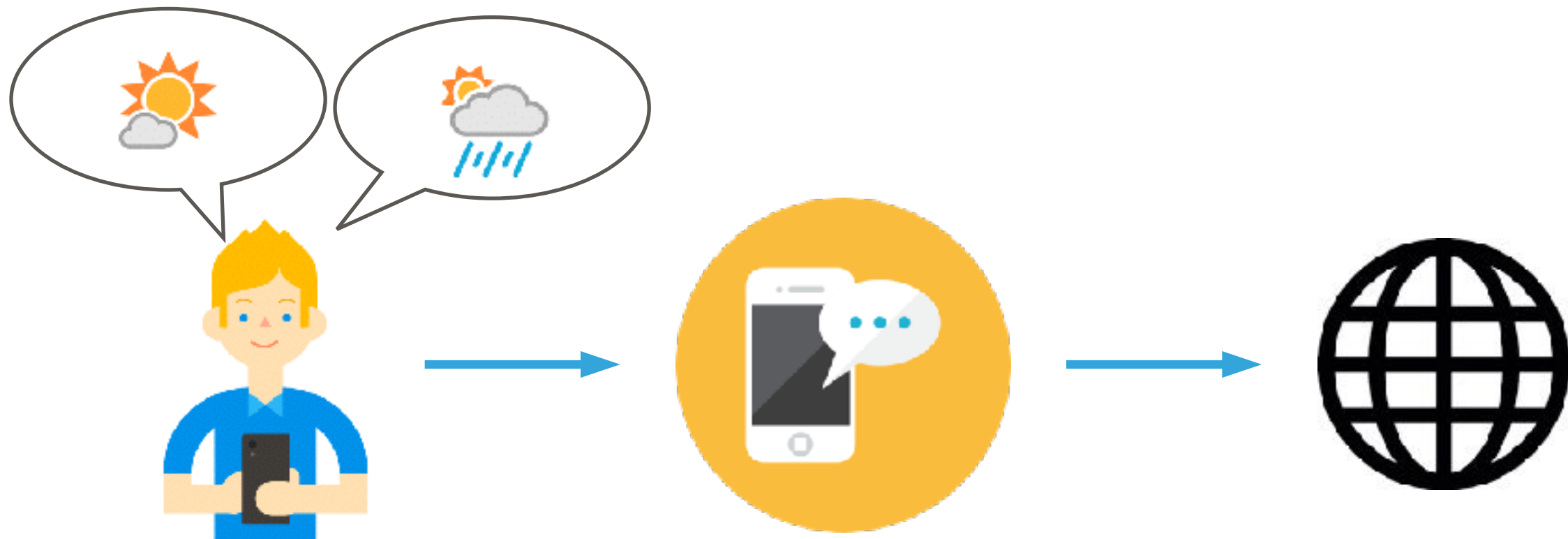

THE WEATHER BOT

SUMMARY

- ▶ 1. The App - The Weather Bot
- ▶ 2. Architecture
- ▶ 3. Next Steps

1. THE APP – THE WEATHER BOT

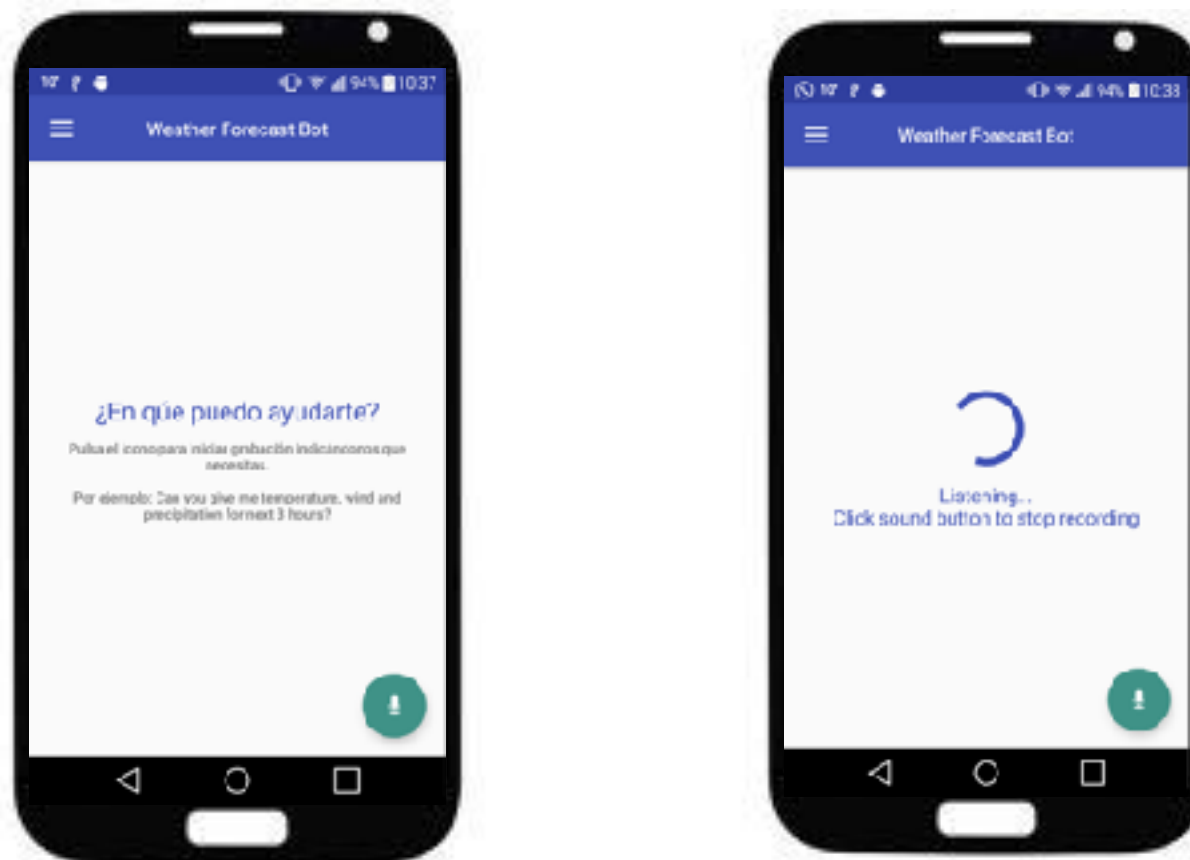


1. THE APP – THE WEATHER BOT



