



BREWIT

Dokumentacja projektowa PZSP2

WERSJA 3.0

10.01.2025

Semestr 2024Z

Zespół nr 18 w składzie:

Jakub Bąba

Gregory Chmielewski

Adrian Murawski

Aleksandra Szymańska

Mentor zespołu: dr hab. inż. Robert Nowak, prof. uczelni

Właściciel tematu: dr inż. Piotr Pałka

Spis treści

1 Wprowadzenie	2
1.1 Cel projektu	2
1.2 Wstępna wizja projektu	2
2 Metodologia wytwarzania	2
3 Analiza wymagań	3
3.1 Wymagania użytkownika i biznesowe	4
3.2 Wymagania funkcjonalne i нефункционалне	6
3.3 Przypadki użycia	7
3.4 Potwierdzenie zgodności wymagań	9
4 Definicja architektury	10
5 Dane trwałe	13
5.1 Model logiczny danych	13
5.2 Przetwarzanie i przechowywanie danych	14
6 Specyfikacja analityczna i projektowa	14
7 Projekt standardu interfejsu użytkownika	15
8 Specyfikacja testów	21
9 <i>Wirtualizacja/konteneryzacja</i>	23
10 Bezpieczeństwo	23
11 Podręcznik użytkownika	23
12 Podręcznik administratora	25
13 Podsumowanie	26
14 Bibliografia	27

1 Wprowadzenie

1.1 Cel projektu

Celem projektu jest stworzenie systemu informatycznego dla właścicieli i pracowników browarów komercyjnych i kontraktowych, który będzie wspomagał kontrakty między wieloma browarami kontraktowymi i wieloma browarami komercyjnymi. Kontrakty takie polegają na wynajmie przez browar kontraktowy części urządzeń browaru komercyjnego na pewien czas, aby wyprodukować piwo.

1.2 Wstępna wizja projektu

Aplikacja umożliwia tworzenie zleceń przez browary kontraktowe dla browarów komercyjnych, które ułatwiają istniejący proces biznesowy wynajmowania urządzeń browaru komercyjnego przez browary kontraktowe.

Browar komercyjny dodaje swoje urządzenia, a następnie tworzy w systemie okna czasowe na te urządzenia, które informują o dostępności tych urządzeń w podanym terminie na wynajem ze strony browaru kontraktowego.

Browar kontraktowy przegląda oferty browarów komercyjnych i wybiera odpowiednie dla siebie okna czasowe na konkretne urządzenia (znajdujące się w obrębie jednego browaru komercyjnego). Kiedy doda wszystkie potrzebne sprzęty, może utworzyć zlecenie w systemie, które spowoduje tymczasową rezerwację okien czasowych, oraz wyświetlanie zlecenia browarowi komercyjnemu.

Browar komercyjny, po otrzymaniu zlecenia, akceptuje je (co powoduje zmianę statusu okna czasowego), co będzie widoczne dla browaru komercyjnego. Oznacza to przystąpienie obu podmiotów do współpracy.

Po zakończeniu współpracy między tymi podmiotami, możliwa jest ocena wykonanej warki przez browar kontraktowy.

W aplikacji planowane jest zrealizowanie również operacji wspomagających ten cel - wyświetlanie urządzeń, okien czasowych, szczegółów konkretnych browarów kontraktowych i komercyjnych i innych (wszystkie opisane w niniejszym dokumencie, w sekcji wymagań).

2 Metodologia wytwarzania

W projekcie praca jest oparta na metodyce Scrum, z około dwutygodniowymi sprintami. Ogólny plan prac na cały semestr jest zdefiniowany w terminarzu, znajdującym się w repozytorium projektu na serwerze Github. Co około dwa tygodnie organizowane są spotkania z mentorem projektu (za pomocą MS Teams lub na żywo), na których podsumowywane są wykonane prace od ostatniego spotkania, wyjaśniane są dodatkowe wymagania i wątpliwości, a także przedstawiany jest plan prac na najbliższy czas. Zadania będą definiowane w Github Issues, na stronie projektu.

Praca nad projektem będzie zawierała proces regularnego testowania, planowane jest sprawdzanie pokrycia kodu. Niezbędne procesy (m.in. testowanie) będą wykonywane za pomocą Github Actions.

Planowany podział prac zakłada rozdzielanie zadań na zadania frontendowe i zadania backendowo-bazodanowe. Do każdego z typów zadań przydzielone są po dwie osoby, jednakże taki podział ról nie jest nienaruszalny i w ramach aktualnego zapotrzebowania członkowie zespołu mogą być przydzielani do zadań wykraczających poza ich ustalone role.

3 Analiza wymagań

Typy użytkowników

- browar kontraktowy
- browar komercyjny
- spółka (pośrednik)

Typy urządzeń wraz z parametrami

- Tanki warzelne (z zacieraniem):
 - pojemność
 - zakres temperatur
 - przeznaczony do piw kwaśnych (tak/ nie)
 - cena wynajmu
- Pojemniki fermentacyjne:
 - pojemność
 - zakres temperatury (inny dla dolnej i górnej fermentacji)
 - przeznaczony do piw kwaśnych (tak/ nie)
 - cena wynajmu
- Kotły do leżakowania:
 - pojemność
 - zakres temperatury (inny dla dolnej i górnej fermentacji)
 - przeznaczony do piw kwaśnych (tak/ nie)
 - możliwość sztucznego nagazowania (CO₂, N₂, mieszanka CO₂/N₂, nie)
 - cena wynajmu
- Urządzenia do rozlewania:
 - obsługiwane pojemniki (butelki, puszki, kegi)
 - przeznaczony do piw kwaśnych (tak/ nie)
 - możliwość sztucznego nagazowania (CO₂, N₂, mieszanka CO₂/N₂, nie)
 - cena wynajmu

Statusy okien czasowych na urządzenia

- wolne
- zarezerwowane (na czas od stworzenia zlecenia przez browar kontraktowy do zaakceptowania przez browar komercyjny)
- zajęte (po zaakceptowaniu)
- niedostępne (nieudostępnione w danym terminie)

Etapy zlecenia

- (nowe) stworzone przez browar kontraktowy (ten status dostaje każde zlecenie, które właśnie zostało utworzone przez browar kontraktowy)
- (aktualne) zlecenie, które zostało zaakceptowane przez browar komercyjny
- (przeszłe) zlecenie, które jest już wykonane (czas zakończenia jest w przeszłości), a poprzednio miało stan "aktualne"
- (odrzucone) zlecenie, które nie zostało zaakceptowane przez browar komercyjny

Elementy zlecenia

- status
- czas rozpoczęcia
- czas zakończenia
- wynajmowane urządzenia
- dane browaru kontraktowego
- dane browaru komercyjnego
- dodatkowe informacje

3.1 Wymagania użytkownika i biznesowe

Wymagania biznesowe:

Celem projektu jest automatyzacja i optymalizacja procesu wynajmu sprzętu browarniczego przez browary komercyjne dla celów browarów kontraktowych.

Browary kontraktowe i komercyjne potrzebują systemu internetowego do bezkontaktowej interakcji w celu tworzenia oraz przyjmowania zleceń na wynajem urządzeń piwowych.

Problemy do rozwiązania:

- brak przejrzystej komunikacji na temat dostępności urządzeń
- ręczne tworzenie rezerwacji

Wymagania użytkowe:

Wymagania spółki pośredniczącej:

- Spółka pośrednicząca będzie mogła się zalogować i wylogować z konta
- Spółka pośrednicząca będzie mogła przeglądać użytkowników aplikacji
- Spółka pośrednicząca będzie mogła wyświetlać okna czasowe wynajmu urządzenia browarniczego
- Spółka pośrednicząca będzie mogła przeglądać zlecenia browaru komercyjnego
- Spółka pośrednicząca będzie mogła przeglądać zlecenia browaru kontraktowego
- Spółka pośrednicząca będzie mogła wyświetlić szczegóły browaru komercyjnego (nazwa browaru, właściciel/ osoba kontaktowa, adres, nip, kontakt: tel, email, opis, statystyki)
- Spółka pośrednicząca będzie mogła wyświetlić szczegóły browaru kontraktowego (nazwa browaru, osoba kontaktowa, kontakt: tel, email, opis, statystyki)

- Spółka pośrednicząca będzie mogła wyświetlić statystyki (typy/ gatunki produkowanych piw, ilość wyprodukowanego piwa danego typu, ilość wyprodukowanego piwa ogólnie - histogramy z różną rozdzielczością czasową, liczba udanych/ nieudanych warek - na podstawie ocen dla zleceń, histogramy przychodów z podziałem na kontraktowe/ komercyjne)

Wymagania browaru komercyjnego:

- Browary komercyjne będą mogły założyć konto
- Browary komercyjne będą mogły się zalogować i wylogować z konta
- Browary komercyjne będą mogły dodawać, usuwać i przeglądać swoje urządzenia do systemu
- Browary komercyjne będą mogły dodawać, modyfikować i przeglądać okna czasowe informujące o dostępności urządzeń (w tym ustawiać jako niedostępne w przypadku awarii)
- Browary komercyjne będą mogły przeglądać zajęte okna czasowe i urządzenia
- Browary komercyjne będą mogły przeglądać swoje zlecenia
- Browary komercyjne będą mogły przeglądać szczegóły zlecenia (statusu; urządzeń, które były w nim wykorzystywane; danych dotyczących stron, typu piwa, docelowej ilości piwa)
- Browary komercyjne będą mogły wyświetlić szczegóły browaru kontraktowego (nazwa browaru, osoba kontaktowa, kontakt: tel, email, opis)
- Browary komercyjne będą mogły zmieniać status zlecenia
- Browary komercyjne będą mogły akceptować lub odrzucać zlecenia

