

**ДЕРЖАВНИЙ УНІВЕРСИТЕТ ТЕЛЕКОМУНІКАЦІЙ  
НАВЧАЛЬНО-НАУКОВИЙ ІНСТИТУТ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ**

Кафедра інженерії програмного забезпечення

**Пояснювальна записка**

до бакалаврської кваліфікаційної роботи  
на ступінь вищої освіти бакалавр

**на тему:** «Розробка системи «OnBoardingTracker» для відстеження кандидатів мовою C# та Java Script»

Виконав: студент 4 курсу, групи ПД– 42

спеціальності

121 Інженерія програмного забезпечення

(шифр і назва спеціальності)

Морозов О.О.

(прізвище та ініціали)

Керівник Дібрівний О.А.

(прізвище та ініціали)

Рецензент \_\_\_\_\_

(прізвище та ініціали)

Нормоконтроль \_\_\_\_\_

(прізвище та ініціали)

Київ – 2022

**ДЕРЖАВНИЙ УНІВЕРСИТЕТ ТЕЛЕКОМУНІКАЦІЙ**  
**Навчально-науковий інститут інформаційних технологій**

Кафедра Інженерії програмного забезпечення

Ступінь вищої освіти - «Бакалавр»

Спеціальність - 121 Інженерія програмного забезпечення

ЗАТВЕРДЖУЮ

Завідувач кафедри

Інженерії програмного

забезпечення

\_\_\_\_\_ О.В. Негоденко

« \_\_\_\_ » \_\_\_\_\_ 2021 року

**ЗАВДАННЯ**  
**НА БАКАЛАВРСЬКУ РОБОТУ СТУДЕНТУ**

\_\_\_\_\_ Морозов Олександр Олегович

(прізвище, ім'я, по батькові)

1. Тема роботи: «Розробка системи «OnBoardingTracker» для відстеження кандидатів мовою C# та Java Script»

Керівник роботи \_\_\_\_\_ Дібрівний Олесь Андрійович

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

затверджені наказом вищого навчального закладу від “18” лютого 2022 року  
№\_\_.

2. Строк подання студентом роботи 03.06.2022

3. Вихідні дані до роботи:

3.1 Офіційна документація Microsoft.

3.2 Visual Studio.

3.3 Visual Studio Code

3.4 MSSQL.

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

4.1 Аналіз актуальності та проблематики розроблюваної системи.

4.2 Аналіз та вибір інструментів для реалізації системи.

4.3 Проектування системи.

4.4 Висновки

## 5. Перелік графічного матеріалу

- 5.1 Титульний слайд
- 5.2 Мета, об'єкт, предмет та наукова новизна дослідження
- 5.3 Актуальність роботи
- 5.4 Аналоги
- 5.5 Технічні завдання
- 5.6 Програмні засоби реалізації
- 5.7 Зв'язок елементів додатку
- 5.8 Апробація результатів дослідження
- 5.9 Висновки
- 5.10 Кінцевий слайд

6. Дата видачі завдання 11.04.2022

**КАЛЕНДАРНИЙ ПЛАН**

№ з/п	Назва етапів бакалаврської роботи	Строк виконання етапів роботи	Примітка
1	Вибір теми бакалаврської роботи	11.04-14.04	Виконано
2	Підбір науково-технічної літератури	15.04 - 17.04	Виконано
3	Дослідження аналогів та актуальності додатку	18.04 - 21.04	Виконано
4	Аналіз та вибір інструментів для розробки додатку	22.04 - 25.04	Виконано
5	Проектування та реалізація	25.04 - 07.05	Виконано
6	Висновки, оформлення роботи	07.05 – 10.05	Виконано
7	Розробка демонстраційних матеріалів	11.05 - 15.05	Виконано
8	Здача роботи	03.06	

Студент

\_\_\_\_\_ ( підпис )

Морозов О.О.

(прізвище та ініціали)

Керівник роботи

\_\_\_\_\_

Дібрівний О.А.





## РЕФЕРАТ

Текстова частина бакалаврської роботи с. 60, рис. 18, джерел 13

*Об'єкт дослідження* – процес відстежування кандидатів рекрутерами.

*Предмет дослідження* – технології, за допомогою яких можливо відстежити всі стадії рекрутменту.

*Мета дослідження* – розробка системи «OnBoardingTracker» для відстежування кандидатів у процесі рекрутменту.

У дипломній роботі був проведений аналіз аналогів що вже існують, переваг та недоліків програмних інструментів для додатка.

Виконано опис програм, які були використані у ході роботи, та середі розробки.

Система реалізована за допомогою мови програмування C# та JavaScript, з використанням фреймворку Angular, та сервісу Auth0 для реалізації аутентифікації.

Ключові слова: Парі, JavaScript, C#, Angular, .NET, MSSQL, GIT.

# ЗМІСТ

ВСТУП.....	8
1. ТЕОРЕТИЧНА ЧАСТИНА .....	9
1.1 Загальне уявлення про процес рекрутменту .....	9
1.2 Методи найму співробітників .....	11
1.3 Які навички потрібні щоб стати рекрутером.....	14
1.4 Аналіз наявних додатків .....	19
1.5 Складання технічного завдання .....	21
2. ОПИС ПРОГРАМНИХ ЗАСОБІВ .....	26
2.1 Мова програмування C# та .NET .....	26
2.2 Angular.....	28
2.3 Figma.....	30
Розділ 2.4 MSSQL.....	35
2.5 Photoshop .....	41
2.6 Auth0.....	44
2.7 Angular Material .....	44
2.8 Entity Framework.....	46
3. ЗАГАЛЬНІ ВІДОМОСТІ ДОДАТКУ .....	49
3.1 Розробка концепції.....	49
3.2 Проектування структури та створення додатку .....	50
ВИСНОВКИ.....	60

## ВСТУП

*Оцінка сучасного стану об'єкта розробки.* На сьогоднішній всі ІТ-компанії мають процес рекрутингу. Але більшість з компаній використовують звичайні Excel таблиці, які не вирішують усі задачі процесу рекрутменту.

*Об'єкт дослідження* – процес відстежування кандидатів рекрутерами.

*Предмет дослідження* – технології, за допомогою яких можливо відстежити кандидатів.

*Мета дослідження* – розробка системи для відстеження кандидатів.

*Наукова новизна проекту* – створення системи для відстеження кандидатів з використанням сервісу Auth0 для аутентифікації, щоб забезпечити максимальну безпеку аккаунтів у системі.

У дипломному проекті був проведений аналіз додатків-аналогів, та виявлені переваги та недоліки.

Backend частину системи було розроблено на мові програмування C# з використанням .NET Core у середовищі розробки Visual Studio 2022. Frontend частину системи було розроблено на мові програмування Html/Css, JavaScript, та фреймворку Angular з бібліотекою Material та Bootstrap.

Додаток може використовуватися не тільки рекрутерами, а й менеджментом компанії, для слідкування за роботою рекрутерів.



## 1. ТЕОРЕТИЧНА ЧАСТИНА

### 1.1 Загальне уявлення про процес рекрутменту

Що таке рекрутмент?

Рекрутмент – це процес пошуку та підбору кандидатів. Коректніше оперуватиме словом «рекрутмент», давайте далі перейдемо на нього. Залежно від обсягу організації рекрутмент є обов'язком низки працівників. У більших організаціях може бути цілі групи рекрутерів, а інших — лише один рекрутер. У невеликих компаніях відповідальність за рекрутмент може нести найманий менеджер. Крім того, багато організацій передають рекрутинг стороннім фірмам. Компанії майже завжди набирають кандидатів на нові посади через оголошення, дошки оголошень, сайти соціальних мереж і т. д. Багато компаній використовують програмне забезпечення для підбору персоналу, щоб ефективно шукати кращих кандидатів. Незважаючи на це, рекрутинг зазвичай працює спільно з відділом кадрів або як його частина

Тоді що таке процес рекрутменту?

Це шлях, через який проходить кандидат для того щоб стати співробітником компанії.

Також цей шлях ділиться на чотири етапи:

- Пошук (Sourcing).
- Відбір (Selection).
- Пропозиція на роботу (Job offer).
- Онбординг (Onboarding).

Пошук – це етап, у якому спеціаліст з підбору персоналу (рекрутер) починає пошук кандидатів. Пошук буває або активний, або пасивний.

Активний пошук – це коли рекрутер активно шукає кандидатів, та на пряму зв'язується з кандидатом на різних спеціалізованих форумах, LinkedIn, соціальних мережах і так далі.

Пасивний пошук – це коли компанія розміщує вакансію, наприклад, на

Work.ua, Robota.ua, LinkedIn і так далі. Та кандидат сам зв'язується з рекрутером.

Після того, як рекрутер знайшов кандидатів, настає етап відбору.

Відбір – це етап, на якому рекрутер вже отримав достатню кількість кандидатів, та за допомогою кілька послідовних кроків відсіюють відсіюють половину, які не підходять, і шукають оптимального кандидата.

Оптимальність оцінюється такими термінами: час (коли кандидат готовий приступити до роботи), якість (відповідність кандидата вимогам посади та культурі компанії та команди), ціна (відповідність зарплатних очікувань кандидата та рівня оплати вакансії). Послідовні кроки етапу відбору кандидатів називають "воронкою відбору" (англ. "recruitment funnel").

Після етапу відбору, компанія дає пропозицію на роботу. Зазвичай, це договір, в якому прописані всі обов'язки двох сторін (компанія, кандидат), та всі юридичні моменти, наприклад, при яких умовах може бути розірвати контракт, яку страховку дає компанія (якщо дає), дати договору, на яку посаду наймають кандидата, заробітна плата та яким способом буде нараховуватися і так далі. Після отримання кандидатом договору він його підписує, та попереджує про припинення трудових відносин на поточному місці роботи.

Коли всі юридичні моменти вирішено, та кандидат вже стає частиною компанії, робота рекрутера не закінчується, бо настає етап онбордингу.

Завершальним етапом рекрутменту є онбординг нового співробітника (англ. onboarding, від "on" - на, і "board" - борт, або "ласкаво просимо на борт!"). Це комплекс заходів спрямованих на скорочення часу непродуктивної роботи нового співробітника через відсутність необхідних знань, навичок та контактів усередині та поза організацією, а також необхідної інфраструктури (інструменти, доступи, обладнання тощо). У радянській літературі був прийнятий термін "адаптація", який має на увазі швидше психофізіологічну адаптацію організму людини до нових умов, цей процес безумовно відбувається і у сучасного офісного співробітника, проте акцент в організаційних заходах з онбордингу спрямований на інфраструктурні

елементи для плавного входження на посаду та організацію, ніж на психофізіологію людини. Фахівці HR та рекрутери організують та керують процесом онбордингу, а не психофізіологічною адаптацією.

Заходи з онбордингу включають:

- вступне навчання - "введення в організацію", в якому новий співробітник знайомиться з компанією, її історією, місією, стратегією, цінностями та культурою;
- вступне навчання - "введення у підрозділ, введення у функцію", в якому співробітник знайомиться з конкретним функціональним підрозділом, його завданнями, командою тощо.

Зазвичай, " запровадження організацію " чи т.зв. загальний вступний курс, вступне навчання (англ. induction training, crash course, bootcamp) організовує і проводить HR, а "введення в підрозділ, посаду" проводить безпосередній лінійний керівник. Деякі компанії, наприклад, європейські бюджетні авіалінії (EasyJet, RyanAir та ін.) виводять частину первинного, вступного навчання своїх майбутніх бортпровідників за периметри трудових відносин. Провівши первинний відбір кандидатів компанії призначають їм дистанційні курси для самостійного навчання, і роблять пропозицію про роботу кандидатам, які успішно пройшли теоретичне навчання (за результатами тесту знань). Таким чином, онбординг може починатися ще до закінчення процесу рекрутменту.

## 1.2 Методи найму співробітників

Існує 7 методів найму співробітників:

### 1. Внутрішнє просування

При пошуку відповідного співробітника рекомендується використовувати як внутрішні, так і зовнішні мережі. Внутрішнє включає розміщення оголошення про вакансію на веб-сайті вашої компанії і звернення до власного відділу кадрів, а також пошук потенційних переказів і просування

по службі всередині вашої організації або серед співробітників, які нещодавно вийшли на пенсію.

Реальність така, що ідеальна кандидатура для вашої вакансії може вже існувати у чотирьох стінах. Підвищення на посаді поточного чи недавнього співробітника може відкрити іншу вакансію, яку значно легше заповнити. Цей тип переїзду також може заощадити гроші вашої компанії, які інакше були б витрачені на адаптацію та навчання нового співробітника. Інститут Саратог повідомляє, що зовнішнє наймання коштує в 1,7 рази більше, ніж внутрішній.

Ці внутрішні найми також часто мають вищі показники успіху. Дослідження Пенсільванського університету показують, що внутрішні співробітники показують вищі результати протягом перших двох років роботи.

## 2. Публікуйте на сайтах вакансій та дошках оголошень

Немає недоліку в кар'єрних сайтах та дошках оголошень про вакансії, які можна використовувати для розміщення оголошень про вакансії — Monster, Career Builder та Indeed, і це лише деякі з них. Використовуючи ці платформи, ви або ваш рекрутер отримаєте доступ до пулу як пасивних, так і активних претендентів.

Можливо, одна з найбільших переваг використання різних дошок вакансій для пошуку талантів полягає в тому, що ви можете побачити різні набори навичок, доступні за допомогою логічних методів пошуку, які допоможуть вам знайти те, що часто називають фіолетовою білкою. кандидатів, необхідних посаду.

## 3. Відвідуйте мережеві заходи

Один з методів найму, який є менш популярним, але часто дуже ефективним, — це відвідування місцевих мережевих заходів. Відвідування заходів не тільки підвищує впізнаваність вашого бренду та зміцнює місцеву присутність вашої компанії, але також може допомогти виявити пасивні таланти. Майте на увазі, що цей тип взаємодії може бути вашим першим і останнім шансом справити враження на певних кандидатів, тому вам потрібно

добре підготуватись перед мережевим заходом.

#### 4. Залучайте співробітників та галузеві рекомендації

Якщо вам потрібен співробітник, який ідеально підходить для вашої корпоративної культури, розгляньте можливість використання рекомендацій від співробітників та колег по галузі. Ваші нинішні співробітники можуть бути не тільки впливовими представниками вашої організації, але й часто можуть бути більш ефективними у виявленні сильних кандидатів, оскільки вони краще знайомі з повсякденними потребами команди.

Насправді, на думку 48% компаній, рекомендації співробітників є головним джерелом якісного найму. Найм потенційного кандидата, який вже має стосунки з одним із ваших співробітників, може значно спростити процес адаптації та допомогти розвинути почуття товариства з самого початку.

Ви також можете співпрацювати з рекрутинговими агентствами, які можуть надати рекомендації від лідерів галузі та експертів. Hunt Club має експертну мережу зі старших керівників, засновників та віце-президентів, які направляють найкращих кандидатів на відкриті вакансії. Спираючись на досвід та мережу кадрового агентства або хедхантера, ви отримаєте доступ до більшої кількості талантів та підвищите ймовірність того, що кандидат зацікавиться вашою роллю.

#### 5. Використовуйте соціальні мережі для пошуку

Понад 84% компаній використовують соціальні мережі для найму, що робить їх надзвичайно популярним методом найму. Крім виходу в свою особисту мережу відразу ж, ви також можете використовувати своє оголошення про вакансії для євангелізації та отримання рекомендацій від сім'ї, друзів та знайомих.

Щоразу, коли ви використовуєте LinkedIn для найму талантів, переконайтеся, що вакансія конкретна, коротка, прямолінійна і її не можна пропустити. Оскільки користувачі швидко прокручують свої часові рамки, ваш пост буде важче побачити, ніж ви думаєте. Обов'язково опублікуйте своє відкриття не лише у власній хроніці та на сторінці вашої компанії, а й у

відповідних мережевих групах на платформі. Це відкриє ще більше можливостей для потенційних клієнтів, щоб знайти ваше відкриття.

