

ДОСЛІДЖЕННЯ ЗАСТОСУВАННЯ АРХІТЕКТУРНИХ ШАБЛОНІВ НА ПРИКЛАДІ ІНФОРМАЦІЙНОЇ СИСТЕМИ НАВЧАЛЬНОГО ЗАКЛАДУ

Запорізька державна інженерна академія, кафедра ПЗАС

В наш час збільшуються вимоги до проектування програмного забезпечення (ПЗ) інформаційних систем, головною складовою якого є створення архітектури програмного застосунку. Правильно підібрана архітектура збільшує гнучкість системи, полегшує її модифікацію та супровід при неминучій зміні вимог в процесі її функціонування. Архітектура – це базова організація системи, втілена в її компонентах, їх відношеннях між собою та оточенням, а також принципи, що визначають проектування та розвиток системи [1]

В програмній інженерії для проектування ПЗ запропоновані так звані архітектурні шаблони – зразки ефективного вирішення типових задач проектування архітектури. Вперше шаблони проектування були запропоновані «бандою чотирьох» (Gang of Four, GoF) на чолі з Еріхом Гамма для детального проектування підсистем. Пізніше типові шаблони (патерни, зразки) були запропоновані також для вирішення загальних типових задач всіх інших етапів життєвого циклу ПЗ, в тому числі для етапу створення архітектури.

Метою даної роботи є дослідження критеріїв вибору архітектурних шаблонів та їх вплив на подальшу структуризацію та реалізацію системи на тестовому прикладі інформаційної системи навчального закладу.

З чим пов'язана архітектура ПЗ: архітектура визначає структуру, архітектура визначає поведінку, архітектура концентрується на значущих елементах, архітектура врівноважує потреби зацікавлених осіб, архітектура втілює рішення на основі логічного обґрунтування, архітектура може відповідати деякому архітектурному стилю, на архітектуру впливає її оточення, архітектура впливає на структуру колективу, архітектура представлена в кожній системі, архітектура має особливу область застосування [2].

Архітектурний шаблон – це набір принципів, високорівнева схема, що забезпечує абстрактну інфраструктуру для сімейства систем. Архітектурний стиль покращує секціонування і сприяє повторному використанню дизайну завдяки забезпеченню рішень проблем, що часто зустрічаються [3]. Архітектурні стилі і шаблони можна розглядати як набір принципів, що формують застосунок.

Основні архітектурні стилі:

- Клієнт/сервер: система розділяється на два застосування, де клієнт виконує запити до сервера. У багатьох випадках в ролі сервера виступає база даних, а логіка застосунка представлена процедурами зберігання.
- Компонентна архітектура: дизайн застосунку розкладається на функціональні або логічні компоненти з можливістю повторного використання, надаючи інтерфейси зв'язку, що ретельно пропрацювали.
- Багат шарова архітектура: функціональні області застосунка розділяються на багат шарові групи (рівні).
- Шина повідомлень: система може приймати і відправляти повідомлення по каналам зв'язку, тому застосунки дістають можливість взаємодіяти, не маючи в розпорядженні конкретних відомостей один про одного.
- N- рівнева / 3-рівнева: функціональність виділяється в окремі сегменти, багато в чому аналогічно багат шаровому стилю, але в даному випадку сегменти фізично розташовуються на різних комп'ютерах.

- Об'єктно-орієнтована: парадигма проектування, заснована на розподілі відповідальності застосунка або системи між окремими багаторазово використовуваними і самостійними об'єктами, що містять дані і поведінку.

- Сервісно-орієнтована архітектура: описує застосунки, що надають і споживають функціональність у вигляді сервісів за допомогою контрактів і повідомлень.

Для створення тестової інформаційної системи, як один з основних варіантів, була обрана багатошарова архітектура, бо вона розподіляє модулі обробки даних, які виконуються на одному або декількох окремих серверах. На першому рівні знаходиться сервер бази даних. На другому – обробка запитів від клієнта. На третьому – інтерфейс користувача. Крім того, різні сервери додатків можуть взаємодіяти між собою для точнішого розділення системи на функціональні блоки, що виконують певні ролі [4].

У багаторівневій архітектурі "тонкий" клієнт не переобтяжений функціями обробки даних, а виконує свою основну роль системи представлення інформації, що поступає з сервера додатків. Це зменшує об'єм даних, що передаються між клієнтом і сервером додатків. До основних переваг багаторівневого архітектурного стилю можна віднести зручність підтримки, масштабованість, гнучкість.

Також запропонована об'єктно-орієнтована реалізація архітектури інформаційної системи на основі шаблону Model-View-Controller та досліджено її особливості. Окремо розглянуто питання створення веб-інтерфейсу інформаційної системи.

В роботі використані також деякі ідеї, запропоновані в Domain Driven Design (DDD)[5]. Зокрема, що архітектура програмної системи повинна ґрунтуватися на моделі предметної області.

Досліджено також архітектури, пов'язані з застосуванням конкретних фреймворків для реалізації інформаційної системи. Фреймворк формує каркас застосунку, який треба враховувати при проектуванні загальної архітектури. Наприклад, для фреймворка ASP.NET MVC проаналізовані шари Presentation Subsystem, View Model, Business Logic Subsystem та шаблони проектування, пов'язані з цими шарами. Розглянуто особливості реалізації багатошарової архітектури на основі фреймворка Yii мови PHP. Виділено три шари: Core layer – шар ядра, Shared layer – шар модулів, що реалізують загальний функціонал системи, та Application layer – шар застосунку.

Висновки:

- Досліджено вплив архітектурних шаблонів на загальні характеристики та реалізацію інформаційної системи навчального закладу.
- Запропоновано декілька практичних реалізацій системи та досліджено їх особливості.

Література

1. Басс Л., Клементс П., Кацман Р. Архитектура программного обеспечения на практике. – СПб.: Питер, 2006 – 575 с.
2. Що таке архітектура програмного забезпечення. [Електронний ресурс]. – Режим доступу: <http://easy-code.com.ua/2012/07/shho-take-arxitektura-programnogo-zabezpechennya-rizne-programuvannya-statti/>
3. Фаулер М. Архитектура корпоративных программных приложений (пер. с англ.). – К.; М.; СПб.: Вильямс, 2004. – 544 с.
4. Ехлаков Ю.П. Принципы построения Web-ориентированной ГИС промышленного предприятия / Ю.П. Ехлаков, О.И. Жуковский, Н.Б. Рыбалов // Изв. Том. политех. ун-та. – 2006. – Т.309, № 7. – С. 146–152.
5. Эванс Э. Предметно-ориентированное проектирование (DDD): структуризация сложных программных систем. – М.: Вильямс, 2011. – 448 с.