|  |
| --- |
| **МОСКОВСКИЙ КОМИТЕТ ОБРАЗОВАНИЯ**  **ЛИЦЕЙ №1533 (ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ)** |

Выпускная работа

(Специальность «Прикладное программирование»)

Учащихся группы 11-4 Молчанова Михаила Алексеевича и Горбачева Александра Максимовича

**Название работы**

Программные средства для проведения электронного тестирования учащихся с помощью устройств на базе Android и web интерфейса.

Заказчик: Гиглавый Александр Владимирович, Лицей информационных технологий №1533

Консультанты: Славнов Константин Анатольевич, Завриев Николай Константинович, Лицей информационных технологий №1533

Оглавление

[1. Введение 3](#_Toc437886165)

[2. Цели и задачи исследования 4](#_Toc437886166)

[3. Обзор аналогов. Актуальность исследования 5](#_Toc437886167)

[3.1. MyTest 5](#_Toc437886168)

[3.2 Moodle 6](#_Toc437886169)

[3.3 Websoft CoursLine 6](#_Toc437886170)

[3.4 LIT Test 6](#_Toc437886171)

[4. Анализ предметной области 6](#_Toc437886172)

[4.1 Что такое тест 6](#_Toc437886173)

[4.2 Виды тестов 6](#_Toc437886174)

[4.3 Виды тестовых заданий 7](#_Toc437886175)

[5 Решение 8](#_Toc437886176)

[5.0 Архитектура системы 8](#_Toc437886177)

[5.1 Хранение данных 8](#_Toc437886178)

[5.1.1 Хранение тестов 8](#_Toc437886179)

[5.1.2 Хранение вопросов 9](#_Toc437886180)

[5.2. Структура данных 9](#_Toc437886181)

[5.2.1 Взаимодействие классов 9](#_Toc437886182)

[5.2.2 Типы тестов 9](#_Toc437886183)

[5.2.3 Типы вопросов 9](#_Toc437886184)

[5.3 Прохождение теста 10](#_Toc437886185)

[5.4 Алгоритм работы сервера 10](#_Toc437886186)

[5.5 Программная реализация 10](#_Toc437886187)

[5.5.1 Реализация Python 10](#_Toc437886188)

[5.5.2 Среды разработки 11](#_Toc437886189)

[5.6 Ход работы 11](#_Toc437886190)

[5.6.1 Проблемы и решения 12](#_Toc437886191)

[6 Итог исследования 12](#_Toc437886192)

[7 Выводы 12](#_Toc437886193)

[8 Список литературы 13](#_Toc437886194)

# Введение

Для проведения контроля знаний учащихся применяют множество различных методов:

* Устный ответ
* Письменная работа
* Тестовые задания

Каждый вариант оценивания знаний подходит для разных ситуаций.

Тестирование сейчас находит, в частности в школе, всё большее применение, в особенности из-за введения ЕГЭ. Тесты позволяют быстро получить результат, при относительно малых затратах времени на проверку. Применяют как бумажные, так и электронные тесты.

Электронное тестирование, в отличие от бумажного, имеет ряд преимуществ:

* Позволяют, почти сразу получить результат, не вынуждая учителя самому проверять ответы учащихся.
* На выходе выдаётся информация, готовая к анализу. Например, в виде таблицы Excel.
* Исключают ошибку преподавателя при проверке.

Для проведения электронного тестирования удобно использовать планшеты. Они мобильны и удобны для ученика.

На сегодняшний момент различают следующие виды тестов:

* Множественный выбор - испытуемому необходимо выбрать один или несколько правильных ответов из приведенного списка
* Альтернативный выбор - испытуемый должен ответить «да» или «нет»
* Установление соответствия - испытуемому предлагается установить соответствие элементов двух списков
* Установление последовательности - испытуемый должен расположить элементы списка в определенной последовательности
* Свободное изложение - испытуемый должен самостоятельно сформулировать ответ; никакие ограничения на них в задании не накладываются
* Дополнение - испытуемый должен сформулировать ответы с учетом предусмотренных в задании ограничений (например, дополнить предложение)

# Цели и задачи исследования

Была поставлена задача разработать среду для проведения электронного тестирования с помощью Android устройств и web интерфейса. В неё входит набор инструментов для учителя и тестируемого:

Учитель:

* + Редактор тестов
  + Просмотр результатов тестов

Ученик:

* + Приложение Android и web сайт для прохождения тестирования

И были выдвинуты следующие критерии:

* + Пользователи делятся на учителя и учащихся. Учитель создаёт тест, а ученики проходят его и отправляют данные для проверки на сервер.
  + Данное приложение может использоваться на уроках по различным предметам для оценки учащихся.
  + Данное приложение должна обеспечивать учителей:
    1. Средством для создания и редактированием тестов.
    2. Возможностью раздать его ученикам через интернет.

А ученик, в свою очередь, имеет возможность:

* 1. Открыть тест
  2. Пройти его
  3. Отправить свои ответы на сервер для проверки

# 3. Обзор аналогов. Актуальность исследования

Существует множество систем для электронного тестирования. Большинство из них предназначено для работы в браузере (Moodle, WebSoft CourseLab), есть системы для работы за компьютером на Windows, Mac, Linux (MyTest).

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Продукт | Поддержка мобильных устройств | Медиа контент | WEB интерфейс | Стационарная версия |
| MyTest | Нет | Да | Нет | Да |
| Moodle | Нет.  Только web интерфейс | Да | Да | Нет |
| WebSoft CourseLab | Нет.  Только web интерфейс | Да | Да | Нет |
| LIT Test  (Зайдель) | Да(Android) | Нет | Нет | Да |
| Мой проект | ДА(android) | Да | Да | Нет |

Данные аналоги здесь представлены, так как все они так или иначе представляют собой именно систему тестирования, которая имеет подразделение на тестируемого (ученик) и тестирующего (учитель).

## 3.1. MyTest

“MyTest - система программ для создания и проведения компьютерного тестирования, сбора и анализа их результатов”.

“Программа MyTestX работает с десятью типами заданий: одиночный выбор, множественный выбор, установление порядка следования, установление соответствия, указание истинности или ложности утверждений, ручной ввод числа, ручной ввод текста, выбор места на изображении, перестановка букв, заполнение пропусков”.

“Программа состоит из трех модулей: Модуль тестирования (MyTestStudent), Редактор тестов (MyTestEditor) и Журнал тестирования (MyTestServer)”.

Таким образом видно, что данный аналог содержит большинство особенностей, которые предполагаются в нашем приложении, но это только PC программа.

## 3.2 Moodle

Для того, чтобы было возможно пользоваться данным приложением, необходимо быть зарегистрированным в системе Moodle.

## 3.3 Websoft CoursLine

“CourseLab – это мощное средство для создания интерактивных учебных материалов (электронных курсов), предназначенных для использования в сети Интернет, в системах дистанционного обучения, на компакт-диске или любом другом носителе”.

Ключевые особенности: Медиа контент, drag & drop, шаблоны оформления, интеграция с Microsoft Office.

## 3.4 LIT Test

Типы заданий: одиночный выбор, множественный выбор, ответ в краткой форме, ответ в виде «да/нет». Автоматическая проверка тестов.

# Анализ предметной области

## Что такое тест

Тест- это измерительная методика оценки, основанная на стандартизированном наборе заданий и формализованных правилах анализа ответов, дающая в результате объективированную оценку испытуемого на определённой количественной шкале, указывающей на выраженность измеряемого свойства или вероятностное отнесение к определённой, заранее заданной диагностической категории.

Примером такого теста может быть часть А и Б в ЕГЭ.

## Виды тестов

Тесты можно классифицировать по различным признакам:

* По целям — информационные, диагностические, обучающие, мотивационные, аттестационные;
* По процедуре создания — стандартизованные, не стандартизованные;
* По способу формирования заданий — детерминированные, стохастические, динамические;
* По технологии проведения — бумажные, в том числе бумажные с использованием оптического распознавания, натурные, с использованием специальной аппаратуры, компьютерные;
* По форме заданий — закрытого типа, открытого типа, установление соответствия, упорядочивание последовательности;
* По наличию обратной связи — традиционные и адаптивные

В образовании в основном используются традиционные тесты, то есть те которые содержат список вопросов, каждый из которых оценивается в определенное количество баллов. Результат такого тестирования зависит от количества вопросов, на которые был дан правильный ответ. В традиционном тесте оценивание происходит с одинаковыми правилами оценивания ответов для всех испытуемых. Главная цель применения традиционных тестов - установить уровень знаний.

Также можно разделить тесты на:

* Образовательные
* Профессиональные
* Медицинские
* Психологические

Образовательные тесты – это «предметно-ориентированные» тесты знаний. В нём задания относятся к определённым академическим предметам-математике, языку, истории, физике и т.д. Этот вид тестов проводят обычно педагоги-предметники. В результате такого теста испытуемый получает свою оценку или балл за выполненные задания.

Профессиональные тесты – это тесты профессиональных знаний, в которых большую роль играют «кейс-задания» - задачи, описывающие практическое производственные ситуации и предлагающие выбор способа действия в этой ситуации.

Психологические тесты – это тесты, ориентированные на установку и измерение индивидуально-психологических отличий испытуемого. К этому виду относятся такие тесты как: IQ, Тест Сонди, Тест Роршаха.

Созданная мною инструментальная среда направлена на работу с образовательными тестами, но также можно также5 работать и с производственными.

## Виды тестовых заданий

Вот список основных видов заданий

1. Контрольные списки суждений или закрытые тесты с ответами «да-нет»
2. Списки суждений с дробной шкалой согласия, например «да-нет-не знаю»
3. Закрытые вопросы, предполагающие качественно различных ответов, из которые надо выбрать только один(multiple-choice).
4. Закрытые тест-вопросы, допускающие выбор нескольких качественно-различных готовых ответов. Это вопросы со множественным ответом (multiple-response).
5. Открытые тест-вопросы с кратким свободным ответом (в виде числа, в виде одного слова). Некоторой визуальной модификацией является суждения с пропусками (gap-filling).
6. Открытые тест-вопросы с развёрнутым ответом. Этот тип используется в части С ЕГЭ.
7. Задания на ранжирование. Испытуемому предлагается выстроить определённое множество элементов в определённой последовательности.
8. Задания на соответствие: в этом случае испытуемый должен сравнить два множества объектов и установить попарные соответствия для всех элементов.

В ЕГЭ используются все из представленных выше типов заданий, но в ответе они сводятся к трём типам: выбор одного варианта из четырёх (часть А), запись краткого текстового ответа (часть Б), развёрнутый ответ (часть С).

# Решение

## 5.0 Архитектура системы

[3]

Электронное тестирование, в отличие от остальных бумажного, предполагает использование компьютерных систем для сбора и анализа результатов.

Система для электронной оценке знаний должна состоять из нескольких обязательных модулей:

* Инструмент для подготовки тестовых заданий (конструктор тестов, только web)
* Тест-плейер(Android & web)
* А также инструмент для проведения тестирования (веб-приложение, приложение-сервер и тд).

## 5.1 Хранение данных

Все данные, такие как логины, пароли, тесты и тд хранятся в базе данных MySQL. БД MySQL одна из самых распространенных баз данных. Был выбор между 3-мя БД:PostgreSQL, SQLite, MySQL. Поводы выбора MySQL:

1. имелся опыт работы в данной БД;
2. SQLite имеет меньший набор инструментов для управления этой БД;
3. MySQL наиболее докуметирована

Рассмотрим систему тестирования данного приложения:

Сайт соединяется с сервером напрямую. В свою очередь Android приложение для взаимодействия с сервером использует API сайта. Из API Android приложение берет информацию о тестах, учениках, вопросах и тд. Для того, чтобы отправить какой-то запрос на сервер используются POST запросы. На сервере происходит обработка запроса и возвращение результата обработки обратно на устройство. Результаты записываются в базу данных.

### 5.1.1 Хранение тестов

XML теста

<?xml version="1.0"?>  
<Test><choice\_count>2</choice\_count>

<edit\_date>2015-08-19T15:42:21</edit\_date>

<id type="integer">85</id>

<name>test</name>

<pub\_date>20150819T15:42:21</pub\_date>

<resource\_uri>/api/v1/tests/85/</resource\_uri>

<teacher type="integer">1</teacher>

<theme>testing</theme>

<visibility type="boolean">True</visibility>

</Test>

### Хранение вопросов

<?xml version="1.0"?>

<Question>

<edit\_date>2015-08-28T02:32:23</edit\_date>

<id type="integer">100</id>

<pub\_date>2015-08-28T02:32:23</pub\_date>

<question\_count type="integer">1</question\_count>

<question\_text>question</question\_text>

<resource\_uri>/api/v1/questions/100/</resource\_uri>

<test type="integer">94</test>

<test1>/api/v1/tests/94/</test1>

<theme>Theme</theme>

<visibility type="boolean">True</visibility>

</Question>

## 5.2. Структура данных

### Взаимодействие классов

На приведенной диаграмме[2] визуализировано взаимодействие между классами.

### Типы тестов

* Обычный
* На время
* Со случайными вопросами из списка

При создании теста учитель сначала вводит его название, тему, для какого он класса, а также название предмета тестирования. Затем он переходит к заполнению теста вопросами. При создании вопроса тестирующий может добавить недостающие варианты ответов, а также добавить картинку к вопросу.

Правильных ответов может быть от 0 до количества вариантов ответов.

#### Обычный тест

У теста фиксированные вопросы.

#### Тест на время

При создании теста учитель указывает время на прохождение теста в секундах. Ученику, для начала прохождения теста, необходимо нажать на соответствующую кнопку, после чего ему будет доступно выполнение теста.

После истечения времени, тест будет закрыт для редактирования.

#### Тест со случайными вопросами

При создании теста, учитель указывает количество вопросов, которые должны быть показано ученикам. После этого учитель заполняет тест вопросами. Ученикам будут предоставлены случайные вопросы с указанным количеством вопросов из данного теста.

### Типы вопросов

В нашем приложении существует два типа вопросов: с картинкой и без.

При создании вопроса к тесту учитель может выбрать изображение к вопросу, загрузив его с устройства. Также при создании вопроса учитель может добавить варианты ответа в необходимом количестве.

## Прохождение теста

Перед учеником не открывается тест до тех пор, пока он не нажмет соответствующую кнопку для начала тестирования. Если же он начал тестирование и не закончил его выполнение, то ученику будет засчитано ноль правильных ответов. При прохождении теста тестируемый может выбрать несколько вариантов ответа. По окончанию процесса ученик нажимает на кнопку отправления результатов, далее этот тест отмечается как пройденный для данного ученика, и он не может пройти его снова.

## Алгоритм работы сервера

Связь сервера с устройством осуществляется через GET/POST запросы.

1.Сайт: использует GET запросы для получения, POST – для отправления данных на сервер.

2.Android приложение: происходит обмен данными через POST запросы. На сервер отправляется запрос и возвращается ответ в виде строки. Сначала использовался GET метод с помощью API сайта, но для ускорения работы предпочтение было отдано POST запросам.

Например, запрос , состоящий из пары(ключ, значение) для определения доступных тестов ученика(available\_tests, student\_id). Результат запроса: (“test\_id1, test\_id2….”).

## Программная реализация

### Реализация Python

Была поставлена задача не только разработать систему тестирования, но также и изучить программирование, поэтому варианты в виде CMS(Content Management System) были отброшены. Был выбор между 4-мя языками веб программирования: PHP, C#, Python, Java. Python был выбран по следующим причинам:

1)относительная новая платформа Django постепенно внедрялась в различные веб проекты;

2)Python имеет отличный от C# и Java синтаксис, а также Python является скриптовым некомпилируемым языком – все это повлияло на выбор ЯП, так как это еще не было пройдено, хотелось узнать новое в программировании.

#### API сайта

API (Application Programming Interface) — это интерфейс программирования, интерфейс создания приложений. Если говорить более понятным языком, то API — это готовый код для упрощения жизни программисту. API создавался для того, чтобы программист реально мог облегчить задачу написания того или иного приложения благодаря использованию готового кода (например, функций). Для работы с API в Django я использовал библиотеку TastyPie. С помощью этой библиотеки можно ограничить выводимую информацию по таким параметрам, как:

1. Класс для вывода объектов(например, Teacher)
2. Метод(например, GET)
3. Можно изменить вывод определенного параметра объекта

Затем нужно в определенном файле подключить библиотеку, создать ее объект, указать, какие API были созданы. На выходе получим XML с указанными параметрами. Данные XML можно открыть в браузере или загрузить на устройство для дальнейшей обработки.

Пример создания API для вывода списка учителей:

class MyModelResource(ModelResource):

user = fields.ForeignKey(UserResource, 'user', null=True, blank=True)

school = fields.ForeignKey(SchoolResource, 'school', null=True, blank=True)

class Meta:

queryset = Teacher.objects.all()

resource\_name = 'teachers'

allowed\_methods = ['get']

serializer = Serializer(formats=['json', 'jsonp', 'xml', 'yaml', 'html', 'plist'])

Создаем объект API:

from api.resources import MyModelResource

from tastypie.api import Api

v1\_api = Api(api\_name='v1')

v1\_api.register(MyModelResource())

#### Работа с API на Android

Работа с API происходит в асинхронных классах, которые выполняются в фоновом режиме , не тормозя саму программу. }

### Среды разработки

Для программирования на Python была выбрана среда разработки JetBrains PyCharm. Также рассматривалась среда Eclipse. Но она не была выбрана, потому что не имеет автодополнения выражений(основной критерий), удобным дизайном в отличие от первой IDE.

