

Wprowadzenie do zarządzania projektami deweloperskimi

laboratorium

Zadanie G



Oliwier Jędrzejczak
Bartłomiej Wołyńiec
Konrad Karkosz
Michał Sobiło
Mateusz Bizoń

Prowadzący mgr inż. Marcin Tracz

Spis treści

1. Temat oraz opis projektu	3
1.1. Temat projektu	3
1.2. Opis projektu	3
1.3. Grupa projektowa	3
1.4. Funkcjonalności	3
1.5. Technologie oraz narzędzia	4
2. Event Storming	5
2.1. Opis	5
2.2. Legenda	5
2.3. Tablica wykonana w aplikacji miro	5
2.4. Wszystkie funkcjonalności	7
3. Backlogi	12
3.1. Backlog frontend-u	12
3.2. Backlog backend-u	13
4. Estymaty	14
4.1. Estymata frontend	14
4.2. Estymata backend	15
5. Sprinty	16
5.1. Podział na sprinty	16
Sprint 1	16
Sprint 2	16
Sprint 3	16
Sprint 4	17
5.2. Zrzuty ekranu	17
6. Działanie aplikacji	20
6.1. Zrzuty ekranu	20
7. Instrukcja uruchomienia	25
7.1. Lokalne uruchamianie systemu	25
7.2. Zdalne uruchamianie systemu	25
8. Podsumowanie	26
8.1. Wnioski	26

1. Temat oraz opis projektu

1.1. Temat projektu

Modern Pantry

1.2. Opis projektu

Projekt “Modern Pantry” zakłada stworzenie aplikacji do zarządzania żywnością oraz napojami alkoholowymi przechowywanymi w prywatnych lub wspólnych magazynach, które to będą miały odzwierciedlenie w postaci wirtualnej. Zakładamy możliwość dodawania siebie nawzajem do spiżarni. Użytkownicy będą dostawać notyfikacje odnośnie przykładowych stanów krytycznych w magazynie np. brak danego artykułu. Zakładamy wykorzystanie kodów QR dla każdej spiżarni, żeby użytkownik miał łatwiejszy dostęp do domowej inwentaryzacji.

1.3. Grupa projektowa

- Oliwier Jędrzejczak - Leader, Tester, Devops
- Bartłomiej Wołyniec - Backend
- Konrad Karkosz - Frontend
- Michał Sobiło - Backend
- Mateusz Bizoń - Frontend

1.4. Funkcjonalności

- Rejestracja i Logowanie
- Rejestracja przez social media
- Rejestracja potwierdzona mailem
- Resetowanie hasła
- Zarządzanie profilem użytkownika
- Prywatny profil
- Przeglądanie strony głównej

- Tworzenie własnej spiżarni
- Generowanie kodu QR do spiżarni
- Dodawanie osób do spiżarni
- Modyfikacja zawartości spiżarni dodawanie/usuwanie
- Wyświetlanie zawartości spiżarni
- Statystyki
- Notyfikacje, po zalogowaniu się
- Panel podglądu stanu wszystkich spiżarni
- Dark mode i light mode

1.5. Technologie oraz narzędzia

- ASP.NET Core - wybrany przez część zespołu backend-ową, jako najlepsze narzędzie do pracy w C#
- C# - wybrany na podstawie średniego największego doświadczenia w grupie backend-owej
- HTML oraz CSS - wymagane są do budowania stron internetowych.
- JavaScript - osoba odpowiedzialna za Frontend, która będzie go używać posiada bardzo dużą wiedzę na temat tego języka. Ciężko znaleźć alternatywę w tej kwestii.
- TypeScript - obecnie jest już standardem przy programowaniu z JavaScript.
- React - wybrany framework z powodu dobrej znajomości członków zespołu, którzy będą go używali.
- Microsoft Visual Studio - wybrane ze względu na ergonomiczną pracę oraz przyzwyczajenie osób z backendu.
- Microsoft Visual Studio Code - środowisko wybrane przez osoby z frontendu ze względu na wygodę i ergonomię pracy
- Prettier - narzędzie odpowiedzialne za formatowanie naszego kodu aby wyglądał jak najlepiej.
- Eslint - jest odpowiedzialny o to aby zadbać o jak najlepszą jakość i standard kodu JS/TS/React.

2. Event Storming

2.1. Opis

Na wstępie nasza grupa stworzyła wspólnie legendę, i na tej podstawie stworzyliśmy funkcjonalności. Kiedy już mieliśmy gotowe wszystko, rozpoczęliśmy pracę nad możliwymi scenariuszami. Dodatkowo jako pomoc przy projektowaniu utworzyliśmy uproszczony schemat bazy danych.

