**DOKUMENTACJA PROJEKTU**

**Klasy**

**Zmienne**

**Metody**

**Zrzuty ekranu**

**Instrukcje obsługi**

**Dlaczego zdecydowałem się na:**

**-Desktop: Windows Forms C#**

**Ponieważ mam doświadczenie w tej technologii i języku programowania.**

**-Webowy: HTML, CSS, JavaScript**

**Ponieważ znam HTML, CSS, JavaScript**

**-Mobilny: Android - Java/Kotlin**

**Ponieważ najwygodniej jest mi w nim pisać pod aplikacje mobilne.**

**-Serwer: Node.js**

**Ponieważ jest najpopularniejszy obecnie.**

**-Baza Danych: MongoDB**

**Ponieważ chcę się go nauczyć.**

**TI - wytyczne do projektu**

Idea projektu:

(w skrócie, ponieważ szczegółowo była omawiana na ostatnich zajęciach)

W ramach zajęć będą realizowane elementy rozproszonego i wielowarstwowego systemu wspomagającego edycję plików tekstowych przez Internet z możliwością pracy offline (dotyczy tylko aplikacji natywnej).

Plan realizacji.

**lab 1** - Zajęcia wprowadzające. Omówienie jak będą przebiegać zajęcia.

**lab 2** - Wykonanie prototypu umożliwiającego **wysłanie (client- HTTP request)** oraz **odbiór(server- HTTP response)** przykładowego komunikatu z wykorzystaniem **REST API** i **protokołu WebSocket!**

lab 3 - Podstawowa funkcjonalność **Klienta Natywnego:**

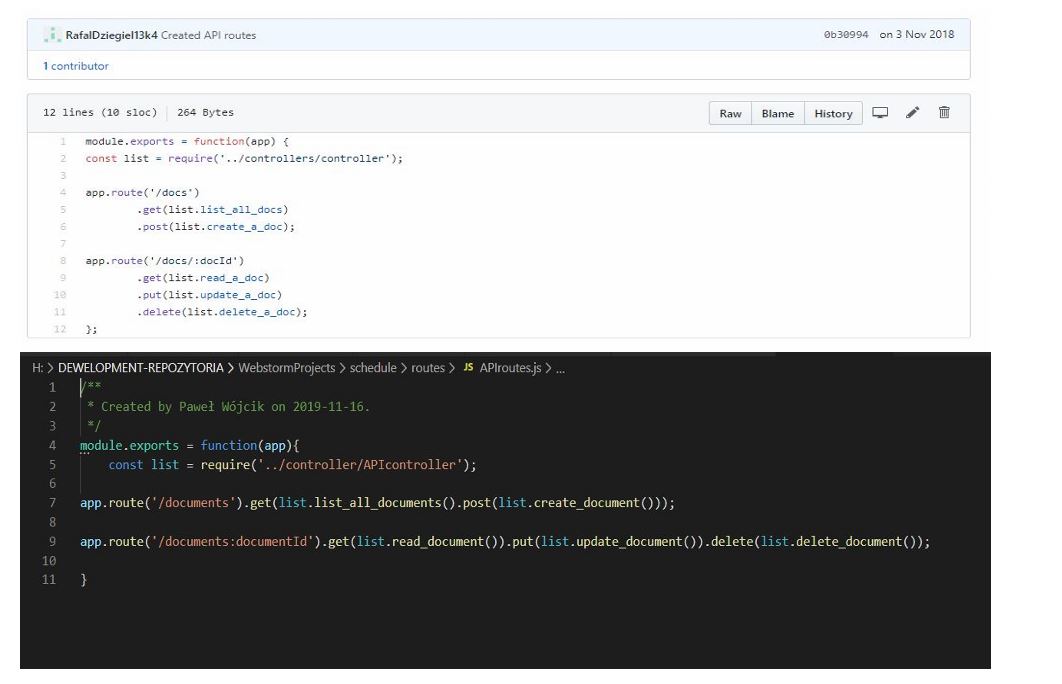
-**pobranie listy plików z serwera, @GET**

