

개발자 스터디, 프로젝트 커뮤니티

팀 명 : 수정용
지도 교수 : 양환석 교수님
팀 장 : 이정림
팀 원 : 권용호
김수빈

2022. 11.
중부대학교 정보보호학과

목 차

1. 서론

1.1 연구 배경	4
1.2 연구 목적 및 주제 선정	4

2. 관련 연구

2.1 Java	4
2.2 Spring boot	4
2.3 Spring Security	5
2.4 XML	5
2.5 JavaScript	5
2.6 React	5
2.7 Node.js	5
2.8 Redux	6
2.9 Mysql	6
2.10 HTML	6
2.11 CSS	6

3. 본론

3.1 유스케이스	7
3.2.1 Spring Security	7
3.2.1 Spring Security	16

4. 결론

4.1 결론	27
--------------	----

4.2 기대 효과 27

5. 별첨

5.1 소스 코드 28

5.2 발표 자료 29

1. 서론

1.1 연구 배경

코로나19는 우리의 일상과 사회 전반의 모습을 완전히 바꾸었다. 실내 공간이나 건물에 들어가려면 열 감지기를 통과해야 하고, 체온 측정은 필수가 되었다. '사회적 거리 두기'가 감염 확산을 막을 최선의 대책이 되면서 수업, 근무, 음식 주문 등 사회활동과 일상이 비대면 방식으로 변한 것이다. 모임에도 인원 제한이 생기면서 스터디, 프로젝트를 위한 모임에 제한이 생기게 되었다. 대면 모임이 어려워지자 비대면 스터디 모임인 '캠스터디(카메라+스터디)'에 나서는 사람들이 늘고 있다. 이런 코로나 시대에 개발을 공부하는 사람들이 제한없이 효율적으로 모일 수 있는 플랫폼의 필요성을 느꼈다. 스터디원이나 프로젝트 팀원을 구하기 위해서는 중간 다리 역할이 필요하다. 그것이 바로 코드미터다. 코드미터를 이용하면 온라인 또는 오프라인으로 스터디와 프로젝트를 진행할 수 있다.

1.2 연구 목적 및 주제 선정

프로그래밍의 공부는 끝이 없다고 생각한다. 혼자 공부하고, 혼자 프로젝트를 해보는 경험도 매우 중요하지만 컴퓨터 개발 관련 공모전, 스터디, 프로젝트를 개인이 준비하는 것은 한계가 있고, 스터디 그룹을 이루어 공부하거나 다른 개발자들과 함께 프로젝트를 진행하는 것도 좋은 방법이라고 생각한다. 다양한 개발환경 중 원하는 주제를 정해 팀원들을 모집하는 일은 쉽지 않기 때문에 그 과정에서 보다 편리하게 개발자들이 스터디, 프로젝트의 팀원들을 모집하고 또한 자신이 원하는 분야의 공모전을 제한 없이 등록하고 지원할 수 있는 커뮤니티가 필요하다고 생각해 코드미터를 시작하게 되었다.

2. 관련 연구

2.1 Java

객체 지향 프로그래밍 언어로서 보안성이 뛰어나며 컴파일한 코드는 다른 운영 체제에서 사용할 수 있도록 클래스(class)로 제공된다. 객체 지향 언어인 C++ 언어의 객체 지향적인 장점을 살리면서 분산 환경을 지원하며 더욱 효율적이다. 자바 프로그램은 운영체제의 종류에 관계없이 대부분의 시스템에서 실행 가능하다.

2.2 Spring boot

스프링 부트(Spring Boot)는 Java 기반의 애플리케이션 개발을 위한 포괄적인 인프라를 제공한다. 스프링 프레임워크를 더 빠르고 쉽게 사용할 수 있게 도와준다. 스프링을 사용하기 위한 설정의 많은 부분을 자동화하여 사용자가 편하게 스프링을 활용할 수 있도록 돕는다. 스프링 부트 starter 디펜던시만 추가해주면 바로 API를 정의하고, 내장된 탬플릿이나 제티로 웹 애플리케이션 서버를 실행할 수 있다. 또한 스프링 홈페이지의 이니셜라이

저를 사용하면 바로 실행 가능한 코드를 만들어준다. 실행환경이나 의존성 관리 등의 인프라 관련 등은 신경 쓸 필요 없이 바로 코딩을 시작하면 된다.

2.3 Spring Security

Spring 기반의 Web 애플리케이션 보안 인증 및 인가를 담당하는 하위 프레임 워크이다. 특히, Servlet Filter를 기반으로 인증을 지원하기 때문에 filter를 등록하면 Servlet container 안에 있는 다른 애플리케이션과 사용이 가능하다는 장점을 가지고 있다. CSRF 및 세션 고정 공격 등 잘 알려진 다양한 공격에 대한 보호 기능을 제공해준다.

2.4 XML

인터넷 웹페이지를 만드는 html을 획기적으로 개선하여 만든 언어이다. XML은 웹에서 구조화된 문서를 전송 가능하도록 설계되었기 때문에 문서를 구성하는 각 요소들의 독립성을 보장함으로써 문서의 호환성, 내용의 독립성, 요소 변경의 용이성 등의 특성을 제공한다. 또한 XML로 문서를 교환할 때 각자가 가지고 있는 응용프로그램이 달라도 호환이 가능하다. 이러한 점때문에 XML언어는 전자상거래, EDI(전자문서교환) 등을 중심으로 차세대 인터넷언어로 빠르게 세력을 확장하고 있다. 어떤 문서가 구조, 내용, 표현으로 구성되어 있다고 했을 때, XML은 구조와 내용만을 규정하기 때문에 XML로 기술된 데이터를 화면에 표현하려면 다른 화면표현용 언어를 사용해야 한다.

2.5 JavaScript

자바스크립트는 크로스 플랫폼(cross platform), 객체지향 스크립트 언어로 웹페이지의 동작을 담당한다. 또한 웹 페이지에서 사용자로부터 특정 이벤트나 입력 값을 받아 동적인 처리를 목적으로 고안된 객체 기반의 스크립트 프로그래밍 언어이다. 주로 웹 브라우저 내에서 사용되는 언어였으나, 자바스크립트 기반의 런타임 플랫폼(예: Node.js)들이 개발되면서 서버측 프로그램 개발에도 사용이 크게 확대되었다.

2.6 React

사용자 인터페이스를 만들기 위한 JavaScript 라이브러리의 하나로서 사용자 인터페이스를 만들기 위해 사용된다. 페이스북과 개별 개발자 및 기업들 공동체에 의해 유지보수된다. 리액트는 싱글 페이지 애플리케이션이나 모바일 애플리케이션 개발에 사용될 수 있다. 대규모 또는 복잡한 애플리케이션 개발에는 보통 라우팅, API통신 등의 기능이 요구되는데 리액트에는 기본적으로 제공되지 않기 때문에 추가 라이브러리를 사용해야 한다. 기존의 웹 기술 HTML, CSS 등과 결합하여 사용할 수 있어 확장성이 뛰어나다.

2.7 Node.js

노드JS는 자바스크립트 엔진 'V8' 위에서 동작하는 이벤트 처리 I/O 프레임워크다. 또한 확장성 있는 네트워크 애플리케이션 개발에 사용되는 소프트웨어 플랫폼이다. 서버 환경에서 자바스크립트로 애플리케이션을 작성할 수 있도록 도와준다. 내장 HTTP 서버 라이

브러리를 포함하고 있어 웹 서버에서 아파치 등의 별도의 소프트웨어 없이 동작하는 것이 가능하며 이를 통해 웹 서버의 동작에 있어 더 많은 통제를 가능하게 한다.

2.8 Redux

오픈 소스 자바스크립트 라이브러리의 일종으로, state를 이용해 웹 사이트 혹은 애플리케이션의 상태 관리를 해줄 목적으로 사용한다. 또한 자바스크립트 애플리케이션에서 상태를 효율적으로 관리할 수 있게 도와주는 도구이다. 복잡한 상태 관리가 이루어지는 SPA(Single Page Application)에서 특히 유용하게 사용된다. 리덕스는 리액트 뿐만 아니라 jQuery, Angular 등을 사용하는 애플리케이션에서도 사용할 수 있다.

2.9 Mysql

오라클 사가 관리 및 배포하고 있는 오픈소스 관계형 데이터베이스 관리 시스템이다. 매우 빠르고, 유연하며, 사용하기 쉬운 특징이 있다. 다중 사용자, 다중 스레드를 지원하고, C, C++, 에펠, 자바, 펄, PHP, 파이썬 스크립트 등을 위한 응용 프로그램 인터페이스(API)를 제공한다. 유닉스나 리눅스, 윈도우 운영체제 등에서 사용할 수 있다. 리눅스 운영 체제와 아파치 서버 프로그램, MySQL, PHP 스크립트 언어 구성은 상호 연동이 잘되면서도 오픈소스로 개발되는 무료 프로그램이어서 홈페이지나 쇼핑몰 등 일반적인 웹 개발에 널리 이용되고 있다.

2.10 HTML

HTML(Hyper Text Markup Language)은 가장 단순한 형태의 웹 언어이다. 인터넷 서비스의 하나인 월드 와이드 웹을 통해 볼 수 있는 문서를 만들 때 사용하는 기본적인 웹 언어의 한 종류이다. 특히 하이퍼텍스트를 작성하기 위해 개발되었으며, 인터넷에서 웹을 통해 접근되는 대부분의 웹 페이지들은 HTML로 작성된다. HTML은 문서의 글자크기, 글자색, 글자모양, 그래픽, 문서이동(하이퍼링크) 등을 정의하는 명령어로서 홈페이지를 작성하는 데 쓰인다. HTML은 전자 문서의 서식을 정의하기 위해 만들어졌으며, 국제표준 SGML의 부분 집합으로 정의되었다. HTML은 SGML에서 특히 하이퍼텍스트를 강조하여 만들어진 언어이며, 아스키코드로 된 일반적인 텍스트로 구성되었다. 이 언어는 별도 컴파일러가 필요치 않으며, 웹 브라우저에서 해석이 가능한 사용하기 쉬운 언어로 각광을 받고 있다.

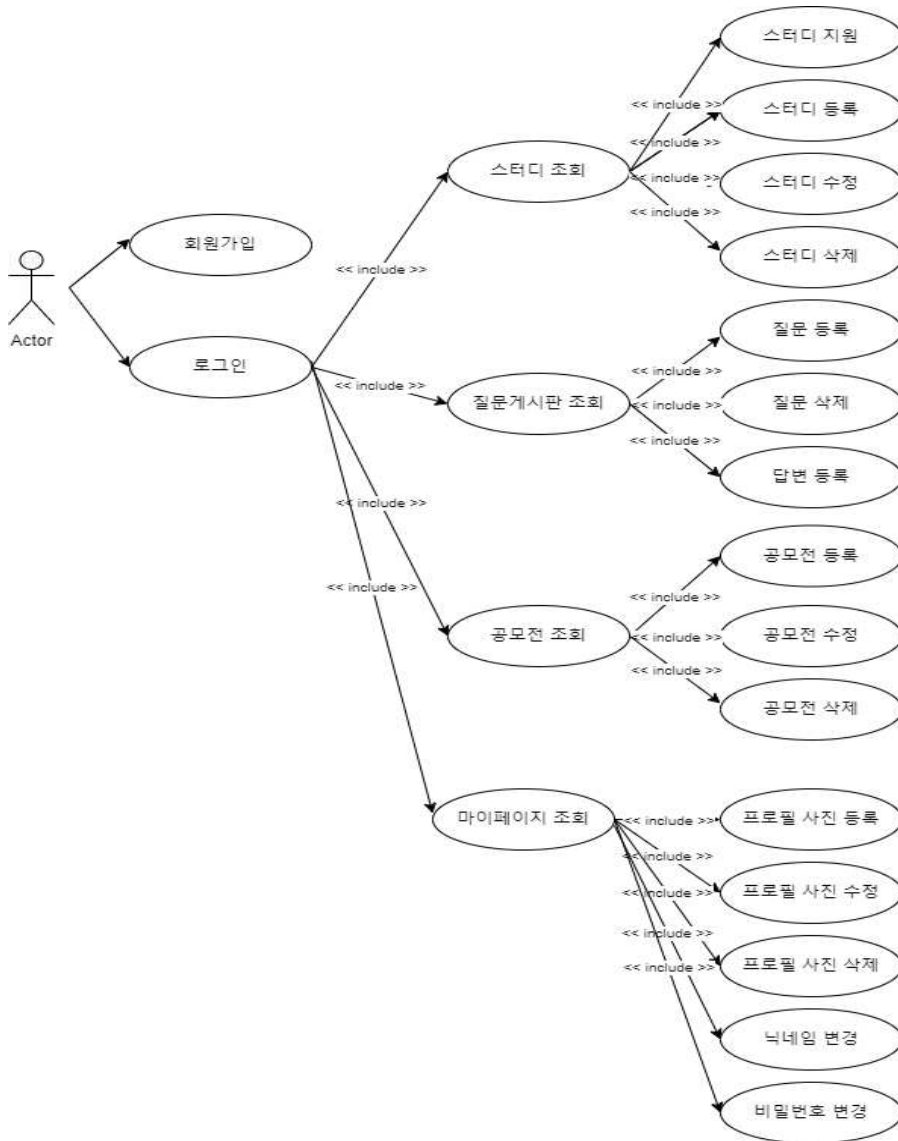
2.11 CSS

기존의 HTML은 웹 문서를 다양하게 설계하고 수시로 변경하는데 많은 제약이 따르는데, 이를 보완하기 위해 만들어진 것이 스타일 시트이고 스타일 시트의 표준안이 바로 CSS이다. 웹 문서의 전반적인 스타일을 미리 저장해 둔 스타일시트이다. 문서 전체의 일관성을 유지할 수 있고, 세세한 스타일 지정의 필요를 줄어줄게 하였다. HTML과 XHTML에 주로 쓰이며, 여러 수준과 프로파일을 가지고 있다. 각 수준의 CSS는 일반적으로 새로운 기능을 담고 있으며 CSS1, CSS2, CSS3, CSS4로 나뉜다. 프로파일들은 일반적으로 특

정한 장치나 사용자 인터페이스를 위해 만들어진 하나 이상 수준의 CSS의 하부 집합이다.

3. 본론

3.1 유스케이스



3.2.1 Spring Security

Spring Security는 Spring기반 애플리케이션을 보호하기 위한 표준으로 직접 커스텀이 가능한 인증 및 Access Control 프레임워크이다. 이는 Java 애플리케이션에 인증과 인가(권한 부여)를 모두 제공하는 데 중점을 뒀다. Spring Security는 다른 Spring프로젝트와 마찬가지로 사용자 요구에 맞춰 쉽게 확장할 수 있다는 점이

최대 장점이다.

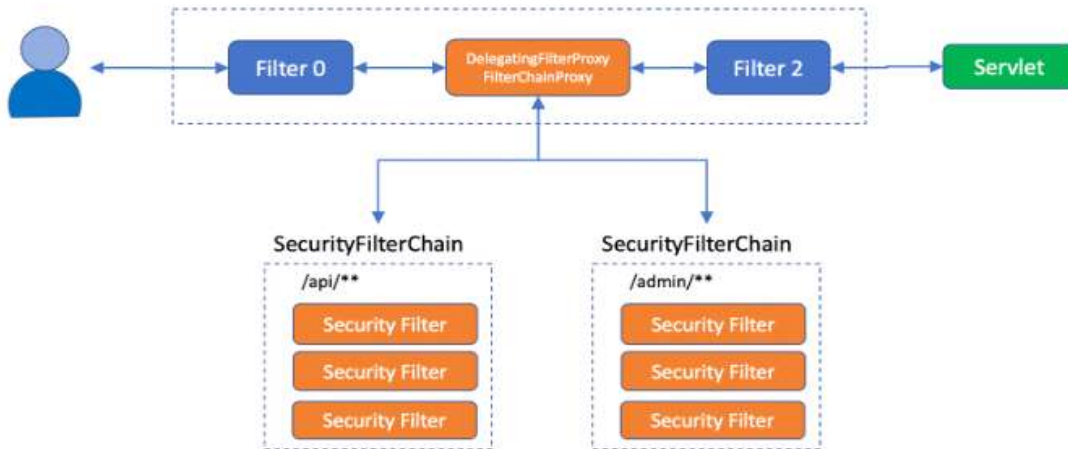


그림 1. Spring Security Filter 동작 순서

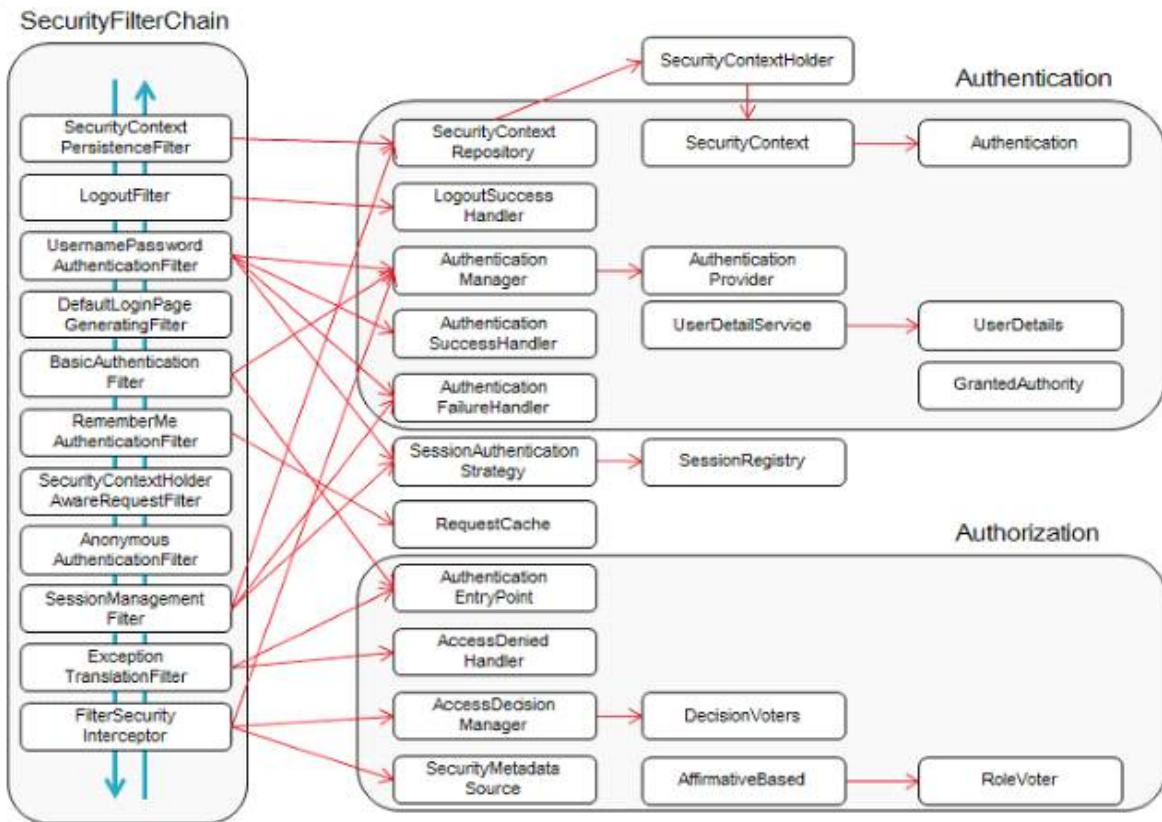


그림 2. SecurityFilterChain 필터 구성도

Spring Security는 DelegatingFilterProxy라는 필터를 만들어 메인 Filter Chain에 끼워넣고 그 아래 다시 SecurityFilterChain 그룹을 등록한다. SecurityFilterChain는 기본적으로 순서가 있는 Security Filter들을 제공하고, Spring Security가 제공하는 Filter를 구현한게 아니라면 필터의 순서를 정해 줘야 한다. 해당 프로젝트에서는 Spring Security를 이용해서 Oauth 카카오, 네이버, 구글 로그인과 Bearer

Auth(JWT)방식으로 로그인을 구현 및 인증 인가를 진행하였다.

```
@Override
protected void configure(HttpSecurity http) throws Exception{

    http.addFilterBefore(new SecurityFilter(), SecurityContextPersistenceFilter.class);
    http.csrf().disable();
    http.sessionManagement().sessionCreationPolicy(SessionCreationPolicy.STATELESS) // 세션사용 x
    .and() HttpSecurity
    .addFilter(corsFilter) // @CrossOrigin 인증x / 시큐리티 필터에 등록 인증o
    .httpBasic().disable() // bearer 인증 방식 사용하겠다
    .formLogin().disable()
    .addFilter(new JwtAuthentication(authenticationManager()))
    .addFilter(new JwtAuthorizationFilter(authenticationManager(), userRepository))
    .authorizeHttpRequests() AuthorizeHttpRequestsConfigurer<...>.AuthorizationManagerRequestMatcherRegistry
    .antMatchers(antPatterns: "/" ,"/auth/**" , "/js/**" , "/css/**" , "/image/**", "/board/**", "/oauth2/**" , "/co
    .permitAll() AuthorizeHttpRequestsConfigurer<...>.AuthorizationManagerRequestMatcherRegistry
    .anyRequest() AuthorizeHttpRequestsConfigurer<...>.AuthorizedUrl
    .authenticated() AuthorizeHttpRequestsConfigurer<...>.AuthorizationManagerRequestMatcherRegistry
    .and().oauth2Login() OAuth2LoginConfigurer<HttpSecurity>
    .successHandler(oauth2SuccessHandler).userInfoEndpoint().userService(principalOauth2UserService);
}
```

