# 

# CS 255 Model Application Short Paper

Ginessa Stefanick

Ginessa.Stefanick@snhu.edu

Southern New Hampshire University

## Process Model Application

When a business wants to look closely at its business processes for either documentation purposes or in preparation for a system upgrade or change, it should begin by looking at the current or desired process flow. In this project business, analysts will begin by looking at the desired flow for the new system. They will also look closely at the processes, sources or sinks, data stores, and desired data flow. Once this information has been gathered and analyzed, a data flow diagram can be created. A data flow diagram is a graphical representation of the system, and a data flow diagram would help describe the DriverPass system.

To begin documenting the DriverPass system, we will first determine the data sources. Data sources are defined as the origin or destination of the data or an external entity. For our purposes, the data sources are external actors such as the DriverPass users, secretary, IT Manager, and the owner of DriverPass. Liam. Each of these sources will interact with the system in its way. Users will need to be able to select online training, take practice exams, and schedule on-the-road training through a user account to improve their driving skills. The secretary needs to be able to conduct these same tasks for users that call into the company. Ian, the IT Manager, needs to have full administrative access to conduct system maintenance and access to user accounts with the ability to reset user passwords. Liam will need access to add, remove or modify employees, run reports, and pull information into a downloadable format.

Now that our data sources have been identified, we can begin to determine the potential processes. Looking deeper into the data sources, we see that all system tasks are performed internally. The process begins with the option to select one of three on-the-road packages. The packages should be accessible to online users and the DriverPass secretary. Package selection is process number 1.0. In our model, arrows would point from the user and secretary to "View Driving Packages" from both the user and secretary. The process flows from "View Driving Packages" to a package selection process. There would be an arrow pointing from "View Driving Packages" on our Process Model" to "Choose Package."

After the selected package has been identified, the next step is to schedule an appointment. The following process, "Schedule Appointment," is numbered 2.0 and has an arrow pointing to the IT Manager data source. The scheduling appointment process allows users to see 2-hour intervals where an in-person lesson is available. Once an appointment has been selected and confirmed by the system, a record of the appointment and the trainer's name will be sent to both the system and the user.

The next process, "Classes and Exams," allows the user to select and complete online classes or take practice exams. The data flow arrow for this process will point from the user to the "Classes and Exams," identified as process 3.0. Users that selected package three from our "Schedule Appointment" process will also have an option to view or print online materials. A "User Record" process will come into action as classes are completed, and exams are taken.

The "User Record" process 4.0 involves providing users a record of classes completed and exams attempted/passed with their scores. This process should point to the IT Manager's data source, so all glitches or errors will be sent to the IT Manager to be vetted and corrected.

The final data source was identified as Liam, the owner of DriverPass. The system needs a "User Management" process to update and modify employees and other users and their assigned system access. A user access data arrow should be pointing from Liam to the "User Management" process. There should be a final data arrow pointing from this process to the IT Manager, who will be able to complete the "User Management" process.

## Object Model Application

An object model application provides a “visual representation of software or systems’ objects, attributes, actions, and relationships,” (Nduati, 2020). An object model relies on objects and classes, rather than processes. Objects are people, places, and things where classes are defined as a representation of the objects. In this model, the major components of an object model are explained to the reader.

For the DriverPass scenario, I would choose to use a UML (Unified Modeling Language) diagram. UML diagrams graphically express the software design. The DriverPass UML would begin with a base User class, other user types would inherit access rights from the User class and would then have either more rights or additional responsibilities assigned based upon the user type assigned when the account is created. All user accounts require an email address to be used as the username and a password, which is required to log into the DriverPass system. The various user type classes will have aggregates of the classes they are composed of.

For example, a user consists of a user account object, a training object, and an in-progress object. The secretary class would not have any objects associated with it. The secretarial class only has functions such as scheduling, updating contact records, and building trainer schedules. Our IT Manager would inherit the same functions as the secretary user and would have full administrative access to user accounts. Liam, the owner, would have functions allowing him to maintain employee access, run reports, and pull data as well as the same access the IT Manager has.

A UML diagram is the best option for this approach. Providing a clear representation of each user type, mapped processes, or available functions can be laid out by the development team. In this diagram, user roles and responsibilities can also be outlined. In this format, it is easier to see what restrictions, if any are needed for each user type. This method also helps us see where there are holes in our process design.

## Process and Object Model Comparison

There are always advantages and disadvantages when comparing two different methods. The key is to determine the best of both worlds for the task at hand. The process model provides an advantage when analyzing DiverPass scenarios, and then analyzes the processes required by the system. The disadvantage of a process model is the behavior of the system is not explained well.

The object model helps explain the behavior of the system and demonstrates which users will complete which tasks. When paired with Gantt charts, and a complete UML diagram, there is a clear picture of the direction the new DriverPass system is planning to go. It is easy to make modifications to a UML diagram, and when timeline adjustments are needed a Gantt chart makes those accommodations and provides a clear visualization of the scope of the project.

In addition to the visual aspects of process models versus object models, we can examine their use during the programming phase. For some users, the visualization of the Object Model offers a clear understanding of the who, how, and what when it comes to users, and data. The process model does not provide this visual, however, for some users, seeing a process map makes coding easier.

Based on my review of the information gathered in the interview, the process model, object model, and all other aspects of the business requirements, I think I would choose to use a process model. For me, I prefer to begin with a 1,000-foot view, to understand the complete picture, before I get down to the granular level of an object view.

## References

Dennis, A., Tegarden, D. P., & Wixom, B. H. (2012). Chapter 4: Business Process and Functional Modeling. In *Systems analysis and design with UML, 4th Edition*. essay, John Wiley & Sons.

IBM Cloud Education, IBM Cloud Education. (2021, October 1). *What is business process modeling?* IBM. Retrieved June 5, 2022, from <https://www.ibm.com/cloud/blog/business-process-modeling>

Nduati, J. (2020, December 1). *The Basics of the Object Model*. Section. Retrieved June 5, 2022, from <https://www.section.io/engineering-education/basics-of-the-object-model/>

Valacich, J. S., George, J. F. (20190103). Modern Systems Analysis and Design, 9th Edition. [[VitalSource Bookshelf version]]. Retrieved from vbk://9780135172841

*What is Unified Modeling Language (UML)?* What is unified modeling language (UML)? (n.d.). Retrieved June 5, 2022, from https://www.visual-paradigm.com/guide/uml-unified-modeling-language/what-is-uml/