Lecture 9:
SDLC and Project Management

CSS 200 - Intro to Information Systems

Module 1

- What is an information system?
- Where do we use information systems?
- What is the difference between Data, Information and Knowledge?

What is an information system?

- An information system is a combination of technology, people, and processes that work together to collect, store, manage, and share data. It helps organizations make decisions, solve problems, and improve efficiency by providing accurate and timely information.

Where do we use information systems?

- Information systems are used in various sectors like business, education, and healthcare to support daily operations and long-term planning. They include hardware, software, databases, and networks, all designed to process and distribute information to users who need it.

What is the difference between Data, Information and Knowledge?

- Data refers to raw, unorganized facts or figures that by themselves have no meaning. For example, numbers, dates, or a list of names are considered data.
- Information is what you get when data is processed, organized, or structured in a way that adds context and meaning. For instance, data about sales figures organized in a report becomes information that can be used to understand business performance.
- Knowledge goes a step further and is the understanding or insight gained from analyzing information. It involves interpreting information and applying it to make decisions or solve problems, such as using sales information to predict future trends or improve strategies.

Module 2

- Explain the role of Enterprise Architecture in IT Governance
- Networking Devices: Hub, Repeater, Switch, Router, Gateway

Understanding Enterprise Architecture in IT Governance

What is Enterprise Architecture (EA)?

- Think of EA as a framework for how an organization's IT (technology) and business processes work together. It helps visualize and organize the different components like systems, data, and processes.

What is IT Governance?

- IT Governance is like a set of rules and guidelines that ensure the organization's IT supports its goals. It helps make sure that technology is used wisely and responsibly.

How Does EA Help with IT Governance?

- Alignment with Business Goals: EA ensures that IT projects and initiatives align with what the business wants to achieve. It's like making sure everyone is moving toward the same goal.
- Standardization: EA helps create standard processes and systems across the organization. This consistency makes it easier to manage and reduces confusion.
- Risk Management: By providing a clear view of all IT components, EA helps identify potential risks (like security issues) and allows organizations to plan ahead to avoid them.
- Informed Decision-Making: EA gives leaders a comprehensive view of technology and business processes, enabling them to make better decisions about where to invest and how to improve.
- Performance Measurement: EA often includes metrics that help track how well IT is performing. This allows organizations to see what's working and what isn't.

How Does EA Help with IT Governance?

- Managing Change: As businesses evolve, EA provides guidance on how to introduce new technologies or processes smoothly, reducing disruption.
- Improved Communication: EA acts as a common language that helps different parts of the organization communicate better, making collaboration easier.
- Regulatory Compliance: EA helps organizations ensure they are following laws and regulations related to technology, making it easier to prove compliance when needed.
- Resource Optimization: By identifying overlapping technologies or processes, EA helps organizations use their resources more effectively, saving time and money.
- Long-term Planning: EA encourages looking ahead and planning for future technology needs, ensuring the organization remains adaptable and sustainable.

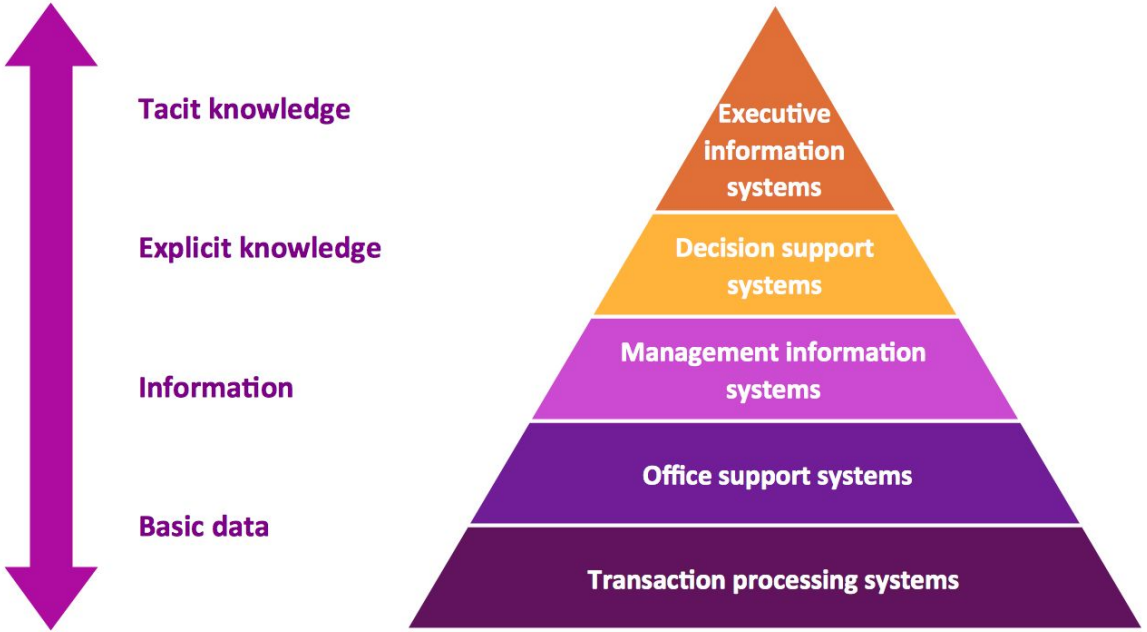
Networking Devices

- Network Definition: A network is a group of connected devices that share data and resources. These networks can vary in scale, from small home setups to global enterprise systems.
- Hub: Broadcasts data to all connected devices. It lacks intelligence, sending data to everyone instead of the intended recipient.
- Switch: Intelligently forwards data only to the intended device within a local network, reducing congestion and allowing full-duplex communication.
- Router: Connects different networks and determines the best path for data between them using IP addresses. Essential for internet connectivity.
- Repeater: Amplifies and retransmits weak signals to extend network range. Operates at the physical layer.
- Gateway: Acts as a translator between different networks, enabling communication by converting protocols.

Module 3

- Explain the role and objectives of Customer Relationship Management (CRM) and Supply Relationship Management (SRM).
- Transaction Processing Systems (TPS)
- Office Automation Systems (OAS)
- Management Information Systems (MIS)
- Decision Support Systems (DSS)
- Executive Information Systems (EIS)

Types of Information Systems Overview



Hierarchy of Information Systems: From Data to Knowledge

- Transaction Processing Systems (TPS): These systems handle basic data, primarily concerned with the day-to-day transactions of an organization. They are foundational, dealing with large volumes of operational data like sales, inventory, and payroll.
- Office Support Systems (OSS): These systems help with the daily operations within an office environment, such as document management, communication (e.g., email), and basic collaboration tools.
- Management Information Systems (MIS): At this level, systems are used to convert raw data from transaction systems into more structured information. MIS provides middle management with reports and summaries, supporting routine decision-making.
- Decision Support Systems (DSS): These systems are used for more complex decision-making, offering tools for data analysis, forecasting, and simulation. DSS helps in processing explicit knowledge, giving managers insights to make informed decisions on non-routine matters.
- Executive Information Systems (EIS): At the top of the hierarchy, these systems are designed for top-level executives. They focus on summarizing and presenting key performance indicators and strategic information, often dealing with tacit knowledge (unwritten, intuitive knowledge) that guides high-level decision-making.

