

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ ХАРЧОВИХ ТЕХНОЛОГІЙ

Інститут (факультет) АКС
 Кафедра Інформаційних систем
 Освітній ступінь бакалавр
 Спеціальність 122 “Комп’ютерні науки”
(код і назва)
 Освітньо-професійна програма Комп’ютерні науки
(назва)

ЗАТВЕРДЖУЮ

Завідувач кафедри **Чумаченко С.В**

“ ” _____ 20__ року

З А В Д А Н Н Я

НА КВАЛІФІКАЦІЙНУ РОБОТУ ЗДОБУВАЧА

Білоус Євгеній Сергійович

(прізвище, ім’я, по батькові)

1. Тема роботи Розроблення сайту для фірми "Qubit Labs"

керівник роботи Бобрівник Катерина Євгенівна доц.

(прізвище, ім’я, по батькові, науковий ступінь, вчене звання)

затверджені наказом закладу вищої освіти від “27” квітня 2021 року №269кв

2. Строк подання здобувачем роботи 31 травня 2021 р.

3. Вихідні дані до роботи Данні про клієнта, Замовлення на сайті

4. Зміст пояснювальної записки (перелік питань, які потрібно розробити)

1) Аналіз фірми Qubit Labs та постановка задачі на розробку сайту

2) Технічне завдання на розробку сайту

3) Розробка комплексу задач автоматизації

4) Охорона праці

5. Перелік графічного матеріалу

1) Функціональна модель діяльності компанії "Qubit Labs"

2) Логічна і фізична модель структури БД Сайту

6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
1	доц. Бобрівник Катерина Євгенівна	13.01.2021	5.04.2021
2	доц. Бобрівник Катерина Євгенівна	13.01.2021	10.05.2021
3	доц. Бобрівник Катерина Євгенівна	13.01.2021	10.06.2021
4	доц. Бобрівник Катерина Євгенівна	13.01.2021	28.06.2021

7. Дата видачі завдання 25 березня 2021 р.

КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів виконання кваліфікаційної роботи	Строк виконання етапів роботи	Примітка
	Дослідження діяльності фірми Qubit Labs	13.03.2021-5.04.2021	Виконано
	Аналіз аналогів систем управління контентом	5.04.2021-10.04.2021	Виконано
	Розрахунок економічного ефекту	12.04.2021-15.04.2021	Виконано
	Розробка Технічного завдання	20.04.2021-28.04.2021	Виконано
	Розробка функцій системи	1.05.2021-20.05.2021	Виконано
	Інструкція користувача	23.05.2021-26.05.2021	Виконано
	Охорона праці	1.06.2021-15.06.2021	Виконано
	Оформлення роботи та розробка презентації	15.06.2021-28.06.2021	Виконано

Здобувач _____
(підпис)

Білоус.Є.С _____
(прізвище та ініціали)

Керівник роботи _____
(підпис)

Бобрівник.К.Є _____
(прізвище та ініціали)

АНОТАЦІЯ

Розроблення сайту для компанії Qubit Labs

Кваліфікаційна робота містить: 68 сторінок, 4 розділи, 37 рисунків, 7 таблиць, 3 додатки та 20 літературних джерел.

Спеціальність: 122 «Комп'ютерні науки»

Метою кваліфікаційної роботи є покращення управління проектами компанії. Клієнти зможуть авторизуватись на сайті, і отримати всю інформацію що до своїх діючих проектів. Призначенням Сайту є зручне управління проектами та збільшення обсягу клієнтів.

У кваліфікаційній роботі: проведено аналіз компанії, обґрунтовано доцільність розроблення сайту, розроблено вимоги до сайту та його функцій, розроблено інтерфейс користувача для взаємодії з сайтом, розроблені функції для зручного управління.

Результатом виконання кваліфікаційної роботи є повноцінний Сайт для управління проектами.

КЛЮЧОВІ СЛОВА : ВЕБ-ДОДАТОК, ІНТЕРФЕЙС КОРИСТУВАЧА, САЙТ, QUBIT LABS, УПРАВЛІННЯ ПРОЕКТАМИ

ANNOTATION

Website development for Qubit Labs. Qualification work contains: 68 pages, 4 sections, 37 figures, 7 tables, 3 appendices and 20 references. Specialty: 122 "Computer Science"

The purpose of the qualification work is to improve the company's project management. Clients will be able to log in to the site and get all the information about their existing projects. The purpose of the Site is convenient project management and increasing the number of clients.

In the qualification work: the analysis of the company is carried out, the expediency of site development is substantiated, the requirements to the site and its functions are developed, the user interface for interaction with. site, developed features for easy management. The result of the qualification work is a full-fledged site for project management.

KEY WORDS: WEB APPLICATION, USER INTERFACE, SITE, QUBIT LABS, PROJECT MANAGEMENT

ЗМІСТ

АНОТАЦІЯ	4
ВСТУП	8
РОЗДІЛ 1. АНАЛІЗ ПІДПРИЄМСТВА «QUBIT LABS» ТА ПОСТАНОВКА ЗАДАЧІ НА РОЗРОБКУ САЙТУ	10
1.1 Опис та особливості функціонування «qubit labs»	10
1.2 Відділи підприємства «qubit labs».....	12
1.3 Моделювання діяльності підприємства.....	14
1.4 Концептуальна модель to-be	17
1.5 Аналіз аналогів систем управління контентом.....	19
1.5.1 Характеристика фреймворку “react”	21
1.5.2 Характеристика фреймворку angular.....	23
1.5.3 Характеристика фреймворку vue.js	24
1.5.4 Порівняльна характеристика фреймворків	25
1.6 Постановка задачі на розробку сайту.....	26
1.6.1 Призначення системи.....	26
1.6.2 Ціль створення сайту	26
1.6.3 Користувачі сайту	26
1.6.4 Вхідні та вихідні дані сайту	27
1.7 Розрахунок економічного ефекту після впровадження Сайту	27
РОЗДІЛ 2. ТЕХНІЧНЕ ЗАВДАННЯ НА РОЗРОБКУ САЙТУ	31
2.1 Призначення та мета створення сайту qubit-labs.com	31
2.1.1 Призначення	31
2.1.2 Мета сайту	31
2.1.3 Цільова аудиторія сайту	31
2.1.4 Основні завдання сайту	31
2.2 Загальні вимоги до сайту.....	32
2.2.1 Вимоги до стилістичному оформленню сайту	32
2.2.2 Вимоги до графічного дизайну сайту.....	32
2.2.3 Вимоги до шрифтового оформлення сайту	32
2.2.4 Браузери	32
2.2.5 Вимоги до безпеки даних	33
2.2.6 Вимоги до засобів перегляду сайту	33
2.2.7 Вимоги до контенту і наповненню сайту	33
2.2.8 Інші вимоги до контенту	33
2.2.9 Вимоги до збереження інформації.....	33
2.2.10 Вимоги до кількості відвідувачів.....	33
2.3 Вимоги до шаблону сайту	33
2.3.1 Елементи навігації.....	33
2.3.2 Сторінка «home».....	34
2.3.3 Сторінка «about».....	34
2.3.4 Сторінка «project»	34
2.3.5 Сторінка «blog»	34
2.3.6 Сторінка «contact us»	34
2.3.7 Блок «авторизації»	34
2.3.8 Блок «реєстрації»	35
2.3.9 Форма «зворотній зв'язок».....	35
2.4 Функції системи.....	35
2.4.1 Вимоги до функцій системи.....	35
2.5 Вимоги до технічного забезпечення сайту.....	36

2.6	Склад і зміст робіт.....	37
2.7	Вимоги до складу та змісту робіт по введенню сайту в експлуатацію.....	38
РОЗДІЛ 3. РОЗРОБКА КОМПЛЕКСУ ЗАДАЧ АВТОМАТИЗАЦІЇ		39
3.1	Обґрунтування вибору використаних програмних засобів	39
3.2	Логічна модель БД.....	40
3.3	Створення інтерфейсу	43
3.4	Інструкція користувача	48
РОЗДІЛ 4. ОХОРОНА ПРАЦІ		58
4.1	Техніка безпеки підприємства	58
ГАРАНТІЇ ПРАВ НА ОХОРОНУ ПРАЦІ		61
4.2	Програміст зобов'язаний	64
4.3	Загальні вимоги безпеки під час виконання роботи	66
4.4	Вимоги безпеки після закінчення роботи.....	66
4.5	Вимоги безпеки в аварійній ситуації.....	66
ВИСНОВКИ		69
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ		70
Додатки		71
Додаток А Vp-win Модель		71
Додаток Б. Логічна та фізична модель БД		72
Додаток В Компонентний підхід розробки.....		73

ВСТУП

Підприємство ТОВ «Qubit Labs» - це компанія яка займається підбором персоналу для вашої ІТ ідеї, та розробкою Web додатків. Кожна команда в нашій компанії унікальна і підбирається виходячи з вимог і потреб клієнта; команду можна масштабувати для роботи над додатковими завданнями або проектами, а розробники працюють з вашою компанією, як з вашої внутрішньої командою. Ви можете зосередити увагу на кожному аспекті розробки програмного забезпечення і контролювати його. Поки ви керуєте і контролюєте процес, ми ведемо бек-офіс, здійснюємо набір, вирішуємо адміністративні питання, управляємо кадровими процесами, бухгалтерським обліком і юридичними питаннями. Ви можете найняти індивідуальних розробників додатків, щоб ефективно розширити свою власну команду, або легко створити свою власну віддалену команду. Розробники будуть працювати з нашого офісу. Ви будете керувати командою безпосередньо за допомогою ваших улюблених інструментів і методологій. Головне правило прийняття рішень полягає в тому, що наші клієнти і віддані їм співробітники є вищим пріоритетом. Ми допомагаємо видатним людям знаходити відмінні компанії і знаходити синергію в роботі для продуктивного результату.

Об'єктом який досліджується в даній роботі є компанія «Qubit Labs» та її потреба в створенні сайту для управління проектами.

Метою розробленого сайту є покращення управління проектами компанії. Клієнти зможуть авторизуватись на сайті, і отримати всю інформацію що до своїх діючих проектів.

Щоб досягти поставленої мети буде використовуватися фреймворк React.

Моя Кваліфікаційна робота складається зі вступу, чотирьох розділів та висновка;

В 1 розділі наведено загальну інформацію про компанію, було обґрунтовано доцільність розробки. проаналізовано ринок, наведено організаційну структуру компанії.

У 2 розділі було складене Технічне завдання на розробку, визначення призначення сайту, загальні вимоги до сайту, вимоги до шаблону сайту, перелік функцій та вимоги до них .

У 3 розділі описано створення логічних та фізичних моделей БД, описано створення інтерфейсу користувача та розроблення функцій.

У 4 розділі описано заходи із Охорони праці Техніка безпека при роботі в офісі, та загальні вимоги до програміста.

У заключній частині наведено висновки, література та додатки.

РОЗДІЛ 1. АНАЛІЗ ПІДПРИЄМСТВА «QUBIT LABS» ТА ПОСТАНОВКА ЗАДАЧІ НА РОЗРОБКУ САЙТУ

1.1 Опис та особливості функціонування «qubit labs»

Qubit Labs - українська офшорна компанія з розробки програмного забезпечення з центром досліджень і розробок в Києві, де ви можете найняти команду розробників для завершення вашого проекту. Наш відділ підбору персоналу може вибрати досвідчених програмістів з великого кадрового резерву, а потім ми допоможемо вам з інтеграцією, адаптацією, розрахунком заробітної плати та бек-офісом. У нас є досвід роботи в різних послугах: розробка програмного забезпечення, створення незалежного науково-дослідного підрозділу, технологічний консалтинг для компаній з різних країн. Ми використовуємо ці знання, щоб надати нашим клієнтам кращі послуги, щоб виправдати їх очікування. [1]

Qubit Labs - компанія, яка має великий досвід у наймі розробників для різних бізнес-моделей. Професіонали компанії можуть підібрати для вас першокласних програмістів, будь то мобільна або веб-розробка, ігровий дизайн або FinTech. Компанія надає юридичну підтримку, піклується про заробітну плату і адаптації, а також забезпечує безпроблемні і прозорі процеси найму.

Qubit Labs починалася як невелика компанія з розробки програмного забезпечення, в якій працює всього пара експертів, але за 1,5 року ми швидко вирости. Тепер в нашому портфоліо є успішні кейси постійної налаштування команди для міжнародних транспортних, туристичних, ріелторських, страхових, ігрових і освітніх компаній. За допомогою нашого сервісу клієнти можуть трансформувати традиційну концепцію ІТ-аутсорсингу. Розробка програмного забезпечення аутстаффірованою командою схожа на те, що ваш власний центр розробки в Україні працює спеціально над вашим проектом.

Основна філософія бізнесу полягає в тому, щоб забезпечити зручність і доступність послуг з розробки програмного забезпечення в Україні для всіх наших клієнтів. Працюючи з компаніями з усього світу і надаючи різні послуги

IT-аутсорсингу, ми трансформуємо модель обслуговування через більш гнучкі та вигідні умови для всіх зацікавлених сторін. Таким чином, ми вибрали модель створення команди розробників з нуля для унікальних вимог кожного нового клієнта як кращу як для клієнтів, так і для розробників.

Компанія є міжнародною там має свої офіси у таких містах та за такими адресами:

ЕСТОНІЯ

- Seraraja tn 6, Таллінн, 15551
- + 1-646-233-1020

УКРАЇНА

- 7 Красилівська, Київ, Україна, 02000
- + 1-646-233-1020

КАНАДА

- 6 Forest Laneway, Торонто, ON M2N 5X9
- + 1-646-233-1020

Компанія Qubit Labs активно співпрацює з компаніями, які шукають надійних технологічних партнерів для своїх клієнтів. Для цього ми створили спеціальну Партнерську програму розвитку бізнесу, в рамках якої ви можете направляти своїх клієнтів в Qubit Labs на взаємовигідних умовах.

Qubit Labs пов'язує вас з кращими технічними талантами України, і чи хочете ви найняти веб-розробника або розширити свою місцеву команду за допомогою офшорних ресурсів, ми можемо допомогти. Ваші програмісти працюватимуть з нашого офісу, але ви можете контролювати процес розробки самостійно.

Крім того, ми готові створити офшорний центр розробки для вашої компанії та найняти спеціалізованих веб-розробників та інших фахівців, які допоможуть вам з аналізом вимог, призначеними для користувача веб-рішеннями, архітектурою веб-сайту, дизайном користувальницького інтерфейсу

/ UX веб-сайту, контрольним якістю і тестуванням, обслуговуванням, системним адмініструванням і підтримкою.

Ви можете найняти індивідуальних розробників додатків, щоб ефективно розширити свою власну команду, або легко створити свою власну віддалену команду. Розробники будуть працювати з нашого офісу. Ви будете керувати командою безпосередньо за допомогою ваших улюблених інструментів і методологій.

Найміть розробників додатків для Android / iOS, щоб розширити свою місцеву команду фахівцями з офшору або створити команду керованого програмного забезпечення в офшорі. Qubit Labs також може допомогти вам створити свій RnD-офіс з провідними розробниками, РМ і дизайнерами UX / UI, які будуть працювати над вашим проектом. Ми надаємо вам спеціальну команду відповідно до ваших вимог, використовуючи наші внутрішні ресурси і можливості підбору персоналу.

1.2 Відділи підприємства «qubit labs»

На підприємстві діє “лінійна структура управління”. Такі підприємства виникають як організації з простою структурою. У плануванні організації відображається її власник у ролі президента з усіма службовцями, які безпосередньо йому підпорядковуються. Проста (лінійна) структура визначається не наявністю, а відсутністю в ній будь-яких характеристик. Маємо низький рівень спеціалізації, нечисленні правила, за якими здійснюється діяльність, та централізацію повноважень у руках однієї особи - власника.

Проста організація є "малоступеневою". Вона переважно складається з двох-трьох вертикальних рівнів та аморфної групи уповноважених службовців, які мають право централізовано приймати рішення. Правила простої структури поступаються більш формалізованим правилам зі зростанням обсягів виробництва або продажу, зі збільшенням чисельності працівників та управлінського персоналу. В цьому випадку формуються нові рівні менеджменту для координування діяльності підприємства.

Схема управління на підприємстві (див. рис 1.1)

Головним є Директор, йому підпорядковуються 4 відділи:

- Відділ роботи з клієнтами
- Відділ кадрів
- Відділ розробки
- Відділ інфраструктури

Відділу роботи з клієнтами підпорядковуються:

- Менеджер

Відділу кадрів підпорядковуються:

- HR-менеджер
- Рекрутер

Відділу розробки підпорядковуються:

- Програмісти



Рис. 1.1 Організаційна структура підприємства

Нижче описані самі головні відділи підприємства, а саме, їхні функції та взаємозв'язок з іншими відділами підприємства.

Таблиця 1.1 Роль і взаємодія відділів підприємства

Відділ	Роль і взаємодія
--------	------------------

Відділ роботи з клієнтами	Відділ відповідальний за роботу з клієнтами, ведення документації, заключення договорів. Збирає інформацію щодо нинішніх потреб клієнтів.
Відділ кадрів	У цьому відділі проходить набір людей. Потік кадрів фільтрується від активних проектів на даний час. Активних проектів як правило більше 10, тому вакансії оновлюються майже кожного тижня.
Відділ розробки	Відділ який безпосередньо відповідає за розробку проектів. До компанії можливо звернутися и метою не тільки сформувати команду, яка буде займатися вашою ідеєю, а й з метою розробити програмний продукт.
Відділ інфраструктури	Відповідальний за розміщення офісів та їх благоустрою. Забезпеченням офісів продуктами, меблями, технікою.

1.3 Моделювання діяльності підприємства

Для формування конкретної задачі, було детально вивчено діяльність підприємства, це надало нам можливість точно визначити перелік вхідних, вихідних даних (рис. 1.2 – рис 1.3).

