



Escola Politécnica Superior d'Enginyeria de Manresa

Desenvolupament d'un braçalet mèdic intel·ligent

Projecte VITALS

8 de juliol de 2022

Treball de fi de grau que presenta

FERRAN FEIXAS FLOTATS

En compliment dels requisits per assolir el

GRAU D'ENGINYERIA EN SISTEMES TIC

Direcció: Pere Palà

Agraïments

El meu agraïment al tutor Pere Palà, als professors Arnau Arumí, Joan Martínez i al tècnic de laboratori Manel Blanes pel seu suport durant tot el projecte i l'ajuda constant pel disseny i elaboració del hardware del sistema així com, els consells i els punts de vista requerits en moments clau.

Resum

Aquest treball té com a objectiu principal crear un sistema de monitoratge continu de les constants vitals dels pacients ingressats als hospitals. Aquest rep el nom de projecte VITALS, i es basa en el disseny i implementació d'un braçalet intel·ligent sense fils, de nom Bibo band, encarregat de prendre les constants vitals als pacients ingressats i enviar aquestes dades a un servidor central perquè puguin ser mostrades a través de la plataforma web. El projecte també compta amb un sistema d'alarms per indicar si les constants dels pacients no són les òptimes i informar així al personal sanitari.

Per a la correcta realització de les tasques plantejades, tant aquelles delegades al hardware com les de software, s'han englobat dins del projecte totes les diverses àrees de coneixement que s'han ofert al llarg del grau d'enginyeria de sistemes TIC. Entre aquestes àrees s'hi troben l'electrònica, la informàtica o bé les comunicacions i totes actuen de forma sinèrgica per a oferir als usuaris un servei tangible i eficaç.

Abstract

The main goal of this project is to create a system for continuous monitoring of the vital signs of patients admitted to hospitals. This project, named VITALS, is based upon the design and implementation of a wireless smart bracelet, called Bibo band, which is responsible for analysing the vital signs of hospitalized patients and sending their data to a central server so said signs can be displayed through the web platform of the project. VITALS also has an alarm system which signs if the patient's constants are not optimal and thus inform health personnel.

For the correct performance of the proposed tasks, both those delegated to hardware and software alike, all the various areas of knowledge that has been offered throughout the degree of ICT systems engineering has been included in the project. These areas include electronics, computing and communications, all of which act synergistically to provide users with a tangible and effective service.

Índex

1. Introducció.....	3
2. Entorn i sistemes actuals.....	5
2.1. Elements d'un sistema de monitoratge.....	5
2.2. Situació hospitalària en l'actualitat.....	7
2.3. Competència actual.....	10
3. Requeriments i arquitectura del projecte VITALS	12
3.1. Requeriments del sistema	12
3.2. Arquitectura del sistema.....	12
4. Perifèrics.....	16
4.1. Bibo band.....	16
4.2. Dispositiu d'alarmes.....	41
5. Servidor	47
5.1. Aplicació web.....	47
5.2. Allotjament del servidor.....	48
5.3. Servidor web.....	49
5.4. WSGI	50
5.5. Seguretat TLS	51
5.6. Nom de domini	53
5.7. Base de dades.....	54
6. Plataforma web.....	62
6.1. Tecnologia i llenguatge utilitzat.....	62
6.2. Funcionalitats	62
7. Manual d'usuari	68
7.1. Manual dels dispositius físics	68
7.2. Manual de la plataforma web	72
8. Conclusions.....	88
8.1. Hardware.....	88
8.2. Software.....	89
9. Bibliografia.....	90

II·lustracions

II·lustració 1: Esquema del projecte VITALS	13
II·lustració 2: Mànquina d'estats Bibo band	23
II·lustració 3: Esquemàtic convertidor USB a UART	32
II·lustració 4: Gestor de la programació del microcontrolador	33
II·lustració 5: Esquemàtic del controlador de càrrega	34
II·lustració 6: Esquemàtic del regulador de tensió	35
II·lustració 7: Esquemàtic ESP32-WROOM-32	36
II·lustració 8: Esquemàtic sensor de temperatura i connector de 4 pins.....	36
II·lustració 9: Part frontal de la placa de desenvolupament i model 3D	38
II·lustració 10: Part inferior de la placa de desenvolupament i model 3D.....	38
II·lustració 11: Resultat final del disseny de la PCB	38
II·lustració 12: Disseny de la caixa de la Bibo band	39
II·lustració 13: Cara superior i inferior de la tapa de la Bibo band	40
II·lustració 14: Resultat final del disseny de la Bibo band	41
II·lustració 15: Mànquina d'estats dispositiu d'alarms.....	42
II·lustració 16: Resultat final del disseny del dispositiu d'alarms.....	46
II·lustració 17: Esquema Entitat/Relació base de dades	54
II·lustració 18: Port microUSB de la Bibo band.....	68
II·lustració 19: Estat de càrrega de la Bibo band	69
II·lustració 20: Botó d'engegar de la Bibo band.....	69
II·lustració 21: Botó d'apagar de la Bibo band	70
II·lustració 22: Sensor de temperatura de la Bibo band	70
II·lustració 23: Sensor de freqüència cardíaca i saturació d'oxigen de la Bibo band	71
II·lustració 24: Alimentació dels dispositius d'alarms	71
II·lustració 25: Pàgina inicial de la plataforma VITALS.....	72
II·lustració 26: Pàgina de registre de la plataforma VITALS.....	73
II·lustració 27: Pàgina d'iniciar sessió de la plataforma VITALS	73
II·lustració 28: Pàgina de perfil de la plataforma VITALS	74
II·lustració 29: Estat de les Bibo bands dels pacients	74
II·lustració 30: Panell de pacients de la plataforma VITALS.....	75
II·lustració 31: Panell d'alarms de la plataforma VITALS	76
II·lustració 32: Històric pacients plataforma VITALS	77
II·lustració 33: Formulari d'edició de pacients de la plataforma VITALS	78
II·lustració 34: Històric d'alarms de la plataforma VITALS	79
II·lustració 35: Pagina de gestió de dades de la plataforma VITALS	80
II·lustració 36: Formulari d'edició de polseres simple de la plataforma VITALS	81
II·lustració 37: Pàgina de pacients de la plataforma VITALS.....	82
II·lustració 38: Formulari de registre de pacients de la plataforma VITALS	83
II·lustració 39: Estats de la xarxa de connexió de les Bibo bands.....	83
II·lustració 40: Pàgina de dispositius de la plataforma VITALS.....	84
II·lustració 41: Formulari d'edició de polseres de la plataforma VITALS.....	85
II·lustració 42: Formulari d'edició de dispositius d'alarms de la plataforma VITALS	85
II·lustració 43: Pàgina d'usuaris de la plataforma VITALS	86
II·lustració 44: Formulari d'edició d'usuaris de la plataforma VITALS	87

1. Introducció

Avui en dia, la tecnologia està a l'abast de la gran majoria de sectors de la societat i, dins d'aquest context, el sector sanitari és un dels que requereix unes prestacions més sofisticades i punteres. Aquesta necessitat s'ha vist evidenciada cap al gran públic arran de la pandèmia de la COVID-19 i els grans reptes tecnològics que aquesta ha generat, especialment a l'hora de poder realitzar l'aïllament dels pacients i seguir-ne l'estat.

El projecte VITALS té com a objectiu millorar el seguiment dels pacients ingressats als hospitals a través d'un monitoratge continu de les seves constants vitals i el control d'aquestes pels professionals sanitaris en temps real. El sistema implementat es basa en un sistema pensat al detall, amb la prèvia anàlisi de l'entorn i dels requeriments que aquest ha de complir. El projecte, doncs, se centra en un braçalet intel·ligent capaç de prendre les constants vitals com són la temperatura, la freqüència cardíaca i la saturació d'oxigen de forma continuada i enviar les dades a un servidor central perquè siguin visibles per tot el personal sanitari a través de la plataforma web. Addicionalment, el sistema també té la capacitat de generar senyals d'alarma visuals i en temps real en cas que les constants dels pacients no siguin les òptimes i alertar, així, al personal mèdic el més ràpid possible.

Aquest sistema, però, no intenta en cap moment substituir la feina de cap professional ni servir com a eina diagnòstica. Simplement, vol ser una eina més de mesura i d'ajuda a l'hora de prendre decisions mèdiques en l'àmbit hospitalari, així com ajudar a reduir tasques dels professionals sanitaris en pics de feina elevats. Per exemple, seguint amb l'estela de la pandèmia, el requeriment d'haver d'emprar la indumentària adient per a entrar cada vegada en una habitació i realitzar una simple presa de les constants vitals dels pacients, no només genera un deteriorament físic i mental important en el personal sanitari, sinó que també comporta una pèrdua de temps que podria dedicar-se a altres tasques. D'aquesta forma, gràcies a aquest sistema, és possible que es pugui estalviar algun d'aquests torns de mesurament de les constants vitals per part dels professionals si es confia en les dades mesurades pels braçalets intel·ligents.

Un altre objectiu d'aquest projecte és poder posar a la pràctica tot el que s'ha après durant els quatre anys del grau d'enginyeria de sistemes TIC, ja que engloba les tres àrees fonamentals d'aquest grau com són l'electrònica, la informàtica i les comunicacions. És per això que com a propòsit personal, s'ha dissenyat i creat el sistema utilitzant únicament les eines apreses durant aquests anys en cada una de les tres àrees esmentades. És una manera de poder veure que el que s'ensenya a la universitat té aplicacions pràctiques i no únicament teòriques com es pot pensar en un principi.

Al llarg de la memòria del treball s'hi descriu de forma detallada cada una de les parts que componen el sistema del projecte VITALS. Primer de tot s'hi troba una explicació i una petita anàlisi de l'actualitat i dels requisits que ha de complir el sistema, i posteriorment s'explica aquest de forma detallada amb cada una de les parts, tant pel que fa a hardware com de software.

Finalment, s'hi troba un manual d'usuari on es detalla com posar en marxa el sistema i com utilitzar la plataforma web, ja que és l'eina de treball del sistema.

2. Entorn i sistemes actuals

Actualment, la sanitat és un camp que està en constant desenvolupament pel que fa a tecnologia i serveis. Un d'aquests avenços és el del monitoratge a distància dels pacients. Aquest monitoratge de pacients es porta a terme a través d'aplicacions senzilles a dins les habitacions dels hospitals, així com, des de les seves llars, podent dur a terme les activitats del dia a dia sense que aquest monitoratge interfereixi en elles. A través de la tecnologia de comunicacions i sensors, s'aconsegueix controlar certes constants vitals essencials, com són la lectura de l'electrocardiograma pel càlcul de la freqüència cardíaca, la freqüència respiratòria, la pressió arterial, la temperatura, la glucosa en sang i l'activitat del sistema neural[1].

Els avantatges del monitoratge remot son: la detecció primerenca i en temps real de malalties, la capacitat de monitorar contínuament els pacients, la prevenció de l'empitjorament de les malalties i de les morts prematures i l'increment de l'eficiència dels serveis sanitaris mitjançant la utilització de les tecnologies de la comunicació[2].

Aquests sistemes de monitoratge remot tenen els següents elements bàsics.

- Sistema d'adquisició de dades
- Sistema de processament de dades
- Xarxa de comunicacions.

2.1. Elements d'un sistema de monitoratge

2.1.1. Sistema d'adquisició de dades

Els sistemes de monitoratge remot de pacients estan dissenyats per obtenir una sèrie de dades fisiològiques dels pacients. Com ja s'ha comentat, les més comunes són la freqüència cardíaca, la freqüència respiratòria, la saturació d'oxigen en sang, els senyals del sistema nerviós, la pressió arterial, la temperatura corporal i el nivell de glucosa en sang[2].

Els sistemes més tradicionals solen utilitzar sensors col·locats al cos tot i que això causa problemes de mobilitat als pacients i a les activitats del seu dia a dia. A causa d'aquesta incomoditat del pacient, les mesures fisiològiques es veuen influenciades i no són reals, fent que no es mostri realment l'estat del pacient, sinó l'estat de la seva incomoditat. És per això que cada vegada es busca més prendre mesures amb sensors que no estiguin en contacte amb el pacient o siguin el menys invasius possible[1].

Així i tot, si es parla de sensors corporals, és freqüent que es parli de la xarxa de sensors corporals sense fils que formen les xarxes (WBAN). Aquestes xarxes consisteixen en un conjunt de dispositius vestibles o implantables que es troben prop del cos de la persona. Aquests nodes cooperen entre ells per tal de realitzar el monitoratge de la persona en qüestió. Els diferents nodes sensors s'encarreguen de varíes aplicacions segons el fabricant com per

exemple, el fet de mesurar les constants vitals d'una persona i així donar informació sobre el seu estat físic amb finalitats mèdiques. Aquestes xarxes de sensors es poden comunicar amb internet a través de diferents tecnologies sense fils com ZigBee, xarxes de sensors sense fils (WSN), Bluetooth o xarxes d'àrea local (WLAN) entre altres[3].

Aquestes xarxes WBAN funcionen de la següent manera:

- 1- Els nodes sensors recullen la informació del cos de la persona i l'envien a la unitat base situada també al cos[4].
- 2- La unitat base és l'encarregada de transmetre la informació dels sensors a través d'internet per aconseguir que la informació sigui visible i es puguin tractar les dades. Tot i la implementació d'aquesta tecnologia, un dels reptes més importants a l'hora de l'elecció dels sensors és la fiabilitat de les dades que aquests proporcionen. És rellevant el fet d'assegurar la màxima fiabilitat sempre que es vulguin utilitzar aquestes dades per fins diagnòstics[4].

2.1.2. Sistema de processament de dades

Aquests sistemes inclouen un sistema de recepció i transmissió de dades amb una unitat o circuit de processament. Aquest sistema sol estar a l'hospital (sigui una base de dades o un ordinador). En l'àmbit de seguretat és molt important poder assegurar la integritat d'aquestes dades emmagatzemades, així com la seva seguretat contra atacs maliciósos. És per això que és necessari un sistema de seguretat com per exemple un xifratge asimètric per garantir l'accés de persones de confiança[5].

2.1.3. Xarxa de comunicació

La xarxa de comunicació s'encarrega de connectar el sistema d'adquisició de dades amb el sistema de processament de dades així com, comunicar aquestes dades al professional de la salut que hi ha darrere a través d'una interfície gràfica. El principal aspecte a tenir en compte o la principal problemàtica és la seguretat. Els sistemes de salut electrònica tenen tres parts principals que són: els pacients, els proveïdors de serveis mèdics i els professionals de la salut. És per això que els sistemes s'han de pensar i dissenyar tenint en compte aquesta estructura. També s'ha de tenir en compte que en un sistema de monitoratge remot, es transmet informació dels pacients i les dades dels seus registres fisiològics. Aquestes dades s'obtenen a través del sistema de sensors comentat anteriorment, juntament amb altres components del sistema. És per això que és de vital importància la seguretat i protecció de la privacitat per poder fer front a qualsevol atac maliciós[6].

Algunes solucions de seguretat basades en l'estructura del sistema on els tres grans blocs són: els pacients, els proveïdors del sistema i els professionals sanitaris, és muntar un accés basat en rols per així poder assegurar la informació delicada del sistema.

Pel que fa a la seguretat de les mateixes comunicacions i de la xarxa, és útil l'ús de protocols d'encaminament segurs, algoritmes basats en el xifratge i l'ús de la signatura digital. D'aquesta manera es pot assegurar la integritat, la protecció i l'autenticitat de les dades transmeses[7].

2.2. Situació hospitalària en l'actualitat

En els hospitals, la pràctica rutinària de monitoratge implica el mesurament de les constants vitals (saturació d'oxigen, temperatura, pressió arterial, freqüència cardíaca i freqüència respiratòria) així com els sistemes de puntuació d'alertes primerenques com són els SAT, aquests últims són sistemes utilitzats pels equips d'atenció hospitalària per tal de reconèixer signes de deteriorament clínic amb la finalitat d'iniciar una intervenció i gestió primerenques amb l'augment de l'atenció d'infermeria i l'activació d'un equip de resposta ràpida o emergències mèdiques. Els mesuraments de monitoratge que es duen a terme de forma manual es poden prendre de forma intermitent amb una freqüència protocol-litzada o, amb la presència de deterioraments puntuals de pacients que requereixin un control extra. Així i tot, aquests mesuraments protocol-litzats es veuen influenciats per la durada del torn clínic, els nivells de personal de planta i la càrrega de treball d'aquest. És per això que en diversos casos no es duen a terme de forma rigorosa, fet que pot comportar a conjunts incomplets d'observacions conduint a més endarreriments en el reconeixement del deteriorament d'un patient[8].

A l'hora de crear un sistema de monitoratge continu no només s'han de tenir en compte els aspectes tècnics del qual ha de constar, sinó que també s'ha de tenir en compte l'entorn d'aquest sistema i l'opinió personal dels professionals de la salut i dels pacients sobre el qual s'ha d'implantar, per veure la seva viabilitat i detallar certs aspectes del sistema, els quals amb la informació tècnica no n'hi ha prou. És per això que s'ha analitzat el següent article que parla de l'experiència dels professionals de salut i dels pacients després de la implantació d'un sistema de monitoratge continu dels signes vitals dels pacients a un hospital del Regne Unit[9].

2.2.1. Anàlisis de l'article

El monitoratge continu és una alternativa als mesuratges intermitents i és vista pel personal com una eina positiva per donar suport a la identificació primerenca del deteriorament dels pacients, sempre que no hi hagi falses alertes.

L'objectiu d'aquest estudi és perfeccionar i integrar els sistemes de monitoratge ambulatori pel seu ús a la pràctica clínica. És per això que es vol comprendre les percepcions actuals dels pacients i personal sanitari sobre les pràctiques de monitoratge actual així com la viabilitat de l'ús del monitoratge continu ambulatori a l'entorn hospitalari.

L'estudi està basat en entrevistes individuals a pacients i infermeres per conèixer la seva opinió al respecte. Aquestes entrevistes van ser portades a terme per dues infermeres titulades. Cap de les entrevistadores tenia relacions prèviament establertes amb els participants. Durant la recollida i ànalisis de dades, l'equip d'investigació va discutir les suposicions prèvies o els prejudicis relatius a les preguntes de la investigació. Els investigadors també van proporcionar un relat reflexiu de cada entrevista com a part del formulari, incloent-hi observacions sobre la relació, l'entorn de l'entrevista i qualsevol biaix potencial. L'enfocament de l'equip per analitzar i desenvolupar temes també va ajudar a reduir els biaixos[9].

Es va entrevistar una mostra intencional de trenta participants (15 infermeres clíniques i 15 pacients) entre juliol de 2018 i 2019.

A l'article es presenten tres temes principals identificats a través de l'anàlisi temàtica. El primer punt consisteix en les dades de les constants vitals com evidència per la intensificació i explora com les infermeres van descriure l'ús de les dades de les constants vitals per informar i defensar la seva decisió d'intensificar el control dels pacients. El segon punt, és sobre la fiabilitat de les dades de les constants vitals, on es discuteix la importància de la confiança en les dades, i la preocupació per l'exactitud de les lectures dels signes vitals fetes de forma automatitzada. L'últim tema fa referència a la cerca de l'equilibri entre el monitoratge continu i l'intermitent. Aquest últim punt es va subdividir en tres subtemes: la reducció del contacte amb el personal sanitari en vers la seguretat del pacient, els aspectes negatius del monitoratge continu i el monitoratge individualitzat del pacient. Al final de les entrevistes s'inclou el concepte de monitoratge ambulatori i s'informa dels resultats a la secció final de l'article[9].

2.2.1.1. Dades de les constants vitals com evidència per l'escalada

Les dades de les constants vitals com evidència per l'escalada en la intervenció d'un pacient va ser un tema identificat per les infermeres al discutir les experiències actuals i els desafiaments en la detecció del deteriorament i l'escalada que aquest comporta. La familiaritat amb la presentació i l'estat basal d'un pacient permet a les infermeres reconèixer fàcilment els canvis marcats en les condicions clíniques que desencadenen la seva atenció. Ja hi havia un bon coneixement de com identificar el deteriorament d'un pacient basant-se en el procés de SAT, ja que les infermeres van informar que aprofitaven els signes vitals i els SAT disponibles quan es comunicava amb els metges. Un dels reptes més comuns va semblar ser el suport a la seva preocupació quan un pacient es deteriora i no s'identifica.

El fet del monitoratge continu ajuda a la detecció primerenca d'aquest deteriorament podent intensificar el control del pacient de forma temprana[9].

2.2.1.2. Fiabilitat de les dades de les constants vitals

Aquest va ser el segon tema identificat, en el qual les infermeres van compartir els seus punts de vista sobre l'exactitud de les lectures, i la majoria van estar d'acord en tornar a comprovar manualment els mesuraments de la freqüència cardíaca, la freqüència respiratòria i la pressió arterial quan tenien dubtes sobre la fiabilitat dels mesuraments automatitzats. Aquest fenomen també es va donar en la saturació d'oxigen quan la sonda de l'orella no era fiable o la sonda del dit no estava ben col·locada. Per altra banda, no es va compartir cap preocupació pel mesurament de la temperatura. Finalment, el personal d'infermeria va indicar que la freqüència respiratòria automatitzada (derivada de l'electrocardiograma) era la menys fiable i el personal clínic havia de refusar aquests mesuraments[9].

2.2.1.3. Equilibri sobre el monitoratge continu i intermitent

- Reducció del contacte d'infermeria davant la seguretat del pacient

Els pacients van reforçar la importància del contacte d'infermeria i com les observacions manuals dels signes vitals creen una oportunitat per la interacció humana. Les infermeres també van fer èmfasis en la importància del contacte cara a cara per permetre l'avaluació visual del pacient i es van trobar amb el fet que aquesta era una valuosa oportunitat per detectar canvis clínics. Certs pacients també van manifestar que la falta de personal i les demandes de cada unitat contribueixen a una freqüència de monitoratge manual subòptima endarrerint potencialment la detecció del deteriorament físic dels pacients. En aquest cas es va indicar que el monitoratge continu podria ser beneficiós tant pels pacients com pel personal sanitari augmentant l'eficiència i la seguretat[9].

- Aspectes negatius del monitoratge continu

En aquest cas es va indicar que no tots els pacients requereixen un monitoratge continu i que la decisió sobre el tipus de monitoratge no només comprèn les necessitats individuals del pacient, sinó també de l'entorn de la unitat. A part, també es va fer menció de la contaminació acústica creada pels monitors, ja que els pacients consideraven frustrant i angoixós el soroll d'aquestes alarmes, fet que provoca interrupcions del son[9].

Per altra banda, els pacients van comentar que les infermeres desatenien les alarmes de forma inconscient a causa de la contaminació acústica generant fatiga a les alarmes i fent que no s'hi presti atenció. Es va trobar interessant el fet de poder establir una personalització sobre les alarmes per cada pacient de forma individualitzada per així evitar falses alarmes[9].

- Monitoratge individualitzat pel pacient

Els dos grups entrevistats van proporcionar opinions mixtes al respecte, ja que si bé es va percebre que el monitoratge continu millora la seguretat i la tranquil·litat del pacient, al mateix temps que dona suport en la gestió de la càrrega de treball i la prioritització d'altres

activitats o pacients, també pot ser restrictiva, perquè es redueix la independència del pacient i això pot causar angoixa a causa del cablejat que aquest ha de portar i la poca mobilitat lliure que li deixa. D'aquesta reflexió en va sortir l'últim apartat, el qual fa referència a la complexitat de la introducció de dispositius de mesurament ambulatoris que siguin còmodes i segurs pels pacients i aportin la mateixa informació[9].

2.2.2. Dispositius portàtils ambulatoris

Aquest tema es va introduir com a idea per tal d'explorar les percepcions de com això pot influir en les pràctiques de monitoratge. Tant les infermeres com els pacients van veure els beneficis dels sistemes de monitoratge ambulatori generant expectatives positives amb aquests per permetre una major mobilitat i una millor tolerància, al mateix temps que es promou la detecció primerenca del deteriorament de la salut a través del monitoratge continu ambulatori[9].

Els dos grups van arribar a la conclusió que aquests sistemes han de donar suport al personal sanitari, però en cap cas substituir-lo perquè la tecnologia visible en els entorns hospitalaris es troba, en la seva major part, en fase de viabilitat, la retroalimentació clínica és crucial per la seva implementació exitosa en aquests entorns[9].

2.3. Competència actual

Actualment, no és molt comú veure aquest tipus de dispositius o sistemes implantats als hospitals. Així i tot, no vol dir que no hi hagi empreses o projectes destinats a aquest objectiu. Tal com s'ha vist anteriorment, ja s'han inventat les xarxes de sensors WBAN amb una finalitat molt similar a la d'aquest projecte i en l'actualitat hi ha molts estudis per veure la viabilitat d'aquests sistemes en entorns hospitalaris. Sense anar més lluny, avui en dia molta gent porta un dispositiu portàtil al seu canell capaç de realitzar les mesures de les constants vitals i enviar-les a una plataforma virtual, sigui una web o una aplicació mòbil, per veure-les i extreure'n informació. Simplement, s'ha d'importar aquest sistema en un entorn hospitalari per tenir un sistema similar al que s'ha creat en aquest treball[2].

Val a dir, que hi ha una empresa anomenada Oxeen que ha dissenyat un sistema amb un dispositiu molt similar a aquest. Els sistemes per això es diferencien en algunes coses. En el seu cas, es requereix un dispositiu extra encarregat de trametre les dades al servidor, i no és la mateixa polsera la que ho fa. Aquest és el disseny que es tenia pensat primerament quan es va pensar en VITALS, però per diferenciar-se de la competència, es va decidir suprimir aquest aparell del sistema i que fos la mateixa polsera la que enviés les dades al servidor. Aquesta decisió té parts bones i dolentes.

L'avantatge de no requerir un dispositiu extra que es connecti via Bluetooth a la polsera i posteriorment enviï les dades, és la possibilitat de mobilitat que té el pacient sense la necessitat de requerir aquest aparell per poder-se connectar al servidor.

La part negativa és que en ser un dispositiu portàtil i requerir bateria, el consum de la connexió Wi-Fi és més elevat que el consum de la connexió Bluetooth, i, per tant, si no es fa una bona gestió d'aquesta, és molt possible que el dispositiu de VITALS tingui una vida més curta que el dispositiu d'Oxeen.

Val a dir que aquesta empresa és una empresa emergent que fa dos anys que està activa, i compta amb un equip de treball del voltant de 10 persones. Per tant, és complicat arribar al seu nivell amb un projecte de final de grau.

3. Requeriments i arquitectura del projecte VITALS

3.1. Requeriments del sistema

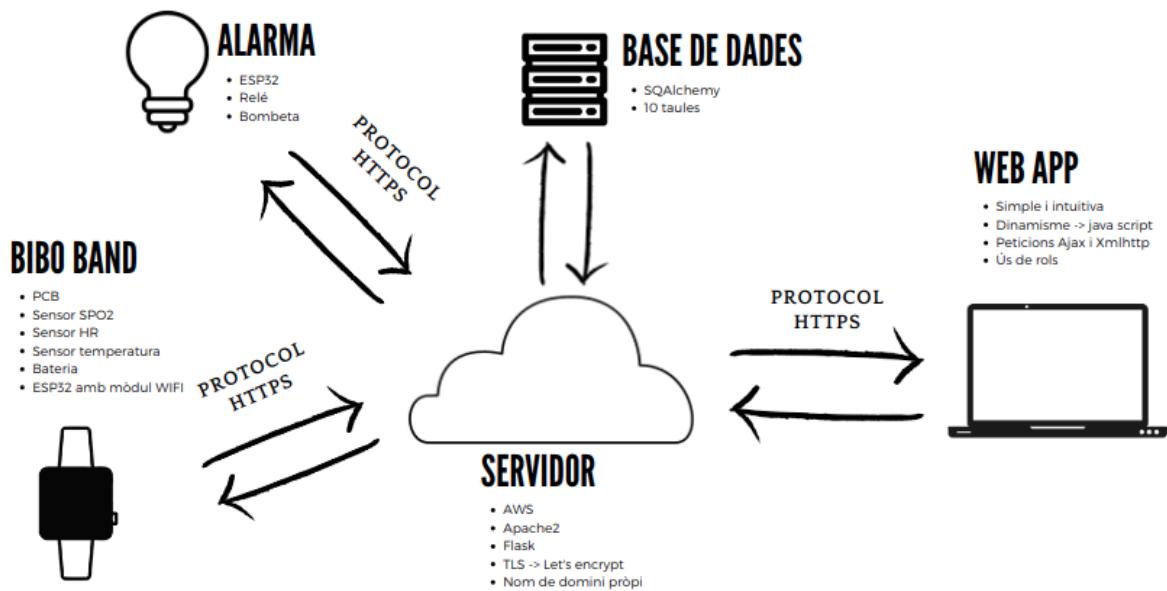
Aquest projecte pretén donar una solució a totes les necessitats i requeriments vistos al llarg del capítol anterior. Gràcies a l'anàlisi de tota la informació esmentada, s'han establert uns objectius o unes característiques que el projecte VITALS ha de complir. Aquests requeriments són els següents.

- Implementació de sensors fiables encarregats d'obtenir les dades de freqüència cardíaca, saturació d'oxigen i temperatura corporal.
- Necessitat d'un dispositiu per tal de donar accés a la xarxa a totes les mesures realitzades pels sensors.
- Consum energètic mínim del dispositiu.
- Protocol de comunicació segur entre el dispositiu emissor i el receptor.
- Emmagatzematge de les dades a un servidor accessible únicament amb clau pública/privada.
- Establiment de rols dins la plataforma pels diferents professionals que en facin ús.
- Fàcil utilització de la plataforma pels professionals que la utilitzin.
- Sistema d'alarmes personalitzat per cada pacient.
- Alarmes que no generen contaminació acústica.
- Sistema sense fils i còmode.

La solució VITALS ha de complir amb tots aquests requisits per poder ser una solució que faci front a la competència i que compleixi amb la demanda actual de com ha de ser un sistema d'aquestes característiques. És per això que s'ha treballat al màxim possible per tal que tots aquests requisits siguin la base d'aquest projecte. El projecte inclou tots els temes que s'han tractat al llarg del grau d'enginyeria de sistemes TIC, i, per tant, és el millor exemple per posar en marxa tot el que s'ha après durant aquests quatre anys.

3.2. Arquitectura del sistema

Un cop establerts els requisits del sistema, s'ha creat el model de com ha de ser aquest. A l'esquema de la Il·lustració 1 es poden veure els diferents components que han de formar el sistema i les comunicacions d'aquests per complir els diferents requisits. A continuació es detallaran tots els elements de l'esquema i les característiques que han de tenir per fer front a la competència i aconseguir el sistema òptim i adequat a les dades analitzades anteriorment.



Il·lustració 1: Esquema del projecte VITALS

3.2.1. Bibo Band

És la base del sistema, consisteix en el dispositiu que conté els sensors i el microcontrolador. Aquest últim s'utilitza pel tractament de les dades rebudes per part dels sensors així com, per poder donar accés a internet i poder comunicar aquestes mesures cap a l'exterior. Com s'ha mencionat, els sensors que componen aquesta polsera intel·ligent són: sensor de temperatura, sensor de freqüència cardíaca i sensor de saturació d'oxigen. Els requisits que ha de complir aquest dispositiu són els següents.

- Sistema sense fils i còmode

El dispositiu és sense fils i el més petit possible a través del disseny d'una PCB amb la integració de tots els components i sensors dins seu i amb el disseny d'una carcassa el màxim ergonòmica i còmode pels pacients.

- Implementació de sensors fiables encarregats d'obtenir les dades de freqüència cardíaca, saturació d'oxigen i temperatura corporal

S'han escollit sensors fiables dins el mercat actual. Per obtenir una millor fiabilitat en les dades, els sensors de freqüència cardíaca i saturació d'oxigen se situen al dit índex del pacient i aquest es connecta amb un cable fins a la polsera. Aquesta és la solució que permet aconseguir les lectures més fiables a través dels sensors adquirits. Per altra banda, el sensor de temperatura està a la mateixa polsera situada al canell del pacient.

- Necessitat d'un dispositiu per tal de donar accés a la xarxa a totes les mesures realitzades pels sensors

El microcontrolador encarregat de governar els diferents sensors que componen la Bibo band és un ESP32 wroom 32 amb el mòdul Wi-Fi integrat que permet l'accés a internet i la comunicació amb l'exterior.

- Consum energètic mínim del dispositiu

Les mesures són realitzades en intervals de temps personalitzats. Durant els períodes de temps que el microcontrolador no estigui fent mesures, aquest està en el mode *Deep Sleep* el qual és l'estat de mínim consum d'energia que pot mantenir el dispositiu activat de forma simultània.

3.2.2. Sistema d'alarma

És la part que permet complir un dels principals objectius del sistema. El sistema d'alarma principalment s'utilitza per poder generar l'estat d'alarma en cas que les lectures realitzades per la Bibo band estiguin fora dels llindars configurats per pacient. Això activa una alarma lluminosa que es pot gestionar des de la plataforma web. Els requisits que ha de complir aquesta part del sistema són els següents.

- Sistema d'alarmes personalitzat per cada pacient

Tal com s'ha comentat, els sensors mesuren la temperatura corporal, la saturació d'oxigen i la freqüència cardíaca. De forma genèrica, en introduir un nou pacient, aquest se li posa per defecte els valors llindar d'alarma estàndard per la població mundial. Així i tot, la plataforma permet modificar els valors llindar de cada pacient perquè es generin les alarmes de forma individualitzada per cada configuració realitzada.

- Alarmes que no generen contaminació acústica

El sistema d'alarma que es porta a terme consisteix en un senyal lluminós que s'activa quan es genera alguna alarma dins de cada unitat mèdica. Aquesta alarma no causa contaminació acústica, ja que simplement emet un senyal lluminós que es pot desactivar en el moment de la confirmació de visibilitat de l'alarma des de la plataforma web.

3.2.3. Servidor

El servidor està en una màquina virtual d'Amazon (AWS) i és la que conté tota la lògica del sistema i l'encarregada de rebre i respondre a totes les peticions que li arribin, ja sigui dels dispositius físics com de la plataforma web. Aquest servidor conté també la base de dades. Aquesta base de dades és el nucli del sistema on es gestionen totes les peticions a

través de fer consultes a ella i modificar la informació del seu interior. El principal requisit que ha de complir és el següent.

- **Emmagatzematge de les dades a un servidor accessible únicament amb clau pública/privada.**

Al servidor només s'hi pot accedir a través del xifratge de clau pública/privada, així que la seguretat de les dades és total i l'accés a elles és totalment restringit per aquelles persones que no tenen accés a la clau privada.

3.2.4. Plataforma web

- **Establiment de rols dins la plataforma pels diferents professionals que en facin ús.**

Dins la plataforma els usuaris es divideixen en 3 rols diferents, el primer d'ells és el rol administratiu, on simplement tindrà la capacitat de poder inscriure pacients i assignar-los a les Bibo bands. El segon rol és el de professional sanitari, el qual tindrà accés a les mateixes funcionalitats que el rol administratiu, i a més a més pot accedir al panell de pacients i a la gestió d'aquests, als històrics i a la gestió de les alarmes. Finalment, el rol d'administrador de sistema, és aquell que té accés a totes les funcionalitats vistes fins al moment i també a la gestió dels diferents usuaris de la plataforma.

- **Fàcil utilització de la plataforma pels professionals que la utilitzin**

La plataforma és molt senzilla d'utilitzar i, les gestions més complexes, com pot ser l'enllaç de Bibo bands amb pacients es fan amb un simple botó. A part, la plataforma és visualment molt fàcil d'entendre i es pot accedir a les mateixes funcionalitat des de diferents pantalles, fet que facilita la interacció amb la plataforma sense necessitat de memòria visual per poder navegar i interaccionar amb ella.

3.2.5. Comunicació del sistema

Totes les parts de l'esquema vistes fins al moment requereixen una comunicació fluida i segura entre elles, és per això que la comunicació del sistema es realitza sota el protocol HTTPS per poder garantir la confidencialitat i el xifratge de les dades de forma segura i que aquestes no puguin ser interceptades per tercers gràcies al xifratge de clau pública i privada de les dades i als certificats digitals.

4. Perifèrics

El projecte Vitals compta amb dos tipus de perifèrics, el primer de tots és la Bibo band. Aquesta és el dispositiu encarregat de la mesura de les constants vitals dels pacients ingressats a planta i la posterior comunicació d'aquestes dades al servidor central per tal de poder fer el tractament adequat de les dades.

El segon perifèric és el dispositiu d'alarmes. Aquest és l'encarregat d'emetre un senyal lluminós quan es detecta que algun dels valors de les constants vitals dels pacients no està dins el llindar configurat per cada un d'ells. A continuació es farà una descripció de cada una de les seves parts.

4.1. Bibo band

Com ja s'ha comentat anteriorment, és el dispositiu intel·ligent que conté els diferents sensors i és l'encarregat de realitzar les mesures en temps real de les constants vitals dels pacients. Aquest dispositiu s'ha dissenyat per tal que compleixi els requeriments vists a l'apartat anterior. En un inici el dispositiu havia de ser únicament de canell, i les lectures de les constants s'havien de fer des d'aquest, posteriorment i al veure que les lectures de la freqüència cardíaca i la saturació d'oxigen no eren del tot acurades, es va pensar un disseny alternatiu on aquestes mesures es farien al dit i, a través d'un enllaç físic, es passarien al microcontrolador situat al canell. S'ha intentat que el dispositiu sigui el més petit i còmode possible amb el mínim consum de bateria.

A continuació s'analitzaran els diferents aspectes que formen part d'aquest dispositiu, des dels components que en formen part, la programació del software, el disseny i fabricació de la PCB i l'encapsulat final.

4.1.1. Hardware

La Bibo band es compon principalment d'un microcontrolador i 2 sensors. La resta de components utilitzats en el disseny de la PCB s'explicaran més explícitament en el capítol referent a aquesta.

4.1.1.1. *ESP32*

L'ESP32 és una sèrie de microcontroladors de sistema en un xip de baix cost i baix consum que compta amb connexió Wi-Fi i Bluetooth de forma dual i integrat. La sèrie d'aquests microcontroladors utilitza un microprocessador Tensilica Xtensa LX6 en les seves variants de doble nucli i d'un sol nucli, un microprocessador Xtensa LX7 de doble nucli o un microprocessador RISC-V d'un sol nucli així com commutadors d'antena integrats, com el de radiofreqüència, amplificador de potència, amplificador de recepció de baix soroll, filtres i mòduls de gestió d'energia. L' ESP32 ha estat creat i desenvolupat per Espressif Systems i és fabricat per TSMC[10].

Per aquest projecte s'ha fet servir en concret el mòdul ESP32-WROOM-32 pel fet que és un potent mòdul MCU genèric amb Wi-Fi, Bluetooth i Bluetooth LE que es dirigeix a una àmplia varietat d'aplicacions, que van des de les xarxes de sensors de baixa potència fins a les tasques més exigents, com la codificació de veu, la transmissió de música i la descodificació MP3. En aquest cas, aquest mòdul conté un nucli que és el xip ESP32-D0WDQ6. El xip està integrat en el disseny per ser escalable i adaptable. Disposa de dos nuclis de CPU que poden controlar-se individualment, i la freqüència de rellotge de la CPU és ajustable de 80 MHz a 240 MHz. El xip també compta amb un coprocessador de baix consum que pot usar-se en lloc de la CPU per estalviar energia mentre es duguin a terme tasques que no requereixen molta potència de càlcul[10].

La integració del Wi-Fi garanteix que es pugui apuntar a una àmplia gamma d'aplicacions i que el mòdul sigui polivalent. Permet un gran abast físic i la connexió directa a Internet a través d'un encaminador Wi-Fi. El mòdul admet una velocitat de dades de fins a 150 Mbps i una potència de sortida de 20dBm a l'antena per garantir l'abast físic més gran possible. El corrent en repòs del xip és inferior a 5 uA, el que el fa adequat per aplicacions alimentades per bateria i d'electrònica portàtil[10].

El sistema operatiu de l'ESP32 és el FreeRTOS. La característica principal d'aquest sistema operatiu és que és multitasca. Això vol dir que pot fer diverses tasques intercalant-les i gestionant-les amb el temps. El problema és que no permet la multitasca real, és a dir, si una tasca s'executa durant un període de temps, es bloquejaran les altres tasques durant aquest període de temps fins que no acabi la tasca que s'està executant[11].

El fet que aquest sistema operatiu sigui multitasca és per la capacitat de partir les tasques en unitats molt més petites, fet que permet intercalar-les a un nivell més baix aprofitant així els temps morts entre elles[11].

Després de veure totes les especificacions i característiques que presenta aquest microcontrolador, s'ha cregut que és el més adequat pel sistema que es vol implementar. El fet que porti incorporat el mòdul Wi-Fi, permet l'accés a internet de les dades sense la necessitat d'un dispositiu amb accés a internet el qual rebi les dades via Bluetooth i les pugui transmetre a la xarxa, tal com estava pensat el sistema en un principi. El fet de no requerir aquest dispositiu permet la transmissió de dades de forma autònoma des del dispositiu de canell.

4.1.1.2. MAXREFDES117

Per la mesura de la freqüència cardíaca i la saturació d'oxigen s'ha utilitzat el mòdul òptic de baixa potència MAXREFDES117 del fabricant Maxim Integrated. Aquest mòdul compta amb el MAX30102 que inclou LEDs vermellos i infrarojos integrats per la detecció de la freqüència cardíaca i la saturació d'oxigen en sang. Aquesta configuració detecta idealment la freqüència cardíaca i la saturació d'oxigen a la cara anterior de la falange distal dels dits,

el lòbul de l'orella o qualsevol altra extremitat carnosa d'una persona. Les mesures de la placa són 12,7 mm x 12,7 mm i la fan ideal per aplicacions portàtils[12].

En aquest cas concret, la placa fa servir com a sensor de ritme cardíac i saturació d'oxigen el MAX30102, un convertidor reductor de baix consum MAX1921 i un precís traductor de nivell MAX14595. Tot el disseny funciona a menys de 5,5 mW de potència i l'usuari li ha de proporcionar un subministrament de tensió de 2 V a 5 V a l'entrada de l'alimentació, fet que el fa perfecte per qualsevol bateria o placa de sistema encastat. La comunicació amb el microcontrolador és a través del protocol I2C[12].

El sensor MAX30102 consta d'un parell de LEDs d'alta intensitat (vermell i infraroig, els dos amb diferent longitud d'ona) i un fotodetector. Les longituds d'ona d'aquests LEDs són de 660 nm i 880 nm respectivament. El sensor funciona fent brillar els dos LEDs sobre el dit o el lòbul de l'orella, de manera que les dues llums puguin penetrar fàcilment al teixit. Tot seguit es mesura la quantitat de llum reflectida mitjançant un fotodetector. Aquest mètode de detecció a través de llum s'anomena fotopletismograma[13].

4.1.1.2.1. Detecció de la freqüència cardíaca i la saturació d'oxigen

- **Freqüència cardíaca**

La freqüència cardíaca és el nombre de vegades que es contrau el cor durant un minut. Pel correcte funcionament de l'organisme, és necessari que el cor actuï bombant la sang cap a tots els òrgans, però a més a més ho ha de fer a una determinada pressió i a una determinada freqüència[14].

La freqüència cardíaca normal en repòs oscil·la entre 50 i 100 pulsacions per minut. Tot i això, hi ha diferents aspectes que s'han de tenir en compte[14].

- A partir del primer més de vida la freqüència cardíaca baixa fins a arribar a l'edat adulta que és quan s'estabilitza (20 anys).
- Varia al llarg del dia i la nit en resposta a diferents estímuls, així que el seu mesurament té gran variabilitat.
- Al realitzar exercici físic el cor produeix una resposta normal que s'anomena taquicàrdia fent que la freqüència cardíaca superi les 100 pulsacions per minut.
- La bradicàrdia es produeix quan la freqüència cardíaca és inferior a 50 pulsacions per minut.

Per calcular la freqüència cardíaca s'utilitza principalment el LED infraroig. El funcionament és el següent: l'hemoglobina oxigenada (HbO_2) de la sang arterial té la característica d'absorbir la llum infraroja. Com més vermella és la sang (com més alta és la concentració d'hemoglobina), més llum infraroja s'absorbeix.

A mesura que la sang és bombejada a través del dit amb cada batec del cor, la quantitat de llum reflectida canvia creant una forma d'ona canviant a la sortida del fotodetector. A mesura que es continua emetent la llum i prenent lectures del fotodetector, es comença a obtenir ràpidament una lectura del pols dels batecs del cor o freqüència cardíaca[15].

- **Saturació d'oxigen**

La saturació d'oxigen en sang o també coneguda com a SpO₂ és una mesura de la quantitat d'hemoglobina que transporta oxigen a la sang amb relació a la quantitat d'hemoglobina que no en transporta. El cos necessita que hi hagi un determinat nivell d'oxigen a la sang pel seu correcte funcionament. Els nivells baixos de SpO₂ poden provocar símptomes molt greus. Aquesta condició s'anomena hipoxèmia provocant un efecte visible a la pell conegut com a cianosis a causa del color blau que aquesta adquireix. La hipoxèmia pot portar a la hipòxia (nivell baix d'oxigen als teixits) provocant la mort d'aquests[16].

Es considera que el percentatge adequat i saludable d'oxigen a la sang és entre el 95% i el 100%. En el moment que la SpO₂ cau per sota del 90% és quan es parla d'hipoxèmia. Nivells inferiors al 80% es considera hipoxèmia greu[17].

Per calcular els nivells d'oxigen en sang, s'utilitza el principi en que la quantitat de llum vermella i els infrarojos varien en funció de la quantitat d'oxigen en sang. Com s'ha comentat, l'espectre d'absorció de l'hemoglobina oxigenada i de l'hemoglobina desoxigenada és diferent. És gràcies al fet que la sang desoxigenada absorbeix més llum vermella mentre que la sang oxigenada absorbeix més llum infraroja. Fent el càlcul de la proporció reflectida de llum vermella i llum infraroja rebuda pel fotoreceptor que es pot calcular el nivell d'oxigen en sang[18].

4.1.1.3. MAX30205

El sensor de temperatura escollit per a la mesura de la temperatura corporal en aquest projecte ha estat el MAX30205 del fabricant Maxim Integrated. Aquest sensor mesura amb precisió la temperatura corporal a través del seu convertidor analògic-digital sigma-delta d'alta resolució. La precisió compleix amb l'especificació de la termometria clínica d'ASTM E1112 quan es solda a una placa de circuit imprès, com és el cas d'aquest projecte. La comunicació es realitza a través del protocol I₂C. El sensor compta amb tres línies de selecció de direcció amb un total de 32 direccions disponibles. Té un rang de tensió d'alimentació d'entre 2,7 V i 3,3 V fet que el fa perfecte pel microcontrolador escollit. El corrent d'alimentació del sensor és de 600 uA[19].

- **Temperatura corporal**

La temperatura corporal canvia segons la persona, l'edat, les activitats i el moment del dia. Es considera com a temperatura mitjana corporal normal generalment és 37 °C. Alguns

estudis per això indiquen que la temperatura corporal pot estar dins un rang que és 36.1 °C i 37.2 °C. Una temperatura superior a 38 °C sempre indica que la persona té febre[20].

Les lectures de temperatura es fan a través de senyals elèctrics. El sensor conté dos metalls que generen una tensió o resistència elèctrica quan es produueix un canvi de temperatura. Aquests mesuren la tensió entre els terminals del diòde. Quan la tensió augmenta, la temperatura també s'incrementa, el que es tradueix en una caiguda de tensió entre els terminals del transistor i l'emissor[21].

4.1.2. Software

4.1.2.1. Eina de programació utilitzada

Per dur a terme la programació del microcontrolador, s'ha utilitzat l'entorn de desenvolupament integrat anomenat Arduino IDE. Aquest és un programa informàtic compost per un conjunt d'eines de programació. En ell s'hi pot trobar un editor de codi, un compilador, un depurador i un constructor d'interfície gràfica. A part, també té les eines per poder carregar el programa compilat a la memòria flash del hardware en qüestió[22].

Aquest IDE permet la programació d'una varietat àmplia de hardware com és el cas de l'ESP32. És per això que s'ha usat aquest programari gràcies a la facilitat de programació i l'existència de totes les llibreries disponibles per cada un dels sensors utilitzats en aquest projecte. Així i tot, per poder utilitzar un software com aquest en un ESP32, primer s'han de realitzar uns passos previs d'instal·lació tal com es mostra a continuació[22].

- 1- Descarregar Arduino IDE des de la plataforma oficial.
- 2- Obrir el programari i seleccionar Fitxer -> Preferències.
- 3- Introduir la següent URL a l'espai disponible d'URL de gestor de plaques:
https://raw.githubusercontent.com/espressif/arduino-esp32/gh-pages/package_esp32_index.json.
- 4- Seleccionar Eines -> Placa “Arduino Uno” -> Gestor de plaques.
- 5- Al cercador del gestor de plaques introduir ESP32.
- 6- Instal·lar l'opció de “ESP32 by Espressif Systems”.

4.1.2.2. Llibreries utilitzades

Per a la gestió dels diferents sensors i la comunicació del dispositiu amb el servidor s'han utilitzat llibreries externes. A continuació es farà una petita descripció de cada una d'elles.

- Llibreria Wire.h

Aquesta llibreria permet la comunicació a través del protocol I2C. Els pins encarregats d'aquesta comunicació són el D21 corresponent al SCL i el D22 corresponent al SDA[23].

- Llibreria MAX30105.h

Aquesta llibreria és l'encarregada de la gestió del MAXREFDES117. Principalment, s'ha usat per fer el set-up de la placa de desenvolupament per poder utilitzar posteriorment el sensor pel càlcul de la freqüència cardíaca i la saturació d'oxigen. Els càlculs de les mesures s'han fet amb les llibreries que es veuran a continuació.

- Llibreria heartRate.h

Com el seu nom indica, és l'encarregada del càlcul de la freqüència cardíaca a través de les mostres obtingudes pel sensor. Aquest càlcul el fa a través de la funció *particleSensor.getIR()* la qual retorna el valor del sensor d'infrarojos, tot seguit es realitzen un seguit de càlculs i s'obté el valor de la freqüència cardíaca presa en un moment determinat. Després de dur a terme un seguit de proves i veure que a l'hora de prendre mostres, les primeres no són valides, s'ha optat per eliminar les 100 primeres mostres calculades per aconseguir valors més reals. Un cop eliminades les 100 mostres, es calcula la freqüència cardíaca mitjana d'un total de 500 mostres. D'aquesta manera s'assegura que el resultat sigui el més acurat possible.

- Llibreria spo2_algorithm.h

Aquesta llibreria conté les funcions necessàries per poder calcular la saturació d'oxigen en sang a través de la utilització del LED vermell i el LED infraroig. El càlcul el fa a través de les funcions *particleSensor.getRed()* i *particleSensor.getIR()*. D'aquesta manera, fent els càlculs adients a través del valor dels dos LEDs es pot aconseguir el valor de la saturació d'oxigen. Igual que en el càlcul de la freqüència cardíaca, també s'eliminen les 100 primeres mostres per aconseguir resultats més fidels a la realitat. El resultat és el càlcul de la mitjana d'un total de 150 mostres. A diferència del cas anterior, s'agafen menys mostres degut a l'elevat temps de càlcul que es requereix per la mesura d'una mostra.

- Llibreria WiFi.h

Aquesta llibreria és la que permet la gestió de la connectivitat Wi-Fi. Dins el codi final, aquesta llibreria ha servit per realitzar les funcions de: saber l'estat de la connexió, el mode del Wi-Fi, establiment d'una connexió Wi-Fi a través d'un SSID i la corresponent contrasenya i la forçada desconexió d'una xarxa Wi-Fi[24].

- Llibreria HTTPClient.h

Aquesta llibreria és la que permet utilitzar el protocol HTTP i HTTPS per les connexions contra el servidor. Els mètodes que permet aquesta llibreria són el de GET, POST, PUT i DELETE. En aquest cas només s'usarà el mètode PUT a través del protocol HTTPS[25].

- Llibreria ArduinoJson.h

Aquesta llibreria ha servit per poder serialitzar les dades que s'envien al servidor en format JSON i poder desserialitzar les dades rebudes del servidor en format JSON i així poder fer el tractament adequat d'aquestes[26].

- Llibreria driver/rtc_io.h

Aquesta llibreria s'ha usat per poder activar en estat alt el pin RTC i que aquest no passi a un estat baix en el moment que el microcontrolador entra en mode *Deep Sleep*. D'aquesta manera es permet la retroalimentació del microcontrolador encara que aquest estigui en el mode d'estalvi d'energia[27].

- Llibreria Protocentral_MAX30205.h

Aquesta llibreria és la que s'encarrega de gestionar el sensor de temperatura MAX30205. El càlcul de la temperatura és el més senzill dels 3 càlculs de les constants vitals. En aquest cas no es llencen mostres i el càlcul de la temperatura es fa a través de la mitjana de 25 mostres. S'ha escollit aquest nombre de mostres després de realitzar varíes proves i veure que tots els valors obtinguts són molt similars.

- Llibreria Preferences.h

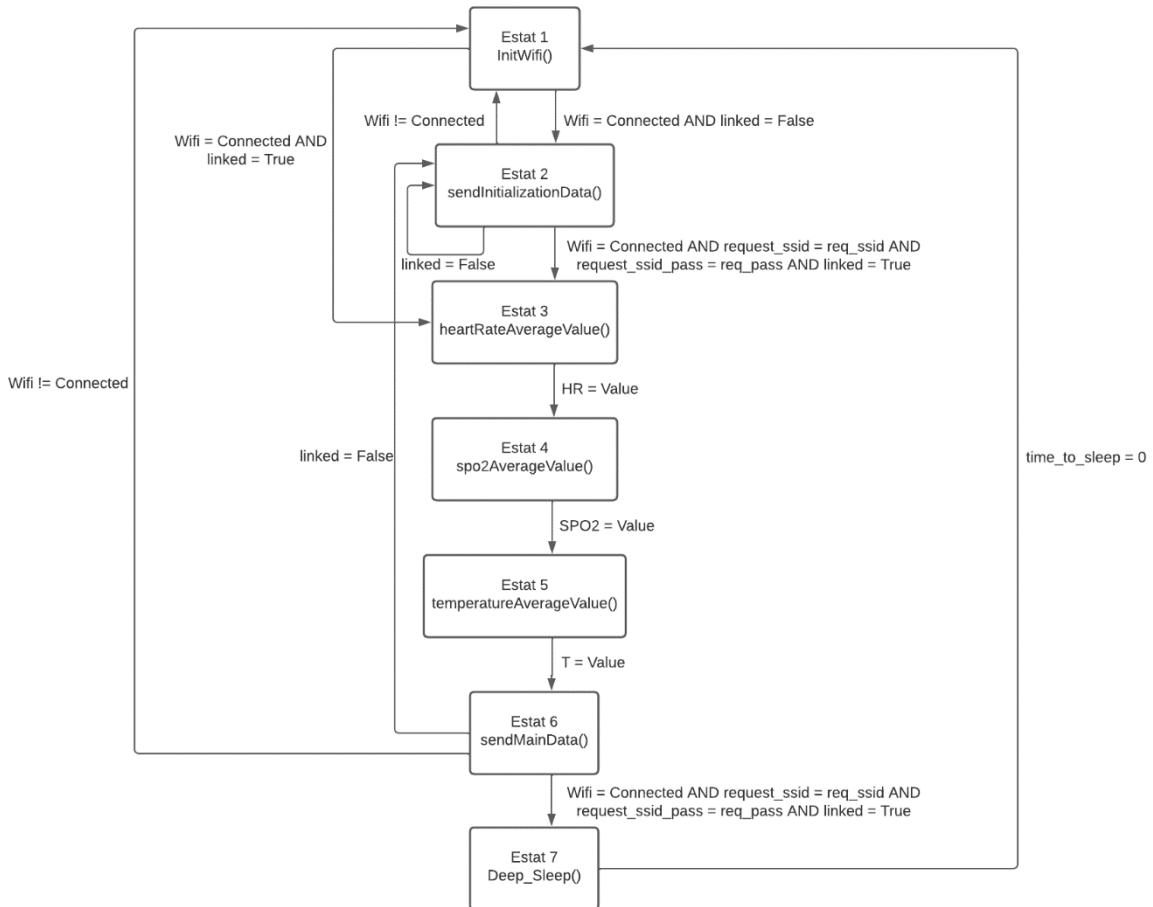
Aquesta llibreria s'ha utilitzat per a la gestió de la connectivitat Wi-Fi. Tal com es veurà més endavant, la connexió de la xarxa Wi-Fi es realitza a través d'un mecanisme condicional. Aquesta mecanisme ha de consultar constantment un fitxer guardat a la memòria Flash del dispositiu on hi ha emmagatzemades les 2 últimes xarxes on s'ha connectat. Aquesta llibreria permet l'escriptura i modificació de dades d'aquest fitxer de la memòria Flash d'una manera fàcil i intuitiva. El dispositiu té la necessitat de guardar dades a la memòria permanent a causa de ser un dispositiu portàtil amb bateria, ja que és molt fàcil que aquesta s'acabi o simplement aquest s'aturi manualment. Per tant, es perdria la xarxa a la qual s'ha de connectar una vegada es torni a engegar. La gestió d'aquest fitxer es fa realment senzilla[28].

