

Practice sheet 5 – due on 2010-11-29

Learning aims:

- Get to know and test communicative and adjustment processes in Open Source projects.
- Get to know Open Source projects und Open Source development processes.

You only have to complete one of the following tasks. Distribution will take place in the tutorial on 2010-11-22. You may work in pairs.

Task 5 – 1 : Submission of the developed JabRef features

On practice sheet 4 you have developed a feature for the Open Source project JabRef. Now present it to the project. Proceed in this way:

1. Include the feedback from the review process (task 4-3) into your feature patch and make sure you have completed the development of the feature. As you have developed in a test-driven manner, you should have enough test cases.
2. Compile a patch you would like to submit to the project (carry out an SVN update on *trunk* first).
3. Write a short English text describing your feature implemented (RequestID, function description, most important part of the API, test description).
4. Think about when to submit the feature, and to whom. Does JabRef give any hints in that respect? Where?
5. Submit the feature (covering letter including description and patch).
6. **Describe your experiences in the next tutorial.** To be able to do so, submit your feature as early as possible (Tuesday or Wednesday).
7. Possibly there is no feedback within the first week. Stay tuned (even longer as the fifth tutorial) and keep on enquiring. In case of feedback from JabRef developers or the maintainer, observe these and continue improving your code.

Practice sheet 5 – due on 2010-11-29

Task 5 – 2: (Discover an Open Source project)

This task gives you an overview of the structure, practices and development process of another Open Source project (so far not dealt with in the lecture or tutorial).

Choose a project fulfilling at least the following conditions: active during the last two months, more than two developers, in a programming language you master, public mailing list, bug tracker and version management available, and interesting to you personally.

Answer the following questions (Attention: it may not always be easy to find answers!):

1. What is the project about?
2. Since when does the project exist?
3. How many participants does the project have?
4. How can you characterize the participants?
5. Who leads the project and in which way?
6. How can you become participants of the project?
7. What kind of communication is there between the project participants and the users?
8. Which communication channels are in use?
9. Is there information on development styles (e.g. code guidelines) or preferred tools?
10. What has been worked on during the last months (SVN logs) and what was the focus of the last release?
11. Where there any changes in the team's configuration? Did someone new arrive or someone stop? Are there any apparent reasons for this?
12. Which defects/bugs have occurred lately? Were these fixed? If so, how fast? If not, are you able to tell why?
13. **Identify and prepare the following numbers and graphics:**
 - a. lines of code over time
 - b. active participants on the mailing list
 - c. number of bugs in the last month
 - d. ratio of open and closed bug reports
 - e. traffic on the mailing list/emails per month
14. Finally, briefly characterize the development process, the release process and the quality assurance process of the project.