Wymagania browaru kontraktowego:

- Browary kontraktowe będą mogły założyć konto
- Browary kontraktowe będą mogły się zalogować i wylogować z konta
- Browary kontraktowe będą mogły przeglądać browary komercyjne
- Browary kontraktowe będą mogły przeglądać informacje o browarze komercyjnym (nazwa browaru, właściciel/ osoba kontaktowa, adres, nip, kontakt: tel, email, opis, statystyki)
- Browary kontraktowe będą mogły przeglądać urządzenia browarów komercyjnych
- Browary kontraktowe będą mogły przeglądać okna czasowe wynajmu urządzenia browarniczego browaru komercyjnego
- Browary kontraktowe będą mogły dodawać i usuwać okna czasowe z koszyka
- Browary kontraktowe będą mogły wyświetlić swój koszyk
- Browary kontraktowe będą mogły tworzyć zlecenia wynajmu
- Browary kontraktowe będą mogły przeglądać swoje zlecenia
- Browary kontraktowe będą mogły przeglądać szczegóły zlecenia
- Browary kontraktowe będą mogły potwierdzić wykonanie zlecenia
- Browary kontraktowe będą mogły tworzyć, edytować, usuwać i przeglądać swoje receptury (składniki, ilość, kolejność dodawania, potrzebne urządzenia i czas ich użycia, dodatkowe wymagania, typ piwa, docelowa ilość piwa, ocena (udana, nieudana))
- Browary kontraktowe będą mogły oceniać wyprodukowane piwo
- Browary kontraktowe będą mogły wyświetlać statystyki (typy / gatunki produkowanych piw, ilość wyprodukowanego piwa danego typu, ilość wyprodukowanego piwa ogólnie - histogramy z różną rozdzielczością czasową, liczba udanych / nieudanych warek)

Wymagania systemowe:

- System umożliwi założenie konta browaru kontraktowego
- System umożliwi założenie konta browaru komercyjnego
- System umożliwi zalogowanie / wylogowanie z konta browaru kontraktowego
- System umożliwi zalogowanie / wylogowanie z konta browaru komercyjnego
- System umożliwi zalogowanie / wylogowanie z konta pośrednika
- System umożliwi wyświetlenie listy browarów komercyjnych
- System umożliwi wyświetlenie listy browarów kontraktowych
- System umożliwi wyświetlanie okien czasowych dla urządzeń danego browaru komercyjnego
- System umożliwi wyświetlanie szczegółów danego browaru komercyjnego
- System umożliwi wyświetlanie szczegółów danego browaru kontraktowego
- System umożliwi dodawanie okna czasowego do koszyka
- System umożliwi sprawdzenie warunku, że okna czasowe w koszyku są w tym samym browarze kontraktowym
- System umożliwi usunięcie okna czasowego z koszyka
- System umożliwi utworzenie zlecenia z oknami czasowymi z koszyka
- System umożliwi przeglądanie listy zleceń dla danego browaru kontraktowego
- System umożliwi przeglądanie listy zleceń dla danego browaru komercyjnego
- System umożliwi przeglądanie szczegółów konkretnego zlecenia
- System umożliwi zmianę statusu zlecenia
- System umożliwi dodawanie urządzeń do browaru komercyjnego
- System umożliwi wyświetlanie listy urządzeń dla danego browaru komercyjnego
- System umożliwi usuwanie urządzeń dla browaru komercyjnego
- System umożliwi dodawanie okien czasowych na dane urządzenie w browarze kontraktowym
- System umożliwi wyświetlanie listy urządzeń wraz z ich zajętymi oknami czasowymi
- System umożliwi ocenę piwa browarowi kontraktowemu
- System umożliwi wyświetlenie koszyka browarowi kontraktowemu

3.2 Wymagania funkcjonalne i нефункционалне

Wymagania funkcjonalne:

Wszyscy użytkownicy systemu będą mogli zakładać konta i logować się na nie.

Browary komercyjne będą mogły umieszczać w systemie informacje o dostępnych oknach czasowych na każde ze swoich urządzeń.

Browary kontraktowe będą mogły przeglądać te okna, a następnie informować o chęci rezerwacji konkretnych okien czasowych.

Browary komercyjne będą mogły widzieć zarezerwowane okna dla swoich urządzeń i reagować na te rezerwacje (w szczególności je akceptować).

Browary kontraktowe po wykonanym zleceniu będą mogły zostawić informację (ocenę) o warce.

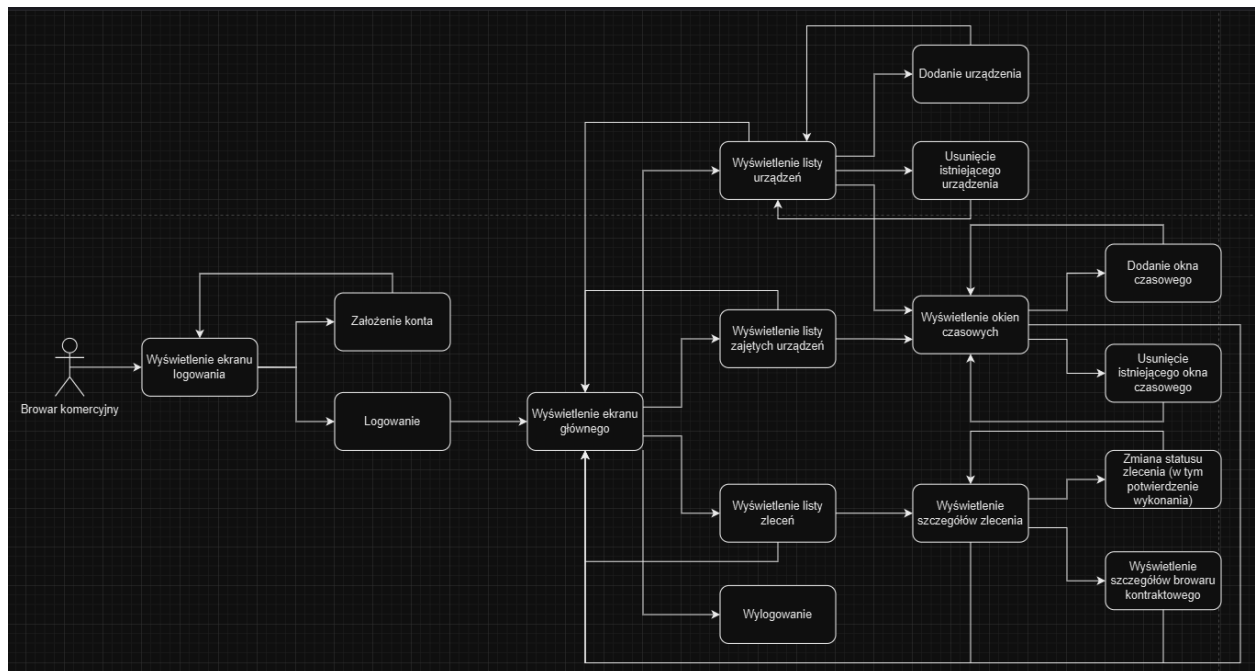
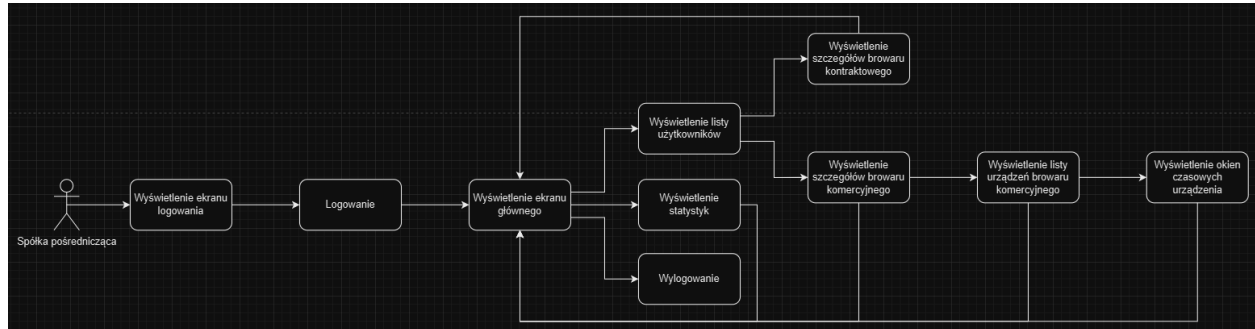
Wymagania нефункционалне:

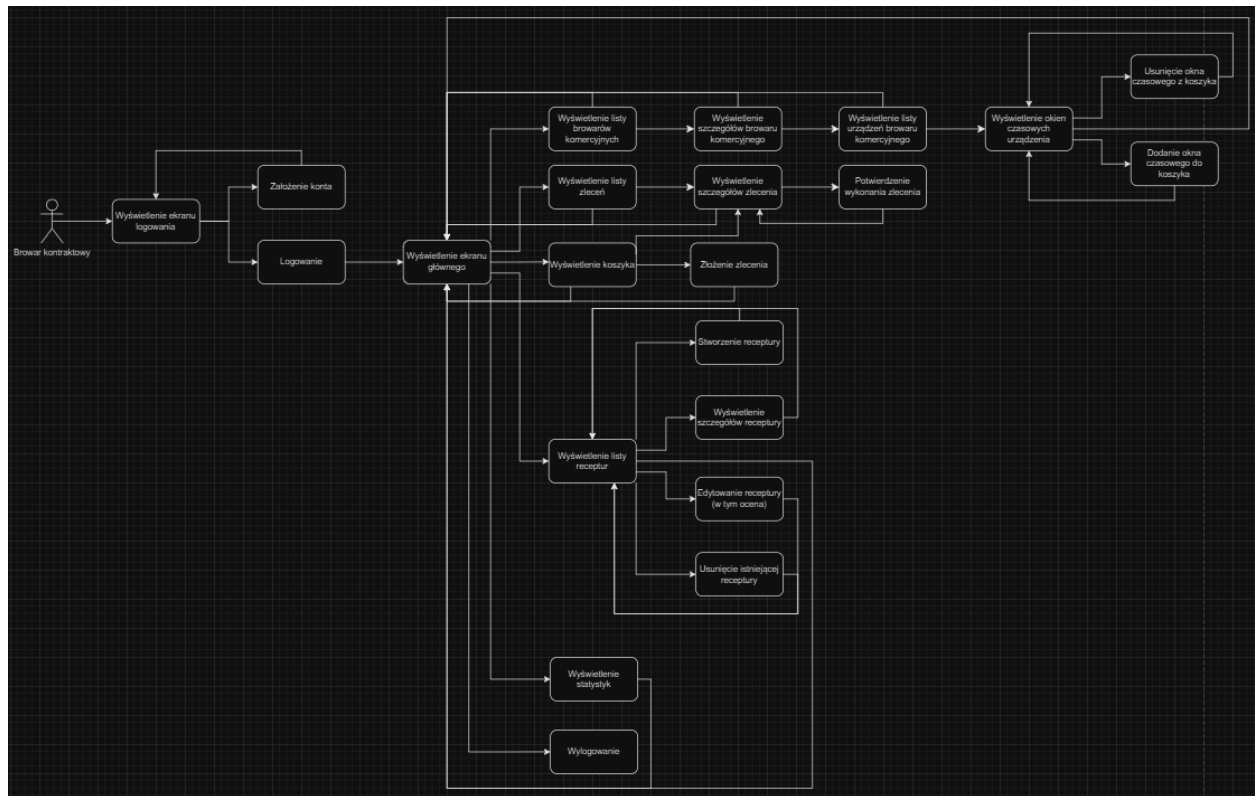
- System ma być bezpieczny (używać bezpiecznych haseł, szyfrowania HTTPS)
- System ma mieć estetyczny interfejs graficzny

- System ma być wygodny w użytkowaniu

3.3 Przypadki użycia

Proste diagramy aktywności (uwzględniają wszystkie funkcjonalności, nie uwzględniliśmy poszczególnych scenariuszy ze względu na skomplikowanie diagramu)





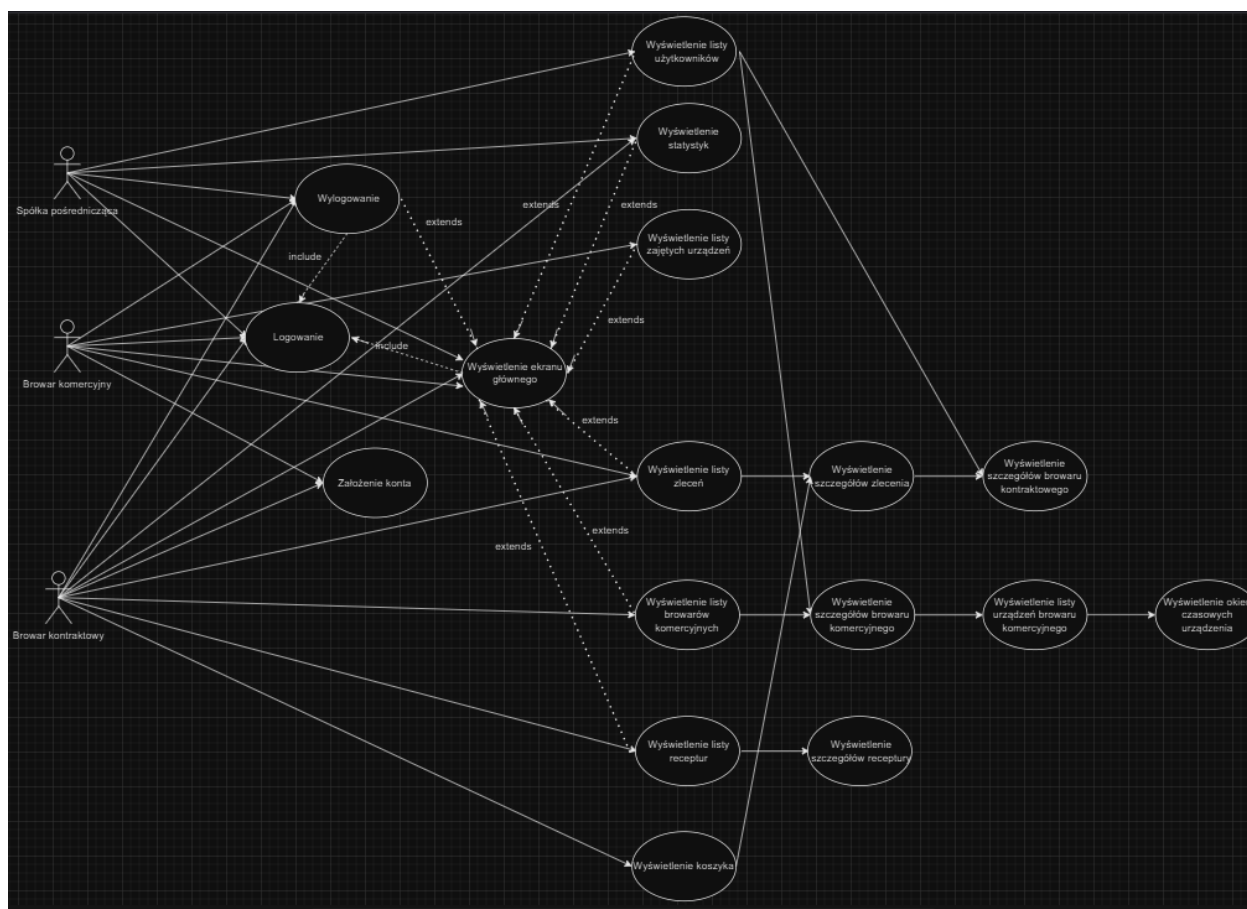
Związki między przypadkami użycia:

Po podpisaniu umowy ze spółką pośredniczącą browary komercyjne i kontraktowe zakładają konto w systemie.


Po złożeniu zlecenia przez browar kontraktowy może być ono zaakceptowane lub odrzucone przez browar komercyjny. Browary podpisują umowę i dokonują rozliczenia finansowego poza systemem.

Po zakończeniu zlecenia browar kontraktowy może dokonać oceny zlecenia, co oznacza koniec współpracy.


Diagram przypadków użycia dla przechodzenia między widokami (nie uwzględniliśmy poszczególnych funkcjonalności ze względu na skomplikowanie diagramu)






3.4 Potwierdzenie zgodności wymagań








Pałka Piotr

Do:  Szymańska Aleksandra 7 (STUD)


DW:  Bąba Jakub (STUD);  Murawski Adrian (STUD);  Chmielewski Gregory (STUD)

Dzień dobry,

Akceptuję wymagania.


 Odpowiedz
  Odpowiedz wszystkim
  Prześlij dalej
 
 ...

