SNS COLLEGE OF ENGINEERING

Kurumbapalayam(Po), Coimbatore – 641 107
Accredited by NAAC-UGC with ‘A’ Grade
Approved by AICTE, Recognized by UGC & Affiliated to Anna University, Chennai

Department of Information Technology

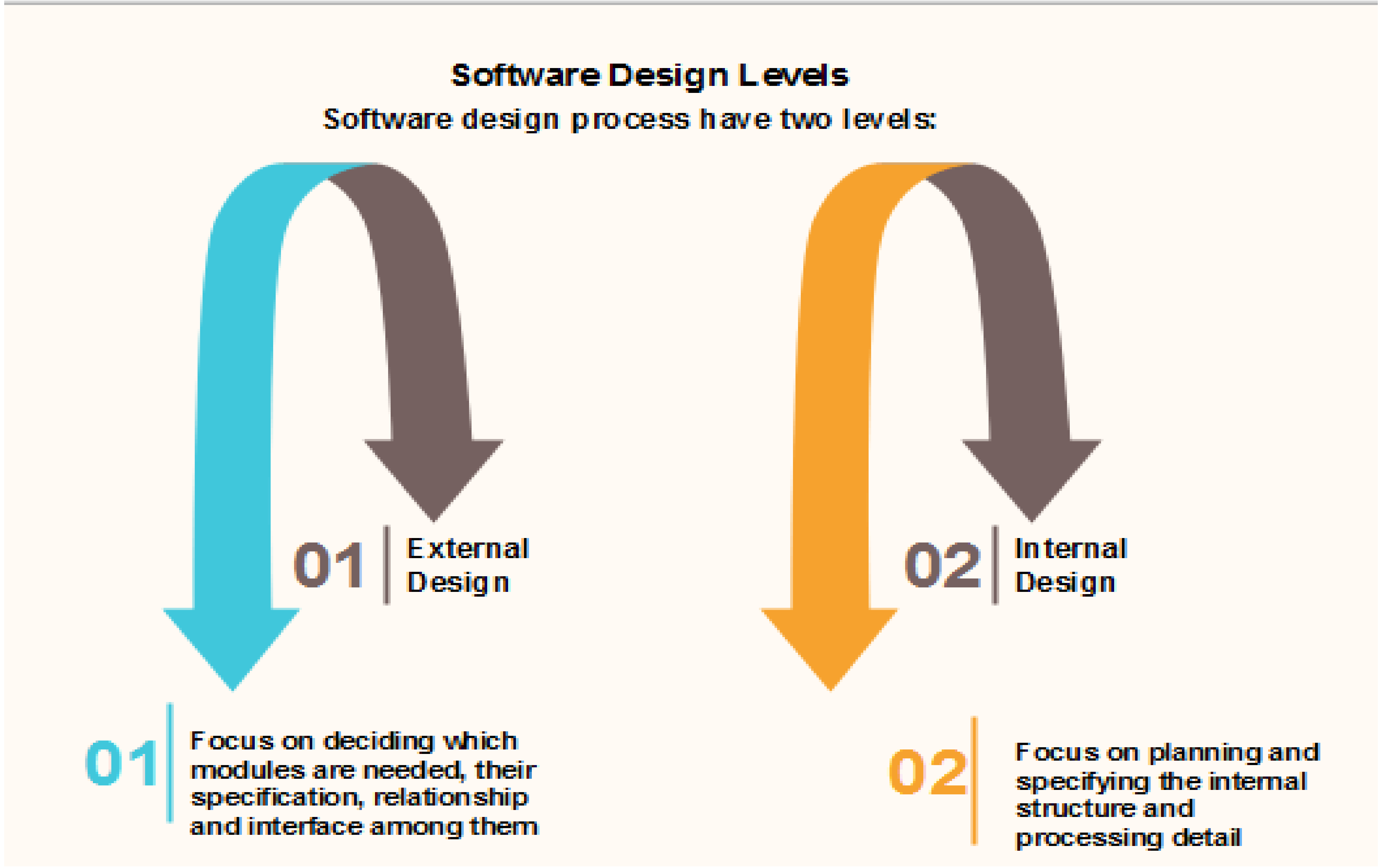
Course Name – Software Engineering

II Year / III Semester

DESIGN CONCEPTS AND PRINCIPLES

Software Design

- Software design is a mechanism to transform user requirements into some suitable form, which helps the programmer in software coding and implementation.
- It deals with representing the client's requirement, as described in SRS (Software Requirement Specification) document, into a form, i.e., easily implementable using programming language.



