

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
СУМСЬКИЙ ДЕРЖАВНИЙ УНІВЕРСИТЕТ
Факультет електроніки та інформаційних технологій

Кафедра комп'ютерних наук

Кваліфікаційна робота магістра

**«ІНФОРМАЦІЙНА ТЕХНОЛОГІЯ КЕРУВАННЯ
ВЕБ-СИСТЕМОЮ СПОРТИВНО-ОЗДОРОВЧОГО ТАБОРУ»**

Здобувач освіти гр. ІК.мз-11с

Антон КАМИШНІКОВ

Науковий керівник
ст. викладач, к. ф.-м. н.

Галина ОЛЕКСІЄНКО

В. о. завідувача кафедри
доцент, к. т. н.

Ігор ШЕЛЕХОВ

Суми 2022

Сумський державний університет

(назва вузу)

Факультет ЦЗДВН Кафедра Комп'ютерних наук
Спеціальність 122 - Комп'ютерні науки

Затверджую:
зав. кафедри _____
“ _____ ” _____ 20__ р.

**ЗАВДАННЯ
НА ДИПЛОМНИЙ ПРОЕКТ (РОБОТУ) СТУДЕНТОВІ**

Камишнікову Антону Юрійовичу

(прізвище, ім'я, по батькові)

1. Тема проекту (роботи) Інформаційна технологія керування веб-системою
спортивно-оздоровчого табору

затверджую наказом по інституту від “ _____ ” _____ 20__ р. № _____

2. Термін здачі студентом закінченого проекту (роботи) _____

3. Вхідні дані до проекту (роботи) _____

4. Зміст розрахунково-пояснювальної записки (перелік питань, що їх належить розробити)

1) Веб-сайт як інформаційний ресурс та засіб комунікації. 2) Вибір програмних засобів реалізації. 3) Програмна реалізація інформаційної веб-системи Спортивно-оздоровчого табору «Орлятко».

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

6. Консультанти до проекту (роботи), із значенням розділів проекту, що стосується їх

Розділ	Консультант	Підпис, дата	
		Завдання видав	Завдання прийняв

7. Дата видачі завдання

Керівник

(підпис)

Завдання прийняв до виконання

(підпис)

КАЛЕНДАРНИЙ ПЛАН

№ п/п	Назва етапів дипломного проекту (роботи)	Термін виконання проекту (роботи)	Примітка
1	<i>Веб-сайт як інформаційний ресурс та засіб комунікації</i>		
2	<i>Вибір програмних засобів реалізації</i>		
3	<i>Програмна реалізація інформаційної веб-системи Спортивно-оздоровчого табору «Орлятко»</i>		
4	<i>Оформлення кваліфікаційної магістерської роботи</i>		

Студент – дипломник _____
(підпис)

Керівник проекту _____
(підпис)

ЗМІСТ

ВСТУП.....	6
1 ВЕБ-САЙТ ЯК ІНФОРМАЦІЙНИЙ РЕСУРС ТА ЗАСІБ КОМУНІКАЦІЇ... 8	8
1.1 Сутність та значення веб-сайту, його компоненти.....	8
1.2 Класифікація і принципи проектування веб-сайтів.....	10
1.3 Сайт як засіб комунікації з користувачами.....	14
1.4 Обґрунтування потреби у веб-сайті для оздоровчого табору.....	15
1.5 Постановка задач для створення нового веб-сайту для оздоровчого табору.....	16
2 ВИБІР ПРОГРАМНИХ ЗАСОБІВ РЕАЛІЗАЦІЇ.....	19
2.1 Технології для реалізації front-end частини.....	19
2.2 Технології для реалізації back-end частини.....	24
2.3 Додаткові програмні засоби для проектування веб-сайту.....	29
3 ПРОГРАМНА РЕАЛІЗАЦІЯ ІНФОРМАЦІЙНОЇ ВЕБ-СИСТЕМИ СПОРТИВНО-ОЗДОРОВЧОГО ТАБОРУ «ОРЛЯТКО»	32
3.1 Аналіз вимог до веб-сайту оздоровчого табору.....	32
3.2 Проектування макета та логічної структури веб-сайту.....	35
3.3 Створення дизайну та інформаційного наповнення сайту.....	38
3.4 Розробка модуля табірної спільноти.....	41
3.5 Тестування на кросбраузерність і коректність відображення.....	50
ВИСНОВКИ.....	54
СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ.....	55
ДОДАТКИ.....	61

ВСТУП

З кожним днем сучасне суспільство все більше інформатизується. Цей процес включає в себе сукупність політичних, правових, соціально-економічних, науково-технічних, організаційних, виробничих та інших процесів, що створюють відповідні умови для задоволення інформаційних потреб окремих громадян та суспільства в цілому [44]. Наразі саме широкомасштабне застосування інформаційних технологій у всіх сферах суспільного життя сприяє підвищенню ефективності поширення і використання інформації.

Сфери спорту, активного відпочинку та оздоровлення не є виключенням. На сьогодні існує безліч різних комп'ютерних застосунків, гаджетів, веб-сайтів тощо, які популяризують спорт, активний відпочинок та оздоровлення задля вирішення нагальних проблем, пов'язаних з різким зниженням рухової активності людини, і, як наслідок, загальним погіршенням самопочуття та здоров'я в суспільстві.

Проблема низької рухової активності стосується не лише дорослих. Сучасна українська дитина шкільного віку з одного боку перевантажена великою кількістю різноманітних навчальних предметів і напруженим графіком контролю знань, з іншого ж боку існує проблема «поринання» дітей у віртуальний світ у вільний від школи час, що також призводить до загального спаду рухової активності і зниження рівня здоров'я підростаючого покоління. Це обумовлює необхідність функціонування таких рекреаційних закладів, як, зокрема, оздоровчі табори.

Метою роботи є створення веб-ресурсу спортивно-оздоровчого табору, який буде привабливо виглядати зовні і вичерпно за змістом, буде виконувати представницькі функції, враховуючи різні категорії потенційних відвідувачів, і грати роль сполучної ланки з дітьми, їхніми батьками та всіма бажаними активно відпочити й оздоровитись.

Актуальність створення сайту полягає у впровадженні процесу інформатизації оздоровчого табору, що позитивно вплине на спортивно-оздоровчий рух, дозволить дітям та їх батькам своєчасно отримувати доступ до необхідної інформації, а також посприє налагодженню комунікації з ними.

Для реалізації мети поставлені наступні **задачі**:

- розгляд веб-сайту як інформаційного ресурсу та засобу комунікації;
- вибір та аналіз програмних засобів розробки веб-сайту;
- проектування макета та логічної структури веб-сайту;
- створення дизайну та інформаційного наповнення веб-сайту;
- розробка модуля табірної спільноти;
- тестування на кросбраузерність і коректність відображення.

Методологічною основою магістерської роботи є такі загальнонаукові методи: системний, структурний, порівняльний, модельний, що дає можливість проаналізувати специфіку створення веб-сайту як інформаційного ресурсу та визначити основні критерії для розробки та використання веб-сайту Спортивно-оздоровчого табору «Орлятко».

Отже, на підставі описаних вище задач, визначений **перелік питань**, які належить розробити:

- 1) Аналіз проблеми, мети. Постановка задач.
- 2) Пошук та аналіз існуючих засобів розробки сайтів.
- 3) Розробка структури та дизайну веб-сайту табору.
- 4) Програмна реалізація та тестування.

У **результаті** написання дипломної роботи було розглянуто інформаційні веб-технології для створення сайту табору, виконаний попередній аналіз необхідних засобів розробки та обрані засоби, які відповідають поставленим задачам; висвітлені та проілюстровані основні етапи, які були виконані під час розробки веб-сайту.

1. ВЕБ-САЙТ ЯК ІНФОРМАЦІЙНИЙ РЕСУРС ТА ЗАСІБ КОМУНІКАЦІЇ

1.1 Сутність та значення веб-сайту, його компоненти

Розвиток мережі Інтернет в останні десятиліття викликав кардинальні зміни в глобальному та національному інформаційному просторі. Сьогодні мережа Інтернет виступає універсальною платформою для надання величезної кількості різноманітних послуг (інформаційних, комунікаційних, фінансових, консалтингових тощо), без яких важко уявити життя сучасної людини. Серед найважливіших елементів цієї мережі – веб-сайти, на яких зосереджена величезна кількість інформаційних ресурсів. Створення веб-сайтів, як правило, є досить складним творчим процесом, що вимагає як технічних знань та навиків, так і креативних зусиль із боку їх творців.

Веб-сайт – це сукупність веб-сторінок та залежного вмісту, доступних у мережі Інтернет, які об'єднані як за змістом, так і за навігацією під єдиним доменним ім'ям або IP-адресою, власником яких є приватна особа або організація, установа. Одним з основних призначень веб-сайтів є представлення і позиціонування в інформаційному просторі організацій, що функціонують у сферах бізнесу, освіти, політики, культури, спорту тощо.

Веб-сторінка – інформаційний ресурс, доступний у мережі World Wide Web, який можна переглянути у веб-браузері. Зазвичай ця інформація записана у форматі HTML або XHTML і може містити гіпертекст із навігаційними гіперпосиланнями на інші веб-сторінки. Оскільки веб-сторінки складаються лише з текстової інформації, весь графічний, відео- і аудіоконтент зберігається на веб-сервері (або декількох серверах) у вигляді окремих файлів у відповідних форматах, а у коді сторінки зберігаються лише вказівки на них. На основі цих вказівок клієнтське програмне забезпечення (веб-браузер) оброблює цю інформацію і виводить її користувачу у звичному форматі.

Отже, веб-сайт є набором веб-сторінок, розміщених на веб-сервері (або декількох веб-серверах). Існують наступні компоненти для створення веб-сайту [26]:

– веб-хост (хостинг) – це місце, де фізично розташований веб-сайт. Група веб-сторінок (пов'язаних веб-сторінок) можуть називатися веб-сайтом лише якщо веб-сторінка розміщена на веб-сервері. Веб-сервером є набір файлів, які передаються на комп'ютери користувачів, коли вони вказують адресу веб-сайту.

– адреса: адреса веб-сайту також відома, як URL-адреса веб-сайту. Коли користувач хоче відкрити веб-сайт, йому потрібно ввести адресу або URL-адресу веб-сайту у веб-браузер, і веб-сервер доставить запитуваний веб-сайт.

– домашня сторінка. Це перша веб-сторінка, яка з'являється, коли користувач відвідує веб-сайт. Домашня сторінка веб-сайту дуже важлива, оскільки вона визначає зовнішній вигляд веб-сайту та спрямовує глядачів на решту сторінок веб-сайту.

– дизайн – це остаточний і загальний вигляд веб-сайту, який є результатом належного використання та інтеграції таких елементів, як навігаційні меню, графіка, макет, навігаційні меню тощо.

– вміст. Усі веб-сторінки, розміщені на веб-сайті, разом складають вміст веб-сайту. Хороший вміст на веб-сторінках робить сайт більш ефективним і привабливим.

– навігаційна структура. Навігаційна структура веб-сайту – це порядок сторінок, набір посилань. Зазвичай її об'єднує принаймні одне навігаційне меню.

Значення веб-сайту в сучасному інформаційному просторі важко переоцінити, оскільки саме веб-сайт є тією сполучною ланкою між всіма учасниками віртуального простору, яка забезпечує належний комунікаційний процес.

Функціональність і цілісність того чи іншого веб-сайту обумовлюється наявністю необхідних структурних компонентів, за допомогою яких створюється або з яких складається будь-який сайт. Всі структурні компоненти повинні враховуватись під час розробки сайту, незалежно від того, яким є вид даного сайту і яку функцію йому належить виконувати.

1.2 Класифікація і принципи проектування веб-сайтів

Кількість сайтів у всесвітній мережі безперервно зростає. Свої власні сайти розробляють комерційні та некомерційні організації, навчальні заклади та навіть звичайним користувачам Інтернету не складає труднощів створення своєї особистої сторінки або навіть повноцінного Інтернет-ресурсу.

Для зручності розробки й визначення основних функцій та особливостей веб-сайтів було написано багато робіт з їх класифікації [3; 27; 54; 55; 59; 64]. Однак нині немає загальноприйнятої системи класифікації. Так, наприклад, М.С. Лебеденко та І.В. Лученко [19] виокремлюють наступні види сайтів:

1. Інформаційні – функціонують для надання певної інформації з метою збільшення рівня обізнаності про певні товари чи послуги, та формування позитивного ставлення до компанії.
2. Комерційні, основна функція яких – розширення ринку збуту та здійснення процесів купівлі-продажу.
3. Навігаційні – слугують для посилення позицій фірми на ринку, зацікавлюють та утримують клієнта на своєму веб-ресурсі.

Розглянемо основні типології сайтів, що виділяються за певними ознаками.

1. Типи сайтів за метою створення:
 - комерційні. Вони є «обличчям» компанії, здійснюючи взаємодію з широкою аудиторією потенційних клієнтів (покупців, замовників тощо). Головне призначення комерційних сайтів полягає в

отриманні максимальної фінансової вигоди, яка може виражатися у прямому збільшенні доходів компанії, або у зростанні інших комерційних показників;

- некомерційні – це сайти, які не мають одержання прибутку основною метою своєї діяльності.

2. Типи сайтів за доступністю:

- а) відкриті – доступні для будь-яких відвідувачів;
- б) напіввідкриті – для доступу необхідна реєстрація;
- с) закриті – доступні для вузького кола людей.

3. Типи сайтів за функціональністю та стилем оформлення:

- 1) статичні – сайт будується із статичних html-сторінок. Ці сторінки пов'язані між собою за допомогою гіперпосилань. На сервері зберігаються готові зверстані сторінки. Користувач отримує веб-сторінки в незмінному вигляді.
- 2) динамічні – основна особливість полягає в тому, що різні частини сторінки можуть зберігатися в різних файлах та базах даних. При видачі користувачеві запитувана веб-сторінка генерується «на льоту».

4. Типи сайтів з фізичного розташування:

- загальнодоступні – доступні для користувачів Інтернету;
- локальні – доступні лише в межах локальної мережі. Це можуть бути як корпоративні сайти організацій, так і сайти приватних осіб локальної мережі провайдера.

5. Типи сайтів за особливостями представлення інформації та категоріями розв'язуваних завдань:

- інтернет-представництва організацій (сайти, що містять інформацію про компанії та їхні послуги, здійснюють функцію електронної торгівлі);

- інформаційні ресурси (призначені для інформування відвідувачів, поширення новин, тематичні сайти, енциклопедії, словники тощо);
- веб-сервіси (сайти, на яких надаються послуги для виконання певних завдань, зокрема, сайти пошукових служб, поштові сайти, сайти служб онлайнного документообігу тощо);
- соціальні мережі (інтерактивні багатокористувацькі веб-сайти, які наповнюються самими учасниками мережі. Сайт являє собою автоматизоване соціальне середовище, що дозволяє спілкуватися групі користувачів, об'єднаних спільним інтересом).

