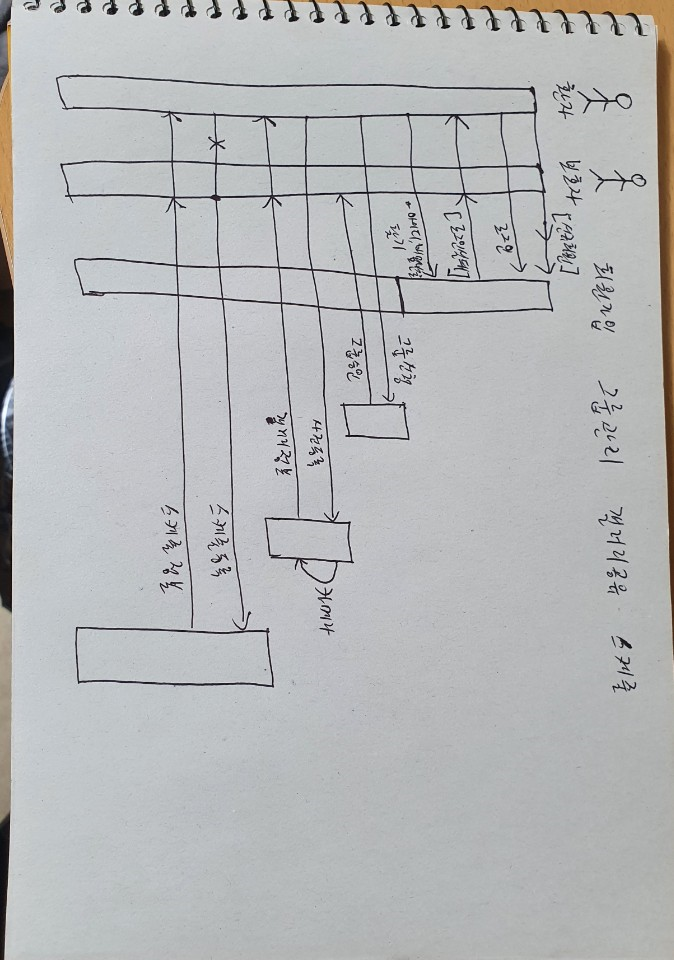
|  |  |  |
| --- | --- | --- |
| **소프트웨어설계 (팀활동) –**  **Behavioral Modeling** | | |
|  | | |
| **제출팀** | 프로젝트명: | 치매 기억 저장소 |
| **2조** | 이름: | 신기철 |

**1. Sequence Diagrams by the use-cases**

**토의 과정 : 디스코드를 통해 회의를 진행하였고 유명음 팀원이 StarUML을 화면공유를**

**통해 use-case를 기반으로 Sequence Diagram을 작성하며 진행했습니다.**

1. **초안 (신기철)**



1. **회의 내용**

**회원가입, 그룹관리는 객체가 아니다**

* **객체로 서버를 두어 작성한다**
* **서버를 거치게 되면 가독성이 떨어지고 모든 기능이 서버를 거치게 된다**
* **회원가입과, 그룹 관리 기능 사용자 클래스의 메소드로 보는게 맞다**

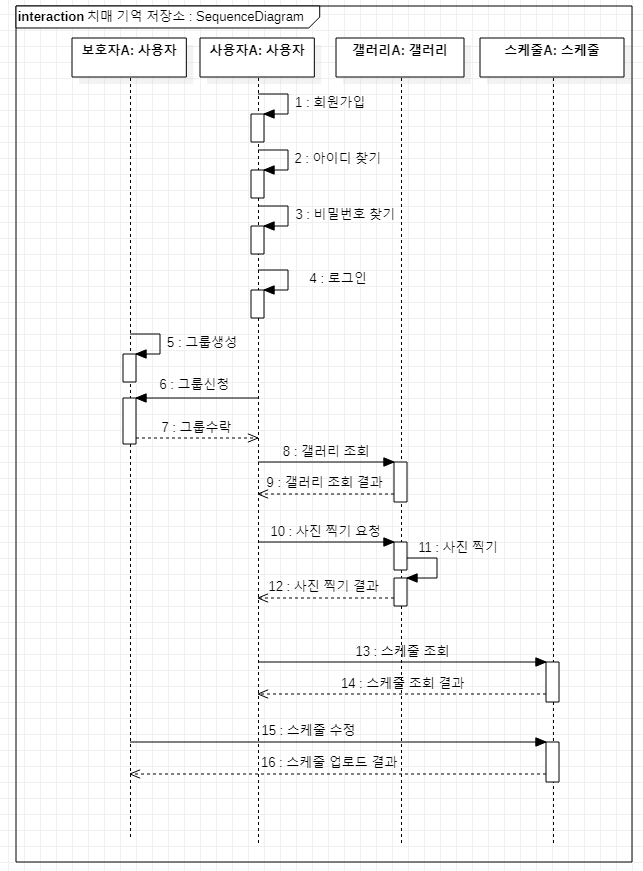
**갤러리 관리도구, 스케줄 관리도구로 이름을 바꿨지만 State diagram을 작성하며 도구의 상태를 정의하기 어려움이 있다.**

* **도구의 개념이 아닌 하나의 객체로 표현한다.**

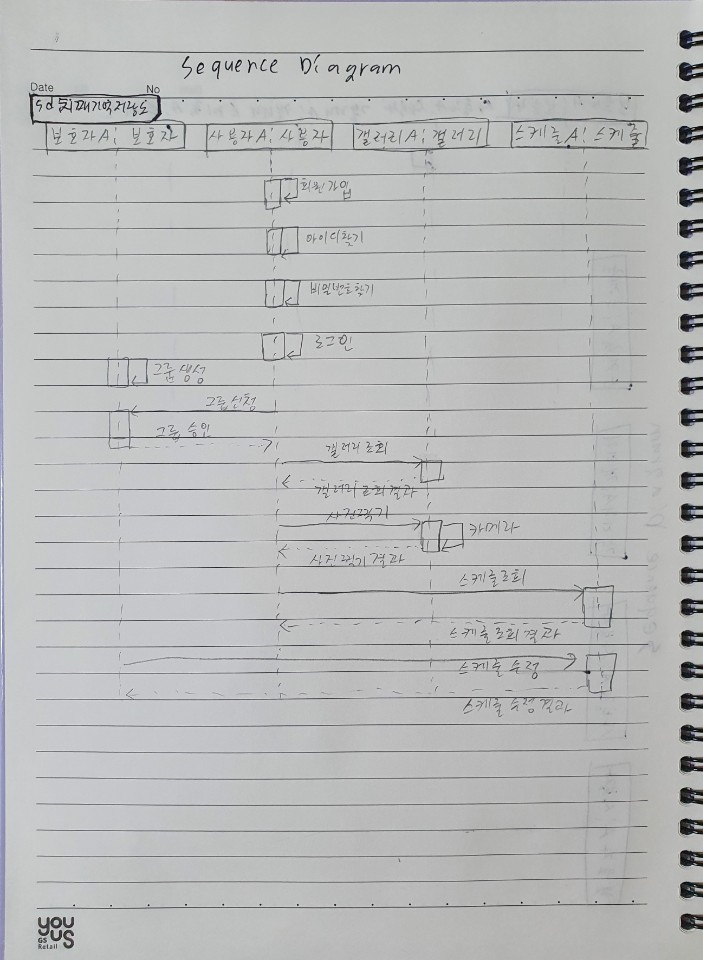
**객체의 이름이 양식의 규칙에 맞지 않다**

* **“객체 이름: 클래스 이름”으로 변경한다.**

**회의 중 작성한 StartUML**



**회의 후 수기로 작성한 Sequence Diagram**



**2. Search & Draw the object’s states (State Diagram)**

**토의 과정 : 디스코드를 통해 회의를 진행하였고 유명음 팀원이 StarUML을 화면공유를**

**통해 앞서 작성한 Sequence Diagram을 바탕으로 State Diagram을 작성하며 진행했습니다.**

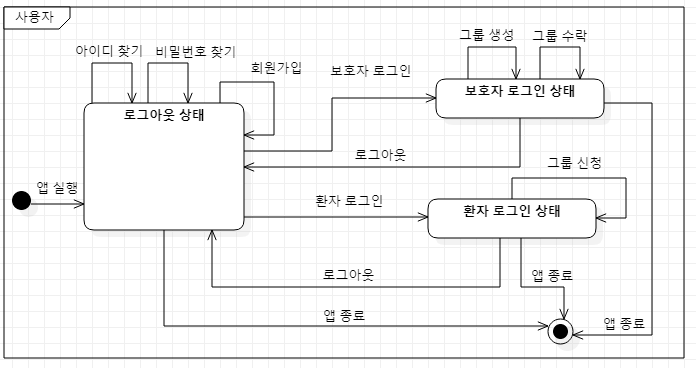
1. **회의 내용**

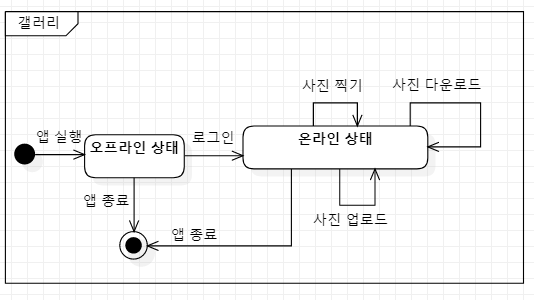
**Sequece Diagram에서 만든 객체들의 상태를 나타내는데 표현이 부적합하여 기존에 만든 객체를 없애고 사용자 클래스의 메소드로 표현했습니다.**

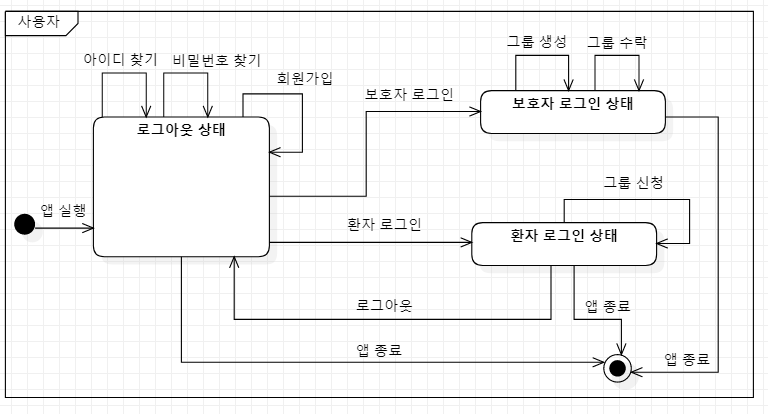
**그리고 Sequce Diagram에서는 셀프 메시지로 이를 표현했습니다.**

**환자와 보호자 권한에 따라 사용할 수 있는 기능이 다르기 때문에 환자와 보호자로 상태를 나누었습니다.**

**회의 중 만든 StartUML**







수기로 작성한 State Diagram

