

Spring Framework

1. What does Spring framework provide?

1. Application Context and dependency injection.

2. Data access

3. Spring MVC → The ability to build application using MVC.

* Application class

```
Triangle MyTriangle = new Triangle();  
MyTriangle.draw();
```

```
Circle MyCircle = new Circle();  
MyCircle.draw();
```

* Using Polymorphism.

```
Shape shape = new Triangle();  
shape.draw();
```

```
Shape shape = new Circle();  
shape.draw();
```

* Spring MVC

```
<form>
<input type="text" name="t1"><br>
<input type="text" name="t2"><br>
<input type="Submit">
</form>
```

Configure and Code our Spring MVC project projects in Just 5 Steps.

→ Front Controller - Servlet Mapping

* step 1 Configure the dispatcher servlet in web.xml

step-2 Create Spring Configuration file.

step 3 Configure view Resolver - Internal Resource view Resolver.

step 4 Create Controller

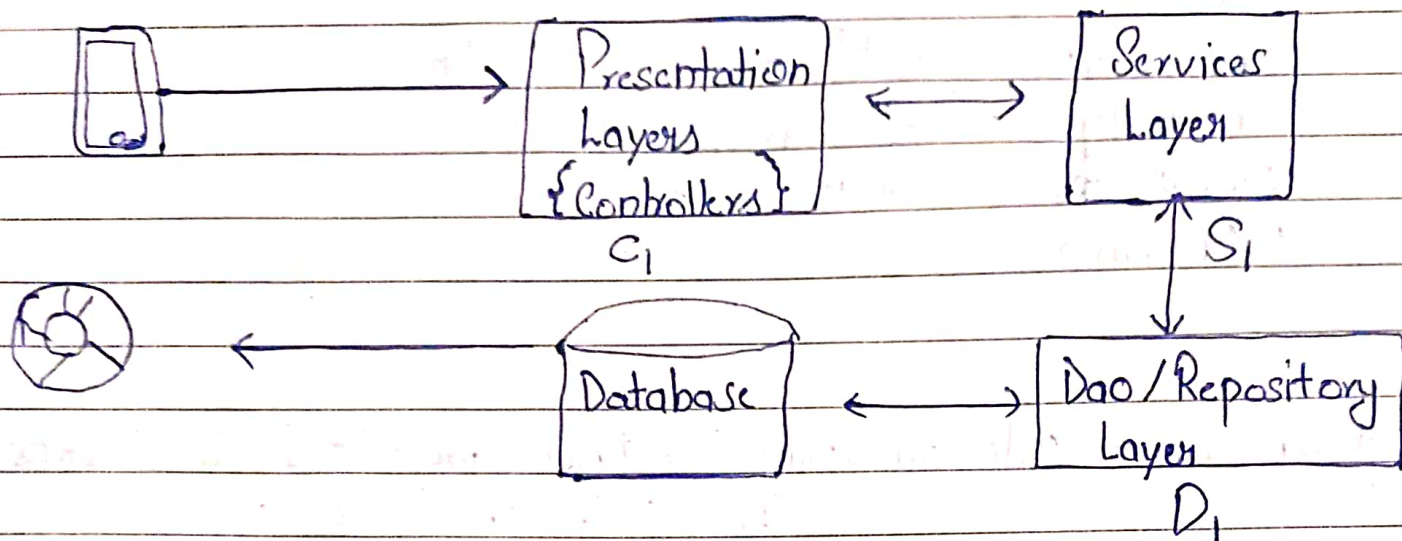
step 5 Create a View to show (Page).

Spring boot

Dependency injection →

@ Component

* Spring Boot



API URL'S for Contact

GET	/Contacts	Get all contact s
GET	/Contacts/{contactId}	Get Single course of given id in url
POST	/Courses	Add new Course
PUT	/Courses	Update the Contacts-
DELETE	/Contacts/{courseId}	Delete the contactId

* Coding Part

firstly we made a package.

Src/main/java

com.Springrest.Springrest

> ☐ SpringRestApplication.java.

⇒ down there is a pom.xml file
JPA dependency
Web dependency

Src/main/Java

✓ Com.Springrest.Springres.Controller
 → ☐ My controller.

