**Zadanie 1.**

**18.10.2019**

**1.1 Temat:** Aplikacja do rozpisywania treningów na siłowni.

**Opis:** Aplikacja będzie umożliwiać zaplanowanie kilku dni treningowych. W obrębie dnia treningowego będzie można wybrać ćwiczenia z listy lub dopisać ręcznie, a dla każdego ćwiczenia będzie można wybrać ilość powtórzeń, obciążenie oraz długość przerwy między seriami. Aplikacja będzie umożliwiała odznaczanie wykonanych ćwiczeń w trakcie treningu oraz aktualizację danych gdy będzie taka potrzeba (na przykład zmiana obciążeń).

Istnieje wiele aplikacji, które swoją funkcjonalnością obejmują nasz temat. Mieliśmy okazję używać kilku z nich lecz każdej czegoś brakowało. Chcemy stworzyć aplikację kompletną, która sprosta wymaganiom każdego użytkownika siłowni.

**1.2 Skład grupy:**

* Piotr Klepczyk – lider
* Jakub Duda
* Mateusz Laskowski

**1.2 Specyfikacja wymagań:**

* **Technologie:** AngularJS, Cordova, Ionic
* **Języki programowania:** JavaScript, HTML, CSS

**1.3 Struktura podziału pracy:**

* Piotr Klepczyk – koordynowanie pracy zespołu, tworzenie dokumentacji, stworzenie okna nowego treningu.
* Jakub Duda – stworzenie okien dodawania ćwiczeń oraz dokumentacja dotycząca tej części,
* Mateusz Laskowski – utworzenie projektu, stworzenie okna aktywnego treningu oraz dokumentacji dotyczącej tej części.
* Ponad to każdy z członków zespołu będzie przeprowadzał testy komponentów tworzonych przez pozostałych, a także brał udział w projektowaniu wyglądu i wszystkich funkcjonalności aplikacji.

**1.4 Wstępny harmonogram projektu:**

* Do końca października – stworzenie pierwszego okna aplikacji, wraz z obsługą pamięci urządzenia
* Do 17.11.2019 – stworzenie wszystkich okien aplikacji bez uwzględniania wyglądu ostatecznego
* Do 24.11.2019 – testy aplikacji
* Do 8.12.2019 – doprowadzenie aplikacji do końca

**1.5 Cechy charakterystyczne wybranych technologii:**

Wybrane przez nas technologie łączą w sobie zalety aplikacji webowych oraz natywnych aplikacji mobilnych. Dzięki wykorzystaniu języków z aplikacji webowych praca deweloperska jest szybsza i łatwiejsza, zwłaszcza w aspekcie tworzenia interfejsu użytkownika. A dzięki możliwości korzystania z SDK systemowych mamy pełen dostęp do podzespołów urządzenia jak pamięć, bądź aparat.