6. За особливостями макету, що застосовується:

- 1) з фіксованою шириною – ширина елементів сторінки не масштабується при зміні ширини вікна. Якщо використовується дизайн з фіксованою шириною, то завдання ширини сторінок сайту провадиться в абсолютних величинах (у пікселях).
- 2) зі змінною шириною (гумовий дизайн). У разі «дизайну з шириною сторінки адаптується під ширину вікна. Для цього значення ширини сторінки та її елементів, наприклад, таблиць, задаються в відсотках.
- 3) з еластичним дизайном – розміри елементів та шрифтів задаються відносними значеннями. У разі застосування еластичного дизайну при зміні ширини вікна змінюються розміри елементів і шрифтів.
- 4) з гібридним дизайном. Гібридний дизайн характеризується тим, що ширина частини елементів задається відносними значеннями, а частина визначається в відсотках. Використовується при верстці у декілька колонок. При цьому ширина сторінки адаптується під ширину вікна.

Ефективність веб-сайту, тобто його практичність, зручність та візуальна привабливість, залежить від слідування низці принципів проектування, до яких відносяться [16]:

- логічне структурування та композиційне впорядкування всіх елементів на веб-сторінках відповідно їх значущості;
- проектування інтерфейсу веб-сайту таким чином, щоб користувач міг зрозуміти, де саме він знаходиться і як потрапити туди, куди йому потрібно в межах даного сайту;
- створення візуального образу сайту з врахуванням такої композиції, яка відповідає змісту, характеру і призначенню сайту;
- наявність загального стилю або теми на всіх сторінках веб-сайту для того, щоб він сприймався цілісно і гармонійно.

На думку Л. Москалець [24], під час розробки сайтів варто опиратися на ряд принципів, зокрема:

1. Активний розвиток мобільного інтернету, і, відповідно, зростання числа користувачів, які заходять у мережу із мобільних пристроїв (саме тому сайт має бути придатним для перегляду на маленькому екрані).
2. Простота та зручність користування сайтом.
3. Мінімальне число дій для досягнення поставленої мети.
4. Збільшення швидкості роботи та завантаження сайту.

На сьогодні не існує загальноприйнятої класифікації сайтів, і в залежності від тієї чи іншої ознаки, дослідники виділяють різні типи сайтів. Тим не менш, різні класифікації веб-сайтів мають багато спільного між собою, здебільшого виокремлюючи ідентичні аспекти, згідно з якими відбувається поділ на типи.

Щодо принципів проектування сайтів, то останнім часом у веб-програмуванні спостерігається тенденція до максимального спрощення дизайну та структури сайтів у напрямку більшої їх зрозумілості та лаконічності, а також надання переваги таким характеристикам, як практичність, зручність та візуальна привабливість.

1.3. Сайт як засіб комунікації з користувачами

Інтернет – не лише новітній засіб комунікації між соціальними суб'єктами, але й такий, що, по суті, здатний формувати нову віртуальну за своєю природою реальність: в умовах небаченої насиченості інформаційних потоків «віртуальний простір» за своїм впливом на масову свідомість може ставати рівнозначним по відношенню до об'єктивної реальності [12].

Елементами, що структурують та впорядковують такий віртуальний простір та водночас виступають комунікаційними каналами, насичуючи його відповідним змістом, є веб-сайти, які представляють собою набір інформаційних блоків та інструментів для спілкування з аудиторією (відвідувачами). Комунікація в Інтернеті містить потужні елементи зворотного зв'язку й тим самим створює безпрецедентні можливості для спілкування.

Основним призначенням веб-сайтів як засобів зв'язку з громадськістю має бути організація цільового інформаційного впливу на соціальне оточення та цільові групи, у співпраці та у залученні яких зацікавлена організація. Переваги веб-сайту як інструмента інформаційної підтримки та просування суб'єкту оздоровчої діяльності очевидні: це інформаційна мобільність, загальнодоступність, а також мультимедійність. Причому вплив інформаційного змісту матеріалів, розташованих на сайті, значно підсилюється формами його представлення, дизайном.

Веб-сайт організації відкриває нові можливості для створення взаємовідносин з користувачами. Ю.В. Шурчкова [45] вважає, що існує низка переваг при використанні інтернет-технологій для комунікації з користувачами:

- підвищення продуктивності компанії;
- поява додаткового нового способу продажів;
- прямий вихід на ринок;
- покращення обслуговування клієнтів;

- удосконалення бренду та бізнесу;
- вдосконалення взаємовідносин з клієнтами;
- вихід на додаткові ринкові ніші та миттєвий спосіб збирання інформації.

Отже, Інтернет-комунікація сьогодні є одним із перспективних шляхів представлення будь-якої організації. Саме тому необхідно пам'ятати, що веб-сайту слід бути зручним та зрозумілим. Це істотно полегшить роботу користувачів із наявними інформаційними ресурсами. Така інформаційна політика стає умовою формування відповідного позитивного іміджу.

1.4. Обґрунтування потреби у веб-сайті для оздоровчого табору

У вітчизняній та зарубіжній літературі представлені різні думки стосовно місця та ролі сайту в рекламуванні компанії. Так, на думку В. Холмогорова, «кожне сучасне підприємство, котре прагне досягти максимальної ефективності в отриманні фінансового прибутку, повинне мати власний веб-сайт» [24], який є ефективним інструментом не тільки для збільшення продаж, а й для зворотного зв'язку з користувачами. І. Манн вважає, що компанія без сайту подібна до ділової людини, яка не має своєї візитівки [23].

А. Уїллер наголошує, що веб-сайт відображає індивідуальність компанії, створює ефект присутності та безпосередньої взаємодії з компанією. «Відвідування веб-сайту компанії дає такий самий ефект, що і реальний візит до неї, а іноді буває й ефективнішим, простішим та швидшим, оскільки дає можливість клієнту прийняти рішення про покупку у спокійній обстановці, без тиску. Сайт дозволяє ефективно виразити індивідуальність бренду, а образи, логотипи та повідомлення – усе те, що завантажуються клієнтами, може використовуватися співробітниками компанії як засоби маркетингу у будь-якій точці світу» [35].

Підсумовуючи все вищезазначене, відзначимо, що веб-сайт оздоровчого табору є важливим інструментом формування та підтримки іміджу, оскільки сприяє підвищенню ступеня упізнавання бренду. Сайт повинен надавати вичерпний обсяг інформації про заклад, сферу діяльності, послуги, які пропонуються. Також він є основою для організації зворотного зв'язку зі споживачами.

1.5. Постановка задач для створення нового веб-сайту для оздоровчого табору

Для реалізації та виконання поставлених задач, адміністрації Спортивно-оздоровчого табору «Орлятко» було запропоновано описати всі необхідні характеристики та параметри (Таблиця 1.1), що у свою чергу полегшило розуміння сторін щодо визначення точного переліку робіт, вигляду структури, елементів та дизайну сайту.

Таблиця 1.1. Технічне завдання щодо створення сайту оздоровчого табору

1. Загальний опис		
1.1.	Основна назва сайту (заголовок сайту)	Спортивно-оздоровчий табір «Орлятко»
1.2.	Короткий опис сайту	Офіційний сайт СОТ «Орлятко» з інформацією про табір та його розташування, інфраструктуру, про умови проживання, харчування, розважальну програму, розклад змін і вартість путівок, різні послуги, контактні дані.
1.3.	Адреса сайту	https://orliatko.com.ua
1.4.	Характер сайту	Інформаційно-розважальний

1.5.	Задачі сайту	<ul style="list-style-type: none"> – інформування; – реклама послуг; – висвітлення подій у таборі; – розважальна задача (перегляд фото та відео, прослуховування табірної музики); – зворотній зв'язок;
1.6.	Основні розділи сайту	<ul style="list-style-type: none"> – «Ласкаво просимо» (розташування та інфраструктура табору) ; – «Дитячий відпочинок» (умови проживання, харчування, розважальна програма, розклад змін і вартість путівок); – «Інші послуги» (оренда, спортивно-масові заходи і тренувальні збори); – «Наші контакти» (адреса табору, телефони, email); – Спільнота «Орлятко» (Реєстрація, вхід до спільноти, табірні фото, табірні відео, табірна музика).
1.7.	Очікувана частота оновлення сайту	Щотижня
1.8.	Мова сайту	Українська, англійська.
1.9.	Ключові слова	Спортивно-оздоровчий табір, «Орлятко» Суми, дитячий відпочинок влітку, дитячий літній табір, спортивно-масові заходи та навчально-тренувальні збори, контакти табору «Орлятко».
2. Дизайн сайту		
2.1.	Наявність макета	Так.

2.2.	Наявність фірмового стилю	Ні.
2.3.	Наявність логотипу	Так.
2.4.	Опис очікуваного стилю	Простий, яскравий, життєствердний.
2.5.	Обов'язкова підтримка браузерів	IE 11, Mozilla Firefox, Google Chrome, Safari.
2.6.	Необхідність адаптивної (мобільної) версії сайту для всіх вищевказаних браузерів	Так.

2. ВИБІР ПРОГРАМНИХ ЗАСОБІВ РЕАЛІЗАЦІЇ

2.1. Технології для реалізації front-end частини

HTML

Мова розмітки гіпертексту (Hypertext Markup Language), або як її частіше називають HTML, – це комп'ютерна мова, що лежить в основі World Wide Web (Всесвітньої павутини). Завдяки мові HTML будь-який текст можна розмітити, перетворивши його на гіпертекст з подальшою публікацією на Web [64].

Мова HTML має власний набір символів, за допомогою яких веб-браузери відображають сторінку. Ці символи називаються дескрипторами і включають елементи, необхідні для створення гіперпосилань.

Однією з відмінних рис HTML-документів є те, що сам документ містить лише текст, а всі інші об'єкти вбудовуються у документ в момент його відображення браузером за допомогою спеціальних тегів та зберігаються окремо. При збереженні HTML-файлу у місці розташування документа створюється папка, в яку поміщаються супутні йому графічні елементи оформлення [8].

Мова HTML5 (остання версія HTML) стала суттєвим кроком вперед у веб-дизайні, розмітці та зручності використання [25]. Вона є простим засобом роботи з графікою у браузері без використання додаткових модулів, наприклад Flash, а також пропонує методи вставки аудіо і відео у веб-сторінки (також без додаткових модулів) та згладжує деякі невідповідності, що виникли в HTML по мірі її розвитку.

Крім того, HTML5 має безліч інших удосконалень, наприклад оброблення місцезнаходження користувача, робочі веб-процеси (web workers), що керують виконанням фонових задач, покращену обробку форм

та доступ до пакетів локального сховища (які значно перевищують обмежені можливості cookie).

CSS

CSS використовується авторами та відвідувачами веб-сторінок, щоб визначити кольори, шрифти, верстку та інші аспекти вигляду сторінки. Одна з головних переваг – можливість розділити зміст сторінки (або контент, наповнення, зазвичай HTML, XML або подібна мова розмітки) від вигляду документу (що описується в CSS).

Таке розділення може покращити сприйняття та доступність контенту, забезпечити більшу гнучкість та контроль за відображенням контенту в різних умовах, зробити контент більш структурованим та простим, прибрати повтори тощо. CSS також дозволяє адаптувати контент до різних умов відображення (на екрані монітора, мобільного пристрою (КПК), у роздрукованому вигляді, на екрані телевізора, пристроях з підтримкою шрифту Брайля або голосових браузерів тощо).

Один і той самий HTML або XML документ може бути відображений по-різному залежно від використаного CSS. Стили для відображення сторінки можуть бути [50]:

1) Стили автора (інформація надана автором сторінки):

- зовнішні таблиці стилів (англ. stylesheet), найчастіше окремий файл або файли .css;
- внутрішні таблиці стилів, включені як частина документу або блоку;
- стилі для окремого елемента.

2) Стили користувача: локальний .css-файл, вказаний користувачем для використання на сторінках і вказаний в налаштуваннях браузера.

3) Стили переглядача (браузера): стандартний стиль переглядача, наприклад стандартні стилі для елементів, визначені браузером,

використовуються коли немає інформації про стиль елемента або вона неповна.

Стандарт CSS визначає порядок та діапазон застосування стилів, тобто, в якій послідовності і для яких елементів застосовуються стилі. Таким чином, використовується принцип каскадності, коли для елементів вказується лише та інформація про стилі, що змінилася або не визначена загальнішими стилями.

До переваг CSS можна віднести наступне:

а) інформація про стиль для усього сайту або його частин може міститися в одному .css-файлі, що дозволяє швидко робити зміни в дизайні та презентації сторінок;

б) різна інформація про стилі для різних типів користувачів: наприклад, великий розмір шрифту для користувачів з послабленим зором, стилі для виводу сторінки на принтер, стиль для мобільних пристроїв;

в) сторінки зменшуються в об'ємі та стають більш структурованими, оскільки інформація про стилі відділена від тексту та має певні правила застосування, і сторінка побудована з їх урахуванням;

г) прискорення завантаження сторінок і зменшення обсягів інформації, що передається, навантаження на сервер та канал передачі. Досягається за рахунок того, що сучасні браузері здатні кешувати (запам'ятовувати) інформацію про стилі і використовувати для всіх сторінок, а не завантажувати для кожної.

Javascript

JavaScript – це динамічна мова комп'ютерного програмування. Вона легка і найчастіше використовується як частина веб-сторінок, реалізації яких дозволяють сценарію на стороні клієнта взаємодіяти з користувачем і створювати динамічні сторінки. Це інтерпретована мова програмування з об'єктно-орієнтованими можливостями [53].

Специфікація ECMA-262 визначила стандартну версію основної мови JavaScript:

- JavaScript – це легка інтерпретована мова програмування.
- Призначена для створення мережевих додатків.
- Доповнює та інтегрується з Java.
- Доповнює та інтегрується з HTML.
- Відкрита та кросплатформна.

JavaScript на стороні клієнта є найпоширенішою формою мови. Сценарій має бути включений до HTML-документа або посилання на нього для того, щоб код інтерпретувався браузером.

Це означає, що веб-сторінка не обов'язково має бути статичним HTML, але може містити програми, які взаємодіють з користувачем, керують браузером і динамічно створюють вміст HTML.

Клієнтський механізм JavaScript надає багато переваг перед традиційними серверними скриптами CGI. Наприклад, ви можете використовувати JavaScript, щоб перевірити, чи користувач ввів дійсну адресу електронної пошти в поле форми. Код JavaScript виконується, коли користувач надсилає форму, і лише якщо всі записи дійсні, вони надсилаються на веб-сервер.

JavaScript можна використовувати для перехоплення ініційованих користувачем подій, таких як натискання кнопок, навігація за посиланням та інші дії, які користувач ініціює явно чи неявно.

Перевагами використання JavaScript є:

1) Менше взаємодії з сервером – можна перевірити введені користувачем дані перед тим, як відправити сторінку на сервер. Це економить трафік сервера, що означає менше навантаження на ваш сервер.

2) Миттєвий зворотний зв'язок з відвідувачами – їм не потрібно чекати перезавантаження сторінки, щоб побачити, чи не забули вони щось ввести.

3) Підвищена інтерактивність – можна створювати інтерфейси, які реагуватимуть, коли користувач наведе на них мишу або активує їх за допомогою клавіатури.

4) Розширені інтерфейси – можна використовувати JavaScript, щоб включити такі елементи, як drag-and-drop компоненти та слайдери, щоб надати відвідувачам вашого сайту розширений інтерфейс.

Отже, JavaScript – це легка, інтерпретована мова програмування, яка дозволяє вбудовувати інтерактивність у статичні сторінки HTML.

jQuery

При всій гнучкості та ефективності Javascript, а також при всьому багатстві вбудованих функцій, які є у цій мові, все ж зберігається потреба у додаткових рівнях коду, що дозволяють спростити, наприклад, отримання ефектів анімації, обробку подій і застосування технології асинхронного обміну даними, тобто зробити те, чого неможна досягнути застосуванням звичайних засобів JavaScript і CSS.

