CSE307 Software Engineering Midterm and Final Project 2020-2021 Fall

Project Description V3 : 2021.01.01

Project Delivery : End of Finals

Please read this document at least 2 times

A.) Project Details

1. First of all you will determine a project
2. As soon as you determine your project, you can start working on it
3. The project can be a web application, mobile application, desktop application
4. The project has to be a project that would likely to be used on business life
5. The project scale is totally up to you but it has to be complex enough to apply software engineering logic
6. You can do the project either solo or as a 2 developer team
7. If you decide to do the project with one of your friend as a 2 developers team, both of you please email me and let me know that you want to work together
8. The project has to be developed by using GitHub or BitBucket code management systems
9. You have to properly commit, push, merge, pull request, and such
10. You can find necessary related information about code management on the Internet
11. You can use GitBash or any program for code management
12. You have to develop software with versioning such as version 1.0.0 and the 1.0.1 and then 1.0.2, etc.

13. You have to make a layout and summary of your project and ask me before starting it. So I will approve your project
14. Send me your project layout and details by email : furkan.gozukara@toros.edu.tr
15. You can do the project in any programming language with using any database management system such as using C#, PHP, ASP.Net, Python, MSSQL, MySQL, SQLite, etc.
16. Rather than the project itself, how well you apply software engineering approaches and methodologies on your project matter more
17. You will prepare a detailed report about your project which will include below software engineering diagrams, tables, descriptions, etc.
18. In addition to the project, you will record a video which will be explained at the end of this document

B.) WHICH DIAGRAMS, TABLES, DESCRIPTIONS WILL BE PREPARED FOR THE PROJECT

(A professional drawing software is a must like a Microsoft Visio, or Microsoft office)

1. Which one will be your project model waterfall, incremental or reuse-oriented development (Lecture 2). Explain in details and why?
2. Write full story of your project features (for each feature) (example case “A ‘prescribing medication’ story” in Lecture 3, page 21)
3. Write detailed test case of your project features (for each feature) (example case “Test case description for dose checking” in Lecture 3, page 31)

4. Write full requirements definition of your project (example : Lecture 4 , User and system requirements page 7)
5. Write full nonfunctional requirements of your project (example : Lecture 4 , Examples of nonfunctional requirements in the MHC-PMS page 18)
6. Fill your project nonfunctional requirements metrics table (example : Lecture 4 , Metrics for specifying nonfunctional requirements page 21)
7. Write full requirements of each part of your project (example : Lecture 4 , Example requirements for the insulin pump software system page 39)
8. Write full structured requirements of each part of your project (example : Lecture 4 , A structured specification of a requirement for an insulin pump page 42-43)
9. Write tabular computation of your each function/model of your software (example : Lecture 4 , Tabular specification of computation for an insulin pump page 45)
10. Write detailed scenarios for your project (example : Lecture 4 , Scenario for collecting medical history in MHC-PMS page 62-63)
11. Draw use cases diagram for all use cases of your project like in Lecture 4 page 65 (a professional drawing software is a must like a Microsoft Visio, or Microsoft office)
12. Draw full details context UML diagram of your project like in Lecture 5 page 10
13. Draw fully detailed process model UML diagram of your project like in Lecture 5 page 12
14. Draw every use cases UML diagram of your project like in Lecture 5 page 15
15. Prepare tabular description of your projects' use cases like in Lecture 5 page 16
16. Draw use cases of each agents' use cases UML diagrams of your application like in Lecture 5 page 17

17. Draw Sequence diagrams of every action in your project like in Lecture 5 page 19-20
18. Draw UML classes associations of all classes like shown in Lecture 5 page 23-24
19. Draw class models like shown in Lecture 5 page 25
20. Draw generalization hierarchy of your classes and all their details like shown in Lecture 5 page 30-31
21. Draw aggregation associations of all your classes like shown in Lecture 5 page 33
22. Draw activity model of your application like shown in Lecture 5 page 36
23. Draw your application processes like shown in Lecture 5 page 37
24. Draw state diagram of your application like shown in Lecture 5 page 40
25. Prepare structured forms of your application's states like shown in Lecture 5 page 41-42
26. Draw the software architecture of your project like shown in Lecture 6 page 42, 49
27. Draw context diagram of your project like shown in Lecture 7 page 9
28. Draw high level architecture of your project like shown in Lecture 7 page 13
29. Draw all object classes of your project like shown in Lecture 7 page 19
30. Prepare a detailed usage scenario for your project like shown in Lecture 8 page 53
31. Prepare reliability terminology of your software project like shown in Lecture 11 page 25
32. Prepare safety terminology of your software project like shown in Lecture 11 page 35
33. Prepare security terminology of your software project like shown in Lecture 11 page 42

34. Develop some vulnerability avoidance techniques for your system such as explained in Lecture 11 page 45
35. Prepare a risk classification tabular view of your system such as explained in Lecture 12 page 15
36. Prepare an example of a software fault tree for your system such as explained in Lecture 12 page 18
37. Prepare an examples of safety requirements for your system such as explained in Lecture 12 page 23
38. Prepare an examples of functional reliability requirements for your system such as explained in Lecture 12 page 39
39. Prepare a threat and control analysis in a preliminary risk assessment for your system such as explained in Lecture 12 page 46

C.) General Things That Deeds to be Considered

Every project will be controlled one by one. So if you cheat or bring a ready code, project, you will get FF

There will not a final exam. Your grade will be based on this Project. Thus, take this Project very seriously and start working on it immediately. Look the internet for the parts that you do not know

Write your code with as many as possible explanations. What does do that code block? Of course, put those explanations into your code with comment blocks such as // or /* */

Create an account on https://stackoverflow.com/ and ask there, or look for answers there.

My email: furkan.gozukara@toros.edu.tr

Also use discord channel for communication

Latest Project delivery date: End of finals but you can also deliver at the butunleme exams

1. **Lastly everyone will record a very detailed video to explain the project.**
2. **Video will be recorded with OBS studio in 1080p (high quality so that I can read explained code easily) resolution.**
3. **It is mandatory to use microphone when recording and explaining the project verbally by speaking.**
4. **Turkish students can record the video in Turkish.**
5. **The project has to be made in English.**
6. **Non-Turkish students can also record their video in Turkish if their Turkish is better than their English. However, they should use English if their English is better than their Turkish.**
7. **When explaining your Project you need to explain every piece of your written code.**
8. **What does that particular code piece or that particular class or that particular method, etc. and how did you write it.**
9. **Moreover, you have to explain and show every feature of your application. So run your application and do example of every feature that your application have. Let's say you have made an E-Commerce website application. Then, show how to search, add to chart, make purchase (this can be demo), list categories, etc.**
10. **Then after you have to upload your video to the YouTube (uploading to google drive is actually a more preferred method**

since YouTube tends to reduce quality) and set visibility as public or unlisted.

11. Because it has to be accessible.
12. Furthermore add your YouTube video link to the project folder as text.
13. Finally ZIP or RAR your entire Project and upload to your Google drive. After you get your Google drive upload link send me that link as email : furkan.gozukara@toros.edu.tr
14. When sharing your Google drive upload link as email, make sure that your upload's visibility is set correctly. Set it as who has link can view or download so that I can download it.
15. Everyone will be sending a Word or PDF document along with their final project files. This document will contain screenshots of your all comments to the lecture videos (all of the 14 lecture videos). Take screenshots with high resolution. Your comment has to be clearly readable. Those who fails to send this document will be counted as not attended to the lectures. Therefore they will get FF.