그림 3. SecurityConfig

SpringSecurity는 SecurityConfig에서 설정 할 수 있다. 해당 코드에서 세션 사용을 금지, 필터 순서 지정, 필터 활성화, 특정 URL에 대한 인증처리, 인증 및 인가 처리를 진행하였다.

```
@Bean
public CorsFilter corsFilter() {
    UrlBasedCorsConfigurationSource source = new UrlBasedCorsConfigurationSource();
    CorsConfiguration config = new CorsConfiguration();
    config.addAllowedOrigin(CorsConfiguration.ALL); // 모든 ip에 응답을 허용하겠다
    config.addAllowedHeader(CorsConfiguration.ALL); // 모든 Header에 응답을 허용하겠다
    config.addAllowedMethod(CorsConfiguration.ALL); // 모든 post , get , put , delete , patch 요청을 허용하겠다.
    source.registerCorsConfiguration( pattern: "**", config);
    return new CorsFilter(source);
}
```

그림 4. CorsFilter

먼저 해당 프로젝트는 React와 SpringBoot를 같이 사용하기 때문에 Cors Filter를 적용해줬다. Cors는 Cross-origin resource sharing의 약자로서, 특정 헤더를 통해 브라우저에게 Origin에서 실행되고 있는 웹 애플리케이션이 Cross-Origin에 리소스에 접근할 수 있는 권한이 있는지 없는지 확인하는 방침이다. 그림4를 통해 CORS 보안 정책을 허용하였다. 다음으로 Spring Security에는 UsernamePasswordAuthenticationFilter가 있는데 사용자가 /login 요청시 username, password를 전송하면 위 필터가 동작한다. 해당 프로젝트는 Bearer Auth(JWT)방식으로 로그인 및 인증, 인가 처리하기 때문에 JwtAuthenticationFilter에서 UsernamePasswordAuthenticationFilter 및 토큰 처리를 해줬다.

```

public class JwtAuthentication extends UsernamePasswordAuthenticationFilter {

private final AuthenticationManager authenticationManager;

// login 요청을 하면 로그인 시도를 위해서 실행되는 함수
@Override
public Authentication attemptAuthentication(HttpServletRequest request, HttpServletResponse response) throws AuthenticationException {

try{
    ObjectMapper om = new ObjectMapper();
    User user = om.readValue(request.getInputStream(), User.class);
    UsernamePasswordAuthenticationToken authenticationToken =
        new UsernamePasswordAuthenticationToken(user.getUsername(), user.getPassword());

    Authentication authentication = authenticationManager.authenticate(authenticationToken);
    //authentication 객체가 session 영역에 저장할 해야하고 그 방법이 return 해 주는것
    return authentication;

}catch (IOException e){
    e.printStackTrace();
}

return null;
}
}

```

그림 5 JwtAuthenticationFilter

전달 받은 username, password로 로그인 시도를 해보고 authenticationManager로 로그인 시도를 하면 loadUserByUsername() 함수가 실행되고 PrincipalDetails를 세션에 담고 JWT토큰을 만들어서 응답해주면 된다.

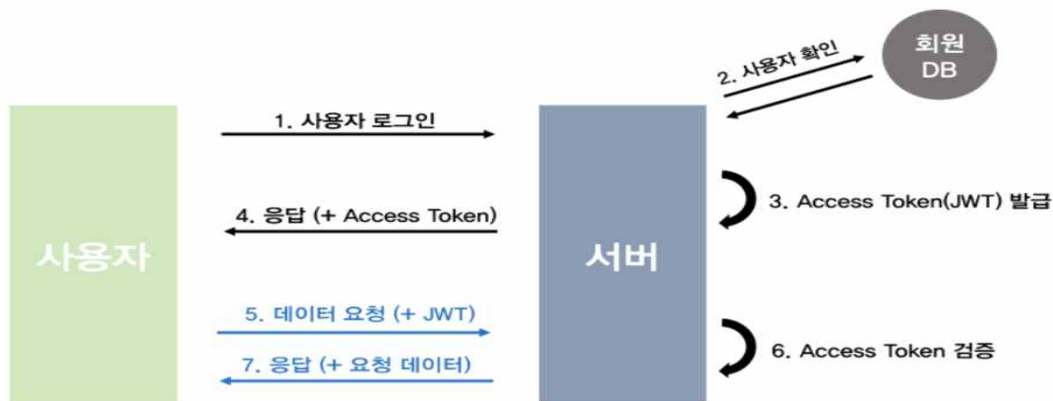


그림 6 토큰 인증 방식 순서

그림6 에서 토큰 인증 방식 순서에 대해서 설명하고 있고 JWT는 웹표준(RFC 7519)으로서 두 개체에서 JSON 객체를 사용하여 가볍고 자가수용적인 방식으로 정보를 안전성 있게 전달하고, 사용자에게 대한 속성을 저장하는 Claim기반의 Web Token이다. 서버측에서 사용자의 세션을 유지하지 사용자가 요청을 했을때 토큰만 확인하면 되기 때문에 세션 관리가 필요 없어서 서버 자원을 많이 아낄 수 있다. JWT는 헤더(header), 내용(payload), 서명(signature) 3부분으로 이루어지며, JSON 형태인 각 부분은 Base64로 인코딩 되어 표현된다.

```

@Override
protected void successfulAuthentication(HttpServletRequest request, HttpServletResponse response,
    FilterChain chain, Authentication authResult) throws IOException,
    PrincipalDetail principalDetail = (PrincipalDetail) authResult.getPrincipal();
    //HMAC Hash암호 방식
    String jwtToken = JWT.create()
        .withSubject(principalDetail.getUsername())
        .withExpiresAt(new Date(System.currentTimeMillis()+60000*10)) //만료시간 10분
        .withClaim( name: "username", principalDetail.getUsername())
        .withClaim( name: "password", principalDetail.getPassword())
        .sign(Algorithm.HMAC512("SJY"));
    response.setHeader( name: "Authorization", value: "Bearer "+jwtToken);
    response.setContentType("application/json; charset=utf-8");
}

```

그림 7 JWT 토큰 생성

JwtAuthenticationFilter에서 attemptAuthentication 실행 후 인증이 정상적으로 되었으면 successfulAuthentication 함수가 실행되기 때문에 JWT 토큰을 만들어서 request 요청한 사용자에게 JWT 토큰을 response 해주면 된다. java-jwt 라이브러리를 이용하여 JWT토큰을 만들어주었고 HMAC(Keyed-Hashed Message Authentication Code)-SHA512 알고리즘을 이용하여 무결성을 보장하였다. 이제 사용자가 로그인 시 Web 로컬 저장소에 Authorization이라는 키에 토큰이 저장되어 있다. Spring Security에서 권한이나 인증이 필요한 특정 주소를 요청하면 BasicAuthenticationFilter를 무조건 거치게 되어있는데 해당 필터를 JwtAuthorizationFilter로 만들어줬다.

```

@Override
protected void doFilterInternal(HttpServletRequest request, HttpServletResponse response, FilterChain chain) throws IOException
    String jwtHeader = request.getHeader( name: "Authorization");

    // header가 있는지 확인
    if(jwtHeader == null || !jwtHeader.startsWith(("Bearer"))){
        chain.doFilter(request, response);
        return;
    }
    String jwtToken = request.getHeader( name: "Authorization").replace( target "Bearer ", replacement: "");
    String username = JWT.require(Algorithm.HMAC512("SJY")).build().verify(jwtToken)
        .getClaim( name: "username").asString();
    if(username != null){
        User userEntity = userRepository.findByUsername(username);
        PrincipalDetail principalDetail = new PrincipalDetail(userEntity);
        //JWT토큰 서명을 통해서 서명이 정상이면 Authentication 객체를 만들어준다
        Authentication authentication = new UsernamePasswordAuthenticationToken(principalDetail, credentials: null, principalDetail);
        //강제로 시큐리티의 세션에 접근하여 Authentication 객체를 저장
        SecurityContextHolder.getContext().setAuthentication(authentication);
        chain.doFilter(request, response);
    }
}

```

그림 8 JwtAuthorizationFilter

해당 필터에서는 header가 있는지 확인하고 해당 토큰의 시작이름을 Bearer로 설정해줬기 때문에 조건식이 맞다면 JWT토큰 서명을 이용해서 서명이 정상이면 Authentication 객체를 만들어줘서 백엔드에서 인증 해주는 방식으로 만들었다.

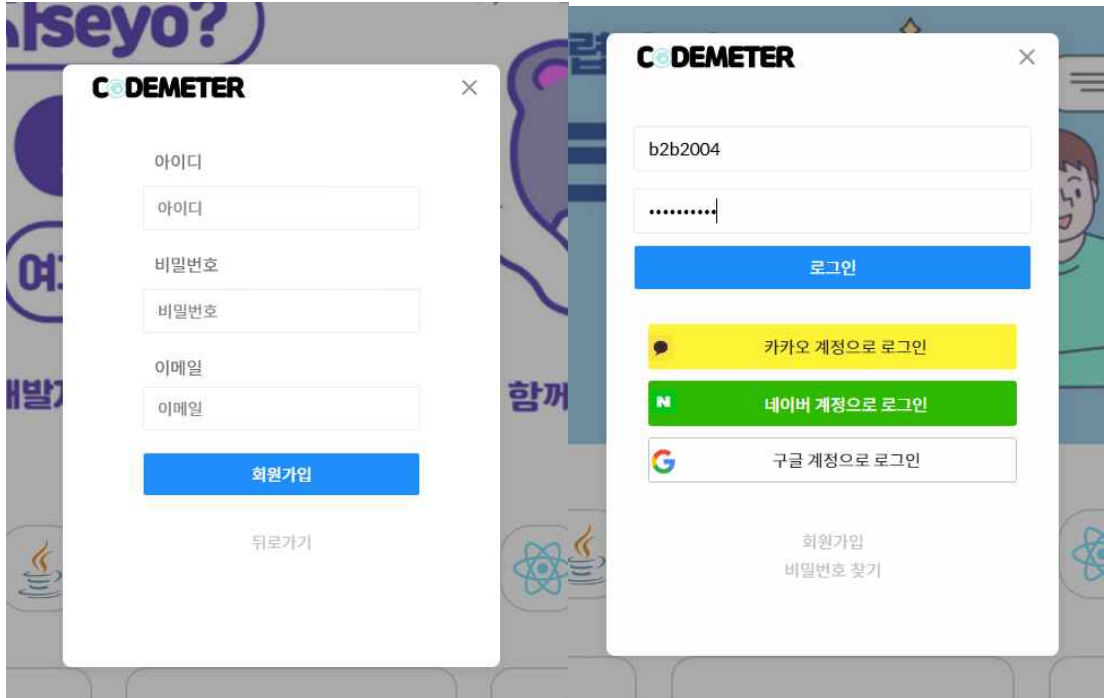


그림 9 회원가입

그림 10 로그인



그림 11 Web JWT Key 생성 확인

다음은 Spring Security를 이용한 OAuth2.0로 로그인을 구현했다. 카카오, 구글, 네이버 3곳의 소셜로그인을 구현하였고 동작 방식이 거의 비슷하기 때문에 카카오 소셜로그인을 예시로 들겠다. 프론트 프레임워크인 React.js 에서 각각 소셜로그인 버튼을 누르게 되면 `http://localhost:8000/oauth2/authorization/kakao(google/naver)` 로 이동하게 되고

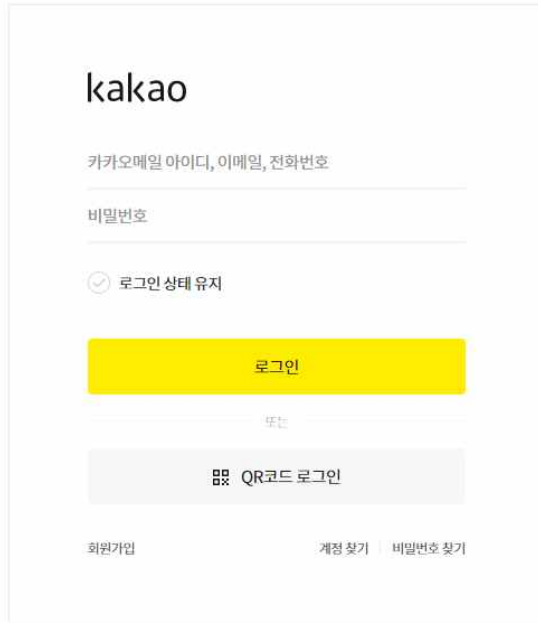


그림 12 카카오 로그인 페이지

그림12와 같은 각 소셜창 로그인 페이지로 이동하게 된다. 미리 각각 회사에서 제공하는 Redirect URI를 설정해두면 기존에 회원이었던 사용자가 로그인하게 되면 (아래로 각각 회사를 예시로 카카오로 하겠다) 카카오에서 인증 및 동의 요청이 되고 동의 한다면 앱에 등록된 Redirect URI로 토큰 발급 요청이 오고 토큰이 발급되면 카카오 로그인이 되어 사용자 정보를 가져올 수 있게 된다.

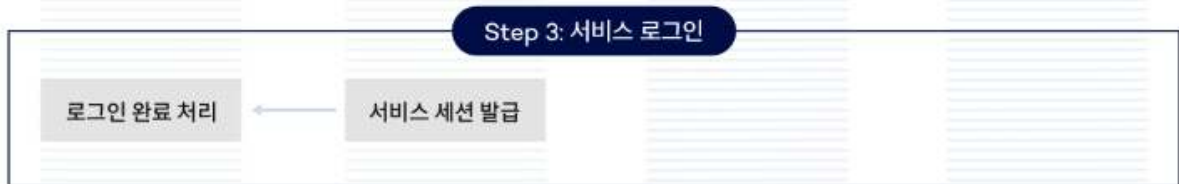
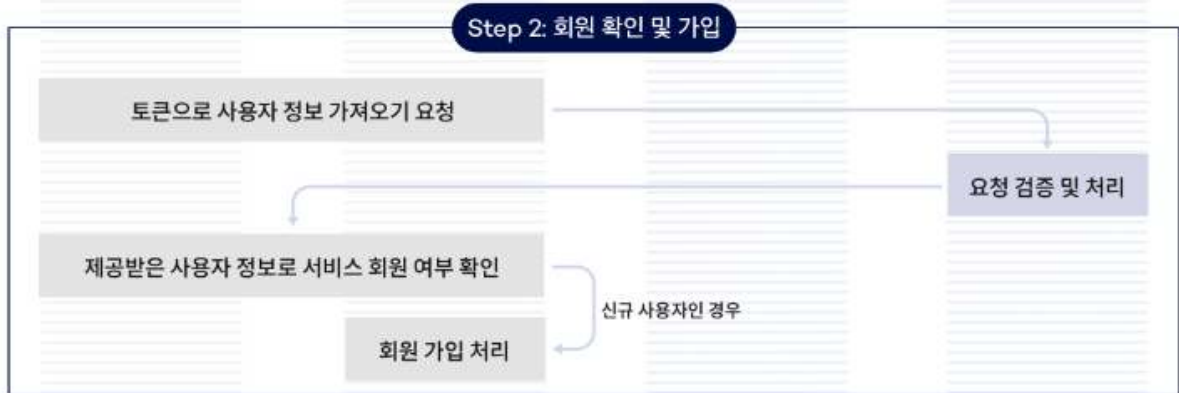
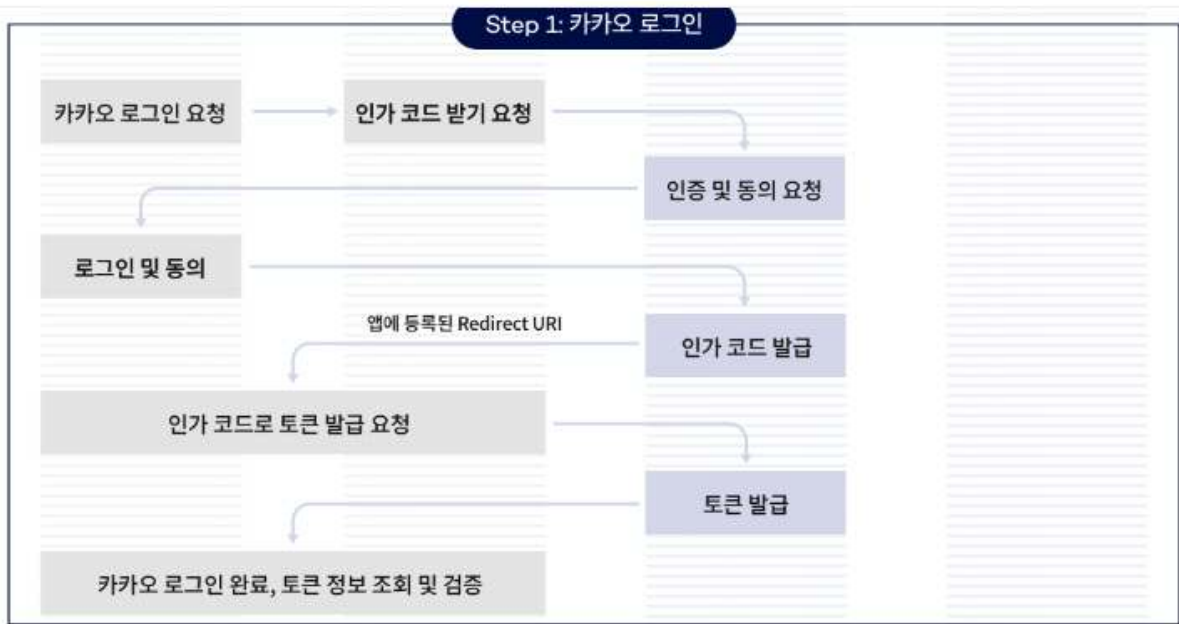


그림 13 (주)카카오 제공

```

SecurityFilter 실행
카카오 로그인 요청
소셜로그인 성공!
Name= null
email= b2b2004@nate.com
ProviderId= 2224409419
Provider= kakao
  
```

그림 14 성공 시 사용자 정보 가져옴

```

@Override
public OAuth2User loadUser(OAuth2UserRequest userRequest) throws OAuth2AuthenticationException {

    OAuth2User oAuth2User = super.loadUser(userRequest);
    OAuth2UserInfo oAuth2UserInfo = null;

    if (userRequest.getClientRegistration().getRegistrationId().equals("google")) {
        System.out.println("구글 로그인 요청");
        oAuth2UserInfo = new GoogleUserInfo(oAuth2User.getAttributes());
    }
    else if (userRequest.getClientRegistration().getRegistrationId().equals("naver")){
        System.out.println("네이버 로그인 요청");
        oAuth2UserInfo = new NaverUserInfo((Map)oAuth2User.getAttributes().get("response"));
    }
    else if (userRequest.getClientRegistration().getRegistrationId().equals("kakao")){
        System.out.println("카카오 로그인 요청");
        oAuth2UserInfo = new KakaoUserInfo(oAuth2User.getAttributes());
    }else {
        System.out.println("구글 네이버 카카오만 지원");
    }
}

```

그림 15 PrincipalOAuth2Service 내부 loadUser

그림15와 같이 PrincipalOAuth2Service 내부에 loadUser에서 각각 소셜로그인 사이트에서 인가를 받고 토큰을 이용해서 사용자 정보에 접근하여 그림14와 같은 정보를 가져오고 해당 정보를 이용하여 해당 User 테이블에 맞춰서 데이터베이스에 저장해주고 기존과 같은 방식으로 JWT Token을 발급하여 사용자 편의성을 높였다. Spring Security의 OAuth는 provider를 제공하지 않기 때문에 따로 만들었다.

3.2.2 기타 로그인 관련 기능

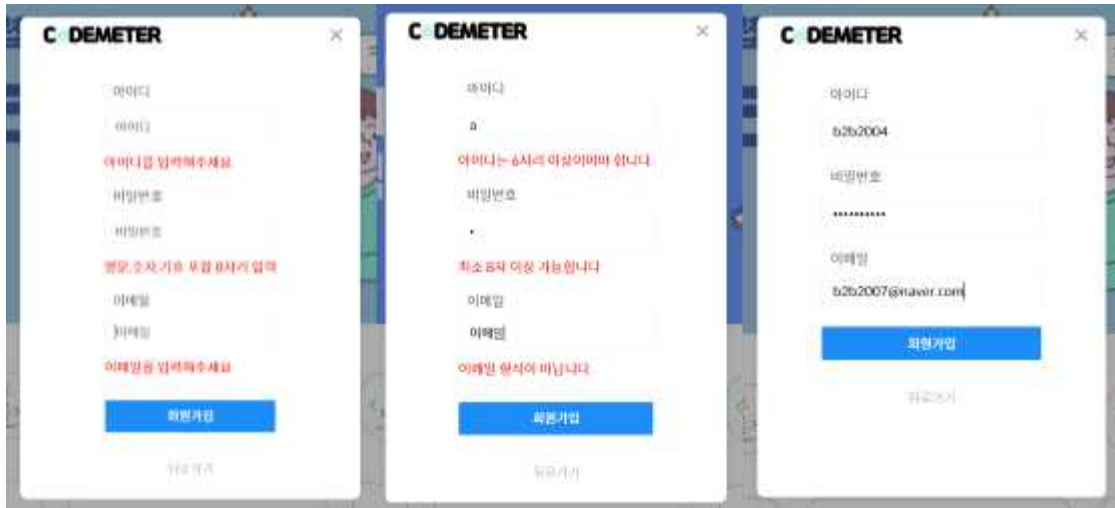


그림 16 회원가입 정규식

```
const formSchema = yup.object({
  username: yup
    .string()
    .required('아이디를 입력해주세요. ')
    .max(12, '아이디는 12자리 이하여야 합니다. ')
    .min(6, '아이디는 6자리 이상이어야 합니다. '),
  email: yup
    .string()
    .required('이메일을 입력해주세요')
    .email('이메일 형식이 아닙니다. '),
  password: yup
    .string()
    .required('영문,숫자,기호 포함 8자리 입력')
    .min(8, '최소 8자 이상 가능합니다')
    .max(15, '최대 15자 까지만 가능합니다')
    .matches(
      /^(?=.*[a-zA-Z])(?=.*\d)(?=.*[!@#$%^&*+=-]).{8,16}$/,
      '영문 숫자 특수문자 포함 8자리 입력'
    ),
});
```

그림 17 React yup

React yup를 이용하여 유효성을 검증하여 회원가입 시 해당 정규식에 맞지 않으면 회원가입이 진행되지 않게 하였다.

The image shows a web form titled "DEMETER" with a close button (X) in the top right corner. The form contains the following elements:

- A label "등록된 아이디" (Registered ID) above a text input field containing the placeholder text "아이디" (ID).
- A label "등록된 이메일" (Registered Email) above a text input field containing the placeholder text "이메일" (Email).
- A blue button with the text "임시 비밀번호 발송" (Send temporary password).
- A link labeled "뒤로가기" (Go back) centered below the button.

그림 18 비밀번호 찾기

해당 프로젝트에서 비밀번호 찾기 기능은 처음 회원가입 시 입력한 이메일과 아이디를 통해서 사용자가 비밀번호를 잊어버렸을때 두 정보가 일치한다면 해당 이메일로 임시 비밀번호를 보내주고 마이페이지에서 바꿀 수 있게 하는 형식으로 만들었다.

```

@Configuration
public class MailConfig {
    @Bean
    public JavaMailSender javaMailService() {
        JavaMailSenderImpl javaMailSender = new JavaMailSenderImpl();
        javaMailSender.setHost("smtp.naver.com");
        javaMailSender.setUsername("보내는이 이메일(관리자 아이디)");
        javaMailSender.setPassword("보내는이 비밀번호(관리자 패스워드)");
        javaMailSender.setPort(465);
        javaMailSender.setJavaMailProperties(getMailProperties());
        return javaMailSender;
    }

    private Properties getMailProperties() {
        Properties properties = new Properties();
        properties.setProperty("mail.transport.protocol", "smtp");
        properties.setProperty("mail.smtp.auth", "true");
        properties.setProperty("mail.smtp.starttls.enable", "true");
        properties.setProperty("mail.debug", "true");
        properties.setProperty("mail.smtp.ssl.trust", "smtp.naver.com");
        properties.setProperty("mail.smtp.ssl.enable", "true");
        return properties;
    }
}

```

그림 19 MailConfig

해당 임시비밀번호 메일 보내는 기능은 JavaMailSender 라이브러리를 이용하였다.
(해당 기능은 네이버만 가능하게 구현했다.)

```

public String getTempPassword() {
    char[] charSet = new char[]{'0', '1', '2', '3', '4', '5', '6', '7', '8', '9', 'A',
        'G', 'H', 'I', 'J', 'K', 'L', 'M', 'N', 'O', 'P', 'Q', 'R', 'S', 'T', 'U',
        'V', 'W', 'X', 'Y', 'Z'};

    String str = "";

    int idx = 0;
    for (int i = 0; i < 10; i++) {
        idx = (int) (charSet.length * Math.random());
        str += charSet[idx];
    }
    return str;
}

public void mailSend(MailDto mailDto){
    SimpleMailMessage message = new SimpleMailMessage();
    message.setTo(mailDto.getAddress());
    message.setFrom(FROM_ADDRESS);
    message.setSubject(mailDto.getTitle());
    message.setText(mailDto.getMessage());
    mailSender.send(message);
}
}

```

그림 20 MailService 일부 코드

먼저 아이디와 이메일을 대조하여 데이터베이스에 저장된 사용자가 맞는지 확인하고 맞다면 데이터베이스에 그림20에서 보이는 형식으로 임시 비밀번호를 생성하여 해당 사용자 테이블에 비밀번호에 덮어 씌우고 해당 임시 비밀번호를 해당 사용자 이메일에 보내주는 식으로 구현하였다.

☆ b2b2004님의 Codmeter 임시비밀번호 안내 이메일입니다. 

보낸사람  <b2b2007@naver.com>
 받는사람 <b2b2007@naver.com>

안녕하세요. Codmeter 임시비밀번호 안내 관련 이메일 입니다.[b2b2004]님의 임시 비밀번호는 A5XEKXT8YZ 입니다.

그림 21 임시 비밀번호를 받게 된 사용자의 메일

임시 비밀번호를 받은 사용자는 내 정보에서 해당 비밀번호를 입력하고 비밀번호로 변경이 가능하다.

Codmeter에서 사용되는 비밀번호입니다.

비밀번호 변경하기

그림 22 내 정보 비밀번호 변경

3.2.3 웹 서버 및 DB

웹 프론트엔드는 React.js를 사용하여 제작하였고 질문게시판, 공모전게시판, 프로젝트 및 스터디원 모집 페이지, 마이페이지 등을 만들어서 사용자 편의를 제공하였다.

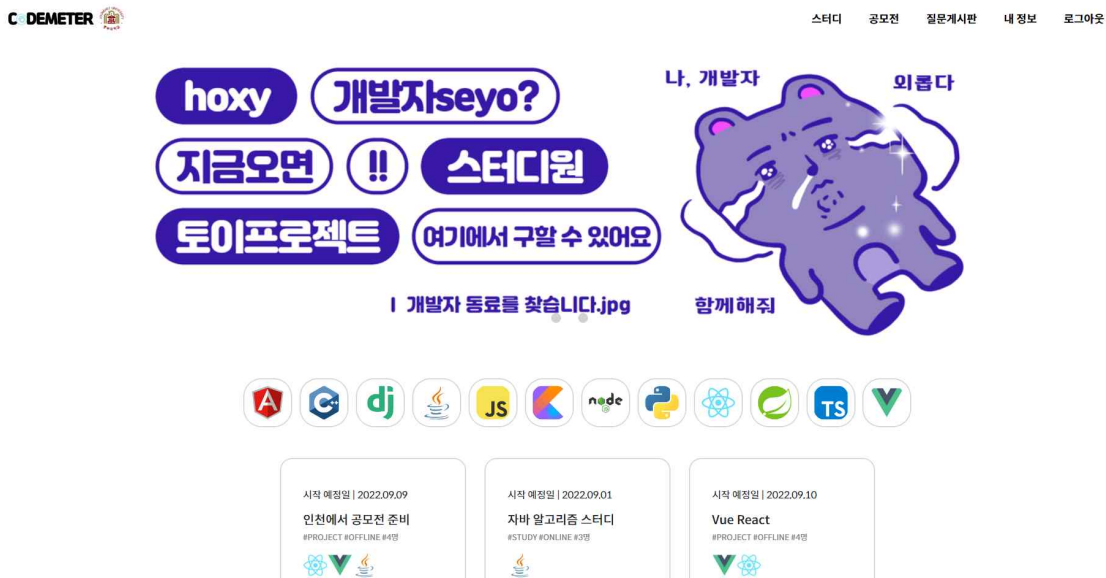


그림 23 메인페이지

메인페이지에서는 스터디/프로젝트 모집창의 최근에 올라온 정보들을 제공하고 사용자가 공부하거나 프로젝트하고 싶은 언어를 이미지로 제공하여 해당 이미지를 클릭하면 관련된 언어를 스터디하거나 프로젝트를 하는 모집창 카드를 제공하였다.

Q&A

질문 게시판

내용을 입력하세요

등록

안녕하세요 CodeMeter에 오신걸 환영합니다.

b2b2005

2022.10.19. 21:49:02

안녕하세요 잘 부탁드립니다.

b2b2006

2022.10.19. 21:49:30

내용을 입력하세요

등록

댓글달기 ^

그림 24 질문게시판

사용자가 모르는걸 물어보거나 관리자에게 물어보거나 어떤 주제로든 활용 할 수 있는 질문게시판을 만들었고 댓글창도 제작하여 활용성을 높였다. 해당 유저가 쓴 게시물이 일 경우에만 수정,삭제 할 수 있게 만들었다.

글쓰기



공모전명	접수기간	조회수
<p>제목: 네트워크 지능화를 위한 인공지능 해커톤 분야: 기획/아이디어, 웹/모바일/IT, 게임/소프트웨어, 과학/공학, 예체능/미술/음악 주최: 송파대학교</p>	-2022.09.30	9
<p>제목: 2022 문화콘텐츠 공모전 분야: 기획/아이디어, 웹/모바일/IT, 게임/소프트웨어, 과학/공학, 예체능/미술/음악 주최: 송파대학교</p>	-2022.10.01	6

그림 25 공모전 게시판 메인화면

스터디/프로젝트와 더불어서 개발자 모임을 통해서 나갈 수 있는 개발 공모전이나 알고리즘 대회 및 아이디어 공모전을 사용자들에게 제공할 수 있게 제작하였다. 해당 페이지는 사용자가 형식에 맞는 공모전 정보들을 모두 입력하면 공모전을 올릴 수 있게 구성하였고 공모전 메인페이지에는 인기 있는 글(조회수)를 통하여 인기가 높은 공모전부터 슬라이드 형식으로 제일 앞에 구성되게 하였다. 그림 25가 공모전 게시판 메인페이지이고 해당 이미지를 클릭하거나 정보를 클릭하면 상세페이지로 이동한다.

recruit

프로젝트 or 스터디

글쓰기

🔥 인기프로젝트 🔥

<p>시작 예정일 2022.09.16</p> <p>React + Spring 웹 개발</p> <p>#PROJECT #ONLINE #3명</p>  <p>bbb</p> <p>27</p>	<p>시작 예정일 2022.09.09</p> <p>정처리 스터디 모임</p> <p>#STUDY #OFFLINE #3명</p>  <p>aaa</p> <p>12</p>	<p>시작 예정일 2022.09.08</p> <p>부산에서 Spring 스터디</p> <p>#STUDY #ONLINE #4명</p>  <p>ccc</p> <p>8</p>
--	--	--

🌟 프로젝트 / 스터디 🌟

<p>시작 예정일 2022.09.09</p> <p>중부대학교 스터디 모임</p> <p>#STUDY #OFFLINE #3명</p>	<p>시작 예정일 2022.09.09</p> <p>정처리 스터디 모임</p> <p>#STUDY #OFFLINE #3명</p>	<p>시작 예정일 2022.09.16</p> <p>React + Spring 웹 개발</p> <p>#PROJECT #ONLINE #3명</p>
--	--	--

그림 26 프로젝트/스터디 모집창

프로젝트/스터디 모집창은 이 웹 프로젝트의 메인창으로써 사용자들이 자유롭게 같이 프로젝트나 스터디를 할 사람을 모집하고 지원하는 페이지이며 그에 그치지 않고 프로젝트/스터디를 할때 도움이되는 기능들을 제공하여 사용자들이 모임을 할때나 회의를 할때 더욱 더 편하게 사용 할 수 있게 만들었다.

