

CodeJam

Week 5 Demo (2026.01.23)

실시간 협업 코딩 플랫폼

Team JAMstack

J044 김선향

J094 노주호

J102 민인애

J141 손혜준

J143 송상화

<https://github.com/boostcampwm2025/web08-JAMstack>

<https://lets-codejam.vercel.app/>

| 목차

01. Intro

서비스 소개

5주차 주요 성과

02. Core Features

방 모드 이원화

인프라 안정화

UI/UX 고도화

03. Wrap Up

기술적 도전

향후 계획 및 데모

Q & A

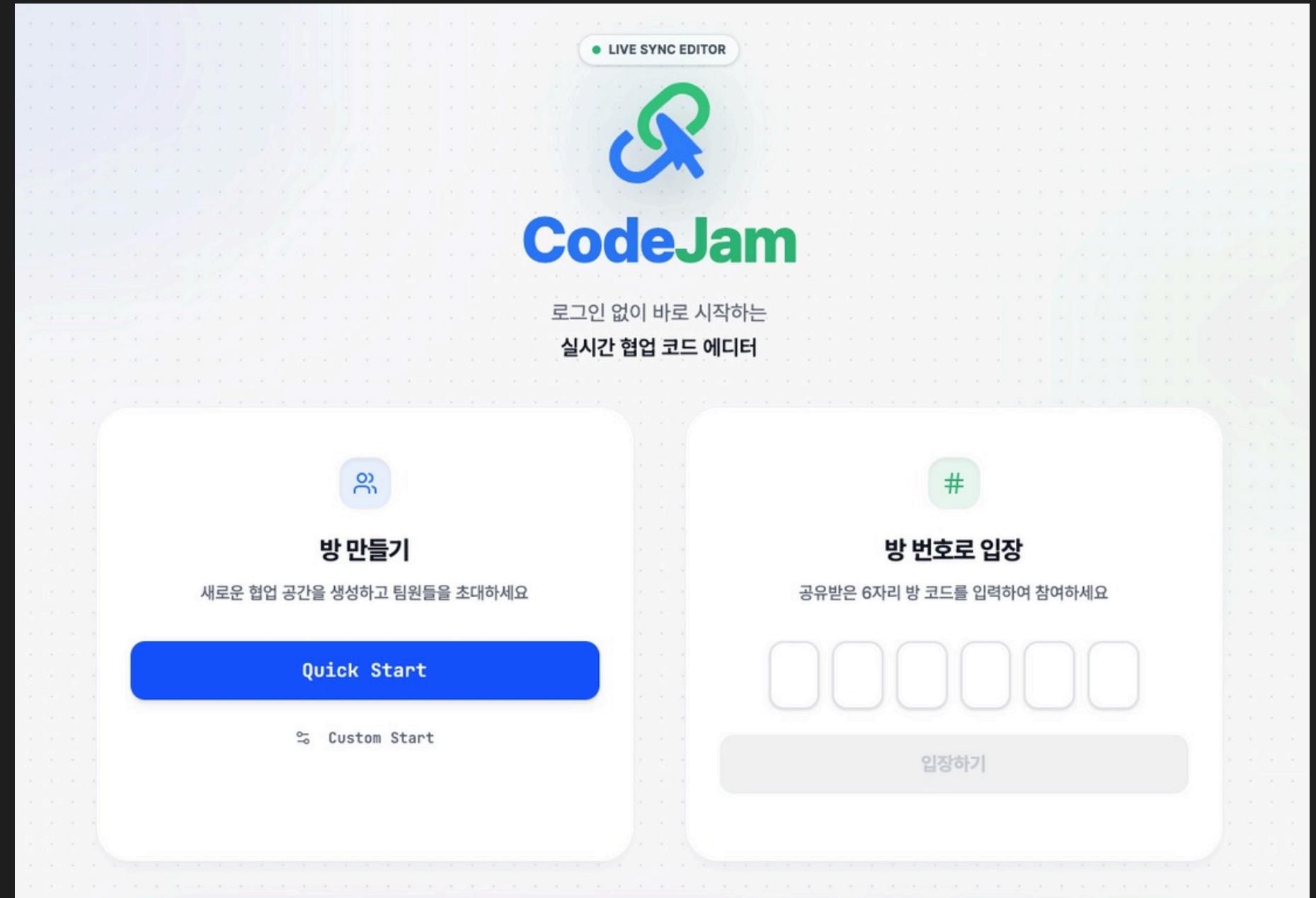
| 1. 서비스 소개

핵심 가치

- > **Speed:** 로그인 없이 클릭 한 번으로 즉시 시작
- > **Lightweight:** 필수 기능만 담은 빠른 경험
- > **Real-time:** CRDT(Yjs) 기반 실시간 동시 편집

