Biodjango, an open framework for bioinformatics publishing

Ennys Gheyouche and ***Stéphane Téletchéa***

Team Protein Design In Silico, UFIP, UMR 6286 CNRS, Université de Nantes
stephane.teletchea@univ-nantes.fr

GT MASIM, Paris, November 16th 2017

Biodjango

(1)A lot of bioinformatics publications propose new methods

(2)Most of the time their usage is limited by user knowledge in web development

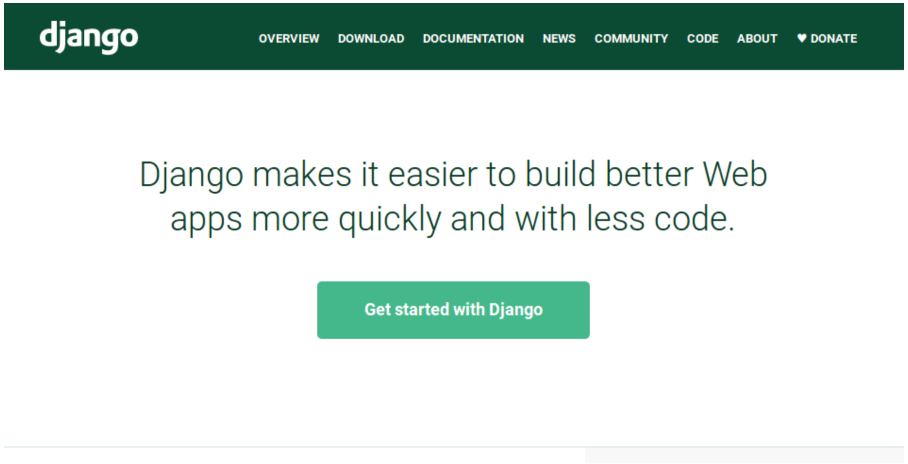
(3)It is difficult to set up a web site from scratch

(4)Mature technologies are available and robust

(5)Python is important in computer science

(6)Django is mature and written in python ...

Biodjango : do not reinvent the wheel ...



Django makes it easier to build better Web apps more quickly and with less code.

Biopython

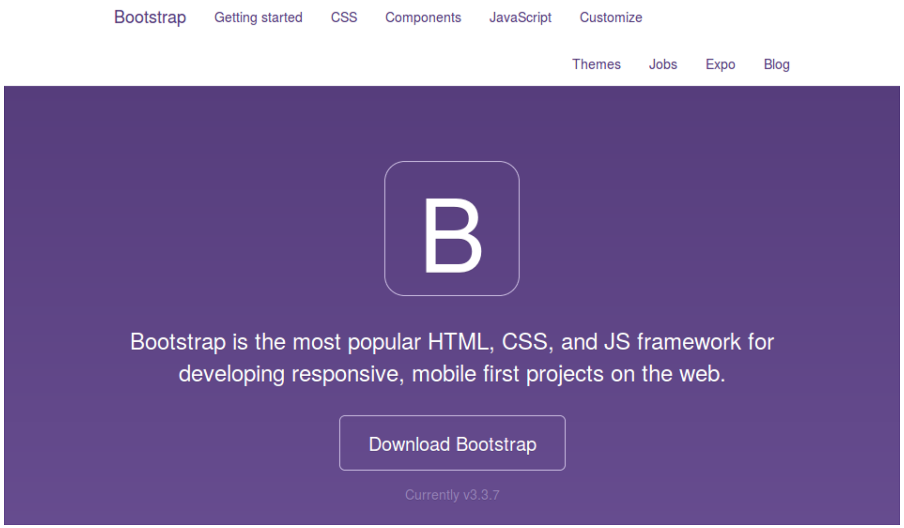
See also our News feed and Twitter.

Introduction

Biopython is a set of freely available tools for biological computation written in Python by an international team of developers.

It is a distributed collaborative effort to develop Python libraries and applications which address the needs of current and future work in bioinformatics. The source code is made available under the Biopython License, which is extremely liberal and compatible with almost every license in the world.

We are a member project of the Open Bioinformatics Foundation (OBF), who take care of our domain name and hosting for our



Bootstrap is the most popular HTML, CSS, and JS framework for developing responsive, mobile first projects on the web.

