

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Національний університет «Запорізька політехніка»

Інститут інформатики та радіоелектроніки,
Факультет комп'ютерних наук і технологій

(повне найменування інституту, назва факультету)

Кафедра програмних засобів

(повне найменування кафедри)

Пояснювальна записка

до дипломного проєкту (роботи)

бакалавра

(ступінь вищої освіти)

на тему РОЗРОБКА ІНФОРМАЦІЙНОГО ПОРТАЛУ “ФУТБОЛЬНА
УКРАЇНА”

DEVELOPMENT OF THE FOOTBALL UKRAINE INFORMATION PORTAL

Виконав: студент(ка) 4курсу, групи КНТ-
128сп

Спеціальності 121 Інженерія програмного
забезпечення

(код і найменування спеціальності)

Освітня програма (спеціалізація)

Інженерія програмного забезпечення

Кучеренко Я.К.

(прізвище та ініціали)

Керівник Неласа Г.В.

(прізвище та ініціали)

Рецензент Терещенко Е.В.

(прізвище та ініціали)

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

Інститут, факультет ІРЕ, 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.
С.О. Субботін
 “ ” 2021 року

З А В Д А Н Н Я

НА ДИПЛОМНИЙ ПРОЄКТ (РОБОТУ) СТУДЕНТА(КИ)

Кучеренко Ярослава Костянтинівича
 (прізвище, ім'я, по батькові)

1. Тема проєкту (роботи) Розробка інформаційного порталу “Футбольна Україна”

Development of the Football Ukraine Information Portal

керівник проєкту (роботи) Неласа Ганна Вікторівна, к.т.н., доцент,
 (прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

затверджені наказом закладу вищої освіти від “ 30 ” березня 2021 року № 103

2. Строк подання студентом проєкту (роботи) 14.05.2021

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

4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити) 1. Аналіз предметної області. 2. Розробка архітектури програми. 3. Розробка програми. 4. Експлуатація та тестування 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
1-4 Основна частина	Неласа Г.В., к.т.н., доцент		
Нормоконтролер	Андрєєв М.О., асистент		

7. Дата видачі завдання “ 12 ”березня 2021 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	2-3 тижні	Розділ 1
3	Розробка архітектури програми	4 тиждень	Розділ 2
4	Розробка програми	5-6 тижні	Розділ 3
5	Тестування та експериментальне дослідження програми	7 тиждень	Розділ 4
6	Оформлення пояснювальної записки та документів до неї. Нормоконтроль та рецензування	8 тиждень	Додатки
7	Захист роботи	9 тиждень	

Студент(ка)

Кучеренко.Я.К.

(підпис) (прізвище та ініціали)

Керівник проєкту (роботи)

Неласа Г.В.

(підпис) (прізвище та ініціали)

РЕФЕРАТ

Пояснювальна записка до дипломної кваліфікаційної роботи бакалавра: 80 с., 21 рис., 3 дод., 15 джерел.

ВЕБ-САЙТ, ФУТБОЛЬНИЙ ВЕБ-САЙТ, ОТРИМАННЯ ФУТБОЛЬНИХ НОВИН, WEBSITE, INFORMATION WEBSITE.

Об'єкт дослідження – процес отримання користувачем футбольних новин та статистичних даних про результати футбольних команд.

Предмет дослідження – веб-сайт, що надає останні новини зі світу футболу, дозволяє переглядати турнірну таблицю чемпіонату України з футболу, а також фото та відео найцікавіших подій або найкрасивіших голів.

Мета роботи – створення веб-сайту, на якому користувач може дізнатись новини з футбольного світу та слідкувати за результатами улюбленої команди.

Матеріали, методи та технічні засоби: мови програмування HTML, JavaScript та CSS, персональний комп'ютер з процесором Intel Core 4 під управлінням операційної системи Microsoft Windows 7, середовище розробки – редактор коду Sublime Text.

Результати. Результатом роботи було виконання аналізу аналогів веб-сайтів та розробка веб-сайту, на якому користувач переглядатиме останні футбольні новини, слідкуватиме за результатами та статистикою команд та трансферами гравців.

Висновки. Розроблено веб-сайт, за допомогою якого можна завжди дізнатись актуальну інформацію про футбольні новини.

Галузь використання – вільне використання в мережі для отримання актуальних новин любителями футболу.

ABSTRACT

Explains the note to the bachelor's degree qualifications: 80 p., 21 fig., 3 dod., 16 sources.

WEBSITE, FOOTBALL WEBSITE, REDUCE FOOTBALL NEWS, WEBSITE, INFORMATION WEBSITE.

The object of work - the process of discarding football novelties and statistical data about the results of football teams.

The subject of the report is a website that has given the last news in the world of football, allowing you to look at the tournament table for the Ukrainian football championship, as well as a photo and video of the most beautiful goals.

The purpose of the work - the root of the website, on which you can find the news from the football match and follow the results of your favorite team.

Materials, methods and technical resources: web software, HTML, JavaScript and CSS, a personal computer with an Intel Core 4 processor for operating system management in Microsoft Windows 7, the middle of the package is a SublimeText code editor.

Conclusions. The result of the robotics was the analysis of the analogs of the web-sites, the development of the web-site, on what the last football news, following the results of the statistics of the teams and the transfers of gravitas.

Visnovki. The website has been broken up, for the help of which it is possible to find out the latest information about football news.

Galuz vikoristannya - free in fancy for the removal of thematic information.

ЗМІСТ

	С.
Перелік скорочень та умовних познач	7
Вступ	9
1 Аналіз предметної області	10
1.1 Проблема інформаційних порталів	10
1.2 Програмне забезпечення інформаційних порталів	10
1.3 Огляд програмних аналогів	10
1.4 Висновки за розділом 1	13
2 Розробка архітектури програми	14
2.1 Вибір мови програмування	14
2.2 Середовище розробки	20
2.3 Функціональні вимоги до програмного забезпечення	27
2.4 Висновки за розділом 2	29
3 Розробка програми	30
3.1 Порядок функціонування програми	30
3.2 Опис реалізації програми	31
3.3 Висновки за розділом 3	45
4 Експлуатація та тестування програми	46
4.1 Призначення програми	46
4.2 Умови виконання програми	46
4.3 Робота з програмою	46
Висновки	51
Перелік джерел посилання	52
Додаток А Технічне завдання	54
Додаток Б Текст програми	58
Додаток В Слайди презентації	77

ПЕРЕЛІК СКОРОЧЕНЬ ТА УМОВНИХ ПОЗНАК

HTML	– HyperText Markup Language;
CSS	– Cascading Style Sheets;
IDE	– Integrated Development Environment,
HTTP	– Hyper Text Transfer Protocol;
JS	–JavaScript;
ANSI	– American National Standart Institute;
XML	– eXtensible markup language;
BC	– вбудована система;
ПИ	– Пользовательский интерфейс;
ПК	– персональний комп'ютер;
ПЗ	– програмне забезпечення;
ТЗ	– технічне завдання.

ВСТУП

Розвиток технологій останнім часом досяг високого рівня і це не могло не відобразитися на розвитку веб-програмування також. Тому, з розвитком веб-технологій, з'явилися веб-сайти, що надають користувачам новини з різних тем, зокрема і футбольних новинних веб-сайтів.

Але, як я уже зазначив, розвиток технологій останнім часом досяг високого рівня тому, дуже важливо, щоб сайт був наповнений не лише текстом, а й був легким у використанні, мав адаптивний дизайн та гарну інтерактивність. Особливо на новинні сайти з футбольною тематикою важливо додавати багато елементів, наприклад турнірних таблиць або відео найкращих голів.

Тому для веб-сайту з футбольними новинами важливо не лише розширити можливості функціонування сайту, а також за допомогою нових технологій можна додавати на сайт багато інтерактивних елементів. Таким чином, можна зрозуміти, що веб-розробка насправді допомагає робити сайти більш прийнятними для користувача, що й є основною ціллю розробки веб-сайту.

Також, хотів би додати, що без новітніх технологій важко комунікувати з користувачем. Наприклад, якщо ми хочемо, щоб користувач залишив свій відгук про користування нашим продуктом, ми повинні додати спеціальну форму для того щоб користувач міг скориставшись нею відправити нам відгук. Кожен з веб-розробників повинен перш за все ціллю своєї праці ставити комфорт користувача та розробляти систему з урахуванням побажань користувача. В моєму випадку дослідивши веб-сайти, що надають футбольні новини, я виявив декілька проблем, які обов'язково хочу вирішити при розробці мого веб-сайту, який буде надавати користувачу можливість передивлятись останні футбольні новини та статистичну інформацію щодо гравців та команд.

1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ

1.1 Проблема інформаційних порталів

Інформаційні портали – це вид новинних веб-сайтів або веб-сторінок, що передбачає отримання тематичної інформації користувачем та пошук необхідної інформації. Такі веб-сайти можуть включати інформацію, яку шукають користувачі мережі в Інтернеті або особисто, і можуть охоплювати такі запити, як отримання новин, пошук необхідної інформації, отримання статистичних даних, при чому такі веб-сайти мають багато переваг, оскільки дозволяють отримувати інформацію необхідну користувачеві у будь який час доби та з будь якого місця земної півкулі, шукати інформацію, яку користувач хоче знайти в Інтернеті, та багато інших корисних функцій та переваг.

Як і для будь-якого іншого виду розробки, важливо робити веб-сайти, які будуть добре відповідати побажанням та інтересам користувача: наприклад – мати гарний дизайн, багато корисної інформації та функцій або володіти легким інтерфейсом для того щоб кожен користувач без великого обсягу необхідної підготовки та комп'ютерної грамоти міг би користуватись продуктом.

Існує декілька різних типів новинних веб – сайтів:

- веб-сайти, що є суто новинними, характеризуються простим інтерфейсом та постійною оновлюванністю контенту. Також можуть підтримувати реєстрацію та авторизацію користувача та коментарі;

- веб-сайти з направленістю по інтересам користувача : характеризуються менш жорсткою направленістю;

- веб-сайти з професійною направленістю: характеризуються направленістю серед користувачів по їх професійним характеристикам;

Також є багато інших веб-сайтів які відповідають іншим запитам користувачів та є теж доволі корисними та комфортними для користувачів.

1.2 Програмне забезпечення інформаційних порталів

Програмне забезпечення інформаційних порталів повинно бути заснованим на таких принципах:

- комфорт для кожного користувача: кожен інформаційний портал повинен бути комфортним для кожного користувача, не зважаючи на його підготовленість, вік або інші показники;
- мати легкий і доступний інтерфейс: виходячи з того що користувач може володіти лише набіром основних прийомів та навичок роботи з ПК, перед розробником завжди постає задача розробити легкий і простий інтерфейс зважаючи на функціонал;
- широкий функціонал: кожен інформаційний портал повинен володіти широким функціоналом для отримання та пошуку інформації новим користувачем;
- кольорова гамма: здавалось би новинний веб-портал не повинен володіти особливою кольоровою гаммою, але важливо підбирати кольори та півтони, які будуть добре підходити для користувача та не відволікати його увагу на сторонні елементи;
- створюючи веб-сайт, потрібно, щоб усі елементи були видимими та доступними.

1.3 Огляд програмних аналогів

Програмні аналоги включають рішення різного рівня: більшість з них є універсальними і дозволяють отримувати інформацію різних типів, інша частина є вузькоспеціалізованими і дозволяють, наприклад, отримувати інформацію лише з певних категорій або інтересів.

Деякі з тих програмних засобів, які було проаналізовано мною мають особливості у розробці, інтерфейсі або інших елементах відносно інших конкурентів.

BBC News (рис. 1.1) – це веб-сайт, що дозволяє користувачам отримувати найновішу інформацію з усього світу, переглядати та вибирати необхідні категорії, оцінювати та аналізувати інформацію та шукати саме необхідну для користувача інформацію.

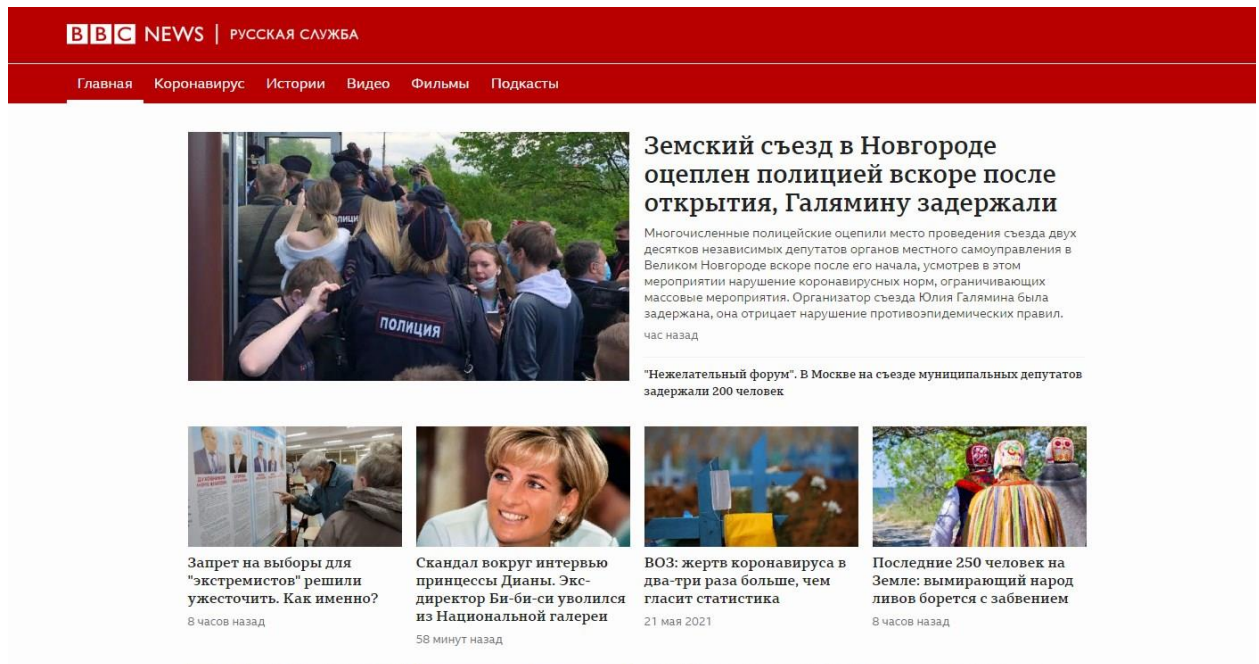


Рисунок 1.1 – Перегляд головної сторінки на вебсайті BBC News

Щодо новинного веб-порталу BBC News можна відзначити:

- інформація, яку отримує користувач є актуальною, та постійно оновлюванною;
- хоч новинний веб-портал BBC News і є доволі популярним порталом у світі, але має у своєму функціоналі простий та легкий інтерфейс користувача, для користування якому не потрібно володіти особливими навичками;
- кольорова гамма порталу BBC News підібрана як найкраще, жоден з елементів головної сторінки не виділяється та не відволікає користувача від основної задачі.

Football.UA (рис. 1.2) – це веб-портал зі спортивною направленістю, який я вибрав як приклад спортивного новинного веб-порталу.

Football.UA відзначається наступним:

- інформація, яку отримує користувач є актуальною, та постійно оновлюванною;
- головна сторінка несе в собі багато корисної для користувача інформації, але при цьому, хотілось би виділити те, що вона не перевантажена контентом, та є приємною в інтерфейсі користувача;
- кольорова гамма порталу Football.UA підібрана добре, жоден з елементів головної сторінки не виділяється та не відволікає користувача від основної задачі.