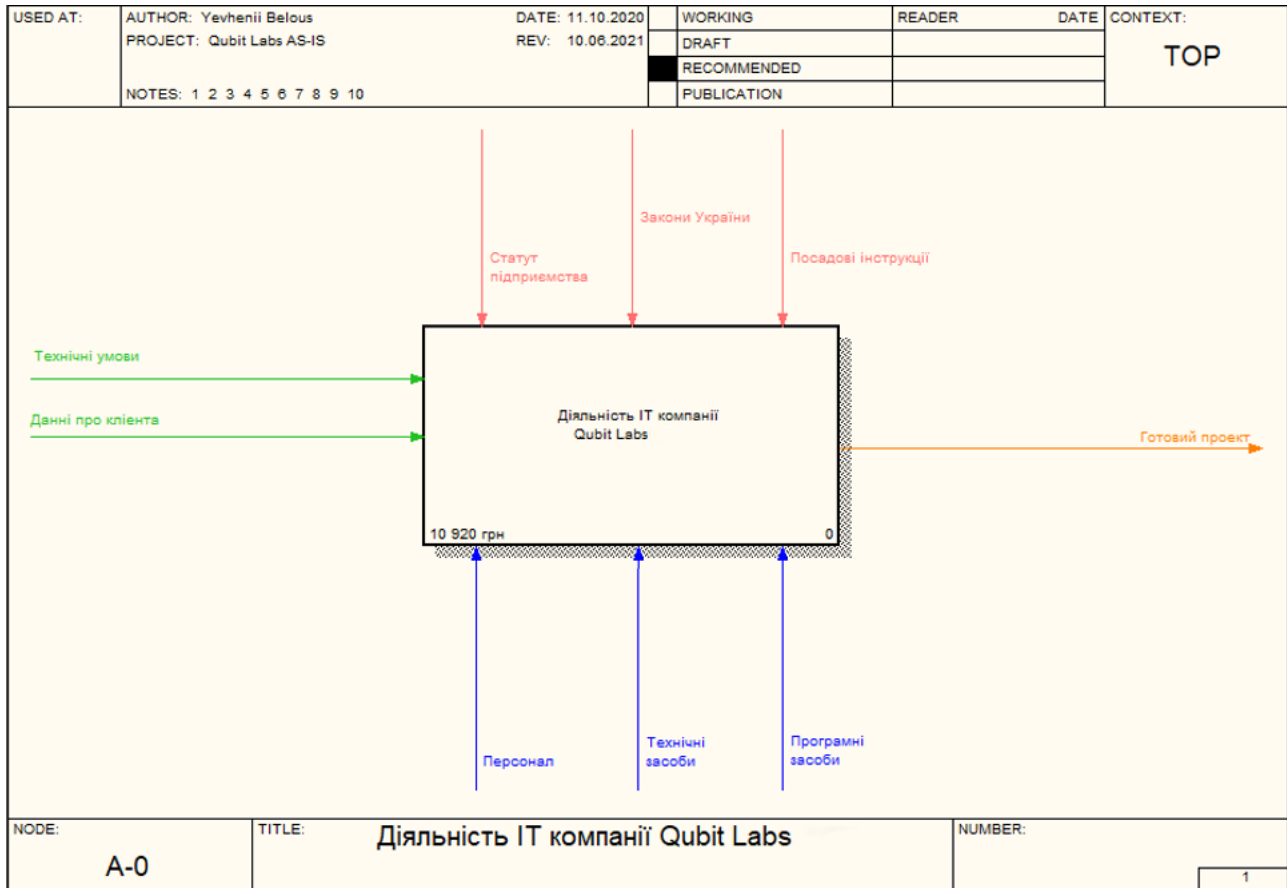


Рис. 1.2 Модель діяльності підприємства AS-IS

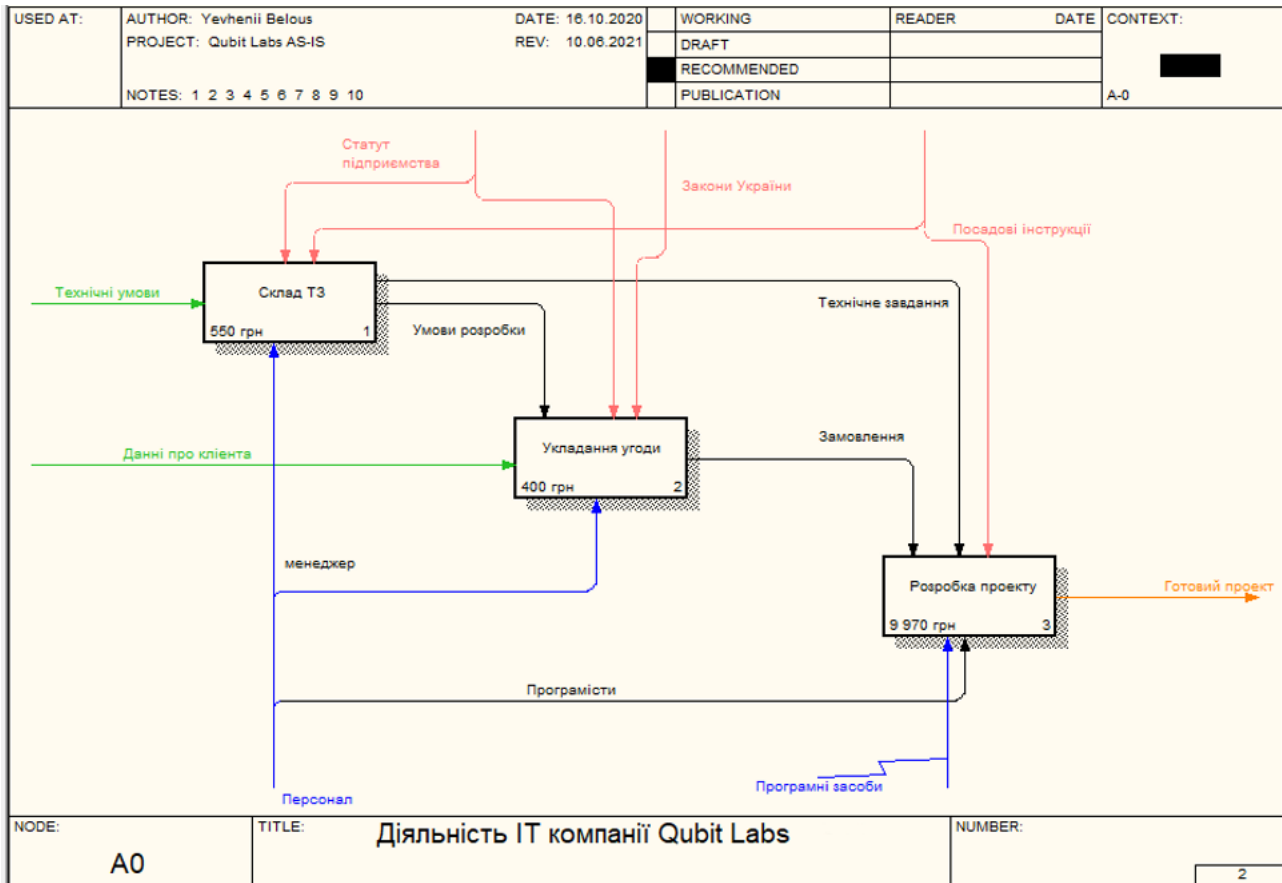


Рис. 1.3 Декомпозиція моделі AS-IS

Після, аналізу отримуємо загальні відомості про систему:
Вхідними даними діяльності є:

- Технічні умови;
- Данні про клієнтів

Вся діяльність виконується під управлінням:

- Статуту підприємства;
- Посадові інструкції
- Законів України

Механізмами системи є:

- Персонал
- Технічні засоби
- Програмні засоби

Результатами діяльності є:

- Готовий проект

Проблеми які були виявлені при аналізі

- Клієнт не мав власного кабінету
- Клієнт не мав змоги переглянути свої проекти та інформацію про них
- Клієнт не міг змінити інформацію про проект та видалити його
- Клієнт не міг переглянути останні новини сайту

1.4 Концептуальна модель to-be

Після аналізу підприємства та виявлення проблем, була сформована концептуальна модель TO-BE (рис. 1.4 - рис 1.5). Додаток (А)

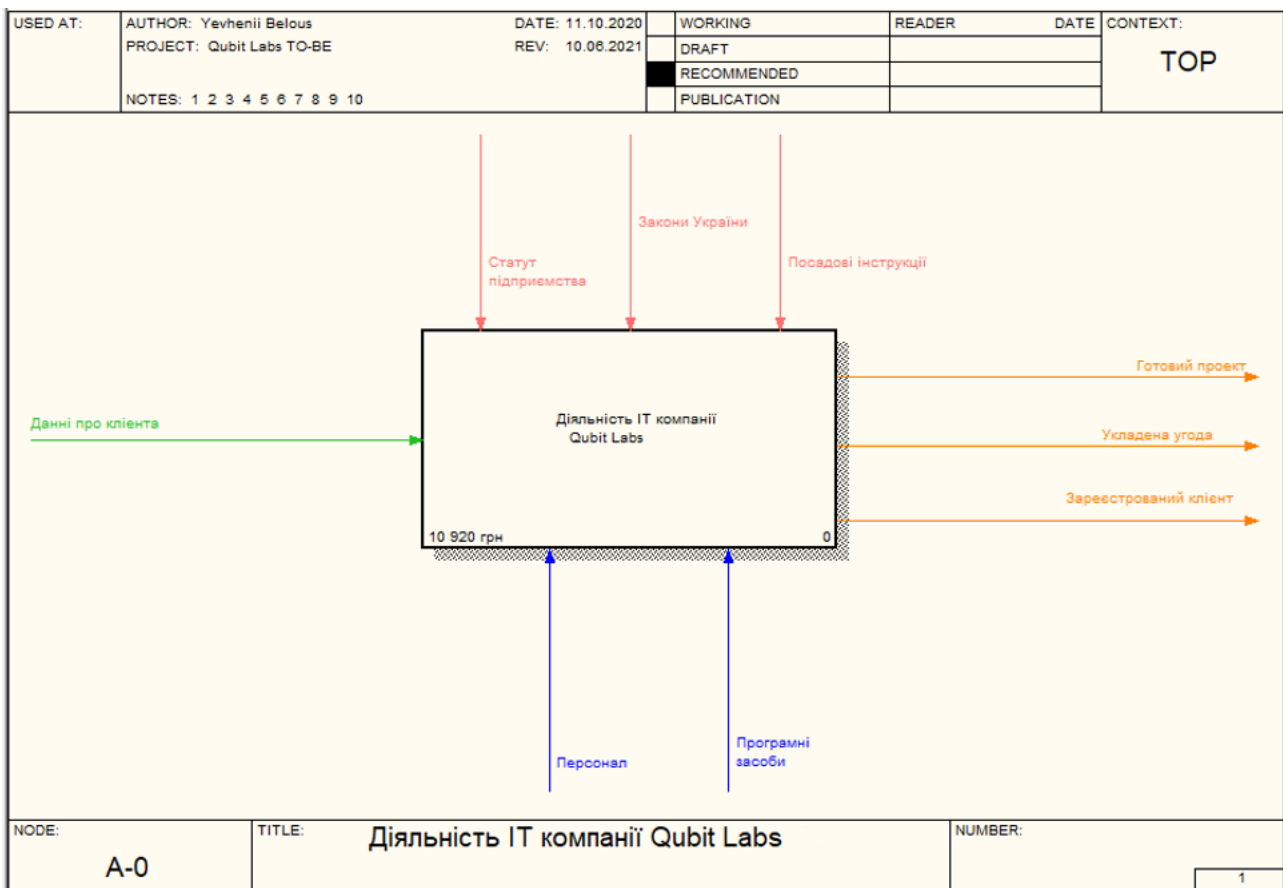


Рис. 1.4 Концептуальна модель TO-BE

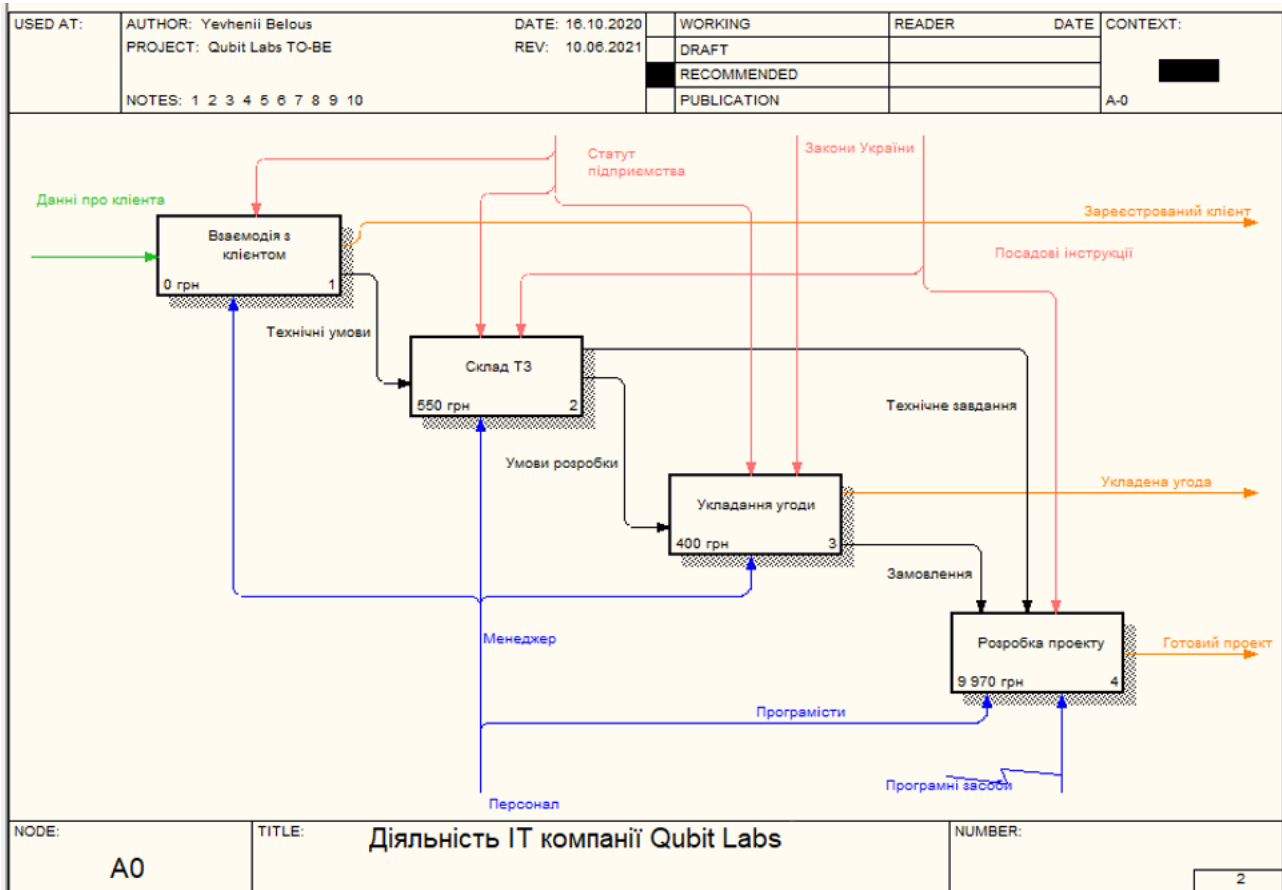


Рис. 1.5 Декомпозиція моделі TO-BE

Вхідними даними діяльності є:

- Данні про клієнтів

Вся діяльність виконується під управлінням:

- Статуту підприємства;
- Посадові інструкції
- Законів України

Механізмами системи є:

- Персонал
- Програмні засоби

Результатами діяльності є:

- Готовий проект
- Укладена угода
- Зареєстрований клієнт

1.5 Аналіз аналогів систем управління контентом

Система керування вмістом - програмне забезпечення для організації веб-сайтів чи інших інформаційних ресурсів в Інтернеті чи окремих комп'ютерних мережах. Існують сотні, а може, навіть й тисячі доступних CMS — систем. Завдяки їх функціональності ці системи можна використовувати в різних компаніях. Незважаючи на широкий вибір інструментальних та технічних засобів, наявних в CMS, існують загальні для більшості типів систем характеристики.

Перші СКВ були розроблені у великих корпораціях для організації роботи з документацією. У 1995-му від компанії CNET відокремилася окрема компанія Vignette, яка започаткувала ринок для комерційних СКВ. З часом діапазон продукції розширювався і дедалі більше інтегрувався у сучасні мережеві рішення аж до популярних веб-порталів. Багато сучасних СКВ поширюються як безкоштовні і легкі у встановленні (інсталяції) програми, які розробляються під ліцензією GNU/GPL групами ентузіастів.

Системи управління веб-сайтом часто розраховані на роботу у певному програмному середовищі. Наприклад, система MediaWiki, під управлінням якої працює Вікіпедія, написана мовою програмування PHP і зберігає вміст і налаштування у базі даних типу MySQL або PostgreSQL; тому для її роботи потрібно, щоб на сервері, де вона розміщена, були встановлені веб-сервер (Apache, IIS чи інший), підтримка PHP та системи керування базами даних MySQL або PostgreSQL, а також, в разі необхідності, додаткові програми для обробки зображень чи математичних формул. Такі вимоги є досить типовими для відкритих СК.

Рейтинг безкоштовних CMS систем станом на 2021 рік. (рис 1.4)

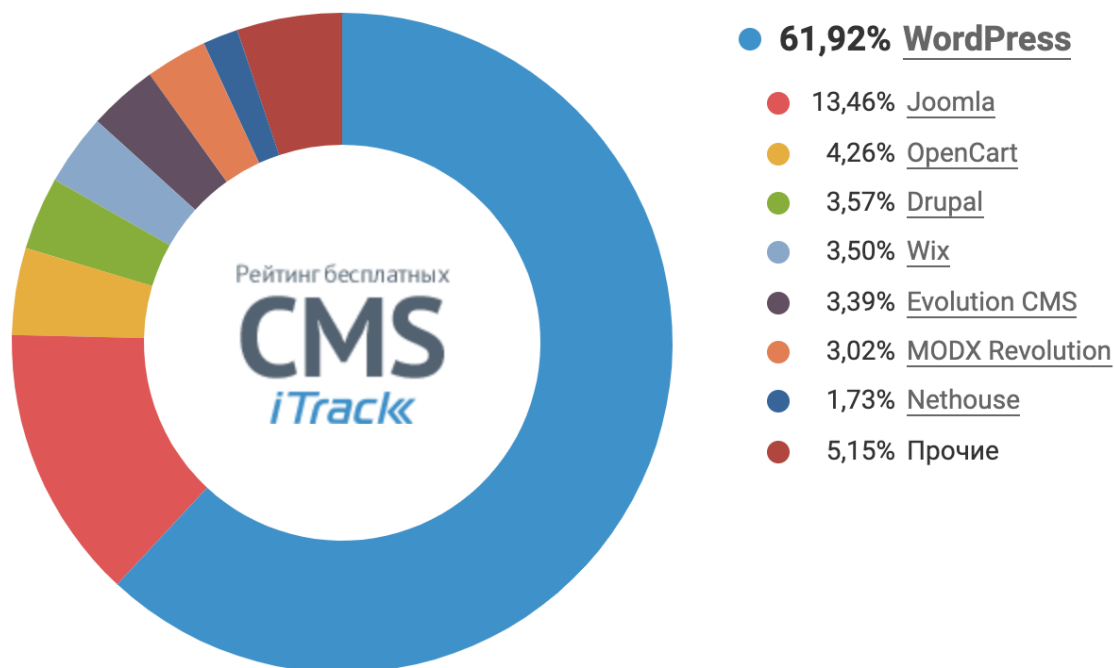


Рис. 1.6 Рейтинг безкоштовних CMS систем

Інший затребуваний спосіб розробки, це Розробка проектів в ручну. Цей спосіб є досить складним але він дає нам можливість зробити набагато більше ніж за допомогою CMS. Для розробки сайту використовують:

HTML - мова тегів, за допомогою якої здійснюється створення веб-сторінок для мережі Інтернет. HTML лише описує структуру веб-сторінки семантично. Елементи HTML можна розглянути як будівельні блоки сторінок HTML. [3].

CSS (Cascading Style Sheets) - це код, який ви використовуєте для стилізації вашої веб-сторінки. Основи CSS допоможуть вам зрозуміти, що вам потрібно для початку роботи. Ми відповімо на такі питання як: Як зробити мій текст чорним або червоним? Як зробити так, щоб контент з'являвся в певному місці на екрані? Як прикрасити веб-сторінку за допомогою фонових зображень і кольорів [4].

JavaScript® (часто просто JS) – це логічна мова програмування, використовується для описання логіки сайту, та динамічної розробки. Але також використовується і в інших програмних продуктах, наприклад, node.js або Apache CouchDB. JavaScript це прототипно-орієнтований, мова з динамічною типізацією, який підтримує об'єктно-орієнтована, імперативний і декларативний (наприклад, функціональне програмування) стилі програмування. [5]

Фреймворки JS - це бібліотеки програмування JavaScript, в яких є попередньо написаний код для використання в стандартних функціях і завданнях програмування. Це основа для створення веб-сайтів або веб-додатків навколо.

