Національний університет «Полтавська політехніка імені Юрія Кондратюка»

(повне найменування вищого навчального закладу)

Навчально науковий інститут інформаційних технологій та робототехніки\_

(повна назва факультету)

Кафедра комп’ютерних та інформаційних технологій і систем\_

(повна назва кафедри)

**Пояснювальна записка**

**до дипломного проєкту (роботи)**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_бакалавра\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(освітньо-кваліфікаційний рівень)

на тему

Розробка апаратно-програмних засобів інформаційної системи для Школи Інтернет-маркетингу на прикладі онлайн курсу по SEO просуванню

Виконав: студент 4 курсу, групи 402-ТК

спеціальності

123 Комп’ютерна інженерія

(шифр і назва напряму)

Довбиш М.С.

(прізвище та ініціали)

Керівник Васюта В.В.

(прізвище та ініціали)

Рецензент Хорунжий Є.І

(прізвище та ініціали)

Полтава – 2021

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ**

**НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ**

**«ПОЛТАВСЬКА ПОЛІТЕХНІКА ІМЕНІ ЮРІЯ КОНДРАТЮКА»**

**НАВЧАЛЬНО НАУКОВИЙ ІНСТИТУТ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ ТА РОБОТОТЕХНІКИ**

**КАФЕДРА КОМП’ЮТЕРНИХ ТА ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ І СИСТЕМ**

**КВАЛІФІКАЦІЙНА РОБОТА БАКАЛАВРА**

**спеціальність 123 «Комп’ютерна інженерія»**

**на тему**

**«Розробка апаратно-програмних засобів інформаційної системи для Школи Інтернет-маркетингу на прикладі онлайн курсу по SEO просуванню»**

**Студента групи 402-ТК Довбиша Максима Сергійовича**

Керівник роботи

кандидат технічних наук,

доцент Васюта В.В.

Завідувач кафедри

кандидат технічних наук,

доцент Фесенко Т.М.

Полтава – 2021

**РЕФЕРАТ**

Загальний обсяг текстової частини дипломної роботи становить 70 сторінку формату А4. Робота складається з переліку скорочень, умовних позначень, вступу, 3-ох розділів, висновків, списку використаних джерел та двох додатків. Дипломна робота містить 27 рисунків, 1 таблицю. В роботі використано 19 науково-технічних джерел.

**Об’єкт дослідження** — процес управління вмістом інтернет-ресурсів.

**Предмет дослідження** — інформаційно-навчальний веб-сайт для Школи Інтернет-маркетингу на прикладі онлайн курсу по SEO просуванню використовуючи бібліотеку jQuery з підтримкою зворотного зв'язку із замовниками.

**Мета дипломної роботи:** підтримка зворотного зв’язку з клієнтами для підвищення ефективності освітньої програми і вивчення нової цифрової професії по SEO просуванню за допомогою веб-ресурсу.

Результатами роботи: порівняння найбільш популярних фреймворків, розгляд інструментів розробки веб-сайтів і створення веб-ресурсу, який допоможе користувачам легко навчитись цифровій професії використовуючи зворотний зв'язок по електронній пошті.

Результати дипломної роботи можуть бути використані для подальших досліджень по темі, а саме для розширення функціональності вищевказаного освітнього сайту, а також створення таких подібних ресурсів.

Ключові слова: інформаційно-навчальний веб-cайт, HTML, SCSS, TypeScript, PHP, Laravel, jQuery, MySQL.

# ABSTRACT

The total volume of the text part of the thesis is 70 A4 pages. The paper consists of a list of abbreviations, symbols, an introduction, 3 sections, conclusions, a list of sources used, and two appendices. The thesis contains 27 Figures, 1 table. The paper uses 19 scientific and technical sources.

**The object of research** — is the process of managing the content of internet resources.

**The subject of the research** —is an informational and educational website for the School of Internet Marketing on the example of an online course on SEO promotion using the jQuery library with support for customer feedback.

**The purpose of the thesis:** to support customer feedback to improve the effectiveness of the educational program and learn a new digital profession in SEO promotion using a web resource.

**Results:** compare the most popular frameworks, review website development tools, and create a web resource that will help users easily learn the digital profession using email feedback.

The results of the thesis can be used for further research on the topic, namely to expand the functionality of the above-mentioned educational site, as well as to create such similar resources.

Keywords: informational and educational website, HTML, SCSS, TypeScript, PHP, Laravel, jQuery, MySQL.

# ЗМІСТ

[ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СКОРОЧЕНЬ І ТЕРМІНІВ 6](#_Toc75351240)

[ВСТУП 7](#_Toc75351241)

[РОЗДІЛ 1 АНАЛІЗ ПРОБЛЕМИ І ПОЯСНЕННЯ ЗАВДАННЯ ДОСЛІДЖЕННЯ 9](#_Toc75351242)

[1.1 Обґрунтування необхідності веб-сайту 9](#_Toc75351243)

[1.2 Класифікація сайтів 10](#_Toc75351244)

[1.3 Переваги інформаційного та навчального сайту 11](#_Toc75351245)

[1.4 Постановка завдання 12](#_Toc75351246)

[1.5 Поліпшення інформаційного та навчального сайту 13](#_Toc75351247)

[1.6 Використання зворотного зв'язку 15](#_Toc75351248)

[1.7 Панель адміністратора 17](#_Toc75351249)

[1.8 Висновки до першого розділу 17](#_Toc75351250)

[РОЗДІЛ 2 ТЕОРЕТИЧНІ АСПЕКТИ РОЗРОБКИ ВЕБ-САЙТІВ 19](#_Toc75351251)

[2.1 Етапи розробки сайту 19](#_Toc75351252)

[2.1.1 Бібліотека jQuery. 21](#_Toc75351253)

[2.1.2 Мова розмітки гіпертексту HTML 23](#_Toc75351255)

[2.1.3 Каскадні таблиці стилів SCSS 24](#_Toc75351256)

[2.1.4 Мова програмування TypeScript 25](#_Toc75351257)

[2.1.5 Backend-фреймворки в PHP 27](#_Toc75351258)

[2.1.6 Laravel 29](#_Toc75351259)

[2.1.7 Yii 30](#_Toc75351260)

[2.1.8 Symfony 31](#_Toc75351261)

[2.1.9 Порівняння Laravel, Yii та Symfony 32](#_Toc75351262)

[2.1.10 Обґрунтування вибору Lavarel 33](#_Toc75351263)

[2.1.11 Веб-сервер Open Server 34](#_Toc75351264)

[2.1.12 Проектування Баз Даних 35](#_Toc75351265)

[2.1.13 Сервер баз даних MySQL 37](#_Toc75351266)

[2.2 Висновки до другого розділу 38](#_Toc75351267)

[РОЗДІЛ 3 ПРАКТИЧНА ЧАСТИНА РОЗРОБКИ ІНФОРМАЦІЙНО-НАВЧАЛЬНОГО САЙТУ 40](#_Toc75351268)

[3.1 Розробка інтерфейсу веб-сайту 40](#_Toc75351269)

[3.2 Інструменти розробки веб-сайту 41](#_Toc75351270)

[3.3 Структура інформаційно-навчального сайту та опис розділів 42](#_Toc75351271)

[3.3.1 Дизайн інформаційно-навчального сайту 43](#_Toc75351272)

[3.3.2 Головна сторінка 46](#_Toc75351273)

[3.3.3 Реєстрація та авторизація 47](#_Toc75351274)

[3.3.4 Сторінка «Відео» 48](#_Toc75351276)

[3.3.5 Форма «Забронювати місце» 49](#_Toc75351277)

[3.4 Панель адміністратора 50](#_Toc75351278)

[3.4.1 Інтерфейс 51](#_Toc75351279)

[3.4.2 Додавання контенту 52](#_Toc75351280)

[3.5 Тестування веб-сайту 53](#_Toc75351281)

[3.6 Встановлення та введення в експлуатацію 54](#_Toc75351282)

[3.7 Висновки до третього розділу 57](#_Toc75351283)

[ВИСНОВКИ 59](#_Toc75351284)

[СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ 60](#_Toc75351285)

[ДОДАТОК А ВИХІДНІ КОДИ РЕСУРСУ — FRONTEND 62](#_Toc75351286)

[ДОДАТОК Б ВИХІДНІ КОДИ РЕСУРСУ — BACKEND 70](#_Toc75351287)

# ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СКОРОЧЕНЬ І ТЕРМІНІВ

|  |  |  |
| --- | --- | --- |
| **SCSS** | — | Sassy Cascading Style Sheets |
| **HTML** | — | Hyper Text Markup Language |
| **MySQL** | — | система керування реляційними базами даних |
| **PHP** | — | Hypertext Preprocessor |
| **БД** | — | База даних |
| **ОС** | — | Операційна система |
| **ПК** | — | Персональний комп’ютер |
| **СУБД** | — | Система управління базами даних |

# ВСТУП

В останні роки Інтернет стає все більш популярним, оскільки кількість користувачів він швидко зростає. Знайти цікаву для користувача інформацію більше не проблема, так як кількість сторінок росте з кожним днем.

Створення веб-сайтів — популярний бізнес, тому багато людей працюють над його простотою і зручністю. Важливою частиною створення і розвитку сучасних веб-проектів є створення панелі адміністратора. За його допомогою можна додавати не тільки текстовий контент, але і різні мультимедійні матеріали. Панель адміністратора дозволяє урізноманітнити сайт без втручання програмістів.

В даний час, кількість освітніх Інтернет-ресурсів зростає, і через це все важче залучити покупця до певного веб-сайту. Тому був винайдений зворотній зв'язок, щоб утримувати аудиторію і вирішувати її проблеми. Це дозволяє потенційному покупцеві навчальних матеріалів відправити електронний лист на веб-сайт, який може бути переглянутий власником веб-сайту або співробітником, відповідальним за спілкування з клієнтами.

Тема дипломної роботи - створення веб-ресурсу на базі бібліотеки jQuery з підтримкою зворотного зв'язку. Актуальність теми заснована на високій інтеграції веб-ресурсів на цих фреймворків з необхідними мережевими і системними рішеннями, які використовуються в Інтернеті. Крім того, завдяки підтримці зворотного зв'язку покупець може висловити свої побажання щодо контенту та його можливих оновлень в майбутньому.

Кінцева мета дослідження - підтримати зворотний зв'язок з клієнтами для поліпшення надання послуг в дослідженні і поглибити знання в області проектування, конструювання та розробки влаcного інтернет курсу по SEO просуванню за допомогою веб-ресурсу.

Для досягнення цієї мети вам необхідно вирішити такі проблеми, як:

* Аналіз існуючих фреймворків, їх можливостей і властивостей.
* Підбір каркасів виходячи з вимог до розташування.
* Розробка дизайну сайту, на який буде оформлений шаблон.
* Створення та дизайн бази даних.
* Написання коду для створення інформаційних та освітніх функцій сторінки, використовуючи мови програмування PHP і TypeScript, а також мову розмітки гіпертексту HTML і каскадні таблиці стилів SCSS.
* Зворотний зв'язок по електронній пошті.
* Тестування сайту з контентом.
* Налаштування віддаленого хостингу для розміщення цього сайту.

**Об’єкт дослідження:** процес управління контентом Інтернет-ресурсу.

**Предмет дослідження:** інформаційно-навчальний веб-сайт для Школи Інтернет-маркетингу на прикладі онлайн курсу по SEO просуванню використовуючи бібліотеку jQuery з підтримкою зворотного зв'язку із замовниками.

**Метод дослідження:** аналітичний по використанню комп'ютерних технологій.

Практична цінність роботи полягає в Інформаційно-освітньому веб-сайту, розробленому для школи Інтернет-маркетингу, допомагає користувачам легко навчитися SEO просуванню та сучасним цифровим професіям.

Структура диплому логічно заснована на цілях дослідження і містить список скорочень, умовні позначення, вступ, три розділи основної частини, висновки, список джерел, додатки.

# РОЗДІЛ 1

# АНАЛІЗ ПРОБЛЕМИ І ПОЯСНЕННЯ ЗАВДАННЯ ДОСЛІДЖЕННЯ

##### Обґрунтування необхідності веб-сайту

Інформаційні та освітні сторінки стають все більш актуальними. Напевно, кожна друга людина робила підписку на веб-сайти, де фахівці розміщують свої власні навчальні відео. Щодня викладаються сотні відеороликів, в яких автори діляться необхідними секретами і власними поглядами на ту чи іншу справу.

В останні кілька років продаж навчальних матеріалів в Інтернеті зростає і користується все більшою популярністю. За допомогою навчальної сторінки у продавця є можливість продати будь-який інформаційний ресурс. А для покупця це відмінний спосіб відточити свої навички, використовуючи його в якості керівництва, відео або статті.

Наявність власних ресурсів в мережі Інтернет сьогодні має велике значення для провідних фахівців в різних областях. Заява Б. Гейтса: «Якщо ти не маєш мережі, ти не при справах» актуально і сьогодні.

Створення інформаційної та навчальної сторінки сприяє досягненню наступних цілей:

* Отримайте вигоду від продажу матеріалів, заснованих на досвіді і навичках фахівця.
* Формування іміджу продавця інформаційних ресурсів. Наявність сайту - необхідний атрибут фахівця, який дає замовнику перше враження про його навичках і говорить про його професійний рівень і досвід в описаній на сайті області.
* Допомагає знайти нових клієнтів. Сайт принесе потенційним клієнтам більше інформації про фахівця, ніж будь-яка реклама.
* Підвищення ефективності рекламних інформаційних ресурсів. Сайт, ймовірно, є єдиною рекламною можливістю, доступною клієнтам 24 години на добу, 7 днів на тиждень, 365 днів на рік. Сайт буде працювати 24/7 і не вимагає великих і регулярних вкладень.
* Дає можливість спілкуватися з відвідувачами і покупцями [1].

##### 1.2 Класифікація сайтів

У сучасному світі мільйони веб-сайтів розміщені у всесвітній павутині, і їх число постійно зростає. Їх класифікація варіюється в залежності від параметрів, але найбільш важливим з них є вид діяльності.

* Сайт-візитка — це сайт, що складається з декількох сторінок, які містять інформацію про діяльність компанії або конкретної людини, послуги та контактну інформацію для потенційних клієнтів. Також є можливість розмістити історію компанії, заробіток, діяльність, фото і т. ін. Такий сайт являє собою повноцінний ресурс з унікальним оформленням і дизайном.
* Корпоративний сайт — це повноцінне уявлення вашої компанії в мережі Інтернет. Це необхідний інструмент, який допомагає успішно вести будь-який бізнес. Мета створення веб-сайту компанії — надати інформацію про компанію, продукти чи послуги тим, в яких компанія зацікавлена ​​як потенційні партнери або споживачі.
* Каталог (вітрина) сайту створений для продажу товарів. На його сторінках необхідно розмістити інформацію, яку покупець повинен побачити, зацікавити і купити певний товар. Ці сторінки містять інформацію про продукти або послуги, які не можуть бути включені в прайс-лист, наприклад докладний опис, ціну та фотографії.
* Рекламна сторінка — це в першу чергу інструмент для просування певної послуги або продукту або навіть заходи. Ці сторінки містять інформацію про бренд, різні акції. На сторінці не повинно бути багато текстового контенту, вона повинна мати легкий дизайн, що поєднує графіку і можливість інформувати клієнта про послугу або продукт та запам'ятовуватися в формі.
* Сторінка брошури, ймовірно, найпоширеніша, оскільки вона містить майже ті ж матеріали, що і звичайні брошури. Ваша мета - просувати компанію.
* Сайти електронної торгівлі, де покупець може розмістити замовлення, вибрати спосіб оплати, спосіб доставки або отримати рахунок на оплату. І все це, не відходячи від комп'ютера [2].

Інформаційно-педагогічна частина моделює інформаційно-педагогічне середовище з можливістю структурування глибини вивчення, елементами автоматизованого контролю та консультування в разі виникнення ускладнень, можливістю швидкого пошуку наукового матеріалу, а також блоком зворотного зв'язку з розробниками системи і викладачами.

##### 1.3 Переваги інформаційного та навчального сайту

З швидким розвитком електронної комерції та технологій більшість людей розуміють переваги наявності інформаційного та навчального сайту. Споживачі також зацікавлені в віртуальних навчальних платформах, тому що ці послуги прості у використанні і блоки інформації можна придбати в будь-якій точці світу.

Алгоритм навчання сторінок дуже простий:

1. замовник заходить на сайт;
2. вибирає потрібний вам інформаційний блок;
3. розміщує замовлення;
4. здійснює оплату;
5. отримує доступ до платного блоку.