Currently v3.3.7

https://www.djangoproject.com/

http://biopython.org/wiki/Biopython

http://getbootstrap.com/

https://www.w3schools.com/html/html5_intro.asp

UNIVERSITÉ DE NANTES

Demo : job management

- To ease job management, an example is presented
  - Based on celery (integrated in django)
  - Uses redis or rabbitmq for job execution
  - Celery-beat: cron tasks made easy
  - Javascript example for integration with django
  - Sample files for simple docking

http://ufip.univ-nantes.fr/tools/biodjango (ask for access ...)

Biodjango live: register and activate your account

Django demonstration

Use this document as a way to quickly start any new project.
All you get is this text and a mostly barebones HTML document.

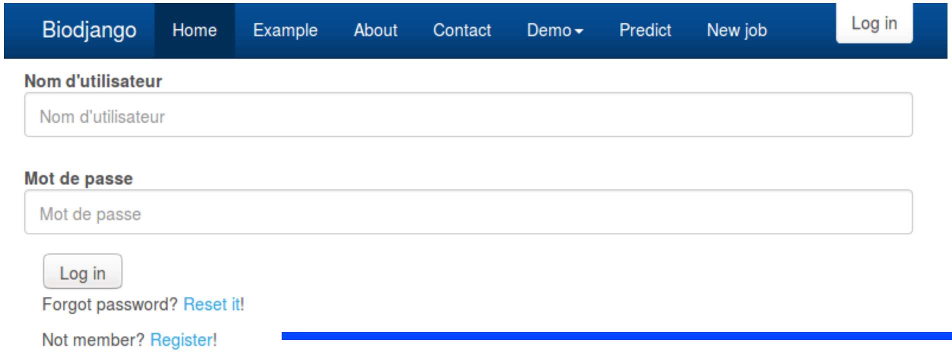
This section contains documentation blocks on all pages for demonstration purpose only. To remove documentation content in all pages, only remove in this template file the following django block:

```
<% block documentation %> ... <% endblock %>
```

Header : menus, registration

Short description (and theme)

Documentation (development only)



Nom d'utilisateur

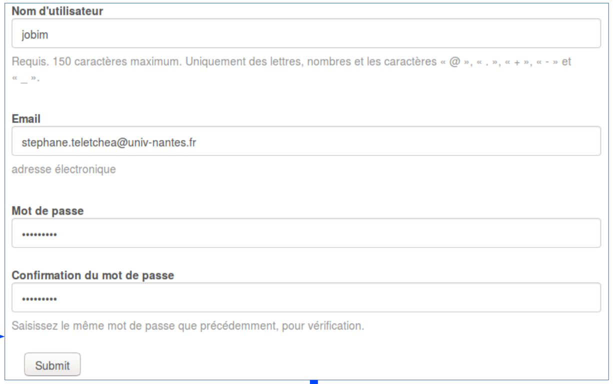
Nom d'utilisateur

Mot de passe

Mot de passe

Forgot password? Reset it!

Not member? Register!



Nom d'utilisateur

jobim

Requis. 150 caractères maximum. Uniquement des lettres, nombres et les caractères « @ », « . », « + », « - » et « _ ».

Email

stephane.teletchea@univ-nantes.fr

adresse électronique

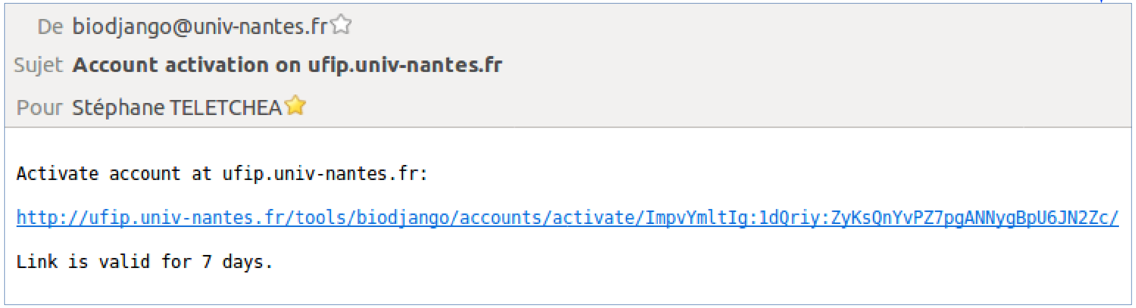
Mot de passe

•••••••••

Confirmation du mot de passe

•••••••••

Saisissez le même mot de passe que précédemment, pour vérification.