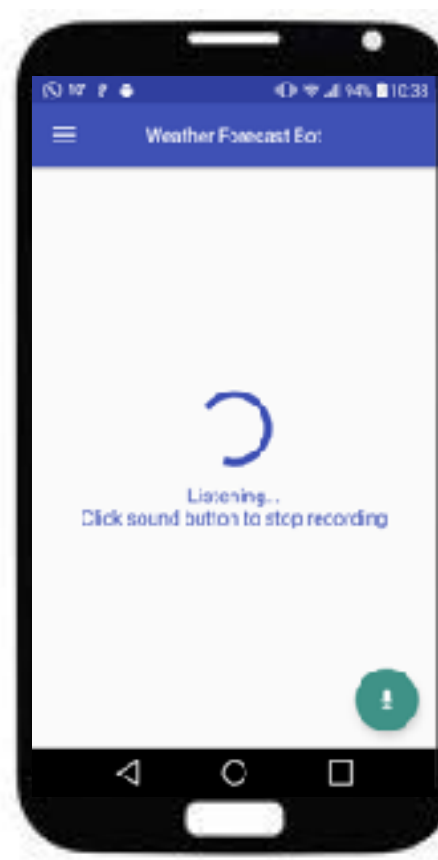
1) User asks his/her phone about the weather

1. THE APP – THE WEATHER BOT



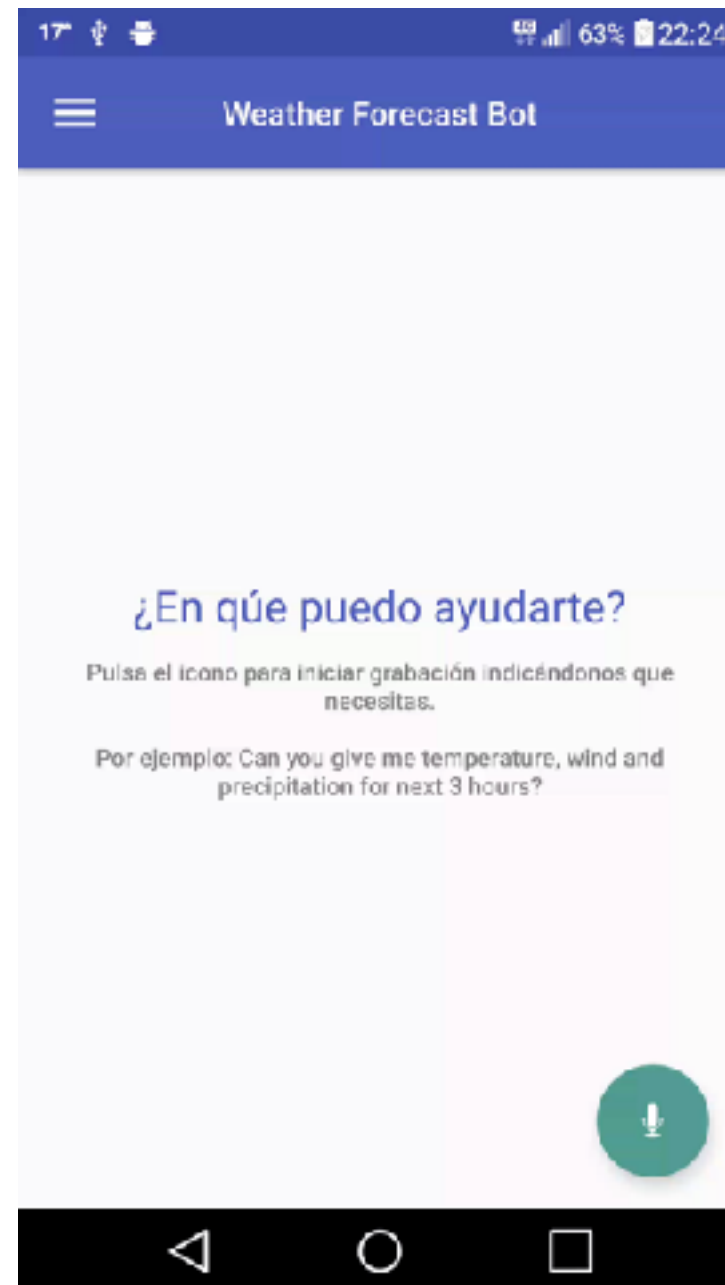
- 1) User asks his/her phone about the weather
- 2) Analyze the user question - WIT.AI, GNLP

