МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ПОЛІСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

Факультет інформаційних технологій,
обліку та фінансів
Кафедра комп'ютерних технологій
і моделювання систем

Кваліфікаційна робота
на правах рукопису

Ровков Олександр Олександрович
(прізвище, ім'я, по батькові здобувача освіти)

УДК 004.9:336.7

КВАЛІФІКАЦІЙНА РОБОТА

Проєктування інформаційної системи інтеграції відкритих даних у єдину клієнтську базу фінансових установ.

(тема роботи)

122 «Комп'ютерні науки»

(шифр і назва спеціальності)

Подається на здобуття освітнього ступеня бакалавр

кваліфікаційна робота містить результати власних досліджень. Використання ідей, результатів і текстів інших авторів мають посилання на відповідне джерело

_____

(підпис, ініціали та прізвище здобувача вищої освіти)

Керівник роботи
Гіваргізов Інвія Геннадійович
(прізвище, ім'я, по батькові)
к.е.н., старший викладач
(науковий ступінь, вчене звання)

Житомир – 2026

Висновок кафедри комп'ютерних технологій і моделювання систем
за результатами попереднього захисту: допущений

Протокол засідання кафедри комп'ютерних технологій і моделювання систем
№ 20 від «05» червня 2026 р.

Завідувач кафедри комп'ютерних технологій і моделювання систем
к. пед н., доцент _____ КОВАЛЬЧУК М. О.
(науковий ступінь, вчене звання) (підпис) (прізвище, ім'я, по батькові)
«05» червня 2026 р.

Результати захисту кваліфікаційної роботи

Здобувач вищої освіти _____ захистив (ла)
(прізвище ,ім'я, по батькові)
кваліфікаційну роботу з оцінкою:

сума балів за 100-бальною шкалою _____
за шкалою ECTS _____
за національною шкалою _____

Секретар ЕК
лаборант кафедри _____ КОРОЛЬЧУК В. В.
(науковий ступінь, вчене звання) (підпис) (прізвище, ім'я, по батькові)

АНОТАЦІЯ

Ровков О.О. Проєктування інформаційної системи інтеграції відкритих даних у єдину клієнтську базу фінансових установ. – Кваліфікаційна робота на правах рукопису.

Кваліфікаційна робота на здобуття освітнього ступеня бакалавр за спеціальністю 122 – Комп'ютерні науки. – Поліський національний університет, Житомир, 2026.

Актуальність кваліфікаційної роботи зумовлена необхідністю автоматизації процесів перевірки клієнтів фінансових установ на основі відкритих даних. Фінансові установи працюють з великою кількістю клієнтів і контрагентів, тому потребують швидкого доступу до інформації про їхній статус, наявність ризикових ознак і боргових записів, які можуть впливати на прийняття рішень. Ручний пошук такої інформації у різних відкритих джерелах потребує значних витрат часу та може призводити до помилок.

Кваліфікаційна робота спрямована на проєктування та розробку інформаційної системи інтеграції відкритих даних у єдину клієнтську базу фінансових установ. У роботі проаналізовано предметну область, визначено основні джерела відкритих даних, сформовано вимоги до системи, спроєктовано структуру бази даних та розроблено модель оцінювання ризику клієнта. Розроблена система забезпечує пошук клієнта за кодом ЄДРПОУ, завантаження й обробку XML/CSV-масивів, виявлення ризик-маркерів, розрахунок інтегрального ризикового бала, визначення рівня ризику, формування PDF-звіту та збереження історії виконаних перевірок.

Практичне значення роботи полягає в тому, що розроблена інформаційна система може використовуватися як інструмент для автоматизації первинної комплаєнс-перевірки клієнтів фінансових установ. Вона дозволяє структурувати інформацію з відкритих джерел, зменшити обсяг ручної роботи, підвищити швидкість перевірки та надати користувачу зрозумілий ризиковий профіль клієнта.

Робота містить 44 сторінки, 15 рисунків, 6 таблиць, 40 літературних джерел, 7

додатків.

Ключові слова: відкриті дані, інформаційна система, клієнтська база, ризик-маркер, фінансова установа.

SUMMARY

Rovkov O.O. Designing an information system for integrating open data into a unified client database of financial institutions. – Qualification work as a manuscript.

Qualification work for a bachelor's degree in specialty 122 – Computer Science. – Polissia National University, Zhytomyr, 2026.

The relevance of the qualification work is determined by the need to automate the process of verifying clients of financial institutions using open data. Financial institutions work with a large number of clients and counterparties, so they need quick access to information about their status, possible risk indicators, debt records, and other data that may affect decision-making. Manual search and comparison of such information from different open sources requires additional time and may lead to errors.

The qualification work is aimed at designing and developing an information system for integrating open data into a unified client database of financial institutions. The work analyzes the subject area, defines the main open data sources, forms system requirements, designs the database structure, and develops a client risk assessment model. The developed system provides client search by EDRPOU code, XML/CSV data processing, risk marker detection, integral risk score calculation, risk level determination, PDF report generation, and storage of verification history.

The practical significance of the work is that the developed information system can be used as a tool for automating the initial compliance verification of clients of financial institutions. It helps structure information from open sources, reduce manual work, speed up verification, and provide the user with a clear client risk profile.

The work contains 44 pages, 15 figures, 6 tables, 40 references, and 7 appendices.

Keywords: client database, financial institution, information system, open data, risk marker.

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ

БД — база даних

ЄДР — Єдиний державний реєстр

ЄДРПОУ — Єдиний державний реєстр підприємств та організацій України

ІС — інформаційна система

ПЗ — програмне забезпечення

СКБД — система керування базами даних

CSV — Comma-Separated Values, формат текстових даних із розділенням значень комами

ETL — Extract, Transform, Load; процес вилучення, перетворення та завантаження даних

HTML — HyperText Markup Language, мова гіпертекстової розмітки

IDEF0 — методологія функціонального моделювання інформаційних систем

PDF — Portable Document Format, формат електронного документа

SQL — Structured Query Language, мова структурованих запитів

SQLite — вбудована реляційна система керування базами даних

UML — Unified Modeling Language, уніфікована мова моделювання

XML — Extensible Markup Language, розширювана мова розмітки

ВСТУП

Фінансові установи у своїй діяльності постійно працюють з великою кількістю клієнтів і контрагентів, тому потребують швидкої та достовірної перевірки інформації про них. Для прийняття обґрунтованих рішень важливо враховувати дані про статус суб'єкта, наявність боргових записів, виконавчих проваджень та інших ризикових ознак. Такі відомості можуть міститися у відкритих реєстрах та офіційних наборах відкритих даних, однак часто вони розміщені в різних джерелах, мають різну структуру та потребують ручного пошуку й аналізу [1; 5; 6].

Актуальність теми полягає в тому, що фінансові установи повинні своєчасно перевіряти клієнтів і контрагентів з урахуванням вимог комплаєнсу, фінансового моніторингу та ризик-орієнтованого підходу. Збільшення кількості клієнтських перевірок, необхідність аналізу даних з різних відкритих реєстрів, наявність регуляторних вимог і ризик помилок під час ручного зіставлення інформації створюють потребу в більш структурованому та швидкому способі обробки таких даних. У зв'язку з цим актуальним є розроблення інформаційної системи, яка забезпечує інтеграцію відкритих даних, формування єдиної клієнтської бази, виявлення ризик-маркерів та підтримку прийняття рішень під час первинної комплаєнс-перевірки клієнтів фінансових установ [3; 4].

Метою кваліфікаційної роботи є дослідження процесів комплаєнс-перевірки клієнтів фінансових установ та розробка інформаційної системи інтеграції відкритих даних у єдину клієнтську базу для автоматизованого виявлення ризик-маркерів і оцінювання ризику клієнта.

Для досягнення поставленої мети було визначено такі завдання: проаналізувати відкриті джерела даних для перевірки клієнтів фінансових установ; сформувати вимоги до інформаційної системи; спроєктувати структуру єдиної клієнтської бази; розробити механізм завантаження, очищення та нормалізації даних; реалізувати виявлення ризик-маркерів; розробити модель оцінювання ризику клієнта; створити вебінтерфейс для пошуку клієнта, перегляду результатів перевірки, формування PDF-

звіту та збереження історії перевірок.

Об’єктом дослідження є процеси комплаєнс-перевірки клієнтів фінансових установ з використанням відкритих реєстрових даних.

Предметом дослідження є методи, моделі та програмні засоби інтеграції відкритих даних, формування єдиної клієнтської бази та оцінювання ризику клієнта.

У роботі використано метод аналізу предметної області для визначення особливостей перевірки клієнтів фінансових установ; метод системного підходу — для визначення структури інформаційної системи та взаємозв’язку її компонентів; функціональне моделювання IDEF0 — для опису процесу інтеграції відкритих даних; інфологічне та даталогічне моделювання — для проєктування бази даних; методи ETL-обробки — для завантаження, очищення та нормалізації даних; методи вебпрограмування — для реалізації інтерфейсу системи; метод тестування — для перевірки працездатності розробленої системи.

Практичне значення роботи полягає в розробці інформаційної системи, яка може використовуватися як інструмент для автоматизації первинної комплаєнс-перевірки клієнтів фінансових установ. Система забезпечує пошук клієнта за кодом ЄДРПОУ, інтеграцію XML/CSV-масивів даних, виявлення ризик-маркерів, оцінювання ризику, формування PDF-звіту та збереження історії перевірок. Під час обробки інформації про клієнтів необхідно враховувати вимоги законодавства щодо захисту персональних даних [2].

Окремі результати кваліфікаційної роботи опубліковано у тезах науково-практичних конференцій за темою дослідження [16; 17].

Кваліфікаційна робота складається з анотації, змісту, переліку умовних позначень, вступу, трьох розділів, висновків, списку використаних джерел і додатків. Загальний обсяг роботи становить 68 сторінок, робота містить 23 рисунки, 6 таблиць, 40 джерел та 7 додатків.

РОЗДІЛ 1. АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ІНТЕГРАЦІЇ ВІДКРИТИХ ДАНИХ У ДІЯЛЬНОСТІ ФІНАНСОВИХ УСТАНОВ

1.1. Роль відкритих даних у процесах перевірки клієнтів фінансових установ

У діяльності фінансових установ важливе місце займає перевірка клієнтів і контрагентів перед початком співпраці або наданням фінансових послуг. Така перевірка дає змогу оцінити надійність клієнта, виявити можливі ризикові ознаки та зменшити ймовірність прийняття необґрунтованих рішень. Особливо це актуально для банків, кредитних організацій, страхових компаній та інших установ, які працюють з фінансовими операціями і несуть відповідальність за дотримання вимог внутрішнього контролю та комплаєнсу.

Одним із джерел інформації для такої перевірки є відкриті дані. В Україні порядок оприлюднення публічної інформації у формі відкритих даних визначається законодавством та відповідними нормативними актами [1; 18]. До них можна віднести відомості з державних реєстрів, дані про суб’єктів господарювання, інформацію про статус юридичних осіб та фізичних осіб-підприємців[7; 8], записи про боржників, виконавчі провадження та інші відомості, що мають значення для оцінювання ризику клієнта [9]. Використання таких даних дозволяє отримати додаткову інформацію про клієнта ще до прийняття рішення щодо співпраці з ним.

Відкриті дані є корисними для фінансових установ, оскільки вони дають змогу перевірити не лише базову інформацію про клієнта, а й виявити ознаки, які можуть свідчити про підвищений ризик. Наприклад, неактивний або припинений статус суб’єкта господарювання може вказувати на обмеження його діяльності, а наявність записів у реєстрі боржників може свідчити про фінансові зобов’язання або проблеми з виконанням зобов’язань [9]. Такі відомості не завжди є достатніми для остаточного рішення, однак вони є важливими для первинної оцінки клієнта.

Разом з тим практичне використання відкритих даних має певні складнощі. Дані можуть зберігатися у різних форматах, мати різну структуру, дублюватися або

містити неповні відомості [22; 23]. Крім того, для отримання повної картини про клієнта часто потрібно звертатися до кількох джерел одночасно. Якщо така перевірка виконується вручну, це потребує додаткового часу і підвищує ризик помилок під час пошуку, зіставлення та аналізу інформації.

Саме тому виникає потреба в інтеграції відкритих даних у єдину клієнтську базу. Такий підхід дозволяє зібрати інформацію з різних джерел, привести її до єдиної структури та використовувати для автоматизованої перевірки клієнта. У результаті користувач отримує не окремі фрагменти інформації з різних реєстрів, а узагальнений профіль клієнта з виявленими ризик-маркерами.

Отже, відкриті дані відіграють важливу роль у процесах перевірки клієнтів фінансових установ. Їх використання дає змогу підвищити якість первинного аналізу, скоротити час перевірки та забезпечити більш обґрунтоване прийняття рішень. Для ефективного використання таких даних доцільним є створення інформаційної системи, яка автоматизує їх завантаження, обробку, зіставлення та подання результатів у зручному для користувача вигляді.

1.2. Особливості комплаєнс-перевірки клієнтів та виявлення ризикових ознак

Комплаєнс-перевірка клієнтів є важливою складовою діяльності фінансових установ, оскільки вона спрямована на зменшення ризиків під час встановлення ділових відносин або надання фінансових послуг [3; 4]. Її основне завдання полягає не лише у зборі загальної інформації про клієнта, а й у виявленні факторів, які можуть свідчити про потенційну ненадійність, фінансові труднощі або інші обставини, що потребують додаткової уваги.

У процесі первинної перевірки клієнта фінансова установа зазвичай аналізує ідентифікаційні дані, актуальний статус суб’єкта, відомості про керівника, місцезнаходження, а також інформацію з відкритих реєстрів [7; 8]. Окрему увагу приділяють наявності записів у реєстрах боржників, виконавчих провадженнях або

інших джерелах, які можуть вказувати на фінансові зобов'язання чи проблеми з їх виконанням [9]. Такі дані не завжди є підставою для відмови у співпраці, однак вони є важливими для формування загального уявлення про рівень ризику клієнта.

Ризиковими ознаками, або ризик-маркерами, у межах такої перевірки можна вважати конкретні виявлені факти, які підвищують імовірність негативного сценарію під час співпраці з клієнтом. До таких маркерів належать, наприклад, неактивний або припинений статус суб'єкта господарювання, наявність запису в реєстрі боржників, збіг клієнта з боржником лише за назвою, а також велика кількість знайдених ризикових записів. Кожен із цих маркерів має різне значення для оцінювання ризику, тому доцільно враховувати не тільки сам факт його наявності, а й ступінь його впливу.

Особливістю комплаєнс-перевірки є те, що інформація про клієнта може бути неповною або неоднозначною. Наприклад, збіг за кодом ЄДРПОУ є більш точним, ніж збіг лише за назвою, оскільки назви суб'єктів можуть бути подібними або змінюватися [7; 8]. Тому під час автоматизованої перевірки важливо враховувати тип збігу даних і відокремлювати точні збіги від потенційних, які потребують додаткової перевірки користувачем.

Для підвищення ефективності перевірки доцільно використовувати інформаційну систему, яка автоматично зіставляє дані з різних джерел, визначає ризик-маркери та формує узагальнений результат. Такий підхід дозволяє не лише скоротити час пошуку інформації, а й зробити сам процес перевірки більш послідовним і зрозумілим. Користувач отримує не просто набір знайдених записів, а структурований результат із зазначенням виявлених ризикових ознак.

Отже, комплаєнс-перевірка клієнтів фінансових установ потребує аналізу різних груп даних і правильного визначення ризикових ознак. Автоматизоване виявлення ризик-маркерів дає змогу швидше оцінити клієнта, зменшити навантаження на працівника та підвищити обґрунтованість прийняття рішень.

1.3. Аналіз існуючих підходів до інтеграції реєстрових даних

Інтеграція реєстрових даних є важливим етапом створення інформаційних систем, які працюють з інформацією з різних джерел. У випадку фінансових установ така інтеграція дає змогу об’єднати відомості про клієнтів та їхній статус [7; 8], можливі боргові записи та інші ризикові ознаки [9] в єдиному інформаційному середовищі. Це спрощує процес перевірки клієнта і дозволяє користувачу швидше отримати узагальнений результат.

Одним із найпростіших підходів є ручний пошук інформації у відкритих реєстрах. У цьому випадку працівник самостійно вводить дані клієнта в різних джерелах, порівнює знайдену інформацію та робить висновок щодо наявності ризикових ознак. Такий підхід не потребує складної технічної реалізації, однак має суттєві недоліки. Він займає багато часу, залежить від уважності користувача та не забезпечує єдиного формату збереження результатів перевірки.

Іншим підходом є використання готових зовнішніх сервісів для перевірки контрагентів. До таких сервісів можна віднести YouControl, Opendatabot та VKURSI. Вони надають користувачу доступ до зведеної інформації про юридичних осіб, фізичних осіб-підприємців, боргові записи, судові справи, податкові відомості та інші дані, які можуть використовуватися під час первинної перевірки клієнта.

YouControl є аналітичною системою для перевірки компаній і контрагентів, яка формує досьє на суб’єкта господарювання та дозволяє оцінювати його за різними відкритими джерелами. Opendatabot орієнтований на швидку перевірку компаній, ФОП, судових рішень, нерухомості, боргів та інших відкритих даних. VKURSI також надає можливості перевірки компаній і фізичних осіб, моніторингу змін та інтеграції даних у внутрішні інформаційні системи підприємств.

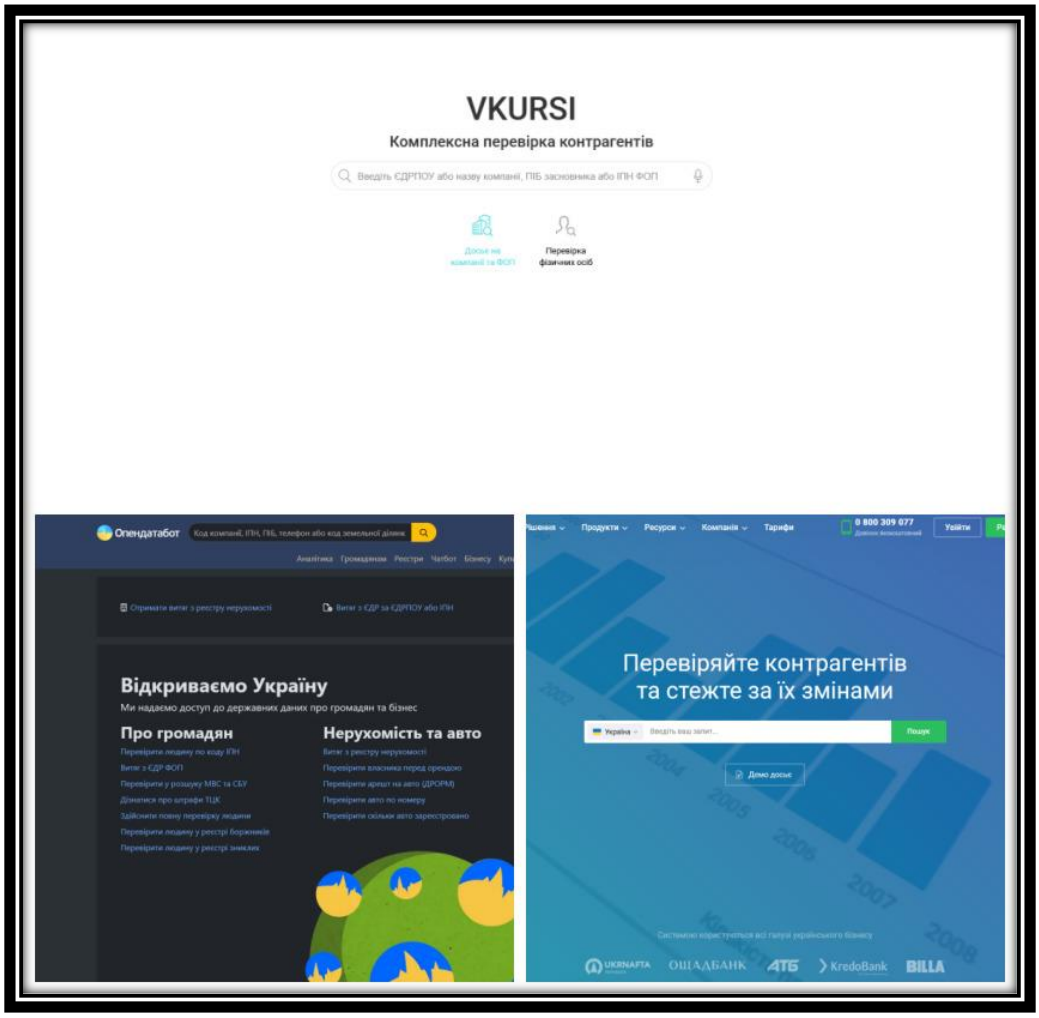


Рис. 1.1. Приклади зовнішніх сервісів для перевірки контрагентів

Перевагою таких сервісів є зручний інтерфейс, швидкий доступ до великої кількості джерел, наявність готових досьє та можливість моніторингу змін за контрагентом. Вони можуть бути корисними для підприємств, які потребують швидкого отримання узагальненої інформації без самостійної розробки програмного забезпечення.

Водночас використання готових сервісів має певні недоліки. До них належать залежність від стороннього постачальника, можлива платність доступу до розширених функцій, обмеженість налаштування власної моделі оцінювання ризику та неможливість повного контролю над структурою обробки даних. Крім того, фінансова установа не завжди може адаптувати зовнішній сервіс під власні внутрішні правила перевірки клієнтів.

Таблиця 1.1 - Порівняльна характеристика зовнішніх сервісів перевірки контрагентів

Сервіс	**Основне призначення**	**Переваги**	**Недоліки**
YouControl	Формування досьє на компанію або ФОП, перевірка контрагентів за відкритими даними	Велика кількість джерел, аналітичне подання інформації, зручне досьє контрагента	Залежність від зовнішнього сервісу, платний доступ до частини функцій
Opendatabot	Швидка перевірка компаній, ФОП, боргів, судових рішень та інших відкритих даних	Простий пошук, швидкий доступ до основних даних, зручність для первинної перевірки	Обмежені можливості налаштування власної моделі оцінювання ризику
VKURSI	Перевірка компаній і фізичних осіб, моніторинг змін, інтеграція даних	Можливість моніторингу, перевірка в реальному часі, наявність API-інтеграції	Потребує зовнішнього доступу, залежить від умов сервісу та тарифів

Отже, готові сервіси перевірки контрагентів є зручними для швидкого отримання інформації, однак вони не завжди забезпечують потрібну гнучкість для побудови власної моделі оцінювання ризику. Саме тому в межах кваліфікаційної роботи доцільним є розроблення власної інформаційної системи, яка дозволяє самостійно визначати структуру клієнтської бази, правила обробки відкритих даних, набір ризик-маркерів і логіку формування ризикового бала.

Також поширеним є підхід, за якого дані з відкритих джерел завантажуються у вигляді файлів або масивів даних, після чого проходять етапи очищення, нормалізації та внесення до внутрішньої бази даних [5; 18]. Такий підхід є більш гнучким, оскільки дозволяє самостійно визначити структуру бази, правила обробки інформації та механізм оцінювання ризику. Саме він є доцільним для розробки системи, що має не

лише зберігати дані, а й виконувати їх аналіз.

У процесі інтеграції реєстрових даних важливо враховувати різні формати подання інформації [22; 23]. Наприклад, дані можуть надходити у форматах XML, CSV або зберігатися в таблицях бази даних [22; 23; 33; 34]. Для подальшого використання їх потрібно привести до єдиної структури: визначити ключові поля, усунути дублювання, нормалізувати назви, коди, статуси та інші атрибути. Це дозволяє коректно виконувати пошук клієнта та зіставлення записів з різних джерел.

Окрему роль відіграє ідентифікація клієнта. Найбільш точним способом зіставлення є пошук за унікальним кодом ЄДРПОУ, оскільки він дає змогу однозначно визначити суб'єкта [7; 8]. Водночас у деяких випадках можуть використовуватися додаткові ознаки, наприклад назва або ПІБ. Такі збіги потребують обережного трактування, оскільки вони не завжди гарантують повну ідентичність записів.

Для розроблюваної інформаційної системи найбільш доцільним є підхід, що передбачає завантаження відкритих даних із підготовлених XML/CSV-масивів, їх очищення, нормалізацію та збереження в єдиній базі даних. На основі цієї бази система виконує пошук клієнта, визначає ризик-маркери та формує результат перевірки. Такий підхід забезпечує контроль над структурою даних, прозорість обробки інформації та можливість подальшого розширення системи новими джерелами.

Отже, аналіз існуючих підходів показує, що ручна перевірка є найпростішою, але недостатньо ефективною для системної роботи з великою кількістю клієнтів. Використання зовнішніх сервісів може бути зручним, проте не завжди забезпечує необхідну гнучкість. Тому для поставленого завдання доцільним є створення власної інформаційної системи, яка інтегрує відкриті дані в єдину клієнтську базу та забезпечує автоматизовану перевірку клієнтів.

1.4. Формування вимог до інформаційної системи

Після аналізу предметної області можна визначити основні вимоги до інформаційної системи інтеграції відкритих даних у єдину клієнтську базу фінансових установ. Такі вимоги мають враховувати як особливості роботи з реєстровими даними, так і потреби користувача, який виконує перевірку клієнта.

Насамперед система повинна забезпечувати збереження інформації про клієнтів у єдиній базі даних. До основних відомостей про клієнта належать код ЄДРПОУ, назва або ПІБ, статус діяльності, адреса та керівник [7; 8]. Наявність таких даних дозволяє сформувати базовий профіль клієнта і використовувати його для подальшого аналізу.

Важливою вимогою є можливість завантаження та обробки даних із зовнішніх джерел. У межах роботи передбачено використання XML/CSV-масивів, які імітують структуру відкритих реєстрових даних [22; 23]. Система має виконувати їх завантаження, очищення, нормалізацію та внесення до бази даних [12; 13; 33; 34]. Це дає змогу привести різні набори інформації до єдиного формату та використовувати їх у процесі перевірки клієнта.

Окремою функціональною вимогою є пошук клієнта за кодом ЄДРПОУ [7; 8]. Такий спосіб пошуку є найбільш точним, оскільки дозволяє однозначно ідентифікувати суб’єкта в базі даних. Після виконання пошуку система повинна відображати основні дані клієнта, знайдені записи з реєстрів та виявлені ризик-маркери.

Система також має підтримувати механізм оцінювання ризику клієнта [3; 4; 19]. Для цього необхідно враховувати наявність записів у реєстрі боржників, актуальний статус клієнта, тип збігу даних і кількість виявлених ризик-маркерів. Результатом такої перевірки має бути узагальнений рівень ризику, який допомагає користувачу швидше оцінити ситуацію та прийняти подальше рішення.

Для зручності роботи користувача система повинна мати вебінтерфейс [14; 28; 29]. Він має забезпечувати авторизацію користувача, пошук клієнта, перегляд результатів перевірки, доступ до історії виконаних перевірок і формування PDF-звіту.

Для адміністратора додатково має бути передбачена можливість оновлення даних та перегляду загальної статистики роботи системи.

До нефункціональних вимог можна віднести простоту використання, зрозуміле подання результатів, структурованість даних і можливість подальшого розширення системи. Зокрема, у майбутньому система може бути доповнена новими джерелами відкритих даних, додатковими ризик-маркерами або розширеною моделлю оцінювання ризику.

Отже, сформовані вимоги визначають основні напрями розробки інформаційної системи. Вона повинна не лише зберігати дані про клієнтів, а й забезпечувати їх інтеграцію, автоматизовану перевірку, виявлення ризикових ознак, формування звіту та збереження результатів перевірки.

Висновки до розділу 1

У першому розділі було проаналізовано предметну область використання відкритих даних у діяльності фінансових установ. Встановлено, що відкриті реєстрові дані можуть бути важливим джерелом інформації під час первинної перевірки клієнтів, оскільки містять відомості про статус суб’єкта, боргові записи, виконавчі провадження та інші потенційні ризикові ознаки.

Розглянуто особливості комплаєнс-перевірки клієнтів та визначено основні ризик-маркери, які доцільно враховувати під час оцінювання клієнта. До таких маркерів належать наявність записів у реєстрі боржників, неактивний або припинений статус діяльності, тип збігу даних та кількість виявлених ризикових записів.

Проаналізовано основні підходи до інтеграції реєстрових даних. Визначено, що ручна перевірка є недостатньо ефективною для роботи з великою кількістю клієнтів, а використання єдиної інформаційної системи дозволяє автоматизувати завантаження, обробку, зіставлення та аналіз даних.

На основі проведеного аналізу сформовано вимоги до інформаційної системи. Вона повинна забезпечувати збереження клієнтських даних, завантаження та

нормалізацію XML/CSV-масивів, пошук клієнта за кодом ЄДРПОУ, виявлення ризик-маркерів, оцінювання ризику, формування PDF-звіту та збереження історії виконаних перевірок.

РОЗДІЛ 2. ПРОЄКТУВАННЯ ІНФОРМАЦІЙНОЇ СИСТЕМИ ІНТЕГРАЦІЇ ВІДКРИТИХ ДАНИХ

2.1. Загальна архітектура інформаційної системи

Проєктування інформаційної системи інтеграції відкритих даних передбачає визначення її основних компонентів, зв’язків між ними та порядку обробки інформації. Розроблена система призначена для автоматизації первинної перевірки клієнтів фінансових установ на основі відкритих реєстрових даних. Її основне завдання полягає в тому, щоб зібрати дані з різних джерел, зберегти їх у єдиній базі, виконати пошук клієнта та сформувати результат перевірки у зручному для користувача вигляді.

Архітектура системи побудована за модульним принципом [16; 17]. Це дає змогу розділити основні функції між окремими частинами системи та спростити подальше розширення функціональності. До складу інформаційної системи входять модуль авторизації користувачів, модуль завантаження та обробки даних, база даних, модуль пошуку клієнта, модуль виявлення ризик-маркерів, модуль оцінювання ризику, модуль формування звіту та вебінтерфейс користувача.

Модуль авторизації забезпечує доступ до системи лише для зареєстрованих користувачів [30; 31]. У системі передбачено розмежування ролей, зокрема звичайного користувача та адміністратора. Звичайний користувач може виконувати пошук клієнта, переглядати результати перевірки та історію запитів. Адміністратор додатково має доступ до оновлення даних, створення користувачів та перегляду статистики роботи системи.

Модуль завантаження та обробки даних відповідає за роботу з XML/CSV-масивами, які містять відомості з відкритих джерел [22; 23]. На цьому етапі виконується зчитування даних, їх очищення, нормалізація та внесення до бази даних.

Такий підхід дозволяє привести інформацію з різних джерел до єдиної структури та забезпечити її подальше використання під час перевірки клієнта.

Центральним елементом системи є база даних, у якій зберігаються відомості про клієнтів, ризик-маркери, користувачів і журнал виконаних перевірок [12; 13]. Наявність єдиної бази даних дозволяє уникнути дублювання інформації, забезпечити швидкий пошук записів і зберігати результати попередніх перевірок.

Модуль пошуку клієнта забезпечує ідентифікацію клієнта за кодом ЄДРПОУ. Після введення коду система виконує пошук у клієнтській базі, отримує основні дані про суб'єкта та перевіряє наявність пов'язаних ризик-маркерів. Якщо клієнта знайдено, система формує його інформаційний профіль, який містить базові відомості та результати перевірки.

Модуль виявлення ризик-маркерів аналізує дані клієнта та пов'язані записи з реєстрів. До ризик-маркерів належать наявність записів у реєстрі боржників, неактивний або припинений статус діяльності, збіг лише за назвою та кількість знайдених ризикових записів. На основі цих даних модуль оцінювання ризику визначає загальний рівень ризику клієнта.

