# CS 255 Model Application Short Paper

Francis Cottrell-Eshaghi

francis.cottrell-eshaghi@SNHU.edu

Southern New Hampshire University

## Process Model Application

In the context of the DriverPass scenario, applying a process model is crucial for effectively designing and developing the system. One process model that can be applied is the Waterfall model. The Waterfall model is a linear and sequential approach to software development, consisting of distinct phases: Requirements, Design, Implementation, Testing, Deployment, and Maintenance.

In the Requirements phase, the project team would thoroughly gather and document the client's needs, as outlined in the interview transcript. This would include identifying functional and non-functional requirements, user roles, and system goals. For example, requirements such as online driver training, reservation management, and security measures would be documented in detail.

In the Design phase, the team would create detailed system specifications based on the gathered requirements. This phase involves designing the system's architecture, user interface, database structure, and data flow diagrams. For instance, the team would design the user interface according to Liam's sketch, ensuring it meets user needs.

In the Implementation phase, the actual coding and development of the system would take place. Developers would follow the design specifications and create the system's functionalities, including user registration, reservation management, and connection with the DMV for updates.

The Testing phase would involve rigorous testing of the system to ensure it meets the specified requirements and functions correctly. This would include unit testing, integration testing, and user acceptance testing to verify that all features work as expected.

Once testing is successful, the system would move to the Deployment phase, where it is made accessible to users. This includes configuring the cloud-based infrastructure, ensuring security measures are in place, and providing training to IT personnel for system maintenance.

Finally, in the Maintenance phase, the system would be continuously monitored, updated, and improved. This phase is essential for staying compliant with DMV regulations, adapting to platform updates, and addressing any issues that may arise.

## Object Model Application

In addition to the process model, applying an object model to the DriverPass scenario is equally important. An object model, such as Unified Modeling Language (UML) class diagrams, helps represent the data structure and relationships within the system.

For example, the class diagram would include entities like "Customer," "Lesson," "Package," and "Instructor," with associations depicting how they relate to each other. Attributes and methods of each class would be defined, specifying what data is stored and what actions can be performed.

The "Customer" class would include attributes such as "First Name," "Last Name," and "Phone Number." Associations would show that a "Customer" can have multiple "Lessons" and "Packages."

The "Lesson" class would have attributes like "Date," "Time," and "Instructor," and associations would link it to the "Customer" class, indicating which customer is taking which lesson.

This object model provides a clear representation of the system's data structure, making it easier for developers to understand how different elements of the system interact. It also serves as a blueprint for database design, helping in the creation of efficient database tables.

## Process and Object Model Comparison

The Waterfall Process Model, characterized by its structured and sequential approach, offers several advantages for a project like DriverPass. It ensures a systematic progression through well-defined phases, beginning with requirements gathering and culminating in deployment and maintenance. This structure is particularly beneficial when dealing with clearly outlined project objectives, as it facilitates task management and resource allocation. Moreover, the Waterfall model emphasizes comprehensive documentation at each stage, providing a detailed record of requirements and design decisions. However, its rigidity can be a drawback when adapting to changing project dynamics, and the late-stage user feedback may result in costly modifications if initial requirements are misinterpreted.

Conversely, object modeling, exemplified by UML class diagrams, provides a visual representation of the system's data structure and relationships, offering advantages in understanding and communication within the development team. In the DriverPass context, this modeling approach allows for the efficient design of database tables by serving as a blueprint. However, it may become complex and challenging to manage in more intricate systems, and its successful application often requires specialized knowledge in modeling languages like UML.

Considering these aspects, an integrated approach that combines the strengths of both models could be advantageous. The Waterfall model can guide the sequential progression of the project phases, ensuring a structured and documented development process. Meanwhile, object modeling can complement this process by providing a visual representation of the data structure, facilitating efficient database design. Such a hybrid approach may strike a balance between structured project management and effective data modeling, addressing the unique needs of the DriverPass scenario.