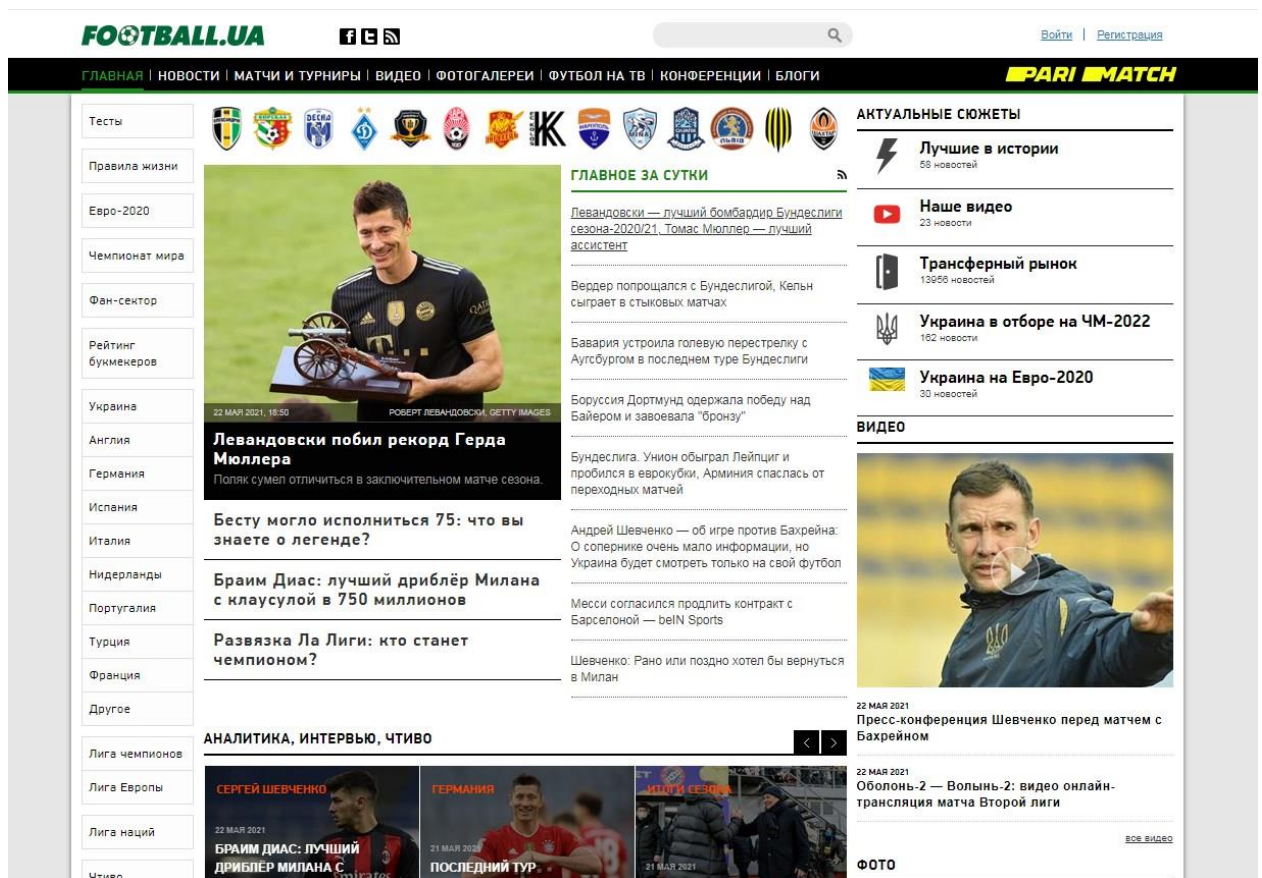


Рисунок 1.2 – Перегляд головної сторінки на вебсайті Football.UA

Особливістю є те, що меню користувача розміщено зліва від основного блоку новин та є інтерактивним.

Недоліком веб-сайту, я вважаю, є складність пошуку серед запитів та авторизація на веб-порталі.

Підводячи підсумок проведеного огляду програмних аналогів, необхідно вказати на те, що кожен з розглянутих мною веб-порталів об'єднує величезну кількість корисних застосунків та функцій користувача.

1.4 Висновки за розділом 1

У першому розділі мною було виконано огляд проблеми виконання та розробки інформаційних порталів: розглянуто особливості, категорії і приклади інформаційних порталів, як з широкою так і з вузькою тематиками. Виконано огляд програмних аналогів, з виділенням їх функціональних можливостей, особливостей та недоліків. Програмні аналоги, якими є новинні футбольні веб-сайти включають сервіси BBC News, Football.UA. Дані сервіси є повністю безкоштовними в користуванні та виконують функції отримання та пошуку користувачем необхідної йому та актуальної інформації.

Завданнями дипломної кваліфікаційної роботи, що мають бути розв'язані в наступних розділах, є:

- розробка технічного завдання і визначення вимог до програми;
- розробка архітектури програми;
- розробка програми;
- тестування програми.

2 РОЗРОБКА АРХІТЕКТУРИ ПРОГРАМИ

2.1 Вибір мови програмування

Аналогічно розглянутим програмним аналогам програмна реалізація буде виконуватися як веб-розробка.

Першим і початковим кроком для того, щоб почати підготовку є вибір мов програмування для створення програмного забезпечення. Я проаналізував всі мови, які можливо використати для розробки. Для цього мною було розглянуто такі мови, як мови HTML (Hyper Text Markup Language — мова розмітки гіпертексту) та CSS (Cascading Style Sheets, Каскадні таблиці стилів) та мову скриптів JavaScript, інформацію про які я додаю надалі.

HTML (Hyper Text Markup Language — мова розмітки гіпертексту) — це мова тегів, за допомогою якої забезпечується розміщення усіх веб-сторінок у всесвітній мережі Інтернет. Ця мова є базовою для розробки сайтів і її використовує більшість розробників. Веб-оглядачі ж отримують HTML-файли з веб-сервера або з пам'яті та передають файли в надані користувачу веб-сторінки. HTML записує шаблон веб-сторінки покроково і першим кроком включає сигнали для приведення вигляду документа. Мова характеризується легкістю та достатньо великим функціоналом. Наприклад, І.В.Мархвіда описує це так «...HTML-документ, або Web-сторінка, шокує своєю лаконічністю. Однак сам документ небагато коштує без середи перегляду Web-сторінок, а саме — браузера.» [1].

Цеглинки HTML є будівельними цеглинами сторінок HTML. За допомогою цих будівельних етапів HTML, фото-елементів та інших часток, такі форми, можуть бути переглянуті у основних сторінках для перегляду користувачем. HTML має засоби для написання гарно описаних документів, особливо включаючи структуровану розмітку тексту, наприклад заголовки, виділення, списки, посилання, цитати та інші елементи. У розробці мого сайту я буду намагатись використати всі ці можливості та інструменти. Блок

HTML виділений елементами, написаний за допомогою своєрідних дужок. Теги, є і такі, що виводять вміст на сторінку. Інші теги, такі як оточують і надають інформацію про текст документа і можуть включати інші теги як піделементи. Веб-переглядачі не показують нам теги HTML, але використовують їх для обробки і показу вмісту сторінки. Користувач не бачить, що саме робить той чи інший тег, але бачить результат роботи цих тегів.

HTML може за допомогою дружніх засобів вбудовувати сценарії, підготовлені на мові сценарію, такого як JavaScript, яку теж використано у розробці і вона сильно впливає на поведінку та контент веб-сторінок. Включення CSS допомагає зробити сторінку більш привабливою. World Wide Web (назва W3C), який розроблює стандарти HTML та CSS, усіляко допомагає у використанні CSS над вже застарілим HTML з 1996 року.

Тепер можна розповісти про те, які є сильні сторони мови HTML і про те, де цю мову можна використати при розробці. Зараз увага: ми будемо переглядати основні можливості HTML 4, і не ускладнювати розуміння основ HTML. Так само потрібно розуміти, що розглядати можливості HTML 5 треба лише після того, як ви вивчили головні можливості, так буде набагато простіше і зрозуміліше. Сильною стороною є те що розробка сайту на мові HTML є доволі простою, але при цьому непідготовлений користувач не зможе зрозуміти призначення того чи іншого тегу. До прикладу, І.В.Мархвіда пише про цю властивість так «...HTML, розроблений для того, щоб відображати текст, схожий на звичний кожному текстовий файл, за виключенням деяких операторів, які браузер інтерпретує як розмітка. Вона впорядковує документ та надає йому структури. Якщо, відкривши файл у тестовому режимі, ви одразу не зможете зрозуміти відкритий текст, знайте, вам до рук потрапила дуже гарна сторінка.» [1].

Нарешті перейдемо до розгляду можливостей HTML. Основною задачею HTML є обробка і оформлення інформації на сторінці та відображення її у зрозумілому вигляді для користувача. Мова HTML

дозволяє обробити отриману інформацію багатьма методами: створювати таблиці, декілька видів списків, параграфи, додавати в документи зображення, розбивати документи на частини за допомогою заголовків, створювати форми для взаємодії з користувачами, пов'язувати різні HTML документи списками.

Основною і головнішою сферою застосування HTML є всесвітня мережа Інтернет. Кожен веб-сайт для подання контенту користувачу в простому форматі використовує і HTML, але інтернетом HTML не обмежується, його дуже активно призначають при розробці мобільних і прикладних додатків для різного роду операційних систем. Однозначно, можу зробити для себе висновок, що цю мову треба використовувати у розробці мого веб-сайту.

CSS (Cascading Style Sheets, Каскадні таблиці стилів) – це особлива мова стилю сторінок, що призначена для описання їхнього вигляду. Самі ж сторінки написані мовами розмітки даних, що можна зрозуміти отримавши документ. А.В Петюшкин має щодо CSS таку думку «...CSS — це мова стилів. Вона використовується, для того щоб надати сторінкам написаним на мові HTML — основоположній мові усієї Всесвітньої павутини — досконалого вигляду. Маю надію, ви в своїй роботі будете використовувати каскадну мову таблиць та стилів, для того щоб зробити свої сторінки найкращими.» [2]. CSS є основною технологією всесвітньої мережі, поряд із іншими провідними технологіями.

Частіше CSS призначається розробниками для візуальної обробки сторінок, написаних HTML та XHTML, але мова CSS може призначатися і до інших видів XML-сторінок. Специфікації CSS були прийняті за основу і розвиваються Консорціумом Всесвітньої мережі. Без використання цієї мови займатися розробкою веб-сайту в 2021 році точно рекомендувати не можна.

CSS має різні особливості та додатки. Наступний додаток CSS базується на основі попереднього, додаючи нові можливості або розширюючи вже наявні функції. Додатки позначаються як CSS першого

порядку, CSS другого порядку та CSS третього порядку. Особливості – збірка основ CSS першого або більших рівнів, які визначені для декількох видів гаджетів або застосунків. Наприклад, є профілі CSS лише для печаті, переносних приладів, тощо. Такі функції нам для розробки не потрібно використовувати, але розглянути їх теж було потрібно.

CSS (каскадна або блокова верстка) розроблялась на зміну верстці таблицями веб-сторінок. Головна з переваг блочної верстки – ділення змісту сторінки (даних) та їхньої переносимості користувачем. Блоки, в порівнянні з таблицями можна ефективніше та простіше використовувати.

Зустрівши набір букв CSS першого разу, кожен зі студентів вже підозрює, що це якимось чином має зв'язок з сайтами. Розберемося: CSS - Cascading Style Sheets – це такі собі таблиці стилів. Якщо казати простою мовою – мова, яка визначає зовнішній вигляд HTML-документа. Переважаюча більшість професійних веб-сайтів мають скелет на основі зв'язки HTML + CSS. Потім вже на цей скелет лягає інші інструменти з інших мов програмування. Також, важливо зазначити, що більшість фреймворків теж засновані на цій зв'язці.

Тепер коснімося того, що саме робить CSS на сторінці. Також кажучи просто: коли HTML показує контент на сторінці, то CSS дозволяє змінити його, зробити більш привабливим для користувача. Спочатку веб-розробники використовували тільки один HTML – так було при розробці перших сайтів і розвитку інтернет-технологій. За допомогою цього ще досі можна виділити параграф, заголовок, змінити стиль шрифту. Але, як вже було зазначено вище, ніхто з професійних веб-розробників сучасного світу не використовує таку застарілу технологію розробки веб-сайтів.

А вже коли і цих можливостей стало мало – тоді з'явилася можливість не змінювати сам документ та його вміст, а прописувати і підключати стилі окремо. CSS змінив проблему тих часів, коли різні браузері використовували і розуміли різні теги. Те що вони стали більш структурованими та описаними у переважній більшості специфікацій, те що з'явилася документація щодо

того які теги потрібно використовувати, звичайно спростила розробку програмістам. Також, в подальшому, це стало не просто гарною звичкою, а й необхідним правилом, і всі веб-розробники повинні його дотримуватись. Пояснити таке правило дуже просто – більшість з великих проєктів пишуться один раз одним програмістом, або командою програмістів, а от підтримувати його вже потрібно буде зовсім іншим програмістам, і щоб всі розуміли проєкт та не витрачали багато часу на розуміння коду потрібно дотримуватись всіх правил та специфікацій.

JavaScript (або JS) – змінювана, мова програмування яка має підтримку прототипів та зазвичай на ній пишуть сценарії сторінки. Описана стандартизаціями ECMAScript. Найчастіше призначається для розробки сценаріїв веб-сторінок, що дозволяє користувачу на його боці взаємодіяти з розробленим інтерфейсом, керувати веб-оглядачем, відправляти та отримувати данні з серверу,самостійно впливати на сценарії, що проходять на стороні користувача системи. П.Б Храмцов заявляв про мову програмування JavaScript таке «...з найперших днів появи мови програмування JavaScript вона стала невід’ємною частиною продукту, який споживають користувачі. Хоча спершу JavaScript починав свою роботу з великих помилок, через декілька десятиліть ця мова зарекомендувала себе таким чином, що мало хто вже думає що мова JavaScript не вдало підібрана для застосування веб-технологій.» [3]. Потрібно розібратись з тим, чому ця мова могла з’явитись.

JavaScript класифікують як мову, яка підтримує або має підтримку прототипів скриптів мови програмування з динамічним форматом та є також типізованою мовою, серед конкурентів. За думкою, М.Холла «...JavaScript став однозначним стандартом у розробці динамічних за стосунків для мережі Інтернет. Кожен раз, коли Вам на очі потрапляє сторінка з динамічною анімацією, автоматично оновлюючимися даними, елементами, що змінюють свій вигляд під дією наведеної їх миші, або динамічним меню, обов’язково зверніть на це свою увагу, бо скоріш за все ви маєте налідки роботи з

JavaScript.» [4]. З цього можна зробити висновок, що JavaScript виконує багато функцій та важливу роль у розробці сторінки. Окрім основної функції, JavaScript ще трохи підтримував інше правило верстки імперативне та частково функціональне та окремі архітектурні властивості, наприклад: прості та невибагливі вимоги до типізації, зручне керування носіями, знайоме наслідування, та функціонал як об'єктів перших класів. Найяскравіше описує функціонал JavaScript М. Холл «...JavaScript спочатку розроблювався для роботи лише у браузері, але ж надалі сталося справжнє диво. В наші дні JavaScript використовують у смартфонах і планшетах, в веб-серверах та інших застосунках, а також усілякого роду мобільних додатках та інших прикладах у побуті. » [4].

1994 року група програмістів отримала задачу вибудувати мову програмування чи «якусь схожу» в браузер, що мав називатись Netscape. Їм для цієї задачі був запрошений Брендон Ай веб-розробник з Америки, котрого характеризували як системного програміста. Також, щоб прискорити розробку, група програмістів почала роботу з компанією Microsystem_Chip. А ось як пригадує розвиток JavaScript М.Холл «...Раніше JavaScript виконувався в браузерах дуже повільно, чим заслугував у програмістів тих часів погану славу. Зараз же код на JavaScript виконується лише на 20% повільніше, ніж компільований код, при чому з часом цей показник все більш покращується. Це означає, що сучасний JavaScript отримав набагато більшу продуктивність, ніж ті його версії, котрі існували лише кілька років назад, не кажучи вже про перший варіант, який випустили в 1995 році. » [4]. І справді, мова хоч і не є компільованою, але її використання доволі комфортне.

З часом, концепція розроблюваної мови програмування була розширена до можливості використання безпосередньо в HTML-коді сторінки. Компанії мали на меті створити мову, що могла зв'язати різні шматки веб-сайтів: зображень, скриптів, моделі документообігу. Ця мова повинна була стати зручною для веб-дизайнерів та некваліфікованих

програмістів. Коли мову все ж розробили її назвали mocha, але вже згодом її змінили на LiveScript в перших двох тестових версіях браузера Netscape. А дещо пізніше, розробники побачили, що мова Java стала неймовірно популярною на ті часи, і наш LiveScript був названий вже JavaScript, назву до якої ми звикли і яка є сучасною і третя тестова версія Netscape вже вийшла з сучасною назвою. Цікавим фактом є також те що була навіть придбана свого роду ліцензія у компанії Sun, що володіла брендом Java, для того щоб не порушувати авторське право. І це було явно правильним рішенням.

Тисяча дев'яност дев'ятого року групою Nombas була розроблена скриптова мова програмування Cmm (C-минус -минус, свого рідку відсилка до мови C++), яка була вже згодом теж переназвана на Scriptase та була дуже схожою до мови яку ми сьогодні розбираємо. Існує така думка, що JavaScript була перейнята у технології Cmm. Але, насправді розробниками мови JS навіть не була відома технологія Cmm. Але, потім час розставив все по місцях і Nombas зупинили роботу над Cmm та почали вивчати можливості JavaScript, а згодом навіть приймали участь у стандартизації JavaScript.

За результатами порівняння було обрано усі з вищезазначених мов програмування для реалізації, оскільки вони активно використовується саме в галузі веб-розробки, та кожна з них доповнює одна одну. Для того щоб зробити найбільш цікавий та корисний інтерфейс потрібно використати найсильніші сторони кожної з мов програмування. Тепер потрібно вибрати середу розробки.

2.2 Середовище розробки

Google Chrome – це знайомий всім безкоштовний браузер, написаний групою Google та котрий став першовідкривачем у світі веб-браузерів з доступним кодом і інших доступних програмних продуктів. Його програмна частина написана буквально з пустого листка паперу (хоча варто зазначити і частки інших програмних продуктів з відкритим кодом) з основною ціллю

задовольнити потреби користувача. Новий продукт перш за все підтримує те правило, що сьогодні майже всі сайти є не лише веб-сторінками, але й веб-програмами. Звичайно, у цього браузера є певні переваги: це і легкість у керуванні і багатий функціонал і гарний інтерфейс і ще багато багато інших переваг.

За даними порталів зі статистикою веб-оглядачів, браузер Chrome займає абсолютну першість за популярністю у світі та випередивши наприклад Internet Explorer ще доволі давно, а саме в далекому дві тисячі дванадцятому році, а його частка у світі серед найпопулярніших веб-браузерів на всіх платформах вже в дві тисячі дев'ятнадцятому році склала близько шестидесяти відсотків. На другому місці браузер Safari з неймовірно великим відривом від першого місця і п'ятнадцятьма відсотками, а бронзову сходинку у топі найвикористованіших браузерів займає Firefox з часткою лише близько чотирьох відсотків ринку. Серед десктопних версій частка Chrome взагалі вражає і скаладає неймовірні сімдесят два відсотки. На другому місці за популярністю браузер Firefox з часткою вісім відсотків. Safari має десь два відсотки користувачів і займає третю сходинку. Але це світова статистика, а якими ж браузерами користуються в Україні?

В українському елементі інтернету серед усіх версій веб-браузерів станом на минулий рік, браузер Chrome використовували близько семи десяти відсотків користувачів. І, ми можемо побачити, що ця цифра не дуже то і відрізняється від світової статистики.

Браузер написаний на основі та заготовці вільного проекту Chromium і відрізняється багатьма корисними функціями: підтримкою Флеш плеєра прямо в браузері, вбудованою підтримкою PDF, інформуванням користувача, та іншими корисними функціями, які настільки припали до вподоби користувачам, що цей браузер став найпопулярнішим серед користувачів. Наприклад, я часто використовую цей браузер для перегляду сканів документів, або текстових документів одразу у браузері, що є неймовірно комфортним.

Якщо ж порівняти з усіма веб-браузерами, в Chrome окрема сторінка є окремою за розвитком сценарію. У випадках, де обробка контенту сторінки заблокується, його можна завершити взагалі без ризиків завершити основні частини між сторінками. Користувач не бачить таких змін, але вони дуже важливі.

Частина від коду браузера, чи основний код на якому був заснований браузер також захищений правами.

На початку розвитку браузера він отримав перевагу, що давала Google основне право на вільне використання, створення схожих сторінок, передачу і публікацію в комерційних цілях всієї передаваної через браузер інформації. Але потім, з ходом розвитку браузера та того як він набирал популярності, цю проблему треба було вирішувати.

Тим більш, що цей браузер вже настільки став цікавим для користувачів, що розробники просто не могли дати йому поглинути у проблемах з правами власника, або іншими не менш важливими для розробників проблемами.

Цікавим фактом про цей браузер,я можу розповісти те, що розробник спочатку не хотів боротись з іншими браузерами та вступати в війни, а тому не хотів відокремлювати свої розробки від основної технології та отримати власний самостійний браузер.Однак, його колеги самі зробили таке рішення,після чого голова правління Шмідт сказав: «вона була настільки хорошою, що я змінив свою думку». Це характеризує розробників, як сміливих людей.

По закінченню ста тридцяти діб роботи над публічним показником Гугл виклав в мережу власний документ свого оглядача і сталось це у грудні дві тисячі сьомого року.

А вже у грудні наступного року Google показав версію свого застосунку для інших операційних систем. Всього лише за рік компанія проробила величезний масив проблем та зробила неймовірним відрізок у розробці.

Перш за все Chrome був розроблений тільки під Microsoft Windows, а вже потім з'явилися дес- і бет-версії для операційних систем Linux і Mac OS X. Спочатку основи цих операційних систем не отримали звичного нам повного набору функцій з аналогічних бета-версій Windows (наприклад, закладок), але з появою гілки 5 усі функції стали однаковими для усіх застосунків.

У дві тисячі десятому році році в Chrome був показаний та перебрендований основний представницький малюнок, який замінив колишній великий та застарілий.

Chrome час від часу отримує поновлення з кількох списків (один для сайтів з низькою репутацією і один для сайтів, що включають програмне забезпечення, котре може нашкодити користувачу) і попереджає того, коли він намагається відвідати шкідливий сайт. Це захищає користувача від небажаних наслідків.

Chrome, як правило, створює для кожної сторінки окремий процес, щоб не допустити ситуації коли вміст однієї сторінки має можливість впливати на вміст іншої (також, в разі, якщо процес оформлення вмісту вкладки зависне, його можна буде завершити без ризику втрати даних інших вкладок). Процесу вкладки не може зчитувати або записувати інформацію. Образно висловлюючись, вкладка знаходиться ніби в пісочниці.

Як правило, плагіни (найпопулярніший відомий нам Adobe Flash Player), не є частиною оглядача і не може бути вбудованим в пісочницю. Наприкінці року Google сказав, що скоро Adobe Flash буде вбудований в браузер, що позбавить користувача від необхідності завантажувати і встановлювати його. Оновлюватися Flash Player буде також абсолютно не потребуючи дій від користувача у браузері. Починаючи з версії був доданий Adobe Flash Player.

При завантаженні виконуваного файлу спочатку необхідно підтвердити завантаження, натиснувши елемент «Збереження», який з'являється на верхній панелі завантажень. Це дозволяє запобігти

автоматичній завантаженні шпигунського програмного забезпечення на комп'ютер або іншій пристрій користувача. Часто розробники таких програм маскують кнопки, або роблять псевдо кнопки і шпигують за користувачами.

Застосунок звичайно дозволяє користувачеві поняття показу без збереження історії. Називається такий режим інкогніто. Сторінка, яку користувач буде продивлятися у своєму вільному користуванні браузером буде, звичайно залишена без збереження історії переглядів сайтів та інших додаткових файлів, які браузери зазвичай збирають у звичному режимі. Але, варто зазначити і те що програми, які користувач захоче завантажити залишаться у його історії завантажень та користувач матиме можливість їх передивлятися та встановлювати.

Типовий інтерфейс Google Chrome представляє собою кнопки: повернутися, наступна сторінка, оновити сторінку /припинити завантаження, додати в закладки, перейти на домашню сторінку (додому). Кнопки переходу на головну сторінку може і не бути, вона може бути відключена або відсутня.

Панель закладок є основним компонентом інтерфейсу і знаходиться у верхній частині вікна, а не під рядком адреси. Ця зміна виглядає досить контрастно та незвично в порівнянні з багатьма іншими браузерами. Вона легким перетягуванням на вільне місце може бути перенесена в окреме вікно (можливий також і зворотний варіант). Кожна вкладка має свій власний набір інструментів, у тому числі Омнібокс.

Омнібокс— це рядок, який знаходиться зверху кожної вкладки, він поєднує в собі адресний рядок і рядок пошуку. Омнібокс перенаправляє запит в систему пошуку в тому випадку, якщо адреса не відповідає вимогам написання – наприклад, не містить точку, імені чи назви, коректних даних, містить пропуски на вводу даних тощо. При введенні пошукової інформації Google, використовуючи функцію пошуку в Інтернеті, пропонує першу за списком адресу (наприклад, якщо Ви введете слово «Гугл» браузер самостійно запропонує варіант головної сторінки цієї пошукової системи). Також є можливість робити авто заповнення форм, на основі тих даних, які

браузер зібрав раніше з тих сайтів куди заходив користувач. Також можна викликати пошукову систему та одразу почати пошук з тої сторінки де зараз знаходиться користувач. Щоб це спрацювало потрібно ввести данні для тої пошукової системи, якою користувач хоже зкористатись саме зараз(наприклад, букву 'g' для google), потім натиснути клавішу для пошуку і ввести пошукову інформацію.

Mozilla Firefox – безкоштовний веб-оглядач з доступним кодом, який можна вільно скачати для особистого користування і використовує у собі ядро іншої системи.

Після випуску у дві тисячі четвертому році першої версії одержав високі оцінки від багатьох програмістів та користувачів. З більш ніж двадцятьма мільйонами завантажень з головного сайту в перші сто днів після випуску вже другої версії, Firefox став однією із найуживаніших програм із відкритим кодом. Дві тисячі п'ятого року (за триста тридцять два дні після випуску версії 1, майже через рік) Firefox було завантажено сто-мільйонний раз. На цей час Firefox має результат у користуванні вісім відсотків з-поміж основними браузерами і посідає наступну позицію після Google Chrome. Має гарні відгуки від користувачів та експертів і розробників та вважається одним з кращих веб-оглядачем. В Україні Firefox отримав четверте місце після Google Chrome та Opera за частотою користування з результатом трохи більше п'яти відсотків станом на вересень цього року. Влітку дві тиячі дев'ятого року кількість завантажень Firefox різних версій перетнула межу в цілий мільярд.

У дві тисячі дев'ятому році він був навіть другим найпопулярнішим браузером після Internet Explorer. Google Chrome вийшов трохи раніше та ще не встиг набрати достатньої популярності.

З тисяча дев'ятсот дев'яносто четвертого року браузер почав отримувати першу популярність. А до дві тисячі тринадцятого року Opera включала додаткові можливості для роботи в інтернеті (пошта та інші служби), але після переходу з власного браузерного двигуна на відкритий

двигун додаткові функції були виокремлені з браузера в супутні проекти. Коли я починав перші кроки роботи в інтернеті, я користувався саме цим браузером, і можу сказати що він є дуже легким у використанні новачками.

Браузер розроблений за допомогою мови програмування C(sic), перевагою якого є висока швидкість роботи, а також цей браузер підтримує усі веб-технології. Відмінністю від своїх аналогів Опера тривалий час була підтримка багатосторінкового перегляд, але потім ці переваги було нівельовано тим що такі ж можливості отримали і інші браузери. Також браузер має свою власну систему безпеки: вбудований захист від фішингу, що захищає користувача; додаткове кодування інформаційного потоку при роботі зі сторінками, що містять запити конфіденційних даних; можливість видалення куки-файлів, очищення історії відвідувань вибираючи певний період, а також панель користувача. Опера безкоштовна. Міжнародна версія браузера має арсенал з багатьох мов, включаючи українську мову.

Крім основних можливостей браузера, за час, розробки і розвитку браузеру, в нього були додані цікаві можливості переглядати пошту та новини, адресної книги, можливість скачування даних через Торент, перегляд завантажувальних, клієнтських даних, основних форм, помічника браузера, а також різні функції, що працюють поза вікнами браузера.

За допомогою різних інструментів та положень користувач може налаштувати інтерфейс за своїм смаком. Існує денна і нічна теми оформлення, користувач може налагодити на свій смак шпалери. Користувач може налаштувати шрифти та кольори для веб-сторінок, і навіть змінити ті налаштування, що прописані при розробці. Це може бути важливим, якщо колір чи стиль сторінок одразу був вибраний розробником невдалим, що повинно було бути зрозумілим. В браузері Опера, що дуже важливо, є можливість користування для людей із порушеннями зору або обмеженими рухами: присутнє голосове управління. Все це насправді вирізняє браузер з-поміж інших і надає йому конкурентні переваги. Для розробки треба знати всі основні можливості браузерів.

2.3 Функціональні вимоги до програмного забезпечення

Функціонування веб-сайту має бути побудовано на взаємодії адміністратора з користувачем.

Веб-сайт має забезпечувати виконання наступних функціональних вимог для підтримки взаємодії адміністратора з користувачем:

- можливість постійного оновлення наповнення сайту;
- перегляд новин;
- редагування новин адміністратором;
- перегляд турнірних таблиць;
- перегляд статистики команд;
- перегляд статистики гравців;
- можливість оцінити відео найкращих голів;
- перегляд переліку робіт;
- можливість отримання зворотнього зв'язку адміністратору від користувача.

Адміністратор веб-сайту (рис. 2.1) може додавати та оновлювати новини про будь-які футбольні команди на сайті, розробляти меню користувача, додавати до нього нові елементи та отримувати оцінки від користувача (рис.2.2), якщо той забажає її надати.

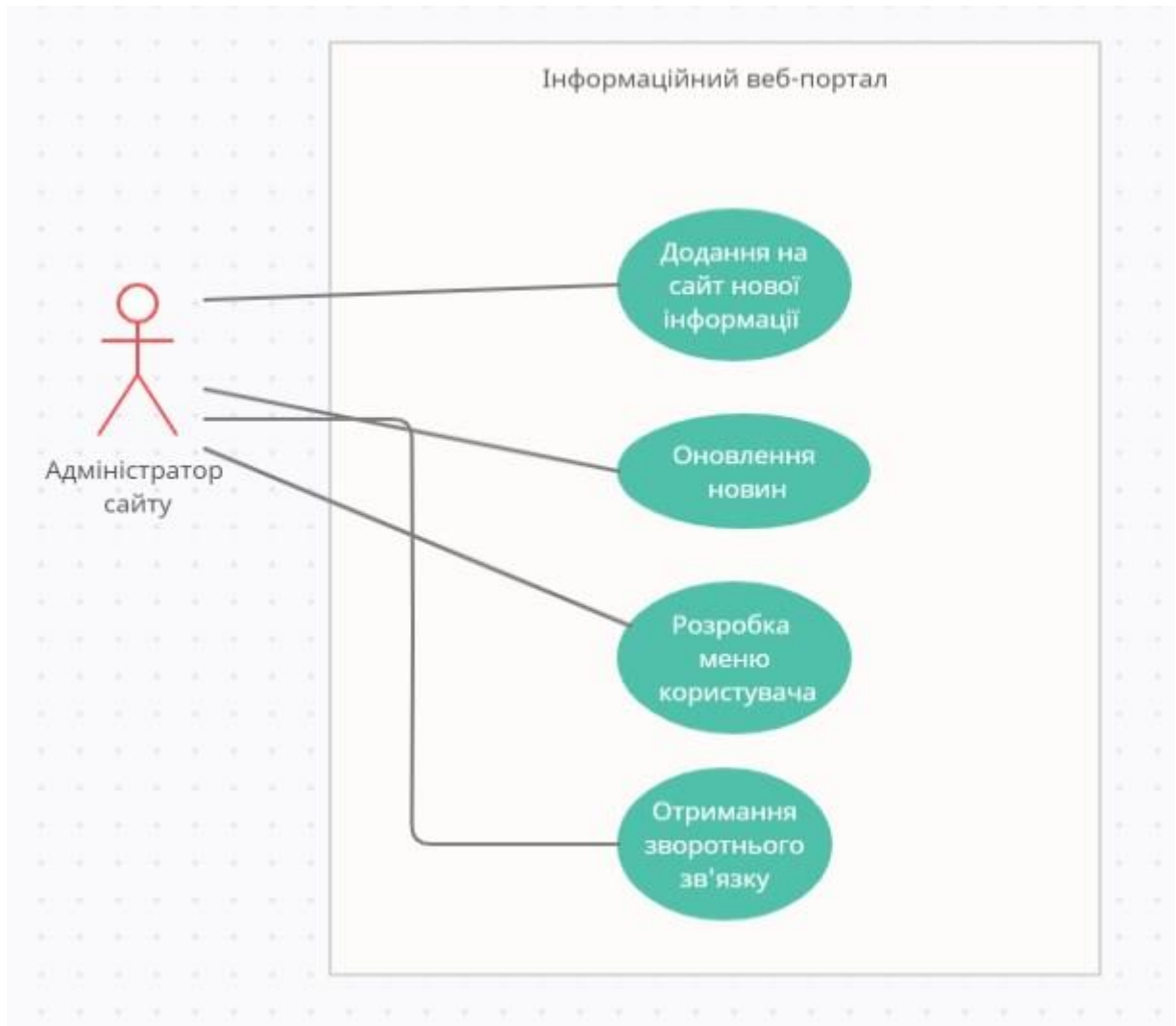


Рисунок 2.1 – Діаграма прецедентів адміністратора сайту

Під отриманням зворотнього зв'язку адміністратору від користувача маємо на увазі розробку спеціальної форми для адміністратора сайту, яким він може скористатись щоб отримати відгук або оцінку та покращити інтерфейс, або наповнення новинами веб-порталу для більш комфортного користування.

Користувач зможе скористатись цією можливістю для того щоб дати свою оцінку для веб-порталу, його наповнення та інтерфейсу, якщо забажає. Користувач (рис. 2.2) може оцінювати веб-сайт та надавати пропозиції з покращення веб-порталу, отримувати нові та актуальні футбольні новини, аналізувати гру команд чи окремих гравців на основі статистичних даних приведених у турнірних таблицях.

