

## 발표 스크립트 및 예상 Q&A

### 담당 파트 소개 (1~2분)

#### 발표 멘트 예시

"안녕하세요, 저는 이번 프로젝트에서 **Company Service**와 **홈 화면 지도 기능**을 담당한 심창용입니다. 먼저 **Company Service**는 기업 회원을 위한 서비스로, 회원가입부터 로그인, 마이페이지 관리까지 기업 회원의 전체 라이프사이클을 담당합니다. 제가 구현한 주요 기능은 크게 **4가지**입니다."

---

### Company Service 주요 기능 설명 (3~4분)

#### 1 회원가입 및 로그인 시스템

"첫 번째로 **회원가입 및 로그인 시스템**입니다. 기업 회원이 가입할 때 **아이디 중복 체크, 이메일 중복 체크, 사업자등록번호 중복 체크**를 수행하고, 비밀번호는 **BCrypt와 Salt**를 사용해 안전하게 암호화합니다. 로그인 시에는 **JWT 토큰**을 발급하며, 토큰에 회사ID, 회사명, 대표자명, 주소, 로고 경로 등의 정보를 담아 프론트엔드에서 활용할 수 있도록 했습니다. 또한 **관리자 승인 전에는 로그인이 불가능**하도록 구현하여 신뢰성 있는 기업만 서비스를 이용할 수 있게 했습니다."

#### 2 이메일 인증 시스템

"두 번째는 **이메일 인증 시스템**입니다. Spring Mail을 활용해 **6자리 인증번호**를 이메일로 발송하고, 인증번호는 **5분간 유효**합니다. 회원가입 시에도 사용하지만, 특히 **마이페이지에서 회원정보를 수정할 때 이메일 인증을 필수**로 요구하여 본인 확인 절차를 강화했습니다."

#### 3 마이페이지 기능 (파일 업로드 포함)

"세 번째는 **마이페이지 기능**입니다. 기업 회원은 회사 정보, 담당자 정보, 비밀번호 등을 수정할 수 있고, **기업 로고와 기업 전경 사진을 업로드**할 수 있습니다. 파일 업로드는 **UUID를 활용해 고유한 파일명을 생성**하여 중복을 방지했고, 기존 파일이 있으면 자동으로 삭제 후 새 파일을 저장하도록 구현했습니다."

#### 4 아이디 찾기 / 비밀번호 재설정

"네 번째는 **아이디 찾기**와 **비밀번호 재설정** 기능입니다. 아이디 찾기는 이메일과 담당자명을 입력하면 등록된 아이디를 알려주고, 비밀번호 재설정은 이메일 인증 후 새 비밀번호를 설정할 수 있도록 했습니다."

---

## 📖 홈 화면 지도 기능 설명 (1분)

- "그리고 홈 화면의 **지도 기능**을 담당했습니다.  
(구체적인 지도 API 사용 내용에 따라 추가)
- Kakao Maps API / Google Maps API 활용
  - 기업 위치 마커 표시
  - 클릭 시 기업 정보 팝업 등"

## 💡 기술적 특징 요약

구분	사용 기술
인증	JWT (JSON Web Token)
비밀번호 암호화	BCrypt + Salt
이메일 발송	Spring Mail (JavaMailSender)
파일 업로드	MultipartFile, UUID 기반 파일명
데이터베이스	JPA (Spring Data JPA), H2/MySQL
삭제 처리	논리 삭제 (DEL_YN 플래그)

## ? 예상 질문 및 답변

### Q1. JWT를 사용한 이유는?

A: "세션 기반 인증은 서버에 상태를 저장해야 하지만, **MSA 환경에서는 여러 서비스가 독립적으로 운영** 되기 때문에 세션 공유가 어렵습니다.  
JWT는 **토큰 자체에 사용자 정보를 담고 있어 Stateless**하게 운영할 수 있고, Gateway에서 토큰만 검증 하면 각 서비스로 요청을 전달할 수 있어 MSA에 적합합니다.  
또한 토큰에 회사ID, 회사명 등의 정보를 담아 **매번 DB 조회 없이 프론트엔드에서 바로 사용할 수 있는 장** 점이 있습니다."

### Q2. 비밀번호 암호화에 BCrypt와 Salt를 함께 사용한 이유는?

A: "BCrypt 자체도 내부적으로 Salt를 생성하지만, 저희는 **별도의 Salt를 DB에 저장**하는 방식을 선택했습니다.

이렇게 하면 **Salt 값을 별도로 관리**할 수 있고, 향후 암호화 알고리즘을 변경하더라도 유연하게 대응할 수 있습니다.

또한 동일한 비밀번호라도 **사용자마다 다른 해시값**이 생성되어 Rainbow Table 공격을 방어할 수 있습니다."

