

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ**  
**Сумський державний університет**  
Факультет електроніки та інформаційних технологій  
Кафедра комп'ютерних наук

«До захисту допущено»  
В.о. завідувача кафедри

\_\_\_\_\_ Ігор ШЕЛЕХОВ  
(підпис)

19 грудня 2023 р.

---

**КВАЛІФІКАЦІЙНА РОБОТА**  
**на здобуття освітнього ступеня магістр**

зі спеціальності 122 - Комп'ютерних наук,  
освітньо-професійної програми «Інформатика»  
на тему: «Інформаційно-комунікаційна технологія забезпечення соціальної  
комунікації навчального центру закладу вищої освіти»  
здобувача групи ІН.м-23 Карпенка Дениса Володимировича

Кваліфікаційна робота містить результати власних досліджень.  
Використання ідей, результатів і текстів інших авторів мають посилання на  
відповідне джерело.

Денис КАРПЕНКО

\_\_\_\_\_  
(підпис)

Керівник,  
старший викладач,  
кандидат фізико-математичних наук,

Дмитро ВЕЛИКОДНИЙ

\_\_\_\_\_  
(підпис)

**Сумський державний університет**  
Факультет електроніки та інформаційних технологій  
Кафедра комп'ютерних наук

«Затверджую»

В.о. завідувача кафедри

Ігор ШЕЛЕХОВ

\_\_\_\_\_  
(підпис)

## ЗАВДАННЯ НА КВАЛІФІКАЦІЙНУ РОБОТУ

### на здобуття освітнього ступеня магістр

зі спеціальності 122 - Комп'ютерних наук, освітньо-професійної програми «Інформатика»

здобувача групи ІН.м-23 Карпенка Дениса Володимировича

1. Тема роботи: «Інформаційно-комунікаційна технологія забезпечення соціальної комунікації навчального центру закладу вищої освіти»

затверджую наказом по СумДУ від «15» грудня 2023 р. №

2. Термін здачі здобувачем кваліфікаційної роботи до 21 грудня 2023 року

3. Вхідні дані до кваліфікаційної роботи Освітні веб-сайти стають все більше необхідним інструментом в системі освіти. З урахуванням технологічного розвитку та змін у способах навчання, дослідження їх ролі та впливу стає дуже актуальним.

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

1) Аналіз проблеми предметної області, постановка й формування завдань дослідження.

2) Огляд технологій, що використовуються для розробки веб-сайтів. 3) Розробка

технології забезпечення соціальної комунікації навчального центру закладу вищої освіти.

4) Аналіз результатів.

5. Перелік графічного матеріалу.

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

Розділ	Консультант	Підпис, дата	
		Завдання видав	Завдання прийняв

7. Дата видачі завдання « \_\_\_\_ » \_\_\_\_\_ 20 \_\_\_\_ р.

Завдання прийняв до виконання \_\_\_\_\_

(підпис)

Керівник \_\_\_\_\_

(підпис)

## КАЛЕНДАРНИЙ ПЛАН

п/п	Назва етапів кваліфікаційної роботи	Термін виконання	Примітка
	<i>Аналіз проблеми предметної області, постановка й формування завдань дослідження</i>		
	<i>Огляд технологій, що використовуються для створення веб-сайтів</i>		
	<i>Розробка технології забезпечення соціальної комунікації навчального центру закладу вищої освіти</i>		
	<i>Аналіз отриманих результатів</i>		
	<i>Оформлення пояснювальної записки до кваліфікаційної роботи</i>		

Здобувач вищої освіти

\_\_\_\_\_

(підпис)

Керівник

\_\_\_\_\_

(підпис)

## АНОТАЦІЯ

**Записка:** 54 стор., 25 рис., 2 додатки, 18 джерел.

**Обґрунтування актуальності теми роботи** – Тема кваліфікаційної роботи є актуальною, оскільки присвячена розв’язанню важливої практичної задачі забезпечення соціальної комунікації навчального центру закладу вищої освіти.

**Об’єкт дослідження** — інформаційний портал навчального центру закладу вищої освіти.

**Мета роботи** — розробка технології забезпечення соціальної комунікації навчального центру закладу вищої освіти.

**Методи дослідження** — Вивчення технічних вимог, складання структури та контент-плану, вибір технічних рішень, дизайн, розробка та тестування.

**Результати** — розроблено веб-сайт, який спрямовано на роботу з учнями в галузі ІТ та є платформою для навчання та обміну знаннями.

ОСВІТНІЙ ВЕБ-САЙТ, HTML, JAVASCRIPT, PHP, CSS, ФОРМА  
РЕЄСТРАЦІЇ, ДИЗАЙН.

## ЗМІСТ

ВСТУП .....	6
1 ТЕОРЕТИЧНІ ВІДОМОСТІ .....	7
1.1 Ключові компоненти .....	7
1.2 Приклади використання .....	8
1.3 Освітні веб-сайти .....	12
1.4 Постановка задачі .....	15
2 ВИБІР МЕТОДУ РІШЕННЯ .....	17
2.1 Актуальність соціальної комунікації з використанням веб-сайтів....	17
2.1 На що варто звернути увагу у кодї? .....	18
2.2 Покрокова інструкція з використання освітнього веб-сайту. ....	19
2.3 Опис мов програмування для створення освітнього веб-сайту. ....	21
3 ІНФОРМАЦІЙНЕ ТА ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ СИСТЕМИ ..	30
3.1 Розробка веб-сайту.....	30
3.2 Тестування веб-сайту навчального центру #Well.com.IT .....	39
ВИСНОВКИ.....	42
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	43
ДОДАТОК А.....	45
ДОДАТОК Б .....	54

## ВСТУП

**Актуальність.** Дослідження освітніх веб-сайтів у сучасному світі важлива через стрімкий розвиток технологій та постійне зростання інформаційного потоку. Запит на доступ до якісної освіти в Інтернеті неабияк збільшується, а користувачі віддають перевагу веб-ресурсам для навчання та розвитку. Сучасні умови вимагають новаторських рішень у сфері освіти, які б забезпечували якість та доступність навчання. Це створює простір для дослідження та впровадження нових методів і підходів у сфері освітніх веб-сайтів.

**Об'єкт дослідження.** Інформаційний портал навчального центру закладу вищої освіти.

**Предмет дослідження.** Функціонал, ефективність та тенденції розвитку освітніх веб-сайтів як інструментів для навчання та саморозвитку.

**Гіпотеза.** Дослідження полягає у тому, що успішне використання освітніх веб-сайтів залежить від їхнього змісту, доступності, інтерактивності та здатності відповідати на потреби користувачів.

**Наукова новизна.** Аналіз актуальних підходів до створення та використання освітніх веб-сайтів, врахуванні індивідуальних потреб користувачів та розробці рекомендацій для оптимізації процесів навчання.

**Структура.** Робота включає вступ, аналіз сучасних тенденцій у розвитку освітніх веб-сайтів, формулювання завдань дослідження, вибір методології та інструментів дослідження, аналіз отриманих результатів, висновки та рекомендації, список використаних джерел і додатки.

Дана робота спрямована на виявлення ключових аспектів, що визначають успішність освітніх веб-сайтів та їхній вплив на процеси навчання, щоб сприяти подальшому розвитку та удосконаленню цього важливого напрямку освіти.

# 1 ТЕОРЕТИЧНІ ВІДОМОСТІ

## 1.1 Ключові компоненти

1. **Веб-сторінки:** це документи, які можуть містити текст, зображення, відео та інші мультимедійні елементи. Веб-сторінки написані такими мовами, як HTML (мова розмітки гіпертексту), і відображаються веб-браузерами.
2. **Доменне ім'я:** це зрозуміла адреса, яка використовується для доступу до веб-сайту. Наприклад, «www.example.ua» – це доменне ім'я.
3. **Веб-сервер:** це комп'ютер, який розміщує веб-сайт і доставляє його вміст користувачам через Інтернет. Він відповідає на запити веб-браузерів і надсилає запитані веб-сторінки.
4. **URL (уніфікований покажчик ресурсів):** це конкретна адреса, яка використовується для пошуку ресурсів в Інтернеті. Він містить протокол (наприклад, "http" або "https"), ім'я домену та шлях до певного ресурсу.
5. **Структура навігації:** веб-сайти часто мають ієрархічну структуру з меню, посиланнями та іншими елементами навігації, які допомагають користувачам переходити між різними сторінками та розділами.
6. **Система керування вмістом (CMS):** деякі веб-сайти використовують CMS для полегшення створення та керування цифровим вмістом. Популярні платформи CMS включають WordPress, Joomla та Drupal.

**Односторінковий сайт** – це невелика сторінка обсягом до 15 розворотів із єдиною метою – збільшення конверсії. Це сайт з обмеженим контентом, без полотен текстів, без навантаження фото та відео, без великої кількості банерів. Але із СТА (call to action, заклик до дії). Завдання створення односторінкового сайту – привернути увагу відвідувача, утримати його та привести до певної дії. Їм може бути купівля, реєстрація, завантаження чи вказівка особистих даних для зв'язку.

## **Кому підходить односторінковий сайт?**

Для будь-яких форм бізнесу, де ціль - продати товар/послугу/інформацію тут і зараз. Це може бути товар, який закуплено у великому обсязі, але його реалізація стандартними методами не виправдала очікувань; новинка на ринку; акція; захід; продаж курсів тощо. Якщо потрібно в короткі терміни продати щонайбільше, односторінковий сайт - те, що потрібно[1].

### **1.2 Приклади використання**

Використання веб-сайту залежить від його призначення та змісту. Ось кілька поширених типів веб-сайтів і їх типове використання:

- Інформаційні веб-сайти:
  - Мета: Надати інформацію про певну тему, організацію чи тему.
  - Приклади: Вікіпедія, освітні веб-сайти, сторінки з інформацією про компанії.
- Веб-сайти електронної комерції:
  - Мета: полегшити онлайн-купівлю та продаж товарів або послуг.
  - Приклади: Amazon, eBay, роздрібні онлайн-магазини.
- Веб-сайти соціальних мереж:
  - Мета: З'єднуватися та взаємодіяти з іншими, ділитися вмістом і брати участь в онлайн-спільнотах.
  - Приклади: Facebook, Twitter, Instagram, LinkedIn.
- Блог або персональні веб-сайти:
  - Мета: поділитися особистими думками, досвідом або знаннями з певної теми.
  - Приклади: особисті блоги, професійні портфоліо.
- Веб-сайти новин і ЗМІ:
  - Мета: Надавати актуальні новини, статті та мультимедійний вміст.
  - Приклади: Уніан, ТСН, і т.д.
- Освітні веб-сайти:
  - Мета: Пропонувати освітні ресурси, курси чи навчальні посібники.



- Приклади: Coursera, веб-сайти навчальних закладів.
  - Державні веб-сайти:
- Призначення: Надавати офіційну інформацію та послуги, пов'язані з діяльністю органів влади.
- Приклади: офіційні урядові веб-сайти, веб-сайти агентств.
  - Корпоративні або бізнес-сайти:
- Мета: продемонструвати продукти, послуги та інформацію про компанію.
- Приклади: веб-сайти компаній, бізнес-цільові сторінки.
  - Веб-сайти портфоліо:
- Мета: продемонструвати роботу, проекти чи досягнення особи чи організації.
- Приклади: Портфоліо дизайну, портфоліо художника.
  - Форуми та веб-сайти спільнот:
- Мета: сприяти обговоренню та взаємодії серед спільноти користувачів.
- Приклади: Reddit, Stack Overflow, спеціальні форуми.

### Приклади сайтів навчальних центрів

Навчальні центри ІТ в Україні, такі як ITstep, ITacademy та GoIT, мають важливі веб-сайти, що використовуються для представлення своїх курсів та послуг. Давайте проведемо порівняльний аналіз їх веб-сайтів та сформулюємо загальні вимоги до сайту навчального центру в галузі ІТ:

#### 1. ITstep (<https://itstep.org/>) (рис.1.1):

- **Дизайн:** Сучасний, з акцентом на інформаційність та візуальну привабливість.
- **Навігація:** Зручна навігація, швидкий доступ до інформації про курси, вакансії та контакти.
- **Інтерактивність:** Є можливість реєстрації на курси прямо на сайті[2].



Рисунок 1.1 – Початкова сторінка сайту itstep.org

## 2. ITEAcademy (<https://itea.ua/>) (рис.1.2):

- **Дизайн:** Мінімалістичний та стильний дизайн, фокус на лаконічності та зручності.
- **Курси:** Чітко визначені категорії курсів з короткими описами та можливістю перегляду плану навчання.
- **Блог:** Наявність блогу для публікацій та оновлень[3].



Рисунок 1.2 – Початкова сторінка сайту itea.ua

### 3. GoIT (<https://goit.ua/>) (рис.1.3):

- **Адаптивність:** Мобільна версія сайту для зручності користувачів на різних пристроях.
- **Проекти та вакансії:** Розділи для демонстрації проектів учасників та публікацій вакансій для випускників.
- **Відгуки:** Секція із відгуками випускників, яка додає довіру та переконливість[4].

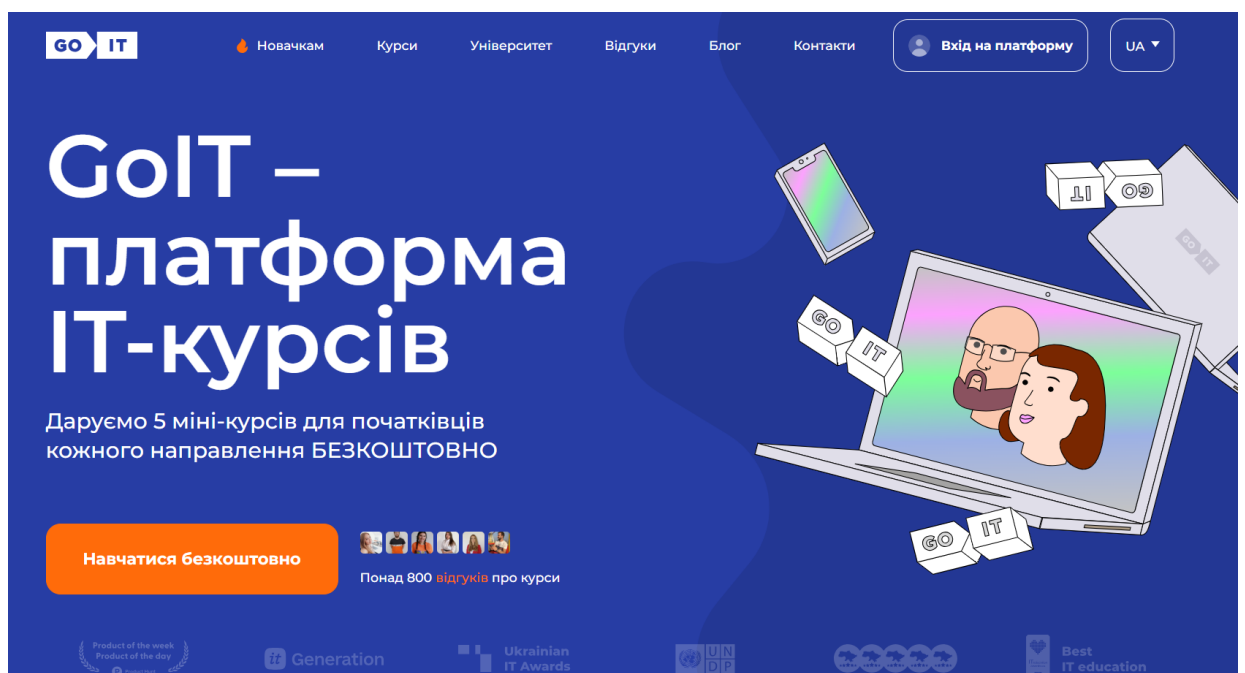


Рисунок 1.3 – Початкова сторінка сайту goit.ua

Загальні вимоги до сайту навчального центру в галузі ІТ можуть включати:

1. **Чітка структура:** Легка навігація, яка дозволяє відвідувачам швидко знаходити інформацію про курси, програми, викладачів тощо.
2. **Інформативність:** Достатня кількість інформації про кожен курс, його тривалість, програму та можливості для учнів.
3. **Контактна інформація:** Легкий доступ до контактів центру, форма зворотного зв'язку, можливість швидкого з'єднання.
4. **Інтерактивність:** Взаємодія з відвідувачами, можливість реєстрації та відстеження статусу навчання.
5. **Дизайн та візуалізація:** Привабливий дизайн, використання візуальних елементів, які відображають професійність та інноваційність.
6. **Адаптивність:** Оптимізація для мобільних пристроїв, щоб користувачі могли зручно переглядати інформацію на будь-якому пристрої.

На основі цього аналізу сформувався вимоги до сайту[5].

### 1.3 Освітні веб-сайти

Освітні веб-сайти охоплюють широкий спектр тем і служать різним цілям. Вони призначені для надання освітніх ресурсів, курсів, навчальних посібників та інформації з різноманітних тем.

Категорії освітніх веб-сайтів: онлайн-курси та посібники, вивчення мови, кодування та програмування, математика та наука, мистецтво і творчість, здоров'я та благополуччя, історія та гуманітарні науки, підготовка до екзаменів, і т.д.

Це лише кілька прикладів, і є багато інших освітніх веб-сайтів, присвячених конкретним предметам, рівням знань і стилям навчання. Освітні веб-сайти відіграють вирішальну роль у тому, щоб зробити

навчання доступним, інтерактивним і часто безкоштовним або доступним[6].

Створення освітнього веб-сайту вимагає розуміння педагогічних та технічних аспектів для забезпечення ефективного навчання та зручного користування. Нижче наведено деякі основні теоретичні аспекти, які можуть бути використані при розробці освітнього веб-сайту:

## **1. Педагогічні аспекти:**

### **➤ Педагогічні цілі:**

- Визначте, які конкретні навчальні цілі ви хочете досягти за допомогою веб-сайту. Розгляньте створення навчальних матеріалів, тестів, взаємодії студентів тощо.

### **➤ Дизайн навчального контенту:**

- Розробіть зміст, який відповідає вашим педагогічним цілям. Використовуйте різні типи навчальних матеріалів, такі як тексти, зображення, відео, анімації тощо.

### **➤ Взаємодія та залучення:**

- Спроектуйте функціонал для взаємодії студентів з матеріалами. Включіть завдання, тести, форуми для обговорення та інші інтерактивні елементи для залучення.

### **➤ Адаптивність та різноманітність:**

- Забезпечте можливість доступу до навчального контенту з різних пристроїв. Розгляньте різноманітність методів навчання для задоволення потреб різних студентів.

### **➤ Оцінювання та звітність:**

- Розробіть систему оцінювання, яка відповідає навчальним цілям. Забезпечте можливість студентам переглядати свої досягнення.

## **2. Технічні аспекти:**

### **➤ Вибір технічних платформ:**

- Визначте технічні платформи, на яких буде заснований ваш веб-сайт (наприклад, WordPress, Moodle, власний веб-додаток).

➤ **Безпека та керування даними:**

- Забезпечте безпеку персональних даних студентів та викладачів. Використовуйте шифрування, захист від SQL-ін'єкцій та інші технічні заходи безпеки.

➤ **Адаптивний дизайн:**

- Забезпечте адаптивний дизайн для оптимального відображення на різних пристроях та розмірах екранів.

➤ **Аналітика та відстеження:**

- Встановіть аналітичні інструменти для відстеження відвідуваності сайту та взаємодії користувачів з контентом.

➤ **Підтримка та оновлення:**

- Розробіть план технічної підтримки та регулярних оновлень, щоб забезпечити безперебійну роботу сайту.

### 3. Доступність та регулятивні вимоги:

➤ **Доступність:**

- Забезпечте, щоб ваш веб-сайт був доступним для всіх користувачів, включаючи тих, які використовують технічні засоби для отримання інформації.

➤ **Відповідність до законодавства:**

- Дотримуйтеся законодавчих вимог, таких як GDPR, щоб забезпечити правильну обробку персональних даних.

➤ **Ліцензування та авторське право:**

- З'ясуйте питання ліцензування контенту та авторських прав на навчальні матеріали.

Ці аспекти представляють собою загальний огляд основних педагогічних та технічних вимог при розробці освітнього веб-сайту.

Реалізація такого сайту вимагає збалансованого підходу до обох цих аспектів.[7]

### **1.4 Постановка задачі**

Метою даної роботи є створення веб-сайту для навчального центру з метою поліпшення процесів навчання, забезпечення доступу до інформації та сприяння ефективному взаємодії між студентами, викладачами та адміністрацією. Розробка веб-сайту виконується в рамках кваліфікаційної роботи магістра за спеціальністю 122 «Комп'ютерні науки».

Створений веб-сайт повинен відповідати наступним вимогам:

#### **1. Інформаційна частина:**

- Представлення курсів, програм та розкладу занять.
- Опис викладачів та їхніх кваліфікацій.
- Новини та оголошення навчального центру.

#### **2. Функціонал для студентів:**

- Реєстрація користувачів.
- Доступ до навчальних матеріалів та завдань.
- Контактна інформація для звертання до викладачів та адміністрації.

#### **3. Функціонал для викладачів:**

- Можливість завантаження та оновлення навчальних матеріалів.
- Взаємодія зі студентами через електронні засоби комунікації.

#### **4. Технічна частина:**

- Вибір технологій для розробки веб-сайту.
- Розробка зручного та інтуїтивно зрозумілого інтерфейсу.
- Забезпечення безпеки та конфіденційності даних користувачів.

Для досягнення поставленої мети необхідно вирішити наступні задачі:

1. Аналіз вимог: Визначення потреб користувачів та функціональних вимог до веб-сайту.

2. Проектування: Розробка структури веб-сайту, визначення бази даних, вибір технічних засобів та інструментів для розробки.
3. Розробка: Створення основних функціональних блоків веб-сайту та розробка серверної та клієнтської частин.
4. Тестування: Перевірка правильності роботи функціоналу та виявлення можливих помилок.



## 2 ВИБІР МЕТОДУ РІШЕННЯ

### 2.1 Актуальність соціальної комунікації з використанням веб-сайтів

Університети часто мають освітні веб-сайти з різних причин, і ці сайти відіграють вирішальну роль у підтримці місії закладу та в обслуговуванні потреб студентів, викладачів, персоналу та широкої спільноти.[8] Причини, чому університетам потрібні освітні веб-сайти вказані у таблиці 1.1:

Таблиця 1.1 Призначення освітнього веб-сайту

Призначення	Приклади
Поширення інформації	- Інформація про курси: програми, розклад, матеріали
	- Деталі програми: вимоги до вступу, програми
Онлайн навчання	- Проведення курсу: онлайн-курси, навчальні ресурси
	- Системи управління навчанням (LMS): Матеріали курсу
Дослідження та публікації	- Профілі факультету: дослідження, публікації, експертиза
	- Наукові центри: проекти, публікації
Адміністративні процеси	- Прийом: процес, форми, терміни
	- Реєстрація та зарахування: навчальні процеси
Ресурси та послуги кампусу	- Ресурси бібліотеки: каталоги, бази даних
	- Студентські послуги: консультації, кар'єрні послуги
Залучення спільноти	- Події та оголошення: оновлення університету, події

	- Залучення випускників: нетворкінг, заходи
<b>Спілкування та співпраця</b>	- Внутрішня комунікація: комунікація між викладачами та персоналом
	- Новини та оновлення: новини університету, досягнення
<b>Доступність та інклюзивність</b>	- Онлайн доступність: доступна інформація
	- Інклюзивне навчання: задоволення різноманітних потреб

Загалом освітній веб-сайт підвищує ефективність, доступність і прозорість діяльності університету, сприяючи створенню позитивного навчального та дослідницького середовища для всієї університетської спільноти [9].

## **2.1 На що варто звернути увагу у коді?**

1) Посилання, куди ви передаєте дані, для того щоб користувач міг без проблемний доступ до реєстрації на курси.

2) Обов'язкові параметри, які необхідні для передачі даних та виклику тих чи інших дій. Відповідно, і варіанти відповідей, які ви можете отримувати під час надсилання цих даних та запитів (рис.2.1).

```

sender.php
1  <?php
2      $name = $_POST['name'];
3      $surname = $_POST['surname'];
4      $phone = $_POST['phone'];
5      $email = $_POST['email'];
6      $text = $_POST['text'];
7
8      $to = "@gmail.com";
9      $date = date ("d.m.Y");
10     $time = date ("h:i");
11     $from = $email;
12     $subject = "Заявка с сайта";
13
14
15     $msg="
16     Ім'я: $name /n
17     Прізвище: $surname /n
18     Телефон: $phone /n
19     Пошта: $email /n
20     Текст: $text";
21     mail($to, $subject, $msg, "From: $from ");
22

```

Рисунок 2.1 - Код для створення форми реєстрації

- 3) Вільний доступ для користувачів, які використовують веб-сайт через телефон.
- 4) Справне показування цифрової інформації та зручність користування.
- 5) Доступний вигляд для розробника.

## 2.2 Покрокова інструкція з використання освітнього веб-сайту.

Покрокова інструкція з використання освітнього веб-сайту є важливою складовою роботи та наведена у таблиці 2.1.

Таблиця 2.1 Інструкція з використання сайту

Крок	Опис
1. Навігація	Вивчіть структуру веб-сайту та

	навігацію.
<b>2. Обліковий запис користувача</b>	Створіть обліковий запис користувача для персоналізованих функцій і відстеження прогресу.
<b>3. Перегляньте курси або вміст</b>	Ознайомтеся з доступними курсами, перегляньте описи курсів і отримайте доступ до матеріалів.
<b>4. Зарахування</b>	Запишіться на курси, пройшовши реєстрацію або оплату, якщо потрібно.
<b>5. Навчальні матеріали</b>	Отримайте доступ до матеріалів курсу, включаючи лекції, конспекти, тести та завдання.
<b>6. Спілкуйтеся з інструкторами та колегами</b>	Спілкуйтеся з інструкторами та колегами через форуми, обмін повідомленнями або чат.
<b>7. Виконайте завдання та оцінки</b>	Беріть участь у завданнях, вікторинах або іспитах, отримуючи відгуки за потреби.
<b>8. Відстежуйте прогрес</b>	Використовуйте функції відстеження або прогресу, щоб контролювати свій прогрес на курсах.
<b>9. Сертифікати та досягнення</b>	Пройдіть курси, щоб отримати сертифікати або досягнення, якщо пропонуються.
<b>10. Використовуйте додаткові ресурси</b>	Досліджуйте форуми, блоги та додаткові матеріали для глибшого розуміння.
<b>11. Надайте відгук</b>	Поділіться думками та відгуками про курси та матеріали, якщо це дозволяє веб-сайт.

<b>12. Будьте в курсі</b>	Регулярно перевіряйте наявність оновлень і оголошень, щоб бути в курсі.
<b>13. Підключайтеся до спільноти</b>	Приєднуйтеся до форумів спільноти або груп у соціальних мережах, пов'язаних із веб-сайтом.
<b>14. Використовуйте служби підтримки</b>	За потреби зверніться за технічною підтримкою, консультацією чи додатковою допомогою.
<b>15. Досліджуйте можливості кар'єри</b>	Перевірте інформацію про кар'єрні шляхи або варіанти подальшої освіти, якщо такі є.

### 2.3 Опис мов програмування для створення освітнього веб-сайту.

Вибір мов програмування для створення освітнього веб-сайту залежить від різних факторів, у тому числі вимог до веб-сайту та функцій, які ви хочете реалізувати.

- HTML, CSS, JavaScript
- PHP
- Python
- C#
- SQL

і навіть більше.[10]

#### HTML, CSS, JavaScript

HTML, CSS і JavaScript є основними технологіями, які використовуються у веб-розробці. Вони працюють разом, щоб створювати динамічні та інтерактивні веб-сторінки. Ось огляд кожного:

## HTML (мова розмітки гіпертексту):

- **Призначення:** HTML є стандартною мовою розмітки для створення структури веб-сторінки. Він визначає такі елементи на сторінці, як заголовки, абзаци, зображення, посилання, форми тощо.
- **Приклад синтаксису (рис.2.2):**

```
<!DOCTYPE html>
<html>
  <head>
    <title>My Web Page</title>
  </head>
  <body>
    <h1>Hello, World!</h1>
    <p>This is a sample web page.</p>
  </body>
</html>
```

Рисунок 2.2 – Приклад синтаксису HTML

## CSS (каскадні таблиці стилів):

- **Призначення:** CSS використовується для стилізації та форматування вмісту HTML. Він керує макетом, зовнішнім виглядом і візуальним представленням веб-сторінок. CSS дозволяє розробникам застосовувати такі стилі, як кольори, шрифти, поля та позиціонування [11].
- **Приклад синтаксису (рис.2.3):**

```
body {  
  font-family: Arial, sans-serif;  
  background-color: #f0f0f0;  
}  
  
h1 {  
  color: #333;  
}  
  
p {  
  margin-bottom: 20px;  
}
```

Рисунок 2.3 – Приклад синтаксису CSS

### JavaScript:

- **Призначення:** JavaScript — це мова сценаріїв, яка забезпечує динамічний вміст та інтерактивність на стороні клієнта. Це дозволяє розробникам маніпулювати HTML і CSS сторінки, обробляти взаємодію користувачів і виконувати асинхронні операції [12].
- **Приклад синтаксису (рис.2.4):**

```
// Using const and arrow functions  
const greeting = (name) => `Hello, ${name}!`;  
  
// Template literals and let  
let person = "John";  
console.log(greeting(person));
```

Рисунок 2.4 – Приклад синтаксису JavaScript

Разом HTML, CSS і JavaScript утворюють основні технології для створення інтерактивних і візуально привабливих веб-сайтів. HTML

забезпечує структуру, CSS стилізує презентацію, а JavaScript додає поведінку та інтерактивність, щоб створити зручну взаємодію з користувачем. Цю комбінацію часто називають «переднім» стеком у веб-розробці.

## **PHP**

PHP в основному використовується для створення сценаріїв на стороні сервера, тобто він працює на веб-сервері, а не в браузері клієнта. Він обробляє дані на сервері та надсилає результат клієнту.

- Ви можете створювати безпечні, масштабовані та надійні програми.
- Його безкоштовний та відкритий вихідний код.

Код PHP вбудовано безпосередньо в HTML, що дозволяє легко змішувати динамічний вміст зі статичним вмістом на веб-сторінці. PHP-скрипти зазвичай вкладено в `<?php ... ?>` теги. PHP (Hypertext Preprocessor) - це серверна мова програмування загального використання, яка широко використовується для розробки веб-застосунків. Вона дозволяє вбудовувати код PHP безпосередньо в HTML-сторінки, що робить їх динамічними.

Ось огляд основних можливостей PHP:

### **1. Динамічна генерація веб-сторінок:**

PHP дозволяє вбудовувати код прямо в HTML-сторінки, що дозволяє динамічно генерувати вміст веб-сайтів залежно від різних умов та даних.

### **2. Обробка форм:**

PHP використовується для обробки даних, які вводяться користувачами через веб-форми. Це включає в себе валідацію, збереження даних та взаємодію з базами даних.

### **3. Робота з базами даних:**



PHP має багатий набір функцій для взаємодії з різними системами управління базами даних (наприклад, MySQL, PostgreSQL). Це дозволяє розробникам легко отримувати та зберігати дані.

#### **4. Робота з файлами:**

PHP може використовуватися для читання та запису даних в файли на сервері. Це корисно для обробки файлів, таких як завантажені користувачами файли.

#### **5. Обробка помилок:**

Мова має вбудовану систему обробки помилок, яка дозволяє розробникам легше виявляти та виправляти проблеми в коді.

#### **6. Взаємодія з зовнішніми серверами:**

PHP може взаємодіяти з іншими серверами за допомогою різних протоколів, таких як HTTP, FTP, інші API, що робить його корисним для обміну даними з іншими веб-сайтами та службами.

#### **7. Створення графіки та робота з зображеннями:**

PHP підтримує роботу з графікою та зображеннями, включаючи можливості створення та обробки зображень, що робить його корисним для веб-розробки та обробки мультимедійного контенту.

#### **8. Розширюваність:**

Мова має широкий вибір розширень та бібліотек, що дозволяє розробникам розширювати функціональність своїх проектів.

