

# 88ERP 컴퓨터 역할 정의

작성일: 2025-07-02

작성자: Ann & Claude

프로젝트: 88ERP 시스템 구축

## 장비 구성 개요

총 4대 장비로 구성된 88ERP 통합 시스템

컴퓨터명	위치	OS	역할	IP/접속
server	회사	Windows + WSL2 Ubuntu 24.04	프로덕션 운영 서버	172.20.158.172
sp1	회사	Windows + WSL2 Ubuntu	개발 환경	172.30.194.247
ds916/ds920	회사	Synology NAS	데이터 저장소	//ds916, //ds920
mac	집	macOS	원격 모니터링	RustDesk 연결

## 상세 역할 정의

### 1. server (프로덕션 운영 서버)

#### 핵심 임무

- 88ERP 웹 대시보드 24/7 운영
- 외부 접속 처리 (도메인: db.88toy.co.kr)
- RustDesk 중앙 서버 (원격 제어 허브)
- API 서버 및 자동화 시스템 운영

#### 주요 소프트웨어

- Docker Desktop (컨테이너 운영)
- RustDesk Server (relay + rendezvous)
- Next.js (웹 대시보드)
- n8n (워크플로우 자동화)
- PM2 (프로세스 관리)
- Nginx (웹서버, 리버스 프록시)

#### 서비스 포트

- 80/443: HTTP/HTTPS 웹서버
- 3000: Next.js 개발서버

- 5678: n8n 워크플로우
- 21115-21117: RustDesk 서버

## 2. sp1 (개발 환경)

### 핵심 임무

- 코드 개발 및 테스트
- Git 버전 관리
- 프로토타입 제작
- server 배포 전 검증

### 주요 소프트웨어

- VSCode (코드 편집기)
- Claude + Claude Code (AI 개발 도구)
- Next.js (개발 서버)
- Git (버전 관리)
- MCP 서버들 (14개)

### 개발 환경

- 위치: `/home/sp1/88ERP/`
- Git 저장소: [https://github.com/kimsooim/88ERP\\_WSL](https://github.com/kimsooim/88ERP_WSL)
- 웹서버: <http://localhost:3000>

## 3. ds916/ds920 (데이터 저장소)

### 핵심 임무

- 이미지 및 파일 저장
- 시스템 백업 및 아카이브
- 교보문고 자동화 데이터 저장
- NAS 동기화 서비스

### 주요 기능

- Docker 컨테이너 운영
- Samba 파일 공유
- 자동 백업 스케줄
- 교보문고 매출 데이터 수집

## 접속 방식

- I: 드라이브 → //ds920/img (이미지)
- Y: 드라이브 → //ds920/2bot (교보문고)
- G: 드라이브 → Google Drive 동기화

## 4. mac (원격 모니터링)

### 핵심 임무

- 집에서 전체 시스템 모니터링
- 성능 분석 및 알림
- 원격 문제 해결
- 보고서 작성 및 관리

### 주요 소프트웨어

- RustDesk Client (원격 제어)
- Docker Desktop (컨테이너 모니터링)
- Git (코드 리뷰)
- 브라우저 (웹 대시보드 접속)

---

## 개발 워크플로우

### 일반적인 개발 흐름

1. sp1에서 코드 개발
- ↓
2. sp1에서 테스트 및 검증
- ↓
3. Git으로 코드 커밋/푸시
- ↓
4. server에서 코드 pull 및 배포
- ↓
5. 외부 접속 테스트 (db.88toy.co.kr)
- ↓
6. mac에서 모니터링 및 성능 확인
- ↓
7. NAS에 백업 저장

### 긴급 대응 흐름

문제 발생 알림 (mac)

↓

RustDesk로 server 원격 접속

↓

문제 진단 및 해결

↓

sp1에서 코드 수정 (필요시)

↓

server에 즉시 배포

↓

서비스 복구 확인

---

## 네트워크 구성

### 회사 내부망

- **server**: 172.20.158.172 (WSL2 Ubuntu)
- **sp1**: 172.30.194.247 (WSL2 Ubuntu)
- **NAS**: ds916/ds920 (고정 IP)
- **공통**: 같은 회사 네트워크 사용

### 외부 접속

- **도메인**: db.88toy.co.kr → server (설정 예정)
- **RustDesk**: server가 중앙 허브 역할
- **웹 대시보드**: HTTPS 보안 연결

### 포트 관리

- **80**: HTTP (일반 웹)
- **443**: HTTPS (보안 웹)
- **3000**: Next.js 개발서버
- **5678**: n8n 자동화
- **21115-21117**: RustDesk 서버
- **22**: SSH (필요시)

---

## 주요 프로젝트

### 88ERP 웹 대시보드

- **위치**: server에서 운영, sp1에서 개발

- 기술: Next.js + TypeScript + Tailwind CSS
- 데이터베이스: Supabase (클라우드)
- 인증: Supabase Auth

## 자동화 시스템

- 교보문고: 매일 매출 데이터 자동 수집
- Cafe24: 주문/상품 정보 동기화
- 에이랜드: 재고 관리 자동화
- 이메일 알림: 작업 완료 시 자동 발송

## 브랜드 관리

- 초바(Choba): 22개 캐릭터
- 코튼푸드(Cottonfood): 28개 캐릭터
- 코튼애니(Cottonani): 라이독, 리우, 에스티베어

---

## 시스템 사양

### server (운영 서버)

- CPU: Intel i5-1135G7 (8스레드)
- 메모리: 7.6GB (6.4GB 여유)
- 디스크: 954GB 여유
- OS: Windows + WSL2 Ubuntu 24.04 LTS

### sp1 (개발 컴퓨터)

- OS: Windows + WSL2 Ubuntu
- 역할: 개발 전용, 리소스 충분
- 네트워크: 회사 내부망

### NAS (저장소)

- 모델: Synology ds916, ds920
- 용량: 충분한 저장 공간
- 기능: Docker, 백업, 동기화

---

## 현재 상태 및 다음 계획

### 완료된 것

- server: RustDesk 서버 설치 완료
- sp1: 개발 환경 구축 완료
- NAS: 운영 중, 백업 시스템 가동
- 프로젝트: Git 저장소 관리 중

## 진행 중

- server: Docker 설치 예정
- 웹 대시보드: 외부 접속 설정
- RustDesk: 전체 PC 연결 테스트

## 다음 목표

1. server에 Docker + 웹서버 구축
2. 도메인 연결 (db.88toy.co.kr)
3. 전체 시스템 통합 테스트
4. mac 모니터링 환경 구축

---

## 최종 목표

"언제 어디서나 88ERP 시스템에 접근하여 업무 처리 가능한 완전 자동화 환경"

### 접속 방식

- 회사: sp1에서 개발, server에서 운영
- 집: mac에서 모니터링 및 원격 관리
- 외부: 웹 브라우저로 db.88toy.co.kr 접속
- 모바일: 반응형 웹으로 언제든지 접근

### 자동화 수준

- 데이터 수집: 완전 자동화
- 백업: 실시간 자동 동기화
- 모니터링: 24/7 무인 감시
- 알림: 문제 발생 시 즉시 통보

---

파일명: `computer-roles.md`

저장 위치: 각 컴퓨터의 프로젝트 루트 디렉토리

업데이트: 시스템 변경 시 즉시 반영