Незаперечний плюс таких придбань - величезна економія часу. Крім того, необхідно виділити наступні переваги інформаційно-навчального сторінки:

* Немає необхідності залишати квартиру.
* Навчальні матеріали доступні завжди, незалежно від часу і місця.
* Все, що вам потрібно, це доступ в Інтернет.
* Інформація оновлюється швидко і легко.
* Індивідуальна обробка матеріалу замовником формує дисципліну, організованість, уміння планувати час.
* Замовники освоїли можливості інформаційних технологій.
* Розвиває почуття відповідальності, ініціативу і вміння ставити цілі.
* Дає вам можливість адаптувати процес навчання до ваших потреб і побажань.
* Використання технології електронного навчання направлено на вивчення особистості, виявлення навичок і здібностей учня.
* Низькі витрати на навчання при високому рівні знань.
* Незалежність від погодних умов і часу доби.
* Можливість зручно розплачуватися картою.
* Немає необхідності спілкуватися з іншими і вчитися з іншими учнями.

##### 1.4 Постановка завдання

Правильність поставленого завдання — запорука досягнення кінцевої мети.

Перед розробниками сайту ставляться різні завдання, спрямовані на досягнення конкретних цілей. Наприклад, ось деякі з них:

* Розробка дизайну з урахуванням стилю клієнта.
* Переконайтеся, що інформація розробляється або оновлюється адміністратором.
* Дозволити адміністратору видаляти застарілі дані в базі даних.
* Створення зручного ресурсу — вибір, покупка і використання товарів або послуг.
* Для досягнення визначеної середньої плинності необхідно залучити певну кількість відвідувачів на сторінки навчального сайту.
* Дайте зареєстрованому користувачу логін та пароль для створення замовлення.
* Розміщення актуальної інформації.
* Мотивація відвідувачів сайту до покупки блоків інформації або безпосередньо через покупку на сторінках сайту, або по телефону власника сайту або відповідального за це співробітника.
* Дотримання безпеки та конфіденційності: реєстрація покупців, створення покупки, контроль платежів.

На першому етапі потрібно правильно визначитися з метою і завданнями сайту, виділивши основні з них. Правильна постановка завдання і розуміння кінцевої мети — ключі до якості і очікуваного результату.

##### 1.5 Поліпшення інформаційного та навчального сайту

Не всі роблять щось ідеальне з першого разу. І за рахунок інформаційних та освітніх сайтів цей вердикт досить справедливий — поліпшити ніколи не пізно.

Така функція, як зворотний зв'язок з клієнтами, може бути основою для зміни або поліпшення пропозиції. Іноді компанія занадто впевнена у своєму продукті і не вносить жодних змін.

Щоб постійно покращувати сервіс, потрібно визнавати свої помилки і вирішувати проблеми. Кожен раз, коли є зворотний зв'язок, є можливість дізнатися щось позитивне або негативне від вашого клієнта. Репутація — це в першу чергу результат зворотного зв'язку. Здебільшого, особливо через такі сайти, як eBay, зворотний зв'язок став обов'язковою частиною їхнього успіху.

Отже, вам необхідно включити зворотний зв'язок, щоб підвищити продуктивність. Це важливий інструмент для прийняття рішень. Відгуки клієнтів допомагають виправити недоліки і привести до появи нових послуг і продуктів. Рішення часто ґрунтуються не на інтуїції чи навіть чистому судженні, а на ринку. Нові можливості для бізнесу все ще потребують ретельного вивчення.

Іноді буває складно знайти відгуки. І у кожної компанії є свої «кращі практики». У гастрономії, наприклад, діє правило: «Ти хороший, як останній прийом їжі». Але як дізнатися про це у ресторатора? Гість нечасто говорить, що йому не подобається. Він просто більше сюди не ходить. І його друзі теж не прийдуть пізніше. Чутка передається по низхідної спіралі, і власник не має поняття, у чому причина її появи. Відповідь на це питання — зворотний зв'язок. Вони могли дивитися, як тарілки повертаються на кухню, або роздавати листівки з безкоштовним десертом для чесного огляду.

Запросити зворотний зв'язок — найкращий спосіб його отримати. Ось кілька варіантів:

* Зробити це зручно і легко для замовника.
* Відповідайте швидко і з вдячністю.
* Покупці в основному звертають увагу на відгуки, які адресовані комусь, а не організації.
* Запитуйте пропозиції. Нехай першим питанням буде: «Що ми можемо зробити краще?»

Часто відгуки представляють собою позитивну, негативну або нейтральну оцінку, а також включають короткий коментар. Клієнти надають важливу інформацію про можливі стилях, перевагах або вимогах. Або вони можуть зробити пропозиції по інших продуктах, які вони хотіли б бачити. Наприклад, в той час, коли Jelly Belly набирав популярність, дівчина запитала власницю сувенірного магазину Діну Джонсон, чому вона не продає його. Діна прийняла це зауваження і стала продавцем Jelly Belly, який протягом багатьох років привертав в магазин безліч покупців [3].

##### 1.6 Використання зворотного зв'язку

Ось кілька способів отримати зворотний зв'язок:

* Запитайте самих клієнтів. Запитайте їх, що їм подобається, чим вони незадоволені і що вони хотіли б змінити.
* Створіть опитування. У компаніях існує безліч областей оцінки.

Отримати важливий зворотний зв'язок може бути складно. Так що задля нього можна запропонувати щось цінне, іноді навіть гроші.

Змагаючись з успішним і великим конкурентом, зазвичай потрібно брати до уваги практику, яку вони вже зрозуміли шляхом спроб і помилок. Гарною відправною точкою для зворотного зв'язку є не винахід колеса самостійно, а використання вже встановлених методів.

Програма зворотного зв'язку повинна бути основною частиною компанії для відстеження тенденцій, смаків, проблем і потенційних загроз з боку нових конкурентів:

* Визначте, які системи зворотного зв'язку найкраще підходять для вашої компанії.
* Призначте представника для перегляду відгуків.
* Звіт про зворотний зв'язок, що містить поліпшення послуги або продукту [4].

Очевидна перевага зворотного зв'язку полягає в тому, що вона дає вашим потенційним клієнтам почуття впевненості. Велика кількість переглядів робить покупця більш впевненим у правильності свого рішення.

Відгуки допоможуть вам краще зрозуміти ваші продукти. Ваші клієнти можуть надати вам успішні ідеї для поліпшення або навіть нові маркетингові ідеї.

Основні конкуренти пропонують легкий доступ до зворотного зв'язку на своїх онлайн-сайтах. Наприклад, Apple і Cisco публікують посилання для пошуку проблем і нових ідей. У Apple є поле «Тип зворотного зв'язку», в якому ви можете вибрати найбільш відповідна нагода.

В Інтернеті відгуки стали джерелом інформації, коли покупець купує (або не купує) в вашому інтернет-магазині. Є багато інструментів зворотного зв'язку. Кожен раз, коли клієнт вирішує купити товар або послугу, у нього є можливість легко і зручно обмінюватися досвідом.

З інтуїтивним і ненав'язливим зворотним зв'язком, ви можете взаємодіяти більш глибоко зі своїми клієнтами і дізнатися, що люди думають про твій сайт. Онлайн-інструменти можуть допомогти збільшити продажі, запитуючи зворотний зв'язок в критичні та болючі моменти. У більшості випадків, наприклад, на eBay, зворотний зв'язок є необхідною частиною того, що зробило ці веб-сайти успішними.

Розробники eBay створюють профіль зворотного зв'язку на основі оцінок і коментарів інших учасників. Нерідко вони складаються з нейтральних, позитивних або негативних, а також коротких коментарів.

Якщо ви встановлюєте зв’язок в приватному порядку, відповідь має бути опублікована з урахуванням про те, як спілкуватися з клієнтом - по телефону або електронній пошті. Це допомагає клієнтам зрозуміти, що ваша компанія несе відповідальність за проблеми, і може підвищити їх довіру.

Уточнюйте інформацію, вибачайтеся за помилки і дякуйте клієнтам за те, що вони витратили свій час на те, щоб зв'язатися з власником сайту. Це хороша ідея — відповідати на кожен запит і не поспішати, щоб показати, що вам не все одно, що думають клієнти. Але є одна проблема з похвалою і критикою, а саме: ефективність залежить від особистої думки. З іншого боку, інформативний зворотний зв'язок відноситься до конкретних питань та інформації. Ось інструкція по конструктивній торгівлі:

* Зосередьтеся безпосередньо на конкретних темах.
* Уникайте повідомлень з описом проблеми, навіть якщо немає інформативних питань.
* Коли похвала доречна, її можна віднести до конкретних досягнень.

##### 1.7 Панель адміністратора

Адміністративна область системи управління контентом називається просто панеллю керування сайтом. Адміністратор цієї панелі має необмежений доступ до всіх функцій. Редактори, автори та інші користувачі — це користувачі з іншим рівнем доступу, тобто вони мають обмежений доступ до панелі адміністратора. У більшості користувачів немає доступу.

Ряд інструментів, що забезпечують доступ до інших функцій управління, розташованих у верхній частині панелі адміністратора. Кожен розділ часто розбивається на підрозділи, в яких показані додаткові параметри.

Робоча область — це область в середині екрану, в якій ви можете писати, видаляти і змінювати записи і налаштування.

Використання адміністративної панелі дає ряд переваг:

* Вам не потрібно постійно просити розробників скорегувати дані.
* Часте оновлення інформації, можливості розвитку ресурсу, додавання нових даних. Це необхідно для навчально-інформаційного сайту.
* Додавання нових сторінок, розширення сайту, наповненого унікальним свіжим контентом. Якщо вам необхідно використовувати унікальний контент для розробки рекламної стратегії для підвищення рівня сайту в пошуковій системі, без адміністративної панелі вам не обійтися [5].

##### 1.8 Висновки до першого розділу

У першому розділі роботи обґрунтовувалася необхідність створення інформаційно-освітнього сайту зі зворотним зв'язком. На підставі отриманих даних визначено переваги веб-ресурсу — розміщення «електронної інформації» дозволяє власникові продавати свої знання і навички, а за допомогою відгуків можна поліпшити сайт і тим самим привернути більше покупців в майбутньому.

Підводячи підсумок, можна сказати, що розробка веб-системи для навчального сайту з підтримкою зворотного зв'язку актуальна, оскільки як покупці, так і продавці інформаційних послуг економлять багато ресурсів і часу.

# РОЗДІЛ 2

# ТЕОРЕТИЧНІ АСПЕКТИ РОЗРОБКИ ВЕБ-САЙТІВ

##### 2.1 Етапи розробки сайту

Основні етапи розробки сайту:

* Дизайн (з урахуванням вимог замовника до сайту, призначенням, основним завданням, аудиторії і т. ін.);
* Планування (створення карти сайту, що дозволяє клієнту оцінити, як буде виглядати весь сайт);
* Дизайн веб-сайту (на цьому етапі створюються шаблони і весь візуальний контент, наприклад, на зображеннях, фото і відео);
* Розкладка шаблонів після того як клієнт стверджує дизайн сайту (дизайн конвертується в реальні динамічні інтерактивні елементи на сторінці сайту);
* Програмування. Частково розробляється паралельно з дизайном і версткою (функціональна розробка);
* Контент сайту (розміщення контенту);
* Тестування, налагодження та повторне тестування;
* Публічне відкриття проекту (розміщення сайту на сервері).

Також є додаткова робота, так як замовник рідко припиняє співпрацю після отримання сайту. Часто замовляє додаткові послуги, без яких йому складно обійтися наприклад:

* Навчання роботі з сайтом співробітника компанії.
* Оновлення вміст сайту.
* Адміністративна підтримка сайту.
* Видобуток ресурсів.
* Написання текстів, статей і т. ін..

Створення сайту може здатися складним, але якщо ви будете дотримуватися всі необхідні пункти, проблем не буде [6].

**2.2 Інструменти розробки cайту**

Інформація, доступна користувачам мережі Інтернет, зберігається на комп'ютерах (серверах), на яких встановлено спеціальне програмне забезпечення. Велика частина цієї інформації організована у вигляді веб-сайтів.

У великої кількості сайтів є частина для користувача і серверна частина. Клієнтська частина цього сайту заснована на бібліотеці jQuery, який, в свою чергу, включає розмітку HTML, стилі SCSS і мову програмування TypeScript. Серверна частина заснована на фреймворку Laravel, написаному на PHP і бази даних MySQL.

jQuery — це невелика, легка і швидка бібліотека JavaScript. Вона багатоплатформова і підтримує різні типи браузерів. Його також називають «пиши менше, роби більше», тому що вона бере на себе безліч загальних завдань, для виконання яких потрібно багато рядків коду JavaScript, які вона пов'язує в методи, які можна викликати за допомогою одного рядка коду, коли це необхідно. Також дуже корисно спростити багато складних речей з JavaScript, такі як виклики AJAX і маніпуляції з DOM.

Laravel є відкритим вихідним кодом призначений для спрощення і прискорення розробки веб-додатків за допомогою вбудованих функцій. З PHP на стороні сервера, дані, що надходять з клієнтської частини обробляються і бази даних обробляються.

MySQL - одна з найпопулярніших і широко використовуваних баз даних (система управління базами даних) в Інтернеті з відкритим вихідним кодом, яка є дуже швидким і надійним сервером для баз даних. Сервер MySQL розроблений як для критично важливих виробничих систем з високим ступенем завантаження, так і для вбудовування в програмне забезпечення масового поширення [14].

2.1.1 Бібліотека jQuery. JQuery - Бібліотека JavaScript, що містить вбудовані функції JavaScript, всі операції jQuery виконуються з коду JavaScript.

Бібліотека jQuery управляє елементами HTML, контролюючи їхню поведінку і використовуючи DOM для зміни структури веб-сторінки. Вихідні файли HTML і CSS не змінюються, зміни вносяться тільки в те, як сторінка відображається для користувача.

Селектори CSS використовуються для вибору елементів. Вибір здійснюється за допомогою функції $ (). Коли викликається функція $ (), вона повертає новий екземпляр об'єкта JQuery, який повертає нуль або більше елементів DOM і дозволяє взаємодіяти з ними різними способами.

Різні сценарії можуть виконуватися тільки після завантаження структури документа, коли браузер перетворює HTML-код сторінки в дерево DOM. Процес завантаження контролюється дизайном:

jQuery(документ).готові(функція() {

...

});

Спочатку екземпляр документа включається у функцію jQuery (), за якою слідує метод ready (), якому передається функція () {...}, яка виконується після завантаження документа.

На практиці короткою формою такого запису зазвичай є jQuery (function () {...}); або $ (function () {...});.

Змінні JavaScript використовуються для зберігання інформації при роботі з бібліотекою jQuery. Змінні можуть зберігати елементи. Імена змінних, які повинні зберігати повертаються елементи, починаються з символу $, наприклад:

$ ч знак дорівнює $(".перелік").батько().батько().відклеюватися();

Масиви JavaScript використовуються для зберігання кількох елементів:

$ k[3] знак дорівнює п'ятнадцять;

Основні переваги jQuery:

* Кросбраузерність (можливість роботи в будь-якому інтернет-браузері)
* Збільште швидкість написання програми
* Доступність освітньої інформації в Інтернеті
* Використовується мало пам'яті

Способи підключення jQuery

По-перше, необхідно завантажити бібліотеку з сайту виробника. Часто це рекомендується робити там, а не на інших ресурсах. Вам потрібно вибрати останню версію. Наприклад, якщо сайт розробляється локально, після збереження на диск ви можете підключитися за допомогою команди:

<Script type = "text / javascript" src = "js / jquery-1.8.2.min.js"> </ script>

// src = "js / -" Це папка, в якій зберігається бібліотека

// jquery-1.8.2.min.js - Це бібліотека, завантажена безпосередньо

Якщо розробляється сайт знаходиться на сервері, то і бібліотека повинна бути розміщена там же. Команда підключення принципово не змінюється.

Ви також можете приєднатися до бібліотеки, без скачування її. Всі чули про хмарних просторах. Бібліотеку JQuery також можна зберігати в хмарному сховищі, такому як Google CDN. Підключення в цьому випадку не становить особливих труднощів, досить ввести шлях до сервера у відповідному рядку.

приклад:

<Script src = "// ajax.googleapis.com/ajax/libs/jquery/3.1.0/jquery.min.js"> </ script>

// jquery / 3.1.0 / - це версія бібліотеки.

2.1.2 Мова розмітки гіпертексту HTML. Мова гіпертексту або HTML-розмітки — це стандартна мова розмітки для документів, що відображаються в веб-браузері. HTML визначає вміст і структуру вмісту сайту, тому він відповідає за те, як буде виглядати конкретний елемент сторінки.

Мова гіпертексту надає тільки інформацію для читання. Це означає, що редагувати сайт може тільки людина, що створила сайт, а не звичайний користувач.

Документ HTML складається з наступних частин: даних, що складають вміст документа, і тегів — спеціальних конструкцій мови HTML, необхідних для розмітки документа. Теги визначають, як текст представлений і які з його компонентів функціонують як гіперпосилання і які графічні або мультимедійні об'єкти повинні бути включені в документ.

Досить часто теги використовуються попарно. Пара складається з відкриваючого тега <tag name> і закриває тега </ tag name>. Кожен парний тег починається там, де знаходиться відкривачий тег, і закінчується там, де знаходиться відповідний закриваючий тег. Часто таку пару називають контейнером, а частина тексту, яка обведена їм з обох сторін, називається елементом.

