

ДонНТУ. Серія "Інформатика, кібернетика та обчислювальна техніка". - 2016. - № 2 (23). - С. 45-51.

5. Дудник О. Аналіз методів фільтрації текстур [Текст] / О. Дудник, О. Н. Романюк // Міжнародна науково-практична Інтернет-конференція "Молодь в технічних науках: дослідження, проблеми, перспективи", 23-26 квітня 2015 р. Вінниця: ВНТУ, 2015.

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

Кирнасюк Є. С.

*здобувач першого (бакалаврського) рівня вищої освіти факультету
інформаційних технологій та комп'ютерної інженерії,
Вінницького національного технічного університету*

Майданюк В. П.

*науковий керівник, кандидат технічних наук,
доцент кафедри програмного забезпечення,
Вінницького національного технічного університету*

Розробка веб застосунків на сьогодні є однією з найпоширеніших сфер у розробці програмного забезпечення. Веб застосунок (іноді веб додаток) – розподілений застосунок, в якому клієнтом виступає браузер, а сервером – веб сервер. 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 веб застосунки є міжплатформовими сервісами. Унаслідок цієї універсальності й відносної простоти розробки веб застосунки 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 веб застосунків 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 веб-сайти керували територією Інтернету, але це змінювалося і продовжуватиме змінюватися, оскільки ми дивимось у

майбутнє, де Інтернет стане більш інтерактивним та занурювальним середовищем. Оскільки програми стали більш інтерактивними, зростає складність створення таких програм. Щоб впоратися з цією складністю, був створений 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 технології створення веб застосунків за допомогою фреймворку Angular.

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

Головною задачею є показати принципи побудови та побудувати клієнтську частину веб застосунку.

Частина проекту, яка містить в собі реалізацію бізнес-логіки, а саме взаємодію з сервером та базою даних, логіку обрахунків (якщо вони робляються безпосередньо в веб додатку), 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.

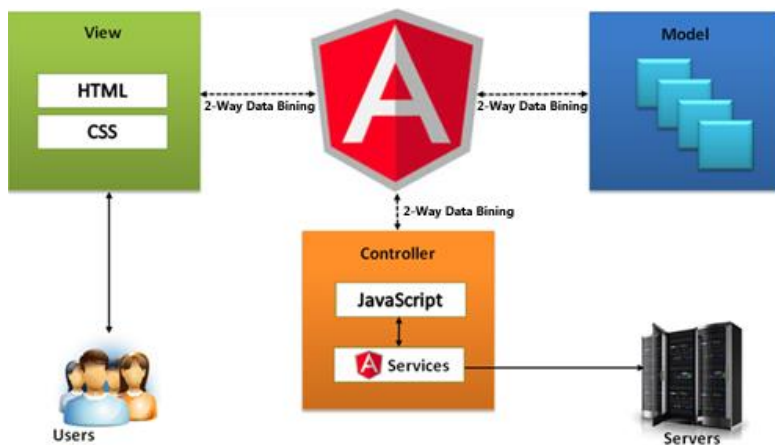


Рис. 1 Схему роботи фреймворку Angular

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 — поводить 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 веб-програм, які існують сьогодні і будуть існувати завтра [3].

На сьогодні існує багато популярних фреймворків та бібліотек для написання веб застосунків, 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 до встановлювати додатково [4], навідмінно 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 непогану ефективність роботи [2].

Отже, ми розглянули застосування Angular для виконання створення бізнес-логіки, що є базовою для більшості сучасних веб додатків. Проте дана технологія дозволяє реалізувати 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.

ЛІТЕРАТУРА

1. Інтернет-протал статистичних даних Wikipedia: <https://uk.wikipedia.org/wiki/Вебзастосунок>
2. Офіційна документація фреймворку Angular: <https://angular.io/>
3. Інтернет-спільнота JS розробників InDepthDev: <https://indepth.dev/posts/1134/working-with-dom-in-angular-unexpected-consequences-and-optimization-techniques>
4. Офіційна документація React: <https://reactjs.org/>
5. Офіційна документація Vue: <https://vuejs.org/>

ВАЖЛИВІСТЬ ПОШИРЕННЯ СТАНДАРТУ WEB 3.0 ДЛЯ КРАЩОГО ДОСВІДУ ВИКОРИСТАННЯ ГЛОБАЛЬНОЇ МЕРЕЖІ

Кічак Б. В.

*здобувач першого (бакалаврського) рівня вищої освіти
Національного університету біоресурсів і
природокористування України*