

**А.В. Гунько**

**М.В. Іванченко**

**В.Ю. Старжинський**

## **АВТОМАТИЗАЦІЯ ВЕБ РОЗРОБКИ**

Вінницький національний технічний університет;

### **Анотація**

*В роботі досліджено методи веб розробки сайтів, їх переваги та недоліки. Розкрито питання автоматизації цього процесу та представлено вирішення у вигляді нового фреймворку.*

**Ключові слова:** веб розробка, frontend розробка, backend розробка.

### **Abstract**

*The methods of web site development, their advantages and disadvantages are investigated. The automation of this process is solved and a new framework solution is presented..*

**Keyword:** web development, frontend development, backend development.

### **Вступ**

Прискорення процесу розробки web-додатків без шкоди надійності і безпеки є однією з ключових стратегічних завдань для будь-якої компанії в сфері веб-технологій.

**Метою дослідження** є підвищення ефективності технологій, створення нових методів, які дозволять значно спростити розробку web-додатків.

**Об'єктом дослідження** є процес створення вдосконаленого фреймворку на базі нових технологій.

**Предметом дослідження** є інструменти, готові рішення та фреймворки, які впливають на розробку у цілому.

**Головною задачею роботи** є розробка сучасного багатофункціонального та гнучкого фреймворку, що буде містити у собі передові технології, інструменти для побудови web-додатків будь-якої складності.

### **Результати дослідження**

Процес створення сайту поділяється на дві основні частини – frontend та backend розробка. Для кожної з них існує багато засобів, які полегшують процес створення. Усі рішення тільки надають базу для майбутнього web-додатку.

Для побудови, розробки масштабних систем використовують фреймворки. Фреймворк - це програмний продукт, який є основою для створення сайтів, але він не має готових рішень для побудови сайтів, не має рішень для виконання певних функцій. Це більш низький рівень ніж CMS. Розробники на фреймворках створюють і інтерфейсну частину, і базу даних, і алгоритми та програмні рішення проблемно орієнтованої частини і скоріше не сайту, а Веб додатку. Створюючи також його адміністративний інтерфейс.

Найбільш поширеними є Request-based і Component-based веб-фреймворки. Більшість WEB-фреймворків побудовані на архітектурі MVC. Model View Controller (MVC, «модель-представлення-контролер», «модель-вид- контролер») – схема використання декількох шаблонів

проектування, за допомогою яких модель додатки, інтерфейс і взаємодія з користувачем розділені на три окремі компоненти таким чином, щоб модифікація одного з компонентів чинила мінімальний вплив на інші. Дана схема проектування часто використовується для побудови архітектурного каркаса, коли переходять від теорії до реалізації в конкретній предметній області.

Архітектурний шаблон Модель-Вид-Контролер (MVC) поділяє програму на три частини. У тріаді до обов'язків компоненту Модель (Model) входить зберігання даних і забезпечення інтерфейсу до них. Вигляд (View) відповідальний за представлення цих даних користувачеві. Контролер (Controller) керує компонентами, отримує сигнали у вигляді реакції на дії користувача, і повідомляє про зміни компоненту Модель. Така внутрішня структура в цілому поділяє систему на самостійні частини і розподіляє відповідальність між різними компонентами.

MVC поділяє цю частину системи на три самостійні частини: введення даних, компонент обробки даних і виведення інформації. Модель, як вже було відмічено, інкапсулює ядро даних і основний функціонал з їх обробки. Також компонент Модель не залежить від процесу введення або виведення даних. Компонент виводу Вигляд може мати декілька взаємопов'язаних областей, наприклад, різні таблиці і поля форм, в яких відображається інформація. У функції Контролера входить моніторинг за подіями, що виникають в результаті дій користувача (зміна положення курсора миші, натиснення кнопки або введення даних в текстове поле). На рисунку 1 зображено принцип роботи даного патерну.

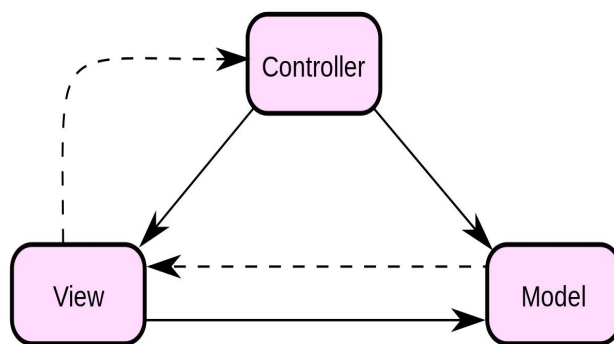


Рис.1 Схема роботи MVC

Yii2 - універсальна PHP-платформа з широким функціоналом і хорошою російськомовною спільнотою. Більш того, фреймворк підтвердив свою якість на прикладі готових торговельних систем від різних веб-студій, які задовольнили побажання замовників. Цей фреймворк має в основі ефективну систему генерації коду, набір інструментів для налагодження коду та гнучкість, що дає можливість для створення web-додатків будь-якої складності. Задля автоматизації процесу розробки є багато готових рішень, як yii2 starter kit.

