МІНІСТЕРСТВО НАУКИ І ОСВІТИ УКРАЇНИ

КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ

УКРАЇНИ «КПІ імені Ігоря Сікорського»

факультет «Інформатики та обчислювальної техніки»

кафедра «Автоматики та управління в технічних системах»

КУРСОВА РОБОТА

по курсу «Бази даних-2»

на тему: «База даних Київської обласної дирекції служби зайнятості населення»

Студента III курсу ІТ-51 групи

спеціальності «Програмна інженерія»

Бессмертного Романа Сергійовича

Керівник: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Національна шкала \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Кількість балів: \_\_\_\_\_\_\_\_\_\_Оцінка: ECTS \_\_\_\_\_

Члени комісії: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(підпис) (прізвище та ініціали)

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(підпис) (прізвище та ініціали)

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(підпис) (прізвище та ініціали)

м. Київ – 2017 рік

**ЗМІСТ**

[ВСТУП 6](#_Toc501764762)

[1 ТЕОРЕТИЧНИЙ АНАЛІЗ ІНФОРМАЦІЙНИХ ПОТОКІВ ТА ОСОБЛИВОСТЕЙ ПРЕДМЕТНОЇ ОБЛАСТІ ДОСЛІДЖЕННЯ 7](#_Toc501764763)

[1.1 Аналіз інформаційних потреб та визначення предметної області дослідження 7](#_Toc501764764)

[1.2 Архітектура та засоби реалізації бази даних формування обліку робочого часу за електронними картками 13](#_Toc501764765)

[1.3 Опис Microsoft SQL Server 15](#_Toc501764766)

[2 ПРОЕКТУВАННЯ БАЗИ ДАНИХ 17](#_Toc501764767)

[2.1 Аналіз інформаційних та датологічних процесів 17](#_Toc501764768)

[2.2 Проектування структури бази даних обліку робочого часу за електронними картками 18](#_Toc501764769)

[3 РЕАЛІЗАЦІЯ ПІДСИСТЕМИ ОБРОБКИ ДАНИХ 23](#_Toc501764770)

[3.1 Проектування інтерфейсу обробки даних 23](#_Toc501764771)

[3.2 Фізична реалізація бази даних 24](#_Toc501764772)

[3.3 Виконання основних запитів 26](#_Toc501764773)

[3.4 Підтримка міграцій бази даних 33](#_Toc501764774)

[4 АДМІНІСТРУВАННЯ БАЗИ ДАНИХ 35](#_Toc501764775)

[4.1 Порядок налаштування сервера 35](#_Toc501764776)

[4.2 Налаштування прав доступу 37](#_Toc501764777)

[СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ 39](#_Toc501764778)

[ДОДАТОК А. Діаграми бази даних 42](#_Toc501764779)

# ВСТУП

На всій планеті веб-сайти пошуку роботи є широко відомою частиною інтернету. Великі гравці, такі як Indeed and Monster, перетворили полювання на роботу та вербування в справжню онлайн-індустрію.

Люди люблять економити час, використовуючи технологічні нововведення; Портал пошуку роботи є ще одним способом працювати розумніше, не важче. Шукачі роботи та компанії усвідомлюють цінність роботи в Інтернеті: вони отримують кращий результат при більшій швидкості та зниженні витрат.

Сфера роботи порталу зараз досить стабілізована, принаймні стосовно обсягів руху. Люди в пошуках роботи використовують ці портали для пошуку позицій у багатьох галузях промисловості, виходячи за межі ІТ до таких галузей, як інжиніринг, продаж, виробництво та фінансові послуги. Тим не менше, вони отримують жорстку конкуренцію з боку соціальних та професійних мереж, таких як LinkedIn. Але на ринку ще існують потенційні можливості, такі як розширення їх проникнення в сільські райони та менші міста.

Метою даної курсової роботи є розробка бази даних Київської обласної дирекції служби зайнятості населення. База даних має надавати дані для управління користувачами, створення профілів та резюме, а також для розміщення, пошуку та подання заявок на роботу.

1. ТЕОРЕТИЧНИЙ АНАЛІЗ ІНФОРМАЦІЙНИХ ПОТОКІВ ТА ОСОБЛИВОСТЕЙ ПРЕДМЕТНОЇ ОБЛАСТІ ДОСЛІДЖЕННЯ
   1. Аналіз інформаційних потреб та визначення предметної області дослідження

Для висунення функціональних та не функціональних вимог слід чітко сформувати предметну область, від якої можна буде відштовхуватись і при розробці бази даних та програмного засобу. Опишемо предметну область.

І роботодавці, і шукачі роботи очікують наступні функціональні можливості від веб-сайту:

1. Люди можуть зареєструватися як шукачі роботи, створювати свої профілі та шукати роботу, відповідну їх сукупності навичок.
2. Користувачі можуть завантажувати свої існуючі резюме. Якщо їх немає, вони повинні мати змогу заповнити форму та отримати відповідне резюме.
3. Люди можуть звертатися безпосередньо до існуючих вакансій.
4. Компанії можуть реєструвати, розміщувати роботи та шукати профілі шукачів роботи.
5. Кілька представників компанії повинні мати можливість зареєструвати та розміщувати роботи.
6. Представники компанії можуть переглядати список претендентів на роботу та можуть зв'язатися з ними, ініціювати співбесіду або виконувати інші дії, пов'язані з їх посадою.
7. Зареєстровані користувачі повинні мати можливість шукати роботу та відфільтрувати результати на основі місця розташування, необхідних навичок, заробітної плати, рівня досвіду тощо.

Покажемо варіанти використання за допомогою діаграми Use Case:

На рисунку 1.1 зображено Use Case діаграми веб-сайту Київської обласної дирекції служби зайнятості населення.