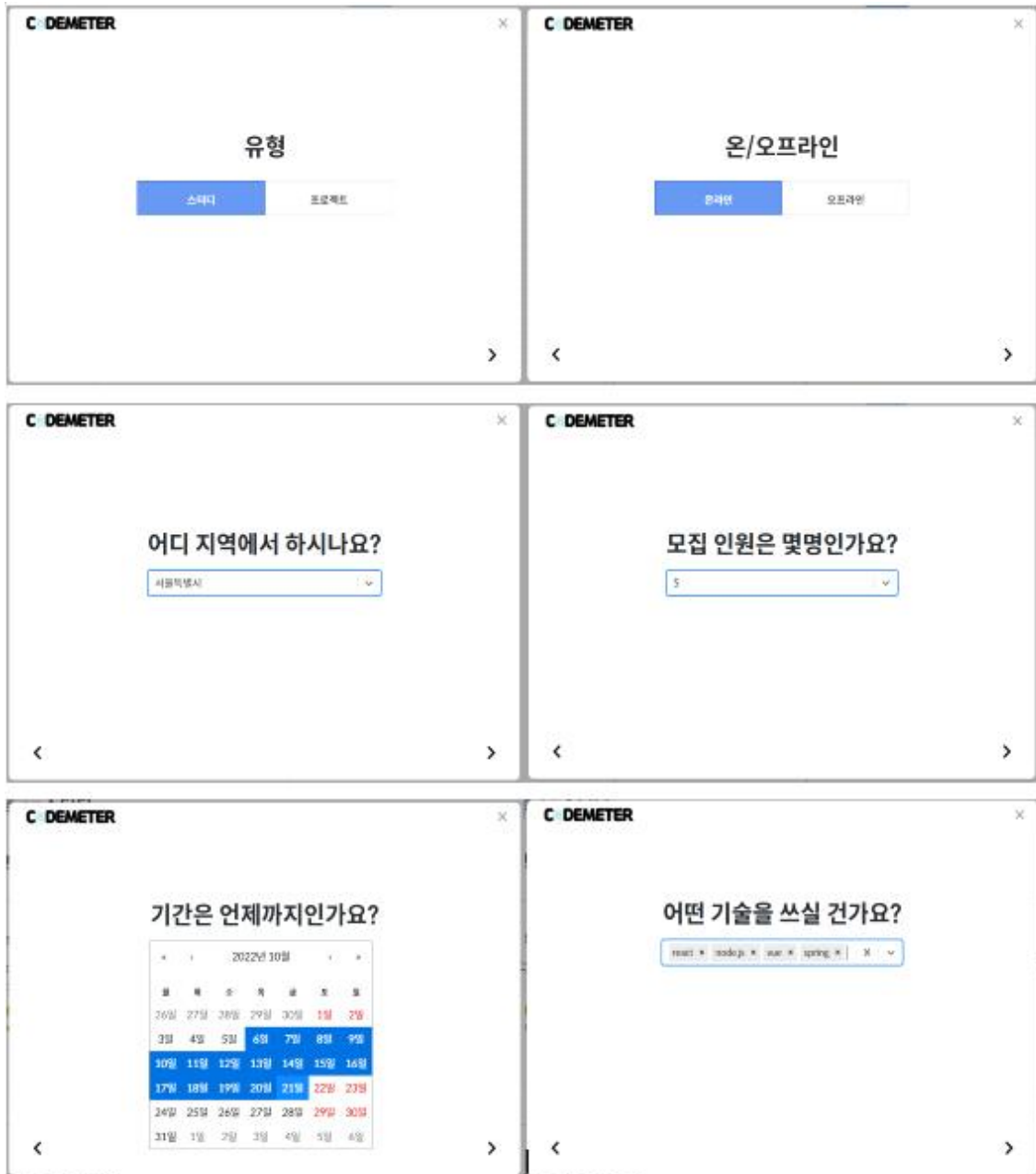


그림 27 스터디/프로젝트 글쓰기 모달창

처음 프로젝트나 스터디 모집 시 정확한 정보를 제공하고 모집하는 팀장에게도 쉽게 모집창을 만들 수 있게 제공하기 위해 모달창을 이용해서 정보를 모두 입력하면 자동으로 프로젝트 모집창을 만들어 주는 방식으로 만들었다.(그림27 에서는 나오지 않지만 제목,내용을 입력해 주는 모달창도 존재한다.)

```

const SopBoardStepSlice = createSlice( options: {
  name: "sopBoardStep",
  initialState,
  reducers: {
    nextStep: (state :Draft<State>, action :PayloadAction<any> ) => ({
      ...state,
      currentStep: state.currentStep + 1,
    }),
    previousStep: (state :Draft<State>, action :PayloadAction<any> ) => ({
      ...state,
      currentStep: state.currentStep - 1,
    }),
    clearStep: () => initialState,
    setSignUpUser: (state :Draft<State>, { payload: { key, value } }) => ({
      ...state,
      [key]: value,
    }),
    setModalVisible: (state :Draft<State>, action :PayloadAction<any> ) => ({
      ...state,
      modalVisible: action.payload,
    }),
  },
});

```

그림 28 모달창 redux

해당 모달창은 그림 28과 같이 redux-toolkit을 이용하여 모달창이 이동할 때 마다 Sopboard(스터디/프로젝트 게시판 model)에 정보가 쌓이게 상태 관리를 해주고 마지막에 백엔드서버로 보내주는 방식을 이용하였다. redux-toolkit 중 createSlice, useDispatch, useSelector등을 이용하여 제작하였다.



중부대학교 스터디 모음

b2b2004 | 2022.09.09

상세페이지

공지사항

질문게시판

관리

이 프로젝트에
관심이 있으신가요?

지원 이유
0 / 3

지원하기

모집 구분	STUDY	진행 방식	OFFLINE
모집 지역	서울특별시	시작 예정	2022.09.09
사용 언어	react.spring.node.js	예상 기간	2022.09.26 까지

프로젝트 소개

안녕하세요 스터디 모임 중입니다. 관심 있으신분들은 알려주세요.

그림 29 스터디/프로젝트 모집 상세페이지

프로젝트/스터디 모집창을 만든 팀장을 제외한 사용자가 모집 상세페이지에 들어가면 해당 모집창에 지원 할 수 있는 기능이 그림29 왼쪽에 존재하고 지원 할 때 지원 사유도 적을 수 있게 제작하였다. 프로젝트/스터디 창에 들어가고 싶은 사용자가 지원 이유와 지원하기를 누르면 팀장이 신청 인원을 확인하여 등록하거나 거절하는 방식으로 팀원 모집을 만들었다.

이 프로젝트에
관심이 있으신가요?

신청 대기 중입니다.

그림 30 사용자 지원시 창 변경

깃합 주소

<https://github.com/b2b2004/SJY>

줌 주소

<https://zoomgov.com/ko-ko/mee>

카카오 오픈채팅방 주소

<https://www.kakaocorp.com/pag>

등록

그림 31 관리창 주소등록



b2b2004 | 2022.09.09

- 상세페이지
- 공지사항
- 질문게시판
- 관리

공지사항 등록

팀원 관리

신청 인원

b2b2006

안녕하세요 평소에 react 공부를 열심히 하고 있었는데 저희 학교에서 스터디를 모집한다는 글을 보고 같이 하고 싶어서 지원하게 되었습니다.

등록 하기 취소 하기

그림 32 팀장의 해당 모집페이지 관리창

팀원 관리

b2b2006

탈퇴 처리

신청 인원

그림 33 신청 수락 시 팀원으로 등록

프로젝트/스터디 팀장이 모집 할 수 있는 팀원 수를 모두 채우면 지원 창은 마감 되었다고 뜨고 더 이상 지원 할 수 없게 되며 해당 팀원들은 공지사항 및 질문게시판을 이용 할 수 있는 권한을 얻게 된다. 팀장은 그림31과 같이 관리창에서 github, zoom, kakaoOpen 주소를 등록하여 팀원들에게 공지사항 페이지에서 보게 할 수 있고 회의록, 공지사항, 과제 등을 공지사항으로 등록하여 팀원들과 공

유하게 할 수 있도록 제작하였다.

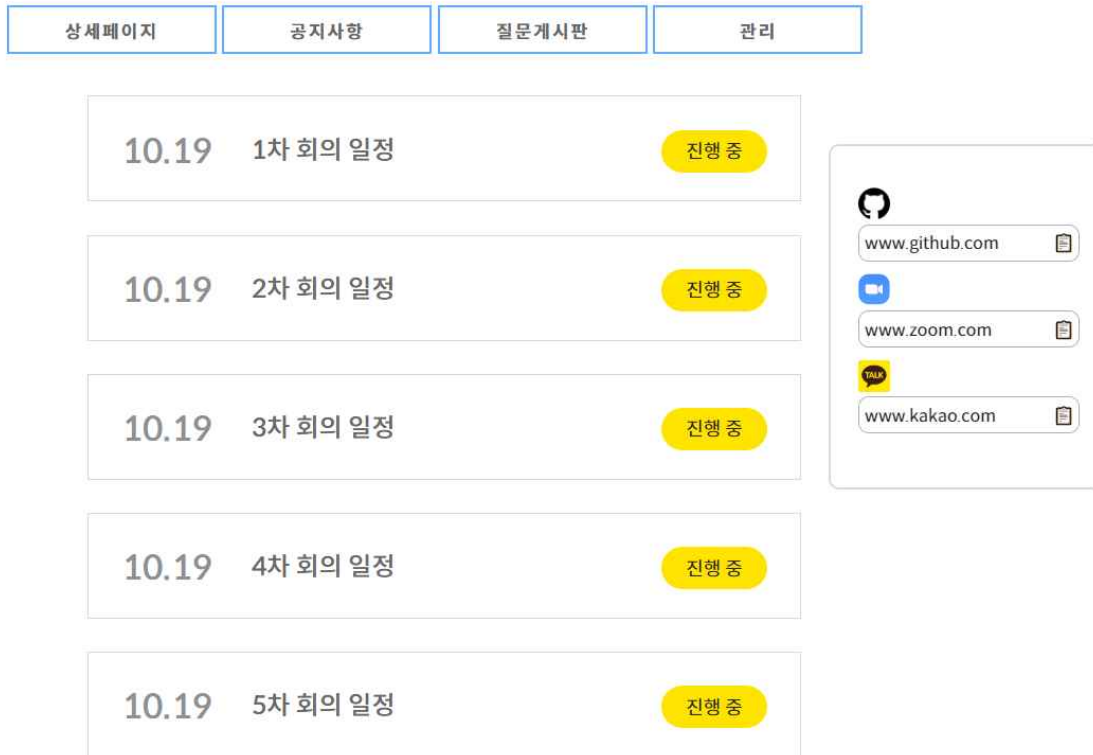


그림 34 스터디/프로젝트 공지사항 페이지

4. 결론

4.1 결론

메인 페이지에서 자신이 관심있는 기술 스택 별로 선택하여 등록되어있는 스터디 또는 프로젝트를 한눈에 확인할 수 있다. 스터디 페이지를 들어가면 인기있는 프로젝트를 따로 볼 수 있으며, 등록된 스터디와 프로젝트를 최신순으로 볼 수 있다. 누구나 코드미터에서 팀원들을 구하기 위해 스터디와 프로젝트를 등록할 수도 있다. 각각 스터디, 프로젝트 별로 상세페이지, 세부일정, 공지사항, 질문게시판, 팀원관리창을 확인할 수 있다.

