CS 3331: Advanced Object-Oriented Programming

HOMEWORK 2: UML

Due date 06/23/2018

This assignment may be done individually or in your designated team. If you work in
team, you need to fill out the contribution form (see the course blackboard).

In this assignment, you are to document the requirements of Connect Five applications using UML and design a reusable model of the applications (see the HW1 handout for Connect Five).

1. Draw a use case diagram for the Connect Five application that you are going to develop throughout this semester. As shown in the first class, the application shall allow a Connect Five to be played by a single player or a pair of players connected through a network. For each use case appearing in your diagram, provide a detailed description in the format used in the class. You should document the main success scenario along with alternatives and exceptions (or errors). Your scenario description should be detailed enough, complete and testable so as to be used as the basis of your design (see below).AND AI

2. Design the Connect Five application to support a 1 player. Your design goal should be coming up with reusable model classes that can be tested once, independent of UI, and reused throughout this semester. You should use the Model-View-Control (MVC) metaphor [2], and your model classes should be completely separated from the view or control classes; there should be no dependency from model classes to view/control classes. Document the static structure of your application by drawing a class diagram.

   - Your class diagram should show the main components (classes) of your application, their roles and their relationships.

   - Your model (business logic) classes should be clearly separated from the view/control (UI) classes with no dependency.

   - For each class in your diagram, define key (public) operations to show its roles or responsibilities in your application.

   - For each association (aggregate and composite), include at least a label, multiplicities and navigation directions.

   - You should also need to provide a short, textual description of each class appearing in your class diagram.

3. Identify important state changes of your application as the game progresses from the start to the end, and draw a state machine diagram to model them. Your states and transitions should be written in terms of classes and operations appearing in your class diagram above. You will need to include a short description of each state and transition that appears in your diagram.

4. Use a UML tool to draw your diagrams; see
   your syllabus for free UML tools.

WHAT AND HOW TO TURN IN

Submit a pdf document of your UML diagrams along with accompanying documents using blackboard 06/18/2018

GRADING

You will be graded on the quality of your design. The presentation of your diagrams and the correctness of notations also matter.

REFERENCES

[1] Martina Seidl, et al., UML@Classroom: An Introduction to Object-Oriented Modeling, Springer, 2015. Free ebook through UTEP library.

[2] Holger Gast, How to Use Objects, Addison-Wesley, 2016. Sections 9.1 and 9.2. Ebook available from UTEP library.