

Санкт-Петербургский Государственный Университет  
Математико-механический факультет

Кафедра Системного Программирования

Кладов Алексей Александрович

# Разработка системы проверки упражнений для образовательной платформы

Дипломная работа

Допущена к защите.  
Зав. кафедрой:  
д. ф.-м. н., профессор Хунта К. Х.

Научный руководитель:  
д. ф.-м. н., профессор Выбегалло А. А.

Рецензент:  
ст. преп. Вяххи Н. И.

Санкт-Петербург  
2014

SAINT-PETERSBURG STATE UNIVERSITY  
Mathematics & Mechanics Faculty

Chair of The Meaning of Life

Edelweis Mashkin

# Empty subset as closed set

Graduation Thesis

Admitted for defence.  
Head of the chair:  
professor Christobal Junta

Scientific supervisor:  
professor Amvrosy Vibegallo

Reviewer:  
assistant Alexander Privalov

Saint-Petersburg  
2014

# Оглавление

Введение	4
1. Платформа Stepic	6
2. Постановка задачи	8
3. Реализованные типы упражнений	9
4. API для создания новых типов упражнений	10
5. Изолированное исполнение кода упражнений	11
Заключение	12

# Введение

## В современном мире

Научно технический прогресс стремительно меняет все сферы современной жизни. В последнее время технологии Интернет активно используются чтобы улучшить качество предоставляемого образования. Комплекс технологий и программ, осуществляющих такое улучшение, получил название online образования.

## Особенности online образования

online образование отличается некоторыми особенностями.

Во-первых, это большое количество студентов в одном потоке, как правило достигающее нескольких десятков тысяч человек.

Во-вторых, хотя учебные материалы уникальны для каждого курса, они, как правило, очень высокого качества – видео лекции, субтитры, аннотированные слайды, книги.

В-третьих, мотивация студентов курса зачастую не очень высокая.

В-четвёртых, свободный формат online образования позволяет учиться в удобное для студента время.

## История

Пожалуй, история компьютерного образования началась с началом распространения персональных компьютеров, если не раньше, однако можно выделить следующие вехи:

Walter Lewin – профессор физики из MIT, лекции которого показывались по местному телевидению

open courseware – MIT сделал все учебные материалы доступными в Интернет.

Udacity – первый стартап, с MOOC в настоящем его понимании.

Coursera – наиболее успешная на настоящий момент платформа для online образования

## Проблемы

online образованию присущи и некоторые специфичные проблемы. Среди них можно выделить следующие.

- Ограничены возможности проверки знаний. Так как упражнения должны проверяться автоматически, то их набор часто ограничен.

- Большая стоимость создания online курса. Создание курса требует значительных затрат на запись видео лекций, оформление электронного конспекта и создание набора упражнений.

- Drop out. Для online образования характерен значительно меньший процент заканчивающих курс.

## **Существующие платформы**

\*\*\* Udacity 2011 год 1.6 млн \*\*\* Coursera 2012 год 7.1 млн \*\*\* edX 2012 год 2.1 млн MOOC.org

# 1. Платформа Stepic

## Статус проекта

Stepic это молодой проект, развивающийся в рамках компании JetBrains. Разработка проекта стартовала в 2013 году. На текущий момент в проекте занято 8 человек.

## Использование

На текущий момент Stepic используют 23 тысячи студентов. На платформе открыто 400 уроков, многие из которых объединены в курсы.

Из завершившихся курсов стоит отметить следующие.

”Алгоритмы в биоинформатике”

Этот курс проходил одновременно с соответствующим курсом на coursera, он состоял из большого количества текстовых материалов и упражнений на программирование из области алгоритмической биологии. В этом курсе приняли участие более 10 тысяч человек со всего мира.

”Алгоритмы и Структуры Данных”

Это закрытый на настоящий момент курс, который использовался для предварительной оценки знаний абитуриентов Computer Science Center в Санкт-Петербурге. В нём приняли участие 500 человек из Петербурга. Курс состоял из видео лекций на русском языке. Для курса использовались разные типы упражнений, но наиболее часто упражнения на программирование.

## Возможности платформы

Stepic ориентирован на интеграцию и сотрудничество с другими инструментами online образования. Для этого поддерживаются стандарты oEmbed и LTI.

