Универзитет „Св. Кирил и Методиј“ - Скопје
ФАКУЛТЕТ ЗА ИНФОРМАТИЧКИ НАУКИ И КОМПЈУТЕРСКО ИНЖЕНЕРСТВО

Напредни веб-базирани алатки за компјутерска едукација

Code веб-базирана околина за програмирање

Томче Делев
tdelev@finki.ukim.mk
Ментор:
Дејан Ѓорѓевиќ
dejan.gjorgjevikj@finki.ukim.mk

Факултет за компјутерски науки и инженерство

Октомври 2013

Агенда

- Вовед
- Мотивација
- Хипотеза
- Дизајн и имплементација
- Идеи за идна работа
- Заклучок

Мотивација

Програмирање

- Програмирањето е една од основните практични вештини која што се изучува во програмите на информатичките факултети и секундарна вештина на многу други технички факултети
- Големата побарувачка на пазарот на трудот за програмери е една од причините за интересот и популарноста меѓу идните студенти
- Резултат се големи групи студенти во воведните курсеви

Мотивација

10 години

- Но програмирањето не е лесно!
- Истражувачите (Bloom (1985), Bryan & Harter (1899), Hayes (1989), Simmon & Chase (1973)) имаат покажано дека се потребни околу десет години да се постигне одредено ниво на експертиза во многу различни области како играње шах, компонирање музика, работа со телеграф, сликање, свирење пиано, пливање, тенис и многу други.
- Програмирањето не е исклучок (Winslow (1996))
- Потребно е да се решат голем број основни алгоритамски проблеми (повеќе од една третина од времето)

Потешкотии при програмирање

1. Инсталација и поставување околина за програмирање
2. Користење текстуален едитор за код
3. Разбирање на проблемот и користење програмски јазик за пишување код
4. Разбирање на грешките при компајлирање
5. Дебагирање (откривање грешки)

Хипотеза

Цел на истражувањето

- Олеснување на учењето програмирање кај почетници со помош на напредна веб-базирана околина која овозможува чести и информативни повратни информации
- Зголемување на мотивираноста и успехот на студентите во воведните курсеви
- Помош на наставниот кадар во администрација на големите групи со студенти

Предложена методологија

- Прашалници и анкети
- Обработка на статистиката за користење
- Анализа и споредба на резултатите
- Издвојување на контролна група

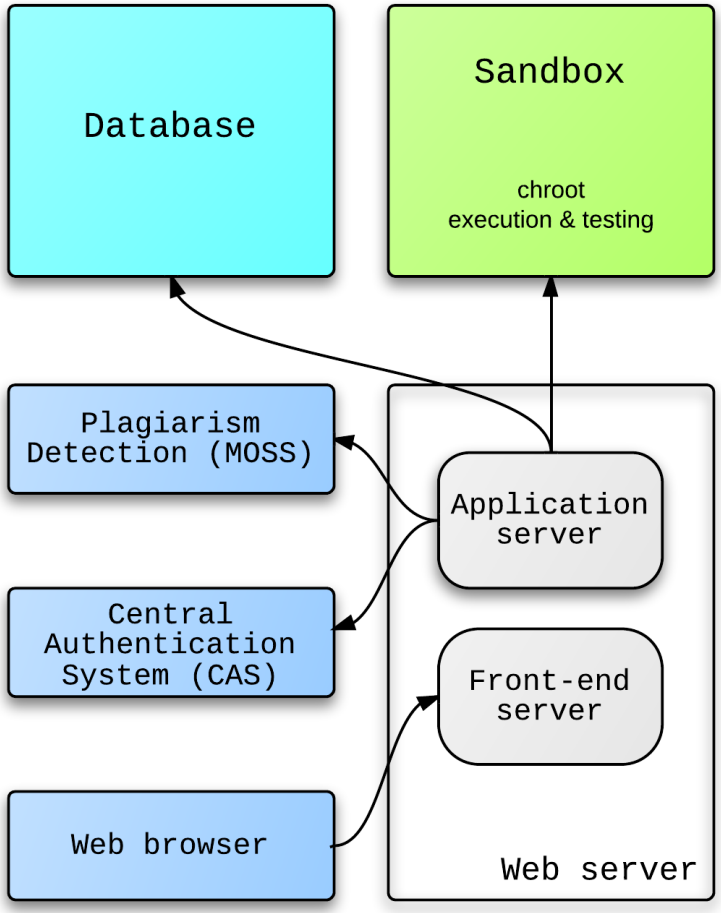
Седум принципи на добра практика (Chickering and Gamson (1987))

