

TEAM E : TECHNICAL BACKGROUNDS

WEEK 6

SEOKHYUN BAE
JIWOO CHUNG
GUHO KIM
JINU PARK

AGENDA

CLIENT-SIDE

FIGMA
REACT.JS
VITE

JIWOO CHUNG
JINU PARK

SERVER-SIDE

SPRING BOOT
JPA, JWT
AWS S3, ELB
SELENIUM

GUHO KIM
SEOKHYUN BAE

DATABASE

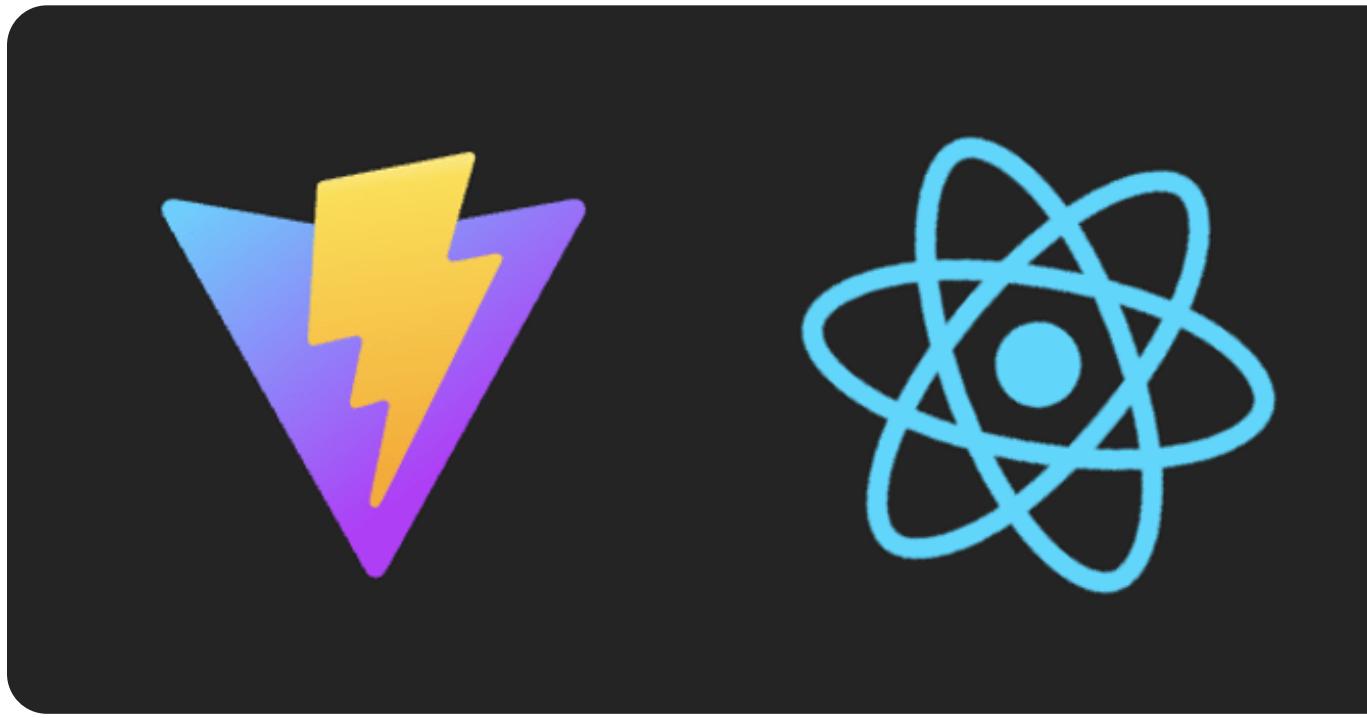
AWS RDS
MySQL

GUHO KIM
SEOKHYUN BAE

CLIENT-SIDE



REACT + VITE



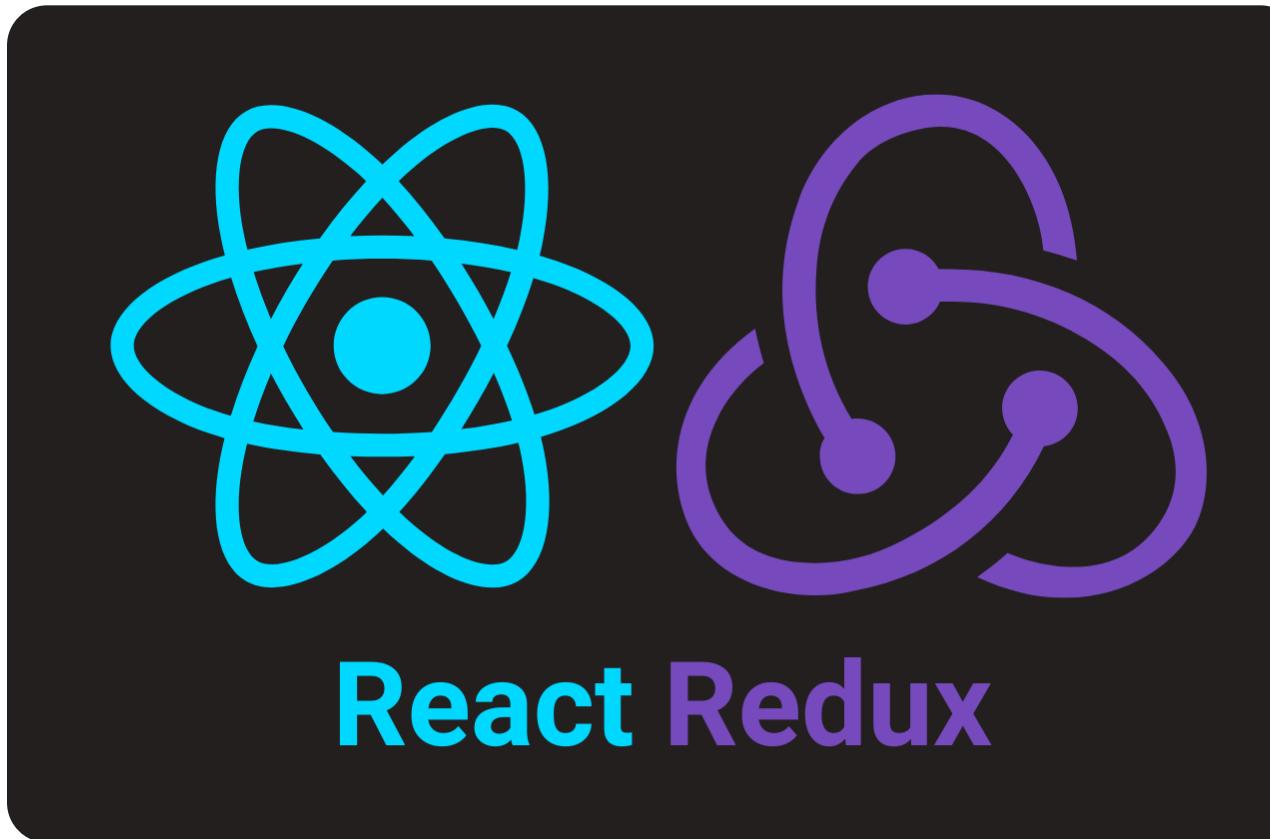
- ➊ Can improve SEO by using VITE SSR
- ➋ Available libraries like REDUX TOOLKIT
- ➌ Provides fast development environment and efficient build process

```
// ssr.js - 서버에서 렌더링을 위한 React 컴포넌트 렌더링
import React from 'react';
import ReactDOMServer from 'react-dom/server';
import App from './App';
import { StaticRouter } from 'react-router-dom/server';

export function render(url) {
  return ReactDOMServer.renderToString(
    <StaticRouter location={url}>
      <App />
    </StaticRouter>
  );
}
```

‘ssr.js’ for implementing server-side rendering.

