

Нарешті, дуже важливо навчити студентів етичним і правовим аспектам роботи зі штучним інтелектом і системами аналізу даних. Цифрові фахівці повинні розуміти потенційний вплив своєї роботи на окремих людей і суспільство в цілому, а також бути обізнаними з правовими рамками і нормами, що регулюють штучний інтелект та аналіз даних.

Отже, використання штучного інтелекту та аналізу даних набуває все більшого значення в різних галузях, і дуже важливо готувати майбутніх фахівців цифрових технологій у цих сферах. Університети та навчальні центри повинні надавати своїм студентам теоретичні знання, практичні навички та можливості для реального застосування штучного інтелекту та аналізу даних. Таким чином, ми зможемо забезпечити наступне покоління цифрових спеціалістів усім необхідним для вирішення проблем цифрового підґрунтя, що стрімко розвивається.

Список використаних джерел

1. Forbes Insights. "Preparing for the future: How to upskill the workforce for the AI era." 2021.
2. Brookings Institution. "Building an AI talent pipeline: How education can address the workforce shortage in AI." 2019.
3. Zimmerman, Joshua and De Veaux, Richard. "Teaching data science to undergraduates: A comprehensive guide." 2020.
4. Garnier, Simon P. et al. "Teaching AI ethics: An interdisciplinary approach." 2021.
5. Estrellado, Ryan et al. "Data Science in Education Using R." 2021.

Козібрда С. В.,
кандидат педагогічних наук
ТНПУ ім. В. Гнатюка
vaaaav91@gmail.com
Придруга Т. П.
ТНПУ ім. В. Гнатюка

АНАЛІЗ ФРЕЙМВОРКУ ANGULAR ЯК НОВОГО ІНСТРУМЕНТУ РОЗРОБКИ ВЕБ-САЙТІВ

Сучасний світ інформаційних технологій розвивається дуже динамічно і швидко пристосовується до змін. Тому змінюються й інструменти, які використовують веб-розробники у своїй роботі. Зараз складно знайти програміста, який досі пише код на "чистому" HTML або JavaScript.

Все частіше для створення інтерактивних та динамічних веб-застосунків використовують платформи, які пропонують вирішення деяких типових проблем. Ці проблеми виникають з того, що для кожного нового застосунку програмісти створюють однотипні каркаси, компоненти, функції з нуля, витрачаючи при цьому надто багато часу. Також написання програм без використання платформ характеризується великим розміром коду, в якому за необхідності важко зорієнтуватись, визначити і виправити помилку, додати компонент тощо. Необхідно врахувати, що робота веб-застосунків на різноманітних пристроях може по-різному виводитись на екран і неоднозначно сприйматись користувачами.

Тому розробники програмного забезпечення почали використовувати новий інструмент – фреймворк. Це платформа або інфраструктура програмних рішень, яка полегшує розробку складних систем, вирішуючи типові проблеми, пов'язані з:

- великим розміром коду;
- часом, затраченим на його написання і тестування;
- адаптивністю програми під різні платформи.

Головна ціль фреймворку – дати розробнику комфортне середовище для створення програм і сайтів з великим та масштабованим функціоналом [4]. Фреймворки створені, щоб задовольняти різноманітні потреби розробників та проектів. У кожного з них є унікальні

функціональні можливості, архітектура та методи розробки. Вони можуть бути більш корисними для певних типів проектів або стилів розробки.

Спираючись на специфіку завдань, фреймворки можна поділити на три категорії:

1. Frontend framework – їх призначенням є створення користувацьких інтерфейсів з різноманітною графікою, анімацією. Приклади таких фреймворків: Svelte, Angular.
2. Backend framework – орієнтовані на виконання окремих функцій, пов'язаних із зберіганням, захистом даних, забезпеченням життєво-важливих процесів і працюють виключно на стороні сервера. Відомі бекенд фреймворки: Django, Express.js, Symfony.
3. Fullstack framework – універсальні інструменти, які використовують одну мову програмування для серверної і клієнтської частин, завдання яких полягає в обміні даними між фронтендом та бекендом. Прикладом є Meteor.

На сьогоднішній день у категорії фронтенд фреймворків одним з найкращих визнано Angular. Angular – це повноцінний frontend-фреймворк, з відкритим програмним кодом, який розробляє Google і призначенням якого є: розробка односторінкових додатків, що складаються з одної HTML сторінки з CSS і JavaScript. Метою використання даного фреймворка є: розширення браузерних застосунків на основі шаблону Модель-вид-контролер (MVC), а також спрощення їх тестування та розробки [5].

Архітектура фреймворка Angular ґрунтується на основі шаблону MVC, а саме на його компонентах:

1. модель – відповідає за формування структури, правил бізнес-логіки, зберігання даних користувача;
2. вид – відповідає за графічне відображення даних, які будуть представлені користувачу;
3. контролер – реалізує зв'язок з користувачем, перетворюючи отриману від нього інформацію в команди для попередніх двох компонентів.

Як платформа Angular включає в себе:

- компонентну основу для створення масштабованих веб-додатків;
- набір інструментів розробника, які допоможуть розробляти, створювати, тестувати і оновлювати код;
- колекцію добре інтегрованих бібліотек, які охоплюють широкий спектр функцій, включаючи маршрутизацію, керування формами, зв'язок клієнт-сервер тощо.

На сьогодні з його допомогою створено понад 400 тисяч сайтів по всьому світу. Його активно використовують Google, Microsoft, YouTube, PayPal, Upwork, Sony, General Motors та інші великі компанії. На нього варто звернути увагу тим, кому хотілося б отримати в своє розпорядження великий набір стандартних засобів і звести до мінімуму використання сторонніх бібліотек.

Попередником Angular є AngularJS, який був розроблений компанією Google ще у 2010 році і став першим повноцінним фреймворком, впровадивши нові стандарти розробки веб-сайтів [1]. З нього розпочалась історія створення і поширення інших фреймворків в області програмування. Сам же інструмент зазнав тотальної переробки і оновлення, в результаті якого у 2016 році вийшов абсолютно новий фреймворк – Angular2. Angular2 або просто Angular суттєво відрізняється від свого попередника.

По-перше, AngularJS був написаний на мові JavaScript, а Angular – на мові TypeScript. TypeScript – це розширення мови JavaScript, яке забезпечує визначення статичних типів в процесі розробки [6]. Статичні типи на відміну від динамічних, дозволяють уникнути багатьох помилок при розробці програмного забезпечення, особливо великих та складних додатків. Також ця мова підтримує підключення модулів та використання повноцінних класів, як у традиційних об'єктно-орієнтованих мовах.

По-друге, в комплект Angular додано інструмент Angular CLI, який може автоматизувати практично увесь процес розробки програмного забезпечення, при цьому

максимально спрощуючи ініціалізацію, налаштування та розробку додатків[2]. Інтерфейс командного рядка дозволяє створювати новий проект, додавати до нього функції та запускати модульні та наскрізні тести за допомогою кількох простих команд, що не тільки полегшує розробку та компіляцію веб-застосунків, а й підвищує якість коду.

По-третє, Angular не використовує концепцію "області видимості" або контролерів як його попередник, натомість як головну архітектурну концепцію він застосовує ієрархію компонентів. Відповідно до неї, додатки поділяються на незалежні логічні та функціональні компоненти. Їх легко можна замінити та роз'єднати, а також повторно використовувати в інших частинах програми. Така незалежність спрощує тестування веб-програми та гарантує безперебійну роботу всіх компонентів.

По-четверте, покращилась якість роботи платформи, його продуктивність, сумісність браузерів, у наступних версіях Angular було впроваджено підтримку розробки мобільних додатків[4]. Оновлений фреймворк виправив більшість недоліків свого попередника та впровадив зовсім нові, але набагато ефективніші інструменти для створення, тестування та підтримки веб-застосунків.

Переваги Angular над іншими фреймворками:

- реалізується як повноцінний фреймворк, який забезпечує розробника усім необхідним;
- відкритий вихідний код;
- забезпечує створення фрагментів коду для наступного використання;
- має власний інтерфейс командного рядка Angular CLI, що автоматизує виконання майже всіх процесів та спрощує створення високоякісних застосунків;
- підтримка різних елементів MVC, що дає можливість чітко розмежувати реалізацію бизнес-логіки від її представлення;;
- наявність модулів, завдяки чому можна попередньо чітко спланувати структуру додатку і слідувати їй;
- велика кількість актуальної та доступної документації щодо складу, особливостей та алгоритмом роботи із фреймворком;
- регулярно вдосконалюється та оновлюється.

Таким чином, можна дійти висновку, що Angular добре підходить для розробки як динамічних веб-додатків, так і великих та складних кросплатформних веб-застосунків. Проте і в нього є деякі недоліки: архітектура фреймворку є складною і розібратися з усіма концепціями Angular на початку буває доволі непросто. А від так і навчання може проходити складніше і довше[4].

Враховуючи всі переваги у роботі та детальною документацією від розробників, вивчення під керівництвом впевненого спеціаліста буде не лише швидким, а й продуктивним. А написання невеликих алгоритмів та компонентів в процесі навчання можна повторно використовувати в майбутньому для будь-якого розгортання: у вигляді веб-програм комп'ютерної та мобільної версій і навіть у вигляді програм для робочого столу.

Список використаних джерел

1. В. В. Босько, Л. В. Константинова, О. К. Коноплітька-Слободенюк, Д. В. Фесечко. Аналіз та дослідження фреймворку AngularJS як засобу розробки веб-сайтів. *Центральноукраїнський науковий вісник*. Технічні науки. Кропивницький, 2022. Вип 5 (36), ч І. С. 124 – 134.
2. Офіційний сайт Angular. URL: <https://angular.io/> (дата звернення: 16.03.2023)
3. Порівнюємо React, Angular і Vue — найпопулярніші бібліотеки й фреймворки у 2022 році. DOU. URL: <https://dou.ua/forums/topic/39933/> (дата звернення: 18.03.2023)
4. Adam Freeman. Pro Angular 9 : build powerful and dynamic web apps, Apress, New York, 2020. 784 pp.
5. Angular (фреймворк). Вікіпедія. URL: [https://uk.wikipedia.org/wiki/Angular_\(%D1%84%D1%80%D0%B5%D0%B9%D0%BC%D0%B2%D0%BE%D1%80%D0%BA\)](https://uk.wikipedia.org/wiki/Angular_(%D1%84%D1%80%D0%B5%D0%B9%D0%BC%D0%B2%D0%BE%D1%80%D0%BA)) (дата звернення: 18.03.2023)
6. TypeScript. Вікіпедія. URL: <https://uk.wikipedia.org/wiki/TypeScript> (дата звернення: 20.03.2023)