# CS 255 Model Application Short Paper

Kyle Mosely

kyle.mosely@snhu.edu

Southern New Hampshire University

## Process Model Application

For development construction and communicative solutions during requirement gathering, process models will be used to understand functions and processes for the Driverpass system as we architect the behaviors and design. Creating process models improves analyst, developer, specialist, and product manager roles’ understanding in the software being created. This model works to dissect actions, behaviors, functions, and classes for reduction in complexity. Process models will assist as we move to solve real world problems. In the case of Driverpass, we look to solve high failure rates for driving students.

For customers, application requires a customer to complete application and questionnaire prior to moving to next steps. The customer then is prompted to order packages that are provided in DriverPass. Once a customer checks out, purchasing one of the packages, the system will make the learning portal available, as well as the ability to reserve on the road (OTR) training appointments. At this point, staff will be involved during the training process, providing customers with a detailed progress report on OTR driving and learning activities. “Reporting Progress” will be the name of the process used for reporting.

For security, process models will have the several processes; Head of security establishes requirements for the program. One such process could be a system for scaling for security programs based on the need of the administrator. Another process would be risk analysis and vulnerability identification. The head of security needs to provide a process to secure process protocol during customer password resetting.

Lastly, a process used for administrators. Administrative users manage modules inside of DriverPass’ learning platform. Administrators decide layout approaches for captivating customers attention. Administrative users’ role is to implement changes to the learning platforms. Concepts in the platform are separated into basic and common concepts. As the application runs, the process shows what occurs when launching.

## Object Model Application

For objects, (customer, security, administrative) these will be dissected to variables needed for launching the program. Both customer and administrator object require phone number, email, password, username, and user id. The functions needed to facilitate this would be check out, place order, user profile update, login, and verify login. For administrators, additional functions include design layout, system update, etc. Administrators require functions to add/delete users as well.

Security user objects require similar methods and variables; password, username, email, etc. Security officers may need additional variables like authentication code, security credentials, user number, etc. Methods provided to security objects include updating security systems, password reset, etc. Some variables for security, there would be user group authentication, 2FA enablement on user accounts, and security settings.

## Process and Object Model Comparison

For program modelling, several models pose advantages and disadvantages for development teams. Process models assist in understanding groups and their use cases when interacting with the platform. Process models itemize aspects of the program. Rather than taking all use cases and bundling them into a single user, process models categorize use cases, providing general concepts for how the system will work, and what mechanisms are needed for the system to work as required. While the process model provides explanation for how the entire system will work, it may not explain how or why the system works the way it does.

Object models assist in further breaking down of the system, providing methods and variables for describing how system parts work and collide. Object models map processes into different variables. The function of the object model assists development teams in understanding how the system should work for the part and for the whole system. Models give an x-ray view of the system being asked by the system owner for full clarity during development.

Using both models described for the development of different programs and systems as different needs arise is a good practice; Opting for one model or the other may still be permissible for understanding parts of a given system. Teams may have higher comprehension in differences of each system part and would do more accurate development as represented when using different models. As development teams work to produce required results, the models will represent how the system parts collide with the whole of the architecture being developed.

## References

Redding, Guy., Dumas, Marlon., M. H. Arthur ter Hofsteded., lordachescu, Adrian. (2007). Transforming Object-Oriented Models to Process-Oriented Models. link.springer.com https://link.springer.com/chapter/10.1007/978-3-540-78238-4\_15#:~:text=In%20an %20object%20model%2C%20a,flow%20dependencies%2C%20resources%2C%20etc.

Scacchi, Walt. (February, 2001). (Process Models in Software Engineering)). ics.uci.edu. https://www.ics.uci.edu/~wscacchi/Papers/SE-Encyc/Process-Models-SE-Encyc.pdf