Единицей учебного материала на Stepic является урок – набор упражнений и/или теории представленный в виде слайдов, общим количеством не превосходящий 16 штук. Уроки можно объединять в курсы.

Курс позволяет создавать секции уроков, назначать разным упражнениям разные стоимости, создавать коллективы студентов и преподавателей, устанавливать предельные сроки сдачи упражнений.

Теория может быть представлена в виде html слайдов, или в виде видео лекций.

Необходимость расширять возможности автоматически проверяемых упражнений.

## Технологии и Инструменты

В качестве базы данных используется MySQL. Разработка серверной части ведётся на языке Python 3, с использованием фреймворка django. Клиентская часть

разрабатывается на CoffeeScript, с использованием Ember.js.

Также используются celery для распределённого выполнения заданий, codejail в качестве основы системы изолированного исполнения кода, flask тоже зачем-то используется.

## 2. Постановка задачи

Целью работы является реализация системы для создания и проверки упражнений для образовательной платформы Stepic, с возможностью легко добавлять новые типы упражнений, в том числе и сторонним разработчикам.

Для достижения этой цели были сформулированы следующие задачи.

- Реализовать в Stepic типы упражнений, часто встречающихся в других образовательных платформах и проверить их работу на практике.
- Обеспечить возможность лёгкого расширения набора типов упражнений сторонними разработчиками (реализовать соответствующий API к платформе Stepic) и проверить на практике его удобство.
- Реализовать возможность масштабирования и изолированного исполнения потенциально не безопасного кода упражнений.



### 3. Реализованные типы упражнений

Общий вид упражнения. Взаимодействие пользователя с упражнением в общем виде можно описать следующим образом. Сначала пользователь читает условие упражнения. Затем он нажимает на кнопку "начать решать". После этого пользователю представлен один из вариантов dataset. Пользователь взаимодействует с клиентской частью упражнения и составляет свой ответ. Когда ответ готов, пользователь нажимает на кнопку отправить, после чего ответ проверяется. После проверки пользователь видит результат (бинарное верно/неверно или оценка – вещественное число от 0 до 1) и возможный отзыв.

dataset отличается от попытки к попытке и генерируется случайным образом на сервере. Вместе с dataset создаётся ключ к решению – объект, с помощью которого можно быстро проверить ответ студент.

Генерация пар dataset/ключ и проверка ответа пользователя происходит асинхронно. При этом dataset/ключ регенерируется заранее, а ключ выбирается таким образом, чтобы проверка ответа выполнялась быстро. Таким образом, большую часть работы можно выполнить заранее, и ускорить получение студентом обратной связи. Примеры конкретных упражнений ... .. Использование упражнений в реальных курсах. В курсе "Алгоритмы в биоинформатике" большую часть упражнений составляли dataset quizzes. Этот тип упражнений оказался наиболее удобен, так как позволяет делать задачи про обработку больших объёмов данных, что характерно для биоинформатики, позволяя при этом использовать любой язык программирования.

В курсе "Алгоритмы и структуры данных" большую часть упражнений составляли code quizzes и free answer quizzes. Code quizzes оказались удобны, так как позволяют ограничить решения по времени и памяти, что необходимо для курса по алгоритмам. Free answer квисы использовались для проверки теоретических задач.

## 4. API для создания новых типов упражнений

\*\* Плагины Плагины позволяют создавать упражнения специально для курса. Для плагинов был создан публичный репозиторий.

\*\* Архитектура решения Серверная часть написана на Python. Сохраняются квидзы в виде JSON. Клиентская часть может быть написана на javascript или coffescript, с необязательным использованием ember. \*\* Сервер для разработки Для упрощения разработки плагинов написан сервер, позволяющий проверить работу плагина без

## 5. Изолированное исполнение кода упражнений

Были реализованы следующие типы упражнений: - choice quiz - string quiz - dataset quiz - code quiz - number quiz - math quiz Они были использованы в курсах по биоинформатике и алгоритмам и структурам данных.

Разработан API для создания серверной и клиентской части упражнений.

На основе code jail и celery создана система безопасного распределённого исполнения кода упражнений.

## Заключение