Les funcionalitats utilitzades són les d'inicialitzar el fitxer i emmagatzemar i consultar informació amb tres simples funcions: *preferences.begin()* per la inicialització del fitxer, *preferences.getString()* per llegir la informació i *preferences.putString()* per escriure-la.

4.1.2.3. Estructura general del codi

El codi o funcionalitat del microcontrolador funciona a través de la màquina d'estats de la Il·lustració 2. Aquesta es gestiona a través de la funció *machineState()*, la qual es crida cada vegada que es fa el bucle principal. D'aquesta manera a cada cicle del bucle, es mira si s'ha de canviar l'estat per posteriorment fer l'acció de l'estat corresponent en qüestió. S'ha creat

un codi genèric amb la idea de que sigui el mateix per cada una de les Bibo bands i que aquestes es puguin configurar de forma diferent des de la plataforma web i la base de dades. A continuació es mostra la màquina d'estats de cada Bibo band.



Il·lustració 2: Màquina d'estats Bibo band

A l'estat 1 es realitza la gestió de la connexió, com es veurà en el següent punt, s'ha utilitzat un mecanisme condicional per poder gestionar correctament aquest apartat. La necessitat d'aquest mecanisme és degut al fet que la gestió de la connexió no és senzilla a causa del requeriment d'haver de guardar les 2 últimes xarxes connectades a la memòria Flash del microcontrolador, així com, tenir una xarxa escrita al codi que serà la xarxa d'emergència en cas que la connexió de les altres xarxes desaparegui de cop i no s'hagi configurat una nova xarxa des de la plataforma web. No es passarà al següent estat fins que el dispositiu no estigui connectat a una xarxa.

A l'estat 2 es fa una identificació de la Bibo band amb el servidor a través del mètode PUT. En cas que aquesta estigui ja identificada i assignada a la base de dades a un pacient, es passarà a l'estat 3, en cas que el dispositiu no estigui enllaçat a cap pacient es quedarà al mateix estat, i tornarà a fer la mateixa petició al cap de 10 segons. En cas que la resposta del servidor contingui una nova xarxa a la qual s'ha de connectar el dispositiu, es passarà altra vegada a l'estat 1 per intentar la connexió amb aquesta nova xarxa.

A l'estat 3 es realitza la mesura de la freqüència cardíaca, que un cop mesurada i sempre que el dispositiu estigui connectat a la xarxa, es passa automàticament a l'estat 4. En cas que no estigui connectat, es passa a l'estat 1.

A l'estat 4 es du a terme la mesura de la saturació d'oxigen, que un cop mesurada, i sempre que el dispositiu estigui connectat a la xarxa, es passa automàticament a l'estat 5. En cas que no estigui connectat, es passa a l'estat 1.

A l'estat 5 es porta a cap la mesura de la temperatura corporal, que un cop mesurada i sempre que el dispositiu estigui connectat a la xarxa passa a l'estat 6. En cas que no estigui connectat, es passa a l'estat 1.

A l'estat 6 es fa el PUT al servidor amb les dades de les mesures acabades de calcular. Un cop rebuda la resposta del servidor, si el dispositiu continua enllaçat al pacient, es passa a l'estat 7. En cas que el servidor retorne que aquell dispositiu no està enllaçat a cap pacient, es passarà a l'estat 2 per realitzar les identificacions. Finalment, i en cas que s'hagi configurat una nova xarxa per aquell dispositiu des de la plataforma web, es passarà a l'estat 1 per intentar dur a terme la connexió de la xarxa.

A l'estat 7 simplement es posa el microcontrolador en el mode *Deep Sleep*. Aquest mode està actiu durant el temps que hagi rebut la resposta de l'estat 6, ja que el temps de pausa entre mesures es configura a través de la plataforma web per cada polsera.

4.1.2.3.1. Gestió de la connexió

Tal com s'ha comentat a l'apartat anterior, la connexió es gestiona a través d'un mecanisme condicional i del guardat de dades a la memòria Flash del microcontrolador. En aquesta memòria Flash s'emmagatzemen un total de 4 entrades pel *ssid* i contrasenya en format de diccionari tal com es mostra a continuació.

```
{
  Legacy_ssid: legacy_pass,
  Ssid1: ssid1_pass,
  Ssid2: ssid2_pass,
  Req_ssid: req_pass
}
```

Legacy_ssid fa referència a aquella xarxa que requereixen els dispositius quan es connecten per primera vegada i poder fer la primera petició al servidor i així obtenir el *ssid* i contrasenya de la xarxa on s'han de connectar. És per això que els dispositius sempre requeriran aquesta connexió quan s'engeguin per primera vegada i facin la primera petició. Aquesta xarxa està escrita al codi de forma fixa.

Ssid1 fa referència a la xarxa a la qual està connectat el dispositiu en aquell moment, per tant, en el cas de ser la primera connexió mai feta pel dispositiu, aquesta tindrà els mateixos valors que la *legacy_ssid*. Cada vegada que s'informi al servidor de la xarxa real a la qual

està connectat el dispositiu, s'utilitzarà el valor que hi ha guardat en aquest espai de la memòria Flash.

Ssid2 fa referència a l'última xarxa en la qual ha estat connectat el dispositiu abans de connectar-se a l'actual. Aquesta es guarda per si en algun moment la connexió de la xarxa *ssid1* cau i així poder usar la xarxa *ssid2* sense problemes i no haver de requerir la xarxa *legacy ssid*.

Req ssid fa referència a la xarxa de configuració que es rep des del servidor. Aquesta és interessant guardar-la de forma permanent per tal de poder determinar si la xarxa que s'està rebent del servidor és nova o continua essent l'última en la qual s'ha realitzat la petició, ja que en el cas que aquesta no es pogués connectar per algun error tipogràfic, mantindria el dispositiu realitzant els intents de connexió cada vegada que el servidor generi una resposta.

Tots els noms que contenen la terminologia *pass* indiquen la contrasenya per l'*ssid* al qual fan referència.

Un cop vistes les nomenclatures per guardar les diferents xarxes a les quals es connecten els dispositius, és interessant veure com es gestiona aquesta connexió a través del mecanisme condicional que hi ha al codi.

- Primer de tot s'activa el fitxer de la memòria Flash de la llibreria Preferences.h
- Tot seguit es comprova si és la primera vegada que aquell dispositiu es connecta. Això es sabrà perquè el valor de la clau *ssid1* no existirà. En cas que sigui així, es realitzarà la connexió al punt d'accés *legacy ssid*. D'aquesta manera el dispositiu ja té una xarxa coneguda on fer la primera connexió. Tot seguit la xarxa *legacy ssid* es posarà com a clau i valor a *ssid1* informant que és la connexió actual del dispositiu.
- En cas que *ssid1* ja existís al fitxer, vol dir que no és la primera vegada que es fa una connexió, per tant, és possible que s'hagi modificat la connexió i s'ha de comprovar si el contingut de *req ssid* del fitxer és igual que el que s'ha rebut a les variables *requested ssid* i *requested ssid pass* corresponents a les respostes del servidor. En cas que els continguts siguin iguals, vol dir que no s'ha modificat la xarxa des de la plataforma web i no s'ha d'actualitzar a una nova xarxa. En cas que siguin diferents, vol dir que la configuració de la xarxa des de la plataforma web sí que ha canviat i, per tant, s'ha de connectar a la nova xarxa a través dels següents passos.
 1. Primer de tot, es posarà la nova xarxa de la nova configuració al fitxer de la memòria Flash, en concret al *req ssid*.
 2. Tot seguit s'intentarà connectar a aquesta nova xarxa.
 3. En cas que ho aconsegueixi, es passarà la xarxa del *ssid1* al *ssid2* i al *ssid1*, s'hi guardarà aquesta nova xarxa indicant que és la de la connexió actual
- En cas que no es pugui connectar, s'intentarà connectar de forma il·limitada a les 3 xarxes fins que una d'elles estableixi connexió. L'ordre de les xarxes en els intents de connexió és el següent.

1. Ssid1
2. Ssid2
3. Legacy ssid

4.1.2.3.2. Gestió dels sensors

Tal com s'ha vist a l'apartat de les llibreries, cada sensor compta amb la seva pròpia llibreria per tal de realitzar el càlcul de les dades corresponents. L'estrategia que s'ha seguit amb el sensor de freqüència cardíaca i saturació d'oxigen després de fer diverses proves, és la d'eliminar les primeres mesures obtingudes a causa del temps que tarden les dades llegides en estabilitzar-se i en donar valors reals i fiables.

També s'ha de tenir en compte que aquest sensor funciona únicament si està en contacte amb la pell del pacient, per tant, és fàcil que per culpa del factor humà o una mala col·locació d'aquest, es donin valors erronis. La llibreria esmentada per cada sensor ja preveu aquest possible error en la lectura de les dades i, si es detecta una lectura on el sensor no està en contacte amb la pell, aquesta genera una lectura error, com per exemple un valor de -999. Dins el codi s'ha establert un protocol on es tenen en compte aquests valors erronis per així poder informar al servidor si la col·locació de la Bibo band és correcte o no. S'ha establert que per cada 500 lectures errònies de la freqüència cardíaca o 350 lectures errònies de la saturació d'oxigen el sensor no està ben col·locat, o que simplement el pacient s'ha tret el dispositiu. Aquesta informació es passa al servidor a través del paràmetre *state* dins el missatge d'enviament de dades tal com es veurà en el protocol de comunicació. Per altra banda, s'ha establert un nombre de mesures per cada una de les constants vitals a monitoritzar per tal d'assegurar el millor resultat en el càlcul d'aquests. En el cas de la freqüència cardíaca es fa el càlcul mitjà de 500 mostres i en el cas de la saturació d'oxigen, el càlcul mitjà és sobre 150 mostres degut a l'alt cost computacional que presenta.

Pel que fa al sensor de temperatura, la llibreria no ofereix l'opció de càlcul de dades errònies, ja que el sensor funciona independentment de si està en contacte amb la pell o no. En cas que no estigui en contacte, simplement s'enviaran les dades de la temperatura ambient sense generar cap error. Després de varies proves, s'ha vist que els valors obtinguts en cada lectura del sensor son pràcticament idèntics, i per tant, s'ha establert un càlcul mitjà d'un total de 25 mostres.

S'ha de tenir en compte per això, que l'ordre en el qual es prenen les mesures està pensat d'aquesta manera pel motiu esmentat. L'únic sensor que determina si el pacient porta el dispositiu ben col·locat o no és el de freqüència cardíaca i saturació d'oxigen. És per això que sempre es realitzen primer aquestes mesures, i en cas que siguin errònies, ja no es fa la mesura de la temperatura.

4.1.2.3.3. Protocol de comunicació

Per tal de tenir un millor control i ordre sobre les dades enviades per part dels dispositius i les respistes per part del servidor, s'ha creat un protocol de comunicació entre el servidor i els diferents dispositius. Tal com s'ha vist a la màquina d'estats, és en els estats 2 i 6 quan es produeix un intercanvi de dades amb el servidor. A l'estat 2 es du a terme el protocol d'identificació dels dispositius, mentre que a l'estat 6 es trameten les dades recollides pels sensors.

- **Protocol de comunicació d'identificació**

En aquest estat, el dispositiu simplement indica al servidor que està actiu. Això ho fa a través de realitzar un PUT al servidor a la ruta indicada per aquesta petició. El missatge enviat és en format JSON i conté la següent informació:

- PUT: <https://vitalstechnology.com/identification>
- *Body*:

```
{
  "id": id,
  "ssid": ssid
  "battery": bateria
}
```

- Id: Correspon a l'id del microcontrolador, d'aquesta manera és com s'identifiquen els diferents dispositius dins la base de dades. Aquest s'obté gràcies a la funció *ESP.getEfuseMac()*.
- Ssid : És el *ssid* de la xarxa on està connectat el microcontrolador. Gràcies a això, es pot veure a la plataforma web a quina xarxa estan connectats els dispositius en temps real.
- Bateria: És el percentatge de bateria que li queda al dispositiu. Aquest es calcula just al fer l'enviament de dades a través d'un pin de lectura ADC i d'una simple regla de tres.

Un cop el servidor rep aquesta petició, respon al dispositiu amb el següent missatge en format JSON:

- Resposta:

```
{
  "linked": linked,
  "ssid": ssid,
  "ssid_pass": ssid_pass,
  "measure_time": measure_time
}
```

- Linked: És un booleà que informa si aquell dispositiu està enllaçat o no a un pacient dins la base de dades. En cas que no ho estigui, tal com es mostra a la màquina d'estats, es seguirà fent aquesta petició cada 10 segons fins que s'enllaci a un pacient.
- Ssid: Correspon al *ssid* que hi ha configurat a la plataforma web, és a dir, és el *ssid* on el dispositiu s'ha de connectar. Tal com s'ha vist a la màquina d'estats de la gestió de la connexió, correspon al *requested_ssid*.
- Ssid_pass: És la contrasenya de la xarxa que s'ha configurat des de la plataforma web, és a dir, és la contrasenya del *ssid* on el dispositiu s'ha de connectar. Tal com s'ha vist a la màquina d'estats de la gestió de la connexió, correspon al *requested_ssid_pass*.
- Measure_time: Correspon al temps que el dispositiu ha d'estar inactiu entre mesures. És a dir, el temps en què el dispositiu ha d'estar en mode *Deep Sleep* tal com es veurà en el pròxim apartat. Aquest també ve configurat des de la plataforma web.
- Protocol de comunicació d'enviament de dades

En aquest estat, el dispositiu envia les dades calculades a través de les mesures dels sensors per informar de les constants del pacient en temps real. Aquest enviament de dades es fa a través del mètode PUT, a la ruta indicada per aquesta petició, amb el següent format de dades en JSON.

- PUT: https://vitalstechnology.com/data_values
- *Body:*

```
{
  "id": id,
  "ssid": ssid,
  "state": state,
  "hr": hr,
  "spo2": spo2,
  "temperature": temperature
  "battery" : bateria
}
```

- Id: Correspon al identificador del microcontrolador per poder indicar de quin dispositiu arriba la informació.
- Ssid: És el *ssid* de la xarxa en el qual està connectat el dispositiu.
- State: En el cas de que les mesures es portin a terme correctament, aquest serà “OK”, en el cas que hi hagi algun problema en les mesures per falta de contacte del sensor amb la pell del pacient, es trametrà un “NOK”.

- Hr: Valor de la freqüència cardíaca calculada a través de la mitjana de 500 mostres
- Spo2: Valor de la saturació d'oxigen calculada a través de la mitjana de 150 mostres
- Temperatura: Valor de la temperatura calculada a través de la mitjana de 25 mostres.
- Bateria: Correspon percentatge de bateria que li queda al dispositiu.

Un cop el servidor rep aquesta petició, respon al dispositiu amb el mateix missatge vist en la resposta de la petició anterior. La resposta rebuda tant en un cas com en l'altre, no deixa de ser la configuració que s'ha establert des de la plataforma web per aquell dispositiu. És per això que en els dos casos interessa rebre aquesta informació per poder actuar i fer les modificacions de configuració del dispositiu el més ràpid possibles. En l'hipotètic cas que només es tingüés en compte la configuració en la resposta d'una de les dues peticions, es podria passar fins a un total de 5 minuts sense que el dispositiu fes un canvi de configuració.

4.1.2.3.4. Gestió de l'estalvi de bateria

Tal com s'ha vist a la màquina d'estats, l'últim estat d'aquesta correspon al temps d'espera entre lectures. Aquest temps d'espera ve proporcionat per la resposta que retorna el servidor en fer el PUT de les dades. Aquest valor en minuts és el que el microcontrolador s'esperarà per tornar a executar la màquina d'estats des de l'estat 1.

En ser un dispositiu sense fils, el sistema incorpora una bateria. És important que aquesta mantingui el sistema actiu el màxim de temps possible sense necessitat de la seva càrrega. Com que el sistema d'alimentació del microcontrolador requereix que aquest estigui actiu per poder-se auto alimentar, no s'han pogut fer aquests temps d'espera apagant el dispositiu. És per això que s'ha optat per utilitzar un dels modes d'estalvi d'energia que ofereix el microcontrolador per consumir el mínim de bateria durant aquests temps d'espera, que poden ser de minuts. A continuació es mostren els diferents modes d'energia que ofereix el microcontrolador.

- Mode actiu: Aquest és el mode normal, on tots els components del microcontrolador funcionen de forma normal i totes les funcionalitats estan actives. El xip requereix més de 240 mA de corrent pel seu funcionament. En cas de fer servir les funcions de Wi-Fi i Bluetooth es poden arribar a pics de potència de fins a 790 mA[29].
- *Modem Sleep*: En aquest mode, només el Wi-Fi, el Bluetooth i la ràdio estan desactivats. La CPU es manté operativa i el rellotge es pot configurar. En aquest mode el xip consumeix al voltant de 3 mA a baixa velocitat i 20 mA a alta velocitat[29].

- *Light Sleep*: Durant aquest mode, els perifèrics digitals, la major part de la RAM i la CPU estan activats pel rellotge. Utilitza la tècnica de *Clock Gating* per tal d'activar i desactivar certes parts del sistema a través dels diferents *clocks* del rellotge. Durant aquest mode, la CPU s'atura apagant els seus polsos de rellotge, mentre que els pins RTC i el coprocessador ULP es mantenen actius. Això resulta en un consum aproximat de 0,8 mA[29].
- *Deep Sleep*: Aquest és el mode fet servir en aquest projecte. És el mode que estalvia més consum de bateria. En aquest mode, la CPU, la major part de la RAM i tots els perifèrics digitals s'apaguuen. Les úniques parts del xip que es mantenen enceses són: El controlador RTC, els perifèrics RTC i les memòries RTC. El xip consumeix entre 0,15 mA si el coprocessador ULP està encès i 10uA. En aquest cas s'ha usat un pin RTC per tal de mantenir el microcontrolador retro alimentat, tal com es veurà en el següent capítol. Un cop el dispositiu es desperta d'aquest mode, es fa un reinici a nivell de software, fet que no fa que el pin RTC passi a un estat baix[29].

Com bé diu la descripció del mode utilitzat, el pin RTC és el que manté l'alimentació del ESP32 activada. A continuació es mostra el fragment de codi que activa el pin RTC a un nivell alt i la funció que fa que el microcontrolador passi a mode *Deep Sleep*.

- Funció que força l'activació el pin RTC:

```
pinMode(4, OUTPUT) // Es determina que el Pin 4 sigui una sortida.  
digitalWrite(4, HIGH)// Funció per fer que el Pin 4 de sortida tingui un estat alt.  
  
rtc_gpio_hold_en(GPIO_NUM_4)// Aquesta funció manté el Pin 4 del microcontrolador a l'estat que s'hagi determinat prèviament.
```

- Funció de *Deep Sleep*:

```
esp_sleep_enable_timer_wakeup(time_to_sleep)// determina el temps que el microcontrolador estarà en aquest mode.  
  
Esp_deep_sleep_start()// Inici del mode Deep Sleep.
```

4.1.3. Placa de circuit imprès

4.1.3.1. Programari utilitzat

El programa utilitzat pel disseny i desenvolupament de la placa de circuit imprès ha estat el KiCAD. Aquest és una aplicació multiplataforma de software lliure pel disseny electrònic automatitzat escrit en C++ amb wxWidgets per funcionar en FreeBSD, Linux, Microsoft i Mac OS X. L'aplicació usa un entorn integrat per totes les etapes del procés de disseny de plaques de circuit imprès: la captura dels esquemes, el disseny de la PCB, la generació/visualització d'arxius Gerber i l'edició de llibreries[30].

4.1.3.2. Components escollits

Tal com s'ha vist a l'inici d'aquest capítol, s'han explicat els components principals que formen la Bibo band, aquests son el microcontrolador ESP32, la placa de desenvolupament MAXREFDES117 (amb el sensor de freqüència cardíaca i saturació d'oxigen) i el sensor MAX30205 encarregat de la mesura de la temperatura corporal. A continuació es detallaran els components escollits pel disseny i fabricació de la placa de circuit imprès i pel seu correcte funcionament. De forma general, les funcionalitats d'aquests components són les de poder dotar de bateria el dispositiu i poder programar el microcontrolador a través del port micro-USB. Val a dir que els components escollits han estat limitats, quant a model, a l'actual falta de distribució de components electrònics a escala mundial.

- Regulador de tensió

NCP114: En aquest cas, es tracta d'un regulador de tensió *Low Dropout* (LDO) que té com a característica que la seva tensió de sortida té una diferència petita en relació amb la tensió d'entrada, aquesta diferència és al voltant dels 200 mV. D'aquesta manera es permet operar amb tensions d'alimentació petites i s'assegura un baix consum de potència amb una eficiència mitjana. El regulador escollit treballa amb tensions d'entrada d'entre 1,7 V a 5,5 V[31][32].

- Controlador de gestió de càrrega

MCP73831: El controlador de càrrega escollit és lineal. Aquest s'encarrega de limitar la velocitat a la qual el corrent elèctric se suma o s'estreu de les bateries elèctriques. En concret el MCP73831 és molt utilitzat en aplicacions d'espai molt limitat, sensibles als costos, i usos portàtils. En aquest cas s'usa un algoritme de corrent/tensió constant amb un precondicionament i terminació de càrrega que es poden seleccionar. La regulació de la tensió constant és fixe amb quatre opcions disponibles: 4,20 V, 4,35 V, 4,40 V, 4,50 V. El valor del corrent constant s'ajusta amb una resistència externa[33].

- Convertidor USB a UART

CP2102N: El convertidor d'USB a UART és un circuit integrat que s'utilitza per enviar o rebre dades en sèrie des d'un port USB a dades en sèrie que poden ser rebudes o enviades per una interfície UART. El convertidor escollit proporciona una solució senzilla per actualitzar els dissenys de RS-232 a USB usant un mínim de components i espai a la PCB. El CP2102N inclou un controlador de funció de velocitat completa USB 2.0, un transceptor USB, un oscil·lador i un receptor/transmissor asíncron universal (UART)[34].

- Bateria LIPO

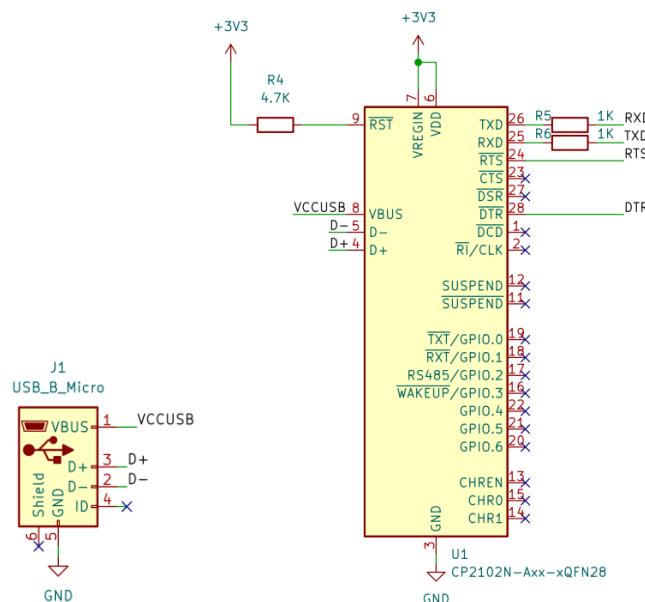
HPL402323-2C190mAh: La bateria escollida és del tipus LIPO (liti i polímer). Aquestes són un tipus de bateria en què l'electròlit és una solució de sals de liti, l'ànode és de carboni i el càtode d'òxid metàl·lic. En concret la bateria escollida és la HPL402323-2C. Aquesta té un voltatge nominal de 3,7 V, suficient per alimentar el microcontrolador a 3,3 V. La capacitat de la bateria és de 0,2C190mAh, fet que indica que la bateria es pot descarregar durant 20 minuts a aquest corrent. En el cas de la PCB és complicat saber amb exactitud el temps de duració de la bateria, ja que cada dispositiu pot configurar-se de forma diferent i realitzar temps d'espera més llargs o més curts on el consum de bateria és mínim[35].

4.1.3.3. Esquemàtic

A continuació es mostraran els esquemàtics usats per tal de representar les diferents parts de la PCB.

- Convertidor USB a UART

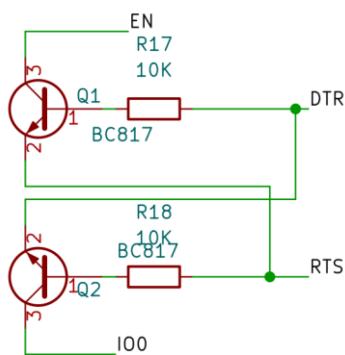
A continuació es mostra l'esquemàtic de la part encarregada de convertir el senyal provinent del microUSB a UART, per així poder programar el microcontrolador. S'han buscat els símbols tant del microUSB com del convertidor CP2102N-QFN28. Un cop trobats els símbols, aquests s'han unit a través de les sortides del microUSB amb les entrades del convertidor. Com es pot observar, els senyals D+, D- i VCCUSB són els senyals que passen del microUSB al convertidor. Un cop al convertidor, aquest s'encarregarà de processar el senyal i treure'l per les sortides TXD, RXD, RTS i DTR, on les dues primeres aniran als pins RXD0 i TXD0 respectivament del microcontrolador i serà per aquí per on es programarà a través de la transmissió UART. Els senyals RTS i DTR s'utilitzen per manipular el senyal Enable i el IO0 del microcontrolador, tal com es veurà a la següent part de l'esquemàtic.



Il·lustració 3: Esquemàtic convertidor USB a UART

- Gestió de la programació

Aquest petit circuit s'encarrega de gestionar la programació del microcontrolador a través de la manipulació dels senyals EN i IO0 amb els senyals DTR i RTS. Aquests dos últims canvien el seu valor en el moment que el convertidor USB a UART vol enviar dades o les vol rebre. És en aquest cas quan es manipularan les entrades EN i IO0 del dispositiu per poder realitzar la programació d'aquest. En el moment que el convertidor USB a UART detecta que té informació per tramestre al microcontrolador, activa el senyal DTR fent que el senyal EN prengui el valor de RTS. És en aquest moment quan el microcontrolador es força a un reinici momentani i es pot programar sense problemes.



Il·lustració 4: Gestor de la programació del microcontrolador

- Controlador de càrrega

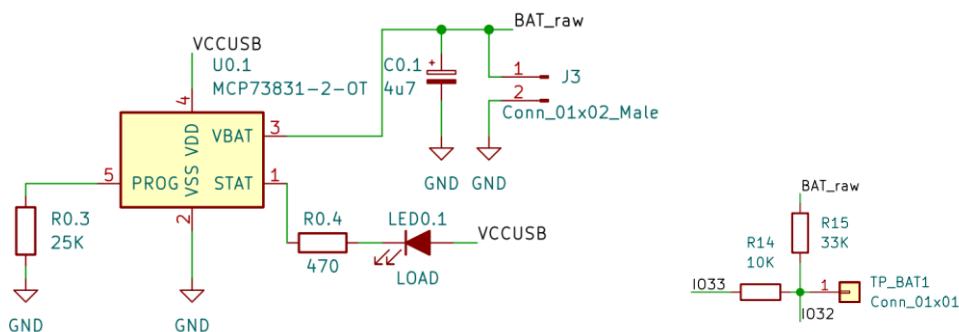
A continuació es pot veure l'esquemàtic que fa referència al control de la càrrega de la bateria a través de l'MCP73831. Tal com es pot observar a l'esquemàtic, aquest té una entrada anomenada PROG. Aquesta entrada s'utilitza per limitar el corrent de càrrega a través del valor de la resistència connectada a aquest pin.