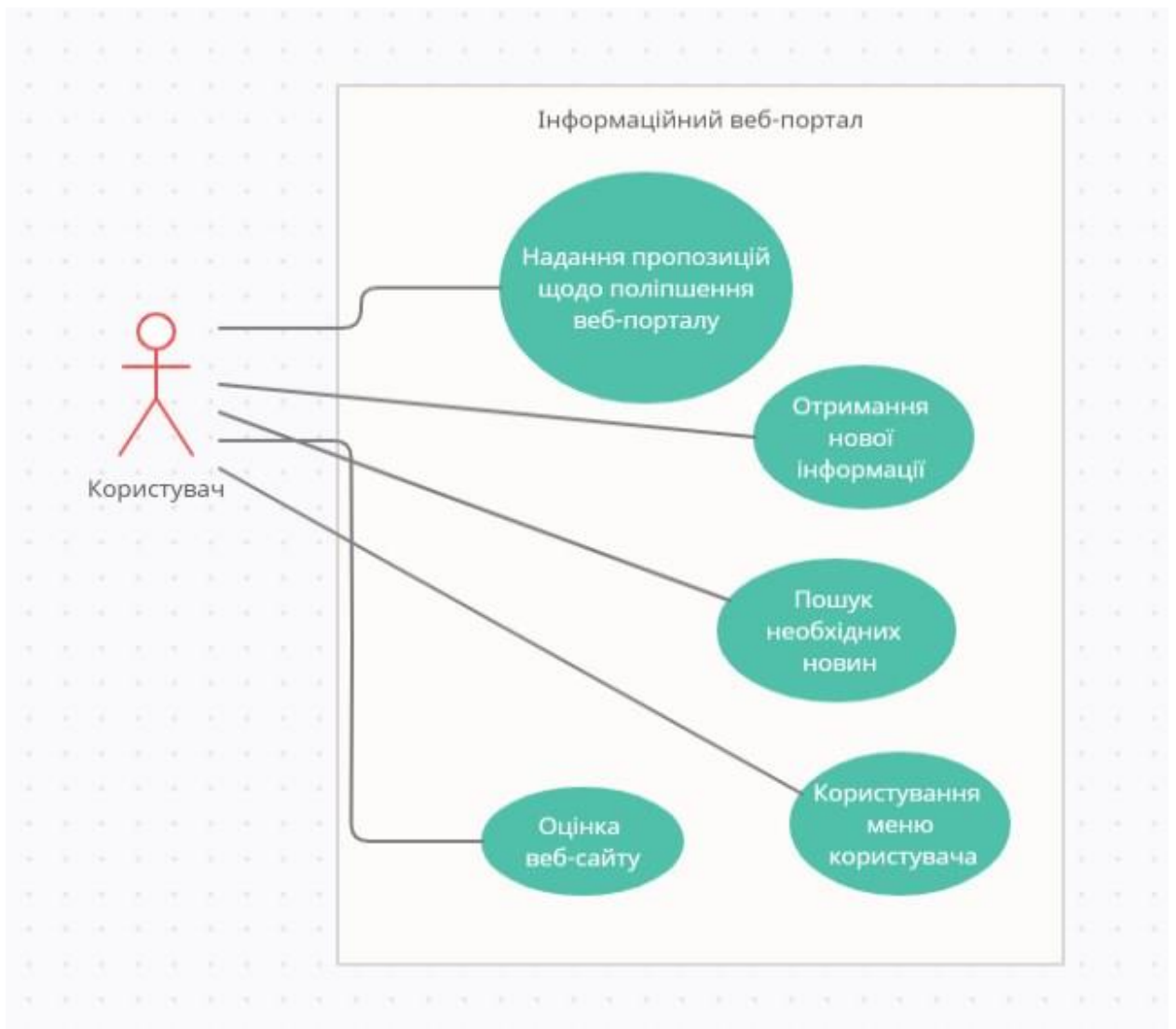


Рисунок 2.2 – Діаграма прецедентів користувача

2.4 Висновки за розділом 2

Для реалізації програмного забезпечення обрано мови програмування HTML, CSS, JavaScript що при використанні їх разом надає розробнику найбільше переваг для виконання тих чи інших функцій. Виконано моделювання функціональних вимог до веб-сайту на основі діаграм прецедентів.

З РОЗРОБКА ПРОГРАМИ

3.1 Порядок функціонування програми

До розробки архітектури сайту я підійшов з розумінням того, що інтерфейс користувача повинен бути доволі простим, але містити у собі деякі інтерактивні елементи. Для головної сторінки свого сайту я вибрав формат блочної верстки сайту. Я розробив три блоки з основною інформацією та два блоки – зверху та знизу, для додаткової інформації. Один з трьох блоків я виділив для найбільш важливої інформації що показано цифрою 5 (рис.3.1), другий блок – для меню користувача що показано цифрою 4 (рис.3.1), а третій для інтерактивного елементу, що показано цифрою 3 (рис.3.1). Два блоки, зверху і знизу, я доповнив логотипом сайту та основною інформацією про розробника та рік розробки веб-сайту. Я вважаю, що така архітектура сайту, буде найбільш комфортною для користувача і найліпшою саме для такої тематики.

Меню користувача я розробив на одному з найбільш видимих місць для користувача, для того щоб було зручно шукати інформацію. Також меню є інтерактивним, тобто, якщо навести на один з блоків меню – він підсвічується що показано цифрою 4 (рис. 3.1).

Це зроблено для того щоб кожен користувач, шукаючи необхідну інформацію в пункті меню користувача міг зорієнтуватись на те, на якій з сторінок меню він знаходиться саме зараз. Я думаю, що таке меню користувача є найбільш комфортним.

На рис.3.1, також позначено цифрою 1 логотип веб-сайту.

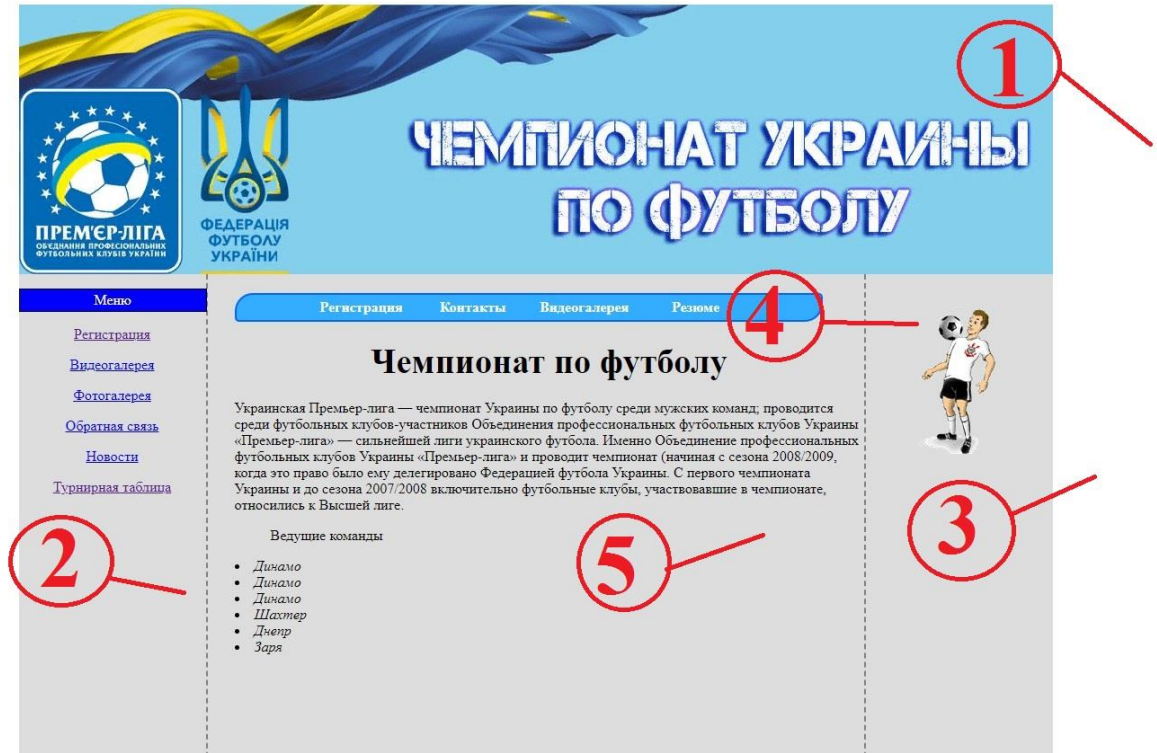


Рисунок 3.1 – Шаблон головної сторінки

3.2 Опис реалізації програми

Для побудови структури сайту використано блокову розмітку та розмітку на декілька розділів.

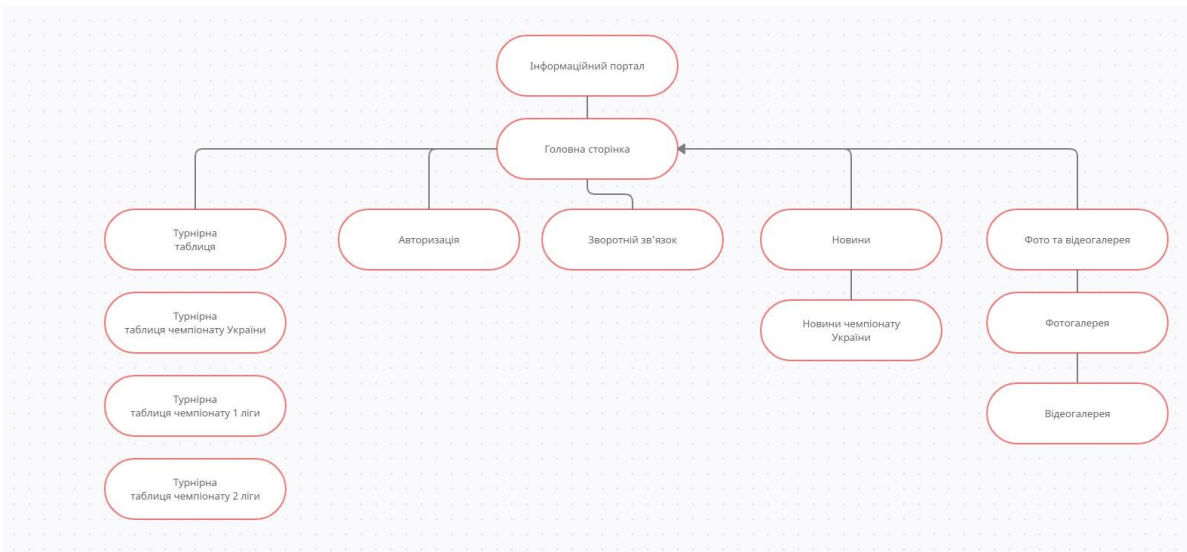


Рисунок 3.2 – Структурна схема сайту

Для початку роботи над створенням веб-сайту потрібно вибрати гарну та зручну середу розробки. Звичайно, HTML – код можна писати навіть у звичайному текстовому документі, але це не зручно. Для розробки веб-сайту я вибрав середу розробки SublimeText. Перевагами цієї середи розробки є можливість автоматичного до заповнення та закриття тегів, а також їх підсвітка.

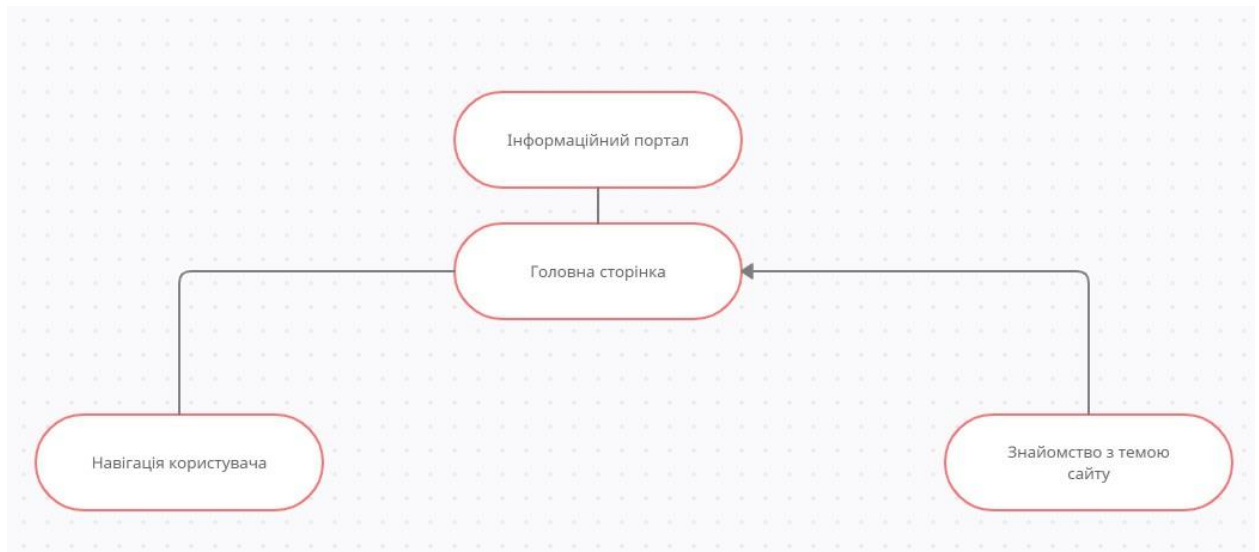


Рисунок 3.3 –Функціональна схема головної сторінки

```

<html>
<meta charset="utf">
<head>
<link rel="shortcut icon" href="/images/4.jpg" type="image/x-icon">
  
```

Я надав веб-пошуковцям основну інформацію про мою сторінку, а саме – мову, якою буде описано мій сайт, а також кодування сторінки. Наступним кроком я додав завбачно підготовлений мною логотип сайту, який я розробив у Adobe Photoshop. Необхідний файл я розмістив в папці images, а потім підключив з нього файл під назвою 4.jpg:

```

<title>
Веб-портал “Футбольна Україна”
</title>
<style type="text/css">
  
```

Тут я додав ще трохи інформації для пошукових систем та користувачів про те, про що моя сторінка і за допомогою тегу <title> я визначив заголовок, який буде показано на моєму сайті – Веб-портал “Футбольна Україна”. Потім я підключив файл під назвою text/css.

У файлі під назвою text/css я, за допомогою наступного коду:

```
div#block {width:1150px; margin:0 auto; background-color:#dddddd}
```

Так як, я ще на стадії проектування сайту вибрав блокову верстку сторінки, то мені потрібно було оголосити ці блоки за допомогою певних процедур. Оголосив перший, основний блок на моїй сторінці, з якого я буду будувати усі інші блоки та поставив розміри які він повинен мати, а саме ширину блоку 1150 пікселів, автоматичний відступ 0 пікселів та фоновий колір, для якого встановив значення: #dddddd:

```
div.header {width:1150px; height:300px; }
div.left_col {width:208px; height:1150px; float:left; border-right:2px
dashed grey; margin-bottom:-15px}
div.center_col {width:700px; float:left;margin-left:30px; }
div.right_col {width:58px; height:1150px; float:left; border-left:2px dashed
grey; margin-bottom:-15px}
div.footer {width:1150px; height:100px; background-color:#717dc9;
clear:both}

$(document).ready(function(){
    $("#menu").click(function(){ $("#list").slideToggle(2000)});
    $("#menu").toggle(function(){
        $("#img").attr("src","menudown.gif"), function(){
            $("#img").attr("src","menuup.gif")
        });
    });
```

```
$("#menu").mouseover(function(){          $("#menu").css("background-
color","#01939a"));
```

```
$("#menu").mouseout(function(){           $("#menu").css("background-
color","#006064"));
```

```
<p align="justify">
```

```
<big><h1><center>Чемпионат                по
футболу</center></h1></big><p>Украинская  Премьер-лига  – чемпионат
Украины по футболу среди команд; проводится среди футбольных клубов-
участников Объединения футбольных клубов Украины «Премьер-лига» –
сильнейшей лиги украинского футбола. Именно Объединение
профессиональных футбольных клубов Украины «Премьер-лига» и проводит
чемпионат начиная с сезона 2008/2009, когда это право было ему
делегировано Федерацией футбола Украины. С первого чемпионата Украины
и до сезона 2007/2008 включительно футбольные клубы, участвовавшие в
чемпионате, относились к Высшей лиге. </p>
```

Далі ми прописуємо за допомогою закриваючого тегу </h1> великий заголовок для нашої сторінки, а для наступного тексту додаємо тег </p>, налаштування до якого ми вже прописали раніше, щоб увесь текст було вирівняно по ширині сторінки:

```
<p><ul>Ведущие команды</ul>
    <li><em> Динамо</li></em>
    <li><em> Динамо</li></em>
    <li><em> Динамо</li></em>
    <li><em> Шахтер</li></em>
    <li><em> Днепр</li></em>
    <li><em> Заря</li></em>
</h4>
</div>
```

