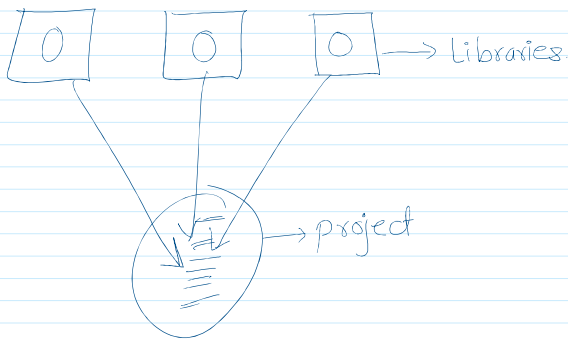
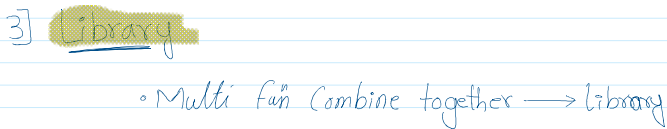
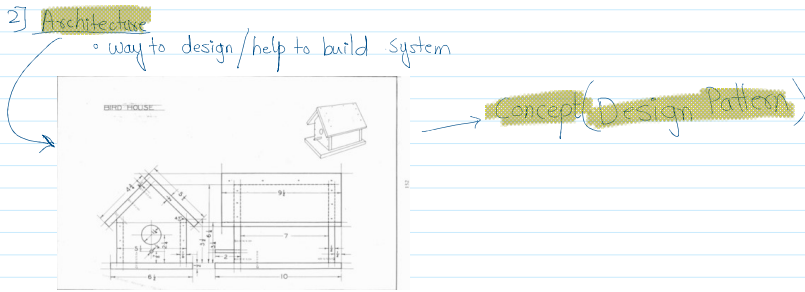
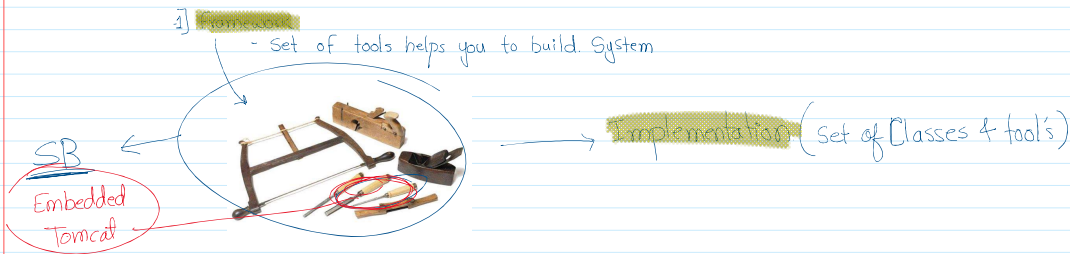
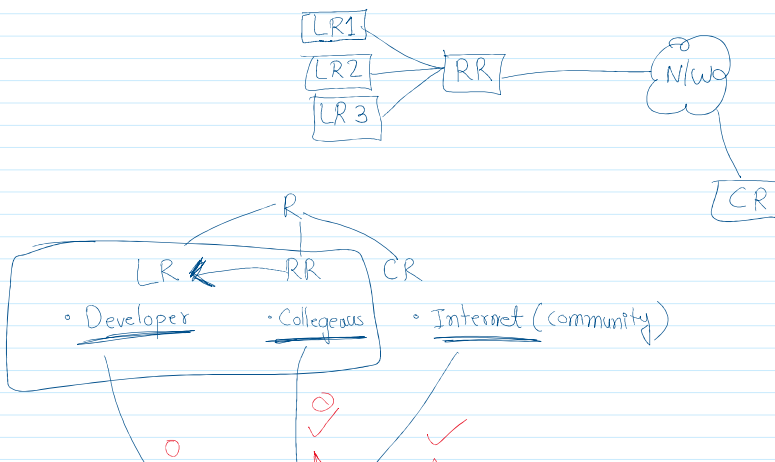
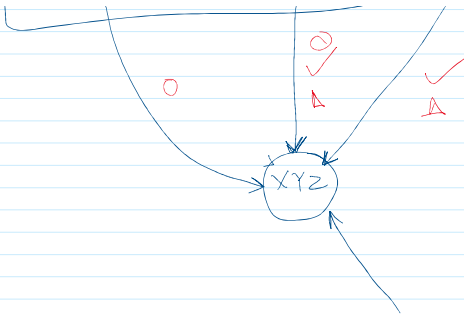


Architecture & Framework & Libraries (Dependencies)

