

■ **Gradu Amaierako Proiektua** ■

Konputazioa

**Maribloki: Hezkuntzara bideratutako
programatzen ikasteko web aplikazioa**

Aizpea Babaze

2017 - ekaina

Tutoreak: Nestor Garay eta Edune Larraza



<https://github.com/aepzia/maribloki.git>



maribloki.esy.es

Proiektua, hau posible egin duten guztiei dedikatu nahi diet. Familiakoei, amari bereziki, nire buruari jartzen dizkionak helburu guztiak betetzera bultzatzegatik, behar dugunean guri laguntzeko beste guztia alde batera uzteagatik eta gure zoriona beste edozerren gainetik jartzeagatik. Anttoni, Granadatik bada ere, edozein momentuan bere aholkuak eskuragarri ditudalako. Iñakiri, gaiaz asko ez dakien arren beti jarri baitu interesa, eta Jokini zure xelebrekeriek liluratu egiten nautelako, gezurra badirudi ere, ezberdin egiten gaituenak baina gauza gehiago ditugu komunean. Aitari, arrazoiekin bat ez natorren arren, informatika egitera bultzatu nindun lehena izan baitzinen.

Kuadrilakoei, duela urte askotatik ni aguantatzeagatik. Guretako bakoitza besteekiko zeharo ezberdina den arren, orduak eta orduak pasa ditzakegulako 5 elkarrizketa batean izaten. Nere larrosa maiteei, asko ikusten ez bagara ere, beti entzuteko prest zaudetelako. Muturbeltz, hori bai kasualitate polita, egun batean ezagutu eta ez gara inoiz banandu. Eta Arkaitzi, duela lau urtetik alboan daraman pertsona delako. Eskerrik asko guztioi, bizitako momentu guztiengatik eta geratzen zaizkigunengatik.

Nola ez, Informatika Fakultateko kide guztiei. Kontseiluko kideei, adiskidetasunaz eta formakuntzaz beteriko 4 urte hauengatik. Klasetik kanpo egin ditudan lagunei eta klasekideei. Nere bizitzako 4 urte hoberenak eman dizkidazuelako eta zuengatik izango ez balitz, ez nintzatekelako graduatzeaz egongo. Eskerrik beroenak Xixari, lehenengo mailatik edozertan laguntzeko prest egon delako eta azken hilabetetan lagundu didan guztiagatik. Rubeni, Suitzan ezagutu bagina ere, Fakultateak asko batu gaituelako.

Azkenik, eskerrik asko Nestor Garay eta Edurne Larrazari, proiektuan zehar emandako aholkuengatik, eta dedikatutako denboragatik

Maribloki: Hezkuntzara bideratutako programatzen ikasteko web aplikazioa

Aizpea BABAZE AIZPURUA

Laburpena

Informatika Ingeniaritzako graduko konputazio zientzietako espezialitaterako Gradu Amaierako Lana da hau. Bertan, hezkuntzara bideratutako programatzen ikasteko web aplikazio bat garatu da. Aplikazio honek minijokoen bitartez programatzen ikasteko aukera ematen die ikasleei eta irakasleek hauen emaitzak ikusi eta aztertu ahal izango dituzte.

Abstract

This is a Bachelor Thesis in the Computer Science speciality. In this project a web application for educational purpose has been developed. Using this application students can learn programming through minigames, and teachers could be able to see and analyze the results.

Resumen

Este es un Proyecto de Fin de Grado de la especialidad de Computación de Ingeniería Informática. En este proyecto, se ha desarrollado una aplicación web para aprender a programar dirigido a la educación. Esta aplicación permite a los estudiantes aprender a programar mediante minijuegos y los profesores podrán ver y analizar los resultados.

EDUKIAK

Laburpena	v
Irudien zerrenda	x
Taulen zerrenda	xii
Kodeen zerrenda	xiv
1 Sarrera	1
1.1 Testuingurua	1
1.2 Proposamena	1
1.3 Motibazioa	2
2 Plangintza	5
2.1 Proiektuaren irismena	5
2.1.1 Helburu nagusiak	5
2.1.2 Atazak	5
2.1.3 LDE diagrama	8
2.2 Denboraren plangintza	9
2.2.1 Atazak eta denbora estimazioa	9
2.2.2 Atazen planifikazioa	9
2.3 Arriskuen kudeaketa	9
3 Marko teorikoa	13
3.1 Blockly	13
3.1.1 Web plataforma eraikitzen	13
3.1.2 Blokeen hizkuntza	14
3.1.3 Blokeak sortzen	14
3.2 Blockly games	16
4 Erabilpen gida	17
4.1 Irakaslea	18
4.2 Ikaslea	21
4.3 Kudeatzailea	21

5	Erabilitako teknologia	23
5.1	MySQL	23
5.2	Php eta html	23
5.3	Java script	23
5.3.1	chart.js eta canvas.js	23
5.3.2	jsPDF	24
5.3.3	AJAX	24
5.4	Hostinger	24
5.5	Gitlab eta github	24
5.6	Texmaker	25
6	Proiektuaren garapena	27
6.1	Minijokoak	27
6.2	Denbora eta saiakera kopurua	28
6.3	Datu-basea	29
6.4	Orri nagusia eta itxura orokorra	31
6.5	Saioa hasieratu	31
6.6	Estatistikak	31
6.6.1	Bariantza	31
6.6.2	Desbideratze estandarra	32
6.6.3	Batez besteko eta desbideratze estandarraren erabilpen batura	33
6.6.4	Garapena	33
6.7	Erregistratu	34
6.8	Emaitza pertsonalak ikusi	35
6.9	Datuak editatzea	36
6.10	Itsuentzako pantaila	36
6.10.1	Irisgarritasuna	36
6.10.2	Laguntza teknikoak	36
6.10.3	Pantaila irakurleak	36
6.10.4	Blockly irisgarria	37
6.10.5	Garapena	38
7	Emaitzak	41
8	Jarraipena eta kontrola	43
8.1	Irismena	43
8.1.1	Proiektuan lortu diren helburuen deskribapena	43
8.1.2	Irismenarekin erlazionaturik gertatu diren arazo deskribapena	43
8.1.3	Kanpo geratu diren helburuen deskribapena	44
8.1.4	LDE-an egindako aldaketak	44
8.2	Denboraren banaketa	44
8.2.1	Egin diren ataza berrien deskribapena	44
8.2.2	Ataza bakoitzari emandako denbora	46
8.2.3	GANTT diagramak jasandako aldaketak	47

9 Ondorioak eta etorkizuneko lana	49
9.1 Etorkizuneko lana	49
9.2 Ikasitako lezioak	50
Bibliografia	I
Eranskinak	III
A Laginetik lortutako emaitzak	III
A. A I Minijokoko emaitzak	III
A. B II Minijokoko emaitzak	IV
A. C III Minijokoko emaitzak	V
A. D IV Minijokoko emaitzak	VI
A. E V Minijokoko emaitzak	VII
A. F VI Minijokoko emaitzak	VIII
A. G VII Minijokoko emaitzak	IX
A. H VIII Minijokoko emaitzak	X
A. I IX Minijokoko emaitzak	XI
A. J X Minijokoko emaitzak	XII

IRUDIEN ZERRENDA

2.1	<i>LDE diagrama</i>	8
2.2	<i>Proiektuko gantt diagrama</i>	11
3.1	<i>closure-directory karpetaren lokazioa</i>	15
3.2	<i>moveForward blokearen definizioa</i>	16
4.1	<i>Erabilpen kasuen diagrama</i>	17
4.2	<i>Irakasle bezala erregistratu</i>	18
4.3	<i>Irakaslearen menu nagusia</i>	18
4.4	<i>Ikasleen lista</i>	19
4.5	<i>.csv fitxategia sortzeko pausuak</i>	19
4.6	<i>Ikasle berriak gehitu .csv fitxategi baten bidez</i>	20
4.7	<i>Ikasle berriak gehitu izena eta abizena idatziz.</i>	20
4.8	<i>Ikasleen emaitzen taula eta grafikoak.</i>	21
4.9	<i>Ikasleen menu nagusia.</i>	21
6.1	<i>compressed.js fitxategiaren kode zatia</i>	29
6.2	<i>Klase diagrama</i>	30
6.3	<i>Pantaila irakurleen bidezko ohol irisgarria</i>	40
7.1	<i>ik8 ikaslearen emaitzak</i>	41
8.1	<i>LDE diagrama</i>	45
8.2	<i>Bukaerako gantt diagrama</i>	47

TAULEN ZERRENDAA

2.1	Atazak eta denbora estimazioa	10
7.1	<i>Minijoko bakoitzak izandako ikasle kopurua</i>	42
8.1	Ataza bakoitzari emandako denbora eta aurreikusikoarekiko desbideratzea .	46

KODE ZERRENDA

3.1	<i>Blocky liburategiko baliabideak kodean gehitu</i>	13
3.2	<i>Workspacea definitu, toolboxa div barruan sartu</i>	13
3.3	<i>Toolboxa definitu, erabili nahi diren blokeekin</i>	14
3.4	<i>Sortutako kodea lortu eta exekutatu</i>	14
3.5	<i>Blokeen mezua gaztelaniaz</i>	14
3.6	<i>Aukeratutako hizkuntzaren fitzategia gehitu</i>	14
3.7	<i>Blockly games dependentziak kargatu</i>	16
3.8	<i>Hizkuntz guztiak kargatu</i>	16
3.9	<i>Ingelesa kargatu</i>	16
4.1	<i>Datu-basera konektatzeko beharrezko datuak</i>	22
6.1	<i>move_forward blokearen itxura definitzeko kodea</i>	27
6.2	<i>move_forward blokeak itzuli beharreko kodea</i>	27
6.3	<i>Denbora eta saiakera kopurua datu-basean gordetzeko ajax deia</i>	29
6.4	<i>Irakasleak erregistratutako ikasleen eta aukeratutako minijokoan lortutako datuak jaso</i>	33
6.5	<i>Estatistiken formulak jasotako datuei aplikatuta</i>	33
6.6	<i>Erabiltzailea automatikoki sortzea</i>	35
6.7	<i>pdf fitzategi bat sortu JavaScript bidez</i>	35
6.8	<i>Karpeta tenporalean dagoen fitzategia irakurri</i>	35
6.9	<i>Interfaze irisgarrian kodetutako kodea lortzeko botoiaren definizioa</i>	37
6.10	<i>xml fitzategiko lehenengo minijokorako definizioa</i>	38
6.11	<i>xml fitzategitik datuak kargatzeko kodea, minijokoaren mailaren arrabera</i>	39
6.12	<i>Minijokoko taularako lasterbidea</i>	40
6.13	<i>Taulan gertatzen den aldaketa ororen abisu mezua</i>	40
6.14	<i>Ahots mezuak gainezartzea ekiditeko denbora kontagailua</i>	40

1. Kapituluia Sarrera

1.1 Testuingurua

Programazioak autozuzenketa eta errore bilaketari aurre egiteko eta arazo handien aurrean ebazpen bat aurkitzeko gaitasuna garatzeko aukera ematen die ikasleei. Gaitasun horiek, informatikatik haratago doaz, programazioaren oinarritzko kontzeptuak eguneroko bizitzako arazoei aurre egiteko aplikagarriak baitira.

"Programatzen ikasterakoan eta pentsamendu konputazionala garatzerakoan, sormena, ekintzailletza eta kultura askea sustatzen dira, motibazioa handitzen da, autonomia hobetu... "(Jesús Moreno, 2014) , [1] erreferentzian aipatzen den bezala.

Egoera honen aurrean, irakasleek metodologia berri baten beharra dute. *Scratch* programazio lengoia bihurtu da hezkuntzara bideratutako programazio lengoia nagusia, umeentzako pentsatutako programazio lengoia bisuala baita.

Scratch programazio lengoia bisualaren eredia jarraituz, *code.org* plataformak programazio irakastearen garrantzia aldarrikatu zuen *kode ordua* kanpainarekin. *Kode ordua* ordu beteko programazio ikastaroetan datza. Ikastaro horiek code.org web orrian aurki daitezke. Bertan edozein pertsonarentzako pentsatutako hainbat ikastaro daude, 4-104 urteko pertsonentzako egokitutakoak eta 30 hizkuntza baina gehiagotan, baita euskaraz ere Informatika Fakultatean 2015ko azaroaren 7an antolatu zen itzulpen maratoiari esker.

Informatika Fakultateak ere JolasMATIKA proiektua jarri du martxan Tolosako Samaniego ikastetxearekin eta Orixe institutuarekin batera. Proiektu honek informatika teknologia izateaz gain zientzia ere badela erakutsi nahi die ikastetxeko ikasleei. Helburu hori betetzeko, informatikako ikasle batzuk ordubeteko saioak eman dituzte. Saio horietan, ordenagailu gabe jolasean eta *code.org* eta *scratch* bidez, programazioko oinarritzko kontzeptuak ikasi dira.

1.2 Proposamena

Code.org plataformako joko ezberdinak, *Google* eskuragarri duen kode libreko *Java Script* lengoaiako liburutegia du oinarri. Gradu Amaierako Lan honetan, *blockly* liburutegiaz baliatuz, joko berriak sortzea proposatu da.

Joko berriak proposatzeaz gain, jokia pasatzeko behar izandako denbora eta saiakera kopurua gordeko da, emaitza hauek aztertzeko helburuarekin. Horretarako, web

plataforma bat garatu da. Bi erabiltzaile mota daude web plataforma honetan, ikaslea eta irakaslea.

Ikasleak hainbat minijokotan jolastuko du eta honen emaitzak bere profilean gordeko dira. Irakasleak bere profilean bere ikasleen datuak ikusi ahal izango ditu, besteak beste ikasle bakoitzaren emaitzak. Gainera, zenbait ondorio ateratzeko baliagarriak izan daitezkeen estatistikak ere jardungo dira, hauen gaitasun mailen berezitasun ezberdinak ikusteko asmoz.

Azkenik, teklatu bidezko elkarrekintza onartzen duen minijokoa sortuko da, mugimendu edo ikusmen urritasuna duten erabiltzaileentzako irisgarria den hezkuntzara bideratutako minijokoak sustatuz.

1.3 Motibazioa

Gaztetik izan da nire helburu behin lan munduan sartzerakoan pertsona laguntzea. Betidanik izan ditut gustuko Matematikak, eta esperientzia pertsonalean oinarrituz, ikasleen heziketa prozesuan parte hartzeak, ikasle hauen momentuko eta etorkizuneko egoera hobera alda dezakeela uste dut. Hori, eta nik hainbeste maite izan dudana ikasgaia orokorrean ikasleengandik gorrotatua dela eta, matematikako irakaslea izateko nahia sortu zidan.

Unibertsitateko matrikula egiterakoan, Ingeniaritza Informatikoa ez nuen planteatu ere egin; Nire nahia Matematikako irakaslea izatea zen. Baina zergatia oso ondo jakin gabe, fakultate honetako irakasle batek Ingeniaritza informatikoko gradua egiteko konbentzitu ninduen. Bide honetatik ere Matematikako irakaslea izatera iritsi nintekeen, eta gainera ez nuke Bilbora joan beharko.

Lehenengo ikasturtean zehar informatikaren mundua ezagutu nuen, momentura arte niretzako guztiz ezezaguna izan zena. Honetaz gain zoritxarrez, ELA gaixotasunaren eraginak gertuko pertsona batean bizi izan nituen. Gaixotasun honekin eguneroko bizitza normaltasunez egiten ahal izateko laguntza teknikoak urriak, eta batez ere garestiak zirela ikusi nuen.

