

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ  
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ  
ІМЕНІ ІГОРЯ СІКОРСЬКОГО»  
КАФЕДРА ПРИКЛАДНОЇ МАТЕМАТИКИ

КУРСОВА РОБОТА  
з навчальної дисципліни “Бази даних та інформаційні системи-2”  
на тему:  
СЕРВІС ЕЛЕКТРОННОЇ ЧЕРГИ

Виконала студентка групи КМ-63:

Без’язична К.Ю.

Перевірили:

Ковальчук-Хімюк Л. О

Терещенко І. О

Оцінка:

Київ – 2019

## ЗАВДАННЯ НА ВИКОНАННЯ КУРСОВОЇ РОБОТИ

Створити інформаційну систему електронної черги, що дозволить користувачу ефективно займати та відслідковувати порядок своєї черги, задля оптимізації навантаження його графіку та з іншого боку це дозволить підлаштовувати приймальні потужності закладів з огляду на кількість клієнтів зареєстрованих у чергу. Для цього необхідно виконати передпроектне дослідження(проаналізувати бізнес-процеси, бізнес правила та елементарні події), визначити сутності , що розміщуються у базі даних, розглянути операції, що можна виконувати над кожною сутністю та забезпечити цілісність системи. Додатково виконати кореляційний аналіз, кластерний аналіз та побудувати модель машинного навчання.

## АНОТАЦІЯ

В пояснювальній записці до курсової роботи описано усі основні етапи розробки даної інформаційної системи. Перший розділ містить інформацію про виконання передпроектного дослідження. У наступних розділах визначено конкретну постановку задачі та описані бізнес правила, моделювання бізнес-процесів, визначаються сутності, їх атрибути та моделюються зв'язки між ними. В останньому розділі розділі для кожної таблиці визначаються ключі, типи полів та надається графічне представлення моделі. В курсовій роботі розглядаються поняття кластерного та кореляційного аналізу, а також задача машинного навчання.

## РЕФЕРАТ

Сервіс електронної черги є найчастіше використовувана система у медичних закладах або установах для отримання українського або закордонного паспорту. Зазвичай людина проводить у черзі близько 1 години, а то і більше та не завжди вдало. Цей процес існує вже десятки років та тільки з появою, розвитком мережі Інтернет та збільшенням користувачів у світі, ми можемо використовувати онлайн черги, що є економією часу, збереженням морального стану, а також користувач може запланувати щось заздалегідь та скласти відповідний розклад. При виконанні роботи було розроблено саме таку автоматичну систему, що дозволяє користувачу запланувати розклад та зекономити час. Розроблений мікросервіс відрізняється від простої автоматизації процесу тим, що реалізує задачу кластеризації та застосовує кореляційний аналіз, а також розв'язує задачу машинного навчання.

В ході виконання курсового проекту було створено інформаційну систему на базі веб-фреймворку Flask. База даних, на якій виконано завдання – PostgreSQL 10. Модель даних створюється на базі Python, SQLAlchemy, а зі сторони PostgreSQL налаштовано обмеження на унікальність, які виникли після оптимізації бази даних та введення штучних ключів. Також з боку PostgreSQL створені послідовності. Веб-застосунок дозволяє переглядати дані, наявні в інформаційній системі, змінювати та видаляти дані, де це допустимо та додавати дані у всі сутності. Додавання даних валідується на WTForms. З Python до HTML-сторінок передається інформація та відображається з допомогою шаблонізатора Jinja, а веб-сторінки зверстані з

використанням фреймворку Bootstrap-4. Також машинне навчання та кореляційний аналіз виконано за допомогою Random Forest Regressor та проект розгорнуто на віддаленому сервері Heroku.

## ЗМІСТ

СПИСОК ТЕРМІНІВ, СКОРОЧЕНЬ ТА ПОЗНАЧЕНЬ .....	8
ВСТУП.....	9
ОСНОВНА ЧАСТИНА .....	11
1 АНАЛІЗ ПІДПРИЄМСТВА АВТОМАТИЗАЦІЇ.....	11
1.1 Діаграма ієрархії бізнес-процесів .....	12
1.2 Опис процесів .....	13
1.3 Висновки до розділу .....	20
2 ПОСТАНОВКА ЗАДАЧІ .....	21
2.1 Категорії користувачів .....	22
2.2 Класи даних.....	23
2.3 Бізнес правила.....	25
2.4 Сценарії .....	26
2.5 Висновки до розділу .....	28
3 МОДЕЛЮВАННЯ БІЗНЕС ПРОЦЕСІВ.....	29
3.1 Use case діаграма .....	29
3.2 Висновки до розділу .....	30
4 ІНФОЛОГІЧНЕ ПРОЕКТУВАННЯ .....	31
4.1 Концептуальна модель .....	31
4.2 Висновки до розділу .....	33
5 ДАТОЛОГІЧНЕ МОДЕЛЮВАННЯ .....	34