## 6. Пасивне вербування

Цілком можливо, що ви ніколи не знайдете ідеального кандидата на вакансію і часто це відбувається тому, що більшість талановитих людей не є активними пошуками роботи. Аутсорсинг ваших потреб у підборі персоналу професійним рекрутинговим компаніям або хедхантеру, мабуть, найкращий спосіб відстежити найкращі таланти.

Хоча у вас може не бути часу, щоб з'ясувати, що саме потрібно вашому бізнесу в новому наймі, наявність надійного рекрутера, який присвятить час виявленню найкращих талантів у вашій галузі, встановленню зв'язків та уявленню вашої можливості працевлаштування, може бути дуже ефективним.

## 7. Долучайте штучний інтелект

Штучний інтелект все частіше використовується для фільтрації заявок та швидкого відновлення роботи, звужуючи коло відповідних кандидатів. Однак на набагато глибшому рівні штучний інтелект можна використовувати навіть у процесах відбору та співбесіди. Замість проведення співбесіди безпосередньо з рекрутером або керівником бізнесу, чат-бот може проводити співбесіду, ставлячи запитання кандидатам і використовуючи їх відповіді для кваліфікації чи дискваліфікації кандидатів у режимі реального часу.

Цей метод проклав шлях компаніям для найму великих обсягів у галузях з високою плинністю кадрів та постійною потребою у нових співробітниках. Принаймні це дозволяє рекрутерам і компаніям просівати великі групи кандидатів, залишаючи лише кілька найкращих кандидатів для фінальних співбесід із компаніями.

### 1.3 Які навички потрібні щоб стати рекрутером

Більшість рекрутерів мають спільні навички, які роблять їх ефективними у своїй роботі. Вони включають поєднання технічних та комунікативних

навичок, які абороджені особистості рекрутера, або вивчені та відточені з часом.

Комунікативні навички для підбору персоналу.

Ця сім'я навичок підбору персоналу тісно пов'язана з вашою особистістю. Це навички, які є частиною того, хто ви є, і ті, які ви можете використати на свою користь у середовищі найму. Багато з них також можна відточити та вдосконалювати з часом, доклавши належних зусиль.

Ось деякі з комунікативних навичок рекрутера, які поділяють більшість чудових рекрутерів:

### 1. Сильні комунікативні навички

Вміння спілкуватися через різноманітні канали є важливою навичкою для рекрутерів. Від вас очікується, що ви будете писати ефективні пости про роботу та соціальні мережі, листуватися з кандидатами та менеджерами з найму електронною поштою, по телефону та особисто. Очікується, що ви будете діяти як обличчя компанії для кандидатів, головна особа для менеджерів з найму та зв'язок між двома сторонами протягом усього процесу найму.

В результаті виконання цих обов'язків рекрутери повинні мати можливість чітко та коротко спілкуватися як з окремими кандидатами та співробітниками, так і з громадськістю.

### 2. Впевненість

Підбір персоналу передбачає багато спілкування та спілкування з різними людьми. Очікується, що ви будете впевнено розмовляти з кандидатами, менеджерами з найму, вашими колегами та експертами в різних галузях. Через це рекрутери повинні мати певну впевненість у своїх комунікативних здібностях.

Впевненість також є цінною навичкою підбору персоналу, коли справа доходить до найму на посаду або галузі, про які ви мало знаєте. Рекрутери повинні бути впевнені у власних здібностях та досвіді, щоб зробити правильний вибір для роботи та мати можливість підтримати процес

прийняття рішень у разі запитання.

### 3. Природна цікавість

Рекрутери повинні прагнути дізнатися про кандидата більше, ніж те, що написано в їх резюме або супровідному листі. Немає способу дізнатися кандидата і як він буде працювати на роботі, якщо у вас немає цікавості та ініціативності, щоб задати правильні запитання.

Розуміння людей є величезною перевагою, коли намагаєтеся стати успішним рекрутером. Так само й цікавість щодо спроб нових тактик і каналів пошуку нових і різних кандидатів. Ніколи не переставайте бути допитливими або пробувати нові речі, і ви ніколи не перестанете вдосконалюватися як рекрутер.

### 4. Навички прислуховування до людей

Слухання йде рука об руку з цікавістю. Якщо ви приділите час, щоб посправжньому вислухати своїх кандидатів, менеджери з найму та колеги дадуть вам ціннішу інформацію, ніж будь-яка інша тактика, яку ви можете знайти.

Уважно слухаючи, що говорять ваші кандидати та колеги, ви точно дізнаєтеся, чого вони хочуть. Потім ви можете використовувати цю інформацію для розробки стратегії набору та ведення переговорів. Чим більше ви слухаєте, тим більш цінну інформацію ви отримаєте, щоб скеровувати ваше рішення.

### 5. Здатність читати та демонструвати позитивну мову тіла

Більшість людей не подумали б, що мова тіла є особливо корисною навичкою персоналу, але це так. Часто, особливо в ситуації з наймом, люди не говорять точно, що вони думають. Уміння читати чийсь мову тіла дасть вам чудове уявлення про їхні справжні думки чи наміри.

Мова тіла також йде іншим шляхом для рекрутерів. Вміння контролювати та демонструвати позитивну мову тіла під час особистих співбесід є цінною тактикою ведення переговорів. Це може допомогти вам контролювати загальний напрямок і настрої розмови.

Технічні навички найму.



Тепер давайте подивимося на деякі навички технічного рекрутингу, які ви повинні мати, щоб бути успішним у своїй кар'єрі.

#### 6. Аналіз даних

З появою складних HR-технологій підбір персоналу перетворився на гру, яка керується даними. Важливою сучасною навичкою підбору персоналу, яку ви повинні розвинути, є здатність аналізувати та використовувати дані на свою користь.

Знайдіть час, щоб дізнатися, які ключові показники ефективності (KPI) або показники набору є найважливішими для вашої компанії. Використовуйте підбір персоналу на основі даних, щоб знайти способи зробити найм кращим та ефективнішим.

#### 7. Навички маркетингу

Підбір персоналу – це маркетинг і продаж вашої компанії потенційним кандидатам. Це означає, що ви думаєте і ведете себе так, ніби ви маркетолог і продавець, об'єднані в одне ціле.

Рекрутери повинні розвивати ті ж навички реклами та веб-аналітики, які використовують маркетологи для тестування та відточування своїх програм генерування потенційних клієнтів. Це допоможе вам:

- Знайдіть найкращі канали.
- Пишіть переконливі оголошення про роботу
- Використовуйте технології, щоб збільшити охоплення ваших оголошень про роботу
- Знайдіть приховані кишень висokoактивних кандидатів

Що стосується продажів, рекрутери повинні зосередитися на розвитку своєї здатності демонструвати переваги роботи в своїй компанії, а також на льоту адаптуватися до запитань і відштовхувати кандидатів. Чим краще ви вмієте просувати компанію та розміщувати вакансії, тим ефективніше ви будете знаходити найкращих кандидатів.

#### 8. Вміння користуватися технологіями

Сьогодні рекрутерам дуже важливо бути технічно підкованими. Це

означає поєднання ваших навичок підбору персоналу зі здатністю ефективно використовувати технологію найму, як-от системи відстеження заявників (ATS), платформи управління продуктивністю та HRIS. Чим кращі ваші технічні навички, тим ефективніше ви зможете використовувати ці платформи на свою користь.

Технологічні навички також знадобляться при прийомі на високотехнічні посади. Спілкування з кандидатами їхньою мовою та впевнене їхнє перевірка – це чудовий спосіб стати успішним рекрутером.

### 9. Критичне мислення

Величезним фактором для прийняття справедливого та успішного рішення про найм є критичне обдумування інформації, яку ви отримуєте. Рекрутери повинні взяти всю доступну інформацію про кандидата, відфільтрувати те, що не є суттєвим, і зробити неупереджений висновок про те, кого найняти.

Таким чином, критичне мислення є надзвичайно важливою навичкою підбору персоналу, яка гарантує, що ви приймаєте рішення на основі фактів, а не інтуїції.

### 10. Багатозадачність і тайм-менеджмент

Рекрутери часто жонглюють кількома вакансіями одночасно. Це означає писати оголошення про роботу, перевіряти кандидатів, проводити перевірку біографії, планувати співбесіди і, зрештою, завершувати пропозиції роботи. Це на додаток до спілкування з багатьма кандидатами, які претендують на будь-яку посаду, а також їхніми обов'язками зв'язку з менеджерами з найму та іншими членами команди з підбору персоналу. Крім того, їхнє робоче навантаження збільшується на кожну вакансію, за яку вони відповідають!

Якщо ви відчуваєте труднощі на своїй нинішній посаді і шукаєте способи стати успішним рекрутером, то хорошим першим кроком було б покращити свої навички багатозадачності та управління часом.

## 1.4 Аналіз наявних додатків

Я знайшов точно такий самий додаток за ідеєю відстеження кандидатів, його назва Zoho Recruit (рис. 1.1).



Рисунок 1.1. – Логотип Zoho додатку

Я зареєструвався на сайті цього додатка та взяв безкоштовний період для ознайомлення з продуктом. Після кілька годин проведенних на цьому сайті, я знайшов чітко виражені плюси цього продукту, та чітко виражені мінусу.

Почнемо з плюсів Zoho Recruit:

- Веб додаток працює стабільно, не знайшов багів при використанні цього продукту.

- Є інтеграція з LinkedIn, Azure, Microsoft, Google.
- Для початківців є документація та багато відео на ютубі з цим продуктом.
- Багато полів для кандидатів та рекрутерів. Можна заповнити всю необхідну інформацію.
- Лояльні ціни.

А тепер продовжимо мінусами:

- Дуже нагромаджено функціями - це дуже великий мінус, бо компанія яка бере цей продукт, повинна виділити ще місяць або два для того, щоб рекрутери змогли розібратися у всьому функціоналу.
- Дуже похмурий та не сучасний дизайн (рис. 1.2).

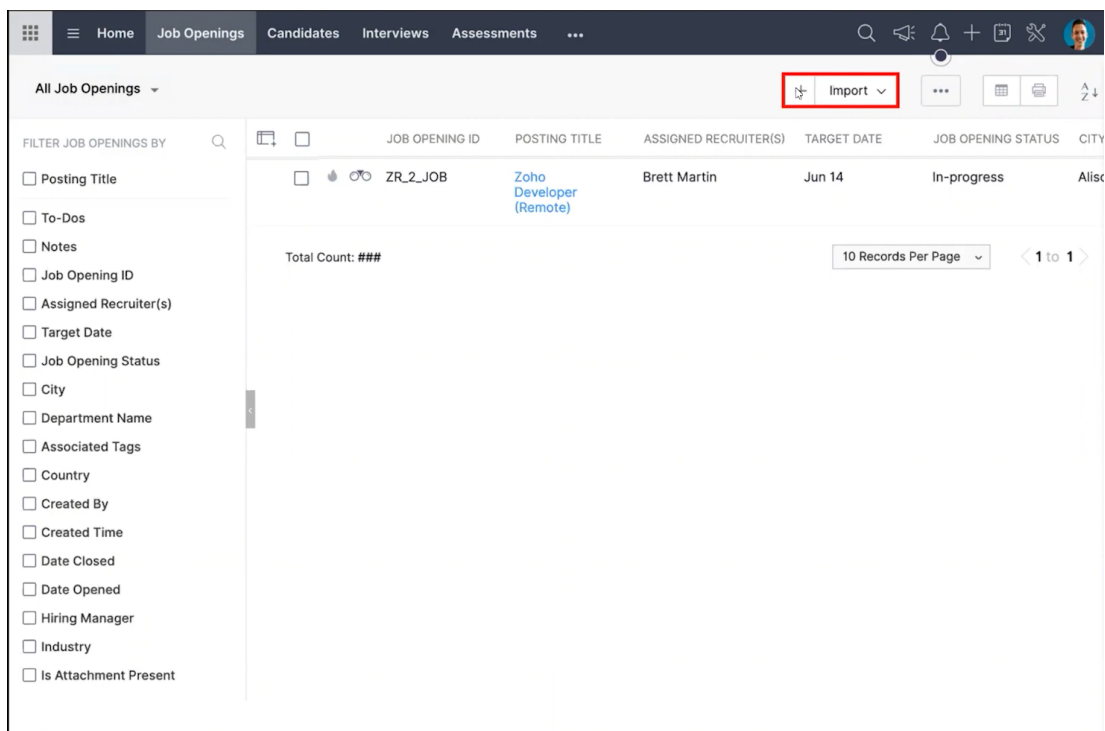


Рисунок 1.2. – Додаток Zoho recruiter

- Підтримка сайту дуже довго відповідає на пошту.
- Не підтримує резервного копіювання, якщо станеться який небудь баг в базі даних на сайті, або рекрутер повидаляє потрібні данні,

то відновити не буде можливості.

- Дуже нав'язлива реклама на мою пошту для купівлі тарифу.

Висновок: продукт непоганий в цій ціновій категорії, для розробників можна працювати з їх API, але є мінуси на які компанія не звертає уваги, тому є над чим працювати.

### 1.5 Складання технічного завдання

У ході дослідження було вирішено створити кросплатформовий веб-додаток. Для рекрутерів які зазвичай мають і комп'ютер, і телефон – веб-додаток це ідеальне рішення тому, що вони зможуть користуватися додатком чим завгодно.

Після того як було вирішено створити веб-додаток, коло вибору мови для frontend частини звужився. Я вирішив обрати HTML/CSS та JavaScript для розробки frontend частини сайту.

Після вибору мови програмування, потрібно було вирішити який саме фреймворк для JavaScript вибрати з двох найвідоміших:

- React
- Angular
- Vue

Плюси фреймворку React:

- Дуже швидкий завдяки реалізації React Virtual DOM і має безліч оптимізацій рендерингу.
- Прив'язка даних є односторонньою, що означає менше небажаних побічних ефектів.
- React реалізує концепції функціонального програмування (FP), створюючи простий у тестуванні та багаторазово використовуваний код.

Мінуси фреймворку React:

- React відходить від компонентів на основі класів, що може стати перешкодою для розробників, яким комфортніше працювати з об'єктно-орієнтованим програмуванням (ООП).
- Спільнота ділиться за способами написання CSS у React, які поділяються на традиційні таблиці стилів (CSS Modules) та CSS-in-JS (тобто Emotion та Styled Components).
- Змішування шаблонів із логікою (JSX).

Плюси фреймворку Vue:

- Посилений HTML. Це означає, що Vue.js має багато параметрів подібних до Angular, а це, завдяки використанню різних компонентів, допомагає оптимізації HTML-блоків.
- Адаптивність. Може бути здійснений швидкий перехід від інших фреймворків до Vue.js через схожість з Angular та React з точки зору дизайну та архітектури.
- Крихітний розмір. Vue.js важить близько 20 КБ, зберігаючи при цьому свою швидкість і гнучкість, що дозволяє досягти кращої продуктивності в порівнянні з іншими платформами.

Мінуси фреймворку Vue:

- Нестача 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 знаходиться на початковій стадії.
- Ризик надмірної гнучкості. Іноді у Vue.js можуть виникнути проблеми при інтеграції у різні проекти, і поки що немає досвіду можливих рішень.

Плюси фреймворку Angular:

- Angular використовується разом з Typescript. Він має виняткову підтримку для цього.
- Одностороння прив'язка даних, що забезпечує виняткову

поведінку програми, що зводить до мінімуму ризик можливих помилок.

- MVVM (Model-View-ViewModel), яка дозволяє розробникам працювати окремо над тим самим розділом програми, використовуючи той самий набір даних.
- Впровадження залежностей від компонентів, пов'язаних з модулями та модульністю в цілому.

Мінуси фреймворку Angular:

- Щодо повільної продуктивності, враховуючи різні показники. З іншого боку, це можна легко вирішити, використовуючи так званий `ChangeDetectionStrategy`, який допомагає вручну контролювати процес рендерингу компонентів.
- Багато різних структур (`Injectables`, `Components`, `Pipes`, `Modules`, etc.) порівняно з `React (Components)`.

Після аналізу плюсів та мінусів двох фреймворків, я вирішив зупинитися на Angular тому, що я збираюся робити `OnPushStrategy` архітектуру яка вирішує проблему повільної продуктивності, та хоч багато людей і відносить розбивку на різні структури мінусом, я рахую що це не значний мінус.

З frontend частиною було вирішено, тепер потрібно вибрати мову програмування для backend частини сайту, та базу даних. Для веб-додатку у котрому важливі данні, було вирішено вибрати SQL-подібну базу даних, тому потрібно вибирати мову програмування з якою буде легше за всього працювати з SQL-подібними базами даних. Найвідоміші та зручні мови програмування для backend частини сайту:

- C#
- Java

Плюси мови програмування Java:

- Програмування Java відповідає стандартам, що пред'являються до корпоративних обчислювальних систем. Завдяки безлічі бібліотек розробники можуть реалізувати широкий функціонал, необхідний

для організації.

- Об'єктно-орієнтована платформа дає змогу позначити типи даних, їх структурні особливості та функціонал, що до них застосовується. Керуючи структурами даних, можна створити зв'язок між об'єктами. ООП дає можливість використовувати об'єкти для інших програм повторно, не показує дані, до яких не покладено доступ для запобігання помилкам, грамотно структурує програми, полегшуючи процес удосконалення застарілих кодів.
- Функції Java усувають основні вразливості. Завдяки наявності Security Manager з політикою безпеки, що дозволяє настроїти правила доступу до баз, вдається уникнути вразливостей.

Мінуси мови програмування Java:

- Низька швидкість і продуктивність, якщо порівнювати з C та C++, викликана компіляцією та абстракцією віртуальної машини. До інших проблем, що уповільнюють швидкість, можна віднести софт, що очищає пам'ять, який займає понад 20% продуктивності процесора та негативно впливає на загальну продуктивність. Неправильні установки кешування часто призводять до навантаження пам'яті. При взаємному блокуванні потоків також виникають помилки, пов'язані з нестачею пам'яті.

Плюси мови програмування C#:

- Останнім часом багато вдосконалюється. Оскільки C# був створений пізніше, ніж Java та інші мови, потрібно було дуже багато допрацювати. Також це стосується популяризації та безкоштовності – було обіцяно відкрити вихідний код, а інструменти (Visual Studio, Xamarin) стали безкоштовними для приватних осіб та невеликих компаній.
- Багато синтаксичного цукру. Синтаксичний цукор – це такі конструкції, які створені для полегшення написання та розуміння коду (особливо якщо це код іншого програміста) та не відіграють



ролі при компіляції.

- Підтримка Майкрософт. На відміну від Java, який не пішов на користь перехід у власність Oracle, C# добре розвивається завдяки зусиллям Microsoft.

Мінуси мови програмування C#:

- Орієнтованість в основному тільки на .NET (на Windows платформу).
- Зберегли оператор go to.
- Безкоштовність тільки для невеликих компаній, учнів та програмістів-одинаків. Для великих команд покупка ліцензій коштуватиме недешево. Тому якщо у вас є своя фірма, то доведеться розщедритися.

Після аналізу плюсів та мінусів обох мов програмування, також враховуючи вимоги для роботи з SQL-подібними базами даних, я вирішив вибрати C# та бібліотек Entity Framework для роботи з SQL-подібними базами даних.

Entity Framework – ідеальне рішення, якщо йдеться про невеликий проект та економію часу. Entity Framework варто використовувати тоді, коли у додатку є багато однотипних нескладних запитів, але якщо при цьому ви готові пожертвувати якщо ви готові пожертвувати швидкістю проекту.

## 2. ОПИС ПРОГРАМНИХ ЗАСОБІВ

### 2.1 Мова програмування C# та .NET

C# – об'єктно-орієнтована мова програмування, розроблена корпорацією Microsoft, метою якої є створення додатків що працюють у .NET. C# має багато спільного з C++, C, Java, бо це мова високого рівня. Якщо ви колись користувалися цими мовами, розпочати роботу над додатком буде набагато швидше й легше.

Ця мова програмування досить розповсюджена при створенні комп'ютерних та веб-додатків. Важко уявити яка мова програмування буде більш доречна при розробці комп'ютерного додатка для Windows. Дуже допомагає наявність великої кількості користувачів даної мови, бо як наслідок, збільшується кількість інформації, людей що зможуть у разі чого допомогти з труднощами які трапляються при розробці, та найголовніше – мова покращується, створюється нове програмне забезпечення та інструменти.

Можна створювати дуже об'ємні програми, які водночас будуть гнучкими, та з легкістю будуть здатними до модифікації за допомогою об'єктно-орієнтованого підходу. Ця мова програмування підтримує головні якості об'єктно-орієнтованого програмування, такі як інкапсуляція, поліморфізм та успадкування.

З новими версіями C# тільки покращується та розвивається, додаючи до свого функціоналу нові можливості, як наприклад асинхронні методи та лямбда вирази, обробка виключень та динамічне з'єднання.

Повнофункціональний веб-сайт також можна написати на C#, бо за наявності об'єктно-орієнтованого підходу, він стає легше обслуговуватись, більш ефективним, та масштабним.

За допомогою C# створюється безліч ігор, він поєднується з ігровим рушієм Unity, найпопулярнішим інструментом для розробки відеоігор. Кінцевий продукт розробки, може буди застосований як на комп'ютерних

системах, так і на ігрових консолях, мобільних пристроях. Такому успіху у створенні різноманітних ігор C# сприяє об'єктно-орієнтований підхід, інтерфейси, автоматичне керування оперативною пам'яттю комп'ютера за допомогою збирання сміття.

Головні переваги C#:

- наявність синтаксичного цукру у великій кількості. Загалом це синтаксичні можливості, які спрощують розробку, та за допомогою яких можливо перетворити велику кількість коду у більш лаконічний, читабельний та компактний. Його використання не впливає на компіляцію;
- фіксований розмір типів даних;
- підтримка переважної більшості продуктів Microsoft;
- написання додатків та програм для операційних систем IOS, Android, Linux та MacOS за допомогою Xamarin;
- індивідуальні розробники та невеликі компанії мають здатність безкоштовно користуватися багатьма інструментами, наприклад Azure, Mac Pro, Windows Server, Visual Studio;
- об'єктно-орієнтований підхід, який насправді використовується правильно;
- безпечний (збір сміття, виключення, managed);
- досить легкий для освоєння як при першому ознайомленні з програмуванням через свою зрозумілість, так і для програмістів, завдяки схожості з іншими мовами програмування.

Якщо мова йде про технології платформи .NET, наприклад про ASP.NET, Xamarin, Windows Forms, WPF, то у багатьох випадках люди мають на увазі мову програмування C#, це працює й у зворотному напрямку. Вони між собою тісно пов'язані, але не треба розглядати їх як одне й теж. Мова програмування C# була спроектована та розроблена для роботи з .NET Framework, та зміст .NET набагато ширший.

Бути середовищем, яке підтримує створення і виконання сильно

розподілених компонентних додатків, ось яку мету має .NET Framework. Цей фреймворк постачає безпеку, сумісне використання мов програмування, та загальну модель розробки під платформу Windows. Він надає можливість використовувати створені API-інтерфейси для створення комп'ютерних додатків як WPF, веб-додатків як ASP.NET, веб-аплетів як Silverlight, для роботи з базами даних як ADO.NET, та інших.

Додатки створенні за допомогою .Net є кросплатформними програмами, а фреймворк має здатність до роботи з мовами програмування COBOL, Visual Basic, C ++, як ми вже знаємо C #, та іншими.

Платформу .NET Framework заведено вважати відповіддю компанії Microsoft на іншу платформу, Java, що мала велику популярність у той час, коли була створена компанією Sun Microsystems.

Мова програмування C# була створена саме для ASP.NET, який у результаті й був написаний за допомогою цієї мови програмування. Технологія активних серверних сторінок Active Server Page, скорочено ASP.NET, стала новим інструментом платформи .NET. Вона сприяла швидкій розробці веб додатків, що взаємодіють з базами даних.

Засновник корпорації Microsoft Білл Гейтс визнав, що найкращий винахід його компанії – це .NET. Це висловлювання не було даремним, бо за допомогою потужної платформи .NET Framework створюються чудові додатки.

Найсильніші сторони .NET:

- величезна бібліотека класів;
- розмаїття технологій;
- підтримка різних мов.

## 2.2 Angular

Angular - структурований фреймворк для динамічних веб-додатків. Дозволяє використовувати HTML як мову шаблонів, а також розширювати

HTML-синтаксис, щоб код вашої програми виглядав коротко і лаконічно. Використовуючи Data-binding і Dependency injection (що йдуть із коробки), ви також зменшуєте кількість свого коду. І все це на браузерному JavaScript та працює з будь-якою серверною технологією.

Angular - це те, чим був би HTML, якби він був призначений для додатків. HTML – чудова декларативна мова для статичних документів. Він не містить багато чого для створення програм, і, як результат, побудова в ньому веб-додатків зводиться до того, щоб обдурити браузер, змусивши його робити те, що я хочу.

Невідповідності між динамічними додатками та статичними документами часто вирішуються такими способами:

бібліотеки - колекції функцій, які часто використовуються під час створення веб-застосунків. Ви розширюєте свій код і викликаєте функції з бібліотеки, коли потрібно. Наприклад, jQuery;

фреймворки - певні реалізації веб-застосунків, де ваш код деталізує логіку. Фреймворк бере на себе управління і викликає ваш код у випадках, коли йому необхідно зробити щось специфічне для програми. Приклади: knockout, sproutcore і т.д.

Angular пропонує інший підхід. Він намагається звести до мінімуму невідповідність між документо-центричним HTML та потребами програми шляхом створення нових HTML-конструкцій. Angular навчає браузер новому синтаксису за допомогою конструкцій, які називають директивами. Можна навести такі приклади:

- Біндинг даних у вигляді `{{}}`.
- Управління структурами DOM для фрагментів DOM, що повторюються/приховуються.
- Підтримка форм та перевірок форм.
- Зв'язування коду з DOM-елементами.
- Угруповання HTML у повторно використовувані компоненти.
- Закінчене рішення

Angular позиціонується як закінчене рішення щодо розробки веб-додатків. Це означає, що це не просто частина загальної головоломки побудови веб-додатку, а є цілим комплексом. Це дає право Angular вказувати на те, як мають будуватися CRUD-програми. Але в той же час він намагається переконатися, що його думка з цього приводу лише відправна точка, яку Ви можете легко змінити. З коробки Angular включає:

- Все, що потрібно при побудові CRUD-додатки - це зв'язування з прив'язки даних, основних директив шаблонів, валідації форм, маршрутизації, deep-linking, компонентів, що використовуються повторно, і впровадження залежностей.
- Тестування: модульні (unit) тести, end-to-end тести, моки, test harnesses.
- Початковий додаток зі структурою директорій та скриптом тестування як відправна точка.

Область застосування Angular.

Angular спрощує розробку додатків надаючи розробнику вищий рівень абстракції. Як і будь-яку іншу абстракцію, за цю доводиться платити гнучкістю. Іншими словами, Angular підійде не для кожної програми. Angular призначений для CRUD-додатків. На щастя, CRUD-програми становлять як мінімум 90% всіх веб-додатків. Але важливо розуміти, коли Angular буде добрим вибором, а коли не дуже.

Ігри та редактори інтерфейсу – яскраві приклади дуже інтенсивних та складних маніпуляцій з DOM. Ці види програм відрізняються від CRUD-програм, і, як результат, не підходять для Angular. У цьому випадку щось ближче до заліза, як, наприклад, jQuery, може бути кращим варіантом.

## 2.3 Figma

Figma – це інструмент створення та графічного маніпулювання векторними елементами, що спеціалізується на розробці мобільних та веб

інтерфейсів, вебсайтів.

За допомогою Figma можна вносити зміни у проєкт в реальному часі через веб-браузер. Цей інструмент розроблений як система дизайну, якою може поділитися команда дизайнерів, та є потужним інструментом, який повністю відповідає сучасним тенденціям. Сьогодні не можна говорити про створення прототипів та засоби спільного проєктування, не згадуючи Figma, вона навіть входить до трійки кращих інструментів для розробки інтерфейсів, за думкою фахівців UI.

Запущений у 2016 році, цей інструмент мав приголомшливий успіх серед спільноти дизайнерів інтерфейсу користувача та фахівців з прототипування, щодо зручності використання інструменту, а також функціональних можливостей, які він може запропонувати. Він швидко стає прямим конкурентом таких інструментів, як Sketch або Adobe XD.

Figma належить до нового покоління інструментів для проєктування дематеріалізованих інтерфейсів. Він призначений для спільної роботи між різними зацікавленими сторонами в проєкті. На додаток до ресурсів та графічних елементів, які вона дозволяє експортувати, Figma пропонує можливість генерації коду, пов'язаного з різними елементами інтерфейсу.

Figma - це інструмент, який дозволяє вам ділитися та розповсюджувати систему дизайну в команді UI, UX та розробників. Він також пропонує обмін у режимі реального часу та спільні робочі функції.

Ці функції допомагають пришвидшити виробничий потік, дозволяючи співпрацю в реальному часі між різними підрозділами над одним проєктом.

Однією з головних характеристик цього інструменту є його легка доступність. Існує веб-версія, доступна через браузер, а також настільна версія для Windows та MacOS.

Попри на те, що Figma є повноцінним інструментом і пропонує безліч функцій, Figma залишається доступним інструментом для початківців. Звичайно, за умови, якщо він знає, що таке система проєктування, розуміє використання стилів і особливо логіку успадкування компонентів та об'єктів,

і дотримується документації від Figma. Його дизайн був розроблений, щоб надати UX-дизайнеру (або UI дизайнеру) інтуїтивно зрозумілий інструмент, який відповідає всім їх потребам редагування графіки та інтерфейсу.

Нові користувачі можуть легко навчитися користуватися цим інструментом та освоїти його досить швидко, щоб почати розробляти інтерактивні інтерфейси.

Інструмент також пропонує функції спільного використання моделей, що дозволяють їх переглядати в Інтернеті за простим посиланням. Ця функція є особливо цікавою, оскільки вона дозволяє отримувати відгуки або презентації під час зустрічей. Таким чином, створені моделі та прототипи доступні за цим простим веб-посиланням, що дозволяє їх переглядати та тестувати в простому веб-браузері.

Ще однією спільною функцією, яку пропонує Figma, є експорт коду. Це дозволяє дизайнерам ділитися CSS-кодом веб-макетів із розробниками інтерфейсів, щоб дозволити їм перевірити кожен елемент макету. Також є можливість експортувати код iOS та Android у мобільні додатки для кращого потоку.

Дуже важлива функція, яку ми можемо використовувати завдяки цьому інструменту – це додавання та модифікація коментарів у режимі реального часу. Ця функція особливо цікава, щоб мати швидший зворотний зв'язок та легше обробляти необхідні зміни.

Figma - це потужний інструмент, який поєднує в собі функціональність багатьох інструментів в одному додатку. Але до цього найбільш інноваційною особливістю є здатність Figma будувати систему дизайну:

1. Дизайн. Figma пропонує широкий спектр важливих інструментів для створення та керування векторними графічними елементами, таких як фігури, зображення, текстів, тощо. Отже, це перш за все інструмент графічного дизайну, який дозволяє створювати моделі веб-сайтів та мобільних додатків.