De biodjango@univ-nantes.fr

Sujet **Account activation on ufip.univ-nantes.fr**

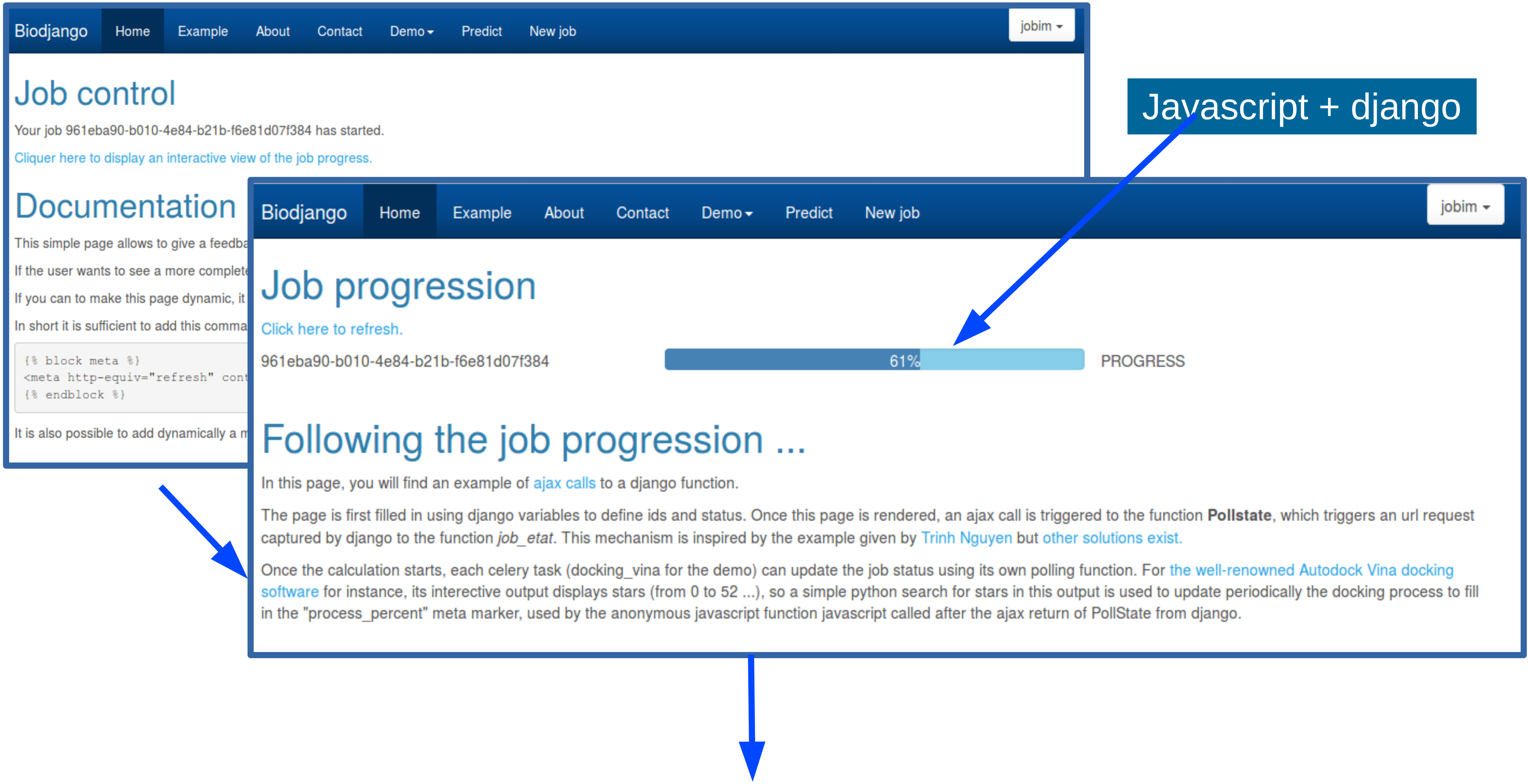
Pour Stéphane TELETCHEA

```
Activate account at ufip.univ-nantes.fr:

http://ufip.univ-nantes.fr/tools/biodjango/accounts/activate/ImpvYmltIg:1dQriy:ZyKsQnYvPZ7pgANNygBpU6JN2Zc/

Link is valid for 7 days.
```

UNIVERSITÉ DE NANTES

Biodjango: job management

Javascript + django



Job control

Your job 961eba90-b010-4e84-b21b-f6e81d07f384 has started.

