Treatise "Use Case". "User Scenario", and the "User Story"
By Dale C Clarke MBCS

A **Use Case** defines the interactions between external actors[**] and the system under consideration to accomplish a goal (task).

There is no universally accepted standard way to write the content of a Use Case, and different formats work well in different cases. A Use Case defines the interactions between external actors and the system under consideration in order to accomplish a goal. Actors must be able to make decisions, but need not be human: "An actor might be a person, a company or organization, a computer program, or a computer system — hardware, software, or both.

For example:
Use Case AA: User authenticates with id and password.

Use Scenarios AA:
1. ID is recognized, password is correct. ("prevalent" scenario)
2. ID is recognized, password is incorrect.
3. ID is recognized, password is incorrect for third time.
4. ID is recognized, password reset requested.
5. *ID is not checked, password is not checked, and server is down Sunday*. (from story)

Use Scenarios (scenarios) are *ALL* the various ways that task can play out, hopefully totally inclusive. So - every Use Case has one or more Scenarios. The Use Case is the abstract; the Use Scenarios are a catalog of all possible instances of that abstract task.

The **User Story** is the high level or a detail articulation of a scenario known by a user, as told in their own words, that can be found in, or added to the requirements/needs, or in some cases, edited against. It may contain one or more use cases and scenarios and may be repeated by one or more actors.

I have always thought of Use Cases as a way of defining or decomposing requirement goals in a standard structured narrative way for the clients/actors/users in their terms. Theoretically, every Use Case and scenario should have some user story to support it, although some have no user-facing effect and are known only to the developer. With regard to the above, what if the user said "But what if they try to login on Sundays when the server is down?" We have discovered from this user story another Scenario (AA/5) for the authentication task, and some additional requirements.

The User Story can describe any level, to wit.

User Story 1 – "To get on the system I enter my ID and a password" Results in Use Case AA.
User Story 2 – "I logon to the system by entering my ID and my password and when the system validates my password it logs me in" results in Use Case AA and Use Scenario AA/1

<u>User Story 3</u> – "What if it is Sunday when the server is down and I try to logon with my ID, password combination?" results in User Scenario AA/5.

Summary

There isn't an exact, universally accepted, definition of any of these terms documenting requirements. It all varies a little bit (sometimes a lot) from methodology to methodology, from client company to client company and from system to system. There must be documentation!

Your best bet is to find an effective accepted example already in place for your current project and follow it. I always ask the contract officer or hiring manager for an accepted, approved set of documents from a previous system deployment. This often negates waste of resource for approvals.

If you are creating a new system, you can find definitions of different types of use cases for whatever system or methodology you prefer -- Just pick the pattern that seems to communicate your intentions best. Expect that there will be the added time in the schedule for facilitation step of QA and compliance review, education, and acceptance given that there is no precedence you can provide.

This lack of agreement on definition is why wise computer-off-the-shelf (COTS) methodology product developers have typically added a "Light" version of their product which touts the caveat "Use the apropos steps, and create the artifacts that are appropriate and feasible to the proper and accurate depiction of the requirements/needs and, so that schedule, development, testing, implementation, and deployment can be successfully completed; and, in a cost effective manner." I have paraphrased, of course.

A final caveat usually added in somehow is, "… in accordance with business policy." That is why there are often section titles as headers to the meaningless descriptive body statement, "Not Applicable."

Obviously, if policy decrees one methodology over all others, learn it.

Good luck…

** A Use Case defines the interactions between external actors and the system under consideration to accomplish a goal. Actors must be able to make decisions, but need not be human: "An actor might be a person, a company or organization, a computer program, or a computer system — hardware, software, or both." Actors are always stakeholders, but many stakeholders are not actors, since they "never interact directly with the system, even though they have the right to care how the system behaves. User, actor, client, and stakeholder who are involved in the goal are often used synonymously.