

**КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ ІМЕНІ ТАРАСА
ШЕВЧЕНКА**

Факультет комп'ютерних наук та кібернетики
Кафедра теорії та технології програмування

Кваліфікаційна робота

на здобуття ступеня бакалавра

за спеціальністю 122 “Комп'ютерні науки”

на тему:

**СЕРВІС ДЛЯ ЗБОРУ ДАНИХ ПРО ПРОФЕСІЙНУ ДІЯЛЬНІСТЬ
ВИПУСКНИКІВ НА ОСНОВІ СОЦІАЛЬНОЇ МЕРЕЖІ “LinkedIn”**

Виконав студент 4-го курсу

Максим МАРЧИШАК

Науковий керівник:

Доцент,

Тарас ПАНЧЕНКО



(підпис)



(підпис)

Засвідчую, що в цій роботі немає запозичень з
праць інших авторів без відповідних
посилань.

Студент



(підпис)

Роботу розглянуто й допущено до захисту
на засіданні кафедри теорії та технології
програмування

« ____ » _____ 2023 р.,

протокол № ____

Завідувач кафедри

Нікітченко Микола Степанович _____
(підпис)

Київ - 2023

РЕФЕРАТ

Обсяг роботи 44 сторінок, 13 ілюстрацій, 8 таблиць, 12 джерел посилань, 1 додаток.

БАЗА ДАНИХ, ТЕЛЕГРАМ-БОТ, ЗБЕРЕЖЕННЯ ІНФОРМАЦІЇ, ІНФОРМАЦІЙНА СИСТЕМА, ТЕХНОЛОГІЇ ПРОГРАМУВАННЯ, ЗБІР ДАНИХ ВИПУСКНИКІВ.

Об'єктом даної роботи є дані щодо працевлаштування випускників ЗВО. Предметом роботи є збір, розробка та аналіз даних про працевлаштування випускників ЗВО.

Метою роботи є розробка та створення інструменту, який дозволить збирати дані профілів студентів та випускників ЗВО. Інструмент спрямований на збір інформації, такої як: основні академічні досягнення, професійний розвиток, стажування та роботу після закінчення навчання.

Методи розроблення: комп'ютерне моделювання, проектування бази даних, розробка програмного продукту на основі моделі. Інструменти розроблення: інтегроване середовище розробки Visual Studio Code, мови програмування Golang, Python, JavaScript, СКБД MongoDB.

Результати роботи: розглянуто наявні застосунки призначені для отримання даних для аналітики та статистики освіти та кар'єри, випускників, розроблено інтерфейс користувача для бота, спроектовано та реалізовано програму для збирання, пошуку та обробки даних студентів.

ЗМІСТ

| | |
|---|-----------|
| ЗМІСТ..... | 3 |
| СКОРОЧЕННЯ ТА УМОВНІ ПОЗНАЧЕННЯ..... | 4 |
| ВСТУП..... | 5 |
| РОЗДІЛ 1. ОСНОВНІ ПОНЯТТЯ ТА ПІДХОДИ ДО ЗБОРУ І АНАЛІТИКИ ДАНИХ СТУДЕНТІВ ЗВО..... | 9 |
| 1.1 Огляд наявних інструментів на ринку..... | 9 |
| 1.1.1 LinkedIn..... | 9 |
| 1.1.2 DataFox..... | 11 |
| 2.2 Опис запропонованого підходу..... | 13 |
| РОЗДІЛ 2. ОПИС ВИКОРИСТАНИХ ТЕХНОЛОГІЙ..... | 16 |
| 2.1 Середовище програмування..... | 16 |
| 2.2 Сервіс для збору даних з профілів студентів..... | 16 |
| 2.2.1 Мова програмування JavaScript..... | 16 |
| 2.2.2 Бібліотека Selenium..... | 17 |
| 2.3 API сервіс..... | 18 |
| 2.3.1 Мова програмування Golang..... | 18 |
| 2.3.2 Фреймворк GIN..... | 19 |
| 2.4 Клієнтський сервіс..... | 19 |
| 2.4.1 Фреймворк Telebot..... | 19 |
| 2.5 База даних MongoDB..... | 20 |
| РОЗДІЛ 3. РЕАЛІЗАЦІЯ ІНСТРУМЕНТУ..... | 22 |
| 3.1 Реалізація бази даних..... | 22 |
| 3.2 Реалізація сервісу..... | 27 |
| 3.2.1 Сервіс Scraper..... | 28 |
| 3.2.2 API Сервіс..... | 29 |
| 3.2.3 Telegram bot client..... | 30 |
| 3.3 Інструкція використання..... | 33 |
| 3.3.1 Інструкція неавторизованого користувача..... | 33 |
| 3.3.2 Інструкція авторизованого користувача..... | 33 |
| 3.3.3 Інструкція адміністратора..... | 37 |
| 3.4 Розвиток інструменту..... | 39 |
| ВИСНОВОК..... | 40 |
| ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ..... | 42 |
| ДОДАТКИ..... | 44 |
| Додаток А. Діаграма бази даних..... | 44 |

СКОРОЧЕННЯ ТА УМОВНІ ПОЗНАЧЕННЯ

CRUD – Create Read Update Delete, основні функції керування даними;

API – Application programming interface, прикладний програмний інтерфейс;

IDE – Integrated Design Environment, інтегроване середовище розробки;

СКБД – система керування базами даних;

HTML – HyperText Markup Language, мова гіпертекстової розмітки;

DOM – Document Object Model, об'єктне представлення вмісту HTML-документу та інтерфейс для управління цим об'єктом;

CSS – Cascading Style Sheets, каскадні таблиці стилів;

ЗВО - заклад вищої освіти

ВСТУП

Оцінка сучасного стану об'єкта розробки. Набуття студентами практичних навичок, які будуть корисними в подальшій кар'єрі, є одним з основоположних завдань освіти. Для того, щоб забезпечити максимальну повноту та користь освітніх програм, корисно було б мати статистику працевлаштування та подальшого навчання студентів після отримання освітніх ступенів. Однак, збір такої статистики вручну освітніми працівниками є довгим та незручним способом. Маючи надійне джерело акумульованої інформації - соціальну мережу LinkedIn [1], яка висвітлює освітні та кар'єрні здобутки користувачів - можна створити ефективний інструмент швидкого отримання такої статистики, що не потребує зайвих ресурсів.

Актуальність роботи та підстави для її виконання. Предметна область, пов'язана зі збором та аналізом даних профілів студентів та випускників закладів вищої освіти, є актуальною і важливою з кількох причин.

По-перше, у сучасному світі, коли велика кількість інформації знаходиться в електронному вигляді, доступ до актуальних та точних даних стає все важливішим для ефективного управління університетом. Збір і аналіз даних профілів студентів та випускників може надати цінну інформацію про їхні освітні досягнення, досвід роботи, навички та інші релевантні деталі. Це дозволить університету зробити обґрунтовані рішення щодо покращення навчального процесу, розвитку програм навчання та кар'єрних послуг для студентів і випускників.

По-друге, зростання популярності соціальних мереж, зокрема LinkedIn, серед студентів і випускників створює потребу у зборі та аналізі даних з цих платформ. LinkedIn надає можливість створювати професійні

профілі, де можна детально вказати свої навички, освіту, досвід роботи та інші важливі аспекти. Але великий обсяг інформації та розподіленість профілів можуть ускладнювати процес збору та аналізу цих даних. Розробка програмного інструменту, який дозволить автоматизувати цей процес, є актуальною задачею.

Розв'язання проблем збору та аналізу даних з профілів студентів та випускників на LinkedIn може приносити значну користь університету та його спільноті. Збір точних та акуратно структурованих даних допоможе університету більш ефективно взаємодіяти зі студентами та випускниками, забезпечувати більш персоналізовану підтримку та розвиток, а також залучати студентів до актуальних проєктів і співпраці з бізнес-середовищем.

Мета й завдання роботи. Метою роботи є розробка та створення інструменту, який дозволить збирати дані профілів студентів та випускників ЗВО. Цей інструмент буде спрямований на збір інформації про студентів та випускників, включаючи основні академічні досягнення, професійний розвиток, стажування та роботу після закінчення навчання.

Застосування такого інструменту надасть можливість зібрати великий обсяг даних про студентське та випускницьке тіла закладів вищої освіти, що дасть можливість провести глибокий аналіз та отримати статистику, необхідну для різних досліджень та розробки політик у сфері освіти.

Для досягнення поставленої мети в рамках даної роботи необхідно виконати такі завдання:

- 1) дослідити наявні системи та інструменти, які можуть бути потенційними конкурентами;
- 2) дослідити методи отримання інформації з соціальної мережі LinkedIn;

- 3) розробити програмний інструмент для автоматизованого збору даних з профілів студентів та випускників з використанням методів скрапінгу та аналізу даних з веб сторінки, вивчених у попередньому завданні;

Ці завдання допоможуть створити програмний інструмент, який забезпечить збір та аналіз даних профілів на LinkedIn. Результатом роботи буде зручний та автоматизований інструмент, який дозволить отримувати актуальну інформацію про студентів та випускників для підтримки та вдосконалення процесу освіти.

Об'єкт, методи й засоби розроблення. Об'єктом даної роботи є збір та аналіз даних щодо працевлаштування випускників закладів вищої освіти (ЗВО), зокрема з факультетів та спеціальностей різних університетів. Предметом роботи є веб-свервіс, який дозволить зібрати інформацію про працевлаштування випускників, створити базу даних та провести аналіз цих даних для отримання корисних висновків та статистичних даних. Дані включатимуть інформацію про місце працевлаштування, сферу діяльності та інші фактори, які допоможуть визначити рівень працевлаштування випускників та знайти можливі шляхи для покращення цього процесу.

Можливі сфери застосування. Отримані результати та статистика, зібрані за допомогою даного інструменту, можуть послужити основою для подальшого вдосконалення програм навчання, розробки нових курсів та допомогти у прийнятті обґрунтованих рішень щодо розвитку університету та покращення якості освіти.

Зібрані дані профілів студентів та випускників можуть бути використані для багатьох цілей. Наприклад, вони можуть бути використані для статистичного аналізу успішності студентів у різних спеціалізаціях,

вивчення профілю випускників та їх подальшого розвитку на ринку праці. Крім того, дані можуть допомогти у виявленні тенденцій у навчальному процесі, вивченні потреб студентів та випускників, а також впливу університетської освіти на подальший кар'єрний шлях студентів.

У результаті роботи над створенням цього інструменту, можна буде отримати цінну інформацію про студентів різних ЗВО. Ця інформація буде корисною для багатьох зацікавлених сторін, включаючи університетську адміністрацію, викладачів та дослідників у галузі освіти та розвитку кар'єри.

Взаємозв'язок з іншими роботами. За методами розробки та інструментальними засобами робота виконувалася сумісно з роботами з розв'язання задач булевої алгебри та статистики.

РОЗДІЛ 1. ОСНОВНІ ПОНЯТТЯ ТА ПІДХОДИ ДО ЗБОРУ І АНАЛІТИКИ ДАНИХ СТУДЕНТІВ ЗВО

1.1 Огляд наявних інструментів на ринку

Важливо вивчити досвід поточних платформ, які успішно пройшли свій шлях, перш ніж розпочати процес розробки програми для збору даних випускників і їх подальшого аналізу. Аналізуючи переваги та недоліки відомих веб-сайтів, таких як LinkedIn і DataFox, ми можемо отримати важливі уроки та визначити важливі елементи для розробки успішної платформи, яка долає поточні проблеми. Ці платформи широко використовуються та пропонують передові функції, які приваблюють споживачів. Розуміння переваг і недоліків цих платформ допоможе нам розробити власний додаток, який задовольнить зростаючі вимоги споживачів.

1.1.1 LinkedIn

LinkedIn - найпопулярніша соціальна мережа з фокусом на обмін професійними, освітніми характеристиками, пошуку роботи та інших кар'єрних можливостей.

Соціальна мережа надає можливість перегляду профілів користувачів, з загальною інформацією про користувача, такою як освіта, досвід роботи, навички, зацікавленості та контактні дані. Однак, нерегульований та ручний процес перегляду профілів може бути часо- та ресурсозатратним завданням, особливо при великому обсязі даних. LinkedIn має низку обмежень на перегляд детальної інформації профілів, що ще більше ускладнює процес збору інформації.

Також можуть виникати складнощі у зборі та аналізі інформації, оскільки дані розміщені на різних профілях та можуть мати різний формат та структуру.

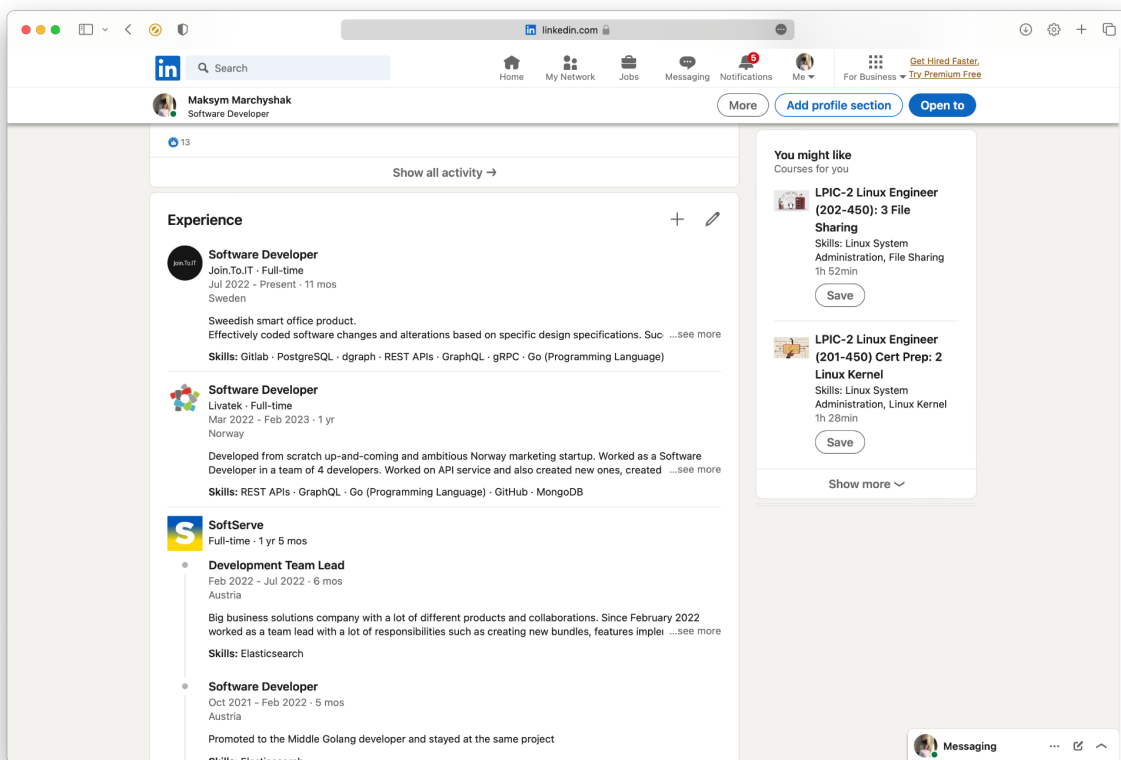


Рисунок 1.1 – Робочий досвід користувача

Даний сайт дає нам можливість отримати статистику вищого навчального закладу, проте вона є дуже узагальнена і не дає нам детальної інформації, такої як: кореляція навчальної дисципліни та позиції на роботі, подальші наукові досягнення випускників.

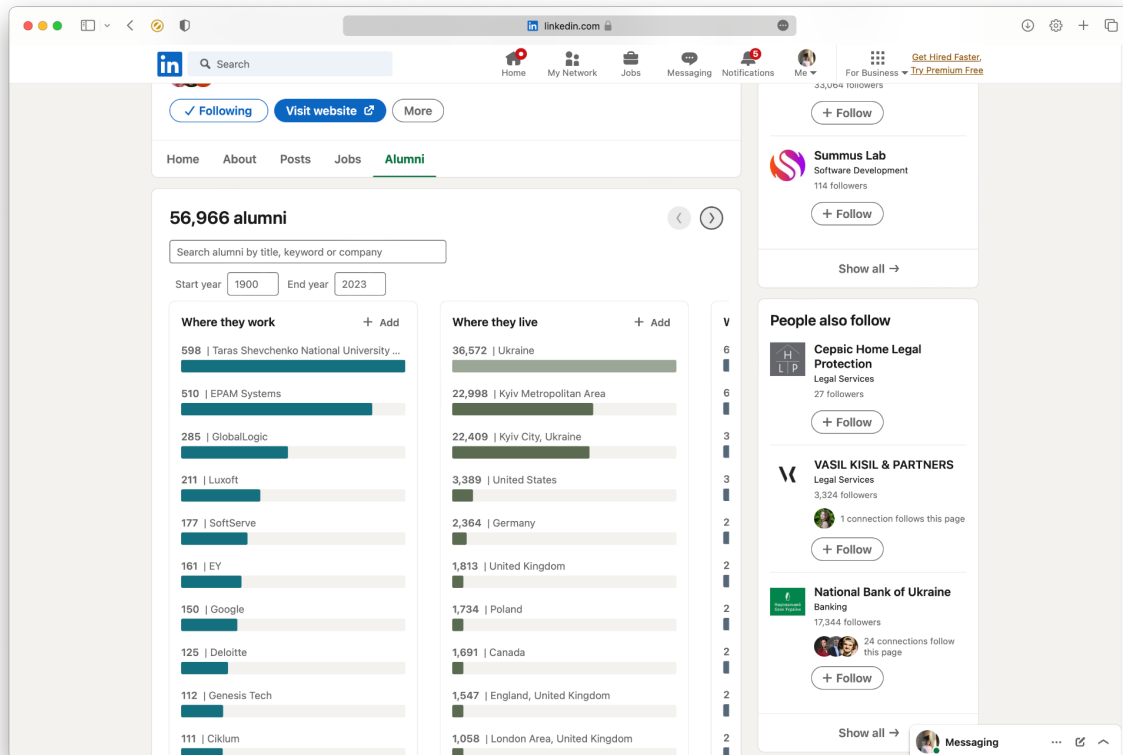


Рисунок 1.2 – Базова статистика випускників університету

1.1.2 DataFox

DataFox є компанією, що спеціалізується на аналізі даних компаній та профілів фахівців. Їхні інструменти дозволяють збирати та аналізувати інформацію з різних джерел, включаючи соціальні мережі, зокрема LinkedIn. Оскільки DataFox фокусується на аналізі даних компаній та профілів фахівців загалом, вони можуть виступати як потенційний конкурент у сфері збору та аналізу даних студентів та випускників.

Проте, існують певні аспекти, які DataFox може не вистачати для конкретної предметної області збору та аналізу даних студентів та випускників. Оскільки DataFox спрямовані на аналіз даних компаній, їхні інструменти можуть бути менш придатними для детального дослідження

освітнього досвіду, академічних досягнень та інших даних студентів та випускників університету.

