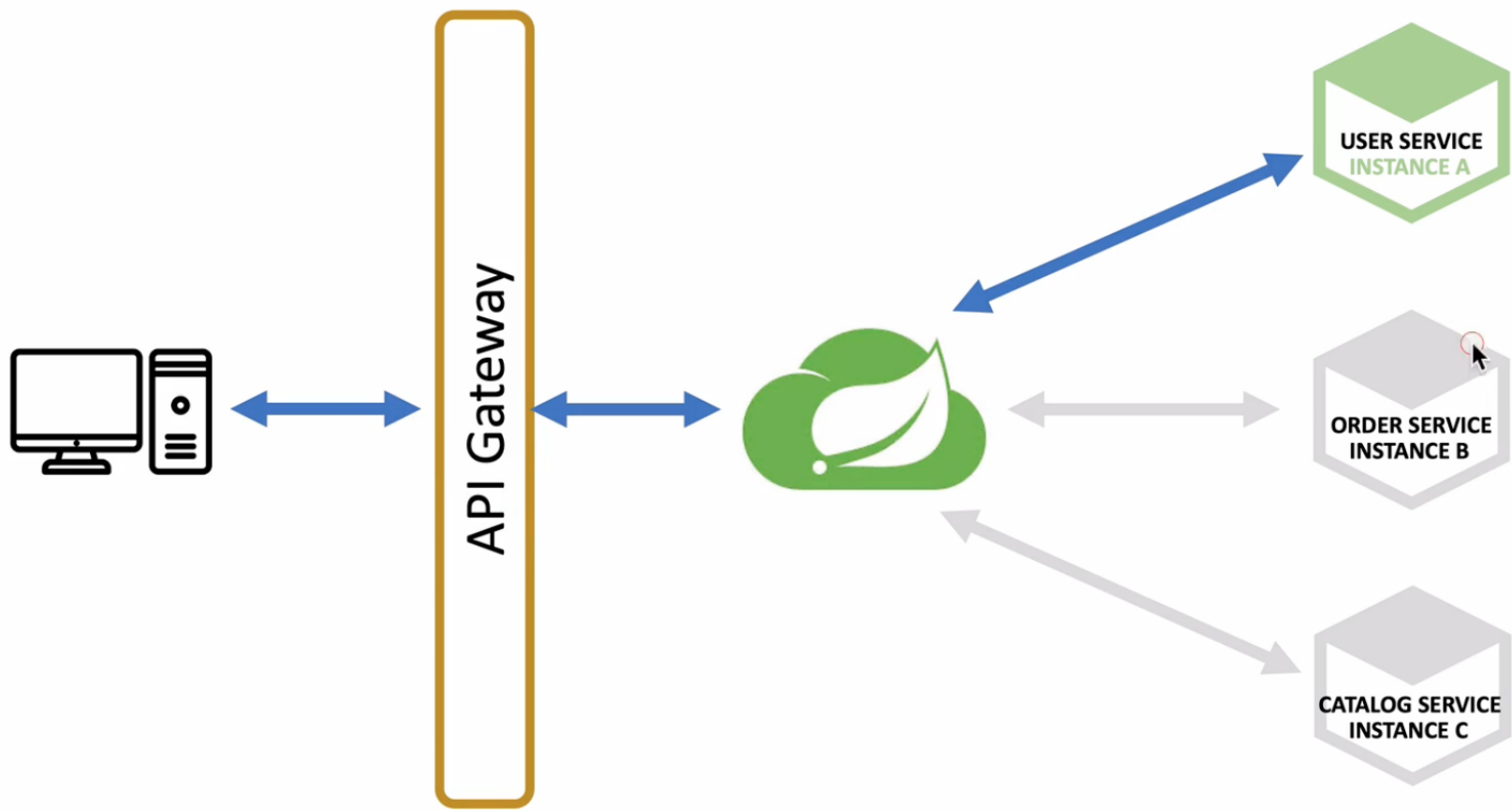


MSA 5주차

≡ 스터디원	소현
⚙ Status	Done
🔗 URL	https://github.com/SPRING-STUDY-2023/sohyeon-spring-cloud-msa/pull/2
≡ 비고	MSA Section4
📅 제출 마감일	@October 11, 2023 1:00 PM

Users Microservice(1)



프로젝트 생성

Dependencies

- Spring Boot DevTools
- Lombok
- Spring Web
- Eureka Discovery Client

Code

<https://github.com/SPRING-STUDY-2023/sohyeon-spring-cloud-msa/pull/2/commits/ff179e27878e8c247600e550a20696805a082953>