1. THE APP – THE WEATHER BOT



- 1) User asks his/her phone about the weather
- 2) Analyze the user question - WIT.AI, GNLP
- 3) Show forecast and speak the info

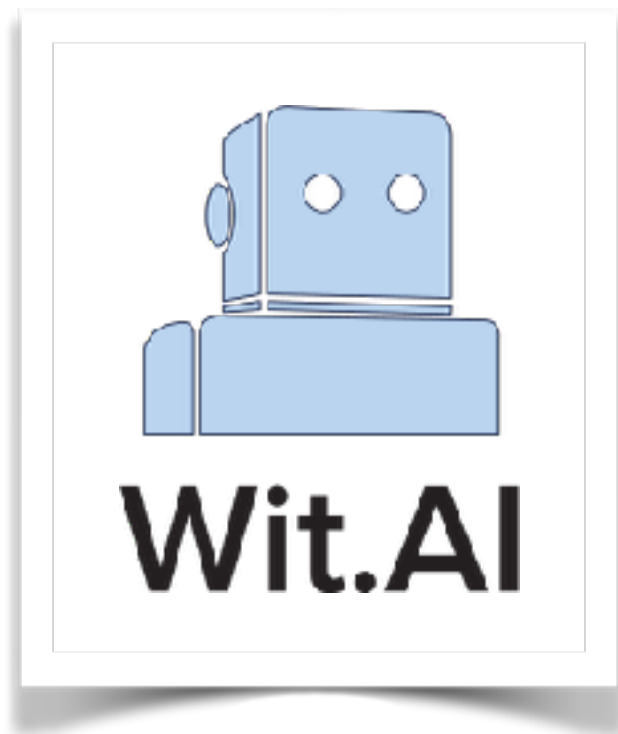
1. THE APP – THE WEATHER BOT



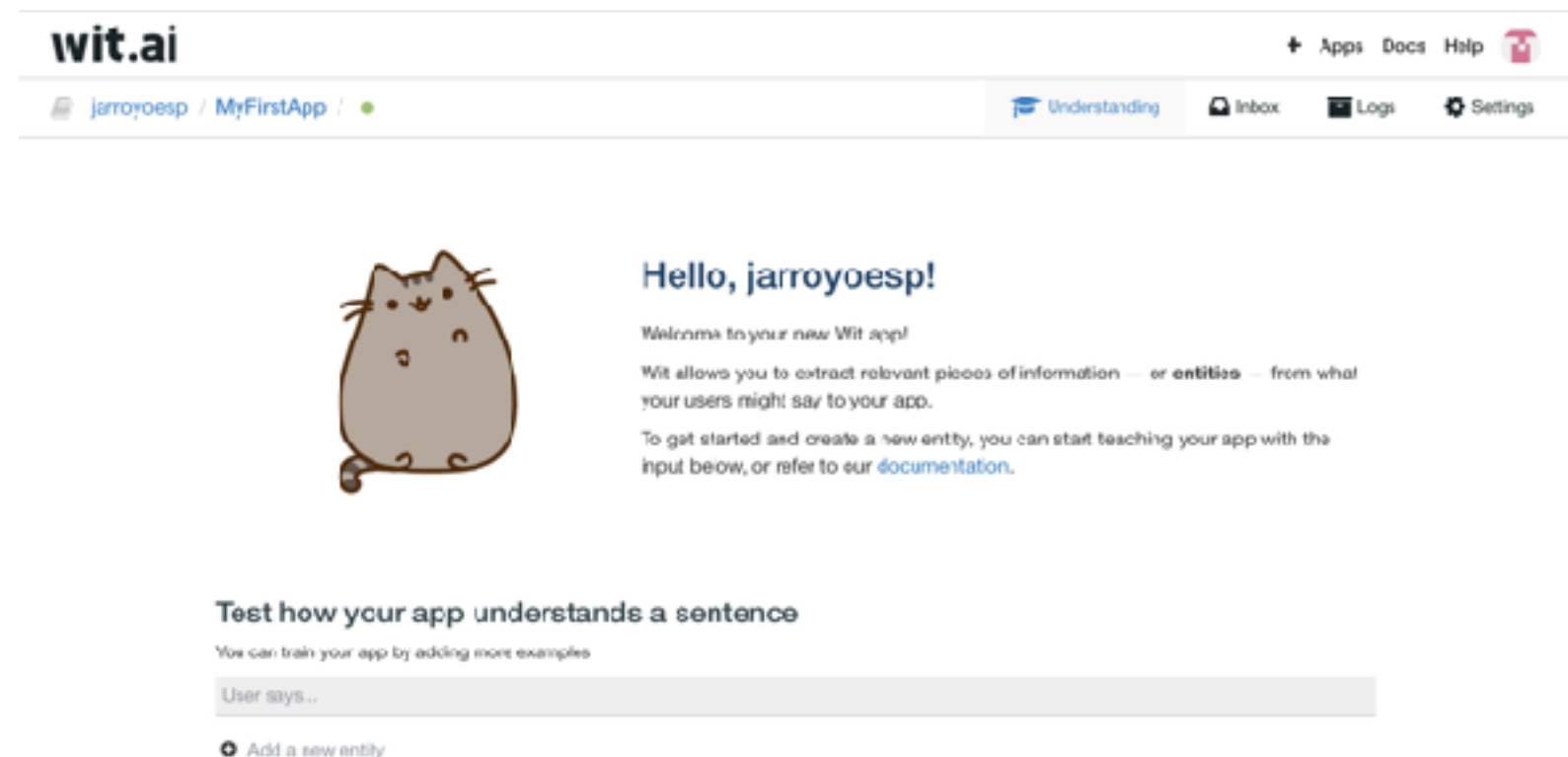
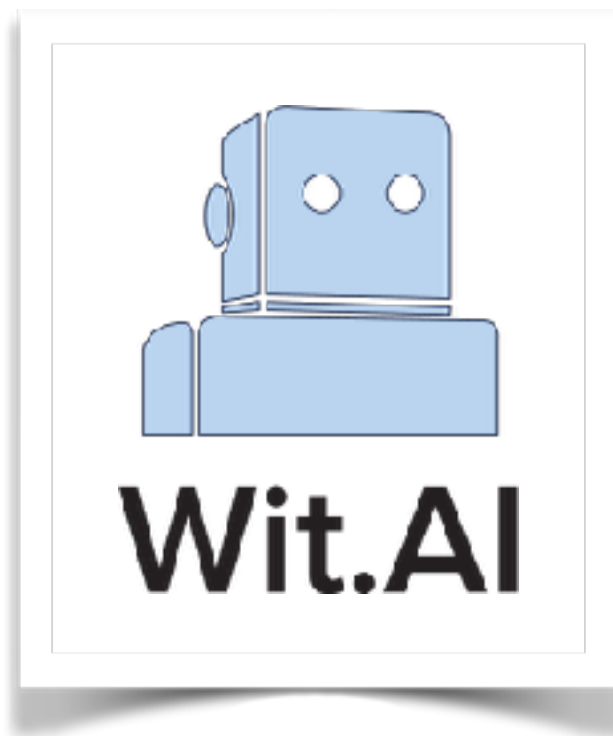
2. THE ARCHITECTURE

