

УДК 004.6

АКТУАЛЬНІСТЬ ВИКОРИСТАННЯ WINDOWS PRESENTATION FOUNDATION ДЛЯ РОЗРОБКИ ІНФОРМАЦІЙНОЇ СИСТЕМИ ПІДБОРУ СПОРТИВНИХ ТРЕНЕРІВ

Березоручька Олена Валеріївна

к.т.н., доцент каф. ІТ, Рудніченко Микола Дмитрович

Національний університет «Одеська політехніка», УКРАЇНА

АНОТАЦІЯ. В рамках виконання даної роботи виконано опис структури проекту створеної інформаційної системи автоматизації підбору тренерів для спортивних організацій, а також проведено аналіз актуальності впровадження технології *WPF* від *Microsoft* до сучасних інформаційних систем.

Вступ. У зв'язку з постійним зростанням вимог до обов'язків щодо автоматизації процесів обліку та проведення трудової діяльності сучасних менеджерів спортивних організацій (фітнес-клуби, тренажерні зали тощо) актуальним є впровадження комплексних рішень щодо інформатизації та управління ключовими операційними процесами у межах компаній [1].

Мета роботи. Мета роботи полягає у розробці інформаційної системи для підбору спортивних тренерів по різних критеріях для проведення консультацій та персональних занять.

Основна частина роботи. Зовнішня частина (графічний інтерфейс, з яким взаємодіють користувачі) інформаційної системи для підбору спортивних тренерів було реалізовано завдяки використанню платформи інтерфейсу користувача *Windows Presentation Foundation (WPF)*. Технологія *WPF* є частиною екосистеми платформи *.NET*. Для створення логіки програми тут використовують *C#*, *F#* або *VB.NET*. Треба зазначити, що хоча *.NET* є кросплатформовою технологією, *WPF* працює лише у *Windows*.

Програми *WPF* засновані на *DirectX*, тобто рендеринг графіки лягає на графічний процесор на відеокарті. Крім того, існує можливість апаратного прискорення графіки. Оскільки у *WPF* всі елементи вимірюються в незалежних від пристрою одиницях, програми на *WPF* легко масштабуються під різні екрани з різною роздільною здатністю. Для декларативного визначення графічного інтерфейсу використовують мову розмітки *XAML (eXtensible Application Markup Language)*, яка заснована на *XML* і яка представляє альтернативу програмному створенню графіки та елементів управління. У *WPF* ця мова використовується перш за все для створення інтерфейсу користувача декларативним шляхом (як *HTML* у веб-програмуванні), хоча функціональність *XAML* не обмежується тільки графічним інтерфейсом. *XAML* - не обов'язкова частина застосунку, елементи управління можна створювати програмно (код елементів управління на *C#* буде громіздкий, на відміну від *XAML*-варіанта) [3]. Використання *XAML* дає можливість відокремити графічний інтерфейс від логіки програми, завдяки чому над різними частинами програми можуть працювати різні фахівці: над інтерфейсом – дизайнери, над кодом логіки – програмісти. При компіляції програми *Visual Studio* код у *XAML*-файлах також компілюється в бінарне представлення *BAML (Binary Application Markup Language)*. Потім код *BAML* вбудовується у фінальне збірку програми – *exe* або *dll*-файл. Новими можливостями, порівняно із *Windows Forms*, яких було складно досягти, є прив'язка даних, створення тривимірних моделей та анімацій, використання стилів і шаблонів, можливість створювати нові елементи управління. Обсяг програм на *WPF* і споживання ними пам'яті у процесі роботи у середньому трохи вище, ніж у *WinForms*, але це компенсується отриманими можливостями.

WPF було вперше представлено в *.Net Framework 3* у 2006 році. Постає питання, чи варто наразі розглядати цю технологію для створення нових програм. Ольга Гавриш, випускниця Харківського національного університету радіоелектроніки, *Senior Product Manager* у *Microsoft*, зазначає, що *WPF* є «дуже зрілим проектом», саме через це у технології не відбувається надто стрімкого розвитку. Але компанія *Microsoft* наголошує на тому, що буде підтримувати цей продукт, поки у нього будуть користувачі. Так, ми можемо бачити підтримку останньої версії

.Net, а також часткове оновлення функціоналу та виправлення помилок. Тож, можна підтримувати існуючі на цій технології проекти, але для створення нових застосунків краще обрати кросплатформові альтернативи, наприклад, *.NET MAUI*.

