Для финансовых отчетностей используется сайт Saleforce, один из сервисов которого называется Sage. В рамках задачи [DP-162](https://daxdev.atlassian.net/browse/DP-162) была реализована автоматическая загрузка курсов валют для Sage, что происходит непосредственно через Saleforce. Для этого был создан класс SageAPI, а также три вспомогательных класса для чтения возвращаемых значений. Подключение к Sage и загрузка курсов состоит из нескольких этапов:

1. Создание внешнего подключения для API на сайте Saleforce (предварительный этап);
2. Логин в сервис;
3. Формирование и отправка новой строки с курсами;
4. Анализ полученного ответа.

Существует два способа логина в сервис.

**Первый** способ использует три основные настройки: **client\_secret, client\_id** и **redirect\_uri.** Этот способ лучше тем, что не требует логин и пароль, соответственно, он более секьюрный. Логин происходит в два этапа: сначала с помощью ключа и url запрашивается код авторизации, который живет только 15 минут, затем с помощью этого кода авторизации и трёх настроек происходит получение access\_token (про его значение будет ниже). Однако этот способ на данный момент не подходит, т.к. он заточен не под api, а под наличие собственного приложения или сервера. На первом этапе логина требуется ручное разрешение на получение кода по данному url, и в последующем код авторизации присылается редиректом на указанный redirect\_uri.

**Второй** способ использует четыре основные настройки: **username, password, client\_secret** и **client\_id.** Логин происходит в один этап - отправляются все основные настройки, в ответ получается access\_token и можно начинать работу. Этот способ выглядит менее секьюрно, но он заточен под api и именно им нам предстоит пользоваться.

Properties для Sage

Сейчас для автоматической выгрузки курсов мы пользуемся TeamCity. Раз в сутки в установленное время в разделе Financical reports запускается утилита Checkout Rates. Все необходимые для логина системные настройки уже записаны в её переменные окружения, доступ к ним есть только у администраторов проекта. Для всех остальных пользователей они скрыты. Для того, чтобы приложение успешно отработало в него передаётся несколько аргументов - дата начала скачивания курсов (с какого числа нужны курсы), дата последнего дня для скачивания (если по сей день - то now вместо даты), валюты, в которые надо конвертировать полученные данные. Для быстрого запуска эти данные имеют дефолтные значения, однако при необходимости можно выбрать Run custom build и в Parameters указать нужные параметры.

Для того, чтобы успешно залогиниться на сайте через запрос необходимо иметь следующие данные:

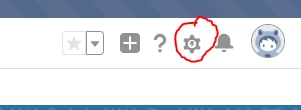
1. Url - ссылка на рабочий профиль Saleforce. Имя настройки на TeamCity - **SAGE\_LOGIN\_URL**
2. Имя пользователя **(username)**- имя пользователя, иначе говоря почта, по которой пользователь зарегистрирован. Имя настройки на TeamCity - **SAGE\_USERNAME**
3. Пароль **(password)** - пароль для входа на сайт. Имя настройки на TeamCity - **SAGE\_PASSWORD**
4. Секретный ключ **(client\_secret)** - секретный код приложения, созданного через личный кабинет в Saleforce. Имя настройки на TeamCity - **SAGE\_CLIENT\_SECRET**
5. Идентификатор приложения **(client\_id)** - секретный ключ приложения, созданного через личный кабинет в Saleforce. Имя настройки на TeamCity - **SAGE\_CLIENT\_ID**

