CSC 648/848 Spring 2023
Milestone 1: Use cases, High Level Requirements and Architecture

03-01-23

Announce: TBD
Due: check instructor's e-mail and CANVAS

Objective:

Based on the high level team project description (posted on Canvas class site) the objective of Milestone 1 (M1) is to start the design of your class team project using modern Agile SE processes and User Centered approach. Specifically:

- Develop initial high-level personas and use cases;
- From use cases develop high level functional requirements for the application (Do not prioritize yet)
- List high level architecture, frameworks and tools to be used (generally the same as in M0). If it changed list the latest but inform Anthony and ask for approval/OK)
- Incorporate AS-IS (no changes allowed) non-functional specifications as defined in high level team project description posted in Canvas
- Understand your competition
- Learn how to write team reports and SE requirements documents (one report for the whole team, with participation of all team members)
- Solidify and confirm team roles
- And last but not least: bring the team together on same objective and ideas about product being developed and implemented – and have them in writing
- NEW: follow class adopted GenAI (e.g. ChatGPT) policies

Note that these are only early and high-level requirements and specs with the idea to get early feedback and iterate before investing in developing more detailed specs and first prototype in Milestone 2. They can evolve and change with feedback. Future designs can deviate from Milestone 1 and Milestone 2 in the spirit of iterative SW design and development.

Initial input for your work is the team project high-level description (posted on Canvas), class slides on the topics, as well as your SW and tool selection for M0. For use cases and functional specs, feel free to also use your own ideas, research similar applications that already exist, talk to your friends, and participate in class discussion (some of general issues related to M1 will be discussed in the class). Then, after the submission, you will get feedback on your Milestone 1 (from instructor) in all phases of the project and future milestones and you should take these into account.

This is the first **team** milestone. The whole student team submits **one** milestone document for each Milestone 1 – 5. Submission details and M1 document structure are below.

You will discuss ongoing work on Milestone 1 during team sessions in each class (among yourselves and with instructors) and you can also send e-mail to instructors with questions.

Expected size of this document is about 7-10 pages, using font and spacing as in this document.

***** Please wrap up M0 before you start on M1 ******

Content and structure for Milestone 1 document for review:

In the document for Milestone 1 (M1) you must have ALL of the following subsections in exact order as below (have a separate numbered section for each) in one PDF file. **Please read the instructions carefully and follow up on all requirements below in terms of what to submit as well as how to format/organize.**

1.**Executive Summary**: Short description of the final product/application and its key advantages, novelty, value (up to 1 page). Make it as an executive summary to be readable to broad and not just technical audience – think also of answering the question of why we should fund this project. We suggest you assign a name to your project for easier reference and good "marketing". This summary should be readable to a general manger/executive that is not a CS specialist and is used to explain and also to advertise/promote your project. Typical outline is: one paragraph on the motivation and importance of the application you are developing, followed by a paragraph on what functions and services your application will be providing and how it helps the users (high level only, no jargon) and what is unique and special in your design (if anything). At the end say in one paragraph something about your team (e.g. about your student startup team…).

2. **Personae and main Use Cases**: Summarize **key personas** (categories of users) for your application – their general characteristics, goals, skills, pain points related to the application you are developing. Use max 1/3 – ½ of a page per persona – see class notes. Provide one personae for each main category of users, 3-5 total.

Then provide 4-5 main **use cases** (one paragraphs for each use case) - see class notes on more detailed format for requirements. Focus only on main use cases. Simple text format for use cases is OK and preferable – tell us a story about who and how the application is used. Focus on WHAT users do, their skill level, not on HOW is the

SW implemented. If you have an idea of special fucntyions please mention them as part of use cases.

NOTE: avoid specific on HOW functions will be done and text resembling user manual: this is supposed to guide the design of the future product and is NOT a description of how the product will work (you don't know that yet) – see class slides for details. Please assign a descriptive title and number to each use case so it can be tracked. Best is to first list all personae then all use cases, do not mix them up (one persona can relate to more than one use case).

3.**List of main data items and entities – data glossary/description**- define main terms and entities in your envisioned system such as types of users (in terms of permissions), data structures and "items" or "*entities*" *at high or logical (not implementation) level* (e.g. name, meaning, usage, and NOT the data format) so it is easier to refer to them in the document. Focus on key terms (main data elements/records used in your app, types of users and their privileges etc. These terms and their names *must be used consistently* from then on in all documents, user interface, in naming SE components and database elements etc. In cases where you attach behavior and privileges to data items (e.g. user types) that also has to drive the design and QA of the SW. In later milestones you will add more implementation details for each item. This will help define planning and design for the DB and also define name space for classes, DB tables, methods, variables as well as UI elements.

