# CS 255 Model Application Short Paper

Brian Bentley

Brian.Bentley1@snhu.edu

Southern New Hampshire University

## Process Model Application

A process model will show how the system processes will proceed. This model will show the tasks that need to be performed. This will also show the input and output of each task. There will also need to be conditions that are stated for each task, both pre and post conditions. We will also need the flow and sequence of each task for the process model. There are different models that are called process models, such as the Waterfall model and the Agile model for example. Another example of process models is use case diagrams or activity diagrams.

For the DriverPass program we could create a use case diagram that will show the different uses of the system. Some of the requirements of the DriverPass system are that students will need to be able to access a driver course from a selection of courses with different features. Such a process would look like the following. A student will need to be able to login with their unique ID, navigate to the course selection screen, choose a course to add to their cart, and purchase that course using their unique ID as well as their financial information.

From this process we can also see the student will need to be able to check their course information and access the class resources if they selected a more premium tier set of classes. A process flow or use case diagram for this process would look like the following. Once the pre-condition of buying the class has been achieved, the course object that contains the resources would need to be opened for access. This would mean that the course object opens, and the user would have to be able to navigate the LMS website to download the resources. The resources for the course would be stored in an html link hosted to a repository at the AWS cloud services.

This type of model would also be similar to a activity diagram and would fall into this quite cleanly. Taking those processes above and breaking them into a set of activities based on the user would allow you to create the flow that should happen in your system. This could later be given to the testers to quickly get to work testing the DriverPass systems efficiency.

## Object Model Application

The object modelling system is a set of modelling requirements that programmers and computer scientists are familiar with. This type of modelling is more in line with UML diagrams and data flow diagrams.

The DriverPass system would benefit from a object model and would probably be best suited to it. If this were my project, I would create a UML diagram since this system is not overly complex. The different objects would be users, the interface, transactions, and other various aspects of the program.

For something like the user, you would want to create an object user which contains userID, userPass, userView. These different objects inside the object would all be private since we do not want them to be modified outside of the object themselves. These objects would have relations leading from the that would form a type of data flow between the objects.

userId, userPass, and userView would also have data types attached to them. For instance, the userID would be a integer data type since this is the most common form of ID number. The userPass would be a string and the userView would be a View object. This type of data modeling allows a programmer to visualize the objects that they will need to create in their program and begin coding as soon as possible since it essentially outlines the objects and methods in the system.

## Process and Object Model Comparison

The advantages of the Process model are that it allows one to see the overall flow of how a system is “supposed” to work. This is a high-level model which gives developers and product owners an overall perspective as to the objectives they will need to set to complete the system. The main drawback with the process model structure is that it is not set up in a way that allows you to quickly program objects in the way that Object models are.

The process models are ways to describe how parts of a business work or the entire business works. These types of models are datamined when you can really describe how a business works and what it does exactly.

Object process models are more software-oriented models that help software engineers get to writing software fairly quickly. These models are “lower-level” in the sense that they are showing minutiae of the overall “business process model”. Object models are weaker in showing the overall business process flow since they are not intended too. This is just a subsection of that overall larger process.

## References

Dennis, A., Barbara, H. W., Tegarden, D. (2012). *Systems Analysis and Design with UML*. Hoboken, New Jersy: Wiley.