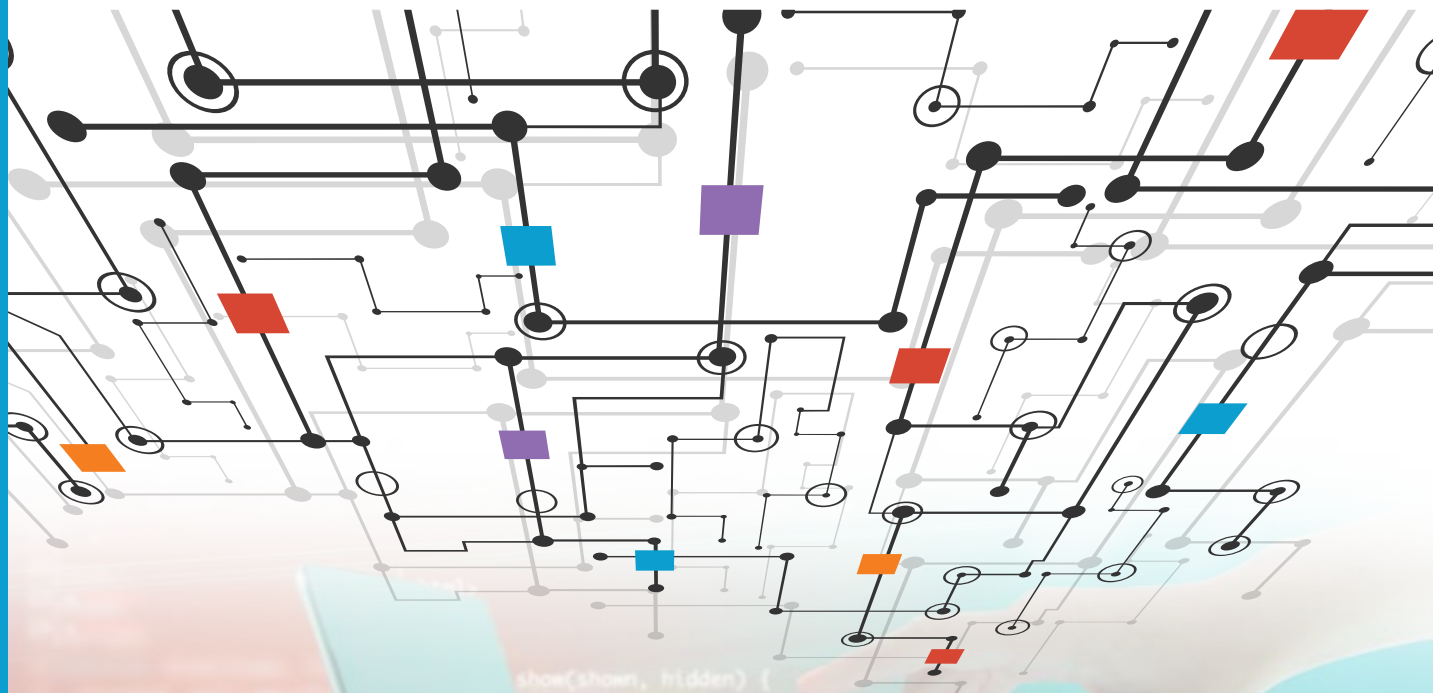
NIBM WORLD WIDE



ebook

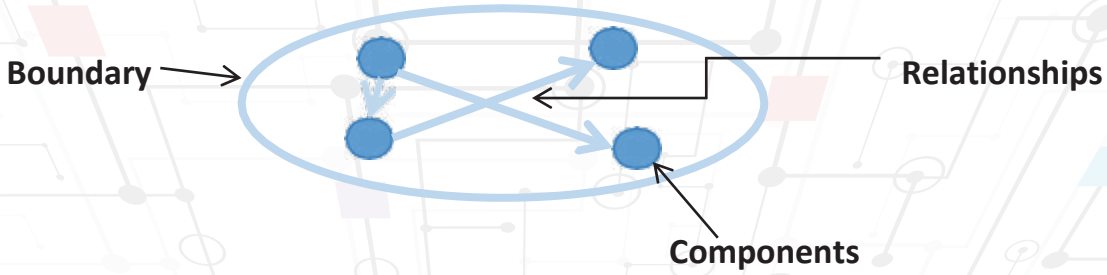
SOFTWARE ENGINEERING DSE

SOFTWARE DEVELOPMENT LIFE CYCLE

Lesson 02 – Software Development Life Cycle

What is a System?

A collection of interrelated parts which act as a whole towards a common goal.



Examples:

- Nervous System
- Communication System
- Business

  (Components of a business System - Marketing, Sales, Manufacturing, Accounts etc.)

Types of Systems

- **Physical**

  Physical system is tangible entities that may be static or dynamic in nature.

- **Open and Closed**

  An open system continually interacts with its environment. A closed system is isolated from environment influences.

- **Sub System and Super System**

  Each system is part of a large system. Sub systems are the smaller systems within a system. Super system denotes extremely large and complex system.

- **Permanent and Temporary System**

  A permanent system is a system enduring for a time span that is long relative to the operation of human. Temporary system is one having a short time span.

- **Natural and Man Made System**

  System which is made by man is called man made system. Systems which are in the environment made by nature are called natural system.

- **Deterministic and Probabilistic**

  A Deterministic system is one in which the occurrence of all events is perfectly predictable. Probabilistic system is one in which the occurrence of events cannot be perfectly predicted.

- **Man-made Information System**

  An information system is the basis for interaction between the user and the analyst. It determines the nature of relationship among decision makers.

- **Natural Systems**

  These are systems that occur in nature, and together they form the ecosystems that make up our natural environment.

Eg: Circulation of Water in our ocean, Weather and Climate Systems, Energy Cycles, Food Chain.