Простий HTML-документ виглядає так:

<Html>

<Head>

<Title> Ім'я </ title>

</ Head>

<Тіло>

<P> Тіло документа </ p>

</ Body>

</ Html>

В даний час існує безліч варіантів створення сторінок сайту, що мають свої позитивні і негативні характеристики. Але ось основні переваги HTML:

* Привабливий зовнішній вигляд;
* Швидке завантаження сторінки;
* Адаптивність, універсальність і гнучкість;
* Пошукова оптимізація;
* Легке створення;
* Легкість використання і зручна навігація.

Головна перевага HTML є те, що сторінки сайту відображається на різних платформах і в Інтернеті-Браузер різних типів [7].

2.1.3 Каскадні таблиці стилів SCSS. SCSS (Sassy Cascading Style Sheets) — це мова, призначена для спрощення створення коду CSS. Тобто SCSS — це мова, код якого транслюється в звичайний код CSS спеціальною програмою, написаною на Ruby. Весь код SCSS скомпільовано в CSS, щоб браузери могли його розуміти і правильно відображати. Браузери в даний час не підтримують SCSS.

SCSS — це «діалект» мови SASS. Його синтаксис досить гнучкий, він враховує багато деталей, які бажані в CSS. Крім того, в ньому є логіка (@if, any) і математика (ви можете додавати лінії, числа і кольору).

Ось кілька причин використовувати SCSS замість CSS:

* Вкладеність — SCSS дозволяє вбудовувати правила CSS один в одного.
* Змінні — є також поняття змінних в стандартній CSS, але ви можете працювати з ним по-різному в SCSS. Наприклад, динамічне створення властивостей.
* Методичні рекомендації @ for, @ while і вираз @ if-else — ви можете писати код CSS з елементами з інших мов.
* Функції — ви можете створювати багаторазові визначення CSS як функції.
* Тригонометрія — SCSS дозволяє писати свої власні функції.

SCSS визначає відображення даних в браузері. Якщо формат HTML надає інформацію про структуру документа, то таблиці стилів показують, як він повинен виглядати. Стиль включає в себе всі види елементів дизайну: текст, фон, шрифт, поля, колір посилань і положення об'єктів. SCSS забезпечує необхідний рівень однаковості в дизайні, управлінні і організації при розробці сайту.

Ви завжди повинні дотримуватися прийнятого стилю. Якщо сайт має велику структуру, його сторінки повинні бути розроблені таким же чином, тобто в тому ж стилі. На всіх сторінках повинні бути однакові шрифти, елементи управління та інтервали [8].

2.1.4 Мова програмування TypeScript. TypeScript (TS) — це мова програмування з відкритим вихідним кодом, заснований на JavaScript (JS). TS пов'язаний з JS, тому що він компілюється з ним. Це означає, що код, написаний на TypeScript, автоматично перетворюється в JavaScript.

Розробка TypeScript почалася в кінці 2012 року. Хоча він був розроблений Microsoft, а фактичним творцем є програміст Андерс Халесберг, цей проект відразу ж почав розвиватися як OpenSource. І з самого початку нова мова швидко поширилась, бо вона гнучка і продуктивна. На TypeScript переведено велику кількість проектів, написаних на JavaScript. Ідеї ​​нової мови популярні і актуальні, тому деякі з цих ідей пізніше стануть частиною нового стандарту JavaScript.

Але навіщо клієнтській стороні в веб-середовищі потрібна інша мова програмування, якщо та ж сама робота зроблена добре, а звичайний JavaScript, що використовується майже на кожному сайті, має багато розробників і досить добре підтримується. Але TypeScript — це не просто новий JavaScript.

Перш за все, слід зазначити, що TypeScript — це скомпільований і строго типізована мова, до якої можуть бути ближче програмісти Java, C# та інших строго типізованих мов. Компілятор виводить той же JavaScript, який пізніше буде виконуватися браузером. Однак сувора типізація знижує кількість потенційних помилок, які можуть виникнути при розробці з використанням JavaScript.

По-друге, TypeScript реалізує багато концепції, властиві об'єктно-орієнтованим мовам, такі як інкапсуляція, успадкування, модифікатори доступу, поліморфізм і т. ін.

По-третє, потенціал TypeScript спрощує і прискорює написання великих і складних програм, а також спрощує обслуговування, тестування, масштабування і розробку, ніж зі звичайним JavaScript.

По-четверте, TypeScript розвивається як проект з відкритим вихідним кодом і, як і багато інших проектів, розміщується на GitHub. Крім того, TS є кросплатформним, що означає, що ви можете використовувати для розробки як Windows, так і MasOS або Linux.

У той же час TypeScript — це надмножина JavaScript, що означає, що кожна програма на JS також є програмою на TypeScript. В TS можна використовувати ті ж конструкції, що і в JS — ті ж оператори, циклічні, умовні конструкції. Крім того, код TS компілюється на JavaScript. В кінцевому підсумку TS — це просто інструмент, призначений для прискорення розробки програмного забезпечення.

TypeScript, згенерований компілятором в коді JS, який підтримується переважною більшістю браузерів, оскільки він в основному орієнтований на стандарт ECMAScript 3, але TS також підтримує інші стандарти ECMAScript 5 і ECMAScript 2015/2017, які ми можемо вказати нам під час розробки.

Як ви використовуєте TypeScript? Оскільки це мову з відкритим вихідним кодом, його інструменти доступні кожному. Ви можете використовувати як Windows, так і Linux і macOS для роботи з TS.

Сам компілятор TS встановлюється за допомогою команди диспетчера пакетів npm, використовуваної в Node.js:

npm install -g typescript

Ви можете використовувати будь-який простий текстовий редактор для написання коду TypeScript. Існує безліч текстових редакторів і середовищ розробки, таких як Sublime, Visual Studio, WebStorm, Visual Code Studio, Atom, Netbeans і інші, які підтримують TypeScript на рівні модуля, що дає такі переваги, як виділення коду або спливаючі підказки [9].

2.1.5 Backend-фреймворки в PHP. У світі розробки програмного забезпечення термін «фреймворк» відноситься до файлових бібліотекам, які містять кілька основних функцій. Мета структури - надати структуру, яка буде використовуватися для більш ефективної розробки проекту, щоб не писати код з нуля. Були створені рамки для структурування і полегшення процесу розробки. Його універсальність полягає в тому, що він об'єднує в одному кадрі різні компоненти, визначені в проекті.

Каркаси в PHP (простий, але і задовольняє мова за професійним програмування запитів) є найбільш популярними з них доступні для початківців. В даний час багато розробників використовують PHP-фреймворки для своїх проектів. Основними причинами цього є:

* Завдяки готовому кадру процес розробки на цій платформі відбувається набагато швидше.
* Фреймворк спочатку визначає структуру програмної системи, яка допомагає писати структурований код, прийнятний для повторного використання.
* Платформа дозволяє масштабувати проект і вносити зміни.
* Більшість фреймворків пропонують високий рівень безпеки.
* Веб-проекти, розроблені на цих фреймворків, гарантують швидке завантаження.

Каркаси працюють так само і позбавляють вас від необхідності винаходити велосипед. При розробці нового PHP-додатки найкраще спочатку знайти фреймворк, який має всі необхідні вам функції. На щастя, є з чого вибрати. Всі вони унікальні в своєму підході до розробки, і у кожного є свої переваги і недоліки.

Який фреймворк PHP найкраще залежить від обсягу і цілей проекту. Вибираючи таку платформу, варто звернути увагу на ряд основних показників.

* Наскільки широкий і гнучкий функціонал фреймворка. У ньому є все необхідне для поточного проекту.
* Чи легко освоїти фреймворк. З огляду на рівень розробника.
* Швидкість і сучасність — наскільки фреймворк орієнтований на передові методи програмування.
* Як масштабувати проект, створений у фреймворку.
* Як часто випускаються оновлення.
* Чи є у фреймворка служба довгострокової підтримки релізів (LTS)?
* Чи є підтримка архітектури MVC (Model View Controller).

Фреймворк PHP — ідеальний інструмент для створення API-інтерфейсів і повнофункціональних веб-додатків, що працюють на стороні сервера. Ви економите час з готовими шаблонами і рішеннями.

Вибір каркаса залежить від складності проекту. Чим він більший, тим більше потрібно ресурсів і технологій. Для невеликих проектів можна вибрати універсальні рішення з оптимальним балансом ключових компонентів і без потреби в ресурсах [10].

2.1.6 Laravel. Ця безкоштовна платформа є однією з найпопулярніших PHP-фреймворків через її активного розвитку і простоти розробки. Новачки, вперше вирішили його освоювати, отримують комфортні умови для покрокового занурення в сферу веб-розробки для самих різних завдань: від звичайних рутинних завдань до складних проектів по інтеграції і масштабування додатків. Завдяки цій розширеній функціональності Laravel допоможе вам впоратися з безліччю серверних завдань, навіть не володіючи достатнім досвідом. На рисунку 2.1 показаний офіційний сайт Laravel.

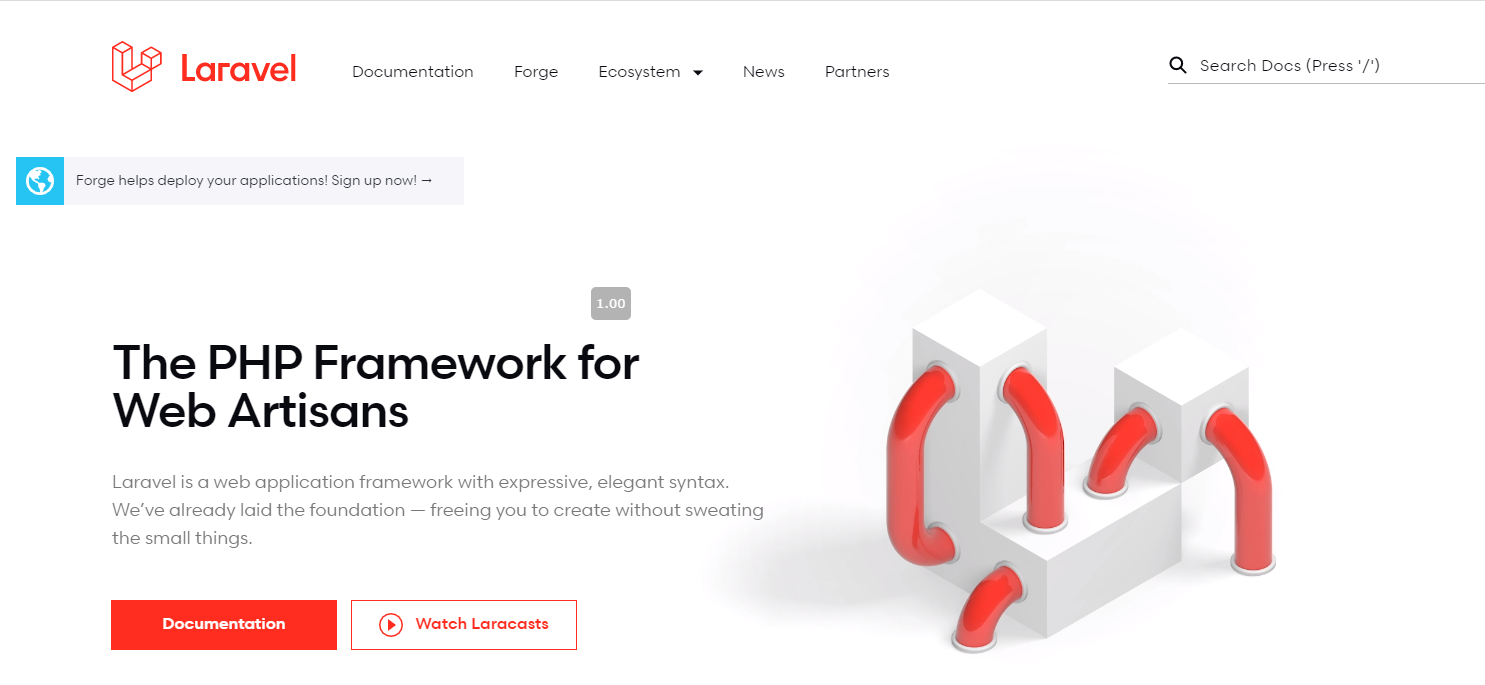


Рис. 2.1 — Офіційний сайт Laravel

Властивості Laravel:

* Добре структурована і докладна документація.
* Структура коду заснована на архітектурі MVC.
* Власну консоль ремісника для роботи з основними елементами структури - міграція, дозволу, контролери та моделі.
* Леза власний шаблон.
* Встановіть jQuery і BootStrap. Укладач пакети після установки можна знайти в файлах app.js і app.css.
* Існують валідатори — структури перевірки і дані за певними правилами, які ви можете використовувати для створення власних шаблонів правил.
* Принцип інверсії управління (IoC) успішно реалізований [11].

2.1.7 Yii. Yii — це загальний фреймворк для веб-програмування, тобто його можна використовувати для розробки всіх видів веб-додатків з використанням PHP. Завдяки своїй компонентної архітектури і досить складною підтримки кешування, він найкраще підходить для розробки великих програм, таких як системи управління контентом (CMS), форуми і т. ін. На рисунку 2.2 показана офіційна сторінка Yii.

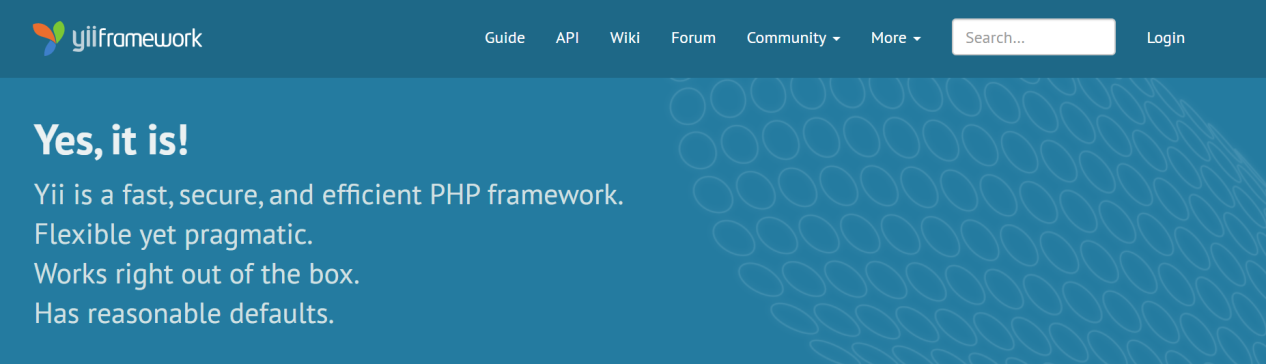


Рисунок. 2.2 — Офіційний сайт Yii

Ключова особливість:

* Гнучкі механізми генерації вихідного коду.
* Повна реалізація парадигми MVC.
* Є функція повного або часткового кешування сторінок.
* Як і Laravel, він підтримує установку за допомогою диспетчера пакетів Composer.

Що стосується мінусів, Yii може бути складним завданням, якщо це ваш перший фреймворк. Це головний конкурент за звання кращого фреймворка PHP, але якщо ви не знайомі з мовою програмування, є інші, більш зручні варіанти для новачків.

Об'єктно-орієнтована PHP-фреймворк Yii вважається лідером з точки зору продуктивності і часто вибирається для високонавантажених додатків. З тих пір, як він був вперше опублікований в 2008 році, структура з роками продовжувала розвиватися. Yii часто потрапляє в топ-фреймворки PHP завдяки високорозвиненому спільноті користувачів, багато з яких є російськомовними [11].

2.1.8 Symfony. Ця структура має гнучку функціональність, засновану на модульному підході до побудови системи. Це дозволяє вибрати необхідні функції мови PHP або керувати всією інфраструктурою в цілому. Цей фреймворк — краще рішення для розробника, який цінує модульність і переваги використання окремих компонентів. На рисунку 2.3 показана офіційна сторінка Symfony.

