

Assignment - 1

- Q What is Software Engineering and give its characteristics and detail description of software methodology and layered technology.

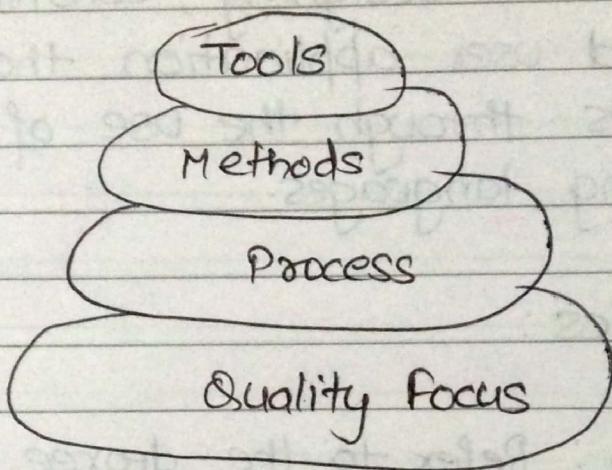
Ans. Software Engineering is the process of analyzing user needs and designing, constructing and testing end user application that will satisfy these needs through the use of software programming languages.

Characteristics :

1. **Functionality** : Refers to the degree of performance of the software against its intended purpose.
2. **Reliability** : Refers to the ability of software to perform a required function under given conditions for a specified period.
3. **Usability** : Refers the degree to which software is easy to use.
4. **Efficiency** : Refers to the ability of software to use system resource in the most effective and efficient manner.
5. **Maintainability** : Refers to the ease with which a software system can be modified to add

capabilities, improve system performance or correct errors.

6. Portability: Refer to the ease with which software developer can transfer software from one platform to another without changes.



1. Quality Focus: Correctness of the functions required to be performed by the software.
 - Maintainability of the software
 - Integrity i.e. providing security so that the unauthorized user cannot access information or data.
 - Usability i.e. the efforts required to use or operate the software.
2. Process:
 - It is the base layer or foundation layer for the software engineering.
 - The Software process is the key to keep all levels together.

- It defines a framework that includes different activities and tasks.

3. Methods

- The method provide the answer of all 'how-to' that are asked during the process.
- It provides the technical way to implement the software.
- It includes collection of tasks starting from communication, testing & support etc.

4. Tools

- It is an automated support for the software Development.
- The tools are integrated i.e., the information created by one tool can be used by other tool.

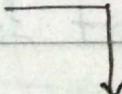
Assignment - 2

Q.1 Give Detail Explanation of all Process Models with suitable Diagram. Give its advantages and dis-advantages.

Ans. 1. Waterfall Model : It is a sequential approach where each fundamental activity of a process represented as a separate phase , arranged in linear order. In the waterfall model , you must plan and schedule all of the activities before starting working on them.

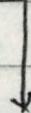
Requirement

Definition

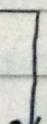


System and
Software

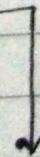
Design



Implementation
and Unit Testing



Integration
and system
Testing



operation and
Maintenance

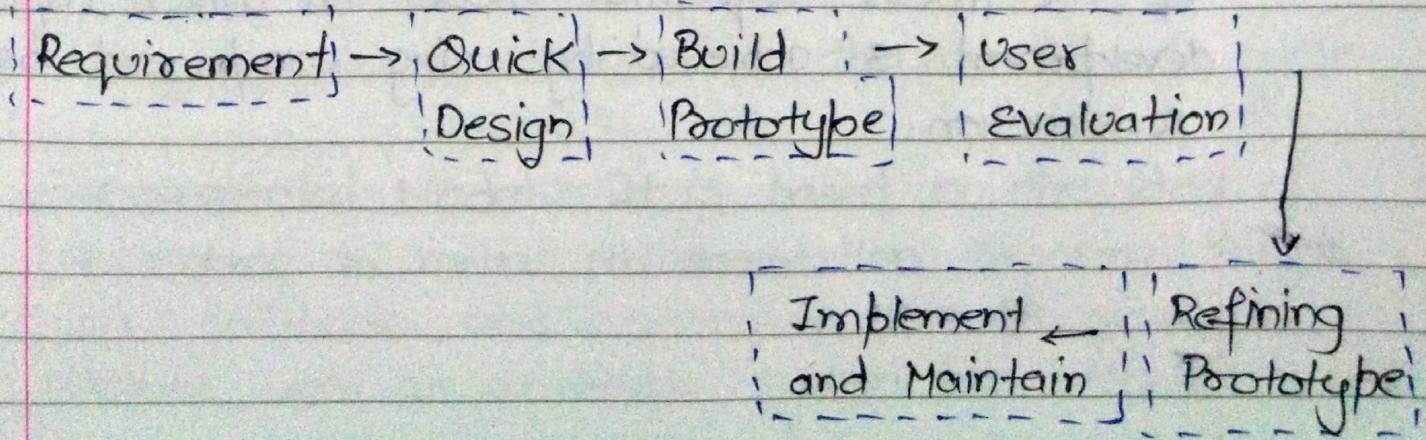
Advantages :

- Simple and easy to understand and use.
- Phases are processed and completed one at a time.
- Clearly defined stages.
- Easy to arrange tasks.