Software Design Levels

Software design process have two levels:

01 External Design

02 Internal Design

01 Focus on deciding which modules are needed, their specification, relationship and interface among them

02 Focus on planning and specifying the internal structure and processing detail

Objectives of Software Design

1.Correctness : Software design should be correct as per requirement.

2.Completeness : The design should have all components like data structures, modules, and external interfaces, etc.

3.Efficiency : Resources should be used efficiently by the program.

4.Flexibility : Able to modify on changing needs.

5.Consistency : There should not be any inconsistency in the design.

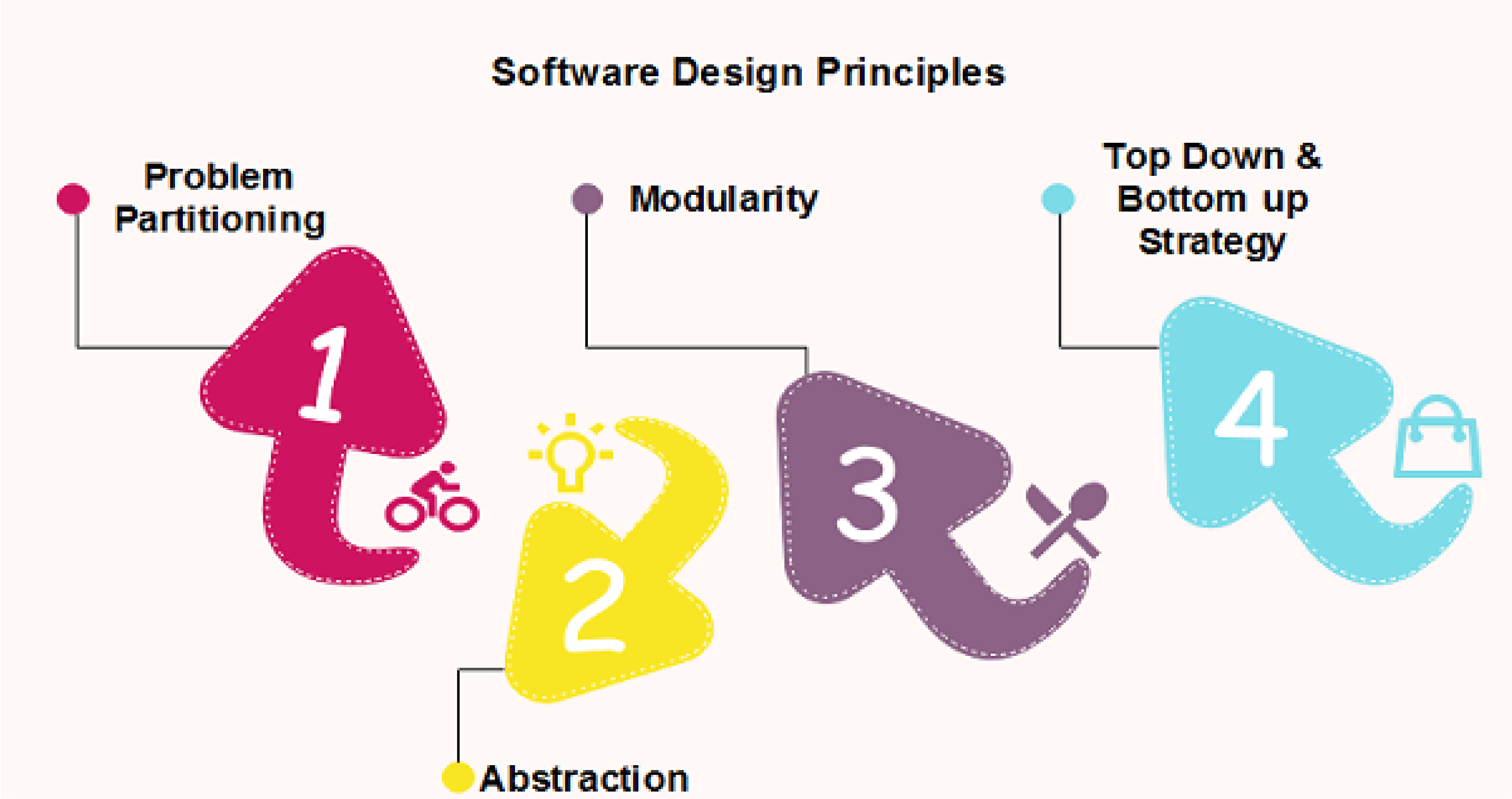
6.Maintainability: The design should be so simple so that it can be easily maintainable by other designers.

