

УДК 681.324:621.325

ІНТЕРАКТИВНА СИСТЕМА ОРГАНІЗАЦІЇ СПОРТИВНОГО КОМПЛЕКСУ

Уляна Дзелендзяк¹, Володимир Самотий^{2,3}

¹ Національний університет «Львівська політехніка»,
вул. С. Бандери, 12, Львів 79013, Україна

² Cracow University of Technology,
24, Warszawska St., Cracow, 31-155, Poland

³ Львівський державний університет безпеки життєдіяльності
вул. Клепарівська, 35, Львів, 79007, Україна

У статті пропонується інтерактивна система організації спортивного комплексу, яка складається з користувацького інтерфейсу, серверної логіки та бази даних. Розроблений програмний продукт є інтернет-додатком для менеджменту даних, який реалізований за ідеєю автоматизованої системи організації та збереження даних. Архітектура розробленої системи забезпечує високий рівень масштабованості та підтримованості. Реалізований інтернет-застосунок забезпечує динамічне керування інформацією, що зберігається на віддаленому сервері. Веб-додаток розроблений у середовищі проектування та розробки програмного забезпечення Visual Studio з використанням реляційної бази даних SQL, бекенду ASP .NET та фронтенду Angular. Щоб забезпечити функціонування системи за будь-якого сценарію, було додано функціонал PWA для можливості роботи без інтернету. Розроблену інтерактивну систему можна використовувати у різних сферах господарської діяльності, залежно від заповнення бази даних.

Ключові слова: інтерактивна система, реляційна база даних, прогресивний мобільний додаток, веб інтерфейс, бекенд, фронтенд.

Постановка проблеми. Сучасні веб-додатки – це інтерактивні системи, які є поєднанням абстрагованих компонент наділених своєю бізнес-логікою та строго визначеним, достатнім функціоналом для перевикористання їх у різних сценаріях роботи програми. Серверна частина таких систем відповідає за опрацювання та збереження даних, що відпраляються на інтерфейс користувача і отримуються з нього. Клієнтська частина відповідає за опрацювання вхідної інформації та інтерактивної взаємодії з кінцевим користувачем. Інтернет-додатки працюють згідно вимог, визначених у міжнародних стандартах, що забезпечує підтримку більшістю браузерів. Основною організацією, яка розробляє та встановлює стандарти вважається World Wide Web Consortium та співтовариство провідних корпорацій, що задають тренди веб-розробки. На сьогоднішній день веб-додатки - це розподілені персоналізовані додатки рівня підприємства, тому розроблення і підтримка таких програмних комплексів є досить складним і актуальним завданням.

Аналіз останніх досліджень. Теоретичні та практичні аспекти розроблення сучасних мобільних додатків висвітлені у багатьох наукових публікаціях, особливо у зарубіжних джерелах. Найбільш відомими серед науковців, які досліджували технічні особливості гібридної розробки мобільних додатків та прогресивних веб-додатків, є Р. Махато, Н. Гок, Н.Кханна та К. Т. Пхан [2]. Однак питання розроблення та використання таких веб-додатків для вирішення різноманітних задач залишаються актуальними.

Все більш широкого застосування набувають прогресивні веб-додатки (PWA). Основним їх завданням є реалізація програмного забезпечення, створеного з використанням певних технологій для досягнення заданих цілей у вигляді веб-додатку. Найчастіше такий додаток завантажується у браузері і є доступним одночасно для різних платформ. Також, PWA розпізнаються пошуковими системами як сайти, що сприяє ефективнішому розповсюдженню програмного забезпечення. У цих додатках реалізована система push-повідомлень, що дозволяє підвищити рівень поінформованості аудиторії та привернути увагу користувачів. [2].

