

# АРХІТЕКТУРА FRONTEND РОЗРОБКИ

Вінницький національний технічний університет

## Анотація

*В даній роботі розглянуто важливість архітектури у розробці сучасних веб-додатків. Також наведені різні моделі, їхні приклади, переваги і недоліки.*

**Ключові слова:** *архітектура веб-додатків, MVC, MVVM, RIA.*

## Abstract

In this work was considered influence web application architecture for developing web-app. There some models and examples with their advantage and disadvantage.

**Keywords:** web application architecture, MVC, MVVM, RIA.

## Вступ

Вивчення потоків даних, в результаті стали архітектурою сучасних веб-додатків. Веб-додатки змінювалися від простих статичних сайтів в більш прогресивні, які в результаті стали SPA, SSR системами, які мають доступ до різних API для отримання інформації. Все починалося з зміни різних елементів DOM, за допомогою бібліотеки jQuery, можна було маніпулювати елементами. А далі архітектура веб-додатків була основана на MVC. Далі бібліотека React почала використовувати однонаправлений потік інформації. Також інші бібліотеки та фреймворки містять у собі певну архітектуру, за допомогою якої здійснюється передача інформації.

Оскільки сучасний веб це різні бібліотеки та фреймворки, які допомагають у розробці веб-додатків, кожен front-end developer повинен знати про архітектуру веб-додатків. А також вміти правильно використовувати ці знання, і покращувати веб-додаток.

## Огляд архітектур веб-додатків

Існує 4 основних архітектури для розробки веб-додатків. Кожна архітектура має свої переваги та недоліки. Чотири основних архітектурних підходи такі: MVC, MVVM, MVP, RIA.

Розглянемо кожен із них:

Перший буде MVC[1]. Такий підхід до архітектури, базується на відокремлені частини представлення і частини де є дані. Базисом для даного типу архітектури був принцип SoC(розділення відповідальності), тобто кожен модуль, має певне призначення, і обробляє певні дані, чи процеси. Дана архітектура розроблялася уже дуже давно, і не враховує деякі моменти в архітектурі сучасних веб-додатків. Контролер контролював на вхід інформації від користувача, і потім робив дії над моделлю. Це і недолік даної архітектури.

Зображення архітектури MVC на рисунку 1.

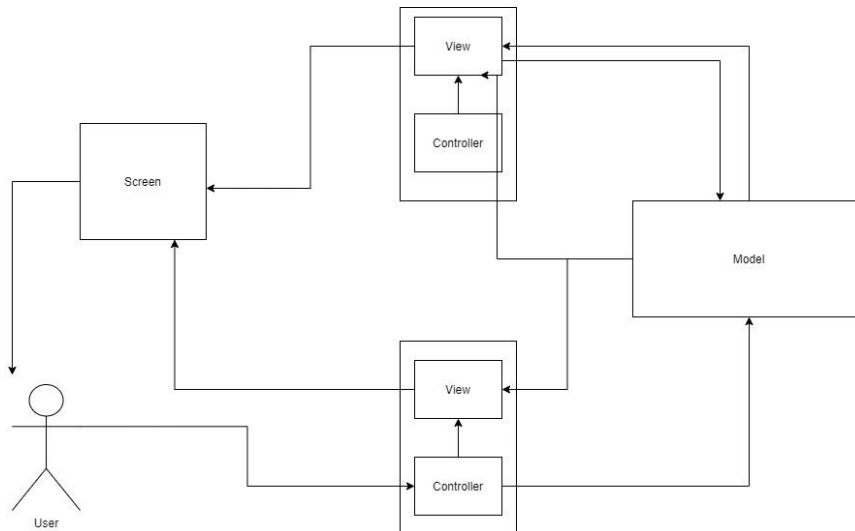


Рисунок 1 – MVC модель

Наступна модель MVVM. Має прив'язку даних, яка покращує розробку веб-додатків і контролювання даних. Дана модель зв'язує дані від користувача, і дані від сервера, і синхронізує їх. Перевагою є те, що не потрібно писати великий код для синхронізації. А також дає можливість працювати на високому рівні абстракції. Такий підхід є більш відомим в веб програмуванні. MVVM зображено на рисунку 2 [2].