Cliquer here to display an interactive view of the job progress.

Documentation

This simple page allows to give a feedba

If the user wants to see a more complete

If you can to make this page dynamic, it

In short it is sufficient to add this comma

```
{% block meta %}
<meta http-equiv="refresh" cont
{% endblock %}
```

It is also possible to add dynamically a m

Job progression

Click here to refresh.

961eba90-b010-4e84-b21b-f6e81d07f384 61% PROGRESS

Following the job progression ...

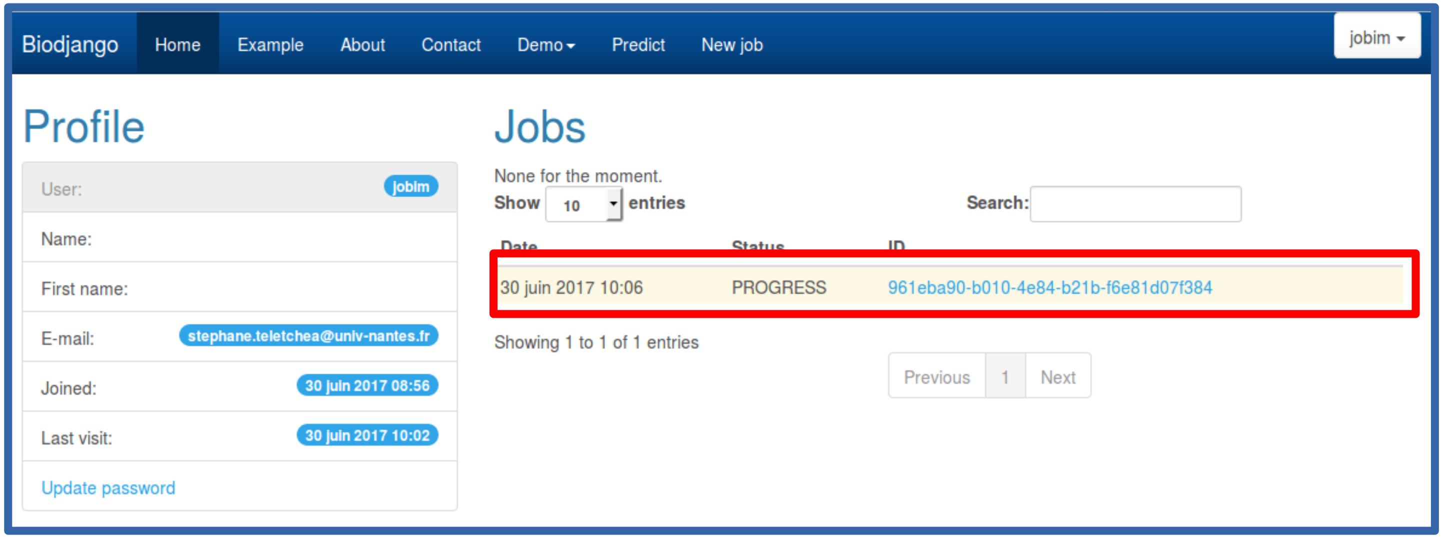
In this page, you will find an example of ajax calls to a django function.

The page is first filled in using django variables to define ids and status. Once this page is rendered, an ajax call is triggered to the function **Pollstate**, which triggers an url request captured by django to the function *job_etat*. This mechanism is inspired by the example given by Trinh Nguyen but other solutions exist.