Інтерактивні комп'ютерні додатки, які розроблені для мережі інтернет, дозволяють користувачам вводити, отримувати і маніпулювати даними за допомогою взаємодії. Такі програми, зазвичай, мають дуже тісний зв'язок із сервером, відправляючи на нього безліч запитів. Інформаційною основою таких додатків є база даних. На теперішній час не існує єдиного підходу до вибору типу баз даних, оскільки це залежить від специфіки поставленого завдання. Нереляційні бази даних набувають все більшої популярності, але все одно є ситуації, коли перевагу слід надати більш структурованим реляційним базам даних, які забезпечують транзакції та надійність системи у сенсі зв'язків між таблицями. Всі реляційні бази даних підтримують SQL, тому при зміні реляційних баз даних у користувача і розробника не виникне суттєвих проблем [3]. Крім того, у статті [4] стверджується, що не завжди є беззаперечна перевага у швидкодії між реляційними та нереляційними базами даних. *Розроблення інтерактивного веб-додатку з використанням нереляційної бази даних та сховища даних Firebase Cloud Storage наведено у статті [1].* На основі аналізу останніх досліджень для розроблення інтерактивної системи організації спортивного комплексу було прийнято рішення використати реляційну базу даних SQL.

Мета статті. Розроблення інтерактивної системи для організації роботи підприємства чи організації, з можливістю її використання для управління, ведення обліку та аудиту, на прикладі організації роботи спортивного комплексу.

Виклад основного матеріалу дослідження. Запропонована інтерактивна система організації спортивного комплексу складається з бази даних, сховища даних, бекенду та веб інтерфейсу. Для реалізації цієї системи був вибраний один із найоптимальніших відомих методів – це зв'язка інтерфейсу користувача Angular [5], бекенду ASP.NET [7] і реляційної бази даних SQL [6]. Основними перевагами такого підходу є: ефективна масштабованість системи, оптимальна обробка великих об'ємів інформації, а також швидка взаємодія

користувацького інтерфейсу і бази даних. На даний момент більшість технічних засобів даного типу не є добре оптимізованими для можливого подальшого розширення функціоналу. Тому у процесі проектування системи був запропонований алгоритм, заснований на архітектурі Onion для бекенду і програмна реалізація контейнерів для користувацького інтерфейсу.

Проектування інтерактивної системи складається з визначення складових елементів та побудови її структури, яка повинна відповідати таким вимогам: оптимальний розподіл даних між частинами системи; безпека інформації, що зберігається у базі даних; швидка взаємодія інтерфейсу користувача та бекенду; легке розширення бази даних. На рис.1 наведено схему реалізації такої системи.

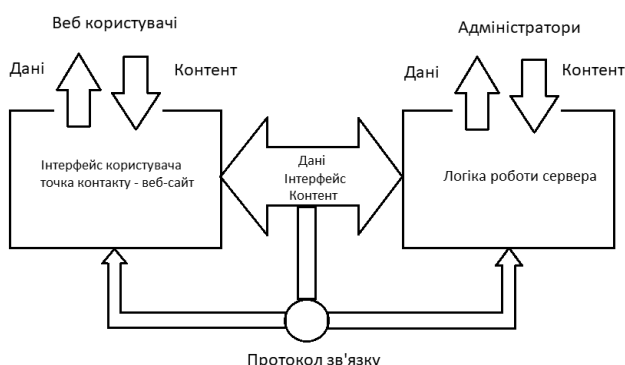


Рис. 1. Схема реалізації зв'язків між компонентами системи

Така зв'язка складових дозволить оптимально розподілити ресурси робочої машини та серверу для ефективної роботи користувача. Вибраний підхід є широко розповсюдженим і надає високу швидкість взаємодії та ефективного заощадження ресурсів.

Розроблена інтерактивна система організації спортивного комплексу реалізується в одному проекті з використанням архітектурного поділу на складові частини. Архітектура системи включає такі директорії: веб-додаток (App); бізнес-логіку (BAL –business architecture level); базу даних – (DAL – database architecture level) і сервер (Web Api).