Перейдемо до формування внутрішньої логіки програми. У інформаційній системі, що розробляється, необхідно передбачити ряд користувальницьких ролей для розмежування прав доступу, зокрема:

1. Режим адміністратора, що забезпечує можливості створення та управління даними, їх модифікації та формування звітних зведень за заданими критеріями.

2. Режим користувача, що надає можливості з читання та відображення наявних даних по тренерах та їхній діяльності, а також записи на консультації з тренувань.

Вхідними потоками є дані щодо досягнень та особистої інформації про тренерів, а також інформація про їхні робочі графіки. Вихідними потоками є зведені звіти щодо загальної інформації про тренерів та подані заявки на підбір тренера для проведення відповідних тренувань. Керуючими потоками є такі, що регулюють правовий та технічний порядок функціонування системи, акти та специфікації, а також норми трудового кодексу щодо надання послуг усіх спортивних тренувань. Механізмами є адміністратори та користувачі, що здійснюють обробку та перегляд даних відповідно. Адміністратор може здійснювати: управління даними тренерів, зокрема операції створення, редагування, видалення та збереження записів за особистими даними, досягненнями, освітою та розкладом прийому на тренування; формування зведених звітів; складання графіків тренувань; здійснювання пошуку та перегляду даних; виконання авторизації в інформаційній системі [2]. Користувач має можливості з: авторизації у системі; перегляду даних (переліку тренерів та спортивних категорій); пошуку тренерів; підбору тренерів на базі фільтрації та сортування результатів за різними критеріями; записи на тренування за чинним розкладом тренера. У зв'язку з використанням фреймворку *Entity Framework* поділ більшості класів організовано за функціоналом їхніх зв'язків із сутностями бази даних та формами представлення (*XAML* шаблонами). Всього реалізовано 40 класів (класи спадкоємці від батьківського класу *Window* для зв'язку з уявленнями, класи бізнес-логіки та класи моделей, що успадковують клас *ObMigration*) і 2 перерахування (для оперування даними за статтю та датою). Головним класом з точкою входу в систему є *MainWindow*, концепт контракту кожного класу міститься у його назві.

Для збереження даних створено реляційну базу даних, структура якої наведена на рис. 1.

Таблиця *Sports* зберігає види спорту у системі, а також їхні мультимедійні відображення у вигляді посилань на зображення.

Таблиця *Coaches* - основна таблиця бази даних, що призначена для зберігання особистих даних про тренерів та для зв'язку інших таблиць між собою. Для більшості таблиць використано зв'язок «багато-до-багатьох» між відповідними таблицями і тренерами.

Таблиця *Languages* призначена для зберігання даних з мови спілкування тренерів, *Phones* - номерів телефонів; *Cities* і *WorkPlaces* відповідають за збереження інформації про робочі локації тренерів.

Таблиці *Ranks*, *Certificates*, *Diplomas*, *Courses* – призначені для зберігання даних спортивних розрядів, сертифікатів, дипломів тренерів та пройдених тренерами курсами. *Achievements* передбачає інформацію про досягнення тренерів у форматі назви, місця проведення і результату спортивного змагання чи заходу.

Таблиця *Schedules* призначена для зберігання даних розкладу тренувань (день, час початку, час закінчення). *Training* та *AwayTraining* зберігають умови консультацій очних і виїзних тренувань (назва, час та вартість); *Reviews* зберігає відгуки про тренерів.

Висновки. На базі створеної інформаційної системи стає можливим автоматизація процесу інформаційної підтримки виробничої діяльності з підбору спортивних тренерів, перегляду їхніх досягнень і результатів праці у формі розробки прикладного програмного застосування та бази даних. Створена система дозволяє підвищити ефективність обліку даних та зменшити часові витрати на пошук та підбір тренерів на 15-20% у порівнянні з ручним режимом.

