# 초거대제조AI 플랫폼 대시보드 화면 설계서

## 1. 문서 개요

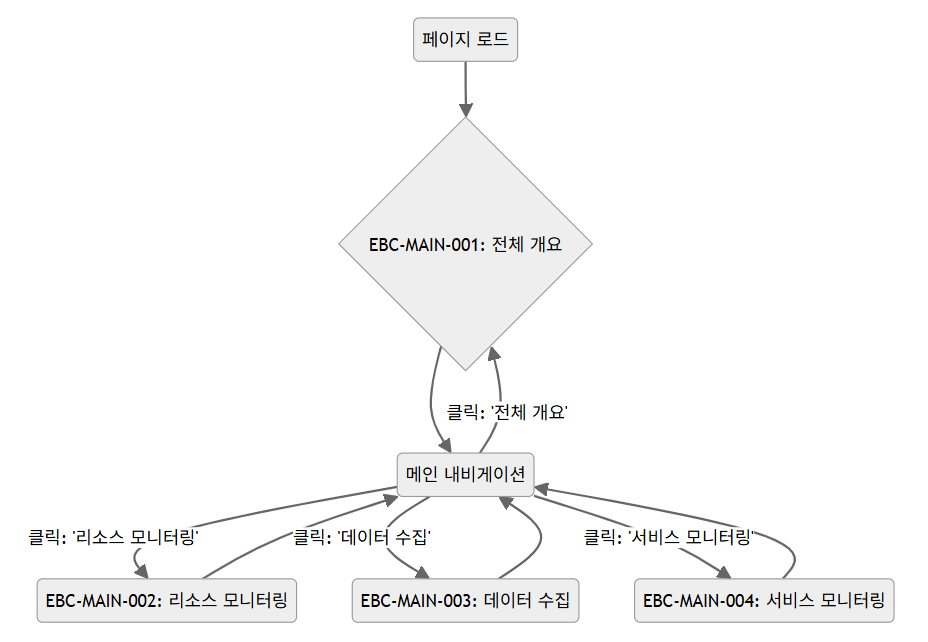
| 항목 | 내용 |
| --- | --- |
| 프로젝트명 | 초거대제조AI 플랫폼 대시보드 구축 |
| 문서 버전 | v1.0 |
| 작성일 | 2025-10-20 |
| 작성자 | 최형우 |
| 문서 목적 | EBC 대시보드 화면의 구성, 흐름(Flow), 기능, 인터랙션을 정의하여 기획/디자인/개발 간의 커뮤니케이션을 지원 |

## 2. 디자인 철학 및 비주얼 아이덴티티

| 항목 | 내용 |
| --- | --- |
| 전체 컨셉 | 하이테크(High-tech) 및 데이터 중심(Data-centric)의 전문적인 모니터링 대시보드 |
| 테마 | 다크 모드(Dark Mode)를 기본으로 하여 시각적 피로도를 줄이고, 데이터 및 상태 표시(초록, 빨강, 노랑)의 가시성을 극대화 |
| 폰트 | Noto Sans KR (웹폰트) |
| 주요 색상 팔레트 | 배경 (Primary): Slate-900 (#0f172a)  카드/컴포넌트 (Secondary): Slate-800 (#1e293b)  테두리: Slate-700 (#334155)  기본 텍스트: Slate-100 (#f1f5f9)  보조 텍스트: Slate-400 (#94a3b8)  강조 (Accent): Sky-400 (#38bdf8) - 활성 탭, 주요 타이틀  성공 (Success): Emerald-400 (#34d399) - 정상 상태, 긍정적 KPI  위험 (Danger): Red-400 (#f87171) - 경고, 부정적 KPI  경고 (Warning): Yellow-400 (#facc15) - 부하 증가  중립 (Neutral): Blue-400 (#60a5fa) - 중립적 KPI |

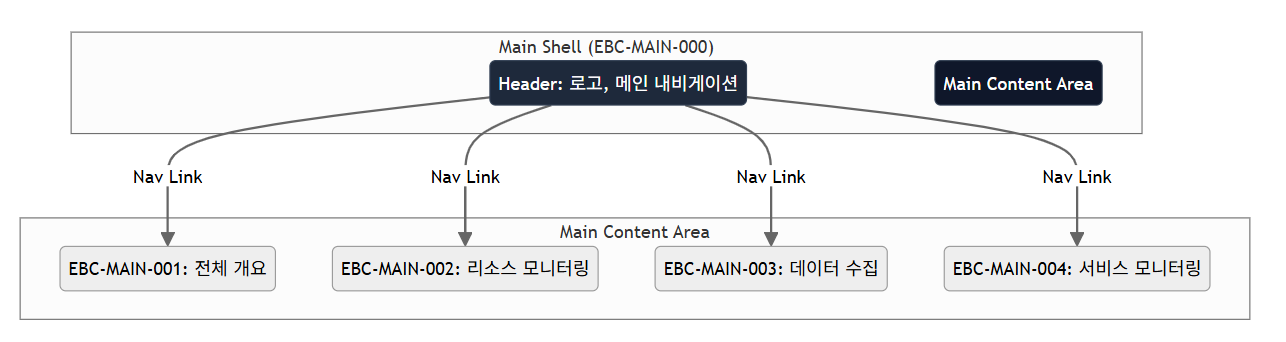
## 3. 플로우차트 (Flowchart)

### 3.1. 메인 내비게이션 플로우차트 (Main Navigation Flow)

사용자가 4개의 주 화면을 탐색하는 기본 플로우차트

## 4. 메뉴 구조 (Sitemap)

메인 셸(Shell) 구조와 4개의 주요 컨텐츠 섹션으로 구성.



* **메인 내비게이션 (#main-nav) 상세**
  + **1. 전체 개요** (Target: overview-dashboard)
  + **2. 리소스 모니터링** (Target: resource-dashboard)
  + **3. 데이터 수집** (Target: data-dashboard)
  + **4. 서비스 모니터링** (Target: service-dashboard)

## 5. 화면 목록 (Screen List)

| 대메뉴 | 중메뉴 | Screen ID | Page Title | Description | 비고 |
| --- | --- | --- | --- | --- | --- |
| 공통 | - | EBC-MAIN-000 | 메인 셸 | 모든 페이지에 공통으로 노출되는 헤더 및 내비게이션 영역. | 헤더, 공통 레이아웃 |
| 전체 개요 | - | EBC-MAIN-001 | 전체 개요 | 리소스, 데이터, 서비스의 핵심 요약과 제조 공정 디지털 트윈을 시각화하여 제공. | 기본 진입 화면 |
| 리소스 모니터링 | - | EBC-MAIN-002 | 리소스 모니터링 | EBC 센터의 서버, 스토리지, 네트워크 등 인프라 리소스 현황을 모니터링. |  |
| 데이터 수집 | - | EBC-MAIN-003 | 데이터 수집 | 데이터 파이프라인 흐름과 실시간 수집 현황, AAS 표준화 현황을 모니터링. |  |
| 서비스 모니터링 | - | EBC-MAIN-004 | 서비스 모니터링 | 각 수요 기업(KG 모빌리티, 신성델타테크)의 AI 서비스 적용 현황을 모니터링. |  |

## 6. UI 설계 (상세 화면 설계)

### 6.1. EBC-MAIN-000: 메인 셸 (공통)

* **화면 개요**: 모든 페이지에 공통으로 노출되는 헤더 및 내비게이션 영역.
* **UI 레이아웃**:  
  
* **컴포넌트별 기능 정의**

| No. | 컴포넌트명 | 데이터/표시 내용 | 기능 및 정책 (Description) |
| --- | --- | --- | --- |
| 1 | 로고/타이틀 | 아이콘, "초거대 제조 AI EBC 대시보드" | 클릭 시 기능 없음. (일반적으로 홈으로 이동) |
| 2 | 메인 내비게이션 | 4개 메뉴 (전체 개요, 리소스, 데이터, 서비스) | - 클릭 시 .nav-item에 active 클래스 토글.  - active 상태의 아이템은 하늘색 텍스트와 하단 테두리 표시.  - 클릭된 아이템의 data-target 속성에 해당하는 .content-section을 active (display: block) 상태로 변경하고 나머지는 숨김. |

### 6.2. EBC-MAIN-001: 전체 개요

* **화면 ID**: EBC-MAIN-001
* **Target ID**: #overview-dashboard
* **화면 개요**: 리소스, 데이터, 서비스의 핵심 요약과 제조 공정 디지털 트윈을 시각화하여 제공.
* **UI 레이아웃**: 2단 그리드 (좌측 2/3, 우측 1/3)  
  
* **컴포넌트별 기능 정의**

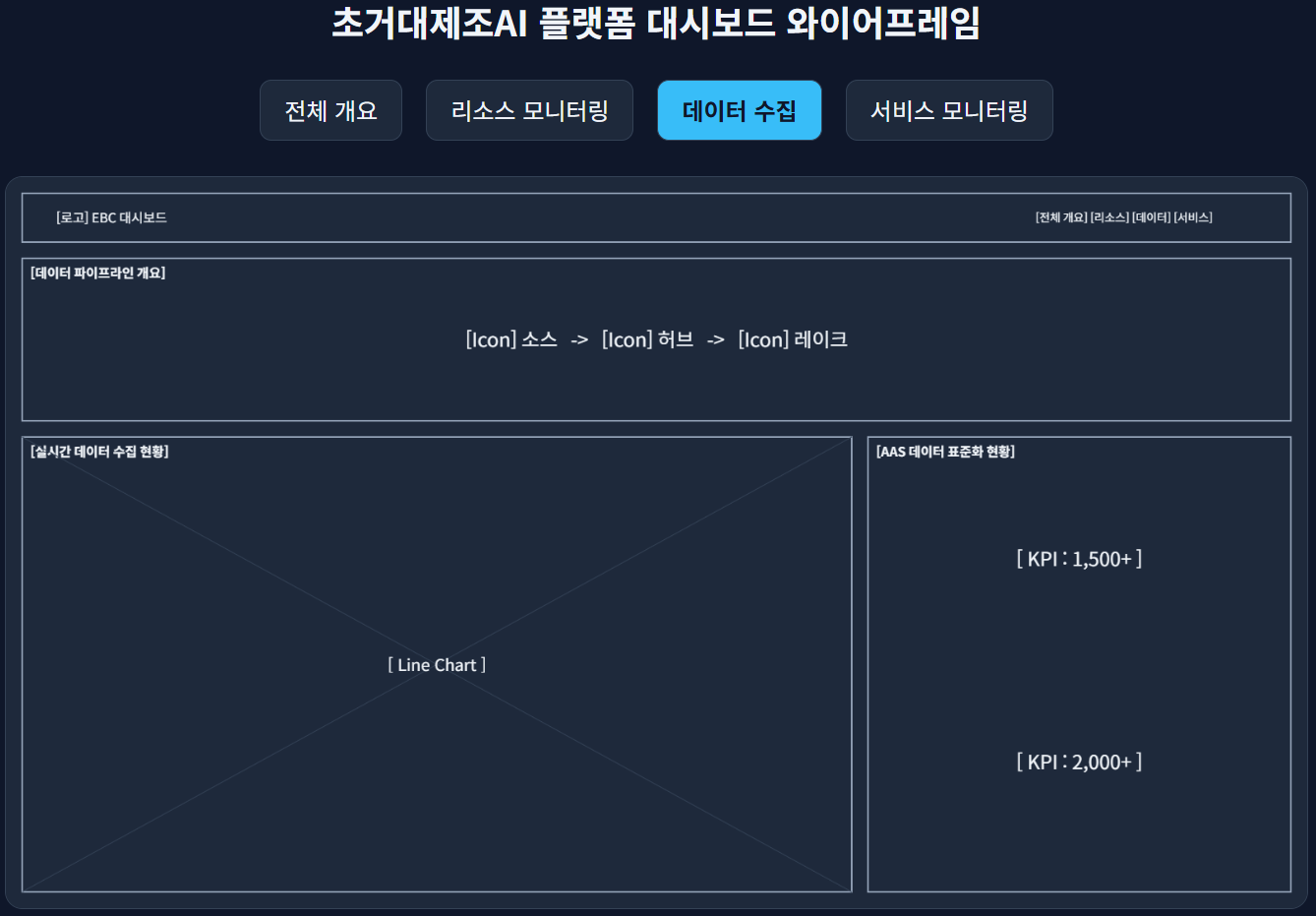
| No. | 컴포넌트명 | 데이터/표시 내용 | 기능 및 정책 (Description) |
| --- | --- | --- | --- |
| 1 | 요약 카드 (3종) | 리소스 요약, 데이터 요약, 서비스 요약 | - HTML에 하드코딩된 정적 텍스트 정보 표시.  - (서비스 요약) 불량률(-1.5%)은 .text-red-400, 효율(+8%)은 .text-emerald-400으로 상태 표시. |
| 2 | 디지털 트윈 | 3D 공장 설비 맵 (#digitalTwinCanvas) | - Three.js로 3D 씬(Scene) 렌더링.  - OrbitControls 라이브러리를 사용하여 마우스 드래그(회전), 휠(확대/축소) 인터랙션 제공.  - animate 함수가 requestAnimationFrame으로 실시간 실행.  - 설비(machine) 객체는 status ('warning', 'danger')에 따라 재질 색상이 하이라이트(점멸)됨. |
| 3 | 프로젝트 마일스톤 | 1~3단계 진행 현황 텍스트 | - HTML에 하드코딩된 정적 텍스트 정보 표시.  - '진행 중'은 text-sky-400, '완료'는 text-emerald-400으로 상태 표시. |
| 4 | 실시간 AI 제안 | 위험' 알림, '제안' 알림 | - HTML에 하드코딩된 정적 텍스트 정보 표시.  - '위험'은 .text-red-400, '제안'은 .text-sky-400으로 상태 표시. |

### 6.3. EBC-MAIN-002: 리소스 모니터링

* **화면 ID**: EBC-MAIN-002
* **Target ID**: #resource-dashboard
* **화면 개요**: EBC 센터의 서버, 스토리지, 네트워크 등 인프라 리소스 현황을 모니터링.
* **UI 레이아웃**: 4열 그리드 기반  
  
* **컴포넌트별 기능 정의**

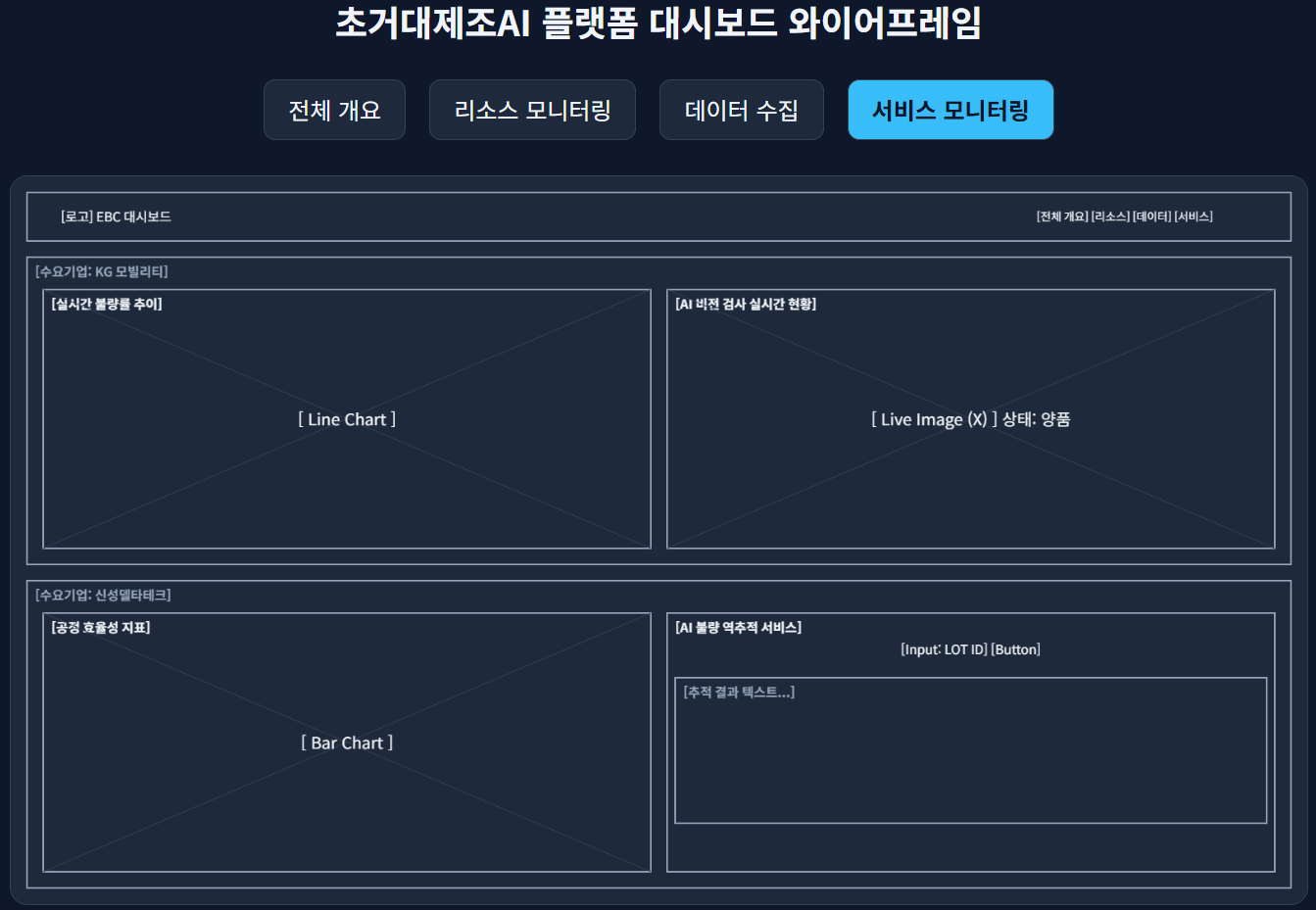
| No. | 컴포넌트명 | 데이터/표시 내용 | 기능 및 정책 (Description) |
| --- | --- | --- | --- |
| 1 | EBC 센터 서버 리소스 | GPU, CPU, 메모리 사용률 (Doughnut 차트 3개) | - Chart.js로 렌더링.  - createDoughnutChart 함수로 생성.  - 표기되는 데이터(78%, 65%, 82%)는 스크립트에 하드코딩됨. |
| 2 | 통합 스토리지 현황 | 데이터 레이크(75%), 온프레미스 백업(40%) | - div의 style="width: ...%" 속성으로 하드코딩된 값 표시. (프로그레스 바) |
| 3 | 시스템 상태 | 온프레미스 서버, 게이트웨이, DB, 보안 시스템 | - HTML 정적 텍스트.  - .status-dot 및 텍스트 색상(.text-emerald-400, .text-yellow-400)으로 상태 표시. |
| 4 | 네트워크 트래픽 | Inbound/Outbound (Line ㅜ구대트) | - Chart.js로 렌더링.  - createLineChart 함수로 생성.  - 10초간의 시뮬레이션 데이터(배열)는 스크립트에 하드코딩됨. |

### 6.4. EBC-MAIN-003: 데이터 수집

* **화면 ID**: EBC-MAIN-003
* **Target ID**: #data-dashboard
* **화면 개요**: 데이터 파이프라인 흐름과 실시간 수집 현황, AAS 표준화 현황을 모니터링.
* **UI 레이아웃**:  
  
* **컴포넌트별 기능 정의**

| No. | 컴포넌트명 | 데이터/표시 내용 | 기능 및 정책 (Description) |
| --- | --- | --- | --- |
| 1 | 데이터 파이프라인 개요 | 데이터 소스 -> 데이터 허브 -> 데이터 레이크 | - Lucide 아이콘과 HTML 정적 텍스트로 구성된 플로우맵. |
| 2 | 실시간 데이터 수집 현황 | 초당 수집 데이터 (Line 차트) | - Chart.js로 렌더링.- 10초간의 시뮬레이션 데이터(배열)는 스크립트에 하드코딩됨. |
| 3 | AAS 데이터 표준화 현황 | AAS 모델 수, 연결된 디지털 자산 (KPI) | - .kpi-value 클래스를 사용한 정적 텍스트. |

### 6.5. EBC-MAIN-004: 서비스 모니터링

* **화면 ID**: EBC-MAIN-004
* **Target ID**: #service-dashboard
* **화면 개요**: 각 수요 기업(KG 모빌리티, 신성델타테크)의 AI 서비스 적용 현황을 모니터링.
* **UI 레이아웃**: 2개의 수직 분할 섹션.  
  [수요기업: KG 모빌리티]  
  [수요기업: 신성델타테크]  
  
* **컴포넌트별 기능 정의 (KG 모빌리티)**

| No. | 컴포넌트명 | 데이터/표시 내용 | 기능 및 정책 (Description) |
| --- | --- | --- | --- |
| 1 | 실시간 불량률 추이 | 엔진 조립라인 불량률 (Line 차트) | - Chart.js로 렌더링.- 주간 시뮬레이션 데이터(배열)는 스크립트에 하드코딩됨. |
| 2 | AI 비전 검사 | 실시간 카메라 피드(이미지), '양품'/'불량' 상태 태그 | - setInterval (3초)을 통해 images 배열(총 4개, 1개는 불량) 순환.- #kgmVisionImage의 src 속성 변경.- #kgmVisionStatus의 텍스트(양품/불량) 및 배경색 클래스(bg-emerald-500/bg-red-500) 변경. |

* **컴포넌트별 기능 정의 (신성델타테크)**

| No. | 컴포넌트명 | 데이터/표시 내용 | 기능 및 정책 (Description) |
| --- | --- | --- | --- |
| 3 | 공정 효율성 지표 | 개선율 (Bar 차트) | - Chart.js로 렌더링.- 3개 지표(비가동 시간, 생산량, 불량률) 데이터는 스크립트에 하드코딩됨. |
| 4 | AI 불량 역추적 서비스 | LOT 번호 입력, 추적 버튼, 결과 표시 영역 | - #sdtTraceBtn 클릭 시 이벤트 리스너 실행.- (유효성 검사) #sdtLotInput 값이 비어있으면 유효성 검사 섹션 참조.- 비어있지 않으면 #sdtTraceResult에 "추적 중..." 표시.- setTimeout (1초) 후, 하드코딩된 HTML 결과 문자열을 #sdtTraceResult에 삽입. |

## 7. 공통 정책 및 유효성 검사 (Validation)

본 대시보드는 대부분 정보 조회를 목적으로 하므로, 유효성 검사는 최소화됨.

| 화면 | 컴포넌트 | 상황 (Condition) | 처리 시나리오 (Action) |
| --- | --- | --- | --- |
| EBC-MAIN-004 | AI 불량 역추적 | 추적' 버튼 클릭 시 #sdtLotInput 값이 null 또는 "" (빈 값) | #sdtTraceResult 영역에 <p class="text-red-400">LOT 번호를 입력해주세요.</p> 메시지를 표시. |

## 8. 향후 작업 (Future Tasks)

본 문서는 현재 시뮬레이션 데이터(하드코딩) 기반의 프론트엔드 프로토타입 설계를 다루고 있음. 3차년도 실제 운영을 위해 다음 작업이 추가적으로 필요함.

### 8.1. 사용자 인증 시스템 (회원가입/로그인)

* **목표**: EBC 플랫폼 접근을 위한 보안 및 사용자 식별.
* **필요 화면**:
  + **EBC-AUTH-001: 로그인**: 이메일/패스워드 입력, 소셜 로그인.
  + **EBC-AUTH-002: 회원가입**: 사용자 정보 입력, 소속(수요 기업) 선택.
  + **EBC-AUTH-003: 비밀번호 찾기**: 이메일 인증을 통한 비밀번호 재설정.
* **정책**: 로그인 성공 시 서버로부터 JWT(Json Web Token) 또는 세션 토큰을 발급받아 이후 API 요청 헤더에 포함.

### 8.2. 권한 관리 시스템 (RBAC: Role-Based Access Control)

* **목표**: 사용자 역할(예: 관리자, 수요 기업 담당자, EBC 운영자)에 따라 접근 가능한 메뉴와 데이터를 차등 적용.
* **필요 기능**:
  + **관리자**: 모든 메뉴 접근, 사용자 관리, 권한 설정.
  + **수요 기업 담당자 (예: KGM)**: 공통 대시보드 및 자신에게 할당된 '서비스 모니터링' (KGM) 페이지만 접근 가능.
  + **EBC 운영자**: '리소스 모니터링', '데이터 수집' 등 플랫폼 운영 메뉴 접근 가능.
* **설계 변경**: 메인 내비게이션(#main-nav)은 로그인한 사용자의 권한에 따라 동적으로 생성되어야 함.

### 8.3. 실제 시스템 연동 (Live Data Integration)

* **목표**: 현재 하드코딩된 시뮬레이션 데이터를 실제 백엔드 시스템과 연동하여 실시간 데이터로 대체.
* **작업**:
  + **API 정의**: 대시보드에 필요한 모든 데이터를 조회하기 위한 RESTful API 또는 WebSocket 엔드포인트 정의. (예: GET /api/v1/resources/summary, GET /api/v1/services/kgm/defect-rate)
  + **데이터 연동**:
    - fetch 또는 axios를 사용하여 API 호출.
    - Chart.js 차트 생성 로직을 API 응답(Response)을 받아 데이터를 주입하도록 수정.
    - setInterval 또는 WebSocket을 사용하여 10초~1분 단위로 데이터 자동 갱신(Auto-refresh) 기능 구현.
  + **시뮬레이션 로직 제거**: setTimeout (AI 불량 역추적), setInterval (AI 비전 검사) 등 시뮬레이션용 스크립트 제거.

### 8.4. 서비스 확장 (Service Expansion)

* **목표**: 신규 수요 기업 및 신규 AI 서비스 추가에 유연하게 대응할 수 있는 모듈식 구조로 변경.
* **작업**:
  + **컴포넌트 모듈화**: EBC-MAIN-004: 서비스 모니터링 섹션을 동적으로 로드할 수 있도록 설계.
  + **신규 수요 기업 추가**: (예: '현대모비스') 신규 수요 기업을 위한 서비스 모니터링 화면(EBC-MAIN-005) 설계 및 추가.
  + **신규 서비스 추가**: (예: '예지 보전', '에너지 최적화') 신규 서비스 모듈을 대시보드에 통합.