타겟 사용자

- > 코딩 면접관/지원자, 페어 프로그래밍 팀
- > 알고리즘 스터디, 대학 강의 (최대 150명)



| 2. 5주차 주요 성과



방 모드 이원화

Quick/Custom Start 분리

비밀번호 시스템, 커스텀 방 UI 구축 완료



방 관리 시스템

방 나가기/폭파, 닉네임 변경, 파일 관리

권한 관리(호스트 양도/탈취)



인프라 안정화

운영/개발 환경 분리

Health Check 도입, Redis 메모리 최적화



UI/UX 고도화

Awareness(커서/아바타)

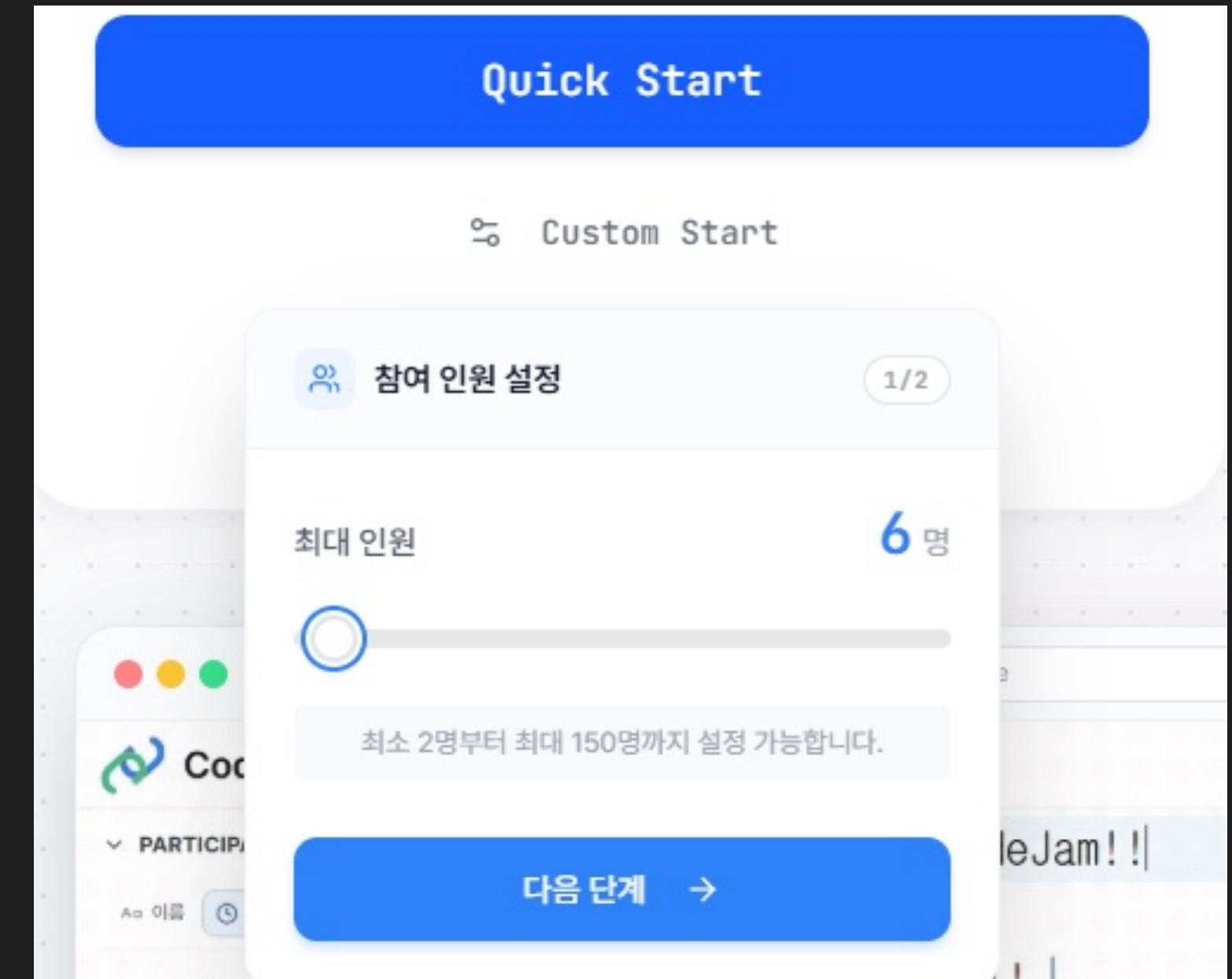
메인 홈페이지 디자인 리뉴얼, CLI 패키지

| 3. 방 모드 이원화 (Quick vs Custom)

문제 인식 및 해결

기존에는 6인 소규모 협업과 150인 강의를 하나의 로직으로 처리하여 기술적, 기획적으로 정책 충돌이 있었습니다.

Solution: 용도별 방 타입 완전 분리



| Quick Start (퀵 스타트)

특징

- > 최대 6인 제한 (소규모 최적화)
- > 수평적 협업 모드 (모두 Editor 권한)
- > 별도 호스트 없음
- > 비밀번호 설정 불가
- > 생성자도 초기 닉네임 설정

타겟

- > 페어 프로그래밍
- > 소규모 알고리즘 스터디
- > 즉석 코드 리뷰
- > 빠른 프로토타이핑 협업

| Custom Start (커스텀 스타트)

