

# Usability

by Sophia

#### WHAT'S COVERED

Software is developed in multiple phases based on a particular software methodology. Depending on the size and scope of the application, the development process can become very complex. Welldesigned software is the result of teams who continually consider how usable their software is. In this tutorial, we will discuss usability and the aspects of what makes a software package useful and usable.

Our discussion will break down as follows:

### 1. Software User Interface

Recall that application software is described as software that does something useful. With regard to computer software, **usefulness** is described as the degree to which the uses of software can meet the goals they were designed to meet. The primary aspects of a software application's usefulness are its utility and usability. **Utility** describes the capacity of a software application to perform a task or set of tasks. The more tasks a software application is designed to perform, the more utility it has.

#### TERMS TO KNOW

#### Usefulness

The degree to which the uses of software can meet the goals they were designed to meet.

#### Utility

The capacity of a software application to perform a task or set of tasks.

## 2. Usability

**Usability** describes how easy it is for a user to interact with software to perform a task. However, as related to software development, usability describes a user-centric approach to software design. This approach to software development seeks to consider the concerns of the user at every phase of the development process, thus making it the most critical aspect of usability. Coincidentally, it is the aspect that is often overlooked in software design. In order to develop software in this way, development teams conduct frequent usability tests where users are able to provide feedback regarding the software's interface. The primary reason for developing software in this manner is to save time, money, and other resources toward the end of the development cycle. The idea is that if the user provides feedback at every phase of development, the

software can be continually adapted, until the best user-centric solution has been reached.

#### IN CONTEXT

A software development team is creating an app for a pizzeria to improve their online ordering interface. The goal is to make online ordering a quick, straightforward, and painless experience. The business believes that if customers enjoy using the online ordering system, they are more likely to become loyal customers and less likely to place orders at other pizzerias.

The development team selected a group of representative users to test the app during iterative design so that they can identify usability problems and address these issues in the next design iteration. The users downloaded beta versions of the app on their phones, and navigated through the app as they normally would as customers: they skimmed the menu for ingredients and pricing information, added items to their shopping cart, and placed their orders for delivery.

During the usability test, a common theme from users was that they wanted information in real time about the status of their delivery after placing an order. This prompted the development team to incorporate this feedback into their next iteration. New specifications and requirements are then developed to support delivery tracking in the app, so that customers can see on the app if their order is being prepped in the kitchen, or is en route to their location.

#### TERM TO KNOW

#### Usability

Describes how easy it is for a user to perform a task.

# 3. Dimensions of Usability

A user-centric approach to software development ensures that software will meet its goal of being useful, while still maintaining a large degree of utility. Listed below are the five dimensions of usability.

Dimension of Usability	Description	How This Dimension Is Tested
Learnability	Refers to how easy a user can adapt to the software interface Related to helpfulness of the interface Refers to training documentation/user guides	One way learnability is tested is to take a group of users and assign them to complete the same task within a software application. The time it takes each user to complete the task is used to determine the general learnability of the task.
Efficiency	Refers to how fast a user can complete their tasks Related to user interface navigation scheme, keyboard shortcuts, placement of on screen	One way efficiency is tested is to measure the time it takes to complete a task.

	controls	
Visibility	Refers to the overall effectiveness of the user interface Related to the efficiency with which a user can complete a desired task or set of tasks	One way usability is tested is to use the "hallway method." In this method, developers are focused on identifying the early stage interface problems. Testers randomly select people from within the company or general public and ask them to use the software product.
Error Management	Refers to the capacity of the user interface to deter user errors; if user does commit an error, can the user recover from the error based on the ease of the user interface	One way error management is tested is to ask people knowledgeable in IT or the related area to anticipate what might go wrong with the application, then make appropriate requirement changes, and then repeat so that error management is an iterative process.
Satisfaction	Refers to how enjoyable or pleasant the user interface is to use	One way satisfaction is tested is to use rating scales, such as Likert scales, for attitude or preferences. Likert scales ask users to rate their satisfaction or preferences on a scale, typically ranging from strongly disagree to strongly agree, with 3-5 options in between.

### 🕽 SUMMARY

As software is designed to be useful and to be used by people, a good part of software development is to consider the user during the development process. **Usability** in software development refers to designing with the **user** at the center of the process. In this tutorial, we discussed usability and the **dimensions of usability** as one of the ways in which software can be designed to be useful.

### TERMS TO KNOW

#### Usability

Describes how easy it is for a user to perform a task.

#### Usefulness

The degree to which the uses of software can meet the goals they were designed to meet.

#### Utility

The capacity of a software application to perform a task or set of tasks.