REACT REDUX TOOLKIT



- 📍 Simplified Setup
- 📍 Efficient State Management
- 📍 Improved Performance

```
import { configureStore } from '@reduxjs/toolkit';
import userReducer from './slices/userSlice';

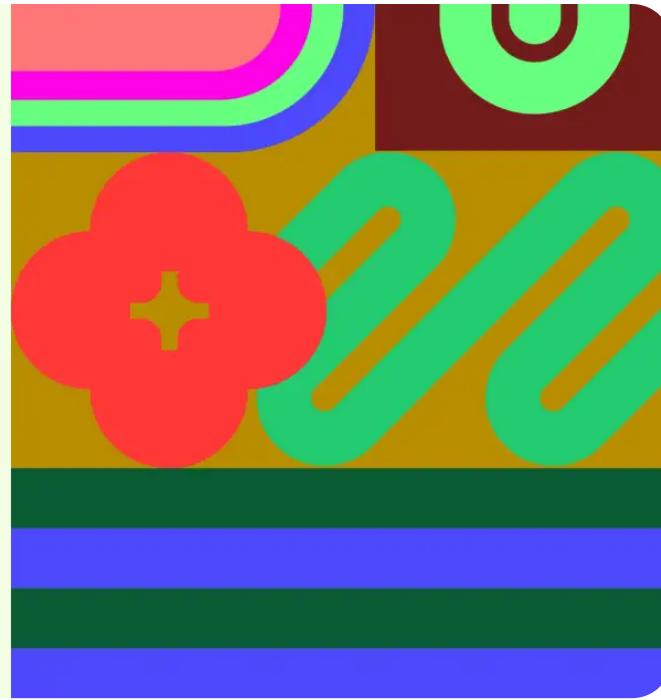
const store = configureStore({
  reducer: {
    user: userReducer,
  },
});

export default store;
```

‘store.js’ for managing user login information with Redux Toolkit.

FIGMA DEVMODE

Figma



- Provides Figma Dev Mode
- Optimal environment for team collaboration
- Efficient management of components and assets

```
color: #000;
font-family: HANAMDAUM;
font-size: 85px;
font-style: normal;
font-weight: 400;
line-height: normal;
letter-spacing: 2.55px;
```

Content

캠퍼스 인싸 되고 싶다면? 나만의 동아리부터 시작해!

Using Dev Mode to implement HTML/CSS code, after designing UI in Figma

The screenshot shows a Java code editor interface with a dark theme. The top bar displays the project name "TeamEApplication" and various icons for navigation and settings. Below the bar, several tabs are visible, including "Club.java", "JWTController.java", "JWTFilter.java" (which is the active tab), "JWTUtil.java", and "TokenBlacklistService.java". The main area contains the source code for the "JWTFilter" class:

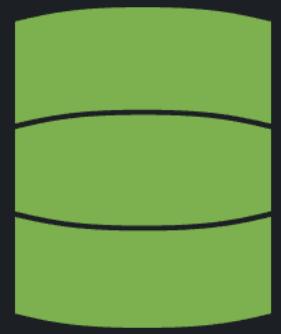
```
17 import java.io.IOException;
18
19 @Log4j2
20 public class JWTFilter extends OncePerRequestFilter {
21
22     private final JWTUtil jwtUtil;
23
24     public JWTFilter(JWTUtil jwtUtil) {
25         this.jwtUtil = jwtUtil;
26     }
27
28     @Override
29     protected void doFilterInternal(HttpServletRequest request,
30                                     HttpServletResponse response,
31                                     FilterChain filterChain) throws ServletException, IOException {
32
33         String authorization= request.getHeader("Authorization");
34         String refreshToken = request.getHeader("Refresh-Token");
35
36         try {
37             if (authorization == null || !authorization.startsWith("Bearer ")) {
38                 filterChain.doFilter(request, response);
39                 return;
40             }
41
42             String token = authorization.split(" ")[1];
43
44             if (jwtUtil.isExpired(token)) {
45                 if (refreshToken != null && jwtUtil.validateToken(refreshToken)) {
46                     String username = jwtUtil.getUsername(refreshToken);
47                     String role = jwtUtil.getRole(refreshToken);
48                     String newAccessToken = jwtUtil.createAccessToken(username, role);
49
50                     response.addHeader("Authorization", "Bearer " + newAccessToken);
51                 } else {
```

On the right side of the code editor, there are status indicators showing 2 errors and 1 warning, along with other icons for file operations like copy, paste, and search.