2.2. Legenda

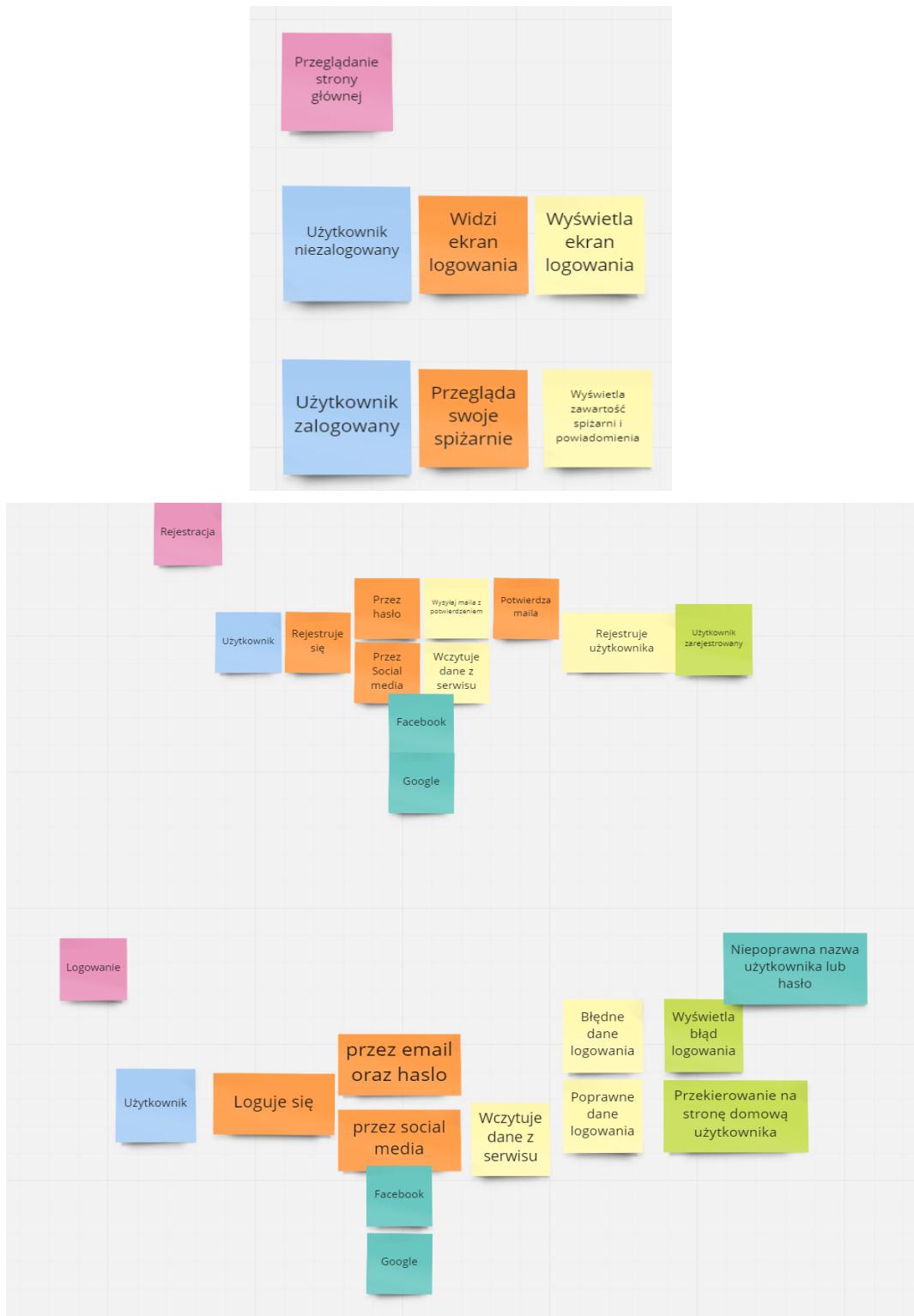


2.3. Tablica wykonana w aplikacji miro



2.4. Wszystkie funkcjonalności

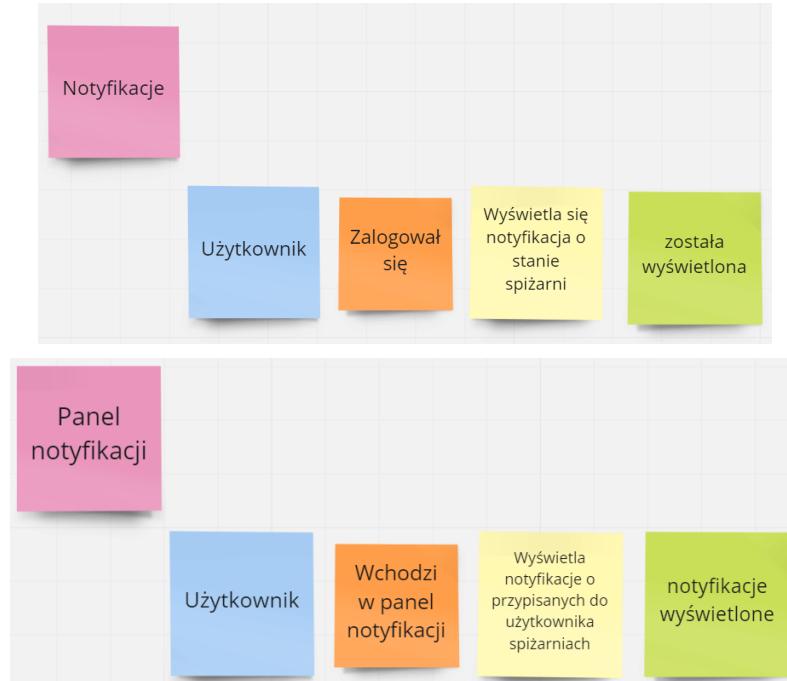












3. Backlogi

3.1. Backlog frontend-u

- Połaczenie z api
- Nagłówek strony
- Sidebar
- Strona 404
- Logowanie
- Rejestracja
- Dashboard
- Tworzenie spiżarni
- Statystyki i podsumowanie
- Notyfikacje
- Black/light mode

3.2. Backlog backend-u

- Inicjalizacja projektu, wraz z konfiguracją środowiska
- Stworzenie szkieletu aplikacji asp.net
- Utworzenie modelu bazy danych
- Konfigurowanie modelu testowego
- Napisanie wstępnych testów
- Implementacja systemu rejestracji i logowania
- Dołączenie dodatkowej funkcjonalności do rejestracji i logowania poprzez social media
- Utworzenie opcji resetowania hasła przez użytkownika w ustawieniach prywatności oraz poprzez resetowanie mailem
 - Możliwość tworzenia własnej spiżarni
 - a) Każda spiżarnia posiada własny kod qr
 - b) Do utworzonej spiżarni można dodawać/usuwać użytkowników
 - Implementacja systemu dodawania/usuwania produktów w spiżarnii
 - Wyświetlanie i edytowanie produktów
 - Stworzenie systemu notyfikacji, o braku notyfikacji
 - Utworzenie panelu statystyk dla danej spiżarni
 - zainportowanie strony na wcześniej przygotowanym serwerze
 - Konfiguracja serwera pod projekt aplikacji internetowej
 - Testowanie aplikacji w środowisku produkcyjnym
 - Publikacja aplikacji

