

Projekt roczny cz. 2 — HTML i CSS

Czas: 2 × 45 min

praca w zespołach 2–3 osobowych

Cel: Zbudowanie wszystkich podstron projektu zespołowego w semantycznym HTML i responsywnym CSS zgodnie z makietami, z regularnymi commitami każdego członka zespołu.

Po zajęciach potrafisz:

- tworzy spójny zestaw podstron z wspólnym nagłówkiem, menu i stopką
- stosuje znaczniki semantyczne, poprawną hierarchię nagłówków, tabele i formularze z etykietami
- pisze arkusz CSS w podejściu mobile first z punktami przerwania, Flexboxem i Gridem
- przygotowuje grafiki o właściwych wymiarach i rozmiarach plików
- pracuje w zespole z Gitem: osobne gałęzie, scalanie, opisowe commity

Sprzęt i materiały

- komputer z Windows 11 i dostępem do internetu
- repozytorium projektu z zadania 31 (karta `PROJEKT.txt`, makiety)

Oprogramowanie

- **Visual Studio Code** — Live Server, panel Source Control
code.visualstudio.com
- **Git i GitHub** — gałęzie i scalanie (zadanie 28)
git-scm.com/downloads
- **GIMP** — skalowanie i eksport grafik (WebP/JPG/PNG)
gimp.org/downloads
- **Validator W3C** — HTML; CSS:
<https://jigsaw.w3.org/css-validator/validator.w3.org>

Przygotowanie przed zajęciami

- Każdy członek zespołu wykonuje `git pull`, aby mieć aktualną wersję repozytorium.
- Przejrzyjcie makiety i podział zadań z karty projektu; ustalcie wspólne nazwy klas (np. `.wrap`, `.karta`, `.przycisk`).

Bezpieczeństwo

- Każdy pracuje na swojej gałęzi i edytuje przede wszystkim swoje pliki — tak unikniecie nadpisania pracy kolegi.
- Grafiki tylko własne lub z licencją wpisaną do karty projektu; bez zdjęć osób bez ich zgody.

Przebieg zadania

1 Wspólny szkielet (cały zespół, 20 min)

Najpierw jedna osoba przygotowuje wspólny szablon strony, wysyła go, a pozostali go pobierają — dopiero potem praca równoległa.

1. H1. Utwórzcie szablon: `<!DOCTYPE html>`, `lang="pl"`, `meta charset` i `viewport`, unikatowy `title` i `meta description`, link do `css/styl.css`, nagłówek z logo i menu (lista odnośników), `main`, stopka. Zatwierdźcie i wyślijcie (`git push`).
2. H2. Każdy pobiera szablon (`git pull`) i tworzy swoją gałąź, np. `git switch -c strona-filmy`.

2 Podstrony (praca równoległa)

Treści mają być prawdziwe dla waszego tematu — bez „lorem ipsum”.

1. H3. Każda podstrona z mapy strony ma treść zgodną z makietą: jeden `h1`, logiczne `h2 / h3`, sekcje semantyczne (`section`, `article`, `aside`), aktywna pozycja menu oznaczona klasą i `aria-current="page"`.
2. H4. W projekcie musi wystąpić co najmniej: tabela z `caption`, `thead` i `th scope`; formularz z `fieldset`, `legend`, etykietami `label for` i walidacją natywną (`required`, `type="email"`, `minlength`); lista; co najmniej 4 grafiki z `alt`, `width` i `height`.
3. H5. Przygotujcie grafiki w GIMP: wymiary dopasowane do miejsca na stronie (np. plakaty 300 × 450 px), każdy plik poniżej 150 KB, nazwy małymi literami bez polskich znaków i spacji. Dodajcie favicon.
4. H6. Każdy członek zespołu wykonuje co najmniej 3 commity z opisowymi komunikatami (np. „Tabela harmonogramu seansów”, nie „zmiany”).

3 CSS — mobile first

Jeden wspólny arkusz `css/styl.css`. Style bazowe piszcie dla telefonu, a szersze ekrany obsłużcie regułami `@media (min-width: ...)`.

1. C1. Zdefiniujcie kolory i odstępy jako zmienne CSS w `:root` (paleta z karty projektu) i używajcie ich w całym arkuszu.
2. C2. Układ: menu zwijane bez JavaScript na telefonie (checkbox + etykieta lub `details`), poziome od szerokości desktopowej; sekcje kart w Gridzie lub Flexboksie — 1 kolumna na telefonie, 2 na tablecie, 3–4 na desktopie, zgodnie z makietą.
3. C3. Tabela na wąskim ekranie przewija się wewnątrz kontenera (`overflow-x: auto`), a strona nie przewija się w poziomie. Obrazy nie wychodzą poza kontener (`max-width: 100%`).
4. C4. Dostępność i efekty: widoczny `:focus-visible`, `:hover` na linkach i przyciskach z `transition`, kontrast zgodny z kartą; reguła `prefers-reduced-motion` wyłączająca animacje.

4 Scalanie i kontrola

1. K1. Scalcie gałęzie do `main` (rozwiązując ewentualne konflikty) i wyślijcie na GitHub. Wyświetlcie `git log --oneline --graph --all` i zróbcie zrzut.
2. K2. Sprawdźcie każdą podstronę walidatorem W3C HTML oraz arkusz walidatorem CSS; poprawcie błędy (ostrzeżenia zapiszcie).
3. K3. W trybie urządzeń F12 sprawdźcie szerokości 360, 768 i 1280 px — zrzuty strony głównej w trzech szerokościach porównajcie z makietami. Zapiszcie różnice i ich powód.

W sprawozdaniu umieść

- ☐ lista podstron z autorem każdej (kto co wykonał)
- ☐ wyniki walidacji HTML i CSS (zrzuty lub liczba błędów przed i po poprawkach)
- ☐ zrzuty strony głównej przy 360, 768 i 1280 px obok makiet

- ☐ zrzut grafu historii Git z commitami wszystkich członków
- ☐ adres repozytorium
- ☐ odpowiedzi na pytania kontrolne (każdy osobno)

Pytania kontrolne

1. Na czym polega podejście mobile first w CSS i jak zapisujecie reguły dla większych ekranów?
2. Do czego służą zmienne CSS i jaką korzyść dały w waszym projekcie?
3. Jak zrobiliście menu zwijane bez JavaScript i jak działa ono z klawiatury?
4. Dlaczego obrazy mają atrybuty `width` i `height`, mimo że ich rozmiar ustala CSS?

Kryteria oceny

Kryterium	Punkty
H1-H3 — wszystkie podstrony z mapy, wspólny szablon, semantyka, aktywne menu	4
H4-H5 — tabela, formularz z walidacją natywną, lista, min. 4 grafiki z alt i wymiarami, pliki < 150 KB	4
C1-C4 — zmienne CSS, mobile first (1/2/3-4 kolumny), menu bez JS, brak przewijania w poziomie, focus i reduced motion	5
K2-K3 — walidacja HTML i CSS bez błędów, porównanie z makietami	2
H6, K1 — min. 3 opisowe commity każdej osoby, scalone gałęzie, graf historii	3
Indywidualne odpowiedzi na pytania kontrolne	2
Razem	20

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