공모전 페이지를 들어가면 등록된 공모전을 포스터로 한눈에 볼 수 있으며, 목록형으로도 등록된 공모전을 확인할 수 있다. 스터디, 프로젝트와 마찬가지로 누구나 코드미터에 공모전을 등록할 수 있다.

질문게시판 페이지를 들어가면 자유롭게 궁금한 점과 모르는 점을 질문할 수 있으며, 그 질문에 대한 답변 또한 자유롭게 등록할 수 있다.

4.2 기대 효과

이 커뮤니티를 통해 많은 개발자들이 자신들의 공부 방법을 공유하고, 스터디 그룹을

이루어 공부할 수 있는 기회를 연결해준다. 아직 개발을 해보지 못한 사람들도 쉽게 접근할 수 있고, 다양한 팀 프로젝트 정보를 얻을 수 있는 커뮤니티가 될 수 있다. 프로젝트의 수행 과정을 명확하게 명시를 해서 누구나 쉽게 직관적으로 이해할 수 있고 개발과 관련한 고민을 털어놓을 수 있는 창구를 통해서 한층 더 성장할 수 있는 장소가 될 수 있다. 또한 개발 관련 팀 프로젝트를 효율적으로 관리할 수 있게 활용할 수 있다.

5. 별첨

5.1 소스 코드

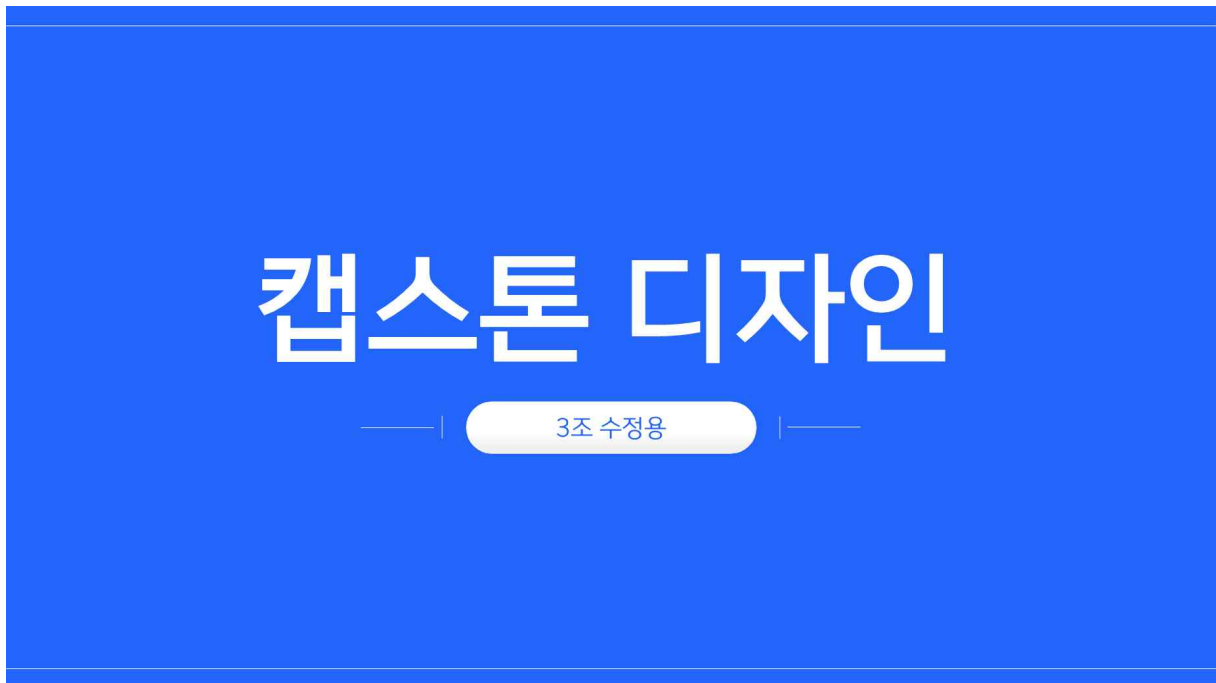
프론트 코드

<https://github.com/b2b2004/SJY-frontend>

백엔드 코드

<https://github.com/b2b2004/SJY-backend>

5.2 발표 자료



1. 팀원 소개 및 주제 선정

3

CONTENTS

001

팀원 소개

주제 선정 개요

002

개발 기간

개발 환경

003

페이지 디자인

004

페이지 소개

005

결론 및 기대 효과

팀원 소개



팀장

이정림

프론트엔드, 백엔드



팀원

권용호

프론트엔드, 백엔드



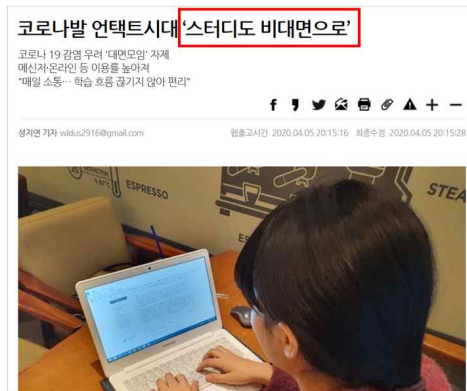
팀원

김수빈

프론트엔드

4

주제 선정 개요



코로나 시대로 인해 여러 제한이 생기면서 비대면 스터디 방식이 확산

코로나 시대에 개발을 공부하는 사람들이 제한없이 효율적으로 모일 수 있는 플랫폼의 필요성을 느꼈다.

스터디원이나 프로젝트 팀원을 구하기 위해서는 중간 다리 역할이 필요하다. 그것이 바로 **코드미터**이다.

코드미터에서는 온라인, 오프라인으로 스터디와 프로젝트를 진행할 수 있으며, 언제든지 궁금한 것이 있으면 질문을 할 수 있다.

다양한 개발환경 중 원하는 주제를 정해 팀원들을 모집하는 일은 쉽지 않다. 그 과정을 보다 간편하게 해결하고 관심분야의 공모전을 제한없이 등록하고 지원할 수 있는 커뮤니티가 필요하다고 생각해 **코드미터**를 시작하게 되었다.

5

2. 개발 기간 / 환경

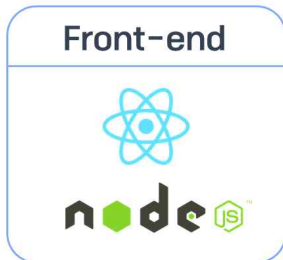
6

개발 기간

	3월	4월	5월	6월	7월	8월	9월	10월
주제선정	■							
개발환경 구축	■	■						
개인 공부		■	■					
개발			■	■	■	■		
마무리 및 최종 점검							■	
PPT, 보고서 제작								■

7

개발 환경



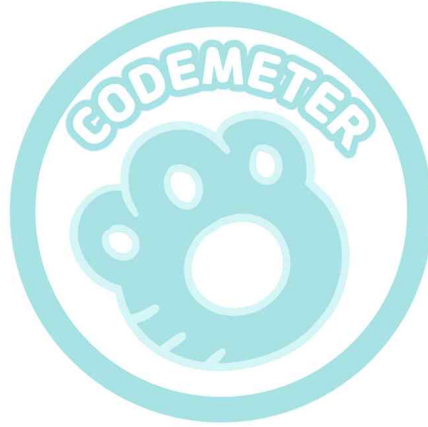
8

3. 페이지 디자인

9

페이지 컨셉

CODEMETER



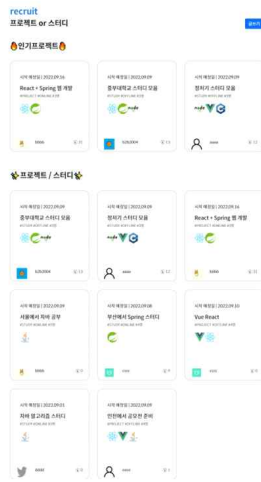
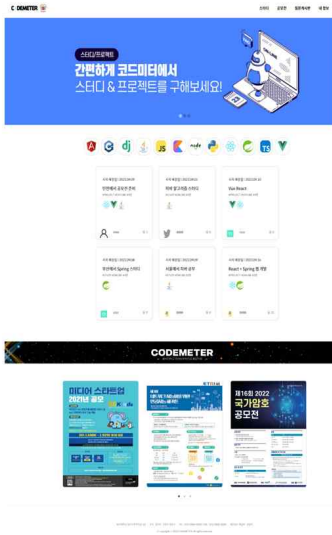
10

와이어 프레임



11

페이지 디자인

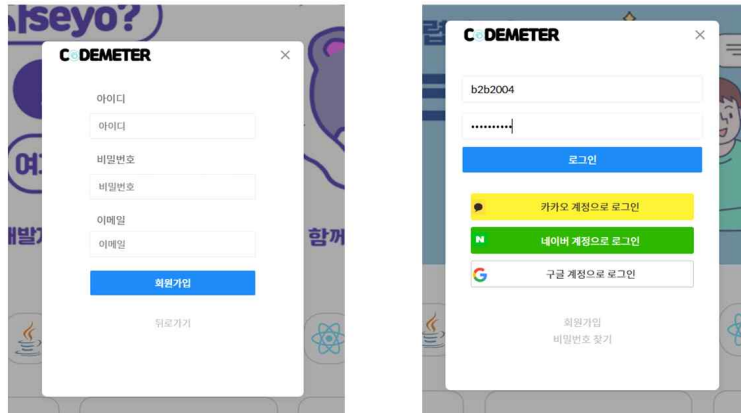


12

4. 페이지 소개

13

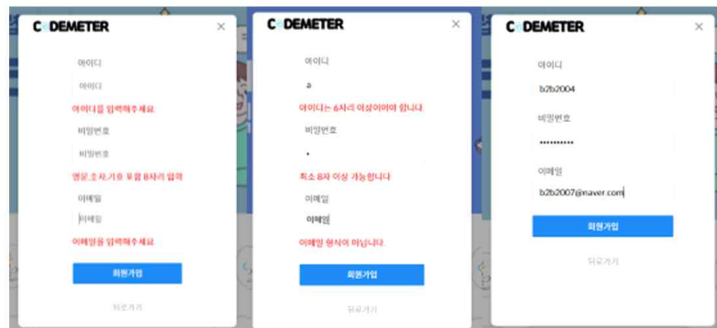
로그인 페이지



Spring Security를 이용한 OAuth2.0로 로그인을 구현
카카오, 구글, 네이버의 소셜 로그인 구현 클릭하면 각각의 소셜 로그인 페이지로 이동

14

회원가입 페이지



회원가입을 할 수 있는 모달창
React yup를 이용하여 유효성을 검증하여 회원가입 시 해당 정규식에 맞지 않으면 회원가입이 진행되지 않게 하였다.