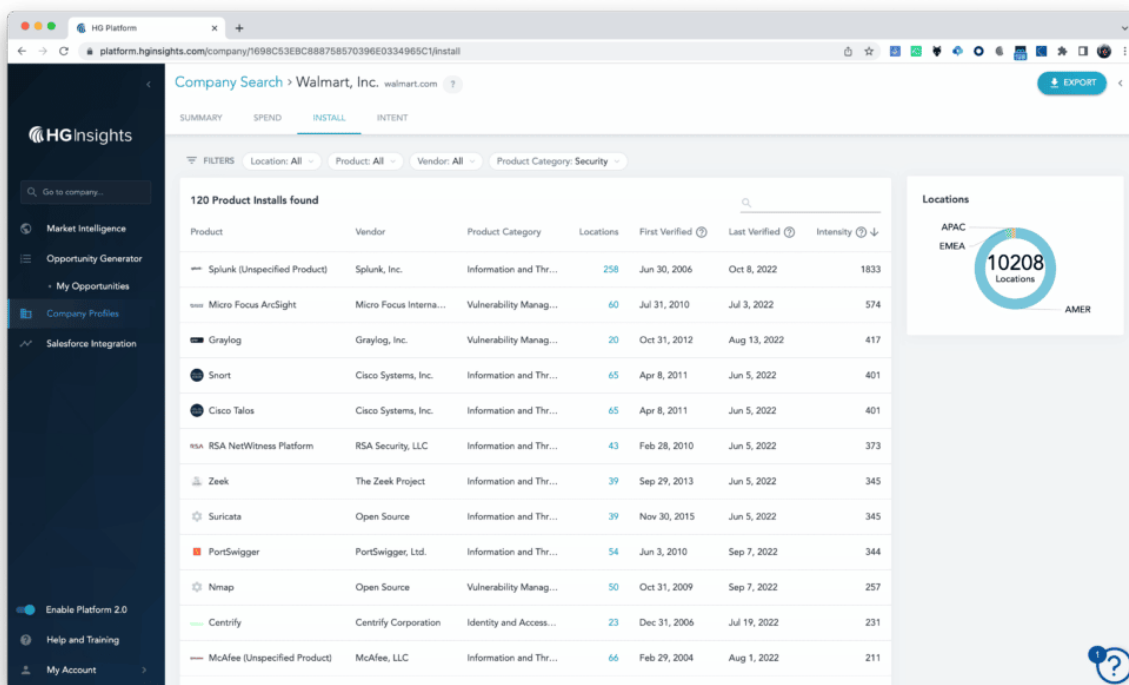


Рисунок 1.3 – Список профілів компаній

Окрім того, DataFox має обмеження щодо доступу до певних даних на LinkedIn, особливо якщо ці дані є приватними або обмеженими обліковими записами. Тому, для детального збору та аналізу інформації з профілів студентів та випускників ЗВО, є необхідність розробити спеціалізований програмний інструмент, який враховуватиме унікальні особливості цієї конкретної групи користувачів та їхніх потреб у зборі та аналізі даних.

Отже, хоча DataFox може бути конкурентом у загальному сенсі збору та аналізу даних профілів фахівців, вони можуть не мати спеціалізованого

фокусу на детальному аналізі даних студентів та випускників університету, що робить можливим розробку спеціалізованого програмного інструменту для вирішення конкретних потреб університетської спільноти.

2.2 Опис запропонованого підходу

Виходячи з наявних інструментів було запропоновано наступні функціональні вимоги:

- Забезпечення детального ознайомлення користувачів з даними про студентів та випускників: сервіс має на меті надати користувачам доступ до актуальної інформації про студентів та випускників навчального закладу. Це включає освітні дані, досягнення, професійний розвиток та інші важливі деталі, які допоможуть користувачам зробити обґрунтовані рішення та знайти взаємозв'язки між студентами та випускниками.
- Автоматизація процесів збору та аналізу даних: веб-сервіс має надавати зручні інструменти для автоматизації збору, обробки та аналізу даних профілів студентів та випускників. Це дозволить зменшити час та зусилля, необхідні для ручного збору та обробки даних, та сприятиме більш точному та систематичному аналізу цих даних. Такі процеси можуть включати збір особистої інформації, академічних досягнень, досвіду роботи.
- Функціональні підсистеми: Телеграм-бот має бути організованим на функціональні підсистеми, які відповідають потребам різних типів користувачів. Адміністративна підсистема буде призначена для керування даними. Підсистема студента буде надавати можливості перегляду статистики, зворотного зв'язку з адміністрацією.

Додаткові завдання для досягнення поставленої мети включають:

- 1) Визначити потрібну інформацію для збору з профілів студентів та випускників. Це може включати особисті дані (ім'я, контактна інформація), академічну інформацію (спеціальність, факультет, рейтингові показники), додаткові навчальні досягнення, досвід роботи та інші професійні навички. Потрібно визначити, які дані будуть корисними для аналізу та отримання статистики.
- 2) Розробити алгоритм або методику для автоматичного збору даних з профілів студентів та випускників на LinkedIn. Це включає розробку програмного коду, який буде сканувати та збирати необхідну інформацію з профілів, використовуючи доступні API або веб-скрапінг техніки.
- 3) Перевірити та забезпечити відповідність процесу збору даних політикам та правилам соціальної мережі LinkedIn. Важливо дотримуватися умов використання та забезпечити безпеку та конфіденційність даних, зброюваних з профілів студентів та випускників.
- 4) Зберегти та організувати зібрані дані в зручному форматі для подальшого аналізу. Це може включати створення бази даних або файлів CSV для зберігання та структурування отриманих даних.
- 5) Застосувати методи аналізу даних для отримання високоякісної інформації та статистики. Це може включати розрахунок середніх значень, створення діаграм та графіків, виявлення кореляційних залежностей тощо.
- 6) Перевірити та оцінити точність та достовірність зібраних даних. Важливо провести перевірку та оцінку якості зібраних

даних, зокрема виявити та виправити можливі помилки, дублікати або неповні дані.

РОЗДІЛ 2. ОПИС ВИКОРИСТАНИХ ТЕХНОЛОГІЙ

2.1 Середовище програмування

Розробку було здійснено з використанням операційної системи **MacOS** від Apple - це операційна система яка підтримує широкий спектр різних програмних інструментів та середовищ для ефективної та продуктивної розробки різного роду програм.

Сервіси веб-серверу були запрограмовані в **Visual Studio Code** - це легкий та при цьому потужний інструмент для написання та редагування коду, а також процеси тестування та потужні функції та інструменти для розробки програмного забезпечення. Підтримує багато різних мов програмування, сторонніх бібліотек, фреймворків та пакетів, що дозволяє всіляко персоналізувати середовище розробки в залежності від потреб.

2.2 Сервіс для збору даних з профілів студентів

2.2.1 Мова програмування JavaScript

Мова програмування JavaScript є широко використовуваною для веб скрапінгу, тобто отримання даних з веб-сторінок [3]. JavaScript має вбудовану підтримку для маніпулювання вмістом HTML-сторінок, зчитування та запису даних, взаємодії з серверами та багатьох інших функцій, що дозволяють збирати необхідні дані з веб-джерел.

Основна перевага JavaScript для веб скрапінгу полягає у тому, що вона вбудована у більшість сучасних браузерів, що дозволяє виконувати скрипти прямо на клієнтському боці. Це робить JavaScript потужним інструментом для отримання даних зі сторінок, які використовують JavaScript для відображення та маніпуляції вмістом.

Для веб скрапінгу з використанням JavaScript можна використовувати різні бібліотеки та фреймворки, такі як Selenium[4], Cheerio, jsdom та інші. Ці інструменти надають зручний спосіб маніпулювання DOM-структурами, виконання запитів до серверів, парсингу HTML-коду та отримання потрібних даних.

JavaScript також підтримує асинхронні запити, що дозволяють ефективно обробляти багато запитів до різних сторінок або джерел даних одночасно. Це особливо корисно при скрапінгу великої кількості даних або при скрапінгу даних з різних джерел одночасно.

2.2.2 Бібліотека Selenium

Бібліотека Selenium є однією з найпопулярніших інструментів для веб скрапінгу з використанням мови програмування JavaScript. Вона надає розширені можливості для автоматизації взаємодії з веб-браузерами.

Основна перевага Selenium полягає в тому, що вона дозволяє програмно керувати браузером, виконувати різні дії, такі як клікання, введення тексту, натискання клавіш і т.д., а також отримувати вміст сторінок і взаємодіяти з елементами DOM.

Selenium підтримує різні браузери, такі як Google Chrome, Mozilla Firefox, Microsoft Edge, Safari тощо. Це дозволяє вам тестувати та скрапити веб-сторінки на різних платформах і браузерах.

Окрім того, Selenium може бути інтегрована з різними мовами програмування, включаючи JavaScript. Це дає можливість розробникам використовувати JavaScript для написання скриптів автоматизації та скрапінгу веб-сторінок з використанням Selenium.

2.3 API сервіс

2.3.1 Мова програмування Golang

Для створення сервісу для аналізу та відображення даних було обрано мову програмування Golang. Go є компільованою мовою програмування, яка пропонує вбудовані засоби для ефективних паралельних обчислень та дозволяє використовувати засоби віддаленого керування пакетами[5]. Ця мова програмування була розроблена з урахуванням потреб сучасних програмних проектів, забезпечуючи швидкість компіляції, низький рівень використання пам'яті та простоту використання.

Go пропонує вбудовану підтримку паралельних обчислень, що дозволяє розробникам легко розпаралелювати свої програми для ефективного використання багатоядерних процесорів і поліпшення продуктивності. Засоби для паралельного програмування в Go включають горутини (goroutines) та канали (channels), які сприяють зручному взаємозв'язку між паралельними процесами[6].

Крім того, Go має вбудовану систему керування пакетами, що дозволяє легко управляти залежностями та встановлювати зовнішні бібліотеки. Це спрощує розробку проектів, забезпечуючи зручну організацію та управління кодом.

В результаті, Go є мовою програмування, яка пропонує потужні засоби для паралельних обчислень і зручну систему керування пакунками, що робить її привабливим вибором для розробки швидких та ефективних програмних рішень.

2.3.2 Фреймворк GIN

Фреймворк Gin є легким та швидким фреймворком для розробки веб-додатків з використанням мови програмування Go [7]. Він пропонує простий та ефективний спосіб побудови веб-серверів, дозволяючи розробникам швидко створювати API і веб-сайти.

Gin володіє високою продуктивністю та низьким рівнем навантаження на сервер, завдяки чому він добре підходить для розробки високонавантажених додатків. Фреймворк пропонує широкі можливості для маршрутизації HTTP-запитів, забезпечуючи гнучкий контроль над обробкою маршрутів та параметрів запитів [8].

Основні особливості Gin включають підтримку обробки запитів методами GET, POST, PUT, DELETE, PATCH, а також можливість роботи з параметрами запитів, заголовками, кукісами та валідацією даних. Фреймворк також пропонує вбудовану підтримку для серіалізації та десеріалізації даних у форматах JSON [9], XML [10] та інших.

Gin відрізняється простотою використання та має дружній для розробника інтерфейс. Він надає потужні можливості для побудови веб-додатків швидко та ефективно, забезпечуючи гнучкість та розширюваність. Фреймворк також має велику спільноту розробників, що сприяє наявності багатьох ресурсів та допомоги при роботі з ним.

2.4 Клієнтський сервіс

2.4.1 Фреймворк Telebot

Фреймворк Telebot є популярним фреймворком для розробки телеграм-ботів з використанням мови програмування Go [11]. Він надає

розробникам зручний спосіб створення та керування ботами, що працюють у Telegram-платформі.

Telebot пропонує широкий набір функцій та інструментів для взаємодії з Telegram API. Він дозволяє створювати різноманітні команди та реагувати на події, такі як отримання повідомлень від користувачів, натискання кнопок, отримання зображень тощо. Фреймворк надає можливість обробляти ці події та відповідати на них за допомогою власного програмного коду.

Основні особливості Telebot включають підтримку текстових повідомлень, стикерів, зображень, голосових повідомлень, аудіо та відеофайлів. Фреймворк також підтримує розсилку повідомлень, роботу з клавіатурами, опитуваннями та іншими елементами взаємодії з користувачами.

Telebot володіє зручним інтерфейсом, що спрощує розробку телеграм-ботів. Він дозволяє швидко створювати та налаштовувати ботів з мінімальними зусиллями. Фреймворк також має велику спільноту розробників, що забезпечує наявність документації, прикладів та підтримки для роботи з ним.

2.5 База даних MongoDB

MongoDB — це потужна й адаптована система керування базами даних, яка пропонує програмістам і компаніям різноманітні інструменти для зберігання, керування й аналізу великих обсягів даних [12].

.Розроблена MongoDB Inc., є документно-орієнтованою системою керування базами даних (СУБД). Відома своєю адаптивністю та швидкістю обробки даних. Ключовими характеристиками бази даних

MongoDB є гнучка структура даних, горизонтальне масштабування, відмінна продуктивність і простота використання.

MongoDB зберігає інформацію в документах у форматі JSON, що дозволяє поєднувати структуровані дані з пов'язаними метаданими в одному документі. Це забезпечує легкий доступ до інформації та гнучкість структури даних. Крім того, MongoDB підтримує низку запитів, індексування та структур агрегації для ефективного аналізу даних.

У міру збільшення обсягу даних і робочого навантаження ви можете швидко масштабувати свою базу даних за допомогою вбудованої масштабованості програмного забезпечення MongoDB. Підтримуються розподілені системи, що дозволяє розподіляти дані на кількох серверах і гарантувати високу доступність і надійність.

Щоб забезпечити конфіденційність і цілісність даних, MongoDB також надає ряд функцій безпеки, включаючи автентифікацію, авторизацію та шифрування. Він взаємодіє з іншими продуктами та рішеннями екосистеми MongoDB, дозволяючи розширити його функціональність і отримати доступ до інструментів для аналізу бізнесу, надійної реплікації та резервного копіювання.

РОЗДІЛ 3. РЕАЛІЗАЦІЯ ІНСТРУМЕНТУ

3.1 Реалізація бази даних

База даних складається з 7 таблиць, структура яких зображена на діаграмі бази даних (див. додаток А). Опис кожної з таблиць наведено у табл. 3.1.

| Номер | Назва | Опис |
|-------|-----------------|--|
| 1 | students | Таблиця для збереження інформації про студента |
| 2 | experience | Таблиця для збереження інформації про досвід роботи студента |
| 3 | skills | Таблиця для збереження інформації про вміння студента |
| 4 | courses | Таблиця для збереження інформації про освітні курси студента |
| 5 | education | Таблиця для збереження інформації про освіту студента |
| 6 | students_search | Таблиця для збереження послідовної черги скрапінгу студентів |
| 7 | users | Таблиця для збереження даних про користувачів веб-сервісом |

Таблиця 3.1 – Короткий опис таблиць

Таблиця `students` (див. табл. 3.2) містить унікальний ідентифікаційний код студента, ім'я, автобіографія, локація, посилання на фото, посилання на профіль користувача, посада. Додаткові дані та їх зв'язок із студентом зберігається в таблицях 3.3, 3.4, 3.5, 3.6. Кожен студент має освіту, досвід роботи, список вмінь, інформація про додаткові учбові курси.

| Атрибут | Тип | Опис |
|--------------------------|--------|-----------------------------------|
| <code>_id</code> | string | Унікальний ідентифікатор студента |
| <code>fullName</code> | string | Ім'я та фамілія студента |
| <code>location</code> | string | Локація студента |
| <code>photo</code> | string | Посилання на фото студента |
| <code>url</code> | string | Посилання на профіль студента |
| <code>description</code> | string | Автобіографія студента |
| <code>sex</code> | string | Стать |

Таблиця 3.2 – Таблиця `students`

Таблиця зберігає інформацію про зв'язок студента з досвідом роботи(див. табл. 3.3). Також для кожного студента зберігається зв'язок з таблицею освіти(табл. 3.6), навичок(табл. 3.4), освітніх курсів(табл. 3.5).

| Атрибут | Тип | Опис |
|--------------------|--------|--------------------------|
| <code>_id</code> | string | Унікальний ідентифікатор |
| <code>title</code> | string | Посада |

| | | |
|----------------|--------|---------------------------|
| company | string | Назва компанії |
| location | string | Локація |
| description | string | Опис робочого досвіду |
| startDate | string | Початок роботи в компанії |
| endDate | string | Кінець роботи в компанії |
| endDatePresent | bool | Чи є досі працює |
| studentId | string | Ідентифікатор студента |

Таблиця 3.3. Таблиця experience

| Атрибут | Тип | Опис |
|-----------|--------|--------------------------|
| _id | string | Унікальний ідентифікатор |
| name | string | Назва вміння |
| studentId | string | Ідентифікатор студента |

Таблиця 3.4 – Таблиця skills

| Атрибут | Тип | Опис |
|----------|--------|--------------------------|
| _id | string | Унікальний ідентифікатор |
| name | string | Назва курсу |
| issuedBy | string | Хто видав |

| | | |
|-------------|--------|----------------------------------|
| location | string | Локація курсу |
| description | string | Опис курсу |
| startDate | string | Початок навчання |
| endDate | string | Кінець навчання |
| url | string | Посилання на підтвердження курсу |
| studentId | string | Ідентифікатор студента |

Таблиця 3.5 – Таблиця courses

| Атрибут | Тип | Опис |
|--------------|--------|---------------------------|
| _id | string | Унікальний ідентифікатор |
| schoolName | string | Назва навчального закладу |
| degreeName | string | Назва ступеня |
| fieldOfStudy | string | Назва спеціальності |
| description | string | Опис навчання |
| startDate | string | Початок навчання |
| endDate | string | Кінець навчання |
| studentId | string | Ідентифікатор студента |

Таблиця 3.6 – Таблиця education

Таблиця students_search (див табл. 3.7) зберігає інформацію для модулю, який відповідає за скрапінг та додавання інформації про студента

в базу даних. Таблиця виконує роль черги, записи з якої видаляються коли дані користувача успішно зібрані.

| Атрибут | Тип | Опис |
|-----------------|--------|-------------------------------|
| _id | string | Унікальний ідентифікатор |
| profileUrl | string | Посилання на профіль студента |
| fullName | string | Повне ім'я студента |
| firstName | string | Ім'я студента |
| lastName | string | Фамілія студента |
| profilePhotoUrl | string | Посилання на фото студента |
| job | string | Посада на роботі |
| location | string | Локація студента |
| query | string | Пошуковий запит для профілю |
| category | string | Категорія пошуку |
| timestamp | int64 | Дата у форматі UNIX |

Таблиця 3.7 – Таблиця students_search

Таблиця users (див. табл. 3.8) містить інформацію користувачів, їх роль та загальну інформацію. Адміністратори мають доступ до керування даними в веб-сервісі, авторизовані користувачі можуть переглядати статистику.

| Атрибут | Тип | Опис |
|-----------|--------|---------------------------------|
| _id | int64 | Унікальний ідентифікатор |
| role | string | Роль користувача |
| fullName | string | Повне ім'я користувача |
| createdAt | int64 | UNIX дата створення користувача |
| updatedAt | int64 | UNIX дата оновлення користувача |

Таблиця 3.8 – Таблиця users

3.2 Реалізація сервісу

Архітектура продукту складається з чотирьох окремих сервісів, які спілкуються між собою. Цей підхід має кілька переваг, серед них:

- 1) Масштабованість: кожен сервіс можна масштабувати незалежно від інших, розділивши продукт на різні сервіси. Це означає, що ви можете лише збільшити ресурси для сервісів, які потребують додаткової продуктивності або обробки даних. За допомогою цієї стратегії ви можете забезпечити продуктивність та швидкодію, а також оптимізувати використання ресурсів.
- 2) Гнучкість: поділ продукту на окремі послуги дає більше свободи в розробці та модифікації. Різні групи розробників можуть інтегрувати нові функції або пришвидшити вирішення проблем, оскільки кожен сервіс можна створювати, випускати та масштабувати незалежно.
- 3) Висока доступність: здатність інших служб продовжувати працювати самостійно, якщо один сервіс має труднощі.
- 4) Розподіл обов'язків: кожен сервіс може відповідати за певну частину функцій або бізнес-логіку. Це дає змогу підтримувати

відповідальність за різні компоненти продукту, а також краще організувати роботу команд. Розподіл обов'язків також полегшує тестування та впровадження змін, оскільки зменшується вплив на інші служби.

