

W02  
기술 스택과 서비스  
구조

어떤 칼과 냄비로 만들  
것인가

# 이제 서비스를 개발할 수 있을까?

서비스 개요

- 이 서비스는 어떤 목적으로 만들어 졌는가?
- 이 서비스는 무엇으로 목적을 달성하는가?

PRD

- 이 서비스를 만들기 위해 나는 무엇을 결정했는가?
- 이 서비스를 만들기 위해 AI는 무엇을 결정했는가?

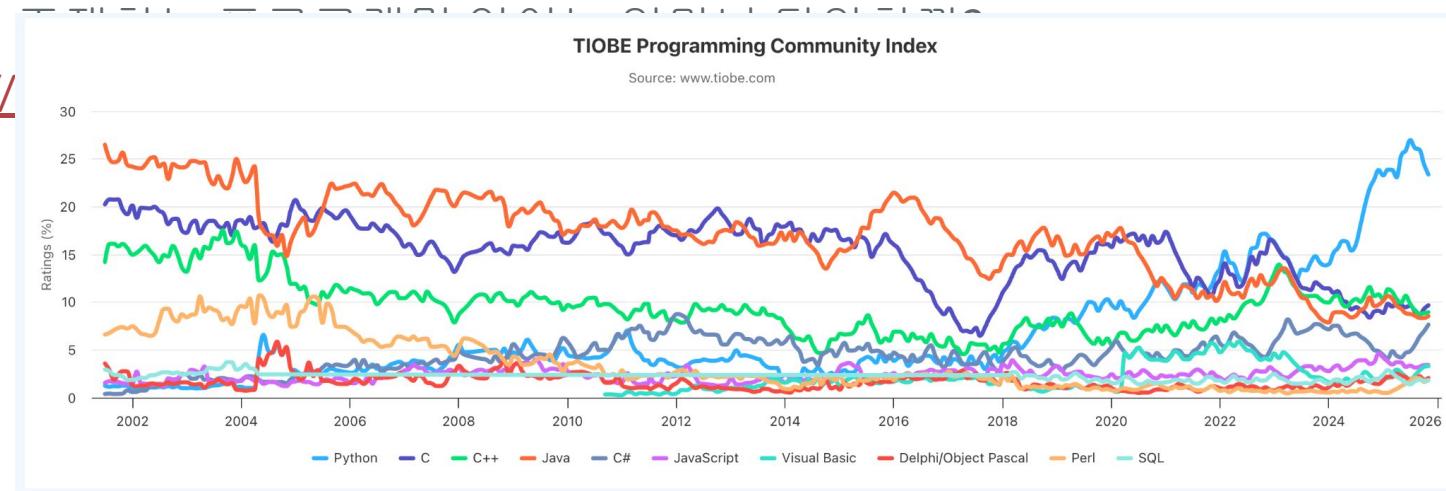
“그 다음 필요한 것은?”

# 프로그래밍 언어 - 나와 IT는 어떻게 연결될 것인가?

- 프로그래밍 언어: 인간의 언어를 컴퓨터가 이해할 수 있도록 만들어진 소통 도구
- 컴퓨터가 내가 원하는 대로 동작하게 하기 위해서는 '적당한' 언어를 선택해야 한다.

세상에

<https://>



# 프레임워크 - 미리 만들어 놓은 검증된 시스템

프로그래밍 언어를 알면 모든 것을 만들 수 있을까?

- 너무 복잡해진 현대 IT 시스템
- 유사한 목적을 가졌으면 비슷하게 만들어도 되지 않을까?

## 프레임워크

- 어떠한 목적을 달성하기 위해 복잡하게 얹혀있는 문제를 해결하기 위한 미리 잘 만들어진 구조
- 여러 개발자들이 함께 시스템을 개선하며 목적 달성을 더 좋은 구조를 발전시켜

\* 서울의 도시 실험, 2017 서울도시건축비엔날레! \*

**I-SEOUL.U**      **서울특별시**

서울시 홈페이지  
  
[seoul.go.kr](http://seoul.go.kr)

---

수신자 수신자참조  
(경유)

제목 여성안심특별시 정책 홍보 협조

1. 서울시와 tvN이 함께하는 여성폭력예방 공동 캠페인 추진 관련입니다.  
2. 여성안심특별시 정책 추진과 관련하여 폭력범죄 해결 드라마인 "크리미널 마인드"를 활용한 홍보물을 아래와 같이 제작 송부하오니 여성안심보안관 및 안심귀가 스카우트 등 캠페인 진행시 활용하여주시기 바랍니다.