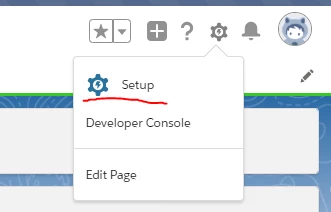
Создание внешнего подключения для API

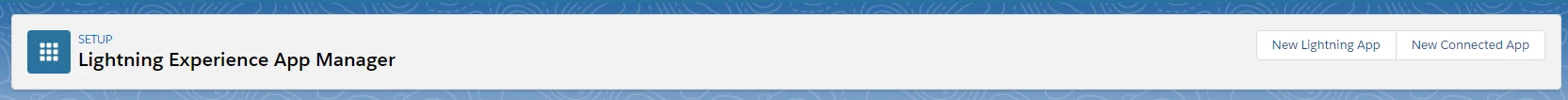
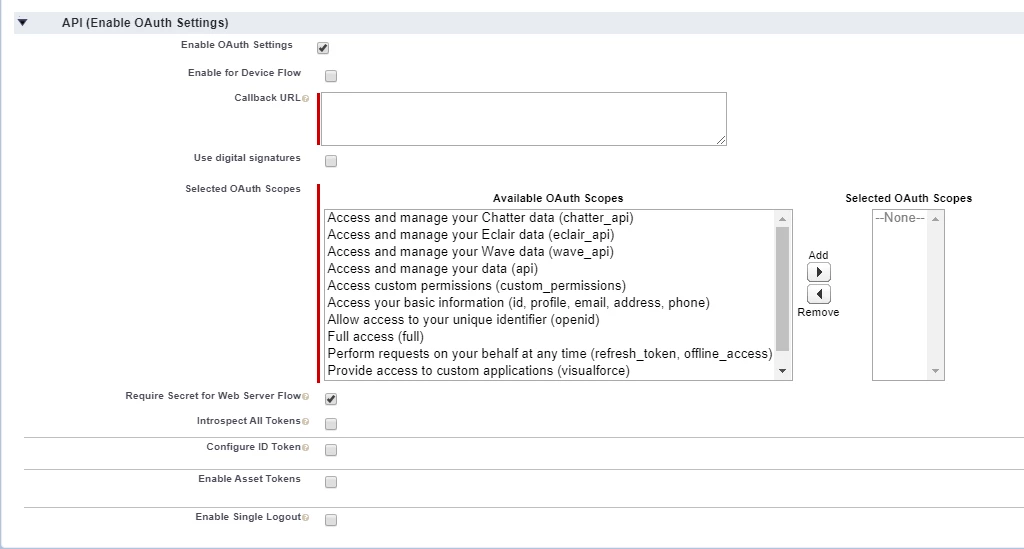