1. Контакт помеѓу студентот и факултетот
2. Соработка меѓу студентите
3. Активно учење
4. Навремени одговори
5. Нагласување на временските ограничувања
6. Соопштување на високите очекувања
7. Почитување на различностите во талентот и начините на учење

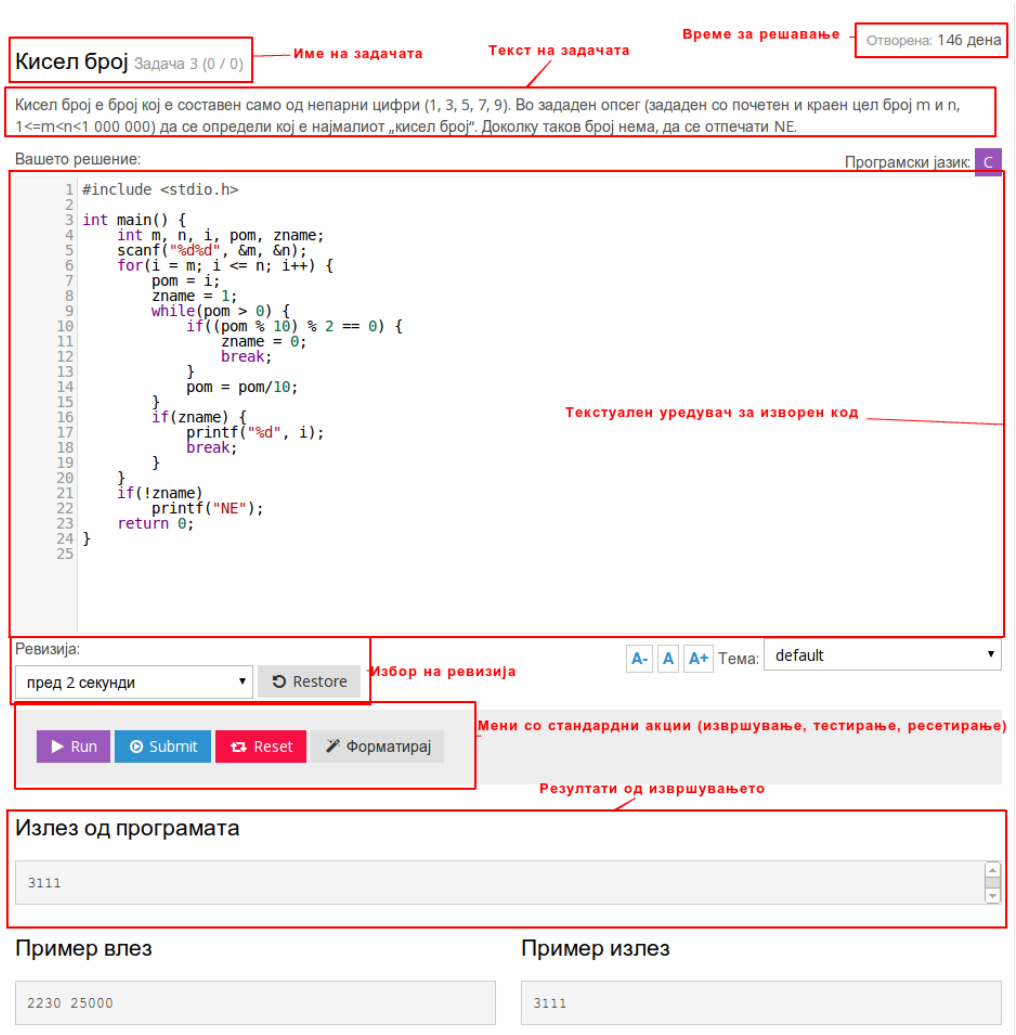
Што е Code?