Dis-advantages

- No working software is produced until late during the life cycle.
- High amount of risk and uncertainty.
- Poor model for long and ongoing projects.
- It is difficult to measure progress within stages.

2. Prototyping Model : It is applied when there is an absence of detailed information regarding input and output requirement in the software. Prototyping model is developed on the assumption that it is often difficult to know all the requirement at the beginning of a project. It is usually used when there does not exist a system or in case of large and complex system where there is no manual process to determine the requirements.



Advantages

- Helps team member to communicate effectively.
- There will be hardly any chance of software rejection.
- Encourages innovation and flexible designing.
- No need for specialised experts to build the model.

Dis-advantages

- Prototyping is slow and time taking process.
- Prototyping may encourage excessive change request.
- Poor documentation because the requirements of the customers are changing.
- It is very difficult for software developers to accommodate all the changes demanded by the clients.

3. RAD (Rapid Application Development): It is an incremental software development process model that emphasizes an extremely short development cycle. The RAD model is a "high speed" adaptation of the linear sequential model in which rapid development is achieved by using component-based construction.

TEAM-1

TEAM-2

Business
Modelling

Data

Modelling

Business
Modelling

Data

Modelling

Process

Modelling

Application
Generation

Process

Modelling

Applic.

Generation

Testing

& Turnover

Testing

& Turnover

← → 60 - 90 days

Advantages :

- changing requirement can be accommodated.
- Progress can be measured.
- Reduced development time.
- Quick initial review occur

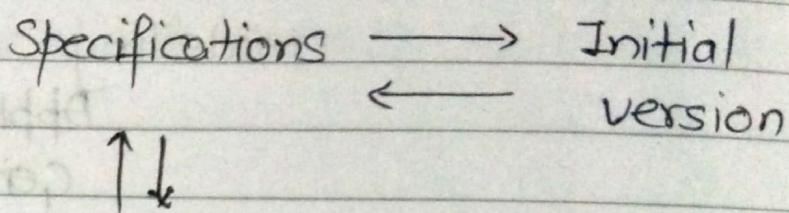
Dis-advantages

- Requires highly skill developers.
- High dependency on modelling skills.
- Management complexity is more.
- Requires user involvement through the life cycle.

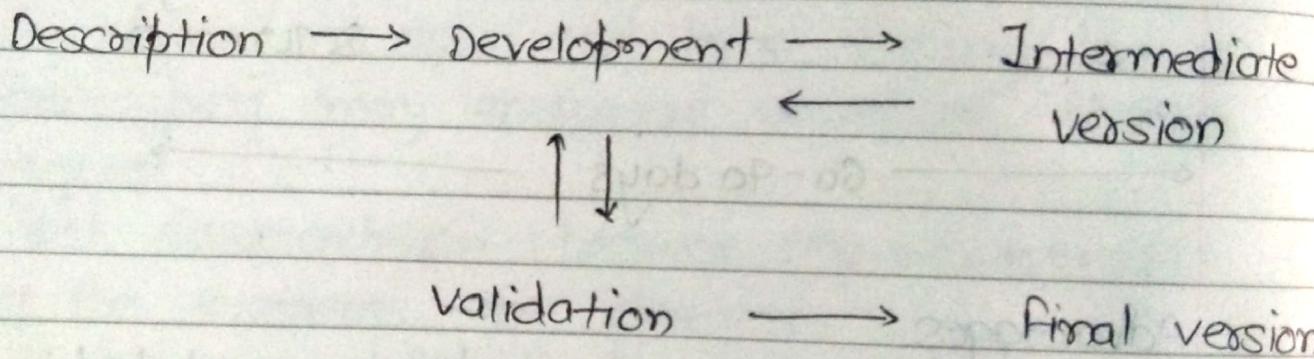
4. Incremental Model : It is based on the idea of developing an initial implementation, exposing this to user feedback, and evolving it through several versions until an acceptable system has been developed.

The activities of a process are not separated but interleaved with feedback involved across those activities.