5.1 Логічна модель.....	35
5.2 Фізична модель .....	36
5.3 Висновки до розділу .....	37
ВИСНОВКИ .....	38
СПИСОК ВИКОРИСТАНИХ ЛІТЕРАТУРНИХ ДЖЕРЕЛ.....	40

## СПИСОК ТЕРМІНІВ, СКОРОЧЕНЬ ТА ПОЗНАЧЕНЬ

Мікросервіс - архітектурний стиль за яким єдиний застосунок будується як сукупність невеличких сервісів кожен з яких працює у своєму власному процесі і комунікує з рештою використовуючи легковагові механізми, зазвичай HTTP. Ці сервіси будуються навколо бізнес-потреб і розгортаються незалежно з використанням зазвичай повністю автоматизованого середовища.

Нейронна мережа - сукупність нейронів головного і спинного мозку центральної нервової системи (ЦНС) і гангліїв периферичної нервової системи (ПНС), які пов'язані або функціонально об'єднані в нервовій системі, виконують специфічні фізіологічні функції.

Кластерний аналіз - задача розбиття заданої вибірки об'єктів (ситуацій) на підмножини, які називаються кластерами, так, щоб кожен кластер складався з схожих об'єктів, а об'єкти різних кластерів істотно відрізнялися. Завдання кластеризації відноситься до статистичної обробки, а також до широкого класу завдань навчання без вчителя.

Кореляційний аналіз — це статистичне дослідження залежності між випадковими величинами, мета якого виявити чи існує істотна залежність однієї змінної від інших.

Машинне навчання — це підгалузь штучного інтелекту в галузі інформатики, яка часто застосовує статистичні прийоми для надання комп'ютерам здатності навчатися з даних, без того, щоби бути програмованими явно.



## ВСТУП

Актуальність теми. На сьогоднішній день вже існують сервіси електронних черг, що допомагають закладам налаштувати неперервну роботу. Такі типові сервіси можна побачити у приймальнях на видачу або оформлення закордонного паспорту, деяких приватних лікарнях та університетах.

Зв'язок роботи з науковими програмами і темами. Байко О. Т. Система організації електронної приймальні. Курсова робота. Львів: 2013

Мета і завдання дослідження. Метою курсової роботи є дослідження відмінностей мікросервісу від простої автоматизації робочого процесу та створення мікросервісу електронної черги.

Мета роботи передбачає виконання таких завдань:

- Задача кластеризації
- Задача кореляційного аналізу
- Задача машинного навчання

Сервіс містить зручний інтерфейс, дозволяє переглядати користувачу його номер, кількість людей у черзі та приблизний час очікування. Результатом роботи є push-повідомлення або голосове повідомлення номеру. Один з таких сервісів був впроваджений для поселення першокурсників КПІ до гуртожитків. Але через велику кількість користувачів припинив свою роботу, через величезне навантаження, що було не розраховано. Невідомо, чи творці електронної черги використовували зібрану інформації і чи робили дослідження з метою виправлення недоліків та збільшення навантаження на базу даних. Метою курсової роботи є створити підґрунтя для мікросервісу, що

зможе доповнити вже існуючий сервіс та зібрати інформацію для проведення досліджень. Згодом на базі виконаних досліджень можна буде збільшити навантаження системи, зробити real-time push-повідомлення . Недоліком такого рішення є те, що не всі користувачі погодяться надати особисту інформацію. Об'єктом дослідження у курсовій роботі є функціонал наявних інформаційних систем, існуючі рішення, які піддаються інтеграції.

Метод дослідження. Пошук та систематизація наявних рішень для створення інформаційних систем та мікросервісів.

## ОСНОВНА ЧАСТИНА

### 1 АНАЛІЗ ПІДПРИЄМСТВА АВТОМАТИЗАЦІЇ

В ході виконання курсової роботи було проведено перед проектне дослідження, метою якого є аналіз підприємства автоматизації. Перший етап проекту складається з побудови ієрархії процесів, об'єднання процесів у інтерфейси для користувача, проектування екранних форм та опис бізнес правил.

## 1.1 Діаграма ієрархії бізнес-процесів

Діаграма створена відповідно до теми курсової роботи. Кожен процес має свій унікальний ідентифікатор (назву) та відповідає напрямку розробки.



