

РОЗРОБКА МІКРОСЕРВІСНОЇ АРХІТЕКТУРИ ДЛЯ ЕФЕКТИВНОГО РОЗПІЗНАВАННЯ ІНФОРМАЦІЇ З ЧЕКІВ

Іткін Д. О.

Науковий керівник – к.т.н., доц. Любченко В. А.

Харківський національний університет радіоелектроніки, каф. ІНФ,
м. Харків, Україна

e-mail: denys.itkin@nure.ua

This research focuses on the development of a microservices architecture tailored for efficient information extraction from receipts. The objective is to enhance the recognition process and streamline the handling of information from receipts using a modular and scalable microservices approach. The proposed architecture aims to optimize the overall efficiency and performance of information retrieval in the context of receipt processing.

У сучасному інформаційному середовищі важливу роль у різних галузях, зокрема в бізнесі та фінансовому секторі, відіграє розпізнавання інформації з чеків [1]. Для досягнення ефективності та точності цього процесу ключовим етапом є розробка мікросервісної архітектури [2].

Мікросервісна архітектура визначається як система, що складається з невеликих та незалежних компонентів, спрямованих на виконання конкретних функцій (рис. 1). У контексті розпізнавання інформації з чеків ця архітектура дозволяє розділити завдання на окремі сервіси, підвищуючи масштабованість, гнучкість та швидкість обробки [3].

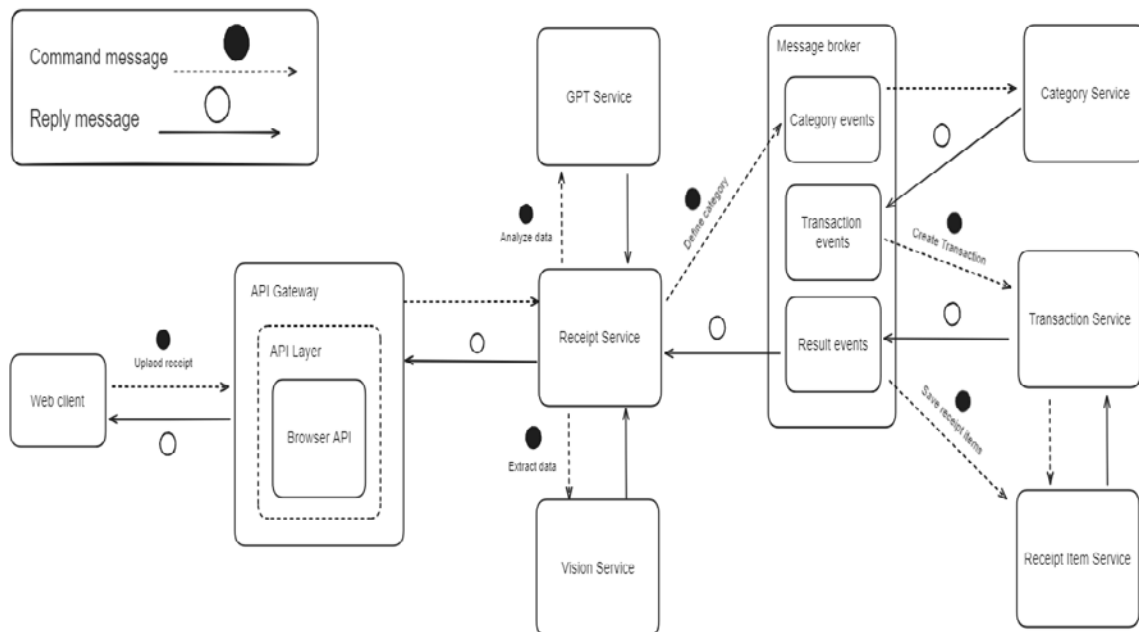


Рисунок 1 – Реалізація мікросервісної архітектури

Процес взаємодії клієнта з мікросервісною архітектурою, використовуючи Spring Cloud Gateway [2] та чотири мікросервіси, може бути описаний, виконуючи послідовність наступних етапів.

Клієнт відправляє запит на завантаження файлу (етап 1):

- клієнт ініціює запит на завантаження файлу через Spring Cloud Gateway;

- Spring Cloud Gateway перенаправляє запит на відповідний мікросервіс (Receipt Service) для обробки завантаження файлу.

Receipt Service (етап 2):

- приймає запит на завантаження файлу від Spring Cloud Gateway;
- обробляє запит і зберігає файл;
- генерує інформацію про завантаження файлу та повертає її клієнту.

Клієнт витягує інформацію з чека (етап 3). Клієнт ініціює запит на витягування інформації з чека через Spring Cloud Gateway.

Vision Service (етап 4):

- Receipt Service приймає запит на витягування інформації з чека та відправляє інформацію в Vision Service;

- витягує текстову інформацію з чека та відправляє відповідь до Receipt Service;

- Receipt Service відправляє інформацію в чергу Kafka (Category events).

GPT Service (етап 5):

- Receipt Service відправляє текст з чеку для аналізу даних в GPT Service;

- GPT Service аналізує дані та відправляє відповідь у форматі JSON;
- Receipt Service відправляє інформацію в чергу Kafka (Category events) для створення транзакції.

