Dokumentacja Projektu: 5 najpiękniejszych plaż na świecie

# 1. Projekt informatyczny – od strony biznesowej i technologicznej

Cel biznesowy:  
- Stworzenie atrakcyjnej wizualnie, statycznej strony internetowej o tematyce podróżniczej,  
- Zwiększenie zainteresowania ofertami turystycznymi,  
- Posłużenie jako materiał promocyjny dla bloga podróżniczego lub agencji turystycznej,  
- Zapewnienie angażującego doświadczenia użytkownika.  
  
Cel technologiczny:  
- Wdrożenie responsywnej, lekkiej i estetycznej strony wykorzystującej HTML, CSS i JavaScript.

# 2. Badanie potrzeb i analiza

Analiza użytkowników:  
- Grupa docelowa: osoby szukające inspiracji podróżniczych,  
- Potrzeby: dostęp do atrakcyjnych treści, intuicyjny interfejs, prostota nawigacji.  
  
Wnioski:  
- Podział treści na sekcje,  
- Optymalizacja obrazów,  
- Proste interakcje użytkownika (toggle opisu).

# 3. Planowanie projektu informatycznego

Etapy realizacji:  
1. Analiza potrzeb i wymagań,  
2. Wybór technologii,  
3. Projekt graficzny,  
4. Kodowanie,  
5. Testowanie,  
6. Publikacja,  
7. Zbieranie opinii.  
  
Zasoby:  
- Czas: 1–2 tygodnie,  
- Osoby: 1 front-end developer , 1 back-end developer, 1 recenzent.

# 4. Projekt informatyczny i specyfikacja techniczna

Struktura plików:  
- index.html  
- css.css  
- /images (plaza1.jpg, plaza2.webp, ...)  
  
Technologie:  
- HTML5, CSS3, JS (Vanilla),  
- Google Fonts – Inter,  
- Format obrazów: .jpg, .webp, .avif.  
  
Interaktywność:  
- Prosty toggle opisu po kliknięciu na obrazek.

# 5. Jak wybrać software house do współpracy?

Kryteria wyboru:  
- Doświadczenie w stronach contentowych,  
- Portfolio UX/UI,  
- Transparentność,  
- Jakość kodu.

# 6. Wybór odpowiedniej technologii do projektu

Technologie:  
- Frontend: HTML5, CSS3, JS,  
- Fonty: Google Fonts – Inter,  
- Media: JPG / WebP / AVIF.  
  
Alternatywy:  
- React / Vue dla dynamicznych projektów,  
- Tailwind CSS dla szybkiego prototypowania.

# 7. Zaangażowanie i komunikacja

Model komunikacji:  
- Teams / Discord,- Trello, Jira lub Notion.

# 8. Testowanie kodu aplikacji i narzędzia

Narzędzia:  
- Lighthouse,  
- W3C Validator,  
- Responsively App,  
- Prettier / ESLint.  
  
Testy JS (opcjonalnie):  
- Możliwość użycia Jest lub Mocha.

# 9. Metoda opinii – recenzja ekspercka

„Strona jest bardzo przejrzysta, zdjęcia robią duże wrażenie. Dobrze dobrana kolorystyka – działa na emocje.”  
— Recenzent, Jakub P.

# 10. Typy testów oprogramowania przeprowadzone

- Test funkcjonalny  
- Test wydajnościowy  
- Test responsywności  
- Test wizualny  
- Test semantyczny