In my opinion the most important principles are:

1. Modularity and Encapsulation: Divide code into smaller, reusable modules or classes. Using encapsulation to hide implementation details and provide clear interfaces.
2. Separation of Concerns (SoC): Separate different concerns (e.g., presentation, business logic, data access) into distinct modules or layers.Follow a design pattern like Model-View-Controller (MVC) or Model-View-ViewModel (MVVM) to maintain clear separation.
3. Consistent Coding Style: Follow a consistent coding style and naming conventions. This will make the code more readable and easier to maintain. For example, use a linter or coding standards.
4. Error Handling: Implement robust error handling and reporting.

Use exceptions or error codes to handle and report errors gracefully.

1. Scalability and Performance: Design your application to scale horizontally or vertically as needed. Optimize critical code paths for performance.

**Angular execution:**

1. Make sure you have Node.js and npm installed:

2. Make sure you have Angular CLI Installed: **npm install -g @angular/cli**

3. The environment variables are in the environment folder.

4. run **npm install**

4. In the angular folder of the project run: **ng s**, the server will run at **http://localhost:4200/**

Angular build:

1. Make sure you have Angular CLI, Node js and npm installed.

2. Environment variables are in the environment folder. To configure the url api

3. In the project folder run: **ng build --prod**

**Java-Spring Execution:**

1. Make sure you have Java installed.

2. In the project folder run: **./gradlew bootRun** or run it with the IDE.

Java Build:

Make sure you have Java installed.

2. Run **./gradlew war**

3. Then you will get the expected war file under **build/libs**.