

Signal Server

API 연동 규격서

Document Information

Project Name: KnowledgeTalk Signal Server	
Project Manager: 이철원	Version Number: v1.1.1
	Version Date: 2020.09.18
Author: 박인범	Written Date: 2020.08.20
Reviewer: 박병권	Reviewed Date: 2020.09.18

Revision History

Version	Version Date	Author	Description
0.7	2020.08.20	박인범	Signal 연동 규격 작성
0.7.1	2020.09.04	이철원	내용 정리
0.8	2020.09.04	박인범	목차, Call-Flow, 규격 (ExitRoom, AliveCheck, SetVideo, SetAudio, ChangeName) 추가
1.0	2020.09.07	박인범	버전 1.0 릴리즈
1.1	2020.09.18	박인범	Turn 연동 정보 및 연동 가이드 추가
1.1.1	2020.09.18	박인범	Turn 연동 정보 (domain 적용) 및 연동 가이드 수정

목차

1. [개요](#)
2. [고려사항](#)
3. [API 사용 유의사항](#)
4. [연동 방법](#)
5. [API 목차](#)
6. [연동 API 규격](#)

1. 개요

- 본 문서는 Web Browser와 Signal Server 사이의 개발 연동을 위한 제반 사항을 기술한다.

2. 고려사항

- 매뉴얼 변경 시에는 근거를 명시하고 버전 관리를 정확하게 수행한다.
- 전문 내용 중 데이터 길이가 명시되지 않은 항목은 가변 길이 이다.
- 전문 내용 중 Request Data가 Server로부터 시작되는 경우 (Server)로 표기하였으며, 이 외의 모든 요청은 Client에서 요청하는 것으로 한다.
- 본 문서는 성능개선, 기능추가 등의 사항으로 내용이 변경될 수 있다.
- + 내용 변경이 있을 경우 버전을 업데이트 한다.

3. API 사용 유의사항

- API 권한 처리
 - I. 인증 정보(CP Code, Auth Key)를 발급하여 접속하는 클라이언트에게 제공한다.
 - II. 인증 정보를 제공받은 클라이언트는 API 요청 시 body에 인증 정보를 포함시킨다.

```
ex) {  
    cpCode: 'CPOFSECRET',  
    authKey: 'XJCJ535JKF23KRJL....'  
    user: 'tester',  
    ...  
}
```

- III. Server에서는 해당 인증 정보를 가지고 API 허용 여부를 결정하여 처리한다.

- 에러 코드 / 메시지 정의

클라이언트와 서버간 오류 메시지는 다음과 같이 정의한다.

코드 구분	구분명	코드	설명
2xx	OK	200	정상적으로 처리 되었습니다.
4xx	Client Error	400	클라이언트의 잘못된 요청으로 처리할 수 없습니다.
		401	클라이언트 인증에 실패 하였습니다.
		403	데이터가 없습니다.
		404	존재하지 않는 URL입니다.
		405	리소스를 허용할 수 없습니다.
		409	이미 등록되어 있는 사용자 입니다.

		410	등록되지 않은 사용자입니다.
		411	아이디나 패스워드가 일치하지 않습니다.
		412	데이터가 일치하지 않아 처리가 불가능합니다.
		413	사용자 권한 오류입니다.
		421	매개변수가 전달되지 않았습니다.
		422	요청 형식이나 매개변수가 유효하지 않습니다.
		440	이미 다른 사용자가 공유를 진행 중입니다.
		441	요청에 맞는 방이 없습니다.
		442	이미 동일한 유저가 방에 입장했습니다.
		445	데이터가 최대치를 초과 하였습니다.
		451	요청에 맞는 채널이 없습니다.
5xx	Server Error	500	서버에서 오류가 발생 하였습니다.
		504	서버 응답 시간이 초과하였습니다.
		543	Sync 서버에서 오류가 발생하였습니다.
		570	Media 서버에서 오류가 발생하였습니다.
		599	알려지지 않은 오류 입니다.

- Turn 연동 가이드 및 연동 시 유의사항 (아래 정책을 따르지 않으면 네트워크 사용량에 따라 기본 요금제 외의 별도 비용이 발생 할 수 있음)

- I. 원활한 영상 통신을 위해 KnowledgeTalk 에서는 STUN/Turn 서버를 운영 중임.
- II. 각 개발사의 자체 STUN/Turn을 이용하거나 기존 Open 된 free STUN/Turn (구글 Turn 서버 등)을 이용해도 무방함. KnowledgeTalk STUN/Turn을 사용하는 경우 아래와 같이 적용.
- III. RTCPeerConnection 생성 시 iceServers configuration에 STUN/Turn 서버 정보를 추가 할 수 있음. (연동 정보는 아래 4. 연동 방법 참고)

ex) new RTCPeerConnection({ iceServers: [{ urls : 'stun:stun.knowledgetalk.co.kr:46000' }, { urls: 'turn:turn.knowledgetalk.co.kr:46000', username: 'kpoint', credential: 'kpoint01' }]})

- IV. 연동 정보를 추가할 때 KnowledgeTalk STUN/Turn 서버를 사용하는 경우 1:1은 STUN+Turn이 연동 되도록 하고, 다자(useMediaSvr이 'Y'인 경우)에는 STUN만 연동 되도록 해야 함.

다자 시 ex) new RTCPeerConnection({ iceServers: [{ urls : 'stun:stun.knowledgetalk.co.kr:46000' }]})

- V. 다자 통화 시 Turn을 연동하여 relay 되는 경우, 트래픽 증가로 인해 네트워크 사용량에 따라 추가 요금이 부과될 수 있음.

- 인원 수에 따른 해상도 정책 (아래 정책을 따르지 않으면 네트워크 사용량에 따라 기본 요금제 외의 별도 비용이 발생 할 수 있음)

- I. 하기 해상도 및 Framerate 정책은 시스템에 최적화 되어있는 해상도 정책으로써, 하기 정책에 맞게 Client 해상도 변경이 필요.
- II. 하기 정책과 보다 높게 해상도 변경을 하여 운영하는 경우, 시스템 부하 및 트래픽 증가로 인해 추가 요금이 부과될 수 있음.

Layout	최대 Resolution	최대 Framerate
p2p (2 자 화면)	960 * 540	15
2x2 (4 자 화면)	640 * 360	12
3x2 (6 자 화면)	480 * 270	12
3x3 (9 자 화면)	320 * 180	12
4x3 (12 자 화면)	320 * 180	10
4x4 (16 자 화면)	240 * 135	10
5x4 (20 자 화면)	160 * 120	10
5x5 (25 자 화면)	160 * 120	8
6x5 (30 자 화면)	120 * 80	8
6x6 (36 자 화면)	120 * 80	8

- III. 화면 공유 정책 (해상도는 최소[120*80, 8f]로 유지하고, 공유 화면 해상도는 640*360, 15f으로 유지)

4. 연동 방법 (상용 정보는 별도로 제공)

<연동 정보>

- Signal Server (TB) URL : <https://dev.knowledgetalk.co.kr:7103>
- Signal Server (TB) Socket URL : <https://dev.knowledgetalk.co.kr:7103/SignalServer>
 - * 상기 URL로 Socket 연결이 필요합니다. (socket.io)
 - * 모든 규격은 JSON(Object) 형태로 전송해야 합니다.
 - * Room ID, User ID는 Signal Server에서 발급 합니다.
 - * Signal Server URL은 TB URL 이며, 추후 서비스 시 상용 URL 전달 예정입니다.
 - * 인증서 오류 시 해당 URL을 웹 브라우저에 입력 후 고급 -> 안전하지 않음 이동 버튼 누른 후 사용하기 바랍니다.
- Turn URL : <turn:turn.knowledgetalk.co.kr:46000> / username: kpoint / credential: kpoint01
- Stun URL : <stun:stun.knowledgetalk.co.kr:46000>
 - * 본 내용은 TB 연동 정보이며, 상용 연동 정보는 TB 연동 검증(상용 연동에 문제가 없는지 검토)후 별도 전달 예정입니다.

<연동 Call Flow>

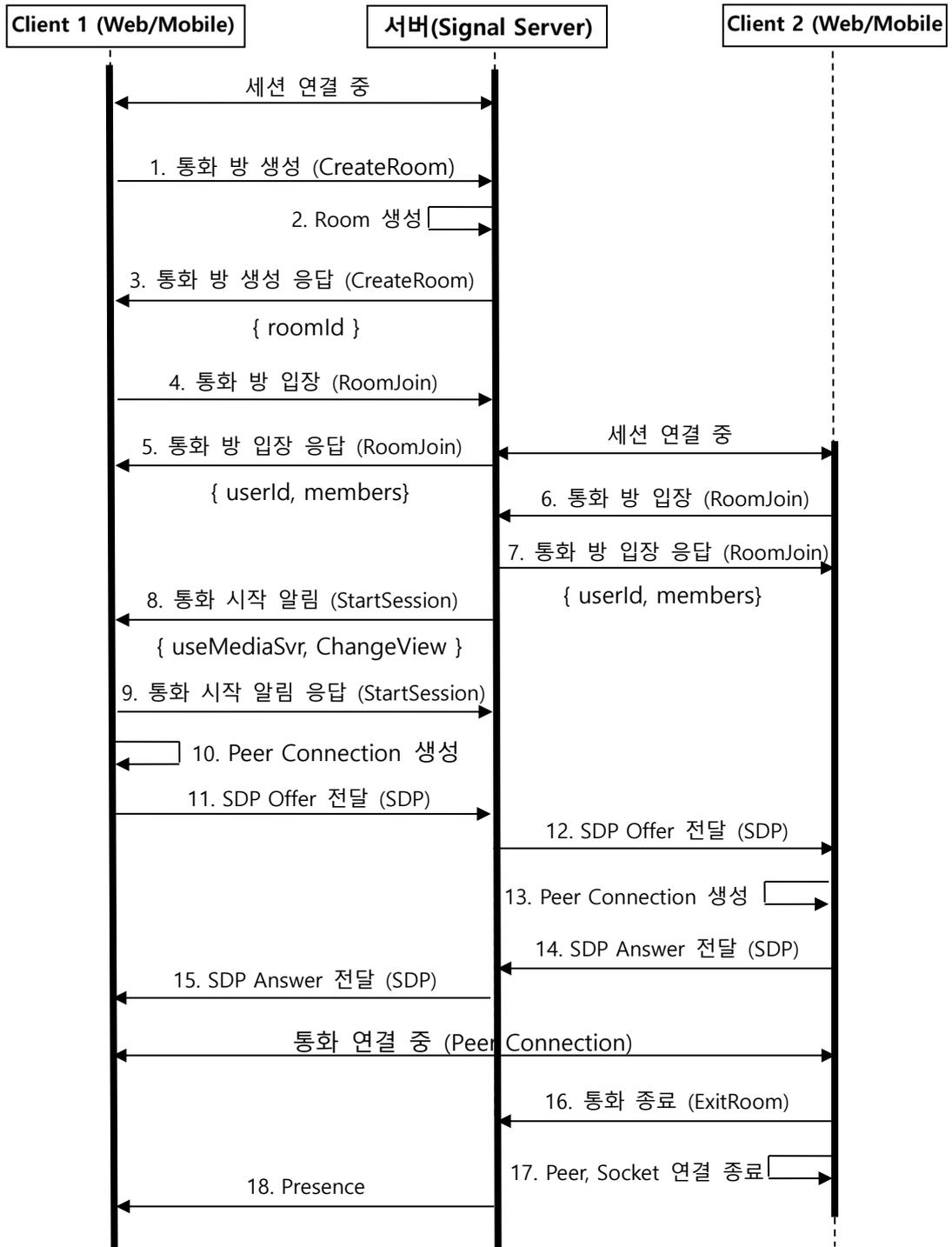
● Session 연결 및 Keep Alive (주기적 세션 체크) Call Flow (필수 구현 사항)

⇒ 서버에서 Keep Alive 응답이 오지 않으면 (아래 플로우 중 4번) Session이 종료된 것으로 판단하고, 재 연결을 한다. (1. 세션 연결 시도, 재 시도 3번까지, 재 시도에도 연결이 되지 않으면 noti 메시지를 사용자에게 보여주고 서비스를 종료한다.)



단계	설명	비고
1. 세션 연결	Socket.IO 로 Signal 서버에 연동한다. 예) <code>io.connect('https://dev.knowledgetalk.co.kr:7103', { reconnect: true, 'transports': ['websocket'] })</code>	
2. 세션 유지	기본 세션 유지는 60 초이며, 60 초 안에 KeepAlive 메시지 미 전송 시 세션이 종료된 것으로 판단	
3. 세션 유지 요청	Keep Alive 메시지를 서버로 전송하여 세션이 유지되고 있는지 체크 요청 (60 초 Session Timeout 전에 주기적으로 요청 필요.)	
4. 세션 유지 응답	서버는 단말에게 유지 상태에 대해 응답하고, 다시 세션 유지 시간을 60 초로 갱신한다.	

● 통화 시작 및 통화 종료 Call Flow (1:1 통화)

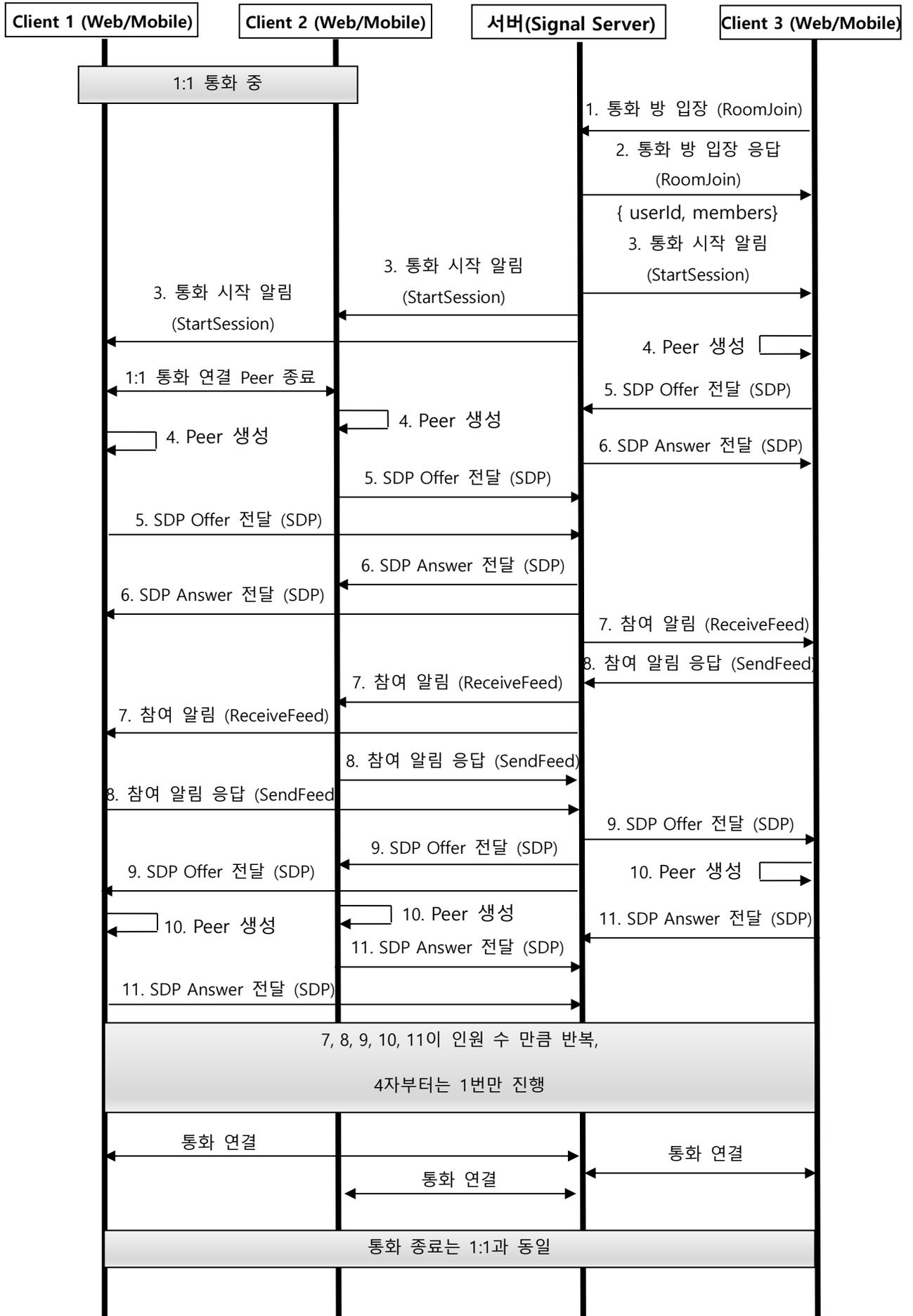


단계	설명	비고
1. 통화 방 생성	영상 통화 방을 생성하고자 하는 사용자가 서버로 방 생성 요청	
2. Room 생성	서버에서 통화 방의 ID (번호)를 발급한다.	

3. 통화 방 생성 응답	발급 된 Room ID 를 응답 메시지로 클라이언트에게 전달한다.	
4. 통화 방 입장	통화 방에 입장 요청	
5. 통화 방 입장 응답	통화 방 입장 완료 및 User ID 발급, 응답 메시지의 member 필드로 현재 방에 입장한 인원 확인 가능	
6. 통화 방 입장	통화 방에 입장 요청	
7. 통화 방 입장 응답	통화 방 입장 완료 및 User ID 발급, 응답 메시지의 member 필드로 현재 방에 입장한 인원 확인 가능	
8. 통화 시작 알림	서버에서 통화 시작을 알림. useMediaSvr 필드로 다자 통화인지 1:1 통화인지 구분, changeView 필드로 1:1 -> 다자 변경인지 구분. 해당 메시지를 받은 클라이언트는 PeerConnection 을 생성.	
9. 통화 시작 알림 응답	잘 받았다면 200, OK 전송	
10. Peer Connection 생성	Peer Connection 객체 생성하고, SDP Offer 생성한 후 SetLocalDescription()	해상도는 960*540, 15frame
11. SDP Offer 전달	SDP Offer 를 서버로 전달 (이 때 usage 를 cam 으로 전달)	
12. SDP Offer 전달	SDP Offer 를 상대방에게 전달.	
13. Peer Connection 생성	Peer Connection 을 생성하고, SetRemoteDescription() 및 SDP Answer 생성	
14. SDP Answer 전달	SDP Answer 를 서버로 전달 (이 때 usage 를 cam 으로 전달)	
15. SDP Answer 전달	SDP Answer 를 상대방에게 전달.	
16. 통화 종료	통화 종료 메시지를 서버로 전달.	
17. Peer, Socket 종료	Peer Connection 과 Socket 연결을 종료한다.	
18. Presence	퇴장 한 사용자에게 대한 정보를 상대방에게 전달	

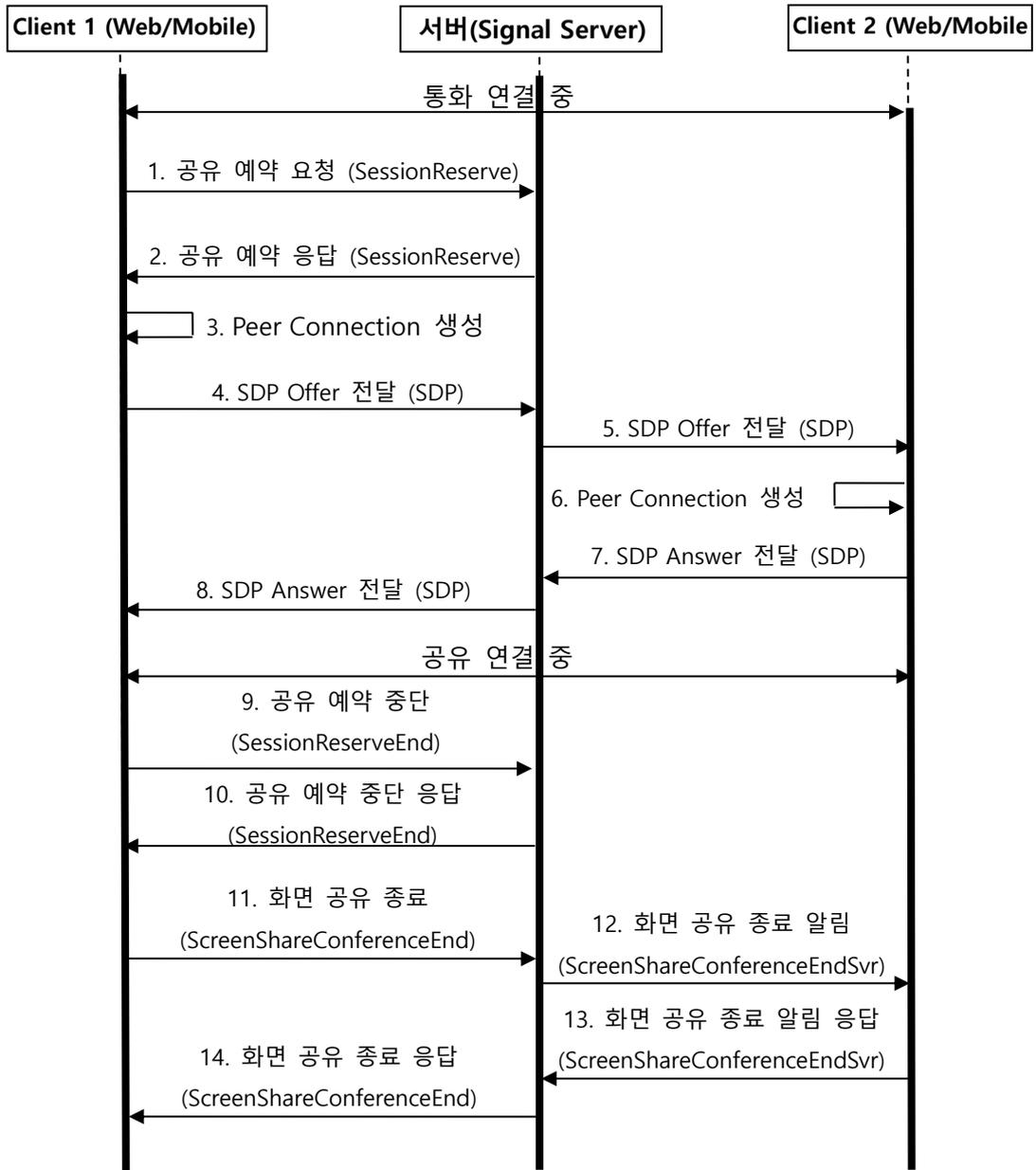
● 통화 시작 및 통화 종료 Call Flow (다자 통화)

⇒ StartSession 메시지의 count 필드를 통해 참여 인원 수 정보를 받으면 맞는 해상도로 설정 (필수)



단계	설명	비고
1. 통화 방 입장	통화 방에 입장 요청	
2. 통화 방 입장 응답	통화 방 입장 완료 및 User ID 발급, 응답 메시지의 member 필드로 현재 방에 입장한 인원 확인 가능	
3. 통화 시작 알림	서버에서 통화 시작을 알림. 받은 host 필드를 SDP 전송 시 그대로 전송 필요. useMediaSvr 필드로 다자 통화인지 1:1 통화인지 구분, changeView 필드로 1:1 -> 다자 변경인지 구분. 해당 메시지를 받은 클라이언트는 PeerConnection 을 생성.	
4. Peer Connection 생성	Peer Connection 을 생성하고, SDP Offer 생성한 후 SetLocalDescription()	
5. SDP Offer 전달	SDP Offer 를 미디어서버로 전달. (이 때 usage 를 cam 으로 전달하며, StartSession 에서 받은 host 필드를 넣어 전송)	
6. SDP Answer 전달	SDP Answer 를 클라이언트에게 전달.	
7. 참여 알림	통화 방에 입장한 사용자의 영상을 볼 것인지에 대한 확인 요청.	
8. 참여 알림 응답	ReceiveFeed 에서 받았던 feeds 필드 내의 feedId 와 displayId, roomId 전송	
9. SDP Offer 전달	SendFeed 메시지를 보냈다면, SDP Offer 를 전달 받을 수 있음.	
10. Peer Connection 생성	Peer Connection 을 생성하고, SetRemoteDescription() 및 SDP Answer 생성	
11. SDP Answer 전달	SDP Answer 를 미디어서버로 전달. (이 때 usage 를 cam 으로 전달)	

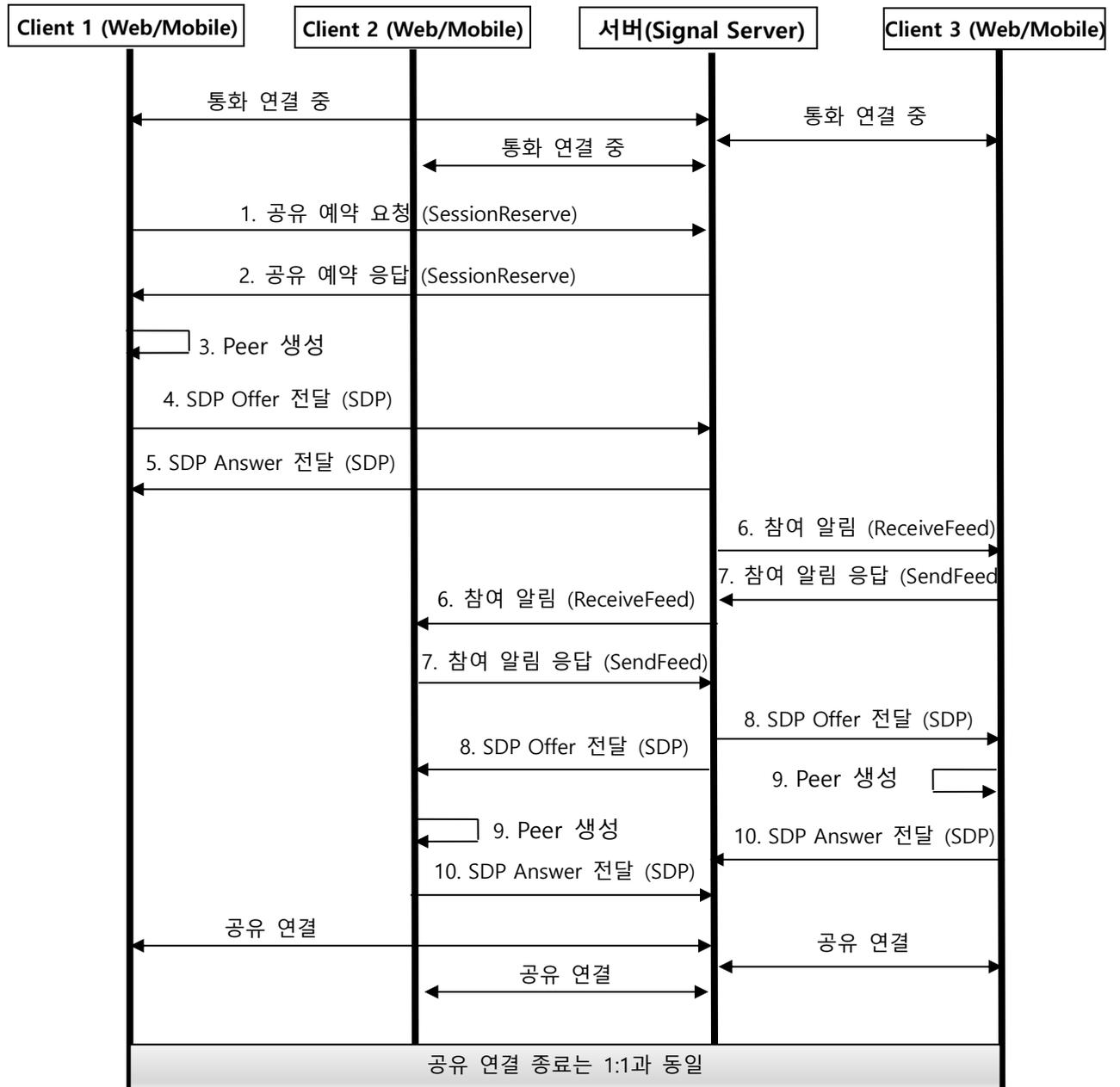
● 회의 중 화면 공유 Call Flow (1:1 통화 중)



단계	설명	비고
1. 공유 예약 요청	화면공유 진행을 예약. 진행 중에는 다른 사용자가 공유를 할 수 없도록 예약 요청	
2. 공유 예약 응답	200 OK 의 경우 PeerConnection 을 생성하고 SDP Offer 를 생성하면 되고, 그 외의 Resp code 는 에러 처리	
3. Peer Connection 생성	Peer Connection 을 생성하고, SDP Offer 생성한 후 SetLocalDescription()	
4. SDP Offer 전달	SDP Offer 를 서버로 전달 (이 때 usage 를 screen 으로 전달)	
5. SDP Offer 전달	SDP Offer 를 상대방에게 전달.	

6. Peer Connection 생성	Peer Connection 을 생성하고, SetRemoteDescription() 및 SDP Answer 생성	
7. SDP Answer 전달	SDP Answer 를 서버로 전달 (이 때 usage 를 screen 으로 전달)	
8. SDP Answer 전달	SDP Answer 를 상대방에게 전달.	
9. 공유 예약 중단	다른 사용자가 화면 공유를 사용할 수 있도록 예약을 중단 요청	
10. 공유 예약 중단 응답	서버에서 응답 전송	
11. 화면 공유 종료	화면 공유 종료 이벤트 전송, 공유 Peer 종료	
12. 화면 공유 종료 알림	해당 메시지를 받은 사용자는 공유 Peer 를 끊어야함.	
13. 화면 공유 종료 알림 응답	메시지를 잘 받았는지에 대한 응답 (클라이언트 전송)	
14. 화면 공유 종료 응답	상대방이 화면 공유 종료 알림을 잘 받았는지에 대한 응답	

● 회의 중 화면 공유 Call Flow (다자 통화 시)



단계	설명	비고
1. 공유 예약 요청	화면공유 진행을 예약. 진행 중에는 다른 사용자가 공유를 할 수 없도록 예약 요청	
2. 공유 예약 응답	200 OK 의 경우 PeerConnection 을 생성하고 SDP Offer 를 생성하면 되고, 그 외의 Resp code 는 에러 처리	
3. Peer Connection 생성	Peer Connection 을 생성하고, SDP Offer 생성한 후 SetLocalDescription()	
4. SDP Offer 전달	SDP Offer 를 서버로 전달 (이 때 usage 를 screen 으로 전달)	
5. SDP Answer 전달	SDP Answer 를 클라이언트에게 전달.	

6. 참여 알림	공유 영상을 볼 것인지에 대한 확인 요청.	
7. 참여 알림 응답	ReceiveFeed 에서 받았던 feeds 필드 내의 feedId 와 displayId, roomId 전송	
8. SDP Offer 전달	SendFeed 메시지를 보냈다면, SDP Offer 를 전달 받을 수 있음.	
9. Peer Connection 생성	Peer Connection 을 생성하고, SetRemoteDescription() 및 SDP Answer 생성	
10. SDP Answer 전달	SDP Answer 를 미디어서버로 전달. (이 때 usage 를 cam 으로 전달)	

5. API 목차 (각 항목에 ctrl + click 시 해당 API로 이동합니다.)

- API 리스트
- 6.1. [통화 방 생성 \(CreateRoom \)](#)
- 6.2. [통화 방 입장 \(RoomJoin \)](#)
- 6.3. [통화 시작 알림 \(StartSession \)](#)
- 6.4. [SDP \(SDP \)](#)
- 6.5. [상대방 참여 정보 \(ReceiveFeed \)](#)
- 6.6. [상대방 참여 확인 \(SendFeed \)](#)
- 6.7. [Presence](#)
- 6.8. [공유 예약 \(SessionReserve \)](#)
- 6.9. [공유 예약 종료 \(SessionReserveEnd \)](#)
- 6.10. [화면 공유 종료 \(ScreenShareConferenceEnd \)](#)
- 6.11. [KeepAlive \(KeepAlive \)](#)
- 6.12. [통화 방 나가기 \(ExitRoom\)](#)
- 6.13. [영상 On/Off \(SetVideo \)](#)
- 6.14. [음성 On/Off \(SetAudio \)](#)
- 6.15. [이름 변경 \(ChangeName \)](#)

6. 연동 API 규격

6.1. 통화 방 생성 (CreateRoom)

Request Data

Parameter	Required	Type	Default	Description
cpCode	○	String		라이선스 인증 코드
authKey	○	String		라이선스 인증 키
eventOp	○	String		Op (CreateRoom)

Request Example

Example	Description
<pre>{ "cpCode": "ABCMART", "authKey": "ABCDEFG...", "eventOp": "CreateRoom" }</pre>	통화 방 생성 요청

Response Data

Parameter (1depth)	Parameter (2depth)	Parameter (3depth)	Type	Description
eventOp			String	Op (CreateRoom)
code			String	응답 코드
message			String	응답 메시지
roomId			String	방 ID

Response Example

Example	Description
<pre>{ "eventOp": "CreateRoom", "code": 200, "message": "OK", "roomId": "12345678" }</pre>	파라미터가 잘못 된 경우 코드 번호 및 메시지가 200, OK가 아닌 상단에 정의된 에러코드로 응답 될 수 있음 (에러 코드 정의 참고).

6.2. 통화 방 입장 (RoomJoin)

Request Data

Parameter	Required	Type	Default	Description
cpCode	○	String		라이선스 인증 코드
authKey	○	String		라이선스 인증 키
eventOp	○	String		Op (RoomJoin)
roomId	○	String		방 번호

Request Example

Example	Description
<pre>{ "cpCode": "ABCMART", "authKey": "ABCDEFG...", "eventOp": "RoomJoin", "roomId": "52772290" }</pre>	통화 방 입장 요청

Response Data

Parameter (1depth)	Parameter (2depth)	Parameter (3depth)	Type	Description
eventOp			String	RoomJoin
code			String	응답 코드
members			Object	방 인원 정보
message			String	응답 메시지
roomId			String	방 ID
userId			String	유저 ID

Response Example

Example	Description
<pre>{ "eventOp": "RoomJoin", "code": "200", "members": {u9195689: {...}}, "message": "OK", "roomId": "52772290", "userId": "u9195689"} </pre>	파라미터가 잘못 된 경우 코드 번호 및 메시지가 200, OK가 아닌 상단에 정의된 에러코드로 응답 될 수 있음 (에러 코드 정의 참고).

6.3. 통화 시작 알림 (StartSession)

Request Data (Server)

Parameter	Required	Type	Default	Description
cpCode	O	String		라이선스 인증 코드
authKey	O	String		라이선스 인증 키
eventOp	O	String		Op (StartSession)
members	O	Object		방 접속 유저 정보
changeView	X	Boolean		1:1 -> 다자 변경 여부
useMediaSvr	O	String		유저 미디어 확인 체크
who	O	String		참석자 정보
host	X	Boolean		다자 상황에서 Host 구분하기 위한 flag. 해당 내용을 받으면 SDP에 host 필드 추가하여 전송 필요

Request Example

Example	Description
<pre>{ "cpCode": "ABCMART", "authKey": "ABCDEFG...", "eventOp": "StartSession", "members": {u9195689: {...}, u3658110: {...}, u3653333: {...}}, "changeView": true, "useMediaSvr": "N", "who": "u3658110", "host": true }</pre>	통화 시작 알림

Response Data (Client)

Parameter (1depth)	Parameter (2depth)	Parameter (3depth)	Type	Description
eventOp			String	StartSession
code			String	응답 코드

Response Example

Example	Description
<pre>{ "eventOp": "StartSession", "code": "200" }</pre>	파라미터가 잘못 된 경우 코드 번호 및 메시지가 200, OK가 아닌 상단에 정의된 에러코드로 응답 될 수 있음 (에러 코드 정의 참고).

6.4. SDP (SDP)

Request Data

Parameter	Required	Type	Default	Description
cpCode	O	String		라이선스 인증 코드
authKey	O	String		라이선스 인증 키
eventOp	O	String		OP (SDP)
pluginId	X	String		다자 상황에서만 사용, Offer로 전달받은 pluginID를 전송
roomId	O	String		Op (RoomJoin)
sdp	O	Object		SDP 정보
usage	O	String		SDP 타입 ('cam' 'screen')
userId	O	String		사용자 ID
host	X	boolean		다자 상황에서 StartSession에서 전달받은 host 데이터를 이곳에 추가하여 전송

Request Example

Example	Description
<pre>{ "cpCode": "ABCMART", "authKey": "ABCDEFGH...", "eventOp": "SDP", "pluginId": "1231233", "roomId": "52772290", "sdp": RTCSessionDescription {type: "answer", sdp: ...}, "usage": "cam", "userId": "u9195689", "host": true, }</pre>	SDP 전송 요청

Response Data

Parameter (1depth)	Parameter (2depth)	Parameter (3depth)	Type	Description
eventOp			String	SDP
code			String	응답 코드

Response Example

Example	Description
<pre>{ "eventOp": "SDP", "code": "200" }</pre>	파라미터가 잘못 된 경우 코드 번호 및 메시지가 200, OK가 아닌 상단에 정의된 에러코드로 응답 될 수 있음 (에러 코드 정의 참고).

6.5. 상대방 참여 정보 (ReceiveFeed)

Request Data (Server)

Parameter	Required	Type	Default	Description
cpCode	○	String		라이선스 인증 코드
authKey	○	String		라이선스 인증 키
eventOp	○	String		Op (ReceiveFeed)
feeds	○	Array		Feed 정보 (Feed의 수 만큼 SendFeed 전송 필요)
roomId	○	String		방 ID번호

Request Example

Example	Description
<pre>{ "cpCode": "ABCMART", "authKey": "ABCDEFGH...", "eventOp": "ReceiveFeed", "feeds": [{...}], "roomId": "90906416" }</pre>	상대방 참여 정보

Response Data (Client)

Parameter (1depth)	Parameter (2depth)	Parameter (3depth)	Type	Description
eventOp			String	ReceiveFeed
code			String	응답 코드

Response Example

Example	Description
<pre>{ "eventOp": "ReceiveFeed", "code": "200" }</pre>	파라미터가 잘못 된 경우 코드 번호 및 메시지가 200, OK가 아닌 상단에 정의된 에러코드로 응답 될 수 있음 (에러 코드 정의 참고).

6.6. 상대방 참여 확인 (SendFeed)

Request Data (Client)

Parameter	Required	Type	Default	Description
cpCode	○	String		라이선스 인증 코드
authKey	○	String		라이선스 인증 키
display	○	String		ReceiveFeed에서 받은 displayId 정보
eventOp	○	Object		Op (SendFeed)
feedId	○	String		ReceiveFeed에서 받은 FeedID 정보
roomId	○	String		방 ID번호

Request Example

Example	Description
<pre>{ "cpCode": "ABCMART", "authKey": "ABCDEFG...", "display": "/SignalServer#l31fwtv2eNEwypCBAAAd" "eventOp": "SendFeed" "feedId": "8bf65aa9-5b18-4e10-86a1-9b1a81f0d010" "roomId": "90906416"} </pre>	상대방 참여 정보 확인

Response Data (Server)

Parameter (1depth)	Parameter (2depth)	Parameter (3depth)	Type	Description
eventOp			String	SendFeed
code			String	응답 코드

Response Example

Example	Description
<pre>{ "eventOp": "SendFeed", "code": "200" }</pre>	파라미터가 잘못 된 경우 코드 번호 및 메시지가 200, OK가 아닌 상단에 정의된 에러코드로 응답 될 수 있음 (에러 코드 정의 참고).

6.7. Presence

Request Data

Parameter	Required	Type	Default	Description
cpCode	○	String		라이선스 인증 코드
authKey	○	String		라이선스 인증 키
signalOp	○	String		Op (Presence)
userId	○	String		참여 혹은 퇴장 인원의 userId
action	○	String		exit (방을 나간 사용자가 있는 경우) join (방에 참여한 사용자가 있는 경우)

Request Example

Example	Description
<pre>{ "cpCode": "ABCMART", "authKey": "ABCDEFGH...", "signalOp": "Presence", "userId": "u4893289", "action": "exit" }</pre>	Presence

Response Data

Parameter (1depth)	Parameter (2depth)	Parameter (3depth)	Type	Description
N/A				

Response Example

Example	Description

6.8. 공유 예약 (SessionReserve)

Request Data

Parameter	Required	Type	Default	Description
cpCode	○	String		라이선스 인증 코드
authKey	○	String		라이선스 인증 키
eventOp	○	String		Op (SessionReserve)
userId	○	String		공유를 예약하고자 하는 사용자의 ID
roomId	○	String		방번호 (ID)

Request Example

Example	Description
<pre>{ "cpCode": "ABCMART", "authKey": "ABCDEFGH...", "eventOp": "SessionReserve" "userId": "u0022745", "roomId": "90906416" }</pre>	공유 예약

Response Data

Parameter (1depth)	Parameter (2depth)	Parameter (3depth)	Type	Description
eventOp			String	RoomJoin
code			String	응답 코드
message			String	응답 메시지

Response Example

Example	Description
<pre>{ "eventOp": "SessionReserve", "code": "200", "message": "OK" }</pre>	파라미터가 잘못 된 경우 코드 번호 및 메시지가 200, OK가 아닌 상단에 정의된 에러코드로 응답 될 수 있음 (에러 코드 정의 참고).

6.9. 공유 예약 종료 (SessionReserveEnd)

Request Data

Parameter	Required	Type	Default	Description
cpCode	○	String		라이선스 인증 코드
authKey	○	String		라이선스 인증 키
eventOp	○	String		Op (SessionReserveEnd)
roomId	○	String		방번호 (ID)
userId	○	String		UserID

Request Example

Example	Description
<pre>{ "cpCode": "ABCMART", "authKey": "ABCDEFG...", "eventOp": "SessionReserveEnd" "roomId": "90906416" "userId": "u0022745" }</pre>	공유 예약

Response Data

Parameter (1depth)	Parameter (2depth)	Parameter (3depth)	Type	Description
eventOp			String	RoomJoin
code			String	응답 코드
message			String	응답 메시지

Response Example

Example	Description
<pre>{ "eventOp": "SessionReserve", "code": "200", "message": "OK" }</pre>	파라미터가 잘못 된 경우 코드 번호 및 메시지가 200, OK가 아닌 상단에 정의된 에러코드로 응답 될 수 있음 (에러 코드 정의 참고).