Honek guztiak nire perspektiba aldatu zuen. Informatikak behar zehatzak dituzten pertsonen eguneroko bizitza alda dezakeela konturatu nintzen, laguntza teknikoen bidez edo eta bestelako gaitasunak garatzen dituzten tresnen bidez. Arlo honetan nire esfortzu eta ezagutzak inbertitzeko interesa sortu zitzaidan.

2016ko udan, APTES (“Asociación para la Promoción de la Tecnología Social”) elkartearekin praktikak egin ahal izateko aukera izan nuen, eta beraiei esker, teknologia sozialaren mundua ezagutu nuen. Teknologia sozialak, ezagutza zientifikoaz eta teknologikoaz baliatuz, biziraupena, osasuna, hezkuntza, zahartze edota ezgaitasun arazoei erantzun egokia ematea ahalbidetzen du.

Kontziente naiz nire ametsetako lanean bukatzea oso zaila dela, eta horregatik, behintzat nire Gradu Amaierako Proiektua gai honetan zentratu nahi nuen. Hasiera batean Gradu Amaierako Lantzat, Linuxerako *Orca* software libreko pantaila irakurlearen garapenaren parte hartzea proposatu zidaten, eta dudarik gabe onartu nuen. Honen inguruan dagoen dokumentazio falta dela eta, ez nuen nire burua 300 orduko tartean ezer ikusgarria egiteko kapaz ikusten, eta ondorioz beste proposamen bat eskatu nuen.

Orca programaren inguruan irakurritako informazioa onuragarria izan da dudarik gabe, eta pantaila irakurleen eta bestelako laguntza teknikoen funtzionamendua ezagutzea

erabilgarria da edozein proiekturen irisgarritasuna bermatzeko.

Proiektu hau hezkuntzara bideratuta dago, eta gainera, dibertsitate bisuala duten erabiltzaileei egokitutako interfazea inplementatuko da. Dударik gabe nire helburuarekin bat datorren proiektu bat da, eta aurreko proiektuaren porrotaren ostean, neureganatutako zenbait ezagutza aplikatzeko aukera ikusten dut.

2. Kapituluia Plangintza

Atal honetan proiektua aurrera eramateko erabili den hasierako plangintza aurkezten da. Bertan proiektuaren irismena, denboraren plangintza eta kalitatearen kudeaketa aipatzen dira.

2.1 Proiektuaren irismena

Proiektu honen helburua, blockly liburutegia erabiliz (3.1 atala), hezkuntzarako diseinatutako programatzen ikasteko web plataforma bat sortzea da. Plataforma horren bidez, irakasleak bere ikasleen datuak eta ikasleen arteko gaitasun mailen berezitasunak ikusteko aukera izango du.

Atal honetan, proiektuan zehar erabiliko den plangintza zehaztu da. Proiektuan zehar bete nahi diren helburuak eta helburu hauek aurrera eramateko garatuko diren atazak azaltzen dira.

2.1.1 Helburu nagusiak

Helburu nagusiak, proiektuaren proposamenean definitutako hiru atal nagusiak dira. Hauek dira proiektuan zehar beteko diren helburu nagusiak:

1. Minijoko bilduma sortu
2. Ikasleen emaitzak aztertzen dituen estatistikak egin
3. Itsuentzako minijoko interfaze egokia proposatu

2.1.2 Atazak

Proiektu hau aurrera eramateko zenbait atazetan banatu da proiektua. Hauek dira proiektua garatzeko bete beharreko atazak:

1. Proiektuaren planifikazioa. Proiektuaren hasieran egin beharreko lana, proiektua nola burutuko den definitzeko.
 - 1.1. Proiektuaren irismena.
 - 1.2. Ataza eta betekizunak zerrendatu.

1.3. Denboraren estimazioa egin.

1.4. Atazen egutegia zehaztu.

1.5. Arriskuen kudeaketa.

2. Ezagutzak eskuratu.

2.1. Blockly.

3. Garapena. Irismena ahalik eta hobekien betetzeko atalak bereizten dira, garrantziaren arabera ordenaturik.

3.1. Minijokoak. Probako kasuan, 3 maila diseinatuko dira.

3.2. Denbora eta saiakera kopurua. Maila bakoitzeko ikasleak behar duen denbora eta saiakera kopurua tratatuko dira.

3.3. Datu-basea. Datu-base bat sortuko da, non ikasle, irakasle, minijokoak eta beren arteko erlazioak gordeko diren.

i. Erabiltzailea. Bi motatako erabiltzaileak egongo diren arren, erabiltzaile guztiak taula bakar batean gordeko dira, non bakoitzak betetzen duen rola agertuko den, eta ikasleen kasuan, beren irakaslea ere agertuko den.

ii. Emaita. Taula honetan, ikasleak maila bat gainditzeko behar izandako denbora eta saiakera kopurua gordeko dira.

3.4. Web plataformako orri nagusia eta itxura orokorra.

3.5. Saioa hasieratu. Lehenik esan bezala, bi erabiltzaile mota egongo dira: ikasleak eta irakasleak. Ikasleak irakaslearekin lotu ahal izateko, irakasleek erregistratu behar dituzte. Ikasleek lehenengo aldiz saioa hasten dutenean, beren informazio pertsonala eskatuko zaie.

i. Lehenengo aldiko saio hasiera. Lehenengo saio hasiera den egiaztatu beharko da, eta, hala izanez gero, informazioa eskuratzeko galdetegi bat azaldu beharko da.

ii. Saio hasiera normala. Erabiltzailearen datuak datu-basean ageri badira, ez da galdetegirik agertuko, eta honen saio hasiera normala izango da.

3.6. Estatistikak. Irakasleak ikasleen emaitza globalak ikusteko aukera izango du eta emaitza global horiei estatistikak aplikatuko zaizkie, emaitza landuagoak emateko asmotan. Estatistika horiek lantzeko, honako datu hauek hartuko dira kontuan:

i. Ikaslearen adina.

ii. Ikaslearen generoa.

iii. Ikaslearen teknologia ezagutza. Gaitasun hau ikasleari egingo zaion galdetegiaren bidez neurtuko da.

iv. Ikaslearen desgaitasun maila.

v. Eskuina edo ezkerria den.

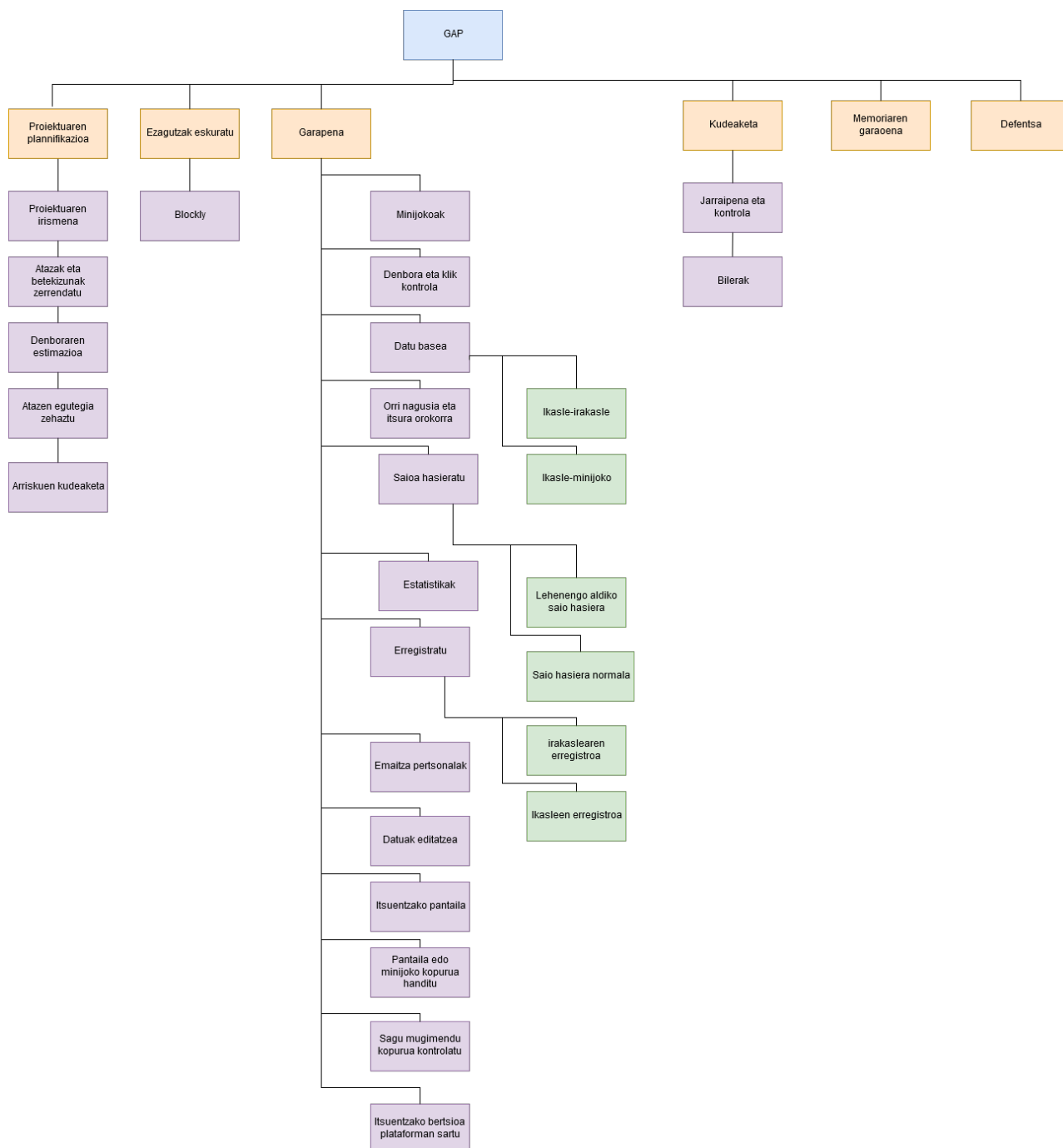
vi. Ikasleak maila pasatzeko behar izan duen denbora.

vii. Ikasleak maila pasatzeko behar izan duen saiakera kopurua.

- 3.7. Izena eman. Irakasleak bere burua erregistratuko du eta ondoren, bere ikasleak erregistratu ahal izango ditu. Beraz, erregistratzeko bi modu egongo dira, ikasleena eta irakasleena.
 - i. Irakasleen erregistroa
 - ii. Ikasleen erregistroa
 - 3.8. Emaila pertsonalak. Ikasleek, beren emaila pertsonalak ikusteko aukera izango dute. Era berean, irakasleek, ikasleen emaila pertsonalak ikusteko aukera ere izango dute. Emaila pertsonal horiek, ikasle zehatz baten emailari egiten die erreferentzia.
 - 3.9. Datuak editatzea. Erabiltzaile orok beren datu pertsonalak aldatzeko aukera izango dute. Datu hauek, erregistratzeko garaian betetako datuak dira.
 - 3.10. Itsuentzako interfaze egokitua. Blockly liburutegiarekin, programa blokeekin "drag and drop"teknika erabiliz, modu bisualean programatzen da. Itsuen kasuan, "drag and drop"teknikak ez du balio, eta beraz beste modu bateko elkarrekintza proposatuko da.
 - 3.11. Pantaila edo minijoko kopurua handitu.
 - 3.12. Saguaren mugimenduen kopurua kontrolatu. Lehenik erabilitako datuez gain, sagu mugimenduak ere kontrolatuko dira, ikaslearen sarrera guztiak kontrolatzeko asmoz.
 - 3.13. Itsuentzako bertsioa plataforman sartu. Lehenago aipatu den itsuentzako interfazea, web aplikazioan sartu.
- 4. Kudeaketa.
 - 4.1. Jarraipena eta kontrola.
 - 4.2. Bilerak.
 - 5. Memoriaren garapena.
 - 6. Defentsa.

2.1.3 LDE diagrama

Atal honetan, aurreko atalean definitutako atazak biltzen dituen LDE diagrama adierazten da.



2.1. Irudia: LDE diagrama

2.2 Denboraren plangintza

2.1.2 atalean deskribatutako atazei eskainiko zaion denboraren estimazioa eta honen arabera egindako planifikazioa azaltzen da atal honetan. Honen helburua, proiektua garatzeko ematen diren 300 orduak modu egokian antolatzea da, proiektuaren irismena betetzeko asmoz.

2.2.1 Atazak eta denbora estimazioa

2.1 taulan, atazen denbora estimazioa ageri da. Bertan ataza nagusiek, hauen lan-paketeen denbora estimazioa bilduko dute, azkenik ataza nagusiaren denbora estimazioa laburbilduz.

2.2.2 Atazen planifikazioa

Lehenik aipatutako ordu kopuruak kontuan hartuta, 2.2 irudian gantt diagrama ageri da. Bertan, zehaztutako datak errespetatuko dira.

2.3 Arriskuen kudeaketa

Atal honetan proiektuan zehar gerta daitezkeen ezustekoak aurreikusi nahi dira, arrisku horiek ekiditeko asmoz.

Esperientzia falta, arrisku garbia da. Izan ere, lehendabiziko aldia da denbora luzerako proiektu bat garatzen dela. Esperientzia falta honek, proiektu bat aurrera emateko antolaketaren ezjakintasuna dakar. Horretarako, zuzendariekin harremanetan mantentzea ezinbestekoa da, haien esperientziaz baliatuz, eta emandako aholkuak jarraituz, epe luzeko proiektu bat aurrera eramaten ikasteko.

Halaber, esperientzia falta horrek ez du proiektu bat aurrera emateko antolaketaren ezjakintasunean bakarrik eragiten. Esperientzia falta horrek, zure burua ondo ez ezagutzea esan nahi du. Argi dago bada, 2.1 taulan ikusitako denbora estimazioa, gutxi gorabeherakoa dela eta ahalik eta zorrotzen betetzen saiatu behar den arren, zenbait desbiderapen egotea saihetsezina dela. Desbiderapen baten aurrean urduri jartzeak eta ondo ez erantzuteak denbora gehiago galtzea eragin dezake. Horregatik, desbiderapenak lasai hartuko dira, lanarekiko jarrera eta dedikazio maila okertu gabe.

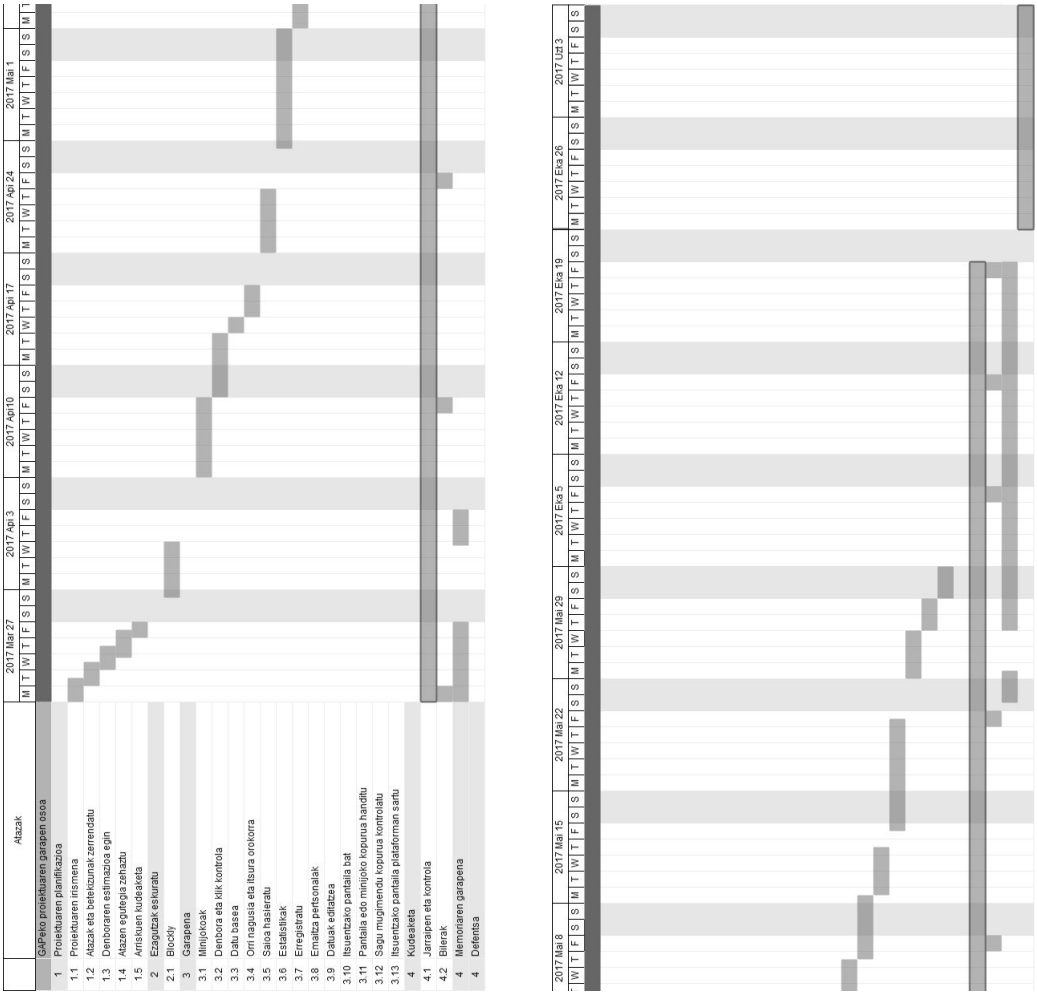
Gure burua eta proiektu bat aurrera eramaten ez jakiteaz gain, proiektuan zehar landuko diren gaien inguruan ere informazio falta zaigu. Plangintza egiteko garaian ezin da jakin nolako zailtasuna edukiko duen proiektuak zehazki, eta horregatik epeak errespetatu eta honen jarraipen eta kontrola egingo da. Atazak garrantziaren arabera burutuko dira eta honela, atzerapenen bat badago eta azkeneko atazak burutu ezin badira, garrantzizko atazak burutuak egongo direla ziurtatzen da.

Nere ingurukoen gradu amaierako lanen arrisku nagusia egindako lana galdu izana da. Kasu honetan, egindako lana galtzea ekiditeko, *gitlab*¹ plataforma erabiliko da, bertsio kontrola egiteko eta momentuko aldaketak gordetzen joateko.

¹Git bertsio kontrolagailuan oinarritutako web zerbitzua.

Ataza	Estimatitako denbora
1. Proiektuaren plangintza	
1.1 Proiektuaren irismena	2h
1.2 Atazak eta betekizunak zerrendatu	5h
1.3 Denboraren estimazioa egin	2h
1.4 Atazen egutegia zehaztu	2h
1.5 Arriskuen kudeaketa	3h
Plangintza guztira	14h
2. Ezagutzak eskuratu	
2.1 Blockly	10h
Ezautzak eskuratu guztira	10h
3. Garapena	
3.1 Minijokoak	15h
3.2 Denbora eta saiakera kopurua	7h
3.3 Datu-basea	
3.3.1 Erabiltzailea	2h
3.3.4 Emaita	2h
3.4 Web plataformako orri nagusia eta itxura orokorra	10h
3.5 Saioa hasieratu	.
3.5.1 Lehenengo aldiko saio hasiera	10h
3.5.2 Saio hasiera normala	3h
3.6 Estatistikak	25h
3.7 Erregistratu	
3.7.1 Irakasleen erregistroa	3h
3.7.2 Ikasleen erregistroa	7h
3.8 Emaita pertsonalak	8h
3.9 Datuak editatzea	5h
3.10 Itsuentzako pantaila	34h
3.11 Pantaila edo minijoko kopurua handitu	12h
3.12 Sagu mugimendu kopurua kontrolatu	6h
3.13 Itsuentzako bertsioa plataforman sartu	15h
Garapena guztira	164h
4. Kudeaketa	
4.1 Jarraipena eta kontrola	8h
4.2 Bilerak	7h
Kudeaketa guztira	15h
5. Memoriaren garapena	77h
6. Defentsa	20h
Guztira	300h

2.1. Taula: Atazak eta denbora estimazioa



2.2. Irudia: Proiektuko gantt diagrama

3. Kapituluia

Marko teorikoa

3.1 Blockly

Atal honetako informazioa, *blockly*ren web orritik atera da [2]. Bloke bidezko programazio editoreak sortzeko JavaScript-eko liburutegia da Blockly. Hezkuntza helburu duten aplikazio askotan erabiltzen da liburutegi hori, code.org barruan besteak beste. Kode blokeak mugituz lortutako kodea, Blockly liburutegiak eskaintzen dituen bost lengoaietara, hots, JavaScript, Python, PHP, Lua eta Dart, itzuli ahal izaten da eta beste lengoaiak gehitzeko aukera ere ematen du.

Lehen aipatu bezala, Blockly JavaScripteko liburutegi bat da. Gainera, ez du zerbitzari baten beharrik eta nabigatzaile gehienekin bateragarria da. Atal honetan, blockly liburutegia nola erabili zehazten da.

3.1.1 Web plataforma eraikitzen

Web aplikaziotan erraz instalatzeko pentsatuta dago Blockly. Lehenik eta behin, github-en dagoen kodea [3] deskargatu behar da. Ondoren, *demos/fixed/index.html* fitxategia nabigatzailean ireki eta blokeak mugigarriak direla konprobatu behar da.

Blockly liburutegia proiektu batean erabili ahal izateko, liburutegi hau erabiliko duen fitxategi orok blocklyk eskaintzen dituen baliabideak inportatu beharko ditu. Besteak beste, blocklyk eskaintzen dituen blokeak, bloke horiek kode bihurtzen dituen scripta eta blokeetako mezuak definitzen dituen hizkuntzaren scripta inportatu beharko dira, beti ere blockly direktorioa non dagoen kontuan hartuz (ikusi 3.1 kodea).

```
1 <script src="../../../blockly/blockly_compressed.js"></script>
2 <script src="../../../blockly/blocks_compressed.js"></script>
3 <script src="../../../blockly/javascript_compressed.js"></script>
```

3.1. Kodea: *Blockly liburutegiko baliabideak kodean gehitu*

Web orri batean, blockly jartzeko modurik errazena *div* etiketa batean txertatzea da. Aplikazioan agertu nahi diren blokeak, *toolbox* batean definitu behar dira, eta ondoren *toolbox* hori *div*-ean txertatu (ikusi 3.2 kodea). Erabili nahi diren blokeak definitzeko, *type* etiketa erabili behar da, definitutako blokeen *type* etiketari erreferentzi egiten diolarik (ikusi 3.3 kodea).

```
1 var workspace = Blockly.inject('blocklyDiv',
2   {toolbox: document.getElementById('toolbox')});
```

3.2. Kodea: *Workspacea definitu, toolboxa div barruan sartu*

```

1 <div id="blocklyDiv" style="height: 480px; width: 600px;"></div>
2 <xml id="toolbox" style="display: none">
3   <block type="controls_if"></block>
4   <block type="controls_repeat_ext"></block>
5   <block type="logic_compare"></block>
6   <block type="math_number"></block>
7   <block type="math_arithmetic"></block>
8   <block type="text"></block>
9   <block type="move_forward"></block>
10 </xml>

```

3.3. Kodea: *Toolboxa definitu, erabili nahi diren blokeekin*

Blokeetik kodea sortzeko, blocklyk bost kode sortzaile eskaintzen ditu, baina, lehen esan bezala, bada sortzaile gehiago gehitzeko aukera. Lengoaia bakoitzari dagokion *workspaceToCode* funtzioa erabiliz, blokeen bidez sortutako kodea eskuragarri izango da. Lortutako kodea exekutatzeko *eval* funtzioa erabiltzen da (ikusi 3.4 kodea).

```

1 var code = Blockly.JavaScript.workspaceToCode(workspace);
2 try {
3   eval(code)
4 } catch (e) {
5   alert(e);
6 }

```

3.4. Kodea: *Sortutako kodea lortu eta exekutatu*

3.1.2 Blokeen hizkuntza

Bloke barruan doan mezua edozein hizkuntzatan ikusteko aukera ematen du blocklyk. Hizkuntza bakoitzari dagokion scripta *blockly/msg/js* direktorioan aurki daiteke, hizkuntzari dagozkien ISO kodeeri erreferentzia eginez. Blockly liburutegiak ez du euskararako itzulpenik.

Direktorio horren barruan dagoen edozein fitxategi hartuz, blocklyk eskaintzen dituen blokeen mezuak ikusi ahal izango dira dagozkien hizkuntzan (ikusi 3.5 kodea). Horrela, hizkuntza berria modu errazean gehitu ahal izango da, beste hizkuntzen eredua jarraitzen badira. Horretarako, mezu horiek nahi den hizkuntzara itzuli behar dira eta aldatutako mezuak gordetzen dituen fitxategia kodean txertatu (ikusi 3.6 kodea).

```

1 Blockly.Msg.ADD_COMMENT = "Anadir comentario";
2 Blockly.Msg.CHANGE_VALUE_TITLE = "Cambiar el valor:";
3 Blockly.Msg.CLEAN_UP = "Limpiar los bloques";
4 Blockly.Msg.COLLAPSE_ALL = "Contraer bloques";
5 Blockly.Msg.COLLAPSE_BLOCK = "Contraer bloque";

```

3.5. Kodea: *Blokeen mezua gaztelaniaz*

```

1 <script src="../blockly/msg/js/es.js"></script>

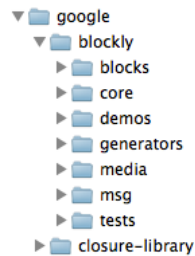
```

3.6. Kodea: *Aukeratutako hizkuntzaren fitxategia gehitu*

3.1.3 Blokeak sortzen

Blocklyk eskaintzen dituen blokeez gain, bloke propioak sortzeko aukera ere ematen du. Bloke horiek eskuz edo modu grafikoan sor daitezke. Blokeak modu grafikoan sortzeko ere, bi aukera daude:

- Internet bidezko editorea: Interneten eskuragarri dagoen blocklyren *Block Factory*¹ demoa erabili daiteke.
- Editore lokala: Blocklyk Interneten dituen demo guztiak eskuragarri daude aurretik jaitsitako Blocky kodean, *blockly/demos* direktorioan zehazki. Gradu Amaierako Lan honetan, aurretik aipatutako *Block Factory* demoa erabiltzen da. Horretarako ireki *blockly/demos/blockfactory/index.html* fitxategia. Fitxategia lehen aldian irekitzean, dependentzia errorea adierazten duen mezua azalduko da. Mezu berean ageri den helbidea jarraituz [4], deskargatu *closure-library* karpeta, eta ziurtatu blockly karpetaren maila berean kokatzen dela.

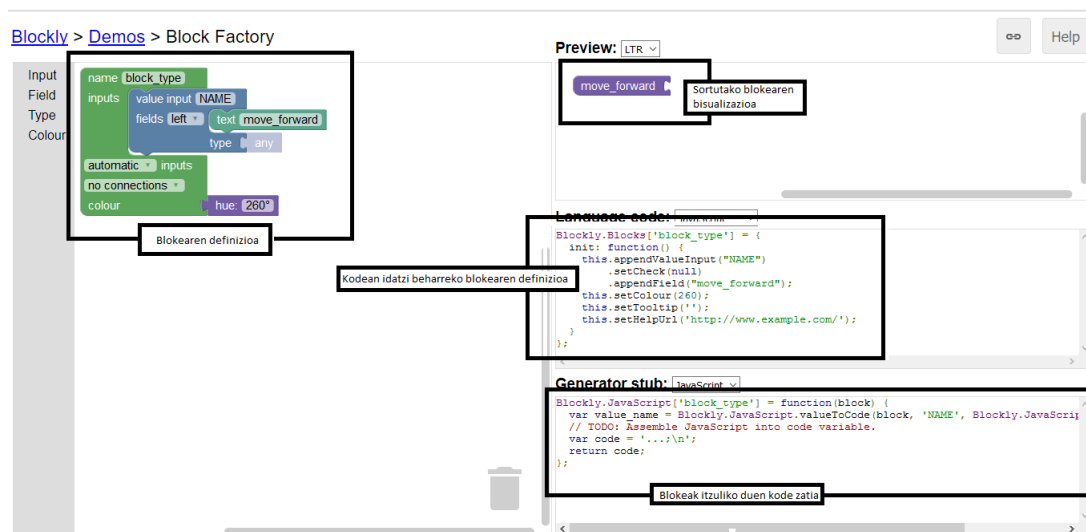
3.1. Irudia: *closure-directory* karpetaren lokazioa

Bertan, lau atal nagusi agertzen dira (ikusi 3.2 irudia). Ezkerraldean, sortu nahi den blokearen definizioa aurkitzen da. Eskuinaldean berriz 3 atal bereizten dira. Lehenengoa, sortutako blokearen bisualizazioa da, hots, *toolbox*ean blokea gehituz ikusaraziko dena.

Horren azpian, bloke honen definizioa dago *json* edo *JavaScript* lengoaian idatzita. Kasu honetan, *JavaScript* lengoaian idatzitako definizioa erabili da. Informazio hori jatorrizko kodean gehitu beharko da, eta informazio hori gordetzen duen scripta aplikazioan inportatu. Blocklyk eskaintzen dituen jatorrizko blokeak *blockly/blocks* barruan dauden *JavaScript* fitxategian aurki daitezke. Direktorio bera erabiliz, fitxategi berri bat sortu behar da, sortuko diren blokeen definizioa gordetzeko.

Azkenik, bloke horrek itzuliko duen kodea egongo da. Plataformak, blocklyk eskaintzen dituen bost lengoaiak eskaintzen ditu baina era berean, beste lengoaietan idazteko aukera ere badago. Blocklyren hierarkia jarraituz, *blockly/tests/generators/generators* karpetan lengoia bakoitzerako scriptak aurki daitezke. Bloke bakoitzak sortuko duen kodea bertan gehitu beharko da, eta jatorrizko kodean script hori inportatu. Blokeak itzuliko duen kodea *code* aldagaian idatzi behar da.

¹https://blockly-demo.appspot.com/static/demos/blockfactory_old/index.html (/2017/06/22 kontsultatuta)

3.2. Irudia: *moveForward* blokearen definizioa

3.2 Blockly games

Googlek blockly liburutegian oinarritutako programatzen ikasteko hezkuntzara bideratutako *Blockly games* joko multzoa du eskuragarri ². Joko multzo honen koda librea da eta githuben [5] aurki daiteke.

Joko multzo hau web aplikazio propioan gehitu ahal izateko, lehenik eta behin koda deskargatu behar da. Koda deskargatu ondoren, ireki sortu berri den *blockly games* karpeta eta kargatu dependentzia guztiak (ikusi 3.7 koda).

```
1 cd blockly-games/
2 make deps
```

3.7. Koda: *Blockly games* dependentziak kargatu

Dependentziak kargatu ondoren, hizkuntza kargatu behar da. Hizkuntzak kargatzeko, hizkuntza guztiak karga daitezke (ikusi 3.8 koda) edo kargatzeko hizkuntz zehatz bat aukera daiteke (ikusi 3.9 koda). Honek joko multzoa konpilatuko du autatutako hizkuntzan.

```
1 make languages
```

3.8. Koda: *Hizkuntz guztiak kargatu*

```
1 make en
```

3.9. Koda: *Ingelesa kargatu*

Blockly games offline bertsioa osoa ere eskuragarri dago ³. Bertsio hau hizkuntza bakarrean deskarga daiteke.

Joko multzo honen orri nagusia karpeta barruan dagoen *index.html* fitxategia da. Honekin batera, joko bakoitzari dagokion karpeta aurki daiteke.

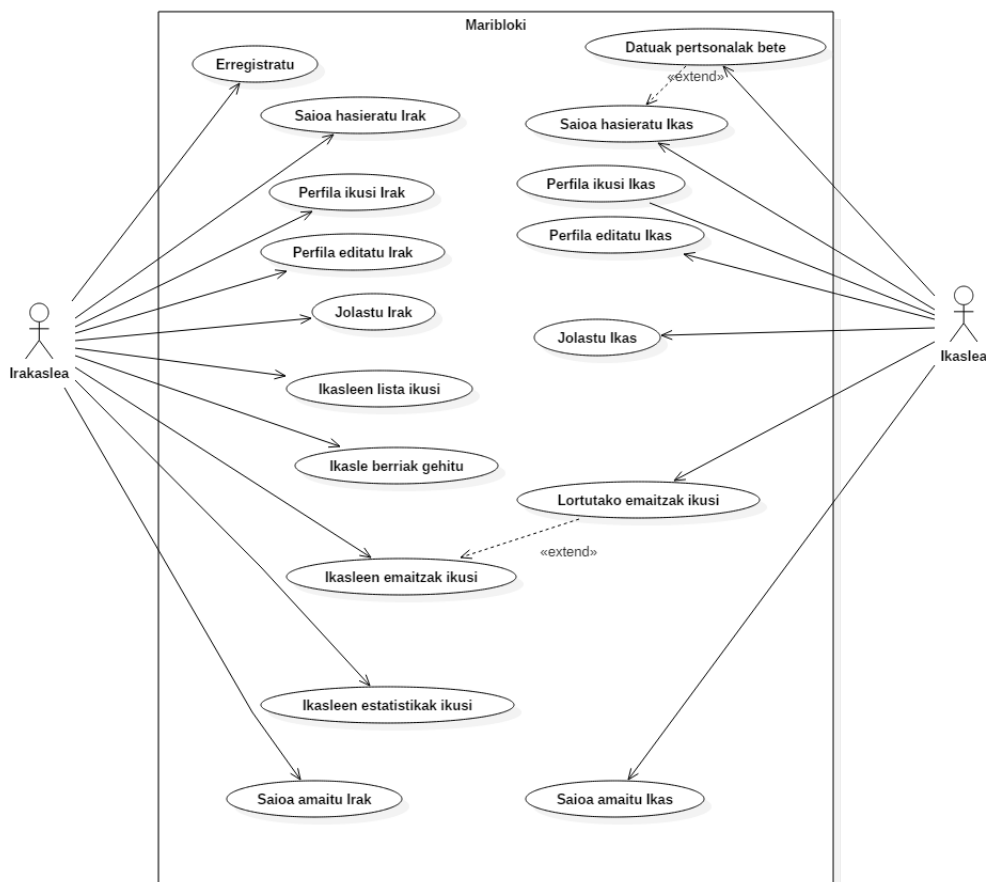
²Joko hauek probatzeko: <https://blockly-games.appspot.com/> (2017/06/22 kontsultatuta)

³<https://github.com/google/blockly-games/wiki/Offline> (2017/06/22 kontsultatuta)

4. Kapitulu

Erabilpen gida

Kapitulu honetan web orriko erabiltzaileek izango dituzten erabilpenak azaltzen dira. Garatutako sisteman bi erabiltzaile mota egongo dira, irakasleak eta ikasleak. 4.1 irudia erabiltzaile hauek izango dituzten erabilpen kasuen diagrama da.



4.1. Irudia: *Erabilpen kasuen diagrama*

4.1 Irakaslea

Irakasleak bere burua erregistratu beharko du, ikasleak erregistratu eta hauen emaitzak ikusi ahal izateko. Irakasleak bere erabiltzailea sortuko du orri nagusiko erregistratu botoia sakatuz (4.2 irudia). Behin eskatzen diren datuak beteta, irakaslearen orri nagusian egongo da irakaslea.

4.2. Irudia: *Irakasle bezala erregistratu*

Irakasleak lau aukera izango ditu menu nagusian: profila, ikasleen lista, ikasleen emaitzak eta minijokoetara jolasteko aukera hurrenez hurren (ikusi 4.3 irudia).

4.3. Irudia: *Irakaslearen menu nagusia*

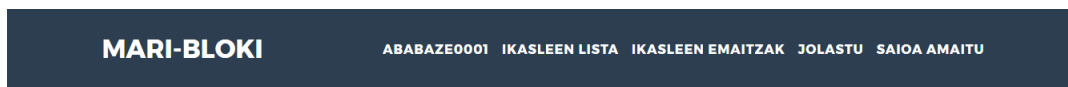
Erabiltzaile izenaren gainean klik eginez, bere profila ikusi eta editatu ahal izango du. Profilean eskuragarri dauden datuak, erregistratzeko garaian emandako datuak dira, hala nola pasahitza, izena, abizena eta ikastetxea.

Ikasleen lista aukeran klik eginez, irakasleak erregistratutako ikasle guztien informazioa ikusi daiteke. Ikasle bakoitzeko, bere izena, abizena, erabiltzailea eta pasahitza ikusiko da. Ikasle bakoitzaren emaitzak eta datuak ikusi ahal izateko, bere erabiltzaile izenaren gainean klik egin beharko da. Irakasleak ezingo du ikasleen daturik aldatu.

Ikasleen lista atalean, ikasleen informazioa ikusteaz gain, ikasle berriak erregistratu eta erregistratutako ikasleak ezaba daitezke.

Erregistratutako ikasle bat datu-basetik ezabatzeko, ikasle honi dagokion lerro ondoan dagoen x gorriari egin klik (ikusi 4.4 irudia).

Taularen gainean dagoen *Ikasleak gehitu* botoian klik eginda, ikasleak gehitzeko orrira birbidaliko du.



Ikasleen erabiltzaile eta pasahitzak

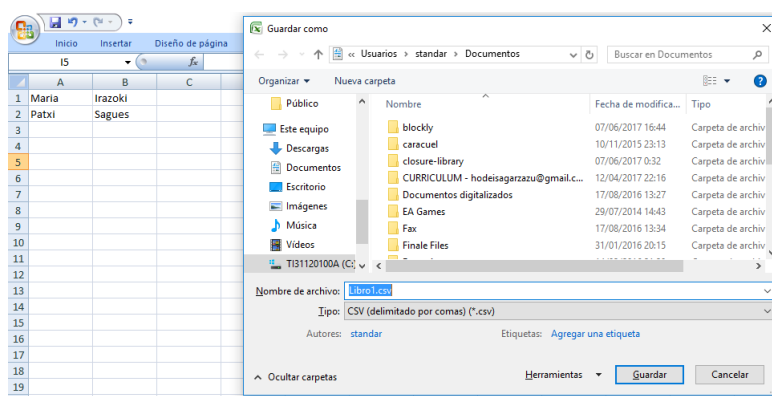
[illegible]

4.4. Irudia: *Ikasleen lista*

Ikasle berriak gehitzeko bi bide daude: *.csv* fitxategi bat kargatuz edo erregistratu nahi den izena eta abizena idatziz.

.csv fitxategi bat sortzeko modurik errazena kalkulu orria erabiltzea da. Ikasleen izenak lehenengo zutabean jarri beharko dira, eta abizenak bigarrenean (ikusi 4.5 irudia). Behin ikasleen izen-abizenak idatzita, kalkulu orriko dokumentua .csv formatuan gorde beharko da. Horretarako, kalkulu orriko menu nagusian aukeratu "Gorde honela" eta aukeratu *CSV (koma bidez banatuta)* formatua (ikusi 4.5 irudia). Lortuko den fitxategian datuak honako itxura izango dute:

Maria; Irazoki
Patxi; Sagues



4.5. Irudia: *.csv* fitxategia sortzeko pausuak

`.csv` fitxategi baten bidez ikasle berriak erregistratzeko, lehenik eta behin *arakatu* (examinar) botoia sakatu eta aukeratu kargatu nahi den fitxategia (4.6 irudiko 1 zenbakia). Fitxategia aukeratu ondoren, klikatu *kargatu* botoia (4.6 irudiko 2 zenbakia) eta aukeratutako `.csv` fitxategiko datuak taulan kargatuko dira. Taula editagarria da, eta beraz, `.csv` fitxategitik taulan kargatutako datuak edita daitezke (4.6 irudiko 3 zenbakia).

Azkenik, taulan dauden datuak datu-basean onartzeko, *gorde* botoia sakatu beharko da (4.6 irudiko 4 zenbakia). Irakasleak, erregistratutako ikasleen erabiltzaile eta pasahitzak jasoko ditu pdf orri batean.

MARI-BLOKI ABABAZE0001 IKASLEEN LISTA IKASLEEN EMAITZAK JOLASTU SAIOA AMAITU

Examinar... No se ha seleccionado ningún archivo. **kargatu**

Ikasleen erabiltzaile eta pasahitzak

Izena	Abizena
Sergio	Iriarte
Ane	Eceiza

Gorde **Ikasle berria**

4.6. Irudia: *Ikasle berriak gehitu .csv fitxategi baten bidez*

.csv fitxategia erabili gabe ere ikasleak erregistra daitezke. Horretarako, sakatu *Ikasle berria* botoia (4.7 irudiko 1 zenbakia) eta lerro berri bat azalduko da taulan. Bertan, idatzi erregistratu nahi den ikaslearen izen eta abizena (4.7 irudiko 2 zenbakia). Aurreko kasuan bezala, ikaslea erregistratzeko, *Gorde* botoia sakatu beharko da (4.7 irudiko 3 zenbakia), eta ikasleen erabiltzaile eta pasahitzak biltzen dituen pdf fitxategi bat jasoko du irakasleak.

MARI-BLOKI ABABAZE0001 IKASLEEN LISTA IKASLEEN EMAITZAK JOLASTU SAIOA AMAITU

Examinar... No se ha seleccionado ningún archivo. **kargatu**

Ikasleen erabiltzaile eta pasahitzak

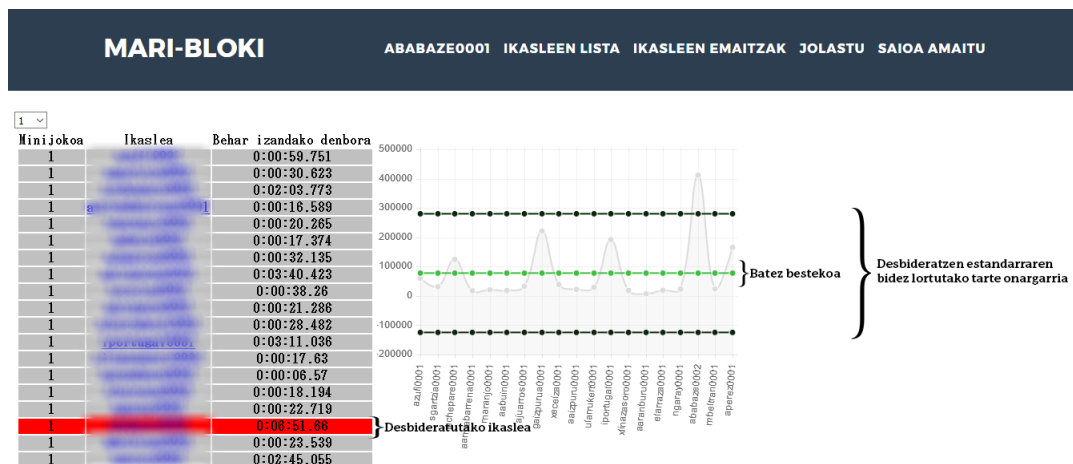
Izena	Abizena
Sergio	Iriarte
Ane	Eceiza
Idatziz IZENA	Idatziz ABIZENA

Gorde **Ikasle berria**

4.7. Irudia: *Ikasle berriak gehitu izena eta abizena idatziz.*

Menu nagusiko *ikasleen emaitzak* sakatuz, minijoko bakoitzean ikasleek izandako emaitzak ikusi ahal izango dira. Emaitza horiek, ikasleek minijokoa gainditzeko behar izandako denbora eta saiakera kopurua izango dira, bi taula ezberdinetan. Tauletan, kolore gorriak dauden ikasleak, bere ikaskideekin konparatuz, gaizki dijoazten ikasleak izango dira, eta berdez daudenak bereziki ondo dijoaztenak.

Taula bakoitzaren ondoan, grafiko bat dago. Grafiko hauek ondoko taulan dauden datuak errepresentatzen dituzte. Grafikoan dagoen lerro berdeak, ikasleen batazbestekoa adierazten du, eta lerro beltzek batez besteko honi desbideratze estandarra aplikatuz lortutako tartea adierazten dute (ikusi 4.8 irudia).

4.8. Irudia: *Ikasleen emaitzen taula eta grafikoak.*

Azkenik, *Jolastu* atalean, ikasleek eskuragarri dituzten joko berak daude.

4.2 Ikaslea

Ikasleak ezin du bere burua erregistratu, bere irakaslearengandik jaso beharko du erabiltzaile eta pasahitza. Saioa lehenengo aldiz hasten duenean, bere profileko datuak betetzea eskatuko zaio.

Ikasleak, bere perfila ikusi eta editatu ahaliko du, bere erabiltzaile izenaren gainean klik eginez. Minijokoetan jolasteko, egin klik *jolastu* atalean. Menu nagusiko *lortutako emaitzak* atalean klik eginez, minijoko bakoitzeko lortutako emaitzak ikusiko dira (4.9 irudia).

4.9. Irudia: *Ikasleen menu nagusia.*

4.3 Kudeatzailea

Erabilpen gida hau, web aplikazioa kudeatuko duen pertsonarentzat da. Web aplikazioa *hostinger* zerbitzuan dagoen arren, doako *hostinger* zerbitzua erabili da. Doako zerbitzuak ez du behin-behitiko zerbitzari bat eskaintzen, beraz web aplikazioa erabili nahi bada, beste zerbitzari batera igo beharko da.

Web aplikazioko kodea github plataforman aurkitzen da ¹. Web aplikaziorako behar den datu-basea, kodearekin batera aurkitzen da. Datu-base hau ere zerbitzari batera igo beharko da. Datu-basearekin konektatzen den fitxategi orok datu-base honen datuak beharko ditu. Datu-baserako konexioa behar duen fitxategi orok 4.1 kodean agertzen diren datuak eduki beharko ditu, datu-basearen zerbitzariaren helbidea, datu-baseko erabiltzailea, datu-baserako pasahitza eta datu-basearen izena hurrenez hurren.

¹<https://github.com/aepzia/maribloki.git>

```
1 $host=" "; // Host name
2 $username=" "; // Mysql username
3 $password=" "; // Mysql password
4 $db_name=" "; // Database name
```

4.1. Kodea: *Datu-basera konektatzeko beharrezko datuak*

Hauek dira datu-baserako konexioa egiten duten fitxategiak: *saioa_hasieratu.php*, *ikaslea_emaizaLista.php*, *ikaslea_perfila.php*, *irakasle_ikasleLista.php*, *irakaslea_ikasleEstatistikak.php*, *irakaslea_perfila.php*, *ikaslea_datuakGorde.php*, *ikaslea_gordeMinijoko.php*, *irakasle_gehituIkasleakDB.php*, *irakasle_gordeDatuak.php* eta *irakasle_ikasleaEzabatu.php*.