Крім того, внаслідок різноманітних браузерних "війн", історично то з'являлись, то зникали несумісності браузерів, які часом виникали на різних платформах і в різних програмах [25]. Для заповнення прогалів була розроблена велика кількість бібліотек-функцій, призначених для зведення до мінімуму відмінностей між браузерами и для сприяння асинхронному обміну даними, а також роботі з подіями та анімацією. В їх перелік входять такі бібліотеки, як AngularJS, jQuery, MooTools, Prototype, YUI, ReactJS, Bootstrap тощо.

Сьогодні jQuery встановлена більш ніж на 77% сайтів і затребувана ширше, ніж всі її конкуренти разом узяті [56; 63]. Використовуючи jQuery, можна отримати не тільки крос-браузерну сумісність доволі високого рівня, а й швидкий і легкий доступ до операцій HTML і DOM, можливість використання функцій для безпосередньої роботи з CSS, керування подіями,

потужні засоби для створення професійних ефектів і анімації, а також функції для керування асинхронним обміном даних з сервером. Крім того, jQuery є основою для широкого кола додаткових модулів й інших допоміжних програм.

2.2. Технології для реалізації back-end частини

PHP

PHP – це серверна мова програмування з відкритим вихідним кодом, яку можна використовувати для створення веб-сайтів, програм, систем управління взаємовідносинами з клієнтами тощо. Це широко використовувана мова загального призначення, яку можна вставляти в HTML. Програмування на PHP можна використовувати для створення більшості речей, які потрібні розробнику програмного забезпечення. Однак є три основні сфери, в яких PHP процвітає [65]:

1. Сценарій на стороні сервера є головною перевагою PHP. Якщо людина тільки вчиться кодувати й хоче вивчити сценарії на стороні сервера, PHP – чудова мова для вивчення.

2. Сценарії командного рядка ідеально підходять для сценаріїв, створених за допомогою cron (Linux) або Task Scheduler (Windows). Мова також чудово підходить для простої обробки тексту.

3. PHP, мабуть, не найкраща мова для створення настільних програм, але для просунутого веб-розробника PHP надає набагато більше можливостей, ніж її конкуренти.

Звичайно, PHP може робити багато інших речей. Наприклад, вона чудово збирає дані форм, шифрує дані користувача та надсилає та отримує файли cookie. Одна з головних особливостей PHP полягає в тому, що вона сумісна з усіма основними операційними системами, тому можна кодувати незалежно від того, яка технологія використовується.

PHP – мова, у код якої можна вбудовувати безпосередньо html-код сторінок, які, у свою чергу, коректно оброблюватимуться PHP-інтерпретатором. Обробник PHP просто починає виконувати код після відкриваючого тегу (<?php) і продовжує виконання до того моменту, поки не зустрине закриваючий тег.

Велика різноманітність функцій PHP дає можливість уникати написання багаторядкових функцій, призначених для користувача, як це відбувається в C або Pascal.

У PHP вбудовані бібліотеки для роботи з MySQL, PostgreSQL, SQLite, mSQL, Oracle, dbm, Hyperware, Informix, InterBase, Sybase. Завдяки стандарту відкритого інтерфейсу зв'язку з базами даних (англ. Open Database Connectivity Standard, ODBC) можна підключатися до всіх баз даних, до яких є драйвер.

Багато конструкцій мови запозичені з C, Perl. Код PHP дуже схожий на той, який зустрічається в типових програмах мовами C або Pascal. Це помітно знижує початкові зусилля при вивченні PHP. PHP – мова, що поєднує переваги Perl та C і спеціально спрямована на роботу в Інтернеті, мова з універсальним і зрозумілим синтаксисом. І хоча PHP є досить молодого мовою, вона здобула таку популярність серед web-програмістів, що в наш час є найпопулярнішою мовою для створення веб-застосунків (скриптів).

Ефективність є дуже важливим чинником у програмуванні для середовищ розрахованих на багато користувачів, до яких належить і web. Важливою перевагою PHP є те, що ця мова належить до інтерпретованих. Це дозволяє обробляти сценарії з достатньо високою швидкістю. За деякими оцінками, більшість PHP-сценаріїв (особливо не дуже великих розмірів) обробляються швидше за аналогічні їм програми, написані на Perl. Проте хоч би що робили розробники PHP, виконавчі файли, отримані за допомогою компіляції, працюватимуть значно швидше – в десятки, а іноді і в сотні разів.

Але продуктивність PHP достатня для створення цілком серйозних веб-застосунків.

MySQL

MySQL наразі є найпопулярнішим програмним забезпеченням системи керування базами даних, яке використовується для керування реляційною базою даних. Це програмне забезпечення для баз даних з відкритим кодом, яке підтримується компанією Oracle. Це швидка, масштабована та проста у використанні система керування базами даних у порівнянні з Microsoft SQL Server та Oracle Database. Вона зазвичай використовується в поєднанні зі скриптами PHP для створення потужних і динамічних корпоративних додатків на стороні сервера або в Інтернеті.

Особливостями MySQL є наступні [58]:

- дозволяє реалізовувати операції бази даних над таблицями, рядками, стовпцями та індексами.
- визначає відношення бази даних у формі таблиць (набір рядків і стовпців).
- забезпечує посиальну цілісність між рядками або стовпцями різних таблиць.
- дозволяє автоматично оновлювати індекси таблиці.
- використовує багато запитів SQL і поєднує корисну інформацію з кількох таблиць для кінцевих користувачів.

Ядром бази даних MySQL є MySQL Server. Цей сервер доступний як окрема програма та відповідає за обробку всіх інструкцій, операторів або команд бази даних.

Робота бази даних MySQL із сервером MySQL наступна:

- 1) MySQL створює базу даних, яка дозволяє формувати багато таблиць для зберігання та обробки даних і визначення зв'язку між кожною таблицею.

2) Клієнти роблять запити через екран GUI або командний рядок, використовуючи певні вирази SQL у MySQL.

3) Нарешті, серверна програма відповідає на вирази, що запитуються, та виводить результат на стороні клієнта.

MySQL є популярним через такі причини:

- MySQL – це база даних з відкритим вихідним кодом.
- MySQL – це дуже потужна програма, яка може працювати з великим набором функціональних можливостей найдорожчих і найпотужніших пакетів баз даних.
- MySQL можна налаштувати, оскільки це база даних з відкритим вихідним кодом, а ліцензія GPL з відкритим вихідним кодом полегшує програмістам модифікацію програмного забезпечення SQL відповідно до їх власного середовища.
- MySQL працює швидше, ніж інші бази даних, тому може добре працювати навіть із великим набором даних.
- MySQL підтримує багато операційних систем з багатьма мовами, такими як PHP, PERL, C, C++, JAVA тощо.
- MySQL використовує стандартну форму відомої мови даних SQL.
- MySQL добре поєднується з PHP, найпопулярнішою мовою веб-розробки.
- MySQL підтримує великі бази даних, до 50 мільйонів рядків або більше в таблиці. Обмеження розміру файлу для таблиці за замовчуванням становить 4 ГБ, але можна збільшити його (якщо операційна система може з цим впоратися) до теоретичного обмеження в 8 мільйонів терабайт (ТБ).

Веб-сервер Apache

Apache є кросплатформним програмним забезпеченням, підтримуючим такі ОС як Linux, BSD, Mac OS, Microsoft Windows. Основними перевагами

Apache вважаються надійність та гнучкість конфігурації. Він дозволяє підключати зовнішні модулі для надання даних, використовувати СУБД для автентифікації користувачів, модифікувати повідомлення про помилки.

Ядро Apache включає основні функціональні можливості, такі як обробка конфігураційних файлів, протокол HTTP та система завантаження модулів. Ядро (на відміну від модулів) повністю розробляється Apache Software Foundation без участі сторонніх програмістів [36].

Теоретично ядро Apache може функціонувати у чистому вигляді, без використання модулів. Однак функціональність такого рішення вкрай обмежена. Ядро Apache повністю написано мовою програмування C.

Система конфігурації Apache заснована на текстових конфігураційних файлах. Вона має три умовні рівні конфігурації:

1. Конфігурація сервера (`httpd.conf`).
2. Конфігурація віртуального хоста (`httpd.conf` з версії 2.2 `extra/httpd-vhosts.conf`).
3. Зміна рівня директорії (`.htaccess`).

Практично всі параметри ядра можуть бути змінені через конфігураційні файли, включно з управлінням MPM (Multi-Processing Modules). Більшість модулів має власні параметри.

Apache має вбудований механізм віртуальних хостів. Він дозволяє повноцінно обслуговувати на одній IP-адресі безліч сайтів (доменних імен), відображаючи для кожного з них власний вміст.

Існує безліч модулів, що додають до Apache підтримку різних мов програмування та систем розробки. До них відносяться:

1. PHP (`mod_php`).
2. Python (`mod_python`).
3. Ruby (`apache-ruby`).
4. Perl (`mod_perl`).
5. ASP (`apache-asp`).

Крім того, Apache підтримує механізми CGI та FastCGI, що дозволяє виконувати програми практично на всіх мовах програмування, зокрема C, C++, sh, Java.

2.3. Додаткові програмні засоби для розробки веб-сайту

Для розробки веб-сайту ми обрали такі програми, як Sublime Text 3, Ampps, phpMyAdmin, cPanel. Розглянемо їх трохи детальніше.

1) Sublime Text 3 – одна з версій швидкого кросплатформного текстового редактора, що підтримує плагіни для кодування на PHP: CTags, PHP Companion, PHP Completions Kit. Sublime Text не є вільним чи відкритим програмним забезпеченням, але деякі його плагіни (в тому числі і вищезгадані) розповсюджуються з вільною ліцензією, розробляються і підтримуються спільнотою розробників.

Редактор містить різні візуальні теми, з можливістю завантаження додаткових. Користувачі бачать весь свій код в правій частині екрану у вигляді міні-карти, при кліці на яку можна здійснювати навігацію [62].

Є кілька режимів екрану. Один з них включає від 1 до 4 панелей, за допомогою яких можна показувати до чотирьох файлів одночасно. Повноцінний (free modes) режим показує тільки один файл без будь-яких додаткових навколо нього меню.

Виділення стовпців цілком або розстановка декількох покажчиків по тексту робить можливим миттєву правку. Покажчики поведуться, ніби кожен з них єдиний в тексті. Команди типу: переміщення на знак, переміщення на рядок, вибірка тексту, переміщення на слово або його частини (CamelCase, розділений дефісом або підкресленням), перехід на початок або кінець рядка тощо, впливають на всі покажчики незалежно і відразу, дозволяючи правити складноструктурований текст швидко, без використання макрокоманд або регулярних виразів.

Коли користувач набирає код, Sublime Text, в залежності від використовуваної мови, буде пропонувати різні варіанти для завершення запису. Редактор також автоматично завершує створені користувачем змінні.

Темний фон Sublime Text призначений для збільшення контрастності тексту. Основні елементи синтаксису виділені різними кольорами, які краще поєднуються з темним тлом, ніж зі світлим.

Sublime Text дозволяє користувачеві збирати програми і запускати їх без необхідності перемикатися на командний рядок. Користувач також може налаштувати свою систему збирання та включити автоматичну збірку програми кожного разу при збереженні коду.

2) AMPPS – це набір рішень, що включає Apache, MySQL, MongoDB, PHP, Perl & Python для Windows, Linux і Mac. AMPPS поставляється з більш ніж 300 готовими PHP веб-додатками та з різними версіями PHP. AMPPS доступний на платформах Windows та Macintosh. AMPPS використовується для розробки PHP з використанням MySQL.

3) phpMyAdmin – веб-додаток з відкритим кодом, написаний мовою PHP із графічним вебінтерфейсом для адміністрування бази даних MySQL або MariaDB. PhpMyAdmin дозволяє через браузер здійснювати адміністрування сервера MySQL, запускати запити SQL, переглядати та редагувати вміст таблиць баз даних. Ця програма користується великою популярністю у веб-розробників, оскільки дозволяє керувати базою даних MySQL без вводу SQL команд через дружній інтерфейс і з будь-якого комп'ютера, під'єданого до Інтернету, без необхідності встановлення додаткового програмного забезпечення. На сьогоднішній день phpMyAdmin широко застосовується на практиці.

4) cPanel – це програмне забезпечення панелі керування веб-хостингом. Воно надає графічний інтерфейс (GUI) і засоби автоматизації, призначені для спрощення процесу розміщення веб-сайту для власника веб-сайту або «кінцевого користувача». Це дозволяє адмініструвати через стандартний веб-

браузер із використанням трирівневої структури. Хоча cPanel обмежується керуванням одним обліковим записом хостингу, cPanel & WHM дозволяє адмініструвати весь сервер. Окрім графічного інтерфейсу, cPanel також має командний рядок і доступ на основі API, що дозволяє стороннім постачальникам програмного забезпечення, організаціям веб-хостингу та розробникам автоматизувати стандартні процеси адміністрування системи.

3. ПРОГРАМНА РЕАЛІЗАЦІЯ ІНФОРМАЦІЙНОЇ ВЕБ-СИСТЕМИ СПОРТИВНО-ОЗДОРОВЧОГО ТАБОРУ «ОРЛЯТКО»

Весь процес проектування сайту був розділений на декілька послідовних етапів:

- провести аналіз вимог до веб-сайту;
- розробити макет та логічну структуру сайту;
- зробити верстку шаблону сайту на основі розробленого макету;
- розробити інтерфейси частин сайту;
- створити модуль табірної спільноти;
- провести тестування.

Далі ми докладніше розглянемо кожен із поданих етапів.

3.1 Аналіз вимог до веб-сайту оздоровчого табору

Після проведення аналізу задач, поставлених перед сайтом, були визначені основні вимоги до веб-сайту:

- 1) представлення спортивно-оздоровчого закладу в мережі Інтернет;
- 2) подання актуальної інформації про послуги та їх вартість;
- 3) комунікація з батьками на предмет необхідних речей та документів перед заїздом дітей на оздоровчу табірну зміну;
- 4) можливість надання контенту декількома мовами;
- 5) розміщення відповідних контактів закладу;
- 6) надання можливості приєднання до спільноти (доступ до аудіо-, відео- та фотоконтенту);
- 7) можливість проведення розсилки електронних листів зареєстрованим користувачам.

Основними користувачами сайту є діти віком 7-16 років (які бажають відпочити в оздоровчому таборі) та їх батьки. Беручи це до уваги, необхідно виконати розробку простого, легкого, але в той же час привабливого та

незабутнього дизайну основних сторінок сайту. Так само важливим фактором останнім часом є наявність мобільної версії сайту, що також є однією із основних вимог до сайту, що розробляється.

З огляду на згадані вище вимоги, найбільш оптимальним засобом для розробки сайту такого роду, на наш погляд, є використання мови розмітки документів – HTML, формальної мови опису зовнішнього вигляду документа, написаного за допомогою мови розмітки – CSS, мови програмування прототипноорієнтованого сценарного типу JavaScript (та її бібліотеки-функції jQuery), скриптової мови програмування загального призначення PHP з використанням баз даних MySQL та веб-сервера Apache.

