

1. Tytuł projektu

Nazwa aplikacji: FitnessManager

Opis: Aplikacja konsolowa do zarządzania siłownią, użytkownikami i sprzętem na podstawie systemu ról.

2. Opis projektu

Cel projektu: Umożliwienie obsługi podstawowych procesów zarządzania siłownią, takich jak: rejestracja/logowanie, przydzielanie ról, obsługa sprzętu, przypisywanie trenerów itp.

Co robi aplikacja: Pozwala na tworzenie i zarządzanie kontami użytkowników, ich rolami i interakcjami w systemie.

Dla kogo: Dla administratorów, pracowników siłowni, trenerów personalnych i klientów.

3. Technologie

Język programowania: C#

Środowisko: .NET Core

IDE: Visual Studio

Biblioteki/narzędzia: System.IO, System.Security.Cryptography, SHA256 do haszowania haseł

4. Struktura katalogów

FitnessManager.sln – plik rozwiązania projektu

Program.cs – główna logika aplikacji

Struktura klas: foldery access(w nim jest plik RBAC.cs zajmujący się dostępem do funkcji na podstawie uprawnień),

classes(w nim są pliki User.cs abstrakcyjna klasa, po której dziedziczą: Member.cs klasa, która zawiera informacje o członku klubu fitness, PersonalTrainer.cs klasa, która zawiera informacje o trenerze klubu fitness, Worker.cs klasa, która zawiera informacje o pracowniku klubu fitness, Admin.cs klasa, która zawiera informacje o administratorze klubu fitness),

managing(w nim jest plik FileManager.cs, który zajmuje się zarządzaniem plików),

other(w nim znajdują się pliki GetUsers.cs - klasa, która zawiera funkcje, które wypisują listy użytkowników ze względu na ich role, Menu.cs klasa, która zawiera metody, które wypisują menu użytkowników ze względu na ich role)

Bin//Debug - w tym folderze znajdują się pliki tekstowe takie jak users.txt lub logs.txt

5. Instalacja i uruchomienie

Wymagania: .NET Core 3.1 lub nowszy

Sposób instalacji:

Pobierz repozytorium.

Otwórz projekt w Visual Studio.

Zbuduj i uruchom aplikację (Ctrl+F5).

6. Opis działania aplikacji

Funkcjonalności:

Rejestracja i logowanie

Zmiana hasła

RBAC – role: Admin, Worker, PersonalTrainer, Member (w pliku ‘role.pdf’ informacje o przykładowych użytkownikach i ich hasłach)

Dodawanie/Usuwanie maszyn

Przydzielanie trenerów do członków

Obsługa logów i plików tekstowych

Przepływ programu:

Menu zależne od roli

Logowanie i wybór opcji

Przetwarzane dane:

Nazwa użytkownika, hasło (zahashowane), rola, interakcje z maszynami

7. Zrzuty ekranu

```
==> Fitness Manager ==>
| 1. Zaloguj się |
| 2. Wyjdź z programu |
|-----|
Wybierz opcję:
```



```
==> Fitness Manager (Admin: MainAdmin) ==>
| 1. Wyświetl wszystkich użytkowników |
| 2. Wyświetl adminów |
| 3. Wyświetl członków |
| 4. Wyświetl trenerów personalnych |
| 5. Wyświetl pracowników |
| 6. Dodaj użytkownika |
| 7. Usuń użytkownika |
| 8. Dodaj maszynę |
| 9. Wyświetl maszyny |
| 10. Usuń maszynę |
| 11. Zmień hasło |
| 12. Wyloguj się |
|-----|
Wybierz opcję:
```

```
==> Fitness Manager (Pracownik: Worker1) ===
| 1. Dodaj maszynę |
| 2. Wyświetl maszyny |
| 3. Usuń maszynę |
| 4. Zmień hasło |
| 5. Wyloguj się |
-----
```

Wybierz opcję:

```
==> Fitness Manager (Trener: Trainer1) ===
| 1. Wyświetl swoich członków |
| 2. Zmień hasło |
| 3. Wyloguj się |
-----
```

Wybierz opcję: |

```
== Fitness Manager (Członek: Member1) ==
| 1. Wyświetl trenerów personalnych |
| 2. Kup usługę trenera           |
| 3. Anuluj usługę trenera       |
| 4. Zmień hasło                 |
| 5. Wyloguj się                  |
|-----|
```

Wybierz opcję: |

```
== Fitness Manager (Admin: MainAdmin) ==
| 1. Wyświetl wszystkich użytkowników |
| 2. Wyświetl adminów                |
| 3. Wyświetl członków               |
| 4. Wyświetl trenerów personalnych |
| 5. Wyświetl pracowników          |
| 6. Dodaj użytkownika             |
| 7. Usuń użytkownika              |
| 8. Dodaj maszynę                 |
| 9. Wyświetl maszyny              |
| 10. Usuń maszynę                 |
| 11. Zmień hasło                  |
| 12. Wyloguj się                  |
|-----|
```

Wybierz opcję: 9
1. LegPressMachine
2. Bench
3. SquatMachine
4. Dumbbells
5. Barbells
6. LatsMachine
7. ExerciseMat
8. Treadmill
9. Ball

Naciśnij dowolny klawisz, aby kontynuować...

