
System Model (Sequence Diagram) Document

Project Name	프라이버시 보호 실시간 지원 서비스
--------------	---------------------

12조

202202624 이예인

202002569 최동현

지도교수: 장진수 교수님

Document Revision History

REV#	DATE	AFFECTED SECTION	AUTHOR
1	2025/04/29	유스케이스 및 시퀀스 다이어그램 작성	이예인
2	2025/05/01	시퀀스 다이어그램 filling	최동현

Table of Contents

1. INTRODUCTION	5
1.1. OBJECTIVE	5
2. USE CASE DIAGRAM	6
3. SEQUENCE DIAGRAM	7
3.1. LOGIN	7
3.2. MAIN 페이지 로딩	8
3.3. WEBRTC 연결	9
3.4. 화면 영역 정밀 지정	11
3.5. 민감 정보 자동 마스킹	13
4. AI 도구 활용 정보	14

List of Figure

그림 1. 유스케이스 다이어그램.....	6
------------------------	---

1. Introduction

1.1. Objective

이 문서는 프라이버시 보호 실시간 지원 서비스의 시퀀스 다이어그램에 대한 내용을 기술하고 있다. 요구사항 명세 단계에서 작성한 유스케이스 다이어그램을 기반으로 각 유스케이스의 상세한 내부 동작 흐름을 시퀀스 다이어그램으로 모델링한다.

유스케이스 다이어그램의 요소들을 시퀀스 다이어그램으로 상세한 내부 흐름을 모델링하여 해당 문서를 읽는 사용자가 다이어그램을 읽고 프로젝트에 대한 동작 순서를 완벽히 이해하는것을 목표로한다.

2. Use Case Diagram

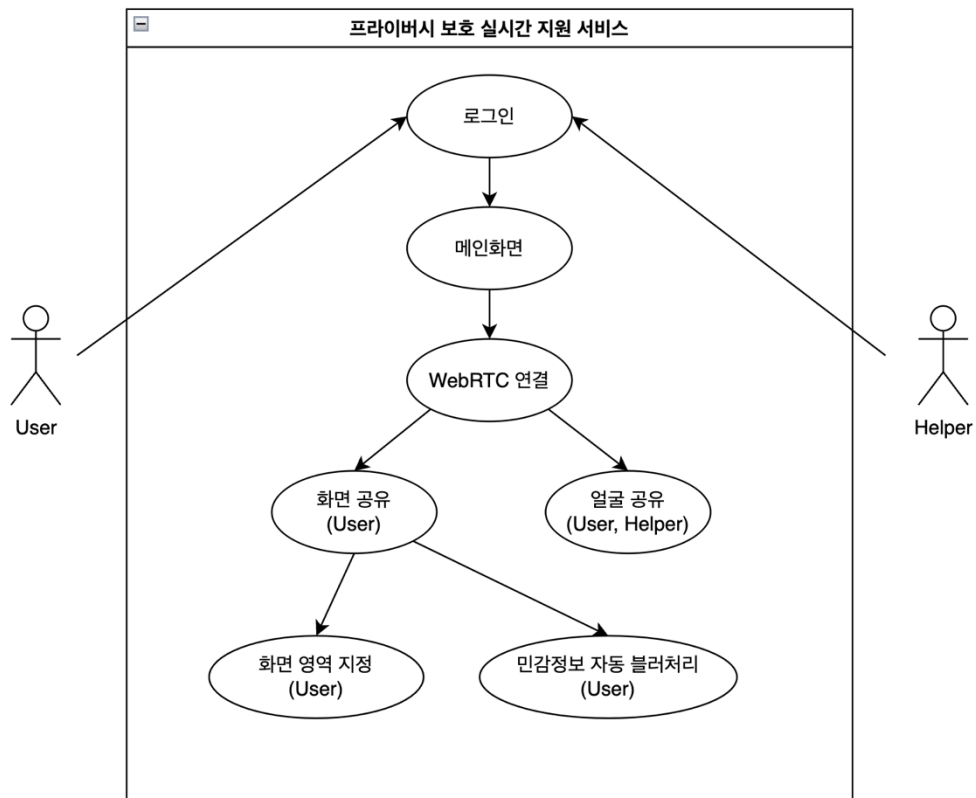


그림 1. 유스케이스 다이어그램

3. Sequence Diagram

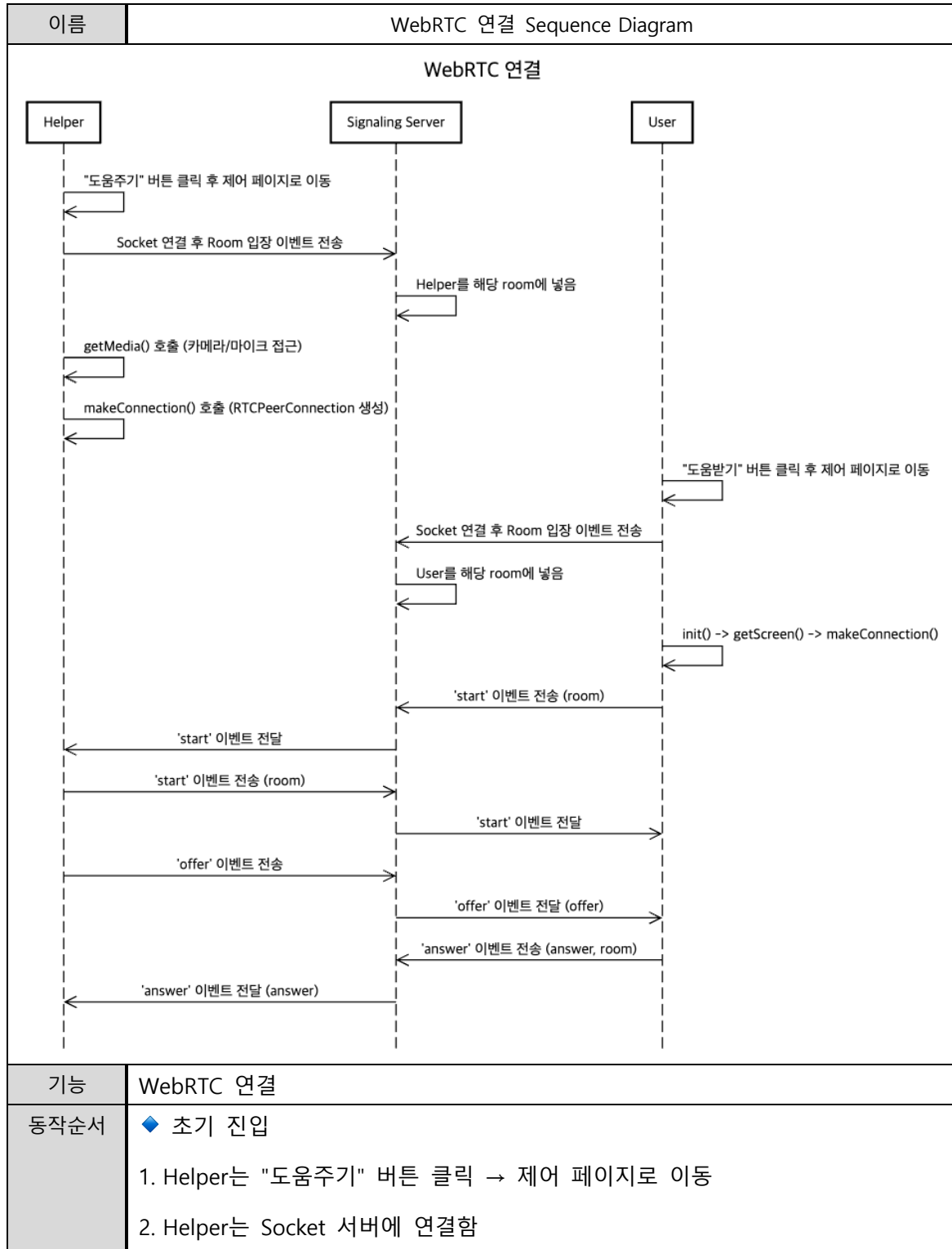
3.1. Login

이름	Login Sequence Diagram
	<p style="text-align: center;">Login</p> <pre> sequenceDiagram participant UH as User & Helper participant SS as Signaling Server UH->>SS: POST /login (email, password) SS-->>UH: 로그인 정보 확인 alt [로그인 성공] SS->>SS: 세션 생성 SS-->>UH: 응답: Success, sid UH->>UH: 메인 페이지 이동 else [로그인 실패] SS-->>UH: 응답: Fail end </pre> <p>The diagram illustrates the login process. It starts with the User & Helper sending a POST request to the Signaling Server. The server then checks the login information. An alternative (alt) block handles two outcomes: success and failure. In the success case, the server creates a session and sends a response with 'Success' and a session ID (sid), after which the user moves to the main page. In the failure case, the server sends a response with 'Fail'.</p>
기능	로그인 시스템
동작순서	<ol style="list-style-type: none"> 1. 유저가 시그널링 서버에 로그인 정보(email, password)를 보냄. 2. 시그널링 서버가 로그인 정보를 확인하고 결과를 유저에게 전송. 3.1. 로그인이 성공하면 시그널링 서버는 세션을 생성하고, 응답(Success, sid)을 유저에게 보냄 3.2. 시그널링 서버는 응답(Fail)을 유저에게 보냄 4. 응답이 Success면 메인 페이지로 이동하고 Fail이면 페이지를 리로드 시킴.

3.2. Main 페이지 로딩

이름	Main 페이지 로딩 Sequence Diagram
기능	메인 페이지 로딩.
동작순서	<ol style="list-style-type: none"> 1. 유저는 메인 페이지를 시그널링 서버에 요청. 2. 시그널링 서버는 사용자 세션을 확인 3.1. 세션이 없으면 유저는 로그인 페이지로 이동 3.2.1. 세션이 있으면 시그널링 서버가 메인 페이지를 유저에게 전송하고, 시그널링 서버에 사용자 목록을 요청 3.2.2. 시그널링 서버는 세션 기반 사용자 목록을 조회하고 사용자에게 사용자 목록을 보냄.

3.3. WebRTC 연결



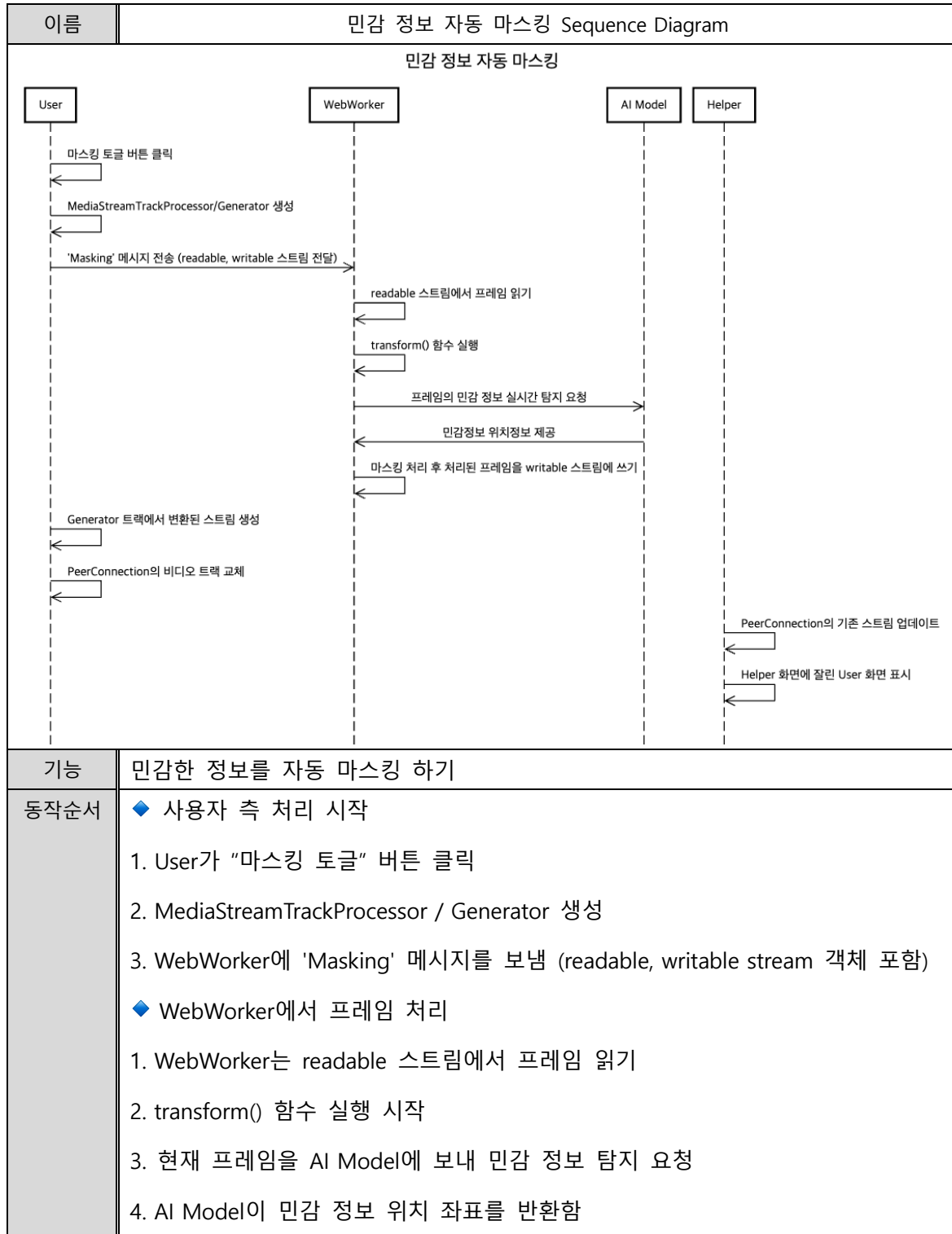
3. Helper는 Room 입장 이벤트를 Signaling Server에 보냄
 4. Signaling Server는 Helper를 해당 room에 넣음
 5. Helper는 getMedia() 호출하여 카메라/마이크 접근 권한 요청
 6. Helper는 makeConnection() 호출 → RTCPeerConnection 생성
 7. User는 "도움받기" 버튼 클릭 → 제어 페이지로 이동
 8. User는 Socket 서버에 연결함
 9. User는 Room 입장 이벤트를 Signaling Server에 보냄
 10. Signaling Server는 User를 해당 room에 넣음
 11. User는 init() → getScreen() → makeConnection() 호출로 화면 공유 준비 및 PeerConnection 생성
- ◆ 연결 시작 및 시그널링 교환
1. User는 'start' 이벤트를 Signaling Server에 전송 (room 포함)
 2. Signaling Server는 'start' 이벤트를 Helper에게 전달
 3. Helper는 'offer' 생성 후 Signaling Server에 전송
 4. Signaling Server는 'offer'를 User에게 전달함
 5. User는 'answer' 생성 후 Signaling Server에 전송
 6. Signaling Server는 'answer'를 Helper에게 전달함

3.4. 화면 영역 정밀 지정

이름	화면 영역 정밀 지정 Sequence Diagram
	<p>화면 영역 세밀 지정</p> <pre> sequenceDiagram participant User participant WebWorker participant Helper User->>User: 영역 선택 버튼 클릭 후 슬라이더 조작 activate User User->>User: applySliderValues() -> croppingScreen(위, 아래, 왼쪽, 오른쪽) 호출 deactivate User User->>User: MediaStreamTrackProcessor/Generator 생성 deactivate User User->>WebWorker: 'crop' 메시지 전송 (readable, writable 스트림 및 crop 값 전달) activate User deactivate User WebWorker->>WebWorker: readable 스트림에서 프레임 읽기 activate WebWorker WebWorker->>WebWorker: transform() 함수 실행 (crop 값 적용하여 프레임 자르기) deactivate WebWorker WebWorker->>WebWorker: 잘린 프레임을 writable 스트림에 쓰기 activate WebWorker deactivate WebWorker WebWorker->>User: Generator 트랙에서 변환된 스트림 생성 activate WebWorker deactivate WebWorker WebWorker->>User: PeerConnection의 비디오 트랙 교체 activate WebWorker deactivate WebWorker User->>User: deactivate User Helper->>Helper: PeerConnection의 기존 스트림 업데이트 activate Helper deactivate Helper Helper->>Helper: Helper 화면에 잘린 User 화면 표시 activate Helper deactivate Helper </pre>
기능	공유하는 화면 영역 조작/지정
동작순서	<p>◆ 사용자 측 조작</p> <ol style="list-style-type: none"> 1. User가 "영역 선택" 버튼 클릭 후 슬라이더로 상하좌우 조절 2. applySliderValues() → croppingScreen() 호출 (위, 아래, 왼쪽, 오른쪽 crop 값 전달) <p>◆ 스트림 처리 초기화</p> <ol style="list-style-type: none"> 1. MediaStreamTrackProcessor 및 MediaStreamTrackGenerator 생성됨 2. crop 메시지를 WebWorker로 전송 (readable, writable stream 객체 및 crop 값 전달) <p>◆ WebWorker에서 영상 프레임 처리</p> <ol style="list-style-type: none"> 1. WebWorker는 readable 스트림에서 프레임을 읽음

	<ul style="list-style-type: none"> 2. transform() 함수 실행 → crop 값 기준으로 프레임 자름 3. 자른 프레임을 writable 스트림에 씀 ◆ User 측 전송 준비 1. Generator에서 변환된 스트림 생성 2. 기존 PeerConnection의 비디오 트랙을 새로 변환된 트랙으로 교체 ◆ Helper 화면에 적용 1. PeerConnection의 스트림이 업데이트됨 2. Helper는 User의 잘린 화면을 수신하고 표시
--	-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

3.5. 민감 정보 자동 마스킹



	<p>5. WebWorker는 해당 위치에 마스킹 처리 후, 처리된 프레임을 writable 스트림에 씀</p> <p>◆ 스트림 교체 및 표시</p> <ol style="list-style-type: none"> 1. Generator에서 마스킹 처리된 스트림이 생성됨 2. 기존 PeerConnection의 비디오 트랙을 새로운 스트림으로 교체 3. Helper 화면에 마스킹된 User 화면이 실시간 표시됨
--	------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

4. AI 도구 활용 정보

사용 도구	Gemini Advanced 2.5 Pro
사용 목적	시퀀스 다이어그램 작성 보조
프롬프트	<ul style="list-style-type: none"> ● 프로젝트 소스 코드 업로드 ● 내 프로젝트를 학습하고 아래 작성한 기능별로 시퀀스 다이어그램 순서를 알려줘. <ul style="list-style-type: none"> ■ 로그인 ■ 메인화면 ■ WebRTC 연결 ■ 화면 영역 지정 기능 ■ 민감정보 자동 탐지기능
반영 위치	시퀀스 다이어그램 디자인하는데에 사용 (섹션3의 그림)
수작업 수정	<ul style="list-style-type: none"> ● 각 시퀀스 다이어그램의 흐름에 틀린 부분이 많고, 문법오류가 다수 존재하여 논리를 올바르게 수정함 ● 시퀀스 다이어그램에서 나타난 WebRTC의 STUN/TURN 서버는 프로젝트의 중심내용에서 벗어났다고 판단하여 제외시킴