
수소가스 누출탐지시스템 애 플리케이션 개발 프로젝트

01 팀 소개

02 주제

03 개발 과정

04 제품 설명

05 어플 소개

06 발전 가능성

07 Q&A

팀 소개

강성주

- 일정
- 협력 기업과 컨택
- 통신관련 프로그래밍
- 애플리케이션 기능구현

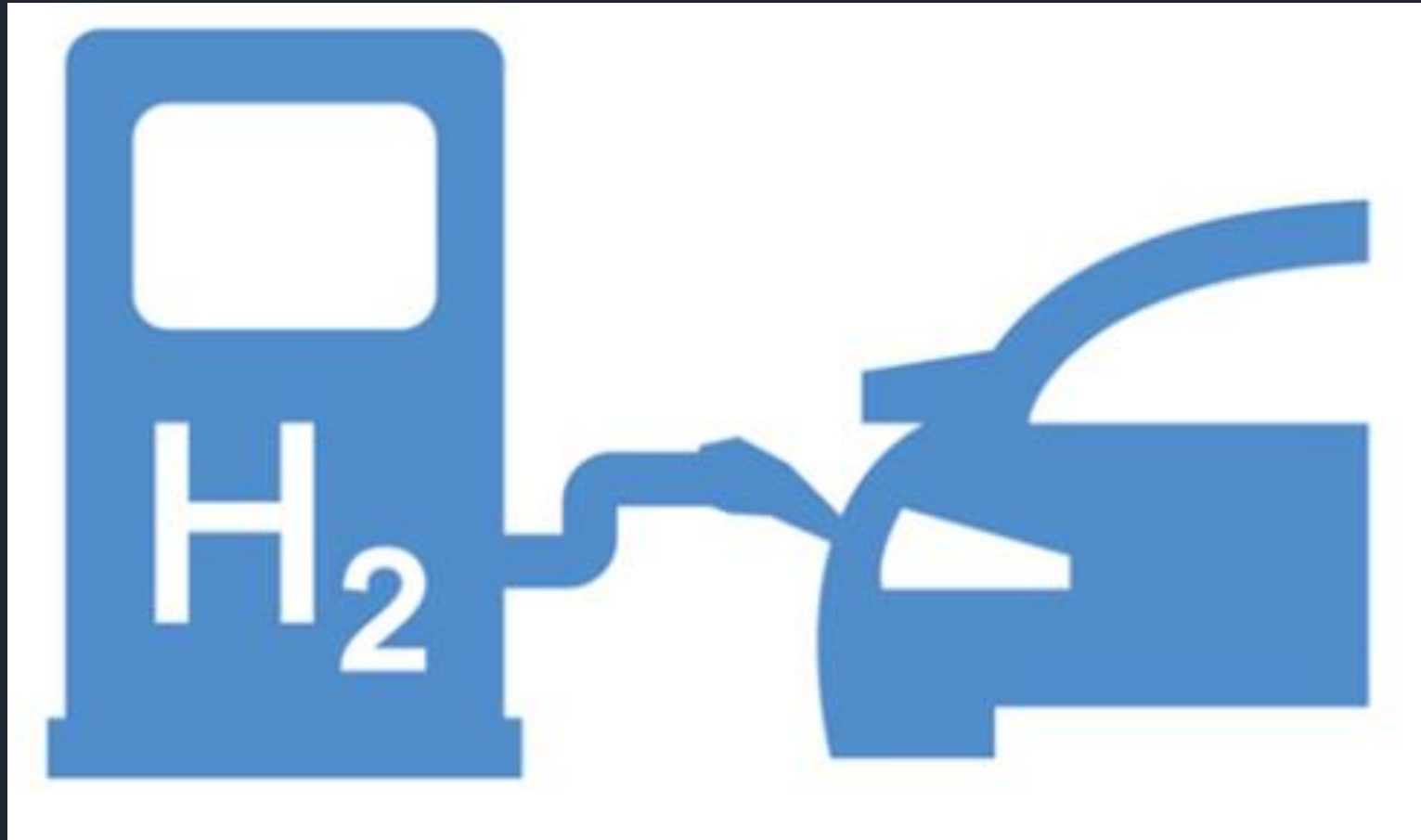
김구민

- 재료 구매 및 조사
- 모듈과 서버 간 통신
- 애플리케이션 개발

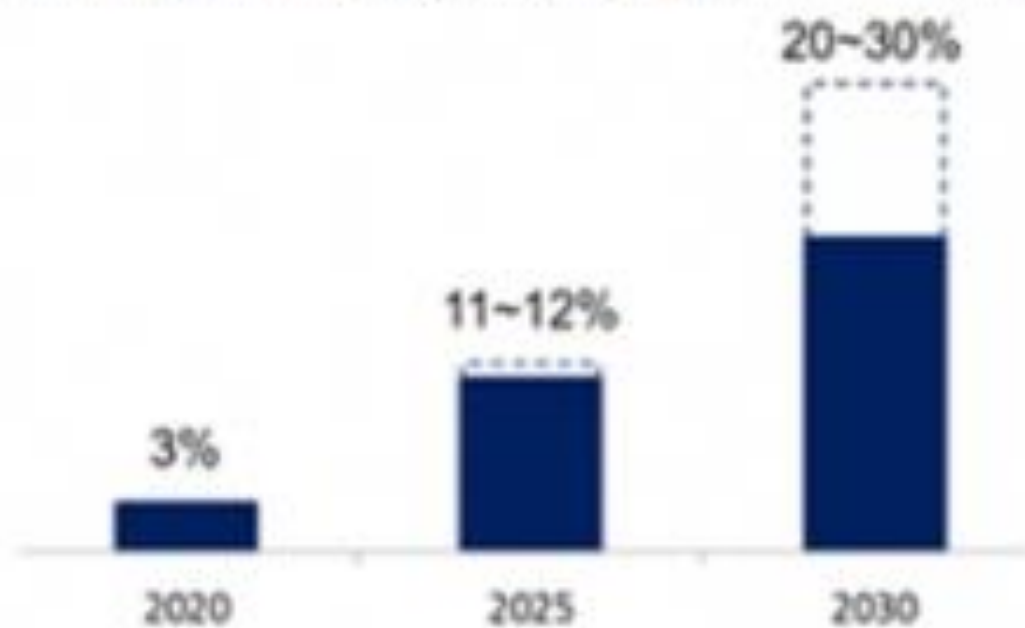
석인기

- 모듈 프로그래밍
- 애플리케이션 개발
- 보고서 작성

주제



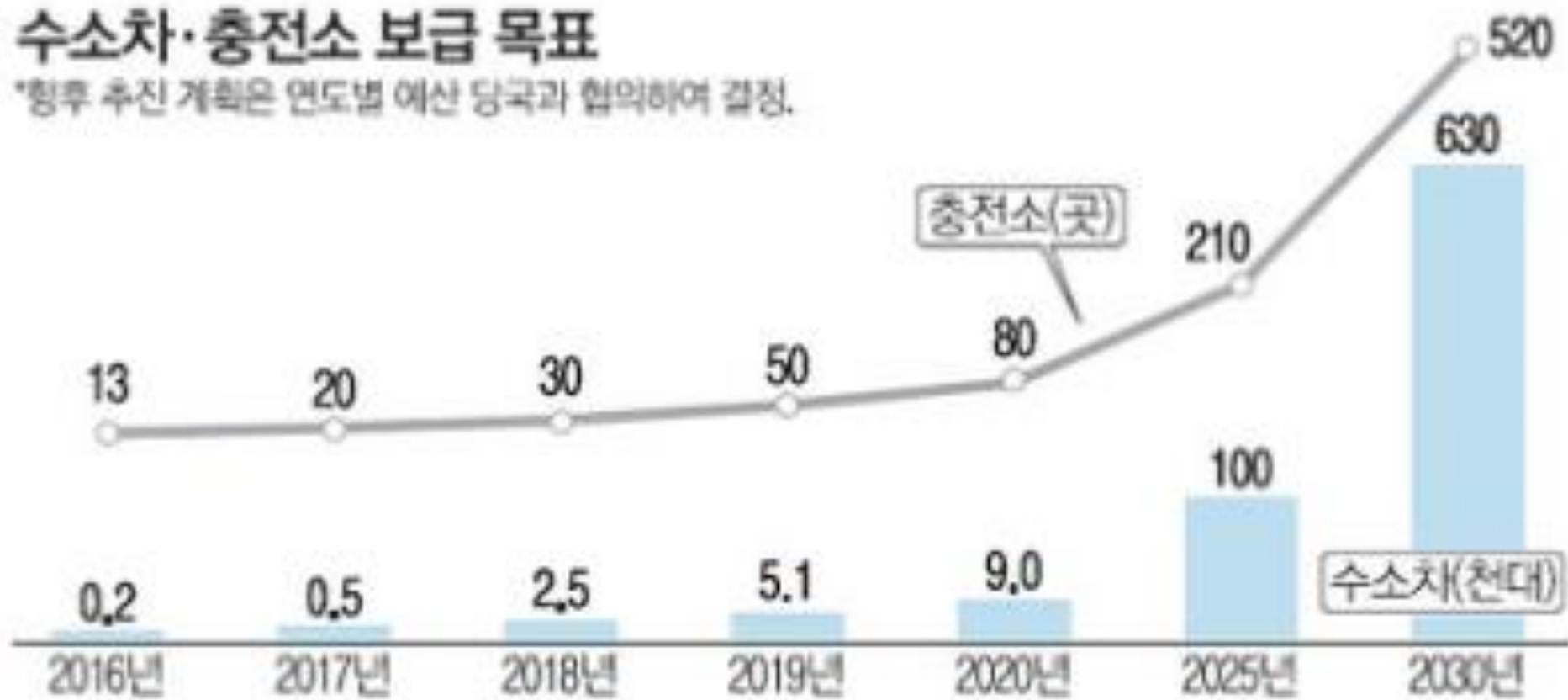
【 수소·전기차(SNE 등) 】



■ '30년 신차의 20 ~ 30%

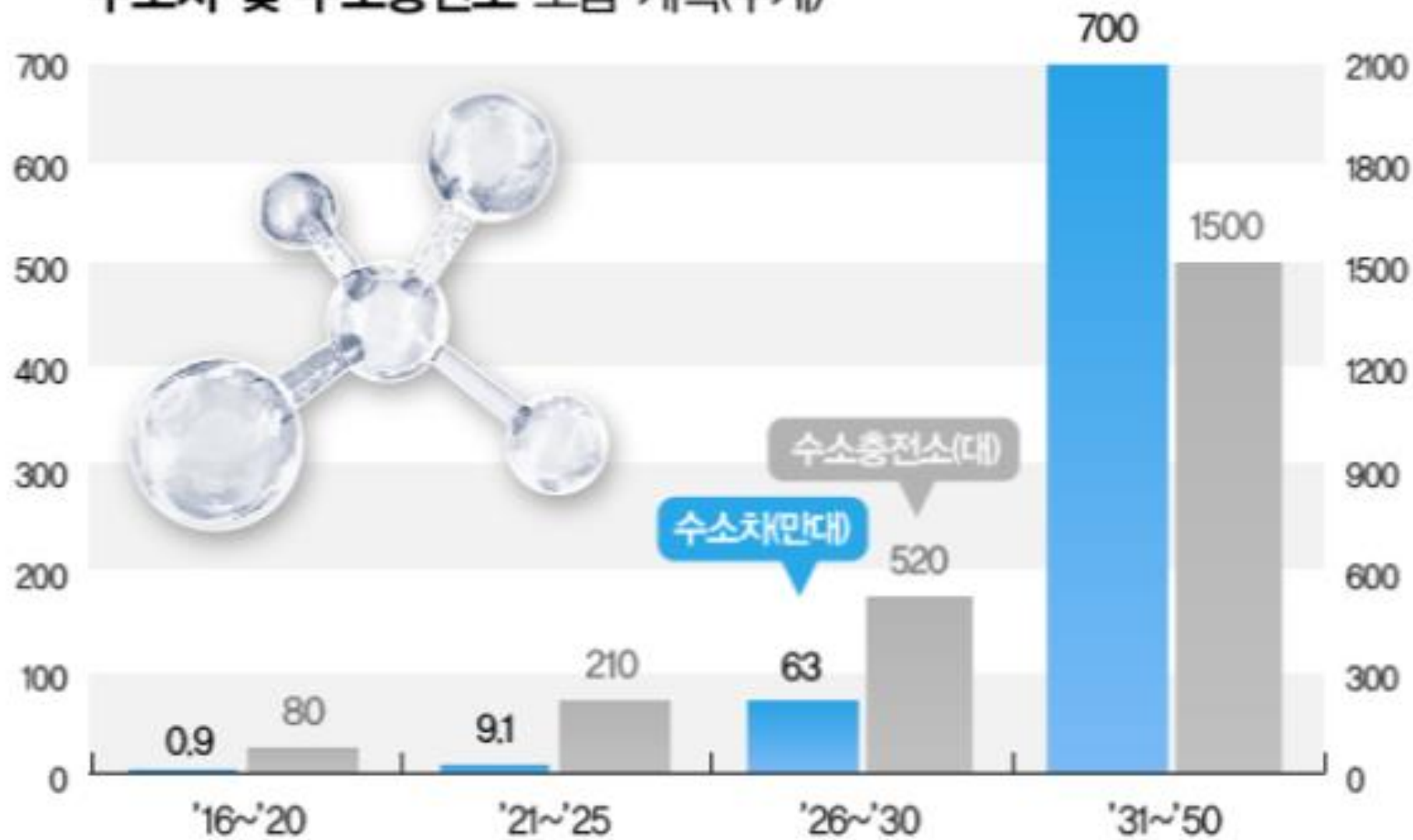
수소차·충전소 보급 목표

*향후 추진 계획은 연도별 예산 당국과 협의하여 결정.



▲ 수소차, 충전소 보급 목표

수소차 및 수소충전소 보급 계획(누계)