3.2.1 Сервіс Scrapper

WEB прикладний програмний інтерфейс (API), реалізований з використанням мови програмування JavaScript та фреймворку Selenium, дозволяє емулювати поведінку користувача з метою отримання даних з профілів студентів на платформі "LinkedIn" і уникнення обмежень, що застосовуються системою для виявлення ботів. Цей сервер приймає посилання на профіль користувача та авторизаційний токен LinkedIn для проходження авторизації. Він здійснює аналіз та вибір ключової інформації зі сторінки профілю та повертає ці дані у структурованому форматі.

Сервіс забезпечує більш ефективне та автоматизоване отримання важливої інформації з профілів студентів на платформі "LinkedIn", дозволяючи уникнути обмежень, що накладаються системою виявлення ботів. Результати аналізу повертаються у структурованому вигляді, що спрощує подальшу обробку та аналіз даних.

Цей підхід дозволяє автоматизувати процес отримання даних з профілів студентів на "LinkedIn" шляхом моделювання поведінки реального користувача. Використання Selenium у поєднанні з JavaScript дозволяє контролювати веб-браузер та взаємодіяти з веб-елементами на сторінці, що дозволяє отримувати доступ до необхідної інформації з

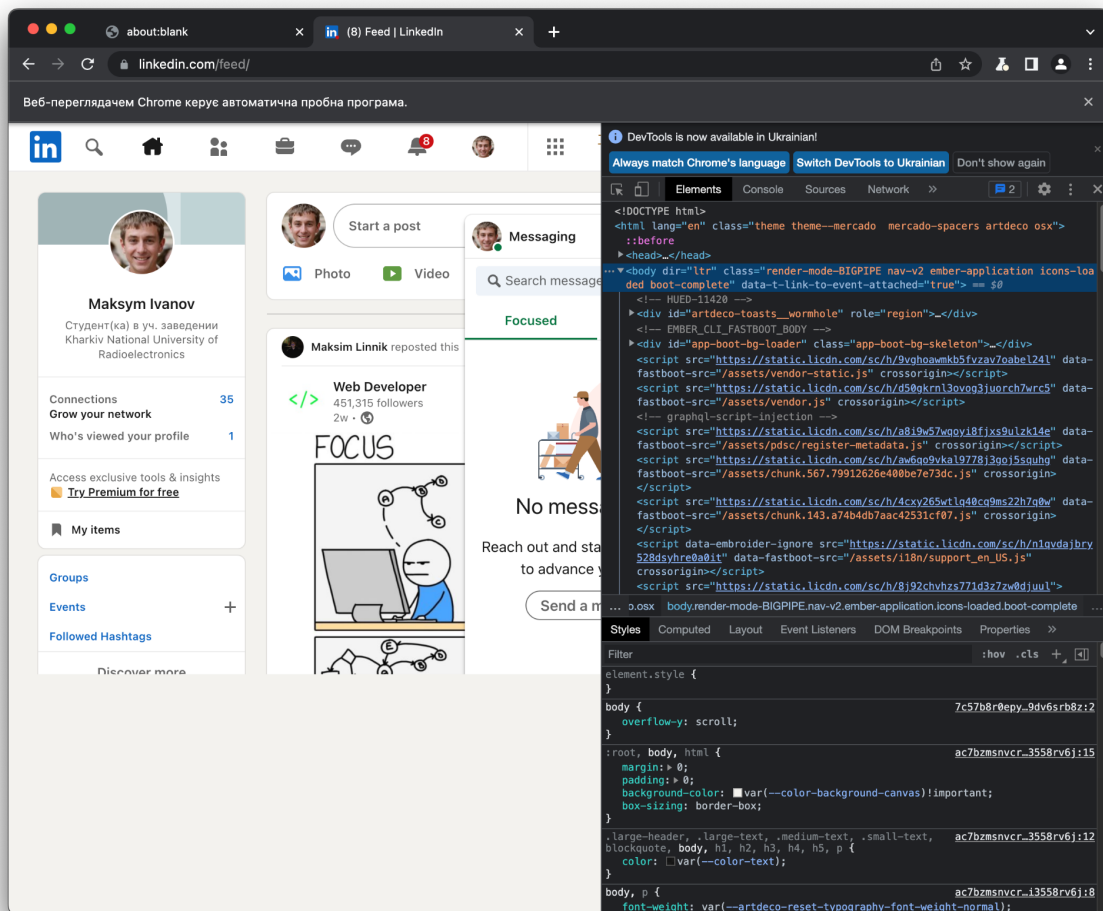


Рисунок 3.1 - Автоматизований веб браузер який емулює дії користувача

3.2.2 API Сервіс

Основний API модуль, що виконує аналіз та обробку даних для відображення, сортування та побудови графіків, реалізований у сервісі з використанням мови програмування Golang та бібліотеки Gin.

```

[GIN-debug] POST    /auth/register      -> github.com/supperdoggy/diploma_university_statistics_tool/api/pkg/handlers.IHandlers.Register-fm (3 handlers)
[GIN-debug] POST    /auth/login        -> github.com/supperdoggy/diploma_university_statistics_tool/api/pkg/handlers.IHandlers.Login-fm (3 handlers)
[GIN-debug] POST    /auth/check_token  -> github.com/supperdoggy/diploma_university_statistics_tool/api/pkg/handlers.IHandlers.CheckToken-fm (3 handlers)
[GIN-debug] POST    /auth/new_email_code -> github.com/supperdoggy/diploma_university_statistics_tool/api/pkg/handlers.IHandlers.NewEmailCode-fm (3 handlers)
[GIN-debug] POST    /auth/check_email_code -> github.com/supperdoggy/diploma_university_statistics_tool/api/pkg/handlers.IHandlers.CheckEmailCode-fm (3 handlers)
[GIN-debug] POST    /api/v1/users/create -> github.com/supperdoggy/diploma_university_statistics_tool/api/pkg/handlers.IHandlers.CreateUser-fm (4 handlers)
[GIN-debug] DELETE  /api/v1/users/delete -> github.com/supperdoggy/diploma_university_statistics_tool/api/pkg/handlers.IHandlers.DeleteUser-fm (4 handlers)
[GIN-debug] PATCH  /api/v1/users/update -> github.com/supperdoggy/diploma_university_statistics_tool/api/pkg/handlers.IHandlers.UpdateUser-fm (4 handlers)
[GIN-debug] GET    /api/v1/users/get/:id -> github.com/supperdoggy/diploma_university_statistics_tool/api/pkg/handlers.IHandlers.GetUser-fm (4 handlers)
[GIN-debug] GET    /api/v1/schools/list -> github.com/supperdoggy/diploma_university_statistics_tool/api/pkg/handlers.IHandlers.ListSchools-fm (4 handlers)
[GIN-debug] POST    /api/v1/schools/top_companies -> github.com/supperdoggy/diploma_university_statistics_tool/api/pkg/handlers.IHandlers.ListSchoolsTopCompanies-fm (4 handlers)
}
[GIN-debug] POST    /api/v1/schools/job_titles -> github.com/supperdoggy/diploma_university_statistics_tool/api/pkg/handlers.IHandlers.ListJobTitlesBySchool-fm (4 handlers)
[GIN-debug] POST    /api/v1/schools/correlation_between_degree_title -> github.com/supperdoggy/diploma_university_statistics_tool/api/pkg/handlers.IHandlers.CorrelationBetweenDegreeAndTitle-fm (4 handlers)
[GIN-debug] POST    /api/v1/schools/degrees -> github.com/supperdoggy/diploma_university_statistics_tool/api/pkg/handlers.IHandlers.SchoolDegrees-fm (4 handlers)
[GIN-debug] GET    /api/v1/companies/list -> github.com/supperdoggy/diploma_university_statistics_tool/api/pkg/handlers.IHandlers.ListCompanies-fm (4 handlers)
[GIN-debug] POST    /api/v1/companies/top_schools -> github.com/supperdoggy/diploma_university_statistics_tool/api/pkg/handlers.IHandlers.ListCompaniesTopSchools-fm (4 handlers)
}
[GIN-debug] POST    /api/v1/companies/top_degrees_hired -> github.com/supperdoggy/diploma_university_statistics_tool/api/pkg/handlers.IHandlers.TopHiredDegrees-fm (4 handlers)
[GIN-debug] [WARNING] You trusted all proxies, this is NOT safe. We recommend you to set a value.
Please check https://pkg.go.dev/github.com/gin-gonic/gin#readme-don-t-trust-all-proxies for details.
[GIN-debug] Listening and serving HTTP on :8080

```

Рисунок 3.2 – Доступні методи та запити в головному модулі

Цей сервіс забезпечує потужні функціональні можливості для обробки та аналізу даних студентів та випускників. Завдяки використанню мови програмування Golang, він відрізняється високою продуктивністю та швидкодією. Бібліотека Gin, що використовується у розробці, надає зручні інструменти для реалізації веб-сервера та роутингу, що спрощує розробку та підтримку API[13].

Цей сервіс дозволяє отримувати, зберігати та обробляти дані про студентів та випускників. Він забезпечує можливість відображення даних у зручному та зрозумілому форматі, а також сортування цих даних за різними параметрами. Крім того, сервіс дозволяє будувати графіки та діаграми на основі отриманих даних, що сприяє зрозумілому візуалізації та аналізу інформації.

API модуль виконує інтеграцію з базою даних, де зберігаються інформація про студентів та випускників. Він забезпечує безпечний доступ до даних, автентифікацію та авторизацію користувачів, а також забезпечує захист даних від несанкціонованого доступу.

3.2.3 Telegram bot client

Цей чат-бот використовується для зручної взаємодії користувачів з основним обчислювальним модулем, який обробляє та аналізує дані про студентів та випускників.

Завдяки використанню мови програмування Golang та бібліотеки Telebot, клієнт-чат-бот забезпечує швидку та ефективну роботу, а також зручний та інтуїтивно зрозумілий інтерфейс для користувачів. Користувачі можуть відображати інформацію про студентів та випускників, відправляти запити на отримання певних даних або виконання певних дій(див. Додаток Б).

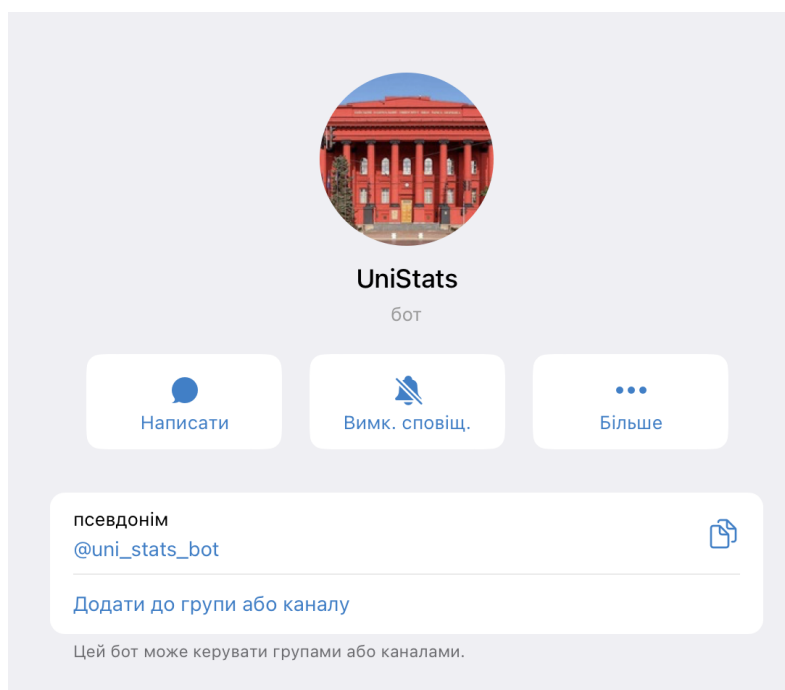


Рисунок 3.3 - Профіль телеграм боту

Завдяки комунікації цього чат-бота з головним модулем комп'ютера користувачі можуть легко отримати актуальну статистику популярності спеціальностей та освітніх програм. З його допомогою можна відправляти команди в обчислювальний модуль і отримувати дані про студентів і випускників .

Визначення популярності навчальних закладів, програм і наукових ступенів у бізнесі є однією з ключових функціональних складових чат-бота. Користувачі можуть вказати критерії сортування, такі як

кількість студентів, назва спеціальності, роки вступу та випуску, ступінь і назва університету. Після завершення запиту чат-бот отримує інформацію з обчислювального модуля та відображає її користувачеві, сортуючи результати відповідно до параметрів.

Через відомий месенджер Telegram цей чат-бот дозволяє користувачам спілкуватися з системою з будь-якого пристрою. Це полегшує вам отримання найновішої інформації про студентів і випускників без необхідності завантажувати додаткове програмне забезпечення або відвідувати веб-сайт. Можливість користувачів ставити запитання, отримувати статистичні дані та взаємодіяти з системою в режимі реального часу робить процес збору даних і отримання висновків більш практичним і ефективним.

Усі ці функції дозволяють користувачам зручно взаємодіяти з системою, спрощують швидкий і ефективний збір відповідних даних про студентів і випускників, а також допомагають системі розвиватися.

3.3 Інструкція використання

3.3.1 Інструкція неавторизованого користувача

Коли користувач вперше заходить в чат з телеграм ботом, та нажимає кнопку “Розпочати”, то бот відповідає йому повідомленням про те що користувач не може користуватися ботом. Для того щоб користувач зміг користуватися ботом, потрібно щоб адміністрація добавила його ідентифікатор в список дозволених користувачів.

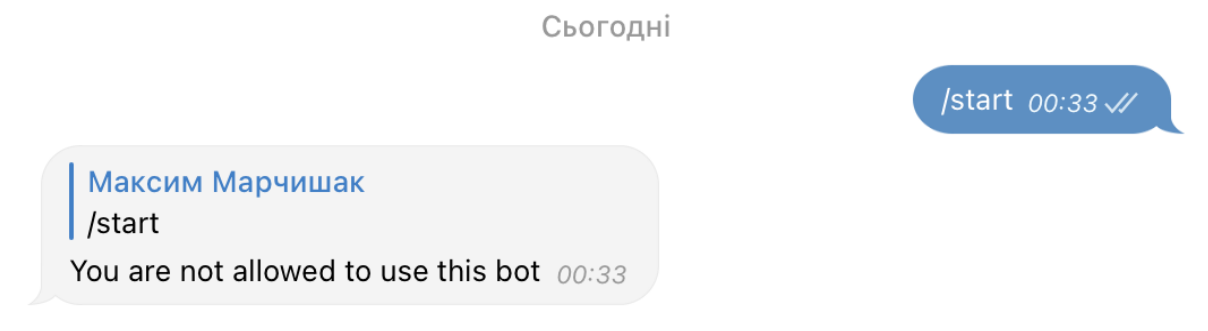


Рисунок 3.4 - Користувач не авторизований

3.3.2 Інструкція авторизованого користувача

Коли користувач отримав дозвіл на користування ботом в нього більше немає цього повідомлення про помилку і він може користуватися всім функціоналом, який доступний в меню.

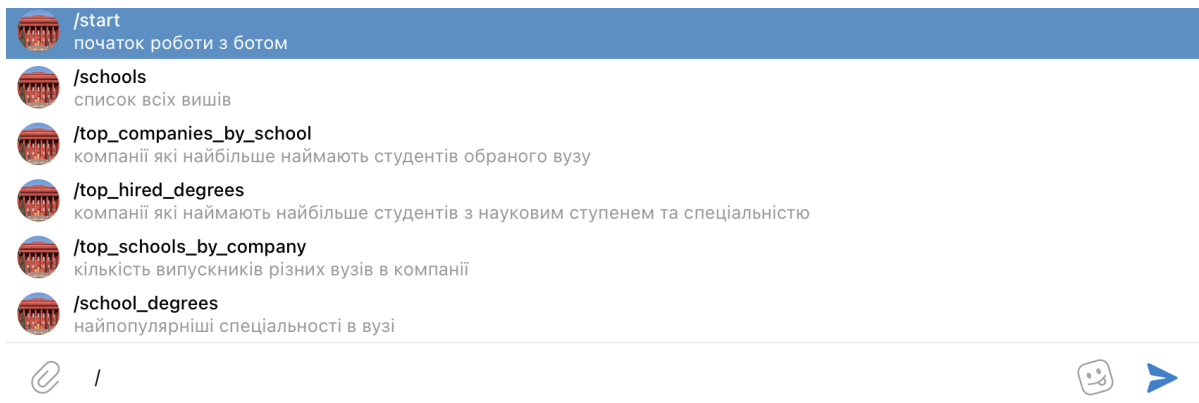


Рисунок 3.5 - Всі доступні команди для користувача

Авторизований користувач, використовуючи чат-бот в месенджері "Telegram", може отримати список всіх вищих навчальних закладів, які доступні в веб-сервері. Ця функція дозволяє користувачеві ознайомитися з усіма навчальними закладами, які включені в систему, та отримати загальну інформацію про них.