L'entrada STAT, és l'encarregada d'indicar l'estat de càrrega de la bateria gràcies al LED que presenta a la seva sortida. En el cas que el LED estigui apagat, indica que la bateria està carregada, si el LED està encès, indica que la bateria s'està carregant.

La sortida VBAT s'encarrega de carregar la bateria a través del connector. Per altra banda, a través de BAT_raw, la bateria alimenta el microcontrolador, en cas que no estigui connectat per USB. Tal com es veurà a la següent part de l'esquemàtic.

Finalment, es pot observar un divisor de tensió a la dreta del circuit. Aquest s'utilitza per calcular el nivell de la bateria restant. És per això que s'ha configurat el pin 32 del microcontrolador com a entrada ADC i el pin 33 com a sortida amb un estat baix. D'aquesta manera llegint el valor d'ADC del pin 32 i fent una simple regla de tres, es pot saber el nivell de la bateria.

Quan la bateria està al màxim, la lectura ADC dona un valor de 1036. En el moment que la bateria està al mínim, l'ADC dona un valor de 700. Simplement, s'han restat els dos valors obtenint un valor de 336 i fent l'equivalència de 336 -> 100% i 0 -> 0%. A través d'aquesta regla de tres és molt simple saber el percentatge de bateria i informar-ho així a l'aplicació web.



Il·lustració 5: Esquemàtic del controlador de càrrega

- Regulador de tensió

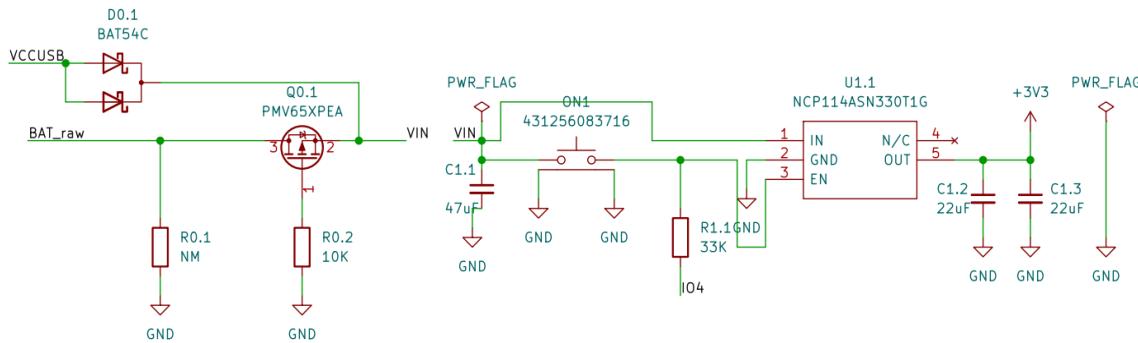
A la següent part de l'esquemàtic, es pot veure el regulador de tensió NCP114 usat per alimentar d'una forma adequada a 3,3 V el microcontrolador. Com es pot observar, hi ha dues vies d'alimentació.

En cas que la Bibo band estigui alimentada pel microUSB, la bateria estarà carregant i el transistor PMOS no deixarà passar corrent del senyal BAT_raw, de manera que l'alimentació es farà per la via del VCCUSB. Per altra banda, en cas que VCCUSB no estigui connectat, el transistor PMOS deixarà passar el corrent provenint de BAT_raw i el microcontrolador s'alimentarà a través de la bateria.

Tot seguit el senyal VIN serà transformat per generar una sortida de 3,3 V. Aquesta transformació o regulació de tensió es farà a través del NCP114, independentment de la font de tensió (VCCUSB o BAT_raw). Tot i això, es pot veure que el regulador NCP114 té una entrada anomenada EN.

Aquesta entrada és la que permetrà que a la sortida OUT hi hagi el senyal de 3,3 V o no per alimentar el microcontrolador. És per això que hi ha un polsador que habilita que aquesta entrada estigui en estat alt. Aquest serà el polsador per engegar el dispositiu, ja que farà que l'entrada EN del regulador s'activi i així activi la sortida per alimentar el microcontrolador.

Un cop que es deixa el polsador, es pot pensar que el senyal EN passarà a un estat baix i, per tant, el regulador deixarà d'alimentar el microcontrolador. Realment és així si no fos pel senyal del pin IO4 provinent del microcontrolador que manté l'entrada EN en estat alt. Com a resultat, es pot dir que un cop el microcontrolador s'engega, és ell mateix que es manté actiu a través del pin IO4 i de mantenir activat el regulador. En el moment que es força un reinici manual del microcontrolador, el pin IO4 deixa d'estar en estat alt, l'entrada EN es desactiva i, per tant, el regulador deixa d'alimentar el microcontrolador provocant l'aturada d'aquest.



Il·lustració 6: Esquemàtic del regulador de tensió

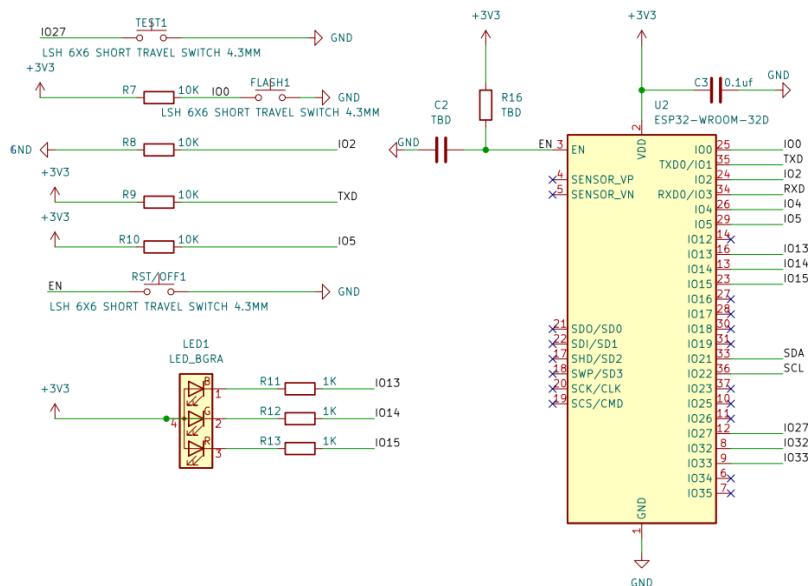
- Microcontrolador

A continuació es mostra l'esquemàtic corresponent al microcontrolador ESP32-WROOM-32 i a les connexions que tenen les seves diferents entrades. Com es pot observar al microcontrolador, cap de les entrades i sortides presenta circuits extra exceptuant l'entrada del senyal EN. Tal com es mostra al microcontrolador, aquest incorpora un circuit RC. La funcionalitat d'aquest circuit és bàsicament generar un temps d'espera per assegurar que arriba corrent a tot el microcontrolador, per posteriorment activar el senyal EN i engegar així el dispositiu.

Per altra banda, a l'esquerra del microcontrolador, es poden veure varíes connexions entre els pins de sortida i elements com resistències, botons o LEDs. És interessant saber que la resistència pull-down del pin IO2 i la resistència pull-up del pin IO5 han de ser-hi perquè el procés de programar el microcontrolador es dugui a terme sense problemes.

Per altra banda, es pot veure una connexió molt important, com és la del botó entre l'EN i GND. Aquest s'utilitzarà per parar el dispositiu. Al pressionar-lo, es forçarà que l'EN del microcontrolador passi a un estat baix, fent que el pin IO4 deixi d'alimentar el regulador de tensió, vist a l'apartat anterior i, per tant, que el dispositiu s'aturi.

Pel que fa al LED RGB, simplement donarà informació visual quan s'engegui la Bibo band per indicar que està activa així com, cada vegada que faci les mesures per continuar informant de la seva activitat.

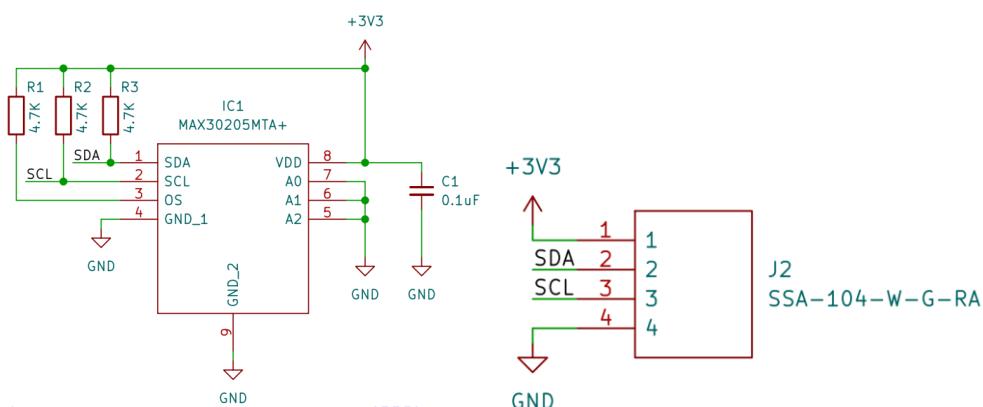


Il·lustració 7: Esquemàtic ESP32-WROOM-32

- Sensor de temperatura i connector

Finalment, queda veure l'esquemàtic del sensor de temperatura MAX30205 i del connector de 4 pins que permetrà la comunicació entre el microcontrolador i la placa de desenvolupament MAXREFDES117, que conté el sensor de freqüència cardíaca i saturació d'oxigen, i que estarà situat a la base del dit índex.

Els dos sensors es comuniquen a través del protocol I2C, per tant, com es pot comprovar, els dos tenen connexions amb els pins SDA i SCL del microcontrolador. Les connexions dels sensors a aquests pins han de tenir una pull-up perquè el microcontrolador pugui fer de mestre sobre els dispositius. Com es pot veure a la imatge, al sensor de temperatura se li han d'aplicar aquestes pull-up, mentre que la placa de desenvolupament MAXREFDES117 ja les porta de sèrie.



Il·lustració 8: Esquemàtic sensor de temperatura i connector de 4 pins

4.1.3.4. Disseny de la PCB

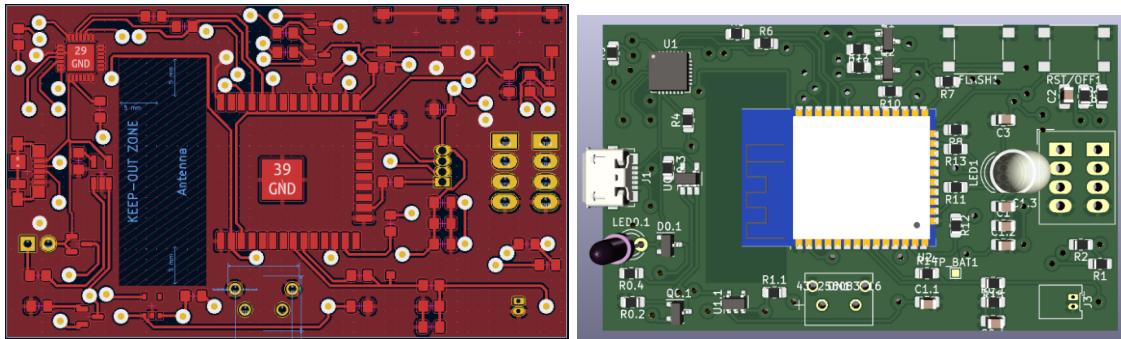
A continuació es mostraran el disseny final de la PCB, tant pel que fa a footprints com de model en tres dimensions. S'ha intentat que el disseny sigui el més petit possible. El resultat ha estat positiu però amb marge de millora per tal de reduir la mida de la placa. S'ha pensat en un disseny rectangular, perquè segueixi la forma del canell on ha d'anar aquest dispositiu tal com es veu a la Il·lustració 9.

A la cara superior de la PCB és on hi ha quasi la totalitat dels components que conformen la placa. Descrivint la placa d'esquerra a dreta, s'hi troba primer de tot el connector microUSB. Aquest connector està connectat directament amb el convertidor USB/UART situa a la part superior i esquerra de la placa. Per altra banda, el microUSB també es connecta directament amb el LED de càrrega del sistema, situat a la part inferior i esquerra.

Seguint per la part inferior, s'hi troba tot el sistema del control de càrrega, juntament amb el botó en forma de 90° encarregat d'enengar el sistema. Aquest botó està situat al lateral de la Bibo band per poder ser accessible fàcilment. Tot i això, si el dispositiu ja està en marxa, per molt que es pressioni el botó, no variarà res. A la part superior i dreta de la placa, s'hi troba el botó de reinici del sistema, que de la manera que s'ha dissenyat, aquest botó el que fa és aturar el dispositiu, ja que en fer el reinici, posa en estat baix el pin que manté actiu el microcontrolador, i, per tant, el para.

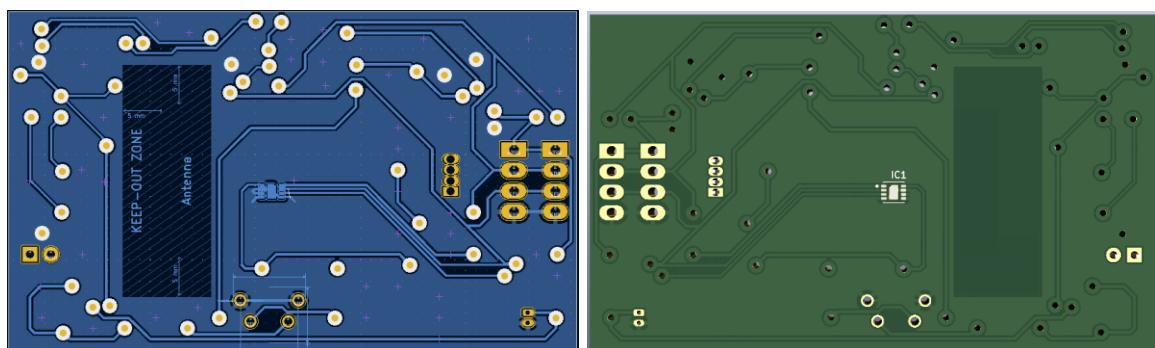
Seguidament i encara que sembli estrany, s'ha escollit la part central de la placa, per situar-hi el microcontrolador. Un dels motius pel qual el microcontrolador està en aquesta posició, és per totes les connexions que presenta al seu voltant creient que aquesta era la millor situació sense tenir en compte l'espai reservat per l'antena. Evidentment, es podria haver situat d'una altra manera, on part d'aquest espai de prudència per l'antena estigués al marge de la placa.

Finalment, es troben les preparacions amb els forats a la placa pel LED RGB i pel connector de 4 pins on s'hi connectarà al sensor de freqüència cardíaca i saturació d'oxigen. Cal recordar que en una primera iteració, aquests sensors havien d'estar integrats a la placa igual que el sensor de temperatura, però després de realitzar certes proves, es va veure que les mesures de la freqüència cardíaca i saturació d'oxigen eren molt més reals si es prenien a la falange proximal del dit índex de la mà. És per això que s'ha decidit fer aquesta connexió a través d'una connexió de 4 cables per obtenir mesures realistes, tot i incrementar la incomoditat del dispositiu.



Il·lustració 9: Part frontal de la placa de desenvolupament i model 3D

Pel que fa a la part inferior de la placa, s'ha decidit col·locar-hi únicament un component de tota la PCB. Aquest és el sensor de temperatura corporal. La situació del sensor és aquesta, degut a la necessitat de què el sensor de temperatura estigui en constant contacte amb la pell del pacient. Per tant, a l'hora de dissenyar l'encapsulat per aquesta placa, s'ha de tenir en compte deixar aquest forat perquè el sensor pugui estar en contacte amb la pell del pacient.



Il·lustració 10: Part inferior de la placa de desenvolupament i model 3D

Finalment, gràcies a la maquinària del TechLab de la universitat s'ha fresat la placa i s'ha obtingut el resultat de la Il·lustració 11. Les dimensions finals de la placa han estat de 69 mm de llargada i 41 mm d'amplada. És un dispositiu força gran per la situació anatòmica on ha d'anar situat. Aquest fet indica que es necessiten futures iteracions en el disseny i l'elaboració de prototips més petits i portables.



Il·lustració 11: Resultat final del disseny de la PCB

4.1.4. Encapsulat

4.1.4.1. Programari utilitzat

Un cop dissenyada la PCB, ha estat moment de dissenyar l'encapsulat on aquesta anirà inserida. Aquest encapsulat s'ha imprès amb una impressora 3D, i per tant, el seu disseny s'ha creat a través d'un programa enfocat a aquest objectiu. Aquest ha estat el FreeCad. Aquest software és el que s'ha utilitzat durant el grau per tal de dissenyar qualsevol prototip en 3D.

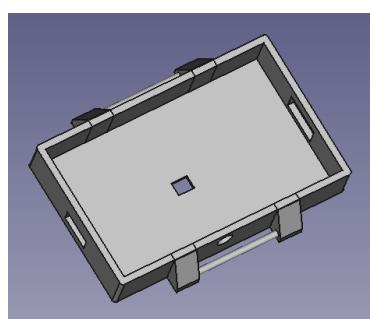
FreeCad és un modelador 3D paramètric i de codi obert enfocat principalment a dissenyar objectes de qualsevol mida. El programa permet dibuixar formes en 2D i utilitzar-les com a base per construir altres objectes. Els dibuixos finals estan llistos per la seva producció d'alta qualitat[36].

4.1.4.2. Disseny de l'encapsulat

Per realitzar el disseny de l'encapsulat, s'han intentat tenir en compte o posar-se a la pell d'un pacient i del personal sanitari, és a dir, s'ha intentat que sigui el més còmode possible a la vegada d'incrementar la seguretat contra errors humans. També s'ha de pensar que és la primera iteració i un primer prototip del dispositiu. Les dimensions són elevades i aconseguir la comoditat és complicat. S'ha dividit el disseny en dues parts. La que fa referència a la tapa del dispositiu, i la que fa referència a la mateixa caixa on hi anirà emmagatzemada la PCB vista a la Il·lustració 11.

- **Disseny de la caixa**

Com es pot veure a la Il·lustració 12, s'ha seguit el disseny rectangular de la PCB. Com s'ha comentat anteriorment, s'ha tingut en compte el sensor de temperatura de cara al disseny de la cara inferior de la caixa amb el forat perquè aquest estigui en contacte amb la pell. Per altra banda, s'han deixat els forats laterals corresponents al connector microUSB, el cablejat del sensor de freqüència cardíaca i saturació d'oxigen que si situarà al dit índex i l'orifici pel botó d'engegat del sistema, el qual també s'ha dissenyat a part per millorar la comoditat de pulsació d'aquest. Finalment, s'han dissenyat les barres laterals que seran les subjeccions de la corretja per fixar el dispositiu al canell.



Il·lustració 12: Disseny de la caixa de la Bibo band

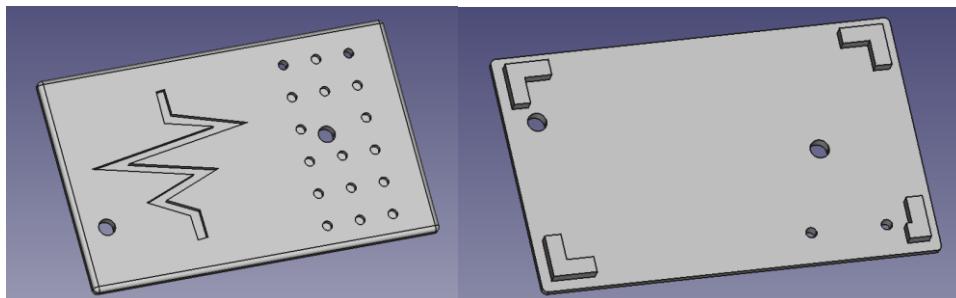
- Disseny de la tapa

Pel disseny de la tapa s'ha intentat buscar més l'estètica i no centrar-se tant en la funcionalitat tot i que en té alguna. Primer de tot s'han implementat els forats pels LEDs del sistema perquè es vegin a simple vista. Aquests forats són els dos més grans de la tapa.

Tot seguit s'han implementat els forats pels botons interns del sistema. Al botó per apagar el dispositiu s'hi accedeix per l'orifici superior dret de la primera imatge de la Il·lustració 13 i al botó de flash del dispositiu per l'orifici real que hi ha a la seva esquerra. A l'interior d'aquests orificis s'hi pot accedir a través d'un cilindre molt prim, de manera que els pacients no puguin o tinguin la temptació de pressionar cap d'aquests botons. Per altra banda, i per confondre, s'han creat moltes simulacions de forats perquè a simple vista no es vegi que es pot accedir a l'interior dels dos comentats anteriorment.

Tot seguit s'ha dissenyat el logotip de VITALS i s'ha situat a la cara superior de la tapa. Aquest criteri de disseny és purament estètic i no aporta cap funcionalitat.

Finalment, s'han implementat els suports que es poden observar a la cara inferior de la tapa. Aquests suports serveixen per poder tancar la tapa a pressió sense necessitat de material extra com cargols i tornavisos.

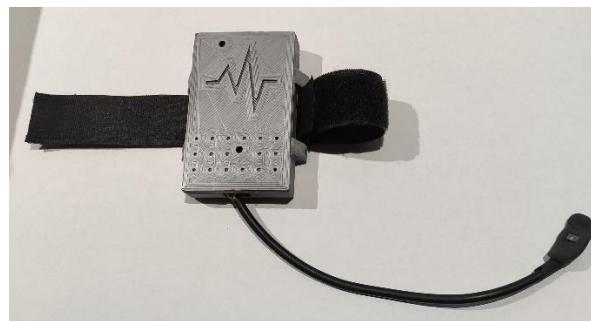


Il·lustració 13: Cara superior i inferior de la tapa de la Bibo band

4.1.4.3. Disseny final real

Un cop pensades i dissenyades totes les parts de la Bibo band, ha estat el moment de plasmar la teoria o els dibuixos creats en un ordinador, a la realitat. El resultat obtingut ha estat un dispositiu gran però estèticament acceptable. Algun dels requeriments del dispositiu s'han vist limitats, com per exemple la mida i la comoditat, a causa de les restriccions que s'han hagut de seguir per intentar dissenyar o dur a terme aquest model a la mateixa universitat i per la poca experiència personal en aquest àmbit.

Per tal de subjectar la caixa al canell s'ha utilitzat una cinta de Belcor, igual que pel sensor de freqüència cardíaca i saturació d'oxigen situat al dit. Per millorar la comoditat d'aquest últim, s'ha usat una pasta anomenada (*Funny gummy*), la qual permet la seva manipulació de forma lliure, fins que al cap d'un dia s'endureix i manté l'última posició adquirida. El disseny final es pot veure a la Il·lustració 14.



Il·lustració 14: Resultat final del disseny de la Bibo band

4.2. Dispositiu d'alarmes

Tal com s'ha vist a l'esquema del sistema, el dispositiu d'alarmes és aquell que detecta quan es genera una alarma a través de consultes periòdiques al servidor. Una alarma es genera quan la Bibo band d'un pacient realitza una mesura de les constants vitals i aquesta no està dins el límít que s'ha configurat des de la plataforma web per aquell pacient. En aquest moment es genera una alarma la qual queda enregistrada a la base de dades. Aquesta es mostra a la plataforma web amb l'opció de poder-se aturar en qualsevol moment a través d'un botó a la mateixa plataforma.

Per altra banda, cada dispositiu d'alarmes estarà situat a una unitat de l'hospital diferent i s'assignarà el dispositiu a aquesta unitat a través de la plataforma web, d'aquesta manera, quan es faci la consulta per veure si hi ha alarmes actives, cada dispositiu consultarà únicament les alarmes que hi hagi a la unitat a la qual estan enllaçades, de manera que un pacient de la unitat A no farà saltar l'alarma del dispositiu d'alarmes que està enllaçat a la unitat B.

4.2.1. Hardware

El hardware que compon el dispositiu d'alarmes és molt simple, ja que únicament es requereix dos dispositius i, en aquest cas, no s'ha dissenyat una PCB per la facilitat d'ús d'aquests dispositius i la poca limitació d'espai físic com sí que requeriria la Bibo band.

Pel sistema d'alarmes es continua requerint un microcontrolador, aquest s'encarregarà de realitzar les peticions al servidor per tal de comprovar si hi ha alarmes actives o no a través de consultes a la base de dades. Per tant, el microcontrolador escollit ha de tenir mòdul Wi-Fi igual que les Bibo band. És per això que el dispositiu escollit torna a ser un ESP-32. En aquest cas no és el mòdul per PCB utilitzat a la Bibo band, sinó la placa de desenvolupament comercial.

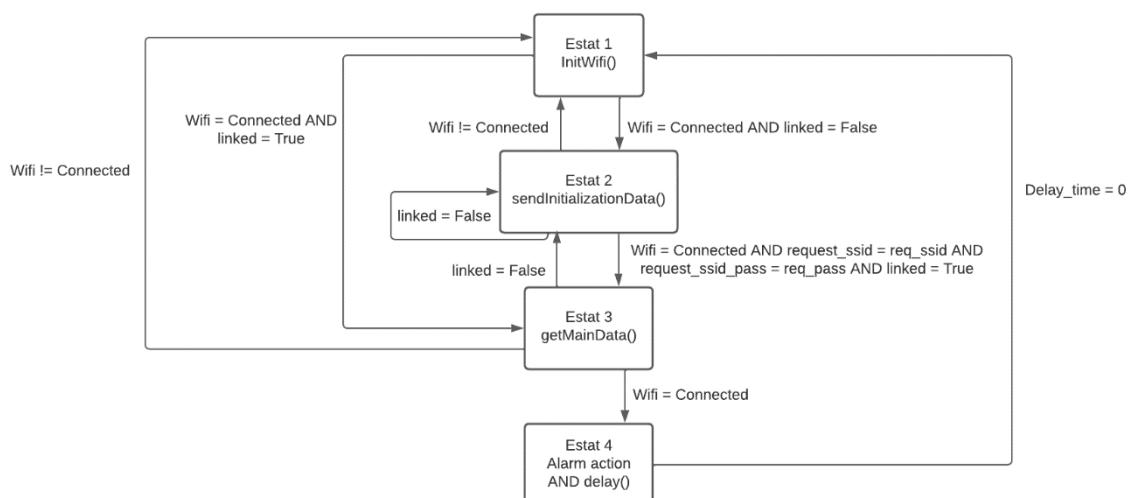
Per altra banda, el segon dispositiu encarregat del funcionament del sistema d'alarmes és un relé. S'ha escollit l'HW482. Aquest és un mòdul de relé de 5 V. Té un total de 3 pins (VCC, GND i Senyal). Aquest actua com un interruptor si el circuit i el circuit de càrrega tenen diferents tensions d'alimentació. És un interruptor que s'usa per connectar una connexió aïllada a través d'un senyal d'ell mateix. Presenta un LED vermell que s'engega cada vegada que la bobina agafa energia o el pin del senyal té un estat alt.

4.2.2. Software

L'eina de programació utilitzada ha sigut l'Arduino IDE igual que la Bibo band i les llibreries utilitzades han estat les mateixes, excepte les que fan referència als sensors de les constants vitals. En certa manera s'ha reaprofitat gran part del codi en comparació al que s'ha usat per a les Bibo band, ja que la funcionalitat que realitzen els dos dispositius és molt similar.

4.2.2.1. Estructura general del codi

El codi o funcionalitat del microcontrolador es fa a través de la màquina d'estats de la Il·lustració 15. Aquesta es gestiona a través de la funció `machineState()`, la qual es crida cada vegada que es fa el bucle principal. D'aquesta manera a cada cicle del bucle, es mira si s'ha de canviar l'estat per posteriorment fer l'acció de l'estat en qüestió. El codi és el més genèric possible per utilitzar-lo en tots els dispositius d'alarmes sense modificacions.



