

### 소프트웨어 공학 과제3 보고서

B711210 허동민, B735365 이윤식, B811128 이선주, B811203 하종훈

#### ■ 팀 내 역할 분담 및 각 팀원이 실제 수행한 내용

##### (1) 팀 내 역할 분담

B711210 허동민	회원탈퇴, 채용 정보 검색, 채용 지원
B735365 이윤식	회원가입, 채용 정보 등록, 등록된 채용 정보 조회
B811128 이선주	로그아웃, 지원 정보 통계
B811203 하종훈	로그인, 지원 정보 조회, 지원 취소

##### (2) 각 팀원이 실제 수행한 내용

###### 1. 첫 번째 회의 (5월 19일, A동 2층 스터디룸)

첫번째 회의 전 : 수정된 명세서를 읽어보고 requirement list, usecase diagram, usecase description을 수정해오기로 하였다.

첫번째 회의 중 : 각자 수정해온 requirement list, usecase diagram, usecase description을 함께 읽어보았고, 각자 usecase를 맡아 수정해온 파일들을 github에 업로드하였다. 다음 회의까지 각자 맡은 usecase에 대한 communication diagram 및 design class diagram을 그려오기로 하였다.

두번째 회의까지 각자 해온 내용은 다음과 같다.

B711210 허동민	맡은 usecase에 해당하는 communication diagram을 먼저 그려보았다. 실제로 코드로 구현을 할 때, 필요한 함수가 무엇인지 생각했고, 이를 토대로 design class diagram를 그렸다. 내가 맡은 부분만 그려서 design class diagram의 완성도는 떨어졌지만, 내가 앞으로 어떤 코드를 짜야할지 계획을 세우는 시간이 되었다.
-------------	---

B735365 이윤식	HW2에서 했었던 <b>communication diagram</b> 을 참고해서 이번 과제의 <b>communication diagram</b> 을 수정해보았다. 특히, 지난 번에 감점되었던 부분은 더욱 세심하게 확인하여 작업하였다. <b>design class diagram</b> 은 HW2에서 했었던 <b>analysis class diagram</b> 을 참고해서 작성하였다. 다만, 실제로 코드에서 사용될 매개 변수와 리턴 타입 등을 엄밀하게 따져본 후, 최대한 실제 코드와 비슷하게 작성하고자 하였다.
B811128 이선주	<ul style="list-style-type: none"> <li>- 채용정보통계 서브시스템의 <b>usecase</b>들(등록한 채용정보 통계 조회, 지원한 채용정보 통계 조회)과 회원 관리 서브시스템의 로그아웃 <b>usecase</b>를 맡았다.</li> <li>- 수정된 과제 명세서에 따라 맡은 <b>usecase</b>들의 <b>communication diagram</b>을 수정했다.</li> <li>- 수정된 과제 명세서에 따라 맡은 <b>usecase</b>들의 <b>detailed class diagram</b>을 새로 그려보았다.</li> </ul>
B811203 하중훈	구현할때 필요한 함수들에 대해 생각해보았고, 담당한 <b>usecase</b> 의 <b>communication diagram</b> 을 그렸다. 이것들을 바탕으로 <b>design class diagram</b> 도 그려보았다. 두 다이어그램을 그려보면서 실제 구현을 할 때 어떻게 해야 좋을지 고민해 보았다.

## 2. 두 번째 회의 (5월 21일, 구글 미트)

두번째 회의 중 : **communication diagram**을 합치는 작업을 하였다. 각자 그려온 **design class diagram**에 대해 설명하는 시간을 가졌고, 피드백하며 수정하였다. 또한, 각자 그려온 **design class diagram**을 하나의 **design class diagram**으로 합쳤다. **main.cpp**를 작성했으며, **convention**과 **comment** 및 함수 이름 기준을 일관성있게 정했다. 또한 모든 **usecase**에서 함께 사용하는 **entity**와 **collection**을 작성하였다.

다음 회의까지 본인 파트에 해당하는 **usecase**의 코드를 짜오기로 하였다.

세번째 회의까지 각자 해온 내용은 다음과 같다.

B711210 허동민	합쳐진 <b>communication diagram</b> 및 <b>design class diagram</b> 으로 과제 코드의 전체적인 흐름을 생각해보았다. 그 후, 내가 맡은 <b>usecase</b> 에 해당하는 <b>class</b> 들을 짰고 <b>attribute</b> 와 <b>operation</b> 을 정의하면서 실제 구현단계에 들어갔다. 다른 팀원들의 작업물을 보며, 처음 정한 기준대로 일관성있게 코드를 짜보았다.
-------------	--

B735365 이윤식	<p>내가 맡은 부분인 회원 가입을 가장 먼저 구현해보았다. 교수님이 짜주신 틀을 바탕으로 C++ 형식으로 바꿔가며 구현하였다. 지난 회의를 바탕으로 완성된 <b>design class diagram</b>을 바탕으로 코드를 짜 나갔다. 껍데기는 이미 완성되어있었고, 구체적인 내용만 짜면 되었기 때문에 큰 어려움은 없었다. 다만, 우리 팀원끼리 이미 상의되었던 내용인 <b>coding convention</b>을 맞추고, 변수나 함수 이름을 정의하는 방식을 고려하며 작성하였다.</p>
B811128 이선주	<ul style="list-style-type: none"> <li>- 내가 맡은 <b>usecase diagram</b>의 <b>detailed class diagram</b>을 보며 코드를 작성하였다.</li> <li>- 기존에는 ‘등록한 채용 정보 통계 조회’ <b>usecase</b>와 ‘지원한 채용 정보 통계 조회’ <b>usecase</b>를 분리하였지만, 코드를 작성하면서 하나의 ‘지원 정보 통계’ <b>usecase</b>로 합쳐야겠다고 판단하였다.</li> <li>- ‘지원 정보 통계’ <b>usecase</b>를 새로 만들었기에, <b>requirement list</b>, <b>usecase diagram</b>, <b>usecase requirement list</b> 를 새로 작성했다.</li> </ul>
B811203 하중훈	<p>합쳐진 <b>communication diagram</b>과 <b>design class diagram</b>을 통해 전체적인 프로그램 구조에 대해서 생각해보고 내가 맡은 부분의 실제 구현에 대해서도 생각해보았다. 먼저 필요한 <b>attribute</b>와 <b>operation</b>들을 정의했고, 각각에 대한 세부 내용을 작성하였다. 작성을 완료한 후 테스트를 통해 구현한 부분이 정상작동하는지 확인하였다.</p>

### 3. 세 번째 회의 (5월 25일, 구글 미트)

세번째 회의 중 : 각자 짰 코드에 대해 **comment**, **class** 이름, **attribute** 및 **operation** 이름 등을 일관성있게 썼는지 검사하였다. 코드를 병합하여 최종 솔루션을 만들었고, 테스트 케이스를 적용해보며, 올바르게 구현이 되는지 확인하였다. 성공적으로 구현이 된 것을 확인 후 보고서 작성을 하고 제출을 하였다.

