

# SYST 17796

## Deliverable 2

### DESCRIPTION

In Deliverable 1 you took the base code and came up with ideas for extending the code into a game of your choice. In Deliverable 2, you will document your new design using fully developed use cases and class diagrams. You will also describe the OO Design principles that will be used in your final code. Please take care to review your feedback from Deliverable 1 (particularly the design document) and incorporate any suggested improvements into your Deliverable 2. The same standards for groupwork, professional writing style and citations apply to all deliverables. If you have questions, you can refer to the project description or ask your instructor. **The project requirements are not to be reduced for groups smaller than 4 students. Any groupwork conflicts will be dealt with using the contract from Deliverable 1.**

To begin, you will take the rules of your game that you provided in your Deliverable 1 Design Document and turn them into fully developed use cases for your game. You should have received feedback on your project scope in Deliverable 1 but to be clear, your fully developed use cases will depict the scope you will develop to in Deliverable 3 (when you complete the code and tests) so please include only those requirements/use cases that you plan to turn into code. Think like a tester and include alternate paths in your use case descriptions.

Next, take the base code class diagrams you produced in Deliverable 1 and extend them (and change as necessary) to include your new requirements. Use proper notation and include multiplicities and associations. You may optionally include methods and return types however this is not part of the rubric (but will make your coding easier).

Finally, complete the Deliverable 2 Design Document Template to comment on how your new design addresses the flexibility of the system, and uses the OOD principles we have discussed in class to date.

### SUBMISSION

Please submit **one PDF document per group**. This means that the document should be professionally organized and have a uniform style throughout. It should look as though it came from one team, not 4 separate students. Please note that instructors may choose to run your submission through TurnItIn or

compare the submission with other students from other sections for academic integrity purposes. Please take the time to properly cite your sources.

#### RUBRIC

Item	Criteria	Weight
<b>Use Case Diagrams and Narratives</b>	Fully developed use case diagrams and narratives produced, including alternate paths. The scope of the code is well defined given the use cases. The use cases are functionally correct based upon the rules of the game cited in Deliverable 1.	4
<b>Class Diagram</b>	Diagram is notationally correct and complete, showing all necessary classes, associations and multiplicities. Also shows an understanding of aggregation, inheritance, and composition where appropriate. The diagram also depicts an architecture that follows the principles of OOD such as cohesion, coupling, delegation, and encapsulation.	4
<b>Design Document Template</b>	All sections of the template are complete. The design choices described represent the principles of OO Design studied in class and are well articulated and presented.	2

#### FINAL DOCUMENT LAYOUT

One PDF with the following sections denoted using page numbers, headers and a table of contents:

- Use Cases (fully developed)
- UML Class Diagram
- Design Document Template