SERVER-SIDE

SPRING BOOT

JPA



Spring Data

Provides a consistent approach to data access
– relational, non-relational, map-reduce, and beyond.

- 📍 Simplified Database Access
- 📍 Automatic Object Mapping
- 📍 Built-In Query Methods: Derived query

methods simplify data retrieval without needing explicit SQL or JPQL queries.

```
import org.springframework.data.jpa.repository.JpaRepository;

public interface MemberRepository extends JpaRepository<Member, Long> {

    boolean existsByUsername(String username);

    boolean existsByPhoneNumber(String phoneNumber);

    boolean existsByDisplayName(String displayName);

    Member findByUsername(String username);

}
```

Process Interacting with the database using only JPA without JPQL by utilizing derived query methods.

SPRING BOOT SECURITY & JWT



Spring Security

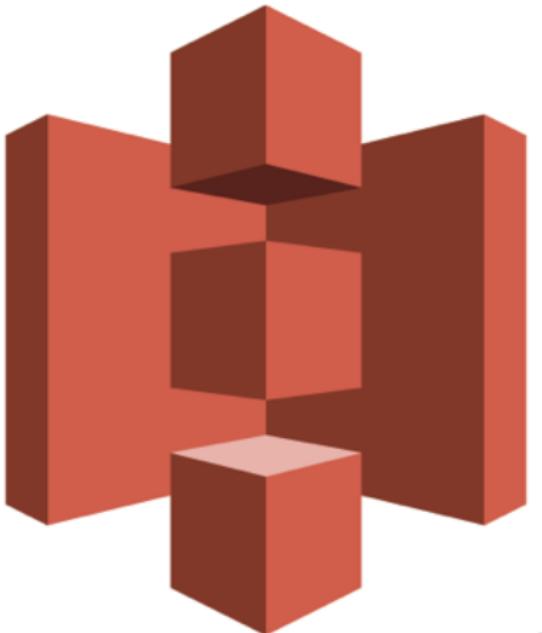
Protects your application with comprehensive and extensible authentication and authorization support.

- 📍 Stateless Authentication : reducing server load and improving scalability
- 📍 Enhanced Security : encryption and signature validation
- 📍 Easier Access Control : simplified role-based access and authorization

```
public String createAccessToken(String username, String role) {  
    return Jwts.builder()  
        .claim("username", username)  
        .claim("role", role)  
        .issuedAt(new Date(System.currentTimeMillis()))  
        .expiration(new Date(System.currentTimeMillis() + accessTokenValidity))  
        .signWith(secretKey)  
        .compact();  
}  
  
public String createRefreshToken(String username, String role) {  
    return Jwts.builder()  
        .claim("username", username)  
        .claim("role", role)  
        .issuedAt(new Date(System.currentTimeMillis()))  
        .expiration(new Date(System.currentTimeMillis() + refreshTokenValidity))  
        .signWith(secretKey)  
        .compact();  
}
```

Improving login security by generating JWT tokens integrated with Spring Security

AWS S3 STORAGE



Amazon S3

- Automatically scales to amount of data
- High Durability & Availability

circlespace/

S3 URI 복사

객체 속성

객체 (0) 정보

C S3 URI 복사 URL 복사 다운로드 열기 삭제 작업 폴더 만들기 업로드

객체는 Amazon S3에 저장되어 있는 기본 엔터티입니다. [Amazon S3 인벤토리](#) 를 사용하여 버킷에 있는 모든 객체의 목록을 얻을 수 있습니다. 다른 사용자가 객체에 액세스할 수 있게 하려면 명시적으로 권한을 부여해야 합니다. [자세히 알아보기](#)

접두사로 객체 찾기 < 1 > 설정

이름 유형 마지막 수정 크기 스토리지 클래스

객체 없음 이 폴더에 객체가 없습니다.