■ Requirement List

Requirement	Use case(s)
회사 회원의 회사 이름, 사업자 번호를 저장해야 한다. 일반 회원의 이름, 주민번호를 저장해야 한다. 회사 회원과 일반 회원 모두 ID와 Password를 입력해야 한다.	1. 회원가입
회원의 정보를 시스템에서 삭제한다.	2. 회원탈퇴
.회원이 등록한 ID와 Password로 로그인 할 수 있어야 한다.	3. 로그인
회원이 로그아웃을 할 수 있어야 한다.	4. 로그아웃
회사 회원이 채용 정보를 등록할 수 있어야 한다. 또한 채용 정보 등록 시 업무, 인원 수, 신청 마감일 등의 정보를 입력하도록 해야 한다.	5. 채용 정보 등록
회원 자신이 등록한 채용 정보 리스트를 조회할 수 있어야 한다.	6. 등록된 채용 정보 조회
일반 회원이 회사 이름으로 검색할 수 있어야 한다. 검색 결과로는 회사이름, 사업자 번호, 업무, 인원 수, 신청 마감일 정보를 출력한다.	7. 채용 정보 검색
해당 채용 정보에 즉시 지원할 수 있는 기능을 제공해야 한다.	8. 채용 지원
일반 회원은 자신의 지원 정보(회사이름, 업무, 인원수, 신청 마감일)를 조회할 수 있어야 한다. 이때 회사 이름의 오름차순으로 정렬한다.	9. 지원 정보 조회
현재 진행 중인 채용 정보를 선택하여 지원 취소를 할 수 있어야 한다.	10. 지원 취소

회사 회원은 현재까지 등록한 모든 채용 정보에 대해 업무별 지원자 수를 출력할 수 있어야 한다. 일반 회원은 본인이 지원한 정보에 대해 업무별 지원 횟수를 출력할 수 있어야 한다.	11.지원 정보 통계
---	-------------

■ Use case Descriptions(step by step breakdown)

1. 회원가입

Actor Action	System Response
1. (None)	2. 회원 종류 선택 창을 표시한다.
3. 비회원이 회원 종류로 일반 회원을 선택한다.	4. 기본 정보(이름, 주민번호)와 ID, password 입력 폼을 표시한다.
5. 기본 정보와 ID, password를 입력한다.	6. 이름, 주민번호, ID, password를 출력한다.
<p>Alternative Course:</p> <p>Step 3-5. 회원 종류로 회사 회원을 선택할 경우 기본 정보로 회사 이름, 사업자 번호를 입력 받는다.</p> <p>Step 6.회원 종류로 회사 회원을 선택할 경우 회사이름, 사업자번호, ID, 비밀번호를 출력한다.</p>	

2. 회원탈퇴

Actor Action	System Response
1. (None)	2. 회원 탈퇴 여부를 재확인하는 메시지를 띄운다.
3. 회원 탈퇴 확인을 선택한다.	4. 회원 탈퇴 시 ID를 출력한다.
<p>Alternative Course:</p> <p>Step 3. 회원 탈퇴 취소를 선택한 경우 이전 화면으로 돌아간다.</p>	

--

3. 로그인

Actor Action	System Response
1. 회원이 ID 와 password를 입력한다.	2. 로그인 후 ID와 password를 출력한다.

4. 로그아웃

Actor Action	System Response
1. (None)	2. 로그아웃 후 ID를 출력한다..

5. 채용 정보 등록

Actor Action	System Response
1. 회사 회원이 업무, 인원 수, 신청 마감일 등의 정보를 입력한다.	2. 채용 정보 등록 완료 후 업무, 인원수, 신청 마감일을 출력한다.

6. 등록된 채용 정보 조회

Actor Action	System Response
1. (None)	2. 회사 회원이 올렸던 채용 정보 (업무, 인원수, 신청 마감일)를 출력한다.

7. 채용 정보 검색

Actor Action	System Response
1. 일반 회원이 회사 이름으로 채용 정보를 검색한다.	2. 검색 결과로 채용 정보 리스트(회사이름, 사업자 번호, 업무, 인원수, 신청 마감일 정보)를 회사 이름의 오름차순으로 출력한다.

#### 8. 채용 지원

Actor Action	System Response
1. 지원할 채용 정보의 사업자 번호를 입력한다.	2. 지원한 채용 정보의 회사이름, 사업자번호, 업무를 출력한다.

#### 9. 지원 정보 조회

Actor Action	System Response
1. (None)	2. 해당 일반 회원의 지원 정보(회사이름, 업무, 인원수, 신청 마감일)를 회사 이름의 오름차순으로 정렬해서 출력한다.

#### 10. 지원 취소

Actor Action	System Response
1. 지원 취소할 채용 정보의 사업자 번호를 입력한다.	2. 취소된 채용 정보의 회사이름, 사업자번호, 업무를 출력한다.

