Weekly Lab Meeting 2 - Agenda

*Notes on formatting:*

*Proposed places for the minute-taker to record the minutes during the meeting are noted in italics, between square brackets [].*

*The actual minutes don’t have to be in square brackets, but they should keep the italic styling, so that they are easy to distinguish.*

*The minutes don’t have to be a word-by-word exact record of what was said, they can be a quick and concise summary of the decisions taken by the team and the answers, feedback and suggestions given by the TA.*

Location: Drebbleweg, PC Hall 2, front-left

Date: Tuesday, 21 January 2023

Time: 14:45

*Start time was 14:45*

Attendees:

* Team 69: Agnese Ēlerte, Georgi Dimitrov, Dimana Stoyanova, Matei Bordea, Pedro Gomes Moreira, Veselin Mitev
* TA: Alexandra Ioana Neagu

*Everyone attended the meeting*

Agenda Items

1. Opening by chair
2. Check-in
   1. make sure everyone has a copy of the agenda open
   2. notify the TA that we have no Mattermost channel

*Mattermost was created today*

1. Approval of the agenda
   1. Is the format of this agenda correct?  
      *The format is good, TA responded that only important points should be included. It’s okay to have the minutes written right after the question.*
   2. Does anyone have any additions/changes to propose for this agenda?  
      *No changes needed*
2. How should we upload documents like the Code of Conduct, Backlog and the Agendas and Minutes?
   1. Should we write the minutes on top of the agenda?  
      *The question-answer file (the one with the minutes) can be reuploaded on top of the agenda (that is on GitLab) after the meeting.*
   2. When is the GitLab going to be open? Should we upload the documents there?  
      *Once it is open we should upload the agenda and the minutes there. The code of conduct and the backlog should also be uploaded there.*
   3. (How) should we upload changes to the documents? Which version of the code of conduct is graded?  
      *Upload the code of conduct on GitLab and if the update the code of conduct we should update it and there is no need to tell the TA about that.*

*Markdown files are preferred;*

*This agenda should be uploaded to Git and the current format is okay.*

1. How should the other agenda members (not the Chair) participate in the meetings if most of it is asking questions that we have prepared beforehand that the Chair asks?  
     *If she is giving us feedback then we all should share opinions. Otherwise, the chair is the one that will be asking all of the questions.*
2. Present our Code of Conduct  
   *[Record feedback from TA]*
   1. Do we have to pick a team name?  
      *No need but would be fun.*
   2. Is our proposed meeting and collaboration schedule good? (Work together and distribute tasks on Tuesday, Short progress meeting on Friday, prepare agenda on Sunday)? Should we prepare the distribution of the tasks for the following week, before the meeting on Tuesday?  
      *The proposed schedule is good, we should add to the code of conduct that sometimes we will do occasional meeting*
3. Present the outline of our Backlog (we will finish the draft at the lab)  
   *[Record feedback from TA]*
   1. Is the backlog pass/fail?  
      *No, it is graded with an actual grade - not just a pass/fail . The draft is not graded.*
   2. How detailed does the draft have to be?  
      *As detailed as possible because more details, the better the feedback will be*
   3. How detailed does the final backlog have to be?  
      *Detailed and if it has too many/ to little details the TA will inform us through the feedback for the draft of the backlog*
   4. Do we need to do Epic > User Story > Acceptance Criteria? What are acceptance criteria?

*We should use the epic -> user story> acceptance criteria*

*Epics are general functionalities*

*User stories as specific uses (ex: add a task)*

*Acceptance criteria is for actual specific features (ex: tasks should appear on this specific page, tasks should be edited from the overview )*

*Must-could-should show priority*

*Another section for non–functional req. (details about language, tools etc.)*

* 1. Do we need to add mocks?  
     *No mocks are needed*
  2. How much should we ask about the requirements?

1. Time management
   1. How should we divide the work? Should we strive for everyone doing the same amount of work, or should we strive for everyone spending the same amount of time on the project? Should we try to balance it?  
      *When dividing the requirements: we should consider the combination of both effort+time and amount of product that is delivered.*
   2. If a team member does more work than others in order to get a better product in the end, does that negatively affect the teamwork assessment?  
      *Teamwise, one member taking significantly more work than others can bring down the grade for the whole work*
   3. How much time are we expected to spend on the project (in order to get a 10)? How much of that is on actual programming?  
      *It will depend on the week. The average would be around 13-14 hour a week (like it is for every course). TA did not specify about what part of the time will be specified coding.*
2. Assessment
   1. What is a realistic grade? Is a 10 realistic? Do a lot of teams get 9s and 10s, or are those reserved only for exceptional teams?  
      *Yes, a grade above 9 is realistic.*
   2. If we do all of the basic and the advanced requirements properly, will we get the maximum amount of points for the Project part of the assessment (40%)?  
      *If we do all of the advanced requirements we will get the full 40%. No need of additional requirements in order to get 40%. It’s more important to fully polish the additional requirements instead of creating new ones that we have thought of.*
   3. Does adding any extra features help get the points for the Project part? Is that expected in order to get a 10?  
      *No. Don’t waste time to add additional features that we think of because we don’t get extra points for that. Adding an extra touch on the mentioned features is needed for a perfect score.*
   4. Testing. Is Unit testing mandatory? Which part of the grade does the unit testing affect? Do we need to do integration testing?  
      *Unit testing is not a ‘must’ but a should requirement which will be graded as a part of the 40%. Testing is mostly unit testing, TA doesn’t believe integration testing is needed. We will probably have a lecture for testing .*
   5. How strict are the Teamwork assessments?  
      *The teamwork assignment: TA is grading that. During the meeting she is observing how active everyone is. She can see the merge requests, the task distribution and the code contribution. Improvements after feedback are graded. Teamwork grading is teambased not individual. Individual adjustments are done but mainly it is teambased assessment. If someone works more that is overall negative for the team.*
   6. Are we supposed to learn stuff specifically for the final examination at the end of the course? Or is it purely an anti-fraud measure, i.e. will we know the answers if we actually did the work like we are supposed to? Is it going to be about the internal workings of the project or about the process? Is it pass/fail?  
      *–*
3. Summary action points
   1. Summarize what the team is going to do after the meeting and get feedback on that from the TA
      1. We will finish the Code of Conduct and the Backlog draft during the lab after the meeting
      2. By the next meeting we will all research and learn (mainly from the resources provided) about the technologies that we are going to use (Git, SpringBoot, JavaFX) and set up our development environments (Setup Git, do the Git assignment, create a SpringBoot JavaFX project to familiarize yourself with the process)
      3. At the next lab we will begin on the actual project: we will create it and start prototyping and managing tasks
4. Question round
   1. Does anyone have any other questions for the TA before the meeting closes? *No other questions were asked*
5. Closure  
   *The meeting ended on time*