Були також розглянуті вимоги до функціональних характеристик. Сайт повинен забезпечувати можливість виконання наступних функцій:

- зберігання актуальної інформації про заклад, послуги, контакти тощо;
- реєстрація користувачів і внесення відповідної інформації (логін, пароль, email) до бази даних спільноти;
- функція входу вже зареєстрованих членів спільноти;
- відтворення аудіо- та відеоконтенту закладу для зареєстрованих користувачів;
- функція перегляду архіву фотогалереї для зареєстрованих членів спільноти «Орлятко»;
- розсилка новин зареєстрованим користувачам на електронну пошту.

Відповідно до термінології, usability (юзабіліті) сайту – це ступінь ефективності, продуктивності та зручності взаємодії людини з інтерфейсом. Також часто під цим словом об'єднують методи підвищення ефективності роботи веб-сайту.

Власник будь-якого сайту бажає бачити на своєму ресурсі якомога більше відвідувачів. Якщо ж веб-сайт розроблений для отримання прибутку, то цифри відвідуваності та конверсії стають життєво необхідними.

Для підвищення якості сайту з погляду юзабіліті нами було складено перелік критеріїв, якого ми дотримувались при розробці клієнтської частини сайту табору. Основним принципом у вирішенні питань юзабіліті стало правило: «Простота та зручність».

1. Дизайн сайту.

- a) Простота та мінімалістичність (інформація для користувача надається у структурованому вигляді і тільки в тому обсязі, який необхідний на даному етапі).
- b) Доступність (основна інформація та функції сайту мають бути доступними для користувача на всіх сторінках сайту).
- c) Актуальність (дизайн сайту має відповідати останнім тенденціям у веб-дизайні, але в той же час не рясніти графічними ефектами, щоб не відволікати користувачів від необхідної інформації).
- d) Легкість (швидкість завантаження сторінки має бути максимальною, але в той же час якість графічних елементів потрібно зберегти на гранично високому рівні).
- e) Універсальність (сторінки сайту мають коректно відображатись як на стаціонарних комп'ютерах, так і на мобільних пристроях).

2. Швидкість роботи сайту.

Необхідно забезпечити стабільно високу швидкість обробки запитів та завантаження сторінки, займаючи при цьому мінімальну кількість ресурсів сервера.

3. Зручність навігації.

Простота навігації сайтом – це один з найважливіших критеріїв при розробці проекту. Користувач повинен інтуїтивно та безперешкодно переміщатися сайтом, отримувати необхідну інформацію чи послугу.

Після створення веб-сайту потрібно змусити його працювати на себе, тобто зробити його відомим цільовій аудиторії. Перерахуємо основні інструменти, за допомогою яких можна вирішувати це завдання:

1. Розробка та оптимізація SEO-компонентів сторінок сайту під час проектування сайту.
2. Розміщення (індексування) сайту в пошукових системах, каталогах та рейтингах, таких як Google, Bing, Yahoo тощо.
3. Реклама на тематичних та відвідуваних сайтах.
4. Контекстна реклама.
5. Участь у банерообмінних мережах.

Розробивши та опублікувавши сайт в Інтернеті, його корисно зареєструвати у пошукових системах, рейтингах, каталогах. Це дозволить легко виявити його користувачам Internet, які цікавляться сайтами даної тематики і дозволить розробнику стежити за відвідуваністю сайту та мати додаткову статистичну інформацію про його відвідування.

Реклама на часто відвідуваних сайтах (instagram.com, facebook.com тощо) – найефективніша. Корисну та цікаву для відвідувачів (потенційних користувачів) інформацію можна розмістити у різних розділах та сприяти позитивному іміджу організації.

Сьогодні створення веб-сайту може окупати витрати далеко не для всіх напрямків діяльності. Тому необхідно максимально ефективно провести роботу з SEO-оптимізації нашого веб-сайту. Цей етап значно скоротить витрати на подальше просування ресурсу в Інтернеті.

3.2 Проектування макета та логічної структури веб-сайту

Під створенням макета сайту, так званого «каркаса», мається на увазі розробка структури сторінок. Після завершення цієї стадії, загалом, стає зрозуміло, як виглядатиме сайт.

Створення макета має велике значення з точки зору юзабіліті сайту, і просто необхідне для розробки зручної системи навігації майбутнім сайтом. Крім того, макет сайту є основою для подальшої роботи над дизайном.

Макети розробляються окремо для кожної з основних сторінок сайту. Далі усуваються всі зауваження, що виникли, і враховуються побажання замовника, внаслідок чого йому на затвердження пропонується остаточний варіант макета. У нашому випадку було розроблено макети двох основних сторінок сайту. У них було враховано основні побажання замовника, як з точки зору дизайну, так і з точки зору функціональності.

Обговоривши актуальність відображення деяких блоків у мобільних версіях сайту, були розроблені макети для планшетних комп'ютерів та мобільних телефонів. В результаті проведених робіт ми отримали макети сторінок «Дитячий відпочинок» (Рис. 3.1) та «Спільнота» (Рис. 3.2) у трьох варіантах відображення: для персональних комп'ютерів, для планшетних комп'ютерів, для мобільних телефонів. Всі інші частини сайту отримали схоже оформлення та стандартні елементи інтерфейсу.

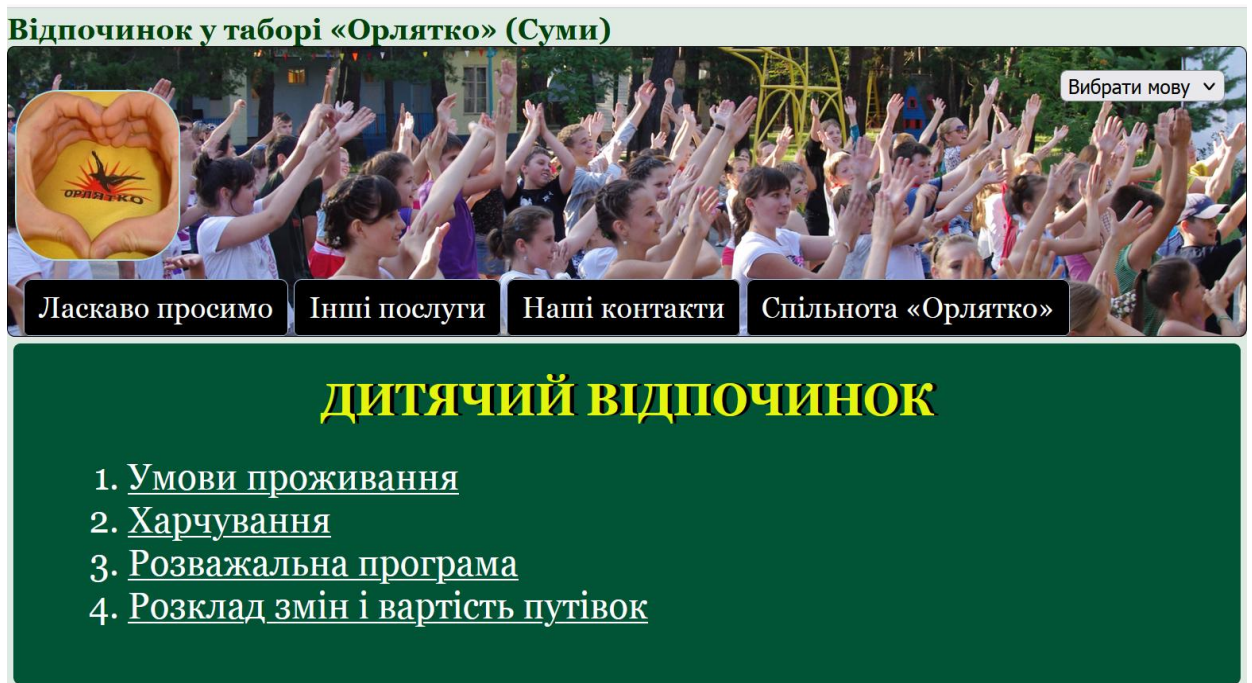


Рисунок 3.1 – Сторінка «Дитячий відпочинок»



Рисунок 3.2 – Сторінка «Спільнота "Орлятко"»

Щодо логічної структури нашого веб-сайту, то вона передбачає декілька рівнів: головна сторінка та розділи сайту, що містять прямі веб-посилання на інші сторінки. На рисунку 3.3 наведено логічну структуру веб-сайту СОТ «Орлятко», яка найбільш відповідає визначеним критеріям.

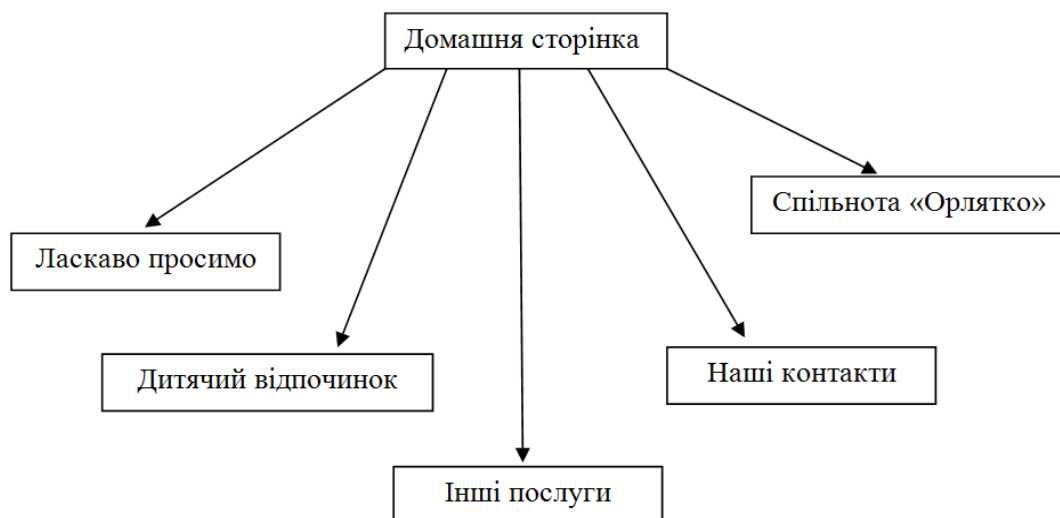


Рисунок 3.3 – Логічна структура веб-сайту СОТ «Орлятко»

Отже, структура веб-сайту була розроблена з урахуванням визначених вимог на етапі проектування. Для всіх основних сторінок сайту були спроектовані макети. Логічна структура вибудована відповідно до вимог замовника з можливістю подальшого масштабування та розширення функціоналу сайту (Додаток А).

3.3 Проектування дизайну та інформаційного наповнення сайту

Дизайн веб-сайту – це оформлення контенту, сукупність усіх графічних елементів на веб-сторінках. Колірна гама, що використовується для розробки інтерфейсу, відіграє важливу роль. Від неї залежить лаконічність візуального ряду, сприйняття тексту для читання тощо. Щоб позитивно вплинути на користувача, палітра повинна відповідати гармонійному поєднанню кольорів.

Існують певні закономірності для вибору колірної гами:

- 1) кольори мають відповідати фірмовому стилю компанії;
- 2) комбінація кольорів повинна відповідати цільовій аудиторії даного сайту;
- 3) не слід використовувати велику кількість кольорів;
- 4) кольори повинні гармонувати між собою;
- 5) потрібно забезпечити контраст між фоном та звичайним текстом [32].

При розробці інтерфейсу сайту табору було вибрано три основні кольори: темно-зелений, світло-зелений та білий (Рис.3.4).

Оренда, заходи і збори



Рисунок 3.4 – Одна зі сторінок сайту табору

Зелений колір у всіх його відтінках асоціюється з природою: з травою, рослинами та деревами. Він також уособлює ріст і оновлення, будучи кольором весни і відродження. Інша асоціація — «отримання зеленого світла» для подальших дій, асоціація з діями [48]. Зелений дарує відчуття оптимізму та бадьорості. Зелений символізує здоров'я, нові починання та багатство.

Білий колір у веб-дизайні вважається нейтральним, таким, що дозволяє зосередитись на інших кольорах. За допомогою білого можна передати відчуття чистоти і простоти – тому його так часто застосовують у мінімалістичних проектах [61]. Крім того, текст білого кольору, на нашу думку, дуже добре виглядає на темно-зеленому фоні.

У верхній частині домашньої сторінки (й усіх інших сторінок) розміщене навігаційне меню у вигляді текстових посилань, яке варіюється в залежності від поточної сторінки, а в даному випадку складається з наступних розділів: «Дитячий відпочинок», «Інші послуги», «Наші контакти», «Спільнота “Орлятко”» (Рис. 3.5).



Рисунок 3.5 – Горизонтальне меню

При наведенні курсору на текстове посилання навігаційного меню текст посилання підсвічується жовтим кольором і спрацьовує анімація фону посилання (Рис. 3.6).



Рисунок 3.6 – Анімація при наведенні курсору

Однією з особливостей структури веб-сайту табору є наявність drop-down списку, в якому можна вибрати зручну для себе мову (Рис. 3.7). На сьогодні мови дві.

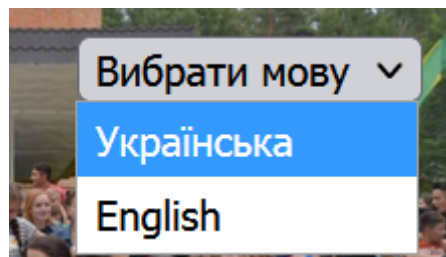


Рисунок 3.7 – Drop-down список вибору мови

Для кожного розділу створено окремі сторінки, на яких міститься відповідна інформація, медіаматеріали та гіперпосилання.

При проектуванні інформаційного наповнення сайту та засобів навігації слід дотримуватися чотирьох базових принципів, які засновані на сприйнятті інформації людиною:

1) використання позначень – застосування слів і термінів, які є сталими і зрозумілими для більшості відвідувачів сайту, слід розміщувати змістовні елементи, посилання і елементи навігації з їх позначеннями, а також уникати невиразних термінів на сторінці;

2) доцільність – розділи сайту повинні містити інформацію та елементи, які відповідають даному розділу чи фрагменту сторінки, невідповідні елементи краще перенести в інше місце або взагалі прибрати;

3) одноманітність – однакові елементи на сторінках повинні мати один і той же розмір і знаходитися в одному і тому ж місці;

4) розділення на частини – відвідувачі сайту краще орієнтуються і швидше знаходять потрібні для них матеріали, коли вони візуально поділені на групи [32].

Таким чином, дизайн був виконаний у гармонійній колірній гамі з врахуванням особливостей зорового сприйняття користувачів (Додаток Б). Було досягнуто максимально простого, легкого для сприйняття дизайну і в той же час такого дизайну, який запам'ятовується. Для належного інформаційного наповнення сайту були взяті до уваги базові принципи сприйняття інформації людиною.

3.4 Розробка модуля табірної спільноти

Окрема увага повинна бути приділена розробці розділу спільноти «Орлятко», оскільки цей модуль, на відміну від інших, включає в себе додаткові функції для взаємодії з користувачами:

- процес реєстрації з відправленням на електронну пошту листа з відповідним посиланням для підтвердження реєстрації;
- процес входу до спільноти;
- можливість перегляду контенту з архіву спільноти.

При переході за посиланням «Спільнота "Орлятко"» користувачу пропонується зареєструватися у спільноті або увійти, якщо він вже зареєстрований (Рис. 3.8).

