалів і вибрати програмні засоби, за допомогою яких будуть розроблятися веб-сторінки.

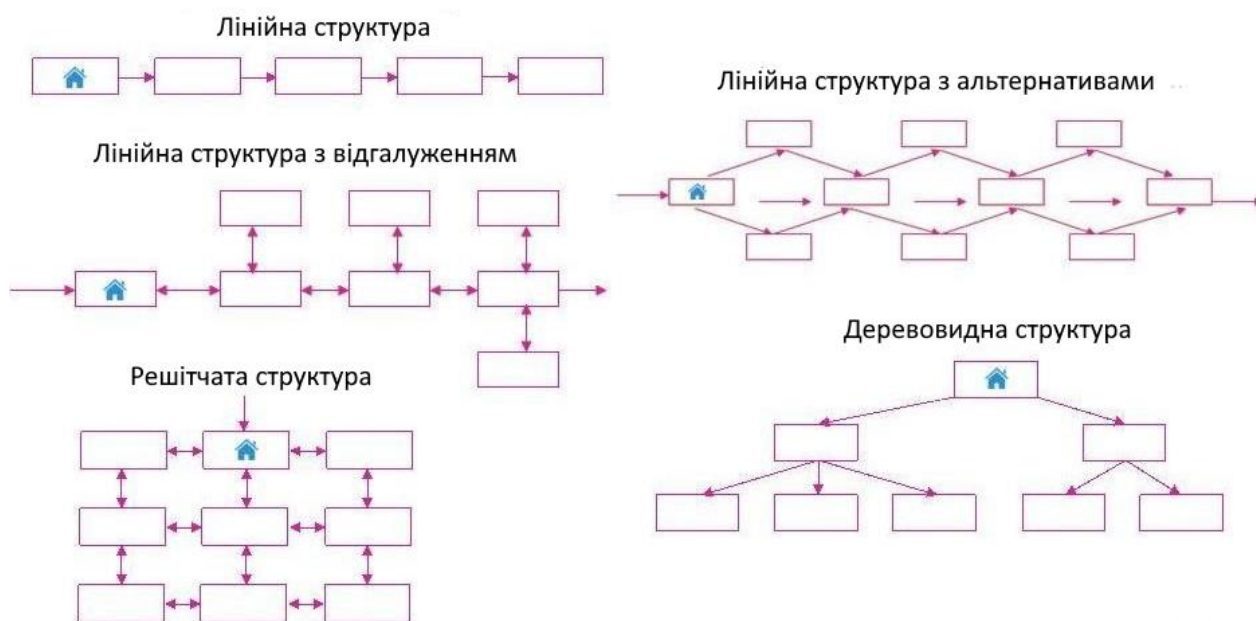


Рис. 1.2.1 Види структури web – сайту

Створення:

На етапі створення відбувається наповнення веб-сторінок конкретними матеріалами, а також створення гіперпосилань для зручної навігації сайтом. При цьому потрібно слідувати правилам оформлення (дизайну) веб-сторінок, якими ви користувалися при створенні презентацій.

Тестування:

Дуже важливим при створенні сайту є етап тестування. На цьому етапі потрібно перевірити:

- чи правильно працюють усі гіперпосилання;
- чи зручною є навігація сайту;
- чи відкриваються при відкритті сторінок графічні зображення;
- чи зручно розташовані для сприйняття матеріали на сторінках тощо.

У разі необхідності, потрібно внести зміни в наповнення або структуру сайту, змінити гіперпосилання.

Розміщення:

На наступному етапі потрібно визначити, де буде розміщено створений вами сайт. Веб-сайт можна розмістити:

- на власному сервері установи;
- на сервері вашого провайдера;
- на сервері організації, яка спеціалізується у наданні послуг розміщення сайтів користувачам Інтернету;
- на сервері, який надає послуги вільного і безкоштовного розміщення сайтів.

Сьогодні окремі організації в мережі Інтернет пропонують користувачам безкоштовні послуги щодо створення веб-сайтів з використанням готових шаблонів та одночасним їх розміщенням на серверах цієї організації.

Після розміщення сайту в Інтернеті потрібно здійснювати його підтримку, щоб сайт не втрачав своєї популярності. Ця підтримка полягає в періодичному оновленні та доповненні існуючих матеріалів, створенні нових цікавих сторінок тощо.

1.3 Види середовищ розробки

Текстові HTML редактори - тип редакторів орієнтований безпосередньо на роботу з текстом (кодом). Щоб використовувати такий редактор, вам потрібні будуть знання мови HTML. У процесі використання такого редактора ви не зможете постійно спостерігати готову реалізацію сторінки сайту.

Однак, застосування текстового редактора дає розробнику значно більше свободи, можливостей оптимізації коду та ін.[5]

Visual Studio Code:

Випущений компанією Microsoft на основі коду Atom, Visual Studio Code має частину функціоналу IDE (Integrated development environment) - інтегрованого середовища розробки - потужної програми, що містить, крім текстового редактора коду, ще ряд механізмів, що дозволяють проводити аналіз коду і його запуск. Часто саме цей інструмент уявляють, коли говорять про те, який функціонал повинна мати IDE для web розробки. У багатьох рейтингах безкоштовних HTML редакторів саме Visual Studio Code (рис. 1.3.1) займає перше місце, та використовується розробниками все частіше, і частіше. Так, наприклад, за даними Stack Overflow, цей редактор в 2017 році використали 24% веб-розробників, а в 2018 році - вже 38,7.

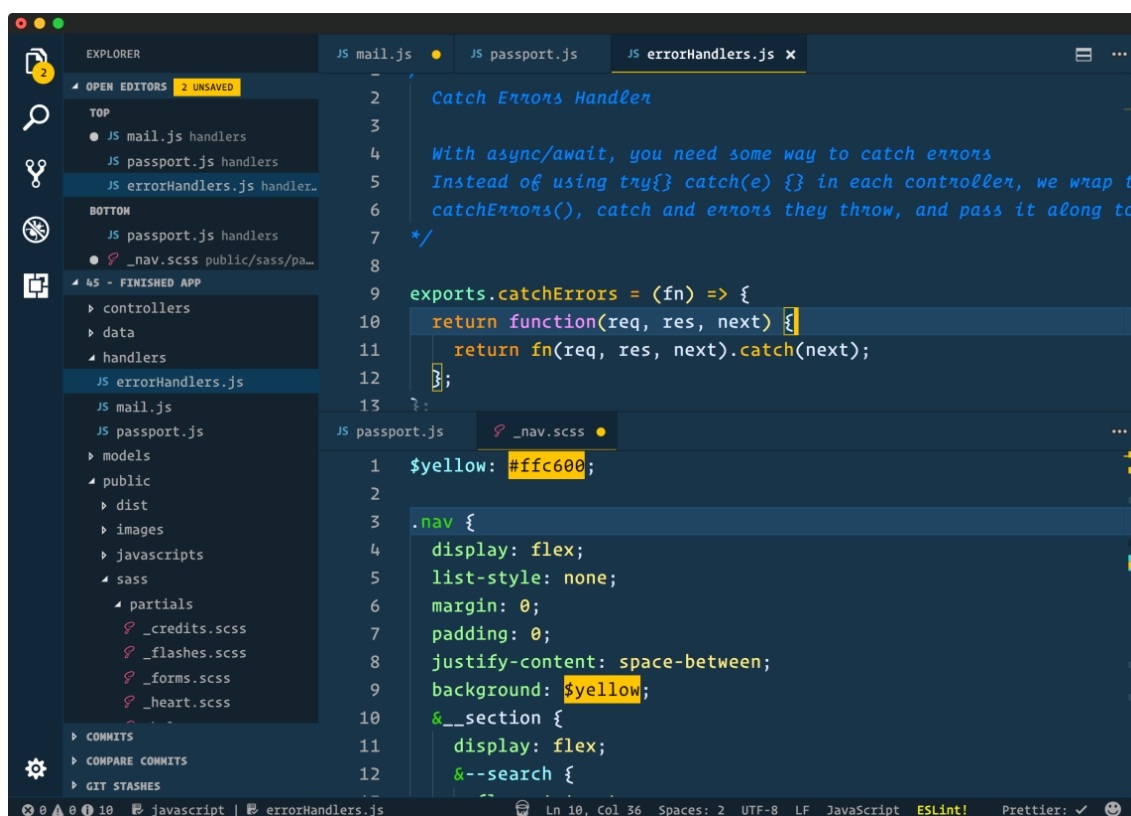


Рис. 1.3.1 Скріншот редактора Visual Studio Code

Основні переваги Visual Studio Code:

- Має значну частину функціоналу IDE;
- Вбудований потужний механізм автозаповнення - IntelliSense;
- Значна кількість розширень і доповнень;
- Інтегрований з Git «з коробки»;

- Є вбудований налагоджувач для коду JavaScript, TypeScript, Node.js
- Відкритий вихідний код програми;
- Visual Studio Code поширюється безкоштовно.

Недоліки Visual Studio Code:

- 3 мінусів розробники відзначають досить великий час запуску програми;
- Пошук за проектами здійснюється відносно повільно.

Notepad ++:

Notepad ++ - легкий текстовий редактор, розроблений для комп'ютерів під управлінням Windows. Користувачі Linux також можуть використовувати його через Wine. Notepad ++, випущений ще в 2003 році, є перевіреним і укоріненим інструментом багатьох розробників, будучи зручним текстовим редактором для HTML коду. Цей редактор поширюється як безкоштовне програмне забезпечення і його репозиторій доступний в GitHub. Notepad ++ (рис. 1.3.2) підтримує сторонні плагіни.