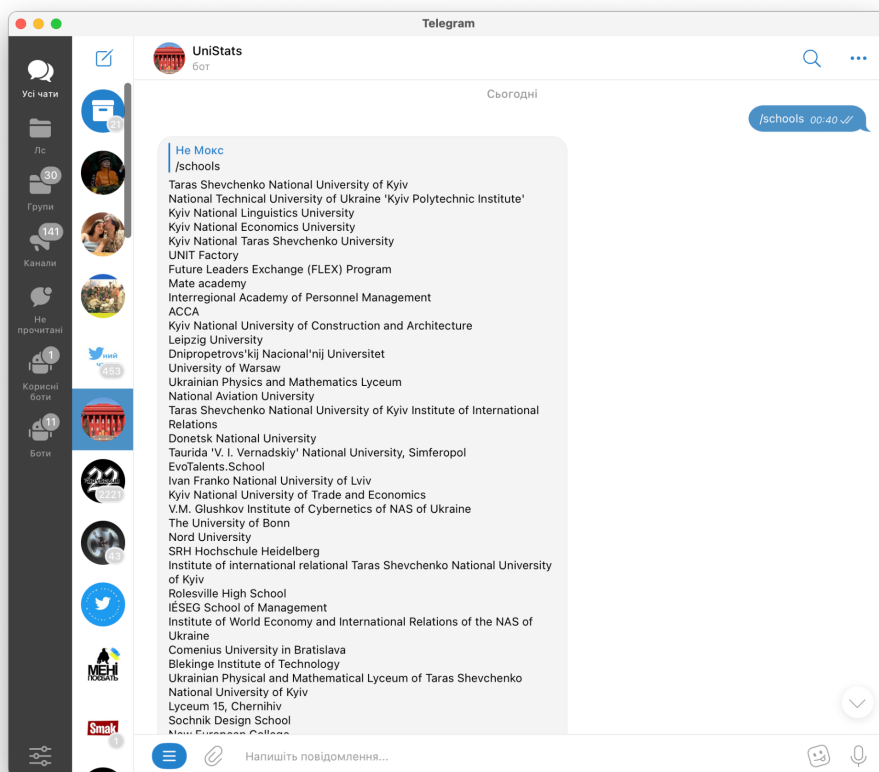


Рисунок 3.6 - Всі доступні навчальні заклади в яких студенти проходили навчання

Для отримання списку навчальних закладів користувач може виконати відповідну команду або запит до чат-бота. Після цього чат-бот звертається до основного обчислювального модуля і отримує актуальний список вищих навчальних закладів. Результати можуть бути відображені користувачу у вигляді повідомлення або спеціального інтерфейсу, що дозволяє зручно переглядати та фільтрувати дані про навчальні заклади.

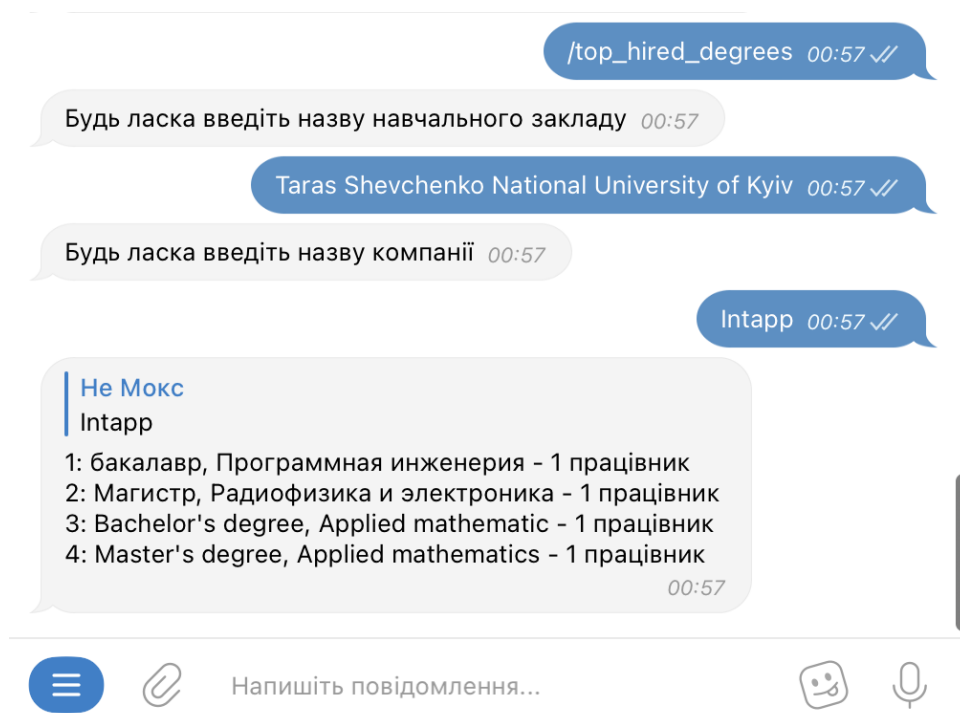


Рисунок 3.7 - Кількість працівників в компаніях за навчальним закладом

Користувач може подивитися список компаній в яких працюють студенти вузу, а також кількість працівників. А також детальніше які спеціальності популярні у роботодавців. Ця інформація дозволяє показати зв'язок між обраною спеціальністю та практичним застосуванням отриманих знань і навичок.

Дослідженням освітніх вимог та вимог до кваліфікації, які компанії встановлюють при прийомі на роботу, можна з'ясувати, які спеціальності чи галузі є найбільш затребуваними.

Включення такої інформації дозволить аудиторії краще зрозуміти перспективи працевлаштування після завершення навчання на обраній спеціальності та додатково підкреслить практичну значимість вашої роботи.

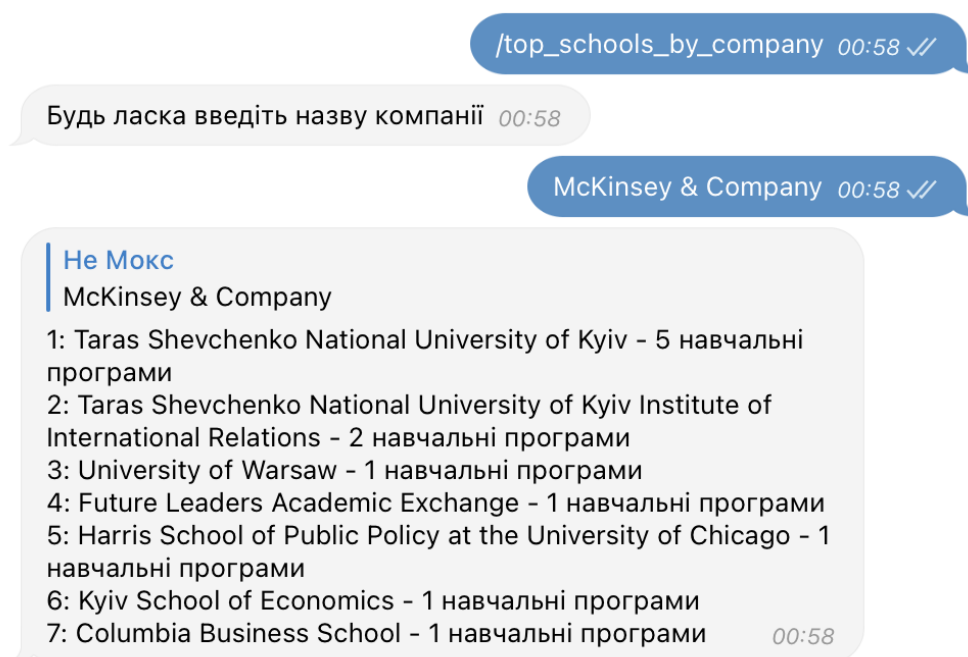


Рисунок 3.8 - Освіта працівників компанії

Також важливим елементом є інформація про найпопулярніші серед студентів спеціальності. Це може бути корисно для контекстуалізації дослідження та відображення його значущості. Найпопулярніші спеціальності можуть варіюватися в залежності від актуальних тенденцій та потреб на ринку праці.

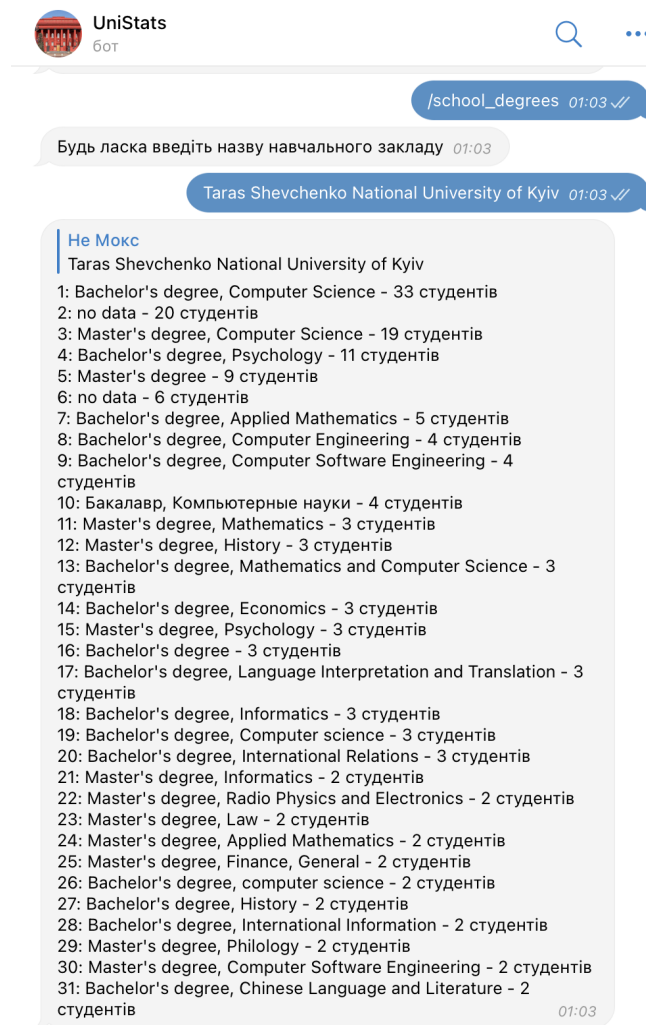


Рисунок 3.9 - Найпопулярніші спеціальності

3.3.3 Інструкція адміністратора

Адміністратори є найважливішим елементом системи, адже вони відповідають за керування доступом користувачів до боту, а також стежать за оновленням інформації та підтримують її в свіжому стані. Для адміністраторів є окремий функціонал який недоступний простим користувачам, який відповідає за оновлення даних студентів та кількістю користувачів, не дозволяють несанкціонований витік конфіденційної інформації.

