

Міністерство освіти і науки України  
НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ «КИЄВО-МОГИЛЯНСЬКА АКАДЕМІЯ»  
Кафедра мультимедійних систем факультету інформатики

**РОЗРОБКА ПОРТАЛУ ВИПУСКНИКІВ ФАКУЛЬТЕТУ  
ІНФОРМАТИКИ НаУКМА**

**Текстова частина до курсової роботи за спеціальністю  
„Інженерія програмного забезпечення” 121**

Керівник курсової роботи  
ас. Корнійчук М. А.

\_\_\_\_\_  
(підпис)  
“ \_\_\_\_ ” \_\_\_\_\_ 2020 р.

Виконав студент ФІ-4  
Василенко Д. Т.

“ \_\_\_\_ ” \_\_\_\_\_ 2020 р.

Київ 2020

## ЗМІСТ

Анотація .....	7
ВСТУП .....	8
РОЗДІЛ 1. ПОРТАЛ ДЛЯ ВИПУСКНИКІВ ФАКУЛЬТЕТУ ІНФОРМАТИКИ НАУКМА .....	10
1.1 Комунікація випускників .....	10
1.2 Комунікація університету та випускників .....	10
1.3 Веб портал як засіб комунікації між випускниками та університетом .....	10
1.4 Стейкхолдери .....	11
1.5 Система управління контактами факультету Інформатики .....	12
РОЗДІЛ 2. ПРОЕКТУВАННЯ ПОРТАЛУ .....	13
2.1 Вимоги до порталу .....	13
2.1.1 Функціональні вимоги.....	13
2.1.1.1 Реєстрація випускників на порталі .....	13
2.1.1.2 Вхід в систему .....	13
2.1.1.3 Перегляд профілю.....	14
2.1.1.4 Оновлення профілю .....	14
2.1.1.5 Запрошення незареєстрованих випускників .....	14
2.1.1.6 Створення операторів.....	14
2.1.1.7 Інтеграція .....	15
2.1.2 Нефункціональні вимоги.....	15
2.1.2.1 Безпека даних .....	15
2.1.2.2 Масштабованість системи.....	15
2.1.2.3 Доступність.....	15
2.2 Сценарії використання .....	16
2.2.1 Персони порталу .....	16
2.2.2 Діаграми використання .....	17
2.2.2.1 Діаграма використання Випускник.....	17

	3
2.2.2.2 Діаграма використання Оператор .....	18
2.2.2.3 Діаграма використання Адмін .....	18
2.3 Вибір технологій .....	19
2.3.1 Front-end технології .....	19
2.3.1.1 Особливості Angular .....	19
2.3.1.2 Angular Material .....	20
2.3.2 Back-end технології .....	21
2.3.2.1 Особливості Flask .....	21
2.3.2.2 API документація .....	21
2.3.3 База даних .....	21
2.3.3.1 Основні властивості Postgres .....	22
2.3.3.2 Модель «сутність — зв'язок» .....	22
2.3.4 Хмарне рішення для розгортання .....	23
РОЗДІЛ 3. РОЗРОБКА ПОРТАЛУ .....	25
3.1 Моделювання процесів .....	25
3.2 Реалізація процесів Оператора .....	25
3.2.1 Сторінки оператора .....	26
3.2.1.1 Сторінка «Зареєстровані випускники» .....	26
3.2.1.2 Сторінка «Незареєстровані випускники» .....	27
3.2.1.3 Сторінка «Потребує модератції» .....	27
3.3 Реалізація процесів Випускника .....	28
3.3.1 Сторінки Випускника .....	29
3.3.1.1 Сторінки «Реєстрація» та «Логін» .....	29
3.3.1.2 Сторінка «Форма оновлення профілю» .....	30
3.3.1.3 Сторінка «Профіль» .....	31
3.3.1.4 Сторінка «Одногрупники» .....	32
3.3.1.5 Сторінка «Усі випускники» .....	32
ВИСНОВКИ .....	33
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ .....	34

ДОДАТКИ .....	36
---------------	----

Міністерство освіти і науки України  
НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ «КИЇВО-МОГИЛЯНСЬКА АКАДЕМІЯ»  
Кафедра мультимедійних систем факультету інформатики

ЗАТВЕРДЖУЮ  
Зав.кафедри мультимедійних  
систем,  
проф., д. ф.-м.н.  
\_\_\_\_\_ В. В. Бублик  
(підпис)  
“ \_\_\_\_ ” \_\_\_\_\_ 2020 р.

ІНДИВІДУАЛЬНЕ ЗАВДАННЯ  
на курсову роботу

студенту 4 курсу факультету інформатики Василенко Денису Тарасовичу

ТЕМА: Розробка порталу випускників факультету інформатики НаУКМА

Вихідні дані:

- Дослідження можливостей Angular в рамках створення веб порталу
- Дослідження можливостей Flask в рамках побудови API
- Розробка порталу випускників за допомогою Angular та Flask

Зміст ГЧ до курсової роботи:

Індивідуальне завдання

Вступ

1. Портал для випускників факультету інформатики НаУКМА
2. Проектування порталу
3. Розробка порталу

Висновки

Список використаних джерел

Дата видачі “ \_\_\_\_ ” \_\_\_\_\_ 2020 р.

Керівник \_\_\_\_\_  
(підпис)

Завдання отримав \_\_\_\_\_  
(підпис)

Тема: Впровадження системи управління контактами факультету інформатики

Календарний план виконання роботи

№ п/п	Назва етапу курсової роботи	Термін виконання етапу	Примітка
1.	Отримання завдання на курсову роботу.	08.10.2019	
2.	Огляд технічної літератури за темою роботи.	31.10.2019	
3.	Виконати аналіз існуючих досліджень за темою	11.11.2019	
3.	Дослідити можливі рішення для проблеми збору контактів	02.12.2019	
5.	Аналіз можливостей та архітектури веб порталів	23.12.2019	
6.	Написання першого та другого розділів	13.01.2020	
7.	Розробка веб порталу	29.02.2020	
8.	Написання третього розділу	09.03.2020	
9.	Створення слайдів для доповіді та написання доповіді.	16.03.2020	
10.	Аналіз виконаної роботи з керівником.	20.03.2020	
11.	Коригування роботи.	25.03.2020	
12.	Остаточне оформлення пояснювальної роботи та слайдів.	30.03.2020	
13.	Захист курсової роботи	14.04.2020	

Студент Василенко Д. Т.

Керівник Корнійчук М. А.

“ \_\_\_\_\_ ”

## Анотація

Мета курсової роботи створити портал випускників факультету інформатики, який має покращити комунікацію серед випускників та випускників з університетом, після закінчення навчання. Робота складається з трьох розділів. Перший розділ присвячено дослідженню питання комунікації в рамках факультету інформатики НаУКМА, другий – описує процес підготовки та проектування порталу, третій – показує кінцевий продукт зі сторони операторів системи та випускників. Результатом курсової роботи стало створення порталу випускників факультету інформатики та закладено підґрунтя для розширення порталу на інші факультети або розгортанню системи в інших вузах.

## ВСТУП

Щороку десятки випускників завершують своє навчання на факультеті інформатики. Проте не кожному з них буде легко знайти своє місце в ІТ сфері, в умовах жорсткої конкуренції. Саме тому виникла ідея створити портал випускників факультету, який збере випускників різних спеціальностей та років навчання в одному місці. Це і є мета роботи.

Запуск такої платформи буде корисним як для випускників, так і для університету. Випускники зможуть відшукати корисні контакти для працевлаштування або ж знайти потенційних співробітників серед випускників своєї альма-матер. А для університету це буде зручним інструментом для зв'язку з випускниками та відслідковування їхньої успішності, адже це є вагомим чинником якості роботи вищого навчального закладу.

Постановка задачі:

- Дослідження можливостей Angular в рамках створення веб порталу
- Дослідження можливостей Flask в рамках побудови API
- Розробка порталу випускників за допомогою Angular та Flask

Роботу поділено на три розділи.

Перший розділ присвячений підготовці до початку роботи над порталом. Було знайдено потенційних стейкхолдерів, які зацікавлені в реалізації проекту та налагоджено комунікацію з розробниками системи управління контактами факультету інформатики. Це система, що створена на основі CRM Odoo та буде джерелом контактів для порталу.

Другий розділ освітлює підготовку до початку написання коду. В ньому показані діаграми використання, перелічено функціональні та нефункціональні вимоги,



обрану технології для розробки front-end та back-end частин. Створено модель «сутність-зв'язок».

Третій розділ присвячено реалізації кінцевого продукту зі сторони випускника та оператора. В цьому розділі показані кроки, які потрібні для реєстрації на порталі, запрошенню незареєстрованих випускників, оновленню профілю.

В результаті роботи створено веб платформу, яка реалізує запланований функціонал та дозволяє керувати порталом випускників факультету інформатики НаУКМА.

## РОЗДІЛ 1. ПОРТАЛ ДЛЯ ВИПУСКНИКІВ ФАКУЛЬТЕТУ ІНФОРМАТИКИ НАУКМА

### 1.1 Комунікація випускників

Протягом навчання в університеті студенти стають частиною однієї спільноти. Розділяють спільні цінності, діляться досвідом та допомагають один одному. Проте після завершення навчання, вони втрачають місце яке їх поєднує – університет.

### 1.2 Комунікація університету та випускників

Для університетів випускники є важливим показником якості освіти та актуальності навчальних програм. Саме тому адміністрація вищих навчальних закладів зацікавлена в зборі та обробці інформації про випускників: місце роботи, досягнення, тощо. Зараз зв'язок з випускниками підтримує відділ по роботі з випускниками НаУКМА. Вони використовують Google Forms на <https://alumni.ukma.edu.ua/ankety.html> та збирають дані в таблицях. Такому підходу бракує систематичності та централізованості, тому працівники деканату або викладачі не мають швидкого доступу до цих даних.

### 1.3 Веб портал як засіб комунікації між випускниками та університетом

В сучасному світі наявність цінних та перевірених знайомств вкрай необхідна. Соціальні мережі цілком задовольняють потребу у спілкуванні, проте їхня надмірна глобалізація не дає змогу створити швидкий та зрозумілий доступ до контактів. Саме тому виникає потреба створення порталу для випускників факультету інформатики НаУКМА. Такий портал дозволить підтримувати зв'язок між студентами після випуску, створить можливості для пошуку

роботодавців або працівників. А для адміністрації це буде засобом для збору та модерації контактів випускників в зручному застосунку. Платформа буде інтегрована з системою управління контактами факультету інформатики, що дозволить створити централізовану систему для різних ланок факультету.

#### 1.4 Стейкхолдери

У реалізації даного порталу зацікавленні декілька груп (Рис 1). Метою ж розробки є забезпечення потреб кожної з них.

- Андрій Глибовець (високий пріоритет) - система обліку контактів людей готових допомогти у покращенні ІТ-освіти в Україні контректно для факультету Інформатики.
- Шумкова Наталія (високий пріоритет) - система обліку контактів випускників для залучення їх у заходах та фінансовій підтримці університету
- Випускники - система обміну контактними даними між випускниками та долучення до підтримки університету
- Працівники деканату та викладачі - система обліку контактів випускників та студентів
- Оператори системи - система обліку контактів випускників та студентів.



Рис 1

## 1.5 Система управління контактами факультету Інформатики

Портал випускників пов'язаний з іншим проектом, а саме системою управління контактами (Odoo). Контакти будуть зберігатись на стороні Odoo, їх отримання буде доступне лише для операторів, а рядові користувачі порталу, зможуть бачити лише зареєстрованих випускників, тобто таких, що зареєструвались та дали згоду на публікування своїх даних. А редагувати дані напряму в Odoo випускники не зможуть, саме тому створено процес створення заявок на оновлення даних. Таким чином ми налагодили отримання даних з системи управління контактами та забезпечили їхню цілісність, завдяки тому що доступ до модерації будуть мати лише оператори.

## РОЗДІЛ 2. ПРОЕКТУВАННЯ ПОРТАЛУ

### 2.1 Вимоги до порталу

Вимоги до порталу сформовані в ході дослідження проблеми комунікації співробітників НаУКМА та випускниками та описані у цьому розділі, як функціональні та нефункціональні вимоги. Збір вимог був проведений шляхом інтерв'ю з деканом факультету інформатики Глибовцем А. М. та викладачем Корнійчуком М. А.

#### 2.1.1 Функціональні вимоги

Функціональні вимоги описують внутрішню роботу системи, її поведінку: калькулювання даних, маніпулювання даними, опрацювання даних, і інші специфічні функції які повинна виконувати система. [1]

##### 2.1.1.1 Реєстрація випускників на порталі

Випускник може зареєструватись в системі отримавши лист запрошення, яке ідентифікує його в системі. Вказавши пароль та електронну пошту, він зможе підтвердити реєстрацію та увійти в систему.

##### 2.1.1.2 Вхід в систему

Вхід в систему реалізовано для двох груп користувачів: Оператори та Випускники. Для авторизації в системі потрібно використати логін та пароль. Для випускників які заходять в систему вперше буде показана сторінка з формою на оновлення профілю.

### 2.1.1.3 Перегляд профілю

Випускники можуть переглянути профіль та створити заявку на редагування даних. Також вони можуть переглянути профілі інших зареєстрованих користувачів, які відкрили доступ до своїх контактів. Доступна фільтрація за факультетом, спеціальністю, роком вступу та випуску.

### 2.1.1.4 Оновлення профілю

Для оновлення профілю випускнику потрібно змінити поточні дані про себе та надіслати заявку, яка буде опрацьована оператором системи. Оператор системи може підтвердити, відхилити або відредагувати заявку. Після підтвердження заявки, дані будуть відправлені до системи управління контактами та зміни будуть застосовані до профілю випускника.

### 2.1.1.5 Запрошення незареєстрованих випускників

Оператор може отримати дані з системи управління контактами з незареєстрованими випускниками та сформулювати посилання-запрошення. За цим посиланням випускник зможе зареєструватись на порталі. Також оператор може змінити статус запрошення для незареєстрованого випускника: запрошено, немає відповіді, відхилено.

### 2.1.1.6 Створення операторів

В системі будуть супер-оператори, які зможуть створювати звичайних операторів.

### 2.1.1.7 Інтеграція

Портал повинен бути інтегрованим з системою управління контактами факультету інформатики НаУКМА.

### 2.1.2 Нефункціональні вимоги

Не функціональні вимоги можна поділити на дві категорії: покращення (безпека, надійність, швидкодія, зручність у використанні ...) та вдосконалення (масштабування, відновлюваність ...) властивостей системи. [1]

#### 2.1.2.1 Безпека даних

Дані користувачів повинні бути захищені, тому паролі будуть зберігатись захешованими. Обробка даних буде здійснюватися лише за згодою користувачів. Транспортування даних буде згідно протоколу HTTPS.