Давайте почнемо з того, навіщо нам потрібні фреймворки JavaScript? Кодування цілком можливо без їх використання, але правильно підібрана середовище може значно полегшити роботу. Більш того, вони безкоштовні і з відкритим вихідним кодом, так що немає ризику. [6]

Найбільш популярні на сьогодні фреймворки js:

- React
- Angular
- Vue

1.5.1 Характеристика фреймворку “react”

React - це декларативна, ефективна і гнучка JavaScript бібліотека для створення користувацьких інтерфейсів. Вона дозволяє вам збирати складний UI з маленьких ізольованих шматочків коду, званих «компонентами». React може використовуватися для розробки односторінкових і мобільних додатків. Його мета - надати високу швидкість, простоту і масштабованість. Як бібліотеки для розробки призначених для користувача інтерфейсів React часто використовується з іншими бібліотеками, такими як MobX, Redux і GraphQL. [8]

Переваги React [7]:

- Легко вивчити, завдяки простому дизайну, використання JSX (HTML-подібний синтаксис) для шаблонів і дуже докладної документації. Розробники витрачають більше часу на написання сучасного JavaScript і менше турбуються про код, специфічному для фреймворка.
- Дуже швидка, завдяки реалізації React Virtual DOM і різним оптимізаціям рендеринга.
- Відмінна підтримка рендеринга на стороні сервера, що робить його потужною платформою для контент-орієнтованих додатків.
- Redux, найпопулярніша платформа для управління станом додатків в
- React реалізує концепції функціонального програмування (FP), створюючи простий в тестуванні і багаторазово використовуваний код.
- Додатки можуть бути створені за допомогою TypeScript або Facebook's Flow, що мають вбудовану підтримку JSX.
- Навички, отримані в React, можуть бути застосовані до розробки на React Native.

Недоліки React [7]:

- React не однозначний і залишає розробникам можливість вибрати кращий спосіб розвитку. Це може бути вирішено сильним лідерством проекту і хорошими процесами.
- Спільнота ділиться по способам написання CSS в React, які поділяються на традиційні таблиці стилів (CSS Modules) і CSS-in-JS (тобто Emotion і Styled Components).
- React відходить від компонентів на основі класів, що може стати перешкодою для розробників, яким більш комфортно працювати з об'єктно-орієнтованим програмуванням (ООП).
- Змішування шаблонів з логікою (JSX) може збити з пантелику деяких розробників при перших знайомствах з React.

Компанії, які використовують React: Facebook, Instagram, Netflix, New York Times, Yahoo, Khan Academy, Whatsapp, Codecademy, Dropbox, Airbnb, Asana, Atlassian, Intercom, Microsoft, Slack, Storybook і багато інших

1.5.2 Характеристика фреймворку angular

AngularJS – JavaScript - фреймворк з відкритим кодом. Призначений для розробки односторінкових додатків. Його мета - розширення браузерних додатків на основі MVC-шаблону, а також спрощення тестування і розробки. Фреймворк працює з HTML, що містить додаткові атрибути, які описуються директивами, і пов'язує введення або виведення області сторінки з моделлю, яка представляє собою звичайні змінні JavaScript. Значення цих змінних задаються вручну або витягуються з статичних або динамічних JSON-даних. [9]

Переваги **Angular** [7]:

- Angular використовується разом з Typescript. Він має виняткову підтримку для цього.
- Angular-language-service - забезпечує інтелектуальні можливості і авто заповнення шаблону HTML-компонента.
- Нові функції, такі як generation Angular, що використовують бібліотеки npm з CLI, generation, і розробка компонентів, що використовує Angular.
- Детальна документація, що дозволяє розробнику отримати всю необхідну інформацію, не вдаючись до допомоги його колег. Однак це вимагає більше часу для навчання.

Недоліки **Angular** [7]:

- Різноманітність різних структур (Injectables, Components, Pipes, Modules і т. Д.) Ускладнює вивчення в порівнянні з React і Vue.js, у яких є тільки «Component».
- Відносно повільна продуктивність, враховуючи різні показники. З іншого боку, це можна легко вирішити, використовуючи так званий

«ChangeDetectionStrategy», який допомагає вручну контролювати процес рендеринга компонентів.

Компанії, які використовують Angular: Microsoft, Autodesk, MacDonald's, UPS, Cisco Solution Partner Program, AT & T, Apple, Adobe, GoPro, ProtonMail, Clarity Design System, Upwork, Freelancer, Udemy.

1.5.3 Характеристика фреймворку vue js

Vue.js - це JavaScript - фреймворк, заснований в 2013 році, який ідеально підходить для створення призначених для користувача інтерфейсів і складних одно сторінкових додатків. Розробники називають Vue.js прогресивний.[10].

Переваги Vue.js [7]:

- **Посилений HTML.** Це означає, що Vue.js має багато характеристик схожих з Angular, а це, завдяки використанню різних компонентів, допомагає оптимізації HTML- блоків.
- **Детальна документація.** Vue.js має дуже детальну документацію, яка може прискорити процес навчання для розробників і заощадити багато часу на розробку програми, використовуючи тільки базові знання HTML і JavaScript.
- **Адаптивність.** Може бути здійснений швидкий перехід від інших фреймворків до Vue.js через схожість з Angular і React з точки зору дизайну і архітектури.

Недоліки Vue.js [7]:

- **Недолік ресурсів.** Vue.js як і раніше займає досить невелику частку ринку в порівнянні з React або Angular, що означає, що обмін знаннями в цьому середовищі все ще перебуває на початковій стадії.
- **Ризик надмірної гнучкості.** Іноді у Vue.js можуть виникнути проблеми при інтеграції в величезні проекти, і поки ще немає

досвіду можливих рішень, але вони обов'язково з'являться найближчим часом.

Компанії, які використовують Vue.js: Xiaomi, Alibaba, WizzAir, EuroNews, Grammarly, Gitlab і Laracasts, Adobe, Behance, Codeship, Reuters.

1.5.4 Порівняльна характеристика фреймворків

Серед усіх представлених фреймворків, відповідно наших вимог нам підходить React. (див.таблиця 2.1)

Створювати інтерактивні інтерфейси на React - приємно і просто. Достатньо описати, як частини інтерфейсу програми відображаються в різних станах. React буде своєчасно їх оновлювати, коли дані змінюються. Декларативні подання зроблять код більш передбачуваним і спростять налагодження.

React Заснований на компонентах з власним станом, а потім об'єднуються в складні інтерфейси. Оскільки логіка компонента написана на JavaScript, а не міститься в шаблонах, можна з легкістю передавати найрізноманітніші дані по всьому додатком і тримати стан поза DOM. Розробляти нову функціональність на React, не змінюючи існуючий код. React також може працювати на сервері, використовуючи Node.js і на мобільних платформах, використовуючи React Native.

Таблиця 1.2 Порівняння характеристика фреймворків

Назва	Складність вивчення	Швидкість	Універсальність	Популярність
React	Легко	Висока	Веб-розробка та Мобільна розробка	Більше 50 тис бібліотек
Angular	Середня	Середня	Веб-розробка	Більше 35 тис бібліотек
Vue	Середня	Висока	Веб-розробка	Більше 15 тис бібліотек

Вимоги, на які орієнтуємось при розробці Сайту:

- Швидкість роботи
- Можливості фреймворку
- Універсальність
- Об'єм бібліотек

Отже, після порівняння фреймворків ми визначили що нам підходить React, він має дуже високу швидкість роботи. Легкий у вивченні, та при розробці на ньому простіше слідкувати за діями користувача на сайті. Кожен компонент має кілька «методів життєвого циклу».

1.6 Постановка задачі на розробку сайту

1.6.1 Призначення системи

Сайт "Qubit Labs", який є метою розробки, повинен мати такі функції:

- Швидкий зв'язок з адміністратором
- Реєстрація клієнта
- Авторизація клієнта
- Облік проектів
- Редагування та видалення інформації проектів
- Захист від випадкового видалення проекту
- Облік новин компанії
- Групування новин по категоріям

Веб додаток повинен мати зручний та функціональний інтерфейс.

1.6.2 Ціль створення сайту

Метою Сайту є покращення управління проектами компанії. Так як на даний момент у компанії немає сервісу який виконував би цю задачу. Клієнти зможуть авторизуватись на сайті, і отримати всю інформацію що до своїх діючих проектів.

1.6.3 Користувачі сайту

Користувачами сайту є:

- клієнти, замовники – вони мають доступ до інформації про свої проекти, можуть додавати інформацію у проекти, створювати нові та видаляти старі.

- менеджер – має доступ до усіх проектів, здійснює перевірку, та може підтвердити або відхилити проект, після створення його клієнтом .

1.6.4 Вхідні та вихідні дані сайту

Вхідними даними є інформація про клієнтів, проекти, статті компанії, загальна інформація про компанію.

Вихідними даними є: Сторінки сайту, який представляє компанію в інтернеті та дає можливість клієнту управляти, переглядати інформацію про свої проекти, зареєстрований клієнт, укладена угода.

1.7 Розрахунок економічного ефекту після впровадження Сайту

Техніко-економічний ефект визначається за співвідношенням витрат на розробку системи і прибутком від її впровадження.

Вихідні дані для розрахунку:

- Ступінь новизни розроблюваних задач – "В" – використання типових проектних рішень за умови їх змін.
- Група складності алгоритму – 1.
- Узагальнені дані вхідної та вихідної інформації Сайту «Qubit Labs» у табл.1.3.

Таблиця 1.3. Узагальнені дані

Вид інформації	Позначення	К-сть наборів даних
Змінна інформація	ЗІ	m = 5
Нормативно-довідкова інформація	НДІ	n = 5
База даних	БД	p = 1
Обробка в режимі реального часу	РЧ	так
Забезпечення телекомунікаційної обробки даних і управління віддаленими об'єктами	ТОУ	ні

Витрати часу на розробку ескізного проекту (передпроектного дослідження) T_1 і технічного завдання T_2 за даними:

Ескізний проект $T_1 = 50$

Технічне завдання $T_2 = 40$

Визначаються витрати часу на стадіях «технічний проект», «робочий проект» і «впровадження».

$$\overline{T_{Б3} = 55 \quad T_{Б4} = 130 \quad T_{Б5} = 40}$$

Визначення витрат часу для стадії «технічний проект» (T_3)

k_{Π} – коефіцієнт трудомісткості.

Формула:

$$k_{\Pi} = \frac{k_1 * m + k_2 * n + k_3 * p}{m + n + p} = \frac{1 * 5 + 0,72 * 5 + 2,08 * 1}{5 + 5 + 1} = 0,97$$

k_0 – коефіцієнт ступеню новизни проекту.

Технічний проект. ТООУ. 1,36.

Визначення витрат часу «технічний проект»

$$\overline{T_3 = T_{Б3} * k_{\Pi} * k_0 = 55 * 0,97 * 1,36 = 72,55}$$

Визначення витрат часу на стадії «робочий проект» (T_4)

11 – вхідні дані і документи різного формату і структури, контроль здійснюється перехресно, тобто враховується зв'язок між показниками різних документів.

22 – друк документів подібної форми та змісту, виведення масивів даних на машині носії.

Коефіцієнт складності контролю вхідної та вихідної інформації $\overline{k_c}$ (11 x 22) – 1,07.

$$k_{\Pi} = \frac{1,2 * 5 + 0,58 * 5 + 0,48 * 1}{5 + 5 + 1} = 0,85$$

k_0 – коефіцієнт ступеню новизни проекту.

Робочий проект. ТОУ. 1,41.

Визначення витрат часу «робочий проект»

$$T_4 = T_{Б4} * k_{п} * k_{О} * k_{С} = 130 * 0,85 * 1,41 * 1,07 = 166,71$$

Визначення витрат часу на стадії «впровадження» (T_5)

$$k_{п} = \frac{1,2 * 5 + 0,58 * 5 + 0,48 * 1}{5 + 5 + 1} = 0,85$$

$k_{О}$ – коефіцієнт ступеню новизни проекту.

Робочий проект. ТОУ. 1,26.

Визначення витрат часу «впровадження»

$$T_5 = T_{Б5} * k_{п} * k_{О} * k_{С} = 40 * 0,85 * 1,26 * 1,07 = 45,83$$

Для визначення загальних витрат часу на розробку системи використовується формула:

$$T_{Σ} = T_1 + T_2 + T_3 + T_4 + T_5 = 50 + 40 + 72,55 + 166,71 + 45,83 = 375,1 \text{ годин}$$

Визначення чисельності виконавців:

$$Ч = \frac{T_{Σ}}{\Phi} = \frac{375,1}{68} = 5,5 \text{ (6 виконавців)}$$

де Φ – кількість робочих днів на виконання проекту.

Витрати на оплату роботи програмістів

Якщо прийняти, що оплата програміста (Junior) рівня складає в розмірі 10 000 грн, то оплата праці всіх виконавців, яка підраховується за формулою складе:

$$V'_1 = Ч * М * ЗП_{пр} = 6 * 2 * 10\,000 = 120\,000 \text{ грн.}$$

Витрати, пов'язані з розробкою програми на ПК.

$T_{ОП} = 2400$

$$T_{ПК} = T_{ОП} - (6 * 8 + 5 * 12) = 2400 - (6 * 8 + 5 * 12) = 2292 \text{ годин}$$

$R = 450$ годин

$$T'_{ПК} = T_{ПК} * \frac{R}{T_{ОП}} = 2292 * \frac{450}{2400} = 429,75 \text{ годин}$$

Витрати на аренду сервера V_2 :

$$Ц_c = 1700/\text{м}$$

$$V_2 = Ц_c * 20 = 34000 \text{ грн}$$

Витрати на навчання персоналу V_3 :

Навчання адміністратора сайту триватиме 1 день, тому можна вважати, що: $V_3 = 1000$ грн.

Загальна вартість розробки і впровадження системи

Загальна вартість розробки і впровадження системи V_Σ , вираховується:

$$V_\Sigma = V_1 + V_2 + V_3 = 120000 + 34000 + 1000 = 155000 \text{ грн.}$$

Оскільки норма амортизаційних втрат для комп'ютерних систем $H_A = 5$, то для обрахування річного економічного ефекту слід брати до розгляду величину:

$$V_p = \frac{V_\Sigma}{H_A} = \frac{155000}{5} = 31000 \text{ грн}$$

Річний прибуток Π_p від провадження Сайту буде досягнуто за рахунок залученням нових клієнтів, реклами компанії, та покращенням позицій компанії у своїй сфері. Прогнозується збільшення на 2-3 проекти що півроку, і орієнтовно складатиме 60 000 грн на рік.

Коефіцієнт економічної ефективності розробки вираховується за:

$$K_{\text{ЕФ}} = \frac{\Pi_p}{V_p} = \frac{20000}{31000} = 0,64$$

Термін окупності розробки дорівнює:

$$T_{\text{ОК}} = \frac{1}{K_{\text{ЕФ}}} = \frac{1}{0,64} = 1,5$$

Таким чином, термін окупності Сайту буде 1 рік, 5 місяців.

РОЗДІЛ 2. ТЕХНІЧНЕ ЗАВДАННЯ НА РОЗРОБКУ САЙТУ

2.1 Призначення та мета створення сайту qubit-labs.com

2.1.1 Призначення

Призначенням цього Веб додатку є зручне управління проектами клієнтами та збільшення обсягу клієнтів. Вона поєднує у собі 2 типи систем:

- Одно-сторінковий додаток (Сайт не перезавантажується під час переходу по сторінками);
- Інформаційний ресурс.

2.1.2 Мета сайту

Метою Сайту є покращення управління проектами компанії. Клієнти зможуть авторизуватись на сайті, і отримати всю інформацію що до своїх діючих проектів.

2.1.3 Цільова аудиторія сайту

Цільова аудиторія Сайту представлена наступними групами користувачів:

- Міжнародні та вітчизняні ІТ компанії;
- Клієнти, Замовники;
- Потенційні Клієнти, Замовники;
- Потенційні працівники.

2.1.4 Основні завдання сайту

Завдання сайту містить наступні частини.

1) Інформаційну.

Сайт повинен надавати користувачам доступ до інформації:

- про організацію (історію створення, співробітників, партнерів, послуги, вакансії, зустрічі);
- про проекти організації (завершені та існуючі проекти);
- про новини організації (безпосередньо новини, прес-анонси прес-релізи, події, про організацію а також містити фото-та відео галереї);

2) Навчальну;

Сайт є унікальним у своїй сфері, розробці проєктів, та в роботі як аутстафінг;

3) SPA (Одно-сторінковий веб-додаток);

Веб додаток повинен дати доступ клієнту до перегляду своїх проєктів, управління ними, редагуванням та видаленням.

2.2 Загальні вимоги до сайту

2.2.1 Вимоги до стилістичному оформленню сайту

Стилістичне оформлення сайту має відповідати корпоративному стилю компанії використовувати його колірні схеми, графічні елементи (логотип) та шрифти.

2.2.2 Вимоги до графічного дизайну сайту

- Дизайн сайту повинен бути консервативним – використовувати неясраві, пастельні кольори і тони;
- Дизайн сайту має бути лаконічним, стильним та сучасним.

2.2.3 Вимоги до шрифтового оформлення сайту

- Основними шрифтовими гарнітурами є лише гарнітури з стандартних «безпечних» шрифтів.
- Шрифти, які використовуються для оформлення графічних елементів сайту. Не повинні суперечити корпоративному стилю компанії.
- Розмір шрифтів повинен забезпечувати зручність сприйняття тексту при мінімально допустимому розмірі екрану.

2.2.4 Браузери

Сайт повинен забезпечувати коректне відображення даних в наступних браузерах:

- Internet Explorer (версія 11.0 і вище);
- Opera (версія 10.0і вище);
- Safari (версія 4.0 і вище);
- Mozilla Firefox (версія 3.0 і вище);

- Google Chrome;
- Yandex.

2.2.5 Вимоги до безпеки даних

Сайт повинен забезпечувати коректне та безпечно зберігати данні такі як:

- Інформацію що до проектів;
- Інформацію що до клієнтів;
- Інформацію для доступу в персональний кабінет.

2.2.6 Вимоги до засобів перегляду сайту

Дизайн повинен правильно відображатися для будь-якого розширення монітору Пк.

2.2.7 Вимоги до контенту і наповненню сайту

Первинна розробка та верстка контенту (інформаційного вмісту) сайту повинна проводитися силами Виконавця за безпосередньої участі Замовника.

2.2.8 Інші вимоги до контенту

- Можливість відправити на e-mail будь-яку статтю із сайту або сторінку;
- Можливість рекомендувати сторінку у Facebook, можливість вподобати сторінку Facebook, можливість відмітити сторінку у Twitter.

2.2.9 Вимоги до збереження інформації

У системі управління сайтом має бути передбачений механізм резервного копіювання структури, файлів та вмісту бази даних.

2.2.10 Вимоги до кількості відвідувачів

Веб-ресурс має бути максимально оптимізованим, щоб багато унікальних відвідувачів могли одночасно користуватися ресурсом.

2.3 Вимоги до шаблону сайту

2.3.1 Елементи навігації

Інтерфейс сайту повинен забезпечувати наочне, інтуїтивно зрозуміле представлення структури розміщеної на ньому інформації, швидкий і логічний перехід до розділів і сторінок.

2.3.2 Сторінка «home»

Головна сторінка призначення для ознайомлення потенційного клієнта з компанією. На най коротко описано чим займається компанія, фото-галерея офісу, новини компанії та контактна інформація.

2.3.3 Сторінка «about»

Сторінка призначена для історії розділів про створення організації, опису міжнародної мережі організацій, представлення співробітників офісу в Києві, переліку партнерських організацій, конкурсів тощо.