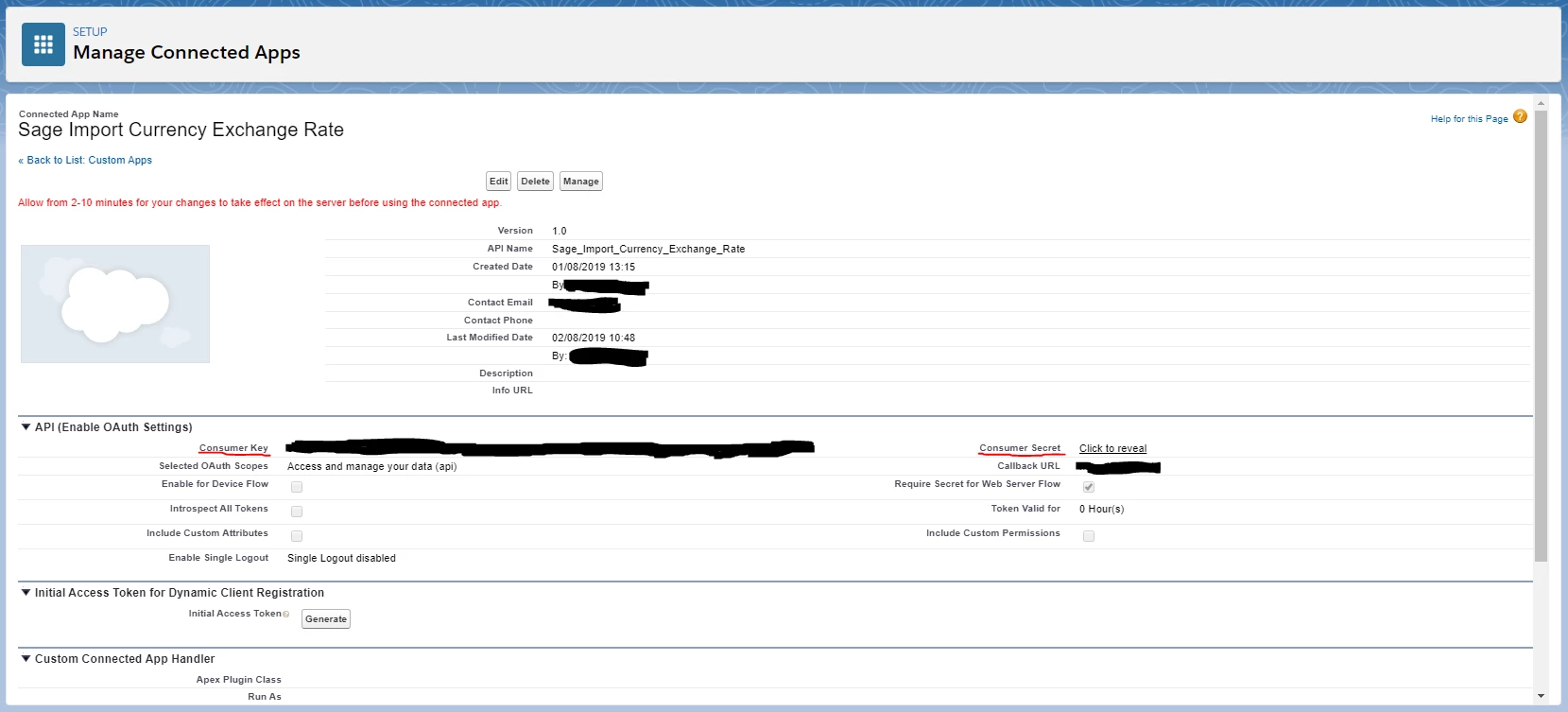
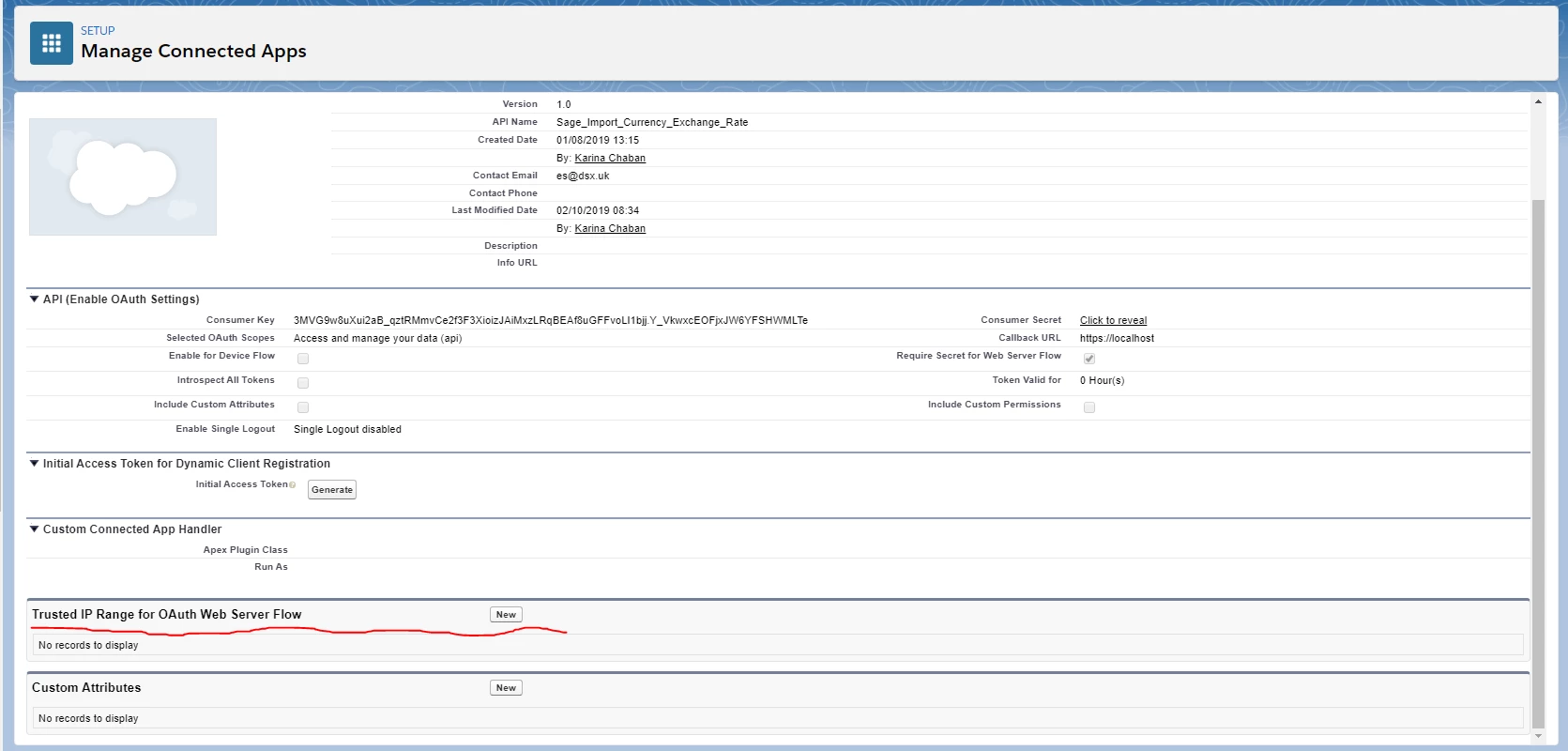
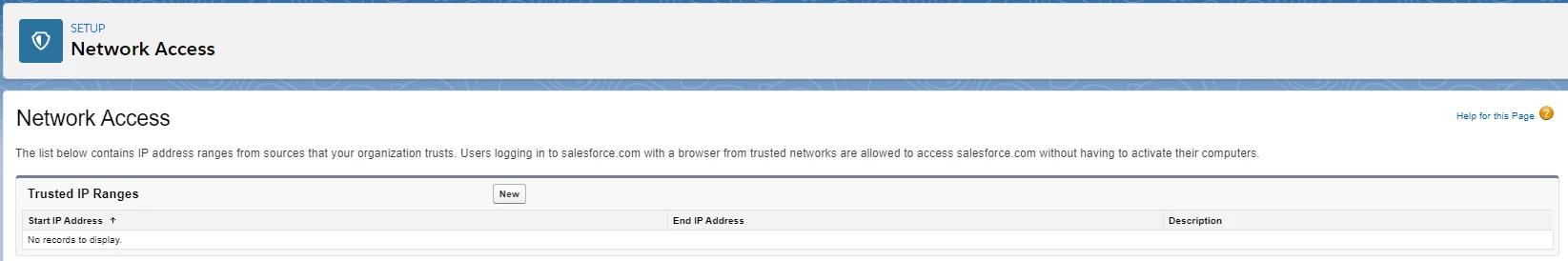
Внешние API подключаются не к самому профилю, а к специально созданному вручную подключению (приложению). Однако создать такое подключение может только пользователь с определенным разрешением профиля - Manage Connected Apps.   


