```lua
-- test for TwilioRestClient

local Rest = require "Twilio.TwilioRestClient"
--My own authentication credentials are not commited to repository. Get your own!
local auth = require "tests.auth" or {}
local json = require("json")
local Util = require "Twilio.Util"

--Plug in your twilio account credentials here in the XXXX's
local ACCOUNT_SID = auth.ACCOUNT_SID or "XXXXXXXX" -- your Account SID
local ACCOUNT_TOKEN = auth.ACCOUNT_TOKEN or "XXXXXX" --your account token

--Outgoing Caller phone number
local CALLER_TO = auth.CALLER_TO or "+1NNNNNNNNNN"
--Incoming Caller Phone Number, previously validated with Twilio
local CALLER_FROM = auth.CALLER_FROM or "+1NNNNNNNNNN"

local r = Rest.create(ACCOUNT_SID, ACCOUNT_TOKEN)

local vars={}

local function network_callback(event)
    if ( not event.success ) then
        print( "Unsucessful Response: ", event.message )
    else
        --Now we have a Lua table of the json resonse!
        local response_table = event.response
        print(Util.to_string(response_table))
    end
end

--Make a call:
vars = {Type = "Calls", To = CALLER_TO, From = CALLER_FROM,
Url="http://demo.twilio.com/docs/voice.xml"}
--r:request(vars, "POST", network_callback)

--Make an invalid call, but valid http request
vars.From = nil
r:request(vars, "POST", network_callback)

--Send a message
vars = {Type="Messages", From=CALLER_FROM, To = CALLER_TO, Body="You are looking
sooo good today!"}
--r:request(vars,"POST", network_callback)
```

```lua
--Call Instance Resource
vars = {Type="Calls", InstanceSid = "1234567890"}
--r:request(vars,"GET",network_callback) --Fails unless InstanceSid is correct

--GET calls list
vars = {Type="Calls"}
--r:request(vars,"GET",network_callback) --this will fail
```