Cz, 14.11.2024 15:40

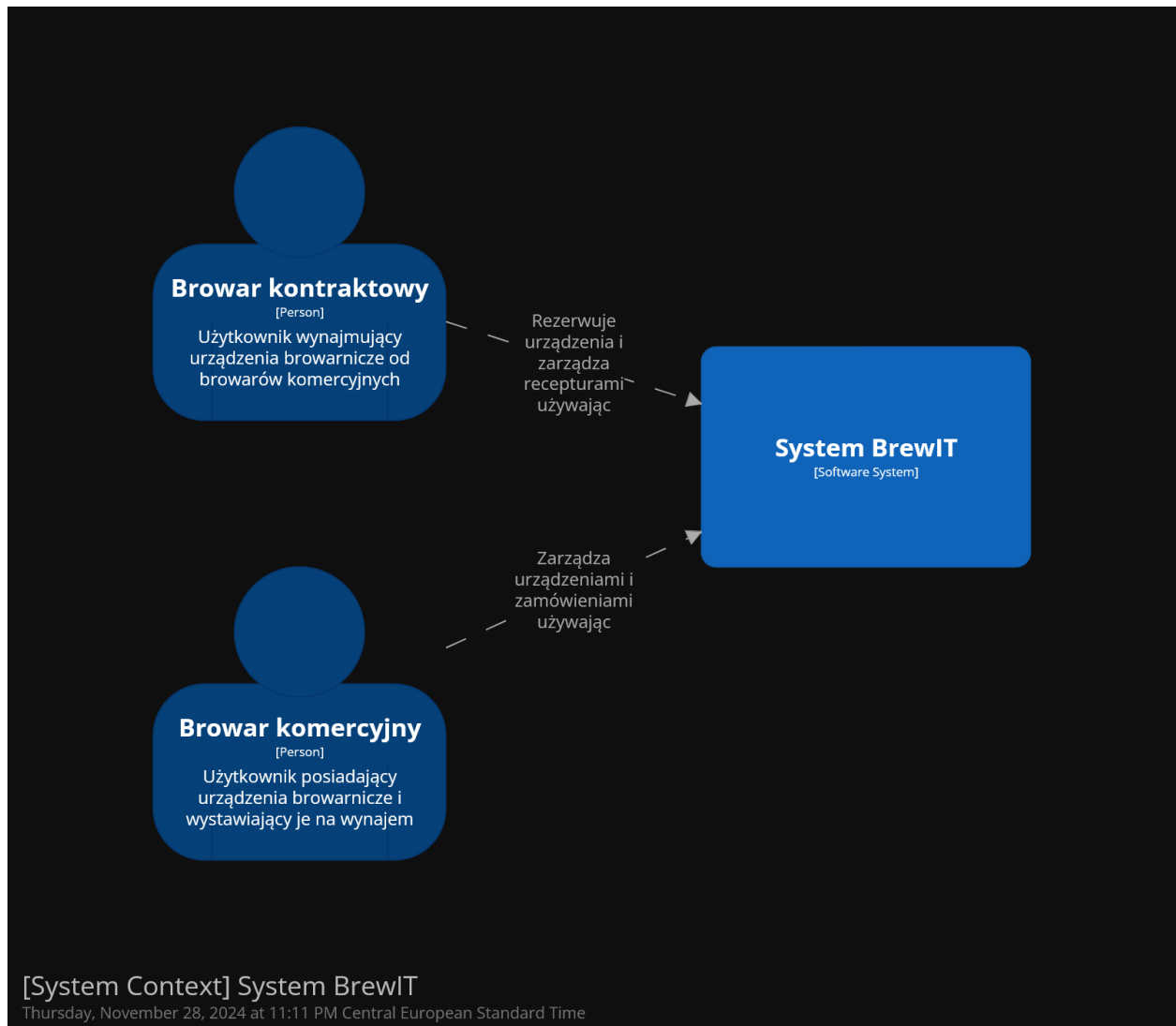


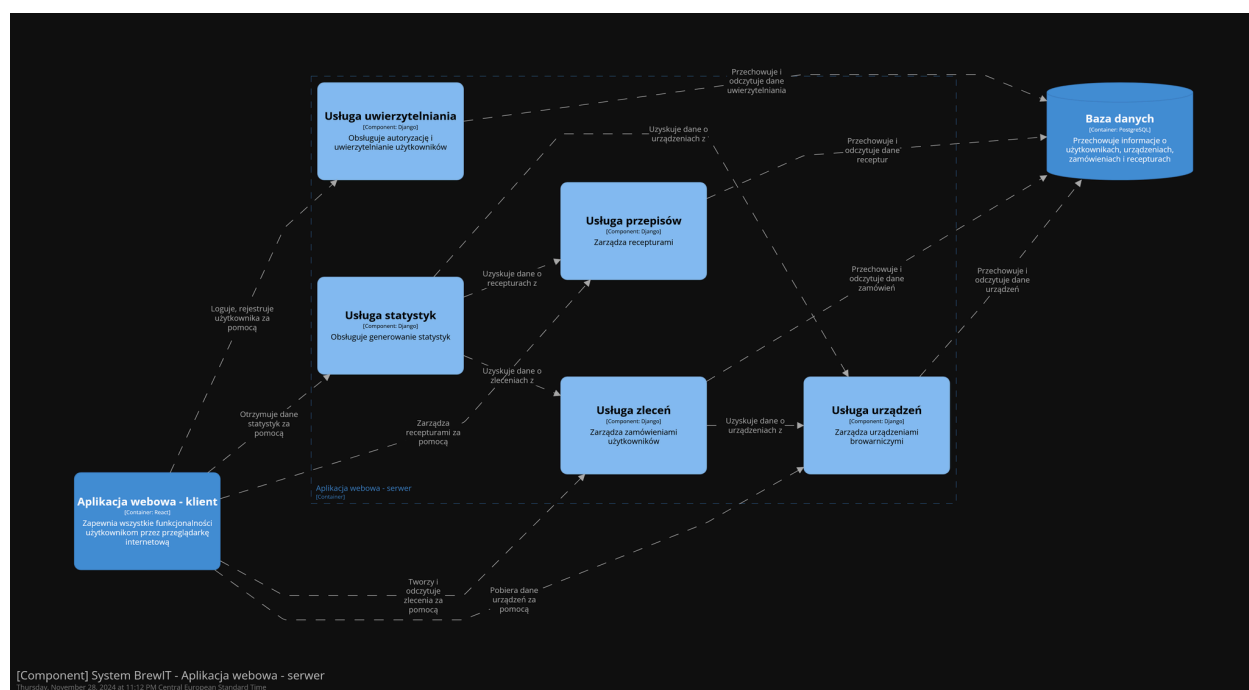
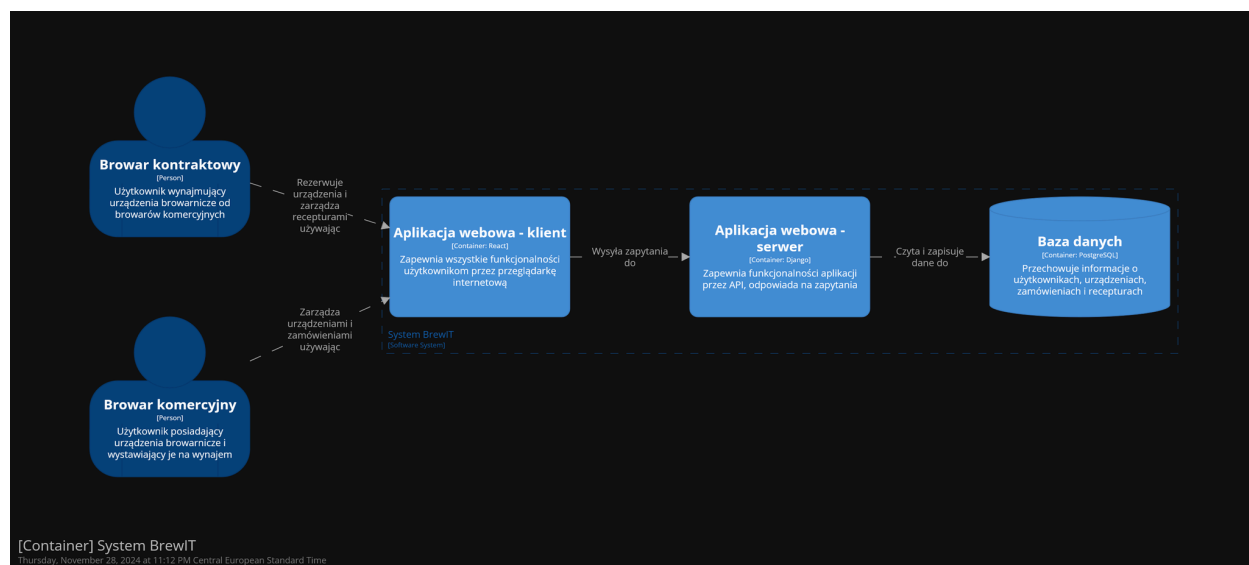
Nowak Robert 17.11 07:52

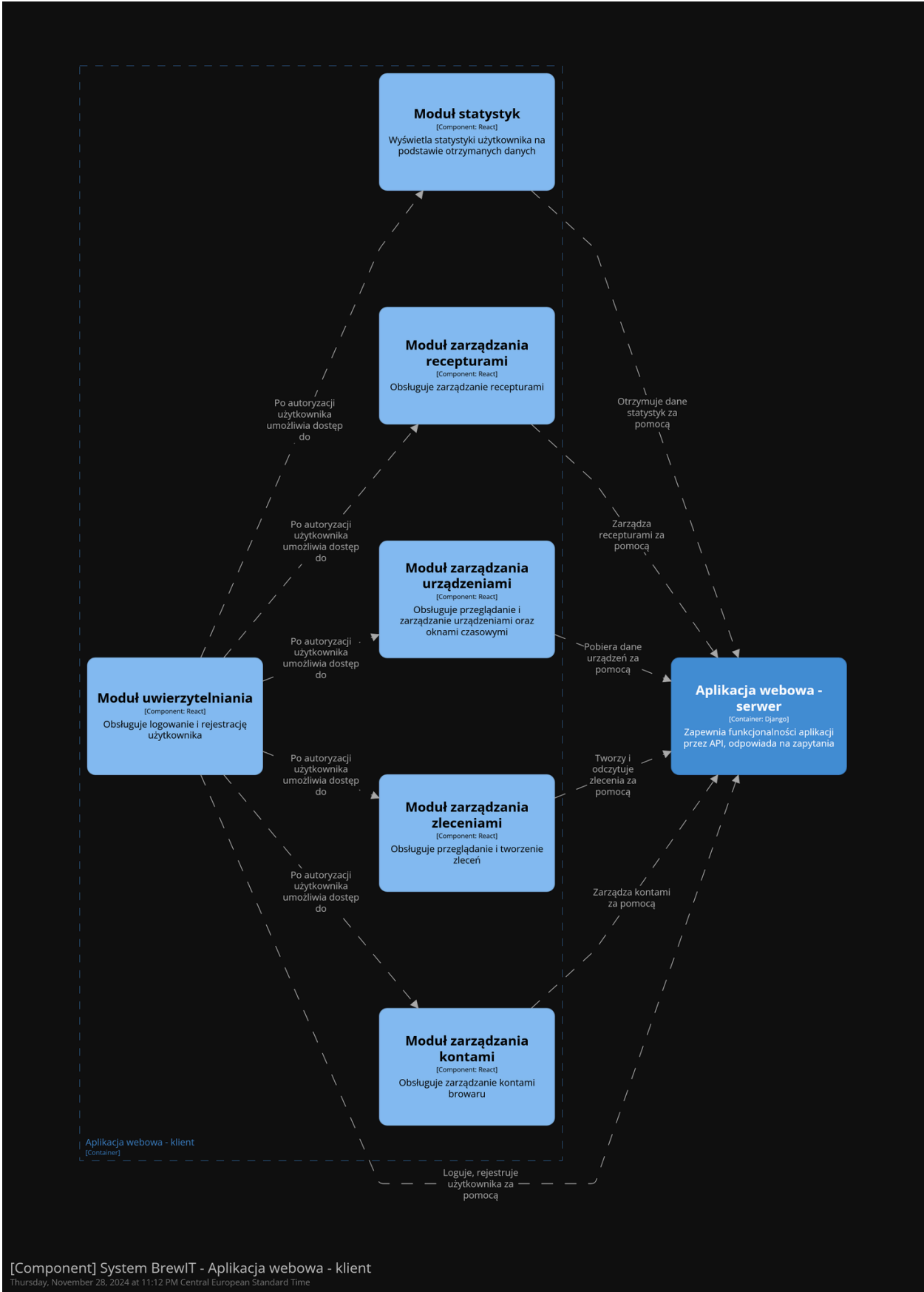
Przejrzałem wymagania i je akceptuję

4 Definicja architektury

Poniżej przedstawiono architekturę systemu w postaci diagramu C4:







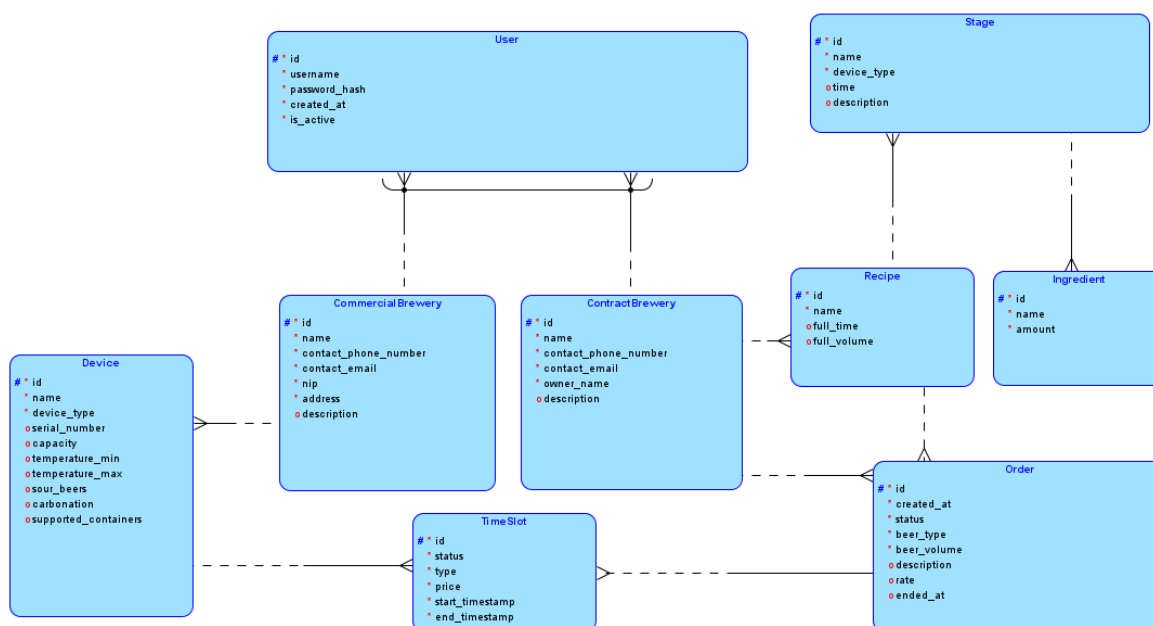
W systemie zastosowano architekturę MVVM (Model - View - ViewModel). Modele w Django odpowiadają strukturom przechowywanym w bazie danych PostgreSQL, ViewModel jest warstwą komunikacyjną pomiędzy frontendem i backendem zapewnioną przez Django REST Framework, widoki (View) realizowane są przez React.

Serwer wraz z bazą danych będą skonteneryzowane przez Docker Compose.

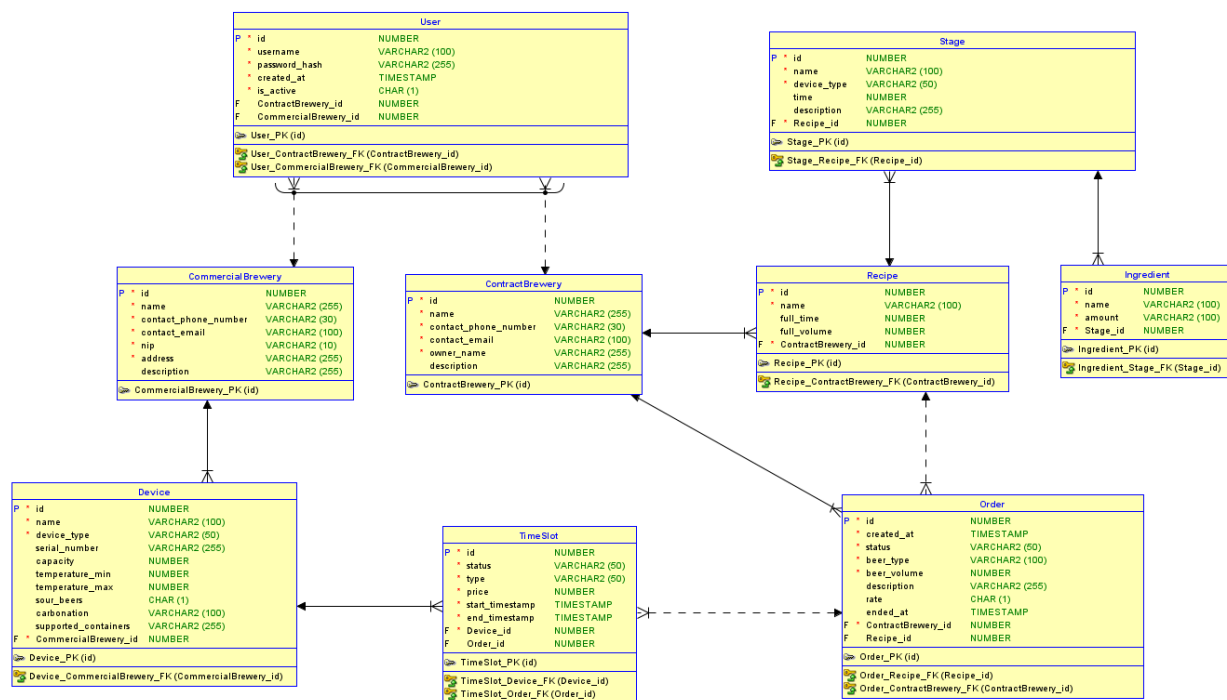
5 Dane trwałe

5.1 Model logiczny danych

Diagram ER:



Model relacyjny:



5.2 Przetwarzanie i przechowywanie danych

Planowane jest zaimplementowanie wyżej zdefiniowanej bazy danych przy użyciu systemu PostgreSQL.

Do komunikacji z bazą z poziomu *backendu* używany będzie wbudowany w Django system komunikacji.

6 Specyfikacja analityczna i projektowa

Repozytorium kodu dla projektu znajduje się pod adresem:

<https://github.com/amurawsk/brewit>

W pliku docs/readme.md opisano cel projektu wraz z dołączonymi zaakceptowanymi wymaganiami.

W projekcie użyte będą następujące technologie:

- backend: Python (Django)
- frontend: HTML + CSS + JS/TS (React)
- baza danych: PostgreSQL
- konteneryzacja: Docker Compose
- skrypty: bash
- automatyczne budowanie i testowanie konfiguracji: GitHub Actions

Statystyki (na dzień 10.01.2025):

Liczba plików: 212

Liczba linii kodu: 15248

Poniżej zaprezentowano diagram klas modeli w Django:

7 Projekt standardu interfejsu użytkownika

Stworzono projekt przedstawiający planowany wygląd interfejsu użytkownika do tworzonego systemu. Do tego celu użyto aplikacji Figma, która pozwala na prototypowanie interfejsów graficznych.

Stworzono tam schematy ekranów, które są przedstawiane użytkownikowi zanim się zaloguje i będzie korzystał z funkcjonalności aplikacji - stronę główną, logowania, rejestracji, a także tworzenia browaru (zarówno kontraktowego, jak i komercyjnego).

Dodatkowo stworzono także koncepcje stron, które umożliwiają główne funkcjonalności w projekcie - panel główny (dla browaru kontraktowego i komercyjnego), stronę dodawania urządzenia, dodawania okna czasowego, a także stronę umożliwiającą przeglądanie zleceń.

W schematach podanych stron zawarto najważniejsze przyciski, czy ułożenie odpowiednich elementów na stronie - pominięto wyświetlanie konkretnych danych, gdyż te, zgodnie z założeniami, będą inne dla każdego użytkownika.

Wszystkie wyżej wymienione strony dostępne są do przeglądania w formie wygodnej prezentacji dostępnej pod linkiem:

<https://www.figma.com/proto/0nEHfkFGy8BXMJQQeUB4Px/pzsp2?node-id=1-54&t=AWoV4OWOFHZAeKif-1>

Zalecane jest, aby po wejściu w prezentację użyć opcji *fit width* bądź *fit width and height*, aby wyświetlić całą szerokość ekranu. Dodatkowo można użyć opcji *hide ui*, gdyż często przy chęci wciśnięcia przycisków rejestracji lub logowania, niepotrzebnie pokazuje się interfejs aplikacji Figma.

Aby sprawnie poruszać się po stronach, dodano funkcje do niektórych przycisków. Aby wyświetlić dostępne do kliknięcia przyciski, wystarczy kliknąć w dowolnym miejscu ekranu - wtedy podświetlone zostaną interaktywne elementy prezentacji, które pozwalają przechodzić między ekranami.