5. Kapituluia

Erabilitako teknologia

Kapitulu honetan, aurretik aipatutako liburutegiez gain proiektuan zehar erabilitako teknologia ezberdinak azaltzen dira.

5.1 MySQL

Proiektu honetan MySQL datu basea erabili da. MySQL datu-basea erabiltzeko gestio sistema bat da eta software askeko datu base motarik ezagunenatzat hartzen da da, batez ere web garapenerako. [6] erreferentzian MySQL-ri buruzko informazio gehiago aurkitzen da.

MySQL datu-base mota erabili da *phpMyAdmin* zerbitzuaren bitartez erabiltzeko erraza delako eta web aplikaziotarako honen erabilpena lehendik ere ezagutzen zelako.

5.2 Php eta html

Web aplikazio bat izanik, proiektua html eta php lengoaia erabiliz garatu da. Web aplikazioko gorputz nagusia *html* markaketa lengoaiari idatzi da eta datu basearekin konektatzeko edo eta orri dinamikoa egiteko *php* lengoaia erabili da.

5.3 Java script

html eta *php* lengoaiak erabiltzeaz gain, *Java Script* lengoaia erabili da zenbait eragiketetarako. Aurretik aipatutako blockly liburutegiaz gain, hauek dira erabili diren *Java Script* lengoaiarako liburutegiak:

5.3.1 chart.js eta canvas.js

Bi liburutegi hauek, *Java Script* bidez grafikoak egiteko erabili dira. Honen bidez, edozein grafiko motak garatu daitezke.

Hasiera batean *chart.js*[7] liburutegia erabili da grafikoak egiteko, *javaScript* lengoaiaren bidez grafikoak egiteko zenbait adibidetan oinarrituta.

Liburutegi hau berria da eta grafiko mota guztiak ez daude erabilgarri. *Chart.js* liburutegiak *x* ardatzean dauden datuak banaka tratatzen ditu eta ez dira balioen arabera tratatzen. Azkeneko grafikoa egiteko, ikasleen denbora eta saiakera kopurua grafiko batean

irudikatzeke, *canvas.js* [8] liburutegia erabili da, mota honetako grafikoak irudikatzeke adibideetan oinarrituz.

5.3.2 jsPDF

jsPDF [9] liburutegia *pdf* fitxategiak sortzeko *Java Script*eko liburutegia da. Liburutegi hau *JavaScript* lengoairaren bidez *pdf* fitxategiak sortzeko adibideetan oinarrituz aukeratu da.

5.3.3 AJAX

Datu-basearekin konexioa *php* bidez zuzenean egin beharrean, AJAX metodoa [10] erabili da, *Java Script* funtzio batetik datu-basea tratatzeko.

Blockly games minijokoa erabiltzerako, honen *JavaScript* fitxategia aldatu da. Fitxategi honen gainean, ikasleak behar izandako denbora eta saiakera kopurua kalkulatu da. *JavaScript* fitxategian lortzen diren datuak datu-basean gordetzeko *AJAX* metodoa erabili da. Metodo honen bidez, *JavaScript* fitxategiak datu-basean gorde nahi diren datuak pasako dizkio *php* fitxategi bati eta *php* fitxategiak datu hauek datu-basean gordeko ditu. *Php* fitxategiak datuak datu-basean gorde dituenean, metodo bera erabiliz erantzun bat bidaliko dio *JavaScript* fitxategiari, erantzun hori tratatuko duelarik.

5.4 Hostinger

Web aplikazio hau zabaltzeko eta emaitza ahalik eta errealean lortzeko, hostinger zerbitzaria erabili da. Hostinger, doako hosting zerbitzua eskaintzen duen web zerbitzua da.

Hostingerren doako zerbitzuaren helburua, web garapenean hasiberriak direnentzako doako hosting zerbitzua eskaintzea da. Zerbitzu honek urte beterako doako zerbitzua eskaintzen du eta mugimendu askoko edo oso gutxiko web aplikazioak itxi egiten ditu. Proiektu honetan hostinger zerbitzua erabili da, probako laginak aplikazioa eskuragarri izan dezan.

5.5 Gitlab eta github

Gitlab, git bertsio kontrolerako biltegi bat da. Proiektu honetan gitlab plataformako biltegitratze pribatu bat erabili da. Gainera, memoria eta kodea bi adar ezberdinetan jardun dira.

Proiektu honetan *gitlab* biltegia erabili da, proiektuko aldaketak momentu oro gordetzeko eta edozein momentuan eskuragarri izateko (2.3 atala). *Git* bertsio kontrolak, proiektuko aldaketak atzera botatzeko prozesua erraztu egiten du eta informazio galerak ekiditen ditu. Proiektuan zehar *gitlab*eko biltegitratze pribatua erabili den arren, proiektuko kodearen azkeneko bertsioaren kodea *github*eko biltegitratze publiko batera igo da, edonork eskuragarri izateko eta ekarpenak jasotzeko.

Github [11] zerbitzua, aurretik aipatutako *gitlab* biltegitratze zerbitzuak bezala funtzionatzen du, baina kasu honetan biltegitratze publikoa da. *Github* web plataforman, bakoitzaren proiektuak eta kontribuzioak gordetzen dituen profila erabilgarri dago

edonorentzat. Zerbitzu hau kode irekiko komunitatean kodea elkarbanatzeko eta ekarpenak egiteko erabilgarria izateaz gain, proiektu pertsonalen bitartez mundu profesionalerako bakoitzaren gaitasunak erakusteko modu bat da.

5.6 Texmaker

Memoria garatzeko, texmaker erabili da. Texmaker, LaTeXeko testu editore bat da. LaTeX *gitlab* plataformarako testu fitxategi bateragarria izateaz gain, kalitate tipografiko handia eskaintzen duelako aukeratu da. Tesiak, artikulak eta liburu teknikoak idazteko erabiltzen da.

6. Kapitula

Proiektuaren garapena

Kapitulu honetan, 2.1.2 atalean definitutako atazen garapena nolakoa izan den azaltzen da.

6.1 Minijokoak

Proiektu honetan labirinto motatako jokoak landu dira. Proiektuan erabiltzen diren minijokoak hasiera batean zerotik sortzea pentsatu zen, googleko *blockly* liburutegia erabiliz (3.1 atala).

Hasierako bertsioan, *blockly* liburutegiak dituen blokeez gain (baldintzazko blokeak, begizta blokeak...), beste bloke batzuk sortu ziren. Zehazki, *move_forward* (mugitu aurrera), *turn_left* (biratu ezkerrean) eta *turn_right* (biratu eskubira). 3.1.3 atalean azaltzen den moduan, bloke berriak sortzerakoan, bloke honen itxura, eta itzuli beharreko kodea adierazi behar da.

Adibide moduan ikusi 6.1 kodea eta 6.2 kodea. Alde batetik, 6.1 kodean, *move_forward* blokearen itxura definitzen da, besteak beste, blokeak izango duen kolorea, testua... Bestalde, blokeak itzuliko duen kodea 6.2 kodean ikus daitekeen modura, *move_forward()* funtzioari dei egitea da. Kasu honetan, dei egiten zaion *move_forward* funtzioa *JavaScript* lengoaiari definituta egongo da.

```
1 Blockly.Blocks['move_forward'] = {  
2   init: function() {  
3     this.appendDummyInput()  
4       .setAlign(Blockly.ALIGN_CENTRE)  
5       .appendField("move_forward");  
6     this.setPreviousStatement(true, null);  
7     this.setNextStatement(true, null);  
8     this.setColour(290);  
9     this.setTooltip('');  
10    this.setHelpUrl('');  
11  }  
12 };
```

6.1. Kodea: *move_forward* blokearen itxura definitzeko kodea

```
1 Blockly.JavaScript['move_forward'] = function(block) {  
2   var code = 'move_forward();\n';  
3   return code;  
4 };
```

6.2. Kodea: *move_forward* blokeak itzuli beharreko kodea

Kontuan izanik joko hauen erabiltzaileak umeak izango direla, blokeak egiten dutena modu dinamiko batean adierazteko saiakera egin zen. Hemen sortu zen ataza honetako arazoa. Blokeen animazioak egiteko CSS3 Animations [12] teknologia erabili zen, elementu baten estiloa gradualki aldatzeko aukera ematen duena. Kasu honetan, elementuaren posizioa gradualki aldatzen zen. Sortutako arazoa zehazki bloke anitz segidan exekutatzean gertatzen zen. Lehenengo animazioa amaitu baino lehen, hau da, objektuaren posizioa eguneratu baino lehen, bigarren blokearen animazioa eta posizio aldaketa exekutatzen ziren. Ondorioz, bigarren posizio aldaketa, hasierako posiziotik egiten zen, lehenengo blokea kontuan hartu gabe.

Esaterako, bi *move_forward* bloke jarraian jarritz, *move_forward()* funtzioa bi aldiz exekutatuko litzateke. Funtzio hori lehenengo aldiz exekutatzerakoan, posizioa aldatzeko animazioa aktibatuko luke, baina animazio hau bukatu aurretik, *move_forward* funtzioa bigarren aldiz deituzerakoan animazioa aktibatuko da berriz, aurretik aktibatutako animazioa deuseztatuz. Horregatik, *move_forward* funtzioa askotan deituta ere, animazio bakarrak funtzionatuko luke.

Arazo hauek direla eta, azkenean *blockly games* minijokoak erabili dira web orrian (3.2 atala). Hasierako helburua labirinto motatako jokia egitea zenez, *blockly games* joko multzotik, *maze*¹ jokia aukeratu zen.

Kasu honetan *blockly games* joko multzoaren offlineko gaztelerazko bertsioa deskargatu da (ikusi 3.2 atala). Honekin, *blockly games* joko multzoko joko guztiak izango dira eskuragarri. *Maze* jokia, hizkuntzari dagokion karpeta barruko *maze.html* fitxategia da (*es/maze/maze.html*).

Blockly games jokoen dependentziak direla eta, garrantzitsua da fitxategiaren izena errespetatzea. Horretaz gain, *common*, *third-party* karpetak eta erabili nahi den jokoaren izena daraman karpeta ere mantendu behar dira, kasu honetan *maze* karpeta.

6.2 Denbora eta saiakera kopurua

Web plataforma honen bidez, irakasleak bere ikasleen arteko gaitasun mailen berezitasunak ikusteko aukera du. Gaitasun maila neurtzeko, ikasleek minijoko bakoitza gaingitzeko behar izan duten denbora eta saiakera kopurua kalkulatu eta gordetzen da.

Behar izandako denbora, minijokoa kargatu denetik, minijokoa ontzat eman arte igarotako denbora izango da. Behar izandako saiakera kopurua berriz, *exekutatu* botoia minijokoa ontzat eman arte klikatua izandako kopurua izango da.

6.1 atalean esan den bezala, *maze.html* fitxategia erabili da eta beraz, behar izandako denbora eta saiakera kopuruaren kontrola egiteko, *maze.html* fitxategia aztertu da. *maze.html* fitxategiak *common/boot.js* scriptari deitzen dio eta honek aplikazioak duen izen berdineko fitxategi barruan eta hizkuntzari dagokion fitxategi barruan dagoen *compressed.js* fitxategia kargatzen du (*maze/generated/es/compressed.js*).

Fitxategi hori automatikoki sortua dirudien arren (ikusi 6.1 irudia), fitxategiaren jatorria ezezaguna da. Minijokoaren gaineko edozein aldaketa egiteko, *compressed.js* fitxategia aldatu da.

¹Maze jokia probatzeko esteka: <https://blockly-games.appspot.com/maze> (2017/06/22 kontsultatuta)

```

4
5 var g,k=this;function aa(){function ba(a){a.Hb=function(){return a.Qg?a.Qg:a.Qg=new a}}
6 function ca(a){var b=typeof a;if("object"==b){if(a){if(a instanceof Array)return"array";if(a i
7 * a.propertyIsEnumerable&&!a.propertyIsEnumerable("call"))return"function"}else return"null";
8 else if("function"==b&&"undefined"==typeof a.call)return"object";return b}function da(a){retu
9 function la(a,b,c){return a.call.apply(a.bind,arguments)}function ma(a,b,c){if(!a)throw Error
10 function oa(a,b){var c=Array.prototype.slice.call(arguments,1);return function(){var b=c.slic
11 * pa(a,b){null!=a&&this.Ya.apply(this,arguments)}g=pa.prototype;g.xc="";g.set=function(a){this.
12 * a.replace(/^[^\s\x00]+|[\s\x00]+$/g,"");function ta(a,b){var c=String(a).toLowerCase(),d=Stri
13 function ua(a){if(!va.test(a))return a;-1=a.indexOf("&")&&(a=a.replace(wa,"&")); -1=a.in
14 function Da(a){var b={"&":"&","<":"<",">":">",""":"'"},c=k.document.createEl
15 function Ea(a){return a.replace(/&([^\s\x00]+|[\s\x00]+)$/g,function(a,c){switch(c){case "amp":return"&";cas

```

6.1. Irudia: *compressed.js* fitxategiaren kode zatia

Bloke bidez programatutakoa abiarazten duen botoiak *runButton* identifikadorea du. Botoi horri egin zaion klik kopuruak adieraziko du zenbat aldiz exekutatu den bloke bidez programatutakoa, eta ondorioz, minijokoa asmatzeko egindako saiakera kopurua. Botoi horri *onclick* gertaera gehitu zaio, honen gainean egin den klik kopurua kalkulatzeko.

Beste aldetik, minijokoa betetzeko behar izandako denbora kontrolatzeko, hasierako momentutik minijokoa asmatu arte igarotako denbora kalkulatu behar da. Minijokoa asmatu denean, *dialogDone* identifikadorea duen mezua agertuko da. Minijokoa kargatzerakoan hasierako ordua gordeko da eta, mezu hau agertzen denean, momentuko ordua eta hasierako orduaren arteko diferentzia kalkulatu da, ikasleak behar izandako denbora milisegundotan lortuko delarik. Era berean, *ajax* bidezko dei baten bidez ikasleak behar izandako denbora eta saiakera kopurua datu-basean gordeko dira (ikusi 6.3 kodea).

```

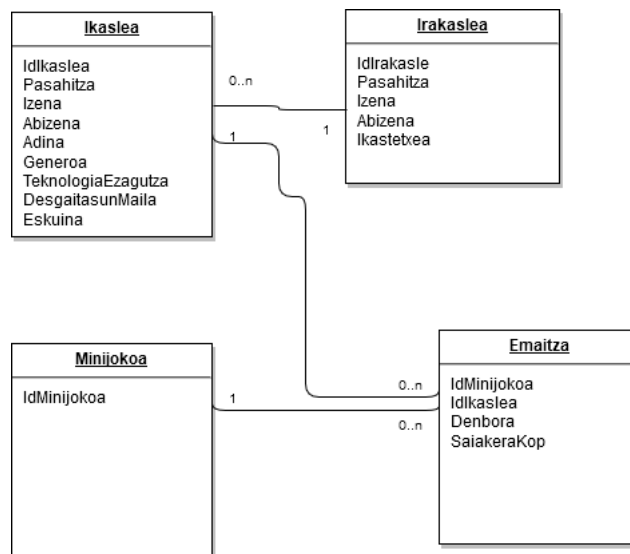
1 function onartua(){
2   end = new Date();
3   if(!updated){
4     diff = end - start;
5     updated=true;
6     var parametros={
7       "idMinijokoa" : F,
8       "saiakeraKop" : klikkop ,
9       "denbora" : diff
10    };
11    $.ajax({
12      data:   parametros ,
13      url:    "server/ikaslea_gordeMinijoko.php" ,
14      type:   'post' ,
15      success: function (response) {
16      }
17    });
18  }
19 }

```

6.3. Kodea: *Denbora eta saiakera kopurua datu-basean gordetzeko ajax deia*

6.3 Datu-basea

Web aplikazio honetan, lau motatako objektuak bereizten dira: minijokoa, ikaslea, irakaslea eta emaitza. Objektu horiek, zenbait atributu izango dituzte. 6.2 irudian taula hauen klase diagrama erakusten da.

6.2. Irudia: *Klase diagrama*

Irakaslea izango da web aplikazio honen gailurreko erabiltzailea. Horregatik, bere informazioa gordetzea ez da bereziki garrantzitsua eta oinarritzko datuak bakarrik izango ditu atributu, hala nola, pasahitza, identifikadoretzat erabiliko den erabiltzailea izena, izena, abizena eta ikastetxea. Erabiltzailea izango da objektu honen identifikadorea. Erabiltzaileak <iniziala><abizena>XXXX itxura izango du eta izena eta abizena emanda, modu automatikoan sortuko da (ikusi 6.7 atala).

Ikasle eta minijokoen datu ahalik eta zehatzenak gordetzea ezinbestekoa da, helburua hauen datuak aztertzea baita.

Ikaslearen oinarritzko perfila eraikitzeaz gain, emaitzak aztertzeko esanguratsuak uste diren datuak biltzen ditu objektu honek. Datu horiek, ikasleen arteko gaitasunetan eragin dezaketen datuak dira: adina, generoa, teknologia ezagutza, ezgaitasun maila eta eskuina edo ezkerria den. Ikaslearen identifikadorea bere erabiltzaile izena izango da. Ikasle bakoitzak bere irakaslearen erabiltzaile izena gordeko du, irakasle eta ikasle erlazioa markatuko duelarik.

Minijokoei dagokienez, identifikadorea bakarrik izango du atributu, eta horregatik ez da datu-baseko minijoko taula sortu. Dena den, aurrerago honen zailtasun maila eta deskripzio bat gehitzea planteatu da. Kasu honetan, minijoko taula sortuko litzateke.

Azkenik, ikasleak lortutako emaitza gordetzen duen *emaitza* taula dago, minijokoa eta ikaslea identifikadoreaz gain, ikasleak behar izandako denbora eta saiakera kopurua gordeko dituen.

Proiektu honetan mysql datu-base mota erabili da. 6.2 irudian dauden klaseak oinarritzat hartuz eraiki da datu-basea.

Datu basetik datuak lortu, idatzi, eguneratu edo ezabatu nahi den bakoitzean, datu-basearekin konexioa egin beharko da. Konexioa sortuta dagoela, *sql* adierazpenen bidez datu-baseari aplikatu nahi zaion eragiketa definituko da. Datu base honen azkeneko bertsioa ez da plangintzan (2.1.2) definitutako, bi erabiltzaile motak bi tauletan banatzea erabaki baita. Aldaketa honen arrazoiak, bi erabiltzaile motek atributu ezberdinak dituztela da.

6.4 Orri nagusia eta itxura orokorra

Web orriaren orri nagusia eta itxura orokorraren garapena sarean aurkitutako boostrapeko txantiloiei batean oinarrituta eraman da aurrera. Boostrapeko txantiloiei hori, bere itxura sinpleagatik aukeratu da. Txantiloia erabilia, hiru atal zehaztu dira orri nagusian. Hasteko, web orriaren Ikonoa eta "lema" azaltzen dira. Bigarren atalean web orriaren saioa hasteko eta erregistratzeko aukera ematen du; zehaztapenak web orriaren erabilpen gidan azaltzen dira (4 kapitulua). Hirugarren atalean web orriaren informazioa eta funtzioa azaltzen duen testua azaltzen da.

Honetaz gain, uneoro webeko orri nagusiaren goiko zatian menu bat agertuko da, "saioa hasi" eta "orriaren informazioa" ataletara lasterbideak dituenak.

Web orriaren saioa hasi eta gero, webeko goiko zatian agertzen den menua mantendu egiten da, baina erabiltzailearen arabera, ikasle nahiz irakasle, menuan agertzen diren aukerak aldatu egingo dira. Aukera hauek erabilpen gidan zehaztutakoak dira (4 kapitulua). Hauek sakatzerakoan web orriaren gorputza aldatzen da, beti ere menua mantenduz.

6.5 Saioa hasieratu

Irakasleek eta ikasleek era berean egingo dute saio hasiera. Orri nagusitik jasotako erabiltzaile izena eta pasahitza *saioa_hasieratu.php* orriaren jasoko dira. Bertan, datu hauek *irakaslea* eta *ikaslea* taulan egiaztatuko dira, eta dagokion erabiltzaile motaren orri nagusira bidaliko du.

Ikasleak lehenengo aldiz saioa hasten duenean, ikasleari dagokion orri nagusiak bere profileko datuak betetzea eskatuko dio. Eskara *ikaslea_datuEskaria.html* orriaren egin da eta datu guztiak ondo bete direla egiaztatu ondoren *ikaslea_datuakGorde.php* orriaren datu-basean gordetzen dira datuak.

6.6 Estatistikak

Proiektu honen helburua, ariketa multzo baten bidez lortutako ikasleen emaitzak aztertzea da. Atal honetan datuen banaketa nolakoa den aztertu eta neurtzeko erabilitako metodo estatistikoak azaltzen dira. Atal honetako informazio teorikoa *Estatistika Aplikatua: Teoria eta Praktika* liburutik atera da [13]. Metodo estatistikoek aztertzen ari garen elementuen arteko berezitasunak zenbaterainokoak diren ikusten edota aztertzen laguntzen dute.

Aldagai baten puntuazioen aldakortasuna neurtzeko modu erabilgarri bat da hurrenez hurreneko balioetatik zutoin zentral baterainoko distantziak neurtzea; zutoin zentral gisa oro har, batez bestekoa erabiltzen da.

6.6.1 Bariantza

Zutoi zentralarekiko distantziak kalkulatzeko badira, honako emaitza hauek lortuko dira:

DISTANTZIAK:

A: -2,-1,0,1,2

Distantzien batura = 0

B: -4,-2,0,2,4

Distantzien batura = 0

C: -10,-5,0,5,10

Distantzien batura = 0

Distantzien arteko batura kasu guztietan 0 dela ikus daiteke. Arazo horren irtenbidea **bariantzak** ematen du, distantzia bakoitzaren karratua kalkulatzen da, hala zeinua kenduz, eta karratu horien batez bestekoa kalkulatzen da.

DISTANTZIAREN KARRATUAK eta BATURA:

A: 4,1,0,1,4

Distantzien batura = 10

B: 16,4,0,4,16

Distantzien batura = 40

C: 100,25,0,25,100

Distantzien batura = 250

Eta hortik bariantzak, S deitutakoak, lortuko dira.

BARIANTZA:

A: Bariantza = $10/5 = 2$ B: Bariantza = $40/5 = 8$ C: Bariantza = $250/5 = 50$

Zenbat eta sakabanatuago egon datuak, hainbat eta handiagoa izango dute bariantza. Bariantza kalkulatzeke, hau da, adibidean eman diren pausoak laburtuko dituen formula gisa, bariantzaren formula hau erabiliko da:

$$s_n^2 = \frac{1}{n} \sum_{i=1}^n (x_i - \bar{x})^2$$

n datu kopurua, \bar{x} batez bestekoa eta x_1, \dots, x_n datuak izanik.

Definizio bat ematearren, esan daiteke datu-multzo baten bariantza dela puntuazio bakoitzetik hasi eta balio horien batez bestekoraino dauden distantzien karratuen batez bestekoa. Beraz, bariantza da distantzien karratuen batez bestekoa. Distantzien karratuekin lan egiteak zaildu egiten du bariantzaren balioa interpretatzea. Arazotxo hori konpontzen duen banaketa estatistikoa desbideratze estandarra da.

6.6.2 Desbideratze estandarra

Karratura jasoak dauden distantziekin lan egiteak sor lezakeen nahasteak deuseztatzeke, desbideratze estandarra erabiliko da.

Probabilitate teorian eta estatistikan, desbideratze estandarra aldagai kuantitatibo bati buruzko datu-multzoen eta probabilitate-banakuntzen sakabanatze neurri absolutu bat da. Jatorrian eta datu-multzo baterako, datu bakoitza batez besteko aritmetiko sinplearen adierazgarritasun neurria da.

Datu multzoen sakabanatze-mailak alderatzeko erabili behar denean, dagokion sakabanatze neurri erlatiboa hobesten da, aldakortasun koefizientea hain zuzen, desbideratze estandarra zati batez bestekoa eginez kalkulatzen dena.

Desbideratze estandarra, bariantzatik lortzen da zuzenean, haren erro karratua kalkulatuz:

$$s_n = \sqrt{s_n^2}$$

6.6.3 Batez besteko eta desbideratze estandarraren erabilpen batura

Desbideratze estandarra batez bestekoarekiko sakabanapen neurria izanik, datuak batez bestekoarekiko zenbateraino sakabanaturik dauden esaten digu. Hortik arrunta da $(\bar{x} - k \cdot s_n, \bar{x} + k \cdot s_n)$ moduko tartea eraikitzea. Tarte horien interpretazioa ondorengoa da: $(\bar{x} - k \cdot s_n, \bar{x} + k \cdot s_n)$ tartean gutxienez laginaren $\%(1 - 1/k^2) \times 100$ barneratzen da ($k \geq 2$).

6.6.4 Garapena

irakaslea_ikasteEstatistikak.php fitxategian, desbideratze estandarraren aplikazioa egin da. Lehenengo lerroetan, datu-basearekin egiten da eskaera eta *sql* kontsulta erabiliz, *emaitza* taulako datuak lortuko dira. Datu hauek, irakasleak erregistratutako ikasleen erabiltzaile izenek eta aukeratutako minijokoaren bat egiten duten emaitzak izango dira (ikusi 6.4 kodea).

```

1 $link = mysqli_connect("$host", "$username", "$password", "$db_name");
2
3 session_start();
4 $erab=$_SESSION['usr'];
5 $content="";
6 $emaitzaKop=0;
7 $emaitzaDen=0;
8 $minijokoa=$_GET['minijokoa'];
9
10 $sql="SELECT * FROM irakaslea INNER JOIN emaitza on irakaslea.idIkaslea
11      =emaitza.idIkaslea WHERE idirakasle='$erab' AND idMinijokoa=minijokoa";
12 $emal=mysqli_query($link,$sql);
```

6.4. Kodea: *Irakasleak erregistratutako ikasleen eta aukeratutako minijokoan lortutako datuak jaso*

Emaitza hauei, 6.6.2 eta 6.6.3 ataletan azaldutako formulak aplikatu zaie ikasleen emaitzen tarte onargarria kalkulatzeko. Kasu honetan, $k = 2$ izango da, tarte honetan gutxienez laginaren $\%75^2$ barneratzen delarik (ikusi 6.5 kodea).

```

1 while($emarow = mysqli_fetch_array($emal))
2 {
3     $emaitzaKop++;
4     $emaitzaDen+= (int) $emarow['denbora'];
5 }
6 $batezbestekoa1=$emaitzaDen/$emaitzaKop;
7 $batukari=0;
8 mysqli_data_seek($emal, 0);
9 while($emarow = mysqli_fetch_array($emal))
10 {
11     $batukari+= ($emarow['denbora']-$batezbestekoa1)**2;
12 }
13 $bariantza1=$batukari/$emaitzaKop;
14 $desbideratzeEstandarra=sqrt($bariantza1);
15 $arteaMin1=$batezbestekoa1-2*$desbideratzeEstandarra;
16 $arteaMax1=$batezbestekoa1+2*$desbideratzeEstandarra;
```

6.5. Kodea: *Estatistiken formulak jasotako datuei aplikatuta*

Ondoren, ikasleen emaitzak batzen dituen taula idatziko da. Hauen emaitzak tartearekin konparatuko dira banan-banan, dagokion kolorean adierazteko. Saiakera kopuruari dagokion tarte onargarria modu berean egin da.

² $k = 2 \Rightarrow \%(1 - \frac{1}{2^2}) \times 100 = \%75$

Grafikoak egiteko *chart.js* eta *canvas.js JavaScript* liburutegiak erabili dira.

Emaitzen datuak eta ikasleen profileko datuak aztertuz irakasleak ikasleen egoera ezagutzea da estatistiken helburua. Ikasleen profileko datuen artean, teknologia ezagutza dago. Teknologia ezagutza, galdetegi baten bidez kalkulatu da ikasleen datuak gordetzerakoan. Galdetegian ageri diren galderak honakoak dira:

- Bideojokoak gustuko dituzu?
- Zenbat ordenagailu dituzu etxean?
- Zenbat denbora pasatzen duzu ordenagailuarekin astean zehar? Gutxi gorabehera
- Ordenagailuarekin trebezia duzula esango zenuke?
- Inoiz programatu al duzu?

Galdera bakoitzak bere zergatia dauka. Batetik, bideojokoetan aritzean pertsona baten abstrakzio maila eta arazo baten aurrean erantzuteko gaitasuna eta independentzia garatzen da. Gainera, bideojokoetan ibiltzeak eguneko teknologiarekin trebetasuna duela esan nahi du.

Ordenagailu kopuruak eta astean zehar ordenagailu baten aurrean pasatzen den denborak ere, eguneroko teknologiarekin trebetasuna adierazten du. Azkenik, ordenagailuarekin trebezia izateak eta lehenagotik programatu izanak ere, eragin handia du minijoko hauek burutzeko garaian.

Emaitza bakoitzak puntuazio bat izango du, teknologia ezagutza maila neurtuko duena. Bideojokoak gustuko izateak puntu bat konputatuko du, honen onurak handiak diren arren, teknologia trebetasuna beste galderekin ere neurtzen baita. Ordenagailu bakoitzak eta ordenagailu aurrean pasatako ordu bakoitzak puntu bat konputatuko du. Ordenagailuarekin trebezia izateak bi puntu konputatuko ditu eta lehenago programatu izanak 4. Galderen bidez lortutako puntuek, teknologia ezagutza maila adieraziko dute.

Ikasleen datuak, inguruko hezitzaile eta psikologoei egindako inkesta informal batetik ateratako proposamenak dira. Proiektu honetan denbora eta saiakera kopurua bakarrik hartzen da kontuan estatistikak egiteko, baina ikaslearen profileko datuen arabera estatistikak egitea da ideia. Ikasleen profileko datuen arabera, irakasleak multzokatu daitezke, eta beren emaitzen arabera nolabaiteko ondorio automatikoak atera.

6.7 Erregistratu

Bi erabiltzaile mota daude, eta erabiltzaile bakoitza modu batean erregistratuko da (4 kapitulua). Irakaslea bere kabuz erregistratuko da, bere datu pertsonalak betez. Datu hauek *irakasle_erregistratu.html* fitxategian eskatzen dira eta *irakasle_gordeDatuak.php* fitxategian datu-basean gordeko dira. Datuak datu-basean gordetzerakoan erabiltzailea automatikoki sortuko da, honen inisial eta abizen bera duten erabiltzaile kopurua kalkulatu. Erabiltzaile hauek *irakaslea* eta *ikaslea* taulan konprobatuko dira, ikusi 6.6 kodea.

Bestalde, irakasleak bere irakasleak erregistratuko ditu 4.1 atalean azaldu den bezala. Ikasleen izen eta abizenak emanik, erabiltzaile eta pasahitzak automatikoki sortuko dira

fitxategian, 6.6 kodean bezala. Irakasleak, informazio hau *pdf* motatako fitxategi batean jasoko du.

pdf fitxategia *jsPDF JavaScript*eko liburutegiarekin sortu da. Ikaslearen izen-abizenak *ajax* bidez bidaltzen zaizkio *irakasle_gehituIkasleakDB.php* fitxategiari erabiltzaile berria sortu eta gordetzeko, eta honek erabiltzaile izena eta pasahitza itzuliko ditu modu berean, eragiketa ontzat eman bada. Erabiltzaile eta pasahitza jasotzean, *pdf* fitxategia sortuko da.

```

1 //<iniziala><abizena>XXXX motako erabiltzaileak bilatu bi taulatan
2 $sql="SELECT * FROM ikaslea WHERE idIkaslea LIKE '$iniziala$abizena%'";
3 $result=mysqli_query($link,$sql);
4 $count=$result->num_rows;
5 $sql="SELECT * FROM irakaslea WHERE idIrakasle LIKE '$iniziala$abizena%'";
6 $result=mysqli_query($link,$sql);
7 $count+=$result->num_rows;
8 $count++;
9 //lortutako emaitza kopuruaren arabera 4 digituzko zenbakia sortu
10 if($count<10){
11     $count = "000".$count;
12 }else if($count<100){
13     $count = "00".$count;
14 }else if($count<1000){
15     $count = "0".$count;
16 }
17 //Erabiltzaile berria eraiki, honen iniziala, abizena eta 4 digituko zenbakiarekin
18 $erabiltzaileBerria=$iniziala.$abizena.$count;

```

6.6. Kodea: *Erabiltzailea automatikoki sortzea*

jsPDF liburutegia erabiltzeko, lehenengo *jsPDF* elementu mota sortu behar da. Bertan zerbait idazteko, *text* funtzioa erabili behar da, testuaren x eta y koordinatuak eta idatzi nahi den testua jasoko duena. Azkenik, *output* funtzioarekin, *pdf* fitxategia bistaratuko da, ikusi 6.7 kodea.

```

1 var doc = new jsPDF();
2 doc.text(20, 20, "testua");
3 doc.output('dataurlnewwindow');

```

6.7. Kodea: *pdf fitxategi bat sortu JavaScript bidez*

Hostingerreko zerbitzarira igotzean, *.csv* fitxategia kargatzeak errorea ematen zuen, fitxategia irekitzerakoan fitxategia ez baitzuen aurkitzen. Fitxategia zerbitzarian tratatzen denean, karpeta tenporalean gordetzen da, horregatik fitxategia irekitzerakoan karpeta tenporaleko karpetan kokatuta dagoela zehaztu behar da, ikusi 6.8 kodea.

```

1 $file = $_FILES['filePath']['tmp_name'];

```

6.8. Kodea: *Karpeta tenporalean dagoen fitxategia irakurri*

6.8 Emaita pertsonalak ikusi

Ikasleek beren emaitza pertsonalak, minijoko bakoitzerako behar izandako denbora eta saiakera kopurua, ikusi ahal izango dituzte. Datu horiek datu-basetik jaso eta taulan idatziko dira. Minijokoa pasatzeko behar izandako denbora, milisegundotan gordeko da. Milisegundotan jasotako denbora, *orduak:minutuak:segundoak:milisegundoak* itxuran azalduko dira.

Atal hau *irakaslea_emaiztaLista.php* fitxategian dago. Ikaslearen erabiltzaile izena *GET* metodoaren bidez jasoko da, irakasleak ikaslearen datu pertsonalak ikusterakoan, fitxategi bera ahal izateko.

6.9 Datuak editatzea

Ikasleek eta irakasleek beren datu pertsonalak editatu ahal izango dituzte. Datuak ikustea eta editatzea *ikaslea_perfila.php* eta *irakaslea_perfila.php* fitxategitan egiten da hurrenez hurren. Datuak editatu ondoren, eta datu guztian ondo bete direla konprobatuta, datu-basean gordeko dira *ikaslea_datuakGorde.php* eta *irakasle_gordeDatuak.php* orrietan.

6.10 Itsuentzako pantaila

Irigarritasuna, dibertsitate funtzionala duten pertsonentzat, beharrezko baldintza da parte-hartze soziala eman dadin. Gaur egun, web orri gehienek irigarritasun oztupoak dituzte, eta hauek atari digitalen erabilera zaildu edo ezinezkoa egiten dute. Alabaina, web zerbitzu eta aplikazioak irigarriak balira, ezgaitasuna duten pertsonak zerbitzu hauek modu eraginkorrean erabili ahalko lituzkete.

Atal honetan, irigarritasuna zer den eta Gradu Amaierako Lan honetan nola aplikatu den azaltzen da.

6.10.1 Irigarritasuna

"Irigarritasunari eman ahal dakioken lehenengo definizioa, alderdi geografikotik dator eta leku batera sartzeko erraztasuna adierazten du"(Martínez eta Lara, 2006, 9.or) [14].

Ingurune fisikoen irigarritasuna, lekuek duten ezaugarri bat da, edozein motatako pertsonen leku eta eraikuntza guztietara modu autonomoz iristeko aukera ematen diena. Era berean, web irigarritasuna erabiltzaile orori webgune baterako sarbidea ahalbidetzean datza.

Web irigarritasunak erabiltzaile guztiei, hauen ezgaitasuna edo nabigazio testuingurua dela ere, web orriko zerbitzu eta eduki guztietara iristeko aukera ematen die.

6.10.2 Laguntza teknikoak

Irigarritasuna sustatzeko, inguruko irismen oztupoen aurre egiteko zenbait laguntza erabiltzen dira, edozein dibertsitate mota duten erabiltzaileek edozein ekintza burutzeko laguntza ematen dietena.

Laguntza teknikoa, ezgaitasun bat duen pertsonak erabiltzen duen produktu, instrumentu, tresna edo sistema tekniko bat da, honen ezintasuna konpentsatzen duelarik. Laguntza teknikoen helburua, aniztasun funtzionala duten pertsonen gizartratze eta parte hartzea lortzea da, hauen ingurunea egokituz.

6.10.3 Pantaila irakurleak

Pantaila irakurle bat, pantailan dagoena ezagutu eta interpretatzen duen software aplikazio bat da. Interpretazio hau, ahots sintetizadore edo braille bidez egiten da.

Pantaila irakurleak itsuentzako, bisio baxuko edo irakurmen zailtasunak dituzten erabiltzaileentzako pentsatutako laguntza tekniko mota bat da.

Lehenengo pantaila irakurlea 80.hamarkadan garatu zen, BBC Micro eta Nec ordenagailu eramangarrientzako. Ordenagailu hauek, komando lerro bidezko interfazea zuten. Arazoa interfaze grafikoekin batera etorri zen. Interfaze grafikoak ez du testuzko irudikapen zehatzik eta beraz, pantaila irakurleak teknika berriak erabiltzera behartuta daude.

Gaur egun, pantaila irakurleak, sistema eragileen eta bertako aplikazioen, web inguruneak barne, erabilgarritasuna bermatzen duten programak dira. Pantaila irakurle ezagunenak windowserako JAWS, Appleko VoiceOver eta NVDA eta linuxerako Orca dira.

6.10.4 Blockly irisgarria

3.1 atalean aipatu den moduan, blockly-k, kode blokeak mugituz lortutako kodea aukeratutako programazio lengoaiara itzultzen du. "Drag and Drop" motatako interakzio hori ezgaitasun bisuala edo mugimendu urritasuna duten erabiltzaileentzako oztopoa da. Erabiltzaile mota hauek, teklatu bidezko elkarrekintza erabiltzen dute. Gainera, blockly liburutegiak eskainitako *toolbox* eta *workspace* eremuak ez dira pantaila irakurleentzako irakurgarriak.

Arazo hori konpontzeko blockly-k ikusmen aniztasuna duten pertsonentzako interfazea proposatzen du [15]. Proposamen honetan, sagu bidezko iterazioa teklatu bidezko nabigazioarekin ordezkatzeko da³.

Proposamen honen kodea *blockly/demos/accessible* direktorioan aurkitzen da. Bertan web aplikazioan *blockly accessible* erabiltzeko, beharrezkoa den kodea dago. Fitxategiak demoan dauden ordena berean inportatu behar dira.

Bertan aurki daitekeen *sidebar*-en edozein botoi gehitu daiteke *ACCESSIBLE_GLOBALS* atributuan definituz. Horretarako, botoiaren *text*, *action* eta *id* atributuak zehaztu behar dira. Atributu hauek botoiaren testua, hau klikatzerakoan exekutatu beharreko funtzioa eta honen identifikadorea dira hurrenez hurren (ikusi 6.9 kodea).

```

1  var ACCESSIBLE_GLOBALS = {
2    // Prefix of path to sound files.
3    mediaPathPrefix: '.././accessible/media/',
4    // Additional buttons for the sidebar.
5    customSidebarButtons: [{
6      text: 'Run your program',
7      // Gets set by musicGame.StageComponent.
8      action: function() {
9        var code = Blockly.JavaScript.workspaceToCode(blocklyApp.workspace);
10       try {
11         alert(code);
12       } catch (e) {
13         alert(e);
14       }
15     },
16     id: 'runBtn'
17   }]
18 };

```

6.9. Kodea: Interfaze irisgarrian kodetutako kodea lortzeko botoiaren definizioa

³Blockly accessible <https://blockly-demo.appspot.com/static/demos/accessible/index.html> (2017/06/22 kontsultatuta)

6.10.5 Garapena

Proiektu honetan, 6.10.4 atalean aipatutako blocklyk eskuragarri duen interfaze irisgarriaren gainean egin da lan. Helburua ikaskide guztien gaitasun ezberdinak aztertzea denez, garrantzitsua da ikasle guztiek minijoko berdinarekin lan egitea.

Proiektu honetan, *blockly* eskuragarri duen *maze* jokoaren bertsio irisgarria egin da. Sagu bidezko hartu-emana ordezkatzekak dibertsitate bisuala edo/eta zenbait motrizitate arazo dituzten pertsonen irisgarritasuna bermatzea du helburu. Horregatik, bertsio irisgarri honek begi-bistaz eta pantaila irakurleen bidez ulertzeko modukoa izan behar du.

Blockly games bildumako *maze* jokoak duen ohola, *table* etiketa motatako elementuaz ordezkatu da. Elementu mota hau bisuala eta pantaila irakurleentzako irakurgarria da.

```

1 <maila id="1">
2   <table>
3     <tr>
4       <td>X</td>
5       <td>X</td>
6       <td>X</td>
7       <td>X</td>
8       <td>X</td>
9       <td>X</td>
10    </tr>
11    <tr>
12      <td>X</td>
13      <td>R</td>
14      <td>B</td>
15      <td>H</td>
16      <td>X</td>
17      <td>X</td>
18    </tr>
19    <tr>
20      <td>X</td>
21      <td>X</td>
22      <td>X</td>
23      <td>X</td>
24      <td>X</td>
25      <td>X</td>
26    </tr>
27  </table>
28  <hasierakoEgoera>
29    <i>1</i>
30    <j>1</j>
31  </hasierakoEgoera>
32  <bukaerakoEgoera>
33    <i>1</i>
34    <j>3</j>
35  </bukaerakoEgoera>
36  <toolbox>
37    <category name="move" colour="210">
38      <block type="move_forward"></block>
39      <block type="biratu_ezkerreta"></block>
40      <block type="biratu_eskubira"></block>
41      <block type="errepikatu_helmuga_arte"></block>
42    </category>
43  </toolbox>
44 </maila>

```

6.10. Kodea: *xml fitxategiko lehenengo minijokorako definizioa*

Taula honetako lauki bakoitzak, hizki bat izango du, egin daitezkeen mugimenduak adierazten dituenak. Hizki hauek lau motatako lauki ezberdinak definitzen dituzte. Hauek

dira egon daitezkeen lauki motak:

- Bidea: Helmugara iristeko igaro daitezken laukiak adierazten ditu, B hizkiaz irudikatuko dira.
- Pareta: Helmugara iristeko igaro ezin diren laukiak adierazten diru, X hizkiaz irudikatuko dira.
- Helmuga: Helmuga adierazten duen lauki bakarra egongo da H hizkiaz irudikatuta.
- Momentuko posizioa: Momentuko posizioa eta norabidea adierazten du lauki honek. Lau hizkien bitartez adieraziko da momentuko posizioa, hizki hauek norabidea adieraziko dutelarik. Euskarazko anbiguitasuna ekiditeko ingelesezko norabideak hartu dira, *up* (gora), *down* (behera), *right* (eskuin) eta *left* (ezkerra). Hala, U , D , R eta L hizkiek momentuko posizioa adieraziko dute, honen norabidea zehaztuz, gora, behera, eskuin eta ezkerre hurrenez hurren.

Joko hauek burutzeko, *maze* jokoak erabiltzen dituen bloke berdinak sortu dira: mugitu aurrera, biratu ezker/eskuinera, errepikatu helmugara iritsi arte, bidea badago egin eta bidea badago egin bestela. Bloke hauek 3.1.3 atalean azaldutako modu berean sortu dira. Bloke berri hauek erabiltzeko, bloke hauek definituta duen fitxategia eta itzuliko duen kodea zehaztuta duen fitxategia gehitu beharko dira minijoko hau eraiki den fitxategian.