'폭발사고 없다'는 수소충전소, 정말일까?

👤 오하종 인턴 (hjoh@motorgraph.com) | 💬 댓글 0 | ⌚ 승인 2019.08.20 18:56



정부가 수소경제 활성화에 적극 나서는 가운데 안전에 대한 시민들의 불안감도 속속 제기되고 있다.

정부는 올해 1월 '수소경제 활성화 로드맵'을 발표하고, 국가 주도적으로 수소경제를 이끌어 나갈 것을 선포했다. 현대차그룹을 필두로 기업체들의 수소 산업에 대한 공격적인 투자도 이어지고 있다.

그러나 지난 5월 23일 강릉에서 수소탱크가 폭발해 2명이 죽고 6명이 중경상을 입는 사고가 발생하며 수소 안전성에 대한 불안감이 커졌다.

일각에서는 "안전을 장담하던 수소탱크가 폭발했으니, 수소차와 수소충전소 등도 절대 안전하다고 할 수 없다"라며 우려의 목소리를 내놓았다.

배경 및 문제

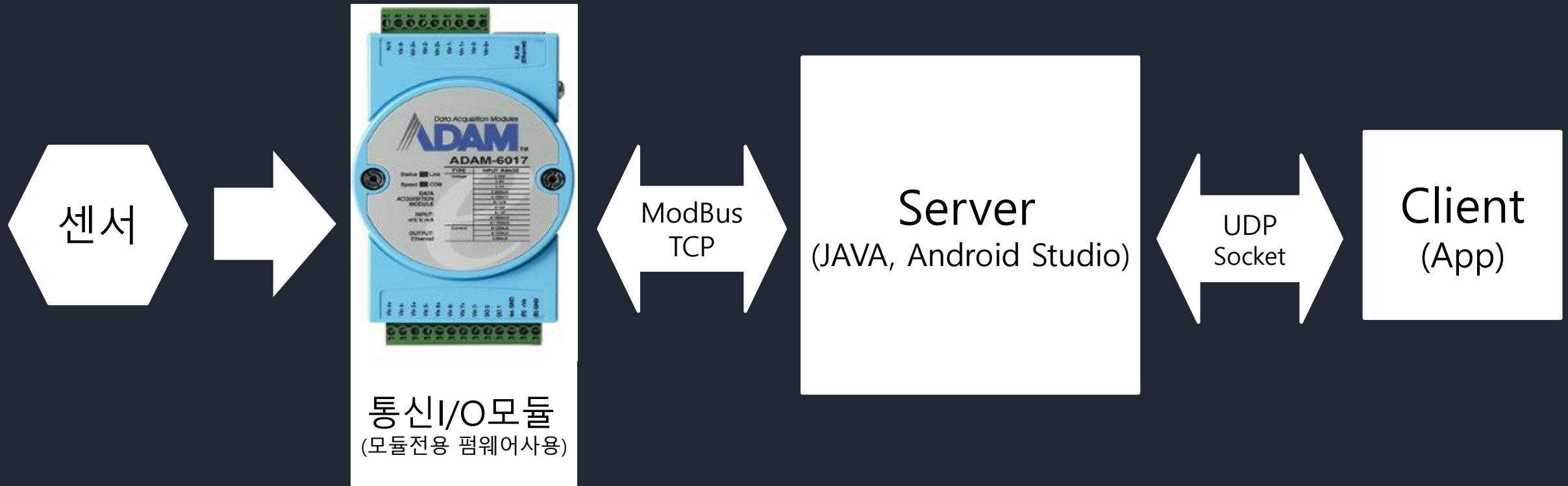
수소차와 수소 충전시설이 상용화됨에 따라
수소가스 누출 폭발사고와 같은 안전사고 문제 역시
사회적으로 대두되고 있고, 관련 안전산업도 중요
산업으로 떠오르고 있다.