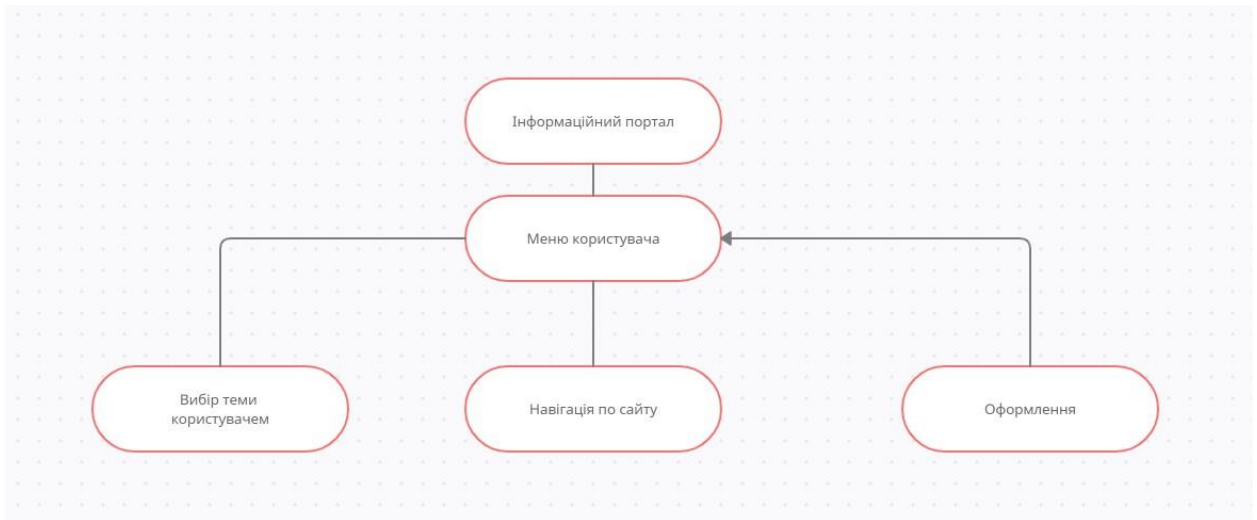


Рисунок 3.4 – Функціональна схема меню користувача

```

<p align="center"><a href="1.html">Регистрация </a><p align="center"><a href="2.html">Видеогалерея </a><p align="center"><a href="3.html">Фотогалерея </a><p align="center"><a href="4.html">Обратная связь </a><p align="center"><a href="5.html">Новости </a><p align="center"><a href="6.html">Турнирная таблица</a></div>
  
```

```

<div class="center_col">
<h2 align="center"></h2>
<h4 align="center">
  
```

```

<ul id="navbar">
<li><a href="1.html">Регистрация</a></li>
<li><a href="4.html">Контакты</a></li>
<li><a href="2.html">Видеогалерея</a></li>
<li><a href="7.html">Резюме</a></li>
</ul>
  
```

Наступним кодом я почав розроблювати інтерактивне меню для користувача сторінки. Спочатку я хотів розробити найбільш легке для

розуміння користувача меню, але потім мені здалося що меню можна зробити і простим і інтерактивним водночас. Саме цю задачу я надалі вирішив реалізувати у ході розробки свого веб-сайту. Для того щоб розробити інтерактивне меню мені потрібно було використати усі найбільш сильні сторони мов програмування які я сьогодні використовував. Для початку я підключив документи до сторінки за допомогою тегу <a href>.

```
<style>
#navbar {
    margin: 0;
    padding: 0;
    list-style-type: none;
}
```

Потім я почав оформлювати стилі меню:

```
#navbar li { display: inline; }

#navbar {
    margin: 0;
    padding: 0;
    list-style-type: none;
    border: 2px solid #0066FF;
    border-radius: 20px 5px;
    width: 650px;
    text-align: center;
    background-color: #33ADFF;
}
```

Спочатку я прописав стилі для статичної форми меню, з якою користувач не взаємодіє. Відступи я поставив в 0, зробив список невиспадаючим, оформив краї за допомогою властивості border та встановив колір, який на мою думку вдало вписується до кольорової гамми усього веб-сайту:

```
#navbar a {  
    color: #fff;  
    padding: 5px 10px;  
    text-decoration: none;  
    font-weight: bold;  
    display: inline-block;  
    width: 100px;  
}
```

Наступним кроком було описати, які ж стилі будуть примінені до меню, з яким користувач почне взаємодіяти. Тут я встановив налаштування щоб краї блоку були ледве зкругленими:

```
#navbar a:hover {  
    border-radius: 20px 5px;  
    background-color: #0066FF;  
}
```

За допомогою цього великого блоку коду я зміг описати кожен з властивостей меню, відступ я виставив в 0, борти меню я зробив світло-синього кольору та розміром 2 пікселі. Зазначив щоб текст у меню розташовувався саме по центру.

Для того щоб меню стало інтерактивним та при наведенні користувача або кліком мишею стиль меню змінювався я додав меню

властивість `a:hover` і зазначив в ній на який саме колір повинно змінюватись меню при наведенні користувача на блок меню. Результатом я залишився задоволений:

```
function show2(){
  if (!document.all&&!document.getElementById)
    return
  thelement=document.getElementById?
document.getElementById("tick2"): document.all.tick2
  var Digital=new Date()
  var hours=Digital.getHours()
  var minutes=Digital.getMinutes()
  var seconds=Digital.getSeconds()
  var dn="PM"
  if (hours<12)
    dn="AM"
  if (hours>12)
    hours=hours-12
  if (hours==0)
    hours=12
  if (minutes<=9)
    minutes="0"+minutes
  if (seconds<=9)
    seconds="0"+seconds
  var ctime=hours+":"+minutes+":"+seconds+" "+dn
  thelement.innerHTML="<b size:16;color:blue;>"+ctime+"</b>"
  setTimeout("show2()",1000)
}
window.onload=show2
```

Далі за допомогою мови програмування JS я намагався розробити годинник для свого веб-сайту, але нажаль мені не вдалося його вдало розмістити його серед блоків тексту. По задумці, годинник мав би мати дві системи з показами часу – ранковим та вечірнім.



Рисунок 3.5 – Функціональна схема меню користувача

```
<input type='checkbox' name='space' value='2' />
```

Я розробив декілька текстових форм для того щоб користувач, який буде вводити інформацію для реєстрації зміг найбільш вдало ввести всю необхідну інформацію. Також я розробив декілька типів полів, в яких можна вводити лише текстові данні, а також вибирати серед уже введених даних адміністратором сайту, а також можливості вибору одного з декількох варіантів (радіобаттон) та множинного вибору (чек бокс):

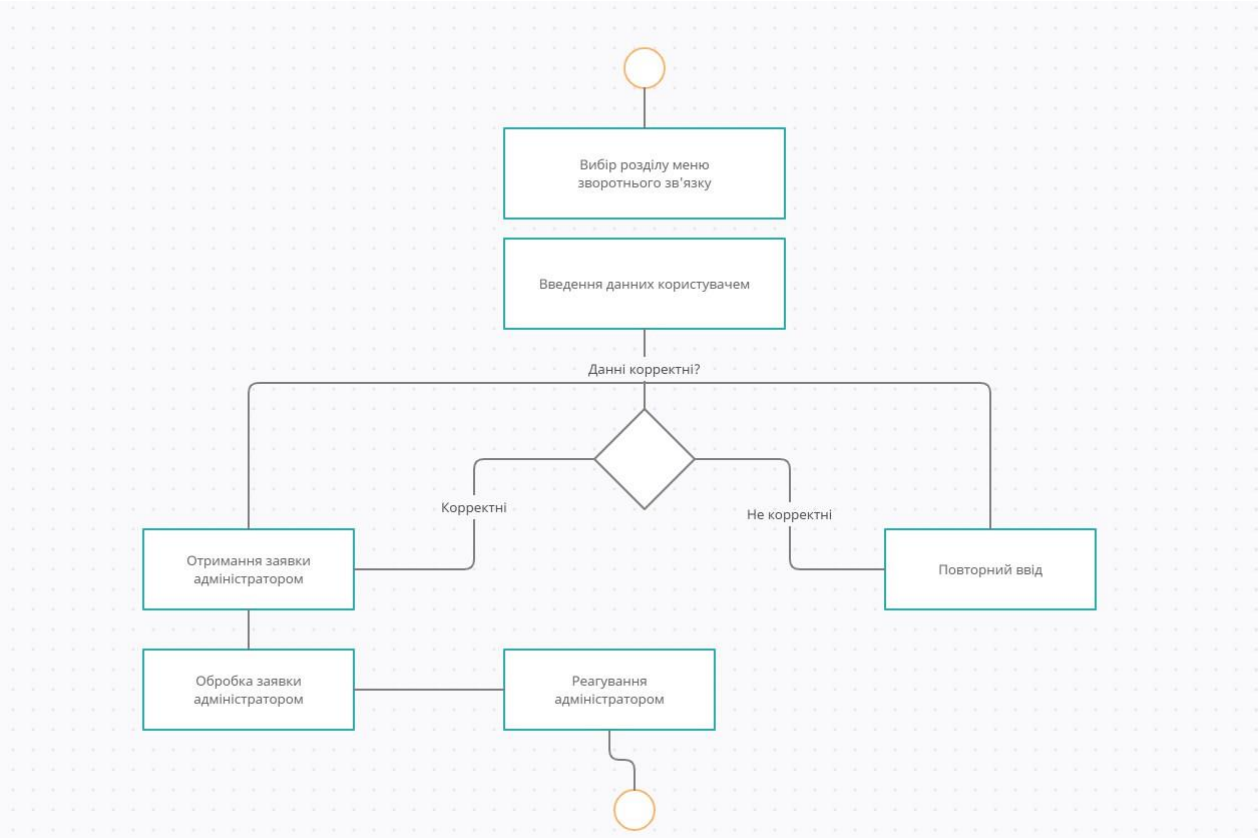


Рисунок 3.6 – Функціональна схема події залишення відгуку

```
table {  
  width: 100%;  
  border-collapse: collapse;  
}
```

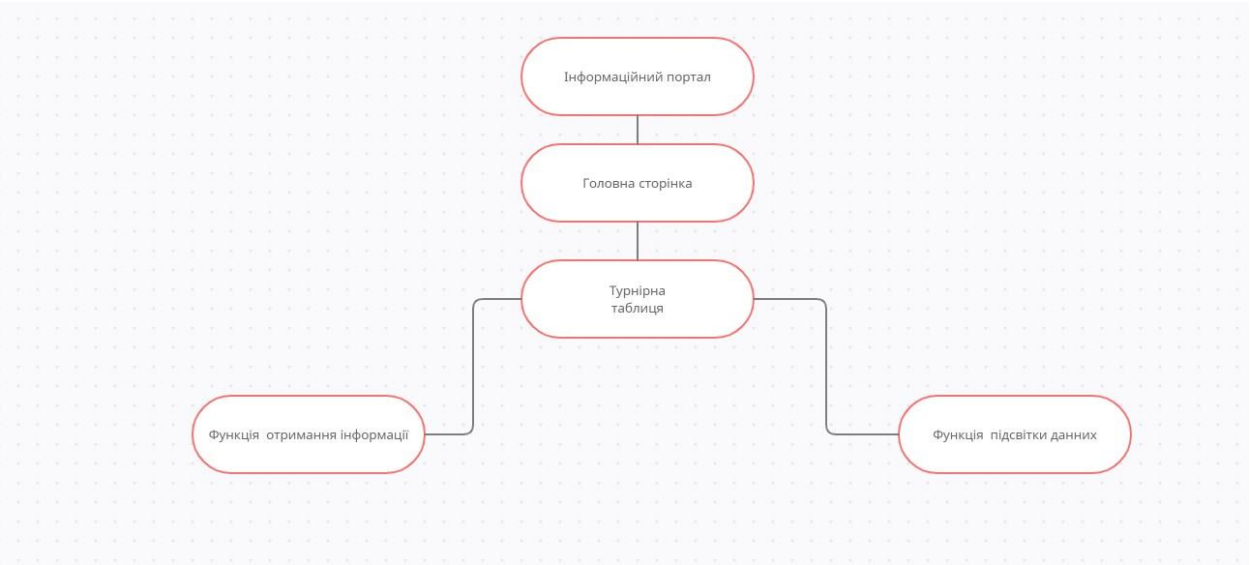


Рисунок 3.7 – Функціональна схема турнірної таблиці

Цей код створює таблицю з можливістю наповнення її текстовими даними:

```
tbody tr:hover {  
    background: #f3bd48;  
    color: #fff;  
}
```

Обов'язково прописуємо спеціальний режим стилів для таблиці з даними, якщо користувач буде взаємодіяти ними. Цим кодом, я позначив, що користувач який наведе на певну строку даних отримає візуальний ефект, а саме – підсвітки жовтим кольором.

Таку функцію я розробив для поліпшення користування таблицями з текстовими даними на моєму сайті:

```
<style>  
    .fig {  
        text-align: center; /* Выравнивание по центру */  
    }  
</style>
```

Наступним кодом я вирівняв усі дані по центру, щоб текстові данні у таблиці були вирівняні по центрі кожної лунки. Після цього я почав розробку сторінки з резюме, для того щоб розробник або адміністратор сайту мали можливість познайомити користувача з тим, хто саме взаємодіє з ними по ту сторону екрану:

<body style="color:#000000; background-color:#AFEEEE">

1. Introduction 2. Background 3. Methodology 4. Results 5. Conclusion 6. References 7. Appendix 8. Notes 9. Tables 10. Figures
--

 |

1.Иван Иванов

2. 01.02.1993

3.ул домовая 22\35

4.276-35-65

<p>5.ivanivanov@gmail.com</p>

Образование	

с	по	название
---	----	----------

заведения	специальность	аттестация
-----------	---------------	------------

| 2013 | 2017 | ЗНТУ | Программист |
 5 |

Для кожної з таблиць окремо я прописав висоту, ширину та розмір “бортів”. Перша таблиця розкаже користувачеві де навчався розробник чи адміністратор:

```
<table width=700px border="1"align=center>
  <thead><th colspan="2">Опыт работы</th></thead>
  <tr><td>с</td><td>по</td><td>название
заведения</td><td>должность</td><td>обязанности </td></tr>
  <tr><td>2017</td><td>2019</td><td>ЗНТУ</td><td>Преподаватель
</td><td>Учить студентов </td></tr>
```

Наступну таблицю я розробив з метою розказати про те де працював та які були обов’язки у адміністратора або розробника:

```
<table width=700px border="1"align=center>
  <thead><th colspan="2">Проекты</th></thead>
  <tr><td>Название</td><td>Доля в участии
</td><td>Описание</td></tr>
  <tr><td>Сайт</td><td>максимальная</td><td>Сайт на тему чемпионат
украины по футболу</td></tr>
```

Наступний пункт імпровізованого резюме – проекти, якими займався розробник чи адміністратор, та яка була його роль у тому чи іншому проекті.

Щодо налаштувань самої таблиці усі вони залишаються як і в першій таблиці резюме:

```
<table width=700px border="1"align=center>
  <thead><th colspan="3">Программы для работы в
графикой</th></thead>
```

```

<tr><td>Adobe Photoshop</td><td>+</td><td>-</td></tr>
<tr><td>Adobe Illustrator</td><td>+</td><td>-</td></tr>
<tr><td>Macromedia Flash</td><td>+</td><td>-</td></tr>
<tr><td>HTML</td><td>+</td><td>-</td></tr>
</tr>
<thead><th colspan="3">Офисные пакеты</th></thead>
<tr><td></td><td>+</td><td>-</td></tr>
</tr>
<thead><th colspan="3">Языки программирования и
разметки</th></thead>
<tr><td>HTML</td><td>+</td><td>-</td></tr>
</tr>
<tr><td>CSS</td><td>+</td><td>-</td></tr>
</tr>
<tr><td>DHTML</td><td>+</td><td>-</td></tr>
</tr>
<tr><td>XML</td><td>+</td><td>-</td></tr>
</tr>
<tr><td>JavaScript</td><td>+</td><td>-</td></tr>
<tr><td>PHP</td><td>+</td><td>-</td></tr>
</tr>
<tr><td>Perl</td><td>+</td><td>-</td></tr>
</tr>
<tr><td>C/C++</td><td>+</td><td>-</td></tr>
</tr>
<tr><td>ActionScript</td><td>+</td><td>-</td></tr>

```

Наступна таблиця має вже інші налаштування. По перше тегом `<th colspan="3">` встановлено, що в цій таблиці буде три лінії даних. З цієї таблиці користувач зможе зрозуміти якими навичками та мовами

програмування володіє розробник або адміністратор. Також ці таблиці можуть використовуватись під інший об'єм даних.

Також я розробив спеціальну можливість для користувача, щоб той міг відправити відгук про користування сайтом, і для того щоб уникнути спаму для адміністратору сайту, я розробив систему, яка перевіряє данні у режимі вводу користувача та, якщо користувач ввів некоректні данні не відправляє їх, а просить ввести нові:

```
<div class="footer"><p>© 2021</p></div>
```

В кінці розробки я оформив останній блок коду, де додав рік розробки веб-порталу. Таким чином ми можемо зрозуміти коли був розроблений сайт та які цілі стояли перед розробником.