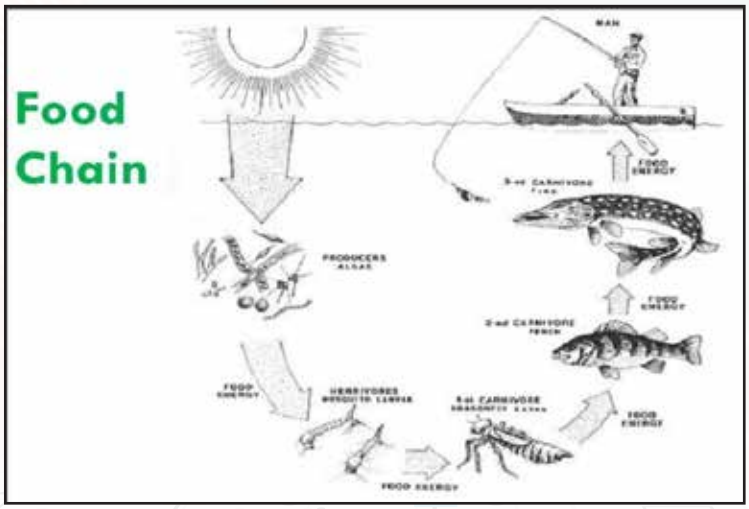


Figure 2.0.1 Natural Systems

- **Information Systems**

  These are systems that are created by people to fill individual and collective needs and wants.

Eg: Human settlements, Transportation routes, Communication systems, Economics, Infrastructure, Energy

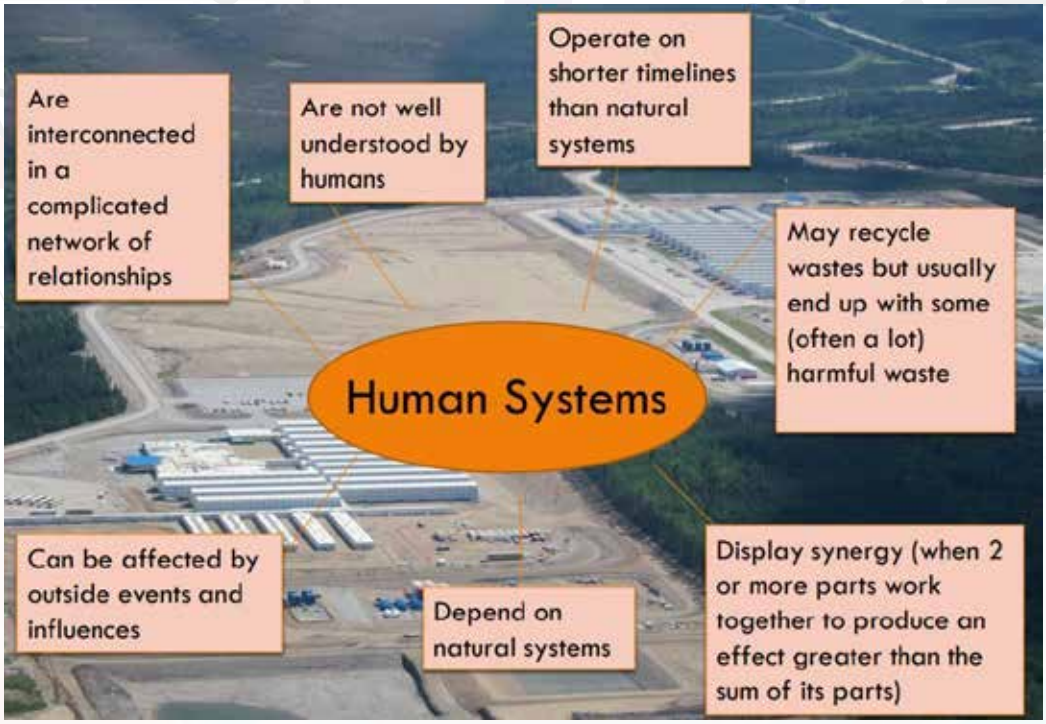


Figure 2.0.2 Human Systems

- **Open Systems**

  Open systems refer to systems that interact with other systems or the outside environment. The boundaries of open systems, because they interact with other systems or environments, are more flexible than those of closed systems, which are rigid, and largely impenetrable.

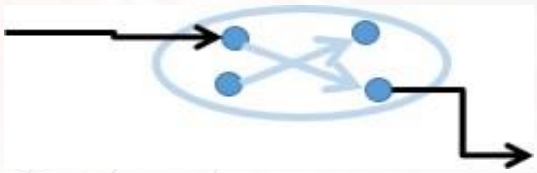


Figure 2.0.3 Open Systems

- **Closed Systems**

  Closed systems refer to systems having relatively little interaction with other systems or the outside environment. A closed system perspective views organizations as relatively independent of environmental influences.

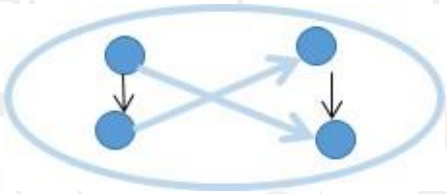


Figure 2.0.4 Closed Systems

What is an Information System?

It is an integrated set of components for collecting, storing, processing, and communicating information.