2. Презентація. Презентація моделей пропонується Figma за допомогою простого посилання. Таким чином, макети розроблені за допомогою цього інструменту, можна передавати для презентацій клієнтам або під час різних зустрічей UX. Просте веб-посилання забезпечує доступ до моделей через браузер.

3. Прототипування. Додавання інтерактивності користувача та взаємозв'язку між різними інтерфейсами додатка також можна реалізувати у Figma. Інструмент дозволяє додавати взаємодії, кнопки що можна натискати та анімацію для створення інтерактивних прототипів. Figma дозволяє в кращому випадку динамічно прототипувати веб-сторінку, але буде важчим у використанні для настільного програмного забезпечення.

4. Експорт із різних розроблених інтерфейсів. Це може бути код CSS для веб-сайтів та веб-програм, а також код iOS та Android для мобільних додатків. Це полегшує співпрацю між розробниками UI та UX, створення інтерфейсів, адаптованих до різних пристроїв

Отже, Figma це веб-додаток, який об'єднує функції чотирьох інструментів. Це робить інструмент потужним і досить повним додатком, який відповідає всім потребам дизайнерів та розробникам інтерфейсу.

Для забезпечення співпраці між різними зацікавленими сторонами одного проекту, Figma надає змогу поділитись всім проектом, або лише одним інтерфейсом з іншою людиною. Ця функціональність корисна у випадку зворотного зв'язку із замовниками, щодо певних інтерфейсів проекту або для контролю прогресу.

Не забувайте, що ця функція спільного доступу полегшує взаємодію вашим співрозмовником, покращує ефективність та сприяє кращій співпраці.

Насправді кожна властивість компоненту обробляється індивідуально для кращої настройки стилю. Ви можете зробити це на рівні кожного екземпляра та створити власні компоненти для своїх проектів.

Також можна поміняти місцями екземпляри для кращого дизайну. Не

забувайте також про функціональність "перетягування", що полегшує обмін екземплярами.

Дуже часто ігнорують те, що Figma - чудовий інструмент для роботи у браузері. У порівнянні зі своїми конкурентами він має багато переваг. Наприклад, Sketch доступний лише як настільна програма для MacOS.

Доступ до Figma можна отримати в будь-якому веб-браузері, що має Інтернет-з'єднання. Тож незалежно від того, перебуваєте ви в дорозі, працюєте віддалено або на зустрічі з клієнтом, ваші проекти завжди будуть під рукою.

Ваші файли будуть зберігатися в хмарі для кращої доступності та миттєвої доступності. Отже, це дозволяє працювати над своїми проектами з будь-якого місця.

Навіть якщо ви не можете отримати доступ до Інтернету, ви можете використовувати версію програми для роботи з комп'ютером. Ви можете встановити її на MacOS або Windows і використовувати в автономному режимі. Документи будуть оновлені після входу в систему.

Отже, з Figma ви можете працювати де завгодно, з Інтернетом або без нього, і це величезна його перевага перед конкурентами.

Для спільного використання графічних ресурсів та забезпечення співпраці між дизайнерами однакової структури, можна створювати та обмінюватися бібліотеками Figma. Вони дозволяють створювати системи дизайну та стандартизувати певні графічні елементи.

Насправді бібліотеки Figma дозволяють спільно використовувати компоненти, а також стилі. Також можна оновити ці елементи, щоб нові версії використовувались автоматично в різних файлах. Отже, така система дизайну дозволить дизайнерам дотримуватися графічних статутів та різних стилів.

Для створення реалістичних прототипів, Figma дозволяє створювати анімації, що ілюструють різні взаємодії. Цей інструмент пропонує додавати анімацію між різними екранами програм для більш плавного використання.

Додавання анімації дозволяє створювати більш реалістичні прототипи, які занурюють користувача в UX. Отже, це дозволяє отримати більш точні та

реалістичні відгуки клієнтів, оскільки вони будуть видаватися на основі інтерактивних прототипів.

Підбиваючи підсумки, Figma - це потужний інструмент для проектування графічних інтерфейсів. Опановуючи його, можна отримати велику перевагу у кар'єрі.

## Розділ 2.4 MSSQL

Microsoft SQL Server 2014 представляє нове покоління масштабованих рішень у галузі систем управління базами та сховищ даних для завдань, що потребують швидкого отримання та аналізу інформації. Нова версія націлена на вирішення широкого кола завдань у всіх галузях бізнесу, включаючи електронну комерцію та хмарні технології, та забезпечує комплексний підхід до управління та аналізу даних. Режим обчислень в оперативній пам'яті збільшує продуктивність та швидкість завантаження. У SQL Server 2014 передбачені інструменти бізнес-аналітики для корпоративних та приватних користувачів, а також підтримка критично важливих додатків. Microsoft розглядає SQL Server 2014 як черговий крок у рамках концепції Cloud OS, представленої в жовтні 2013 р., яка передбачає доопрацювання існуючих рішень для їх повної інтеграції з хмарними сервісами з точки зору інфраструктури, технологій, SQL Server 2014 додатків, ідентифікації, баз даних.

### Основні переваги Microsoft SQL Server 2014:

Повна Web орієнтованість. Здійснення запитів, аналіз та управління через Web. Використання мови XML для обміну інформацією між віддаленими системами. Простий та безпечний доступ до ресурсів за допомогою Web-браузерів, швидкий пошук необхідних документів. Аналіз потоків даних та отримання інформації про користувачів, у тому числі через Web.

Масштабованість та надійність. SQL Server 2014 забезпечує практично

необмежене зростання обсягів зберігання за рахунок збільшення надійності та масштабованості системи, використовуючи всі переваги мультипроцесорної обробки файлів. Це безпечна, надійна, масштабована платформа, що захищає інформацію в додатках та підвищує її доступність. Інноваційна інфраструктура управління, заснована на політиках, в ній дозволяє визначати політики для явного та автоматичного адміністрування серверних сутностей на одному або декількох серверах. Крім того, оптимізована платформа SQL Server 2014 відкриває шлях до передбачуваної продуктивності обробки запитів. Інфраструктура SQL Server 2014 стала масштабованою. Вона здатна формувати звіти та виконувати аналіз будь-якого обсягу та складності, одночасно полегшуючи користувачам доступ до ресурсів за рахунок більш тісної інтеграції з Microsoft Office. В результаті IT-фахівці можуть поширити використання бізнес-аналітики у всій організації. SQL Server 2014 дозволяє користувачам консолідувати різноманітні дані в корпоративному сховищі, виводячи організацію сховищ на новий рівень.

Швидкість створення рішень. SQL Server 2014 у поєднанні з .NET Framework зменшує час розробки, впровадження та виходу на ринок сучасних додатків, прискорює процес пошуку інформації, спрощує управління, дозволяє використовувати функції, що створюються користувачем в інших додатках, надає широкі можливості для створення Web-додатків. Середовище ADO.NET Entity Framework підвищує ефективність праці розробників, оскільки тепер вони мають справу не безпосередньо з таблицями та полями, а з логічними інформаційними сутностями, узгодженими з бізнес-вимогами. Більше того, вони можуть створювати програми, що дозволяють користувачам копіювати дані на власні пристрої, а потім синхронізувати їх із центральними серверами.

Можливість обробки обчислень в оперативній пам'яті (in-memory OLTP).

Можливість взаємодії з публічною хмарою Windows Azure. Хмарне резервування локальних систем та аварійне відновлення з хмари підвищує

відмовостійкість критичних для бізнесу додатків

Покращене колонкове сховище в пам'яті, вперше представлене в SQL Server 2012. Тепер воно стало оновлюваним та підтримує операції читання та запису з більш високими швидкостями обробки запитів та високими коефіцієнтами стиснення.

Покращена технологія AlwaysOn, що забезпечує підтримку високого рівня доступності даних та аварійного відновлення. Тепер платформа підтримує до 8 вторинних реплік, які залишаються доступними для читання навіть за наявності збоїв у мережі.

### 10 причин перейти на Microsoft SQL Server 2019

#### Серце сучасної платформи обробки даних

За останні 10 років SQL Server пройшов шлях від рішення для невеликих та середніх СУБД до потужної платформи даних рівня підприємства, розрахованої на критичні бізнес-програми з надійності та відмовостійкості. З кожним новим релізом SQL Server дедалі більше має право називатися єдиним центром управління всіма даними. SQL Server враховує всі сучасні вимоги щодо роботи з даними різних форматів та з різноманітних джерел та стає природним вибором для побудови платформи інтеграції, управління та аналізу будь-яких даних.

#### Вимоги до сучасної платформи обробки даних

В останні роки генерується та обробляється все більше даних, збільшується різноманітність їх форми та сенсу. Частина даних, як і раніше, має реляційний формат і генерується традиційними транзакційними інструментами. Зазвичай такі дані структуровані, їхній сенс і цінність добре зрозумілі та відомі. Але величезна кількість даних має набагато сиріший вигляд. Це дані з датчиків і сенсорів (той самий Інтернет речей), відеокамер, що записують пристрої. Ці дані, швидше за все, несуть цінність, але витягти її поки що набагато складніше.

Роль сучасної платформи даних – прийняти такі різноманітні дані, інтегрувати їх, обробити та отримати цінну для бізнесу інформацію. Крім того,

така платформа має:

Обробляти дані як у існуючих локальних інфраструктурах, і у хмарах. Цей гібридний стан триватиме ще невизначено довгий час.

Дозволяти переносити існуючі інструменти обробки даних до хмарної платформи без серйозних змін.

Дозволяти розробляти сучасні хмарні програми з нуля, використовуючи всі хмарні інструменти.

Аналізувати дані однаково успішно як у локальному середовищі, так і у хмарній платформі.

Якщо за хмарну частину обробки даних відповідає Azure SQL, то за локальну складову платформи Microsoft для зберігання та обробки даних – SQL Server 2019.

Еволюція SQL Server.

Підтримка Linux та Docker вперше реалізована в SQL Server 2017.

1. SQL Server спрощує розгортання, передачу та інтеграцію великих даних

Рішення обробки великих даних на основі Kubernetes, вбудоване в SQL Server, дозволяє легко розгорнути кластер великих даних і працювати з ним. Kubernetes забезпечує розгортання сховищ HDFS, реляційного модуля SQL Server та засобів аналітики Spark у вигляді контейнерів у рамках одного зручного пакету.

До складу SQL Server 2019 входять Spark та HDFS, які дозволяють виконувати читання та запис безпосередньо у HDFS, використовуючи SQL Server або Spark. Архітектура Kubernetes забезпечує гнучке масштабування обчислювальних потужностей та сховищ за запитом.

2. Інтеграція структурованих та неструктурованих даних

Сьогоднішні обсяги даних роблять нерозумним та не вигідним конвертацію всіх доступних даних у реляційні таблиці для зберігання в СУБД. Ще 2 роки тому Microsoft представила технологію PolyBase, що дозволяє екземпляру SQL Server обробляти запити Transact-SQL, які звертаються до

даних Hadoop та об'єднувати дані з Hadoop та SQL Server. У SQL Server зовнішня таблиця або зовнішнє джерело даних забезпечує з'єднання з Hadoop, віртуалізуючи зовнішні джерела даних без необхідності прямого імпорту в реляційну базу, і потім дозволяє звертатися до цих даних із запитами.

Таким чином, дані накопичуються у своєму природному форматі, не обов'язково реляційному, але можуть бути представлені у вигляді віртуальної таблиці. Віртуалізація дозволяє інтегрувати дані різного формату з різнорідних джерел та місць зберігання без їх реплікації та переміщення, створюючи єдину віртуальну матрицю даних.

### 3. Висока продуктивність

Не перший рік Microsoft підтверджує високу продуктивність SQL Server транзакційними тестами та тестами продуктивності сховищ даних. Версія 2019 відзначена відмінними результатами у наступних тестах:

- продуктивність OLTP;
- продуктивність DW для 1 ТБ, 10 ТБ та 30 ТБ;
- співвідношення ціни та продуктивності OLTP;
- співвідношення ціни та продуктивності DW для 1 ТБ, 10 ТБ та 30 ТБ.

### 4. Підтримка постійної пам'яті (PMEM)

Постійна пам'ять (Persistent Memory, PMEM) – це швидка пам'ять, що має можливість зберігати дані після відключення живлення. Вона дозволяє обробляти дані in-memory, позбавляючи необхідності передавати дані по каналах передачі і прискорюючи обробку запитів на 30% для інтенсивних робочих навантажень введення-виводу.

Будь-який файл SQL Server, розміщений на пристрої PMM, тепер доступний безпосередньо, минаючи стек зберігання операційної системи, використовуючи ефективні операції метасру.

### 5. Гібридна транзакційна/аналітична обробка (HTAP)

Модель HTAP дозволяє одночасно здійснювати операційні транзакції та аналітику на одних і тих же даних в одній пам'яті, також реалізуючи підхід in

memory.

#### 6. Інтелектуальне оброблення запитів

Паралелізація запитів та покращене масштабування частих запитів завдяки механізмам інтелектуальної обробки запитів роблять продуктивність значно вищою. Відкладена компіляція табличних змінних більш ніж 50% прискорює обробку запитів.

Сімейство функцій інтелектуальної обробки запитів.

Програми та інструменти аналітики працюють з усіма реляційними та великими даними через провідний екземпляр SQL Server за допомогою T-SQL.

#### 7. Безпека та відповідність вимогам

Захист конфіденційних даних за допомогою технології Always Encrypted із захищеними анклавами. Шифрування дома дозволяє виконувати криптографічні операції з конфіденційними даними без їх переміщення межі бази даних.

Криптографічні операції включають шифрування стовпців, і ці операції тепер можна виконувати за допомогою Transact-SQL, вони не вимагають переміщення даних з бази даних. Усередині захищених анклавів підтримуються всі повнофункціональні обчислення, включаючи зіставлення та порівняння діапазонів, що значно розширює можливості їх застосування.

Технологія Always Encrypted із захищеними анклавами доступна у Windows Server 2019.

#### 8. Azure Data Studio

Azure Data Studio (колишній SQL Operations Studio) – це спрощений кросплатформовий графічний засіб управління та редактор коду. Дозволяє створювати запити до реляційних та нереляційних баз даних за допомогою різних операційних систем та джерел даних. Azure Data Studio дозволяє підключатися до SQL Server локально та у хмарі, у Windows, macOS та Linux.

#### 9. Вибір ОС та контейнерів

SQL Server 2019 відрізняється гнучкістю щодо вибору платформи, мови програмування та засоби доставки.



Підтримка Red Hat Enterprise Linux, SUSE Linux Enterprise Server, Ubuntu та Windows.

Контейнери Docker для Linux та Windows. Установка із вбудованою підтримкою інструментів Linux: Yum Install, Apt-Get та Zypper.

Один і той же рівень абстракції з SQL Server на Linux.

Можливість використання R, Python та Java при роботі з T-SQL. Тепер розширення мови Java доступне для виконання коду Java в SQL Server.

## 10. Інтелектуальний аналіз даних

Розвиток SQL Server пішов шляхом інтеграції з іншими аналітичними платформами, зокрема Spark, яка включена тепер у постачання SQL Server.

Spark є дуже популярним інструментом для машинного навчання, для аналітики, має ефективну in memory машину. І все це інтегровано з SQL, який є дуже ефективним для візуалізації аналітики.

Правильний аналіз та ефективне представлення результатів безпосередньо впливає на ефективність аналізу даних та можливість приймати на їх основі управлінські рішення.

## 2.5 Photoshop

Photoshop - програмне забезпечення для обробки зображень та фотографій та ретушування, виготовлене компанією Adobe. Він використовується на комп'ютерах з ОС Windows або MacOS. Photoshop став стандартом для управління растровими зображеннями. Растрові зображення складаються з сітки точок, званих пікселями. Цікавість цих зображень полягає у відтворенні тонких градацій кольорів.

