**«САЙТ-АГРЕГАТОР**

**ДЛЯ ПОДБОРА МЕДИЦИНСКОГО ПЕРСОНАЛА»**

*Руководство по развертыванию веб-интерфейса*

Эксплуатационный документ

в текстовом виде

на пяти листах

Разработал:

студент IV курса

направления «Программная инженерия»

Подолян Александр Вячеславович

Рыбница, 2020

# Структура проекта

Структура проекта состоит из следующих файлов и папок:

* *settings* – содержит файлы формата *json* для записи аргументов, таких как специальность, штат, город и адрес почты;
* *spiders* – содержит веб-скраперов;
* *templates –* хранит шаблоны веб-страниц;
* *requirements.txt* – файл зависимостей приложения;
* *app.py* – сервер приложения.

# Настройка базы данных

1. *Создание кластера*

Для регистрации учетной записи MongoDB Atlas перейдите на [домашнюю страницу](https://www.mongodb.com/cloud/atlas?jmp=tuts&Americas_USA_tuts_stitch_tutorials_ATLAS) и нажмите кнопку Start free.

После того, как вы заполните короткую регистрационную форму, вы будете перенаправлены на страницу создания кластера. Вам нужно будет выбрать облачный провайдер Azure и регион США.

В разделе Cluster Tier выберите параметр M0, чтобы создать кластер бесплатного уровня. Дайте значимое имя кластеру и нажмите кнопку **Create Cluster**. Теперь подождите около пяти минут, пока ваш кластер настроится.

1. *Настройка кластера*

Прежде чем использовать кластер, вам нужно будет предоставить несколько сведений, связанных с безопасностью, поэтому перейдите на вкладку Database Access. Вы должны создать нового пользователя, нажав кнопку Add New Database User. В появившемся диалоговом окне введите нужное имя пользователя и пароль, выберите Atlas Admin или Read and write to any database и нажмите кнопку Add User.

Затем в разделе Network Access нужно указать список IP-адресов, с которых будут обращаться к кластеру.

Нажмите кнопку Add IP address, чтобы создать новую запись IP-адреса. В поле Whitelist Entry введите 0.0.0.0/0. Наконец, нажмите Confirm, чтобы добавить запись.

1. *Получение строки подключения*

Для подключения к кластеру из приложения вам понадобится действительная строка подключения. Чтобы получить ее, перейдите на вкладку Overview и нажмите кнопку Connect.

В открывшемся диалоговом окне выберите параметр Connect Your Application и выберите DRIVER – Python, VERSION – 3.6 or later. Теперь вы можете увидеть строку подключения. У нее не будет вашего фактического пароля, поэтому вам придется его вводить вручную.

# Создание приложения Python в Службе приложений Azure

Предварительные требования:

* Подписка Azure — [создайте бесплатную учетную запись](https://azure.microsoft.com/free/?ref=microsoft.com&utm_source=microsoft.com&utm_medium=docs&utm_campaign=visualstudio);
* Python 3.6 или последующей версии;
* [Git](https://git-scm.com/downloads);
* [Azure CLI](https://docs.microsoft.com/ru-ru/cli/azure/install-azure-cli) 2.0.80 или последующей версии.

Процесс развертывания веб-приложения Python для службы приложений на платформе Linux описан ниже.

1. *Скачивание приложения*

В окне терминала выполните следующую команду, чтобы клонировать приложение на локальный компьютер:

git clone https://github.com/Podolyan98/Azure.git

1. *Локальный запуск*

Перед развертыванием на Azure необходимо настроить подключение к базе данных, отправку email-сообщений и зарегистрировать приложение.

Для всех веб-скраперов и сервера необходимо указать строку подключения к MongoDB Atlas в коде.

Также для перезагрузки приложения нужно получить application id(client id), directory id(tenant), client secret(secret) с помощью Azure CLI и указать их в коде сервера:

az ad sp create-for-rbac –name < name>

Идентификатор приложения (клиент) subscription\_id можно получить на портале Azure в разделе Регистрация приложений. Его также нужно указать в коде сервера.

Чтобы использовать функцию получения уведомлений после завершения работы веб-скрапера укажите логин и пароль аккаунта Gmail в конфигурации сервера – MAIL\_USERNAME, MAIL\_DEFAULT\_SENDER, MAIL\_PASSWORD. После этого [разрешите](https://myaccount.google.com/lesssecureapps) использовать аккаунт для отправки сообщений, включите доступ по протоколу IMAP и подтвердите доступ с [нового](https://accounts.google.com/b/0/DisplayUnlockCaptcha) устройства (после развертывания на Azure повторите действие).

В окне терминала выполните приведенные ниже команды, чтобы установить необходимые зависимости и запустить встроенный сервер разработки:

py -3 -m venv env

env\scripts\activate

pip install -r requirements.txt

SET FLASK\_APP=app.py

flask run

Откройте веб-браузер и перейдите по адресу <http://localhost:5000/>.

Сервер запущен в режиме разработки. Для развертывания на Azure используйте app.run() в точке входа приложения.

1. *Развертывание в облаке Azure*

Чтобы выполнить команды Azure в интерфейсе командной строки Azure, необходимо сначала выполнить вход с помощью команды az login. Эта команда открывает окно браузера для ввода учетных данных.

Команда az webapp up создает веб-приложение в Службе приложений и развертывает код. В папке azure, содержащей код, выполните следующую команду az webapp up. Замените <app-name> глобальным уникальным именем приложения (допустимые символы: a-z, 0-9 и - ):

az webapp up --sku B1 -n <app-name>

Аргумент --sku B1 создает веб-приложение в ценовой категории "Базовый". Этот аргумент можно опустить, чтобы использовать ценовую категорию "Премиум" с почасовой стоимостью. Вы также можете добавить аргумент -l <location-name>, где <location\_name> — это регион Azure, например centralus, eastasia, westeurope, koreasouth, brazilsouth, centralindia и т. д. Список допустимых регионов для учетной записи Azure можно получить, выполнив команду az account list-locations. Выполнение команды az webapp up может занять несколько минут.

Найдите развернутое приложение в веб-браузере по URL-адресу http://<app-name>.azurewebsites.net. Повторно развернуть приложение можно с помощью команды az webapp up. Эта команда использует значения, которые кэшируются в файле .azure/config, включая имя приложения, группу ресурсов и план службы приложений.

Вы можете получить доступ к журналам консоли, созданным в приложении, и контейнеру, в котором он выполняется. Журналы содержат все выходные данные, созданные с помощью инструкций print. Чтобы включить потоковую передачу журналов, выполните команду az webapp log tail.

Перейдите на портал Azure, чтобы управлять созданным приложением. Найдите в поиске и выберите Службы приложений. Выберите имя приложения Azure. Отобразится страница обзора вашего приложения. Вы можете выполнять базовые задачи управления: обзор, завершение, запуск, перезагрузку и удаление.

В меню службы приложений слева доступно несколько страниц для настройки приложения. Выберите пункт Горизонтально увеличить масштаб и установите количество экземпляров – 3. В пункте Конфигурация включите режим ‘Всегда включено’. Настроить ограничение доступа по IP-адресу можно в пункте Сеть.