Zdefiniowaliśmy aktorów w systemie brewIT:

1. Janusz Pijanka (browar kontraktowy)

- Opis roli:

Janusz reprezentuje jednoosobowy browar kontraktowy, który nie posiada własnego sprzętu do warzenia piwa. Chce wyprodukować swoją najnowszą recepturę, ale nie ma do tego warunków. Dotychczasowe metody współpracy wymagały wielokrotnego kontaktowania się z browarami komercyjnymi, co było dla niego uciążliwe i czasochłonne.

- Potrzeby i frustracje:

- Szybki dostęp do informacji o dostępności sprzętu.
- Możliwość składania zleceń i zarządzania nimi oraz swoimi recepturami w jednym miejscu.
- Zmniejszenie liczby osobistych kontaktów z browarami komercyjnymi.

2. Kierownik browaru komercyjnego

- Opis roli:
Kierownik zarządza browarem komercyjnym, który posiada sprzęt do warzenia piwa, ale nie jest on w pełni wykorzystywany. Chce wynajmować sprzęt browarom kontraktowym w sposób efektywny i uporządkowany.
- Potrzeby i frustracje:
 - Minimalizacja liczby pytań o dostępność sprzętu.
 - Automatyzacja komunikacji z klientami na każdym etapie umawiania wynajmu.
 - Zarządzanie dostępnością sprzętu i harmonogramami w przejrzystym sposób.

3. Kierownik spółki

- Opis roli:
Kierownik zarządza spółką koordynującą współpracę browarów kontraktowych i komercyjnych. Jego celem jest usprawnienie procesów, takich jak umawianie dostępności sprzętu, przyjmowanie zleceń i komunikacja między browarami.
- Potrzeby i frustracje:
 - Kierownik oczekuje intuicyjnego systemu, który zmniejszy liczbę błędów organizacyjnych i ograniczy czasochłonność manualnych procesów zarządzania.
 - Chce móc ją nadzorować przez przeglądanie szczegółów kont użytkowników i statystyk.

Stworzono także przykładowe historyjki użytkownika, które przedstawiają podstawowe czynności, które wykonuje użytkownik w systemie. Zaprezentowane są one poniżej, wraz z najważniejszymi ekranami, przez które przechodzi użytkownik.

1) Browar komercyjny rejestruje się i przechodzi do panelu głównego

The screenshot shows the 'Zarejestruj się' (Sign up) form on the brewiT website. The header includes the brewiT logo and navigation links: 'O nas', 'Rozwiązania', 'Klienci', 'Wyniki', 'Informacje', 'Kontakt', 'Zaloguj', and 'Zarejestruj'. The form itself is titled 'Zarejestruj się' and contains the following fields and elements:

- Nazwa użytkownika**: A text input field with the placeholder 'Podaj nazwę'.
- Hasło**: A text input field with the placeholder 'Podaj hasło'.
- ☒ **Akceptuję regulamin serwisu**: A checkbox that is checked, with the text 'Wymagane' below it.
- Zarejestruj**: A dark button to submit the form.

Wybierz typ browaru

[Mam browar kontraktowy](#)
[Mam browar komercyjny](#)

Podaj podstawowe dane

Nazwa browaru

Email kontaktowy

Telefon kontaktowy

Opis (opcjonalne)

Podaj pozostałe dane

NIP

Adres

Zatwierdzając, podświadczasz, że podane dane są prawdziwe

[Anuluj](#)
[Zatwierdź](#)


- Panel boczny
- Wybierz zakładkę**
- Mój browar**
Przejdź do widoku browaru
 - Urządzenia**
Przejdź do widoku urządzeń
 - Okna czasowe**
Przejdź do widoku okien czasowych
 - Zlecenia**
Przejdź do widoku zleceń
 - Współpracownicy**
Przejdź do widoku współpracowników
 - Statystyki**
Wyświetl statystyki

Nowe zlecenia
 <dane>

Aktualne zlecenia
 <dane>

Moje urządzenia
 <dane>

2) Browar kontraktowy rejestruje się i przechodzi do panelu głównego



[O nas](#) [Rozwiązania](#) [Klienci](#) [Wyniki](#) [Informacje](#) [Kontakt](#) [Zaloguj](#) [Zarejestruj](#)

Zarejestruj się

Nazwa użytkownika

Podaj nazwę

Hasło

Podaj hasło

☒ Akceptuję regulamin serwisu
Wymagane

Zarejestruj



[O nas](#) [Rozwiązania](#) [Klienci](#) [Wyniki](#) [Informacje](#) [Kontakt](#) [Zaloguj](#) [Zarejestruj](#)

Wybierz typ browaru

Mam browar kontraktowy

Mam browar komercyjny

Podaj podstawowe dane

Nazwa browaru

Email kontaktowy

Telefon kontaktowy

Opis (opcjonalne)

Podaj pozostałe dane

Imię i nazwisko właściciela

Zatwierdzając, poświadczasz, że podane dane są prawdziwe

[Anuluj](#)
[Zatwierdź](#)

- Panel boczny
- Wybierz zakładkę**
- Mój browar**
Przejdź do widoku browaru
 - Moje zlecenia**
Przejdź do widoku zleceń
 - Przeglądaj okna czasowe browarów komercyjnych**
Wybierz odpowiednie okna czasowe i stwórz nowe zlecenie
 - Receptury**
Przejdź do widoku receptur
 - Współpracownicy**
Przejdź do widoku współpracowników
 - Statystyki**
Wyświetl statystyki

Aktualne zlecenia
<dane>

Stworzone zlecenia
<dane>

Zlecenia do oceny
<dane>

Receptury
<dane>

3) Browar komercyjny przechodząc z panelu głównego dodaje urządzenie, a następnie dodaje okno czasowe na to urządzenie

- Panel boczny
- Wybierz zakładkę**
- Mój browar**
Przejdź do widoku browaru
 - Urządzenia**
Przejdź do widoku urządzeń
 - Okna czasowe**
Przejdź do widoku okien czasowych
 - Zlecenia**
Przejdź do widoku zleceń
 - Współpracownicy**
Przejdź do widoku współpracowników
 - Statystyki**
Wyświetl statystyki

Nowe zlecenia
<dane>

Aktualne zlecenia
<dane>

Moje urządzenia
<dane>

Panel boczny

Wybierz zakładkę

Mój browar
Przejdź do widoku browaru

Urządzenia
Przejdź do widoku urządzeń

Okna czasowe
Przejdź do widoku okien czasowych

Zlecenia
Przejdź do widoku zleceń

Współpracownicy
Przejdź do widoku współpracowników

Statystyki
Wyświetl statystyki

Urządzenia → Dodaj urządzenie

Dodaj urządzenie

Wypełnij pola, aby dodać urządzenie

Wybierz typ urządzenia

Pojemnik fermentacyjny

Podaj nazwę urządzenia

Podaj nazwę

Podaj numer seryjny (opcjonalnie)

Nr seryjny

Podaj zakres temperatur (w stopniach Celsjusza)

Minimalna temperatura

Maksymalna temperatura

☒ Obsługa piw kwaśnych

Zaznacz, jeśli urządzenie jest przystosowane do piw kwaśnych

Anuluj

Dodaj urządzenie

Panel boczny

Wybierz zakładkę

Mój browar
Przejdź do widoku browaru

Urządzenia
Przejdź do widoku urządzeń

Okna czasowe
Przejdź do widoku okien czasowych

Zlecenia
Przejdź do widoku zleceń

Współpracownicy
Przejdź do widoku współpracowników

Statystyki
Wyświetl statystyki

Okna czasowe → Dodaj okno

Dodaj okno czasowe

Wypełnij pola, aby dodać okno

Wybierz urządzenie

Pojemnik fermentacyjny - nr seryjny 0123456789

Podaj cenę (za dzień)

Wpisz cenę (w zł)

Wybierz daty dostępności

5 - 17 Sierpnia

Sierpień 2023

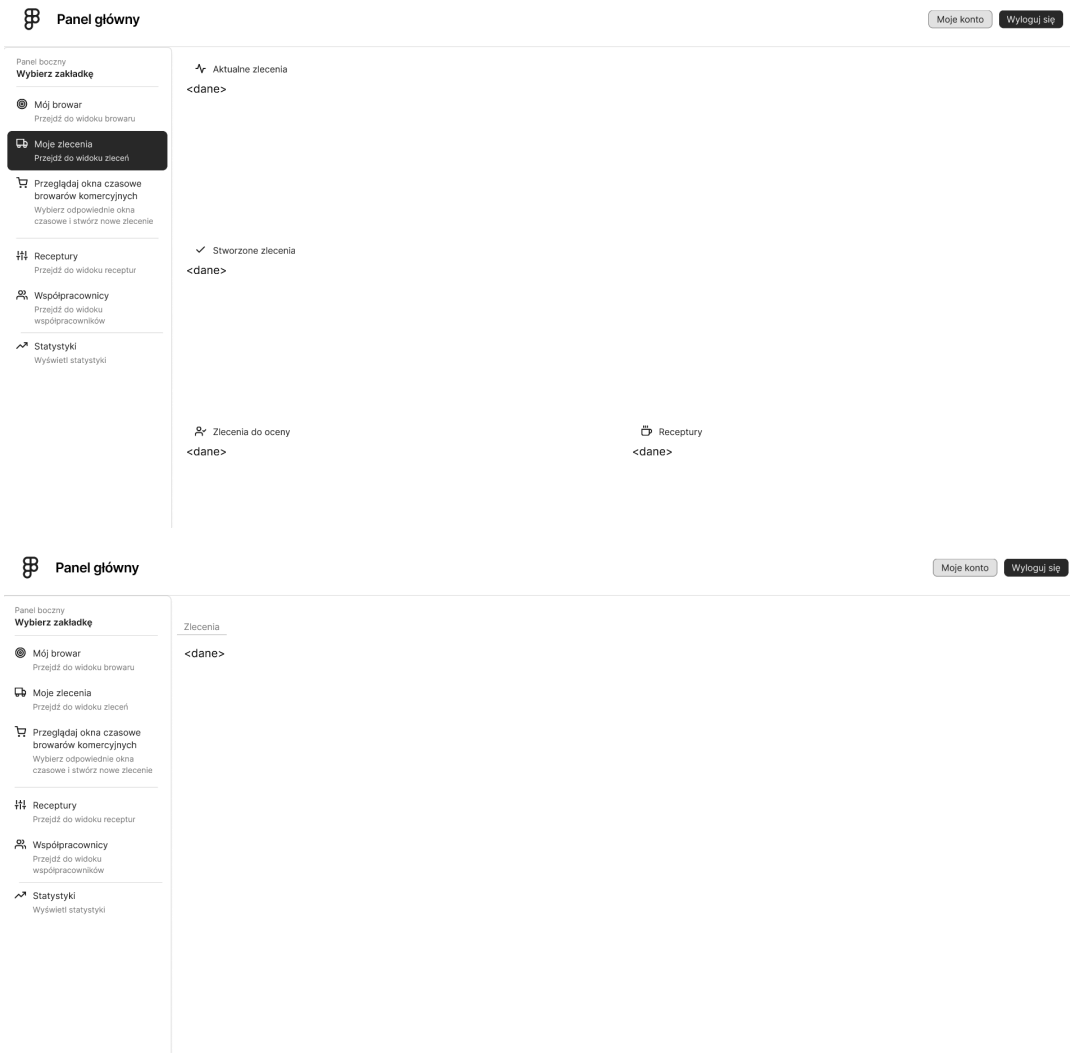
N	M	T	W	T	F	S
		1	2	3	4	5
6	7	8	9	10	11	12
13	14	15	16	17	18	19
20	21	22	23	24	25	26
27	28	29	30	31		

