

Analiza systemu wspomagającego zarządzanie zdjęciami

Etap III | Grupa nr.4

Produkt etapu	Autor
Projekt architektury systemu	Tomasz Jarnutowski
Projekt algorytmów	
Projekt interfejsu użytkownika	Emilian Bochenek
Projekt testów kontrolnych	Tomasz Jarnutowski

Spis treści

1. Projekt architektury systemu	3
2. Projekt algorytmów	4
3. Projekt interfejsu użytkownika	4
4. Projekt testów kontrolnych	8
4.1. Test zdjęć	8
4.1.1. Dodawanie zdjęć	8
4.1.2. Usuwanie zdjęć.....	8
4.1.3. Edycja zdjęcia.....	9
4.2. Test autoryzacja	9
4.2.1. Logowanie	9
4.2.2. Rejestracja.....	9
4.2.3. Wylogowanie	9
4.3. Test dodawania do ulubionych.....	9
4.3.1. Dodawanie zdjęcia do ulubionych	9
4.3.2. Dodawanie albumu do ulubionych	9
4.4. Test albumów	10
4.4.1. Dodawanie albumu	10
4.4.2. Usuwanie albumu	10
4.4.3. Edytowanie albumu.....	10

1. Projekt architektury systemu



Architektura jest zgodna z konwencją frameworku django. Gdy użytkownik wysyła żądanie, serwer weryfikuje adres oraz odsyła do odpowiedniego serializera. Serializer zajmuje się weryfikacją uprawnień oraz przygotowaniem zapytania do bazy danych. Gdy żądanie okaże się prawidłowe, a baza danych odeśle wynik – wtedy serializer przygotowuje odpowiedź i przekaże ją do backendu, a ten do użytkownika (czyli frontendu). W przypadku błędnej struktury danych bądź braku uprawnień – serializer wysyła do backendu kod błędu oraz treść

