

ДОДАТОК А

Графічний матеріал кваліфікаційної роботи

Харківський національний університет радіоелектроніки

Кафедра ЕОМ

Архітектурна модель Full-Stack проекту

Автор:
студент групи СПм-22-4
Комаричев А. В.

Керівник:
доцент каф. ЕОМ, к.т.н.
Філімончук Т.В.

Мета:

- дослідити наявні моделі створення Full-Stack проекту та створення більш гнучкої та масштабованої моделі;
- проаналізувати наявні моделі створення Full-Stack застосунків та технології, які доступні для використання;
- запропонувати більш новітню та гнучку модель, проаналізувати можливості її використання в застосунках різної складності

Значення Full-Stack розробки

- гнучкість;
- комплексність;
- оптимізація;
- актуальність.

3

Огляд запропонованих технологій



Flutter

мобільні девайси
та веб



FastAPI

серверний
застосунок



PostgreSQL

база даних



docker

контейнеризація

4

Існуюча модель фреймворку

$$M = \{UI, CF, IL, BL, \{DB, CS\}, API, SM, MLog, MA, TM\},$$

UI (User Interface): інтерфейс взаємодії користувача;

CF (Client Frontend): клієнтська частина для iOS та Android;

IL (Interaction Logic): обробка даних та комунікація з функціями програми;

BL (Business Logic): виконання основних функцій програми;

DB (Database): зберігання та організація інформації;

CS (Cloud Services): хмарні сервіси для зберігання файлів та управління ресурсами;

5

Існуюча модель фреймворку

$$M = \{UI, CF, IL, BL, \{DB, CS\}, API, SM, MLog, MA, TM\},$$

API (Application Programming Interface): комунікація між клієнтом та сервером;

SM (Security Module): механізм безпеки, який включає аутентифікацію та авторизацію;

MLog (Monitoring and Logging): аналіз та моніторинг використання додатку;

MA (Monitoring and Analytics): модуль моніторингу та аналітики;

TM (Testing Module): модуль для створення та проведення тестів.

6

Запропонована модель фреймворку

$$M' = \{UI, CF, IL, BL, \{DB, CS\}, API, SM, MLog, MA, TM, \\ MS, DO, ML, KB, AD, MP\}$$

MS (Microservice Architecture): мікросервісна архітектура;

DO (Data Orchestration): оркестрація даних;

ML (Machine Learning): машинне навчання;

KB (Caching): кешування;

AD (Automated Deployment): автоматизація розгортання.

7

Переваги розширеної моделі

- підвищена гнучкість:
 - мікросервісна архітектура дозволяє незалежно розробляти та розгортати окремі компоненти;
 - використання кешування та оркестрації даних забезпечує вищу продуктивність;
- розширена аналітика та машинне навчання:
 - забезпечує більш інтелектуальні функції додатку та передбачення поведінки користувачів;
 - забезпечує нові можливості для монетизації та підвищить зручність для користувачів;
- покращена безпека та автоматизація розгортання:
 - дозволяє швидко та ефективно впроваджувати зміни та нові функції;
 - економить ресурси команди.

8

Рішення для малих проєктів

- у повному обсязі модель буде надто складною для простих застосунків:
 - можливе часткове використання моделі, що вирішує цю проблему;
 - розробка простого проєкту з використанням моделі допоможе з часом швидко масштабувати проєкт та доповнити його функціоналом, не змінюючи основу проєкту.

9

Використання моделі для застосунку для підрахунку калорій



Flutter

UI, CF, IL



Firebase

API, CS

10

Використання моделі для застосунку для трейдингової біржі



11

Висновки

- проведено порівняльний аналіз існуючої моделі розробки Full-Stack застосунків;
- запропоновано комплексну модель, яке є більш масштабованою та може використовуватися для розробки застосунків ширшого спектру;
- запропоновано використання актуальних технологій, які можливо використовувати з моделлю з ціллю ефективної розробки.

12