gits Documentation

Release rst-documentation

January 03, 2017

Contents

Gits is a web-based terminal emulator. The project is based on completely reworked source code of Ajaxterm. It understands only Linux console escape and control sequences so far. The main goal of the project is to be used in OpenStack one day.

Installation

Gits consists of two parts: a client and a server. The following sections describe how to install Gits automatically and manually. You will need `npm` to get the client package or build it from the source code. If you don't have `npm` installed, have a look at nvm.

Automatic installation

The Gits client is listed in npm search and can be installed with `npm`. For example:

```
npm install gits-client
```

The `node_modules` directory will be created inside the current working directory.

The Gits server is listed in PyPI and can be installed with `pip` or `easy_install`.

First, (*optionally*) prepare a virtualenv:

```
virtualenv -p python3 gits-env
. gits-env/bin/activate
```

Then, install the server:

```
pip install gits
```

Finally, go to the directory where you executed `npm install gits-client` and run `server.py`. The server tries to use the `node_modules/gits-client/static` and `node_modules/gits-client/templates` directories by default. Also you can explicitly specify which directories should be used through the `--static-path` and `--templates-path` parameters.

Manual installation

First, get the Gits source code:

```
git clone https://github.com/tolstoyevsky/gits.git
cd gits
```

Then, build the client:

```
npm install
npm run start
```

Next, (*optionally*) prepare a virtualenv:

```
virtualenv -p python3 gits-env
. gits-env/bin/activate
```

After that, intall the server:

```
python setup.py install
```

Finally, run the server:

```
server.py --static-path=static --templates-path=templates
```

or

```
./bin/server.py --static-path=static --templates-path=templates
```

As previously mentioned in *Automatic installation*, the server tries to use the `node_modules/gits-client/static` and `node_modules/gits-client/templates` directories by default. In this case they don't exist, so you have to explicitly specify which directories should be used through the `--static-path` and `--templates-path` parameters.

Prerequisites

Gits uses

- Tornado to create a WebSocket server and multiplex input/output in a platform-independent way

- PyYAML to store escape and control sequences in a YAML file.

Platforms

Theoretically Gits is platform-independent software (generally because of using Tornado), but practically the quality of its work may vary from platform to platform.

Licensing

Gits is available under the [Apache License, Version 2.0](#).