Once the calculation starts, each celery task (docking_vina for the demo) can update the job status using its own polling function. For the well-renowned Autodock Vina docking software for instance, its interective output displays stars (from 0 to 52 ...), so a simple python search for stars in this output is used to update periodically the docking process to fill in the "process_percent" meta marker, used by the anonymous javascript function javascript called after the ajax return of PollState from django.

```
[2017-06-30 10:06:39,920: INFO/MainProcess] Received task: job_management.tasks.docking_vina[961eba90-b010-4e84-b21b-f6e81d07f384]
[2017-06-30 10:07:43,295: INFO/PoolWorker-13] Task job_management.tasks.docking_vina[961eba90-b010-4e84-b21b-f6e81d07f384] succeeded in 63.373209678s: None
```

Biodjango: job management



Profile

User: jobim

Name:

First name:

E-mail: stephane.teletchea@univ-nantes.fr

Joined: 30 juin 2017 08:56

Last visit: 30 juin 2017 10:02

Update password

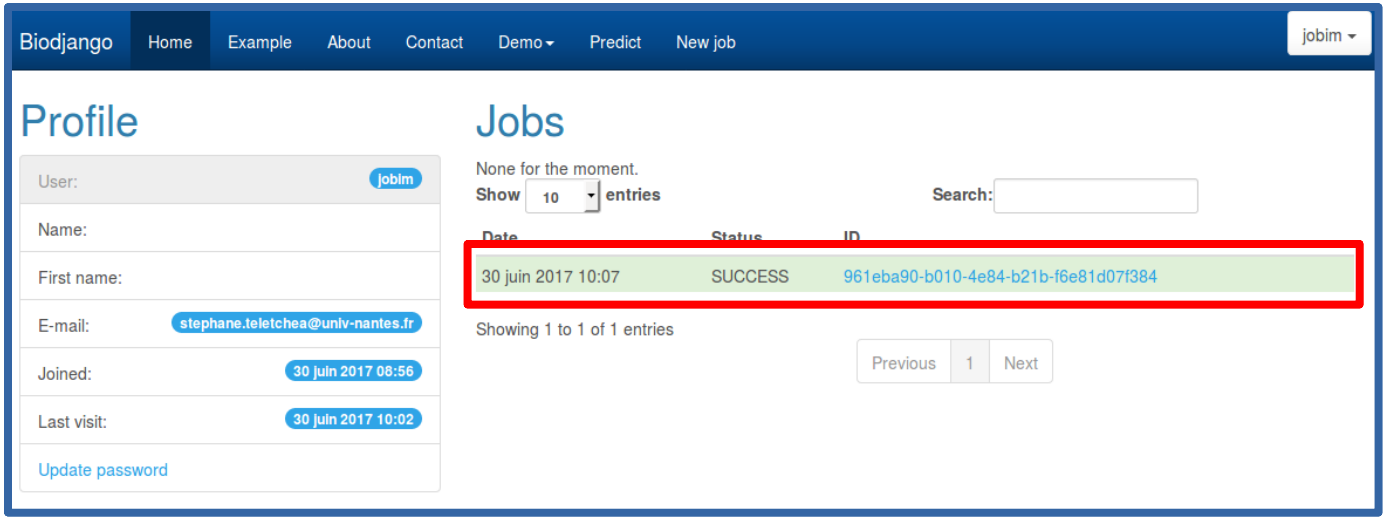
Jobs

None for the moment.

Show 10 **entries** **Search:**

Date	Status	ID
30 juin 2017 10:06	PROGRESS	961eba90-b010-4e84-b21b-f6e81d07f384

Showing 1 to 1 of 1 entries



Profile

User: jobim

Name:

First name:

E-mail: stephane.teletchea@univ-nantes.fr

Joined: 30 juin 2017 08:56

Last visit: 30 juin 2017 10:02

Update password

Jobs

None for the moment.

Show 10 **entries** **Search:**

Date	Status	ID
30 juin 2017 10:07	SUCCESS	961eba90-b010-4e84-b21b-f6e81d07f384

Showing 1 to 1 of 1 entries

Biodjango: python, django and celery

```
url(r'^launch',job_launch, name="job_launch"),
```

job_management/views.py :