H2 Database

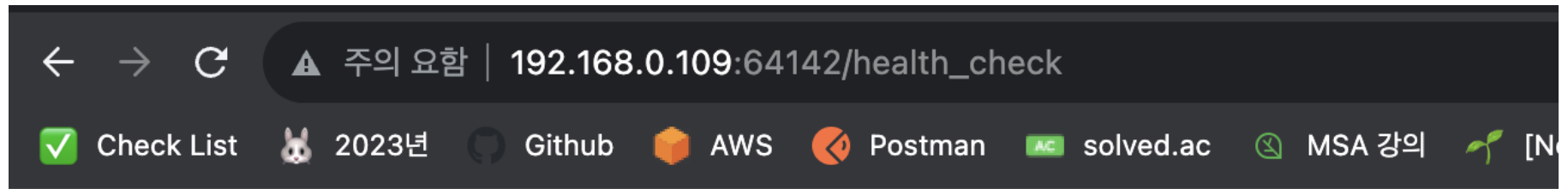
- 자바로 작성된 오픈소스 RDBMS
- Embedded, Server-Client 가능

- JPA 연동 가능

welcome() 메소드

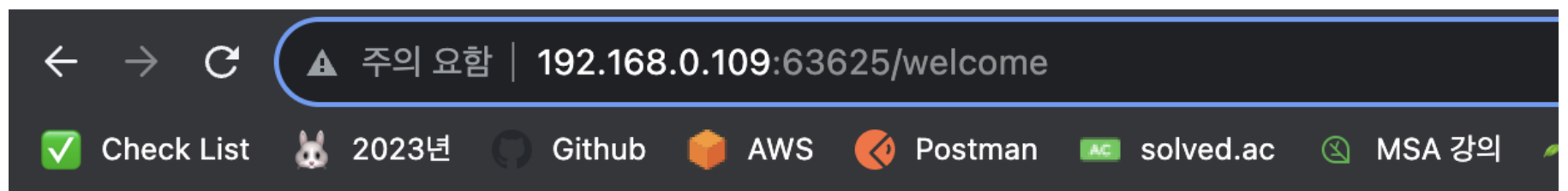
UserController

- GET /health_check



It's working in User Service.

- GET /welcome



Welcome to Simple E-Commerce.

H2 데이터베이스 연동

Trouble Shooting

- H2 Test Connection에 실패할 때

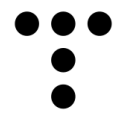
[H2] Database "mem:testdb" not found, either pre-create it or allow remote database creation 해결

Intro 스프링부트에서는 아래와 같은 설정으로 간단하게 인메모리 H2 데이터베이스를 사용 할 수 있습니다.

application.yml spring: datasource: url: jdbc:h2:mem:testdb username: sa build.gradle.kts

runtimeOnly("com.h2database:h2") 이후 h2:console 설정을 켜다면, 콘솔 페이지에서 데이터베이스를 직접 확인 해

<https://shanepark.tistory.com/435>



사용자 추가

회원가입

- POST /users/
- UserController ↔ UserService ↔ UserRepository 흐름
- RequestUser(RequestDTO), UserDTO, UserEntity 데이터 교환

JPA 설정

- dependency 추가

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

JPA(1)



createUser() 메소드 추가

Code


<https://github.com/SPRING-STUDY-2023/sohyeon-spring-cloud-msa/pull/2/commits/cb96fcb558aed9a74e19ff215b9d03059ec664b9>

Trouble Shooting

- 어노테이션을 사용할 수 없을 때

@Size와 @Past import가 안됩니다. - 인프런 | 질문 & 답변

유효성 체크를 위한 Validation API 사용 강의에서 @Size와 @Past를 사용하려 하는데 자동으로 import javax.validation.constraints.Size, mport javax.validation.constraints.Past가 안됩니다. 수동...

 <https://www.inflern.com/questions/38577/size와-past-import가-안됩니다>



JPA(2)



createUser() 메소드 수정

Code

<https://github.com/SPRING-STUDY-2023/sohyeon-spring-cloud-msa/pull/2/commits/5b27ce02d90b85fbbc9258456b29e16ad87456a3>

테스트

POST

localhost:49940/users

ParamsAuthorizationHeadersBodyPre-request ScriptTestsSettings

none

form-data

x-www-form-urlencoded

raw

binary

GraphQL

JSON

1

{

2

"email": "thgus345@naver.com",

3

"name": "Sohyeon Kim",

4

"pwd": "test1234"

5

}

BodyCookiesHeaders (5)Test Results

Body

PrettyRawPreviewVisualizeText

1 Create user method is called

RunRun SelectedAuto completeClearSQL statement:

SELECT * FROM USERS USERS |

SELECT * FROM USERS USERS;

ID	EMAIL	NAME	ENCRYPTED_PWD	USER_ID
1	thgus345@naver.com	Sohyeon Kim	encrypted_password	747a3cb8-8e50-442c-9c2c-d443bd679460

(1 row, 3 ms)

Edit

커스텀 후

POST

localhost:50397/users

Send

ParamsAuthorizationHeadersBodyPre-request ScriptTestsSettings

none

form-data

x-www-form-urlencoded

raw

binary

GraphQL

JSON

1

{

2

"email": "thgus345@naver.com",

3

"name": "Sohyeon Kim",

4

"pwd": "test1234"

