

## Open API

---

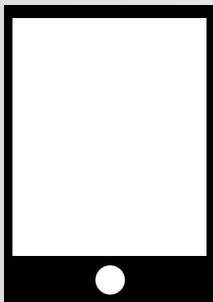
### INDEX

- 01 진행 상황
  - 02 최종 목적
  - 03 UUID DataCollect
  - 04 Hybrid WebApp
  - 05 참조
-

7/1	2	3	4	5	6	7
Open API 구조 확립 (~13)						
8	9	10	11	12	13	14
15	16	17	18	19	20	21
	Bluetooth API clustering					
22	23	24	25 APP connect	26	27	28
			bluetooth connect (~30)			
29	30	31	8/1	2	3	4
		bluetooth list (~8/5)				

Home Automation							01. 상세 일정
5	6	7	8	9	10	11	
		wifi connect (~13)					
12	13	14	15	16	17	18	
	wifi list (~20)						
19	20	21	22 Module API	23	24	25	
			connect(~9/3)				
26	27	28	29	30	31	9/1	
2	3	4	5	6	7	8	

Home Automation							01. 상세 일정
구분	단계	내용				비고	
UUID	1	bluetooth 장치의 UUID를 수집화				△	
APP	2	bluetooth connect LIB				○	
	3	bluetooth list LIB				△	
	4	wifi connect LIB				○	
	5	wifi list LIB				△	
UNO	6	bluetooth connect				△	
	7	wifi connect					
WEB	8	Log print				△	



Smart Device

1. Bluetooth\_connect
  2. Bluetooth\_list
  3. WIFI\_connect
  4. WIFI\_list
  5. WEB\_connect
- + Module\_search  
+ Voice\_record\_cmd  
+ SNS\_share



WEB SERVER

1. LOG\_time
- + Module\_list



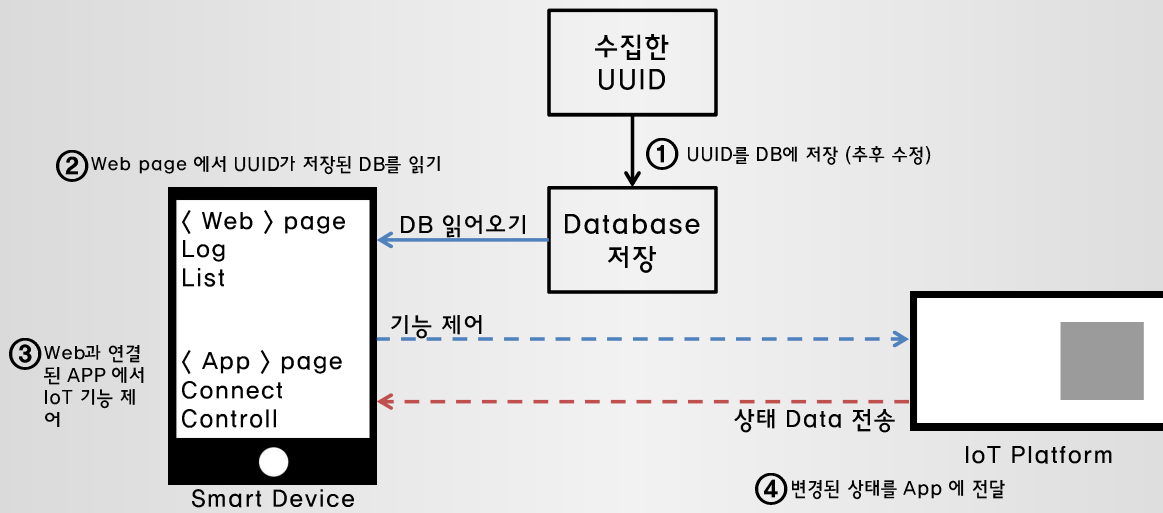
IoT Platform

1. Bluetooth\_connect
  2. WIFI\_connect
  3. wire\_PIN
- + LED\_controll  
+ Voice\_record

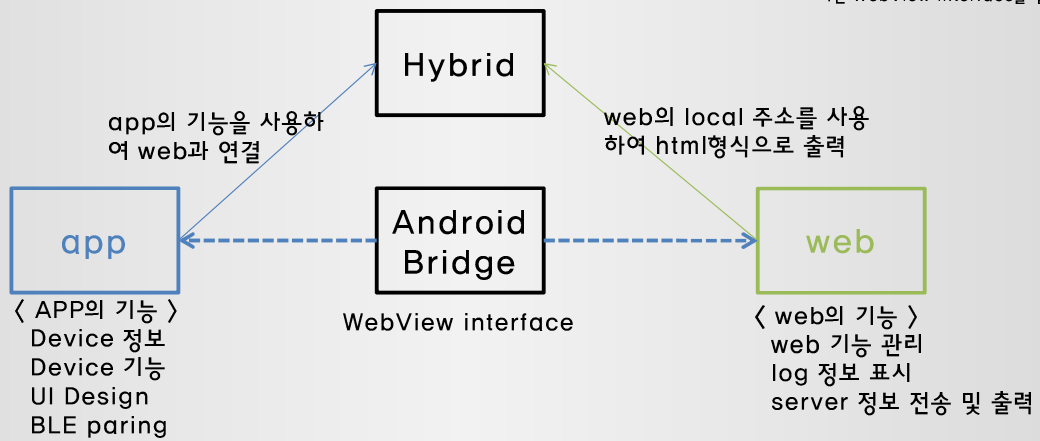
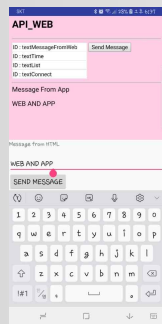
» 편리한 개발 환경