업로드

```
private final S3Service s3Service;

@PostMapping("/api/s3/getPreSignedUrl")
public ResponseEntity<Map<String, String>> getPreSignedUrl(@RequestBody Map<String, String> request) {
    try {
        String path = request.get("path");
        String preSignedUrl = s3Service.createPreSignedUrl(path);
        Map<String, String> response = new HashMap<>();
        response.put("url", preSignedUrl);
        return ResponseEntity.ok(response);
    } catch (Exception e) {
        return ResponseEntity.status(HttpStatus.INTERNAL_SERVER_ERROR).build();
    }
}
```

Created S3 Bucket,
Implemented Presigned-Url Request

SELENIUM CRAWLING



- 📍 Automates browser actions
- 📍 Handles JavaScript-rendered content
- 📍 Supports HTML, CSS selectors for precise element targeting

```
public static void main(String[] args) {
    System.setProperty("webdriver.chrome.driver", "./chromedriver-win64/chromedriver.exe");
    WebDriver driver = new ChromeDriver();

    try (BufferedWriter writer = new BufferedWriter(new FileWriter("output.txt", true))) {
        try {
            // everytime club board url
            String url = "https://everytime.kr/418760";
            driver.get(url);
            System.out.println("Successfully opened the webpage.");
            Thread.sleep(10000);

            List<WebElement> articleLinks = driver.findElements(By.cssSelector("a.article"));
            for (WebElement articleLink : articleLinks) {
                String linkHref = articleLink.getAttribute("href");
                if (!linkHref.startsWith(url)) continue;

                try {
                    String articleUrl = articleLink.getAttribute("href");
                    driver.get(articleUrl);
                    System.out.println("Navigated to article URL: " + articleUrl);
                    Thread.sleep(3000);

                    List<WebElement> paragraphTitle = driver.findElements(By.cssSelector("h2.large"));
                    List<WebElement> paragraphBody = driver.findElements(By.cssSelector("p.large"));

                    String title = paragraphTitle.isEmpty() ? "No title found" :
                        paragraphTitle.get(0).getText().replaceAll("[^\\p{L}\\p{N}\\p{P}]\\s", "");

                    StringBuilder body = new StringBuilder();
                    for (WebElement paragraph : paragraphBody) {
                        String filteredText = paragraph.getText().replaceAll("[^\\p{L}\\p{N}\\p{P}]\\s", "");
                        body.append(filteredText).append("\n");
                    }

                    writer.write("Title: " + title + "\n");
                    writer.write("Body:\n" + body + "\n");
                }
            }
        }
    }
}
```

