**센서 모니터링 시스템 개발 보고서**

1. **소스 구조**  
   센서 모니터링 시스템 프로젝트는 깃허브로 코드를 공유한다. 팀원 별로 개발 범위와 프로그래밍 언어가 다르므로 담당 파트에 나눠 레포지토리를 구분해 놓았다. 가장 큰 틀인 Sensor-Monitoring-System 오거니제이션에 BackEnd, SlaveMCU, WEB(FrontEnd) 3개의 레포지토리가 있다.

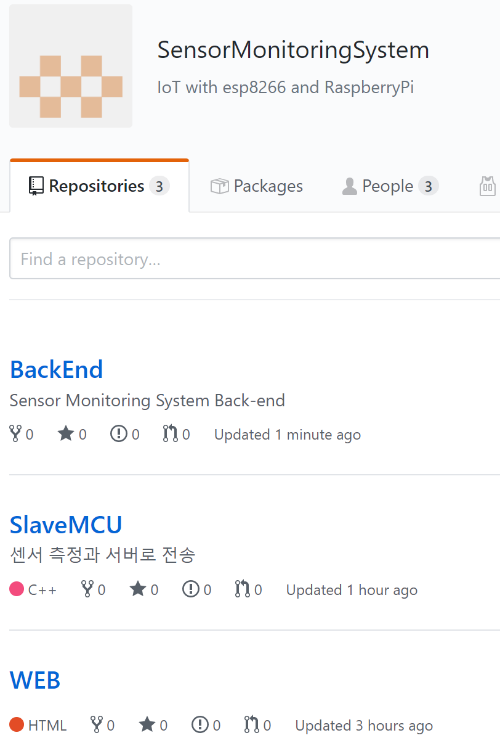


그림 1. <https://github.com/Sensor-Monitoring-System>

각 레포지토리 안에는 해당 파트의 기능을 수행하기 위한 코드들이 들어있다.

스크린샷이(가) 표시된 사진

자동 생성된 설명

그림 2. 오거니제이션 아래 레포지토리의 내용

1. **빌드 가이드**  
   - Slave(MCU) 빌드 가이드  
   슬레이브는 esp8266 WiFi모듈이 장착된 NodeMCU나 위모스d1 등을 MCU로 사용한다. 가장 범용적인 개발 IDE은 Arduino를 이용하는 것이나 코드 하이라이트, 단어 추천 등등 빈약한 부분이 많아서 Visual Studio 확장 도구인 Arduino IDE for Visual Studio를 선택했다. IDE Location의 Optional additional boards manager urls에  
   <http://arduino.esp8266.com/stable/package_esp8266com_index.json>를 입력하고 확인 버튼을 누르면 esp8266에 대한 예제, 기본 라이브러리를 사용할 수 있고 코드 업로드를 할 수 있게된다.