concurrent
Activities



Outline



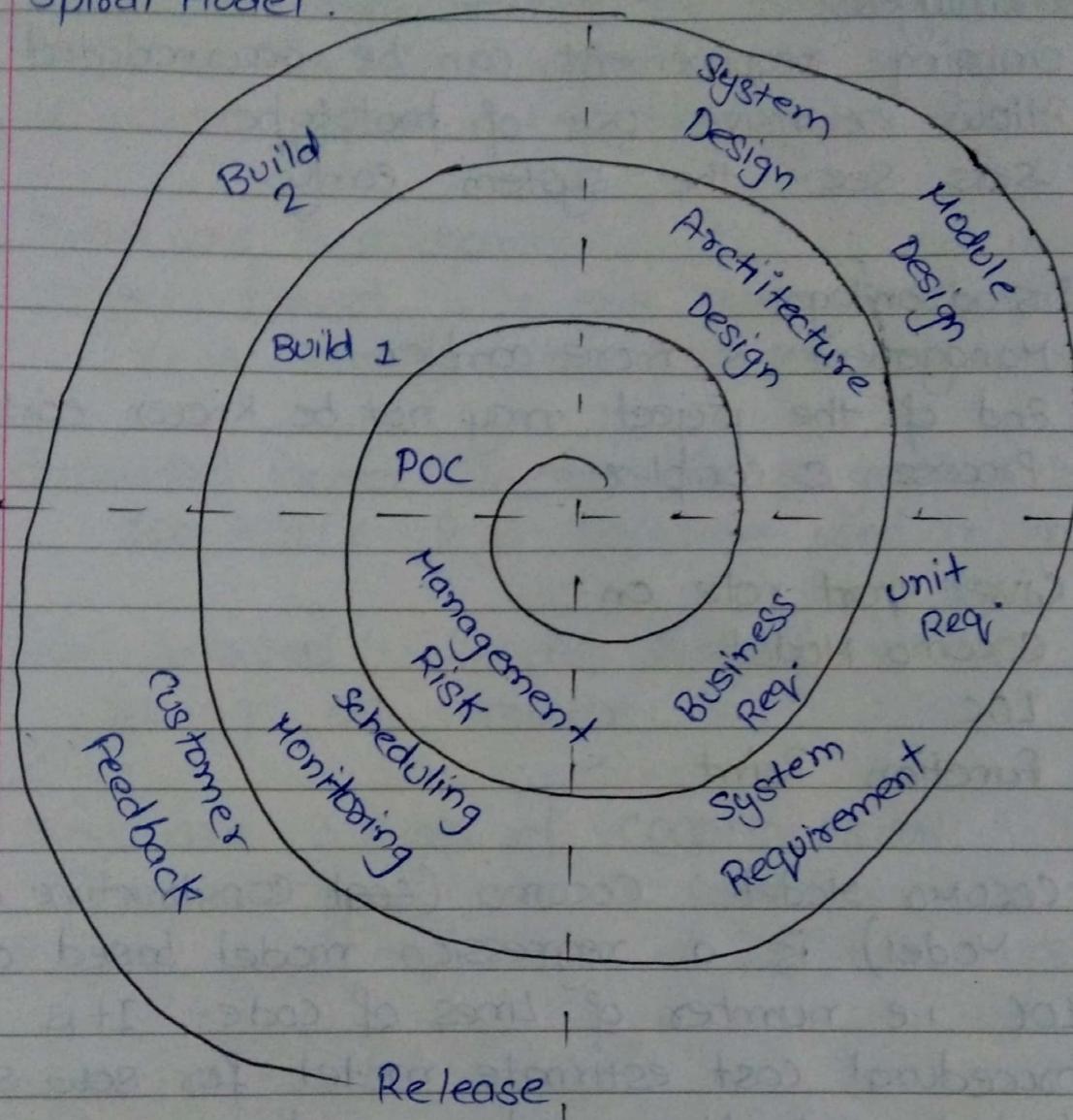
Advantages

- Generates working software quickly and early during the software life cycle.
- More flexible - less costly to change scope & requirements.

Dis-advantages

- Each phase of an iteration is rigid and do not overlap each other.
- Problems may arise pertaining to system architecture because not all requirements are gathered up front for the entire software life cycle.

3. Construct & Build
5. Spiral Model :



4. Evaluation and Risk Analysis

1. Identification

The spiral model combines the idea of iterative development with the systematic, controlled aspects of the waterfall model. It is a combination of iterative development process model and sequential linear i.e. the waterfall model with a very high emphasis on risk analysis.

Advantages:

- changing requirement can be accommodated.
- Allow extensive use of prototype.
- Users see the system early.

Dis-advantages

- Management is more complex.
- End of the project may not be known early.
- Process is complex.

Q.2 Give short note on :

1. Cocomo Model
2. LOC
3. Function Point.

Ans. 1. Cocomo Model : Cocomo (Cost Constructive Cost Model) is a regression model based on LOC i.e number of Lines of code. It is a procedural cost estimate model for software projects and often used as a process of reliably predicting the various parameters associated with making a project such as size, effort, cost, time and quality. It was proposed by Barry Boehm in 1970.

The key parameters which defines the quality of any software products, which are also an outcome of Cocomo are:

- Effort: Amount of labour that will be required to complete a task.

- Schedule : Simply means the amount of time required for the completion of the job, which is of course proportional to the effort but.

There are 3 categories:

- Organic Project : In this size is less than 50 KLOC. It is mainly used for small business system.
- Embedded Project : In this size is greater than 300 KLOC. e.g.: Software used in military.
- Semi-detached :- In this size is less than 300 KLOC. for e.g.: OS, compiler

There are 3 types of COCOMO Model

- Basic Model : It can be used for quick and slightly rough calculations of Software Costs.

$$E = A \times (\text{SIZE})^B$$

↳ Effort

- Intermediate Model : In this Software reliability and software complexity are also considered as a parameter along with size & effort.

$$\text{Total Effort} = EAP \times \epsilon;$$

- Advance Model : In this effort is calculated as a program size and a set of Post driver. for each

phase of software engineer. This model incorporates all characteristics of the intermediate model and provide procedure for adjusting the phase-wise distribution of the development schedule.

$$\text{Total Effort} = \sum_i \times \text{ACAP}$$

2. LOC :- Line of code , as the name suggest It counts the total number of lines of source code in project. Units are:

- KLOC : Thousand line of code
- NLOC : Non comment lines of code.
- KDSI : Thousands of delivered source instruction
The size is estimated by comparing it with the existing system of same kind.

Advantages:

- Universally accepted & is used in many models like COCOMO
- Estimation is set closer to developer's perspective.

Dis-advantages

- Different programming language contain diff. number of lines.
- No proper industry standard exist for the technique.