Настройка этого разрешения есть только у пользователя с типом лицензии Salesforce, т.е., пользователь, который является владельцем лицензии для работы с Salesforce. Это главный администратор, System Administrator. *Возможно, этот тип лицензии можно дать сабпользователям, созданным администратором, либо же настроить им конкретно работу с Manage Connected Apps, однако в данный момент этот вопрос в процессе изучения. Не исключено, что это невозможно.*

**Во избежание неприятностей у аккаунта обязательно должен быть настроен 2FA. В этом случае при негативном сценарии логина и пароля будет не достаточно, чтобы зайти в аккаунт.**  
  
Чтобы создать подключение, необходимо после логина:

1. В верхнем правом углу нажать на шестеренку, и затем на **Setup**  
     
   



1. В левом меню пройти по пути **Apps → App Manager**;
2. Выбрать пункт **New Connected App**  
     
   
3. В появившемся меню установить все необходимые настройки, такие как имя приложения, лого (по желанию), а также настроить права доступа для этого приложения. Для этого надо включить пункт **Enable OAuth Settings.**В нем нужно указать url для обратной связи (для запросов, которые этого требуют) и выбрать один или несколько пунктов того, к каким данным это приложение имеет право получить доступ.  
   
4. После создания приложения сайт перенаправит на страницу этого приложения, откуда его можно будет редактировать и удалять. На этой же странице содержатся необходимые нам для соединения данные. Здесь **Consumer Key = client\_id,**а **Consumer Secret = client\_secret**.  
   
5. После создания приложения необходимо настроить доступ к нему. Для этого нужно на странице редактирования выбрать Manage и удостовериться, что настройки выглядят следующим образом:  
     
   Настройка Permitted users означает, что пользователи могут логиниться через это приложение без обязательного утверждения этого действия администратором. Настройка IP Relaxtion устанавливаем, позволяем ли мы любому ip доступ к приложению, позволяем ли доступ тем ip, которые уже коннектились и мы их проверили или же мы закрываем доступ всем ip, кроме тех, которые самостоятельно прописали в настройках.
6. Дальше опять же на странице редактирования приложения необходимо прописать те ip, которым мы позволяем доступ к приложению. Там используется диапазон, но если это один четко заданный ip, то его можно вписать в оба поля.  
   
7. Разрешенные ip надо также указать в **Security → Network Accsess.** Если этого не сделать, приложение не сможет получить доступ к аккаунту. **Если ip не прописан, то никакое стороннее подключение не получит доступ, даже если client\_id и client\_secret будут известны.**  
   Также [здесь](https://stackoverflow.com/questions/12794302/salesforce-authentication-failing/29112224#29112224) есть подробная инструкция этого процесса. Стоит обратить внимание на пункт 9. Сейчас он уже настроен, но если что-то не работает, стоит перепроверить.

На этом создание внешнего подключения завершено и все дальнейшие запросы идут из кода. Чтобы снова попасть на страницу редактирования приложения необходимо выбрать **Apps → App Manager → нажать на стрелочку на нужном приложении → Edit или Manage.**

**Если надо добавить новый разрешенный ip**, а на странице Manage в нужном разделе пропала кнопка New, попробуйте сначала выбрать Edit → Save, ничего при этом не меняя. Кнопка должна появиться.

Логин в сервис

Запрос для входа в систему выглядит следующим образом:

***https://{{YOUR\_URL}}/services/oauth2/token?grant\_type=password&client\_id={{client\_id}}&client\_secret={{client\_secret}}&username={{username}}&password={{password}}***

где данные в скобках {{ }} - это личные данные пользователя и приложения. В ответ на этот запрос сайт присылает json-объект, который содержит код и, если запрос был успешен, **access\_token** - токен, который действует как идентификатор сеанса. Приложение использует его  
для отправки запросов, поэтому он содержится в заголовке всех следующих запросов с**Key = Authorization** и **Value = Bearer {{access\_token\_from\_Salesforce}}**. В качестве **Content-Type** проставляется **application/json.**

Для первого способа входа в систему запросы выглядят так:

***https://{{YOUR\_URL}}/services/oauth2/authorize?response\_type=code&client\_id={{client\_id}}&redirect\_uri={{redirect\_uri}}&state={{random\_string}}***

**State**- совершенно любая строка, которую можно генерировать случайным образом. Она не влияет на работу запроса и в дальнейшем подставляется сгенерированную в ответ ссылку. Этот параметр не обязательный, но рекомендуемый, так как он минимизирует риск атаки.  
Полученная в ответ ссылка предназначена  для перенаправления, и когда редирект произойдет, в url ссылки будет указан код авторизации, который надо будет подставить в следующий запрос:

***https://{{YOUR\_URL}}/services/oauth2/token?grant\_type=authorization\_code&client\_id={{client\_id}}&client\_secret={{client\_secret}}&redirect\_uri={{redirect\_uri}}***

