

3. Atroschenko V. I., Kargin S. I. (1980), Technology of nitric acid / L.: Chemistry.
 4. *Production of nitric acid in aggregates of large unit capacity* (1985) / ed. by V. M. Olevsky. M.: Chemistry.
 5. *Production of ammonium nitrate in aggregates of large unit capacity* (1990) / ed. by V. M. Olevsky, M.: Chemistry.
 6. Kucheryavy V. I., Lebedev V. V. (1970) Synthesis and application of urea / M.: Chemistry.
 7. Gorlovsky D. M., Altshuler L. N., Kucheryavy V. I. (1981), Technology of urea / L.: Chemistry.
 8. Atroschenko V. I. (1978), Methods of calculations based on the technology of bound nitrogen / Kiev: Vysh. SHKOLA.
 9. Demidenko I. M. (2001), Ammonia. Questions of technology / Donetsk: GIK "New seal": LLC "Swan".
 10. Chemical technology of inorganic substances: textbook. manual: in 2 vols. / ed. by T. G. Akhmetov (2002), M.: Higher school, Vol. 1, 2.
 11. Azotchik's Handbook: in 2 vols. (1987) / ed. by E. Ya. Melnikov, M.: Chemistry, Vol. 1.
-

УДК 004.415.2

**КОВАЛЮК Д. О. к.т.н., доц., КАРМАЗІН І.М, магістрант
Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського»
КОВАЛЮК О. О. к.т.н., доц.
Вінницький національний технічний університет**

ВИКОРИСТАННЯ ВЕБ-СЕРВІСІВ В СИСТЕМАХ КЕРУВАННЯ ТЕХНОЛОГІЧНИМИ ПРОЦЕСАМИ

В статті розглянуто проблеми функціонування сучасних систем керування технологічними процесами, зокрема інтеграції різних компонентів таких систем та обміну даними між ними. Проаналізовано використання веб-сервісів в системах керування, контролю доступу, SCADA-системах та в «розумних будинках». Обґрунтовано основні переваги використання веб-сервісів на виробництві та функції, які вони можуть виконувати. Проведено аналіз засобів (платформ) для розробки програмного забезпечення, що реалізує технологію веб-сервісів. Спроектовано архітектуру веб-сервісу для моніторингу температурного режиму печі, розроблено спеціальне програмне забезпечення на базі ASP.NET Core 3, проведено тестування роботи веб-сервісу на прикладі вимірювання температури в кімнаті з використанням апаратно-обчислювальної платформи Arduino.

Ключові слова: автоматизація, системи, керування, веб-сервіси, виробництво, ASP.NET Core.

DOI: 10.20535/2617-9741.4.2020.219783

© Ковалюк Д. О., Кармазін І. М., Ковалюк О.О., 2020.

Постановка проблеми. Для оптимального керування технологічними об'єктами та процесами як правило проектуються системи керування, які мають достатньо складну структуру і зв'язки між своїми частинами. З однієї сторони, це дозволяє виконувати функції автоматичного контролю та регулювання, технологічної та аварійної сигналізації, реєстрації стану процесу, візуального відображення для оператора. З іншої сторони, це приводить до суттєвого ускладнення систем, що пов'язане з підвищенням інтенсивності процесів, швидкостей, вимог до точності та якості. Якщо говорити про структуру сучасних систем керування, то всі вони є багаторівневими, як показано на рисунку 1. Зазначені рівні покриваються однією або кількома системами.



Рис. 1 – Структура багаторівневої системи керування

Як видно з рисунку 1, система поділена на декілька рівнів, що включають технічне та програмне забезпечення.

- PLANT – рівень виробництва, який містить всі апарати що зосереджені на виробництві.
- Field Instruments – поєднує усі вимірювальні та виконавчі пристрої, які під'єднані до апаратів чи трубопроводів.
- PLC – рівень програмованих логічних контролерів, призначених для збору/передачі інформації від польових пристроїв та реалізації алгоритмів керування.
- SCADA/HMI – рівень збору та візуалізації інформації про перебіг технологічного процесу з можливістю операторського управління.
- MIS/MES – рівень розв'язання задач синхронізації, координації, аналізу та оптимізації випуску продукції в рамках дільниць та цехів.
- ERP – глобальний рівень розподілення та планування ресурсів по всьому виробництву.