- ▶ Divide and Rule
 - ▶ 1) Speech Recognition
 - ▶ 2) API Weather
 - ▶ 3) Application

2.1 SPEECH RECOGNITION

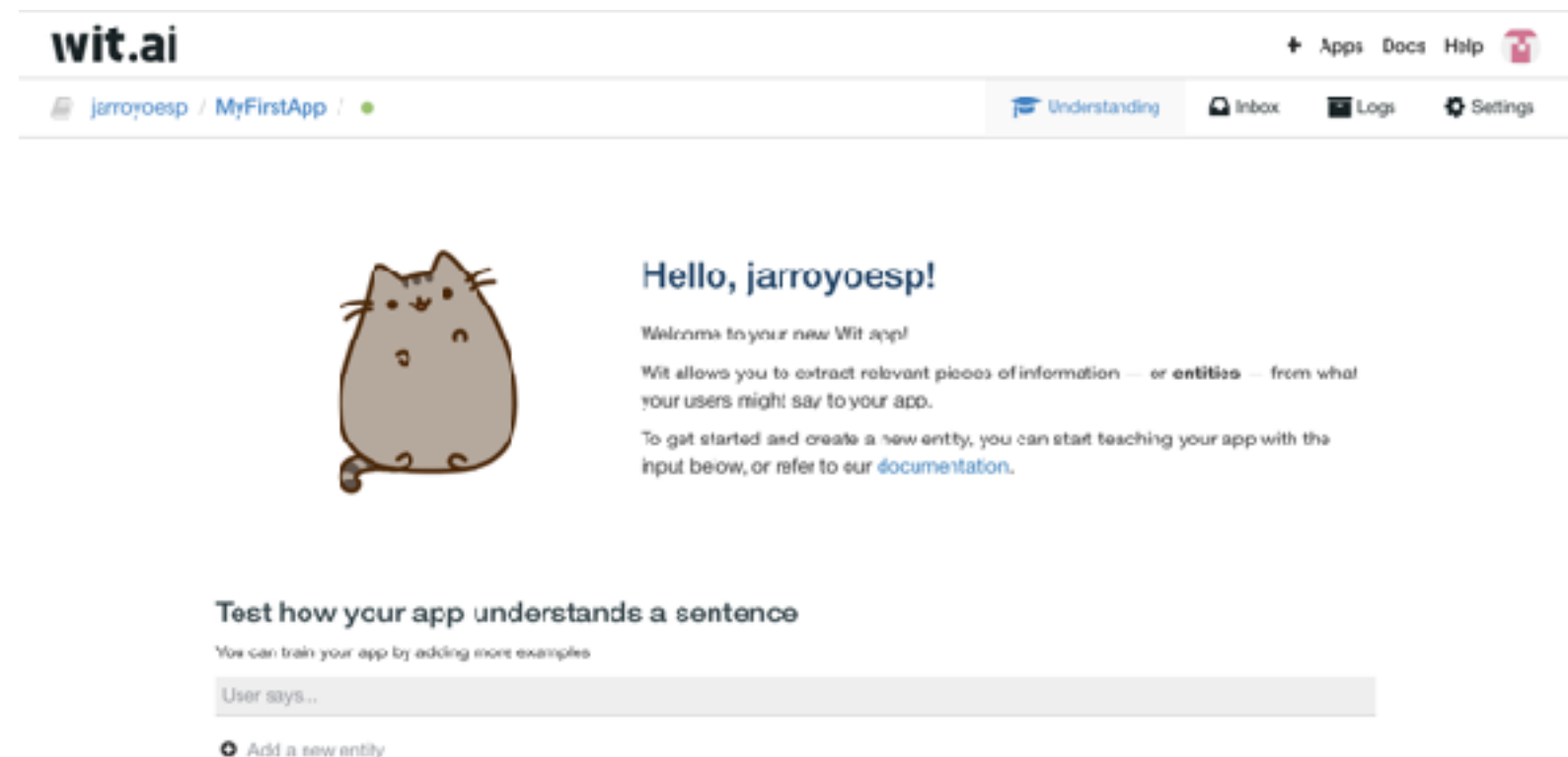
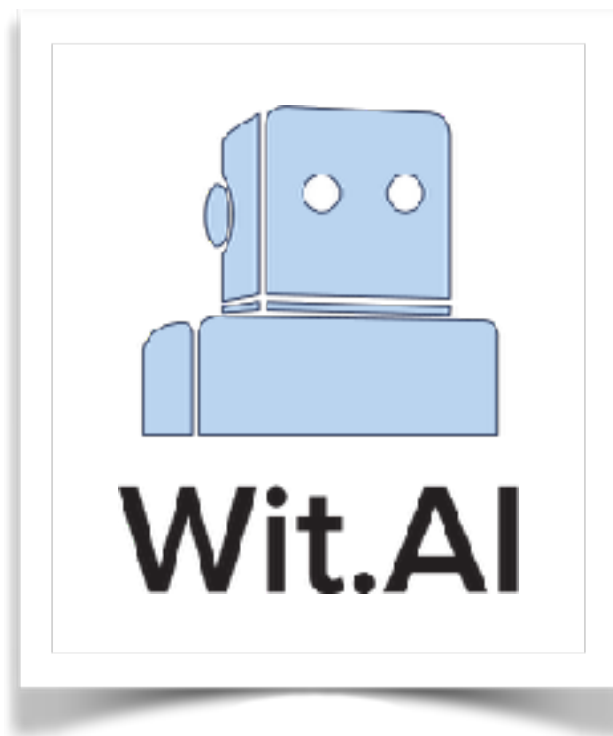