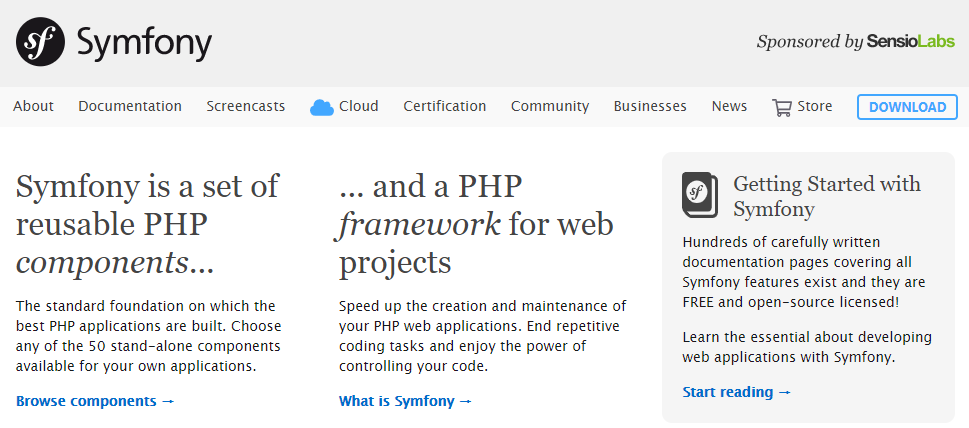


Рисунок. 2.3 — Офіційний сайт Symfony

Можливості Symfony:

* За допомогою власного обробника шаблонів Twig ви можете створити чистий код і додати функціональність PHP.
* Може бути встановлений з Composer.
* Для установки є безліч розширень.
* Сумісність з Codeception спрощує написання тестів.
* Активна спільнота підтримки користувачів.

Плюс в тому, що в платформі є вбудовані тестові функції. Однак додатки, створені за допомогою Symfony, часто не працюють оптимально. Навіть в цій структурі це досить складно для розуміння в порівнянні з Laravel. Symfony може бути хорошим варіантом, якщо вас цікавить модульність [11].

2.1.9 Порівняння Laravel, Yii та Symfony. Laravel, Yii і Symfony — PHP-фреймворки, які використовуються для більш ефективної розробки проектів, щоб не писати код з нуля. Порівняємо фреймворки між собою. Відібрано найважливіші аспекти. Таблиця 2.1 [12] показує порівняння фреймворків.

Таблиця 2.1 — Порівняння Laravel, Yii і Symfony

|  |  |  |  |
| --- | --- | --- | --- |
| 1 | 2 | 3 | 4 |
|  | Laravel | Yii | Symfony |
| Офіційна сторінка | laravel.com | yiiframework.com | symfony.com |
| Опис | Найпопулярніший фреймворк PHP через його активної розробки і простоти розробки. | Yii — найшвидший фреймворк PHP. Також відрізняється безпекою. | Symfony — це набір компонентів PHP, які ви можете використовувати для створення масштабованих і продуктивних додатків. |
| Зручність для користувача | Laravel найлегше вивчити, і більшість посібників доступні в Інтернеті. | Простіше, ніж Symfony, складніше, ніж Laravel. | Найскладніший з трьох фреймворків. |
| Особливості | Має добре структуровану документацію і  Консоль майстра.  Структура коду заснована на архітектурі MVC.  Має власний шаблон клинка. | Yii має гнучкі механізми генерації вихідного коду і повну реалізацію парадигми MVC.  Є функція повного або часткового кешування сторінок. | Він має власний обробник шаблонів Twig і велика кількість розширень, доступних для установки.  Високий рівень підтримки користувачів. |

Продовження таблиці 2.1

|  |  |  |  |
| --- | --- | --- | --- |
| 1 | 2 | 3 | 4 |
| Оптимальне використання | Для малих і середніх проектів. | Для простих проектів. | Для середніх і великих проектів. |

2.1.10 Обґрунтування вибору Lavarel. Оскільки Lavarel є найпопулярнішим фреймворком PHP через його активної розробки і простоти розробки, він був обраний. З Lavarel ви можете швидко почати роботу над своїми проектами. Ви також можете пропустити багато основи, оскільки у вас є доступ до таких функцій, як аутентифікація користувачів, управління сеансами і кешування. Загалом, Laravel має всі функції, необхідні для створення сучасних PHP-додатків.

Якщо подивитись на технічні причини вибору цього фреймворка, їх можна легко резюмувати. По-перше, ядро ​​Laravel надійно в продуктивності, і платформу можна розширити за допомогою безлічі надбудов.

Laravel також добре інтегрується з іншими сторонніми платформами і бібліотеками, тому ви можете створювати великомасштабні програми. Для довгострокових завдань, ви можете поставити їх в чергу, щоб запустити асинхронно у фоновому режимі, що ще більше підвищує продуктивність.

Ключова особливість:

* Структура з елегантним синтаксисом.
* Можливість розширювати функціонал Laravel надбудовами.
* Інтегровані функції для управління маршрутизацією, адміністрування користувачів, кешування і багато чого іншого.
* Інтеграція Laravel зі сторонніми платформами і бібліотеками.
* Завдання виконуються асинхронно у фоновому режимі для підвищення продуктивності.

У Laravel дуже активна спільнота, а це значить, що пошук допомоги або керівництв ніколи не буде проблемою. Якщо ви вперше використовуєте фреймворк, Laravel — ще кращий варіант [12].

2.1.11 Веб-сервер Open Server. Open Server — це безкоштовна портативна програмна платформа, спеціально розроблена для веб-розробників з урахуванням їх побажань і рекомендацій. Програмний пакет включає в себе ряд серверного програмного забезпечення, зручний, багатофункціональний, складний інтерфейс і потужні функції для управління і налаштування компонентів.

Платформа широко використовується для налагодження, розробки і тестування веб-проектів, а також для надання веб-сервісів в локальних мережах. Хоча пов'язані програмні продукти з самого початку не були призначені для спільної роботи, це з'єднання стало дуже популярним серед користувачів Windows, в основному тому, що вони отримали безкоштовний пакет програмного забезпечення з надійністю на рівні сервера Linux.

Комфорт і простота використання, безумовно, є великою перевагою. Open Server виявився першокласним і надійним інструментом, який потрібен кожному веб-майстру.

Розглянемо можливості Open Server:

* кілька режимів управління доменом;
* підтримка адміністрування командного рядка;
* зручне відображення логів всіх компонентів;
* перемикання між модулями HTTP, MySQL і PHP;
* доступ до доменів в один клік.

Відкриті функції сервера:

* не вимагає установки (портативність);
* можливість роботи з флешкою;
* працювати на локальному / мережевому / зовнішньому IP-адресу;
* створіть домен, створивши звичайну папку;
* підтримка альясів (покажчики доменів);
* захист сервера від зовнішнього доступу [13].

2.1.12 Проектування Баз Даних. База даних (БД) — це організований набір структурованої інформації або даних, які зазвичай зберігаються в електронному вигляді в комп'ютерній системі. Дані в найбільш популярних на сьогоднішній день типах баз даних зазвичай моделюються у вигляді рядків і стовпців в серії таблиць, щоб зробити обробку даних і запити ефективними. Після цього ви можете легко отримати доступ, управляти, змінювати, оновлювати, відстежувати і систематизувати свої дані. Більшість баз даних використовують мову структурованих запитів (SQL) для запису і запиту даних.

Реляційна база даних полягає в розробці таблиць і у визначенні відносин між ними.

Модель відносин сутностей або ER-модель — це модель даних, яка надає можливість описувати концептуальні діаграми за допомогою узагальнених блокових конструкцій. Модель ER — це метамодель даних, яка є засобом опису моделей даних, показаних на рис. 2.4.

Laravel використовує кілька пов'язаних таблиць. Між ними існує кілька зв'язків «один до багатьох». Перевірялася нормалізація відносин, щоб виключити дублювання даних, а саме перших 3-х нормальних форм.

Наведемо базу даних, наприклад, за допомогою основної таблиці «users», в якій зберігаються всі призначені для користувача дані:

* id (унікальний ідентифікатор користувача);
* name (ім'я користувача);
* last\_name (ім'я користувача);
* role\_id (роль користувача, наприклад - адміністратор сайту);
* status (статус, який показує, чи купив користувач відеоблок);
* telephone (номер телефону);
* email (адреса електронної пошти);
* social\_id (ідентифікатор користувача Facebook);
* password (пароль користувача);
* avatar\_url (url аватара користувача);
* remember\_token (токен, який видається користувачеві при вході в систему);
* created\_at (дата реєстрації користувача);
* update\_at (дата, коли користувач вніс зміни до свого облікового запису);

Більшість таблиць пов’язані до поля. Це поле є унікальним ідентифікатором для кожного запису даних.

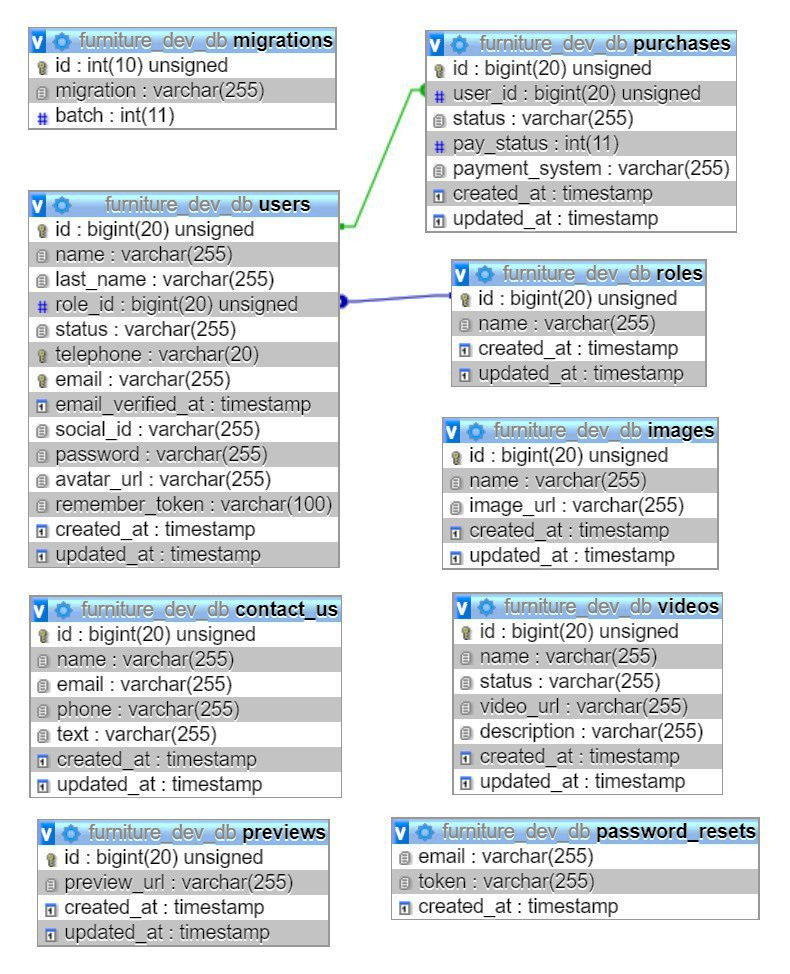


Рисунок. 2.4 — Модель відносин сутностей

Для цієї системи краще всього підходить організація бази даних БД. Ця організація дозволяє вам вносити деякі зміни в профілі користувачів системи.

Основна кодування бази даних — utf8\_general\_ci. Вибір такого кодування був обумовлений необхідністю використання кириличних символів для відображення. Текстова інформація, що дозволяє обране кодування [14].

2.1.13 Сервер баз даних MySQL. MySQL — одна з найпопулярніших і широко використовуваних СУБД (систем управління базами даних) в Інтернеті, яка не призначена для обробки великих обсягів інформації, але її використання ідеально підходить для Інтернет-сайтів, як невеликих, так і досить великих.

MySQL володіє хорошою швидкістю, надійністю і гнучкістю. Робота з ним зазвичай не викликає особливих труднощів. Підтримка сервера MySQL автоматично включена в поставку PHP.

Одним з важливих чинників є те, що це безкоштовно. MySQL заснований на мові структурованих запитів (SQL), який працює практично на всіх платформах, включаючи Linux, UNIX і Windows.

MySQL — це компактний багатопотоковий сервер баз даних. MySQL відрізняється високою швидкістю, стабільністю і простотою використання.

Короткий список функцій MySQL:

* MySQL підтримує мову запитів SQL у стандарті ANSI 92, а також має безліч розширень цього стандарту.
* У таблицях кількість рядків досягає 50 мільйонів.
* Ефективна і проста система безпеки.

Основні конкурентні переваги MySQL:

* безкоштовно для некомерційного використання (для комерційного використання необхідно придбати ліцензію);
* Продуктивність (ось чому Google і Yahoo використовують MySQL);
* Масштабованість (Omniture використовує 7000 серверів MySQL в реальному часі);
* Надійність (код пропрієтарних продуктів містить більш ніж в десять разів більше вразливостей);
* Простота використання, простота впровадження (можна завантажити і запустити систему за 15 хвилин).

##### 2.3 Висновки до другого розділу

У другому розділі дисертації проаналізовано методи створення веб-сайту, а саме клієнтська частина веб-сайту заснована на бібліотеці jQuery , яка в свою чергу, включає в себе розмітку HTML, стилі SCSS і мову програмування TypeScript. Серверна частина, в свою чергу, заснована на написаному на PHP фреймворку Laravel і базі даних MySQL, їх недоліки та переваги.

На основі даних з офіційних сайтів був виправданий порівняльний аналіз найбільш популярних систем управління сайтами: Laravel, Yii і Symfony і виправданий вибір Laravel. Були розглянуті переваги і аргументи на користь вибору веб-сервера Open Server, сервера бази даних MySQL і розроблена база даних.

# РОЗДІЛ 3

# ПРАКТИЧНА ЧАСТИНА РОЗРОБКИ ІНФОРМАЦІЙНО-НАВЧАЛЬНОГО САЙТУ

##### 3.1 Розробка інтерфейсу веб-сайту

Дизайн інтерфейсу користувача (UI) візуалізує робочий прототип, відмальовує кнопки, іконки, форми та інші його компоненти і збирає їх в гармонійний працюючий макет. UI зосереджується на передбаченні того, що можуть зробити користувач та об'єднує концепції дизайну взаємодії, візуального дизайну та інформаційної архітектури.

Головні правила розробки інтерфейсу сайту:

* Зберігати інтерфейс простим. Кращі інтерфейси майже непомітні для користувача.
* Бути цілеспрямованим у макеті сторінки. Розміщення предметів може допомогти привернути увагу до найбільш важливих відомостей.
* Стратегічно використовувати колір і текстуру.
* Використовувати типографіку для створення ієрархії та ясності. Різні розміри, шрифти та розташування тексту сприяють збільшенню масштабованості, розбірливості та читабельності.
* Переконатись, що система повідомляє, що відбувається. Завжди повідомляйте своїх користувачів про місцезнаходження, дії, зміни стану чи помилки.
* Ретельно продумати та передбачити цілі людей, які заходять на ваш сайт, ви можете зменшити навантаження на користувача. Це стає особливо важливим, коли мова йде про дизайн сайту.

Графічний інтерфейс користувача є найбільш важливою частиною веб-сайту. Від інтерфейсу залежить як швидко користувач освоїть систему і чи якісно виконуватиметься робота. Графічний інтерфейс має довести до мінімуму помилки користувача. Тому, до його розробки необхідно підходити серйозно та виділяти багато часу на його проектування. В інтерфейсі є загальні моменти, яких треба дотримуватися. Іншими словами інтерфейс повинен бути інтуїтивно зрозумілим, очевидним та ненав’язливим [15].

##### 3.2 Інструменти розробки веб-сайту

Кожен програміст використовує певні інструменти під час розробки веб-сайту. В процесі розробки було використано ряд інструментів, нижче розглянемо основні із них.

Для написання клієнтської частини сайту обрано Visual Studio Code. Він досить простий в освоєнні, наділений зручним інтерфейсом і всіма необхідними функціями для створення додатків, а те чого не вистачає можна заповнити шляхом установки додаткових розширень.

Основні можливості і переваги програми:

* Visual Studio Code підтримує роботу з TypeSсript, JavaSсript;
* Є вбудовані відладчик і командний рядок;
* Підтримка практично всіх мов програмування;
* Автозаповнення при введенні коду та підсвічування синтаксису;
* Розширення функціоналу за допомогою плагінів;
* Інтеграція з GitHub і GIT.

Для написання серверної частини сайту був обраний PhpStorm — крос-платформенне інтегроване середовище розробки для PHP, яке має колосальний ряд переваг, а саме:

* У кожному файлі запам'ятовується курсор і місце редагування;
* На полях файлів можна робити спеціальні позначки, тому легко знайти ключові місця в файлах;
* Можна легко створювати Scratch файли, які не потрібно зберігати, вони залишаються в проекті, поки їх не видалити;
* Пошук працює значно швидше і набагато функціональніше в порівнянні з іншими середовищами;
* Підказки в функціях, стилі та підсвічування невикористаних змінних.

PhpStorm дозволяє зберігати файли без натискання кнопок збереження. Тобто файли зберігаються при будь-якій зміні, що економить час. У PhpStorm вбудована консоль, логи і багато іншого функціоналу.

Наступним використаним інструментом є Postman - зручний HTTP-клієнт для тестування веб-сайтів. За допомогою розширення можна складати і редагувати прості або складні HTTP-запити. Складені запити автоматично зберігаються на майбутнє для повторного застосування. У програмі є вбудований редактор запитів, з можливостями кодування запитів, завантаження з файлу і відправки бінарних даних.

