

**UNIWERSYTET RZESZOWSKI**

**Kolegium Nauk Przyrodniczych**

Łukasz Domin

Nr albumu: 123631

Informatyka I stopnia

**Aplikacja internetowa do zarządzania ogrodem z elementami społecznościowymi**

Praca inżynierska

Praca wykonana pod kierunkiem

….....……………………………………..

Rzeszów, data

Spis treści

[1 Wstęp 4](#_Toc187930830)

[2 Cel i zakres pracy 5](#_Toc187930831)

[3 Funkcjonalności 6](#_Toc187930832)

[3.1 Narzędzia i technologie użyte w pracy 6](#_Toc187930833)

[3.1.1 Narzędzia użyte do części frontendowej 6](#_Toc187930834)

[3.1.2 Narzędzia użyte do części backendowej 6](#_Toc187930835)

[3.2 Diagram Przypadków Użycia 8](#_Toc187930836)

[3.3 Diagram Architektura klient-serwer 11](#_Toc187930837)

[4 Implementacja aplikacji 11](#_Toc187930838)

[4.1 Scrapper 11](#_Toc187930839)

[4.1.1 Pobranie roślin 12](#_Toc187930840)

[4.1.2 Pobranie obrazów roślin 12](#_Toc187930841)

[4.2 Struktura bazy danych 13](#_Toc187930842)

[4.2.1 Źródła danych 14](#_Toc187930843)

[4.2.2 Węzły, etykiety i relacje bazy danych 14](#_Toc187930844)

[4.3 Struktura backendu aplikacji 24](#_Toc187930845)

[4.3.1 Przykładowa klasa modelu danych 26](#_Toc187930846)

[4.3.2 Przykład procesu żądania i odpowiedzi 28](#_Toc187930847)

[4.3.3 Logowanie zdarzeń i błędów 31](#_Toc187930848)

[4.3.4 Zadania okresowe w tle 32](#_Toc187930849)

[4.3.5 Seedery 32](#_Toc187930850)

[4.4 Struktura frontendu aplikacji 33](#_Toc187930851)

[4.4.1 Przykładowy komponent 33](#_Toc187930852)

[4.4.2 Przykład metody API z serwisu 35](#_Toc187930853)

[5 Opis funkcjonalności systemu 36](#_Toc187930854)

[5.1 Funkcjonalności zwykłego użytkownika 36](#_Toc187930855)

[5.1.1 Rejestracja 36](#_Toc187930856)

[5.1.2 Zarządzanie kontem 37](#_Toc187930857)

[5.1.3 Przeglądanie roślin w katalogu 40](#_Toc187930858)

[5.1.4 Zarządzanie własnymi roślinami 42](#_Toc187930859)

[5.1.5 Zarządzanie postami i komentarzami 42](#_Toc187930860)

[5.1.6 Przeglądanie użytkowników 46](#_Toc187930861)

[5.1.7 Zarządzanie ogrodem 46](#_Toc187930862)

[5.1.8 Zarządzanie rozmowami 51](#_Toc187930863)

[5.1.9 Zarządzanie powiadomieniami 53](#_Toc187930864)

[5.2 Funkcjonalności pracownika 55](#_Toc187930865)

[5.2.1 Zarządzanie roślinami 55](#_Toc187930866)

[5.2.2 Banowanie użytkowników 58](#_Toc187930867)

[5.2.3 Tworzenie specjalnego powiadomienia 58](#_Toc187930868)

[5.3 Funkcjonalności administratora 59](#_Toc187930869)

[5.3.1 Tworzenie konta pracownika 59](#_Toc187930870)

[5.3.2 Banowanie pracowników 59](#_Toc187930871)

[5.3.3 Wysyłanie powiadomień do pracowników 59](#_Toc187930872)

[5.4 Zarządzanie zasobami 59](#_Toc187930873)

[6 Testy 59](#_Toc187930874)

[7 Podsumowanie 61](#_Toc187930875)

[8 Literatura 62](#_Toc187930876)

## Wstęp

Aplikacja internetowa pt. „Wirtualny Ogród Yukka” jest systemem umożliwiającym zarządzanie wirtualnym ogrodem, wymianę wiedzy i doświadczeń pomiędzy użytkownikami w przestrzeni społecznościowej.

Tworzona aplikacja dostarcza użytkownikom intuicyjne rozwiązania, takie jak system powiadomień o społecznościowych (np. odpowiedź na komentarz, otrzymanie zaproszenia) lub stricte ogrodowych (np. powiadomienie o okresie owocowania roślin) w ogrodzie, możliwość projektowania własnych działek z roślinami wybranymi z katalogu, które można następnie dostosować do swoich upodobań. Możliwe jest również stworzenie własnej rośliny we własnym katalogu.

Rośliny oficjalne lub własne mogą być posadzone na swojej działce reprezentowanej w formie siatki kafelków, na których znajduje się roślina oraz obszar jej przydzielony pokolorowany wybranym kolorem, albo z nałożoną przez użytkownika teksturą. Sam widok działki wraz z zasadzonymi na jej roślinami może być udostępniony innym użytkownikom.

Dodatkowo, poprzez integrację części społecznościowej możliwa jest interakcja z innymi użytkownikami. Jednym z tych sposobów jest tworzenie postów na różne tematy, które można oceniać lub komentować. Same komentarze też mogą być oceniane i komentowane, pozwalając na aktywną dyskusję pomiędzy członkami aplikacji. W razie, gdyby dwójka użytkowników chciała porozmawiać na cztery oczy, możliwe jest prowadzenie rozmowy prywatnej. W razie, gdyby jakiś użytkownik zachowywał się niepoprawnie, możliwe jest zgłoszenie go przez innych użytkowników i późniejsze zbanowanie go przez pracowników. Dzięki temu system wspiera zarówno początkujących, jak i zaawansowanych ogrodników, tworząc przyjazne środowisko do dzielenia się swoją pasją.

## Cel i zakres pracy

Celem aplikacji jest stworzenie interaktywnego narzędzia dla ogrodników, które umożliwia projektowanie własnych wirtualnych ogrodów, zarządzanie roślinami, otrzymywanie powiadomień o zabiegach pielęgnacyjnych bądź o wydarzeniach społecznościowych oraz dzielenie się doświadczeniem i wiedzą z innymi użytkownikami.

Aplikacja ma na celu ułatwienie planowania i pielęgnacji ogrodu, jednocześnie wspierając społeczność ogrodników, pozwalając na łatwy dialog pomiędzy sobą.

**Zakres pracy**

* Opracowanie struktury bazy danych
  + Stworzenie modelu grafowej bazy danych, który uwzględnia wszystkie potrzebne funkcje aplikacji
  + Ustalenie relacji między węzłami
  + Napisanie scrappera zbierającego dane użycia w bazie danych
* Projektowanie wstępnych widoków aplikacji
  + Opracowanie schematów interfejsu użytkownika oraz ustawanie sztywnych danych
  + Tworzenie backendu aplikacji
  + Implementacja serwerowej strony aplikacji
  + Zaimplementowanie API do komunikacji z frontendem
* Tworzenie frontendu aplikacji
  + Tworzenie interfejsu użytkownika, który umożliwia interaktywne korzystanie z aplikacji
  + Łączenie backendu z frontendem
  + Połączenie API z interfejsem użytkownika, umożliwiająca komunikację między frontendem a backendem
  + Testowanie i poprawa błędów
  + Przeprowadzenie testów aplikacji, sprawdzające poprawne działanie jej elementów
  + Znajdowanie i poprawa ewentualnych błędów
  + Pisanie pracy inżynierskiej

## Funkcjonalności

### Narzędzia i technologie użyte w pracy

#### Narzędzia użyte do części frontendowej

**TypeScript 5.5.2** - Rozszerzenie języka JavaScript dodające typowanie statyczne, klasy oraz interfejsy. Używany w Angularze do programowania części frontendowej. Link: <https://www.typescriptlang.org/download/>

**Angular 18.2.11[[1]](#footnote-1)** - Framework oparty na języku TypeScript do budowy aplikacji webowych. Posiada mechanikę sygnałów, wsparcie dla SSR (rendererowania po stronie serwera), hydracji (użycie statycznej części wygenerowanej strony przez SSR), oraz lepsza obsługa ładowania asynchronicznego. (Joshi, 2024). Link: <https://v18.angular.dev/installation>

**Bootstrap 5** - biblioteka CSS. Zawiera zestaw narzędzi ułatwiających tworzenie interfejsu aplikacji internetowych. Bazuje na gotowych rozwiązaniach HTML oraz CSS i może być stosowany m.in. do stylizacji poszczególnych elementów strony. Link: <https://getbootstrap.com>

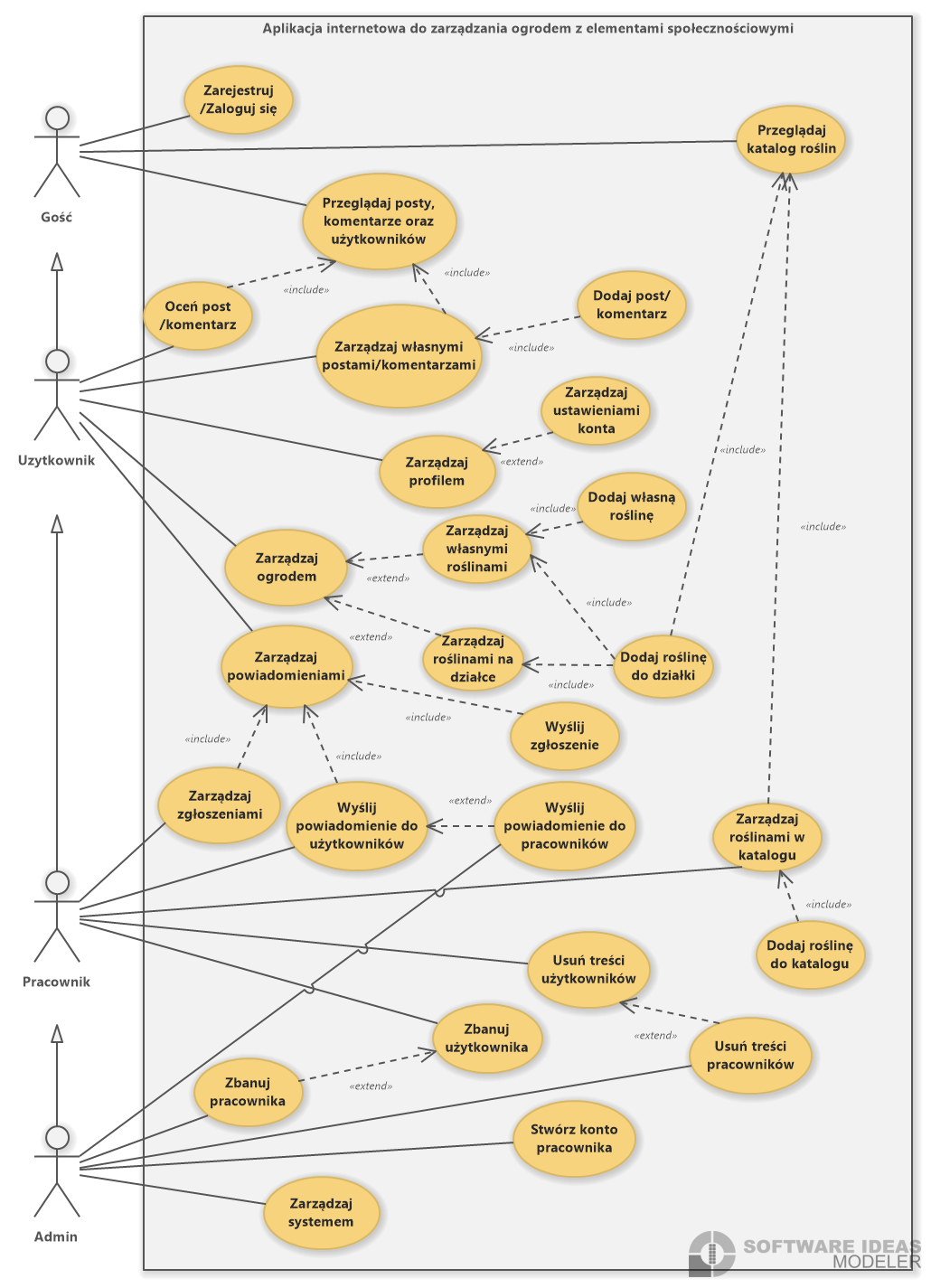
**Npm (Node Package Manager**) - Menedżer pakietów dla JavaScriptu, dostarczany wraz z Node.js. Służy do zarządzania bibliotekami oraz narzędziami JavaScript. Link: <https://docs.npmjs.com/downloading-and-installing-node-js-and-npm>

#### Narzędzia użyte do części backendowej

**Java 17[[2]](#footnote-2)** - Język obiektowy działający na różnych systemach operacyjnych bez większych trudów. Jest szeroko stosowana do tworzenia przeróżnych aplikacji czy to internetowych, czy to mobilnych (Kathy Sierra, 2023). Link: <https://www.oracle.com/java/technologies/javase/jdk17-archive-downloads.html>

**Spring Boot 3.3.2[[3]](#footnote-3)** - Framework języka Java, który upraszcza proces konfiguracji i uruchamiania aplikacji, automatycznie konfigurując większość komponentów, skracając kod przy użyciu adnotacji, a także upraszcza komunikację z bazą danych i frontendem (Greg L. Turnquist, 2022). Link: <https://docs.spring.io/spring-boot/installing.html>

### Diagram Przypadków Użycia



Rysunek 1 Diagram Przypadków Użycia

**Definicja aktorów**

**AKTOR:** Gość

**OPIS:** Gość może przeglądać katalog roślin, posty, komentarze, profile użytkowników i może się zarejestrować lub zalogować.

**PRZYPADKI UŻYCIA:**

* PU Zarejestruj/Zaloguj się
* PU Przeglądaj posty, komentarze oraz użytkowników
* PU Przeglądaj katalog roślin

**AKTOR:** Użytkownik

**OPIS:** Użytkownik oprócz funkcjonalności gościa oceniać posty i komentarze oraz tworzyć własne i nimi zarządzać. Może również zarządzać kontem, swoimi powiadomieniami roślinami własnymi, ogrodem i działkami wewnątrz niego, a także zgłaszać innych użytkowników.

**PRZYPADKI UŻYCIA:**

* PU Zarządzaj własnymi postami/komentarzami
* PU Oceń post/komentarz powiązane przez <<include>> z PU Przeglądaj posty, komentarze oraz użytkowników
* PU Zarządzaj własnymi postami/komentarzami powiązane przez <<include>> z PU Przeglądaj posty, komentarze oraz użytkowników
* PU Dodaj post/ komentarz powiązane przez <<include>> z PU Zarządzaj własnymi postami/komentarzami
* PU Zarządzaj profilem
* PU Zarządzaj ustawieniami konta powiązane przez <<extend>> z PU Zarządzaj profilem
* PU Zarządzaj ogrodem
* PU Zarządzaj własnymi roślinami powiązane przez <<extend>> z PU Zarządzaj ogrodem
* PU Dodaj własną roślinę powiązane przez <<include>> z PU Zarządzaj własnymi roślinami
* PU Zarządzaj roślinami na działce powiązane przez <<extend>> z PU Zarządzaj ogrodem
* PU Dodaj roślinę do działki powiązane przez <<include>> z PU Zarządzaj roślinami na działce, PU Zarządzaj własnymi roślinami oraz PU Przeglądaj katalog roślin
* PU Zarządzaj powiadomieniami
* PU Wyślij zgłoszenie powiązane przez <<include>> z PU Zarządzaj powiadomieniami

**AKTOR:** Pracownik

**OPIS:** Pracownik oprócz funkcjonalności użytkownika zarządza roślinami w katalogu, usuwa niestosowne treści, przegląda zgłoszenia i banuje.

**PRZYPADKI UŻYCIA:**

* PU Zarządzaj zgłoszeniami powiązane przez <<include>> z PU Zarządzaj powiadomieniami
* PU Wyślij powiadomienie do użytkowników powiązane przez <<include>> z PU Zarządzaj powiadomieniami
* PU Zarządzaj roślinami w katalogu powiązane przez <<include>> z PU Przeglądaj katalog roślin
* PU Dodaj roślinę do katalogu powiązane przez <<include>> z PU Zarządzaj roślinami w katalogu
* PU Usuń treści użytkowników
* PU Zbanuj użytkownika

**AKTOR:** Administrator

**OPIS:** Administrator oprócz funkcjonalności pracownika zarządza systemem oraz pracownikami, wysyła im powiadomienia, usuwa ich treści, banuje, a także tworzy nowe konta pracowników.

**PRZYPADKI UŻYCIA:**

* PU Wyślij powiadomienie do pracowników powiązane przez <<extend>> z PU Wyślij powiadomienie do użytkowników
* PU Zbanuj pracownika powiązane przez <<include>> z PU Zbanuj użytkownika
* PU Usuń treści pracowników powiązane przez <<extend>> z PU Usuń treści użytkowników
* PU Stwórz konto pracownika
* PU Zarządzaj systemem

**PU ZAREJESTRUJ/ZALOGUJ SIĘ**

**OPIS**

**CEL:** ZALOGOWANIE DO SYSTEMU

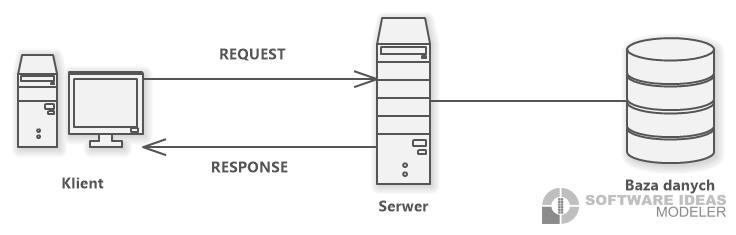
**WS:** może być wywołany z PU Załóż konto/Rejestruj

**WK:** podanie niezbędnych danych do logowania tj. login i hasło

**PRZEBIEG:**

Do logowania należy podać e-mail/nazwę użytkownika i hasło. W razie błędu wyświetla komunikat. Po poprawnym logowaniu użytkownik trafia na stronę główną. Rejestracja wymaga podania nazwy, e-maila, hasła i jego potwierdzenia. Jeśli e-mail lub nazwa jest zajęta, pojawia się stosowny komunikat. Po poprawnej rejestracji na e-mail wysyłany jest kod weryfikacyjny do aktywacji konta.

### Diagram Architektura klient-serwer



Rysunek 2 Diagram Architektura klient-serwer

Diagram (zob. Rysunek 2 Diagram Architektura klient-serwer) przedstawia architekturę aplikacji internetowej modelu klient-serwer. Składa się z trzech warstw:

**Klient (Frontend):**

Klient działa na aplikacji Angular w przeglądarce, która komunikuje się z serwerem backendowym przy użyciu żądań HTTP, wysyłając je do REST API i otrzymując odpowiedzi.

**Serwer (Backend):**

Backend będący projektem Spring Boot. Odpowiada za obsługę żądań z frontendu i komunikację z bazą danych, wysyłając odpowiedzi do klienta.

**Baza danych:**

Baza grafowa Neo4j.Przyjmuje zapytania od frontendu w języku CYPHER, odczytując lub modyfikując dane i zwracając je.

## Implementacja aplikacji

### Scrapper

Jest to scrapper napisany w języku Python używany do pobierania danych o roślinach ze strony <https://e-katalogroslin.pl> do pliku katalog\_roslin.csv. Wykorzystuje sterownik Firefox, wielowątkowość oraz asynchroniczność do szybszego pobierania danych ze stron.

#### Pobranie roślin

Aby rozpocząć proces pobierania roślin, należy uruchomić plik scrap\_threads.py pobierający dane od początkowej strony aż do ostatniej. Uwaga: czasami program może mieć trudności z dostępnością do strony, przejściem na inną stronę oraz inne czynniki zewnętrzne które mogą zakłócić bądź uniemożliwić działanie. Czas: ok. 50-70 minut.

#### Pobranie obrazów roślin

Plik katalog\_roslin.csv zawiera informacje o roślinach takie jak nazwa rośliny, nazwa łacińska, opis, rodzaj gleby, a także link do obrazu rośliny (jeśli takowy posiada). Aby pobrać obrazy, należy uruchomić plik *download\_images.py,* który iteruje przez kolejne rzędy pliku katalog\_roslin.csv i zapisuje obrazy do folderu *images.* Czas: ok. 5 minut.

W kodzie (Rysunek 3) wykonywana jest pętla scrappowania kolejnych stron. Funkcja get\_all\_plant\_info pobiera informacje o roślinach z listy adresów URL. Używa biblioteki Selenium do nawigacji po stronach internetowych i oczekiwania na załadowanie elementów. Po przetworzeniu strony próbuje przejść do następnej. Po zebraniu informacji zwraca je jako listę. Na koniec zamyka przeglądarkę. Ta funkcja jest przydzielana dla każdego wątku.



Rysunek 3 Funkcja get\_all\_plant\_info z pliku scrap\_threads.py

### Struktura bazy danych

**Neo4j 5.12.0[[4]](#footnote-4)** – Jest to grafowa baza danych, która przechowuje dane w postaci węzłów, ich etykiet, relacji pomiędzy nimi oraz właściwościami węzłów bądź relacji. Została ona wybrana dlatego, że ten typ bazy danych jest polecany w budowaniu aplikacji społecznościowych ze względu na budowę relacji pomiędzy poszczególnymi elementami bazy (Bruggen, 2014).

#### Źródła danych

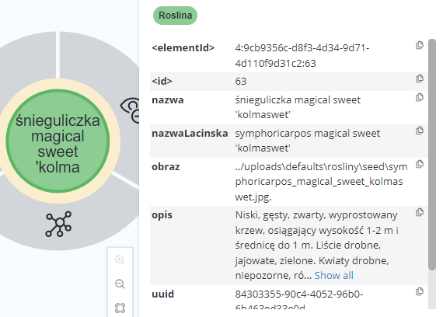
Źródłem danych jest strona *e-katalogroslin.pl* należąca do Związku Szkółkarzy Polskich będący stowarzyszeniem producentów drzew, krzewów i innych roślin. Strona jest katalogiem zawierającym ponad 10 000 roślin o różnych właściwościach takich jak rodzaj gleby, zimozieloność liści, zakres wysokości itp. Link: <https://e-katalogroslin.pl>

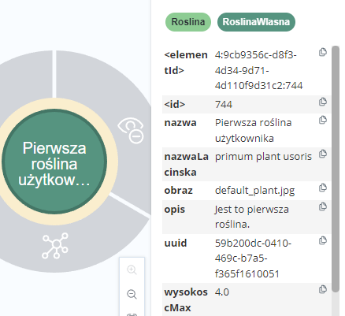
#### Węzły, etykiety i relacje bazy danych

##### Węzły

###### Roslina, RoslinaWlasna

Roslina (Rysunek 4) jest jednym z głównym węzłów w bazie danych. Reprezentuje roślinę z katalogu oraz roślinę użytkownika i jest wykorzystywana w ogrodzie. Rośliny własne nienależące do oficjalnego katalogu posiadają etykietę Roslina oraz RoslinaWlasna, a roślina z katalogu ma tylko etykietę Roslina.

**Właściwości:**

* uuid – Identyfikator rośliny
* nazwa – nazwa rośliny
* nazwaLacinska – nazwa łacińska rośliny
* opis – opis rośliny
* wysokoscMin, wysokoscMax – wysokość minimalna i maksymalna rośliny
* obraz – ścieżka do obrazu rośliny, jeśli takowy posiada

Rysunek 4 Węzeł Roslina i węzeł Roslina z etykietą RoslinaWlasna

###### Cecha

Rośliny posiadają 16 cech opisanych etykietami takimi jak *Owoc, Gleba,* które klasyfikują typ cechy. Zwykła cecha ma etykietę *Cecha*, a cecha stworzona przez użytkownika ma dodatkowo etykietę *CechaWlasna (*Rysunek 5*).*

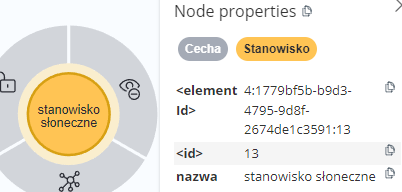
**Etykiety główne:** Cecha, CechaWlasna

**Etykiety dodatkowe:**

|  |  |  |
| --- | --- | --- |
| **Nazwa** | **Opis** | **Zdjęcie** |
| Forma | Forma rośliny |  |
| Gleba | Rodzaj gleby przyjazny dla rośliny |  |
| Grupa | Grupa rośliny |  |
| Kolor | Kolor kwiatów albo liści rośliny |  |
| Kwiat | Rodzaj kwiatu rośliny |  |
| Odczyn | Odczyn ph gleby przyjazny dla rośliny |  |
| Okres | Okres owocowania albo kwitnienia rośliny wyrażony w miesiącach |  |
| Owoc | Rodzaj owocu rośliny |  |
| Podgrupa | Podgrupa rośliny |  |
| Pokroj | Pokrój lub kształt rośliny |  |
| SilaWzrostu | Siła wzrostu rośliny, czyli jak szybko rośnie |  |
| Stanowisko | Odpowiednie nasłonecznienie rośliny |  |
| Wilgotnosc | Optymalna wilgotność gleby rośliny |  |
| Zastosowanie | Do czego pasuje roślina albo czy ma jakiś użytek poza wyglądem |  |
| Zimozielonosc | Jak zachowują się liście/igły rośliny podczas zimy, czy opadają, czy nie) |  |

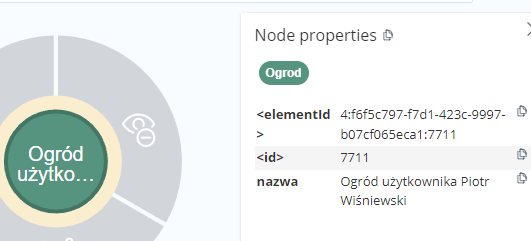
**Właściwości:**

* nazwa – nazwa cechy



Rysunek 5 Węzeł Cecha i węzeł Cecha z etykietą CechaWlasna

###### Ogrod

Ogród (Rysunek 6) jest węzłem tworzonym dla każdego użytkownika. Reprezentuje ogród podzielony na 10 węzłów Dzialka złączonych z nim.

**Właściwości:**

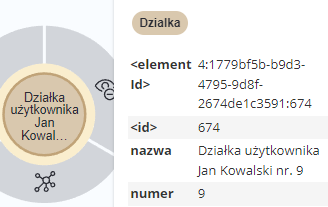
* nazwa – nazwa ogrodu

Rysunek 6 Węzeł ogrodu

###### Dzialka

Dzialka (

Rysunek 7) jest węzłem związanym z ogrodem, tworzony wraz z nim. Łącznie dla każdego użytkownika jest tworzone po 10 działek na ogród, gdzie każda działka ma wymiary 20x20 kafelków.

**Właściwości:**

* nazwa – nazwa działki
* numer – numer działki od 1 do 10

Rysunek 7 Węzeł Dzialka

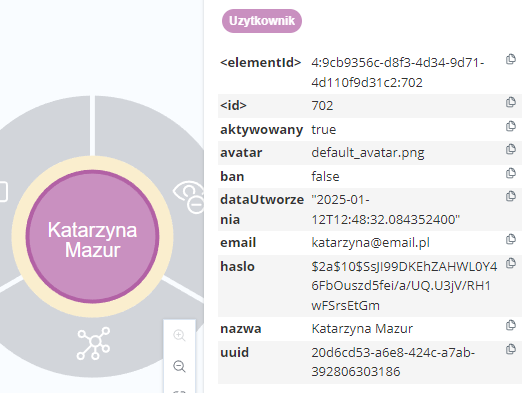
###### Uzytkownik, Pracownik, Admin

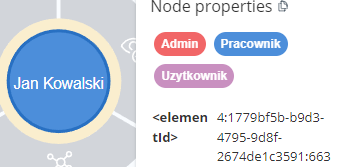
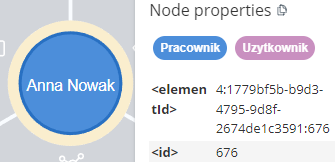
Użytkownik (Rysunek 8) jest węzłem pełniącym rolę użytkownika w bazie danych. Zwykli użytkownicy posiadają tylko etykietę Uzytkownik, za to pracownicy posiadają dodatkowo etykietę Pracownik, a administratorzy posiadają wszystkie z tych etykiet. Każda etykieta oznacza inny poziom uprawnień zdefiniowany w (Diagram Przypadków Użycia).

Dodatkowo, każdy użytkownik ma węzeł Ustawienia, węzeł Ogrod oraz 10 węzłów Dzialka przypisanych do tego ogrodu. Informacje o użytkowniku są w ten sposób rozdzielone, aby zoptymalizować pobór danych z bazy danych i pobierać tylko te części danych, które są potrzebne.

**Właściwości:**

* uuid – Identyfikator użytkownika; nazwa – nazwa użytkownika
* email – adres email użytkownika
* haslo – zaszyfrowane hasło użytkownika
* avatar – zdjęcie użytkownika
* dataUtworzenia – data utworzenia użytkownika
* aktywowany – oznacza, czy konto zostało aktywowane po rejestracji
* ban – czy użytkownik jest zbanowany
* banDo – data, do której użytkownik jest zbanowany
* banPowod – powód zbanowania
* imie – imię użytkownika na jego profilu
* nazwisko – nazwisko użytkownika na jego profilu
* miasto – miasto użytkownika na jego profilu
* miejsceZamieszkania – miejsce zamieszkania użytkownika na jego profilu
* opis – opis użytkownika na jego profilu





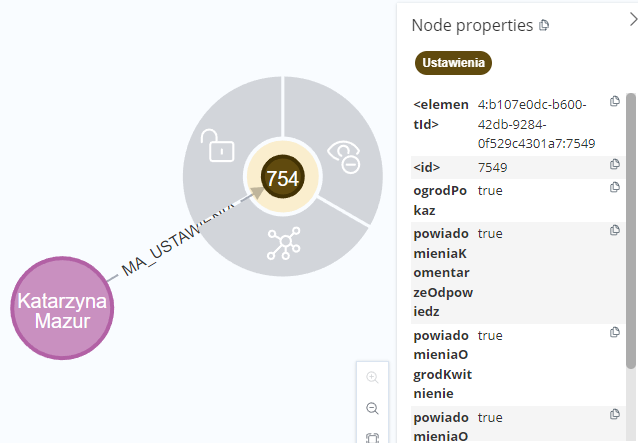
Rysunek 8 Węzeł Uzytkownik oraz węzły z etykietami Pracownik i Admin

###### Ustawienia

Ustawienia (Rysunek 9) definiują ustawienia użytkownika, w których decyduje się o powiadomieniach i widoczności swojego profilu.

**Właściwości:**

* ogrodPokaz – pozwala innym przeglądać ogród użytkownika (nie dotyczy pracowników i administratorów)
* powiadomieniaKomentarzeOdpowiedz – powiadomienia o odpowiedziach na komentarze
* powiadomieniaOgrodKwitnienie – powiadomienia o kwitnieniu roślin w ogrodzie
* powiadomieniaOgrodOwocowanie – powiadomienia o owocowaniu roślin
* statystykiProfilu – widoczność statystyk na profilu



Rysunek 9 Węzeł Ustawienia

###### Post

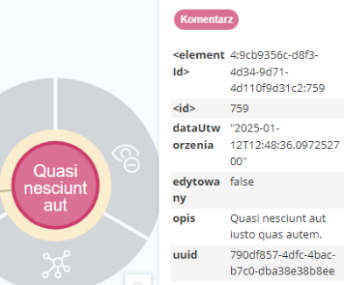
Węzeł Post (Rysunek 10) pełni rolę postu w części społecznościowej. Oprócz nazwy i opisu może również zawierać obraz. Do tego postu mogą być przypisywane komentarze użytkowników, a także może być oceniany.

**Właściwości:**

* uuid – Identyfikator postu
* tytul – tytuł postu
* opis – opcjonalny opis postu
* obraz – ścieżka do obrazu postu
* dataUtworzenia – data utworzenia postu

Rysunek 10 Węzeł Post

###### Komentarz

Węzeł Komentarz (Rysunek 11) pełni rolę komentarza w części społecznościowej. Są one używane do komentowania postów, odpowiadania na komentarze, a także służą jako wiadomości w rozmowach prywatnych.

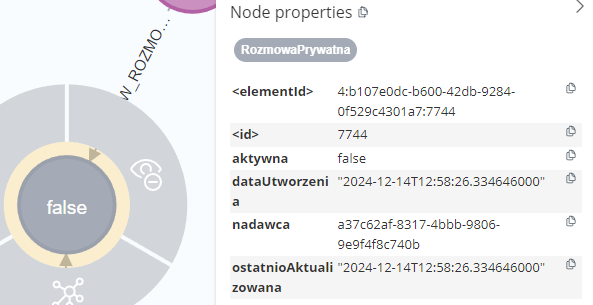
**Właściwości:**

* uuid – Identyfikator komentarza
* opis – treść
* obraz – ścieżka do obrazu
* edytowany –czy komentarz był edytowany
* dataUtworzenia – data utworzenia

Rysunek 11 Węzeł Komentarz

###### RozmowaPrywatna

Węzeł RozmowaPrywatna (Rysunek 12) pełni rolę rozmowy prywatnej pomiędzy dwoma użytkownikami w części społecznościowej. Użytkownik A musi zaprosić użytkownika B do rozmowy, a odbiorca musi ją zaakceptować. Wtedy możliwe jest pisanie wiadomości pomiędzy nimi.

**Właściwości:**

* aktywna –czy rozmowa prywatna jest aktywna (odbiorca zaakceptował zaproszenie)
* nadawca – identyfikator nadawcy
* dataUtworzenia – data utworzenia rozmowy
* ostatnioAktualizowana – data zmiany

Rysunek 12 Węzeł RozmowaPrywatna

###### Powiadomienie, Zgloszenie

Węzeł Powiadomienie (Rysunek 13) powiadania użytkownika o różnych zdarzeniach w systemie, takie jak odpowiedź na komentarz, rozpoczęcie okresu owocowania roślin na działce czy zaproszeniu przez użytkownika do rozmowy. Jeśli posiada ono dodatkowo etykietę Zgloszenie, to pełni ono rolę zgłoszenia przez użytkownika, które jest przypisywane do pracownika o najmniejszej aktualnej liczbie zgłoszeń. Możliwe jest ustawienie powiadomienia jako przeczytane, albo je usunąć pod warunkiem, że nie jest zgłoszeniem.

**Właściwości:**

* typ – typ powiadomienia (komentarz\_post, zaproszenie, owocowanie\_roslin)
* opis – wygenerowany opis powiadomienia
* odnośnik – odnośnik do wspomnianego elementu (post, rozmowa prywatna, ogród)
* dataUtworzenia – data utworzenia powiadomienia
* avatar – ścieżka do avataru powiadomienia.

Rysunek 13 Węzeł Powiadomienie bez i z etykietą Zgloszenie

##### Relacje

###### Relacje bez właściwości

|  |  |  |  |
| --- | --- | --- | --- |
| **Nazwa** | **Kierunek** | **Opis** | **Przykład** |
| STWORZONA\_PRZEZ | Uzytkownik  |  V  RoslinaWlasna | Wskazuje roślinę nienależącą do katalogu stworzoną przez użytkownika |  |
| SKOMENTOWAL | Uzytkownik  |  V  Komentarz | Wskazuje komentarz, który utworzył użytkownik. |  |
| ODPOWIEDZIAL | Komentarz  |  V  Komentarz | Wskazuje komentarz, na który odpowiedział użytkownik. |  |
| JEST\_W\_POSCIE | Komentarz  |  V  Post | Wskazuje post zawierający wybrany komentarz. |  |
| JEST\_W\_ROZMOWIE | Uzytkownik  |  V  RozmowaPrywatna | Wskazuje rozmowę, w której jest użytkownik. |  |
| MA\_KOMENTARZ | Post  |  V  Komentarz | Wskazuje komentarz komentujący dany post. |  |
| MA\_POST | Uzytkownik  |  V  Post | Wskazuje post stworzony przez użytkownika. |  |
| MA\_WIADOMOSC | RozmowaPrywatna  |  V  Komentarz | Wskazuje komentarz w rozmowie. |  |
| WYSYLA\_POWIADOMIENIE | Uzytkownik  |  V  Powiadomienie | Wskazuje powiadomienie globalne wysyłane przez użytkownika. |  |
| ZGLASZA | Uzytkownik  |  V  Powiadomienie | Wskazuje zgłoszenie, które wysłał użytkownik do pracownika. |  |
| MA\_USTAWIENIA | Uzytkownik  |  V  Ustawienia | Wskazuje ustawienia użytkownika. |  |
| MA\_OGROD | Uzytkownik  |  V  Ogrod | Wskazuje ogród użytkownika. |  |
| MA\_DZIALKE | Ogrod  |  V  Dzialka | Wskazuje działkę ogrodu użytkownika. |  |
| MA\_FORME | Roslina,  RoslinaWlasna  |  V  Forma | Wskazuje formę rośliny. |  |
| MA\_GLEBE | Roslina,  RoslinaWlasna  |  V  Gleba | Wskazuje stosowną glebę dla rośliny. |  |
| MA\_GRUPE | Roslina,  RoslinaWlasna  |  V  Grupa | Wskazuje grupę rośliny. |  |
| MA\_KOLOR \_KWIATOW | Roslina,  RoslinaWlasna  |  V  Kolor | Wskazuje kolor kwiatów rośliny. |  |
| MA\_KOLOR\_LISCI | Roslina,  RoslinaWlasna  |  V  Kolor | Wskazuje kolor liści rośliny. |  |
| MA\_OKRES \_KWITNIENIA | Roslina,  RoslinaWlasna  |  V  Okres | Wskazuje miesiące, w której roślina kwitnie. |  |
| MA\_OKRES\_OWOCOWANIA | Roslina,  RoslinaWlasna  |  V  Okres | Wskazuje miesiące, w której roślina owocuje. |  |
| MA\_KWIAT | Roslina,  RoslinaWlasna  |  V  Kwiat | Wskazuje rodzaj kwiatów, jakie ma roślina. |  |
| MA\_ODCZYN | Roslina RoslinaWlasna  |  V  Odczyn | Wskazuje odczyn gleby, w jakich roślina powinna być sadzona. |  |
| MA\_OWOC | Roslina, RoslinaWlasna  |  V  Owoc | Wskazuje rodzaj, kształt albo kolor owocu rośliny. |  |
| MA\_PODGRUPE | Roslina, RoslinaWlasna  |  V  Podgrupa | Wskazuje podgrupę, do której należy roślina. |  |
| MA\_POKROJ | Roslina, RoslinaWlasna  |  V  Pokroj | Wskazuje pokrój rośliny. |  |
| MA\_SILE\_WZROSTU | Roslina, RoslinaWlasna  |  V  SilaWzrostu | Wskazuje szybkość rośnięcia rośliny. |  |
| MA\_ZIMOZIELONOSC\_LISCI | Roslina, RoslinaWlasna  |  V  Zimozielonosc | Wskazuje zachowanie liści podczas zimy. |  |
| MA\_WILGOTNOSC | Roslina, RoslinaWlasna  |  V  Wilgotnosc | Wskazuje optymalną wilgotność gleby dla rośliny. |  |
| MA\_STANOWISKO | Roslina, RoslinaWlasna  |  V  Stanowisko | Wskazuje optymalne stanowisko rośliny. |  |
| MA\_WALOR | Roslina, RoslinaWlasna  |  V  Walor | Wskazuje walory rośliny. |  |
| MA\_ZASTOSOWANIE | Roslina, RoslinaWlasna  |  V  Zastosowanie | Wskazuje zastosowanie rośliny. |  |

###### Relacje z właściwościami

Relacja: ZASADZONA\_NA

Połączenie (Rysunek 14) kierunku Roslina/RoslinaWlasna -> Dzialka. Wskazuje działkę, na której została zasadzona roślina.

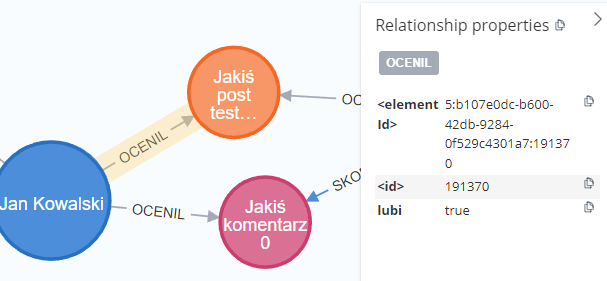
**Właściwości:**

* x, y – pozycja rośliny.
* tabX, tabY – jednowymiarowe tablice kafelków (x, y); używane zamiast tablic wielowymiarowych (ograniczenie Neo4j).
* kolor – kolor kafelków w HEX.
* obraz – ścieżka do obrazu rośliny
* tekstura – ścieżka do tekstury kafelków
* wyswietlanie – tryb wyświetlania rośliny
  + - KOLOR –pokolorowane kafelki.
    - TEKSTURA – tekstura kafelków (wymaga tekstury rośliny).
    - TEKSTURA\_KOLOR – pokolorowane kafelki z teksturą.



Rysunek 14 Relacja ZASADZONA\_NA

Relacja: OCENIL

Połączenie (Rysunek 15) kierunku Uzytkownik -> Post albo Uzytkownik -> Komentarz. Wskazuje węzeł oceniony przez użytkownika, którym jest post albo komentarz, gdyż mogą one być oceniane przez niego.

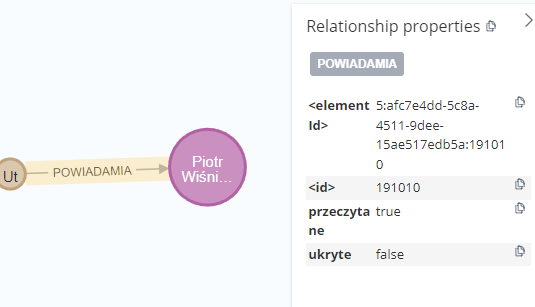
**Właściwości:**

* lubi – ocena pozytywna albo negatywna

Rysunek 15 Relacja OCENIL

Relacja: POWIADAMIA

Połączenie (Rysunek 16) kierunku Powiadomienie -> Uzytkownik. Wskazuje użytkownika, który jest powiadamiany.

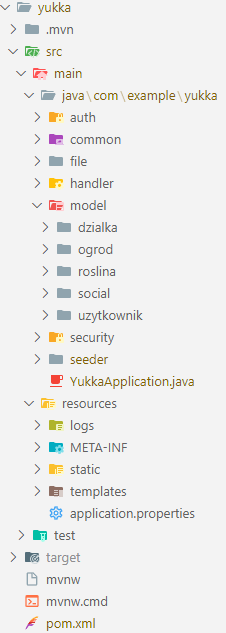
**Właściwości:**

* przeczytane – status przeczytania powiadomienia.
* ukryte – status ukrycia powiadomienia przez użytkownika (dotyczy zgłoszeń i specjalnych powiadomień).

Rysunek 16 Relacja POWIADAMIA

### Struktura backendu aplikacji

Backend, oparty na frameworku Spring Boot, obsługuje żądania REST od frontendu, waliduje dane, wykonuje zapytania do bazy danych i zwraca odpowiedzi w formacie JSON[[5]](#footnote-5) (Marrs, 2017). Projekt (zob. Rysunek 17) podzielono na część programową (zawiera logikę aplikacji) i część testową (obejmuje testy jednostkowe weryfikujące działanie poszczególnych elementów).



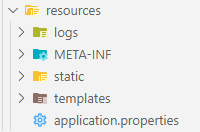
Rysunek 17 Struktura backendu

* Auth – Obsługa autoryzacji: logowanie, rejestracja, potwierdzenia e-mail, usuwanie konta.
* Common – Obiekty wspólne, np. odpowiedzi z paginacją.
* File – Zarządzanie plikami i folderami w katalogu uploads (Rysunek 18).



Rysunek 18 Katalog uploads

* Handler – Obsługa błędów i odpowiedzi dla użytkownika.
* Model – Reprezentacja elementów bazy danych, kontrolery, serwisy, repozytoria, żądania i odpowiedzi:
  + Dzialka – Obsługa działek.
  + Ogrod – Obsługa ogrodów.
  + Roslina – Obsługa roślin.
  + Social – Obsługa społeczności (posty, komentarze, oceny, rozmowy, powiadomienia).
  + Uzytkownik – Obsługa użytkowników i ich usług.
* Security – Konfiguracja zabezpieczeń: uprawnienia, filtry, limity żądań.
* Seeder – Seedowanie bazy danych (przy seed-database=true).
* YukkaApplication – Główna klasa aplikacji.
* Resources (Rysunek 19) – Logi, pliki .html (e-maile), application.properties (ustawienia aplikacji: porty, e-mail, JWT, ścieżki plików).



Rysunek 19 Katalog resources

* Pom.xml – Plik z zależnościami programu

#### Przykładowa klasa modelu danych

##### Obiekt: Oceniany

Nie jest węzłem w bazie danych, lecz rodzicem dla obiektów (Post, Komentarz), które dziedziczą relację OCENIL (@Relationship). Dzięki temu posiadają metody do liczenia ocen. W (Rysunek 20) znajduje się klasa obiektu.

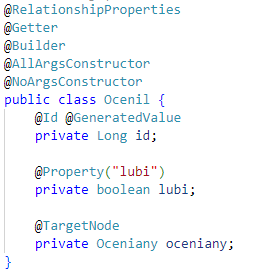


Rysunek 20 Kod obiektu Oceniany

##### Obiekt: OcenilReverse

Klasa w (Rysunek 21) reprezentuje relację OCENIL z właściwościami:

* Lubi – pozytywna/negatywna ocena
* Uzytkownik – wskazywany węzeł Uzytkownik.



Rysunek 21 Kod obiektu Ocenil

Obiekt: Post

Rysunek 22 Kod obiektu Post

Klasa Post (zob. Rysunek 22) reprezentuje węzeł Post i dziedziczy klasę Oceniany.

**Adnotacje:**

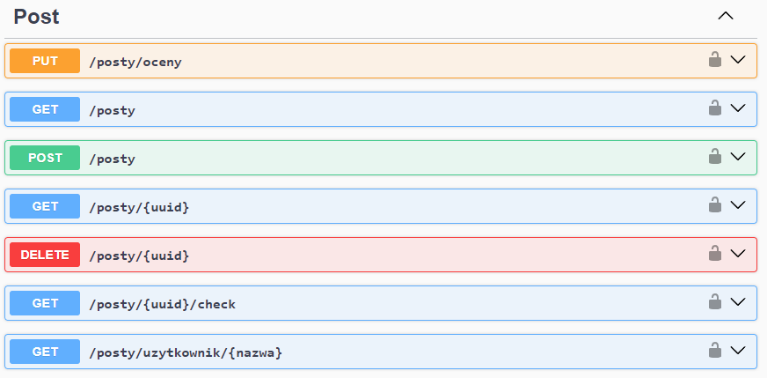
* @Node – definiuje węzeł w bazie danych.
* @Property – mapuje pola na właściwości węzła.
* @Relationship – definiuje typ i kierunek relacji.
* @Getter, @Setter, @AllArgsConstructor, @NoArgsConstructor, @Builder – automatyczne generowanie metod pomocniczych.
* @Id, @GeneratedValue – Oznajmia, że jest to pole identyfikatora węzła, które jest generowane dla każdego obiektu mapowanego z bazy danych.
* @CreatedDate – Data utworzenia.



Rysunek 22 Kod obiektu Post

#### Przykład procesu żądania i odpowiedzi

Aby użytkownik mógł wykonać operacje, musi on wysłać żądanie do serwera na backendzie. Na (Rysunek 23) znajduje się widok możliwych żądań dla Kontrolera o swaggerowym tagu „Post”:

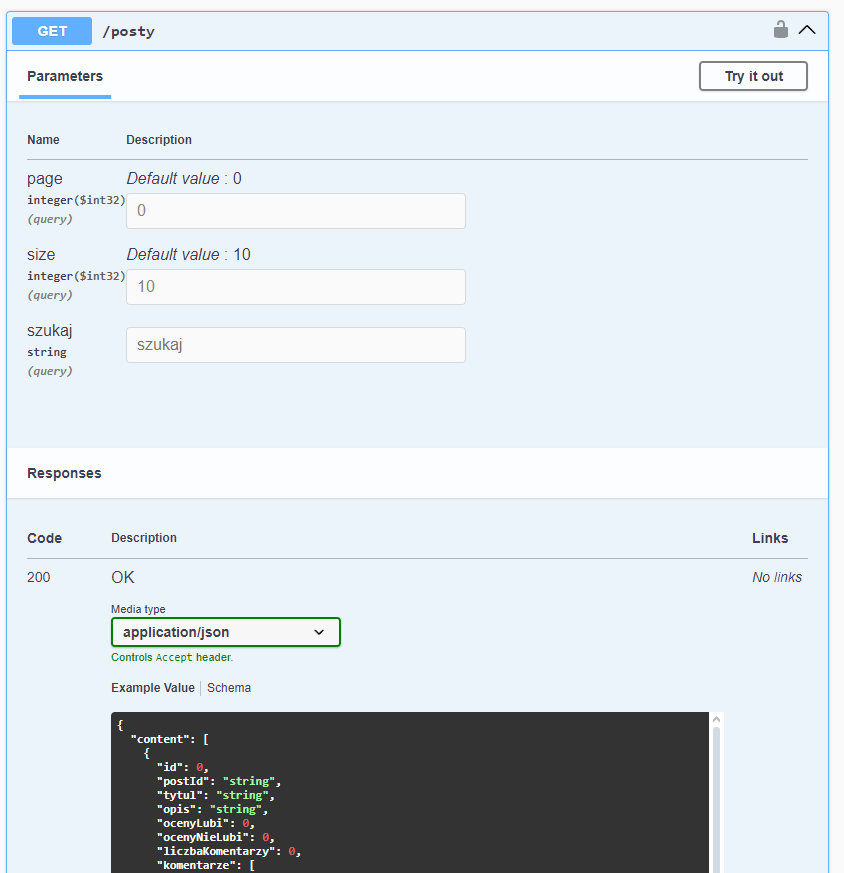


Rysunek 23 Żądania dla kontrolera Post w Swaggerze

Kontroler posiada następujące ścieżki:

* /posty/oceny PUT – Dodanie lub aktualizacja oceny postu.
* /posty GET – Lista postów z paginacją.
* /posty POST – Dodanie nowego postu.
* /posty/{uuid} GET – Szczegóły postu o podanym identyfikatorze.
* /posty/{uuid} DELETE – Usunięcie postu o podanym identyfikatorze.
* /posty/{uuid}/check GET – Post o podanym identyfikatorze, bez komentarzy.
* /posty/uzytkownik/{nazwa} – Lista postów użytkownika z paginacją.

Załóżmy, że użytkownik chce pobrać wszystkie posty. Do tego zadania należy użyć metody GET /posty. Na (Rysunek 24) znajduje się widok ze Swaggera:



Rysunek 24 Widok żądania GET /posty

Do żądania można opcjonalnie podać następujące parametry:

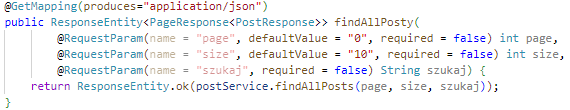
* page (strona) o domyślnej wartości 0
* size (ilość zwracanych elementów na stronę) o domyślnej wartości 10
* szukaj (nazwa postu).

Przykładowa odpowiedź dla tego żądania zawiera listę postów wraz z takimi informacjami jak aktualna strona, całkowitą ilość stron, całkowitą ilość elementów, oraz czy jest to pierwsza/ostatnia strona (zob. Rysunek 25):



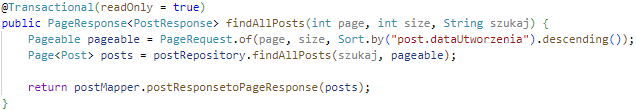
Rysunek 25 Odpowiedź żądania

Samo żądanie w kodzie wygląda tak, jak na (Rysunek 26):

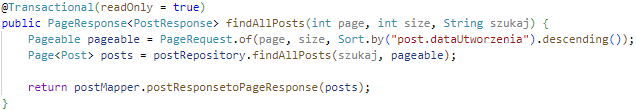


Rysunek 26 Kod funkcji w kontrolerze

Kontroler wykonuje funkcję z serwisu, którą opakowuje w ciało odpowiedzi o kodzie 200 (OK). Przechodząc do serwisu można znaleźć następujący kod (zob. Rysunek 27):

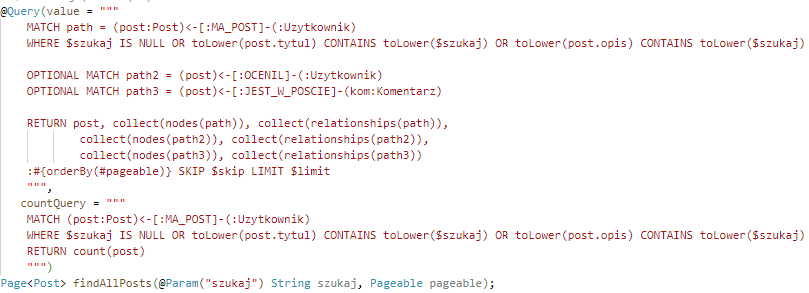


Rysunek 27 Kod funkcji serwisu



Rysunek 27 Kod funkcji serwisu

Serwis używa metody z repozytorium, aby zwróciło odpowiedź zawierającą daną stronę z paginacji postów, a następnie mapuje je odpowiednio do odpowiedzi, którą zwraca. Przechodząc do repozytorium, można znaleźć zapytanie widoczne na (Rysunek 28):



Rysunek 28 Zapytanie do bazy danych

Zapytanie w języku CYPHER przyjmuje parametr wyszukujący i obiekt mapujący wynik na stronę w paginacji, a następnie wysyła je do bazy danych, zwracając odpowiedź lub błąd. Zapytanie jest podzielone na dwa etapy. Pierwszy zwraca stronę wyników, a drugi całkowitą liczbę elementów.

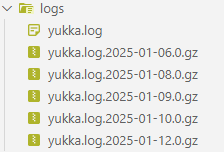
Po wykonaniu tych operacji, otrzymuje się odpowiedź (zob. Rysunek 29), która może wyglądać przykładowo tak (z pominięciem zawartości obrazów, które są dużymi tablicami bajtów):



Rysunek 29 Odpowiedź żądania pobrania postów

#### Logowanie zdarzeń i błędów

Aplikacja aktywnie loguje zdarzenia w aplikacji takie jak Dodawanie nowych roślin, powiadomienia o błędach, czy też ostrzeżenia. Pliki logów są zapisywane w folderze logs (zob. Rysunek 30), gdzie przechowywane są logi do tygodnia wstecz:



Rysunek 30 Folder z logami

Przykładowy log informacyjny (Rysunek 31):



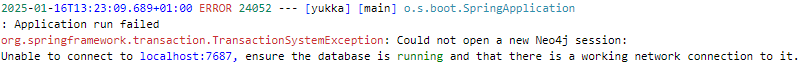
Rysunek 31 Log informacyjny

Przykładowy log ostrzeżenia (Rysunek 32):



Rysunek 32 Log ostrzeżenia

Przykładowy log o błędzie (Rysunek 33):



Rysunek 33 Log błędu

#### Zadania okresowe w tle

Zadania okresowe wykonywane w tle ograniczają się do sprawdzania zbanowanych użytkowników (codziennie) oraz sprawdzania okresów kwitnięcia i owocowania roślin na działkach (na początku miesiąca)

Są one wykonywane dodatkowo przy starcie aplikacji. Przykładowo, na (Rysunek 34) funkcja checkUnban() odbanowuje użytkowników, dla których ban się zakończył:



Rysunek 34 Kod codziennego sprawdzania banów

#### Seedery

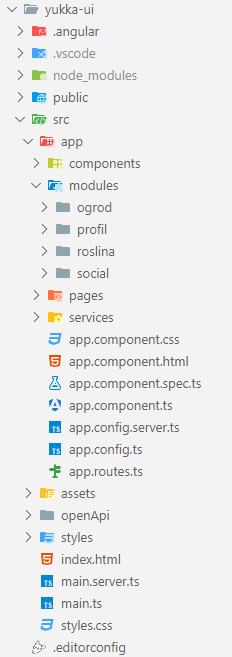
Aplikacja posiada seedery do poszczególnych części aplikacji.

* RoslinaImporterService – Importuje rośliny z podanego pliku .csv w ilości roslina.seed.amount. Ze względu na ilość i złożoność roślin, zamiast usuwania kolejno każdego węzła i relacji, czyszczona jest cała baza danych.
* GeneralSeederService – Importuje użytkowników, posty, komentarze, rozmowy prywatne, rośliny użytkownika i rośliny na działce.
* SocialSeeder – Seeduje część społecznościową (posty, komentarze, odpowiedzi) w ramach GeneralSeederService, z możliwością dostosowania liczby postów, komentarzy i głębokości odpowiedzi.

RoslinaWlasnaSeeder – Seeduje rośliny użytkownika. Używane w GeneralSeederService.

### Struktura frontendu aplikacji

Frontend jest projektem zrealizowanym w frameworku Angular zawierający strony aplikacji budowanych z pomocą komponentów komunikujących się między sobą głównie poprzez sygnały. Struktura (zob. Rysunek 35) zawiera strony, komponenty je budujące, serwisy do określonych działań, a także moduły grupujące daną część aplikacji w jedno.



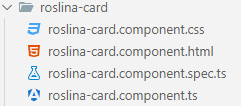
* Modules – Grupa powiązanych elementów, np. moduł profil zawiera komponenty, serwisy, routy i strony dla profilu użytkownika.
* Components – Niewielkie elementy aplikacji przeznaczone do wielokrotnego użytku, np. karta rośliny.
* Services – Funkcje współdzielone pomiędzy elementami, np. obsługa żądań do backendu, czy weryfikacja parametrów routa.
* Pages – Strony łączące moduły, serwisy i inne części aplikacji.
* App.routes – Routy prowadzące do stron aplikacji.

Rysunek 35 Struktura frontendu

#### Przykładowy komponent

Struktura frontendu jest zbudowana z wielu komponentów. Przykładowo, komponent roslina-card (zob. Rysunek 36) służy do wyświetlania karty rośliny. Składa się z następujących plików:

* roslina-card.component.css: definiuje style.
* roslina-card.component.html: struktura html.
* roslina-card.component.spec.ts: Testy jednostkowe.
* roslina-card.component.ts: Logika komponentu (dane, metody, zdarzenia).

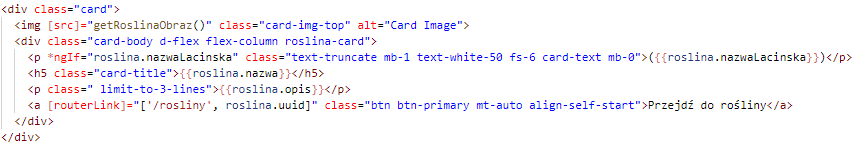
Rysunek 36 Komponent roslina-card oraz jego widok

Plik roslina-card.component.ts (Rysunek 37) zawiera importy potrzebnych modułów, klasę komponentu z obiektem wejściowym @Input dla rośliny, getter, setter i metodę zwracająca obraz rośliny w formacie base64.



Rysunek 37 Kod roslina-card.component.ts

W pliku roslina-card.component.html (Rysunek 38) wywołuje się metodę getRoslinaObraz() zwracająca obraz rośliny do tagu <img> , a także wypisywane są właściwości rośliny z pominięciem nazwy łacińskiej, jeśli jest to roślina użytkownika (\*ngIf).



Rysunek 38 Kod roslina-card.component.html

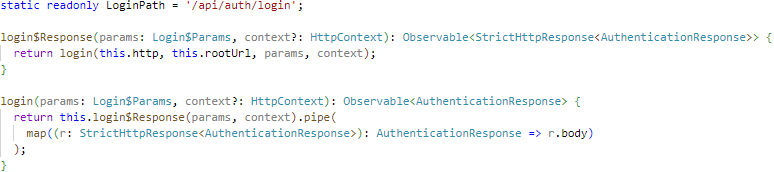
Komponent ten jest używany do pokazywania wyników z listy roślin, np. na (Rysunek 39) korzystając z pętli ngFor, można dla każdej rośliny w liście utworzyć komponent roslina-card, przekazując do niego dane konkretnej rośliny:



Rysunek 39 Przykładowa pętla tworząca karty roślin

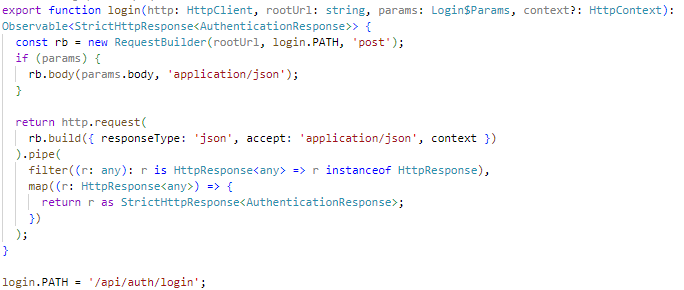
#### Przykład metody API z serwisu

Metody API zostały wygenerowane przy użyciu narzędzia ng-openapi-gen. Są podzielone na metody zwracające cały HttpResponse i metody zwracające samo ciało. Każda metoda przyjmuje odpowiednie parametry oraz opcjonalnie HttpContext. Na (Rysunek 40) znajduje się przykładowa metoda API w pliku authentication.service.ts:



Rysunek 40 Fragment kodu z authentication.service.ts

Korzysta się tutaj dodatkowo z wygenerowanej funkcji login osadzonej w osobnym pliku login.ts (zob. Rysunek 41), gdzie wykonuje żądanie logowania użytkownika zgodnie z dokumentacją OpenApi, przyjmując parametry i zwracając odpowiedź.



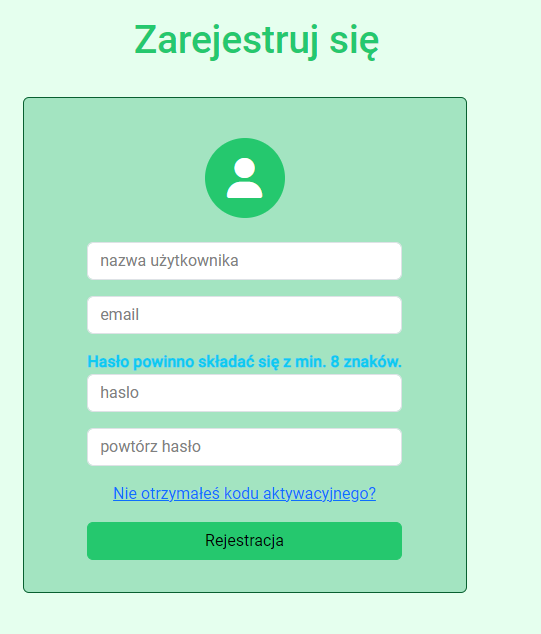
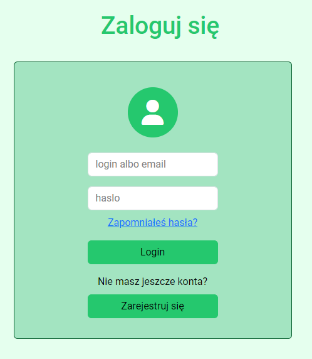
Rysunek 41 funkcja login w login.ts

## Opis funkcjonalności systemu

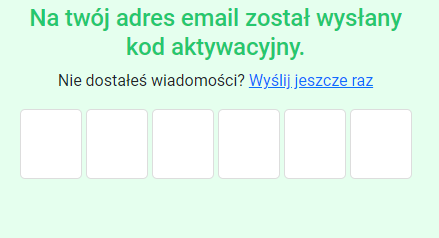
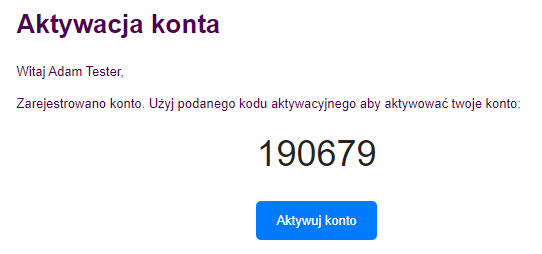
### Funkcjonalności zwykłego użytkownika

#### Rejestracja

Aby się zarejestrować, należy w pasku nawigacyjnym wcisnąć przycisk „Zaloguj się” przenoszący do strony logowania (zob. Rysunek 42). Następnie panelu wybrać opcję „Zarejestruj się” i wprowadzić wymagane dane. Po rejestracji następuje przeniesienie na stronę aktywacji konta (zob. Rysunek 43), gdzie wprowadza się kod otrzymany na adres e-mail.



Rysunek 42 Panel logowania i rejestracji

Rysunek 43 Panel aktywacji i kod z wiadomości email

Aby się zalogować, należy w panelu logowania podać adres email lub nazwę użytkownika i hasło. Po kliknięciu przycisku „Login” użytkownik zostanie zalogowany i przeniesiony do strony głównej (zob. Rysunek 44).

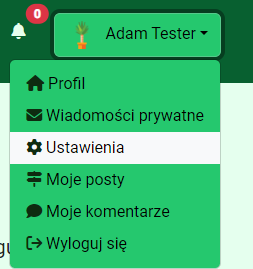
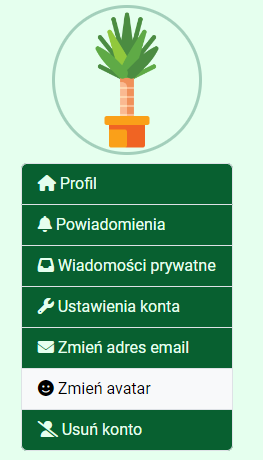


Rysunek 44 Widok po zalogowaniu się

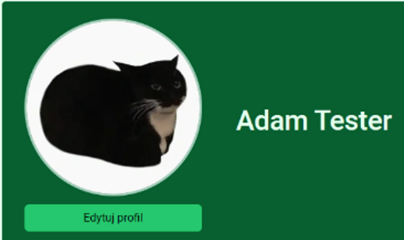
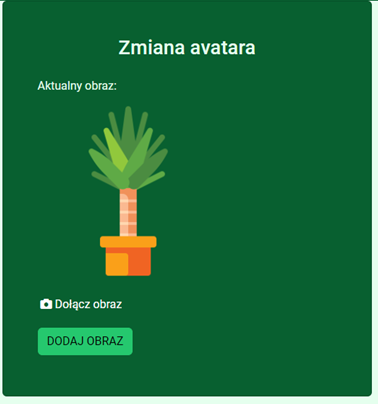
#### Zarządzanie kontem

##### Zmiana avatara

Aby zmienić avatar, należy przejść do ustawień użytkownika i wybrać opcję „Zmień avatar” (zob. Rysunek 45). Następnie w panelu (zob. Rysunek 46) dołączyć obraz, a następnie kliknij „Dodaj obraz”.

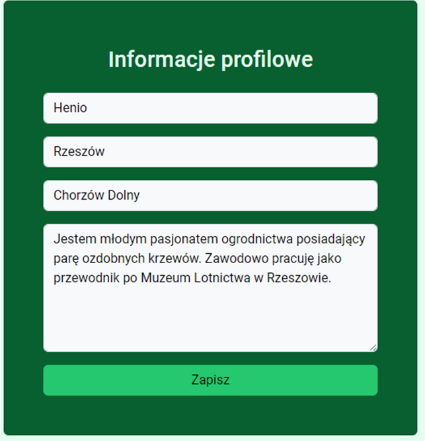
Rysunek 45 Przejście do ustawień i do zmiany avatara



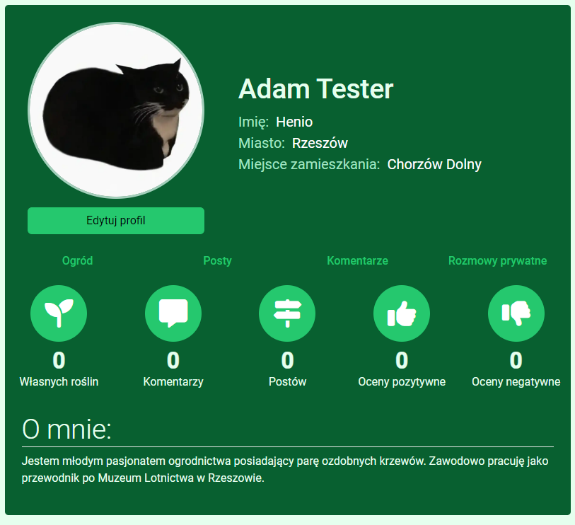
Rysunek 46 Panel zmiany avatara i profil po zmianie

##### Edycja profilu

W panelu ustawień kliknij „Profil”, aby edytować dane profilowe (zob. Rysunek 49). Można zmieniać lub usuwać imię, miasto, miejsce zamieszkania i opis. Po zapisaniu profil zostanie zmieniony (zob. Rysunek 48).



Rysunek 47 Panel edycji profilu przed i po podaniu danych



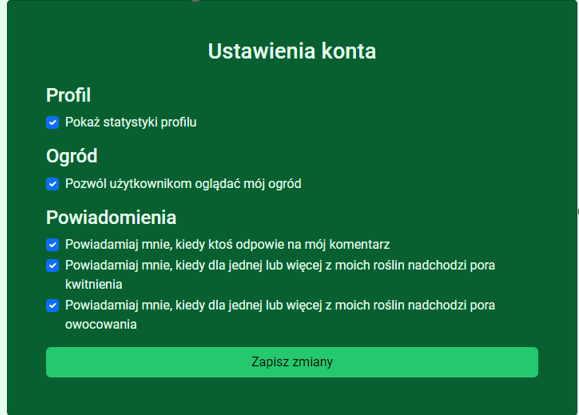
Rysunek 48 Profil po edycji

##### Zmiana ustawień konta

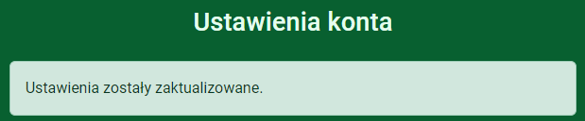
Aby zmienić ustawienia konta, należy w panelu ustawień przejść do opcji „Ustawienia konta” (zob. Rysunek 49). W panelu można przełączyć:

* Pokazywanie statystyk profilu
* Ukrywanie ogrodu przed innymi użytkownikami
* Powiadomienia aplikacji

Po zapisaniu zmian profil zostanie zmieniony (zob. Rysunek 50).



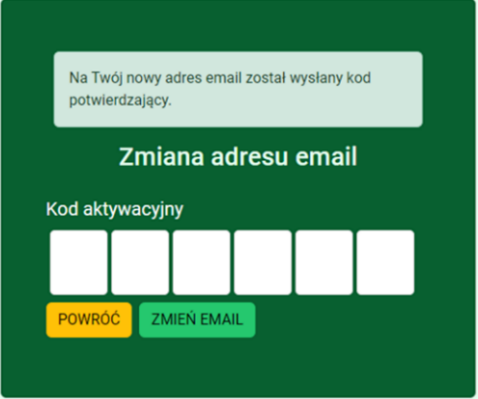
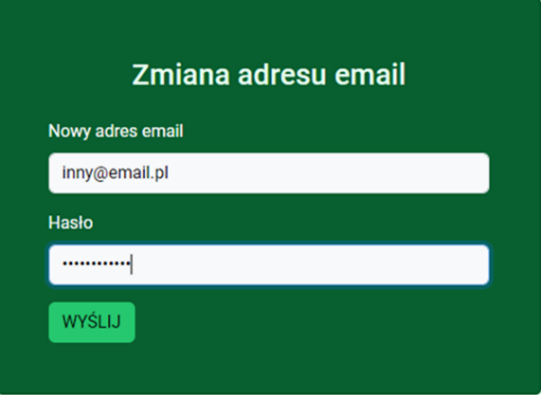
Rysunek 49 Panel ustawień konta



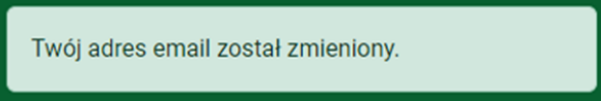
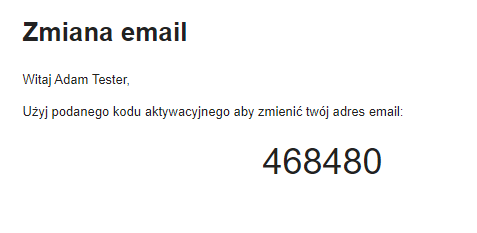
Rysunek 50 Komunikat o zmianie ustawień i profil po zmianie

##### Zmiana adresu email

Aby zmienić adres email, należy w ustawieniach przejść do opcji „Zmień adres email”, gdzie trzeba w panelu (zob. Rysunek 51) podać nowy adres email oraz obecne hasło. Po kliknięciu przycisku „Wyślij” zostanie wysłana wiadomość na nowy adres email, z kodem, który trzeba podać w następnym kroku. Po podaniu kodu i kliknięciu przycisku „Zmień email” pokaże się komunikat o zmianie adresu email (zob. Rysunek 52).



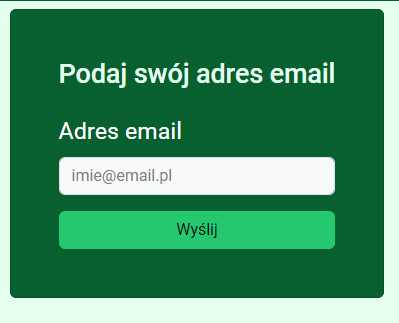
Rysunek 51 Panel zmiany adresu email



Rysunek 52 Kod aktywacyjny i komunikat po podaniu

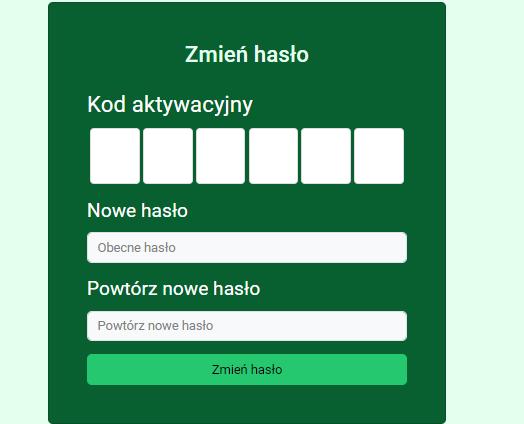
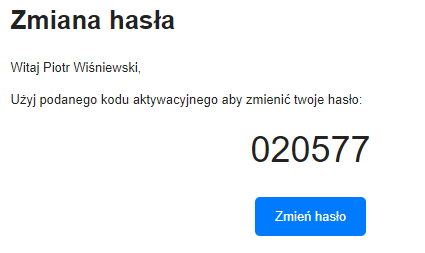
##### Zmiana hasła

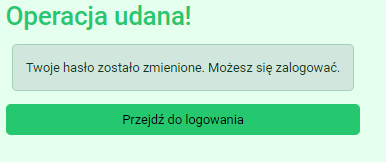
Aby zmienić hasło, należy w panelu logowania kliknąć link „Zapomniałeś hasła?”, który przeniesie użytkownika do strony z resetowaniem hasła (zob. Rysunek 53). Tam należy podać adres email użytkownika, na który zostanie wysłana wiadomość o zmianie hasła:



Rysunek 53 Link do zmiany hasła i formularz

Po wysłaniu wiadomości trzeba będzie podać w następnej części kod oraz nowe hasło, gdzie po poprawnym podaniu hasło zostanie zmienione (zob. Rysunek 54):

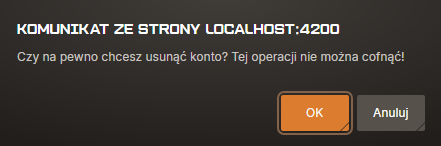
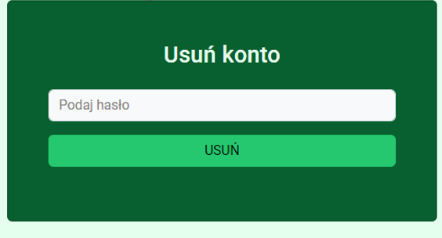




Rysunek 54 Formularz zmiany hasła

##### Usuwanie konta

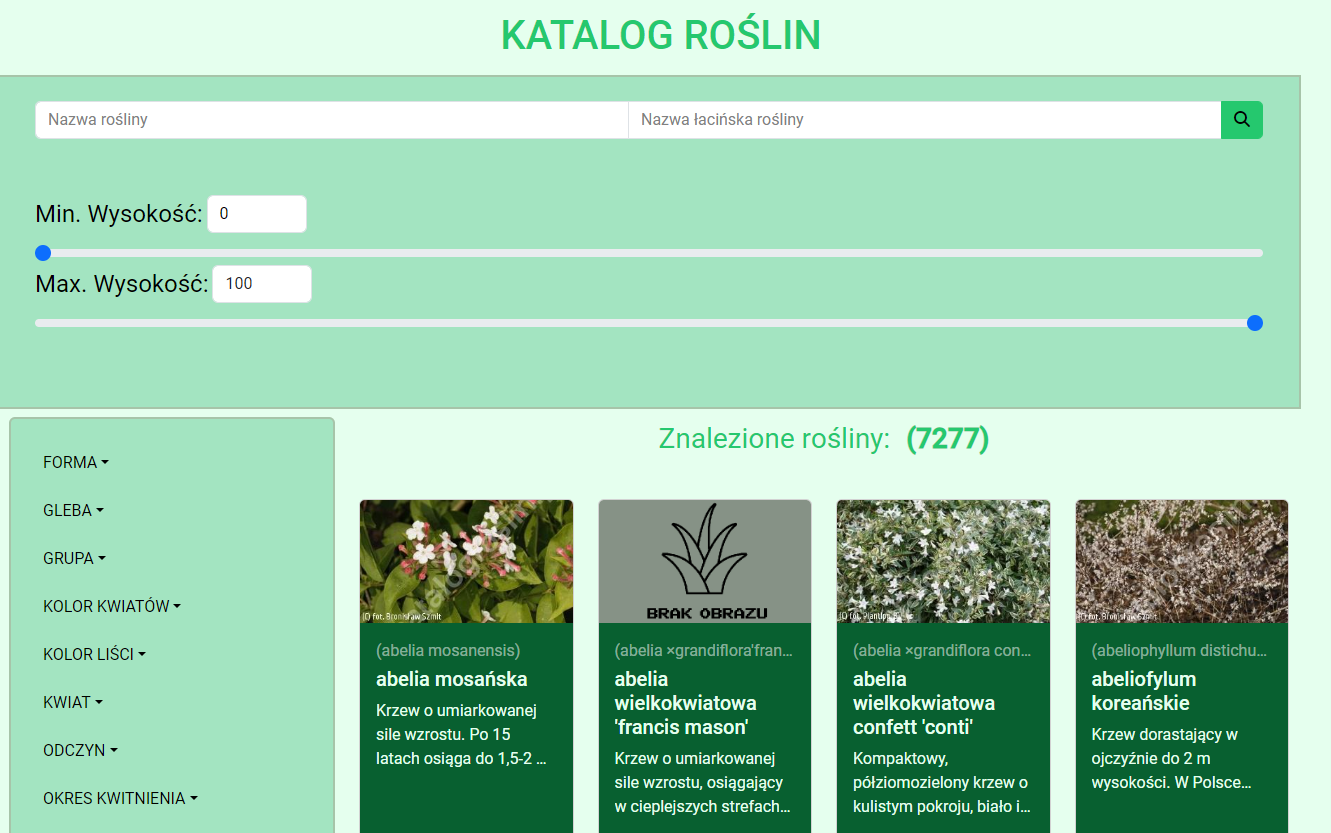
Aby usunąć konto, należy w ustawieniach wybrać panel „Usuń konto” (zob. Rysunek 55), a następnie wpisać hasło i kliknąć przycisk „Usuń”. Po jego naciśnięciu pojawi się ostrzeżenie, po którego potwierdzeniu aplikacja wyloguje użytkownika, usunie konto i zmieni widok na stronę główną.



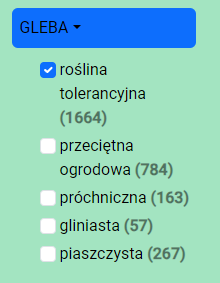
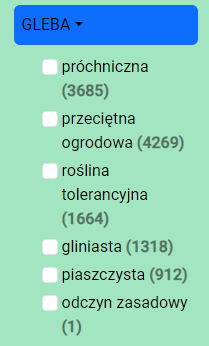
Rysunek 55 Panel usuwania konta i potwierdzenie

#### Przeglądanie roślin w katalogu

Aby przeglądać rośliny w katalogu, należy wybrać „Przeglądaj katalog” w pasku nawigacyjnym, przenoszący na stronę katalogu roślin (zob. Rysunek 56). W katalogu można wyszukiwać rośliny według nazwy, wysokości, cech (np. rodzaj gleby, miesiące kwitnienia) oraz wprowadzać dodatkowe filtry (zob. Rysunek 57). Użytkownik może łatwo usunąć wybrane cechy klikając znak „X” (zob. Rysunek 58). Katalog obsługuje paginację, umożliwiając przechodzenie między stronami.



Rysunek 56 Widok katalogu roślin



Rysunek 57 Lista cech typu Gleba przed i po wyszukaniu



Rysunek 58 Widok tagów cech

Aby zobaczyć roślinę z katalogu, należy kliknąć przycisk „Przejdź do rośliny”, który przeniesie użytkownika do strony rośliny (zob. Rysunek 59). Na stronie rośliny znajduje się jej nazwa, nazwa łacińska, obraz, opis oraz wszystkie jej cechy:



Rysunek 59 Widok strony rośliny

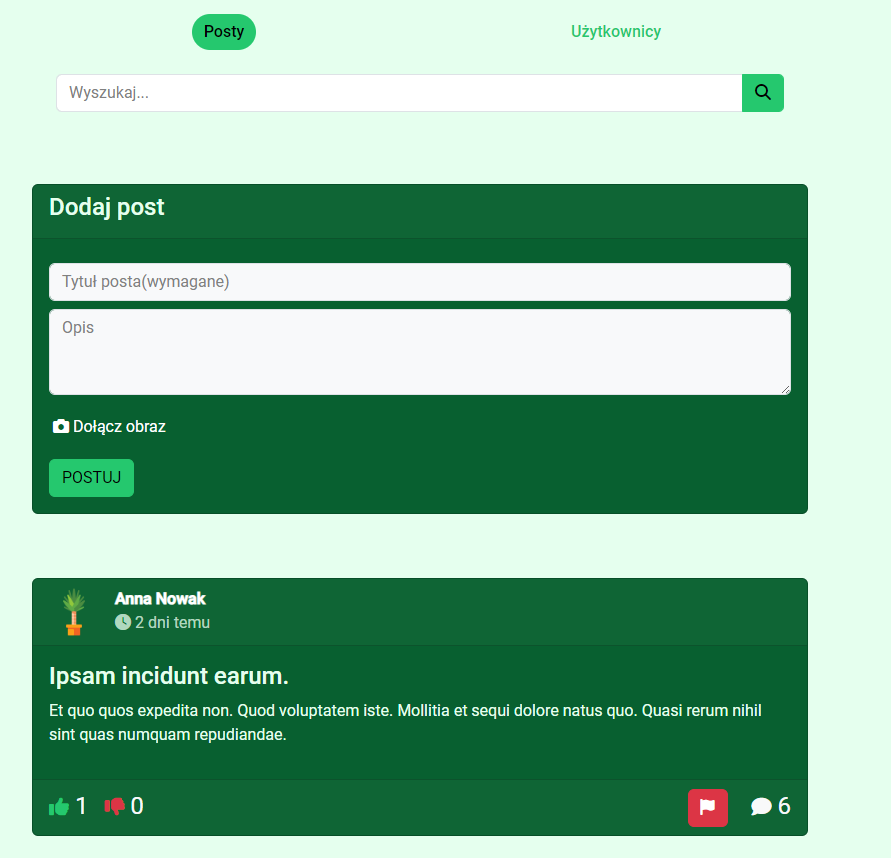
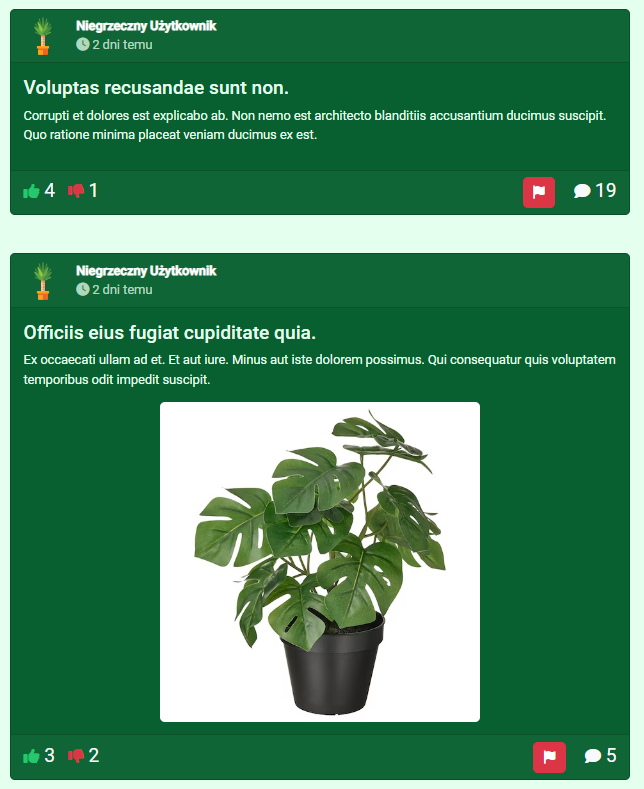
#### Zarządzanie własnymi roślinami

Aby zarządzać własnymi roślinami, należy w pasku nawigacyjnym wybrać opcję „Mój ogród”, a następnie w części „Rośliny użytkownika” kliknąć przycisk „Zobacz”, który przeniesie użytkownika do jego katalogu z roślinami. Tam można takie same operacje, jak w rozdziale (Zarządzanie roślinami), lecz są one wykonywane na własnych roślinach, zamiast na roślinach w oficjalnym katalogu.

#### Zarządzanie postami i komentarzami

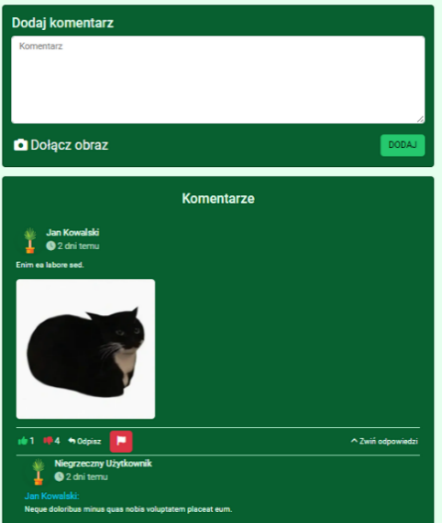
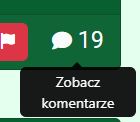
##### Przeglądanie postów i komentarzy

Aby przejść do postów, należy wybrać „Społeczność” w pasku nawigacyjnym. Panel społeczności (zob. Rysunek 60) ma wyszukiwarkę postów i użytkowników. W postach zastosowano mechanizm „Infinite Scrolling”[[6]](#footnote-6), ładujący nowe posty po dotarciu do końca strony (Lamis Chebbi, 2024). Posty mają autora, tytuł, opis, obraz (opcjonalnie), ocenę oraz liczbę komentarzy. Jest też przycisk zgłaszania postu.

Rysunek 60 Wyszukiwarka postów i przykłady postów

Klikając na liczbę komentarzy lub ikonkę dymku przechodzi się do strony postu (zob. Rysunek 61). Strona umożliwia dodanie komentarza oraz wyświetla listę wszystkich komentarzy. Komentarze mają opcje oceniania, odpowiadania i zwijania zagnieżdżonych odpowiedzi.



Rysunek 61 Widok komentarzy postu

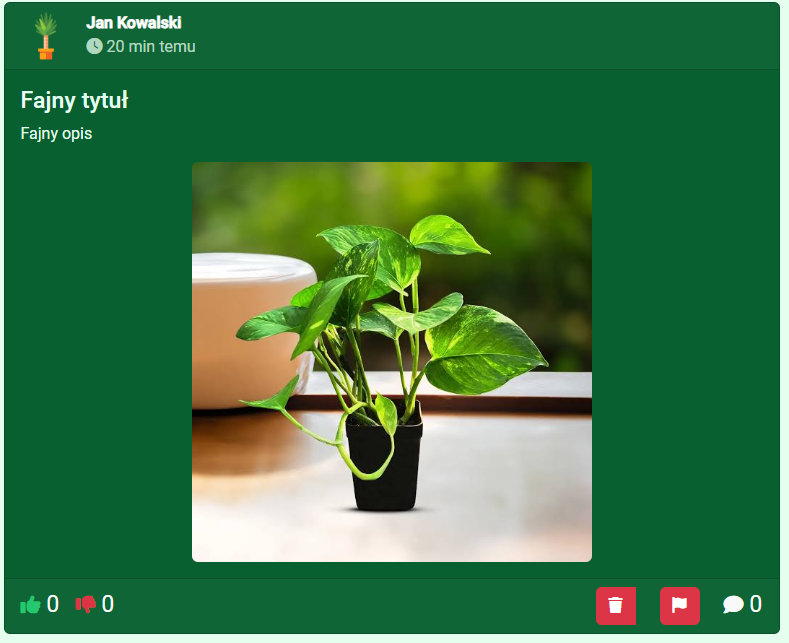
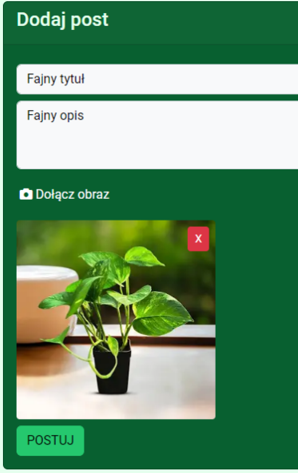
Swoje posty i komentarze można przeglądać, klikając w panelu użytkownika (zob. Rysunek 62) opcję „Moje posty” lub „Moje komentarze”, które przekieruje użytkownika do strony z jego postami/komentarzami:



Rysunek 62 Przejście do postów użytkownika

##### Dodanie postu

Post można dodać w panelu „Społeczność”, gdzie w formularzu (zob. Rysunek 63) należy podać tytuł oraz opcjonalnie opis i obraz postu. Po zatwierdzeniu użytkownik zostanie przeniesiony do strony postu. Post można dodawać co 30 sekund.



Rysunek 63 Formularz dodania postu i post po dodaniu

##### Usuwanie postu

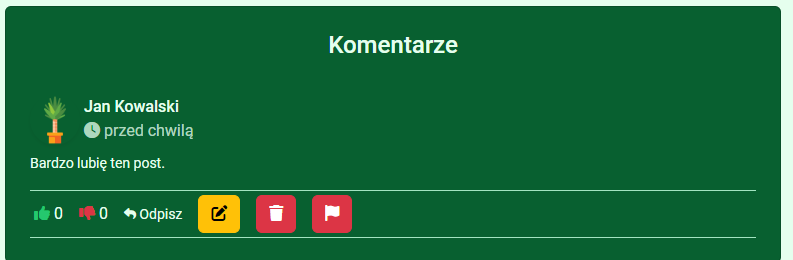
Aby usunąć post, należy kliknąć ikonę śmietnika i zatwierdzić usuwanie (zob. Rysunek 64). Po zatwierdzeniu, post zostanie usunięty, a użytkownik zostanie przekierowany z powrotem do strony społeczności.



Rysunek 64 Usuwanie postu

##### Dodanie komentarza

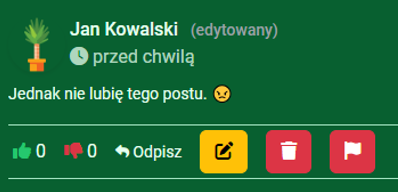
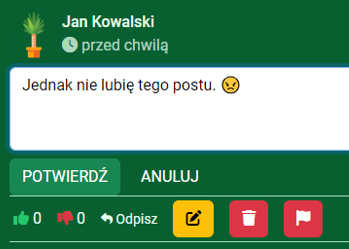
Aby dodać komentarz, należy na stronie postu wypełnić formularz (zob. Rysunek 65), a następnie zatwierdzić. Komentarz można dodawać co 30 sekund.



Rysunek 65 Dodawanie komentarza i dodany komentarz

##### Edycja komentarza

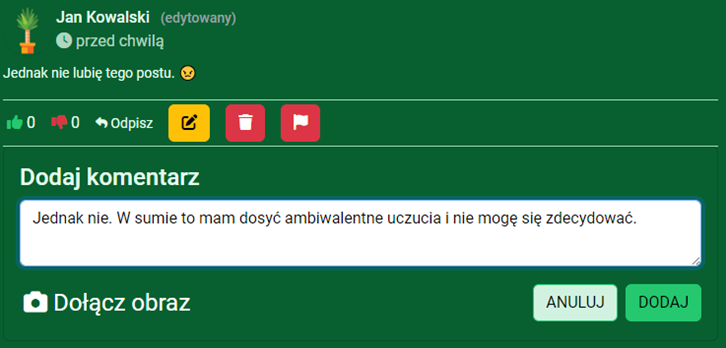
Aby edytować komentarz, należy kliknąć żółtą ikonę wywołującą formularz edycji, po którego zatwierdzeniu komentarz zostanie zmieniony (zob. Rysunek 66):

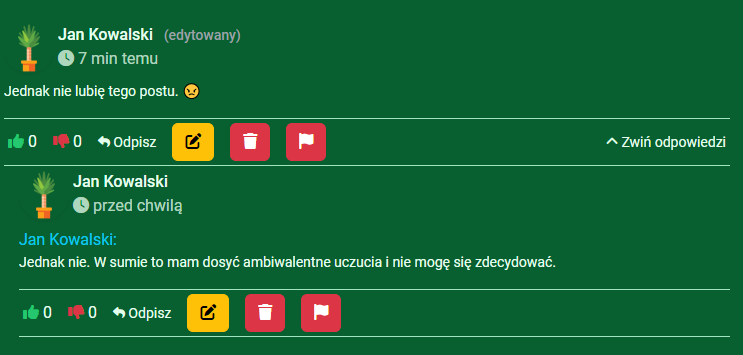


Rysunek 66 Edycja komentarza i komentarz po edycji

##### Dodanie odpowiedzi

Aby dodać odpowiedź do komentarza, należy kliknąć „Odpisz” w komentarzu, a następnie wypełnić formularz (zob. Rysunek 67):





Rysunek 67 Dodawanie odpowiedzi i odpowiedź po edycji

##### Ocena postu lub komentarza

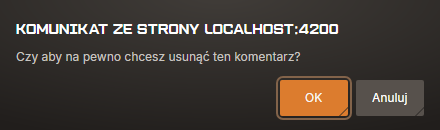
Komentarz lub post można ocenić, jako zalogowany użytkownik, klikając na kciuk w górę lub kciuk w dół (zob. Rysunek 68):



Rysunek 68 Ocena postu/komentarza

##### Usuwanie komentarza

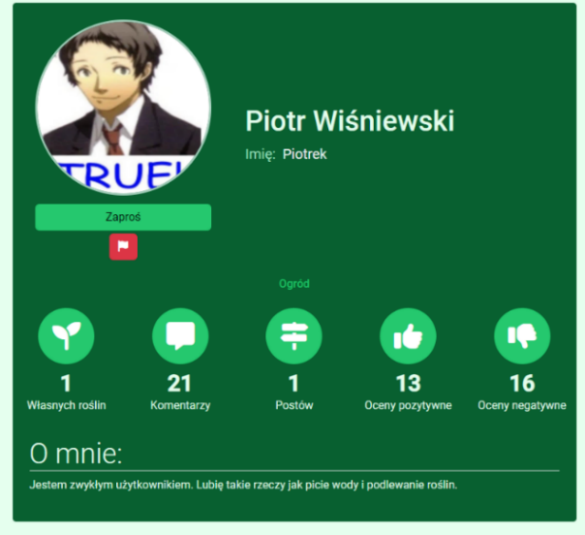
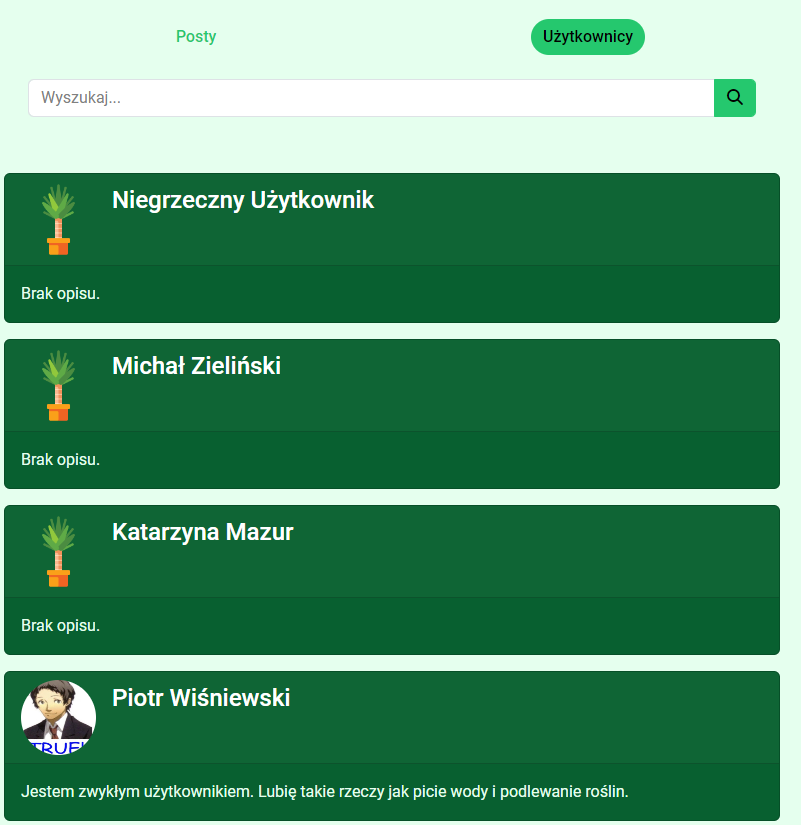
Aby usunąć komentarz, należy w komentarzy kliknąć ikonę śmietnika i potwierdzić (zob. Rysunek 69).



Rysunek 69 Ikona usuwania

#### Przeglądanie użytkowników

W panelu „Społeczność” należy kliknąć na przycisk „Użytkownicy”, który zabierze użytkownika do listy użytkowników (zob. Rysunek 70):



Rysunek 70 Lista użytkowników i profil jednego z nich

#### Zarządzanie ogrodem

##### Przeglądanie ogrodu

Aby przeglądać ogród, zalogowany użytkownik musi przejść do ogrodu klikając „Mój ogród” w pasku nawigacyjnym lub wybierając opcję „Ogród” na profilu, jeśli jest on publiczny. Po przejściu do ogrodu (Rysunek 71) użytkownik zobaczy nazwę ogrodu, listę działek oraz przejście do katalogu własnych roślin.

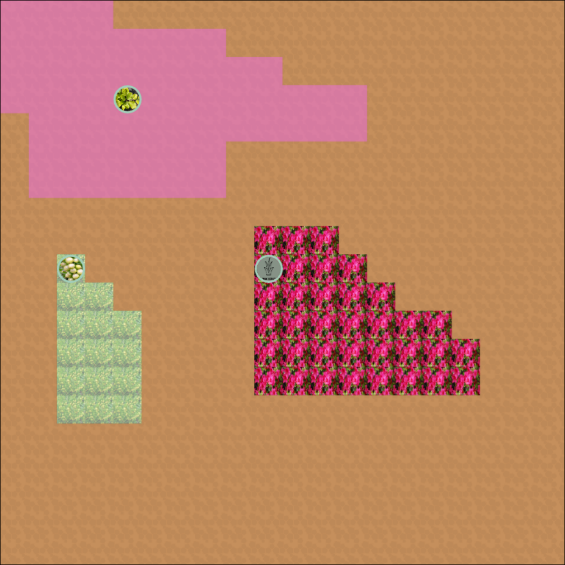
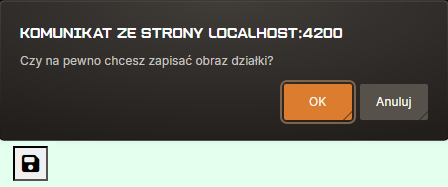


Rysunek 71 Widok ogrodu użytkownika

Aby przejść do działki, należy kliknąć „Zobacz” na wybranej działce. Strona działki (zob. Rysunek 72) pokazuje nazwę działki i rośliny w formie obrazów i kafelków. Istnieje możliwość wyboru różnych trybów wyświetlania roślin, takich jak kolorowanie kafelków, wyświetlanie samej tekstury roślin lub oba tryby jednocześnie. Użytkownik może przybliżać/oddalać widok działki oraz przemieszczać się po niej poprzez przeciąganie. Dodatkowo, można zapisać obraz działki klikając w ikonę dyskietki i potwierdzając (zob. Rysunek 73).



Rysunek 72 Widok działki

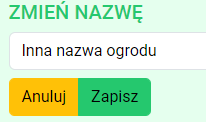


Rysunek 73 Potwierdzenie zapisu i stworzony obraz działki

##### Zmiana nazwy ogrodu lub działki

Aby zmienić nazwę ogrodu lub działki, należy kliknąć ikonę po prawej od nazwy, a następnie w polu zmienić nazwę i potwierdzić (zob. Rysunek 74):

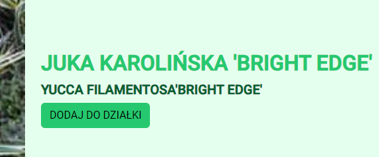




Rysunek 74 Zmiana nazwy ogrodu

##### Dodanie rośliny do działki

Aby dodać roślinę do działki, należy będąc zalogowany przejść do strony rośliny i kliknąć przycisk „Dodaj do działki” (zob. Rysunek 75) przenoszący do strony dodawania rośliny do działki (zob. Rysunek 76).



Rysunek 75 Przycisk "Dodaj do działki"

Aby dodać roślinę do działki, należy wykonać parę kroków:

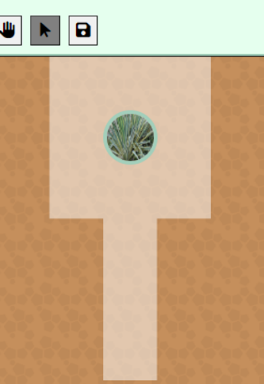
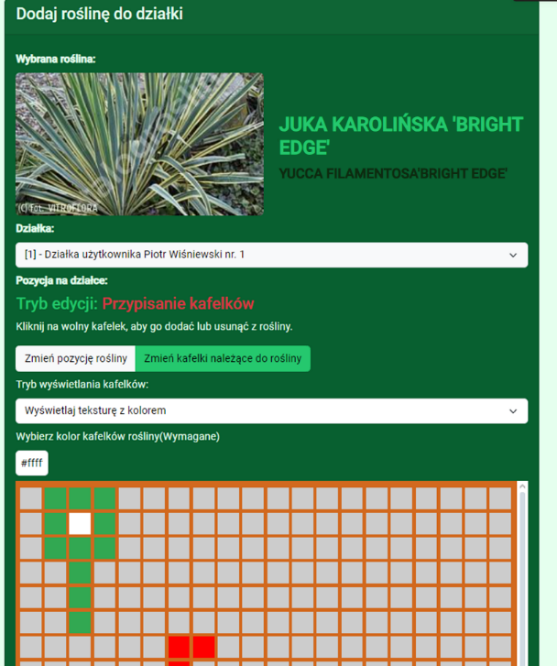
1. Wybrać działkę.
2. Wybrać tryb wyświetlania kafelków:

* Wyświetlanie samego koloru.
* Wyświetlanie samej tekstury.
* Wyświetlanie tekstury z nałożonym kolorem.

1. Wybrać kafelki, które zajmie roślina:

* Białe: Pozycja obrazu rośliny.
* Zielone: Zaznaczone kafelki rośliny.
* Szare: Wolne kafelki.
* Czerwone: Kafelki zajęte przez inne rośliny.

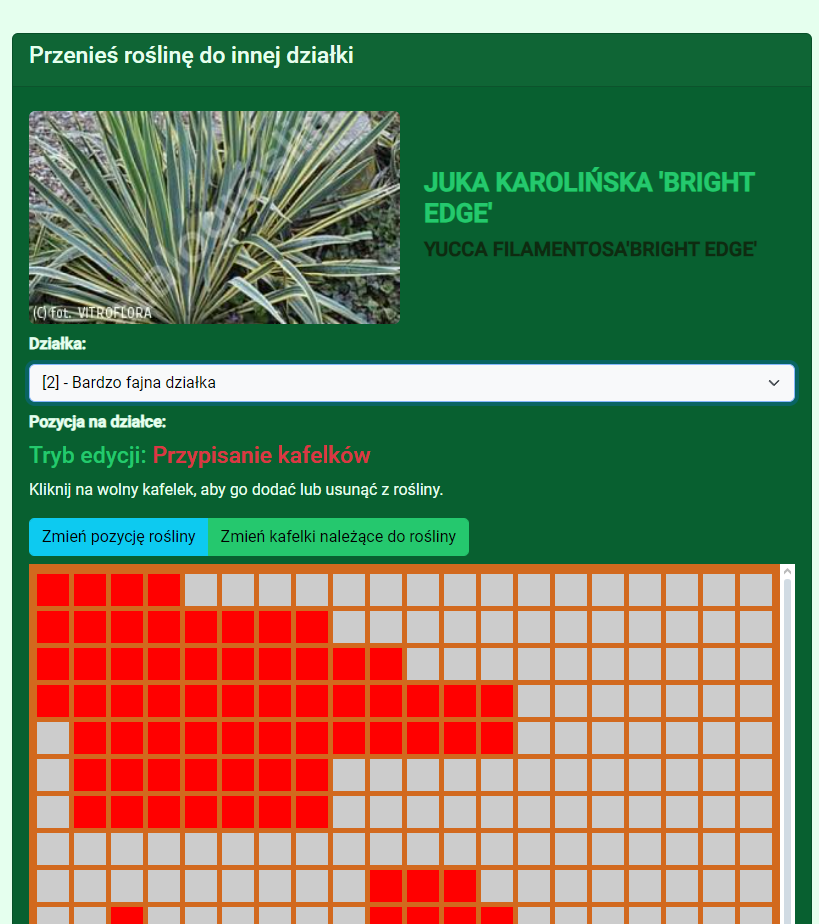
1. (Opcjonalnie) Wybrać obraz rośliny.
2. (Opcjonalnie) Wybrać teksturę kafelków, jeśli wybrano wyświetlanie samej tekstury.
3. Wcisnąć przycisk „Dodaj roślinę”, po którym zostanie się przeniesionym do działki z nowo dodaną rośliną.



Rysunek 76 Strona dodawania rośliny do działki i dodana roślina

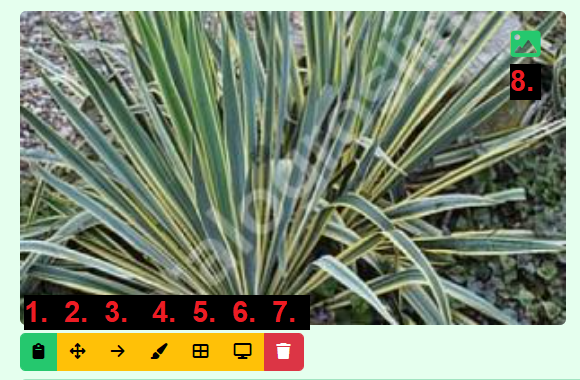
##### Przeniesienie rośliny do innej działki

Aby przenieść roślinę do innej działki, należy w trybie wybierania kliknąć na roślinę, a następnie kliknąć przycisk przenoszenia do innej działki, zabierający użytkownika do strony (zob. Rysunek 77). Następne czynności są takie same, jak w rozdziale (Dodanie rośliny do działki) z tym wyjątkiem, że ustawia się jedynie pozycję rośliny na innej działce, a reszta cech pozostaje niezmieniona.



Rysunek 77 Przycisk przenoszenia do innej działki i widok

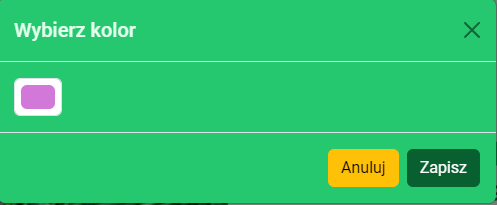
##### Zmiana właściwości rośliny

Właściwości rośliny na działce można zmieniać w jej panelu (zob. Rysunek 78), gdzie każdy przycisk może zmienić jej właściwość:

1. Dodanie notatki
2. Zmiana pozycji na działce
3. Przeniesienie do innej działki
4. Zmiana koloru kafelków
5. Zmiana tekstury kafelków
6. Zmiana trybu wyświetlania
7. Usunięcie rośliny z działki
8. Zmiana obrazu

Rysunek 78 Opcje zmiany właściwości rośliny

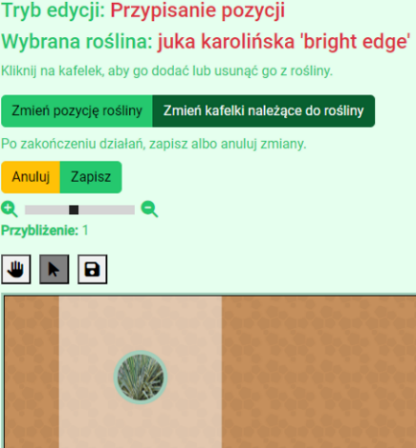
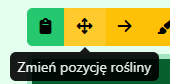
Przykładowo, można zmienić kolor kafelków rośliny, klikając na ikonę nr.4. Pojawi się wtedy okienko, gdzie wybiera się kolor (Rysunek 79), który dynamicznie zmienia się na działce po zatwierdzeniu:



Rysunek 79 Okno z wybieraniem koloru i roślina po zmianie

##### Zmiana pozycji rośliny na działce

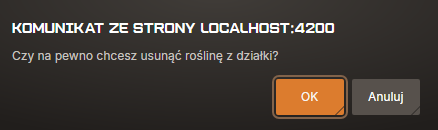
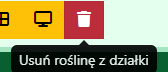
Aby zmienić pozycję rośliny na działce, należy w trybie wybierania wybrać roślinę, następnie w jej panelu wybrać opcję „Zmień pozycję rośliny” (zob. Rysunek 80). Tak jak w dodawaniu rośliny do działki, trzeba wybrać kafelki i pozycję rośliny, klikając na dany kafelek, a na końcu zapisać zmiany przyciskiem „Zapisz”. Po zapisaniu zmian roślina wraz z działką zostanie zaktualizowana:

Rysunek 80 Zmiana pozycji rośliny

##### Usuwanie rośliny z działki

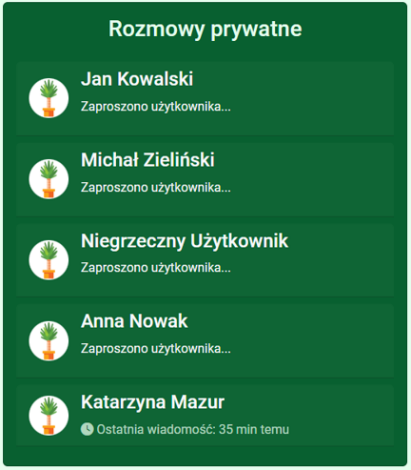
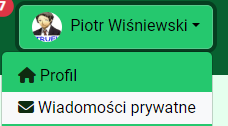
Aby usunąć roślinę z działki, należy kliknąć ikonę śmietnika i potwierdzić (zob. Rysunek 81).



Rysunek 81 Usuwanie rośliny z działki

#### Zarządzanie rozmowami

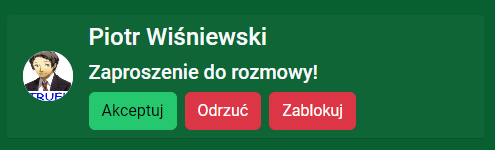
Aby przeglądać rozmowy, trzeba być zalogowanym. Można przejść do rozmów poprzez wybranie opcji „Wiadomości prywatne”. W panelu rozmów prywatnych są rozmowy użytkownika z innymi użytkownikami, a także oczekujące zaproszenia od/do.



Rysunek 82 Przejście do rozmów i rozmowy prywatne

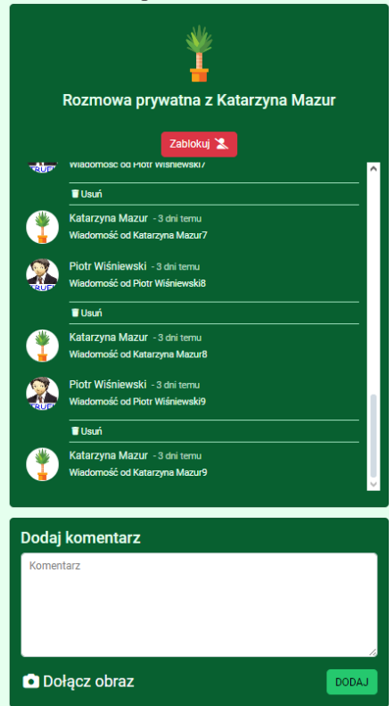
##### Wysyłanie zaproszenia

Na profilu użytkownika należy kliknąć przycisk „Zaproś”. Wysłane zaproszenie można zaakceptować, odrzucić lub zablokować nadawcę (Rysunek 83). Akceptacja zaproszenia umożliwia użytkownikom wysyłanie wiadomości do siebie nawzajem.



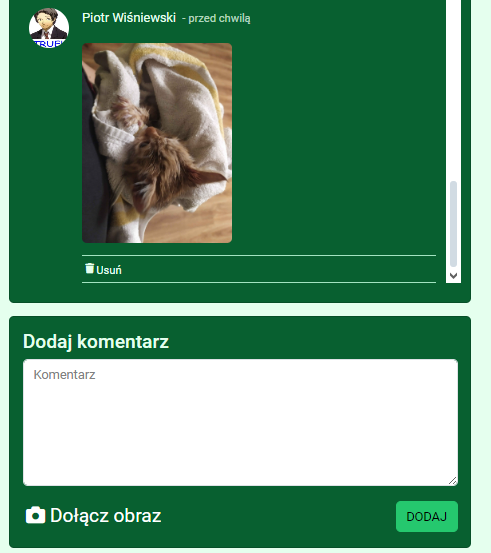
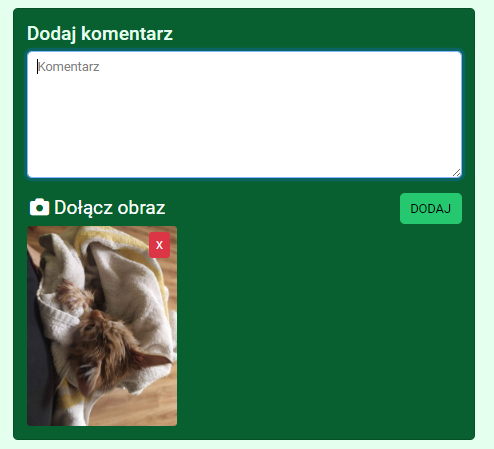
Rysunek 83 Przycisk "Zaproś" i wysłane zaproszenie

##### Wysyłanie wiadomości

Po zaakceptowaniu zaproszenia możliwe jest wysyłanie wiadomości. W liście wiadomości (zob. Rysunek 84) znajdują się wszystkie wiadomości, które są ładowane od końca mechanizmem „Infinite Scrolling”, a na końcu strony można dodać kolejną wiadomość. Ze względów na problemy techniczne z ustawianiem web socketów w Spring Boocie, nie są one tutaj używane. Zamiast tego strona co 5 sekund aktualizuje rozmowę prywatną, jeśli wystąpią jakieś zmiany.

Przykładowo, na (Rysunek 85) zostaje wysłana wiadomość ze zdjęciem, która po dodaniu pokazuje się na liście wiadomości:

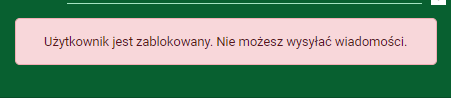
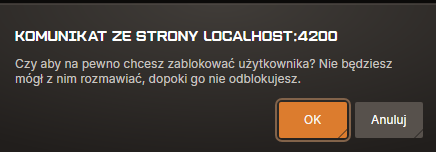
Rysunek 84 Widok rozmowy prywatnej



Rysunek 85 Dodanie wiadomości

##### Blokowanie użytkownika

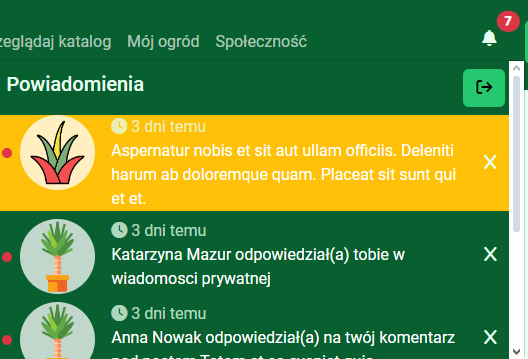
Aby zablokować użytkownika, trzeba kliknąć przycisk „Zablokuj”, a następnie potwierdzić operację (zob. Rysunek 86). Po zablokowaniu użytkownika nie jest możliwe wysyłanie do niego wiadomości, ocenianie jego postów/komentarzy ani komentowanie ich.



Rysunek 86 Potwierdzenie blokowania i wiadomość po zablokowaniu

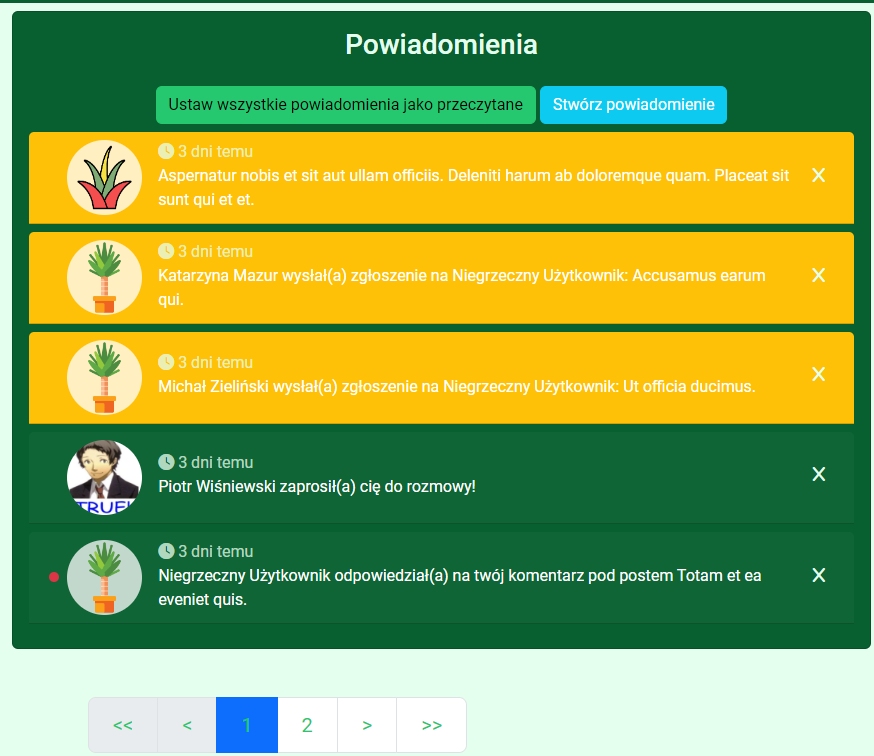
#### Zarządzanie powiadomieniami

Aby zarządzać powiadomieniami, należy na pasku nawigacyjnym kliknąć w ikonę dzwonka. Ukaże się wtedy lista powiadomień (zob. Rysunek 87), które oznaczane są jako przeczytane po najechaniu myszką i można je usuwać przyciskiem „X”. Zgłoszenia i powiadomienia specjalne są oznaczone żółtym kolorem.



Rysunek 87 Okno powiadomień

Przycisk w prawym górnym rogu otwiera stronę z powiadomieniami (zob. Rysunek 88), gdzie są one paginowane. Na stronie można oznaczyć wszystkie powiadomienia jako przeczytane lub utworzyć powiadomienie specjalne (dostępne dla pracowników i administratorów).



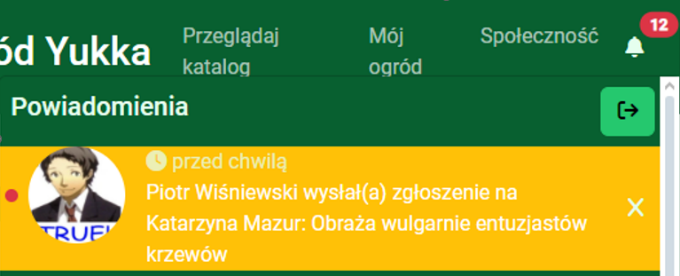
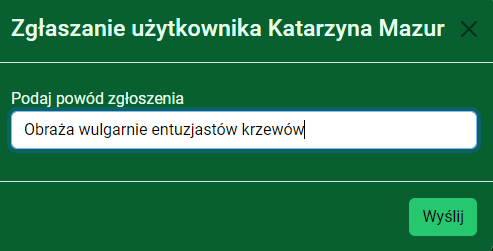
Rysunek 88 Panel powiadomień pracownika

##### Dodawanie zgłoszeń

Zgłoszenie można dodać, klikając flagę (Rysunek 89) przy poście, komentarzu czy użytkowniku na profilu. Po kliknięciu wyskoczy okno z powodem zgłoszenia. Po potwierdzeniu zgłoszenie zostanie wysłane do jednego z pracowników (zob. Rysunek 90).



Rysunek 89 Przycisk zgłaszania



Rysunek 90 Opis zgłoszenia i widok wysłanego zgłoszenia

##### Usuwanie powiadomienia

Aby usunąć powiadomienie, należy kliknąć „X” po prawej stronie powiadomienia. Operacja ta usuwa relację użytkownika z powiadomieniem, ale samo powiadomienie jest trwale usuwane dopiero wtedy, gdy żaden użytkownik nie ma z nim relacji.

**Uwaga:** Zgłoszenia nie są usuwane, lecz ukrywane, ponieważ są przypisane do pracowników z najmniejszą liczbą zgłoszeń.

### Funkcjonalności pracownika

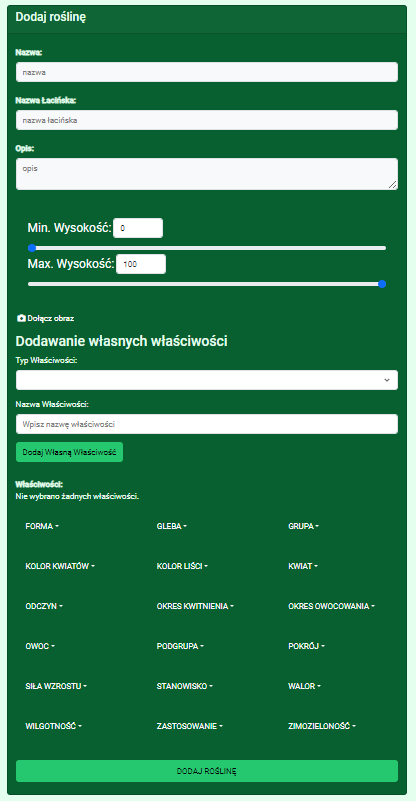
Pracownik dziedziczy wszystkie funkcjonalności użytkownika. Oprócz tego posiada swoje własne zdefiniowane w (Diagram Przypadków Użycia).

#### Zarządzanie roślinami

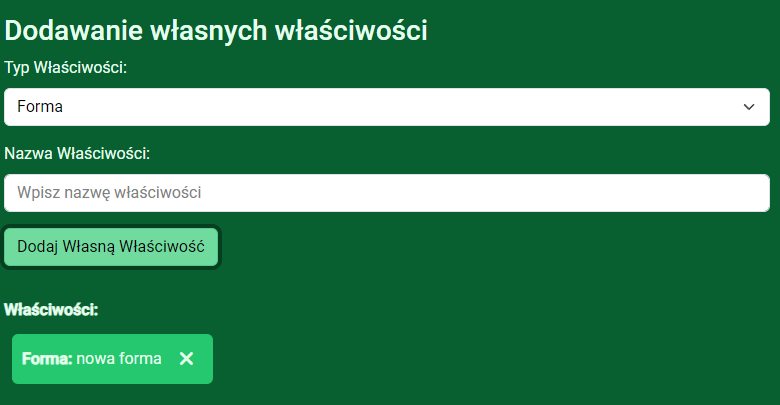
##### Dodawanie rośliny

Aby dodać roślinę do katalogu, należy w katalogu roślin kliknąć przycisk „Dodaj roślinę” przenoszący użytkownika do strony dodawania (zob. Rysunek 91). Jeśli zamiast tego chce się dodać własną roślinę, należy przejść do strony „Mój ogród”, wybrać przeglądanie swoich roślin i tam kliknąć przycisk „Dodaj roślinę”.

Na stronie dodawania rośliny należy wpisać parametry rośliny, opcjonalnie dodać obraz i przypisać wybrane rodzaje cech. Można również stworzyć jedną z cech poprzez wybranie jej typu i podanie własnej nazwy (zob. Rysunek 92).



Rysunek 91 Przycisk dodawania rośliny i formularz



Rysunek 92 Nowa właściwość rośliny

Po podaniu danych należy kliknąć przycisk zatwierdzenia, który po dodaniu rośliny przeniesie użytkownika do jej strony (zob. Rysunek 93).

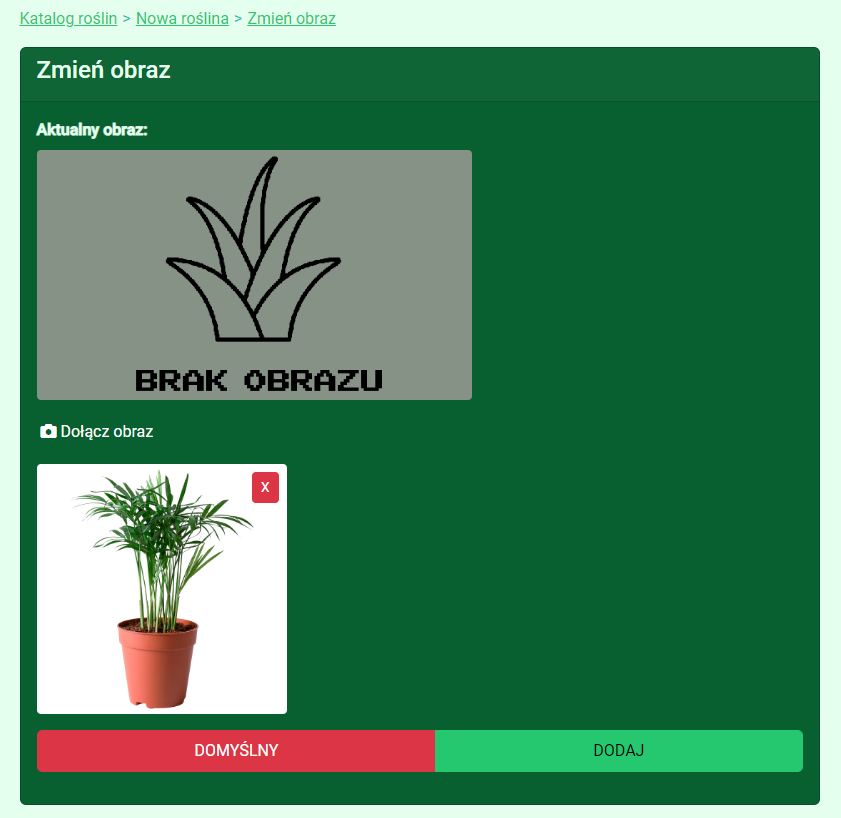


Rysunek 93 Dodana roślina

##### Aktualizacja rośliny

Aby aktualizować roślinę, należy na jej stronie kliknąć przycisk „Edytuj”, który przeniesie użytkownika do strony edycji rośliny. Jest ona podobna do strony dodawania rośliny, z tą różnicą, że nie ma opcji dodawania obrazu. Po zatwierdzeniu zmian zostanie się z powrotem przeniesionym do strony rośliny.

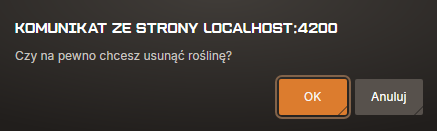
Aby zmienić obraz rośliny, należy kliknąć przycisk „Zmień obraz” na stronie rośliny, co przeniesie użytkownika do strony zmiany obrazu (zob. Rysunek 94). Tam można ustawić nowy obraz lub przywrócić domyślny. Po dokonaniu wyboru obraz rośliny zostanie zaktualizowany.

Rysunek 94 Zmiana obrazu rośliny

##### Usuwanie rośliny

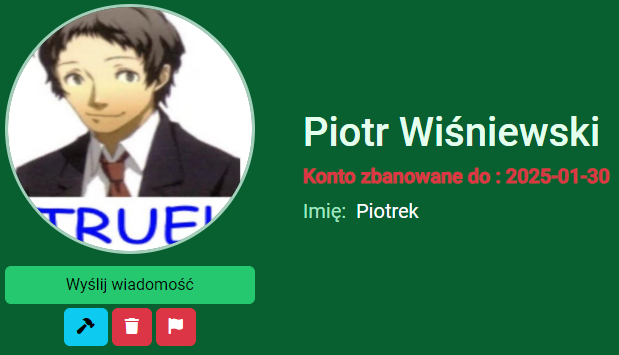
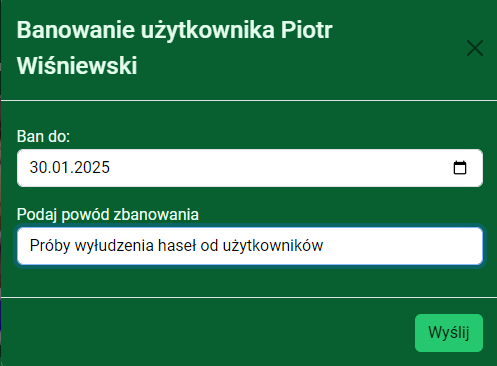
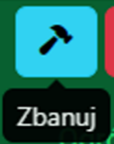
Aby usunąć roślinę, należy na jej stronie kliknąć przycisk „usuń” i zatwierdzić (zob. Rysunek 95).



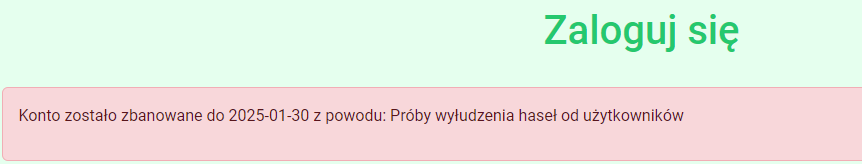
Rysunek 95 Potwierdzenie usuwania rośliny

#### Banowanie użytkowników

Pracownicy mogą banować użytkowników przechodząc do ich profilu i klikając ikonę młotka, wywołując okno banowania użytkownika, gdzie podaje się datę zakończenia oraz powód (zob. Rysunek 96). Konto zostanie automatycznie odbanowane po tej dacie. Ewentualnie, pracownik może odbanować go ręcznie klikając jeszcze raz na ikonę młotka i potwierdzając.



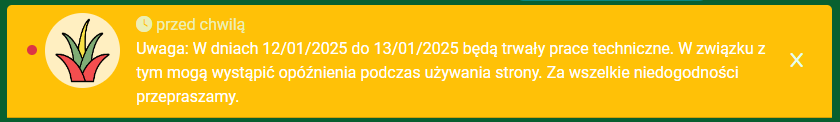
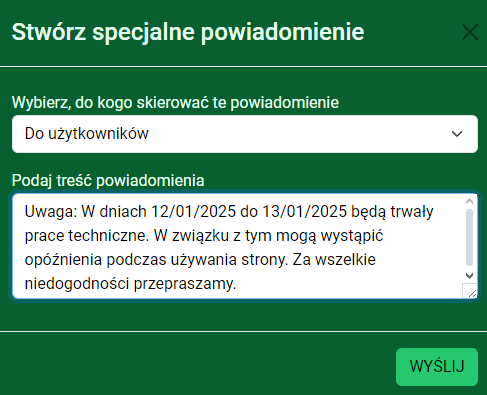
Rysunek 96 Formularz banowania i użytkownik po zbanowaniu



Rysunek 97 Komunikat o zbanowaniu przy próbie zalogowania się

#### Tworzenie specjalnego powiadomienia

Aby stworzyć specjalne powiadomienie, na stronie z powiadomieniami kliknąć przycisk „Stwórz powiadomienie”, które pokaże okienko, gdzie trzeba wpisać treść powiadomienia i do kogo jest ono adresowane (użytkownicy lub tylko pracownicy).

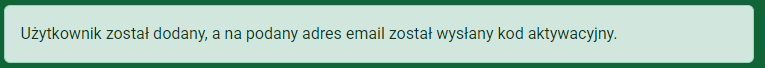
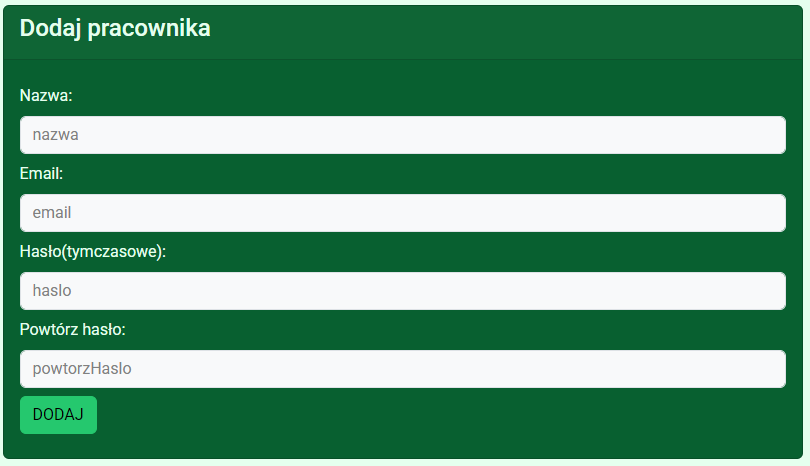


Rysunek 98 Okno specjalnego powiadomienia i stworzone powiadomienie

### Funkcjonalności administratora

#### Tworzenie konta pracownika

Aby stworzyć pracownika, należy w Społeczności wybrać opcję „Użytkownicy” i kliknąć przycisk „Dodaj pracownika” przenoszący do strony, gdzie podaje się dane użytkownika, a po zatwierdzeniu wysyłana jest wiadomość na podany adres email (zob. Rysunek 99).



Rysunek 99 Panel dodawania pracownika i komunikat po dodaniu

#### Banowanie pracowników

Banowanie pracowników działa dokładnie tak jak w rozdziale (Banowanie użytkowników), z tą różnicą, że tylko administrator ma do tego uprawnienia.

#### Wysyłanie powiadomień do pracowników

Administrator może wysyłać powiadomienia do pracowników, wybierając opcję odbiorcy jako pracowników w rozdziale (Tworzenie specjalnego powiadomienia).

### Zarządzanie zasobami

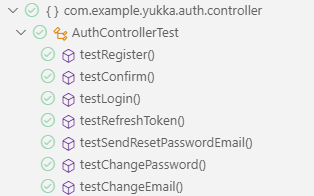
## Testy

W backendowej części aplikacji napisano testy integracyjne mające sprawdzać działanie programu w przypadkach, gdzie wymagany jest udział wielu komponentów. Testy integracyjne sprawdzają, jak różne części systemu współpracują ze sobą. Są one uruchamiane po pełnym załadowaniu kontekstu aplikacji i wykonywane są rzeczywiste interakcje pomiędzy komponentami.

Klasy testowe w tej aplikacji składają się z:

* Użytkownika i jego autoryzacji
* Jednego lub więcej obiektów testowych
* Funkcji wykonywującej się przed wszystkimi testami, tworząca dane testowe w bazie
* Funkcji testowych wykonujących się w określonej kolejności
* Funkcji kończącej, mająca za zadanie wyczyszczenie bazy z danych testowych

W panelu testowym można łatwo uruchamiać wybrane testy lub ich grupy. Każdy test posiada oznaczenie, czy się powiódł, a także ile czasu zajęło wykonanie danej funkcji. Przykładowo, AuthControllerTest obejmuje testy dotyczące AuthController, gdzie testuje się kolejno rejestrację, aktywację konta, logowanie, odświeżanie tokenu oraz zmianę hasła/emaila.



Rysunek 100 Lista testów AuthControllerTest

Przykładowo, funkcja testLogin() (zob. Rysunek 101) testuje logowanie użytkownika, gdzie tworzony jest request z danymi użytkownika zdefiniowanego w klasie testowej, który jest wysyłany do kontrolera, a potem sprawdza się poprawność odpowiedzi.



Rysunek 101 Test logowania

## Podsumowanie

Aplikacja internetowa „Wirtualny Ogród Yukka” jest aplikacją internetową umożliwiającą zarządzanie wirtualnym ogrodem oraz interakcję społecznościową między użytkownikami.

**Spełnione funkcjonalności:**

* Zarządzanie roślinami w oficjalnym katalogu bądź tworzenie własnych, pozwalając na dużą swobodę w dostosowaniu ich cech.
* Projektowanie ogrodu z własnymi działkami, gdzie zasadza się rośliny z katalogu lub własne, przydzielając im obszar i dostosowując wygląd.
* Tworzenie postów, komentowanie, ocenianie oraz prowadzenie rozmów prywatnych.
* System powiadomień i zgłoszeń informujący użytkowników i pracowników o zdarzeniach w aplikacji.
* Rejestracja, aktywacja konta, logowanie, zarządzanie kontem i informacjami profilowymi, a także banowanie użytkowników przez pracowników i administratorów.

Aplikacja zawiera testy integracyjne sprawdzające poprawność działania poszczególnych komponentów oraz ich współpracę. Testy obejmują m.in. rejestrację, logowanie, zarządzanie roślinami, ogrodem, postami i komentarzami.

Planowany rozwój obejmuje dalsze usprawnienia interfejsu użytkownika (w szczególności ogrodu), zwiększyć możliwości modyfikacji roślin na działce, optymalizację wydajności, poprawę prowadzenia rozmów prywatnych poprzez implementację websocketów w celu prowadzenia rozmów w czasie rzeczywistym, dodanie galerii zdjęć użytkownika, rozszerzenie postów poprzez możliwość dodawania wielu obrazów naraz, a także optymalizacja pobierania komentarzy poprzez użycie paginacji albo Infinite Scrolling, mając również na uwadze ilość zagnieżdżonych odpowiedzi.

## Literatura

Bruggen, R. V. (2014). *Learning Neo4j. Run blazingly fast queries on complex graph datasets with the power of the Neo4j graph database.* Packt Publishing.

Greg L. Turnquist, e. a. (2022). *Learning Spring Boot 3.0. Simplify the development of production-grade applications using Java and Spring - Third Edition.* Packt Publishing.

Joshi, S. (2024). *Full Stack Development with Angular and Spring Boot.* BPB Publications.

Kathy Sierra, e. a. (2023). *Java. Rusz głową! Wydanie III.* Helion.

Lamis Chebbi, e. l. (2024). *Reactive Patterns with RxJS and Angular Signals.* Packt Publishing.

Marrs, T. (2017). *JSON at Work. Practical Data Integration for the Web.* O'Reilly Media.

**Spis ilustracji**

[Rysunek 1 Diagram Przypadków Użycia 8](#_Toc187930877)

[Rysunek 2 Diagram Architektura klient-serwer 11](#_Toc187930878)

[Rysunek 3 Funkcja get\_all\_plant\_info z pliku scrap\_threads.py 13](#_Toc187930879)

[Rysunek 4 Węzeł Roslina i węzeł Roslina z etykietą RoslinaWlasna 14](#_Toc187930880)

[Rysunek 5 Węzeł Cecha i węzeł Cecha z etykietą CechaWlasna 16](#_Toc187930881)

[Rysunek 6 Węzeł ogrodu 16](#_Toc187930882)

[Rysunek 7 Węzeł Dzialka 17](#_Toc187930883)

[Rysunek 8 Węzeł Uzytkownik oraz węzły z etykietami Pracownik i Admin 18](#_Toc187930884)

[Rysunek 9 Węzeł Ustawienia 18](#_Toc187930885)

[Rysunek 10 Węzeł Post 19](#_Toc187930886)

[Rysunek 11 Węzeł Komentarz 19](#_Toc187930887)

[Rysunek 12 Węzeł RozmowaPrywatna 19](#_Toc187930888)

[Rysunek 13 Węzeł Powiadomienie bez i z etykietą Zgloszenie 20](#_Toc187930889)

[Rysunek 14 Relacja ZASADZONA\_NA 23](#_Toc187930890)

[Rysunek 15 Relacja OCENIL 24](#_Toc187930891)

[Rysunek 16 Relacja POWIADAMIA 24](#_Toc187930892)

[Rysunek 17 Struktura backendu 25](#_Toc187930893)

[Rysunek 18 Katalog uploads 25](#_Toc187930894)

[Rysunek 19 Katalog resources 26](#_Toc187930895)

[Rysunek 20 Kod obiektu Oceniany 26](#_Toc187930896)

[Rysunek 21 Kod obiektu Ocenil 26](#_Toc187930897)

[Rysunek 22 Kod obiektu Post 27](#_Toc187930898)

[Rysunek 23 Żądania dla kontrolera Post w Swaggerze 28](#_Toc187930899)

[Rysunek 24 Widok żądania GET /posty 29](#_Toc187930900)

[Rysunek 25 Odpowiedź żądania 29](#_Toc187930901)

[Rysunek 26 Kod funkcji w kontrolerze 30](#_Toc187930902)

[Rysunek 27 Kod funkcji serwisu 30](#_Toc187930903)

[Rysunek 28 Zapytanie do bazy danych 30](#_Toc187930904)

[Rysunek 29 Odpowiedź żądania pobrania postów 31](#_Toc187930905)

[Rysunek 30 Folder z logami 31](#_Toc187930906)

[Rysunek 31 Log informacyjny 31](#_Toc187930907)

[Rysunek 32 Log ostrzeżenia 32](#_Toc187930908)

[Rysunek 33 Log błędu 32](#_Toc187930909)

[Rysunek 34 Kod codziennego sprawdzania banów 32](#_Toc187930910)

[Rysunek 35 Struktura frontendu 33](#_Toc187930911)

[Rysunek 36 Komponent roslina-card oraz jego widok 34](#_Toc187930912)

[Rysunek 37 Kod roslina-card.component.ts 34](#_Toc187930913)

[Rysunek 38 Kod roslina-card.component.html 34](#_Toc187930914)

[Rysunek 39 Przykładowa pętla tworząca karty roślin 35](#_Toc187930915)

[Rysunek 40 Fragment kodu z authentication.service.ts 35](#_Toc187930916)

[Rysunek 41 funkcja login w login.ts 35](#_Toc187930917)

[Rysunek 42 Panel logowania i rejestracji 36](#_Toc187930918)

[Rysunek 43 Panel aktywacji i kod z wiadomości email 36](#_Toc187930919)

[Rysunek 44 Widok po zalogowaniu się 36](#_Toc187930920)

[Rysunek 45 Przejście do ustawień i do zmiany avatara 37](#_Toc187930921)

[Rysunek 46 Panel zmiany avatara i profil po zmianie 37](#_Toc187930922)

[Rysunek 47 Panel edycji profilu przed i po podaniu danych 37](#_Toc187930923)

[Rysunek 48 Profil po edycji 38](#_Toc187930924)

[Rysunek 49 Panel ustawień konta 38](#_Toc187930925)

[Rysunek 50 Komunikat o zmianie ustawień i profil po zmianie 38](#_Toc187930926)

[Rysunek 51 Panel zmiany adresu email 39](#_Toc187930927)

[Rysunek 52 Kod aktywacyjny i komunikat po podaniu 39](#_Toc187930928)

[Rysunek 53 Link do zmiany hasła i formularz 39](#_Toc187930929)

[Rysunek 54 Formularz zmiany hasła 40](#_Toc187930930)

[Rysunek 55 Panel usuwania konta i potwierdzenie 40](#_Toc187930931)

[Rysunek 56 Widok katalogu roślin 41](#_Toc187930932)

[Rysunek 57 Lista cech typu Gleba przed i po wyszukaniu 41](#_Toc187930933)

[Rysunek 58 Widok tagów cech 41](#_Toc187930934)

[Rysunek 59 Widok strony rośliny 42](#_Toc187930935)

[Rysunek 60 Wyszukiwarka postów i przykłady postów 43](#_Toc187930936)

[Rysunek 61 Widok komentarzy postu 43](#_Toc187930937)

[Rysunek 62 Przejście do postów użytkownika 43](#_Toc187930938)

[Rysunek 63 Formularz dodania postu i post po dodaniu 44](#_Toc187930939)

[Rysunek 64 Usuwanie postu 44](#_Toc187930940)

[Rysunek 65 Dodawanie komentarza i dodany komentarz 44](#_Toc187930941)

[Rysunek 66 Edycja komentarza i komentarz po edycji 45](#_Toc187930942)

[Rysunek 67 Dodawanie odpowiedzi i odpowiedź po edycji 45](#_Toc187930943)

[Rysunek 68 Ocena postu/komentarza 45](#_Toc187930944)

[Rysunek 69 Ikona usuwania 46](#_Toc187930945)

[Rysunek 70 Lista użytkowników i profil jednego z nich 46](#_Toc187930946)

[Rysunek 71 Widok ogrodu użytkownika 46](#_Toc187930947)

[Rysunek 72 Widok działki 47](#_Toc187930948)

[Rysunek 73 Potwierdzenie zapisu i stworzony obraz działki 47](#_Toc187930949)

[Rysunek 74 Zmiana nazwy ogrodu 48](#_Toc187930950)

[Rysunek 75 Przycisk "Dodaj do działki" 48](#_Toc187930951)

[Rysunek 76 Strona dodawania rośliny do działki i dodana roślina 49](#_Toc187930952)

[Rysunek 77 Przycisk przenoszenia do innej działki i widok 49](#_Toc187930953)

[Rysunek 78 Opcje zmiany właściwości rośliny 50](#_Toc187930954)

[Rysunek 79 Okno z wybieraniem koloru i roślina po zmianie 50](#_Toc187930955)

[Rysunek 80 Zmiana pozycji rośliny 51](#_Toc187930956)

[Rysunek 81 Usuwanie rośliny z działki 51](#_Toc187930957)

[Rysunek 82 Przejście do rozmów i rozmowy prywatne 51](#_Toc187930958)

[Rysunek 83 Przycisk "Zaproś" i wysłane zaproszenie 52](#_Toc187930959)

[Rysunek 84 Widok rozmowy prywatnej 52](#_Toc187930960)

[Rysunek 85 Dodanie wiadomości 53](#_Toc187930961)

[Rysunek 86 Potwierdzenie blokowania i wiadomość po zablokowaniu 53](#_Toc187930962)

[Rysunek 87 Okno powiadomień 53](#_Toc187930963)

[Rysunek 88 Panel powiadomień pracownika 54](#_Toc187930964)

[Rysunek 89 Przycisk zgłaszania 54](#_Toc187930965)

[Rysunek 90 Opis zgłoszenia i widok wysłanego zgłoszenia 54](#_Toc187930966)

[Rysunek 91 Przycisk dodawania rośliny i formularz 56](#_Toc187930967)

[Rysunek 92 Nowa właściwość rośliny 56](#_Toc187930968)

[Rysunek 93 Dodana roślina 57](#_Toc187930969)

[Rysunek 94 Zmiana obrazu rośliny 57](#_Toc187930970)

[Rysunek 95 Potwierdzenie usuwania rośliny 57](#_Toc187930971)

[Rysunek 96 Formularz banowania i użytkownik po zbanowaniu 58](#_Toc187930972)

[Rysunek 97 Komunikat o zbanowaniu przy próbie zalogowania się 58](#_Toc187930973)

[Rysunek 98 Okno specjalnego powiadomienia i stworzone powiadomienie 58](#_Toc187930974)

[Rysunek 99 Panel dodawania pracownika i komunikat po dodaniu 59](#_Toc187930975)

[Rysunek 100 Lista testów AuthControllerTest 60](#_Toc187930976)

[Rysunek 101 Test logowania 60](#_Toc187930977)

**Streszczenie**

**Aplikacja internetowa do zarządzania ogrodem z elementami społecznościowymi**

**(Web Application for Garden Management with Social Elements)**

Celem pracy jest stworzenie aplikacji internetowej do zarządzania ogrodem z elementami społecznościowymi, zawierającą zarządzanie roślinami ze zróżnicowanymi cechami, projektowanie ogrodu podzielonego na działki w postaci siatek kafelków, na których można sadzić rośliny i je modyfikować, a także stworzenie części społecznościowej w formie postów, komentarzy, rozmów prywatnych, powiadomień o wydarzeniach, a także możliwości moderacji społeczności przy użyciu zgłoszeń i banowania. Wymienione cele zostały zrealizowane. Praca zawiera dokładny opis struktury aplikacji, procesu implementacji oraz przykłady użycia funkcjonalności.

1. Joshi, S. (2024). Full Stack Development with Angular and Spring Boot. BPB Publications. [↑](#footnote-ref-1)
2. Kathy Sierra, e. a. (2023). Java. Rusz głową! Wydanie III. Helion. [↑](#footnote-ref-2)
3. Greg L. Turnquist, e. a. (2022). Learning Spring Boot 3.0. Simplify the development of production-grade applications using Java and Spring - Third Edition. Packt Publishing. [↑](#footnote-ref-3)
4. Bruggen, R. V. (2014). Learning Neo4j. Run blazingly fast queries on complex graph datasets with the power of the Neo4j graph database. Packt Publishing. [↑](#footnote-ref-4)
5. Marrs, T. (2017). JSON at Work. Practical Data Integration for the Web. O'Reilly Media. [↑](#footnote-ref-5)
6. Lamis Chebbi, e. l. (2024). Reactive Patterns with RxJS and Angular Signals. Packt Publishing. [↑](#footnote-ref-6)