8. Przykład użycia (krok po kroku)

Użytkownik uruchamia aplikację

Tworzy konto lub loguje się

W zależności od roli, widzi różne opcje:

Admin: zarządza maszynami, użytkownikami

Member: kupuje dostęp do trenera

Worker: zarządza sprzętem

Akcje są zapisywane w pliku logs.txt

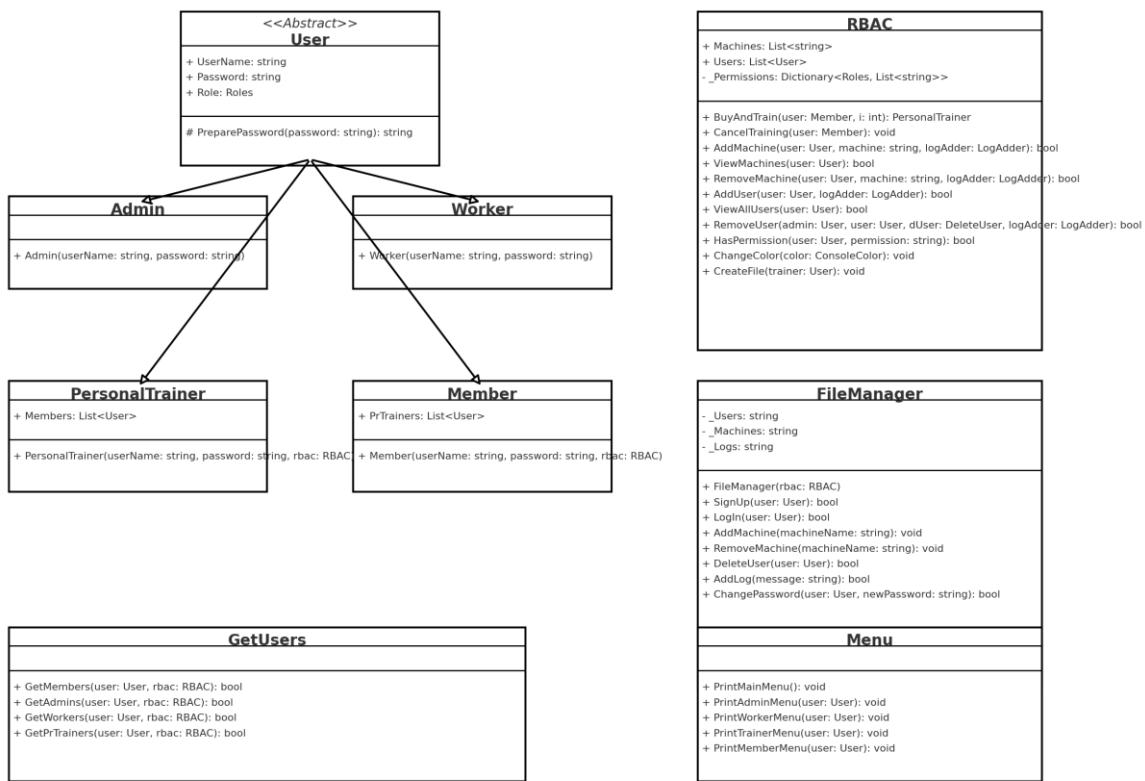
9. Struktury danych i klasy

User (abstrakcyjna): pola UserName, Password, Role

Admin / Worker / PersonalTrainer / Member – klasy dziedziczące z różnymi funkcjami

RBAC – przypisanie uprawnień do ról

FileManager – logika logowania, rejestracji, logowania akcji do pliku



10. Obsługa błędów

Błędy typu: użytkownik już istnieje, niepoprawne hasło, brak uprawnień

Komunikaty tekstowe w konsoli

11. Testowanie

Testy ręczne na różnych ścieżkach: logowanie, złe dane, zmiana hasła, dostępność maszyn

Brak testów jednostkowych w projekcie

12. Problemy i ograniczenia

Brak interfejsu graficznego

Dane są przechowywane w plikach .txt, a nie w bazie danych

13. Plany rozwoju

Dodanie interfejsu GUI

Przeniesienie danych do bazy (np. SQLite, MongoDB)

Rozszerzenie o REST API

Możliwość zarządzania przez przeglądarkę

14. Autorzy

Imię i nazwisko: Miłosz Hasik, Piotr Tomaszewski

Kontakt: hasikmilosz@gmail.com,

milosz.hasik@uczen.zsk.poznan.pl,

piotr.tomaszewski@uczen.zsk.poznan.pl