Chapter 1
System Development Concept

System

System is a set of interacting or interdependent components forming an integrated whole. A system can be described as a set of objects joined together for a common objective.

Components of system:

1. **Hardware**: hardware refers to the physical layer of the information system. It includes computers, networks, scanners, digital capture devices, and other technology-based infrastructure.

2. **Software**: software refers to the logical layer of information system that makes hardware function properly. It includes system software (OS, device driver, etc.) and application software.

3. **Data**: data is raw facts, figures, and measurements, etc. which are used as input for the system. It may be alphabets, digits, figures and other symbols.

4. **Processes**: processes, or procedures, describe the tasks that users, managers, and IT staff members perform.

5. **People**: people refer to users who interact with information system. The users within the organization include mangers, technicians, sales representatives, corporate officers, etc. are called internal users and the users outside the organization include customers, suppliers, etc. are called external users.

Information System

An information system is a computer based system which is an integrated set of different components for collection, process, storage and transmission of data. Simply, it us a system which processes supplied/collected data and generates information that can be used for decision making at different levels for the betterment of an organization. The types of information system are:

1. **Transaction processing system**: it processes data resulting from business transactions, updates operational database such as sales and inventory processing and accounting systems.

2. **Management information system:** it is the integrated modern approach of management, information and computerized system. It provides information to support the operations, management, decision making functions of an organization.

3. **Decision support system**: It is the information system at the organization's senior level management that combines data and sophisticated analytical models or data analysis tools to support semi-structures and unstructured decision makings.

4. **Executive support system:** it is also known as executive information system. It operates on the executive level of management. It provides critical information from many sources customized to the information needs of executives.

System Analyst

System analyst is a people who is involved in analyzing, designing, implementing and evaluating computer based information to support the decision making and operation of an organization.

The roles of system analyst area as follows:

1. Change event
2. Investigator and event
3. Architect
4. Psychologist
5. Motivator
6. Defining and prioritizing the requirements
7. Analysis and evaluation
8. Designing system

The characteristics of system analyst are as follows:

1. Knowledge of organizations
2. Knowledge of computer system
3. Good inter-personal relation
4. Communication skills
5. Analytical skill
6. Breadth of knowledge

SDLC
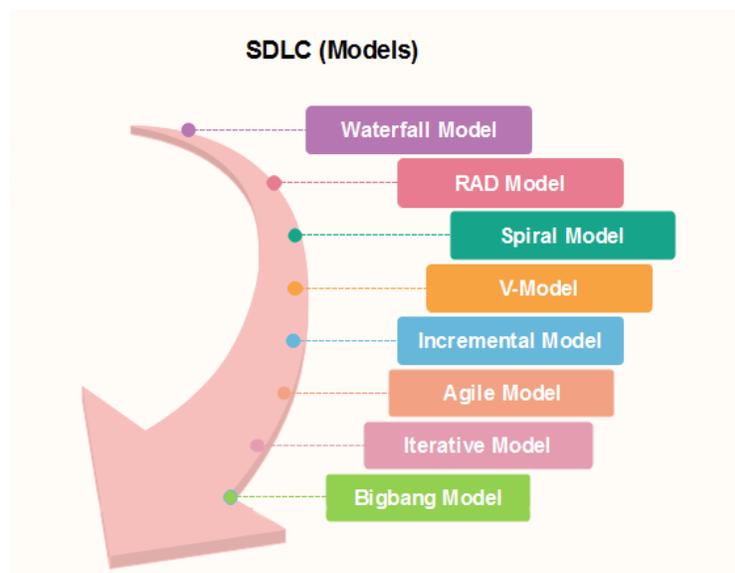
Software development life cycle (**SDLC**) is a series of phases that provide a common understanding of the software building process. How the software will be realized and

developed from the business understanding and requirements elicitation phase to convert these business ideas and requirements into functions and features until its usage and operation to achieve the business needs. The good software engineer should have enough knowledge on how to choose the SDLC model based on the project context and the business requirements.

SDLC consists of a set of development activities that have a prescribed order. It is the development of software in chronological order.

The different phases of SDLC are as follows:

1. **System study**: A system is intended to meet the needs of an organization. Thus the first step in the design is to specify these needs or requirements. The top manager of the organization takes the basic decision to use a computer based (information) system for managing the organization.