Module 3

- Explain the role and objectives of Customer Relationship Management (CRM) and Supply Relationship Management (SRM).
- Transaction Processing Systems (TPS)
- Office Automation Systems (OAS)
- Management Information Systems (MIS)
- Decision Support Systems (DSS)
- Executive Information Systems (EIS)

Summary

System	Purpose	Users	Key Features	Example
TPS	Handle routine, high-volume transactions	Operational staff (clerks, cashiers)	Structured, repetitive, real-time processing	POS systems, payroll systems
OAS	Automate routine office tasks	Clerical staff, knowledge workers	Productivity software (word processing, emails, etc.)	Microsoft Office suite
MIS	Provide reports for decision-making	Middle management	Summarized reports from structured data	Sales management systems
DSS	Support decision-making with data analysis	Managers, analysts	Analytical tools, "what-if" analysis, simulations	Forecasting, investment systems
EIS	Provide top-level information for executives	Executives, senior managers	High-level summaries, real-time dashboards	Executive dashboards

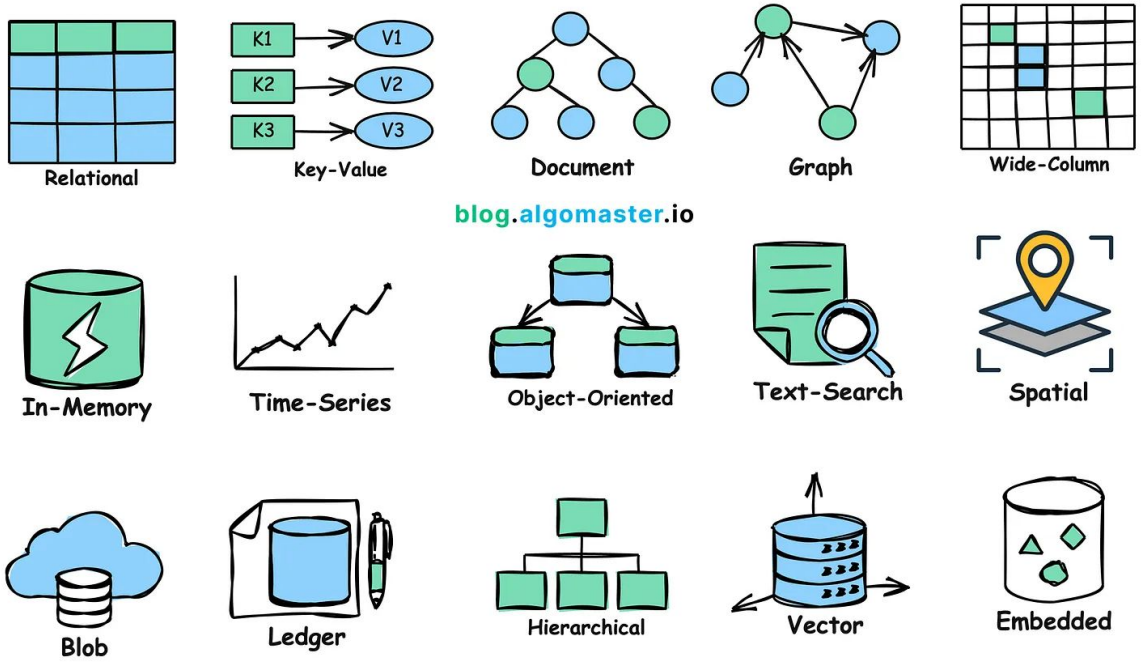
Module 4

- Microsoft Access
- Relational Databases (RDBMS)
- Key-Value Store
- Document Databases
- Graph Databases
- Object-Oriented Databases
- Hierarchical Databases

Database vs. Database Management System (DBMS)

- A Database and a Database Management System (DBMS) are closely related terms, but they serve different purposes:
- A database is a structured set of data. The data can be structured or unstructured and stored in various formats like tables, documents, and key-value pairs. It could be anything from a simple shopping list to a picture gallery or the vast amount of information in a corporate network.
- A Database Management System (DBMS) is software used to interact with a database. It provides an interface for users or applications to manipulate data, making the handling of large amounts of data more efficient and less error-prone. A DBMS oversees core administrative tasks such as data storage, retrieval, security, and query processing.

Different Types Of Databases



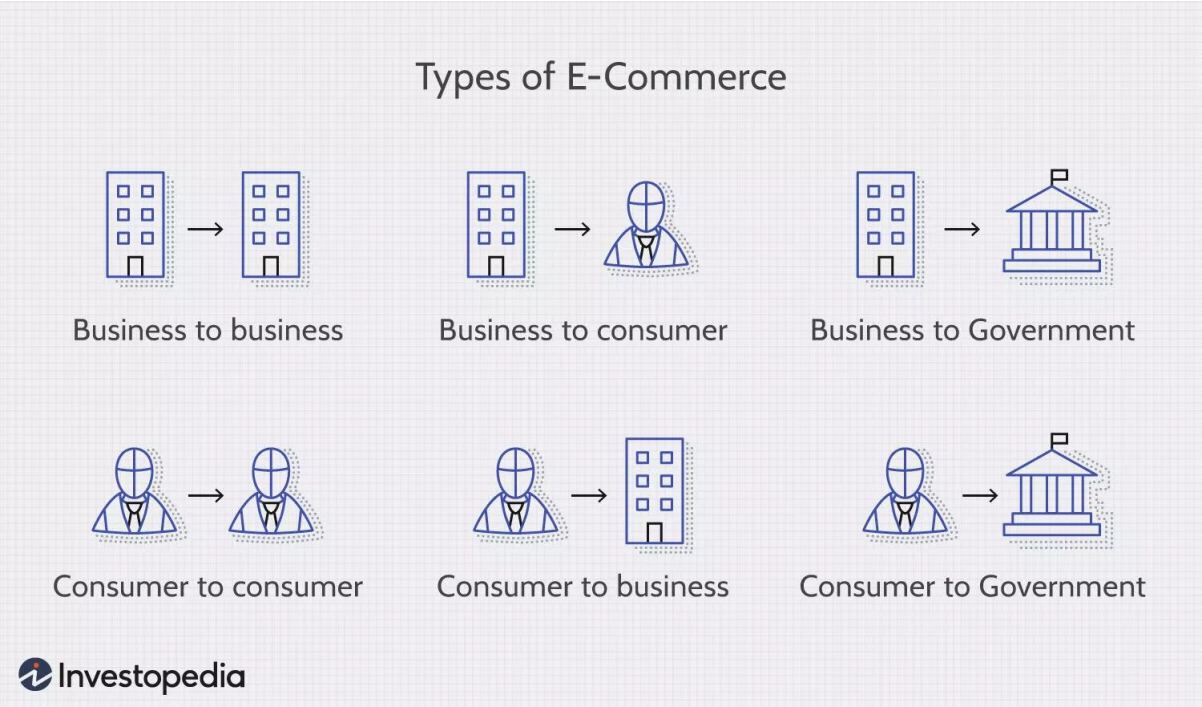
Summary

Database Type	Data Structure	Use Cases	Advantages	Examples
Relational Databases	Tables with rows and columns, structured relationships (SQL-based)	Enterprise applications, banking, e-commerce platforms	Data integrity, complex queries	MySQL, PostgreSQL, Oracle DB
Key-Value Store	Key-value pairs	Caching, session, storage, real-time data processing	Simple, fast retrieval, highly scalable	Redis, DynamoDB
Document Databases	Semi-structured documents	Content management, real-time analytics, IoT	Flexible schema, fast reads/writes, good for evolving data	MongoDB, Couchbase,Apache Couchbase
Graph Databases	Graphs, nodes, edges, properties	Social networks, recommendation systems, knowledge graphs	Efficient traversal of connected data, flexible querying	Neo4j, Amazon Neptune
Object-Oriented Databases	Objects (similar to OOP languages)	Object-oriented applications, multimedia databases	Seamless OOP integration, efficient object management	ObjectDB, db4o
Hierarchical Databases	Tree-like structure (parent-child relationships)	Organizational charts, file systems	Efficient for one-to-many relationships	IBM IMS, Windows Registry

