

Informatika Ingeniaritzako Gradua

Konputagailuen Ingeniaritza

Gradu Amaierako Lana

Gasen presioen datu tratamendurako komunikazio eta datu eskuraketa modulua

Egilea

Pablo Corral Andueza

2021

Informatika Ingeniaritzako Gradua Konputagailuen Ingeniaritza

Gradu Amaierako Lana

Gasen presioen datu tratamendurako komunikazio eta datu eskuraketa modulua

Egilea

Pablo Corral Andueza

Zuzendariak

Jose A. Pascual Saiz eta Borja Calvo Molinos

Laburpena

Euskal Herriko Unibertsitateko Donostiako kimika fakultatean hainbat esperimentu eta ikerketa burutzen dira egunero. Kimikaren arloko ikerketa lanak burutu ahal izateko, beste edozein tokitan erabiltzen ez diren gailu eta material erabiltzen dira. Beste arloetan gertatzen den moduan, erabilitako tresnen hasierako instalaziotik ez da aldaketarik burutzen eta ekipamendua zahartzen da erabiltzailearen deserosotasuna eta gailuen moteltasuna sustatuz.

Arazo hori, kimika fakultatean gasekin esperimentuak burutzeko erabiltzen den gas-ganbera baten ekipamenduarekin gertatu da. Gailu hauen funtzionamendu egokia ezinbestekoa da eta horretarako gaur egungo teknologiak eusteko gai izan behar dira, egunerapen periodikoak jasateko aukerarekin.

Proiektu honen helburua, gas-ganberarekin komunikatzen den ekipamendua berritzea da eguneratutako beste komunikazio modulu berri bat sortuz. Komunikazio modulu berriaren bitartez, presioen datuen eskuraketa kontrolatuko da serie portuan komandoak idatziz eta erantzun-datuak irakurriz. Honekin batera, web modulu bat inplementatu da erabiltzaileak esperimentua maneiatu eta emaitzak jaitsi ahal izateko.

Gaien aurkibidea

Laburpena	i
Gaien aurkibidea	iii
Irudien aurkibidea	vii
Taulen aurkibidea	ix
1 Sarrera	1
1.1 Zer da datu eskuraketa?	1
1.1.1 Zeintzuk dira DAQ sistemen aplikazioak?	2
1.2 Nola funtzionatzen du DAQ sistema batek?	2
1.3 Proiektuaren justifikazioa	4
1.4 Hasierako arazoa	4
1.5 Elikadura eta iragazkortasun presioa	5
1.6 Hardwarea	5
1.6.1 PR4000	5
1.6.2 Raspberry Pi 4	6
1.7 Softwarea	6
1.8 Proiektuaren helburuak	7

2	Aurretik inplementaturik zegoen modulua	11
2.1	Programaren xehetasunak	12
2.1.1	Erabilitako DAQ gailuak	12
2.1.2	Funtzionamendu orokorra	14
3	Komunikazio modulua	19
3.1	Komandoak	19
3.1.1	Kontrol-seinaleak	20
3.1.2	Komandoen bidalketa	21
3.1.3	DAQ gailuaren erantzunen irakurketa	21
3.2	Serie portuaren konfigurazioa	21
3.3	Inplementazioa	22
3.3.1	Datu-egiturak	24
3.3.2	Komunikazioa	26
3.3.3	Emaitzen errepresentazioa	28
4	Web modulua	31
4.1	Egitura	33
4.1.1	Orri nagusia	34
4.1.2	Esperimentu berri bat hasi	34
4.1.3	Jarraibideak	35
4.1.4	Uneko esperimentuaren datuak bistaratu	35
4.1.5	Emaitza-fitxategiak jaitsi	36
4.2	Inplementazioan erabilitako tresnak	36
4.2.1	Nginx	37
4.2.2	HTML	37
4.2.3	PHP	38

4.2.4	Javascript	39
4.2.5	CSS	40
4.2.6	XSLT	40
4.2.7	Bootstrap	41
4.2.8	Chartjs	41
5	Ondorioak	43
5.1	Proiektuaren ondorioak	43
5.2	Ondorio pertsonalak	43
6	Plangintza	45
6.1	LDE diagrama	45
6.2	Faseen deskribapena	46
6.2.1	Kudeaketa	47
6.2.2	Garapena	47
6.2.3	Dokumentazioa	49
6.3	Lanen estimazioak eta desbiderapenak	50
6.4	Arrisku plana	50
6.5	Desbiderapen analisia	51
6.6	Lan-metodologia	52
	Bibliografia	53
A	Inplementazioak	55

Irudien aurkibidea

1.1	Datuak lortzeko prozesua	4
1.2	Proiektuaren diagrama	9
2.1	Termobikote mota	12
2.2	Termobikote motaren xehetasunak	13
2.3	Tentsioa neurtzeko jarraitu beharreko prozesua	14
2.4	Sarrera parametroen egituraketa irteera fitxategian	15
2.5	DAQ gailuaren laginketa-ratioa eta lagin kopurua zehaztu	15
2.6	Irteera bitan banatu Signal Split objektuaren bidez	16
2.7	Temperatura ezberdinen balioak eta grafikoak Front Panel-ean irudikatu	16
2.8	Averaging aplikatu irteera seinaleari eta seinalea bitan banatu	17
3.1	Kontrol-seinaleen konstante posibleak	20
3.2	Konfigurazio fitxategiaren parametro batzuk	22
4.1	Programa nagusitik web zerbitzariak eskuratutako datu-fitxategiak	32
4.2	Web gunearen egitura	33
6.1	Proiektuaren ataza nagusiak	46
6.2	Garapenaren azpiatazak	46

Taulen aurkibidea

1.1	Sentsore motak	3
6.1	Proiektuaren aurreikusitako eta benetako orduak.	50

1. KAPITULUA

Sarrera

Txosten honetan proiektuaren helburuak lortzeko jarraitu diren pausuak, proiektuan zehar erabili diren tresnak, teknologiak eta ataza bakoitzari eskainitako denbora aipatuko dira besteak beste. Hasteko, txostenean zehar aipatuko diren kontzeptu garrantzitsuak azalduko dira.

1.1 Zer da datu eskuraketa?

Datuen eskuraketa edo DAQ prozesua [1] (Data AcQuisition), ordenagailuarekin tentsioa, korronea, temperatura, presioa edo soinua bezalako fenomeno elektriko edo fisikoak neurtzean datza. DAQ sistema baten funtzionamendurako, sentsoreak, DAQ gailu bat eta ordenagailu bat dira beharrezkoak. DAQ gailuak, sentsoreak neurtzen dituen seinale elektrikoak egokituko ditu; bestalde, ordenagailuak eskaintzen dituen software programagariaren bitartez datu horiek ikuskatuko dira.

Neurketa-sistema tradizionalekin alderatuta, ordenagailuetan oinarritutako DAQ sistemek industria-estandarreko ordenagailuen prozesatze-ahalmena, produktibitatea, bistaratzea eta konektibitate gaitasunak aprobeztatzen dituzte neurketa irtenbide indartsuagoa, malguagoa eta kostu eskuragarriagoa eskainiz.

1.1.1 Zeintzuk dira DAQ sistemen aplikazioak?

Atal honetan datu eskuraketak izan ditzakeen aplikazio batzuk ikus daitezke:

- **Datuen bilketa:** Tenperatura, argia, soinua, indarra, posizioa errotazioa, azelerazioa eta abar.
- **Ibilgailu probak:** ABS eta balazta sistemen probak, gidari asistenteen probak, balazten soinuaren probak, ingurune zailen detekzioa, ibilgailuen dinamika, barne errekuntza motorraren neurketa, analisia eta termodinamikaren aplikazioa eta abar.
- **Gailu elektronikoen probak:** Haririk gabeko mugikorren txip probak, zuntz optikoaren neurketa automatikoa, automozio motorren kontrolagailuaren monitorizazioa, azelerometroaren kalibratze sistema automatikoa eta abar.
- **Manufaktura eta kalitate probak:** Produktuen funtzionaltasuna frogatzeko sistema mailan probak egin, gaintu/huts moduko probak burutu kalitatea neurtzeko eta produktu eta azpisistema akastunak egiaztatu.
- **Ekipoen egoeraren monitorizazioak:** Ekipoen epe luzeko etengabeko monitorizazioa, arazoak identifikatu porrota gertatu aurretik eta errorearen baldintzetan oinarrituz, joera, erregistroak eta alarmak aztertu.
- **PC-an oinarritutako automatizazio eta kontrola:** Gizakien elkarreraginik gabeko kontrol prozesuak kontrolatu eta makineriaren funtzionamenduaren automatizazioa burutu.

1.2 Nola funtzionatzen du DAQ sistema batek?

DAQ sistema batek hiru pausu nagusi burutu behar ditu neurtu nahi diren datuak modu irakurgarri batean lortzeko. Hiru pausu horiek burutzeko, beste hiru gailu behar dira: sentsorea, DAQ gailua eta ordenagailua.

- **Sentsorea:** Gailu hau edozein fenomeno fisiko neurtzeko erabiltzen da, hala nola, tenperatura, argia, presioa, indarra eta abar. Gailu honek neurtutako fenomeno fisiko hori, neurtu daitekeen seinale elektriko batean bilakatzen du. Beraz, DAQ sistemetan sentsorearen lana, fenomeno fisikoaren baliokidea den lortutako seinale elektriko DAQ gailuari bidaltzea izango da.

Proiektu honetan presio sentsore bat erabiltzen da gasen presio ezberdinak neurtzeko. 1.1 taulan, DAQ sistema batean normalean aurkitu daitezkeen sentsore motak ikus daitezke.

Temperatura	Termobikoteak, termistoreak, RTD
Argia	Fotosentsoreak
Soinua	Mikrofonoak
Indarra eta Presioa	Galga extensiometroa, transduttore piezoelektrikoa
Posizioa eta desplazamendua	Potentiometroa, LVDT, kodifikadore optikoa
Azelerazioa	Azelerometroa
PH	PH elektrodoa

1.1 Taula: Sentsore motak

- **DAQ gailua:** DAQ gailuaren hardwarea, ordenagailua eta sentsoretik datozen seinaleen arteko interfaze baten moduko lana egiten du. Honen funtzio nagusia, sentsoreek bidalitako seinale analogikoak digitalizatzea da. DAQ gailuak hiru atal nagusi ditu: seinalea egokitzeko zirkuitua, analogiko/digital bihurtzailea eta ordenagailuarekin konektaturiko bus bat.

Aipatutako egokitze zirkuituaren bitartez, seinaleak izan ditzakeen soinuak ezabatu daitezke (kontuan hartu nahi ez direnak), A/D bihurgailuari sarrera egokia bidaltzeko.

A/D bihurgailuak berriz, digitalizazio prozesua burutuko du, egokitze zirkuituak pasatako seinalearekin. Digitalizazioa burutu ahal izateko, seinalea lagindu behar da lehenik (seinalearen zabalaren datuak periodikoki neurtu); seinalea laginduta koan kuantifikazioa burutu behar da (aztertutako seinale baten balio barrutia irteera maila bakar bati esleitu); azkenik kodeketa burutu behar da, kuantifikazioan lortutako balioak kode bitarrera bihurtuz.

DAQ gailuaren azken atala, ordenagailuarekin konektatzeko erabiltzen den busa da. Bus honi esker, ordenagailuak igorritako aginduak jasotzea eta ordenagailuari datuak erantzun moduan bidaltzea posible da.

- **Ordenagailua:** Ordenagailuaren funtzioa, DAQ gailuaren funtzionamendua kontrolatzea da, neurketa-datuak prozesatuz, gordez eta bistaratuz. Ordenagailuak eta DAQ gailuak partekatutako busa, haien arteko datu eta komando bidalketa ahalbidetzen du.

Kasu honetan ordenagailuan kokaturik dagoen programa nagusiak busaren jariora kontrolatuko du DAQ gailuari datuak eskatzeko komandoak bidaliz eta datu errea-

lak eskuratuz. Ondorioz, datu-eskaerak ordenagailuak kudeatzen ditu, sinkronizazio puntu bat lortuz PC eta DAQ gailuaren artean.

Azaldutako prozesu osoaren eskema 1.1 irudian ikus dateke:



1.1 Irudia: Datuak lortzeko prozesua

1.3 Proiektuaren justifikazioa

Proiektu honetan kimika fakultateko laborategi batean dagoen gasen presioak kalkulatzeko erabiltzen den gas-ganbera batekin egingo da lan. Gasen presioa neurtzeko erabiltzen den ganbera ez da arazoa, instalatuta dagoen hardware eta softwarea baizik. Gas-ganbera hori DAQ gailu batekin dago konektatuta 1.2 atalean azaldutako DAQ prozesua burutzeko.

DAQ gailu horrekin komunikatzeko behar den modulua berdiseinatu eta berregingo da C lengoaiaren erabiliz, behar diren komando eta eskakizunetara egokituz. Programatutako modulu hau, beste ordenagailu berriago batean instalatuko da, gaur egungo teknologietara egokituz. Hardware eta softwarearen berriztatzeak, ikertzaile eta ikasleentzat onuragarria izango da, lan egiteko erraztasun handia suposatuko duelako.

1.4 Hasierako arazoa

EHU unibertsitateko kimika fakultatean (Gipuzkoako kanpusean kokatuta) hainbat esperimentu, ikerketa eta ikaslan burutzen dira urtero. Kimika fakultateak ikasle, irakasle eta ikerlarientzat gailu eta zerbitzu eguneratuak izan behar dituzte eskuragarri, beharrezkoak

diren esperimenduak ahalik eta hoberen burutzeko. Orokorrean, urte asko irekita daramatzaten unibertsitate eta ikastola askok hardware eta software zaharkitua erabiltzen dute oraindik, ekipoa lehen aldiz instalatu zenetik ez dutelako ezer aldatu.

Gainera, informatika urte gutxitan asko hazi den arloetako bat da eta ezinbestekoa da noizbehinka eguneraketak burutzea, bai software bai hardwarearen alderdian. Eguneraketa ezak hainbat desabantaila ekar ditzake, hala nola, softwarea eta gaur egungo aplikazioen arteko bateraezintasuna, haustutako hardwarea erreparatzeko jada saltzen ez diren piezen beharra, erabiltzaileen deserosotasuna ekipo zaharkituen ondorioz, ordenagailuaren erantzun-denboraren mantsotasun nabaria eta abar.

1.5 Elikadura eta iragazkortasun presioa

Elikadura eta iragazkortasun presioa txostenean zehar asko aipatzen diren kontzeptuak dira, beraz ezer azaldu baino lehen, hauek argitzea komenigarria da.

Elikadura presioa, gas-ganberaren barruan jarritako mintzaren gainean aplikatzen den sarrerako presioa da. Presio hori gutxinaka handiagotzen da, gasak mintza zeharkatzea lortzen duen arte. Sarrerako elikadura presioa denborarekin areagotzen da, geroz eta gehiago egonkortuz. Mintzak iragazi duen gasaren presioari iragazkortasun presioa deritzo. Iragazkortasun presioaren balioak dira neurtu nahi direnak esperimentuan eta mintzaren materialaren eta gas motaren elikadura presioaren arabera emaitza desberdinak lortuko dira.

1.6 Hardwarea

Atal honetan, proiektuan instalatu eta erabili diren hardware gailuen funtzio eta xehetasun orokorrak jorratuko dira.

1.6.1 PR4000

Erabiltzen den DAQ gailua, PR4000 izenekoa, gas fluxu kontrolagailu eta presio sentso-reentzat diseinatuta dagoen gailua da; oso malgua da eta zuzenean edo PC bidez konfiguratu daiteke.

Lehen azaldu den moduan, gailu honen papera datuen eskuraketan ezinbestekoa da. Sentsoreak, magnitude fisikoaren baliokidea den seinale analogikoa bidaltzen dio PR4000 DAQ gailuari. Bestalde, digitalizazio prozesua oso garrantzitsua da datuak ordenagailuarentzat interpretagarriak izateko [2].