2.3.4 Сторінка «project»

Сторінка призначена для виведення інформації про проекти (якщо клієнт авторизований). Якщо авторизація не відбулась на сторінці буде напис авторизуйтесь для перегляду проектів.

2.3.5 Сторінка «blog»

Сторінка призначена для розміщення новин, фото-та відео галереї, висвітлення новин про Qubit Labs в ЗМІ розміщення прес-анонсів та прес-релізів, опису подій.

Кожний запис має містити таку інформацію:

1. Автора;
2. заголовок матеріалу;
3. графічне зображення, медіа файли.

2.3.6 Сторінка «contact us»

Сторінка призначена для зв'язку між компанією та клієнтом, на ній міститься форма Зворотного зв'язку.

2.3.7 Блок «авторизації»

Блок призначений для авторизації клієнта на сайті. Після авторизації клієнта на сайті, на сторінку Projects автоматично завантажується інформацію що до його проектів.

2.3.8 Блок «реєстрації»

Блок призначений для реєстрації клієнта на сайті. Після реєстрації клієнта на сайті, на сторінку Projects автоматично завантажується інформацію що до його проектів.

2.3.9 Форма «зворотній зв'язок»

На сторінці «Зворотній зв'язок» користувач повинен мати можливість написати листа організації.

Форма повинна містити такі поля:

- Ваше Ім'я (обов'язково для заповнення);
- Ваша Фамілія (не обов'язково);
- Ваша електронна адреса (обов'язково для заповнення);
- Ваш телефон (не обов'язково);
- Текст листа Примітка-максимум 500 символів.

2.4 Функції системи

Основними функціями системи є:

- Швидкий зв'язок з адміністратором;
- Реєстрація клієнта;
- Авторизація клієнта;
- Облік проектів;
- Редагування та видалення інформації проектів;
- Захист від випадкового видалення проекту;
- Облік новин компанії;
- Групування новин по категоріям.

2.4.1 Вимоги до функцій системи

Таблиця 2.1 Вимоги до розроблюваних функцій

Функція	Вхідна Інформація	Вихідна інформація
Швидкий зв'язок з адміністратором	Таблиця: <ul style="list-style-type: none"> • Message 	Сторінка формування запиту клієнта

	<ul style="list-style-type: none"> • Clients • Register 	
Реєстрація клієнта	Таблиця: <ul style="list-style-type: none"> • Register 	Сторінка реєстрації користувача
Авторизація клієнта	Таблиця: <ul style="list-style-type: none"> • Client 	Сторінка авторизації клієнта
Облік проектів	Таблиця: <ul style="list-style-type: none"> • Projects • Clients 	Сторінка з переліком проектів авторизованого клієнта
Редагування та видалення інформації проектів	Таблиця: <ul style="list-style-type: none"> • Projects 	Вікно з інформаційними полями для редагування проекту та кнопками видалення, оновлення
Захист від випадкового видалення проекту	Таблиця: <ul style="list-style-type: none"> • Projects 	Вікно захисту з підтвердженням видалення проекту
Облік новин компанії	Таблиця: <ul style="list-style-type: none"> • News • Manager 	Сторінка з переліком новин сайту
Групування новин по категоріям	Таблиця: <ul style="list-style-type: none"> • News • Category 	Сторінка з новинами обраної категорії

2.5 Вимоги до технічного забезпечення сайту

Таблиця 2.2 Вимоги до серверу

Вимоги до серверу
Intel Xeon E3-1230 / 2x 3.3 ГГц / 6 ГБ ОЗУ / 1x 500 ГБ

Вимоги для сайту “Qubit Labs”:

- Повинен в автоматичному режимі оновлювати інформацію у персональному кабінеті;
- Повинен мати можливість на доопрацювання;
- Повинен витримувати до 10 тисяч користувачів за добу;
- Має бути надійним, та захищати інформацію з бази даних ;
- Має бути простим в використанні.

Таблиця 2.3 Вимоги до Пк користувача

Вимоги для персонального комп'ютера клієнта
Процесор Intel pentium 3 1GHz
Не менше 1ram оперативної пам'яті;
256 Kbit/s мінімальна пропускна здатність

Вимоги надійності сайту:

- Сайт повинен функціонувати цілодобово;
 - Повинен бути стійким до навантажень користувачів - до 10 000 тисяч за добу;
 - Повинен мати можливість на створення резервних копій;
- Для користування Сайтом не потрібно мати спеціальних навичок.

2.6 Склад і зміст робіт

Стадії створення сайту і терміни наведені в таблиці 2.4.

Таблиця .2.4 Назви робіт при створенні сайту

№ п/п	Назва роботи	Строки виконання робіт
1	Перед проектне дослідження підприємства	01.04.2021
2	Технічне завдання	15.05.2021
3	Технічний проект	20.05.2021
4	Оформлення документації	28.06.2021

2.7 Вимоги до складу та змісту робіт по введенню сайту в експлуатацію

Для створення умов функціонування, при яких гарантується відповідність створюваного сайту вимогам цього ТЗ. На хостинг переноситься (сайт), зверстаний шаблон дизайну і структура і наповнення бази даних. Перенесення даних здійснюється протягом семи робочих днів (обговорюються терміни з Замовником).

- **Технічна підтримка веб-ресурсу**

Технічна підтримка це - комплекс послуг, пов'язаних з підтримкою безперебійної роботи веб-ресурсу. Забезпечення працездатності сайту та швидким усуненням причин збоїв роботи ресурсу. З моменту передачі сайту технічна підтримка необхідна протягом одного року.

РОЗДІЛ 3. РОЗРОБКА КОМПЛЕКСУ ЗАДАЧ АВТОМАТИЗАЦІЇ

3.1 Обґрунтування вибору використаних програмних засобів

Під час розробки використовувались наступні засоби:

- React (Javascript фреймворк);
- Html/Css;
- Node js (сервер для зв'язку між базою та сайтом) [20];
- Visual studio Code (2021);
- MySQLWorkbench;

React - це декларативна, ефективна і гнучка JavaScript бібліотека для створення користувацьких інтерфейсів. Вона дозволяє вам збирати складний UI з маленьких ізольованих шматочків коду, званих «компонентами». React - JavaScript-бібліотека з відкритим вихідним кодом для розробки призначених для користувача інтерфейсів[Розділ 1].

HTML - мова тегів, за допомогою якої здійснюється створення веб-сторінок для мережі Інтернет. HTML лише описує структуру веб-сторінки семантично. Елементи HTML можна розглянути як будівельні блоки сторінок HTML [Розділ 1].

CSS (Cascading Style Sheets) - це код, який ви використовуєте для стилізації вашої веб-сторінки. Основи CSS допоможуть вам зрозуміти, що вам потрібно для початку роботи[Розділ 1].

Node.js - платформа з відкритим кодом для створення власного серверу та налаштування його

Node.js характеризується наступним чином[11]:

- асинхронна одно-нитева модель виконання запитів;
- неблокуючий ввід/вивід;
- система модулів CommonJS;

- рушій JavaScript Google V8;

Для керування модулями ми можемо використовуватися пакетний менеджер npm або yarn.

Visual Studio Code – Зручне середовище розробки на платформі Windows, Mac OS, Linux, розроблений компанією Microsoft.[12]

MySQL Workbench - це уніфікований візуальний інструмент для архітекторів баз даних, розробників та адміністраторів баз даних. MySQL Workbench забезпечує моделювання даних, розробку SQL та комплексні інструменти адміністрування для конфігурації сервера, адміністрування[13].

3.2 Логічна модель БД

Першим етапом створення БД було створення логічної та фізичної моделі. (Додаток Б) Для створення логічної моделі ми використали програму Erwin. (рис 3.2.1)

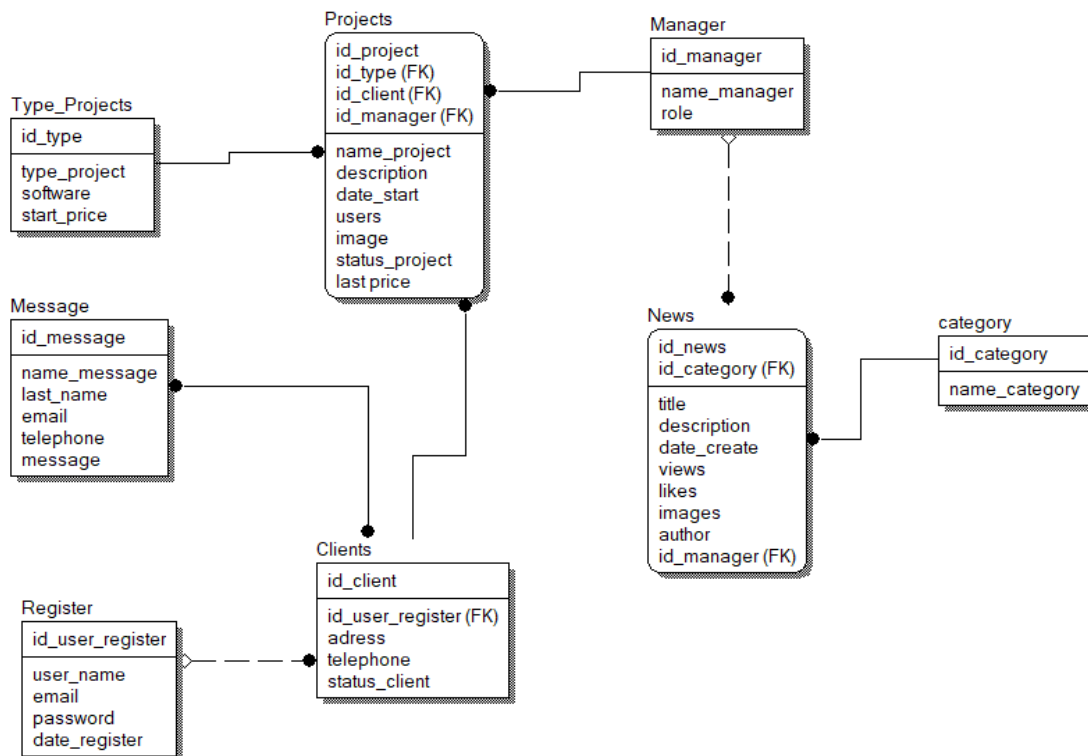


Рис 3.2.1 Логічна модель БД

На основі фізичної моделі даних була сгенерована БД за допомогою інструменту MySQL Workbench, і за допомогою його та серверу на Node ми зв'яжемо данні з нашим сайтом(рис 3.2.2).

Name	Engine	Version	Row Format
Category	InnoDB	10	Compact
Clients	InnoDB	10	Compact
Manager	InnoDB	10	Compact
Message	InnoDB	10	Compact
News	InnoDB	10	Compact
Projects	InnoDB	10	Compact
Register	InnoDB	10	Compact
Type_projects	InnoDB	10	Compact

Рис 3.2.2 Сгенерована БД “Qubit Labs”

Сгенерована БД містить 8 таблиць:

- Category – Таблиця категорій новин сайту;
- Clients - Таблиця клієнтів компанії;
- Manager – Таблиця менеджерів компанії;
- Message -Таблиця швидкого зв'язку користувачів з адміністратором;
- News – Таблиця переліку новин компанії;
- Projects – Таблиця проектів компанії;
- Register – Таблиця тільки зареєстрованих користувачів;
- Type_projects – Таблиця типів проектів.

Таблиця 3.2 «Поля БД Qubit Labs»

Назва таблиці	Список полів	Тип поля
Category	id_category	integer
	name_category	varchar(50)
Clients	id client	integer
	id user register	integer
	adress	varchar(100)
	telephone	integer
	status client	varchar(50)
Message	id message	integer

Назва таблиці	Список полів	Тип поля
	name message	varchar(45)
	last name	varchar(45)
	email	varchar(150)
	telephone	integer
	message	varchar(500)
Manager	id manager	integer
	name manager	varchar(45)
	role	varchar(45)
News	id news	integer
	id category	integer
	title	varchar(50)
	description	varchar(200)
	date create	date
	views	integer
	likes	integer
	images	varchar(150)
	author	varchar(45)
	id manager	integer
Projects	id project	integer
	id type	integer
	id client	integer
	id manager	integer
	name project	varchar(45)
	description	varchar(200)
	date start	date
	image	varchar(150)
	users	integer
	status project	varchar(50)
	last price	integer
Register	id user register	integer
	email	varchar(150)
	user name	varchar(45)
	password	varchar(45)
	date register	date
Type_projects	id type	integer
	type project	varchar(45)

Назва таблиці	Список полів	Тип поля
	start price	integer
	software	varchar(50)

3.3 Створення інтерфейсу

При розробці був застосований компонентний підхід розробки, це буде показано у середовищі розробки Visual Studio Code (Додаток В).

За такою же технологією було розроблено інші сторінки сайту.

Першим етапом була розробка головної сторінки яка умовно поділена на 7 розділів, вони показані на рис.3.3.1 – рис 3.1.7.