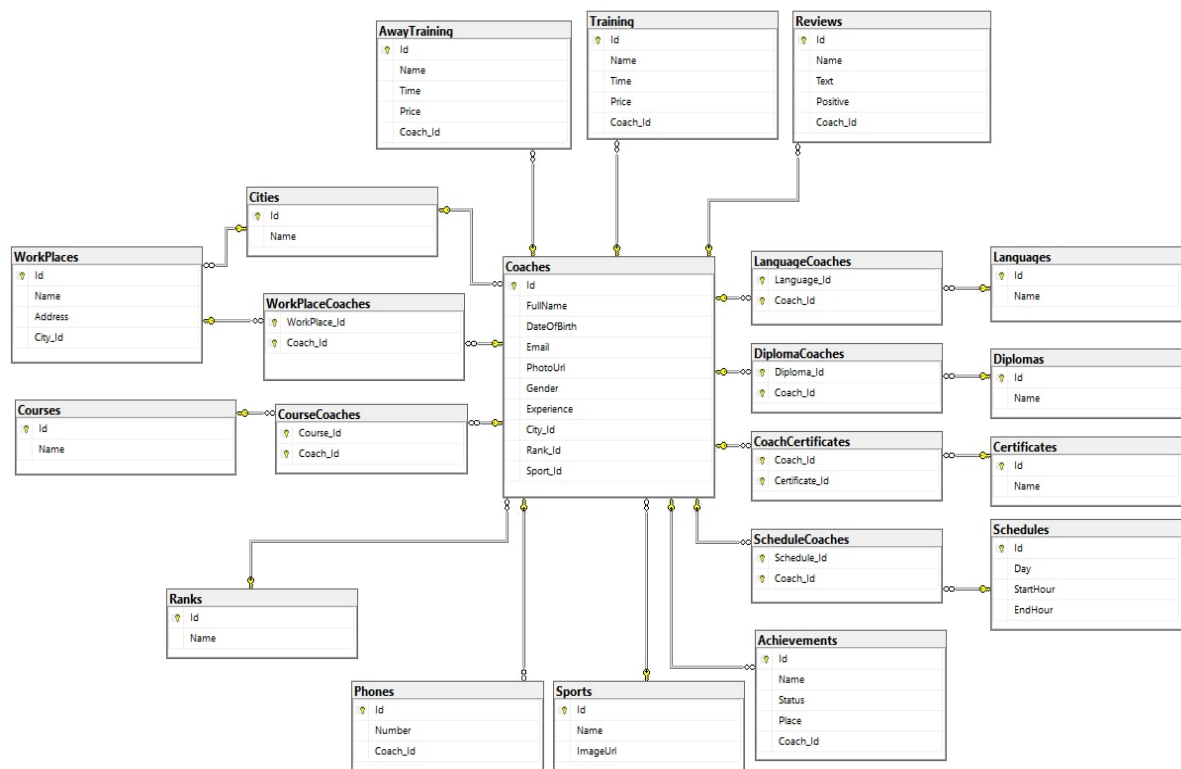


Рисунок 1 – Схема зв'язків між таблицями БД

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Березоручька О.В. Застосування канбан-методу для розробки програмного забезпечення для полегшення процесу вибору подарунків. «Сучасні інформаційні технології - 2021» «Modern Information Technology - 2021»: матеріали одинадцятої Міжнародної наукової конференції студентів та молодих учених, м.Одеса, 13-14 травня 2021 р. / МОН України; Державний університет «Одеська політехніка»; Ін-т комп'ют. систем. – Одеса : Наука і техніка, 2021. – с. 22-23.
2. Connolly T. Database Systems: A Practical Approach to Design, Implementation, and Management. London: Pearson, 2014. 1440 p.
3. Yuen S. Mastering Windows Presentation Foundation: Build responsive UIs for desktop applications with WPF. Birmingham: Packt Publishing, 2020. 638 p.

RELEVANCE OF USING THE WINDOWS PRESENTATION FOUNDATION FOR THE DEVELOPMENT OF AN INFORMATION SYSTEM FOR THE SELECTION OF SPORTS COACHES

Olena Berezorutska.,
 PhD., associate professor of IT Department, Mykola RudnichenkoYf
 National University "Odesa Polytechnic", UKRAINE

ANNOTATION. As part of this work, a description of the project structure of the created information system for automating the selection of coaches for sports organizations was made, as well as an analysis of the relevance of the introduction of WPF technology from Microsoft to modern information systems.