

Міністерство освіти і науки України
Харківський національний університет радіоелектроніки

Факультет Центр післядипломної освіти
(повна назва)

Кафедра Програмної інженерії
(повна назва)

АТЕСТАЦІЙНА РОБОТА

Пояснювальна записка

другий (магістерський)

(рівень вищої освіти)

Тема:

«Дослідження методів побудови веб-орієнтованих програмних систем замовлення
послуг з урахуванням галузі їх надання»

Виконав: студент 2 курсу, групи ІПЗмзд-17-1
спеціальності 121- Інженерія програмного забезпечення

(код і повна назва спеціальності)

Освітньо-наукової програми

Інженерія програмного забезпечення

(повна назва освітньої програми)

Паньшина О.А.

(прізвище, ініціали)

Керівник доц. Лановий О.Ф.

(посада, прізвище, ініціали)

Допускається до захисту

Зав. кафедри, проф. _____

З.В.Дудар

Харків 2019

Харківський національний університет радіоелектроніки

Факультет Центр післядипломної освіти

Кафедра Програмної інженерії

Рівень вищої освіти другий (магістерський)

Спеціальність 121-Інженерія програмного забезпечення

(код і повна назва)

освітньо-наукова програма Інженерія програмного забезпечення

(повна назва)

ЗАТВЕРДЖУЮ:

Зав. кафедри _____

(підпис)

« _____ » _____ 20 ____ р.

ЗАВДАННЯ НА АТЕСТАЦІЙНУ РОБОТУ

студентові Паньшиній Ользі Андріївні

(прізвище, ім'я, по батькові)

1. Тема роботи Дослідження методів побудови веб-орієнтованих програмних систем замовлення послуг з урахуванням галузі їх надання

затверджена наказом по університету від “ _____ ” _____ 20 ____ р № _____
заповнюється вручну після отримання наказу

2. Термін подання студентом роботи до екзаменаційної комісії
_____ червня 2019 р.

3. Вихідні дані до роботи види веб-орієнтованих систем, надання послуг на замовлення, формування змісту сторінок в залежності від контексту

4. Перелік питань, що потрібно опрацювати в роботі мета роботи, аналіз проблемної галузі і постановка задачі, огляд методів замовлення послуг, методи погодження різноракурсних зображень однієї сцени та детектування відрізків ліній на основі перетворень, пошук зображень у базах даних, методи швидкого детектування відрізків ліній за допомогою розроблених алгоритмів перетворення

5. Перелік графічного матеріалу із зазначенням креслеників, схем, плакатів, комп'ютерних ілюстрацій (слайдів) Мета завдання, обґрунтування доцільності розроблення, постановка задачі, об'єктна модель системи, базові моделі, методи й алгоритми, структура бази даних, структурно-логічна схема взаємодії даних, інтерфейс програмної системи, результати тестування програмної системи, демонстраційні матеріали

6 Консультанти розділів роботи

Найменування розділу	Консультант (посада, прізвище, ім'я, по батькові)	Позначка консультанта про виконання розділу	
		підпис	дата
Спецчастина	Доц. Лановий О.Ф.		

КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів роботи	Терміни виконання етапів роботи	Примітка *
1.	Аналіз предметної галузі	10 квітня 2019р.	
2.	Огляд існуючих методів	27 квітня 2019р.	
3.	Методи швидкого детектування відрізків ліній		
4.	Підготовка пояснювальної записки	20 травня 2019р.	
5.	Спецчастина	26 травня 2019р.	
6.	Підготовка презентації та доповіді	28 травня 2019р.	
7.	Попередній захист	14 травня 2019р.	
8.	Нормоконтроль, рецензування	18 червня 2019р.	
9.	Занесення диплома в електронний архів	20 червня 2019р.	
10.	Допуск до захисту у зав. кафедри	25 червня 2019р.	
* заповнюється вручну після виконання чергового пункту			

Дата видачі завдання _____ 2019 р.

Студент _____
(підпис)

Керівник роботи _____ доц. Лановий О.Ф.

РЕФЕРАТ / ABSTRACT

Пояснювальна записка до атестаційної роботи «Дослідження методів побудови веб-орієнтованих програмних систем замовлення послуг з урахуванням галузі їх надання»: 84 с., 29 рис., 8 табл., 26 джерел.

Мета роботи: дослідження процесу розробки веб-системи замовлення послуг, що враховує конкретну область їх надання. Об'єкт дослідження: методи і моделі побудови веб-сайту надання послуг з урахуванням потреб користувача із застосуванням системного аналізу.

В роботі використана матриця Шеррингтона, яка дозволяє впорядкувати і систематизувати знання про споживача, застосована методика вербалізації інформаційних портретів користувача. Отримано програмну систему щодо надання послуг на замовлення.

ВЕБ-САЙТ, БАЗА ДАНИХ, PHP, mySQL, WordPress , CMS

Master`s thesis “Dates on the methods of web-based software systems, which are assigned to the users of the galaxy on the back”: 84 p., 29 fig., 8 tab., 26 dzherel.

Purpose: the study of the process of developing a web-based service order system that takes into account the specific area of their provision. Object of research: methods and models of building a website for providing services tailored to the needs of the user with the use of system analysis.

This work is use the Sherrington matrix which allows to organize and systematize knowledge about the consumer, the method of verbalization of information portraits of the user is applied. The software system for providing custom-made services has been received.

WEB SITE, DATABASE, PHP, mySQL, WordPress , CMS

ЗМІСТ

Вступ.....	6
1 Аналіз стану дослідження предметної області.....	8
1.1 Характеристики та види веб-орієнтованих систем.....	8
1.2 Характеристики веб-сторінок.....	11
1.3 Інформаційна архітектура веб-сайту.....	13
1.4 Контент веб-сайту.....	15
1.5 Аналіз аналогічних сайтів.....	21
1.6 Визначення цільового користувача.....	24
2 Розробка моделі веб-сайту.....	32
2.1 Склад і взаємозв'язок компонентів веб-сайту.....	32
2.2 Модель адаптації веб-сайту.....	35
2.3 Вибір засобів реалізації.....	38
3 Система реалізації сайту.....	42
3.1 Розробка дизайну веб-сайту.....	42
3.2 Діаграма варіантів використання UseCase.....	45
3.3 Розробка теми в системі WordPress.....	47
3.4 Користувачі WordPress.....	53
3.5 Розробка бази даних.....	54
3.6 Розробка багатомовності та медіагалереї.....	56
3.7 Реалізація адаптивності.....	57
3.8 Функціональне тестування системи.....	61
3.9 Тестування адаптивності.....	65
3.10 Тестування веб-відображень.....	66
3.11 Юзабіліті тестування сайту.....	68
Висновки.....	70
Перелік джерел посилань.....	72
Додаток А Слайди презентації.....	74

ВСТУП

Кожна фірма, намагаючись слідувати духу часу, в своїй діяльності застосовує сучасні технології. На сьогоднішній день більшість компаній прагне якомога ефективніше впливати на споживачів за допомогою реклами, але не всі її методи є дієвими. В умовах загострення ринкової конкуренції, пов'язаної із зростанням кількості підприємств, розширенням асортименту товарів та послуг, що надаються, все складніше стає відшукати саме свого споживача. Для того, щоб привернути увагу замовника послуг, аргументовано вплинути на його вибір, підприємства використовують новітні технології. Це потрібно для того, щоб кінцевий результат діяльності у вигляді нового або вдосконаленого товару, послуги чи технологічного процесу досяг оптимального рівня. Таким чином, переважна більшість компаній звертається до Інтернет-простору для інформування про свою продукцію відвідувачів Інтернету. На допомогу приходять веб-орієнтовані програмні системи замовлення послуг. Ці розробки не просто допомагають в реалізації, а формують взаємовигідні довгострокові відносини. Адже і виробникам і споживачам потрібно знати, що являє собою ринок, хто на ньому діє, як він функціонує, які його запити й пропозиції. Слід зауважити, що для розвитку та підвищення конкурентноспроможності компанії необхідні засоби, які б надавали споживачеві можливість прискорити пошук потрібного товару або послуги, а також отримати дані у зручній формі. Таким чином, необхідні, як оперативне отримання інформації щодо пропозицій, так і зручність користування з ефективним інтерфейсом. І це є, мабуть, основними вимогами до веб-систем замовлення послуг.

У даний час на ринку існує багато продуктів, покликаних автоматизувати процес пошуку послуг і товарів. Більшість з них мають недоліки, які є спільними. Наприклад, основний з них - бажання охопити максимальний спектр можливостей обертається надлишковістю функціоналу.

З огляду на вищенаведене можна констатувати, що надати підтримку постачальникам товарів та послуг на ефективному рівні призвані веб-орієнтовані програмні системи замовлення послуг, спроможні звертатися до цільової аудиторії. А запити свого користувача і особливості його поведінки треба ретельно вивчати ще на етапі розробки програмної системи. Тобто вже на початкових етапах розробки необхідно враховувати особливості галузі, для якої створюється програмна система, що передбачає пропонувати ті чи інші послуги. Отримати бажаний результат можливо завдяки використанню різноманітних методів побудови веб-систем, адаптованих до тієї чи іншої галузі. В дуже великій мірі все вищесказане стосується і ремонтно-будівельного сектору. Тут також необхідні автоматизовані методики вивчення попиту.

Саме дослідженню таких методів і присвячена дана робота.

Таким чином, дана роботи має на меті дослідження процесу розробки веб-системи замовлення послуг, що враховує конкретну область їх надання і методів побудови таких систем. Для досягнення мети треба вирішити наступні задач: ознайомитись з видами веб-орієнтованих систем;

- провести порівняльний аналіз сайтів, що надають аналогічні послуги;
- виконати проектування макетів сторінок з подальшою розробкою дизайну;
- вивчити існуючі CMS та обрати оптимальну з них;
- дослідити сучасні методи і моделі побудови веб-сайту надання електромонтажних послуг та енергоаудіту ремонтно-будівельного підприємства з урахуванням потреб користувача за допомогою сучасних методик і системного аналізу.

1 АНАЛІЗ СТАНУ ДОСЛІДЖЕННЯ ПРЕДМЕТНОЇ ОБЛАСТІ

1.1 Характеристики та види веб-орієнтованих систем

Залежно від конкретної області застосування інформаційні системи можуть дуже сильно відрізнятися за своїми функціями, архітектурою, реалізацією. Однак можна виділити дві властивості, які є загальними для всіх інформаційних систем. По-перше, будь-яка інформаційна система призначена для збору, зберігання і обробки інформації. Тому в основі будь-якої інформаційної системи лежить середовища зберігання і доступу до даних. Середовище повинне забезпечувати рівень надійності зберігання та ефективність доступу, які відповідають області застосування інформаційної системи. По-друге, інформаційні системи орієнтуються на кінцевого користувача, це можуть бути будівельники, лікарі, пересічні користувачі. Тому для них інформаційна система повинна мати простий, зручний інтерфейс, який повинен надати кінцевому користувачеві всі необхідні для його роботи функції і одночасно захистити його від виконання зайвих дій. Іноді цей інтерфейс може бути графічним з меню, кнопками, підказками і т.д. У будь-якому випадку наявність розвинених інтерфейсних засобів є обов'язковим для будь-якої сучасної інформаційної системи.

Веб-система - клієнт-серверний додаток, в якому клієнт взаємодіє з сервером за допомогою браузера, а за сервер, в свою чергу, відповідає веб-сервер. Логіка веб-додатків розподілена між сервером і клієнтом, зберігання даних здійснюється, переважно, на сервері, обмін інформацією відбувається по мережі. Одним з переваг такого підходу є факт, що клієнти не залежать від конкретної операційної системи користувача, тому веб-додатки є міжплатформеними службами.

Слід сказати декілька слів про архітектуру системи клієнт/сервер (E / AS).

Система E / AS це клієнт-серверний додаток, що складається з ПЗ сервера (надалі - сервер) і ПЗ клієнтів (надалі - клієнт). Сукупність сервера і клієнта називається домен. Клієнт і сервер, по суті, є рівноправними компонентами

(сервісами) роботи з об'єктами системи. Вони використовують окрему бібліотеку, що підключається для роботи сервера та клієнтського ПЗ. Об'єкти системи створюються на підставі раніше описаних класів і зберігаються в базі даних (далі - БД). Методи об'єктів можуть бути описані на будь-якій мові, який може бути трансльований в байт-код, в якому ці методи зберігаються і використовуються в БД.

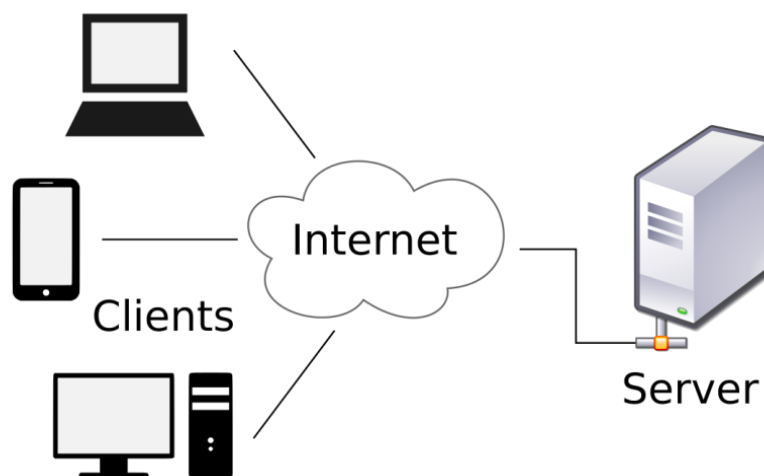


Рисунок 1.1- Домен клієнт/сервер

Взаємодія сервера і клієнтів здійснюється за певним протоколу поверх протоколу TCP / IP. За цим протоколом здійснюється отримання об'єктів, виклик методів об'єктів на сервері або ж скачування байт-коду методів. На малюнку 1.1 представлений домен клієнт /сервер.

Принципи роботи клієнта з сервером:

- клієнт після аутентифікації отримує форму своєї робочої області. Це форма, яка містить всі необхідні інструменти для роботи користувача відповідно до його прав доступу і привілеїв;

- коли клієнт викликає певну форму, вона запитується на сервері. Після отримання форми (і необхідних методів для роботи з нею локально), вона адресується через драйвер форм;

– оператор працює з даними в формі, події роботи з формою обробляються через байт-машину через локальні або серверні методи роботи з формою і об'єктом;

– крім зазначеної роботи з формою, оператор може отримати друковану форму необхідного документа або звіт (посередництвом виклику певного методу формування друкованих форм). Цей документ може бути роздрукований або експортований в іншу програму.

Клієнт/серверна архітектура передбачає розподіл завдань і обчислювального навантаження по декількох комп'ютерах, об'єднаних в мережу. Програмне забезпечення інформаційної системи, реалізоване в цій архітектурі, логічно поділяється на серверне ПЗ, ПЗ середнього рівня і ПЗ клієнтських робочих місць (клієнтів). При цьому програмне забезпечення клієнтів відповідає за презентаційні функції і взаємодію з користувачем, середній рівень - за виконання логічних правил і обробку інформації йде до клієнта або від клієнта, сервер забезпечує безпечне і надійне зберігання даних і здійснення вибірок даних за запитом.

Якщо кожен рівень представлений окремим програмним модулем, то ми маємо справу з логічною трирівневою клієнт/серверної архітектурою. Модулі можуть інстальоватися як на три і більше окремих машин, так і на дві залежно від обставин.

Реалізація інформаційних систем в архітектурі клієнт/сервер забезпечує:

– надійність зберігання інформації - можна не турбуватися про те, що апаратні збої або помилкові дії персоналу приведуть до втрати даних;

– цілісність інформації - вбудовані механізми зменшують ризик виникнення колізій при введенні і редагуванні даних;

– регламентований доступ до інформації - кожен користувач може працювати тільки з тими даними, до яких він отримав доступ, і тільки в тому режимі, який відповідає даним йому прав; можливість безпечного підключення до глобальних мереж;

- багатокористувацький режим роботи з інформацією, в тому числі і віддалених користувачів; високу продуктивність і зниження навантаження на мережу за рахунок розподілу процесів між серверами і робочими станціями;
- мінімізацію мережевого трафіку;
- масштабованість - продуктивність системи зростає пропорційно числу процесорів і оперативної пам'яті серверів;
- оптимізацію розподілу обчислювального навантаження між сервером і клієнтом.

Ключовою інформаційною сутністю веб-системи є веб-сторінка, яка має свій унікальний ідентифікатор – URL-адресу.

