

Programowanie zespołowe

Zadanie 2 - faza określenia wymagań i planowania

Piotr Popis
Mateusz Wałęjko
Adrian Majcher
Jakub Kazimierski

November 6, 2020

1 Specyfikacja wymagań

1. Użytkownik po połączeniu się z serwerem wyświetla stronę główną aplikacji.
2. Użytkownik widzi responsywną mapę polski z województwami pokolorowanymi stosunkowo do ilości zachorowań w danym dniu.
3. Użytkownik ma dostęp do wersji beta aplikacji prognozującej
 - (a) Przyrost zachorowań następnego dnia
 - (b) Przyrost zgonów następnego dnia
 - (c) Przyrost śmierci następnego dnia
4. Użytkownik ma dostęp do statystyk dla kraju, zawierających:
 - (a) Wykres liczby dziennych zachorowań od czasu
 - (b) Wykres liczby dziennych zgonów od czasu
 - (c) Wykres liczby dziennych wyzdrowień od czasu
 - (d) Wykres liczby dziennych testów od czasu
 - (e) Wykres liczby zajętych respiratorów od czasu
 - (f) Wykres liczby aktualnie zakażonych od czasu
 - (g) Wykres liczby totalnych zachorowań od czasu
 - (h) Wykres liczby totalnych zgonów od czasu
 - (i) Wykres liczby totalnych wyzdrowień od czasu
 - (j) Wykres liczby totalnych testów od czasu

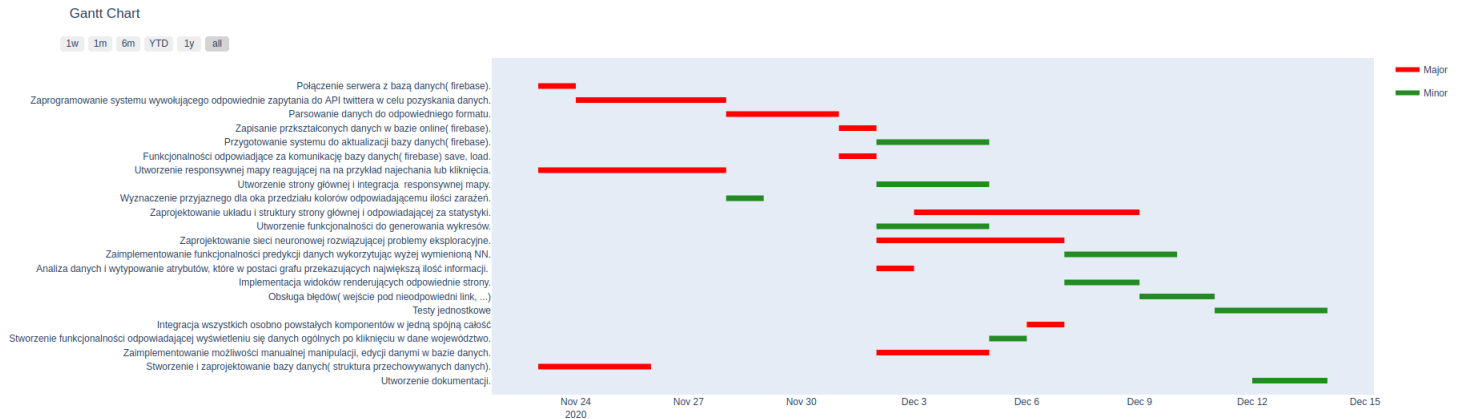
- (k) Wskaźnik ilości zachorowań na 100 tys mieszkańców
 - (l) Wskaźnik ilości aktualnie chorych na 100 tys mieszkańców(w ciągu 7 dni)
 - (m) Wskaźnik ilości zgonów na 100 tys mieszkańców(w ciągu 7 dni)
 - (n) Stosunek zachorowań do wykonanych testów danego dnia
 - (o) Przyrost zachorowań w ciągu ostatniego dnia
 - (p) Przyrost zgonów w ciągu ostatniego dnia
 - (q) Przyrost wyzdrowień w ciągu ostatniego dnia
 - (r) Przyrost testów w ciągu ostatniego dnia
5. Użytkownik poprzez najechanie ma dostęp do danych ogólnych dla danego województwa.
- (a) Przyrost zachorowań w ciągu ostatniego dnia, dla każdego województwa
 - (b) Przyrost wyzdrowień w ciągu ostatniego dnia, dla każdego województwa
 - (c) Przyrost zgonów w ciągu ostatniego dnia, dla każdego województwa
6. Użytkownik po wyświetleniu danych ogólnych dla województwa może przejść do strony poświęconej danym szczegółowym- statystyki.
7. Użytkownik po wejściu ma dostęp do statystyk takich jak:
- (a) Wykres liczby dziennych zachorowań od czasu
 - (b) Wykres liczby dziennych zgonów od czasu
 - (c) Wykres liczby dziennych wyzdrowień od czasu
 - (d) Wykres liczby zajętych respiratorów od czasu
 - (e) Wykres liczby totalnych zachorowań od czasu
 - (f) Wykres liczby aktualnie zarażonych od czasu
 - (g) Wykres liczby totalnych zgonów od czasu
 - (h) Wykres liczby totalnych wyzdrowień od czasu
 - (i) Wskaźnik ilości zachorowań na 100 tys mieszkańców
 - (j) Wskaźnik ilości aktualnie chorych na 100 tys mieszkańców(w ciągu 7 dni)
 - (k) Wskaźnik ilości zgonów na 100 tys mieszkańców(w ciągu 7 dni)
 - (l) Przyrost zachorowań w ciągu ostatniego dnia
 - (m) Przyrost zgonów w ciągu ostatniego dnia
 - (n) Przyrost wyzdrowień w ciągu ostatniego dnia