- *Code* е веб-базиран систем за решавање задачи од програмирање во почетни курсеви за програмирање
  - Интегриран уредувач за код со поглед на задачата
  - Интегриран уредувач на задачи и тест примери
  - Автоматско оценување и генерирање повратни информации (feedback)
  - Управување со корисници и курсеви (централна автентикација)
  - Подржува различни програмски јазици (C, C++, Java, Python)
  - Заштитено извршување (Sandbox)
  - Детекција на плагијати

Архитектура на системот



Интегриран поглед на задача



Оценување

Генерирање повратни информации (feedback)

- Автоматско
  - Динамичка анализа
  - Статичка анализа
- Од инструкторите
- Помеѓу студентите (peer assesment)

Динамичка анализа

- Динамичка анализа или автоматско тестирање е извршување на програмата на студентот врз одредено множество на тест податоци
- Алгоритамски видови на задачи
- Black-box и White-box тестирање
- Се користи во системи за натпревари во програмирање
- Ја потенцираат важноста во програмирањето да се има решение кое работи точно

Статичка анализа

- Процес на анализирање на програмата без да се извршува
  - Комплексност
  - Големина
  - Разбирливост
  - Структура на код
- Начини на статичка анализа
  - Проверка на стилот на програмирање
  - Детекција на можни грешки
  - Мерење на карактеристиките со помош на софтверски метрики
  - Оценување на дизајнот на програмата
  - Проверка на одредени функционалности

Податоци за користење

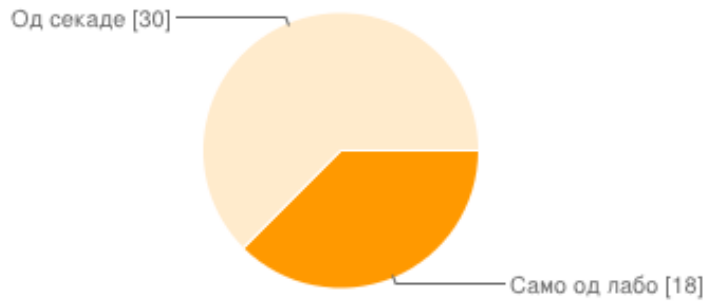
Корисници и курсеви

- Корисници
  - 2243
- Курсеви
  - Структурно програмирање
  - Алгоритми и податочни структури
  - Напредно програмирање
  - Објектно-ориентирано програмирање
  - Напредни алгоритми

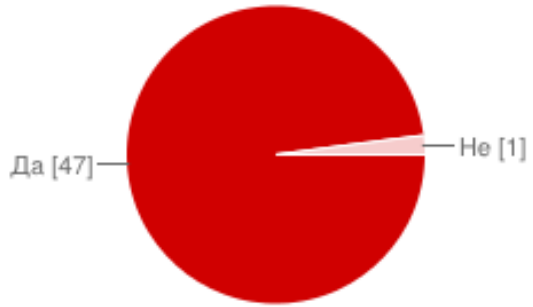
Анкета (48 одговори)

Пристап

На code пристапувам од?

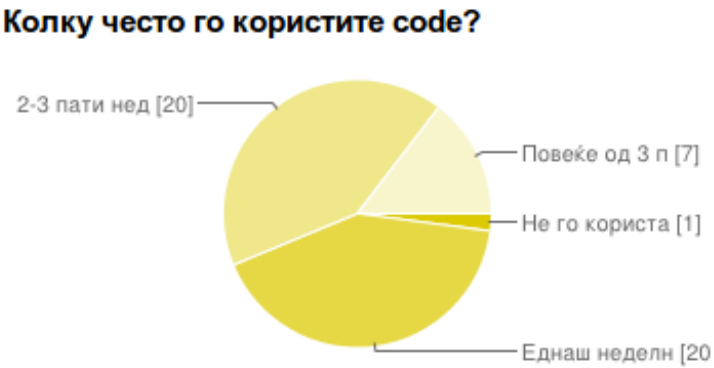


Дали сакате да имате пристап од секаде?