Адміністратори мають доступ до збирання інформації через сервіс Scrapper, через командний термінал обчислювального серверу.

```
> linkedin-profile-scraper-ezdevs@1.0.0 start
> node index.js

Server setup: Setting up...
Adding helper methods to page
Server setup: All done. Listening on port 3000!
https://www.linkedin.com/in/...
Expanding all sections by clicking their "See more" buttons
.pv-profile-section.pv-about-section .inline-show-more-text__button false
#experience-section .pv-profile-section__see-more-inline false
.pv-profile-section.education-section .pv-profile-section__see-more-inline false
.pv-skill-categories-section .pv-profile-section__card-action-bar false
.pv-profile-section--certifications-section .pv-profile-section__see-more-inline false
https://www.linkedin.com/in/... 5148
```

Рисунок 3.10 - Збирання даних по посиланню

Системи адміністратора можуть обмежувати доступ користувачів до дозволених функцій, а також додавати, блокувати та обмежувати доступ.

Надсилання запрошень дозволяє адміністраторам включати нових користувачів до авторизованої групи. Запрошення користувача автоматичну прийметься коли користувач напише будь-яке повідомлення боту.

Адміністратори також мають можливість надавати певним користувачам доступ до дозволених функцій. Цю функцію можна застосувати, коли потрібно обмежити доступ користувача через порушення політики використання системи або в інших ситуаціях.

Крім того, адміністратори мають можливість блокувати користувачів, що повністю припиняє їхній доступ до системи. Цю функцію можна застосувати для підтримки безпеки системи або припинення небажаних дій.

Адміністратори можуть здійснювати контроль над авторизованими користувачами системи за допомогою цих функцій, вивчати контроль доступу та гарантувати безпеку та ефективну роботу системи в цілому.

3.4 Розвиток інструменту

Веб-сервіс можна покращити додавши наступний функціонал:

- Додавання нових алгоритмів аналізу та обробки даних для отримання більш детальної та цінної інформації про студентів та випускників.
- Вдосконалення алгоритмів сортування, фільтрації та побудови графіків для зручного відображення та аналізу даних.
- Додавання нових команд та функцій, що дозволяють користувачам отримувати більш детальну інформацію про студентів та випускників, наприклад, за певними критеріями або розширеними фільтрами.
- Зробити повноцінний веб клієнт для користувачів

ВИСНОВОК

Створений інструмент, дає можливість збирати дані профілів студентів та випускників ЗВО, а також надавати детальну статистику щодо популярності освітніх програм серед роботодавців та подальший кар'єрний шлях випускників. Результати є корисні для подальшого вивчення, аналізу та застосування в контексті взаємодії зі студентами та випускниками.

Для досягнення визначеної мети було:

- 1) Проаналізовано існуючі додатки, які дозволяють збирати дані профілів студентів та випускників університету та аналізувати їх освіту та кар'єру. Визначено переваги та недоліки кожного з них, визначено аспекти для покращення та поставлено відповідні задачі.
- 2) Успішно розроблено програмний додаток для автоматизованого збору даних з профілів студентів і випускників у соціальній мережі LinkedIn. Створено алгоритм, який ефективно вивчає та збирає необхідну інформацію з профілів за допомогою доступних API та методів веб-скрапінгу.
- 3) Умови використання соціальної мережі LinkedIn були досліджені та взяті до уваги, щоб переконатися, що процедура збору даних відповідає її політикам і вказівкам. При розробці програмного забезпечення було приділено належну увагу тому, щоб гарантувати безпеку та конфіденційність даних, отриманих із профілів студентів і випускників.
- 4) Зібрану інформацію було збережено та поміщено у зручний для використання формат для аналізу. Щоб зберегти та впорядкувати зібрану інформацію, було створено базу даних.

- 5) Високоякісні дані та статистичні дані були отримані за допомогою методів аналізу даних. Можна було отримати важливу інформацію про профілі студентів і випускників завдяки застосуванню різних аналітичних підходів, таких як розрахунок середнього значення та розробка діаграм і графіків.
- 6) Правильність і достовірність даних перевіряли та оцінювали після отримання результатів. Це включало пошук і виправлення будь-яких потенційних недоліків, дублікатів даних або неповних даних, щоб гарантувати точність і достовірність результатів.

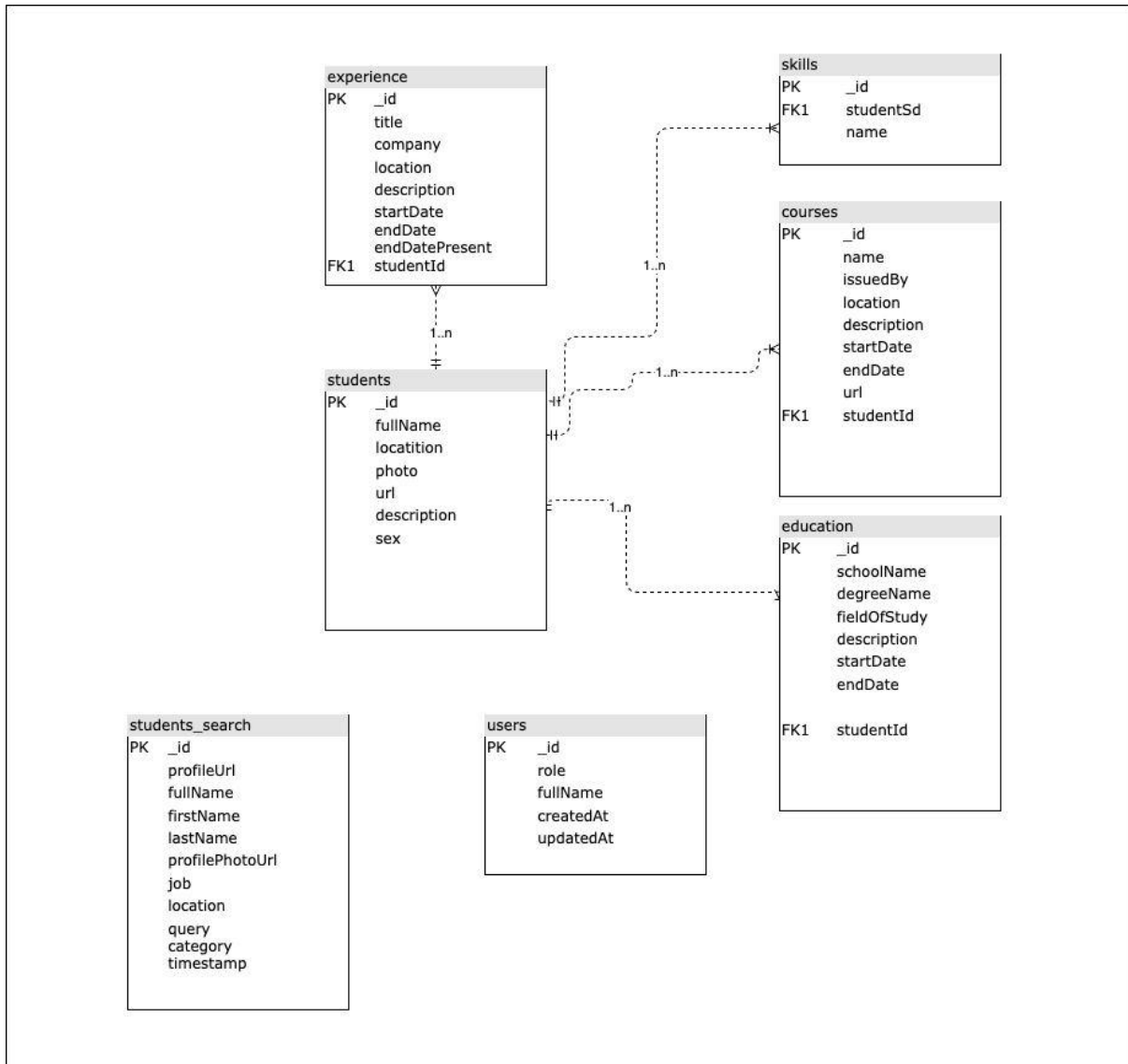
ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. LinkedIn [Електронний ресурс]. — Режим доступу: <http://linkedin.com>
2. Історія Київського університету : монографія / [І. В. Верба та ін. ; редкол. : Л. В. Губерський та ін.] ; Київ. нац. ун-т ім. Тараса Шевченка. - Київ : ВПЦ "Київський університет", 2014. - 895 с..
3. Крокфорд Д. JavaScript: The Good Parts / Дуглас Крокфорд., 2008 - 176 с.
4. Selenium documentation [Електронний ресурс]. — Режим доступу: <https://www.selenium.dev/documentation/>
5. Golang documentation [Електронний ресурс]. — Режим доступу: <https://go.dev/doc/>
6. Донован, Алан А. А., Керниган, Брайан, У. Язык программирования Go = The Go Programming Language. — М.: ООО «И.Д. Вильямс», 2016. — С. 432.
7. Gin documentation [Електронний ресурс]. — Режим доступу: <https://gin-gonic.com/docs/>
8. Суббу Алламараджу RESTful Web Services Cookbook / Суббу Алламараджу., 2010 - 314 с.
9. Introducing JSON [Електронний ресурс]. — Режим доступу: <https://www.json.org/>
10. Елліот Расті Гарольд і В. Скотт Мінз XML in a Nutshell / Елліот Расті Гарольд і В. Скотт Мінз., 2001 — 600 с.
11. Telebot documentation [Електронний ресурс]. — Режим доступу: <https://pkg.go.dev/gopkg.in/tucnak/telebot.v2>
12. MongoDB documentation [Електронний ресурс]. — Режим доступу: <https://docs.mongodb.com/>

13.Роберт Дайно Service Design Patterns: Fundamental Design Solutions for SOAP/WSDL and RESTful Web Services / Роберт Дайно., 2011 - 352 с.

ДОДАТКИ

Додаток А. Діаграма бази даних



Додаток Б. Діаграма прецедентів

