

# ЧАТ-БОТ СТУДЕНТА ТПУ

В.А. Бокор

Научный руководитель: А.С. Фадеев  
Томский государственный университет  
E-mail: vab24@tpu.ru

## Введение

Информационные технологии – быстро развивающаяся отрасль, поэтому, вполне закономерно, что в 21 веке она проникает во многие сферы деятельности человека. Образование не стало исключением. Будет естественно предположить, что высшие учебные заведения будут следовать прогрессу и начнут (продолжат) широко использовать и развивать цифровые технологии в своей деятельности.

Актуальность информации теряется со временем. Только вовремя полученная информация может быть полезна, поэтому очень важно, чтобы она поступала своевременно. Благодаря чат-ботам, ВУЗы могут полностью изменить способ взаимодействия учащихся со своими университетами.

Данный вопрос разрабатывается в разных учреждениях уже несколько лет. Так, например, участникам хакатона EdHack: ChatBots было предложено решить проблемы современного онлайн образования и использовать чат-боты для повышения эффективности обучения и качества взаимодействия с учебными материалами [1]. В Государственном университете управления был разработан чат-бот с новостями и расписанием в социальной сети «Вконтакте» [2]. В университете ИТМО был разработан бот, который работает в связке с онлайн-платформой Центра дистанционного обучения (ЦДО) вуза, и сейчас он рассылает пользователям информацию о начисленных баллах за выполненные учебные задачи, а также расписании [3].

Чат-боты могут служить круглосуточной справочной системой, стоимость которой намного меньше, чем содержание большого количества дополнительных сотрудников. Университеты, которые создали чат-бот для ответов на запросы студентов, фактически создали систему эффективного обмена информацией.

## Описание принципиального решения

Согласно данным компании **Mediascope**, аудитория Рунета в 2019 году достигла 93 млн человек (выросла на 7% за 3 года). Основным типом устройства для выхода в интернет в России на сегодняшний день являются смартфоны: за последние три года их проникновение выросло на 22% и составляет 61% [4]. Наиболее используемыми мессенджерами среди мобильных-пользователей считаются: VK, WhatsApp, Telegram.

На данный уже есть разработанные приложения для мобильных устройств, например, ТПУ Абитуриент [5]. Однако данное решение может быть неудобным для пользователя, т.к. ему необходимо привыкать к новому интерфейсу, скачивать отдельное приложение. Таким образом, необходимо решение, которое позволит общаться с пользователями именно в привычных им приложениях.

Для данной работы был выбран Telegram, т.к. в нем, в отличие от WhatsApp, есть возможность создания ботов и каналов. Кроме того, Telegram бесплатно предоставляет доступ к API [6].

Для ускорения разработки было решено ввести дополнительную прослойку и использовать архитектуру «клиент-серверного» приложения. Данная модель позволит:

- Вести разработку независимо от разработчиков API ТПУ;
- Иметь возможность подключать дополнительные «клиенты» (Telegram бот, VK бот, собственное приложение, и т.д.);
- Иметь локальную визуализацию данных через админ-панель.

Для разработки API было использовано сочетание Django/Django REST Framework.

Таким образом, используемую в решении архитектуру можно показать с помощью следующей диаграммы (рис.1)



Рис. 1. Архитектура предлагаемого решения

Предполагается, что пользователь (студент/преподаватель) будет регистрироваться в системе через клиентское приложение, введя свой уникальный идентификатор в системе ТПУ, а также номер телефона для подтверждения личности. В дальнейшем, с помощью этих данных можно получить всю информацию через описанное открытое API ТПУ. Кроме того, такая система позволяет подключить и другие ВУЗы.

## Описание реализации Telegram бота

Для написания Telegram бота был использован язык python с активным использованием модуля asyncio. За основу работы был взят фреймворк Telepot, который позволяет строить приложение на

основе официального Telegram API, а также имеющий асинхронную версию.

Заранее закладывается асинхронная версия, т.к. это позволяет обрабатывать большое количество запросов от пользователей одновременно.

По структуре созданная версия приложения придерживается архитектурного паттерна MVC, выделяя связанные компоненты в отдельные модули и описывая в каждом конкретном случае отдельную API.

В текущей версии был написан основной блок, который позволит расширять бота:

- Реализовано основное взаимодействие с пользовательскими сообщениями;
- Добавлена регистрация новых пользователей;
- Добавлены эмуляции запросов за расписанием.

