

## ЕФЕКТИВНІСТЬ FRONT END ФРЕЙМВОРКІВ ДЛЯ ПАНЕЛІ КЕРУВАННЯ ІНФОРМАЦІЙНИМИ ПРОЦЕСАМИ В КОМП'ЮТЕРНИХ СИСТЕМАХ

Однією з важливих частин веб- додатку є панель керування, через яку здійснюється адміністрування цього веб- додатку чи комп'ютерної системи. Для забезпечення комфортної роботи адміністратору потрібно, щоб панель керування була зручною для роботи та мала гнучку структуру та каркас для додавання чи зміни програмного продукту. Саме це забезпечує фреймворк, а широкий спектр вимог та цілей до розроблюваного продукту в області фронт енд розробки породжує безліч фреймворків і бібліотек, які мають свої недоліки та переваги. Причому універсального фреймворку в розумінні універсального інструменту немає. Саме тому постає важливе завдання зважити всі переваги і недоліки існуючих JavaScript фреймворків і визначити їх зручність при найменшому часі виконання функцій, тобто ефективність роботи веб-додатку для панелі керування інформаційними процесами в комп'ютерних системах. При створенні веб- додатків є багато факторів які впливають на ефективність їх виконання, зокрема: рівень абстракції фреймворків, спосіб взаємодії з DOM, управління залежностями та ін.

Одним з параметрів фреймворків, що впливає на ефективність є рівень абстракції при створенні об'єктів з неповністю відомим описом. Абстрагування це потужний інструмент, який покращує функціональність веб-додатків, тому деякі з існуючих фреймворків мають декілька рівнів абстракції. Але цей інструмент слід використовувати з обережністю, інакше він може призвести до дій, які неможливо відстежити. Зокрема Ember.js фреймворк за допомогою якого створюють веб-додаток використавши кілька рядків коду. Але цей фреймворк створює три маршрути, і за кожним з них закріплені контролери. Кожен з маршрутів має багато рівнів абстракції та класів об'єктів, які можна не використовувати, але вони потрібні фреймворку для роботи. Тому створений веб-додаток буде громіздким і повільно виконуватись.

На противагу Backbone.js має кілька рівнів абстракції та класів об'єктів, які містять базовий функціонал, а решта функцій реалізуються програмістом. Зокрема клас DocumentView розширює Backbone.View тобто існує лише один рівень між кодом написаним програмістом і кодом ядра фреймворку. Ефективність роботи такого веб-додатку буде вищою бо об'єм коду буде меншим і швидко виконуватись.

Також на ефективність виконання веб-додатків впливає взаємодія з DOM. Особливо коли відбувається зміна вузлів дерева на сторінці веб-додатку яка тягне за собою перерахунок розмірів або перемальовування, що забирає багато машинного часу.

Управління залежностями визначає залежності виконуваних функцій від під'єднаних зовнішніх функцій, модулів або бібліотек, які безпосередньо їх виконують. При створенні веб-додатків відбувається розподіл виконуваних функцій по модулях які їх виконують, а потім їх об'єднання в єдиний інструмент. В ідеалі ці модулі мають поводитись як чорні ящики без паразитних взаємозв'язків між собою. Вибір модулів чи бібліотек та їх об'єднання може призводити до збоїв та затримок в часі, а отже погіршувати ефективність роботи веб-додатків.

Визначення ефективності роботи веб-додатків в залежності від використаних фреймворків є багато варіантною задачею, яка потребує подальших досліджень. Особливо це є актуальним при створенні панелі керування веб-додатків