Для тестування розмітки та стилів зручним інструментом є Adobe Photoshop, яка допомагає у виявленні помилок, таких як невідповідність шрифтів, відмінність в розмірах блоків, різні кольори компонентів сайту, відсутність певних деталей та інші. Ця програма допоможе звірити отриману версію сайту з його початковим макетом [16].

##### 3.3 Структура інформаційно-навчального сайту та опис розділів

Структура сайту — це логічна побудова всіх сторінок ресурсу.  Від її правильності, логічної побудови залежить зручність користувача. Якщо структура розроблена неправильно, навігація незручна для пошуку споживачем, то він надовго не затримається на сайті, закривши вкладку з ним в браузері. Структура залежить від виду сайту, семантичного ядра і цільової аудиторії, тому завжди індивідуальна. При розробці архітектури ресурсу, необхідно аналізувати розміщення кожного розділу і підрозділу, щоб все зробити грамотно, задовольнивши потреби користувача і відповівши на вимоги пошукових роботів. Структура даного сайту показана на Рис. 3.1.

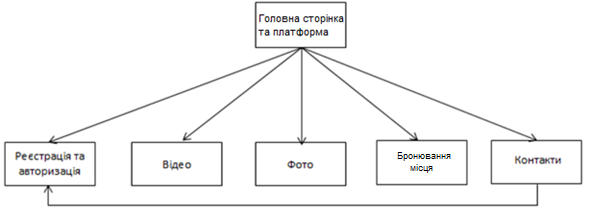


Рисунок. 3.1 — Структурна схема сайту

3.3.1 Дизайн інформаційно-навчального сайту. Дизайн сайту — це оформлення контенту, сукупність всіх графічних елементів на веб-сторінці. Це не виключно візуальне оформлення, а й зручність користувача.

Основне завдання дизайну — познайомити користувача зі сторінкою. Він полегшує взаємодію користувача з веб-сторінкою, а отже позитивно впливає на конверсію і поведінкові чинники. Продуманий дизайн створює потрібне враження про компанію.

Є основні правила створення дизайну:

* Наявність стратегії. Яким би красивим не був сайт, якщо користувачеві незрозуміло його призначення, він подивиться і піде нічого і не купивши.
* Зручність використання. Простота використання охоплює безліч параметрів, від швидкості завантаження до зручності навігації та наявності пошуку.
* Наявність стилю. Хороший сайт відповідає вже існуючим маркетинговим матеріалами компанії, корпоративним кольорам і цінностям. При цьому на ньому дотримуються закони композиції і розумно використовуватися порожній простір.
* Якість контенту. Воно складається з двох показників: читабельності та корисності. Якщо корисний контент буде нечитабельним, то користувач ніколи не дізнається, наскільки він був цінним. У той же час нецікаве наповнення, яке не несе користі, не врятує навіть найкращий дизайн.
* Чи можна довіряти цьому сайту? Корпоративний сайт, де в якості засобу зв'язку вказана лише адреса електронної пошти; блоги, про авторів яких нічого не відомо здаються ненадійними [17].

Відповідно до розробленої структури було спроектовано головну сторінку сайту, яка представлена на рис. 3.2. Вона містить в собі всі основні структурні елементи, перехід зроблено за допомогою гіперпосилань.

Як було сплановано на етапі постановки завдання, сайт містить всі потрібні навігаційні і структурні елементи.

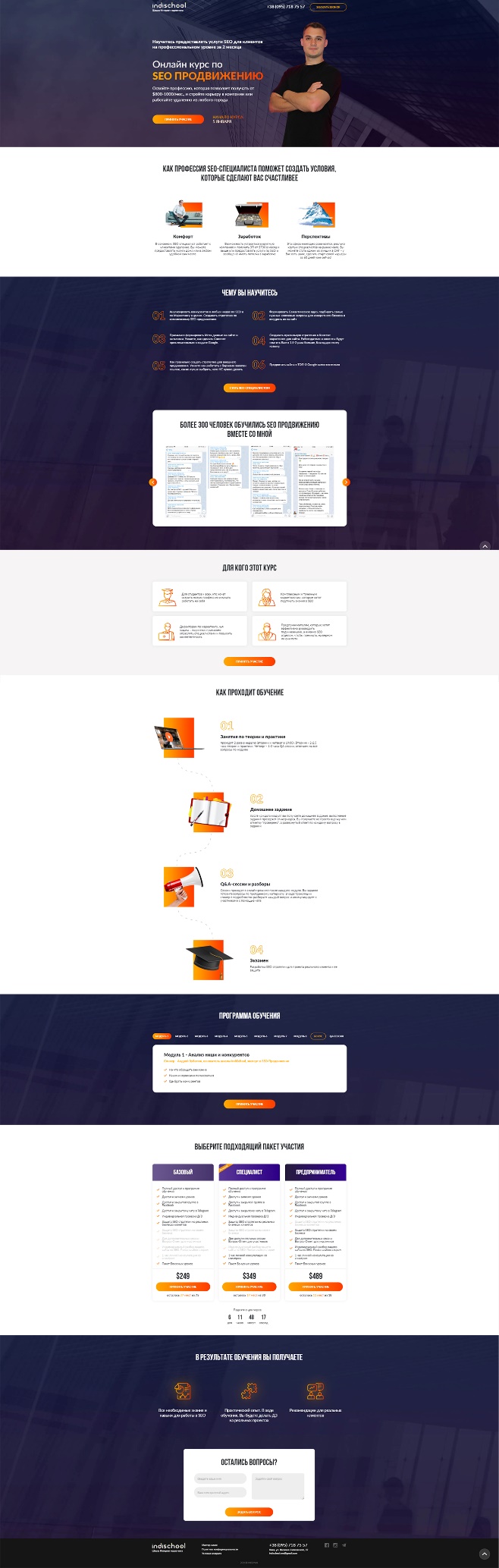


Рисунок. 3.2 — Дизайн сайту

3.3.2 Головна сторінка. Головна сторінка, представлена на рис. 3.3. Коли користувач заходить на наш сайт, то потрапляє прямо на головну сторінку. Головна сторінка складається з верхньої частини, меню навігації, змістовної частини і нижньої частині. На верхній частині розташований логотип. Меню навігація складається з кнопок (заказати дзвінок, взяти участь) для зручного та швидкого переходу на цікавий нам розділ сайта.

Змістовна частина несе в собі інформацію про власника сайту і які послуги він надає, про його досвід, вміння та навички. Нижня частина на всіх рівнях сайту однакова і містить контакти та посилання на бронювання місця.

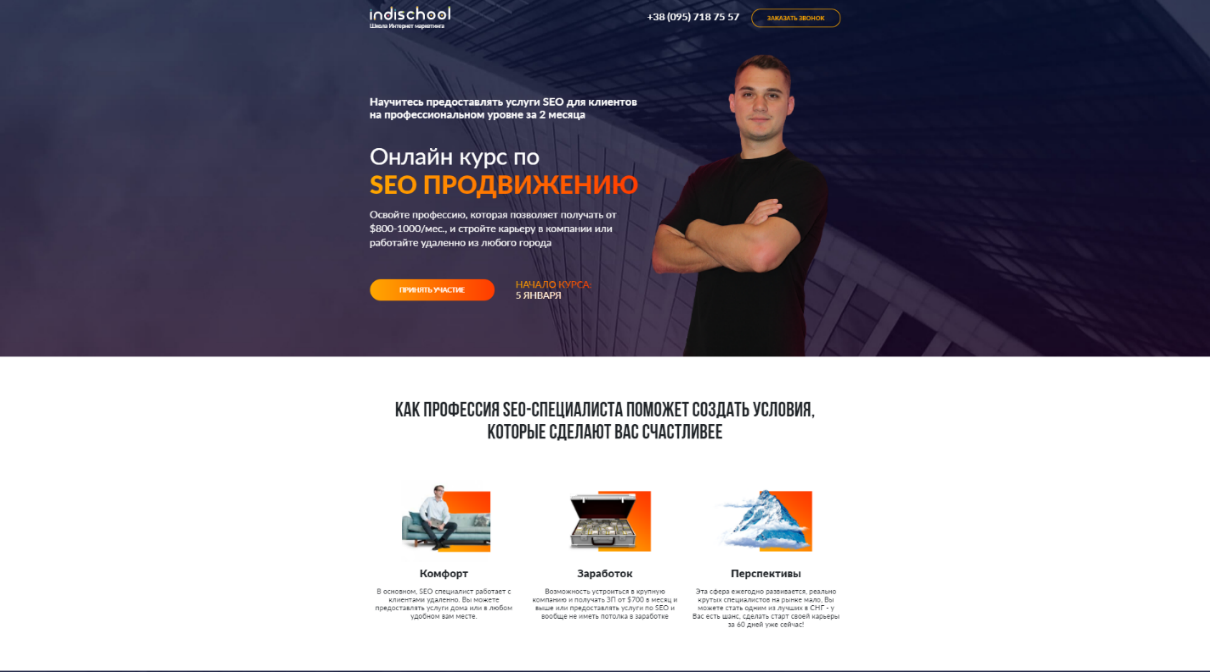


Рисунок. 3.3 — Головна сторінка

3.3.3 Реєстрація та авторизація. Як тільки клієнт оплатив навчання менеджер додає його на навчальну платформу GetCourse. Натиснувши в меню користувачів на кнопку «Додати користувача», менеджер побачить форму реєстрації, представлену на рис. 3.4. Або ж якщо клієнт вже зареєстрований, то необхідно поля вводу даних електронної пошти та паролю для авторизації на платформі, щоб отримати можливість здійснювати навчання. Форма авторизації представлена на рис. 3.5.

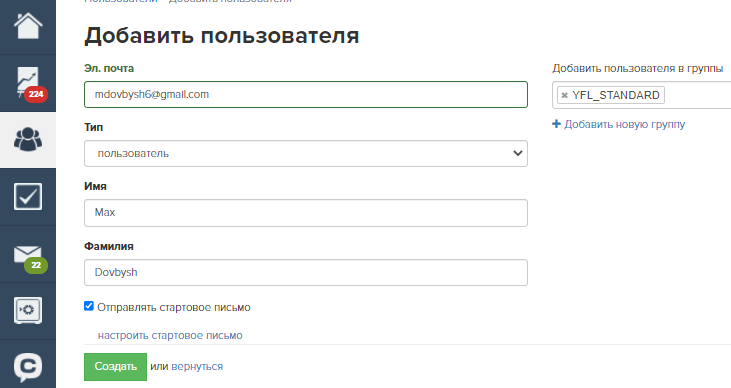


Рис. 3.4 — Реєстрація

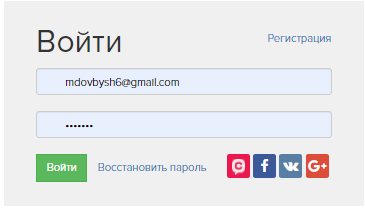


Рисунок. 3.5 — Авторизація

Виконавши вхід, клієнт має змогу переглянути в особистому кабінеті навчальний матеріал курсу, який представлений на рис. 3.6.

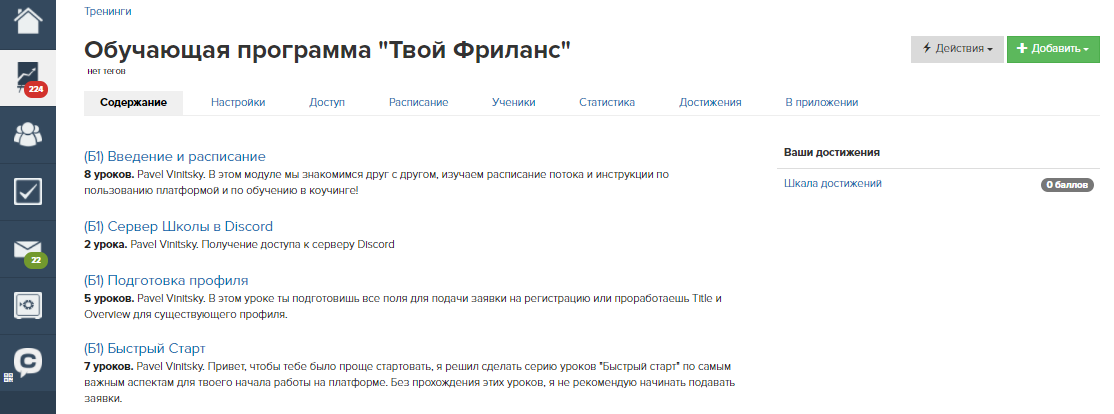


Рисунок. 3.6 — Навчальні відео в особистому кабінеті

Зайшовши у власний профіль та натиснувши на кнопку «Змінити пароль», авторизований користувач побачить форму зміни паролю, представлену на рис. 3.7. Необхідно ввести старий та новий пароль.

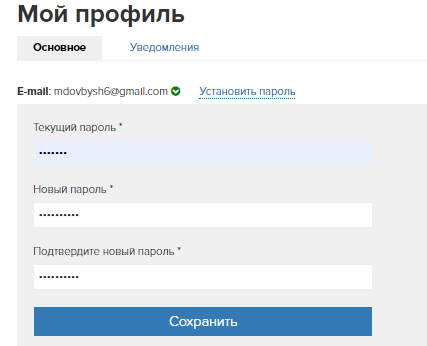


Рисунок. 3.7 — Зміна паролю

Якщо при авторизації користувач забуде пароль, його можна відновити натиснувши кнопку «Скинути пароль» та ввівши свою адресу електронної пошти, представлено на рис. 3.8. На пошту прийде посилання для відновлення пароля, представлено на рис. 3.9.

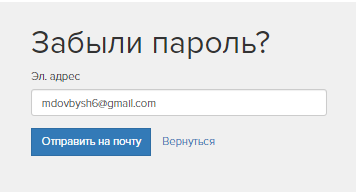


Рисунок. 3.8 — Скидання пароля

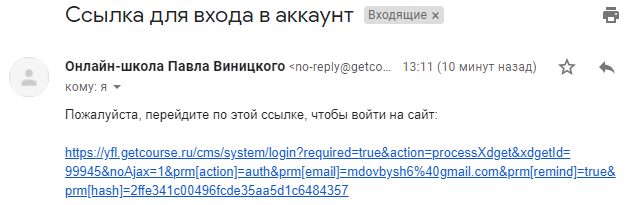


Рисунок. 3.9 — Посилання, щоб відновити пароль прийшло на email

3.3.4 Сторінка «Відео». Відображення відео відбувається по різному для різних клієнтів. Коли клієнт не придбав навчання, то нього не має доступу до жодного блоку відео. В разі коли клієнт придбав певний інформаційний блок, то він зможе побачити саме його в особистому кабінеті, представлено на рис. 3.10.

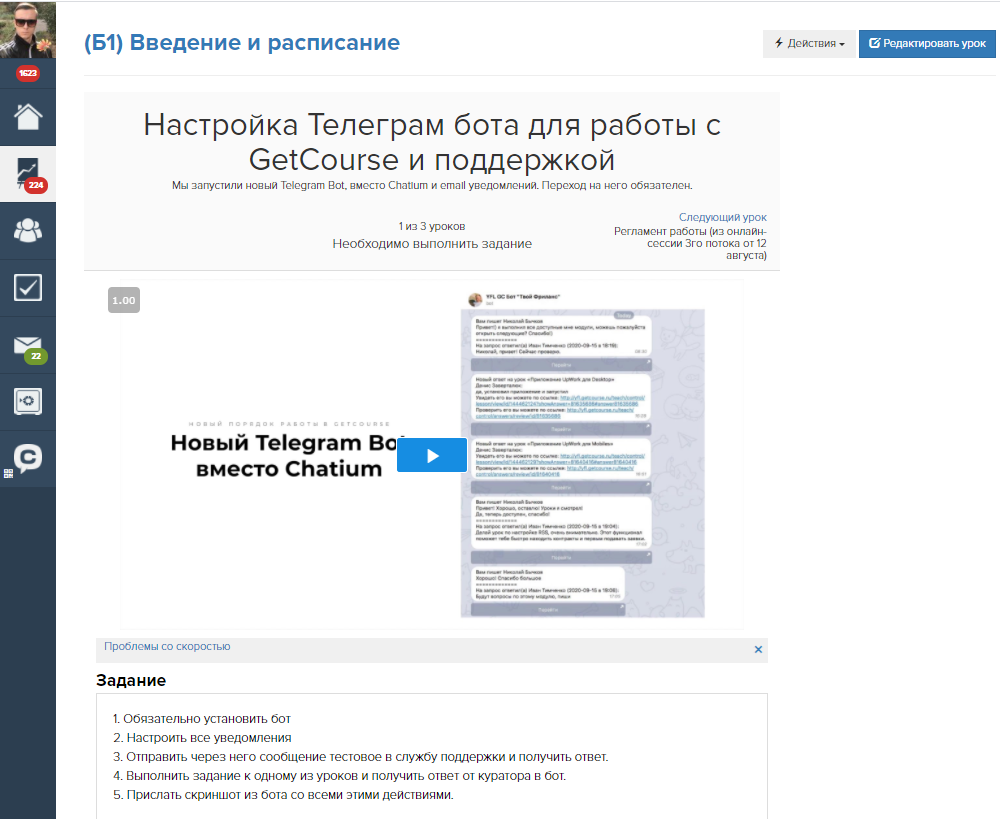


Рисунок. 3.10 — Вкладка «Відео» в особистому кабінеті

3.3.5 Форма «Забронювати місце». Для зручності клієнтів створено бронювання місця. Користувач може заповнити форму, щоб забронювати за собою місце в навчанні. Для цього потрібно лише ввести ваше ім'я, адресу електронної пошти та номер телефону. Форму бронювання місця представлено на рис. 3.11.