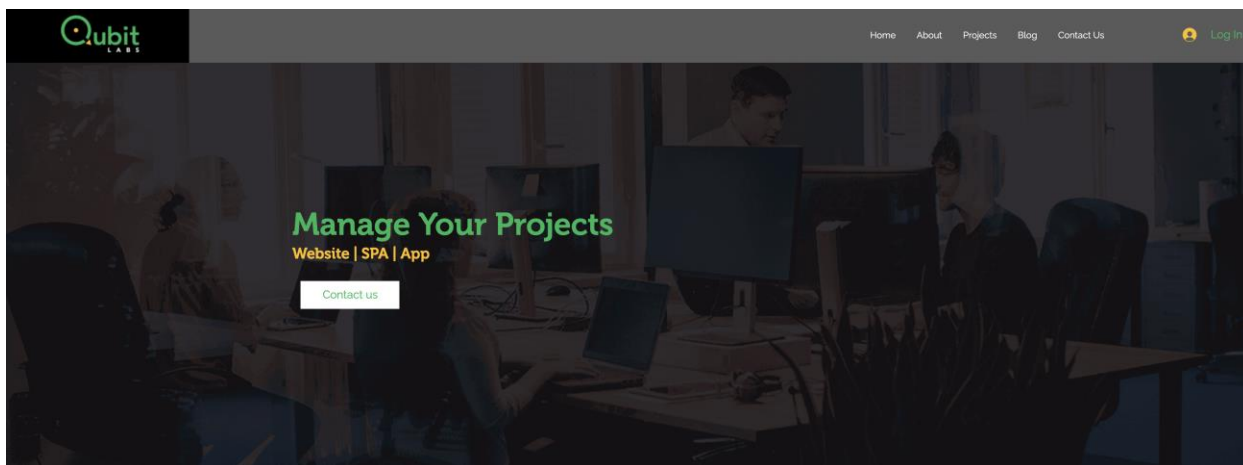


Рис 3.3.1 Головна сторінка Розділ 1 сайту Qubit Labs

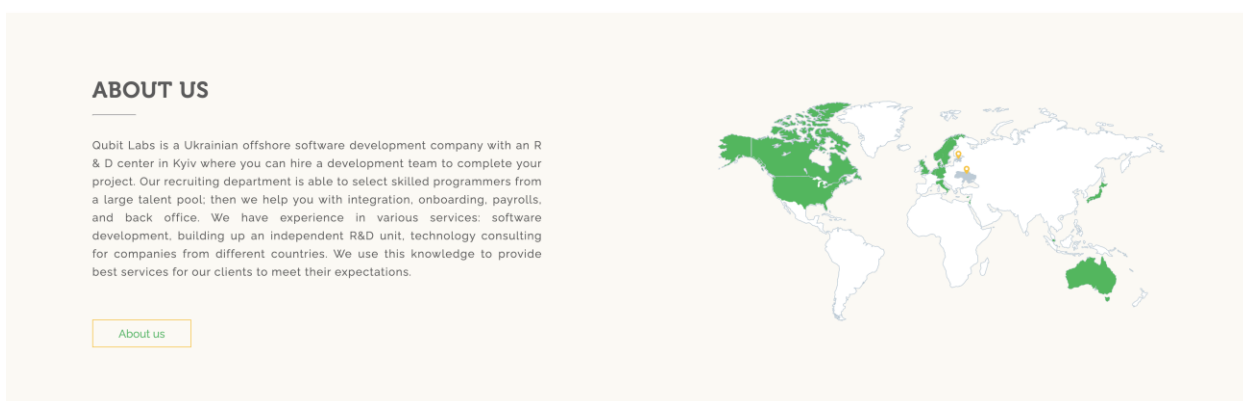


Рис 3.3.2 Головна сторінка Розділ 2 сайту Qubit Labs

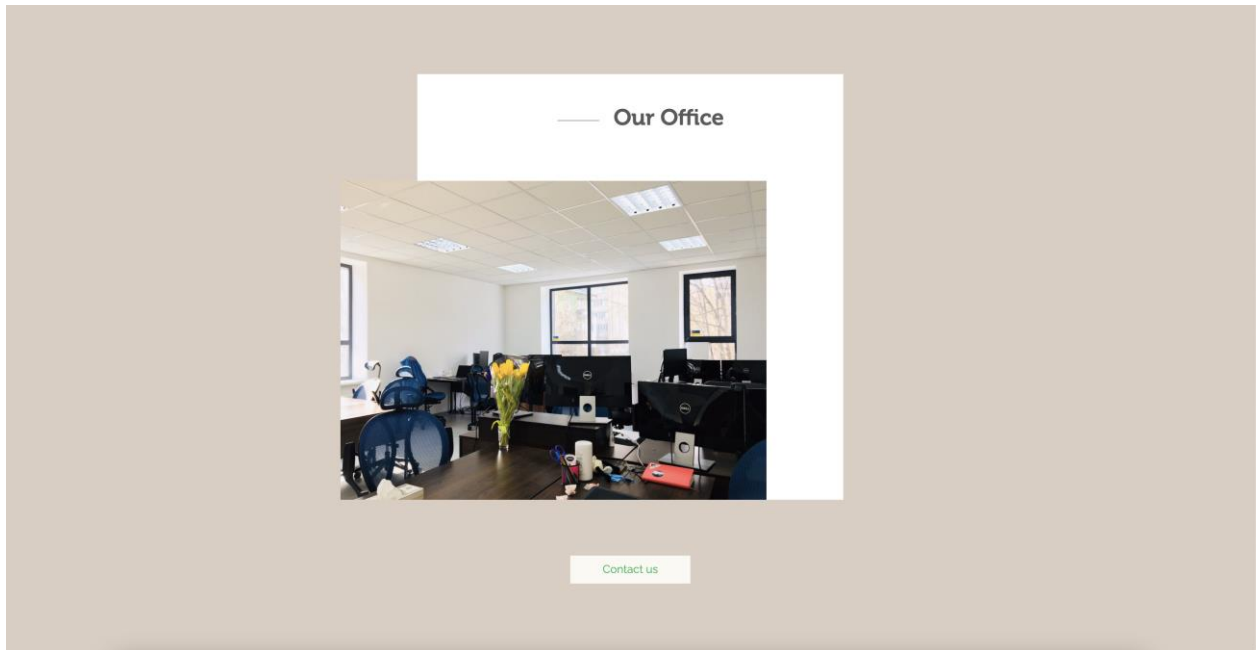


Рис 3.3.3 Головна сторінка Розділ 3 сайту Qubit Labs

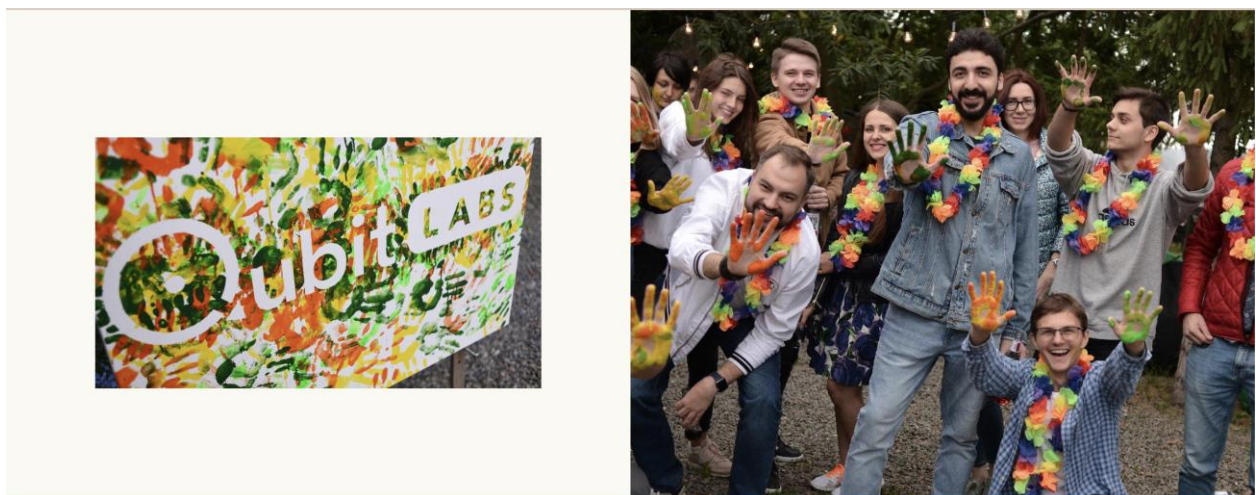


Рис 3.3.4 Головна сторінка Розділ 4 сайту Qubit Labs

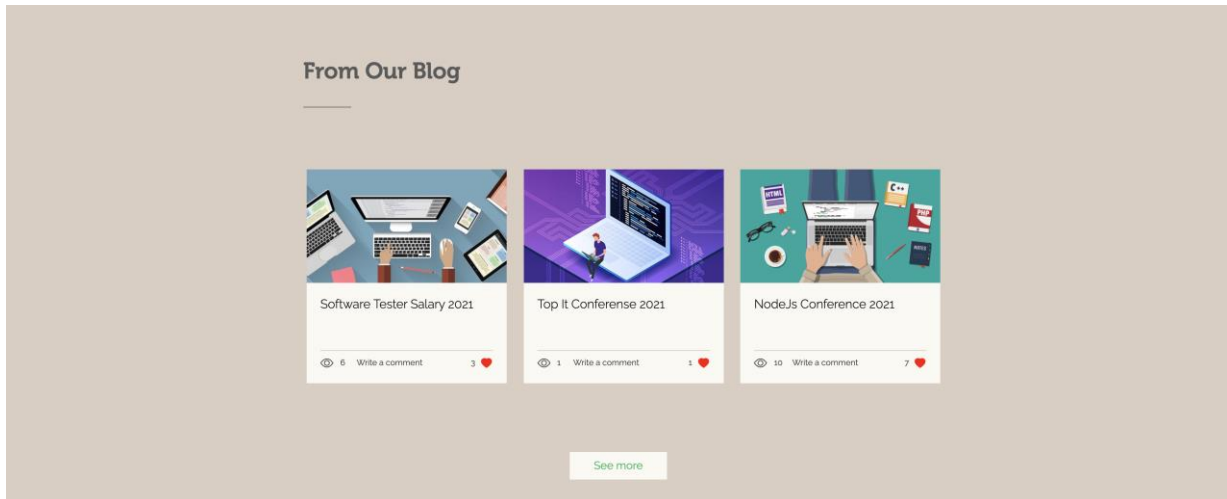


Рис 3.3.5 Головна сторінка Розділ 5 сайту Qubit Labs



Рис 3.3.6 Головна сторінка Розділ 6 сайту Qubit Labs

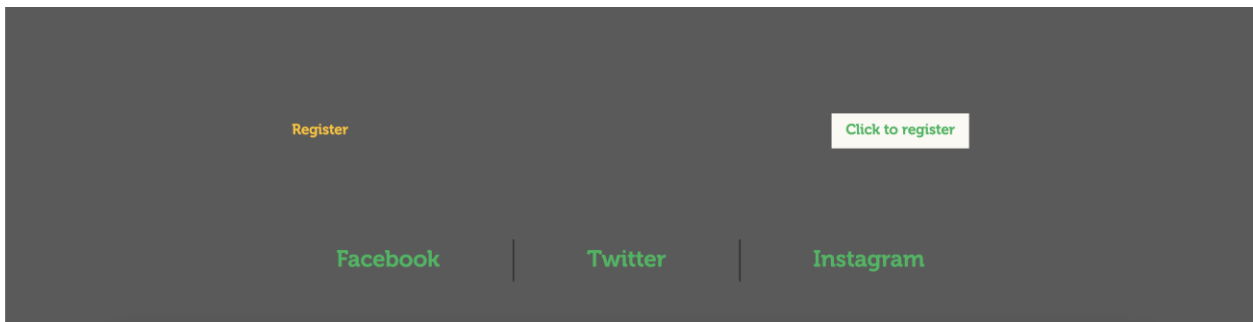


Рис 3.3.7 Головна сторінка Розділ 7 сайту Qubit Labs

Register

Yevhenii [Click to register](#)

billoys1337@gmail.com

●●●●●●●●

Рис 3.3.8 Головна сторінка Розділ 8 сайту Qubit Labs

```
app.post("/login", (req, res) => {
  const username = req.body.username;
  const password = req.body.password;

  db.query(
    "SELECT * FROM Clients WHERE email = ? AND password = ?",
    [username, password],
    (err, result) => {
      if (err) {
        res.send({ err: err });
      }

      if (result.length > 0) {
        res.send(result);
      } else {
        res.send({ message: "wrong password or username" });
      }
    }
  );
});
```

Рис 3.3.9 Код реалізації авторизації

```

const getAuth = () => {
  if (loginUser) {
    return "";
  } else {
    return (
      <div className="auth">
        <div className="auth__form">
          <h2>Login Please</h2>
          <input
            type="text"
            placeholder="Email"
            onChange={e => {
              setUsername(e.target.value);
            }}
          />
          <input
            type="password"
            placeholder="Password"
            onChange={e => {
              setPassword(e.target.value);
            }}
          />
          <button className="login" onClick={login}>
            Login
          </button>
        </div>
      </div>
    );
  }
};

```

Рис 3.3.10 Код реалізації авторизації

```

app.post("/register", (req, res) => {
  const name = req.body.name;
  const email = req.body.email;
  const password = req.body.password;

  db.query(
    "INSERT INTO users (name,email,password) VALUES (?,?/?)",
    [name,email, password],
    (err, result) => {
      if (err) {
        console.log(err);
      } else {
        res.send("Values Inserted");
      }
    }
  );
});

```

Рис 3.3.11 Код відправки даних з серверу до БД

```

app.post("/update", (req, res) => {
  const idPr = req.body.idPr;
  const idUs = req.body.idUs;
  const name = req.body.name;
  const type = req.body.type;
  const ND = req.body.ND;
  const price = req.body.price;
  const desc = req.body.desc;

  db.query(
    "UPDATE `Qubit_Labs`.`Projects` SET `name` = ?, `type` = ?, `description` = ?, `number_developer` = ?, `id_client` = ?, `price` = ? WHERE (`id` = ?)",
    [name, type, desc, ND, idUs, price, idPr],
    (err, result) => {
      if (err) {
        res.send({ err: err });
      }
    }
  );
});

```

Рис 3.3.12 Код запиту проектів клієнта

3.4 Інструкція користувача

Перехід здійснюється за допомогою навігаційною панелі яка показана на рисунку 3.3.8, на ній знаходяться такі кнопки як:

- Home (Головна сторінка)
- About (Інформація про компанію)
- Projects (Сторінка проектів, які показані після авторизації)
- Blog (Сторінка новин компанії)
- Contact us (Сторінка з формою зворотного зв'язку)
- Log In (Відправляє на авторизацію)

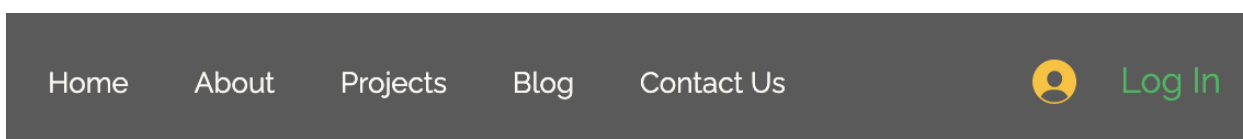


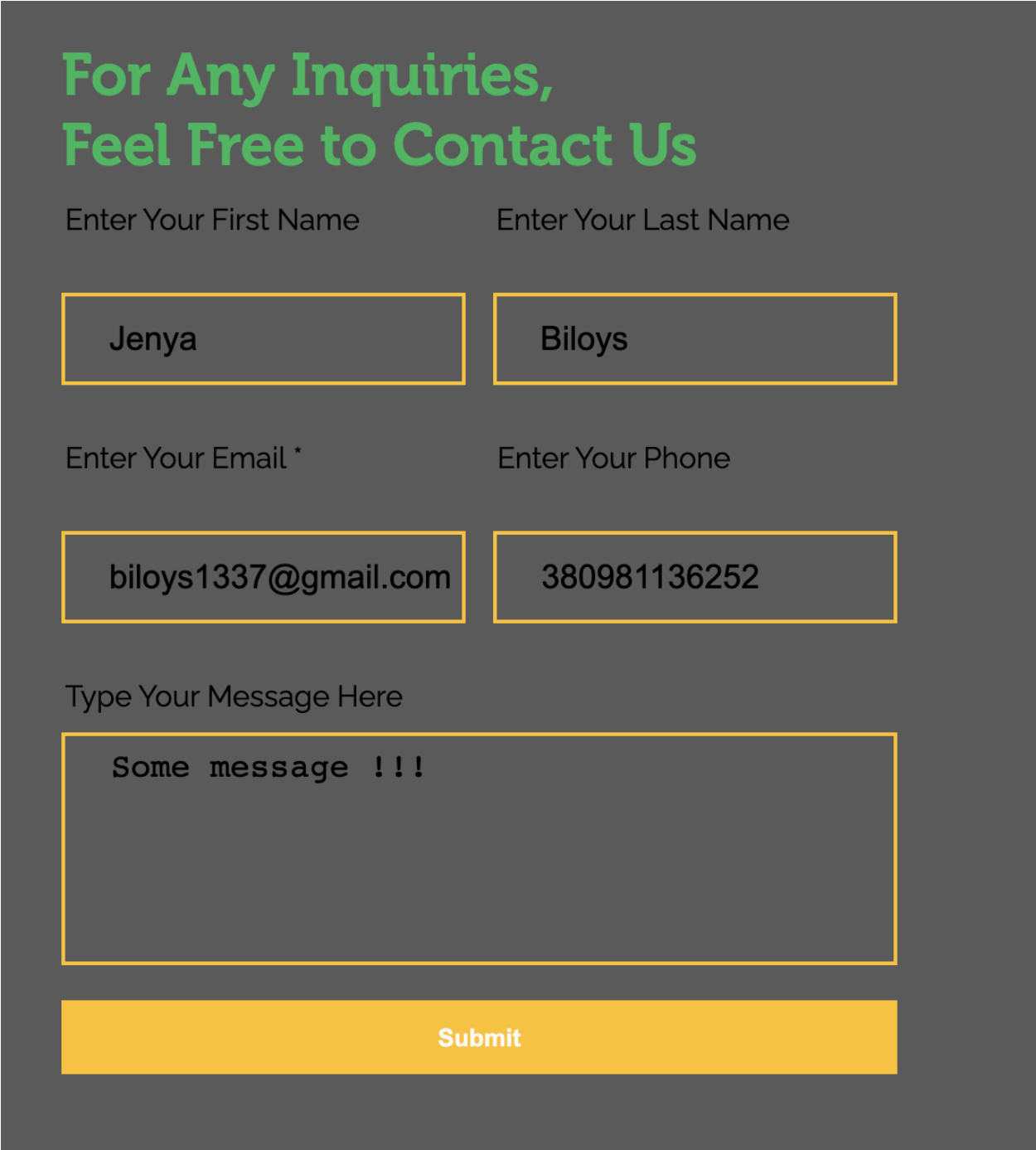
Рис 3.4.1 Навігаційна панель сайту

Першою розробленою функцією була «Швидкий зв'язок користувача з адміністратором». Це одна з найперших функцій сайту якою може скористатись не зареєстрований користувач для того що б зв'язатись з адміністратором компанії. Сторінка з формою зворотного зв'язку, знаходиться на сторінці Contact us. Поля для заповнення:

- Ваше Ім'я (обов'язково для заповнення):
- Ваша Фамілія (не обов'язково)

- Ваша електронна адреса (обов'язково для заповнення):
- Ваш телефон (не обов'язково)
- Текст листа Примітка-максимум 500 символів;

Після заповнення форми натискаємо Submit (Відправити) (рис 3.4.2)



The image shows a contact form on a dark grey background. At the top, there is a green heading: "For Any Inquiries, Feel Free to Contact Us". Below the heading are four input fields arranged in a 2x2 grid. The first row contains "Enter Your First Name" with the value "Jenya" and "Enter Your Last Name" with the value "Biloy". The second row contains "Enter Your Email *" with the value "biloy1337@gmail.com" and "Enter Your Phone" with the value "380981136252". Below these fields is a larger text area labeled "Type Your Message Here" containing the text "Some message !!!". At the bottom of the form is a yellow "Submit" button.

Рис 3.4.2 Заповнення форми зворотного зв'язку

Після відправки перевіriamo БД, на перелік відправлених даних.(рис 3.4.3)

	id	name	last_name	email	telephone	message
▶	1	Jenya	Biloys	billoys1337@gmail.com	380981136252	Some message !!!
	NULL	NULL	NULL	NULL	NULL	NULL

рис 3.4.3 Таблиця Message після відправлення запиту

Після цього було розроблено функцію реєстрації користувача. (рис 3.4.4)

Рис 3.4.4 Форма реєстрації користувача

Після заповнення даних натискаємо на кнопку “Click to register”, після чого в нас з'явиться свій персональний кабінет. Сервер отримує данні , і відправляє їх до БД. Перевіримо роботу функції, заповнюємо поля та натискаємо «Click to register». Заходимо до БД та перевіряємо нашу таблицю Register (рис 3.4.5)

Result Grid		Filter Rows:	Search	Edit:	Export
	id_user_register	user_name	email	password	date_register
▶	1	Yevhenii	billoys1337@gmail.com	12345sss123	2010-05-20 21:00:00
	NULL	NULL	NULL	NULL	NULL

Рис 3.4.5 Таблиця Register

Наступною функцією була розроблена **Авторизація**. Вона доступна на сторінці Projects (рис 3.4.6)

Please log in to your personal account to display your projects

Login Please

Рис 3.4.6 Функція Авторизації

Після вводу даних для авторизації, таких як «email» та «password». В нас відкриється список наших проєктів. Показано на (рис 3.4.6 – 3.4.7).

Login Please

Рис 3.4.7 Введення даних для авторизації

Welcome back Jenya

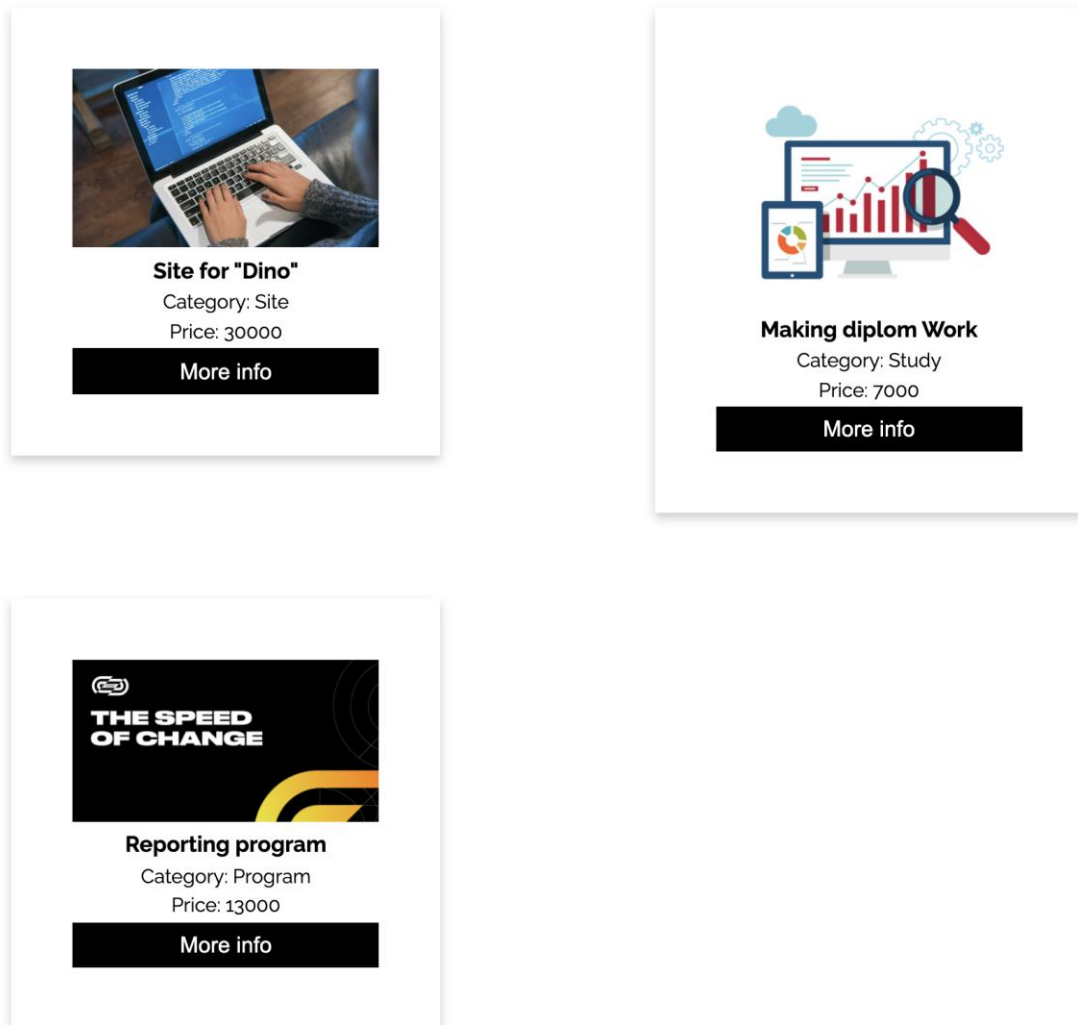
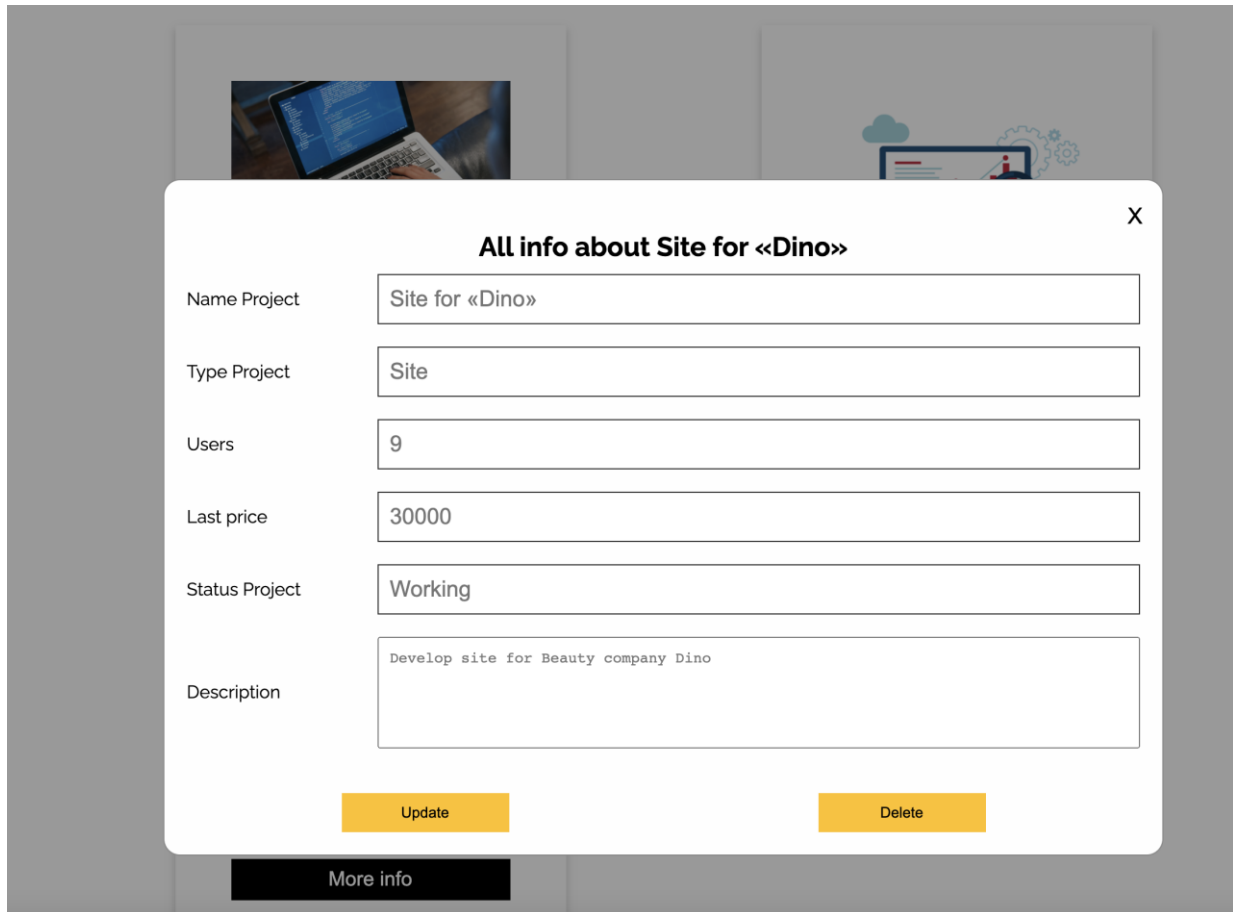