4. Estymaty

4.1. Estymata frontend

Zadanie	Estymacja 1	Estymacja 2	Średnia
Inicjalizacja projektu z podstawową konfiguracją (Eslint, Prettier, VSC)	4	5	5
Kontekst autoryzacji	6	6	6
Połączenie z api	4	4	4
Stworzenie globalnych stylów oraz motywów	3	4	4
Nagłówek strony / header	3	4	4
Nawigacja	3	4	4
Pasek boczny	6	6	6
Ciemny motyw strony	4	5	5
Widok rejestracji	6	6	6
Widok statystyk profilu użytkownika	5	4	5
Widok logowania	4	4	4
Widok detali profilu użytkownika	5	3	4
Widok strony startowej	7	6	7
Widok spiżarni	8	7	8
Widok statystyk	8	8	8
Autoryzacja przez social media	6	5	6
Widok notyfikacji	5	5	5
Dodawanie produktu	4	3	4
Wyświetlanie kodu qr	4	4	4
Dodawanie nowej spiżarni	4	4	4
		Suma:	98h

4.2. Estymata backend

Zadanie	Estymacja 1	Estymacja 2	Estymacja 3	Średnia
Rejestracja nowego użytkownika	4	8	6	6
Potwierdzanie adresu email przy tworzeniu konta	4	3	4	4
Logowanie użytkownika	3	2	2	2
Pobieranie danych z portali społecznościowych: Facebook i Instagram	8	10	10	9
Resetowanie hasła przez "zapomnienie hasła"	4	6	4	5
Utworzenie bazy danych	4	4	4	4
Dodawanie osób do spiżarni	4	4	4	4
Wyświetlanie produktów	4	4	4	4
Tworzenie kategorii produktów	4	4	4	4
Tworzenie bazy danych	4	5	4	4
Dodawanie podstawowych Crud-ów	4	3	4	4
Usuwanie spiżarni	2	2	3	2
Wyświetlanie Spiżarni	2	3	3	3
Statystyki spiżarni	4	8	6	6
Resetowanie hasła do konta	3	4	4	4
Logowanie się za pomocą zewnętrznych kont	2	2	3	2
Testy aplikacji	18	22	30	23
			Suma:	90h

5. Sprinty

5.1. Podział na sprinty

Czas trwania poszczególnych sprintów: 2 tygodnie.

Sprint 1

- Utworzenie modelu bazy danych
- Utworzenie szkieletu aplikacji
- Zaprojektowanie logo strony
- Stworzenie strony głównej
- Stworzenie paska bocznego
- Stworzenie widoku rejestracji
- Stworzenie widoku logowania

Sprint 2

- Stworzenie dodawania i przeglądania spiżarni należących do użytkownika
- Rejestracja konta przez email i weryfikacja
- Dodanie Rejestracji konta przez Gmail/facebook
- Stworzenie CRUD-ów dla produktów
- Stworzenie CRUD-ów dla spiżarni
- Stworzenie panelu użytkownika
- Stworzenie strony nie znaleziono (404)

Sprint 3

- Generowanie kodów QR
- Stworzenie strony ze statystykami i podsumowaniem
- Stworzenie okna dodawania produktów
- Utworzenie Notyfikacji w backend-zie
- Zaproszenia do spiżarni
- Podłączanie backend-u z frontend-em
- Resetowanie hasła poprzez email
- Resetowanie hasła poprzez ustawienia profilu użytkownika

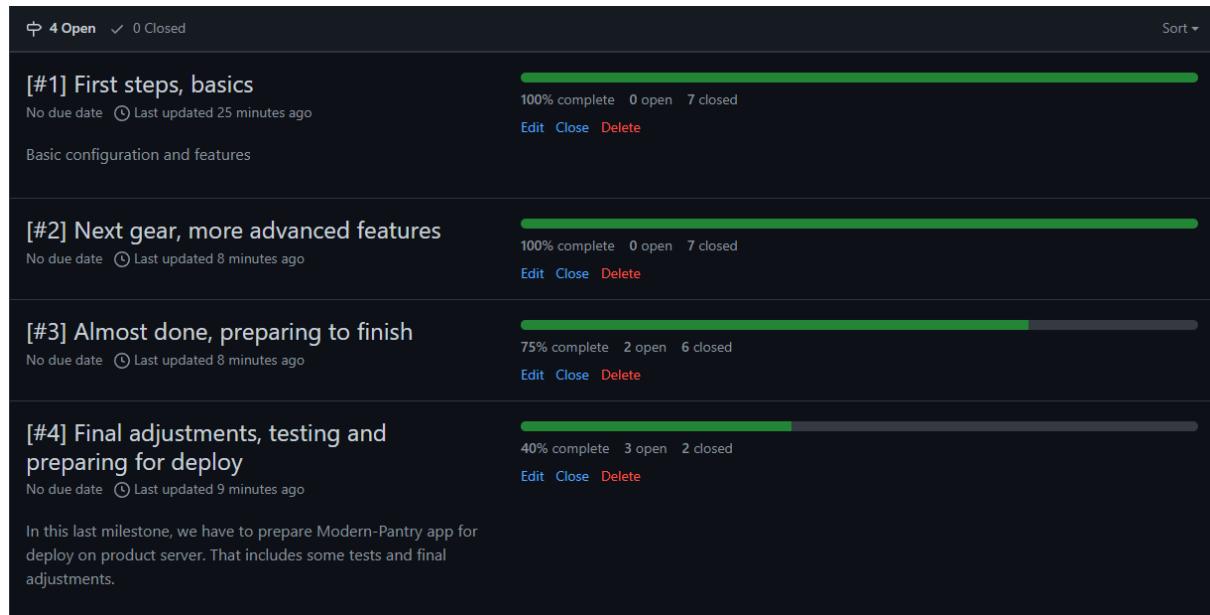
Sprint 4

- Utworzenie statystyk w backend-zie
- Tworzenie podpowiedzi do notyfikacji
- Testowanie frontend-u
- Testowanie backend-u
- Hostowanie aplikacji wraz z bazą danych

5.2. Zrzuty ekranu

Poszczególne zadania tylko dla frontend-u mają oznaczenie “[FE]”, jeżeli dla danego sprintu opóźniło się wykonanie zadania, zostało to umieszczone poniżej zrzutu zakończonych dla danego kamienia milowego.