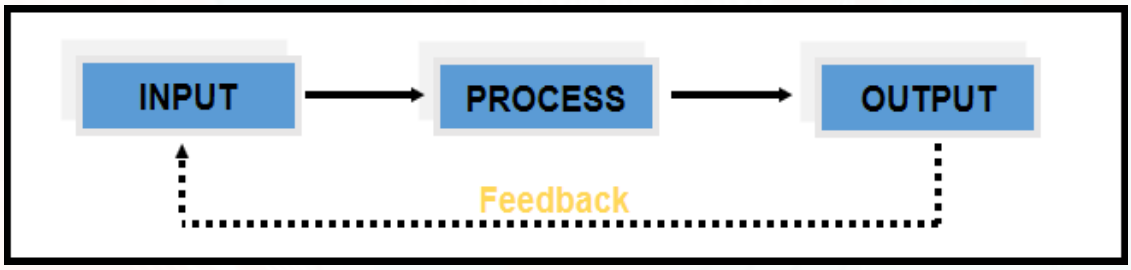


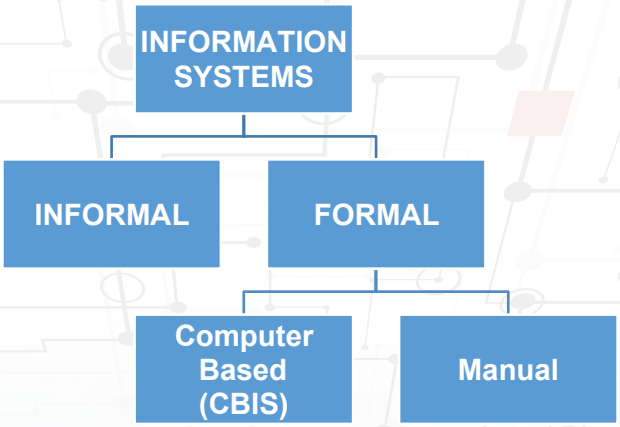
Figure 2.0.5 Flow Diagram of Information System

Components of a Computerized Information System

- Hardware
- Software
- Data Bases
- Personnel

Why Information Systems?

Business firms, other organizations, and individuals in contemporary society rely on information systems to manage their operations, compete in the marketplace, supply services, and augment personal lives.



Types of Information Systems

- Office Automation Systems (OAS)
- Transaction Processing Systems (TPS)
- Management Information Systems (MIS)
- Decisions Support Systems (DSS)
- Executive Support Systems (ESS)
- Geographical information systems (GIS)
- Knowledge Management Systems (KMS)
- Content Management Systems (CMS)
- Enterprise Resource Planning Systems (ERP)
- Smart systems

Office Automation Systems:

The office automation systems, also called OAS consist of applications designed to help the daily work of the administration of an organization, are part of this type of software the word processors, the leaves calculation, the editors of presentations, customer email, etc. When several of these applications are grouped into a single software package for easy distribution and installation, the set is known by the name of office suite.

- Perhaps the most popular software package that can fit the definition of OAS (and to the office suite) is Microsoft Office in all its versions.
- Raw data storage, electronic transfer, and the management of electronic business information comprise the basic activities of an office automation system. Office automation helps in optimizing or automating existing office procedures.

Advantages are:

- Office automation can get many tasks accomplished faster.
- It eliminates the need for a large staff.
- Less storage is required to store data.
- Multiple people can update data simultaneously in the event of changes in schedule.

Transaction Processing System:

A Transaction Processing System (TPS) is a type of information system that collects, stores, modifies and retrieves the data transactions of an enterprise. The success of commercial enterprises depends on the reliable processing of transactions to ensure that customer orders are met on time. The field of transaction processing, therefore, has become a vital part of effective business management.

Examples for TPS:

- Hotel Reservation Systems
- Payroll systems
- Order Entry Systems
- Airline ticket reservation system

Features of Transaction Processing Systems

- Rapid response – fast performance with rapid results.
- Reliability – well-designed backup and recovery with a low failure rate.
- Inflexibility – treat every transaction equally.
- Controlled processing – maintain specific requirements for the roles and responsibilities of different employees.

Types of TPS

- **Batch Processing system**: Batch processing is where the information is collected as a batch and then processed later on. An example of batch processing is paying by cheque. Batch processing is useful for enterprises that need to process large amounts of data using limited resources.
- **Real Time Processing**: Real time processing is where all details of the transaction are recorded and changed at the time as it occurs.

  Examples of real time processing are ATM's.

Management Information Systems:

A management information system (MIS) is an information system used for decision-making, and for the coordination, control, analysis, and visualization of information in an organization.

It produces regular reports on operations for every level of management in a company. The main purpose of the MIS is to give managers feedback about their own performance; top management can monitor the company as a whole.

MIS extract, process and summarizes data from the TPS and provides periodic reports.

Decision Support System:

A decision support system (DSS) is a computer-based application that collects, organizes and analyzes business data to facilitate quality business decision-making for management, operations and planning. DSS analysis helps companies to identify and solve problems, and make decisions.

Knowledge Management Systems

Knowledge management (KM) is the process of creating, sharing, using and managing the knowledge and information of an organization. Knowledge management efforts typically focus on organizational objectives such as improved performance, competitive advantage, innovation, the sharing of lessons learned, integration and continuous improvement of the organization.

A knowledge base software allows the company to publish necessary documents for organization-wide consumption. With right knowledge management tools, the company can put together a collection of best practices, tips for customer support and other documents for workforce enablement.

Best Knowledge Management Software:

- **LiveAgent** is the Most reviewed and #1 Rated help desk software for SMB. Companies like BMW, Yamaha, Huawei, Orange use LiveAgent to deliver customer wow to 150M end users worldwide.

- **Zoho** Connect is a team collaboration app, that unifies people, resources, and the apps they need. Users can share ideas, hold real-time discussions, contact anyone in the network, create their own apps, build their knowledge base.

- **Bitrix24** is a leading free social knowledge management and collaboration platform used by over 4 million companies worldwide. Available in cloud and on-premise with open source code access. Share and discuss ideas, manage knowledge, manage projects and do more with Bitrix24.

- **Confluence** is an open and shared workspace that connects people to the ideas and information they need to do their best work.

Content Management System

A content management system (CMS) is a software application or set of related programs that are used to create and manage digital content.

Features of CMS

- Content creation (allows users to easily create and format content)
- Workflow management (assigns privileges and responsibilities based on roles such as authors, editors and admins)
- Intuitive indexing, search and retrieval features index all data for easy access through search functions and allow users to search by attributes such as publication dates, keywords or author.
- Revision features allow content to be updated and edited after initial publication. Revision control also tracks any changes made to files by individuals.
- Publishing functionality allows individuals to use a template or a set of templates approved by the organization, as well as wizards and other tools to create or modify content.