1.2 Характеристики веб-сторінок

Веб-сайт – це мультимедійний та періодично оновлюваний продукт, який існує тільки в цифровому середовищі. Веб-сайт являє собою масив пов'язаних даних, об'єднаних спільною темою, дизайном, який має унікальну адресу та сприймається користувачем, як одне ціле. Стратегічно важливим вже на ранніх етапах розробки (маркетинговому дослідженні і створенню концепції) є питання про те, до якої категорії віднести сайт. Тільки за допомогою певної структури і форми подачі матеріалу можна досягти успіху у конкретної цільової аудиторії. Веб-сторінка – це складова веб-сайту. Веб-сторінки можуть бути статичні, на яких не передбачена часта зміна даних (створені засобами мови HTML), динамічні – для відображення часто оновлюваних матеріалів (створені з використанням мови PHP), флеш-сторінки – для створення яскравого та привабливого контенту (з використанням технології Adobe flash).

Після проведеного аналізу веб-сайтів створюємо модель веб-сайту для ПК [1]. Модель настільної версії сайтів представлена на рисунку 1.2.

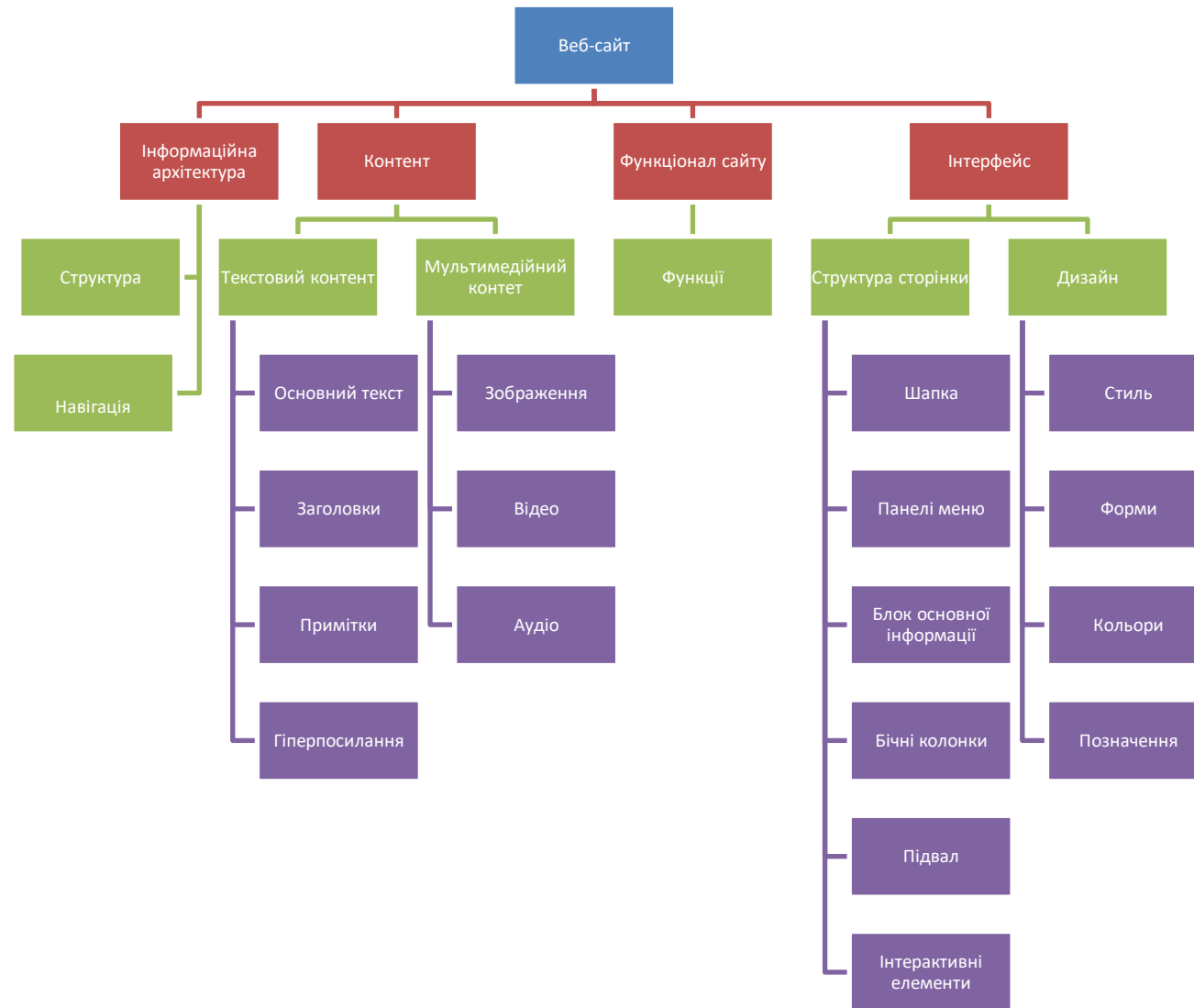


Рисунок 1.2 - Модель веб-сайту

1.3 Інформаційна архітектура веб-сайту

Інформаційна архітектура - це своєрідна схема організації інформації, вона визначає структуру і навігацію сайту [2]. Структура сайту – це взаємне розташування всіх розділів і підрозділів сайту. Розглянемо наступну ситуацію: в процесі розробки сайту багато уваги було приділено розробці чудового дизайну, застосуванню новітніх технологій і формуванню контенту. В результаті, через деякий час після запуску сайту, виявляється, що користувачам важко знайти необхідну інформацію, а адміністратори не можуть визначити до якого розділу віднести нові матеріали. Причина - відсутність продуманої структури. Існують різні типи структур, але найбільш поширеною є ієрархічна деревоподібна (рис.1. 2). Тому далі під «структурою сайту» буде матися на увазі саме цей тип.

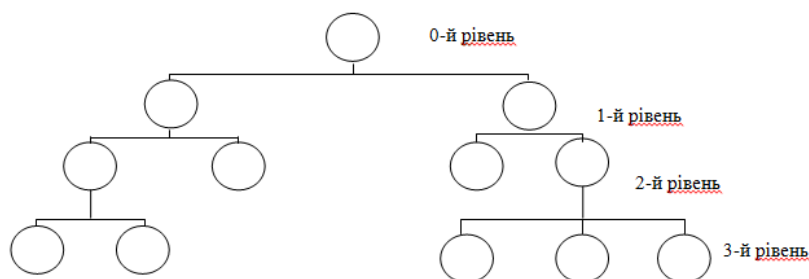


Рисунок 1.3 - Приклад деревовидної структури сайту

Навігація - система візуальних засобів, за допомогою яких користувач може переміщатися по сайту і знаходити потрібну інформацію. Інформаційна архітектура не є частиною призначеного для користувача інтерфейсу, що відбивається на екрані, її можна назвати «невидимою» частиною сайту. Саме елементи навігації є відображенням ІА в інтерфейсі сайту. Завдяки ним користувач може переміщуватися по розробленій структурі і з легкістю орієнтуватися в ній.

Існують різні види навігації:

- основна навігація - це панель меню сайту, що містить посилання на основні розділи;
- допоміжна навігація - містить посилання доступу до основних функцій сайту, таких як реєстрація, пошук і т.д., даний вид навігації не обов'язковий, або призначений до другорядних розділів;
- локальна навігація - навігація всередині розділу, містить посилання на підрозділи, може бути у вигляді меню, що випадає;
- контекстна навігація - посилання на розділи, тематично пов'язані з поточним;
- міжтекстова навігація - посилання в тексті на внутрішні або зовнішні сторінки;
- дублююча навігація - дублювання посилань на основні розділи і підрозділи сайту в підвалі сайту;
- рядок шляху або «хлібні крихти» - елемент навігації, що представляє собою пересування по сторінках сайту від головної до тієї, яку в даний момент переглядає користувач. Даний елемент навігації підходить тільки для деревовидної структури сайту.

Приклад різних навігаційних елементів представлений на рисунку 1.4.

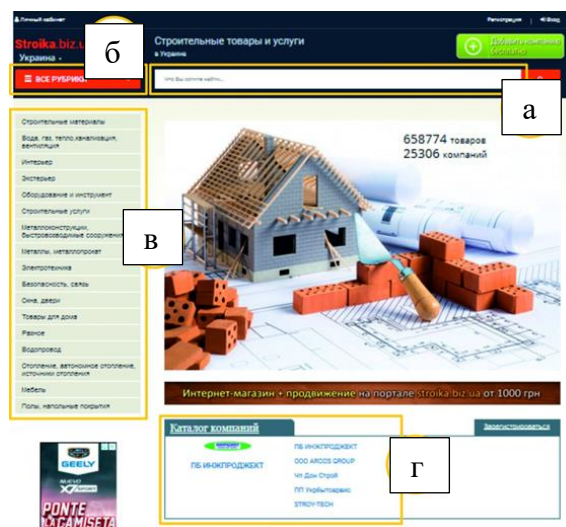


Рисунок 1.4 - Приклад різних елементів навігації

Перелік видів навігації:

- а) допоміжна навігація.
- б) основна навігація.
- в) локальна навігація.
- г) контекстна навігація.

Також велику роль в навігації, а, отже, і в зручності сайту грають назви розділів. В результаті розробки структури сайту виділяються його розділи та підрозділи. Кожен розділ і підрозділ сайту повинен мати назву, що відображає ту інформацію, яка в ньому міститься. Оскільки процес роботи з інформацією досить трудомісткий, заголовки, які логічно обгрунтовані, дозволять користувачам краще орієнтуватися на веб-сайті і значно скорочують тимчасові витрати на пошук потрібної інформації.

1.4. Контент веб-сайту

Контент (від англ. Content - зміст) - це вся інформація, що міститься на сайті. Вона розміщується в блоці основної інформації і є головним інтересом користувачів.

Контент, розміщений на сайті може бути двох типів:

- текстовий контент;
- мультимедійний контент.

Основний текст є найголовнішим елементом сторінки. Саме від цінності й унікальності наданої інформації залежить успіх будь-якого сайту. Тема - це вступна фраза тексту. Заголовки відіграють важливу роль в залученні уваги користувачів, адже саме виходячи з них приймається рішення, чи цікава надана на сторінці інформація. Важливо не плутати заголовки з назвами розділів. Також до цієї групи відносять підзаголовки, які є вступною фразою підтеми тексту. Примітки - це пояснення до тексту, фото чи відео. Даний вид текстового контенту

зустрічається не часто, але функція, яку вона несе недооцінена. Наприклад, на великих настільних сайтах складно встежити за тим, щоб всі зображення залишалися доступними для перегляду. Саме тут примітки допоможуть користувачам зрозуміти, що на них було зображено, в разі, якщо їх ще не встигли оновити.

Гіперпосилання, також відомі, як «посилання». Гіперпосилання - це графічне зображення, відео або текст на сайті, що встановлюють зв'язок і дозволяють переходити до інших об'єктів Інтернету. Посилання можуть вести як на внутрішні сторінки ресурсу, так і на інші ресурси.

Мультимедійний, тобто візуальний і звуковий, контент також дуже актуальний, він допомагає структурувати і сприймати інформацію. Але бездумна публікація може привести до зворотного ефекту. За визначенням Microsoft до «мультимедіа» відносяться такі матеріали:

- зображення;
- відео;
- аудіо.

Сучасні веб-сайти є не тільки джерелом інформації, але і інструментом, що допомагає в рішенні безлічі завдань. Для цього будь-який сайт має певний функціонал, тобто набір функцій.

Але не всі технології коректно працюють на різних браузерях. Наприклад, к Flash-технології мають низький ступінь безпеки, надійності і продуктивності. Компанія Google в кінці 2016 року відмовилася від автоматичного запуску Flash-елементів в своєму браузері Chrome [19]. Їх приклад наслідували й багато інших компаній. Тому необхідно уникати реалізації функціоналу веб-сайтів за допомогою даної технології.

Інтерфейс сайту - це його візуальне відображення на екрані монітора. Під візуальним відображенням розуміється, як розташування різних блоків на сторінці, тобто її структура, так і дизайн сайту в цілому. Інтерфейс є одним з найбільш важливих елементів, оскільки відповідає за те, наскільки зручно користувачеві буде взаємодіяти з сайтом. Вже згадувалось, що будь-який сайт - це

сукупність сторінок, на яких розміщується інформація. За час існування сайтів склалися певні правила побудови структури сторінок. На рисунку 1.5 представлена структура, характерна для більшості веб-сторінок.



Рисунок 1.5 - Структура веб-сторінки

Важливо відзначити, що шапка, панель навігації, бічні колонки і підвал не мають параметрів важливих в контексті даної роботи, тому далі будуть розглянуті лише деякі рекомендації для даних елементів сторінки.

Шапка сайту розташовується у верхній частині сторінки і залишається незмінною для кожного розділу. Тому, як правило, на ній розміщують елементи, які однозначно ідентифікують компанію:

- логотип і назва сайту / організації;
- слоган або девіз сайту / організації;
- опис сайту / організації;
- контакти;
- банер.

Всі перелічені елементи служать для формування у відвідувача чіткого асоціативного зв'язку між одержуваною інформацією і конкретною компанією. Також в шапці може розташовуватися доступ до деяких функцій сайту. Наприклад, до пошуку, зміні мови, позиціонуванню правильних авторизації, кошику і т.д.

Панель навігації (меню) сайту - нумерований впорядкований список посилань для переходу між сторінками сайту. Як і шапка, меню сайту має залишатися доступним на будь-якій сторінці. Навігаційна панель веб-сайту може бути:

- горизонтальною, коли меню вважається частиною шапки;
- вертикальною, коли меню, зазвичай, розташовується в лівій колонці і вважається її частиною.

У деяких випадках, наприклад, якщо сайт має багато сторінок, на сайті можливі обидва типи меню. Одне з них відповідає за навігацію по основних розділах сайту, інше - за навігацію всередині одного розділу.

Блок основної інформації містить весь контент сайту. Це найбільш значуща для користувача частина сайту, тому даний блок розташовується в центральній частині сторінки і повинен бути найбільшою її частиною.

Бічні колонки сайту - це окремі секції сайту, графічно відокремлені від інших елементів сайту. Як правило, в бічних панелях розміщують додаткові інформаційні та навігаційні елементи, пов'язані з тематикою сайту, і елементи, що забезпечують швидкий доступ до деяких функцій сайту. Також в бічних колонках розміщують рекламу. Прикладом сайту з двома колонками є рекламний сайт «Прем'єр» (рис. 1.6).

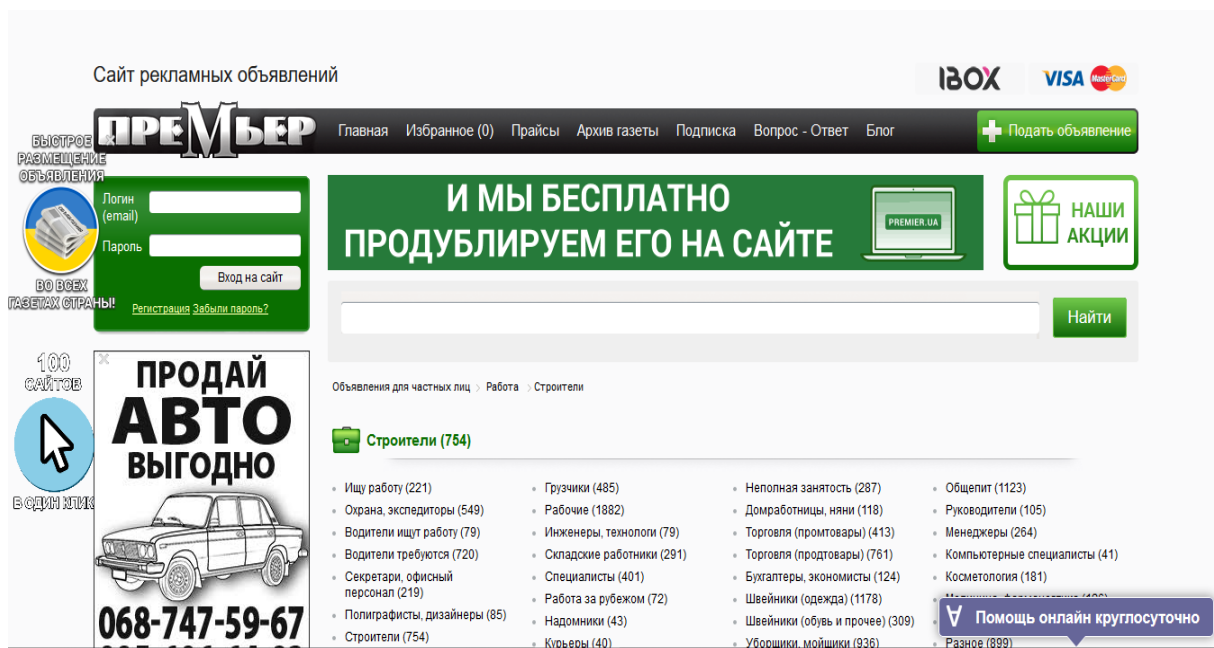


Рисунок 1.6 - Приклад сайту з двома колонками

Бічні панелі покликані зробити взаємодію користувача з сайтом більш комфортним, однак, розміщення їх на сайті не є обов'язковою вимогою. У разі, якщо такі особливості сайту як структура, кількість функцій і дизайн дозволяють, бічних панелей на сайті може не бути зовсім.

