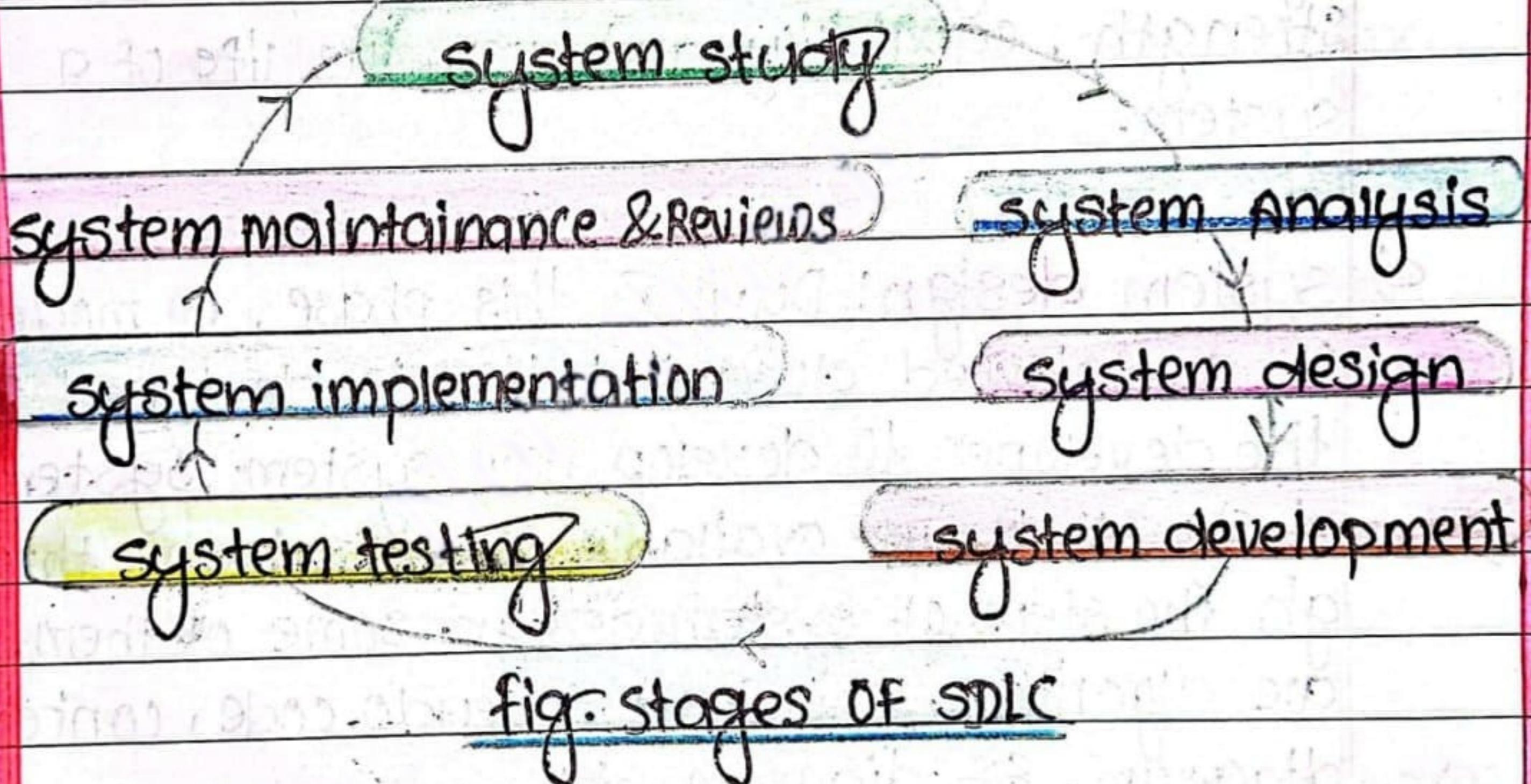


## Chapter 6

# Software process Model

- Q. List out different stages of system development life cycle (SDLC) and explain all stages.
- SDLC is a methodology used to develop, maintain and replace information systems.



The differentiate phases of stages of SDLC

are :

1. **System study**: During this phase, the system analyst perform a lot of feasibility study which includes technical feasibility, economical feasibility, operational feasibility, social feasibility, legal feasibility, time feasibility and behaviour feasibility. After all these study

the system analyst decides whether to undertake the software or not.

b) System Analysis: During system analysis, data are collected by using tools such as interview, on-site observation, questionnaire, sampling, research, and documentation. Finally, System analysis determines the shapes, form, strength, capability and even the life of a system.

c) System design: During this phase, a model or a detailed documentation is designed for the developer to develop new system. System design tools are available to help teams through the steps of system design. Some of them are algorithm, flowchart, pseudo code, context diagram, ER diagram, etc.

d) System development: During the development phase, the developers play a key role, creating or customizing the system. The programmer is also responsible for coding and documenting the program, providing an explanation of how and why a certain procedure is coded in specific ways.

e. system testing: system testing is done to check whether the newly developed system is error free or not. Only this phase can verify whether the system is ready for the implementation or not. There are four types of system testing. They are unit testing, integration testing, system testing and user acceptance testing.

f. system implementation: The way of carrying out a develop system into working condition is called system implementation. Implementation activity include initial user support such as the documentation, training program and on-going user support assistance. The processing of moving from the old system to the new system is called system conversion.

g. system maintenance: correcting and up grading process of the system is called system maintenance. The maintenance can be classified as:-

- correcting maintenance :- It means repairing, processing and performance failure or making alterations.
- Adaptive maintenance :- corrects a problems or upgrade the performance of the system.
- perfective maintenance : modifying problems to

make it even more perfect.

b) **System review**:- The last phase of SDLC is the system review. The review is carried out from the client department, external audit and system analyst. The basic purpose of this stage is to see whether the new system has met the objectives or not.

VVI

- \* Define the features of Quality software.
- The software can be consider as quality software. If it contains the following features:
- The software should be correct.
  - It shouldn't contain error.
  - It should be easy to understand.
  - It should be easy to maintain & update.
  - It should be reliable.
  - It should be able to operate with a wide range of possibilities.
  - It should have an easier user interface.
  - It should be portable & flexible.
  - It should be efficient in terms of processing and memory utilization.

Q. Explain about the different testing techniques during the system development.

→ The different techniques during the system development are as follows:-

#### a. Unit testing

It is the process of testing the individual component of the system after the component is developed. This testing process is performed by the developer.

#### b. Integration testing

Integration is the process of combining all the individual components given together to form a complete system. This testing is also performed by the developer.

#### c. System test

The system can be tested in two ways:-

- Black box testing: It is the testing where internal code of the program is tested.
- white box testing: It is the testing where structure of the program is tested.

#### d. User acceptance testing

It is the process of testing the complete system by the user. The user tests whether

the newly developed system meets the requirement or not.

**Q: Explain any two designing tool in detail.**

→ The two designing tools used in system design phase are :-

1. Algorithm
2. Flowchart

### 1. Algorithm

An Algorithm is a stepwise logical instruction written in human understandable language for solving a problem.

#### Characteristics of algorithm :-

- a. It should be clear & accurate
- b. It should have fixed number of steps.

**for eg:** Algorithm to find the area of rectangle.

Step 1 : Start

Step 2 : Accept l, b

Step 3 : calculate, Area = l × b

Step 4 : Display Area

Step 5 : Stop

## 2. Flowchart

The pictorial representation of algorithm is called flowchart.

Some symbols used in flowchart:

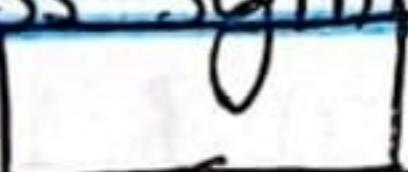
a. Input / Output symbol



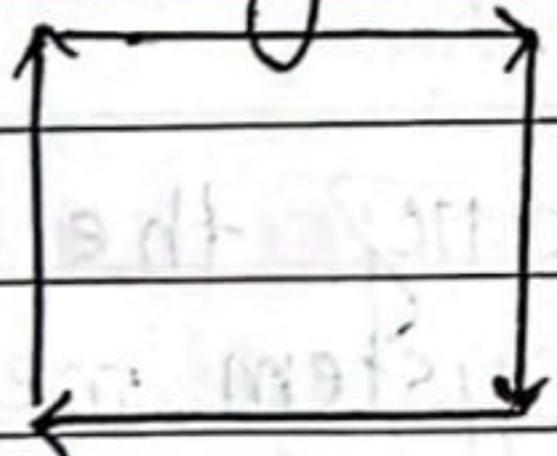
b. Start / End symbol



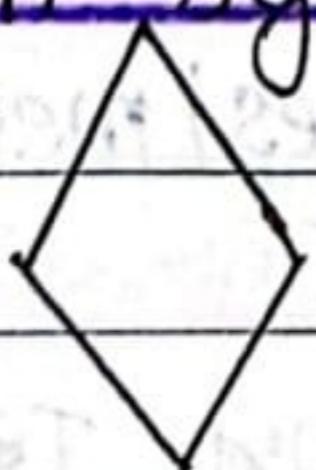
c. process symbol



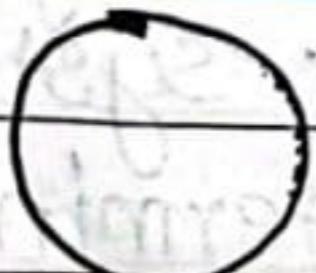
d. line symbol



e. Decision symbol

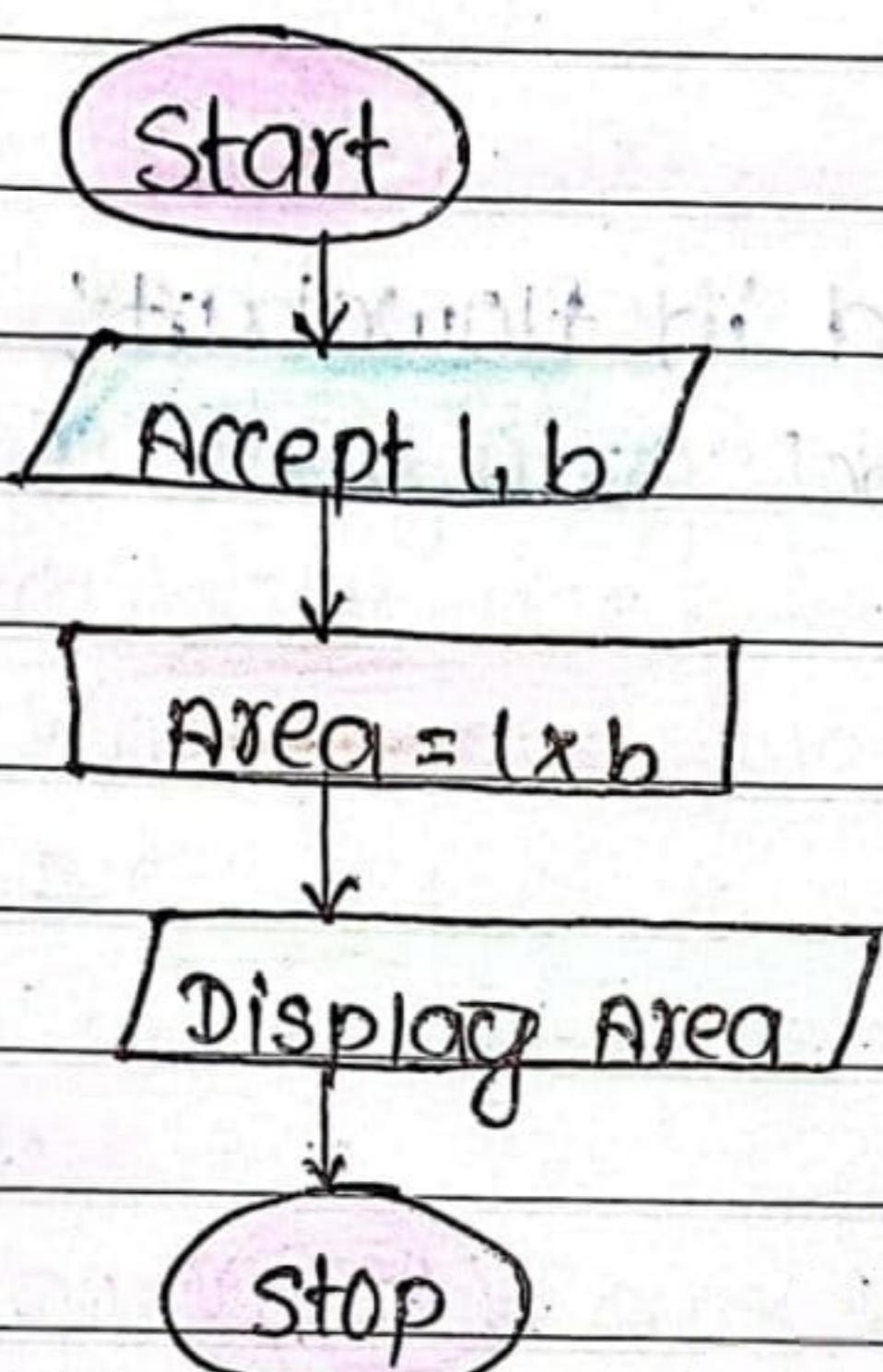


f. connector



for egr!

Flowchart to find Area of rectangle:



\* What is feasibility study? Explain different levels of feasibility study.

→ Feasibility study is basically the test that determines whether the system meets the goal of the clients or not.

Different levels / types / methods of feasibility study.

a. Technical feasibility: During technical feasibility study, the system analyst identifies the existing computer systems (hardware and software) and determines

whether these technical resources are sufficient for the proposed system or not.

b. **Economical Feasibility**: Economical feasibility is the most important study that determines the cost and benefits of the proposed system and compares with the budget.

c. **Operational feasibility**: During operational feasibility study, it is determined whether the system will operate in the way that user wants.

d. **Social Feasibility**: Social feasibility is a determination of whether a proposed system will be acceptable to the people or not.

e. **Legal and contractual feasibility**: Legal feasibility is a determination of whether a proposed system infringes on known Acts, statutes, as well as any pending legislation.

f. **Time feasibility**: Time feasibility is a determination of whether a proposed system can be implemented within a deadline or not.

g. **Behavioral Feasibility**: The behaviour feasib-

ility includes a study of organizational behaviour which also includes the mental status of staff.

\* Describe the major role and responsibility of system analyst in SDLC phase:

OR, Who is system analyst? Explain the role of system analyst.

→ system analyst is a computer specialist person who is involved in analyzing, designing, implementing and evaluating computer-based information systems to support the decision-making and operations of an organization.

### Role & Responsibilities / Duties of the system Analyst :

1. Defining requirement: It involves understanding user's requirements by interviewing users, finding out what information is being used in a current system, how it is used & what is their expectation towards the proposed system.

2. Prioritizing requirement: There is a need

to set priorities among the requirements of various users. This responsibility of the system analyst requires good interpersonal relations.

### 3. Gathering data, facts and Opinions of Users:

Having determined the information needs & their priority, the system analyst must develops the system with the active and willing cooperation of all the users.

### 4. Analysis & evaluation:

The system analyst analyses the working current information system in the organization. He then shifts through the facts and opinions gathered by him and find the best characteristics of a new or modified system.

### 5. Solving problems:

A system analyst must suggest study the problem in-depth and suggest an alternate solution to the management.

### 6. Drawing up specification:

A key job of a system analyst is to obtain the functional specification of the system to be designed in

a form, which can understood by users. The specification must be non-technical so that the users and managers can understand it. The specification must be precise and detail so that it can be used by the system implementer.

7. **Designing system:** Once the specification is accepted, the system analyst designs the system. The design must be understandable to the system implementers and it must be modular to accommodate changes easily.
8. **Evaluating system:** A system analyst must evaluate system after it has its use for a reasonable period of time. The time, at which evaluation is to be done, how it is to be done comments of the users are to be gathered and used must be decided by the system analyst.

\* Who is system analyst? List out characteristics / qualities / attributes of system analyst.

→ The characteristics or qualities or attributes of system analyst are as follows:-

1. knowledge of organization: A system analyst must understand the management structure and the relationship among the departments in the organization, its day-to-day operations for which the system is being developed.

2. Technical Knowledge: A system analyst should be well-trained in relevant areas of computer science. She must know enough programming ideas and techniques to be able to design a system, which can be implemented.

3. Interpersonal communication skill: System Analysts must be able to talk intelligently with high-level management technical staff, and non-technical users, programmers, and operators. She is required to influence people to change their minds and attitudes and motivate them to work in a group.

4. Character and ethics: System analyst should understand the difference between "right"

and "wrong" and acts accordingly. In short, a successful system analyst must be professional. He must be resourceful, inventive and creative.

**5. problem-solving skill:** systems are developed to solve problems. system analyst must have wide experience in solving problems. The objective can be achieved in the following ways:

1. Defining the problem
2. Analyzing the problem
3. Considering many alternatives evaluating many alternatives.
4. choosing the best alternatives.

### Difference between system Analyst & system engineer

#### System Analyst

- A system analyst explores and documents requirements

#### System engineer

- A software engineer designs software to meet requirements.

- A system analyst is more responsible for documenting the program. software engineer's responsibilities lie with developing and program.

ss of a project for quality control.

ramming the software.) In some cases, the terms software engineer & programmer or developer are interchangeable.

- system analysts need to work with a big amount of data. software engineers work on building new software (desktop, web or cloud, mobile, etc.)
- A systems analyst does more problem research their product to show the and plans solutions.) and software and system recommendations to solve these problems. software engineer develops result.
- A person working as a system analyst is more experienced & hence more paid than a software engineer. A person working as a software engineer is less experienced and hence less paid than system analyst.

## \* Explain requirement collection methods.

→ During system analysis, data are collected from the various available sources by using different methods, some of them are:

1. Interview: Interviews of stakeholders and users are critical to developing great software without understanding the goals and expectations of the users and stakeholders, we are very unlikely to satisfy them.
2. Survey / Questionnaire: Individual interviews present several challenges. They can be tricky to schedule and time consuming for the interviewers. So, we can collect information from many people by doing a survey or questionnaire.
3. On-site Observation: By observing users and the work process, an analyst can identify a process flow, steps, pain points, and opportunities for improvement.
4. Brainstorming: Brainstorming is used in requirement gathering to get as many ideas as possible from a group of people. It generally used to identify possible solutions to

problems, and clarify details of opportunities.

5. Document Analysis: reviewing the documentation of an existing system can help when creating requirement documents, as well as driving gap analysis for scoping migration project.

6. Focus group: A focus group is a gathering of people who are representative of the users or customers of a product to get feedback.

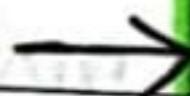
7. prototyping: prototyping is a relatively modern technique for gathering requirements. In this approach, we gather preliminary requirements that we use to build an initial version of the solution a prototype. We show this to the client, who then gives you additional requirements.

8. Requirement workshops: workshops can be very effective for gathering requirements structured than a brainstorming session; involved parties collaborate to document requirements.

# Software Development model

1. The waterfall model
2. prototype model
3. Spiral model
4. Agile model

Q. Describe waterfall model with its advantages and disadvantages.



Initial Investigation

Requirements generation

Design

Development

Test & Deploy

post deployment Review

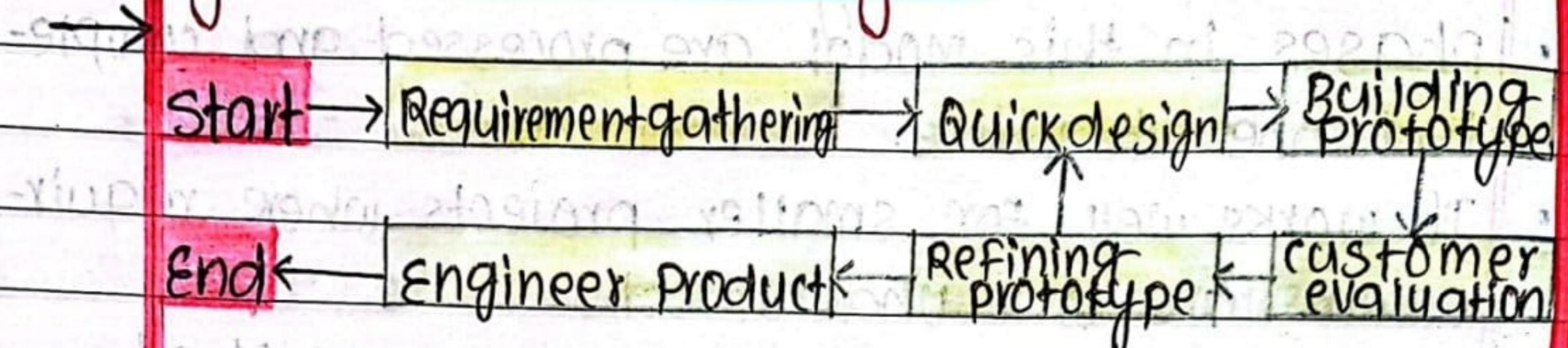
## Advantages of the waterfall Model

- It is simple and easy to understand.
- phases in this model are processed and completed one at a time.
- It works well for smaller projects where requirements are well understood.
- Each stage in this model is clearly defined.
- After the completion of each phase, well documentation is done.

## Disadvantages of the waterfall model

- It is not a good model for complex and object-oriented projects.
- There is a high amount of risk and uncertainty in this model.
- This model cannot accept the changes in requirements during development.
- It focuses very little on the end user or client's involvement with a project.

Q. Describe prototype model with its advantages and disadvantages.



### Advantages of prototype

- Users are actively involved in development. Therefore, errors can be detected in the initial stage of the software development process.
- The customers get to see the partial product early in the life cycle. This ensures a greater level of customer satisfaction and comfort.
- New requirements can be easily accommodated as there is scope for refinement.
- Customer satisfaction exists because the customer can feel the product at a very early stage.
- Quicker user feedback helps you achieve better software development solutions.

### Disadvantages of prototype

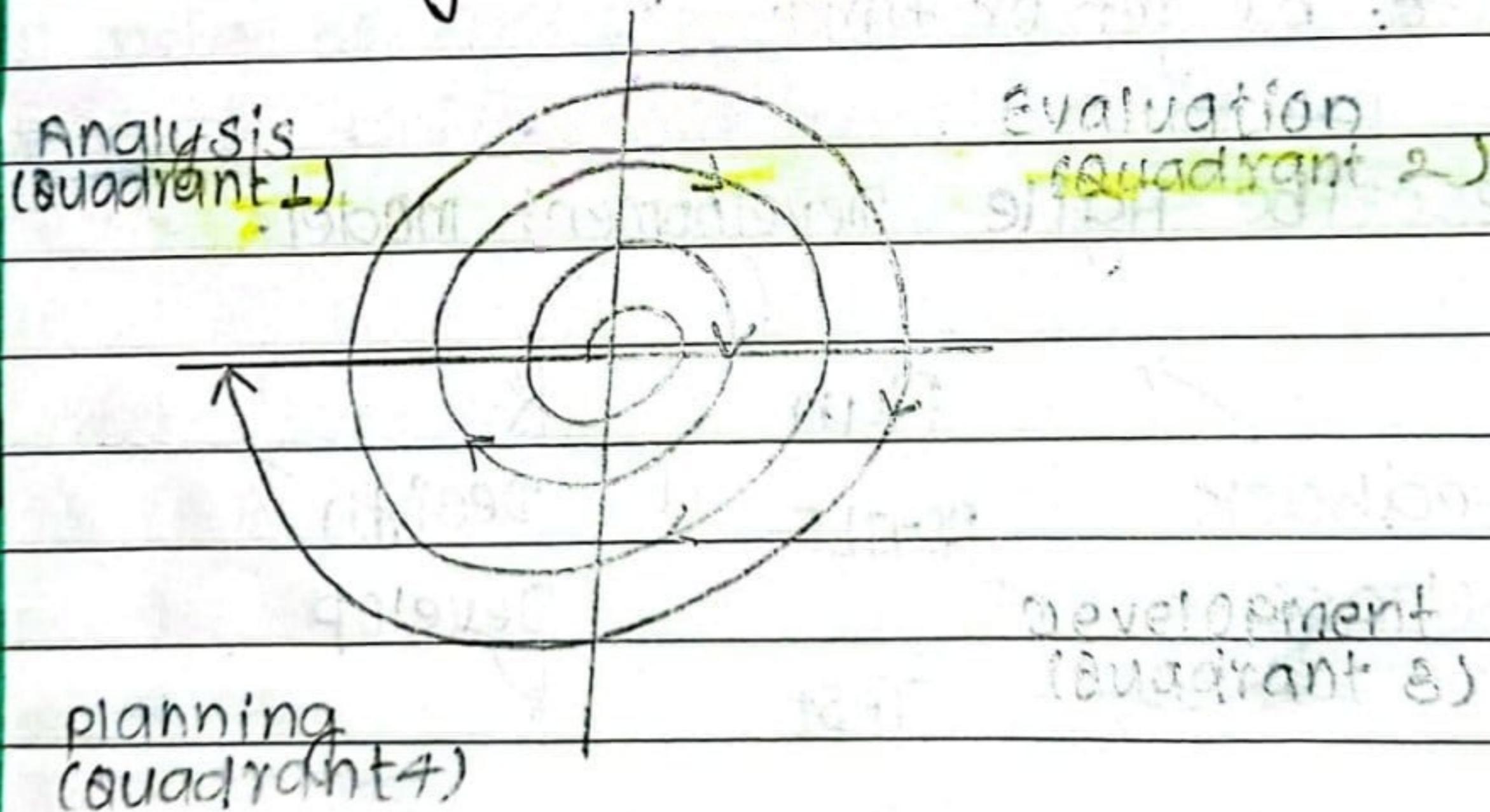
- Prototyping model is a slow and time-consuming process.
- The cost of developing a prototype is a total

waste as the prototype is ultimately thrown away.

- It is very difficult for software developers to accommodate all the changes demanded by clients.
- After seeing an early prototype model, the customer may think that the actual product will be delivered to him/her soon.

~~nomadic~~

\* Explain spiral model with its advantages and disadvantages?



### Advantages

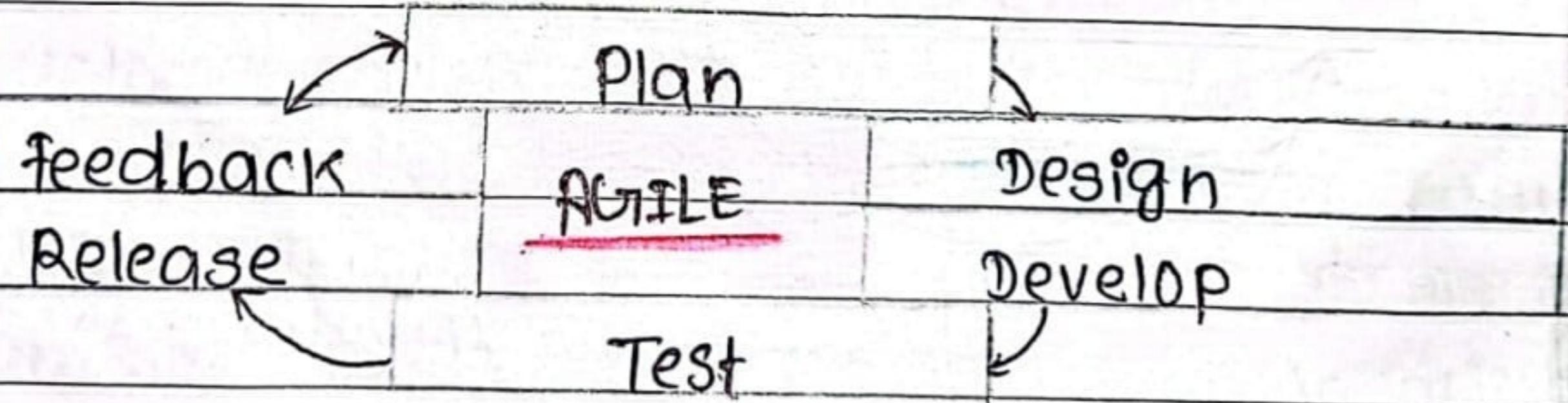
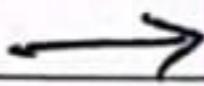
- It estimates (i.e. budget, schedule) get more realistic as work progresses, because important issues are discovered earlier.

- It is more able to cope with the changes that software development generally entails or involves.
- Software engineers (who can get restless with protracted design processes) can get their hands in and start working on a system or project earlier.

### Disadvantages

- It is highly customized limiting re-usability.
- It is applied differently for each application.
- It has a risk of not being completed within allocated budget or time.

\* Describe Agile Development model.



### Advantages

- Working software is delivered frequently.
- Customers, developers and testers have to interact with each other frequently.
- It is efficient model that fulfills the business

requirements.

- It reduces total development time.

### Disadvantages

- It is not suitable for handling complex dependencies.
- There is a lack of emphasis on the necessary design and documentation.
- The entire project is totally dependent on customer interaction, so if the customer is not clear, the project can easily get taken off track.
- Due to the lack of proper documentation, the transfer of technology to new team members may be quite challenging.