Android приложение написано в среде JetBrains IntellijIdea. Данная среда схожа с Android Studio, так как вторая основана на программном обеспечении первой. Главной причиной выбора Intllij Idea является ее большая стабильность по сравнению с Android Studio. Была рассмотрена среда Eclipse, но не использована так как не обладает некоторым функционалом, таким как автодополнением выражений, удобным дизайном в отличие от аналогов.

## Ход работы

Изначально связь между Adnroid приложением и сервером предполагалась за счет API. Использование API довольно тяжело реализовывалось в разработке под смартфон. Поэтому позже, когда появилась необходимость аутентифицировать пользователей, на смену API пришли POST запросы. Они проще и в использовании.

На финальной стадии разработки начали использоваться ajax технологии, но широкого применения они получить не успели.

### Проблемы и решения

1. Прежде чем записать изменения в базу данных Django генерирует файл “migrations.py «номер изменения»” с изменениями. Он генериуется командой “makemigrations”. Затем командой “migrate” изменения записываются окончательно в БД. После непродолжительного использования данной системы возникают проблемы, а именно: при попытке отправить изменения на сервер, Django сообщает, что, допустим, такая таблица в БД уже есть и создать ее не получится, таким образом отменяются все последние изменения. Чтобы исправить данную ошибку, приходилось открывать файл “migrations” с соответствующим номером изменения, находить часть кода, которая не давала успешно завершиться отправке данных и либо удалять ее, либо изменять. На начальном этапе программирования данная неполадка потребовала около трех дней разработки, чтобы научиться обходить эту ошибку.
2. PyCharm хоть и является одной из лучших IDE для веб разработки, данное приложение имеет большой недостаток в отладчике. Во-первых, он работает только при условии запуска сайта так, чтобы к нему имел доступ только компьютер, на котором совершен запуск. Соответственно отладка Android приложения была возможна только благодаря ручному выводу определенного значения в командную строку/терминал. Также отладчик PyCharm не справляется с Javascript, чего сильно не хватало. Система отладки в данной ситуации схожа с отладкой приложения для смартфона – необходимое значение переменной выводилось на экран командой “alert”.
3. В Android приложении подключение к серверу невозможно без использования AsyncTask (синхронных процессов). Потребовалось много времени, чтобы научиться ими пользоваться.

# Итог исследования

В результате работы была разработана система для проведения электронного тестирования на Android устройствах и в web интерфейсе.

Данная система имеет инструменты для тестируемого и тестирующего:

* + Ученик – Приложение Android и web сайт для прохождения тестирования
  + Учитель – редактор тестов, выставление оценки, просмотр тестов.

Пример работы сайта:

Учитель

# Выводы

На данный момент тема электронного тестирования очень актуальна. И вскоре подобные системы тестирования будут сильно распространены.

Перспективы разработки проекта:

* Фильтрация реальных пользователей
* Приближение функционала Android приложения к функционалу сайта
* Реализовать еще несколько видов теста
* Большее применение ajax технологий
* Добавить функцию массового добавления изображений
* Добавить функцию выбора картинки в качестве ответа
* Дополнить возможности личных настроек

# Список литературы

[habrahabr.ru](file:///C:\Users\Леново\Documents\habrahabr.ru) – множество статей по android/django разработке

[startandroid.ru](file:///C:\Users\Леново\Documents\startandroid.ru) – уроки по android разработке

[developer.alexanderklimov.ru](file:///C:\Users\Леново\Documents\developer.alexanderklimov.ru) – android разработка

[djbook.ru](file:///C:\Users\Леново\Documents\djbook.ru) – django учебник

[django-tastypie.readthedocs.org/](http://django-tastypie.readthedocs.org/) - учебник по tastypie библиотеке

[stackoverflow.com](http://stackoverflow.com/) - англоязычный форум

[ru.stackoverflow.com](http://stackoverflow.com/) - русскоязычный форум

[www.cyberforum.ru](http://www.cyberforum.ru/) - русскоязычный форум

[toster.ru](file:///C:\Users\Леново\Documents\toster.ru) – русскоязычный форум