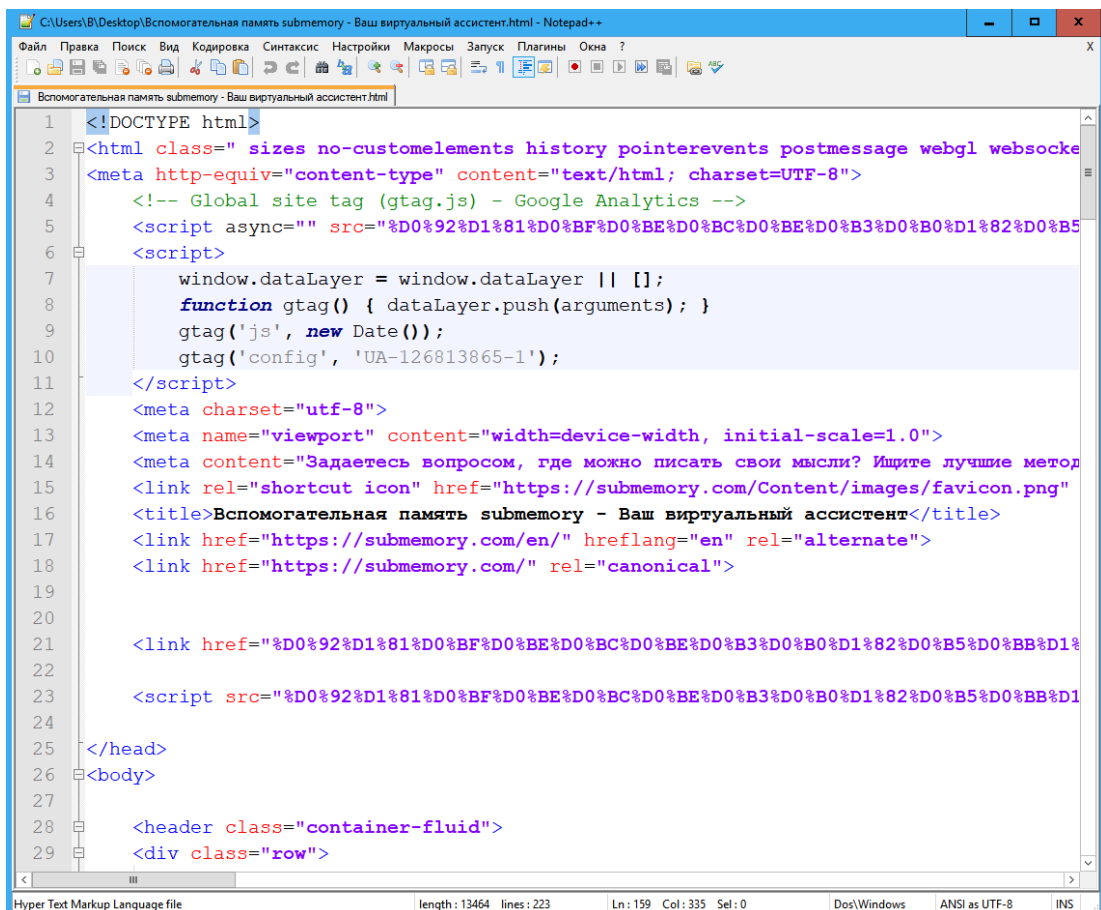


Рис. 1.3.2 Скріншот редактора Notepad ++

Основні переваги Notepad ++:

- Notepad ++ є простим, чи не вимогливим до ресурсів інструментом;
- Є портативна версія;
- Функціонал програми легко розширюється безліччю плагінів. При бажанні такий плагін можна створити самому;
- Інтерфейс програми також легко налаштовується;
- Підтримується робота з великою кількістю вкладок одночасно;
- Notepad ++ є на 100% безкоштовною програмою.

Недоліки Notepad ++:

- Можна відзначити деяку мінімалістичний інтерфейсу, яка не підходить ряду користувачів;
- Цей редактор не є IDE і не несе в собі її додатковий функціонал. З цієї причини багатьом користувачам доводиться використовувати якусь середу розробки на додаток до редактора Notepad ++.

Sublime Text:

Ще одним прикладом відмінного текстового редактора для HTML є Sublime(рис. 1.3.3). Ця програма поставляється в безкоштовному вигляді з деякими обмеженнями. Іншими словами - ви можете використовувати Sublime безкоштовно, але вам доведеться купити ліцензію, якщо ви захочете користуватися всіма функціями цього редактора.

Sublime пропонує відмінну підтримку, забезпечуючи постійний вихід актуальних оновлень. Користувачі можуть додавати плагіни, створені спільнотою, або створювати свої власні. Для значної частини розробників використання безкоштовної версії Sublime буде цілком достатнім. Якщо ж вам знадобиться більше можливостей, ви зможете купити ліцензію пізніше.

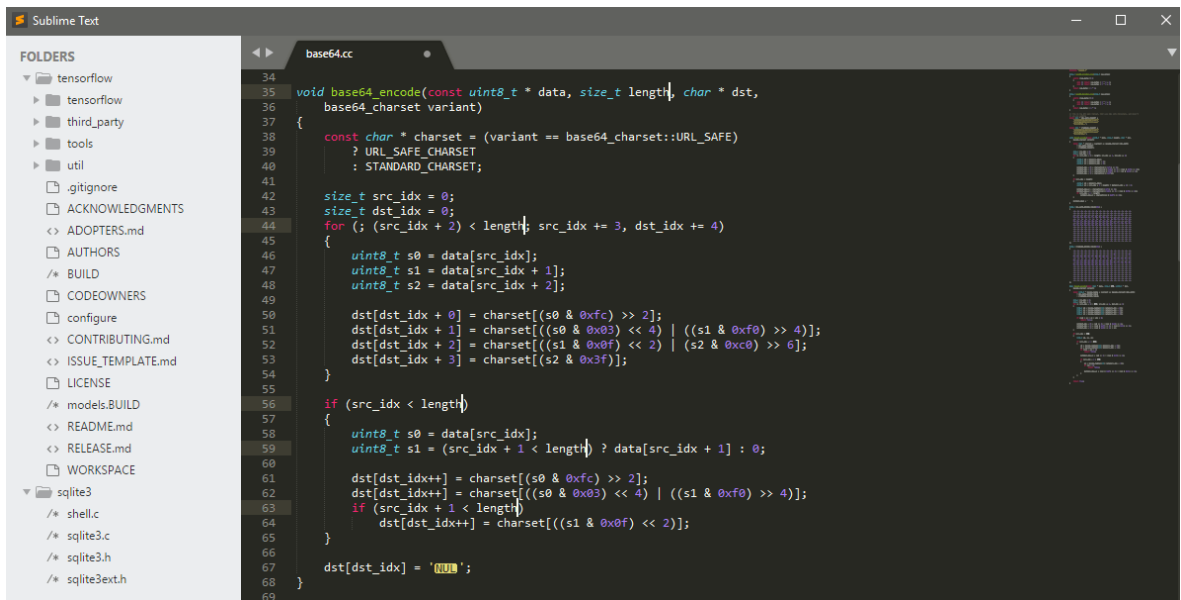


Рис. 1.3.3 Скріншот редактора Sublime Text

Основні переваги Sublime:

- Кросплатформеність. Sublime працює в таких операційних системах як Windows, OS X і Linux;
- Sublime є легким інструментом, що не перевантажує систему;
- Є портативна версія;
- Sublime надає тисячі різних доповнень з відкритим вихідним кодом, які створені великоюю активною спільнотою;
- Роздільне редагування. Розробники можуть використовувати кілька моніторів і редагувати різні ділянки коду одночасно.

Недоліки Sublime:

- Не весь функціонал доступний користувачеві безкоштовно;
- Рядом користувачів відзначається незручність роботи з менеджером плагінів;
- Ряд плагінів сторонніх розробників може працювати некоректно.

Webstorm на базі IntelliJ:

WebStorm - зручна для web розробки середовище розробки. WebStorm (рис. 1.3.4) була розроблена компанією JetBrains на основі іншого їхнього продукту - IDE IntelliJ.

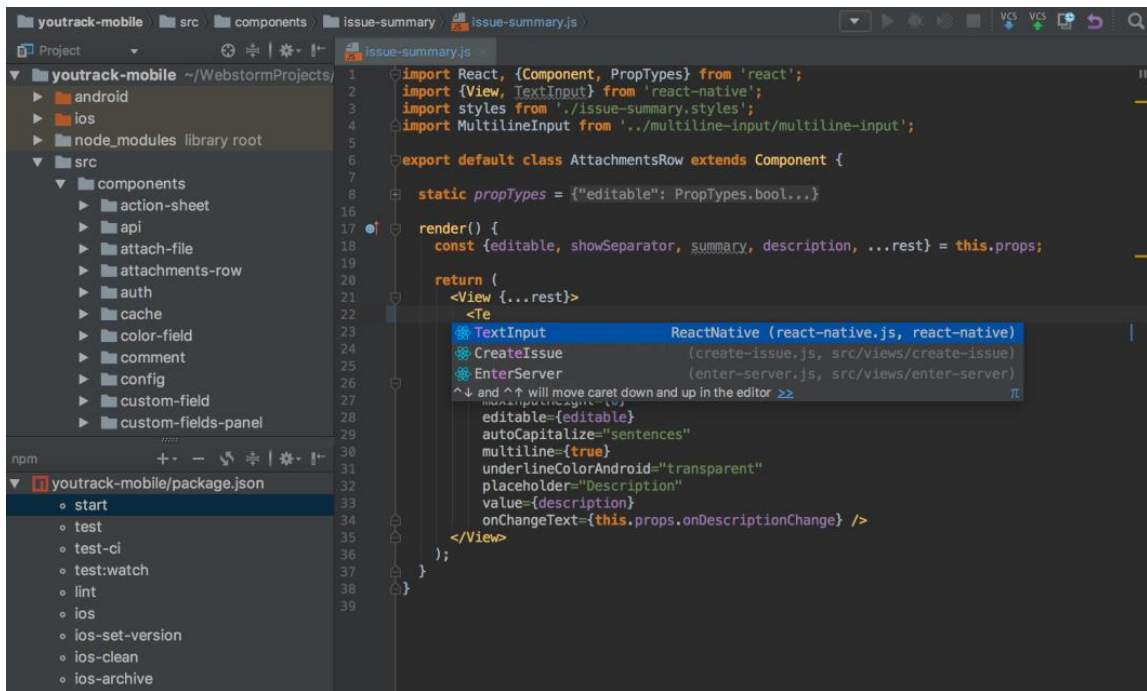


Рис. 1.3.4 Скріншот редактора Webshorm

Основні переваги WebStorm:

- Зручне автодоповнення як коду на HTML, CSS, так і на JavaScript;
- Перевірка на наявність помилок і зручна налагодження коду забезпечується за допомогою інтеграції з рядом систем стеження за вадами;
- Вбудована інтеграція з такими системами управління версіями як GitHub, Git, а також Subversion, Perforce і Mercurial;
- Гнучкість налаштувань;
- Досить велика кількість плагінів.