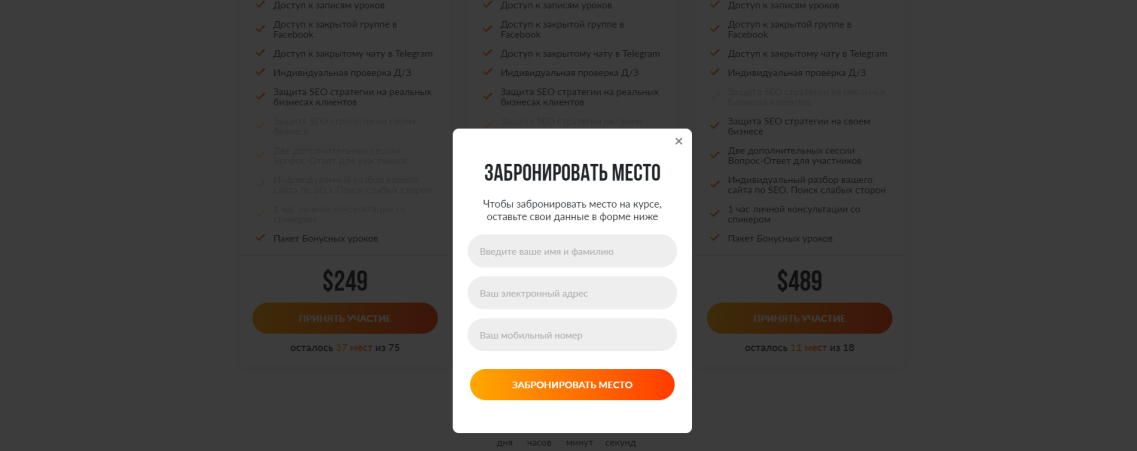


Рисунок. 3.11 — Сторінка «Забронювати місце»

Після відправки форми власнику сайту прийде подібний лист, де вказані всі необхідні параметри, представлено на рис. 3.12.

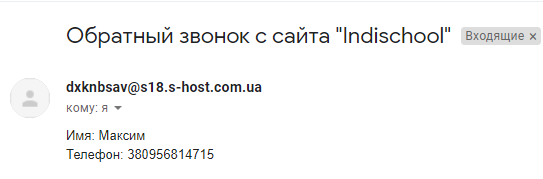


Рисунок. 3.12 — Лист, що прийшов на email власнику сайту

##### 3.4 Панель адміністратора

Для доступу до адміністративної панелі управління навчальної платформи GetCourse використовується адреса веб-сайту. Після правильного введення адреси, відкриється авторизація панелі адміністратора, яка представлена на рис. 3.13. Для входу в адміністративну частину сайту потрібно ввести логін і пароль головного адміністратора сайту.

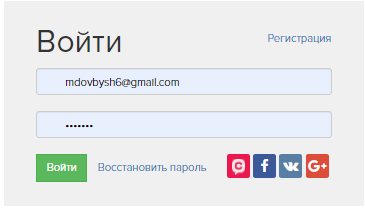


Рисунок. 3.13 — Вхід у панель управління

При правильному введенні імені користувача і пароля відкриється головна сторінка адміністрування, яка представлена на рис. 3.14. Тобто буде відкрита панель управління, яка надасть управління контентом веб-сайту та покаже основну статистику.

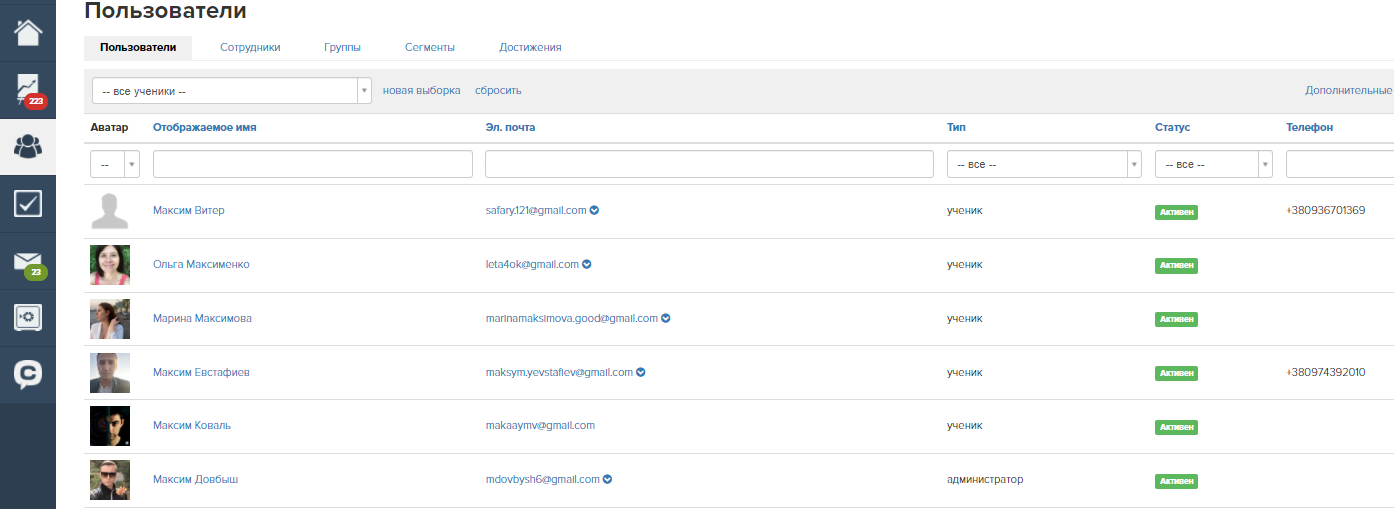


Рисунок. 3.14 — Панель адміністратора

3.4.1 Інтерфейс. Інтерфейс панелі адміністратора інтуїтивно зрозумілий і простий у використанні. Основним елементом є функціональний блок, який представлено на рис 3.15. Даний блок містить основні інструменти управління контентом сайту. В ньому розміщена стисла інформацію про те, що відбувається з веб-сайтом. А саме відображається така інформація, як кількість користувачів (просто зареєстрованих на сайті та тих, що придбали товар), для навчання курсу.

В функціональному блоці розташовані:

* Кнопка "Користувачі" — дозволяє переглянути користувачів, яких зареєстрували на платформі та тих, що придбали певний інформаційний блок.
* Кнопка "Тренінги" — дозволяє переглянути, додати, змінити чи видалити необхідне відео з відповідного інформаційного блока.
* Кнопка "Фото" — дозволяє переглянути, додати, змінити чи видалити фото користувача.
* Кнопка "Змінити пароль" — користувач може змінити поточний пароль на новий.
* Кнопка "Вихід" — вихід з панелі управління.



Рисунок. 3.15 — Функціональний блок адміністративної панелі

3.4.2 Додавання контенту. Оновлення контенту є однією з найважливіших і невід’ємних частин роботи зі створення сайту. У вкладці "Тренінги" можна додати, змінити чи видалити відео, що представлено на рис. 3.16. У цій вкладці обирається відео, його назва та його опис. У вкладці "Фото" можна додати, змінити чи видалити фото уроків власника сайту, які представлено на рис. 3.17. У цій вкладці обирається фото та його назва. Даний функціонал надає можливість власнику сайту власноруч змінювати контент та зменшує необхідність супроводу програмістами даного сайту.

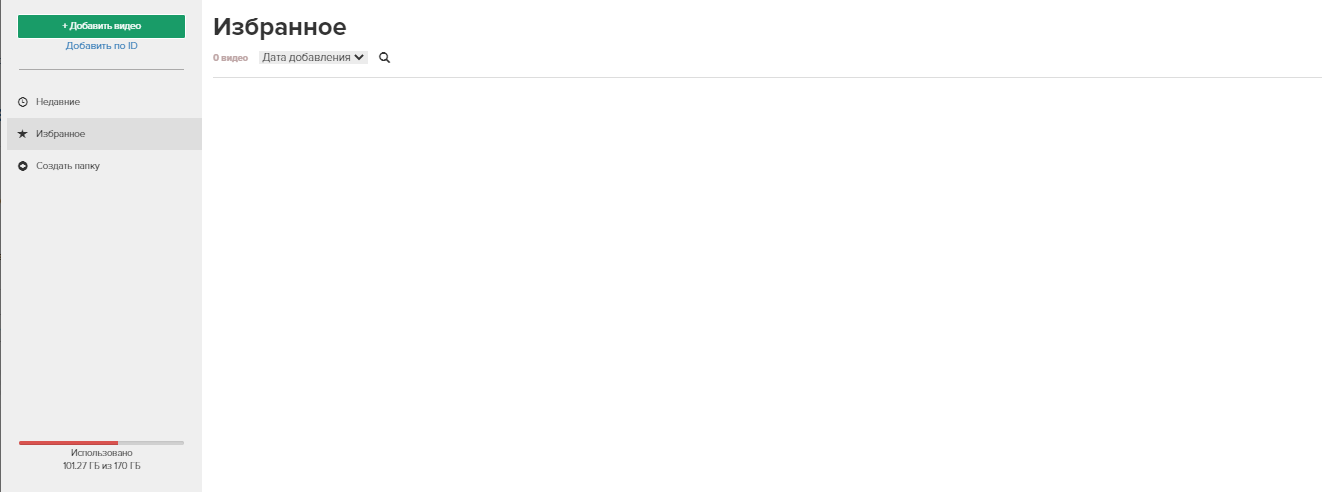


Рисунок. 3.16 — Додавання відео

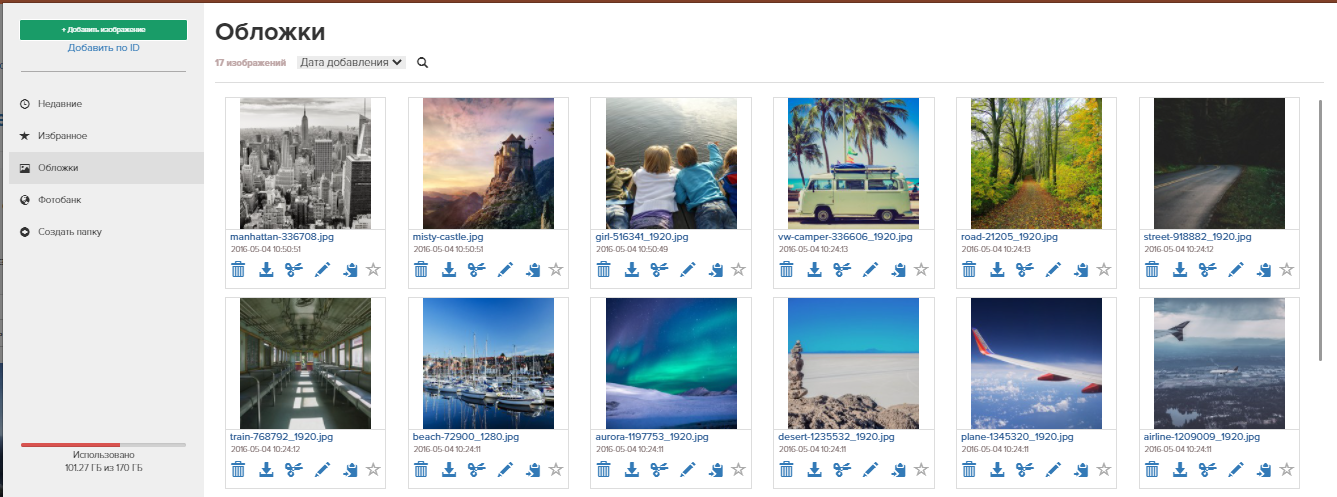


Рисунок. 3.17 — Додавання фото

3.5 Тестування веб-сайту

Для перевірки роботи сайту та відповідності його реалізації до вимог було використано такі три види тестування:

* тестування користувацького інтерфейсу (GUI);
* функціональне тестування;
* регресійне тестування.

Тестування користувацького інтерфейсу (GUI) — це тестування програмного забезпечення, яке перевіряє графічний інтерфейс користувача, який включає всі елементи, такі як меню, прапорець, кнопки, кольори, шрифти, розміри, піктограми, вміст та зображення. Перш ніж вивчити особливості та потужності системи, користувачі в першу чергу звертають увагу на інтерфейс та дизайн і таким чином формують своє ставлення до всього продукту. Тому тестування GUI потрібне для перевірки функціональності та зручності використання елементів дизайну для користувача програми, що була протестована [18].

Функціональне тестування — це тип тестування програмного забезпечення, що підтверджує програмну систему щодо функціональних вимог. Його метою є тестування кожної функції програмного додатку.

Дане тестування перевіряє інтерфейс користувача, API, базу даних, безпеку, зв'язок клієнта/сервера та інші функціональні можливості програми, що перебуває під тестом. Тестування може проводитися вручну або за допомогою автоматизації.

Регресійне тестування визначається як тип тестування програмного забезпечення для підтвердження того, що нещодавня зміна програми чи коду не вплинула негативно на існуючі функції. Це тестування проводиться для того, щоб нові зміни коду не мали побічних ефектів на існуючі функції. Це гарантує, що старий код все ще працює, як тільки будуть зроблені останні зміни коду.

Таке тестування проводилося по закінченню розробки програмного забезпечення. Усі виявлені помилки було виправлено [19].

### 

3.6 Встановлення та введення в експлуатацію

Оскільки нами реалізовано повністю готовий до введення в експлуатацію програмний продукт, то необхідно його перенести з локального серверу на хостинг. Розглянемо покрокову інструкцію:

1. Заходимо в панель адміністратора хостингу, на який будемо переносити сайт. Домен для сайту на цей момент вже повинен бути зареєстрований, а сайт доданий в панель хостингу (Мої сайти -> Додати сайт).
2. Після створення ми побачимо хост, порт, логін та пароль до серверу, а також логін та пароль бази даних. Ці дані знадобляться далі.
3. Тепер потрібно ввести ці дані та підключити сервер. В PhpStorm знаходимо вкладку Tools->Deployment->Configuration та заповнюємо даними, представлено на рис. 3.18.

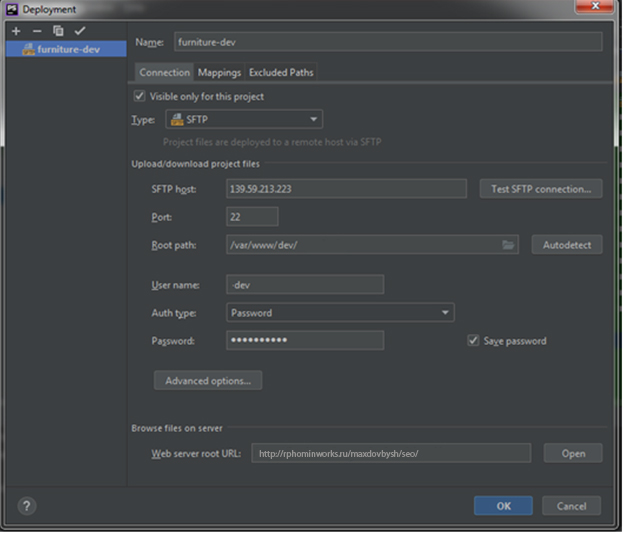


Рисунок. 3.18 — Заповнення серверними даними

1. Перевіряємо з’єднання натиснувши Test SFTP connection, представлено на рис. 3.19.

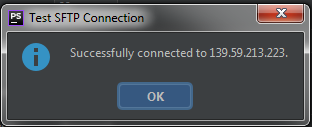


Рисунок. 3.19 — Проведення Test SFTP connection

1. Тепер потрібно перенести всі файли з локальної в серверну частину.
2. Далі необхідно вже на сервері змінити .env файл. Даний файл знаходиться в кореневій папці проекту.
3. Відкриваємо файл .env і зміняємо дані про базу даних на ті, які ми отримали в попередньому кроці. Нам потрібно вказати нове ім’я бази даних, ім’я користувача та пароль. Дані зміни представлено на рис. 3.20.