Wszystkie Sprints



Sprint 1

[#1] First steps, basics

No due date 100% complete

Basic configuration and features

0 Open ✓ 7 Closed

☐ [FE] Create register view

#11 by jedrzejowsky was closed on Dec 3, 2022 2 tasks done

☐ [FE] Create login view

#12 by jedrzejowsky was closed on Dec 3, 2022 2 tasks done

☐ [FE] Design the site logo

#22 by Zayooo00 was closed on Nov 11, 2022

☐ [FE] Basic project configuration

#6 by jedrzejowsky was closed on Nov 11, 2022 2 tasks done

☐ [FE] Create Home page

#10 by jedrzejowsky was closed on Nov 11, 2022

☐ [FE] Create sidebar

#13 by jedrzejowsky was closed on Nov 11, 2022

☐ [FE] Create database

#7 by jedrzejowsky was closed on Nov 11, 2022 1 task done

↑↓ 2



Sprint 2

[#2] Next gear, more advanced features

No due date 100% complete

0 Open ✓ 7 Closed

☐ [FE] Creating and viewing user owned pantries

#42 by jedrzejowsky was closed 2 weeks ago



1

☐ Register account via Gmail/facebook

#9 by jedrzejowsky was closed 3 weeks ago 3 tasks done



1

☐ Register account via email and verify

#8 by jedrzejowsky was closed on Dec 5, 2022 2 tasks done



1

☐ Product CRUD

#25 by Btwol was closed on Nov 19, 2022

↑↓ 1



☐ [FE] Create not-found page

#24 by Zayooo00 was closed on Nov 11, 2022



☐ [FE] Create dashboard

#15 by jedrzejowsky was closed on Nov 11, 2022



☐ [FE] Pantry CRUD

#20 by Btwol was closed on Nov 11, 2022

↑↓ 1



Sprint 3

[#3] Almost done, preparing to finish

No due date 75% complete

2 Open ✓ 6 Closed

- Connecting backend with frontend #29 by jedrzejowsky was closed 2 hours ago
- [FE] Adding products #43 by jedrzejowsky was closed last week
- Notifications - Backend #40 by Btwol was closed 2 weeks ago
- Pantry Invites #37 by Btwol was closed 2 weeks ago
- [FE] Create statistics + summary #19 by jedrzejowsky was closed on Dec 3, 2022 2 tasks done
- Generate QR code #14 by jedrzejowsky was closed on Dec 1, 2022 3 tasks

[#3] Almost done, preparing to finish

No due date 75% complete

2 Open ✓ 6 Closed

- Reset password (forgotten password) #17 opened on Nov 2, 2022 by jedrzejowsky 2 tasks
- Reset password in profile settings (change password) #18 opened on Nov 2, 2022 by jedrzejowsky

Sprint 4

[#4] Final adjustments, testing and preparing for deploy

No due date 40% complete

In this last milestone, we have to prepare Modern-Pantry app for deploy on product server. That includes some tests and final adjustments.

3 Open ✓ 2 Closed

- [FE] Testing app #72 by jedrzejowsky was closed 2 hours ago 1 task done
- Statistics - Backend #41 by Btwol was closed last week

[#4] Final adjustments, testing and preparing for deploy

No due date 40% complete

In this last milestone, we have to prepare Modern-Pantry app for deploy on product server. That includes some tests and final adjustments.

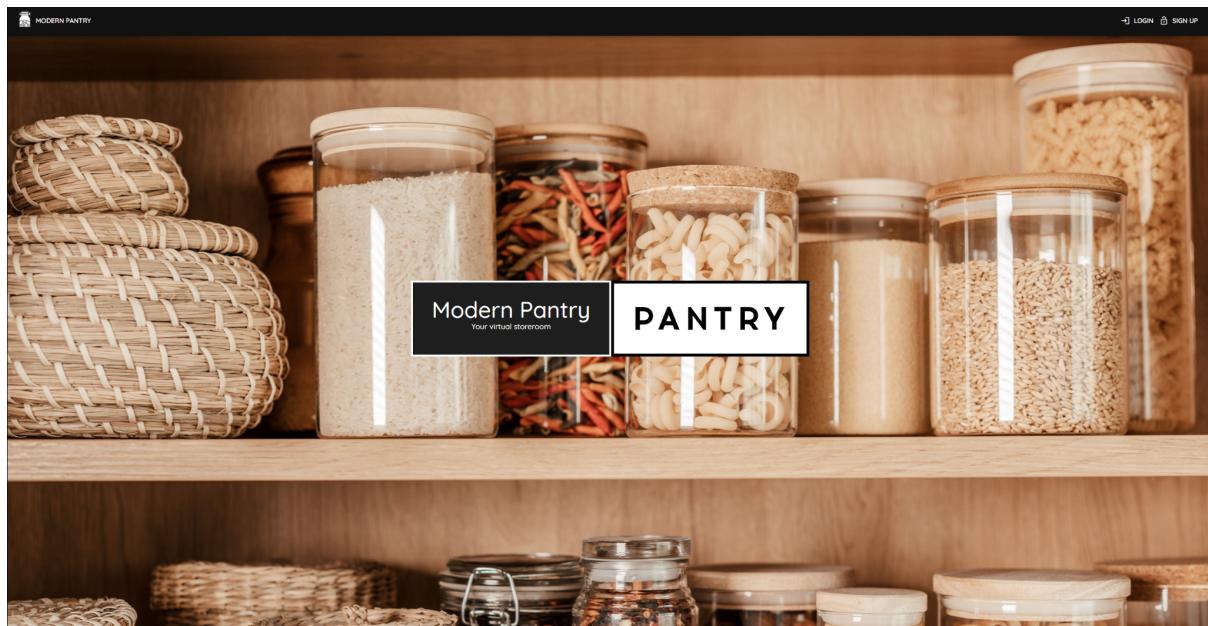
3 Open ✓ 2 Closed

- Deploy app #51 opened last week by jedrzejowsky 2 of 3 tasks
- Testing app #52 opened last week by jedrzejowsky 1 of 3 tasks
- Notifications tooltip #23 opened on Nov 11, 2022 by Zayooo00 1 of 2 tasks