#### 2.1.2.2 Масштабованість системи

Розробка має бути виконана зрозуміло для подальшого удосконалення та розширення.

#### 2.1.2.3 Доступність

Доступ до порталу повинен бути лише для зареєстрованих користувачів. Користувацький інтерфейс повинен бути зрозумілим та дружелюбним, відповідати принципам доступності.

## 2.2 Сценарії використання

**Сценарій використання** (*Use Case*) — у розробці програмного забезпечення та системному проектуванні це опис поведінки системи, як вона відповідає на зовнішні запити, вказує на те «хто» і «що» може зробити з розглянутою системою. Методика різновидів використання застосовується для виявлення вимог до поведінки системи, відомих також як функціональні вимоги. [2] Для створення сценаріїв було підготовлено список користувачів системи, так званих «персон». Для кожного типу користувачів було створено діаграму з сценаріями користування системою, відповідно до їхньої ролі в системі.

### 2.2.1 Персоналі порталу

#### 1. Оператор системи (студент або працівник університету)

Джерелом контактів будуть ехе1-таблиці, соціальні мережі, дані від адміністрації університету. Вимогами до студента будуть базові навички користування комп'ютером. Модерувати вхідні заявки на оновлення профілю користувачів. Має доступ до зареєстрованих та незареєстрованих випускників, може створювати посилання-запрошення, змінювати статус запрошення випускника. Модерувати вхідні заявки на оновлення профілю.

#### 2. Головний Адмін системи

Андрій Миколайович Глибовець. Надає права іншим операторам. Займається обробкою контактів.

#### 3. Зовнішні користувачі системи

Випускники факультету Інформатики. Можуть зареєструватись чи авторизуватись в системі за допомогою електронної адреси та оновити свій профіль. Дати дозвіл на поширення свого профілю серед інших випускників в системі, а також отримати доступ до профілів інших випускників.



## 2.2.2 Діаграми використання

Для кращої наочності було створено діаграму використання для кожного типу користувача. Така діаграма показує відношення між акторами та прецедентами в системі. [3]

### 2.2.2.1 Діаграма використання Випусник

Діаграму використання для випусників показано на Рис 2.

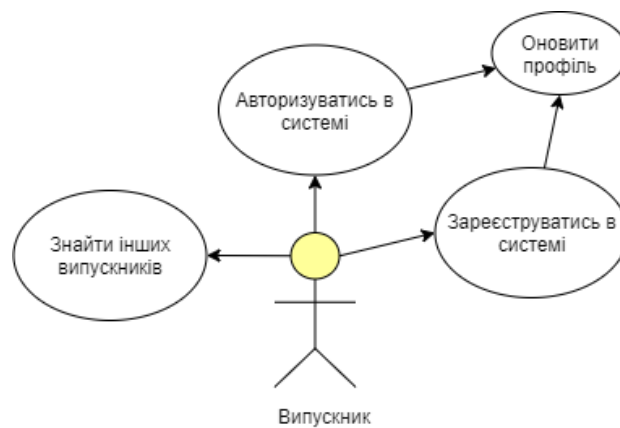


Рис 2

### 2.2.2.2 Діаграма використання Оператор

Діаграму використання для випускників показано на Рис 3.

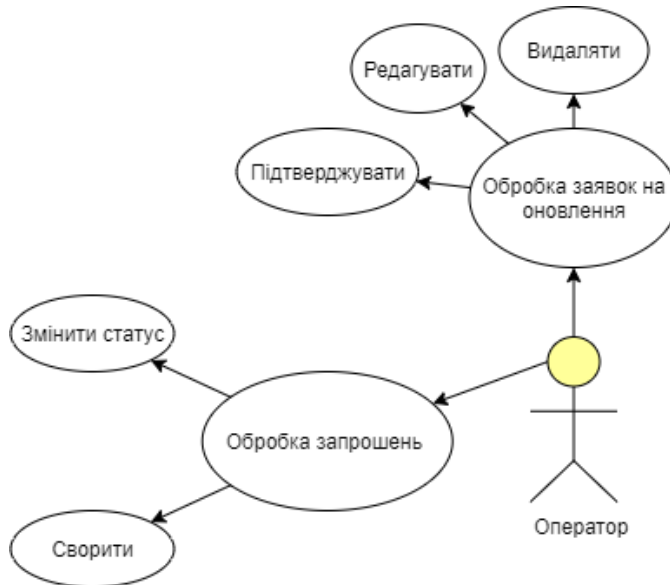


Рис 3

### 2.2.2.3 Діаграма використання Адмін

Діаграму використання для випускників показано на Рис 3.

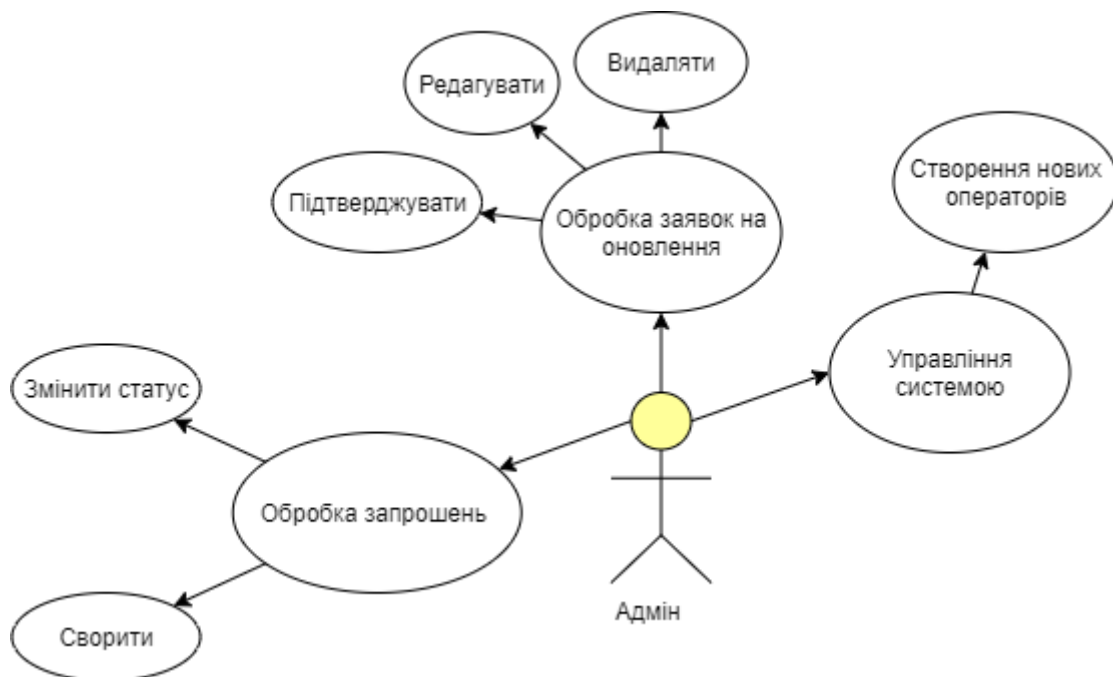


Рис 4

## 2.3 Вибір технологій

Портал випускників заплановано як веб застосунок з back-end та front-end частин. Front-end частина буде виконана як SPA та буде побудована незалежно від back-end. Back-end частина буде відповідати за створення API, збереження даних в базі даних та інтеграції з системою керування контактами (Odoо).