Недоліки WebStorm:

- Властива всім IDE повільність в роботі і вимогливість до ресурсів;
- Відносно складні налаштування;
- Платна IDE, яка поширюється на підписці.

Vim:

Vim (скорочення від - Vi Improved) це потужний портативний текстовий редактор з дуже багатою історією - йому вже більше 27 років. Володіє багатим функціоналом, з можливістю глибокого налаштування програми під себе. В оригінальному вигляді працює у вікні консолі. Можна використовувати версію

з графічним віконним інтерфейсом - Gvim. Варто відзначити, що багато сучасних IDE, для поліпшення процесу розробки, містять в собі емулятор функціональності Vim (рис.1.3.5).

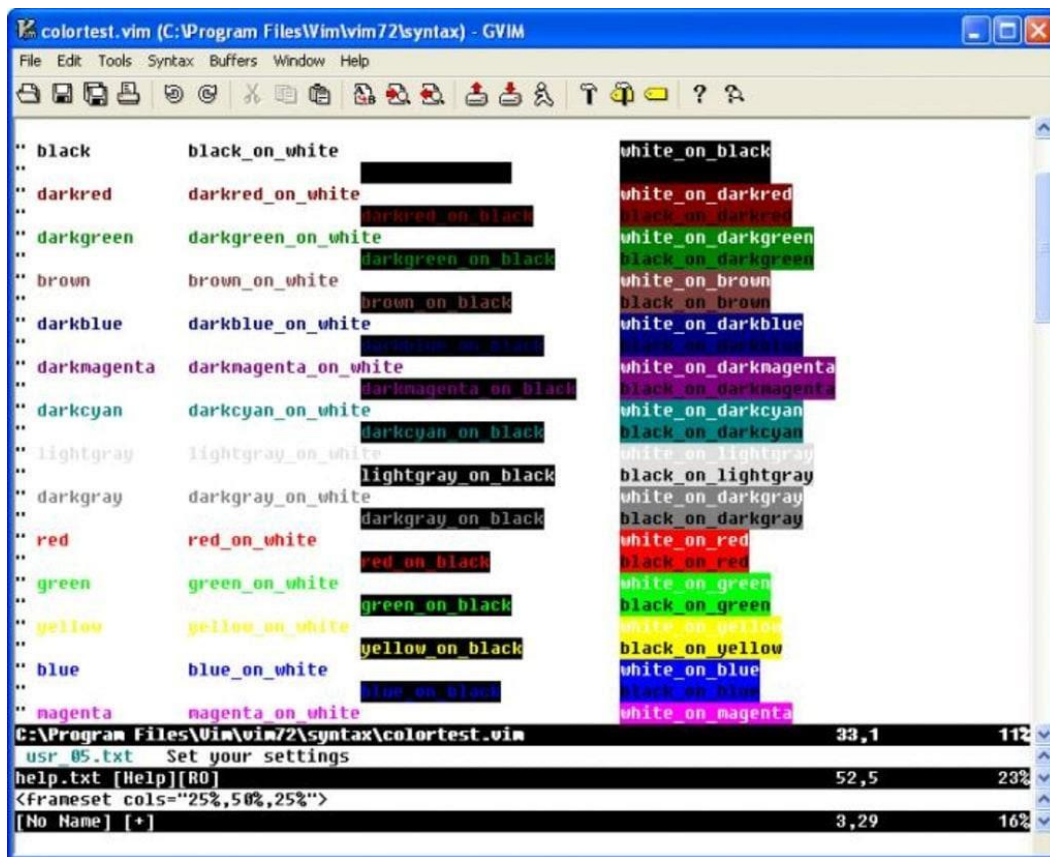


Рис. 1.3.5 Скріншот редактора Vim

Основні переваги Vim:

- Повноцінна робота в безлічі операційних систем - Windows, Linux, Amiga, Mac OS X, Unix, OpenVMS, OS / 2;
- Глибока настройка роботи редактора під себе;
- Дуже низькі вимоги до ресурсів. І, відповідно - висока швидкість роботи;
- Можливість редагування або перегляду файлу на віддаленому сервері через термінал;
- Більш 14000 доступних пакетів розширень.

Недоліки Vim:

- Один з найскладніших для вивчення інструментів розробки. Високий поріг входження вимагає від користувача значних витрат часу на запам'ятовування його особливостей, команд, плагінів і тд.

1.3.1. Обґрунтування вибору середовища розробки PhpStorm

JetBrains PhpStorm — комерційне крос-платформове інтегроване середовище розробки для PHP, яке розробляється компанією JetBrains на основі платформи IntelliJ IDEA.

PhpStorm(рис. 1.3.1.1) являє собою інтелектуальний редактор для PHP, HTML, CSS і JavaScript з можливостями аналізу коду “на льоту”, запобігання помилок у програмному коді і автоматизованими засобами рефакторинга для PHP, HTML і JavaScript. Автодоповнення коду в PhpStorm підтримує генератори, співпрограми, простори імен, замикання, типажі і синтаксис коротких масивів. Присутній повноцінний SQL-редактор з можливістю редагування отриманих результатів запитів.

PhpStorm розроблений на основі платформи IntelliJ IDEA, написаної на Java. Користувачі можуть розширити функціональність середовища розробки за рахунок установки плагінів, розроблених для платформи IntelliJ, або написавши власні плагіни. [6]

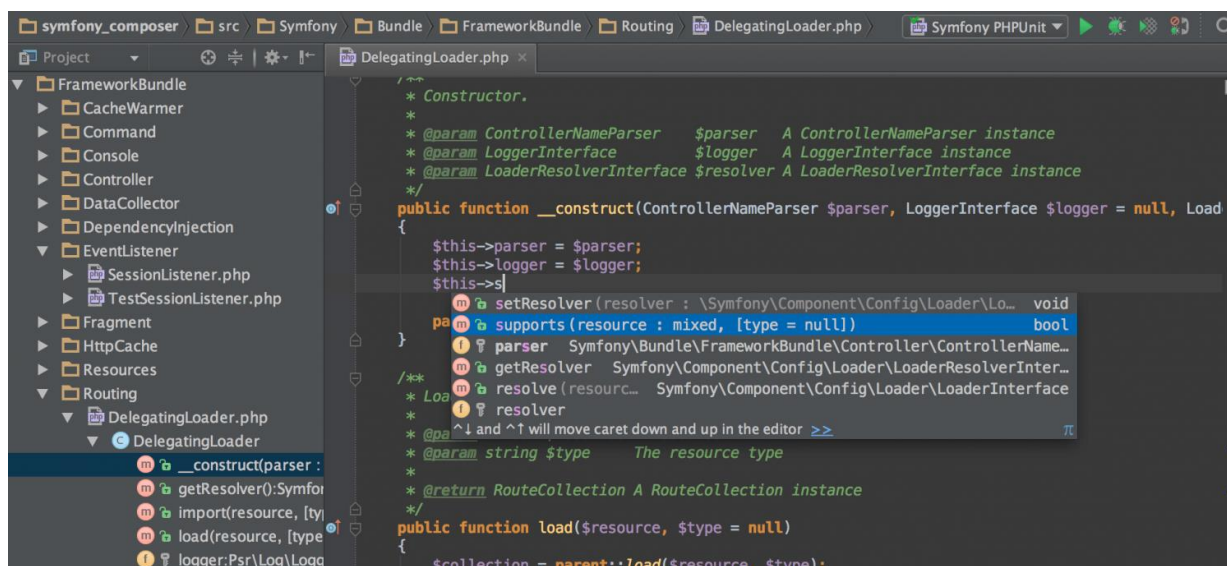


Рис. 1.3.1.1 Скріншот редактора PhpStorm

Мій вибір пав на PhpStorm тому що, у порівнянні з простими редакторами типу Notepad ++, Sublime Text, Text Mate, Coda 2, Espresso - він однозначно виграє за функціоналом, а у порівнянні з деякими з них ще й у зручності. З важкими, типу Eclipse — швидкістю і зручністю.

Плюси:

- Функціональність;
- Модулі;
- Зручний інтерфейс;
- Не гальмує;
- Можливість аналізу коду “на льоту”.

Мінуси:

- Не можна використовувати як швидкий редактор для одного єдиного файлу;
- Потрібно створювати проекти і тримати файли в них;
- Завжди буде локальна копія на вашому комп'ютері.

Висновки до розділу:

У цьому розділі було розглянуто сучасні технології проектування веб - ресурсів, їх принципи та етапи створення web - сайтів. Також було проаналізовано переваги та недоліки кожного з найпопулярніших редакторів коду HTML та обрано найоптимальніший за всіма параметрами - PhpStorm. Ці дані є підґрунтям для майбутнього вибору ієрархічної структури проекту.

2. ПРОЕКТУВАННЯ WEB – САЙТУ ТЮНІНГ АТЕЛЬЄ

2.1. Аналіз існуючих web – сайтів тюнінг ательє

Тюнінг (tuning) - в перекладі з англійської мови - «налаштування». Мається на увазі, що тюнінг - це удосконалення технічних характеристик автомобіля відповідно до потреб його власника.

А слово «стайлінг» походить від слова «стиль». Тут автовласник приділяє увагу зовнішності автомобіля, не зачіпаючи його технічні характеристики.

Стайлінг - видові зміни автомобіля. Такі як аерографія, нанесення захисної або кольорової плівки, тонування, наклейки і зміна дисків. [7]

Web - сайти поділяються на Інтернет - ресурси компаній:

- що продукують деталі, які необхідні для різних видів удосконалення автомобілів;
- що використовують деталі, створені різними компаніями-виробниками.

Розглянемо web – сайт — “Top Level Motorsport” (рис. 2.1.1).

Недоліки:

Дана сторінка має не досить зручного інтерфейсу, за допомогою якого користувач не може легко зорієнтуватися в подальших діях, у клієнта є можливість тільки самостійно зателефонувати, або написати на пошту та в месенджери. Довга стрічка текстової інформації, яка не має естетичної цінності.

На мою думку сей web - сайт не має суттєвих переваг, з тачки зору естетичності дизайну.