Його основними функціональними можливостями програми є:

- дизайн виділених елементів у різних режимах, що дозволяє з великою точністю вибирати частини зображення, що підлягають модифікації;
- безліч інструментів, що дозволяють обрізати зображення повністю, або частково, для редагування;

- створення різноманітних масок, що дозволяють виконувати монтажі між різними візуальними елементами;
- багато фільтрів, що дозволяють створювати спецефекти.

Це найбільш широко використовуваний програмний інструмент для редагування фотографій, маніпуляцій із зображеннями та ретуші для багатьох форматів файлів зображень та відео. Інструменти Photoshop дозволяють редагувати як окремі зображення, так і великі партії фотографій. Існує кілька версій Photoshop, включаючи Photoshop CC, Photoshop Elements та Photoshop Lightroom.

Adobe Photoshop є важливим інструментом для веб-розробників, графічних дизайнерів, фотографів та творчих професіоналів.

Також у Photoshop можна створювати макети веб-сайтів, а їх конструкції можна допрацьовувати до того, як розробники приймуть етап кодування. Окрему графіку можна створювати та експортувати для використання в інших програмах.

Photoshop CC – (Creative Cloud Photoshop) це вдосконалене програмне забезпечення для обробки зображень, яке використовується дизайнерами, веб-професіоналами, відеоредакторами, професіоналами у галузі комунікацій або рекламними агенціями, та фотографами для модифікації цифрових зображень або управління ними. Потужність його інструментів залишає повну свободу творчості вашій уяві та дозволяє вносити надзвичайно точні зміни до всіх видів знімків.

Існує кілька способів вивчити Adobe Photoshop. Найпопулярніші методи включають особисте відвідування уроків Photoshop, онлайн-уроків, навчання за допомогою онлайн-посібників Photoshop та книг. Заняття розроблені, щоб допомогти студентам скористатися як груповими навчальними заходами, так і індивідуальними інструктажами. Навчання в класі також допомагає учням долати виклики чи перешкоди за допомогою керованого навчання. Ці можливості навчання особливо корисні, коли випускаються нові функції чи інструменти.

Adobe Photoshop може зберігати або експортувати зображення та графіку у різноманітних форматах файлів.

Зображення можуть містити лише пікселі, тоді як інші можуть містити суміш пікселів та векторів. Деякі формати файлів також використовують методи стиснення для зменшення розміру зображення, а деякі варіанти стиснення навмисно видаляють дані, щоб зменшити розмір файлу. Для збереження всіх функціональних можливостей Adobe Photoshop, включаючи ефекти, маски та шари, використовується формат файлу Photoshop (PSD).

Розроблений спеціально для комп'ютерів, Photoshop доступний у 32 або 64 бітах залежно від архітектури вашої операційної системи. Він також є крос-платформним, оскільки доступний як у Windows, так і в MacOS. Інтернет-версія, доступна через будь-який веб-браузер, також існує. Це Photoshop Express.

Якщо Photoshop загалом отримує позитивні відгуки від своїх користувачів, це тому, що він має стільки переваг. Незважаючи на свій професіоналізм, він не для всіх. Більше того, це не ідеальне програмне забезпечення, хоча з кожним новим оновленням воно виправляє свої попередні недоліки.

Завдяки багатьом параметрам, які воно пропонує, програмне забезпечення Adobe Photoshop широко використовується в мультимедіа та Інтернеті. Найвідоміший, завдяки своїй здатності виправляти фізичні недоліки. Наприклад, на обличчі можна буде видалити невеликі плями або прищі за допомогою спеціальної щітки. Завдяки вдосконаленому алгоритму Photoshop результат є цілком природним.

У професійному середовищі це інструмент, який найчастіше використовується для моделювання флаєрів, логотипів чи візиток. На веб-сайті це може бути джерелом багатьох присутніх графічних елементів, що дозволяє надати сайту індивідуальність. Неможливо процитувати всі можливі варіанти використання Photoshop, оскільки вони нескінченні.

## 2.6 Auth0

Що таке Auth0?

Auth0 - це аутентифікація як службовий інструмент, який спрощує реалізацію функцій, пов'язаних із автентифікацією на вашому сайті. Якщо ви створили програму і хочете просто передати функції автентифікації та авторизації, такий сервіс, як Auth0, ви повинні розглянути.

Auth0 може запропонувати:

- єдиний вхід до системи
- багатофакторна автентифікація
- вхід до системи без пароля
- керування користувачами

І багато іншого.

У ній реалізований універсальний вхід для всіх ваших програм, що підвищує зручність користувачів. Це як SSO між кількома програмами.

Таким чином, користувач не дратуватиметься від необхідності окремого входу в інші додатки; натомість він буде автоматично підписаний через SSO.

Існує центральна панель управління функціями для соціальних логінів, виявлення проблем та MFA.

Можна також використовувати домени для безшовного брендингу. Більше того, є магічні посилання без пароля для зручнішої роботи користувачів.

Auth0 має гнучкі підписки для багатьох випадків використання, включаючи безкоштовний пакет.

## 2.7 Angular Material

Material Design (Матеріальний дизайн) — зовнішній вигляд програмного забезпечення та програм операційної системи Android від компанії Google.

У процесі активного використання Angular Material розробники бачили

багато однотипних варіантів використання та завдань із кастомізації готових компонентів цієї бібліотеки:

- Створення та кастомізація спливаючої панелі (меню, список, автозаповнення, підказка) для довільного компонента.
- Angular
- Використання клавіатури для навігації всередині меню, списку тощо.
- Перехоплення фокусу всередині секції однієї сторінки (діалогове вікно, бічна панель)
- Динамічна зміна відображення контенту натисканням вкладок або степером.

Все частіше стикаючись з такими завданнями, розробники Angular поступово почали створювати та повторно використовувати абстрактніші типові рішення. До 2017 року з'явилася бібліотека Angular Component Dev Kit (CDK).

Вперше його було представлено публіці на конференції Google I/O у 2014 році. Ідея полягає у додатках, які відкриваються та згортаються як картки, використовуючи ефекти тіней. Тобто в Google зібралися повністю позбутися гострих кутів.

Такий інтерфейс має операційна система iOS на iPhone. Думаю, що багато хто звернув увагу на заокруглені краї додатків на яблучному девайсі. За ідеєю дизайнерів Google, у додатків всі краї мають бути заокругленими, а відкриватися вони будуть як картки, перемикаючись між собою плавно і практично непомітно для ока.

Думаю, що в сучасному світі всім стало ясно: мало зробити гарно, потрібна ще функціональність. Тому, звичайно, будь-який дизайн має на увазі свої особливості. При розробці Google мали дві основні цілі. Перша – всі продукти на базі операційної системи Android мають бути уніфіковані. Друга — величезна кількість програм у Google Play повинні сприйматися користувачами не відокремленими одиницями, а продовження однієї системи.

## 2.8 Entity Framework

Entity Framework – це набір технологій у ADO.NET, які підтримують розробку програмних програм, орієнтованих на дані. Архітекторам та розробникам додатків, орієнтованих на обробку даних, доводиться враховувати необхідність досягнення двох абсолютно різних цілей. Вони повинні моделювати сутності, зв'язки та логіку вирішуваних бізнес-завдань, а також працювати з ядрами СУБД, що використовуються для збереження та отримання даних. Дані можуть розподілятися за декількома системами зберігання даних, у кожній з яких застосовуються свої протоколи, але навіть у додатках, що працюють з однією системою зберігання даних, необхідно підтримувати баланс між вимогами системи зберігання даних та вимогами написання ефективного та зручного для обслуговування коду програми.

Платформа Entity Framework дозволяє працювати з даними у формі специфічних для домену об'єктів та властивостей (наприклад, з клієнтами та їх адресами) без необхідності враховувати формат базових таблиць та стовпців бази даних, де ці дані зберігаються. Entity Framework дає розробникам можливість працювати з даними на вищому рівні абстракції, створювати та супроводжувати програми, орієнтовані на роботу з даними, одночасно з цим скорочуючи обсяг коду порівняно з традиційними програмами. Оскільки Entity Framework є компонентом платформи .NET Framework, Entity Framework програми можуть працювати на будь-якому комп'ютері, на якому встановлена платформа .NET Framework з пакетом оновлень 1 (SP1) версії 3,5.4f.

### Надання життєвого циклу моделям

Багаторічним і загальним підходом до розробки є підхід, при якому побудова програми або служби є його поділом на три частини: модель домену, логічну модель та фізичну модель. Модель домену визначає сутності та зв'язки в моделі, що моделюється. Логічна модель для реляційної бази даних

забезпечує нормалізацію сутностей та зв'язків з метою створення таблиць з обмеженнями зовнішнього ключа. У фізичній моделі враховуються можливості конкретної системи обробки даних шляхом визначення залежать від ядра бази даних докладних відомостей про зберігання даних, що стосуються секціонування та індексування.

Фізична модель удосконалюється адміністраторами бази даних з метою підвищення продуктивності, але програмісти, які розробляють код програми, переважно змушені обмежуватися роботою з логічною моделлю, готуючи SQL-запити і викликаючи процедури, що зберігаються. Моделі домену в основному використовуються як інструмент для представлення та обміну думками про вимоги до додатку, тому найчастіше служать як схем, що практично не змінюються, які розглядаються і обговорюються на ранніх стадіях проекту, після чого виходять зі сфери уваги. У багатьох колективах розробників прийнято пропускати етап створення концептуальної моделі та починати з визначення таблиць, стовпців та ключів у реляційній базі даних.

Entity Framework дає життя моделі, дозволяючи розробникам вимагати сутності та зв'язку в моделі предметної області (названої концептуальною моделлю в Entity Framework), водночас покладатися на Entity Framework перетворення цих операцій на команди, які стосуються джерела даних. Це дозволяє відмовитись від застосування у додатках жорстко заданих залежностей від конкретного джерела даних.

При роботі в режимі Code First концептуальна модель зіставлена з режимом зберігання коду. Entity Framework може обчислювати концептуальну модель на основі типів об'єктів та додаткових конфігурацій, які ви визначаєте. Метадані зіставлення формуються під час виконання на основі поєднання визначень типів домену та додаткової інформації про конфігурацію, зазначену в коді. Entity Framework створює базу даних при необхідності на основі метаданих. Для отримання додаткових відомостей див. Створення моделі.

При роботі із засобами роботи з моделлю EDM концептуальна модель, модель зберігання та зіставлення між ними виражені у схемах на основі XML

та визначені у файлах з іменами з відповідними розширеннями.

Мова CSDL визначає концептуальну модель. Мова CSDL – це реалізація EDM Entity Framework. Розширення файлу – CSDL.

Мова SSDL визначає модель зберігання даних, яка також називається логічною моделлю. Розширення файлу – SSDL.

Мова MSL визначає зіставлення моделі зберігання та концептуальної моделі. Розширення файлу – MSL.

Модель зберігання та зіставлення при необхідності можуть бути змінені без зміни концептуальної моделі, класів даних та коду програми. Моделі зберігання залежить від постачальника, тому можна працювати з узгодженою концептуальною моделлю через різні джерела даних.

Entity Framework використовує ці файли моделі та зіставлення для створення, читання, оновлення та видалення операцій з сутностями та зв'язками в концептуальній моделі в еквівалентних операціях у джерелі даних. Entity Framework навіть підтримує зіставлення сутностей у концептуальній моделі зі збереженими процедурами у джерелі даних. Додаткові відомості див. у статті специфікації на мові CSDL, SSDL та MSL.



### 3. ЗАГАЛЬНІ ВІДОМОСТІ ДОДАТКУ

#### 3.1 Розробка концепції

Аналізуючи актуальність додатку, для найефективнішого розповсюдження, та використання цільової аудиторії було прийняте рішення його розробки у вигляді веб-додатку.

Цільова аудиторія – це рекрутмент, якому потрібно автоматизувати свою роботу у вигляді веб-додатку. Також цільовою аудиторію можуть бути компанії, які хочуть напряду стежети не тільки за кандидатами а й за роботою самих рекрутерів, та актуальністю інформації у вакансіях та кандидатиів.

Головна ідея цього додатку також полягає в тому щоб забезпечити рекрутменту ідеальні умови для праці, а саме: легкість у використанні, сучасність дизайну та респонсів дизайн, стабільна робота програми без багів.

Короткий опис застосування – цей додаток націлений на відстеження повного процесу рекрутменту починаючи від спілкування у режимі телефонної розмови та закінчуючи підписанням договору від компанії.

Головні вимоги додатку:

- стабільність додатку
- респонсив дизайн
- легкість у використанні
- відстеження процесу рекрутменту
- вхід на сайт
- неможливість перегляду контету для людей у який немає доступу
- можливість вибору мови ENG та UKR
- швидкість завантаження контенту
- виведення всіх нещодавно доданих рекрутерів, вакансій та кандидатів на головну сторінку

Унікальність додатку:

- можливість логіну компанії тільки через пошту з корпоративним доменом у назві пошти (наприклад: employee@devoxsoftware.com)
- реалізація пагінації на бекенд частині сайту
- швидкодія сайту
- простота використання

### 3.2 Проектування структури та створення додатку

Для того щоб написати бекенд частину сайту, спочатку потрібно вибрати архітектуру проекту. Я вирішив зупинитися на Clean Architecture з використанням CQRS паттерну, та Mediator паттерну для реалізації додатку, з залишенням можливості додати ще одну базу даних для покращення швидкості при необхідності. Вибрав я Clean Architecture (рис. 3.1) тому, що ця архітектура об'єднала у собі ідеї кількох інших архітектурних підходів, які сходяться на тому, що архітектура повинна:

- бути тестованою;
- не залежати від UI;
- не залежати від БД, зовнішніх фреймворків та бібліотек.

Це досягається поділом на шари та слідуванням Dependency Rule (правила залежностей).

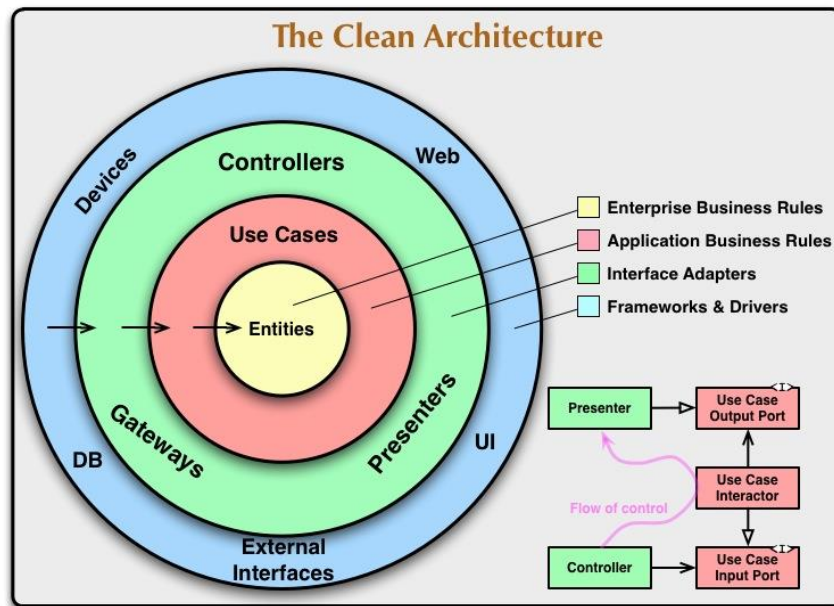


Рисунок 3.1. – Архітектура Clean Architecture

Після проектування структури бекенд частини, я створив всю бізнес логіку для наступних об'єктів: Candidates, Catalogs, Interviews, Recruiters, Vacancies. Також було додано глобальний middleware для логування помилок за допомогою сервісу Loggly.