Рис 3.4.8 Облік проектів клієнта

Якщо натиснути на кнопку More info в нас відкриється форма з повною інформацією про проект. На цій формі ми можемо змінити інформацію про проект, або взагалі видалити його. Після натиснення на видалення проекту в нас відкриється вікно підтвердження видалення, це зроблено для того щоб випадково не видалити проект. Показано на (рис 3.4.9 – 3.4.14)



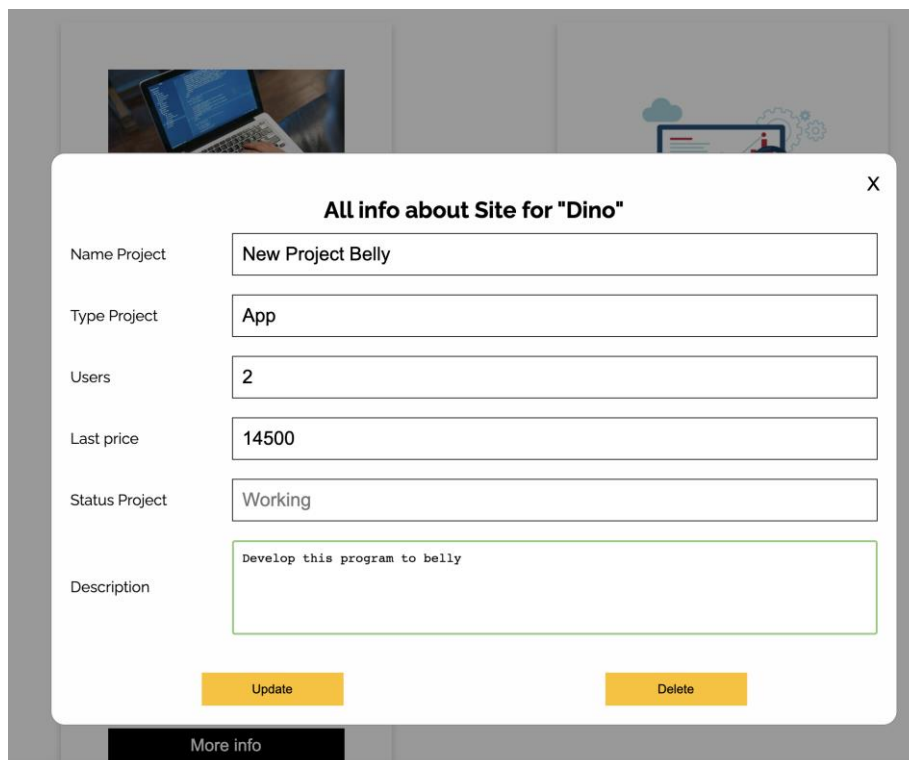
The screenshot shows a modal window titled "All info about Site for «Dino»". It contains several input fields for project details. The fields are: Name Project (Site for «Dino»), Type Project (Site), Users (9), Last price (30000), Status Project (Working), and Description (Develop site for Beauty company Dino). At the bottom of the modal are two yellow buttons: "Update" and "Delete". Below the modal, a "More info" button is visible on the background page.

Name Project	Site for «Dino»
Type Project	Site
Users	9
Last price	30000
Status Project	Working
Description	Develop site for Beauty company Dino

Update Delete

More info

Рис 3.4.9 Форма повної інформації щодо проекту



The screenshot shows a modal window titled "All info about Site for \"Dino\"". It contains several input fields for project details. The fields are: Name Project (New Project Belly), Type Project (App), Users (2), Last price (14500), Status Project (Working), and Description (Develop this program to belly). At the bottom of the modal are two yellow buttons: "Update" and "Delete". Below the modal, a "More info" button is visible on the background page.

Name Project	New Project Belly
Type Project	App
Users	2
Last price	14500
Status Project	Working
Description	Develop this program to belly

Update Delete

More info

Рис 3.4.10 Редагування інформації

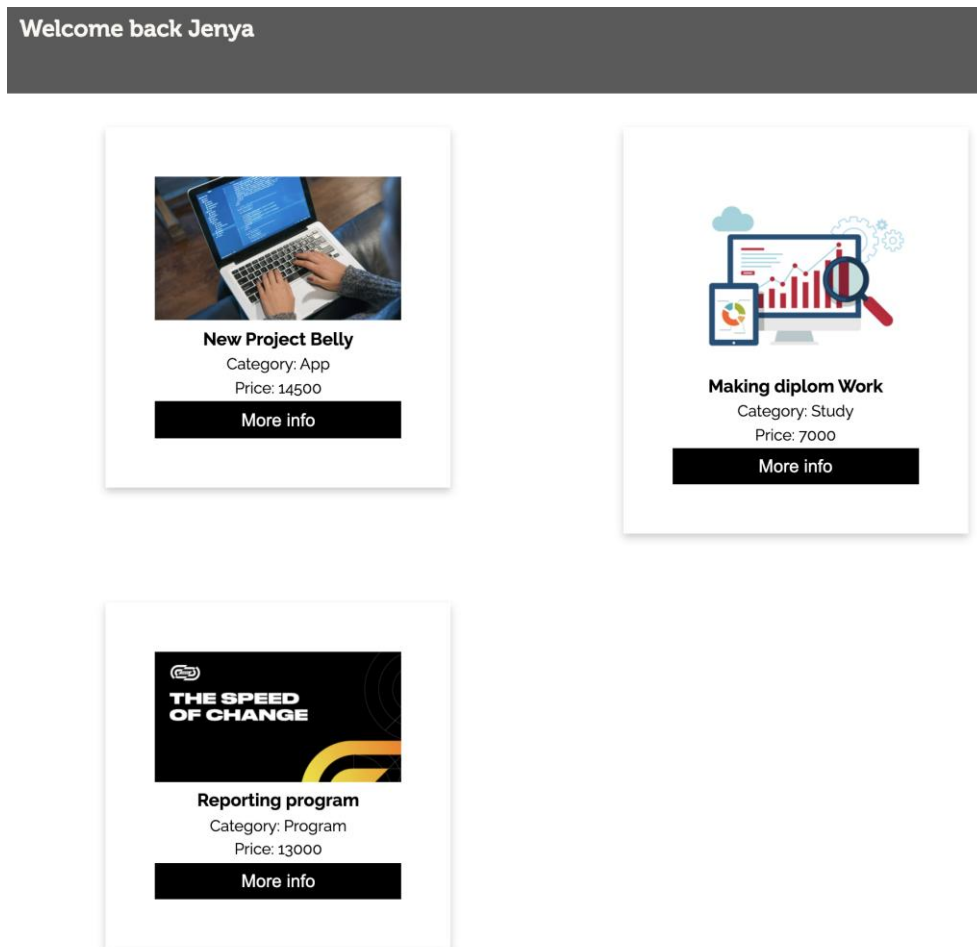


Рис 3.4.11 Результат оновлення інформації

X

All info about New Project Belly

Name Project	<input type="text" value="New Project Belly"/>
Type Project	<input type="text" value="App"/>
Users	<input type="text" value="2"/> <div style="border: 1px solid gray; border-radius: 10px; padding: 2px; display: inline-block; margin-left: 10px;"> not delete Yes, delete </div>
Last price	<input type="text" value="14500"/>
Status Project	<input type="text" value="Working"/>
Description	<input style="height: 40px;" type="text" value="develop this program to belly"/>

Update
Delete

Рис 3.4.12 Видалення проекту

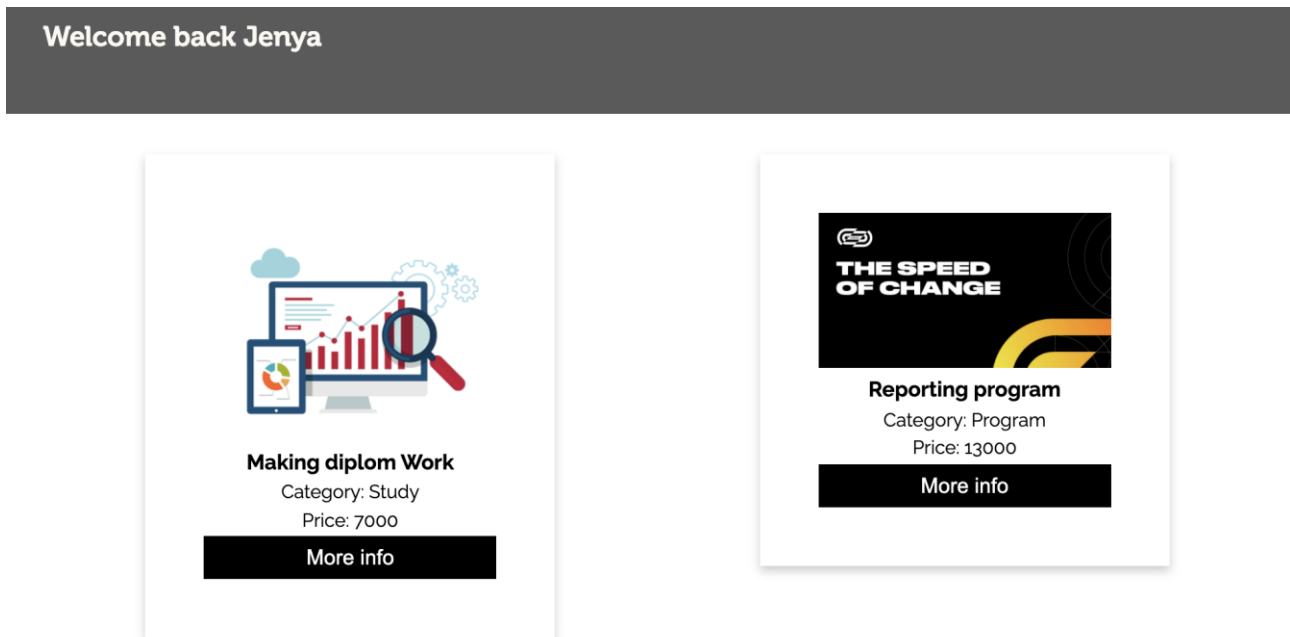


Рис 3.4.13 Результат видалення проекту

Для додавання проекту скористаємось кнопкою додавання проектів.(рис 3.4.14)

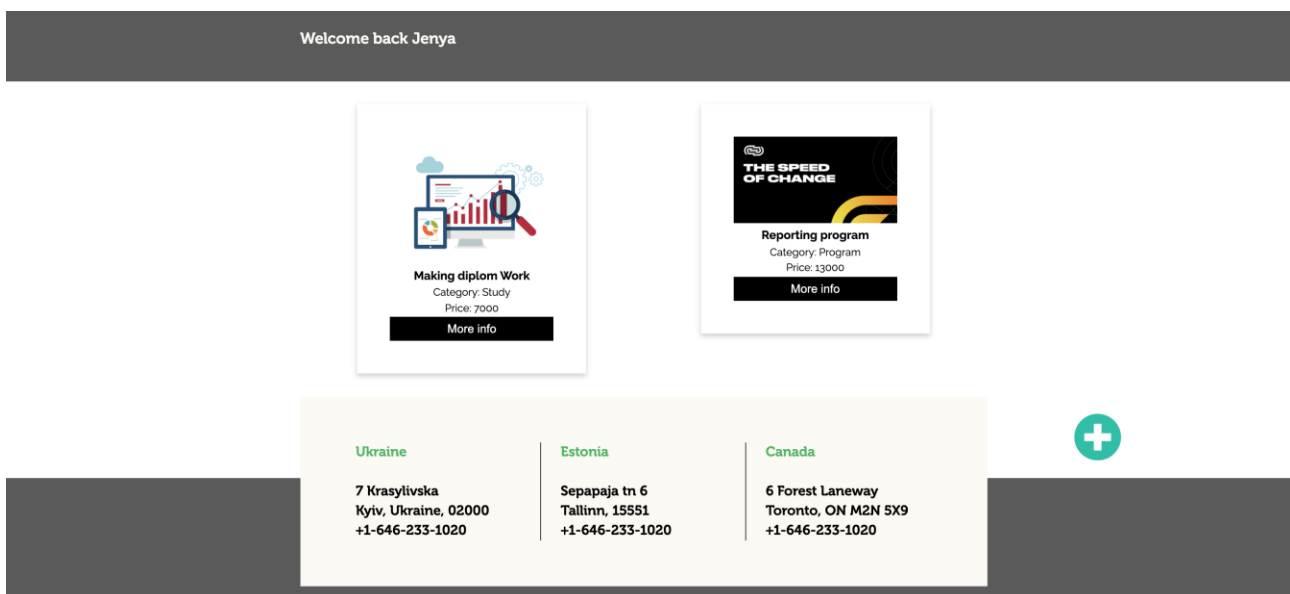


Рис 3.4.14 Кнопка додавання нового проекту

Після натиснення на кнопку додавання проекту в нас відкриється вікно з інформаційними полями для введення інформації що до нового проекту рис(3.4.15)

X

Add new project


Name Project	Project for Qubit
Type Project	System
Users	4
Last price	82000
Image project url	https://upload.wikimedia.org/wikipedia/commons/3/3c/IMG_logo.svg
Description	New project

Add


Рис 3.4.15 Додавання нового проекту

Після натиснення на кнопку “Add” оновлюємо сторінку та перевіряємо список наших проектів. (рис 3.4.16)


Welcome back Jenya



Making diplom Work
Category: Study
Price: 7000
[More info](#)



Reporting program
Category: Program
Price: 13000
[More info](#)



Project for Qubit
Category: System
Price: 82000
[More info](#)

Рис 3.4.16 Результат додавання проекту

Переглянути новини компанії можна на сторінці Блог. Рис(3.4.17)

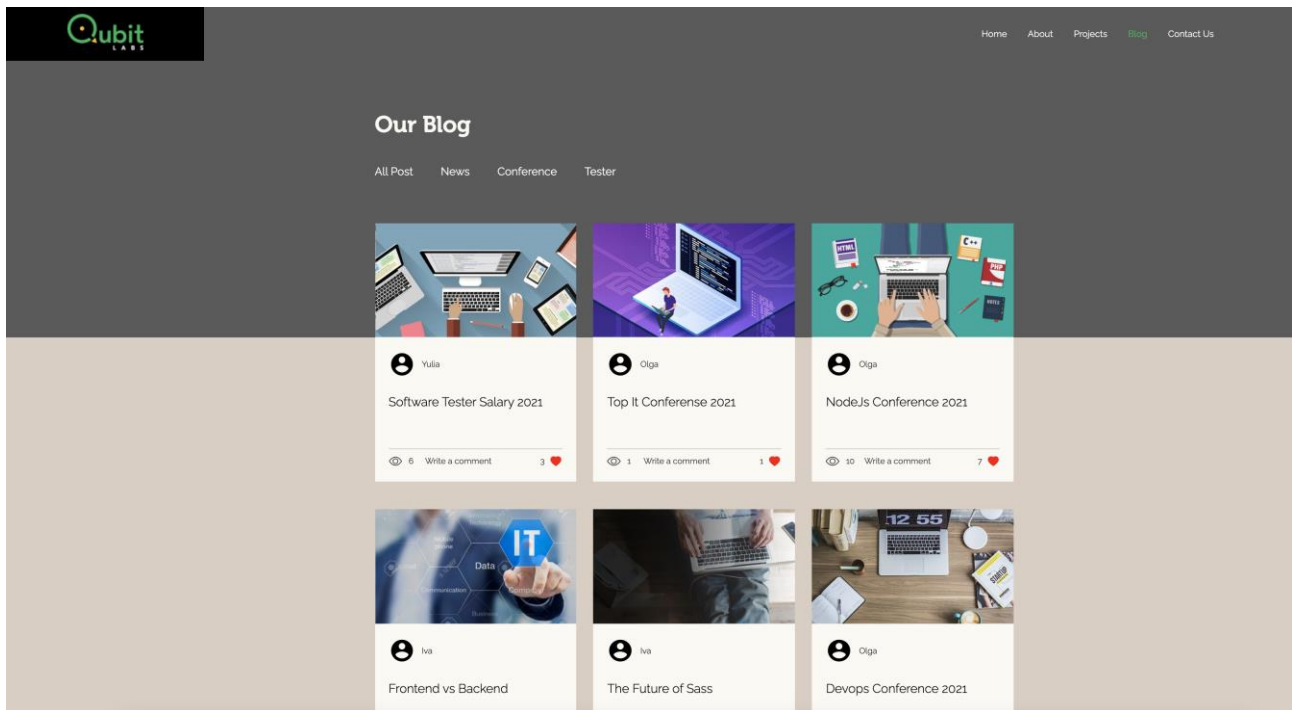


Рис 3.4.17 Результат додавання проекту

На сторінці також є можливість згрупувати новини по категоріям. (рис 3.4.18)