Кафедра ПІ				НАУ 21 01 64 – 000 ПЗ			
Виконав	Блейчик Д.О			Проектування web-сайту тюнінг ательє	Літ.	Арк.	Аркушів
Керівник	Сінько Ю.І.					26	46
Консульт.					122 ТП-415Б		
Нормоконт.	Боровик В.М.						
Зав. каф.	Гамаюн В.П.						

КОМПЛЕКСНИЙ ТЮНІНГ АВТОМОБІЛІВ ВІД TOP LEVEL MOTORSPORT

+38-067-465-37-37, +38-067-213-37-37 tuning@top-level.ua

ПІДГОТОВКА АВТОМОБІЛІВ До ЗМАГАНЬ

TOP LEVEL Motorsport надає послуги комплексної споруди (тюнінгу) і підбору тюнінгових деталей двигуна і трансмісії, підвіски під індивідуальні потреби клієнта. Тюнінг-ательє **Top Level motorsport** постійно працює над тим, що б кожен проєкт був індивідуальним, що б бренд **Top Level motorsport** асоціювався завжди з правильним розумінням, що таке тюнінг. Тюнінг-ательє **Top Level motorsport** завжди вибирає тільки якість і надійність для своїх клієнтів. У роботу ми беремо тільки ті проєкти, які, за певними критеріями, відповідають стандартам компанії. Основним принципом тюнінг-ательє **Top Level motorsport** є створення проєктів топ рівня і відповідність поставленим планці в усьому. Маючи великий досвід, **Top Level Motorsport** завжди допоможе Вам підібрати оптимальну для Вас комплектацію тюнінгу, яка буде відповідати Вашим потребам в потужності, еластичності, надійності і ціною. Тюнінг-ательє **Top Level motorsport** співпрацює тільки з кращими виробниками, які займають лідируючі позиції в області тюнінга і кастомайзинга.

Не важливо, у Вас масл-кар як Ford Mustang, Chevrolet Camaro або Dodge Challenger, який Ви бажаєте перетворити в 3000л.с. драгстер, або Ви хочете побудувати Nissan 200SX або 350Z для дрифту, кільцевої болід на базі Mitsubishi Evolution або Subaru WRX STI, або навіть просто хочете прикрасити



Рис. 2.1.1 Сторінка сервісного центру “Top Level Motorsport”

Наступним розглянемо web – сайт — “Tuning&Service” (рис. 2.1.2).

Переваги:

Простий та зручний у використанні дизайн сторінки. На сторінці велика кількість якісних ілюстрацій, що гармонують з загальною стилістикою дизайну web – сайту.

Недоліки:

На даному ресурсі навігація представлена у вигляді вертикального меню, що виїжджаючи закриває собою всю сторінку. Інформаційний текст замалий для візуального сприйняття поряд з великими назвами розділів. Не привабливий тип шрифту, який згубно впливає на очі клієнта.

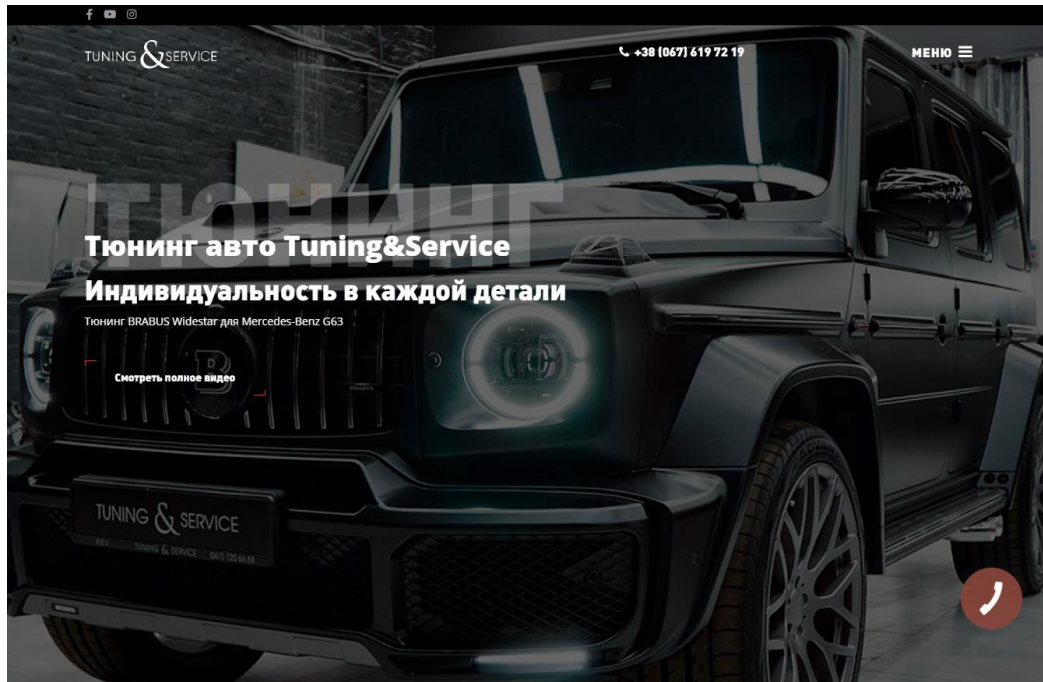


Рис. 2.1.2 Головна сторінка тюнінг ательє “Tuning&Service”

Третім розглянемо український web – сайт виробника тюнінгу Mansory — “Mansory Ukraine” (рис. 2.1.3).

Переваги:

Коротко представлена інформація про всі доступні послуги в швидкому доступі.

Недоліки:

Не симетричний дизайн головної сторінки, що не є естетично правильним. Фотографії не зовсім вдало закомпоновані в просторі.

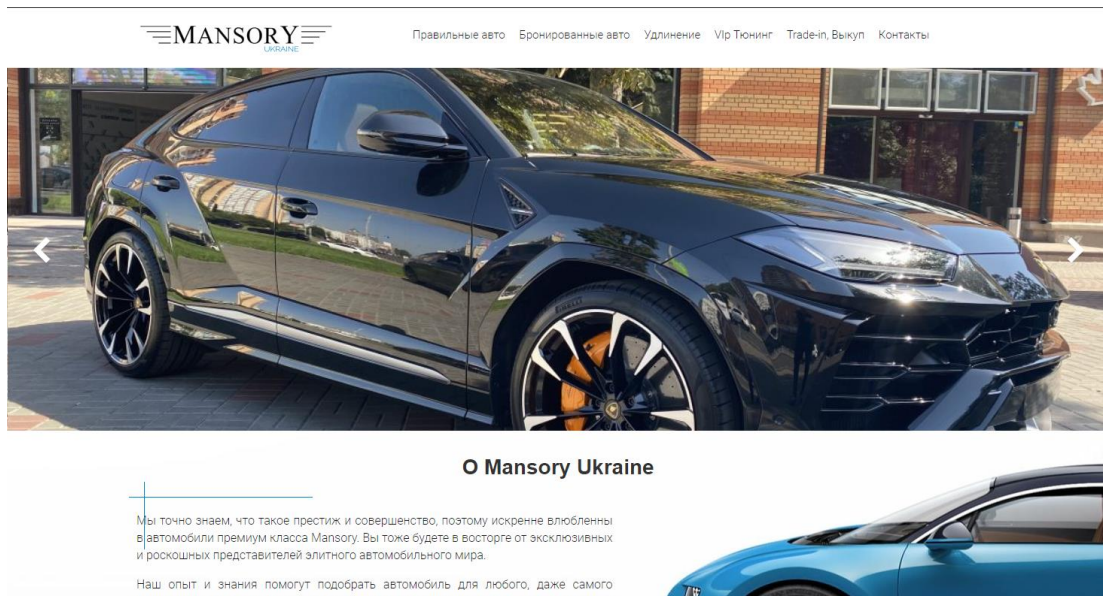


Рис. 2.1.3 Головна сторінка тюнінг ательє “Mansory Ukraine”

Останнім розглянемо web – сайт — “G-Tuning” (рис. 2.1.4).

Переваги:

Відсутнє перенасичення інформацією на web - сайті, який дає змогу зорієнтуватися в подальших діях клієнтам.

Недоліки:

Швидка анімація основної галереї не дає можливості ознайомитись з інформацією, що представлена на слайдах. Невдале поєднання контрастних кольорів у дизайні сторінки.