### 2.3.1 Front-end технології

Для реалізації front-end сторони обрано фреймворк Angular 8, оскільки цей фреймворк дозволяє створити SPA.

#### 2.3.1.1 Особливості Angular

Angular - це платформа для веб-розробок, яка надає розробникам надійні інструменти для створення клієнтської сторони веб-додатків за допомогою мови TypeScript. Випущений в 2010 році і раніше відомий як AngularJS, Angular - це JavaScript фреймворк, який використовується для створення односторінкових веб-додатків. [4]

Далі буде наведено ключові переваги Angular, які вплинули на вибір даного фреймворку.

- **Документація**

Angular має чітку та повну документацію, це дуже важливо, оскільки проект платформи заплановано, як довготривалий, розробникам, які продовжать роботу на ним, буде простіше знайти всю необхідну інформацію. Також для Angular написано багато статей, які покривають імплементацію більшості розповсюджених задач.

- **Підтримка Google**

Підтримка від найбільшої технологічної корпорації забезпечує фреймворк надійністю та впевненість в довготривалій підтримці Angular.

- **Архітектура компонентів**

Відповідно до цієї архітектури додаток розділений на незалежні логічні та функціональні компоненти. Компонент містить HTML шаблон, який відповідає за відображення та клас з декоратором, що контролює логіку відображення. Компоненти використовують сервіси, які надають певні функціональні можливості, безпосередньо не пов'язані з відображенням.[5] Крім того, незалежність компонентів робить код модульним, багаторазовим та ефективним.

- **АОТ компілятор**

Компілятор АОТ компілює TypeScript та HTML в JavaScript на етапі збирання проекту. Це означає, що код компілюється до того, як браузер завантажить сторінку з веб-застосунком, таким чином відображення сторінки відбудеться набагато швидше.

- **CLI**

Командний інтерфейс Angular дозволяє створювати готові шаблонні конструкції, дозволяє запускати різні сценарії для тестування, побудови та розробки проекту. Це робить процес розробки автоматизованим та швидким.

### 2.3.1.2 Angular Material

Для прискорення процесу розробки користувацького інтерфейсу було вирішено використовувати Angular Material, що слідує принципам Material Design.

Material Design - це мова дизайну, розроблена та випущена Google у 2014 році. Вона диктує набір правил, якого повинні дотримуватися всі програми, що використовують Матеріал дизайн. Оскільки Google також розробляє Angular, підтримка Material Design в рамках Angular на високому рівні, всі особливості Material втілено в бібліотеці Angular Material. [6]

### 2.3.2 Back-end технології

Для реалізації back-end частини було використано Python фреймворк Flask.

#### 2.3.2.1 Особливості Flask

Flask – це WSGI фреймворк для веб-додатків, що призначений для швидкого та легкого початку роботи з можливістю масштабування до складних програм. Він розпочався як простий обгортка навколо Werkzeug та Jinja, але згодом став однією з найпопулярніших Python бібліотек.

Flask підтримує використання залежностей супутніх бібліотека, але не вимагає застосування будь-яких залежностей та макетів проекту. Розробникам надана свобода вибору рішень та застосувань, а Flask спільнота пропонує багато розширень, які полегшують додавання нових функціональних можливостей. [7]

#### 2.3.2.2 API документація

Для формування API документації обрано Flask-RESTPlus. Це розширення для Flask, яке додає підтримку для швидкого створення API REST. Flask-RESTPlus пропонує якісні рішення з мінімальними налаштуваннями. Він надає цілісну колекцію декораторів та інструментів для опису вашого API та належної експозиції його документації (за допомогою Swagger). [8] Створену документацію можна отримати за посиланням <https://alumni-service-api.herokuapp.com/api/v1/docs>.

### 2.3.3 База даних

База даних створена за допомогою PostgreSQL. PostgreSQL - це система управління базами даних та об'єктно-реляційна система, одна з найбільших систем з відкритим кодом.

PostgreSQL вимагає мінімальних витрат ресурсів, тому підтримка створеної бази коштує менше ніж в конкурентів Postgres. [9]

### 2.3.3.1 Основні властивості Postgres

PostgreSQL має багато вбудованих функцій, такі як:

- Типи, визначені користувачем
- Таблиця спадкування
- Замикаючий механізм
- Цілісність зовнішніх ключів
- Відображення, правила, підзапити
- Вкладені транзакції (точки збереження)

### 2.3.3.2 Модель «сутність — зв'язок»

Модель «сутність-зв'язок» (ER-модель) (англ. Entity-relationship model або entity-relationship diagram) — модель даних, яка дозволяє описувати концептуальні схеми за допомогою узагальнених конструкцій блоків. ER-модель — це мета-модель даних, тобто засіб опису моделей даних. Існує ряд моделей для представлення знань, але одним з найзручніших інструментів уніфікованого представлення даних, незалежного від програмного забезпечення що його реалізує, є модель «сутність-зв'язок». [10] Саме тому в даному проекті також було застосовано такий підхід. (Рис 5).

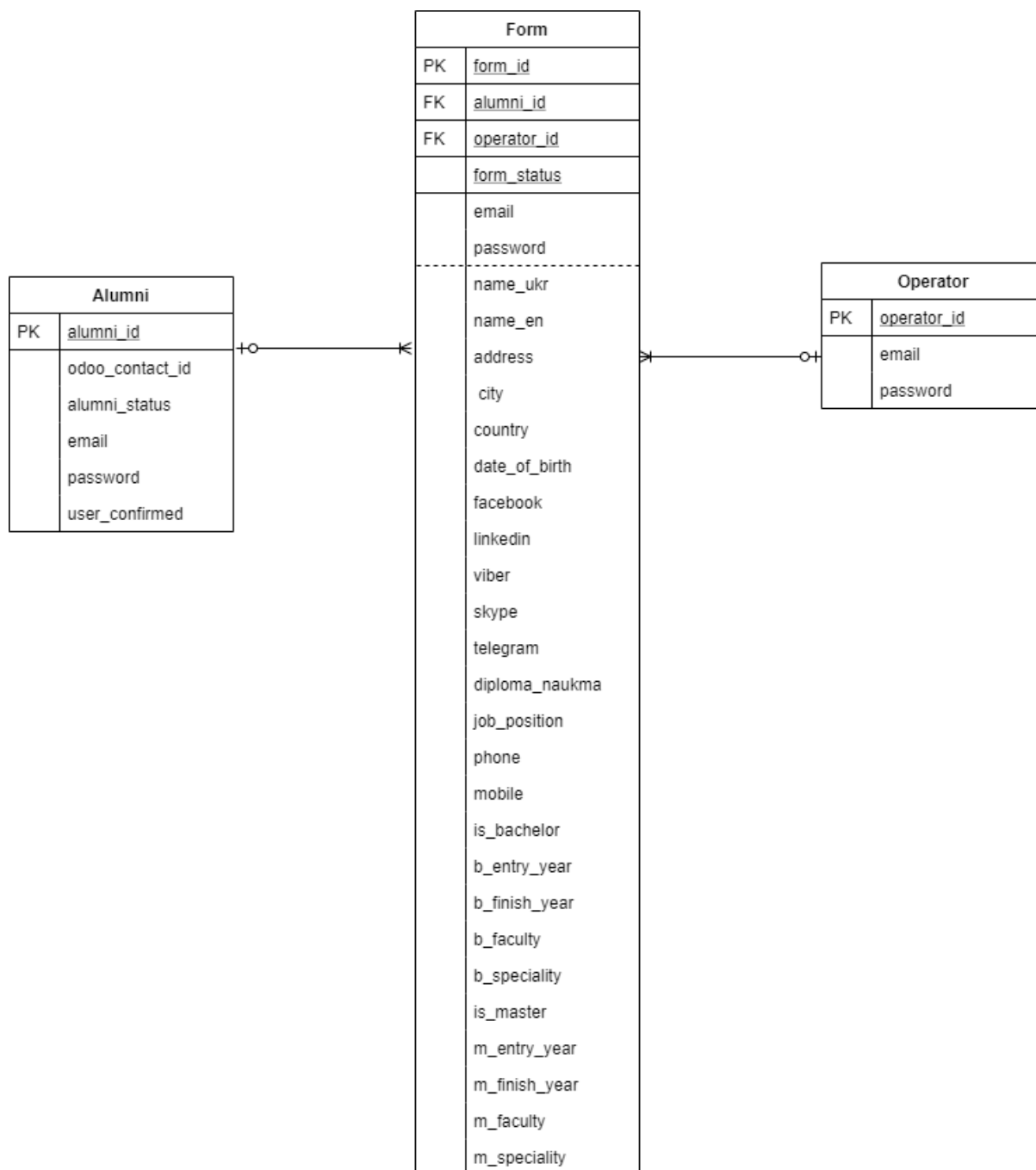


Рис 5

#### 2.3.4 Хмарне рішення для розгортання

Для розгортання тестового середовища було використано хмарний сервіс Heroku. Вибір Heroku обумовлений безкоштовністю сервісу, простотою використання, швидкістю розгортання та підтримкою усіх трьох складових

проекту: front-end Angular, back-end Flask та DB Postgres. Також за допомогою GitHub та Heroku було налаштовано процес автоматичного розгортання проекту. Проект автоматично збирається та запускається на серверах Heroku, після внесення змін до гілки master. (Рис 6)

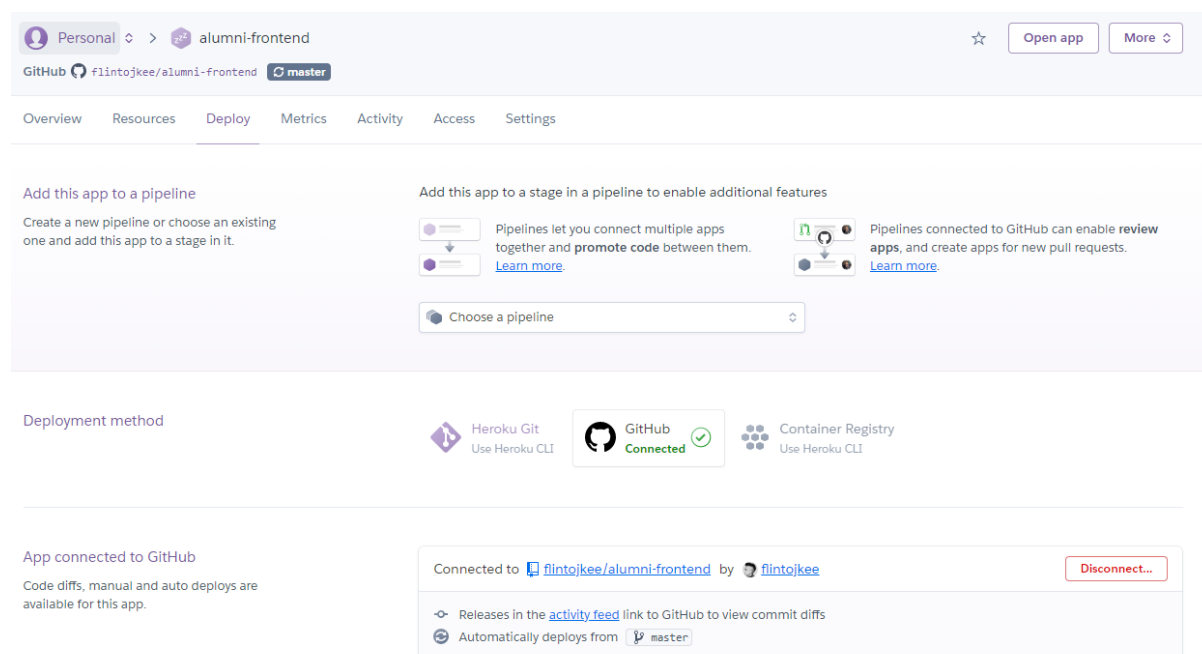


Рис 6



## РОЗДІЛ 3. РОЗРОБКА ПОРТАЛУ

### 3.1 Моделювання процесів

Перед початком написання коду потрібно остаточно було визначити процеси за якими буде відбуватись взаємодія користувачів з платформою. Для опису процесів було створено BPMN діаграму, в якій відображено шлях реєстрації випускника на порталі. Діаграма розділена на дві частини Випускник та Оператор, дії кожного знаходяться у відповідній частині діаграми, а при взаємодії Випускника та Оператора шлях процесів переходить від одного учасника діаграми до іншого.

### 3.2 Реалізація процесів Оператора

Оператор буде мати особливі привілеї відмінні від звичайного користувача, тому доцільно створити окрему точку входу для оператора. Як показано в ER-моделі, у оператора є поля електронна пошта (відіграє роль логіну) та пароль. Сторінку входу оператора зображено на Рис 7.

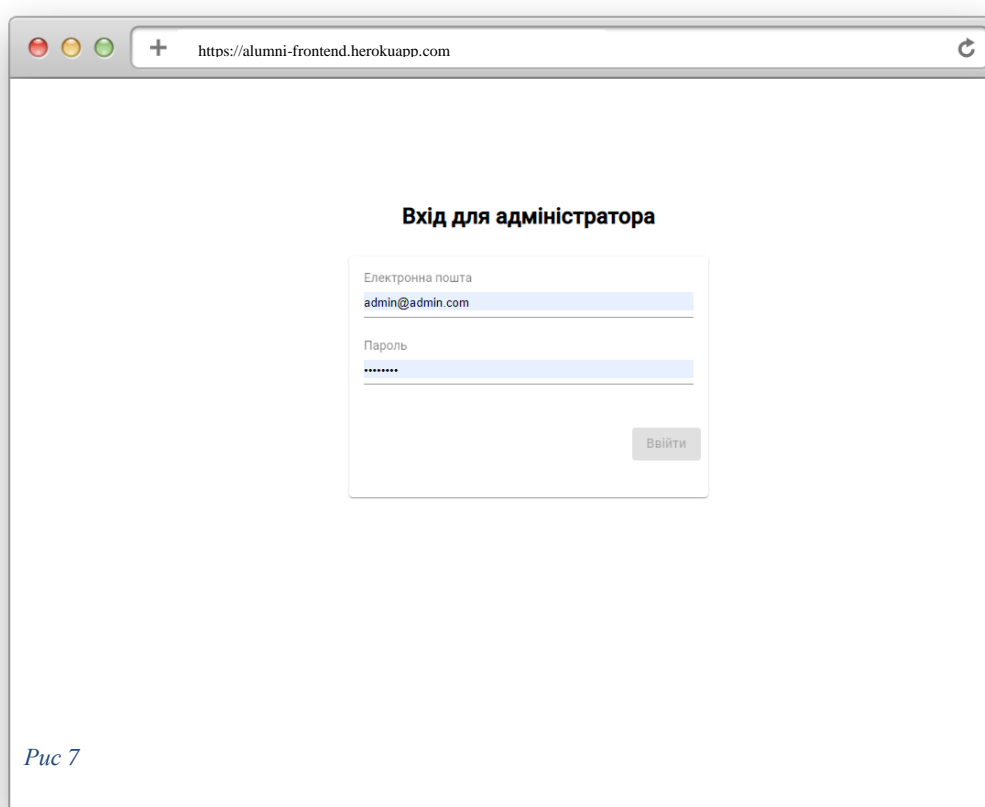


Рис 7

### 3.2.1 Сторінки оператора

Одразу після авторизації оператора доступні три сторінки: «Зареєстровані випускники», «Незареєстровані випускники» та «Потребує модерації».