Il·lustració 15: Màquina d'estats dispositiu d'alarmes

A l'estat 1 es realitza la gestió de la connexió, tal com s'ha vist al capítol de la Bibo band. Per tant, és important recordar que la configuració de la connexió dels dispositius d'alarmes també es pot gestionar des de la plataforma web i es tracta de la mateixa manera que les Bibo band.

És per això que a l'hora d'estructurar el codi s'ha intentat que sigui el més genèric possible per poder-lo utilitzar en els 2 dispositius que tenen finalitats molt diferents. Per tant, en aquest cas, també es compta amb un guardat de les xarxes Wi-Fi a la memòria Flash del microcontrolador. Un cop el dispositiu es connecta a una xarxa Wi-Fi es passa a l'estat 2.

A l'estat 2 es fa una identificació del dispositiu d'alarmes al servidor a través del mètode PUT. En cas que el dispositiu estigui ja identificat i assignat a una unitat mèdica, es passa a l'estat 3, per altra banda, en cas que no estigui enllaçat a cap unitat mèdica, es queda al mateix estat repetint la petició de forma periòdica cada 10 segons. En cas que la resposta del servidor contingui una nova xarxa a la qual s'ha de connectar el dispositiu, es passa altra vegada a l'estat 1 per intentar la connexió a aquesta nova xarxa.

A l'estat 3 es fa un GET al servidor demanant si hi ha alarmes actives a la unitat a la qual el dispositiu està assignat. La resposta del servidor no només conté la informació de si hi ha alarmes actives, sinó que informa de si el dispositiu continua enllaçat, la configuració de la xarxa Wi-Fi que s'ha fet des de la plataforma web i el temps d'espera que ha d'estar en repòs mentre no es torna a fer una petició per determinar si hi ha alarmes actives o no. En el cas que el dispositiu continui enllaçat i la configuració de la xarxa Wi-Fi no hagi variat, es passarà a l'estat 4. En cas que el dispositiu no estigui enllaçat es passarà a l'estat 2 i en cas que la configuració de la xarxa hagi variat es passarà a l'estat 1.

A l'estat 4 és on es porta a terme l'acció d'activar o desactivar el relé que hi ha connectat en un dels pins del microcontrolador. Aquesta activació o desactivació és causada per la resposta del servidor que s'ha obtingut a l'estat 3. Si la resposta indica que l'alarma ha d'estar activada, el microcontrolador activarà el relé fent que la bombeta generi un senyal lluminós. Per altra banda, en cas que la resposta del servidor sigui negativa pel que fa a l'activació de l'alarma, el microcontrolador desactivarà el relé fent que la bombeta s'apagui en cas que estigui activada. Tot seguit el microcontrolador romandrà esperant tant de temps com estigui configurat a la plataforma web i, un cop passat aquest temps, es tornarà a l'estat 1 i es tornarà a realitzar tot el cicle.

4.2.2.2. Protocol de comunicació

Per tal de tenir un millor control i ordre sobre les dades enviades per part dels dispositius i les respostes per part del servidor, s'ha creat un protocol de comunicació entre el servidor i els diferents dispositius d'alarmes. Tal com s'ha vist a la màquina d'estats, és en els estats 2 i 3 quan es produeix un intercanvi de dades amb el servidor. A l'estat 2 es du a terme el protocol d'identificació dels dispositius, mentre que a l'estat 3 es demanen les dades de les alarmes per la unitat on els dispositius estan assignats.

- Protocol de comunicació d'identificació

En aquest estat, el dispositiu simplement indica al servidor que està actiu, això ho fa a través de realitzar un PUT al servidor, a la ruta corresponent del servidor. El missatge enviat és en format JSON i conté la següent informació:

- PUT: <https://vitalstechnology.com/> identification_alarm_device
- Body:*

```
{
  "id": id,
  "ssid": ssid
}
```

- Id: Correspon a l'id del microcontrolador, d'aquesta manera és com s'identifiquen els diferents dispositius dins la base de dades. Aquest s'obté gràcies a la funció *ESP.getEfuseMac()*.
- Ssid: És el *ssid* de la xarxa on està connectat el microcontrolador. Gràcies a això, es pot veure a la plataforma web a quina xarxa estan connectats els dispositius en temps real.

Un cop el servidor rep aquesta petició, respon al dispositiu amb el següent missatge en format JSON:

- Resposta:

```
{
  "linked": linked,
  "ssid": ssid,
  "ssid_pass": ssid_pass,
  "time_to_sleep": time_to_sleep
}
```

- Linked: És un booleà que informa si aquell dispositiu està enllaçat o no a una unitat dins la base de dades. En cas que no ho estigui, tal com es mostra a la màquina d'estats, es continuarà fent aquesta petició cada 10 segons fins que s'enllaci a una unitat.
- Ssid: Correspon al *ssid* que hi ha configurat a la plataforma web, és a dir, és el *ssid* on el dispositiu s'ha de connectar.
- Ssid_pass: És la contrasenya de la xarxa que s'ha configurat des de la plataforma web, és a dir, és la contrasenya del *ssid* on el dispositiu s'ha de connectar.

- Time_to_sleep: Correspon al temps que el dispositiu ha d'estar inactiu entre mesures. És a dir, el temps en què el dispositiu ha d'estar esperant fins a tornar a l'estat 1 i tornar a fer el cicle. Aquest temps també ve configurat des de la plataforma web.
- Protocol de comunicació de consulta de dades

En aquest estat, el dispositiu fa una consulta de l'estat de l'alarma per la unitat que ell té assignada a la base de dades. Aquesta consulta es realitza a través del mètode GET a la ruta del servidor corresponent.

- GET: https://vitalstechnology.com/alarm_detection/<id>
- Id: Correspon l'identificador del microcontrolador per poder indicar de quin dispositiu es vol consultar la informació.

Un cop el servidor rep aquesta petició de consulta, respon al dispositiu amb el següent missatge en format JSON.

- Resposta:

```
{
  "linked": linked,
  "ssid": ssid,
  "ssid_pass": ssid_pass,
  "time_to_sleep": time_to_sleep,
  "alarm_activation": alarm_activation
}
```

- Alarm_activation: Correspon a un booleà que informa de si l'alarma ha d'estar activa (True) o ha d'estar inactiva (False).

Com es pot veure, la resposta és igual a l'anterior afegint la informació de l'estat de l'alarma, això és degut a que si hi ha canvis de configuració, aquests es puguin fer de forma ràpida, després d'activar o desactivar l'alarma i no s'hagi d'esperar al temps d'espera del microcontrolador fins la propera petició.

4.2.3. Disseny final

Com s'ha mencionat, en aquest cas no s'ha dissenyat una PCB, ja que no hi havia limitacions d'espai. El disseny és simple i senzill. Aquest consta d'una caixa on, a la cara superior, presenta una bombeta encarregada de donar el senyal lluminós en cas d'alarmes. Dins la caixa, és on es troben els dos components descrits a l'apartat de hardware. Aquests estan connectats a través de cables.

Tal com es veu a la Il·lustració 16, la caixa té dos orificis. Per un d'ells surt el cable microUSB encarregat d'alimentar el microcontrolador, i per l'altre, surt el cable de l'alimentació de la bombeta. Aquest últim està connectat al relé perquè activi o desactivi la bombeta quan el sistema ho requereixi.



Il·lustració 16: Resultat final del disseny del dispositiu d'alarmes

5. Servidor

El servidor és el cervell del sistema, és l'encarregat de gestionar totes les peticions que rep i fer les gestions necessàries a la base de dades per a donar la resposta adient a aquestes peticions. És al servidor, per tant, on hi ha tota la lògica del sistema a través de l'aplicació web, i és on els dispositius externs i la plataforma web realitzen les comunicacions amb ell.

5.1. Aplicació web

En l'actualitat existeixen moltes opcions i tecnologies a l'hora de crear aplicacions web. En aquest cas s'ha escollit el *framework* de Flask, ja que és el que s'ha utilitzat més durant el grau. Aquest *framework* permet crear aplicacions web a través de Python[37].

Una de les característiques principals de Flask és que proporciona les eines necessàries per crear aplicacions web funcionals, però si es requereix noves funcionalitats, Flask les pot dur a terme a través d'extensions (*plugins*) que es poden instal·lar i dotar d'aquestes noves funcionalitats a Flask. Una altra característica de Flask és facilitar el desenvolupament d'applicacions web sota el patró MVC (Model Vista Controlador). Aquest patró és una manera o forma de treballar que permet separar el que és el model de dades (dades que contindrà l'aplicació), la vista (pàgina html) i el controlador (on es gestionen les peticions web)[37].

Els avantatges que proporciona l'ús de Flask són els següents:

- Proporciona una estructura de projecte, és a dir, totes les aplicacions Flask tindran els mateixos elements i els mateixos fitxers.
- Facilitat per trobar llibreries adaptades a aquest *framework*.
- Inclou un servidor web de desenvolupament per poder generar proves d'una manera ràpida i senzilla.
- És compatible amb Python3.
- Fàcil gestió de les rutes.
- És compatible amb WSGI, que correspon a un protocol de utilitzen els servidors web per servir pàgines escrites amb Python.

Tal com s'ha mencionat, el llenguatge utilitzat per la gestió de l'aplicació web ha estat Python. Python és un llenguatge de programació d'alt nivell que s'usa principalment per desenvolupar aplicacions de tota mena. A diferència d'altres llenguatges, es tracta d'un llenguatge interpretat, és a dir que no es requereix la seva compilació per executar aplicacions escrites amb ell. Aquestes aplicacions s'executen directament per l'ordinador usant un programa denominat interpretador i, per tant, no es requereix que es tradueixi a llenguatge màquina[38].

5.2. Allotjament del servidor

El servidor està allotjat a una màquina virtual *d'Amazon Web Services* (AWS), en concret una màquina *Amazon Elastic Compute Cloud* (Amazon EC2). Aquesta proporciona una capacitat de computació escalable al núvol d'AWS. El fet d'utilitzar una màquina EC2 elimina la necessitat d'invertir inicialment en hardware, de manera que es pot desenvolupar i implementar aplicacions en menys temps. Per altra banda, una màquina EC2 permet llençar tants servidors virtuals com es necessiti, configurar la seguretat i les xarxes i administrar l'emmagatzematge[39].

En concret la màquina creada, és el que Amazon anomena una t2.micro pel cost de 0 euros que té. Les característiques d'aquesta màquina són les següents:

- 1 CPU amb processador de 3.3 GHz Intel Xenon Scalable
- Memòria d'1 GiB
- Emmagatzematge en bloc d'alt rendiment
- Rendiment de la xarxa de baix a moderat
- IP pública: 13.38.34.202

El sistema operatiu que s'ha escollit és Linux Ubuntu 18.04 degut a la facilitat d'ús i l'experiència prèvia amb aquest sistema operatiu.

Per tal de poder llençar la instància i posar en funcionament aquesta màquina virtual, es requereixen uns passos de seguretat importants per complir un dels requisits de seguretat del sistema. Aquest pas requereix el sistema de xifratge de clau pública/privada per poder accedir a la màquina virtual sense problemes. Aquest xifratge s'utilitza per protegir arxius, carpetes i unitats completes contra l'accés no autoritzat[39].

El primer pas és descarregar el parell de claus (pública i privada) de tipologia RSA (Rivest, Shamir y Adleman). Aquest mètode utilitza la factorització de números enters per la generació del parell de claus. Aquesta opció de descàrrega la proporciona el propi servei d'AWS quan s'intenta iniciar la instància. Un cop descarregades el parell de claus, s'han de guardar dins la màquina des d'on es voldrà accedir a la màquina virtual.

Tot seguit s'han de moure el parell de claus privades a la carpeta *ssh* del sistema. Aquest pas es fa amb la següent comanda.

- `ssh mv vitals.pem ~/.ssh/vitals.pem`

El següent pas i per millorar la seguretat, s'ha d'assegurar que la clau no sigui visible des de l'exterior a través de la següent comanda.

- `Chmod 400 ~/.ssh/vitals.pem`

Finalment, ja es pot accedir a la màquina virtual utilitzant les claus generades i la IP pública de la màquina virtual a través de la següent comanda.

- `ssh -i ~/.ssh/vitals.pem ubuntu@13.38.34.202 -v`

5.3. Servidor web

Un cop es té accés a la màquina virtual, és l'hora d'instal·lar el que serà el servidor que s'encarregarà de servir l'aplicació web creada a través del *framework* Flask. S'ha escollit Apache perquè porti a terme aquesta tasca.

Apache HTTP Server és un software de servidor web gratuït i de codi obert per plataformes Unix. Aquest és mantingut i desenvolupat per la Fundació de Software Apache. El servidor no és un servidor físic, sinó un servidor software que s'executa en un servidor físic. La seva principal tasca és establir la connexió entre un servidor (en aquest cas el servidor Flask) i els navegadors dels visitants, mentre envia arxius entre ells[40].

Per exemple, quan un usuari vol accedir a una pàgina web, el seu navegador trameu una sol·licitud al servidor i Apache li torna una resposta amb tots els arxius sol·licitats. El client i el servidor es comuniquen a través del protocol HTTP i Apache simplement s'encarrega de garantir una comunicació fluida i segura entre les dues màquines[40].

Per tal d'instal·lar aquest servidor a la màquina virtual, s'ha d'utilitzar la següent comanda.

- `apt-get install apache2`

Un cop instal·lat, apareix el directori /var/www on Apache hi té un fitxer html de benvinguda perquè es pugui provar que tot funciona correctament. Per fer aquesta prova, primer s'ha d'activar Apache2 a través de la següent comanda.

- `service apache2 start`

Tot seguit s'ha d'accendir a un navegador web i introduir la IP pública de la màquina en qüestió juntament amb el port 80.

- `http://13.38.34.202:80`

Si tot ha estat instal·lat de forma correcta, el navegador ha de mostrar un missatge de benvinguda d'Apache2. El fitxer de configuració d'Apache2 es troba al següent directori.

- `/etc/apache2/sites-available/`

En aquest directori s'hi troba un *fitxer.conf* que és el que per defecte crea aquest servidor quan s'instal·la i el que permet que es pugui veure el missatge de benvinguda des del navegador web.

Per tal de configurar el servidor perquè comuniqi l'aplicació web Flask amb l'exterior, s'ha utilitzat el següent fitxer de configuració.

```
<VirtualHost *:80>

    ServerName vitalstechnology.com

    ServerAdmin ferran.f6f@gmail.com
    WSGIScriptAlias / /var/www/vitals/vitals.wsgi
    <Directory /var/www/vitals/vitals>
        Require all granted
    </Directory>

    ErrorLog ${APACHE_LOG_DIR}/error.log
    CustomLog ${APACHE_LOG_DIR}/access.log combined

</VirtualHost>
```

Com es pot veure al fitxer de configuració, el servidor està escoltant el port 80. Aquest correspon al servidor web amb protocol HTTP. Dins seu es pot veure que hi ha la referència a un fitxer de nom *vitals.wsgi* on se'n parlarà al pròxim capítol. Per tant, totes les peticions que entrin pel port 80 es redirigiran a aquest fitxer anomenat *vitals.wsgi*.

Un cop el fitxer de configuració s'ha guardat al directori */etc/apache2/sites-available/*, s'ha d'activar aquest fitxer a través de la següent comanda.

- *A2enmod fitxer.conf*

Un cop activat el fitxer, s'ha de comprovar que efectivament està al directori de fitxers activats.

- */etc/apache2/sites-enabled/*

Si el fitxer està en aquest directori simplement s'ha de reiniciar el servidor Apache2 perquè apliqui els canvis realitzats.

- *service apache2 restart*

5.4. WSGI

Web Server Gateway Interface o WSGI és una especificació que descriu com es comunica un servidor web (Apache) amb un servidor aplicació web escrita amb Python (Flask) i com es poden arribar a encadenar diferents aplicacions web per processar una sol·licitud/petició.

WSGI permet la comunicació sense problemes entre el servidor web (Apache), on l'únic que fa és esperar peticions de l'exterior, i l'aplicació web (Flask), on l'únic que fa és esperar que el servidor web li passi aquestes peticions per poder-les tractar i donar una resposta. L'intermediari que permet aquesta comunicació fluida és el WSGI. Ell s'encarrega d'adaptar les peticions del servidor web, independentment d'on vinguin, perquè siguin intel·ligibles per l'aplicació web, i a la vegada que siguin intel·ligibles pel servidor web quan l'aplicació web genera una resposta[41].

Per poder utilitzar aquesta interfície, primer de tot s'ha d'instal·lar WSGI a la màquina virtual a través de la següent comanda.

- *pip install mod_wsgi*

La configuració del WSGI es fa a través d'un fitxer que s'allotja al següent directori.

- */var/www/fitxer.wsgi*

En aquest fitxer s'ha d'indicar on es troba l'aplicació web escrita amb Python que ha de rebre les peticions que li envii el servidor web Apache2.

Per altra banda, com s'ha vist al fitxer de configuració de l'Apache2, s'ha d'indicar el directori d'aquest fitxer, ja que Apache2 és cap on redirigiran totes les peticions.

El fitxer WSGI configurat per aquest projecte és el següent.

```
import sys
import logging
logging.basicConfig(stream=sys.stderr)
sys.path.insert(0, '/var/www/vitals/vitals/')

from app import app as application
```

Com es pot veure, al fitxer hi ha la ruta on es troba l'aplicació web per poder indicar a la petició que li arriba del servidor web (Apache) a on ha de redirigir aquesta petició.

Finalment, només queda situar tots els fitxers de l'aplicació web necessaris, perquè es pugui executar sense problemes, al directori on s'ha indicat en el fitxer de configuració WSGI.

5.5. Seguretat TLS

Amb la configuració feta fins al moment, l'aplicació és accessible a través del protocol HTTP sense seguretat. És a dir, fins ara només es pot accedir a l'aplicació web a través del port 80 del servidor web. Un dels requisits d'aquest sistema és que totes les comunicacions siguin segures, per tant, és necessari que les comunicacions es facin a través del protocol HTTPS incorporant el certificat TLS (seguretat de la capa de transport)[42].

El certificat TLS és un certificat digital. Bàsicament, és un arxiu que conté dades per vincular claus criptogràfiques amb les dades d'una pàgina web. L'objectiu principal és proporcionar seguretat en les comunicacions. La seva funció és xifrar el contingut que s'envia i es rep quan es navega per una pàgina web. És important aplicar aquest certificat digital al nom de domini que s'ha escollit (vitalstechnology.com) perquè totes les connexions a aquest servidor siguin sota el protocol HTTPS, tant les que provenen dels microcontroladors, com són la Bibo band i els dispositius d'alarmes, com totes les peticions que es fan a través de la plataforma web[42].

Per tal d'inserir aquest certificat al servidor, s'ha utilitzat l'autoritat certificadora *Let's Encrypt*. L'objectiu d'aquesta autoritat certificadora és fer possible la configuració d'un servidor HTTPS i fer que obtingui automàticament un certificat confiat pel navegador, sense cap intervenció humana[43]. Aquesta incorporació del certificat es fa a través dels següents passos.

Primer de tot i en cas de no tenir, s'ha d'instal·lar Python3 a la màquina virtual.

- *apt install python3*

Tot seguit s'ha d'instal·lar el client de *Let's Encrypt Certbot*, el qual s'encarregarà de renovar automàticament el certificat i indicar a quin nom de domini se li ha d'aplicar.

- *certbot -apache -d vitalstechnology.com*

Tot seguit s'ha d'indicar si es vol rebre informació al correu o no i després d'aquests passos ja s'obté el certificat digital.

Per comprovar que *Certbot* està actiu es fa a través de la següent comanda.

- *systemctl status certbot.timer*

Tot seguit Apache haurà guardat el certificat digital. Automàticament haurà creat un nou fitxer de configuració on hi haurà una redirecció del port 80 al port 443 per totes les peticions que vagin a vitalstechnology.com. S'ha de modificar el fitxer de configuració perquè quedi de la següent manera.

```
<VirtualHost *:80>
    ServerName vitalstechnology.com
    ServerAdmin ferran.f6f@gmail.com
    WSGIScriptAlias / /var/www/vitals/vitals.wsgi
    <Directory /var/www/vitals/vitals>
        Require all granted
    </Directory>
    Redirect / https://vitalstechnology.com
```

```

ErrorLog ${APACHE_LOG_DIR}/error.log
CustomLog ${APACHE_LOG_DIR}/access.log combined

</VirtualHost>

# vim: syntax=apache ts=4 sw=4 sts=4 sr noet

<IfModule mod_ssl.c>
<VirtualHost *:443>

    ServerName vitalstechnology.com
    ServerAdmin f6f@gmail.com

    WSGIScriptAlias / /var/www/vitals/vitals.wsgi
    <Directory /var/www/vitals/vitals>
        Require all granted
    </Directory>

    ErrorLog ${APACHE_LOG_DIR}/error.log
    CustomLog ${APACHE_LOG_DIR}/access.log combined

ServerName vitalstechnology.com
SSLCertificateFile /etc/letsencrypt/live/vitalstechnology.com/fullchain.pem
SSLCertificateKeyFile /etc/letsencrypt/live/vitalstechnology.com/privkey.pem
Include /etc/letsencrypt/options-ssl-apache.conf
</VirtualHost>
</IfModule>
```

Com es pot veure, a l'afegir el certificat digital, automàticament Apache ja s'encarrega de modificar el seu fitxer de configuració. Simplement, el que fa és escoltar també pel port 443 (que és el port de les peticions amb protocol HTTPS) i simplement al port 80 es veu com ha afegit la sentència del *redirect*, que simplement farà que les peticions que arribin pel port 80, es redirigeixin al port 443 a través del protocol HTTPS.

5.6. Nom de domini

Tal com s'ha anat veient al llarg de la memòria, sempre s'ha fet referència a un nom de domini, tant quan s'ha parlat de les URLs de les peticions dels microcontroladors, com quan s'ha parlat de la configuració del *Certbot*. Per aquest projecte s'ha volgut adquirir un nom de domini que faci referència al projecte VITALS i que tingui el domini *.com*. És per això que s'ha utilitzat el servei d'allotjament *Hostinger* per adquirir el nom de domini propi. Aquest ha sigut el següent.

- vitalstechnology.com

Un cop acceptat i validat el nom de domini, s'ha hagut d'enllaçar amb la IP pública que ha proporcionat la màquina virtual AWS a l'hora de crear la màquina EC2. LA IP en qüestió és 13.38.34.202.

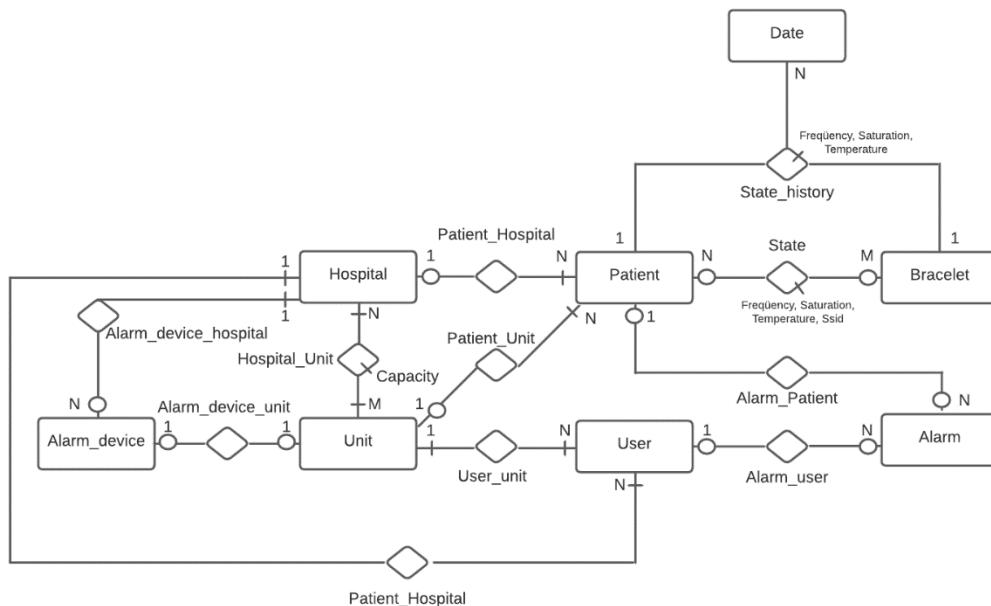
5.7. Base de dades

La base de dades relacional s'ha creat a través d'una extensió de Flask anomenada SQLAlchemy. Aquesta és un mapejador relacional d'objectes (ORM) el qual permet convertir llenguatges de programació orientats a objectes en la seva representació en bases de dades relacionals a través de la definició de les correspondències entre els diferents sistemes. A part, permet interactuar amb gran varietat de bases de dades, permetent la creació de models de dades i consultes que es veuen com classes normals i sentències de Python.

Aquesta llibreria treballa sobretot amb bases de dades comunes com és Postgres, MySQL, SQLite i Oracle[44][45].

L'avantatge principal d'utilitzar aquesta llibreria és la facilitat en la seva sintaxi i el conjunt de sentències i tipus de dades que utilitza per assegurar-se que les seves sentències SQL són creades de forma adequada i eficient per cada base de dades sense necessitat de pensar en ell[45]. El llenguatge que usa s'anomena *SQL Expression Language*, aquest és la forma que té Python per representar les sentències i expressions SQL[45].

La base de dades del projecte té un total de 10 taules. A la Il·lustració 17 es mostra l'esquema entitat/relació d'aquesta base de dades.



Il·lustració 17: Esquema Entitat/Relació base de dades

Tot seguit s'explicaran les relacions entre les diferents entitats, així com la tipologia dels seus atributs.

5.7.1. Taula User

Aquesta taula emmagatzema la informació dels usuaris del sistema, és a dir, el personal d'administració de l'hospital, els professionals de salut i els administradors del sistema. En total la taula té 11 atributs que es detallaran a continuació.

- Id: Enter i clau primària. És l'atribut que identifica els usuaris del sistema.
- Dni: Cadena de caràcters. Conté el número i lletra del DNI de l'usuari.
- Name: Cadena de caràcters. Correspon al nom de l'usuari.
- Surname1: Cadena de caràcters. Correspon al primer cognom de l'usuari
- Surname2: Cadena de caràcters. Correspon al segon cognom de l'usuari
- Email: Cadena de caràcters. Correspon al correu de l'usuari
- Phone: Cadena de caràcters. Correspon al número de telèfon de l'usuari
- Password: Cadena de caràcters. Correspon a la contrasenya de l'usuari. Es guardarà el Hash en base de dades per tal de fer el sistema més segur.
- Hospital: Enter. Fa referència a la identificació d'un hospital. És clau forana de la taula Hospital.
- Unit: Enter. Correspon al filtre d'unitat que ha seleccionat l'usuari per última vegada. D'aquesta manera sempre es mostrerà per defecte l'última unitat que s'hagi cercat i no es necessitarà filtrar per unitat tots els filtres de la plataforma.
- Rol: Cadena de caràcters. S'indica el rol que pot prendre l'usuari: “Administratiu/va”, “Sanitari” o “Superadmin”.

Aquesta taula es relaciona amb la taula *Alarms*, ja que hi ha un atribut d'aquesta taula que indica l'usuari que ha apagat l'alarma activa dins del panell d'alarmes perquè en quedí constància.

5.7.2. Taula Patient

La taula *Patient* conté tota la informació dels pacients de l'hospital que han estat ingressats, independentment de si els hi ha estat donada l'alta o no. La taula té un total de 18 atributs que es detallaran a continuació.

- Id: Enter i clau primària. És l'atribut que identifica els pacients.
- Dni: Cadena de caràcters. Correspon al DNI del pacient.
- Name: Cadena de caràcters. Correspon al nom del pacient.
- Surname1: Cadena de caràcters. Correspon al primer cognom del pacient.
- Suername2: Cadena de caràcters. Correspon al segon cognom del pacient.
- Age: Cadena de caràcters. Correspon a l'edat del pacient.
- Email: Cadena de caràcters. Correspon al correu electrònic del pacient.
- Hospital: Enter. Correspon a l'hospital on està ingressat l'usuari. Aquest ve determinat per l'Hospital que té com atribut l'usuari que li realitza el registre. És una clau forana de la taula Hospital.

- Unit: Enter. És la unitat on està ingressat el pacient. És una clau forana de la taula Unit.
- Room: Enter. És el número d'habitació on està ingressat l'usuari.
- Phone: Cadena de caràcters. És el número de telèfon de contacte del pacient.
- State: Cadena de caràcters. Informa si el pacient està ingressat o no. En el cas que el valor sigui “admit” indicarà que el pacient està ingressat. En el cas que el valor sigui “discharged” indicarà que el pacient ha estat donat d’alta.
- Linked: Booleà. Indica si el pacient està enllaçat a una Bibo band o no. En cas que sigui 1 indicarà que està enllaçat i en cas de 0 indicarà que no està enllaçat.
- Min_temp: Flotant. Indica el llindar mínim de temperatura corporal normal per aquest pacient. Tot valor per sota d'aquest llindar generarà una alarma.
- Max_temp: Flotant. Indica el llindar màxim de temperatura corporal per aquest pacient. Tot valor per sobre d'aquest llindar generarà una alarma.
- Min_freq: Enter. Indica el llindar mínim de freqüència cardíaca normal per aquest pacient. Tot valor per sota d'aquest llindar generarà una alarma.
- Max_freq: Enter. Indica el llindar màxim de freqüència cardíaca normal per aquest pacient. Tot valor per sobre d'aquest llindar generarà una alarma.
- Min_spo2: Enter. Indica el llindar mínim de saturació d’oxigen en sang normal per aquest pacient. Tot valor per sota d'aquest llindar generarà una alarma.

Aquesta taula es relaciona amb les taules *States*, *Satates_history* i *Alarm*. Com es veurà a continuació les 3 taules requereixen l’identificador del pacient per la seva gestió.

5.7.3. Taula Bracelet

La taula *Bracelet* conté tota la informació referent a les Bibo bands i a la seva configuració. La taula té un total de 8 atributs que es detallaran a continuació.

- Bracelet_id: Cadena de caràcters i clau primària. Correspon a l’identificador del microcontrolador de la Bibo band.
- Linked: Booleà. Indica si el dispositiu està enllaçat a algun pacient o no.
- Ssid: Cadena de caràcters. Correspon al *ssid* de la xarxa Wi-Fi on s’ha de connectar la Bibo band.
- Ssid_pass: Cadena de caràcters. Correspon a la contrasenya de la xarxa Wi-Fi on s’ha de connectar la Bibo band.
- State: Cadena de caràcters. Informa l'estat del dispositiu i de la informació rebuda. En cas que hi hagi un “OK” indica que el dispositiu està engegat i està enviat les lectures de les dades de forma correcta. En cas que hi hagi un “NOK” indica que el dispositiu està engegat i funcionant, però les mesures de les constants estan generant error, possiblement per una mala col·locació del dispositiu en el pacient. Finalment si l'estat és “OFF” indica que el dispositiu està apagat i fa estona que no està enviant dades.
- Date: Data i hora. Informa de la data i la hora de la última identificació del dispositiu sense estar enllaçat a cap pacient.

- Measure_time: Enter. Informa del temps en que la Bibo band ha d'estar en mode *Deep Sleep* entre mesures. Aquesta dada ve donada en format de segons.
- Battery: Enter. S'indica del % de bateria que li queda al dispositiu.

Aquesta taula es relaciona amb la taula *State*, ja que és la taula on s'enllacen les Bibo bands amb els pacients.

5.7.4. Taula Alarm_Device

La taula *Alarm_device* conté tota la informació i configuració referent als dispositius d'alarma. La taula té un total de 10 atributs que es detallaran a continuació.

- Device_id: Cadena de caràcters i clau primària. Correspon a l'identificador del microcontrolador del dispositiu d'alarma.
- Linked: Booleà. Indica si el dispositiu d'alarmes està enllaçat a alguna unitat o no.
- Ssid: Cadena de caràcters. Correspon al *ssid* de la xarxa Wi-Fi on s'ha de connectar el dispositiu d'alarmes.
- Ssid_pass: Cadena de caràcters. Correspon a la contrasenya de la xarxa Wi-Fi on s'ha de connectar el dispositiu d'alarmes.
- State: Cadena de caràcters. Informa l'estat del dispositiu d'alarma. En cas que hi hagi un “OK” indica que el dispositiu està engegat i està generant les peticions de forma correcta. Si l'estat és “OFF” indica que el dispositiu està apagat i fa estona que no està fent peticions.
- Hospital: Enter. Correspon a l'identificador de l'hospital on està assignat aquest dispositiu d'alarma. Aquest se li posa de forma automàtica al assignar-lo a una unitat, ja que agafa l'hospital assignat de l'usuari que realitza l'assignació. És una clau forana de la taula Hospital.
- Unit: Enter. Correspon a l'identificador de la unitat on està enllaçat el dispositiu d'alarma. És una clau forana de la taula Unitat.
- Real_ssid: Cadena de caràcters. Correspon al *ssid* de la xarxa Wi-Fi en la que està connectat el dispositiu d'alarma.
- Measure_time: Enter. Correspon al temps d'espera en segons que ha de passar entre les consultes d'alarma dels dispositius d'alarma.
- Date: Data i hora. Informa de la data i la hora de la última identificació del dispositiu sense estar enllaçat a cap unitat.

5.7.5. Taula State

Aquesta taula és la que emmagatzema una de les principals funcionalitats d'aquest projecte. En ella hi ha les relacions entre els pacients i les Bibo bands. És una taula dinàmica i en temps real, és a dir, els seus registres són temporals, ja que estaran a la taula mentre el pacient estigui enllaçat a una Bibo band. En el moment que es desenllaça el dispositiu del pacient, s'elimina automàticament el registre. A part de tenir aquesta relació, s'hi troben altres atributs clau, com són la freqüència cardíaca, la saturació d'oxigen i la temperatura del pacient en temps real. Per tant, cada vegada que es reben dades noves sobre les constants dels pacients, aquestes dades es modifiquen per les més actuals.

Aquesta taula s'ha creat principalment amb la intenció que hi hagi pocs registres (només el nombre de pacients que estiguin enllaçats a una Bibo band). Això és degut al fet que la plataforma web està consultant constantment aquesta taula per tal de mostrar la informació de temps real a la pàgina web. Al ser una taula amb pocs registres, les consultes són molt més ràpides en comparació a si es consulten a l'historic de dades que es veurà a continuació i que té una informació bastant similar.

En concret la taula State té un total de 6 atributs que es detallaran a continuació.

- Patient: Enter i clau primària. Correspon a l'identificador del pacient que està ingressat i enllaçat a una Bibo band. És clau forana de la taula Patient.
- Bracelet: Enter i clau primària. Correspon a l'identificador de la Bibo band a la qual està enllaçat el pacient. És clau forana de la taula Bracelet.
- Frequency: Enter. Correspon a l'últim valor de la freqüència cardíaca rebut des de la Bibo band.
- Saturation: Enter. Correspon a l'últim valor de la saturació d'oxigen rebut des de la Bibo band.
- Temperature: Flotant. Correspon a l'últim valor de la temperatura rebut des de la Bibo band.
- Ssid: Cadena de caràcters. Correspon al *ssid* de la xarxa a la qual està connectada la Bibo band.

5.7.6. Taula State_history

La taula *State_history* emmagatzema el registre de totes les mesures realitzades per les Bibo bands als pacients. Gràcies a aquesta taula, es pot consultar la informació històrica dels pacients i del seu estat en el llarg del temps. Aquesta també és una de les funcionalitats principals del projecte, ja que és interessant poder tenir un registre de totes aquestes dades per poder tenir informació extra sobre l'estat del pacient que no es tindria sense la implementació d'aquest sistema. Aquesta taula té un total de 5 atributs que es detallaran a continuació.

- Patient: Enter i clau primària. Correspon a l'identificador del pacient del qual s'ha realitzat una entrada a l'historic. És clau forana de la taula Patient.

- Date: Data i hora i clau primària. Correspon a la data i hora de la mesura de les constants vitals del pacient. Aquesta informació serà necessària a la hora de crear l'històric i els gràfics.
- Frequency: Enter. Correspon a la freqüència cardíaca del pacient en aquell dia i hora.
- Saturation: Enter. Correspon a la saturació d'oxigen del pacient en aquell dia i hora.
- Temperature: Flotant. Correspon a temperatura del pacient en aquell dia i hora.

Com es pot observar, la informació que conté aquesta taula és molt similar a la de la taula State. Però per millorar en el rendiment s'ha decidit implementar el sistema amb dues taules, ja que aquesta taula contindrà cada vegada més registres al llarg del temps i les consultes cada vegada seran més costoses computacionalment per obtenir la informació dels pacients en temps real.

5.7.7. Taula Alarm

La taula *Alarm* no deixa de ser un històric de les alarmes que s'han generat al llarg del temps. En ella s'hi troba informació extra d'aquestes alarmes tal com es detallarà a continuació amb l'explicació dels seus 10 atributs.

- Id: Enter i clau primària. Correspon a l'identificador de cada registre d'alarma.
- Patient: Enter. Correspon a l'identificador del pacient que ha generat l'alarma. És clau forana de la taula Patient.
- Frequency: Enter. Correspon a la freqüència cardíaca del pacient en el moment de generar l'alarma.
- Saturation: Enter. Correspon a la saturació d'oxigen del pacient en el moment de generar l'alarma.
- Temperature: Flotant. Correspon a la temperatura del pacient en el moment de generar l'alarma.
- Datetime_alarm: Data i hora. Correspon a la data i hora en la qual s'ha generat l'alarma.
- User_checked: Enter. Correspon a l'identificador de l'usuari que ha aturat manualment l'alarma. És clau forana de la taula User.
- Datetime_user_checked: Data i hora. Correspon a la data i hora en que l'usuari ha aturat manualment l'alarma.
- Recovery: Booleà. Indica si l'usuari s'ha recuperat de l'alarma actual o no. Aquest fet es donarà de forma automàtica en el moment que totes les dades de les constants vitals rebudes del pacient estiguin dins dels seus límits. Això indicarà que el pacient s'ha recuperat i està preparat per tornar a generar una nova alarma en el moment que aquests valors tornin a estar fora dels límits configurats.
- Datetime_recovery: Data i hora. Fa referència a la data i hora en que el pacient s'ha recuperat de l'alarma tenint tots els valors de les constants vitals dins dels límits configurats.

Aquesta és la taula principal per la funcionalitat de les creacions d'alarmes. Tal com s'ha comentat a l'explicació dels atributs. Una alarma es generarà en el moment que alguna de les constants del pacient estigui fora dels llindars configurats per ell. Això farà que es generi un nou registre en aquesta taula i, per tant que s'activi el dispositiu d'alarmes. Un cop l'usuari desactivi manualment l'alarme de la plataforma, el dispositiu d'alarmes es desactivarà i no serà fins que el pacient s'hagi recuperat del tot i l'atribut *recovery* estigui a *true*, que aquest pacient no tornarà a generar una nova alarma.

5.7.8. Taula Hospital

Aquesta taula conté la informació general dels hospitals que utilitzen la plataforma. Consta d'un total de 5 atributs que es detallaran a continuació.

- Id: Enter i clau primària. Correspon a l'identificador de l'hospital dins el sistema.
- Name: Cadena de caràcters. Correspon al nom de l'hospital.
- City: Cadena de caràcters. Correspon a la ciutat on es troba l'hospital.
- Adress: Cadena de caràcters. Correspon a l'adreça on es troba l'hospital.
- Capacity: Enter. Correspon a la capacitat de pacients que accepta l'hospital.

Tal com s'ha vist a la resta de taules. Aquesta es relaciona amb la taula *Users*, *Patients*, *Alarm_device* i *Hospital_units*.

5.7.9. Taula Unit

La taula Unit conté la informació de les diferents unitats mèdiques que es troben en el sistema i en els diferents hospitals. Consta d'un total de 2 atributs que es detallaran a continuació.

- Id: Enter i clau primària. Correspon a l'identificador de la unitat mèdica dins el sistema.
- Name: Cadena de caràcters. Correspon al nom de la unitat.

Aquesta taula es relaciona amb les taules *Users*, *Patients*, *Alarm_device* i *Hospital_units*.

5.7.10. Taula Hospital_unit

Aquesta taula és la que conté les relacions entre hospitals i unitats, ja que no tots els hospitals del sistema tenen perquè tenir totes les unitats de la base de dades. És per això que s'ha requerit aquesta taula per poder indicar la relació i la capacitat de pacients per cada unitat de cada hospital. La taula té un total de 3 atributs que es detallaran a continuació.

- Hospital_id: Enter i clau primària. Correspon a l'identificador de l'hospital. És clau forana de la taula Hospital.

- Unit_id: Enter i clau primària. Correspon a l'identificador de la unitat mèdica. Es clau forana de la taula Unit.
- Capacity: Enter. Correspon a la capacitat de pacients a cada unitat per cada hospital.

6. Plataforma web

6.1. Tecnologia i llenguatge utilitzat

La plataforma web com bé indica el seu nom, és una pàgina web on es poden visualitzar i interactuar amb totes les funcionalitats del sistema. El principal llenguatge per donar estructura a la pàgina ha estat HTML5. Aquesta és l'última versió d'HTML que inclou models de processament detallats per fomentar implementacions més interpolables. És per això que HTML5 és un candidat per aplicacions multiplataforma[46].

Per tal de donar estil a la plataforma s'ha utilitzat un *framework* de CSS anomenat Bulma. Aquest és un marc de treball i de codi obert que proporciona components del frontal llestos per fer servir i que es poden combinar fàcilment per crear interfícies web receptives[47].

Per realitzar les gestions d'informació dins els HTML, s'ha usat el motor de plantilles Jinja2, aquest proporciona expressions similars a les de Python. Permet la personalització d'etiquetes, filtres, proves i globals. Permet també fer crides de funcions amb arguments sobre objectes. Aquest és el motor de plantilles per defecte de Flask[48].

Per tal de donar dinamisme a la plataforma, s'ha usat el llenguatge JavaScript. Aquest és un llenguatge compilat d'alt nivell. És de tipus dinàmic, orientat a objectes basat en prototips i funcions de primera classe. Admet estils de programació orientats a eventualitats, funcionalitats i imperatius. Aquest també disposa d'interfícies de programació d'applicacions per treballar amb text, dates, expressions regulars i estructures de dades estàndard[49].

Finalment, les consultes periòdiques al servidor s'han dut a terme a través de peticions Ajax. Aquesta modalitat de peticions al servidor es basa en el fet que la petició HTTP es fa des de JavaScript, de manera que l'usuari la pot tractar sense necessitat de mostrar-la directament a la pantalla. Això permet fer actualitzacions del contingut de la pàgina de forma parcial de manera que s'actualitzi aquesta en temps real sense necessitat de recarregar tota la pàgina. El format de dades que utilitzen aquestes peticions és el JSON[50].

6.2. Funcionalitats

6.2.1. Inici de sessió i Registre d'usuaris

L'inici de sessió i registre, es realitza a través de dos formularis HTML5. Aquests formularis envien la informació entrada en els diferents camps, al servidor. Aquest detecta que la petició per aquella ruta es tracta d'un POST i, per tant, agafa tota la informació rebuda per tractar-la. En cas que s'estigui fent un registre, simplement comprova que totes les dades entrades siguin correctes i que el correu electrònic de l'usuari introduït no estigui ja a la base de dades.

Tot seguit guarda les dades, amb el detall que se li aplica una funció criptogràfica de Hash a la contrasenya i es guarda en aquest format a la base de dades. D'aquesta manera, en cas que es rebi un atac maliciós i es pugui tenir accés a la base de dades, és impossible saber la contrasenya dels usuaris del sistema a causa d'aquesta codificació.

Al formulari d'inici de sessió es fa una acció similar. Un cop l'aplicació web rep les dades del formulari d'inici de sessió, comprova que l'usuari estigui a la base de dades, i en cas que sigui així, aplica la funció de Hash a la contrasenya rebuda i la compara amb la que hi ha guardada a la base de dades. En cas que els resultats siguin iguals, es permet l'inici de sessió a l'usuari.

6.2.2. Rols dins la plataforma

Tal com s'ha vist al capítol dels requisits del sistema, la plataforma ha de delimitar l'accés a certa informació als usuaris segons el seu rol dins l'aplicació web. Hi ha els tres següents rols.

- Administratiu/va: Únicament té accés a la inserció de nous pacients i a la gestió de dades encarregada de l'enllaç dels pacients amb les Bibo bands.
- Sanitari/a: Aquest rol compta amb les dues funcionalitats vistes en el rol Administratiu/va i a més a més pot veure les constants dels pacients en temps real, filtrar dins la plataforma per unitat, gestionar les alarmes que es generen, consultar historials de pacients i d'alarmes així com, gestionar tots els pacients dins la plataforma i les configuracions de temporització entre mesures de les Bibo bands.
- Superadmin: Aquest rol és el que té totes les funcionalitats de la plataforma, per tant, a més a més de poder dur a terme les funcionalitats dels altres dos rols, és capaç de poder modificar i configurar les dades de xarxa dels dispositius externs del sistema, ja siguin Bibo bands o dispositius d'alarma, així com, poder gestionar els usuaris del sistema editant les seves dades, eliminant-los o creant-ne de nous.

Aquests sistema de rols és fàcil d'implementar gràcies al motor de Jinja2 i l'atribut rol de la taula *User*. On simplement es mostra o s'amaga la informació segons el rol de l'usuari que ha iniciat sessió.

6.2.3. Actualització de les dades dels pacients en temps real

Una de les principals funcionalitats d'aquest projecte, és poder veure en temps real les dades de les constants vitals dels pacients que s'estan enviant al servidor. Aquestes es veuen des del panell de pacients. Com s'ha comentat al capítol de base de dades, la informació de les constants vitals dels pacients es guarda en una taula dinàmica, la qual s'actualitza amb les dades més recents trameses per les Bibo bands.

La plataforma web s'encarrega d'actualitzar la informació de les dades de la pantalla a través de peticions Ajax consultant aquesta taula dinàmica de la base de dades. Simplement, el que s'ha fet és crear un fitxer HTML que s'encarregui de mostrar únicament el contingut del panell. Gràcies a les peticions Ajax, el que es fa és recarregar el contingut d'aquest panell cada 20 segons. D'aquesta manera, sempre es pot veure en temps real les últimes dades de les constants vitals dels pacients.

6.2.4. Estat dels dispositius externs

El sistema és capaç de saber quan un dispositiu extern, sigui una Bibo band o un dispositiu d'alarmes està encès o apagat. Aquesta funcionalitat es porta a terme de la següent manera. Mentre un dispositiu estigui encès, estarà realitzant peticions de forma periòdica, tant si es tracta d'una Bibo band o d'un dispositiu d'alarmes. És per això que al servidor hi ha una llista global on s'emmagatzemen els identificadors d'aquests dispositius externs cada vegada que fan una petició. El servidor, cada 6 minuts, executa una funció que mira aquesta llista i automàticament posa estat en "OFF" tots aquells dispositius que no estiguin a la llista. Un cop s'ha fet aquesta acció, s'esborren tots els identificadors de la llista i es torna a iniciar el procés.

Aquesta és una manera de saber quins dispositius han estat actius en els últims 6 minuts. Per tant, l'actualització de l'estat real dels dispositius a l'aplicació web es realitzarà cada 6 minuts. És important mencionar que no es permetrà que els temps d'espera entre mesuraments de les constants vitals siguin superiors a 5 minuts. És per això que aquesta comprovació es farà cada 6 minuts.

En cas que les Bibo bands estiguin actives, l'estat dependrà de si es realitzen correctament les lectures o no. En cas que els sensors estiguin en contacte amb la pell i les lectures dels sensors es duguin a terme sense problemes, les Bibo bands estaran emetent l'estat "OK" tal com s'ha vist en el protocol de comunicació, però en el cas que les lectures no siguin correctes perquè no es detecta contacte amb l'usuari, les polseres trameran l'estat "NOK" informant que l'usuari no porta el dispositiu ben col·locat.

6.2.5. Afegir i editar nous pacients

Aquesta funcionalitat es porta a terme d'una manera molt similar a la de l'inici de sessió i el registre d'usuaris. Tant la funcionalitat d'afegir pacients com la d'editar la seva informació es fa a través de formularis HTML5. Els dos formularis envien les dades dels camps omplerts al servidor i aquest és l'encarregat de tractar les dades. En el cas d'afegir nous pacients, simplement es mira que el DNI d'aquell pacient no estigui ja a la base de dades de pacients. En cas que no hi sigui, el pacient s'hi incorpora sense problemes.

Per altra banda, a l'hora d'editar un pacient, apareixen més camps a omplir en comparació a quan s'afegeix. Aquests camps són els dels llindars de les constants vitals que aquell pacient ha de complir perquè no salti una alarma. A l'hora d'afegir un pacient s'afegeixen aquests llindars de forma automàtica següint els estàndards vists en capítols anteriors, ja que possiblement qui realitzi la inscripció d'un pacient a la base de dades no sigui cap professional sanitari.

6.2.6. Enllaçar pacients a Bibo bands i dispositius d'alarms a unitats

Aquesta és una de les funcionalitats que, aparentment sembla més complexa de realitzar, però que s'ha intentat que sigui el més simple possible des de la plataforma web. A la pàgina de gestió de dades, hi ha 3 apartats principals. El primer de tots correspon als dispositius d'alarms sense enllaçar i que estan fent peticions d'identificació, el segon correspon a les Bibo bands que hi ha sense enllaçar i que també estan fent peticions d'identificació i al tercer apartat, hi ha les relacions de pacients amb Bibo bands, és a dir, aquells pacients que estan enllaçats a cada un d'aquests dispositius.

Per tal d'enllaçar un pacient a una Bibo band, el que s'ha de fer és, a l'apartat de Bibo bands actives i no enllaçades, buscar i seleccionar el pacient que es vulgui enllaçar a aquella Bibo band i posteriorment pressionar el botó d'enllaçar. Això generarà una petició de POST a una ruta del servidor on es rebrà l'identificador de la Bibo band i l'identificador del pacient, tot seguit es modificaran els atributs *linked* del pacient i de la Bibo band i s'entrarà un nou registre a la taula *State* indicant que hi ha un nou enllaç fet i la Bibo band està preparada per començar a realitzar mesures.

Si el que es vol és enllaçar un dispositiu d'alarms a una unitat mèdica, simplement el que s'ha de fer és seleccionar la unitat que es vol en el desplegable d'unitats que hi ha dins de cada element de dispositiu d'alarms i pressionar el botó enllaçar. Aquesta acció també generarà un POST a la ruta corresponent del servidor enviant l'identificador de la unitat seleccionada i l'identificador del dispositiu d'alarms. Tot seguit s'introduirà la unitat a la taula *Alarm_device*, en concret al registre on la clau primària sigui l'identificador del dispositiu d'alarms. Això farà que s'enllaci aquella unitat mèdica al dispositiu d'alarms corresponent.

6.2.7. Visualitzar l'històric de pacients

L'històric de pacients es pot visualitzar des de la mateixa pantalla d'històric. En ell simplement s'ha de filtrar pel pacient que es vol veure i apareixerà la informació històrica d'aquest en forma de 3 gràfiques, una per cada una de les constants vitals calculades. Les dades del gràfic es treuen de la taula *State_history*, ja que emmagatzema tota la informació històrica de totes les dades enviades per les Bibo bands de cada pacient. Per tant, simplement el sistema filtra aquesta taula per l'identificador de l'usuari que s'està consultant i obté totes les dades històriques d'aquest.

Per tal de mostrar les dades a les gràfiques, s'ha utilitzat la llibreria de JavaScritp Chart.js. Aquesta és una llibreria gratuïta i de codi obert per la visualització de dades que admet fins a 8 tipus de gràfics.

A part de mostrar gràfiques de línies sobre la informació històrica dels pacients, aquestes també es poden filtrar temporalment per tal de veure les dades en un període temporal més acotat. Aquesta acció es realitza a través dels 4 filtres temporals que es poden aplicar sobre les gràfiques. En ells s'hi troba, data d'inici, data final, hora d'inici i hora final. Un cop es premsiona el botó de filtrar amb les dades seleccionades, el servidor rep un POST amb les dades d'aquests filtres. Tot seguit s'encarrega de consultar la taula *State_history* i agafar aquelles dades que estan compreses entre les dades del filtre seleccionat. Finalment, s'envien aquestes noves dades a la pàgina web i s'actualitza el contingut. Prèviament, el servidor comprova que les dades dels filtres siguin les correctes, és a dir que els filtres escollits tinguin coherència i, per exemple, no s'hagi escollit una data final anterior a una data inicial. En cas de ser així, el servidor avisarà de l'error i es mostrerà en pantalla.

6.2.8. Editar la informació de les Bibo bands i dispositius d'alarms

A la plataforma hi ha un espai reservat per poder editar la informació dels dispositius externs. Tal com s'ha comentat al llarg de la memòria, la informació que es pot editar en ells és: les credencials de la xarxa a la qual s'han de connectar i el temps d'espera entre lectura de dades, en el cas de les Bibo bands i la unitat a la que formen part els dispositius d'alarms.

Aquestes modificacions es fan a través de formularis HTML5. Simplement, s'envien les dades de les noves credencials o del valor de temps d'espera al servidor i aquest ja s'encarrega de modificar les taules *Bracelet* i *Alarm_device*, amb la informació actualitzada. Tot seguit quan un dels dispositius, del qual se li han modificat les dades, faci una petició al servidor, aquest el respondrà amb les noves dades actualitzades perquè ell les gestioni de la forma adient.

6.2.9. Actualització i interacció de les alarmes en temps real

Una altra funcionalitat és la gestió de les alarmes. Tal com s'ha comentat, cada vegada que una Bibo band realitza una lectura de les constants vitals d'un pacient i aquestes no estan dins del llindar que té configurat, es genera una alarma. Les alarmes es poden consultar des del panell d'alarms. Aquestes s'actualitzen de forma automàtica igual que els valors de les constants vitals del panell de pacients. És a dir, cada 20 segons, es du a terme una petició Ajax consultant si hi ha alarmes, per posteriorment, carregar-les i mostrar-les a la pantalla. És en aquest panell on s'hi poden trobar totes les alarmes que hi ha actives per la unitat mèdica que hi hagi filtrada. Cada alarma conté la informació de les constants vitals del pacient, el nom d'aquest, l'habitació i l'hora a la qual s'ha produït l'alarme.

Per tal d'aturar una alarma activa, hi ha un botó a cada alarma, el qual permet aturar-la. Un cop es pressiona el botó, s'envia un POST a la ruta en qüestió del servidor i aquest el que fa és posar l'identificador de l'usuari que l'ha pressionat dins l'alarme corresponent de la taula *Alarms*, així com, la data i la hora en la qual s'ha produït aquesta acció.

Per altra banda, si només es té en compte el llindar de les constants vitals per generar una alarma, aquestes es podrien generar de forma prolongada en el temps. És a dir, si un pacient presenta una temperatura corporal elevada durant tres dies, estaria generant alarmes cada vegada que un professional sanitari l'aturés. És per això, que la taula *Alarms* compta amb l'atribut *recovery* a la base de dades. Aquest camp indica si un pacient s'ha recuperat totalment després d'una alarma, per tant, en cas que un pacient estigui amb una temperatura per sobre del seu llindar durant tres dies, només generarà una alarma el primer dia, i no serà fins que la temperatura torni a estar dins el llindar corresponent, que no podrà generar cap més alarma. Aquestes comprovacions es realitzen cada vegada que la Bibo band envia les dades.

7. Manual d'usuari

Tot seguit es representaran tres manuals destinats als usuaris que utilitzin el sistema dissenyat en aquest treball. Els dos primers manuals estan destinats al hardware del sistema o als dispositius físics, com són les Bibo bands i els dispositius d'alarms. El tercer manual està enfocat en l'ús i les funcionalitats de la plataforma web.

7.1. Manual dels dispositius físics

A continuació es detallaran els manuals d'usuari dels dos dispositius físics que presenta el sistema. Els dos per això tenen una cosa en comú. Es requereix un punt d'accés conegit per realitzar la primera connexió dels dispositius. Un cop feta la primera connexió, ja es connectaran a la xarxa configurada des de la plataforma web. Les credencials de la xarxa per la primera connexió són les següents.

