

# Introduction to Software Development

---

INTRODUCTION, SDLC, ROADMAP, COURSES

# WHAT IS SOFTWARE DEVELOPMENT?

# SOFTWARE DEVELOPMENT

**Software Development** is defined as the process of designing, creating, testing, and maintaining computer programs and applications.

This diverse field combines creativity, engineering expertise, and problem-solving abilities to produce software that satisfies particular requirements and goals.

# What is **Software Development?**



# SOFTWARE DEVELOPMENT

This diverse field combines creativity, engineering expertise, and problem-solving abilities to produce software that satisfies particular requirements and goals.

# SOFTWARE DEVELOPMENT

**Software developers**, also known as **programmers or coders**, use a variety of programming languages and tools to create solutions for end-users or businesses.

# TYPES OF SOFTWARES

# SYSTEM SOFTWARE

**System software** is software that directly operates computer hardware and provides basic functionality to users as well as other software for it to run smoothly.



# APPLICATION SOFTWARE

**Application software** is a software that is designed for end-user to complete a specific task. It is a product or program that is only intended to meet the needs of end users.

# PROGRAMMING SOFTWARE

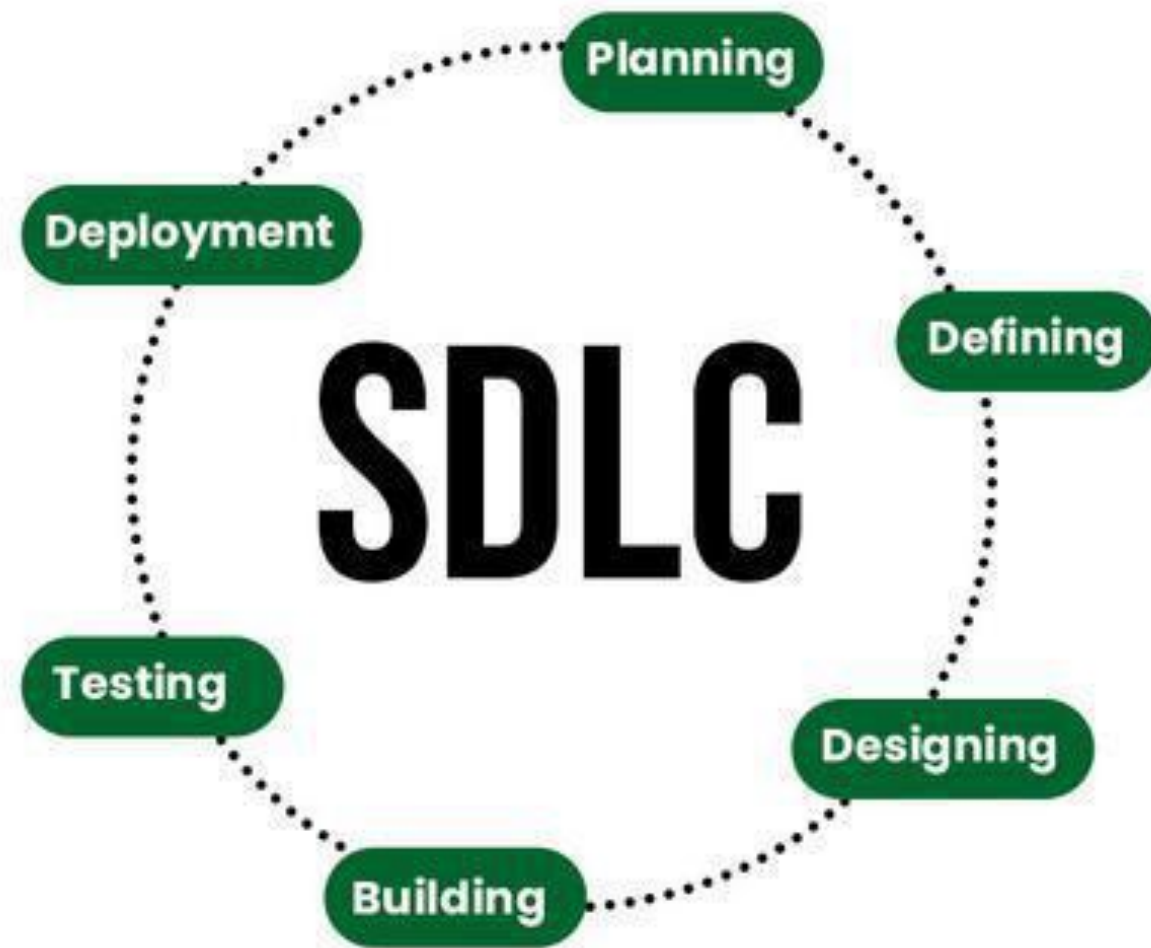
**Programming software** is a software that is designed for programmers to develop program. It consist of code editor, compiler, interpreter, debugger etc.