One major advantage of a CMS is its collaborative nature. Multiple users can log on and contribute, schedule or edit content to be published. Because the interface is usually browser-based, a CMS can be accessed from anywhere by any number of users.

The second major advantage of a CMS is that it allows non-technical people who don't know programming languages to easily create and manage their own web content.

CMS Examples

- Joomla
- Magento
- ModX
- Wix
- Weebly
- Wordpress

Enterprise Resource Planning Systems

Enterprise resource planning (ERP) is business process management software that allows an organization to use a system of integrated applications to manage the business and automate many back office functions related to technology, services and human resources.

ERP software typically integrates all facets of an operation including product planning, development, manufacturing, sales and marketing in a single database, application and user interface.

Some of the most common ERP modules include those for product planning, material purchasing, inventory control, distribution, accounting, marketing, finance and HR.

Figure 2.0.6 ERP System

The following new and continuing computing trends have an impact on the growth of enterprise ERP software:

- **Mobile ERP:** Executives and employees want real-time access to information, regardless of where they are. It is expected that businesses will embrace mobile ERP for the reports, dashboards and to conduct key business processes.
- **Cloud ERP:** The cloud has been advancing steadily into the enterprise for some time, but many ERP users have been reluctant to place data in the cloud. Those reservations have gradually been evaporating, however, as the advantages of the cloud become apparent.
- **Social ERP:** There has been much hype around social media and how important —or not — it is to add to ERP systems. Certainly, vendors have been quick to seize the initiative, adding social media packages to their ERP systems with much fanfare. But some wonder if there is really much gain to be had by integrating social media with ERP.

Smart Systems

Smart systems incorporate functions of sensing, actuation, and control in order to describe and analyze a situation, and make decisions based on the available data in a predictive or adaptive manner, thereby performing smart actions. In most cases the "smartness" of the system can be attributed to autonomous operation based on closed loop control, energy efficiency, and networking capabilities.

- **First-generation smart systems**: object recognition devices, driver status monitoring, and multifunctional devices for minimally invasive surgery.
- **Second-generation smart systems**: active miniaturized artificial organs like cochlear implants or artificial pancreas, advanced energy management systems, and environmental sensor networks.
- **Third-generation smart systems**: combine technical "intelligence" and cognitive functions so that they can provide an interface between the virtual and the physical world.

Software Development Life Cycle (SDLC)

- SDLC is a process followed for a software project, within a software organization. It consists of a detailed plan describing how to develop, maintain, replace and alter or enhance specific software.
- The life cycle defines a methodology for improving the quality of software and the overall development process.
- Traditionally, the Software Development Life Cycle consisted of five stages. That has now increased to seven phases. Increasing the number of steps helped systems analysts to define clearer actions to achieve specific goals.

Seven phases of the SDLC

1. Planning
2. Systems Analysis and Requirement Gathering
3. Systems Design
4. Development (Coding)
5. Integration and Testing
6. Implementation (Deployment)
7. Operations and Maintenance

1. Planning

It identifies whether or not there is the need for a new system to achieve a business's strategic objectives. This is a **preliminary plan (or a feasibility study)** for a company's business initiative to acquire the resources to build on an infrastructure to modify or improve a service.

The purpose of this step is to find out the **scope of the problem and determine solutions**. Resources, costs, time, benefits and other items should be considered at this stage.

2. Systems Analysis and Requirement Gathering

The second phase is where businesses will work on the source of their problem or the need for a change. In the event of a problem, possible solutions are submitted and analyzed to identify the best fit for the ultimate goal(s) of the project. **This is where teams consider the functional requirements of the project or solution.** It is also where system analysis takes place—or analyzing the needs of the end users to ensure the new system can meet their expectations.

Once the requirement is clearly understood, the SRS (Software Requirement Specification) document is created.

3. Systems Design

The third phase describes, in detail, the necessary specifications, features and operations that will satisfy the functional requirements of the proposed system which will be in place. It brings down whole knowledge of requirements and analysis on the desk and design the software product. The output of this step comes in the form of two designs; logical design and physical design. **Engineers produce meta-data and data dictionaries, logical diagrams, data-flow diagrams and in some cases pseudo codes.**

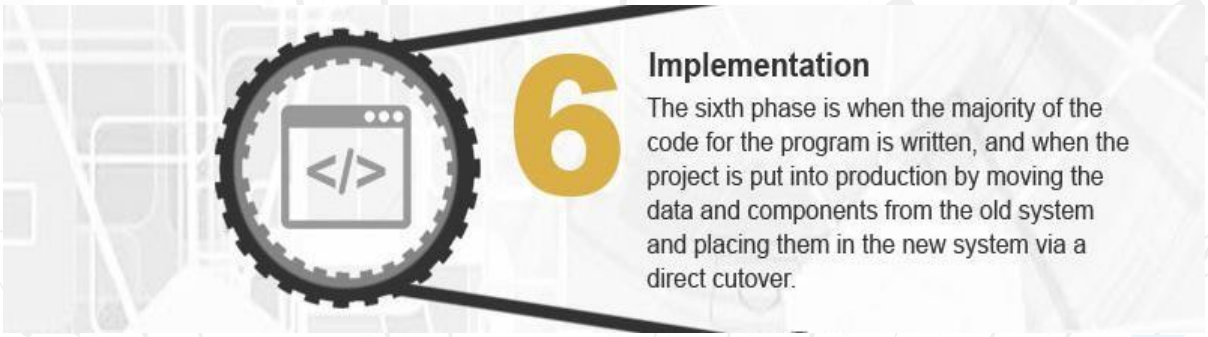
4. Development

The fourth phase is **when the real work begins**—in particular, when a programmer, network engineer and/or database developer are brought on to do the major work on the project. Different high level programming languages such as C, C++, Pascal, Java and PHP are used for coding. The programming language is chosen with respect to the type of software being developed.

