**Sistemi Embedded e IoT - a.a. 2018-2019**

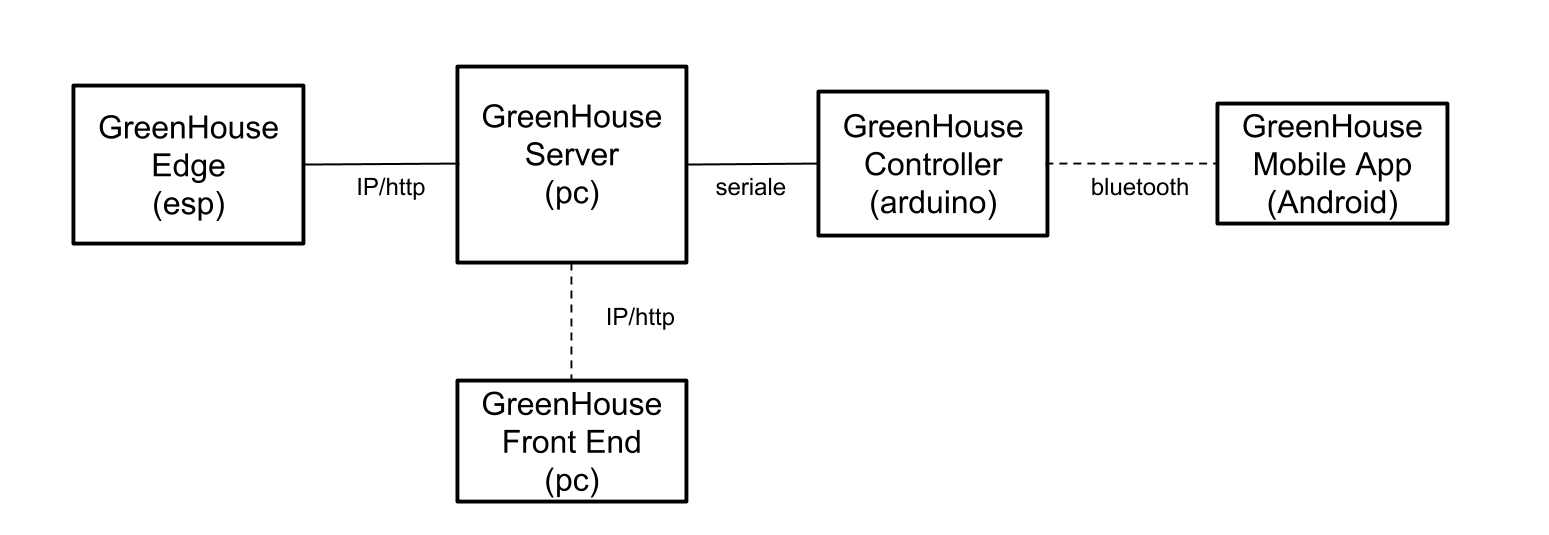
**Progetto #3 - *Smart Greenhouse***

v1.0-20181207

Si vuole realizzare un sistema embedded integrato che rappresenti una versione semplificata di una *serra smart.*

Il compito della serra smart è l’irrigazione automatizzata (di un certo terreno o pianta) implementando una strategia che tenga conto dell’umidità percepita, con la possibilità di controllare e intervenire manualmente mediante mobile app.

Il sistema è costituito da 5 parti (sotto-sistemi):



* **GreenHouse Server (PC)**
  + contiene la logica che definisce e attua la strategia di irrigazione
* **GreenHouse Controller (Arduino)**
  + permette di controllare l’apertura e chiusura degli irrigatori (pompe acqua), quindi della quantità di acqua erogata al minuto
* **GreenHouse Edge** **(ESP)**
  + permette di percepire l’umidità del terreno
* **GreenHouse Mobile App (Android)**
  + permette di controllo manuale della serra
* **GreenHouse Front End** **(PC)**
  + Front endper visualizzazione/osservazione/analisi dati

*Dettaglio componenti HW di Controller (Arduino) e Edge (ESP)*

* GreenHouse Controller
  + due led verdi L1 e L2  un led rosso Lm
  + 1 sonar S per distanza
  + 1 servo motore M con cui si attua l’apertura e chiusura di una pompa acqua
  + 1 modulo Bluetooth HC-06 o HC-05
* GreenHouse Edge
  + 1 sensore umidità - simulato dal potenziometro analogico presente nel kit
    - oppure, per chi dispone di un proprio kit: sensore digitale DHT11 o DHT22

*Comportamento dettagliato del sistema*

Il sistema parte in modalità AUTO, in cui avviene l’irrigazione automatica. In questo caso:

* Il led L1, è acceso a indicare che il sistema è attivo, in modo AUTO. L2 e Lm sono spenti.
* Quando viene percepito un valore di umidità U (in percentuale) inferiore a soglia Umin, viene aperta la pompa erogando una certa portata (quantità di acqua nel tempo), pari a Y litri al minuto, dove Y può assumere tre valori: Pmin (portata minima), Pmed (media), Pmax(massima)
  + Y è legata a U dalla formula Firrig
  + quando viene erogata acqua, si deve accendere L2 con intensità che riflette la portata
* L’erogazione si ferma quando:
  + il valore supera la soglia Umin + un certo *DeltaU*
  + La durata dell’erogazione ha superato un tempo *Tmax*. In questo caso viene creata una segnalazione
* Mediante Front End deve essere possibile visualizzare lo stato della serra, i dati storici (andamento umidità nel tempo, quando c’è stata irrigazione e per quanto tempo) e segnalazioni

Mediante il dispositivo mobile, considerando di essere nelle vicinanze della serra a una distanza dal Controller inferiore o uguale a DIST - deve essere possibile connettersi al sistema e passare a una modalità MANUAL in cui mediante Mobile App si possa:

* manualmente aprire/chiudere/regolare l’erogazione dell’acqua, specificando la portata (litri al minuto)
* visualizzare (continuamente) il valore corrente dell’umidità percepita

Quando il sistema è in modalità di controllo manuale, il led L1 si deve spegnere e deve essere acceso Lm.

Realizzare il sistema con le seguenti specifiche:

* **GreenHouse Server** in esecuzione su un PC
  + La logica applicativa deve essere implementata come applicazione Java, usando un’architettura a event-loop che implementa una macchine a stati finiti asincrona
  + La parte server che interagisce con Edge e Front End può essere implementata usando la tecnologia che si ritiene più opportuna
* **GreenHouse Controller**  basato su piattaforma Arduino
  + Implementare la logica in termini di macchina a stati finiti sincrona
* **GreenHouse Edge** basato su piattaforma ESP
  + E’ possibile usare lo stack ESPRUINO o Arduino Core,
* **GreenHouse Mobile App** basato su piattaforma Android
  + Fisica o emulata: nel caso fisico, la comunicazione con Controller deve avvenire mediante bluetooth; nel caso emulato, la comunicazione con Controller può avvenire mediante seriale, usando lato PC ove è in esecuzione l’emulatore Android il bridge emulatore-seriale presentato in laboratorio
* **GreenHouse Front End** - basato su browser o client su PC
  + Può essere implementata come web app (quindi basato su protocollo HTTP) con la tecnologia che si ritiene più opportuna o anche come client usando socket TCP o UDP

Assumere come valori per simulare il sistema:

DIST (Engagement distance) = 0.3 m

Umin = 30% DeltaU = 5%

Firrig: U → {Pmin, Pmed, Pmax) =

Pmin se U in {20..30}, Pmed se U in {10..20}, Pmax se U < 10

Tmax (tempo massimo erogazione) = 5 sec

Per tutti gli aspetti non specificati, fare le scelte che si credono più opportune.