4.**Initial list of functional requirements** – see class notes. This refers to high level functions you plan to develop to the best of your knowledge at this point. Focus on WHAT and not HOW. Keep the user in mind. Develop these functions to be consistent with use cases and requirements above. Number each requirement with *unique numeric value* (use 1., 2., 3. Not a, b, c or Roman numerals) and use these numbers consistently from then on. For each functional requirement use 1-3 line description. At this stage no need to prioritize the requirements, collect all ideas from the team. We are looking for 20-30 or so requirements. The best way to present this is to group them in distinct groups by increasing level of user privileges, from unregistered users, to registered, to admin. If you plan to deliver some unique functions have a requirement for each.

5.**List of non-functional requirements** (performance, expected load, security requirements, storage, availability, fault tolerance…). Note that mandatory high level non-functional specs are given in high level team project document, so for M1 – M5 we request that you **simply copy them from high level document from Canvas .** Please observe and adhere to these non-functional requirements in your design and development from now on – you are not allowed to change them unless you get permission (this is how it is done in industry). Please also do not add to this list, there are many reasons why not to do so and will be covered in the class.

6.**Competitive analysis**: Find 3-4 competitive products. Present competitors' features vs. your planned ones. First, create a table with key features of competitors vs. yours planed, only very high level, 5-6 entries max (as shown in the class). After the table, you must summarize in one paragraph what are the planned advantages or competitive relationship of your planned product to what is already available. In the table clearly mark your product, e.g. shade its column/data. If you plan some competitive (unique) feature of your design it must also be addressed in competitive analysis (hopefully your envisioned product would have it and competitors would not). This is how you get management attention.

7.**High-level system architecture and technologies used:** Briefly provide itemized list (no graphics) as below. This list is generally the same as you got approved in M0. If you want it changed mention it in the text so Anthony will approve it as part of M1 review. Please use list format, no need to draw block diagrams.

- List all main SW components and versions (DB, WWW server)
- List deployment cloud servicer you plan to use
- List front end frameworks
- List browsers you plan to support (chose 2 market leading browsers, last two versions from each)
- List any major additional external open source APIs you plan to use (e.g. Google analytics, Google map APIs, APIs/service for creating thumbnails – check Architecture class slides)

- In no more than one brief paragraph describe how will you implement search functionality (OK to do it as suggested in class architecture slides, but say it here)

8.**Team and roles**: list student names, mark their roles (team leader, front and back team lead and github master, document master (optional but recommended, can be team lead), team member front end, team member back end etc.

9. **Checklist**: for each item below team lead must answer with only one of the following: **DONE/OK**; or **ON TRACK** (meaning it will be done on time, and no issues perceived); or **ISSUE** (you have some problems, and then define what is the problem with 1-3 lines)

- So far all team members are engaged and attending team sessions when required

- Team found a time slot to meet outside of the class

- Back end, Front end leads and Github master chosen

- Team ready and able to use the chosen back and front end frameworks and those who need to learn are working on learning and practicing

- Team lead ensured that all team members read the final M1 and agree/understand it before submission

- Github organized as discussed in class (e.g. master branch, development branch, folder for milestone documents etc.)

- NEW: Use of any GenAI tool like ChatGPT: say if you used ChatGPT or like and how and for what segment of Milestone 1 (brief paragraph). As per class policy: this is allowed as a help BUT you cannot copy and paste its output and claim it is your own text, you need to put it in quotes or modify it, and only for short sentences You also are responsible for accuracy of your submission, so any ChatGPT content needs to be checked by you.

Background reading:

- Document we posted about high-level vision of our application.
- Class material on requirements and specs
- Relevant existing applications and products (Google it…)
- Info about allowed frameworks – class notes and posted on Canvas
- M0 document and documentation on SW tools and frameworks you plan to use
- Git and Github tutorials
- If questions ask Prof. Petkovic via e-mail or in class

Submission for Milestone 1 (M1) document for review – you must follow the instructions below:

Teams must collaborate in creating Milestone 1 (M1) document by having working M1 document on their team Github private repository (similar to managing code) in "Milestones" folder so all team members and instructors can access it. For actual editing you can use other methods like Google doc but final version of M1 document (and other milestones documents) must be in github folder called "Milestones"

We strongly suggest the following collaborative approach for creation and completion of M1 document (NOTE: creating a team document is similar to creating a code by the team of programmers):

- Team lead assigns M1 editor (often this can be done by the team lead). We recommend one person in a team gets charged as editor for all milestone documents
- Team lead/M1 editor assign individual chapters to team members
- Students work on their assigned chapters
- M1 editor collect chapters/assignments from each student, edits/corrects then integrates them into a well formatted document (with same font and formats)
- M1 editor posts final candidate full document on team repo so that all team members read full document again for one more review and any feedback before submission – do not forget this step – it is implant for checking as well as to make sure all team members understand the "big picture"
- M1 editor completes the final version as per feedback
- Team lead submits M1 info for review as per submission instructions below. **Submission instructions (below) must be followed precisely and completely or grade penalty will be imposed**

The whole student team submits one Milestone 1 document in PDF format, as follows: Team leads will send e-mail as below to Petkovic@sfsu.edu :

- **Submission e-mail subject line:** MUST be "CSC648-848 Spring 2023 Milestone1 Team N" in the subject line (N is a team number 01, 02…).

- **e-mail body is** to contain brief courtesy text and direct link to actual Milestone 1 document PDF file (not the folder) stored in github Milestones folder

- **Milestone 1 file name:** MUST be CSC648-848 Spring 2023 Milestone1 Team *N.PDF* (*N* is your team number) (We use only PDF so I can send you feedback as yellow sticky notes).

Submission must be done by the deadline specified and by following submission rules above. Any deadline extension has to be asked for via e-mail to Petkovic@sfsu.edu at least 24 h ahead of the deadline.

M1 document format and structure MUST be as below

- **Title page** MUST include (nicely formatted)
  –"SW Engineering CSC648-848 Spring 2023"
  Project/application title and name (you can use the name you chose for your application)

  - Team number
  - Names of students (team lead first) with e-mail of team lead. Please mark those who are team lead, front end, back end leads and github master
  - "Milestone 1"
  - History table (it should in general contain two entries: date submitted and date revised after instructor comments)

- **The rest of the document** has to contain ALL sections as described above as enumerated separate chapters with titles in **bold** (see "Content and structure for Milestone 1 document for review")

Team leads and document editors: make sure document is well formatted, reads well, is complete, and looks professional. This will be part of your portfolio and will influence the grade. Make sure all team members read final version and give comments before submission.

Instructor's feedback and creating final Milestone document for Final Project delivery

Fist submitted M1 document is NOT graded except in case sections are missing (team gets negative points) or submission is late with no extension granted.

Upon submission of M1 you will get feedback from instructors by any of the following: e-mail, markings on your document and in class during team meetings. This feedback must be analyzed and taken into account by your team in order to revise your M1 and this must be used subsequently for the rest of the project. Instructors will comment from the standpoint of CEO, VP of Marketing (who translates customer and marketing requirements) and CTO (Architecture etc.). You may choose not to agree with the comments - this is OK as long as you justify this and are prepared to live with that design and deliver it. In some cases, instructors may insist on some features or decisions.

Upon getting instructors' feedback on your questions and submitted document, and after analyzing it, you need to revise your first draft, freeze it (meaning no more changes on this document even if future design changes) and use it as a basis for developing Milestone 2 (M2). The frozen document M1 will be submitted as part of final project delivery in Milestone 5 in the last class. Please enter the revision date in summary history table on title page.

Do not start working on M2 before you get feedback on M1 and make sure all team members read frozen M1 document.

Evaluation and grading

We will grade only each "frozen" milestone when it is submitted with final project at the end of the class ("Milestone 5 folder"), after it has been modified for instructors' feedback. Note that instructors' feedback before Milestone 5 final delivery is NOT graded in order to encourage interaction (but your dealing with the feedback and timely response is graded as well as the document is missing some sections).

Milestones improperly submitted (e.g. not following ALL required submission riles) will be returned. Only one error in submission in M1 will be "forgiven" for the whole class time for each team, any subsequent problem submission in any milestone documents will be recorded with negative points as outlined under the grading rubric SE *Process grade: submissions.*

Milestone documents have to have all required sections or negative points will be recorded for final grading (grading rubric: *SE Process grade: document quality*).

Milestone documents have to be submitted on time and in a way as specified above. In case of justifiable reasons to delay, permission has to be obtained by e-mailing to Prof. Petkovic Petkovic@sfsu.edu 24 h prior to the deadline. **Late submissions with no extension grnated incur 10% penalty on the team grading of that milestone in SE Process rubric.**