Підвал сайту розташовується в нижній частині сторінки і, також, як і шапка, залишається незмінним для кожного розділу. Найчастіше в підвал сайту виносять важливу, але не першорядну інформацію. Наприклад, в підвалі можуть розташовуватися такі елементи:

- контакти власника сайту;
- дані про копірайт;
- дублювання посилань на основні або всі розділи сайту;
- лічильники і банери;
- посилання на соціальні мережі.

Інтерактивні елементи сторінки - це такі, з якими взаємодіє користувач, тобто кнопки, форми введення інформації, випадаюче меню, слайдери і так далі.

Дизайну сайту є творчим процесом, основними його характеристиками можна назвати стиль і кольорову гаму.

Стиль сайту - це його характерний вигляд, що виражається в особливостях художнього оформлення. Будь-який сайт має свій унікальний стиль, що виділяє його серед інших сайтів. У ряді випадків стиль сайту може визначатися тільки формою об'єктів і їх кольором. У разі, якщо сайт розробляється для певної організації, стиль буде визначатися фірмовим стилем компанії.

На даний момент можна виділити два основних напрямки в оформлення веб-сайтів - це скевоморфізм і плоский дизайн.

Скевоморфізм - це стиль дизайну, який імітує об'єкти і текстури з реального життя. Наприклад, коли вид електронного календаря в точності повторює паперовий.

Плоский дизайн - це стиль дизайну, протилежний скевоморфізму. Основна концепція плоского дизайну полягає у відмові від градієнтів, тіней і віддзеркалень, що імітують реальні об'єкти. Такий стиль дозволяє зменшити навантаження на сайти або програми, що прискорює їхню роботу.

Під формою мають на увазі форму всіх блоків і інтерактивних елементів. Форма - головна ознака сутності об'єкта. Для розрізнення форм потрібна велика концентрація уваги, ніж для розрізнення кольору, тому для залучення уваги користувачів її краще не використовувати форми [4].

Колір відіграє важливе значення у створенні певного настрою і акцентує увагу на певному елементі. На одному сайті не повинно бути занадто багато квітів, ефект веселки перевантажує сприйняття користувача [4].

Позначення - це іконки і підписи різних елементів сайту. Так як на веб-сайтах немає можливості навчати смаки користувача, то для позначення кнопок і інших елементів краще користуватися загальноприйнятними, знайомими іконками і звичайними підписами (наприклад, «ім'я користувача» або «пароль», а не «нікнейм» і «секретний код»).

1.5 Аналіз аналогічних сайтів

Початковий аналіз показав, що типова й необхідна інформація для сайту будівельної фірми - опис виду послуг, матеріалів, цін. Але, крім переліку послуг, необхідно показати й іншу інформацію - відгуки, варіанти послуг, що вже надавалися, новації, напрямки розвитку підприємства. Також часто на сайті публікуються статті щодо ситуації в галузі, освітлюються новітні технології. Розглянемо більш ретельно наступні сайти фірм:

а) UkrMarket (ukrmarket.net)

1) на головній сторінці розміщені фільтр вибору регіонів, фільтр вибору послуг за категоріями, строка пошуку (рис. 1.7);

2) у вкладці Об'явлення розташовні списки оголошень надання послуг по категоріям, виведені ТОП оголошень;

3) у вкладці Магазины розташовні списки магазинів з зазначенням адреси та контактного телефону;

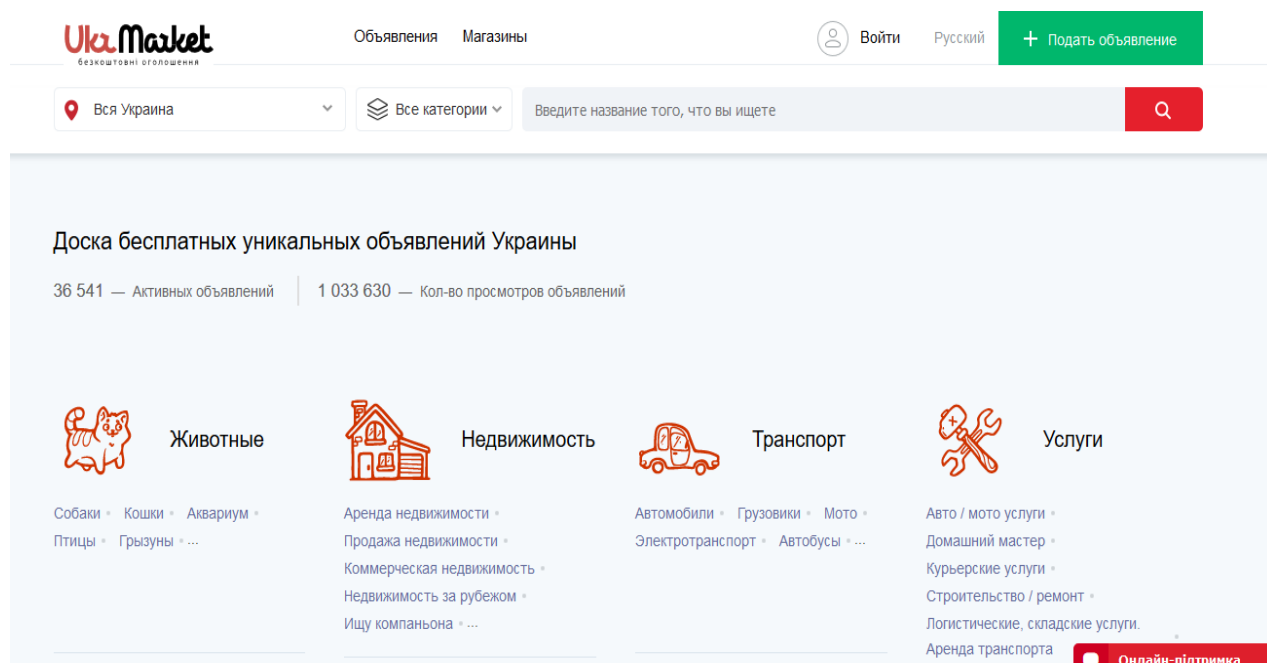


Рисунок 1.7 - Скриншот сайту UkrMarket

б) Stroika.biz.ua (stroika.biz.ua/)

1) на головній сторінці розміщен малюнок, який пояснює напрямок діяльності фірми, строка пошуку, особистий кабінет (рис. 1.8);

2) на бічній панелі розташоване меню пропонуванних послуг;

3) в кожному з пунктів меню міститься інформація про послуги, відповідно за розділу: Будівельні матеріали; Вода, газ, тепло, каналізація, вентиляція; Інтер'єр; Обладнання та інструменти; Будівельні послуги; Металоконструкції; Безпека, зв'язок; Вікна, двері; Різне та ін.

4) в пунктах підменю розташовано інформація про фірми, які надають послуги згідно до категорій із зазначенням адреси та телефону.

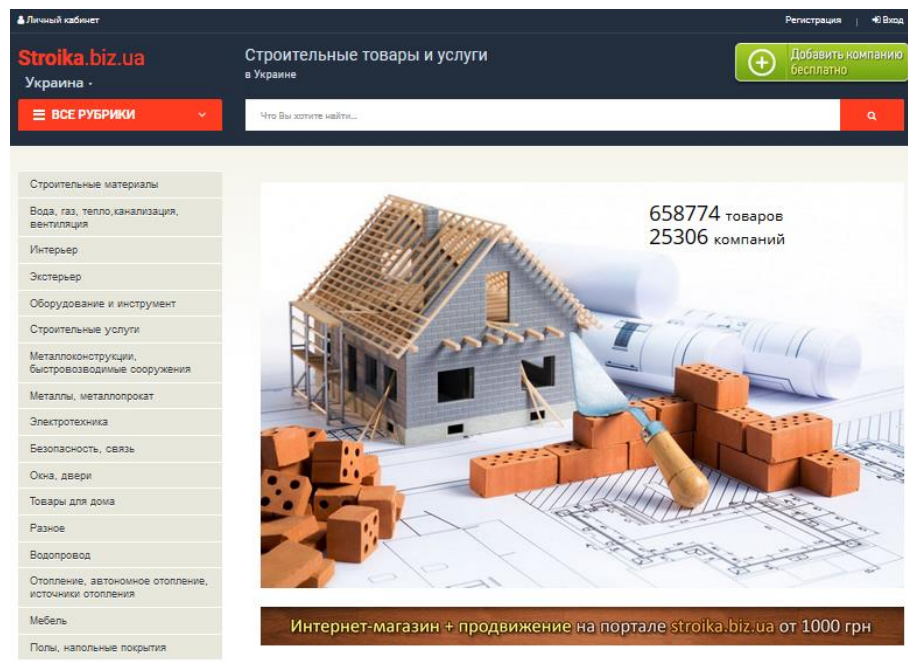


Рисунок 1.8 - Скриншот сайту Stroika.biz.ua

в) ТОВ Енергоінвестпроект (ua.kompass.com)

1) на головній сторінці розташована строка меню, адреса фірми, телефон, загальна інформація, інформація щодо кількості робітників (рис. 1.9);

2) меню містить пункти: Основні показники, Керівництво, Продукція та послуги;

- 3) у розділі Основні показники міститься інформація о кількості робітників та прізвище директора;
- 4) у розділі Керівництво міститься інформація про керівника;
- 5) у розділі Продукція та послуги міститься перелік пропонуванних послуг та продукції;
- 6) на головній сторінці є кнопка «Поділитись» з посиланням на соціальні мережі.

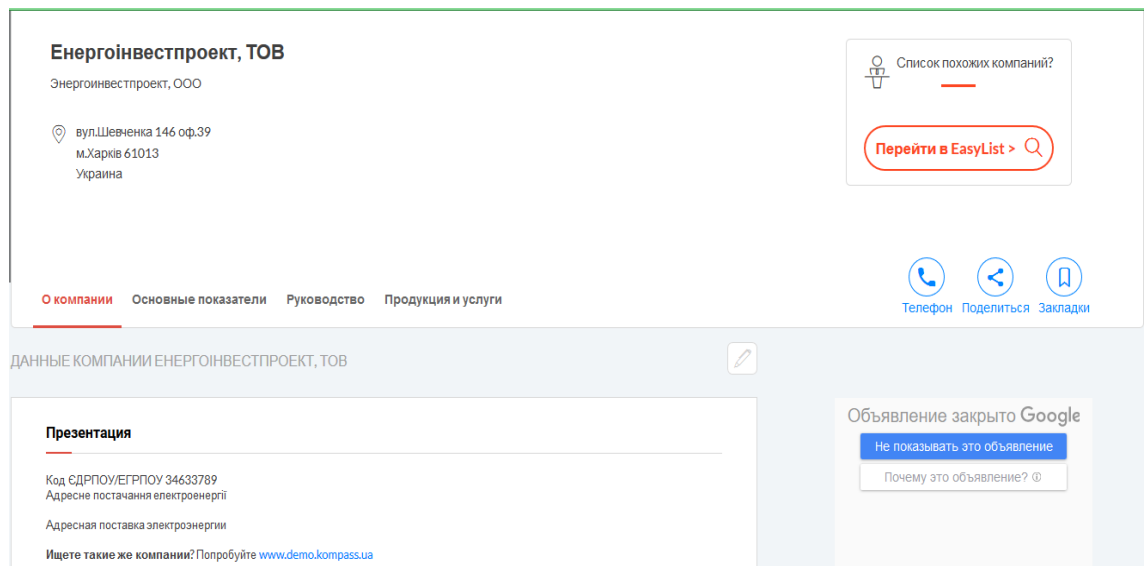


Рисунок 1.9 - Скриншот сайту ТОВ Енергоінвестпроект

На всіх розглянутих сайтах присутні розділи із загальною інформацією про фірму, а також наявне інформування про послуги, що надаються, містяться посилання на нормативні документи. На деяких сайтах є розділи, у яких вказані приклади виконаних робіт та відгуки клієнтів.

З огляду на вищесказане, можемо підсумувати, що був проведений аналіз існуючих на ринку рішень, а також основних груп користувачів. На основі цього аналізу складається технічне завдання, що містить вимоги до дизайну, структури, навігації, технічні вимоги та вимоги до CMS.

1.6 Визначення цільового користувача

Успіх інформаційного проекту прямо залежить від вибірки цільової аудиторії (сегмента ринку). Сегмент ринку складається зі споживачів, що однаково реагують на один і той же набір спонукальних стимулів маркетингу. Для визначення цільових сегментів потрібно скористатися матрицею Шеррингтона, так як, по-перше, даний метод універсальний, а по-друге, він дозволяє впорядкувати і систематизувати знання про споживача.

Марк Шеррингтон в своїй книзі «Додана цінність: алхімія зростання за допомогою бренду» пропонує відповіді на 5 запитань матриці (5W).

- 1) *What?* (Що?) – Товар
- 2) *Who?* (Хто?) – Користувач
- 3) *Why?* (Чому?) – Мотивація
- 4) *When?* (Коли?) – Момент
- 5) *Where?* (Де?) – Канал збуту

Отримані в ході застосування методики дані доцільно узагальнити для того, щоб зрозуміти, що являє собою продукт, ким є його споживач, коли і де споживач буде шукати продукт. Результати дослідження 5W можуть стати відмінним базисом у визначенні інформаційного портрету користувача (ІПК), виходячи з тих чи інших стійких характеристик (географічні, демографічні, економічні, тощо), оскільки частково містять необхідну інформацію.

Цільова аудиторія сайту – це група користувачів, зацікавлених в інформації, товарах або послугах, які надає сайт. Для того, щоб контент-проект активно розвивався, необхідно вивчити цільову аудиторію, сегментувати її і скласти портрет представника кожного сегмента [10]. Для цього існує набір маркетингових інструментів, раціональне використання яких може внести значний внесок в успіх проекту. Спробуємо проаналізувати цільову аудиторію за методом 5W (табл. 1).

Таблиця 1 - Сегментація ринку за методом Шеррінгтона (5W)

<i>What?</i> / Що?	Сайт «ЕНЕРГО Х»
<i>Who?</i> / Хто?	Фізичні та юридичні особи, які потребують постачання, електроенергії, проведення енергоаудиту, виконання монтажних робіт. Вік цільової аудиторії від 22 до 60 років, гендерна приналежність значення не має
<i>Why?</i> / Чому?	Необхідність отримати актуальну інформацію про тарифи, матеріали, вартість виконання робіт та ін. Бажання знайти нормативну документацію про постачання електроенергії, проведення енергоаудиту, про норми виконання електро-монтажних робіт
<i>When?</i> / Коли?	Цілорічно. Пік активності користувача спостерігається з березня по грудень.
<i>Where?</i> / Де?	Сайт, спільноти в соціальних мережах

Для того щоб уявлення про цільову аудиторію було повнішим, одного методу недостатньо, тому скористаємося одночасно методикою вербалізації інформаційного портрету користувача (ІПК).

Під час обговорення матеріалів і структури сайту ІПК використовуються замість абстрактних цільових груп. По завершенні робіт, можна провести тестування юзабіліті - досить запропонувати представнику конкретної цільової групи сторінку, що розроблена для відповідного ІПК, і дізнатися про його думку. Безумовно, жоден ІПК не може повністю замінити собою всю групу цільової аудиторії, але все ж таки може послужити відправною точкою. Основні параметри, необхідні для моделювання сайту ЕНЕРГО Х:

- а) демографічні характеристики: вік, стать, освіта, рід зайнять.
- б) психографічні характеристики: прагнення, недоліки, способи мотивації.

в) Інтернет: досвід користування, місце користування, пристрій виходу, частота використання, наявність реєстрації в соціальних мережах.

г) потреби ПК, необхідні для роботи над юзабіліті - досить 3–4.

д) визначення ключової персони - якщо будуть задоволені її мети, то, ймовірно, і цілі інших персон будуть досягнуті.

