

## **Web-based administration & process management**

Providing operational running processing services requires often a tool to list executed processes with their current status to have an overview about usage and possible errors. Users and administrators could easily view a process log file, if generated from the process, and further information about the executed processes. Furthermore process actions - like an easy way to restart or delete an executed process - could be very helpful for the process management. An administration part could extend PyWPS to make it more user-friendly in creating services and changing settings.

The administration part could include:

- Create services, change settings of services
- Add new processes (e.g., provide a template process for further manual adjustments)
- Change process settings
- User management (?)

The process management part could include:

- List executed processes with their status (e.g., running, failed, succeeded)
- View process details (e.g., inputs, outputs, log file)
- Provide process actions (e.g., restart, delete)
- further ideas are welcome

## **Time schedule from proposal:**

As I stated in the proposal, please understand, that I have last two weeks of the semester in university, then I hope only 2-3 weeks of final, so I am a little bit in time pressure. I am trying to balance workload, so I would like to start with something easier like a web interface. After finals, I am working full time.

**April 22 - May 23 (Before the official coding period)**

- Testing current implementation
- With help of my mentor precisely define all goals and plan steps
- Create subdomain on my website to document my progress

### **May 23 - June 27 (Official coding period starts)**

- Creating with the current implementation user-friendly web interface
- Integrating current functionality into the web interface
- Start to implementing new functions, mainly the process control

### **June 27 - August 8**

- Programming to meet all defined goals
- Writing unit tests in the style of Test Driven Development
- Documenting all code

### **August 8 - August 15**

- Integration tests
- Solving conflicts with tests
- Passing through the pull request process, resolving conflicts on the repository

### **August 15 - August 23 (Final week)**

- Review of code - make sure of code clarity
- Review of documentation