

2023-하계방학 자기주도학습동아리 활동보고서

팀명	JUMP		
학습일시	2023년 08월 20일 (일요일)	진행차수	8주차
학습시간	00시 00분 ~ 02시 00분 (2시간)		
참석자	이 영 교 (서명)	김 태 현 (서명)	손 민 재 (서명)
	이 호 영 (서명)	최 재 신 (서명)	진 정 원 (서명)
불참자	진정원(불참 사유 : 여행)		
구 분	내 용		
학습목표	- 현재까지 공부했던 내용들을 리마인드하는 시간		
학습 방법 및 학습 내용	<p>0. 이번 주차는 노선에 내용을 정리해서 세션처럼 설명하는 방식으로 진행했습니다. 웹사이트를 개발하는 과정에서 어려웠던 점을 공유하는 시간을 갖게 되었습니다. 추가로, 개발하면서 있었던 오류사항들을 공유하는 시간이었습니다. 마지막 정기 스터디로, 자신이 더 공부해보고 싶은 부분을 공유하면서 마무리했습니다.</p> <p>깃허브 링크 : https://github.com/JUMP-Hongik/JUMP-Web-Study</p> <p>1. 각자 파트에서 어려웠던 점 공유</p> <p>- 프론트엔드:</p> <p>클론 코딩을 진행하면서, 어떻게 개발하는 것인지 이해할 수 있었지만 이후 원하는 틀을 구현하기에는 CSS 측면에서 어려움을 많이 겪었다. HTML, JS 보다는 CSS를 잘 다룰 수 있어야 프론트엔드를 마스터할 수 있을 것 같다. 또한, 리액트 서적에 나와있는 버전과 실제 배포되어 있는 버전이 달라서 구동이 되지 않는 이슈들도 있어 공유하는 시간을 갖게 되었다.</p> <p>- 백엔드:</p> <p>Springboot를 활용해서 개발하니까, 많은 부분이 이미 개발되어 있어 편리성 있게 모듈들을 이해하고 가져다 쓰면 기능들을 구현할 수 있는 것이 좋았다. 장점만 있는 것이 아니라, 너무 많은 모듈들이 이미 구현되어 있어, 로그인 같은 회원 플로우를 구현할 때 하나의 기능을 구현하기 위해 너무 많은 모듈들을 이해해야 하는 점이 진도를 많이 못 나가게 되는 이유 중에 하나였다.</p> <p>한가지 개발하면서 이슈 사항이 있었는데, 버전이 업데이트 되면서 기존에 사용하던 코드가 deprecated되어, 다른 방식으로 구현해야 하는 점 때문에 이를 구글링하는 과정에서 어려움을 많이 겪었다.</p> <p>이전 방식과 현재 수정한 방식은 다음과 같이 변경했다.</p>		



```
// 이전 방식
@Configuration
@EnableWebSecurity
public class securityConfig extends WebSecurityConfigurerAdapter {

    @Override
    protected void configure(HttpSecurity http) throws Exception {
        http.csrf.disable();
        http.authorizeRequest()
            .antMatchers("/user/**").authenticated()
            .antMatchers("/manager/**").access("hasRole('ROLE_ADMIN')")
            .anyRequest().permitAll();
    }
}
```

```
// 권장 방식
@Configuration
@EnableWebSecurity
public class SecurityConfig {

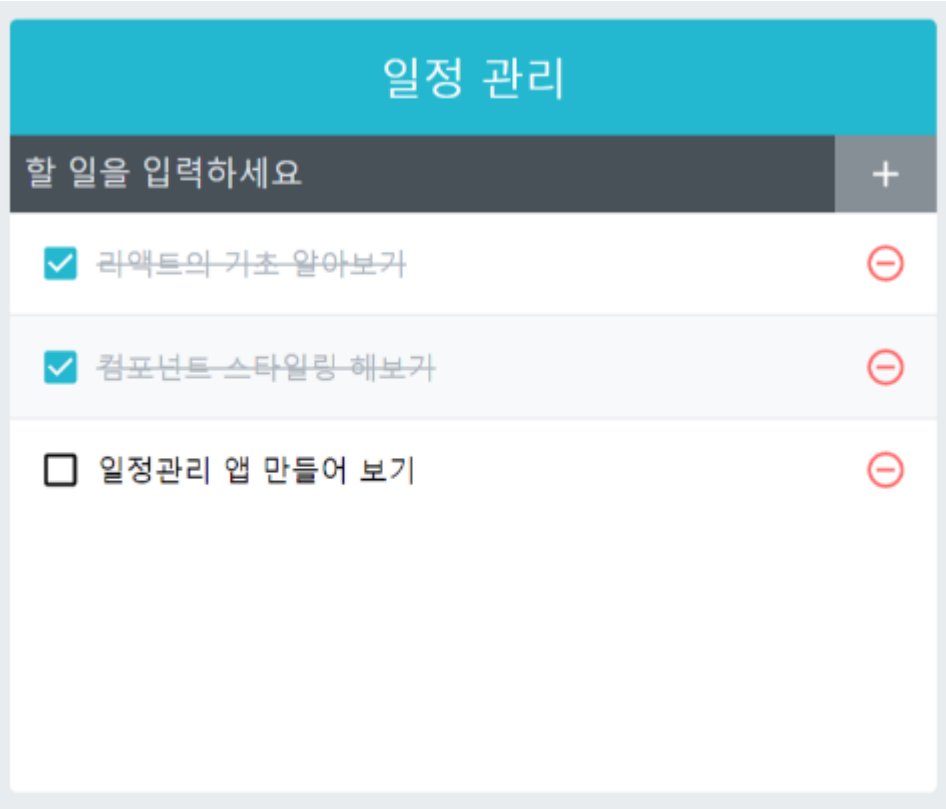
    @Bean
    public SecurityFilterChain filterChain(HttpSecurity http) throws Exception {
        http.csrf.disable();
        http.authorizeHttpRequests()
            .requestMatchers("/user/**").authenticated()
            .requestMatchers("/manager/**").access(string 으로 들어가면 에러);
        //
            .requestMatchers("/manager/**").hasAnyAuthority("ROLE_ADMIN", "ROLE_MANA
            .requestMatchers("/admin/**").hasAuthority("ROLE_ADMIN")
            .anyRequest().permitAll(); // 다른 페이지는 모두 허용

        return http.build();
    }
}
```