스크린샷이(가) 표시된 사진

자동 생성된 설명

그림 3. 확장 도구 설치

스크린샷이(가) 표시된 사진

자동 생성된 설명

그림 4. 추가 보드 매니저 url 입력

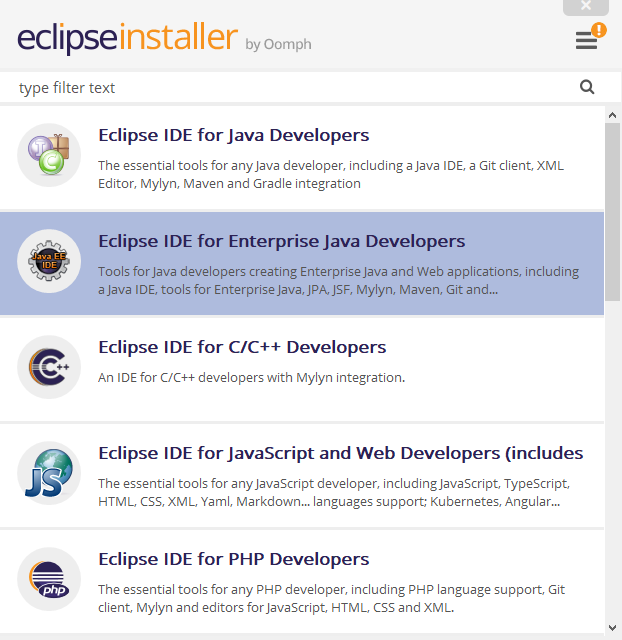
Arduino IDE for Visual Studio의 설치와 기본 설정이 끝났으면 비주얼 스튜디오에서 아두이노 프로젝트를 만들 수 있고 \*.ino파일을 작성해서 타겟 MCU에 업로드하면 된다.  
- WEB 빌드 가이드  
 이클립스 설치 파일을 이클립스 홈페이지에 들어가서 각 컴퓨터 구성에 맞게 설치하고 설치파일을 누르게 되면 eclipseinstaller가 나타날텐데 여기서 2번째 부분(Enterprise Java Developers)을 눌러 설치해주면 된다.  


그림 . Enterprise Java Developer 설치 화면

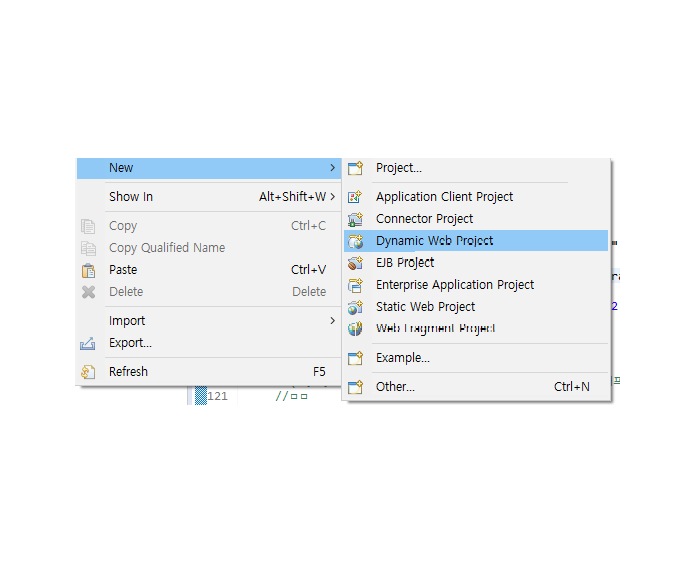
그 다음 tomcat을 설치해준다. eclipse버전에 맞는 tomcat을 홈페이지에서 찾아서 설치해줘야 한다. zip파일을 다운 받은 후에 압축 해제해서 폴더형식으로 저장해두면 되고 Eclipse에 들어가서 우선 project Explorer에 우클릭을 해서 New -> others -> server ->server를 눌러서 생성해주면 되고 설치한 apache-tomcat의 버전에 맞게 눌른 후 finish를 눌러서 server를 생성해준다.  
 웹 페이지를 구성하는 파일을 저장할 공간을 만들어야 되는데 위와 같이 project Explorer에 우클릭을 해서 dynamic web project를 생성해주고 원하는 프로젝트명을 입력한다.  