2.1 SPEECH RECOGNITION



1) Wit.AI Free

2.1 SPEECH RECOGNITION



INTEGRATION PROBLEM! Compilation fails Android ≥ 6

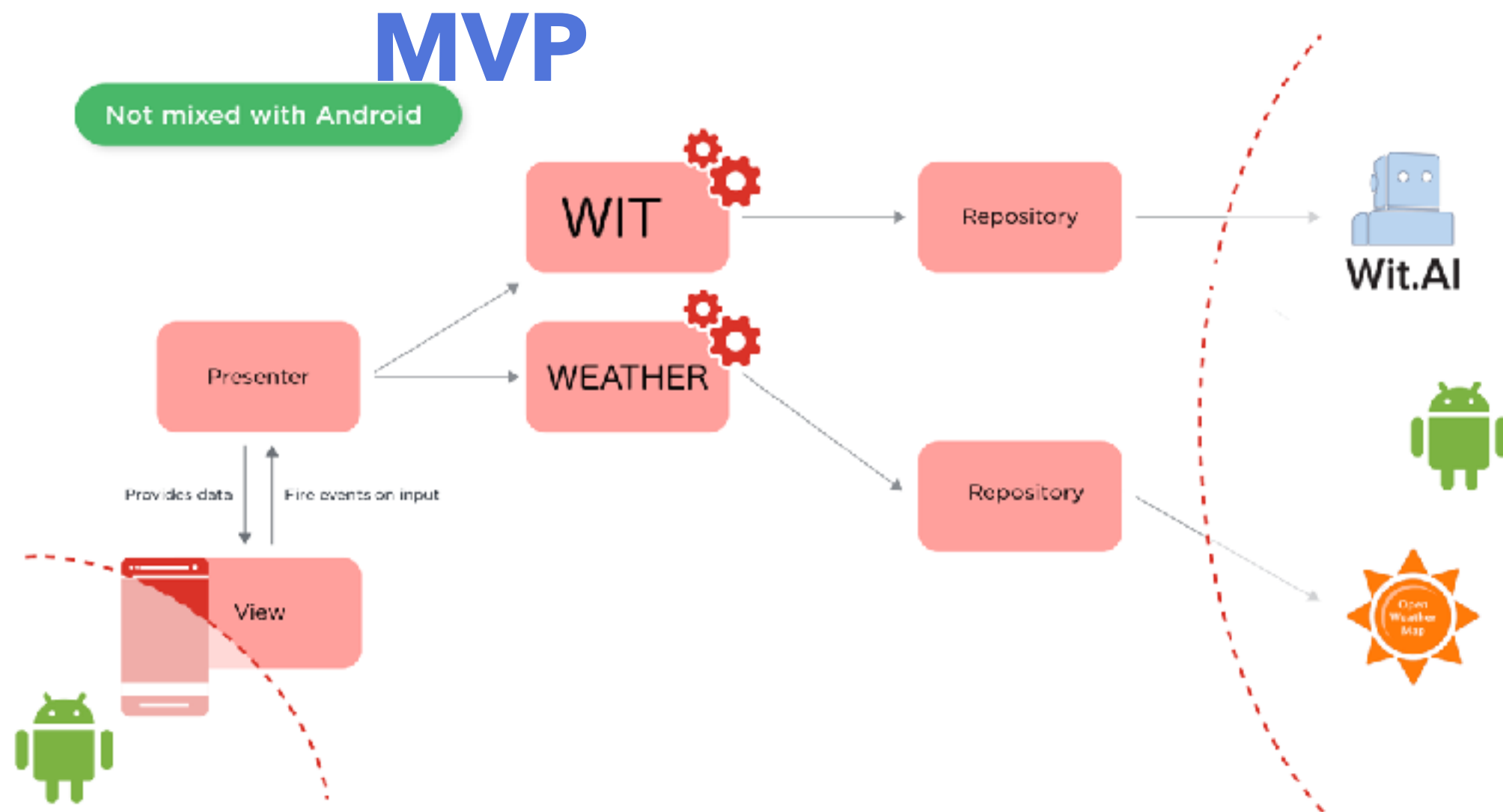
Simulate responds OK

2.2 API OPEN WEATHER MAP



```
{
  "city": {
    "id": 1851632,
    "name": "Zaragoza",
    "coord": {
      "lon": 138.933334,
      "lat": 34.966671
    },
    "country": "JP",
    "cod": "200",
    "message": 0.0045,
    "cnt": 38,
    "list": [
      {
        "dt": 1406106000,
        "main": {
          "temp": 298.77,
          "temp_min": 298.77,
          "temp_max": 298.774,
          "pressure": 1005.93,
          "sea_level": 1018.18,
          "grnd_level": 1005.93,
          "humidity": 87,
          "temp_kf": 0.26
        },
        "weather": [
          {
            "id": 804,
            "main": "Clouds",
            "description": "overcast"
          }
        ],
        "clouds": {
          "all": 88
        },
        "wind": {
          "speed": 5.71,
          "deg": 229.501
        },
        "sys": {
          "pod": "d"
        },
        "dt_txt": "2014-07-23 09:00:00"
      }
    ]
  }
}
```