Результати перевірки відображаються у вебінтерфейсі системи. Користувач бачить основні дані клієнта, виявлені ризик-маркери, рівень ризику та може сформувати PDF-звіт. Окремо передбачено збереження історії виконаних перевірок, що дозволяє переглядати попередні результати та контролювати роботу із запитами.

Отже, загальна архітектура інформаційної системи побудована таким чином, щоб забезпечити повний цикл роботи з відкритими даними: від їх завантаження та нормалізації до пошуку клієнта, оцінювання ризику і формування звіту. Така структура є зручною для практичного використання та може бути розширена шляхом додавання нових джерел даних або додаткових ризик-маркерів.

2.2. Проєктування процесу інтеграції відкритих даних засобами IDEF0

Для опису процесу інтеграції відкритих даних у роботі використано

методологію функціонального моделювання IDEF0 [20]. Вона дозволяє подати роботу інформаційної системи у вигляді функціональних блоків, для яких визначаються вхідні дані, керуючі умови, механізми виконання та вихідні результати. Такий підхід є зручним для проєктування, оскільки дає змогу наочно показати логіку роботи системи від моменту надходження даних до формування результату перевірки клієнта.

На контекстній діаграмі основним процесом визначено інтеграцію відкритих даних та оцінювання ризику клієнта фінансової установи. Вхідними даними для цього процесу є відкриті реєстрові дані та код ЄДРПОУ клієнта, за яким виконується перевірка [7; 8]. До керуючих умов належать правила обробки даних, вимоги до перевірки клієнта та модель оцінювання ризику. Механізмами виконання є розроблена вебсистема, база даних SQLite та модуль завантаження й обробки даних. Вихідним результатом є ризиковий профіль клієнта, який містить основні відомості про клієнта, виявлені ризик-маркери, рівень ризику та можливість формування PDF-звіту.

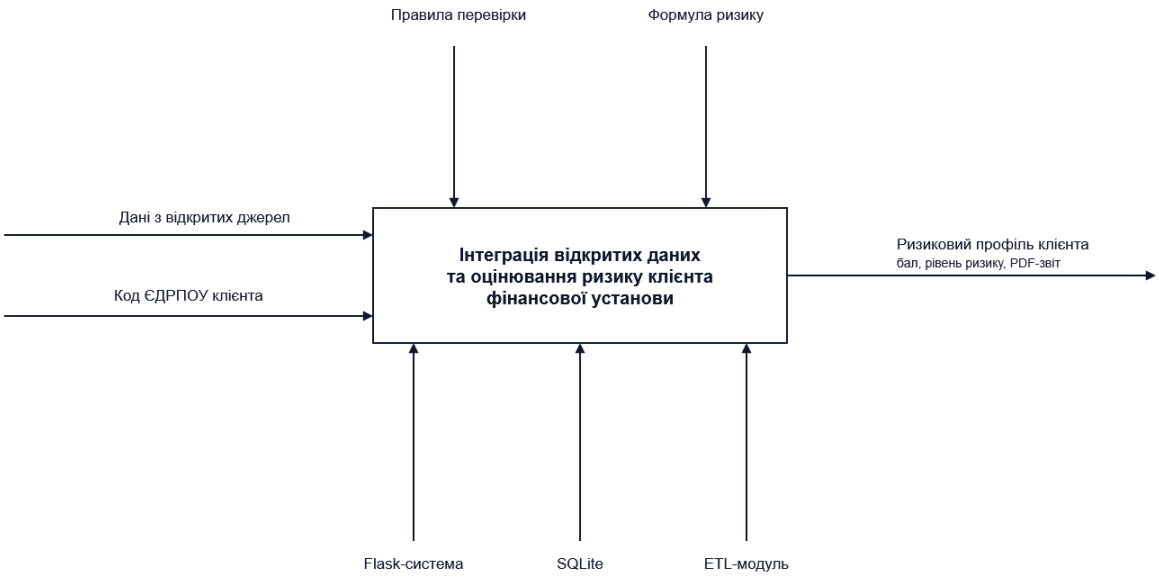


Рис. 2.1. Контекстна діаграма процесу інтеграції відкритих даних та оцінювання ризику клієнта

Контекстна діаграма дозволяє розглядати систему як єдиний процес, що

отримує дані з відкритих джерел і перетворює їх на структурований результат перевірки. Вона визначає межі системи, основні потоки даних, механізми обробки та кінцевий результат, який отримує користувач.

Для детальнішого опису роботи системи виконано декомпозицію основного процесу. У межах декомпозиції виділено кілька послідовних функціональних етапів: завантаження відкритих даних, очищення та нормалізація даних, формування єдиної клієнтської бази, перевірка клієнта та виявлення ризик-маркерів, оцінювання ризику клієнта і формування звіту.

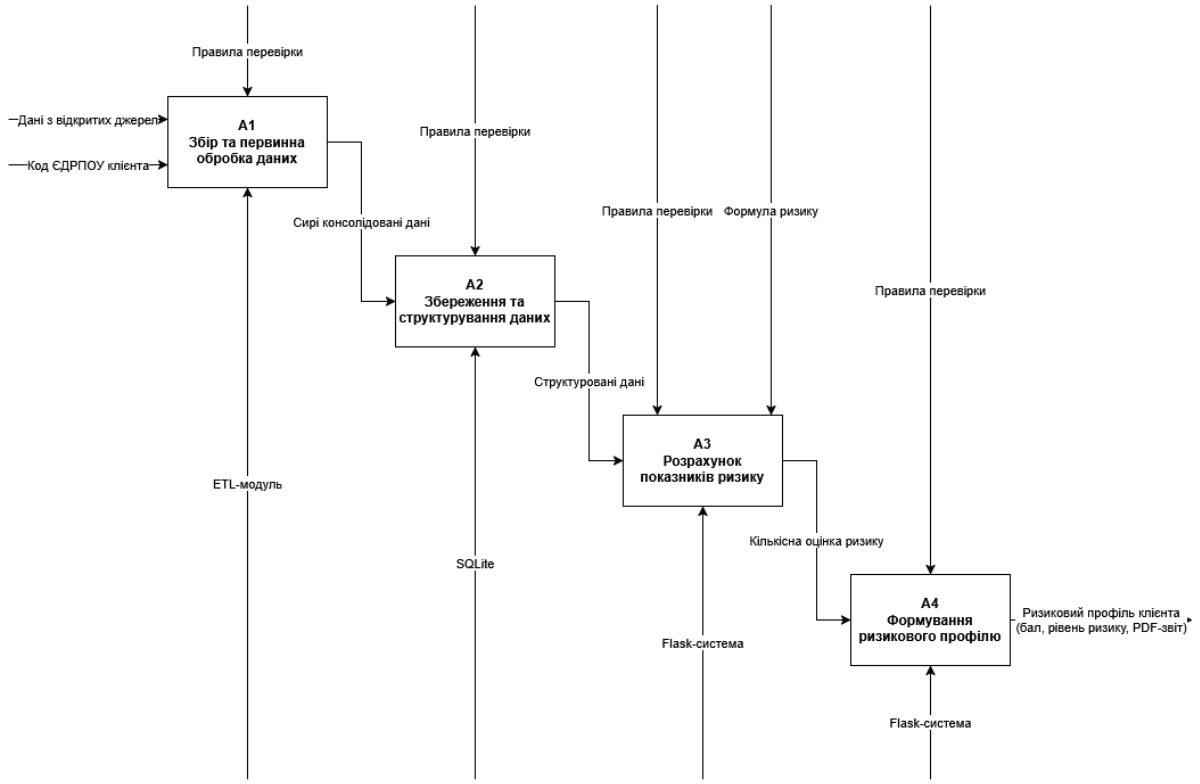


Рис. 2.2. Декомпозиція процесу інтеграції відкритих даних у єдину клієнтську базу

На першому етапі виконується завантаження відкритих даних із підготовлених XML/CSV-масивів. Ці дані містять інформацію про клієнтів, їхній статус, а також записи, які можуть свідчити про наявність ризикових ознак. На другому етапі дані проходять очищення та нормалізацію: приводяться до єдиного формату, перевіряються ключові поля, усуваються зайві або некоректні значення. Це необхідно

для подальшого коректного зіставлення записів.

На третьому етапі нормалізовані дані зберігаються у єдиній клієнтській базі. Така база містить основні відомості про клієнтів, ризик-маркери, користувачів системи та журнал виконаних перевірок. Після цього система може виконувати пошук клієнта за кодом ЄДРПОУ та отримувати пов’язані з ним записи.

Четвертий етап передбачає перевірку клієнта та виявлення ризик-маркерів. Система аналізує статус клієнта, наявність записів у реєстрі боржників, тип збігу даних та кількість знайдених ризикових записів. На основі цих даних формується перелік ризик-маркерів, які впливають на подальше оцінювання.

На завершальному етапі виконується оцінювання ризику клієнта та формування результату перевірки. Користувач отримує узагальнений ризиковий профіль клієнта у вебінтерфейсі системи. За потреби результат може бути збережений у вигляді PDF-звіту, а сам факт перевірки фіксується в журналі системи.

Отже, використання IDEF0 дозволило формалізувати процес інтеграції відкритих даних і показати послідовність основних дій системи. Побудована модель відображає логіку роботи інформаційної системи та є основою для подальшого проєктування бази даних, функціональних модулів і програмної реалізації.

2.3. Проєктування структури бази даних системи

Для збереження та обробки інформації в розробленій системі спроєктовано реляційну базу даних [12; 13; 40]. Вона є центральним елементом інформаційної системи, оскільки забезпечує зберігання клієнтських записів, ризик-маркерів, даних користувачів і журналу виконаних перевірок. Використання єдиної бази даних дозволяє впорядкувати інформацію з різних джерел та забезпечити швидкий доступ до результатів перевірки клієнтів.

У структурі бази даних передбачено чотири основні таблиці: users, registry_edr, risk_markers та search_logs. Таблиця users використовується для збереження даних про користувачів системи. Вона містить інформацію про логін, пароль у хешованому

вигляді та роль користувача. Наявність ролей дає змогу розмежувати права доступу між звичайним користувачем і адміністратором.

Таблиця registry_edr призначена для збереження основних відомостей про клієнтів, отриманих із реєстрових даних [7; 8]. До таких відомостей належать код ЄДРПОУ, назва або ПІБ клієнта, статус діяльності, адреса, керівник та дата оновлення запису. Саме ця таблиця є основою для пошуку клієнта та формування його інформаційного профілю.

Таблиця risk_markers містить записи про виявлені ризикові ознаки клієнтів [9]. Вона пов’язана з таблицею registry_edr через ідентифікатор клієнта. У цій таблиці зберігаються тип ризику, опис ризик-маркера, джерело інформації та дата виявлення. Така структура дозволяє зберігати декілька ризикових записів для одного клієнта та враховувати їх під час оцінювання ризику.

Таблиця search_logs використовується для збереження історії виконаних перевірок. У ній фіксується користувач, який виконав пошук, пошуковий запит, дата і час перевірки та результат. Це дає змогу переглядати попередні перевірки, контролювати роботу користувачів і повторно звертатися до результатів аналізу.

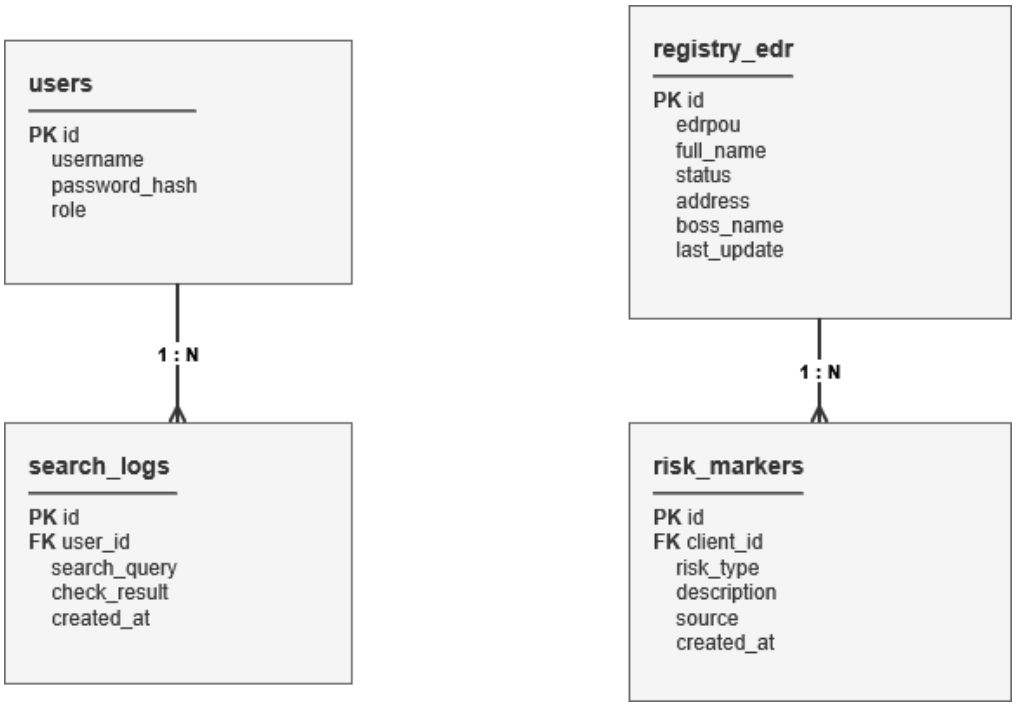


Рис. 2.3. Структура бази даних інформаційної системи

Зв’язки між таблицями побудовані таким чином, щоб забезпечити логічну цілісність даних. Один клієнт із таблиці registry_edr може мати кілька пов’язаних ризик-маркерів у таблиці risk_markers. Один користувач із таблиці users може виконувати багато перевірок, які зберігаються в таблиці search_logs. Така структура є достатньою для реалізації основних функцій системи та водночас залишається простою для подальшого розширення.

Проєктування бази даних виконувалося з урахуванням потреб системи: швидкого пошуку клієнта за кодом ЄДРПОУ, збереження результатів перевірки, відображення ризик-маркерів і формування історії запитів. У разі подальшого розвитку системи структура бази може бути доповнена новими таблицями для інших відкритих джерел даних, додаткових типів ризиків або розширених звітів.

Отже, спроєктована база даних забезпечує збереження основних інформаційних об’єктів системи та підтримує її ключові функції. Вона дозволяє об’єднати клієнтські дані, ризик-маркери, користувачів і журнал перевірок у єдину структуру, що є необхідним для стабільної роботи інформаційної системи.

2.4. Модель оцінювання ризику клієнта та опис ризик-маркерів

Для автоматизованого визначення рівня ризику клієнта в інформаційній системі розроблено модель оцінювання ризику [3; 4; 19]. Її призначення полягає в тому, щоб на основі відкритих даних і виявлених ризикових ознак сформувати узагальнений ризиковий профіль клієнта. Такий підхід дозволяє не лише відобразити знайдені записи, а й надати користувачу зрозумілий результат перевірки.

Модель оцінювання ризику побудована на використанні ризик-маркерів. Під ризик-маркером у роботі розуміється виявлена ознака, яка може впливати на рівень ризику клієнта фінансової установи. До таких ознак належать наявність записів у реєстрі боржників, неактивний або припинений статус клієнта, збіг даних лише за назвою, а також кількість знайдених ризикових записів.