5

}

BodyCookiesHeaders (5)Test Results

Body

PrettyRawPreviewVisualizeJSON

1

2

"email": "thgus345@naver.com",

3

"name": "Sohyeon Kim",

4

"userId": "6eed0d58-1c76-409b-a225-c0330f9f8cef"

5

}

Status: 201 CreatedTime: 139 msSize: 268 BSave as Example

Spring Security 연동

Spring Security

- Authentication + Authorization (인증 + 권한)

- Step 1: 애플리케이션에 **spring security jar**을 Dependency에 추가
- Step 2: **WebSecurityConfigurerAdapter**를 상속받는 **Security Configuration** 클래스 생성
- Step 3: Security Configuration 클래스에 **@EnableWebSecurity** 추가
- Step 4: Authentication → **configure(AuthenticationManagerBuilder auth)** 메서드를 재정의
- Step 5: Password encode를 위한 **BCryptPasswordEncoder** 빈 정의
- Step 6: Authorization → **configure(HttpSecurity http)** 메서드를 재정의

BCryptPasswordEncoder

- Password 해싱을 위해 Bcrypt 알고리즘 사용
- 랜덤 Salt 부여 및 여러 번 Hash를 적용한 암호화 방식

Code


<https://github.com/SPRING-STUDY-2023/sohyeon-spring-cloud-msa/pull/2/commits/ff6291441a09aa7a37a468736cc57ba74064d9ba>

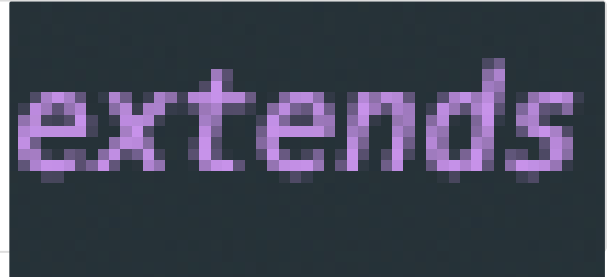
Trouble Shooting

- **WebSecurityConfigurerAdapter** deprecated

Deprecated된 WebSecurityConfigurerAdapter, 어떻게 대처하지?

스프링 버전이 업데이트 됨에 따라 WebSecurityConfigurerAdapter와 그 외 몇 가지들이 Deprecated 됐습니다.스프링에서는 다른 방식으로 시큐리티 설정을 권장하고 있는 듯 해보였는데요. 방식이 바뀐 탓에 시큐리티 설정에 어려움을 겪었고 제가 사


 <https://vlog.io/@pjh612/Deprecated된-WebSecurityConfigurerAdapter-어떻게-대처하지#변경전-설정과-변경-후-설정>

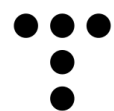


- filterChain 일부 기능 deprecated (and(), authorizeHttpRequests() 등)

security 6.1 마이그레이션

spring boot 3.1 로 변경하면서 security에 deprecated 된 코드들이 있다. 예시는 아래와 같으며 수정을 원하는 경우 configurer 를 사용하면되며 기존 설정은 Customizer.withDefaults() 을 사용하면된다. # 기존 코드 public SecurityFilterChain filterChain(final @NotNull HttpSecurity http) throws Exception { http .httpBasic().disable()


 <https://uriu.tistory.com/435>



- authorizeHttpRequests 컴파일 오류

[Spring] This method cannot decide whether these patterns are Spring MVC patterns or not.

Caused by: java.lang.IllegalArgumentException: This method cannot decide whether these patterns are Spring MVC patterns or not. If this endpoint is a

 <https://vlog.io/@shdbtjd8/Spring-This-method-cannot-decide-whether-these-patterns-are-Spring-MVC-patterns-or-not>



Security 및 Bcrypt 적용 후 API 테스트

POSTlocalhost:51888/usersSend

ParamsAuthorizationHeadersBodyPre-request ScriptTestsSettingsCookies

noneform-data x-www-form-urlencodedrawbinaryGraphQLJSONBeautify

```
1 {
2   ... "email": "thgus345@naver.com",
3   ... "name": "Sohyeon Kim",
4   ... "pwd": "test1234"
5 }
```

BodyCookiesHeaders (10)Test ResultsStatus: 201 CreatedTime: 251 msSize: 415 BSave as Example

PrettyRawPreviewVisualizeJSON

```
1 {
2   "email": "thgus345@naver.com",
3   "name": "Sohyeon Kim",
4   "userId": "fc22a91f-b09f-4f0d-a818-56456548652d"
5 }
```

SELECT * FROM USERS;

ID	EMAIL	NAME	ENCRYPTED_PWD	USER_ID
1	thgus345@naver.com	Sohyeon Kim	\$2a\$10\$tUxFifymzwwkRRj5le1Efmen5cz.mdzbRWPIg5jXJf1056MinJI8va	fc22a91f-b09f-4f0d-a818-56456548652d

(1 row, 3 ms)

Edit