**Title: The Role of Use Cases in Requirements and Analysis Modeling**

**Author: Hasan Gomaa and Erika Mir Olimpiew**

**Article Summary:**

The article Role of use cases in requirements and analysis modeling lays emphasis on use case model. It can also be described as Black Box model because of the fact that it helps in observing the external view of the system. There are 2 perspectives of viewing a software system. One described above is black box model and the other one is known as White box model. White Box model reveals out the internal details of the system which shows object and class, internal code of the system. For requirements modeling, black box comes into picture where we have various diagram beginning from the use case model to interaction model, context class model, state charts and system test model. For Analysis modeling we need to look deeper into the system with its internal workings. Object interaction model, Entity Class model and Integration models are used for white box view. All diagrams in black box model are the basis for white box models, so there exist a relationship between all the models in requirements phase to Analysis phase. To identify black box system and white box system there exist a Meta model which discriminates between the two with the help of stereotypes.

**Article Analysis:**

Author Gomaa and Olimpiew has covered use case modeling which is the basis for starting the software building process in modern era. First step in the requirement modeling begins with the development of use cases which indicates the black box model. As the name black box itself says that it reveals out on the external parameters of the system. This paper presents how the use cases are in line with other models for software analysis and testing models. There also exist 4+1 view model which has 4 different views to see a software architecture.

Author then explained an approach with black box view which considers only the external details of the software system such as interfaces, input, outputs, actors and any hardware devices participating in the software development. Following are the black box views explained by author and their relation with the use case:

1. Use Case Model: In UML use case model incorporates use case diagrams. There can be several use cases in a single use case diagram. There are the actor who participate in the use cases and are external to the system. Use case description includes several details such as pre-condition, post-condition, flow of events, alternate flows etc.
2. System Interaction Model: System level sequence diagram is drawn from the main sequence of use case model description. Interaction with system of all the actors is exposed out in interaction diagrams. As it reveals out only the external interaction, input and outputs so it is said to be in black box view of the system.
3. System Context Class Model: This model represents external classes of actors, devices, hardware etc. which interacts with the system. With regards to use case diagrams it enables users to help with extra information and particularly useful for developing real-time systems.
4. Statecharts: Statecharts falls into the black box view of system as they consider external inputs as the events which causes the change of state of the system and then gives the output. Events corresponds to input sequence and actions to output sequence of use case diagrams.
5. Test model: Test model comprise of test template, steps, parameters which are used for testing the system as black box. Author has explained how this model relates to other previous models of black box view which is shown below in simple terms:

Forms

Interaction Diagram Test Model

Forms

Sequence Diagram Test Steps

Message Parameters Test Parameters

Forms

System State Variables

Forms

Precondition Test Condition

Forms

Postcondition Expected Output Value

Proceeding onto further with the paper author explained another view of the system known as white box view which is used in next phase of SDLC known as Analysis Modeling. Here we analyze use cases developed in requirements modeling of the system with the help of classes and objects. The major steps is to identity the objects using object structuring criteria which using stereotypes to distinguish between the objects. There are various types of software objects such as entity objects which stores data, boundary objects which communicated with external and internal inputs or interfaces, control objects which controls the system or helps in decision making.

Three White box models author presented in the paper are as follows:

1. Object Interaction model: Use case are realized according to the objects along with series of interactions between the objects. There should be complete correspondence between the input and output message sequence in state dependent control object and event in the statecharts.
2. Entity Class Model: As explained earlier above, these are data intensive systems. Main purpose is to hide or encapsulate data. Entity classes are again determined from the use cases and their interaction is depicted out from interaction diagram.
3. Integration Test Model: Integration testing is done which includes the validation of interfaces between the objects. Test Template of black box becomes the test driver and Call logging traces is done with the help of sequence diagrams.

Author at last explained Relationship between use cases and other views of the system and presented Hotel system reservation example with all the models of the system and test template.

**Article Critique:**

This article laid stress on use cases and go very well with the title of the paper. As per my understanding of the paper below are strengths and weakness:

**Strengths:**

* Author explained very important views of the system i.e. black box and white box models which is very important for everyone in the software industry to understand.

After reading this paper it is easy to distinguish between black box and white box system models.

* Use case which forms foundation for the software to be developed are highlighted in this paper as every next activity must correspond to use cases. Because use cases are the functional requirements of the system.
* It is easily understood from this paper that how much requirements are important for developing a system and use cases drawn are to be 100% according to requirements otherwise the system to be built will not be upto what it is meant for.

**Weakness:**

* This paper should have included a little more information on Meta Model.
* This paper could have been shorter in length or in terms of word count, as at some places there is repetitive information regarding use case mapping to all other diagrams.
* This research paper is very general, does not cover white box or classes and objects in depth.