Анкета (48 одговори)

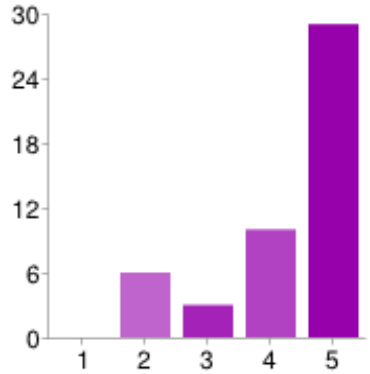
Користење и едноставност



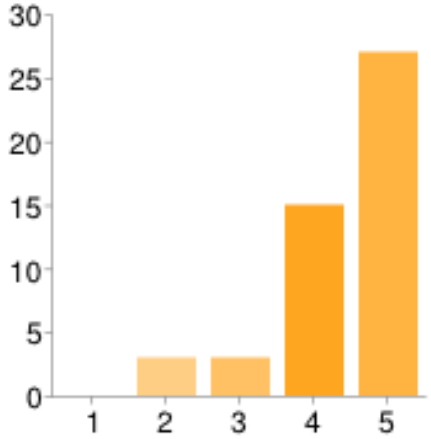
Анкета (48 одговори)

Приказ и перформанси

Приказ на задачата која треба да се реши?



Перформанси и брзина на одзив?



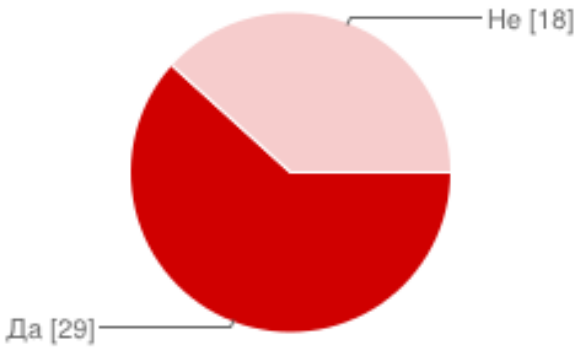
Анкета (48 одговори)

Начин на користење

Анкета (48 одговори)

Дали Code помага во точно решавање на задачите

Дали сметате дека code ви помага во процесот на точно решавање на задачата?



Да	**29**	62%
Не	**18**	38%

Плагијаризам

- Measure of Software Similarity (MOSS) - A. Aiken
- Направена е анализа за плагијаризмот во рамките на два предмети **Концепти за развој на софтвер (C)** и **Напредно програмирање (Java)**
- Кај задачите од лабораториски вежби („контролирана средина“) забележани се бројни плагијати (на некои задачи и преку 30%)
- Кај испитите и колоквиумите не се забележани случаи на плагијати
- Постојат и позитивни примери на индивидуални домашни работи

Заклучок и понатамошна работа

Досегашна работа

- Реализација и имплементација на веб-базирана околина за програмирање Code
- Преглед и споредба со постеочките околини и алатки за автоматско оценување на задачи за програмирање
- Истражување на теоријата на учење во контекст на техничките науки со посебен аспект на програмирањето како практична вештина
- Прв чекор за приклучок кон светските трендови на приближување на образованието кон поголема маса на луѓе (MOOC), со користење на најновите технологии

- Збогатување и подобрување на повратните информации (feedback)
- Интеграција со систем за учење
- Колаборативно учење
- Персонализирано и адаптибилно учење
- Следење и визуелизација на извршувањето
- Архитектура за скалабилна имплементација во облак

Колаборативно учење

Идеи за понатамошна работа

- Тековните истражувања имаат идентификувани многу бенефити од колаборативно учење посебно во воведни прогрмерски курсеви (DeClue, 2003; GILD, 2005; McDowell et al., 2003; Nagappan et al., 2003; Preston, 2005; Wilson et al., 1993)
- Покажано е дека и крајните резултати се подобри кога студентите работат заедно за време на семестарот (McDowell et al., 2002; McDowell et al., 2003; Nagappan et al., 2003)
- Синхрони алатки (chat, колаборативен едитор)
- Асинхрони алатки (форуми, емаил)

Прашања

http://code.finki.ukim.mk

Ви благодарам на вниманието

Прашања?