


MCP로 AI와 디자인을 연결하는 차세대 개발

Model Context Protocol 완벽 이해와 실습

강사명: 조성호

소요시간: 60분

 bg right:40% w:400

9교시 학습 목표

- MCP(Model Context Protocol) 핵심 원리 이해
- Figma 디자인을 AI와 연동하는 실습
- 디자인 토큰 자동화 및 컴포넌트 코드 생성
- 디자인-개발 완전 통합 워크플로우 구현

기존 디자인-개발 문제점

- 디자인과 코드 불일치
- AI 도구가 디자인을 읽지 못함
- 반복적인 수작업
- 디자인 변경 시 개발팀 부담

MCP가 바꾸는 미래

Before

디자인 → 수작업 변환 → 코드 작성 → AI와 별개


After

디자인 ↔ MCP 서버 ↔ AI ↔ 자동 코드 생성

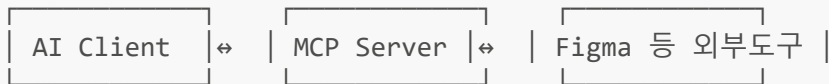
MCP = AI, 디자인, 개발을 실시간으로 연결하는 핵심 기술

MCP 핵심 개념

- **AI-도구 통신을 표준화**
- 실시간 디자인 데이터 공유
- Figma, GitHub, Notion 등 확장 가능
- 디자인 시스템을 코드로 자동 변환

 bg right:35% w:350

MCP 아키텍처



- **AI Client:** Claude, GPT 등
- **MCP Server:** 중간 통합 서버
- **External Tool:** Figma, GitHub 등

실습 Step 1: 환경 준비

```
node --version # v18 이상
npm install -g @anthropic-ai/mcp-cli
```

프로젝트 초기화

```
mkdir mcp-workspace && cd mcp-workspace
touch claude_desktop_config.json
touch .env
```

Figma Access Token 생성

1. figma.com 로그인 → Settings → Token 생성
2. `.env` 파일에 저장

Step 2: MCP Figma 연동

```
npm install @mcp/server-figma
npx @mcp/server-figma --version
```

설정파일 예시

```
{
  "mcpServers": {
    "figma": {
      "command": "npx",
      "args": ["@mcp/server-figma"],
      "env": {
        "FIGMA_ACCESS_TOKEN": "figd_..."
      }
    }
  }
}
```

Step 3: Claude Desktop 연동

1. OS별 설정파일 위치 확인
2. `claude_desktop_config.json` 복사
3. Claude 재시작 후 MCP 연결 확인

정상 연결 확인

```
[INFO] Figma MCP Server started
[INFO] Listening for MCP connections...
```

Step 4: 디자인 데이터 활용

Figma 파일 구조 분석

```
https://www.figma.com/file/ABC123/My-Design
```

Claude에게 요청:

```
"Figma 파일 ABC123의 컴포넌트 목록과 스타일을 분석해줘"
```

디자인 정보 → AI가 바로 읽고 코드화

Step 5: 디자인 토큰 자동 생성

컬러 팔레트 CSS 변수 변환

```
:root {  
  --primary-blue: #3498db;  
  --text-primary: #2c3e50;  
}
```

타이포그래피 시스템

```
.heading-1 {  
  font-size: 48px;  
  font-weight: 700;  
}
```

간격 시스템 (8px 그리드)

```
--spacing-sm: 8px;  
--spacing-md: 16px;
```

Step 6: 컴포넌트 코드 자동화

React 예시

```
interface ButtonProps {  
  size: "small" | "large";  
  disabled?: boolean;  
}  
  
export const PrimaryButton = ({ size, disabled }: ButtonProps) => (  
  <button className={`btn ${size}`} disabled={disabled}>  
    Click me  
  </button>  
);
```

Vue, HTML/CSS 코드도 자동 생성 가능

Step 7: AI 디자인 분석

접근성 검사

- 색상 대비
- 터치 타겟 크기
- 키보드 내비게이션

일관성 검증

- 색상, 폰트, 간격
- 컴포넌트 재사용률

성능 최적화

- 이미지 최적화
- CSS 구조 개선

실시간 워크플로우 구축

자동화 명령어

```
npm run design:sync      # Figma 동기화
npm run design:tokens    # 토큰 추출
npm run design:components # 컴포넌트 생성
npm run design:analyze   # 접근성/성능 분석
```

VSCode 플러그인으로 실시간 업데이트

도전과제: 완전한 MCP 시스템 구축

- ☒ MCP 서버-클라이언트 연동 ☒ Figma 디자인 AI 연결 ☒ 디자인 토큰 CSS 자동화 ☒ 컴포넌트 코드 생성
☒ 실시간 변경사항 감지

보너스: 접근성 분석, 성능 최적화, CI/CD 연계

수업 정리

- MCP 개념과 필요성 이해
- Figma 실시간 AI 연동 실습
- 디자인-개발 통합 자동화 구축
- 미래 지향적 워크플로우 완성

다음 시간: 정적 웹사이트 미니 프로젝트 시작!