# STEPS OF SOFTWARE DEVELOPMENT

Software development is a well-structured process with several key stages.

# STEPS OF SOFTWARE DEVELOPMENT

**System software** is software that directly operates computer hardware and provides basic functionality to users as well as other software for it to run smoothly.



# REQUIREMENT ANALYSIS

The first step in software development is understanding the requirements and based on that requirement analysis happen. This stage involves identifying the needs, objectives, and constraints of the project. The goal is to define what the software should do and what problems it will solve.

# DESIGN

In the design phase, the software's architecture and user interface are developed. This step defines how the software will work and how users will interact with it. Design includes creating wireframes, prototypes, and system architecture diagrams.

# DESIGN

After completing the architectural design phase, developers move on to creating detailed designs for each component of the system. This includes designing not only the user interface but also encompassing databases and APIs. The intricate decisions made in these detailed designs provide valuable guidance throughout the coding phase.



# IMPLEMENTATION

The most important phase of the Software Development is the implementation phase, which comes after the design phase. This phase will see the implementation of the design phase's output.

# IMPLEMENTATION

All of the planning done in the planning phase and the designing done in the designing phase are implemented in this phase. Physical source code is created and deployed in the real world during this phase.

# TESTING

Developers utilize unit tests to evaluate small code components, such as functions or methods. These tests play a crucial role in identifying and resolving bugs within isolated elements.

# TESTING

Integration testing evaluates the smooth functioning of various software components. Its purpose is to ensure seamless interactions between modules and efficient data transfer among them, resulting in a robust system.

# TESTING

In order to ensure that the software meets all the specified requirements, system testing evaluates it as a whole. This comprehensive evaluation includes functional, performance, security, and other necessary types of testing.

# TESTING

User Acceptance Testing (UAT) occurs during the phase where end-users or clients validate the software to ensure it meets their requirements. Identified issues or discrepancies are promptly addressed before proceeding with deployment.

# DEPLOYMENT

Before deployment, the development team configures the target environment, whether it's on-premises servers, cloud-based infrastructure, or end-user devices. This may involve setting up servers, databases, and configuring software dependencies.

# DEPLOYMENT

Developers carefully plan the process of deploying software, which includes aspects such as data migration strategies, software installation procedures, and contingency measures for unexpected issues.



# DEPLOYMENT

The software is deployed to end-users or production environments. Ongoing monitoring is critical for quickly identifying and addressing any issues that may arise following the deployment.

## **MAINTENANCE AND UPDATES**

Once the software has been deployed, it is common for issues and bugs to arise. The dedicated team of developer's actively works on identifying, fixing, and thoroughly testing these problems. Regular updates are provided to address any necessary improvements or changes that may arise

# **MAINTENANCE AND UPDATES**

Feature enhancements are made to the software as user needs evolve or new requirements arise. Developers consistently implement new features and improvements in response to these changes.

# **MAINTENANCE AND UPDATES**

Regular security updates are crucial to address vulnerabilities and protect the software from cyber threats.

# DOCUMENTATION

The software developer provides user guides, manuals, and online help documentation to assist end-users effectively navigate its features.

# DOCUMENTATION

Developers are responsible for creating technical documentation that outlines the architecture, code structure, and APIs of a system. This documentation is crucial in helping future developers comprehend and maintain the software.

# FEATURES OF SOFTWARE DEVELOPMENT

- Collaborative Nature
- Continuous Learning
- Problem Solving
- Creativity
- Quality Assurance



# **COLLABORATIVE NATURE**

Software development is a collaborative process that involves a diverse group of professionals, including developers, designers, project managers, and stakeholders. Software project success is heavily dependent on effective communication and seamless teamwork.

