



UNIVERSITAT OBERTA DE CATALUNYA (UOC)  
MÀSTER UNIVERSITARI EN ENGINYERIA INFORMÀTICA

## TREBALL FINAL DE MÀSTER

ÀREA: SERVEIS BASATS EN LOCALITZACIÓ I ESPAIS INTEL·LIGENTS

### **Control d'aforament en un espai tancat** **Implementació d'un sistema capaç de comptar les persones que** **entren i surten d'un espai tancat.**

---

Autor: Enric Bartrés Jurado

Tutor: Jordi Pino Lacosta

Professor: Sergio Trilles Oliver

---

Barcelona, 3 de febrer de 2021



# Crèdits/Copyright

Aquest document està subjecte a una llicència de reconeixement -NoComercial- SinObraDerivada. [3.0 España de CreativeCommons](#).



El codi està subjecte a una llicència de Reconeixement – NoComercial – SinObraDerivada [3.0 España de CreativeCommons](#).





# FITXA DEL TREBALL FINAL

Títol del treball:	Control d'aforament en un espai tancat
Nom de l'autor:	Enric Bartrés Jurado
Nom del professor:	Sergio Trilles Oliver
Nom del tutor:	Jordi Pino Lacosta
Data d'entrega (mm/aaaa):	12/2020
Titulació o programa:	Màster Universitari en Enginyeria Informàtica
Àrea del Treball Final:	Serveis basats en localització i espais intel·ligents
Idioma del treball:	Català
Paraules clau	Aforament, IOT, Sensor, Detecció, MQTT, Dashboard, Placa electrònica, Arduino IDE, Node-RED, MySQL



# Dedicatòria/Cita

Als meus fills Gerard i Bruno per les estones que ens hem perdut i que de ben segur recuperarem després d'aquest treball. Al meu marit Xavi, per la paciència que ha tingut i per tots els moments que vivim junts en el que ben aviat, esperem també, poder afegir la celebració d'aquest títol.

Al professor Sergio Trilles, per ajudar-me a donar forma a la idea de crear un prototip d'aquestes característiques.

Als meus pares per estar sempre al meu costat. I als meus companys de feina i amics per creure en mi i donar-me la vostra confiança.





# Resum

La situació d'alerta sanitària que estem vivint actualment en tot el planeta fa que ens haguem de plantejar solucions eficaçes per tal de que el virus no segueixi propagant-se, i a la vegada poder seguir realitzant amb la màxima normalitat possible les tasques diàries i socials de la nostra vida quotidiana. Segons l'Organització Mundial de la Salut (OMS), una de les mesures de seguretat de protecció bàsiques contra la COVID-19 és mantenir la distància social d'almenys un metre. De la mateixa manera, el ministeri de sanitat recomana fixar una distància de seguretat interpersonal de dos metres i usar mascareta higiènica. Aquesta mesura és fàcil de complir quan es passeja pel carrer però no resulta tant assequible en espais tancats. Per tant, un element clau per aconseguir aquesta distància és establir una limitació d'aforament en els espais tancats.

En aquest treball s'ha realitzat un mecanisme capaç de controlar l'aforament de persones que hi ha en una sala d'espera d'un centre sanitari tenint en compte l'entrada i la sortida de persones i en el cas de que s'arribi a l'aforament màxim permès el sistema dona una alerta informant de que ningú més pot accedir a la sala.

**Paraules clau:** Aforament, IOT, Sensor, Detecció, MQTT, Dashboard, Placa electrònica, Arduino IDE, Node-RED, MySQL



# Abstract

The health alert situation that we are currently living all over the planet means that we must contemplate effective solutions so that the virus does not continue to spread and so that at the same time, we are able to carry on with our daily tasks and social lives. According to the World Health Organization (WHO), one of the basic safety measures against COVID-19 is to maintain a social distance of at least one meter. In the same manner, the Ministry of Health recommends having an interpersonal safety distance of two meters and wearing a mask. This measure is easy to comply with when walking down the street but it is not as feasible when people are indoors. Therefore, a key element to achieve this distance is to establish a capacity limitation in enclosed spaces.

In this essay I have created a system capable of controlling the capacity of people in a waiting room of a health centre taking into account the entry and exit of people and, in the event that it reaches its maximum capacity, the system will give an alert informing that no one else is allowed to access the room.

**Keywords:** Capacity, IOT, Sensor, Detection, MQTT, Dashboard, Electronic Board, Arduino IDE, Node-RED, MySQL



# Índex

Resum	vii
Abstract	ix
Índex	xi
Llistat de Figures	xiii
Llistat de Taules	1
<b>1 Introducció</b>	<b>3</b>
1.1 Temàtica del projecte . . . . .	4
1.2 Competències adquirides . . . . .	4
1.3 Descripció de la problemàtica . . . . .	4
1.4 Objectius . . . . .	5
1.4.1 Objectiu principal . . . . .	5
1.4.2 Objectius secundaris . . . . .	5
<b>2 Antecedents</b>	<b>7</b>
2.1 Detecció de persones . . . . .	8
2.1.1 Sensors . . . . .	8
2.1.2 Altres sistemes de control . . . . .	10
2.1.3 Treballs i projectes similars . . . . .	13
<b>3 Metodologia de treball</b>	<b>15</b>
3.1 Pla de treball . . . . .	16
3.2 Calendari . . . . .	18
3.2.1 Diagrama de Gantt . . . . .	19

---

<b>4</b>	<b>Detalls de la proposta</b>	<b>21</b>
4.1	Hardware	22
4.1.1	Sensor VL53L0X	24
4.1.2	Ports de la placa	26
4.2	Programació del prototip	28
4.2.1	Arduino IDE	28
4.3	MQTT	30
4.4	Node-RED	31
4.5	Base de Dades	32
4.6	Funcionament del Sketch	33
4.7	Codi font	34
<b>5</b>	<b>Resultats i discussió</b>	<b>35</b>
5.1	Conceptes previs	35
5.1.1	Exposició del mètode utilitzat	35
5.2	Anàlisi dels resultats obtinguts	36
5.2.1	Conclusió de l'experiment	37
5.3	Despesa econòmica i hores emprades	37
<b>6</b>	<b>Conclusions</b>	<b>39</b>
6.1	Conclusions i visió personal	39
6.2	Continuïtat del projecte	40
	<b>Bibliografia</b>	<b>42</b>
<b>A</b>	<b>Instal·lació del software necessari per el desenvolupament del prototip</b>	<b>45</b>
A.1	Arduino IDE	46
A.2	Eclipse Mosquitto	47
A.3	Node-RED	48
A.4	MySQL	51

# Índex de figures

2.1	Sensor de microones a la dreta i sensor d'ultrasons a l'esquerra . . . . .	8
2.2	Sensor reflectiu . . . . .	9
2.3	Sensor passiu PIR . . . . .	9
2.4	Torn de control d'accés . . . . .	10
2.5	Catifa de pressió . . . . .	11
2.6	Càmera tèrmica . . . . .	11
2.7	Lector RFID . . . . .	12
3.1	Model prototipat . . . . .	15
3.2	Diagrama de flux . . . . .	17
3.3	Diagrama de Gantt . . . . .	19
4.1	Flux del prototip final . . . . .	21
4.2	Placa electrònica HiLetgo UNO R3 D1 R32 ESP32 i sensor ToF model VL53L0X	22
4.3	Esquema de connexions del prototip amb placa Arduino UNO . . . . .	23
4.4	Esquema del funcionament del ToF . . . . .	24
4.5	Esquema protocol I2C . . . . .	25
4.6	Sensor ToF VL53L0X . . . . .	26
4.7	Esquema de la distribució dels pins a la placa . . . . .	27
4.8	Flux del prototip a Node-RED . . . . .	31
4.9	Dashboard Node-RED . . . . .	32
4.10	Digrama Base de Dades . . . . .	33
A.1	Visualització de la pantalla d'Arduino IDE . . . . .	46
A.2	Servei Mosquitto MQTT actiu . . . . .	47
A.3	Execució via Terminal de Node-RED . . . . .	49
A.4	Vista de l'aplicació Node-RED . . . . .	49
A.5	Instal·lació de mòduls addicionals de Node-RED . . . . .	50
A.6	Dashboard de Node-RED . . . . .	50

A.7 Panell de control de MySQL . . . . . 51



# Índex de taules

3.1	Recerca i anàlisi . . . . .	18
3.2	Disseny . . . . .	18
3.3	Desenvolupament del producte . . . . .	18
3.4	Aplicacions del producte . . . . .	19
4.1	Tòpics MQTT . . . . .	30
4.2	Sentències SQL . . . . .	33
4.3	Codi font del prototip: GitHub . . . . .	34
5.1	Despesa econòmica en hardware . . . . .	37



# Capítol 1

## Introducció

En el següent text s'exposa el projecte que s'ha dut a terme per dissenyar i crear un prototip on l'objectiu principal és donar solució per controlar l'aforament de persones dins d'un espai tancat de manera automatitzada. En el context de pandèmia mundial donada pel coronavirus que estem vivint actualment hi ha la necessitat de tenir un control de les persones que hi ha a l'interior d'un espai tancat.

La solució a aquest problema s'ha realitzat mitjançant una placa electrònica de codi obert model *HiLetgo UNO R3 D1 R32 ESP32*, la qual ha sigut programada mitjançant la plataforma de creació Arduino IDE<sup>1</sup>. A la placa hi ha connectats els sensors necessaris per poder comptabilitzar les persones que entren i surten d'una sala, permetent veure en una aplicació web l'estat actual de la sala i informant mitjançant colors semafòrics de la capacitat actual. Al mateix temps cada acció que els sensors detectin quedarà registrat a una base de dades.

Aquest treball s'enfoca en l'àmbit hospitalari, ja que es idoni comptar amb una solució com aquesta que permet comptabilitzar les persones en un espai tancat, on es imprescindible que es pugui seguir accedint en qualsevol cas. Donada l'experiència que dispo en aquest àmbit, ja que es el entorn de treball actual des de fa més de deu anys, i després de parlar amb diferents responsables del hospital comarcal s'ha decidit conjuntament amb ells que val la pena elaborar aquest treball.

Fins el moment, hi ha varies eines que implementen el descrit anteriorment amb diferents tipus de hardware com càmeres, però el que s'ha volgut fer en aquest treball és simplificar-ho al màxim per tal de donar una solució que sigui econòmica i fàcil de mantenir. D'aquesta manera s'ha obtingut un prototip capaç d'assolir els objectius descrits.

---

<sup>1</sup>Entorn de desenvolupament que s'usa per escriure i carregar programes a les plaques electròniques compatibles. Admet llenguatges C i C++.

## 1.1 Temàtica del projecte

Fa anys que treballo en un Hospital comarcal i quan va arribar l'hora d'escollir quin projecte final de màster realitzar es van analitzar propostes que puguin ser aplicables en el meu àmbit de treball i després de vàries reunions amb la direcció es va arribar a la conclusió que el treball es podria encarar en el control d'aforament a les sales d'espera de dins de l'Hospital. Donada la situació d'alerta sanitària produïda pel coronavirus la limitació de l'aforament és clau per evitar contagis.

Un cop decidit com enfocar el treball, es va haver d'encaixar-lo dins de les categories definides a la UOC<sup>2</sup>. La temàtica del treball s'inclou dins de l'àrea de **Serveis basats en localització i espais intel·ligents**. Al contactar amb el professorat responsable de la categoria i al plantejar la proposta s'ha anat perfilant i finalment el títol del treball, és clar, i defineix exactament el que es vol aconseguir: Control d'aforament en un espai tancat.

Arribats en aquest punt ja estava clar el projecte a realitzar i el títol, però el més complicat va ser decidir com desenvolupar-lo ja que no es disposa d'experiència prèvia en el software ni hardware utilitzats. Això ha suposat un repte que s'ha hagut de superar a base de l'autoaprenentatge, aplicant els coneixements previs d'altres àmbits i l'enginy.

## 1.2 Competències adquirides

En aquest Treball Final de Màster s'han aplicat i desenvolupat les competències adquirides durant tot el programa formatiu realitzat en el màster, i on queda palès, l'assoliment dels objectius generals d'aprenentatge. També ha jugat un paper important l'àmbit on em desenvolupo dia a dia professionalment que ha permès emmarcar i definir un problema real per poder donar una solució realitzant un disseny capaç d'executar un pilot que permetrà fonamentar l'objecte de l'estudi que es desenvolupa en aquest treball.

## 1.3 Descripció de la problemàtica

La situació d'alerta sanitària que s'està vivint actualment en tot el planeta per culpa del COVID-19 fa que s'hagi de plantejar solucions eficaçies per tal de que el virus no segueixi propagant-se i a la vegada poder seguir realitzant amb la màxima normalitat possible les tasques diàries i socials de la nostra vida quotidiana. Segons l'Organització Mundial de la Salut (OMS), una de les mesures de seguretat de protecció bàsiques contra la COVID-19<sup>3</sup> és mantenir la

---

<sup>2</sup>Universitat Oberta de Catalunya

<sup>3</sup>Malaltia infecciosa causada pel virus SARS-CoV2.

distància social d'almenys un metre. De la mateixa manera, el ministeri de sanitat de l'Estat Espanyol recomana fixar una distància de seguretat interpersonal de dos metres i usar mascareta higiènica. Aquesta mesura és fàcil de complir quan es passeja per l'aire lliure però no resulta tant assequible en espais tancats. Per tant, un element clau per aconseguir aquesta distància és establir una limitació d'aforament en els espais tancats.

Per tot això, en aquest treball es realitza un mecanisme capaç de controlar l'aforament de persones que hi ha en una sala d'espera d'un centre sanitari, tenint en compte l'entrada i la sortida de persones i en el cas de que s'arribi al aforament màxim permès, s'informi mitjançant una alerta que els usuaris interpretaran de manera clara arribant a la conclusió de que ningú més pot accedir a la sala.