3.3 Висновки за розділом 3

Розробку програмного забезпечення виконано на основі мов програмування HTML (Hyper Text Markup Language – мова розмітки гіпертексту) та CSS (Cascading Style Sheets, Каскадні таблиці стилів) та мову програмування JavaScript. Описав порядок функціонування програми та її поетапну розробку.

4 ЕКСПЛУАТАЦІЯ ТА ТЕСТУВАННЯ ПРОГРАМИ

4.1 Призначення програми

Програма призначена для отримання користувачами в Інтернеті необхідної тематичної інформації, пошуку необхідної інформації.

4.2 Умови виконання програми

Сторона користувача має відповідати таким вимогам:

- процесор: 2 ядра;
- оперативна пам'ять – 1 Гб;
- вільний доступ до мережі Інтернет;
- жорсткий диск – 1 Гб вільного простору.

4.3 Робота з програмою

Якщо користувач матиме вільний доступ до мережі Інтернет, то проблем з доступом до головної сторінки не повинно бути. Мною було проведено певний аналіз аналогів, на предмет того, які функції мають ту чи іншу важливість для користувача.

Після проведеного аналізу мною розроблений веб-сайт, де для кожного користувача надається весь спектр можливих функцій, при чому доступ до них повинен бути вільним, щоб люди з різним рівнем комп'ютерної грамотності могли ними користуватись.

При розробці головної сторінки, я, звичайно, намагався користуватись основною метою розробки веб-порталу – це комфорт користувача. Меню користувача ми можемо побачити на головній сторінці (рис. 4.1). Меню є інтерактивним, та при наведенні на один з елементів меню, воно змінює колір та з'являються характерні борти. Меню включає в себе такі пункти як: Контакти, Відеогалерея та Резюме. Користувач може вибрати кожен з

елементів. Зліва ми також бачимо додаткове меню, яке є допоміжним для користувача, та має декілька унікальних функцій, як, наприклад, Зворотній зв'язок, Турнірне становище команд та Фотогалерея. Додаткове меню потрібне на сторінці для того, щоб винести допоміжні функції, які не відносяться до основних. Зправа на сторінці ми можемо побачити інтерактивний елемент, який є по-перше тематичним для сайту, а по-друге надає сторінці більшої інтерактивності та жвавості. Зверху можна побачити логотип сайту.

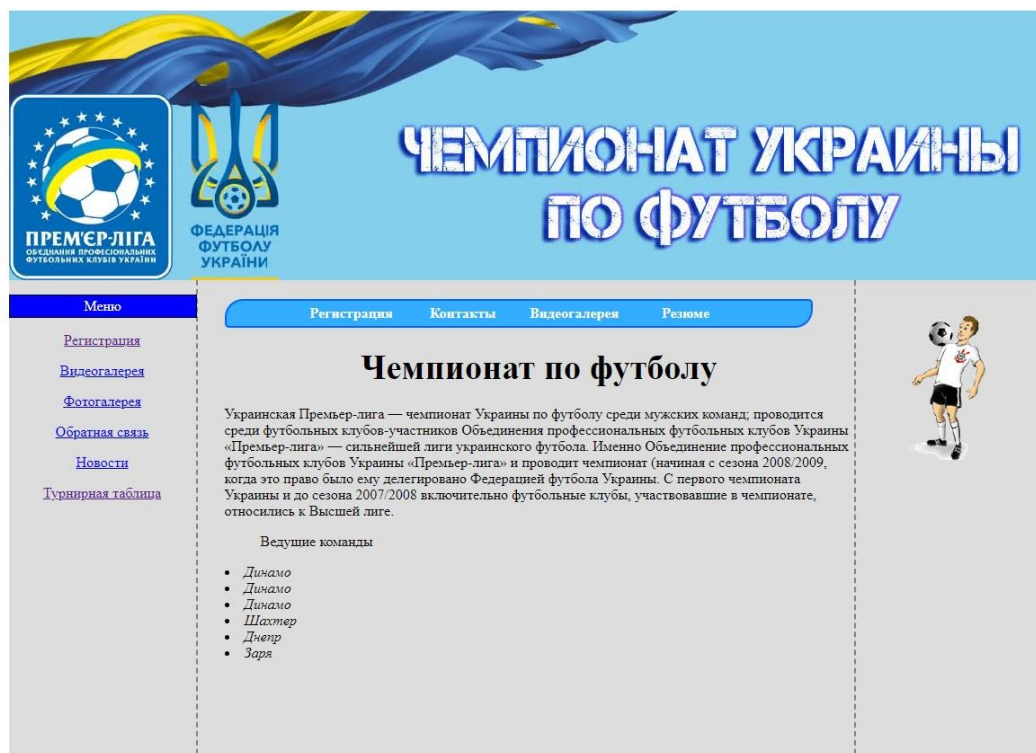


Рисунок 4.1 – Головна сторінка

На наступному рисунку (рис. 4.2) ми бачимо форму вводу даних про користувача. Вона розмічена для простоти вводу, а також включає текстові поля для вводу Ім'я, Прізвища та По-батькові користувача. Також є можливість, яка не є обов'язковою вказати стать користувача. І останнім ми можемо побачити форму для множинного вводу (чек бокс) який допомагатиме адміністратору дізнатися про те звідки користувач дізнався

про наш веб-портал. Це буде важливо для адміністратора сайту та розуміння звідки користувач дізнався про веб-сайт.

Даные о пользователе:

Имя:

Фамилия:

Отчество:

Укажите ваш пол:

☐ Мужской

☐ Женский

откуда вы узнали о нашем сайте?:

☐ Из рекламы

☐ Из интернета

☐ Другие источники

Рисунок 4.2 – Форма вводу даних про користувача

На сторінці відправки відгуку (рис. 4.3) ми можемо побачити поля для введення Імені, Пошти, Номеру телефону, та Відгуку. Хотілось би зазначити, що всі з цих полів, для того щоб уникнути спаму перевіряються на правильність та коректність введення даних. Якщо данні введені неправильно відгук відправити буде неможливо. В майбутньому хотілось би розглянути можливість підключення бази даних та прив'язки відгуків до конкретних користувачів з історією їх користування веб-порталом.

**ЧЕМПИОНАТ УКРАИНЫ
ПО ФУТБОЛУ**

Имя:

Email:

Телефон:

Сайт:

Причина вашего отзыва:

Рисунок 4.3 – Перегляд сторінки відправки відгуку

Наступною ми можемо побачити (рис. 4.4) можливість користувача, при підключенні до мережі інтернет передивлятися найновіші відео. Ця сторінка отримала назву відео галерея. Всі відеоролики підключені напряду з сервісу Youtube. Користувач має можливість, як передивлятися відео на розробленому веб-сайті так і переходити на сторонні сайти.

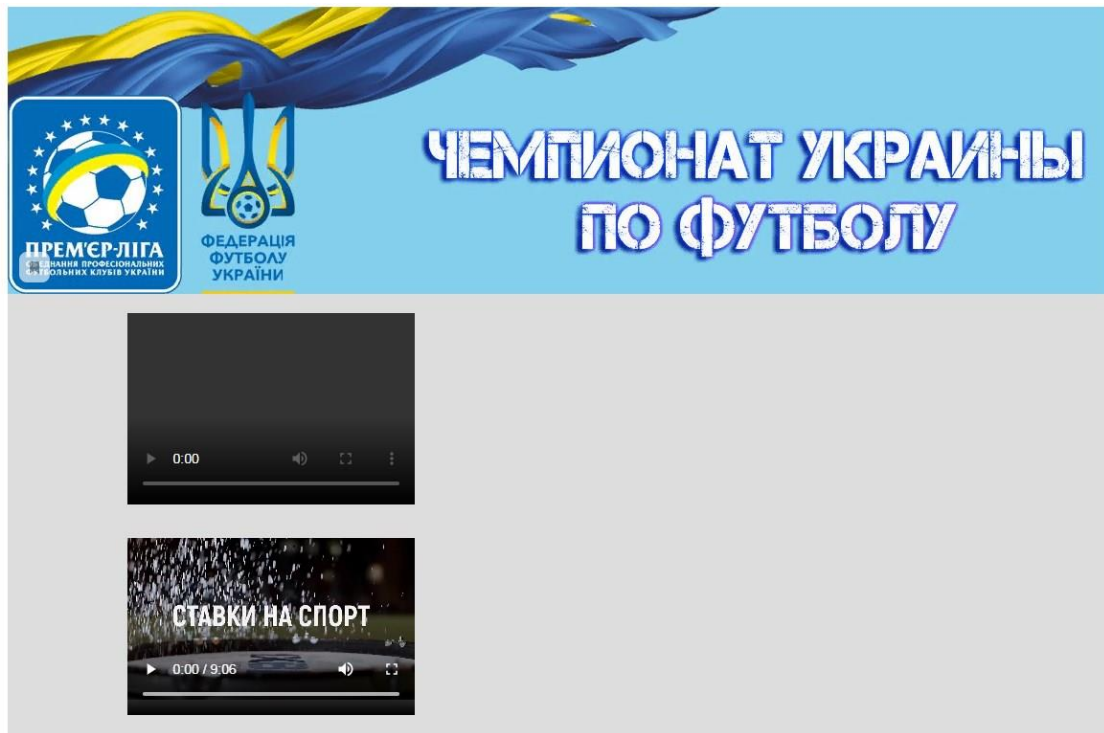


Рисунок 4.4 – Відеогалерея

Наступною можливістю користувача є можливість подивитися статистичні данні одно з чемпіонатів по футболу (рис. 4.5). Як я вже зазначав в минулих пунктах, я розробляв веб-сайт з думкою про комфорт користувача, тому для того щоб статистичні данні було комфортно передивлятися, аналізувати та порівнювати я оформив їх у відповідну таблицю даних. Також я передбачив можливість підсвітки кожної з ліній таблиці для того щоб користувач не міг переплутати одні чи іншу данні при аналізі даних таблиць. Таблиця з даними також має властивість оновлювання даних. Тобто, якщо з часом інформація перестане бути актуальною адміністратор

буде мати можливість оновити данні або додати нові. При цьому чарунки тої чи іншої таблиці даних є доволі широкими і мають можливість містити в собі велику кількість даних.



**ЧЕМПИОНАТ УКРАИНЫ
ПО ФУТБОЛУ**

Турнирная таблица чемпионата Украины

	Забитые голы	Пропущенные голы	Очки
1. Шахтер	87	21	79
2. Динамо	83	34	76
3. Зоря	63	31	75
4. Ворскла	83	51	66
Олимпик	44	34	63
5. Александрия	49	53	48
6. Черноморец	23	51	43
7. Зирка К	18	34	23
8. Сталь К	9	87	6

Рисунок 4.5 – Таблица статистичних даних

Також можливістю користувача є дізнатись резюме або розробника, або людини, яка адмініструє сайт. Це теж є доволі важливим елементом сторінки, тому що людина, яка переглядатиме веб-сторінку матиме уяву про освіту, вік та навички людини, що займалася розробкою цієї людини. Данні які можна побачити в таблиці з резюме повинні бути актуальними, тому таблиця має можливість постійно оновлюватись та доповнюватись. Чарунки таблиці спеціально налаштовані таким чином, щоб якщо нова інформація додана до таблиці перевищувала ліміт наданого для цього місця то чарунка автоматично збільшувалась.

ВИСНОВКИ

Переді мною стояло завдання дослідження процесу отримання користувачем футбольних новин та статистичних даних про результати футбольних команд. Предметом дослідження був веб-сайт, що повинен надавати останні новини зі світу футболу, дозволяє переглядати турнірну таблицю чемпіонату України з футболу, а також фото та відео найцікавіших подій або найкрасивіших голів.

Метою роботи було створення веб-сайту, на якому користувач може дізнатись новини з футбольного світу та слідкувати за результатами улюбленої команди.

Для того щоб виконати поставлену мету я застосовував такі матеріали методи та технічні засоби: мови програмування HTML, JavaScript та CSS, персональний комп'ютер з процесором Intel Core 4 під управлінням операційної системи Microsoft Windows 7, середа розробки – редактор коду SublimeText. Результатом розробки став веб-сайт, що має можливість надавати користувачам постійно оновлювану інформацію щодо останніх змін у футболі, трансферів гравців, турнірних таблиць, кількості забитих голів та іншої футбольної інформації. Сайт може стати корисним для кожного любителя футболу.

Я вважаю що усі завдання дипломної кваліфікаційної роботи повністю виконано.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Мархвида, И. В. Основы WEB-сторінок : HTML, CSS, JavaScript / И. В. Мархвида. – Мн. : Новое знание, 2002. – 352 с.
2. Петюшкин, А. В. HTML. Экспресс-курс / А. В. Петюшкин. – СПб : БХВ – Петербург, 2003. – 256 с.
3. Храмцов, П. Б. Основні Web-технології / П. Б. Храмцов, С. А. Брик, А. М. Русак, А. И. Сурин : под редакцией П. Б. Храмцова. – М. : ИНТУИТ.РУ, 2003. – 512с.
4. Холл, Марти Програмування для Web.Бібліотек майстра: пер. с англ. / Марти Холл, Лэрри Браун. – М. : Издательский дом «Вильямс», 2002. – 1264 с.
5. Гилмор, В. PHP 4 : учебный курс / В. Гилмор. – СПб : Питер, 2001. – 352 с.
6. Томсон, Л. Розробка Web-застосунків на PHP и MySQL : пер. с англ. / Л. Томсон, Л. Веллинг. – 2-е изд., испр. – СПб. : ООО "ДиаСофтЮП", 2003. – 672 с.
7. Хаген, Граф Макети веб-сайтів за допомогою Joomla! : пер. с англ. / Граф Хаген. – М. : Издательский дом «Вильямс», 2008. – 304 с.
8. J2EE. Розробка бізнес-застосунків: пер. с англ. / [П. Аллен, Дж. Бамбара, М. Ашнаульт и др.]. – СПб. : ДиаСофт, 2002. – 348с.
9. Ратшиллер, Т. PHP4 : Розробка Web-застосунків. Библиотека программиста / Т. Ратшиллер, Т. Геркен. – СПб : Питер, 2001. – 384 с.
10. ASP, NET : советы, рекомендации, примеры : пер. с англ. / [С. Митчелл, С. Уолтер, Д. Севен и др.]. – М. : Вильямс, 2002. – 864 с.
11. Дюбуа, Поль. MySQL и Perl в Web-застосунках : пер. с англ. / Поль Дюбуа. – М. : Издательский дом "Вильямс", 2002. – 480 с.
12. Старыгин, А. А. XML: розробка Web-застосунків / А. А. Старыгин. – СПб : БХВ-Петербург, 2003. – 592 с.
13. Хабибуллин, И. Розробка WEB-служб за допомогою Java.–

СПб. : БХВ-Петербург, 2003. – 400с.

14. Методичні вказівки до виконання лабораторних робіт з дисципліни “Сучасні інтернет-технології” в першому модулі для студентів спеціальностей 7.091501 “Комп’ютерні системи та мережі” та 7.091503 “Спеціалізовані комп’ютерні системи” денної та заочної форм навчання / укл. : М. М. Касьян, К. М. Касьян, М. Ю. Юрич. – Запоріжжя : ЗНТУ, 2008. – 52 с.

15. Методичні вказівки до виконання лабораторних робіт із дисципліни “Сучасні інтернет-технології” в другому модулі для студентів спеціальностей 7.091501 “Комп’ютерні системи та мережі” та 7.091503 “Спеціалізовані комп’ютерні системи” денної та заочної форм навчання / укл. : М. М. Касьян, К. М. Касьян, М. Ю. Юрич. – Запоріжжя : ЗНТУ, 2008. – 62 с.

ДОДАТОК А
Технічне завдання

Вступ

Наразі кожен з тих хто користується мережею Інтернет у повному доступі, хоч раз отримував необхідну собі інформацію, або навіть розважальну інформацію з новинних веб-сайтів, або інформаційних порталів. Тому вибрану мною тему я вважаю корисною, сучасною та актуальною. Тема вибрана мною відповідає на питання часу, та є актуальною для певної групи людей.

A.1 Підстави для розробки

Підставою для виконуваної розробки в межах дипломної кваліфікаційної роботи бакалавра є завдання, що було визначене за темою «Розробка інформаційного порталу “Футбольна Україна”» і затверджене у відповідності з наказом № 103 від 30 березня 2021 р. за Національним університетом «Запорізька політехніка».

A.2 Призначення розробки

Програма призначена для забезпечення користувачу мережі Інтернет можливості для перегляду інформації актуальної, постійно оновлюваної та пошуку інформації серед рубрик сайту.