Так же, как и в первом случае, в ответ будет получен **access\_token.**

Формирование и отправка строки с курсами

Загрузка курсов валют происходит через создание новых **Exchange records**. У этих строк есть ряд значений, которые полностью описаны в классе ExchangeRecordJSON. Часть из них Sage заполняет автоматически после создания записи. Передаются же в запрос следующие параметры:

| **Параметр** | **Формат** | **Описание** |
| --- | --- | --- |
| s2cor\_\_UID\_\_c | GBP\_EUR\_2019-07-29\_00:00:00 | Используется для внутреннего поиска Saleforce, в нём указывается конверсия валют и дата актуальности данного курса. |
| s2cor\_\_Type\_\_c | "Spot" | Тип курса обмена валют. У этого параметра может быть два значения - Spot и Fixed. Спотовая ставка обычно используется в качестве ставки по умолчанию для автоматических процессов, её и используем. |
| s2cor\_\_Rate\_\_c | 1.0988 | Непосредственно курс валют. Т.к. тип integer в json-строке съедает нули на конце (и в принципе может быть не точен), курс передаем как строку. Так её в дальнейшем можно перевести в BigDecimal. |
| s2cor\_\_To\_Currency\_\_c | a4k1d000000lLh7UTF | В какую валюту конвертируем. Передаем внутренний id из Saleforce. |
| s2cor\_\_Label\_\_c | GBP\_EUR\_2019-07-29\_00:00:00 | Описание записи, отображаемое пользователю. Для удобства ориентации в записях совпадает с uid. |
| s2cor\_\_Effective\_From\_\_c | 29/07/2019 00:00 | Дата актуальности данного курса валют. |
| s2cor\_\_Currency\_\_c | a4k1d000000lLh7UTS | Какую валюту конвертируем. Передаем внутренний id из Saleforce. |

Параметры передаются в качестве тела запроса. Сама строка запроса имеет следующий вид:

***https://{{YOUR\_URL}}/services/oauth2/token/services/data/v20.0/sobjects/s2cor\_\_Sage\_COR\_Currency\_Exchange\_Rate\_\_c***

Анализ полученного ответа

Если строка создана успешно, то можно переходить к следующей строке. Если же нет, то необходимо разобраться с тем, какая ошибка вернулась. Результат запроса сериализуется в класс ErrorResultFromSaleforceJSON, который содержит три поля: код ошибки, сообщение и некорректные поля. Последняя часть ответа может быть пустая, остальные две NotNull.

Чаще всего возвращаются три типа ошибок:

**INVALID\_SESSION\_ID** - токен устарел, необходим перелогин.

**MALFORMED\_ID** - неправильно указаны id валют. Так как это значение константно, следует перепроверить их написание и их значение в личном кабинете.

**DUPLICATE\_VALUE** - строка с таким uid уже существует. В этом случае составляется запрос getCurrencyExchangeRecordById, куда передается id из полученного ответа. У полученной и создаваемой строк сравнивается rate, и если он одинаковый - скорее всего, мы по ошибке скачали данные, которые уже есть. Если разный - алёрт, что-то пошло не так, надо смотреть админу.

Может вернуться ещё много разных ошибок, их мы пишем в стандартной форме. Все ошибки складываются в лист **adminMessage**, и если по завершении работы этот лист не пустой, то происходит вылет с логированием всех ошибок в этом листе.

Полезные ссылки:

[Документация по API для Saleforce](https://resources.docs.salesforce.com/sfdc/pdf/api_rest.pdf)

[Варианты запросов для Sage](https://developer.sage.com/api/financials/api/#operation/updates2cor__Sage_COR_Currency_Exchange_Rate__c)

[Инструкция по составлению GET-запросов с использованием SOQL](https://developer.sage.com/api/financials/how-to/)

[Инструкция по синтаксису SOQL](https://developer.salesforce.com/docs/atlas.en-us.soql_sosl.meta/soql_sosl/sforce_api_calls_soql_select.htm)