Проаналізувавши головні вимоги додатку, а саме стабільність роботи додатку, було вирішено зайнятися покриванням бекенд частини Unit тестами для забезпечення правильної роботи додатку, та виявленням додаткових багів у роботі системи. Після написання тестів було виявлено декілька значних помилок при написанні системи але я вирішив їх. Всього було написано 241 тестів для усієї бізнес логіки системи (рис. 3.2):

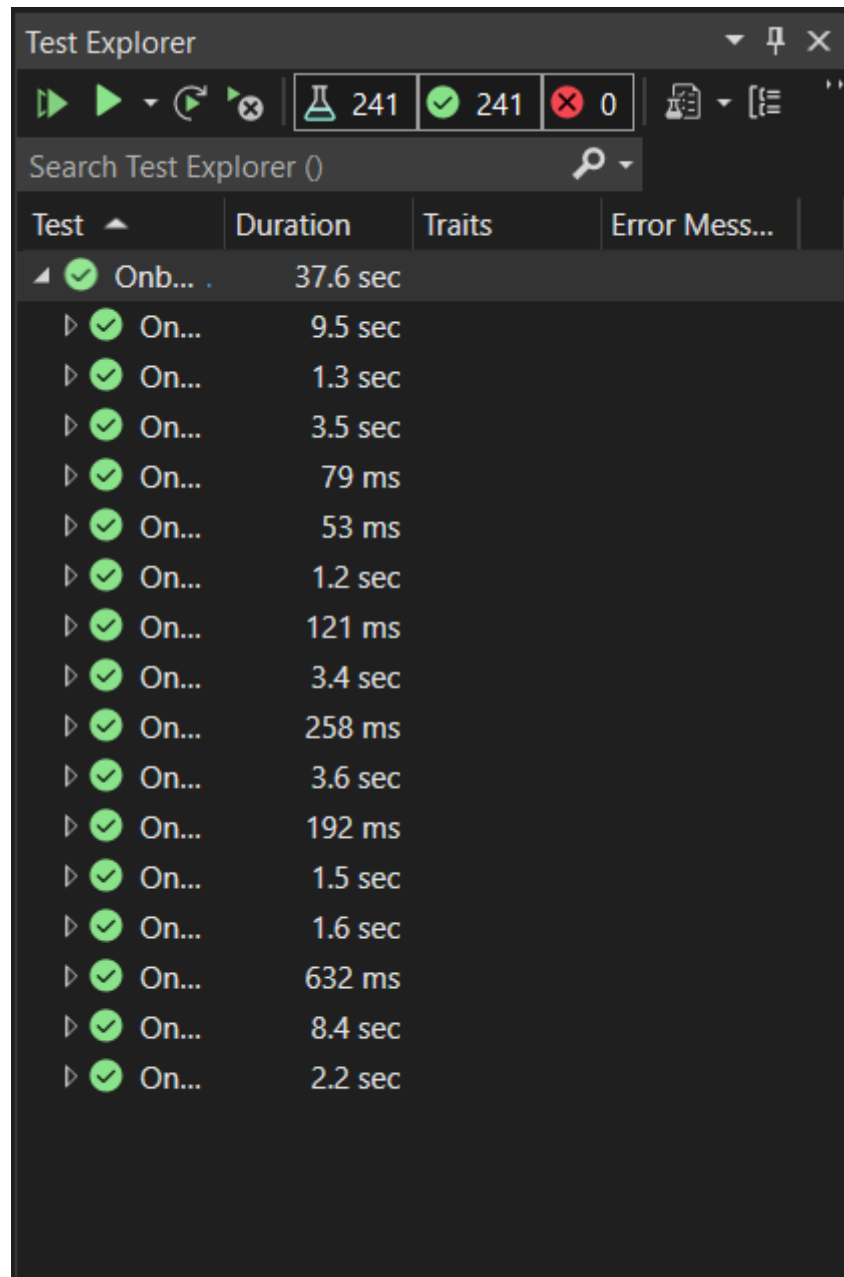


Рисунок 3.2. – Тестування Unit Tests

Для написання Unit тестів було вирішено використовувати бібліотеку Moq для так званого “мокування” даних і деяких реалізацій, та бібліотеки Shouldy, Xunit для перевірки заданих умов проходження Unit тесту. Також було вирішено додати до додатку Swagger щоб протестувати усю бізнес логіку своїми руками (рис. 3.3).

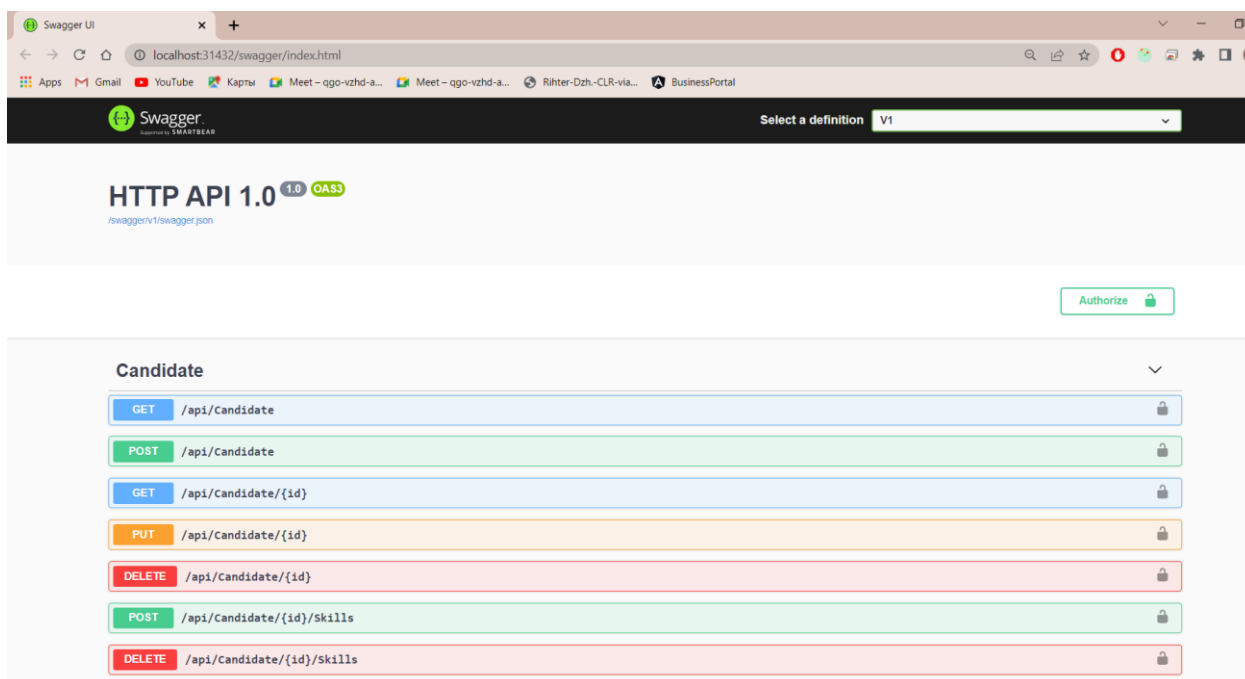


Рисунок 3.3. – Swagger

Після написання бекенд частини було вирішено почати створення Frontend частини сайту. Почав я роботу з верстки головної сторінки, для людей які ввійшли без доступу (рис 3.4).

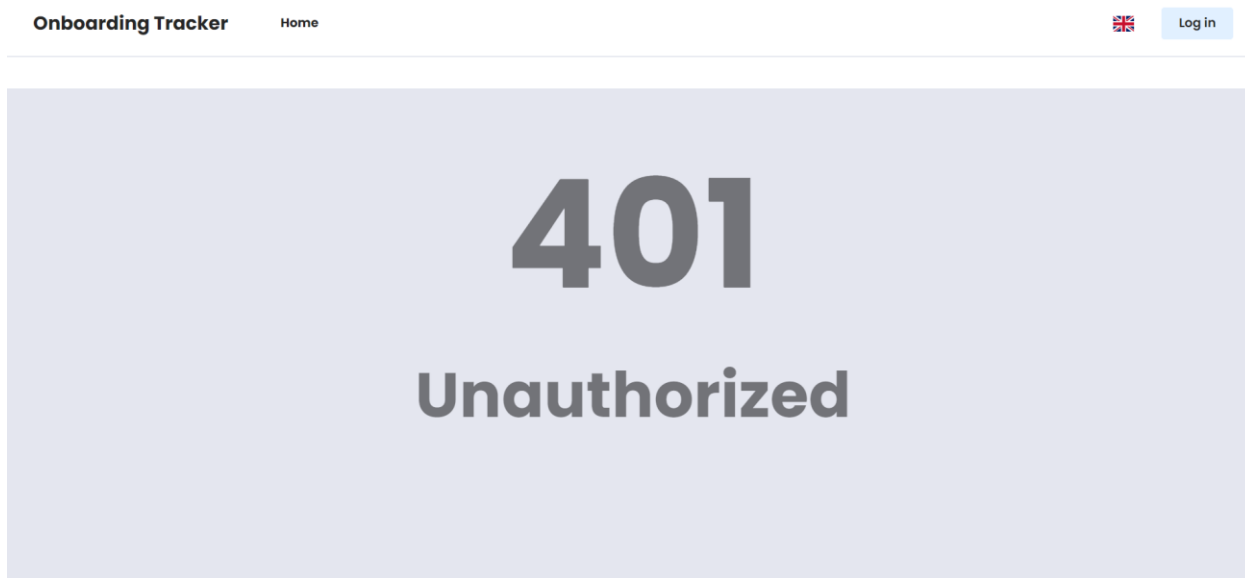


Рисунок 3.4. – Помилка 401

Далі реалізував вхід на сайт за допомогою Auth0 (рис. 3.5):

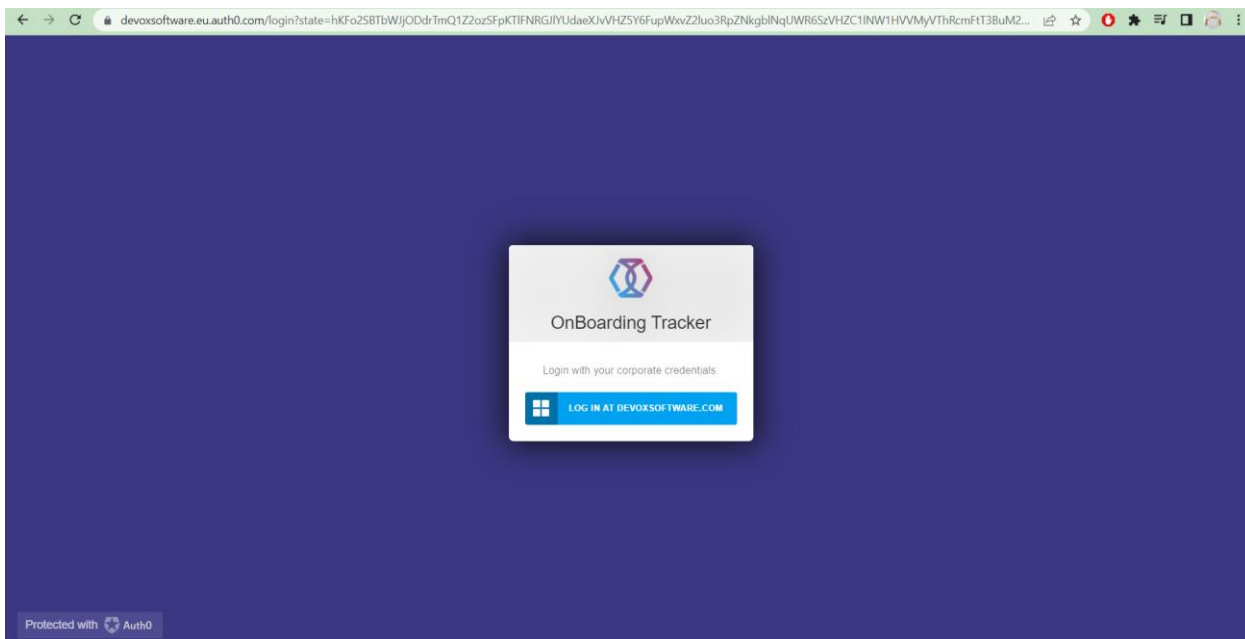


Рисунок 3.5. – Вхід Auth0

Також в Auth0 при вході в аккаунт є інтеграція з Microsoft (рис.3.6).

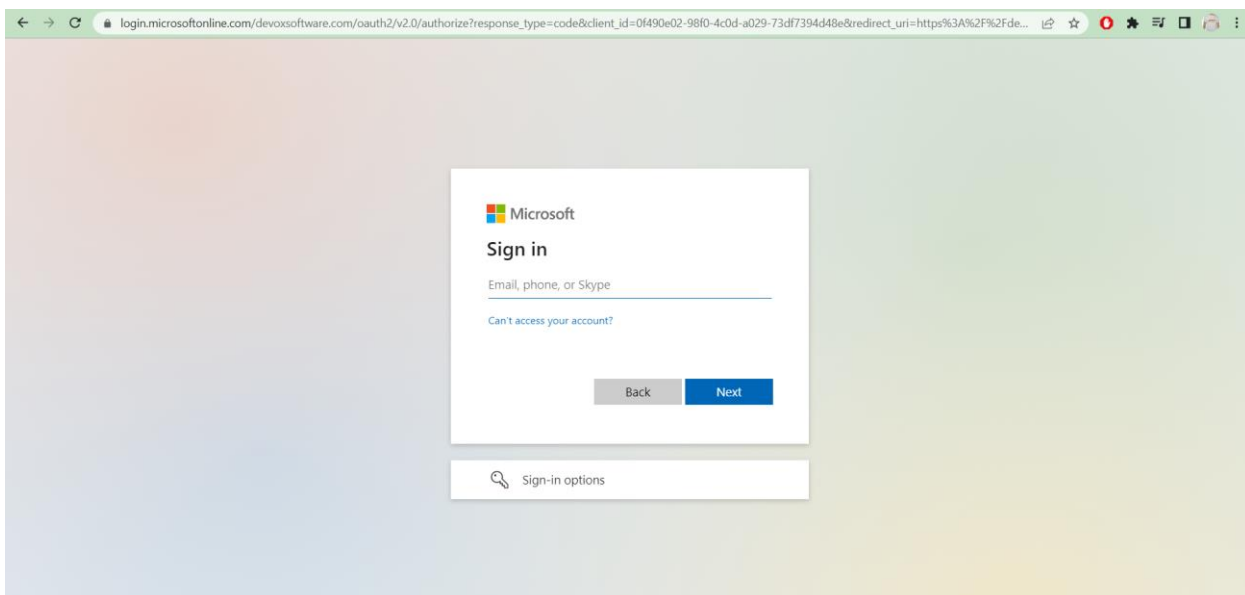


Рисунок 3.6. – Інтеграція з Microsoft

Після входу за допомогою Auth0 було виведено дані користувача, такі як фото, ім'я та фамілія (рис. 3.7).