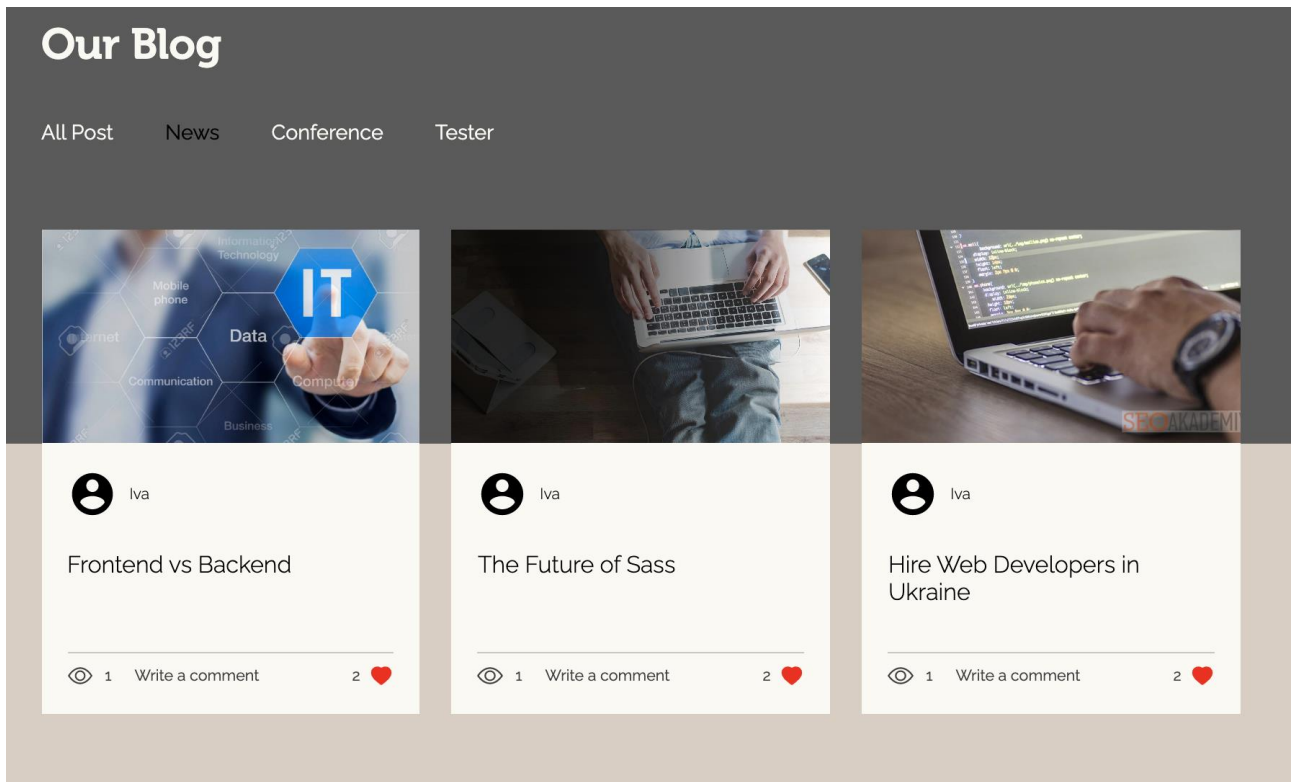


Рис 3.4.18 Групування новин по категорії News

РОЗДІЛ 4. ОХОРОНА ПРАЦІ

4.1 Техніка безпеки підприємства

Сучасний офіс - це не тільки місце роботи, а й джерело потенційної небезпеки для здоров'я і життя співробітників. Адже за місцем працевлаштування останні проводять більшу частину свого життя. Та й експлуатація електроприладів і техніки не додає оптимізму.

На підприємстві вживаюся заходи до того, щоб праця працюючих була безпечною. На підприємстві встановлені проти пожежні системи. На підприємстві є відповідальна людина, яка час від часу перевіряє стан проти пожежних систем, та вогнегасників, які знаходяться в кожній кімнаті офісу. [17]

Ключове питання безпеки в офісі - використання комп'ютерів і оргтехніки. Такі пристрої викликають чимало нарікань з боку користувачів. Найчастіше мова йде про швидкої стомлюваності, зниженні зору, появі психологічної пригніченості і навіть алергічних реакціях. Однак це не привід звільнитися або відмовлятися від виконання трудових зобов'язань. Набагато простіше дотримуватися встановлених законодавством норм безпечної роботи.

Кількість працівників з комп'ютерами в одному приміщенні певного метражу чітко регламентовано. Норма площі на людину з плоским монітором становить 4,5 метра квадратних. Для тих офісів, де співробітникам доводиться працювати з моніторами старого зразка, така норма вище. Вона становить 6 квадратних метрів на працівника. Виходячи з цього визначають кількість персоналу в одному приміщенні. Не варто забувати і про перегородках для офісу.

Медики наполягають, що за комп'ютером більше шести годин поспіль проводити шкідливо. Рекомендується робити перерви щогодини. Так само як кожні дві години провітрювати кімнату. У місцях скупчення електроприладів краще проводити вологе прибирання щодня.

В обов'язковому порядку потрібно подбати про правильне розміщення комп'ютера і інших потрібних речей на робочому столі. Кожен співробітник повинен мати зручний стілець, який буде знімати напругу з м'язів спини і шиї.

Не менше значення мають показники температури в робочих приміщеннях. Це близько 22-24 градусів за Цельсієм взимку і до 25 градусів - в жаркий період. Коригувати їх можна за допомогою систем кондиціонування.

Важливий фактор безпеки в офісі - форма одягу. Статистикою встановлено, що в компаніях, де співробітники зобов'язані носити уніформу, травмоопасність значно нижче. Так само як і кількість виписаних лікарняних листів. Наприклад, носіння жінкою взуття на високих підборах рано чи пізно може стати причиною захворювань опорно-рухового апарату або ніг. Також каблук може застрягти в решітці килимка або на вході в ліфт, що може бути причиною травм.

Особливу увагу варто приділити рівню шуму в приміщенні офісу. Він становить 50-65 дБ. В іншому випадку може знизитися продуктивність праці.

Охорона праці - це система забезпечення безпеки роботи, збереження життя і здоров'я працюючих, в процесі усієї їхньої трудової діяльності.

Охорона праці на сьогодні є однією з головних систем на підприємствах різного напрямку будь-яких форм власності. Дотримуючись елементарних правил безпечного виконання робіт і вимоги охорони праці, роботодавець може домогтися істотного зниження виробничого травматизму, практично до 95%. Відсоток, ймовірності складе людський фактор, від якого складно піти. Саме таке процентне співвідношення випадків травматизму виникають через недотримання вимог охорони праці, як з боку роботодавця, так і з боку працівника.

Охорона праці як система зачіпає практично кожную сторону життя людей. Вона переплітається з великою кількістю технічних і гуманітарних наук, виконуючи одне з найбільш важливих завдань - збереження життя і здоров'я людини при виконанні будь-яких робіт.

Охорона праці включає в себе цілий комплекс заходів, що мають нормативно-правове забезпечення. До числа таких заходів відносяться організаційно-технічні, соціально-економічні, санітарно-гігієнічні та лікувально-профілактичні.

При роботі в сфері охорони праці слід керуватися, перш за все, Законом України про охорону праці. Крім того, Конституцією України, Кодексом законів про працю України, Законом України «Про загальнообов'язкове державне страхування від нещасного випадку на виробництві та професійного захворювання, які спричинили втрату працездатності», а також інших нормативно-правових актів, що регулюють відносини в сфері охорони праці, а також чинних міжнародних договорів, за згодою Верховної Ради України.

Норми Закону України про охорону праці:

визначають основні положення щодо реалізації конституційного права працівників на охорону їх життя і здоров'я в процесі трудової діяльності, на безпечні і здорові умови праці;

регулюють за участю відповідних органів державної влади відносини між працівником і роботодавцем з питань безпеки, гігієни праці та виробничої санітарії;

встановлюють єдиний порядок організації охорони праці в Україні.

Слід розуміти, що:

Охорона праці - система правових, соціально-економічних, організаційно-технічних, санітарно-гігієнічних і лікувально-профілактичних заходів та засобів, які спрямовані на збереження життя, здоров'я, а так само працездатності працівника в процесі трудової діяльності.

Працівник - особа, яка здійснює свою трудову діяльність на підприємстві або в організації, установі, яка виконує свої трудові обов'язки (функції) у відповідність з трудовим контрактом (договором).

Роботодавець - власник підприємства або організації, установи або уповноважений ним орган, незалежно від форми власності, виду діяльності, господарювання, а також фізична особа, яка використовує найману працю.

Нормативно-правові акти з охорони праці (НПАОП) - правила, норми, регламенти, положення, стандарти, інструкції та інші документи, обов'язкові для виконання.

4.2 Гарантії прав на охорону праці

Права на охорону праці під час укладання трудового договору

Умови трудового договору не можуть містити положень, що суперечать законам та іншим нормативно-правовим актам з охорони праці.

При укладанні трудового договору роботодавець повинен проінформувати працівника про умови праці та про наявність на його робочому місці небезпечних і шкідливих виробничих факторів, які ще не усунуто, можливі наслідки їх впливу на здоров'я та про права працівника на пільги і компенсації за роботу в таких умовах відповідно до законодавства і колективного договору.

Працівникові не може пропонуватися робота, яка за медичним висновком протипоказана йому за станом здоров'я. До виконання робіт підвищеної небезпеки та потребують професійного добору, допускаються особи за наявності висновку психофізіологічної експертизи.

Усі працівники згідно із законом підлягають загальнообов'язковому державному соціальному страхуванню від нещасного випадку на виробництві та професійного захворювання, які спричинили втрату працездатності.

Права працівників на охорону праці під час роботи

Умови праці на робочому місці, безпека технологічних процесів, машин, механізмів, устаткування та інших засобів виробництва, стан засобів колективного та індивідуального захисту, що використовуються працівником, а також санітарно-побутові умови повинні відповідати вимогам законодавства.

Працівник має право відмовитися від дорученої роботи, якщо створилася виробнича ситуація, небезпечна для його життя чи здоров'я або для оточуючих його людей, або для виробничого середовища чи довкілля. Він зобов'язаний негайно повідомити про це безпосереднього керівника або роботодавця. Факт наявності такої ситуації за необхідності підтверджується спеціалістами з охорони праці підприємства за участю представника профспілки, членом якого він є, або уповноваженої працівниками особи з питань охорони праці (якщо професійна спілка на підприємстві не створювалася), а також страхового експерта з охорони праці.

За період простою з причин, передбачених частиною другою цієї статті, які виникли не з вини працівника, за ним зберігається середній заробіток.

Працівник має право розірвати трудовий договір за власним бажанням, якщо роботодавець не виконує законодавство з охорони праці, не додержується умов колективного договору з цих питань. У цьому випадку працівникові виплачується вихідна допомога в розмірі, передбаченому колективним договором, але не менше тримісячного заробітку.

Працівника, який за станом здоров'я відповідно до медичного висновку потребує надання легшої роботи, роботодавець повинен перевести за згодою працівника на таку роботу на термін, зазначений у медичному висновку, і в разі потреби встановити скорочений робочий день та організувати проведення навчання працівника з набуття іншої професії відповідно до законодавства.

На час призупинення експлуатації підприємства, цеху, дільниці, окремого виробництва або устаткування органом державного нагляду за охороною праці чи службою охорони праці за працівником зберігається місце роботи, а також середній заробіток.

Право працівників на пільги і компенсації за важкі та шкідливі умови праці

Працівники, зайняті на роботах з важкими і шкідливими умовами праці, безоплатно забезпечуються лікувально-профілактичним харчуванням, молоком або рівноцінними харчовими продуктами, газованою солоною водою, мають право на оплачувані перерви санітарно-оздоровчого призначення, скорочення тривалості робочого часу, додаткову оплачувану відпустку, пільгову пенсію, оплату праці в підвищеному розмірі та інші пільги і компенсації, що надаються у визначеному законодавством порядку.

У разі роз'їзного характеру працівникові виплачується грошова компенсація на придбання лікувально-профілактичного харчування, молока або рівноцінних йому харчових продуктів на умовах, передбачених колективним договором.

Роботодавець може за свої кошти додатково встановлювати за колективним договором (угодою, трудовим договором) працівникові пільги і компенсації, не передбачені законодавством.

Протягом дії укладеного з працівником трудового договору роботодавець повинен не пізніше ніж за 2 місяці письмово інформувати працівника про зміни виробничих умов та розмірів пільг і компенсацій з урахуванням тих, що надаються їм додатково.

Забезпечення працівників спецодягом, іншими засобами індивідуального захисту, мийними та знешкоджувальними засобами

На роботах з шкідливими і небезпечними умовами праці, а також роботах, пов'язаних із забрудненням або несприятливими метеорологічними умовами, працівникам видаються безплатно за встановленими нормами спеціальний одяг, спеціальне взуття та інші засоби індивідуального захисту, а також мийні та знешкоджувальні засоби. Працівники, які залучаються до разових робіт, пов'язаних. Залежно від орієнтації вікон рекомендується наступне забарвлення стін і підлоги: вікна орієнтовані на південь: - стіни зеленувато-блакитного або світло-блакитного кольору; підлога - зелений; вікна орієнтовані на північ: - стіни світло-оранжевого або оранжево-жовтого; підлога

- червонувато-оранжевий; вікна орієнтовані на схід: - стіни жовто-зеленого кольору; підлога зелена або червонувато-оранжевий; вікна орієнтовані на захід: - стіни жовто-зеленого або голубувато-зеленого кольору; підлога зелена або червонувато-оранжевий.

У приміщеннях, де знаходиться ПК, необхідно забезпечити наступні величини коефіцієнта: для стелі: 60 ... 70%, для стін: 40 ... 50%, від підлоги: близько 30%. Для інших: 30 ... 40%.

Правильно спроектоване освітлення покращує умови роботи, знижує стомлюваність, сприяє підвищенню продуктивності, благотворно впливає на виробниче середовище, надаючи позитивну психологічну дію на працюючого, підвищує безпека і знижує травматизм.

Недостатність освітлення призводить до напруги зору, послаблює увагу, приводить до настання передчасної стомленості. Надмірно яскраве освітлення викликає осліплення, роздратування і різь в очах.

Неправильний напрямок світла на робочому місці може створювати різкі тіні, відблиски, дезорієнтувати працюючого. Всі ці причини можуть призвести до нещасного випадку, тому такий важливий правильний розрахунок освітленості.

Існує три види освітлення - природне, штучне і поєднане

Природне освітлення - освітлення приміщень денним світлом, що потрапляє через світлові прорізи в зовнішніх огорожувальних конструкціях приміщень.

4.3 Програміст зобов'язаний

- піклуватися про особисту безпеку і здоров'я, а також про безпеку і здоров'я оточуючих людей у процесі виконання будь-яких робіт або під час знаходження на території підприємства;
- знати і виконувати вимоги інструкцій з охорони праці і по видах робіт на своєму робочому місці;

- виконувати роботу відповідно до вимог інструкційно-технологічної карти;
- вміти користуватися засобами індивідуального і колективного захисту;
- знати і виконувати Правила поведінки з устаткуванням, інвентарем, користуватися технічним паспортом на устаткування;
- знати і виконувати обов'язки з охорони праці, передбачені колективним договором (трудовим договором), правилами внутрішнього трудового розпорядку підприємства, в тому числі:
- вчасно починати і закінчувати роботу, дотримуватися розкладу технологічної і обідньої перерв;
- не виконувати роботи, що не передбачені змінним завданням;
- не перебувати на роботі в неробочій час без відповідного розпорядження керівника;
- дотримуватись правил корпоративного поведінки;
- проходити в установленому порядку медичні огляди;
- вміти надати першу допомогу потерпілому від нещасного випадку;
- перед початком роботи перевіряти справність устаткування, огорожень, інженерно-технічних засобів безпеки, інвентарю, засобів пожежогасіння;
- співпрацювати з роботодавцем у справі організації безпечних і нешкідливих умов праці, особисто вживати можливих заходів щодо усунення будь-якої ситуації, що створює загрозу її життю чи здоров'ю або людям, які її оточують та навколишньому природному середовищу;
- при виявленні недоліків чи небезпеки зобов'язана повідомляти безпосереднього керівника або іншу посадову особу.

Перевірити:

- справність обладнання, інструменту, приладів;
- наявність і справність достатнього освітлення, вентиляції, обладнання тощо;

- перевірити справність рубильників, розеток, штепсельних з'єднань тощо.

У випадку виявлення будь-яких відхилень, несправностей, пошкоджень негайно повідомити директора Підприємства.

4.4 Загальні вимоги безпеки під час виконання роботи

Виконувати роботу згідно із своїми посадовими обов'язками. Не залишати без нагляду своє робоче місце, коли обладнання підключено до електромережі. У випадку виявлення будь-яких відхилень, несправностей, пошкоджень негайно повідомити директора Підприємства.

4.5 Вимоги безпеки після закінчення роботи

- Перевірити своє робоче місце.
- Відключити від електромережі електрообладнання.
- Закрити вікна.
- Вжити заходів особистої гігієни: старанно вимити руки, при можливості прийняти душ.
- Привести в порядок спеціальний одяг, зняти і прибрати його в окреме місце.

4.6 Вимоги безпеки в аварійній ситуації

При виявленні небезпечної ситуації (пожежа, землетрус, радіаційна безпека, неполадки в електрогосподарстві тощо) для власного життя та життя співробітників заспокоїти і заспокоїти оточуючих. Не усувати самому несправностей електромережі та електрообладнання, а вимкнути загальне електропостачання. При виявленні пожежі зобов'язаний негайно викликати пожежну частину. Вжити заходів згідно з планом евакуації на випадок пожежі, виробничих та природних явищ та вивести працівників у безпечне місце. Організувати роботу ДПД щодо збереження майна та цінних паперів. При появі сторонньої особи, яка застосовує протиправні дії щодо безпеки життєдіяльності оточуючих, викликати поліцію.

У випадку травмування працівників або клієнтів під час роботи

підприємства необхідно викликати швидку допомогу або за потреби надати першу долікарську допомогу, за необхідності створити комісію по розслідування нещасного випадку, видати акт встановленого зразка, наказ про підсумки розслідування, повідомлення про наслідки нещасного випадку. Дії при наданні першої долікарської допомоги. Надання першої медичної допомоги починати з оцінки загального стану потерпілого і на підставі цього скласти думку про характер пошкодження. У разі різкого порушення або відсутності дихання, зупинки серця негайно зробити штучне дихання та зовнішній масаж серця, викликати швидку медичну допомогу.

Дії при ураженні електричним струмом:

- необхідно звільнити потерпілого від дії електричного струму, відключивши електрообладнання від джерела живлення, а при неможливості відключення – відтягнути його від струмоведучих частин за одяг або застосувавши підручний ізоляційний матеріал;
- за відсутності у потерпілого дихання і пульсу необхідно робити йому штучне дихання і непрямий (зовнішній) масаж серця, звернувши увагу на зіниці. Розширені зіниці свідчать про різке погіршення кровообігу мозку. При такому стані оживлення необхідно починати негайно, після чого викликати швидку медичну допомогу.

Дії при пораненні:

- для надання першої допомоги при пораненні необхідно розкрити індивідуальний пакет, накласти на рану стерильний перев'язувальний матеріал і зав'язати її бинтом;
- якщо індивідуального пакету немає, то для перев'язки необхідно використати чисту носову хустинку, чисту полотняну ганчірку тощо. На те місце ганчірки, що приходиться безпосередньо на рану, бажано накапати декілька капель настойки йоду, щоб одержати пляму розміром більше рани, а після цього накласти ганчірку на рану.

Дії при переломах, вивихах, ударах, розтягненні:

- при переломах і вивихах кінцівок необхідно пошкоджену кінцівку укріпити шиною, фанерною пластинкою, палицею, картоном або іншим подібним предметом. Пошкоджену руку можна також підвісити за допомогою перев'язки або хустки до шиї і прибинтувати до тулуба;
- при передбачуваному переломі черепа (несвідомий стан після удару голови, кровотеча з вух або рота) необхідно прикласти до голови холодний предмет (грілку з льодом або снігом, чи холодною водою) або зробити холодну примочку;
- при підозрі перелому хребта необхідно потерпілого покласти на дошку, не підіймаючи його, чи повернути потерпілого на живіт обличчям у низ, наглядаючи при цьому, щоб тулуб не перегинався з метою уникнення ушкодження спинного мозку;
- при переломі ребер, ознакою якого є біль при диханні, кашлю, чханні, рухах необхідно туго забинтувати груди чи стягнути їх рушником під час видиху.

Дії при теплових опіках:

- при опіках вогнем,- парою, гарячими предметами ні в якому разі не можна відкривати пухирі, які утворюються, та перев'язувати опіки бинтом;
- при опіках першого ступеня (почервоніння) обпечене місце обробляють ватою, змоченою етиловим спиртом; при опіках другого ступеня (пухирі) обпечене місце обробляють спиртом, 3 % марганцевим розчином або 4 % розчином таніну;
- при опіках третього ступеня (зруйнування шкіряної тканини) накривають рану стерильною пов'язкою та викликають лікаря.

Дії при кровотечі:

- для того, щоб зупинити кровотечу, необхідно підняти поранену кінцівку вгору, кровоточиву рану закрити перев'язувальним матеріалом

(із пакета), складеним у клубочок, придавити її зверху, не торкаючись самої рани, потримати протягом 4

- при сильній кровотечі, яку не можна зупинити пов'язкою, застосовується здавлювання кровоносних судин, які живлять поранену область, за допомогою згинання кінцівок у суглобах, а також пальцями, джгутом або закруткою;

Всі зазначені норми закріплені на законодавчому рівні і покликані забезпечити своїм співробітникам безпечну робочу обстановку.

ВИСНОВКИ

В результаті виконаної роботи було покращено управління проектами компанії. Клієнти можуть авторизуватись на сайті, і отримати всю інформацію що до своїх діючих проектів. Призначенням Сайту є зручне управління проектами та збільшення обсягу клієнтів.

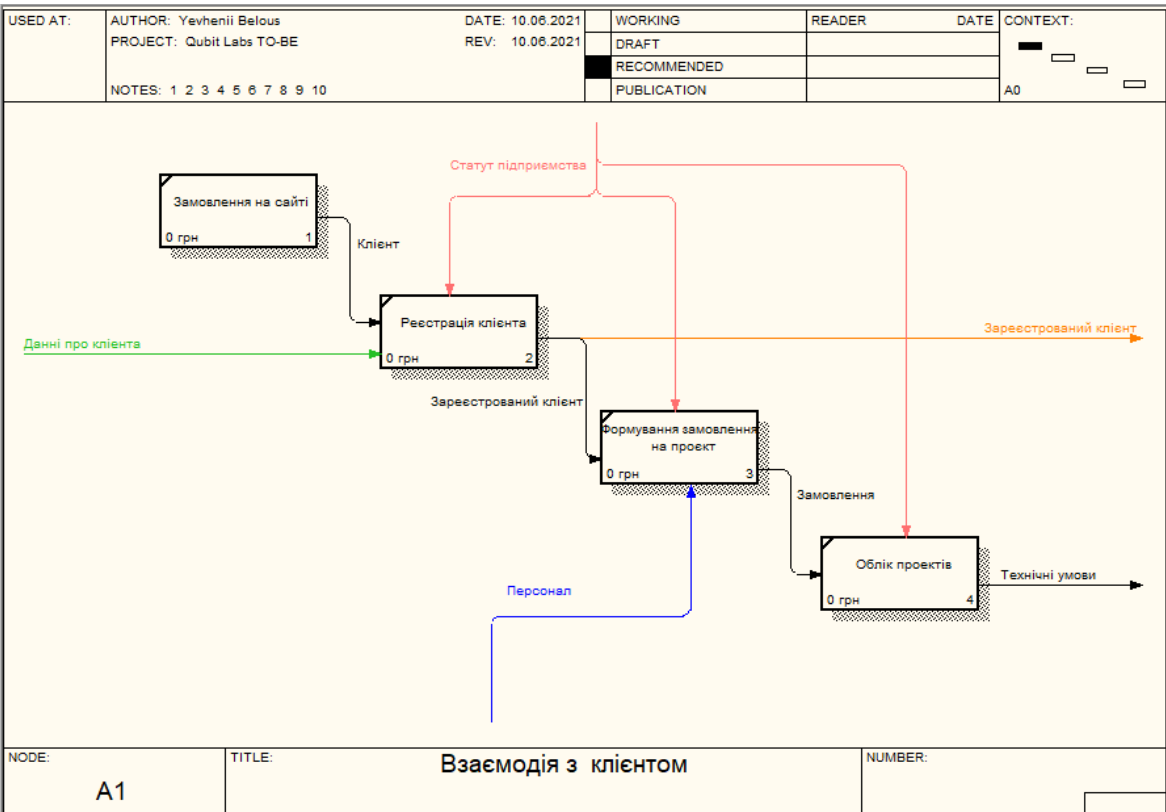
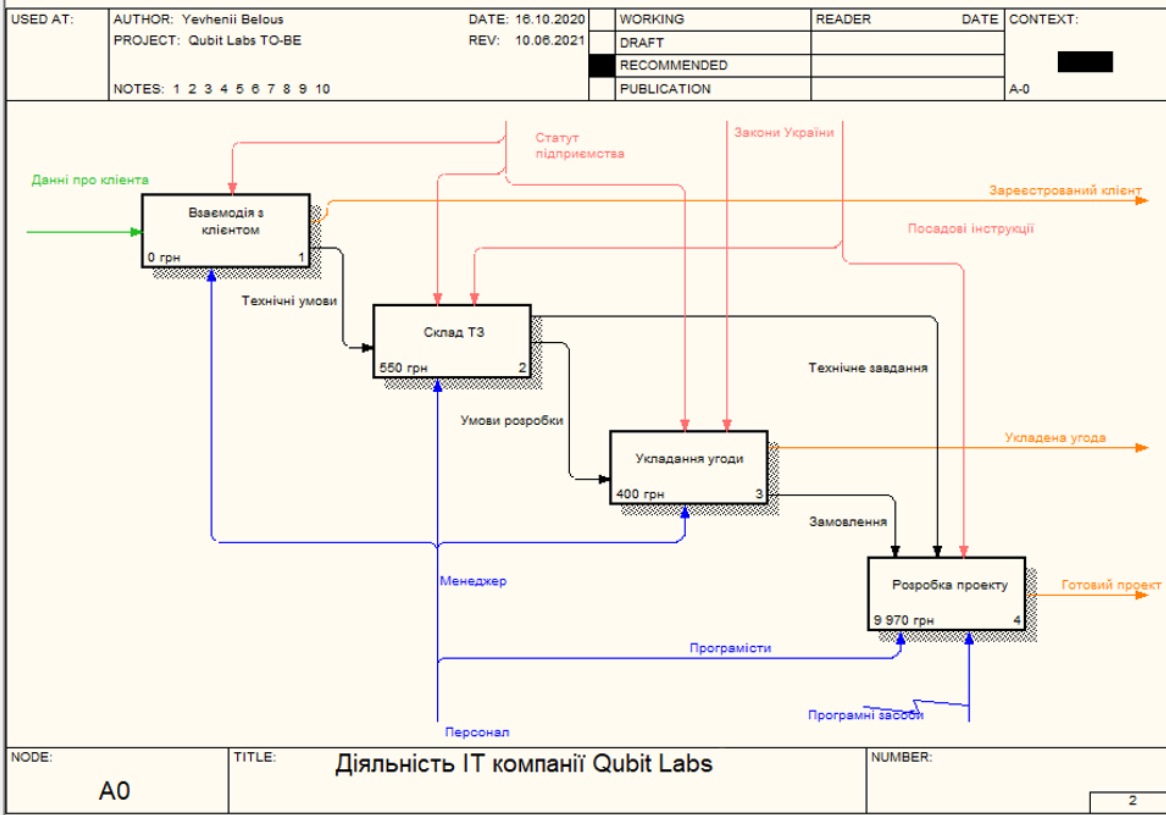
У кваліфікаційній роботі було: Проведено аналіз компанії, Створено br-win модель діяльності підприємства, Розроблено концептуальну модель TO-BE після впровадження Сайту, Розраховано ефект від впровадження, Обґрунтовано доцільність розроблення сайту, Розроблено вимоги до сайту та його функцій, Розроблено вимоги до дизайну, Створено макет сайту, Розроблено інтерфейс користувача для взаємодії з сайтом, Розроблені функції для зручного управління, Розроблена інструкція користувача для користування Сайтом, Описана Охорона праці на фірмі. Сайт відповідає всім вимогам які були поставлені при формування задачі.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

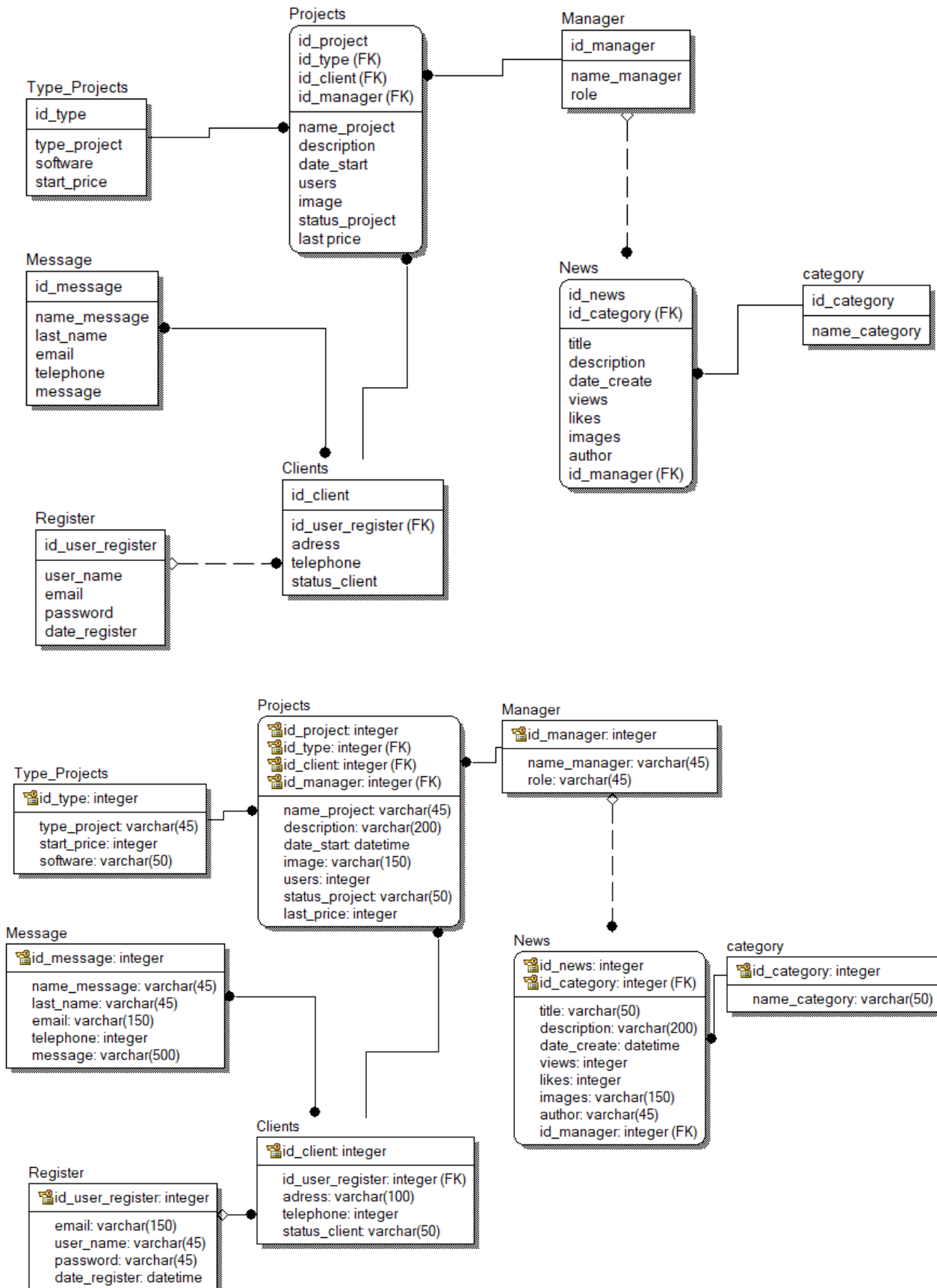
1. Qubit Labs - <https://qubit-labs.com/>
2. CMS сравнение - https://en.wikipedia.org/wiki/Content_management_system
3. HTML - <https://uk.wikipedia.org/wiki/HTML>
4. CS - https://developer.mozilla.org/ru/docs/Learn/Getting_started_with_the_web/
5. Javascript - <https://developer.mozilla.org/ru/docs/Web/JavaScript>
6. Popular Framework - <https://vc.ru/dev/147263-populyarnye-freymvorki-javascript>
7. React/Angular/Vue - <https://habr.com/ru/post/476312/>
8. React - <https://ru.reactjs.org/>
9. Angular - <https://angular.io>
10. Vue.js - <https://vuejs.org>
11. Node - <https://uk.wikipedia.org/wiki/Node.js>
12. VS Code - https://uk.wikipedia.org/wiki/Visual_Studio_Code
13. MySQLWorkbench - <https://www.mysql.com/products/workbench/>
14. BP win - <http://bourabai.kz/cm/bpwin.htm>
15. Er win - https://ru.wikipedia.org/wiki/ERwin_Data_Modeler
16. Купчик М.П., Гандзюк, Литвиненко А.М., Іваненко. О. В. Основи Охорони праці. - К.:
17. Охорона праці - <https://legalitgroup.com/service/ohorona-praci-v-it-kompanii/>
18. Основи дизайну сайту - <https://webstudio2u.net/ua/design-web/342-webdesign-start.html>
19. Основи React - <https://learn-reactjs.ru/basics/topics>
20. Основи створення Серверу - <https://nodejs.org/ru/docs/guides/getting-started-guide/>

Додатки

Додаток А Вр-win Модель



Додаток Б. Логічна та фізична модель БД



Додаток В Компонентний підхід розробки

```
import React from "react";
import { BrowserRouter as Router, Switch, Route } from "react-router-dom";
import { Home } from "./home/Home";
import { About } from "./about/About";
import { OurProducts } from "./ourproducts/OurProducts";
import { Blog } from "./blog/Blog";
import { ContactUs } from "./contact/ContactUs";
import { Header } from "./header/Header";
import { Footer } from "./footer/Footer";
export class App extends React.Component {
  render() {
    return (
      <Router>
        <Header />
        <Switch>
          <Route exact path="/">
            <Home />
          </Route>
          <Route path="/about">
            <About />
          </Route>
          <Route path="/our-produce">
            <OurProducts />
          </Route>
          <Route path="/blog">
            <Blog />
          </Route>
          <Route path="/contact-us">
            <ContactUs />
          </Route>
        </Switch>
        <Footer />
      </Router>
    );
  }
}
```

(Код головної сторінки)

```
import React from "react";
import { Fade } from "../components/slider/slider";
import { HomeDesc } from "./HomeDesc";
import { Products } from "./products";
import { Button } from "../components/button/button";
import { HomeParallax } from "./HomeParallax";
import { HomeBlog } from "./homeBlog";
import img1 from "../images/slider-1.webp";
import img2 from "../images/slider-2.webp";
import img3 from "../images/slider-3.webp";
import img4 from "../images/slider-4.webp";
import img5 from "../images/slider-5.webp";
import img6 from "../images/slider-6.webp";
import img7 from "../images/slider-7.webp";
import img8 from "../images/slider-8.webp";

export class Home extends React.Component {
  render() {
    const photos = [
      { name: "Photo-1", id: 1, url: img1 },
      { name: "Photo-2", id: 2, url: img2 },
      { name: "Photo-3", id: 3, url: img3 },
      { name: "Photo-4", id: 4, url: img4 },
    ];
    const photos2 = [
      { name: "Photo-1", id: 1, url: img5 },
      { name: "Photo-2", id: 2, url: img6 },
      { name: "Photo-3", id: 3, url: img7 },
      { name: "Photo-4", id: 4, url: img8 },
    ];
    const slider = photos.map((photo) => {
      return (
        <div>
          <img src={photo.url} alt={photo.name} />
        </div>
      );
    });
    const slider2 = photos2.map((photo) => {
      return (
        <div>
          <img src={photo.url} alt={photo.name} />
        </div>
      );
    });
  }
}
```

```

</div>
);
});
const setting = {
  accessibility: false,
  dots: true,
  arrows: false,
  fade: true,
  infinite: true,
  speed: 1500,
  slidesToShow: 1,
  slidesToScroll: 1,
  autoplay: true,
  autoplaySpeed: 3000,
  pauseOnHover: false,
};
return (
  <div className="home-page">
    <div className="my-custom-fade">
      <div className="slider__content">
        <h2 className="slider__title">Welcome to Our Farm</h2>
        <h3 className="slider__subtitle">Fresh | Organic | Local</h3>
        <Button class="slider__button" text="Learn More" link="/about" />
      </div>
      <Fade pic={slider} setting={setting} />
    </div>
    <HomeDesc />
    <Products
      pic={slider2}
      title="Season's Produce"
      btn="yes"
      setting={setting}
    />
    <HomeParallax />
    <HomeBlog />
  </div>
);
}
}

```

(Код Сторінки Home)

```

import React from "react";
import aboutIcon from "../images/icon.svg";
import { Desc } from "../components/desc/desc";
import "../scss/home-desc.scss";
const descTitle = "Support Sustainable Farming";
const descContent = [
  {
    content:
      "I'm a paragraph. Click here to add your own text and edit me. It's easy. Just click "Edit Text" or double click me
to add your own content and make changes to the font. ",
  },
];
export class HomeDesc extends React.Component {
  render() {
    return (
      <div className="about-home">
        <Desc
          title={descTitle}
          content={descContent}
          img={aboutIcon}
          picClass="desc__col_color1"
          btn="yes" />
      </div>
    );
  }
}

```

(Код компонента HomeDesc)

```

import React from "react";
import { Button } from "../button/button";
import "../desc.scss";

export class Desc extends React.Component {
  render() {
    const btn = this.props.btn;
    const content = this.props.content;
    const list = [];

    for (let i = 0; i < content.length; i++) {
      let par = content[i];
      list.push(par);
    }
  }
}

```

```
}  
  
function CheckBtn(props) {  
  if (btn === "yes") {  
    return <Button class="desc__btn" text="Join Our CSA" link="/" />;  
  } else {  
    return "";  
  }  
}  
  
return (  
  <div className="desc__wrapper">  
    <div className="desc__content desc__col">  
      <div className="desc__text">  
        <h2 className="desc__title">{this.props.title}</h2>  
        <hr className="desc__line" />  
        {list.map((par) => {  
          return <p className="desc__text">{par.content}</p>;  
        })}  
      </div>  
  
      <CheckBtn btn={btn} />  
    </div>  
  
    <div className={this.props.picClass + " desc__img desc__col"}>  
      <img  
        src={this.props.img}  
        alt={this.props.img}  
        className="desc__picture"  
      />  
    </div>  
  </div>  
>;  
}
```