Cancel OK

Anuluj

Dodaj okno

4) Browar kontraktowy przechodząc z panelu głównego przegląda swoje zlecenia



Szkielet aplikacji (prototyp) oraz nawigacja między ekranami zostały wstępnie zaimplementowane w JavaScript i React, na aktualnym etapie bez stylizacji. Frontend będzie pobierał dane z bazy danych poprzez widoki serwera Django za pomocą zapytań HTTPS.

Od poprzedniej aktualizacji stworzono i odpowiednio wystylizowano wszystkie ekrany dla browaru komercyjnego, zdecydowaną większość ekranów browaru kontraktowego. Do uzupełnienia pozostają jeszcze ekrany spółki pośredniczącej.

8 Specyfikacja testów

Standardy obsługi błędów i sytuacji wyjątkowych

Głównym miejscem wystąpienia potencjalnego błędu w tworzonej aplikacji jest wywołanie zapytań do backendu. Przy wykonywaniu takich zapytań stosujemy bloki try-catch. W przypadku niepowodzenia

użytkownikowi wyświetlany jest komunikat na ekranie informujący o błędzie. Przykładem takiej sytuacji może być podanie niepoprawnych danych logowania, co jest opatrzone stosownym komunikatem dla użytkownika.

Scenariusze testowe

Te testy będą przeprowadzane manualnie.

1. Użytkownik nie może się zalogować bez założenia konta.
2. Użytkownik rejestruje się, loguje i wylogowuje.
3. Użytkownik wyświetla szczegóły swojego konta.
4. Użytkownik wyświetla szczegóły browaru i je edytuje.
5. Browar komercyjny dodaje urządzenie i okno czasowe.
6. Browar komercyjny wyświetla i usuwa urządzenia.
7. Browar komercyjny wyświetla, modyfikuje i usuwa okno czasowe.
8. Browar kontraktowy składa nowe zlecenie.
9. Browar komercyjny przegląda i zmienia status zleceń.
10. Browar kontraktowy ocenia zlecenie.
11. Użytkownik dodaje, przegląda i usuwa współpracowników.
12. Browar kontraktowy dodaje nową recepturę.
13. Użytkownik przegląda receptury, wyświetla ich szczegóły i edytuje.
14. Użytkownik wyświetla różne rodzaje statystyk.

Rodzaje testów

Poza wyżej wspomnianymi testami manualnymi, przygotowane są testy automatyczne, zarówno do ekranów aplikacji (frontendu), jak i do serwera (backendu). Obecnie zaimplementowane są testy jednostkowe, sprawdzające poprawność logiki za wykonywanymi operacjami, zarówno na serwerze, jak i w komponentach w aplikacji React. Poza nimi, planowane są testy integracyjne (sprawdzające odpowiednie przechodzenie między kolejnymi ekranami aplikacji oraz zweryfikowanie, np. czy wprowadzane dane są poprawnie przechowywane i przesyłane po kliknięciu przycisku “zatwierdź”) oraz testy end-to-end (które sprawdzą przechodzenie od początku do końca konkretnego scenariusza testowego).

Miary jakości testów

Do głównej miary jakości testów możemy zaliczyć pokrycie kodu, które będzie wyliczone dla części serwerowej (backendu) oraz aplikacyjnej (frontendu).

Bieżące testowanie aplikacji

W celu weryfikowania poprawności działania aplikacji, zestawione jest CI/CD, które weryfikuje poprawne zbudowanie aplikacji, jak i spełnienie wymagań testów jednostkowych. W ramach testowania mierzone jest także pokrycie kodu, które docelowo będzie przesyłane na platformę Codecov. W *wersji z dnia 10.01.2025 przesyłanie pokrycia kodu nie jest zapewnione, a żeby je sprawdzić, należy uruchomić polecenia testujące lokalnie*. Pipelines są uruchamiane za każdym razem, gdy na gałęzi *main* pojawia się zmiana lub przygotowany zostaje do niej *Pull Request*. Dzięki temu, można mieć pewność, że kod znajdujący się na głównej gałęzi jest weryfikowany pod kątem poprawności.

9 Wirtualizacja/konteneryzacja

W projekcie zastosowano konteneryzację przy użyciu Docker Compose, co umożliwia szybkie wdrażanie i mobilność projektu. Każda usługa w ramach aplikacji: backend (serwer z Django REST API), baza danych i frontend (klient w React), działa w osobnym kontenerze. Frontend hostuje interfejs użytkownika na porcie 3000, backend obsługuje logikę aplikacji na porcie 8000, a baza danych PostgreSQL komunikuje się z innymi kontenerami na porcie 5432 (z zewnątrz baza nie komunikuje się używając tego portu) oraz przechowuje dane wykorzystując wolumen (volume) Dockera. Przy pierwszym uruchomieniu lub wykryciu braku wymaganej bazy danych następuje jej automatyczna inicjalizacja, która poza utworzeniem bazy i użytkownika modyfikuje ustawienia skryptem `init.sql`. Zmienne środowiskowe są przechowywane w pliku `.env` dla bezpieczeństwa i łatwego zarządzania. Plik ten nie jest przechowywany w repozytorium. W repozytorium znajduje się plik `example.env`, który pokazuje jakie pola są wymagane, lecz który należy zedytować ze względów bezpieczeństwa. Frontend uruchamiany jest po uruchomieniu backendu (instrukcja `depends_on`), a uruchomienie backendu uzależnione jest od stanu bazy danych. Sprawdzenie stanu określone jest komendą `healthcheck`, backend do uruchomienia wymaga statusu `"service_healthy"`.

10 Bezpieczeństwo

Nasza aplikacja została zaprojektowana tak, aby spełniać podstawowe wymagania bezpieczeństwa. W aplikacji zadaliśmy o następujące rzeczy:

1. Hasła są szyfrowane w bazie danych, a aplikacja korzysta w tym celu z wbudowanego mechanizmu Django wykorzystującego algorytm z solą, dzięki czemu nawet przy wycieku bazy danych hasła są trudne do złamania.
2. Chronimy się przed atakami SQL injection poprzez zastosowanie ORM Django, który dba o odpowiednie zabezpieczenia. Ponadto korzystamy z odpowiednich serializerów, na poziomie serwera dodatkowo weryfikują i walidują dane wejściowe.
3. Do zarządzania sesjami wykorzystywane są JWT (JSON Web Tokens). Służą one do bezpiecznego potwierdzania tożsamości zalogowanych użytkowników. Tokeny mają odpowiedni czas życia, aby zmniejszyć ryzyko ich nieautoryzowanego użycia, a także obsługiwany jest mechanizm odświeżania tokenów, aby zapewnić długoterminową sesję.
4. Dostęp do bazy danych jest możliwy jedynie z poziomu kontenera w sieci Dockera, co minimalizuje ryzyko nieautoryzowanego dostępu.
5. Wrażliwe dane, sekrety oraz dane dostępu są przechowywane w pliku `.env`, który jest ignorowany przez system kontroli wersji. Dzięki temu dane te nie są nigdzie publikowane.

11 Podręcznik użytkownika

[instrukcja użycia funkcjonalności systemu]

System wynajmu urządzeń browarniczych umożliwia browarom kontraktowe rezerwowanie i wynajmowanie sprzętu od browarów komercyjnych. System zapewnia łatwą interakcję między browarami, automatyzując proces rezerwacji urządzeń oraz umożliwiając ocenę wykonanych warek. Poniższy podręcznik przedstawia szczegółowe instrukcje dotyczące korzystania z systemu.

- Rejestracja konta
 1. Wejdź na stronę główną systemu.