3. Function Point: In this method, the number and type of function supported by the software are utilized to find FPC. The steps in function point are:
- count the number of functions of each proposed type.
 - Compute the Unadjusted function Points.
 - Find Total Degree of influence.
 - compute value adjustment factor.
 - Find the function point count.

$$FP = \text{Count} * WF$$

→ weighting factor.

Assignment-3

Q.1 Case study and detailed description of Higher education online library system with diagram.

Ans. As the name suggests, online library system is related to the storage of information regarding the library. It is place from where the students and the faculties issue the books for their reference purposes. But the maintenance of keeping the records of issuing and borrowing is difficult. To make this task easier, online library system will be very useful. It helps in maintaining the information regarding the issuing and borrowing of books by the students and faculties.

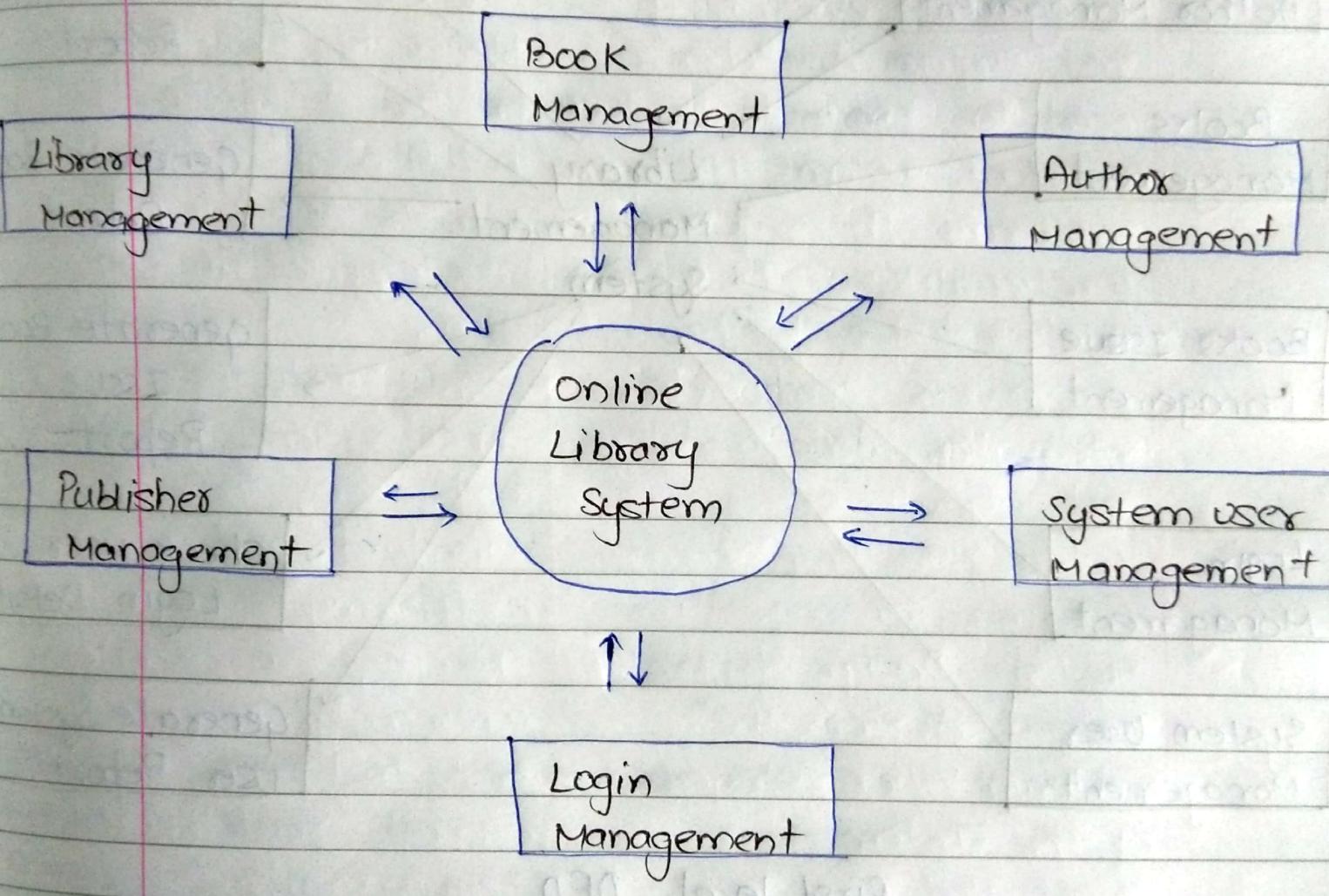
Purpose:-

- To provide a friendly environment to maintain the details of books and library members.
- Improved user service through greater access to accurate information.
- Due to computerized information it reduces the risk of paper work such as file lost, file damaged & time consuming. It can help user to manage the transaction or record more effectively and timesaving.

Functional Requirements.

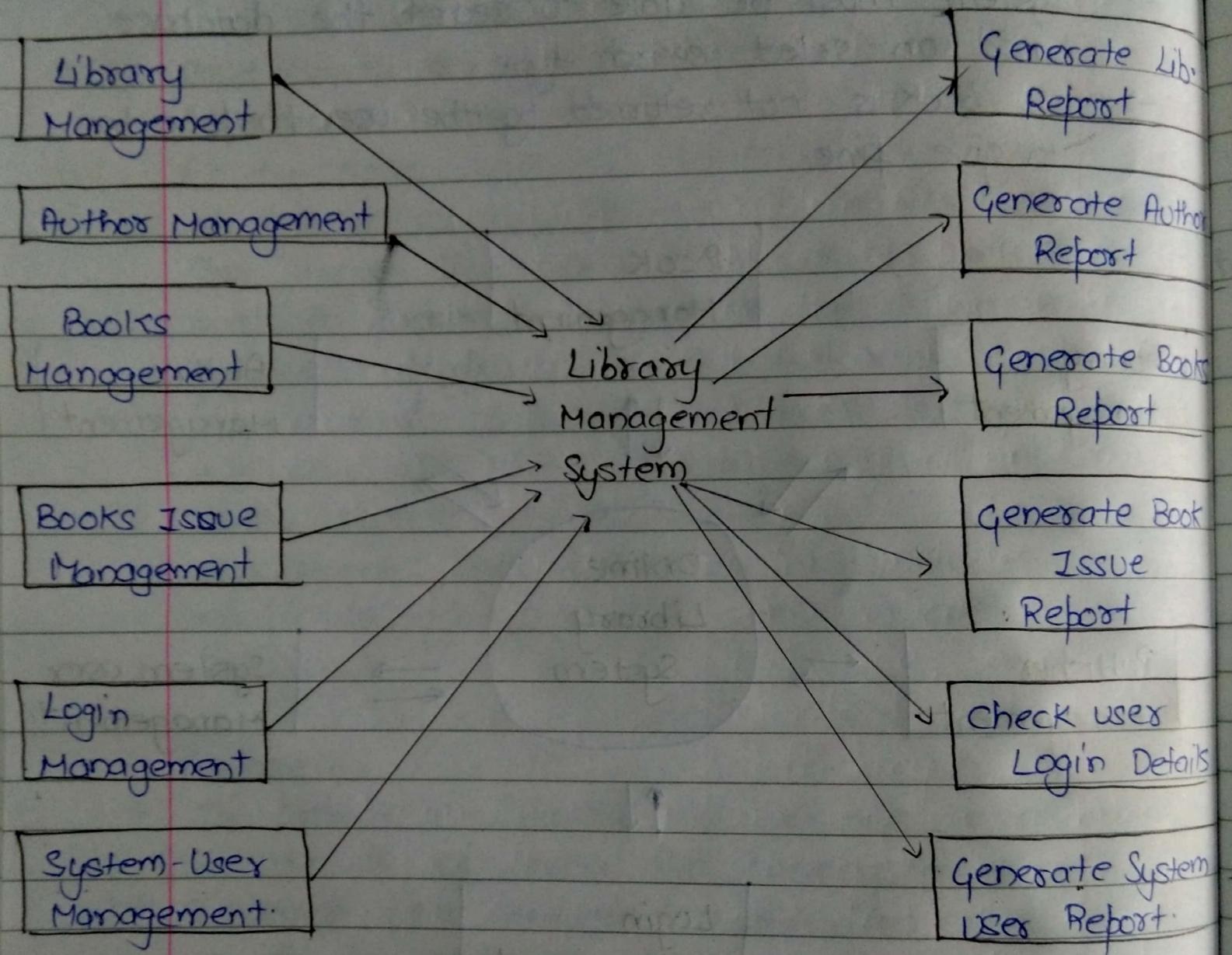
- The system must only allow user with valid id and password to enter the system.

- System must be able to search the database based on select search type.
- If book is not returned by the user, he/she has to give fine.