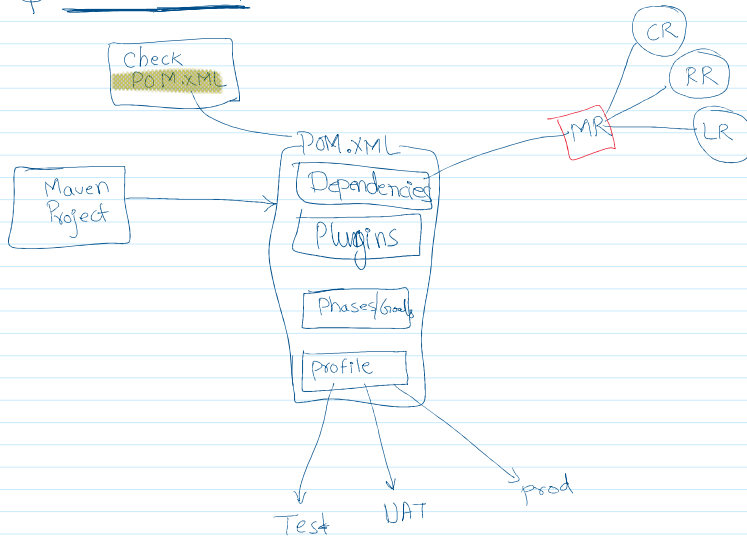


Maven Repository





\$ mvn install



Spring Boot framework

Web framework

WebApp



① Client

- Web Application
- Mobile App
 - Android
 - iOS
- IoT
- Embedded

Operations

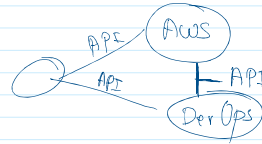
HTTP VERBS				
Read	Write	Info	Update	Del
GET	POST	HEAD	PUT	DEL

② Controller

- Engage client request
- Web REST API → AWS API Gateway



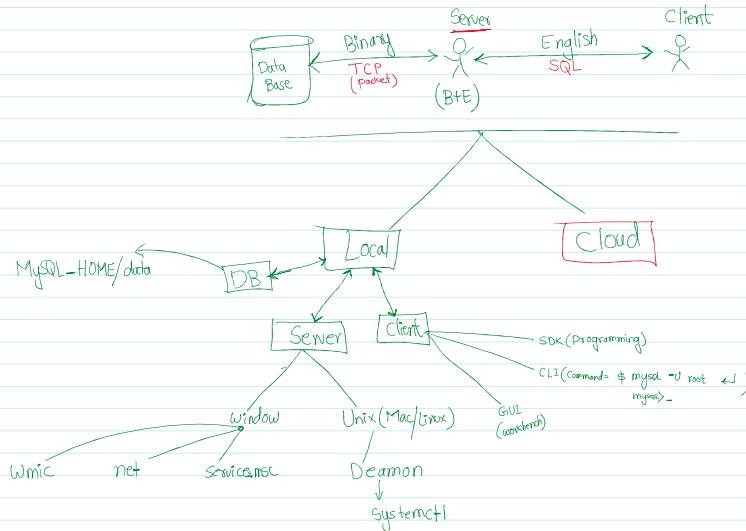
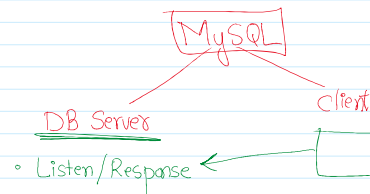
Web REST API



Base Add {
 /hi
 /login
 /register

Project

- PL = Java → check
 - DB = MySQL
 - Way = Web REST API Calls.
 - WFW = Spring Boot
- Version Require = POM.XML (1.8)
 Check Same Version = java -version 17



MySQL Database Commands

1. Connect to MySQL
 - a. \$ mysql -u root
2. Display all Databases
 - a. \$ show databases;
3. Create Database
 - a. \$ create database DBName;
4. Select Database
 - a. \$ use DBName;
5. Display all Tables
 - a. \$ show tables;
6. Read values from table
 - a. \$ select * from tableName;

Understanding Project

1. Folders Src
 2. File = POM.xml
- Apache Maven and Java Based Project

RDBMS

If you open pom.xml found spring-boot -> Uses Spring Boot Web Framework

If you found dependency as MySQL then -> MySQL as a database

```
<dependency>
<groupId>mysql</groupId>
<artifactId>mysql-connector-java</artifactId>
<version>8.0.23</version>
</dependency>
```

If you found dependency as H2 then -> In Memory H2 as a database

```
<dependency>
<groupId>com.h2database</groupId>
<artifactId>h2</artifactId>
<scope>runtime</scope>
</dependency>
```

Root-Project -> Src -> main -> resources -> application.properties OR application.yml

server.port=8090

➤ Application will run on Port 8090

➤ AWS EC2 -> Security Group -> Inbound Rule(Protocol = HTTP, Port = 8090,3306)

➤ AWS RDS -> Security Group -> Inbound Rule(Protocol = HTTP, TCP, Port = 8090,3306)

spring.datasource.url=jdbc:mysql://localhost:3306/notesDB

jdbc:mysql://localhost:3306/notesDB -> Connection String
localhost -> Database Server IP/DNS Value

notesDB -> Database Name

spring.datasource.driverClassName=com.mysql.cj.jdbc.Driver
spring.datasource.username=root
spring.datasource.password=