function	Action
Bluetooth_connect	App과 IoT의 bluetooth 연결 요청
Bluetooth_list	Device에 연결된 bluetooth 장비 목록 읽기
WIFI_connect	App과 IoT의 WIFI 연결 요청
WIFI_list	Device에 연결 가능한 WIFI 목록 읽기
WEB_connect	App 실행시 Web 연결
LOG_time	시간형식에 맞는 log data 읽기
Module_search	Bluetooth와 wifi로 연결된 Module 탐색
Voice_record_cmd	녹음 기능 및 녹음 명령 전송
SNS_share	SNS 공유기능
Module_list	APP 에 연결된 Module을 WEB에 출력
LED_controll	LED 제어
Voice_record	녹음 기능

## UUID Data Collect



➡ 수집한 UUID를 기반으로 WebApp을 통해 IoT Controll

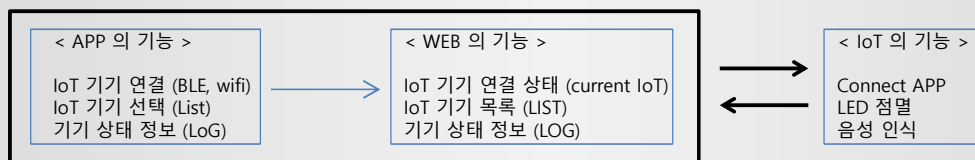


Android Bridge

javascript bridge로 안드로이드 환경에서 JavaScript 오브젝트를 생성하여 사용하는 WebView interface를 말한다.

➡ Android Bridge class 를 통해 app과 web의 데이터 교환

13



➡ App 하나로 web 기능을 사용, 관리 가능

14

## UUID collect

BrowseGroupDescriptorServiceClassID	00001001-0000-1000-8000-00805F9B34FB
PublicBrowseGroupServiceClass	00001002-0000-1000-8000-00805F9B34FB
SerialPortServiceClass	00001101-0000-1000-8000-00805F9B34FB
LANAccessUsingPPPSERVICEClass	00001102-0000-1000-8000-00805F9B34FB
DialupNetworkingServiceClass	00001103-0000-1000-8000-00805F9B34FB
IrMCSyncServiceClass	00001104-0000-1000-8000-00805F9B34FB
OBEXObjectPushServiceClass	00001105-0000-1000-8000-00805F9B34FB
OBEXFileTransferServiceClass	00001106-0000-1000-8000-00805F9B34FB
IrMCSyncCommandServiceClass	00001107-0000-1000-8000-00805F9B34FB
HeadsetServiceClass	00001108-0000-1000-8000-00805F9B34FB
CordlessTelephonyServiceClass	00001109-0000-1000-8000-00805F9B34FB
AudioSourceServiceClass	0000110A-0000-1000-8000-00805F9B34FB
AudioSinkServiceClass	0000110B-0000-1000-8000-00805F9B34FB

## 수집한 UUID 목록

UUID 기능 실행 lib

```

11 public static class Basic {
12     public static UUID service = UUID.fromString("0000fee0-0000-1000-8000-00805f9b34fb");
13     public static UUID batteryCharacteristic = UUID.fromString("00000000-0000-3512-2118-0009af100700");
14 }
15
16
17 public static class AlertNotification {
18     public static UUID service = UUID.fromString("00001802-0000-1000-8000-00805f9b34fb");
19     public static UUID alertCharacteristic = UUID.fromString("00002a06-0000-1000-8000-00805f9b34fb");
20 }
21
22 public static class HeartRate {
23     public static UUID service = UUID.fromString("0000180d-0000-1000-8000-00805f9b34fb");
24     public static UUID measurementCharacteristic = UUID.fromString("00002a37-0000-1000-8000-00805f9b34fb");
25     public static UUID descriptor = UUID.fromString("00002902-0000-1000-8000-00805f9b34fb");
26     public static UUID controlCharacteristic = UUID.fromString("00002a39-0000-1000-8000-00805f9b34fb");
27 }
28
29 public static class Information {
30     public static UUID service = UUID.fromString("0000fee0-0000-1000-8000-00805f9b34fb");
31     public static UUID Characteristic = UUID.fromString("00000007-0000-3512-2118-0009af100700");
32     public static UUID descriptor = UUID.fromString("00002902-0000-1000-8000-00805f9b34fb");
33 }
34

```