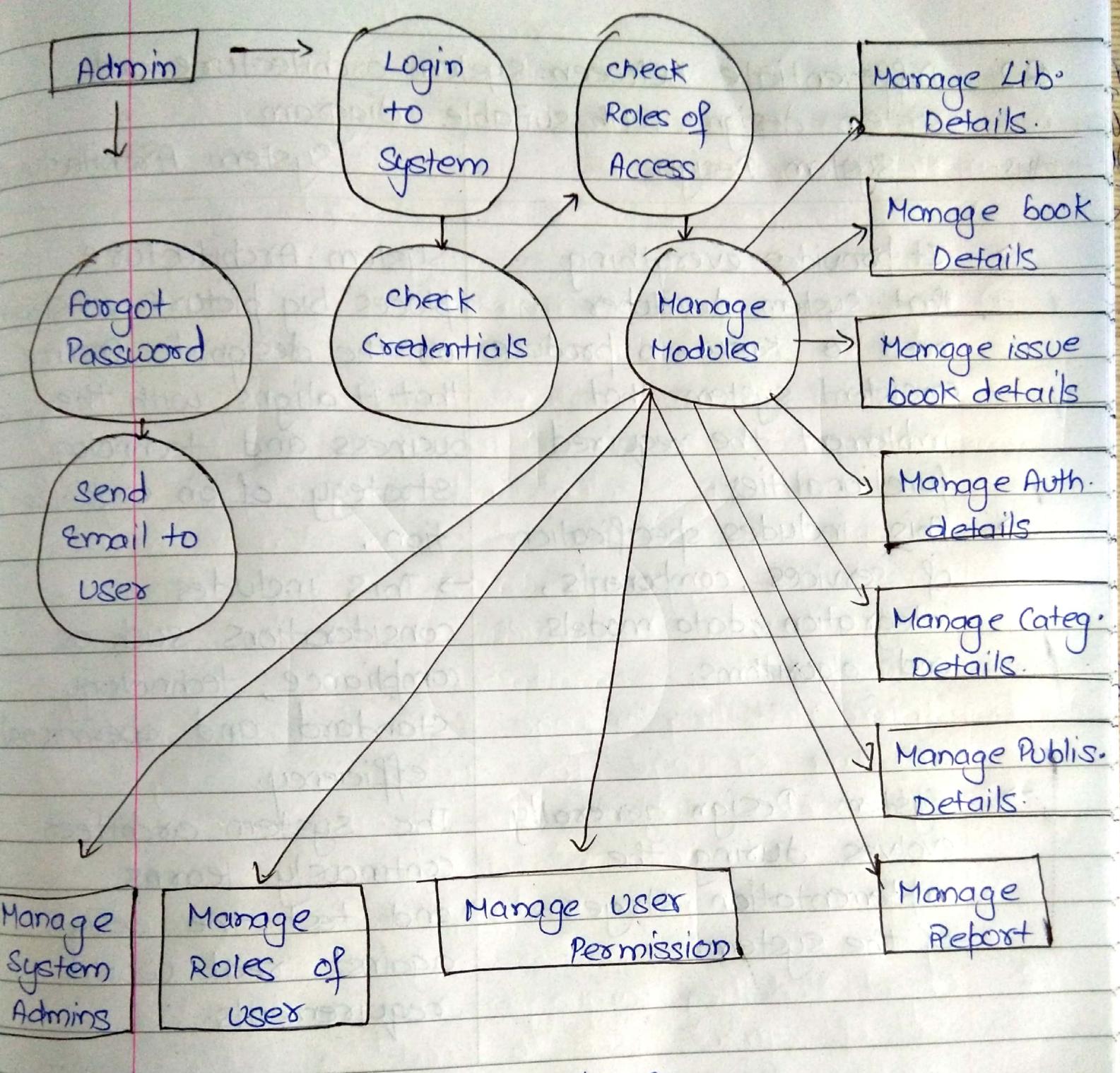
Zero-Level DFD

It's the basic overview of the whole online library management system or process being analyzed or modeled.



First-level DFD

It shows the the system is divided into sub-system each of which deals with one or more of the data flows to or from an external agent and which together provide all of the functionality of the online library System as whole.



Second-level DFD

It goes one step deeper into parts of level 1 of Library Management. It requires more functionalities of Library management to reach the necessary level of detail about the Library Management functioning.

Q2 Differentiate between system architecture and system design with suitable diagram.

Ans.

System Design

- It provides everything that system developer need to know to produce consistent system that implement the required functionalities.
- This includes specification of services, components, integration, data models and algorithms.
- System Design generally evolves during the implementation stages of the system.