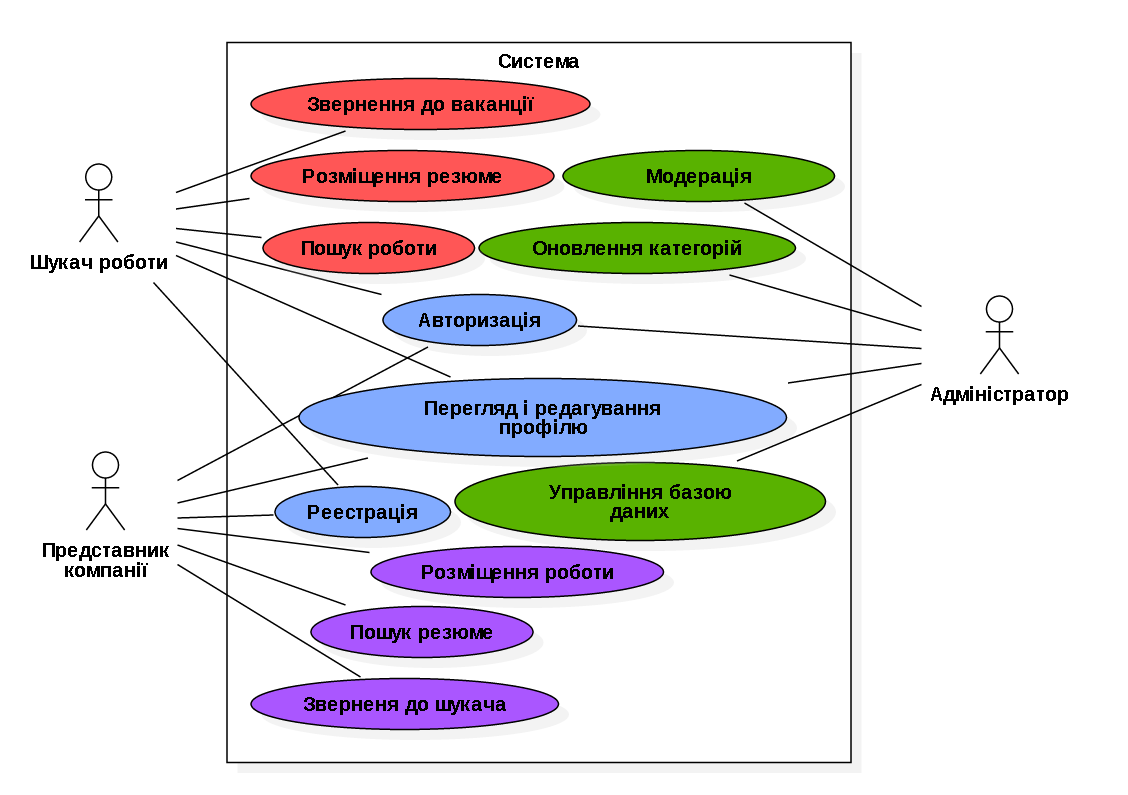


Рисунок 1.1 – Use Case діаграма веб-сайту Київської обласної дирекції служби зайнятості населення.

Опишемо наведену діаграму більш детально. Першою основною функціональною категорією є управління користувачами - як саме портал обробляє користувачів, тобто шукачів роботи, робітників HR, а також незалежних або допоміжних рекрутерів.

Користувачі повинні зареєструватися перш ніж вони зможуть користуватися порталом. Для реалізації реєстрації та авторизації потрібно зберігати основні дані – логін і пароль - незалежно від типу користувачів. Усі інше дані користувачів, такі як телефон та місце проживання, також заносяться при реєстрації. Потрібно зауважити, що оскільки певна інформація, наприклад освіта, залежать від типу користувача, процес реєстрації також змінюється в залежності від типу.

Якщо користувач вже зареєстрований, йому потрібно авторизуватися. Авторизація включає в себе введення логіну та паролю, і перевірити їх валідність за допомогою збережених раніше відповідних даних.

Незалежно від типу, користувач може переглянути і редагувати свій профіль. Тип користувача визначає, яку додаткову інформацію потрібно зберегти, і, відповідно, яку додаткову інформацію користувач може редагувати.

Як ми можемо побачити на рисунку 1.1, усі інші варіанти використання залежать від типу користувача. Розглянемо їх окремо відповідно до типів.

На рисунку 1.2 зображено діаграму активності для шукача роботи.

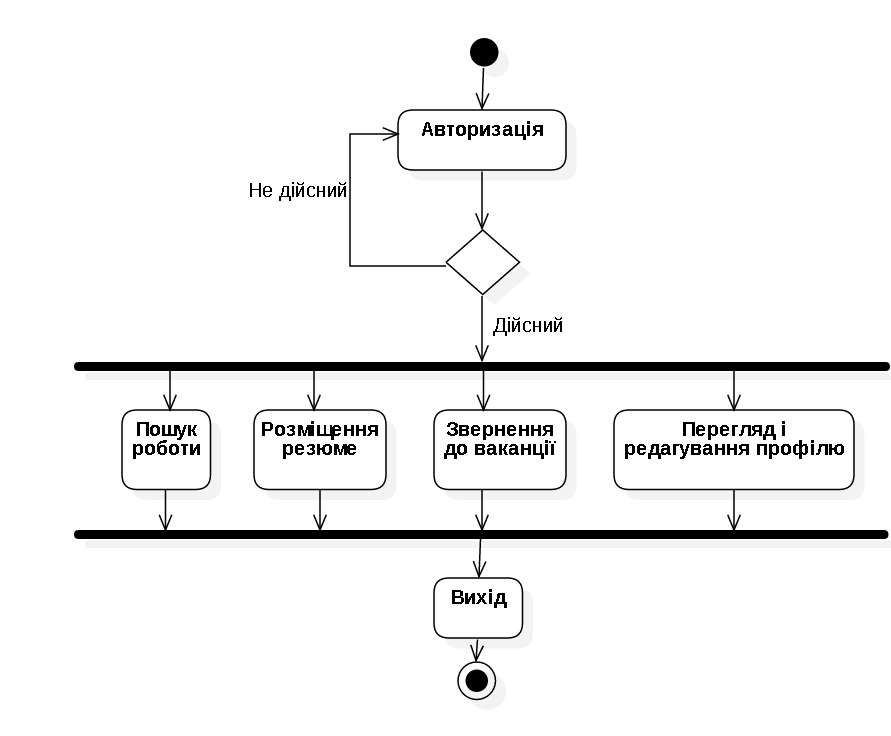


Рисунок 1.2 – Діаграма активності для шукача роботи веб-сайту Київської обласної дирекції служби зайнятості населення.

Відповідно до рисунку 1.2, унікальними для шукача роботи є пошук роботи, розміщення резюме та звернення до існуючої вакансії. Розглянемо ці дії більш детально.