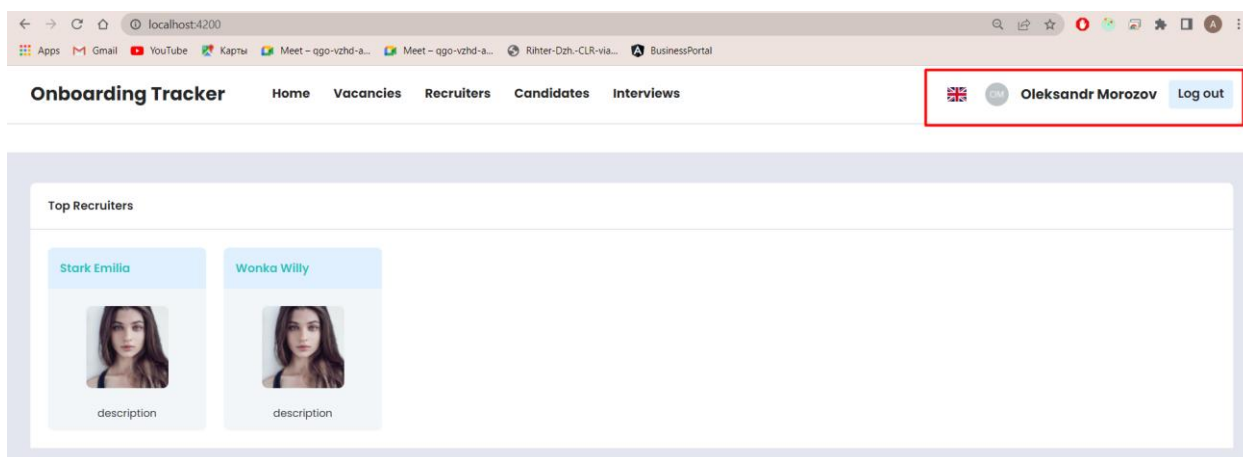


Рисунок 3.7. – Відображення даних користувача

Після реалізації входу реалізував логіку виведення щойно добавлених об'єктів на головну сторінку сайту (рис. 3.8).

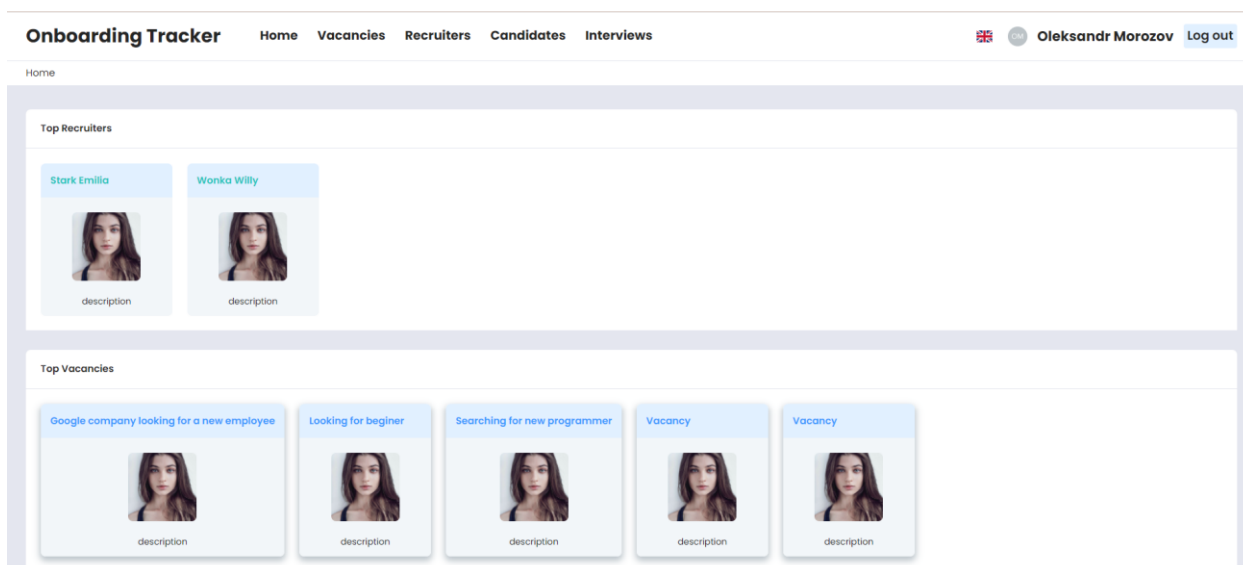


Рисунок 3.8. – Головна сторінка

Далі почав реалізацію роботи з вакансіями, а саме додавання нової вакансії (рис. 3.9), виправлення вже існуючої вакансії (рис. 3.10), видалення вакансії (рис. 3.11).

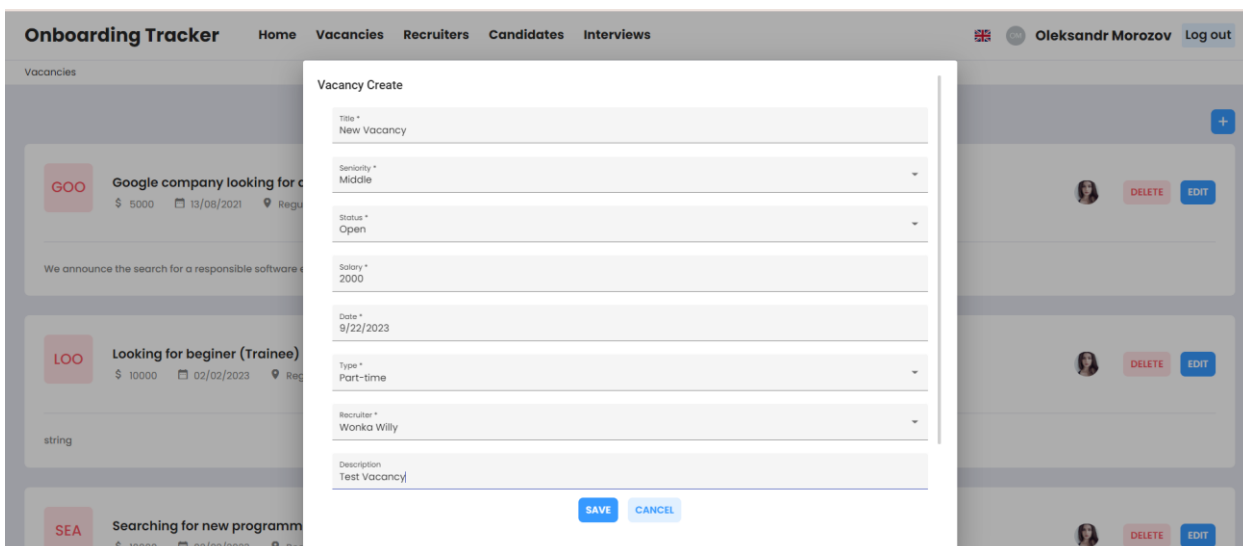


Рисунок 3.9. – Додавання нової вакансії

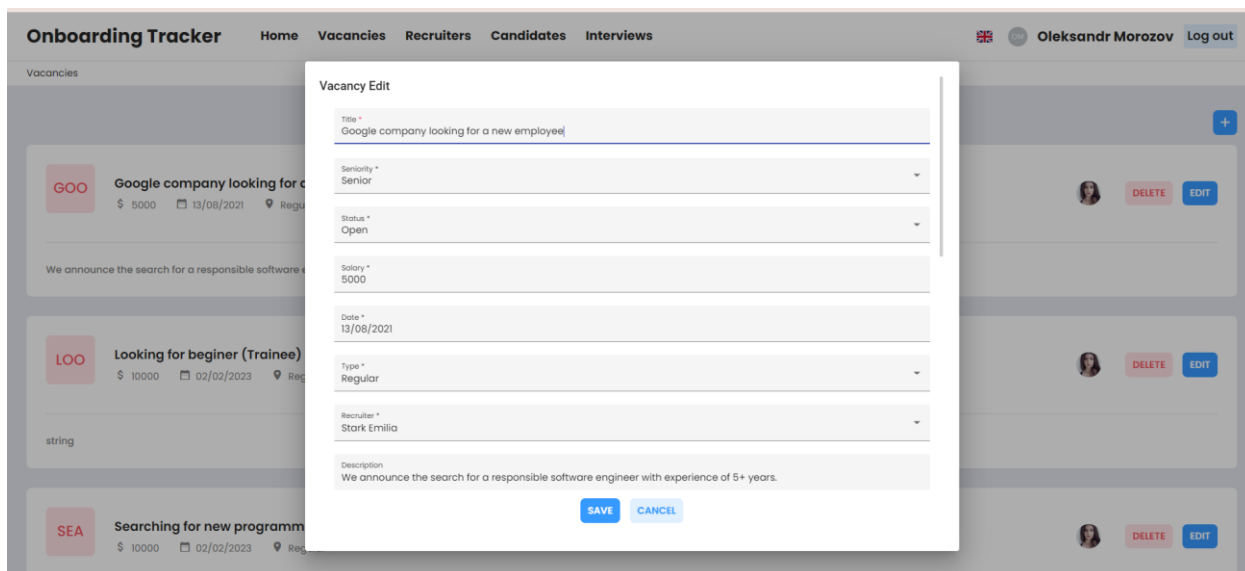


Рисунок 3.10. – Виправлення вже існуючої вакансії



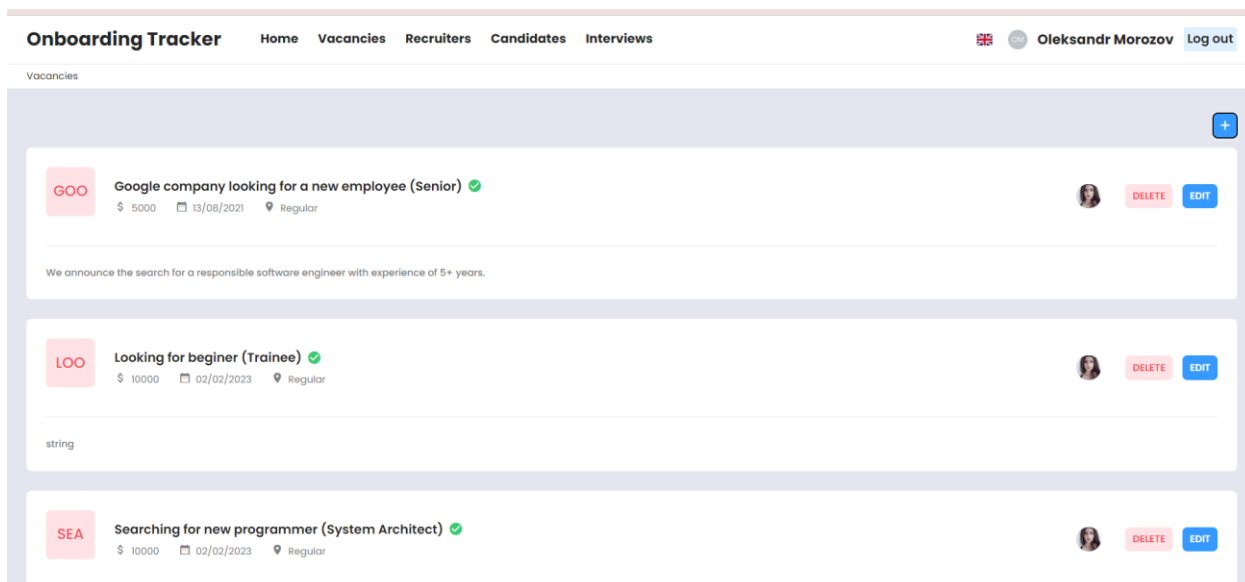


Рисунок 3.11. – Видалення вакансії

Далі за таким само принципом я реалізував рекрутерів (рис. 3.12), та кандидатів (рис. 3.13).

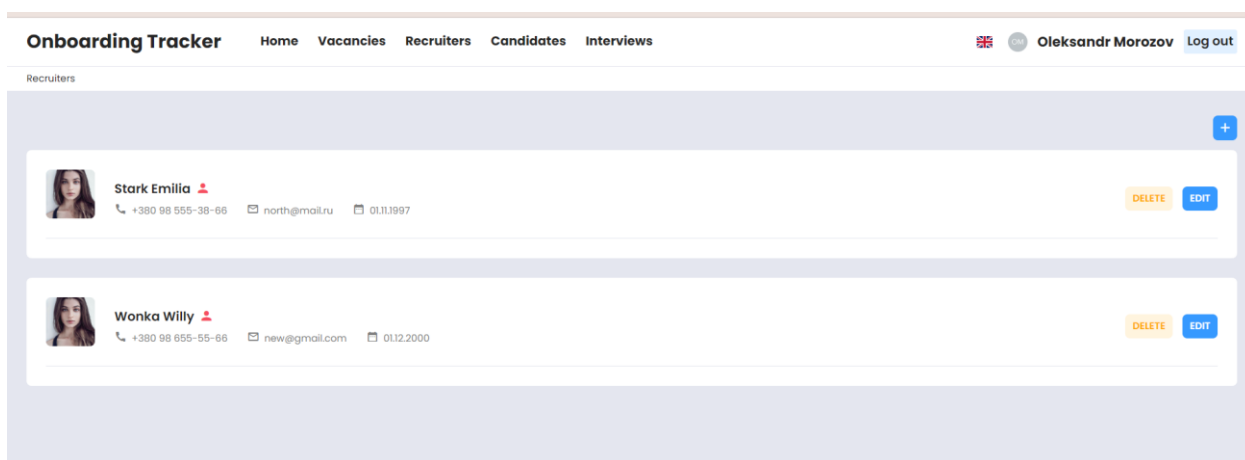


Рисунок 3.12. – Відображення рекрутерів

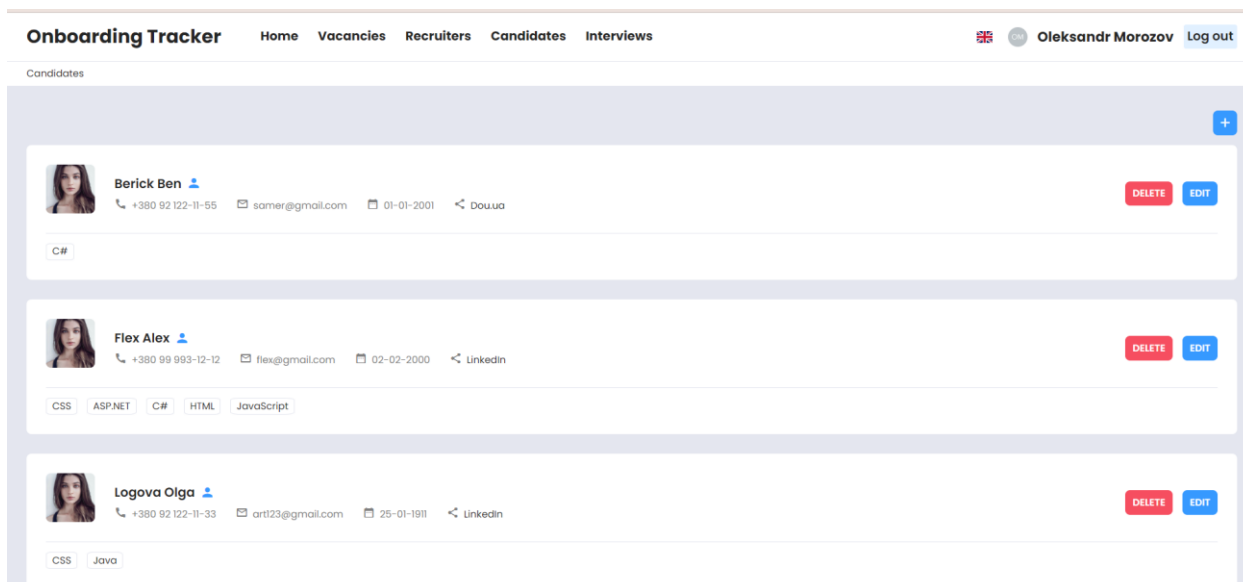


Рисунок 3.13. – Відображення кандидатів

Для реалізації секції інтерв'ю було вирішено зробити таблицю (рис. 3.14), для сучасності дизайну, та спрощення знаходження потрібного інтерв'ю.