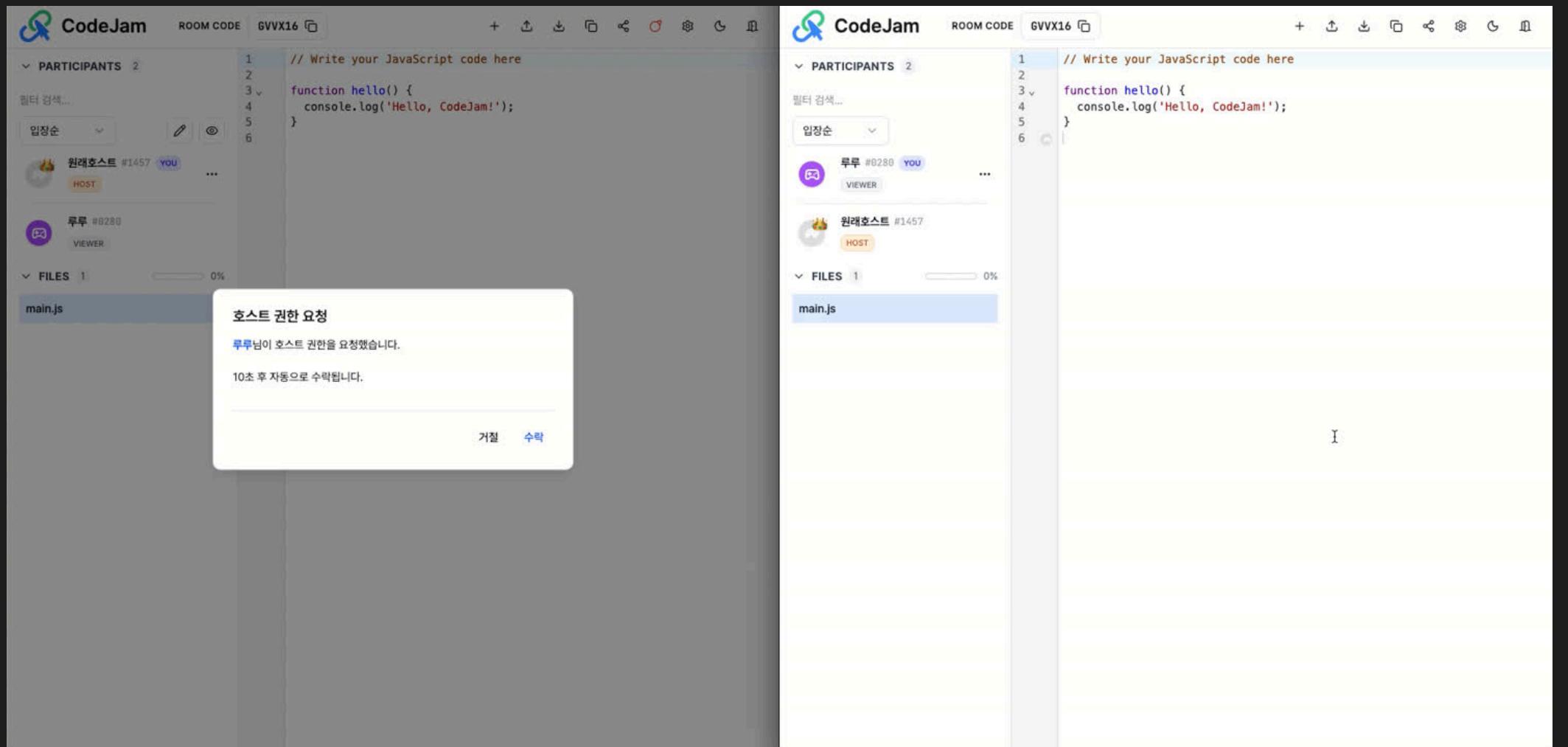
특징

- > 최대 150인 수용 (대규모 최적화)
- > 명확한 호스트 중심 제어 시스템
- > 비밀번호 설정 가능 (입장/호스트 PW)
- > Step-by-step UI (인원 → 보안 설정)
- > 수동 권한 관리 (Host/Editor/Viewer)

타겟

- > 대학 강의 및 실습
- > 코딩 테스트 감독
- > 대규모 기술 세미나

| 보안 및 권한 관리 시스템