✓ @RestController @Controller — It tells that it is a controller class. → Representation state Transfer.

```
Public class MyController {
```

```
    @GetMapping
```

```
    Public String home ( ) {
```

```
        return "Welcome to the Contact Application";
    }
```

```
// get the contacts
```

```
    @GetMapping ("/Courses")
```

```
    public List <contacts> getContacts ( )
```

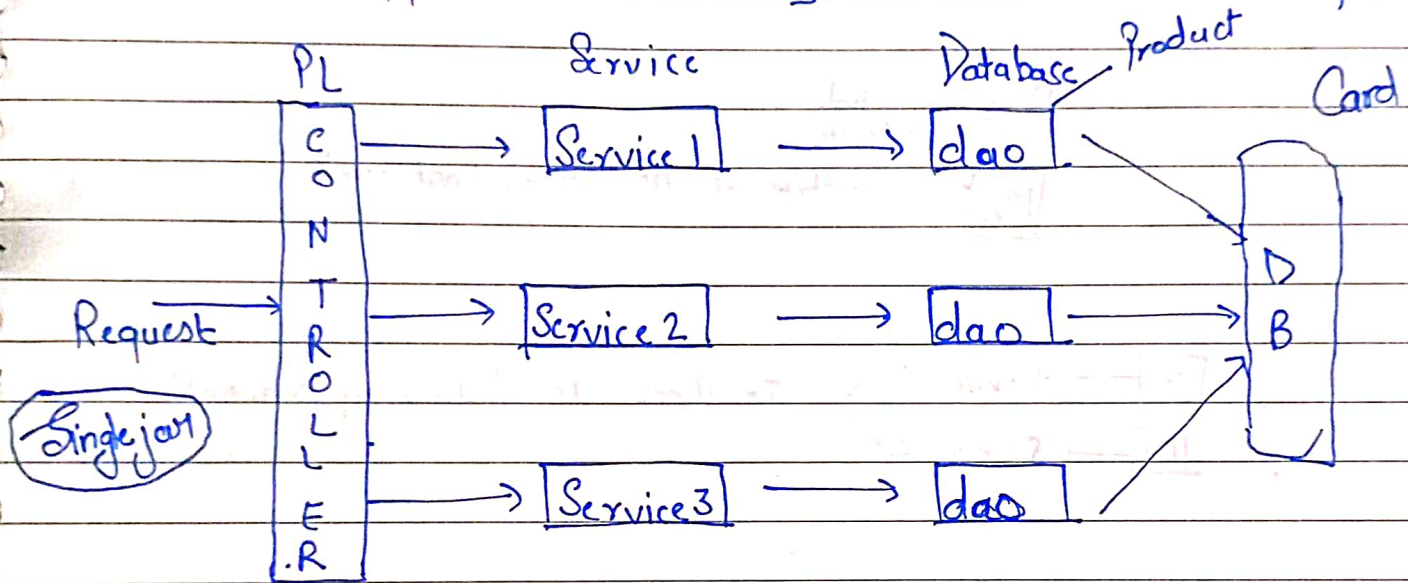
Steps to Make Spring Boot Application

1. firstly we made the Packages — 3 Packages
2. In that package made the class by model class, Controller class.
- 3.

* Microservices for Beginners By durgesh.

1. Monolithic architecture \Rightarrow
Composed all in one piece

Monolithic application has Single code base with Multiple modules

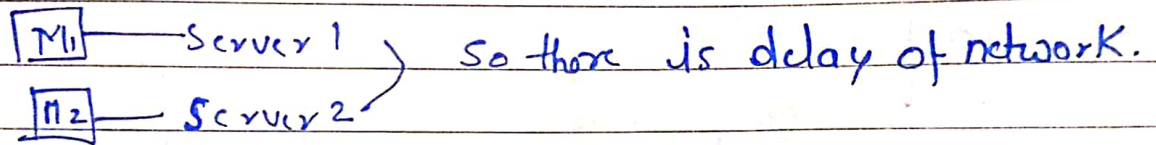
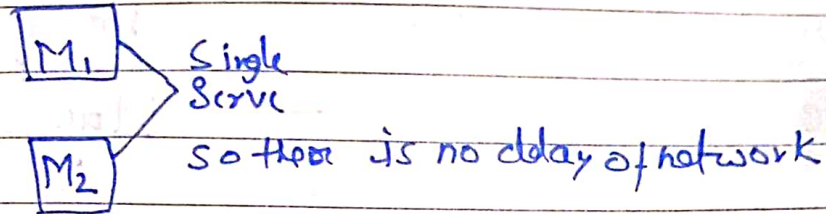


Disadvantages of Monolithic Architecture-

- As project scale, it becomes difficult to manage.
- For a single change ^{1000 modules} redeployment of whole application needed.
- Difficult to adopt new technology for single functionality.
- Single bug may down your application.

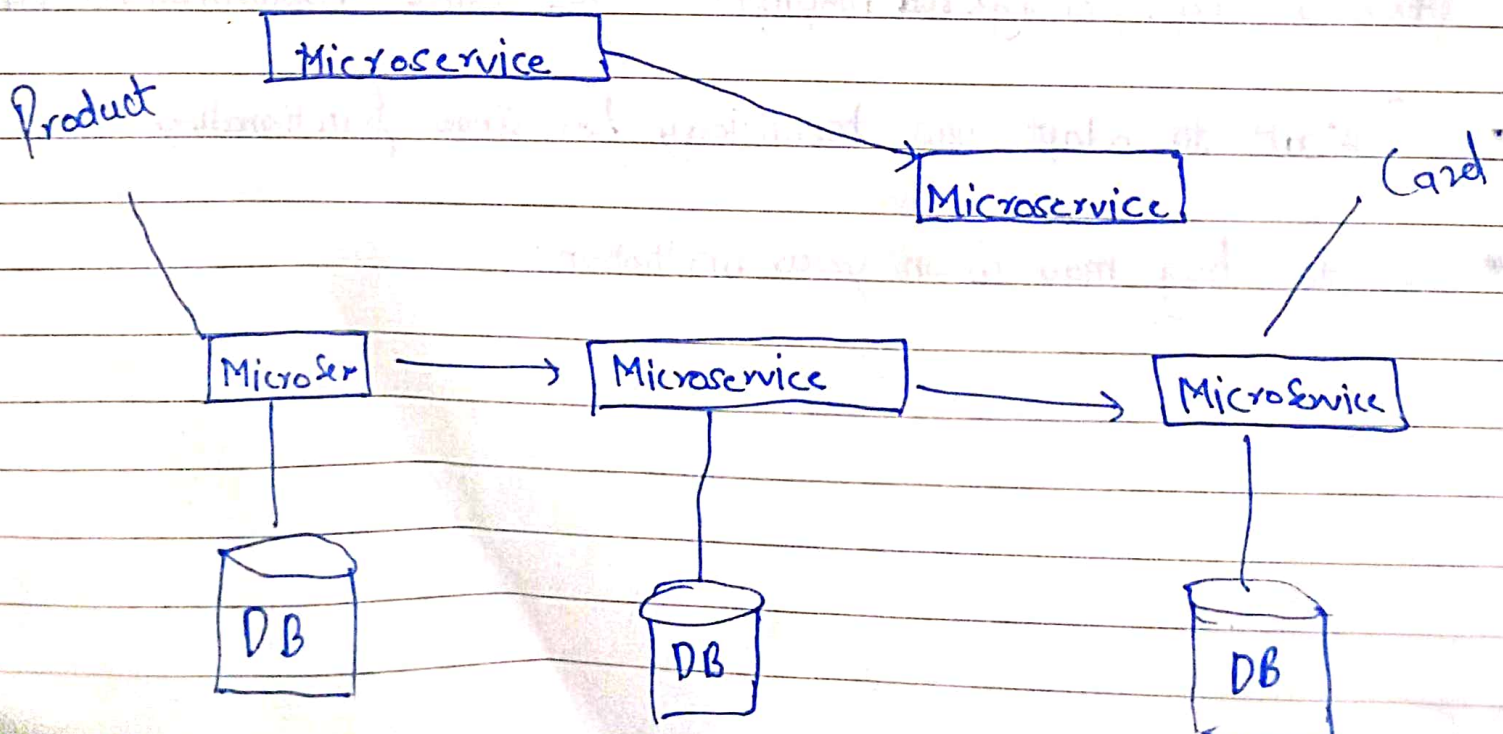
* Advantages of Monolithic Application.

- ⇒ Simple to develop.
- ⇒ Simple to build and deploy
- ⇒ Problem of network latency are relatively less



* What are Microservices?

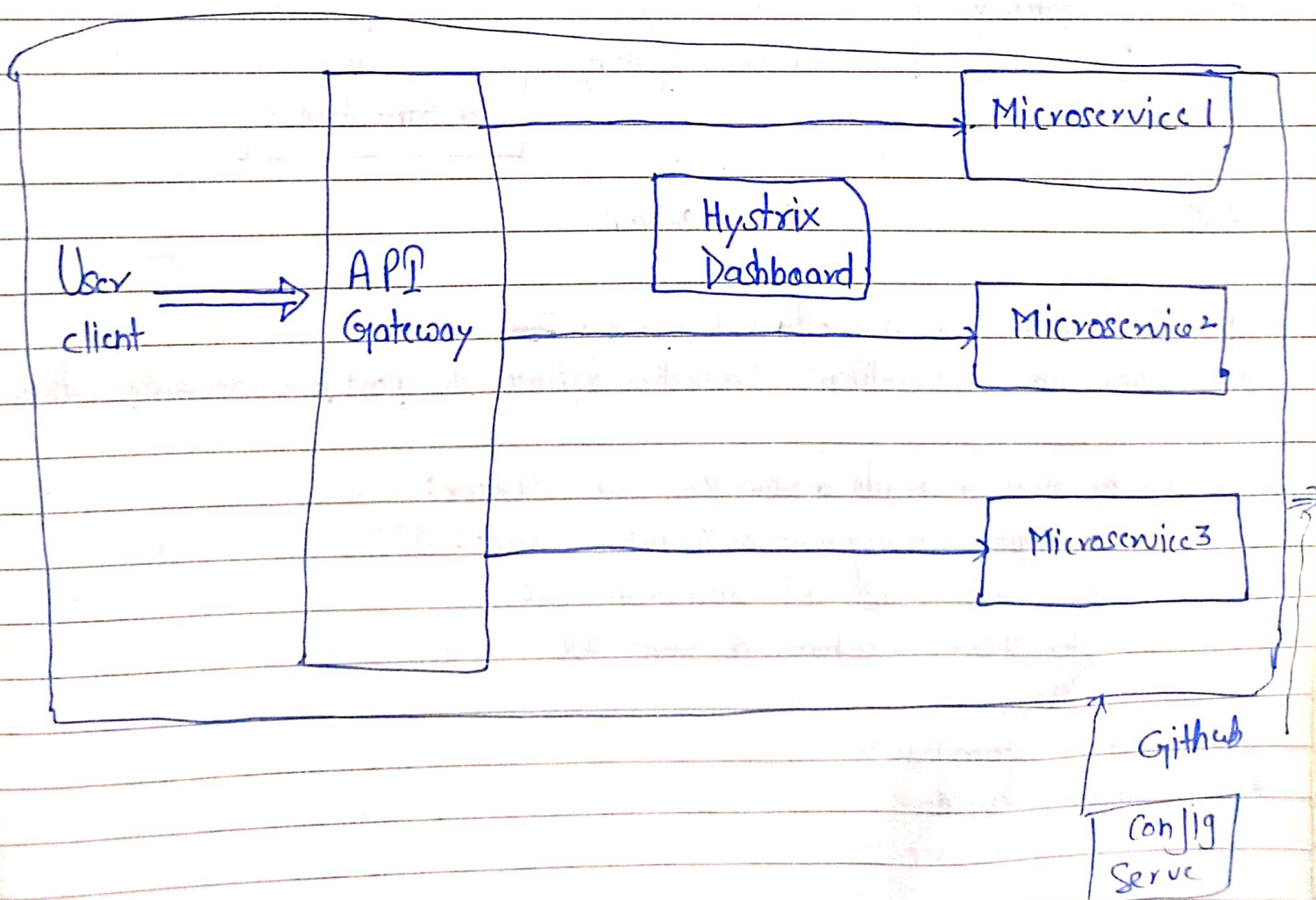
- Microservices are the Small Service that work together.
- These smaller services communicating with each other directly using light weight protocols like HTTP.