Category Service (слухач Kafka) (етап 6) [4]:

- слухає події з черги Kafka для категорій;
- визначає категорії;
- відправляє запит в чергу транзакцій (Transaction events).

Transaction Service (етап 7):

- приймає запит на створення транзакції від мікросервіса Category Service;

- створює транзакцію з наданою інформацією;
- відправляє створену транзакцію в чергу створених транзакцій Response events.

Мікросервіс, який ініціював подію, перехоплює транзакцію (етап 8):

- слухає чергу створених транзакцій;
- перехоплює створену транзакцію та повертає інформацію користувачу.

Приклад бізнес-процесу розпізнавання інформації з чеків наведений на рис. 2.

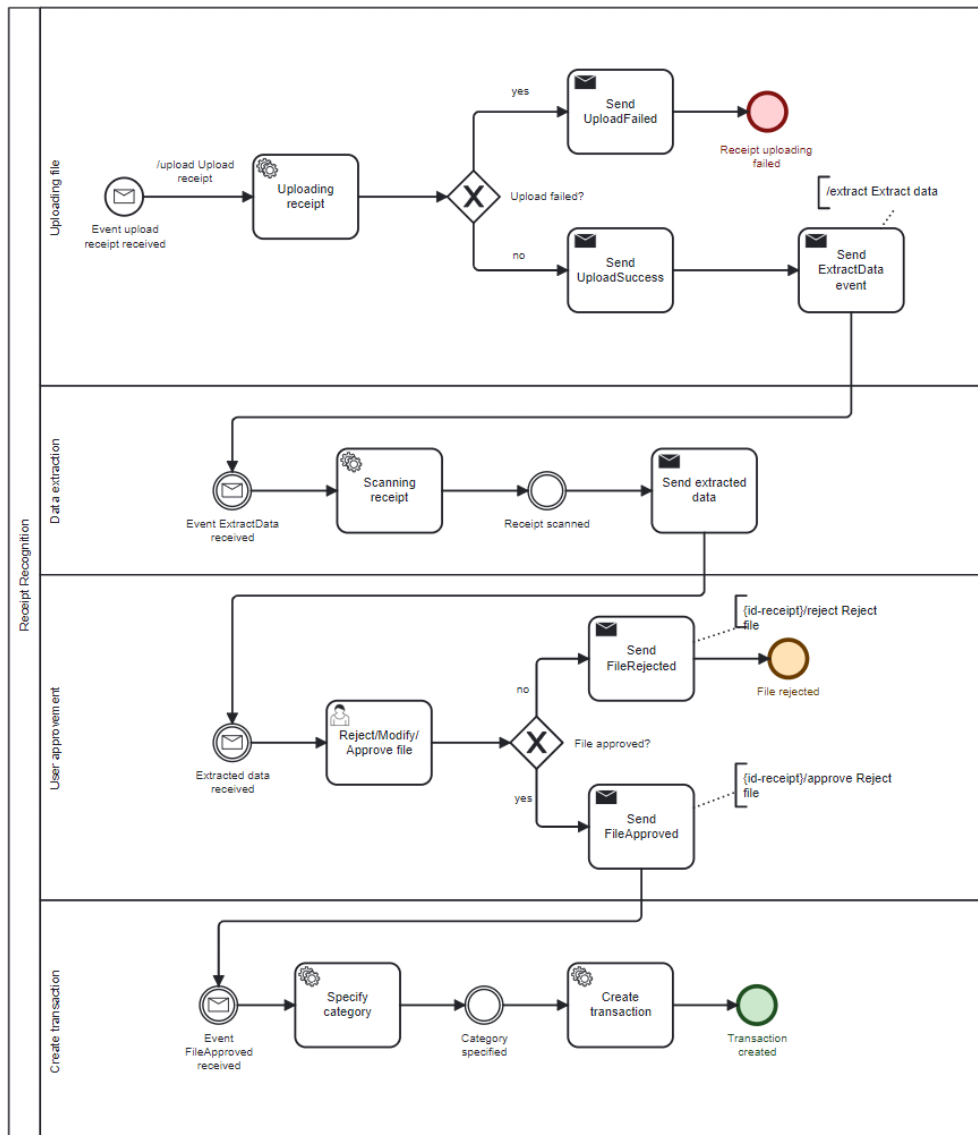


Рисунок 2 – Приклад бізнес-процесу розпізнавання інформації з чеків

Список використаних джерел:

1. Ковтуненко А. Р., Яковлева О. В., Любченко В. А., Янголенко О. В. Дослідження сумісного використання математичної морфології та згорткових нейронних мереж для вирішення задачі розпізнавання цінників / Вісник Національного технічного університету «ХПІ». Серія: Системний аналіз, управління та інформаційні технології, 2020. № 1(3). С. 24-31. URL: <https://scholar.google.com.ua/scholar?oi=bibs&cluster=10539543689639644462&btnI=1&hl=ru> (дата звернення: 15.03.2024).
2. Richardson C. Microservices Patterns: With Examples in Java. 2019. 520 p.
3. Reucker B., Freund J. Real-life BPMN. 4th Edition. 2019. P. 50–73.
4. Stanley K. Kafka Connect: Build and Run Data Pipelines. 2023. 400 p.