Assignment-1
Software Engineering

Aaditya Sharma
19930217, IT-5B

(Q1) Requirement Engineering is the disciplined application of proven principles to describe a proposed system's behavior and its constraints.

Present state of methods

1) Requirements change
2) Over reliance on CASE tools
3) Tight project schedule
4) Barriers of communication
5) Market driven software development
6) Lack of resources

There are a number of ways of improving the present state of practice :-

1) Conduct a feasibility study. There needs to be technical feasibility for designing, communicating etc.
2) Purpose and focus of feasibility studies :-

a) Viability of product concept
b) project's vision statement
c) cost and schedules
d) Risk management.
e) Stable development work.

(Q2) → SRS: Software Requirement specification is a written text b/w an organisation and clients.

2) Contains the features and functionalities of products being manufactured.

3) Helps in understanding common goals and feature requirements of clients, stakeholders.

Advantages :-

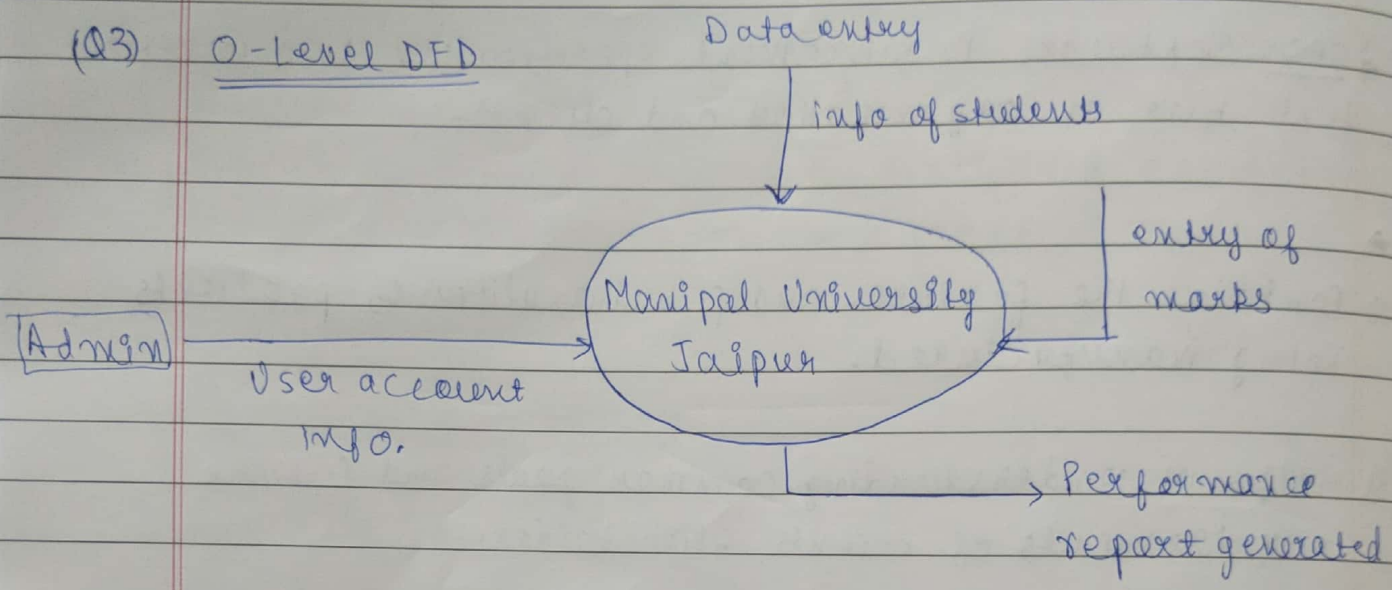
1) Makes the process of testing easy.
2) Reduces the development effort.
3) Minimizes confusion and misunderstandings.
4) Defines the requirements specification.

Since its internal details are not known and only its visible external behavior is recognized. Hence it is known as black box specification of a system.

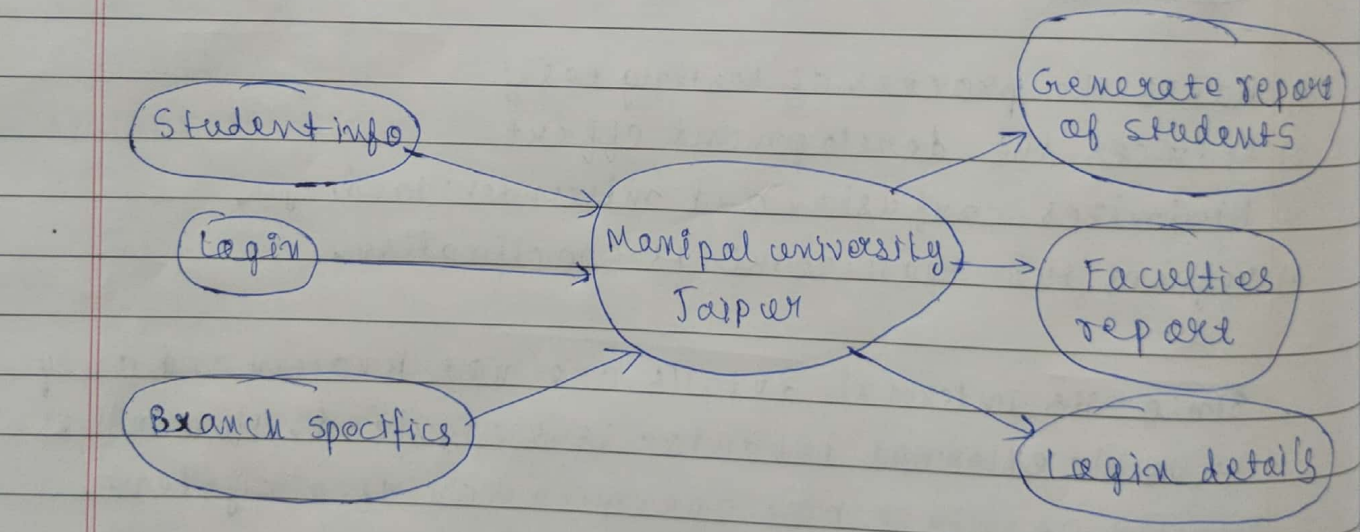
Important issues SRS must address:-

1) Issue of performance
2) Interfaces
3) Design constraints
4) Functionality.

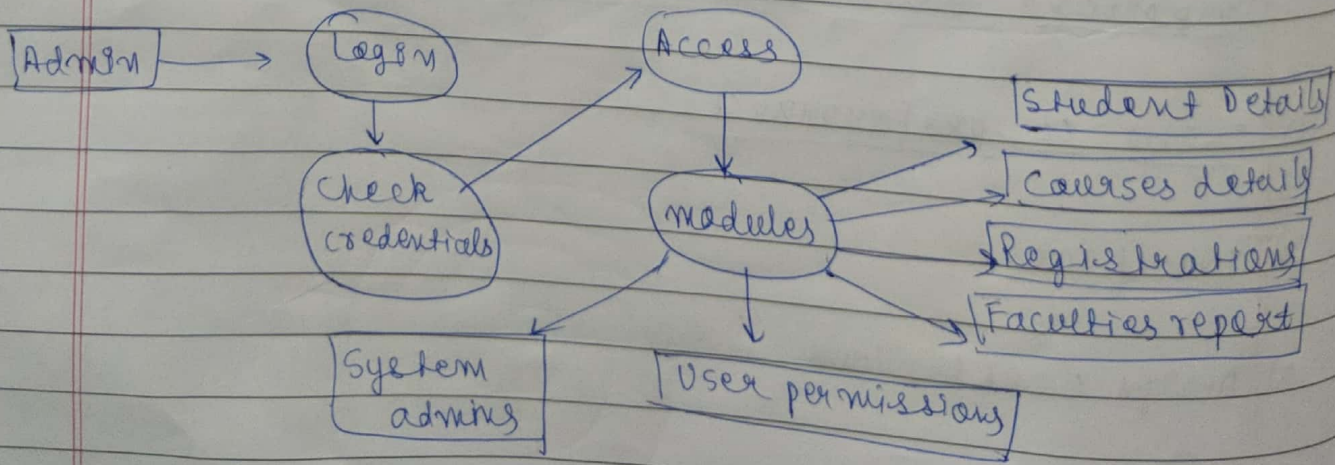
(Q3) 0-Level DFD



1-Level DFD



2-Level DFD



(Q4) Coupling can be defined as the degree of interdependence between various modules.