5. Integration and Testing

The fifth phase involves systems integration and system testing (of programs and procedures)—normally carried out by a Quality Assurance (QA) professional—**to determine if the proposed design meets the initial set of business goals.** Testing may be repeated, specifically to check for errors, bugs and interoperability. This testing will be performed until the end user finds it acceptable.

6. Implementation



The sixth phase is when the majority of the code for the program is written. Additionally, **this phase involves the actual installation of the newly-developed system.** This step puts the project into production by moving the data and components from the old system and placing them in the new system.

7. Operations and Maintenance

The seventh and final phase **involves maintenance and regular required updates.** This step is when end users can fine-tune the system, if they wish, to boost performance, add new capabilities or meet additional user requirements.

Software Development Process

- When you work to build a product or system, it's important to go through a series of predictable steps—a road map that helps you create a timely, high-quality result. The road map that you follow is called a **"Software Process."**
- **Software Process Model** is an abstract representation of the software development cycle. SDLC models might have a different approach but the basic phases and activity remain the same for all the models.

Water Fall Model

Waterfall model is the very first model that is used in SDLC. It is also known as the linear sequential model.

In this model, the outcome of one phase is the input for the next phase. Development of the next phase starts only when the previous phase is complete.

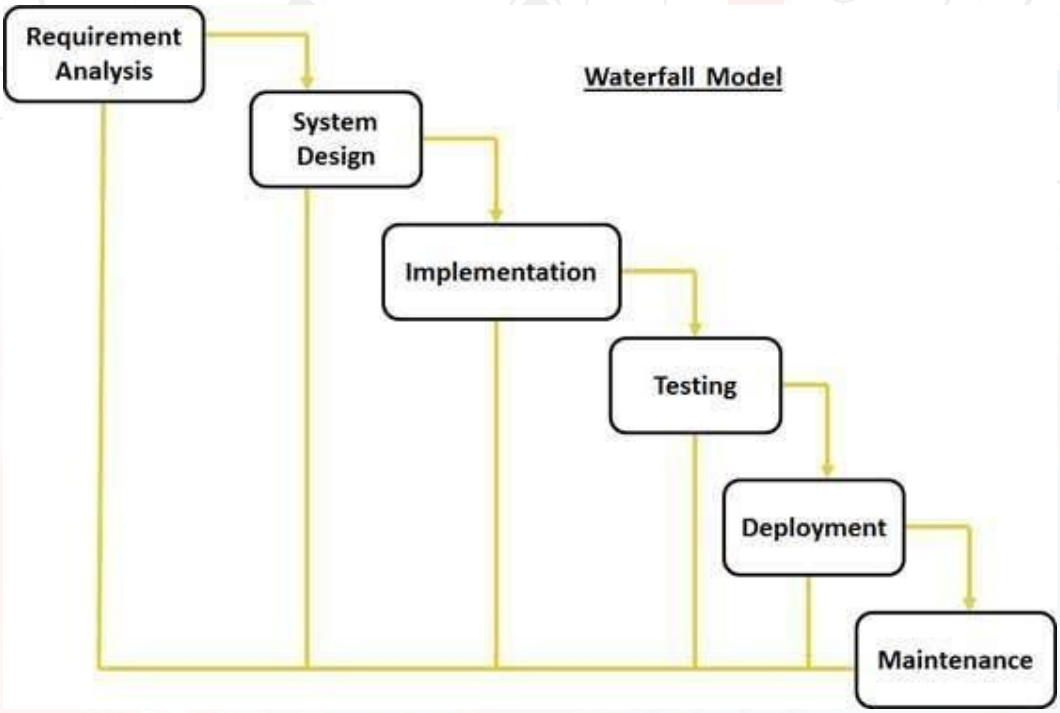


Figure 2.0.7 Waterfall Model

- First, Requirement gathering and analysis is done. Once the requirement is freeze then only the System Design can start.
- In System Design Software Architecture Design, documents which act as an input for the next phase are created.
- In the Implementation phase, coding is done and the software developed is the input for the next phase.
- In the testing phase, the developed code is tested thoroughly to detect the defects in the software. Defects are logged into the defect tracking tool and are retested once fixed.
- In the Deployment phase, the developed code is moved into production after the sign off is given by the customer.

- Any issues in the production environment are resolved by the developers which come under maintenance.

Some situations where the use of Waterfall model is most appropriate are −

- Requirements are very well documented, clear and fixed.
- Product definition is stable.
- Technology is understood and is not dynamic.
- There are no ambiguous requirements.
- Ample resources with required expertise are available to support the product.
- The project is short.

Waterfall Model - Advantages

- Simple and easy to understand and use.
- Easy to manage because each phase has specific deliverables.
- Phases are processed and completed one at a time.
- Works well for smaller projects where requirements are very well understood.
- Well understood milestones.
- Easy to arrange tasks.
- Process and results are well documented.

Waterfall Model - Disadvantages

- No working software is produced until late during the life cycle.
- High amounts of risk and uncertainty.
- Not a good model for complex and object-oriented projects.
- Poor model for long and ongoing projects.
- Not suitable for the projects where requirements are changing.

- It is difficult to measure progress within stages.
- Adjusting scope during the life cycle can end a project.

Examples of Waterfall Model

In the olden days, Waterfall model was used to develop enterprise applications like;

- Customer Relationship Management (CRM) systems
- Human Resource Management Systems (HRMS)
- Supply Chain Management Systems
- Inventory Management Systems,
- Point of Sales (POS) systems for Retail chains.
- Development of Department Of Defense (DOD), military and aircraft programs followed Waterfall model in many organizations.
- This is because of the strict standards and requirements that have to be followed.
- In such industries, the requirements are known well in advance and contracts are very specific about the deliverable of the project.