목표

일정치 이상의 수소가스 탐지 시 알람이 울리며 관리가
가능한 애플리케이션 개발

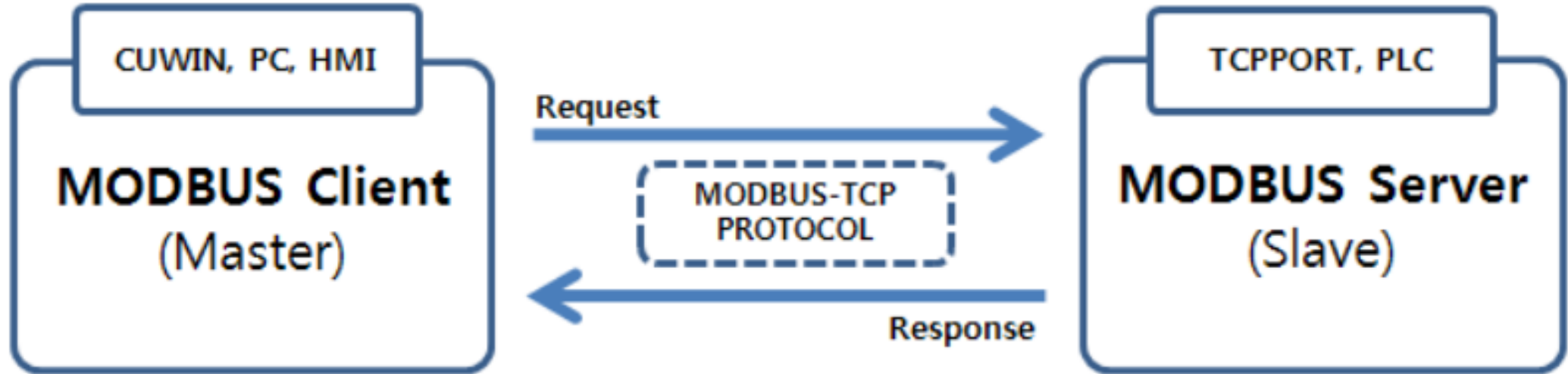
개발 기술

동작 flow



ModBus TCP

MODBUS-TCP는 산업용 필드버스 분야에서 자주 사용되는 통신 프로토콜입니다. TCP/IP 네트워크 상에서 상위 장치와 하위장치간의 인터페이스를 위하여 고안된 통신방법입니다.



ADAM-6017



- 산업용 통신 I/O 모듈
- Modbus TCP 방식의 프로토콜 지원
- 모듈 전용 자체 프로그램 지원
- 8채널 지원
- ANALOG INPUT / DIGITAL OUTPUT

개발

핵심 코드

```
import de.re.easymodbus.exceptions.ModbusException;  
import de.re.easymodbus.modbusclient.ModbusClient;  
  
ModbusClient modbusClient = new ModbusClient("192.168.0.121", 5000);  
  
modbusClient.Connect();
```

핵심 메소드

ReadHoldingRegister(int address, int length)

핵심 메소드

`WriteSingleRegister(int address, int value)`

어플 소개

가스 현황

1동

가스_0101	0
가스_0102	0
가스_0103	0
가스_0104	0

2동

가스_0105	0
가스_0106	0
가스_0107	0
가스_0108	0

3동

가스_0109	0
가스_0110	0
가스_0111	0
가스_0112	0

4동

가스_0113	0
가스_0114	0
가스_0115	0
가스_0116	0

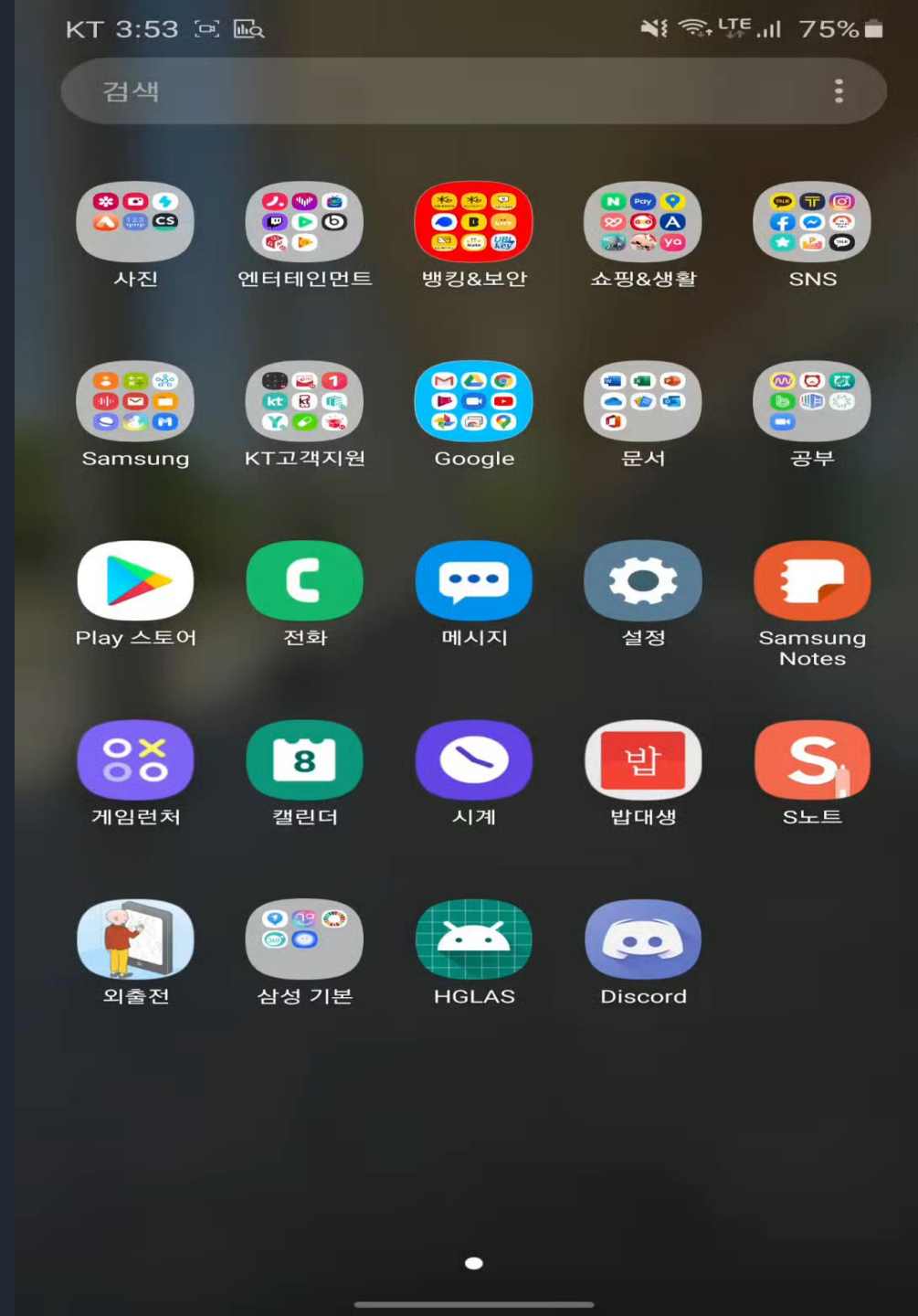
총 16개소 정상 작동 중

자세히

일시정지

리셋

사용자 계정 화면



관리자 계정 화면

KT 3:50 HGLAS

수소가스누출탐지 시스템

아이디 아이디를 입력해주세요

비밀번호 패스워드를 입력해주세요

로그인

(주) 솔락

The screenshot displays the Android Studio environment. The top toolbar shows icons for running the app, viewing logs, and other development tools. The main editor window shows the `LoginActivity.java` file. The code includes a `case 1:` section with a `try-catch` block for a login attempt, and a `case 2:` section with a loop for a password strength check. The `Run` console at the bottom shows the app starting successfully, with various system logs and a message indicating the app was installed successfully in 2 seconds and 509 milliseconds.

```

149         }
150         break
151     case 1:
152         System.out.println("라셋");
153         if (control) {
154             try {
155                 String reset = "reset";
156                 Toast.makeText(getApplicationContext(), "text: 라셋되었습니다.", Toast.LENGTH_SHORT).show();
157             } catch (Exception e) {}
158         }
159     } else {
160         try {
161             Toast.makeText(getApplicationContext(), "text: Application을 완전히 종료 후 관리자 계정으로 로그인 하세요.",
162             } catch (Exception e) {}
163         }
164     }
165     break
166 case 2:
167     System.out.println("1");
168     for (int i = 0; i < 16; i++) {
169         gas_s[i] = Byte.toString(gas[i]);
170         if (gas[i] > 20) {
171             flag[i] = true;
172             current_s[i] = (i + 1) + " / ";
173         } else {flag[i] = false;}
174         tv[i].setText(gas_s[i]);
175     }

```

Run: app x

```

D/InputTransport: Input channel constructed: fd=59
D/ViewRootImpl@f1672ab[MainActivity]: setView = DecorView@ec11d08[MainActivity] touchMode=true