가. 홍보 물 : 부제  
나. 홍보내용 : 여성안심귀가스카우트, 안심이 앱, 여성안심지킴이집, 여성 안심택배, 데이트폭력 상담 전용콜, 여성안심보안관

다. 수 량 : 400EA/자치구  
라. 협조사항 : 하절기 거리 캠페인 등 배포요청  
※ 여성안심보안관이 홍보를 배포시 홍보실적 인정

불 임 : 홍보물 1부, 광.

서울특별시장

수신자 자치구 여성정책부서

주무관	이용철	여성정책기획팀장 고광현	여성정책담당관 08/08 박현숙
협조자	주무관	권진영	

시행 여성정책담당관-15247 ( ) 접수 ( )  
우 04524 서울특별시 종로구 세종대로 110 (태평로1가) /  
전화 02)2133-5016 /전송 02)2133-0729 /\*\*\*\*\*@seoul.go.kr /대시민공개

# 데이터베이스 - 데이터를 어떻게 저장할 것인가?

- IT = Information Technology : IT에서 제일 중요한 것은 '정보'
- 내가 다루는 정보의 특징에 맞게 어떻게 저장할 것인가를 중요하게 생각해야 한다.

정보를 다루는 방식에 따라 저장이 달라진다

- 숫자, 표 중심의 수치 데이터
- 일반 자연어 문서 데이터
- 엑셀과 같은 스프레드시트 데이터
- 이미지, 동영상, 음악과 같은 미디어 데이터
- 채팅, 댓글 등의 규칙 없는 데이터
- 등등등

# 컴퓨팅 시스템 - 어떤 컴퓨터에서 동작하게 되는가

이 모든 것들이 돌아가는 컴퓨터는 어디에 있는가

- 내 컴퓨터에서만 돌아가는 시스템? 그저 계산과 실행을 위해서만 사용할 수 있다
- 인터넷과 연결되어 다양한 사람들이 정보를 주고 받을 수 있어야 한다.

IT 서비스에서 쓰이는 두 종류의 컴퓨팅 시스템

- 클라이언트: 실제 사용자가 사용하는 컴퓨터(노트북, 스마트폰, 스마트 패드, IoT 등)
- 서버: 데이터가 계산되고 저장되는 컴퓨터(보통 보이지 않는 곳에 있다)

"컴퓨팅 시스템은 클라이언트와 서버가 서로 다른 기술 스택을 가져야 한다"

# 그 외에 어떤 것들이 있을까?

-

# 기술 스택은 한번 정하면 바꾸기가 힘들다

- 한번 영어로 쓰여진 책을 다시 한국어로 쓰는 것은 거의 불가능하다
- 프로그래밍 언어끼리는 서로 전혀 대화를 나누기가 어렵다.
- 언어별로 문법 구조가 달라 기존에 작성된 코드를 다른 언어로 전환하는 것은 불가능에 가깝다.

“처음부터 최선의 기술 스택을 정하기 위한 노력을 해야 한다”

# 질문 1. 기술 스택을 정하기 위한 사전 질문

“이제 실제 개발에 적용할 기술 스택을 정해야 해. 시스템 개요와 PRD를 살펴보고 목적 달성에 필요한 최선의 기술 스택을 결정하기 위해 너가 추가로 알아야 할 것들이 무엇인지에 대해 질문 해줘”

- 기술 스택을 이해하는데 필요한 의사결정 요소들을 이해한다.
- 내가 어떤 수준의 서비스를 만들고자 하는지에 대해 명확하게 의사결정한다.

“이 모든것들이 명확히 결정되면 최선의 기술 스택을 결정할 준비가 된다”

## 질문 2. 기술 스택 초안 제안 받기

“앞서 나는 대화를 바탕으로, 이 서비스를 개발하는데 어떤 기술 스택이 적절할지에 대해 깊게 고민하고, 최선의 결정을 제안해줘.”

- 하나의 제안이 아닌 여러개의 제안을 받아 본다.
- 각 제안들의 장단점, 서비스에 주는 영향들에 대해 깊이있게 파악한다.
- 각 스택별로 서로 어떻게 연결될 수 있으며 이것이 어떤 시너지를 내는지도 알아본다.