비밀번호 시스템

- > 입장 시 비밀번호 검증 (Token 기반 재입장)
- > Join Room Payload에 password 필드 암호화 전송

호스트 권한 탈취

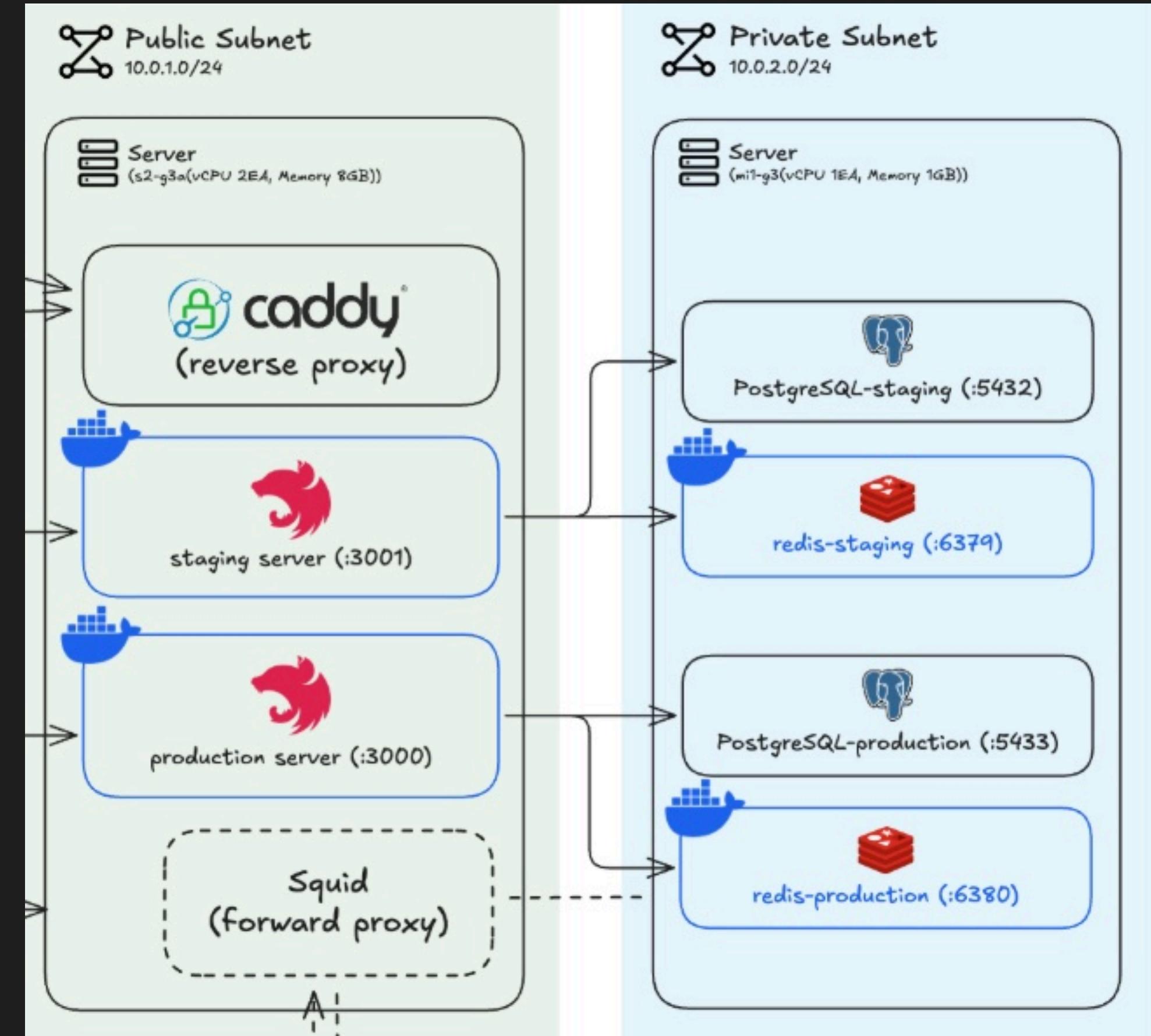
- > 비호스트가 호스트 비밀번호 입력 시도
- > 기존 호스트에게 알림 발송 (10초 카운트다운)
- > 수락 시 권한 교환, 거절/Timeout 시 요청 취소
- > 동시 요청 차단 및 요청자 퇴장 시 자동 취소

4. 인프라 안정화

운영/개발 환경 분리

기존 단일 DB 사용으로 인한 스키마 충돌 및 서비스 중단 위험을 해결했습니다.

- > **PostgreSQL**: 5432(Dev) / 5433(Prod) 포트 분리
- > **Redis**: 6379(Dev) / 6380(Prod) 포트 분리
- > NCP ACG 인바운드 규칙으로 엄격한 접근 제어



| 5. UI/UX 고도화 (Awareness)

리모트 커서 시스템

Yjs Awareness와 로컬 스토어를 단방향 동기화하여 타 사용자의 커서 위치, 닉네임 라벨, 고유 색상을 실시간으로 표시합니다.

라인 아바타 (Gutter)

CodeMirror Gutter 영역에 사용자 아바타를 표시하여 현재 라인에 누가 있는지 직관적으로 확인 가능합니다.

The screenshot shows a CodeJam interface with a dark theme. At the top, it displays 'CodeJam' with a logo, 'ROOM ID 5369L3', and a participant list. Below this, the 'PARTICIPANTS' section lists five users: 'Host #6575 YOU' (with a game controller icon), '1 #3897' (with a person icon), '2 #0273' (with a yellow circle icon), and '3 #3577' (with a blue circle icon). The 'FILES' section shows one file with a progress bar at 1%. On the right, a code editor window is open, showing a snippet of JavaScript code:

```
// Write your JavaScript code here
function hello() {
    console.log('Hello, CodeJam!');
}
text 입력
oh wow~
wowo~!!!!
type
yes~!!
yes
4 Write your JavaScript code here
function hello() {
    console.log('Hello, CodeJam!');
}
text
```

The code editor includes line numbers from 1 to 21. Above the code, there are several small icons with numbers (1, 2, 3, 4) and colored dots (blue, red, green, purple) corresponding to the participants listed in the gutter.

| UI/UX 고도화 (방 관리)



방 나가기

DB 참가자 제거

GOODBYE 이벤트 전송

JWT 토큰 삭제로 깔끔한 세션 정리



방 폭파

폭파 권한을 가진 사람이 방 폭파 가능

3초 카운트다운 후 소켓 종료 및 DB 삭제



편의 기능

파일 내용 원클릭 복사

실시간 닉네임 변경

Empty View 구현

| 6. 기술적 도전 (CLI 패키지)



```
~ codejam start --custom-env local
```

✓ Custom room created!