Виконуючи пошук роботи, користувач хоче бачити вакації в такому вигляді, що дозволяє знайти роботу, що якомога краще відповідає його потребам, за найменшої витрати часу і сил. Для досягнення цієї мети, необхідно надати користувачу можливість переглядати список існуючих вакансій, а також можливість відфільтрувати та відсортувати доступні дані відповідно до потреб шукача роботи - місця розташування, необхідних навичок, заробітної плати, рівня досвіду тощо. Викладання повної інформації про кожну роботу у список не є доцільним. Кращим варіантом буде використання лише основних даних для процесу пошуку. Тоді якщо користувач зацікавлений певною роботою, він має мати можливість переглянути детальну інформацію.

Резюме – найбільш критична частина системи. Якщо сайт не зможе зберегти якомога більше інформації про шукачів роботи, рекрутерам буде важко оптимально знаходити кандидатів. Необхідно зберігати детальну інформацію про людину, її освіту, спеціалізацію, досвід роботи. Будемо вважати, що шукач роботи бажає влаштуватись на чітко визначену спеціальність. Це дозволить скоротити кількість інформації, і підвищити її якість. Наприклад, для одного інтерв’ю необхідно вказувати тільки ту освіту і досвід роботи, що чітко мають відношення до бажаної роботи.

Якщо користувач зацікавлений вакансією і бажає встановити зв’язок, він має мати можливість відправити резюме представникам компанії, які, в свою чергу, будуть відповідно мати можливість зв'язатися з ним, ініціювати співбесіду або виконувати інші дії, пов'язані з їх посадою.

Ці та інші дії представникам компанії розглянемо більш детально на діаграмі активності представників компанії, зображеної на рисунку 1.3.

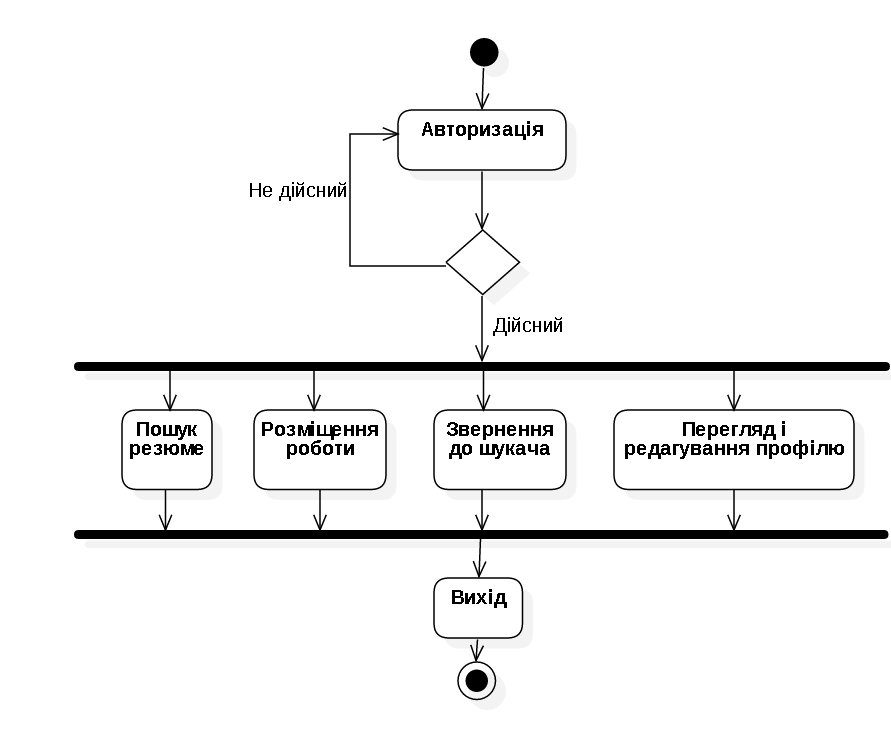


Рисунок 1.3 – Діаграма активності для представника компанії веб-сайту Київської обласної дирекції служби зайнятості населення.

Виконуючи пошук потенційного робітника, представник компанії хоче бачити резюме в такому вигляді, що дозволяє знайти людину, що якомога краще відповідає його потребам, за найменшої витрати часу і сил. Для досягнення цієї мети, необхідно надати представнику можливість переглядати список існуючих резюме, а також можливість відфільтрувати та відсортувати доступні дані відповідно до потреб компанії - місця розташування, освіти, спеціалізації, досвіду роботи тощо. Викладання повної інформації про кожне резюме у список не є доцільним. Кращим варіантом буде використання лише основних даних для процесу пошуку. Тоді якщо представник зацікавлений певним користувачем, він має мати можливість переглянути детальну інформацію.

Розміщення, і, відповідно, пошук робіт – головна особливість порталу. Необхідно зберігати детальну інформацію про тип роботи, необхідну освіту, необхідну сукупність навичок, місце роботи. Необхідно пам'ятати, що не вся інформація має бути доступною для шукачів роботи. Наприклад, представник компанії може забажати використовувати терміни виду «глобальна автомобільна компанія», «ІТ-компанія що базується в Києві» тощо.

Якщо рекрутер знайшов відповідне резюме і бажає встановити зв’язок, він має мати можливість зв'язатися з ним, ініціювати співбесіду або виконувати інші дії, пов'язані з їх посадою.

