**Class Diagram (Implementation Perspective) – 10 points**

**Submission**

1. Due date: Friday, **Nov. 8 by 11:59pm**
2. Submit (a) the Word document with the required information, (b) team report, to the repository or Canvas. **-2 points** for not submitting the team report.

**Requirement**

1. Create an implementation level class diagram for your software project. You MUST use a drawing tool, such as Visual Paradigm, to create the diagram and export the diagram as a picture file to be included in the Word document. **You will get a 0** **for this deliverable,** **if you don’t use a drawing tool**.
2. Use the identification technique we discussed in class to help you identify the classes you may need. You should include major attributes and operations for each class, as well as the associations between classes.
3. **Create a Word document and include the following sections**:
4. Title, including the project name, team name and names of all team members.
5. System requirements, revise and copy the requirements from the SRS to this section. Underline all nouns.
6. Use a table and list all unique nouns from (b). For each, consider whether it meets all of the following:
   * Has state (i.e., information to store)
   * Has responsibilities
   * Has behavior
   * Not redundant
   * Within project scope
7. Create a Class Diagram with the classes identified in (c) and include the utility classes or control classes if necessary. Use a drawing tool to create the diagram and export it as a picture and include it in this section.
8. If a team member’s contribution to this deliverable is significantly less than other team members, he/she will only receive a partial credit of the total points earned by the team.

|  |  |  |  |
| --- | --- | --- | --- |
| **Expectation** | **Meet** | **Does Not Meet** | **Missing** |
| The Word document is organized according to the requirements | 2 | 0 | 0 |
| Reasonable classes have been identified in 3(c) | 3 | 1 | 0 |
| Class Diagram   * Correctness of the notation used * Consistent with 3(c) * Include necessary utility or control classes | 5 | 2 | 0 |
| **Total** | 10 |  | |

1. Grading