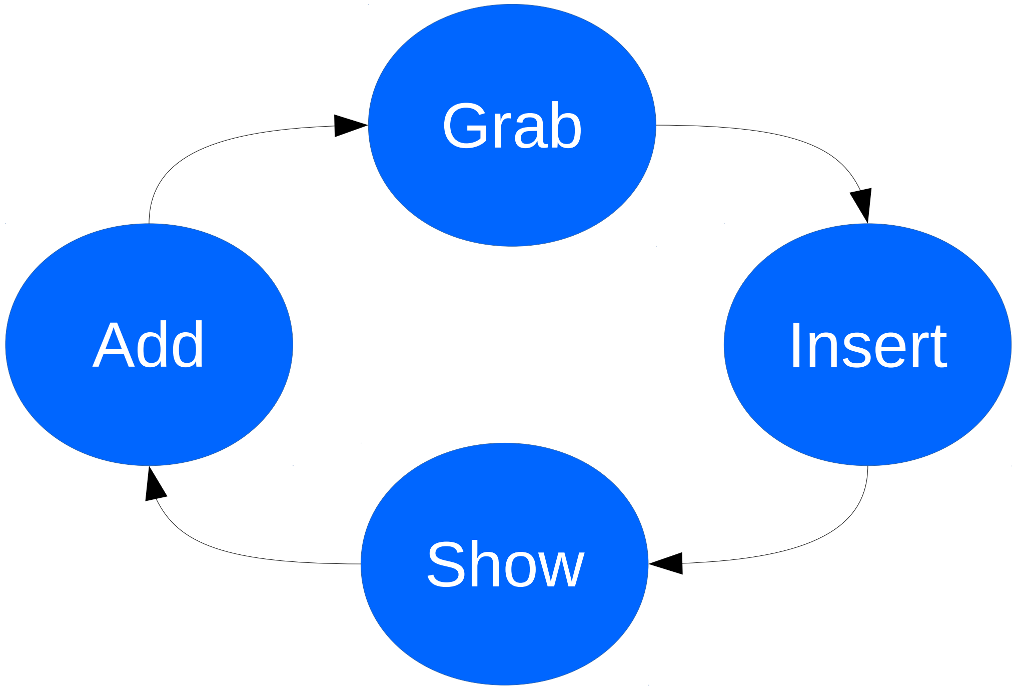
```
@login_required()
def job_launch(request):
    j=docking_vina.delay('ligand','receptor')
    task_result = TaskResult(task_id=j.task_id)
    task_result.meta='author_id:'+str(request.user.id)
    task_result.save()
    return render(request,'job_launch.html',{'tache':j})
```

job_management/tasks.py :

```
@task(bind=True)
def docking_vina(self,ligand,receptor):
    tmp=tempfile.mkdtemp(prefix='docking', dir=settings.MEDIA_ROOT+'tmp')
    f=open(tmp+'/progression.log','w')
    os.chdir(tmp)
    #print "Le chemin par defaut est {}".format(tmp)
    shutil.copy(settings.DATADIR+'receptor.pdbqt',tmp)
    shutil.copy(settings.DATADIR+'ligand.pdbqt',tmp)
    dockstring=settings.BINDIR+'vina --receptor '
                    +tmp+'/receptor.pdbqt --ligand '
                    +tmp+'/ligand.pdbqt --center_x -30 --center_y 12
                    --center_z 7 --size_x 18 --size_y 13 --size_z 10 --cpu 1
                    --log '+tmp+'/vina_docking.log --exhaustiveness 16'
```

Biodjango : functional annotations

- Uniprot entries for proteins: done
- PDB entries: done
- PFAM annotations: in progress
- CATH/SCOP families: in progress
- References: done
- ...
- More? As required ...



Integration of existing apps

- Just add one application :
  - python manage.py startapp mynewapp
  - Edit biodjango/settings.py to add mynewapp
  - python manage.py makemigrations
  - python manage.py migrate
- Use python function in tasks.py to launch the job, celery automatically processes it
- Provide a progress-bar if wanted (**your own one!**)
- Everything is stored in the database

Biodjango

- Everything is in the docs
- Many examples in biodjango
- Very simple for now
- May be merged with waves for larger job management (http://waves.atgc-montpellier.fr/documentation/installation.html)
- May include bioservices for wider function annotations (http://www.pythonhosted.org/bioservices/)

Thank you for your attention

Y-H Sanejouand
B. Offmann

I. André
J. Esque

C. Etchebest

UNIVERSITÉ DE NANTES

INSTITUT NATIONAL
DES SCIENCES
APPLIQUÉES
TOULOUSE