V Model

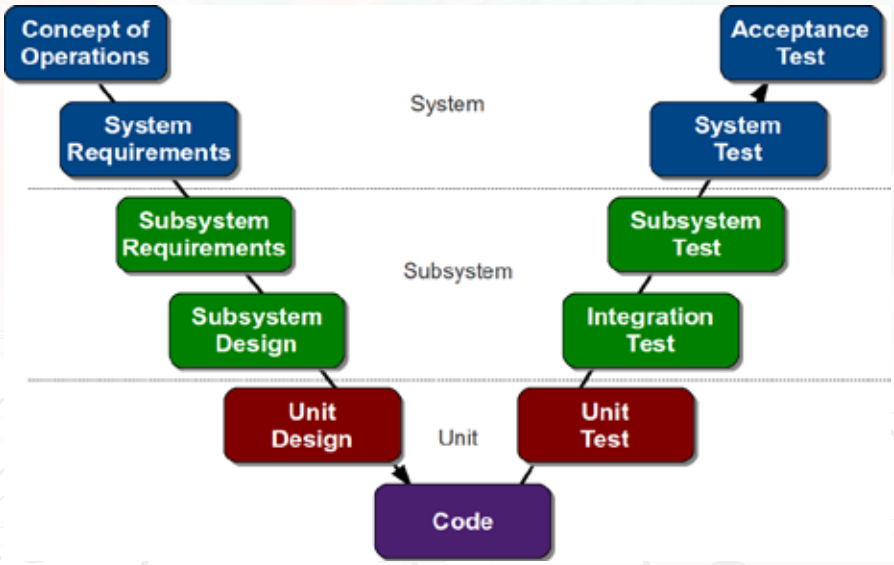


Figure 2.0.8 V Model

V- Model is also known as Verification and Validation Model. In this model Verification & Validation goes hand in hand i.e. development and testing goes parallel.

V model and waterfall model are the same except that the test planning and testing start at an early stage in V-Model.

Following are some of the most suitable scenarios to use the V-Model application

- Requirements are well defined, clearly documented and fixed.
- Product definition is stable.
- Technology is not dynamic and is well understood by the project team.
- There are no ambiguous or undefined requirements.
- The project is short.

V Model - Advantages

- This is a highly-disciplined model and Phases are completed one at a time.
- Works well for smaller projects where requirements are very well understood.
- Simple and easy to understand and use.
- Easy to manage.

V Model - Disadvantages

- High risk and uncertainty.
- Poor model for long and ongoing projects.
- Not suitable for the projects where requirements are changing.
- Once an application is in the testing stage, it is difficult to go back and change.

Iterative Incremental Process Model

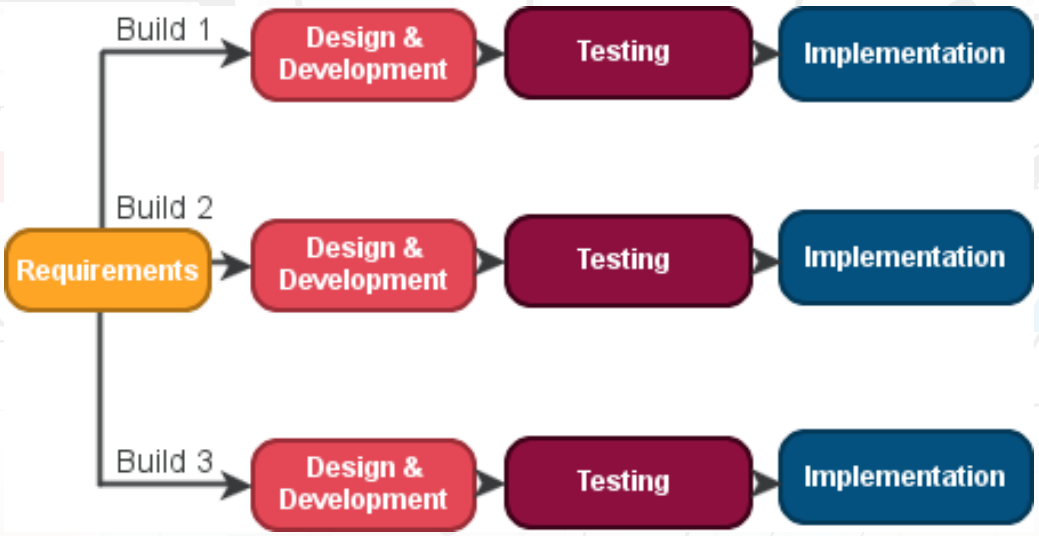


Figure 2.0.9 Iterative Incremental Process Model

The iterative incremental model divides the product into small chunks.

For Example, Feature to be developed in the iteration is decided and implemented. Each iteration goes through the phases namely Requirement Analysis, Designing, Coding, and Testing. **Detailed planning is not required in iterations.**

Once the iteration is completed, a product is verified and is delivered to the customer for their evaluation and feedback. Customer's feedback is implemented in the next iteration along with the newly added feature. Hence, the product increments in terms of features and once the iterations are completed the **final build holds all the features of the product.**

This model is most often used in the following scenarios;

- Requirements of the complete system are clearly defined and understood.
- **Major requirements must be defined;** however, some functionalities or requested enhancements may evolve with time.
- A new technology is being used and is being learnt by the development team while working on the project.
- There are some high-risk features and goals which may change in the future.

Iterative Incremental Process Model – Advantages

- Some working functionality can be developed quickly.
- Results are obtained early and periodically.
- Parallel development can be planned and Progress can be measured.
- Less costly to change the scope/requirements.
- Testing and debugging during smaller iteration is easy.
- Risks are identified and resolved during iteration.
- Easier to manage risk - High risk part is done first.
- With every increment, operational product is delivered.

Iterative Incremental Process Model - Disadvantages

- More resources may be required.
- Although cost of change is lesser, but it is not very suitable for changing requirements.
- More management attention is required.
- System architecture or design issues may arise because not all requirements are gathered in the beginning of the entire life cycle.
- Not suitable for smaller projects.
- Management complexity is more.
- Highly skilled resources are required for risk analysis.