D/ViewRootImpl@f1672ab[MainActivity]: dispatchAttachedToWindow
D/ViewRootImpl@f1672ab[MainActivity]: Relayout returned: oldFrame=[0,0][0,0] newFrame=[0,0][1200,1920] result=0x7 surface={isValid=true -566026240}
D/OpenGLRenderer: Initialized EGL, version 1.4
D/OpenGLRenderer: Swap behavior 1
D/mali_winsys: EGLint new_window_surface(egl_winsys_display*, void*, EGLSurface, EGLConfig, egl_winsys_surface**, egl_color_buffer_format*,
D/ViewRootImpl@f1672ab[MainActivity]: MSG_RESIZED_REPORT: frame=Rect(0, 0 - 1200, 1920) ci=Rect(0, 36 - 0, 0) vi=Rect(0, 36 - 0, 0) or=1
MSG_WINDOW_FOCUS_CHANGED 1
D/OpenGLRenderer: initializeIfNeeded()#2 mSurface={isValid=true -566026240}
V/InputMethodManager: Starting input: tba=android.view.inputmethod.EditorInfo@c9cb186 nm : com.example.hglas ic=com.android.internal.widget.
I/InputMethodManager: [IMM] startInputInner - mService.startInputOrWindowGainedFocus
D/InputTransport: Input channel constructed: fd=69
V/InputMethodManager: Starting input: tba=android.view.inputmethod.EditorInfo@c9cb186 nm : com.example.hglas ic=com.android.internal.widget.
MSG_RESIZED: frame=Rect(0, 0 - 1200, 1920) ci=Rect(0, 36 - 0, 0) vi=Rect(0, 36 - 0, 678) or=1

```

Install successfully finished in 2 s 509 ms.
App restart successful without requiring a re-install.

Run Logcat TODO Terminal Build Profiler Event Log

Install successfully finished in 2 s 509 ms: App restart successful without requiring a re-install. (a minute ago) 121:27 CRLF UTF-8 4 spaces



발전 가능성

데이터베이스 연동

스마트 밸브

제 2020-01 호

감사장



전남대학교 전자컴퓨터공학부 소프트웨어공학 전공

강 성 주

위 학생은 2019년12월26~2020년02월 24일까지 현장실습
교육과정 동안 (주)솔락 에서 국가연구개발 사업과제로
개발중인 수소융합 충전소용 안전관리 시스템에 적용 가능한
App 개발을 성공적으로 수행 하였기에 감사장을 수여합니다.

2020년 2월 24일

주식회사 솔락 대표 김 정



제 2020-02 호

감사장



전남대학교 전자컴퓨터공학부 소프트웨어공학 전공

김 구 민

위 학생은 2019년12월26~2020년02월 24일까지 현장실습
교육과정 동안 (주)솔락 에서 국가연구개발 사업과제로
개발중인 수소융합 충전소용 안전관리 시스템에 적용 가능한
App 개발을 성공적으로 수행 하였기에 감사장을 수여합니다.

2020년 2월 24일

주식회사 솔락 대표 김 정



Q&A