Рисунок 1.1 - Діаграма ієрархії бізнес-процесів

- На 1 рівні розкрито процеси «Використання сервісу».
- На 2 рівні розкрито процеси «Обробка користувача» та «Обробка черги»
- На 3 рівні розкрито процеси «Робота з користувачем», «Робота з розкладом», «Робота з чергою» та «Робота з місцем надання послуг».
- На 4 рівні розкрито процеси «Додавання , вилучення, оновлення, перегляд черги», «Додавання , вилучення, оновлення, перегляд місця надання послуг», «Додавання , вилучення, оновлення, перегляд користувача», «Додавання , вилучення, оновлення, перегляд розкладу» та «Аналіз використання послуг».

## 1.2 Опис процесів

Після складання діаграми процесів необхідно описати процеси. Опис наведено нижче у вигляді таблиць:

Таблиця 1.1 Обробка користувача

Назва процесу	Обробка користувача
Сутності	Client, Schedule
Вхідні атрибути сутності	Client (client_documents, client_fullname,) Schedule (time_in_queue, date, push_notification)
Опис функціоналу	Створюється користувач та його розклад
Змінені атрибути сутності	Створюється новий екземпляр сутності User

Таблиця 1.2 Обробка черги

Назва процесу	Обробка черги
Сутності	Queue, Place
Вхідні атрибути сутності	Queue(queue_name, queue_number, number_of_people, waiting_time) Place(place_name, place_site, type_of_service)
Опис функціоналу	Створення місця надання послуг та черги до відповідного місця
Змінені атрибути сутності	Створюється новий екземпляр сутності Place, Queue

Таблиця 1.3 Додавання користувача

Назва процесу	Додавання користувача
Сутності	Client
Вхідні атрибути сутності	Client (client_documents, client_fullname)
Опис функціоналу	Створення нового користувача
Змінені атрибути сутності	-

Таблиця 1.4 Оновлення користувача

Назва процесу	Оновлення користувача
Сутності	Client
Вхідні атрибути сутності	Client (client_documents, client_fullname)
Опис функціоналу	Після створення користувача можна відредагувати інформацію
Змінені атрибути сутності	client_documents, client_fullname

Таблиця 1.5 Вилучення користувача

Назва процесу	Вилучення користувача
Сутності	Client
Вхідні атрибути сутності	Client (client_documents, client_fullname)
Опис функціоналу	Видалення інформації про користувача
Змінені атрибути сутності	-

Таблиця 1.6 Перегляд користувача

Назва процесу	Перегляд користувача
Сутності	Client
Вхідні атрибути сутності	Client (client_documents, client_fullname)
Опис функціоналу	Процес перегляду інформації про користувача
Змінені атрибути сутності	-

Таблиця 1.7 Додавання розкладу

Назва процесу	Додавання розкладу
Сутності	Schedule
Вхідні атрибути сутності	Schedule (time_in_queue, date, push_notification)
Опис функціоналу	Створення нової дати у розкладі
Змінені атрибути сутності	time_in_queue, date, push_notification

Таблиця 1.8 Оновлення розкладу

Назва процесу	Оновлення розкладу
Сутності	Schedule
Вхідні атрибути сутності	Schedule (time_in_queue, date, push_notification)
Опис функціоналу	Після створення розкладу можна відредагувати інформацію
Змінені атрибути сутності	time_in_queue, date, push_notification

Таблиця 1.9 Вилучення розкладу

Назва процесу	Вилучення розкладу
Сутності	Schedule
Вхідні атрибути сутності	Schedule (time_in_queue, date, push_notification)
Опис функціоналу	Видалення інформації про розклад
Змінені атрибути сутності	-

Таблиця 1.10 Перегляд розкладу

Назва процесу	Перегляд розкладу
Сутності	Schedule
Вхідні атрибути сутності	Schedule (time_in_queue, date, push_notification)
Опис функціоналу	Процес перегляду інформації про розклад
Змінені атрибути сутності	-

Таблиця 1.11 Додавання черги

Назва процесу	Додавання черги
Сутності	Queue
Вхідні атрибути сутності	Queue(queue_name, queue_number, number_of_people, waiting_time)
Опис функціоналу	Створення нової черги
Змінені атрибути сутності	queue_name, queue_number, number_of_people, waiting_time

Таблиця 1.12 Оновлення черги



Назва процесу	Оновлення черги
Сутності	Queue
Вхідні атрибути сутності	Queue (queue_name, queue_number, number_of_people, waiting_time)
Опис функціоналу	Після створення черги можна відредагувати інформацію
Змінені атрибути сутності	queue_name, queue_number, number_of_people, waiting_time

