**Understanding Job Roles: Developer, Tester, and Client**

1. **Developer:** A developer is responsible for writing and maintaining the source code of an application. They implement business logic, fix bugs, and ensure that the software meets functional and non-functional requirements. Developers work in the development environment to build and test their code before handing it over to testers.
2. **Tester:** A tester is responsible for verifying the functionality, performance, and security of the application. They ensure that the application behaves as expected under different conditions and report defects to developers. Testers work in the test environment, where they execute test cases and validate the stability of the application.
3. **Client:** The client is the end-user or the business stakeholder who provides requirements and feedback on the application. They use the User Acceptance Testing (UAT) environment to validate that the software meets business needs before it goes into production.

**Why We Need Multiple Environments**

Having multiple environments ensures smooth development, testing, and deployment processes. Each environment serves a distinct purpose:

* **Development (Dev) Environment:** Used by developers to write, debug, and unit-test code.
* **Testing (Test) Environment:** Used by testers to validate software functionality and performance.
* **User Acceptance Testing (UAT) Environment:** Used by clients to validate that the software meets business requirements before going live.

This separation helps in identifying and fixing issues early in the software development lifecycle (SDLC), ensuring a stable production release.

**Source Code**

Source code is the set of human-readable instructions written by developers to define the behavior of an application. It is usually written in programming languages like Java, Python, or JavaScript and is maintained in a version control system.

Example of Java source code for adding two numbers:

public class Addition {

public static int add(int a, int b) {

return a + b;

}

public static void main(String[] args) {

int result = add(5, 10);

System.out.println("Sum: " + result);

}

}

**Business Logic vs Non-Business Logic**

* **Business Logic:** The part of the application that implements the core functionality and rules of the business domain. For example, calculating the interest on a loan in a banking application.
* **Non-Business Logic:** The supporting functions such as logging, authentication, and data validation that do not directly contribute to the business rules but are essential for application functionality.

Example:

* Business logic: calculateInterest(principal, rate, time)
* Non-business logic: logTransactionDetails(transactionID)

**Build, Release, and Deploy**

1. **Build:** The process of converting source code into an executable format (e.g., JAR, WAR, or EXE). This involves compiling the code, resolving dependencies, and packaging the application.
2. **Release:** The process of making a particular version of the application available for testing or deployment. A release may include new features, bug fixes, or enhancements.
3. **Deploy:** The process of moving the built application to an environment where it can be tested or used. Deployment can be done in multiple environments, such as Test, UAT, or Production.