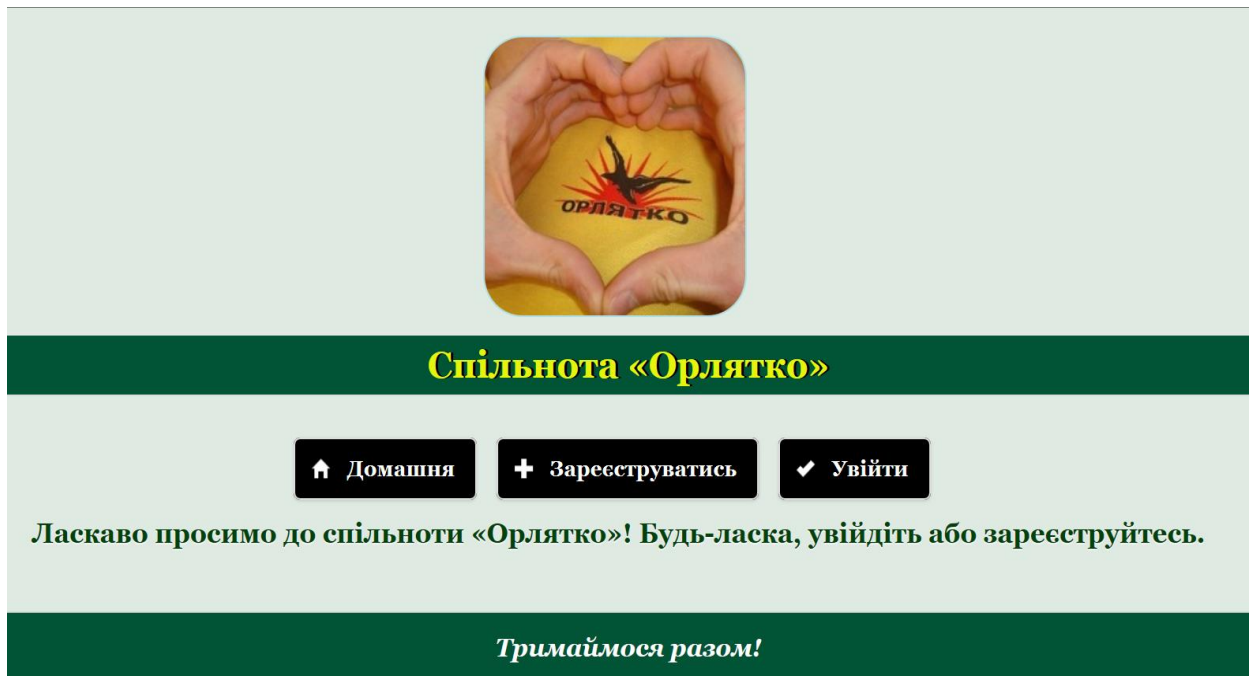


Рисунок 3.8 – Стартова сторінка спільноти "Орлятко"

Для успішного проходження процесу реєстрації необхідно ввести логін, email, пароль, підтвердження пароля та шестизначний код (Рис. 3.9).

Рисунок 3.9 – Форма реєстрації

Під час введення логіна, за допомогою асинхронного обміну даними засобами AJAX jQuery (Додаток В) перевіряється, чи існує такий логін в базі даних. Результат відразу відображається під полем введення (Рис. 3.10).

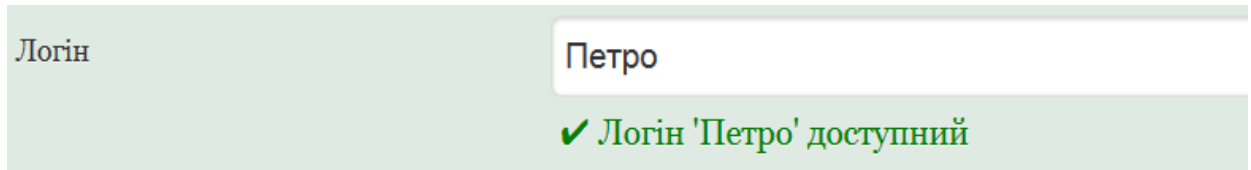
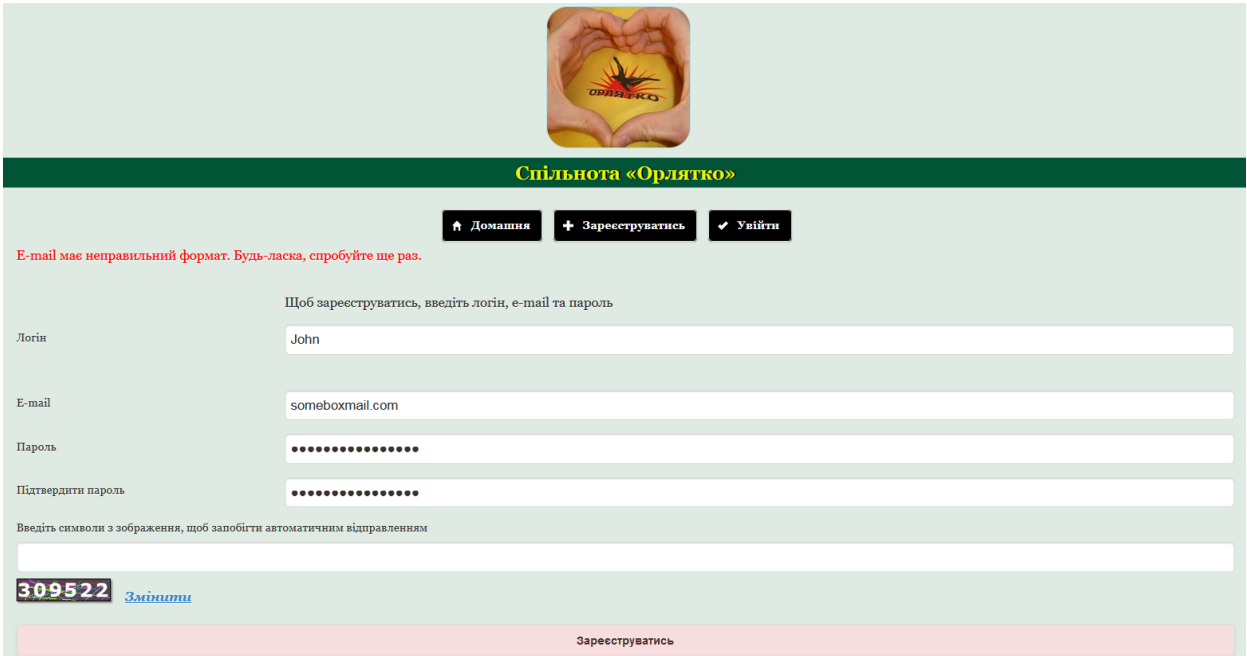


Рисунок 3.10 – Перевірка, чи існує логін в базі даних

Після того як користувач заповнив всі необхідні поля та клікнув кнопку "Зареєструватись", система спочатку перевіряє, чи заповнені всі поля. Якщо так, то до кожного поля з міркувань безпеки застосовуються наступні PHP-функції:

- 1) `strip_tags` – видаляє теги HTML та PHP;
- 2) `htmlentities` – перетворює всі можливі символи у відповідні HTML-сутності;
- 3) `stripslashes` – видаляє екранування символів;
- 4) `real_escape_string` – екранує спеціальні символи у рядку для використання в SQL-виразі, використовуючи поточний набір символів з'єднання.

Наступним кроком є перевірка формату введеного email. Це реалізується за допомогою таких PHP-функцій, як `strpos` та `preg_match`. У випадку невідповідності введеного користувачем email необхідному формату система виводить відповідне повідомлення (Рис. 3.11).



Спільнота «Орлятко»

[Домашня](#)
[Зареєструватись](#)
[Увійти](#)

E-mail має неправильний формат. Будь-ласка, спробуйте ще раз.

Щоб зареєструватись, введіть логін, e-mail та пароль

Логін:

E-mail:

Пароль:

Підтвердити пароль:

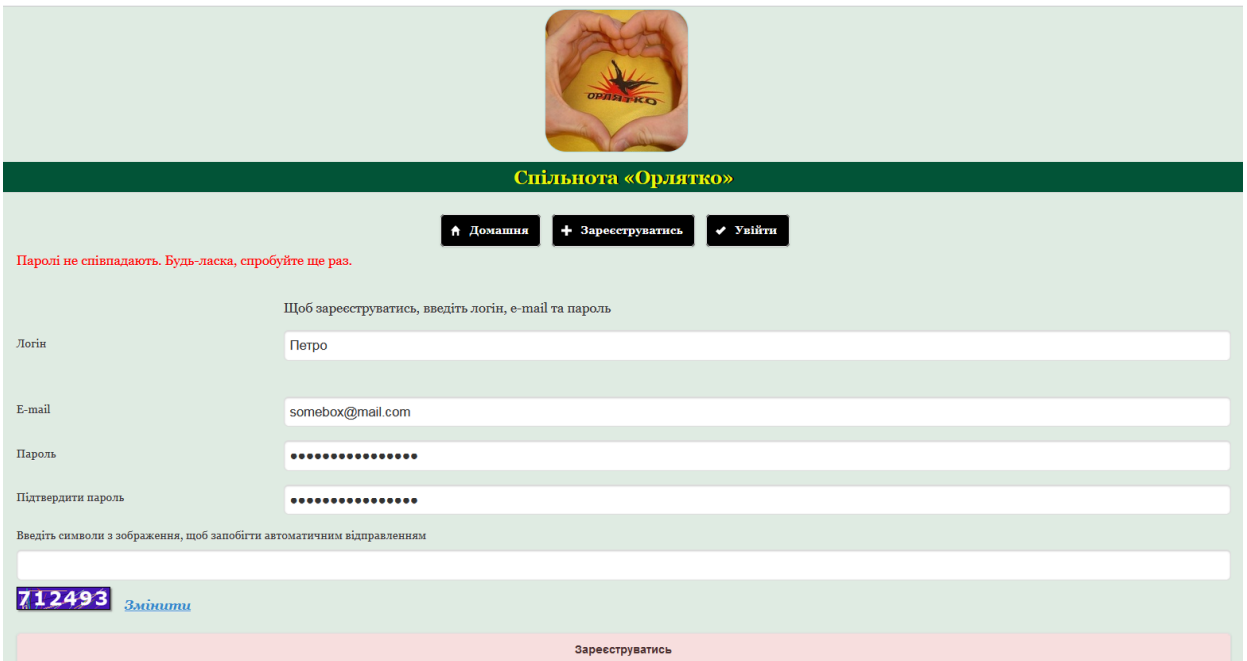
Введіть символи з зображення, щоб запобігти автоматичним відправленням

309522 [Змінити](#)

Зареєструватись

Рисунок 3.11 – Повідомлення про неправильний формат введеної електронної адреси

Окрім перевірки формату електронної пошти система також порівнює дані полів "Пароль" та "Підтвердження пароля". У разі неспівпадіння введених даних, на екран виводиться повідомлення про це, і процес реєстрації продовжується (Рис. 3.12).



Спільнота «Орлятко»

[Домашня](#)
[Зареєструватись](#)
[Увійти](#)

Паролі не співпадають. Будь-ласка, спробуйте ще раз.

Щоб зареєструватись, введіть логін, e-mail та пароль

Логін:

E-mail:

Пароль:

Підтвердити пароль:

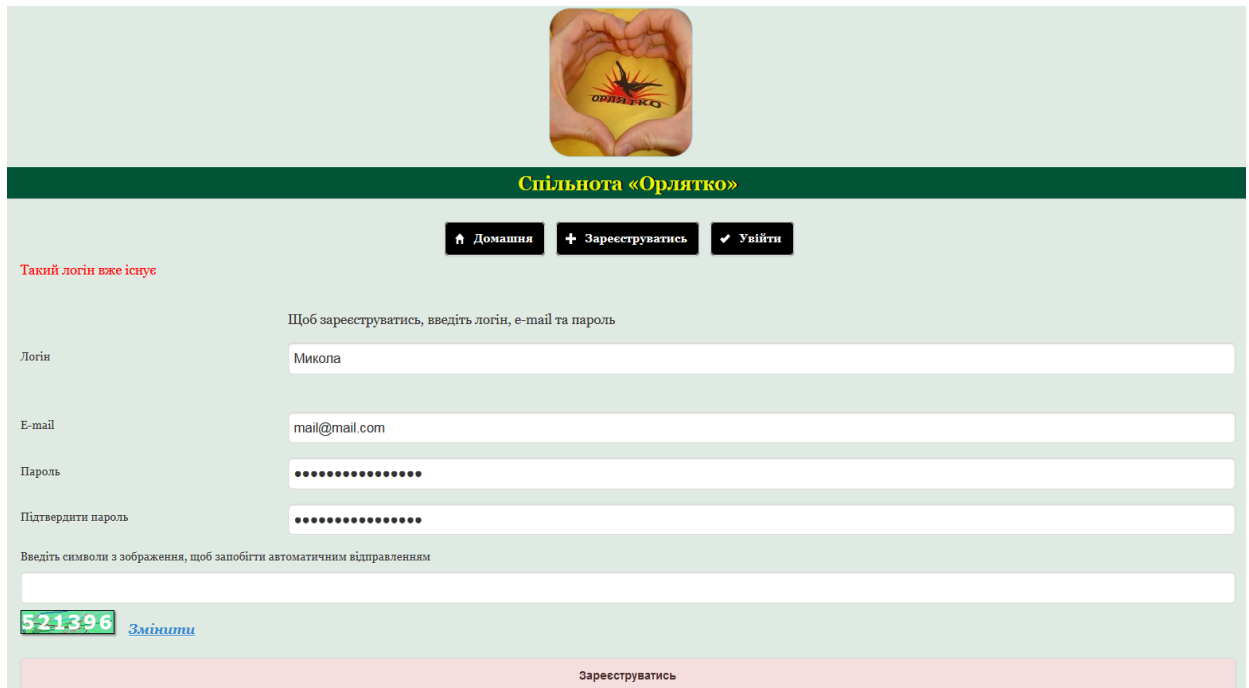
Введіть символи з зображення, щоб запобігти автоматичним відправленням

712493 [Змінити](#)

Зареєструватись

Рисунок 3.12 – Повідомлення про неспівпадіння паролів


У випадку якщо попри попередження про наявність введеного логіна в базі даних (Рис. 3.9), користувач продовжив заповнення реєстраційної форми та клікнув кнопку "Зареєструватись", в системі передбачена додаткова перевірка зі зверненням до бази даних. При наявності введеного логіна в базі даних система виводить повідомлення: "Такий логін вже існує" (Рис. 3.13).



The screenshot shows the registration page for the community "Спільнота «Орляtko»". At the top, there is a logo of two hands holding a yellow circle with a red and black bird-like symbol and the text "ОРЛЯТКО". Below the logo, the text "Спільнота «Орляtko»" is displayed. A navigation bar contains three buttons: "Домашня", "Зареєструватись", and "Увійти". A red error message "Такий логін вже існує" is displayed. Below the message, there is a form with the following fields: "Логін" (Login) with the value "Микола", "E-mail" with the value "mail@mail.com", "Пароль" (Password) with masked characters, and "Підтвердити пароль" (Confirm password) with masked characters. A CAPTCHA code "521396" is displayed with a "Змінити" (Change) link. At the bottom, there is a "Зареєструватись" (Register) button.

Рисунок 3.13 – Повідомлення про існування логіну після перевірки у базі даних.