Spiral Model

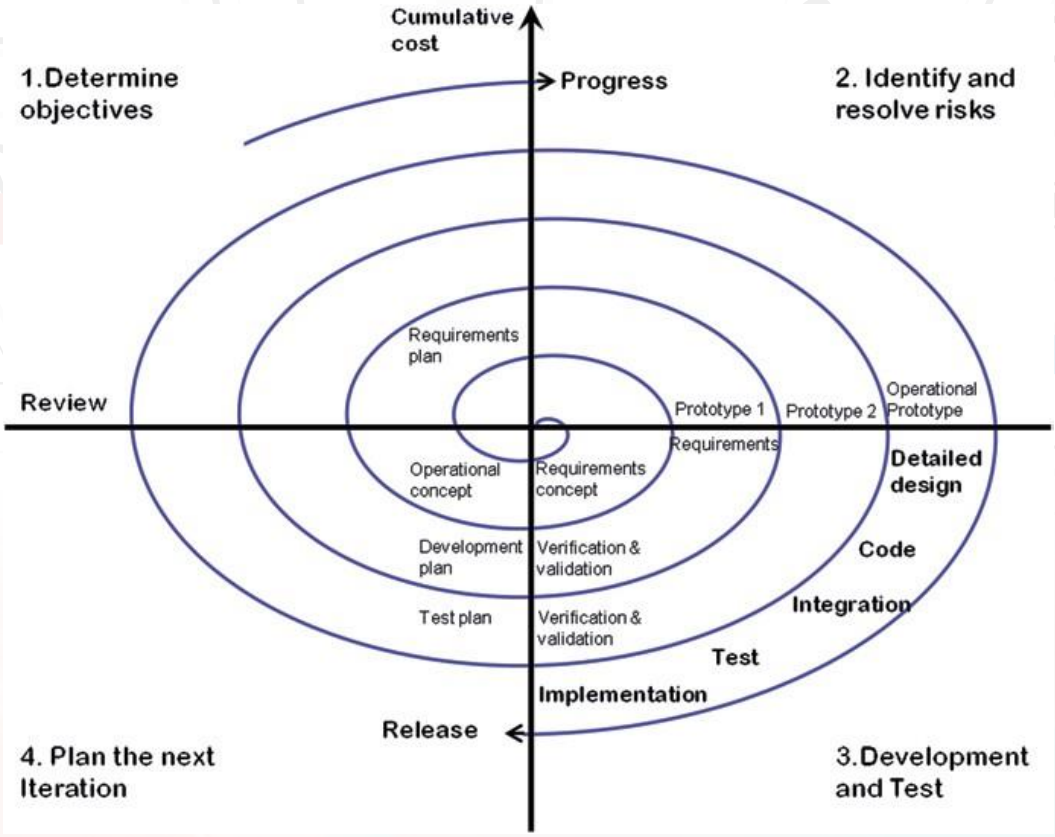


Figure 2.0.10 Spiral Model

The Spiral Model includes iterative and prototype approach.

Spiral Model has four phases:

- **Planning:**

The planning phase includes requirement gathering wherein all the required information is gathered from the customer and is documented.

- **Risk Analysis:**

In this phase, the best solution is selected for the risks involved and analysis is done by building the prototype.

For Example, the risk involved in accessing the data from a remote database can be that the data access rate might be too slow. The risk can be resolved by building a prototype of the data access subsystem.

- **Engineering:**

Once the risk analysis is done, coding and testing are done.

- **Evaluation:**

Customer evaluates the developed system and plans for the next iteration.

Based on the customer evaluation, the software development process enters the next iteration and subsequently follows the linear approach to implement the feedback suggested by the customer. The process of iterations along the spiral continues throughout the life of the software.

This model is most often used in the following scenarios;

- When there is a budget constraint and risk evaluation is important.
- For medium to high-risk projects.
- Customer is not sure of their requirements which is usually the case.
- Requirements are complex and need evaluation to get clarity.
- New product line which should be released in phases to get enough customer feedback.

Spiral Model - Advantages

- Changing requirements can be accommodated.
- Allows extensive use of prototypes.
- Requirements can be captured more accurately.
- Users see the system early.
- Development can be divided into smaller parts and the risky parts can be developed earlier which helps in better risk management.

Spiral Model - Disadvantages

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

- Not suitable for small or low risk projects and could be expensive for small projects.
- Process is complex.
- Large number of intermediate stages requires excessive documentation.

Prototype Model

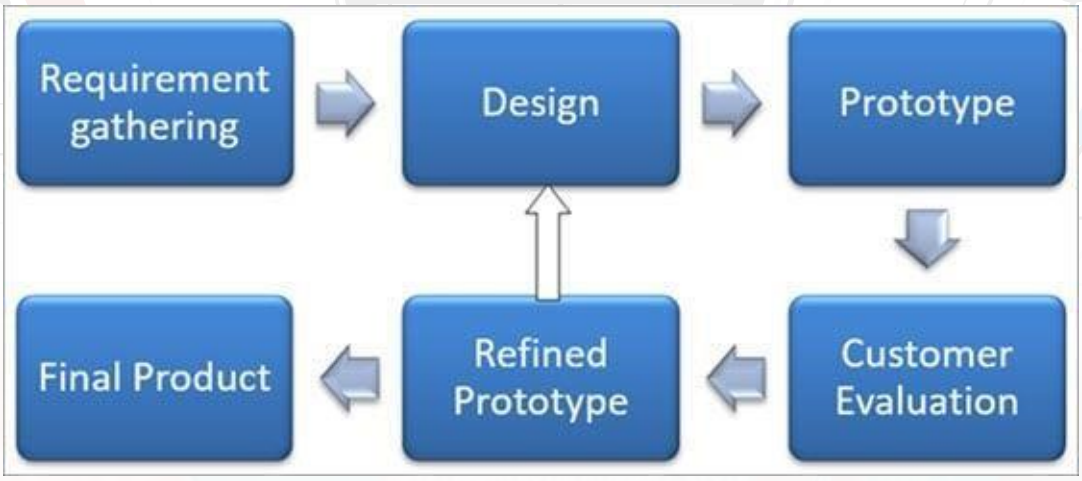


Figure 2.0.11 Prototype Model

The Prototype Model is a model in which the prototype is developed prior to the actual software. Prototype Models have limited functional capabilities and inefficient performance when compared to the actual software. Dummy functions are used to create prototypes. This is a valuable mechanism for understanding the customers' needs. Software prototypes are built prior to the actual software to get valuable feedback from the customer.