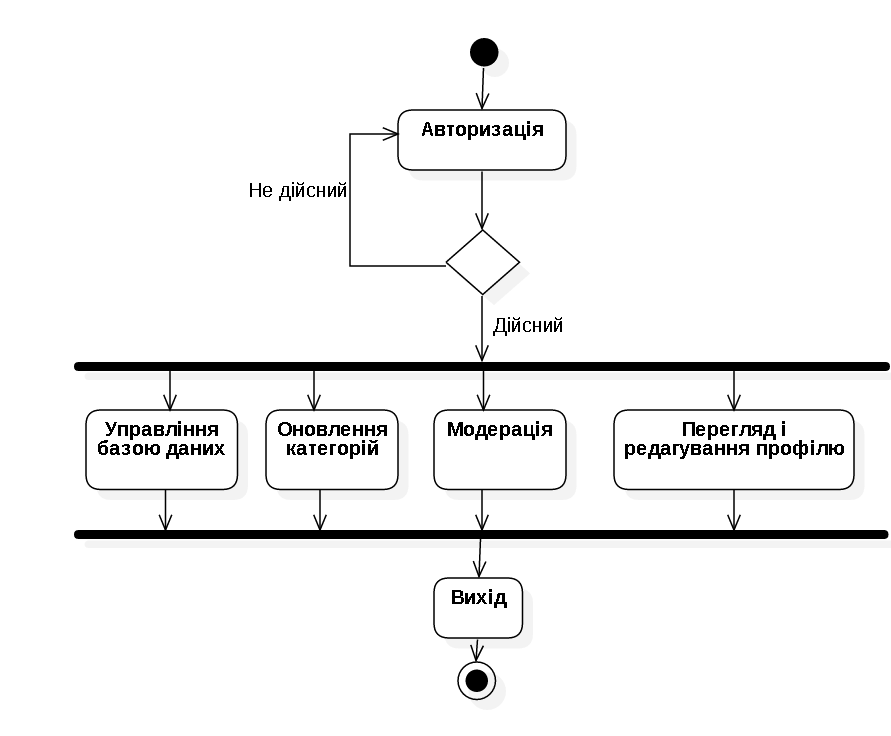


Рисунок 1.3 – Діаграма активності для адміністратора веб-сайту Київської обласної дирекції служби зайнятості населення.

Адміністратор сайту стежить за працездатністю сервера (серверного устаткування і програм), на якому знаходиться сайт. Адміністратор несе відповідальність за мережеву безпеку. В деяких випадках він стає веб-майстером і може займатися розкручуванням, вести статистику відвідуваності, виконувати обов'язки контент-менеждера, стежачи за своєчасним оновленням інформації.

У нашому випадку ми можемо виділити три основні дії, які виконує адміністратор веб-порталу – управління базою даних, оновлення категорій та модерація.

Задля забезпечення працездатності сервера адміністратор має доступ до усіх записів бази даних, і може редагувати їх як за допомогою інтерфейсу, так і напряму через SQL запити.

Адміністратор має слідкувати за актуальністю категорій – види роботи, можливі місця роботи, тощо. Таким чином виконується своєчасне оновлення інформації, що знаходиться на порталі.

Необхідно слідкувати за відповідністю поведінки користувачів правилам сайту та чинному законодавству, тому адміністратор має можливість переглядати активність користувачів і відстежувати виконання правил.

1.2 Архітектура та засоби реалізації бази даних формування обліку робочого часу за електронними картками

Основною операцією для розроблюваної бази даних є запис нових профілів та резюме. Так, за достатнього трафіку, нові дані створюються майже неперервно. На кожну операцію проводиться звернення до бази даних.

Проте, немає необхідності постійного зв’язку апаратної частини із базою даних. Апаратна частина буде проводити усі операції через сервер, який вже, в свою чергу, буде взаємодіяти із базою даних. Тому, вибір СУБД, найбільш сумісний із серверною частиною, є доцільним.

Серверна частина буде реалізована як Web API, та може бути розгорнутою на платформі під керівництвом операційної системи Windows.

Порівняємо системи управління базами даних MS SQL SERVER та Oracle. Результати порівняння занесемо до таблиці 1.1.

Таблиця 1.1 – Порівняльна характеристика СУБД Microsoft SQL Serve та Oracle

|  |  |  |
| --- | --- | --- |
| Характеристика | Microsoft SQL Server | Oracle |
| Адміністративне керування | Відмінно | Відмінно |
| Графічні інструменти | Відмінно | Добре |
| Простота обслуговування | Відмінно | Відмінно |
| Механізм данних | Добре | Відмінно |
| Робота с декількома ЦП | Задовільно | Відмінно |
| Функції з’єднання і вибір індексів | Відмінно | Відмінно |
| Одночасний доступ декількох користувачів | Добре | Відмінно |
| Обробка даних мультимедіа | Не задовільняє | Відмінно |
| Подключення к Web | Відмінно | Відмінно |
| Повнотекстовий пошук | Добре | Відмінно |
| Функціональна сумісність | Добре | Добре |
| Інтеграція з іншими СУБД | Добре | Добре |
| Єдина реєстрація | Добре | Добре |
| Робота під керівництвом ОС | Задовільно | Добре |
| Можливості програмування | Задовільно | Відмінно |
| Процедури, що зберігаються та тригери | Добре | Відмінно |
| Вбудована мова програмування | Задовільно | Відмінно |
| Побудова БД | Відмінно | Відмінно |
| Мова SQL | Відмінно | Відмінно |
| Підтримка об’єктно-орієнтованої парадигми | Задовільно | Відмінно |
| Робота у режимі віддаленого доступу | Відмінно | Відмінно |
| Тиражування | Відмінно | Відмінно |
| Розподілена обробка транзакцій | Відмінно | Відмінно |
| Дистанційне адміністрування | Добре | Відмінно |
| Організація сховищ даних і підготовка звітів | Відмінно | Відмінно |
| Засоби завантаження | Відмінно | Відмінно |
| Засоби аналізу | Відмінно | Відмінно |

Також, порівняємо системи управління базами даних за вимогами технічного завдання.

Таблиця 1.2 – Порівняння СУБД за вимогами технічного завдання

|  |  |  |
| --- | --- | --- |
| Необходимые требования | MS SQL Server | Oracle |
| Локалізація інтерфейсу користувача, можливість побудови і сортування полів БД, що містять символи кирилиці | + | + |
| Підтримка структури відносних даних | + | + |
| Підтримка технології клієнт/сервер | + | + |
| Підтримка багатопроцесорної архітектури | + | + |
| Підтримка кластерної архітектури | + | + |
| Наявність засобів для створення індексів і кластерів для підвищення ефективності використання даних | + | + |
| Відновлення баз даних із використанням журнала трансакцій | + | + |
| Механізм блокування трансакцій під час запису або на рівні сторінок | + | + |
| підтримка ANSI SQL | + | + |
| Підтримка стандарту SQL-3 (новое название – SQL99) | + | + |
| підтримка ODBC | + | + |
| Контроль цілісності БД | + | + |
| Підтримка утиліт резервування БД | + | + |
| Імпорт/експорт таблиць БД | + | + |
| Сумісність з ОС модулів користувача та сервера | + | + |
| Підтримка визначених мережевих протоколів | + | + |
| Наявність графічного інтерфейсу для адміністраторів БД | + | + |
| Контроль доступу до даних. Аутентифікація засобами СУБД | + | + |
| Централізоване керування користувачами | + | + |
| Наявність оптимізатора запитів для оптимізації планів виконання | + | + |
| Підтримка великих двійкових об’єктів (BLOB) | + | + |
| Підтримка OLAP технологій, спеціалізованих засобів OLAP-аналізу | + | + |
| Підтримка протоколів VI SAN (Virtual Interface System Area Network) | + | + |
| Відлагоджений механізм реплікації даних | + | + |
| Підтримка служби єдиного каталогу | + | + |