## 1.4 Objectius

Els objectius definits en aquest treball els podem dividir en dos: objectiu principal i objectius secundaris. Al final del treball analitzarem com i amb quin percentatge s'han assolit els objectius.

### 1.4.1 Objectiu principal

El principal objectiu d'aquest treball és desenvolupar un prototip capaç de comptabilitzar les persones que entren i surten d'un espai tancat, permetent així el control de l'aforament de manera automàtica. D'aquesta manera volem convertir una sala d'espera en un espai intel·ligent sense haver de dependre de cap individu per tal de que faci aquest control.

### 1.4.2 Objectius secundaris

Els objectius secundaris formen part de l'objectiu principal i ens ajudaran a aconseguir-lo fent-lo més fort i evident. Els objectius secundaris definits són els següents:

- Emetre una alerta quan s'arribi a l'aforament màxim, permetent interpretar als usuaris que fan ús de l'espai que la sala està complerta. D'aquesta manera l'usuari que vulgui accedir sabrà que haurà d'esperar que un altre abandoni la sala.
- Enregistrar les dades de l'aforament per poder estudiar-les i/o analitzar-les posteriorment
- Instal·lació i posta en producció del prototip en un espai real i que funcioni de la manera esperada.



# Capítol 2

## Antecedents

Des de fa molt temps el control d'aforament[9] és una eina fonamental per diferents raons:

- **Per seguretat**, permetent tenir un control de les persones que hi ha en un espai en concret.
- **Per traçabilitat**, permetent al organitzador o empresa conèixer quins són els pics d'aforament màxims i mínims i així poder prendre decisions tant a temps real com a posteriori per a millorar.
- **Per normativa**, hi ha lleis que regulen els sistemes de control d'aforament i en segons quins àmbits com els concerts i grans esdeveniments aquests sistemes han d'estar homologats segons l'ordre ITC/378/2006<sup>1</sup>

Les principals eines de control de persones són:

- **Clicker electrònic** capaç de mesurar l'aforament en funció de les vegades que un operari polsa el botó tant en l'entrada com la sortida de persones.
- **Dispositius mecànics** els quals no precisen de cap tipus d'intervenció humana ja que per ells sols s'encarreguen de realitzar el comput de persones.
- **Lectors** que són una eina que es basa en la lectura d'identificadors unipersonals, com ara polseres. Aquest sistema té un encert del 100% i a més permet conèixer la persona concreta que entra o surt del espai.

---

<sup>1</sup>Ordre que té com objecte regular el control metrològic del Estat sobre els sistemes de control d'afuència de persones en els locals de pública concurrència, dotant a l'administració pública vetllar pel compliment de la seguretat en espectacles i activitats recreatives.

## 2.1 Detecció de persones

Dins del camp de la seguretat i vigilància existeixen moltes solucions diferents i molts tipus de sensors[8] que són capaços de detectar persones i ajudar-nos a comptabilitzar-les.

### 2.1.1 Sensors

Dins dels sensors per a la detecció de persones els que s'utilitzen de manera més freqüent són els següents:

#### 2.1.1.1 Sensor actiu

Existeixen varis tipus de sensors actius depenent del tipus de senyals que utilitzin, però el funcionament de tot aquest grup és el mateix. El sensor envia una senyal que ha de rebotar en una superfície a una distància coneguda, si alguna cosa interfereix en l'enviament d'aquesta senyal el sensor detecta un canvi en el temps de retorn (ToF<sup>2</sup>) de la senyal i l'avís o alarma s'activa. Les senyals més utilitzades són les microones i els ultrasons.



Figura 2.1: Sensor de microones a la dreta i sensor d'ultrasons a l'esquerra

També existeixen els sensors reflectius, que utilitzen feixos de llum amb un dispositiu emissor i un receptor, si aquest feix de llum és entravessat per algun objecte s'activa la senyal d'alarma.

---

<sup>2</sup>Eina que serveix per calcular el "Temps de Vol" permetent mesurar la distància entre el sensor i un objecte sòlid.





Figura 2.2: Sensor reflectiu

### 2.1.1.2 Sensor passiu

Els sensors passius són els més comuns en els sistemes de seguretat domèstics. Aquests sistemes també denominats PIR<sup>3</sup>, en anglès *Passive InfraRed*, funcionen detectant canvis de moviment o calor a través d'infrarojos.



Figura 2.3: Sensor passiu PIR

<sup>3</sup>Sensor infraroig passiu, és un sensor capaç de mesurar la llum infraroja radiada dels objectes situats en el seu camp de visió.

Aquests sensors són anomenats passius perquè no emeten radiacions, sinó que les rep. Estan compostats d'un component electrònic dissenyat per detectar canvis en la radiació infraroja rebuda. Generalment incorporen un transistor que amplifica la senyal elèctrica que genera quan es produeix aquesta variació de radiació rebuda.

### 2.1.1.3 Sistemes mixtes

Molts sistemes de control i detecció de persones combinen l'ús dels sensors actius i passius per així obtenir una major fiabilitat i estalvi energètic. També existeixen sistemes més complexos que incorporen càmeres per així augmentar encara més la fiabilitat.

## 2.1.2 Altres sistemes de control

Un sistema a tenir present seria els **torns d'accés**, com els que hi ha als centres esportius o parcs d'atraccions, que neguen o permeten l'accés en el pas d'una persona. Si bé podrien realitzar el control d'aforament resulta un sistema invasiu i amb un cost molt elevat a més s'han de fixar al terra per assegurar l'estabilitat i solen ser aparells molt voluminosos.



Figura 2.4: Torn de control d'accés

Per la definició d'aquest dispositiu de control i per la norma dels hospitals de que són recintes oberts aquesta sol·lució queda descartada d'entrada.

Les **catifes de pressió** permeten identificar el número de persones que el traspassen a través d'una sèrie de sensors que porta incorporada la mateixa catifa. Aquesta tecnologia es bastant recent i el seu cost és força elevat comparant-lo amb altres tecnologies.



Figura 2.5: Catifa de pressió

Les **càmeres tèrmiques** estan destinades a comptar el número de persones que passen per un determinat lloc mitjançant el software corresponent. Aquest tipus de càmeres treballen independentment de les condicions lluminoses. Al detectar les persones únicament per la imatge tèrmica es preserva la seva intimitat i privacitat. Aquests sistemes es poden instal·lar al sostre de l'edifici com si fossin detectors de fum.



Figura 2.6: Càmera tèrmica

Els **lectors RFID**[6] o la identificació per radiofreqüència es un sistema d'emmagatzement i recuperació de dades remotes que utilitza dispositius d'etiquetes o tarjetes. El propòsit fonamental és transmetre la identitat de un objecte. Aquest sistema podria servir per comptabilitzar les persones i a més identificar-les però s'hauria de subministrar una etiqueta o tarjeta RFID a totes les persones que hagin d'accedir a la sala.

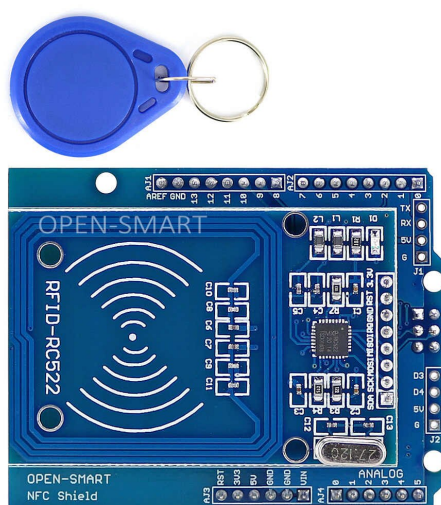


Figura 2.7: Lector RFID

Amb tot el descrit anteriorment es conclou que en el moment actual que s'està vivint, les persones de peu, per culpa de l'emergència sanitària provocada pel coronavirus[15], són conscients de que aquests sistemes existeixen, i s'han d'aplicar i complir en espais reduïts i sobretot en espais tancats on no circula l'aire amb freqüència. Per això actualment existeixen moltes empreses que donen múltiples solucions de control d'aforament amb diferents hardwares: sensors de presència, sensors tèrmics, càmeres[7], entre d'altres sistemes.

En un Hospital Comarcal, aquesta necessitat[14] es veu agreujada i en ocasions està suposant un problema ja que no hi ha personal suficient per a controlar qui entra o no en un espai en concret. Tot i que en les sales d'espera actualment hi ha menys cadires disponibles si els pacients tenen l'accés permès aquests s'asseuen o s'esperen on poden, fent que no s'acabin de complir les distàncies de seguretat recomanades per frenar la propagació del virus. Donada aquesta problemàtica, es vol crear un prototip que permeti saber les persones que hi ha dins d'una sala i mitjançant una alerta avisi els pacients que volen accedir i fer-los saber si l'entrada és permesa o no. Donada la gran oferta d'aquest tipus de servei, en aquest treball s'ha volgut realitzar un prototip reduint els costos a la mínima expressió obtenint resultats satisfactoris.

### 2.1.3 Treballs i projectes similars

En aquest apartat es vol fer menció a altres treballs, projectes i articles que han servit tant d'inspiració com per completar i donar forma aquest TFM.

#### 2.1.3.1 Desenvolupament i experimentació d'un programa de monitorització i control mitjançant visió artificial

[8]Es tracta d'un treball on es dissenya un programa de seguretat i vigilància utilitzant la visió artificial. El programa es capaç de detectar persones, seguir-les i reconèixer i detectar els rostres.

No utilitza cap tipus de hardware ja que únicament desenvolupa un software que rep un vídeo que analitza i fa el seguiment de les persones.

#### 2.1.3.2 Sistema de còmput automàtic de flux de persones mitjançant la visió artificial

[7]Es tracta d'un sistema de control d'aforament pensat per a llocs de caràcter públic. S'implementen dos algorismes per identificació de vianants capaç de comptabilitzar-los en un entorn real controlat. Es un treball que s'enfoca en l'anàlisi d'imatges pel monitoratge i control de l'àrea facial.

#### 2.1.3.3 La importància del control d'aforament en espectacles

[4]Es tracta d'un article realitzat pel director d'una empresa de seguretat on s'expliquen les diferents mesures de seguretat que es poden aplicar es espectacles: **Preventives**, com filtres en l'accés, registre individualitzat del públic, control previ i final del esdeveniment i les mesures **reactives** com els sistemes de vídeo vigilància, polseres RFID, equips d'escàner... Es tracta d'un article interessant el qual ha servit per entendre millor la gestió de persones en els espais públics i donar forma aquest TFM.

#### 2.1.3.4 Control d'aforament mitjançant RFID

[6]Es tracta d'un treball que vol proporcionar un sistema de control i identificació de persones en un esdeveniment de grans dimensions. Pel control d'accés dels usuaris s'utilitza un lector capaç d'identificar polseres personals RFID que portaran els assistents i es podrà identificar tant l'entrada, la sortida i l'assistent. També es desenvolupa un aplicació per smartphones que permetran als usuaris veure l'aforament en temps real.

#### **2.1.3.5 Control d'aforament, pes i altura en un espai tancat**

[12]Es planteja la implementació d'un sistema de control per gestionar l'aforament de persones en un local, tenint en compte els paràmetres de mesura de l'individu: pes i altura així també com el número d'individus que hi ha al recinte. Aquest projecte utilitza sensors de pes i d'altura que aniran connectats a una placa de circuit integrat amb un microcontrolador.

# Capítol 3

## Metodologia de treball

En aquest capítol és descriu i s'explica el procés, la metodologia el pla de treball i el calendari utilitzats per resoldre el problema de comptabilitzar les persones en un espai tancat.

La metodologia emprada és la de model de prototip[11], en enginyeria del software, pertany als models de desenvolupament evolutiu i va ser proposat i considerat un mètode formal per Boehm<sup>1</sup> l'any 1988. Aquesta metodologia condueix a la construcció d'un prototip en poc temps, usant el software i hardware adequats i utilitzant pocs recursos. El resultat serà avaluat pel client, obtenint d'aquest els requisits del software perfilats. D'aquesta manera el client veurà els resultats a curt termini i fent que l'àrea de desenvolupament entengui millor el que ha de fer.

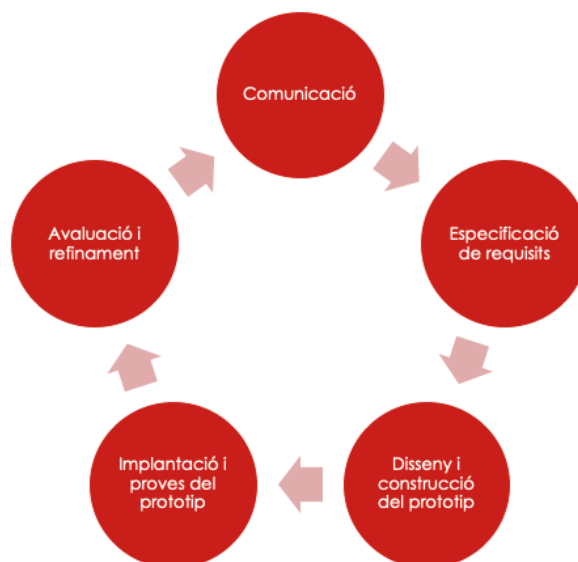


Figura 3.1: Model prototipat

---

<sup>1</sup>Enginyer informàtic americà i també professor emèrit en el departament de ciències tecnològiques en la Universitat del Sud de Califòrnia. És conegut per les múltiples aportacions en el camp de l'enginyeria del software.

S'ha decidit utilitzar aquesta metodologia ja que des de l'inici es coneix l'objectiu principal, on la finalitat és construir un sistema que compleixi la funció desitjada. També s'ha acordat amb l'hospital les regles de joc on queda clar que el que s'està desenvolupant és un prototip que compleixi els objectius proposats, i tot i que es vol desenvolupar quelcom proper al producte real, al finalitzar aquest treball s'haurà de desenvolupar un software i un hardware real enfocat a la qualitat i en l'estètica.