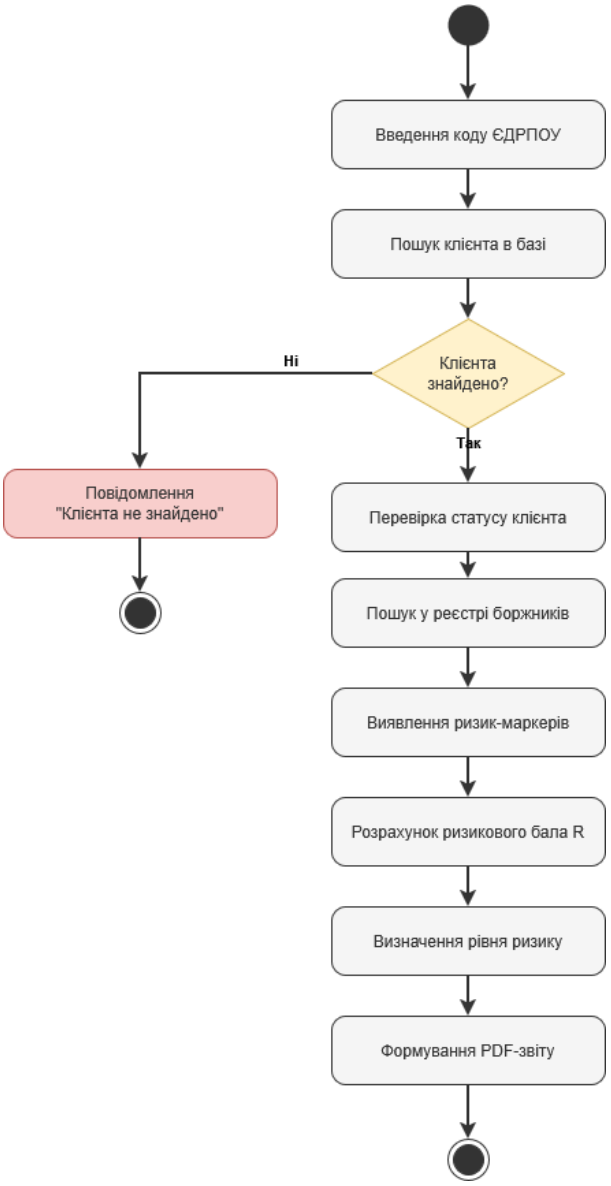


Рис. 2.4. Модель оцінювання ризику клієнта в інформаційній системі

У межах розробленої системи ризик клієнта визначається за 100-бальною шкалою. Кожен ризик-маркер має власне значення, яке відображає його вплив на загальний рівень ризику. Найбільшу вагу має наявність запису в реєстрі боржників, оскільки така ознака безпосередньо пов’язана з можливими фінансовими зобов’язаннями клієнта [9]. Додатково враховуються статус діяльності клієнта, тип збігу даних та кількість виявлених ризикових записів.

Таблиця 2.1 - Ризик-маркери, що використовуються в моделі оцінювання ризику клієнта

Позначення	**Ризик-маркер**	**Значення**	**Обґрунтування**
D	Наявність запису в реєстрі боржників	50	Свідчить про можливі фінансові зобов’язання клієнта
S	Неактивний або припинений статус клієнта	25	Може вказувати на обмеження або припинення діяльності суб’єкта
M	Збіг лише за назвою	15	Потребує додаткової перевірки, оскільки збіг не є точним за кодом ЄДРПОУ
N	Кількість виявлених ризик-маркерів	до 30	Посилює загальний рівень ризику залежно від кількості знайдених записів

Загальний ризиковий бал клієнта визначається за формулою:

$$R = \min(100, D + S + M + N),$$

де R — інтегральний ризиковий бал клієнта; D — показник наявності запису в реєстрі боржників; S — показник статусу діяльності клієнта; M — показник типу збігу даних; N — показник кількості виявлених ризик-маркерів.

Використання функції min(100, ...) необхідне для того, щоб значення ризикового бала не перевищувало 100 балів. Це дозволяє зберігати результат у межах єдиної шкали та коректно відображати рівень ризику в інтерфейсі системи.

Для інтерпретації отриманого бала в системі використано три рівні ризику. Якщо значення ризикового бала становить від 0 до 30 балів, клієнт належить до низького рівня ризику. Якщо значення перебуває в межах від 31 до 70 балів, клієнт належить до середнього рівня ризику. Якщо ризиковий бал становить від 71 до 100 балів, клієнт належить до високого рівня ризику.

Таблиця 2.2 - Шкала інтерпретації ризикового бала клієнта

Ризиковий бал	**Рівень ризику**	**Характеристика**
0–30	Низький	Критичних ризикових ознак не виявлено або їх вплив є незначним
31–70	Середній	Виявлено ризикові ознаки, які потребують додаткової уваги
71–100	Високий	Виявлено суттєві ризикові ознаки, що можуть впливати на рішення щодо співпраці

Розроблена модель не замінює остаточного рішення фахівця, а використовується як інструмент підтримки прийняття рішень. Її завдання полягає в тому, щоб швидко узагальнити знайдену інформацію, показати основні ризикові ознаки та допомогти користувачу зорієнтуватися в результатах перевірки.

Отже, модель оцінювання ризику клієнта забезпечує формалізований підхід до аналізу відкритих даних. Вона дозволяє врахувати кілька груп ризик-маркерів, сформувати інтегральний ризиковий бал і визначити рівень ризику клієнта у зрозумілому для користувача вигляді.

Висновки до розділу 2

У другому розділі було виконано проєктування інформаційної системи інтеграції відкритих даних у єдину клієнтську базу фінансових установ. Визначено загальну архітектуру системи, яка включає модуль авторизації, модуль завантаження та обробки даних, базу даних, модуль пошуку клієнта, модуль виявлення ризик-маркерів, модель оцінювання ризику, вебінтерфейс користувача та механізм формування звітів.

За допомогою методології IDEF0 описано процес інтеграції відкритих даних і послідовність їх обробки. Контекстна діаграма показує систему як єдиний процес, що отримує відкриті реєстрові дані та код ЄДРПОУ клієнта, а результатом формує ризиковий профіль клієнта. Декомпозиція процесу деталізує основні етапи роботи системи: завантаження даних, очищення й нормалізацію, формування клієнтської

бази, перевірку клієнта, виявлення ризик-маркерів та формування звіту.

Спроєктовано структуру бази даних інформаційної системи. Вона містить таблиці для збереження користувачів, клієнтських записів, ризик-маркерів і журналу виконаних перевірок. Така структура забезпечує збереження основних даних, зв’язок клієнтів із виявленими ризиковими ознаками та можливість перегляду історії запитів.

Також розроблено модель оцінювання ризику клієнта, яка враховує наявність записів у реєстрі боржників, статус діяльності клієнта, тип збігу даних і кількість виявлених ризик-маркерів. На основі цих показників система формує ризиковий бал і визначає рівень ризику клієнта. Запропонована модель дозволяє подати результати перевірки у зрозумілому вигляді та може використовуватися як інструмент підтримки прийняття рішень.

РОЗДІЛ 3. ПРОГРАМНА РЕАЛІЗАЦІЯ ТА ТЕСТУВАННЯ ІНФОРМАЦІЙНОЇ СИСТЕМИ

3.1. Засоби програмної реалізації системи

Програмну реалізацію інформаційної системи виконано у вигляді вебзастосунку, який забезпечує роботу з клієнтськими даними, пошук клієнта, виявлення ризик-маркерів, оцінювання ризику та формування звіту. Для розробки системи було використано мову програмування Python, оскільки вона має зручні засоби для роботи з файлами, базами даних, вебмаршрутами та обробкою інформації [10].

Основою вебзастосунку є фреймворк Flask [11; 38]. Він використовується для створення маршрутів системи, обробки запитів користувача, авторизації, відображення сторінок і передавання даних між серверною частиною та інтерфейсом. Flask є доцільним вибором для цієї роботи, оскільки дозволяє швидко реалізувати вебсистему з необхідною логікою та не потребує надмірно складної структури проєкту.

Для збереження інформації використано базу даних SQLite [12; 13; 15]. Вона забезпечує зберігання даних про користувачів, клієнтів, ризик-маркери та історію виконаних перевірок. Використання SQLite є зручним для кваліфікаційної роботи, оскільки база даних зберігається локально, не потребує окремого серверного налаштування та дозволяє реалізувати всі необхідні функції системи.

Для завантаження та обробки вхідних даних використано XML- та CSV-файли [22; 23]. XML-масив застосовується для зберігання основних відомостей про клієнтів, зокрема коду ЄДРПОУ, назви, статусу діяльності, адреси та керівника. CSV-масив використовується для зберігання записів, пов’язаних із ризиковими ознаками, зокрема даних про боржників і виконавчі провадження. Під час завантаження ці дані очищуються, нормалізуються та вносяться до бази даних.

Користувацький інтерфейс системи реалізовано за допомогою HTML, CSS, Bootstrap та шаблонізатора Jinja2 [14; 24; 28; 29]. Це дозволило створити сторінки для

авторизації, пошуку клієнта, перегляду результатів перевірки, історії запитів та адміністративної панелі. Для зручності користувача результати перевірки подаються у структурованому вигляді: окремо відображаються основні дані клієнта, виявлені ризик-маркери, рівень ризику та можливість формування PDF-звіту.

Для формування звітів у системі використано бібліотеку FPDF [26]. Вона дозволяє створювати PDF-документи з результатами перевірки клієнта. У звіті відображаються основні відомості про клієнта, результат оцінювання ризику, знайдені ризик-маркери та загальний висновок системи.

Загальна структура програмної реалізації складається з кількох основних файлів. Файл app.py відповідає за маршрути вебзастосунку, авторизацію користувачів, пошук клієнта, відображення результатів і формування PDF-звіту. Файл backend.py містить логіку роботи з базою даних, профілем клієнта, ризик-маркерами та моделлю оцінювання ризику. Файл data_loader.py використовується для завантаження та обробки XML/CSV-масивів. Файл generate_data.py призначений для створення демонстраційних даних, які використовуються під час тестування системи.

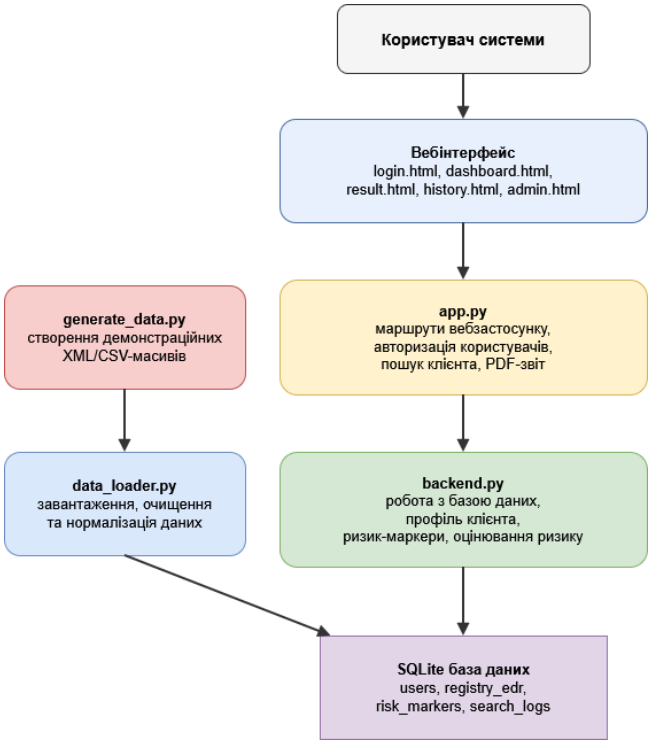


Рис. 3.1. Структура програмних компонентів інформаційної системи

Таким чином, обрані засоби програмної реалізації дозволили створити вебсистему, яка поєднує роботу з базою даних, обробку відкритих даних, оцінювання ризику клієнта та формування звітів. Використання Python, Flask, SQLite, HTML/CSS, Bootstrap і FPDF забезпечило достатню функціональність системи та можливість її подальшого розширення.

3.2. Опис функціональних модулів розробленої системи

Розроблена інформаційна система складається з кількох функціональних модулів, кожен із яких виконує окрему частину загального процесу перевірки клієнта. Такий поділ спрощує структуру системи, робить її зрозумілою для супроводу та дозволяє в майбутньому розширювати функціональність без повної зміни програмної логіки.

Першим функціональним модулем є модуль авторизації користувачів [30; 31]. Він забезпечує вхід до системи за логіном і паролем, перевірку облікових даних та розмежування доступу залежно від ролі користувача. У системі передбачено дві основні ролі: користувач і адміністратор. Користувач має доступ до пошуку клієнта, перегляду результатів перевірки та історії запитів. Адміністратор додатково може оновлювати базу даних і створювати нових користувачів.

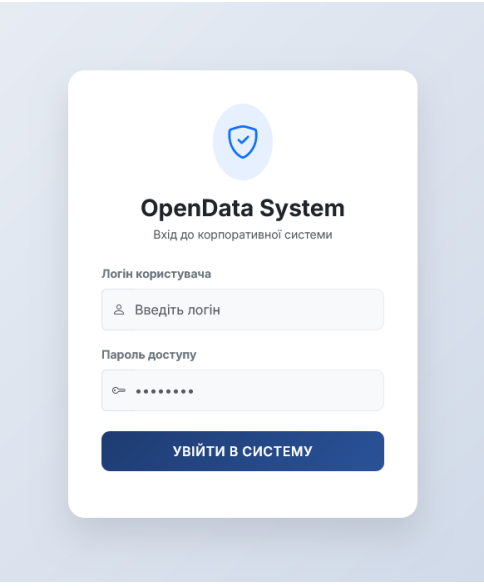


Рис. 3.2. Сторінка авторизації користувача в інформаційній системі

Наступним модулем є модуль завантаження та обробки даних. Він відповідає за зчитування XML/CSV-масивів, очищення даних, приведення їх до єдиної структури та внесення до бази даних. XML-файл використовується для збереження основних відомостей про клієнтів, а CSV-файл — для записів, пов’язаних із ризиковими ознаками. Під час обробки даних система зіставляє записи за кодом ЄДРПОУ або назвою клієнта.

Окреме місце займає модуль пошуку клієнта. Його основна функція полягає в пошуку клієнта в єдиній базі даних за кодом ЄДРПОУ. Після введення коду система перевіряє наявність відповідного запису в базі, отримує основні відомості про клієнта та передає їх для подальшого аналізу. Якщо клієнта не знайдено, користувач отримує відповідне повідомлення.

Модуль виявлення ризик-маркерів аналізує інформацію, пов’язану з клієнтом, і визначає наявність ризикових ознак. До таких ознак належать записи у реєстрі боржників, неактивний або припинений статус клієнта, збіг лише за назвою та кількість знайдених ризикових записів. Виявлені ризик-маркери зберігаються у базі даних і відображаються користувачу на сторінці результату перевірки.

Модуль оцінювання ризику клієнта використовує виявлені ризик-маркери для формування узагальненого ризикового бала. На основі цього бала система визначає рівень ризику клієнта: низький, середній або високий. Такий підхід дозволяє подати результат перевірки не лише у вигляді окремих записів, а як зрозумілий ризиковий профіль.

Важливим компонентом системи є модуль формування звітів. Він дозволяє створити PDF-звіт за результатами перевірки клієнта. У звіті відображаються основні дані клієнта, рівень ризику, виявлені ризик-маркери та загальний висновок системи. Це дає змогу зберігати результат перевірки в окремому документі та використовувати його для подальшого аналізу.

Також у системі реалізовано модуль журналу перевірок. Він зберігає інформацію про виконані пошукові запити, дату й час перевірки, користувача та

результат. Наявність такого журналу дозволяє переглядати історію роботи із системою, повторно відкривати результати перевірок і контролювати виконані дії.