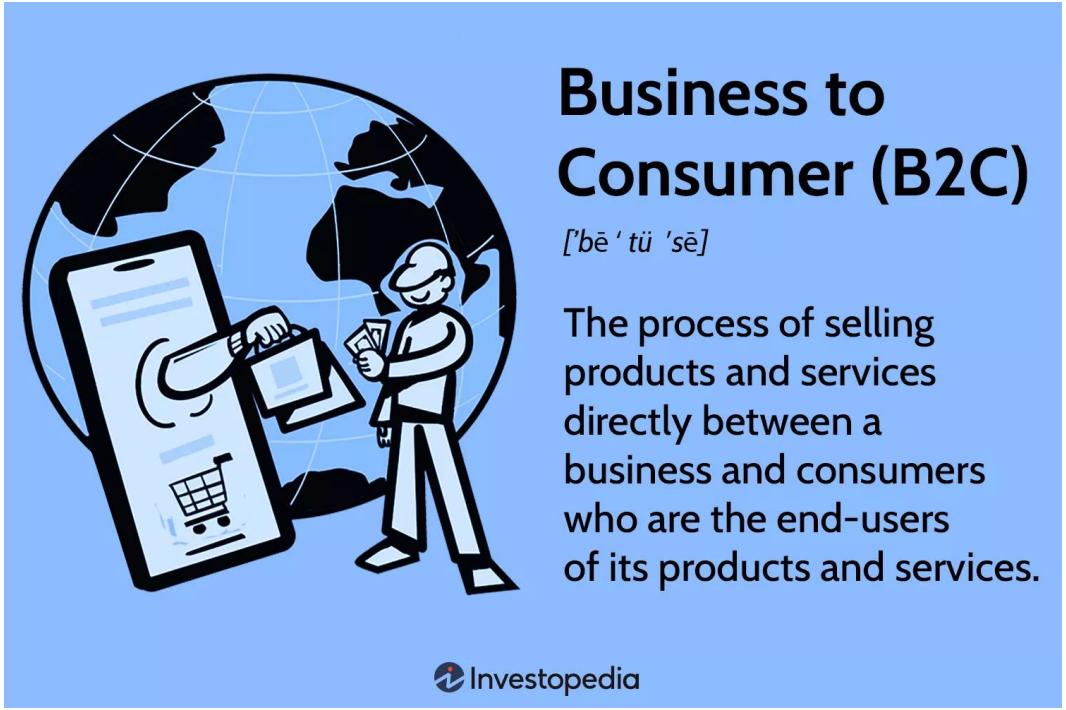
Module 5

- What is e-commerce?
- What types of e-commerce are there?
  - B2C
  - B2B
  - C2C
- What is HTML?
  - Headings
  - Paragraphs
  - Links
  - Lists
  - Forms
- What is CSS?

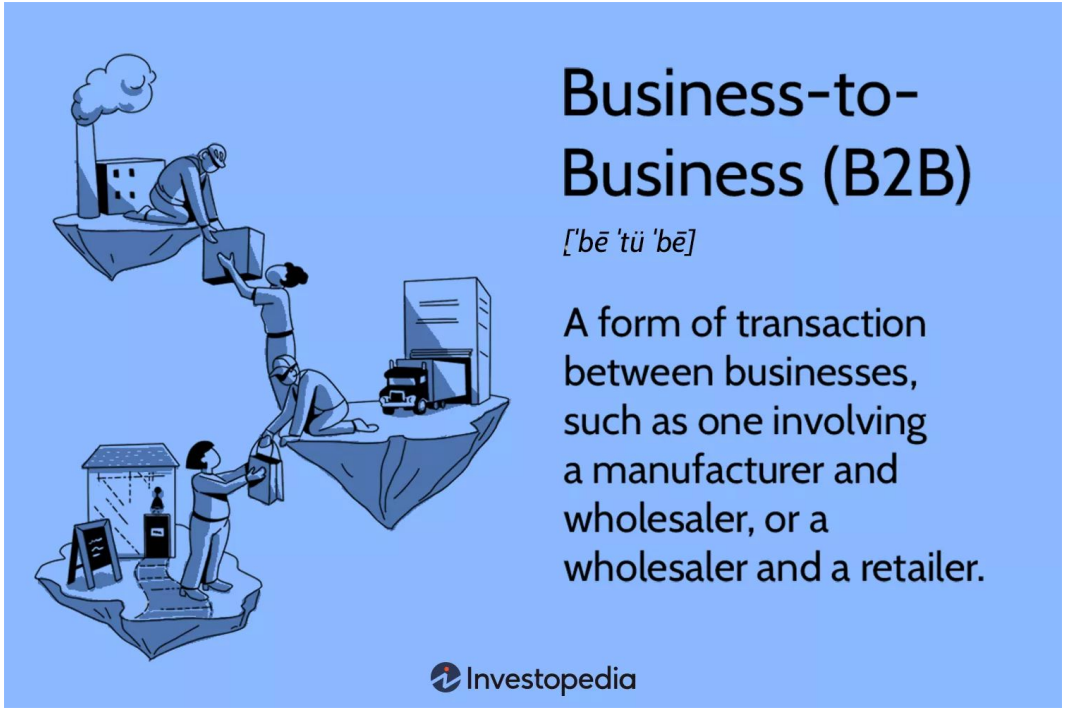
Types of e commerce



Business to Consumer (B2C)



Business to Business (B2B)



Customer to Customer (C2C)

C2C / B2C / B2B Comparison

Characteristic	C2C	B2C	B2B
Definition	Transaction between consumers	Transactions between businesses and consumers	Transactions between businesses
Target Audience	Individual consumers	General public	Other businesses or organizations
Platform Type	Marketplaces or auction sites	Retail websites	Wholesale platforms or direct sales
Example Business	eBay	Amazon	Alibaba

Introduction to HTML

- What is HTML?
  - HTML stands for HyperText Markup Language. It's the language used to create web pages.
  - HyperText refers to links that connect web pages.
  - Markup Language means that it uses tags to define elements within a document.
- What does HTML do?
  - It structures content on the web. It DOESN'T style or control how the content looks (that's CSS).
  - HTML is the foundation of any web page. It organizes text, images, links, and other content into a coherent structure.

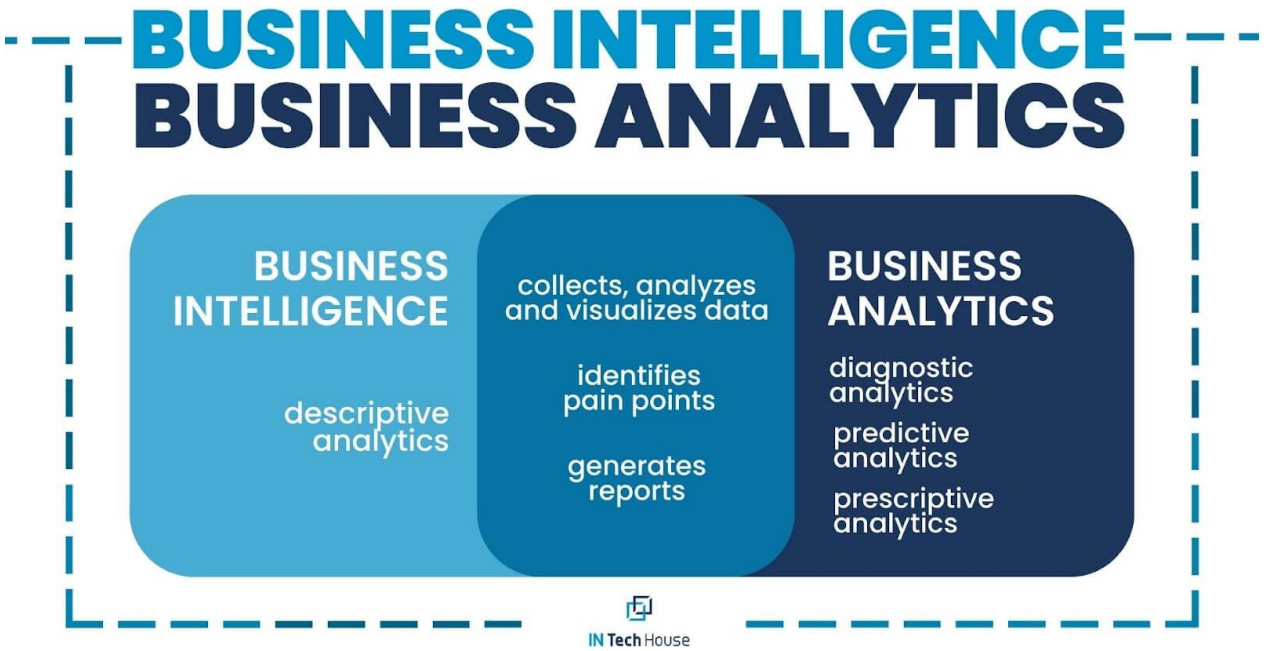
Introduction to CSS

- What is CSS?
  - CSS stands for Cascading Style Sheets. It's used to style and layout web pages.
- What does CSS do?
  - CSS controls the appearance of HTML elements, such as colors, fonts, layout, and spacing.
  - Separates the structure (HTML) from the presentation (CSS).