Найбільшими проблемами при функціонуванні складних систем керування є:

1. Відсутність уніфікації технічних засобів. Вся система керування містить ряд підсистем і спеціальних систем, виробництво яких виконується окремими організаціями на своїй елементно-конструкторській базі. Різноманітність технічних засобів призводить до збільшення витрат на їх обслуговування.

2. Ієрархічний розподіл функцій управління і обробки даних призводять до зайвих апаратних витрат і зниження надійності системи, виникає необхідність узгодження даних та інтерфейсів їх взаємодії між різними рівнями.

3. Наявність різного програмного забезпечення і сховищ даних, що ускладнює підтримання їх актуальності і надлишковість.

Зазначені недоліки існуючих АСУ ТП призводять до значного збільшення обсягу технічних засобів, внаслідок чого зростають витрати на запуск та експлуатацію систем, збільшуються терміни виведення виробництва на проектну потужність.

Аналіз попередніх досліджень.

Для вирішення задач інтеграції і узгодження взаємодії між частинами системи на сьогодні одним з найбільш поширених є підхід використання веб-сервісів. Він передбачає обмін повідомленнями спеціального (універсального) формату між частинами системи і обробку (виконання певних дій) при отриманні повідомлень. При чому, завдяки чітко визначеній специфікації та стандартам, веб-сервіси забезпечують взаємодію програмних систем незалежно від платформи, а використання HTTP протоколу може забезпечити взаємодію програмних систем через зовнішні інтернет ресурси [1-3].

Веб-сервіси уже знайшли широке застосування в різних системах: SCADA [4, 5], контролю доступу [6], керуванням інженерними комунікаціями будинків [7, 8], аналізу даних [9]. Крім того на сучасному ринку розробки програмного забезпечення досить популярною є мікросервісна архітектура, коли один монолітний програмний засіб краще представити у вигляді «мікросервісів» – невеликих вузькоспеціалізованих програмних компонентів, що взаємодіють між собою. Це дозволяє зробити проект більш гнучким до внесення змін та технічної підтримки.

Проектування або модернізація вже існуючих систем керування технологічними процесами на основі веб-сервісів – дозволить зменшити кількість апаратних засобів, зробити централізовані точки доступу до даних і підвищити надійність виробництва. Все це підтверджується великою кількістю платформ і фреймворків для реалізації веб-сервісів на мовах програмування PHP, Java, Python, C# [10].

До переваг веб-сервісів можна також віднести можливість побудови автоматичних систем, оскільки обмін повідомленнями можна налаштувати за графіком.

Ще однією суттєвою перевагою є універсальність підходу сервісної архітектури, що дозволяє використовувати розроблені системи та їх складові для різних технологічних процесів, без додаткових змін.

Метою статті є підвищення ефективності систем керування за рахунок використання веб-сервісів, що дозволить скоротити кількість проміжних вузлів, зменшити вартість спроектованої системи та підвищити її надійність.

Виклад основного матеріалу.

Спершу розглянемо роль веб-сервісу в системах керування і відповідно функції, які він повинен виконувати. Місце веб-сервісу в архітектурі системи керування показано на рисунку 2. Найбільш доцільним є розміщення між рівнями PLC – контролерів і систем рівня (SCADA), що реалізують людино-машинний інтерфейс. Це дасть змогу отримати поточну інформацію про стан технологічного процесу і реалізувати наступні функції:

- Збір даних від контролерів;
- Збереження даних до БД;
- Надання інформації оператору та її візуалізація в реальному масштабі часу;
- Моніторинг та сигналізація параметрів технологічного процесу;

