

# CSC 490 – Spring 2022

## Software Design (Delivery No. 2)

**Due Date: 10 / 4 / 2022 (soft copy uploaded to Blackboard)**

In the second phase of the “CSC490 Software Engineering” project, you are required to produce the design of the “**Virtual Learning Assistant**” software that you created its specifications in Phase # 1. The main operations that are required from you in this phase are described below:

1. Identify the systems that **interact** with the VLA and draw the **Context Diagram**: Here, you go over the Requirements Document from Phase # 1, and deduce from the **requirements** the **software/tools** that your VLA interacts with, then you draw the Context Diagram, as we took at the beginning of Chapter 5.
2. Design the VLA architecture: If the VLA that you are creating contains two or more **subsystems**, you should draw the VLA **Architecture Diagram**. Identify all the **connections** between the VLA subsystems and include them in the diagram. If there are multiple types of connections between the subsystems you draw them using different notations (for example, different arrow shapes), and identify the type of each connection using UML stereotypes. For each subsystem in the VLA that contains multiple **components** (i.e., packages), you include the components within the subsystem, as shown in **Figure 7.5** in the textbook (page 202), and you draw the connections between the components.
3. Design and draw the VLA Use Case Diagram(s): Here, you go over all the requirements in the Requirements Document, and identify all the interactions between the VLA and external actors. For each interaction, you create a use case in the Use Case Diagram and connect it with the corresponding actor(s) using the “communicate” relationship. Next, you identify the functionalities that are included in multiple use cases, split each such functionality into a separate use case, and then use the “include” relationship to connect it to the parent use cases. In addition, you identify the use cases that contain special cases (alternative paths), and you represent each alternative path as a separate use case and connect it to the original use case using the “extend” relationship. Finally, you identify if there is any generalization between actors/use cases and express it using the “generalize” use case. If your VLA contains a large number of use cases in multiple subsystems, you can draw a separate Use Case Diagram for each subsystem, then combine the use case diagrams of the subsystems to obtain the Use Case Diagram of the VLA. This will be easier to implement than drawing all the use cases directly in the same diagram.
4. Expand the design of selected Use Cases: In this part, you are required to produce the complete design of **three** Use Cases (**for groups that contain four students, you are required to produce the complete design of four Use Cases**). Each student should work on expanding the design of one of the selected Use Cases. In the Delivery report, you should state the name of the student who worked on the design of each Use Case that you selected.
  - a. First, you select the use cases that you want to implement from the various use cases in the Use Case Diagram. Make sure that you are able to implement (i.e., write

the code of) each Use Case that you select. For each selected Use Case, you write the Use Case description. This can be written either as a tabular format, similar to Figure 5.4 in the textbook, or using a use case scenario, similar to the examples in the folder "Use Case Scenarios" that was uploaded by the instructor to Blackboard. The use case scenario is better, as it can be used to describe the various steps of the use case in detail. Pay attention to describing all the details of each use case, since you will depend on these details when producing the detailed design diagrams.

- b. Translate the steps of the use case into an Activity diagram. Make sure you include all activities that are executed to accomplish the use case. Include all occurrences of branching (decision points), insert synchronization bars for activities that are executed in parallel, and include swimlanes whenever needed (a separate Activity Diagram is drawn for each use case).
  - c. Identify the **classes** that are required to execute the use case, and identify how the use case will be realized by executing various methods in the classes that you identified. Draw the classes and method calls in a Sequence Diagram. Include sequence diagram fragments (such as loop, ref, alt ...) whenever needed (a separate Sequence Diagram is drawn for each use case).
5. Design and draw the VLA Class Diagram: **For each class** that you used in the sequence diagrams in Step number 4, you deduce from the diagrams that you draw so far the **variables** and **methods** (functions) that are required in the class and add them to the UML representation of the class in the Class Diagram. Next, you identify the various **relationships** between the classes and add them to the Class Diagram (Associations, Whole-Part, and Generalization). You should include the **multiplicity** values, which are the modality (i.e., minimum participation) and cardinality (i.e., maximum participation) of each class in the relationship whenever needed.
6. Design and draw the VLA State Diagram: Here, you examine the Class Diagram in step number 5, and deduce the various **states** that the VLA could be in (for example, "setting the study hours", "calculating the average study hours", "notifying the student", "searching the course content", etc.). Next, you draw the State Diagram of the VLA, similar to Figure 5.16 in the Textbook. The diagram should include, in addition to the states, the **events** that cause the VLA to move from one state to another. In addition, the diagram should include the description of the **action** that the system performs when it enters each state (which is written as "**do: [action]**" under the name of the state, as shown in Figure 5.16). Optionally, the diagram could include the description of the action that the system performs when it exits the state (which is written as "**exit: [action]**" under the "**do: [action]**"). The exit action is optional and is added when needed.
7. Based on the State Diagram that you produced in Step # 6, review the Class Diagram that you created in Step # 5: make sure that you added the variables and methods that are needed by each class in each state of the system; identify cases in which **abstract classes** or **interfaces** are required, and modify the Class Diagram accordingly.

After you finish the system design, you need to write the Delivery document of this phase. The Delivery document should contain the following **main sections**:

1. **Introduction:** A summary of the system design that highlights various steps that you performed, the design models that you applied, the links between the various parts of the produced design, and the importance of each part in the intended system implementation.
2. **VLA Context and Architecture:** In this section you present the Context and Architecture diagrams and explain the details of each diagram: you describe in detail the elements and components of each of the two diagrams and link them with the Requirements document that you produced in Phase # 1 of the project (for example, you identify the requirements from which each external system (for Context Diagram) or Subsystem/Component (for Architecture Diagram) were deduced).
3. **VLA Interactions:** In this section you present and discuss the details of the Use Case diagram.
4. **VLA Components:** In this section you present the detailed design of the Use Cases that you selected in Step # 4 in the previous paragraph. For each Use Case that you designed, you present the Use Case description (or Use Case Scenario), and the corresponding Activity and Sequence diagrams of the Use Case. You need to describe the details of each diagram so that the reader can understand what you are expressing in the diagram (this description will play the role of the design documentation for the stakeholder).
5. **VLA Structure:** In this section you present and describe the VLA Class Diagram. You include the details of how the classes will be categorized into packages, the functionalities in which each class will be used in the system implementation, and the description of the various relationships between the classes. In addition, you present and discuss the VLA State Diagrams. You describe the State Diagram using a Table similar to that in Figure 5.18 in the Textbook. In the table, you briefly describe each state and event in the diagram.
6. **Conclusion:** In this section you highlight the importance of the design that you produced in this phase and describe how the design will be used in the next phase to implement the VLA system.