

УДК 681.1

О.В. Токаренко, студент гр. ПА-91мп, к.т.н., доц. Богомазов С.А.
КПІ ім. Ігоря Сікорського

СИСТЕМА ЗБОРУ ЕКСПЕРИМЕНТАЛЬНИХ ДАНИХ НА ОСНОВІ JAVA-ФРЕЙМВОРКІВ

Анотація. У статті проведений опис та аналіз доцільності використання Java-фреймворків в програмних системах збору та обробки експериментальних даних. Визначені основні переваги використання даного підходу та перспективи його застосування. Наведено приклад реалізації системи на основі фреймворків Spring та Hibernate сумісно з мережевим протоколом CoAP. Проведений аналіз буде корисним для оптимізації програмного забезпечення систем які виконують збір та обробку експериментальних даних.

Ключові слова: Java, фреймворк, збір та обробка даних, база даних.

ПОСТАНОВКА ПРОБЛЕМИ

В системах автоматизованих вимірювань, контролю, технічного моніторингу важливу роль відіграють процеси збору, зберігання та аналізу експериментальних даних від різноманітних джерел, а саме проводових і безпроводових вимірювальних і сенсорних засобів. При створенні таких систем виникає необхідність розробки програмного забезпечення для зберігання, отримання та обробки експериментальних даних. Для цього проведено аналіз існуючих рішень для автоматизації даних процесів за допомогою сучасних Java-фреймворків.

ОСНОВНІ ПОЛОЖЕННЯ

Бібліотеки-фреймворки надають розробнику деяку постійну частину програмного забезпечення, до якої необхідно додати код користувача. Користувача частина вбудовується в програмне забезпечення фреймворку. Фреймворк визначає загальну архітектуру програмного продукту та реалізує взаємодію між його компонентами.

На рис.1 зображені основні шари програмного забезпечення типового Web-орієнтованого програмного додатку. У випадку Web-орієнтованих програмних систем збору та обробки даних, що реалізовані за допомогою мови програмування Java, є доцільним застосування сучасних фреймворків в проміжних шарах такої системи, а саме:

1. При передачі вимірювальних даних шляхом запитів до серверу.
2. При зберіганні та обробці вимірювальної інформації.

Саме на цих рівнях програмного забезпечення заміщення процесу ручної розробки програмного забезпечення розробкою з використанням фреймворків буде найбільш ефективним. На допомогу приходять сучасні бібліотеки-фреймворки, такі як Spring Framework, Hibernate, React. На даний момент це одні з найвідоміших представників інструментів для автоматизації створення та покращення роботи програмних систем що виконують збір, передачу, обробку та аналіз вимірювальних даних.

Spring Framework дозволяє суттєво спростити реалізацію Web-орієнтованих систем шляхом автоматизації базових процесів [1]. В його основі лежить реалізація принципу зменшення кількості залежностей між частинами програмної системи шляхом використання технології інжекції залежностей

(Dependency Injection). Згідно з принципом інверсії керування (Inversion of Control, IoC), ядро фреймворку самостійно створює об'єкти конкретних класів та передає посилання на них (інjektує) в залежні від відповідних абстракцій об'єкти. Тим самим зберігається залежність лише від абстракцій, а не від конкретних класів, що значно спрощує модифікацію системи.

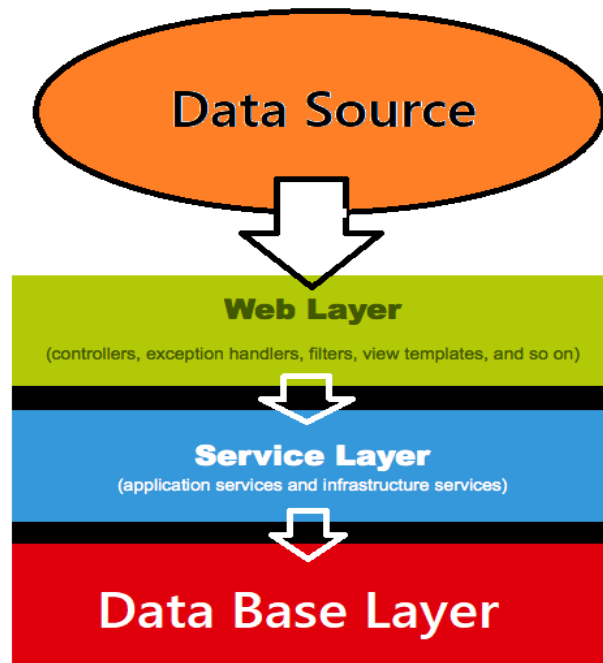


Рисунок 1. Шари типового Web-додатку для збору та обробки даних

За допомогою анотацій або конфігураційних файлів фреймворк Spring дозволяє співставити URI відправленого на сервер запиту відповідному контролеру для обробки та формування відповіді. Цей фреймворк дозволяє пришвидшити створення Web-орієнтованих програмних систем з архітектурою MVC (Model-View-Controller) та зменшує вимоги до кваліфікації розробників програмного забезпечення. Якщо необхідна модифікація системи для певних користувацьких завдань, Spring дозволяє легко вносити корективи в конфігурацію системи.

Важливою частиною системи збору та обробки експериментальних даних є збереження отриманої інформації в базі даних. Найбільш поширеним рішенням, яке дозволяє зменшити час на проектування та розробку шару зберігання даних є використання ORM-фреймворків. ORM – це Object Relationship Mapping, тобто співставлення об'єктних відношень з реляційними таблицями баз даних.

Одним з представників таких фреймворків є Hibernate [2]. Основним завданням цього фреймворку є автоматичне перетворення об'єктів, що використовуються в об'єктно-орієнтованій програмі в записи бази даних. Це дозволяє скоротити час розробки програмного забезпечення та пришвидшити процес зберігання та отримання інформації. Порівняно з ручною реалізацією за допомогою драйвера JDBC операцій CRUD (Create, Read, Update, Delete), даний фреймворк дозволяє перейти від низькорівневого програмування до високорівневого, використовуючи вже готові рішення для CRUD операцій. Для

цього розробнику достатньо лише сконфігурувати цей процес шляхом використання конфігураційних XML-файлів або анотацій. Анотації використовуються безпосередньо в класах, які описують структуру об'єктів, що створюються на базі інформації, отриманої з зовнішніх систем. Клієнти при цьому який можуть бути реалізовані як на основі персональних комп'ютерів або мобільних пристроїв, так і на основі інтелектуальних мережевих сенсорів, що надсилають потік вимірювальних даних. Для оптимізації роботи фреймворк Hibernate використовує кешування, скорочуючи часові витрати на обробку запитів.

На базі фреймворків Spring та Hibernate розроблено програмне забезпечення серверної частини демонстраційної Інтернет-системи збору експериментальних даних. За допомогою датчика DHT11, що підключений до одноплатного комп'ютера RaspberryPi, вимірюється температура та вологість в приміщенні для моніторингу його кліматичних параметрів. Для передачі результатів вимірювань через мережу Інтернет до серверної частини системи використано прикладний мережевий протокол CoAP (Constrained Application Protocol), що призначений для обміну інформацією для пристроїв з обмеженими ресурсами [3]. Це дозволило забезпечити передачу повідомлень малого розміру та зменшити енерговитрати автономних частин системи. Протокол CoAP відповідає структурі REST (Representational State Transfer) та забезпечує логічну сумісність запитів з запитами протоколу HTTP при значно меншій кількості службової інформації [4]. Технологія REST-сервісів підтримується сучасними версіями фреймворку Spring. Серверна частина системи за допомогою CoAP-проксі перетворює запити CoAP в запити HTTP, які обробляються серверним програмним забезпеченням.

ВИСНОВОК

Таким чином, використання сучасних Java-фреймворків дозволило оптимізувати процес розробки програмного забезпечення системи збору даних та полегшити внесення змін до програмного забезпечення системи. При цьому кожний елемент системи може бути модифікований незалежно від інших частин. Використання спеціалізованого прикладного протоколу CoAP дозволило зменшити мережевий трафік та збільшити час робочого циклу автономних мережевих вимірювальних пристроїв.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

- [1] Spring Framework . – Режим доступу: <https://spring.io> – 21.11.2019 р.
- [2] Hibernate ORM. – Режим доступу: <https://hibernate.org/> – 21.11.2019 р.
- [3] Гойхман В. Аналитический обзор протоколов Интернета вещей / В. Гойхман, А. Савельева // Технологии и средства связи. – 2016. – № 4. – С. 32–37.
- [4] ITU-T/ The Constrained Application Protocol (CoAP) / RFC 7252 – Proposed Standard. 2014.

Наук. керівник – к.т.н., доц. Богомазов С.А.