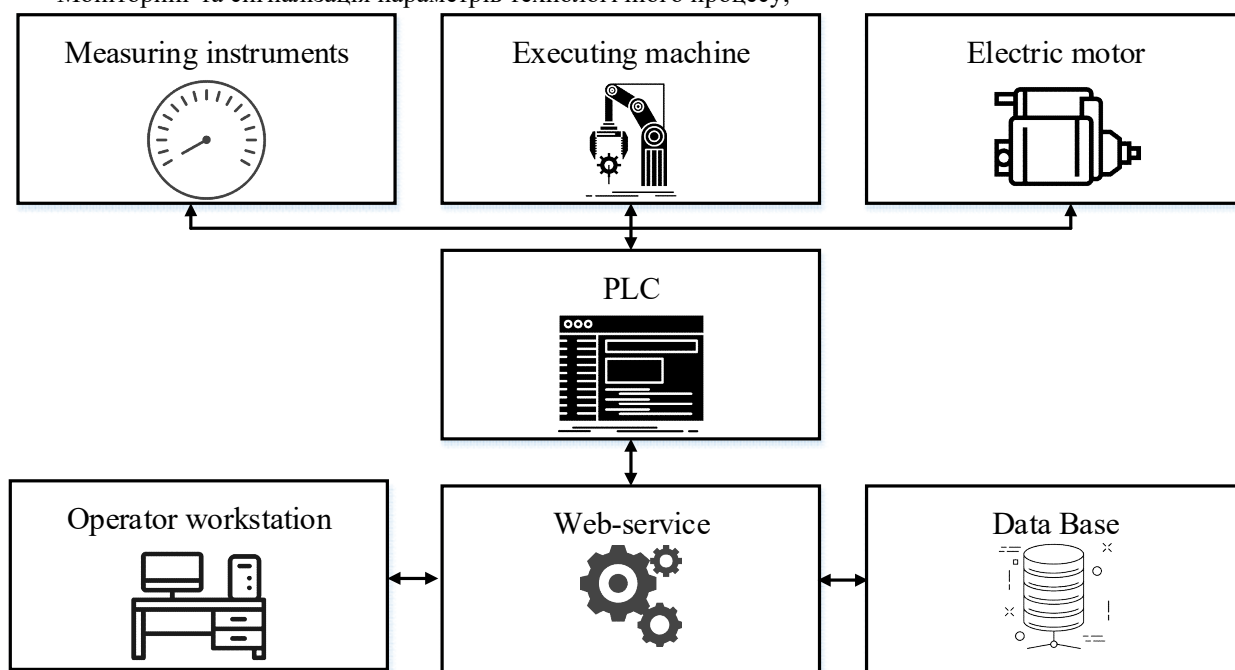


Рис. 2 – Архітектура комп’ютерно-інтегрованої системи на базі веб-сервісу

Аналіз засобів розробки веб-сервісів.

Визначивши мінімально необхідний набір функцій системи, який необхідно реалізувати, можемо перейти до процесу розробки. Існує багато мов програмування та платформ, які дозволяють реалізовувати систему з веб-сервісом. Також розроблено значну кількість фреймворків, які прискорюють розробку веб-сервісів. Найбільш популярні з них: Spring Boot на мові програмування Java, Flask та Django на Python, ASP.Net Core на C# та Laravel на PHP.

Для реалізації веб-сервісу системи керування пропонується використати фреймворк ASP.NET Core 3 від компанії Microsoft. До переваг ASP.NET можна віднести:

- *Відкритість.* Платформа .NET є розробкою і власністю компанії Microsoft, але була і залишається безкоштовною у використанні. У 2014 році для подальшої популяризації платформи була заснована незалежна організація .NET Foundation. Програмний код платформи був розміщений в публічному репозиторії GitHub і тепер поширюється як продукт з відкритим вихідним кодом.
- *Кросплатформеність.* Можливість розробити програмний продукт, яке зможе працювати на різних операційних системах: Windows, Linux, macOS.

- *Швидкість розробки.* Велика кількість готових до використання рішень вже розроблені працівниками Microsoft, іншими організаціями чи окремими розробниками, що суттєво пришвидшує і спрощує процес розробки.

Архітектура ASP.NET Core 3 з основними можливостями наведена на рисунку 3. Серед найбільш важливих слід відзначити вбудований механізм авторизації, підтримка баз даних, розсилки повідомлень засобами електронної пошти та SMS, ведення журналу, інтеграція з іншими веб-додатками.

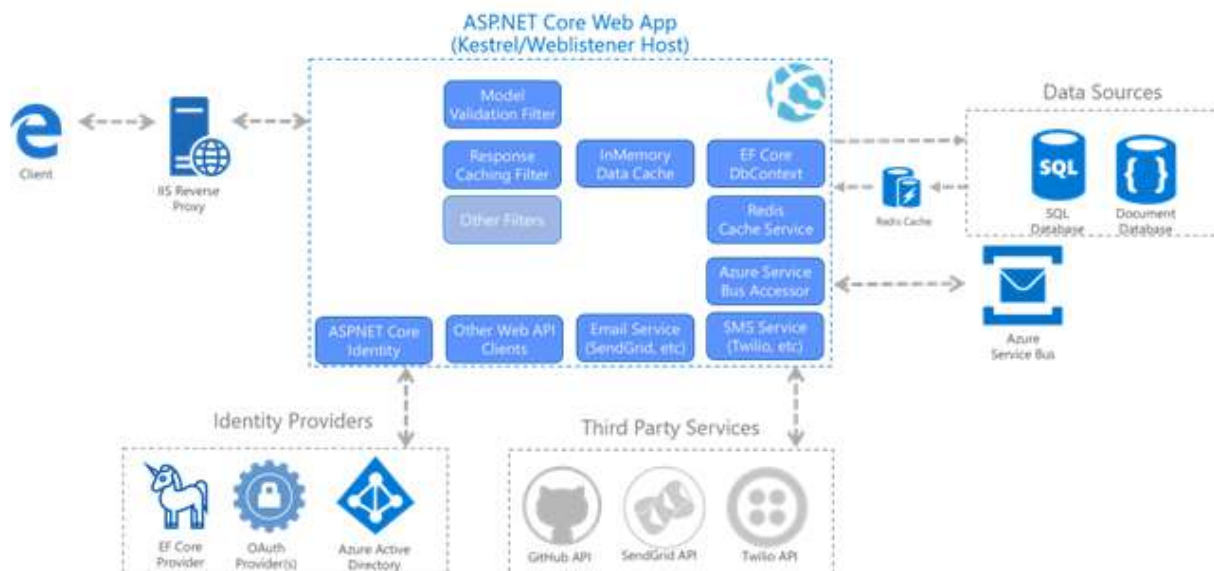


Рис. 3 – Архітектура ASP.NET Core 3

Розробка веб-сервісу для моніторингу температурного режиму печі.

Як було сказано раніше, розробка систем керування на основі веб-сервісів є універсальним підходом, що може бути застосований для великого класу об'єктів чи процесів. В статті розглянуто приклад побудови веб-сервісу для моніторингу температурного режиму печі випалу вапна. За характером процесів, що протікають в печі, розрізняють по висоті зони: підігріву, випалу і охолодження. У зоні підігріву, до якої відносять верхню частину печі з температурою пічного простору не вище 850°C, матеріал підсушується і підігрівається розжареним димовими газами, що підіймаються. Зона випалу розміщується в середній частині печі, де температури випалюваного матеріалу змінюється від 850 до 1200°C і потім до 900 °C тут з вапняку видаляється вуглекислий газ. Зона охолодження – нижня частина печі. В цій зоні вапно охолоджується від 900 до 50-100°C повітрям яке надходить знизу, а далі піднімається в зону випалу.

Ключовим завданням системи буде отримання даних про температуру з об'єкту керування, збереження інформації в базу даних, візуалізація температурної кривої, сигналізація при виході температури за гранично допустимі межі.

ASP.NET Core 3 базується на MVC-шаблоні (model – view – controller) в якому:

Контролер (controller) – реалізує бізнес-логіку системи і забезпечує зв'язок між користувачем, інтерфейсом, базою даних. Контролер отримує дані, що вводяться користувачем і здійснює їх обробку. В залежності від виконаної задачі користувачеві відправляється результат у вигляді веб-сторінки, xml-даних, файлу або перенаправлення запити.