“스택을 확인하면서 이해가지 않는 내용이나 모르는 부분은 반드시 AI에게 물어보며 이해하고 넘어간다”

# 질문 3. 기술 스택 로드맵 구성하기

"이 모든 스택을 적용하는 것이 부담이 될 수 있어. 만약 단계적으로 적용하는 전략을 세운다면 어떤 순서로 어떻게 적용하며 어떤 방식으로 확장해 나가는 것이 최선인지에 대해 앞서 작성한 문서들을 모두 참고해서 제안해줘"

- 단계적 적용이 가능하다: 초기에는 서버 없이 로컬에서 구현 -> SQLite -> Postgre로 전환
- 쉬운 것에서 어려운 것으로 전환: 스크립트 기반 배포 -> Docker 배포 -> Teraform 전환
- 확장에 따라 프레임워크 전환을 검토: 간단한 웹앱으로 -> Flutter 기반 -> 네이티브 앱

# 좋은 결정을 위한 팁

초기에는 무료 서비스들을 적극 활용할 수 있다

- 웹 클라이언트 컴퓨팅: Vercel 등
- 서버 컴퓨팅: Supabase 등

유명한 스택들은 비교적 안전할 수 있다

- 웹 개발: React, Vue.js 등
- 클라우드 컴퓨팅: AWS, Azure 등

특정 목적을 위한 프레임워크를 찾아본다

- 대시보드 전문 프레임워크: Grafana

# 서비스의 전체 구조를 이해하자

다양한 기술들이 우리 서비스에서 어떻게 동작하는가?

- 전체 기술 스택들과 시스템의 동작을 이해하지 못하면 관리가 불가능하다
- 내가 관리할 수 없는 시스템은 절대 제대로 동작하게 관리할 수 없다

어떤 구조를 알아야 하는가?

- 시스템 구조: 실제 서비스가 어떤 방식으로 동작하는가에 대한 구조
- 유저 경험 구조: 사용자가 어떻게 서비스에 들어와 기능을 사용하고 움직이는 구조
- 데이터 흐름 구조: 데이터가 어디에서 들어와 어떻게 나가는지에 대한 구조

# 시스템 구조

- 이 서비스가 동작하는 전체 시스템의 구조
- 컴퓨터, 저장소, 통신, 인터페이스 등 다양한 요소들이 어떻게 연결되어있느냐에 대한 이해
  - 스마트폰 앱에서 사용자 입력 발생
  - 입력된 데이터가 AWS EC2 서비스로 전달
  - Webhook으로 알림이 생성되면 이미지 컴퓨팅 서버로 정보가 전달
  - 처리된 데이터가 코어 알고리즘으로 해석되어 결과 생성
  - 결과 데이터가 텍스트 형태로 동작하면 스마트폰에서 자동 PDF 생성
- 이 구조를 이해하지 못하면 서비스가 동작하지 않아도 왜 그런지에 대해 전혀 이해할 수 없다

# 유저 흐름 구조

- 이 서비스를 사용하는 사용자가 충분한 목적을 달성할 수 있는가?
- 단순한 순서도가 아닌 사용자가 어떤 경험을 통해 페이지를 이동하는지에 대해 파악
  - 로그인 화면 - 로그인을 한다(X) 소셜 로그인으로 원클릭 가입
  - 홈 화면 - 주요 정보 제공(X) 오늘 할일 리스트, 목표 달성 D-day 표시
  - 설정 화면 - 설정 정보 제공(X) 프로필 꾸미기 메뉴, 다크 모드 활성화
- 사용자가 각 화면에서 어떤 목적을 달성해 다음으로 넘어가고 이 경험들이 어떻게 유기적으로 연결되는지 파악하고 탐색한다
- 문제가 있는 부분들은 하나씩 검토하는 것도 좋지만 전반적인 문제를 전달하고 개선책을 제안하는 것도 가능
  - “사용자는 보통 사진을 10개 이상을 한번에 등록해야 하는데 사진 등록 할 때마다 매번 편집을 요구하는건 이 서비스 목적에 맞지 않아”

# 데이터 흐름 구조

- 서비스 내에서 데이터가 어떻게 이동하고 처리되는지에 대한 구조
- 데이터 흐름을 이해하면 서비스의 어떤 부분에서 데이터 처리의 문제가 발생할 수 있는지 파악이 가능하다

내가 다루는 데이터의 종류과 형식을 이해할 수 있어야?