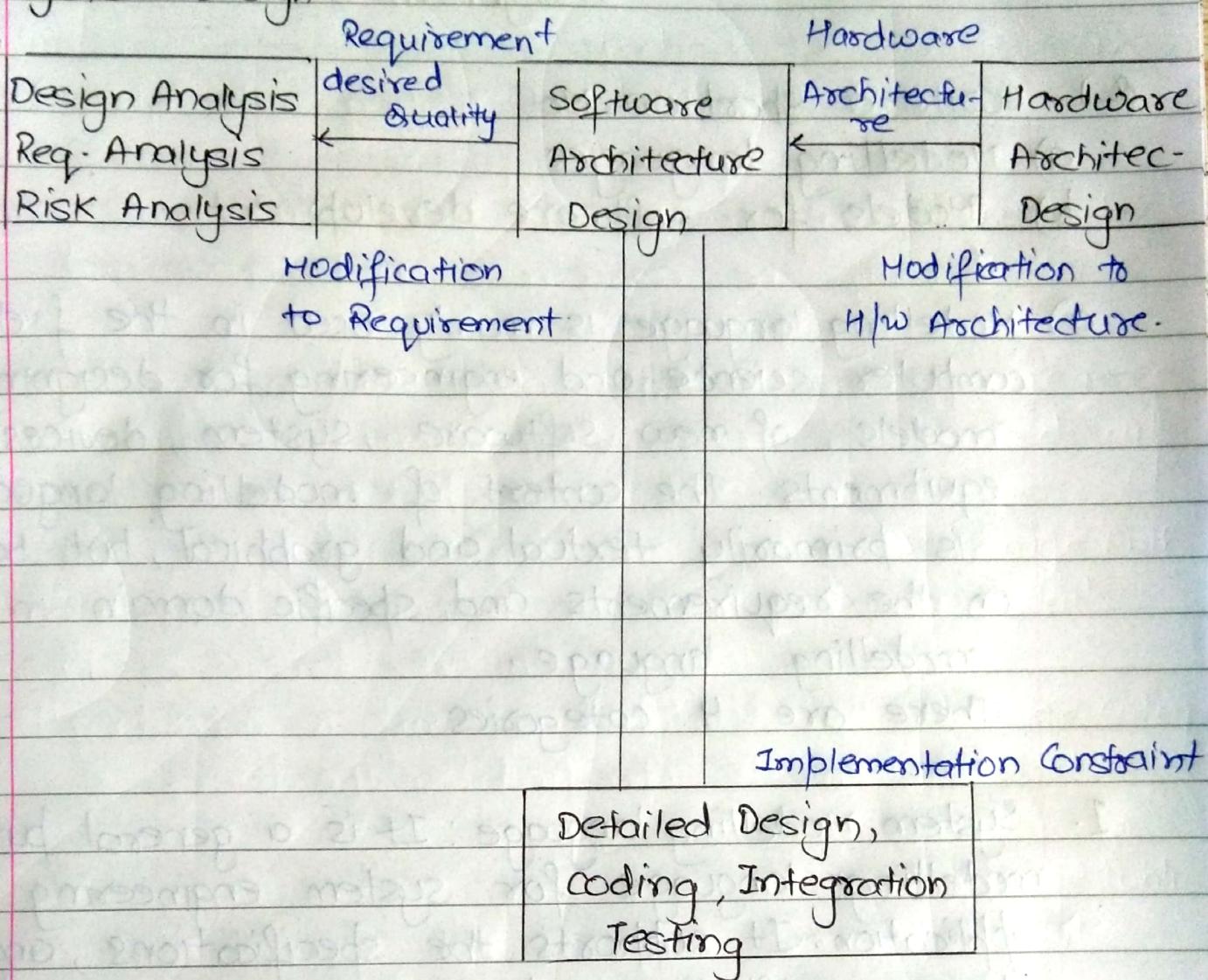
System Architecture

System Architecture places big picture constraint on the design to ensure that it aligns with the business and technology strategy of an organization.

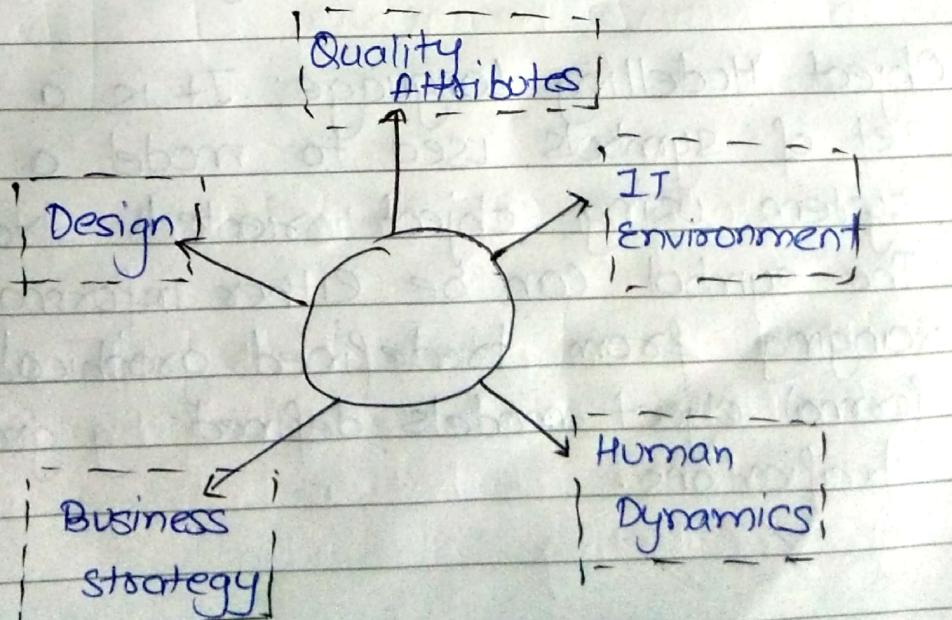
→ This includes considerations such as compliance, technology standard and operational efficiency.

The system architect continuously learns and test the design against real world requirements.

System Design :



System Architecture :



Assignment - 4

Q.1 Write a short on Mo

- a) Modelling languages
- b) Models for software development.

a) Modelling language is mainly used in the field of computer science and engineering for designing models of new software, system, devices and equipments. The context of modelling language is primarily textual and graphical, but based on the requirements and specific domain in use, modelling language.

There are 4 categories.

1. System Modelling language: It is a general purpose modelling language for system engineering application. It supports the specifications, analysis, design, verification and validation of a broad range of system and system of system.

2. Object Modelling Language: It is a standardized set of symbols used to model a software system using object oriented framework. The symbol can be either informal or formal ranging from predefined graphical templates to formal object models defined by grammar & specifications.

3. Virtual Reality Modelling language: It is a standard file format for representing 3-dimensional interactive vector graphics , design particularly with the world wide web in mind. It has been superseded by X3D.
4. Data Modelling language :- Data Modelling in software engineering is the process of creating a data model for an information system by applying certain formal techniques.

Assignment -5

Q.1 Detailed description of Testing with types and suitable diagram.

Ans. Software testing is a process, which is used to identify the correctness, completeness and quality of software.

Software testing is often used in association with the terms verification and validation.

Verification refers to checking or testing of items, including software for conformance and consistency with an associated specification.

Validation refers to the process of checking that the developed software is according the requirements specified by the user.

