Experiment No.: 7
Identifying Design requirements for an Architecture for any specific domain.

Learning Objective: Student should be able to understand Design requirements for an Architecture for any specific domain.

Theory:

Software design:

Software design provides a design plan that describes the elements of a system, how they fit, and work together to fulfill the requirement of the system. The objectives of having a design plan are as follows −

To negotiate system requirements, and to set expectations with customers, marketing, and management personnel.

Act as a blueprint during the development process. Guide the implementation tasks,

including detailed design, coding, integration, and testing.
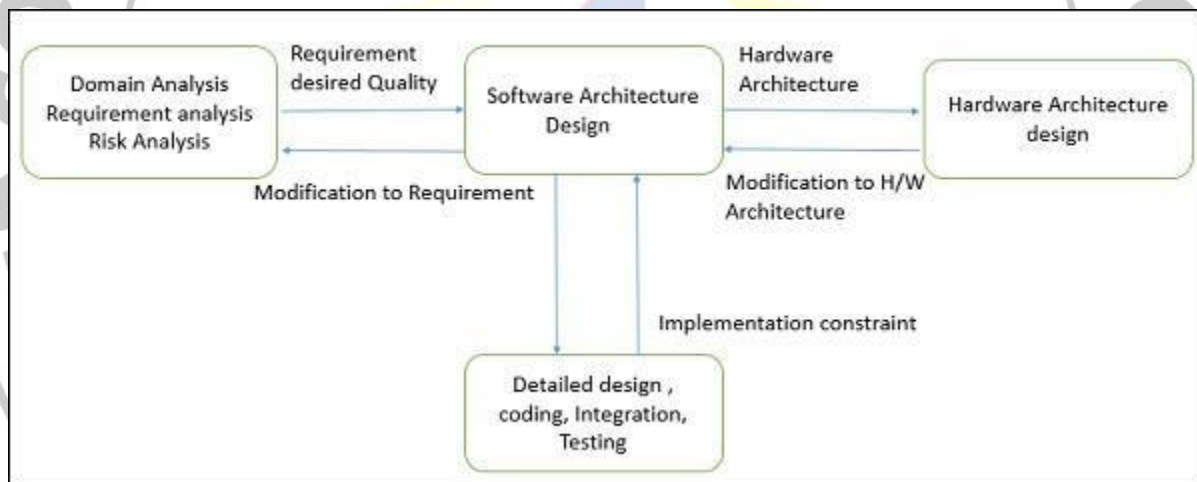
It comes before the detailed design, coding, integration, and testing and after the domain analysis, requirements analysis, and risk analysis.

Software design is responsible for the code level design such as, what each module is doing, the classes scope, and the functions purposes, etc. When used strategically, they can make a programmer significantly more efficient by allowing them to avoid reinventing the wheel, instead using methods refined by others already. They also provide a useful common language to conceptualize repeated problems and solutions when discussing with others or managing code in larger teams.

Major artifacts of the software design process include:

- **Software requirements specification.** This document describes the expected behavior of the system in the form of functional and non-functional requirements. These requirements should be clear, actionable, measurable, and traceable to business requirements. Requirements should also define how the software should interact with humans, hardware, and other systems.
- **High-level design.** The high-level design breaks the system's architectural design into a less-abstracted view of sub-systems and modules and depicts their interaction with each other. This high-level design perspective focuses on how the system, along with all its components, implements in the form of modules. It recognizes the modular structure of each sub-system and their interaction among one another.
- **Detailed design.** Detailed design involves the implementation of what is visible as a system and its sub-systems in a high-level design. This activity is more detailed towards modules and their implementations. It defines a logical structure of each module and their interfaces to communicate with other modules.

The purpose of an architecture schema or design record is to serve as a vehicle for software understanding by functioning as a collection point for knowledge about the components that make up a DSSA. In particular, the design record organizes

- domain- specific knowledge about components or design alter natives and
- Implementation in knowledge about alternate implementations,

The primary goal of a design record is to adequately describe the components in a reference architecture such that the application engineer can make design decisions and component selections without looking at inn implementations. The secondary goal of a design record is to provide information that the tools in the supporting environment can use. The design record data elements used by Loral Federal Systems phases in the software life cycle, include:

1. Name/type
2. Description
3. reference requirements satisfied,
4. design structure (data flow and control flow diagrams),
5. design rationale,
6. interface and architecture specifications and dependencies,
7. P D L (program Design Language) text,
8. implementation, 9. configuration and version data, a