-**wyświetlenie zawartości pliku, @GET**

-**edycja pliku i wysłanie zmian na serwer, @POST**

-**utworzenie nowego pliku @POST**

Komunikacja z **Serwerem** powinna odbywać się za pomocą **REST API.**



 lab 4 - Podstawowa funkcjonalność **Klienta Webowego (implementacja w technologii "aktywnej", czyli bez odświeżania strony):**

-**wyświetlanie aktualnej listy plików,**

-**tworzenie,**

-**wyświetlenie i edycja pliku.**

Komunikacja z **Serwerem** powinna odbywać się za pomocą **REST API.**

lab 5 - Użycie komunikacji za pomocą protokołu **WebSocket** do informowania o różnych aktywnościach w systemie typu: **użytkownik zablokował (edytuje) plik, pojawił się nowy plik (lub zmieniła się lista plików), wewnętrzny chat pozwalający na komunikację użytkowników między sobą** (np. w celu poproszenia o zwolnienie pliku). Materiał wideo - kernelgonnapanic

lab 6 - **Synchronizacja plików w Kliencie Natywnym oraz zapewnienie możliwości pracy w trybie offline**. Proszę pozwolić również na edycję plików offline i przesłanie go na serwer kiedy pojawi się Internet oraz jeśli na serwerze nikt nie zmienił pliku.

lab 7 - Implementacja dodatkowej funkcjonalności poprawiającej użyteczność systemu lub ergonomię pracy z systemem. Przykład: **implementacja powiadomień natywnych** (mogą to być również powiadomienia z przeglądarki internetowej). Można również zgłaszać własne propozycje.

Uwagi:

\* Oczekuję od wszystkich zespołów zrealizowania pełnej funkcjonalności wskazanej w punktach lab 3 - lab 6. Za wykonanie tego zakresu prac maksymalna ocena to 4,5. Realizacja lab 7 nie jest obowiązkowa jednak pozwoli na uzyskanie oceny 5,0.

\* Pomimo tego, że projekt UI strony i aplikacji nie jest najważniejszy, to powinien być on przygotowany starannie z zachowaniem obowiązujących standardów (np. warto skorzystać z **Bootstrapa** lub innego **frameworka CSSowego** do zagwarantowania **responsywności**).

\* Zastrzegam sobie możliwość precyzowania wymagań na kolejnych zajęciach. Wspólnie będziemy mogli dyskutować jak dane zagadnienie zrealizować. Proszę się zatem nie zamykać na opis, który jest powyżej.

\* Proszę o terminowe realizowanie kolejnych etapów. Dany etap powinien być najpóźniej zrealizowany do końca kolejnych zajęć. Przykład: zalecane jest, aby do końca 3 zajęć był zrealizowany 3 temat. Jeśli jednak nie będzie skończony to nie będzie z tego powodu żadnych konsekwencji. Problemem jednak będzie to, że ktoś przykładowo na 3 zajęciach nie będzie miał etapu 2.

\* Do zaliczenia projektu niezbędne będzie pokazanie swojej aktywności w repozytorium kodu.

\* Informuję również Państwa, że podczas zaliczenia projektu będą mogły pojawić się pytania nie koniecznie z obszaru, który ktoś indywidualnie realizował. Chcę w ten sposób uniknąć sytuacji, że ktoś mimo iż napracuje się w projekcie nie wyjdzie poza zakres pewnych prostych technologii typu html, css, jquery.

**Funkcjonalności:**

**-Rejestracja,**

**-Logowanie,**

**-Autoryzacja,**

**-Uwierzytelnianie,**

**-Dodanie użytkownika do bazy danych,**

**-Chat**

**-Responsywność**

**-REST API**

**pomoc:**

**https://www.youtube.com/watch?v=mY5tCERAads&ab\_channel=MarekZaj%C4%85c**