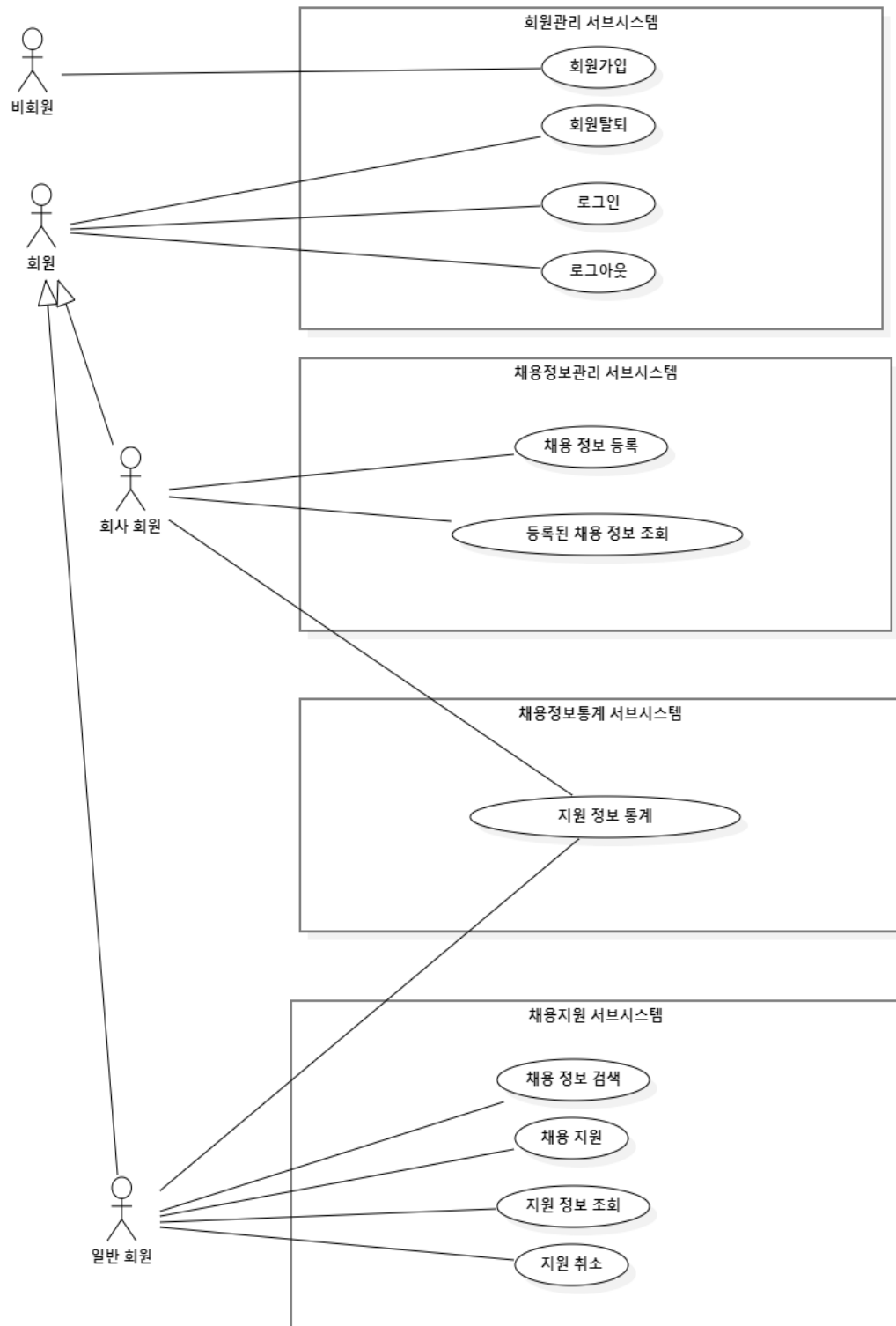
#### 11. 지원 정보 통계

Actor Action	System Response
--------------	-----------------

1. (None)	2. 로그인한 회원이 회사 회원인 경우, 본인이 현재까지 등록한 모든 채용 정보에 대해 업무별 지원자 수를 출력한다.
<p>Alternative Course:</p> <p>Step 2. 로그인한 회원이 일반 회원인 경우, 본인이 지원한 정보에 대해 업무별 지원 횟수를 출력한다.</p>	

