# CS 255 Model Application Short Paper

Matthew Trembley

matthew.trembley@snhu.edu

Southern New Hampshire University

## Process Model Application

The process model for DriverPass would be relatively simple in relation to the overall system. DriverPass needs a system that will help them track appointments, in which are in progress, yet to be started, or finished. The process would start with a customer signing up by creating their own unique username and password. This will allow them to purchase a package, in which after they do that – they can finally book appointments. Depending on the package purchased, they will have an allotted number of driving appointments they can book, 6, 8, or 12 hours. This change in schedule will be directly and immediately shown to DriverPass and the corresponding customer (and to a certain extent, other customers so there are NO double-bookings). On the other hand, if any changes need to be made, such as an emergency cancellation – an employee of DriverPass may have the ability to make a change. In cases of an employee leaving, the IT administrator will have the ability to block access to the ex-employee so they can no longer make any unwanted changes. The process model will create the steps for each function to show easily visualization of data flow and communication. This is important for DriverPass, as they wish for data to be changed in an almost instantaneous timeframe, allowing them to see changes in real time.

## Object Model Application

The object model for DriverPass has a few different objects that can easily interact with one another. For starters, we have a Reservation object, with attributes such as time, data, instructor, customer. There may also be functions for this object that only certain users can access, like creating a reservation, canceling a reservation, or modifying a reservation. There are also other objects like Customer, Administrator, and Employee. These objects will have their own private data like usernames, passwords, IDs’ and on a public note, appointment dates, times, and instructor/customer. These will all be relative to a Schedule object, in which appointment dates, times, and instructors and customers will be public data. On the same note about the Schedule Object, only the Administrator and Employee users can make changes to the schedule like cancelling or modifying reservations after a phone call from the customer. The object model for this system will be comprehensive to include all the available functions and attributes for each object to visually create the entire system. Although, unlike the process model, it will not create the step-by-step processes on how reservations can be changed, users be created, or employees changing and so on and so forth. It is just a cement diagram showing what makes each class and object, and which attributes are needed to help it.

## Process and Object Model Comparison

There are many advantages for each model, process and object, for the DriverPass scenario. On the positive side for process models it will show easily how the system will interact between the customer and DriverPass. It will allow for easy visualization of the process on how reservations can be created, modified, or canceled. The process model can show steps for any function and what it needs to be complete, from creating a reservation to updating the system with new purchasable packaahes. Also, it can point out any flaws of the process or anywhere where the process may be able to be improved. The negatives for the process model are it can not necessarily show WHO can edit the schedule and reservations. It may show what steps it will go through, but the matter of who is up to an object model to show how it will all come together.

The object model for DriverPass will be the stronger visualization tool for the system. The object model can show each object, like reservation, schedule, and user with sub-classes of user like Administrator, Employee, Customer. Each of these objects will diagrammatically show the functions they can each utilize and what data each object can edit. The idea for this is to show the amount of abilities each user type has. As for the data, there are items that may be somewhat necessary like name, date of birth, phone number, address and other identifying data that will allow for the Administrator to find an employee or customer within the system. The downside is unlike the process model and it may seem kind of obvious is that it will not show the exact steps each function will need to go through to complete a change in the system. The object model is more so to create an understanding of what each object is and what attributes it needs to fit under each class and what functions it has access to.

All in all, it will be useful to utilize both models while creating the system. It is important to understand the flow of the system as a whole; from beginning to end how it will all work together. The object model will be just as important, especially if DriverPass would like an update on how the development is coming along, it is easy to show and explain to DriverPass the model and what each diagram will need. This will allow for DriverPass to have any input to help the project move along and clear things up, especially if used in conjunction with an Agile management approach.