Представлення (view) – це власне візуальна частина або призначений для користувача інтерфейс програми. Як правило, html-сторінка, яку користувач бачить, зайшовши на сайт.

Модель (model) – представляє клас, що описує логіку використання даних.

Для задачі керування температурним режимом створено два контролери:

- Home – призначений для відображення інформації оператору
- Kiln – представляє інтерфейс (API) для отримання і збереження інформації в базу даних.

Розглянемо більш детально API, які надає контролер печі для взаємодії:

- Temperatures – вибирає необхідну кількість значень температури з бази даних, а також виконує апроксимацію даних за МНК.
- Temperature – POST метод який дає можливість додавати записи до бази даних.
- Temperature – GET метод повертає поточне значення температури для сигналізації.

На рисунку 4 наведено структуру створеної бази даних. Оскільки вже використовується стек технологій Microsoft, то в роботі було обрано MS SQL. Для зручної роботи з базою даних використано додаток Entity Framework Core, який дозволяє працювати на основі контексту даних та параметрів підключення. Для безпеки роботи з БД та уникнення помилок було створено бібліотеку для доступу до БД (Рис. 5). В представленій бібліотеці реалізовано два інтерфейси Repository та Unit Of Work. Repository – інтерфейс взаємодії з кожним об'єктом що знаходиться в базі даних. Перевагою цього інтерфейсу є те, що робота з об'єктом бази даних не пов'язана з конкретною реалізацією класу.

Основними таблицями бази даних є EFMigrationsHistory – механізм міграції Entity Framework Core (EF), Kilns – перелік датчиків температур та їх назв, Temperatures – зберігає значення температури, номер датчика та часу вимірювання.

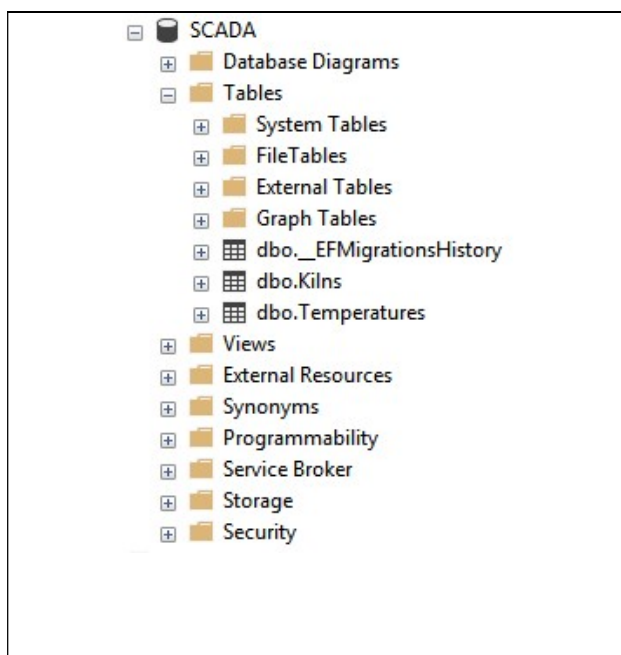


Рис. 4 – Структура БД у MS SQLServer Management Studio

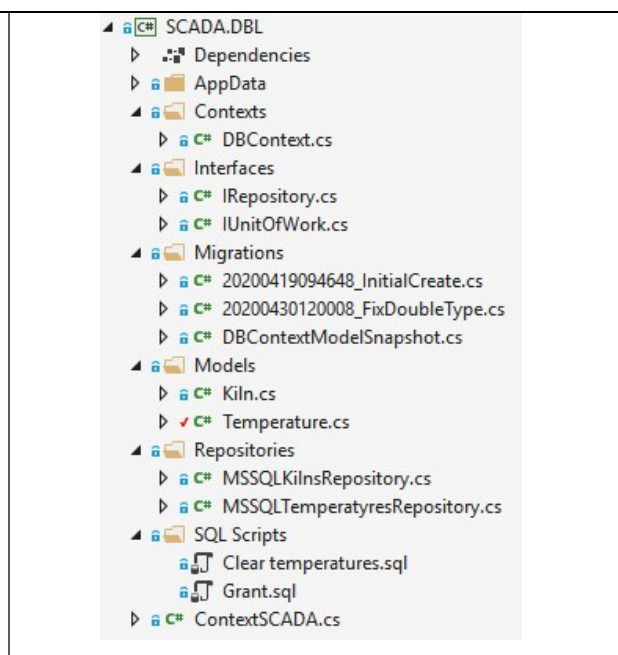


Рис. 5 – Структура бібліотеки для взаємодії з БД

Тестування веб-сервісу на прикладі моніторингу температури в кімнаті.

Для перевірки та тестування розробленого веб-сервісу було використано апаратну програмовану платформу Arduino з наступними модулями:

- Arduino Uno – апаратна платформа на мікроконтролері Atmega 328, має вбудований інтерфейс SPI, I2C, UART, 5 аналогових входів та 13 дискретних вбудовані лінії живлення 3.3 та 5 В.
- MAX6675 – модуль перетворення сигналу термопар (12-розрядний АЦП). Використовує інтерфейс SPI та дозволяє зчитувати дані з термопар в дискретному форматі
- Термопара типу К.
- ESP8266-01 – модуль роботи з інтерфейсом Wi-Fi. Дозволяє не тільки відправляти повідомлення, але й приймати їх від сервера для здійснення керування чи отримання сигналу завдання. Під'єднаний до контролера через програмний UART інтерфейс

- DS3132 – модуль годинника реального часу, під'єднаний по інтерфейсу I2C до Arduino, що забезпечує сповіщення модуля про час реєстрації температури.
- Всі перераховані вище елементи з'єднуються за схемою представленою на рисунку 6.

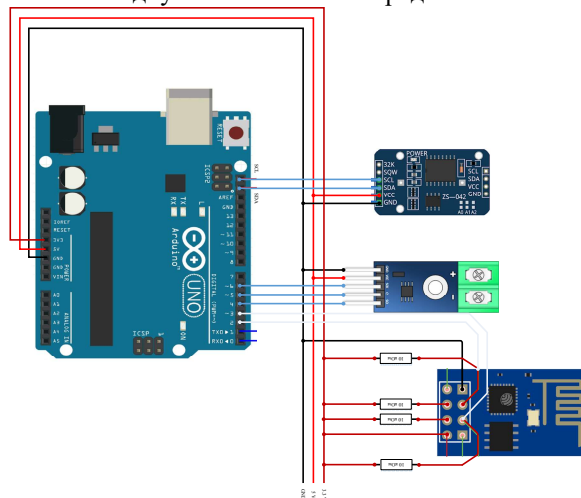


Рис. 6 – Схема підключення модулів Arduino для вимірювання температури

Модулі працюють за наступним алгоритмом: Arduino опитує модуль термопари та запам'ятовує значення, далі опитує часовий модуль та формує строку даних, в яку розміщує час та значення температури. Модуль контролера передає по запрограмованому UART інтерфейсу дані до ESP8266-01. Wi-Fi модуль зчитує дані по шині UART формує заголовки HTTP запиту та відправляє текст у форматі JSON до сервера.

На стороні веб-сервіса дані зберігаються в БД, візуалізуються у вигляді графіку, порівнюються з гранично допустимим значеннями і виводиться повідомлення для сигналізації оператору.

Висновки.

Розроблено програмне забезпечення веб-сервісу для моніторингу температурного режиму. На базі апаратної платформи Arduino виконане моделювання роботи системи для вимірювання температури в кімнаті. Проведене тестування показало, що веб-сервіс має необхідний рівень безпеки та зручний людинно-машинний інтерфейс, не перевантажує оператора надмірною інформацією, дозволяє проводити аналітику об'єкта, відстежувати поточний та минулий стан системи. За потреби розроблену систему можна масштабувати. Таким чином, веб-сервіс може бути впроваджений на будь-якому виробництві.

Список використаної літератури.