Dit això és decideix dur a terme un **prototipat reutilitzable (reusable prototyping)** on no es vol perdre l'esforç efectuat en la construcció del prototip ja que les seves parts o el conjunt de tots elles poden ser reutilitzades per construir el producte real. A continuació es detallen es cinc fases per les quals passa el model prototipat que apareixen a la figura 3.1:

- **Comunicació:** Identificar el client i conèixer-lo identificant les fortaleses, debilitats, necessitats, disponibilitat i responsabilitats.
- **Especificació de requisits:** Recopilar els requeriments reals i determinar la importància de cadascun.
- **Disseny i construcció del prototip:** Centrat en els aspectes visibles per l'usuari: interfície d'usuari, entrades de dades...
- **Implantació o proves:** Implantar el prototip en un entorn de test i comprovar que el prototip es comporta de la manera esperada.
- **Avaluació i refinament:** Verificació dels requisits definits i modificar-lo en cas de que sigui necessari.

## 3.1 Pla de treball

El pla de treball establert està dividit en cinc fases:

1. **Definició de les especificacions:** En aquesta primera fase l'objectiu principal és auditar la informació relativa al problema amb la finalitat d'obtenir totes les dades necessàries per a la resolució del control d'aforament en una sala d'espera d'un hospital. En aquesta fase també s'estudiarà i s'investigarà sobre la detecció de persones mitjançant plaques electròniques compatibles amb la plataforma de desenvolupament Arduino IDE.
2. **Disseny conceptual:** L'objectiu d'aquesta fase es construir un model d'informació que reflecteixi l'esquema conceptual del prototip. A la figura 3.2 podem veure el diagrama de flux del projecte.



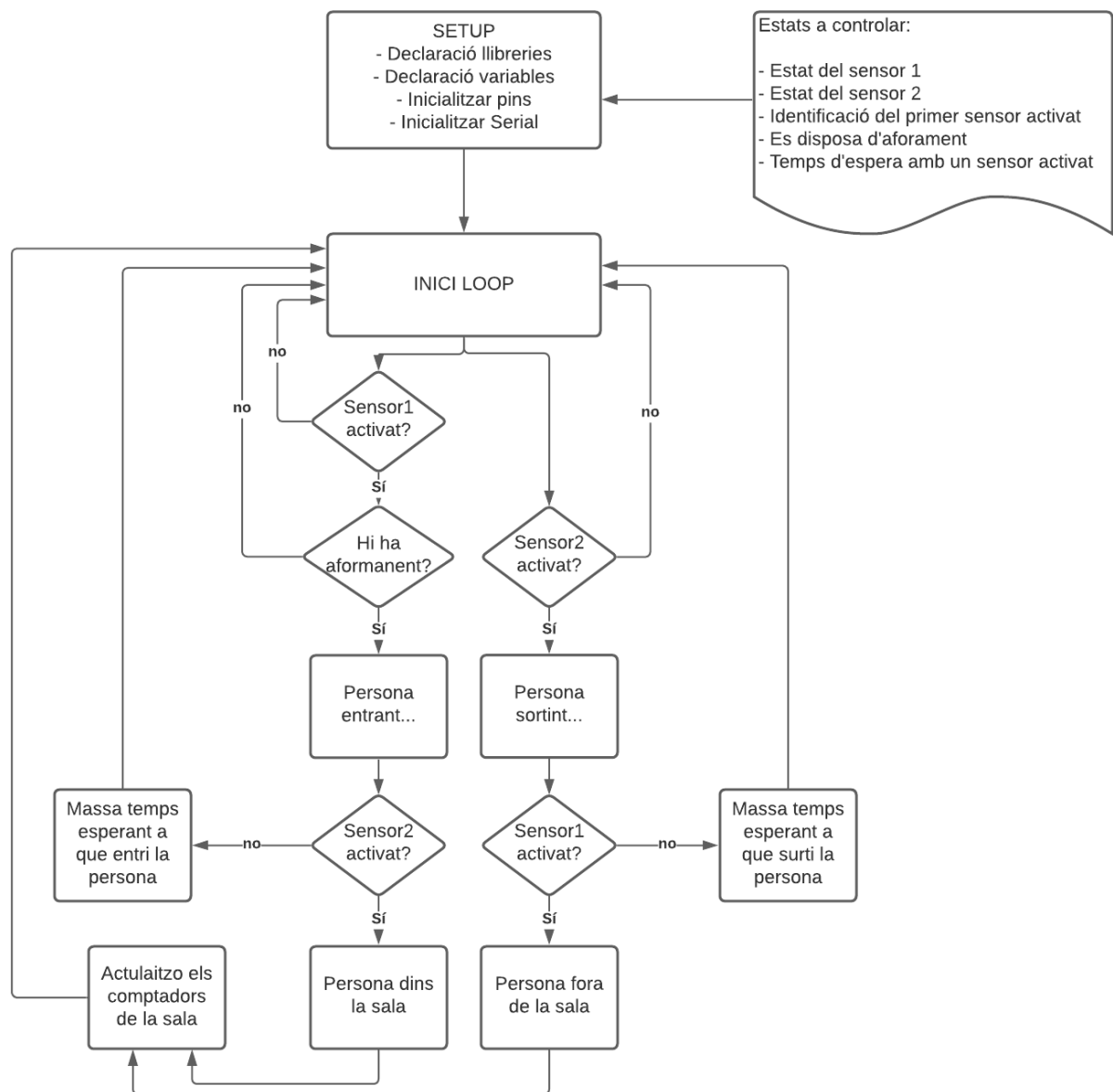


Figura 3.2: Diagrama de flux

- Desenvolupament del prototip:** Aquesta fase té com objectiu la construcció del prototip i que sigui operatiu per a la seva aplicació. S'expliquen tots els detalls en el Capítol 4.
- Proves:** En aquesta fase es realitzaran totes les proves necessàries per validar el prototip desenvolupat en la fase anterior. Si el resultat d'aquestes proves fa que es detecti la necessitat de modificar el prototip per a corregir defectes o afegir alguna funcionalitat es tornarà a la fase anterior i es realitzaran les iteracions necessàries fins arribar al prototip

esperat. Aquesta fase està desenvolupada en el Capítol 5.

5. **Auditoria i seguiment:** En aquesta fase es realitzarà un seguiment del rendiment del prototip i es calcularà la fiabilitat d'aquest podent així determinar les modificacions necessàries si s'escauen. Aquesta fase també queda reflectida en el Capítol 5.

## 3.2 Calendari

En les següents taules es mostra una descripció de cadascuna de les tasques definides inicialment per a dur a terme el projecte. Apareix la data d'inici i de finalització i l'output que s'obté.

Name	Start	End	Codi	Output
Recerca de solucions de control d'aforament	14/9/2020	18/9/2020	REC-1	Projecte
Definició del TFM	21/9/2020	23/9/2020	REC-2	Memòria
Redacció del Pla de Treball	23/9/2020	29/9/2020	REC-3	Memòria
Recerca de la solució hardware	29/9/2020	2/10/2020	REC-4	Projecte
Anàlisi de resultats desats a la base de dades	9/11/2020	11/11/2020	REC-5	Memòria
Joc de proves	16/11/2020	18/11/2020	REC-6	Memòria
Presentació virtual	23/11/2020	27/11/2020	REC-7	Video
Redacció de les conclusions	7/12/2020	11/12/2020	REC-8	Memòria

Taula 3.1: Recerca i anàlisi

Name	Start	End	Codi	Output
Definició de l'arquitectura	12/10/2020	14/10/2020	DIS-1	Memòria
Hardware: configuració	5/10/2020	9/10/2020	DIS-2	Projecte
Disseny de la base de Dades	14/10/2020	16/10/2020	DIS-3	Memòria

Taula 3.2: Disseny

Name	Start	End	Codi	Output
Detecció de les persones	19/10/2020	23/10/2020	DES-1	Projecte
Comptar persones entrants i sortints	26/10/2020	30/10/2020	DES-2	Memòria
Emmagatzematge de les dades en BBDD	2/11/2020	6/11/2020	DES-3	Memòria
Gestió de les alertes si es supera l'aforament	11/11/2020	13/11/2020	DES-4	Projecte
Estadístiques amb les dades obtingudes	30/11/2020	4/12/2020	DES-5	Memòria

Taula 3.3: Desenvolupament del producte

Name	Start	End	Codi	Output
Instal·lació del sistema en una sala d'espera real	18/11/2020	23/11/2020	DES-6	Memòria

Taula 3.4: Aplicacions del producte

### 3.2.1 Diagrama de Gantt

A continuació es detallen les tasques en el diagrama de Gantt on es pot veure el temps de dedicació previst per cadascuna d'elles:

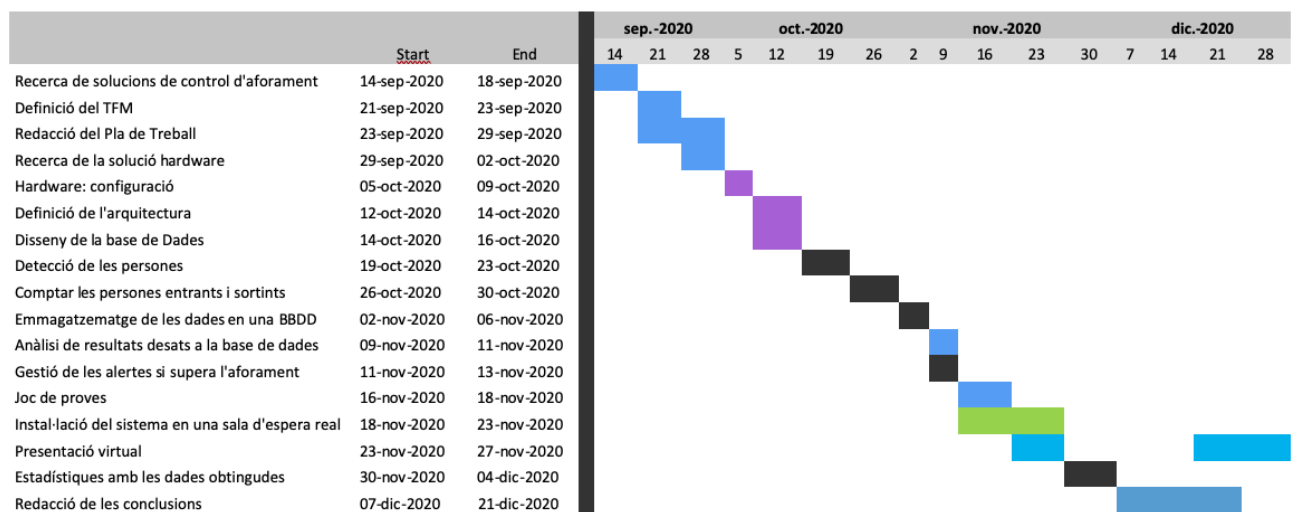


Figura 3.3: Diagrama de Gantt



# Capítol 4

## Detalls de la proposta

En aquest capítol s'explica en detall el software i hardware utilitzats per a la realització del projecte. Pel funcionament del prototip desenvolupat en aquest TFM intervenen quatre components: placa electrònica, servidor MQTT, servidor Node-RED i Base de dades.

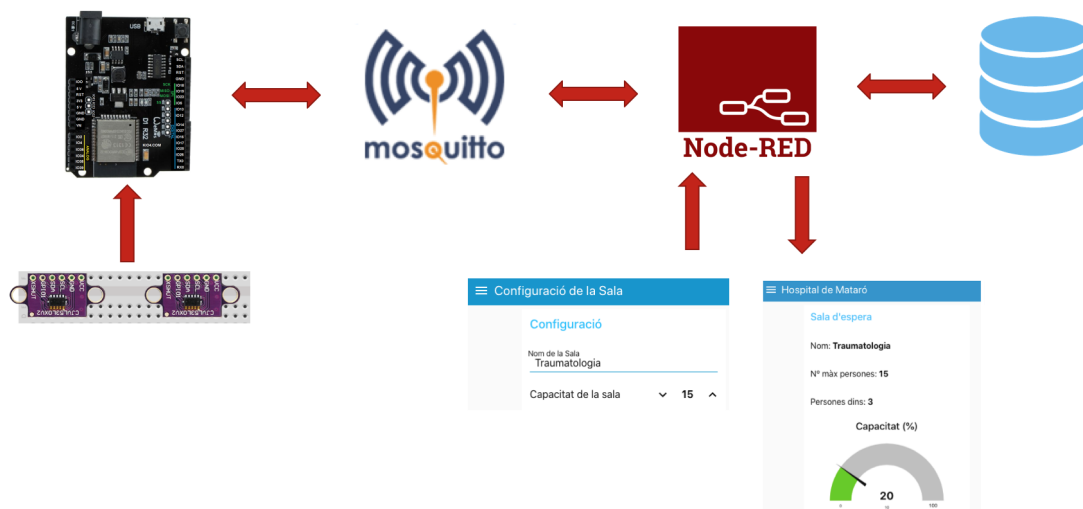


Figura 4.1: Flux del prototip final

Els sensors connectats a la placa informen de si entra o surt alguna persona de la sala. La placa envia al servidor MQTT, en endavant broker<sup>1</sup>, les persones que hi ha dins la sala cada vegada que entra o surt una persona i el broker comunica a la placa l'aforament màxim de la sala, que aquest valor el servidor Node-RED el recupera de la BBDD i també es pot modificar a través del panell de configuració de la interfície gràfica. Finalment, Node-RED mostra per la interfície gràfica, en endavant, dashboard la informació que li envia el broker i l'emmagatzema a la BBDD.

<sup>1</sup>És el programa que s'encarrega de rebre els missatges enviats pels clients i distribuir-los entre sí en un sistema pub-sub.

## 4.1 Hardware

Per crear el prototip s'ha apostat per utilitzar una placa electrònica model HiLetgo UNO R3 D1 R32 ESP32<sup>2</sup> i s'hi han connectat dos sensors làser ToF model VL53L0X<sup>3</sup>.



