

Бакасов В.М., магістрант, Попівций В.І., доцент, науковий керівник
**ДОСЛІДЖЕННЯ ОСОБЛИВОСТЕЙ ВИКОРИСТАННЯ ШАБЛОНІВ
ПРОЕКТУВАННЯ В СУЧАСНИХ РНР-ФРЕЙМВОРКАХ**

Запорізька державна інженерна академія, кафедра ПЗАС

Жорстка конкуренція в сфері програмного забезпечення вимагає постійного підвищення якості програмних продуктів і змушує їх авторів якомога частіше випускати нові версії своїх програм і доповнення до них. При створенні нових інформаційних систем, а тим більше при випуску їх оновлених версій, часто трапляється нагода використання створених раніше рішень. Ці рішення можуть істотно спростити і прискорити процес розробки програмного забезпечення.

Розробка і розвиток програмного забезпечення були б значно простіше, якби існували правила, що дозволяють створити бездоганну структуру - гнучку, ефективну, зрозумілу та легко масштабовану. Зокрема, при пошуку структури, в цілому можна скористатися рекомендаціями, створеними з урахуванням досвіду розробників, які зустрічалися зі схожою проблемою раніше. Багато з таких рекомендованих рішень вже знайдені, систематизовані, отримали стандартні імена і готові до застосування – мова іде про шаблони проектування.

Швидке зростання числа користувачів Інтернету протягом останніх двох десятиліть, розширення можливостей та доступності розробки програмного забезпечення та великий попит на такі програми – все це сприяло значному збільшенню кількості людей, які працюють над проектуванням веб-додатків та засобів їх розробки – так званих фреймворків. РНР фреймворки займають чільне місце в розробці веб-додатків і при їх використанні важливим є усвідомлення втілення в цих фреймворках основних принципів розробки програмного забезпечення і зокрема шаблонів проектування. Їх опису, а також демонстрації їх використання при проектуванні веб-додатків з використанням сучасних РНР фреймворків, і присвячена дана робота.

Об'єктом дослідження даної роботи є сучасні РНР фреймворки.

Предмет дослідження – шаблони проектування, особливості їхнього використання в РНР-фреймворках.

Мета роботи – на прикладі тестових веб-додатків дослідити вплив використання шаблонів проектування на якість створюваного додатку та можливості його розширюваності та розвитку.

В роботі наведені загальні відомості про шаблони проектування, розглянуті особливості використання шаблонів проектування в таких фреймворках, як Laravel та Zend Framework.

Також розглядаються технології розробки веб-додатків з використанням фреймворків.

Далі в роботі наведено огляд використання архітектурного патерну MVC у фреймворку Laravel, розглянуто вплив даного патерну на загальну структуру веб-додатку.

Слід відзначити, що багато популярних шаблонів проектування повинні розглядатися по-різному, коли мова заходить про веб-додатки. З усіх доступних шаблонів проектування в роботі були розглянуті: Abstract Factory, Application Controller, Active Record, Adapter, Builder, Command, Composite, Decorator, Dependency Injection, Domain Model, Factory Method, Front Controller, Iterator, Model-View-Controller, Observer, Proxy, Registry, Singleton, Specification, State, Strategy, Visitor.

Для кожного шаблону вказано прототип мікроархітектури, який розробник може скопіювати і адаптувати до свого власного дизайну для вирішення повторюваних проблем, описаних патерном проектування. Мікроархітектура - це набір програмних компонент (класи, методи і т.д.) і їх взаємозв'язки. Розробники використовують патерни

проектування, привносячи в свій дизайн цю мікроархітектуру, в результаті їх дизайн буде мати архітектуру, схожу з описаною в обраному патерні.

Додатковою перевагою патернів є спрощення комунікації між розробниками, тому що дозволяє їм використовувати однаковий словник термінів при обговоренні архітектури додатку.

При дослідженні PHP фреймворків та розробці веб-додатків були виявлені наступні особливості:

- Використання патернів проектування дозволяє прискорити процес розробки ПО, завдяки використанню випробуваної, доведеної парадигми розробки.
- Побудова правильного дизайну системи вимагає розгляду великої кількості питань, деякі з яких можуть виникнути тільки на етапі впровадження. Повторне використання патернів проектування при розробці дизайну допомагає уникнути невірних дій, які ведуть до виникнення проблем на пізніх стадіях реалізації, а також підвищує прозорість і читаність коду для інших розробників і архітекторів, знайомих з патернами.

Негативні моменти застосування шаблонів наступні:

- Для досягнення гнучкості патерни проектування часто вносять в дизайн додаткові непрямі рівні, які в деяких випадках можуть ускладнити дизайн і погіршити продуктивність програми.
- За визначенням, патерн повинен програмуватися заново для кожного додатка. Тому деякі автори вбачають, що це крок назад по відношенню до повторного використання коду, що надається компонентами. Тому дослідники працюють над перетворенням патернів в компоненти.

Висновки :

1. При створенні веб-додатків так чи інакше ми стикаємося із задачею повторного використання та структуризацією коду.
2. Гарним рішенням цієї задачі є використання шаблонів проектування. Можливості шаблонів залежать від фреймворка який буде обраний для написання веб-додатка.
3. В роботі описуються результати, що стосуються використання шаблонів проектування в веб-додатках, що розробляють на основі PHP фреймворків.
4. Досліджено документацію та вихідний код декількох фреймворків, проаналізовано виявлені в них шаблони проектування.
5. Результати та висновки роботи можуть бути корисними при плануванні та розробці нових проектів, оскільки існуючий досвід може бути врахований. В роботі також наведено інформацію про те, які шаблони проектування не використовуються, оскільки вони можуть бути штучними або важкими для використання в реальних проектах.

Література

1. Design Patterns: Elements of Reusable Object-Oriented Software / [Gamma Erich, Richard Helm, Ralph Johnson and John Vlissides]. – Addison-Wesley, 1995
2. Fowler Martin. Patterns of Enterprise Application Architecture / Fowler Martin. – Addison-Wesley, 2002.
3. Kelt Dockins Design Patterns in PHP and Laravel. – Apress, 2017