Таблиця 1.13 Вилучення черги

Назва процесу	Вилучення черги
Сутності	Queue
Вхідні атрибути сутності	Queue (queue_name, queue_number, number_of_people, waiting_time)
Опис функціоналу	Видалення інформації про чергу
Змінені атрибути сутності	-

Таблиця 1.14 Перегляд черги

Назва процесу	Перегляд черги
Сутності	Queue
Вхідні атрибути сутності	Queue (queue_name, queue_number, number_of_people, waiting_time)
Опис функціоналу	Процес перегляду інформації про чергу
Змінені атрибути сутності	-

Таблиця 1.15 Додавання місця

Назва процесу	Додавання місця
Сутності	Place
Вхідні атрибути сутності	Place (place_name, place_site, type_of_service)
Опис функціоналу	Створення нового місця
Змінені атрибути сутності	place_name, place_site, type_of_service

Таблиця 1.16 Оновлення місця

Назва процесу	Оновлення місця
Сутності	Place
Вхідні атрибути сутності	Place (place_name, place_site, type_of_service)
Опис функціоналу	Після створення місця можна відредагувати інформацію
Змінені атрибути сутності	place_name, place_site, type_of_service

Таблиця 1.17 Вилучення місця

Назва процесу	Вилучення місця
Сутності	Place
Вхідні атрибути сутності	Place (place_name, place_site, type_of_service)
Опис функціоналу	Видалення інформації про place
Змінені атрибути сутності	-

Таблиця 1.18 Перегляд місця

Назва процесу	Перегляд місця
Сутності	Place
Вхідні атрибути сутності	Place (place_name, place_site, type_of_service)
Опис функціоналу	Процес перегляду інформації про місце
Змінені атрибути сутності	-

Таблиця 1.19 Аналіз використання послуг

Назва процесу	Аналіз використання послуг
Сутності	Place, Queue
Вхідні атрибути сутності	Place (place_name, place_site, type_of_service) Queue (queue_name, queue_number, number_of_people, waiting_time)
Опис функціоналу	Процес перегляду інформації про тип послуги та кількість людей, що її використовують
Змінені атрибути сутності	-

### 1.3 Висновки до розділу

Аналіз підприємства автоматизації є першим етапом у розробці проекту. Для якісної розробки необхідно проведення передпроектного дослідження, що розуміє під собою створення чіткої бізнес структури розробленого мікросервісу. Для зображення цієї структури побудовано бізнес процеси, що проілюстровано за допомогою діаграми ієрархії процесів, а також надано детальний опис процесів.

## 2 ПОСТАНОВКА ЗАДАЧІ

Завдання курсового проекту полягає у розробці мікросервісної архітектури, основною функцією якого є автоматичний запис у чергу. Результатом мікросервісу є push-повідомлення про стан у черзі.

Метою курсової роботи є дослідження відмінностей між простою автоматизацією робочого процесу та створення мікросервісу для електронної черги.

Основні завдання до виконання:

- Задача кластеризації
- Задача кореляційного аналізу
- Задача машинного навчання

Нижче описується виконання поставлених задач:

Завдяки використаній штучній мережі, реалізуються методи кластерного аналізу, система класифікує тип послуг в залежності від кількості користувачів у черзі. Завдяки реалізованому кореляційному аналізу можливо розглянути залежність типу послуги від часу очікування в системі, що наглядно демонструється за допомогою матриці кореляції.

## 2.1 Категорії користувачів

Основною функцією цієї системи є запис користувача (user role - standard) до черги.

В процесі виконання було створено єдину роль користувача сервісу. Користувач має змогу вільно використовувати розроблену платформу для її потреби, переглядати інформацію про черги та місця, де використовується цей сервіс, а також виконувати всі CRUD операції над сутностями.

## 2.2 Класи даних

Розглянемо класи даних наведені у таблицях 2.1-2.4

Таблиця 2.1 Клас даних – Клієнт

Сутність	Client	
Опис сутності	Головний користувач мікросервісу	
Атрибути	Опис атрибуту	Пов'язана сутність з атрибутом
Client_fullname	Інформація про клієнта	-
Client_documents	Ідентифікатор клієнта, його документи	-
Place_name	Ідентифікатор місця, назва місця	Place
Date	Ідентифікатор розкладу, дата	Schedule

Таблиця 2.2 Клас даних – Місце

Сутність	Place	
Опис сутності	Місце надання послуг клієнту	
Атрибути	Опис атрибуту	Пов'язана сутність з атрибутом
Place_name	Ідентифікатор місця, назва місця	-
Place_site	Сайт місця	-
Type_of_service	Тип послуг, що надаються в місці	-

Таблиця 2.3 Клас даних – Черга

Сутність	Queue	
Опис сутності	Черга в яку записується клієнт в відповідне місце.	
Атрибути	Опис атрибуту	Пов'язана сутність з атрибутом
Queue_number	Номер в черзі	-
Queue_name	Ідентифікатор черги, її назва	-
Place_name	Ідентифікатор місця, назва місця	Place
Date	Ідентифікатор розкладу, дата	Schedule
Number_of_people	Кількість клієнтів в черзі	-
Waiting_time	Час очікування	-

Таблиця 2.4 Клас даних – Розклад

Сутність	Schedule	
Опис сутності	Розклад, що містить інформацію про чергу	
Атрибути	Опис атрибуту	Пов'язана сутність з атрибутом
Time_in_queue	Час на який заплановано чергу	-
Push_notification	Повідомлення про стан в черзі	-
Date	Ідентифікатор розкладу, дата	-



### 2.3 Бізнес правила

1. Користувач не має змогу переносити дату надання відповідної послуги.
2. Можна створити декілька черг в 1 день.
3. Користувач може запланувати лише 1 послугу на день.
4. Користувач має змогу користуватися сервісом за умови надання свого номеру паспорту.
5. Користувач не може дублювати паспортні дані.
6. Місце надання послуг може бути зараєстроване лише 1 раз.
7. Користувач має змогу запланувати 1 чи більше черг в 1 місце, де надаються послуги.
8. Назви черг не повинні дублюватися.
9. Користувач має змогу отримати інформацію про його стан в черзі.
10. Користувач має змогу переглянути інформацію про кількість людей у черзі.

## 2.4 Сценарії

Сценарії є важливою частиною для тестувальників та розробників проекту. Їх застосовують для широкого спектру задач:

1. якщо потрібна якісна та повна специфікація,
2. для підтримки системи,
3. для спрощеного пошуку помилки
4. у випадку, коли потрібно описати якусь частину функціональності.

Сценарії наведено нижче у вигляді таблиць:

Таблиця 2.1 Створення користувача

Діючі особи	Клієнт, Система
Мета	Клієнт: заповнення інформації, яку запитує система. Система: створити запис про клієнта у БД.
Передумова	Клієнт не має облікового запису.
Успішний сценарій	1. Користувач заходить у систему. 2. Користувач натискає на кнопку створити клієнта. 3. Система видає форму для заповнення користувачу. 4. Користувач натискає на кнопку надіслати. Якщо він вводить номер паспорту, що існує, то форма йому наголошує про це і необхідно змінити дану інформацію. 5. Система видає заповнену інформацію.
Результат	Користувач зареєстрований у системі та може використовувати сервіс.

Таблиця 2.2 Створення місця

Діючі особи	Користувач, Система
Мета	Користувач: Створює місце для надання послуг. Система: Створює запис про місце у БД.
Передумова	Місце ще не було створене.
Успішний сценарій	1. Користувач запускає систему. Натискає на кнопку створити місце. 2. Система надає форму для заповнення інформації про місце. 3. Користувач заповнює інформацію та натискає надіслати форму. Якщо інформація про назву місця вже існує, то надається форма з повідомленням, що це місце вже є у базі і необхідно створити інше. 4. Система надає інформацію про місце, що було створене.
Результат	Користувач успішно створив місце, де надаються послуги.

Таблиця 2.3 Оновлення користувача

Діючі особи	Користувач, Система
Мета	Користувач: хоче оновити інформацію про себе. Система: надає доступ до редагування БД.
Передумова	Користувач вже створений.
Успішний сценарій	1. Користувач натискає на редагування та вносить зміни у інформацію. 2. Система показує вже відредаговану інформацію.
Результат	Користувач успішно оновив інформацію.

## 2.5 Висновки до розділу

Постановка задачі є другим етапом розробки, в ході якого були визначені цілі і призначення створення системи, сформульовані основні вимоги до розроблюваної системи та надано перелік основних функцій.

В результаті аналізу сформована модель, яка містить класи даних, категорії користувачів, бізнес правила та сценарії.

### 3 МОДЕЛЮВАННЯ БІЗНЕС ПРОЦЕСІВ

#### 3.1 Use case діаграма

Змодельований Use case для процесів, що можуть виконуватися у мікросервісі, зображено на рисунку 3.1.

Під кейсом розуміється інтерфейс для користувача, що дозволяє йому взаємодіяти з одним або декількома процесами. Кожний кейс має свою унікальну назву та список процесів, що він реалізує.

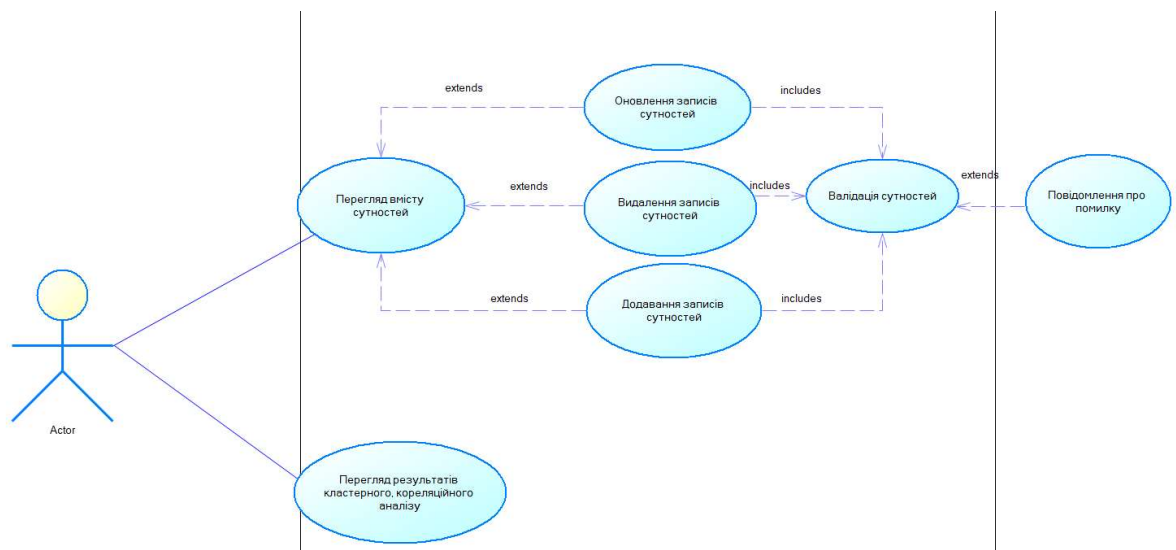


Рисунок 3.1 – Use case діаграма для мікросервісу.

### 3.2 Висновки до розділу

Розділ моделювання бізнес процесів належить до третього етапу розробки та базується на попередніх розділах. Основним завданням розділу є побудова use case діаграм, які слугують ілюстрацією взаємодії користувача з процесами.

## 4 ІНФОЛОГІЧНЕ ПРОЕКТУВАННЯ

### 4.1 Концептуальна модель

У розділ інфологічне моделювання необхідно створити концептуально модель, яка буде відображати «сутність-зв'язок». На концептуальному рівні визначаються основні поняття предметної області та їх взаємозв'язок.

На рисунку 4.1 наведено концептуальну модель даних та опис сутностей, зв'язків між ними та їх атрибутів.

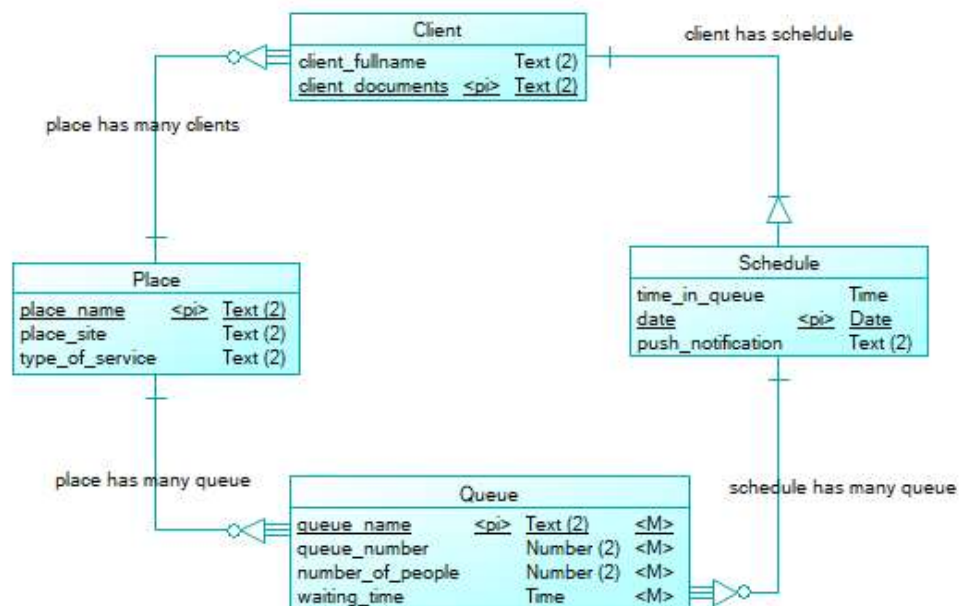


Рисунок 4.1 – Концептуальна модель

Відповідно до рисунку 4.1 було визначено такі сутності:

1. Client ( client\_fullname, client\_documents)
2. Place (place\_name, place\_site, type\_of\_service)
3. Queue (queue\_name, queue\_number, number\_of\_people, waiting time)
4. Schedule (date, time\_in\_queue, push\_notification )



## 4.2 Висновки до розділу

Метою розділу "Інфологічне моделювання" є отримання семантичної моделі даних, що відображає інформаційний зміст конкретного проекту. На цьому етапі було виконано чотири основні кроки:

1. визначення сутностей;
2. визначення атрибутів сутностей;
3. ідентифікація ключових атрибутів;
4. визначення зв'язків між сутностями.

## 5 ДАТОЛОГІЧНЕ МОДЕЛЮВАННЯ

У розділі даталогічного моделювання необхідно побудувати логічну та фізичну моделі даних. Логічна модель - це уточнення концептуальної моделі, що була розглянута у попередньому розділі, а фізична – таблиці, що необхідні для реляційної бази даних.

## 5.1 Логічна модель

На цьому етапі можливе створення додаткових сутностей та зовнішніх ключів.

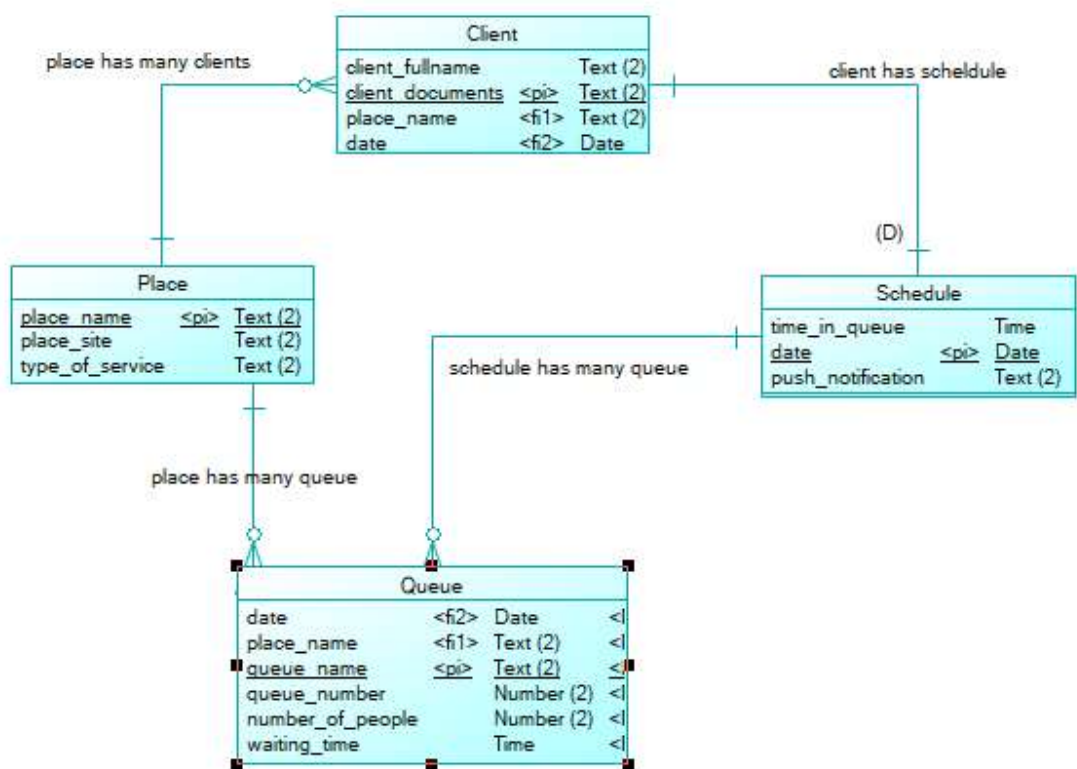


Рисунок 5.1 – Логічна модель

## 5.2 Фізична модель

На цьому етапі створену модель можна використовувати у реляційній базі даних.