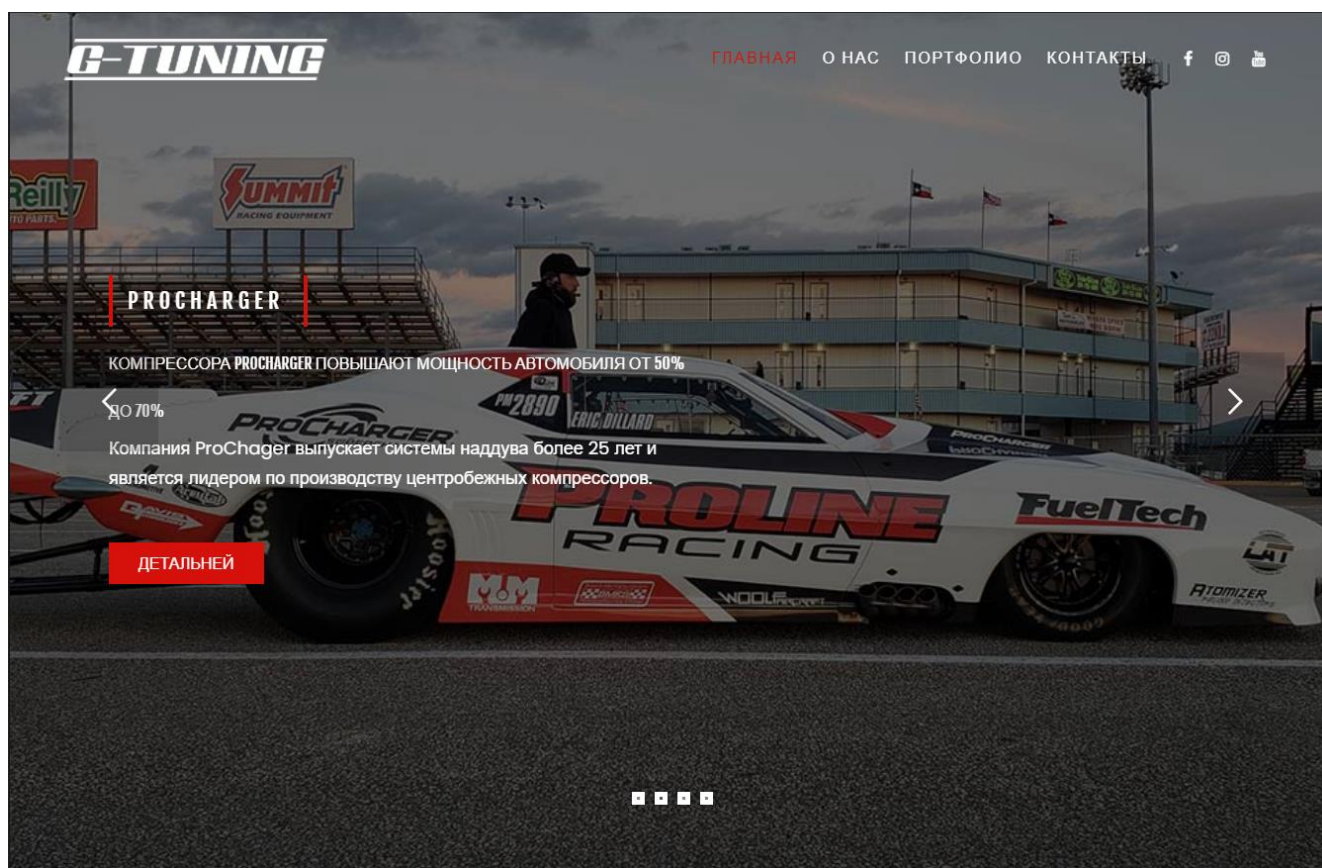


Рис. 2.1.4 Головна сторінка тюнінг ательє “G-Tuning”

2.2. Створення концепту web – сайту

2.2.1 Опис вибраної структури web - сайту

Структура сайту повинна припускати логічність переходу від одного тесту до іншого. Була створена змішана структура “павутина” сайту (рис 2.2.1.1). Ця структура складається з двох та більше компонентів структур в одній, але використовується доволі рідко через складності в реалізації. “Павутина” значно складніша всіх розглянутих вище в розділі. Всі сторінки в ній також розміщені на різних відгалуженнях. Переміщуватися між відгалуженнями на глибокому рівні можливо за допомогою посилань на сторінки інших розділів.

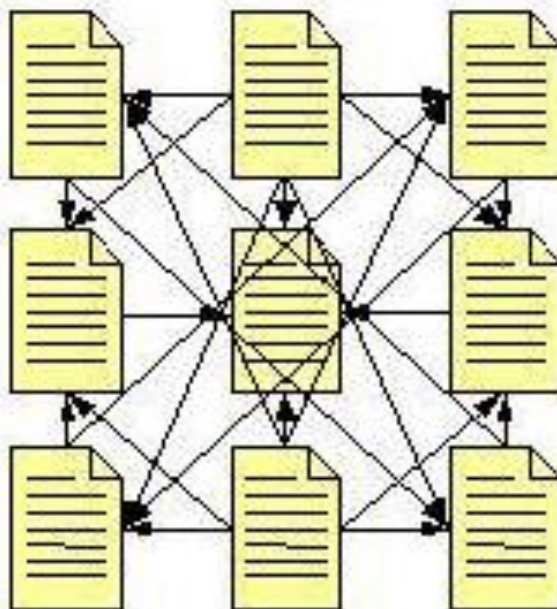


Рис. 2.2.1.1 Змішана структура

2.2.2 Створення прототипу web – сайту

Головний екран прототипу web - сайту зображено на рис. 2.2.2.2. На ньому можна побачити основні елементи цієї сторінки. Де розташоване меню навігації, рід діяльності компанії, форма зворотного зв’язку, “підвал”: логотип, контактні данні, місце розташування на карті.



Рис. 2.2.2.1 Умовні позначення, які використовуються в макеті

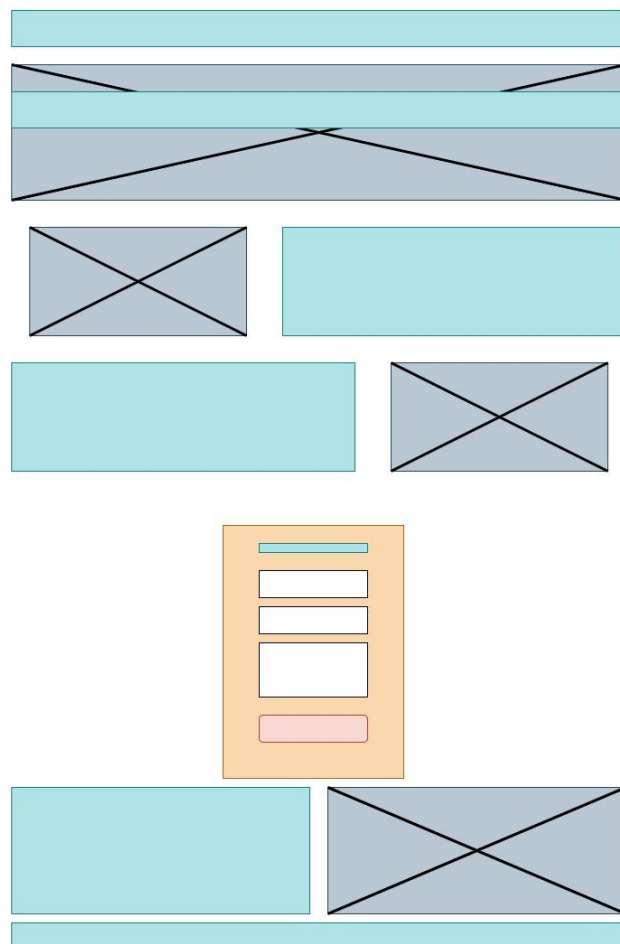


Рис. 2.2.2.2 Головний екран прототипу web - сайту тюнінг ательє.

На сторінці Стайлінгу розміщено блок порівняння зміненого та стокового зовнішнього вигляду автомобіля (рис. 2.2.2.3). Також розміщено опис функцій частин аеродинамічного обвісу, “підвал”, панель навігації.

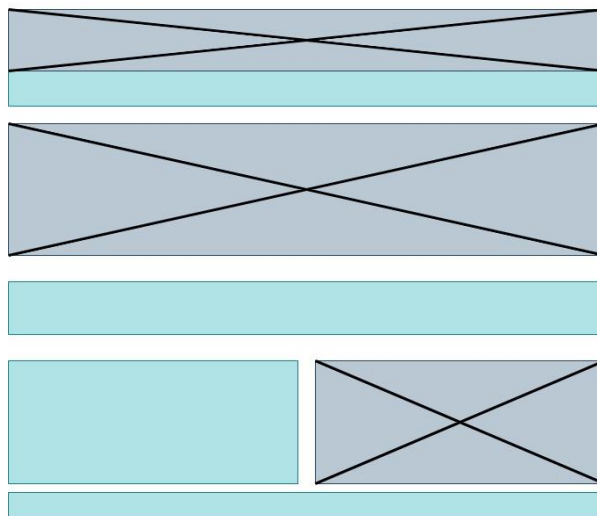


Рис. 2.2.2.3 Прототип сторінки Стайлінгу(Body Kit)

На сторінці Портфоліо розміщена панель навігації, галерея, ‘підвал’.

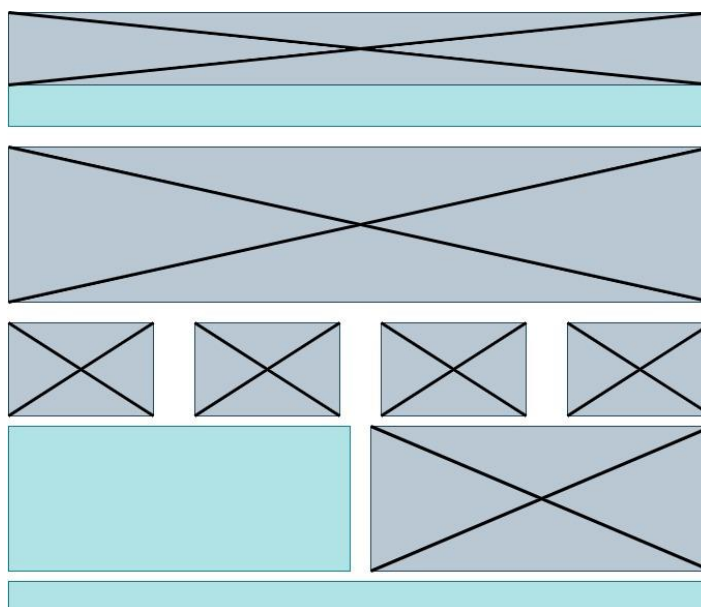


Рис. 2.2.2.4 Прототип сторінки Портфоліо(Вибраний автомобіль)

На сторінці Контакти (рис. 2.2.2.5) розміщена карта зі спутника, основні контактні дані, форма зворотного зв'язку, ‘підвал’, режим роботи, панель навігації.

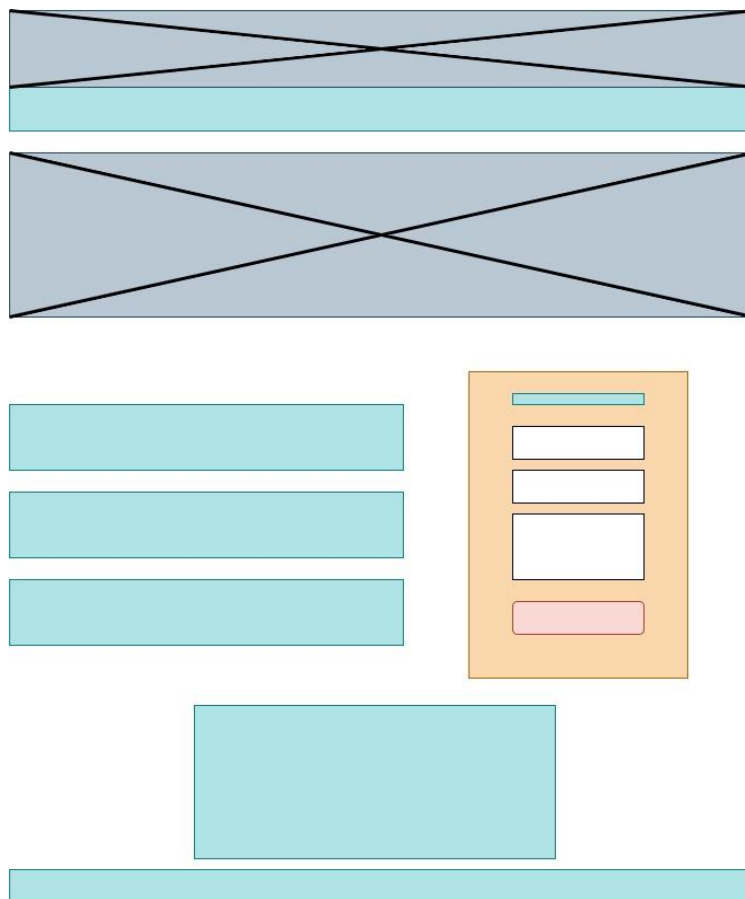


Рис. 2.2.2.5 Прототип сторінки Контакти

На сторінці Питання розміщено блок запитань та відповідей на них, навігаційна панель та “підвал”.