Implementing web crawling on an everytime website with JAVA

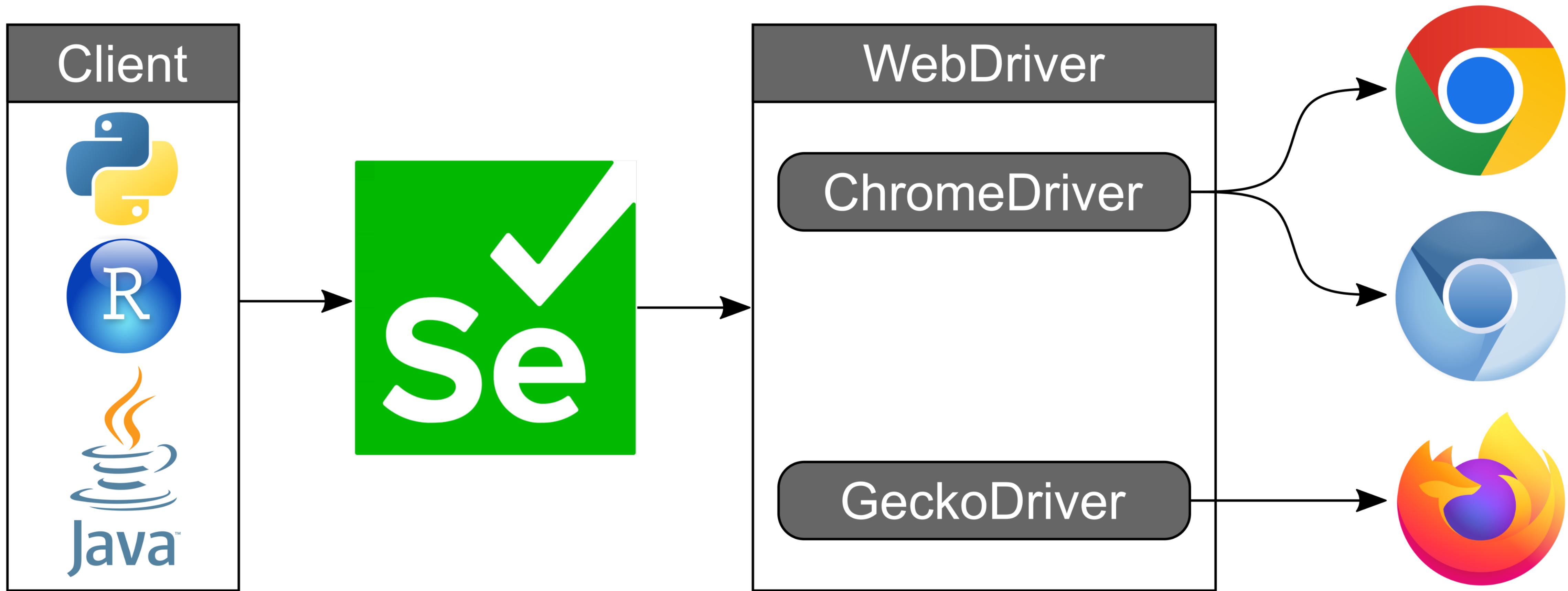
SELENIUM

CRAWLING

output.txt

```
1 Title: 자과캠 만화동아리 그림마당에서 24-2 신입 부원을 모집합니다!
2 Body:
3 | 자과캠 만화동아리 그림마당에서 24-2 신입 부원을 모집합니다!
4
5 안녕하세요, 성균관대학교 자연과학캠퍼스 중앙만화동아리 그림마당입니다.
6 2024년 2학기를 함께 할 신입 부원을 모집하고 있습니다!
7
8 그림마당 소개:
9 그림마당은 그림 그리기 / 만화 그리기 / 만화 및 애니메이션 등을 좋아하는 사람들이 모인 동아리입니다.
10 만화 및 일러스트가 담긴 회지를 발간하고, 일러스트를 그려 전시하는 활동을 하고 있습니다!
11
12 모집 기간: 상시 모집
13 지원 자격: 그림 실력 무관 / 학번 무관 / 나이 무관 / 성균관대학교 학생이라면 모두 지원 가능합니다.
14 그림 그리기 / 만화 그리기 / 만화 및 애니메이션과 관련된 것에 관심 있는 분이라면 누구나 지원이 가능합니다!
15
16 활동 소개:
17 모든 활동은 자율 참여로, 활동을 강제하지 않습니다!
18
19 #회지
20 매 학기 부원들이 창작한 만화와 일러스트로 동아리 회지를 발간합니다. 회지 참여는 그림실력과 무관하고, 누구나 자유롭게 할 수 있습니다.
21
22 #그림 강의 및 연출 강의
23 동아리 방에서 매주마다 그림 강의를 진행합니다. 기초적인 드로잉, 만화 연출 등을 다룹니다.
24
25 #일러스트전시회
26 일러전은 부원들이 그린 일러스트를 삼성학술정보관(디도)에 전시하는 행사이입니다.
27 매년 2학기에 진행하며 일러전 참가자에게는 경품을 주는 행사도 진행합니다!
28
29 #홀터밍데이
30 매년 2학기 중간고사가 끝난 후 그림마당의 선배분들과 같이 밥을 먹고 대화를 나누는 '홀터밍데이'를 진행합니다.
31
32 #총회(정기모임)
33 그림마당은 매 학기마다 개강총회, 중간총회, 종강총회로 총 세 번의 총회를 진행하고 있습니다.
```

SELENIUM WORKFLOW



AWS ELB DEPLOYMENT



- Automatically handles infrastructure provisioning
- Built-In Monitoring & Management
- Supports Multiple Platforms

DATABASE

AWS

RDS



- Managed Database Service: backups, updates and patching, reducing administrative overhead
- High Availability & Scalability
- Built-In Security

DATABASE STRUCTURE

