

УДК 004.4

В. О. Крушельницький

(Тернопільський національний технічний університет імені Івана Пулюя, Україна)

ІНТЕГРАЦІЯ ТА УПРАВЛІННЯ БІЗНЕС АКАУНТОМ FACEBOOK ТА INSTAGRAM В МОБІЛЬНОМУ ДОДАТКУ НА ПРИКЛАДІ “EASY ACCOUNTING FOR SOCIAL BIZ”

V. O. Krushelnystkyi

INTEGRATION AND MANAGEMENT OF FACEBOOK AND INSTAGRAM BUSINESS ACCOUNTS IN THE MOBILE APPLICATION USING THE EXAMPLE OF "EASY ACCOUNTING FOR SOCIAL BIZ"

Розглянемо процес інтеграції Facebook API зі сторони сервера. Робота серверної частини обробляється Hasura GraphQL Engine. Спочатку потрібно визначити схему GraphQL, яка описує доступні типи даних та запити. Схема визначає поля, які можна запитувати, а також способи отримання даних. Після визначення схеми можна створити запит, який відповідає необхідним потребам. Запит до серверу вказується мовою запитів GraphQL.

Після створення запиту потрібно надіслати його на сервер GraphQL Engine. Зазвичай використовуються HTTP-запити, такі як POST або GET. Сервер GraphQL Engine отримує запит та обробляє його згідно з визначеною схемою GraphQL. Він перетворює стрічку зі скриптом на мові GraphQL у звичайний SQL запит та звертається до бази даних або іншого джерела даних, отримує результат та формує відповідь.

Після обробки сервер GraphQL Engine повертає відповідь на клієнтську частину у вигляді об'єкту json. Відповідь може містити дані, які запитувались, або повідомлення про помилку, якщо запит був недійсним або виникла проблема під час обробки. Після отримання відповіді можна обробити дані згідно з потребами. Відповідно є можливість використовувати отримані дані для відображення на користувацькому інтерфейсі або подальшої обробки.

Використовувати Hasura GraphQL Engine ми будемо тільки для того, щоб зберегти дані про переписку користувача із клієнтами. Вхідні дані будуть однакові із вихідними (id клієнта, id користувача, дата повідомлення, текст повідомлення), оскільки немає причин зберігати лишню інформацію.

Для реалізації передачі повідомлень в реальному часі між клієнтом та продавцем використано механізм підписок (subscription) у GraphQL та механізм WebHooks. Підписки дозволять клієнтській частині підписатися на певні події або оновлення даних та отримувати повідомлення в реальному часі при виникненні цих подій. WebHooks, у свою чергу, дозволяють серверу надсилати дані до клієнтської частини, щоб повідомляти про події у режимі реального часу.

Висновок. Розглянуто розробку інтеграції Facebook API в мобільний додаток зі сторони серверу. Реалізовано функціонал взаємодії.

Література

1. Офіційна документація GraphQL: веб-сайт. URL: <https://graphql.org/learn>
2. Офіційна документація Hasura. <https://hasura.io/docs>
3. Клієнт-серверна архітектура з використанням ресурсного серверу Hasura. <https://hasura.io/blog/hasura-for-existing-graphql-rest-apis-architect-guide/>
4. Механізм підписок (Subscription) у базі даних PostgreSQL та серверу Hasura. URL: <https://hasura.io/docs/latest/subscriptions/postgres/index/>
5. Webhook mode Hasura. URL: <https://hasura.io/learn/graphql/hasura-authentication/webhook-mode/> Instagram Message API for Instagram. URL: <https://developers.facebook.com/products/messenger/messenger-api-instagram/>