- “데이터들은 어떤 포맷으로 생성되고 이동하는지에 대해서 알려줘”
- “그 데이터 포맷들이 이 서비스에서 동작하기에 최선의 선택인 이유에 대해 설명해줘”
- “이 형식이 전송되고 처리되는데 어떤 특징과 장점이 있는지에 대해 설명해줘”

# 질문 1. 시스템 구조 결정하기

"앞서 결정한 서비스 개요, PRD, 기술 스택에 맞게 이 시스템이 목적을 달성하도록 하기 위해 어떤 시스템 구조를 갖추는 것이 필요한지 내용을 검토하고 최선의 구조를 제안해줘"

- 이 구조가 최선인 이유에 대해 물어본다.
- 이 구조가 만들어내는 시너지가 무엇인지 확인한다.
- 이 구조가 가지는 위험성에 대해 물어본다.
- 이 구조에서 구현이 불가능한 추후 추가될 가능성이 있는 기능의 범위가 무엇인지 확인한다

## 질문 2. 유저 흐름 구조 결정하기

"앞서 결정한 내용들을 모두 참고해 이 서비스를 실제로 사용하는 사용자가 제일 효과적으로 목적을 달성할 수 있도록 유저 흐름 구조를 설계해줘. 각 흐름에서 고객이 어떤 목적으로 이 페이지에 도착하고 어떤 경험을 해서 어떤 목표를 달성할 수 있는지가 서비스 개요와 목적에 명확하게 부합해야 해"

- 제안해준 사용자 흐름을 실제로 상상해보고, 이것이 최선인지에 대해 검토한다.
- 필요하다면 흐름만 경험할 수 있는 Mockup 페이지를 제작해 달라고 요청한다.
- 구체적인 UI까지 지금 결정하기보다는 사용자 경험이 적절한지를 중심으로 검토한다.

## 질문 3. 데이터 흐름 구조 점검하기

“앞서 작성한 시스템구조, 유저 흐름 구조를 바탕으로 전체 서비스에서 데이터가 어떻게 다뤄지는지에 대해 정리해줘.”

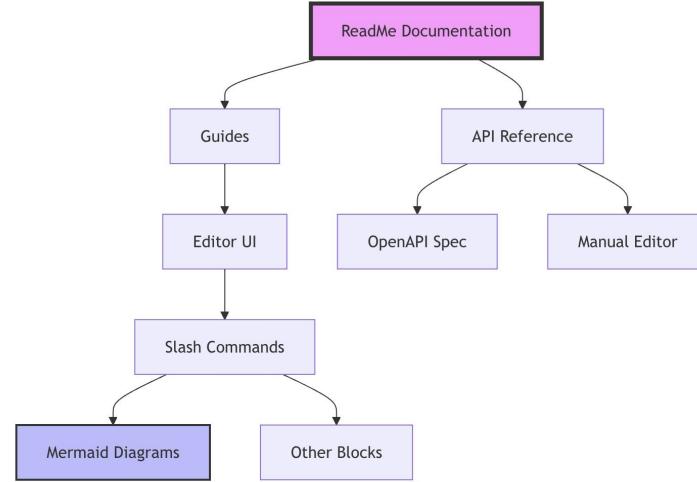
- 여러분이 우리 시스템의 데이터가 전체 시스템 안에서 어떻게 다뤄지는지에 대해 파악하는 것이 주요 목적

추가: “그리고 데이터 흐름이 우리의 목적에 맞지 않거나 더 개선할 수 있는 부분이 있다면 검토하고 이를 기반으로 시스템 전체의 구조 개선을 제안해줘”

- 데이터가 어떻게 움직이는지에 따라 전체 시스템이 변화할 가능성에 대해 점검

# “Mermaid로 표현해줘”

- Chart 형태를 표현하는데 최적화된 스크립트
- 이 형식으로 만들면 코드로 차트를 파악하는데 편리하다
- 노션에서 바로 차트를 보는 것도 가능



# 반드시 검토하고 개선점을 찾아 발견한다

- AI가 정리한 내용이 반드시 정답이라고 생각하면 안된다
  - 실제로 개발해보면 많이 뒤통수 맞게 된다
  - 개발 스택 수준에서 갈아 엎은 경험이 수십번이 넘는다
- 내가 많이 알면 알수록 AI는 정확하게 일한다.
- 스스로 공부하고 이해하고, 동의하는 부분만 확정짓고 나머지는 계속 피드백으로 개선한다.

"이 과정이 탄탄히 완성되어야 이후 만들게 될 로드맵이 정확하게 작성된다"

# 오늘 해결해야 할 문제

- 서비스 개요, PRD를 기반으로 기술 스택을 어떻게 선택하는 것이 좋은지에 대해 파악하고 각 기술 스택의 장단점과 시스템의 구현 목적, 확장성 등을 고려해 최선의 기술 스택을 선정한다.
- 이를 기반으로 시스템 구조, 유저 흐름 구조, 데이터 흐름 구조를 이해하고 이 서비스가 전반적으로 어떻게 동작하는지에 대해 파악한다.
- (선택) 결정된 기술 스택과 구조 정보들을 바탕으로 이 서비스의 기술 문서 페이지를 노션에 작성해 본다.

# Q&A