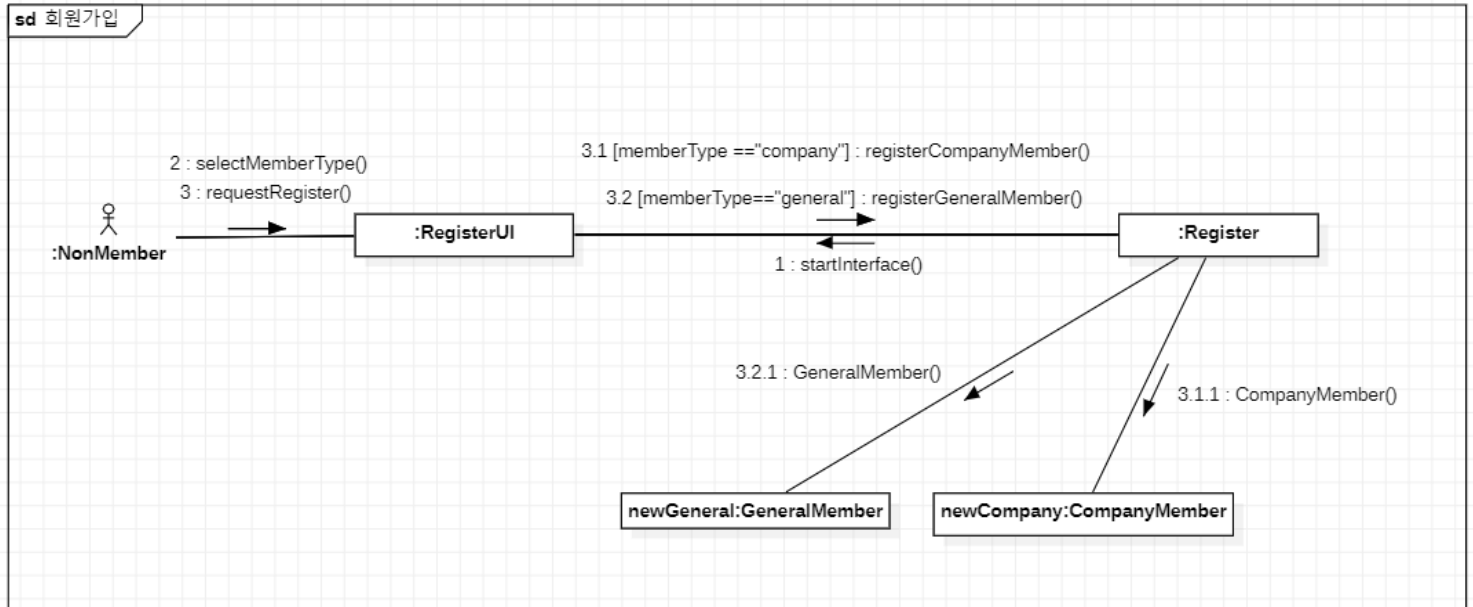


## ■ Use case Diagram

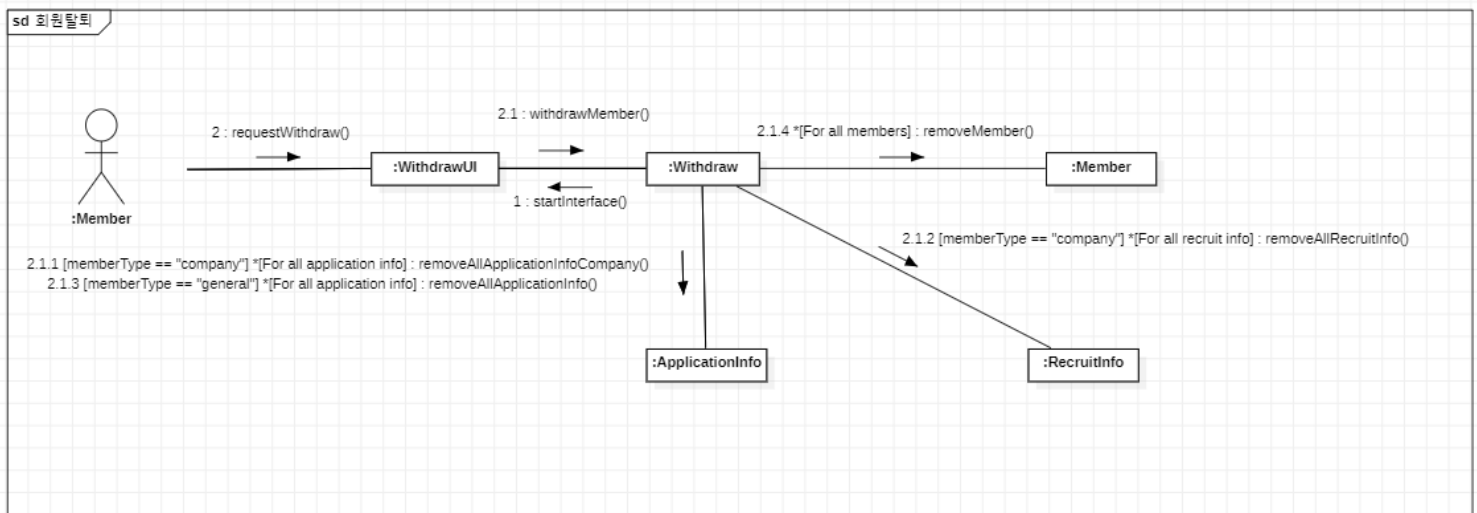


## ■ Communication Diagram

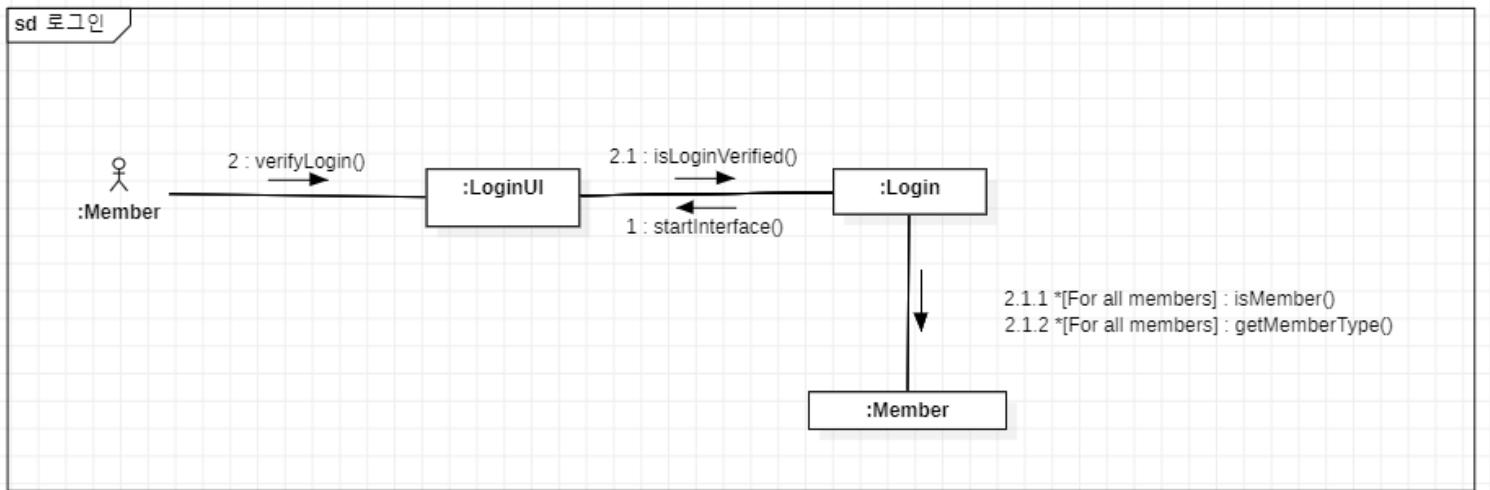