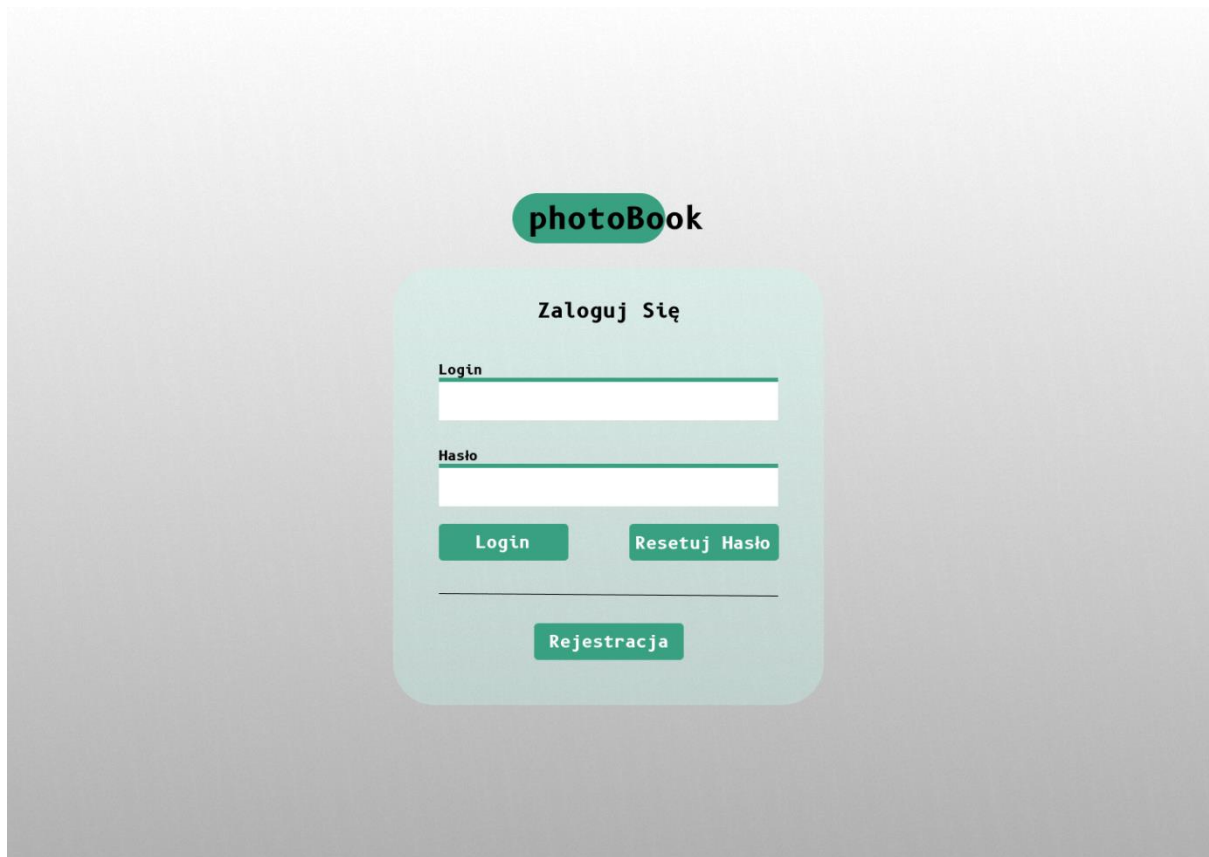
2. Projekt algorytmów

3. Projekt interfejsu użytkownika

1. Strona główna



2. Ekran Logowania



The image shows a login screen for an application named "photoBook". The background is a light gray gradient. At the top center, the "photoBook" logo is displayed in white text on a dark teal rounded rectangle. Below the logo is a light teal rounded rectangle containing the login form. The form is titled "Zaloguj Się" in bold black text. It features two input fields: "Login" and "Hasło", both with white text on a light gray background and a thin teal border. Below the "Hasło" field are two teal buttons with white text: "Login" and "Resetuj Hasło". A horizontal line separates these from a single teal button with white text labeled "Rejestracja" at the bottom.

photoBook

Zaloguj Się

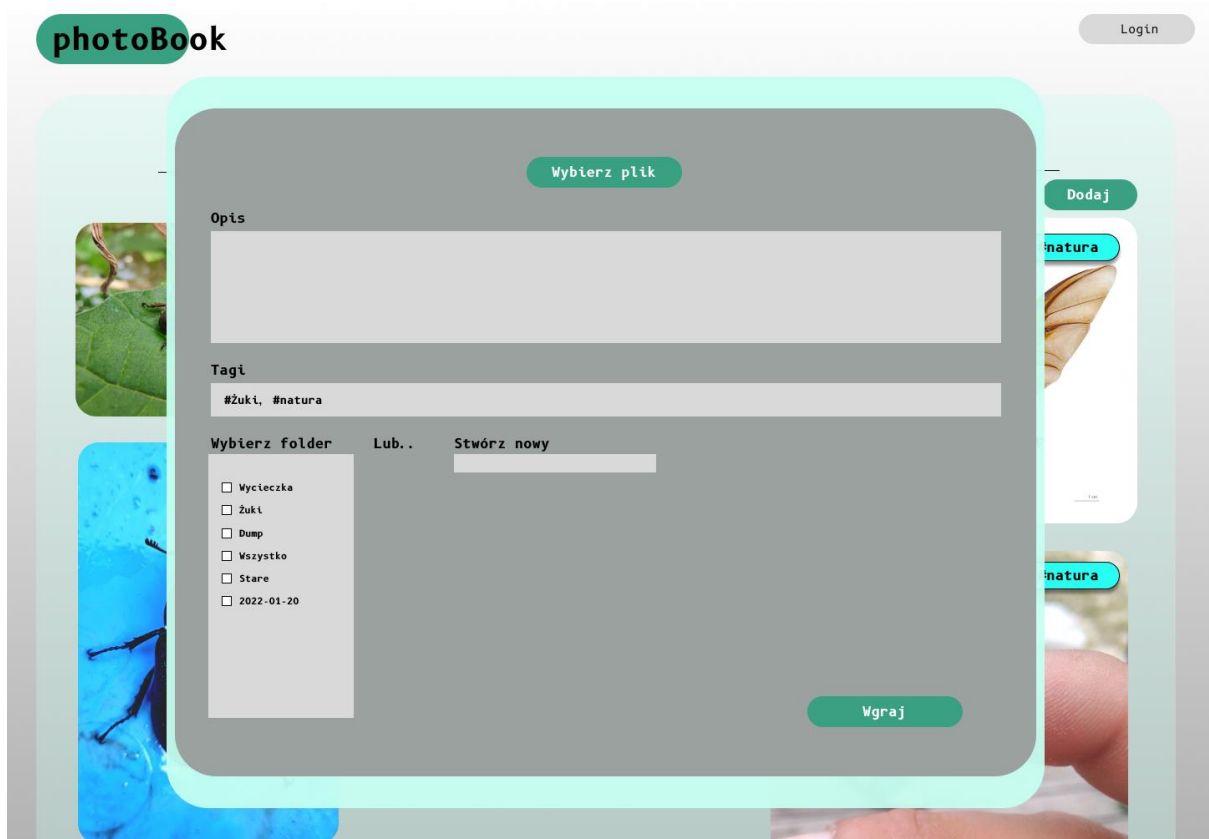
Login

Hasło

Login Resetuj Hasło

Rejestracja

3. Edycja Zdjęcia



4. Album

photoBook

Login

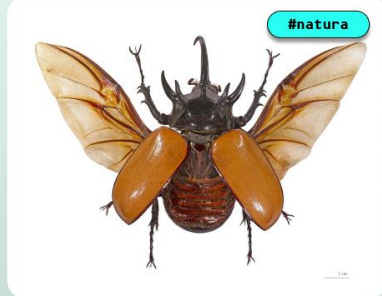
Twoja Galeria



#natura



#natura



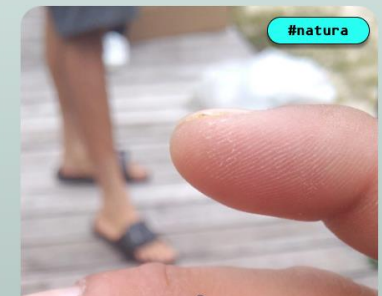
#natura



#natura




#natura



#natura

5. Rejestracja



The image shows a registration form for a service called "photoBook". The form is titled "Rejestracja" and is set against a light green background. It contains several input fields: "Login", "Email", "Hasło" (Password), and "Powtórz hasło" (Repeat password). Below these fields are three checkboxes, each followed by the text "I agree that I have read and accepted the Terms of Use and Privacy Policy." At the bottom of the form is a green button labeled "Rejestracja".

4. Projekt testów kontrolnych

Testy backendu będą wykonane za pomocą frameworka django i modułu wbudowanego w pythona jakim jest „unittest”, natomiast do frontendu „React Testing Library”. Poniżej opiszę wybrane testy na backendzie

4.1. Test zdjęć

4.1.1. Dodawanie zdjęć

- Sprawdzenie czy zdjęcie się dodało
- Sprawdzenie czy autor zdjęcia został dodany prawidłowo
- Przetestuje czy wysokość, szerokość i data dodania została dodana automatycznie
- Pobiorę wszystkie zdjęcia użytkownika i sprawdzę czy dodane zdjęcie jest zwracane
- Upewnię się że publiczne zdjęcie jest rzeczywiście publiczne, a niepubliczne – prywatne

4.1.2. Usuwanie zdjęć

- Sprawdzenie czy pole deleted zostało zmienione na true
- Sprawdzenie czy data usunięcia jest prawidłowa

- Sprawdzenie czy użytkownik ma uprawnienia aby usunąć zdjęcie, i czy system zwraca odpowiednie kody odpowiedzi
- Sprawdzenie czy przywrócenie zdjęcia zmienia pole deleted na false
- Sprawdzenie czy permanentnie usunięcie usuwa zdjęcie

4.1.3. Edycja zdjęcia

- Sprawdzenie czy użytkownik posiada prawo do edycji zdjęcia
- Sprawdzenie czy zdjęcie i dane zostały edytowane pomyślnie
- Sprawdzenie czy data aktualizacji została zaktualizowana
- Sprawdzenie czy zmiana dostępności zdjęcia działa prawidłowo

4.2. Test autoryzacja

4.2.1. Logowanie

- Sprawdzenie czy użytkownik istnieje bądź nie istnieje i odesłanie adekwatnej odpowiedzi
- Sprawdzenie czy token zostanie utworzony

4.2.2. Rejestracja

- Sprawdzenie czy utworzy konto z duplikowanym emailiem bądź loginem
- Sprawdzenie czy poprawnie zweryfikuje hasło
- Sprawdzenie czy dana utworzenia jest poprawna
- Sprawdzenie czy uprawnień utworzonego konta
- Sprawdzenie czy został utworzony token

4.2.3. Wylogowanie

- Sprawdzenie czy token został usunięty

4.3. Test dodawania do ulubionych

4.3.1. Dodawanie zdjęcia do ulubionych

- Sprawdzenie czy zdjęcie zostało dodane do ulubionych
- Sprawdzenie czy zdjęcie jest zwracane w odpowiedzi w lista polubionych zdjęć

4.3.2. Dodawanie albumu do ulubionych

- Sprawdzenie czy album został dodany do ulubionych
- Sprawdzenie czy album jest zwracane w odpowiedzi w liście polubionych albumów

4.4. Test albumów

4.4.1. Dodawanie albumu

- Sprawdzenie czy album się utworzył
- Sprawdzenie czy autor albumu został dodany prawidłowo
- Dodanie zdjęć do albumu i sprawdzenie czy dodane zdjęcie są zwracane

4.4.2. Usuwanie albumu

- Sprawdzenie czy usunięcie albumu nie spowoduje usunięcia zdjęć
- Sprawdzenie czy permanentnie usunięcie usuwa album
- Sprawdzenie czy data usunięcia jest prawidłowa
- Sprawdzenie czy użytkownik ma uprawnienia aby usunąć album, i czy system zwraca odpowiednie kody odpowiedzi

4.4.3. Edytowanie albumu

- Sprawdzenie czy użytkownik posiada prawo do edytowania albumu
- Sprawdzenie czy zdjęcia i dane albumu zostały edytowane pomyślnie
- Sprawdzenie czy data aktualizacji została zaktualizowana