Кількість персонажів і ступінь деталізації їхнього опису визначається різноманітністю цільової групи. У міру отримання додаткової маркетингової інформації, персонажі будуть доопрацьовуватися і деталізуватися. В таблицях 2 – 5 ПК наведено характеристики відвідувачів будівельного сайту.

Таблиця 2 - ПК 1

Поважний Андрій Володимирович		
Соціально-демографічні характеристики	Технічні навички	Особисті дані
Вік: 60 Місто: Харків Освіта: вища Рід зайнятості: головний електрик	ПК: впевнений користувач Місце користування: вдома, в офісі Частота використання інтернету: щодня Пристрій входу: ноутбук	За освітою будівельник, працює головним електриком у великій фірмі. Займається питаннями енергопостачання свого підприємства
Соціальні мережі:	<i>Facebook, Twitter</i>	
Потреби		
1. Ознайомитися з тарифами на поставку електроенергії 2. Ознайомитися з відгуками 3. Задати питання менеджеру		

Таблиця 3 - ПК 2

Солодова Ольга Василівна (Ключова особа)		
Соціально-демографічні характеристики	Технічні навички	Особисті дані
Вік: 49 Місто: Харків Освіта: вища Рід зайнятості: завідувач дитячого садочка	ПК: впевнений користувач Місце користування: вдома, в офісі Частота використання інтернету: щодня Пристрій входу: ноутбук	За освітою педагог, працює завідуючою в дитячому садку. Займається питаннями забезпечення необхідними умовами для функціонування навчального закладу
Соціальні мережі:	<i>Facebook, Twitter, YouTube</i>	
Потреби		
<ol style="list-style-type: none"> 1. Ознайомитися з цінами на виконання енергоаудиту 2. Ознайомитися з цінами на виконання монтажних робіт 3. Ознайомитися з тарифами на постачання електроенергії 4. Ознайомитися з відгуками 5. Задати питання менеджеру 6. Задати питання головному інженеру 		

Таблиця 4 - ПК 3

Борисенко Оксана Володимірівна		
Соціально-демографічні характеристики	Технічні навички	Особисті дані
Вік: 32 Місто: Харків Освіта: вища Рід зайнятості: Головний менеджер	ПК: впевнений користувач Місце користування: вдома, в офісі Частота використання інтернету: щодня Пристрій входу: ПК, ноутбук	За освітою менеджер, працює на міні-виробництві. Займається питаннями постачання продукції свого виробництва
Соціальні мережі:	<i>Facebook, Twitter, YouTube</i>	
Потреби		
<ol style="list-style-type: none"> 1. Ознайомитися з цінами на поставку обладнання 2. Ознайомитись з цінами на виконання електромонтажних робіт 3. Ознайомитися з відгуками 4. Задати питання менеджеру 5. Задати питання головному інженеру 		

Таблиця 5 - ПК 4

Корольов Олексій Михайлович		
Соціально-демографічні характеристики	Технічні навички	Особисті дані
Вік: 22 Місто: Харків Освіта: неповна вища Рід зайнятості: студент	ПК: впевнений користувач Місце користування: вдома, в офісі, в кафе Частота використання інтернету: щодня Пристрій входу: ноутбук	Студент, вчиться в університеті, спеціальність будівництво, до навчання ставиться відповідально, є старостою групи.
Соціальні мережі:	<i>Instagram, YouTube</i> <i>Facebook, Twitter</i>	
Потреби		
<ol style="list-style-type: none"> 1. Ознайомитися з цінами на проведення енергоаудиту 2. Ознайомитись з цінами на виконання робіт 3. Ознайомитися з відгуками 4. Задати питання менеджеру 5. Задати питання головному інженеру 		

Таким чином, можна зробити висновки, що цільова аудиторія проекту досить велика. Її основні характеристики наступні:

- вік від 22 до 60 років.
- гендерна приналежність значення не має.
- геолокація – Україна.
- рівень достатку: середній і вище.

– переважно люди з вищою освітою, які займаються розумовою працею, а також студенти.

– споживачі - активні користувачі Інтернету та соціальних мереж.

Одним з дієвих методів підвищення лояльності користувачів є соціальні кнопки. Соціальні кнопки для сайту дозволяють одним натисканням додати до соціального ресурсу інформацію з сайту, що сподобалась, або статтю. Кнопки «Поділитися» дозволяють відвідувачам сайту ділитися контентом в соціальних мережах. Установка таких кнопок може значно збільшити трафік сайту, тому такі кнопки повинні бути доступні не тільки в статтях, але і в розсилках, на сторінках сайту і таке інше.

Кнопки соціальних мереж, що мають лічильники натискань, можуть надавати позитивний вплив на формування враження про сайт, так як вказують на рівень довіри відвідувачів до ресурсу. Чим вище цифра лічильника, тим більше це свідчить про високу зацікавленість інших відвідувачів і тим більше сприяє зацікавленості кожного нового відвідувача (рис. 1.10).

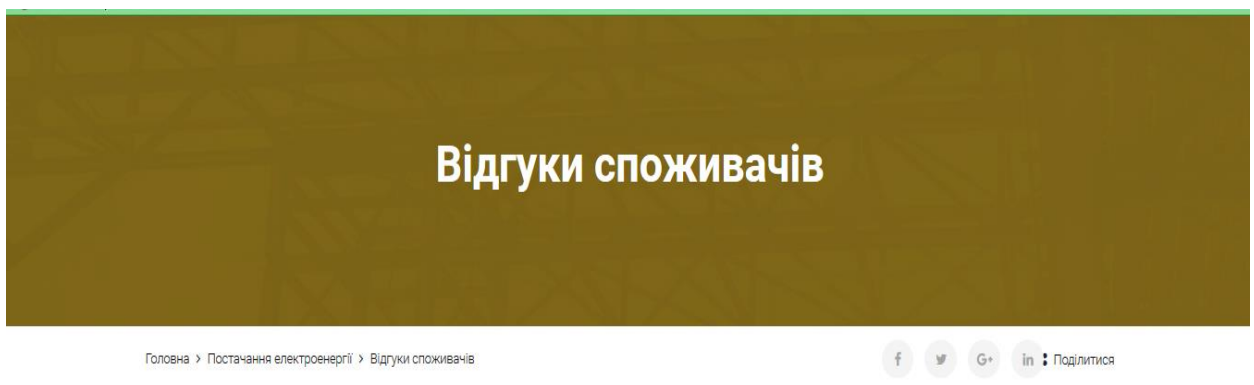


Рисунок 1.10 - Соціальні кнопки

Але підвищення лояльності відвідувачів - далеко не єдина функція соціальних кнопок. Завдяки ним можна провадити ефективніше просування сайту в пошукових системах і соціальних сервісах.

Під час аналізу сайтів конкурентів було відзначено, що ними також активно використовується соціальна кнопка для просування за допомогою соціальних

мереж. Зазвичай, використовуються такі мережі, як: Facebook, Twitter. Значно рідше - Instagram і Telegram.

Виходячи з аналізу цільової аудиторії сайту «ЕНЕРГО Х» було встановлено, що найбільшою популярністю користуються такі мережі, як: Facebook і Twitter. Але впроваджуючи на сайт віджет «Поділитися» з метою розширення потенційної цільової аудиторії проекту, слід додатково включити такі ресурси, як Instagram, Twitter і Google+.

Публікація прес-релізів компанії на різних ресурсах підвищує кількість зворотних посилань на сайт, сприяє зростанню популярності сайту серед аудиторії. Крім того, соціальні мережі активно індексуються пошуковими системами, а значить, додані прес-релізи максимально швидко проходять індексацію і «приводять» пошукового робота на основний сайт. В результаті індексація основного сайту відбувається швидше. Таким чином, слід ще на етапі розробки концепції планувати впровадження на сайт соціальних плагінів, так як вони підвищують довіру цільової аудиторії до ресурсу, що життєво важливо для тематичного проекту.

Подача великих масивів даних може бути ефективною із застосуванням методу сторітеллінгу (*storytelling*). Це засіб передачі інформації і винайдення змісту через розповіді історій. Як інструмент маркетингу він був успішно апробований Девідом Армстронгом, главою міжнародної компанії Armstrong International. На його думку, історія - це сюжетно-пов'язана розповідь. А історії легше асоціюються з особистим досвідом, ніж суха інформація. Вони краще запам'ятовуються, і їм надають більше значення, вони сильніше впливають на поведінку людей. Правильно розказана історія може впливати на людину і його вчинки більше, ніж звичайний спосіб передачі інформації. Історії інтерактивні і асоціативні. Використання цієї технології на сайті Енерго Х планується для розділу «Енергоаудит».

2 РОЗРОБКА МОДЕЛІ ВЕБ-САЙТУ

2.1 Склад і взаємозв'язок компонентів веб-сайту

Для сайту, який надає можливість надавати послуги, розробити модель адаптації, що дозволяє підлаштовуватися під смаки користувачів шляхом автоматичної генерації контенту для прискореного пошуку інформації та утримання клієнта.

Для досягнення поставленої мети необхідно вирішити такі завдання:

- розробити склад і взаємозв'язок компонентів веб-сайту;
- розробити узагальнену модель адаптації сайту;
- розробити критерії, що дозволяють оцінити якість функціонування передбачуваного ресурсу.

Функціональна система веб-сайту замовлення послуг з урахуванням галузі їх надання складається з наступних компонентів:

- мультиагентна система (МАС);
- система обробки даних користувача (СОДК);
- експертна система прийняття рішень (ЕСПР);
- база даних сайту (БД).

Мультиагентна система представляється трійкою (формула 1)

$$\begin{aligned} \text{МАС} &= \langle \Lambda, M, \Omega \rangle, \\ M &= \text{АК} \cap \text{АС} \cap \text{ТА}, \end{aligned} \quad (1)$$

де Λ - кінцева множина станів зовнішнього середовища;

M – групи агентів: агенти користувача (АК), агенти специфікації (АС), транспортні агенти (ТА);

$\Omega : \Lambda \times \text{АК} \times \text{АС} \times \text{ТА} \rightarrow 2^\Lambda$ - функція, що описує можливу реакцію зовнішнього середовища на дії всіх агентів системи.

Кожен з агентів має такі властивості:

- він є автономною Java або PHP-програмою, здатною сприймати навколишнє середовище і своєчасно реагувати на будь-які зміни або порушення,

які відбуваються в навколишньому середовищі з метою досягнення власних цілей;

- взаємодія з іншими агентами шляхом координації, спілкування і ведення переговорів за допомогою стандартизованого мови IDL (Interface Definition Language);

- можливість як споживати, так і поставляти інформацію, а також виступати з ініціативами на спрямовану поведінку;

- взаємодія з браузером, БД, ЕСПР, СОДК, а також можливість збереження, тобто здатність бути репозиторієм (сховищем) реалізацій для забезпечення зручного віддаленого доступу на вимогу (наприклад, мереж передачі даних, серверів, пристроїв зберігання даних, додатків і сервісів - як разом, так і окремо).

Взаємодія з користувачем, який вибирає потрібну йому предметну область, підтримується АК, що зв'язується з системою обробки документів користувача для аналізу інформаційного портрета користувача (ІПК), запитує відповідну онтологію з БД, а також інформує інших агентів про те, яка онтологія буде використовуватися.

Завданням транспортного агента є підключення до заданої URL веб-сторінки, її зчитування і аналіз, а також обробка виняткових ситуацій.

Агенти специфікації призначені для надання сервісних послуг. Наприклад, для розрахунку вартості монтажних робіт необхідні креслення, схеми, підключення обладнання, затверджені розцінки, ринкові ціни на матеріали і техніку, існуючий рівень заробітної плати робітників та інше. Для цього АС звертається до віддаленого сервісу для обробки результатів запиту, наприклад, витрати матеріалу і вартість утеплення фасаду в рамках програми енергозбереження.

Схема функціонування системи для задачі надання послуги наведена на рисунку 2.1.



Рисунок 2.1 - Схема функціонування системи для задачі надання послуги

Система обробки даних користувача (СОДК) здійснює такі дії:

- збір і зберігання доступних даних про користувача;
- аналіз даних про користувача і створення припущень;
- створення інформаційного портрету користувача при першому відвідуванні сайту;
- модифікація ІПК при кожному наступному відвідуванні сайту користувачем;
- формування змісту початкової сторінки сайту на основі інформації, що міститься в ІПК.

Експертна система прийняття рішень (ЕСПР) Застосування експертної системи прийняття рішень на Інтернет-ресурсі надання будівельних і ремонтних послуг дає величезний економічний ефект організації, що надає послуги і приносить велику користь користувачу, який здійснює пошук цих послуг. Це відбувається завдяки дистанційним можливостям програмної системи, що пропонує послуги, оскільки її база даних акумулює інформацію про складові

частини послуги, а також містить знання великої кількості висококваліфікованих фахівців, консультацією яких можна будь-коли скористуватися. Це також і інформація про сучасні матеріали, терміни служби, вартість, доречність їх застосування в тих чи інших випадках, порівняльні характеристики тощо.

Слід згадати ще й про бази даних - БД. Структура сайту зберігається безпосередньо в коді (в PHP-скриптах) і в окремих файлах, так званих шаблонах. Для зберігання контенту сайту використовується база даних. З точки зору движка веб-сайту база даних є набором таблиць. Кожна таблиця - це сутність, що зберігає однотипні дані.

2.2 Модель адаптації веб-сайту

Адаптація веб-сайту R виражається як перетворення вхідних значень P у вихідні величини Λ (формула 2):

$$R \subset P \times \Lambda, \quad (2)$$

де $P(i) = [P_U^a(i), P_U^s(i), P_U^h(i), P_U^b(i)]$ - інформаційний портрет i -го користувача (ПК) являє собою сукупність всіх наявних даних про користувачів, які зберігаються в базі даних;

$P_U^a(i)$ - доступні дані про користувача;

$P_U^s(i)$ - дані, засновані на сукупності вихідної інформації;

$P_U^h(i)$ - дані, засновані на припущеннях;

$P_U^b(i)$ - аналіз поведінки користувача на веб-сайті;

$\Lambda = \{ \Lambda_1, \Lambda_2, \dots, \Lambda_Q \}$, $(\lambda \subset \Lambda)$ - безліч інтерфейсів веб-ресурсів.

Отже, простір $E = P \times \Lambda$ містить $R \subset (P \times \Lambda)$, що означає, що існує така підмножина ПК $P = \{P(1), P(2), \dots, P(\tau)\}$ і відносин між ними, на яких будується модель R . Таким чином, для своєчасного забезпечення користувача інформацією

адаптація інтерфейсу веб-ресурсу на основі його інформаційного портрету $P(i)$ налаштовується завдяки моделі R при кінцевому числі образів Q .

Для вихідних величин Λ побудовано безліч завдань, вирішення яких належить множині (формула 3)

$$D_P = D_{quest} \cup D_{gather} \cup D_{SOM} \cup D_{service}, \quad (3)$$

де D_{quest} - завдання складання вектора опитування (вік, стать, країна, мова);

D_{gather} - задача автоматичного збору інформації про користувача (тематика запитів, час відвідування, тип браузера, роздільна здатність екрана, cookies та ін.);

D_{SOM} - задача кластеризації користувачів за допомогою мережі Кохонена;

$D_{service}$ - задача надання сервісу.

Відображення $\Pi : P \rightarrow \Lambda$ дозволяє для кожного $P(i)$ отримати таке $\lambda_j \in \Lambda$ ($j = 1, Q$), що є рішенням задачі D_P , отриманих у вигляді розподіленої моделі агентно-орієнтованого веб-ресурсу.

Для оцінки якості процесу функціонування пропонованої системи R складемо сукупність критеріїв оцінки ефективності $K = \{k_1, k_2, k_3\}$. Ці функції дозволяють визначити реакцію адаптації сайту.

Одним з головних критеріїв є ступінь релевантності (формула 4):

$$k_1 : \varpi = \max(\varpi), \quad (4)$$

де ϖ - ступінь релевантності відображеної інформації.

Іншим критерієм є час адаптації веб-ресурсу (формула 5):

$$k_2 : \tau = \min(\tau^s, \tau^d), \quad (5)$$

де τ^s - час статичної генерації сторінок;

τ^d - час динамічної генерації.