15

비밀번호 찾기 페이지

The screenshot displays three main components of the password recovery process:

- Left Panel (Form):** A 'C. DEMETER' window with fields for '등록된 아이디' (Registered ID) and '등록된 이메일' (Registered Email). A blue button '임시 비밀번호 발송' (Send temporary password) is visible, along with a '뒤로가기' (Back) link.
- Middle Panel (Code):** A JavaMailSender configuration class named 'MailConfig'. It includes a '@Bean' annotation and a 'javaMailService()' method that sets host, username, password, and port. A 'getMailProperties()' method is also present.
- Right Panel (Form):** A confirmation box with a warning message: 'b2b2004님의 Codmeter 임시비밀번호 안내 이메일입니다.' (This is the email for user b2b2004's Codmeter temporary password). Below it, a 'Codmeter에서 사용되는 비밀번호입니다.' (This is the password used in Codmeter) section contains fields for '현재 비밀번호' (Current password), '바꿀 비밀번호' (New password), and '바꿀 비밀번호 확인' (Confirm new password), followed by a '비밀번호 변경하기' (Change password) button.

비밀번호 찾기 기능은 처음 회원가입 시 입력한 이메일과 아이디를 통해서 사용자가 비밀번호를 잊어버렸을 때 두 정보가 일치한다면 해당 이메일로 임시 비밀번호를 보내주고 마이페이지에서 바꿀 수 있게 하는 형식으로 만들었다.

16

메인 페이지

The main page layout includes:

- Navigation Bar:** 'C. DEMETER' logo on the left and '스터디', '공모전', '일문제서만', '내 정보', '로그아웃' on the right.
- Central Banner:** A purple-themed banner with the text 'hoxy 개발자seyo?' and '나, 개발자 외롭다' (I, developer, am lonely). It also says '지금오면 !! 스테디원' (If you come now!! Steady member) and '토이프로젝트' (Toy project). A cartoon cat character is on the right.
- Project Grid:** A row of icons for various technologies: A, G, dj, JS, nade, and TS. Below this are three project cards:
 - 스터디 예정일 | 2022.09.09: 연천에서 공모전 준비 (Preparation for a competition in Yeoncheon)
 - 스터디 예정일 | 2022.09.01: 지바 알고리즘 스테디 (Ziba Algorithm Steady)
 - 스터디 예정일 | 2022.09.10: Vue React (Vue React)

메인 페이지에서는 스테디/프로젝트 모집창의 최근에 올라온 정보들을 제공하고 사용자가 공부하거나 프로젝트하고 싶은 언어를 이미지로 제공하여 해당 이미지를 클릭하면 관련된 언어를 스테디 하거나 프로젝트를 하는 모집 목록을 제공하였다.

17

질문 게시판

Q&A 질문 게시판

등록

b2b2005

안녕하세요 CodeMeter에 오신걸 환영합니다.

2022.10.19. 21:49:02

b2b2006

안녕하세요 잘 부탁드립니다.

2022.10.19. 21:49:30

등록

댓글달기 ^

사용자가 모르는 걸 물어보거나 관리자에게 물어보거나 어떤 주제로든 활용 할 수 있는 질문게시판을 만들었고 댓글창도 제작하여 활용성을 높였다. 해당 유저가 쓴 게시물이 경우에만 수정,삭제 할 수 있게 만들었다.

18

공모전 게시판 메인

글쓰기



해당 페이지는 사용자가 형식에 맞는 공모전 정보를 모두 입력하면 공모전을 올릴 수 있게 구성하였고

공모전 메인 페이지에는 인기 있는 글(조회수)을 통하여 인기가 높은 공모전부터 슬라이드 형식으로 제일 앞에 구성되게 하였다.

공모전명		접수기간	조회수
	<p>제목: 네트워크 지능화를 위한 인공지능 해커톤</p> <p><small>주최: 가천대학교, 웹코퍼레이션, 제임스소프트웨어, 유망기업, 미래능기회(유망)주최, 홍익대학교</small></p>	-2022.09.30	9
	<p>제목: 2022 문화콘텐츠 공모전</p> <p><small>주최: 가천대학교, 웹코퍼레이션, 제임스소프트웨어, 유망기업, 미래능기회(유망)주최, 홍익대학교</small></p>	-2022.10.01	6

19

프로젝트/스터디 모집 창

recruit

프로젝트 or 스터디

윤쓰기

🔥 인기 프로젝트 🔥

시작 예정일 | 2022.09.16

React + Spring 웹 개발

#PROJECT #ONLINE #3명



bbbb 27

시작 예정일 | 2022.09.09

정치기 스터디 모음

#STUDY #OFFLINE #3명



aaaa 12

시작 예정일 | 2022.09.08

부산에서 Spring 스터디

#STUDY #ONLINE #4명



cccc 8

프로젝트/스터디 모집창은 이 웹 프로젝트의 메인 창으로써 사용자들이 자유롭게 같이 프로젝트나 스터디를 할 사람을 모집하고 지원하는 페이지이며

그에 그치지 않고 프로젝트/스터디를 할 때 도움이 되는 기능들을 제공하여 사용자들이 모임을 할 때나 회의를 할 때 더욱 더 편하게 사용할 수 있게 만들었다.

🌟 프로젝트 / 스터디 🌟

시작 예정일 | 2022.09.09

중부대학교 스터디 모음

#STUDY #OFFLINE #3명

시작 예정일 | 2022.09.09

정치기 스터디 모음

#STUDY #OFFLINE #3명

시작 예정일 | 2022.09.16

React + Spring 웹 개발

#PROJECT #ONLINE #3명

20

프로젝트 모집 모달 창

C DEMETER

유형

온라인 오프라인

C DEMETER

온/오프라인

온라인 오프라인

C DEMETER

어디 지역에서 하시나요?

C DEMETER

모집 인원은 몇명인가요?

C DEMETER

기간은 언제까지인가요?

2022년 9월

일	월	화	수	목	금	토	일
26일	27일	28일	29일	30일	1일	2일	3일
4일	5일	6일	7일	8일	9일	10일	11일
12일	13일	14일	15일	16일	17일	18일	19일
20일	21일	22일	23일	24일	25일	26일	27일
28일	29일	30일	31일	1일	2일	3일	4일

C DEMETER

어떤 기술을 쓰실 건가요?

처음 프로젝트나 스터디 모집 시 정확한 정보를 제공하고 모집하는 팀장에게도 쉽게 모집창을 만들 수 있게 제공하기 위해

모달 창을 이용해서 정보를 모두 입력하면 자동으로 프로젝트 모집창을 만들어 주는 방식으로 만들었다.

21

모집 상세페이지

←

중부대학교 스터디 모음

b2b2004 | 2022.09.09

상세페이지 | 공지사항 | 질문게시판 | 관리

모집 구분 STUDY 진행 방식 OFFLINE

모집 지역 서울특별시 시작 예정 2022.09.09

사용 언어 react.spring.node.js 예상 기간 2022.09.26 까지

이 프로젝트에 관심이 있으신가요?

지원 이유 0/3

지원하기

프로젝트 소개

안녕하세요 스터디 모임 중입니다. 관심 있으신분들은 알려주세요.

프로젝트/스터디 모집창을 만든 팀장을 제외한 사용자가 모집 상세 페이지에 들어 가면 해당 모집창에 지원 할 수 있는 기능이 왼쪽에 존재하고 지원 할 때 지원 사유도 적을 수 있게 제작하였다.

프로젝트/스터디 창에 들어가고 싶은 사용자가 지원 이유와 지원하기를 누르면 팀장이 신청 인원을 확인하여 등록하거나 거절하는 방식으로 팀원 모집을 만들었다.

22

신청자 지원 / 관리 페이지

b2b2004 | 2022.09.09

상세페이지 | 공지사항 | 질문게시판 | 관리

공지사항 등록

이 프로젝트에 관심이 있으신가요?

신청 대기 중입니다.

팀원 관리

신청 인원

b2b2006

안녕하세요 평소엔 react 공부를 열심히 하고 있었는데 저희 학교에서 스터디를 모집한다는 글을 보고 같이 하고 싶어서 지원하게 되었습니다.

등록하기 | 취소하기

팀원 관리

b2b2006

탈퇴하기

신청 인원

프로젝트/스터디 팀장이 모집 할 수 있는 팀원 수를 모두 채우면 지원 창은 마감 되었다고 뜨고

더 이상 지원 할 수 없게 되며 해당 팀원들은 공지사항 및 질문게시판을 이용 할 수 있는 권한을 얻게 된다.

23

신청자 지원 / 관리 페이지

상세페이지	공지사항	질문게시판	관리
10.19	1차 회의 일정	진행 중	
10.19	2차 회의 일정	진행 중	
10.19	3차 회의 일정	진행 중	
10.19	4차 회의 일정	진행 중	
10.19	5차 회의 일정	진행 중	

깃헙 주소
<https://github.com/b2b2004/SJY>

줌 주소
<https://zoomgov.com/ko-ko/mee>

카카오 오픈채팅방 주소
<https://www.kakaocorp.com/pag>

등록

팀장은 관리창에서 github, zoom, kakaoOpen 주소를 등록하여 팀원들에게 공지사항 페이지에서 보게 할 수 있고

회의록, 공지사항, 과제 등을 공지사항으로 등록하여 팀원들과 공유하게 할 수 있도록 제작하였다.

24

5. 결론 및 기대효과

25

결론 및 기대효과

결론

메인 페이지에서 자신이 관심있는 기술 스택 별로 선택하여 등록 되어있는 스터디 또는 프로젝트를 한눈에 확인할 수 있다. 스터디 페이지를 들어가면 인기있는 프로젝트를 따로 볼 수 있으며, 등록된 스터디와 프로젝트를 최신순으로 볼 수 있다. 누구나 코드 미터에서 팀원들을 구하기 위해 스터디와 프로젝트를 등록할 수도 있다. 각각 스터디, 프로젝트 별로 상세페이지, 세부일정, 공지사항, 질문게시판, 팀원리창을 확인할 수 있다.

공모전 페이지를 들어가면 등록된 공모전을 포스터로 한눈에 볼 수 있으며, 목록형으로도 등록된 공모전을 확인할 수 있다. 스터디, 프로젝트와 마찬가지로 누구나 코드 미터에 공모전을 등록할 수 있다.

기대효과

이 커뮤니티를 통해 많은 개발자들이 자신들의 공부 방법을 공유하고, 스터디 그룹을 이루어 공부할 수 있는 기회를 연결해준다.

아직 개발을 해보지 못한 사람들도 쉽게 접근할 수 있고, 다양한 팀 프로젝트 정보를 얻을 수 있는 커뮤니티가 될 수 있다.

프로젝트의 수행 과정을 명확하게 명시를 해서 누구나 쉽게 직관적으로 이해할 수 있고 개발과 관련한 고민을 털어놓을 수 있는 창구를 통해서 한층 더 성장할 수 있는 장소가 될 수 있다.

또한 개발 관련 팀 프로젝트를 효율적으로 관리할 수 있게 활용할 수 있다.

26

Q&A

27