Передові рішення використовують JS розробку. У JavaScript є свій перелік переваг, що дозволяє створити новий підхід до принципу роботи web-додатків. Існує багато інструментів: ReactJS (FrontEnd), Node(BackEnd), Angular(FrontEnd) та інші. Основна ідея FrontEnd розробки – це динамічний контент сайту. Підстановка окремих компонентів сторінки дозволяє зменшити час очікування, аніж завантаження повністю іншої сторінки.

Компонент даної системи виступатиме функцією, класом або бібліотекою класів в залежності від його складності. Це рішення дозволить значно спростити процес розробки та вивести його на

новий рівень, що акцентуватиме увагу саме на унікальності web-додатку. Прискорення процесу створення на пряму вплине на успішність та прибуток компанії.

## Висновки

Результати дослідження демонструють, що не існує єдиної системи чи готового рішення, що дозволяє автоматизувати процес розробки. У кожного з представлених засобів є свої недоліки та переваги. Запропоновано створення складного фреймворку, який буде вмщати в себе різні шаблони проектування, що забезпечить гнучкість, готові компоненти, що реалізовуватимуть базову логіку web-додатку (наприклад компонент форми зворотнього зв'язку, або компонент товару, що включає в себе весь процес, від публікації в магазині до купівлі його клієнтом) та інтерфейс частину, де буде реалізована не тільки адміністративна складова, а й буде представлений набір frontend-компонентів, які можна редагувати, чи за потреби модернізувати.

В процесі розробки можна спостерігати процеси, які досить часто повторюються. Новий фреймворк буде гнучким та дозволить вирішити ці проблеми.

Переваги процесу, що пропонується, стосуються лише сторони розробника, що полегшить роботу в цілому (як в написанні логіки, так і в зменшенні часових затрат), фреймворк позитивно вплине на прибуток компанії.

## СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

- Методи Web-розробки[Електронний ресурс]:[Веб-сайт]. – Електронні дані. – Режим доступу: <https://sites.google.com/site/tz5103voinovakateryna/metodi-rozrobki-web-sajtiv>://  
[HYPERLINK](#)
- Автоматизация процесса разработки[Електронний ресурс]:[Веб-сайт]. – Електронні дані. – Режим доступу:<https://cyberleninka.ru/article/n/avtomatizatsiya-protssessa-razrobotki-web-prilozheniy-na-primere-framework-sistemy-kompanii-digart>

*Гунько Артур Васильович*— студент групи ІАКІТ-176, факультет комп'ютерних систем і автоматики, Вінницький національний технічний університет м.Вінниця,,e-mail: fksa.lakit.gav@gmail.com

*Іванченко Максим Валерійович* — студент групи ІАКІТ-176, факультет комп'ютерних систем і автоматики, Вінницький національний технічний університет, м.Вінниця,e-mail: fksa.lakit.imv@gmail.com

*Старжинський Валерій Юрійович*— студент групи ІАКІТ-176, факультет комп'ютерних систем і автоматики, Вінницький національний технічний університет, м.Вінниця, e-mail: fksa.lakitsvyu@gmail.com

Науковий керівник: *Васюра Анатолій Степанович* — професор кафедри автоматики і інформаційно-вимірювальної техніки, Вінницький національний технічний університет, м.Вінниця

*Gunko Arthur V.* — Department of Computer System and Automation, Vinnytsia National Technical University, Vinnytsia, e-mail: fksa.lakit.gav@gmail.com

*Ivanchenko Maxim V.* — Department of Computer System and Automation, Vinnytsia National Technical University, Vinnytsia, e-mail: fksa.lakit.imv@gmail.com

*StarzhinskiiValeriiYu.*—Department of Computer System and Automation, Vinnytsia National Technical University, Vinnytsia,, e-mail: fksa.lakitsvyu@gmail.com

Supervisor: *Vasyura Anatoliy S.* — Professor of Automation and Information and Measurement Engineering Department, Vinnytsia National Technical University, Vinnytsia.