

# Group Assignment – Agile Development Processes 2012

---

The following assignment is mandatory in the course Agile Development Processes, EDA397 held at Chalmers University of Technology. The submissions will be in the form of continuous weekly acceptance tests with your customer and a post mortem report.

**Description:** This is a graded and mandatory assignment that shall be solved and reported together with your assigned group.

## **Deliverables:**

1. Weekly deliveries of completed stories in the application to your customer
2. Documents from daily scrum, retrospective and reviews
3. Postmortem – Written report

## **Grading:**

The project is graded on 100 points. In order to pass the project part you need to accumulate at least 50 points. If you receive more than 50 points, i.e. 50 to 100 points, you will receive bonus points for the course written exam. However, these bonus points cannot be used towards a passing grade, i.e. to upgrade your exam point from fail to 3 for Chalmers students or from fail to G for GU students. In order to pass the course you need to have a passing grade in both the project and the written exam.

The project's 100 points are divided as follows:

- 30 points for your personal performance in the group that will be graded by your supervisor/customer and your team members.
- 15 points on your frequent submissions to the project. You are expected to have at least two correct and meaningful submissions each week where you have added new functionality or fixed a serious bug in the system. Failing to do so will decrease your points based on the extent you have missed this. In each sprint you will receive a maximum of 3 points.
- 30 points on how well your group and you personally have followed the process and the practices proposed in the project description. This is decided by your supervisors and customers continuously during the project.
- 25 points on your analysis in the post mortem report. This is based on the completeness of your report and the depth and breadth of your analysis of the project, delivered product and methods used.

## **Description of assignment and practices**

Senior researchers at the Software Engineering division (SEDiv) at Chalmers and GU has a long-term strategy to develop an open-source framework that help software developing organizations in understanding their software development. The framework is called KnowSE (as in "Know your SE!") and its slogan is "From raw data to organizational change!" The basic premise is that visualization of

real, objective data from software development projects, i.e. including both product, process and organizational data, can foster discussions and understanding that ultimately leads to changes. This helps start positive feedback loops of learning, change and improvement. But the key difference compared to previous metrics systems is that this is based more on visualization and interactivity and not simply on metrics. Organizational change happens only when individuals change. Individuals change only when they understand the problem with their current ways of working. Understanding happens when one can clearly see and connect data in their context.

Based on this vision we are currently working with three different companies in the Gothenburg region to prototype the framework and the type of collection, statistics and visualization that can be effective for them. The work is in its infancy and the KnowSE system is far from stable. However, already a few months into the project it is clear that a mobile client solution is needed. Since the core prototyping and development of KnowSE is focused on the data collection probes, the KnowSE analysis server and a simple web presentation layer there are currently no resources left to develop a mobile application. Thus you have been assigned this task.

Thus your group should use agile software development methods to develop a mobile KnowSE client application that securely accesses SE data available on a server (accessible in the form of CSV and JSON files) and then presents key data and visualizations on Android and/or iPhone mobile phones. The application you develop needs to be highly configurable since different roles and individuals that will use it have very different information needs. Configuration can happen both through JSON files available on the server and via the application interface. The application should also support alerts on specific changes in the data/analyses. It is essential that the applications are easy to use with a focus on usability and meaningful interactivity.

The customer has asked you to follow certain development processes and practices, which will be introduced below.

### **Task 1 – Logging**

During all of the work you do on this project you must log how much time you spend on each activity and which person(s) in your group does what. Your log should be added as an appendix to your post mortem report described below. Furthermore, the scrum master should have a report for each week specifying how much each person has worked and on what task. This report should be submitted to the GIT repository in a directory called “process” prior to the start of next sprint.

In your submissions, the name of all the people working on that submission should be specified in the commit comment to the GIT repository. This should be done in the following format at the end of your comment:

1\_Developers: [name\_Developer1], [name\_Developer2]

2\_Time: [Ideal\_time\_HH:MM]

3\_Use\_case: [UserStory\_Nr\_from\_Pivotal], [Name\_of\_US]

### Task 2 – Scrum

The main process that you are to follow in this project is Scrum. You will assign a unique scrum master in the start of each sprint who documents the progress of your project in a backlog, which will be shared with your customer through your repository.

In the start of each sprint you will have a sprint planning meeting. Furthermore, in the beginning of each development day you will have a short stand-up meeting.

Pivotal Tracker is the software to be used for management of your project. The software can be accessed from <http://www.pivotaltracker.com/>. At workshop 2 you will see a demo on the tool. You can get some information about how to use Pivotal Tracker here: <http://www.slideshare.net/danpivotal/pivotal-tracker-overview>

Every week you should export your project from Pivotal Tracker and submit it to a directory called “process” in your project repository. The time log and other documents should also be kept updated in this directory. Your project logging and reporting will be sampled and assessed continuously during the project so ensure the repository is always updated. Furthermore, you should submit your notes from every scrum meeting, retrospective and review. These will be submitted to the “process” directory in your repository.

### Task 3 – Group and Pair programming

Your coding has to be done in pairs when instructed. In that case it is important that you never work alone and always in pairs. However, the time you spend working should be logged individually. Hence, if you pair program for 4 hours you both should log 4 hour of work!

It is also important that everyone in the group, i.e. all pair programming teams, all work at the same location at the same time. This is related to the praxis of team development in Scrum. If all of you are not there you are not allowed to work on the project. You can change the schedule in a way that all of you can be there. If there is a last minute emergency like sickness or personal emergency you should let the supervisors know. **Planned travels, vacations, other course obligations, etc, are not valid excuses not to attend the group work.**

### Task 4 – Weekly acceptance tests

In the end of each sprint, your team should show the results from that sprint to the customers and receive new user stories if needed. Your team will be allocated a slot on Fridays when at least one person (the scrum master) will meet the customers and show the results. Make sure you are there in time since we have many groups and cannot be flexible with the time.

### Task 5 – Automated unit

One of the most important practices you need to follow is complete automated unit and acceptance tests. You are allowed to use any framework for unit testing that you find suitable for the task (JUnit being the natural choice in Java development).

You should use Test-Driven Development if you can. This is not mandatory but will give you an extra star in the corner for effort.

### Task 6 – Continuous integration

Everything you do should be continuously checked in to your project repository. We expect at least two updates from each member of the group with a proper description on what has been changed or added. This will be considered as a resource for grading your effort. Every night your system will be automatically compiled and should be working and free of any bugs. To submit a buggy version will result in reduction of your weekly point for your submissions.

### Task 7 – Postmortem report

You should conduct a post mortem analysis of the project of your group assignment. The results from the post mortem meeting should be reported in a pdf file not longer than 10 pages excluding the appendix and uploaded to your repository.

1. Which processes and practices did you use in your project? Show all the steps in time, describe what they constituted.
2. Based on the process description from question 1 and the detailed logging information you should summarize how much time was spent (in total and by each group member) on the steps/activities involved as well as for the project as a whole?
3. For each of the techniques and practices used in your project you should answer all the questions:
  - a. What was the advantage of this technique based on your experience in this assignment?
  - b. What was the disadvantage of this technique based on your experience in this assignment?
  - c. How efficient was the technique given the time it took to use?
  - d. In which situations would you use this technique in a future project?
  - e. In which situations would you not use this technique in a future project?
  - f. If you had the practice/technique in a part of the project and not the entire project, how was using it compared to not using it?
4. For your next project which set of techniques (that you used here or that you have not used here but know from theory or other projects) would you use? Why? Clearly motivate your selected set of techniques and discuss how they complement each other. How much time is needed to effectively use the technique?
5. Compared to other projects using a more plan-driven/waterfall process what were the benefits and drawbacks with the process used in this project?
6. What worked well in how you worked in this project?
7. What did not work well in how you worked in this project?
8. How did you work together as a group in the project? What worked and not in your interaction(s)?
9. What would you do differently in a future but similar project?

## Rules of conduct

This assignment is to be performed by and within your group. Be advised that any form of plagiarism will render an immediate fail on the course, and the matter will be reported to the university disciplinary board. Everything you hand in must be written by your group or properly quoted. Please note that you are responsible for everything that bears your name no matter if it is a group assignment or an individual assignment.

## Excuses we have already heard

The following is a list of excuses for not referencing material properly that we have already heard, and will not accept. This is not a complete list of excuses - you may very well come up with a new and innovative excuse that we have not heard but is equally invalid.

Excuse 1: "I did not know that I had to add a reference."

Answer. Anything that you have not written or thought up yourself must be referenced.

Excuse 2: "I added the reference at the end of the paragraph that I copied."

Answer. Your fault here is that you copied a sentence, paragraph or a whole section word by word from another source. This is called a quote, which you surround by quotation marks (a " and a "). You should try to avoid using quotes. We are much more interested to see your interpretation of what you have read and/or heard (with proper referencing to the inventors of the ideas that you present).

Excuse 3: "The reference [to the copied paragraph] is in the section heading."

Answer. This is not only as wrong as the aforementioned, it is also ugly.

Excuse 4: "The reference [to the copied paragraph] is in the reference list."

Answer. This is basically another variant of excuses 2 and 3. As soon as you use the text or ideas of someone else in your work, you provide a reference so that the readers can understand that they are no longer reading your ideas, your thoughts, your labor.

Excuse 5: "In my home country, the teachers do not check this."

Answer: Welcome to Sweden. The teachers in Sweden are by law required to both check for and report suspected cheating to the disciplinary board. Improper referencing is cheating, since it is an attempt to take credit for what someone else has thought or written.

Excuse 6: "I interpreted it as an exercise in forming a report, not actually writing it."

Answer. When we read an assignment we do so in order to provide the best possible feedback to you as a student so that you may learn something. To hand in text that you have not written yourself is not only depriving yourself a valuable opportunity for feedback and personal improvement, it is also nonchalant towards your fellow students that could have made better use of the teacher resources.