PHP використовується багатьма веб-розробниками, оскільки він простий у вивченні та має велику спільноту розробників, а також відкритий код [13].

## **C#(C Sharp)**

C# (C Sharp) - це мова програмування, яка часто використовується в розробці програм для платформи .NET. І хоча на початку свого існування вона була в основному призначена для створення Windows-додатків, з часом вона також стала популярною в розробці веб-застосунків.

Ось деякі аспекти використання C# у веб-розробці:

### **1. ASP.NET:**

ASP.NET - це фреймворк для побудови веб-застосунків, який використовує мову програмування C#. За допомогою ASP.NET можна створювати різноманітні веб-сайти та веб-додатки, включаючи корпоративні, електронні комерції та інші.

### **2. ASP.NET Core:**

ASP.NET Core - це нова версія ASP.NET, яка є більш легкою та масштабованою. Вона підтримує крос-платформенність і відкритий код. З допомогою ASP.NET Core ви можете розробляти веб-застосунки, які працюють як на Windows, так і на Linux або macOS.

### **3. Мова на Стороні Сервера:**

C# використовується як мова на стороні сервера для обробки логіки та генерації HTML-коду. Вона надає потужний об'єктно-орієнтований підхід та велику кількість функцій для розробників.

### **4. Entity Framework:**

Entity Framework є ORM (Object-Relational Mapping) для .NET, що полегшує взаємодію з базами даних. За допомогою EF, ви можете

працювати з базами даних, використовуючи С#-об'єкти, що робить код більш зрозумілим та легше зберігається та модифікується.

## 5. **Visual Studio:**

Для розробки веб-застосунків на С# часто використовується Visual Studio, інтегроване середовище розробки, яке надає розширені засоби для відлагодження, створення та керування проектами.

## 6. **SignalR:**

SignalR - це бібліотека для реалізації веб-сокетів та інших методів веб-заснованої взаємодії в реальному часі. Вона підтримується .NET і дозволяє створювати асинхронні та взаємодіючі веб-додатки.

## 7. **Razor Pages:**

Razor Pages - це фреймворк в ASP.NET Core, який дозволяє створювати прості та ефективні веб-сторінки з меншою кількістю шаблонного коду.

## 8. **Використання ASP.NET MVC:**

ASP.NET MVC - це паттерн проектування та фреймворк, який дозволяє розробляти веб-застосунки за принципом Model-View-Controller. Він дозволяє розділити логіку, представлення та дані для полегшення розробки та тестування.

## 9. **LINQ (Language Integrated Query):**

LINQ в С# дозволяє робити запити до джерел даних (наприклад, колекцій чи баз даних) за допомогою синтаксису, який схожий на SQL. Це полегшує взаємодію з даними у веб-застосунках.

Використання C# у веб-розробці дозволяє розробникам створювати ефективні, безпечні та легко збережені веб-застосунки з використанням потужних інструментів та бібліотек [14].

## **Python**

Python - це мова програмування високого рівня, яка широко використовується в різних сферах, включаючи веб-розробку. Вона пропонує ряд фреймворків і інструментів, які полегшують розробку веб-застосунків [.

Ось деякі аспекти використання Python у веб-розробці:

### **1. Django:**

Django - це один з найпопулярніших фреймворків для веб-розробки на Python. Він надає повноцінний стек технологій для створення веб-застосунків, включаючи модель-вигляд-контролер (MVC) архітектуру, систему адміністрування, ORM (Object-Relational Mapping) та багато іншого.

### **2. Flask:**

Flask - це легкий та простий у використанні мікрофреймворк, який використовується для швидкого створення веб-застосунків. Він має менше обов'язкових компонентів порівняно з Django, що надає розробникам більшу свободу вибору і адаптації.

### **3. FastAPI:**

FastAPI - це сучасний фреймворк для створення API на Python. Він є швидким, автоматично генерує документацію та підтримує типізацію даних за допомогою Python type hints.

### **4. Pyramid:**

Pyramid - це фреймворк, який прагне бути гнучким та легким. Він дозволяє розробникам вибирати компоненти, які вони хочуть використовувати, і розширювати їх функціональність за потреби.

### **5. Tornado:**

Tornado - це фреймворк для створення асинхронних веб-застосунків. Він особливо підходить для розробки веб-застосунків, які вимагають високої продуктивності та підтримки асинхронного програмування.

#### 6. **CherryPy:**

CherryPy - це мінімалістичний фреймворк, який дозволяє швидко створювати веб-застосунки. Він вбудовує в себе сервер WSGI, що дозволяє запускати веб-застосунки безпосередньо з коду Python.

#### 7. **Web2py:**

Web2py - це фреймворк для створення веб-застосунків, який намагається зробити розробку максимально простою та ефективною. Він включає в себе вбудовану систему адміністрування та інструменти для безпеки.

#### 8. **SQLAlchemy:**

SQLAlchemy - це бібліотека для взаємодії з базами даних у Python. Вона надає ORM та можливість використання чистого SQL для обробки даних.

#### 9. **Jupyter Notebooks:**

Jupyter Notebooks - це інтерактивне середовище, яке дозволяє вам створювати та ділитися документами, що містять живий код, рівень та інтерактивні візуалізації. Воно часто використовується для аналізу даних у веб-розробці.

Python веб-розробка приваблива завдяки своїй зручності, розширюваності та активній спільноті розробників. Вона також широко використовується у сфері науки про дані та штучного інтелекту, що робить її важливим інструментом у сучасному програмуванні [15].

## 3 ІНФОРМАЦІЙНЕ ТА ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ СИСТЕМИ

### 3.1 Розробка веб-сайту

Сайт розроблено на мовах програмування HTML, JavaScript та PHP.

Для початку потрібно створити проект на GitHub. Шаблон сайту має такий вигляд (рис.3.1):

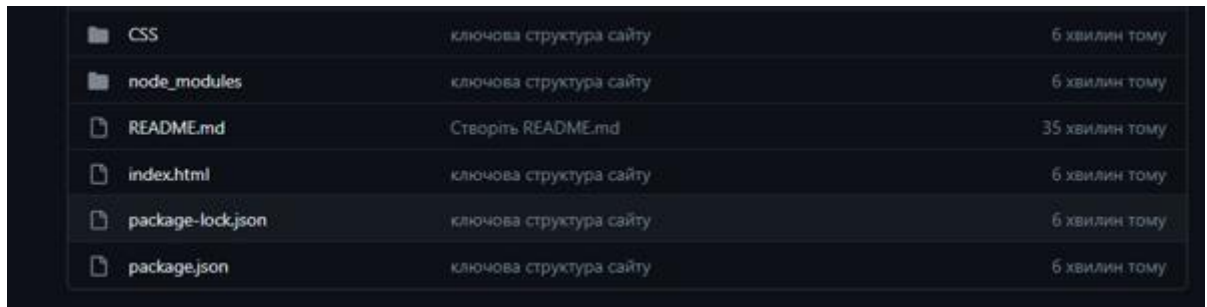


Рисунок 3.1 – Шаблон сайту

Далі потрібно створити гілки. Створення гілок (branches) є важливою стратегією в системах контролю версій, таких як Git. Гілка - це відокремлене напрямок розвитку в проєкті, який дозволяє працювати над новими функціями чи виправленнями без впливу на основний код проєкту (рис.3.2).

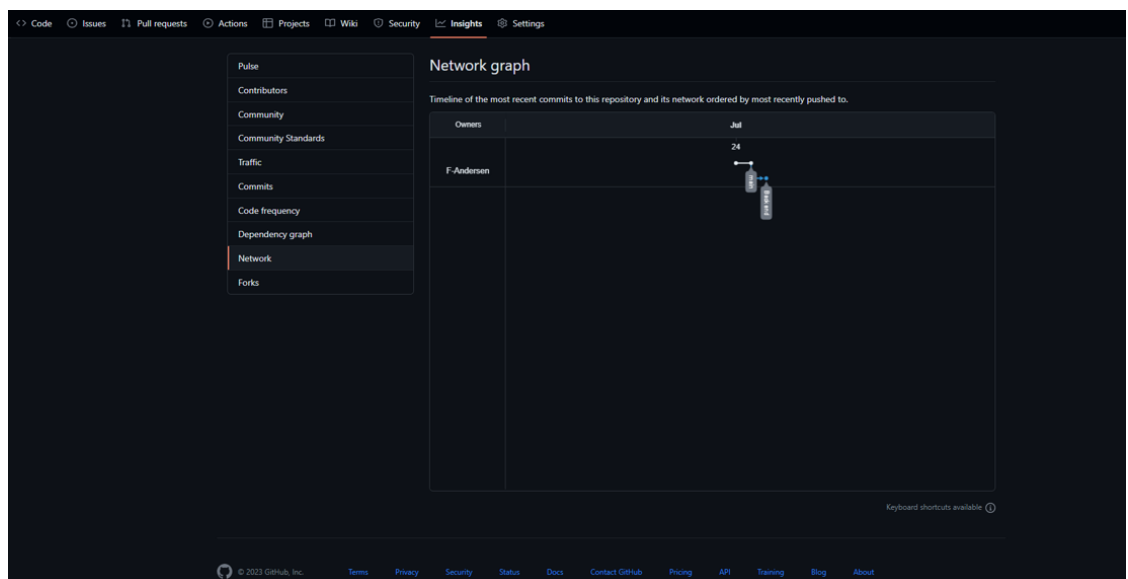


Рисунок 3.2 – Створення гілок

Після цього відкрити утиліту Adobe Photoshop, та розробити макет веб-сайту. Для цього треба створити проєкт, імпортувати картинки в

програму, розмістити сітки та блоки, додати фон та визначитися з кольоровою гаммою, вставити графіку та зображення в макет. У подальшому отримати такий результат (рис.3.3):

