a. Determine the type of software system developed in the project. **Explain your answer.**

For this project, the software being developed is organic. I came to this conclusion by looking at the requirements for a project to be embedded where I do not see the level of complexity or strict constraints of an embedded system. I also do not see the increased complexity for a semi-detached project given the current constrains and use cases for our project. Team experience (myself) might not be as nominal as other projects but will be close enough that I do not believe it will result in the significant increase in KDSI as seen in the increase of a project going from organic to semi-detached.

b. Study and assign a number to each EAF from the EAF rating table. **Explain your answer.**

|  |  |  |
| --- | --- | --- |
| Cost Drivers | Value | Explanation for value |
| RELY | 1.4 | This software interfaces with student grades, errors can impact several aspects of a student including their financial aid. While the tool may not need to be used constantly, it will need to work properly every time. |
| DATA | 1.08 | The number of data stores is relatively low, and the attributes stored within are also quite basic, but there will be many entries very rapidly. Scaling would be a strong consideration, and effectiveness of querying at a large capacity would also be important. |
| CPLX | 0.85 | The complexity is fairly low, with a small number of use cases and limit scope of use. |
| TIME | 1 | With these being basic CRUD operations and files that are being parsed will contain known amounts of data, there should not be a large emphasis on performance constraints. |
| STOR | 1 | Like the justification for time, there should not be much a serious constraint to the size of the software. It’s expected to run on commodity hardware, and only serves as a data interface. The main consideration falls into the data considerations given the constantly increasing amount of student grade data that will occur. |
| VIRT | 1 | The usage of virtual machine is not a consideration for this software. Based off project requirements, the direct user interaction would require the software to be on their machine. |
| TURN | 1.07 | Response time would be a moderate consideration given the volume of operations that may be needed for many records to be imported. |
| ACAP | 1.19 | The analyst capability is lower than nominal due to lack of experience in analyzing software products from the ground up. I have some experience, but it is not a task I do on a regular basis. |
| AEXP | 1 | Application experience is about average for this type of problem. By no means am I an application expert but I do have relevant experience. |
| PCAP | 0.86 | For this type of product, where it is mainly data interactions with a GUI, I have experience through both work and school with multiple product solutions of this type. |
| VEXP | 1 | Basic experience in virtualization work, with mainly deploying VMs on AWS and running local software instances with Docker. |
| LEXP | 0.95 | The application is being developed in C#, with a SQL database. Between relevant course work and time working as a government .NET contractor, I feel that I should be above average on this rating. |
| MODP | 1 | The software utilizes a modern, object oriented programming language. Strict requirements elicitation and heavy documentation makes it about average for this type of product. |
| TOOL | 0.91 | Tools are utilized in Visual Studio for coding and in GUI creation, Office for documentation, Project for PERT/GANTT charts, and Visio for diagraming. The usage of quality tools should be a higher than average coefficient. |
| SCED | 1.04 | The schedule for the project is strict, and slightly compressed due to semester limitations. |

c. Use the formulas to calculate the efforts and time needed for the project. **Express your own opinions about these two numbers.**

Unadjusted function points:

Data Functions:

Number of user inputs:

Name (3)

ID (3)

Semester (3)

Year (3)

Course Prefix (3)

Course Number (3)

Grade (3)

Total = 21

Number of user outputs:

Grade Report File (7)

Grade Record Search Results (5)

Total = 12

Number of user inquiries

Button Selection - Main Menu Selection (3)

Button Selection - Missing Data Dialogue Box Close (3)

Button Selection – Existing Record Found Close (3)

Button Selection – Action Success Message Close (3)

Button Selection – Search Result Option Edit/Delete (3)

Button Selection – Database Connection Error Close (3)

Total = 18

Transactional Functions.

Number of Database Tables:

Student Info Table (7)

Student Grades Table (10)

Course Credit Hours Table (7)

Total = 24

Number of External Interfaces

Student Grade Report (10)

Total = 10

UFP = 21+12+18+24+10 = 85

Adjustment Factors:

1. Reliable backup and recovery (5)
2. Data communication (3)
3. Distributed functions (0)
4. Performance (3)
5. Operational environment (3)
6. Online data entry (4)
7. Multiple screens for input (3)
8. Online Update(0)
9. Interface Complexity (2)
10. Reusability (2)
11. Process complexity (4)
12. Installation Ease (3)
13. Multiple Sites (0)
14. Ease of Use (5)

Total = 37

FP = 85\*(0.65+0.01\*37) = 86.7

LOC = 54 (C#) \* 86.7 = 4682

EAF = 1.265.

Organic Effort Equation:

E = 1.265 \* 3.2\*(4.682)^1.05 = 20.47 PM

Schedule Equation:

TDEV = 2.5\*20.4^0.38 = 7.87 Months

These two numbers seem quite high in my opinion for the type of software solution we are developing for the term project. While software developed for a class would not necessarily be at production quality, even with robust testing and revisions I don’t picture a basic grade import software to take that long to develop. The effort and scheduling values also point towards 2-3 people working on the project and while that sounds reasonable, it just does not make sense for the timeline. 2-3 engineers should be able to develop that type of system in a much faster period of time. I imagine these calculations for the COCOMO model are a bit more accurate on larger scale projects and definitely are great for estimating timelines and cost, but the specific project that we are using the calculations for makes it feel like a massive overestimate.