Пояснительная записка к контрольному домашнему заданию по дисциплине «программирование».

Образовательная программа «Бизнес Информатика»

Бакалавриат, 1-й курс.

Рузиев Жамшид ББИ166

1. Тема: Разработка информационной базы сотрудников НИУ ВШЭ (website, python)
2. Адрес репозитория: https://github.com/ruziev/HSE-Staff
3. Одиночный проект.
4. Аннотация:

Сайт/приложение позволяет искать сотрудников НИУ ВШЭ, получать их контактные телефонные номера, электронные почты и адреса постоянного пребывания в НИУ ВШЭ. Есть возможность пополнять базу, обновлять, редактировать данные сотрудника. Разработано удобное мобильное приложение для iOS, синхронизирующее свою работу с сайтом.

1. Используемые технологии и библиотеки:

Проект состоит их трех частей: веб-скрапер, сайт и iOS приложение.

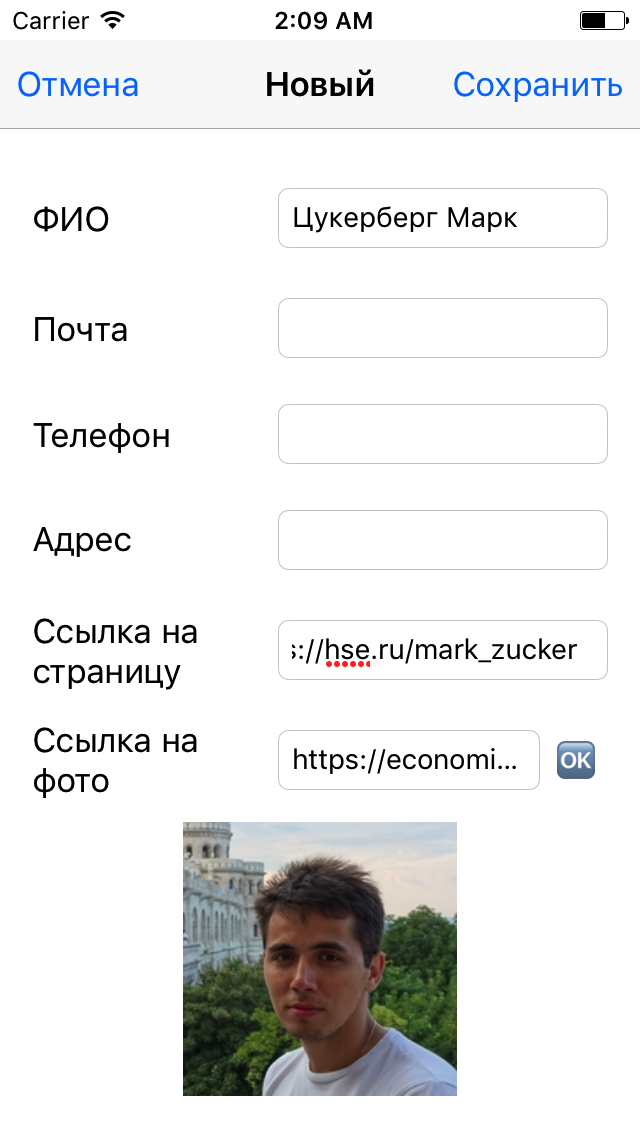
Веб-скрапер:

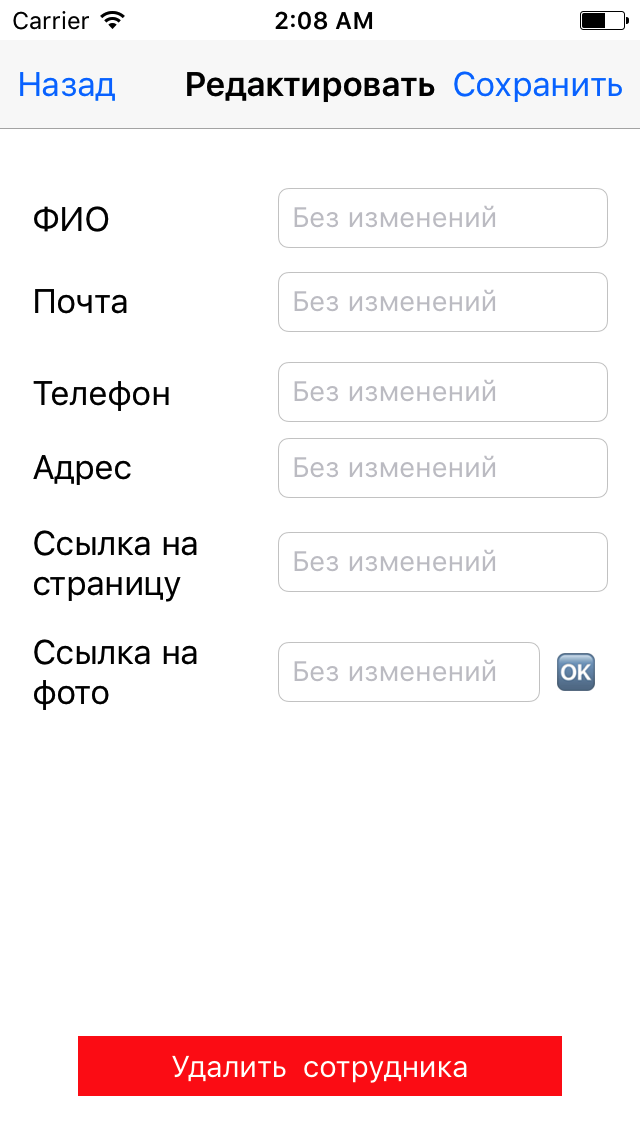
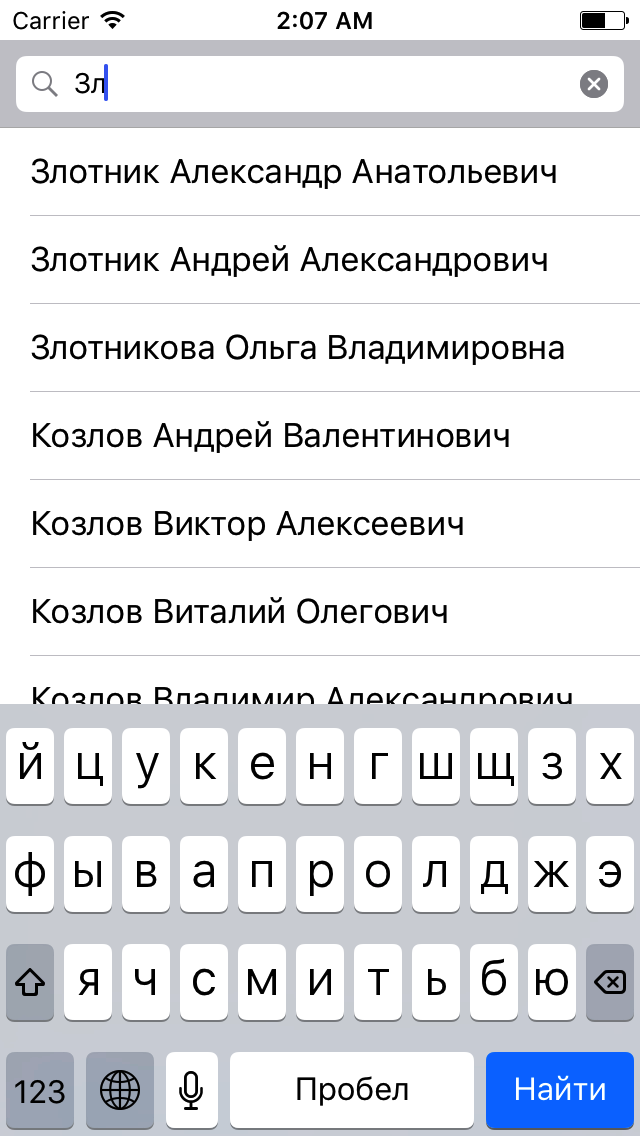
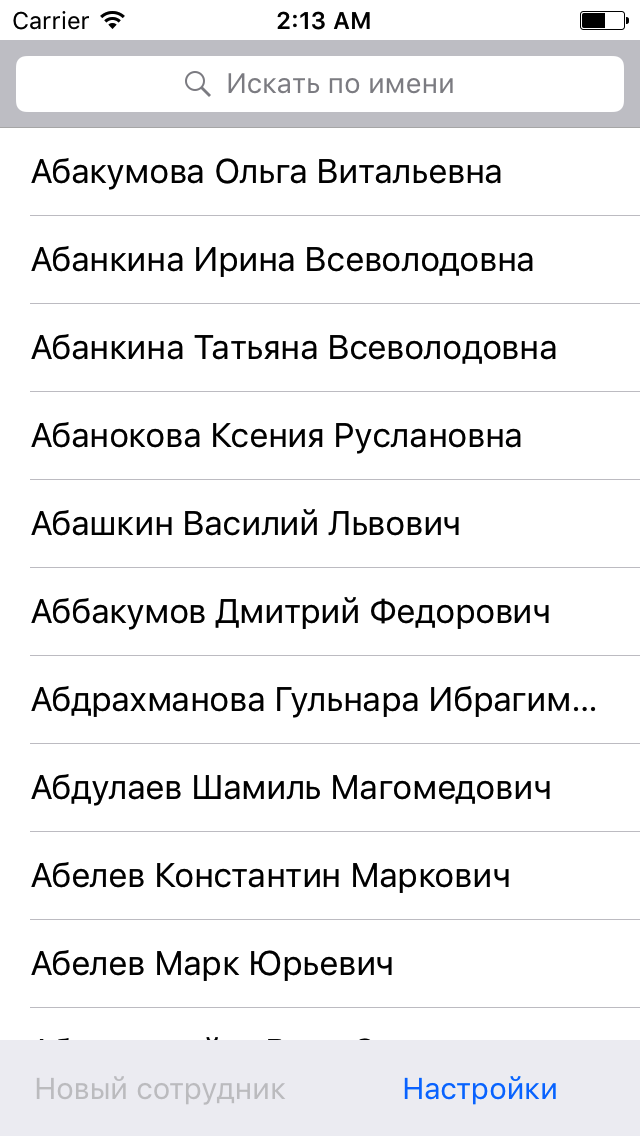
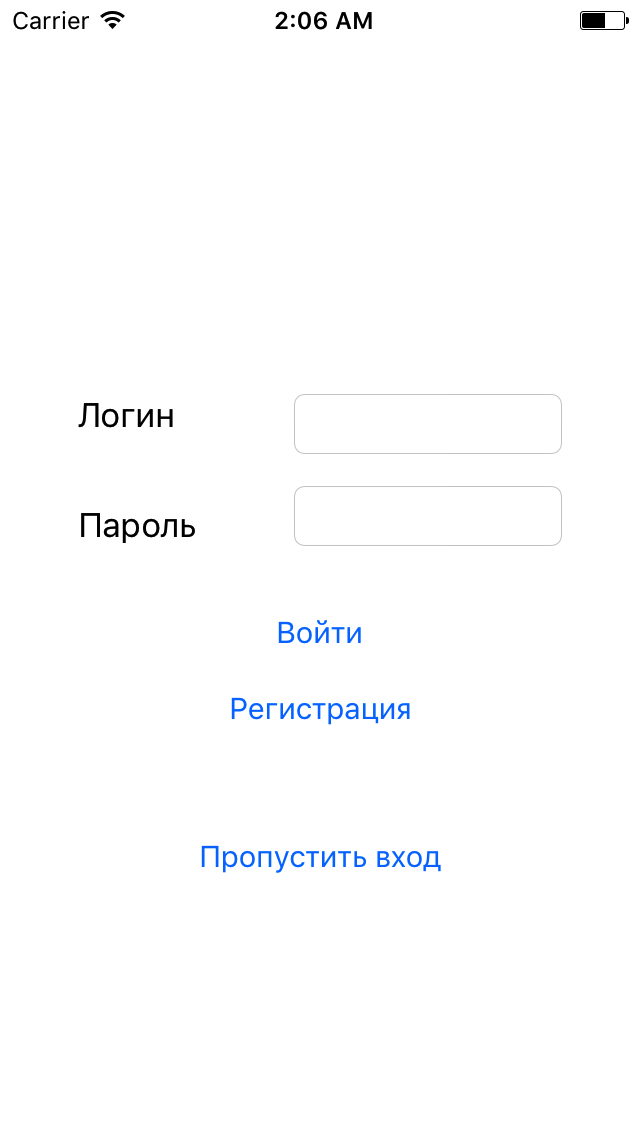
IPython Notebook, написан на python. Использует сторонние библиотеки requests (работа с веб-запросами), BeautifulSoup4 (HTML парсер на принципе построения DOM), ponyORM (удобный минималистичный Object Relational Mapper, использующий синтаксис python list comprehensions (~LINQ в С#)). Из встроенных pickle (сериализация), re (regular expessions). Принцип работы простой: Ходим по [hse.ru](http://hse.ru) (поиск сотрудников, алфавитный). Отправляем запросы по буквам алфавита - начало фамилии и получаем ссылки на личные страницы сотрудников. Затем для каждого сотрудника парсим необходимые данные для полей в объявленной классе модели. На этом этапе возникли некоторые трудности с получением email, однако проблема была устранена регулярными выражениями и прочими очевидными приемами. Полученную базу записываем в базу данных (SQLite), которую затем использует сайт.