Room Code: 08WP19

Browser opened!

```
~ codejam enter T0B5C4
```

Connecting to room...

CodeJam CLI 패키지 구현

터미널 환경에서 브라우저를 거치지 않고 방을 생성하고 관리할 수 있는 개발자 친화적 도구입니다.

- > codejam start: 퀵 룸 생성 (6명)
- > codejam start --custom: 커스텀 룸 (150명)
- > codejam enter : 기존 룸 입장
- > codejam health: 서버 상태 확인

| 기술적 도전 (Redis 최적화)

압축 전략 (Compression)

업데이트 조각은 이중 연결 리스트로 관리합니다.

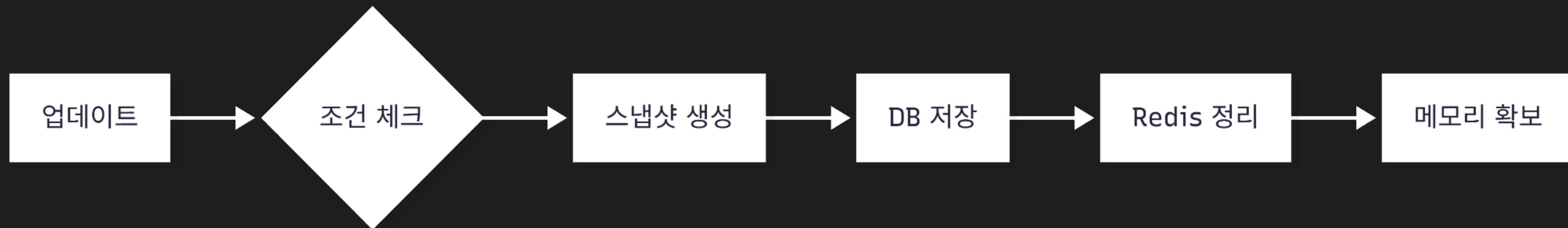
업데이트 조각이 500개 또는 64KB를 초과하면 압축을 진행합니다.

업데이트 조각을 병합한 스냅샷을 DB에 쓰고 Redis 리스트를 정리하여 메모리를 확보합니다. (Write-Back)

데이터 정합성 및 복구

Clock, Offset 모델을 도입하여 클라이언트와 서버, Redis 와 DB 간 데이터 정합성을 보장하고 연산을 최적화합니다.

Redis 영속성 설정을 통해 서버 장애 시에도 데이터 유실을 방지 합니다.



7. 향후 계획

이번 주 완료

- > 방 모드 이원화 (Quick/Custom)
- > 호스트 권한 관리 및 비밀번호 시스템
- > 인프라 환경 분리 및 Redis 메모리 최적화
- > Awareness (커서/아바타)
- > CodeJam CLI 패키지
- > 방 나가기/ 방 폭파 기능

남은 과제

- > 비밀번호 해시 함수 적용 (Security)
- > CLI npm 배포
- > 소켓 레이어 리팩터링 (중앙화)
- > 비밀번호, 닉네임 모달 우회할 때의 비정상 접근 처리
- > 한글 입력 씹힘 문제 해결
- > 코드 실행기능 구현

| 8. 데모 시나리오

- > 01. 홈 리뉴얼 디자인 확인
- > 02. CLI 명령어로 터미널에서 방 생성
- > 03. 실시간 동시 편집 & 멀티 커서 확인
- > 04. 파일 추가/삭제 및 설정 변경
- > 05. Custom Start 인원/비밀번호 설정
- > 06. 비밀번호 입장 및 호스트 권한 탈취

Q & A

피드백 요청 사항

- > 방 모드 분리가 직관적으로 이해되시나요?
- > 참가자 정렬 및 필터의 UI 는 직관적인가요?
- > Quick Start 6인 제한이 적절한가요?
- > 방 만들기 ui는 사용자 편의성이 괜찮나요?

감사합니다!