

ОЛИМПИАДА ШКОЛЬНИКОВ «ШАГ В БУДУЩЕЕ»

НАУЧНО-ОБРАЗОВАТЕЛЬНОЕ СОРЕВНОВАНИЕ «ШАГ В БУДУЩЕЕ, МОСКВА»

регистрационный номер

Информатика и системы Управления

название факультета

Программное обеспечение ЭВМ и информационные технологии (ИУ7)

название кафедры

Система учета выполнения лабораторных работ

название работы

Автор:

Никутьшин Павел Алексеевич

фамилия, имя, отчество

ГБОУ «Инженерная школа № 1581», 10 И

наименование учебного заведения, класс

Научный руководитель:

Николаева Ольга Юрьевна

фамилия, имя, отчество

ГБОУ «Инженерная школа № 1581»

место работы

учитель

звание, должность

подпись научного руководителя

Москва 2021

Аннотация

В процессе работы преподаватели сталкиваются с необходимостью учета выполнения студентами лабораторных работ по различным дисциплинам. Оценка складывается из двух составляющих: качества работы и сроков выполнения.

Целью данной работы было реализовать систему, которая будет отслеживать сроки и качество выполнения работ.

В системе предусматривается возможность вести учет по нескольким дисциплинам

По одному предмету может быть несколько лабораторных работ, в свою очередь каждую лабораторную работу может принимать несколько преподавателей. Также необходимо учесть, что у преподавателя могут быть помощники: студенты магистратуры, которые тоже имеют право принимать и оценивать некоторые работы.

Каждая лабораторная работа имеет свой срок выполнения и максимальный балл, который возможно за нее получить. Преподаватель оценивает качество работы. Система должна откорректировать оценку в соответствии со сроком выполнения: то есть надо учесть скорость снижения балла в зависимости от сроков. При этом, «стоимость» работы не может быть отрицательной.

Для достижения цели работы было разработано веб-приложение на базе фреймворка Django. Для хранения данных пользователей системы используется объектно-реляционная система управления базами данных SQL-lite

Содержание

1. Аннотация	2
2. Содержание	3
3. Введение	4
a. Актуальность работы	4
b. Цель работы	4
c. Задачи и этапы работы	4
d. Методы исследования	4
4. Основная часть	5
a. Анализ предметной области	5
b. Назначение	5
c. Требования к системе	5
d. Средства разработки системы	6
e. Архитектура	7
f. Пользовательский интерфейс	8
5. Заключение	9
6. Список литературы	10
7. Приложение	11

Введение

Актуальность работы

Преподаватели часто сталкиваются с проблемой учета выполнения учебного плана. Это происходит из-за большого количества разрозненной информации, включающую, которую необходимо все время держать в голове. Ведение «плоских» таблиц, типа Excel, конечно, в определенной степени решают эту задачу, но при этом сильно теряется удобство ведения и наглядность.

Анализ ситуации привел к необходимости создания программного продукта, которой будет помогать учитывать выполнение учебного плана

Цель работы

Разработка системы, позволяющей вести учет выполнения лабораторных работ.

Задачи и этапы работы

Задачи, которые необходимо решить для выполнения поставленной цели:

1. Изучить и проанализировать проблемы, возникающие при работе преподавателя в части учета выполнения учебного плана
2. Определить функциональные требования к системе;
3. Выбрать средства разработки;
4. Разработать удобный пользовательский интерфейс
5. Реализовать приложение (написать код, протестировать и отладить);

Методы исследования

Для исследования задачи проводилось интервьюирование заинтересованных лиц, непосредственно участвующих в процессе приема лабораторных работ.

Основная часть

Анализ предметной области

На сегодняшний день не существует готовых систем, «заточенных» для решения поставленной задачи.

Есть возможность ведения учета с помощью «плоских» таблиц, но они имеют не полный функционал и не удобны для ведения и проведения анализа данных

Назначение

Приложение предназначено для учета качества и сроков выполнения лабораторных работ.

Требования к системе

Система должна обеспечивать возможность выполнения следующих функций:

1. Обеспечение авторизованного доступа пользователя в систему и предоставления ему прав в зависимости от его роли
 - а. Права администратора – для ввода первичных данных
 - б. Права пользователя – для регистрации оперативных данных
2. Ввод и редактирование данных для учета: по дисциплинам, преподавателям, ученикам, лабораторным работам.
3. Предоставление информации по сданным\не сданным лабораторным работам, в разрезе учеников, дисциплин, преподавателей.

Средства разработки системы

Язык программирования - Python 3.7. Язык удобен, имеет большое количество библиотек.

В качестве СУБД выбрана SQL-Lite. При этом для коммуникации использовался объектный преобразователь кода (ORM) - стандартная библиотека, автоматизирующая перенос из реляционной базы данных, в объекты, которые используются в коде приложения.

Веб-фреймворк – Django

Веб интерфейс выбран в связи с тем, что сейчас очень часто приходится работать и учиться удаленно. Это позволит сделать процесс учета более универсальным

Архитектура



Пользовательский интерфейс.

В системе предусмотрено использования веб-интерфейса. Для его создания использовался фреймворк Django

Вход в систему авторизованный. В зависимости от роли пользователя, отличается и его интерфейс.

В рамках проекта разработано несколько страниц: для ввода данных, для предоставления информации в определенном виде.

Заключение

Приложение предоставляет возможность удобного и наглядного учета и анализа выполнения учащимися учебного плана.

В работе заложена возможность легкого и быстрого расширения функционала в зависимости от нужд пользователей.

Список литературы

1. <https://tutorial.djangogirls.org/ru/> - Учебник Django
2. <https://www.djangoproject.com> - Документация Django

Приложения

Объекты представления данных:

```
class People(models.Model):
    FirstName = models.CharField(max_length=20)
    LastName = models.CharField(max_length=20)
    Otchestvo = models.CharField(max_length=20)
    POSITIONS = (('rank0', 'Студент'),('rank1', 'Магистр'),('rank2', 'Ассистент'), ('rank3', 'Ст.
преподаватель'))
    Position = models.CharField(max_length=20, choices=POSITIONS,default='rank0')
    Email = models.EmailField(max_length=20)
```

```
class Group (models.Model):
    NUMBERS = (('1', '1'), ('2', '2'), ('3', '3'), ('4', '4'), ('5', '5'), ('6', '6'))
    number = models.CharField(max_length=20, choices=NUMBERS,default='1')
    Kurator = models.ForeignKey('People', on_delete=models.SET_NULL, null=True)
    Year = models.CharField(max_length=20)
```

```
class Students (models.Model):
    Name = models.ForeignKey('People', on_delete=models.SET_NULL, null=True)
    Group = models.ForeignKey('Group', on_delete=models.SET_NULL, null=True)
    Zachetka = models.CharField(max_length=20)
    Year = models.CharField(max_length=20)
```

```
class DisciplineName(models.Model):
    Name = models.CharField(max_length=20)
```

```
class Discipline(models.Model):
    Lector = models.ForeignKey('People', on_delete=models.SET_NULL, null=True)
```

```
Name = models.ForeignKey('DisciplineName', on_delete=models.SET_NULL, null=True)
```

```
NUMBERS = (('1', '1'), ('2', '2'))
```

```
number = models.CharField(max_length=20, choices=NUMBERS, default='1')
```

```
class Laboratory(models.Model):
```

```
    Discip = models.ForeignKey('Discipline', on_delete=models.SET_NULL, null=True)
```

```
    NUMBERS = (('1', '1'), ('2', '2'), ('3', '3'), ('4', '4'), ('5', '5'), ('6', '6'), ('7', '7'), ('8', '8'), ('9', '9'), ('10', '10'), ('11', '11'), ('12', '12'), ('13', '13'), ('14', '14'), ('15', '15'))
```

```
    number = models.CharField(max_length=20, choices=NUMBERS, default='1')
```

```
    Name = models.CharField(max_length=40)
```

```
    Date = models.CharField(max_length=20)
```

```
    MARKS = (('1', '1'), ('2', '2'), ('3', '3'), ('4', '4'), ('5', '5'), ('6', '6'))
```

```
    MaxMark = models.CharField(max_length=20, choices=MARKS, default='1')
```

```
class Completing(models.Model):
```

```
    Student = models.ForeignKey('Students', on_delete=models.SET_NULL, null=True)
```

```
    Lab = models.ForeignKey('Laboratory', on_delete=models.SET_NULL, null=True)
```

```
    Teacher = models.ForeignKey('People', on_delete=models.SET_NULL, null=True)
```

```
    PlanningDate = models.CharField(max_length=20)
```

```
    RealDate = models.CharField(max_length=20)
```

```
    STATUSES = (('Y', 'Зачет'), ('N', 'Незачет'))
```

```
    Status = models.CharField(max_length=20, choices=STATUSES, default='1')
```

```
    Mark = models.CharField(max_length=20)
```