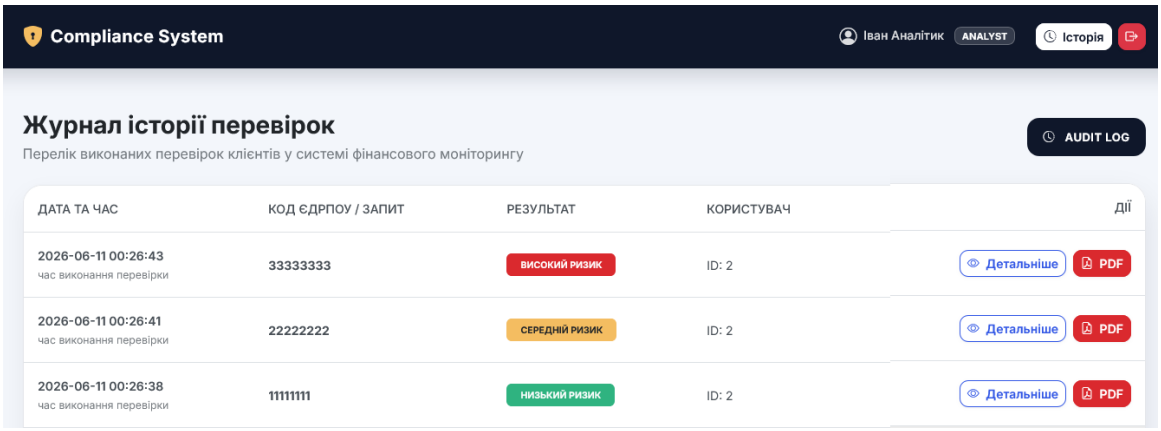


Рис. 3.3. Журнал виконаних перевірок клієнтів

Отже, функціональні модулі розробленої системи забезпечують повний цикл роботи з клієнтськими даними: від авторизації користувача та завантаження відкритих даних до пошуку клієнта, виявлення ризик-маркерів, оцінювання ризику, формування PDF-звіту та збереження історії перевірок.

3.3. Інтерфейс користувача та порядок роботи із системою

Інтерфейс розробленої інформаційної системи побудований таким чином, щоб користувач міг швидко виконати перевірку клієнта та отримати зрозумілий результат. Основна увага приділена простоті пошуку, наочному відображенню даних клієнта, ризик-маркерів, доступу до історії перевірок і можливості формування PDF-звіту.

Після авторизації користувач потрапляє на головну сторінку системи. На ній розміщено форму пошуку клієнта за кодом ЄДРПОУ. Такий спосіб пошуку є основним, оскільки код ЄДРПОУ дозволяє однозначно ідентифікувати клієнта в базі даних. Користувач вводить код у відповідне поле та запускає перевірку.

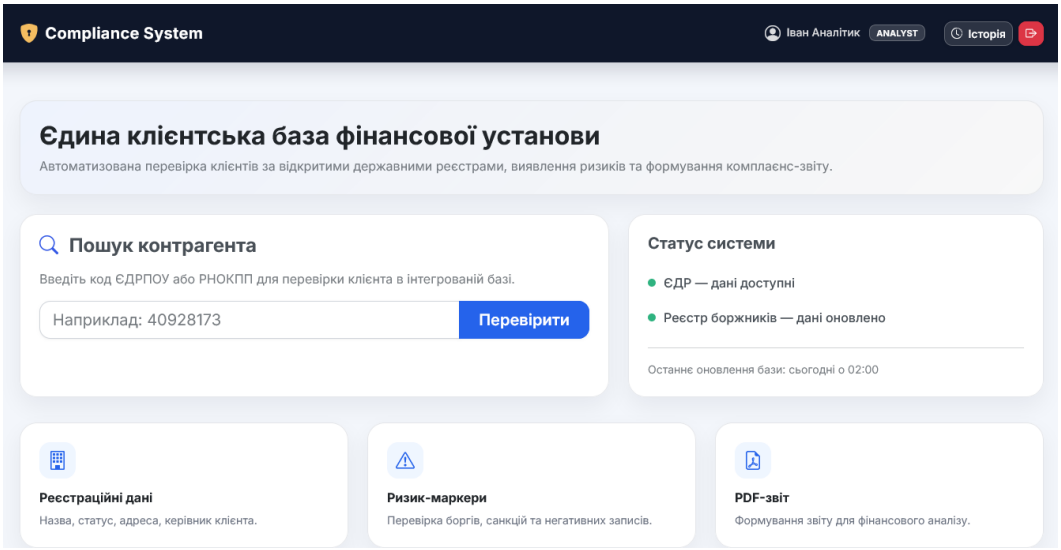


Рис. 3.4. Головна сторінка пошуку клієнта за кодом ЄДРПОУ

Після виконання пошуку система формує сторінку результату перевірки. У верхній частині сторінки відображається загальний результат аналізу: ризиковий бал, рівень ризику та короткий опис моделі оцінювання. Це дає змогу користувачу швидко оцінити загальний стан клієнта без ручного перегляду всіх знайдених записів.

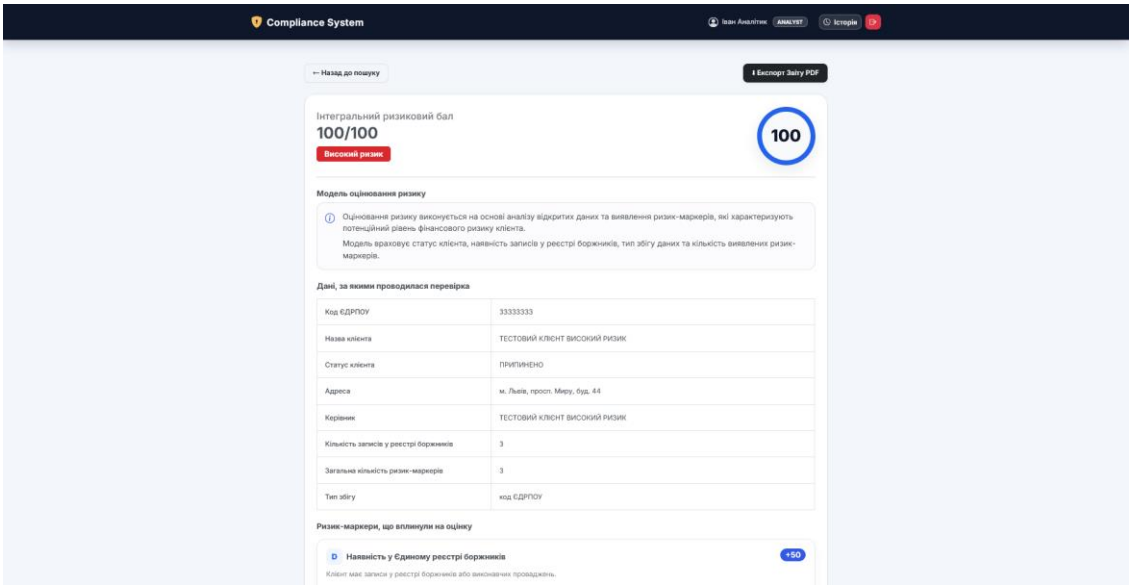


Рис. 3.5. Загальний результат перевірки клієнта

Окремо на сторінці результату подається досьє клієнта. У ньому відображаються основні реєстраційні дані: код ЄДРПОУ, назва клієнта, статус діяльності, адреса та керівник. Такий блок потрібний для того, щоб користувач міг одразу перевірити, до якого саме суб’єкта належать результати аналізу.

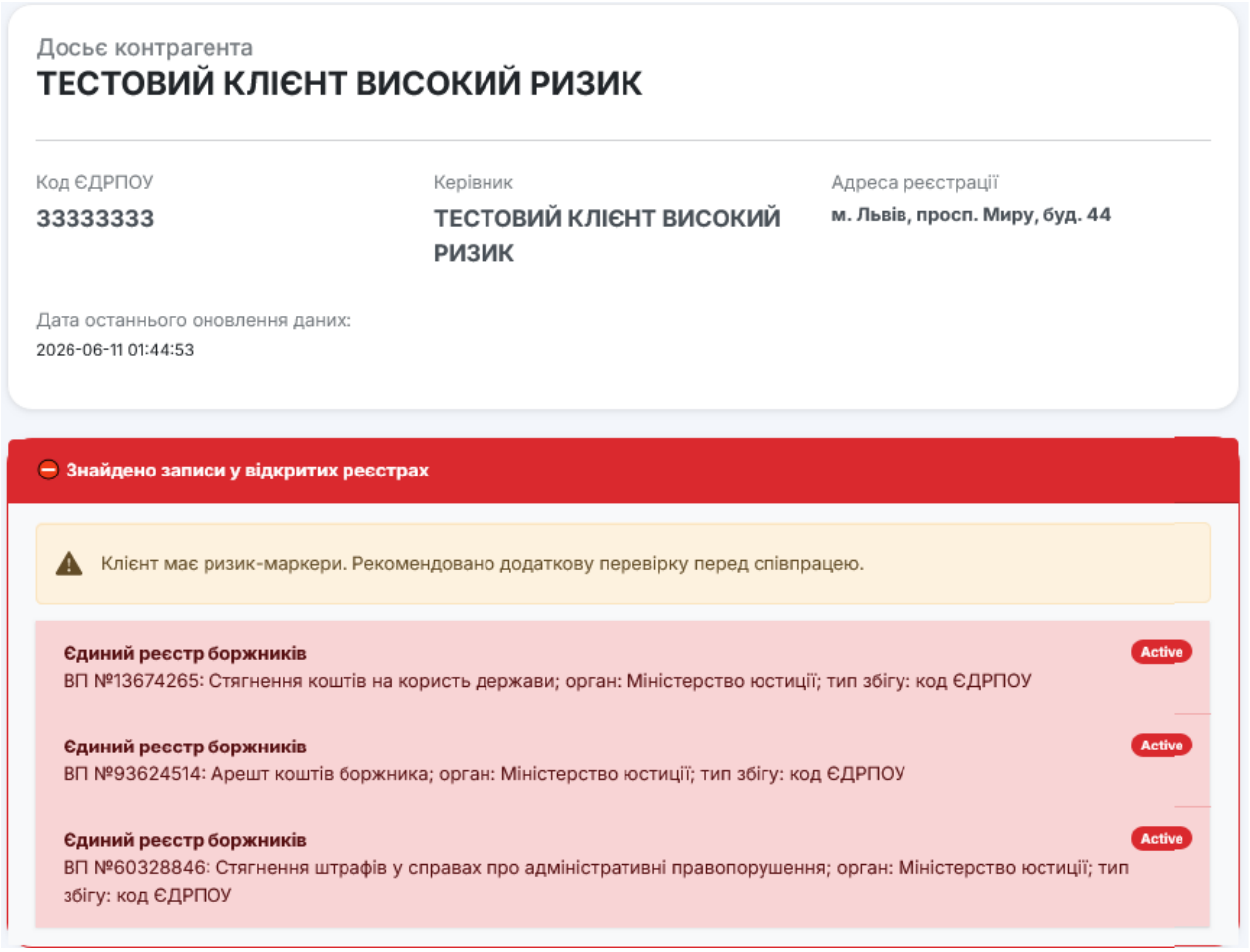


Рис. 3.6. Досьє клієнта в інформаційній системі

Важливою частиною сторінки результату є блок ризик-маркерів. У ньому система показує виявлені ризикові ознаки, джерело їх походження, умову спрацювання та значення, яке було враховано під час перевірки. Такий підхід робить результат прозорим для користувача, оскільки він бачить не лише підсумковий рівень ризику, а й причини його формування.

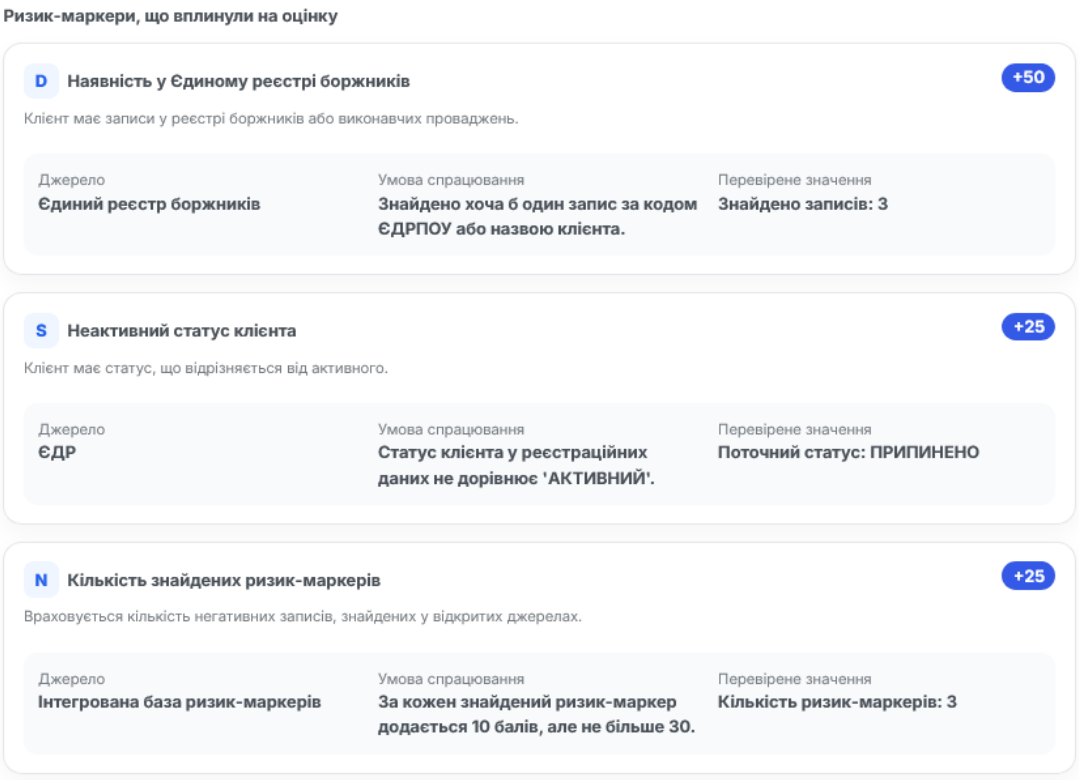
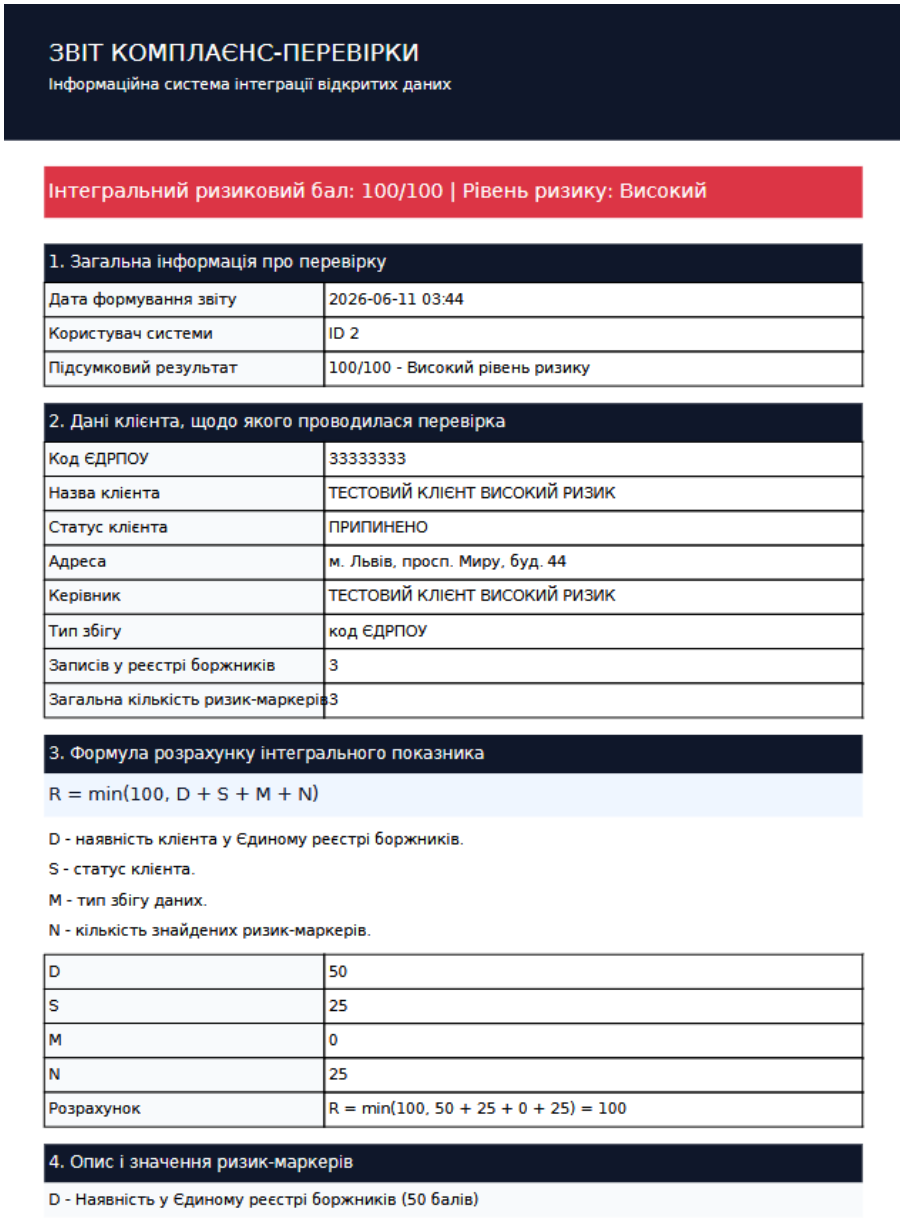


Рис. 3.7. Відображення ризик-маркерів клієнта

Для збереження результату перевірки у системі передбачено формування PDF-звіту. У звіті подаються основні дані клієнта, результат оцінювання ризику, перелік виявлених ризик-маркерів і загальний висновок системи. Це дозволяє використовувати результат перевірки як окремий документ для подальшого аналізу або внутрішнього зберігання.