З порівняльної таблиці бачимо, що обидві СУБД добре вправляються із необхідними задачами. Хоча Oracle в деяких моментах є кращим за Microsoft SQL Server, остання легша в користуванні та простіше інтегрується з ASP Web API, тому для реалізації бази даних буде обрано Microsoft SQL Server.

1.3 Опис Microsoft SQL Server

Це система керування базами даних, розроблена корпорацією Microsoft.

Основний використовуваний мову запитів - Transact-SQL, створений Microsoft. Transact-SQL є реалізацією стандарту по структурованій мові запитів (SQL) з розширеннями. Використовується для роботи з невеликими і середніми за розміром базами даних.

Завдяки інтеграції з середовищем розробки Microsoft Visual Studio SQL

Server дозволяє розробляти керовані додатками дані з широкими можливостями, які забезпечують поліпшену безпеку сховищ і швидке розгортання.

У своєму складі система має засоби створення баз даних, роботи з інформацією баз даних, перенесення даних з інших систем і в інші системи, резервного копіювання та відновлення даних, розвинену систему транзакцій, систему реплікації даних, реляційну підсистему для аналізу, оптимізації та виконання запитів клієнтів, систему безпеки для управління правами доступу до об'єктів бази даних та ін. Система не містить засобів розробки клієнтських додатків.

SQL Server будується за структурою таблиці на основі рядків, а також з'єднує пов'язані елементи даних у різних таблицях, уникаючи необхідності резервного зберігання даних у кількох місцях в межах бази даних. Реляційна модель також забезпечує дотримання принципів атомності, узгодженості, ізоляції та довговічності - загальновідомі як властивості ACID і призначені для гарантування того, що транзакції бази даних надійно обробляються.

Основним компонентом Microsoft SQL Server є SQL Server Database Engine, який керує збереженням, обробкою даних та безпекою даних. Вона включає в себе реляційний движок, який обробляє команди та запити, а також двигун зберігання, який керує файлами, таблицями, сторінками, індексами, буферами даних та транзакціями. Збережені процедури, активатори, перегляди та інші об'єкти бази даних також створюються та виконуються за допомогою двигуна бази даних.

# 2 ПРОЕКТУВАННЯ БАЗИ ДАНИХ

## 2.1 Аналіз інформаційних та датологічних процесів

Основні потоки даних, які наявні у розроблюваній системі були описані у попередньому пункті при описанні предметної області системи. Найбільш основні деталі були показані за допомогою діаграм IDEF0 та IDEF3.

У цьому пункті більшу увагу приділимо опису датологічних процесів та поступово будемо зв’язувати їх із майбутньою базою даних.

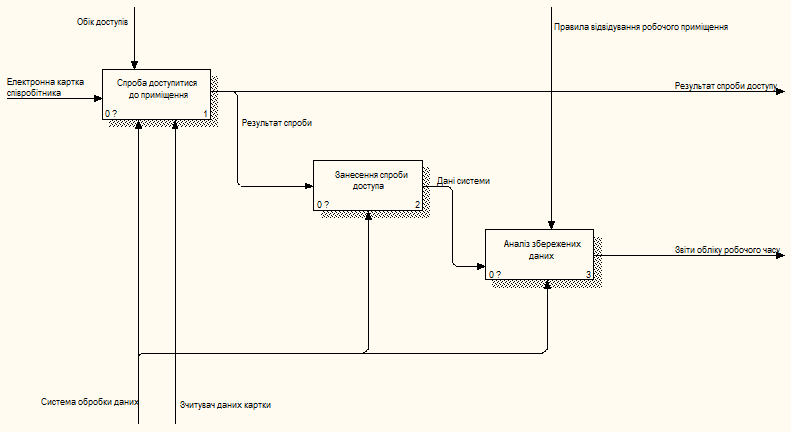


Рисунок 2.1 – Основні інформаційні стани системи та датологічні потоки

Опишемо основний потік даних, який буде проходити у системі.

Коли співробітник піднесе карту до зчитувального апарату, дані про картку будуть зчитані та збережені локально.

Надалі збережені дані проходять обробку для пропуску співробітника, або відмови у пропуску. Зчитуються усі картки які мають доступ до «проходу» на даному апараті, та порівнюються із зчитаною карткою.

Після вищезазначених операцій проводиться збереження даних. Для збереження формується група даних: поточна дата, поточний автомат перепустки, ідентифікатор зчитаної картки.

Сформовані дані переправляються на сервер для обробки.

Надалі, при необхідності, зібрані дані аналізуються. Проводяться запити на формування різник звітів, статистик. Для цих цілей проводиться суміщення власників карток та їх карток, їх правил відвідування офісу.

## 2.2 Проектування структури бази даних Київської обласної дирекції служби зайнятості населення

Аналізуючи предметну область, виведемо необхідні дані, які на необхідно зберігати. Обрані дані для збереження надалі будуть прес давлені у схемі бази даних для проектуємої системи.

Після розгляду вищезазначених вимог було виділено три широкі функціональні категорії:

1. Керування користувачами - як портал управляє користувачами, тобто шукачами роботи, робітників HR, а також незалежними або консалтинговими вербувальниками. (Для цілей даної моделі окремі представники HR та незалежні або консалтингові вербувальники розглядаються як компанії, принаймні з точки зору використання порталу.)
2. Створення профілів - як портал дозволяє шукачам роботи та організаціям створювати профілі та резюме.
3. Публікація та пошук робочих місць - як портал полегшує процес публікації, пошуку та подання заявки на роботу.