Correctness
Completeness
Efficiency
Flexibility
Consistency
Maintainability

Objectives of Software Design

Software Design Principles

- Software design principles are concerned with providing means to handle the complexity of the design process effectively.
- Effectively managing the complexity will not only reduce the effort needed for design but can also reduce the scope of introducing errors during design.



Software Design Principles

Problem Partitioning

- For small problem, we can handle the entire problem at once but for the significant problem, divide the problems and conquer the problem it means to divide the problem into smaller pieces so that each piece can be captured separately.
- For software design, the goal is to divide the problem into manageable pieces.

Benefits of Problem Partitioning

1. Software is easy to understand
2. Software becomes simple
3. Software is easy to test
4. Software is easy to modify
5. Software is easy to maintain
6. Software is easy to expand

Software Design Principles

Abstraction

An abstraction is a tool that enables a designer to consider a component at an abstract level without bothering about the internal details of the implementation. Abstraction can be used for existing element as well as the component being designed.

Here, there are two common abstraction mechanisms

1.Functional Abstraction

2.Data Abstraction

Functional Abstraction

i.A module is specified by the method it performs.

ii.The details of the algorithm to accomplish the functions are not visible to the user of the function.

Functional abstraction forms the basis for **Function oriented design approaches**.

Data Abstraction

Details of the data elements are not visible to the users of data. Data Abstraction forms the basis for **Object Oriented design approaches**

Software Design Principles

Modularity

- Modularity specifies to the division of software into separate modules which are differently named and addressed and are integrated later on in to obtain the completely functional software.
- It is the only property that allows a program to be intellectually manageable.
- Single large programs are difficult to understand and read due to a large number of reference variables, control paths, global variables, etc.

The desirable properties of a modular system are:

- Each module is a well-defined system that can be used with other applications.
- Each module has single specified objectives.
- Modules can be separately compiled and saved in the library.
- Modules should be easier to use than to build.
- Modules are simpler from outside than inside.

Software Design Principles

Strategy of Design

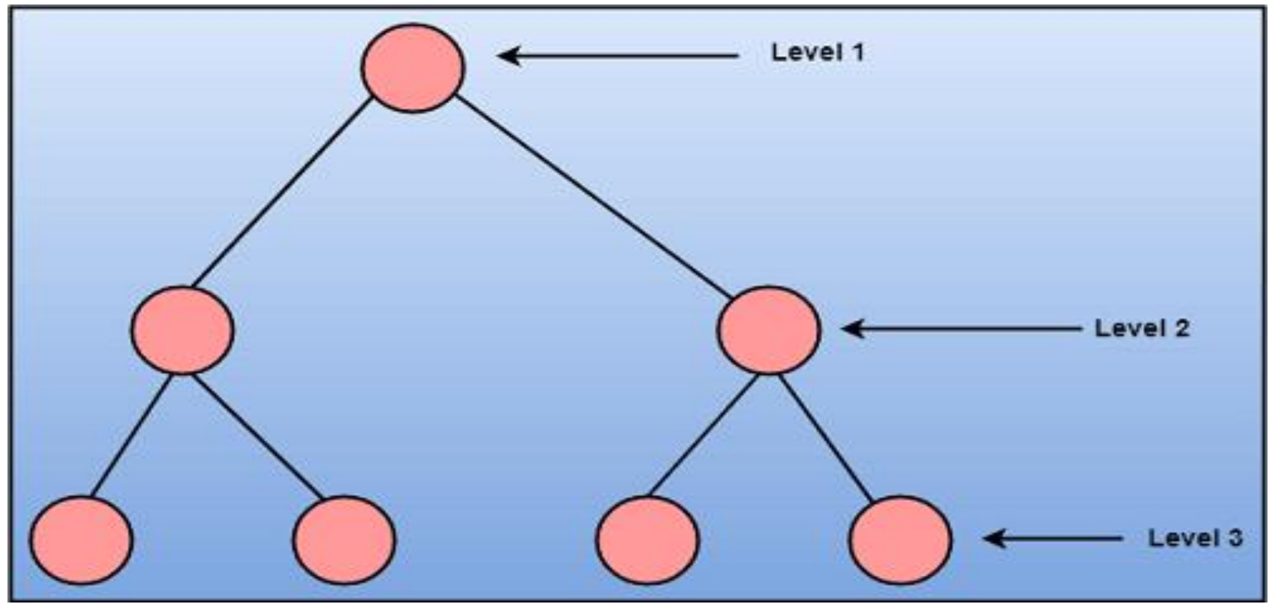
- A good system design strategy is to organize the program modules in such a method that are easy to develop and latter too, change.
- Structured design methods help developers to deal with the size and complexity of programs.
- Analysts generate instructions for the developers about how code should be composed and how pieces of code should fit together to form a program.
- To design a system, there are two possible approaches:

    1.**Top-down Approach**

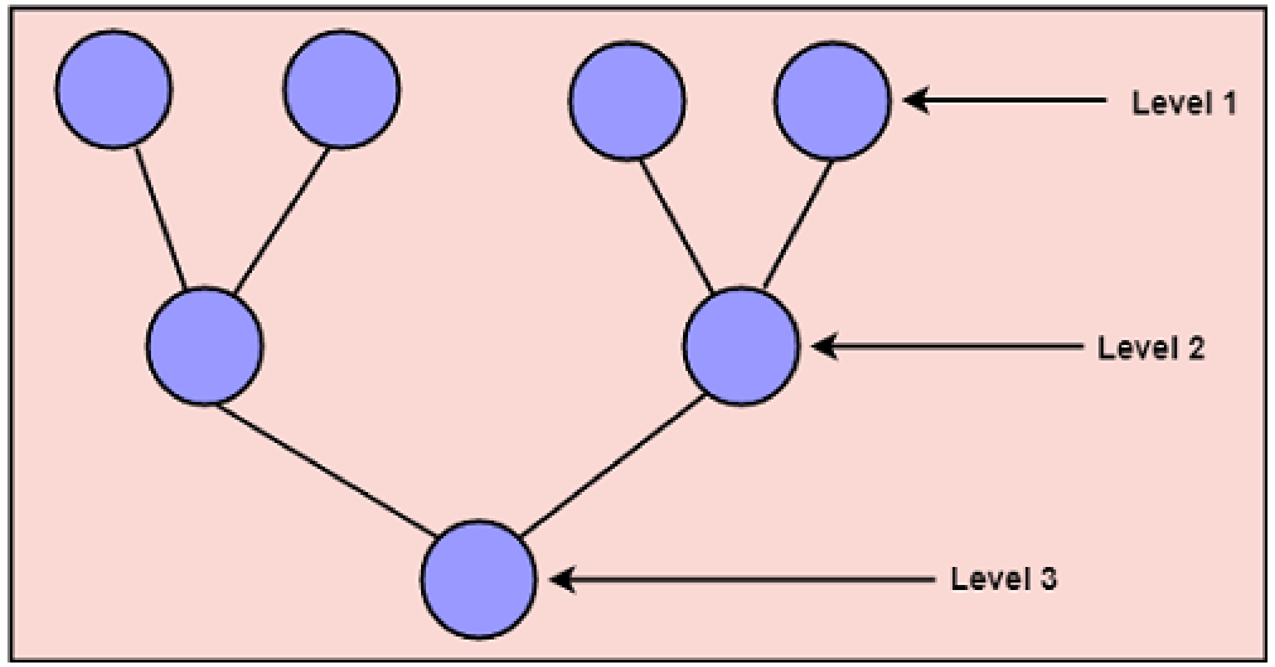
    2. **Bottom-up Approach**

Software Design Principles

1. Top-down Approach: This approach starts with the identification of the main components and then decomposing them into their more detailed sub-components.



2. Bottom-up Approach: A bottom-up approach begins with the lower details and moves towards up the hierarchy, as shown in fig. This approach is suitable in case of an existing system.



THANK YOU