10. test cases.
11. metric data,

12. access rights,
13. search points,
14. catalog information,
15. library and DSSA links, and hypertext paths
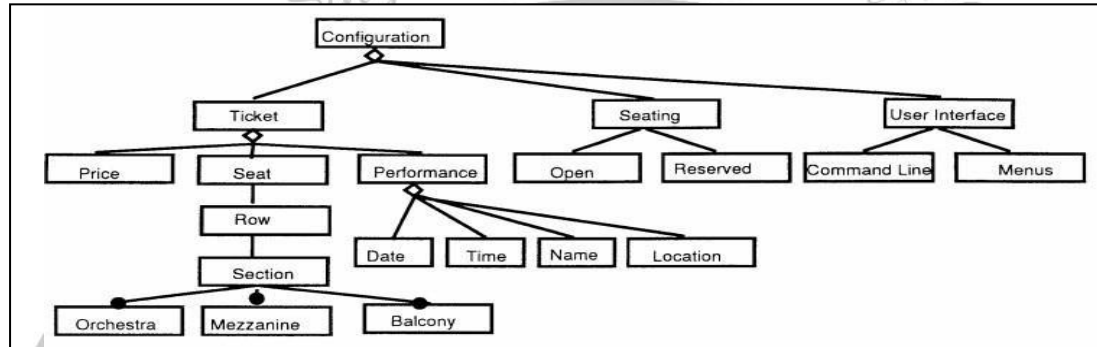
Many different programming formats incorporate the same essential elements. In all cases, the design programming fits within a larger context of planning efforts which can also be programmed. For design programming for a building, we propose a six-step process as follows:
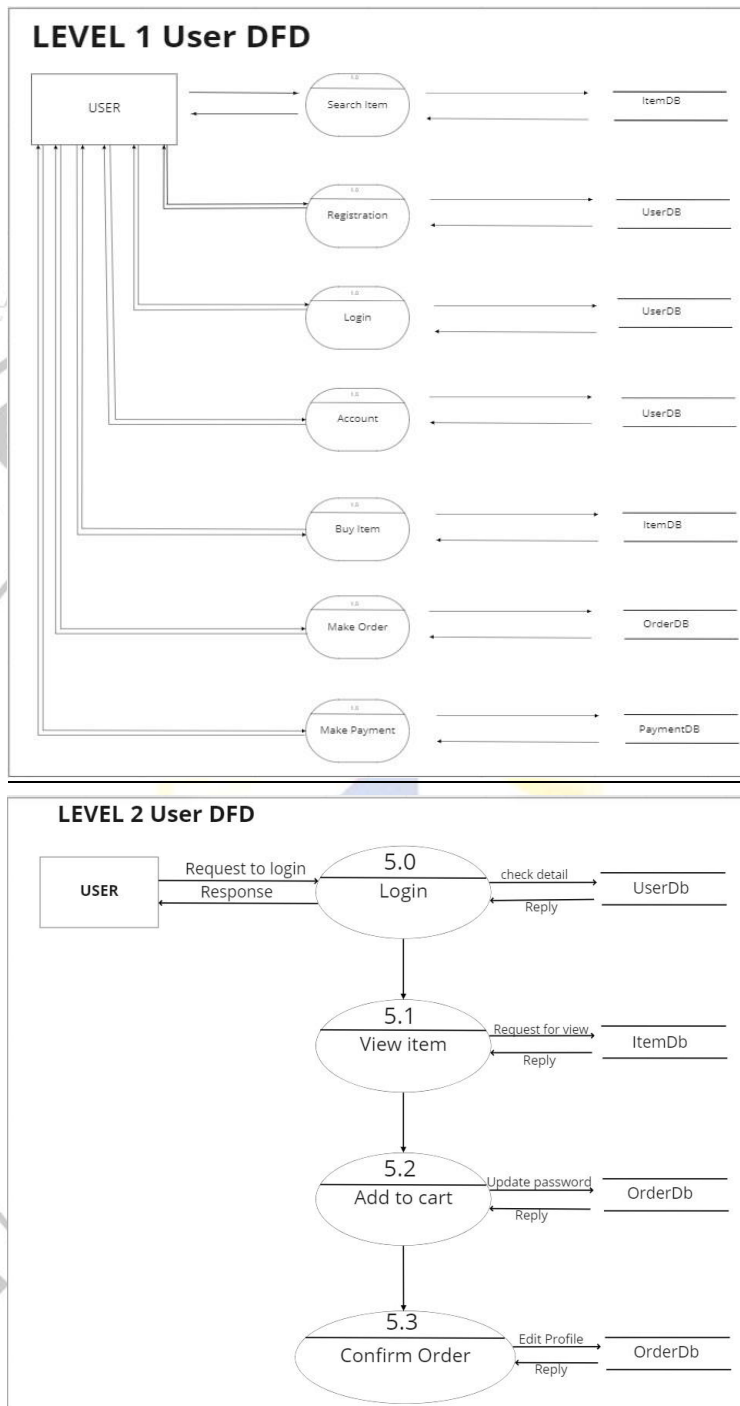
1. Research the project type
2. Establish goals and objectives
3. Gather relevant information
4. Identify strategies
5. Determine quantitative requirements
6. Summarize the program

Output:

Project Title: E Commerce System

LEVEL 1 User DFD

LEVEL 2 User DFD

Result & Discussion:

Learning Outcomes: Students should have been able to understand LO1:

Define software design.

LO2: Identify different design requirements for software. LO3:

Explain implementation of design requirements of software.

Course Outcomes: Upon completion of the course students will be able to understand design requirement of software Architecture.

Conclusion:

Viva Questions:

1. Define design requirement in software.
2. Explain different design requirements in software Architecture.
3. Explain the phases of programming design.
4. Explain any three design requirements.

For Faculty Use

Correction Parameters	Formative Assessment [40%]	Timely completion of Practical [40%]	Attendance / Learning Attitude [20%]	