1. Pop E., Gifu D. A Cyber-Physical Systems Oriented Platform Using Web Services, 2019 22nd International Conference on Control Systems and Computer Science (CSCS), Bucharest, Romania, 2019, pp. 602-609, doi: 10.1109/CSCS.2019.00110.
2. Optimization of web-application performance / Kovaliuk D.O., Kovaliuk O.O., Pinaieva O.Y., Kotyra A., Kalizhanova A // Proceedings of SPIE - The International Society for Optical Engineering // Proc. SPIE 11176, Photonics Applications in Astronomy, Communications, Industry, and High-Energy Physics Experiments 2019, 1117626 (6 November 2019); doi: [10.1117/12.2537163](https://doi.org/10.1117/12.2537163)
3. Інтеграція програмних засобів систем керування / Д.О. Ковалюк, О.О. Ковалюк, В.І. Бородин, М.М. Степанюк // Вчені записки Таврійського національного університету імені В.І. Вернадського, Серія: технічні науки. Том 30 (69) Ч. 1 № 1, – 2019, ст. 56-60.
4. Turc, Traian Using WEB Services in SCADA Applications. // Procedia Technology. 19. 584-590. 10.1016/j.protecy.2015.02.083.
5. Bai Jianbo, Hao Yuzhe, Miao Guochang Integrating Building Automation Systems based on Web Services // Journal of software, Vol. 6, No. 11, November 2011 2209-2216

6. An Access Control System for Web Service Compositions / Srivatsa M., Iyengar A., Mikalsen T., Rouvellou I., Yin J. // IEEE International Conference on Web Services (ICWS 2007), Salt Lake City, UT, 2007, pp. 1-8, doi: 10.1109/ICWS.2007.31.
7. Khaoula Karimi, Salah-ddine Krit. Internet of Thing for Smart Home System Using Web Services and Android Application. 10.1007/978-981-13-8614-5_12.
8. Interoperability for Smart Home Environment Using Web Services / [Thinagaran Perumal](#), [Abdul Rahman Ramli](#), [Shattri Mansor](#), [Chui Yew Leong](#) // [International Journal of Smart Home](#), November 2008, https://www.researchgate.net/publication/228342322_Interoperability_for_Smart_Home_Environment_Using_Web_Services
9. An Analysis of a Web Service based Approach for Experimental Data Sharing / Kravev V., Kraveva R., Sinyagina N., Koprinkova-Hristova P., Bocheva N. // International Journal of Online Engineering (iJOE). 14. 19. 10.3991/ijoe.v14i09.8740.
10. Snell J, Tidwell D., Kulchenko P. Programming Web services with SOAP. O'Reilly & Associates, Inc., USA.

Надійшла до редакції 10.10.2020

Kovaliuk D. O., Karmazin I.M., Kovaliuk O.O.

USE OF WEB SERVICES IN TECHNOLOGICAL PROCESS CONTROL SYSTEMS

The optimal control of technological objects and processes is usually implemented by computer control systems with complicated structure and relations between elements. On the one hand, it allows to perform the functions of automatic control and regulation, alarms, process status registration and visualization. On the other hand, this leads to a significant complication of the systems because of increased process intensity, speed, accuracy and quality requirements.

Use of web services is the most popular approach to solve the problem of integration and coordination of system elements. It is based on the exchange of messages of a special (universal) format between parts of the system and processing the responses.

The role of web service and its place in control system has been analyzed. The most appropriate place is between the levels of PLC - controllers and SCADA one. It allows you to receive current information about the state of the technological process and implement the following functions: data acquisition by controllers; storage data to the database; providing data to the operator on demand and it's visualization in real time; monitoring and alarms of process parameters.

The web service build with Microsoft ASP.NET has been proposed for use in computer control systems. The key benefit of this technology are: open-source, cross-platform, fast and scalable.

The development of web-services in control systems is a universal approach that can be applied to a large class of objects or processes. In the article the web service for temperature control has been proposed. The key task of the system is temperature data acquisition from the control object, to save information to the database, visualize the temperature curve, notify when the temperature exceeds limits.

Since the Microsoft technology stack is already in use, MS SQL has been chosen. To increase the speed of development the Entity Framework Core application has been used, which provides a way to interact with data. To provide the security of the database and prevent development errors, a library has been implemented to access the database. It provides two interfaces Repository and Unit Of Work. Repository is an interface for interacting with each object in the database. The advantage of this interface is abstraction that removes dependency of a specific class implementation.