# **COLLABORATIVE LEARNING**

In Software Development it's super important to keep learning because things are always changing. New ways of writing code, tools, and technologies are always popping up. To do well and keep up, programmers need to keep on learning and getting better at what they do. It's like an ongoing adventure of picking up new skills to stay on top of the game.

# PROBLEM-SOLVING

Developers play a crucial role as problem solvers. They actively identify and address issues, craft innovative solutions, and optimize code to achieve the desired outcomes. Problem-solving skills lie at the heart of the software development process.

# CREATIVITY

When Developers making computer programs, it's not just about following rules . There's also room for being creative. Coding needs a lot of attention to detail and clear thinking, but it's also a chance to let developers imagination run wild.

# QUALITY ASSURANCE

In development, ensuring the quality and reliability of the software is a crucial aspect. To ensure exceptional results, the development cycle includes stringent testing and quality assurance procedures.

WHY IS SOFTWARE  
DEVELOPMENT  
IMPORTANT?

- Enabling Technological Innovation
- Improved Efficiency
- Adapting to Changing Needs
- Global Reach

# ENABLING TECHNOLOGICAL INNOVATION

Software development plays a crucial role in technological advancements. Software developers are responsible for creating innovative smartphone applications, designing websites, or developing complex enterprise software.



## **IMPROVED EFFICIENCY**

In various industries, software development plays a crucial role in automating tasks and processes. This automation leads to enhanced efficiency. Consider the business sector as an example. It utilizes software applications to streamline operations, effectively manage resources, and facilitate informed decision-making processes.

# **ADAPTING TO CHANGING NEEDS**

Software development offers the necessary flexibility and adaptability, allowing developers to continually update and modify software in response to evolving user needs, regulatory requirements, and business demands. This ability to adapt holds paramount importance in effectively navigating the rapid changes of the digital domain.

## **GLOBAL REACH**

The internet has revolutionized connectivity by bridging gaps across continents. With the aid of software applications, both businesses and individuals can effortlessly tap into a worldwide audience, shattering geographical boundaries and unlocking boundless market potential.

# JOBS THAT REQUIRE SOFTWARE DEVELOPMENT

- Software Developer / Programmer
- Front-End Developer
- Back-End Developer
- DevOps Engineer
- Quality Assurance(QA) Engineer
- Software Architect
- Product Manager
- Data Scientist/Engineer
- Cybersecurity Analyst

# **Software Developer/Programmer**

Software developers, also known as programmers, have the important task of writing code and developing applications to meet project requirements. They specialize in various areas such as web development, mobile app development, or back-end systems development. Their role involves ensuring that the software functions effectively and fulfills its intended purpose.

# FRONT-END DEVELOPER

In the field of web development, a Front-End Developer is responsible for crafting the visual interface and enhancing user experience on websites and applications. Their expertise lies in utilizing HTML, CSS, and JavaScript to design and implement visually compelling elements within software.

# **BACK-END DEVELOPER**

In the field of software development, there exists a crucial role known as the Back-End Developer. These talented individuals possess expertise in server-side programming, managing databases, and ensuring efficient server functionality. It is their responsibility to construct the underlying infrastructure



# DevOps Engineer

The DevOps Engineer plays a crucial role in bridging the gap between development and IT operations. They facilitate a seamless process by automating deployment, testing, and monitoring of software. Their responsibilities encompass ensuring efficient development and deployment procedures.

# **QUALITY ASSURANCE (QA) ENGINEER**

The QA engineer is responsible for testing and ensuring the quality and functionality of software. They carefully design test cases, execute tests, and diligently report any defects to the development team.

# SOFTWARE ARCHITECT

The software architect is responsible for designing the overall structure and system of a software project. They make important high-level design decisions and establish the project's technical direction.

# PRODUCT MANAGER

A Product Manager oversees the entire development process, from gathering requirements to deployment. They are responsible for defining project goals, prioritizing features, and ensuring that the final product aligns with business objectives.

# **DATA SCIENTIST/ENGINEER**

Data scientists and engineers are experts in the manipulation and analysis of data. Their focus lies in creating data-driven applications and algorithms that benefit both businesses and research endeavors.

# **CYBERSECURITY ANALYST**

With the growing importance of cybersecurity, analysts in this field focus on securing software and systems against cyber threats and vulnerabilities.