---

### Q3. 회원 삭제를 물리 삭제가 아닌 논리 삭제로 구현한 이유는?

A: "실제 서비스에서는 **데이터 복구 가능성**과 **이력 관리**를 위해 논리 삭제를 많이 사용합니다.

저희 플랫폼에서도 기업 회원이 탈퇴해도 해당 기업이 등록했던 채용 공고나 지원 이력 등의 **연관 데이터**를 **보존**해야 할 수 있기 때문에 DEL\_YN 플래그를 사용한 논리 삭제를 구현했습니다.

조회 시에는 항상 `DEL_YN = 'N'` 조건을 추가하여 삭제된 데이터는 조회되지 않도록 했습니다."

---

### Q4. 이메일 인증번호의 유효시간은 어떻게 관리하나요?

A: "VerificationService에서 **인메모리 Map**에 이메일과 인증번호, 생성 시간을 저장하고, 검증 시 **5분이 지났는지** 체크합니다.

실제 운영 환경에서는 **Redis**를 사용해 TTL(Time To Live)을 설정하면 더 효율적으로 관리할 수 있습니다."

---

### Q5. 파일 업로드 시 보안은 어떻게 처리했나요?

A: "파일 업로드 시 **UUID**를 사용해 **파일명**을 랜덤하게 생성하여 원본 파일명이 노출되지 않도록 했습니다.

또한 파일 저장 경로를 **application.properties**에서 설정할 수 있게 하여 환경별로 다른 경로를 사용할 수 있습니다.

추가로 **파일 확장자 검증**이나 **파일 크기 제한** 등을 구현하면 보안을 더 강화할 수 있습니다."

---

### Q6. 관리자 승인 전 로그인 불가 처리는 왜 했나요?

A: "주한미군 전역장병 채용 플랫폼의 특성상 **신뢰할 수 있는 기업만 서비스를 이용**해야 합니다.

아무 기업이나 가입 즉시 채용 공고를 올릴 수 있다면 허위 공고나 스팸이 발생할 수 있기 때문에, **관리자가 사업자등록번호 등을 검증한 후 승인**해야만 로그인이 가능하도록 구현했습니다.

이를 통해 플랫폼의 신뢰성을 높일 수 있습니다."

---

### Q7. MSA 환경에서 서비스 간 통신은 어떻게 하나요?

A: "저희 프로젝트에서는 **Spring Cloud Gateway**를 통해 API 라우팅을 하고, **Eureka**로 서비스 디스커버리를 합니다.

Company Service에서 생성한 기업 정보를 Job Service에서 사용해야 할 때는 **Gateway**를 통한 **REST API 호출**이나 **공유 데이터베이스**를 활용할 수 있습니다.

현재는 JWT 토큰에 필요한 기업 정보를 담아 전달하는 방식을 사용하고 있습니다."

---

## Q8. 개발하면서 가장 어려웠던 점은?

A: "JWT 토큰에 어떤 정보까지 담을지 결정하는 것이 어려웠습니다.

토큰에 정보를 많이 담으면 편리하지만 **토큰 크기가 커지고**, 정보가 변경되면 **토큰을 재발급**해야 하는 문제가 있습니다.

결국 **자주 사용하는 기본 정보만 토큰에 담고**, 상세 정보는 필요할 때 API로 조회하도록 설계했습니다."

---

## Q9. 홈 화면 지도 기능은 어떻게 구현했나요?

A: "(실제 구현 내용에 따라 답변)

- Kakao Maps API를 사용했습니다
  - 기업 주소를 좌표로 변환하여 마커로 표시했습니다
  - 마커 클릭 시 기업 정보 팝업이 표시됩니다
  - 등등..."
- 

## Q10. 향후 개선하고 싶은 점은?

A: "현재 이메일 인증번호를 **인메모리**로 관리하고 있는데, **Redis**를 도입하면 서버 재시작 시에도 데이터가 유지되고 TTL 관리가 더 쉬워질 것 같습니다.

또한 파일 업로드 시 **AWS S3** 같은 클라우드 스토리지를 사용하면 확장성과 가용성을 높일 수 있을 것 같습니다.

그리고 **\*\*소셜 로그인(OAuth2)\*\***을 추가하면 기업 담당자의 가입 편의성을 높일 수 있을 것 같습니다."

---

## 한 줄 요약 (마무리 멘트)

"Company Service는 기업 회원의 **가입 → 인증 → 로그인 → 정보관리 → 탈퇴**까지의 전체 흐름을 담당하며, **JWT 인증, 이메일 인증, 파일 업로드** 등의 기능을 구현했습니다. 감사합니다."