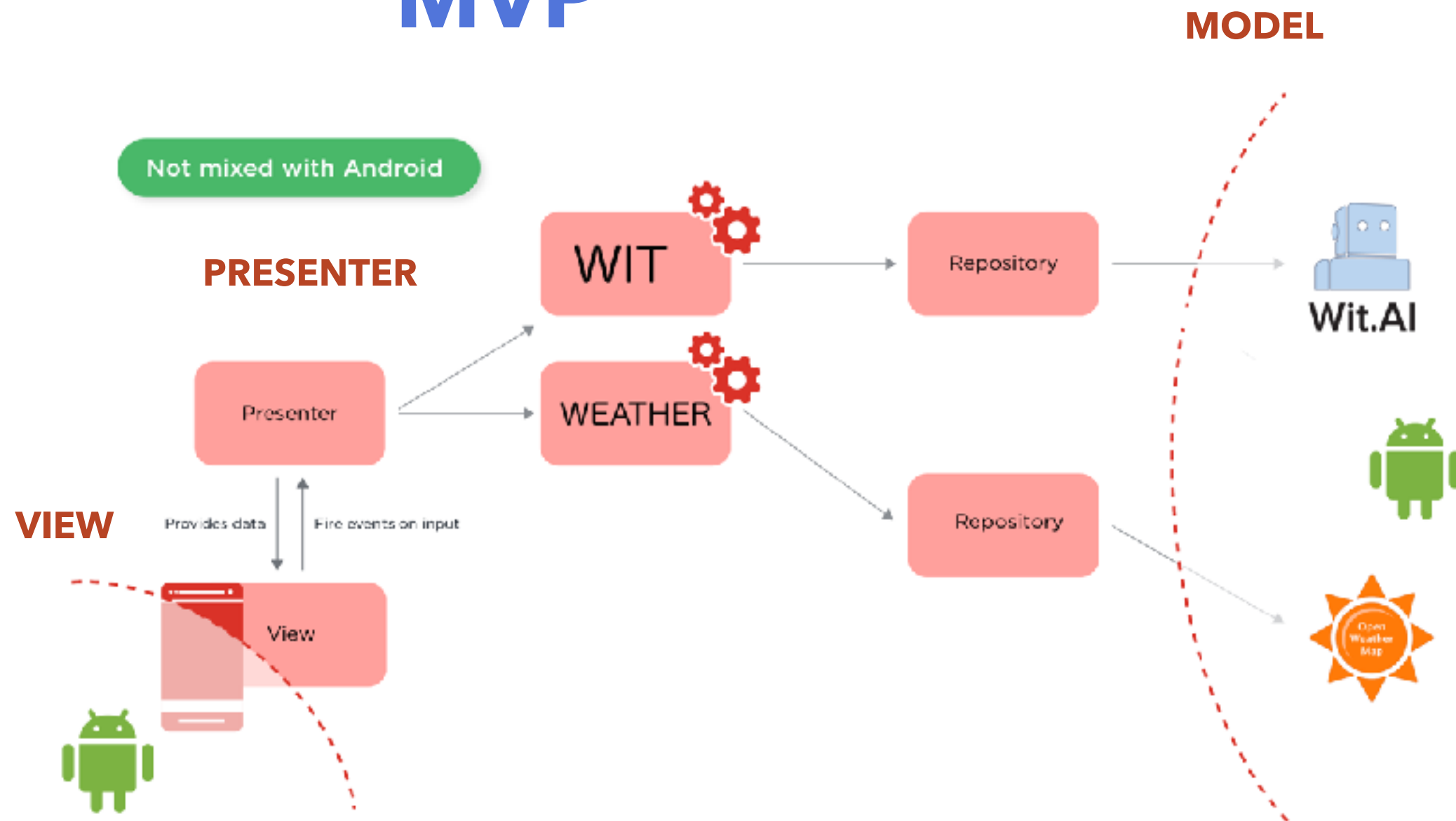
2.3 APPLICATION – CLEAN ARCHITECTURE