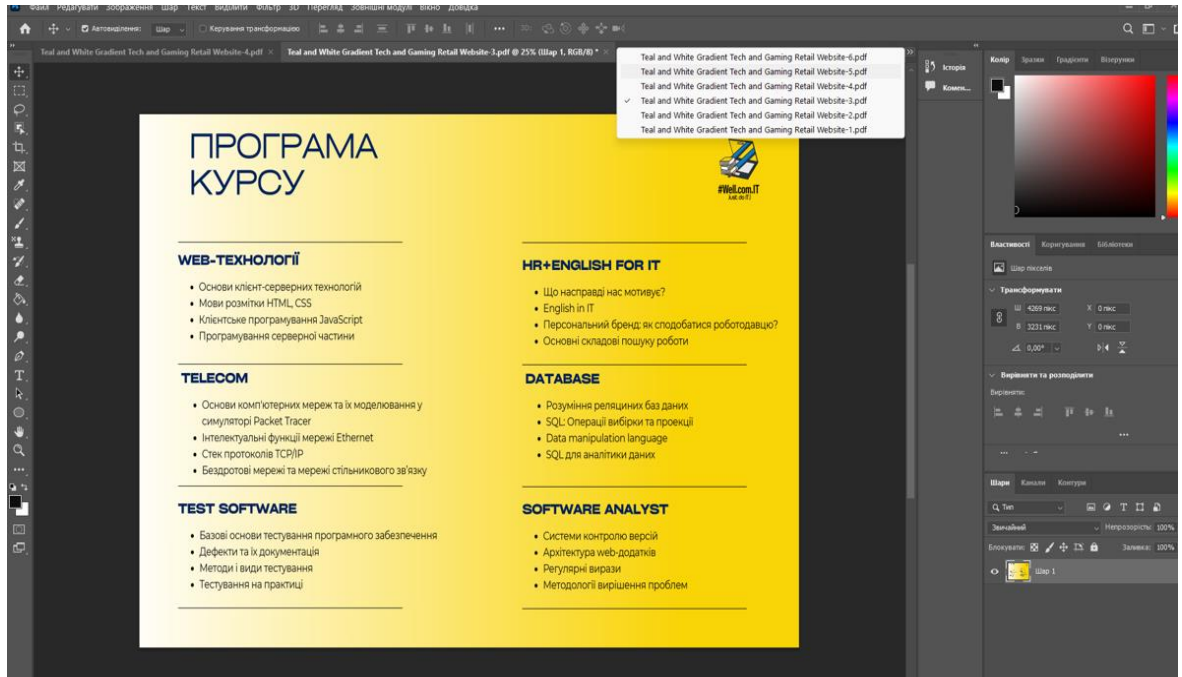


Рисунок 3.3 – Макет веб-сайту

Створення веб-сайту починається безпосередньо з завантаження потрібних графічних засобів, тобто шаблону та іконок (рис.3.4).

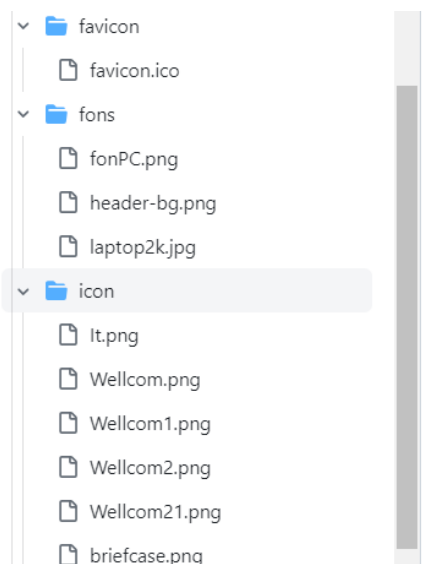


Рисунок 3.4 – Завантаження графіки

### Створення мета-тегів.

HTML-теги "meta" використовуються для надання додаткової інформації про сторінку або документ. Ці теги зазвичай розташовуються в розділі **<head>** HTML-документа. Основні мета-теги включають:

1. **<meta charset="UTF-8">**:

- Вказує кодування символів для документа. Зазвичай використовується **UTF-8**, що підтримує широкий спектр символів.

2. **<meta name="viewport" content="width=device-width, initial-scale=1.0">**:

- Встановлює налаштування відображення для мобільних пристроїв, забезпечуючи адаптивний дизайн.

3. **<meta name="keywords" content="ключові слова">**:

- Вказує ключові слова, пов'язані зі сторінкою. Цей тег рідко використовується пошуковими системами, але може бути корисним для інших інструментів аналізу.

4. **<meta name="description" content="опис сторінки">**:

- Надає короткий опис вмісту сторінки. Цей опис може використовуватися пошуковими системами для створення опису сторінки в результатах пошуку.

5. **<meta name="author" content="автор">**:

- Вказує ім'я автора сторінки.

6. **<meta http-equiv="refresh" content="час; URL=адреса">**:

- Встановлює автоматичне перенаправлення сторінки на інший URL через визначений час.

7. **<meta http-equiv="X-UA-Compatible" content="IE=edge">**:

- Вказує браузеру Internet Explorer використовувати останній двіжок рендерингу.

8. **<meta name="robots" content="index, follow">**:



- Контролює індексацію та слідування посиланням для пошукових систем. Значення "index" означає індексацію, а "follow" — слідування посиланням [16]. (рис.3.5).

```

1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <meta name="viewport" content="width=device-width, initial-scale=1.0">
6
7   <meta name="description"
8     content="Наші курси ІТ-МУЛЬТИКУРС допоможуть школярам отримати перші навички в інформатиці та технол
9   <meta name="keywords"
10    content="курси ІТ, навчання для школярів, перші навички в ІТ, онлайн-навчання, програмування, веб-ро
11  <meta name="robots" content="index, follow">
12
13  <!-- Мета-теги для соціальних мереж -->
14  <meta property="og:title" content="Курси ІТ-МУЛЬТИКУРС для школярів - Здобудьте перші навички в ІТ">
15  <meta property="og:description"
16    content="Наші курси ІТ-МУЛЬТИКУРС допоможуть школярам отримати перші навички в інформатиці та технол
17  <meta property="og:image" content="/IMG/icon/wellcom1.png">
18  <!-- <meta property="og:url" content="посилання на ваш сайт" -->
19
20  <meta name="twitter:card" content="summary_large_image">
21  <meta name="twitter:title" content="Курси ІТ-МУЛЬТИКУРС для школярів - Здобудьте перші навички в ІТ">
22  <meta name="twitter:description"
23    content="Наші курси ІТ-МУЛЬТИКУРС допоможуть школярам отримати перші навички в інформатиці та технол
24  <meta name="twitter:image" content="/IMG/icon/wellcom1.png">
25
26  <script src="https://code.jquery.com/jquery-3.5.1.min.js"
27    integrity="sha256-9/aliu8dGd2tb60SsuzixeV4y/faTqgFtohetphbbj0=" crossorigin="anonymous"></script>
28  <script src="js/jquery.maskedinput.js"></script>
29  <script src="js/jquery.validate.min.js"></script>
30  <script src="js/script.js"></script>
31
32
33  <title>SUNDU BLOG</title>
34
35
36

```

Рисунок 3.5 – Створення мета-тегів

Підключення файлів CSS та шрифтів у веб-розробці виконується за допомогою HTML-тегів та CSS-властивостей. “**Styles.css**” - це зовнішній файл стилів, який має бути в тому ж каталозі, що і HTML-документ (рис.3.6).

```

<link rel="shortcut icon" href="/IMG/favicon/favicon.ico" type="image/x-icon">
<link rel="icon" href="/IMG/favicon/favicon.ico" type="image/x-icon">

<link href="node_modules/modern-normalize/modern-normalize.css" rel="stylesheet" type="text/css">
<link href="CSS/style.css" rel="stylesheet" type="text/css" >
<!-- css -->

<link rel="preconnect" href="https://fonts.googleapis.com">
<link rel="preconnect" href="https://fonts.gstatic.com" crossorigin>
<link href="https://fonts.googleapis.com/css2?family=REM:wght@200;400;700;900&display=swap" rel="stylesheet">
<!-- google/fonts -->

```

Рисунок 3.6 – Підключення файлів CSS

Далі потрібно написати код header, для основи вигляду сайту, він відповідає за заголовок сайту, тобто за його представлену частину. Також туди входить посилання на гугл форму реєстрації. Для зручності використання сайту, були додані дві форми реєстрації: Одна – це посилання на гугл форму, друга це код на php, який відповідає за перенесення даних за допомогою бази даних сайту на пошту відповідального за надання курсів (рис.3.7).

```

<header id="header" class="header">
  <div class="container">
    <div class="nav">
      
      
      
    </div>
    <div class="offer">
      <div class="logo-IT">
        
      </div>
      <h1>
        <span class="span-text">МУЛЬТИКУРС</span>
      </h1>
      </div>
      <p>Зроби перший крок в IT!</p>
      <div class="btn-flex">
        <a href="https://docs.google.com/forms/d/e/1FAIpQLSezpdKDLs75NebfDeCOjiumWkqGbhgfCFQhzTLxsb71eY81g/viewform" class="btn" target="_blank">ЗАРЕЄСТРУВАТИСЬ</a>
      </div>
    </div>
  </div>
</header>

```

Рисунок 3.7 – Створення header

Після цього за допомогою мови HTML створюються секції контенту сайту, тобто основна інформація для тих кому цікаві курси. Приклад секції разом з програмою (рис.3.8):

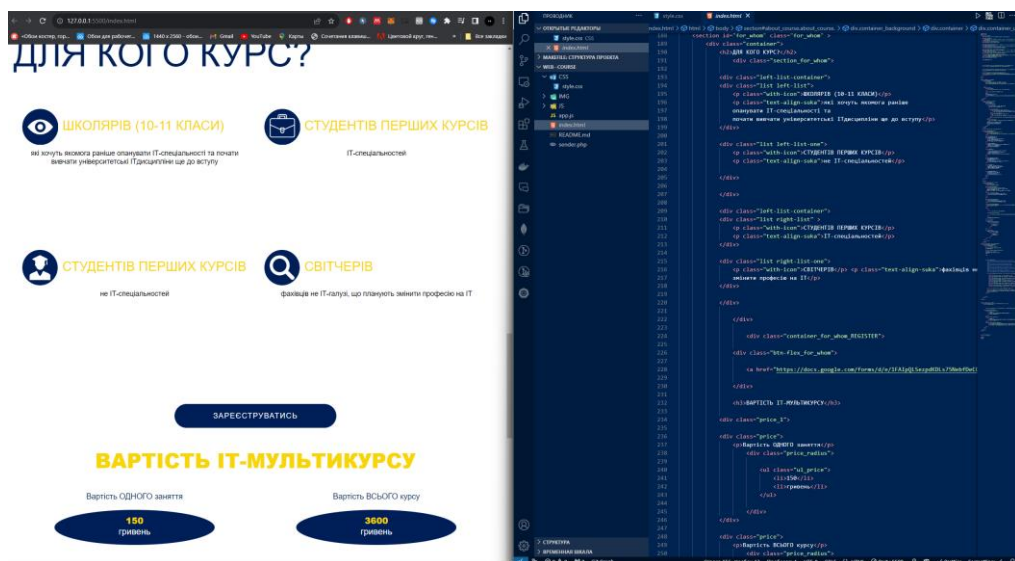


Рисунок 3.8 – Секція з сайтом

Після завершення оформлення сайту, робимо його зручним для користувачів мобільних пристроїв.

