# Слайд 2:

Данная работа очень актуальна, так как этот модуль разрабатывался по заказу «Абитуриент-центра», который проводит около 30 мероприятий в год. И на каждое из них записывается примерно по 70 абитуриентов. Автоматизация работы с их данными будет очень полезна сотрудникам центра.

# Слайд 3:

Разработанный модуль поможет оптимизировать работу сотрудников центра с их CRM системой. В качестве исходных данных у нас есть сайт системы. На нём мы будем взаимодействовать с базой данных абитуриентов. Мы будем дополнять её их идентификационными данными и академическими успехами.

Результатом этой работы будет функционирующий модуль, который поможет быстро и без ошибок заниматься обычно рутинной работой.

# Слайд 4:

Для каждого мероприятия «Абитуриент-центр» составляет анкеты, в которых абитуриенты записываются на эти мероприятия. Анкеты составляются в сервисе Google Forms. Результаты анкетирования автоматически сохраняются в таблицы, которые хранятся в облачном хранилище автора анкеты. Также сервис от гугл предоставляет удобные инструменты для автоматического оповещения о новых ответах на анкеты.

# Слайд 5:

Для разработки модуля нужно было определить его функциональные требования. (пауза) Приложение должно:

* считывать данные из электронных таблиц;
* форматировать данные подобно их представлению в CRM системе, которую настроила Наталья;
* заносить данные в систему.

Данный модуль можно запускать на любых версиях операционной системы Windows начиная XP.

# Слайд 6:

Электронные таблицы находятся в облачном хранилище, поэтому для доступа к ним нужны ссылки. В первых версиях приложения планировалось хранить эти ссылки в файле. Также туда можно было бы дописывать новые ссылки на новые таблицы. Но для этого понадобился бы графический интерфейс и приложение не было бы автоматическим.

Но затем был обнаружен способ получать ссылки на таблицы через автоматические оповещения об ответах, что позволило упростить работу приложения и отказаться от интерфейса. Теперь модуль автоматически работает на фоне и не требует ручного вмешательства.

И теперь для достижения поставленной задачи необходимо выполнить следующие шаги:

1. Проверить почту, на которую приходят оповещения о новых ответах в виде писем;
2. Найти в этих письмах вложенные ссылки на таблицы с ответами;
3. Получить доступ к таблицам и собрать данные оттуда;
4. Отформатировать данные;
5. Отправить их в CRM систему;

# Слайд 7:

Модуль реализован на языке Python с использованием среды разработки PyCharm и различных библиотек, помогающих получать доступ к сервисам и обрабатывать данные.

# Слайд 8:

Для корректной работы модуля, необходимо провести небольшую настройку анкет и таблиц. Анкеты нужно составлять по шаблону, называть определённым образом и настраивать оповещения об ответах. Сервисным пользователям нужно предоставить доступ к таблицам ответов. Для всех этих действий разработана подробная инструкция.

# Слайд 9:

Модуль представляет из себя бесконечный цикл с таймером, по истечению которого происходят описанные на диаграмме последовательностей действия.

Сначала мы с помощью библиотек IMAP и EMAIL получаем доступ к почтовому ящику сервисного аккаунта Гугл и ищем непрочитанные письма от Google Forms.

# Слайд 10:

Затем, используя регулярное выражение, ищем в теле письма ссылку на таблицу с ответами, а также подсчитываем количество писем от одинаковых анкет для дальнейших действий.

# Слайд 11:

По ссылкам мы переходим к таблицам в облачное хранилище и собираем все данные оттуда, затем преобразуем их в удобный вид.

# Слайд 12:

С помощью библиотеки PySugarCRM получаем доступ к системе, формируем данные в модуль абитуриентов и загружаем его в базу.

# Слайд 13:

Результаты.

Для решения поставленной задачи была проанализирована предметная область, изучены некоторые интернет-сервисы, спроектирован и реализован программный модуль, выполняющий все необходимые действия, а также составлена подробная инструкция по использованию этого модуля для сотрудников «Абиуриент-центра».

На данный момент модуль можно запускать с рабочих машин сотрудников центра. Но в будущем планируется переписать его на php, чтобы перенести на сервер университета, где он будет работать полностью автоматически.