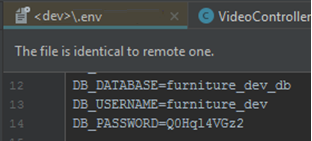


Рисунок. 3.20 — Редагування .env файлу на сервері

1. Тепер потрібно відкрити консоль сервера. Це можна зробити перейшовши в PhpStorm на вкладку Tools->Start SSH session->dev.
2. Потім потрібно зайти в папку з проектом. Це можна зробити ввівши команду «cd www».
3. Далі підключаємо міграції в базу даних командою «php artisan migrate», після чого з’являться в базі даних таблички та зв’язки між ними.
4. Для того щоб вручну не заповнювати інформацію в базі даних, то використовуємо команду «php artisan db:seed», після якої база заповниться необхідними даними.

Отже, отримаємо інформаційно-навчальний веб-сайт для Школи Інтернет-маркетингу на прикладі онлайн курсу по SEO просуванню, який повністю готовий до використання.

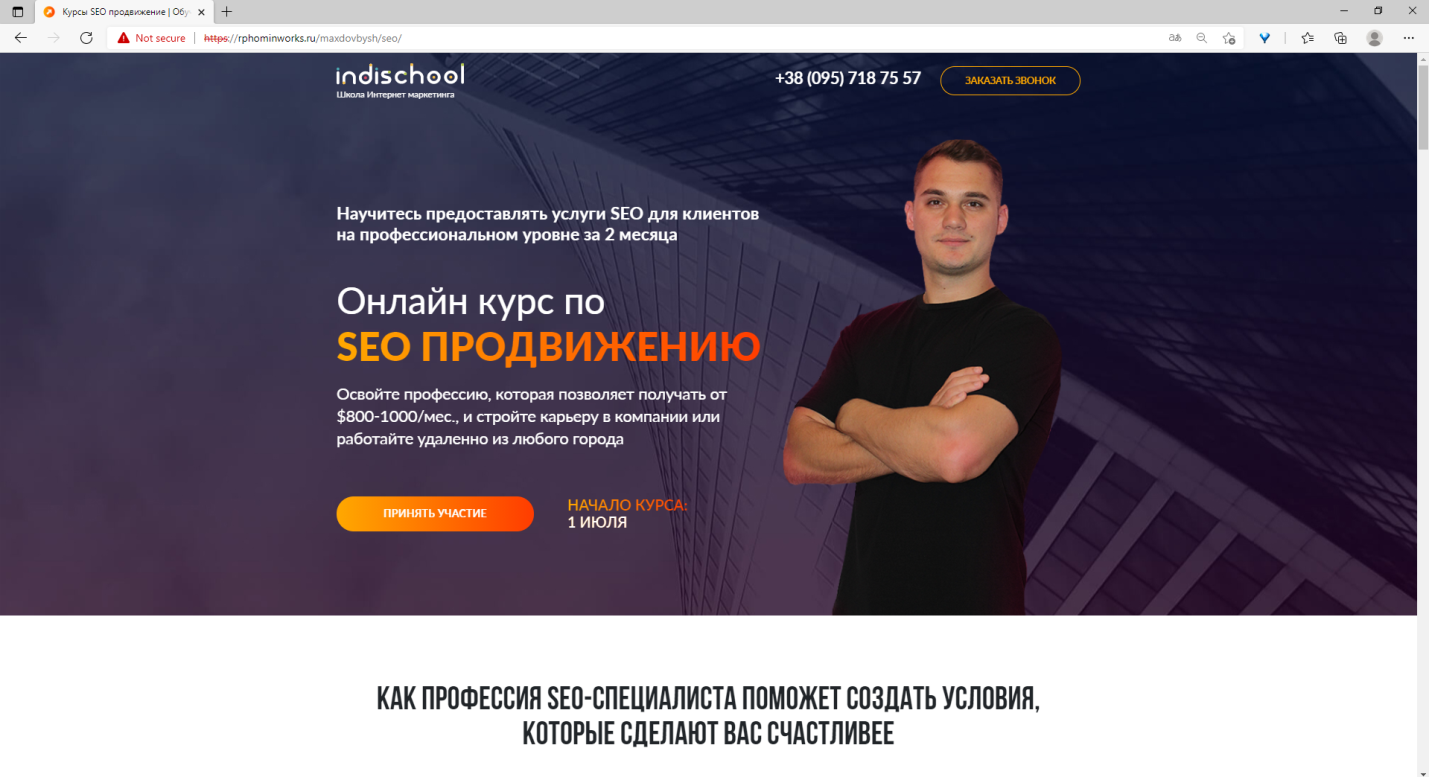


Рисунок. 3.21 — Робота ресурсу в Edge

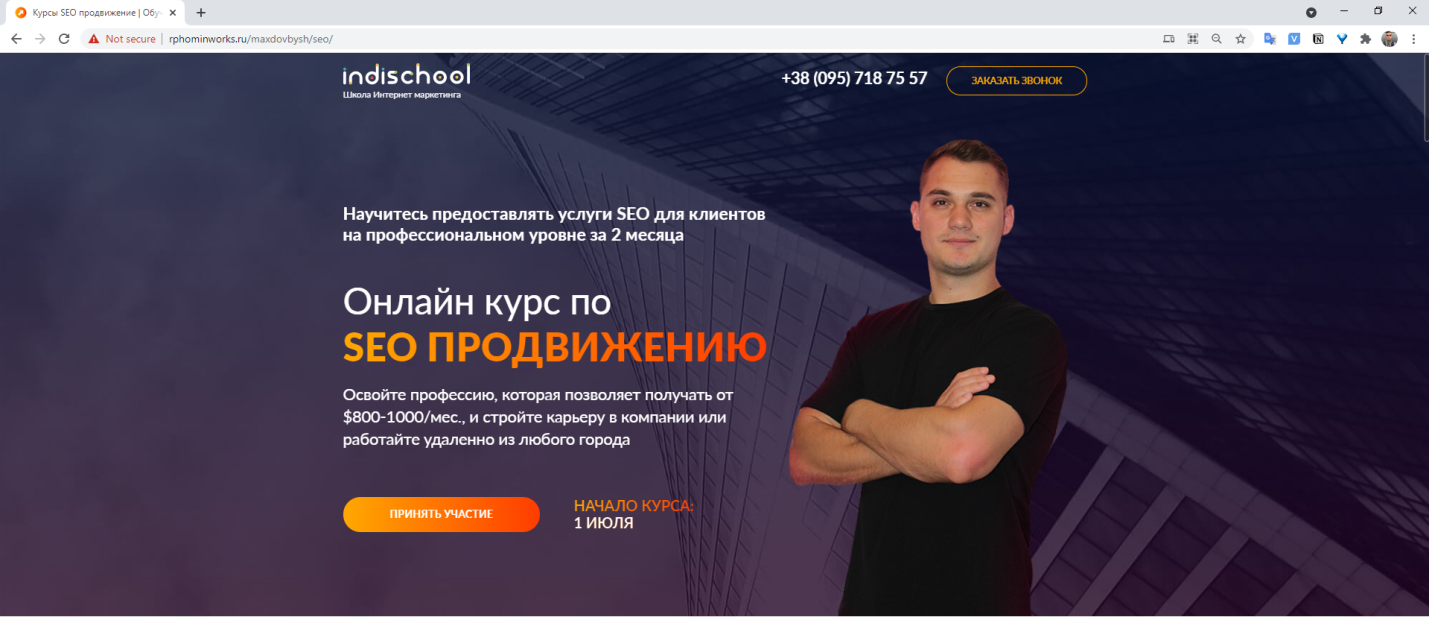


Рисунок. 3.22 — Робота ресурсу в Google Chrome

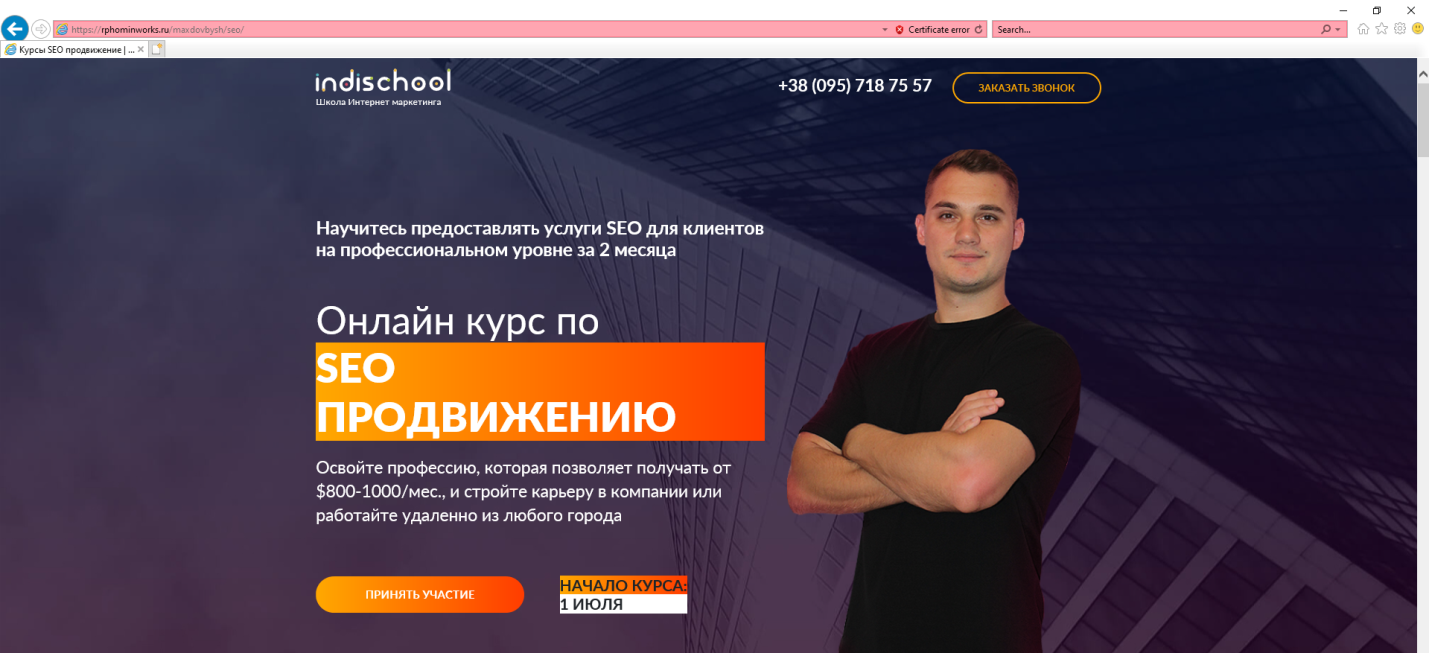


Рисунок. 3.23 — Робота ресурсу в Internet Explorer

3.7 Висновки до третього розділу

У третьому розділі дипломної роботи була проведена розробка інтерфейсу та дизайну сайту, розглянуто додавання контенту на веб-сайт за допомогою панелі адміністратора. Представлено основні інструменти для розробки веб-сайту, які використовувалися в ході виконання роботи, а саме PhpStorm, Postman, Visual Studio Code та Photoshop.

Результатом практичної частини дипломної роботи є розроблений інформаційно-навчальний веб-сайт для Школи Інтернет-маркетингу на прикладі онлайн курсу по SEO просуванню, починаючи з установки проекту на сервер та закінчуючи покроковою інструкцією для розробників та користувачів щодо управлінням сайтом. Розглянуто основні аспекти оформлення зовнішнього вигляду сайту, на основі яких він був створений.

# ВИСНОВКИ

У ході виконання кваліфікаційної роботи бакалавра було розроблено інформаційно-навчальний веб-сайт для Школи Інтернет-маркетингу на прикладі онлайн курсу по SEO просуванню. Даний сайт орієнтований для широкого спектру застосування в online торгівлі. З його допомогою користувачі зможуть придбати інформаційний блок та навчатися з його допомогою не виходячи з дому, а також отримувати необхідну інформацію він власника сайту по електронній пошті. При його розміщенні в мережі Internet географія розповсюдження збільшується до масштабів всього світу.

Розроблений сайт задовольняє всі вимоги, поставлені на етапі постановки завдання. При розробці веб-сайту було проведено аналіз існуючих фреймворків: Laravel, Yii, Symfony. Розглянуто основні функції: їх переваги та недоліки. В ході розробки веб-сайту інформаційно-навчального сайту обрано саме Laravel, тому що вони потужні, гнучкі та відносно не складні в вивченні. Також було створено панель управління, яка призначена для організації та управління контентом сайту. ЇЇ використання допомагає організовувати роботу веб-сайту і скоротити витрати часу та ресурсів.

# СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Офіційний сайт MyOwnConference [електронний ресурс]. – Режим доступу: <https://myownconference.ru/blog/index.php/online-obuchenie/>. – Дата доступу: 07.01.2020.
2. Офіційний сайт Webkonstruktor [електронний ресурс]. – Режим доступу: <https://www.webkonstruktor.com/українською/типи-сайтів/>. – Дата доступу: 12.01.2019.
3. Офіційний сайт Scu.edu [електронний ресурс]. – Режим доступу: <https://www.scu.edu/mobi/business-courses/business-expansion/session-3-customer-feedback/>. – Дата доступу: 23.10.2019.
4. Офіційний сайт Lemarbet [електронний ресурс]. – Режим доступу: <https://lemarbet.com/ua/razvitie-internet-magazina/e-mail-rassylka-kak-instrument-marketinga-dlya-internet-magazina/> . – Дата доступу: 23.10.2019.
5. Офіційний сайт Semantica [електронний ресурс]. – Режим доступу: <https://semantica.in/blog/chto-takoe-adminka-sajta.html>. – Дата доступу: 14.01.2020.
6. Офіційний сайт My Master [електронний ресурс].– Режим доступу: <https://my-master.net.ua/ua/etapi-stvorennya-sajtu-z-nulya/>. – Дата доступу: 02.03.2019.
7. Офіційний сайт Znannya [електронний ресурс].– Режим доступу: <http://www.znannya.org/?view=html> . – Дата доступу: 04.03.2020.
8. Офіційний сайт Tproger [електронний ресурс].– Режим доступу: <https://tproger.ru/translations/complete-sass-guide/>. – Дата доступу: 04.03.2020.
9. Офіційний сайт Xsltdev [електронний ресурс].– Режим доступу: <https://xsltdev.ru/typescript/guide/intro/about-ts/>. – Дата доступу: 08.03.2020.
10. Офіційний сайт Legko [електронний ресурс].– Режим доступу: https://legko.org.ua/8-krashih-php-framework-dlia-veb-rozrobnikiv. – Дата доступу: 04.03.2020.
11. Офіційний сайт Hostinger [електронний ресурс].– Режим доступу: <https://www.hostinger.com.ua/rukovodstva/8-luchshih-php-framework-dla-web-razrabotchikov/#1-Laravel> . – Дата доступу: 15.04.2020.
12. Офіційний сайт Eternalhost [електронний ресурс].– Режим доступу: <https://eternalhost.net/blog/sozdanie-saytov/luchshie-php-freymvorki>. – Дата доступу: 15.04.2020.
13. Офіційний сайт Sovety [електронний ресурс].– Режим доступу: <https://sovety.pp.ua/index.php/ua/cms/1555-open-server2>. – Дата доступу: 27.04.2020.
14. Офіційний сайт Wiki Fizmat [електронний ресурс].– Режим доступу: <http://wiki.fizmat.tnpu.edu.ua/index.php/%D0%91%D0%94>. – Дата доступу: 04.05.2020.
15. Офіційний сайт Stfalcon [електронний ресурс].– Режим доступу: <https://stfalcon.com/ru/blog/post/user-interface-web-design-principles>. – Дата доступу: 04.05.2020.
16. Офіційний сайт Jetbrains [електронний ресурс].– Режим доступу: <https://www.jetbrains.com/ru-ru/phpstorm/features/>. – Дата доступу: 07.05.2020.
17. Офіційний сайт Ag.Marketing [електронний ресурс].– Режим доступу: <https://ag.marketing/dizayn-i-seo/>. – Дата доступу: 07.05.2020.
18. Леонтьев Б.В. Веб-Дизайн: Тонкости, хитрости и секреты / Леонтьев Б.В. - М. Майор, 2001.–170 с.
19. Офіційний сайт QA Evolution [електронний ресурс].– Режим доступу: <https://qaevolution.ru/testirovanie-po/vidy-testirovaniya-po/>. – Дата доступу: 14.05.2020.