Для того, щоб зробити сайт зручним для мобільних пристроїв, слід використовувати техніки адаптивного дизайну. По-перше треба визначити стилі, які будуть застосовуватися для різних розмірів екрану. Замість традиційного горизонтального меню можна вставити вертикальне розташування або випадаюче меню. Також забезпечити достатній розмір тач-дружніх областей. Замість фіксованих ширин використовувати відсотки або інші відносні одиниці для розташування та розмірів елементів. Використовувати відносні одиниці для розміру шрифту, щоб вони адаптувались до розміру екрану. (рис.3.9) Перевірити, як сайт виглядає та веде себе на різних мобільних пристроях та веб-переглядачах [17].

```
@media only screen and (max-width: 600px) {
  /* Стили для екранів шириною до 600px (мобільні пристрої) */
  body {
    font-size: 14px;
  }

  header {
    padding: 0.5em;
  }

  nav {
    flex-direction: column;
  }
}
```

Рисунок 3.9 – Приклад коду для мобільних пристроїв

Залишається зробити форми реєстрації. Перша форма (рис.3.10) – проста, як для розробника так і для користувача – Google Form. Для створення гугл форми потрібно:

1. **Обрати тип форми:** Запропоновано вибрати між створенням опитування або форми збору відповідей. Потрібно обрати потрібний тип.
2. **Додати питання:** Коли створюється форма, треба додати питання, використовуючи різні типи відповідей, такі як текстові поля, варіанти вибору, флажки тощо.
3. **Налаштувати параметри питань:** Для кожного питання можна налаштувати додаткові параметри, такі як обов'язкові для заповнення, додатковий опис і т.д.
4. **Додати нові розділи або розділові питання (опціонально):** Якщо форма складається з різних частин, можна додавати розділи, щоб її легше розуміти.
5. **Налаштування оформлення:** Натискати на іконку "Теми" (вигляд шестерні) для вибору кольорової схеми і оформлення форми.
6. **Налаштування параметрів форми:** Натискати на іконку шестерні в правому верхньому куті, щоб змінити налаштування, такі як дозвіл повторного заповнення, обмеження відправлення тощо.
7. **Налаштування доступу:** На сторінці з формою натиснути на кнопку "Загальний доступ", щоб налаштувати, кому дозволено переглядати або заповнювати форму.
8. **Поділіться формою:** Після того як форма виглядає як потрібно, додати посилання в код сайту і прив'язати до кнопки «Зареєструватись» [18].

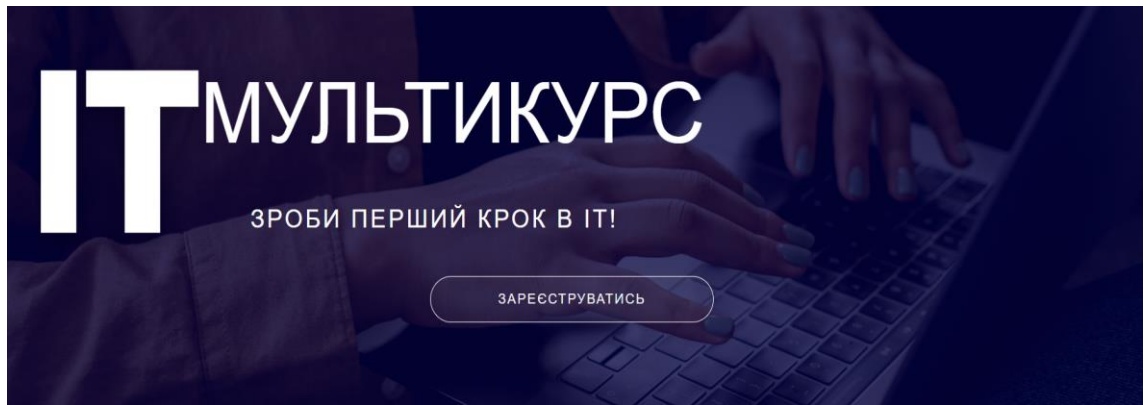


Рисунок 3.10 – Кнопка реєстрації

Так виглядає гугл форма після її обробки, ця гугл форма з'являється після натискання кнопки «Зареєструватись» (рис.3.11):

Рисунок 3.11 – Google форма реєстрації на курси

Друга форма (рис.3.12), більш складна в розробці, але дуже зручна у використанні для користувача. Ця форма розроблена на мовах JavaScript та PHP. Її Суть у збереженні інформації на базу даних сайту, та пересилання цієї інформації пошту розробнику. Мова JavaScript відповідає за

заповнення коректної інформації та за збереження інформації. Мова PHP відповідає за пересилання цієї інформації на пошту власника форми.

The image shows a registration form on a yellow background. The title 'РЕЄСТРАЦІЯ' is in large blue letters. The form consists of several input fields with blue labels on the left and yellow input boxes on the right:

- Ім'я** (Name): Input field containing 'Денис'.
- Прізвище** (Surname): Input field containing 'Карпенко'.
- E-mail**: Input field containing 'Email'.
- Номер телефону** (Phone number): Input field containing '+380 ( ) \_\_\_-\_\_-\_\_'.
- Свій текст** (Your text): Input field containing 'Текст'.

Рисунок 3.12 – Форма реєстрації

Для покращення дизайну та інформаційності веб-сайту було вирішено додати слайдер на мові CSS[19]. Слайдер на веб-сайті є корисним інструментом, який може бути використаний для різних цілей та покращення користувацького досвіду. У цьому випадку слайдер слугує для демонстрації викладачів, які проводять курси(рис.3.13).

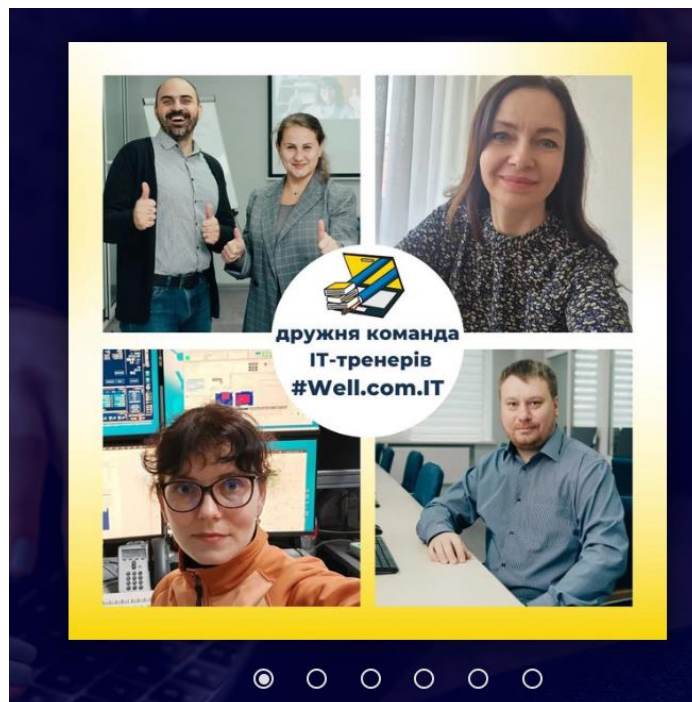


Рисунок 3.13 – Слайдер

Через події які відбуваються в теперішній час, було вирішено захистити сайт від проксі, ботів та ір-адресів, які не належать Україні. Для цього був використаний сайт `hide.click`[20] (рис.3.14).

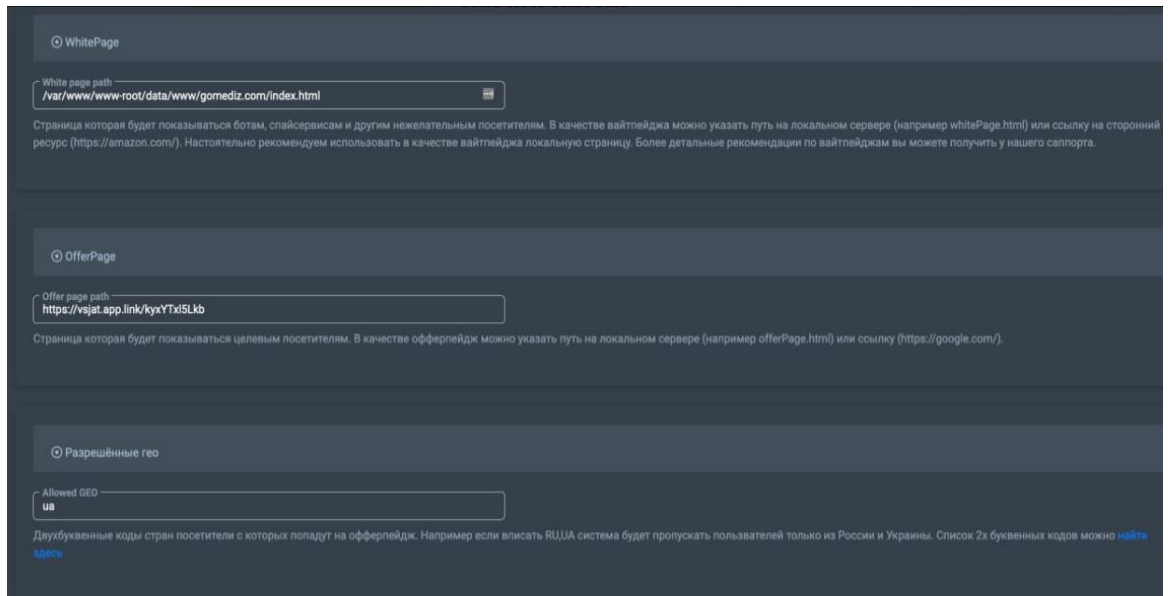


Рисунок 3.14 – Захист сайту від зайвих ІР-адресів

Ці заходи спрямовані на зменшення ризику атак, які можуть бути виконані через проксі, і покращення загальної безпеки вашого веб-сайту.