В связи с тем, что в данный момент времени публичное API не имеет возможности получить расписание, данный запрос был эмулирован (рис. 2).

```
GET /api/schedule/schedule_emulation/

HTTP 200 OK
Allow: GET, OPTIONS
Content-Type: application/json
Vary: Accept

{
  "data": [
    {
      "date": "24.10.2019",
      "lessons": [
        {
          "index": 3,
          "name": "Пркт.Интернет-прил. (ПР)",
          "lector": "Соколова В. В.",
          "location": "К. КЦ, ауд. 418"
        },
        {
          "index": 4,
          "name": "Пркт.Интернет-прил. (ЛБ)",
          "lector": "Соколова В. В.",
          "location": "К. КЦ, ауд. 418"
        },
        {
          "index": 5,
          "name": "Польз.интерф.ИП (ПР)",
          "lector": "Саврасов Ф. В.",
          "location": "К. 10, ауд. 113Б"
        }
      ]
    },
    {
      "date": "25.10.2019",
      "lessons": []
    },
    {
      "date": "26.10.2019",
      "lessons": [
        {
          "index": 3,
          "name": "Администр.в ИС (ЛБ)",
          "lector": "Фадеев А. С.",
          "location": "К. 10, ауд. 109"
        },
        {
          "index": 4,
          "name": "Польз.интерф.ИП (ПР)",
          "lector": "Саврасов Ф. В.",
          "location": "К. КЦ, ауд. 413"
        }
      ]
    }
  ]
}
```

Рис. 2. Ответ в формате JSON на запрос расписания с API сервера

Полученный ответ позволил показать возможность отображения расписания в Telegram (рис. 3,4)

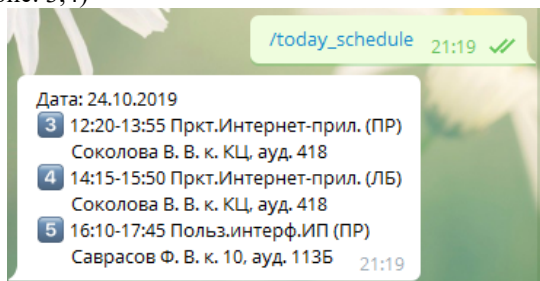


Рис. 3. Получение расписания на сегодня

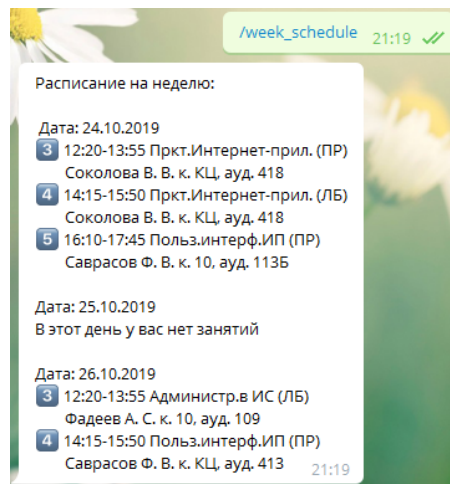


Рис. 4. Получение расписания на неделю

## Заключение

В результате проведенной работы был разработан:

- Промежуточный API сервер для работы с БД и API TPU;
- MVP Telegram бота, в котором на данный момент реализованы функции регистрации, помощи, получения расписания.

Следующим этапом работы будет создание API по выборке из БД ТПУ, а также реализация функций необходимых для полноценной работы Telegram бота.

## Список использованных источников

1. «Мой Универ» стал платформой для образовательных чат-ботов на EdHack: ChatBots [Электронный ресурс] – URL: <http://sk.ru/news/b/news/archive/2016/09/13/moy-univer-vystupila-platformoy-dlya-obrazovatelnyh-chatbotov-na-edhack-chatbots.aspx> (Дата обращения: 01.10.2019).
2. Чат-бот с новостями и расписанием появился в Государственном университете управления [Электронный ресурс] – URL: [https://guu.ru/news\\_ru/55084/](https://guu.ru/news_ru/55084/) (Дата обращения: 10.10.2019).
3. Бот расскажет студентам Университета ИТМО о начисленных баллах, расписании и личном рейтинге [Электронный ресурс] – URL: <http://news.ifmo.ru/ru/education/students/news/7276/> (Дата обращения: 15.10.2019).
4. Аудитория Рунета выросла на 7% за три года [Электронный ресурс] – URL: <https://mediascope.net/news/1035826/> / (Дата обращения: 16.10.2019).
5. ТПУ абитуриент [Электронный ресурс] – URL: <https://play.google.com/store/apps/details?id=ru.tpu.enrollee/> (Дата обращения: 10.01.2020).
6. Официальный сайт документации Telegram [Электронный ресурс] / – URL: <https://core.telegram.org/> (Дата обращения: 04.10.2019).