Module 6

- What is Business Intelligence?
- What is Business Analytics?
- Difference between Business Intelligence and Business Analytics
- Decision Support Systems
- Types of Decision Support Systems
- Components of a Decision Support System
- Examples of Decision Support System Software

Business Intelligence vs. Business Analytics



What is Business Intelligence?

- Traditionally, business intelligence has been defined as the use of data to manage day-to-day operational management within a business.
- Business intelligence tools can include a variety of software tools and other systems. Some of these include spreadsheets, online analytical processing, reporting software, business activity monitoring software, and data mining software.
- Overall, business intelligence helps leaders navigate organizational and industry-related challenges and ensures that companies stay focused on their primary target to successfully get where they want to go.

What is Business Analytics?

- Business analytics has generally been described as a more statistical-based field, where data experts use quantitative tools to make predictions and develop future strategies for growth.
- For example, while business intelligence might tell business leaders what their current customers look like, business analytics might tell them what their future customers are doing.
- Business analytics tools are employed for many functions, including correlational analysis, regression analysis, forecasting analysis, text mining, image analytics, and others.

Decision Support Systems Definition

- A decision support system (DSS) is an interactive information system that analyzes large volumes of data for informing business decisions.
- A DSS supports the management, operations, and planning levels of an organization in making better decisions by assessing the significance of uncertainties and the tradeoffs involved in making one decision over another.

Module 7

- What is Knowledge Management?
- What is a Knowledge Management system?
- Describe at least two types of Knowledge Management systems.
- Types of Knowledge

What is knowledge management?

- Knowledge management involves a sequence of processes involving the storage, management, sharing, and usage of an organization's knowledge and information.
- The objective is to efficiently store organizational knowledge for optimal utilization.
- This continuous process emphasizes identifying and refining organizational knowledge, ensuring accessibility, and fostering a culture of continuous sharing and learning.

What is a knowledge management system?

- A knowledge management system (KMS) is software designed to facilitate the creation, organization, sharing, and utilization of knowledge within an organization.
- It includes features that facilitate the systematic gathering, storage, retrieval, and sharing of knowledge.
- A KMS aims to enhance overall organizational efficiency by providing tools and processes that enable individuals and teams to access, contribute to, and leverage the collective knowledge of the organization.

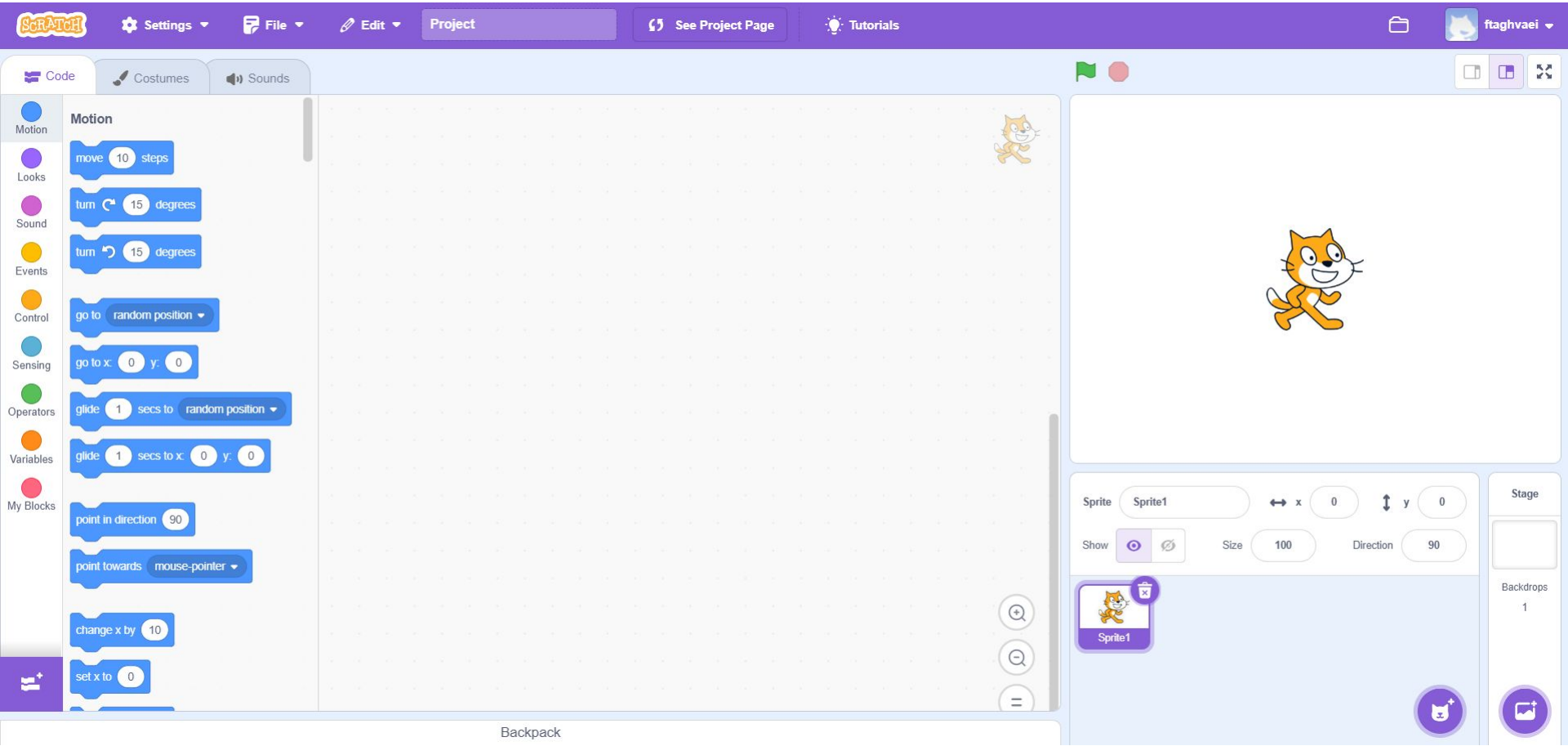
Types of Knowledge

Module 8

- What is Scratch?
- Scratch Interface Overview

What is Scratch?

- A block-based programming language developed by MIT.
- No need to write code—drag and drop colored blocks to build programs.


Settings
File
Edit
Project
See Project Page
Tutorials
ftaghvaei
Code
Costumes
Sounds
Motion
Looks
Sound
Events
Control
Sensing
Operators
Variables
My Blocks
Motion
move 10 steps
turn 15 degrees
turn 15 degrees
go to random position
go to x: 0 y: 0
glide 1 secs to random position
glide 1 secs to x: 0 y: 0
point in direction 90
point towards mouse-pointer
change x by 10
set x to 0
Backpack
Sprite
Sprite1
x 0
y 0
Show
Size 100
Direction 90
Stage
Backdrops 1

Scratch Interface Overview

Block Categories:

- Motion: Controls movement.
- Looks: Manages appearance and speech.
- Sound: Adds sound effects.
- Events: Starts actions, such as clicking a flag or key.
- Control: Adds loops, waits, and conditional blocks.
- Sensing, Operators, Variables: For advanced control.

What will we cover today?

- MLO 1: Identify the phases necessary for the Software Development Life Cycle (SDLC): planning, analysis, design, implementation, and maintenance.
- MLO 2: Develop hands-on experience using MS Office.
- MLO 3: Identify Project Management Skills and Resources.
- MLO 4: Explain challenges encountered while designing the game in Scratch and how they were addressed.

What is Project Management?

- Project management is the planning and organization of a company's resources to move a specific task or event toward completion. It can involve a one-time project or an ongoing activity, and resources managed include personnel, finances, technology, and intellectual property.

Project Management

Phase 1: Initiation

- The initiation phase serves as the foundation of the project, where an idea is nurtured into a clear vision. It involves brainstorming, feasibility studies, and refining the project concept to align with organizational goals and stakeholder needs. In the initiation stage, project objectives are defined, along with the scope of work and the desired outcomes.

Phase 2: Planning

- Planning is the phase where the roadmap for the project is crafted. Planning involves breaking down the project into manageable tasks, sequencing them logically, estimating resources, and developing a comprehensive project plan.
- Planning usually includes some sort of resource allocation tasks.
- Resource allocation ensures that the necessary people, materials, and budget are available when needed.
- Risk management identifies potential threats when those resources may not be achievable at the right time or quantity.

Phase 3: Execution

- The execution phase is where the project plan comes to life, and the project team starts working. Tasks are assigned, and the project team members collaborate. Quality assurance processes are implemented to verify that project meet the specified quality standards. The execution phase is the period of intense activity as the project potentially visibly progresses toward its goals.

Phase 4: Monitoring

- Once the project is underway, the monitoring phase involves tracking project performance against the plan, identifying any deviations or issues, and taking corrective action. Change management processes are implemented to address any changes to the project scope, schedule, or resources. Project managers use this phase to tackle any obstacles that come up (i.e. late deliveries, personnel being unavailable, etc.).

Phase 5: Closing

- In the closing phase, the final deliverables are handed over to the customer or end user, and any remaining administrative tasks such as contract closure completed. It's usually a good idea to debrief on "lessons learned" to implement better processes or project management techniques for future similar projects.

Waterfall Project Management

- Each task needs to be completed before the next one starts. Steps are linear and progress flows in one direction—like a waterfall. Because of this, attention to task sequences and timelines is very important in this type of project management.

Agile Project Management

- Agile project management does not follow a sequential stage-by-stage approach. Instead, phases of the project are completed in parallel to each other by various team members in an organization. This approach can find and rectify errors without having to restart the entire procedure.

Lean Project Management

- This methodology is all about avoiding waste, both of time and of resources. The main idea is to create more value for customers with fewer resources. When managing a project with this approach, the goal is similar to that of the lean enterprise production principle. The only resources that will be used on the project are those that directly contribute to its successful completion.

Six Sigma Project Management

- Six Sigma utilizes a structured methodology known as DMAIC: Define, Measure, Analyze, Improve, and Control.
- In the Define phase, project goals and objectives are clarified, and key metrics are established to measure process performance.
- The Measure phase involves collecting relevant data and analyzing process performance against established metrics to identify areas for improvement.

Scrum Project Management

- Scrum is a popular agile framework designed to enhance team collaboration and deliver value iteratively. Scrum breaks down project work into manageable units called sprints, usually lasting between one to four weeks. Each sprint begins with a planning session where the team selects a set of tasks from the product backlog to complete during the sprint. Once the sprint starts, the team works collaboratively to achieve the sprint goal, holding daily stand-up meetings to discuss progress, address challenges, and adapt as needed.

Project Management Tools

- Project management software: provides a digital platform for organizing, planning, and tracking project activities. Some more common project management software tools include Microsoft Project, Asana, Trello, or Jira.
- Communication tools: facilitate real-time collaboration and communication among project teams, stakeholders, and project managers. For example, Slack is a popular messaging platform that lets team members to communicate through channels, direct messages, and file sharing. Other examples include Microsoft Teams or Zoom.

What Makes a Good Project Plan?

- Communication is key to a good project plan. Each team's responsibilities should be detailed with a goal, a time frame, and resources available, for example.
- Visual explanations such as Gantt charts also are helpful. These are bar graphs that can show each stage of a project and, for example, the time when that stage will take place.

What is a Gantt Chart?

- A Gantt chart, commonly used in project management, is one of the most popular and useful ways of showing activities (tasks or events) displayed against time.
- On the left of the chart is a list of the activities and along the top is a suitable time scale.
- Each activity is represented by a bar; the position and length of the bar reflects the start date, duration and end date of the activity. This allows you to see at a glance:

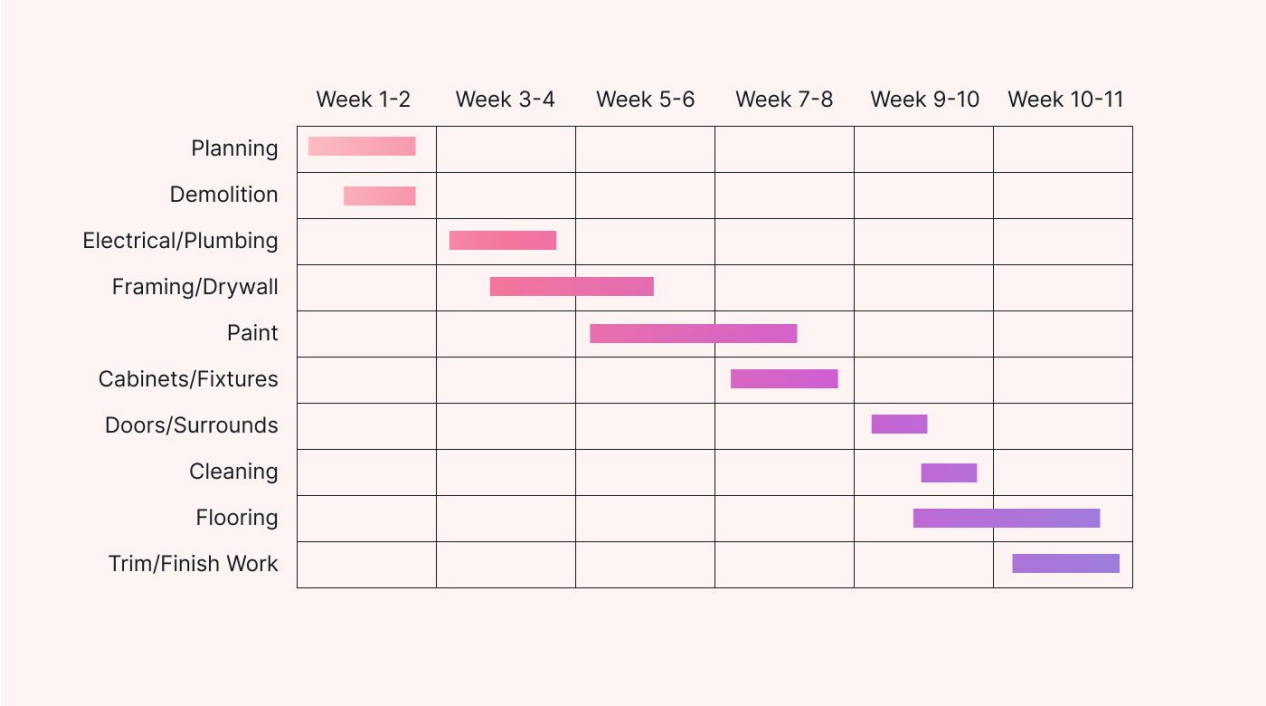
Gantt Chart

- What the various activities are
- When each activity begins and ends
- How long each activity is scheduled to last
- Where activities overlap with other activities, and by how much
- The start and end date of the whole project

For Module 9 Lab: Creating a Gantt chart using Excel, consider using the following website or any other websites.

- https://www.onlinegantt.com/#/gantt

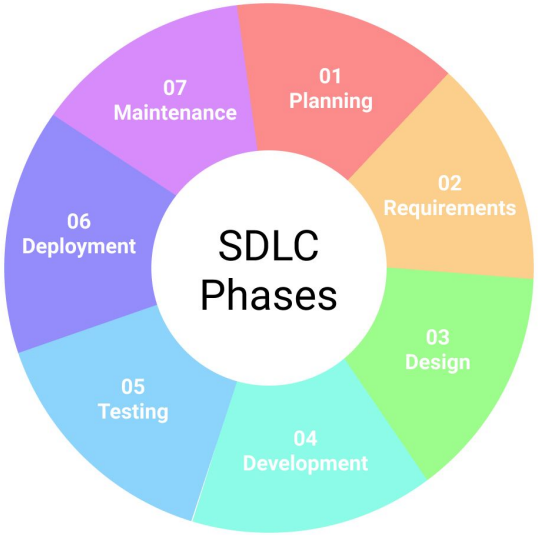
Gantt Chart


Week 1-2
Week 3-4
Week 5-6
Week 7-8
Week 9-10
Week 10-11
Planning
Demolition
Electrical/Plumbing
Framing/Drywall
Paint
Cabinets/Fixtures
Doors/Surrounds
Cleaning
Flooring
Trim/Finish Work

What is Software Development Life Cycle (SDLC)?

- The software development lifecycle (SDLC) is the cost-effective and time-efficient process that development teams use to design and build high-quality software.
- The goal of SDLC is to minimize project risks through forward planning so that software meets customer expectations during production and beyond.
- This methodology outlines a series of steps that divide the software development process into tasks you can assign, complete, and measure.

Software Development Life Cycle



1. Planning

- The planning phase typically includes tasks like cost-benefit analysis, scheduling, resource estimation, and allocation. The development team collects requirements from several stakeholders such as customers, internal and external experts, and managers to create a software requirement specification document.
- The document sets expectations and defines common goals that aid in project planning. The team estimates costs, creates a schedule, and has a detailed plan to achieve their goals.

2. Design

- In the design phase, software engineers analyze requirements and identify the best solutions to create the software.
- For example, they may consider integrating pre-existing modules, make technology choices, and identify development tools.
- They will look at how to best integrate the new software into any existing IT infrastructure the organization may have.

3. Testing

- The development team combines automation and manual testing to check the software for bugs.
- Quality analysis includes testing the software for errors and checking if it meets customer requirements.

4. Deployment

- When teams develop software, they code and test on a different copy of the software than the one that the users have access to. The software that customers use is called production, while other copies are said to be in the build environment, or testing environment.
- Having separate build and production environments ensures that customers can continue to use the software even while it is being changed or upgraded.
- The deployment phase includes several tasks to move the latest build copy to the production environment, such as packaging, environment configuration, and installation.

5. Maintenance

- In the maintenance phase, among other tasks, the team fixes bugs, resolves customer issues, and manages software changes.
- In addition, the team monitors overall system performance, security, and user experience to identify new ways to improve the existing software.

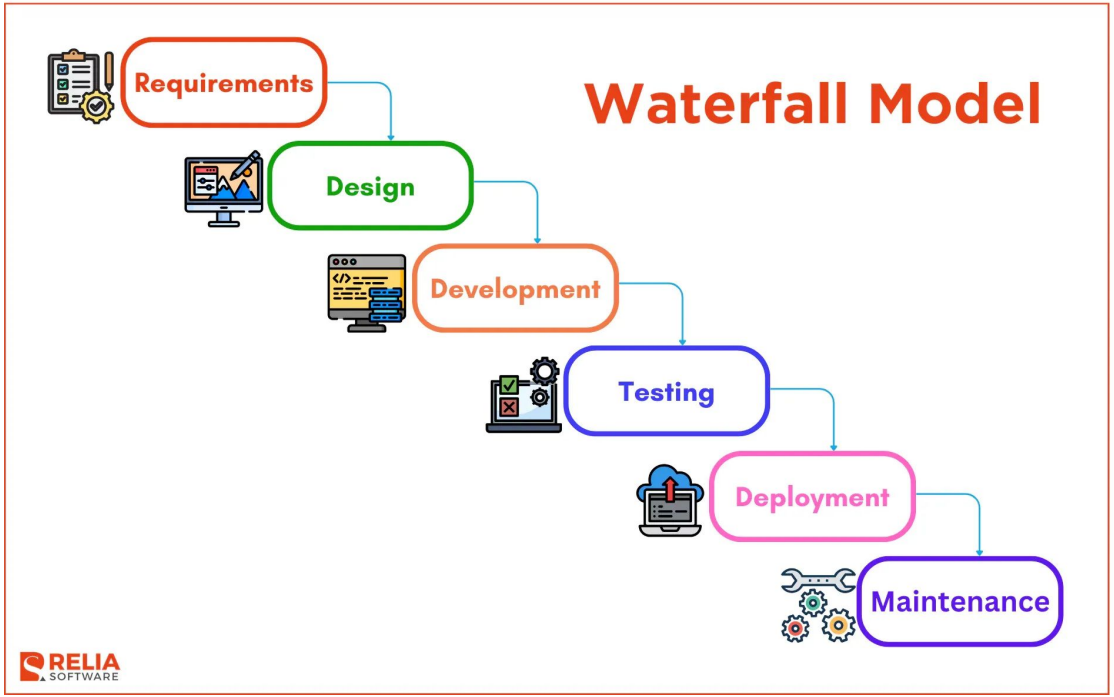
What are SDLC models?

- A software development lifecycle (SDLC) model conceptually presents SDLC in an organized fashion to help organizations implement it.
- Different models arrange the SDLC phases in varying chronological order to optimize the development cycle. We look at some popular SDLC models below.

Waterfall Model

- The Waterfall Model is the most traditional Software Development Life Cycle (SDLC) methodology. It follows a linear, sequential approach, where each phase of the development process is completed one after another before moving on to the next. This model is likened to a waterfall, cascading through various stages without turning back.

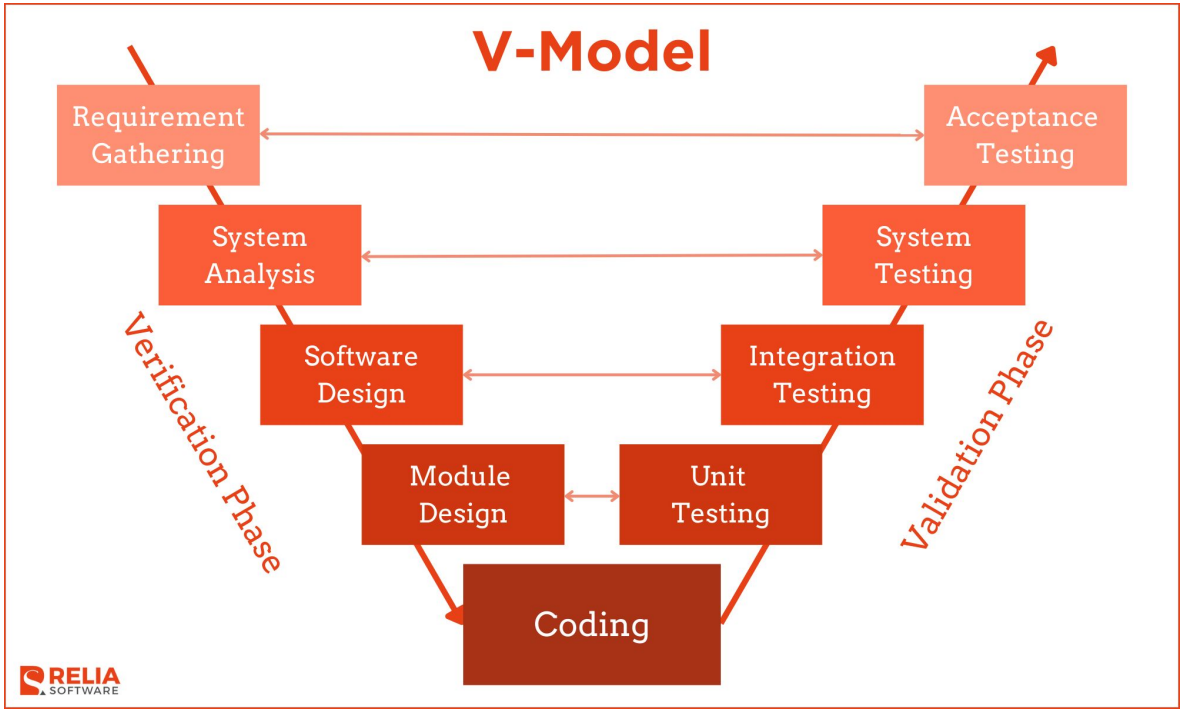
Waterfall Model



V-Model

- The V-Model, also known as the Verification and Validation model, is an extension of the Waterfall model. It emphasizes the parallel relationship between development stages and their corresponding testing phases, forming a V-shaped lifecycle. Each development phase has a directly associated testing phase, ensuring that verification and validation are integrated throughout the development process.

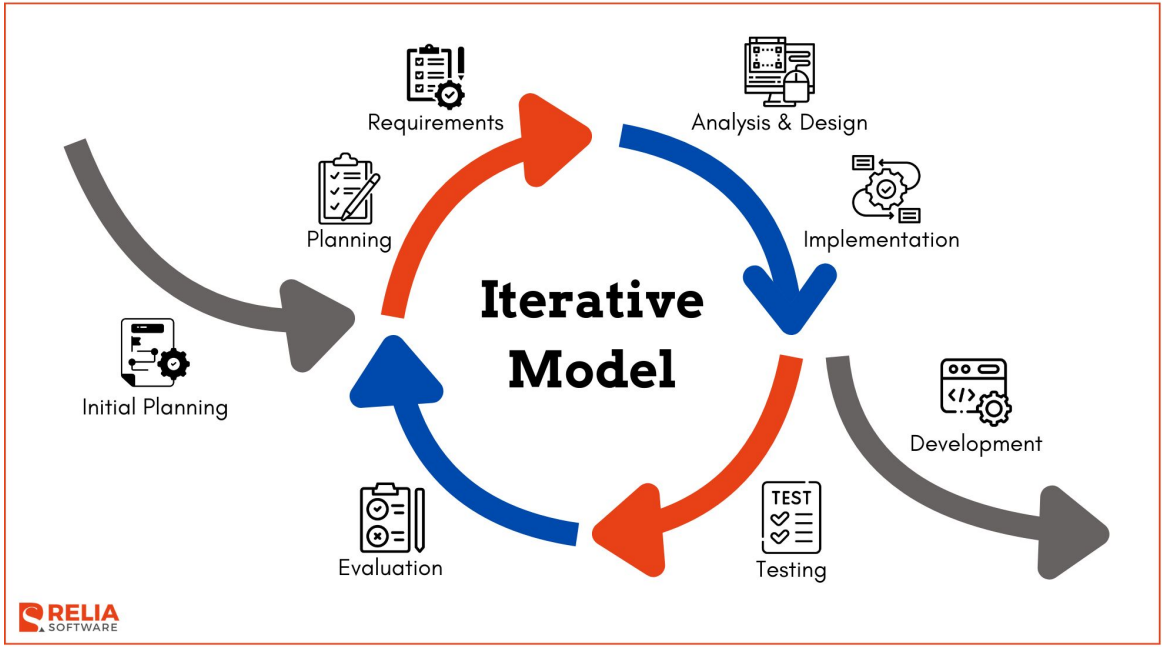
V-Model


V-Model
Requirement Gathering
Acceptance Testing
System Analysis
System Testing
Software Design
Integration Testing
Module Design
Unit Testing
Coding
Verification Phase
Validation Phase
RELIA SOFTWARE

Iterative Model

- The Iterative Model is a software development approach that builds a system incrementally through repeated cycles (iterations). Unlike the Waterfall model, where each phase must be completed before the next begins, the Iterative Model revisits phases, allowing for refinement and improvements based on feedback and evolving requirements.
- The core principle is to break down the project into smaller, manageable iterations. Each iteration typically involves activities like planning, requirements definition, design, development, testing, and deployment. User feedback is then gathered and incorporated into the following iteration.

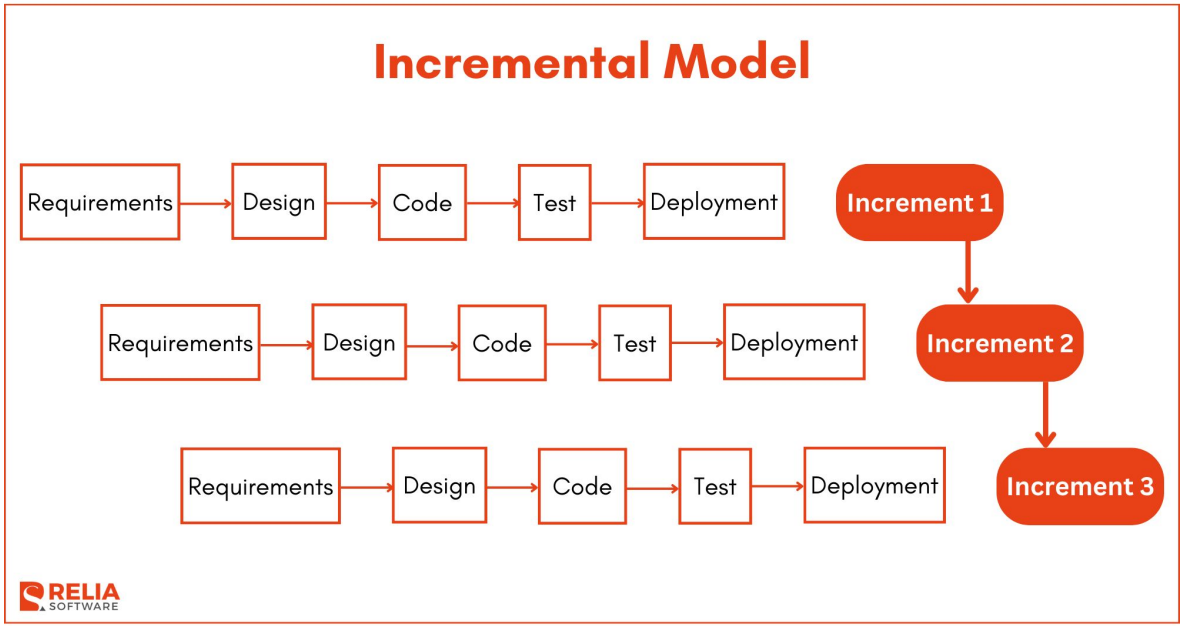
Iterative Model



Incremental Model

- The Incremental Model is a software development approach that divides the system into smaller, manageable increments or parts, which are developed and delivered in successive cycles. This model allows for partial deployment of the system and early delivery of some functional components, providing users with usable software early in the development process.
- This model shares some similarities with the Iterative Model. However, the difference lies in their delivery focus: while the Iterative Model focuses on continuous improvement of the whole product, the Incremental Model focuses on progressively adding functionality until the entire system is complete.

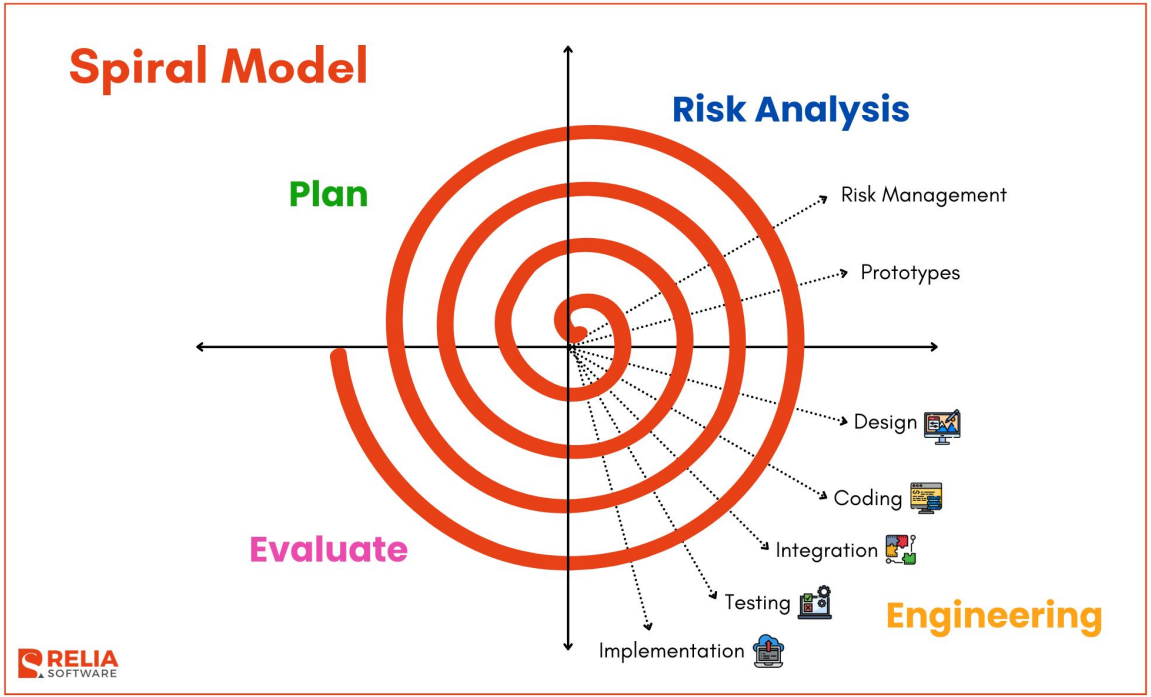
Incremental Model



Spiral Model

- The spiral model combines the iterative model's small repeated cycles with the waterfall model’s linear sequential flow to prioritize risk analysis.
- You can use the spiral model to ensure software's gradual release and improvement by building prototypes at each phase.
- The spiral model is suitable for large and complex projects that require frequent changes. However, it can be expensive for smaller projects with a limited scope.

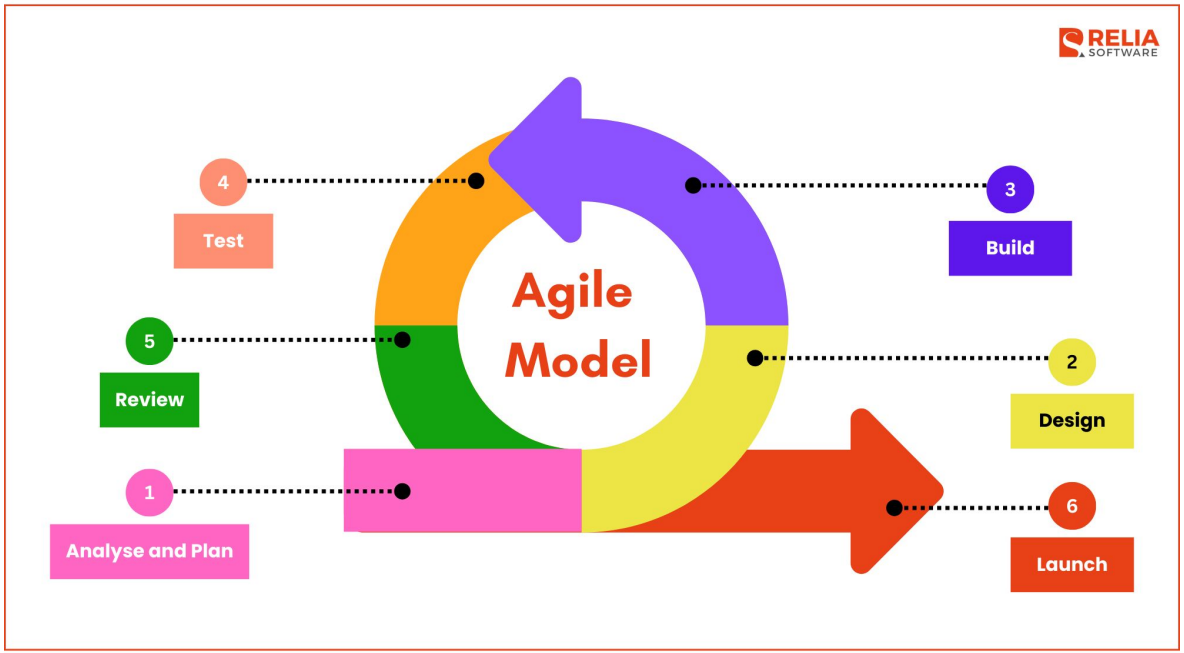
Spiral Model



Agile Model

- The agile model arranges the SDLC phases into several development cycles.
- The team iterates through the phases rapidly, delivering only small, incremental software changes in each cycle.
- They continuously evaluate requirements, plans, and results so that they can respond quickly to change. The agile model is both iterative and incremental, making it more efficient than other process models.

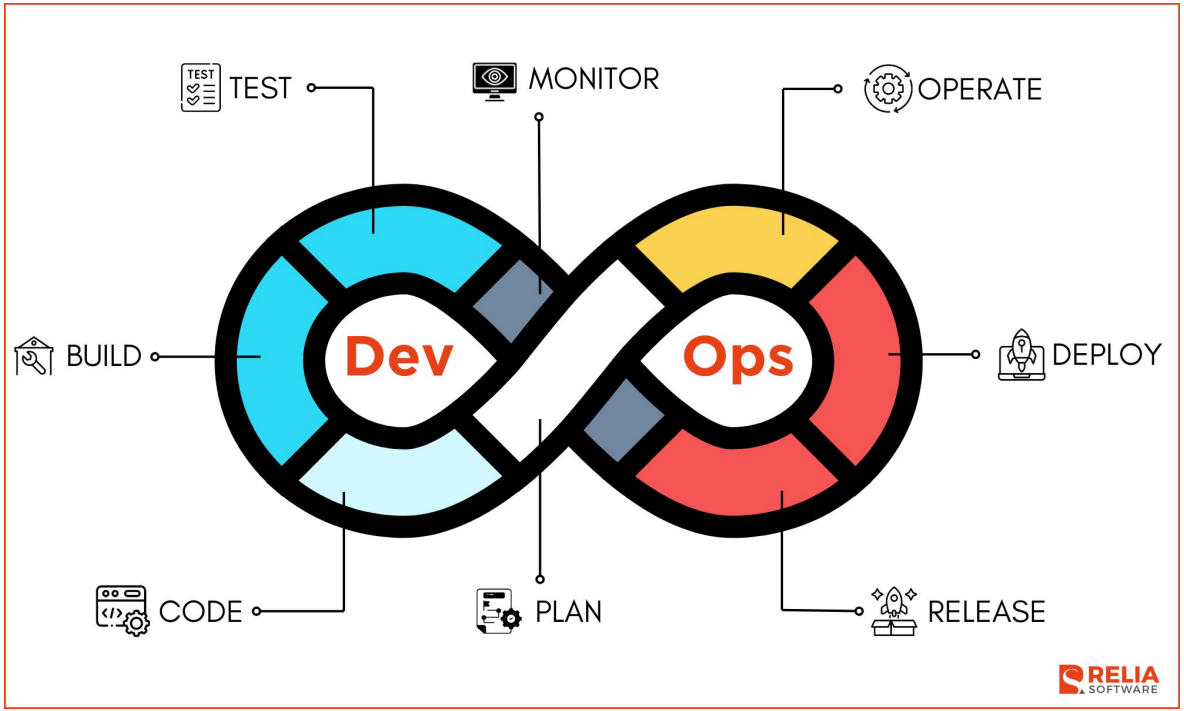
Agile Model


RELIA SOFTWARE
Agile Model
1
Analyse and Plan
2
Design
3
Build
4
Test
5
Review
6
Launch

DevOps Model

- The DevOps Model is a software development methodology that integrates development (Dev) and operations (Ops) teams to improve collaboration, efficiency, and the continuous delivery of high-quality software. It emphasizes automation, continuous integration and continuous delivery (CI/CD), and the use of shared tools and processes to streamline the development lifecycle.

DevOps Model



Factor	Waterfall Model	V-Model	Iterative Model	Incremental Model	Spiral Model	Agile Model	DevOps Model
Approach	Linear and Sequential	Linear with Verification and Validation	Repeated Increments	Successive Increments	Risk-Driven Cycles	Iterative and Incremental	Integration of Development and Operation
Risk Management	Low	Medium	Medium	Medium	Very High	Medium	High
Development Speed	Slow	Slow	Moderate	Moderate	Moderate	Fast	Fast
Testing	Post-development	Throughout Development	Continuous	Continuous	Continuous	Continuous	Continuous
Best Suited For	Well-defined Requirement	Well-defined Requirement	Evolving Requirement	Complex Projects	Large and Complex Projects	Dynamic and Changing Requirement	Continuous Deployment and Rapid Iteration