2 Podział zadań

1. Połączenie serwera z bazą danych(firebase).
2. Zaprogramowanie systemu wywołującego odpowiednie zapytania do API twittera w celu pozyskania danych.
3. Parsowanie danych do odpowiedniego formatu.
4. Zapisanie przetworzonych danych w bazie online(firebase).
5. Przygotowanie systemu do aktualizacji bazy danych(firebase).
6. Funkcjonalności odpowiadające za komunikację bazy danych(firebase) save, load.
7. Utworzenie responsywnej mapy reagującej na przykład najechania lub kliknięcia.
8. Utworzenie strony głównej i integracja responsywnej mapy.
9. Wyznaczenie przyjaznego dla oka przedziału kolorów odpowiadającemu ilości zarażeń.
10. Zaprojektowanie układu i struktury strony głównej i odpowiadającej za statystyki.
11. Utworzenie funkcjonalności do generowania wykresów.
12. Zaprojektowanie sieci neuronowej rozwiązującej problemy eksploracyjne.
13. Zaimplementowanie funkcjonalności predykcji danych wykorzystując wyżej wymienioną NN.
14. Analiza danych i wytypowanie atrybutów, które w postaci grafu przekazywających największą ilość informacji.
15. Implementacja widoków renderujących odpowiednie strony.
16. Obsługa błędów(wejście pod nieodpowiedni link, ...)
17. Testy jednostkowe
18. Integracja wszystkich osobno powstałych komponentów w jedną spójną całość
19. Stworzenie funkcjonalności odpowiadającej wyświetleniu się danych ogólnych po kliknięciu w dane województwo.
20. Zaimplementowanie możliwości manualnej manipulacji, edycji danymi w bazie danych.
21. Stworzenie i zaprojektowanie bazy danych(struktura przechowywanych danych).
22. Utworzenie dokumentacji.

3 Harmonogram projektu(zilustrować za pomocą schematu Gantta).

Uprzejmie jednak prosimy o skorzystanie z wersji interaktywnej dostępnej pod linkami: [Files.fm](https://files.fm), [Github.com](https://github.com) . Poniżej znajduję się tylko screenshot.



4 Opis cech charakterystycznych technologii wybranych do realizacji projektu oraz uzasadnienie, dlaczego powinny one zostać użyte w projekcie.

1. **Firestore**- baza danych online pozwala na aktualizowanie danych z dowolnego miejsca. Jeśli jakikolwiek użytkownik spróbuje połączyć się z naszą aplikacją nasze dane zostaną zaktualizowane niezależnie od miejsca i urządzenia. Kolejnym atutem użycia takiej technologii jest to, że pozwala nam ona na przechowywanie danych do generowania wykresów czy diagramów.
2. **Django**- pośredniczy w komunikacji klient serwer. Odpowiada za operacje nad bazą danych, przekazywanie danych i renderowanie odpowiednich template'ów.
3. **HTML**- korzystamy z jego "pudełkowej" struktury do grupowania i uporządkowania elementów naszej strony.
4. **CSS**- pozwala na odpowiednia koloryzację, dodanie atrybutów, pozycjonowanie.
5. **Javascript**- jest potrzebny do utworzenia pewnych dynamicznych zachowań na naszej stronie na przykład do utworzenia mapy, wykorzystu-

jemy go do dynamicznego zmieniania pewnych cech elementów pod wpływem jakiegoś działania na przykład przesunięcia paska.

6. **canvasJS, chartJS** Generowanie wykresów, diagramów w sekcji statystyki, ogólna wizualizacja danych.
7. **tweepy** Wrapper do api twittera, wykorzystywany do pobierania danych z oficjalnego konta ministerstwa zdrowia