Співвідношення безлічі реально відображених розділів характеризуються такими характеристиками (формула 6):

$$\psi = \{\rho, \eta, e, q, v\}, \quad (6)$$

де $\rho = \frac{a}{a+c}$ - коефіцієнт повноти, що характеризує частку релевантних сторінок на веб-сайті до загальної кількості релевантних документів;

$\eta = \frac{a}{a+b}$ - коефіцієнт точності, що характеризує частку відображених релевантних сторінок на веб-сайті до загального обсягу видачі;

$e = \frac{b}{a+b} = 1 - \eta$ - коефіцієнт шуму, що характеризує частку відображених нерелевантних документів у всій безлічі відображених документів;

$q = \frac{b}{b+f}$ - коефіцієнт осаду, що характеризує частку відображених нерелевантних документів у всій безлічі нерелевантних документів;

$v = \frac{f}{b+f}$ - коефіцієнт специфічності, що характеризує частку невідображення нерелевантних документів у всій безлічі нерелевантних документів;

a - кількість адаптованих релевантних розділів;

b - кількість адаптованих нерелевантних розділів;

c - кількість неадаптованих релевантних розділів;

f - кількість неадаптованих нерелевантних розділів.

Для ефективного функціонування запропонованої моделі необхідно, щоб система задовольняла наступним вимогам (формула 7):

$$k_3 : \forall (d_i \in D_P) [(\tau < \tau^{\max}) \& (\varpi > \varpi^{\min}) \& (\rho, \eta \rightarrow 1) \& (e, q, v \rightarrow 0)] \Rightarrow R, (7)$$

де d - підзавдання загальної задачі D ;

τ^{\max} - максимально допустимий час вирішення завдання;

ϖ^{\min} - мінімально допустима релевантність.

При моделюванні отриманих теоретичних результатів адаптації веб-сайту були систематизовані користувачі будівельного сайту.

2.3 Вибір засобів реалізації

Сайт, що досліджується, є багатосторінковим сайтом, статичні сторінки якого мають один шаблон. Нестандартні сторінки сайту будуть реалізовані вручну. Контентом сайт буде наповнюватися авторами - некомпетентними в області розробки сайтів людьми. Таким чином, потрібна проста, інтуїтивно зрозуміла CMS, що надає можливість створювати свої шаблони сторінок.

Розглянемо деякі найбільш популярні системи управління контентом сайту:

WordPress [20] - система управління вмістом сайту з відкритим вихідним кодом; написана на PHP; сервер бази даних - MySQL; випущена під ліцензією GNU GPL версії 2. Сфера застосування - від блогів до досить складних новинних ресурсів і інтернет-магазинів.

Вбудована система «Тем» і «плагінів» разом з вдалою архітектурою дозволяє конструювати проекти широкої функціональної складності.

Переваги:

- зручний візуальний редактор;
- підтримка власних тем;
- велика кількість безкоштовних плагінів;
- проста інтеграція з соціальними мережами і сервісами аналітики;
- можливість створення нових і редагування існуючих плагінів;
- управління правами користувача.

Недоліки:

- стандартний функціонал (без розширень) дозволяє створити тільки блог або сайт-візитку;
- для створення мета-даних сторінки необхідно встановити плагін або змінити код теми.

Joomla! [19] - система управління вмістом сайту, написана на мові PHP і JavaScript, що використовує в якості сховища бази даних СУБД MySQL або інші

стандартні промислові реляційні СУБД. Є вільним програмним забезпеченням, поширюваним під ліцензією GNU GPL.

Переваги:

- можливість створення і редагування плагінів;
- створення мета-даних для кожної сторінки є стандартною функцією;
- управління правами користувачів входить в стандартний функціонал;
- публікація контенту на кількох мовах без плагінів.

Drupal [18] - система управління вмістом сайту, яка використовується також як каркас для веб-додатків (CMF), написана на мові PHP і використовує як сховище даних реляційну базу даних (підтримуючи ються MySQL, PostgreSQL та інші). Drupal є вільним програмним забезпеченням, захищеним ліцензією GPL, і розвивається зусиллями ентузіастів з усього світу.

Переваги:

- функція канонічних URL включена за замовчуванням;
- велика кількість стандартних функцій для розробки корпоративного сайту;
- дуже гнучке управління правами користувачів;

Недоліки:

- неповноцінний редактор (для розширення функціоналу необхідно розширення);
- для створення і редагування мета-даних необхідний SEO-модуль (розширення);
- порівняно невелика кількість безкоштовних розширень.

1С-Бітрікс [17]: для зберігання даних сайту використовується файлова система сервера і реляційна СУБД. Підтримуються наступні СУБД: MySQL, Oracle, MS SQL. Продукт працює на Microsoft Windows і UNIX-подібних платформах, включаючи Linux. Кількість модулів у встановлюваній системі залежить від редакції продукту.

Переваги:

- безліч стандартних функцій для розробки будь-якого виду сайту;

- інтеграція з 1С;
- низька ймовірність зараження сайту вірусом через закритого вихідного коду

Недоліки:

- дана CMS є платною;
- відсутність полігонів;
- вимагає використання певних серверних модулів і версій програмного забезпечення для коректної роботи всіх модулів системи.

На основі аналізу декількох CMS було прийнято рішення використовувати CMS WordPress, тому що дана система управління сайтом дозволяє створювати і впроваджувати свої шаблони і теми. Крім того, на рішення вплинула велика кількість плагінів для WordPress у вільному доступі, за допомогою яких можна буде вирішити такі завдання як багатомовність, Медіагалерея і настройка ролей користувачів.

WordPress - система керування вмістом сайту з відкритим вихідним кодом; написана на PHP; сервер бази даних - MySQL. Бажані функціональність і зовнішній вигляд досягаються за допомогою застосування тем і плагінів.

Запис - структурована інформація, яка містить наступні поля [22]:

- заголовок;
- дата;
- тема запису;
- стан публікації (опубліковано, на твердженні, чернетка);
- постійне посилання;
- мітки;
- цитати;
- рубрики;
- коментарі;
- довільні поля - дозволяє додати новий тип поля.

Нові типи записів можуть створюватися функцією `register_post_type($post_type,$args)`, де `$post_type` - ім'я нового типу

записів, `$args` - масив аргументів для найменування або завдання значення різних параметрів створення запису. Функція визначається у файлі шаблону `functions.php`.

Сторінка - контейнер для вмісту (записів або статичної інформації), яка формується на основі шаблону. Сторінки можуть використовувати різні шаблони, які можуть включати в себе інші шаблони або код PHP [22].

Тема - набір файлів, які відповідають за зовнішнє подання сайту. Одна тема може містити будь-яку кількість шаблонів [22].

Шаблон - файл, який контролює відображення сторінок і записів на сайті. Ці файли беруть інформацію з бази даних СУБД MySQL, генерують html-код і відображають її на сторінках сайту [22].

Блог - сторінка, що складається із записів [22].

Модулі - це програмне додаток для внесення додаткової функціональності в роботу сайту. З технічного боку плагін складається з набору файлів `.php`, всередині яких міститься програмний код, який додає нові можливості [22].

На підставі проведеного аналізу моделі програмної реалізації сайту дійшли висновку до доцільність використання шаблонів для отримання можливості налаштування змісту сайту на предметну область. Перейдемо до проектування та програмної реалізації системи.

3 СИСТЕМА РЕАЛІЗАЦІЇ САЙТУ

3.1 Розробка дизайну веб-сайту

Відповідно до технічного завдання і фірмового стилю, який відбивається в кожному елементі дизайну [9] слід розробити ескізи трьох варіантів дизайну для різних пристроїв: десктоп (рис. 3.1), планшет (рис. 3.2), мобільний (рис. 3.3), дотримуючись у кожному з варіантів правила модульних сіток [8].



Рисунок 3.1 - Дизайн головної сторінки сайту для версії «десктоп»

Дизайн має бути функціональним і зручним одночасно [2]. Для цього для

всіх сторінок сайту передбачається розробка однієї шапки, що містить логотип фірми. Нижче розміщується однакова для всіх сторінок меню – зручна навігація по сайту.

Крім шапки на кожній сторінці однаково відображується футер, де вказуються контакти, посилання на сторінки підприємства в соціальних мережах для зручності пошуку контактів.

Для всіх сторінок сайту передбачається єдиний стиль – заголовки й основний текст мають відображатися однаково на будь-якій сторінці сайту. Наприклад, другорядна сторінка «Контакти» (рис. 3.4).

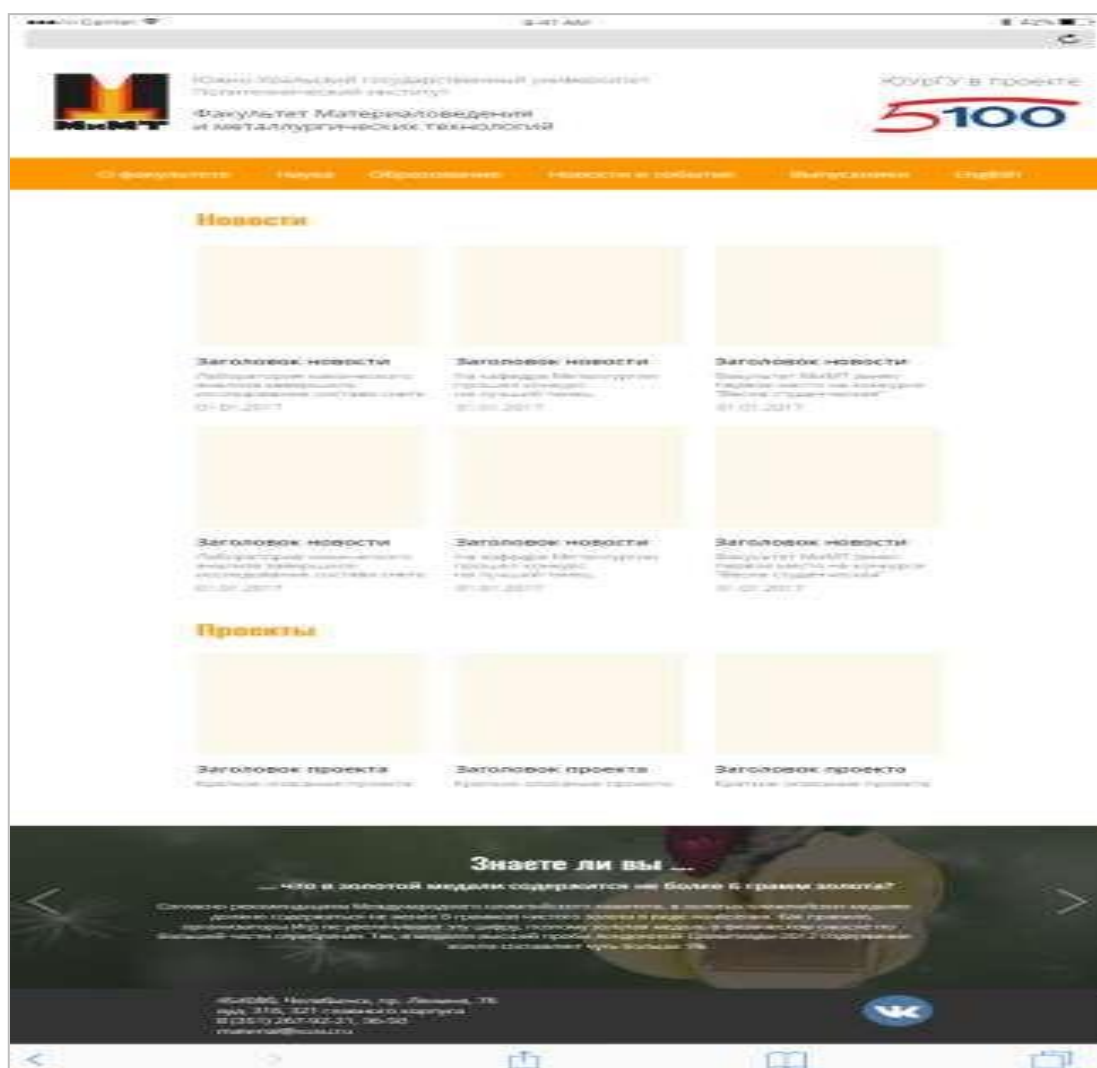


Рисунок 3.2 - Дизайн головної сторінки сайту для планшета



Рисунок 3.3 - Дизайн головної сторінки сайту для мобільних пристроїв



Рисунок 3.4 - Дизайн другорядної сторінки «Контакти»

3.2 Діаграма варіантів використання UseCase

На основі технічного завдання складається діаграма прецедентів, що наведена на рисунку 3.5.

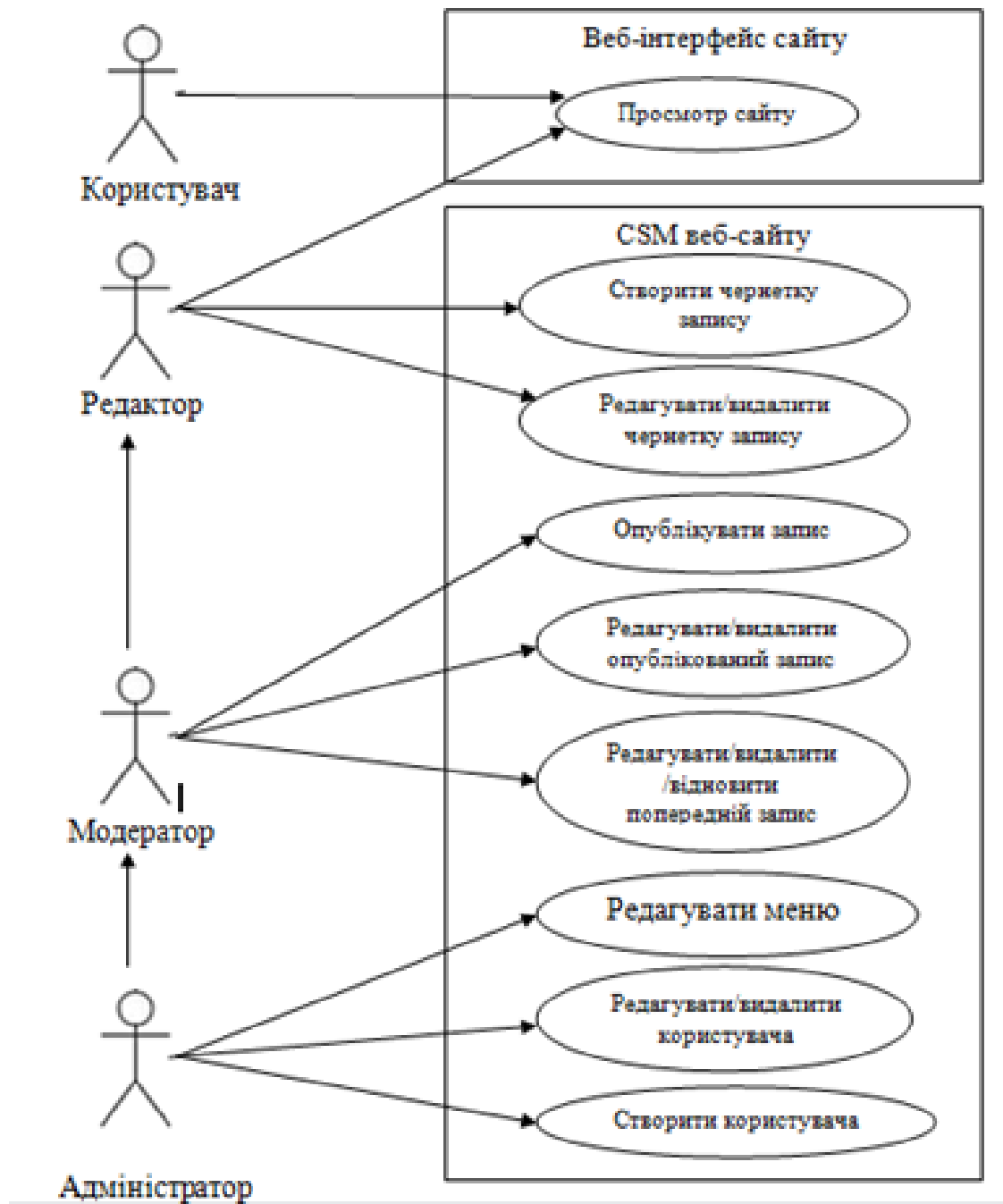


Рисунок 3.5 - Діаграма варіантів використання

Зробимо короткий опис варіантів використання:

- перегляд сайту: перехід по посиланнях основного меню і меню, що випадає;
- створити чернетку записи: користувач створює запис одного з доступних видів і відправляє її на модерацію;
- редагувати / видалити чернетку записи: зміна полів ще неопублікованої записи;
- опублікувати запис: чернетка записи публікується на сайті у відповідному розділі;
- редагувати / видалити опубліковану запис: зміна полів або видалення опублікованої на сайті записи з відповідного розділу;
- редагувати попередню версію сторінки: зміна полів сторінки, її видалення або відновлення попередньої опублікованої версії цієї сторінки
- редагувати меню: змінити рівень, місце пункту меню або його видалення;
- редагувати / видалити користувача: редагувати привілеї користувача або його повне видалення;
- створити користувача: реєстрація в CMS нового користувача.