Figura 4.2: Placa electrònica HiLetgo UNO R3 D1 R32 ESP32 i sensor ToF model VL53L0X

Per arribar fins aquí es va haver de fer recerca[3] del funcionament de les plaques Arduino i fer diverses proves amb plaques diferents (Arduino Mega, Arduino UNO, ESP32 DevKitC V2, NodeMCU Lolin V3 i per últim HiLetgo UNO R3 D1 R32 ESP32) i sensors (ultrasons i làser). El primer prototip es va realitzar amb sensors d'ultrasons connectats a una placa Arduino UNO<sup>4</sup>. Aquest prototip donava solució al problema, però en un ambient sorollós aquests tipus de sensors donen falsos positius i falsos negatius fent molt difícil de controlar i augmentant considerablement el marge d'error i reduint la fiabilitat. Per aquest motiu finalment s'implementa la solució amb els sensors ToF làser com es veu a la figura 4.3.

<sup>2</sup>Placa electrònica amb processador ESP32 que compta amb WiFi i Bluetooth i és compatible amb Arduino IDE.

<sup>3</sup>Sensor actiu infraroig làser que permet mesurar distàncies de forma precisa.

<sup>4</sup>Placa de microcontrolador que disposa de 14 pins d'entrada i sortida i 6 entrades analògiques.

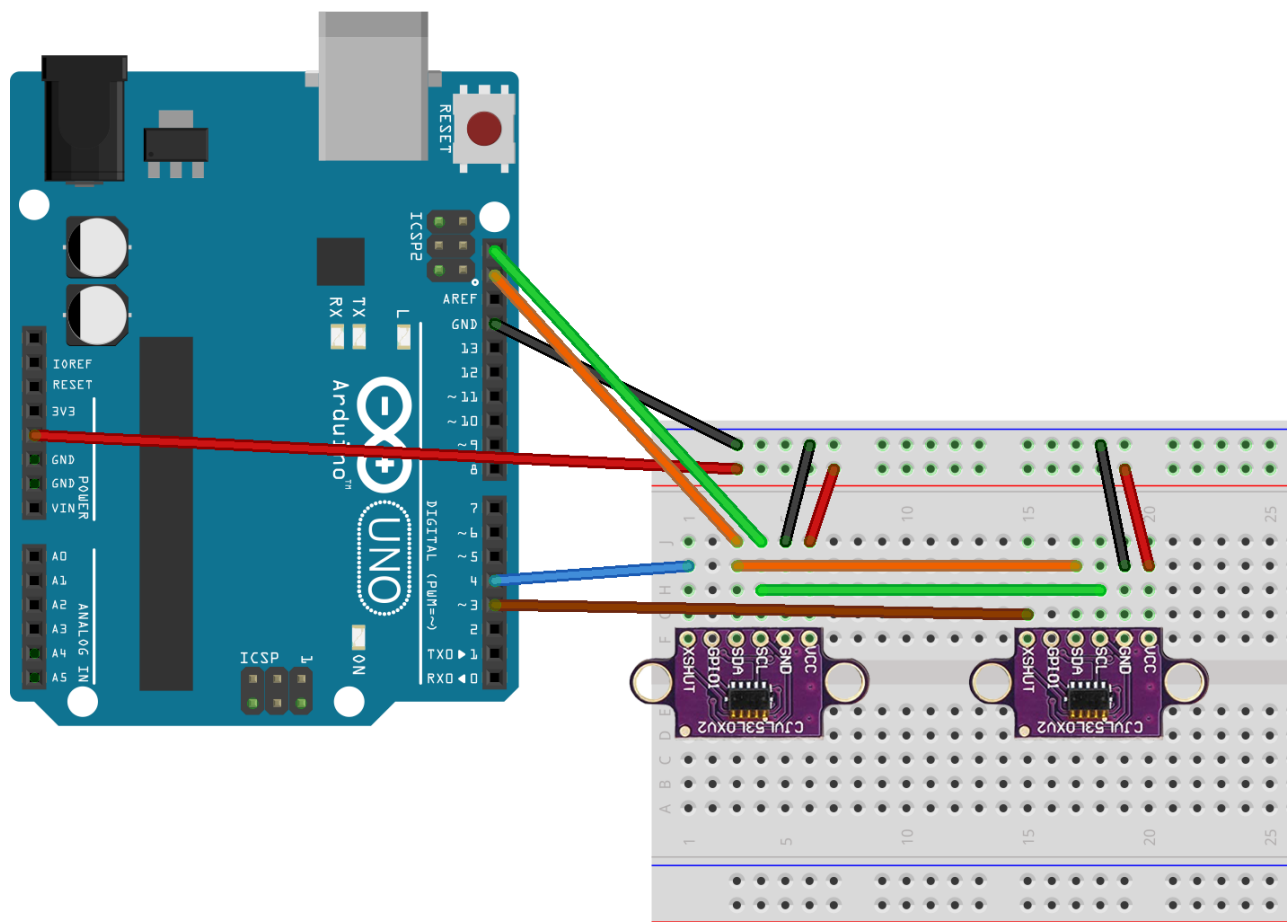


Figura 4.3: Esquema de connexions del prototip amb placa Arduino UNO

Totes les plaques electròniques testejades funcionen de manera molt semblant i donen un resultat satisfactori excepte ESP32 DevKitC V2 i NodeMCU Lolin V3 que no permeten connectar dos sensors ToF a la vegada, a més, la problemàtica afegida al treball es que el sistema ha de comunicar-se amb l'exterior i la manera com és fa, es mitjançant WiFi, per això finalment la placa usada és la HiLetgo UNO R3 D1 R32 ESP32 que incorpora tecnologia WiFi.

El funcionament del prototip és el següent: La placa connectada als dos sensors rep la senyal quan aquests se'ls posa quelcom davant. Es disposen de dos sensors perquè així sabem si la persona està entrant o sortint de la sala depenent de quin sensor és el primer a detectar un obstacle. Quan es traspasa el sensor 1 i seguidament el sensor 2 l'usuari ha entrat i si es traspasa el sensor 2 i seguidament el sensor 1 l'usuari ha sortit de la sala.

### 4.1.1 Sensor VL53L0X

El sensor VL53L0X forma part de la nova generació de sensors de distància per temps de vol (ToF: Time of Flight). Ofereix mesures exactes sense importar la superfície reflectant amb un rang de mesures de 50mm fins a 2m.

El principi de funcionament d'aquests sensors és similar al dels sensors d'ultrasons: el sensor posseeix un emissor làser que cada cert temps emet un feix de llum, al trobar-se amb un objecte la llum es reflexa i el sensor es capaç de mesurar el temps entre l'emissió i la detecció de la llum i com que la velocitat de la llum és un valor conegut es pot calcular la distància del objecte amb el temps de vol.

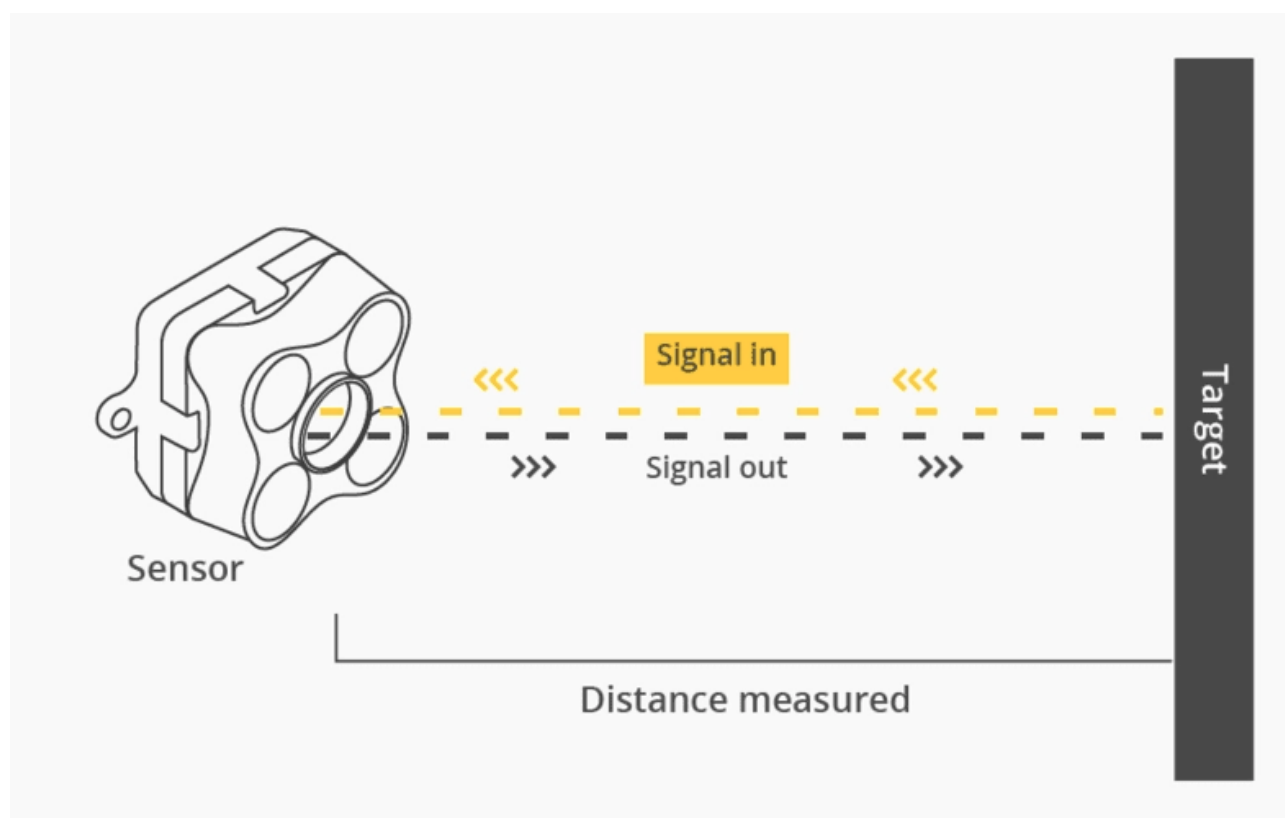


Figura 4.4: Esquema del funcionament del ToF

El xip VL53L0X ha estat dissenyat per funcionar inclús quan la llum ambiental es bastant elevada. Això és important ja que per treballar amb mesures òptiques com major és la contaminació lumínica del ambient, més complicat es captar adequadament el rebot de la senyal.

El sensor VL53L0X[10] és compatible amb I2C[5]. I2C és un protocol síncron i utilitza dos cables, un pel rellotge (SCL<sup>5</sup>) i un altre per les dades (SDA<sup>6</sup>). També és necessària una tercera

<sup>5</sup>System Clock: és la línia de polsos que sincronitzen un sistema.

<sup>6</sup>Serial Data: és la línia per la que es mouen les dades entre dos dispositius.



línia, però aquesta només referència al terra (GND<sup>7</sup>). La primera versió data de l'any 1992 i la versió 2.1 de l'any 2000 i el seu creador va ser Philips. La velocitat és de 100 Kbit/s en el mode estàndard, tot i que permet velocitats de 3.4 Mbit/s. És un bus molt utilitzat en la indústria, principalment per comunicar microcontroladors i els seus perifèrics en sistemes integrats ja que permet definir un perifèric com a **Master** (principal) i  $N$  en **Slave** (esclau).

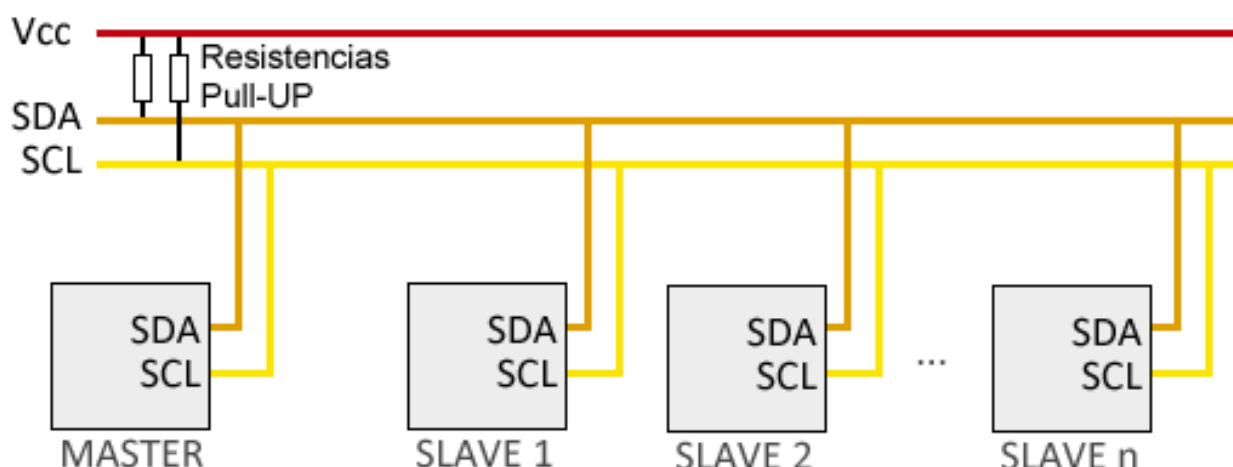


Figura 4.5: Esquema protocol I2C

El sensor adapta les línies I2C al mateix nivell lògic que el voltatge VCC<sup>8</sup> utilitzat, facilitant la connexió de la placa amb voltatges de 3,3V o 5V. Els pins del sensor són els següents:

- **VCC**: alimentació de 2,6V a 5,5V.
- **GND**: Connexió a terra (0V).
- **SDA**: Línia de dades I2C on HIGH es el Voltatge i LOW 0V.
- **SCL**: Línia de rellotge I2C on HIGH es el Voltatge i LOW 0V.
- **XSHUT**: Quan és posa a LOW el sensor es posa en mode d'espera. Aquest pin l'utilitzem per poder connectar més d'un sensor a la mateixa placa.
- **GPIOI**: sense connexió.

<sup>7</sup>Ground, es el punt de referència en un circuit elèctric des del que es mesuren els voltatges.

<sup>8</sup>És el voltatge més alt que GND. VCC és l'entrada d'energia d'un dispositiu.

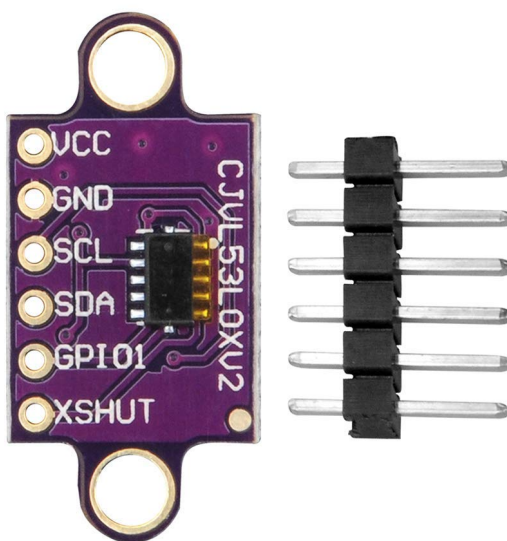


Figura 4.6: Sensor ToF VL53L0X

Encara que sigui invisible, el sensor funciona amb un làser Classe1 i han de prendre's les precaucions necessàries.

#### 4.1.2 Ports de la placa

La placa HiLetgo UNO R3 D1 R32 ESP32 escollida per realitzar aquest projecte, porta incorporat un xip ESP32 i disposa de les següents característiques tècniques:

- Alimentació 5-12V
- Wifi 802.11 b/g/n/e/i fins a 150 Mbps
- Bluetooth v4.2 BR/EDR y BLE
- Compatible amb shields d'Arduino UNO
- 6 entrades analògiques
- 20 entrades/sortides digitals (3.3V)
- Connexió mitjançant micro USB.
- 4Mb memòria flash
- Consum màxim 250mA



## 4.2 Programació del prototip

### 4.2.1 Arduino IDE

Aquest tipus de plaques electròniques es programen mitjançant el software Arduino (IDE) de codi obert que ens facilita l'escriptura del codi de programació i la posterior pujada a la placa electrònica. Aquest software es pot utilitzar amb plaques Arduino genuïnes o amb qualsevol altra placa compatible, com és el cas d'aquest treball. La versió del software Arduino IDE utilitzada per a realitzar el prototip és la 1.8.13 i com totes les altres versions utilitza el llenguatge estàndard C++ adaptat per utilitzar en els microcontroladors d'aquestes plaques electròniques.

Per començar a programar s'ha de preparar Arduino IDE. Primer de tot cal indicar el tipus de placa, en el nostre cas cal descarregar els components d'una placa DOIT ESP32 DEVKIT V1, d'aquesta manera Arduino IDE es podrà comunicar amb la placa electrònica i bolcar el codi de programació.

En segon terme cal incloure les llibreries<sup>9</sup> necessàries per a interactuar amb els sensors i amb la WiFi. En el nostre cas cal incloure la llibreria VL53L0X.h la qual disposa de les funcions necessàries per treballar amb els sensors ToF, la llibreria WiFi.h la qual permet realitzar una connexió a una xarxa que estigui dins de l'abast i finalment PubSubClient.h que ens permetrà connectar la placa al servidor MQTT el qual es detalla més endavant la seva funció.

Un cop definida la placa i les llibreries cal iniciar un projecte o **sketch** el qual genera un fitxer amb l'extensió .ino i pot contenir més d'un fitxer que es imprescindible que estiguin dins del mateix directori pel seu funcionament. L'estructura bàsica d'un sketch es bastant simple i es compon de les següents parts:

- Declaració de variables i llibreries.
- **setup()**, funció utilitzada per inicialitzar les variables, pins de la placa o la càrrega de biblioteques externes. Aquesta funció només s'executarà una vegada després de cada engegada o reinici de la placa.
- **loop()**, va seguida de la funció setup() i aquesta fa exactament el que suggereix el seu nom, realitza una iteració de manera consecutiva, permetent que el programa pugui canviar i respondre a esdeveniments i d'aquesta manera poder controlar activament la placa.

---

<sup>9</sup>Les llibreries són trossos de codi fet per tercers que utilitzem en el projecte o sketch. Ens facilita molt la programació i fa que el programa sigui més senzill de fer i d'entendre.

#### 4.2.1.1 Programació en Temps Real (RTC)

El prototip realitzat en aquest TFM s'inclou en el grup de RTC. Els programes en temps real han de garantir la resposta dins de les limitacions de temps especificades. Un sistema no especificat en temps real normalment no podrà garantir una resposta dins d'un període de temps.

Un sistema en temps real és descrit com aquell que “controla un entorn rebent dades, processant-les i retornant els resultats ràpidament”. Això és el que s'aconsegueix amb Arduino IDE, programació en temps real, rebent contínuament dades dels sensors que s'analitzen i es dona resposta ràpidament.

#### 4.2.1.2 Detall de les variables definides al inici del sketch

A l'inici del projecte o Sketch es defineixen les variables i objectes que seran necessaris pel funcionament del prototip. A continuació es detallen els més importants:

- Llibreries necessàries WiFi.h, Wire.h, VL53L0X.h i PubSubClient.h
- Construcció dels objectes del tipus VL53L0X els quals rebran la informació dels sensors i WifiClient que crearan l'objecte per a connectar-nos a la WiFi.
- També es creen les variables que permetran comptabilitzar les persones que hi ha dins la sala, la capacitat de la sala o el moment quan una persona està entrant o sortint de la sala.

#### 4.2.1.3 Detall de les funcions utilitzades al setup()

- **pinMode(pin, mode)**: Configura el pin especificat perquè es comporti com entrada (*INPUT*) o com sortida (*OUTPUT*).
- **digitalWrite(pin, value)**: Escriu un valor *HIGH* o *LOW* a un pin digital, la tensió serà de 5V o 3,3V per *HIGH* i 0V per *LOW*.
- **delay(ms)**: Posa en pausa el programa durant el temps en mil·lisegons especificat. Molt útil per poder visualitzar i/o donar temps entre una acció i una altra.
- **Serial.print(val)**: Imprimeix el valor al port sèrie com a text ASCII.
- **Serial.begin(speed)**: Estableix la velocitat de dades en bits per segon per a la transmissió de dades en sèrie. En el cas del prototip la velocitat és 115200, que és la que accepta la placa escollida.

- **ConnectWiFi\_STA()**: Realitza les instruccions necessàries per a connectar la placa a la WiFi. Dins d'aquesta funció se'n crida d'altres com **wifi.begin(ssid, password)** on se li ha d'indicar el SSID i la clau de pas de la WiFi.

#### 4.2.1.4 Detall de les funcions utilitzades al loop()

Per tal de que la placa es connecti als sensors s'utilitzen funcions de la llibreria VL53L0X.h:

- **VL53L0X sensor**, crea una instància del tipus VL53L0X que serà el sensor. Aquesta la definim a l'inici del Sketch.
- **sensor.readRangeSingleMillimeters()**, retorna un *long* que és la distància en mil·límetres que llegeix el sensor.

### 4.3 MQTT

La placa es comunica transmetent i rebent dades del broker, amb un servidor MQTT[13]. MQTT és un protocol de transport de missatges client/servidor basat en publicacions i subscripcions denominats tòpics. Això el fa adequat per a missatges d'IoT<sup>10</sup>, com és el cas, la interacció de sensors de baixa potència o plaques amb microcontroladors. En concret s'implementa **Eclipse Mosquitto** que és de codi obert (amb llicència EPL/EDL) i treballa amb les versions 5.0, 3.1.1 i 3.1 del protocol MQTT. En la següent taula s'enumeren els tòpics que s'han definit a la placa, dividits en tòpics de subscripció i tòpics de publicació:

Tòpic	Subscripció	Publicació
hmataro/salaespera1/persones_dins	X	
hmataro/salaespera1/capacitat	X	
hmataro/salaespera1/persones		X
hmataro/salaespera1/persones_percentatge		X

Taula 4.1: Tòpics MQTT

D'aquesta manera el **broker** informará a la placa mitjançant els tòpics a que s'està subscrit de les persones que hi ha dins de la sala i de la capacitat total de la sala i la placa informará al broker, mitjançant els tòpics de publicació, el total de persones que hi ha dins de la sala i del percentatge d'ocupació de la sala.

Per a la connexió amb el broker es fa ús de la funció **client.setServer(server, port)**, que realitza la connexió al **broker** indicant la seva **IP**<sup>11</sup> i el port, que habitualment és el 1883.

<sup>10</sup>Agrupació i interconnexió de dispositius i objectes a través d'una xarxa, on tots ells poden ser visibles i interaccionar.

<sup>11</sup>És un número que identifica inequívocament un dispositiu lògic connectat a la xarxa.

## 4.4 Node-RED

Node-RED[1] és una eina de programació per interconnectar dispositius hardware, API's i serveis en línia. Aquesta eina permet programar de manera visual un flux de treball basat en el navegador web on es poden afegir o eliminar nodes i connectar-los entre ells amb la finalitat de que es comuniquin entre ells en temps real.

Per tal de que tot funcioni, cal crear un flux a Node-RED amb diferents nodes. En el flux hi ha nodes MQTT que permeten establir una connexió amb el Broker gestionant la informació rebuda i enviada, establir una connexió amb la base de dades i mostrar i recollir informació en el dashboard.

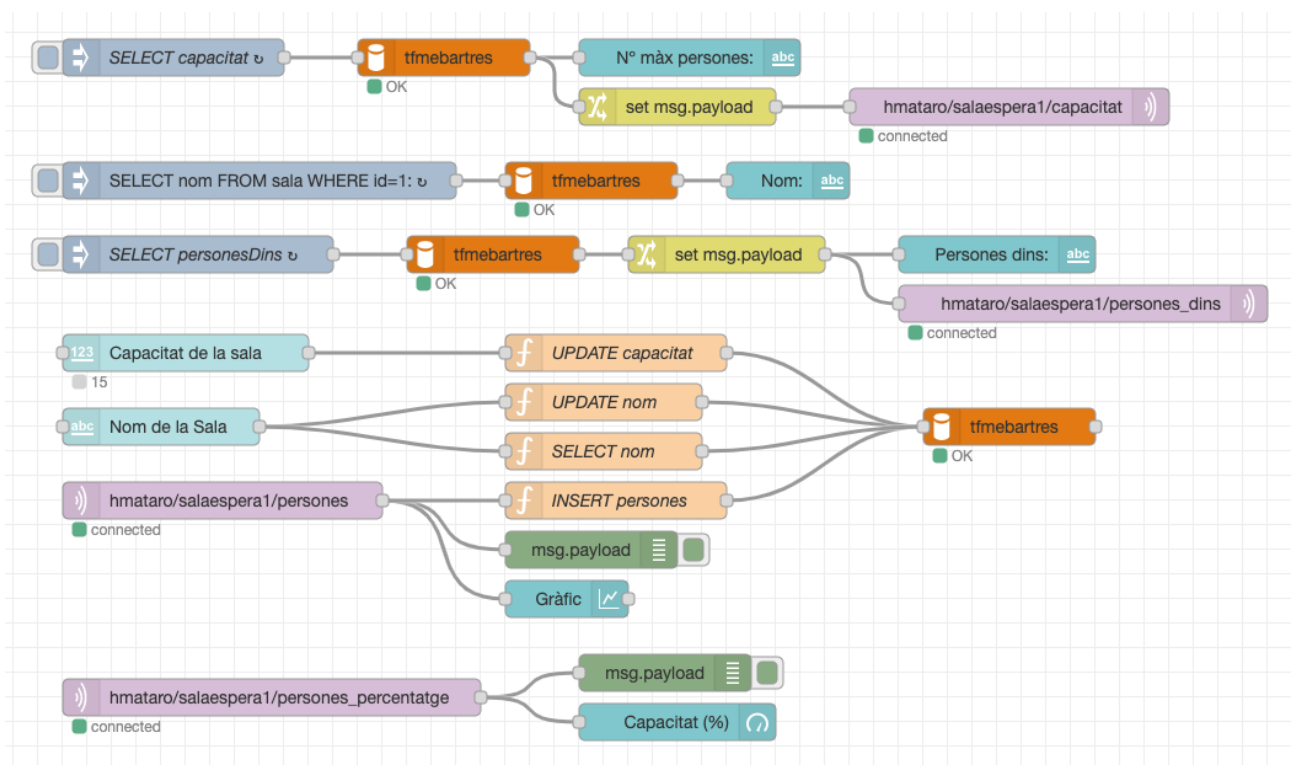


Figura 4.8: Flux del prototip a Node-RED

Quan el Broker informa a través del tòpic, de les persones que hi ha dins de la sala, s'actualitza el comptador i el gràfic del dashboard. El dashboard és la interfície gràfica que també es genera amb Node-RED, tal i com es pot veure a la figura 4.9.

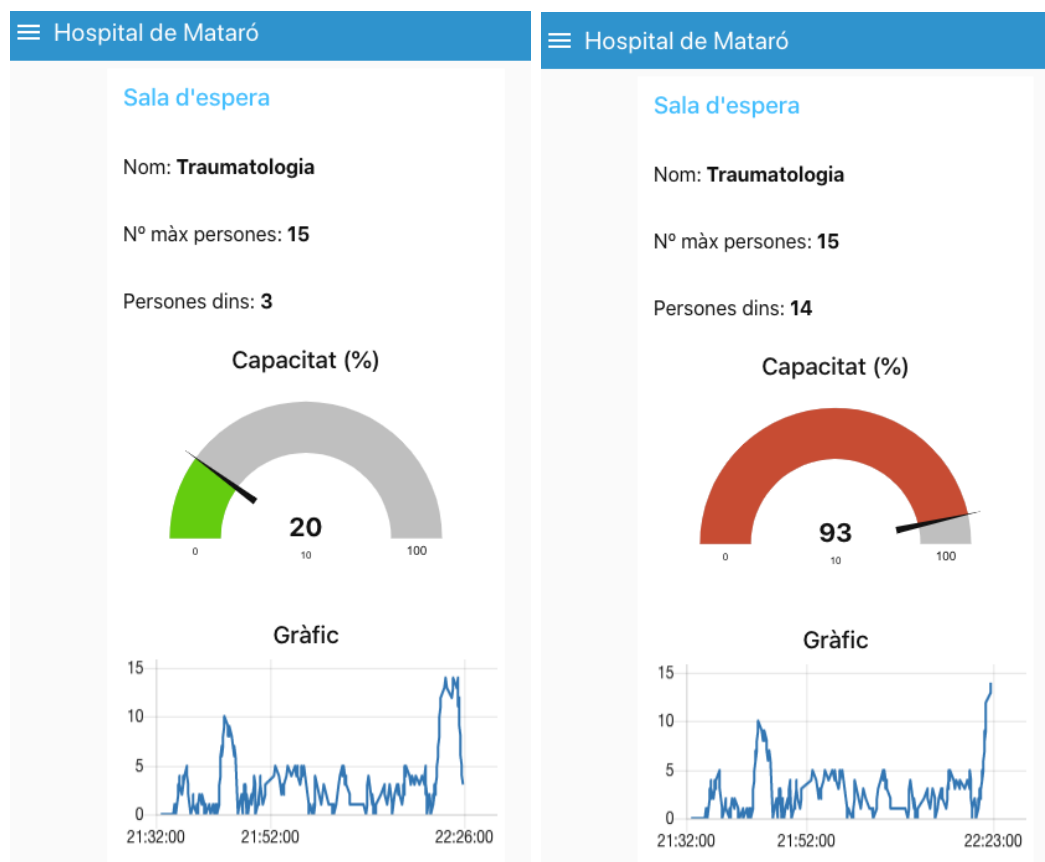


Figura 4.9: Dashboard Node-RED

El dashboard 4.9 mostra al usuari la següent informació: El nom de la Sala, la capacitat total de la sala, la capacitat en temps real (que es mostra en valor absolut i en percentatge) i finalment un gràfic on es veu l'evolució de la sala al llarg del temps.

Finalment, com es pot veure en el flux de la figura 4.8 també hi ha nodes MySQL que són els que inseriran, consultaran o modificaran nous registres a la base de dades.

## 4.5 Base de Dades

Per donar consistència i traçabilitat a tot el flux, s'emmagatzema les dades que recull la placa electrònica a través dels sensors a una base de dades. Es tracta d'una base de dades MySQL[2] on l'estructura és la següent:



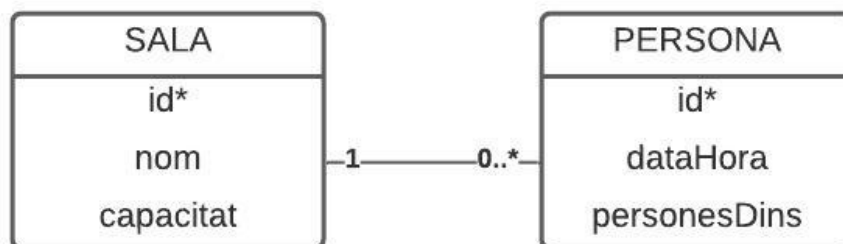


Figura 4.10: Digrama Base de Dades

La base de dades consta de dues taules. La taula **SALA** on s'emmagatzema el *nom* i la *capacitat* de la sala i la taula **PERSONA** on s'emmagatzema les el total de persones (*personesDins*) que hi ha dins de la sala cada vegada que entra o surt algú i la data i hora (*dataHora*) en el moment que passa. Des de Node-RED s'insereixen i es consulten registres a la base de dades. A la següent taula es mostren les sentències SQL<sup>12</sup> que rep la base de dades:

Definició	Sentència SQL
Capacitat de la sala	SELECT capacitat FROM sala WHERE id=1
Nom de la sala	SELECT nom FROM sala WHERE id=1
Persones que hi ha dins la sala	SELECT personesDins FROM persona ORDER BY id...
Modificar la capacitat	UPDATE sala SET capacitat+=msg.payload+ WHERE id=1
Modificar el nom de la sala	UPDATE sala SET nom+=msg.payload+WHERE id=1
Inserir persones a la sala	INSERT INTO persona (dataHora,personesDins) VALUES ...

Taula 4.2: Sentències SQL

## 4.6 Funcionament del Sketch

En aquest apartat es detallen totes les accions que es realitzen el Sketch (projecte) de la placa electrònica.

1. Declaració de les llibreries necessàries
2. Definició de les variables i els objectes
3. Definició del **setup()**
  - (a) Connexió de la placa a la WiFi

<sup>12</sup>És un llenguatge estàndard de comunicació amb bases de dades relacionals.

(b) Connexió de la placa al broker, en aquest punt es comprova els tòpics als que s'està subscript i es recuperen els valors que hi ha a la base de dades, tant de les persones que hi ha dins com la capacitat de la sala. Això és així per si la placa es reinicia inesperadament recuperarà els valors que tenia en el moment del reinici.

(c) Inicialització dels sensors

#### 4. Definició del `loop()`

(a) Comprovar les distàncies dels sensors

(b) Si la distància del sensor 1 és més petita que l'amplada de la porta vol dir que algú està entrant.

(c) Si la distància del sensor 2 és més petita que l'amplada de la porta després d'haver passat pel sensor 1 vol dir que algú ha entrat, se suma 1 al total de les persones i s'envia el total al broker.

(d) Es comprova el mateix però a l'inrevés i si es dona el cas vol dir que algú ha sortit s'actualitza el número de persones que hi ha dins i s'envia al servidor MQTT.

(e) Tant si una persona comença a entrar com si comença a sortir però no acaba de fer-ho del tot, a cada `loop()` es calcula el temps que el sensor ha estat activat i si fa més de 5 segons es considera un fals positiu o simplement que la persona ha decidit canviar d'opinió tornem el sensor a l'estat de repòs.

## 4.7 Codi font

El codi font del prototip detallat en aquesta proposta està a la plataforma de desenvolupament GitHub. Es pot accedir mitjançant la següent URL [https://github.com/ebartres/tfm\\_ebartres](https://github.com/ebartres/tfm_ebartres) i a continuació es detalla el contingut de cadascun dels fitxers:

FITXER	CONTINGUT
ESP32.ino	Codi de l'Sketch desenvolupat en Arduino IDE
ESP32_Utils.hpp	Arxiu de configuració per connectar la placa a la WiFi
config.h	Arxiu amb les funcions per connectar la placa a la WiFi
flux_Node-RED.json	Flux Node-RED
tfm_database.sql	BBDD MySQL amb dades

Taula 4.3: Codi font del prototip: GitHub

# Capítol 5

## Resultats i discussió

En aquest capítol s'exposen alguns anàlisis bàsics sobre dades extretes d'una sala d'espera de test. La sala d'espera no va poder ser real donat que les limitacions i restriccions provocades per la COVID-19 han fet que els permisos per usar-la triguessin més del compte i finalment es va optar per instal·lar el prototip en una sala privada sense pacients reals i les persones que van participar-hi van ser col·laboradors.

### 5.1 Conceptes previs

Una sala d'espera pot contenir cap o molts pacients. El pacients o persones poden entrar o sortir de la sala segons els convingui però abans d'entrar el sistema els dirà si l'accés és permès mitjançant la pantalla que s'instal·larà a l'entrada on es visualitzarà dashboard.

S'entén que els pacients i/o les persones que facin ús de la sala disposen de la capacitat per entendre el que veuran a la pantalla del dashboard i sabran interpretar si poden accedir o contràriament s'han d'esperar a que la sala s'alliberi.

#### 5.1.1 Exposició del mètode utilitzat

Per analitzar les dades obtingudes en l'experiment dut a terme es realitza la metodologia de l'anàlisi descriptiu que ens proporciona una visió que es confecciona amb la informació que donen les dades d'una mostra. Es a dir, la finalitat es recol·lectar i ordenar la informació obtinguda i extreure les característiques més representatives.

## 5.2 Anàlisi dels resultats obtinguts

Per extreure resultats es situa el prototip en una sala privada i es compara les dades obtingudes pel sistema de control d'aforament amb la realitat. La sala d'espera es defineix amb un aforament màxim de 15 persones i s'analitza l'entrada i sortida de les persones de manera normal, sense aplicar cap norma d'accés a la sala. Durant l'experiment s'observa:

- Les persones accedeixen dins la sala un cop estan dins les 15 persones n'hi ha 2 que no són detectades.
- Les 15 persones surten de la sala n'hi ha una que no és detectada i una altre al sortir el prototip detecta la sortida però també detecta una entrada .

La conclusió és que el sistema funciona correctament amb la majoria de persones que entren i surten de la sala però hi ha casos que no donen els resultats esperats, els analitzo:

- **A l'entrada:** la primera persona que no es detecta es degut a que entra molt de pressa i tot i que el primer sensor ha detectat que hi ha passat algú el segon ni se n'ha adonat i per tant no ha comptabilitzat l'entrada. En el cas de la segona persona no detectada ha sigut perquè han entrat dues persones juntes i el prototip només n'ha detectat una.
- **A la sortida:** la primera persona que no ha estat detectada ha sigut pel mateix cas de l'entrada, dues persones surten alhora de la sala i el prototip només en detecta una i en el segon cas hi ha una persona que ha sortit aturant-se just davant del sensor i això ha fet que es generi un fals positiu d'entrada, tot i que la sortida l'ha comptabilitzat correctament.

En vista dels resultats obtinguts es proposa una solució. Sense haver de modificar la tecnologia utilitzada es torna a realitzar l'experiment a la mateixa sala i amb el mateix aforament però ara s'explica als usuaris que la sala disposa d'un sistema per comptar les persones i que cal que entrin i surtin d'un en un. Amb aquest nou escenari es torna a fer la prova i s'observa que:

- Totes les persones que entren a la sala el sistema les detecta.
- Totes les persones que surten de la sala el sistema les detecta.

### 5.2.1 Conclusió de l'experiment

La conclusió és que en aquest nou escenari el sistema realitza la funció pel que ha estat pensat però a costa d'afegir una restricció (que les persones entrin d'una en una) i informar els usuaris de que la sala disposa d'un sistema de control d'aforament ha fet que el resultat siguin del tot satisfactoris. Pel simple fet d'informar i explicar a l'usuari de que la sala disposa d'un dispositiu mecànic aquests han modificat el seu comportament habitual aconseguint que el sistema faci bé la seva funció.

Un cop obtinguts aquests resultats un dels objectius secundaris definits al inici del TFM era instal·lar el prototip en una sala d'espera real de l'Hospital. Malauradament no es va poder dur a terme perquè donada la situació de restriccions cada dia més severes provocades pel coronavirus han fet que no s'obtingui el permís de la gerència del hospital en el moment de l'entrega d'aquest TFM. Aquest problema ha suposat un entrebanc ja que no s'ha pogut complir uns dels objectius secundaris però en les següents setmanes s'espera poder obtenir el permís adequat.

## 5.3 Despesa econòmica i hores emprades

Donat que en aquest projecte tot el software que s'ha utilitzat és software lliure, en aquest aspecte només cal tenir en compte la despesa en recursos humans. Per poder iniciar la investigació ha sigut necessari comprar de material per a realitzar el prototip. La despesa per a la implementació del prototip final es pot veure en la següent taula:

Article	Preu
Placa HiLetgo UNO R3 D1 R32 ESP32	10,49€
Cables DuPont per interconnexió	6,99€
Placa ESP32 NodeMCU	6,75€
Placa NodeMCU V3 ESP8266EX	7,79€
Sensors VL53L0X	24,99€
Kit d'iniciació amb placa Arduino UNO	32,99€
<b>TOTAL</b>	<b>90,00€</b>

Taula 5.1: Despesa econòmica en hardware

A continuació es detallen les hores emprades per al desenvolupament d'aquest TFM:

- **Definició del projecte** i reunions amb el professor i responsables del hospital per definir el projecte i fer-ne el seguiment. *Temps dedicat: 15 hores.*

- **Estudi del llenguatge C** que s'utilitza a Arduino IDE i del funcionament de les plaques tipus Arduino així com la implementació de les funcionalitats detallades als objectius del TFM. *Temps dedicat 40 hores.*
- **Investigació sobre la detecció** i els diferents sensors que ho permeten. *Temps dedicat: 70 hores.*
- **Investigació sobre l'enviament de les dades captades** amb els sensors i mostrar-les en un dashboard i emmagatzemar-les en una base de dades. *Temps dedicat: 30 hores.*
- **Desenvolupament de l'aplicació.** Implementar el codi de la placa que gestiona els sensors i connectar-se a la WiFi i al servidor MQTT. *Temps dedicat: 60 hores.*
- **Proves** amb l'aplicació i comprovació del seu funcionament. *Temps dedicat: 15 hores.*
- **Documentació** d'aquesta memòria. *Temps dedicat 60 hores.*
- **Presentació** del treball i gravació de la defensa. *Temps dedicat 20 hores.*

La suma total de les hores dedicades és de **310 hores**.

# Capítol 6

## Conclusions

En aquest capítol es pretén explicar si s’han complert els objectius previstos en aquest projecte. També mostra la visió personal sobre la seva continuïtat i posta en marxa en un escenari real així com quins serien els següents passos per a fer-ho possible.

### 6.1 Conclusions i visió personal

La solució proposada en aquest TFM és viable ja que el cost de la infraestructura és mínim i els serveis que es necessiten es poden instal·lar a qualsevol servidor inclús poden córrer en un microordinador com pot ser una RaspBerry Pi<sup>1</sup>.

En referència a desenvolupar un prototip capaç de comptabilitzar l’aforament d’un espai tancat, l’objectiu principal que es va definir a l’inici del projecte, s’ha assolit de manera satisfactòria i amb un percentatge alt de fiabilitat. A més el prototip es de dimensions petites, i en una fase posterior, es podrà encapsular fàcilment per poder-lo instal·lar a qualsevol entrada d’un espai tancat.

Pel que fa l’objectiu secundari d’emetre alertes quan es superi l’aforament màxim fent-les visibles als usuaris, es possible gràcies al dashboard desenvolupat amb Node-RED, i per tal de fer les alertes visibles als usuaris és necessari una tauleta o ordinador connectat a la mateixa xarxa amb visió el servidor Node-RED. D’aquesta manera carreguem el dashboard i el deixem visible a l’entrada i/o sortida de la sala per tal de que els usuaris visualitzin si poden accedir o no a la sala.

Enregistrar les dades de l’aforament per poder-les estudiar i/o consultar-les també s’ha complert. S’emmagatzemen a la base de dades cada vegada que entra o surt algú de la sala la data i hora i les persones que hi ha dins la sala. Amb aquesta informació es pot extreure

---

<sup>1</sup>És una sèrie d’ordinadors de placa reduïda de baix cost.

informació de quan la sala està més ocupada o quan ho està menys, i d'aquesta manera poder citar a les persones en les franges menys ocupades per exemple.

Finalment, instal·lar el prototip en un espai real i que funcioni de manera eficient, és un objectiu que es pot implementar en una fase futura donats els bons resultats obtinguts en els escenaris de test. Actualment s'està valorant amb els responsables del hospital una possible ubicació per posar el sistema en producció.

## 6.2 Continuitat del projecte

El prototip construït en aquest TFM pot ser aprofitat per instal·lar-lo en qualsevol sala d'espera o espai tancat. La pandèmia provocada pel coronavirus ha fet que s'hagin de prendre mesures excepcionals i que fins ara mai s'havien pres i aquest treball des del meu punt de vista, té sentit de que el prototip es perfeccioni i inclús fer-lo extens en altres àmbits o altres centres sanitaris que ho necessitin.



# Glosari

**API** Aplication Programming Interface.

**BBDD** Base de Dades.

**COVID-19** Malaltia vírica provocada per un nou coronavirus.

**Dashboard** També anomenat tauler de comandaments. Document on es reflecteixen, mitjançant una representació gràfica, les dades proporcionades pel prototip.

**GND** Ground.

**I2C** Inter-Integrated Circuit.

**IoT** Internet of Things, Internet de les Coses.

**IP** Internet Protocol.

**MQTT** Message Queue Telemetry Transport.

**OMS** Organització Mundial de la Salut.

**PIR** Passive Infrared.

**RFID** Radio Frequency Identification.

**SCL** System Clock.

**SDA** Serial Data.

**SQL** Structured Query Language.

**SSID** Service Set Identifier.

**TFM** Treball Final de Màster.

**ToF** Time of Flight.

**UOC** Universitat Oberta de Catalunya.

**VCC** Voltage Common Collector.

# Bibliografía

- [1] Somphop Chanthakit and Choopan Rattanapoka. Mqtt based air quality monitoring system using node mcu and node-red. In *2018 Seventh ICT International Student Project Conference (ICT-ISPC)*, pages 1–5. IEEE, 2018.
- [2] Oracle Corporation. Mysql 8.0 reference manual. <https://dev.mysql.com/doc/refman/8.0/en/>, 2020. Revisió: Desembre 2020.
- [3] Enrique Crespo. Aprendiendo arduino. <https://aprendiendoarduino.wordpress.com/2017/01/23/pr-arduino-5/>, 2017. Revisió: Gener 2017.
- [4] Jörg Eschmeier and Roland Wolff. Composition of inner mappings on the ball. *Proceedings of the American Mathematical Society*, 130(1):95–102, 2002.
- [5] Ryan Fukuhara, Leonard Day, Huy H Luong, Robert Rasmussen, and Savio N Chau. I2c bus protocol controller with fault tolerance, April 27 2004. US Patent 6,728,908.
- [6] Marcos Manuel Gallego Ferrer. *Safe Events: control de aforos mediante RFID*. PhD thesis, 2017.
- [7] Andrés Mauricio García Piragua, William Alexander Moreno Gómez, et al. Sistema de conteo automático de flujo de personas por medio de visión artificial. 2016.
- [8] Damián Ibáñez Fernández. *Desarrollo de un sistema de monitorización y control mediante visión artificial y redes neuronales siamesas*. PhD thesis, 2019.
- [9] Idasfest. Control de aforo en eventos: ¿quÉ modalidades existen? <https://www.idasfest.es/blog/modalidades-control-aforo-eventos>. Revisió: Març 2019.
- [10] Nikola Laković, Miodrag Brkić, Branislav Batinić, Jovan Bajić, Vladimir Rajs, and Nenad Kulundžić. Application of low-cost vl53l0x tof sensor for robot environment detection. In *2019 18th International Symposium INFOTEH-JAHORINA (INFOTEH)*, pages 1–4. IEEE, 2019.

- [11] Walter Maner. Model de prototip: sidar.org. <http://www.sidar.org/recur/desdi/traduc/es/visitable> 2000. Revisió: Febrer 2000.
- [12] José Antonio Salmerón Marín et al. Control de aforo, peso y altura máxima/mínima en un recinto cerrado. 2008.
- [13] Dipa Soni and Ashwin Makwana. A survey on mqtt: a protocol of internet of things (iot). In *International Conference On Telecommunication, Power Analysis And Computing Techniques (ICTPACT-2017)*, 2017.
- [14] Grupo SPEC. ¿por qué deberías usar un sistema de control de aforo en tu empresa? <https://www.grupospec.com/es/blog/116-control-aforo>, 2020. Revisió: 2020.
- [15] Víctor Yepes Piqueras. Método simplificado de cálculo del aforo de las playas en tiempos de coronavirus. 2020.

# Apèndix A

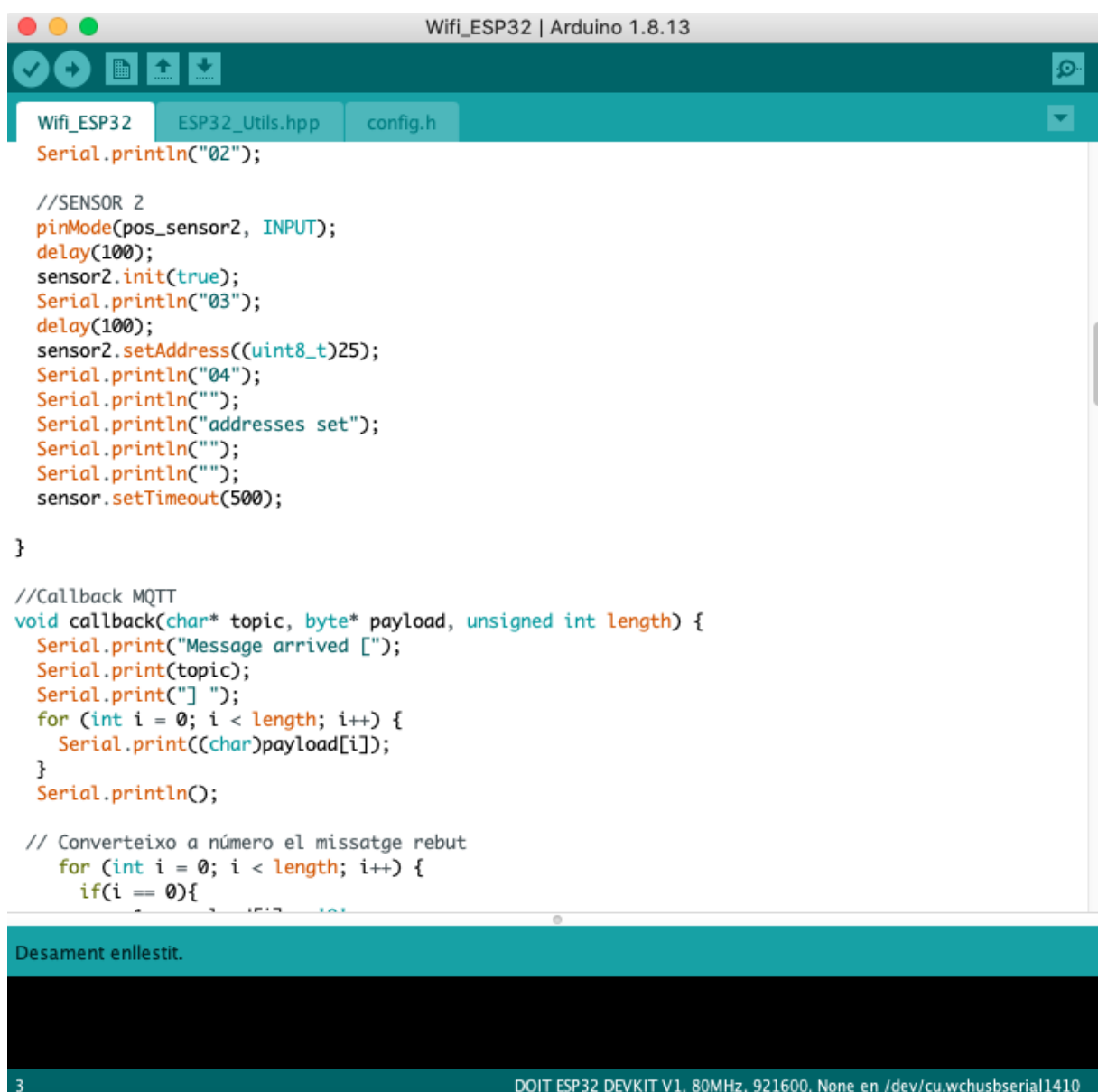
## Instal·lació del software necessari per el desenvolupament del prototip

En aquest apèndix explicaré com instal·lar el software utilitzat pel desenvolupament del prototip. En aquest cas s'ha utilitzat un ordinador MacBook Pro amb el sistema operatiu macOS Catalina versió 10.15.7 però tot el software és compatible amb Windows i Linux per tant es pot adaptar a les necessitats que més ens convinguin a cada moment.

## A.1 Arduino IDE

La instal·lació d'Arduino IDE és molt senzilla, simplement cal descarregar l'executable, instal·lar-lo i començar a programar, com es pot veure a la figura A.1. La pàgina oficial és la següent: <https://www.arduino.cc/en/software> i un cop instal·lat és molt intuïtiu de fer servir.

Arduino també disposa d'un Editor Online (<https://create.arduino.cc/editor/>) que permet utilitzar l'IDE sense haver d'instal·lar res l'equip. Hi ha una limitació important, aquesta interfície únicament permet treballar amb plaques oficials d'Arduino i per això en el cas d'aquest treball finalment va quedar descartat. Per fer-lo servir únicament cal registrar-se.



```
Wifi_ESP32 | Arduino 1.8.13
Wifi_ESP32 ESP32_Utils.hpp config.h
Serial.println("02");

//SENSOR 2
pinMode(pos_sensor2, INPUT);
delay(100);
sensor2.init(true);
Serial.println("03");
delay(100);
sensor2.setAddress((uint8_t)25);
Serial.println("04");
Serial.println("");
Serial.println("addresses set");
Serial.println("");
Serial.println("");
sensor.setTimeout(500);

}

//Callback MQTT
void callback(char* topic, byte* payload, unsigned int length) {
  Serial.print("Message arrived [");
  Serial.print(topic);
  Serial.print("] ");
  for (int i = 0; i < length; i++) {
    Serial.print((char)payload[i]);
  }
  Serial.println();

  // Converteixo a número el missatge rebut
  for (int i = 0; i < length; i++) {
    if(i == 0){

```

Desament enllestit.

3 DOIT ESP32 DEVKIT V1, 80MHz, 921600, None en /dev/cu.wchusbserial1410

Figura A.1: Visualització de la pantalla d'Arduino IDE

## A.2 Eclipse Mosquitto

Eclipse Mosquitto és el servei MQTT utilitzat en el TFM. Com que es tracta d'un servei no es tracta d'una aplicació pròpiament dita. Per instal·lar-lo a macOS es fa seguint els següents passos:

1. Obrir una finestra Terminal i instal·lem Homebrew que ens permetrà instal·lar Mosquitto de manera senzilla. Cal executar l'ordre:

```
/usr/bin/ruby -e "$(curl -fsSL
https://raw.githubusercontent.com/Homebrew
/install/master/install)"
```

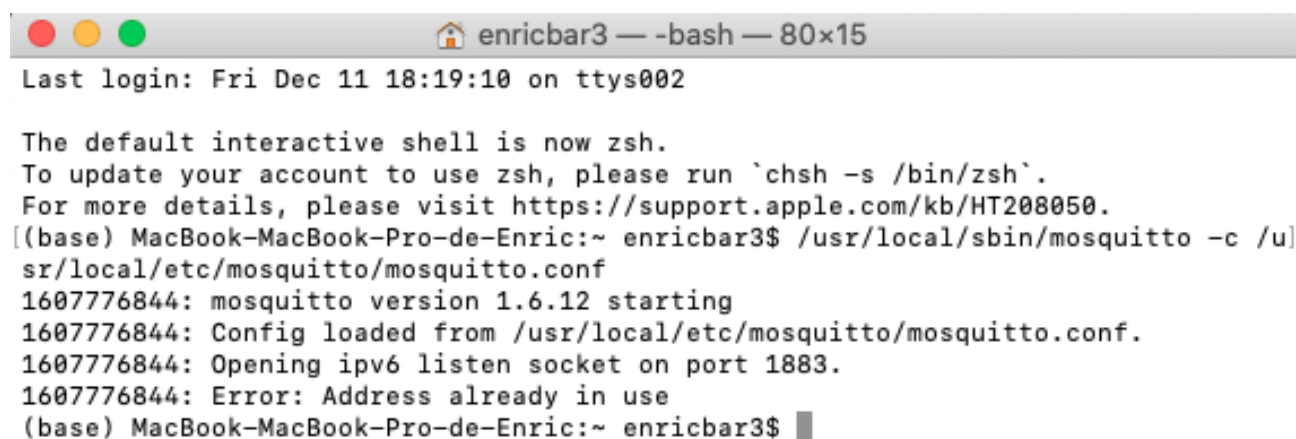
2. Seguidament cal demanar a Homebrew que instal·li Mosquitto:

```
brew install mosquitto
```

3. Un cop finalitzada la instal·lació de Mosquitto cal executar l'ordre següent en una nova finestra de terminal per iniciar Mosquitto.

```
/usr/local/sbin/mosquitto -c /usr/local/etc/mosquitto/mosquitto.conf
```

4. Mosquitto MQTT ja està corrent en el nostre ordinador i preparat per començar a rebre missatges.



```
enricbar3 — -bash — 80x15
Last login: Fri Dec 11 18:19:10 on ttys002

The default interactive shell is now zsh.
To update your account to use zsh, please run `chsh -s /bin/zsh`.
For more details, please visit https://support.apple.com/kb/HT208050.
[(base) MacBook-MacBook-Pro-de-Enric:~ enricbar3$ /usr/local/sbin/mosquitto -c /u]
sr/local/etc/mosquitto/mosquitto.conf
1607776844: mosquitto version 1.6.12 starting
1607776844: Config loaded from /usr/local/etc/mosquitto/mosquitto.conf.
1607776844: Opening ipv6 listen socket on port 1883.
1607776844: Error: Address already in use
(base) MacBook-MacBook-Pro-de-Enric:~ enricbar3$ █
```

Figura A.2: Servei Mosquitto MQTT actiu

## A.3 Node-RED

Per instal·lar Node-RED primer de tot cal tenir instal·lada una versió compatible de NODE.JS. Per instal·lar NODE.JS simplement cal descarregar l'instal·lador de la pàgina oficial <https://nodejs.org/es/download/> i un cop instal·lat podrem seguir els següents passos per tenir Node-RED:

1. Obrir un terminal i escriure l'ordre npm que s'inclou amb node.js executant la següent sentència:

```
sudo npm install -g --unsafe-perm node-red
```

2. S'instal·larà Node-RED com a mòdul global juntament amb les seves dependències. En acabar apareixerà un missatge dient que s'han afegit els packages.

```
+ node-red@1.1.0
added 332 packages from 341 contributors in 18.494s
found 0 vulnerabilities
```

3. Un cop instal·lat el mòdul global ja es pot començar a utilitzar Node-RED però cal iniciar-lo a través d'un terminal amb l'ordre *node-red*. Per aturar el programa es pot fer mitjançant *Ctrl-C* o tancant la finestra.

A la figura [A.3](#) es pot veure la sortida del registre que proporciona dades com la versió de Node-RED i Node.js, si s'ha produït algun error al carregar els nodes, la ubicació del fitxer de configuració i el nom del fitxer del flux que s'està utilitzant. Al executar Node-RED podem incloure arguments per canviar el port TCP, el nom de la finestra del procés,... per realitzar aquest treball s'han deixat les opcions per defecte.

A continuació, es pot accedir a l'editor a través d'un navegador web mitjançant l'adreça <http://localhost:1880/>



```
enricbar3 — node-red TERM_PROGRAM=Apple_Terminal — 80x24
Last login: Fri Dec 4 10:16:01 on console

The default interactive shell is now zsh.
To update your account to use zsh, please run `chsh -s /bin/zsh`.
For more details, please visit https://support.apple.com/kb/HT208050.
[(base) MacBook-MacBook-Pro-de-Enric:~ enricbar3$ node-red
4 Dec 16:13:35 - [info]
Welcome to Node-RED
=====

4 Dec 16:13:35 - [info] Node-RED version: v1.2.5
4 Dec 16:13:35 - [info] Node.js version: v14.15.1
4 Dec 16:13:35 - [info] Darwin 19.6.0 x64 LE
4 Dec 16:13:35 - [info] Loading palette nodes
4 Dec 16:13:37 - [info] Dashboard version 2.24.0 started at /ui
4 Dec 16:13:37 - [info] Settings file : /Users/enricbar3/.node-red/settings.js
4 Dec 16:13:37 - [info] Context store : 'default' [module=memory]
4 Dec 16:13:37 - [info] User directory : /Users/enricbar3/.node-red
4 Dec 16:13:37 - [warn] Projects disabled : editorTheme.projects.enabled=false
4 Dec 16:13:37 - [info] Flows file : /Users/enricbar3/.node-red/flows_MacBoo
k-MacBook-Pro-de-Enric.local.json
4 Dec 16:13:37 - [info] Server now running at http://127.0.0.1:1880/
```

Figura A.3: Execució via Terminal de Node-RED

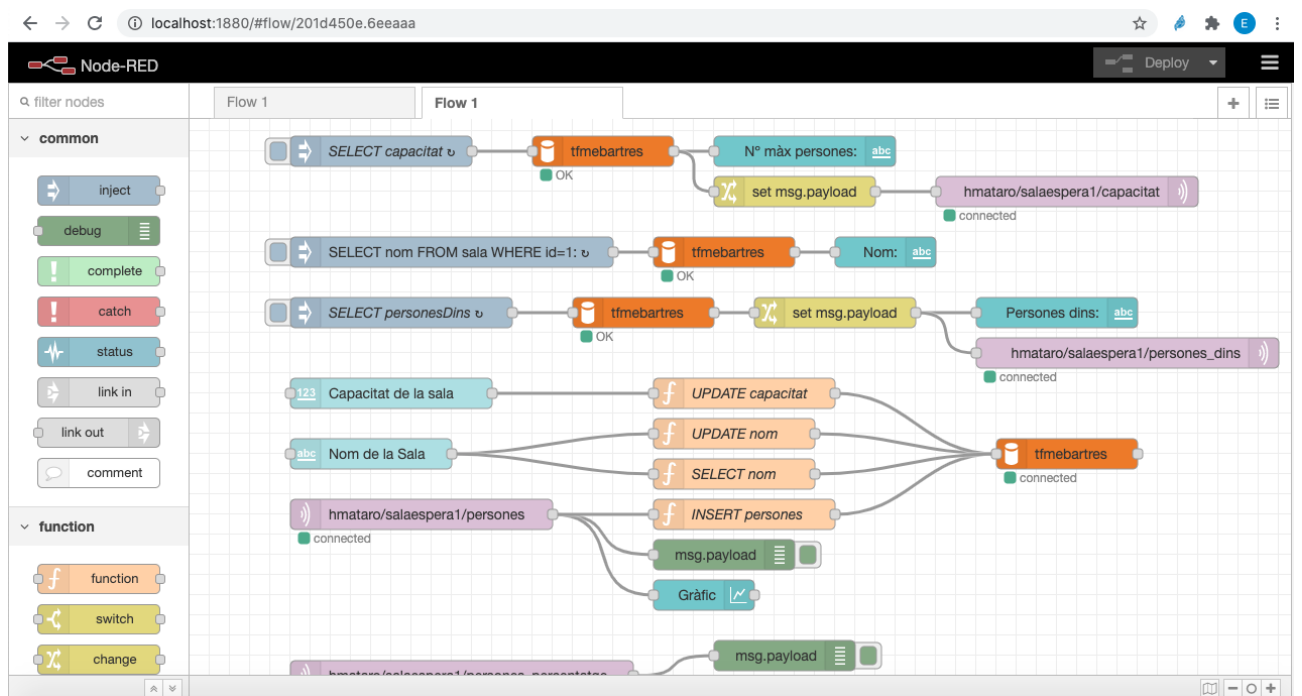


Figura A.4: Vista de l'aplicació Node-RED

Com es pot veure a la figura A.4 a l'esquerra hi ha tots els nodes per crear el flux que es

vulgui. En el cas d'aquest treball es van haver d'instal·lar els mòduls per disposar dels nodes MQTT i els nodes MySQL. Per fer-ho es tant senzill com anar al menú de la dreta a *Manage Palette* i seleccionar la pestanya *Install*. Allà hi ha un cercador on es pot filtrar la cerca (figura A.5 i instal·lar el que es necessiti fent clic a *install*.

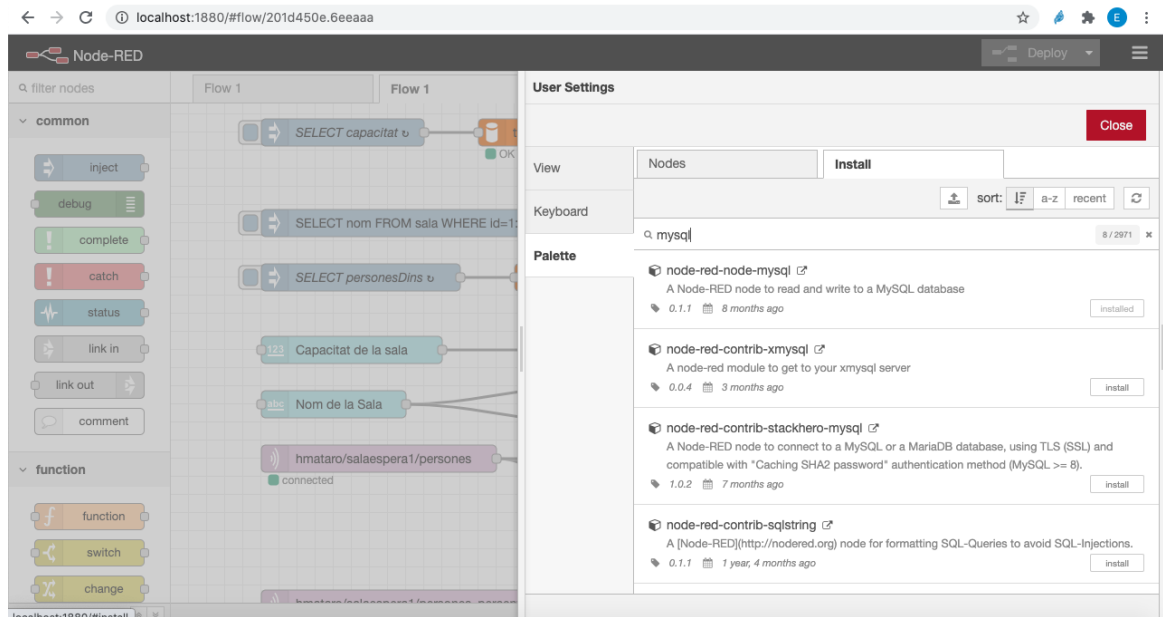


Figura A.5: Instal·lació de mòduls addicionals de Node-RED

Per acabar, com que en el flux s'ha incorporat nodes del tipus dashboard es podran visualitzar a través de la url <http://localhost:1880/ui> com es veu a la figura A.6



Figura A.6: Dashboard de Node-RED

## A.4 MySQL

Per instal·lar MySQL existeixen infinitat de paquets que incorporen aquest gestor de base de dades. En el cas d'aquest treball s'ha optat per instal·lar únicament el MySQL per tal d'optimitzar recursos. Per fer-ho cal dirigir-se a la pàgina oficial de Oracle MySQL <https://dev.mysql.com/downloads/mysql/> i descarregar l'instl·lador i seguir els passos.

MySQL no disposa d'interfície gràfica i s'ha de treballar a través d'un terminal. El que sí que disposa és d'un panell que es pot accedir a través del tauler de control on ens mostra la versió de MySQL si el servei està actiu i els directoris on es troben els fitxers necessaris pel seu funcionament.

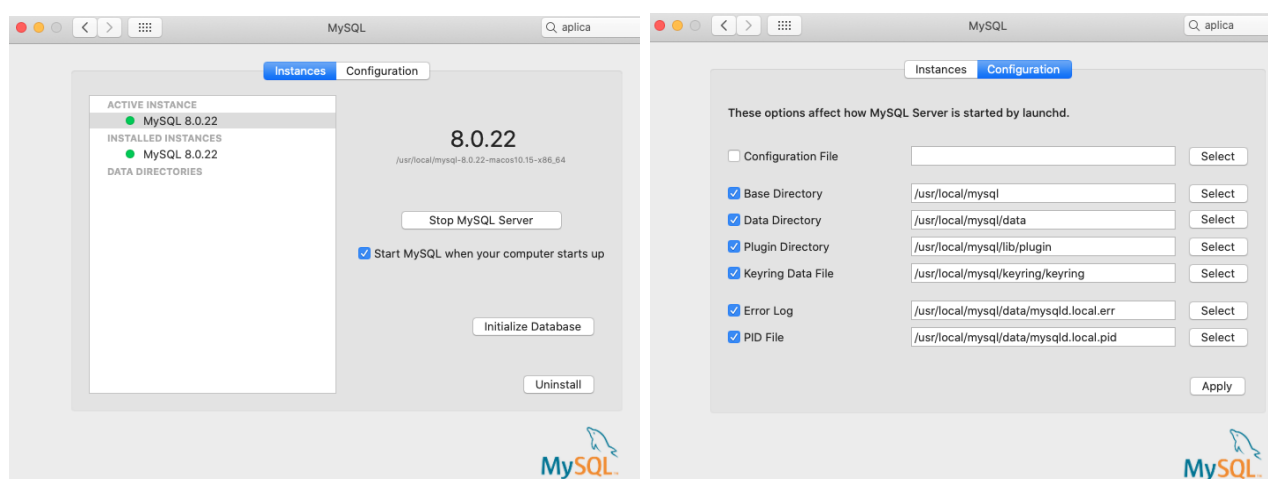


Figura A.7: Panell de control de MySQL

Per poder manejar les bases de dades de MySQL des d'un terminal s'ha d'executar la comanda *mysql* al directori d'instal·lació:

```
cd /usr/local/mysql-8.0.22-macos10.15-x86_64/bin
./mysql -u root -p
```

Un cop dins es poden usar les comandes pròpies de MySQL per crear bases de dades, taules, afegir registres,...