A.3 Основні вимоги до програми

A.3.1 Вимоги до функціональних характеристик

Програма повинна забезпечувати виконання наступних функціональних характеристик:

- пошук необхідної інформації;
- перегляд інформації;

- отримання та відправку відгуку;
- можливість редагування новин адміністратором;
- можливість додання новин адміністратором ;
- пошук новин серед інформації на сайті;
- перегляд статистичних даних ;
- заповнення форм;
- перегляд заповнення форм.

A.3.2 Вимоги до надійності

Для забезпечення надійності має виконуватися перевірка даних, що було введено користувачем в поля форм.

A.3.3 Умови експлуатації

Експлуатація програмного забезпечення має виконуватися на основі персонального комп'ютера з обов'язковим підключенням до мережі Інтернет. Для мати персональний комп'ютер.

A.3.4 Вимоги до складу та параметрів технічних засобів

Сторона користувача має відповідати таким вимогам:

- процесор: 2 ядра;
- оперативна пам'ять – 1 Гб;
- вільний доступ до мережі Інтернет;
- жорсткий диск – 1 Гб вільного простору.

A.3.5 Вимоги до транспортування та збереження

Збереження на окремих фізичних носіях передбачається.

А.3.6 Вимоги до програмної документації

Основою розробки даного програмного проєкту є технічне завдання на нього.

Опис прийнятих рішень виконується в пояснювальній записці. У розділі «Експлуатація та тестування програми» має бути представлено опис процесу роботи з програмою. Інших програмних документів не передбачається.

А.4 Порядок контролю та приймання

Календарний план завдання на дипломну кваліфікаційну роботу визначає набір етапів, за якими має забезпечуватися виконання, а також результати, які мають бути отримані за виконання відповідного етапу. Саме на основі даних результатів має забезпечуватися приймання роботи.

ДОДАТОК Б
Текст програми

Б.1 Текст файла main.html та text/css

```

<html>
<meta charset="utf">
<head>
<link rel="shortcut icon" href="/images/4.jpg" type="image/x-icon">
<title>Футбольна Україна – новинний веб-портал</title>
<style type="text/css">
div#block {width:1150px; margin:0 auto; background-color:#dddddd}
div.header {width:1150px; height:300px; }
div.left_col {width:208px; height:1150px; float:left; border-right:2px
dashed grey; margin-bottom:-15px}
div.center_col {width:700px; float:left;margin-left:30px; }
div.right_col {width:58px; height:1150px; float:left; border-left:2px dashed
grey; margin-bottom:-15px}
div.footer {width:1150px; height:100px; background-color:#717dc9;
clear:both}
</style>
</head>
<meta charset="utf">
<body>
<div id="block">
<div class="header"><h1 align="center"></h1></div>
<div class="left_col"><p align="center"><style type='text/css'>
#mes
{
color:white;
border-style:solid;
width:207px;

```

```

text-align:center;
border-color:black;
border-width:1px;
padding-top:3px;
padding-bottom:3px;
background-color:blue;
}
</style>
<script type='text/javascript'>
function textChange() {
    document.getElementById('mes').style.backgroundColor='blue';
}
function textReturn() {
    document.getElementById('mes').style.color='white';
    document.getElementById('mes').style.backgroundColor='grey';
}
</script>
</head>
<body>
<p >
<BODY>

<span id=tick98></span>
<script>

function show2(){
if (!document.all&&!document.getElementById)
return
thelement=document.getElementById?
document.getElementById("tick2"): document.all.tick2

```

```

var Digital=new Date()
var hours=Digital.getHours()
var minutes=Digital.getMinutes()
var seconds=Digital.getSeconds()
var dn="PM"
if (hours<12)
dn="AM"
if (hours>12)
hours=hours-12
if (hours==0)
hours=12
if (minutes<=9)
minutes="0"+minutes
if (seconds<=9)
seconds="0"+seconds
var ctime=hours+":"+minutes+":"+seconds+" "+dn
thelement.innerHTML="<b
style='font-
size:16;color:blue;'>"+ctime+"</b>"
setTimeout("show2()",7000)
}
window.onload=show2

</script>
</BODY>
</p>
#menu
{
border:1px solid black;
color:white;
width:178px;

```

```
padding:6px;
margin:0px;
background-color:#006064;
font-size:1.1em;
}
#list
{
border-bottom:4px solid;
width:192px;
list-style-type:none;
margin:0px;
padding:0px;
font-size:1.1em;
}
#list li
{
padding:6px;
border-bottom-style:solid;
border-bottom-color:white;
border-width:1px;
background-color:#ffd873;
}
#img
{
margin:0px;
}
</style>

<script type="text/javascript">
$(document).ready(function(){
```

```

$("#menu").click(function(){ $("#list").slideToggle(2000)});
$("#menu").toggle(function(){
    $("#img").attr("src","menudown.gif"), function(){
    $("#img").attr("src","menuup.gif")
    });
    $("#menu").mouseover(function(){          $("#menu").css("background-
color","#01939a"));
    $("#menu").mouseout(function(){          $("#menu").css("background-
color","#006864"));

});
</script>
</head>
<body>

<div class="center_col">
<h2 align="center"></h2>
<h4 align="center">

<ul id="navbar">
<li><a href="1.html">Регистрация</a></li>
<li><a href="4.html">Контакты</a></li>
<li><a href="2.html">Видеогалерея</a></li>
<li><a href="7.html">Резюме</a></li>
</ul>
<style>
#navbar {
    margin: 0;
    padding: 0;

```

```

    list-style-type: none;
}
#navbar li { display: inline; }

#navbar {
    margin: 0;
    padding: 0;
    list-style-type: none;
    border: 2px solid #0066FF;
    border-radius: 20px 5px;
    width: 690px;
    text-align: center;
    background-color: #33ADFF;
}
#navbar a {
    color: #fff;
    padding: 5px 10px;
    text-decoration: none;
    font-weight: bold;
    display: inline-block;
    width: 100px;
}
#navbar a:hover {
    border-radius: 20px 5px;
    background-color: #0096FF;
}

</style>
<p align="justify">
    <meta charset="utf">

```

```

        <li><em> Динамо</li></em>
        <li><em> Динамо</li></em>
        <li><em> Динамо</li></em>
        <li><em> Шахтер</li></em>
    </li><em> Днепр</li></em>
        <li><em> Заря</li></em>

</h4>
</div>

<div class="right_col"><p align="center">
</p></div>

<img src=images/9.gif>

<div class="footer"><p>© 2019 zntu </p></div>
</div>

<!-- HelloPreload http://hello-site.ru/preloader/ -->
<style
type="text/css">#hellopreload>p{display:none;}#hellopreload_preload{display:
block;position: fixed;z-index: 99999;top: 0;left: 0;width: 100%;height:
100%;min-width: 1000px;background: #000000 url(http://hello-
site.ru//main/images/preloads/tail-spin.svg) center center no-repeat;background-
size:41px;}</style>

    <div id="hellopreload"><div id="hellopreload_preload"></div><p><a
href="http://hello-site.ru">Hello-Site.ru.      Áãñïèàòíúé      êĩñòðóéêîïð
ñàéòîâ.</a></p></div>

    <script      type="text/javascript">var      hellopreload      =
document.getElementById("hellopreload_preload");function
fadeOutnojquery(el){el.style.opacity      =      1;var      interhellopreload      =
setInterval(function(){el.style.opacity = el.style.opacity - 0.05;if (el.style.opacity
<=0.05){      clearInterval(interhellopreload);hellopreload.style.display      =
"none";}},16);}window.onload      =

```

```

function(){setTimeout(function(){fadeOutjquery(hellopreloader);},1000);};</scr
ipt>
    <!-- HelloPreload http://hello-site.ru/preloader/ -->
    </body>
    </html>
    function show2(){
    if (!document.all&&!document.getElementById)
    return
    thelement=document.getElementById?
document.getElementById("tick2"): document.all.tick2
    var Digital=new Date()
    var hours=Digital.getHours()
    var minutes=Digital.getMinutes()
    var seconds=Digital.getSeconds()
    var dn="PM"
    if (hours<12)
    dn="AM"
    if (hours>12)
    hours=hours-12
    if (hours==0)
    hours=12
    if (minutes<=9)
    minutes="0"+minutes
    if (seconds<=9)
    seconds="0"+seconds
    var ctime=hours+":"+minutes+":"+seconds+" "+dn
    thelement.innerHTML="<b
size:16;color:blue;>"+ctime+"</b>"
    setTimeout("show2()",1000)
    }
    style='font-

```

window.onload=show2

</script>

</BODY>

</p>

<script type="text/javascript"

src="http://ajax.googleapis.com/ajax/libs/jquery/1.5/jquery.min.js"></script>

<style type="text/css">

#menu

{

border:1px solid black;

color:white;

width:178px;

padding:6px;

margin:0px;

background-color:#096064;

font-size:1.1em;

}

#list

{

border-bottom:4px solid;

width:198px;

list-style-type:none;

margin:0px;

padding:0px;

font-size:1.1em;

}

#list li

{

padding:6px;

```
border-bottom-style:solid;
border-bottom-color:white;
border-width:1px;
background-color:#ffd273;
}
#img
{
margin:0px;
}
</style>
```

```
<script type="text/javascript">
$(document).ready(function(){

    $("#menu").click(function(){ $("#list").slideToggle(2000)});
    $("#menu").toggle(function(){
        $("#img").attr("src","menudown.gif"), function(){
            $("#img").attr("src","menuup.gif")
        });
    $("#menu").mouseover(function(){          $("#menu").css("background-
color","#01939a")});
    $("#menu").mouseout(function(){          $("#menu").css("background-
color","#006064")});

});
</script>
</head>
<body>
```

```

        id='mes'
        onmouseover='textChange()'
onmouseout='textReturn()')>Меню</p>
</body>
</html></p>

<p align="center"><a href="1.html">Регистрация </a><p
align="center"><a href="2.html">Видеогалерея </a><p align="center"><a
href="3.html">Фотогалерея </a><p align="center"><a href="4.html">Обратная
связь </a><p align="center"><a href="5.html">Новости </a></p ><p
align="center"><a href="6.html">Турнирная таблица</a></div>

```

```

<div class="center_col">
<h2 align="center"></h2>
<h4 align="center">

```

```

<ul id="navbar">
<li><a href="1.html">Регистрация</a></li>
<li><a href="4.html">Контакты</a></li>
<li><a href="2.html">Видеогалерея</a></li>
<li><a href="7.html">Резюме</a></li>
</ul>

```

```

<style>
#navbar {
    margin: 0;
    padding: 0;
    list-style-type: none;
}
#navbar li { display: inline; }

```

```

#navbar {
    margin: 0;

```

```
padding: 0;
list-style-type: none;
border: 2px solid #0066FF;
border-radius: 20px 5px;
width: 650px;
```

```
<legend> </legend>
```

```
Èìÿ: <br /><input type='text' name='firstname' /><br />
```

```
Ôàìèèèÿ:<br /><input type='text' name='lastname' /><br />
```

```
Îò÷åñòâî: <br /><input type='text' name='lastname' /><br />
```

```
<p>Óêàæèòå âàø ïë:</p>
```

```
<input type='radio' name='s' value='m' /> Íóæñêîé<br />
```

```
<input type='radio' name='s' value='f' /> Æåíñêé
```

```
</fieldset>
```

```
<fieldset>
```

```
<legend></legend>
```

```
<p> p>
```

```
<input type='checkbox' name='space' value='1' />
```

```
<br />
```

```
<input type='checkbox' name='space' value='2' />
```

```
<br />
```

```
<input type='checkbox' name='space' value='3' />
```

```
<br />
```

```
<input type='button' value=" onclick='showMessage()' />
```

```
<script type="text/javascript">
```

```
function showMessage()
```

```
{
```

```
alert(Ви ввели необхідні данні, дякую.)
```

```
}
```

```

div#block {width:1150px; margin:0 auto; background-color:#dddddd}
div.header {width:1150px; height:300px; }
div.center_col {height:900px;width:550px; float:left}
div.footer {width:1150px; height:100px; background-color:#717dc9;
clear:both}

</style>
</head>
<body>
<div id="block">
<div class="header"><h1 align="center"></h1></div>
<div class="center_col">
<h2 align="center"><video width="300" height="200"
controls="controls">
<source src="images/20.mp4" type="video/mp4" /><p align=:center> 3-
0</h4></p>
<h2 align="center"><video width="300" height="200"
controls="controls">
<source src="images/21.mp4" type="video/mp4" /></h2>
<h2 align="center"><video width="300" height="200"
controls="controls">
<source src="images/22.mp4" type="video/mp4" /></h2>
</div>
<div class="footer"><p>© 2021</p></div>
</div>
<!-- HelloPreload http://hello-site.ru/preloader/ -->
<style
<=0.05){ clearInterval(interhellopreloader);hellopreloader.style.display =
"none";}},16);}window.onload
=

```

```
function(){setTimeout(function(){fadeOutjquery(hellopreloader);},1000);};</script>
```

```
div#block {width:1150px; margin:0 auto; background-color:#dddddd}
div.header {width:1150px; height:300px; }
div.center_col {height:900px;width:550px; float:left}
div.footer {width:1150px; height:100px; background-color:#717dc9;
clear:both}

</style>
</head>
<body>
<div id="block">
<div class="header"><h1 align="center"></h1></div>
<div class="center_col">
<h2 align="center"><video width="300" height="200"
controls="controls">
<source src="images/20.mp4" type="video/mp4" /><p
align=:center>Âåđîâ-Ìèèàí 3-0</h4></p>
<h2 align="center"><video width="300" height="200"
controls="controls">
<source src="images/21.mp4" type="video/mp4" /></h2>
<h2 align="center"><video width="300" height="200"
controls="controls">
<source src="images/22.mp4" type="video/mp4" /></h2>
</div>
<div class="footer"><p>© 2021</p></div>
</div>
<!-- HelloPreload http://hello-site.ru/preloader/ -->
```

```

<style
type="text/css">#hellopreloader>p{display:none;}#hellopreloader_preload{displa
y: block;position: fixed;z-index: 99999;top: 0;left: 0;width: 100%;height:
100%;min-width: 1000px;background: #000000 url(http://hello-
site.ru//main/images/preloads/tail-spin.svg) center center no-repeat;background-
size:41px;}</style>

<div id="hellopreloader"><div id="hellopreloader_preload"></div><p><a
href="http://hello-site.ru">Hello-Site.ru.      Áãñëàòíué      êĩñòðóêêîð
ñàéòâ.</a></p></div>

<script      type="text/javascript">var      hellopreloader      =
document.getElementById("hellopreloader_preload");function
fadeOutnojquery(el){el.style.opacity      =      1;var      interhellopreloader      =
setInterval(function(){el.style.opacity = el.style.opacity - 0.05;if (el.style.opacity
<=0.05){      clearInterval(interhellopreloader);hellopreloader.style.display      =
"none";}},16);}window.onload      =
function(){setTimeout(function(){fadeOutnojquery(hellopreloader);},1000);};</scr
ipt>

<!-- HelloPreload http://hello-site.ru/preloader/ -->