2. Wybierz opcję "Zarejestruj" w prawym górnym rogu.
 3. Wypełnij formularz podstawowy (adres e-mail i hasło).
 4. Wybierz opcję z odpowiednim browarem (kontraktowy/ komercyjny).
 5. Wypełnij formularz na pozostałe dane (nazwa browaru, dane kontaktowe, adres i nip lub godność właściciela).
 6. Po wypełnieniu formularza kliknij "Zarejestruj".
 7. Klikając "Anuluj" cofniesz się do ekranu wyboru typu browaru.
- Logowanie
 1. Na stronie głównej wybierz opcję "Zaloguj się".
 2. Wprowadź swoje dane logowania (adres e-mail i hasło).
 3. Kliknij "Zaloguj".
 - Wylogowanie
 1. Będąc zalogowanym, wybierz opcję "Wyloguj" w prawym górnym rogu ekranu.
 - Wyświetlanie szczegółów konta
 1. Wybierz opcję "Moje konto" w prawym górnym rogu ekranu.
 2. Wyświetl informacje o koncie (nazwa, data założenia, przypisany browar).
 - Przeglądanie szczegółów browaru
 1. Aby wyświetlić szczegóły browaru, przejdź do zakładki Mój browar.
 2. Wyświetl informacje o browarze (nazwa browaru, dane kontaktowe, adres i nip lub godność właściciela), liczbę urządzeń/ przepisów, zleceń i współpracowników.
 - Przeglądanie urządzeń i okien czasowych

Browar komercyjny może przeglądać, dodawać, edytować i usuwać urządzenia:

1. Aby wyświetlić swoje urządzenia, przejdź do zakładki Urządzenia.
2. Wyświetl listę urządzeń.
3. Usuń urządzenie wybierając opcję "Usuń"
4. Wyświetl szczegóły lub edytuj urządzenie, klikając na nie.
5. Dodaj urządzenie, wybierając opcję "Dodaj urządzenie" i wprowadź szczegóły dotyczące urządzenia (nazwa, typ, pojemność, czy nadaje się do produkcji "kwaśnego" piwa, czy i jakie nagazowanie wspiera, jakie pojemniki obsługuje).

Browar komercyjny może dodawać, usuwać i przeglądać okna czasowe:

1. Aby dodać okno czasowe, przejdź do zakładki Okna Czasowe.
2. Wyświetl listę okien czasowych z przypisanymi urządzeniami.
3. Wybierz Dodaj okno czasowe i wprowadź szczegóły dotyczące okna czasowego (typ urządzenia, cena wynajmu, dni/ godziny).
4. Po kliknięciu w okno czasowe pojawia się możliwość edycji jego ceny lub usunięcia.

- Przeglądanie, tworzenie, zmiana statusu i usuwanie zlecenia
 1. Aby wyświetlić swoje zlecenia, przejdź do zakładki Zlecenia.
 2. Wyświetl listę zleceń.
 3. Filtruj zlecenia przyciskami "oczekujące", "aktualne", "przeszłe", "odrzucone".
 4. Wyświetl szczegóły zlecenia, zmień jego status lub usuń, klikając na nie i wybierając odpowiednią opcję.

- Przeglądanie, dodawanie i usuwanie współpracowników
 1. Aby wyświetlić swoich współpracowników, przejdź do zakładki Współpracownicy.
 2. Dodaj współpracownika, wybierając opcję “Dodaj współpracownika” i jego dane logowania (adres e-mail i hasło).
 3. Usuń współpracownika wybierając opcję “Usuń”
- Przeglądanie statystyk
 1. Przejdź do zakładki Statystyki.
 2. Wybierz typ statystyk (typy produkowany piw, ilość wyprodukowanego piwa, udane/ nieudane warki).
 3. Wyświetl wykresy kołowe i histogramy z odpowiednimi danymi.
 4. Doprecyzuj typ statystyk (liczba warek/ objętość lub dziennie/ miesięcznie/ rocznie).
- Przeglądanie, dodawanie, edytowanie i usuwanie przepisów
 1. Aby wyświetlić swoje przepisy, przejdź do zakładki Przepisy.
 2. Wyświetl listę przepisów.
 3. Usuń przepis wybierając opcję “Usuń”
 4. Wyświetl szczegóły lub edytuj przepis, klikając na nie.
 5. Dodaj przepis, wybierając opcję “Dodaj przepis” i wprowadź szczegóły dotyczące przepisu (nazwa, całkowita objętość, całkowity czas, etapy, składniki).

12 Podręcznik administratora

Instrukcja konfiguracji, budowy i instalacji systemu:

Wymagania wstępne:

1. Zainstalowany Docker oraz Docker Compose

Budowa systemu:

1. Pobierz repozytorium wykonując polecenie:

```
git clone https://github.com/amurawsk/brewit
```

2. Stwórz plik `.env` zgodnie z przykładowym plikiem `example.env`
3. Zbuduj aplikację wykonując polecenie:

```
docker compose up --build
```

Po wykonaniu powyższych kroków, aplikacja będzie dostępna pod adresem `localhost:3000`, a serwer Django pod adresem `localhost:8000`. Aplikacja może być również dostępna w sieci prywatnej lub publicznej po podaniu poprawnego adresu.

Testowanie aplikacji:

Aby przetestować aplikację backendową wraz z eksportem informacji o pokryciu kodu, należy użyć komendy:

```
docker compose exec backend pytest --cov=core
```

Aby przetestować aplikację frontendową należy użyć komendy:

```
docker compose exec frontend npm test
```

Instrukcja aktualizacji oprogramowania:

1. Zatrzymaj uruchomioną aplikację
2. W repozytorium wykonaj polecenie:

```
git pull origin main
```

3. Zbuduj ponownie aplikację używając polecenia:

```
docker compose up --build
```

Instrukcja zarządzania użytkownikami, uprawnieniami i danymi:

Aby zarządzać użytkownikami i innymi informacjami w bazie danych, należy wejść do panelu */admin/* w serwerze Django. Po zalogowaniu można zarządzać danymi znajdującymi się w bazie danych.

Aby móc się zalogować, należy stworzyć konto superużytkownika; w tym celu należy wpisać komendę

```
docker compose exec backend python manage.py createsuperuser
```

a następnie wpisać dane do logowania.

Instrukcja tworzenia kopii zapasowych i odtwarzania systemu

Aby utworzyć kopię zapasową należy wykonać zrzut bazy danych za pomocą komendy:

```
docker compose exec db pg_dump -U <użytkownik> <nazwa_bazy> > backup.sql
```

Aby przywrócić bazę danych należy użyć komendy:

```
docker compose exec db psql -U <użytkownik> <nazwa_bazy> < backup.sql
```

Instrukcja zarządzania zasobami systemu

Założeniem jest, że cały projekt jest uruchomiony w Dockerze. Żeby sprawdzić wykorzystanie zasobów, należy wpisać komendę:

```
docker stats
```

Aby sprawdzić dzienniki, należy użyć komendy:

```
docker logs <nazwa_kontenera>
```

Aby wyczyścić nieużywane kontenery, obrazy i woluminy, należy użyć komendy:

```
docker system prune
```

13 Podsumowanie

Krytyczna analiza osiągniętych wyników:

Projekt jest intensywnie rozwijany, ale nie został jeszcze w pełni ukończony. Obecnie stworzono wszystkie widoki dla browaru komercyjnego, a także większość widoków dla browaru kontraktowego - brakuje widoków dla spółki pośredniczącej, ekranu głównego browaru kontraktowego oraz funkcji

umożliwiającej składanie zleceń przez browar komercyjny. Dodatkowo, wiele widoków nie działa jeszcze w pełni, głównie z powodu braku integracji z backendem. Te kwestie wymagają dalszych prac, aby system mógł osiągnąć pełną funkcjonalność.

Uważamy, że udało się stworzyć przystępny, intuicyjny, a zarazem estetyczny interfejs użytkownika. Zakończenie pozostałych prac nie powinno zająć więcej niż tydzień.

Projekt jest w fazie rozwoju. Nie zapewniliśmy jeszcze planowanego poziomu bezpieczeństwa. Wiele komponentów wymaga stworzenia testów jednostkowych.

Możliwe kierunki rozwoju:

W przyszłości, po dokładnym przemyśleniu, można wprowadzić ulepszenia zaproponowane przez właściciela projektu, które zostały odrzucone na etapie wstępnych ustaleń. Przykładem takich zmian mogą być bardziej rozbudowane statystyki, obejmujące szczegółowe analizy zysków i strat, lub wprowadzenie alternatywnego systemu ocen, czy też dodanie modułu płatności wewnątrz aplikacji.

Dodatkowo, warto rozważyć podział użytkowników browarów na różne poziomy uprawnień. Na przykład, jedni użytkownicy mogliby mieć możliwość usuwania kont innych współpracowników, podczas gdy inni nie posiadaliby takich uprawnień. Taki podział zwiększyłby elastyczność zarządzania systemem i bezpieczeństwo danych.

Aby zapewnić wyższy poziom bezpieczeństwa danych w systemie, warto rozważyć wprowadzenie mechanizmu szyfrowania receptur. Dzięki temu, nawet osoby zarządzające systemem, posiadające dostęp do infrastruktury aplikacji, nie miałyby bezpośredniego dostępu do samych receptur.

14 Bibliografia

Aplikacja Figma, której użyto do stworzenia prototypu interfejsu: <https://www.figma.com/>

Biblioteka React, z której korzystano w projekcie: <https://react.dev/>

Przy tworzeniu tej dokumentacji wykorzystano materiały z ćwiczeń i wykładów zamieszczone na kanale dedykowanym temu przedmiotowi na stronie <https://leon.pw.edu.pl/>

Wzór notacji diagramu UML dla przypadków użycia: <https://www.geeksforgeeks.org/use-case-diagram/>

Zatwierdzam dokumentację.	<div>.....</div> <div>.....</div> <div>Data i podpis Mentora</div>
---------------------------	--