David Anderson
Module 6 Assignment

- **What is the purpose of Use-Case Analysis?**
    - During use-case analysis, we are further developing our architecture by defining classes and class associations using our use case flow of events. During this activity, we are also developing Use-Case realizations to define the behavior of our use cases and associate them with particular classes. Our output artifact is a model of the system that is more detailed than our use case model.

- **What is a Use-Case Realization?**
    - A use-case realization is a collection of design diagrams that can be traced back to a particular use case. This includes class diagrams, communication diagrams, and sequence diagrams that "realize" a particular use case from the use-case model.
- **What is an analysis class?**
    - An analysis class is an early prototype of one of the classes that will make up the system. They attempt to identify what responsibilities and behaviors should be assigned to which classes. These are created during use-case analysis. Most of the time, these go through a lot of modification during the design phase.
- **Name and describe the three analysis stereotypes.**
    - **Boundary classes** represent the boundary between the system and an actor. This is often in cases of user input such as forms in web applications. There is a boundary class for every use-case actor connection.
    - **Entity classes** represent the information a system stores and the structure of that information. These are derived from the use-case flow of events.
    - **Control classes** manage coordination of behavior and interactions between multiple objects. These are also derived from the use-case flow of events.
- **Describe some considerations when allocating responsibilities to analysis classes.**
    - If a class has data needed to perform a functionality, make that class responsible for the operations on that data (encapsulation). If multiple classes have it, they need to have a class relationship.

    - Reuse should always be a priority. If a class already has a similar responsibility, group it with that class.
    - There should be a responsibility for every message in an interaction diagram.
    - Responsibilities should be consistent.
- **What two questions does multiplicity answer?**
    - Is the association mandatory or optional?
    - What is the minimum and maximum number of instances that can be linked to one instance?