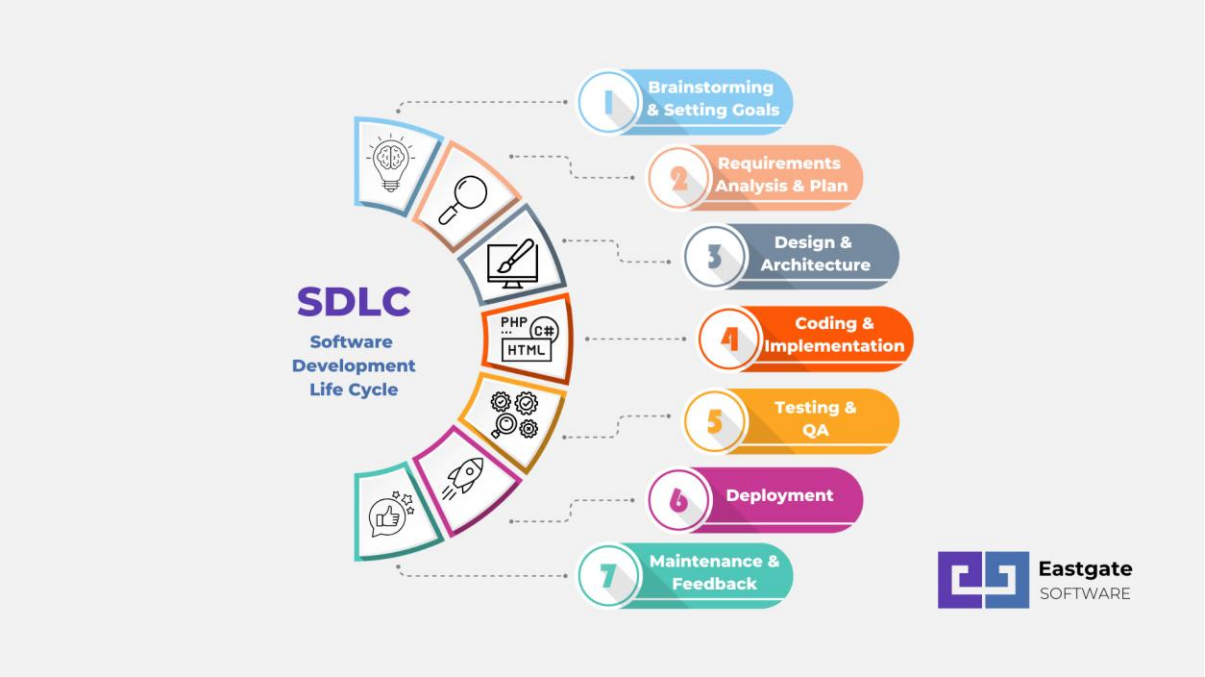
SDLC — Software Development Life Cycle



1. Requirement Document

This is the first and most important step.

- Requirements are collected in **text form** along with **wireframes**.
- The document contains **abstract details**, basic flow, and what the client wants.
- Requirements are usually given by the **PO (Product Owner)** or **Client**.

Goal: understand *what the system should do*.

---

2. Technical Document

Prepared by the **Technical Lead (TL)** or **Software Architect**.

- It includes **multiple reviews** before approval.
- Defines the **complete technical structure** of the project:
    - Number of classes
    - Interfaces
    - Inheritance details
    - Components
    - Modules

Goal: decide *how the system will be built*.

---

3. Coding Phase

Development begins after the technical document approval.

- Coding is usually started by **SSE (Senior Software Engineer)** and **SE (Software Engineer)**.
- Follows coding standards, best practices, and modular design.

Goal: convert requirements into working code.

---

4. Testing

Testing is done by the **QA (Quality Assurance)** team.

- Performed on **QA Server**.
- QA engineers check functionality, usability, and performance.
- Bugs are **raised** and sent back to developers.

Goal: ensure the software works correctly.

---

5. Staging Environment

Before going live, the build is deployed to **Staging** for final checks.

- **Regression Testing** – ensures old features still work.
- **Load Testing** – checks performance under heavy usage.
- **Automation Test Cases** – automated scripts to test repeated scenarios.

Related Environments:

- **UAT Environment (User Acceptance Testing)**
    - Client/business team tests and confirms that the product meets requirements.
- **PPE – Pre-Production Environment**
    - Final environment before going live; mirror of production.
- **PE – Production Environment (GO LIVE)**
    - Application becomes available to end users.

Goal: prepare software for real-world usage.

---

6. Maintenance & Bug Fixing

After going live:

- Users may report bugs.
- Developers fix issues and release updates or patches.

- Continuous monitoring and improvement.

Goal: maintain stability, performance, and user satisfaction.