그림 . Dynamic Web Project 선택 화면

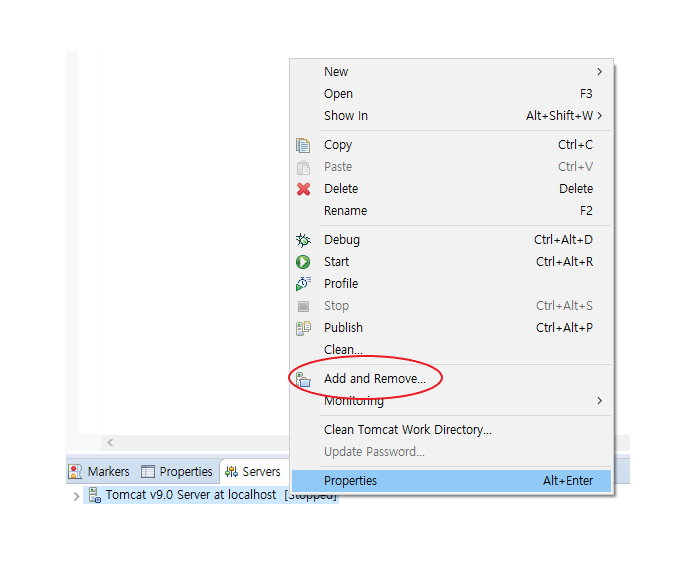
그 다음 서버에 프로젝트를 올려줘야 실행할 때마다 서버를 통해 실행이 가능해진다. 아래 목록중에 server를 우클릭하고 add and remove를 눌러 원하는 프로젝트를 오른쪽으로 옮겨 추가해주면 된다.  


그림 . 프로젝트 실행 방법

이 후 프로젝트에 기본적으로 만들어지는 WebContent에 github의 Web에 존재하는 모든 파일을 넣어주고 Sensor\_monitor.html파일을 통해 실행을 누르면 만들어진 페이지를 실행할 수 있다.

- BackEnd 빌드 가이드  
node

∘ 서버를 개발할 Node 설치

∘ sudo apt-get install nodejs를 통해 라즈베리 위에 설치

npm (node package manager )

∘ node.js 로 만들어진 모듈을 관리해주는 프로그램

∘ sudo apt-get install npm을 통해 라즈베리 위에 설치

npm init을 통한 프로젝트 생성

express

∘ 서버를 손쉽게 구축하기 위한 웹 서버 프레임워크 express 설치

∘ npm install express를 통해 생성한 프로젝트 위에 설치

http

∘ 데이터 통신을 위한 http module 설치

∘ npm install http를 통해 생성한 프로젝트 위에 설치

body-parser

∘ 요청을 받을 때(request) body-parser를 통해 값을 받아오기 위해 사용

∘ npm install body-parser를 통해 생성한 프로젝트 위에 설치

sync-mysql

∘ mysql 의 DB를 사용할 때 동기적으로 사용하기 위해 sync-mysql 설치

∘ npm install sync-mysql을 통해 생성한 프로젝트 위에 설치

async

∘ 각 서버 기능에서 callback 함수를 사용하기 위한 async module 설치

∘ npm install async를 통해 생성한 프로젝트 위에 설치

1. **실행 가이드**  
   Sensor Monitoring System을 실행하려면 esp8266기반의 MCU와 라즈베리파이가 있어야 한다. 라즈베리파이는 서버의 기능을 수행할 수 있는 가장 작은 크기의 컴퓨터를 추려낸 것이고 데스크톱이나 노트북이라도 상관은 없다.  
   - MCU는 SlaveMCU 레포지토리의 ino파일을 받아서 아래 코드에 자신의 와이파이 정보를 입력해 넣는다.  
     
   const char\* ssid = "와이파이 이름";  
   const char\* password = "와이파이 비밀번호";  
     
   제대로 와이파이 정보를 입력하고 실행하면 시리얼 모니터로 다음과 같은 출력을 확인할 수 있다.  
     
   WiFi connected  
   --------------  
   와이파이 이름  
   IP address: xxx.xxx.xxx.xxx  
   --------------

그 다음엔 현재 센서 값과 서버와 통신하는 과정을 시리얼 모니터로 확인할 수 있다.

1. **테스트 가이드**  
   서버 ip로 전송한 센서값이 웹 화면에 잘 출력되는지 확인한다. (until zero feature)
2. **기타 가이드**  
   - NodeMCU에 연결하는 센서들의 배선도를 확인하려면 SlaveMCU 레포지토리의 pinMap파일을 확인한다. (still zero feature)  
   - 요구사항 분석 UML다이어그램에 관한 문서는 Document 레포지토리를 확인한다. (still zero feature)