**1.6 Uzasadnienie wyboru technologii:**

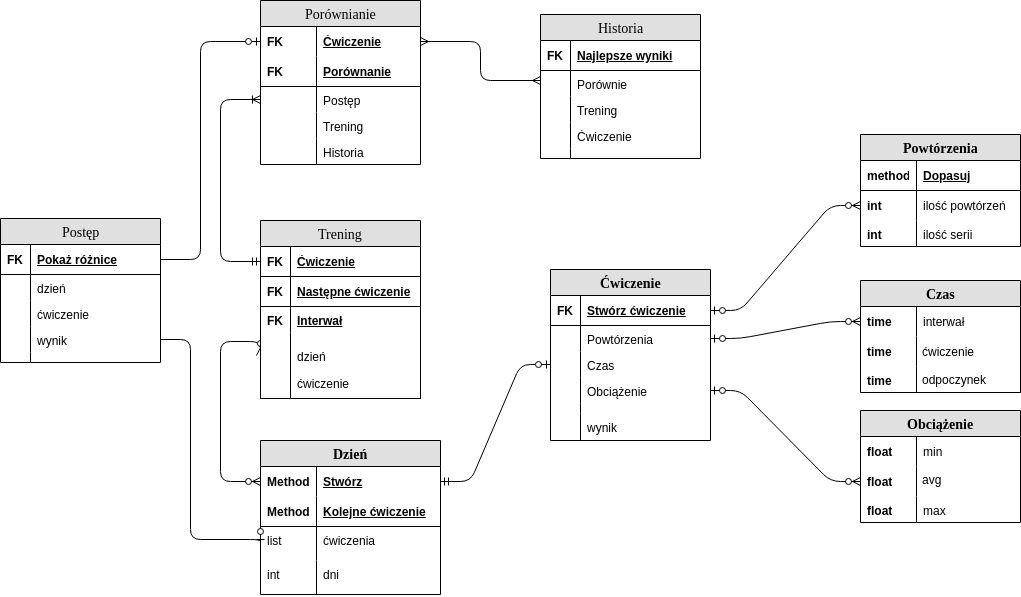
Technologia ta świetnie nadaje się do naszego projektu ze względu na to że większość funkcjonalności ma miejsce w obrębie interfejsu.

**Zadanie 2.**

**25.10.2019**

**2.1 Projekt systemu:**

* Aplikacja będzie umożliwiać zaplanowanie kilku dni treningowych.
* W obrębie dnia treningowego będzie można wybrać ćwiczenia z listy lub dopisać ręcznie.
* Dla każdego ćwiczenia będzie można wybrać ilość powtórzeń, obciążenie oraz długość przerwy między seriami.
* Aplikacja będzie umożliwiała odznaczanie wykonanych ćwiczeń w trakcie treningu oraz aktualizację danych gdy będzie taka potrzeba (na przykład zmiana obciążeń).
* Ćwiczenia można analizować porównując wyniki z już odbytymi ćwiczeniami.
* Dni treningowe wraz z ćwiczeniami zapisują się.

**2.1 Diagram UML**

**2.1 Projekt systemu:**

|  |  |  |
| --- | --- | --- |
| **Ryzyko projektowe** | **Plan naprawczy** | **Metody zapobiegania** |
| Utrata projektu poprzez utrata danych z serwera dyski sieciowe i repozytorium. | Stworzenie projektu poprzez połączenie elementów (merge) uzyskanych przez członków zespołu. | Zapisywanie projektu lokalnie na dysku (nie sieciowym). |
| Nie pożądane zmiany w projekcie, zniszczenie części projektu, np. dodanie swojego projektu bez łączenia z istniejącą wersją. | Powrót do ostatniej działającej wersji z systemu kontroli wersji. | Założenie systemu kontroli wersji częste dodawanie zmian wraz z komentarzami tagami. |
| Utrata jednego z kluczowych członków zespołu projektowego. | * Zaproszenie do zespołu nowego członka np. z innej grupy. * Przejęcie obowiązków nieobecnego członka | Rozmowa z członkami projektu, zapewnienie dyspozycji. |
| Nie zdążenie zakończyć projektu na czas. | * Poprosić o dłuższy termin oddania pracy. * Zaproszenie do zespołu nowego członka np. z innej grupy. * Zwiększenie obowiązków i czasu pracy | * Dokładnie uzgodnić działania z czasem ich wykonania. * Przy tworzeniu harmonogramu zostawienie czasu (awaryjnego) * Utworzyć punkty milowe aby kontrolować postęp pracy * kontrolować prace zespołu |

**2.3 Podział prac implementacyjnych :**

* Stworzenie repozytorium .
* Stworzenie pierwszego okna aplikacji, wraz z obsługą pamięci urządzenia.
* Stworzenie wszystkich klas.
* Relacje między klasami.
* Stworzenie wizualnych okien aplikacji.
* Dodawanie funkcjonalności tworzenia treningu .
* Dodawanie funkcjonalności analizy postępu treningu .
* Tworzenie historii treningów.
* Testowanie.