**Use case Diagram:**

UML is a standard language for specifying, visualizing, constructing, and documenting the artifacts of software systems.

UML was created by Object Management Group (OMG) and UML 1.0 specification draft was proposed to the OMG in January 1997.

OMG is continuously putting effort to make a truly industry standard.

* UML stands for **U**nified **M**odeling **L**anguage.
* UML is a pictorial language used to make software blue prints.

# UML Modeling Types:

It is very important to distinguish between the UML model. Different diagrams are used for different type of UML modeling. There are three important type of UML modeling’s:

# Structural modeling:

Structural modeling captures the static features of a system. They consist of the followings:

* Classes diagrams

Structural model represents the framework for the system and this framework is the place where all other components exist. So the class diagrams the part of structural modeling. They all represent the elements and the mechanism to assemble them.

But the structural model never describes the dynamic behavior of the system. Class diagram is the most widely used structural diagram.

# Behavioral Modeling:

Behavioral model describes the interaction in the system. It represents the interaction among the structural diagrams. Behavioral modeling shows the dynamic nature of the system. They consist of the following:

* Activity diagrams
* Sequence diagrams
* Collaboration diagrams
* Use case diagrams

All the above show the dynamic sequence of flow in a system.

**Use Case Diagram:**

Use case diagrams are considered for high level requirement analysis of a system. So when the requirements of a system are analyzed the functionalities are captured in use cases.So we can say that uses cases are nothing but the system functionalities written in an organized manner. Now the second things which are relevant to the use cases are the actors. Actors can be defined as something that interacts with the system.

The actors can be human user, some internal applications or may be some external applications. So in a brief when we are planning to draw an use case diagram we should have the following items identified.

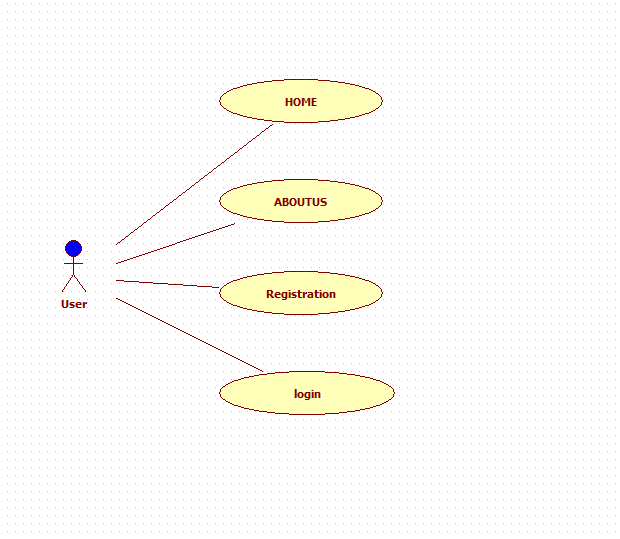
* Functionalities to be represented as an use case
* Actors
* Relationships among the use cases and actors.

Use case diagrams are drawn to capture the functional requirements of a system. So after identifying the above items we have to follow the following guidelines to draw an efficient use case diagram.

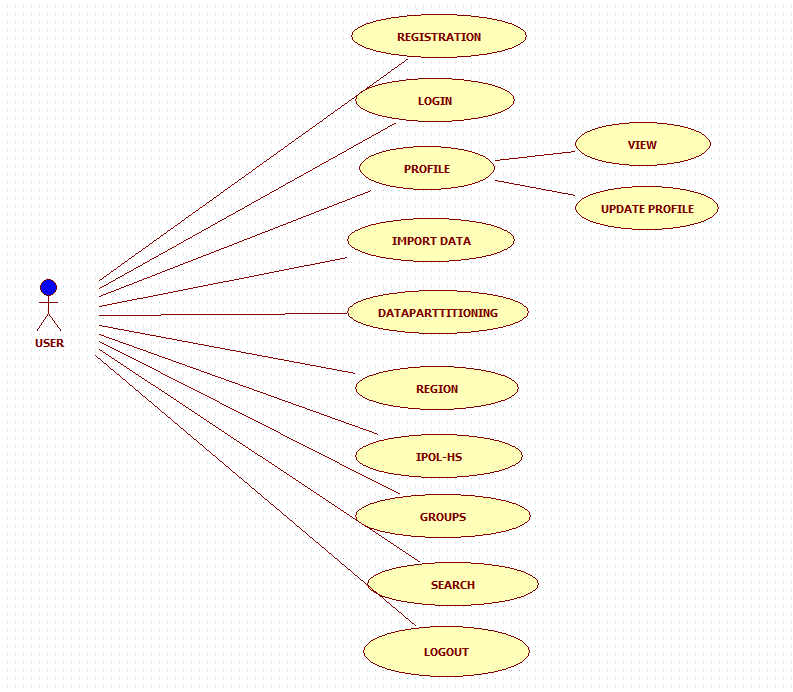
* The name of a use case is very important. So the name should be chosen in such a way so that it can identify the functionalities performed.
* Give a suitable name for actors.
* Show relationships and dependencies clearly in the diagram.
* Do not try to include all types of relationships. Because the main purpose of the diagram is to identify requirements.
* Use note when ever required to clarify some important points…

The main purpose of a use case diagram is to show what system functions are performed for which actor

**Homepage usecae diagram**

****

**User usecase diagram**

****