

РОЗРОБКА ВЕБ-ДОДАТКІВ ЗА ДОПОМОГОЮ ФРЕЙМВОРКА CODEIGNITER

Codeigniter – це фреймворк, написаний на мові програмування PHP для розробки повноцінних веб-додатків, компанією EllisLab. Фреймворк – це готовий каркас, на основі якого буде будуватись архітектура додатку. До нього включають найбільш частіше використовувані бібліотеки, через це відпадає необхідність детального продумування структури додатку, організації роботи з модулями, класами і т. д.

Головна ціль фреймворка CodeIgniter заключається в тому, щоб дозволити розробляти проекти набагато швидше, ніж якщо б ви писали весь код самі, представляючи величезний набір готових бібліотек для вирішення задач, які найчастіше зустрічаються, а також простий інтерфейс і логічну структуру для доступу до цих бібліотек. CodeIgniter дозволяє вам повністю зосередитись на проекті, мінімізуючи кількість коду, який вам потрібно самостійно писати для кожної конкретної задачі.

Далі буде перераховано декілька переваг CodeIgniter над іншими фреймворками:

1. Він розповсюджується під відкритою ліцензією Apache/BSD-style.
2. Базова платформа потребує тільки декількох бібліотек. Це дуже сильно вирізняє його посеред інших фреймворків, які потребують значно більших ресурсів. Додаткові бібліотеки завантажуються динамічно при запиті, в залежності від потреб для конкретного процесу, саме тому базова платформа дуже швидка.
3. Використовує підхід Model-View-Controller, який дозволяє значне розділення між логікою та представленням. Це особливо корисно для проектів, в яких дизайнери працюють над файлами шаблонів, так як місткість коду в цих файлах мінімізована.
4. CodeIgniter має широкий асортимент бібліотек, які дозволяють виконувати часто затребувані задачі у розробці веб-додатків, такі як доступ до бази даних, відправлення пошти, валідація даних форм, обслуговування сесій, маніпуляції зображеннями, робота з XML-RPC і багато інших.
5. CodeIgniter дуже добре документований. В його документації описані абсолютно усі деталі роботи функцій, бібліотек та плагінів. Також досить приємним доповненням для вітчизняних розробників є наявність документації на російській мові. З усіх фреймворків, саме CodeIgniter має найкращу документацію на російській мові.

CodeIgniter оснований на шаблоні розробки Model-View-Controller (Модель-Відображення-Контроллер). MVC – це підхід до розробки додатків, який дозволяє відділити логіку від представлення. На практиці це дозволяє веб-сторінкам містити мінімальну кількість програмного коду в представленні. Модель відображає структуру даних. Зазвичай класи моделей будуть містити функції, які допомагають отримати, додати та оновити інформацію у базі даних. Відображення (представлення) – це інформація, яка буде відправлена користувачу. Представлення – це звичайні веб-сторінки, але у CodeIgniter представлення також можуть бути фрагментами сторінок. Контролер працює як посередник між моделлю, представленням та іншими ресурсами, які необхідні для обробки HTTP-запиту та генерації веб-сторінок. CodeIgniter має досить вільний підхід до MVC, оскільки використання моделей не обов'язкове. Якщо не потрібне додаткове розділення, чи помітно, що обслуговування моделей значно важче, ніж очікується, то можна ігнорувати їх та будувати додатки мінімалістичними, використовуючи лише контролери та представлення CodeIgniter. Також CodeIgniter має легко розширювану систему: створення бібліотек – особисті бібліотеки (класи) зберігаються в папці “application/libraries” для розділення ресурсів фреймворка між локальними та глобальними. “Хелпери” – колекція функцій, яка допомагає у вирішенні деяких задач. Плагіни – практично ідентичні “хелперам”. Головна їх відмінність у тому, що плагін – це одна функція.

Процес виконання додатку CodeIgniter наступний:

- index.php працює як фронт-контролер, ініціалізуючи основні ресурси, потрібні для запуску CodeIgniter.
- Роутер перевіряє HTTP-запит, щоб визначити, що з ним потрібно зробити.
- Якщо існує файл кеша, він буде відправлений в браузер, в обхід нормальному виконанню додатка.
- Перед тим, як буде завантажено контролер додатку, HTTP-запит та будь-які дані, передані користувачем фільтруються для забезпечення безпеки.
- Контролер завантажує моделі, бібліотеки ядра, помічників, та будь-які інші ресурси, потрібні для обробки конкретного запита.

Фінальне відображення підготовлюється та відправляється у браузер. Якщо увімкнено кешування, відображення спочатку кешується, таким чином воно може бути використане при послідовних запитах.