6.10. 화면 공유 종료 (ScreenShareConferenceEnd)

Request Data

Parameter	Required	Type	Default	Description
cpCode	○	String		라이선스 인증 코드
authKey	○	String		라이선스 인증 키
eventOp	○	String		Op (ScreenShareConferenceEnd)
roomId	○	String		방번호 (ID)
useMediaSvr	○	String		다자 상황 여부 (Y or N)
userId	○	String		UserID

Request Example

Example	Description
<pre>{ "cpCode": "ABCMART", "authKey": "ABCDEFGH...", "eventOp": "ScreenShareConferenceEnd" "roomId": "90906416" "useMediaSvr": "N" "userId": "u0022745" }</pre>	화면 공유 종료

Response Data

Parameter (1depth)	Parameter (2depth)	Parameter (3depth)	Type	Description
eventOp			String	RoomJoin
roomId			String	방 번호 (ID)
code			String	응답 코드
message			String	응답 메시지

Response Example

Example	Description
<pre>{ "eventOp": "ScreenShareConferenceEnd", "roomId": "90906416", "code": "200", "message": "OK" }</pre>	파라미터가 잘못 된 경우 코드 번호 및 메시지가 200, OK가 아닌 상단에 정의된 에러코드로 응답 될 수 있음 (에러 코드 정의 참고).

6.11. KeepAlive (KeepAlive)

Request Data

Parameter	Required	Type	Default	Description
cpCode	○	String		라이선스 인증 코드
authKey	○	String		라이선스 인증 키
eventOp	○	String		Op (KeepAlive)

Request Example

Example	Description
<pre>{ "cpCode": "ABCMART", "authKey": "ABCDEFG...", "eventOp": "KeepAlive" }</pre>	Session 연결 유지를 위한 Message (60초 동안 Message가 들어오지 않으면 연결이 끊어집니다.)

Response Data

Parameter (1depth)	Parameter (2depth)	Parameter (3depth)	Type	Description
eventOp			String	KeepAlive
code			String	응답 코드
message			String	응답 메시지

Response Example

Example	Description
<pre>{ "eventOp": "KeepAlive", "code": "200", "message": "OK" }</pre>	파라미터가 잘못 된 경우 코드 번호 및 메시지가 200, OK가 아닌 상단에 정의된 에러코드로 응답 될 수 있음 (에러 코드 정의 참고).

6.12. 통화 방 나가기 (ExitRoom)

Request Data

Parameter	Required	Type	Default	Description
cpCode	○	String		라이선스 인증 코드
authKey	○	String		라이선스 인증 키
eventOp	○	String		Op (ExitRoom)
roomId	○	String		방 번호
userId	○	String		RoomJoin시 할당 받은 User ID

Request Example

Example	Description
<pre>{ "cpCode": "ABCMART", "authKey": "ABCDEFG...", "eventOp": "ExitRoom" "roomId": "90906416" "userId": "u0022745" }</pre>	통화 방 퇴장 요청

Response Data

Parameter (1depth)	Parameter (2depth)	Parameter (3depth)	Type	Description
eventOp			String	KeepAlive
code			String	응답 코드
message			String	응답 메시지

Response Example

Example	Description
<pre>{ "eventOp": "ExitRoom", "code": "200", "message": "OK" }</pre>	파라미터가 잘못 된 경우 코드 번호 및 메시지가 200, OK가 아닌 상단에 정의된 에러코드로 응답 될 수 있음 (에러 코드 정의 참고).

6.13. 영상 On/Off (SetVideo)

Request Data

Parameter	Required	Type	Default	Description
cpCode	O	String		라이선스 인증 코드
authKey	O	String		라이선스 인증 키
signalOp	O	String		Op (SetVideo)
roomId	O	String		방 번호 (ID)
status	O	Boolean		true: 영상 Off, false: 영상 On

Request Example

Example	Description
<pre>{ "cpCode": "ABCMART", "authKey": "ABCDEFG...", "signalOp": "SetVideo", "roomId": "90906416", "status": true }</pre>	본인 영상 On/Off 여부 전달

Response Data

Parameter (1depth)	Parameter (2depth)	Parameter (3depth)	Type	Description
N/A				

Response Example

Example	Description

6.14. 음성 On/Off (SetAudio)

Request Data

Parameter	Required	Type	Default	Description
cpCode	○	String		라이선스 인증 코드
authKey	○	String		라이선스 인증 키
signalOp	○	String		Op (SetAudio)
roomId	○	String		방 번호 (ID)
status	○	Boolean		true: 음성 Off, false: 음성 On

Request Example

Example	Description
<pre>{ "cpCode": "ABCMART", "authKey": "ABCDEFGH...", "signalOp": "SetAudio", "roomId": "90906416", "status": true }</pre>	본인 음성 On/Off 여부 전달

Response Data

Parameter (1depth)	Parameter (2depth)	Parameter (3depth)	Type	Description
N/A				

Response Example

Example	Description

6.15. 이름 변경 (ChangeName)

Request Data

Parameter	Required	Type	Default	Description
cpCode	○	String		라이선스 인증 코드
authKey	○	String		라이선스 인증 키
signalOp	○	String		Op (ChangeName)
roomId	○	String		방 번호 (ID)
userId	○	String		사용자 ID
name	○	String		변경 할 이름

Request Example

Example	Description
<pre>{ "cpCode": "ABCMART", "authKey": "ABCDEFG...", "signalOp": "ChangeName" "roomId": "90906416", "userId": "u0022745", "name": "홍길동" }</pre>	이름 변경 요청

Response Data

Parameter (1depth)	Parameter (2depth)	Parameter (3depth)	Type	Description
N/A				

Response Example

Example	Description