#### 3.2.1.1 Сторінка «Зареєстровані випускники»

Сторінка «Зареєстровані випускники» (Рис 8) містить список з зареєстрованими випускниками на порталі, тобто таких що отримали листа запрошення в систему та зареєстрували свою електронну пошту. Випускник має статус «Підтверджено», якщо пошту було підтверджено та «Не підтверджено» у випадку, якщо випускник не підтвердив свою електронну пошту, таким чином сигналізуючи, що процес реєстрації почався проте не закінчився.

Картки випускників можна фільтрувати за факультетом, спеціальністю, роком вступу та випуску по бакалаврській або магістерській програмі.

The screenshot shows a web browser window with the URL `https://alumni-frontend.herokuapp.com`. The page header includes the logo of the community and the text "Спільнота випускників НАУКМА" along with the user role "admin (Оператор)" and a "Вийти" button.

The main content area is titled "Зареєстровані випускники" and features two tabs: "Бакалаврська програма" and "Магістерська програма". Below the tabs are filter menus for "Факультет", "Рік вступу", "Спеціальність", and "Рік випуску".

The list of graduates includes the following entries:

Ім'я	Посада	Електронна пошта	Бакалавр	Магістр	Статус
Denys Vasylenjko			Не вказано	Не вказано	Підтверджено
Marc Demo		mark.brown23@example.com	Не вказано	Не вказано	Не підтверджено
Корнійчук Максим Анатолійович	Викладач	m.kornichuk@ukma.edu.ua	Не вказано	Факультет інформатики	Підтверджено

Рис 8

### 3.2.1.2 Сторінка «Незареєстровані випускники»

Сторінка «Незареєстровані випускники» (Рис 9) містить список з незареєстрованими випускниками. Цей список автоматично завантажується з системи управління контактами. Усі нові контакти з позначкою «Випускник» будуть відображатись як незареєстровані випускники. Картку випускника можна розгорнути та отримати доступ до взаємодій з випускником.

Найважливіша функція це генерація запрошення. Запрошення – це посилення з унікальним токеном, яке буде ідентифікувати конкретного випускника та має термін дії. Це зроблено для того, щоб прив'язати сторінку реєстрації до випускника та забезпечити захист даних зі сторони системи управління контактами. Згенероване запрошення можна передати випускникові у будь-який зручний спосіб. Також присутні тригери на зміну стану картки на випадок якщо випускник не відповідає, тоді це буде сигналом до пошуку іншого каналу зв'язку, або якщо випускник відмовився від участі на порталі. Кнопка «Odo profile» слугує переходом на сторінку конкретного випускника в системі управління контактами.

Картки випускників можна фільтрувати за факультетом, спеціальністю, роком вступу та випуску по бакалаврській або магістерській програмі. А також доступна фільтрація за станом запрошення випускника. За замовченням будуть відображені всі випускники.

### 3.2.1.3 Сторінка «Потребує модерації»

Ця сторінка містить заявки на оновлення профілю від випускників. На даній сторінці оператору будуть проводити модерації форм оновлення, для забезпечення цілісності та верифікації даних, які потім будуть передані в систему управління контактами, на цьому кроці буде завершено процес «Реєстрації та оновлення даних випускника».

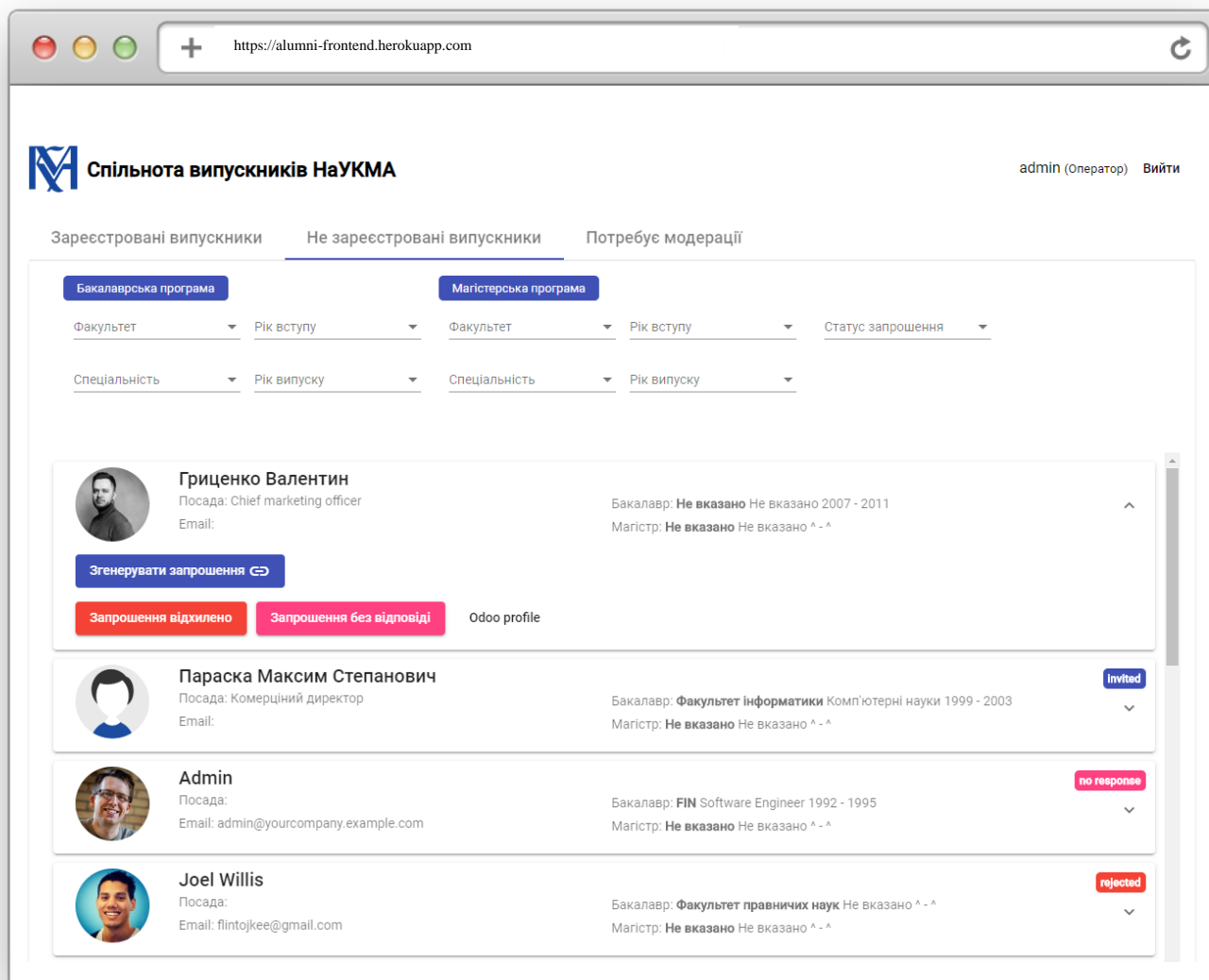


Рис 9

### 3.3 Реалізація процесів Випускника

Самостійна реєстрація на порталі випускників не передбачена, оскільки дані беруться з системи управління контактами, тобто ми наперед знаємо хто може стати учасником portalу, потрібна реалізація системи запрошень. Таке запрошення випускники зможуть отримати від операторів.