### 3.2 Тестування веб-сайту навчального центру #Well.com.IT

Створення форми реєстрації на основі PHP включає в себе створення HTML-форми та відповідного PHP-коду для обробки введених користувачем даних. Після того як усі дані були введені вірно PHP-код переправляє інформацію на електронну пошту [21] (рис.3.15 – 3.17).

```

1  app.js
2  <?php
3  use PHPMailer\PHPMailer\PHPMailer;
4  use PHPMailer\PHPMailer\Exception;
5
6  require 'phpmailer/src/Exception.php';
7  require 'phpmailer/src/PHPMailer.php';
8
9  $mail = new PHPMailer(true);
10 $mail->CharSet = 'UTF-8';
11 $mail->setLanguage('ru', 'phpmailer/language/');
12 $mail->isHTML(true);
13
14 //От кого письмо
15 $mail->setFrom('devicedustbusters@gmail.com!', 'Фрилансер по жизни');
16 //Кому отправить
17 $mail->addAddress('devicedustbusters@gmail.com');
18 //Тема письма
19 $mail->Subject = 'Привет! Это "Фрилансер по жизни"';
20
21 //Рука
22 $hand = "Правая";
23 if($_POST['hand'] == "left"){
24     $hand = "Левая";
25 }
26
27 //Тело письма
28 $body = '<h1>Встречайте супер письмо!</h1>';
29
30 if(trim(empty($_POST['name']))){
31     $body.='<p><strong>Имя:</strong> '.$_POST['name'].'</p>';
32 }
33 if(trim(empty($_POST['email']))){
34     $body.='<p><strong>E-mail:</strong> '.$_POST['email'].'</p>';
35 }
36 if(trim(empty($_POST['hand']))){
37     $body.='<p><strong>Рука:</strong> '.$hand.'</p>';
38 }
39 }

```

Рисунок 3.15 – Частина коду PHP форми

Після остаточної обробки маємо SMS-повідомлення на пошту у вигляді форми заповненої раніше:

Рисунок 3.16 – Заповнена форма



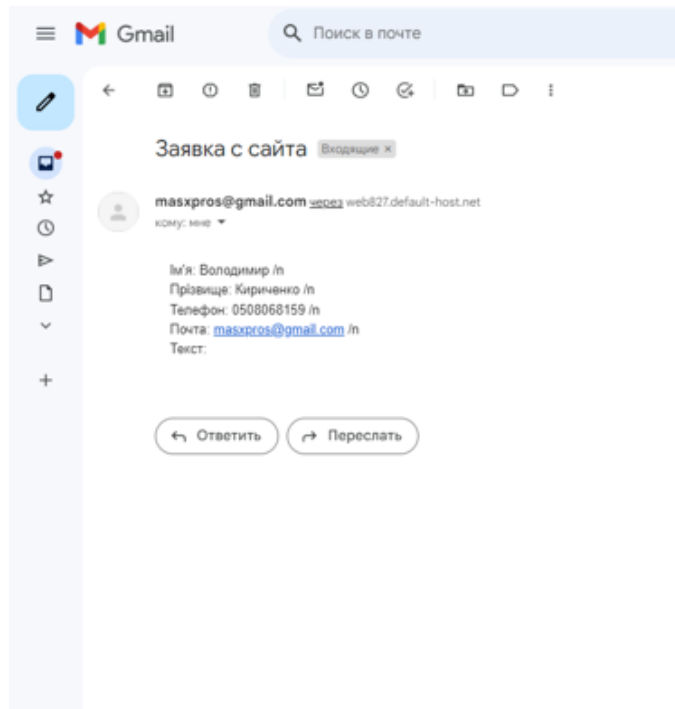
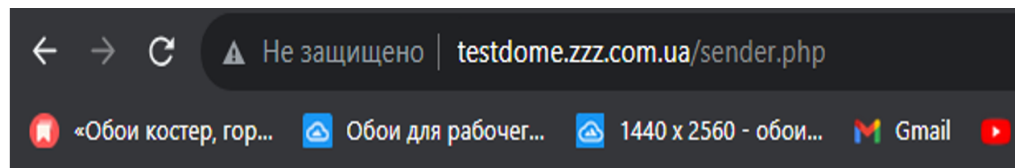


Рисунок 3.17 – SMS-повідомлення від сайту

Від чого маємо готовий результат (рис.3.18).



Привіт форма відправлена

Рисунок 3.18 – Результат роботи PHP

На виході маємо робочу PHP-форму яка додає зручності у користуванні сайту для користувачів та у складанні списків користувачів для викладача.

Отже, маємо поліпшений дизайн, зручність використання для студентів, викладачів та розробників, які у подальшому будуть розвивати проєкт, тобто доступний, читабельний код розбитий на секції та пояснення до нього. Викладачі мають гарно оформлені списки студентів зі зручним доступом до них. Студенти матимуть гарний дизайн, цільову інформацію та своєчасну підтримку у разі виникнення питань.

## ВИСНОВКИ

Результатом виконання роботи є створений веб-сайт спрямований на роботу з учнями в галузі ІТ та є платформою для навчання та обміну знаннями. Реєстрація та обмеження доступу для користувачів з України було впроваджено для забезпечення безпеки.

У ході виконання кваліфікаційної роботи магістра було виконано наступні завдання:

1. Розглянуті та проаналізовані сучасні мови програмування які використовують для створення освітніх веб-сайтів.
2. Проведено аналіз тематичної літератури, вибір програмних засобів для реалізації проекту.
3. Проведено порівняльний аналіз сайтів навчальних центрів ІТ/
4. Обрані технології на розробку освітнього веб-сайту.
5. Реалізовано дизайн та роботу веб-сайту навчального центру #Well.com.IT.
6. Проведено тестування веб-сайту, шляхом перевірки його компонентів.

Створений веб-сайт стане ефективним інструментом для розвитку та навчання в галузі ІТ, а його функціонал та можливості відповідають вимогам сучасного освітнього середовища. У майбутньому планується розширення функціоналу та привертання нових користувачів.

## СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Роль і значення освітніх веб-ресурсів? [Електронний ресурс] – Режим доступу до ресурсу: <http://www.galanet.at.ua/forum/15-4-1> .
2. IT Education Academy (ITEA) — міжнародний освітній центр, який готує спеціалістів-практиків IT-сфери [Електронний ресурс] – Режим доступу до ресурсу: [https://itea.ua/about\\_itea/](https://itea.ua/about_itea/)
3. Комп'ютерна Академія IT Step [Електронний ресурс] – Режим доступу до ресурсу: [https://itstep.org/uk/about-academy\\_3](https://itstep.org/uk/about-academy_3)
4. GoIT – платформа IT-курсів [Електронний ресурс] – Режим доступу до ресурсу: <https://goit.global/ua/blog/>
5. Порівняльний аналіз ефективності веб-сайтів державних та комерційних з во сфери культури україни [Електронний ресурс] – Режим доступу до ресурсу: <https://molodyivchenyi.ua/>
6. Комунікація в мережі [Електронний ресурс] – Режим доступу до ресурсу: <http://elbib.in.ua/komunikatsiya-v-mereji-sotsiologiya-internetu.html>
7. Освітні веб-сайти [Електронний ресурс] – Режим доступу до ресурсу: [https://www.marschool30.net./](https://www.marschool30.net/)
8. Основні елементи веб-сайту для закладу освіти [Електронний ресурс] – Режим доступу до ресурсу: <https://oplatforma.com.ua/article/2828-yak-stvoriti-ta-objiviti-sayt-zakladu-osvti>
9. Етапи створення веб сайтів: які є основні кроки розробки [Електронний ресурс] – Режим доступу до ресурсу: <https://webtune.com.ua/statti/web-rozrobka/etapy-stvorennya-veb-sajtiv/>
10. Як вибрати мову програмування [Електронний ресурс] – Режим доступу до ресурсу: <https://lemon.school/blog/yaku-movu-programuvannya-vibrati>
11. Що таке CSS? [Електронний ресурс] – Режим доступу до ресурсу: [https://css.in.ua/article/shcho-take-html\\_10](https://css.in.ua/article/shcho-take-html_10)

12. JavaScript в HTML [Електронний ресурс] – Режим доступу до ресурсу: <https://angularlessons.gitbook.io/javascriptiseasy>
13. Форма реєстрації на PHP [Електронний ресурс] – Режим доступу до ресурсу: <https://themewp.inform.click/uk/jak-stvoriti-sistemu-reiestracii-php-dlja-vashogo-veb-sajtu/>
14. Як використовувати .NET [Електронний ресурс] – Режим доступу до ресурсу: <https://dou.ua/lenta/articles/how-to-learn-dotnet/>
15. PYTHON для створення сайтів [Електронний ресурс] – Режим доступу до ресурсу: <https://webcase.com.ua/uk/python-sozdaniye-sayta/>
16. Знайомство з Visual Studio Code - Фріланс створення сайтів [Електронний ресурс] – Режим доступу до ресурсу: <https://romul.name/blog/znayomstvo-z-visual-studio-code/>
17. Як зробити сайт зручним для мобільних пристроїв [Електронний ресурс] – Режим доступу до ресурсу: <https://www.ranktracker.com/uk/blog/how-to-make-a-mobile-friendly-website-8-best-practices/>
18. Як створити Google форму [Електронний ресурс] – Режим доступу до ресурсу: <https://www.google.com/intl/uk/forms/about/>
19. Красиве і зручне оформлення сайту на CSS3 [Електронний ресурс] – Режим доступу до ресурсу: <https://chili-web.com.ua/css3/>
20. Документ захисту від небажаних ір-адресів [Електронний ресурс] – Режим доступу до ресурсу: [https://my.hide.click/download\\_ua.html](https://my.hide.click/download_ua.html)
21. Як тестувати веб-сайт основні етапи і поради [Електронний ресурс] – Режим доступу до ресурсу: <https://brainlab.com.ua/uk/blog-uk/yak-testuvati-veb-sayt-osnovn-etapi-poradi>

## ДОДАТОК А

## Програмна реалізація на мові програмування HTML(файл index.html):

```

<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">

  <meta name="description"
    content="Наші курси ІТ-МУЛЬТИКУРС допоможуть школярам отримати перші
    навички в інформатиці та технологіях. Долучайтесь до наших курсів і розвивайте
    свій потенціал в галузі ІТ.">
  <meta name="keywords"
    content="курси ІТ, навчання для школярів, перші навички в ІТ, онлайн-
    навчання, програмування, веб-розробка, інформатика">
  <meta name="robots" content="index, follow">

  <!-- Мета-теги для соціальних мереж -->
  <meta property="og:title" content="Курси ІТ-МУЛЬТИКУРС для школярів -
  Здобудьте перші навички в ІТ">
  <meta property="og:description"
    content="Наші курси ІТ-МУЛЬТИКУРС допоможуть школярам отримати перші
    навички в інформатиці та технологіях. Долучайтесь до наших курсів і розвивайте
    свій потенціал в галузі ІТ.">
  <meta property="og:image" content="/IMG/icon/Wellcom1.png">
  <!-- <meta property="og:url" content="посилання_на_ваш_сайт" -->

  <meta name="twitter:card" content="summary_large_image">
  <meta name="twitter:title" content="Курси ІТ-МУЛЬТИКУРС для школярів -
  Здобудьте перші навички в ІТ">
  <meta name="twitter:description"
    content="Наші курси ІТ-МУЛЬТИКУРС допоможуть школярам отримати перші
    навички в інформатиці та технологіях. Долучайтесь до наших курсів і розвивайте
    свій потенціал в галузі ІТ.">
  <meta name="twitter:image" content="/IMG/icon/Wellcom1.png">

  <script src="https://code.jquery.com/jquery-3.5.1.min.js"
    integrity="sha256-9/aliU8dGd2tb6OSsuzixeV4y/faTqgFtohetphbbj0="
    crossorigin="anonymous"></script>
  <script src="js/jquery.maskedinput.js"></script>
  <script src="js/jquery.validate.min.js"></script>
  <script src="js/script.js"></script>

  <title>SUMDU BLOG</title>