### 1. 회원가입



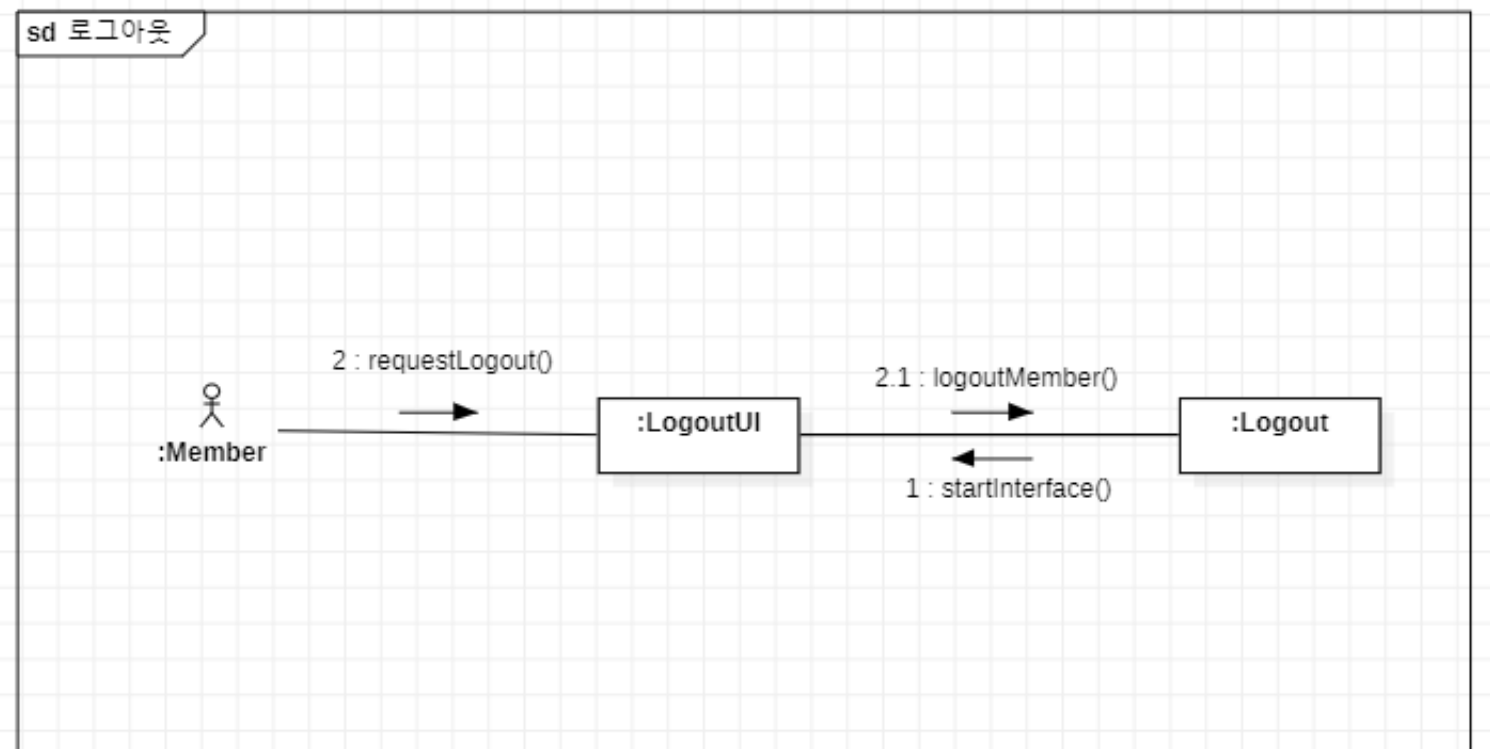
### 2. 회원탈퇴



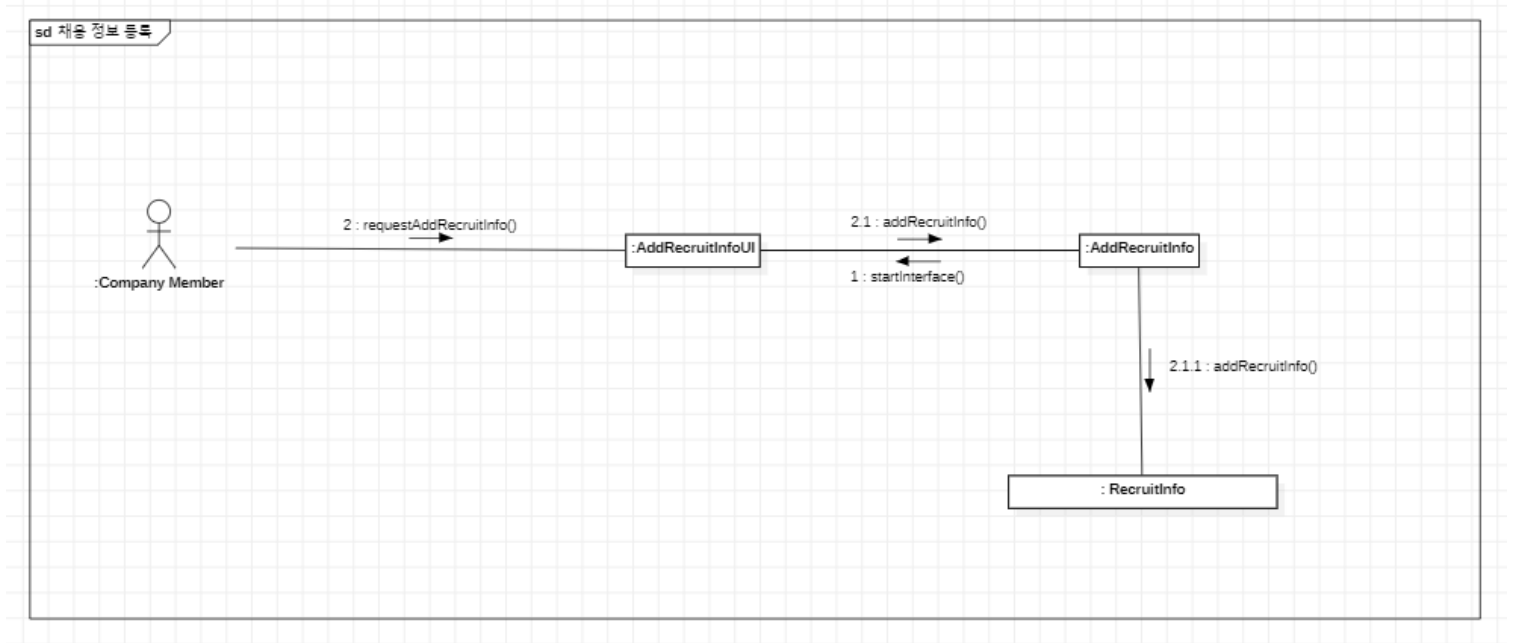
### 3. 로그인



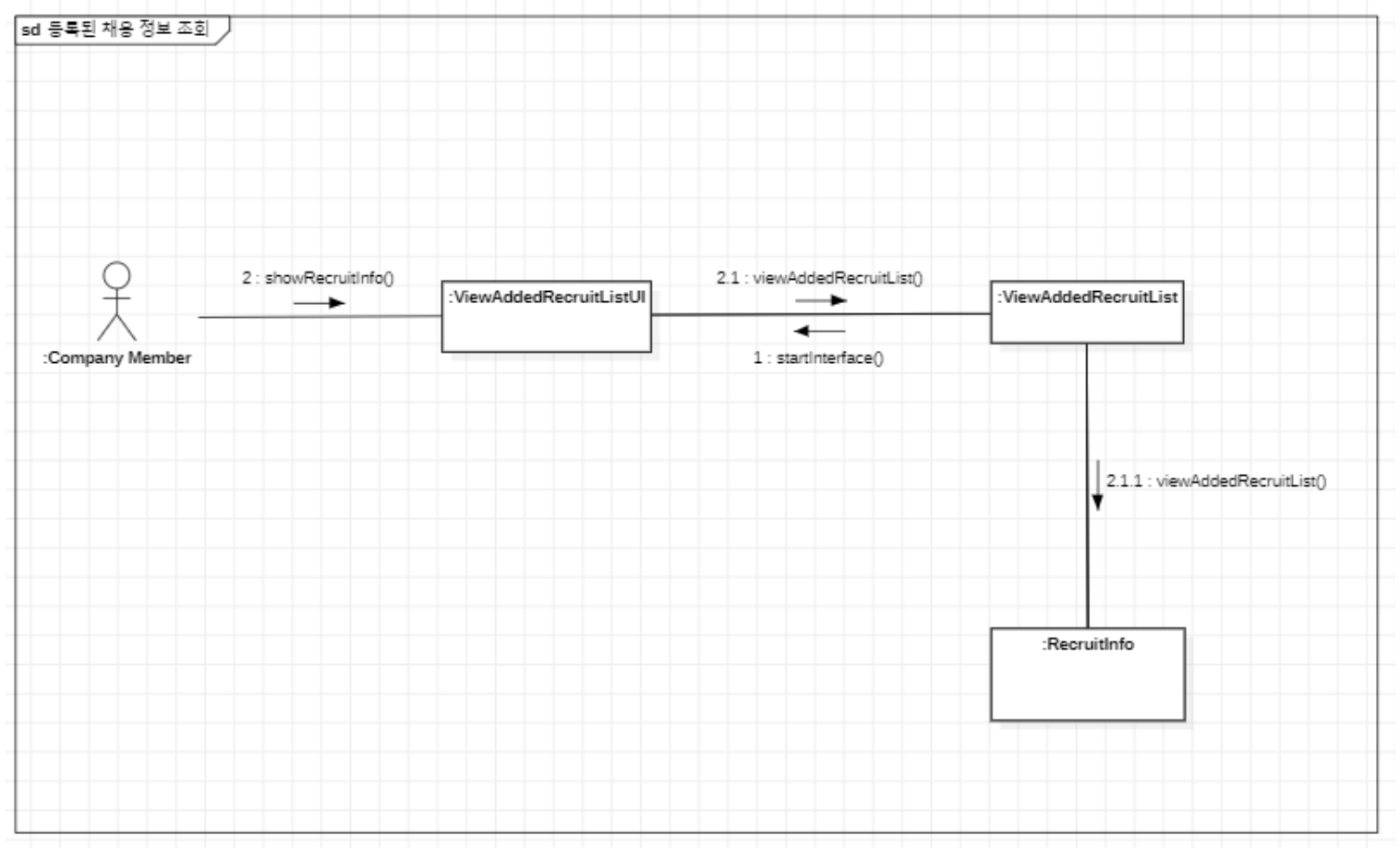