ЗВІТ КОМПЛАЄНС-ПЕРЕВІРКИ

Інформаційна система інтеграції відкритих даних

Інтегральний ризиковий бал: 100/100 | Рівень ризику: Високий

1. Загальна інформація про перевірку	
Дата формування звіту	2026-06-11 03:44
Користувач системи	ID 2
Підсумковий результат	100/100 - Високий рівень ризику

2. Дані клієнта, щодо якого проводилася перевірка	
Код ЄДРПОУ	33333333
Назва клієнта	ТЕСТОВИЙ КЛІЄНТ ВИСОКИЙ РИЗИК
Статус клієнта	ПРИПИНЕНО
Адреса	м. Львів, просп. Миру, буд. 44
Керівник	ТЕСТОВИЙ КЛІЄНТ ВИСОКИЙ РИЗИК
Тип збігу	код ЄДРПОУ
Записів у реєстрі боржників	3
Загальна кількість ризик-маркерів	3

3. Формула розрахунку інтегрального показника

R = min(100, D + S + M + N)

D - наявність клієнта у Єдиному реєстрі боржників.

S - статус клієнта.

M - тип збігу даних.

N - кількість знайдених ризик-маркерів.

D	50
S	25
M	0
N	25
Розрахунок	R = min(100, 50 + 25 + 0 + 25) = 100

4. Опис і значення ризик-маркерів

D - Наявність у Єдиному реєстрі боржників (50 балів)

Рис. 3.8. Фрагмент PDF-звіту за результатами перевірки клієнта

Окремо в системі реалізовано сторінку історії перевірок. Вона містить журнал виконаних запитів із зазначенням дати й часу перевірки, коду ЄДРПОУ, результату та користувача, який виконав перевірку. Завдяки цьому користувач може переглядати попередні результати та повторно відкривати сформовані звіти.

Адміністративна панель призначена для керування даними системи. Через неї адміністратор може виконувати оновлення бази даних, запускати завантаження демонстраційних XML/CSV-масивів, створювати нових користувачів і переглядати

загальну статистику роботи системи. Це забезпечує зручне обслуговування системи без необхідності безпосередньо працювати з файлами бази даних.

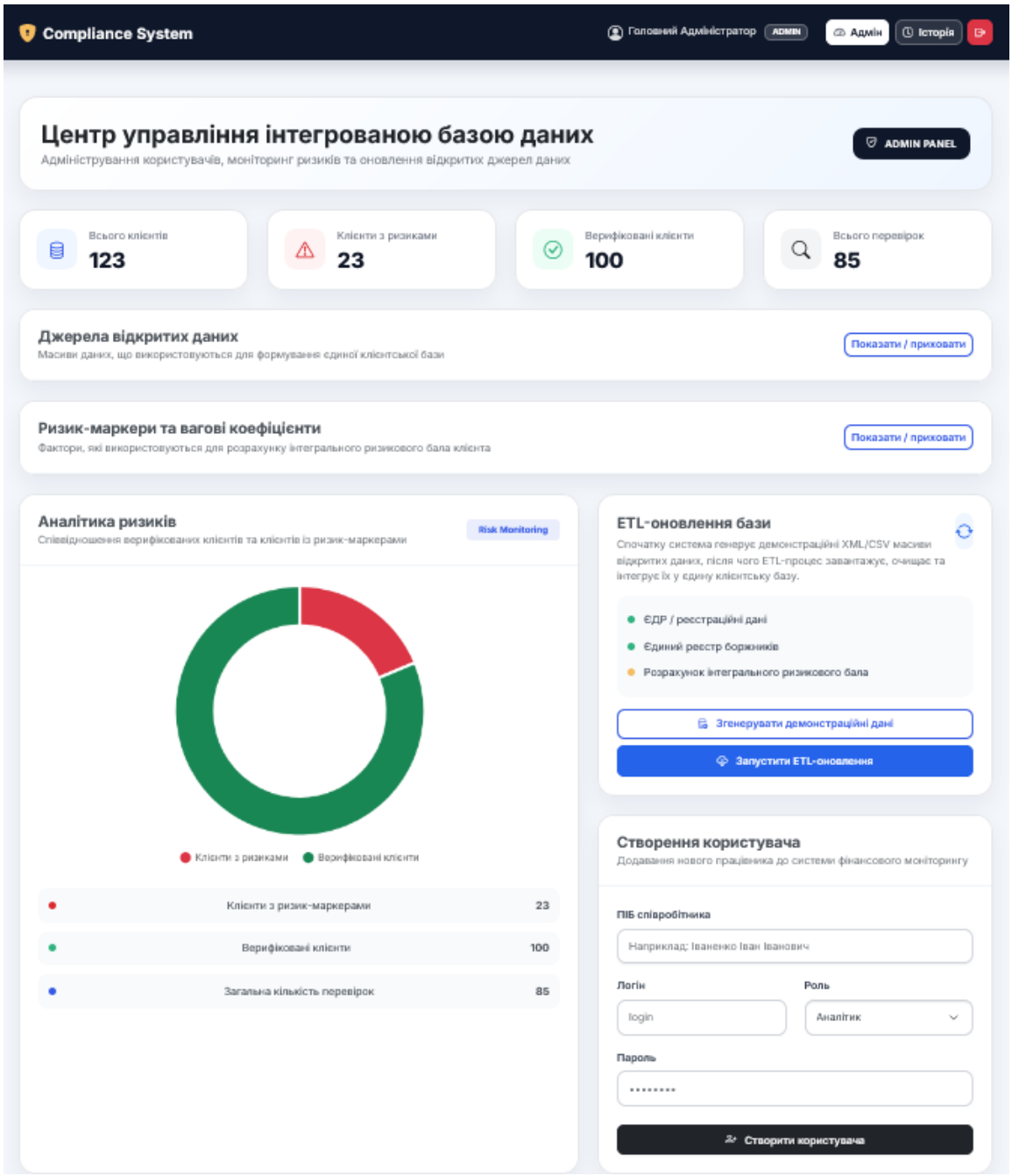


Рис. 3.9. Адміністративна панель інформаційної системи

Порядок роботи користувача із системою є послідовним. Спочатку користувач проходить авторизацію, після чого вводить код ЄДРПОУ клієнта на головній сторінці. Далі система виконує пошук у базі даних, отримує дані клієнта, перевіряє наявність ризик-маркерів, визначає рівень ризику та відображає результат. За потреби

користувач може сформувати PDF-звіт або переглянути попередні перевірки в журналі.

Отже, інтерфейс системи забезпечує зручну роботу з основними функціями: пошуком клієнта, переглядом результатів перевірки, аналізом досьє клієнта, відображенням ризик-маркерів, формуванням PDF-звіту, переглядом історії та адмініструванням даних.

3.4. Тестовий приклад перевірки клієнта та аналіз результатів

Для перевірки працездатності інформаційної системи було використано демонстраційні дані, сформовані у вигляді XML/CSV-масивів. Вони містять записи про клієнтів, їхній статус, а також ризикові ознаки, пов’язані з наявністю боргових записів. Такий підхід дав змогу перевірити основні функції системи без використання реальних персональних або конфіденційних даних [2].

У межах тестування було виконано перевірку клієнта за кодом ЄДРПОУ 33333333. Цей запис використано як приклад клієнта з високим рівнем ризику, оскільки для нього в базі даних наявні ризик-маркери, які впливають на підсумковий результат перевірки.

Таблиця 3.1 - Тестові дані для перевірки клієнта

Показник	**Значення**
Код ЄДРПОУ	33333333
Назва клієнта	Тестовий клієнт високий ризик
Статус клієнта	Припинено
Джерело основних даних	XML-масив реєстрових даних
Джерело ризикових записів	CSV-масив записів реєстру боржників

Після введення коду ЄДРПОУ система виконала пошук клієнта в єдиній базі даних, отримала основні відомості про нього та перевірила наявність пов’язаних ризикових записів. У результаті було сформовано ризиковий профіль клієнта, у якому відображено загальний рівень ризику та перелік виявлених ризик-маркерів.

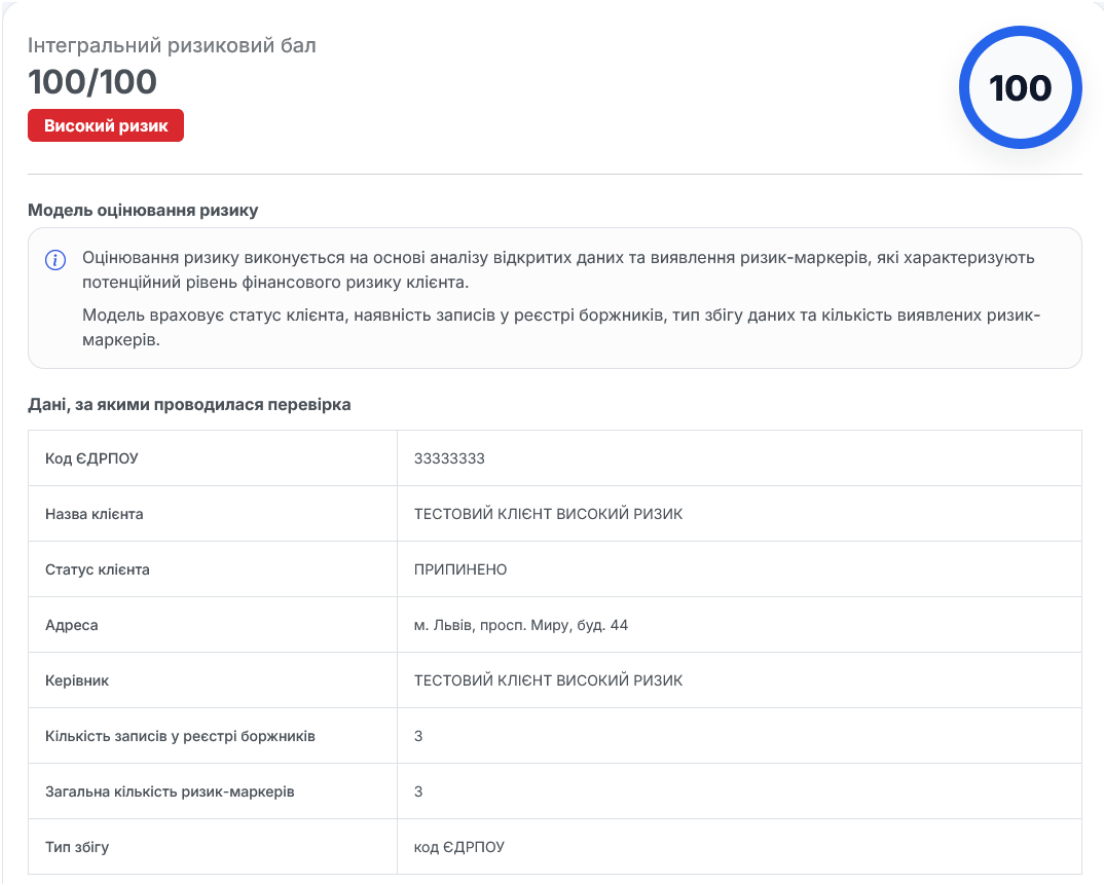


Рис. 3.10. Тестова перевірка клієнта з високим рівнем ризику

Під час перевірки система врахувала кілька ризикових ознак: наявність записів у реєстрі боржників, припинений статус клієнта, а також кількість виявлених ризик-маркерів. Сукупність цих факторів вплинула на формування високого рівня ризику. Такий результат є логічним, оскільки клієнт має не лише неактивний статус, а й додаткові записи, які можуть свідчити про фінансові зобов'язання.

Таблиця 3.2 - Результат тестової перевірки клієнта

Критерій перевірки	**Результат**
Клієнта знайдено в базі	Так
Статус клієнта	Припинено
Записи в реєстрі боржників	Виявлено
Ризик-маркери	Виявлено
Рівень ризику	Високий
PDF-звіт	Сформовано
Запис в історії перевірок	Збережено

Після завершення перевірки результат було збережено в журналі системи. Це підтверджує роботу модуля історії перевірок, який фіксує пошуковий запит, дату й час виконання перевірки, користувача та підсумковий результат. Також було сформовано PDF-звіт, що підтверджує роботу модуля звітності.

Для додаткової перевірки роботи системи можуть використовуватися клієнти з різними рівнями ризику. Наприклад, клієнт без ризикових записів дає змогу перевірити сценарій низького ризику, а клієнт з окремими ризиковими ознаками - сценарій середнього ризику. Це дозволяє переконатися, що система коректно реагує на різні комбінації вхідних даних.

Отже, тестування показало, що розроблена інформаційна система коректно виконує основні функції: здійснює пошук клієнта за кодом ЄДРПОУ, отримує дані з бази, виявляє ризик-маркери, визначає рівень ризику, формує PDF-звіт і зберігає результат перевірки в журналі. Отримані результати підтверджують працездатність системи та відповідність її основним функціональним вимогам.

Висновки до розділу 3

У третьому розділі було описано програмну реалізацію інформаційної системи інтеграції відкритих даних у єдину клієнтську базу фінансових установ. Для створення системи використано мову програмування Python, фреймворк Flask, базу даних SQLite, HTML/CSS, Bootstrap, шаблонізатор Jinja2 та бібліотеку FPDF для формування звітів.

Розглянуто основні функціональні модулі системи: авторизацію користувачів, завантаження та обробку XML/CSV-масивів, пошук клієнта за кодом ЄДРПОУ, виявлення ризик-маркерів, оцінювання ризику, формування PDF-звіту та збереження історії виконаних перевірок. Така структура забезпечує повний цикл роботи системи — від підготовки даних до отримання користувачем результату перевірки.

Описано інтерфейс користувача та порядок роботи із системою. Користувач може авторизуватися, виконати пошук клієнта, переглянути досьє, оцінку ризику,

перелік виявлених ризик-маркерів, сформувати PDF-звіт і звернутися до журналу попередніх перевірок. Для адміністратора передбачено окрему панель керування, яка дозволяє оновлювати дані, створювати користувачів і переглядати загальну статистику.

Проведено тестову перевірку системи на демонстраційних даних. У результаті тестування підтверджено, що система коректно знаходить клієнта в базі даних, визначає ризикові ознаки, формує рівень ризику, створює PDF-звіт і зберігає результат перевірки в журналі. Отримані результати свідчать про працездатність розробленої інформаційної системи та її відповідність визначеним функціональним вимогам.

ВИСНОВКИ

У кваліфікаційній роботі було розглянуто питання проєктування та розробки інформаційної системи інтеграції відкритих даних у єдину клієнтську базу фінансових установ. У результаті виконання роботи було досягнуто поставленої мети — розроблено систему, яка дозволяє автоматизувати первинну комплаєнс-перевірку клієнта, виявляти ризикові ознаки та формувати узагальнений результат перевірки.

