# SNS COLLEGE OF TECHNOLOGY

## Coimbatore-37.
## An Autonomous Institution

**COURSE NAME : 19CST201-Agile Software Engineering**

**II YEAR/ III SEMESTER**

**Topic: User Interface Design**

Ms.G.Swathi

Assistant Professor

Department of Computer Science and Engineering

- The visual part of a computer application or operating system through which a client interacts with a computer or software.

- It determines how commands are given to the computer or the program and how data is displayed on the screen.

**Types of User Interface**

There are two main types of User Interface:

- Text-Based User Interface or Command Line Interface

- Graphical User Interface (GUI)

# Text Based User Interface

This method relies primarily on the keyboard. A typical example of this is UNIX.

Advantage:

Many and easier to customizations options.

Typically capable of more important tasks.

Disadvantages

Relies heavily on recall rather than recognition.

Navigation is often more difficult.

# Graphical User Interface

- GUI relies much more heavily on the mouse. A typical example of this type of interface is any versions of the Windows operating systems.

**GUI Characteristics**

- Windows
- Icons
- Menus
- Pointing
- Graphics

**Advantage**

- Less expert knowledge is required to use it.
- Easier to Navigate and can look through folders quickly in a guess and check manner.
- The user may switch quickly from one task to another and can interact with several different applications.

**Disadvantages**

- Typically decreased options.
- Usually less customizable. Not easy to use one button for tons of different variations.

UI Design Principles

1 Structure 2 Simplicity 3 Visibility 4 Feedback 5 Tolerance
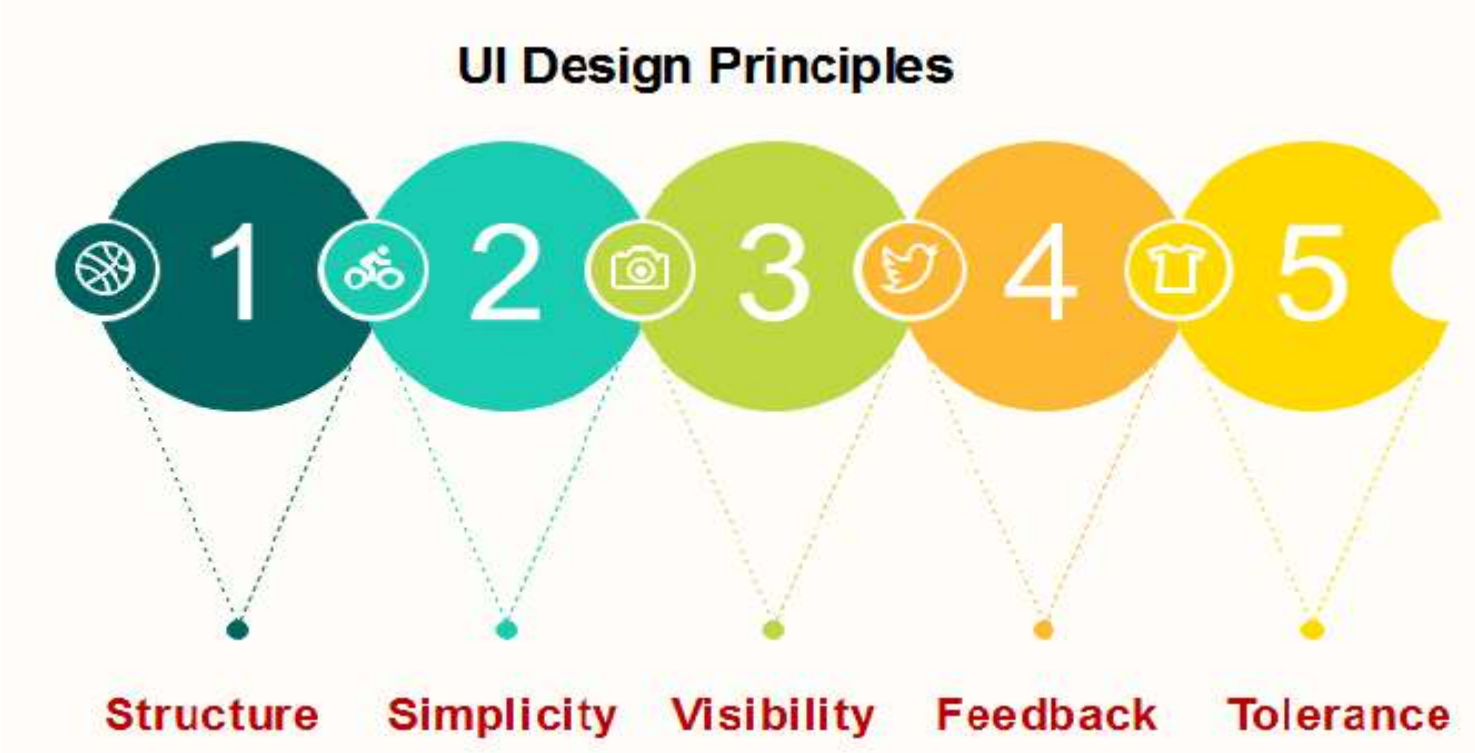
**Structure**:

• Design should organize the user interface purposefully, in the meaningful and usual based on precise, consistent models that are apparent and recognizable to users, putting related things together and separating unrelated things, differentiating dissimilar things and making similar things resemble one another.

• The structure principle is concerned with overall user interface architecture.

**Simplicity**:

The design should make the simple, common task easy, communicating clearly and directly in the user's language, and providing good shortcuts that are meaningfully related to longer procedures.

**Visibility**:

The design should make all required options and materials for a given function visible without distracting the user with extraneous or redundant data

**Feedback**:

The design should keep users informed of actions or interpretation, changes of state or condition, and bugs or exceptions that are relevant and of interest to the user through clear, concise, and unambiguous language familiar to users.

**Tolerance**:

The design should be flexible and tolerant, decreasing the cost of errors and misuse by allowing undoing and redoing while also preventing bugs wherever possible by tolerating varied inputs and sequences and by interpreting all reasonable actions.

# References

- Lisa Crispin, Janet Gregory, "Agile Testing; A Practical Guide for Testers and Agile Teams", Addison Wesley, 3rd Edition, 2015. 1

- Robert C.Martin, " Agile Software Development, Principles, Patterns and Practices", Prentice Hall, 2nd Edition, 2014.

- Alistair Cockburn, "Agile Software Development: The Cooperative Game", Addison Wesley, 2nd Edition,2015.

- Mike Cohn, "User Stories Applied: for Agile Software", Addison Wesley, 2nd Edition,2015.