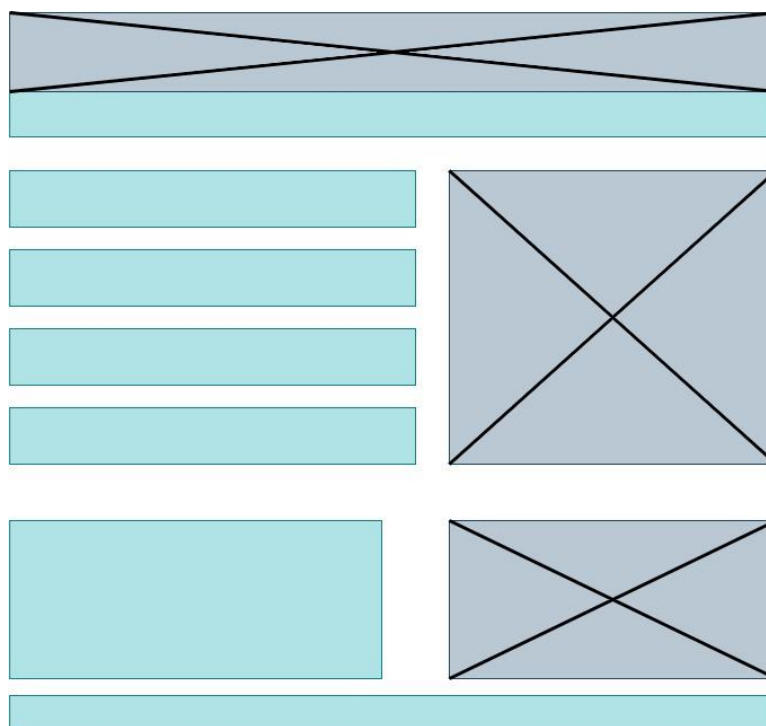


Рис. 2.2.2.6 Прототип сторінки Питання

2.3 Дизайн web - сайту

Ми отримали легкий у сприйнятті прототип web - сайту, на ньому не багато тексту та представлена тільки необхідна інформація. Після розробки прототипу можна більш детально промалювати остаточний дизайн майбутнього сайту.

Рисунок (рис. 2.3.1) створено в програмі Adobe Photoshop, так як ця програма дозволяє швидко і якісно виконати його. В дизайні використовується палітра з сірими та жовтими кольорами. Акцентом являється колір сонця – жовтий. В логотипі зображено перші літери з назви сайту.



Рис. 2.3.1 Дизайн головної сторінки web – сайту тюнінг ательє “TuninGuru”

Висновки до розділу:

У цьому розділі було розглянуто схожі сучасні web – сайти різних тюнінг ательє. Обрано структуру сайту відповідно до вимог. Створено концепти кожної з різних сторінок сайту та оформлено дизайн головної сторінки з використанням технологій Adobe Photoshop.

РОЗДІЛ 3. РОЗРОБКА WEB – САЙТУ ТЮНІНГ АТЕЛЬЄ

3.1 Верстка web – сайту

Перед безпосереднім створенням web - сайту необхідно підготувати файли основних сторінок та папки (рис 3.1.1), що будуть містити необхідні документи, такі як: шрифти, зображення, таблиці стилів та скрипти.

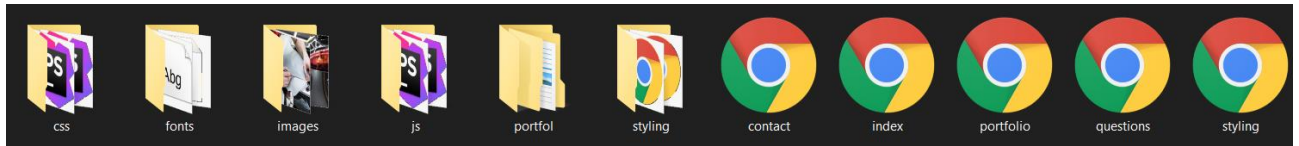


Рис. 3.1.1 Структура папок

Перед початком верстки необхідно запустити функцію debug у редакторі PhpStorm. Буде створено локальний сервер на комп'ютері, що дозволить в режимі реального часу слідкувати за станом верстки.

Для початку пишемо у файлі index.html всю необхідну службову інформацію, додаємо favicon та титульну назву сторінки, а також указуємо шлях до каскадної таблиці стилів. Все це робиться всередині тегу <head> (рис. 3.1.2).

```
5 <head>
6
7 <meta charset="UTF-8" />
8 <link rel="icon" href="images/fav.png" type="image/x-icon">
9 <link rel="stylesheet" href="css/font-awesome.css" type="text/css">
10 <link rel="stylesheet" href="css/style.css" type="text/css">
11 <link rel="stylesheet" href="css/main-page.css" type="text/css">
12
13 <link href="https://fonts.googleapis.com/css2?family=Cuprum&display=swap" rel="stylesheet">
14 <title>TuninGuru</title>
15 </head>
```

Рис. 3.1.2 Службова інформація web – сторінки

Орієнтуючись на макет починаємо створювати блоки з необхідним змістом всередині тегу <body>. Повний текст HTML коду міститься у Додатку А.

У тезі <header> створюємо список посилань з підписками необхідних сторінок. Такий список називається навігаційним меню (рис. 3.1.3) і дає змогу

Кафедра ПІ				НАУ 21 01 64 – 000 ПЗ			
Виконав	Блейчик Д.О.			Розробка web-сайту тюнінг ательє	Літ.	Арк.	Аркушів
Керівник	Сінько Ю.І.					36	46
Консульт.					122 ТП-415Б		
Нормоконт.	Боровик В.М.						
Зав. каф.	Гамаюн В.П.						

потрапити з поточної сторінки, в даному випадку, на будь-яку сторінку web – сайту.

```
18 <header>
19   <nav>
20     <ul class="main-menu">
21       <li class="active"><a href="index.html">Головна</a></li>
22       <li class="menu-children">
23         <a href="styling.html">Стайлінг</a>
24         <ul>
25           <li><a href="styling/kit.html">Body kit</a></li>
26           <li><a href="styling/det.html">Детейлінг</a></li>
27         </ul>
28       </li>
29       <li class="menu-children">
30         <a href="portfolio.html">Портфоліо</a>
31         <ul>
32           <li class="menu-children"><a href="#">Audi</a>
33             <ul>
34               <li><a href="portfol/audi/l11.html">3 Series</a></li>
35               <li><a href="portfol/audi/l12.html">5 Series</a></li>
36               <li><a href="portfol/audi/l13.html">7 Series</a></li>
37             </ul>
38           </li>
39           <li class="menu-children"><a href="#">Volkswagen</a>
40             <ul>
41               <li><a href="portfol/Volkswagen/l21.html">T6</a></li>
42               <li><a href="portfol/Volkswagen/l22.html">T6.1</a></li>
43               <li><a href="portfol/Volkswagen/l23.html">Toureg</a></li>
44             </ul>
45           </li>
46           <li class="menu-children"><a href="#">Cupra</a>
47             <ul>
48               <li><a href="portfol/Cupra/l31.html">Ateca</a></li>
49               <li><a href="portfol/Cupra/l32.html">Formentor</a></li>
50             </ul>
51           </li>
52         </ul>
53       </li>
54       <li><a href="contact.html">Контакти</a></li>
55       <li><a href="questions.html">Питання</a></li>
56     </ul>
57   </nav>
58 </header>
```

Рис. 3.1.3 HTML код навігаційного меню

Далі потрібно розробити “підвал” web – сайту. У його структурі (рис. 3.1.4) зазначаємо координати карти, логотип, назву компанії, контактні телефон та e-mail, та скорочене навігаційне меню.

```

99     <footer>
100     <table class="foot">
101     <tr>
102     <td class="td-f">
103     <div class="img-tun">
104     
105     <a href="index.html">TuninGuru</a>
106     </div>
107     <p class="phone-f">+38 (077) 777-77-77</p>
108     <p class="email-f">Email:
109     <a href="mailto:tuninguru@gmail.com.com">tuninguru@gmail.com</a>
110     </p>
111     <div class="nav-f">
112     <a href="index.html">Головна</a>
113     |
114     <a href="styling.html">Стайлінг</a>
115     |
116     <a href="portfolio.html">Портфоліо</a>
117     |
118     <a href="contact.html">Контакти</a>
119     |
120     <a href="questions.html">Питання</a>
121     </div>
122     </td>
123     <td class="td-f">
124     <iframe class="maps" src="https://www.google.com/maps/embed?pb=!1m14!1r
125     </td>
126     </tr>
127     <tr>
128     <td colspan="2">
129     <div class="copy">
130     <p>© 2021 TuninGuru. All Rights Reserved | Design by
131     <a href="#">Dmytro</a>
132     </p>
133     </div>
134     </td>
135     </tr>
136     </table>
137     </footer>

```

Рис. 3.1.4 HTML код “підвалу” web – сайту

Для реалізації блоку порівняння на сторінці Стайлінг > Body Kit додаємо наступний HTML (рис. 3.1.5) код.

```

71     <figure class="cd-image-container">
72     
73     <span class="cd-image-label" data-type="original">Original</span>
74     <div class="cd-resize-img">
75     
76     <span class="cd-image-label" data-type="modified">Modified</span>
77     </div>
78     <span class="cd-handle"></span> <!-- slider handle -->
79     </figure>

```

Рис. 3.1.5 HTML код блоку порівняння

3.2 Розробка інтерфейсу

Розробка дизайну випадаючого підписку у навігаційному меню (рис. 3.2.1, 3.2.2).

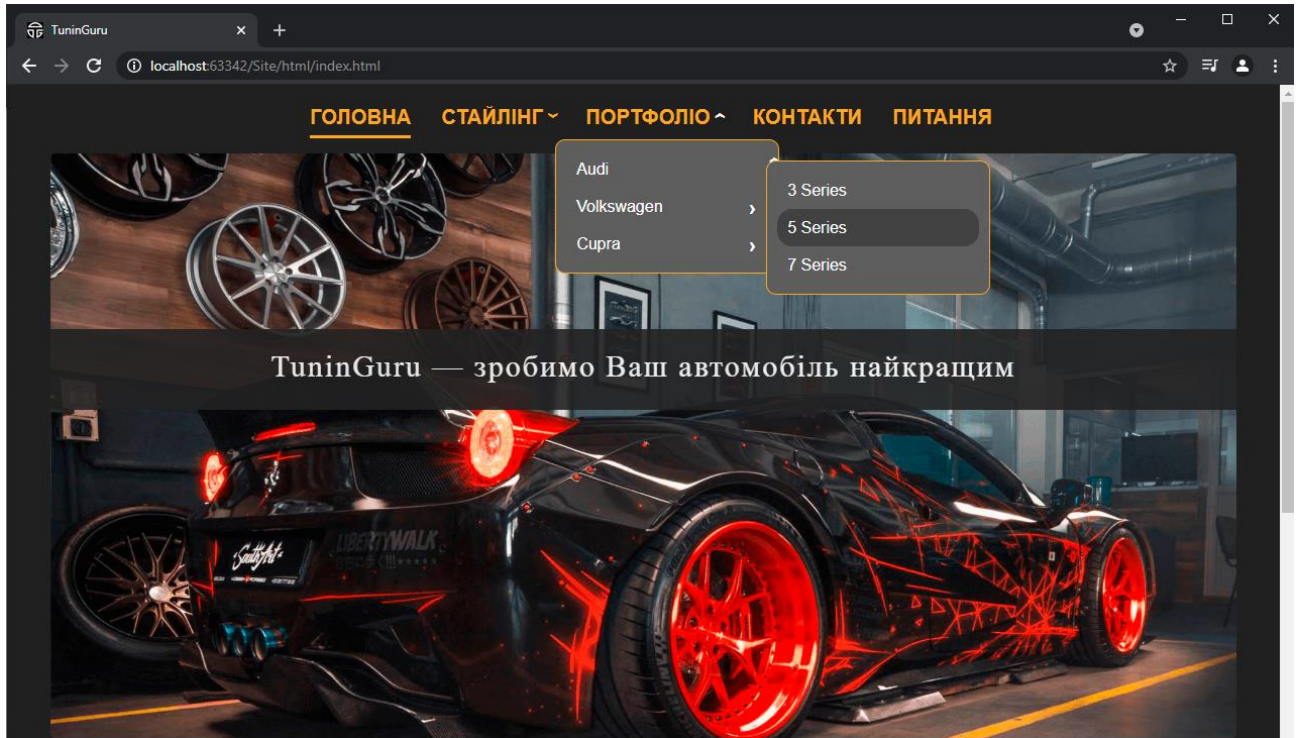


Рис. 3.2.1 Візуалізація навігаційного меню (Портфоліо) web – сайту тюнінг ательє “TuninGuru”

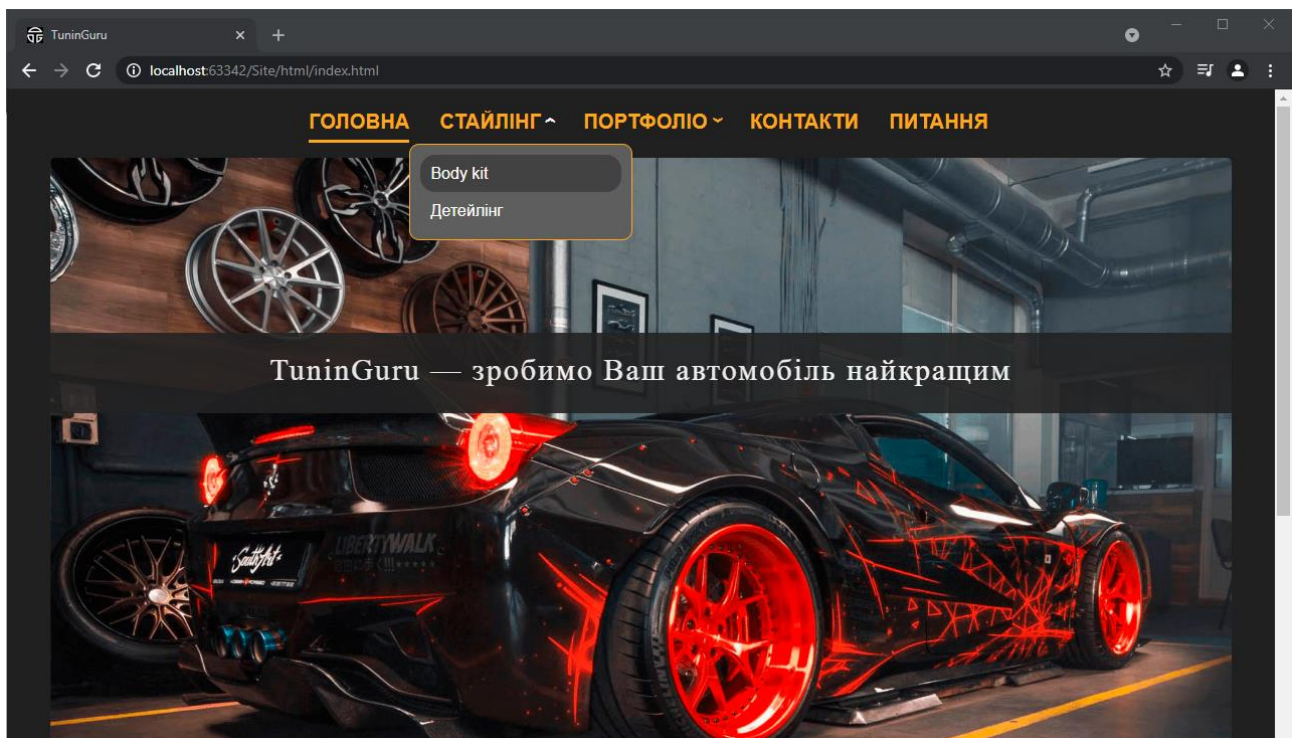


Рис. 3.2.1 Візуалізація навігаційного меню (Стайлінг) web – сайту тюнінг ательє “TuninGuru”

Написаний код (рис. 3.2.3) передбачає анімацію розкриття підписків та руху пожчика їх стану (розкритий/закритий).

```
121     .main-menu li.menu-children {
122         position: relative;
123         margin-right: 12px;
124     }
125     .main-menu li.menu-children:after {
126         position: absolute;
127         content: "\2039";
128         color: #ffa81d;
129         font-size: 20px;
130         font-weight: bold;
131         right: -12px;
132         top: -2px;
133         transform: rotate(-90deg);
134     }
135     .main-menu li.menu-children:hover::after {
136         position: absolute;
137         color: #fff;
138         right: -14px;
139         top: -2px;
140         transform: rotate(90deg);
141         transition: all 200ms;
142     }
143     .main-menu li li.menu-children:after {
144         position: absolute;
145         content: "\2039";
146         color: #FFF;
147         font-size: 20px;
148         font-weight: bold;
149         right: 10px;
150         top: 12px;
151         transform: rotate(180deg);
152     }
153     .main-menu li.menu-children:hover > ul {
154         opacity: 1;
155         visibility: visible;
156         transform: translateY(0);
157     }
```

Рис. 3.2.3 Код дизайну та розкриття підписків

Ще одним важливим місцем на сайті є сторінка Контакти. Вона поєднує в собі форму зворотного зв'язку (рис. 3.2.4) та контактну інформацію (рис. 3.2.5) компанії TuninGuru. Форма зворотного зв'язку поділяється на три поля та кнопку (рис 3.2.6). Поля мають можливість вводу даних, а кнопка анімацію “заповнення градієнтом” (рис. 3.2.7). CSS код дизайну останньої та зміни при наведенні (рис. 3.2.8).

```

19  .contact-form{
20      width: 30vw;
21  }
22  .contact-form_title{
23      color: #fff;
24      font-size: 25px;
25      margin-left: 165px;
26      margin-top: 20px;
27      margin-bottom: auto;
28  }
29  .contact-form_input-wrapper > input{
30      width: 100%;
31      height: 25px;
32      margin-left: 50px;
33      margin-bottom: 20px;
34      color: black;
35      background: #5e5e5e;
36      border: 0px;
37      border-radius: 2px;
38  }
39  .contact-form_input-wrapper > input::placeholder{
40      color: black;
41  }

```

```

25  .contacty{
26      margin-top: 10px;
27      margin-left: auto;
28      margin-right: auto;
29      width: 90vw;
30  }
31  .vertborder {
32      margin-left: auto;
33      margin-right: auto;
34      border-left: 3px solid #ffa81d;
35      height: 50px;
36      width: 1px;
37  }
38  .blok-abo{
39      text-align: center;
40      font-size: 40px;
41      color: #ffa81d;
42  }
43  .liniya{
44      position: relative;
45      height: 1em;
46      width: auto;
47      margin: 10px;
48      padding: 20px;
49      padding-left: 0px;
50      padding-right: 0px;
51      border-bottom: 2px solid #ffa81d;
52  }