2. 개발 진행 상황

클론 코딩을 진행하면서 일정 관리 웹 페이지를 제작하였는데, 이 앞 부분에 회원가입과 로그인 을 하는 기능을 추가하고, 일정 관리 웹에서 구글폼과 같이 세부사항을 변경하면 일단 구현해보 고자 하는 기능들을 만들 수 있을 것 같다. 금요일 제출일 전까지 최대한 원하는 기능들을 붙여 서 개발하고자 한다.

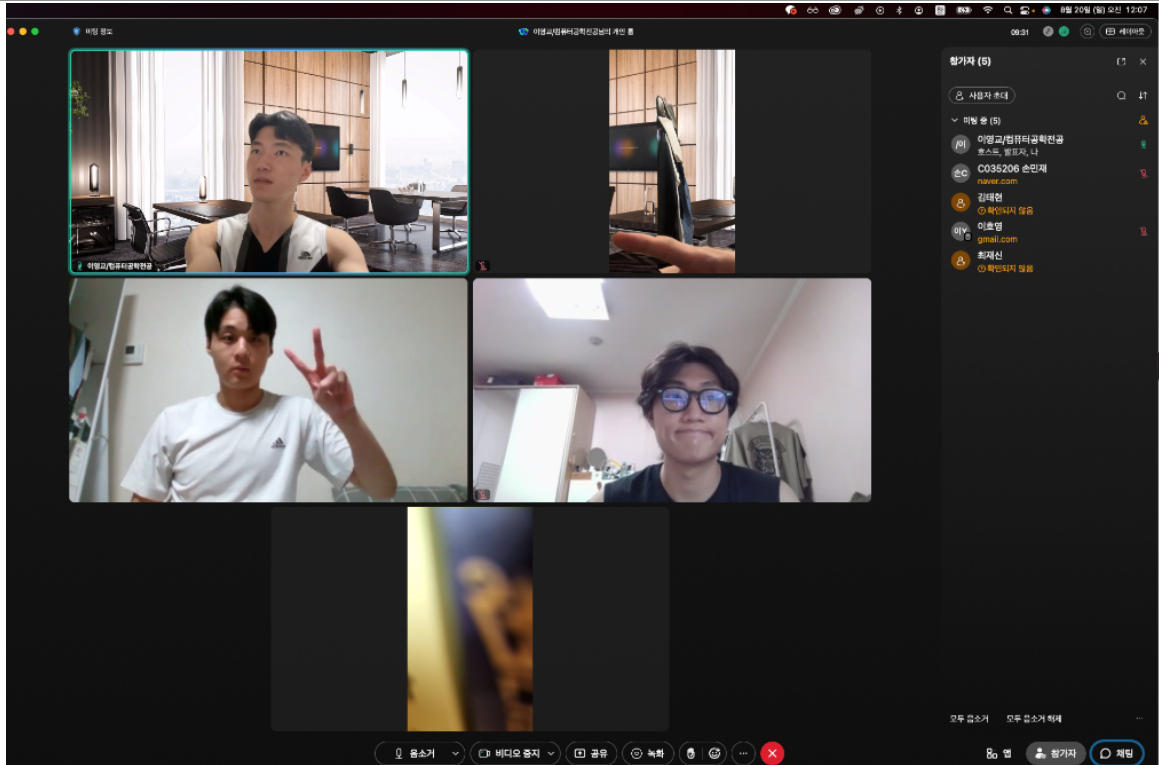




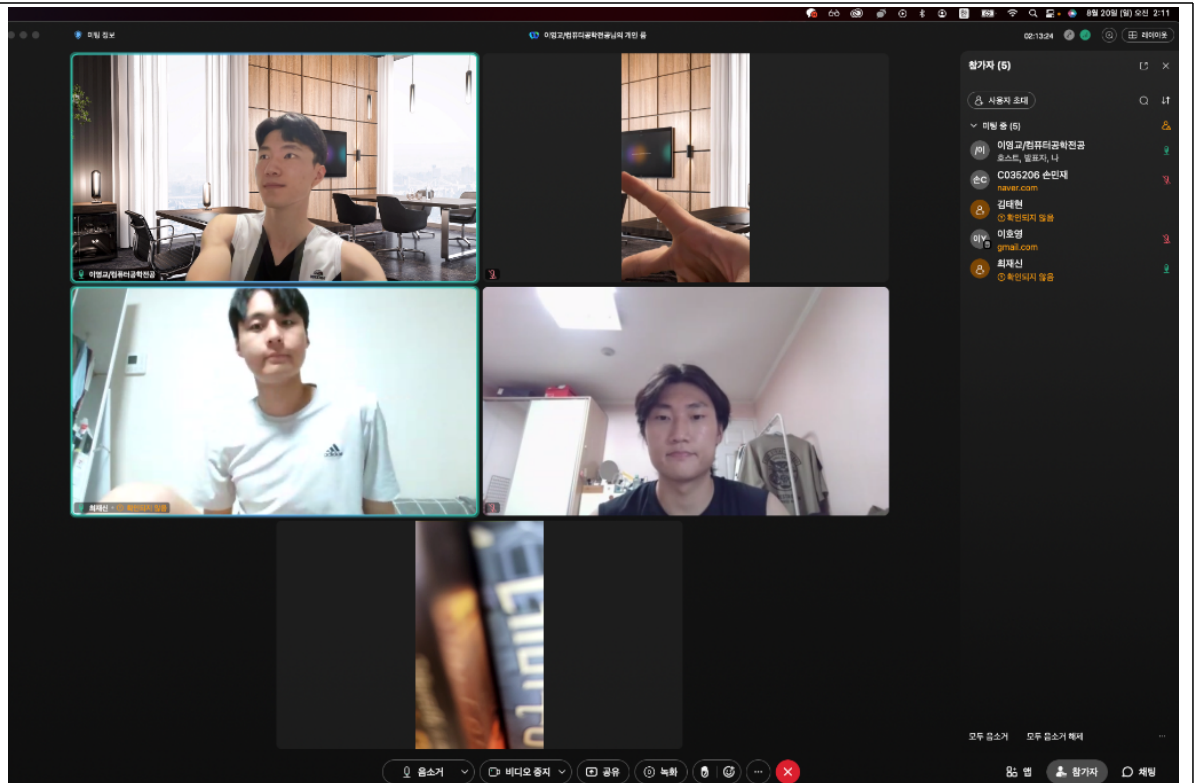
구현하고 있는 코드는 깃허브를 들어가서 확인할 수 있다.