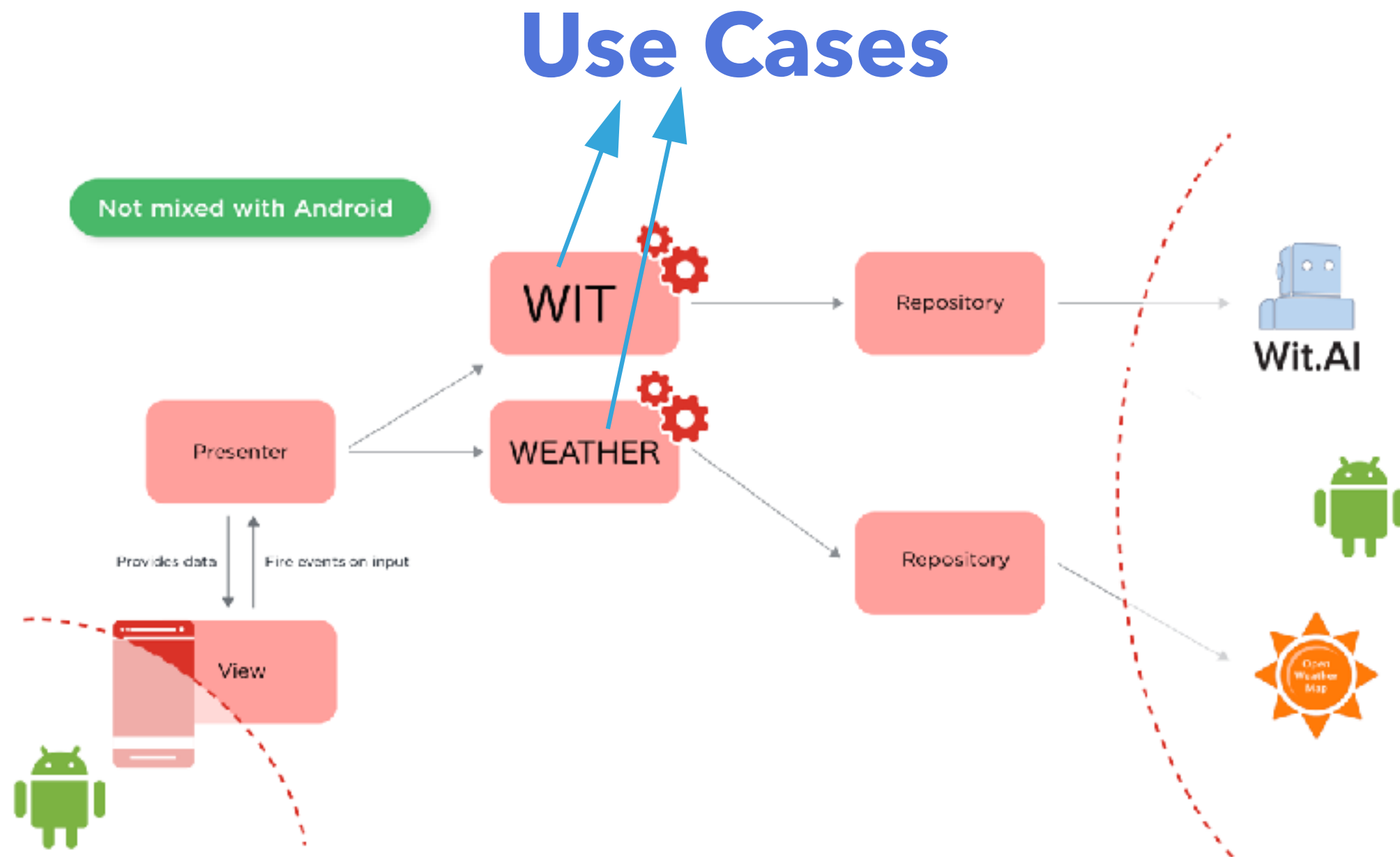
- 1) Scalable
- 2) Maintainable
- 3) Testable

