

## 1 spoj.pl

SPOJ jest serwisem typu Online judge umożliwiającym sprawdzenie swoich umiejętności programistycznych poprzez rozwiązywanie problemów. Serwis zawiera szereg problemów różnej trudności, które można rozwiązać w różnych językach programowania. Rozwiązaniem jest kod źródłowy programu napisanego w wybranym języku programowania. Odpowiedzi wysyła się do serwisu poprzez specjalny formularz.

Do rozwiązania jest ponad 8000 problemów sformułowanych m.in. w językach: polskim, angielskim, portugalskim i po wietnamsku. Można wysyłać kody źródłowe w 55 językach programowania, jest możliwość organizowania własnych konkursów.

SPOJ jest używany do prowadzenia zajęć dydaktycznych z programowania i algorytmiki jako narzędzie do wstępnej, automatycznej oceny programów oraz do przeprowadzenia konkursów, w tym rekrutacyjnych.

## 2 codeschool

CodeSchool naucza technologii webowych w komfortowy sposób, czyli przez naszą przeglądarkę. Na stronie są dostępne lekcje wideo oraz konkursy programistyczne i inne wyzwania. To sprawia, że nauka programowania staje się przyjemna.

## 3 topcoder

TopCoder to amerykańska firma zajmująca się organizacją konkursów programistycznych.

TopCoder ocenia swoich zawodników, przyznając im punkty rankingowe, wyliczane m. in. na podstawie osiągniętych wyników i częstotliwości uczestnictwa w konkursach. Na podstawie ratingu zawodnicy zostali podzieleni na dwie dywizje: do pierwszej z nich należą ci, których rating przekroczył 1200 punktów, do drugiej kwalifikują się pozostali.

## 4 Podstawowy opis projektu

Nowoczesne technologie wymagają nowoczesnych metod uczenia, które w niebanalny sposób zmieniają podejście do nauki. Często młodzi ludzie (licealiści) są zniechęceni uczeniem się nowych technologii, gdyż muszą przedzierać się przez nudne manuale. Naszym projektem chcieliśmy zmienić, pokazać nową drogę jaką może iść edukacja. A mianowicie kurs programowania z bazą danych pełną zadań do samodzielnej pracy. Nasze podejście i inowacyjność projektu polega na interaktywności – uczeń dostaje natychmiastową informację zwrotną o rezultatach nauki. Informacja zwrotna zawiera czas wykonania programu, ocenę obliczaną na podstawie wykonania programu. Nasz program jest w postaci aplikacji internetowej, do której dostęp będą mieli wszyscy do podłączeni do sieci. Celem aplikacji jest dostarczenie wygodnych i ciekawych sposobów nauki w języku programowania C++. Kurs będzie podzielony na lekcję, którą wraz z zadaniami będą tworzyć pełną całość. Taka część kursu może być oceniana i stanowić ocenę częściową całego zadania, z których można wystawić ocenę z całości kursu.

## 5 Wymagania funkcjonalne

- użytkownik posiada konto w serwisie
- użytkownik posiada podstronę, na której znajdują się informacje o nim, jak i dotychczasowe osiągnięcia w nauce, statystyki, informacje kontaktowe

- użytkownik może dodawać swoje zadania z serwisu
- użytkownik może modyfikować swoje zadania w serwisie
- użytkownik może kasować swoje zadania w serwisie
- użytkownik może wysyłać swoje zadania w celu sprawdzenia
- użytkownik może zostać administratorem serwisu, nabywać w ten sposób więcej praw do zarządzania
- użytkownik dodający zadania do serwisu może nakładać ograniczenia na wykonywanie zadania; ograniczenia mogą być takie jak czas, pamięć, dane wejściowe
- administrator może dodawać nowe zadania do serwisu
- administrator może modyfikować wszystkie zadania w serwisie
- administrator może kasować wszystkie zadania w serwisie
- użytkownik może się rejestrować w serwisie
- użytkownik aby wykonać zadanie jest potrzebne konto w serwisie
- każdy użytkownik ma dostęp do kursu
- każdy kurs jest podzielony na lekcję (moduły), którą kończą się testem/zadaniem
- w serwisie jest dostępny ranking najlepszych programistów
- użytkownik ma prawo do dodawania komentarzy pod zadaniem oraz pod lekcją w kursie
- użytkownik może skasować swoje konto w serwisie
- użytkownik może posiadać na swojej podstronie informację o sobie; np. adres email, stronę www, szkołę, ulubiony język programowania, avatar
- strona posiada formularz kontaktowy do administratora(ów)
- zadania w serwisie są podzielone na poziomy trudności np. łatwe, średnie, trudne
- niezarejestrowany użytkownik będąc na stronie startowej widzi tylko pierwszą lekcję kursu, dopiero po rejestracji ma dostęp do pełnego kursu

## 6 Wymagania niefunkcjonalne

## 7 Zastosowane technologie

W naszym projekcie zastosowaliśmy następujące technologie:

- Ruby 2.2
- Ruby on Rails 4.2.0
- Javascript
- HTML5/CSS3

- Postgresql

**Ruby** interpretowany, w pełni obiektowy i dynamicznie typowany język programowania stworzony w 1995 roku przez Yukihiro Matsumoto.

