StrongsteamQuickStart Guide

Release 0.0.1

Ian Ozsvald, Kyran Dale, Balthazar Rouberol

May 09, 2012

CONTENTS

Welcome to the Strongsteam documentation !

Strongsteam is a cross-platform toolbox for developpers who want to "give a pair of eyes" to computers or mobile devices, to recognize images, words, semantic context, etc.

All results sent from Strongsteam are JSON-based. They thus can be interfaced with any programming language with JSON support:

- C
- C++
- C#
- D
- Java
- Javascript
- Matlab
- Objective-C
- Perl
- PHP
- Python
- R
- Ruby
- and many more...

CHAPTER

ONE

PREREQUISITES AND INSTALLATION

1.1 Requirements

- Python 2.6, 2.7 or 3.2
- requests 0.9.1
- progressbar 2.3

1.2 Install Strongsteam

Strongsteam is a cloud-based API, which means you do not need to install a whole bunch of packages or software. Everything you need is hosted on our server: you just have to send us the data and we'll take care of the rest.

All you have to do is install the `strongsteam` python library.

We strongly suggest to install `strongsteam` in a virtual environnement, using virtualenv.

```
$ virtualenv path/to/yourproject/
$ source yourproject/bin/activate
```

Note: `virtualenv` is a tool allowing you to create isolated python environnements. Read the documentation to learn more about it.

You can install the `strongsteam` library in your virtual environnement with pip, with the command

```
pip install http://dl.dropbox.com/u/6113789/Strongsteam-client/strongsteam-0.0.1.tar.gz
```

Important: Do not install the modules using `sudo`: this will install them in your system python path and not in the virtual environnement.

Note: Some examples are installed along with the strongsteam package. They are located in `lib/pythonX.X/site-packages/strongsteam/examples/` if you installed `strongsteam` in a virtualenv. Otherwise, they are located in `/lib/local/lib/pythonX.X/dist-packages/strongsteam/examples`.

CHAPTER

TWO

FIRST TEST

To validate that the installation went well and that our server is running, you can run the demo_hello_world.py test script.

You should see something like this in your shell :

```
$ python demo_hello_world.py
Processing job on uri /user/kyran/processes/hello_world...
{
        u'status': u'succeeded',
        u'time_stamps':
        {
                u'ts_job_ps_end': 1333542151.78501,
                u'ts_job_ws_receive': 1333542025.628467,
                u'ts_job_js_receive': 1333542151.619243,
                u'ts_job_ps_start': 1333542151.783706,
                u'ts_job_post': u'1333542150.91'
        },
        u'input_uri': None,
        u'process_name': u'hello_world',
        u'results':
        {
                u'hello_world_string': u'/user/kyran/vnd_ss_results/4f7c3d0711b3f4031f000000'
        },
        u'uri': u'/user/kyran/vnd_ss_results/4f7c3d0711b3f4031f000000',
        u'job_uri': u'/user/kyran/jobs/4f7c3c89fa1d115c3500004b',
        u'msg': u'Your job succeeded. Find the result-uris in results'
}
Job succeeded in 1.5700 seconds for /user/kyran/processes/hello_world
Hello world ian!
```

Note: Running this script can take more time (~30s) in the case where our server is asleep. Once awaken, everything will be blazing fast!

CHAPTER

THREE

SEND A JOB TO STRONGSTEAM

You can send us a job with just a few lines of code!

To understand how to do that, we'll go trough the demo_hello_world.py test script:

```
from strongsteam.clients import ss_client as ssc
from strongsteam.clients.ss_client import log

# set log to INFO if you want lots of progress information or
# use WARNING just to see the main client messages
log.setLevel(ssc.logging.WARNING)

if __name__ == "__main__":
        cli = ssc.StrongSteam()
        hello = cli.add_job(None, 'hello_world', data={'name':'oh, mighty Strongsteam user'})
        print hello.get_data()
```

3.1 Set-up a logger

You first need to import the `ss_client` class from the `strongsteam.clients` submodule, the `log` class from `cc_client` and set up a console logger.

```
from strongsteam.clients import ss_client as ssc
from strongsteam.clients.ss_client import log

# set log to INFO if you want lots of progress information or
# use WARNING just to see the main client messages
log.setLevel(ssc.logging.WARNING)
```

You then need to setup a `StrongSteam` client:

```
cli = ssc.StrongSteam()
```

3.2 Send a job

Whenever you want to send us a job, just use the `ss_client.StrongSteam.add_job()` method:

```
hello = cli.add_job(None, 'hello_world', data={'name':'oh, mighty Strongsteam user'}) # Add job of ty
```

Note: Do not invoke the `cli.add_job(*args)` without storing the result in memory. You use `hello` which is returned by `add_job` to query the status of the job and to extract results when the computation is finished.

CHAPTER

FOUR

THE JOBS WE SUPPORT

Strongsteam being in alpha release, more jobs will be gradually added, as we mature.

4.1 OCR (Optical Character Recognition)

If you want to extract text information from images, you can send us to Strongsteam using the following API call:

```
BlahBlah
```

CHAPTER

FIVE

YOU'RE STUCK? GET HELP.

If you have any questions regarding Strongsteam, do not hesitate to send us an email at help@strongsteam.com.