

РОЗРОБКА КЛІЄНТСЬКОЇ ЧАСТИНИ АДАПТИВНОЇ ТЕСТУВАЛЬНОЇ СИСТЕМИ З ФОТОКОНТРОЛЕМ З ВИКОРИСТАННЯМ ТЕХНОЛОГІЙ JAVASCRIPT/TYPESCRIPT ТА ФРЕЙМВОРКУ ANGULAR

Вінницький національний технічний університет

Анотація

Розглянемо принципи розробки клієнтської частини адаптивної тестувальної системи з фотоконтролем з використанням технологій Javascript/Typescript та фреймворку Angular.

Ключові слова: *фотоконтроль, Angular, JavaScript, TypeScript, клієнтська частина, фреймворк.*

Abstract

The paper is devoted to the issues of developing the frontend of the adaptive testing system with photocontrol using Javascript / Typescript technologies and the Angular framework.

Keywords: *photocontrol, Angular, JavaScript, TypeScript, frontend, framework.*

Вступ

Розробка вебзастосунків на сьогодні є однією з найпоширеніших сфер у розробці програмного забезпечення. Вебзастосунок (іноді вебдодаток) – розподілений застосунок, в якому клієнтом виступає браузер, а сервером – вебсервер. Браузер може бути реалізацією так званих тонких клієнтів – логіка застосунку зосереджується на сервері, а функція браузера полягає переважно у зображенні інформації, завантаженої мережею з сервера, і передачі назад даних користувача. Однією з переваг такого підходу є той факт, що клієнти не залежать від конкретної операційної системи користувача, тому вебзастосунки є міжплатформовими сервісами. Унаслідок цієї універсальності й відносної простоти розробки вебзастосунків стали широко популярними в кінці 1990-х – початку 2000-х років [1].

Істотною перевагою побудови вебзастосунків для підтримки стандартних функцій браузера є те, що функції повинні виконуватися незалежно від операційної системи клієнта. Замість того, щоб писати різні версії для Microsoft Windows, Mac OS X, GNU/Linux й інших операційних систем, застосунок створюється один раз для довільно обраної платформи та на ній розгортається. Проте різна реалізація HTML, CSS, JavaScript DOM й інших специфікацій в браузерах може викликати проблеми при розробці вебзастосунків і подальшої підтримки. Крім того, можливість користувача налаштовувати багато параметрів браузера (наприклад, розмір шрифту, кольори, відключення підтримки сценаріїв) може перешкоджати коректній роботі застосунку.

Довгий час прості, неінтерактивні вебсайти керували територією Інтернету, але це змінювалося і продовжуватиме змінюватися, оскільки ми дивимось у майбутнє, де Інтернет стане більш інтерактивним та поглинаючим середовищем. Оскільки програми стали більш інтерактивними, зросла складність створення таких програм. Щоб впоратися з цією складністю, у мові програмування JavaScript був створений фреймворк Angular.

Метою роботи є опис принципів розробки клієнтської частини адаптивної тестувальної системи з фотоконтролем з використанням технологій javascript/typescript та фреймворку Angular.

Об'єктом дослідження є процес створення адаптивних тестувальних систем з фотоконтролем.

Предметом дослідження є засоби програмування та побудови клієнтської частини вебзастосунку з використанням наступних мов програмування та технологій: Angular, JavaScript, TypeScript.

Головною задачею є показати принципи побудови та побудувати клієнтську частину вебзастосунку.

Розробка клієнтської частини адаптивної тестувальної системи

Частина проекту, яка містить в собі реалізацію бізнес-логіки, а саме взаємодію з сервером та базою даних, логіку обрахунків (якщо вони розробляються безпосередньо в вебдодатку), чи логіку структуризації даних та взаємодії з ними, написана за допомогою фреймворку Angular та мови програмування JavaScript/TypeScript. Схему роботи фреймворку подано на рисунку 1.

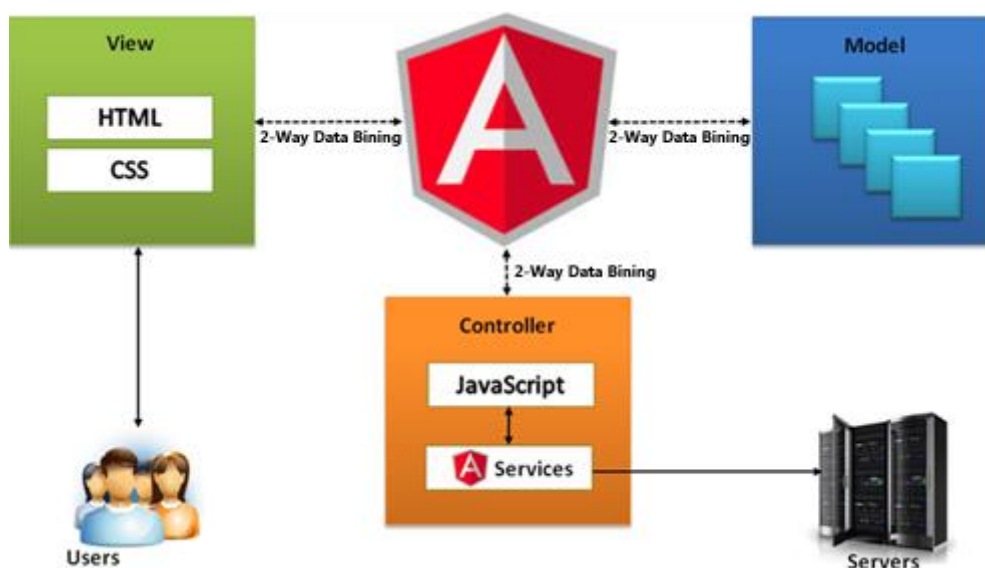


Рис. 1. Схему роботи фреймворку Angular

В основі Angular лежить архітектурний шаблон Model-View-Controller (MVC). Метою MVC є розділення логічних одиниць на окремі блоки дій, щоб дати розробникам спосіб логічно розділити відповідальність робочих частин своєї програми.

Шаблон MVC ділить додаток на три частини:

Модель — поводитьися як рушійна сила програми, де зберігаються всі дані та завантажуються з сервера. Кожного разу, коли користувач взаємодіє з інтерфейсом користувача, що включає дані, вони будуть виходити з моделі.

Перегляд — це основний інтерфейс користувача, який бачить і з яким взаємодіє користувач. Він зміниться залежно від поточної моделі програми.

Контролер — ось де знаходиться бізнес-логіка. Це рівень презентації, який виконує різні дії, такі як отримання даних, як представити модель, що відображати тощо. Як звучить назва, він контролює дії програми [2].

Як згадувалося раніше, сила дизайну MVC полягає в його розподілі праці. Кожна з вищезгаданих частин відповідає за одне і лише одне. Модель — це дані, подання — це інтерфейс користувача, а контролер — бізнес-логіка. Кожна з цих частин незалежна одна від одної, що робить код більш модульним, придатним для повторного використання та набагато легшим у догляді для постійно мінливих вебпрограм, які існують сьогодні і будуть існувати завтра [3].

На сьогодні існує багато популярних фреймворків та бібліотек для написання вебзастосунків, але найпопулярніші з них це: Angular, React та Vue. Всі вони використовують мови програмування JavaScript/TypeScript. React це бібліотека яка немає великого набору вбудованих інструментів, більшість з них потрібно встановлювати додатково [4], на відміну від Vue та Angular. Vue це відносно молодий фреймворк, який поєднує у собі якості як React та Angular та має непоганий набір інструментів [5]. Angular має великий набір інструментів для рішення різних задач, чітке архітектурне рішення, хорошу вбудовану взаємодію з сервером та добру ефективність роботи [2].

Висновок

Отже, ми розглянули застосування Angular для виконання створення бізнес-логіки, що є базовою для більшості сучасних вебдодатків. Проте дана технологія дозволяє реалізовувати набагато складніші системи, із тривіальними математичними операціями, взаємодією з базою даних,

серверними додатками та навіть мобільними. Тобто Angular є однією з найбільш сильних та інноваційних технологій і її застосування є оптимальним для побудови клієнтської частини адаптивних тестувальних систем з фотоконтролем.

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Інтернет-протал статистичних даних Wikipedia [Електронний ресурс].-Режим доступу: <https://uk.wikipedia.org/wiki/Вебзастосунок> (дата звернення: 09.05.2021).-Назва з екрана.
2. Офіційна документація фреймворку Angular [Електронний ресурс].-Режим доступу: <https://angular.io/> (дата звернення: 09.05.2021).-Назва з екрана.
3. Інтернет-спільнота JS розробників InDepthDev [Електронний ресурс].-Режим доступу: <https://indepth.dev/posts/1134/working-with-dom-in-angular-unexpected-consequences-and-optimization-techniques> (дата звернення: 09.05.2021).-Назва з екрана.
4. Офіційна документація React [Електронний ресурс].-Режим доступу: <https://reactjs.org/> (дата звернення: 09.05.2021).-Назва з екрана.
5. Офіційна документація Vue [Електронний ресурс].-Режим доступу: <https://vuejs.org/> (дата звернення: 09.05.2021).-Назва з екрана.

Кирнасюк Євген Сергійович — студент групи 2ПІ-186, факультет інформаційних технологій та комп'ютерної інженерії, Вінницький національний технічний університет, м. Вінниця, e-mail: kirnasuk1284@gmail.com

Кательніков Денис Іванович, кандидат технічних наук, доцент кафедри програмного забезпечення, Вінницький національний технічний університет, м. Вінниця, e-mail: fuzzy2dik@gmail.com.

Kyrnasyuk Yevhen Serhiiovych – student of group 2PI-18b, Faculty of Information Technologies and Computer Engineering, Vinnytsia National Technical University, Vinnytsia, e-mail: kirnasuk1284@gmail.com

Katielnikov Denys Ivanovych, PhD, Associate Professor of Software Engineering Department, Vinnytsia National Technical University, Vinnytsia, E-mail: fuzzy2dik@gmail.com.