2. **System analysis**: system analysis is the dissection of a system into its component pieces to study how those component pieces interact and work. System analysis is a term that collectively describes the early phases of development. It is defined as those phases and activities that focus on the business problem, independent of technology.

3. **Feasibility study**: feasibility study is the most important activity in the system analysis phase. It analyses the proposed system from different aspects so that it makes us clear that how practical or beneficial the system will be to the organization. So it tells us whether the system is feasible to design nor not.

4. **System design**: the next step is to develop the logical design of the system. During this phase, the logic of the system, namely, the information requirement of users, and use this to find the necessary database.

5. **System development**: after designing a logical diagram of a system then next step is to convert into program. This process is called system development. Flowchart, algorithm, Pseudo code, etc. are the outlines the procedures for taking the input data and processing it into usable output.

6. **System testing**: it is an investigation conducted to provide stakeholders with information about the quality of the product or service under test. System testing also provides an objective, independent view of the software to allow the business to appreciate and understand the risks of software implementation.

7. **Implementation**: implementation involves testing the installed system, converting from the old system to the new one and training the users. This phase consists of implementation of the system into a production environment, and resolution of the problem identified in testing phase.

8. **Maintenance and review**: it begins after the system is implemented. Like any system, there is an ageing process that requires periodic maintenance of hardware and software. The content of the review will include objectives met, cost, performance, standards and recommendation.

1. Waterfall model

It is a oldest type of model for software engineering. **It** is a sequential model that divides software development into pre-defined phases. Each phase must be completed before the next phase can begin with no overlap between the phases. Each phase is designed for performing specific activity during the SDLC phase. It was introduced in 1970 by Winston Royce.

The fundamental processes of waterfall model are as follows:
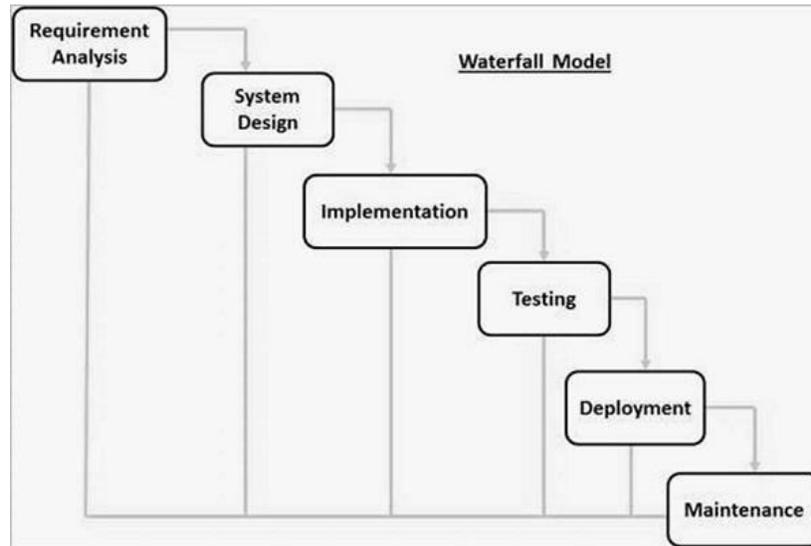
1. **Requirements analysis and definition**: it is the first stage of waterfall model. In this stage, the developer should identify the actual requirements of the given problem.

2. **System design**: in this stage the systems design process partition the requirements to either hardware or software systems.

3. **Implementation and unit system:** During this stage, the system design is realized and tested as a complete system to ensure that the software requirements have been met.

4. **Operation and maintenance**: in this stage, the system is installed to the desire location. The maintenance involves correcting errors which were not discovered in earlier stages of the life cycle, improving the implementation of system units and enhancing the system's service as new requirements are discovered.
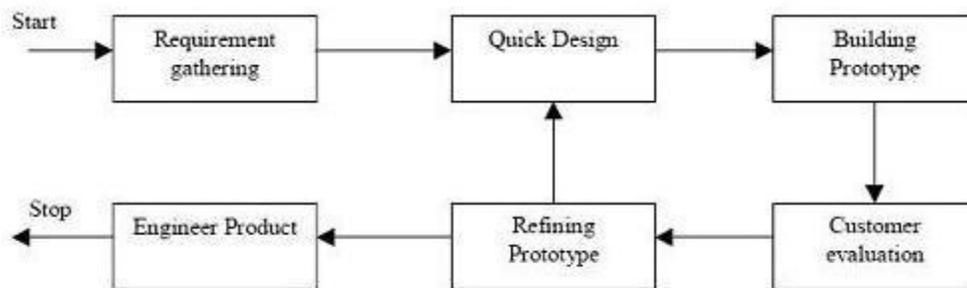
Advantages of the Waterfall Model