- Ssid : vitals
- Constrasenya: vitalspass

7.1.1. Manual Bibo band

La Bibo band és el dispositiu destinat als pacients i el dispositiu encarregat de realitzar les mesures de les constants vitals d'aquests i enviar-les al servidor. Per tant, aquests dispositius seran portats únicament pels pacients ingressats a l'hospital. El dispositiu és molt fàcil d'utilitzar i consta de 3 parts principals que s'han de tenir en compte.

7.1.1.1. Càrrega de la bateria

Les Bibo bands són dispositius sense fils, i, per tant, disposen d'una bateria pel seu correcte funcionament. Aquesta bateria s'ha de carregar en el moment que el dispositiu es queda sense. Tal com es veu a la Il·lustració 18, la bateria es carrega a través d'un cable microUSB connectat directament a un port USB o al corrent. La polsera indicarà amb una llum blava que la bateria s'està carregant tal com es veu a la Il·lustració 19.



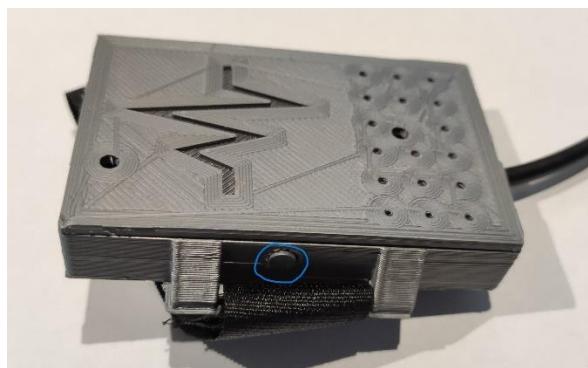
Il·lustració 18: Port microUSB de la Bibo band



Il·lustració 19: Estat de càrrega de la Bibo band

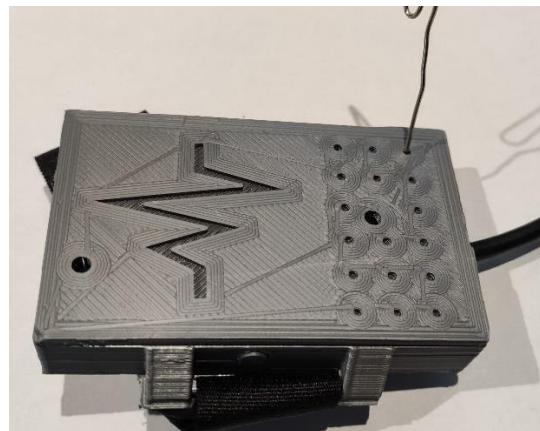
7.1.1.2. Engegar i aturar el dispositiu

La Bibo band és molt senzilla d'engegar i aturar, aquestes accions es fan a través de dos botons. El d'engegar es troba a la part lateral del dispositiu tal com es veu a la Il·lustració 20, de manera que sigui accessible per tothom. Simplement, s'ha de pressionar aquest botó durant 2 segons i el dispositiu ja estarà en funcionament. El dispositiu indicarà amb un parpelleig de color verd que està actiu. Aquest parpelleig el farà cada vegada que es realitzin les mesures per informar de l'activitat de la polsera a simple vista.



Il·lustració 20: Botó d'engegar de la Bibo band

Per altra banda, el botó per aturar el dispositiu està amagat perquè els pacients no hi puguin tenir accés o aquest sigui molt complicat i poc intuïtiu. Com es pot observar a la Il·lustració 21, la polsera presenta un conjunt d'orificis a la seva cara superior. S'ha d'introduir un element fi per l'orífcici situat a la part superior dreta, ja que és on s'hi troba aquest botó. Simplement amb una simple pressió, el dispositiu s'apagarà.



Il·lustració 21: Botó d'apagar de la Bibo band

7.1.1.3. Sensors

La Bibo band és l'encarregada de mesurar la temperatura, la freqüència cardíaca i la saturació d'oxigen dels pacients.

El sensor de temperatura està situat a la part inferior de la polsera tal com es mostra a la Il·lustració 22, és per això que és important que aquesta part faci contacte amb la pell del pacient, per poder tenir les mesures de temperatura més fiables possibles.



Il·lustració 22: Sensor de temperatura de la Bibo band

Per altra banda, el sensor de saturació d'oxigen i freqüència cardíaca està situat al llarg del cable que surt de la part distal de la polsera, tal com es veu a la Il·lustració 23. Aquest sensor s'ha de situar a la cara interna de la falange proximal del dit índex i s'ha d'assegurar que el sensor està en contacte amb la pell. En cas que el sensor no estigui en contacte amb la pell, es veurà reflectit a la plataforma web.



Il·lustració 23: Sensor de freqüència cardíaca i saturació d'oxigen de la Bibo band

7.1.2. Manual dispositius d'alarmes

El dispositiu d'alarmes és un aparell el qual estarà situat a cada unitat. Aquest és l'encarregat d'informar de forma lumínica quan una Bibo band realitzi mesures de les constants vitals d'un pacient i aquestes no estiguin dins els llindars configurats per aquest. És aleshores on el dispositiu emetrà un senyal lluminós fins que algú aturi l'alarma des de la plataforma web.

Per tal de posar en marxa aquest dispositiu, únicament s'han d'endollar els dos cables de la Il·lustració 24 al corrent. Pel que fa a l'USB, aquest es pot connectar a un dispositiu físic amb entrada USB i que estigui alimentat. Un cop fet aquest pas, el dispositiu està en marxa i preparat per funcionar.



Il·lustració 24: Alimentació dels dispositius d'alarmes

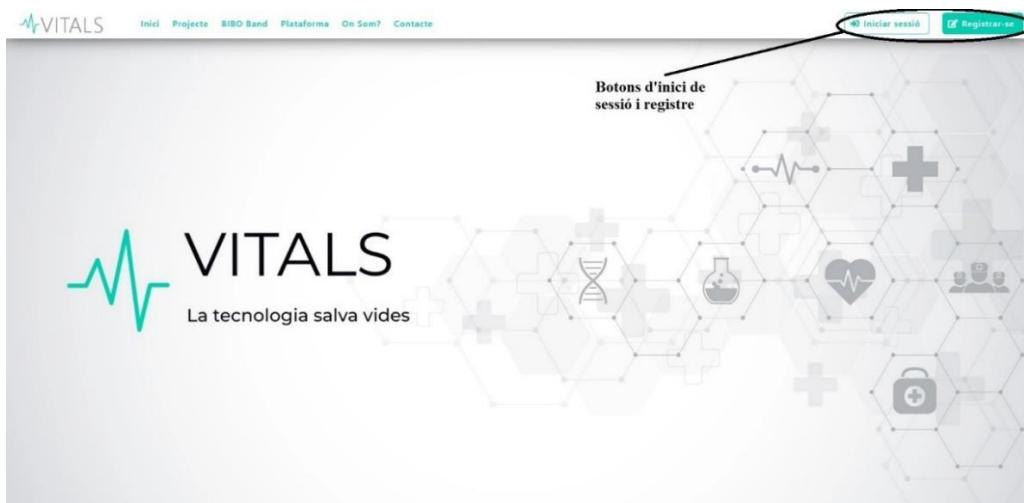
7.2. Manual de la plataforma web

La plataforma web de VITALS és una eina molt potent on es pot gestionar i visualitzar tota la informació del sistema, així com, la configuració dels dispositius físics. Aquesta s'hi accedeix a través d'un navegador web amb la URL de <https://vitalstechnology.com>. A continuació es mostren les diferents pàgines que presenta i les funcionalitats que hi ha en cada una d'elles.

7.2.1. Pàgina inicial

La pàgina inicial és simplement informativa, en ella s'hi pot trobar un petit resum del projecte i de les funcionalitats del sistema de forma genèrica. A part també s'hi troba la ubicació de la universitat així com, les dades de contacte amb el creador del sistema.

Per altra banda, tal com es veu a la Il·lustració 25, en aquesta pàgina s'hi troben els botons d'*Iniciar sessió* i *Registrar-se*. Pàgines que es veuran a continuació.



Il·lustració 25: Pàgina inicial de la plataforma VITALS

7.2.2. Registre

El registre es realitza a través del formulari que es veu a la Il·lustració 26. En ell s'han d'introduir totes les credencials que demana el formulari, així com, acceptar els termes i condicions que hi ha accessibles si es pressiona *termes i condicions*. No es pot deixar cap camp sense omplir, ja que el sistema ho detectarà i no deixarà finalitzar el formulari. També s'ha de tenir en compte que les contrasenyes han de ser les mateixes en els camps que ho demana. El sistema també està preparat per detectar si aquestes no coincideixen.

Finalment, si el correu entrat ja forma part de la base de dades, l'usuari no es crearà i s'informarà per un missatge a la pantalla de què el compte no s'ha pogut crear perquè que el correu electrònic ja existeix.

Quan l'usuari ha estat registrat, depenent del rol escollit (Administratiu/va i Sanitari) entrarà a una pàgina o una altra dins la plataforma web.

Registre
Omple els següents camps per a registrar-te a VITALS.

Nom	Primer cognom
<input type="text"/> El meu nom	<input type="text"/> El meu primer cognom
Segon cognom	DNI
<input type="text"/> El meu segon cognom	<input type="text"/> El meu DNI
E-mail	Telèfon
<input type="text"/> El meu correu	<input type="text"/> El meu telèfon
Hospital	Contrasenya
<input type="text"/> El meu hospital	<input type="text"/> La meva contrasenya
Repeteix la contrasenya	Rol
<input type="text"/> Repeteix la contrasenya	<input type="text"/> Administratiu/va
<input type="checkbox"/> He llegit i accepto els termes i condicions	
Finalitza	

[Torna](#)

Il·lustració 26: Pàgina de registre de la plataforma VITALS

7.2.3. Inici de sessió

A la Il·lustració 27 es pot veure el formulari d'inici de sessió, aquest consta únicament de dos camps. En el primer s'ha d'introduir el correu electrònic de l'usuari i en el segon, la contrasenya escollida. En cas que algun dels camps sigui incorrecte, s'informarà amb un missatge. Un cop iniciada sessió, depenent del rol de l'usuari, s'entrarà a una pàgina o una altra dins la plataforma web.

Com es pot veure a la Il·lustració 27 també, des d'aquest formulari permet la navegació al formulari de registre si es pressiona la paraula *Registra't* a la part inferior del botó *Iniciar sessió*.

Inici sessió
Siusplau, introduceix les teves credencials per a continuar.

Email
<input type="text"/> El meu correu
Contrasenya
<input type="text"/> La meva contrasenya

Iniciar sessió

[Registra't](#) · [Torna](#)

Il·lustració 27: Pàgina d'iniciar sessió de la plataforma VITALS

7.2.4. Perfil

Aquesta és una pàgina comuna per tots els rols. Aquesta permet veure les dades de l'usuari i modificar-les en cas que es vulgui. La pàgina mostra un formulari com el de la Il·lustració 28 amb els camps prèviament omplerts amb les dades de l'usuari. En cas que es vulgui modificar alguna dada, és tan simple com modificar un dels camps i pressionar el botó de *Finalitza*.

Il·lustració 28: Pàgina de perfil de la plataforma VITALS

7.2.5. Panell de pacients

Aquesta és una pàgina exclusiva del rol Sanitari i Superadmin, en concret és la pàgina que es mostra un cop es fa el login a la plataforma. El panell de pacients és la pàgina principal de la plataforma VITALS, en ella es poden veure les constants vitals en temps real dels pacients ingressats a l'hospital. Com es pot observar a la Il·lustració 30, a la part superior hi ha un filtre d'unitat. Aquest filtre permet mostrar únicament els pacients de la unitat que hi ha seleccionada. La informació que es mostra en ell és la següent.

7.2.5.1. Estat

L'estat informa de l'estat en què es troba la polsera. Aquest pot adquirir tres colors tal com es pot veure a la Il·lustració 29 i la informació d'aquests es pot obtenir si es passa el ratolí per sobre de cada un d'ells.

- El verd indica que la polsera està funcionant correctament
- El vermell informa que el pacient s'ha tret la polsera o la porta mal col·locada i el sensor no està adquirint les dades correctament.
- El gris indica que la polsera de l'usuari està apagada i no està enviant dades.



Il·lustració 29: Estat de les Bibo bands dels pacients

7.2.5.2. Dades informatives

A continuació de l'estat, es troben un conjunt de dades que són purament informatives. La primera d'elles és el nom i cognom del pacient. Tot seguit s'hi troba el número d'habitació en la que es troba el pacient. Tot seguit es troben les dades de les constants vitals del pacient en aquell moment determinat, és a dir, les dades de l'última adquisició de dades que ha realitzat la Bibo band. Com es pot veure, els valors poden aparèixer de color negre o de color vermell. En el cas que sigui de color negre indica que el valor està dins dels llindars configurats per aquell pacient. Si el valor és de color vermell informa que el valor està fora del llindar configurat per aquell pacient, i, per tant, informa de què el pacient pot estar en perill.

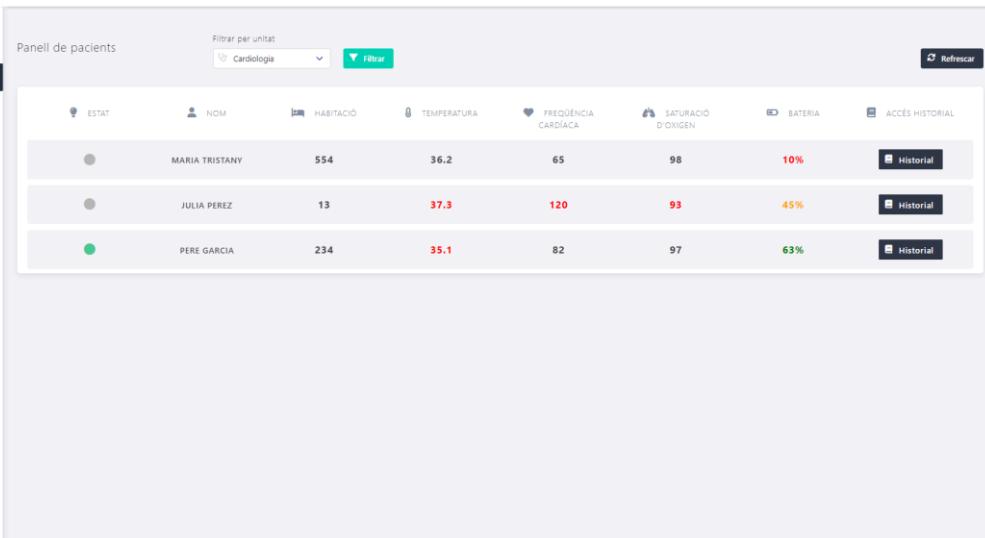
7.2.5.3. Bateria

Tot seguit es troba la bateria del dispositiu, aquesta s'actualitza juntament amb les dades rebudes de les constants vitals. El valor és donat en percentatge i és molt important estar atent per si s'ha de canviar la polsera per una que tingui la bateria carregada. Com es pot observar a la Il·lustració 30, la bateria pot prendre tres colors diferents.

- Verd: Bateria per sobre del 50%.
- Taronja: Bateria entre 20% i 50%.
- Vermell: Bateria per sota del 20%.

7.2.5.4. Botons

A l'última columna del panell s'hi troba el botó *Historial*, aquest accedeix directament a l'historial d'aquell pacient tal com es veurà més endavant. Per altra banda, també s'hi troba el botó de *Refrescar*. Aquest botó s'encarrega de refrescar la pàgina per si hi ha pacients nous que no s'hagin actualitzat i no apareguin al panell.



ESTAT	NOM	HABITACIÓ	TEMPERATURA	FREQUÈNCIA CARDIÀCA	SATURACIÓ D'OXYGEN	BATERIA	ACCÉS HISTORIAL
●	MARIA TRISTANY	554	36.2	65	98	10%	<button>Historial</button>
●	JULIA PEREZ	13	37.3	120	93	45%	<button>Historial</button>
●	PERE GARCIA	234	35.1	82	97	63%	<button>Historial</button>

Il·lustració 30: Panell de pacients de la plataforma VITALS

7.2.6. Panell d'alarmes

Igual que el panell de pacients, aquesta pàgina és exclusiva dels rols Sanitari i Superadmin. El panell d'alarmes serveix per informar de les alarmes que hi ha actives en aquell moment. Aquestes alarmes es poden filtrar per unitat, igual que el panell de pacients, per veure així únicament les que formen part d'una unitat en concret.

Tal com es pot veure a la Il·lustració 31 el panell d'alarmes permet la desactivació d'aquestes a través del botó *Desactivar alarma*. Aquest botó farà desaparèixer l'alarma del panell indicant que ja ha estat vista i supervisada.

En el mateix panell també s'hi mostra altra informació com el nom i cognom del pacient, el número d'habitació d'aquest, les seves constants en temps real i el dia i la hora a la qual s'ha generat l'alarma.

The screenshot shows the 'Panell d'alarmes' (Alarms Panel) interface. On the left is a sidebar with navigation links: Perfil, Panell pacients, Panell alarmes (which is selected and highlighted in dark blue), Històric pacients, Històric alarmes, Gestió de dades, Patients, Dispositius, and Usuaris. At the bottom of the sidebar is a 'Tancar sessió' (Logout) button. The main panel has a header 'Panell d'alarmes' with a dropdown menu 'Filtrar per unitat' set to 'Cardiologia' and a 'Filtrar' button. Below the header is a table with the following data:

	DEACTIVAR ALARMA	NOM	HABITACIÓ	TEMPERATURA	FREQUÈNCIA CARDÍACA	SATURACIÓ D'OXIGEN	DATA I HORA
	Desactivar alarma	PERE GARCIA	234	34.1	59	98	2022-07-05 00:37:26

Il·lustració 31: Panell d'alarmes de la plataforma VITALS

7.2.7. Històric de pacients

Aquesta pàgina és exclusiva dels rols Sanitari i Superadmin. La pàgina de l'històric de pacients és on es visualitza tota la informació recollida per les Bibo bands al llarg de l'ingrés del patient.

Com es pot observar a la part superior de la pàgina, hi ha un filtre de pacient, on es pot escollir el pacient del qual es volen visualitzar les dades.

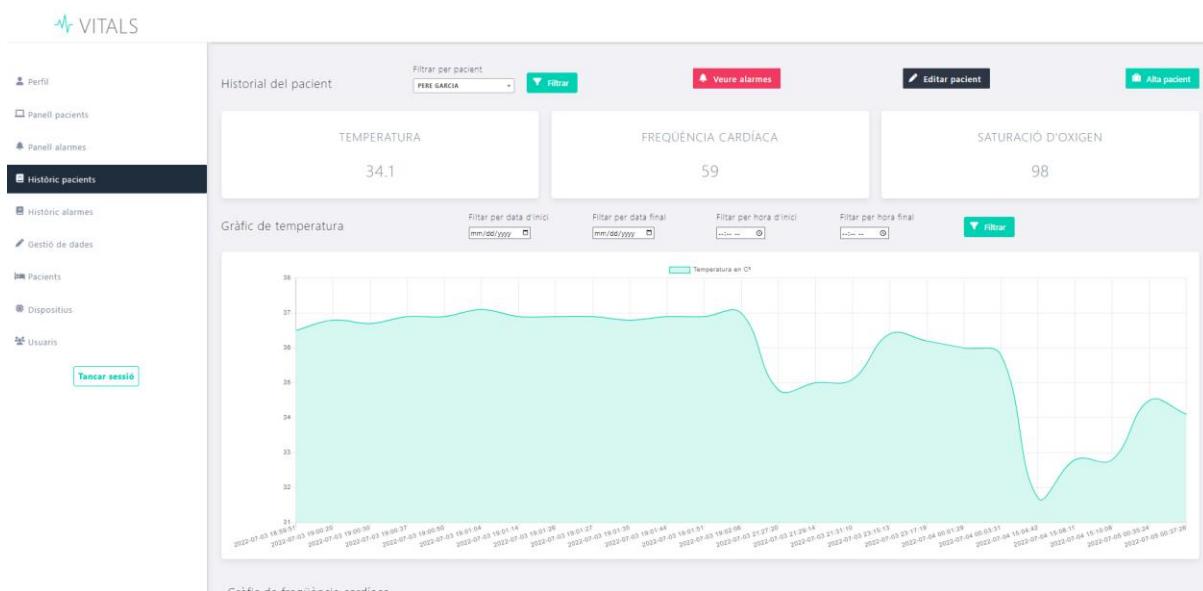
Tot seguit hi ha 3 botons.

- El botó de *Veure alarmes* accedeix a la pàgina de l'històric d'alarmes d'aquell pacient, que, com es veurà a continuació, es pot veure tota la informació de les alarmes que ha generat aquell pacient al llarg del temps.

- El botó d' *Editar pacient* porta a un formulari per editar la informació del pacient tal com es veurà més endavant.
- El botó *Donar d'alta* farà que aquell pacient ja no estigui ingressat a l'hospital, i, per tant, no es podrà enllaçar a cap Bibo band fins que no torni a estar ingressat.

A continuació es poden veure tres quadres on indica la informació de les constants vitals d'aquell pacient en temps real. En el cas que el pacient no estigui enllaçat a cap polsera, aquesta informació no apareixerà, ja que serà desconeguda.

Tot seguit es mostren les gràfiques. Hi ha un total de tres gràfiques que es poden veure si es baixa per la pantalla fent *scroll*. Aquestes gràfiques corresponen a cada una de les constants vitals mesurades amb les Bibo bands. Les gràfiques es poden filtrar a través dels filtres que apareixen sobre la gràfica de la Il·lustració 32. Aquests filtres permeten fer-ho per dies així com, per hores. Els filtres contenen una petita interfície gràfica que permet seleccionar els dies i les hores d'una forma visual i fàcil. Si hi ha un error en les dades introduïdes als filtres, la plataforma informarà de l'error, com per exemple que la data final sigui anterior a la data d'inici del filtre.



Il·lustració 32: Històric pacients plataforma VITALS

7.2.8. Editar pacient

Tal com s'ha vist a la pàgina anterior, la informació d'un pacient es pot editar fàcilment. Un cop es pressiona un dels botons de la plataforma d'*Editar pacient*, s'obrirà un formulari com el de la Il·lustració 33 amb totes les dades del pacient carregades dins els camps. Si es vol modificar alguna dada, simplement s'han de modificar els camps que es vulguin i pressionar el botó de *Finalitzar*. En aquest formulari s'han de tenir en compte varis coses.

- El DNI del pacient no pot estar repetit.
- La temperatura mínima no pot ser superior a la màxima.
- La freqüència cardíaca mínima no pot ser superior a la màxima.
- La saturació d'oxigen mínima no pot ser més gran de 99.

Totes aquestes casuístiques estan contemplades i la plataforma avisarà amb un missatge en cas que es doni algun dels casos. Les dades de les constants que es configurin aquí, són les que es tindran en compte a l'hora de generar alarmes.

Il·lustració 33: Formulari d'edició de pacients de la plataforma VITALS

7.2.9. Històric d'alarmes

Igual que l'històric de pacients, aquesta pàgina és exclusiva del rol Sanitari i Superadmin. A la pàgina de l'històric d'alarmes s'hi pot accedir des del menú de l'esquerra de la plataforma, on posteriorment s'haurà de filtrar pel nom de pacient del qual es vulgui consultar l'històric d'alarmes. També s'hi pot accedir des del botó de *Veure alarmes* situat a la pàgina de l'històric del pacient vist a la Il·lustració 32, el qual directament portarà a la pàgina de l'històric d'alarmes del pacient sense necessitat de filtres.

Aquesta pàgina és simplement informativa, amb la característica del botó de *Veure historial* situat a la part superior dreta de la pantalla, el qual accedeix directament a la pàgina de l'històric del pacient. La informació que proporciona aquesta pàgina és la següent.

- Dia i hora: Informa del dia i l'hora que s'ha generat l'alarma

- Constants vitals: Les tres columnes de constants informen de l'estat del pacient en el moment que es va generar l'alarma. D'aquesta manera queda registrat quin va ser el motiu de l'alarma.
- Alarma aturada: Informa del nom i cognom de la persona que va aturar l'alarma en el seu moment.
- Dia i hora alarma aturada: Informa del dia i l'hora que el professional sanitari va aturar manualment l'alarma.
- Recuperat: Informa si l'usuari s'ha recuperat després de l'última alarma. Això s'aconsegueix en el moment que els valors de les constants del pacient tornen a estar dins els llindars configurats.
- Dia i hora recuperació: Informa del dia i l'hora en què el pacient es va recuperar i va tornar a tenir totes les constants vitals dins els llindars configurats.

DIA I HORA	TEMPERATURA	FREQUÈNCIA CARDÍACA	SATURACIÓ D'OXYGEN	ALARMA ATURADA	DIA I HORA ALARMA ATURADA	RECUPERAT	DIA I HORA RECUPERACIÓ
2022-07-07 20:35:13	34.6	67	99	FERRAN FEIXAS	2022-07-07 20:35:29	NO	NO RECUPERAT
2022-07-07 20:31:27	33.5	67	99	FERRAN FEIXAS	2022-07-07 20:31:44	SI	2022-07-07 20:33:58
2022-07-07 20:28:58	34.8	67	99	FERRAN FEIXAS	2022-07-07 20:29:30	SI	2022-07-07 20:30:13
2022-07-07 20:07:00	2.7	67	99	FERRAN FEIXAS	2022-07-07 20:10:25	SI	2022-07-07 20:10:06
2022-07-07 19:53:13	32.7	67	99	FERRAN FEIXAS	2022-07-07 19:53:23	SI	2022-07-07 19:55:43
2022-07-07 19:50:42	35.2	67	99	FERRAN FEIXAS	2022-07-07 19:51:07	SI	2022-07-07 19:51:58

Il·lustració 34: Històric d'alarmes de la plataforma VITALS

7.2.10. Gestió de dades

La gestió de dades és una pàgina comuna en els tres rols de la plataforma tot i que hi ha algunes diferències entre rols. Aquesta és una de les pàgines més importants, ja que permet enllaçar i desenllaçar dispositius físics amb pacients, en el cas de les Bibo bands, i amb unitats, en el cas dels dispositius d'alarmes.

Com es pot veure a la Il·lustració 35, la pàgina es divideix en tres seccions. A la primera d'elles, la que fa referència als dispositius d'alarmes, apareixen aquells dispositius que no estan enllaçats a cap unitat i que estan actius fent peticions per tal d'enllaçar-se.

En ells es pot veure l'identificador del dispositiu, el dia i l'hora que ha fet la petició per enllaçar-se, un selector per escollir la unitat i dos botons. La funcionalitat del botó d'*Enllaçar* és la de vincular el dispositiu amb la unitat que s'hagi seleccionat en el selector. En el cas que es prensioni el botó d'*Eliminar*, simplement el dispositiu d'alarma desapareixerà de la llista fins que torni a fer una nova petició.

A la segona secció s'hi troba un comportament molt similar amb les Bibo bands. En aquesta secció apareixen aquelles Bibo bands que estan actives i no estan enllaçades a cap pacient, és a dir, aquelles polseres que estan engegades fent peticions d'identificació. En elles s'hi troba l'identificador de la Bibo band, la data en què ha fet la petició i un buscador de pacients per seleccionar aquell que es vol enllaçar. Val a dir que aquest selector només proporciona els pacients que estiguin ingressats i no estiguin enllaçats a cap Bibo band. Tot seguit s'hi troben dos botons, el d'*Enllaçar* realitza la vinculació entre la polsera i el pacient seleccionat i el d'*Eliminar* simplement elimina la polsera de la llista fins que aquesta no torni a fer una nova petició.

Finalment, a l'última secció, s'hi troben aquells pacients que estan enllaçats a polseres, aquests es poden filtrar per unitat per agilitzar la cerca. A cada una de les files de la taula s'hi pot veure l'identificador de la polsera, el nom i cognom del pacient i un seguit de botons. El primer de tots és per modificar la configuració de la Bibo band, tal com es veurà més endavant, en aquest cas aquesta funcionalitat només hi és pels rols Sanitari i Superadmin. El segon botó fa referència al formulari d'edició de pacient vist a la Il·lustració 33. El tercer botó fa referència a donar d'alta el pacient provocant així la seva desvinculació amb aquella polsera i fent que el pacient ja no estigui ingressat, aquesta acció està disponible únicament pels rols Sanitari i Superadmin. Finalment, l'últim botó s'encarrega de desenllaçar la polsera amb el pacient. D'aquesta manera, la polsera tornarà a aparèixer a la secció de Bibo bands per enllaçar i el pacient tornarà a estar disponible per ser enllaçat a una altra polsera.