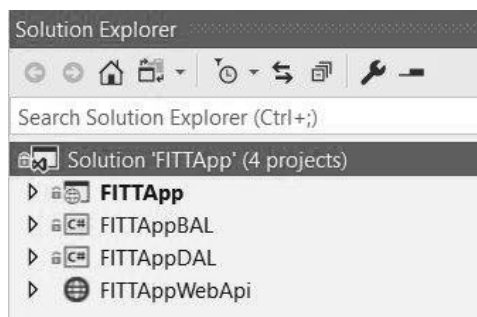


Рис. 2 Архітектура системи

Веб-додаток реалізований у середовищі розробки програмного забезпечення Visual Studio з використанням фреймворку Angular [5]. Архітектурне рішення розподілу складових системи дозволить забезпечити високий рівень маштабованості та якісний, підтримуваний код. Архітектура додатку включає: сервіс авторизації (Auth); контролери (Controllers); утиліти роботи з файлами (FileUploads); фільтри (Filters); моделі (Models); провайдери (Providers); представлення (Views).

Папка авторизації містить файли перевірки і забезпечення аутентифікації користувача в системі. Тримає токен – ідентифікатор, що визначає сесію та основні метадані у зашифрованому вигляді. Контролери вміщують більшу частину бізнес-логіки веб-додатку, та керують прив'язаними до представлення даними. Часто, в контролер імпортуються сервіси, фільтри, моделі та директиви. Такий підхід дозволяє перевикористовувати частини коду, зменшити затрати часу на розробку та підвищити читабельність коду.

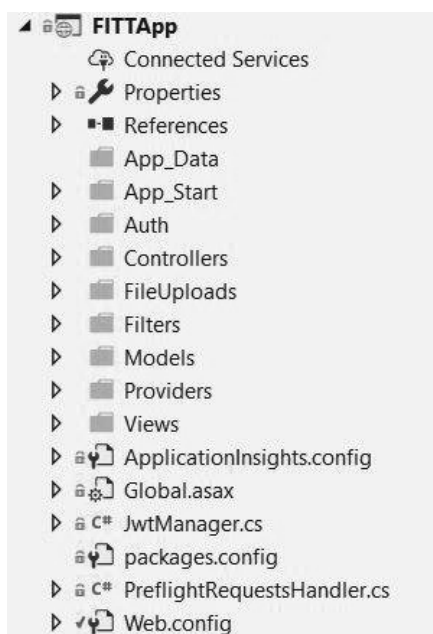


Рис. 3. Архітектура веб-додатку

Утиліти – це службові методи, які обробляють дані, трансформують їх, модифікують та видаляють. Фільтри – методи, що здійснюють вибірку. Бувають фільтри, які можуть вибирати дані зі сховища даних веб-додатку, локального сховища, чи, рідше, з куків. Моделі – методи та сервіси, що дозволяють модифікувати методи, завдяки переданим функціям зворотнього виклику. Можуть перевикористовуватись в різних функціональних частинах додатку. Моделі повертають модифіковані дані, у певному, заданому задалегідь форматі. Працюють із цими даними компоненти. Представлення –

це структурні складові додатку, що містять HTML розмітку і визначають відображення даних у DOM (Document Object Model).

У веб-АРІ широко використовується метод передачі даних для підключення інтерфейсу (клієнт) і бекенд (сервер). Це шлюз між клієнтом і сервером. Передача даних між веб-АРІ відбувається за допомогою протоколів HTTP.

Загальний вигляд архітектури серверної логіки складається з двох частин DAL та BAL. DAL містить залежності, що відповідають за зв'язок серверної логіки додатку з базою даних. BAL забезпечує функціонування серверної логіки, зв'язку з базою даних та користувацьким інтерфейсом.

На рис. 4 наведена архітектура бази даних інтерактивної системи організації спортивного комплексу.