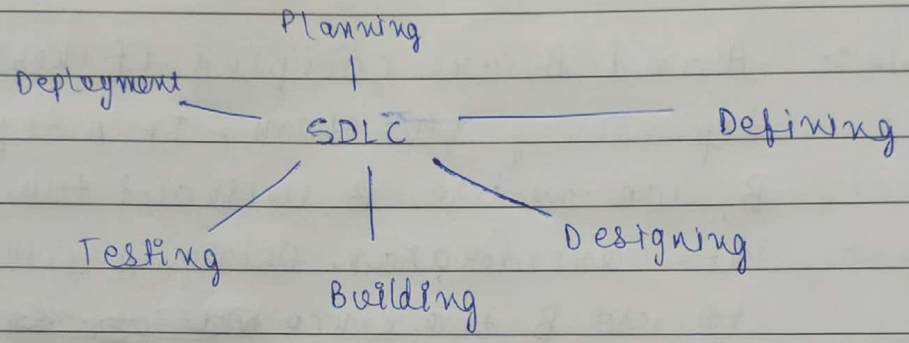
For example:- A and B are coupled if there's a dependency b/w them. If A depends on B, we can use B without taking A into consideration. Using A will imply us to use B too since ~~both are dependent~~, A is dependant on B.

No, we cannot remove coupling between two modules completely. What we can do though is to apply Decoupling which means loosening the existing coupling. Making sure that each component knows as little as possible about the other components.

(Q5)

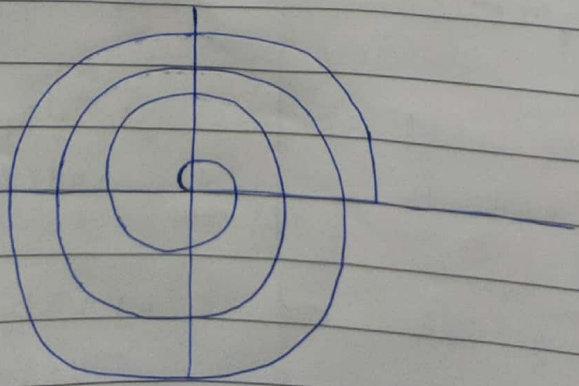
Flow Chart	Structure Chart
1) Represents flow of control in a program	1) Represents the architecture of a software.
2) Involves simple symbols	2) Complex symbols
3) Demonstrates the sequential ordering of inherent tasks	3) Suppresses the sequential ordering of inherent tasks
4) Hard to identify various modules.	4) Easy to identify various modules.

(Q6) SDLC is defined as a process with a detailed plan describing how to develop and maintain softwares. It aims at improving the quality of software and the process of development.



It is important to adhere to a properly defined methodology since it helps in providing better estimats and understanding the task in a better way.

(Q7) A software development process is filled with several phases. Spiral model comes in handy here since is contains several loops where each loop is representing a phase. Since, the number of phases keep changing, it becomes important for the project manager to develop number of phases and plan the development of the product using the spiral model



Features and uses of the spiral model :-

1) Risk Handling :- It handles the risks and its analysis at each phase of development.

2) Requirement flexibility :- Any changes can be made using the spiral model.

3) Beneficial to use in Bigger projects.

4) Satisfaction in performance

(08) Agile Model

Agile model refers to a group of development process. Agile model were designed to help projects adapt to change requests quickly.

Agile model incorporates an iterative development procedure. Hence, no long term planning.

With numerous advantages, agile model are an important part of software Engineering :-

1) Reduces total development time of the project
2) Making changes is easy due to the iterative development
3) less errors in programs.