In the object-oriented methodology, the objects of the program become the focus. In this system, it is understood that the system is made up of interacting objects. The goal is to understand the similarities and differences between objects and how to manipulate them according to what we want. First we observe and analyze the system we will be working with, then we identify what the objects within a system are, what is their relationship to other objects within the system, how they behave with those objects and finally we develop them into a working program.

The first phase in object oriented development is analyzing the system. When analyzing the system in this methodology, the developer first talks to the people who use the system to note the requirements and get some understanding of what the system’s functions are. From there, the analyst takes this information and develops a model what the desired product should do (keeping the project requirements in mind).

Following this, we begin to decide on the architecture of the system we will develop. The way the architecture is designed is by looking at its contents as sub-systems within a system all interacting with each-other. Requirements are set for each sub-system. Then, we identify the objects within the system and how it will relate with other objects; In other words, we begin to design what our objects will do and what it will look like.

The way we design objects is by first creating a class, or a collection of similar objects like balls, cars, books. Then we sort out to include those objects that are similar in their characteristics like for cars (length, author, publisher, genre, etc.). We also need to observe the inheritance of the object, which really only refers to its ability to be reused. If for example I have created the object car with the attributes of wheels, motor, windows, etc.) I could reuse this same object for semi-trucks, construction vehicles, maybe even motorcycles.

We then move on to the implementation phase. This is the phase where the developers take the design ideas for the object and create the code for it. This is where the object analyzing that was done earlier comes into play. Programmers in this phase need to be aware of what the attributes, functionality, and relation of the objects are. The clearer these aspects are allowing the objects to be developed faster more accurately, the designs are more easily understood, and changes can easily be made.

When creating a new system that you know will need constant tweaking and updating, I believe object oriented programming is the perfect methodology to implement. It is even more so if the system admin will change hands frequently.

Free Tutes . *Object Oriented Life Cycle Model*. Retrieved from: https://www.freetutes.com/systemanalysis/sa2-object-oriented-methodology.html