В якості основних акторів прийняті ролі авторизації:

- користувачі (без авторизації) - відвідувачі сайту;
- редактори - редактори розділів сайту (з авторизацією в CMS) - співробітники фірми, що мають права для редагування певних розділів сайту;
- модератори сайту (з авторизацією в CMS) - співробітники фірми, що мають права для редагування всіх розділів сайту, які здійснюють модерацію новин, редагування всіх розділів сайту і меню, відновлення попередніх версій сторінок Сайту;
- адміністратор (з авторизацією в CMS) – співробітник (керівник) фірми, що забезпечує реєстрацію і визначення ролей користувачів, модерації новин, редагування всіх розділів сайту і меню, відновлення попередніх версій сторінок Сайту.

3.3 Розробка теми в системі WordPress

Оскільки розроблюваний дизайн оригінальний та не має схожих аналогів, він потребує розробки власної теми для WordPress [3]. Для коректного функціонування теми сайту, для якого активована ця тема, необхідні файли, які описані в попередніх розділах. Слід створити й наповнити такі файли.

1. Шаблон шапки сайту – *header.php*.

У відповідності до розроблених макетів шапка повинна містити логотип й назву, а також посилання на головний офіс, і ще посилання на сторінку проекту, і, крім цього випадаюче меню.

Меню створюється за допомогою стандартної функції WordPress `wp_nav_menu()`; аргумент якої – назва меню в темі WordPress. Адаптивність меню та його представлення у вигляді кнопки, після натиснення на яку меню розкривається, на мобільних пристроях забезпечено за допомогою класів `navbar` и `navbar- default` фреймворка `bootstrap` и стилів `CSS` [11].

Далі, в кожному шаблоні, слід використати підключення шапки за допомогою коду: `<?php get_header();?>`.

2. Шаблон футеру сайту – *footer.php*.

В футері сайту слід розмістити контакти організації, посилання на сторінки в соціальних мережах.

Існує Гугл. Метрика – сервіс для підрахунку відвідувань й унікальних відвідувачів, а також складання статистики використання сайту. Для підключення даного сервісу до сайту, необхідно зареєструватися або авторизуватися за допомогою будь-якої соціальної мережі з подальшою реєстрацією лічильника. Для цього потрібно вказати адресу сайту та можливі додаткові адреси. Далі слід налаштувати додаткові можливості. Наприклад, налаштування - інформер, це віджет для сайту, що відображує кількість переглядів сторінки, візитів й унікальних відвідувачів. Інформер можна розмістити на своєму сайті, використовуючи код, який генерується сервісом Гугл. Метрика після

налаштування лічильника.

Як і шапка сайту, підвал підключається до кожної сторінки в шаблоні за допомогою функції `<?php get_footer();?>`.

3. Файл функцій, що підключаються – *functions.php*.

В даному файлі реалізовано підключення шрифтів за допомогою функції `function enqueue_styles();` наступний код реалізує підключення меню до теми:

```
if (function_exists ('add_theme_support'))
{ add_theme_support('menus'); }.
```

Папка теми містить підпапки «js» и «css», в яких розміщуються файли *bootstrap.min.js* і *bootstrap.min.css*. В файлі *functions.php* реалізовано підключення bootstrap за допомогою підключення цих файлів як файлу скриптів й файлу стилів відповідно.

Тут же реалізовано створення типів записів для новин за допомогою функції `function register_post_type()` [16]. Повний лістинг наведено на рис. 3.6. Аналогічно реалізується створення типу записів іншого призначення на головній сторінці.

Крім власних типів записів в цьому файлі прописується реалізація своїх полів для записів, наприклад, поле додаткового опису послуг.

4. Шаблон головної сторінки сайту – *front-page.php*.

На головній сторінці передбачається розміщення розділів «Новини», «Проекти» й каруселі «Чи знаєте ви...?». Вміст всіх розділів визначається динамічно – отримуються останні записи типу *news*, *projects* и *didYouKnow* відповідно. Реалізація створення типів записів описана вище, в п.3 (*functions.php*).

Під час створення запису всі її поля заносяться в базу даних в таблицю «wp_posts». Відповідно, щоб отримати дані про запис для розміщення, необхідно зробити запит в цю таблицю. За допомогою функції `<?php $query = new WP_Query("post_type='news'"); ?>` [8] отримуємо дані про всі записи типу *news*. Далі, за допомогою функції `$query->the_post();?>` отримуємо дані про кожен наступний запис такого типу, а функції `the_title();` и `the_excerpt();` дозволяють отримати дані про поля заголовків й уривків відповідно.

```

add_action('init',
'news_post_register'); function
news_post_register()
{
    $labels = array(
        'name' => 'Новость',
        'singular_name' =>
        'Новость', 'add_new' =>
        'Добавить новость',
        'add_new_item' => 'Добавить новую
        новость', 'edit_item' => 'Редактировать
        новость', 'new_item' => 'Новая новость',
        'view_item' => 'Посмотреть
        новость', 'search_items' =>
        'Найти новость', 'not_found'
        =>
            'Новостей не
            найдено',
        'not_found_in_trash' => 'В корзине новостей не найдено',
        'menu_name' => 'Новость' );
    $rewrite = array('slug' => 'новости', );
    $args = array(
        'labels' =>
        $labels, 'public'
        => true,

```

Рисунок 3.6 - Лістинг функції створення нового типу запису *news*

5. Шаблон сторінки стрічки новин – *blog-page-news.php*.

В даному шаблоні реалізовано отримання всіх постів типу *news* із бази даних за допомогою функції `$recent = new WP_Query("showposts=$n&post_type=$type")`. В параметрах вказаний тип записів і кількість постів, що виводяться на сторінці. Далі в циклі виводяться основні поля: прикріплений медіа-файл, поле заголовку і анотації.

Лістинг наведено на рис. 3.7.

6. Шаблон сторінки проєктів – *blog-page-projects.php*.

Принцип отримання даних із бази аналогічний з попереднім пунктом. Різниця між шаблонами у відображенні: отримані записи проєктів фільтруються на ті, які завершені та поточні. Фільтр реалізовано шляхом додавання на сторінку форми html: при зміні значення фільтра сторінка оновлюється, відображуючи тільки ті записи, тип яких указаний у фільтрі.

```

<?php
    $n=4; // количество выводимых записей
    $type="news";
    $recent = new
    WP_Query("showposts=$n&post_type=$type");
    while($recent->have_posts()) : $recent-
    >the_post();

    ?>
<div class="row">
    <div class="col-lg-3" style="margin-top: 20px;">
        <a href="<?php the_permalink() ?>" rel="bookmark">
            <?php
                $media = get_attached_media( 'image', $post->ID );
                $media = array_shift( $media );
                $image_url = $media->guid;?>
                <?php echo '';?>
            </div>
            <div class="col-lg-7" style="margin-top: 30px;">
                <br> <h4><?php the_title();?> </h4><br></a>
                <p><?php the_excerpt();?></p>
            </div>

```

Рисунок 3.7 - Лістинг відображення постів в стрічці новин

Тип запису (завершений проект або поточний) вказується під час його створення з CMS: реалізується довільне поле запису. Дане поле редагується також користувачем з CMS.

7. Шаблон сторінки запису – *single.php*.

Даний шаблон призначений для відображення повного тексту запису типу новина або проект. Крім повного тексту, сторінка містить велику версію фото, прикріпленого до запису. Реалізується відображення за допомогою отримання поточного запису із бази даних. Для покращання індексації сайту пошуковими системами повний текст запису, що виводиться обгортуються в тег `<main>`.

8. Шаблон сторінки – *page.php*. Такий шаблон використовується за замовченням для сторінок, для яких в CMS не вказаний спеціальний шаблон. Сторінка повністю заповнюється з CMS й код шаблону містить виключно отримання заголовка сторінки, зображення й основного вмісту.

Щоб розробити власну тему в CMS WordPress, необхідно створити і заповнити вмістом наступні файли:

- *style.css* - головний файл таблиці стилів. Файл повинен бути в темі обов'язково і обов'язково повинен містити коментарі в заголовку файлу
- *index.php* - головний файл шаблонів. При використанні своїх шаблонів цей файл обов'язковий. Може містити код головної сторінки, може бути використаний як шаблон для всіх сторінок;
- *comments.php* - шаблон коментарів. Якщо його немає, використовується файл *comments.php* з теми за замовчуванням;
- *comments-popup.php* - додає спливаюче меню для коментарів на JS. При відсутності викликається *comments-popup.php* з теми за замовчуванням;
- *home.php* - шаблон головної сторінки блогу;
- *single.php* - шаблон сторінки одиночного поста. Використовується, коли запитаний конкретний пост. Для цього і інших шаблонів за запитом використовується *index.php*, якщо в темі немає відповідного файлу шаблону;
- *page.php* - шаблон окремої сторінки. Якщо для сторінки не вказано шаблон, *page.php* застосовується за умовчанням;
- *category.php* - шаблон категорії *category template*. Використовується при запиті категорії;
- *author.php* - шаблон автора блогу *author template*. Використовується при запиті автора блогу;
- *date.php* - шаблон виводу дати-часу. Використовується при запиті дати і часу. Рік, місяць, день місяця, година, хвилина, секунда;
- *archive.php* - шаблон архіву. Використовується, коли запитані категорія, автор або дата. Цей шаблон може бути перевизначений шаблонами *category.php*, *author.php* і *date.php* для відповідних типів запитів;
- *search.php* - шаблон результатів пошуку. Використовується після виконання пошуку;
- *functions.php* - файл для додавання функцій і можливостей в тему WordPress. По поведінці аналогічний з плагіном;

– *404.php* - шаблон для повідомлення про помилку 404 Not Found. Використовується коли WordPress не може знайти повідомлення або сторінку, яка відповідає запиту.

Обов'язковою з усіх є тільки файл *index.php*, але його заповнення не обов'язково, необхідність є тільки в його існуванні. Інші файли шаблонів теми використовуються за бажанням або необхідності.

Відповідно до структури сайту, вимогам до динамічних сторінок і необхідної структурою теми Word-Press була розроблена схема теми для веб-сайту. Схема розробленої теми представлена на рис. 3.8.

На схемі кожному розділу сайту - шапка, підвал, стилі, контент і додатковий функціонал - відповідають окремі файли. Контенту відповідає кілька файлів - розділ входять головна сторінка, внутрішні динамічні сторінки, шаблон статичних, що заповнюються з CMS сторінок і сторінки з окремим записом.

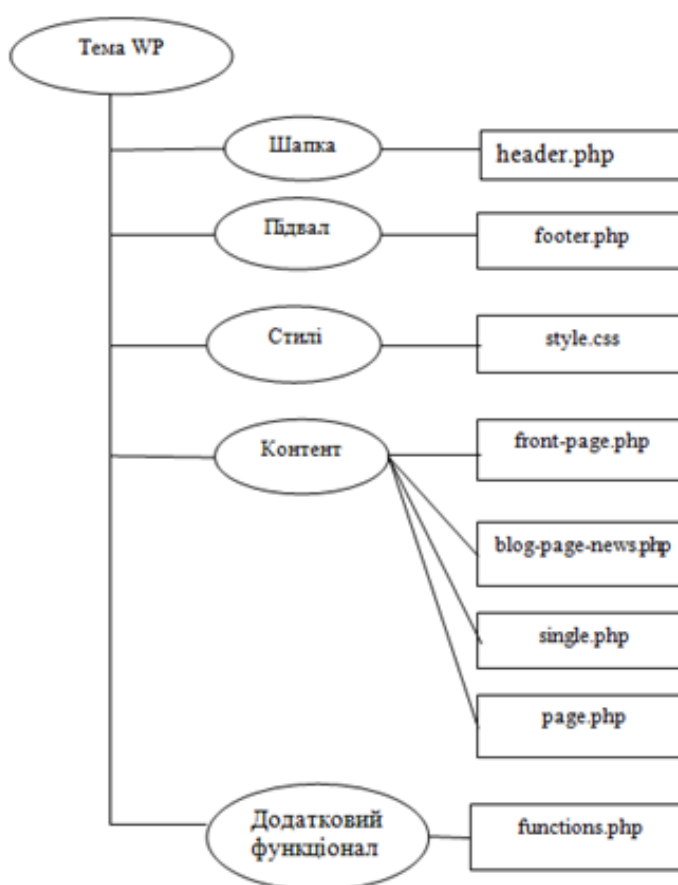


Рисунок 3.8 - Структура теми веб-сайту

3.4 Користувачі WordPress

В системі WordPress існують наступні стандартні користувачі: адміністратор WordPress, редактор WordPress, автор WordPress, неавторизований користувач.

Для створення користувачів сайту – Адміністратор, Модератор, Редактор, Відвідувач сайту (див. п. 1.1) – можливо використовувати лише стандартних користувачів WordPress:

- адміністратор сайту – адміністратор WordPress;
- модератор сайту – редактор WordPress;
- редактор сайту – автор WordPress;
- відвідувач сайту – неавторизований користувач.

Для коректної взаємодії користувачів доречно провести коректировку привілеїв. Нові привілеї наведені в таблиці 6.

Таблиця 6 - Привілеї ролей користувачів

	Редактор	Модератор	Адміністратор
Перегляд сторінок сайту	+	+	+
Створення чернеток записів	+	+	+
Публікація записів	-	+	+
Видалення чернеток записів	+	+	+
Видалення / редагування опублікованих записів	-	+	+
Видалення / редагування сторінок	-	+	+
Створення/редагування/видалення користувача	-	-	+

3.5 Розробка бази даних

В базі даних зберігається весь контент, розміщений на сайті, включаючи видалені записи і сторінки. Таблиці бази даних для будь-якого сайту, створеного за допомогою CMS WP наведені у таблиці 7.

Таблиця 7 - База даних WordPress

Таблиця	Дані
wp_posts	Записи, сторінки, вкладення, редакції, призначені для користувача записи
wp_postmeta	Метадані записів, сторінок і т.д.
wp_comments	Коментарі
wp_commentmeta	Метадані коментарів
wp_term_relationships	Зв'язки між таксономії і записами, сторінками і т.д.
wp_term_taxonomy	Таксономії (включаючи категорії і мітки)
wp_terms	Категорії, мітки і терміни призначених для користувача таксономій
wp_links	Посилання в блоці
wp_users	Користувачі
wp_user_meta	Метадані для кожного користувача
wp_options	Опції і налаштування сайту

База даних створюється системою за умови наявності WordPress на сервері. Нижче на рисунку 3.9 наведена схема бази даних WordPress

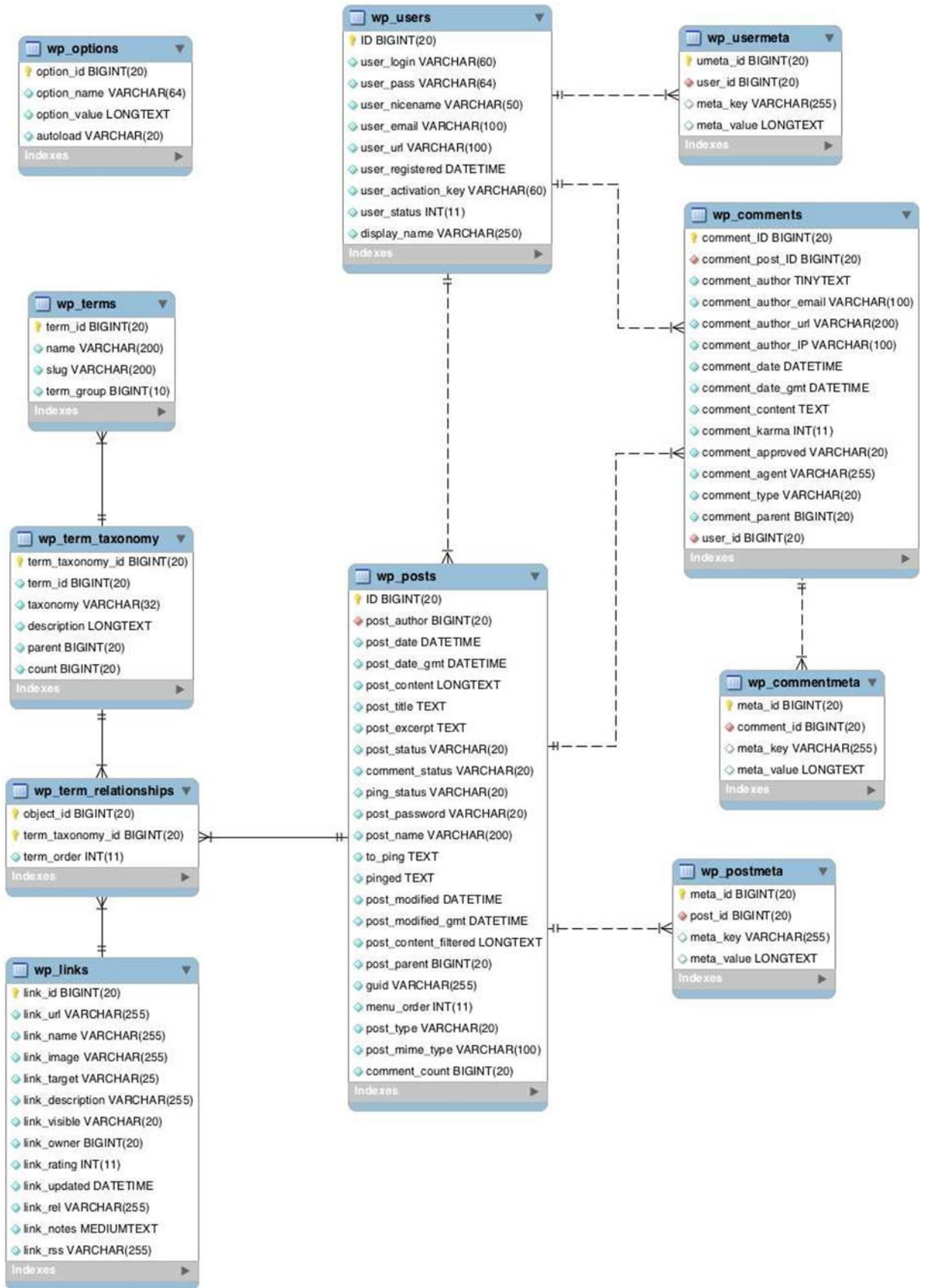


Рисунок 3.9 - Схема бази даних WordPress

3.6 Розробка багатомовності та медіагалереї

Для реалізації багатомовності був встановлений і запроваджений плагін Polylang. До можливості даного плагіну входить прикріплення перекладу окремих елементів сайту – сторінок, записів, коментарів тощо. Таким чином ми отримуємо можливість публікації перекладу лише частини сайту. Переклад в даному плагіні не є автоматичним, тобто для кожної сторінки або запису чи коментаря і таке інше необхідно вручну складати переклад. Переклад відбувається завдяки створенню нової сторінки з подальшим заповненням її новим контентом мовою переклада.

Крім контенту, необхідно складати переклад меню. Нове меню складається із сторінок-перекладів, які створено. Для закріплення нового меню за версією сайту іншою мовою, в тему додається підтримка областей теми за допомогою функції `register_nav_menus()`, що додається в файл `functions.php`. Потому в самій CMS зв'язується область теми іншою мовою із створеним меню.

Для реалізації фото-відео галереї використовується плагін WordPress NextGEN Gallery. За його допомогою на одній сторінці можна розміщувати одну чи декілька галерей, які складаються з будь-якої кількості зображень, клікнувши яке отримуємо можливість передивитися фото на всю сторінку.

Галерея в даному плагіні виконує роль альбому, таким чином на одній сторінці може бути розміщено декілька альбомів. При публікації галереї на сторінці можна додати перед модулем назву розділу.

Для впровадження галереї в тему, копіюється код, який генерується плагіном код і вставляється в код сторінки.

3.7 Реалізація адаптивності

Як згадувалось вище, для реалізації адаптивності використовується фреймворк Bootstrap 3 [23]. Фреймворк підключається таким чином: з офіційного сайту `bootstrap-3.ru` завантажується мінімальний пакет. З нього необхідно перемістити в папки `css` и `js` файли `bootstrap.min.css` й `bootstrap.min.js` відповідно. Далі ці файли підключаються до теми за допомогою тегів `<link>` и `<script>` в шаблоні шапки. Основний принцип даного фреймворку – 12-колончаста розмітка (рис. 3.10-3.12). При реалізації верстки використовуються класи `col-*-**`, де замість `*` тип пристрою (`xs`, `md`, `lg`), а замість `**` кількість колонок, які займає елемент (1-12).

Первісна верстка інтерфейсу мовою розмітки `html` [12] повинна бути перетворена в адаптивну. Після підключення фреймворка стають доступними такі основні класи блоків як `container`, `row`, `navbar` інші.

Таким чином, весь вміст сторінки, за виключенням шапки й футера, обгортається в блок `container`, а внутрішні блоки, наприклад, записи в стрічці новин, обгортаються в блоки класів `col-*-**`, та їм задається ширина, відповідна кількості колонок на сторінці.

В деяких випадках вміст обгортається в блок класу `row` для забезпечення правильного відображення без зміщення за вертикаллю. Наприклад, рядок новин і проектів на головній сторінці.



Рисунок 3.10 - 12-колончаста верстка для версії «десктоп»



Рисунок 3.11 - 12-колончатая верстка для планшетів



Рисунок 3.12 - 12-колончатая верстка для мобільних пристроїв

При моделюванні отриманих теоретичних результатів адаптації веб-сайту були систематизовані користувачі будівельного сайту.

Для користувача, що вперше відвідав сайт «ТОВ Енерго Х» відобразиться початкова сторінка, зображена на рис. 3.13, яка містить загальну інформацію з можливостями переходу на всі розділи сайту - енергоаудит, енергопостачання, електромонтаж, контакти.

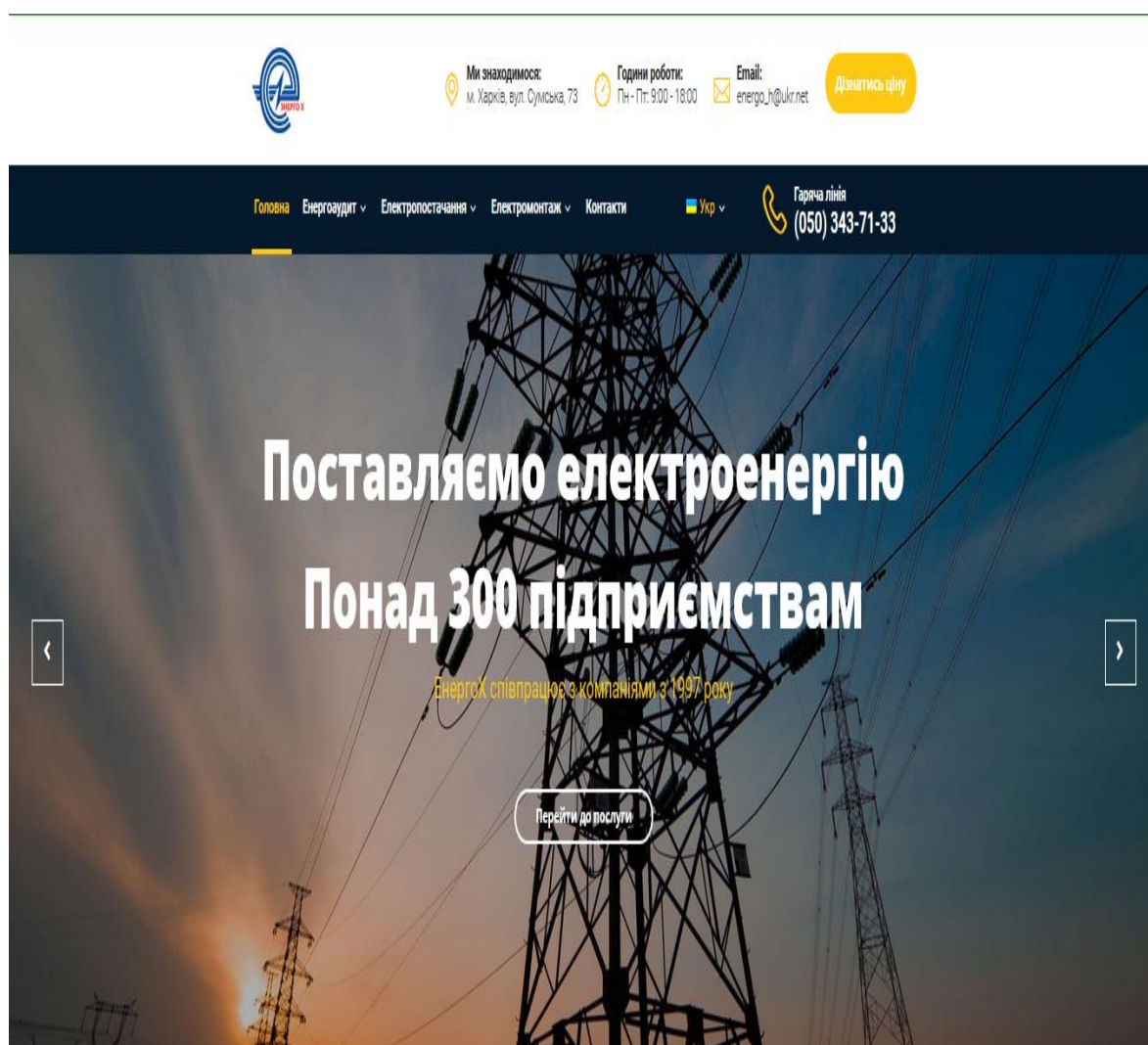
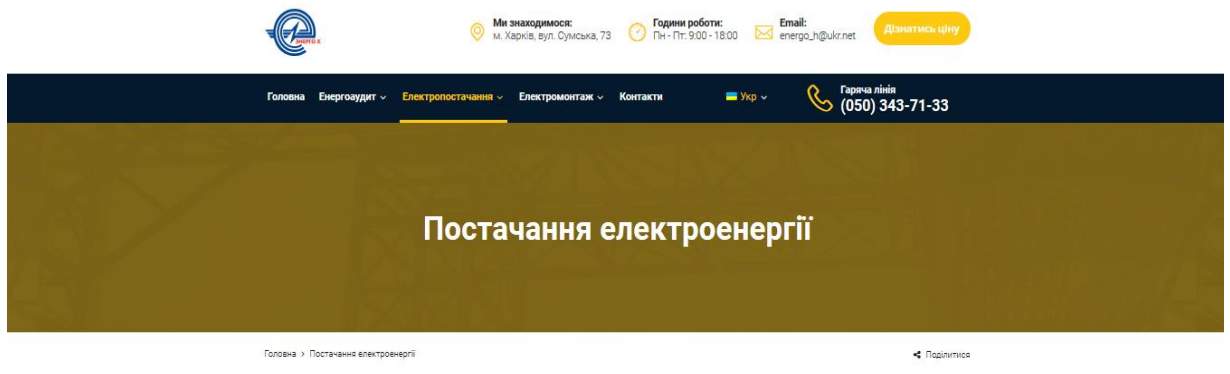


Рисунок 3.13 - Категорії користувачів сайту

Для користувача, який цікавиться купівлею електроенергії, відобразиться сторінка, яку ми можемо побачити, подивившись на рис. 3.14. Переходячи по кнопках відвідувач може знайти інформацію, яка може зацікавити саме його.



Компанія «Енерго Х» є постачальником електроенергії споживачам.

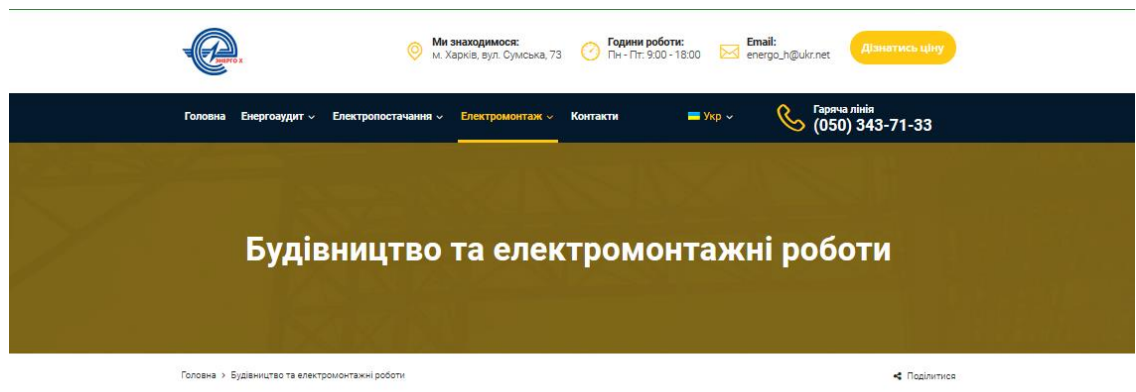
Переваги при роботі з «Енерго Х»

При роботі з ТОВ «Енерго Х» підприємство-споживач електроенергії може отримати ряд значних переваг:

1. При укладенні договору на поставку електроенергії тарифи обумовлюються в індивідуальному порядку. При цьому вони можуть бути нижче, ніж ціна за 1 кВт* год у Постачальника Універсальних Послуг (ПУП) або Постачальника, створеного на базі обленерго.

Рисунок 3.14 - Категорії користувачів сайту

Для користувача, що цікавиться виконанням монтажних робіт, відобразиться сторінка, представлена на рис. 3.15. Тут можна знайти інформацію стосовно послуг з будівництва та електроремонтних робіт – орієнтовні ціни, технологічні новації тощо.



У відповідності до ліцензії № 45-Л від 05 грудня 2016 р. (термін дії до 2021 р.) ми виконуємо наступні роботи:

- будівельно-монтажні роботи по зведенню і ремонту будівель і споруд;
- монтаж нових та реконструкція діючих електромереж 0,4; 6-35 кВ;
- облаштування нових, реконструкція діючих ПС, ТП, КТП 0,4; 6-110 кВ;
- монтаж нових і реконструкція діючих ВРП 6-330кВ;
- монтаж зовнішнього освітлення з використанням самонесучого ізоляованого проводу (СІП);
- монтаж та налагодка обладнання КВПА.

Можливе виконання робіт «під ключ» – від отримання технічних умов, до передачі на баланс.

Рисунок 3.15 - Категорії користувачів сайту

3.8 Функціональне тестування системи

Тести 1-2 описані для типу запису «Корисні статті». Аналогічні тести були проведені для решти типів записів, тести були пройдені.

Тест № 1. Публікація новини Модератором.

Вхідні дані: користувач заповнює поля:

- заголовок;
- анотація;
- основний текст та зображення;
- автор;
- натискає на кнопку «Опублікувати».

Очікуваний результат:

- вікно з повідомленням «Стаття опублікована! посилання»;
- на головній сторінці та в розділі «Корисні статті» з'явилась тільки що опублікована новина.

Отриманий результат:

- з'явилося вікно із повідомленням «Стаття опублікована! посилання»;
- на головній сторінці і в розділі «Корисні статті» з'явилась тільки що опублікована новина (рис. 3.13).

Тест пройдено.

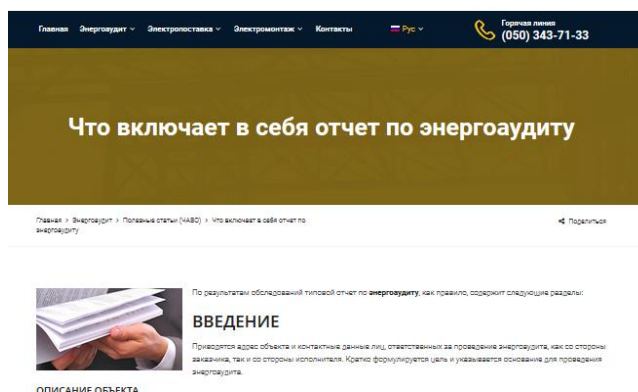


Рисунок 3.13 - Опублікована новина в розділі «Корисні статті»

Тест № 2. Публікація новини Модератором з пустими полями.

Вхідні дані: користувач заповнює поля:

- заголовок або анотація;
- основний текст або зображення;
- автор;
- натискає на кнопку «Опублікувати».

Очікуваний результат:

- вікно із повідомленням «Новина не опублікована! Заповніть поле «анотація»;
- вікно з повідомленням «Новина не опублікована! Заповніть поле «заголовок».

Отриманий результат:

- з'явилося вікно з повідомленням «Новина не опублікована! Заповніть поле «анотація».
- вікно з повідомленням «Новина не опублікована! Заповніть поле «заголовок».

Тест № 3. Створення адміністратором сторінки на двох мовах.

Вхідні дані: сторінки українською ще не існує, але російською вже є, далі користувач заповнює поля:

- заголовок;
- основний текст та зображення;
- автор;
- обирає мову «українська»;
- повторює процедуру заповнення, контент українською;
- натискає кнопку «Опублікувати».

