

УДК 004.93:72

ДВА ПРИНЦИПИ ЯКІСНОЇ АРХІТЕКТУРИ, ВИКОРИСТАНІ У WEB-ДОДАТКУ PHOTOBOOM

Ларченко О.А., здобувачка вищої освіти гр. ЗМКІп-201

Науковий керівник: **Базилевич В.М.**, к.е.н., доц.

Національний університет «Чернігівська політехніка»

Цей WEB-додаток запланований для професійних фотографів, які можуть залишати свої фотографії на цьому сайті, тут буде їх портфоліо, можна буде залишати вподобайку та коментар під кожним постом.

На Back-end-i використовувався Node.js, який зв'язується з базою даних MySQL. На Front-end-i використовувався фреймворк Angular.

Архітектура - це базова організація системи, втілена в її компонентах, їх відносинах між собою і з оточенням, а також принципи, що визначають проектування і розвиток системи.

Два стовпи якісної архітектури:

– принцип розділення проблем (separation of concern або SoC) – це відокремлення комп'ютерної програми на окремі рівні, так що кожен рівень відповідає за окрему проблему. Іншими словами, цей принцип допоможе нам визначити необхідні рівні та обов'язки кожного рівня;

– принцип "тримайте все простим" (keep it simple stupid або KISS) – більшість систем працюють краще, якщо вони залишаються простими, а не ускладненими, тому простота повинна бути головною метою в дизайні і непотрібної складності слід уникати.

Дана архітектура WEB-додатку складається з трьох рівнів:

– рівень контролерів – це верхній рівень веб-додатку. Він відповідає за обробку запитів користувача та повертання коректної відповіді назад. Рівень контролерів має також обробляти виняткові ситуації, які виникли на інших рівнях. Оскільки рівень контролерів є вхідною точкою нашого додатку, тому він має піклуватися аутентифікацією і має діяти як перша лінія оборони проти неавторизованих користувачів;

– рівень сервісів знаходиться нижче веб-рівня за ієрархією. Він діє як проміжний рівень та містить як інфраструктурні, так і сервіси додатку. Сервіси додатку надають публічне API сервісного рівня. Вони також діють як проміжні рівні транзакцій та відповідальні за авторизацію. Інфраструктурні сервіси містять код, який з'єднується за зовнішніми ресурсами, такими як файлові системи, бази даних або поштові сервери. Часто ці методи використовують більш, ніж один сервіс додатку;

– рівень моделей є найнижчим рівнем у веб-додатку. Він відповідає за зв'язок із використовуваним сховищем даних.

Важливе правило: компоненти, які належать до певного рівня, можуть використовувати компоненти, які належать до того ж самого рівня або рівня під ним.

На рисунку 1 зображена архітектура WEB-додатку PhotoBoom, яка складається з трьох рівнів.

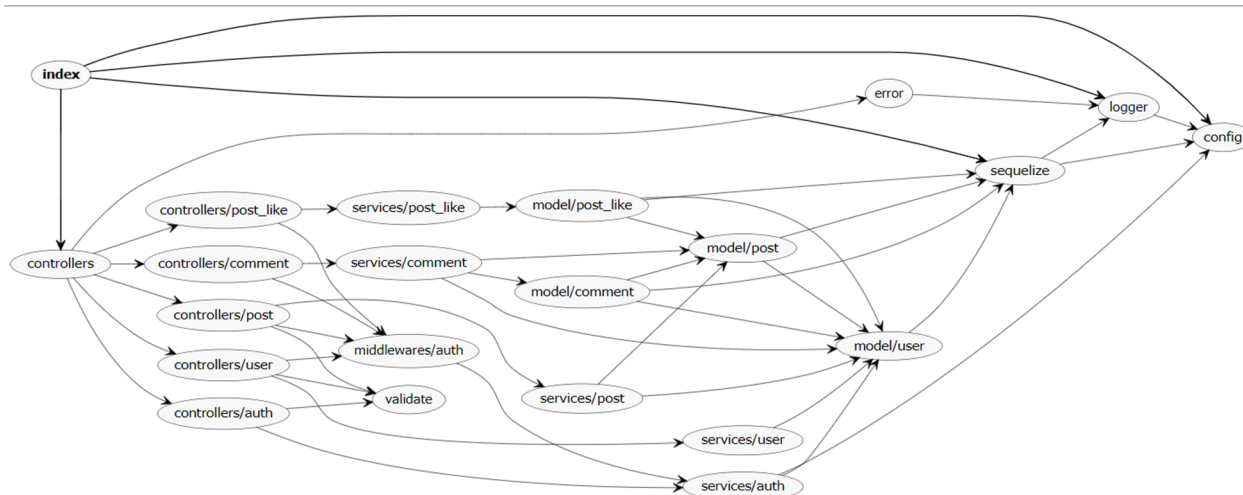


Рисунок 1 – Архітектура back-end-частини додатку PhotoBoom

Отже, можна сказати, що ці два принципи найважливіші в проектуванні додатків. Вони розділяють програму на окремі рівні, так що кожен рівень відповідає за окрему проблему. І це в майбутньому допоможе програмісту додати новий функціонал в програму, бо кожен рівень розділений на окремі компоненти, в яких описано одну чи дві поведінки системи.

Список використаних джерел

1. Поняття архітектури системи [Електронний ресурс]. – Режим доступу до ресурсу: <http://lib.mdpu.org.ua/e-book/vstup/L6.htm>
2. KISS (принцип) [Електронний ресурс]. – Режим доступу до ресурсу: [https://ru.wikipedia.org/wiki/KISS_\(%D0%BF%D1%80%D0%B8%D0%BD%D1%86%D0%B8%D0%BF\)](https://ru.wikipedia.org/wiki/KISS_(%D0%BF%D1%80%D0%B8%D0%BD%D1%86%D0%B8%D0%BF))
3. Принцип розділення проблем [Електронний ресурс]. – Режим доступу до ресурсу: https://ru.wikipedia.org/wiki/%D0%A0%D0%B0%D0%B7%D0%B4%D0%B5%D0%BB%D0%B5%D0%BD%D0%B8%D0%B5_%D0%BE%D1%82%D0%B2%D0%B5%D1%82%D1%81%D1%82%D0%B2%D0%B5%D0%BD%D0%BD%D0%BE%D1%81%D1%82%D0%B8#