Насамкінець процес реєстрації передбачає введення користувачем шестизначного captcha-коду для підтвердження того, що реєструється людина, а не комп'ютерна програма, яка намагається отримати доступ до захищеного паролем облікового запису. За бажанням користувач може змінити captcha-код. Якщо був введений неправильний captcha-код, система пропонує спробувати ще раз (Рис. 3.14).



Спільнота «Орлякко»

[🏠 Домашня](#) [+ Зареєструватись](#) [✔ Увійти](#)

Введені Вами символи не співпадають із зображеними на малюнку. Спробуйте ще раз.

Щоб зареєструватись, введіть логін, e-mail та пароль

Логін:

E-mail:

Пароль:

Підтвердити пароль:


Введіть символи з зображення, щоб запобігти автоматичним відправленням

[Змінити](#)

Зареєструватись

Рисунок 3.14 – Повідомлення про неправильно введений captcha-код

Окрім усіх вищезазначених перевірок, система також перевіряє негативні сценарії, пов'язані з незаповненням будь-якого одного, декількох або всіх полів (Рис. 3.15).



Спільнота «Орлякко»

[🏠 Домашня](#) [+ Зареєструватись](#) [✔ Увійти](#)

Не всі поля були заповнені

Щоб зареєструватись, введіть логін, e-mail та пароль

Логін:

E-mail:

Пароль:

Підтвердити пароль:

Введіть символи з зображення, щоб запобігти автоматичним відправленням

[Змінити](#)


Зареєструватись

Рисунок 3.15 – Повідомлення про те, що під час реєстрації не всі поля були заповнені

При успішному проходженні реєстрації система відправляє користувачеві електронного листа з посиланням, перейшовши за яким він завершує процес реєстрації. В базу даних додається запис з інформацією про користувача і в тому числі адреса його електронної пошти, використовуючи яку можна проводити розсилку з актуальною інформацією.

Вхід до спільноти можливий після того, як користувач зареєструвався. Форма входу містить два поля: «Логін» та «Пароль», а також посилання «Забули пароль?» на випадок, якщо користувач не може згадати пароль, який він вводив під час реєстрації (Додаток Г).

Як і у випадку реєстрації під час процесу входу система проводить декілька перевірок даних, що вводяться. Це перевірки таких негативних сценаріїв, як наявність в базі даних зазначеного логіна та правильність введеного пароля. Для кожного з випадків існує відповідне повідомлення, яке виводиться системою на екран для інформування користувача (Рис. 3.16, 3.17).



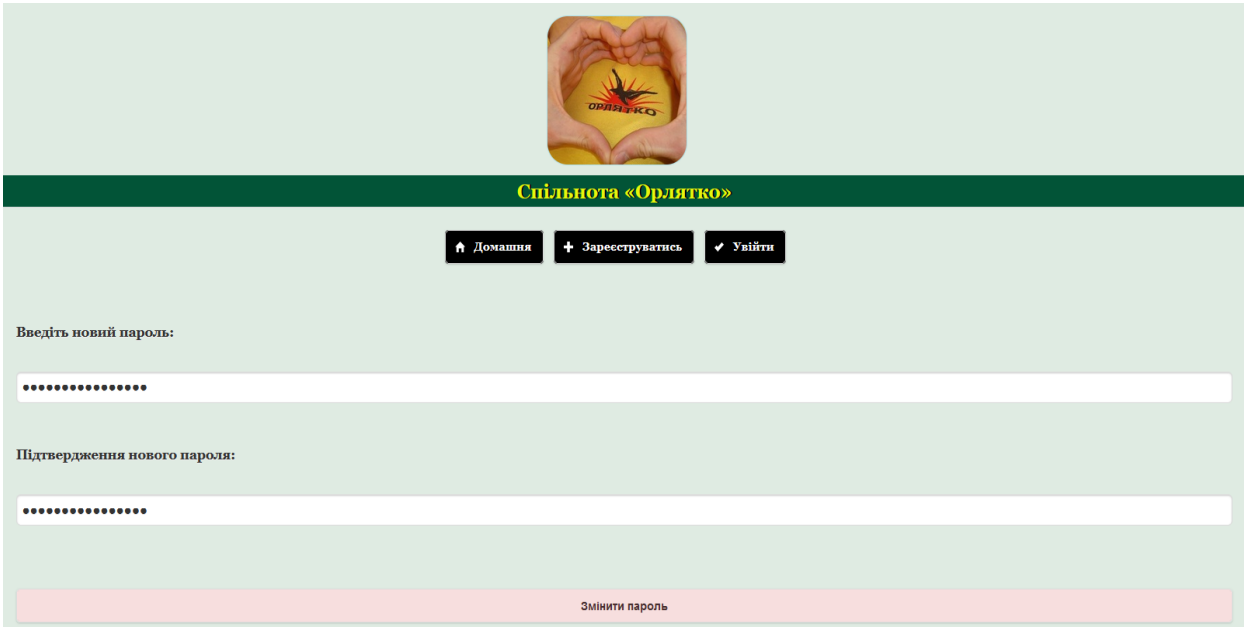
The image shows a web interface for the community «Орлятко». At the top, there is a logo featuring two hands holding a yellow circle with a red and black bird-like figure and the text «ОРЛЯТКО». Below the logo is a dark green header with the text «Спільнота «Орлятко»». Underneath the header are three navigation buttons: «Домашня», «Зареєструватись», and «Увійти». The main content area displays a red error message: «Неправильний логін. Спробуйте ще раз.» Below this message is the text «Будь-ласка, введіть логін та пароль». There are two input fields: the first is labeled «Логін» and contains the text «Петро»; the second is labeled «Пароль» and contains a series of dots. Below the password field is a link «[Забули пароль?](#)». At the bottom of the form is a pink button labeled «Увійти».

Рисунок 3.17 – Повідомлення про неправильний логін



Рисунок 3.18 – Повідомлення про неправильний пароль

Як зазначалося вище, посилання «Забули пароль?» за необхідності може бути використане відвідувачем сайту. Для змінення пароля користувачеві пропонується форма, в якій потрібно ввести адресу своєї електронної пошти і клікнути кнопку «Відправити» (Рис. 3.19). Після виконання цих дій, система перевіряє у базі даних наявність введеної електронної адреси. Негативний сценарій обумовлює вивід на екран повідомлення про відсутність вказаної електронної адреси у базі даних. Якщо ж зазначена електронна адреса наявна, то на неї відправляється електронний лист з інструкціями, як змінити пароль. У листі міститься посилання, яке направляє користувача на веб-сторінку з полями «Введіть новий пароль» та «Підтвердження нового пароля» (Рис. 3.19).



Спільнота «Орлятко»

[↑ Домашня](#) [+ Зареєструватись](#) [✓ Увійти](#)

Введіть новий пароль:

.....


Підтвердження нового пароля:

.....

Змінити пароль

Рисунок 3.19 – Форма зміни пароля

У випадку якщо обидва поля заповнені та введений і підтверджений паролі співпадають, система змінює пароль у базі даних після того, як користувач тисне кнопку «Змінити пароль». На екрані з'являється повідомлення про те, що пароль змінено, і можна увійти до системи, використавши щойно оновлений пароль (Рис. 3.20).



Спільнота «Орлятко»

[↑ Домашня](#) [+ Зареєструватись](#) [✓ Увійти](#)

Ваш пароль змінено

Тепер можна увійти

Рисунок 3.20 – Повідомлення про змінення пароля

Слід зазначити, що як і у випадку реєстрації, до всіх полів форм, які стосуються входу до спільноти з міркувань безпеки застосовуються аналогічні РНР-функції. Це допомагає запобіганню несанкціонованого доступу до бази даних, чим можуть скористатися зловмисники.

Після входу до спільноти система надає користувачеві доступ до архіву спільноти (Рис. 3.21). В архіві міститься фотогалерея, відео табору і табірна музика.

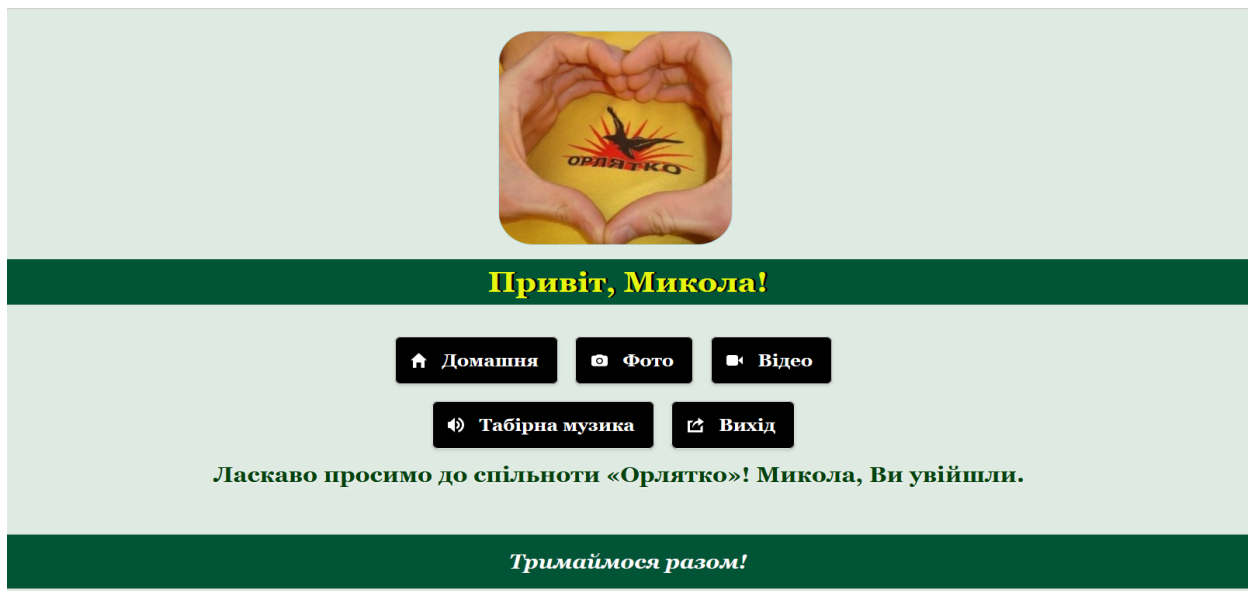


Рисунок 3.21 – Інтерфейс після входу

Таким чином, розділ табірної спільноти надає користувачам можливість доступу до архіву табору, до участі у табірному житті незалежно від їх місця перебування.

Веб-сайт розроблено на замовлення Спортивно-оздоровчого табору «Орлятко». Перегляд результатів доступний за посиланням <https://orliatko.com.ua>

3.5 Тестування на кросбраузерність і коректність відображення

Після розробки сайту було перевірено відповідність сайту поставленим задачам, перевірено працездатність посилань, перевірено зрозумілість та інтуїтивність інтерфейсу для користувачів.

Тестування головної сторінки в браузері Internet Explorer 11 зображено на рисунку 3.22.

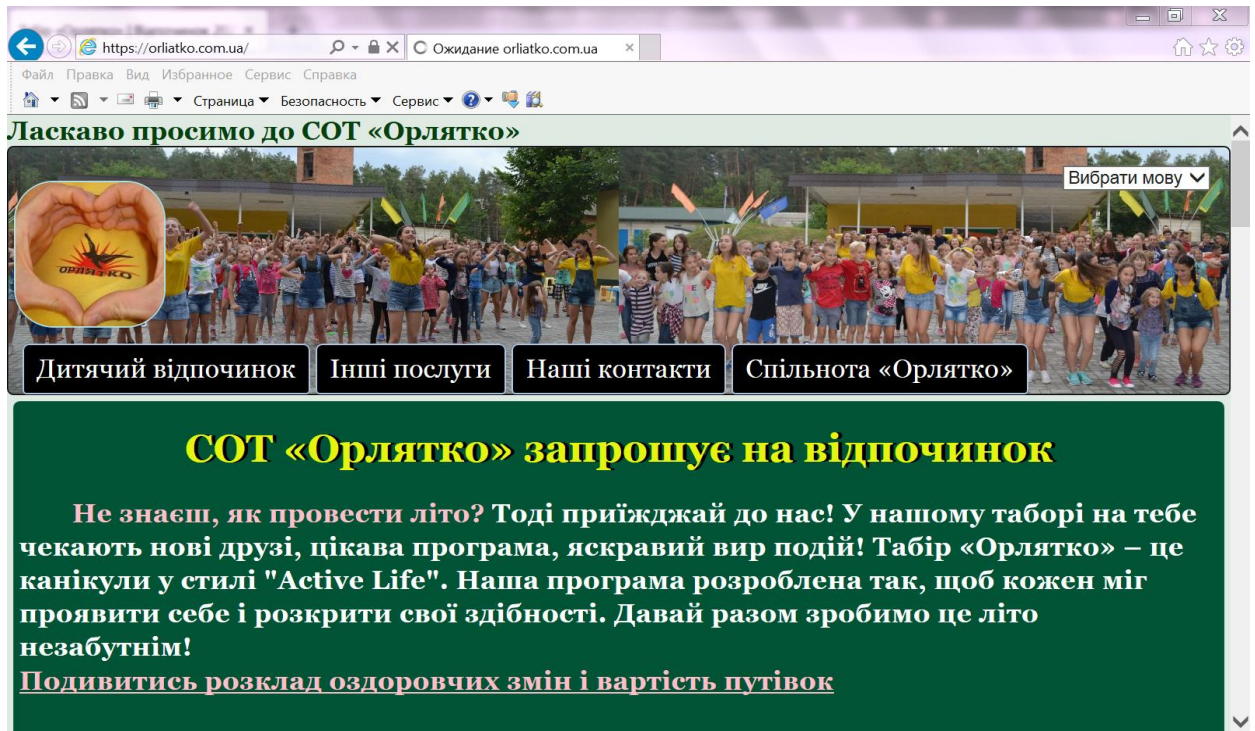


Рисунок 3.21 – Головна сторінка в браузері Internet Explorer 11

Тестування головної сторінки в браузері Mozilla Firefox зображено на рисунку 3.23.

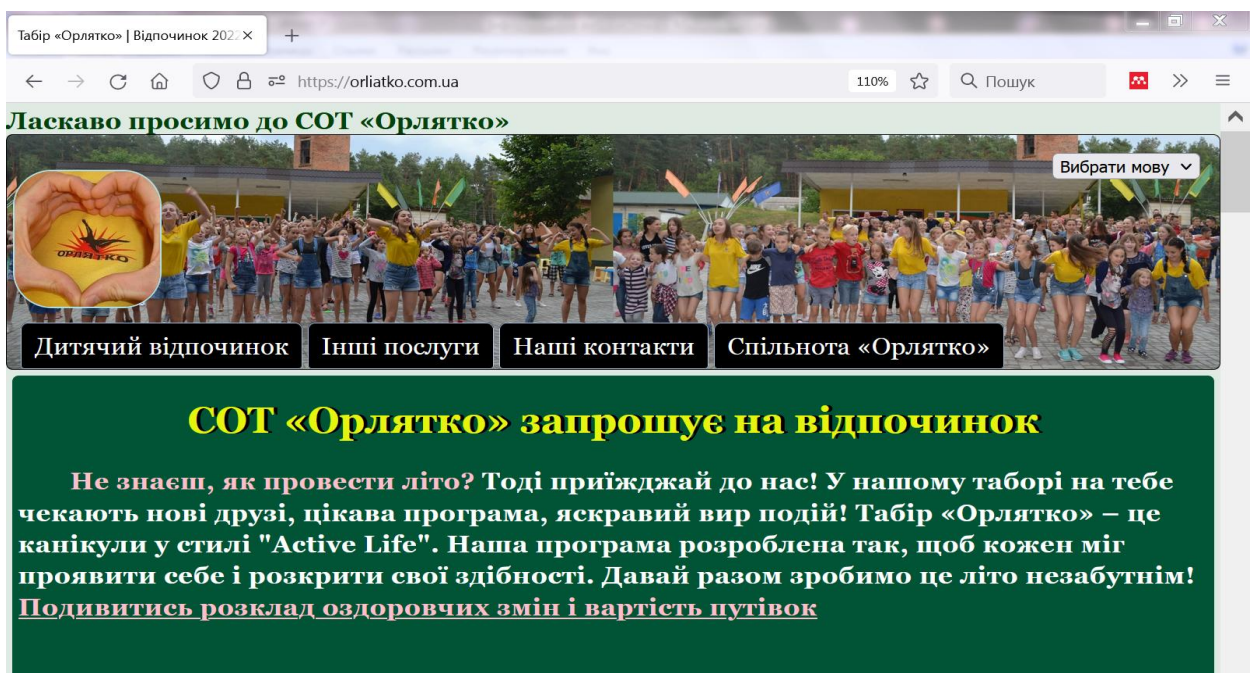


Рисунок 3.23 – Головна сторінка в браузері Mozilla Firefox

Тестування головної сторінки в браузері Google Chrome зображено на рисунку 3.24.

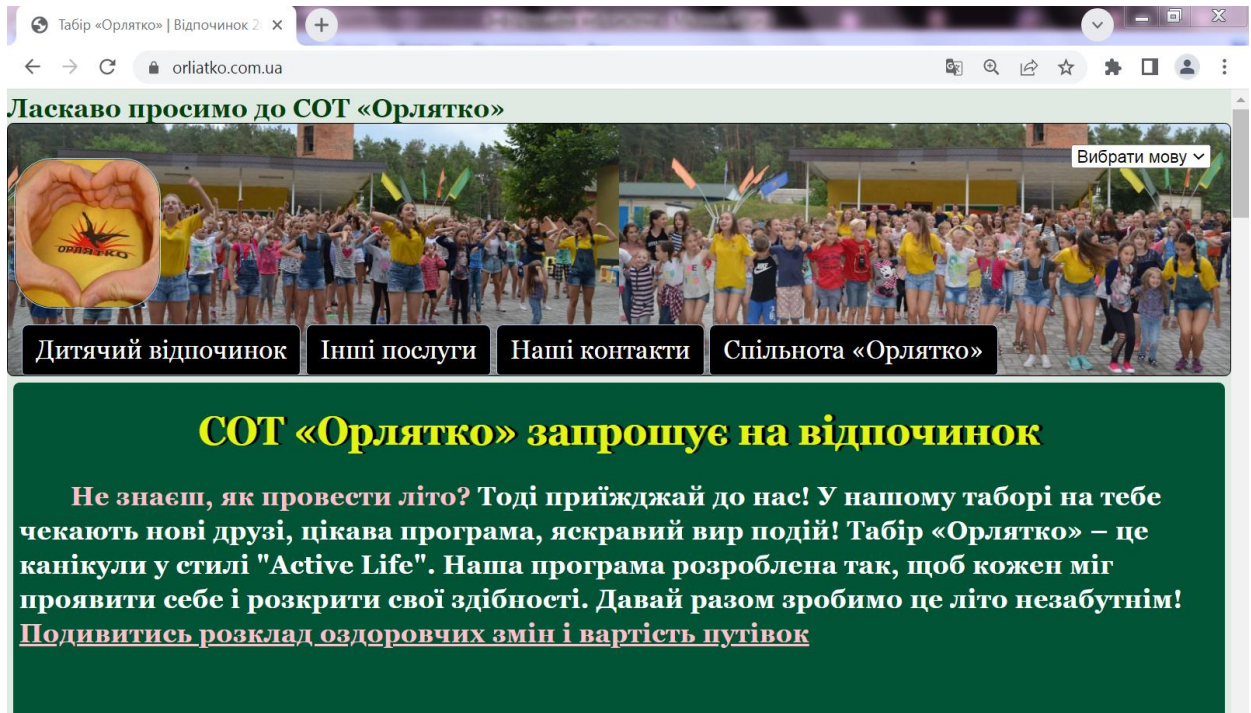


Рисунок 3.24 – Головна сторінка в браузері Google Chrome

Тестування головної сторінки в браузері Safari зображено на рисунку 3.25.

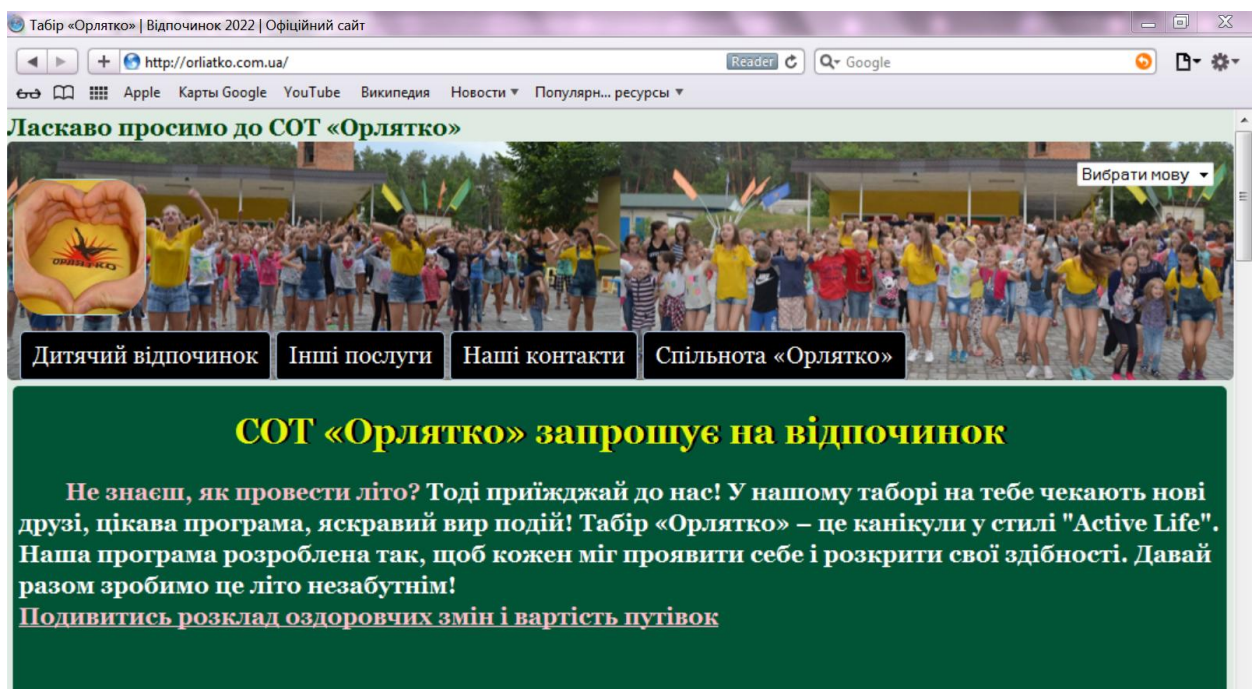


Рисунок 3.25 – Головна сторінка в браузері Safari

Таким чином, проведені тестування сайту на кросбраузерність виявило, що сайт відображається однаково в таких популярних браузерах, як Internet Explorer, Mozilla Firefox, Google Chrome, Safari. Сайт коректно працює та придатний для подальшого використання.

ВИСНОВКИ

За результатами дипломної роботи було виконане наступне:

- проаналізована наукова література, яка стосується сутності, значення та компонентів веб-сайтів;
- визначено типи та принципи проектування веб-сайтів;
- акцентовано увагу на використанні веб-сайту як засобу комунікації з користувачами;
- обґрунтована потреба у веб-сайті для оздоровчого табору;
- розглянуто технології для реалізації front-end та back-end частини веб-сайту, а також додаткові програмні засоби для його розробки;
- виконана програмна реалізація інформаційної веб-системи для оздоровчого закладу.

Для розробки веб-системи були використані такі технології, як HTML, CSS, Javascript, jQuery, PHP, MySQL та Apache.

У звіті був проведений аналіз вимог до веб-сайту оздоровчого табору, спроектований макет та логічна структура веб-сайту, створений дизайн та інформаційне наповнення веб-сайту, розроблений модуль табірної спільноти, а також проведено тестування на кросбраузерність і коректність відображення.

В результаті роботи можемо зробити висновок, що створення та використання сайту даного оздоровчого закладу може стати чудовим вирішенням проблеми комунікації з користувачами послуг та допомогти залучити до спільноти більшу кількість людей, що сприятиме популяризації спорту, активного відпочинку та оздоровлення.

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Берд Дж. Веб-дизайн: довідник розробника / Дж. Берд. К., 2012. 224 с.
2. Березко О. Л. Персоніфікаційна класифікація веб-сайтів // ВЕЖПТ, 2009. №2 (40). [Електрон. ресурс]. – Режим доступу: <https://cyberleninka.ru/article/n/personifikatsiy-na-klasifikatsiya-veb-saytiv>
3. Бородаєв Д. В. Веб-сайт як об'єкт графічного дизайну. Монографія. Х.: «Септима ЛТД», 2006. 288 с.
4. Бублик М. І. Інформаційні системи в маркетингу. Конспект лекцій. Львів: В-во ЛДІНТУ, 2007. 162 с.
5. Бурило Ю.П. Веб-сайт як інформаційний ресурс та об'єкт права інтелектуальної власності. Науковий вісник Ужгородського національного університету, 2015. Випуск 34. Том 2. С. 67-70.
6. Васильєв А. Н. Програмування на PHP у прикладах та завданнях. К., 2021. 352 с.
7. Веллінг Л., Томсон Л. Розробка веб-додатків за допомогою PHP та MySQL. К., 2017. 768 с.
8. Дідур О.П. Технології розробки web-додатків. [Електрон. ресурс]. – Режим доступу: <https://conferences.vntu.edu.ua/index.php/mn/mn2019/paper/viewFile/8192/6848>
9. Дронов В.А. PHP и MySQL. 25 уроків для початківців. К., 2021. 432 с.
10. Зандстра М. PHP 8: об'єкти, шаблони та методики програмування. К., 2021. 866 с.
11. Значення бекенду і все, що вам потрібно знати про нього [Електронний ресурс] Режим доступу: <https://blog.back4app.com/ru/значение-бэкенда-и-все-что-вам-нужно-зн/>
12. Калашнікова Л. В. Інтернет-сайти вищих навчальних закладів як інноваційні канали комунікації в умовах формування інформаційного

- суспільства в Україні : автореферат дис. ... к. соціол. н.: спеціальність 22.00.04. / Л. В. Калашнікова. Харків, 2005. 20 с.
13. Каліщук Н.О. Веб-сайт як документально-інформаційна система: функції, структура, зміст. Тернопіль, 2017. 109 с. – Режим доступу: <http://dspace.wunu.edu.ua/handle/316497/17608>
 14. Кириченко О.В., Хрустальов О.О. HTML5+CSS3. Основи сучасного Web-дизайну. К., 2018. 352 с.
 15. Кириченко О., Нікольський О., Дубовик Є. Web на практиці. CSS, HTML, JavaScript, MySQL, PHP для fullstack-розробників. К., 2021. 432 с.
 16. Компанеєтс М.О. Візуальні компоненти веб-сайту. Перспективи розвитку сучасної науки. Харків. 4-5 грудня 2015. С.15-17.
 17. Котеров Д.В., Симдянов І.В. PHP 7. К., 2020. 1088 с.
 18. Купер А. Про інтерфейс. Основи проектування взаємодії. К., 2009. 688 с.
 19. Лебеденко М.С., Лученко І.В. Веб-ресурс як ефективний інструмент маркетингових комунікацій. Вісник Хмельницького національного університету. 2011. №2. Том 1. С. 178-182.
 20. Леонт'єв Б. К. Web-дизайн: Тонкощі, хитрощі та секрети. Х., 2003. 640 с.
 21. Лук'янов М. Ю. PHP. Повне керівництво та довідник функцій. К., 2020. 432 с.
 22. Кращі CSS-фреймворки для Front-End розробників у 2021 році [Електронний ресурс] – Режим доступу: <https://senior.ua/articles/luchshie-cssfreymvorki-dlya-frontend-razrabotchikov-v-2021-godu>
 23. Манн І. Б. Маркетинг на 100 %: ремікс. Як стати гарним менеджером з маркетингу. К., 2009. 416 с.
 24. Москалець Л. Вебсайт як ефективний інструмент рекламування компанії. Матеріали V всеукраїнської науково-практичної конференції, присвяченої 90-річчю Національного університету «Полтавська політехніка імені Юрія Кондратюка» та 30-річчю створення кафедри українознавства,

- культури та документознавства. Полтава. 25 листопада 2020 року. С. 327-331.
25. Ніксон Р. Створюємо динамічні веб-сайти за допомогою PHP, MySQL, Javascript, CSS та HTML5. К., 2019. 816 с.
 26. Пасічник О. Г., Пасічник О. В., Стеценко І. В. Основи веб-дизайну / О. Г. Пасічник, О. В. Пасічник, І. В. Стеценко : [Навч. посіб.]. К.: Вид. група ВНУ, 2009. 336 с.
 27. Пелещишин А.М. Позичонування сайтів у глобальному інформаційному середовищі (монографія). Львів: Видавництво Львівської політехніки, 2007. 260 с.
 28. Поляков Є.В. PHP у прикладах. Х., 2017. 256 с.
 29. Розробка з боку Front end [Електронний ресурс] – Режим доступу: <https://dan-it.com.ua/blog/razrabotka-so-storony-front-end-chto-jeto-takoe-i-chem-otlichaetsja-ot-back-end/>
 30. Резиг Дж., Бибо Б., Марас І. Секрети JavaScript-ніндзя. К., 2019. 544 с.
 31. Роббінс Дж. Н. Веб-дизайн для початківців. HTML, CSS, JavaScript і веб-графіка. Х., 2021. 956с.
 32. Романюк О. Н. Веб-дизайн і комп'ютерна графіка. Навч. посібник/ О.Н. Романюк, Д.І. Кательніков, О.П. Косоєць. В.: ВНТУ, 2007. 147 с.
 33. Сімпсон К. (Ви поки ще не знаєте JS) Познайомтесь, JavaScript. К., 2021. 192с.
 34. Скляр Д. Вивчаємо PHP 7. Посібник зі створення інтерактивних веб-сайтів. К.: «Діалектика», 2020. 464 с.
 35. Уиллер А. Індивідуальність бренду: посібник зі створення, просування та підтримки сильних брендів. К., 2004. 235 с.
 36. Ульман Ларрі. MySQL: Довідник з вивчення мови. К.: ДМК Прес, 2007. 352 с.