У першому розділі було проаналізовано предметну область використання відкритих даних у діяльності фінансових установ. Визначено, що відкриті реєстрові дані можуть бути важливим джерелом інформації під час перевірки клієнтів, оскільки містять відомості про статус суб'єкта, боргові записи, виконавчі провадження та інші потенційні ризикові ознаки. Також було розглянуто особливості комплаєнс-перевірки клієнтів і сформовано основні вимоги до інформаційної системи.

У другому розділі було виконано проєктування інформаційної системи. Визначено її загальну архітектуру, побудовано IDEF0-модель процесу інтеграції відкритих даних, спроєктовано структуру бази даних та розроблено модель оцінювання ризику клієнта. Запропонована модель враховує наявність записів у реєстрі боржників, статус клієнта, тип збігу даних і кількість виявлених ризик-маркерів. Це дозволяє сформувати ризиковий бал і визначити рівень ризику клієнта.

У третьому розділі було описано програмну реалізацію системи. Розробку виконано з використанням мови програмування Python, фреймворку Flask, бази даних SQLite, HTML/CSS, Bootstrap, шаблонізатора Jinja2 та бібліотеки FPDF. Реалізована система забезпечує авторизацію користувачів, завантаження та обробку XML/CSV-масивів, пошук клієнта за кодом ЄДРПОУ, виявлення ризик-маркерів, оцінювання ризику, формування PDF-звіту та збереження історії перевірок.

Практичне значення отриманих результатів полягає в тому, що розроблена інформаційна система може використовуватися як інструмент для автоматизації первинної перевірки клієнтів фінансових установ. Вона дозволяє зменшити обсяг ручної роботи, структурувати інформацію з відкритих джерел і надати користувачу

зрозумілий ризиковий профіль клієнта.

Тестування системи на демонстраційних даних підтвердило її працездатність. Система коректно виконує пошук клієнта, отримує дані з бази, визначає ризик-маркери, формує рівень ризику, створює PDF-звіт і зберігає результат перевірки в журналі. Отже, розроблена інформаційна система відповідає визначеним функціональним вимогам і може бути розширена шляхом підключення нових джерел відкритих даних та додавання нових ризик-маркерів.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Про доступ до публічної інформації : Закон України від 13.01.2011 № 2939-VI. Верховна Рада України. URL: https://zakon.rada.gov.ua/go/2939-17 (дата звернення: 06.06.2026).

2. Про захист персональних даних : Закон України від 01.06.2010 № 2297-VI. Верховна Рада України. URL: https://zakon.rada.gov.ua/go/2297-17 (дата звернення: 06.06.2026).

3. Про запобігання та протидію легалізації доходів, одержаних злочинним шляхом, фінансуванню тероризму та фінансуванню розповсюдження зброї масового знищення : Закон України від 06.12.2019 № 361-IX. Верховна Рада України. URL: https://zakon.rada.gov.ua/go/361-20 (дата звернення: 06.06.2026).

4. Про затвердження Положення про здійснення банками фінансового моніторингу : постанова Правління Національного банку України від 19.05.2020 № 65. Національний банк України. URL: https://bank.gov.ua/ua/legislation/Resolution_19052020_65 (дата звернення: 06.06.2026).

5. Єдиний державний вебпортал відкритих даних. Data.gov.ua. URL: https://data.gov.ua/ (дата звернення: 06.06.2026).

6. Що таке відкриті дані. Дія.Відкриті дані. URL: https://diia.data.gov.ua/info-center/wodi (дата звернення: 06.06.2026).

7. Єдиний державний реєстр юридичних осіб, фізичних осіб-підприємців та громадських формувань. Міністерство юстиції України. URL: https://usr.minjust.gov.ua/ (дата звернення: 06.06.2026).

8. Єдиний державний реєстр юридичних осіб, фізичних осіб-підприємців та громадських формувань. ДП «Національні інформаційні системи». URL: https://nais.gov.ua/p/ediniy-derjavniy-reestr-yuridichnih-osib-fizichnih-osib-pidpriemtsiv-ta-gromadskih-formuvan (дата звернення: 06.06.2026).

9. Боржник чи ні? Як перевірити себе в Єдиному реєстрі боржників. Центральне міжрегіональне управління Міністерства юстиції. URL:

https://pzmrujust.gov.ua/novyny/5643-borzhnyk-chy-ni-yak-pereviryty-sebe-v-iedynomu-reiestri-borzhnykiv (дата звернення: 07.06.2026).

10. Python 3 Documentation. Python Software Foundation. URL: https://docs.python.org/3/ (дата звернення: 07.06.2026).

11. Flask Documentation. Pallets Projects. URL: https://flask.palletsprojects.com/ (дата звернення: 08.06.2026).

12. SQLite Documentation. SQLite. URL: https://sqlite.org/docs.html (дата звернення: 08.06.2026).

13. sqlite3 — DB-API 2.0 interface for SQLite databases. Python Software Foundation. URL: https://docs.python.org/3/library/sqlite3.html (дата звернення: 08.06.2026).

14. Bootstrap 5.3 Documentation. Bootstrap. URL: https://getbootstrap.com/docs/5.3/getting-started/introduction/ (дата звернення: 08.06.2026).

15. SQL As Understood By SQLite. SQLite. URL: https://sqlite.org/lang.html (дата звернення: 08.06.2026).

16. Ровков О. О. Проєктування інформаційної системи інтеграції відкритих даних у єдину клієнтську базу фінансових установ. Інформаційні технології та моделювання систем : зб. праць учасн. Всеукр. наук.-практ. конф. здобувачів вищої освіти і молодих вчених, м. Житомир, 22 трав. 2026 р. Житомир : Поліський національний університет, 2026. С. 31–32. (дата звернення: 08.06.2026).

17. Ровков О. О. Особливості проектування архітектури інформаційної системи клієнтської бази на основі відкритих даних. Збірник матеріалів конференції, м. Львів, 29–30 трав. 2026 р. Львів, 2026. С. 104–106. URL: http://lviv-forum.inf.ua/save/2026/29-30.05/%D0%97%D0%B1%D1%96%D1%80%D0%BD%D0%B8%D0%BA.pdf (дата звернення: 08.06.2026).

18. Про затвердження Положення про набори даних, які підлягають

оприлюдненню у формі відкритих даних : постанова Кабінету Міністрів України від 21.10.2015 № 835. Верховна Рада України. URL: https://zakon.rada.gov.ua/go/835-2015-%D0%BF (дата звернення: 09.06.2026).

19. Про затвердження Положення про організацію системи управління ризиками в банках України та банківських групах : постанова Правління Національного банку України від 11.06.2018 № 64. Верховна Рада України. URL: https://zakon.rada.gov.ua/go/v0064500-18 (дата звернення: 09.06.2026).

20. Integration Definition for Function Modeling (IDEF0) : Federal Information Processing Standards Publication 183. National Institute of Standards and Technology, 1993. URL: https://nvlpubs.nist.gov/nistpubs/Legacy/FIPS/fipspub183.pdf (дата звернення: 09.06.2026).

21. OMG Unified Modeling Language, Version 2.5.1. Object Management Group. URL: https://www.omg.org/spec/UML/2.5.1/About-UML (дата звернення: 10.06.2026).

22. Shafranovich Y. Common Format and MIME Type for Comma-Separated Values (CSV) Files : RFC 4180. RFC Editor, 2005. URL: https://www.rfc-editor.org/rfc/rfc4180.html (дата звернення: 10.06.2026).

23. Extensible Markup Language (XML) 1.0 (Fifth Edition). W3C Recommendation. URL: https://www.w3.org/TR/xml/ (дата звернення: 12.06.2026).

24. Jinja Documentation. Pallets Projects. URL: https://jinja.palletsprojects.com/ (дата звернення: 10.06.2026).

25. Template Designer Documentation. Jinja Documentation. Pallets Projects. URL: https://jinja.palletsprojects.com/en/stable/templates/ (дата звернення: 10.06.2026).

26. fpdf2 Documentation. py-pdf. URL: https://py-pdf.github.io/fpdf2/index.html (дата звернення: 10.06.2026).

27. Chart.js Documentation. Chart.js. URL: https://www.chartjs.org/docs/ (дата звернення: 10.06.2026).

28. HTML: HyperText Markup Language. MDN Web Docs. URL: https://developer.mozilla.org/en-US/docs/Web/HTML (дата звернення: 10.06.2026).

29. CSS Reference. MDN Web Docs. URL: https://developer.mozilla.org/en-US/docs/Web/CSS/Reference (дата звернення: 10.06.2026).

30. Authentication Cheat Sheet. OWASP Cheat Sheet Series. URL: https://cheatsheetseries.owasp.org/cheatsheets/Authentication_Cheat_Sheet.html (дата звернення: 10.06.2026).

31. Authorization Cheat Sheet. OWASP Cheat Sheet Series. URL: https://cheatsheetseries.owasp.org/cheatsheets/Authorization_Cheat_Sheet.html (дата звернення: 11.06.2026).

32. OWASP Top Ten Web Application Security Risks. OWASP Foundation. URL: https://owasp.org/www-project-top-ten/ (дата звернення: 12.06.2026).

33. csv — CSV File Reading and Writing. Python Documentation. Python Software Foundation. URL: https://docs.python.org/3/library/csv.html (дата звернення: 12.06.2026).

34. xml.etree.ElementTree — The ElementTree XML API. Python Documentation. Python Software Foundation. URL: https://docs.python.org/3/library/xml.etree.elementtree.html (дата звернення: 12.06.2026).

35. hashlib — Secure hashes and message digests. Python Documentation. Python Software Foundation. URL: https://docs.python.org/3/library/hashlib.html (дата звернення: 12.06.2026).

36. datetime — Basic date and time types. Python Documentation. Python Software Foundation. URL: https://docs.python.org/3/library/datetime.html (дата звернення: 12.06.2026).

37. threading — Thread-based parallelism. Python Documentation. Python Software Foundation. URL: https://docs.python.org/3/library/threading.html (дата звернення: 12.06.2026).

38. Quickstart. Flask Documentation. Pallets Projects. URL: https://flask.palletsprojects.com/en/stable/quickstart/ (дата звернення: 12.06.2026).

39. Components. Bootstrap 5.3 Documentation. Bootstrap. URL:

https://getbootstrap.com/docs/5.3/components/ (дата звернення: 12.06.2026).

40. CREATE TABLE. SQLite Documentation. SQLite. URL: https://sqlite.org/lang_createtable.html (дата звернення: 12.06.2026).

ДОДАТКИ

ДОДАТОК А

Діаграми роботи інформаційної системи

У додатку А наведено діаграми, які відображають логіку роботи інформаційної системи інтеграції відкритих даних у єдину клієнтську базу фінансових установ. Діаграма діяльності показує послідовність дій користувача та системи під час перевірки клієнта. Діаграма послідовності відображає взаємодію між користувачем, вебінтерфейсом, серверною частиною, базою даних та модулем формування PDF-звіту.

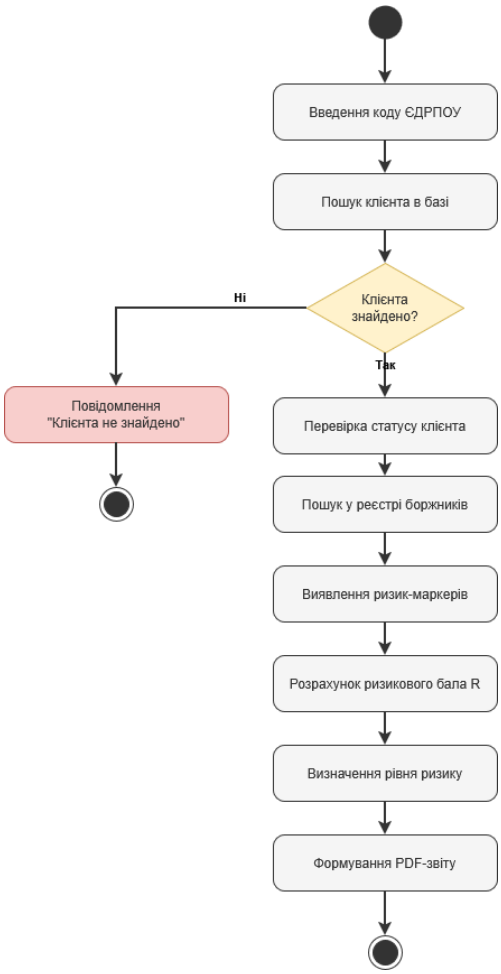


Рис. А.1. Діаграма діяльності процесу перевірки клієнта в інформаційній системі

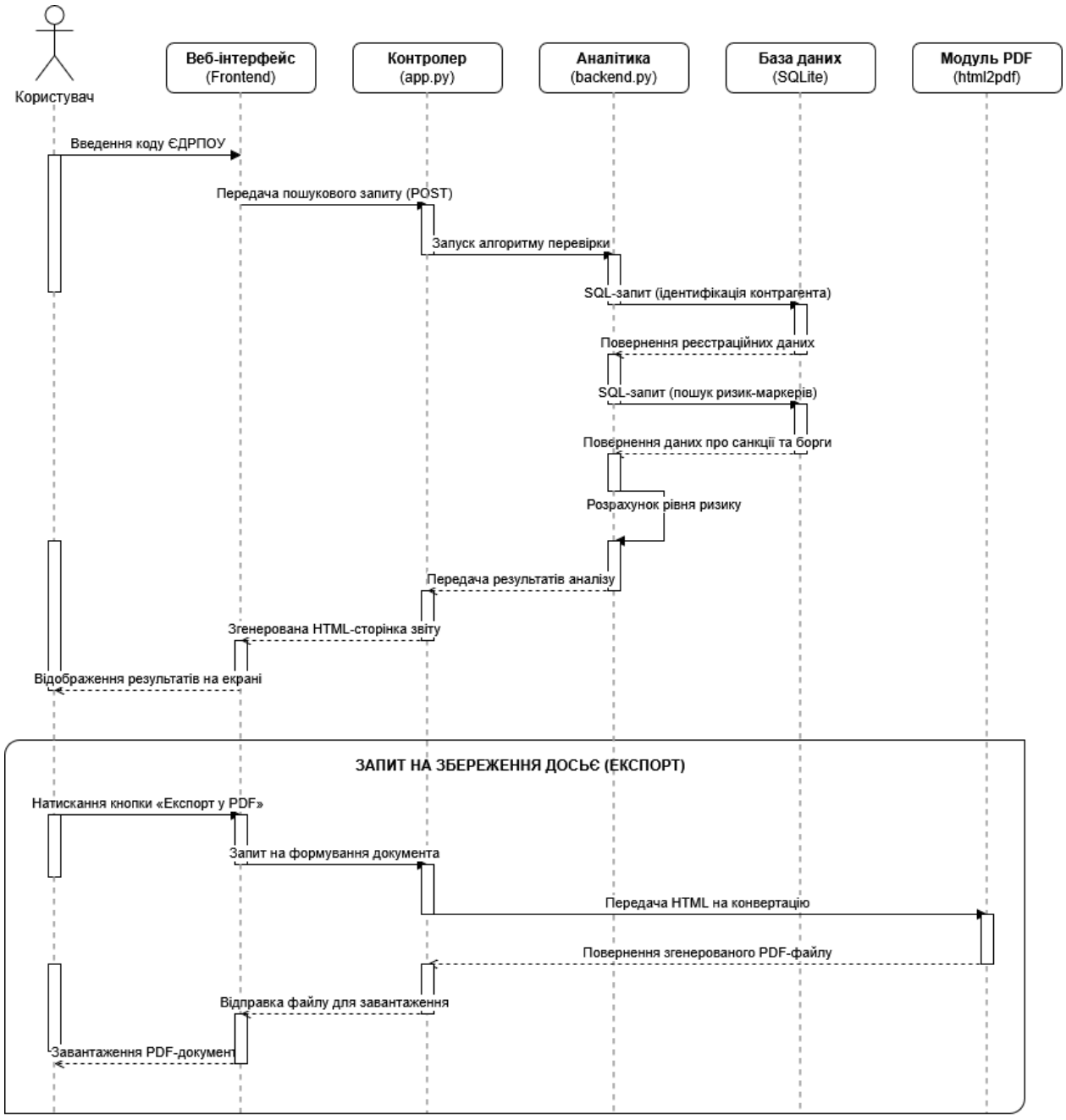


Рис. А.2. Діаграма послідовності виконання перевірки клієнта

ДОДАТОК Б

Структура програмного проєкту інформаційної системи

У додатку А наведено структуру програмного проєкту інформаційної системи інтеграції відкритих даних у єдину клієнтську базу фінансових установ. Проєкт складається з серверної частини, модулів обробки даних, шаблонів інтерфейсу користувача та допоміжних файлів для генерації демонстраційних даних.