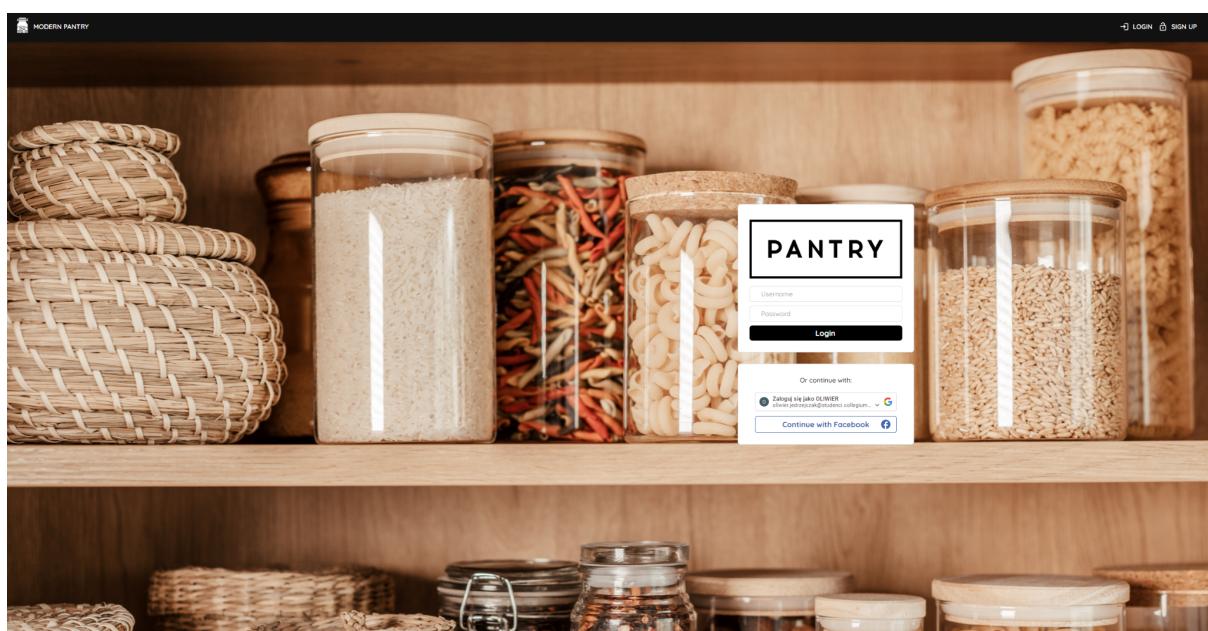
6. Działanie aplikacji

6.1. Zrzuty ekranu

Strona główna



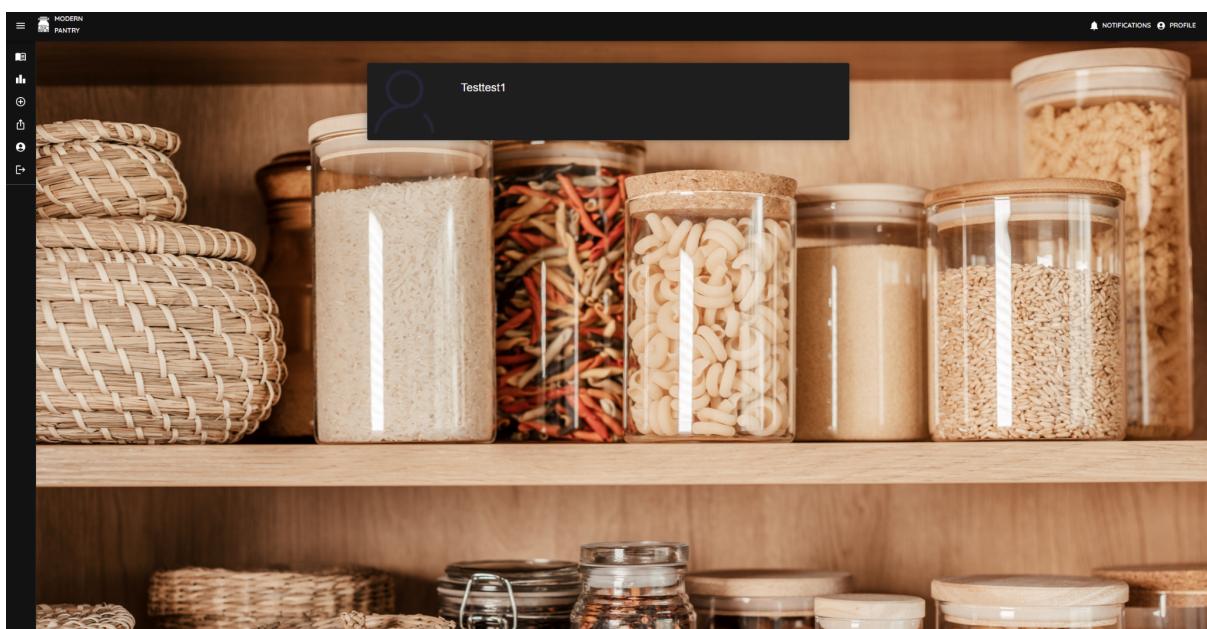
Strona logowania



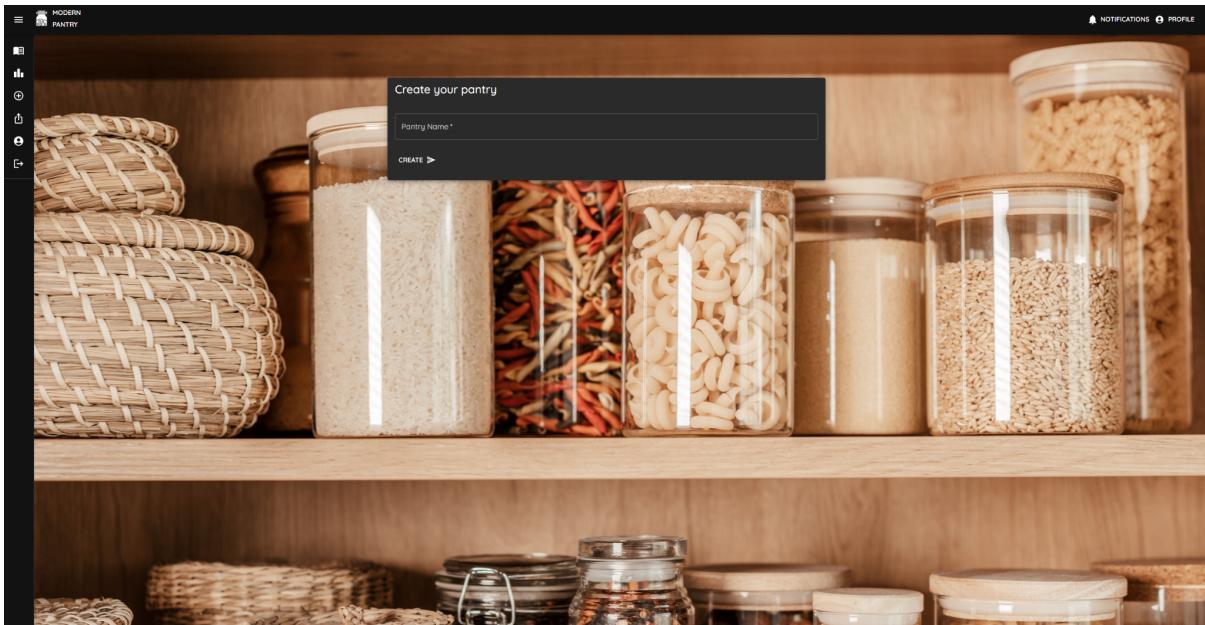
Strona rejestracji



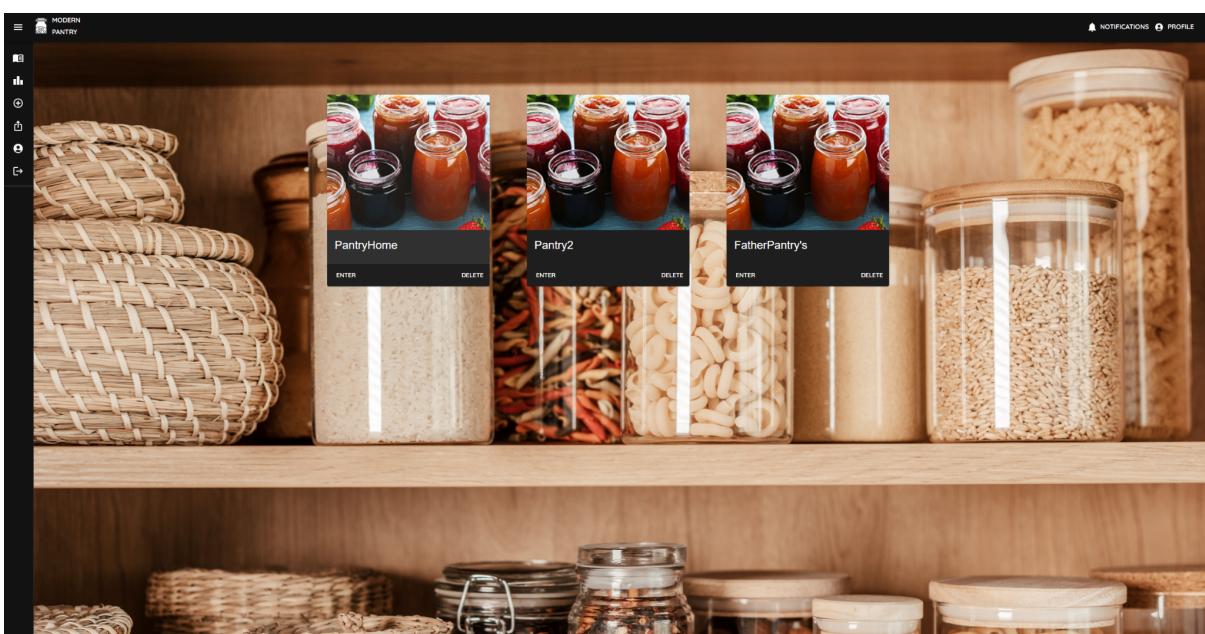
Strona po zalogowaniu



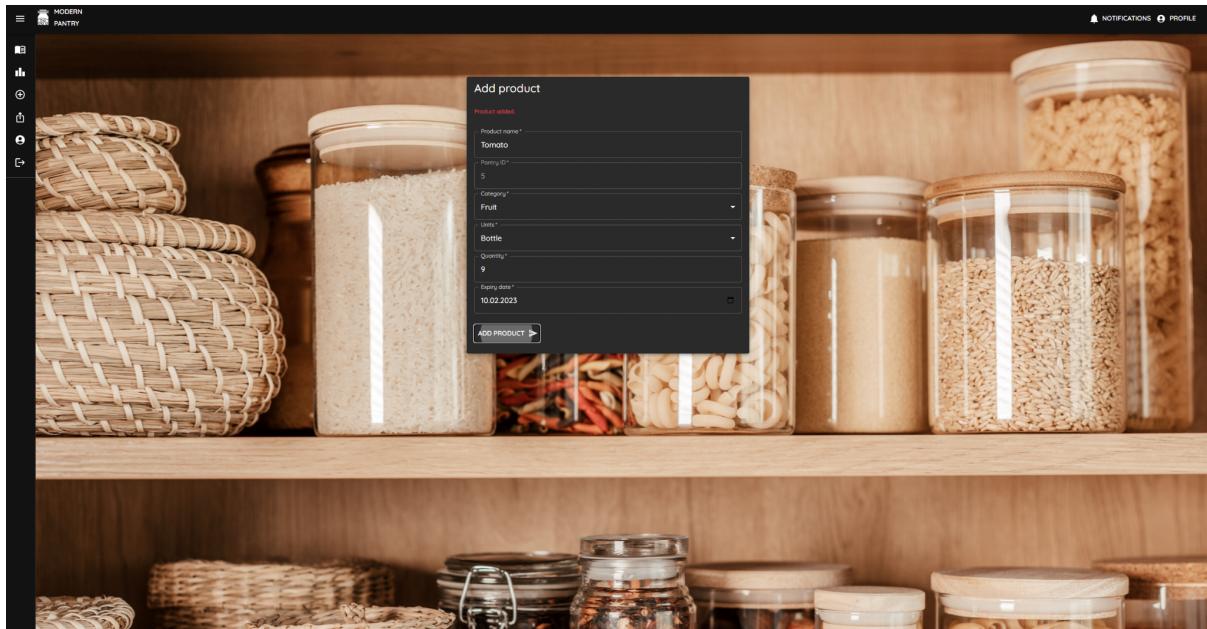
Tworzenie pantry



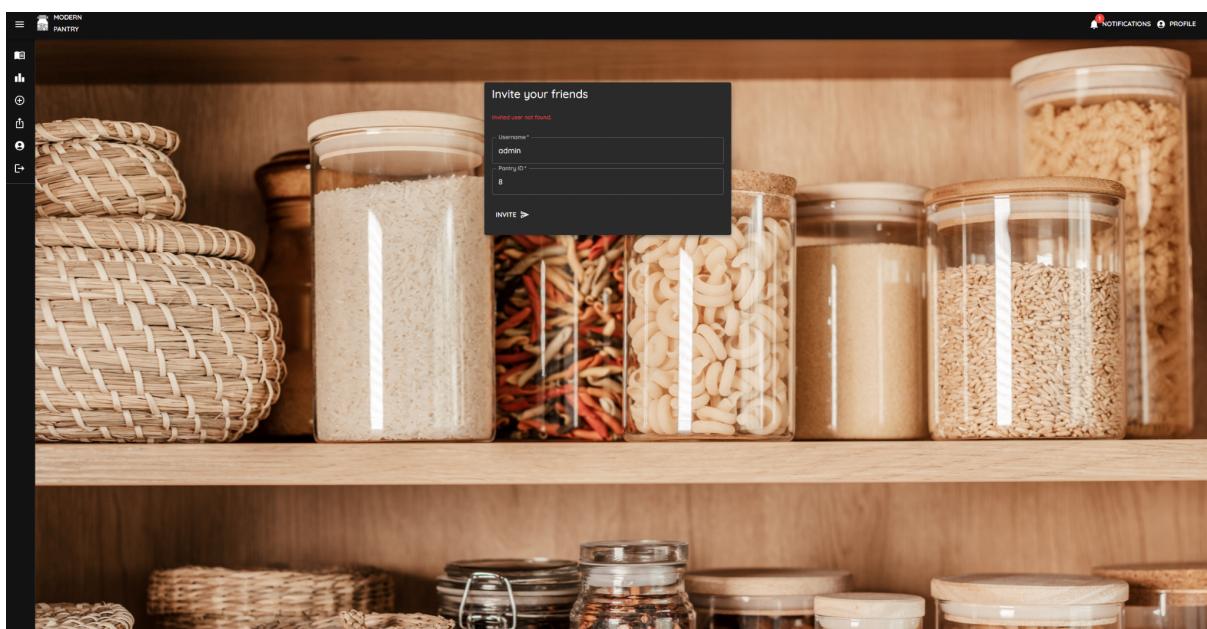
Widok na pantry



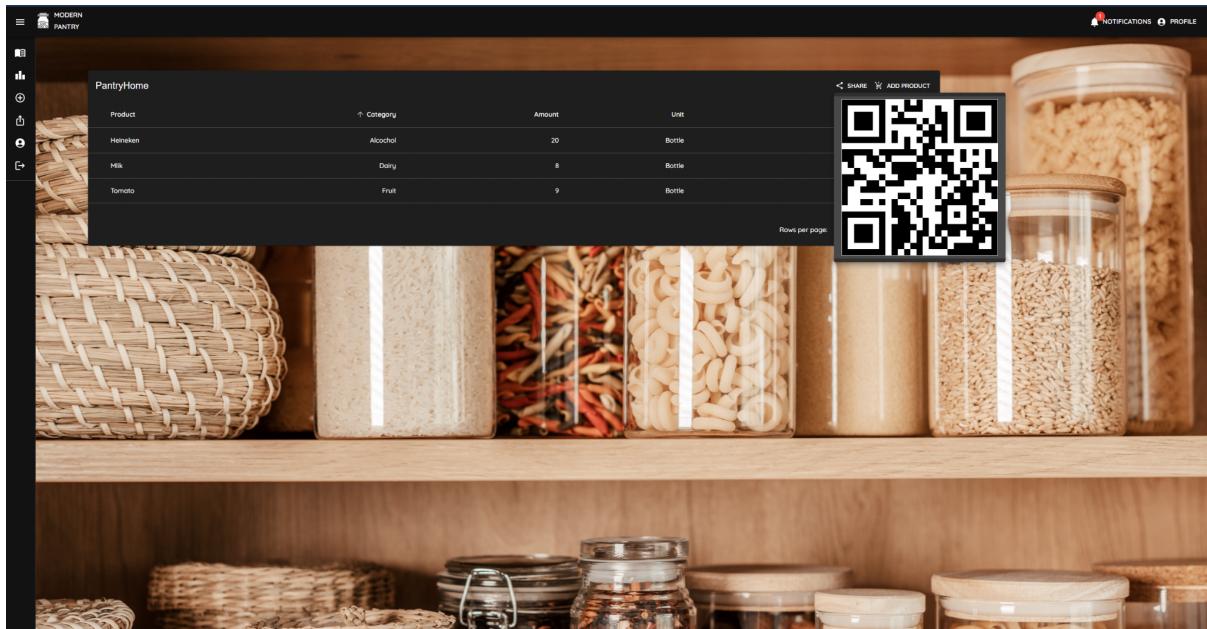
Okno dodawania produktów do pantry



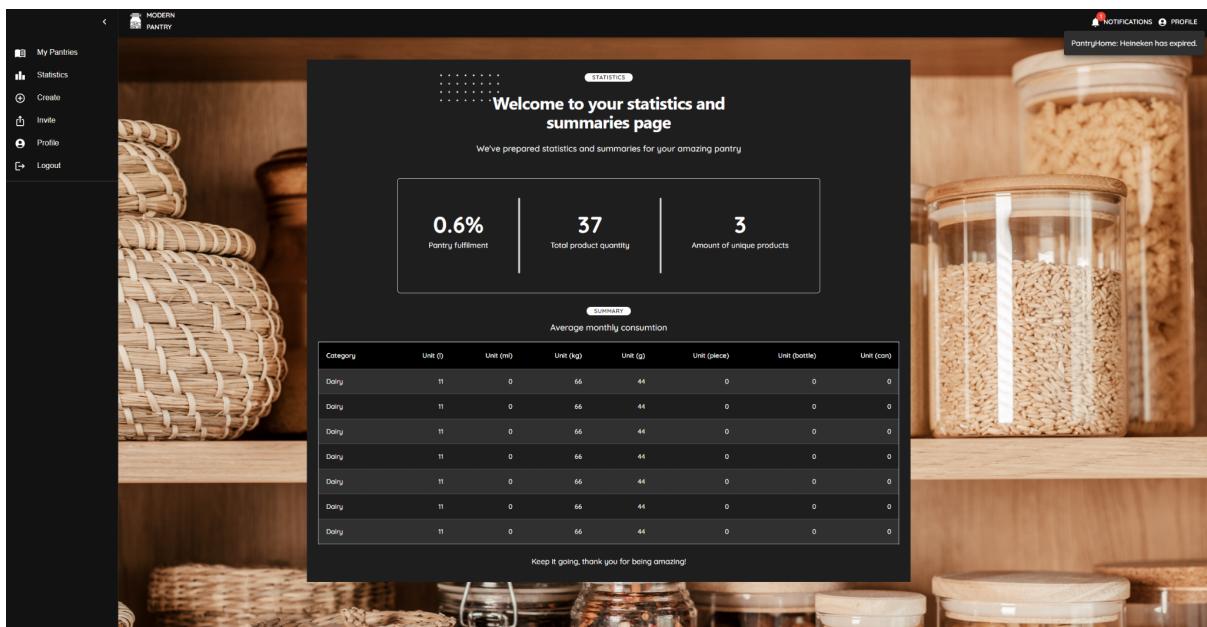
Dodawanie użytkowników do pantry



Wyświetlanie swojego pantry z kodem QR



Statystyki, podsumowanie i notyfikacja



7. Instrukcja uruchomienia

7.1. Lokalne uruchamianie systemu

Uruchomienie strony

Najpierw należy pobrać i zainstalować Node.js ze strony <https://nodejs.org/en/>.

Następnie otworzyć terminal oraz przejść do folderu Modern-Pantry oraz wpisać ‘npm install’