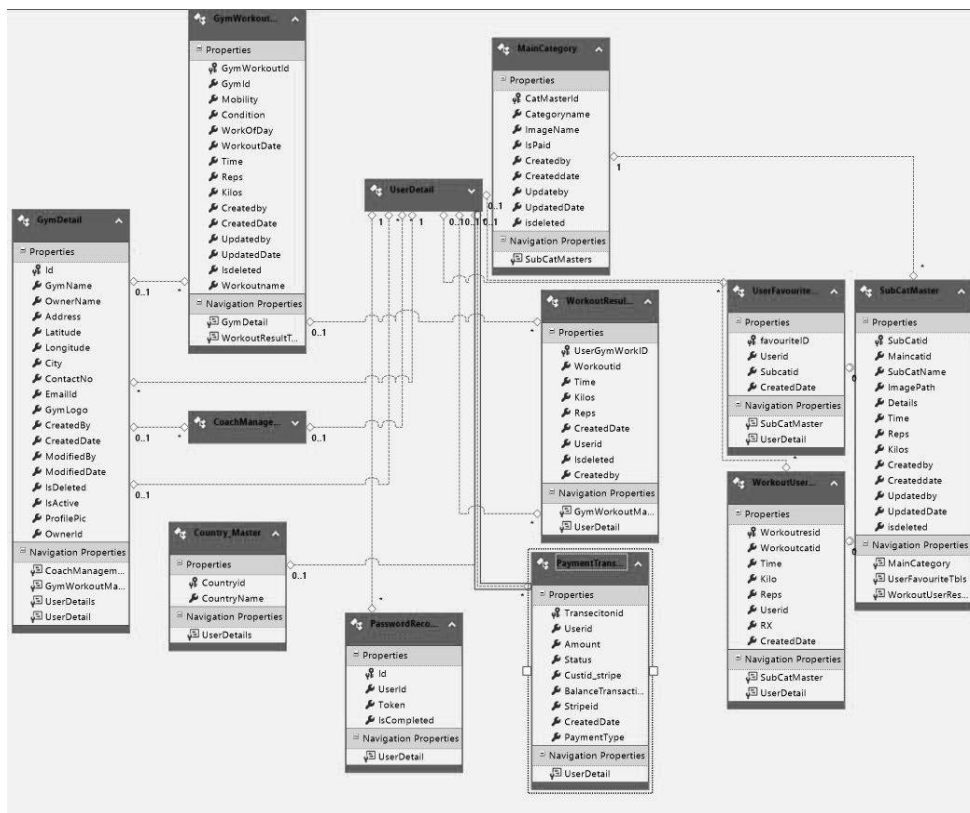


Рис. 4. Архітектура бази даних

База даних інтерактивної системи організації спортивного комплексу містить такі таблиці: деталі спортзалу; результати тренувань; деталі клієнта; деталі тренувань; скидання паролю; загальну таблицю зв'язків; дані тренера/менеджера. Наведена база даних є реляційною та зведеною до третьої нормальної форми [6]. Розроблену базу даних можна заповнити довільною інформацією, в залежності від сфери використання інтерактивної системи.

Висновок. Розроблено інтерактивну систему організації спортивного комплексу, яка складається з користувацького інтерфейсу, серверної логіки та бази даних. Спроектовано архітектуру користувацького інтерфейсу, серверної логіки і бази даних таким чином, щоб забезпечити високий рівень масштабованості та підтримованості системи. Для забезпечення функціонування системи за будь-якого сценарію було додано функціонал PWA для можливості роботи без інтернету. Розроблену інтерактивну систему можна використовувати у різних сферах господарської діяльності, залежно від заповнення бази даних.

