CENG350 Software Engineering, Spring 2023

SRS part-1 Feedback

This document contains feedback on the CENG350 SRS part-1 from last year's project. Please take note of the general mistakes made last year to ensure they are avoided in your SRS documents.

"Dear All,

SRS part-1 submissions have been evaluated, grades are going to be announced soon. There is no time to provide individual feedback on the SRS part-1. Here is a general feedback. Please pay attention to the following points.

Submission:

1)   Although it is clearly stated in SRS Grading Rubric for afetbilgi.com document as "**SRS document (including diagrams) named as srs.pdf",** there were some Word and Latex files in the submissions. Only PDF files are going to be accepted and evaluated.

2)   Diagrams should be created in StarUML. Diagrams that were created with tools other than StarUML are not going to be evaluated.

3)   There were some submissions that do not include StarUML project files for diagrams. For the SRS full, all the diagrams' StarUML project files should be included in the submission.

4)   It is stated in SRS Grading Rubric for afetbilgi.com document that submission should include **UML diagrams' project files as separate files, which must be able to be opened by StarUML**. However, there were some submissions that include only one StarUML project file, in other words, all the diagrams were created in one single project. Each UML diagram should be created in a separate StarUML file.

Format:

1)   Format of the SRS document should be the same as the one that we have shared with you as **SRS Outline for afetbilgi.com.docx**

2)   There were some SRS documents that did not include a Table of Contents, List of Tables, and List of Figures. SRS document must include every part in the outline that we have shared with you.

3)   Diagram figures should be exported from the corresponding StarUML project. Any other kind of figure (e.g. screenshots) will not be acceptable.

4)   For the use case description templates, you may use sample projects' templates or textbook's template. It is stated in SRS Grading Rubric for afetbilgi.com document as **"Use-case description format: Pick and adapt a good format from available examples. A properly extended form of the textbook's format is ok.".** Any simpler format will not be accepted.

5)   Figures and tables for the SRS document should be listed and ordered with numbers. There were some documents that contain figures and tables without numbers and names. You should pay attention to put table numbers, figure numbers, table names, and figure names to tables and figures.

6)   There were some documents having small and unreadable figures. You should put figures with a proper size for readability.

7)   Please include your names, surnames, and group numbers on the cover page of SRS document.

System Context Diagram:

1)   System Context Diagram is used to show the overall view of the system (i.e. the afetbilgi.com) in its environment. It depicts external entities and their interactions with the system. A system component, by definition, cannot be an external entity. In this direction; database, and code files of afetbilgi.com cannot be external entities. If you show a context having a system component as an external entity, you should correct your diagram.

2)   System context diagrams should show the links or connections of external entities with the system. Links (i.e. data flows in the diagrams) should contain data/signal exchange, for example, indicating "account details" from the user (external entity) to the Uber system. There were some System Context Diagrams that show links but fail to mention the information exchanged on the link. If information exchange is not mentioned, we cannot understand the nature of the relationship between external entities and system.

3)   There were some System Context Diagrams that show relationships between two external entities. Is that relationship between two external entities really relevant for our system? If not, delete it.

4)   For section 1.3.1, a System Context Diagram and explanations for this diagram should be provided. There were some documents that has only diagrams but no explanations. Diagrams need some explanation to be understood unequivocally.

5)   The System Context Diagram should be as simple as possible. There were some unnecessarily complex System Context Diagrams that are hard to read and understand.

Use case Diagram:

1)   An actor is an external entity, be it a human or a non-human system. A system component should not be an actor in a use-case diagram. However; there were many use-case diagrams that show web API and database as actors. If your diagram has a system component as an actor, you should correct your diagram.

2)   As it is stated in the SRS Grading Rubric for afetbilgi.com document as **"You should aim to write 10 use-cases**, you are expected to aim for 10 use-cases. Slightly less than 10 would be acceptable, but much less than 10 would not.

3)   For section 3.2, as it is stated in the SRS Grading Rubric for afetbilgi.com document as **"a use-case model consists of a use-case diagram and the descriptions of the use-cases)."**, you are expected to create description tables for all the use cases. However, there were some diagrams that do not have descriptions or use cases without descriptions. If your diagram has this deficiency, you should provide the missing pieces.

4)   There were a few documents that has description tables but no use-case diagrams. You should keep an eye on consistency.

6)   If your diagram contains some "include" related use cases or "extend" related use cases, you should also provide description tables for all these use cases.

7)   Use case diagram should be as simple as possible not complex. There were unnecessarily complex use case diagrams that were hard to read and understand. You should also avoid unwarranted use case generalization/include/extend.

I recommend you review the sample SRS document and IEEE 29148 (highlighted and commented) or read them.

Best wishes,
TA İbrahim Tarakcı"