|  |
| --- |
| Национальный Исследовательский Университет  Высшая Школа Экономики |
| Групповой проект |
|  |

по программированию

Develop a search engine for storage and processing of student works

Выполнили: Желандовская Е.И., Кашицына О.А., Лебедев Г.Д.

Группа: ББИ141, факультет Бизнеса и менеджмента

Преподаватель: Ефремов С.Г.

**Аннотация**

Программа для хранения и поиска студенческих работ. Для того, чтобы воспользоваться данной системой, пользователю необходимо пройти регистрацию. Зарегистрированный пользователь может найти необходимую работу, введя в поле поиска необходимое слово, фразу или предложение. Пользователь может посмотреть описание найденных работ и скачать подходящую работу. Также пользователю доступна функция добавления собственных файлов, в процессе которой он может добавить хештеги для более быстрого поиска. Ко всему этому, пользователь может проверить любой файл на плагиат при помощи API сайта text.ru. Все файлы в системе типа doc и docx.

**Адрес главного репозитория.**

[https://github.com/LebGrigori/StudentWorksSearch.git](https://vk.com/away.php?to=https%3A%2F%2Fgithub.com%2FLebGrigori%2FStudentWorksSearch.git)

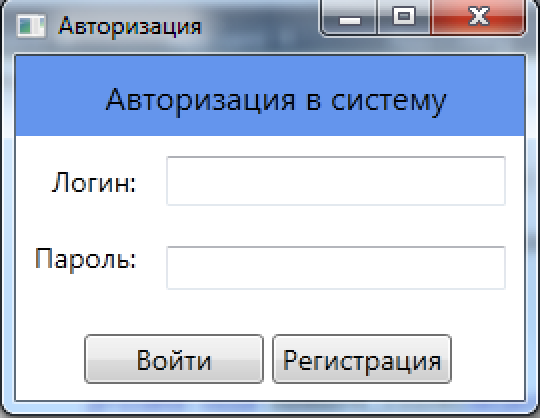
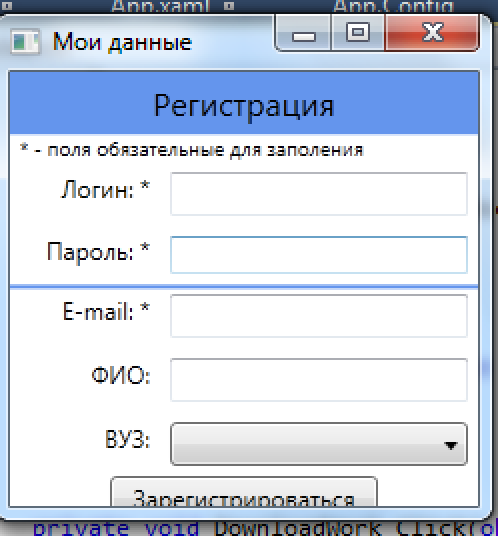
**Члены команды и их роли.**

* Лебедев Григорий 141: Проверка плагиата при помощи API, логика приложения.
* Кашицына Оксана 141: Создание интерфейса, разработка базы данных SQL.
* Желандовская Екатерина 141: Подключение базы данных, индексирование.