활동 후기 * 본인이 맡은 역할 활동과 학습한 내용에 대한 후기 작성	팀장 / 팀원 이름	작성란
	이영교	마지막 활동보고서를 작성하면서 느꼈던 점은, 방학 기간 동안 많은 내용들을 이해할 수 있는 계기를 만든 것 같아 좋았다는 것입니다. 개인이 스터디에 집중할 수 있는 시간이 다르다보니 진도가 생각보다 많이 못 나간 것이 아쉽기는 하나, 이후에도 개발을 같이 진행할 것으로 천천히 나아갈 계획입니다.
	김태현	8주 동안 백엔드에 대해서 이해하고, 팀원들과 스터디를 하면서 재밌었습니다. 지금까지 한번도 개발해보지 못해서 조금은 어려웠던 과정이 많았지만, 이제는 조금 익숙해져서 속도를 붙여서 개발할 수 있을 것 같습니다.
	손민재	방학이 끝나가는데 구현하려 목표했던 기능들에 비해 공부량이 부족해서



		아쉬운 점이 많은데 학기 중에도 틈틈히 스터디를 이어가서 내년 신입부원 모집에 사용 가능하게 만들어 보고싶다.
	이호영	스터디원들 말고도 다른 동아리 부원들과 의논해서 추가했으면 하는 기능, 디자인적인 요소들도 개선시켜 가고싶다. 개강 후에도 동아리 방에서 같이 프로젝트를 이어가고 싶다.
	최재신	홈페이지를 만드는 과정을 이해한 것으로 만족하며, 기획하는 부분에서도 API 명세서 작성, 기능 설계 등 오히려 실제 개발하는 것보다 더 많은 것을 고려해야한다는 것을 알 수 있었다.
	진정원	방학 기간 동안 새로운 프로젝트를 한번 경험해봤다. 친한 친구들이랑 스터디를 하면서 이전에는 놀기만 했던 방학을 조금이라도 알차게 보낼 수 있었던 것 같다.
활동 증빙 사진	학습시작	
		
	학습종료	





학습활동

Cookie: JSESSIONID=1234567890; Domain=bank.example.com; Secure; HttpOnly
Content-Type: application/x-www-form-urlencoded

amount=100.00&routingNumber=1234&account=9876

- 로그아웃을 하지 않은 상태로, 다음과 같은 제출 양식의 악의적인 웹사이트에 접속하여 **submit** 누른다면, 의도하지 않게 악의적인 사용자에게 돈을 송금할 수 있다
- 악의적인 웹 사이트에선 쿠키를 볼 수 없지만, 은행과 관련된 쿠키는 Request와 함께 전송되기 때문이다

```
<form action="https://bank.example.com/transfer" method="post">
<input type="hidden"
  name="amount"
  value="100.00"/>
<input type="hidden"
  name="routingNumber"
  value="evilsRoutingNumber"/>
<input type="hidden"
  name="account"
  value="evilsAccountNumber"/>
<input type="submit"
  value="Win Money!"/>
</form>
```

- CSRF protection** 기능은 브라우저를 통해 Request를 받을 때 사용하라고 나와 있다
- REST API 서버는 **stateless** 하게 개발되기 때문에, 사용자의 정보를 **session** 에 저장하지 않는다
- JWT 같은 토큰을 사용하여 인증하기 때문에, 토큰을 **Cookie** 에 저장하지 않는다면, 안전하다고 생각