```

Додаток В

СЛАЙДИ ПРЕЗЕНТАЦІЇ

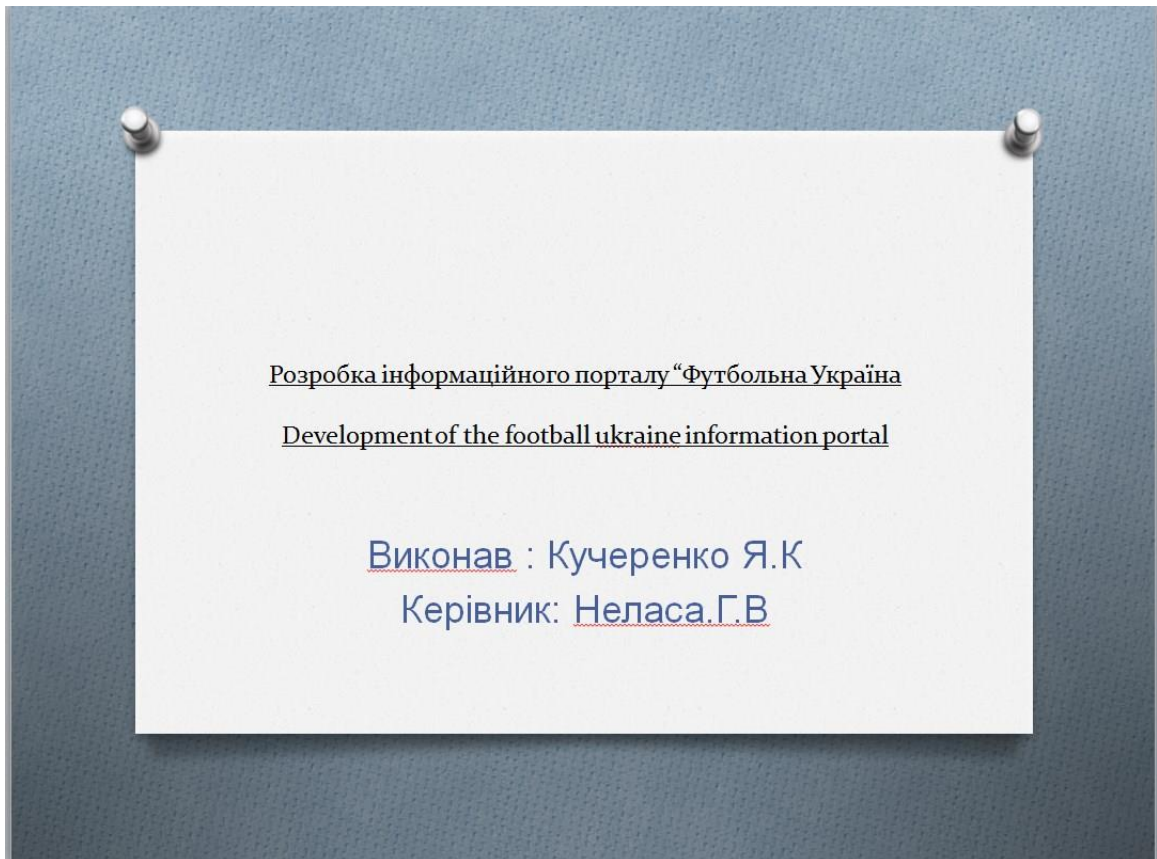


Рисунок В.1 – Слайд № 1

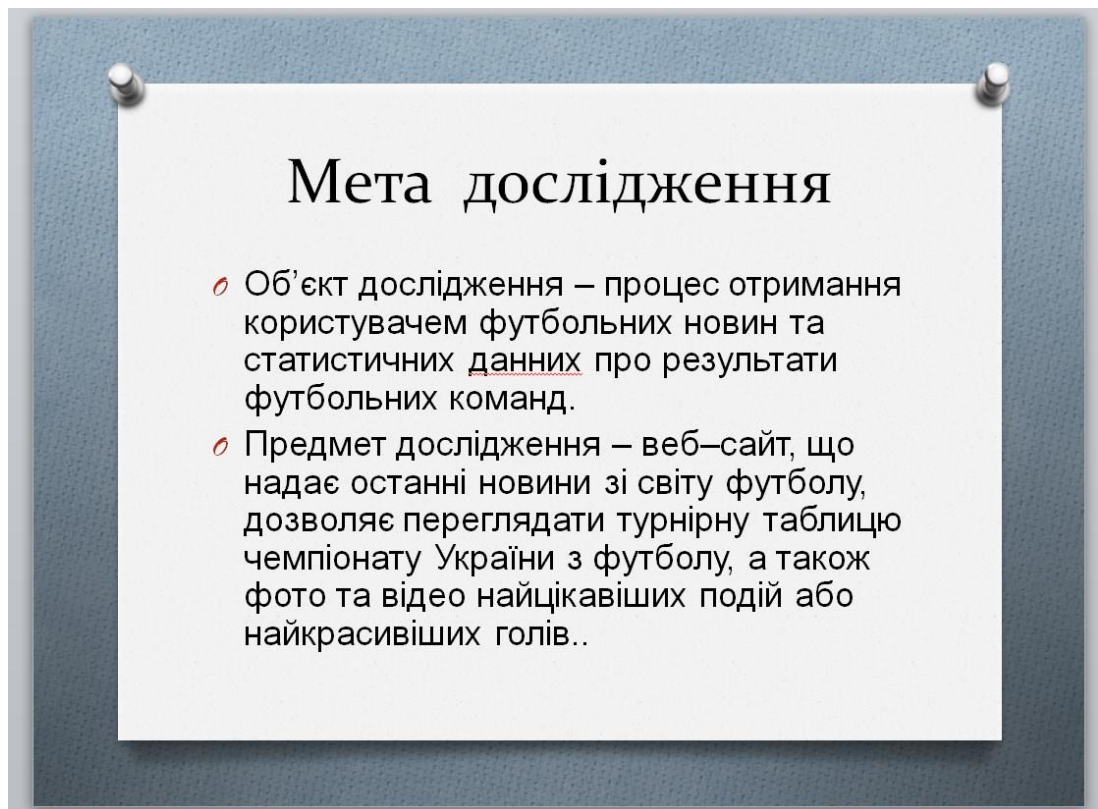


Рисунок В.2 – Слайд № 2

Вибір мови програмування

Параметр	HTML	CSS
Популярність	+++	++
Сумісність з браузерами	+	+
Легкість вивчення	++	+
Підтримка скриптів	+	-
Складність розробки	+	++

Рисунок В.3 – Слайд № 3

Огляд аналогів



Рисунок В.4 – Слайд № 4

Структура сайту

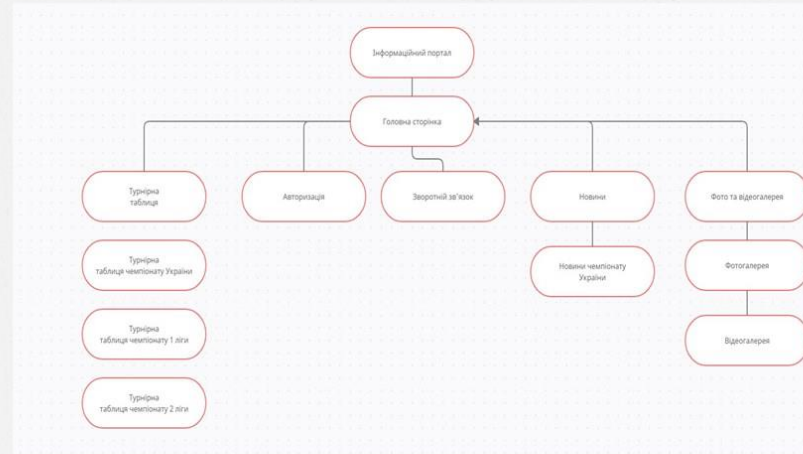


Рисунок В.5 – Слайд № 5

Функціональна схема події залишення відгуку

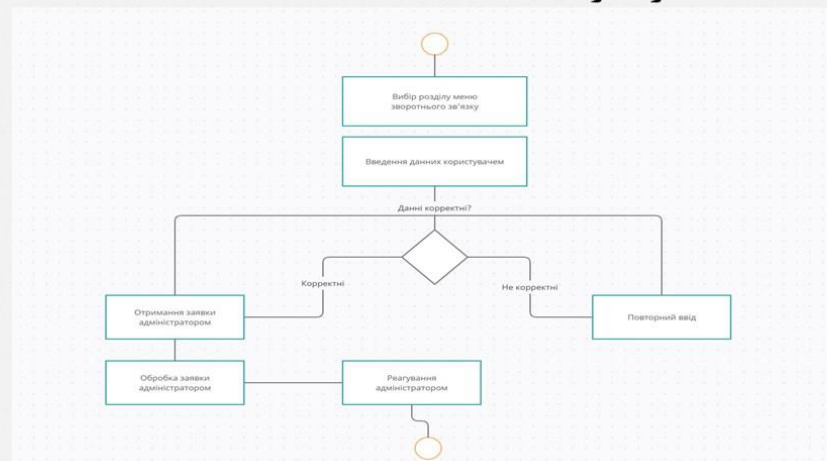


Рисунок В.6 – Слайд № 6

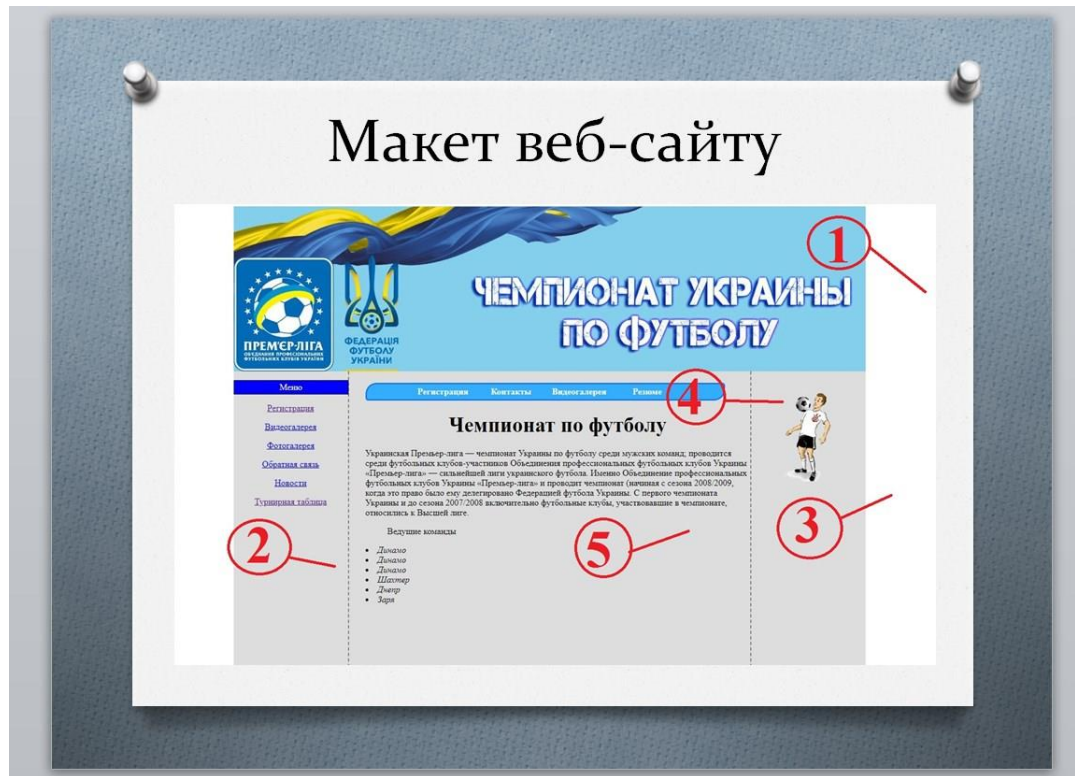


Рисунок В.7 – Слайд № 7

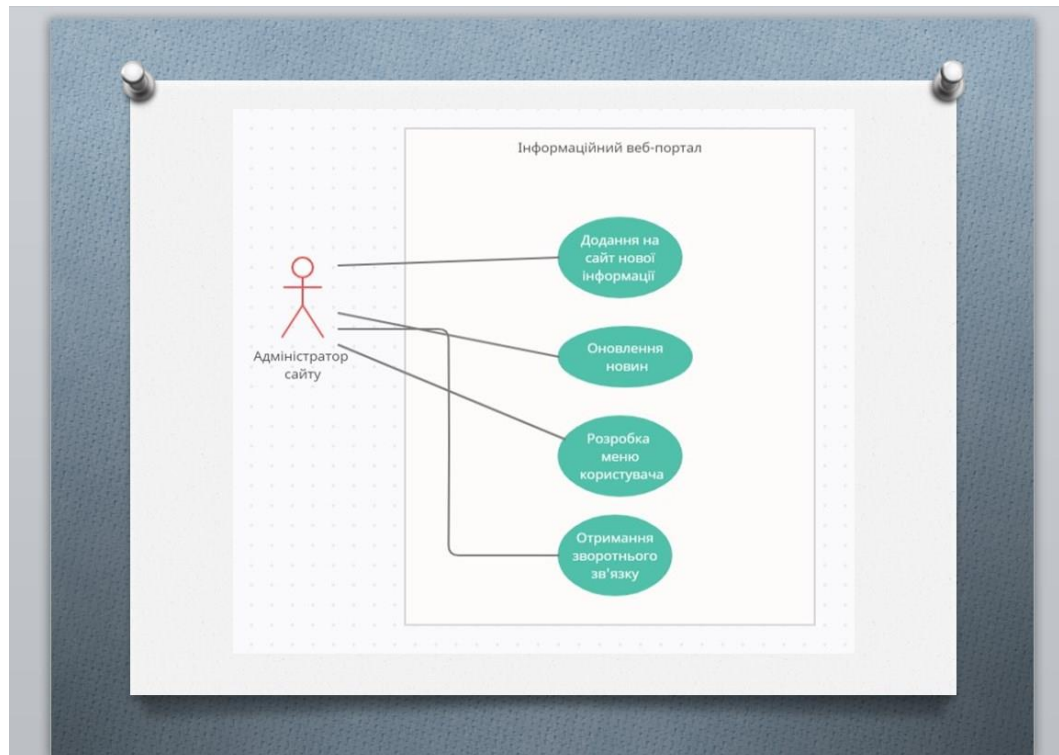


Рисунок В.8 – Слайд № 8

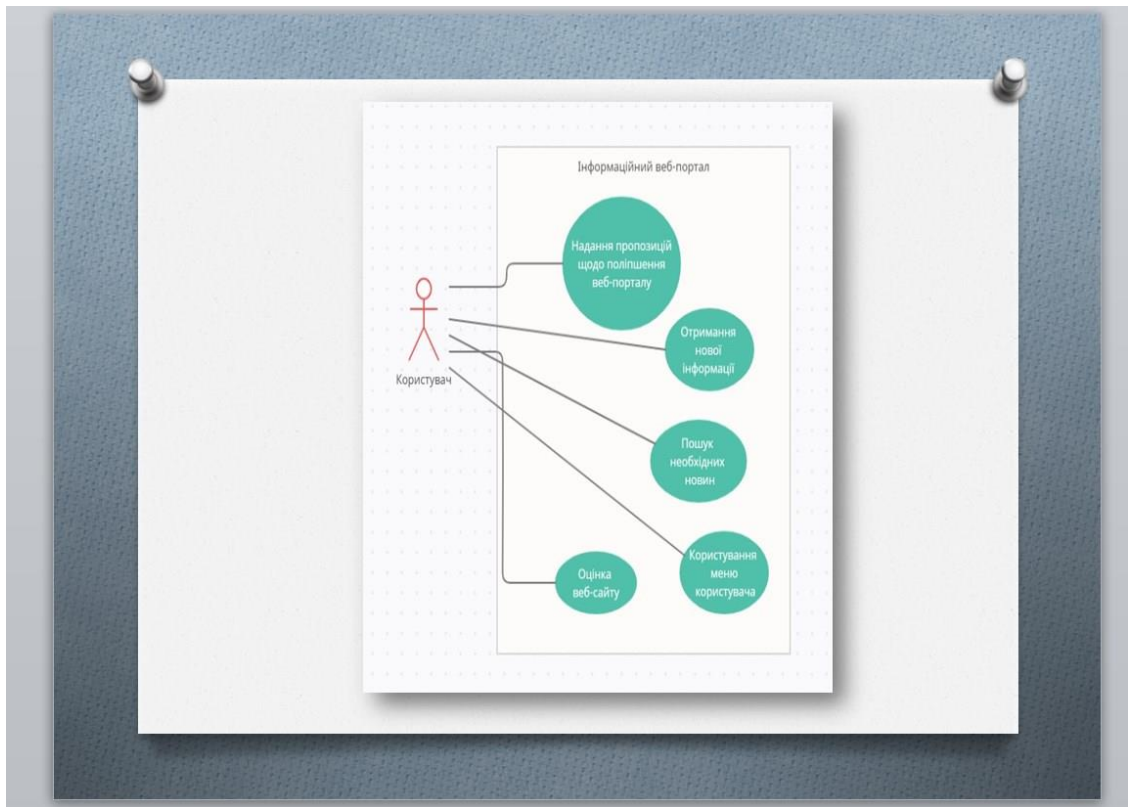


Рисунок В.9 – Слайд № 9

Перегляд сторінки відправки відгуку

ЧЕМПІОНАТ УКРАЇНИ ПО ФУТБОЛУ

Ім'я:

Email:

Телефон:

Сайт:

Причина вашого отзыва:

Рисунок В.10 – Слайд № 10

Висновки

- Переді мною стояло завдання дослідження процесу отримання користувачем футбольних новин та статистичних даних про результати футбольних команд. Предметом дослідження був веб-сайт, що повинен надавати останні новини зі світу футболу, дозволяє переглядати турнірну таблицю чемпіонату України з футболу, а також фото та відео найцікавіших подій або найкрасивіших голів.
- Метою роботи було створення веб-сайту, на якому користувач може дізнатись новини з футбольного світу та слідувати за результатами улюбленої команди.
- Для того щоб виконати поставлену мету я застосовував такі матеріали методи та технічні засоби: мови програмування HTML, JavaScript та CSS, персональний комп'ютер з процесором Intel Core 4 під управлінням операційної системи Microsoft Windows 7, середовище розробки – редактор коду SublimeText. Результатом розробки став веб-сайт, що має можливість надавати користувачам постійно оновлювану інформацію щодо останніх змін у футболі, трансферів гравців, турнірних таблиць, кількості забитих голів та іншої футбольної інформації. Сайт може стати корисним для кожного любителя футболу.
- Я вважаю що усі завдання дипломної кваліфікаційної роботи повністю виконано.

Рисунок В.11 – Слайд № 11