

Podstawy inżynierii programowania - Projekt

Grupa: 2 Temat: **Gospodarstwo rolne**

Opis systemu:

System do zarządzania gospodarstwem rolnym, ułatwiający rolnikom zarządzanie zasobami, kontrolę i śledzenie postępów prac. System zapewni właścicielowi generowanie raportów dotyczących poszczególnych jednostek gospodarstwa. To pozwoli na dokładną analizę i optymalizowanie osiągnięć gospodarstwa. Da możliwość zdalnie zarządzać zadaniami pracowników i monitorować ich postęp. Śledzenie stanu magazynów oraz informacje o poszczególnych polach, zaoszczędzi dużo czasu właścicielom gospodarstw. System umożliwi kontrolę prac na polach, poprzez aktualizowanie wykonanych prac przez pracowników, jak i samodzielną edycję i aktualizację przez właściciela. Duże gospodarstwa mogą posiadać mapę, ułatwiającą poruszanie się nowych pracowników oraz podstawowe informacje dotyczące pól. Każdy z użytkowników systemu ma możliwość wyszukania skupów znajdujących się najbliżej gospodarstwa lub w określonej odległości. Dodatkowo system musi powiadamiać użytkowników o pilnych zadaniach oraz specjalnych ofertach w sąsiednich skupach z produktami rolnymi.

Role użytkowników:

- Właściciel - wgląd we wszelkie informacje o gospodarstwie, możliwość przydzielania zadań pracownikom, aktualizacja danych o zasobach, anulacja prac, edytowanie danych
- Pracownik - wgląd na przydzielone zadania i aktualizacja postępów nad wykonanymi pracami, sprawdzenie stanu magazynów

Byty wykorzystywane w systemie:

- Działka - informacje o zasiewach, rodzaj gleby, data zasiewu;
- Pracownik – liczba godzin, dostępność, płaca, kwalifikacje, możliwość przypisania zadań;
- Jednostka sprzętu - rodzaj, stan, koszty amortyzacji, dostępność;
- Jednostka hodowlana – wiek, rodzaj, waga;
- Jednostka zasobu – rodzaj, cena, ilość;

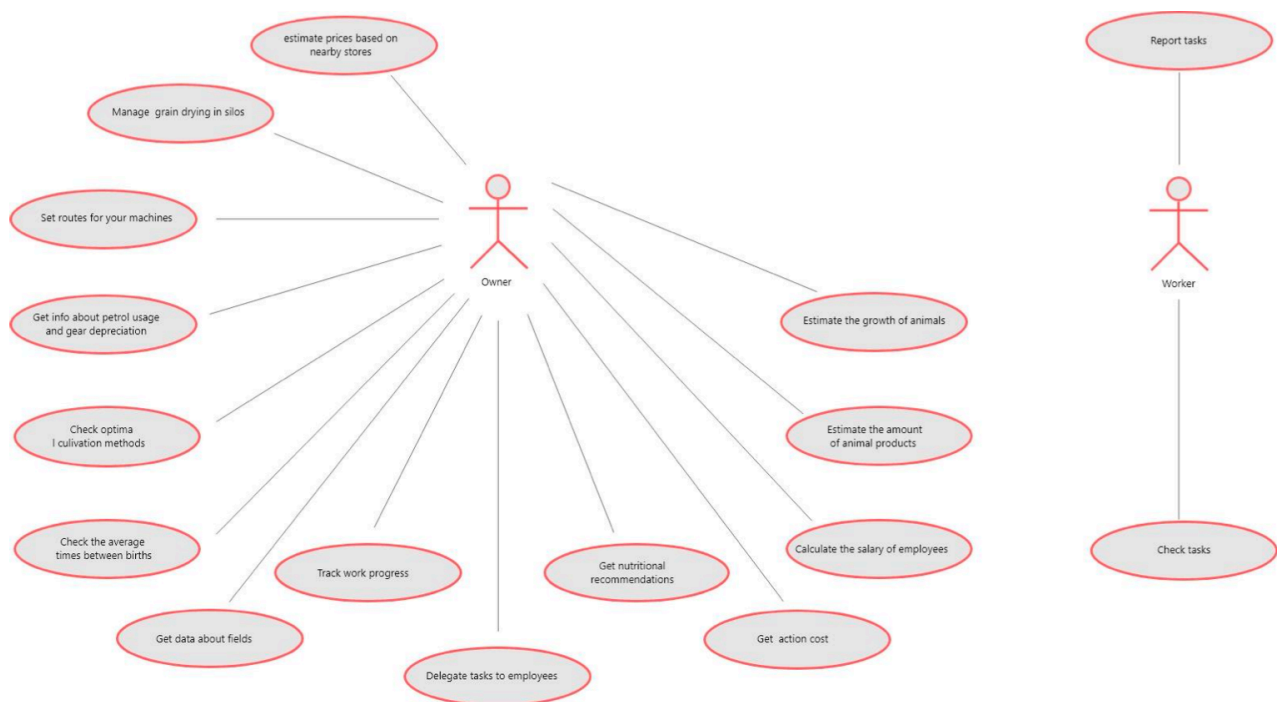
Wymagania funkcjonalne:

- Przegląd ofert skupów
- Mapa skupów
- Wyszukiwarka produktów
- Przegląd skupów
- Rezerwacja pól
- Wystawy sprzedaży produktów z silosów
- Tworzenie wystaw przez użytkowników
- Informacje o stanie gleby na danym polu
- Informacje o czasie zasiewu, rodzaju uprawy
- Przewidywanie przyrostów mas zwierząt
- Możliwość pobrania raportu
- Ilość uzyskiwanych produktów w przedziale czasu
- Informacje o pensji pracownika
- Sprawdzenie stanu silosa
- Połączenie z API serwisów globalnych do śledzenia cen rynkowych
- Dostępność pracowników
- Kontakt z właścicielem
- Tworzenie raportu

Założenia нефunkcyjne:

- System powinien być w stanie obsługiwać do 10000 użytkowników jednocześnie
- Czas oczekiwania na dane przez użytkownika nie powinien wynosić więcej niż 2 sekundy
- Aplikacja powinna posiadać API
- Aplikacja powinna działać płynnie dla przeciętnego użytkownika
- Użytkownik indywidualny powinien móc otrzymać większość oczekiwanych przez niego funkcjonalności przy maksymalnie dwóch kliknięciach myszą
- Aplikacja powinna być dostępna 24 godziny na dobę 7 dni w tygodniu

Diagram przypadków użycia:



Scenariusz przypadków użycia:

Use Case:	Wyznacz sugerowane ceny towarów
Language:	Polish
Preconditions:	<ul style="list-style-type: none">• Użytkownik musi być zalogowany jako właściciel
Postconditions:	<ul style="list-style-type: none">• Właściciel dostaje od systemu dane wyliczone na podstawie cen z pobliskich skupów
Actors:	<ul style="list-style-type: none">• Właściciel
Main flow:	<ol style="list-style-type: none">1. Właściciel wybiera opcję sprawdź sugerowaną cenę2. System sprawdza ceny w skupach3. System oblicza cenę na podstawie określonego algorytmu4. System wyświetla sugerowaną cenę5. System sprawdza odległości do skupów6. System przelicza koszt transportu i wybiera najlepszy skup
Alternate flow:	<ul style="list-style-type: none">• 2 Brak towaru w skupach GOTO 1

Use Case:	Oszacuj dane na temat posiadanych zwierząt
Language:	Polish
Preconditions:	<ul style="list-style-type: none">• Użytkownik musi być zalogowany jako właściciel
Postconditions:	<ul style="list-style-type: none">• Użytkownik otrzymuje estymaty przyrostów• Użytkownik otrzymuje estymaty ilości produktów
Actors:	<ul style="list-style-type: none">• Właściciel
Main flow:	<ol style="list-style-type: none">1. Właściciel wybiera opcję oszacuj przyrosty zwierząt2. System na podstawie danych o żywieniu i wieku zwierząt szacuje przyrosty3. Właściciel wybiera opcję oszacuj ilość produktów odzwierzęcych4. System na podstawie danych o żywieniu i wieku zwierząt szacuje ilość produktów
Alternate flow:	<ul style="list-style-type: none">• END FLOW

Use Case:	Oblicz pensje pracowników
Language:	Polish
Preconditions:	<ul style="list-style-type: none"> • Użytkownik musi być zalogowany jako właściciel
Postconditions:	<ul style="list-style-type: none"> • Właściciel otrzymuje dane o pensji dla każdego z pracowników
Actors:	<ul style="list-style-type: none"> • Właściciel
Main flow:	<ol style="list-style-type: none"> 1. Właściciel wybiera pracownika dla którego chce obliczyć pensję 2. Sprawdzanie pracownika 3. System na podstawie danych o godzinach pracy i wykonanych zadaniach oblicza pensję dla każdego z pracowników 4. System na podstawie danych o godzinach oblicza składki i ubezpieczenie dla każdego pracownika
Alternate flow:	<ul style="list-style-type: none"> • 2 Brak takiego pracownika GOTO 1

Use Case:	Sprawdzanie danych o silosach
Language:	Polish
Preconditions:	<ul style="list-style-type: none"> • Użytkownik musi być zalogowany jako właściciel
Postconditions:	<ul style="list-style-type: none"> • Właściciel otrzymuje dane odnośnie wolnego miejsca w silosach • Właściciel otrzymuje dane odnośnie suszenia w silosach
Actors:	<ul style="list-style-type: none"> • Właściciel
Main flow:	<ol style="list-style-type: none"> 1. Właściciel wybiera suszenie ziarna w silosach 2. System na podstawie danych o ilości ziarna w silosach oblicza ilość wolnego w danym silosie 3. Właściciel otrzymuje dane czy ziarno w silosie jest wysuszone 4. System na podstawie danych o suszeniu informuje czy można przetwarzać je dalej
Alternate flow:	<ul style="list-style-type: none"> • 2 Brak miejsca w silosie GOTO 1

Diagram klas:

