# CS 255 Model Application Short Paper

Nicholle Caudy

Nicholle.caudy@snhu.edu

Southern New Hampshire University

## Process Model Application

The process model or Data Flow Diagram for the DriverPass scenario would be quite extensive as there is a lot of information that needs to be moved from one place to another. DriverPass offers three different packages to their customers. They include various hours in a car with a trainer with the option of having an in-person lesson for the explanation of the DMV rules and policies. There is also the option of access to an online class with all the content and material with practice tests for the customer. There will be several users involved including the employees/trainers, IT, administrator, secretary, and the customer. They all have different roles, and a few can do the same thing within a process. For instance, IT and the administrator both can reset passwords and accounts as needed for the customer users and trainer users. The customer user information such as name, address, phone number, email address, picture, package selection, any special needs, and payment information will need to be stored. There will also need to be a vehicle process because DriverPass has several vehicles that are available for their on-the-road driving instruction and tests. There will also need to be a DMV source as they will need to add any updates to keep the information to the customer user current for the new rules and policies of driving.

The process model for the DriverPass application should show the processes and the data flow between those processes. Starting with the customer side as a source, the customer information data would flow to the customer information process where their information will be stored throughout their time using the application. This information would be sent to the scheduling process. The scheduling process will have data flow to and from the secretary source, administrator source, and trainer source. The secretary source data flow will connect to the scheduling process as they can change appointments within the system as needed for the customer. The administrator source’s data flows to the customer information process as they can reset passwords and delete accounts as needed. The IT source will also connect to the customer information process as they can reset passwords and accounts. The trainer source will connect to the scheduling process as they will need to see when they are scheduled to take a customer for their driving instruction or test. The online class process will have data flow from the DMV source as they make updates. There will also be a payment information process to process the payments from the customer users.

## Object Model Application

The object model is a more thorough look into the system that needs to be developed. They include object classes, attributes, and functions. Object models have encapsulation and inheritance properties. Encapsulation hides some information from the outside world and inheritance allows one class to inherit attributes from another class. These object classes are connected based on the information that needs to be shared such as aggregation or a stronger form of aggregation called composition, association, and inheritance.

The object model for the DriverPass scenario would start with the use of a UML diagram. There will be a customer user class, with attributes that include name, address, phone number, email address, picture, package selection, any special needs, username, password, and payment information. The functions for this class will include update options for all this information. There will also be a trainer class with attributes that include their personal/contact information, and the functions will also have the option to update as needed. The administrator class connected through aggregation, will inherit permissions from the other classes and have administrative attributes that include the ability to reset passwords and accounts, create reports, and delete accounts as needed. The IT class connected through aggregation will inherit permissions from the other classes as they will have the ability to reset accounts and passwords as well as make changes to and update the system as needed. The secretary class would have the function to schedule appointments and update contact information for the customer user. The DMV class will have the function to update the DriverPass application with new rules and policies. An appointment class will include the trainer, customer, and appointment slot. There will need to be an online class object class that will include the attributes of classes and tests available.

## Process and Object Model Comparison

The advantage of using a process model for the DriverPass scenario is it gives the client a visual representation of how the system is going to work and how the data flows from one process to another giving them a better understanding of how the system will work. A process model gives the client a deeper understanding of the system they are requesting to be designed without needing to understand the technical language that the object model will include. The process model serves as a roadmap of the system from start to finish.

The advantage of using an object model for the DriverPass scenario is it visually represents the deeper technical aspects of the system. The object model shows each class within the system and what methods for those classes will be used. It demonstrates what attributes are private and public. The development team will use the object model to have an idea of where to start building the application, what classes will be inherited from another, and what information needs to be encapsulated.

There are some disadvantages to a process model. They don’t show how the system is going to be designed. The process model gives a very broad view of how the system will be developed and lacks technical terms. A process model does not really demonstrate how extensive the process might be for developing the system.

A disadvantage to using an object model is that it uses more technical terms, while beneficial for the development team may be confusing to the client if they lack the knowledge of the technical side of the development. They also have a steep learning curve when creating an object model because you must understand how a system needs to be developed for it to function properly.

## References

Valacich, J. S., & George, J. F. (2019). Modern Systems Analysis and Design (9th ed.). Pearson Education (US). https://mbsdirect.vitalsource.com/books/9780135172827

For a website: