

# Spring Projects

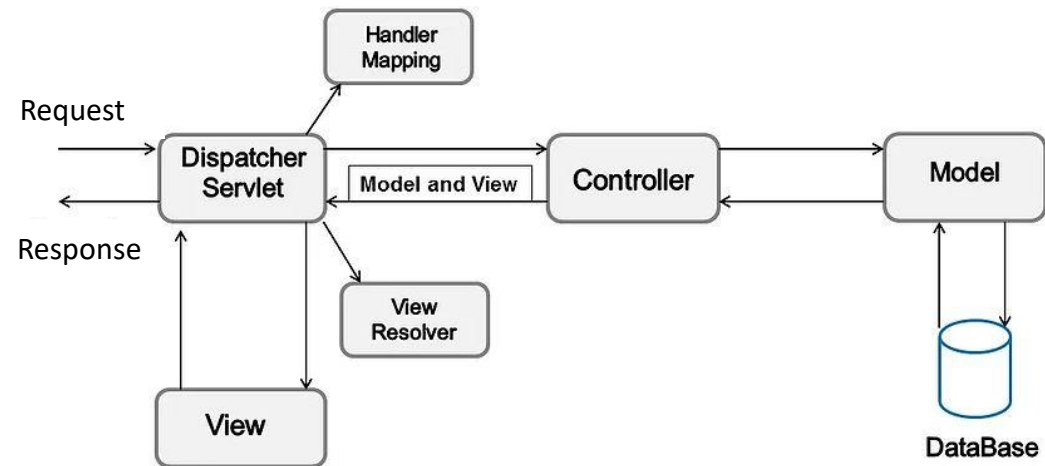
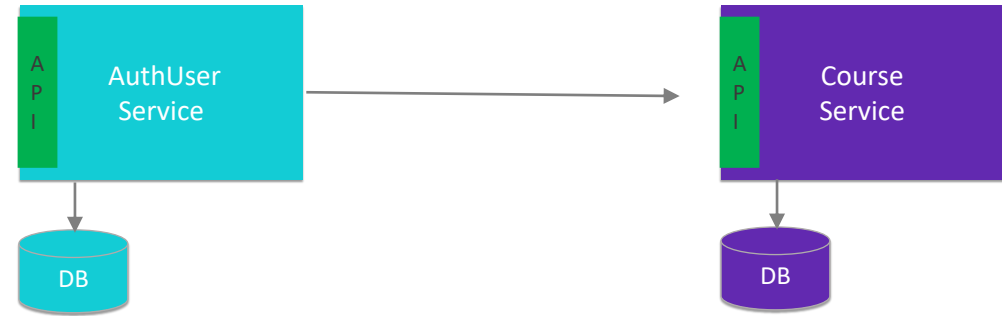
for Microservices Architecture

# Spring Web MVC

@Controller

@RestController

@RequestMapping



```
<dependency>
  <groupId>org.springframework.boot</groupId>
  <artifactId>spring-boot-starter-web</artifactId>
</dependency>
```

# Spring Validation

@Validated

@NotNull

@Size

@NotEmpty

@Email

@NotBlank

```
@NotBlank
private String name;
@NotBlank
private String description;
@email
private String email;
@Size(min = 4, max = 50)
private String username;
```

```
<dependency>
  <groupId>org.springframework.boot</groupId>
  <artifactId>spring-boot-starter-validation</artifactId>
</dependency>
```

# Spring HATEOAS

```
{
  "userId": "9743eddb-31d5-43a7-9986-e20a900d6bbd",
  "username": "michellibrito",
  "email": "contato@michellibrito.com",
  "fullName": "Michelli Brito Teste",
  "userStatus": "ACTIVE",
  "userType": "STUDENT",
  "phoneNumber": "+5511987542587",
  "cpf": "12598456985",
  "imageUrl": "teste-image-validated",
  "creationDate": "23-09-2021 21:42:26",
  "lastUpdateDate": "25-09-2021 13:02:47",
  "links": [
    {
      "rel": "self",
      "href": "http://localhost:8087/users/9743eddb-31d5-43a7-9986-e20a900d6bbd"
    }
  ]
},
```

```
public class UserModel extends RepresentationModel<UserModel> implements Serializable {

    user.add(LinkTo(methodOn(UserController.class).getOneUser(userId)).withSelfRel());
```

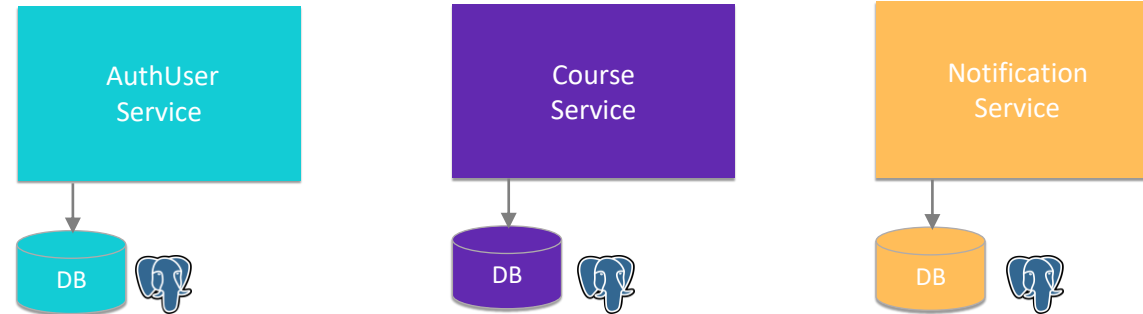
<dependency>

<groupId>org.springframework.boot</groupId>

<artifactId>spring-boot-starter-hateoas</artifactId>

</dependency>

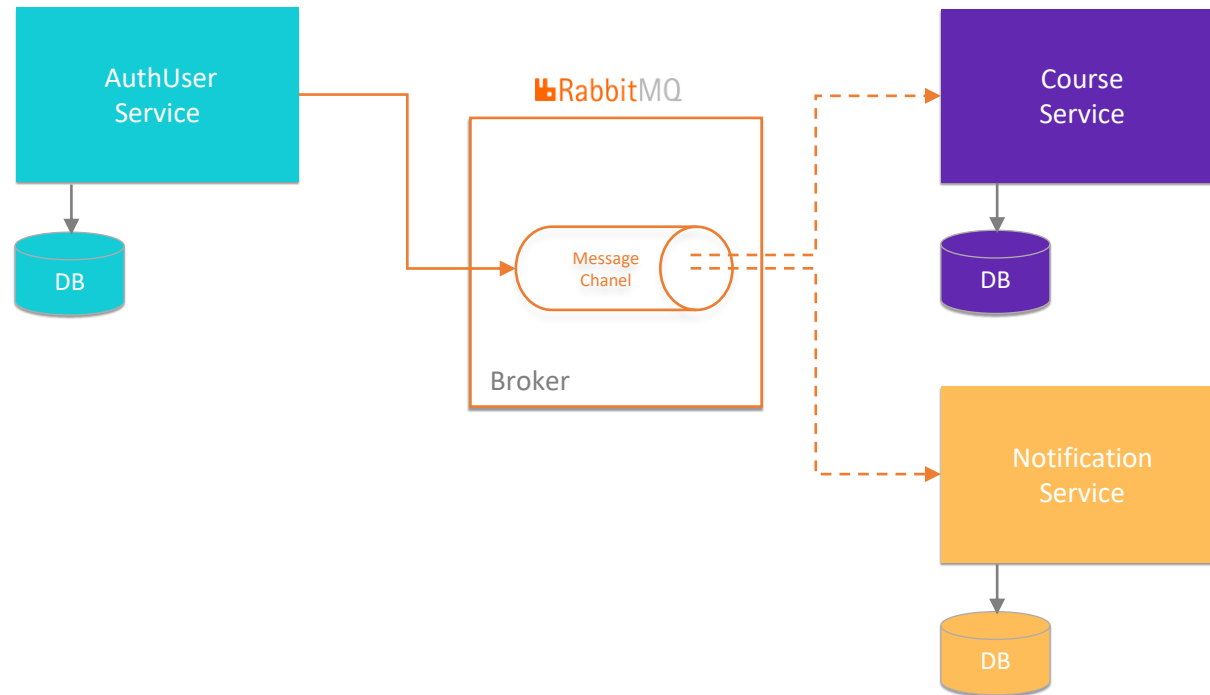
# Spring Data JPA



```
public interface RoleRepository extends JpaRepository<RoleModel, UUID> {  
    Optional<RoleModel> findByRoleName(RoleType name);  
}
```

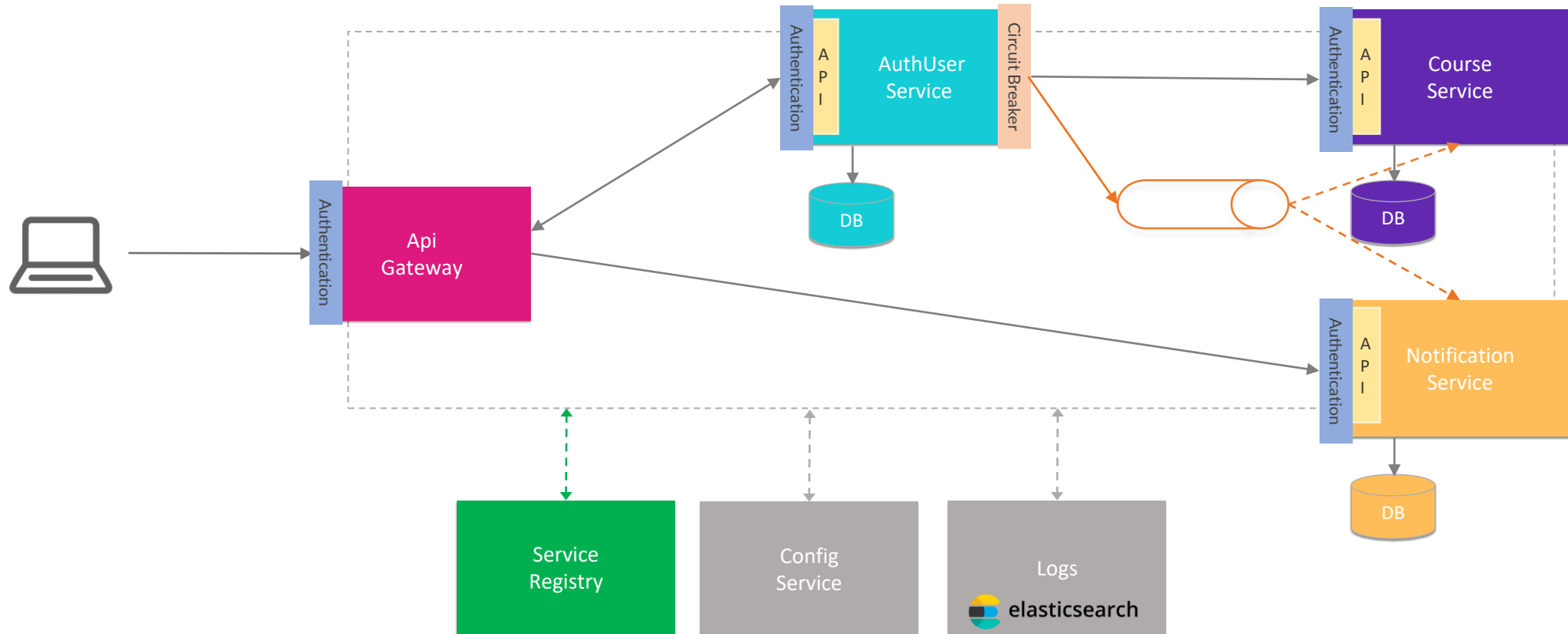
```
<dependency>  
  <groupId>org.springframework.boot</groupId>  
  <artifactId>spring-boot-starter-data-jpa</artifactId>  
</dependency>
```

# Spring AMQP



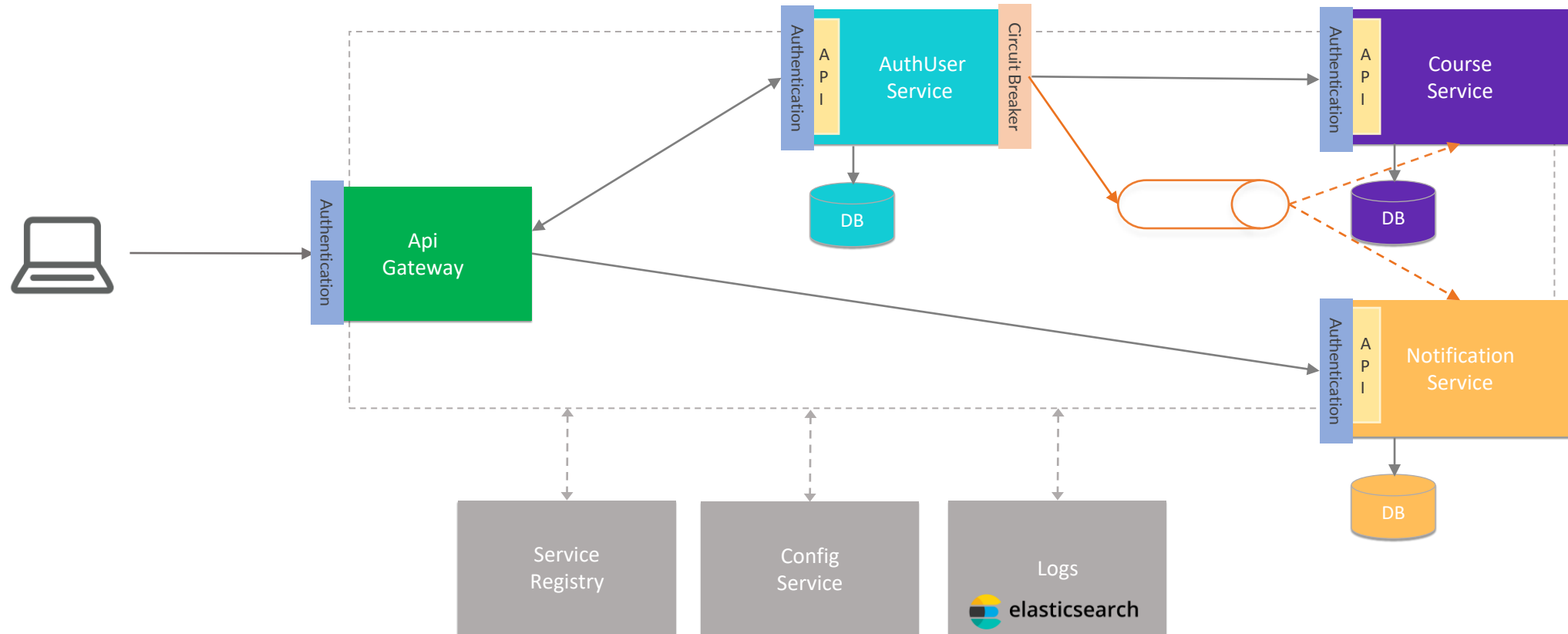
```
<dependency>
  <groupId>org.springframework.boot</groupId>
  <artifactId>spring-boot-starter-amqp</artifactId>
</dependency>
```

# Spring Cloud Netflix Eureka



```
<dependency>
  <groupId>org.springframework.cloud</groupId>
  <artifactId>spring-cloud-starter-netflix-eureka-server</artifactId>
</dependency>
```

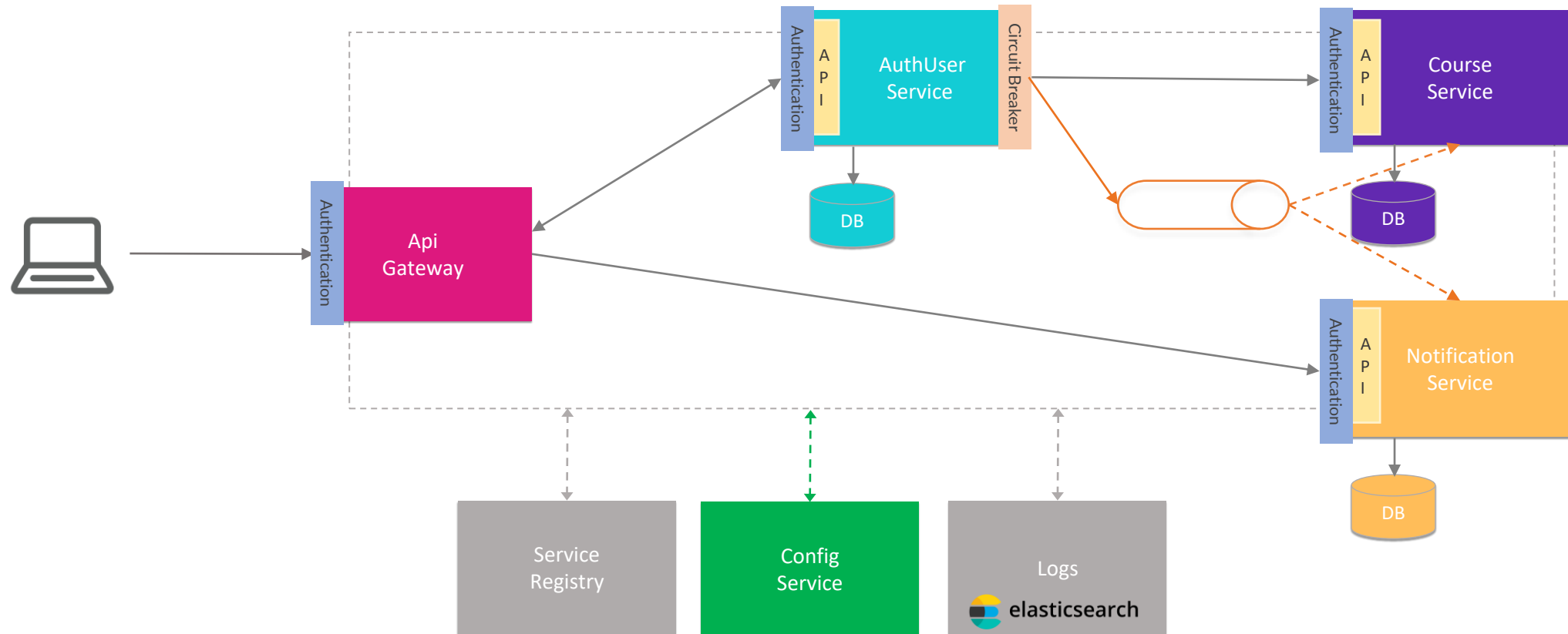
# Spring Cloud Gateway



```
<dependency>
  <groupId>org.springframework.cloud</groupId>
  <artifactId>spring-cloud-starter-gateway</artifactId>
</dependency>
```

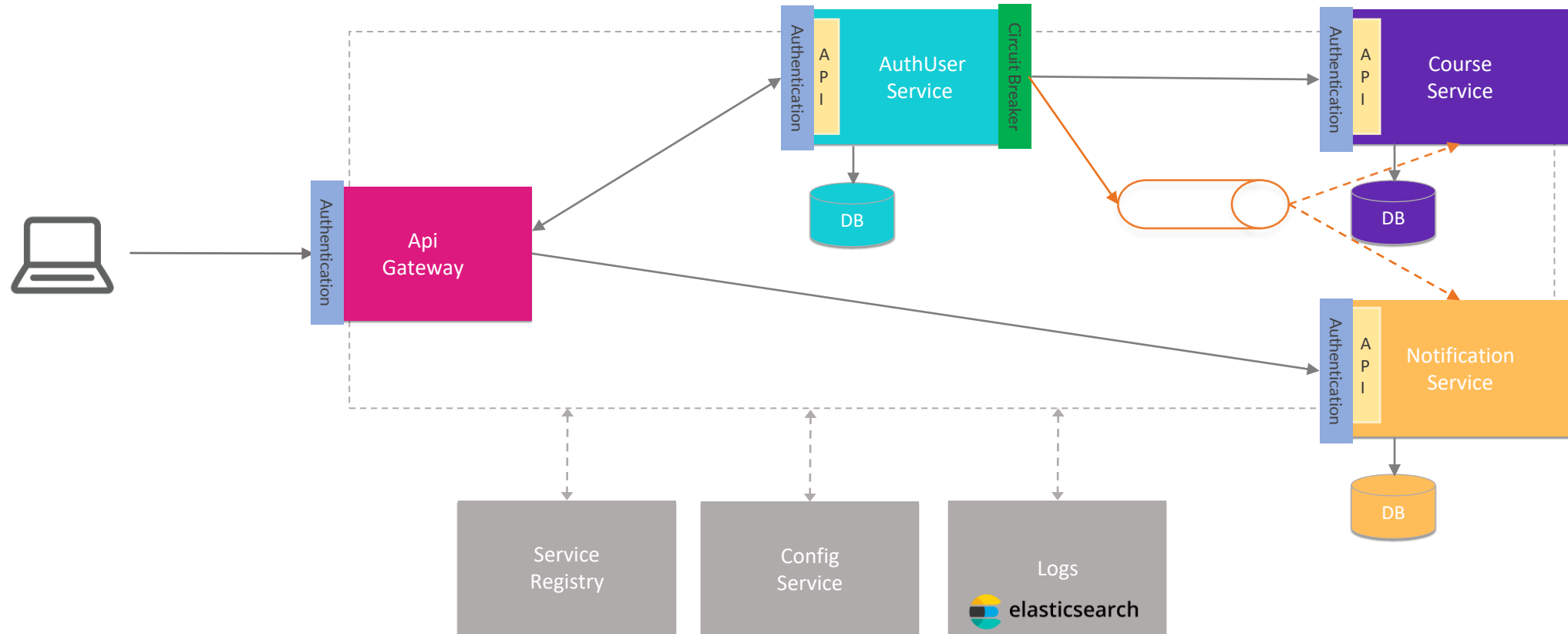


# Spring Cloud Config



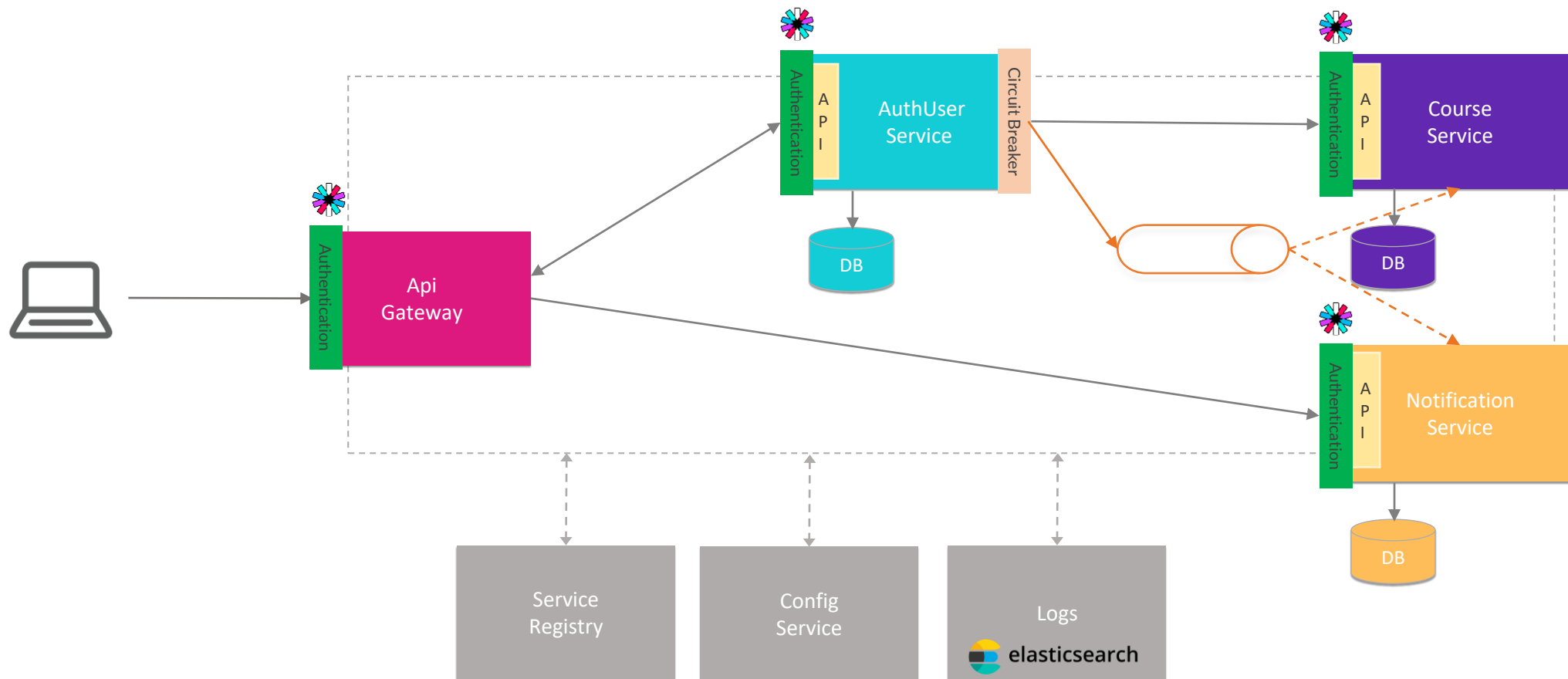
```
<dependency>
  <groupId>org.springframework.cloud</groupId>
  <artifactId>spring-cloud-config-server</artifactId>
</dependency>
```

# Spring Cloud Circuit Breaker Resilience4J



```
<dependency>
  <groupId>org.springframework.cloud</groupId>
  <artifactId>spring-cloud-starter-circuitbreaker-resilience4j</artifactId>
</dependency>
```

# Spring Security



```
<dependency>
  <groupId>org.springframework.boot</groupId>
  <artifactId>spring-boot-starter-security</artifactId>
</dependency>
```

# Microservices Architecture with Spring - EAD