**Классы.**

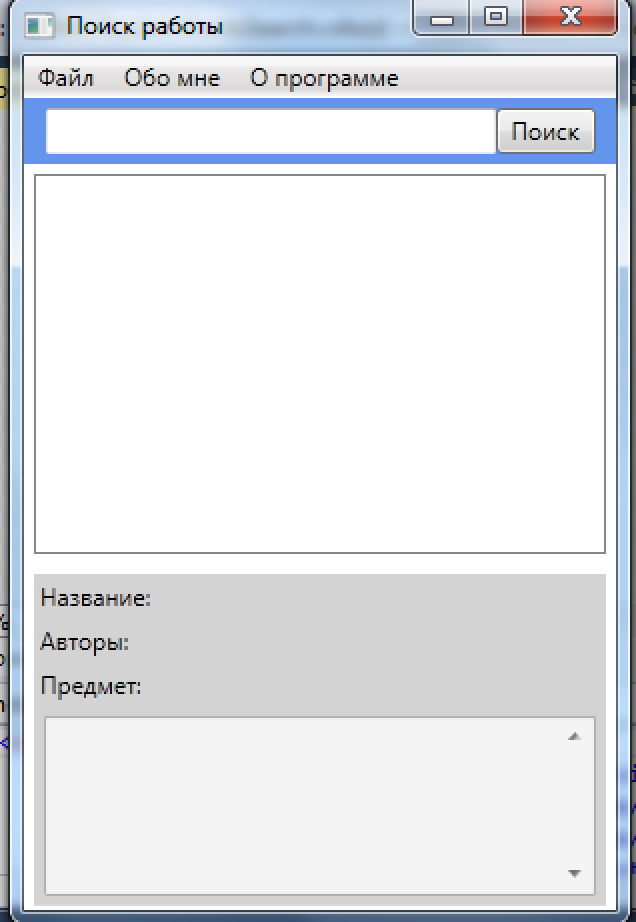
APIEngine – отвечает за проверку на плагиат, связываясь с сервером text.ru, и возвращает процент плагиата в переданной ему строке.  
DBEngine – возвращение двух коллекций для подгрузки их к соответствующим элементам в форме (вузы и дисциплины).  
FileEngine – считывает текст с документа doc или docx в строку, при добавлении файла создается работа и файл в бд.  
UserEngine – осуществление входа, регистрации, хэширования пароля и получения всей информации о пользователе.  
FileToIndex – сущность, которая подвергается индексированию.   
LuceneEngine – отвечает за индексацию и поиск.  
Repository – хранение данных, к которым нужен доступ сквозь все классы, например, сущность пользователя, который авторизовался.  
UserData – информация о пользователе.

**Интерфейс.**

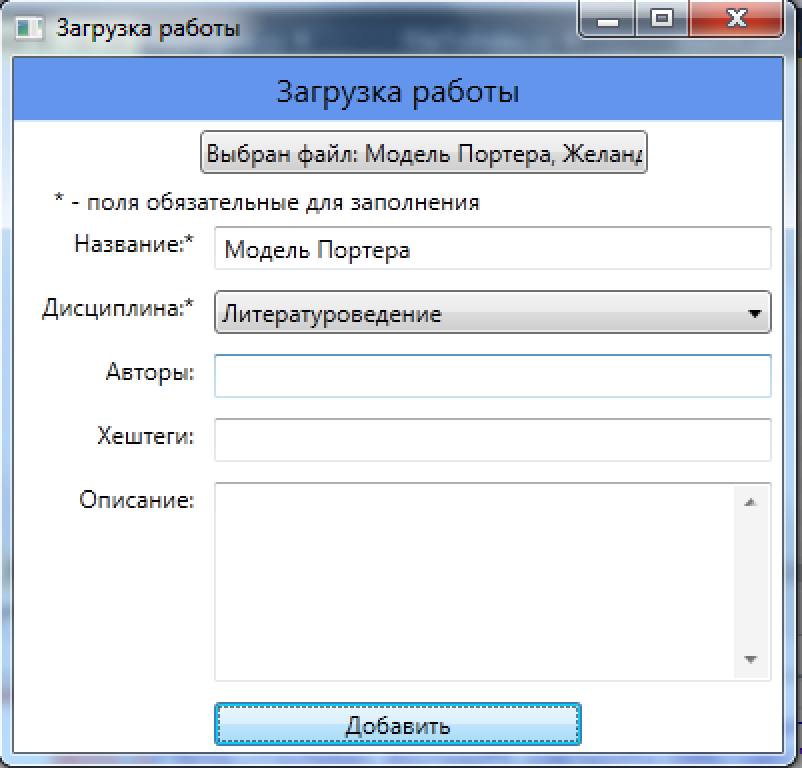
 

Первое окно, которое возникает перед пользователем. Необходимо ввести логин и пароль для входа в систему, или же пройти регистрацию.

При прохождении регистрации пользователю обязательно нужно ввести логин, пароль и почту (not null), по желанию он может ввести ФИО и выбрать ВУЗ из выпадающего списка.



Окно, открывающееся перед пользователем после авторизации. В строку поиска вбивается слово/фраза, после чего в листбоксе отображаются найденные файлы. Выбрав файл, можно узнать краткую информацию, а именно: название, авторов, предметную область и описание. А также можно его скачать (Файл => Скачать работу).



Пользователь может загрузить работу (Файл => Добавить работу). Обязательными полями являются название и дисциплина. Также можно добавить авторов, хештеги и описание.