Очікуваний результат:

- сторінка російською оновлюється;
- у відповідному місці українського меню з'являється пункт з текстом заголовка української версії сторінки, що редагується;

– після натискання на новий пункт українського меню відбувається перехід на українську версію сторінки.

Отриманий результат:

- сторінка російською оновилась;
- у відповідному місті українського меню з'явився пункт з текстом заголовка української версії сторінки, що редагується;
- після натискання на новий пункт українського меню відбувся перехід на українську версію сторінки.

Тест пройдено (рис. 3.14).



Рисунок 3.14 - Оpubлікована новина російською мовою в розділі «Корисні статті»

Тест № 4. Завантаження зображень в галерею через плагін «Галерея».

Вхідні дані: існує галерея, в ній чотири зображення, далі користувач здійснює такі дії:

- відкрити підпункт галереї «створити галерею/завантажити зображення»;
- обрати пункт випадаючого меню з найменуванням галереї, яка існує;
- натиснути на кнопку «Додати файли»;

– натиснути на кнопку «Завантажити файли».

Очікуваний результат:

– сторінка оновлюється;

– в галереї, що існує на сторінці «Наші роботи» з'являються завантажені зображення.

Отриманий результат:

– сторінка оновилась;

– в галереї, що існує на сторінці «Наші роботи» з'явилися завантажені зображення.

Тест пройдено (рис. 3.15).

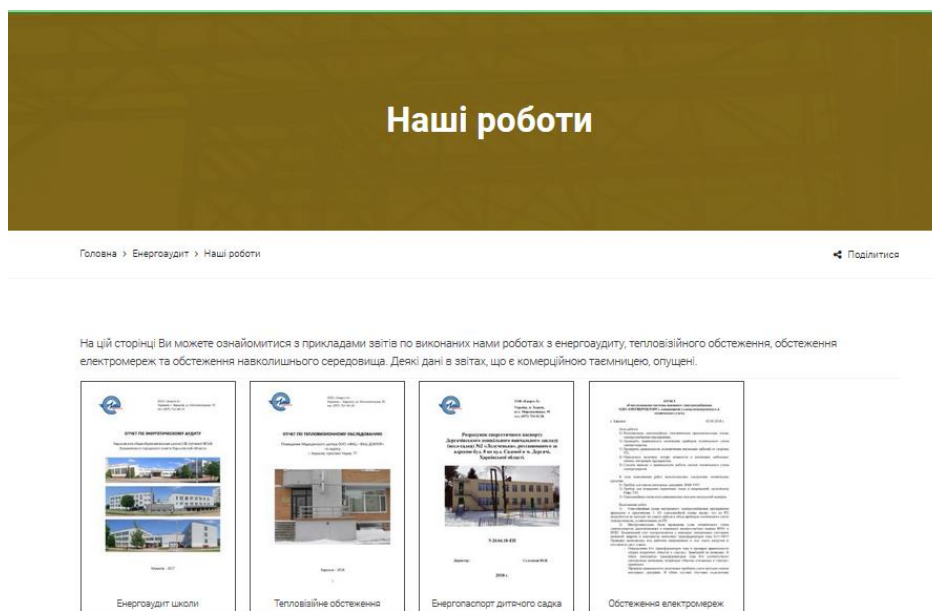


Рисунок 3.15 - Завантаження зображень на сайт

Тест № 5. Видалення та відновлення сторінок адміністратора.

Вхідні дані: існує сторінка, користувач здійснює дії:

– в розділі «Сторінки» CMS видаляє сторінку;

– після отримання повідомлення про видалення відновлює її з корзини.

Очікуваний результат:

– сторінка переміщується в корзину й зникає з меню веб-сайта;

– після відновлення з корзини сторінка з'являється в меню й у веб-

відображення сайту в попередньому вигляді.

Отриманий результат:

– сторінка перемістилася в корзину й зникла з меню у веб-відображенні сайту;

– після відновлення з корзини сторінка з'явилася в меню й у веб-відображенні сайту в попередньому вигляді.

3.9 Тестування адаптивності

Тест № 6. Адаптивність головної сторінки.

Вхідні дані: зміна розміру вікна з повноекранного до планшетного і мобільного.

Очікуваний результат:

– меню змінюється з розмірами вікна і відображується у відповідності з макетами;

– новини й проекти в повноекранному розмірі відображуються в три колонки, в планшетному в дві, в мобільному в одну;

– розмір каруселі та футера змінюються у відповідності до розміру екрану.

Отриманий результат: всі частини сторінки повністю задовольняють вимогам адаптивності.

Тест пройдено.

Тест № 7. Адаптивність статичних сторінок.

Вхідні дані: зміна розміру вікна з повноекранного до планшетного й мобільного.

Очікуваний результат:

– меню змінюється з розмірами вікна і відображується у відповідності до макетів;

– вміст сторінки і текст вистроюються в колонки у відповідності до розмірів вікна;

– всі елементи сторінки залишаються в межах вікна.

Отриманий результат: всі частини сторінки повністю задовольняють вимогам адаптивності.

Тест пройдено.

Тест № 8. Адаптивність на мобільних пристроях і планшетах.

Вхідні дані: завантаження сайту з мобільного телефону і планшета.

Очікуваний результат:

– меню відображується в згорнутому стані та розкривається завдяки натисканню кнопки, кожен пункт відображується в рядок;

– вміст сторінки відображується аналогічно відображенню при зміні вікна на десктопній версії (тести №4 і 5).

Отриманий результат:

– всі частини сторінки задовольняють вимогам адаптивності;

– під час тестування з мобільног пристрою була виявлена та виправлена помилка під час відображення меню: пункти відображувалися не в рядок, а вміщались в прямокутник фіксованого розміру.

Тест пройдено.

3.10 Тестування веб-відображення

Тест № 9. Перевірка відображення сторінок веб-сайту.

Вхідні дані: перегляд всіх сторінок сайту, перевірка всіх посилань і пунктів меню.

Очікуваний результат:

– сторінки завантажуються не довше двох секунд;

- всі сторінки завантажуються без помилок;
- структура сайту відповідає біжіній;
- перехід за посиланнями на зовнішні сайти проходять успішно;
- сторінки відображуються коректно, у відповідності до макетів.

Отриманий результат: всі сторінки задовольняють вимогам технічного завдання й відображуються коректно.

Тест пройдено.

Тест № 10. Тестування кроссбраузерності.

Вхідні дані:

– тестування кроссбраузерності сайту на ОС Windows в браузерах Opera, Mozilla Firefox, Chrome, Microsoft Edge и IE, перевірка коректного відображення сторінок, шрифтів і адаптивності.

– тестування кроссбраузерності сайту на macOS Sierra в браузерах Safari, и Chrome, перевірка коректного відображення сторінок, шрифтів і адаптивності;

– тестування кроссбраузерності сайту на мобільних пристроях і планшетах iOS іAndroid в браузерах Safari и Chrome відповідно, перевірка коректного відображення сторінок, шрифтів і адаптивності.

Очікуваний результат:

- всі сторінки завантажуються без помилок в усіх браузерах;
- підключення шрифтів відбувається у всіх браузерах;
- коректне та ідентичне відображення у всіх браузерах;
- адаптивність функціонує і всіх браузерах.

Отриманий результат: всі сторінки задовольняють вимогам технічного завдання і відображаються коректно у всіх протестованих браузерах.

Тест пройдено.

3.11 Юзабіліти тестування сайту

В процесі виконання етапів тестування – певних завдань користувачі промовляють те, що виконують. Цей процес можна назвати – «думки вголос». Респондентам видаються завдання пошуку інформації, власні дії в процесі виконання цих завдань вони озвучують і записують на диктофон. Після тестування проводиться аналіз записів – замірюється час виконання завдання, кількість помилок з подальшою аналітикою загального враження користувача. Для наочності процесів, що відбулися, результати аналізу зафіксовано в таблиці 3.

Обладнання для тестування: ноутбук, браузер Chrome, диктофон.

Учасники тестування: замовник будівельної послуги, менеджер фірми, модератор.

Завдання для учасників.

Замовник будівельної послуги:

– почати з головної сторінки сайту, знайти інформацію і фото про необхідну послугу;

– починаючи з головної сторінки сайту, знайти електронну адресу і контактний телефон;

– почати з головної сторінки сайту, знайти відгуки тих, кому надавалися послуги раніше.

Менеджер фірми:

– почати з головної сторінки сайту, знайти інформацію про замовлення послуг;

– почати з головної сторінки сайту, знайти інформацію з відгуками користувачів послуг;

– почати з головної сторінки сайту, знайти інформацію про нові заявки.

Модератор сайту:

– оновити пропозиції, перевірити їх відображення на сторінці сайту;

– опублікувати новину;

– опублікувати новину з пустими полями;

Дії Модератора під час тестування, як в тестах №1 і №2, але із застосуванням методики «думки вголос».

Таблиця 8. Результати юзабіліті-тестування.

	Частка задач, які виконані	Частка задач, які виконані правильно	Час на виконання завдання	Кількість помилок
Замовник	100%	90%	2,5хвилини	1
Менеджер	100%	85%	3 хвилини	2
Модератор	100%	35%	4 хвилини	5

З таблиці, в якій наведено результати тестування, бачимо, що 90% завдань Замовника вірні (лише одна помилка). Менеджер помилився двічі. Модератор допустив 5 помилок. Результати тестування відправлені Замовнику для узгодження.

На підставі проведених експериментів та отриманих результатів тестування можна зробити висновки про те, що мету роботи досягнуто, сайт має властивість бути налаштованим на предметну область у відповідності до вимог користувачів.

ВИСНОВКИ

В ході виконання атестаційної роботи, були проведені дослідження методів побудови веб-орієнтованих програмних систем для замовлення послуг з урахуванням галузі їх надання. Слід відмітити, що під час розробки подібних інтернет-проектів важливе одночасне врахування досить великої кількості факторів (багатофакторний аналіз). Але принципи структурування таких сайтів формуються, в першу чергу, з огляду на особливості цільової аудиторії. Все це потребує розробки теоретичної бази проекту, основаної на чималій дослідницькій роботі, чіткому й продуманому аналізі відповідної наукової літератури, вивченні практичної складової роботи з проектування програмних веб-систем, ознайомленні із сайтами-аналогами.

Таким чином, виконана робота мала за мету дослідження процесу розробки веб-системи замовлення послуг, що враховує конкретну область їх надання, а більш конкретно, методів побудови таких систем. Для досягнення поставленої мети були вирішені наступні задачі:

- досліджено види веб-орієнтованих систем;
- проведено порівняльний аналіз сайтів, що надають аналогічні послуги;
- спроектовано макети сторінок з подальшою розробкою дизайну;
- досліджені існуючі CMS та обрані оптимальні;
- розроблено веб-систему замовлення послуг;
- реалізовано веб-систему замовлення послуг, що враховує конкретну область їх надання;
- виконано тестування розробленої веб-системи.

Для отримання очікуваного результату була розроблена інформаційна архітектура веб-сайту і здійснений аналіз аналогічних інтернет-ресурсів. Особлива увага приділялась питанням вивчення цільової аудиторії. Для впорядкування і систематизації інформації про споживача була використана матриця Шеррінгтона з моделлю абстрактних цільових груп, завдяки якій була

розроблена модель сегментації ринку. Також була застосована методика вербалізації інформаційних портретів користувача, в якій діють конкретні персонажі. Застосування обох технологій одночасно забезпечило більш повне уявлення про цільового споживача. Також, була розглянута методика цифрового сторітеллінгу, яка є доцільною за наявності великих масивів даних і базується на засобах передачі інформації з винайденням смислів через розповідання історій.. Цифровий сторітеллінг був запропонований до використання в одному з розділів досліджуваної програмної системи.

В роботі реалізована модель веб-сайту, де наведений склад і взаємозв'язок його компонентів, а також розроблена модель адаптації з подальшим вибором засобів реалізації.

Розроблена веб-система була протестована. Застосовувалося функціональне тестування, тестування адаптивності, а також тестування веб-відображення, кроссбраузерності і юзабіліті-тестування.

Підводячи підсумки можна зазначити, що з урахуванням отриманих результатів мету роботи було досягнуто та методика орієнтації контексту веб-орієнтованої системи на предметну область можна успішно використати у подальших розробках.

ПЕРЕЛІК ПОСИЛАНЬ

1. Л. Шкляр, Р. Розен. Архитектура веб-приложений. - СПб .: Питер, 2011. – 640с. — ISBN 978-5-699-44273-7.
2. Розенфельд Л., Морвил П. Информационная архитектура в Интернете. – СПб.: Символ-Плюс, 2005. – 544 с.
3. Уильямс Б., Дэмстра Д., Стэрн Х. WordPress для профессионалов. Разработка и дизайн сайтов. Россия, – Питер, 2014. – 464 с.
4. Купер А., Рейман Р., Кронин Д. Алан Купер об интерфейсе. Основы проектирования взаимодействия. – СПб.: Символ-Плюс, 2009. – 688 с.
5. Уолтер А. Эмоциональный веб-дизайн. Россия, – Манн, Иванов и Фербер, 2000. – 160 с.
6. Денис Колисниченко. Движок для вашего сайта. CMS Joomla!, Slaed, PHP-Nuke. — Петербург: БХВ, 2008. — 352 с. — ISBN 978-5-9775-0258-0.
7. Flanagan, David, "JavaScript - The Definitive Guide", 5th ed., *O'Reilly, Sebastopol, CA, 2006*, p.497.
8. Мюллер-Брокман Й. Модульные системы в графическом дизайне. Россия, – Издательство Студии Артемия Лебедева, 2014. – 184 с.
9. Эйри Д. Логотип и фирменный стиль. Россия. – Питер, 2011. – 216 с.
10. Кажарнович В.Ф. SEO на результат. Простые и понятные методы продвижения в Интернете. – Питер, 2017. – 320 с.
11. Фримен Э. Изучаем HTML, XHTML и CSS. Россия. – Питер, 2012. – 565с.
12. Хеник Б. HTML и CSS. Путь к совершенству. Россия. – Питер, 2011. – 336 с.
13. Web-орієнтовані інформаційні системи. Методичні вказівки до вивчення дисципліни для студентів напряму підготовки 6.050103 «Програмна інженерія» / Титенко С. В. – К.: НТУУ «КПІ» 2015. – 51с. - Режим доступу - [www/ URL: http://www.setlab.net/downloads/articles/cms-tytenko.pdf](http://www.setlab.net/downloads/articles/cms-tytenko.pdf) - Загол. з екрану.

14. Gil Fink, Ido Flatow. Pro Single Page Application Development: Using Backbone.js and ASP.NET. — Apress, 2014. — ISBN 978-1-4302-6674-7.
15. Emmit Scott. Spa Design and Architecture: Understanding Single Page Web Applications. — Manning Publications Company, 2015. — ISBN 978-1-61729-243-9.
16. Руководство по PHP. [Электронный ресурс]: <https://www.bitrix.ua/> (дата звернення: 25.05.2019).
17. CMS 1С-Битрикс. [Электронный ресурс]: <https://www.bitrix.ua/> (дата звернення: 25.05.2019).
18. CMS Drupal. [Электронный ресурс]: <https://www.drupal.org/> (дата звернення: 25.05.2019).
19. CMS Joomla! [Электронный ресурс]: <https://joomla.ua/> (дата звернення: 25.05.2019).
20. CMS WordPress. [Электронный ресурс]: <https://wordpress.org/> (дата звернення: 25.05.2019).
21. JAVASCRIPT.RU [Электронный ресурс] - Режим доступа - [www/ URL: https://learn.javascript.ru/websockets](http://www.learn.javascript.ru/websockets) - Загол. з екрану; (дата звернення: 25.05.2019).
22. Документація CMS WordPress [Электронный ресурс] URL: <https://codex.wordpress.org/> (дата звернення: 25.05.2019).
23. Фреймворк для разработки адаптивных и мобильных web-проектов «Bootstrap 3» [Электронный ресурс] URL: [https:// bootstrap-3/](https://bootstrap-3/) (дата звернення: 25.05.2019).
24. Веб-сайт UkrMarket [Электронный ресурс] URL: <https://ukrmarket.net/> (дата звернення: 25.05.2019).
25. Веб-сайт Stroika.biz.ua [Электронный ресурс] URL: <http://stroika.biz.ua/> (дата звернення: 25.05.2019).
26. Веб-сайт ТОВ Енергоінвестпроект [Электронный ресурс] URL: <https://ua.kompass.com/c/енергоінвестпроект-тов/ua214202/> (дата звернення: 25.05.2019).