

Contents

INTRO	1
DEPENDENCIES	1
INSTALLATION	2
FILES	2
DATABASE	2
WEBSERVER	3

INTRO

Emanta is a part of SIMBiOMS software package. Emanta web application is meant to ease the task of managing multiple projects under AIMS, SIMS and/or SAIL.

Naturally it can keep track and store information regarding projects that don't use any of the above, too.

Emanta's been coded in Technology Centre of Finnish Institute of Molecular Medicine.

DEPENDENCIES

Emanta is written in python, which you have to have in your server in order to run Emanta. Please get the latest version from <http://www.python.org>.

DON'T USE PYTHON 3, SINCE CURRENTLY DJANGO IS NOT COMPATIBLE!!!

You also need django framework, which is available from <http://www.djangoproject.com/download/>

Emanta has been build and run with version 1.2.1 (stable), but it should work with later version, or even the nightly build. Whether it's advisable to use such in production environment, is up to you to decide.

By default Emanta also assumes that you have django_extensions installed, available from <http://code.google.com/p/django-command-extensions/>

While under development, we also use South database migration tool, which should be installed (at least during the installation)

For database you need the database and appropriate python library. Django supports MySQL, PostgreSQL, sqlalchemy and Oracle. Since both AIMS and SIMS use PostgreSQL and

SAIL uses MySQL, if you plan to use any of them you do need to install the corresponding python connection libraries:

mysql <http://mysql-python.sourceforge.net/>
postgresql <http://initd.org/projects/psycopg>

Emanta was developed using same PostgreSQL database where AIMS and SIMS data resided but you're free to configure your own solution.

Finally you need a web server with wsgi compatibility. Emanta has been tested and developed under both Apache 2 with mod_wsgi and nginx fronting for gunicorn wsgi server.

INSTALLATION

FILES

Copy the emanta directory tree to a suitable place on your web server. For Apache 2 the web server process has to have suitable rights to run the scripts and read the files. If you use gunicorn (or any other wsgi server solution) that is not required.

DATABASE

Open settings.py in the emanta top directory and fill in proper values for the 'default' database. Then run the command `manage.py syncdb` in the emanta root directory. You should see something like this:

```
Syncing...
Creating table auth_permission
Creating table auth_group_permissions
Creating table auth_group
....
```

Not synced (use migrations):

- aims
- projects

(use `./manage.py migrate` to migrate these)

[It will probably ask you to create a superuser in about the middle of synchronization, you should provide proper credentials for this account.

Then, as the commands prompts you to do, you must currently run south's migration to create the final projects table on the database.

```
manage.py migrate
```

You should now have a completely empty emanta database on the database server you configured in the settings.py

WEBSERVER

- APACHE 2

In your apache configuration you have to create an Alias directive to point to the 'media' folder of the Emanta directory tree.

Then you have to create a WSGIScriptAlias pointing to the wsgi.py script on the top level of the Emanta directory tree.

Install and/or enable the mod_wsgi module of Apache and restart or reload the server.

For (a very simplistic) example:

```
WSGIScriptAlias / /applications/emanta/wsgi.py
<Directory /applications/emanta>
    Order allow,deny
    Allow from all
</Directory>

Alias /media/ /applications/emanta/media/
```

- NGINX + WSGI HTTP server

Configure nginx to serve the static files from media subdirectory, and create an upstream server and then point it to serve the content from the wsgi server.

Again a simple configuration example:

```
upstream emantaserver {
    server 127.0.0.1:8091 fail_timeout=0;
}

server {
    listen 80 default;
    server_name localhost;

    access_log /var/log/nginx/localhost.access.log;

    location /media {
        root /applications/emanta/media;
        autoindex on;
    }

    location / {
        proxy_set_header X-Forwarded-For
        $proxy_add_x_forwarded_for;
        proxy_set_header Host $http_host;
        proxy_redirect off;

        if (!-f $request_filename) {
            rewrite ^(.*)/emanta/(.*)$ $1/$2 break;
            proxy_pass http://emanta-server;
            break;
        }
    }
}
```

```
}
```

Reload/restart the nginx server and launch the wsgi server set to port 8091 (or any other port you wish to use, just use the same port for nginx configuration).

NOTICE: both example configurations have been tested, but as they appear here, please refer to respective documentation since configuring either server is beyond the scope of this readme file.