2.3 APPLICATION – CLEAN ARCHITECTURE

MVP

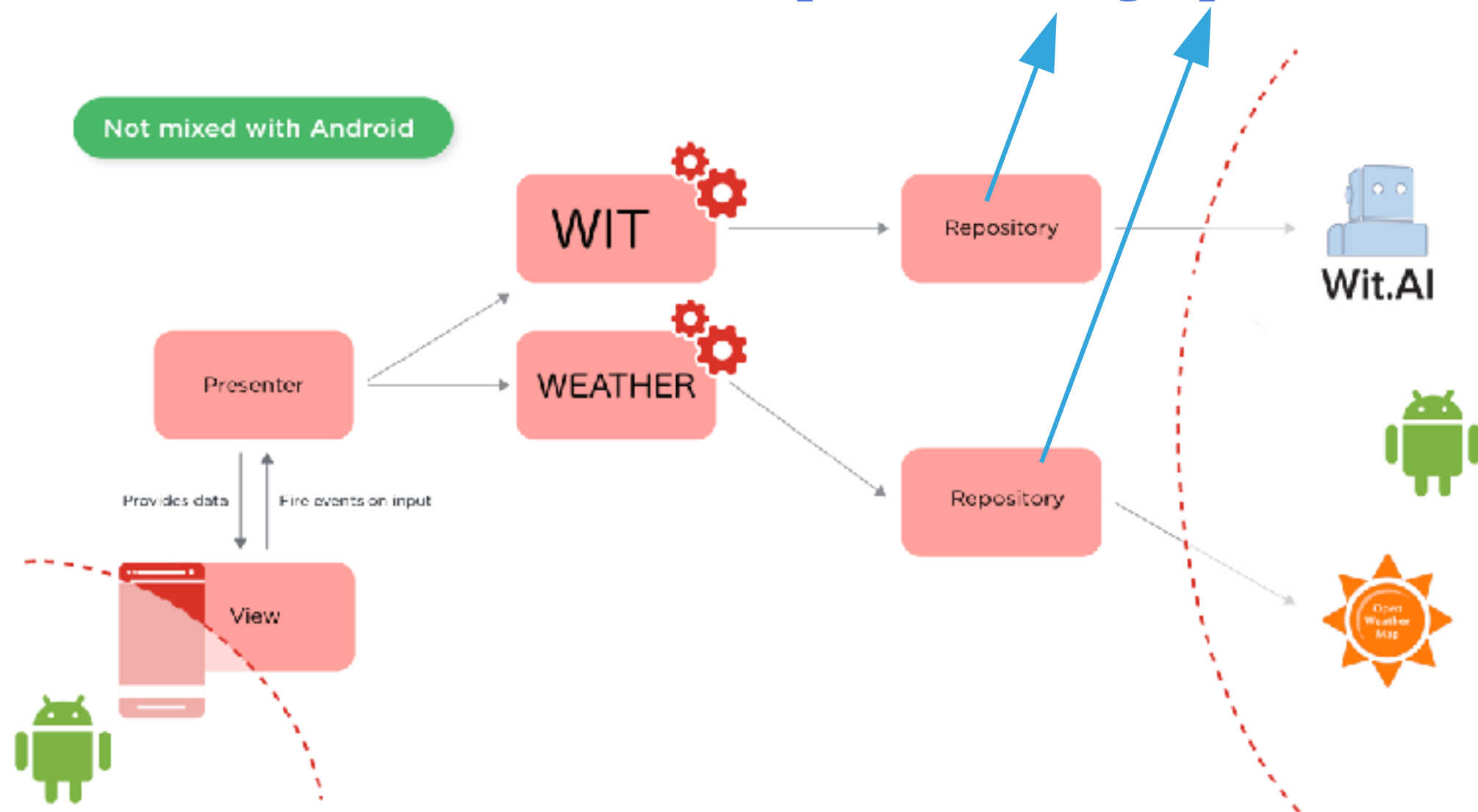


2.3 APPLICATION – CLEAN ARCHITECTURE



2.3 APPLICATION – CLEAN ARCHITECTURE

Repository pattern



3. NEXT STEPS

- ▶ Solve problem with Wit.AI Android > 6
- ▶ Dependency Injection - Using Dagger 2
- ▶ Android Rx
- ▶ Tests
- ▶ Independency Data Model between layers

**THANKS FOR
YOUR ATTENTION**