The hardware of control system consists of programmable platform Arduino and such modules: Arduino Uno – the platform on the Atmega 328 microcontroller with built-in interface SPI, I2C, UART, 5 analog inputs and 13 discrete built-in power lines 3.3 and 5 V; MAX6675 - signal conversion module (12-bit ADC); thermocouple type K; ESP8266-01 - module with Wi-Fi interface; DS3132 - real time clock module.

The modules interact according to the following algorithm: Arduino calls the thermocouple module and remembers the value, then calls the time module and forms a data period in which it places the time and temperature value. The controller module transmits data to the ESP8266-01 via a programmed UART interface. The Wi-Fi module reads data on the UART bus, forms HTTP request headers and sends JSON text to the server.

Thus, web service for SCADA-level has been developed. The control object based on the Arduino hardware platform has been simulated. The developed web service provides the necessary level of security, high indicators of convenience of the human-machine interface, allows to analyze the object and track the past state of the system.

Keywords : automation, control systems web services, ASP.NET Core.

References

1. Pop E., Gifu D. A Cyber-Physical Systems Oriented Platform Using Web Services, 2019 22nd International Conference on Control Systems and Computer Science (CSCS), Bucharest, Romania, 2019, pp. 602-609, doi: 10.1109/CSCS.2019.00110.
2. Optimization of web-application performance / Kovaliuk D.O., Kovaliuk O.O., Pinaieva O.Y., Kotyra A., Kalizhanova A // Proceedings of SPIE - The International Society for Optical Engineering // Proc. SPIE 11176, Photonics Applications in Astronomy, Communications, Industry, and High-Energy Physics Experiments 2019, 1117626 (6 November 2019); doi: [10.1117/12.2537163](https://doi.org/10.1117/12.2537163) (SCOPUS)
3. Kovaliuk D.O. Intehratsiia prohramnykh zasobiv system keruvannia: D.O. Kovaliuk, O.O. Kovaliuk, V.I. Borodin, M.M. Stepaniuk // Vcheni zapysky Tavriiskoho natsionalnoho universytetu imeni V.I. Vernadskoho, Seriia: tekhnichni nauky. – Tom 30 (69) Ch. 1 # 1, – 2019, st. 56-60. Turc, Traian. (2015). Using WEB Services in SCADA Applications. Procedia Technology. 19. 584-590. 10.1016/j.protcy.2015.02.083.
4. Turc, Traian Using WEB Services in SCADA Applications. //Procedia Technology. 19. 584-590. 10.1016/j.protcy.2015.02.083.
5. Bai Jianbo, Hao Yuzhe, Miao Guochang Integrating Building Automation Systems based on Web Services // Journal of software, Vol. 6, No. 11, November 2011 2209-2216
6. An Access Control System for Web Service Compositions / Srivatsa M., Iyengar A., Mikalsen T., Rouvellou I., Yin J. // IEEE International Conference on Web Services (ICWS 2007), Salt Lake City, UT, 2007, pp. 1-8, doi: 10.1109/ICWS.2007.31.
7. Khaoula Karimi, Salah-ddine Krit. Internet of Thing for Smart Home System Using Web Services and Android Application. 10.1007/978-981-13-8614-5_12.
8. Interoperability for Smart Home Environment Using Web Services / [Thinagaran Perumal](#), [Abdul Rahman Ramli](#), [Shatri Mansor](#), [Chui Yew Leong](#) // [International Journal of Smart Home](#), November 2008, https://www.researchgate.net/publication/228342322_Interoperability_for_Smart_Home_Environment_Using_Web_Services
9. An Analysis of a Web Service based Approach for Experimental Data Sharing / Krlev V., Krleva R., Sinyagina N., Koprinkova-Hristova P., Bocheva N. // International Journal of Online Engineering (iJOE). 14. 19. 10.3991/ijoe.v14i09.8740.
10. Snell J, Tidwell D., Kulchenko P. Programming Web services with SOAP. O'Reilly & Associates, Inc., USA.