Ruby bazuje na wielu językach, takich jak CLU, Eiffel, Lisp, Perl, Python czy Smalltalk. Składnia jest zorientowana liniowo i oparta na składni CLU oraz, w mniejszym stopniu, Perla. Ruby posiada:

- automatyczne odśmiecanie pamięci
- iteratory
- przeciążanie operatorów (ang. operator overloading)
- normalne i zaawansowane właściwości zorientowania obiektowego (klasa, metoda ...)
- obsługa wyjątków (ang. exception handling)
- wyrażenia regularne wbudowane w składnię
- liczby całkowite o dowolnych rozmiarach
- dodawanie metod do instancji klasy - możliwa jest zmiana lub dodanie metody do instancji danej klasy
- bloki i lambdy (closures) - wygodne przekazywanie funkcji jako parametrów
- "Duck typing- rozpoznawanie typów na podstawie ich zachowania, a nie deklaracji
- moduły - rodzaj wielodziedziczenia pozwalający włączyć gotową implementację zbioru metod do danej klasy
- możliwość zmiany praktycznie wszystkiego - dodanie dodatkowych metod do klasy Array, czy zmiana sposobu drukowania liczb całkowitych są niezmiernie proste.
- zmienne leksykalne modyfikowalne w czasie działania programu
- Programy pisane w Ruby charakteryzują się wysoką przenośnością pomiędzy platformami (istnieją implementacje na wiele systemów Uniksowych, DOS, Windows, OS X, BeOS itd.).

Ruby jest rozwijany jako otwarte oprogramowanie i dostępny na licencji GPL oraz na własnej licencji wymagającej zmiany nazw plików wykonywalnych w przypadku zamknięcia kodu.

**Ruby on Rails** framework open source do szybkiego tworzenia aplikacji webowych stworzony głównie przez duńskiego programistę Davida Heinemeiera Hanssona w ramach pracy nad oprogramowaniem Basecamp. RoR został napisany w języku Ruby z użyciem architektury MVC (ang. Model-View-Controller).

**JavaScript** skryptowy język programowania, stworzony przez firmę Netscape, najczęściej stosowany na stronach internetowych. Najczęściej spotykanym zastosowaniem języka JavaScript są strony WWW. Skrypty służą najczęściej do zapewnienia interaktywności poprzez reagowanie na zdarzenia, sprawdzania poprawności formularzy lub budowania elementów nawigacyjnych. Skrypty JavaScriptu uruchamiane przez strony internetowe mają znacznie ograniczony dostęp do komputera użytkownika.

## 8 Modele, relacje i schematy

## 9 Ewolucja systemu

**HTML** (ang. HyperText Markup Language) – hipertekstowy język znaczników, obecnie szeroko wykorzystywany do tworzenia stron internetowych.

**CSS** (ang. Cascading Style Sheets, w skrócie CSS) to język służący do opisu formy prezentacji (wyświetlania) stron WWW. Arkusz stylów CSS to lista dyrektyw (tzw. reguł) ustalających w jaki sposób ma zostać wyświetlana przez przeglądarkę internetową zawartość wybranego elementu (lub elementów) (X)HTML lub XML. Można w ten sposób opisać wszystkie pojęcia odpowiedzialne za prezentację elementów dokumentów internetowych, takie jak rodzina czcionek, kolor tekstu, marginesy, odstęp międzywierszowy lub nawet pozycja danego elementu względem innych elementów bądź okna przeglądarki. Wykorzystanie arkuszy stylów daje znacznie większe możliwości pozycjonowania elementów na stronie, niż oferuje sam (X)HTML.

**Postgresql** jeden z najpopularniejszych wolnodostępnych systemów zarządzania relacyjnymi bazami danych.

## 10 Modele, relacje i schematy

## 11 Ewolucja systemu

Nie wiem co tutaj masz na myśli... Jak system będzie się rozwijał/zmieniał? Moim zdaniem nie ma potrzeby tego tutaj pisać bo sami się wykopujemy, chyba że zamierzasz rozwijać ten projekt w ramach dobrego serca :)