Projekt roczny — specyfikacja

Czas: 2 × 45 min

praca indywidualna lub w parach (za zgodą nauczyciela; w parze — podział zadań w specyfikacji) Wykład 1.30

Cel: Przygotujesz specyfikację własnej aplikacji WPF: wymagania, makiety okien, projekt klas, plan testów i harmonogram.

Po zajęciach potrafisz:

- formułuje sprawdzalne, ponumerowane wymagania
- projektuje makiety okien z nazwami kontrolek
- dzieli aplikację na klasy logiki i interfejs
- planuje testy i harmonogram prac

Sprzęt i materiały

- komputer z Windows 11 i dostępem do internetu
- pendrive lub konto w repozytorium do zapisania projektu

Oprogramowanie

- **Visual Studio Community 2026 (obciążenie „Programowanie aplikacji klasycznych platformy .NET")** — darmowe dla uczniów i nauczycieli; w pracowni zainstalowane visualstudio.microsoft.com/pl/downloads
- **.NET 10 SDK (LTS)** — instalowany razem z Visual Studio; sprawdzenie: `dotnet --version` dotnet.microsoft.com/pl-pl/download/dotnet/10.0
- **Git for Windows** — lub Git wbudowany w Visual Studio git-scm.com/downloads/win
- **Narzędzie do makiet (dowolne): kartka, Paint, Figma lub Penpot** — Penpot — darmowy, open source penpot.app

Bezpieczeństwo

- Zapisuj projekt w swoim folderze (np. `C:\Users\<login>\source\repos`), a na koniec zajęć skopiuj go na pendrive lub wyślij do repozytorium.
- Nie uruchamiaj programów i pakietów z nieznanych źródeł; pakiety NuGet instaluj tylko te, które wskazuje zadanie.

Przebieg zadania

1 Temat i wymagania

Wybierz temat (przykłady w lekcji 1.30) i uzgodnij go z nauczycielem.

1. Zapisz 6–10 ponumerowanych wymagań funkcjonalnych (W1, W2...), każde sprawdzalne.
2. Wymagania obowiązkowe: co najmniej 2 okna, zapis i odczyt danych z pliku (CSV lub JSON), walidacja danych, klasy logiki z testami.
3. Dopisz 2–3 wymagania pozafunkcjonalne (np. minimalny rozmiar okna, polskie znaki, czas wczytania).

2 Makiety

Narysuj każde okno aplikacji.

1. Na makiecie podpisz nazwy kontrolek (TxtTytul, BtnDodaj...) i kontenery układu.
2. Zaznacz, które wymagania realizuje każde okno.
3. Opisz przejścia między oknami (co otwiera które okno, Show czy ShowDialog).

3 Projekt klas

Rozpisz klasy logiki.

1. Dla każdej klasy: nazwa, właściwości (z typami), metody (parametry, wynik, wyjątki).
2. Zaznacz, które klasy nie zależą od WPF (będą testowane).
3. Narysuj prosty diagram klas.

4 Testy i harmonogram

Plan pracy.

1. Zaplanuj co najmniej 5 testów jednostkowych: metoda, dane, oczekiwany wynik.
2. Rozpisz harmonogram na zadania 33 i 34 oraz pracę domową.
3. Utwórz repozytorium Git z `.gitignore`, folderem `docs` ze specyfikacją i pierwszym commitem.
4. Przedstaw specyfikację nauczycielowi do zatwierdzenia.

W sprawozdaniu umieść

- ☐ specyfikacja: temat, wymagania W1..., wymagania pozafunkcjonalne
- ☐ makiety okien
- ☐ projekt i diagram klas
- ☐ plan testów
- ☐ harmonogram
- ☐ adres lub zrzut repozytorium

Pytania kontrolne

1. Które z Twoich wymagań było najtrudniej sformułować w sposób sprawdzalny?
2. Które klasy będą testowane jednostkowo i dlaczego właśnie one?
3. Jakie ryzyko widzisz w harmonogramie i jak je ograniczysz?

Kryteria oceny

Kryterium	Punkty
Wymagania funkcjonalne i pozafunkcjonalne	6
Makiety z nazwami kontrolek	5
Projekt i diagram klas	4
Plan testów	3
Harmonogram i repozytorium	2
Razem	**20**

Sprawozdanie przygotuj na wzorze dołączonym do zadania (plik .docx o tym samym numerze).