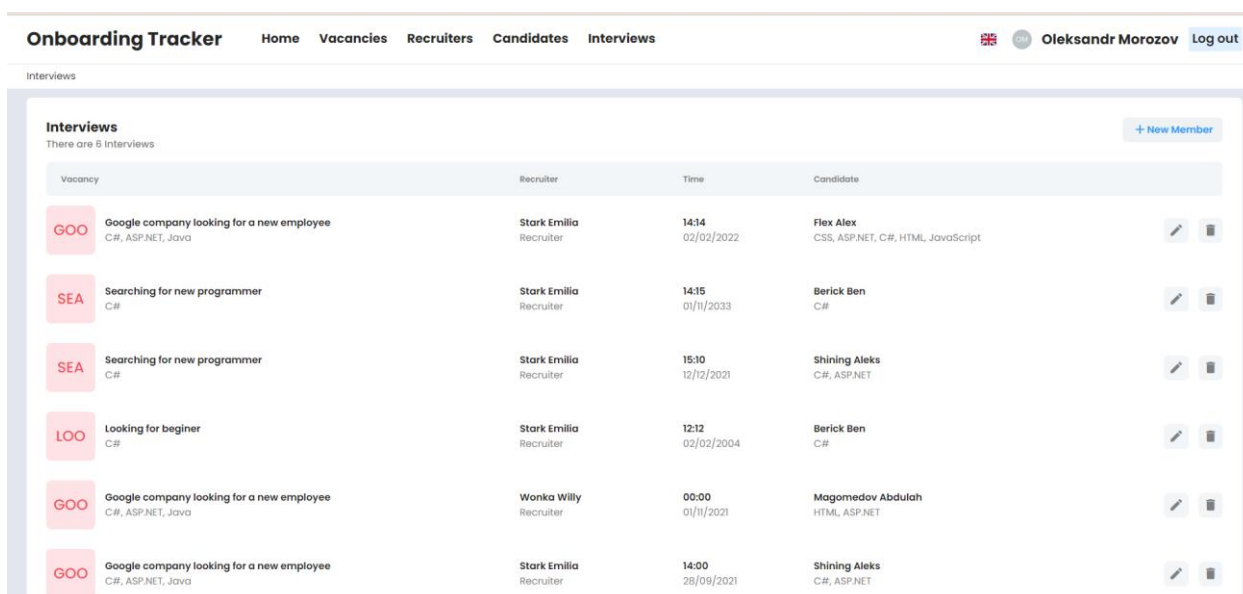


Рисунок 3.14. – Таблиця інтерв'ю

Також було реалізовано дві мови на сайті, а саме українська (рис. 3.15), англійська (рис. 3.16).

### Рисунок 3.15. – Українська мова інтерфейсу

### Рисунок 3.16. – Англійська мова інтерфейсу

## ВИСНОВКИ

У результаті виконання даної дипломної роботи було розроблено систему відстеження кандидатів. У роботі описується шлях створення системи від ідеї, її аналізу, до її втілення у життя.

1. Під час роботи над дипломним проектом, було досліджено рекрутинг процеси, проаналізовано актуальність цієї теми, та визначено її наукову новизну. Було зроблене рішення створити саме веб додаток, через його комфортного застосування, найкращу відповідність вимогам мого додатку й найбільшим потенціалом успіху в майбутньому. Для покращення розуміння можливостей та функцій, які реалізуватиме мій додаток, був проведений аналіз конкурентів.

2. Було проведено аналіз інструментів та програмних засобів реалізації, які найліпше впораються зі створенням графічного інтерфейсу та усього функціоналу додатку. У ході аналізу було обрано використання C# як мову програмування для backend, Figma для створення прототипу, Microsoft Visual Studio 2019 як середовища розробки для backend, Visual Studio Code як середовище розробки для frontend, Java Script як мову програмування для frontend, Photoshop для створення елементів графічного дизайну, MSSQL як основну базу даних, та GIT для контролю версій.

3. Визначення концепції, вимог та головних елементів інтерфейсу веб-додатку. Далі був розроблений детальний прототип, який дозволяє переглянути увесь функціонал та графічний інтерфейс.

4. Розробка графічного інтерфейсу за допомогою Photoshop, підключення проекту до GIT, зв'язок з MSSQL, та створення там бази даних, написання логіки додатку за допомогою мови програмування C# у середовищі розробки Microsoft Visual Studio 2019 з використанням Entity Framework для backend частини, та написання frontend частини за допомогою мови програмування Html/CSS та JavaScript у середовищі розробки Microsoft Visual Studio Code.

Розроблений додаток має сприяти на розвиток рекрутингу в Україні, відстеження кандидатів, інформування про інтерв'ю з кандидатами.

Створений веб-додаток має значні перспективи подальших досліджень. Можна буде доповнити функціонал, додавши можливість інтегрування кандидатів з різних платформ, рейтинг кандидата, можливість завантажувати файли з excel таблиці.

## СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Andrew Troelsen, Phil Japikse, «Pro C# 7: With .NET and .NET Core /- Apress» – с. 403 – 420.
2. Adam Freeman, «Pro Angular 9: Build Powerful and Dynamic Web Apps» – с.101-104.
3. Jeffrey Richter, «CLR via C#. Programming on the Microsoft .NET Framework 4.5 on C# language. 4th ed» – с.1-28.
4. Jonathan Kehayias, Kalen Delaey, «Microsoft SQL Server 2012 Internals 1st Edition» – с.55-84.
5. Kyle Simpson, «You Don't Know JS: Scope & Closures 1st Edition» - с. 220 – 240.
6. Jon Duckett, «HTML and CSS: Design and Build Websites 1st Edition» – с.110-143.
7. Liberti D. «Програмування на C#» – с.26-266.
8. Konvac James, «C#.NET History» – с.50-95.
9. Herbert Schildt, «Повний довідник по C#» – с. 200-300.
10. Mads Torgensen, «New Features in C#» – с.110-152.
11. Christian Nagel, Carly Watson, «Beginning Microsoft Visual Studio C# 2010» – с.200-400.
12. Andrew Troelsen, «Мова програмування C# 5.0 та платформа .Net 4.5» – с. 15-74.
13. Joseph Albahari, «C# 6.0» – с. 10-52.

## ДОДАТОК А



ДЕРЖАВНИЙ УНІВЕРСИТЕТ ТЕЛЕКОМУНІКАЦІЙ  
НАВЧАЛЬНО-НАУКОВИЙ ІНСТИТУТ ІНФОРМАЦІЙНИХ  
ТЕХНОЛОГІЙ  
КАФЕДРА ІНЖЕНЕРІЇ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ



Розробка системи «OnBoardingTracker» для відстеження  
кандидатів мовою C# та Java Script

Виконав студент 4 курсу  
групи ПД-42  
Морозов Олександр Олегович  
Керівник роботи  
Дібрівний Олесь Андрійович

Київ – 2022

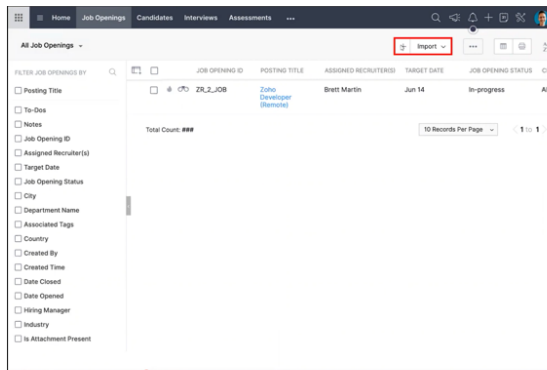
### МЕТА, ОБ'ЄКТ ТА ПРЕДМЕТ ДОСЛІДЖЕННЯ

**Мета роботи** - покращення процесу відстежування кандидатів у процесі рекрутменту за допомогою створеного додатку.

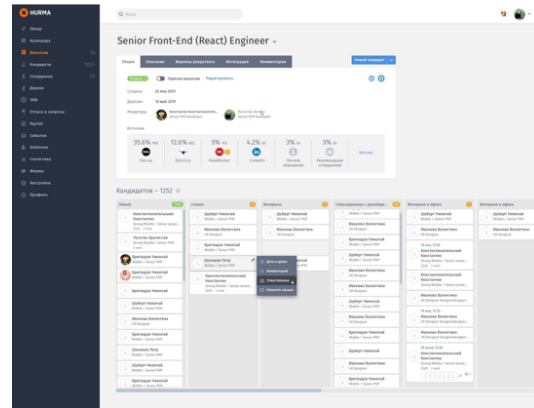
**Об'єкт дослідження** - процес відстеження кандидатів.

**Предмет дослідження** - сучасні технології в процесі відстеження та знаходження кандидатів.

# НАЙБІЛЬШ ПОДІБНІ АНАЛОГИ





Додаток Zoho Recruiter



Додаток Hurma System

3

# ПОРІВНЯННЯ АНАЛОГІВ

Назва	Переваги	Недоліки
Zoho Recruiter 	Стабільна робота сервісу	Застарілий дизайн
	Двофакторна автентифікація	Високий поріг входу
Hurma System 	Можливість резервного копіювання	Відсутність резервного копіювання
	Двофакторна автентифікація	Високий поріг входу
		Застарілий дизайн
		Відсутність авторизації через корпоративну пошту

4



## ТЕХНІЧНЕ ЗАВДАННЯ

- Розробити структуру додатку.
- Реалізувати можливість відображення кандидатів та рекрутерів.
- Реалізувати можливість змінити мову.
- Реалізувати можливість для користувача продивляти дані кандидатів.
- Реалізувати інтграцію з Microsoft при авторизації.
- Реалізувати можливість авторизації тільки через корпоративну пошту.

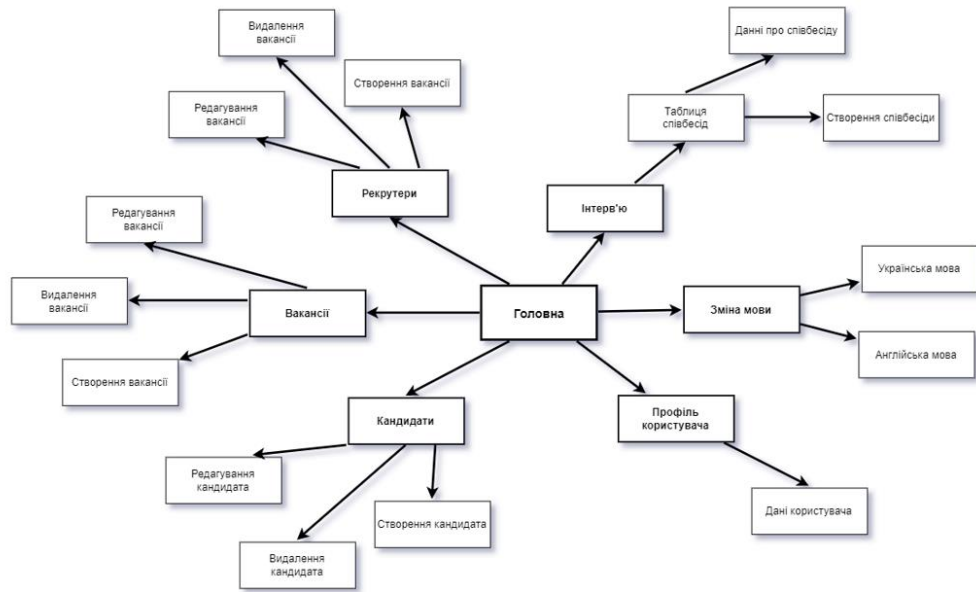
5

## ПРОГРАМНІ ЗАСОБИ РЕАЛІЗАЦІЇ



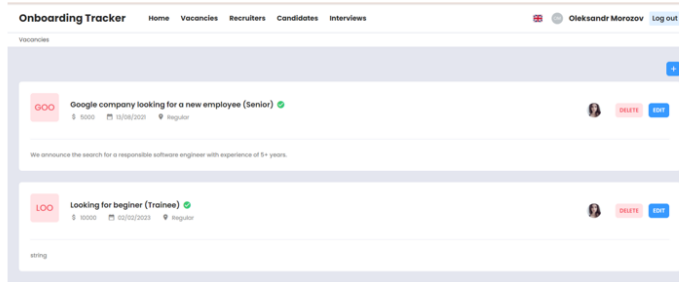
6

# СТРУКТУРА ДОДАТКУ



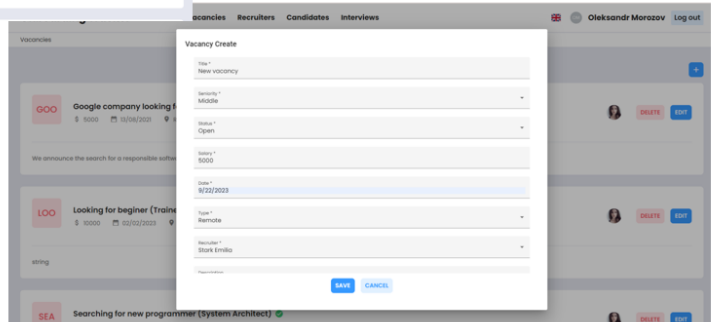
7

# ЕКРАННІ ФОРМИ



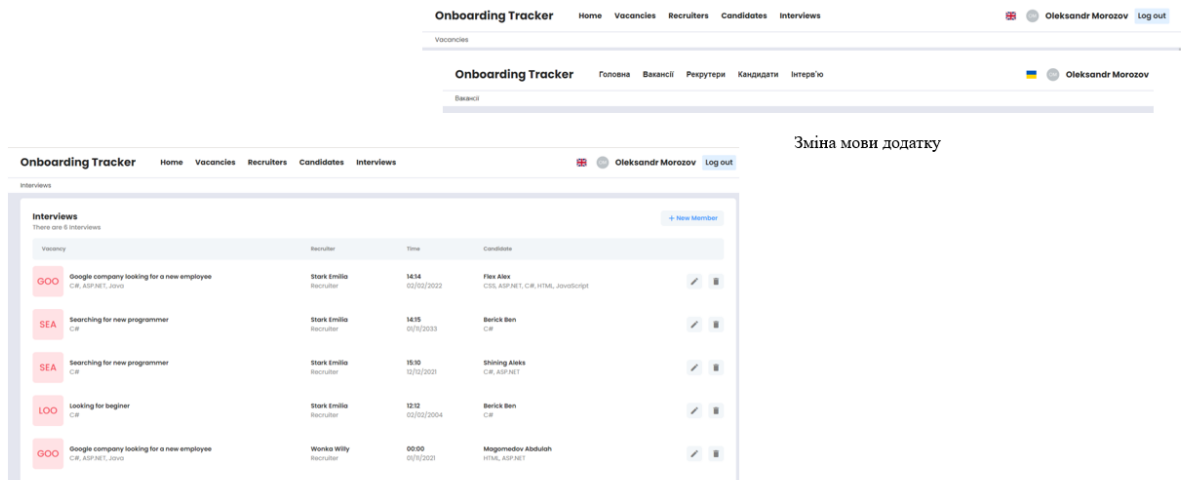
Представлення вакансій

Форма створення вакансії



8

# ЕКРАННІ ФОРМИ



Таблиця інтерв'ю

9

## АПРОБАЦІЯ РЕЗУЛЬТАТІВ ДОСЛІДЖЕННЯ

Морозов О.О., Дібрівний О.А. Актуальність створення системи для відстеження кандидатів / Міжнародна науково-технічна конференція студентства та молоді «Застосування програмного забезпечення в ІКТ». Збірник тез 20.04.2022, ДУТ, м. Київ – К.: ДУТ, 2022.

10

## ВИСНОВКИ

1. Обґрунтовано актуальність та новизну розробленого додатку.
2. Проведено аналіз предметної галузі.
3. Розроблено алгоритми роботи системи.
4. Проведено тестування.

Перспективи подальшого розвитку:

1. Додати можливість інтегрування кандидатів з різних платформ.
2. Додати рейтинг кандидата.
3. Додати можливість завантажувати файли з excel таблиці.

**ДЯКУЮ ЗА УВАГУ!**