Neemi

I. Installation
  A. Infrastructure
    - mongoDB 2.4.2
      http://www.mongodb.org

    - Python 2.7.5

    - Django 1.5.1
      https://www.djangoproject.com

    - djangotoolbox (pip install djangotoolbox)

    - django-nonrel (pip install django-nonrel)

    - pymongo 2.6.3 (pip install pymongo)
      http://api.mongodb.org/python/current/index.html

    - mongoengine 0.8.2 (pip install mongoengine)
      http://mongoengine.org

    - oauth2 (pip install oauth2)
      http://oauth.net/2/

    - oauth2client (pip install oauth2client)
      https://code.google.com/p/google-api-python-client/wiki/OAuth2Client

    - requests (pip install requests)
      http://www.python-requests.org

  B. APIs
    1. Dropbox
       Dropbox Module
       https://www.dropbox.com/developers/core/sdks/python
       pip install dropbox

    2. Facebook
       Facebook requires changes to the rauth package:
       step 1) download rauth
       pip install rauth
       step 2) find rauth installation on local machine — /usr/local/lib/python2.7/dist-packages/rauth — and replace files service.py and session.py by the ones provided in neemi/packages/rauth/

    3. Foursquare
       Python wrapper for the foursquare v2 API

https://github.com/mLewisLogic/foursquare
pip install foursquare

4. LinkedIn

5. Twitter
   Python wrapper for the Twitter API
   https://github.com/ryanmcgrath/twython
   pip install twython

6. Google
   Google APIs Client Library for Python
   https://developers.google.com/api-client-library/python/start/installation
   pip install —upgrade google-api-python-client

   Google Data API
   pip install gdata

II. Configuration
   A. Django settings
   Edit file neemi/webapps/settings.py specifying name of database to be created to store all user data:
   Example - connect('neemi_mongodb_test')

   B. keys.py
   To be able to access an API the user has to register an application with their services. During registrations, the API's will provide at least a pair of id/secrets for each application registered and each application can serve multiple users.

A keys.py (neemi/webapp/keys.py) file should look like the example below:

```
import os

## DROPBOX INFO
DROPBOX_APP_KEY = 'XXXXXXXXX'
DROPBOX_APP_SECRET = 'XXXXXXXXX'
# ACCESS_TYPE should be 'dropbox' or 'app_folder' as configured for your app
DROPBOX_ACCESS_TYPE = 'dropbox'
DROPBOX_REDIRECT_URI = 'http://lvh.me:8000/dropbox_authorize/callback'

## TWITTER INFO
TWITTER_CONSUMER_KEY = 'XXXXXXXXX'
TWITTER_CONSUMER_SECRET = 'XXXXXXXXX'
TWITTER_REDIRECT_URI = 'http://lvh.me:8000/twitter_authorize/callback'

## LINKEDIN INFO
LINKEDIN_API_KEY = 'XXXXXXXXX'
LINKEDIN_API_SECRET = 'XXXXXXXXX'
LINKEDIN_REDIRECT_URI = 'http://lvh.me:8000/linkedin_authorize/callback'

## FOURSQUARE INFO
FOURSQUARE_CLIENT_ID = 'XXXXXXXXX'
FOURSQUARE_CLIENT_SECRET = 'XXXXXXXXX'
FOURSQUARE_REDIRECT_URI = 'http://lvh.me:8000/foursquare_authorize/callback'

## FACEBOOK INFO
FACEBOOK_CLIENT_ID = 'XXXXXXXXX'
FACEBOOK_CLIENT_SECRET = 'XXXXXXXXX'
FACEBOOK_REDIRECT_URI = 'http://lvh.me:8000/facebook_authorize/callback'

## GOOGLE INFO
GOOGLE_CLIENT_ID = 'XXXXXXXXX'
GOOGLE_CLIENT_SECRET = 'XXXXXXXXX'
GOOGLE_EMAIL_ADDRESS = 'XXXXXXXXX'
GOOGLE_REDIRECT_URI = 'http://lvh.me:8000/google_authorize/callback'
```

- Registering a dropbox application:
  Go to https://www.dropbox.com/developers/apps and follow the steps to create a new application. At the end, add the app key and secret to the keys.py file.

- Registering a twitter application:
  Go to https://dev.twitter.com/apps and follow the steps to create a new application. At the end, add the consumer key and secret to the file keys.py.

- Registering a linkedin application:
  Go to http://developer.linkedin.com/documents/authentication. Step 1 explain how to create a new application. Once you've registered your LinkedIn app, you will be provided with an API key and secret key that should be added to the keys.py file.

- Registering a foursquare application:

Go to https://foursquare.com/developers/apps and follow the steps to create a new application. At the end, add the client id and secret to the file keys.py.

- Registering a facebook application:

Go to https://developers.facebook.com/apps and follow the steps to create a new application. Client id and secret should be added to the file keys.py.

- Registering a google application:

Go to https://code.google.com/apis/console and register a new application. In the option "API Access" you can find your client id and secret. Besides that, you will have to enable each service that you are planning to use. It can be done under "Services".

III. How to use Neemi

A. Start server and call neemi interface

1. Open a terminal, go to your local copy of the neemi code and type:

   python manage.py runserver

   More details can be found in:
   https://docs.djangoproject.com/en/1.5/intro/tutorial01/

2. Open a browser and type:

   lvh.me:8000

B. The Neemi interface

1. The first page asks for a neemi user and password. A new user does not need to be registered a priori to have access to the application. The registration is done as soon as the user enters with the desired username and password.
2. In the main page the user can choose between the following options:
   a) Register for Services: show all APIs implemented allowing users to register with a variety of different services.
   b) Get Data from Services: user can get as much data as allowed by the APIs. This step requires the user to be authenticated with the service (Register for Services) before any data can be requested.
   c) Search your Data (Not working on this version)
   d) Delete Account and Data: clean all user data from database

IV. Possible problems

A. Foursquare - SSL3 certificate verify failed

   http://stackoverflow.com/questions/13321302/python-foursquare-ssl3-certificate-verify-failed

1. Download http://curl.haxx.se/ca/cacert.pem

wget http://curl.haxx.se/ca/cacert.pem

2. Go to Python httplib2 dir (/usr/local/lib/python2.7/dist-packages/httplib2)
   cd /usr/local/lib/python2.7/dist-packages/httplib2

3. Backup the current certificate
   cp cacerts.txt backup_cacerts.txt

4. Copy the downloaded file and rename it as cacerts.txt
   mv cacert.pem cacerts.txt