Feedbacks are implemented and the prototype is again reviewed by the customer for any change. This process goes on until the model is accepted by the customer. Following Software use to design Prototype quickly,

- AXURE RP:

https://www.axure.com/

- INVISION:

https://www.invisionapp.com/

Prototype Model - Advantages

- Increased user involvement in the product even before its implementation.
- Since a working model of the system is displayed, the users get a better understanding of the system being developed.
- Reduces time and cost as the defects can be detected much earlier.
- Quicker user feedback is available leading to better solutions.
- Missing functionality can be identified easily.
- Confusing or difficult functions can be identified.

Prototype Model - Disadvantages

- Risk of insufficient requirement analysis owing to too much dependency on the prototype.
- Users may get confused in the prototypes and actual systems.
- Practically, this methodology may increase the complexity of the system as scope of the system may expand beyond original plans.
- Developers may try to reuse the existing prototypes to build the actual system, even when it is not technically feasible.
- The effort invested in building prototypes may be too much if it is not monitored properly.

RAD (Rapid Application Development)

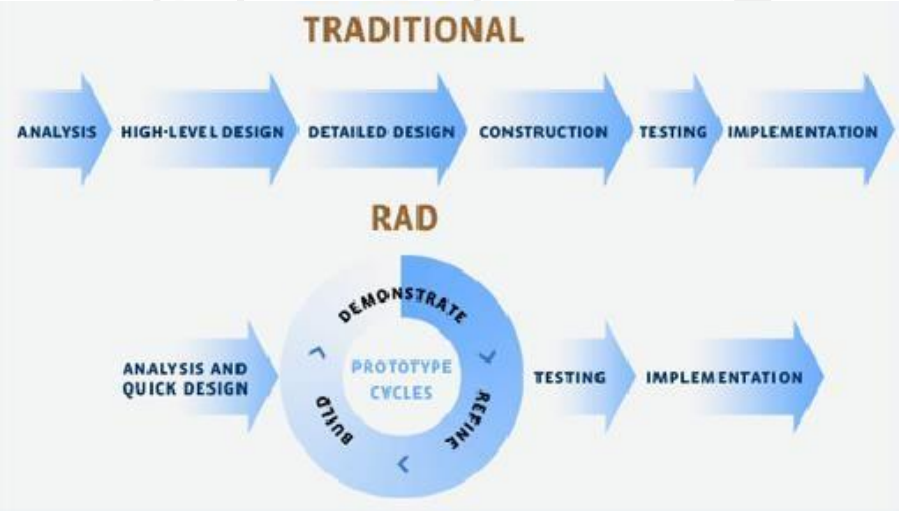


Figure 2.0.12 Rapid Application Development

The RAD (Rapid Application Development) model is based on prototyping and iterative development with no specific planning involved.

In the RAD model, the functional modules are developed in parallel as prototypes and are integrated to make the complete product for faster product delivery.

The most important aspect for this model to be successful is to make sure that the prototypes developed are reusable.

- DSSs include knowledge-based systems. A properly designed DSS is an interactive software-based system intended to help decision makers compile useful information from a combination of raw data, documents, and personal knowledge, or business models to identify and solve problems and make decisions.
- A decision support system may present information graphically and may include an expert system or artificial intelligence (AI).

Typical information that a decision support application might gather and present would be:

- Comparative sales figures between one week and the next.
- Projected revenue figures based on new product sales assumptions.
- The consequences of different decision alternatives, given past experience in a context that is described.

Executive Support Systems:

Executive Support System (ESS) is a reporting tool (software) that allows you to turn your organization's data into useful summarized reports. These reports are generally used by executive level managers for quick access to reports coming from all company levels and departments such as billing, cost accounting, staffing, scheduling, and more.

In addition to providing quick access to organized data from departments, some Executive Support System tools also provide analysis tools that predicts a series of performance outcomes over time using the input data.

Advantages of EIS

- Easy for upper-level executives to use, extensive computer experience is not required in operations.
- Provides strong drill-down capabilities to better analyze the given information.
- Information that is provided is better understood.
- EIS provides timely delivery of information. Management can make decisions promptly.
- Improves tracking information.
- Offers efficiency to decision makers.

Geographical Information Systems:

A geographic information system (GIS) is a system designed to capture, store, manipulate, analyze, manage, and present spatial or geographic data. GIS applications are tools that allow users to create interactive queries (user-created searches), analyze spatial information, edit data in maps, and present the results of all these operations.

Geographic Information Systems really comes down to just 4 simple ideas: These are the primordial functions of a GIS.

- Create geographic data
- Manage it
- Analyze it and
- Display it on a map

GIS can be used to Mapping where things are, Mapping quantities, Mapping densities, finding what is inside, finding what is nearby, Mapping changes.

Following are some open-source desktop GIS projects.

- GRASS GIS – Originally developed by the U.S. Army Corps of Engineers: a complete GIS.

- ILWIS (Integrated Land and Water Information System) – Integrates image, vector and thematic data.

- MapWindow GIS – Free desktop application and programming component.

- QGIS (previously known as Quantum GIS) – Runs on Linux, Unix, Mac OS X and Windows.

- SAGA GIS (System for Automated Geoscientific Analysis) –- A hybrid GIS software. Has a unique Application Programming Interface (API) and a fast-growing set of geoscientific methods, bundled in exchangeable Module Libraries.