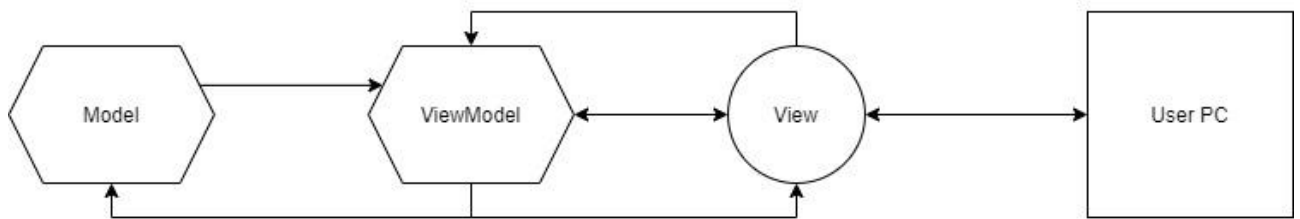


Рисунок 2 – MVVM модель

І остання модель RIA [3]. В сучасній веб-розробці є дуже актуальною, оскільки дозволяє отримувати дані з інтернет джерел, і використовувати на своїй сторінці, що є перевагою. Налаштована на використання фреймворків та бібліотек, які є все більш необхідними при розробці. Використовується для односторінкового додатку. Модель зображена на рисунку 3.

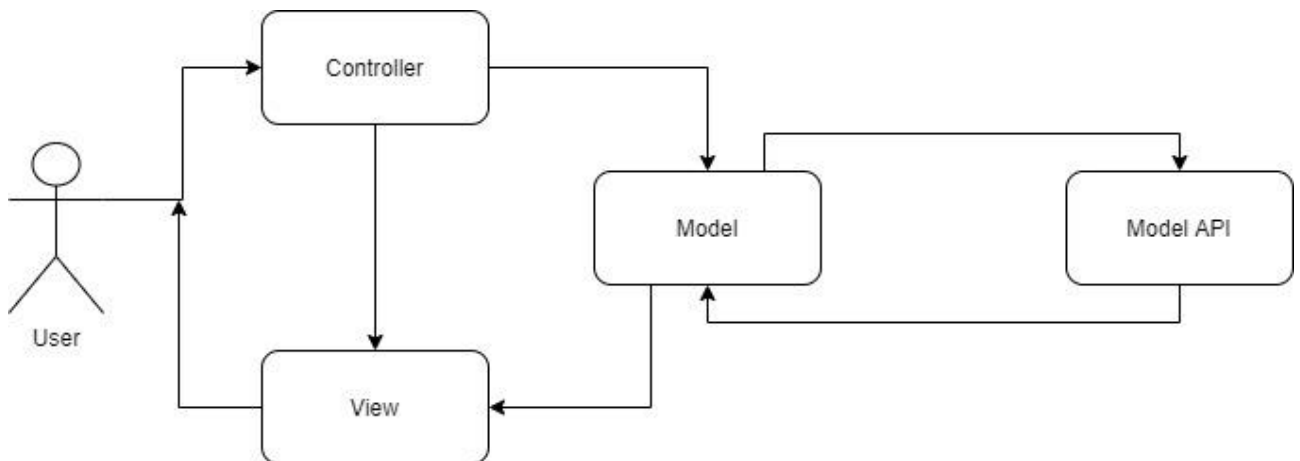


Рисунок 3 – RIA модель

В результаті чого, дана модель архітектури веб-додатків стає все більш актуальною.

## Висновки

В результаті, можна зробити такий висновок, що архітектура є дуже важливою частиною розробки веб-додатків, і потрібно не тільки знати про дані моделі архітектури, а також вміти використовувати їх на практиці. Кожна модель має свої переваги та недоліки, що можна використовувати для розробки веб-додатків.

## СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Что такое MVC: рассказываем простыми словами – [Електронний ресурс]. – Режим доступу до матеріалу: <https://ru.hexlet.io/blog/posts/chto-takoe-mvc-rasskazyvaem-prostymi-slovami>
2. Патерн MVVM – [Електронний ресурс]. – Режим доступу до матеріалу: <https://metanit.com/sharp/wpf/22.1.php>
3. Современные архитектуры фронт-энда – [Електронний ресурс]. – Режим доступу до матеріалу: <https://habr.com/ru/post/500072/>

**Козубенко Максим Володимирович** — студент групи ІПІ-176, факультет інформаційних технологій та комп'ютерної інженерії, Вінницький національний технічний університет, м. Вінниця, e-mail: [max.kozubenko4@gmail.com](mailto:max.kozubenko4@gmail.com)

Науковий керівник – **Бабюк Наталія Петрівна**, кандидат технічних наук, доцент кафедри програмного забезпечення, Вінницький національний технічний університет, м. Вінниця, e-mail:

**Kozubenko Maksym Volodymyrovych** — Faculty of Information Technologies and Computer Engineering, Vinnytsia National Technical University, Vinnytsia, e-mail: [max.kozubenko4@gmail.com](mailto:max.kozubenko4@gmail.com)

Supervisor – **Babuk Natalia Petrivna**, PhD, Associate Professor of Software Engineering Department, Vinnytsia National Technical University, Vinnytsia, e-mail: