

# PROJECTE ADSTR

## PROJECTE 19/20:

### OBJECTIUS:

- Dissenyar i implementar una aplicació de control d'un sistema en una plataforma embedded per assolir les competències específiques.
- Desenvolupar les competències transversals establertes en l'assignatura d'Anàlisi i disseny d'un sistema de temps real.

## 1. SISTEMA EMBEDDED MONITOR ESTACIÓ SOLAR

El controlador que implementarem es desenvoluparà sobre un sistema de plataformes embedded (Raspberry Pi en diversos formats i plaques d'expansió)

El sistema monitoritza diferents paràmetres de la estació solar que disposem a l'EUSS.

## 2. PARTS DEL PROJECTE

A nivell de maquinari es disposa d'una placa Raspberry Pi i plaques d'expansió. Posteriorment s'afegiran altres elements a controlar de forma distribuïda i utilitzar diferents sistemes de comunicació (IoT).

El projecte es dividirà en 3 parts:

### 2.1 Part 1: Sistema de captura de diverses magnituds físiques (temperatura, humitat relativa, tensions, intensitats,...) i informe diari per e-mail.

#### Aplicació captura

- Cal crear un programa en codi C anomenat 'captura' que s'executarà des de RPi.
- Aquest **llegeix cada X temps** les magnituds físiques
- Insereix la informació a la base de dades implementada sobre SQLite que es guarda a la RPi.
- La base de dades ha de tenir la següent estructura:
  - Taula sensors
    - Camps:
      - Identificador sensor
      - Tipus
      - Descripció
  - Taula lectures
    - Camps:
      - Data/hora
      - Identificador sensor
      - Valor
  - Taula alarmes
    - Camps:
      - Data/hora
      - Motiu
- El programa haurà de registrar a la base de dades si el sistema porta **Y minuts** amb unes condicions no desitjades (a definir en cada cas).
- Cal encendre un led quan inicia la lectura i apagar-ho quan acaba (per visualitzar quan es fa la mesura)**
- Posar-ho en marxa de forma automàtica quan s'inicia el sistema en la Rpi.

#### Aplicació informe

-Cal crear un programa en codi C anomenat 'informe' que s'executarà des de RPi.  
-El programa consulta la base de dades implementada en SQLite, i realitza un informe de l'evolució de les diverses magnituds de l'últim dia.  
Aquest informe es guardarà en un fitxer local i haurà de tenir la següent informació per cada sensor:

- Data/hora inici,
- Data/hora final,
- Valor màxim,
- Valor mínim,
- Valor mig.

-Llistat d'alarmes: data/hora inici, motiu

- Aquesta aplicació s'executarà periòdicament un cop cada 24 hores (cada dia) a través del servei *cron* del sistema operatiu.

#### En comú

En el desenvolupament de les anteriors funcionalitats caldrà satisfer els següent requeriments:

-El codi ha d'estar en un projecte a *GitHub* i cal portar un control de versions amb *GIT*

(fer 'fork' del projecte [https://github.com/afontquerni/ADSTR\\_2019](https://github.com/afontquerni/ADSTR_2019) )

-Caldrà que el codi C segueixi el **conveni** d'estil establert en el projecte kernel Linux i la documentació del codi s'ha de fer en DOXYGEN.

-El projecte s'ha de compilar, netejar i instal·lar usant d'un fitxer *Makefile*

S'utilitzaran diverses RPi per a aquesta finalitat.

## **2.2 Part 2: Integrar comunicacions**

El sistema haurà d'incorporar les següents funcionalitats afegides a les de la primera part:

#### Aplicació captura

-Amb un client HTTP obtenir els valors de configuració de «**X segons lectura**» i «**Y minuts incidència**» d'un servidor.

-Cal crear una funció que farà la petició web. Aquesta funció ha de compilar-se per separat i crear una biblioteca dinàmica.

- El programa principal haurà d'utilitzar aquesta biblioteca per fer la petició web.

-El programa permetrà passar per paràmetres per línia de comandes. (p.ex. nom\_bbdd, ip\_servidor, ...)

#### Aplicació informe

-L'informe generat també cal enviar-lo per e-mail.

-Cal crear una funció que enviarà el correu electrònic amb l'informe. Aquesta funció ha de compilar-se per separat i crear una biblioteca dinàmica.

- El programa principal haurà d'utilitzar aquesta biblioteca per enviar correu.

-La resta del codi del programa de l'informe diari ha d'estar en dos fitxers font (.c): un amb la funció main i l'altre amb la resta de funcions que s'hagin creat.

-El programa permetrà passar per paràmetres per línia de comandes (p.ex. nom\_bbdd, adreça\_email\_destí, ...)

#### En comú

En el desenvolupament de les anteriors funcionalitats caldrà satisfer els següent requeriments:

-El codi ha d'estar en un projecte a *GITHUB* i cal portar un control de versions amb *GIT*

-Caldrà que el codi C segueixi el **conveni** d'estil establert en el projecte kernel Linux i la documentació del codi s'ha de fer en DOXYGEN.

-El projecte s'ha de compilar, netejar i instal·lar usant d'un fitxer *Makefile*

S'utilitzaran diverses RPi per a aquesta finalitat.

### 2.3 Part 3: Portal central informació

Es vol centralitzar totes les dades del sistema distribuït de sensors enviant les dades a un portal central d'informació i control

Els sensors via peticions web concretes (API) registraran les seves mesures i obtindran la configuració necessària pel seu funcionament.

-El sistema servirà una pàgina web amb la informació del sistema i les dades registrades. Així com la configuració. Es dóna llibertat per implementar-ho segons la creativitat del grup.

- Ampliar sistemes amb més sensors

- Treure estadístiques de funcionament i quina precisió de temps real s'aconsegueix i com millorar-ho.

#### En comú

En el desenvolupament de les anteriors funcionalitats caldrà satisfer els següent requeriments:

-El codi ha d'estar en un projecte a *GITHUB* i cal portar un control de versions amb *GIT*

-Aquesta part s'implementarà amb llenguatge Python

## 3. ORGANITZACIÓ I TERMINIS

El projecte es realitzarà en 2 gran grups de 6 estudiants dividits en 2 subgrups de 3 estudiants. A EUSSternet es publicarà la composició de cada grup.

- **subgrup A1: Sensor Temperatura + Captura**
- **subgrup A2: Sensor Humitat + Informe**
- **subgrup B1: Sensor Tensió + Captura**
- **subgrup B2: Sensor Corrent + Informe**
- 

Cada grup tindrà un secretari que realitzarà l'acta de seguiment del treball. Aquest càrrec serà rotatori.

Periòdicament caldrà realitzar una reunió de 10 minuts amb presència del professor per revisar l'estat d'execució del projecte i planificar la feina fins a la següent setmana.

Els terminis són els següents:

- Part 1: 29-octubre-2019
- Part 2: 26-novembre-2019
- Part 3: 15-gener-2020