#### 4. 로그아웃



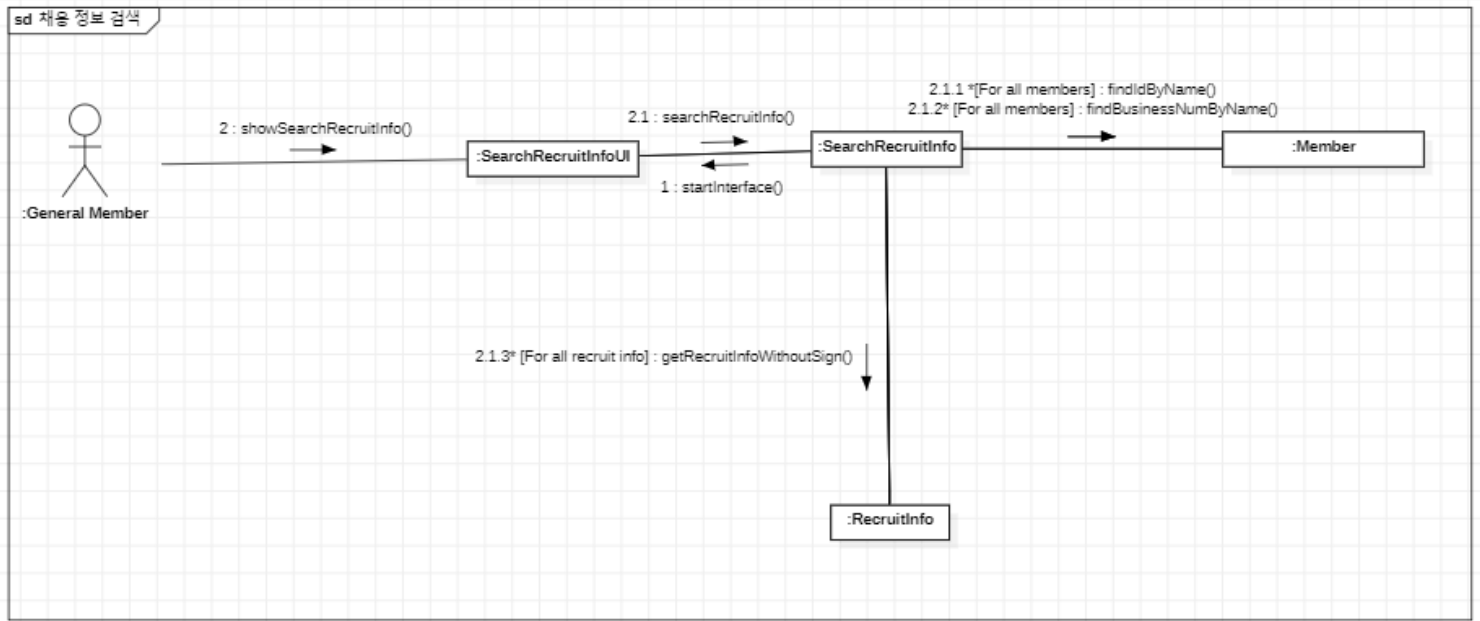
#### 5. 채용 정보 등록



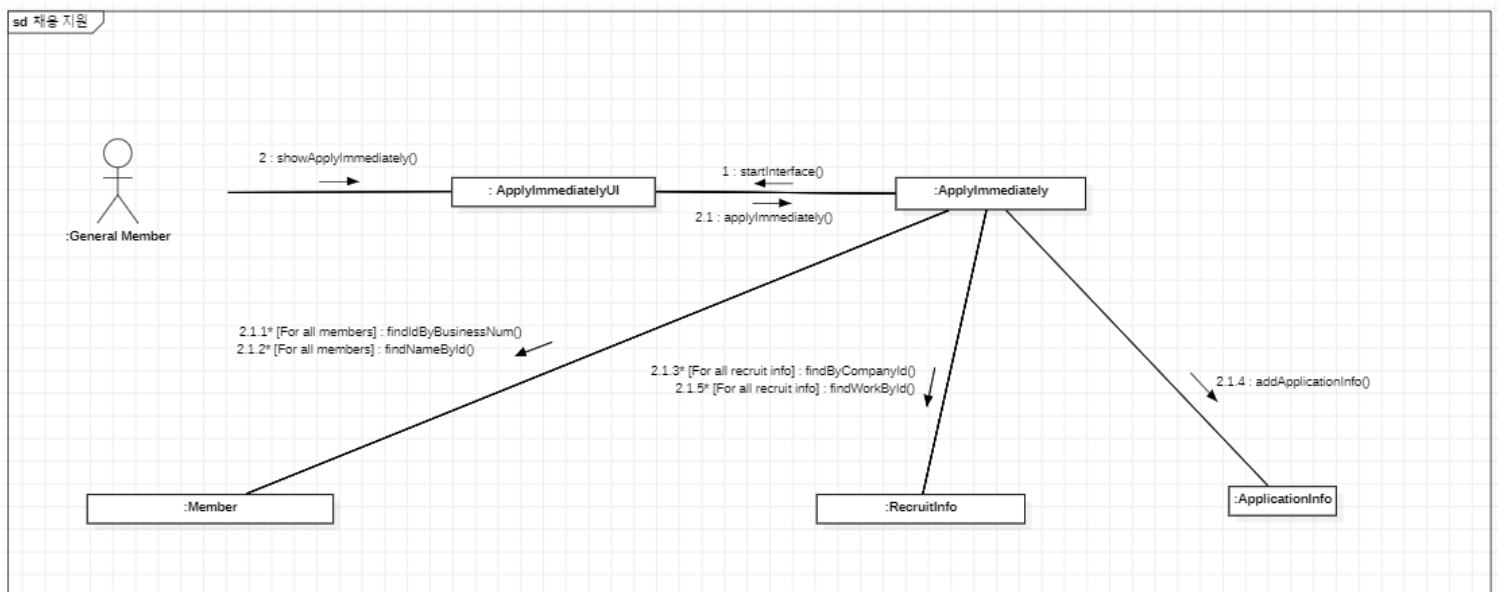
## 6. 등록된 채용 정보 조회



## .7. 채용 정보 검색

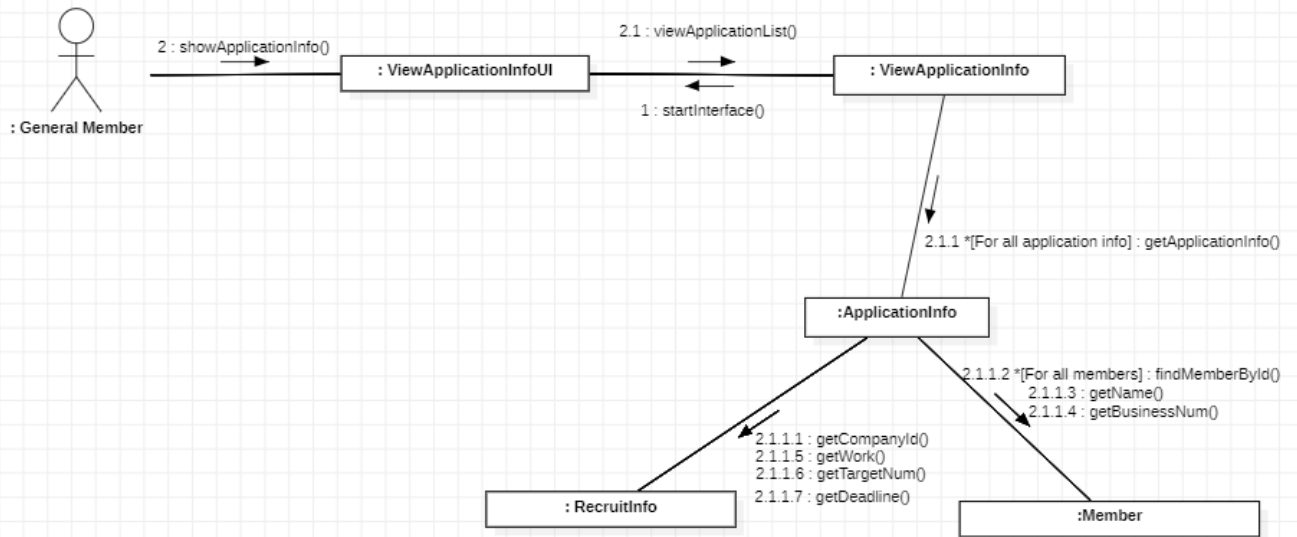


## 8. 채용 지원



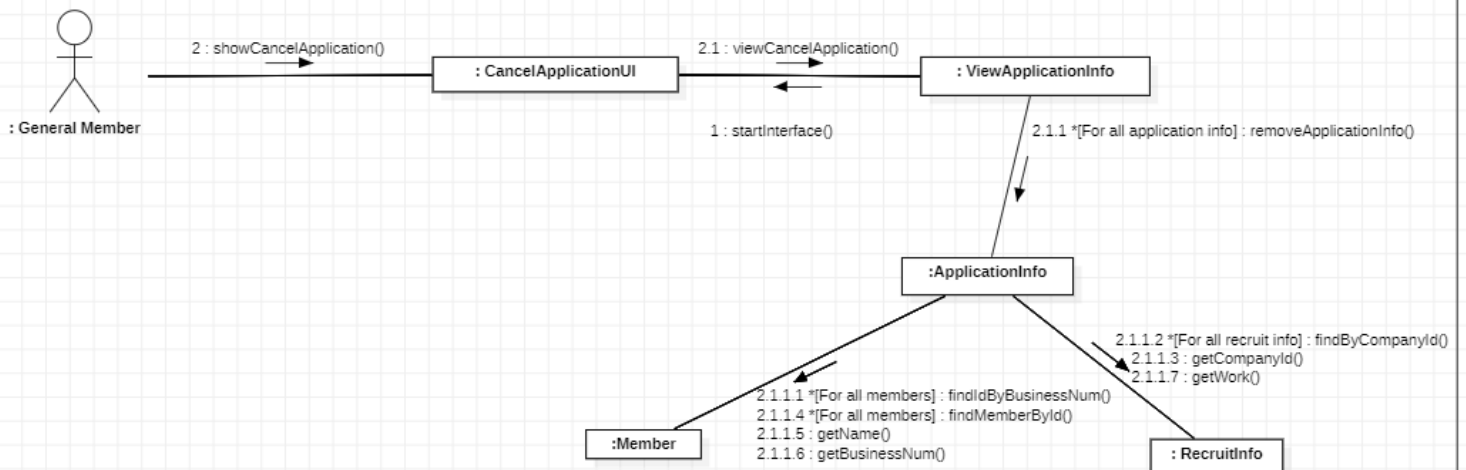
## 9. 지원 정보 조회

sd 지원 정보 조회

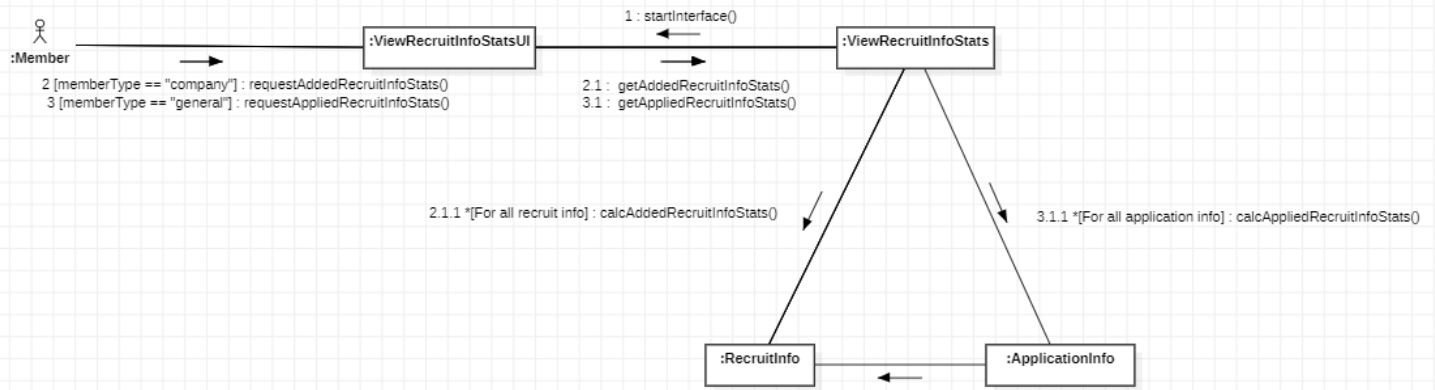


10. 지원 취소

sd 지원 취소

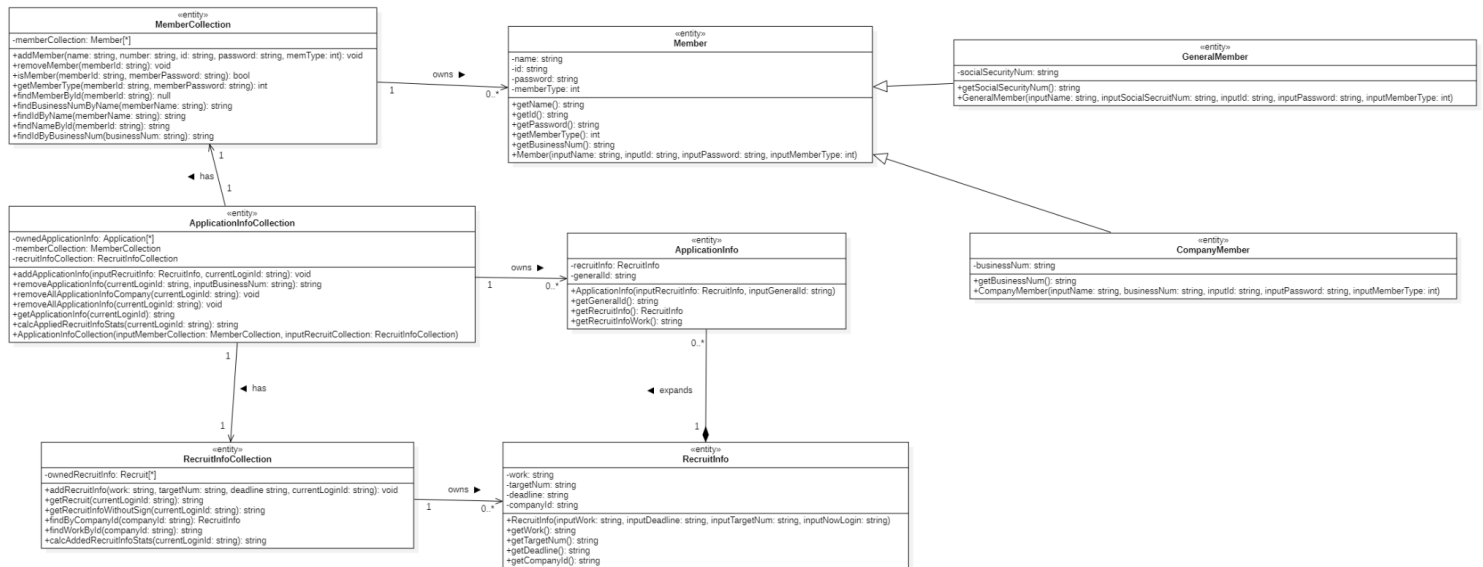


11. 지원 정보 통계

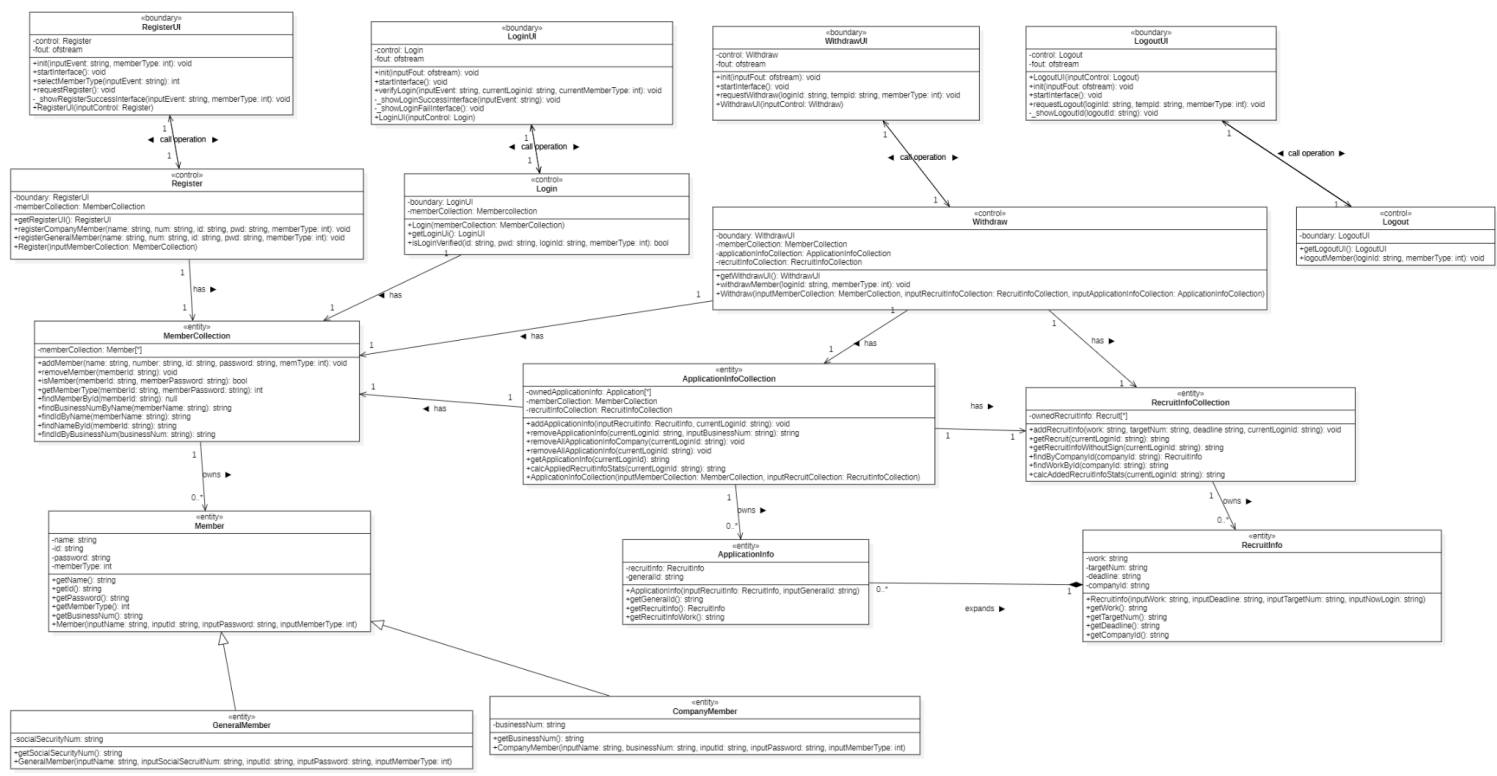


## ■ Design Class Diagram

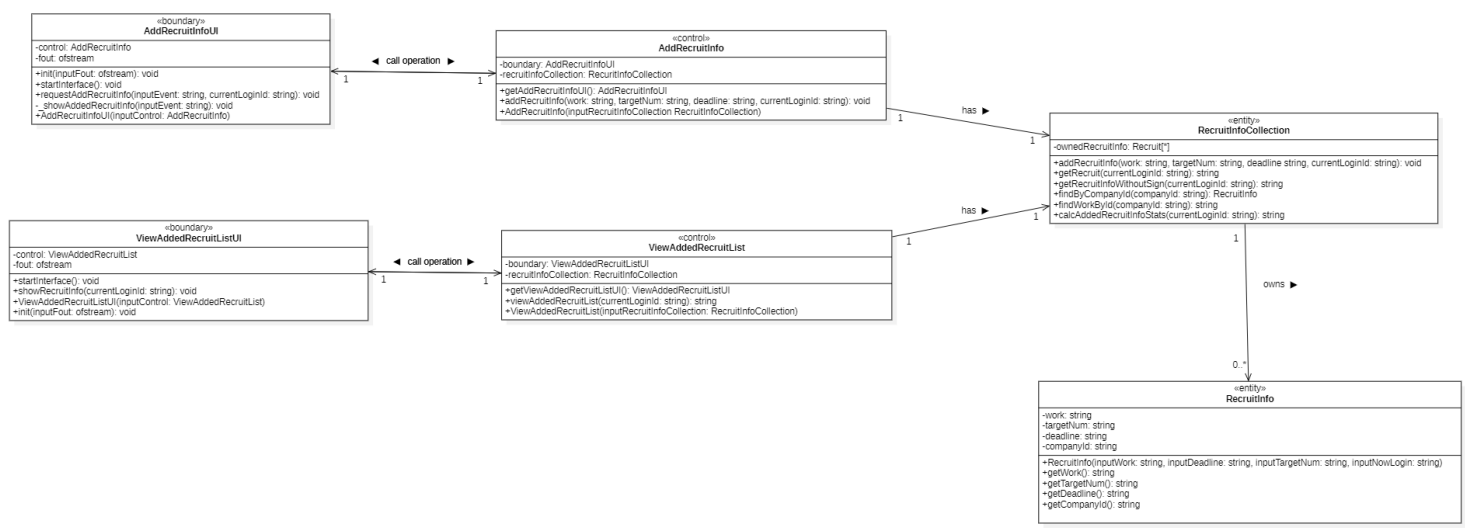
<entity>



<회원 관리 서버 시스템>



<채용 정보 관리 서버 시스템>



<채용 지원 서버 시스템>