Bukatzeko, ordenagailua eta DAQ gailuak partekatzen duten busaren bitartez, seinale digitala ordenagailuari bidaltzen zaio neurtutako presioaren balioarekin. PR4000 eta ordenagailuak partekatzen duten bus hori serie portu bat da eta ordenagailuak DAQ gailuari komandoak bidaltzeko erabiltzen da ere.

Serie portua datuen komunikaziorako interfaze digitala da, ordenagailuek eta periferikoek maiz erabiltzen dutena, informazioa bitaz bit transmitituz, hau da, bit bakarra aldi berean bidaliz. Dagoeneko nahiko zaharkituta dagoen konexio mota da eta gaur egun beste konexio arinagoak erabiltzen dira.

1.6.2 Raspberry Pi 4

DAQ gailuarekin komunikatuko den ordenagailu berria, *Raspberry pi 4* izeneko gailua da. Ordenagailu honekin, nahikoa dugu gasen presioen balioak lortzeko. Haren prozesadorea, ARM Cortex-172 da eta lau nukleo ditu 1,5 GHz-ko abiaduran; GPU-ari dagokionez, VideoCore VI erabiltzen da (OpenGL ES 3.x-rako euskarria du); instalaturik dagoen RAM memoria, 4 GB LPDDR4 SDRAM da; konektibitatearen alderdian Bluetooth 5.0, Wi-Fi 802.11ac Gigabit Ethernet eusteko kapaza da, gaur egungo teknologietara egokituz; azkenik, aurki daitezkeen portuak hurrengoak dira: GPIO 40 pin, 2 x micro HDMI, 2 x USB 2.0, 2 x USB 3.0, CSI (Raspberry Pi kamara), DSI (ukimen pantaila), Micro SD, jack audio konektorea eta USB-C (elikadura).

Proiektuan erabili diren portuak Ethernet sarrera (sare lokalera konektatzeko), USB 3.0 sarrera (DAQ-aren serie portuaren irteera USB egokitzailerik bat erabiliz Raspberry-ra konektatzeko), USB-C sarrera (elikadurarako), micro HDMI portua (monitore batean sistema eragilea ikuskatzeko) eta Micro SD sarrera (sistema eragilea duen SD txartela sartzeko) izan dira.

1.7 Softwarea

Raspberry-an burutu dira eginbeharreko modulu guztiak, baina programazioa erosoagoa izateko, beste IDE (Integrated Development Environment) batzuk erabili dira modu ero-

soan lan egiteko; horietan burututakoak Raspberry-ra pasa dira bukaeran.

Komunikazio modulu osoa C lengoaian burutu da *Visual Studio Code* IDE-a erabiliz, tresna honek eskaintzen duen erraztasunarengatik; hainbat luzapen, testu editore erraza, ssh urruneko konexioa eta beste aukera asko eskaintzen ditu. Web modulurako ere editore berdina erabili da, hainbat lengoai eusteko kapaza delako.

Bestalde, Raspberry Pi 4-an *socket* izeneko tresnarekin serie portu birtualak simulatu dira, urrunetik programa nagusiaren funtzionamendu egokia bermatzeko.

Software hauek nire erosotasunarentzat erabili dira, baina ez dute azken erabiltzailean eraginik. Raspberry Pi 4 ordenagailuari dagokionez, Debian sistema eragilea instalatu zaio.

1.8 Proiektuaren helburuak

Atal honetan, proiektua burutzeko lortu behar diren helburu guztiak hartuko dira kontuan.

Lehenik eta behin, zein da kimika fakultatean burutzen den esperimentua? Laborategian gas ezberdinen portaerak gainazal desberdinetan neurtzea da helburua. Gas-ganberaren barruan hainbat zuntz eta balbula daude gasen intentsitatea kontrolatzeko eta gasak ganberaren barruan era seguruan maneiatzeko. Esperimenturako, material zehatz bateko mintz bat ipintzen da eta barruko egoera PR4000 DAQ gailuaren bitartez kontrolatzen da hainbat parametro kontuan hartuz.

Beraz, esperimentuaren saiakera zehatz batean erabiltzen den gasa eta mintza kontuan hartuz, neurtutako gasaren presioen balioak noiz bilakatzen diren egonkor da jakin nahi dena. Hori lortzen denean, esperimentua bukatutzat ematen da ordenagailuan emaitzak irudikatuz.

Proiektu honen helbururik nabarmenena, kimika fakultateko gasen esperimentuen emaitzak ikuskatzeko ekipamendua eta komunikazio modulua guztiz berritzea da, softwarea guztiz berrinplementatuz eta eguneratutako gailuak instalatuz. Programatuko den modulu berriak serie portuaren konfigurazio parametroak ezarri beharko ditu, serie portuan idazketak (eskaera komandoak) eta irakurketak (datuen irakurketa) burutzea kapaz izateko.

Lortutako esperimentuen emaitzak, nginx web zerbitzaria erabiliz sortuko den web orri batean argitaratuko dira, modu txukun eta ikusgarri batean. Era berean, web orriaren bitartez esperimentu berriak hasieratu edo gelditzeko aukera dago. Web gunean, esperimen-

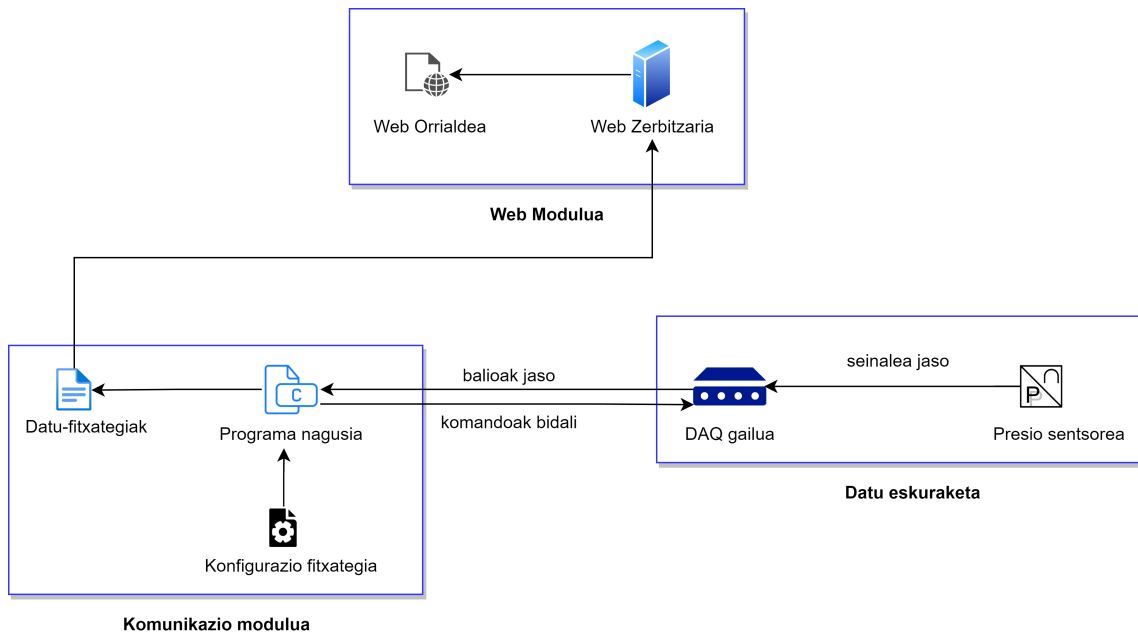
tuaren grafikoak, sarrera-datuak, presioen balioak eta burututako esperimentu guztien karpetak daude eskuragarri besteak beste. Software irekiko nginx web zerbitzaria baliagarria izan da web gunea martxan jartzeko hainbat teknologia erabiliz, hala nola: html, css, php, xslt eta javascript.

Hardwareari dagokionez, instalatuko den ordenagailu berria Raspberry Pi 4 bat izango da. Ordenagailu berri honek onura asko dakartza, hala nola, okupatzen duen espazio fisikoa txikia, ordenagailu zaharrarekin konparatuz eskaintzen duen azkartasuna, prezioa, errendimendua eta abar; gainera, ekipamendu berri honek gaur egungo aplikazioen betebeharrak moldatzen da. Bestalde edozein hausketa edo hondaketen aurrean ez dago arazorik beste Raspberry bat erosteko, oso merkea delako eta instalazioa oso erraza delako.

Bestalde, ez da monitorerik instalatuko, soilik Raspberry-a, hau da, ikerlariak eskuragarri duten beste monitoreak erabiliz, haien eramangarri pertsonalak erabiliz edo mugikorrek erabiliz, web orria atzitu dezakete web nabigatzailea erabiliz. Web orrira sartzeko modua oso erraza da: erabiltzaileak Raspberry-ak duen IP helbidea nabigatzailean ipiniko du web orria atzitzeko. Fakultateko sarearen kanpotik datozen konexioak ez dira baimentzen web modulua ez delako internetera ateratzen. Beste modu batean esanda, fakultatearen sarean konektatuta dagoen edozein ikerlariak web gunea atzitzeko ahalmena izango du, Raspberry-aren sare lokal berdinean kokatuta egongo dagoelako.

Honek ere beste aukera bat irekitzen du, hau da, etxetik unibertsitateko VPN-ra konektatuz, Raspberry-aren sare berdinean sartzeko aukera ahalbidetu daiteke web gunea atzitzeko baiena eskainiz. Honi esker esperimentuak urrutetik hasieratu, gelditu edo datuak ikusteko aukera egongo litzateke. VPN-aren aukera ez da eskuragarri egongo proiektua bukatzerakoan baina etorkizunean gehitu litekeen funtzionaltasuna da.

Aipatutako hiru ataza edo helburu nagusi hauek (komunikazio modulua, web modulua eta hardware berria), hardware eta softwarearen eguneraketa osoa suposatzen dute, ikerlari eta ikasleentzat aldaketa onuragarriak ekarriz. 1.2 eskeman, lortu nahi diren helburuen laburpena agertzen da, modulu eta gailu ezberdinen loturak azalduz.



1.2 Irudia: Proiektuaren diagrama

Laburbilduz, web moduluen barruan, web zerbitzaria muntatu behar da, esperimentuak kontrolatzeko erabiliko den web orria eusteko. Gainera, web gunearen bitartez esperimentua kontrolatuko da modu intuitibo batean. Web orriak diseinu simple bat edukiko du, beharrezkoak diren argibide eta emaitzak erakutsiz.

Bestalde, komunikazio moduluak hiru eginkizun izango ditu: serie portua erabiltzeko konfigurazioa zehaztu, serie portua erabiliz DAQ gailuarekin komunikatzeko komandoak idatzi eta DAQ gailuak bidalitako emaitzak jaso, grafikoak eta lortutako datuak webgunean irudikatuz. Esperimentua bukatzerakoan, hainbat fitxategi sortzen dira lortu diren datuak, emaitzak eta bestelako informazioa gordez.

Bukatzeko, datu eskuraketaren moduluari dagokionez, jada instalaturik dauden gailuak erabiltzen dira eta horiek ez dira ukituko, proiektuaren helburua haiekin komunikatzea delako soilik.

2. KAPITULUA

Aurretik implementaturik zegoen modulua

Aurreko atalean aipatu den bezala, kimika fakultatean jada inplementaturik dagoen modulua ordezkatu da, komunikazio modulurik ez balego bezala. Hau da, aurreko inplementazio hau ez da erabili proiektu honetan burutu diren moduluetan, baina esperimenduaren funtzionamendua ezagutzeko eta bertan gertatzen diren erreakzio eta probak ulertzeko balizkoa izan da. Atal honetan esperimenduaren xehetasunak behatuko dira aurreko inplementazioan oinarrituz, proiektuan burutu denaren ulermenari begira. Programa nagusia Labview plataforman inplementatu zenez, lehenik LabView-ren azalpen txiki bat burutuko da.

LabView tresna, hainbat sistema diseinatzeko erabiltzen den aplikazioa da. Honek programazio lengoai grafikoa erabiltzen du software eta hardwarearen arteko erlazioak ikusgarriagoak bihurtzeko. LabView-ko edozein proiektuk Front Panel eta Block Diagram izeneko inguruneak erabiltzen dituzte. Block diagram-ak, iturri kode grafikoa dauka eta Front Panel-ak berriz, erabiltzailearen interfazeari egiten dio erreferentzia, gailuaren kontrolak eta mandoak simulatuz. Block diagram-an erantsitako terminal (sarrera edo irteera portuak), aldagai edo objektuak Front Panel-ean eguneratuko dira ere, eta alderantziz (adibidez, kontrol terminal bat ezartzen bada bloke diagraman sarrera portu moduan, automatikoki sortuko da Front Panelean terminal horri dagokion gelaxka, parametroren bat sartzeko).

Programa honek, kimika fakultateko DAQ gailu batetik jasotako datuak prozesatu eta irudikatzen ditu; datu hauek tenperatura eta presioa dira. Magnitude hauek prozesatu ahal izateko DAQ prozesuan zehaztutako pausuak burutu behar dira. Bestalde ordenagailuak

software kontroladore eta aplikazio software bat erabiliko du, Labview tresnaren bidez esperimentua irudikatu eta exekutatu ahal izateko. Software kontrolagailuak, aplikazio softwareari beste gailu batekin elkarlanean ibiltzeko gaitasuna ematen dio, gure kasuan DAQ gailu bat, eta gailu horrekin burututako komunikazioa sinplifikatzen du maila baxuko hardware komandoak erabiliz. Aplikazio softwareak berriz, ordenagailua eta erabiltzailearen arteko harremana errazten du, hainbat neurketa aztertu, lortu edo irudikatzeko.

2.1 Programaren xehetasunak

2.1.1 Erabilitako DAQ gailuak

Jasotako kodeak gasen presioa eta temperatura neurtzen ditu, baina proiektu honetarako erabiliko den esperimentuetarako makina beste bat da eta honek soilik presioa hartzen du kontuan. Hala ere, kode honen zati guztiak aztertuko dira esperimentuaren xehetasun guztiak ongi barneratzeko.

Lehenik Block diagram-an (kode grafikoa sortzeko atala) burutu den kodea aztertuko da. Kodearen helburua, gasen tenperaturak (C°) eta presioak (PSI-Pounds per Square Inch) ikusaraztea da. Horretarako sentsoare eta DAQ gailu desberdinak erabiliko dira temperatura eta presioa neurtzeko (DAQ gailu ezberdinen xehetasunak eta konfigurazioa ikusteko, DAQ assistant objektua erabiltzen da).

2.1.1.1 Temperaturarako DAQ gailua

Temperatura neurtu ahal izateko lehenik sentsoare batetik tenperaturaren seinale analogikoa lortu behar da. Kasu honetan termobikote sentsoare bat erabiltzen da [3]. Sentsoare honek bi metal dispareen arteko ukimenaren ondorioz potentzial diferentzia bat eragiten du (milivoltaren parekoa) seinale elektriko bat sortuz, tenperaturaren baliokidea dena. Tenperaturaren neurketarako erabiltzen diren bi metalak constantana (kobrea eta nikela biltzen duen aleazioa) eta burdina dira. 2.1 irudian eta 2.2 irudian, LabView inguruneak termobikote sentsoari buruz eskaintzen duen informazioa ikus daiteke.

Thermocouple Type	Positive Conductor	Negative Conductor
J	Iron	Constantan

2.1 Irudia: Termobikote mota

Temperature Range (°C) for Polynomial Coefficients or for Table Conversion	
-210 to 1200	
Temperature Range (°C) for Inverse Polynomial Coefficients	
-210 to 1200	

2.2 Irudia: Termobikote motaren xehetasunak

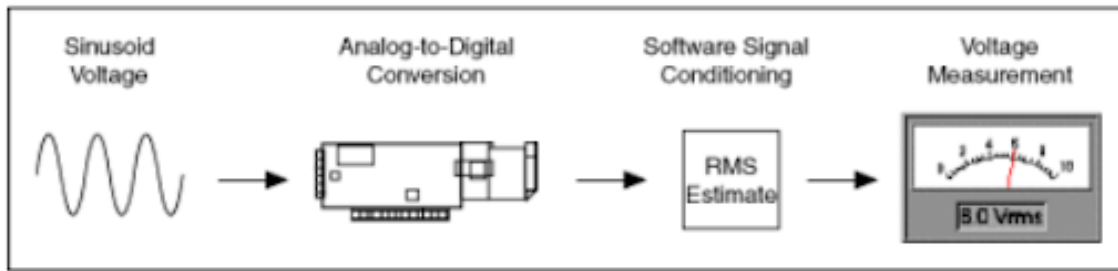
Temperaturarako erabiltzen den DAQ gailu honek bi kanal ditu, bata elikadura tenperatura neurtzeko eta bestea tenperaturaren iragazkortasuna neurtzeko. DAQ assistant-aren interfazetik ere ikus ditzakegu errealitatean erabiltzen diren bi kanal fisikoak zeintzuk diren eta kanal fisiko horietatik lortuko dira sentsoretik jasotako seinaleak.

Bestalde, laginak programa hasten den unean eta portuak irekitzen direnean kanal fisikoe-tatik uneoro jasoko dira.

Azken konfigurazioa parametroa seinalearen egokitzapenarekin du zerikusia, zehazki iturriko CJC-arekin (Cold-Junction Compensation). Termobikote batek tenperatura erreferentzia bat behar du, nahi ez diren termobikote parasitikoek eragina konpentsatzeko; horri deritza Cold-Junction Compensation deituriko egokitzapena. Termobikotea gailu batzuei konektatuta dagoenez, eta instrumentu edo gailu horiek beste materialez osatuta daudenez, termobikoteen kableen lotuneetan tentsioa sortzen da (lotune hotzak deritze hauri). Tentsio hau orekatzeko eta irteeran termobikote beraren tentsioa lortzeko, CJC erabiltzen da.

2.1.1.2 Presiorako DAQ gailua

Kasu honetan DAQ assistant baten bitartez, aurrekoaren era berean, konfigurazioak eta xehetasunak editatu daitezke. Presioa neurtzeko ez da aurreko metodo berdina jarraitzen, kasu honetan, DC tentsioa neurtzen da presioa neurtzeko; tentsio mota hau denboran zehar aldaketa motelak jasaten duten fenomenoak neurtzeko erabiltzen da. DC tentsioa neurtzerakoan, seinalearen zabalera denboraren une konkretu batean zein den jakin behar da ahal den era zehatzenean (DC korrontean, intentsitatea eta tentsioa beti berdina dira). AC tentsioa ere neur daiteke baina ez da egokia presioa bezalako magnitudeak neurtzeko, tentsio mota hau gehikuntza, gainbehera eta polaritate aldaketa asko jasaten duten magnitudeentzat delako egokia [4]. Tentsioa neurtzeko jarraitu beharreko pausuak 2.3 irudian agertzen direnak dira [5]:



2.3 Irudia: Tentsioa neurtzeko jarraitu beharreko prozesua

Presiorako erabiltzen den DAQ gailu honek ere bi kanal ditu eskuragarri, bata presio elikadura neurtzeko eta bestea iragazkortasun presioa neurtzeko. Kanal fisiko horietatik lortuko dira sentsoretik jasotako seinaleak. Kasu honetan ere, laginak programa exekutatzen hasten den unetik uneoro hartuko dira.

Azkenik, tenperaturaren kasuan gertatzen den parean, behin DC korrontea duen seinalea neurtu denean, nahi ez diren beste uhinen soinuak oinarrizko seinaleari akoplatuta egotea da arruntena. Ondorioz, soinu horiek ekiditeko seinalearen konpresioa burutzen ohi da. Soinu asko duten seinaleetan neurketak zehatzagoak izateko burutzen den konpresio mota arruntena, *Averaging* deituriko prozesua da. Averaging teknikan presioaren seinale sinusoidala denbora periodo txiki batean hartuko da, eta bataz besteko balioak erabiliz nahi ez diren soinuak ezabatuko dira.

2.1.2 Funtzionamendu orokorra

Orain programa honen implementazio-kodeari buruzko azalpena burutuko da esperimintutik espero diren datuak eta eragiketak argitzeko.

Hasiera batean, kontrol terminalak sortzen dira. Kontrol terminalen funtzioa, Front Panel-ean jartzen diren balioak Block Diagram-an sarrera bezala ikusaraztea da, hau da, sarrera edo irteera portuak dira. Adibidez, Front Panel-ean denbora-tarte bat sarrera moduan pasatzerakoan, exekuzioa hasterakoan gelaxka horretan zehaztu den balioa kontrol terminalaren bidez Block Diagram-an sortutako programan erabiliko da.

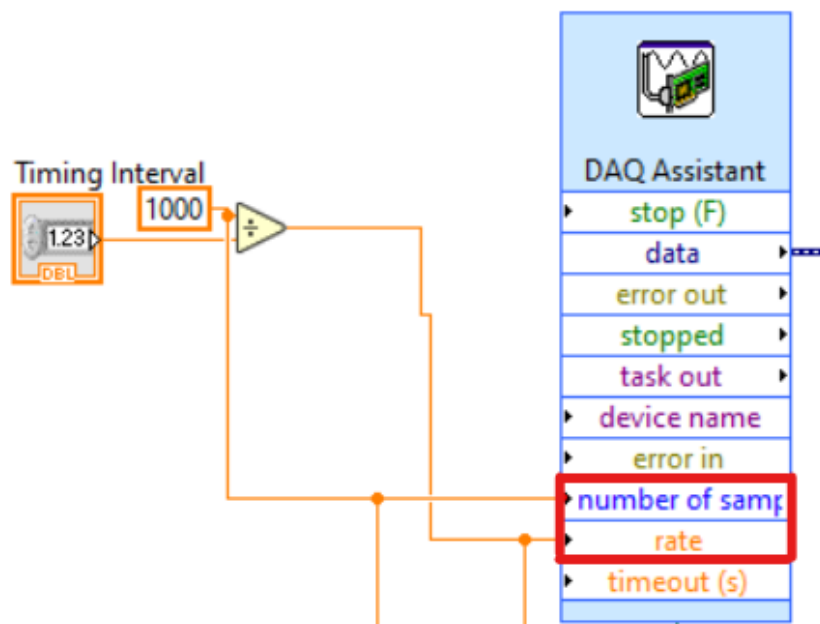
Xehetasun orokor hauek behin ulertuta, jada implementaturik zegoen kodea azaltzeko ordua da. Lehenik, irteeran lortuko den fitxategiaren informazioa eta haren egitura zehazten da (parametroak lerrotan egituratuz), beste era batean esanda, zeintzuk diren sarreran sartu diren parametro ezberdinak eta zer errepresentatzen duten. 2.4 irudian, irteera-fitxategian lortzen dena ikus daiteke.

```
Membrane
Area (cm2) 13
Support Porosity 1
Membrane Thickness (um) 150
Gas Constant [mL PSI/(mol K)] 1206
Feed Volume (cc) 321
Permeate Volume (cc) 17
```

2.4 Irudia: Sarrera parametroen egituraketa irteera fitxategian

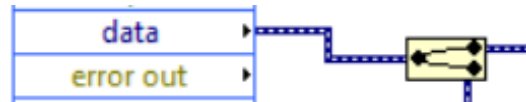
Jarraian neurketen datuak lortzeko garaia da. Horretarako lehen azaldutako bi DAQ gailuak erabiliko dira. Tenperatura eta presioak lortzeko kode-prozesua, begizta baten barruan dago eta begiztatik ateratzeko baldintzak stop botoia sakatzea edo esperimentua bukatzea dira soilik. Laginak hartzen hasi baino lehen, laginak jasotzeko laginketa-ratioa eta lagin kopurua zehaztu beharko dira.

Denbora-tartea sarrera parametro bezala jasoko da (Front Panel-ean zehaztuko da balioa) kontrol terminal bat erabiliz eta laginketa-ratioa jakiteko balio du. Laginketa-ratioa kalkulatzeko, denbora-tartea eta lagin kopuruaren arteko zatiketa burutu behar da, lagin kopurua aurretik zehaztu den konstante bat da izanik. Datu hauek DAQ gailuei sarrera bezala ematen zaie (ikus 2.5 irudia).



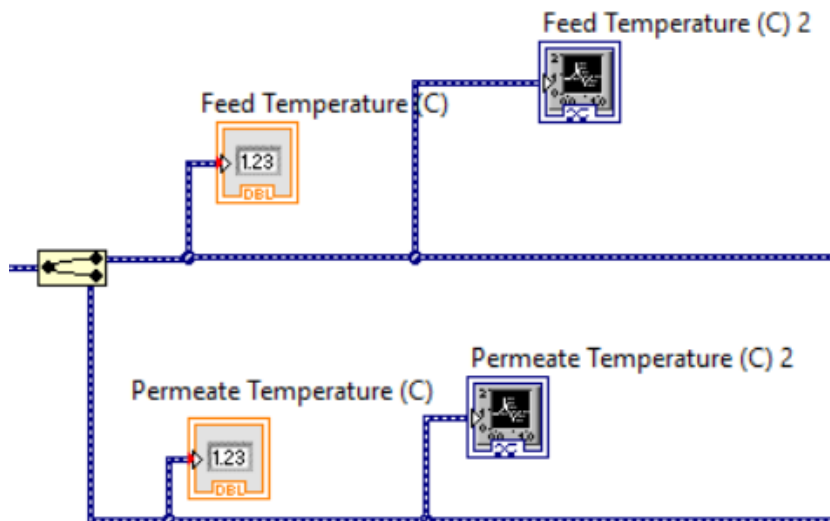
2.5 Irudia: DAQ gailuaren laginketa-ratioa eta lagin kopurua zehaztu

Temperaturari dagokion DAQ gailua hartuko dugu erreferentzia bezala orain. Ikus daiteke (2.6 irudian) *data* portuak irteera bakarra duela; baina kasu honetan bi kanal behar dira bi tenperatura ezberdinak lortzeko. Beraz seinale bakarrean ateratako tenperaturen balioak bereizi nahi dira, tenperatura mota bakoitzari dagokion seinalea isolatuz. Ondorioz, *data* irteera bi kanaletan banatzeko, *Signal Split* izeneko objektua erabiltzen da (2.6 irudia).



2.6 Irudia: Irteera bitan banatu *Signal Split* objektuaren bidez

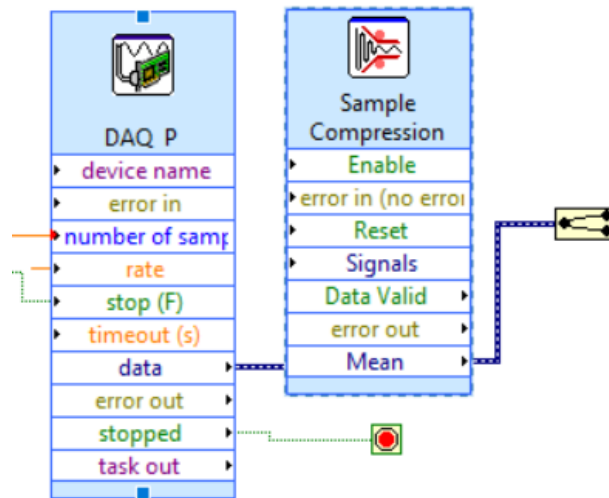
Behin irteera-seinalea bitan banatu denean, elikadura tenperatura eta iragazkortasun tenperaturaren balioak lortuko dira. Front Panel-ean emaitzak adierazteko, erakusle terminalak erabiliko dira (2.7 irudian laranja daudenak). Erakusle terminalak Front Panel-ean emaitzak bistartzeko erabiltzen dira eta emaitzak aldatzen diren momentuan Front Panel-ean eguneratzen dira. Bestalde tenperaturen grafikoak Front Panel-ean irudikatzeko *wavefront* izeneko beste objektu bat erabili da (ikus 2.7 irudia).



2.7 Irudia: Tenperatura ezberdinen balioak eta grafikoak Front Panel-ean irudikatu

Presioaren aldean, tenperaturaren prozesu bera burutzen da datuak lortzeko. Kasu honetan, lehen aipatutako Averaging teknika aplikatzeko, *Signal Conditioning* motako objektu bat erabili da (*mean* aukera hautatu da Averaging aplikatzeko). Behin seinaleari Averaging-a aplikatuta, lortzen den irteera bakarra denez, lehen egin den prozesu bera

burutu behar da, hau da, irteera seinale bitan banatu beharko da nahi ditugun presio ezberdinen seinaleak lortzeko (ikus 2.8 irudia).



2.8 Irudia: Averaging aplikatu irteera seinaleari eta seinalea bitan banatu

Azkenik, tenperaturarekin burutu den moduan 2.7 irudian, elikadura presioa eta presioaren iragazkortasunei dagozkien balioak eta grafikoak lortuko dira .

Nahi diren datuak jada lortu dira eta Front Panel-ean balioak eta grafikoak irudikatu dira; hortaz hurrengo pausua, emaitzak fitxategi batean gordetzea izango da. Irteera fitxategiaren egitura, lehen zehaztu da, beraz fitxategi hori bete beharko da soilik. Horretarako bektore bat sortzen da, tenperatura eta presioen balio ezberdinekin eta exekuzioaren denbora totalarekin. Exekuzio-denbora ere Front Panel-ean ikusarazten da programaren exekuzioan zehar erakusle terminal baten bitartez. Bukatzeko, sortutako bektore hori irteera fitxategian egokituko da lortutako emaitza guztiak irudikatuz.

3. KAPITULUA

Komunikazio modulua

Atal honetan, helburuetan azaldutako komunikazio moduluaren xehetasunak eta inplementazioaren funtzionamendua azalduko dira.

3.1 Komandoak

Modulu honen helburu nagusia DAQ gailuari komandoak bidaltzea eta presioen datuak jasotzea da. Serie portutik trukutzen diren komandoak zeintzuk diren jakiteko eta zer informazio bidaltzen den ikusteko, sniffer bat instalatu zen laborategiko ordenagailuan. Sniffer bat serie portua zelatatuko duen programa bat da eta trafiko guztia ikuskatzeko erabiltzen da. Tresna honen bitartez, DAQ gailua eta ordenagailuaren arteko mezu jariora ikus daiteke, datuak, komandoak eta hainbat mezu bistaratu. Erabili den doako snifferra *Portmon* du izena eta gailuen arteko mezu trukea ikuskatzeko dago pentsatuta. Hurrengo komandoak bidaltzen direla behatu da:

- **Open:** Komando honen bitartez, DAQ eta ordenagailuaren arteko komunikazioa has daitekeela zehazten da. Komando honen ondoren bi gailu hauen arteko handshake edo akordio bat burutuko da, serie portuaren konfigurazioa zehazteko.
- **AV1:** Parametro eta konfigurazio guztiak zehaztu ondoren, ordenagailuak DAQ gailuari datuak eskatzen hasiko da. Komando honek zehazki elikadura presioaren balioa lortzeko bidaltzen da.

- **AV2:** Elikadura presioa lortu ondoren, iragazkortasun presioa lortuko da AV2 komandoaren bitartez. Honen ondoren komandoak berriro bidaliko dira AV1-ekin hasiz eta AV2-rekin jarraituz.
- **Close:** Automatikoki bidaltzen da esperimientua amaitzerakoan, bai zehaztutako denbora bete delako, bai sarreran zehaztutako baldintzak bete direlako.

Aipatu den moduan, datuak eskuratzeko bi komandoak orden zehatz batean bidaltzen dira, hau da, lehenik AV1 bidaltzen da elikadura presioaren balioa jakiteko eta jarraian AV2 komandoa bidaltzen da iragazkortasun presioaren balioaren berri jakiteko. Bi komando hauei dagozkien txandetan komando bat bidali ondoren, arazoren bat gertatu bada datuen eskuraketan edo 0 byte jaso badira, berriro bidaliko da komando hori erantzun onargarri bat lortu arte.

3.1.1 Kontrol-seinaleak

Azaldutako komandoak bidaltzeko, irekita dagoen serie portuan komandoari dagozkion karaktereak idatzi behar dira.

Portuan idatzi baino lehen, serie portuaren egoera zein den jakin behar da. Burutu nahi den ekintzaren arabera kontrol-seinale jakin bat erabili behar da eta portuaren egoera lortu ondoren, kontrol-seinale hori aplikatu beharko da, aukeratutako ekintza burutu ahal izateko [6]. 3.1 irudian, kontrol-seinale batzuk ikus ditzakegu:

Constant	Description
TIOCM_LE	DSR (data set ready/line enable)
TIOCM_DTR	DTR (data terminal ready)
TIOCM_RTS	RTS (request to send)
TIOCM_ST	Secondary TXD (transmit)
TIOCM_SR	Secondary RXD (receive)
TIOCM_CTS	CTS (clear to send)
TIOCM_CAR	DCD (data carrier detect)
TIOCM_CD	Synonym for TIOCM_CAR
TIOCM_RNG	RNG (ring)
TIOCM_RI	Synonym for TIOCM_RNG
TIOCM_DSR	DSR (data set ready)

3.1 Irudia: Kontrol-seinaleen konstante posibleak

3.1.2 Komandoen bidalketa

Komandoa serie portuan idazteko interesatzen zaigun kontrol-seinalea RTS (Request To Send) deritzona da (ikus 3.1 irudia). Kontrol-seinale honen bitartez, ordenagailua daturen bat bidaltzeko prest dagoela ohartarazten zaio DAQ gailuari. Kontrol-seinale hau, lortu den portu seriearen egoerari gehitu behar zaio OR eragiketa aplikatuz, DAQ gailua datuak jasotzeko prest egoteko.

Honen ondoren komandoa serie portuan idatziko da (erroreren bat badago komando berdina bidaliko da berriro). Idazketa jada burutzerakoan, datua idazteko erabili den RTS seinalea ezabatu behar da, RTS kontrol-seinaleari dagozkion bitak desgaituz eta serie portuaren egoeraren bitak eguneratuz.

3.1.3 DAQ gailuaren erantzunen irakurketa

Komandoen idazketarako erabili den prozeduraren antzera, DAQ gailuak bidalitako balioak irakurtzeko beste kontrol-seinale bat erabiliko da. Kontrol-seinale hori DTR (Data Terminal Ready) izango da eta honen bitartez, ordenagailuak DAQ gailuari jakinaraziko dio datuak jasotzeko prest dagoela (ikus 3.1 irudia). Beraz, ordenagailuak serie portuaren egoera eguneratuko du DTR kontrol-seinalea gehituz eta DAQ gailuaren datuei prest itxaroten geldituko da. Behin DAQ gailuak bidalitako balioak jasotakoan, serie portuaren egoera eguneratuko da berriro, DTR kontrol-seinalea desgaituz (DAQ gailuak hurrengo datua prest duenean berriro gaitzen da DTR kontrol-seinalea).

3.2 Serie portuaren konfigurazioa

Serie portua ongi konfiguratzea, ezinbestekoa da bi gailuen arteko komunikazioa ahalbidetzeko. Konfigurazio fitxategia, komunikazio moduluaren programa nagusitik kanpo dagoen fitxategi bat da, erabiltzaileari konfigurazioa aldatzeko malgutasuna eskaintzeko. Kanpo konfigurazioa fitxategian konfigurazioaren parametroren bat aldatzen bada, programa nagusia konpilaziorik gabe exekuta daiteke.

Serie portuaren funtzionamendurako konfiguratu behar diren parametririk garrantzitsuenak hurrengoak dira: portuaren kokapena (normalean /dev direktorioan), baud ratea (segundoko seinale unitate kopurua; baud batek hainbat bit izan ditzake), zenbat datu-bit transmititzen diren byte bakoitzeko, stop bitaren erabilpena, paritate bitaren erabilpena

eta fluxu kontrolaren erabilpena. Parametro hauez gain badaude beste hainbat parametro gehigarri, aurrerago aipatuko direnak.

Konfigurazio fitxategian, parametroak modu ulergarrian irakurri eta aldatu ahal izateko, 0 eta 1 zenbakiak erabiliko dira aukerak gaitu eta desgaitzeko (baud rate-arentzat edo byte bakoitzaren bit kopurua zehazteko, nahi den zenbaki konkretua zehaztuko da, ez 0 edo 1). Konfigurazio aukerak programa nagusiaren bitartez irakurri eta prozesatuko dira, konfigurazio fitxategia lerroz-lerro irakurriz. Lerro bakoitza interpretatu ondoren, konfigurazio bitak eguneratuko dira hautatutako aukerak gehituz edo kenduz. 3.2 irudian konfigurazio fitxategiaren zati bat ikus daiteke adibide moduan:

```
#Serie-portuaren kokapena
port=/dev/ttyS1
#paritate bita: kendu (0), jarri (1)
parity_bit=0
#stop bit bakarra: erabili (1), bi stop bit erabili (2)
stop_bit=1
#size bit guztiak: hustu(1), ez hustu (0)
size=1
#zenbat datu-bit transmititzen diren byte bakoitzeko:
#5, 6, 7, 8
number of bits=8
```

3.2 Irudia: Konfigurazio fitxategiaren parametro batzuk

3.3 Inplementazioa

Moduluak egin behar duen betebeharrak nagusia, konfigurazioak irakurri eta aplikatu, komandoak bidali eta erantzunak jaso eta emaitzak modu irakurgarri batean gordetzea da. Bi modutan exekutatu daiteke programa nagusia, web gunean ipinitako informazioan oinarrituz.

Lehenengo exekuzio motan, programak denbora konkretu bat iraungo du eta esperimentuan zehar lortzen diren presioen balioak lortuko dira. Denbora hori betetzean, programa nagusiaren exekuzioa bukatuko da, esperimentuaren saiakera bukatuz. Esperimentuen zehar, datuen hainbat fitxategi sortzen dira, gero web guneak erabiliko dituen erabiltzaileek jaisteko eta grafikoak irudikatzeko. Programak neurtutako presio guztiak bi csv fitxategietan gordetzen dira web gunean grafikoki irudikatzeko. Gainera, batz besteko presioak eta esperimentuak iraun duen denbora ere gordeko da erabiltzaileak esperimentuaren emaitza orokorrak ikus ditzan.

Bigarren exekuzio motan, programak ez du iraupen konkreturik eta presioen balioak denbora zehatz batez egonkortzean bukatzen da programa. Esperimentuan presioaren balioak geroz eta egonkorragoak bilakatzen dira exekuzioan denbora pasa ahala, beraz balioak guztiz egonkorrak direnean, esperimentera bukatzen da (grafikoaren malda egonkortzean).

Balioak egonkortu diren edo ez ikusteko biderik errazena, laginen bigarren deribatuaren leunketa kalkulatzeko da. Hortaz, bigarren deribatuaren leunketa kalkulatzeko balioak zero badira egonkortzat hartuko dira. Balioak egonkorrak diren jakiteko, balio-muga bat jarri behar da. Balio-muga horrek, bigarren deribatuaren leunketa izan dezakeen balio maximoari egiten dio erreferentzia, hau da, muga hori baino txikiagoa diren zenbakiak zero moduan interpretatuko dira, egonkortzat hartuz.

Bigarren deribatua kalkulatzeko, lehenengo deribatua kalkulatu behar da hasteko. Lehen deribatua kalkulatzeko hurrengo formula jarraitu behar da, 'i' uneko lagina izanik:

$$\frac{presioa[i+1]-presioa[i]}{denbora[i+1]-denbora[i]}$$

Lehenengo deribatua lortu denean, balio horiek leuntzea komenigarria da emaitza zehatzagoak lortzeko. Lortutako balio bati leunketa aplikatzeko, aurreko k lagin kopurua hartuko da kontuan, eta uneko eta aurreko k laginen arteko batz bestekoa kalkulatu da. Bigarren deribatua kalkulatzeko, lehenengo deribatuaren leunketaren deribatua kalkulatu behar da. Bigarren deribatuan leundutako balioak erabiliko dira malda egonkortu den edo ez jakiteko. Leundutako bigarren deribatuaren leunketen balioak hasieran erabiltzaileak zehaztu duen denbora-tartean, zehaztutako balio-muga baino txikiagoak badira, esperimentera bukatuko da malda egonkortzat hartzen delako.

Elikadura presioaren eta iragazkortasun presioaren bigarren deribatuaren leunketaren grafikoak web gunean islatzen dira, ikerlariak maldaren egonkortzea argi ikus dezaten. Exekuzio mota honetan esperimentera iraungo duen denbora ezezaguna denez erabiltzaileen erosotasunari begira, korreo elektronikoa bidaliko zaie esperimentera bukatu dela jakinarazteko. Muga-balioa, k leunketa faktorea, malda egonkorra izateko bete behar den denbora-tartea eta erabiltzaileari dagokion posta elektronikoa, erabiltzaileak web gunean zehaztutako datuetatik lortzen dira.

Hurrengo azpiataletan, programa nagusirako burutu den inplementazioa azalduko da, zati ezberdinak bereiziz.

3.3.1 Datu-egiturak

Hiru datu-egitura sortu dira modulu honetan: lehenengoa konfigurazio fitxategiaren balioak gordetzeko, bigarrena lortutako presioen balioak gordetzeko eta azkena bigarren exekuzio motaren parametroak gordetzeko (muga-balioa, leunketa faktorea eta egonkortze denbora).

Konfigurazioaren datu-egiturari dagokionez, serie porturako kontuan hartuko diren konfigurazio-parametro guztiak definitu dira. Parametro mordo bat dago eskuragarri portuen konfigurazioa burutzeko, baina programa honetan hurrengoak definitu dira soilik:

- **port:** serie portuaren kokapena ordenagailuan.
- **parity bit:** paritate bitaren erabilpena.
- **stop bit:** stop bitaren erabilpena.
- **size:** size biten erabilpena.
- **number of bits:** zenbat datu-bit transmititzen diren byte bakoitzeko.
- **flow control:** hardware fluxu kontrolaren erabilpena.
- **read local:** irakurketaren aktibazioa eta ctrl lerroak kontuan hartuko diren edo ez zehaztu.
- **icanon:** canonical aukeraren erabilpena.
- **echo:** echo aukeraren erabilpena.
- **erasure:** erasure aukeraren erabilpena.
- **new line echo:** new line echo aukeraren erabilpena.
- **int quit susp:** INTR, QUIT eta SUSP karaktereen interpretazioa.
- **s/w flow kontola:** software fluxu kontrolaren erabilpena (Xon/Xoff).
- **special handling:** jasotako byteentzat maneiatze bereziaren erabilpena zehaztu.
- **special handling out chars bytes:** irteerako byteentzat interpretazio berezia zehaztu.
- **newline carriage:** lerro berriaren itzulera-lerroaren jarioa zehaztu.
- **vtime eta vmin:** bidalketa bateko datu guztiak modu eraginkorrean har daitezkeela mezuak gainjarri gabe ziurtatzeko.
- **custom baud rate1:** baud ratearen maskara-biten erabilpena zehaztu.
- **custom baud rate2:** aurredefinitutako baud-rateen kanpoko abiadura pertsonalizatuen erabilpena zehaztu.
- **baud rate:** sarrera/irteerako baud ratea zehaztu.

Aldagai guzti hauek, C lengoaiaren struct egitura batean definitu dira. Struct baten bidez,

datu mota ezberdin asko gorde daitezke izen bakar bat erabiliz; honela, edozein funtziotik nahikoa da struct egitura atzitzea nahi den aldagaiaren balioa lortzeko. Aurreko aldagaien balioak normalean 0 edo 1 izango dira (gaitu/desgaitu) eta horren arabera eguneratuko dira konfigurazio bitak.

Konfigurazioak aurrera eramateko, termios [7] izeneko egitura erabili da. Egitura honek terminal interfaze orokor bat deskribatzen du komunikazio portu asinkronoak kontrolatzeko. Termios egiturak (struct bat da ere) lau flag ditu konfigurazio bitak eguneratzeko: `c_iflag` (sarrera moduetarako), `c_oflag` (irteera moduetarako), `c_lflag` (modu lokaletarako), `c_cc[NCCS]` (karakterere berezietarako). Flag hauek erabiliz, nahi diren konfigurazio-bitak editatu daitezke 'or' edo 'and' eragiketak burutuz. Adibidez, paritate bita erabili nahi ez bada, sarreren aukeretarako `c_iflag` aldagaian 'and' funtzio bat burutuko da honela: `c_cflag &= PARENB` (PARENB paritate bitari dagozkion bit multzoa izanik).

Parametroak beharren arabera alda daitezke, baina esperimentuaren funtzionamendu egokirako hurrengo parametroak finkoak izan behar dira beti:

- **Portua:** /dev/ttyUSB0.
- **Baud ratea:** 9600.
- **Paritate bita:** Paritaterik ez.
- **Stop bita:** Stop bit bakarra.
- **s/w fluxu kontrola:** Xon/Xoff ahalbidetu.
- **hardware fluxu kontrola:** Desaktibatu.

Konfigurazioaren egituraz gain, beste struct egitura bat sortu da bektore dinamikoa definitzeko. Bektore edo array honek memoria erreserbatuko du dinamikoki elementu bat sartu behar denean. Memoria alokatzeko `realloc()` funtzioa erabili da eta funtzio honek aurretik alokatutako memoriaren (`malloc` erabiliz) erakusletik abiatuz, dagokion memoria-blokea birdimentsionatuko du.

Egitura honek hurrengo aldagaiak ditu definiturik: `array` izeneko bektorea (float motako presioen balioak edo laginak lortu diren denbora-uneak gordeko dira), `erabilitakoa` (bektorearen tamaina osotik zenbat gelaxka dauden okupaturik adierazten du), `size` (bektore osoaren tamaina adierazten du), `batura` (bektorean gorde diren balioen batura gordetzeko) eta `bb` (bektorean gorde diren balio guztien arteko bataz bestekoa gordetzeko). Iragazkortasun eta elikadura presioen balioak, datuak lortu diren denbora-uneak, deribatuak eta leunketak gordetzeko, azaldutako bektore dinamiko hau erabiltzen da.

Hirugarren egitura ere struct motakoa da eta hurrengo aldagaiak daude definiturik: k (leunketa faktorea, hau da, aurreko zenbat lagin kontuan hartu leunketa aplikatzeko), *denbora* (malda egonkor mantendu beharreko denbora) eta *muga* (muga-balioa baino txikiagoak diren bigarren deribatuen leunketen balioak egonkor bezala hartu).

3.3.2 Komunikazioa

Komunikazioari dagokionez, programak DAQ gailuak ulertzen dituen komandoak idatziko ditu serie portuan eta ondoren DAQ gailuak erantzundako datu errealak jasoko ditu, dagokion datu-egituretan balioak gordez.

Lehenik egitura eta datu guztiak hasieratuko dira, bai bektore dinamikoak, bai konfigurazio-egitura bai erabiliko diren beste aldagaiak. Jarraian termios egitura erabiliz definitutako "tty"aldagaia erazagutzen da, serie portuaren konfigurazioa maneiatzeko. Ondoren, konfigurazio fitxategia irakurri beharko da lerroz-lerro eta serie portuaren konfigurazio parametro guztien bitak eguneratuko dira fitxategi horretan jartzen duenaren arabera (0 = desgaitu, 1 = gaitu eta abar). Konfigurazio fitxategia ez bada ongi irekitzen, errore mezu bat aterako da pantailan.

Konfigurazio fitxategia ongi irakurri denean, serie portua ireki behar da *open* sistema deia erabiliz (struct konfigurazio egituran "port"aldagaian zehaztutakoa). Serie portua ongi irekitzen bada programak aurrera jarraituko du, bestela exekuzioa bertan behera utziko da.

Jarraian, serie portuaren defektuzko konfigurazioa lortuko da *tcgetattr* funtzioa erabiliz eta konfigurazioa tty aldagaian (serie portuaren informazioa duen termios motako aldagaia) gordeko da. Serie portuaren konfigurazio pertsonalizatua honen gainean gordeko da struct egituran gorde diren konfigurazio-balioetan oinarrituz. Horretarako struct guztia korrituko da intereseko konfigurazio-bit guztiak eguneratuz. Azkenik, *tcsetattr* funtzioa erabili da konfigurazioa gordetzeko.

Serie portua konfiguratu da jada eta orain DAQ gailua eta PC-aren arteko komunikazioa hasieratu daiteke. Aipatzekoa da exekuzio motaren arabera, programak aldagai eta funtzio desberdinak erabiltzen dituela. Jarraian AV1 eta AV2 komandoak bidaltzeko ordua denez, *while* begizta bat erabili da esperimentuaren datu guztiak lortzeko.

3.3.2.1 Lehen exekuzio mota

Lehenengo exekuzio mota aukeratzen bada, hasieran esperimentuak iraungo duen denbora jakin behar da. Horretarako, erabiltzaileak formularioan zehaztu duen denbora, sarrera parametro moduan lortuko da; erroreren bat gertatzen bada programa errore batekin exekuziotik aterako da. Denbora segundoetan neurtzeko *time_t* (long int motaren balioakidea) motako aldagaiak erabili dira. Begiztaren iraupena, esperimentuak iraungo duen denborearen berdina izango da eta ondorioz denbora hori agortzerakoan, exekuzioa begiztatik aterako da emaitzak gordez eta esperimentua bukatuz.

Begiztaren barruan, switch/case bat erabili da AV1 eta AV2-ren txandak antolatzeko, hau da, lehenengo elikadura presioa lortuko da AV1 komandoa erabiliz eta ondoren iragazkortasun presioa lortuko da AV2 komandoarekin. Txanda bakoitzari dagokion komandoa serie portuan idatziko da, RTS kontrol aldagaia erabiliz. Komando bera idazteko *write()* sistema deia erabili da eta idazketa burutu ondoren, RTS desgaituko da, hurrengo komandoa bidali behar den arte.

Idazketa burutu ondoren, DAQ gailuak bidalitako presioaren neurketaren balioa jaso behar da. Horretarako, DTR kontrol aldagaia gaituko da eta *read()* funtzioaren bitartez DAQ gailuak bidalitako byte guztiak irakurriko dira. Balioak jasotzerakoan, DTR kontrol-seinalea desgaituko da.

Lortutako datuak presioen bektoreetan gordeko dira eta lortu diren denbora-uneak beste bektore batzuetan gordeko dira era berean; modu horretan, presio bakoitza exekuzioarekiko noiz lortu den jakin daiteke. Iterazio bakoitzean serie portua itxi den egiaztatuko da, portua ixten badelako programaren exekuzioa moztu behar baita. Bestalde, DAQ gailuak bidaltzen dituen balioak gordetzeko erabiltzen den bufferra iterazio bakoitzean berritu behar da, aurreko iterazioko datuekin ez teilakatzeko.

3.3.2.2 Bigarren exekuzio mota

Bigarren exekuzio motari dagokionez, prozedura antzekoa jarraitzen da. Kasu honetan begiztak ez du denbora konkretu bat irauten, hau da, maldaren egonkortasuna lortzean bukatuko da programa. DAQ gailuarekin komunikatzeko bidaltzen diren komandoak idazteko eta DAQ gailuak bidalitako erantzunak irakurtzeko, lehenengo exekuzio motaren prozesu bera betetzen da.

Lehenengo exekuzio motarekin aldatzen den atala, ondorengoa da: DAQ gailuak pasatako presioen balioei deribatu eta leunketak aplikatzen zaizkie. Lehen azaldu den moduan,

presioei lehen deribatua aplikatuko zaie, lortutako balioak leunduko dira, jarraian bigarren deribatua kalkulatu da eta bigarren deribatutik lortutako balioak leunduko dira.

Leunketa faktorea, egonkortze denbora eta balio-muga, erabiltzaileak betetzen duen formulariotik lortzen dira eta 3.3.1 azpiatalean aipatutako struct egituran gordetzen dira.

Komunikazioaren ordena jarraituz, lehenik elikadura presioaren deribatuak eta leunketak kalkulatu dira eta ondoren iragazkortasun presioaren deribatuak eta leunketak (switch/-case egituraren bitartez). Presio mota bakoitzari dagokion txandan, azken leunketaren balioa, gordetako balio-muga baino txikiagoa den edo ez konprobatzen da. Balio-muga baino txikiagoa den bigarren deribatuaren leunketaren bat lortzen bada, tenporizadore bat hasieratzen da. Tenporizadore honek, balioak egonkorak izan behar duten denbora-tartea kontrolatzeko balio du. Beraz, leundutako balio horiek zehaztu den denbora-tartean zehar egonkorak badira, exekuzioa begiztatik ateratzen da eta emaitza fitxategiak sortzen dira datuak gordez .

Bukaeran esperimentera hasieratu duen erabiltzaileari korreo elektronikoko bat bidaltzen zaio esperimentera bukatu dela jakinaraziz.

3.3.3 Emaitzen errepresentazioa

Komunikazioan zehar lortu diren garrantzizko balioak nolabait gorde behar dira, ikertzaileek esperimentera lortu diren emaitzak era egokian ikusteko. Presio zehatzen balioak eta balio horiek zein denbora unean lortu diren, grafikoen bidez irudikatzen dira (presioak 'y' ardatzan eta denborak 'x' ardatzan).

Grafikoak sortzeko, esperimentera zehar programa nagusiak sortzen dituen csv fitxategiak hartzen dira kontuan; *feedPressure.csv* fitxategian elikadura presioaren balioak eta laginak lortu diren denbora-uneak gordeko dira eta *permeatePressure.csv* fitxategian iragazkortasun presioari dagokion balioak eta denborak.

Esperimentera hasi baino lehen bigarren exekuzio mota aukeratu bada, bi grafiko horiekin batera, beste bi grafiko irudikatzen dira ere, elikadura eta iragazkortasun presioen bigarren deribatuen leunketen balioak denboran zehar adieraziz. Grafiko horiek irudikatzeke beste csv fitxategi batzuk erabiltzen dira.

Azkenik, bi exekuzio moten kasuan formularioan bete diren eremu guztietan ipini dena, grafikoekin batera irudikatzen da, gas-ganberaren eta mintzaren hasierako egoera zein den jakiteko. Gainera formularioaren datu horiek beste csv fitxategi batean gordetzen dira,

erabiltzaileak automatikoki esperimentuaren bukaeran sortzen den karpetaren emaitzak eta datu fitxategiak bere saiakerari dagozkiola jakin dezan.

Aipatzekoa da aukeratzen den exekuzio motaren arabera oinarrizko fitxategiekin batera, beste emaitza fitxategi desberdinak lortzen direla, hau da, lehen exekuzio motan emaitza orokorrak lortzen dira (bataz besteko iragazkortasun eta elikadura presioak eta exekuzio-denbora) eta bigarren exekuzio motan bigarren deribatuen leunketak dituzten fitxategiak lortzen dira.

4. KAPITULUA

Web modulua

Web modulua datuak era txukun eta egokian bistartzeko zati garrantzitsu bat da eta eskuragarri dauden orrialdeen bitartez erabiltzaileen lana erraztea du helburu. Web orriak hurrengo funtzionalitateak betetzen ditu: esperimentu berriak martxan jarri, esperimentua gelditu, uneko datuak bistaratu (grafiko eta taulen bidez) eta emaitza-fitxategiak automatikoki sortutako karpeta batean gordetzea. Gainera hiru hizkuntza desberdinetan dago eskuragarri: euskaraz, gazteleraz eta ingelesez.

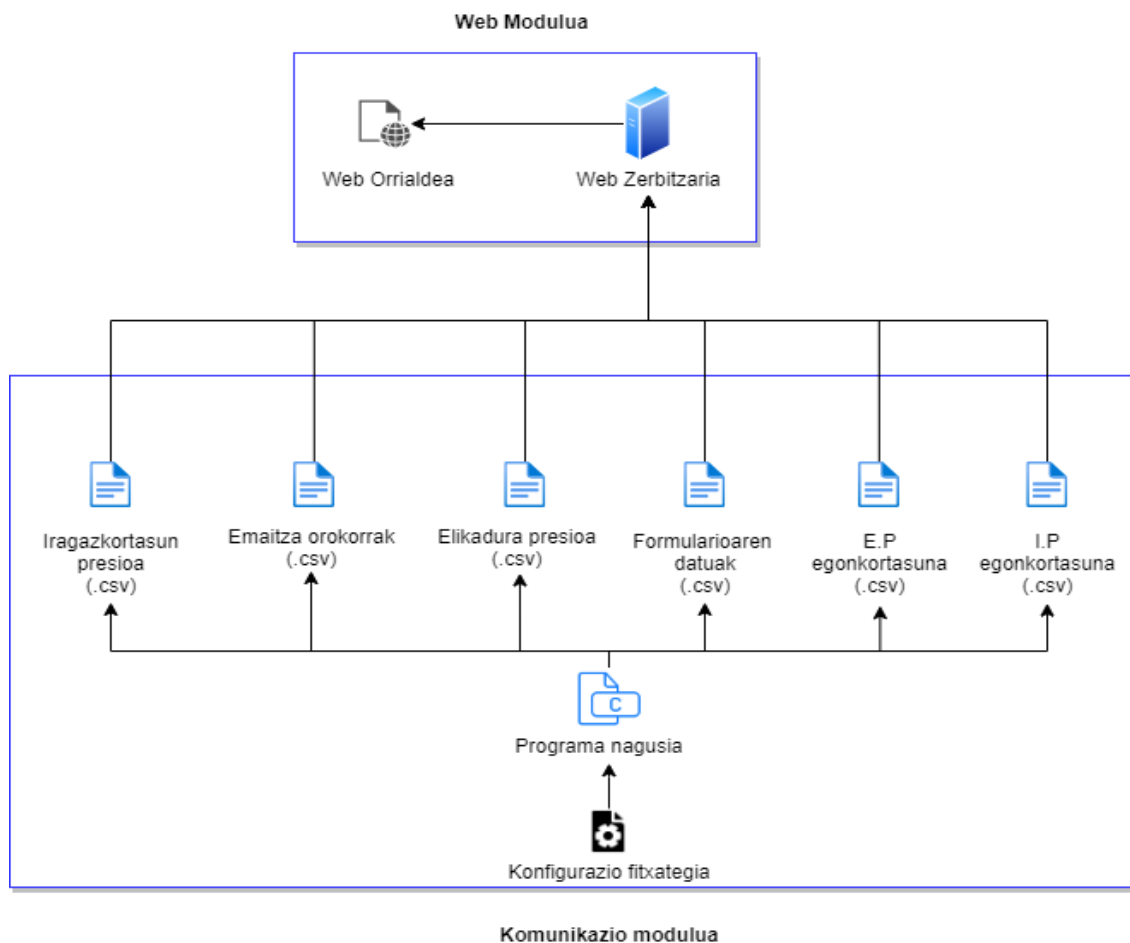
Funtzionamenduari dagokionez, web zerbitzaria martxan dagoen bitartean programa nagusitik lortutako datu-fitxategiak irakurtzen dira uneoro, presioen balioak web gunean eguneratuz eta grafikoak irudikatuz. Sei emaitza fitxategi desberdin posible bereiz daitezke (guztiak csv formatuan):

- **feedPressure.csv**: elikadura presioaren balioak denboraren funtzioan.
- **permeatePressure.csv**: iragazkortasun presioaren balioak denboraren funtzioan.
- **resultados.csv**: emaitza orokorrak.
- **formulario.csv**: erabiltzaileak esperimentua martxan jartzeko bete duen formularioaren informazioa.
- **estabilizacion_pendiente_feedP.csv**, **estabilizacion_pendiente_PermeateP.csv**: bigarren deribatuen leunketak denboran zehar.

Lehen bi fitxategiak, esperimentuan zehar presio ezberdinen balioak gordetzeko eta grafikoak irudikatzeko erabiltzen dira. Formularioaren fitxategiak bi funtzio ditu: programa nagusiak zein exekuzio mota aukeratu den jakitea eta erabiltzaileak burututako esperimentuaren saiakerari buruzko informazioa gordetzea. Emaitza orokorren fitxategian, lortu diren presioen balioen batez bestekoa eta esperimentuaren iraupen denbora gordetzen da. Bestalde, azkeneko bi fitxategiek grafikoaren malden egonkortasunari egiten diote erreferentzia. Datu hauek presioen balioak egonkorrak diren edo ez zehazteko erabiltzen dira, 3.3.2 azpiatalean azaldu den moduan.

Fitxategiek csv formatua dutenez, fitxategi ezberdinak jaisterakoan datuak formatu ulergarrian irudikatuko dira.

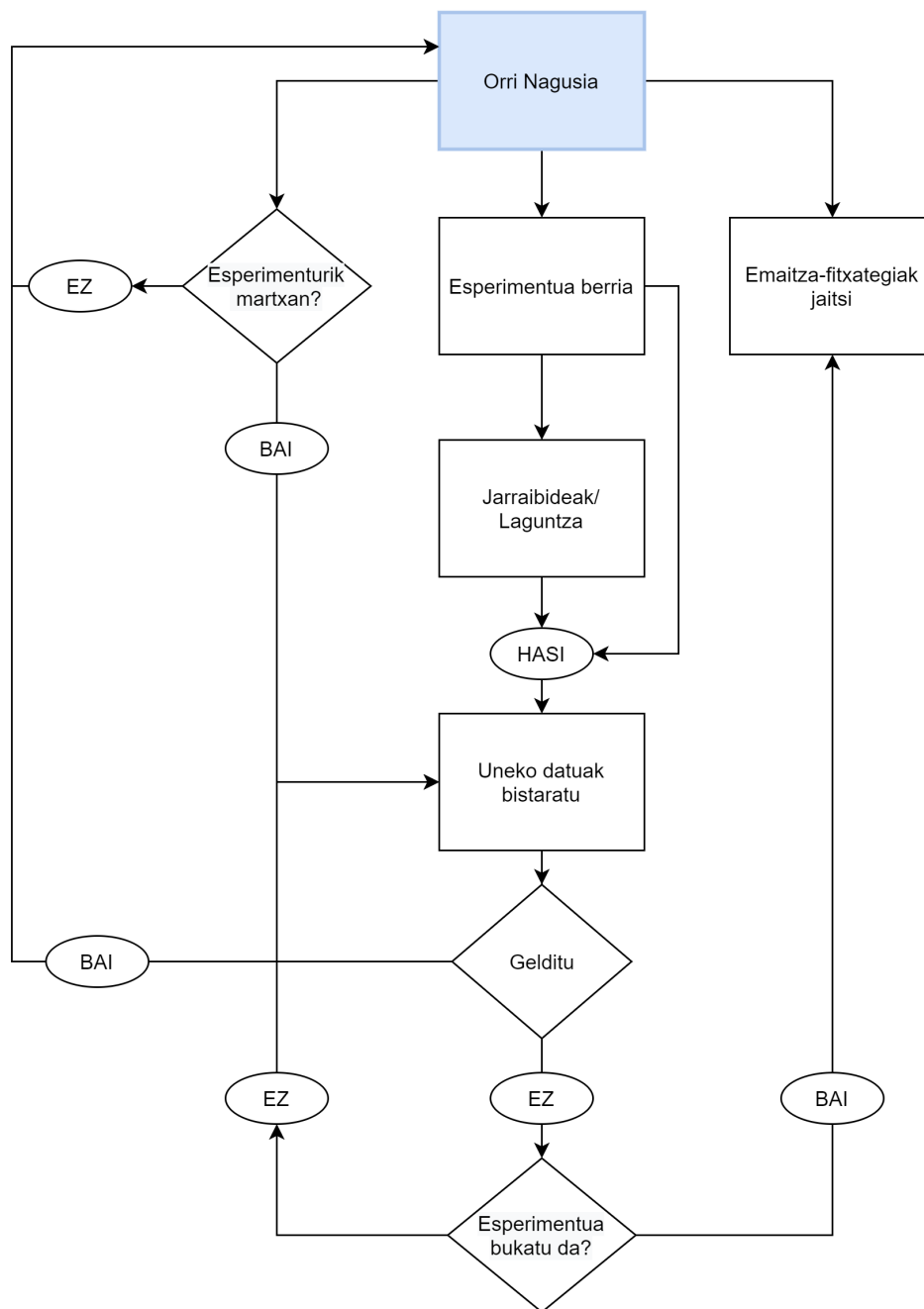
4.1 irudian, aurreko azalpeni dagokion diagrama ikus daiteke:



4.1 Irudia: Programa nagusitik web zerbitzariak eskuratutako datu-fitxategiak

4.1 Egitura

Hurrengo irudiak web gunetik nabigatzeko eskuragarri dauden orrialdeen arteko erlazioa erakusten du (ikus 4.2 irudia). Nabigazio diseinu hau erabiltzaileari erraztasunak eskaintzeko eta datuak era egokian irudikatzeko dago pentsatuta.



4.2 Irudia: Web gunearen egitura

4.1.1 Orri nagusia

Orri nagusiak menu nagusiaren funtzioa burutzen du eta atzigarri dauden beste orrialdeak esteka edo sakagailuen bidez daude eskuragarri. Hurrengoak dira atzitu daitezkeen orrialdeak:

- **Hizkuntza aldaketa:** Beste hizkuntza baten orrialde nagusira abiatu. Hiru hizkuntza daude eskuragarri: euskara, ingelesa edo gaztelania.
- **Esperimentu berri bat hasi:** Esperimentu berri bat hasteko esteka, sarrera-datuak betetzeko formularioa atzitzeko. Formulario honetan, ikertzailearen datuak, aukeratutako exekuzio mota eta esperimentuaren hasierako parametroak ipini behar dira.
- **Uneko esperimentuaren datuak bistaratu:** Uneko esperimentuaren hasierako formularioan ipinitako datuak eta presioen datuak irudikatzen dituzten grafikoak irudikatzen dira denbora errealean. Esperimenturik ez badago martxan mezu baten bidez jakinarazten zaio erabiltzaileari ezin daitekeela uneko daturik irudikatu.
- **Emaitza-fitxategiak jaitsi:** Burutu diren esperimentu guztiak karpetetan daude antolatuturik. Orrialde honetan burutu berri den esperimentuaren emaitza-fitxategiak jaisteko aukera dago. Karpeta bakoitzaren izena, ikerlariak formularioan jarri duen izena izango du, bere karpeta pertsonala zein den identifika dezan. Karpeta pertsonalen barruan, presioen datuak, formularioan sartutako datuak eta azken emaitzak dituzten fitxategiak daude.

4.1.2 Esperimentu berri bat hasi

Formularioan proiektuaren xehetasun orokorrak idatzi behar ditu erabiltzaileak. Hurrengoak dira erabiltzaileak derrigorrez bete behar dituen datuak: izen abizenak, azalera (cm^2), porositatearen euskarria, mintzaren lodiera (μm), gas konstantea ($[mL PSI/(mol K)]$), elikaduraren bolumena (cc) eta iragazkortasunaren bolumena (cc). Bestalde, 3. atalean aipatutako bi exekuzio moduei begira erabiltzaileak beste eremu batzuk ditu eskuragarri.

Malda egonkortu arte martxan egongo den exekuziorako, lau eremu gehigarri bete behar dira: posta elektronikoa, egonkortasunerako balio-muga, leunketa faktorea eta malda zenbat denboran zehar izan behar den egonkorra. Denboraren menpeko exekuzio motan berriez soilik eremu gehigarri bakarra bete behar da: esperimentuaren iraupen-denbora.

Sartutako datu guzti hauek csv fitxategi batean gordetzen dira, ikerlari edo ikasleak bere esperimentuari dagokion hasierako datu guztiak zeintzuk diren jakin dezan. Esperimentuarekin jarraitu nahi bada formularioaren datuak bidaltzeko sakagailuan klik egingo da.

Erabiltzaileak formularioan bete behar diren eremuei buruz dudarik badu, laguntza orri batera birbideratuko dion esteka sakatu dezake. Laguntza orrian formularioaren xehetasunak eta exekuzio motaren arabera zein eremu bete behar diren azaltzen da.

4.1.3 Jarraibideak

Gasekin esperimentua burutzeko eta gas-ganbera ongi doitzeko, ezinbestekoa da segurtasun neurri batzuk jarraitzea inor kaltetuta ez ateratzeko. Gas moten arabera gertatu daitezkeen arazoengatik, segurtasun jarraibide batzuk burutzea ezinbestekoa da. Jarraibide hauek, balbulak, makinak, mintzak eta beste hainbat gailu martxan jartzeko erabilgarriak dira. Jarraibideak ongi burutzen badira, esperimentua behar bezala burutuko da eta ez da inor fisikoki kalteturik aterako.

Erabiltzaileak hainbat alditan burutu badu esperimentu bera eta jarraibideak buruz bada-kizki, jarraibideak irakurri gabe esperimentua hasteko aukera dago. Esperimentua hasteko sakagailuan klik eginenez, komunikazio moduluan burututako programa nagusia exekutatu-ko da, datu-eskuratze prozesua hasieratuz eta uneko esperimentuaren datuak bistaratzeko eskuragarri dagoen orrialdea erakutsiz.

4.1.4 Uneko esperimentuaren datuak bistaratu

Orrialde nagusitik abiatuz edo jarraibideen orrialdetik abiatuz, uneko esperimentuaren hasierako datuak eta grafikoen datuak denbora errealean ikus daitezke. Exekuzio motaren arabera, ikusgai dauden grafiko kopurua ezberdina da.

Denboraren menpeko exekuzioa aukeratzen bada (lehen exekuzio mota), soilik bi grafi-ko egongo dira ikusgai, bata elikadura presioaren balioak erakutsiko dituena eta bestea iragazkortasun presioaren balioak erakutsiko dituena. Bestalde, presioaren balioak egon-kortu arte exekutatzea aukeratzen bada (bigarren exekuzio mota), lau grafiko erakutsiko dira, aipatutako aurreko biak eta leundutako elikadura presioaren eta iragazkortasun pre-sioaren bigarren deribatuen erakutsiko dutenak. Leundutako bigarren deribatuen bidez, malda egonkortzen ari den edo ez argi ikusten da.

Grafiko hauetan irudikatzen diren datuak, komunikazio moduluko programa nagusiak sortzen dituen csv fitxategietan oinarritzen dira, datu base baten moduko funtzionamendua burutuz. Formularioan betetako datu guztiak irudikatzen dituen taula, grafikoekin batera irudikatzen da, gas-ganberaren hasierako datu guztiak zein diren erabiltzaileari jakinarazteko.

Esperimenturik ez badago martxan, beste orrialde batera birbideratzen zaio erabiltzaileari, unean esperimenturik martxan ez dagoela adieraziz.

Bestalde, esperimentua arazorik gabe bukatzerakoan emaitza eta datu-fitxategi guztiak erabiltzailearen karpetan gordetzen dira automatikoki. Karpeta horiek arakatu daitezke bertan dauden fitxategiak jaitsi ahal izateko.

4.1.5 Emaitza-fitxategiak jaitsi

Esperimentuaren saiakera bukatzen denean, lortu diren emaitzak automatikoki sortutako karpetatik jaitsi daitezke.

Karpeta erraz identifikatzeko, formularioan zehaztu den izena, uneko data eta ordua zehazten da karpeta izen moduan. Hau da, Pablo Corral ikasleak esperimentu bat egin badu ekainak 3-an goizeko 9-tan, Pablo_Corral_2021-06-03_09:00 izeneko karpeta bat egongo da eskuragarri esperimentua bukatzerakoan.

Karpeta bakoitzaren barruan exekuzio motaren arabera sortu diren fitxategi desberdinak daude kokatuta. Oinarritako fitxategiekin batera, exekuzio motaren arabera fitxategi gehigarriak desberdinak sortzen dira (3.3.3 atalean azaldu direnak).

Modu honetan, ikasle edo ikerlari bakoitzak, norberari dagokion karpeta identifikatzeko gai izango da eta esperimentuaren fitxategi hauek jaisteko aukera izango du.

Bestalde, fitxategientzat csv formatua aukeratu da, Microsoft Excel, LibreOffice Calc eta Google Kalkulu-orriekin bateragarriak direlako. Aipatzekoa da, jaitsitako edozein csv fitxategi aplikazio hauen bitartez irekitzean, datuak tauletan modu txukun eta ordenatuan irudikatzen direla.

4.2 Inplementazioan erabilitako tresnak

Atal honetan web gune osoa inplementatzeko erabili diren tresna guztien azalpen bat egingo da beharrezkoak diren kasu partikularrak azalduz.

4.2.1 Nginx

Web zerbitzariak martxan jartzeko apache eta nginx [8] tresnak dira ezagunenak. Proiektu honetarako nginx erabiltzea erabaki da, aldiberekotasun maila handia eskaintzeko kapaza delako, apacherekin konparatuz errendimendu hobea lortzen delako eta konfigurazioaren konplexutasuna baxuagoak delako.

Web eskaera bakoitzerako prozesu berriak sortu beharrean, nginx-ek gertaeretan oinarritutako ikuspegi asinkronoa erabiltzen du, non eskaerak hari bakarrean kudeatzen diren. Nginx-ekin, prozesu maisu batek langile prozesu anitz kontrola ditzake. Prozesu maisuak langileen prozesuak mantentzen ditu, eta horiek dira benetako prozesamendua egiten dutenak. Apacheren alderdian berriz ez da aldiberekotasun maila bera lortzen, eskaera bakoitzarentzat hari berri bat sortzen delako.

Nginx kode irekiko doako softwarea eskaintzen du hasiera batean eta hau izan da proiektu honetan hautatu den aukera. Hala ere badago nginx plus izeneko ordainpeko aukera bat konfigurazio gehiago eta laguntza zerbitzua eskaintzen duena. Proiektu honetan nginx-ek eskaintzen dituen oinarritzko baliabideak erabili dira eta ondorioz ez dira aspektu teknikoetan sakonduko.

Konfigurazioen alderdian, soilik */etc/nginx/sites-available/default* fitxategian egin dira aldaketak. Honetan, web gunearen erro direktorioa zehaztu da eta PHP teknologia ahalbidetzeko hainbat konfigurazio burutu dira ere.

4.2.2 HTML

HTML marka lengoia, web gune estatikoak sortzeko erabiltzen den teknologiarik ohikoena da eta etiketa batzuen bitartez web orrialdeak nahi diren moduan egituratu daitezke. Hurrengoak dira web gunea osatzeko sortu diren html fitxategi guztiak (fitxategi bat hizkuntza bakoitzarentzat).

- **Orri nagusia:**
index.html, index_en.html, index_eus.html
- **Esperimentu bat jada martxan dagoela jakinarazteko orrialdea:**
en_marcha.html, running.html, martxan.html
- **Unean esperimenturik martxan ez dagoela jakinarazteko orrialdea:**
no_exp.html, no_exp_en.html, ez_exp.html

- **Esperimentura bukatu dela jakinarazteko orrialdea:**
final.html, final_en.html, bukaera.html
- **Jarraibideen orrialdea:**
instrucciones.html, instructions.html, jarraipenak.html
- **Formularioari buruzko laguntza orrialdea:**
ayuda.html, help.html, laguntza.html

Beste PHP fitxategietan ere HTML lengoaia erabili da web gunean pantailaratu beharreko gauza guztien posizio, diseinu eta funtzionamendua zehazteko.

4.2.3 PHP

Web gune osoa burutzeko gehien erabili den lengoaia PHP izan da. PHP web-zerbitzariaren aldeko script-ak burutzeko lengoaia da eta honen bitartez, HTMLko informazio estatikoa eta script-ak konbinatzeko aukera ematen du. Zerbitzarian exekutatzeko den kodea da eta ezaugarriak nabarmenena, HTML orriak dinamikoki sortzen dituela da. Formulario bateko datuen prozesaketa, eduki dinamikokoak dituzten orrien sorrera, fitxategiak ireki, irakurri, idatzi, ezabatu eta datubaseen atzipena burutu daiteke besteak beste.

Web gune zehatz honetan hurrengo kasutan dinamikotasuna lortu da: esperimenturen bat jada martxan badago, erabiltzaileak ezingo du formularioa duen orrialdea atzitu; uneko esperimentu bat ikusi nahi bada eta esperimenturik martxan ez badago, ez da daturik bistaratuko; uneko esperimentuaren datuak denbora errealean erakutsiko dira eta esperimentua bukatzean automatikoki karpeta pertsonala sortu eta karpeten orrialdera birbideratuko zaio erabiltzaileari.

Karpeten edukiak ikusteko, fitxategi-sistemaren barnean dagoen proiektuetarako karpeta arakatzen da modu errekursiboan. Honi esker, erabiltzaileak bere karpetan klik eginez, burutu duen esperimentuaren fitxategiak ikusi ditzake. Fitxategien izenetan klik eginez esperimentuaren emaitza fitxategiak jaisten dira.

Bestalde, erabiltzaile batek bere esperimentuaren saiakeraren karpeta ezabatu nahi badu, eskuragarri dagoen zabalgarri barran ezabatu nahi duen karpeten izena aukeratuko du eta ezabatzeko botoian sakatuko du.

Azkenik, erabiltzaileak programa nagusia web gunetik exekutatzeko edo exekuzioa gelditzeko, PHP erabili da ordenagailuaren terminalean komandoak idatziz. Zehazki PHP-k eskaintzen duen `system()` deia erabili da bi funtzio hauek burutzeko.

Sortu diren php fitxategiak hurrengoak dira:

- **Formularioa:**
form.php, form_en.php, formularioa.php
- **Jarraibideak:**
instrucciones.php, instructions.php, jarraibideak.php
- **Burututako proiektu zerrenda:**
proyectos_lista.php, proyect_list, proiektu_lista
- **Esperimentuaren exekuzioaren grafiko eta datuak irudikatu:**
visualizar_exp.php, visualize_exp.php, irudikatu_esp.php
- **Esperimentua martxan badago datuak erakutsi:**
visualiar.php, visualize.php, irudikatu.php

4.2.4 Javascript

JavaScript berriz, gertaerei zuzendutako lengoia da eta bezero zein zerbitzarietan erabili daiteke, normalean HTML orrien barruan daude kokatuak edo bestela beste fitxategi dedikatu batean. JavaScript web nabigatzailean exekutatzen den kode interpretagarria da (pertsona guztientzat ikusgai) eta hainbat gauza egiteko erabili daiteke, hala nola, HTML orrietan objektu aktiboak sartzeko (kalkulagailuak, kalkulu-orriak,...), orriaren aurkezpenean aldaketa dinamikoak egiteko (menuak, dokumentuen eguneraketa zatika...), interakzio-gaitasuna areagotzeko (formularioetan erroreen detekzioari eta tratamenduari aurre egiteko...) eta abar.

Web gunean, lau javascript funtzio desberdin erabili dira. Funtzioak sortzeko javascript tresnak eskaintzen duen *async* aukera erabili da, javascripteko promesekin lana errazteko.

Sortu diren lehenengo bi javascript funtzioak emaitzak grafikatzeko erabiltzen dira: elikadura presioaren grafikoa, iragazkortasun presioaren grafikoa, elikadura presioaren bigarren deribatuen leunketaren grafikoa eta iragazkortasun presioaren bigarren deribatuen leunketaren grafikoa. Funtzioen izenak *grafikasortu()* eta *datuakLortu()* dira.

Lehenik *datuakLortu()* funtzioa aplikatuko da. Kasu honetan, csv fitxategiak errenkada eta zutabeetan bereiziko dira bi bektore sortuz, presioaren balioak gordetzeko eta presioak lortu diren unekak gordetzeko. Split() funtzioaren bitartez ',' karakterean oinarrituz

banatuko da testua errenkadetan. `Slice()` funtzioak berriz, aurretik banatutako datuak zati txikiagoetan banatuko ditu denboraren eta presioen balioak isolatuz. Funtzioak, lortutako bi bektoreak bueltatuko ditu.

Bestalde, grafikoak sortu eta irudikatzeko, `datuakLortu()` funtzioak bueltatutako bi bektoreak erabiliko dira (bektoreak x eta y ardatzetan erabiliko dira). Halaber, *Chartjs* izeneko liburutegi bat erabili da grafikoak sortzeko. Kasu honetan marradun grafikoa erabiltzea erabaki da, presioen datuak irudikatzeko modu egokiena delako.

Hirugarren funtzioa, `lortuDatuak()` izeneko, formularioan betetako eremu guztiak irudikatzeko erabiltzen da. Honela, esperimentuaren uneko datuak dituzten grafikoarekin batera, esperimentuari dagokion formularioan adierazitako datuak taula batean erreprezentatzen dira erabiltzaileak bere saiakera dela jakin dezan. Aurreko kasuaren antzera csv fitxategia errenkada eta zutabeetan banatzeko `split()` eta `slice()` erabili da.

Laugarren eta azkeneko funtzioa, hau da, `setTimeout()` funtzioa, orrialdea guztiz freskatzeko erabiltzen da, grafikoetan irudikatutako datu guztiak denbora errealean eguneratu daitezzen. Orria bost segundoro freskatzea erabaki da.

Bestalde, formularioaren errore-tratamendurako eta aukeratutako eremuen arabera beste eremu batzuk ezeztatzeko, javascript erabili da ere.

4.2.5 CSS

CSS teknologia, arakatzzaileak web orrietako elementuak nola bistaratu behar dituen adierazteko eta kontrolatzeko aukera ematen duten estilo-orriak dira eta diseinu grafikoa zehazten dute, hau da, informazioaren aurkezpena. Tresna honen bitartez letraren kolorea, fondoaren kolorea, web orriko elementuen posizioak zehaztu eta aurkezpenarekin zerikusia duten beste hainbat gauza burutu daitezke.

Web gune honetan CSS fitxategi bakarra sortu da *style.css* izeneko. Bertan, orrialdeetan ikus daitezkeen argazki, letra, sakagailu, esteka eta diseinu orokorra zehaztu da.

4.2.6 XSLT

XSLT (eXtensible Stylesheet Language Transformations) XML lengoaiaren estiloarentzat erabiltzen da eta XML dokumentuak beste formatu batzuetan bilakatzeko gaitasuna du, hala nola, HTML dokumentuetan, testu lauan, pdf-an eta abar.

Web gune honetan, xslt dokumentu bakarra sortu da *formato_ficheros.xslt* izenekoa. Hasi-eran, fitxategi-sistema arakutzen denean erakusten diren fitxategi guztiak html dokumentu formatu gabe baten moduan irudikatzen dira. Web guneak duen estiloa jarraitzeko, aipatutako xslt fitxategia sortu da, web gunean dauden karpeten fitxategiei estiloa aplikatzeko. Honela, edozein karpetatan sartzera, barruko fitxategi guztiak modu txukonean irudikatzen dira, diseinatutako estilo pertsonala jarraituz.

4.2.7 Bootstrap

Web-aplikazioa garatzeko Bootstrap [9] izeneko framework-a erabili da. Bootstrap-ek HTML, CSS zein JavaScript tresnen adibide eta laguntzak eskaintzen ditu aplikazioen front-end zatia garatzen laguntzeko. Plataforma honen bitartez, hainbat eredu eta inplementazioen azalpenak irakurri dira proiektu honen web orria sortzeko.

4.2.8 Chartjs

Chartjs [10] kode irekiko javascript liburutegia da eta datuak modu erraz eta intuitiboan irudikatzeko erabiltzen da. Liburutegi hau erabiliz, hainbat grafiko sor daitezke, hala nola, barra diagramak, marradun grafikoak eta abar. Liburutegi honen erabilera ezinbestekoa izan da esperimentuan zehar lortu diren iragazkortasun presioaren eta elikadura presioaren balioak lortu eta grafikoak sortzeko.

5. KAPITULUA

Ondorioak

5.1 Proiektuaren ondorioak

Proiektuaren ideia nagusia asetasunez burutu da. C lengoian inplementatu den komunikazio modulua diseinatu da esperimentuaren datu guztiak lortuz eta beharrezkoak diren emaitza-fitxategiak sortuz. Gainera web moduluen bitartez proiektuaren hainbat alderdi maneia daitezke erosotasun osoz, esperimentuak martxan jarri eta emaitzak ikusteko erraztasunak emanez.

Komunikazio moduluen alderdian, etorkizunean beste beharren bat edo hobekuntzaren baten beharra badago, programa nagusia malgutasunez alda daiteke. Web moduluak segurtasunaren alderdiari dagozkion hobekuntzak inplementatzeko potentzial handia dauka, proiektu honetan oinarritzko segurtasun neurriak kontuan hartu direlako soilik.

Proiektuaren irismena zehaztu zenean oinarritzko komunikazio eta web modulu funtzionalak sortzea adostu zen. Modulu eta inplementazio guztiak adostu zen moduan burutu dira beharrezkoak izan diren proba, simulazio eta saiakerak modu eraginkorrean aplikatuz.

5.2 Ondorio pertsonalak

Proiektuaren garapena oso aberasgarria izan da. Alde batetik, konputagailuaren ingeniarriztan zerikusia duen inplementazio bat burutzeko aukera izan dudalako eta beste aldetik,

errealitatean erabiliko den proiektu batean lan egiteak motibazio estra bat suposatu delako. Gradu amaierako lana garatzen hasi baino lehen, soilik gelan proposatutako ariketa edo ikasgaiak barneratutako proiektuak burutu nituen, horregatik, lan honek kasu erreal batean lan egiteko aukera eskaini dit gustuko dudan kimika bezalako gai batean sakonduz.

Halaber, Raspberry pi 4-aren moduko IoT (Internet of Things) gailuekin lehen harremana izateko aukera izan dut, geroz eta gehiago erabiltzen diren IoT gailuek duten potentzialaz ohartuz.

Batzuetan gertatzen ziren zailtasunez gain, proiektuaren garapenean lan egiten disfrutatu dut. Proiektuaren hasiera izan zen zailena, esperimentuari buruzko xehetasunak eta funtzionamenduan sakondu behar izan nuelako eta inoiz erabili ez nituen tresnak erabiltzen ikasi behar izan nuelako. Hasierako atazak garatu ondoren proiektuaren egoera hobetu zen, baina lanaren azken ataza eta inplementazioetan hainbat zailtasun izan nituen estimatutako denbora baino gehiago inbertitzea behartu zidana.

Pertsonalki kimikaren arloarekin zerikusia duten gaiak gustuko ditut; horregatik kimika fakultatearekin zerikusia zuen lan bat burutzearen ideia nire jakin-mina piztu zuen. Gainera tutoreak proiektuaren xehetasunak azaltzerakoan eta kimika fakultatea bisitatzerakoan, proiektuarekiko interesa handitu zen.

Gradu amaierako lana hasi baino lehen, C lengoaia, PHP, HTML, Javascript eta Nginx tresnen inguruan oinarrizko ezagutzak nituen eta burututako lanaren ondorioz, tresna guzti horiekiko ezagutza handitu da. Bestalde, informazio bilaketa eta inplementazioak, ikerketa eta lan gaitasunak sustatzeko balio izan dit.

6. KAPITULUA

Plangintza

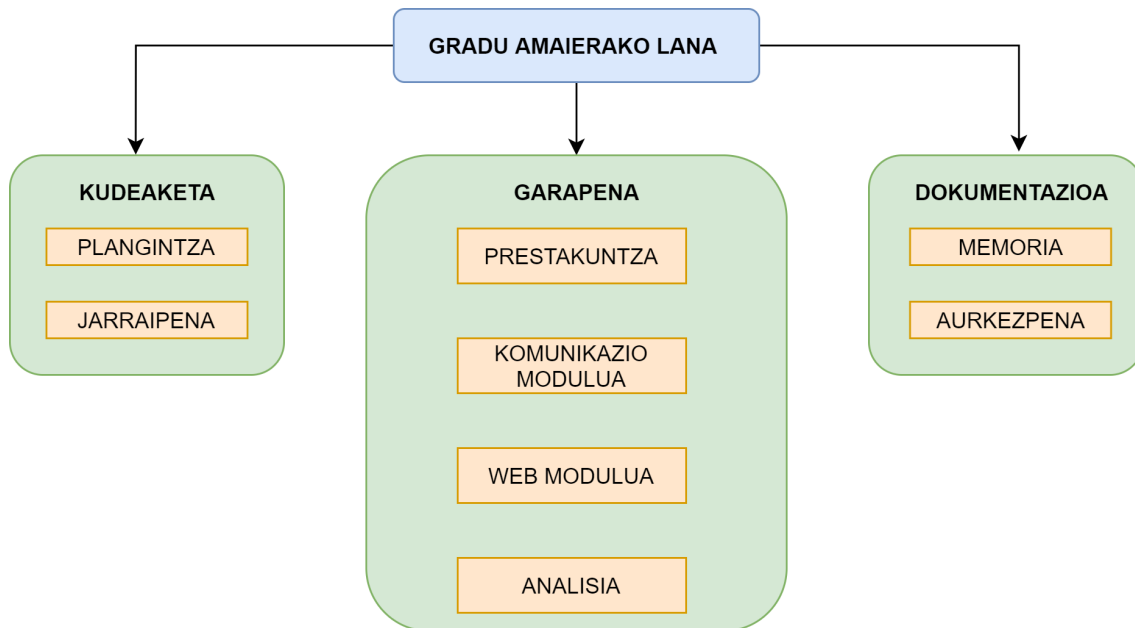
Proiektu hau egiteko, ezinbestekoa da burutu beharreko helburu eta atazak ongi planifikatzea, eginkizun bakoitzaren inguruan gerta litezkeen arriskuak jorratuz. Proposatutako atazen denbora estimazioa eta lan kargaren arteko erlazioa zuzena da, eta planifikazioan zehar kontuan hartuko diren bi irizpide garrantzitsu dira, proiektuaren helburuarekin bat eginez eta koherenteak izanik.

Atazak sailkatzeko, hiru fase identifikatu dira: kudeaketa, garapena eta dokumentazioa. Hiru fasea hauek proiektuaren ataza guztiak bilduko dituzte eta proiektuaren hasieratik, bukaerara iristeko jarraitu diren pausoak islatzen dira. Hau da, proiektuaren helburuak definitu eta aurrera eraman, proiektuaren ekintza praktikoak burutu finkatutako helburuak betez eta azkenik, horiek nola gauzatu diren azaldu, hurrenez hurren.

Bestalde, aipatutako fase bakoitzak beste azpifase batzuk izango ditu eta azpifase horiek, ataza zehatzagoak bilduko dituzte. Honen helburua, planifikazioaren zehaztasuna eta koherentzia mantentzea da, atazak era errazagoan identifikatzeko.

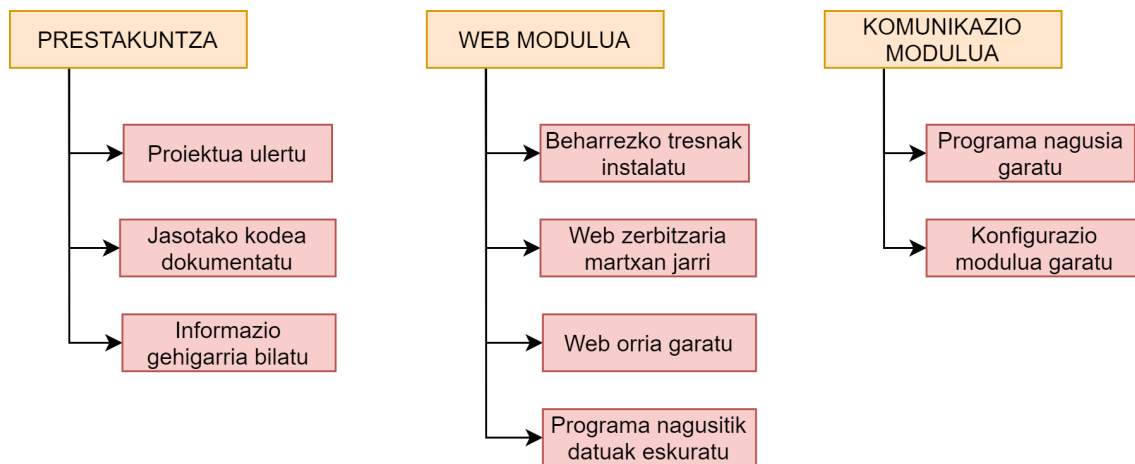
6.1 LDE diagrama

Faseen sailkapena errazago ulertzeko, LDE (Lanaren Deskonposaketa Egitura) diagrama erabili da. Aipatutako faseak eta ataza nagusiak ondorengo diagraman ikus daitezkeenak dira:



6.1 Irudia: Proiektuaren ataza nagusiak

Garapenaren atalean hurrengo azpiatazak bereiz daitezke:



6.2 Irudia: Garapenaren azpiatazak

6.2 Faseen deskribapena

Atal honetan, LDE diagraman erakutsi diren fase, azpifase eta atazen azalpen zehatza burutuko da. Diagrama horretan agertutakoak, ezinbestekoak izango dira lanaren denbora estimatzeko eta lan banaketa kudeatzeko.

6.2.1 Kudeaketa

Kudeaketa faseak proiektuaren iraupen bera izango du. Hau da, proiektuarekin hasi bezain laster, lanaren antolakuntza eta plangintza egin beharko da, eta ondoren, egindako estimazioaren jarraipena eta kontrola landu beharko da, ustekaberik gerta ez dadin eta arriskuak uneoro kontrolpean izateko.

Aipatutako arrisku hauek, proiektuaren hasieran kontuan izan behar den aspektu garrantzitsu bat da eta horregatik denbora nahikoa eskaini beharko zaio arriskuen identifikazioari proiektuaren hasieran, proiektuak aurrera egin ahala gerta litezkeen arriskuei era errazean antzeman eta gainditzeko.

6.1 irudiko diagraman ikus daitezkeen bezalaxe, fase hau, bi fase txikiagotan bana daiteke. Alde batetik, plangintza, eta bestetik, jarraipena bereizten dira.

- **Plangintza:** atal honen lehen eginbeharra proiektuaren helburuak identifikatzea da. Behin helburuak finkatuta daudela, proiektuaren ataza desberdinak zeintzuk diren adieraziko dira, eta ondoren, horiei eskaini beharreko orduen estimazioaren kalkulua eta gerta daitezkeen desbideratzeen arriskuen zehaztapena egingo da.
- **Jarraipena:** atal honetan, plangintzan egindako estimazioak betetzen ari diren aztertutako beharko da. Gainera, aurreikusitako atzerapenak gertatzen ari diren, zenbateko eragina duten eta atzerapen horien arrazoiak azalduko dira. Honela, erraz jakingo da zenbat denbora eskaini zaion ataza bakoitzari momenturo.

Horrez gain, proiektuaren tutoreekin bi astero biltzea adostu da. Horrela, sortu daitezkeen zalantzak argitu daitezke, ustekabeak eta atzerapenak ekidinez eta lan-karga hobeto banatuz.

6.2.2 Garapena

Fase honi eskainiko zaio denbora eta lan-karga gehien, proiektuaren helburuak lortzeko egin beharreko lan praktikoa guztia barneratzen baitu.

Lehendabizi, fase nagusi hau beste azpifaseetan banatu da: prestakuntza (PR), web modulua (WM), komunikazio modulua (KM) eta analisisa (AN). Ataza bakoitzak parentesien artean adierazitako kodea propioa du, testuan zehar aipamenak egin behar direnean erraz burutu ahal izateko.

- **Prestakuntza:** proiektuan zehar erabiliko diren tresna desberdinei buruzko informazioa bilatu eta ulertuko da fase honetan. Hori dela eta, gaiari buruz egindako lan akademikoak, webgune ofizialak dakarren dokumentazioa, ikasleak bilatutako informazioa eta irakasleak eskainitako informazioa erabiliko dira.

PR fase honetan ondorengo atazak identifikatu ditzakegu:

- **Proiektua ulertu:** proiektuari buruzko informazioa irakurriko da, proiektuaren irismena, egin beharrekoak, xehetasunak eta helburuak ongi ulertuz.
 - **Informazio gehigarria bilatu:** proiektuari buruzko oinarrizko informazioaz gain, ikasleak bilatutako informazio gehigarria erabiliko da proiektuaren ulermen osoari laguntzeko.
 - **Jasotako kodea dokumentatu:** irakasleak, proiektu honen aurreko bertsioaren kodea bidali ondoren, programa horren funtzionamendua, neurketak eta prozedurak ongi ulertu eta dokumentatuko da.
- **Web modulua:** Esperimentua hasieratu, gelditu edo emaitzak ikusteko ezinbestekoa den atala da hau. Erabiltzailearentzat eroso den interfazea diseinatu, orrialdeen arteko koherentzia mantenduko da edozein esperimendu modu egokian betetzeko.

WM fase honetan ondorengo atazak identifikatu ditzakegu:

- **Beharrezko tresnak instalatu:** Webgunea garatzeko beharrezkoak diren tresna guztien instalazioa burutu eta haien funtzionamendua aztertu. Martxan jarriko den web zerbitzari posibleak ikertu eta bat aukeratu (kasu honetan nginx aukeratu da).
 - **Web zerbitzaria martxan jarri:** web zerbitzaria martxan jarri, sare lokaleko erabiltzaileek sortutako web orria atzitzeko.
 - **Web orria garatu:** esperimentuan lortutako emaitzak, datuak eta grafikoak ikuskatzeko web orrialdea sortu, nginx bertako zerbitzari batean exekututako dena. Honen bitartez, ikasle zein irakasleek esperimendua kontrolatu dezakete.
 - **Programa nagusitik datuak eskuratu:** DAQ gailuarekin burutzen dituen komunikazioen ondorioz lortutako balio eta emaitzak web gunean bistaratzeko, web orriak emaitza fitxategiak atzitu behar ditu uneko balioak erakutsiz.
- **Komunikazio modulua:** lortu nahi diren helburuak gauzatzeko, eraikiko diren programen lengoaiak eta diseinuak zehaztea premiazkoa da. Horren ostean, garapenean

diseinu zehatza eta eraginkorra jarraitu beharko da. Ataza honetan, DAQ gailua eta ordenagailuaren arteko komunikazioa betetzeko azpiatazak deskribatzen dira.

KM fase honetan, ondorengo atazak gauzatuko dira:

- **Programa nagusia:** serie portutik bidaltzen diren mezuak aztertu behar dira. Horretarako serie portua ireki, itxi, idatzi eta irakurtzea baimenduko duen programa bat sortu behar da. DAQ gailuarekin komunikazioa burutu behar da, hark bidalitako datuak prozesatzeko eta emaitzak era egokian irudikatze-ko. Laburbilduz, atal honetan, serie portua kudeatzeko programa kodetuko da, esperimentuaren emaitzak lortuko dituen programa sortuz, serie portutik lortutako informazioaz baliatuz.
- **Konfigurazio modulua garatu:** Serie portutik beharrezkoak diren datuak lortzeko, serie portuaren konfigurazioa doitu behar da. Horretarako, kanpoko konfigurazio fitxategia irakurriz, programa nagusiak zehaztu diren konfigurazio parametroak eguneratuko ditu gailuen arteko komunikazioa ahalbidetzeko.
- **Analisia:** fase honetan, proiektuan zehar bete diren inplementazioen eta azalpenen lotura burutzen da, prozesu osoaren ondorioak lortuz.

AN fase honetan ataza hau bereiz daiteke:

- **Ondorioak:** burutako lanean oinarrituz, lortu den inplementazioaren bukarako hausnarketa burutu.

6.2.3 Dokumentazioa

Fase honetan, proiektua dokumentatuko da, egin diren lan guztiak azalduz eta horiekin lortutako emaitzak erakutsiz. Lana edonori modu antolatuan aurkezteko testua egokia eta koherentea izan behar du.

- **Memoria:** atal hau betetzeko, *Overleaf* tresna erabiliko da, proiektua era txukunean dokumentatu ahal izateko. Gainera, edizioa internet bidez burutu denez, edozein momentuan bai tutoreek baita ikasleak ere memoria atzitzea posible izan dute. Bestalde, Overleaf editorearen komunitatea handia denez, edizioari buruzko zalantzak erraz aurki daitezke interneten.

- **Aurkezpena:** lanaren sintesia landuko da, aurkezpenean informazio esanguratsue-
na azaldu dadin, kontuan hartuz iraupena 15-20 minutu tartekoa izango dela. Hor-
taz, edukia eta hori aurkezteko era (estiloa, hizkuntza...) zainduko dira honakoan.

6.3 Lanen estimazioak eta desbiderapenak

Jarraian 6.1 taulan ikus daiteke ataza bakoitzari eskaini beharreko ordu kopurua eta bene-
tan bakoitzarekin pasatako ordu kopurua.

	Aurreikusitako orduak	Benetako orduak
KUDEAKETA	43h	33h
Plangintza	28h	18h
Jarraipena	15h	15h
GARAPENA	167h	199h
Prestakuntza (PR)	16h	16h
Proiektua ulertu	6h	3h
Informazio gehigarria bilatu	2h	1h
Jasotako kodea dokumentatu	8h	12h
Web Modulua (WM)	42h	72h
Beharrezko tresnak instalatu	1h	1h
Web zerbitzaria martxan jarri	1h	4h
Web orria garatu	35h	57h
Programa nagusitik datuak eskuratu	5h	10h
Komunikazio modulua (KM)	100h	109h
Programa nagusia garatu	85h	93h
Konfigurazio modulua garatu	15h	16h
Analisia (AN)	9h	2h
Ondorioak	9h	2h
DOKUMENTAZIOA	90h	84h
Memoria	75h	70h
Aurkezpena	15h	14h
GUZTIRA	300h	316h

6.1 Taula: Proiektuaren aurreikusitako eta benetako orduak.

6.4 Arrisku plana

Proiektu baten garapenean zehar, oso garrantzitsua da gerta daitezkeen arriskuak kon-
tuan hartzea. Arriskuak kontuan ez badira hartzen, atzerapenak, galerak edo beste hain-

bat ezusteko gerta daiteke. Normalean proiektuak burutzean arrisku plan bat idazten da, agertu daitezkeen arrisku guztiak bilduz eta horietako baten bat gertatzerakoan zer egin adieraziz. Arrisku hauek estimatutako denboraren desbiderapenarekin du erlazio zuzena eta desbiderapenik nahi ez dugunez, plan sendo bat sortu behar da.

Lehenik, gerta liteke ataza batentzat estimatutako denbora luzatzea ataza jakin horren inguruko ezjakintasuna, zalantzen edo zailtasunaren ondorioz. Arrisku hori arruntena da eta errealitatean eskaini beharreko denbora askoz handiagoa izatea dakar. Hortaz, denbora estimatzerakoan, zailtasunak ekar ditzakeen atazei denbora gehiago eskaini behar izatea suposatu beharko da, planifikazioan estimatutako denbora handituz. Ordea ataza txikien estimazioa eta kontrola zehatzagoa izango dela espero denean, ez da hainbesteko arriskurik aurreikusten eta denbora gutxiago eskainiko zaiola ondorioztatu daiteke. Bestalde, momenturen batean ataskaturik gelditzerakoan, irakasleei laguntza eskatzea irtenbide ona izan daiteke.

Lanak ongi dokumentatzeko ohitura eduki arren, gradu amaierako lanaren memoria egiterako orduan, luzera dela eta, atzerapenak sortzeko arriskua egon daiteke. Proiektuko datei dagokienez, ez da epeak betetzeko arriskurik aurreikusi.

Beste arrisku nabarmena COVID-19-a da. Kutsadurak direla eta, gerta liteke GrAL-eko aurkezpena atzeratu behar izana edo bileraren bat atzeratu behar izana. Gainera, estatu mailan kutsadura indize handia mantentzen bada, beste konfinamendu orokor bat gertatzea posible izango litzateke. Ondorioz fisikoki burutu behar diren atazak, hala nola, kimika fakultatean burutzeko lanak, ezingo dira burutu desbiderapen orokor batean bukatuz. Horretarako, kimika fakultateko laborategiko makinaren portaera simulatzea ezinbestekoa da.

6.5 Desbiderapen analisia

Orokorrean proiektuaren atazak burutzeko garaian, lan-karga handiena zuten atazak izan ziren desbiderapen handiena sortu zutenak. Batez ere web moduluan eta komunikazio moduluan gertatu ziren desbiderapen nabarmenenak hasieran estimatutako denborarekin konparatuz.

Jarraipenari dagokionez, bilerak modu presentzialean era arruntan burutu ziren eta COVID-19-ak sortu ahal zituen arazoak ez ziren gertatu azken batean.

Garapenari dagokionez, berriz, hasierako prestakuntza pixka bat zaildu zen, zenbait tresna

eta lengoaiekin nuen lehen kontaktua zelako. Bestalde programa nagusian hainbat arazo egon ziren eta kimika fakultatera gutxitan joan ginen. Honek zekarren arazoa, programa etxean simulatu beharra izatea izan zen eta horregatik programaren lehen proba errealean, komunikazio moduluak ez zuen espero bezala funtzionatu.

6.6 Lan-metodologia

Lan-metodologiari dagokionez, atazak epean burutzeko, tutorearekin komunikazio erregularra jarraituko da. Horrela, zalantzak garaiz argituko dira, eta egin beharreko hurrengo lanak denborarekin ulertu eta finkatuko dira. Bilerak tutoreak proposatutako atazak betetzerakoan burutuko dira, baina bai gutxienez bi astero tutorearekin hitz egingo dela lanaren jarraipena epearen barruan burutzen ari dela jakiteko. Gainera malgutasunarekin beste edozein momentuan bilerak egingo dira, zalantza puntualak sortzen badira, baita posta elektronikoz ere, ahal den neurrian.

Proiektuaren ezaugarriak direla eta, garrantzi handia izango du ikerketa prozesuak batez ere gailuena arteko komunikazioan. Hala ere, oztopo handiak sortzen badira, egin beharrekoak gehiago zehaztuko dira atzerapenak txikituz. Ataza bakoitza egin ahala, ikertutakoa eta programatutako balizkoa izango da proiektuaren mamia ongi ulertzeko.

Gradu amaierako lana ekainean entregatzea erabaki zen, gainerako ikasgaiak une berean bukatzen direlako eta honela, lan guztiak lasaitasunez egiteko denbora egongo delako. Izan ere, lanak egiteko beharrezkoa den astia hartzeak badu bere garrantzia ikasitakoa hobe barneratzen baita.

Bibliografia

Erabilitako estekak

- [1] DAQ prozesua:
<https://www.omega.com/en-us/resources/daq-systems>
- [2] Digitalizazioa
https://en.wikipedia.org/wiki/Analog-to-digital_converter
- [3] Termobikoteak:
<https://www.omega.co.uk/prodinfo/thermocouples.html>
- [4] AC eta DC korronea:
[https://www.onulec.com/blog/noticias-sector-electrico/
208-diferencias-entre-corriente-continua-dc-y-corriente-alterna-ac](https://www.onulec.com/blog/noticias-sector-electrico/208-diferencias-entre-corriente-continua-dc-y-corriente-alterna-ac)
- [5] Presioaren neurketa:
[https://www.ni.com/es-es/innovations/white-papers/06/
how-to-measure-pressure-with-pressure-sensors.html](https://www.ni.com/es-es/innovations/white-papers/06/how-to-measure-pressure-with-pressure-sensors.html)
- [6] Serie portuaren kontrola:
https://www.cmrr.umn.edu/~strupp/serial.html#2_5_3
[https://blog.mbedded.ninja/programming/operating-systems/linux/
linux-serial-ports-using-c-cpp/](https://blog.mbedded.ninja/programming/operating-systems/linux/linux-serial-ports-using-c-cpp/)
- [7] Konfigurazioa zehazteko termios egitura:
<https://code.woboq.org/gcc/include/bits/termios.h.html>

Interesekeo estekak

- [8] [Nginx](#)
- [9] [Bootstrap: PHP, HTML, Javascript, CSS](#)
- [10] [Chartjs](#)
- [11] [W3Schools tutorialak: PHP, HTML, Javascript, CSS](#)
- [12] Erabilgarriak izan diren foroak:
 - [StackOverflow](#)
 - [Debian Forums](#)
 - [Geeks for Geeks](#)

Kontsultatutako Manualak

- [13] PR4000 DAQ gailuaren manuala:
https://drive.google.com/file/d/1qJQ_i7dhFwHIsLB8pLrBUj6odyzoWYA1/view?usp=sharing
- [14] C lengoaiaren oinarrizko manuala:
<https://drive.google.com/file/d/1q1QB5dL7nGqKrTl1o08g7qZ4yI7luSRJ/view?usp=sharing>
- [15] LabView oinarrizko manuala:
https://drive.google.com/file/d/1-JsQmOwbFzmMzFZJ9IeSCTneGrTKEZ_I/view?usp=sharing

A. ERANSKINA

Implementazioak

Eranskin honetan txostenean zehar aipatu diren inplementatutako bi moduluen kodea dago eskuragarri:

Komunikazio Modulua:

<https://drive.google.com/drive/folders/1er7i7w9oambuSH5dkK1dIh7iNWDsTqR8?usp=sharing>

Web Modulua:

<https://drive.google.com/drive/folders/1JnYs66EmFoh7VImWwOvxRZ7ZoavL2zGu?usp=sharing>

Esperimentuaren jarraibideak:

Euskaraz: <https://drive.google.com/file/d/1oZY2dUDFSHaym86k--ZgLOvKB9SExfez/view?usp=sharing>

Gazteleraz: https://drive.google.com/file/d/1esG9e02-q4DHQyZyJFUrz_c9YsA5wKUH/view?usp=sharing

Ingeleseaz: <https://drive.google.com/file/d/1T6W8EczeG9JqyhF23YZjURVbw7VCSy10/view?usp=sharing>

Laguntza orrialdeak:

Euskaraz: <https://drive.google.com/file/d/1Sk7ICEtQ28UGp3KDYVLOBVw763W9TBhI/view?usp=sharing>

Gaztelera: <https://drive.google.com/file/d/1DFzQG2DaKjhdcb2REtrFTKviIrCtIpkX/view?usp=sharing>

Ingelese: https://drive.google.com/file/d/13NSrLZUAI5VKK66_AvohWG2LYVO-Qtt0/view?usp=sharing