

Uniwersytet Gdański
Wydział Matematyki, Fizyki i Informatyki
Instytut Informatyki
Aplikacje Internetowe i Bazy Danych
Studia dzienne

P r o j e k t Z e s p o ł o w y

**Edytor tekstu on-line do współpracy programistycznej
w czasie rzeczywistym**

**SPRAWOZDANIE 2: Ustalenia kwestii technicznych
i formalnych, doprecyzowanie założeń projektu**

Skład osobowy zespołu:

Lipkowski Paweł
Mazepa Mariusz (lider)
Zakrzewski Michał

Gdańsk 2017

1. Rozwiązanie problemu wyboru lidera i podział obowiązków w zespole

Liderem grupy jest, zgodnie z ustaleniami zawartymi w sprawozdaniu pierwszym, Mariusz Mazepa.

Podział obowiązków w zespole przedstawia się następująco (kolejność alfabetyczna względem nazwiska):

- 1) **Lipkowski Paweł** – administrator bazy danych
 - a. Stworzenie i sprawowanie pieczy nad bazą danych
 - b. Obsługa zapytań bazodanowych
 - c. Łączenie bazy danych z aplikacją – wyciąganie danych z bazy i umożliwienie przechwycenia ich z poziomu aplikacji
- 2) **Mazepa Mariusz** (lider zespołu) – webmaster, grafik komputerowy
 - a. Front-end tworzonego systemu
 - b. Obsługa skryptów wykonywanych po stronie użytkownika
 - c. Dbanie o walory estetyczne i wizualny wygląd aplikacji webowej, kwestie graficzne
- 3) **Zakrzewski Michał** – webdeveloper (programista)
 - a. Back-end tworzonego systemu
 - b. Odpowiedzialność za całe zaplecze aplikacji
 - c. Stworzenie silnika aplikacji i dbałość o jego poprawne funkcjonowanie w procesie użytkowania

Z racji, że na tym etapie projektu nie mamy jeszcze pełnej wizualizacji koniecznych do wykonania zadań i obowiązków do spełnienia, zastrzegamy sobie prawo do ewentualnej modyfikacji wyżej wymienionych założeń. Jako oczywisty (nieuwzględniony w podziale) uznajemy również fakt, że niektóre etapy rozwoju aplikacji będą wymagały współpracy więcej niż jednej osoby – zakładamy możliwość wspierania współtwórców projektu w trakcie realizacji zadań uznanych przez nas za trudne.

2. Rozwiązanie problemu komunikacji w zespole

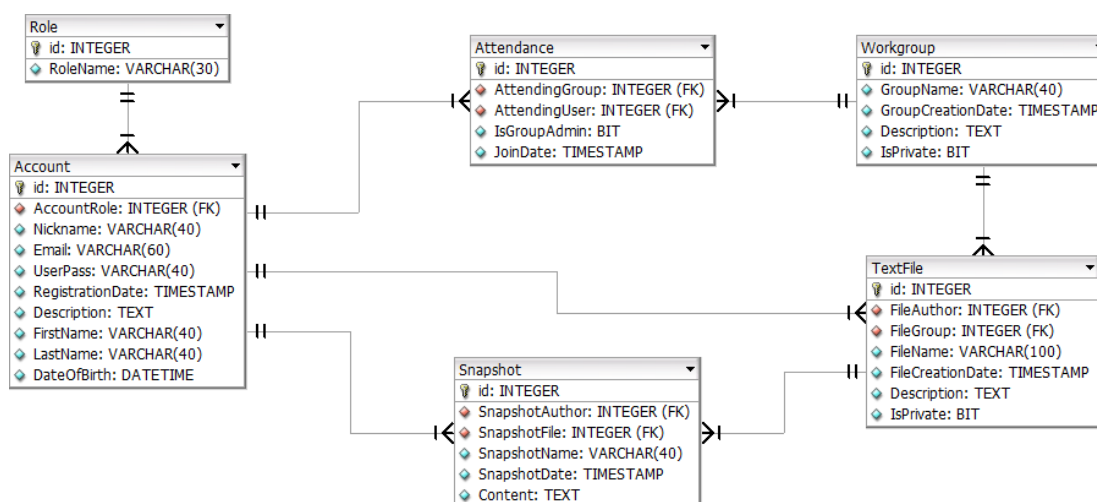
Komunikacja w zespole odbywać się będzie na dotychczas ustalonych (i sprawdzonych) zasadach wyróżniających dwa etapy. Pierwszym z nich, dość ogólnym, zawierającym przede wszystkim ustalenia ogólne, jest kontakt bezpośredni, m.in. w trakcie trwania zajęć z Projektu Zespołowego. Drugi etap, to komunikacja zdalna poprzez popularną w ostatnim czasie aplikację Messenger udostępnianą m.in. dla użytkowników serwisu Facebook.