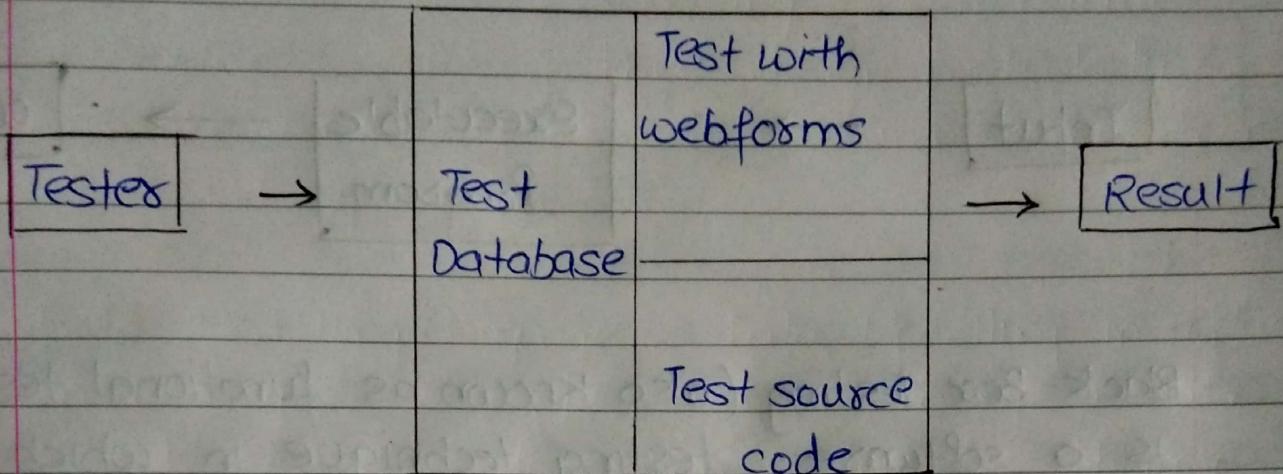
Objective:-

- To remove errors, which prevent software from producing outputs according to user.
- To remove error that leads to software failure.
- To improve the quality of software by removing maximum possible error from it.

Case-study on an Online book app (Wattpad)

Wattpad is an online book app that has 1000s of books available for free. But before releasing it, it has to undergo from various types of testing.

1. White Box Testing :-



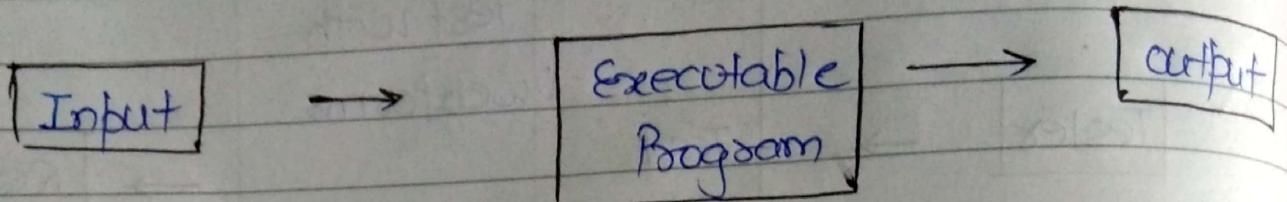
White Box Testing (also known as clear box testing, glass box testing, transparent box testing and structural testing) verifies the internal structure or workings of a program as opposed to the functionality exposed to the end user. In white box testing, an internal perspective of the system as well as programming skills are used to design test cases.

Working process :-

- Input - Requirement, functional specification, source code
- Processing - Performing risk analysis for guiding
- Proper test planning - Designing test cases.
- Output - Preparing final report of the entire testing process.

[P.T.O] =>

2. Black-Box Testing.

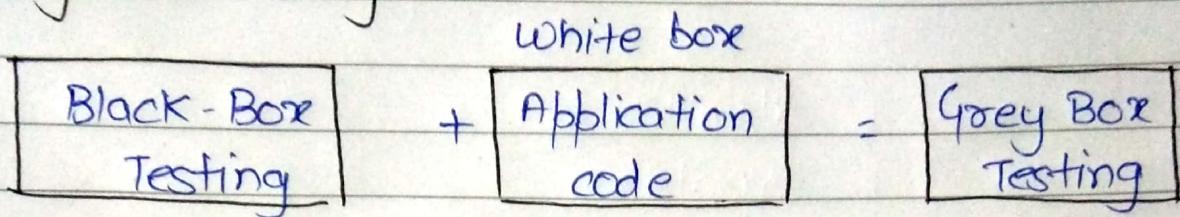


Black-Box Testing (also known as functional testing) is a software testing technique in which functionality of the software under Test (SUT) is tested without looking at the internal code structure, implementation details and knowledge of internal paths of the software. This type of testing is based entirely on the software requirement and specifications. In Black-Box testing we just focus on input and output of the software system without bothering about internal knowledge of the software program.

It can be done in following way:-

1. Syntax Driven Testing :- This type of testing is applied to the system that can be syntactically represented by some language.
2. Equivalence partitioning :- It is often seen that many types of inputs work similarly so instead of giving all of them separately we can group them together & test only one input of each group.

3. Grey-Box Testing



Grey-Box testing is a technique to test the application with having a limited knowledge of the internal workings of an application. The tester will often have access to both the source code and the executable binary. Grey-Box testing may also include reverse engineering (using dynamic code analysis) to determine, for instance boundary values or error messages.

Benefits:-

It provides combined benefits of both white box and black-box testing. It can handle design complex test scenario more intelligently.

Drawbacks:-

complete white-box testing cannot be done due to inaccessible source code / binaries.
It is difficult to associate defects when we perform grey-box testing for a distributed system.