- The advantage of waterfall development is that it allows for departmentalization and control. A schedule can be set with deadlines for each stage of development and a product can proceed through the development process model phases one by one.
- The waterfall model progresses through easily understandable and explainable phases and thus it is easy to use.
- It is easy to manage due to the rigidity of the model – each phase has specific deliverables and a review process.
- In this model, phases are processed and completed one at a time and they do not overlap. The waterfall model works well for smaller projects where requirements are very well understood.

Disadvantages of Waterfall Model

- It is difficult to estimate time and cost for each phase of the development process in waterfall model.
- Once an application is in the testing stage, it is very difficult to go back and change something that was not well-thought-out in the concept stage.
- Not a good model for complex and object-oriented projects.
- Not suitable for the projects where requirements are at a moderate to high risk of changing.

2. Prototyping model

It is the iterative process of system development which is more appropriate for developing new system where there is no clear idea of requirements, inputs and outputs.

Prototyping Model

1. **Identify the user needs**: the system analyst interviews the user to obtain an idea of what is required from the system.

2. **Develop a prototype**: the system analyst, working uses one or more prototyping tools to develop a prototype.

3. **Determine if prototype is acceptable**: the analyst educates the user in prototype use and provides an opportunity from becoming familiar with the system.

4. **Use the prototype**: the prototype becomes the operational system.

Advantages of **Prototyping**

1. Communications between the systems analyst and user are improved.

2. The expert system can do a better job of determining the user's needs.

3. The user plays a more active role in system development.

4. The expert and the user spend less time and effort in developing the system.

5. Implementation is much easier because the user knows what to expect.

6. A system is developed through operational prototyping is capable of easy adaptation.

Disadvantages of **Prototyping**

1. Because of the iterative nature of prototyping, there is no definite deliverable or competition deadline.
2. Code inefficiencies may be a drawback in terms of functionality.
3. The computer-human interface provided by certain prototyping tools may not reflect good design techniques.

3. Spiral system

In this model, process is represented as a spiral rather than as a sequence of activities with backtracking.

It is combining elements of both design and prototyping-in-stages, in an effort to combine advantages of top-down and bottom-up concepts. This model of development combines the features of the prototyping model and the waterfall model. The spiral model is favored for large, expensive, and complicated projects. This model uses many of the same phases as the waterfall model, in essentially the same order, separated by planning, risk assessment, and the building of prototypes and simulations.
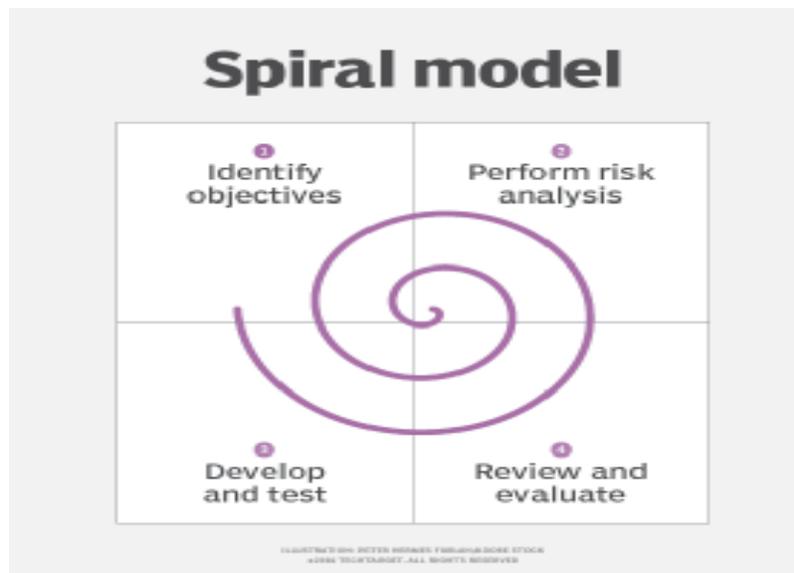
1. **Planning**: the project is reviewed and a decision made whether to continue with a further loop of the spiral. If it is decided to continue, plans are drawn up for the next phase of the project.
2. **Risk analysis**: for each of the identified project risks, a detailed analysis is carried out. Steps are taken to reduce the risk. For example, if there is a risk that the requirements are inappropriate, a prototype system may be developed.
3. **Software development**: after risk evaluation, a development model for the system is chosen.
4. **User evaluation**: specific objectives for the phase of the project are defined by the evaluation of users. Constraints on the process and the product are identified. And a detailed management plan is drawn up.