3. Rozwiązanie problemu wersjonowania tworzonego systemu

Wersjonowanie tworzonego systemu zamierzamy przeprowadzać dzięki wolnodostępnemu, rozproszonemu systemowi kontroli wersji, jakim jest Git. W tym zakresie korzystać będziemy z funkcjonalności hostingowego serwisu internetowego GitHub, na którym zamierzamy rozwijać nasz projekt pod postacią repozytorium.

4. Dopracowanie diagramów utworzonych na laboratorium 1

Schemat ERD (dopracowany):



Chwilowo nie przewidujemy konieczności dopracowywania pozostałych stworzonych przez nas diagramów. W przypadku zaistnienia jakiegokolwiek zmiany, nie omieszkamy uwzględnić jej w jednym z kolejnych sprawozdań.

5. Wybór technologii implementacji systemu (dobór np. frameworka)

W kwestii technologii implementacji systemu planujemy wykorzystać następujące technologie:

1) Baza danych

- a. Wolnodostępny system zarządzania relacyjnymi bazami danych MySQL lub PostgreSQL (sprecyzujemy później w zależności od dalszych potrzeb i możliwości współdziałania z resztą projektu)

2) Front-end

- a. Skryptowy język programowania JavaScript
- b. Hipertekstowy język znaczników HTML w wersji 5
- c. Kaskadowe arkusze stylów CSS w wersji 3 i Less
- d. Frameworki: AngularJS lub Angular2 (rozważymy, co nam bardziej podpasuje, gdy zaczniemy pracę nad projektem) i Bootstrap

3) Back-end

- a. Obiektowy język programowania Java (z wykorzystaniem możliwości technologii Java Platform Enterprise Edition)
- b. Narzędzie automatyzujące budowę oprogramowania na platformę Java – Apache Maven

6. Dobór modułów, które mogą zostać wykorzystane podczas realizacji systemu (np. moduł autoryzacji)

Prawdopodobnie będziemy poszukiwać modułu umożliwiającego kompilację kodu źródłowego wprowadzanego przez użytkownika do tworzonego przez nas edytora tekstu on-line. Wykorzystać też chcemy darmowe wsparcie open source'owe do rozpoznawania i kolorowania semantyki kodu – dwóch pretendentów: Highlight.js i Google Code Prettify.

7. Dobór środowiska pracy i jego implementacja/konfiguracja

Pracę planujemy wykonywać na uniksopodobnych systemach operacyjnych opartych na jądrze Linux (wolne i otwarte oprogramowanie), których dystrybucja zależy wyłącznie od preferencji i upodobań każdego z nas. Cały projekt opierać się będzie, jak już wspomnieliśmy podczas rozplanowania technologii, na narzędziu automatyzującym budowę oprogramowania na platformę Java – Apache Maven. Dobór edytora tekstowego również zależy od indywidualnych upodobań i potrzeb każdego z nas, jednakowoż najprawdopodobniej wykorzystamy platformę (framework) do tworzenia aplikacji typu rich client, jakim jest Eclipse, desktopowy edytor programistycznych kodów źródłowych z kolorowaniem składni dla wielu języków Atom, z racji łatwości edytowania dużych plików.