```

```

<link rel="shortcut icon" href="/IMG/icon/Wellcom.png" type="image/x-icon">
<link rel="icon" href="/IMG/icon/Wellcom.png" type="image/x-icon">

<link href="node_modules/modern-normalize/modern-normalize.css"
rel="stylesheet" type="text/css">
<link href="CSS/style.css" rel="stylesheet" type="text/css" >
<link href="CSS/style-2.css" rel="stylesheet" type="text/css" >
<!-- css -->

<link rel="preconnect" href="https://fonts.googleapis.com">
<link rel="preconnect" href="https://fonts.gstatic.com" crossorigin>
<link
href="https://fonts.googleapis.com/css2?family=REM:wght@200;400;700;900&display
=swap" rel="stylesheet">
<!-- google/fonts -->

</head>

<body>

<header id="header" class="header">
  <div class="container-2">

    <div class="nav">
      
      
      

    </div>
    <div class="offer">
      <div class="logo-IT">
        

        <h1>
          <span class="span-text">МУЛЬТИКУРС</span>
        </h1>
      </div>
      <p>Зроби перший крок в IT!</p>
      <div class="btn-flex">

```

```

        <a
href="https://docs.google.com/forms/d/e/1FAIpQLSezpdKDLs75NebfDeCOJiuwNKqGbHgFC
FQhzTLLxxb71eY81g/viewform" class="btn" target="_blank">ЗАРЕЕСТРУВАТИСЬ</a>
    </div>

</div>
<div class="wrapper">
    <input type="radio" name="point" id="slide1" checked>
    <input type="radio" name="point" id="slide2">
    <input type="radio" name="point" id="slide3">
    <input type="radio" name="point" id="slide4">
    <input type="radio" name="point" id="slide5">
    <input type="radio" name="point" id="slide6">

    <div class="slider">
        <div class="slides slide1"> </div>
        <div class="slides slide2"></div>
        <div class="slides slide3"></div>
        <div class="slides slide4"></div>
        <div class="slides slide5"></div>
        <div class="slides slide6"></div>
    </div>
    <div class="controls">
        <label for="slide1"></label>
        <label for="slide2"></label>
        <label for="slide3"></label>
        <label for="slide4"></label>
        <label for="slide5"></label>
        <label for="slide6"></label>
    </div>
</div>
</header>

<section id="about" class="about">
    <div class="container">
        <h2>ІТ-МУЛЬТИКУРС</h2>
        <div class="description">

            <div class="desc-left">
                <h3>- це можливість сформувати ІТ-бекграунд
                та почати зростати професійно</h3>

                <div class="custom-list">
                    <p class="li-icon-one">Тривалість навчання:<br>
                    <span class="span-bolt span-bolt-one">3
місяці</span> <br>
                    <span class="span-bolt span-bolt-one">24
заняття</span>

```

```

        </p>
        <p class="li-icon-two">Заняття
            <span class="span-bolt">у вихідні дні <br>
        </span>
            <span class="span-bolt-one">Тривалість:</span><span
class="span-bolt">80 хвилин </span>
        </p>
        <p class="li-icon-three"><span class="span-bolt">On-
line та Off-line</span> формат</p>
    </div>
</div>
<div class="desc-right">
    
    <p>
        Навчання проводять досвідчені тренери,
        які допоможуть сформувати бекграунд
        знань з 6 актуальних ІТ-напрямів.
    </p>
</div>
</div>
</div>
</section>

<section id="about_course" class="about_course " >

<div class="container_background">
    <div class="container">
        <div class="container_header_about_course ">
            <h2>ПРОГРАМА <br>КУРСУ</h2>
            
        </div>

        <div class="container_ul ">
        <div class="container_ul_left">
        <ul> <h3>WEB-ТЕХНОЛОГІЇ</h3>
            <li>Основи клієнт-серверних технологій</li>
            <li>Мови розмітки HTML, CSS</li>
            <li>Клієнтське програмування JavaScript</li>
            <li>Програмування серверної частини</li>
        </ul>

        <ul> <h3>TEST SOFTWARE</h3>
            <li>Базові основи тестування програмного забезпечення</li>
            <li>Дефекти та їх документація</li>
            <li>Методи і види тестування</li>
            <li>Тестування на практиці</li>
        </ul>

```



```

<ul> <h3>TELECOM</h3>
    <li>Основи комп'ютерних мереж та їх моделювання у симуляторі Packet Tracer</li>
    <li>Інтелектуальні функції мережі Etherne</li>
    <li>Стек протоколів TCP/IP</li>
    <li>Бездротові мережі та мережі стільникового зв'язку</li>
</ul>

</div>

<div class="container_ul_right ">
<ul> <h3>HR+ENGLISH FOR IT</h3>
    <li>Що насправді нас мотивує?</li>
    <li>English in IT</li>
    <li>Персональний бренд: як сподобатися роботодавцю?</li>
    <li>Основні складові пошуку роботи</li>
</ul>

<ul> <h3>DATABASE</h3>
    <li>Розуміння реляційних баз даних</li>
    <li>SQL: Операції вибірки та проєкції</li>
    <li>Data manipulation language</li>
    <li>SQL для аналітики даних</li>
</ul>

<ul> <h3>SOFTWARE ANALYST</h3>
    <li>Системи контролю версій</li>
    <li>Архітектура web-додатків</li>
    <li>Регулярні вирази</li>
    <li>Методології вирішення проблем</li>
</ul>
</div>

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

</section>

<section id="for_whom" class="for_whom" >
    <div class="container">
        <h2>ДЛЯ КОГО КУРС?</h2>
        <div class="section_for_whom">

            <div class="left-list-container">

```

```

<div class="list left-list">
  <p class="with-icon">ШКОЛЯРІВ (10-11 КЛАСИ)</p>
  <p class="text-align-suka">які хочуть якомога раніше
    опанувати ІТ-спеціальності та
    почати вивчати університетські ІТдисципліни ще до вступу</p>
</div>

<div class="list left-list-one">
  <p class="with-icon">СТУДЕНТІВ ПЕРШИХ КУРСІВ</p>
  <p class="text-align-suka">не ІТ-спеціальностей</p>

</div>

</div>

<div class="left-list-container">
<div class="list right-list" >
  <p class="with-icon">СТУДЕНТІВ ПЕРШИХ КУРСІВ</p>
  <p class="text-align-suka">ІТ-спеціальностей</p>
</div>

<div class="list right-list-one">
  <p class="with-icon">СВІТЧЕРІВ</p> <p class="text-align-
suka">фахівців не ІТ-галузі, що планують
  змінити професію на ІТ</p>
</div>

</div>

  </div>

  <div class="container_for_whom_REGISTER">

  <div class="btn-flex_for_whom">

    <a
href="https://docs.google.com/forms/d/e/1FAIpQLSezpdKDLs75NebfDeCOJiuwNKqGbHgFC
FQhzTLLxxb71eY81g/viewform" class="btn" target="_blank">ЗАРЕЄСТРУВАТИСЬ</a>

  </div>

  <h3>ВАРТІСТЬ ІТ-МУЛЬТИКУРСУ</h3>

<div class="price_1">

<div class="price">
  <p>Вартість ОДНОГО заняття</p>
  <div class="price_radius">

```

```

        <ul class="ul_price">
            <li>175</li>
            <li>гривень</li>
        </ul>
    </div>
</div>

<div class="price">
    <p>Вартість ВСЬОГО курсу</p>
    <div class="price_radius">

        <ul class="ul_price">
            <li>4200</li>
            <li>гривень</li>
        </ul>

    </div>
</div>

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

<section id="registration" class="registration ">
    <div class="container_background">
        <div class="container">
            <div class="container_input">

                <div class="container_registration_left">
                    <h2>РЕЄСТРАЦІЯ</h2>

                    <div class="container_registration_left_one">

                        <h3>Успішний старт в IT</h3>
                        <p>починається зі відомого вибору</p>
                    </div>

                </div>

            </div>

            <div class="input_form">

                <form action="sender.php" method="post" class="text-field"
name="simple form" autocomplete="on">

```

```

        <!-- Ім'я -->
        <label class="text-field__label"
for="firstname">Ім'я</label>
        <input class="text-field__input name" type="text"
name="name" id="username" placeholder="Ім'я">
        <!-- Прізвище -->
        <label class="text-field__label" for="firstname">Прізвище
</label>
        <input class="text-field__input" type="text "
name="surname" placeholder="Прізвище" required>
        <!-- E-mail pattern=" .+@globex\.com" -->
        <label class="text-field__label" for="firstname">E-
mail</label>
        <input class="text-field__input" type="email" name="email"
id="email1" placeholder="Email">
        <!-- Номер телефону -->
        <label class="text-field__label" for="firstname">Номер
телефону</label>
        <input class="text-field__input phone" type="tel"
id="phone1" name="phone" placeholder="+380">
        <!-- -->
        <label class="text-field__label" for="firstname">Свій
текст</label>
        <textarea class="text-field__input" name="text" id=""
cols="30" rows="10" placeholder="Текст"></textarea>

        <button class="send-form btn" id="button">Відправити
дані</button>

        <input class="checkbox" type="checkbox" id="agreement"
name="agreement" checked />
        <label class="checkbox_text" for="agreement">Даю згоду на
обробку персональних даних</label>
        <div class="status"></div>
    </form>
</div>

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

</section>

<footer id="footer" class="footer">

```

```
<div class="container"> <div class=" border-top-suka"><p></p></div>
    <div class="footer-text">

        <div class="footer_container">
            <div class="footer_container_ul">
                <ul>
                    <li>PHONE</li>
                    <li><a href="tel:+380951760940">+38(095)-17-60-940</a>
                    <li>Dmytro Velykodnyi</li>

                </ul>
            </div>
            <div class="footer_container_ul" >
                <ul>
                    <li>EMAIL</li>
                    <li><a
href="mailto:wellcomit@elit.sumdu.edu.ua">wellcomit@elit.sumdu.edu.ua</a></li>
                </ul>
            </div>
            <div class="footer_container_instagram">
                <a href="https://www.facebook.com/well.com.it"
target="_blank">
</a>

                <a href="https://www.instagram.com/well.com.it/"
target="_blank"></a>

                <a href="https://t.me/velykodnyi_dmytro"
target="_blank"></a>
            </div>
        </div>
    </div>
</footer>

<!-- <h2>РЕЄСТРАЦІЯ</h2> -->
<script src="/JS/app.js">

</script>
</body>
</html>
```

## ДОДАТОК Б

### Програмна реалізація форми реєстрації на php:

Код, який відповідає за пересилання даних:

```
<?php
$name = $_POST['name'];
$surname = $_POST['surname'];
$phone = $_POST['phone'];
$email = $_POST['email'];
$text = $_POST['text'];

$to = "devicedustbusters@gmail.com";
$date = date ("d.m.Y");
$time = date ("h:i");
$from = $email;
$subject = "Заявка с сайта";

$msg="
Ім'я: $name /n
Прізвище: $surname /n
Телефон: $phone /n
Почта: $email /n
Текст: $text";
mail($to, $subject, $msg, "From: $from ");

// to
?>

<p>Привіт форма відправлена</p>
```