Projekt indywidualny,

Małgorzata Szwed

**Elektroniczny system obsługi siłowni: Twoja siłownia, online**

Celem projektu jest stworzenie dynamicznej aplikacji webowej we frameworku Ruby on Rails do obsługi elektronicznego systemu zarządzania siłownią.

W systemie współegzystują dwa rodzaje użytkowników:

-trenerzy

-osoby ćwiczące/kursanci

Trenerzy układają każdemu kursantowi plan treningowy (ćwiczenie/udział w zorganizowanych zajęciach) dostosowany do jego potrzeb i aktualnej kondycji. Analizują postępy kursantów. Układają dietę kursantom. Prowadzą konkretne zajęcia (plan zajęć). Przeprowadzają egzaminy kursantom badając ich postępy.

Kursant – ma wgląd do planu treningowego na dany tydzień/miesiąc/okres. Monitoruje swoje postępy i urealnia termin osiąganych celów. Po wykonaniu zadanych ćwiczeń (niekontrolowanych bezpośrednio przez trenera) wpisuje swoje wyniki w portalu. Wpisuje także odstępstwa od diety.

Taka wizja projektu daje możliwość znacznego jej rozbudowywania. Na cele uczelniane powstanie jednak tylko podstawowa funkcjonalność.

Potrzebne programy:

-Git

-Rvm

-interpreter języka ruby

-rails

-postgresql

-rubyMine

-vim

-firefox

Szczegółowa dokumentacja:

Etap 1

Nauka rubyiego I frameworku ruby on rails.

<http://www.railstutorial.org/book>

Instalacja oraz konfiguracja niezbędnego oprogramowania

Etap 2

Wizualizacja początkowa projektu; wstępny zarys; decyzje architektoniczne.

Etap 3

Skrótowy opis działań:

-stworzenie pustej aplikacji railsowej (rails new gym)

-skonfigurowanie Gemfile

-założenie repozytorium git I powiązanie z githubem

-dostosowanie gitignore

-ustawienie frameworków do testów automaczynych (rspec, spork, guard) oraz TDD/BDD (Capybara, Cucumber)

-powiązanie repozytorium z heroku, wysłanie strony na publiczny serwer

-stworzenie stron statycznych (główna, o nas)

-dostosowanie wyglądu (scss, twitter bootstrap), dobranie obrazów

-napisanie testów do stron statycznych sprawdzających zawartość

-wygenerowanie plików konfiguracyjnych (modelu, kotrolera, …) dla diet, zajęć oraz trenerów (rails generate scaffold …)

-dostosowanie praw dostępu do wygenerowanych stron (routes.rb)

-wpisanie przykładowych informacji na publiczny serwer heroku do bazy postgresql

-zablokowanie praw edycji od strony anonimowego użytkownika

-obsługa logowania

-obsługa gravataru

-obsługa planu zajęć

-obsługa wiadomości

Wnioski:

Po długiej pracy udało się stworzyć portal dokładnie według założeń projektowych.