Зобразимо логічну структуру бази даних за допомогою Entity Relationship діаграми (рисунок 2.2).

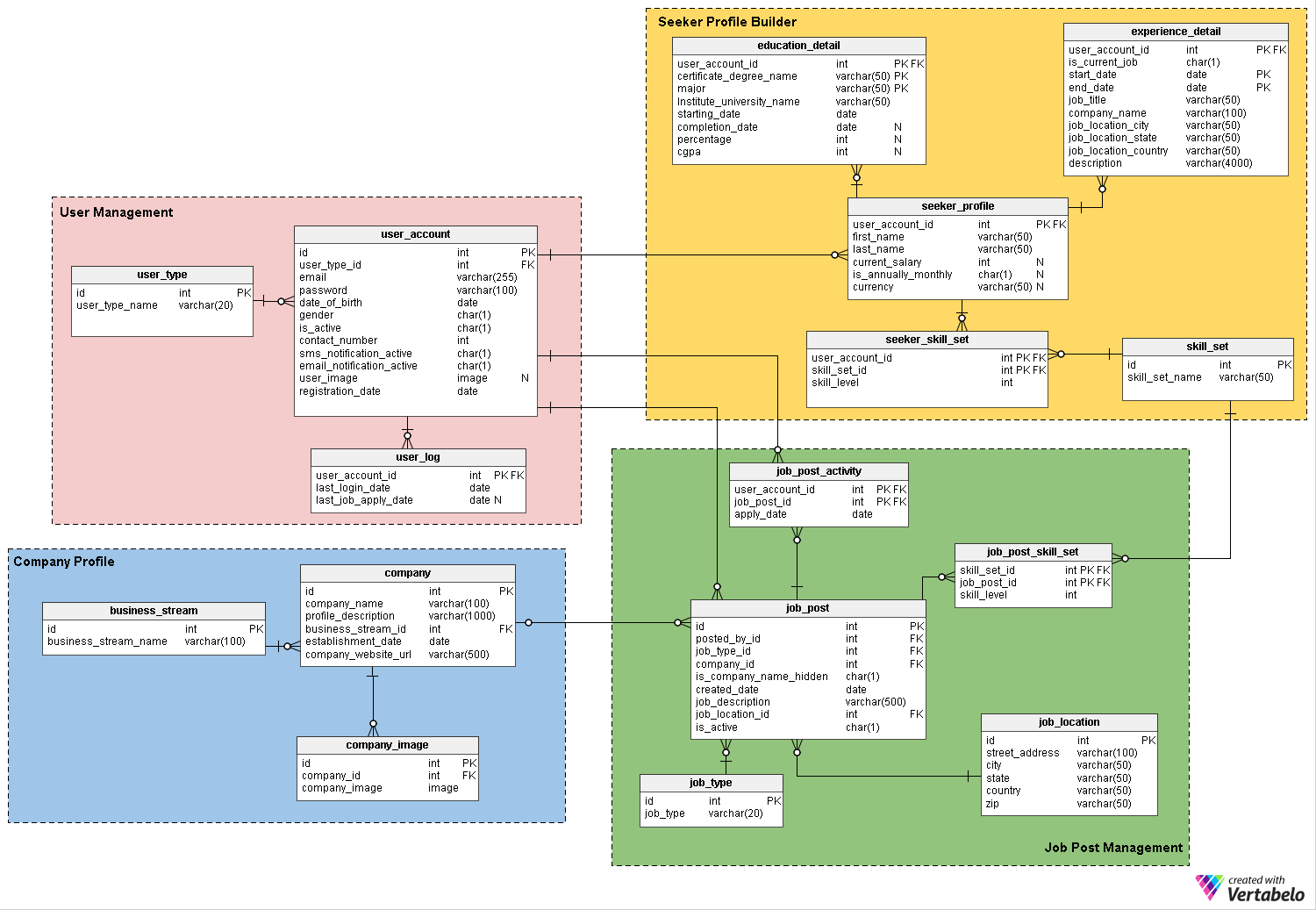


Рисунок 2.2 – Логічна структура бази даних, описана за допомогою Entity Relationship діаграми

Давайте розглянемо кожну з цих областей окремо.

1. Керування користувачами

Існує, перш за все, два типи користувачів інтернет-порталу: індивідуальні шукачі роботи та HR-рекрутери (або незалежні консультанти з підбору персоналу). Давайте створимо таблицю з ім'ям user\_type для зберігання цих записів. Для початку він буде мати два рекорди - одне для шукачів роботи, а інше для рекрутерів. (Ми завжди можемо створювати додаткові типи записів за потребою.)

Користувачі повинні зареєструватися, перш ніж користуватись порталом. У таблиці user\_account зберігаються основні дані облікового запису.

У таблиці user\_account є такі стовпці:

* id – являє собою як первинний ключ таблиці, так і унікальний ідентифікатор для кожного користувача. Цей ідентифікатор буде використаний іншими таблицями в моделі даних.
* user\_type\_id – визначає, чи є користувач шукачем роботи або рекрутером.
* email - цей стовпець містить електронну адресу користувача. Він діє як альтернативний ідентифікатор користувача для порталу.
* пароль - зберігає зашифрований пароль облікового запису (створеного користувачами під час реєстрації).
* date\_of\_birth та gender - ці стовпці містять дату народження та стать користувача.
* is\_active - спочатку цей стовпець буде "Y", але користувачі можуть встановити свій профіль неактивним або "N". Цей стовпець зберігає їхній вибір.
* contact\_number - це номер телефону (зазвичай мобільний), що надається під час реєстрації. Користувачі можуть отримувати SMS (текстові) сповіщення на цей номер. Це може бути той самий номер (чи ні), що шукачі роботи вказують у своєму профілі або резюме.
* sms\_notification\_active та email\_notification\_active - ці стовпці зберігають налаштування користувачів щодо одержання сповіщень через текст та / або електронну пошту.
* user\_image - це атрибут типу BLOB, в якому зберігається зображення профілю кожного користувача. Оскільки цей портал допускає лише одне зображення профілю для кожного користувача, має сенс його зберігати тут.
* registration\_date - цей стовпець зберігає запис про те, коли користувач зареєструвався на сайті.

Ми створимо ще одну таблицю - user\_log, в якій зберігається запис останньої дати входу користувачів та їх останню дату подання заявки на роботу. Є багато функцій, які можуть бути побудовані з цих знань. Наприклад, ми можемо використовувати цю інформацію, щоб відповісти на запитання «Чи активно користувач X шукає роботу?» Якщо так, то їм може запропонувати продукт для створення ефективного резюме. Користувачі, які не активно шукають роботу, не отримають такої пропозиції.

1. Створення профілів

Ми можемо додатково поділити цей розділ на дві області: профіль компанії або організації, а також профілі шукачів роботи.

Профіль компанії.

Зазвичай HR команди формують профілі компаній, вносячи деталі про свою організацію та зображення своїх офісів, будівель тощо. Їх головна мета - залучення хороших талантів. Коли рекрутери реєструються на порталі, вони також можуть створювати профілі своїх компаній (або їх особистий бренд, якщо вони незалежні), надаючи деякі основні відомості про те, як довго вони займаються бізнесом, їх місцезнаходження та основний бізнес-потік (наприклад, виробництво, ІТ-послуги, фінансові та інші).

Портал дозволяє HR та консультуючим рекрутерам завантажувати стільки зображень, скільки їм подобається (на відміну від тих, хто шукає роботу, вони можуть завантажувати лише одне). Тому ми створили таблицю company\_image для зберігання кількох зображень для кожного облікового запису рекрутера. Стовпець company\_id у цій таблиці є зовнішнім ключем, який посилається на унікальний ідентифікатор, який використовується в таблиці company.

У таблиці company є такі стовпці:

* id - первинний ключ цієї таблиці, також використовується для унікального визначення компаній.
* company\_name - як випливає з назви стовпця, це - юридична назва компанії.
* profile\_description - короткий опис кожної компанії.
* business\_stream\_id - у цьому стовпчику показано, який бізнес-потік належить компанії. Наприклад, компанія з розвідки нафти та газу може найняти інженерів з інформаційних технологій, але основним напрямком їх діяльності залишається "Нафта та газ".
* establishment\_date - у цьому стовпчику повідомляється, скільки років компанії.
* company\_website\_url - це обов'язковий (не підлягає скасуванню) стовпець. Він вказує на офіційний веб-сайт компанії, тому шукачі роботи можуть отримати більше інформації.

Нарешті, таблиця business\_stream має лише два атрибути: ідентифікатор, який є основним ключем для цієї таблиці, а також опис основного бізнес-потоку компанії (business\_stream\_name).

Профіль шукачів роботи.

Резюме – найбільш критична частина системи. Якщо сайт не зможе зберегти якомога більше інформації про шукачів роботи, рекрутерам буде важко оптимально знаходити кандидатів.

Таблиця seeker\_profile містить додаткові відомості, які не були зроблені під час процесу реєстрації. Вона містить такі поля:

* user\_account\_id - цей стовпець передається з таблиці user\_account, і він діє як основний ключ для цієї таблиці. Це гарантує тільки одного профіль для кожного шукача роботи.
* first\_name and last\_name - ці стовпці містять імена та прізвища шукачів роботи.
* current\_salary - Цей атрибут містить поточну зарплату шукача роботи. Це значення - nullable, оскільки люди, можливо, не хочуть його розкривати.
* is\_annually\_monthly - визначає, чи зарплата визначається на рік або на місяць.
* currency - зберігає валюту заробітної плати.

У таблиці "education\_detail" зберігаються історія освіти шукачів роботи, принаймні яку вони надають. Він має складний первинний ключ, що складається з user\_account\_id, name\_ certificate\_degree\_name і major стовпчиків. Це гарантує, що користувачі вводять лише один запис за кожним ступенем або сертифікатом. Таблиця містить такі атрибути:

* user\_account\_id - цей стовпець передається з таблиці user\_account і слугує основним ключем для цієї таблиці.
* certificate\_degree\_name - це сертифікат або тип ступеня; наприклад, середня школа, вища середня, аспірантура, аспірантура або професійний сертифікат.
* major - Ця колонка містить основний курс навчання за сертифікатом або ступенем - наприклад, ступінь бакалавра з фахом з комп'ютерних наук.
* institute\_university\_name - це інститут, школа або університет, у якому користувач отримав ступень чи сертифікат.
* start\_date - цей атрибут зберігає дату, коли користувач був прийнятий у навчальну програму.
* completion\_date - це дата отримання ступеня чи сертифікату. Однак цей атрибут є nullable; люди можуть продовжувати виконувати свою програму, коли шукають роботу, або вони взагалі можуть випасти з програми.
* mark – цей стовпчик містить загальну середню оцінку, що досягається користувачами на курсах їх ступеня або сертифікату.

У таблиці experience\_detail зберігаються записи про минулий та поточний професійний досвід користувачів. Він містить такі важливі стовпці:

* user\_account\_id - цей стовпець передається з таблиці user\_account і є основним ключем для цієї таблиці.
* is\_current\_job - це стовпчик-індикатор, який вказує поточну роботу користувача. Цей стовпець також відіграє важливу роль у виведенні поточних розташувань користувачів і про час їхньої поточної позиції.
* start\_date – зберігає дату, коли користувач починає роботу.
* end\_date - зберігає дату, коли користувач закінчує роботу.
* job\_title - містить інформацію про ролі роботи користувача.
* company\_name - атрибут містить відповідне ім'я компанії, пов'язане з роботою.
* job\_location\_city - місто, де була розташована робота.
* job\_location\_state - область, де була розташована робота.
* job\_location\_country - країна, де ця робота була розташована.
* description - у цьому стовпчику зберігаються відомості про ролі роботи та обов'язки, проблеми та досягнення.

Шукачі роботи можуть володіти декількома навичками. Щоб вести облік усіх цих наборів навичок, ми створимо таблицю seeker\_skill\_set. Колонки:

* user\_account\_id - цей стовпець передається з таблиці user\_account і є основним ключем для цієї таблиці.
* skill\_set\_id - цей ідентифікатор означає, який набір навичок має користувач.
* skill\_level - цей числовий атрибут кількісно визначає досвід роботи шукачів у певній кваліфікації. Число від 1 (початківець) до 10 (експерт) вказує на рівень їхньої кваліфікації.

Нарешті, таблиця skill\_set містить описи всіх навичок, наведених у атрибуті skill\_set\_id наведеної таблиці. Вона містить лише два стовпці, ім'я skill\_set\_name та пов'язаний з ним ідентифікатор.

1. Публікація та пошук робочих місць

Це головна особливість порталу пошуку роботи. Тільки зареєстровані вербувальники можуть розміщувати роботу на порталі, і лише зареєстровані шукачі роботи можуть звертатися до них.

Таблиця job\_post є основною таблицею у цій темі. Як ви можете здогадатися, вона містить інформацію про роботу. Всі інші таблиці в цьому розділі створені навколо неї та пов'язані з нею.

id - первинний ключ цієї таблиці. Кожен пост роботи призначається унікальним номером, а цей номер використовується в інших таблицях.

posted\_by\_id - у цьому стовпці зберігається register\_user\_id рекрутера, який опублікував роботу.

job\_type\_id - цей стовпець визначає, чи тривалість роботи є постійною або тимчасовою (контракт).

company\_id - у цьому стовпчику зберігається ідентифікатор компанії, пов'язаний з посадою вакансії. Це посилання на таблицю компаній.

is\_company\_name\_hidden – стовпчик-прапорець, який показує, чи повинно ім'я компанії бути показано шукачам роботи. Рекрутери можуть віддати перевагу не показувати назви компаній у своїх публікаціях. Замість цього вони використовують такі терміни, як «Глобальна автомобільна компанія».

created\_date - зберігає дату публікації.

job\_description - короткий опис роботи.

job\_location\_id - посилання на атрибут у таблиці job\_location, де зберігається фактичне місце роботи: адресу, місто, область, країна та поштовий індекс.

is\_active - Це означає, що робота все ще відкрита. Рекрутери можуть відмічати свої посади неактивними, як тільки позиції будуть заповнені.

Таблиця work\_post\_skill\_set зберігає подробиці про набори навичок, необхідні для роботи. Структура таблиці ідентична таблиці seeker\_skill\_set.

І остання таблиця у цьому розділі, job\_post\_activity, містить відомості про тих, хто шукає роботу, і коли.

Тоді образимо фізичну діаграму бази даних на рисунку 2.3:

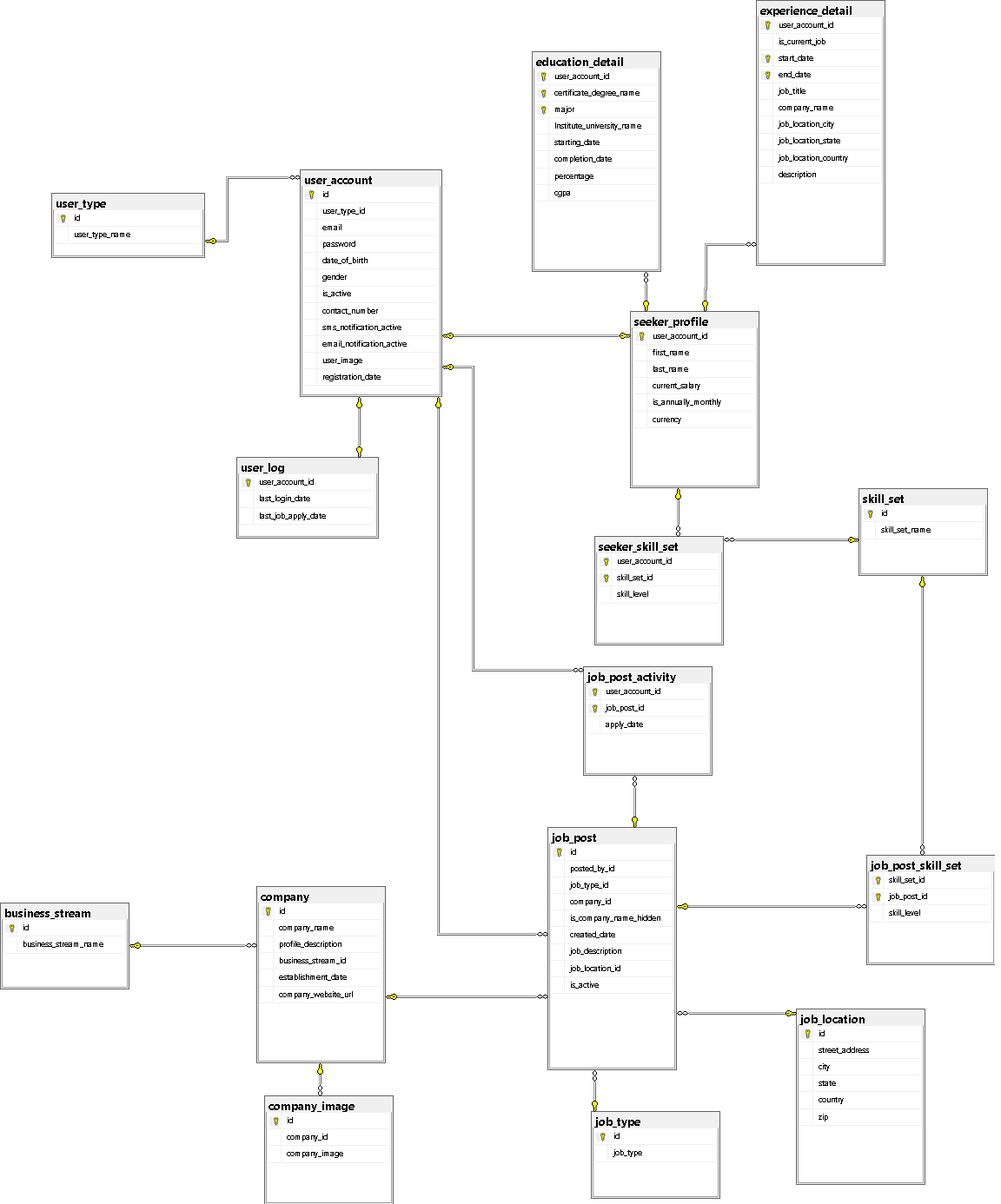


Рисунок 2.3 – Фізична діаграма бази даних