

Group Assignment 1: Requirements

1 Background

Over the course of the year, you will be working in teams in the setting of an agile development project. **In this semester you will be eliciting the requirements of a system, which you will be building in semester two.** Each team is interacting with a client (represented by 4th year Genesys students), and it is your job to elicit requirements (in Autumn) and build a software product (in Spring) to satisfy your client, using an Agile approach. We will not be enforcing a specific Agile methodology for this (but if you want to apply e.g. SCRUM or Kanban you are free to do so).

2 Requirements Capture Sessions

In the autumn semester, you will be meeting with your clients to elicit and elaborate requirements for your software project. Some general advice for these meetings:

- For all meetings, prepare an agenda, and introduce the agenda at the beginning of the meeting. The client is not going to run the meeting for you, you have to run the meeting!
- Make notes — decide on a note keeper before the meeting. Even if you think you have a perfect memory, as time passes memory blurs, and different people will have different recollection of discussions.
- Following the meeting, write up and circulate minutes of the meeting within your team. All team members should read and check these minutes for accuracy, and ask for corrections if necessary.

The main result of the three meetings will be user stories. For example, you could use the format:

As a <type of user>,
I want <some goal>
so that <some reason>.

Write user stories that are easy to understand. Keep them simple and concise. Focus on what's important, and leave out non-essential information.

2.1 Collecting user stories

1. Start by introducing yourselves.
2. Identify the scope of the project.
3. Identify the stakeholders and users from whose point of view you will be writing user stories.
4. Brainstorm to generate stories. The goal is to write as many stories as possible. (You will receive index cards for this)

It is fine to start with epics and iterate. Starting with epics allows you to sketch the product functionality without committing to the details. This way you can capture the rough scope, and it buys you time to learn more about the users and how to best meet their needs. If you have lots of detailed stories, then it may be more difficult to relate any feedback to the right stories.

Break your epics into smaller, detailed stories until they are ready: clear, feasible, and testable. Everyone should have a shared understanding of the story's meaning; the story should not be too big, and there has to be an effective way to determine if the story is done.

2.2 Acceptance criteria and user story refinement

- Acceptance criteria complement the story's narrative: They allow you to describe the conditions that have to be fulfilled so that the story is done. The criteria enrich the story and make it more precise and testable, and they ensure that the story can be demoed or released to the users and the other stakeholders.
- Try to identify non-functional properties or constraints. Are there requirements on performance, quality, usability, etc.? Are these global constraints on the project, or are they rather acceptance criteria for some of the user stories? There can also be non-functional user stories.

Good user stories should satisfy the INVEST criteria discussed in the lecture.

2.3 Estimation and prioritisation - do not forget about it

Collate your user stories based on the results of the second meeting. Independently of your client, you will have to estimate the effort for every user story (this does not have to be done in the meeting). Without experience in implementing such user stories, your estimates may be way off — that is perfectly fine. You will be refining your estimates and adapting them to your team's velocity over the sprints that will follow in the spring semester. Do not estimate the effort in person-hours; we are rather interested in the *relative* difficulty of each user story. Decide on a set of values that you will use for estimates; common choices are user story points with the Fibonacci sequence of 1, 2, 3, 5, 8, and 13, or powers of two (1, 2, 4, 8, and 16). A common and fun way to derive these estimates is to use planning poker, as discussed in lecture 8-2.

In the last meeting with your client, you have another chance to refine your user stories. For example, you might discover more epics while estimating user stories, or you might require more acceptance criteria to clarify a user story. If you and your client are both happy with the user stories, the stories have to be prioritised by the client. The aim is to rank the stories by their priority, and to do so you could use a card shuffling procedure: Pick a story and determine with the client whether it is high or low priority, then take the next story and decide whether it is higher or lower than the one previous, and continue with the remaining, relative to each other. This method facilitates a discussion with customers which can be very beneficial to the project.

3 Requirements Document

The result of the requirements capture phase is a requirements document summarising the user stories, acceptance criteria, and constraints. The document should contain the following parts:

- The group number and the names of all team members.
- A (brief) explanation of the scope of the project.
- Identification of the stakeholders of the project.
- The main component of your document is the collection of requirements represented as user stories. Each user story should consist of:
 - A short textual description
 - Acceptance criteria
 - Client priority (ranking position)
 - Estimated effort (e.g. story points)
- If there are general constraints or non-functional requirements, include these as well. This does not have to be in terms of user stories (but may be, if you prefer).

4 Marking criteria

Each team is assessed according to the following criteria; the weighting of these criteria is described in the rubric on Blackboard.

Story description Stories should not be epic or too small, all of them have to have a value with user roles clearly defined.

Acceptance criteria These have to be complete and testable.

Process This reflects how a team organised meetings, how good estimation was and whether your customers have been guided through requirements elicitation and prioritisation.

Customer satisfaction Whether there was a clear contribution of the team above customer's own ideas and whether everything intended by the customer has been reflected in the story cards.

The above criteria are applied to the team as a whole, resulting in a team score. Where the contribution of all team members is the same, they all get the same scores; otherwise, each individual contribution is equal to the team score multiplied by the scaling factor from WebPA described below.

5 Assignment Submission

Please submit the requirements document as a PDF document via MOLE (by clicking on "Group Assignment 1: Requirements"). The submission is per-group so it is enough if a single member of your team submits the document, just make sure they are not submitting late. If multiple submissions are made by the team, the last one will be marked and will attract standard late submission penalties if it is late.

In addition, each team member has to evaluate the contributions of all team members using WebPA. Shortly after the submission deadline passes you will find a link to WebPA on the module's MOLE page ("Group Assignment 1: Peer Assessment"). Using the WebPA link, you will have to rate yourself and your team members on the following criteria:

1. Attendance and punctuality (to group meetings, sessions with clients, etc.)
2. Ability to work effectively with other team members
3. Contribution to content and organisation of project deliverables
4. Quality of contributions
5. Timeliness of contributions

You will only be able to submit your ratings once and you will not be able to change them later.

The peer assessment determines a scaling factor that is applied to the overall group mark. That is, if you contributed more, then you will get a higher mark, if you contributed less, you will get a lower mark. You can read about WebPA at

<http://webpaproject.lboro.ac.uk/academic-guidance/a-worked-example-of-the-scoring-algorithm>

In your case the PA Weighting will be 30%, that is, 30% of your grade will be affected by WebPA.

Students who did not attend and/or did not participate in teamwork will get a zero for the whole assignment.

The deadline for submission is

Friday, December 4, 15:00

This will count for one third of the autumn semester contribution to the grade of the module.