External companies that wish to programmatically connect with our Salesforce org need to use the built in REST API provided by Salesforce. REST is an acronym for Representational State Transfer and API is an acronym for Application Programming Interface. When used in the context of web development, an API is typically a defined set of Hypertext Transfer Protocol (HTTP) request messages, along with a definition of the structure of response messages.

Salesforce has [excellent documentation](#) on how to connect using the REST API. Below is a high-level summary of their documentation along with some explanation specific to the way Sedera has implemented our data schema.

The process begins with an out-of-band POST request to https://login.salesforce.com/services/oauth2/token using the client ID and Secret Question assigned to you by Sedera. That is combined with username and password to request an access token. Using the following username and password and example ID and Secret question the request would look like this:

username: iuser@sedera.com (note that isn't a real email address)
Password: Seder@user123
Consumer Key: [sent by Sedera at account creation]
Consumer Secret: [sent by Sedera at account creation]

Example token request POST
grant_type=password&client_id=[exampleGUID]&client_secret=
[example secret]username=iuser%40sedera.com&password=Seder@user123

Note: Token is not required since the Profile for the Integration User account is IP restricted. (note: any computer or server trying to POST to our Salesforce org will need to be whitelisted by Sedera in advance. This includes development, staging and production hardware)

Salesforce will reply with an access token. The response should look like:

Example token response
{"id":"https://login.salesforce.com/id/00Dx0000000BV7z/005x00000012Q9P",
"issued_at":"1278448832702","instance_url":"https://mysedera.salesforce.com/",
"signature":"0CmxinZir53Yex7nE0TD+zMpvIWYGb/bdJh6XfOH6EQ=","access_token":
"00Dx0000000BV7z!AR8AQAxo9UfVkh8AlV0Gomt9Czx9LjHnSSpwBMmbRcgKFmxOtvxjTrKW
19ye6PE3Ds1eQz3z8jr3W7_VbWmEu4Q8TVGSTHxs"}

"Issued_at" is the number of seconds since the beginning of UNIX time (1/1/1970) when the signature was created.

"Signature" is Base64-encoded HMAC-SHA256 signature signed with the client_secret (private key) containing the concatenated ID and "issued_at" value.

Adding a New Member - Simple Setup

The simplest way to add a New Member is to POST the minimum information needed for Sedera's to send an automated welcome email and a separate SMS text with a link to our mobile web app. The minimum required information is (with Salesforce API name in parentheses):

1. Account Number (Account_Employer_Name__c)
2. Account Name (Name)
3. First Name (First_Name__c)
4. Last Name (Last_Name__c)
5. Mobile Phone (Primary_Phone_Number__c)
6. Enrollment Date (Enrollment_date__c_) (note this is the first day of the month the prospect wishes to become a member)
7. Unique Partner ID (Unique_Partner_id__c) (this is the unique identifier you use in your DB schema for each member)

The POST would look like:

curl https://mysedera.salesforce.com/services/data/v20.0/sobjects/Account/ -H "Authorization: Bearer token -H "Content-Type: application/json" -d "@newmember.json"

The Bearer token is the Access Token received in the Oauth step above.

The newmember.json would look like:
```
{
        "Account_Employer_Name__c" : "0011N00001SJoTG"
        "Name" : "FirstName LastName"
        "First_Name__c" : "Jane"
        "Last_Name__c" : "Doe"
        "Primary_Phone_Number__c" : "555-555-0000"
        "Enrollment_Date__c" : "yyyy-mm-dd" (note this format must be exact or POST will fail
        "Unique_Partner_id__c" ; "  [up to 40 characters]  "
}
```

Salesforce will reply with the new Member number in a JSON block like this:
```
{
  "id" : "001D000000IqhSLIAZ",
  "errors" : [ ],
  "success" : true
}
```

Note that Salesforce uses 15 digit unique IDs for Accounts and Contacts. However, when you make an API call Salesforce usually returns an 18 digit version where the last 3 digits are a checksum on the first 15 digits. In subsequent REST calls you can use either the 15 or 18 digits. Most likely you will want to store this number in your database in order to easily match the data payload coming from Salesforce once the new member completes their application using the mobile app.

Salesforce uses different URLs, or routes, for accessing objects.

To access the Contact object use the Contact route:

curl https://mysedera.salesforce.com/services/data/v20.0/sobjects/Contact/ -H "Authorization: Bearer token -H "Content-Type: application/json" -d "@newcontact.json"

and then the newcontact.json block would look like:

```
{
    'First_Name__c' : 'Joe',
    'Last_Name__c' : "Tester',
    'Gender__c' : 'M',
    'Relationship_to_Member__c' : 'Spouse',
    'DOB__c' : 'mm/dd/yyyy',
    'App_Signature_signs' : 'sharing_initials_Chronic_Conditions'
    "AccountId": "001D000000IqhSL" <-- this is the 15 digit GUID for the Primary Member
Account of record type Member
    etc.
}
```

curl https://mysedera.salesforce.com/services/data/v20.0/sobjects/Contact/ -H "Authorization: Bearer token -H "Content-Type: application/json" -d "@chronic.json"

If you were just posting to this field the JSON block for Chronic would look like:

```
{

'App_Signature_Signs__c' : 'sharing_initials_Chronic_Conditions'

}
```

However, nearly 100% of the time if a user is having to send sharing_initials_Chronic_Conditions they are also having to send 'sharing_initials_prior_treatment' in which case you can send

```
{

'App_Signature_Signs__c' : 'sharing_initials_Chronic_Conditions ;
sharing_initials_prior_treatment'

}
```

PATCH
To make changes to an existing object, use PATCH instead of POST on the Account or Contact route. For example, to change the Discount Tier on an existing Account:

'patch' to:
curl

https://mysedera.salesforce.com/services/data/v20.0/sobjects/Account/Id/0011N00001hpugwQAA

 with JSON of:

```
{

 'AccountNumber' : '0011N00001hpugwQAA',

 'Member_Discount_Tier_Manual__c' : 'T1',

 'Member_MEC_Product__c' : 'SHA Preventive'

}
```

To write to Condition object

The simplest way to add the Condition is to POST the minimum information which is needed through the Condition route as below-

curl https://mysedera.salesforce.com/services/data/v20.0/sobjects/Condition__c/ -H "Authorization: Bearer token -H "Content-Type: application/json" -d "@newcondition.json"

and then the newcondition.json block would look like:


```
{
```

```
        "Account__c" :"001D000000IqhSL" <-- this is the 15 digit GUID for the Primary
Member Account of record type Member,
        "Contact__c" : "0031N000020TeXj",
        "Description__c" : "Heart attack",
        "Is_Cured__c" : "1" [ 1 for Yes and 0 for No],
        "Is_Ongoing__c" : "0",
        "Treatment_End_Date__c" : "11/21/2015",
        "Locked__c" : conditions[i].locked ? true : false
    };
```