### 3.3.1 Сторінки Випускника

Для випускника доступні 6 сторінок. Відправною точкою в роботі з платформою є реєстрація або логін, без цих кроків доступу до платформи у випускника не буде. Після авторизації випускник побачить сторінки з іншими зареєстрованими випускниками та зможе відкрити їхні профілі, для отримання контактних даних.

#### 3.3.1.1 Сторінки «Реєстрація» та «Логін»

Після переходу за покликанням з електронного листа, випускник повинен заповнити поля електронної пошти, паролю та підтвердження паролю. На цьому етапі пошта проходить валідацію на стороні клієнта за допомогою регулярного виразу, а пароль повинен містити як мінімум 1 букву у верхньому регістрі, 1 у нижньому, 1 цифру та загалом не менше 6 символів. Робота з формами відбувається за допомогою реактивного підходу (вбудовано в Angular), саме тому зміни відображаються динамічно, а не після взаємодії з кнопкою відправки форми. Окрім цього електронна пошта потребує підтвердження, тому випускник отримає лист-підтвердження на вказану пошту. (Рис 10).

Для входу в систему потрібно вказати логін (електронну пошту) та пароль, після цього випускникові буде надано доступ до порталу.

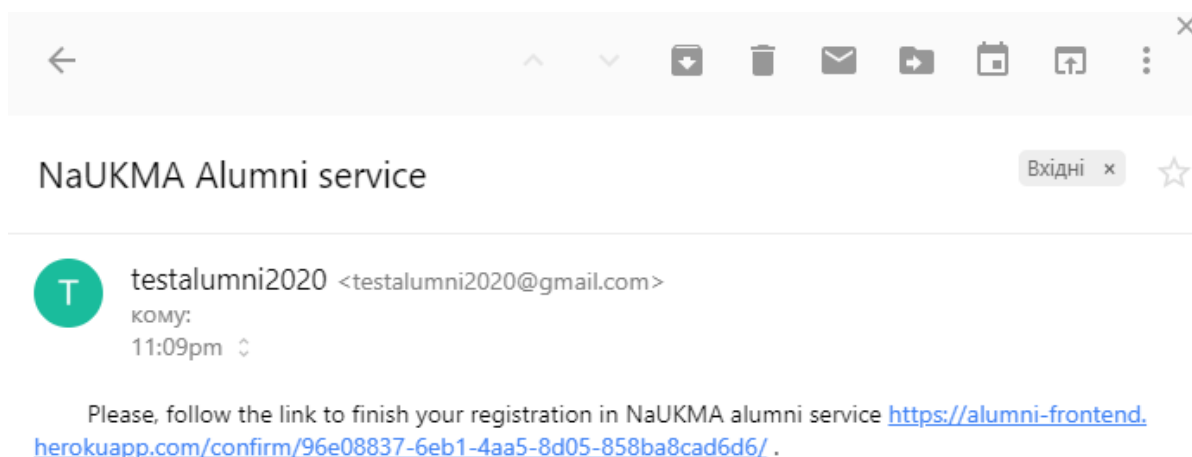


Рис 10

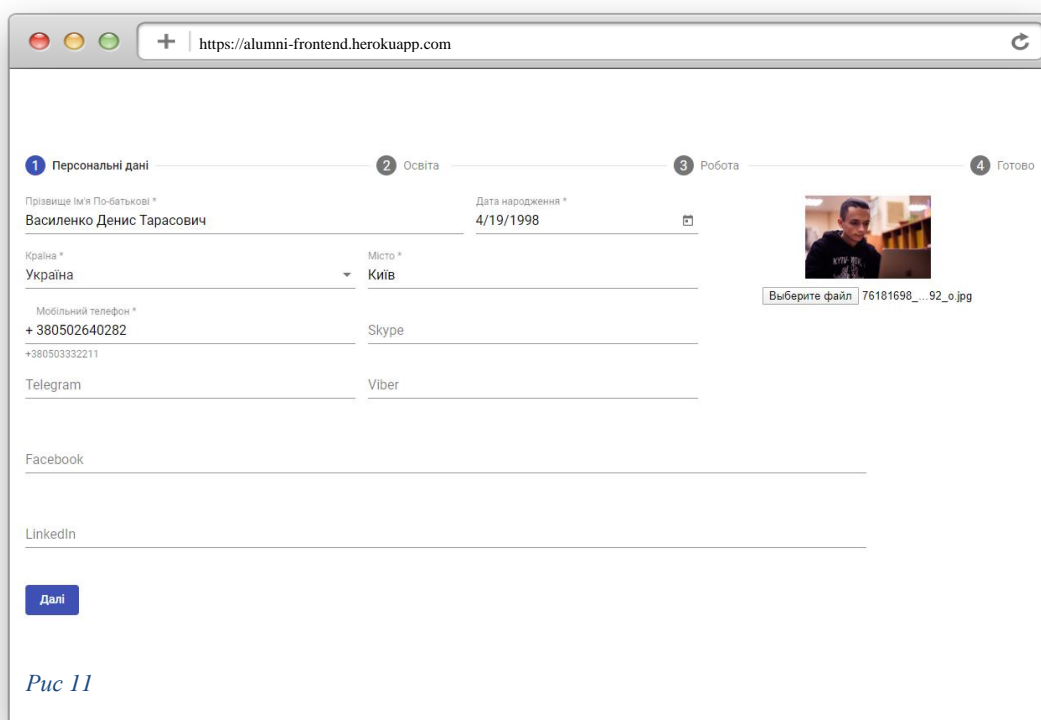
### 3.3.1.2 Сторінка «Форма оновлення профілю»

Ця сторінка буде доступна одразу після завершення реєстрації, а саме підтвердження електронної пошти. Випускник отримує форму для оновлення своїх даних. Частина даних з них може бути наявна в системі управління контактами (Odoо), тому поля з цими даними будуть заповнені, а випускник зможе або залишити як є, або оновити інформацію, якщо вона невірна. Це зроблено для полегшення та пришвидшення проходження форми.

Форма розділена на три частини:

- Персональні дані (Рис 11)

На цьому кроці форми випускнику потрібно заповнити загальні дані про себе, а також вказати покликання на будь-яку з соціальних мереж для зв'язку. Обов'язкові поля відмічені зірочкою, без їхнього заповнення неможливо буде перейти на наступний крок.



The screenshot shows a web browser window with the URL <https://alumni-frontend.herokuapp.com>. The page displays a multi-step form for updating a profile. The first step, 'Персональні дані' (Personal Data), is active. The form includes the following fields:

- Прізвище (ім'я По-батькові) \***: Василенко Денис Тарасович
- Дата народження \***: 4/19/1998
- Країна \***: Україна
- Місто \***: Київ
- Мобільний телефон \***: +380502640282
- Skype**: +380503332211
- Telegram**
- Viber**
- Facebook**
- LinkedIn**

There is a photo upload section with a placeholder image and the text 'Выберите файл 76181698...92\_o.jpg'. A blue 'Далі' (Next) button is located at the bottom left of the form.