Advantages	Disadvantages
• Estimates (i.e. budget, schedule, etc.) become more realistic as work progressed because important issues are discovered earlier. • Early involvement of developers. • Manages risks and develops the system into phases.	• High cost and time to reach the final product. • Needs special skills to evaluate the risks and assumptions. • Highly customized limiting re-usability

Feasibility study

Feasibility study is the most important activity in the system analysis phase. It analyses the proposed system from different aspects so that it makes us clear that how practical or beneficial the system will be to the organization. So it tells us whether the system is feasible to design nor not. Thus it is necessary before system design.

The different levels of feasibility study are as:

1. **Economical feasibility**: it concerns with cost effectiveness of the system. The main objective of economical feasibility is to calculate approximate cost-both the development cost and the operational cost and the benefits from the system.

2. **Technical feasibility**: it concerns with the availability of the hardware, software and the support equipments for the complete development of the system.

3. **Operational feasibility**: it concerns with smooth operation of the system. It is all about the problems that may occur during operation of the system after its development.

4. **Behavior feasibility**: it concerns with behavior of the users and the society towards the new system. Generally, most of the traditional employees are not easily ready to upgrade them with the new system.

5. **Schedule feasibility:** it is the process of splitting project into tasks and estimate time and resources required to complete each task. It determines the deadline to complete a system and schedule the task accordingly.

6. **Legal feasibility**: it concerns with legal issue of the system. If the system is illegal then the system designing is meaningless. Everything is measured whether it is legal or illegal. It considers copyright law, foreign law, foreign trade, tax, etc.

System testing

It is an investigation conducted to provide stakeholders with information about the quality of the product or service under test. System testing also provides an objective, independent view of the software to allow the business to appreciate and understand the risks of software implementation.

1. **White box testing**: white box testing of software is predicted on close examination of procedural details. Logical path through the software and collaborations between

components are tested by providing test case that exercises specific sets of conditions or loops. It is used when the tester has access it the internal data structures and algorithms including the code that implement these.

2. **Black box testing**: black box testing treats the software as a black box â€"without any knowledge of internal implementation. Black box testing methods include: equivalence partitioning, boundary value analysis, specification based testing, etc. it is also called functional testing because it tests whether a system is functioning or not.

Flowchart

System flowchart describes the internal architecture of a system that describes how data are moved inside the internal components of a system. Program flowchart describes to solve the application types of real world problem. The different symbols used in system flowchart are defined below:

Symbol	Name	Function
	Start/end	An oval represents a start or end point.
	Arrows	A line is a connector that shows relationships between the representative shapes.
	Input/Output	A parallelogram represents input or ouptut.
	Process	A rectangle represents a process.
	Decision	A diamond indicates a decision.

DFD

DFD is the logical diagram to describe the flow of data inside the components of system. It is easier to understand or grasp when being explained and most important to all, it is much more precise and less ambiguous than a narrative one. The main components are: process, data store, data flow, external entities.

ER diagram

The E-R diagram is an overall logical structure of a database that can be expressed graphically. It was developed to facilitated database design and the simplicity and pictorial clarity of this diagramming technique have done great help in the designing part of database. The main components are attributes, entities and relationship.

Case diagram

computer aided software engineering tool is automatic computer based program that helps for software engineering and SDLC process. It us very fast and effective tools for the development of big scale software. It helps in analysis, design, implementation, testing and maintenance.

UML

Unified Modelling Language is a standardized general purpose modelling language in the field of object-oriented software engineering. The standard is managed, and was created by, the object management group. UML includes a set of graphic notation techniques to create visual models of object-oriented software.

System flowchart

System flowchart describes the internal architecture of a system that describes how data are moved inside the internal components of a system. Program flowchart describes to solve the application types of real world problem.