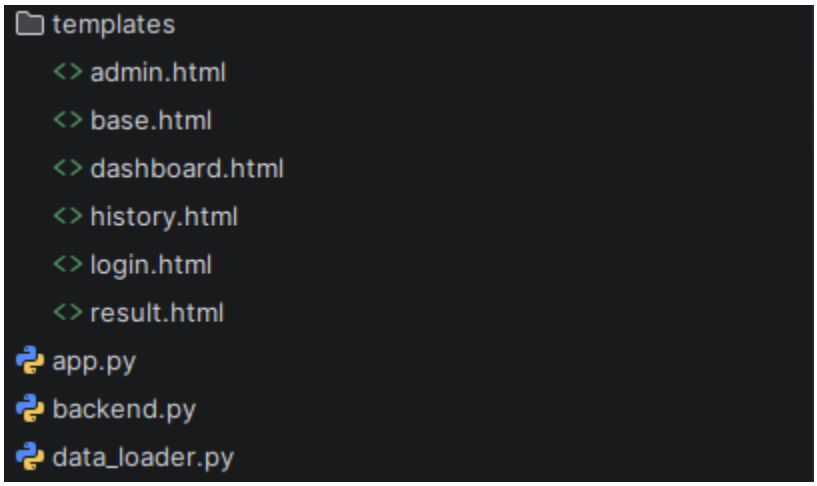


Рис. Б.1. Структура файлів програмного проєкту

ДОДАТОК В

Фрагмент програмного коду основного модуля системи

У додатку Б наведено фрагмент програмного коду основного модуля вебзастосунку. Він відповідає за маршрути системи, авторизацію користувачів, пошук клієнта, відображення результатів перевірки та формування PDF-звіту.

```
def login_required(role=None):
    def wrapper(fn):
        def decorated_view(*args, **kwargs):
            if "user_id" not in session:
                return redirect(url_for("login"))

            if role and session.get("role") != role:
                return render_template(
                    "dashboard.html",
                    error="У вас немає прав доступу до Адмін-панелі!",
                    user=session
                )

            return fn(*args, **kwargs)

        decorated_view.__name__ = fn.__name__
        return decorated_view

    return wrapper


@app.route("/auth", methods=["POST"])
def auth():
```

```
    username = request.form.get("username")
    password = request.form.get("password")

    db = DatabaseManager()
    user = db.check_user(username, password)

    if user:
        session["user_id"] = user["id"]
        session["full_name"] = user["full_name"]
        session["role"] = user["role"]

        if user["role"] == "admin":
            return redirect(url_for("admin_dashboard"))

        return redirect(url_for("dashboard"))

    return render_template("login.html", error="Невірний логін або пароль")


@app.route("/search", methods=["POST"])
@login_required()
def search():
    code = request.form.get("edrpou")

    if not code or not code.isdigit() or len(code) < 6:
        return render_template(
            "dashboard.html",
            error="Некоректний код ЄДРПОУ. Введіть числовий код клієнта.",
```

```
            user=session
        )

    client = ClientProfile(code, user_id=session["user_id"])

    if client.load_data():
        return render_template(
            "result.html",
            client=client.data,
            risks=client.risks,
            risk_score=client.risk_score,
            risk_level=client.risk_level,
            risk_factors=client.risk_factors,
            formula_details=client.formula_details,
            checked_data=client.checked_data,
            user=session
        )

    return render_template(
        "dashboard.html",
        error="Клієнта не знайдено",
        user=session
    )


@app.route("/export_pdf/<edrpou>")
@login_required()
def export_pdf(edrpou):
```

```
client = ClientProfile(edrpou, user_id=session["user_id"])
client.load_data()

pdf = FPDF()
pdf.add_page()
pdf.set_auto_page_break(auto=True, margin=15)

# Формування PDF-звіту:
# - загальна інформація про перевірку;
# - дані клієнта;
# - ризиковий бал;
# - ризик-маркери;
# - висновок системи.

filename = f"report_{edrpou}.pdf"
pdf.output(filename)

return send_file(filename, as_attachment=True)
```

Рис. В.1. Фрагмент програмного коду основного модуля app.py

ДОДАТОК Г

Фрагмент програмного коду модуля роботи з базою даних та оцінювання ризику

У додатку В наведено фрагмент програмного коду, який забезпечує роботу з клієнтськими даними, пошук ризик-маркерів та формування ризикового профілю клієнта.

```
class ClientProfile:
    def __init__(self, edrpou, user_id=1):
        self.edrpou = edrpou
        self.user_id = user_id
        self.data = None
        self.risks = []
        self.risk_score = 0
        self.risk_level = "Низький"
        self.risk_factors = []
        self.formula_details = {}
        self.checked_data = {}
        self.db = DatabaseManager()

    def calculate_risk_score(self):
        score = 0
        factors = []

        debt_records = [
            risk for risk in self.risks
            if risk["risk_type"] == "Єдиний реєстр боржників"
        ]

        name_matches = [
```

```
    risk for risk in self.risks
    if "тип збігу: назва" in risk["description"]
]

code_matches = [
    risk for risk in self.risks
    if "тип збігу: код ЄДРПОУ" in risk["description"]
]

debt_points = 0
status_points = 0
match_points = 0
records_points = 0

if debt_records:
    debt_points = 50
    score += debt_points
    factors.append({
        "marker_code": "D",
        "name": "Наявність у Єдиному реєстрі боржників",
        "points": debt_points,
        "source": "Єдиний реєстр боржників"
    })

if self.data and self.data["status"] != "АКТИВНИЙ":
    status_points = 25
    score += status_points
    factors.append({
```

```
            "marker_code": "S",
            "name": "Неактивний статус клієнта",
            "points": status_points,
            "source": "ЄДР"
        })

    if name_matches:
        match_points = 15
        score += match_points
        factors.append({
            "marker_code": "M",
            "name": "Збіг встановлено за назвою",
            "points": match_points,
            "source": "Єдиний реєстр боржників"
        })

    if len(self.risks) > 0:
        records_points = min(len(self.risks) * 10, 30)
        score += records_points
        factors.append({
            "marker_code": "N",
            "name": "Кількість знайдених ризик-маркерів",
            "points": records_points,
            "source": "Інтегрована база ризик-маркерів"
        })

    score = min(score, 100)
```

```
        if score <= 30:
            level = "Низький"
        elif score <= 70:
            level = "Середній"
        else:
            level = "Високий"

        self.risk_score = score
        self.risk_level = level
        self.risk_factors = factors

        self.formula_details = {
            "formula": "R = min(100, D + S + M + N)",
            "D": debt_points,
            "S": status_points,
            "M": match_points,
            "N": records_points,
            "calculation": f"R = min(100, {debt_points} + {status_points} + {match_points} +
{records_points}) = {self.risk_score}",
            "risk_level": self.risk_level
        }

    def load_data(self):
        self.db.connect()
        cursor = self.db.conn.cursor()

        cursor.execute(
            "SELECT * FROM registry_edr WHERE edrpou = ?",
```

```
    (self.edrpou,)
)
self.data = cursor.fetchone()

if self.data:
    client_id = self.data["id"]

    cursor.execute(
        "SELECT risk_type, description FROM risk_markers WHERE client_id = ?",
        (client_id,)
    )
    self.risks = cursor.fetchall()
    self.calculate_risk_score()
    if self.risk_score >= 71:
        status = "HIGH RISK"
    elif self.risk_score >= 31:
        status = "MEDIUM RISK"
    else:
        status = "LOW RISK"
    self.db.close()
    self.db.log_search(self.user_id, self.edrpou, status)
    return True
self.db.close()
self.db.log_search(self.user_id, self.edrpou, "NOT FOUND")

return False
```

Рис. Г.1. Фрагмент програмного коду модуля backend.py

ДОДАТОК Д

Приклад вхідних даних у форматі XML

У додатку Г наведено приклад XML-масиву, який використовується для збереження основних відомостей про клієнтів. Такий масив містить код ЄДРПОУ, назву клієнта, статус діяльності, адресу та керівника.

<?xml version='1.0' encoding='utf-8'?> <DATA> <SUBJECT> <NAME>ТЕСТОВИЙ КЛІЄНТ НИЗЬКИЙ РИЗИК</NAME> <RECORD>11111111</RECORD> <STAN>зареєстровано</STAN> <ADDRESS>м. Житомир, вул. Київська, буд. 10</ADDRESS> <BOSS_NAME>ТЕСТОВИЙ КЛІЄНТ НИЗЬКИЙ РИЗИК</BOSS_NAME> <REGISTRATION_DATE>2020-04-12</REGISTRATION_DATE> </SUBJECT> <SUBJECT> <NAME>ТЕСТОВИЙ КЛІЄНТ СЕРЕДНІЙ РИЗИК</NAME> <RECORD>22222222</RECORD> <STAN>зареєстровано</STAN> <ADDRESS>м. Київ, вул. Хрещатик, буд. 15</ADDRESS> <BOSS_NAME>ТЕСТОВИЙ КЛІЄНТ СЕРЕДНІЙ РИЗИК</BOSS_NAME> <REGISTRATION_DATE>2019-08-21</REGISTRATION_DATE> </SUBJECT> <SUBJECT> <NAME>ТЕСТОВИЙ КЛІЄНТ ВИСОКИЙ РИЗИК</NAME> <RECORD>33333333</RECORD> <STAN>припинено</STAN> <ADDRESS>м. Львів, просп. Миру, буд. 44</ADDRESS> <BOSS_NAME>ТЕСТОВИЙ КЛІЄНТ ВИСОКИЙ РИЗИК</BOSS_NAME> <REGISTRATION_DATE>2017-02-05</REGISTRATION_DATE> </SUBJECT> </DATA>

Рис. Д.1. Приклад структури XML-масиву з даними клієнтів

ДОДАТОК Е

Приклад вхідних даних у форматі CSV

У додатку Д наведено приклад CSV-масиву, який використовується для збереження записів, пов’язаних із ризиковими ознаками клієнтів. У файлі містяться дані про назву боржника, код, номер виконавчого провадження, категорію стягнення та орган, який веде провадження.

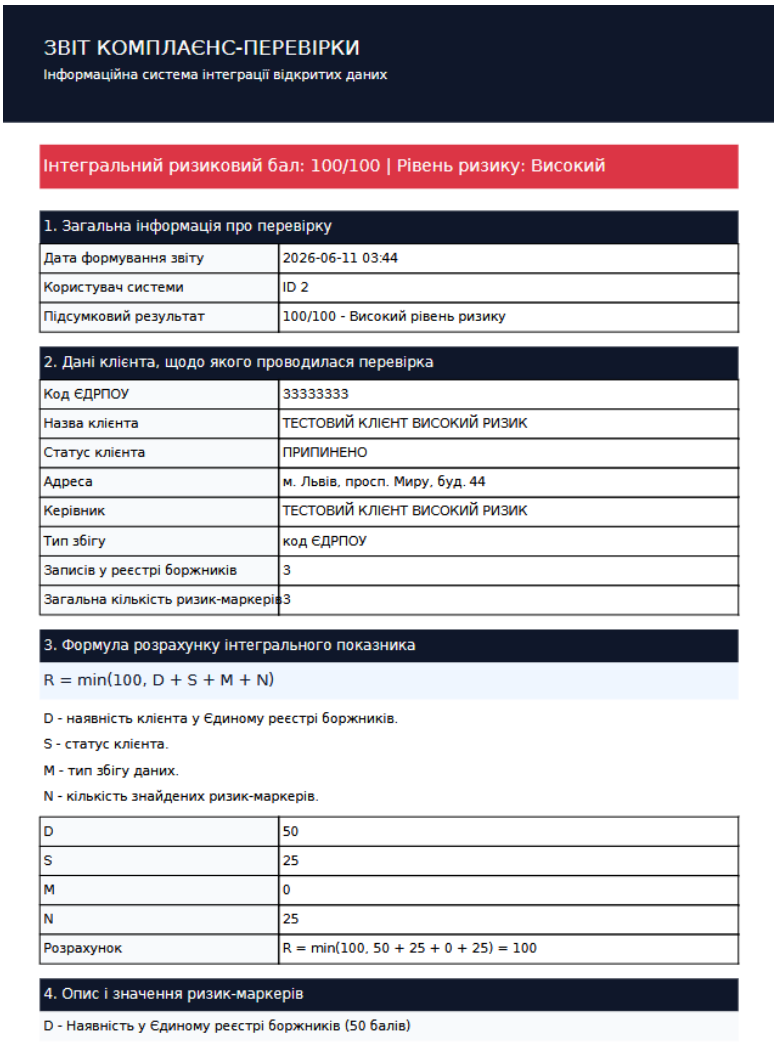
DEBTOR_NAME;DEBTOR_CODE;VP_ORDERNUM;VD_CAT;PUBLISHER;ORG_NAME ТЕСТОВИЙ КЛІЄНТ СЕРЕДНІЙ РИЗИК;22222222;58177477;Стягнення податкового боргу;Районний відділ ДВС;Міністерство юстиції ТЕСТОВИЙ КЛІЄНТ ВИСОКИЙ РИЗИК;33333333;13674265;Стягнення коштів на користь держави;Районний відділ ДВС;Міністерство юстиції ТЕСТОВИЙ КЛІЄНТ ВИСОКИЙ РИЗИК;33333333;93624514;Арешт коштів боржника;Районний відділ ДВС;Міністерство юстиції ТЕСТОВИЙ КЛІЄНТ ВИСОКИЙ РИЗИК;33333333;60328846;Стягнення штрафів у справах про адміністративні правопорушення;Районний відділ ДВС;Міністерство юстиції

Рис. Е.1. Приклад структури CSV-масиву з ризиковими записами

ДОДАТОК Ж

Приклад сформованого PDF-звіту

У додатку Е наведено приклад PDF-звіту, сформованого системою за результатами перевірки клієнта. Звіт містить основні дані клієнта, рівень ризику, перелік виявлених ризик-маркерів та загальний висновок системи.



ЗВІТ КОМПЛАЄНС-ПЕРЕВІРКИ

Інформаційна система інтеграції відкритих даних

Інтегральний ризиковий бал: 100/100 | Рівень ризику: Високий

1. Загальна інформація про перевірку	
Дата формування звіту	2026-06-11 03:44
Користувач системи	ID 2
Підсумковий результат	100/100 - Високий рівень ризику

2. Дані клієнта, щодо якого проводилася перевірка	
Код ЄДРПОУ	33333333
Назва клієнта	ТЕСТОВИЙ КЛІЄНТ ВИСОКИЙ РИЗИК
Статус клієнта	ПРИПИНЕНО
Адреса	м. Львів, просп. Миру, буд. 44
Керівник	ТЕСТОВИЙ КЛІЄНТ ВИСОКИЙ РИЗИК
Тип збігу	код ЄДРПОУ
Записів у реєстрі боржників	3
Загальна кількість ризик-маркерів	3

3. Формула розрахунку інтегрального показника

R = min(100, D + S + M + N)

D - наявність клієнта у Єдиному реєстрі боржників.

S - статус клієнта.

M - тип збігу даних.

N - кількість знайдених ризик-маркерів.

D	50
S	25
M	0
N	25
Розрахунок	R = min(100, 50 + 25 + 0 + 25) = 100

4. Опис і значення ризик-маркерів

D - Наявність у Єдиному реєстрі боржників (50 балів)

Рис. Ж.1. Приклад PDF-звіту за результатами перевірки клієнта