The screenshot shows the 'Gestió de dades' section of the VITALS platform. It includes three main tables:

- Dispositius d'alarma sense vincular:** Shows a device with ID 2343440716 from July 7, 2022, at 20:38:07. Actions: Asignar (green), Eliminar (red).
- Polseres sense assignar:** Shows a wristband with ID 3816741900 from July 3, 2022, at 21:19:55. Actions: Enllaçar (green), Eliminar (red).
- Pacients i polseres enllaçats:** Shows two linked patient-wristband pairs:
 - ID 12: MARIA TRISTANY (Edit patient, Edit wristband, Alta patient, Desenllaçar)
 - ID 11: JULIA PEREZ (Edit patient, Edit wristband, Alta patient, Desenllaçar)
 A filter for 'Cardiologia' is applied.

Il·lustració 35: Pagina de gestió de dades de la plataforma VITALS

7.2.11. Editar polsera

El botó d'editar polsera des de la pàgina de gestió de dades únicament està disponible pels rols Sanitari i Superadmin i permet modificar únicament el valor del temps d'espera entre mesures de les constants vitals tal com es veu a la Il·lustració 36. Aquest temps s'escull a través d'un selector en el propi formulari. El selector permet escollir el temps d'espera entre 1 i 5 minuts.

Il·lustració 36: Formulari d'edició de polseres simple de la plataforma VITALS

7.2.12. Pacients

Aquesta pantalla també és comuna en els tres rols, de fet és la pàgina que apareix quan es fa el login del rol Administratiu/va. En aquesta pantalla és on s'hi troba el llistat de totes les persones ingressades a l'hospital en aquell moment determinat. Com es pot observar a la part superior de la pantalla, s'hi torna a trobar el filtre d'unitat. D'aquesta manera només apareixeran els pacients ingressats de la unitat per la qual s'apliqui el filtre.

A la dreta d'aquest botó, s'hi troben dos altres botons, el primer de tots i un dels més importants de tota la plataforma és el *d'Afegir un nou pacient*. Tal com es veurà en el següent punt, aquest botó obrirà un formulari per introduir un nou pacient a la plataforma. El botó *Refrescar* únicament recarrega la pantalla per si s'ha introduït algun pacient nou des d'algun altre dispositiu.

Tot seguit es troba el llistat de pacients, el qual aporta la informació bàsica de cada un d'ells, com és el nom, primer cognom i l'habitació on està ingressat. Si es requereix més informació del pacient, es pot pressionar el botó d'*Editat pacient*, i s'obrirà el formulari vist anteriorment a la pàgina d'*Editat pacient*. Aquesta informació es podrà consultar o modificar en cas que es requereixi.

Si es vol consultar l'històric del pacient, simplement s'ha de pressionar el botó d'*Historial* i s'obrirà la pàgina de l'històric de dades filtrat de forma automàtica pel pacient seleccionat. Finalment i com a última acció que es pot dur a terme en aquesta pàgina, és la de donar d'alta un pacient. Aquesta acció es fa pressionant el botó d'*Alta pacient* el qual el donarà d'alta i el farà desaparèixer d'aquesta llista de pacients ingressats.

NOM	HABITACIÓ	EDITAR PACIENT	MOSTRAR HISTORIAL	DONAR D'ALTA
PERE GARCIA	234	<button>Editar patient</button>	<button>Historial</button>	<button>Alta patient</button>
CARLES MARTINEZ	345	<button>Editar patient</button>	<button>Historial</button>	<button>Alta patient</button>
ANNA GARCIA	16	<button>Editar patient</button>	<button>Historial</button>	<button>Alta patient</button>
JOSEP LOPEZ	221	<button>Editar patient</button>	<button>Historial</button>	<button>Alta patient</button>
MARIA TRISTANY	554	<button>Editar patient</button>	<button>Historial</button>	<button>Alta patient</button>
JULIA PEREZ	13	<button>Editar patient</button>	<button>Historial</button>	<button>Alta patient</button>
MARC LOPEZ	65	<button>Editar patient</button>	<button>Historial</button>	<button>Alta patient</button>

Il·lustració 37: Pàgina de pacients de la plataforma VITALS

7.2.13. Afegir un nou patient

Tal com s'ha vist al punt anterior a la pantalla de pacients, és on hi ha un dels botons més importants de la plataforma. Aquest és el d'afegir un nou patient. Aquesta acció únicament es pot fer des de la pantalla de pacients, i un cop pressionat, s'obre el formulari de la Il·lustració 38. Aquest és el formulari que permet afegir un nou patient a la plataforma. Simplement, s'han d'emplenar tots els camps del formulari i pressionar el botó *Finalitza*.

Com es pot veure el formulari és molt similar al d'editar un patient, ja que la informació que es pot afegir i editar és la mateixa. Així i tot, hi ha una petita diferència entre els dos. Al formulari d'edició de pacients hi ha els camps dels llindars de les constants vitals per tal de configurar les alarmes per aquell patient, en canvi, en el formulari de registre de patient, aquests camps no hi són. Això té una fàcil explicació. Normalment, els encarregats d'introduir nous pacients a la plataforma serà el personal administratiu o que tenen el rol Administratiu/va, i, per tant, no tenen perquè saber quins són els llindars òptims de les constants vitals per cada patient. És per això que a l'afegir un nou patient a la plataforma, aquests llindars es posen de forma automàtica aplicant els valors reconeguts de forma mundial com a òptims. Aquests són els següents.

- Freqüència cardíaca: entre 60 i 100 pulsacions per minut.
- Temperatura corporal: entre 36,1 i 37,2 graus centígrads.
- Saturació d'oxigen mínima: 95%

Registre pacient

Omple els següents camps per registrar un nou pacient.

Nom	Primer cognom
Segon cognom	Edat
DNI	E-mail
Unitat	Habitació
Telefó	

Finalitza

Il·lustració 38: Formulari de registre de pacients de la plataforma VITALS

7.2.14. Dispositius

La pantalla de dispositius només està disponible pel rol Superadmin i és similar a la pantalla de pacients, ja que es poden veure tots els dispositius que estan registrats al sistema. En primer lloc, s'hi troben les Bibo bands. La informació que ofereix aquesta pantalla és la de l'estat del dispositiu, tal com s'ha vist a la Il·lustració 29, l'identificador d'aquest, el nom del pacient al qual està enllaçat la polsera, el percentatge de bateria del dispositiu i la informació de la connexió.

La connexió pot prendre 3 formes diferents tal com es veu a la Il·lustració 39. Sempre que es passi el ratolí per sobre de cada un d'aquest símbols, es mostrarà la xarxa real a la qual està connectat el dispositiu.

- Connexió i xarxa correcta: En el cas que aparegui el símbol de connexió de color negre i sense estar ratllat, informa de què la polsera està connectada a la xarxa que està configurada.
- Connexió i xarxa incorrecta: En el cas que aparegui el símbol de connexió de color vermell i sense estar ratllat. Informa que la polsera està connectada, però no a la xarxa a la qual està configurada, per tant, s'ha de modificar la configuració de xarxa d'aquella polsera.
- No connexió: En el cas que la polsera estigui apagada, apareixerà el símbol de connexió ratllat. D'aquesta manera s'informa de què aquell dispositiu no està emetent dades. Informació que també es pot veure amb l'estat del dispositiu.



Il·lustració 39: Estats de la xarxa de connexió de les Bibo bands

Per altra banda, s'hi troben els botons d'*'Editar polsera'* i d'*'Eliminar polsera'*. El primer d'ells obra el formulari vist a la Il·lustració 33 encarregat de l'edició de dades de la polsera i el segon elimina el dispositiu del sistema. Per tant, un cop eliminat desapareix d'aquesta pantalla.

Tot seguit hi ha els dispositius d'alarma, la informació que proporcionen és molt similar a la de les Bibo bands, amb el detall que el dispositiu d'alarma mostra la unitat a la qual està enllaçat i no el pacient, ja que aquest s'enllaça a unitats mèdiques. També es poden veure els dos botons amb les mateixes funcionalitats que les polseres.

Les dades de les polseres i dels dispositius d'alarmes es poden modificar de forma individual, tal com s'ha vist fins al moment, o de forma global, pressionant el botó d'*'Editar polseres massivament'* o *'Editar dispositius d'alarma massivament'*. Aquests botons simplement obriran un formulari on únicament es podrà modificar la xarxa a la qual estan connectats. Per tant, de forma massiva, la configuració de la xarxa és l'única que podrà ser modificada.

Polseres						
ESTAT	ID POLSERA	PACIENT ENLLAÇAT	BATERIA	CONNEXIÓ	EDITAR POLSERA	ELIMINAR POLSERA
●	ID 12	MARIA TRISTANY	10%	⚡	<button>Edita polsera</button>	<button>Elimina polsera</button>
●	ID 11	JULIA PEREZ	45%	⚡	<button>Edita polsera</button>	<button>Elimina polsera</button>
●	ID 3816741900	CARLES MARTINEZ	89%	⚡	<button>Edita polsera</button>	<button>Elimina polsera</button>
●	ID 6	NO ENLLAÇAT	54%	⚡	<button>Edita polsera</button>	<button>Elimina polsera</button>

Dispositius d'alarma						
ESTAT	ID DISPOSITIU	UNITAT	CONNEXIÓ	EDITAR DISPOSITIU	ELIMINAR DISPOSITIU	
●	ID 2343440716	CARDIOLOGIA	⚡	<button>Edita dispositiu</button>	<button>Elimina dispositiu</button>	

Il·lustració 40: Pàgina de dispositius de la plataforma VITALS

7.2.15. Editar polseres

El botó d'editar polsera té un funcionament molt similar al d'editar pacient. Un cop pressionat s'obrirà un formulari amb les dades de configuració d'aquell usuari carregades a cada camp tal com es mostra a la Il·lustració 41. Els camps que es poden modificar són els següents.

- Ssid: Correspon al nom de la xarxa a la qual s'han d'intentar connectar les polseres.
- Contrasenya ssid: Correspon a la contrasenya de la xarxa a la qual s'han d'intentar connectar els dispositius.

- Temps entre mesura de constants: Desplegable en el qual es pot seleccionar valors entre 1 i 5 indicant els minuts en el qual la polsera ha d'estar esperant entre lectura de mostres.

Un cop modificats els camps adients, s'ha de pressionar el botó de *Finalitza* perquè els canvis es portin a terme.

Editar polsera
Omple els següents camps per editar la configuració de la polsera.

SSID	Contraseña SSID
<input type="text" value="vitals"/>	<input type="password" value="vitalspass"/>
Temps entre mesura de constants (en minuts entre 1 i 5)	
<input type="button" value="1"/>	

Finalitza

Il·lustració 41: Formulari d'edició de polseres de la plataforma VITALS

7.2.16. Editar dispositiu d'alarms

Aquesta funcionalitat és exclusiva del rol Superadmin. Igual que les Bibo bands, la configuració dels dispositius d'alarma es pot modificar a través del botó *Edita dispositiu* vist a la Il·lustració 40. Aquest obra un formulari amb 4 camps tal com es veu a la Il·lustració 42. Com es pot veure, els dos primers camps corresponen a la configuració de la xarxa. El tercer camp correspon a la unitat mèdica a la qual estan assignats aquests dispositius podent desplegar el selector i modificar-la sense problemes. Finalment, s'hi troba el temps d'espera entre peticions de consulta d'alarms. Aquest s'ha d'introduir en segons.

Editar dispositiu d'alarma
Omple els següents camps per editar la configuració del dispositiu d'alarma.

SSID	Contraseña SSID
<input type="text" value="vitals"/>	<input type="password" value="vitals"/>
Unitat	Temps d'espera (en segons)
<input type="button" value="Cardiologia"/>	<input type="button" value="15"/>

Finalitza

Il·lustració 42: Formulari d'edició de dispositius d'alarms de la plataforma VITALS

7.2.17. Usuaris

La pantalla d'usuaris és visible únicament pel rol de Superadmin. Aquesta pantalla mostra la informació de tots els usuaris del sistema, com són el nom i cognoms, el correu, el número de telèfon i el rol que aquest té dins la plataforma. El rol de Superadmin, per tant, pot editar la informació dels usuaris a través del botó d'*'Editar usuari'*, els pot eliminar a través del botó *'Eliminar usuari'* o pot afegir-ne de nous a través del botó *'Afegir usuari'*, el qual obrirà un formulari molt semblant al de la Il·lustració 26.

NOM	CORREU	TELEFON	ROL	EDITAR USUARI	ELIMINAR USUARI
TEST TEST	test@gmail.com	679030356	Sanitari/a	<button>Edita usuari</button>	<button>Elimina usuari</button>
FERRAN FEIXAS	ferran.f6f@gmail.com	679030356	Superadmin	<button>Edita usuari</button>	<button>Elimina usuari</button>
TEST2 TEST2	test2@gmail.com	679030356	Administratiu/va	<button>Edita usuari</button>	<button>Elimina usuari</button>

Il·lustració 43: Pàgina d'usuaris de la plataforma VITALS

7.2.18. Editar usuari

Tal com s'ha vist a la Il·lustració 43 el rol de Superadmin pot editar els usuaris de la plataforma. Aquesta edició es fa des del formulari de la Il·lustració 44. Per seguretat, es poden editar totes les dades menys l'hospital al qual pertanyen i la contrasenya d'aquest. La funcionalitat més interessant és la de poder modificar el rol de l'usuari on aquest pot ser un dels tres (Administratiu/va, Sanitari, Superadmin), a diferència del formulari de registre de la Il·lustració 26 on només es podia escollir Administratiu/va o Sanitari, per evitar que es puguin registrar usuaris autònomament amb el rol de Superadmin.

Un cop editades les dades de l'usuari en qüestió, simplement es pressiona el botó *Finalitza*. En el cas que el correu electrònic ja existeixi a la plataforma, apareixerà un missatge i no deixarà modificar les dades entrades.

Editar usuari

Omple els següents camps per editar l'usuari.

Nom	Primer cognom
<input type="text" value="TEST"/>	<input type="text" value="TEST"/>
Segon cognom	DNI
<input type="text" value="TEST"/>	<input type="text" value="1234567890"/>
E-mail	Telèfon
<input type="text" value="test@gmail.com"/>	<input type="text" value="679030356"/>
Rol	
<input type="text" value="Sanitari/a"/>	

Finalitza

Il·lustració 44: Formulari d'edició d'usuaris de la plataforma VITALS

8. Conclusions

El projecte VITALS ha estat un treball molt interessant de dur a terme a causa de la gran afinitat personal pel món de la salut i la tecnologia amb un resultat molt acceptable, i que compleix quasi tots els requisits plantejats que ha de tenir un sistema d'aquestes característiques i amb el repte d'utilitzar tecnologies i eines únicament treballades a la universitat. Com ja s'ha comentat a l'inici de la memòria, cada vegada més, la salut i la tecnologia van agafades de la mà. Aquesta unió permet facilitar el treball als professionals sanitaris i millorar els diagnòstics i tractaments pels pacients. Penso que no es tardarà a començar a veure sistemes similars als d'aquest projecte a hospitals, ja que fora d'aquests, ja es comercialitzen dispositius encarregats de realitzar mesuraments mèdics de forma automàtica i la posterior tramesa de les dades a servidors per consultar la informació.

Aquest projecte és un treball de final de grau, amb les limitacions que això comporta, tant pel que fa a temps de dedicació, com a escala de persones involucrades en ell. Aquest sistema presenta una gran complexitat a l'hora de tenir en compte totes les variables que hi intervenen, així com la gestió de la connexió de dispositius, on algun dia, poden ser centenars. És per això que aquest projecte ha sigut el primer prototip o la primera iteració cap a un sistema o producte que pot tenir un gran potencial. Per aquesta raó, aquest prototip o aquesta primera iteració presenta tants aspectes a millorar o a optimitzar de cara a futures actualitzacions. A continuació es comentaran aquells aspectes que s'han detectat que es poden millorar de cara a un futur.

8.1. Hardware

A nivell de hardware és on hi ha més aspectes a millorar.

8.1.1. Bibo band

- Disseny més petit amb la integració de tots els sensors dins seu.
- Millora en el rendiment de la bateria.
- Disseny pensat perquè pugui ser resistent a l'aigua i a la desinfecció.
- Integració de més sensors per dotar l'aparell de més funcionalitats.
- Disseny més atractiu estèticament
- Millorar el contacte dels sensors amb la pell.
- Millora de l'ergonomia del dispositiu per millorar la comoditat d'aquest.
- Millora per la identificació física dels dispositius de forma visual.

8.1.2. Dispositiu d'alarms

- Disseny més atractiu estèticament.
- Estudiar viabilitat d'alarmes sonores en casos puntuals.
- Disseny d'una PCB per la funcionalitat d'aquests dispositius.

- Millora per la identificació física dels dispositius de forma visual.

8.2. Software

La principal millora en l'àmbit de software està enfocada en la seguretat del sistema i la portabilitat d'aquest.

8.2.1. Servidor

- Migrar el servidor a màquines més potents que acceptin més peticions simultànies.
- Augmentar els sistemes de seguretat d'accés al servidor i a la plataforma.

8.2.2. Plataforma web

- Pàgina web *responsive* per poder-se consultar des de qualsevol dispositiu.
- Creació d'una aplicació per mòbil i tauletes.
- Millorar l'estrètia d'identificació de dispositius físics dins la plataforma.
- Establir contacte amb un hospital per poder creuar les bases de dades i utilitzar la informació en entorns reals i amb usuaris reals.
- Millora l'estètica de la plataforma web.
- Millora d'experiència d'usuari de la plataforma web.
- Permetre la descàrrega de l'històric amb format HL7

Aquests són alguns dels molts aspectes que hi ha a millorar en aquest sistema. Personalment m'agrada seguir treballant en ell i implementant aquestes millors de cara a una segona iteració. Segueixo pensant que és un sistema amb molt potencial, i que en un futur serà imprescindible en el món hospitalari tot i no estar pensat en ser un sistema diagnòstic. Estic molt content del resultat obtingut i amb tot el que he pogut aprendre en tots els àmbits de la carrera, així com, tots els coneixements apresos durant aquests quatre anys que s'han pogut posar en pràctica per crear un sistema d'aquestes característiques.

9. Bibliografia

- 1- ul Islam, Saif, et al. 'Implanted Wireless Body Area Networks: Energy Management, Specific Absorption Rate and Safety Aspects'. *Ambient Assisted Living and Enhanced Living Environments*, Elsevier, 2017.
- 2- Gómez, Jorge, et al. 'Patient Monitoring System Based on Internet of Things'. *Procedia Computer Science*, vol. 83, Jan. 2016.
- 3- Jovanov, E., et al. 'A WBAN System for Ambulatory Monitoring of Physical Activity and Health Status: Applications and Challenges'. *2005 IEEE Engineering in Medicine and Biology 27th Annual Conference*, 2005.
- 4- Malasinghe, Lakmini P., et al. 'Remote Patient Monitoring: A Comprehensive Study'. *Journal of Ambient Intelligence and Humanized Computing*, vol. 10, no. 1, Jan. 2019.
- 5- Kim, Yena, and SuKyoung Lee. 'Energy-Efficient Wireless Hospital Sensor Networking for Remote Patient Monitoring'. *Information Sciences*, vol. 282, Oct. 2014.
- 6- Paksumiemi, M., et al. 'Wireless Sensor and Data Transmission Needs and Technologies for Patient Monitoring in the Operating Room and Intensive Care Unit'. *2005 IEEE Engineering in Medicine and Biology 27th Annual Conference*, 2005.
- 7- Varady, P., et al. 'An Open Architecture Patient Monitoring System Using Standard Technologies'. *IEEE Transactions on Information Technology in Biomedicine*, vol. 6, no. 1, Mar. 2002.
- 8- Atta, Raghied M. 'Cost-Effective Vital Signs Monitoring System for COVID-19 Patients in Smart Hospital'. *Health and Technology*, vol. 12, no. 1, Jan. 2022.
- 9- Areia, Carlos, et al. 'Experiences of Current Vital Signs Monitoring Practices and Views of Wearable Monitoring: A Qualitative Study in Patients and Nurses'. *Journal of Advanced Nursing*, vol. 78, no. 3, Mar. 2022.
- 10- ESP32-WROOM-32 Datasheet, Espressif Systems, version 3.3, 2022.
- 11- Inam, Rafia, et al. 'Support for Hierarchical Scheduling in FreeRTOS'. *ETFA2011*, 2011.
- 12- Ashisha, G. R., et al. 'IoT-Based Continuous Bedside Monitoring Systems'. *Advances in Big Data and Cloud Computing*, edited by J. Dinesh Peter et al., Springer, 2019.
- 13- Bertsch, J. Michael, and Stephen P. Gent. 'Design of a Wearable Health Monitoring System for In-Home and Emergency Use'. *2020 Design of Medical Devices Conference*, American Society of Mechanical Engineers, 2020.
- 14- Kleiger, Robert E., et al. 'Time Domain Measurements of Heart Rate Variability'. *Cardiology Clinics*, vol. 10, no. 3, Aug. 1992.
- 15- Contardi, Uriel Abe, et al. 'MAX30102 Photometric Biosensor Coupled to ESP32-Webserver Capabilities for Continuous Point of Care Oxygen Saturation and Heartrate Monitoring'. *The 2nd International Electronic Conference on Biosensors*, MDPI, 2021.

- 16- Wongsrichanalai, V. 'CORRELATION BETWEEN THE SPO2/FIO2 RATIO AND PAO2/FIO2 RATIO IN ARDS'. *Chest*, vol. 155, no. 4, Apr. 2019.
- 17- M. R. Banks, P. J. Kumar, H. E. Mulcahy. 'Pulse Oximetry Saturation Levels during Routine Unsedated Diagnostic Upper Gastrointestinal Endoscopy'. *Scandinavian Journal of Gastroenterology*, vol. 36, no. 1, Jan. 2001.
- 18- Bento, Antonio Carlos. 'An Experimental Survey with NodeMCU12e+Shield with Tft Nextion and MAX30102 Sensor'. 2020 11th IEEE Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON), 2020.
- 19- Saari, M., et al. 'Experimenting with Means to Store and Monitor IoT Based Measurement Results for Energy Saving'. 2020 43rd International Convention on Information, Communication and Electronic Technology (MIPRO), 2020.
- 20- Guinea, Gustavo V., et al. 'Increases of Corporal Temperature as a Risk Factor of Atherosclerotic Plaque Instability'. *Annals of Biomedical Engineering*, vol. 36, no. 1, Jan. 2008.
- 21- Vaz, A., et al. 'Full Passive UHF Tag With a Temperature Sensor Suitable for Human Body Temperature Monitoring'. *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 57, no. 2, Feb. 2010.
- 22- Mitchell, Neil, et al. 'Building an Integrated Development Environment (IDE) on Top of a Build System: The Tale of a Haskell IDE'. IFL 2020: Proceedings of the 32nd Symposium on Implementation and Application of Functional Languages, ACM, 2020.
- 23- Evans, Brian. *Beginning Arduino Programming*. Apress ; Distributed to the book trade worldwide by Springer Science+ Business Media, 2011.
- 24- Tojeiro Calaza, German, and J. Gerardo Reino B??rto. *TALLER DE ARDUINO. EXPERIMENTANDO CON ARDUINO MKR 1010*. MARCOMBO, 2020.
- 25- Springborg, Anders Aaen, et al. 'Cpp-Tiny-Client: A Secure API Client Generator for IoT Devices'. Proceedings of the 37th ACM/SIGAPP Symposium on Applied Computing, Association for Computing Machinery, 2022.
- 26- Jang, Jisun, and Tomasz Bednarz. 'HoloSensor for Smart Home, Health, Entertainment'. ACM SIGGRAPH 2018 Appy Hour, Association for Computing Machinery, 2018.
- 27- Babiuch, Marek, et al. 'Using the ESP32 Microcontroller for Data Processing'. 2019 20th International Carpathian Control Conference (ICCC), 2019.
- 28- Meje, Kelebogile Confidence, et al. 'Real-Time Power Dispatch in a Standalone Hybrid Multisource Distributed Energy System Using an Arduino Board'. *Energy Reports*, vol. 7, Nov. 2021.
- 29- Maier, Alexander, et al. 'Comparative Analysis and Practical Implementation of the ESP32 Microcontroller Module for the Internet of Things'. 2017.
- 30- Borg, Jonathan C., et al. 'A KICAD Tool for Pro-Active Exploration Support to "Design Synthesis for Multi-X"'. Knowledge Intensive Computer Aided Design: IFIP TC5 WG5.2 Third Workshop on Knowledge Intensive CAD December 1–4, 1998, Tokyo, Japan, edited by Susan Finger et al., Springer US, 2000.

- 31- Chen, Chao, et al. ‘Modeling Wheat and Maize Productivity as Affected by Climate Variation and Irrigation Supply in North China Plain’. *Agronomy Journal*, vol. 102, no. 3, May 2010.
- 32- Gómez-Ortiz, Héctor Iván, et al. ‘Regulador de Rápido Transitorio de Bajo Diferencial de Tensión En Tecnología de 90 Nm’. *ITECKNE*, vol. 12, no. 1, June 2015.
- 33- Dudzik, G., et al. Single-Frequency, Fully Integrated, Miniature DPSS Laser Based on Monolithic Resonator. Edited by W. Andrew Clarkson and Ramesh K. Shori, 2014.
- 34- Zeng, Xiangzhen, and Bin Ren. ‘Experimental Industrial Signal Acquisition Board in a Large Scientific Device’. *IOP Conference Series: Earth and Environmental Science*, vol. 113, Feb. 2018.
- 35- Galos, Joel, et al. ‘Vibration and Acoustic Properties of Composites with Embedded Lithium-Ion Polymer Batteries’. *Composite Structures*, vol. 220, July 2019.
- 36- Vasques, Mayra Torres, et al. ‘Three-Dimensional Printing of Occlusal Devices for Temporomandibular Disorders by Using a Free CAD Software Program: A Technical Report’. *The Journal of Prosthetic Dentistry*, vol. 123, no. 2, Feb. 2020.
- 37- Bonney, Matthew S., et al. ‘Development of a Digital Twin Operational Platform Using Python Flask’. *Data-Centric Engineering*, vol. 3, 2022.
- 38- Chun, Wesley. *Core Python Programming*. 2nd ed, Prentice Hall, 2007.
- 39- Correia, Raúl Cordeiro, et al. ‘Euronet Lab a Cloud Based Laboratory Environment’. Proceedings of the 2012 IEEE Global Engineering Education Conference (EDUCON), 2012.
- 40- Hu, Yiming, et al. ‘Measurement, Analysis and Performance Improvement of the Apache Web Server’. 1999 IEEE International Performance, Computing and Communications Conference (Cat. No.99CH36305), 1999.
- 41- Rubio, Daniel. *Beginning Django*. Apress, 2017.
- 42- Szalachowski, Pawel, et al. ‘PoliCert: Secure and Flexible TLS Certificate Management’. Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security, Association for Computing Machinery, 2014.
- 43- Aas, Josh, et al. ‘Let’s Encrypt: An Automated Certificate Authority to Encrypt the Entire Web’. Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security, Association for Computing Machinery, 2019.
- 44- Myers, Jason, and Rick Copeland. *Essential SQLAlchemy*. Second edition, O’Reilly Media, 2015.
- 45- Gardner, James, editor. ‘Introducing the Model and SQLAlchemy’. The Definitive Guide to Pylons, Apress, 2009.
- 46- Lawson, Bruce, and Remy Sharp. *Introducing HTML5*. New Riders, 2011.
- 47- Rippon, Carl. *ASP.NET Core 5 and React Full-Stack Web Development Using .NET 5, React 17, and Typescript 4*. 2021.
- 48- Lokhande, P. S., et al. *Efficient Way of Web Development Using Python and Flask*. Mar. 2015.
- 49- Jensen, Simon Holm, et al. ‘Type Analysis for JavaScript’. *Static Analysis*, edited by Jens Palsberg and Zhendong Su, Springer, 2009.

50- Nicholus, Ray. ‘AJAX Requests: Dynamic Data and Page Updates’. Beyond JQuery, edited by Ray Nicholus, Apress, 2016.