

# Global Journal of Engineering, Design

## Concepts and Considerations of Software Design

### Cheng Chi\*

Department of Computer Science, City University of Hong Kong, Hong Kong, China

## DESCRIPTION

Software design is the process by which an agent develops a specification of a software artifact intended to achieve goals using a collection of simple building blocks and subject to limitations. Software design can be defined as either "the activity after requirements definition and before programming, in a stylized software engineering process," or "the activity involved in envisioning, framing, implementing, commissioning, and eventually altering large systems."

Planning a software solution and problem-solving are typical components of software design. This covers both high-level architecture design and low-level component and algorithm design.

#### Design concepts

The design concepts give the software designer a base from which to apply more complex techniques. Several crucial design concepts have emerged. They are as follows:

**Abstraction:** Abstraction is the activity or outcome of generalizing by decreasing the information content of a notion or an observable occurrence, usually in order to keep just information that is pertinent to a certain purpose. It is a representation of key elements without providing any context or justification.

**Refinement:** It is the elaboration process. A hierarchy is created by gradually breaking down a macro statement of function until programming language statements are reached. One or more instructions from a given programme are broken down into more specific instructions in each stage. Refinement and abstraction are complimentary ideas.

**Modularity:** The modules that make up software architecture are separated into sub-components.

**Software Architecture:** It speaks of the overall organization of the programme and the ways in which that organization ensures the conceptual consistency of a system. Regarding the project's desired outcome, such as performance, quality, schedule, and cost,

effective software architecture will provide a strong return on investment.

**Control Hierarchy:** A programme structure that illustrates how a programme component is organized and suggests a chain of command.

**Structural Partitioning:** Both a horizontal and vertical division of the programme structure is possible. For each key programme function, horizontal partitions establish distinct branches of the modular hierarchy. According to vertical partitioning, the programme structure should spread control and work from top to bottom.

**Data Structure:** It is a visual representation of the logical relationship between several data items.

**Software Procedure:** It concentrates on handling each module separately.

**Information Hiding:** Modules should be declared and created in a way that prevents information from being accessed by modules that don't require it.

#### Design considerations

When designing a piece of software, there are numerous factors to take into account. The goals and expectations that the programme is being developed to satisfy should be reflected in the priority of each aspect. Some of these aspects are:

**Compatibility:** The software can work with other products that are made to be interoperable with one another. A piece of software, for instance, might be backward-compatible with an earlier version of itself.

**Extensibility:** Without making significant changes to the underlying architecture, new features can be introduced to the software.

**Modularity:** The software that results from this has well defined, distinct components, which makes it easier to maintain. Before being combined to create the required software system, the components could first be created and evaluated separately. This

Correspondence to: Cheng Chi, Department of Computer Science, City University of Hong Kong, Hong Kong, China, E-mail: Chengchi2020@yahoo.com Received: 21-Feb-2022, Manuscript No. GJEDT-22-21490; Editor assigned: 24-Feb-2022, PreQC No. GJEDT-22-21490 (PQ); Reviewed: 11-Mar-2022, QC No. GJEDT-22-21490; Revised: 21-Mar-2022, Manuscript No. GJEDT-22-21490 (R); Published: 29-Mar-2022, DOI: 10.35248/2319-7293.22.11.148 Citation: Chi C (2022) Design Concepts and Design Considerations of Software Design. Global J Eng Des Technol.11:148 Copyright: © 2022 Chi C. This is an open-access article distributed under the terms of the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited. enables a software development project to divide the tasks.

**Fault-tolerance:** The software is resilient to component failure and capable of recovering from it.

Maintainability: A gauge of how straightforward it is to make

functional or bug changes. Modularity and extensibility can result in high maintainability.

**Reliability:** Under given circumstances and for a predetermined amount of time, the software can carry out a necessary function.