# ДОДАТОК А

# ВИХІДНІ КОДИ РЕСУРСУ — FRONTEND

**index.html**

<!DOCTYPE html>

<html lang="ru">

<head>

<meta charset="UTF-8">

<meta http-equiv="X-UA-Compatible" content="IE=edge">

<meta name="viewport" content="width=device-width, initial-scale=1">

<meta name="description" content="Онлайн курс по SEO продвижению! Старт курса 1 июля. Успей вступить по скидке, уже скоро поднятие цен! Школа Интернет маркетинга IndiSchool">

<title>Курсы SEO продвижение | Обучение SEO оптимизации | IndiSchool</title>

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

<link rel="stylesheet" href="css/bootstrap.min.css">

<link rel="stylesheet" href="css/libs.min.css">

<link rel="stylesheet" href="css/main.css">

<!-- HTML5 shim and Respond.js for IE8 support of HTML5 elements and media queries -->

<!-- WARNING: Respond.js doesn't work if you view the page via file:// -->

<!--[if lt IE 9]>

<script src="https://oss.maxcdn.com/html5shiv/3.7.3/html5shiv.min.js"></script>

<script src="https://oss.maxcdn.com/respond/1.4.2/respond.min.js"></script>

<![endif]-->

</head>

<body>

<section class="main" id="main">

<header class="header">

<div class="container">

<div class="row align-items-center">

<div class="col-lg-3">

<div class="logo">

<img src="img/logo.svg" alt="">

<p class="logo-text">Школа Интернет маркетинга</p>

</div>

</div>

<div class="col-lg-9">

<div class="header-right">

<a href="tel:+38 (095) 718 75 57" class="header-phone">+38 (095) 718 75 57</a>

<a href="#callback-popup" class="header-callback-btn modal-btn">Заказать звонок</a>

</div>

</div>

</div>

</div>

</header>

<div class="container">

<div class="row">

<div class="col-lg-7">

<h2 class="main-suptitle fadeInDown wow" data-wow-delay=".1s">Научитесь предоставлять услуги SEO для клиентов на профессиональном уровне за 2 месяца</h2>

<h1 class="main-title fadeInDown wow" data-wow-delay=".4s">Онлайн курс по <span>SEO продвижению</span></h1>

<p class="main-desc">Освойте профессию, которая позволяет получать от $800-1000/мес., и стройте карьеру в компании или работайте удаленно из любого города</p>

<div class="main-info">

<a href="#price" class="main-btn btn scroll">Принять участие</a>

<div class="main-date main-date-desctop">Начало курса: <span>1 июля</span></div>

<div class="main-date-mob">Начало курса: <span>1 июня</span></div>

</div>

</div>

<div class="col-lg-5">

<img class="main-img" src="img/main-img.png" alt="">

</div>

</div>

</div>

</section>

<section class="advantages-proffesion">

<div class="container">

<div class="row">

<div class="col">

<h3 class="advantages-proffesion-title s-title">Как профессия SEO-специалиста поможет создать условия, которые сделают вас счастливее</h3>

</div>

</div>

<div class="row">

<div class="col-lg-4">

<div class="advantages-proffesion-item fadeInLeft wow" data-wow-delay=".3s">

<img src="img/advantages-proffesion-img-1.png" alt="">

<h4 class="advantages-proffesion-item-title">Комфорт</h4>

<p class="advantages-proffesion-item-desc">В основном, SEO специалист работает с клиентами удаленно. Вы можете предоставлять услуги дома или в любом удобном вам месте.</p>

</div>

</div>

<div class="col-lg-4">

<div class="advantages-proffesion-item zoomIn wow" data-wow-delay=".3s">

<img src="img/advantages-proffesion-img-2.png" alt="">

<h4 class="advantages-proffesion-item-title">Заработок</h4>

<p class="advantages-proffesion-item-desc">Возможность устроиться в крупную компанию и получать ЗП от $700 в месяц и выше или предоставлять услуги по SEO и вообще не иметь потолка в заработке</p>

</div>

</div>

<div class="col-lg-4">

<div class="advantages-proffesion-item fadeInRight wow" data-wow-delay=".3s">

<img src="img/advantages-proffesion-img-3.png" alt="">

<h4 class="advantages-proffesion-item-title">Перспективы</h4>

<p class="advantages-proffesion-item-desc">Эта сфера ежегодно развивается, реально крутых специалистов на рынке мало, Вы можете стать одним из лучших в СНГ - у Вас есть шанс, сделать старт своей карьеры за 60 дней уже сейчас!</p>

</div>

</div>

</div>

</div>

</section>

<script src="https://cdnjs.cloudflare.com/ajax/libs/jquery/3.2.1/jquery.min.js"></script></section>

<script src="js/libs.min.js"></script>

<script src="js/common.js"></script>

</body>

</html>

**thanks.html**

<!DOCTYPE html>

<html lang="ru">

<head>

<meta charset="UTF-8">

<meta http-equiv="X-UA-Compatible" content="IE=edge">

<meta name="viewport" content="width=device-width, initial-scale=1">

<title>Онлайн курс по SEO продвижению</title>

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

<link rel="stylesheet" href="css/bootstrap.min.css">

<link rel="stylesheet" href="css/libs.min.css">

link rel="stylesheet" href="css/main.css">

<!-- HTML5 shim and Respond.js for IE8 support of HTML5 elements and media queries -->

<!-- WARNING: Respond.js doesn't work if you view the page via file:// -->

<!--[if lt IE 9]>

<script src="https://oss.maxcdn.com/html5shiv/3.7.3/html5shiv.min.js"></script>

<script src="https://oss.maxcdn.com/respond/1.4.2/respond.min.js"></script>

<![endif]-->

</head>

<body>

<section class="main thanks">

<header class="header">

<div class="container">

<div class="row align-items-center">

<div class="col-lg-3">

<div class="logo">

<img src="img/logo.svg" alt="">

<p class="logo-text">Школа Интернет маркетинга</p>

</div>

</div>

<div class="col-lg-9">

<div class="header-right">

<a href="tel:+38 (095) 718 75 57" class="header-phone">+38 (095) 718 75 57</a>

<a href="#callback-popup" class="header-callback-btn modal-btn">Заказать звонок</a>

</div>

</div>

</div>

</div>

</header>

<div class="container">

<div class="row">

<div class="col">

<div class="thanks-wrapper">

<div class="thanks-suptitle">Спасибо, что забронировали участие в курсе</div>

<div class="thanks-title">SEO продвижение</div>

<img src="img/smille.svg" alt="">

<div class="thanks-welcome">Добро пожаловать и до встречи на курсе!</div>

<div class="thanks-text">С Вами свяжутся в течении 15 минут, чтобы<br> уточнить все детали</div>

<a href="index.html" class="thanks-btn btn">Вернуться на главную</a>

</div>

</div>

</div>

</div>

</section>

<div class="popup">

<div class="callback-popup zoom-anim-dialog" id="callback-popup">

<h4 class="callback-popup-title">Заказать звонок</h4>

<form class="callback-popup-form">

<input class="callback-popup-form-input" type="text" name="name" placeholder="Введите ваше имя" required>

<input class="callback-popup-form-input" type="tel" name="phone" placeholder="Ваш мобильный номер" required>

<input class="callback-popup-form-btn btn" type="submit" value="Перезвонить мне">

</form>

<div class="callback-popup-success">

<img src="img/phone-icon.svg" alt="">

<div class="callback-popup-success-title">Спасибо</div>

<p class="callback-popup-success-text">С Вами свяжутся в течении 15 минут, чтобы уточнить все детали</p>

</div>

</div>

</div>

<script src="https://cdnjs.cloudflare.com/ajax/libs/jquery/3.2.1/jquery.min.js"></script>

<script src="js/libs.min.js"></script>

<script src="js/common.js"></script>

</body>

</html>

**common.js**

$(document).ready(function() {

// Всплывающее окно

$('.modal-btn').magnificPopup({

type: 'inline',

fixedContentPos: false,

fixedBgPos: true,

overflowY: 'auto',

closeBtnInside: true,

preloader: false,

midClick: true,

removalDelay: 300,

mainClass: 'my-mfp-zoom-in'

});

// Слайдер отзывов

$('.reviews-slider').slick({

slidesToShow: 4,

slidesToScroll: 1,

prevArrow: '<div class="arrow arrow-prev"><img src="img/arrow-prev.svg" alt="prev"></div>',

nextArrow: '<div class="arrow arrow-next"><img src="img/arrow-next.svg" alt="next"></div>',

responsive: [

{

breakpoint: 1024,

settings: {

slidesToShow: 3

}

},

{

breakpoint: 767,

settings: {

slidesToShow: 2

}

},

{

breakpoint: 480,

settings: {

slidesToShow: 1

}

}

]

});

// Галерея отзывов во всплывающем окне

$('a[data-rel^=lightcase]').lightcase({

swipe: true,

maxWidth: 1500,

maxHeight: 1500

});

// Вкладки модулей

$('.programm-tabs-nav-item').on('click', function() {

$(this).addClass('active').siblings().removeClass('active').closest('.programm-tabs').find('.programm-tabs-content').removeClass('active').eq($(this).index()).addClass('active');

});

// Отправка данных на почту

$('.callback-popup-form').submit(function() {

var th = $(this);

$.ajax({

type: 'POST',

url: 'mail-a.php',

data: th.serialize()

}).done(function() {

$('.callback-popup-success').css('opacity', '1');

$('.callback-popup-success').css('z-index', '3');

setTimeout(function() {

$('.callback-popup-success').css('opacity', '0');

$('.callback-popup-success').css('z-index', '-1');

$.magnificPopup.close();

th.trigger('reset');

}, 3000);

});

return false;

});

$('.questions-form').submit(function() {

var th = $(this);

$.ajax({

type: 'POST',

url: 'mail-b.php',

data: th.serialize()

}).done(function() {

$('.questions-form-success-wrapper').css('opacity', '1');

$('.questions-form-success-wrapper').css('z-index', '99');

setTimeout(function() {

$('.questions-form-success-wrapper').css('opacity', '0');

$('.questions-form-success-wrapper').css('z-index', '-1');

th.trigger('reset');

}, 3000);

});

return false;

});

$('.questions-form-success-close').click(function() {

$('.questions-form-success-wrapper').css('opacity', '0');

$('.questions-form-success-wrapper').css('z-index', '-1');

});

$('.reservesion-popup-form').submit(function() {

var th = $(this);

$.ajax({

type: 'POST',

url: 'mail-c.php',

data: th.serialize()

}).done(function() {

window.location.replace('thanks.html'),

setTimeout(function() {

th.trigger('reset');

}, 3000);

});

return false;

});

// Плавный скролл

$('.scroll').on('click', function (event) {

event.preventDefault();

var id = $(this).attr('href'),

top = $(id).offset().top + 0;

$('body,html').animate({scrollTop: top}, 2000);

});

// Таймер

$('.timer').countdown('2021/07/06', function(event) {

var $this = $(this).html(event.strftime(''

+ '<div class="timer-item-wrap"><span class="timer-item">%-d</span> <span class="timer-desc">дня</span></div> '

+ '<div class="timer-item-wrap"><span class="timer-item">%-H</span> <span class="timer-desc">часов</span></div> '

+ '<div class="timer-item-wrap"><span class="timer-item">%M</span> <span class="timer-desc">минут</span> </div>'

+ '<div class="timer-item-wrap"><span class="timer-item">%S</span> <span class="timer-desc">секунд</span></div>'));

});

$(window).scroll(function() {

if($(window).scrollTop() >= 400) {

$('.totop').css('opacity', '1');;

} else {

$('.totop').css('opacity', '0');;

}

});﻿

// Wow animation

new WOW().init();

});

});

**main.css**

@font-face {

font-family: 'Lato-Regular';

src: url("../fonts/Lato-Regular.eot");

src: url("../fonts/Lato-Regular.eot?#iefix") format("embedded-opentype"), url("../fonts/Lato-Regular.woff2") format("woff2"), url("../fonts/Lato-Regular.woff") format("woff"), url("../fonts/Lato-Regular.ttf") format("truetype"), url("../fonts/Lato-Regular.svg#Lato-Regular") format("svg");

font-weight: normal;

font-style: normal;

}

@font-face {

font-family: 'BebasNeue-Bold';

src: url("../fonts/BebasNeue-Bold.eot");

src: url("../fonts/BebasNeue-Bold.eot?#iefix") format("embedded-opentype"), url("../fonts/BebasNeue-Bold.woff2") format("woff2"), url("../fonts/BebasNeue-Bold.woff") format("woff"), url("../fonts/BebasNeue-Bold.ttf") format("truetype"), url("../fonts/BebasNeue-Bold.svg#BebasNeue-Bold") format("svg");

font-weight: bold;

font-style: normal;

}

body {

font-family: "Lato-Regular", sans-serif;

min-width: 320px;

overflow-x: hidden;

position: relative;

}

a {

display: inline-block;

text-decoration: none;

color: #fff;

}

a:hover, a:focus, a:active {

text-decoration: none;

color: #fff;

outline: none;

}

h1,

h2,

h3,

h4,

h5,

h6,

p {

margin: 0;

line-height: normal;

}

ul {

margin: 0;

padding: 0;

}

input,

button,

textarea,

select {

outline: none;

}

img {

max-width: 100%;

height: auto;

}

.btn {

background: -webkit-gradient(linear, left top, right top, from(#FFA800), to(#FF3C00));

background: -webkit-linear-gradient(left, #FFA800, #FF3C00);

background: linear-gradient(to right, #FFA800, #FF3C00);

-webkit-border-radius: 30px;

border-radius: 30px;

padding: 18px 70px;

line-height: 1;

font-family: "Lato-Bold", sans-serif;

font-size: 16px;

text-transform: uppercase;

-webkit-transition: .5s;

transition: .5s;

-webkit-transform: perspective(0) translateZ(0);

transform: perspective(0) translateZ(0);

border: none;

}

.btn:hover {

-webkit-transform: scale(1.1);

-ms-transform: scale(1.1);

transform: scale(1.1);

}

.s-title {

font-size: 48px;

font-family: "BebasNeue-Bold", sans-serif;

font-weight: bold;

line-height: 1.1;

text-transform: uppercase;

text-align: center;

margin-bottom: 70px;

}

/\* Main section \*/

.main {

background: url("../img/main-bg.jpg") no-repeat top;

background-size: cover;

}

.main-suptitle {

font-size: 26px;

font-family: "Lato-Bold", sans-serif;

line-height: 1.2;

color: #fff;

margin-bottom: 50px;

margin-top: 95px;

}

.main-title {

font-size: 55px;

font-weight: normal;

color: #fff;

margin-bottom: 20px;

}

.main-title span {

display: block;

font-family: "Lato-Black", sans-serif;

font-size: 58px;

text-transform: uppercase;

background: -webkit-gradient(linear, left top, right top, from(#FFA800), to(#FF3C00));

background: -webkit-linear-gradient(left, #FFA800, #FF3C00);

background: linear-gradient(to right, #FFA800, #FF3C00);

-webkit-background-clip: text;

-webkit-text-fill-color: transparent;

}

.main-desc {

font-size: 24px;

color: #fff;

line-height: 1.4;

margin-bottom: 70px;

}

.main-info {

display: -webkit-box;

display: -webkit-flex;

display: -ms-flexbox;

display: flex;

-webkit-box-align: center;

-webkit-align-items: center;

-ms-flex-align: center;

align-items: center;

}

.main-date {

margin-left: 50px;

font-size: 22px;

font-family: "Lato-Bold", sans-serif;

text-transform: uppercase;

line-height: 1.2;

background: -webkit-gradient(linear, left top, right top, from(#FFA800), to(#FF3C00));

background: -webkit-linear-gradient(left, #FFA800, #FF3C00);

background: linear-gradient(to right, #FFA800, #FF3C00);

-webkit-background-clip: text;

-webkit-text-fill-color: transparent;

}

.main-date span {

display: block;

background: -webkit-gradient(linear, left top, right top, from(#fff), to(#fff));

background: -webkit-linear-gradient(left, #fff, #fff);

background: linear-gradient(to right, #fff, #fff);

-webkit-background-clip: text;

-webkit-text-fill-color: transparent;

}

# ДОДАТОК Б

# ВИХІДНІ КОДИ РЕСУРСУ — BACKEND

**mail-a.php**

<?php

$recepient = "maxdovbysh13019@gmail.com";

$sitename = "Indischool";

$name = trim($\_POST["name"]);

$phone = trim($\_POST["phone"]);

$message = "Имя: $name \nТелефон: $phone";

$pagetitle = "Обратный звонок с сайта \"$sitename\"";

mail($recepient, $pagetitle, $message, "Content-type: text/plain; charset=\"utf-8\"\n From: $recepient");

**mail-b.php**

<?php

$recepient = "maxdovbysh13019@gmail.com";

$sitename = "Indischool";

$name = trim($\_POST["name"]);

$email = trim($\_POST["email"]);

$question = trim($\_POST["question"]);

$message = "Имя: $name \nПочта: $email \nВопрос(ы): $question";

$pagetitle = "Новые вопросы с сайта \"$sitename\"";

mail($recepient, $pagetitle, $message, "Content-type: text/plain; charset=\"utf-8\"\n From: $recepient");

**mail-c.php**

<?php

$recepient = "maxdovbysh13019@gmail.com";

$sitename = "Indischool";

$name = trim($\_POST["name"]);

$email = trim($\_POST["email"]);

$phone = trim($\_POST["phone"]);

$message = "Имя: $name \nПочта: $email \nТелефон: $phone";

$pagetitle = "Новая бронь на курс с сайта \"$sitename\"";

mail($recepient, $pagetitle, $message, "Content-type: text/plain; charset=\"utf-8\"\n From: $recepient");