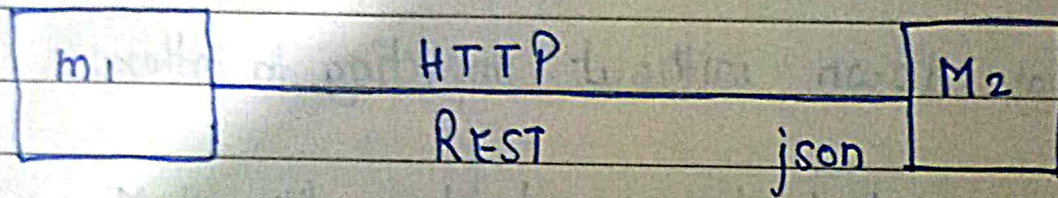
Advantages of Microservices.

- * It is possible to change or upgrade each Service individually rather than upgrading in the entire application.
- * One Service may down without impacting to others
- * Easily use different technology for building different Microservices.
- * Less dependency. loosely coupled
- * Many other advantages.

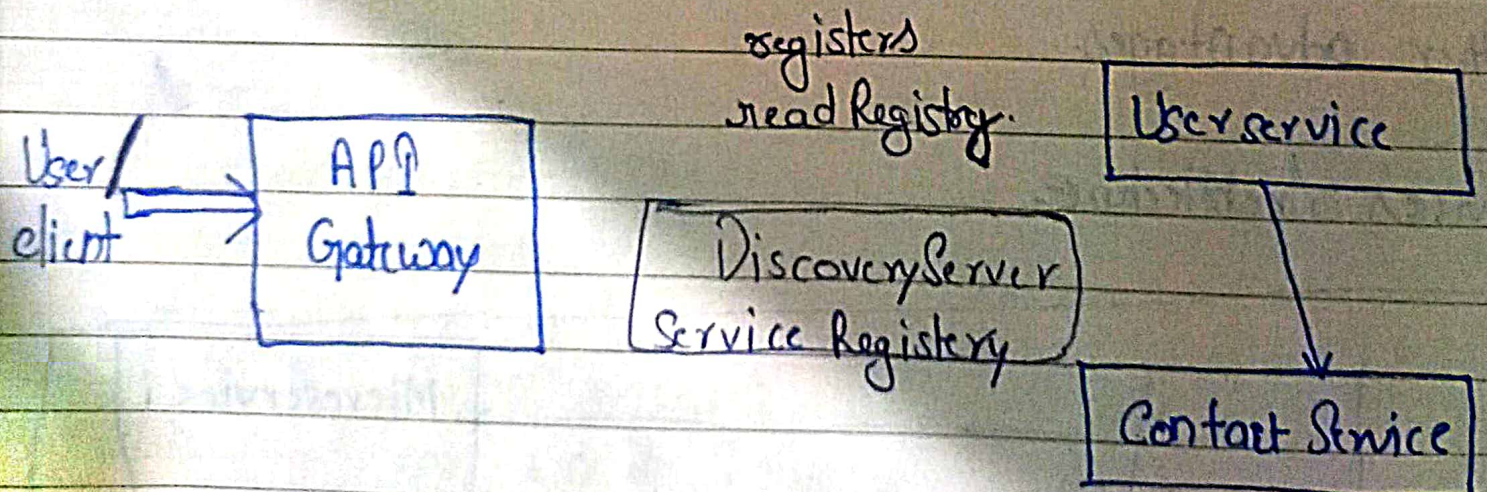
* Microservices Architecture



* How Microservice Contact Each other



* Use Case



Steps

EurekaServer

1. firstly made a project for use x 2
2. Then in application properties defined the port for user