Po instalacji wystarczy wpisać ‘npm start’ w terminalu, po chwili komplikacji powinna otworzyć się nasza strona pod adresem ‘<http://localhost:3000>’.

Uruchomienie API

W terminalu należy wpisać ‘dotnet tool install ef –global’.

Następnie należy przejść do folderu API tj. ‘ModernPantryBackend’ znajdującego się pod ścieżką ‘Modern-Pantry\ModernPantryBackend’ oraz wpisać ‘dotnet ef database update’.

Po instalacji wystarczy wpisać ‘dotnet run’, po krótkim buildzie nasze API powinno być dostępne pod adresem ‘<https://localhost:7183/swagger/index.html>’.

7.2. Zdalne uruchamianie systemu

Uruchomienie strony

Aby zhostować nasz frontend będziemy potrzebowali darmowego konta na platformie Vercel.com, po rejestracji należy przejść przez formularz w którym wypełniamy wartości dla naszego githuba aby zachować ciągłość CI/CD wraz z każdym merge-m Main-a. Strona powinna po autoryzacji konta na githubie, wykonać pierwszy “deploy” wraz z instalacją zależności npm dla naszego frontend-u. Adres do naszej strony powinien się znajdować w wygenerowanym raporcie po udanym deploymencie.

Uruchomienie API

Hostowanie api naszej aplikacji internetowej wymaga założenia konta na stronie Azure.microsoft.com. Następnie w programie Microsoft Visual Studio 2022 wybieramy opcję publish dla naszej aplikacji w .net core 6. Następnie łączymy się z wcześniejszym kontem w usłudze Azure, i wybieramy ustawienia dla naszej aplikacji. Po tym aplikacja przygotuje się do procesu wdrażania do chmury i zostawi nam kilka możliwości, zanim zdecydujemy się na opcję “publikacji”. W tym momencie wybieramy jako dodatkową opcję połączenie z Azure SQL server oraz bazą Azure SQL, konfigurujemy pod ścieżkę naszej bazy danych i możemy w końcu wybrać opcję publikacji. Nasza aplikacja powinna się już znajdować na chmurze wraz z bazą. Po stronie serwisu Azure.microsoft musimy ustawić właściwości dla tokenu JWT, polityki CORS oraz routingów domyślnych. Adres naszego api będzie znajdująć się w zakładce App service w serwisie Azure.

8. Podsumowanie

8.1. Wnioski

Projektowanie, pisanie i wdrażanie aplikacji Modern Pantry, było nie lada wyzwaniem dla zespołu. Każdy miał nieco inną perspektywę aplikacji, dzięki dobrej komunikacji, częstym spotkaniom oraz sesji Event Stormingu, udało nam się wykłarać ostateczny punkt widzenia. Ciekawym rozwiązaniem atomizacji pracy, na poszczególnych etapach okazał się Github issues, który również ułatwia podgląd postępu nad projektem. Całość bardzo dobrze “spiął” podział na sprinty, które narzucały rygor pracy nad projektem i zachęcały do zamknięcia danego epizodu aplikacji internetowej.