

- 액티비티 (레이어)
- 레이아웃 (영역) = 화면의 구조 (화면이 어떻게 구상돼 있느냐)
- 뷰 = 구성요소 (화면에 보이는 버튼, 텍스트) (이미지 xml)

백엔드를 하기위한 서버 프레임워크(물리적 친구 접근) 중 하나 : Django(python), Plask(python으로 가능), (c++)

서버 :

HW 서버, 라즈베리파이(3~4만원) - 서버, 랜선, 공유기 ip 외부 접속 포트포워딩

DB(standard) : 구매 필요 / posting,url이 필요

웹 호스팅시 DB 사용 가능 : AWS(서버+DB), Oracle(DB),

호스트 서버 : naver.com

API : naver.com/medicin_list (web을 사자!)

신나게 디자인 - 레이아웃/뷰 프론트 -> 1차

1. 서버(라즈베리파이 / 웹(cafe24))를 구축할 수 있는 환경을 구매
2. 서버 구축(Find how to 구축) - 프레임워크(장고, 플라스크)
3. 프론트와 백을 연결해주는 친구 - (socket통신, http를 이용한 API를 백엔드가 만듦)를 프론트에게 url로 연결, 프론트는 이 url로 http 통신을 함. 제대로 통신되면 화면에 뿌려줌(통신 안될 시 서버 문제:404, 프론트문제:error)
 1. jango api 만들기 plask api 만들기
 2. Django url 만들고(devide) api 구축
 3. 어떤식으로 프론트에게 데이터를 넘기는가?
 4. DB(약학정보원, AI) 연결은 코드를 짜라 (plask api구축 시 AWS 쿼리문을, select를 날려서 가져옴)
 5. plask같은 친구로 API 짤 때 Algorithm도 그 위에 올림

- naver cloud DB제공