```

Рис. 3.2.4 CSS код зворотного зв'язку Рис. 3.2.5 CSS код контактної інформації

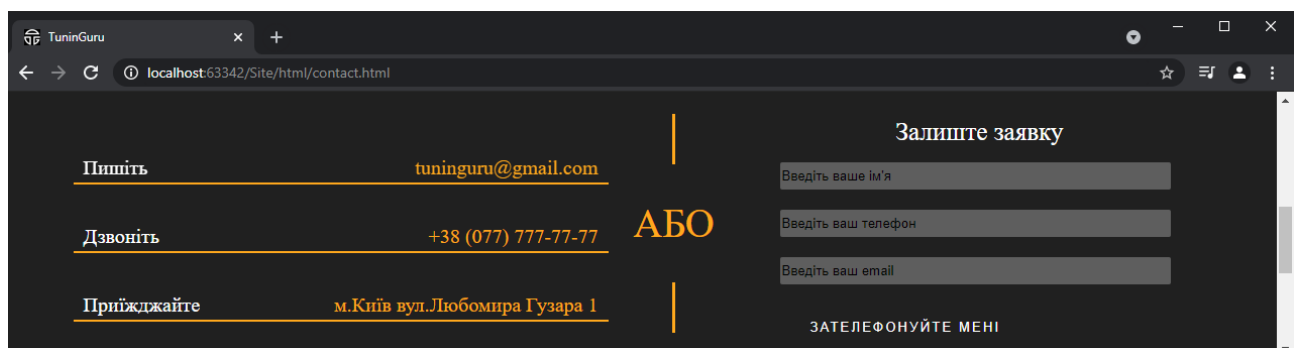


Рис. 3.2.6 Не заповнений блок зворотного зв'язку

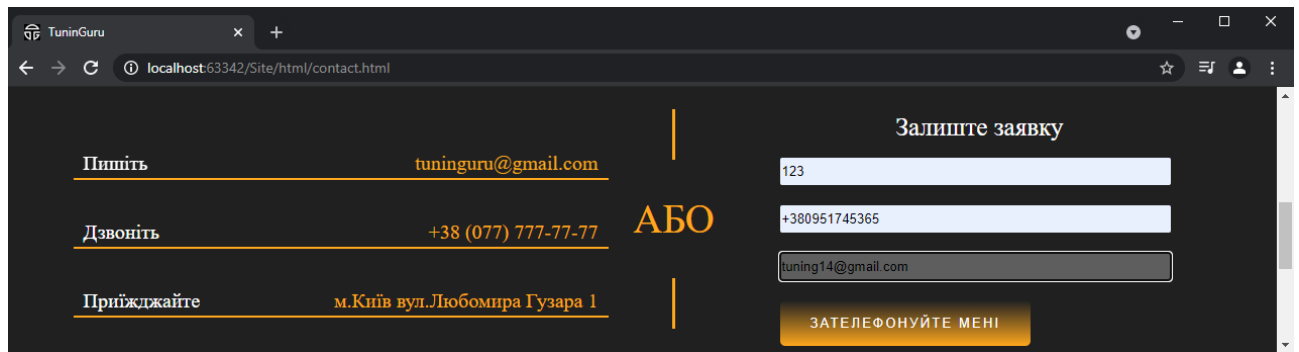


Рис. 3.2.7 Заповнений блок зворотного зв'язку

```

43  .contact-form_sliding-button {
44      text-decoration: none;
45      color: white;
46      display: inline-block;
47      position: relative;
48      padding: 15px 30px;
49      border: 0px solid;
50      border-radius: 5px;
51      background: #202020;
52      border-image-slice: 1;
53      margin: 0px 50px;
54      font-family: 'Montserrat', sans-serif;
55      text-transform: uppercase;
56      overflow: hidden;
57      letter-spacing: 2px;
58      transition: 0.1s cubic-bezier(.165, .84, .44, 1);
59  }
60  .contact-form_sliding-button:before {
61      content: "";
62      position: absolute;
63      left: 0;
64      top: 0;
65      height: 0;
66      width: 100%;
67      z-index: -1;
68      color: white;
69      background: linear-gradient(0deg, #ffa81d, #222222);
70      transition: 0.8s cubic-bezier(.4, .84, .44, .9);
71  }
72  .contact-form_sliding-button:hover {
73      background: rgba(255, 255, 255, 0);
74  }
75  .contact-form_sliding-button:hover:before {
76      bottom: auto;
77      top: 0%;
78      height: 100%;
79  }

```

Рис. 3.2.8 CSS код анімації кнопки

Останнім блоком для будь-якої сторінки є так званий “підвал” (рис. 3.2.9, 3.2.10, 3.2.11).

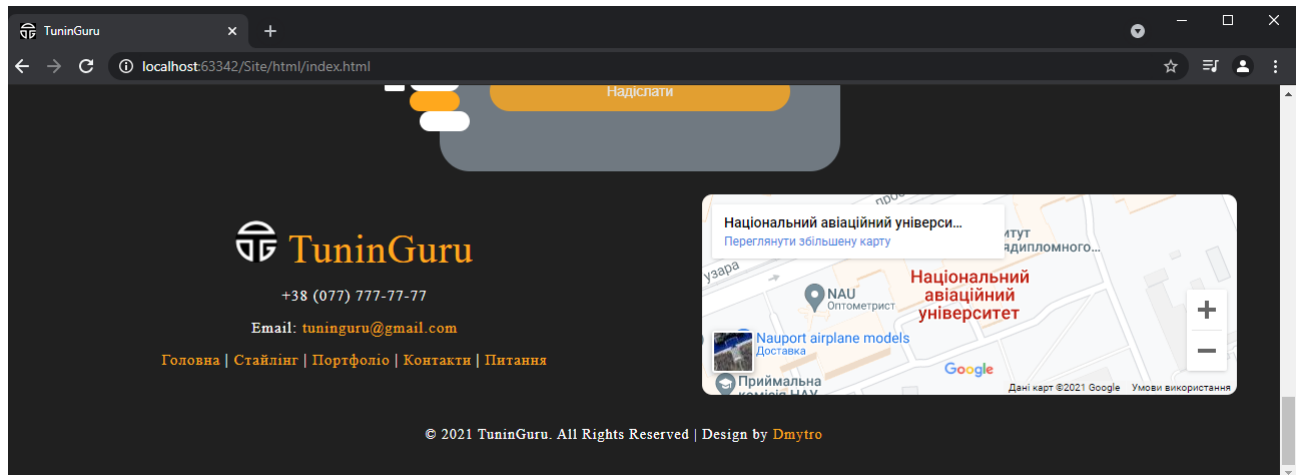


Рис. 3.2.9 Зовнішній вигляд ‘підвалу’

```

212  .foot{
213      width: 95vw;
214  }
215  .td-f{
216      text-align: center;
217      color: #fff;
218      font-size: 15px;
219      letter-spacing: 1px;
220  }
221  .td-f a{
222      color: #ffa81d;
223      text-decoration: none;
224      transition: 0.5s all;
225  }
226  .td-f a:hover{
227      color:#fff;
228      transition: 0.5s all;
229  }
230  .img-tun{
231      font-size: 40px;
232  }
233  .img-tun img{
234      border-radius: 21px;
235      height: 42px;
236  }
237  .nav-f {
238      margin-bottom: 0;
239  }

```

Рис. 3.2.10 CSS код ‘підвалу’

```

240  .maps{
241      float: right;
242      width: 100%;
243      height: 200px;
244      border-radius: 10px;
245  }
246  .copy {
247      text-align: center;
248      background: #202020;
249  }
250  .copy p {
251      color: #fff;
252      font-size: 14px;
253      letter-spacing: 1px;
254      padding: 1em;
255  }
256  .copy p a {
257      color: #ffa81d;
258      text-decoration: none;
259      transition: 0.5s all;
260  }
261  .copy p a:hover{
262      color:#fff;
263      transition: 0.5s all;
264  }

```

Рис. 3.2.11 CSS код ‘підвалу’

Висновки до розділу:

У цьому розділі було розглянуто етапи створення веб – сайту тюнінг ательє, рисунками наведено приклади основних сторінок та код, що дозволяє позиціонувати їх та більш точно розташувати у просторі відносно один одного.

ВИСНОВКИ

На основі аналізу існуючих web – ресурсів був зроблений висновок, що представлені в Інтернеті сайти мають не гармонійні текстову інформацію та кольорову гамму. Також потрапляли у поле зору такі сторінки, що мали не природну анімацію (занадто повільна або швидка).

Були поставлені задачі - Аналізувати існуючі web – сайти в мережі Інтернет, підкреслити недоліки, які слід усунути в розроблені web – сайту.

На основі аналізу проведених робіт створено web - сайт.

У результаті виконаних робіт було використано відомі знання сучасних технологій для створення web – сайту. Використовуючи оптимальний за всіма параметрами редактор PhpStorm був розроблений web - сайт для абстрактного тюнінг ательє TuninGuru. Web – сайт має сучасний, привабливий дизайн зовнішнього вигляду, який буде добре впливати на позитивне сприйняття Інтернет сторінки користувачами.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Сукупність вебсторінок доступних у мережі [Електронний ресурс] // uk.wikipedia.org [сайт]. Режим доступу: <https://uk.wikipedia.org/wiki/%D0%92%D0%B5%D0%B1%D1%81%D0%B0%D0%B9%D1%82>
2. Перший у світі сайт [Електронний ресурс] // w3.org [сайт]. Режим доступу: [сайт]. Режим доступу: <https://www.w3.org/People/Berners-Lee/>
3. Всесвітня павутина складається з веб-сторінок [Електронний ресурс] // Вивчаємо WEB-дизайн дистанційно [сайт]. Режим доступу: <https://sites.google.com/site/vivcaemowebdizajndistancijno/html/lekcja-2-sucasni-veb-tehnologiie>
4. Веб-сайти створюються у кілька етапів [Електронний ресурс] // Все про веб-сайти [сайт]. Режим доступу: <https://sites.google.com/site/siteingschool/etapi-stvorennja-veb-sajtiv>
5. Текстові HTML редактори [Електронний ресурс] // ТОП 10 ЛУЧШИХ HTML РЕДАКТОРОВ [сайт]. Режим доступу: <https://itvdn.com/ru/blog/article/top10-html#4>
6. JetBrains PhpStorm [Електронний ресурс] // uk.wikipedia.org [сайт]. Режим доступу: <https://uk.wikipedia.org/wiki/PhpStorm>
7. Тюнінг [Електронний ресурс] // Автоприбамбасы [сайт]. Режим доступу: <http://www.autopribambasy.ru/news/194/>