Список літератури

1. Дзелендзяк У.Ю., Самотий В.В., Палюшок В.О. Розроблення інтерактивного веб-додатку з використанням нереляційної бази даних //Науковий журнал «Автоматика, вимірювання та керування» №1 (1)– 2019. С. 25 - 31.
2. Мороз Т.О., Ендрес В.С. Переваги використання гібридних мобільних додатків та прогресивних веб-додатків у бізнесі// Вісник аграрної науки Причорномор'я. – 2019. – Вип. 1. – С. 96-102.
3. Швець М.Ю., Заруба Д.С., Хохлов Ю.В. Порівняння SQL та NoSQL баз даних. / Вчені записки ТНУ імені В.І. Вернадського. Серія: Технічні науки. – Том 29 (68) ч.2. №6, 2018 - С. 21-25.
4. Li Y., Manohara S. Aperformance comparation of SQL and NoSQL databases. Conference: Communications, Computers and Signal Processing (PACRIM), 2013 IEEE Pacific Rim Conference on, pp. 15-19, IEEE, August 2013.
5. Ari Lerner. Ng-book - The Complete Book on Angular, 2017 - 340 p.
6. Batra Rahul. SQL Primer: An Accelerated Introduction to SQL Basics, 2018 – 213 p.
7. Lok Andrew. ASP. NET Core in Action, Manning Publications, 2018 – 712p.

REFERENCES

1. Dzelendzyak U.Yu., Samoty V.V., Paliushok V.O. (2019). Rozroblennia interaktyvnogo veb-dodatku z vykorystanniam nereliatsiinoi bazy danykh //Naukovyi zhurnal "Avtomatyka, vymiryuvannia ta keruvannia" №1 (1)– S. 25 - 31. (in Ukrainian)
2. Moroz T.O., Endres V.S. (2019). Perevagy vykorystannia gibrydnykh mobilnykh dodatktiv ta progresyvnykh veb-dodatktiv u biznesi// Visnyk agrarnoi nauky Prychornomoria. — Vyp. 1. – S. 96-102. (in Ukrainian)
3. Shvets M.Yu., Zaruba D.S., Khokhlov Yu.V. (2018). Porivniannia SQL ta NoSQL baz danykh / Vcheni zapysky TNU imeni V.I. Vernadskogo. Serii: Tekhnichni nauky. – Tom 29 (68) ch.2. №6 - S. 21-25. (in Ukrainian)
4. Li Y., Manohara S. Aperformance comparation of SQL and NoSQL databases. Conference: Communications, Computers and Signal Processing (PACRIM), 2013 IEEE Pacific Rim Conference on, pp. 15-19, IEEE, August 2013. (in English)
5. Ari Lerner. (2017). Ng-book - The Complete Book on Angular - 340 p. (in English)
6. Batra Rahul. (2018). SQL Primer: An Accelerated Introduction to SQL Basics – 213 p. (in English)
7. Lok Andrew. (2018). ASP. NET Core in Action, Manning Publications – 712p. (in English)

DOI 10.32403/2411-9210-2021-1-45-61-67

THE INTERACTIVE SYSTEM OF SPORT COMPLEX ORGANIZATIONUlyana Dzelendzyak¹, Volodymyr Samoty^{2,3}¹Lviv Polytechnic National University
12, S.Bandera St., Lviv, 79013, Ukraine²Cracow University of Technology,
24, Warszawska St., Cracow, 31155, Poland,³Lviv State University of Life Safety,
35, Kleparivska St., Lviv, 79007, Ukraine
dzelandzyakl@gmail.com

In this article the interactive system of sport complex organization, which consists of user interface, server logic and database is considered. The developed program product is Internet application for data management which is realized according to idea of organization and store data automatized system. The architecture of developed system provides high level of scalability and maintenance. The implemented internet application provides dynamic information control which is stored on distant server. Web application is designed in the Visual Studio environment with the usage of relational database SQL, ASP .NET backend and Angular frontend. In order to extend system design functionality the algorithm based on Onion architecture and the containers program realization for users interface has been proposed. The developed interactive system provides processing and storing data on the server; data transferring to the user interface, data editing, storing, updating and deleting data. In order to supply system functionality for any scenario even without Internet the PWA has been added. The developed interactive system could be used in different areas of economic activity depending on database content.

Keywords: interactive system, relational database, progressive mobile application, web interface, backend, frontend.

Стаття надійшла до редакції 12.01.2021

Received 12.01.2021