*Рис 11*

- Освіта (Рис 12)

На цьому кроці потрібно заповнити інформацію про здобуту освіту. Важливо, що відмітивши наявність бакалаврської чи магістерської програми, поля факультет, спеціальність та рік вступу стають обов'язковими.

Рис 12

- Робота

На цій сторінці користувач може заповнити дані про місце роботи та посаду. Усі раніше внесені компанії завантажуються, тому якщо компанія випускника вже є, користувач може обрати її з випадуючого списку.

### 3.3.1.3 Сторінка «Профіль»

На цій сторінці відображено профіль випускника та всі наявні дані про нього. Зайшовши на цю сторінку випускник може створити заявку на оновлення профілю та відправити її на модерацію, як тільки форма буде затверджена оператором, дані оновляться.

#### 3.3.1.4 Сторінка «Одногрупники»

Аналогічна до сторінки всі випускники, створена для швидкого доступу до тих випускників, які були одногрупниками до поточного користувача. Одногрупниками вважаються всі ті хто закінчив навчання по бакалаврській чи магістерській програмі на одному факультеті та одній спеціальності.

#### 3.3.1.5 Сторінка «Усі випускники»

Ця сторінка містить список усіх зареєстрованих випускників та фільтр за факультетом, спеціальністю, роком вступу та випуску. Список було реалізовано через віртуальний скролінг та пагінацію, щоб покращити продуктивності. Так коли сторінка завантажується на сервер йде запит лише по перший 10 випускників, далі коли користувач гортає список, підвантажуються та кешуються наступні 10, при цьому віртуальний скролінг залишає в структур HTML лише видимих користувачів.



## ВИСНОВКИ

В ході роботи досліджено проблему комунікації випускників та університету. Створено портал випускників факультету інформатики з двома типами користувачів: операторами та випускниками. Ця робота показала, що процеси співпраці з випускниками можуть бути якісно покращені шляхом створення подібних платформ або оновлення існуючих. Джерелом даних стала система управління контактами факультету інформатики. Інтеграція з цією системою дозволила відокремити логіку збереження та обробки контактів від платформи випускників, такий підхід дозволяє в майбутньому використати іншу систему для інтеграції забезпечуючи універсальність платформи. Обрані технології для розробки цілком себе виправдали, адже модульність системи дозволяє покращувати її та доповнювати. Тому цей проект може бути поширеним серед інших факультетів або навіть університетів.

## СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Функціональні та Не Функціональні Вимоги [Електронний ресурс] – Режим доступу до ресурсу: [http://lvivqaclub.blogspot.com/2008/10/blog-post\\_17.html](http://lvivqaclub.blogspot.com/2008/10/blog-post_17.html).
2. Сценарій використання [Електронний ресурс] – Режим доступу до ресурсу: [https://uk.wikipedia.org/wiki/Сценарій\\_використання](https://uk.wikipedia.org/wiki/Сценарій_використання).
3. Діаграма прецедентів [Електронний ресурс] – Режим доступу до ресурсу: [https://uk.wikipedia.org/wiki/%D0%94%D1%96%D0%B0%D0%B3%D1%80%D0%B0%D0%BC%D0%B0\\_%D0%BF%D1%80%D0%B5%D1%86%D0%B5%D0%B4%D0%B5%D0%BD%D1%82%D1%96%D0%B2](https://uk.wikipedia.org/wiki/%D0%94%D1%96%D0%B0%D0%B3%D1%80%D0%B0%D0%BC%D0%B0_%D0%BF%D1%80%D0%B5%D1%86%D0%B5%D0%B4%D0%B5%D0%BD%D1%82%D1%96%D0%B2).
4. Angular: Best Use Cases [Електронний ресурс] – Режим доступу до ресурсу: <https://yalantis.com/blog/when-to-use-angular/>.
5. Introduction to Angular concepts [Електронний ресурс] – Режим доступу до ресурсу: <https://angular.io/guide/architecture>.
6. How to Use Angular Material to Build an Appealing UI [Електронний ресурс] – Режим доступу до ресурсу: <https://medium.com/better-programming/how-to-use-angular-material-to-build-an-appealing-ui-927a8b426676>.
7. Flask [Електронний ресурс] – Режим доступу до ресурсу: <https://github.com/pallets/flask>.
8. Flask rest [Електронний ресурс] – Режим доступу до ресурсу: <https://flask-restplus.readthedocs.io/en/stable/>.
9. What is Postgresql [Електронний ресурс] – Режим доступу до ресурсу: <https://www.postgresqltutorial.com/what-is-postgresql/>
10. Модель «сутність – зв'язок» [Електронний ресурс] – Режим доступу до ресурсу: [https://wiki.cuspu.edu.ua/index.php/%D0%9C%D0%BE%D0%B4%D0%B5%D0%BB%D1%8C\\_%C2%AB%D1%81%D1%83%D1%82%D0%BD%D1%96%D1%81%D1%82%D1%8C\\_%E2%80%93%D0%B7%D0%B2%E2%80%99%D1%8F%D0%B7%D0%BE%D0%BA%C2%BB.\\_%D0%9E%D1%81%D0%BD%D0%BE%D0%B2%D0%BD%D1%96\\_%D0%BF%D0%BE%D0%BD%D1%8F%D1%82%D1%82%D1%8F\\_%D0%BC%D0%BE%D0%B4%D0%B5%D0%BB%D1%96\\_%C2%AB%D1%81%D1%83%D1%82%D0%BD%D1%96%D1%81%D1%82%D1%8C\\_%E2%80%93%D0%B7%D0%B2%E2%80%99%D1%8F%D0%B7%D0%BE%D0%BA%D0%B8,\\_%D0%B0%D1%82%D1%80%D0%B8%D0%B1%D1%83%D1%82%D0%B8\\_%D1%8](https://wiki.cuspu.edu.ua/index.php/%D0%9C%D0%BE%D0%B4%D0%B5%D0%BB%D1%8C_%C2%AB%D1%81%D1%83%D1%82%D0%BD%D1%96%D1%81%D1%82%D1%8C_%E2%80%93%D0%B7%D0%B2%E2%80%99%D1%8F%D0%B7%D0%BE%D0%BA%C2%BB._%D0%9E%D1%81%D0%BD%D0%BE%D0%B2%D0%BD%D1%96_%D0%BF%D0%BE%D0%BD%D1%8F%D1%82%D1%82%D1%8F_%D0%BC%D0%BE%D0%B4%D0%B5%D0%BB%D1%96_%C2%AB%D1%81%D1%83%D1%82%D0%BD%D1%96%D1%81%D1%82%D1%8C_%E2%80%93%D0%B7%D0%B2%E2%80%99%D1%8F%D0%B7%D0%BE%D0%BA%C2%BB:_%D1%81%D1%83%D1%82%D0%BD%D0%BE%D1%81%D1%82%D1%96,_%D0%B7%D0%B2%E2%80%99%D1%8F%D0%B7%D0%BA%D0%B8,_%D0%B0%D1%82%D1%80%D0%B8%D0%B1%D1%83%D1%82%D0%B8_%D1%8)

2%D0%B0\_%D1%97%D1%85\_%D0%BA%D0%BB%D0%B0%D1%81%D0  
%B8%D1%84%D1%96%D0%BA%D0%B0%D1%86%D1%96%D1%8F

## ДОДАТКИ