Website:

Используется связка Python/Flask/PonyORM. Является сайтом, но, в основном, предоставляет открытый REST API (с авторизацией), которое используется приложением. Соблюден принцип MVC; API предоставляет CRUD интерфейс. Помимо Flask, PonyORM используются встроенные библиотеки hashlib (для генерации MD5 суммы пароля), json (json сериализация для передачи API данных). Данные хранятся в бд (SQLite).

iOS приложение:

Основной, с точки зрения юзера, компонент проекта. Написан на Swift, использует, в основном, стандартные средства разработки Xcode 8 и iOS 10. Как приложение, предоставляет пользователю (возможно ограниченное пользование без авторизации) возможность искать (live search), добавлять, редактировать, удалять сотрудников из базы, а также менять собственные настройки. Пароли при веб запросах в открытом виде не передаются, используется его MD5 хеш. Использует технологию CocoaPods для подключения сторонних библиотек, среди которых Alamofire (для работы с веб-запросами), SwiftyJSON (удобный интерфейс для манипуляции с JSON данными) и SwiftHash (генерация MD5). Соблюдается принцип MVC.

1. Интерфейс программы:

7. Состав классов

Две основные сущности, используемые по всему проекту: пользователь приложения/сайта и сотрудник. Сотрудник имеет поля

ФИО

Почта/почты

Адрес

Ссылка на личную страницу

Телефон

Ссылка на фото

Пользователь ограничивается логином и паролем (его MD5 суммой)

8. Использованные ресурсы

[stackoverflow.com](http://stackoverflow.com)

[github.com](http://github.com)