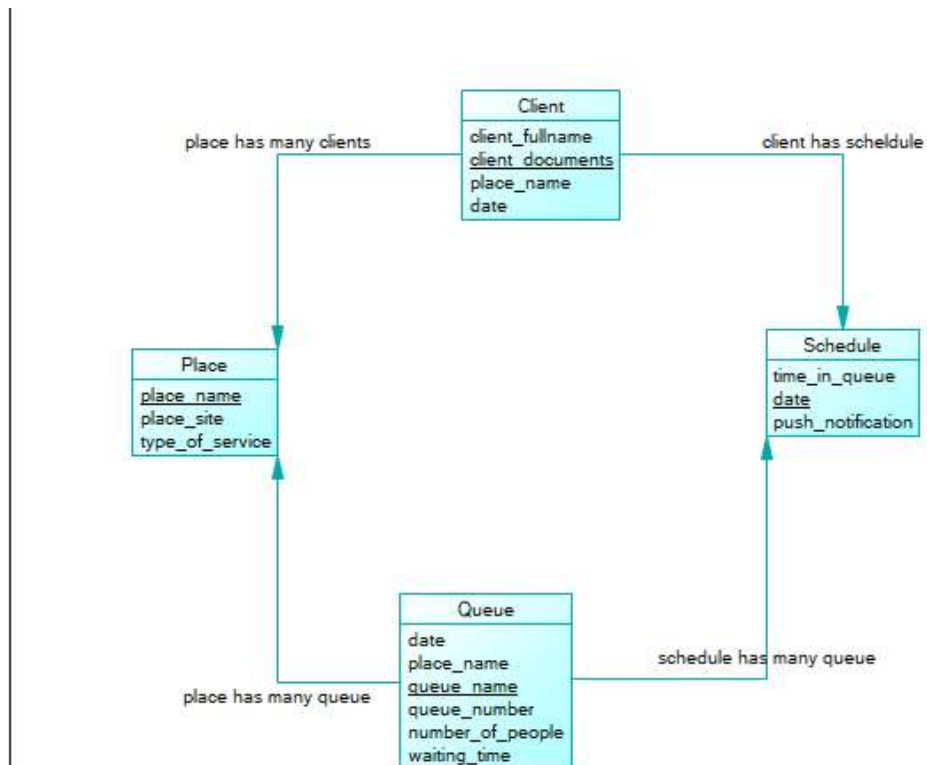


Рисунок 5.2 – Фізична модель

### 5.3 Висновки до розділу

Метою розділу "Даталогічне моделювання" є побудова чіткої моделі даних, яка буде застосована при проектуванні бази даних. На цьому етапі було створено логічну та фізичну моделі.

Основною відмінність між цими моделями було помічено, що сутності, атрибути і групи ключів можна створювати тільки на логічному рівні моделі, а таблиці, стовпці, індекси можна створювати тільки на фізичному рівні.

## ВИСНОВКИ

Завданням курсової роботи було побудова мікросервісу, що забезпечить основну функцію – записатися у чергу. Було визначено основні етапи проекту: побудова ієрархії процесів, визначено постановку задачі, визначено сценарії елементарних процесів, класів даних та сутностей з їх атрибутами.

Результат виконання : мікросервіс, що дозволяє переглядати , додавати, змінювати та видаляти інформацію сутностей. База даних – PostgreSQL, для програмування - SQLAlchemy. З Python до HTML-сторінок передається інформація та відображається з допомогою шаблонізатора Jinja, а веб-сторінки зверстані з використанням фреймворку Bootstrap-4.

Окремо на сайті є сторінка, що дозволяє переглянути результати машинного навчання, кластерного та кореляційного аналізу. У частині, що відповідає за машинне навчання було використано Random Forest Regressor для визначення кореляції , тобто залежності типу послуги від часу очікування – дані взято з створеної власної бази даних. Результат поданий в матриці кореляції.

Розроблена система потребує доопрацювання. Для того, щоб з нею могли працювати люди, необхідно насамперед , створити авторизацію, різні ролі, зробити інтеграцію з Google Calendar або налаштувати для використання стандартного розкладу влаштованого у телефони користувача. Також необхідно розробити real-time push-повідомлення для зручнішого відстеження своєї черги, а також навчити модель представляти список черг у які зараз вигідніше записатися на основі аналізу кількості людей та часу очікування.

Також необхідно дозволити користувачу отримувати не одну послугу, а декілька з метою економії часу, для цього необхідно перебудувати модель даних та сформувати БД по іншому.

## СПИСОК ВИКОРИСТАНИХ ЛІТЕРАТУРНИХ ДЖЕРЕЛ

1. Flask-Sqlalchemy documentation [Електронний ресурс] – Режим доступу до ресурсу : <https://flask-sqlalchemy.palletsprojects.com/en/2.x/>
2. PostgreSQL 10.11 Documentation [Електронний ресурс] – Режим доступу до ресурсу : <https://www.postgresql.org/docs/10/index.html>
3. SQLAlchemy 1.3 Documentation[Електронний ресурс] – Режим доступу до ресурсу : <https://docs.sqlalchemy.org/en/13/>