37. Фесенко А. М. Веб-сайт як засіб комунікативної діяльності ВНЗ: проблема ефективності використання. Наукові праці. Соціологія. Випуск 232. Том 244. С. 76-79.
38. Фльонов М.Є. PHP. Об'єкти, шаблони та методики програмування. К.: «Діалектика», 2020. 736 с.
39. Фреймворки для веб-розробки [Електронний ресурс] Режим доступу: <https://voll.com.ua/blog/frejmvorki-dlya-veb-razrabotki>
40. Фрейн Б. HTML5 та CSS3. Розробка сайтів для будь-яких браузерів та пристроїв / Бен Фрейн. К., 2014. 304 с.
41. Фрімен Е., Робсон Е. Вивчаємо програмування на JavaScript. К., «Наука та техніка», 2018. 640 с.
42. Хавербеке М. Виразний JavaScript. Сучасне веб-програмування. Х., 2019. 480 с.
43. Чиннатхамбі К. JavaScript з нуля. К., 2021. 400 с.
44. Штанько В.І., Бордюгова Т.Г. Інформаційне суспільство: соціально-філософські проблеми становлення: навч. посібник. Харків: ХНУРЕ, 2012. 172 с.
45. Шурчкова Ю.В. Веб-сайт компанії як комунікативний маркетинговий інструмент. Вісник ВДУ. Серія: Економіка та управління. 2013. №1. С.211-218.
46. Ярکا У., Бойчук О., Ляхович Л. Веб-сайт як засіб комунікативної діяльності суспільства. Information, communication, society. Lviv. 19-21 MAY 2016. P. 90-91.
47. AMPPS. Wikipedia, the free encyclopedia. [Електрон. ресурс]. – Режим доступу: <https://en.wikipedia.org/wiki/AMPPS>
48. Color meanings and the art of using color symbolism. [Електрон. ресурс]. – Режим доступу: <https://99designs.com/blog/tips/color-meanings/>
49. CPanel. Wikipedia, the free encyclopedia. [Електрон. ресурс]. – Режим доступу: <https://en.wikipedia.org/wiki/CPanel>

50. CSS. Wikipedia, the free encyclopedia. [Электрон. ресурс]. – Режим доступа: <https://uk.wikipedia.org/wiki/CSS>
51. Fielding J. Beginning Responsive Web Design with HTML5 and CSS3. Apress, Berkeley, CA, 2014.
52. HTML. Wikipedia, the free encyclopedia. [Электрон. ресурс]. – Режим доступа: <https://uk.wikipedia.org/wiki/HTML>
53. JavaScript - Overview. [Электрон. ресурс]. – Режим доступа: https://www.tutorialspoint.com/javascript/javascript_overview.htm
54. Kriegel Hans-Peter, Schubert Matthias. Classification of Websites as Sets of Feature Vectors. In: Databases and applications. 2004. P. 127-132.
55. Lindemann C, and Littig L. Classifying web sites. In Proceedings of the 16th international conference on World Wide Web. 2007. ACM. P. 1143-1144.
56. Market Share & Web usage statistics, jQuery [Электрон. ресурс]. – Режим доступа: <https://www.similartech.com/technologies/jquery>
57. McIntire P. Visual Design for the Modern Web. Berkeley: New Riders, 2008, 352 p.
58. MySQL Tutorial [Электрон. ресурс]. – Режим доступа: <https://www.javatpoint.com/mysql-tutorial>
59. Najlah Gali, Radu Mariescu Istodor, Pasi Frănti. Functional Classification of Websites. In Proceedings of the Eighth International Symposium on Information and Communication Technology (SoICT 2017). Association for Computing Machinery. New York. 2017. P. 34–41.
60. PhpmyAdmin documentation [Электронный ресурс] – Режим доступа: <https://www.phpmyadmin.net/docs/>
61. Salazar Luis. Effective Examples of Color in Web Design. Instant Shift [Электронный ресурс]. – Режим доступа: <http://www.instantshift.com/2012/11/08/effective-examples-ofcolor-in-web-design/>
62. Sublime Text. Wikipedia, the free encyclopedia. [Электрон. ресурс]. – Режим доступа: https://en.wikipedia.org/wiki/Sublime_Text

63. W3Techs - World Wide Web Technology Surveys [Электрон. ресурс]. – Режим доступа: <https://w3techs.com/>
64. Website. Wikipedia, the free encyclopedia. [Электрон. ресурс]. – Режим доступа: <https://en.wikipedia.org/wiki/Website>
65. What is PHP? [Электрон. ресурс]. – Режим доступа: <https://codeinstitute.net/global/blog/what-is-php-programming/>

ДОДАТКИ

Додаток А

Структура однієї із сторінок веб-сайту

```
<body><h1 class = "h1" > Ласкаво просимо до СОТ «Орлятко»</h1>
<header><div class = 'switcher' style="text-align: right;"><form
autocomplete="off"><select id="lang_switcher"><option disabled selected
hidden="true">Вибрати мову</option><option
value="https://www.orliatko.com.ua/default.html">Українська</option><option
value="https://orliatko.com.ua/ru/ru_start.html">English</option><script>
document.getElementById("lang_switcher").onchange = function() {
if (this.selectedIndex!==0) { window.location.href = this.value; } };
</script></select></form></div><nav><ul><div id = 'logo'><a
href="default.html"></a></div><div class="hoverChange"><div
class='header_innerHeaderDivs'><li><a href="dytiachiy.html">Дитячий
відпочинок </a></li><li><a href="posluhy.html">Інші послуги</a></li><li><a
href="kontakty.html">Наші контакти</a></li><li><a href="
index.php">Спільнота «Орлятко»
</a></li></div></div></ul></nav></header><div class="col"><h2 class = "h2">
СОТ «Орлятко» запрошує на відпочинок</h2><h3 class = "h3"><mark>Не
знаєш, як провести літо?</mark> Тоді приїжджай до нас! У нашому таборі на
тебе чекають нові друзі, цікава програма, яскравий вир подій! Табір
«Орлятко» – це канікули у стилі "Active Life". Наша програма розроблена
```

так, щоб кожен міг проявити себе і розкрити свої здібності. Давай разом зробимо це літо незабутнім! </br>Подивитись

розклад оздоровчих змін і вартість путівок<div class = 'vid'><iframe width="280" height="157"

src="https://www.youtube.com/embed/SVZyTcxSiOQ?&hd=1" frameborder="0"

allow="accelerometer; autoplay; clipboard-write; encrypted-media; gyroscope;

picture-in-picture" allowfullscreen></iframe></div>

<mark>Розташування.</mark> Табір «Орлятко» займає площу 3 га і

знаходиться у мальовничому, екологічно чистому, сприятливому для

дитячого відпочинку та оздоровлення місці, на території рекреаційної зони

міста Суми (урочище Баранівка), у сосновому лісі поряд з річкою Псел.

</br></br><mark>Безпека.</mark> Територія табору огорожена з усіх

сторін, цілодобово ведеться відеоспостереження. Табір знаходиться під

охороною поліції. Діти відпочивають у групах, сформованих за віком та

інтересами. За кожним загоном закріплено два або три вожатих, в залежності

від віку підопічних. Вожаті цілодобово несуть відповідальність за життя і

здоров'я дітей. Купання дітей у річці або на озері відбувається організовано,

під наглядом інструктора з плавання та з дотриманням усіх необхідних

заходів безпеки. З огляду на активну програму табору, для дітей постійно

проводяться інструктажі з безпеки життєдіяльності.</h3></div></br><div

class="col2" style="text-align: center;"></br></div><div class="col"><h3 class = "h3"><mark>Медичне обслуговування.</mark>На території табору є медпункт з усіма необхідними медпрепаратами. За здоров'ям дітей цілодобово стежить кваліфікований лікар. Про всі зміни в стані здоров'я дитини батьки інформуються негайно.

</br></br><mark>Інфраструктура табору:</mark></br>

- три житлові корпуси;</br>
- їдальня;</br>
- актовий зал;</br>
- літня естрада;</br>
- баскетбольний майданчик;</br>
- два волейбольних майданчики;</br>
- критий майданчик для настільного тенісу;</br>
- майданчик зі штучним покриттям для футзалу (40 * 20 м);</br>
- футбольне ґрунтове поле (60 * 40 м);</br>
- футбольна коробка з ґрунтовим покриттям (15 * 8 м);</br>
- мотузковий парк;</br>
- наметове містечко;</br>
- доріжка здоров'я;</br>
- альтанки, місця для загонів;</br>
- скеледром;</br>
- столи для "Head-ball";</br>
- сауна;</br>
- пляж табору на березі річки Псел.</h3>

</div><div class = "col3" style="text-align: center;"></br>


```


</div><div class="col"
style="padding: 0px .5%;"><h4 class = "h4">«Можливо, для когось ти просто
дитина.<br>Для нас ти – цілий світ!»</h4></div><div id="myModal"
class="modal"><span class="close">&times;</span><img class="modal-content"
id="img01"><div id="caption"></div></div> <script src="js/main1.js"></script>
<footer style="text-align: center;">Спортивно-оздоровчий табір «Орлятко»
2022</footer></body>

```

Додаток Б

У даному додатку наведений код стилів веб-сайту

```
body{
  margin: 0;
  background: #dfebe2;
}
#header_navigation{
  padding-right:15px;
  padding-top:3px;
  color:#03420d;
  font-size:20px;
}
header{
  color: white;
  padding: 15px 15px 0px 15px;
  border-radius: 5px;
  border: 0.5px solid black;
  background-image: url(../img/back10.JPG);
  background-size:100%;
  background-position: center;
background-repeat: no-repeat;
-moz-background-size: cover;
-o-background-size: cover;
background-size: cover;
}
nav ul{
  display: inline;
  margin:0;
```



```
padding: 0 0 0 0px;
}
nav ul li{
    background: black;
    color: white;
    display: inline-block;
    list-style-type: none;
    padding: 7px 10px;
    margin: 0;
    border: 1px solid #acc6e6;
    border-top-left-radius: 5px;
    border-top-right-radius: 5px;
    border-bottom-left-radius: 3px;
    border-bottom-right-radius: 3px;
    font-size: 20px;
    position: relative;
    left: -4px;
    top: 0px;

}
nav ul li a{
text-decoration: none;
font-family: Georgia, serif;
}
.vid{
margin: auto;
width: auto;
position: relative;
padding-bottom: 56.25%; /* 16:9 */
```

```
    height: 0;
  }
.vid iframe{
  position: absolute;
  top: 10%;
  left: 10%;
  width: 80%;
  height: 80%;
}
.hoverChange :hover{
  border-radius: 90%;
  -webkit-transition: all 500ms ease;
  -moz-transition: all 500ms ease;
  -ms-transition: all 500ms ease;
  -o-transition: all 500ms ease;
  transition: all 500ms ease;
}
.h1{
  font-size: 20px;
  font-family: Georgia, serif;
  padding: 0 0 0 0px;
  margin: 0;
  padding-right:15px;
  padding-top:3px;
  color:#03420d;
}
.h2{
  margin: 10px 0;
  font-family: Georgia, serif;
```

```
    text-align:center;
    font-size: 30px;
    color: #e5f50f;
    text-shadow: 3px 0 black;
}
.h3{
font-size: 20px;
font-family: Georgia, serif;
text-align: left;
text-indent: 40px;
line-height: 1.3;
letter-spacing: 0.3px;
}
.h4{
font-size: 20px;
font-family: Georgia, serif;
text-align:center;
}
mark {
background-color: #025437;
color: pink;
}
.col{
float: left;
border-radius: 5px;
color: white;
padding: 10px .5%;
background: #025437;
width: 98%;
```

```
        margin: 0.5%;
    }
    #myImg {
        border-radius: 5px;
        cursor: pointer;
        transition: 0.3s;
    }
    #myImg:hover {opacity: 0.7;}
    .modal {
        display: none
        position: fixed
        z-index: 1;
        padding-top: 100px;
        left: 0;
        top: 0;
        width: 100%;
        height: 100%;
        overflow: auto;
        background-color: rgb(0,0,0);
        background-color: rgba(0,0,0,0.9);
    }
    .modal-content {
        margin: auto;
        display: block;
        width: 80%;
        max-width: 700px;
    }
    #caption {
        margin: auto;
```

```
display: block;
width: 80%;
max-width: 700px;
text-align: center;
color: #ccc;
padding: 10px 0;
  height: 150px;
}
.modal-content, #caption {
  animation-name: zoom;
  animation-duration: 0.6s;
}
@keyframes zoom {
  from {transform:scale(0)}
  to {transform:scale(1)}
}
.close {
  position: absolute;
  top: 15px;
  right: 35px;
  color: #f1f1f1;
  font-size: 40px;
  font-weight: bold;
  transition: 0.3s;
}
.close:hover,
.close:focus {
  color: #bbb;
  text-decoration: none;
```

```
    cursor: pointer;
}
@media only screen and (max-width: 700px){
  .modal-content {
    width: 100%;
  }
}
```

Додаток В

Код асинхронного обміну даними для перевірки унікальності логіна,
введеного під час реєстрації

```
echo <<<<_END
<script>
function checkUser(user)
{
if (user.value == "")
{
$('#used').html('&nbsp;');
return
}
$.post
(
'checkuser.php',
{ user : user.value },
function(data)
{
$('#used').html(data)
}
)
}
</script>
_END;
```

Додаток Г

Код файлу входу до спільноти

```
<?php // login.php
require_once 'header.php';
$error = $user = $pass = "";
if (isset($_POST['user'])) {
    $user = sanitizeString($_POST['user']);
    $pass = sanitizeString($_POST['pass']);
    if ($user == "" || $pass == "")
        $error = 'Не всі поля були заповнені. Спробуйте ще раз.';
    else {
        $result = queryMySQL("SELECT name FROM users
            WHERE name='$user'");
        if ($result->num_rows == 0) {
            $error = "Неправильний логін. Спробуйте ще раз.";
        } else {
            $query = queryMySQL("SELECT password FROM users WHERE
                name='$user'");
            $hash = mysqli_fetch_assoc($query);
            $hashstring = $hash['password'];
            if (!password_verify($pass, $hashstring)) {
                $error = "Неправильний пароль. Спробуйте ще раз.";
            } else {
                $_SESSION['user'] = $user;
                $_SESSION['pass'] = $pass;
                header("Location: members.php?view=$user");
                exit(); } } } }
```



```
echo <<<<_END
```

```
<form method='post' action='login.php'>
```

```
<div data-role='fieldcontain'>
```

```
<label></label>
```

```
<span class='error'>$error</span>
```

```
</div>
```

```
<div data-role='fieldcontain'>
```

```
<label></label>
```

```
Будь-ласка, введіть логін та пароль
```

```
</div>
```

```
<div data-role='fieldcontain'>
```

```
<label>Логін</label>
```

```
<input type='text' maxlength='32' name='user' value='$user'>
```

```
</div>
```

```
<div data-role='fieldcontain'>
```

```
<label>Пароль</label>
```

```
<input type='password' maxlength='16' name='pass' value='$pass'>
```

```
<div class='link'>
```

```
<a data-transition='slide' id = 'link' href='mailcheck.php'>Забули  
пароль?</a>
```

```
</div>
```

```
</div>
```

```
<div data-role='fieldcontain'>
```

```
<label></label>
```

```
<input data-transition='slide' class = 'login1' type='submit'  
value='Увійти' >
```

```
</div>
```

```
<script>
```

```
$('.login1').css ('background-color','red')
$('.login1').css ('text-shadow','0px 0 black')
$('.login1').css ('font-family', 'Georgia, serif')
$('#link').css ('font-size','small')
$('#link').css ('background-color','#dfebe2')
$('a').css ('float','center')
</script> </form></div> </body></html>
_END;
?>
```