Minijoko mailak *xml* fitxategi batean daude. Maila bakoitzak honi dagokion taula, hasierako posizioa, helmugako posizioa eta eskuragarri egongo diren blokeak izango ditu (ikusi 6.10 kodea). Minijoko bakoitza fitxategi berean egongo da, eta *xml* fitxategian aurkitzen diren datuak aldatuko dira maila bakoitzeko (ikusi 6.11 kodea). 6.3 irudia, *maze* jokoko lehenengo mailako ohol irisgarria da.

```

1 function kargatu_minijokoa(m){
2   var xmlDoc=loadXMLDoc();
3   var arniveles = xmlDoc.getElementsByTagName("maila");
4   var momentuko_minijokoa = arniveles[m];
5   var minijokoaTaula = momentuko_minijokoa.getElementsByTagName("table");
6   tr = minijokoaTaula[0].getElementsByTagName('tr');
7   var table = document.getElementById("minijokoaTable");
8   table.innerHTML = "";
9   for (e=0;e<tr.length;e++){
10    row = table.insertRow(e);
11    td = tr[e].getElementsByTagName('td');
12    for(p=0;p<td.length;p++){
13      cell = row.insertCell(p);
14      cell.innerHTML = td[p].innerHTML;
15    }
16  }
17  var minijokoToolbox = momentuko_minijokoa.getElementsByTagName("toolbox");
18  console.log(minijokoToolbox);
19  for(i=0; i<minijokoToolbox.length; i++){
20    document.getElementById("blockly-toolbox-xml").appendChild(minijokoToolbox[i]);
21    console.log(document.getElementById('blockly-toolbox-xml'));
22  }
23  var hEgoera = momentuko_minijokoa.getElementsByTagName("hasierakoEgoera");
24  i = parseFloat(hEgoera[0].getElementsByTagName('i')[0].innerHTML);
25  j = parseFloat(hEgoera[0].getElementsByTagName('j')[0].innerHTML);
26  var bEgoera = momentuko_minijokoa.getElementsByTagName("bukaerakoEgoera");
27  bi = parseFloat(bEgoera[0].getElementsByTagName('i')[0].innerHTML);
28  bj = parseFloat(bEgoera[0].getElementsByTagName('j')[0].innerHTML);
29 }

```

6.11. Kodea: *xml fitxategitik datuak kargatzeko kodea, minijokoaren mailaren arrabera*

```

XX XXXX
XMBBHXX
XX XXXX

```

6.3. Irudia: Pantaila irakurleen bidezko ohol irisgarria

Minijokoaren taula edozein momentutan eskuragarri izateko, *accesskey* atributua erabili da, taula honetarako lasterbidea definitzen duena. *Accesskey* atributua ez da etiketa mota guztietan erabilgarri. Horregatik, *a* etiketa duen elementua sortu da, *accesskey* atributuaren bidez taulara birbidaliko duena (ikusi 6.12 kodea).

```

1 <a href="#minijokoaTable" accesskey="m">Minijokoa taula </a><br>
2 <table id="minijokoaTable"></table>

```

6.12. Kodea: Minijokoko taularako lasterbidea

Taulan gertatzen den aldaketa bakoitzeko abisu mezu bat bidaliko da, bisualki ikusten dena adierazteko. Abisu mezu hauek blocklyk erabiltzen dituen abisu mezueta oinarritzen dira. Abisu hauek aurretik definitutako *blocklyAriaLiveStatus* motako elementuan agertuko dira (ikusi 6.13 kodea).

```

1 <div id="emaitza" class="blocklyAriaLiveStatus">
2   <span aria-live="polite" role="status"></span>
3 </div>

```

6.13. Kodea: Taulan gertatzen den aldaketa ororen abisu mezua

Abisu mezuek 6.1 atalean azaldutako antzeko problema eragin du. Bi abisu mezu jarraian bidaltzean, bigarren abisu mezua bakarrik entzuten zen. Hori ekiditeko, denbora kontagailu bat erabili da. Deitzen den funtzio bakoitzak 3 segundo gehituko dizkio denbora kontagailu honi, eta era berean, funtzioak egin beharko lukeena exekutatzeko, denbora kontagailu horrek markatzen duena itxarongo du (ikusi 6.14 kodea). Honela, ahots mezu bakoitzeko 3 segundoko tartea egongo da, eta ez dira gainezarriko.

```

1 function move_forward() {
2   setTimeout(function() {
3     move_forward2();
4   }, time);
5   time=time+3000;
6 }

```

6.14. Kodea: Ahots mezuek gainezartzea ekiditeko denbora kontagailua

7. Kapituluia Emaitzak

Kapitulu honetan, lagin erreal batetik bigarren hezkuntzako irakasle batek ateratako ondorioak adierazten dira. Lagin bariatua bilatu da, hainbat adinetako eta genero ezberdineko ikasleak izanik. Horien artean batzuk programatzaileak dira. Ondorio hauen bidez, web aplikazioaren erabilgarritasuna erakutsi nahi da, probako lagin honetatik irakasle batek izan ditzakeen ondorio pedagogikoak simulatuz. Ikasleen erabiltzaile izenak aldatu dira, horien pribatutasuna errespetatzeko. Lagin honek lortutako emaitzak A eranskinean ikus daitezke.

Lehenengo minijokoko emaitzetan (A. A atala), *ik16* erabiltzailea gorritz azaltzen da bi tauletan. Honek, batz bestekoa kontuan hartuz beste ikasleekiko desbideratzen dela esan nahi du. Erabiltzaile honen jarraipena egiten bada, beste minijokotan tarte onargarriaren barruan dagoela ikusten da. Lehenengo emaitzak deigarriak diren arren ez du esan nahi ikaslea gaizki dabilela, milaka arrazoi egon daitezke. Lehenengo minijokoa izanik, minijokoa eta burutu beharreko problema ulertzeko arazoak izan duela ondoriozta daiteke. Arazo hau konpontzeko, kontzeptua ulertzeko azalpen gehiago emango zaizkie ikasleei.

Lehenengo minijokoan hurrengo emaitza okerrean lortu duen ikaslea *ik8* erabiltzailea da. Honen jarraipena egiten bada, orokorrean emaitza okerrean lortzen dituen ikaslea dela ikusten da. Gainera, 5. minijokoaren ondoren minijokoak alde batera utzi dituela ikusten da (7.1 irudia). Denbora gehiegi behar izateak motibazioa galtzea eragiten du. Ikasleek minijoko hauek edo beste motatako ariketak burutzea bermatzeko, hauen motibazioa sustatu behar da. Ikasleen motibazioa sustatzeko, ikasleen ezaugarriak ezagutzeko onuragarria da. Ikasleen ezaugarriak ezagutzeko, ikasle hauen profila erabiltzen da.

Minijokoa	Behar izandako denbora	Egindako saiakera kopurua
'1'	'0:03:40.423'	'3'
'2'	'0:04:09.19'	'4'
'3'	'0:03:57.34'	'2'
'4'	'0:04:42.432'	'2'
'5'	'0:02:59.323'	'4'

7.1. Irudia: *ik8* ikaslearen emaitzak

Minijokoen parte hartzea begiratzuz, 3 behetze nagusi daudela ikusten da. Behetze hauek 6,8 eta 10. minijokoan gertatzen dira (ikus 7.1 taula). Honek maila batetik bestera dagoen desberdintasuna adierazten du. 6. minijokoan bloke berri bat agertzen da: baldintza blokea. Baldintza kontzeptua berria da, eta 3 ikasleek ezin izan dute jarraitu. 8.

minijokoan eren beste 3 ikaslek utzi dute aplikazioa alde batera. Behetze handiena dudarik gabe azkeneko minijokoa izan da, 18 ikasletik 6k bakarrik bukatu baitute minijoko sorta.

Minikojoa	Ikasle kopurua
1	18
2	18
3	17
4	17
5	16
6	15
7	15
8	12
9	12
10	5

7.1. Taula: *Minijoko bakoitzak izandako ikasle kopurua*

Minijokoak asmatzeko behar izandako denbora eta saiakera kopurua bat datoz minijokoak abandonatzearekin. Bataz bestekotik gehien aldentzen diren ikasleak, minijokoak alde batera uzten lehenengoak izan dira. Bestalde, minijoko guztietan emaitza onak lortu dituen talde bat dago, minijoko honen zailtasunak eragin ez diona. Talde honen barruan azken minijokoetan diferentzia nabarigarriagoa den arren, ikasleen %33.33-k bakarrik jarraitu duela ikusita, ez da diferentzia kezkarria.

Argi dago beraz, ikasleen artean 3 multzo bereizten direla, nolabait ere, 3 behetze nagusiek banatzen dutena. Ikasle multzo bakoitzak metodo pedagogiko propioa beharko du, helburu berdina duen arren. Emaitza oso onak lortu dituzten ikasleek, metodo berdina jarraitu dezaketela argi dago, eta minijokoen zailtasun maila igo daitekeela.

Ikasleen %66.66-k ezin izan du minijoko sorta bukatu eta horietako batzuk erdia bakarrik bete dute (ikus 7.1 taula). Ikasleen gehiengoak irakasleak jarritako lana burutu ezin izateak, ikasle eta irakasle arteko komunikazio falta bat erakusten du. Datu hauei esker, ikasleen benetako egoera ezagutu daiteke eta kasu honetan, irakasleak eskatzen duena irakatsitakoarekin bat ez datorrela ikusten da. Datu hauek ikusita, ariketen zailtasun maila aldatu behar dela ondoriozta daiteke edo eta ikasleei eskatu nahi zaien maila klasean irakastea bermatzea.

Zailtasun mailaz gain, motibazioan eragin dezaketene beste faktore batzuk daude. Kasu honetan, ikasle guztiek modu jarraian bete dituzte minijokoak eta kontzentrazio mailak, denboraren dedikazioak... jokuak abandonatzea eragiten du. Horregatik ikasle hauen kasuan, 10 minijoko jarraian eman beharrean, eraginkorragoa izan daiteke jokoak 3naka bidaltzea.

8. Kapituluia

Jarraipena eta kontrola

8.1 Irismena

Proiektu honen helburua, blockly liburutegia erabiliz (3.1 atala), hezkuntzarako diseinatutako programatzen ikasteko web plataforma bat sortzea izan da. Plataforma honen bidez, irakasleak bere ikasleen datuak eta ikasleen arteko gaitasun mailen berezitasun ezberdinak ikusteko aukera izango du.

Atal honetan, proiektuan zehar erabilitako plangintzari eginiko aldaketak azaltzen dira.

8.1.1 Proiektuan lortu diren helburuen deskribapena

Web aplikazio bat egin da, non bi erabiltzaile mota egongo diren. Erabiltzaile hauentzako saio hasiera burutu da, eta baita erabiltzaile bakoitzari dagokion orri nagusia ere.

Alde batetik, irakaslearen erregistro orria, bere datuak ikusi eta editatzeko orria eta ikasleen zerrenda orria egin da. Ikasleen zerrendaren bidez, ikasleen datuak eta hauen emaitzak ikusten dira, eta baita ikasle berriak gehitu edo daudenak ezabatuko dira. Gainera, ikasleen emaitzak biltzen dituen grafikoak egin dira, sistemak eginiko estatistikak erakusten dituenak.

Beste aldetik, ikasleek beren datuak ikusi eta editatzeko orria eta beren emaitzen zerrenda orria egin da.

Sistemari dagokionez, ikasleen minijoko bakoitzeko behar izandako denbora eta saiakera kopurua gordetzen da eta datu hauen inguruko estatistikak kalkulatzeko dira.

Gainera, tekla bidezko hartu emana onartzen duen minijoko sorta ere egin da, mugimendu urria eta dibertsitate bisuala duten erabiltzaileen irisgarritasuna bermatuz.

8.1.2 Irismenarekin erlazionaturik gertatu diren arazo deskribapena

Proiektua garatzeko garaian, zenbait arazo gertatu dira, hasierako plangintzan definitutako irismena guztiz betetzea galarazi duena. Eragin handien izan duen arazoa, minijoko sorta sortzerakoan izandako arazoa izan da. 6.1 atalean adierazten den bezala, posizio aldaketa modu dinamikoan erakusteko erabilitako *CSS3 Animations* teknologiak posizio aldaketak gainjarrita adierazten zituen, eta ez zen posizioa modu dinamiko egokian adierazten.

Arazo hau proiektuaren hasieran gertatu da eta proiektuan eragin dezaken domino-efektua ekiditeko *blockly games* joko multzoko *maze* jokoa hartu da. Arazoari irtenbidea aurkitu zaion arren, *CSS3 Animations* teknologia eta *blockly games* liburutegia

aztertze erabilitako denbora planifikaziotik at zegoen, eta proiektuaren atazetan atzerapen bat eragin du.

8.1.3 Kanpo geratu diren helburuen deskribapena

8.1.2 atalean azaldu den bezala, *blockly* liburutegia erabiltzen ikasi den arren, gertatutako arazoen ondorioz ez da hasierako planifikazioan zegoen joko sorta burutu. Jada egina dagoen joko bat hartzeak ere zenbait eragin izan ditu. Joko honetan aldaketak egitea zaila izan da eta sagu mugimenduen kopurua ezin izan da kontrolatu.

8.1.4 LDE-an egindako aldaketak

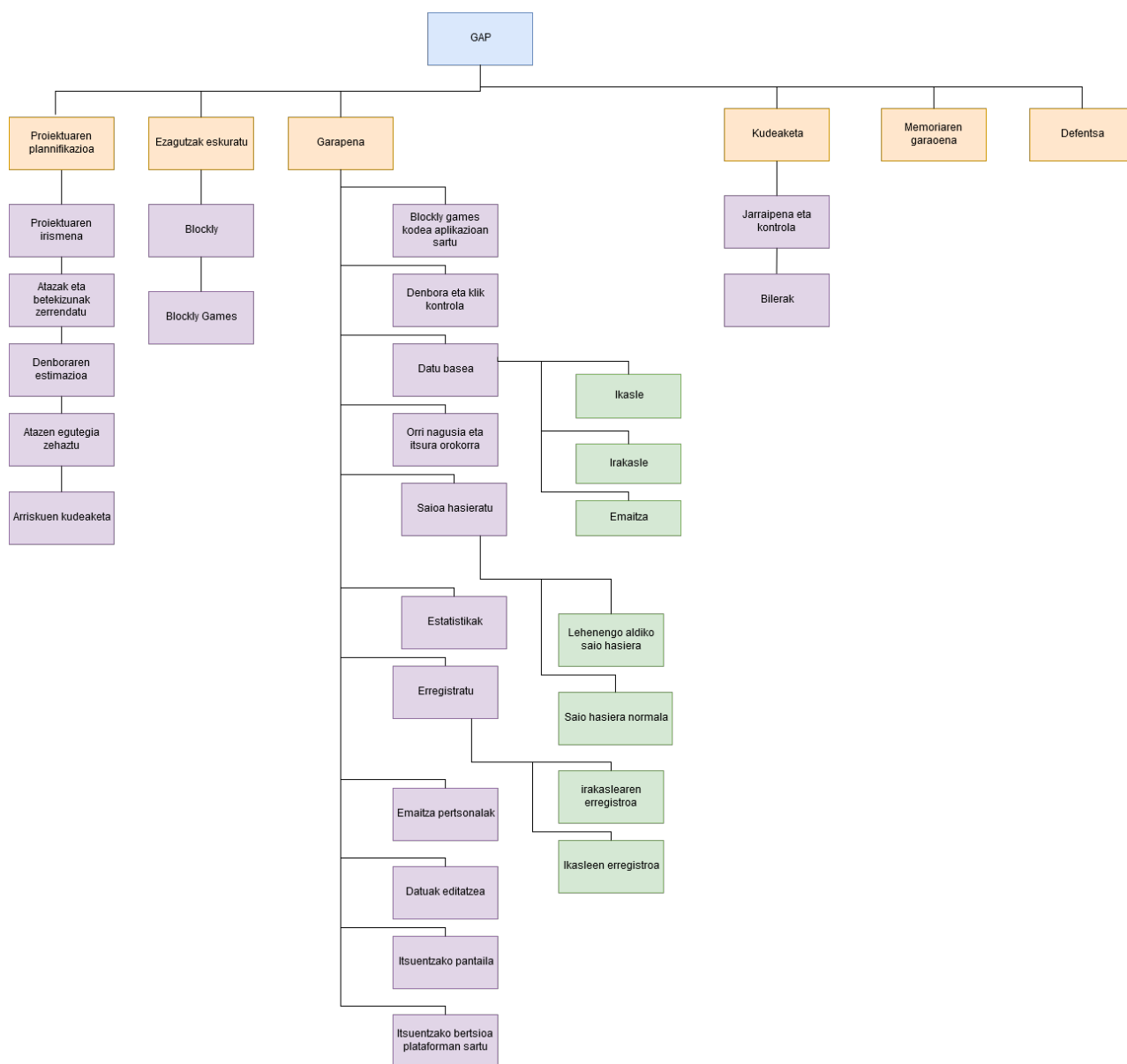
8.1 irudia proiektuaren bukaerako LDE diagrama da.

8.2 Denboraren banaketa

8.2.1 Egin diren ataza berrien deskribapena

8.1.2 atalean azaltzen den bezala, minijoko sorta egin beharrean, *blockly games* joko multzoa erabili da. Honek, *blockly games* joko bilduma aztertzea eragin du. Bi ataza berri sortu behar izan dira kasu honetan: *blockly games* bildumari buruzko informazioa jasotzeko ataza eta honen kodea web aplikazioan gehitzeko ataza.

Beste aldetik, 6.3 atalean adierazten den moduan, hasierako plangintzan zehazten zen datu-basea aldatu da. Horrek, denboraren plangintzan eragin handirik izan ez duen arren, bi ataza berri sortzea eragin du: irakasle taula eta ikasle taula sortzea.

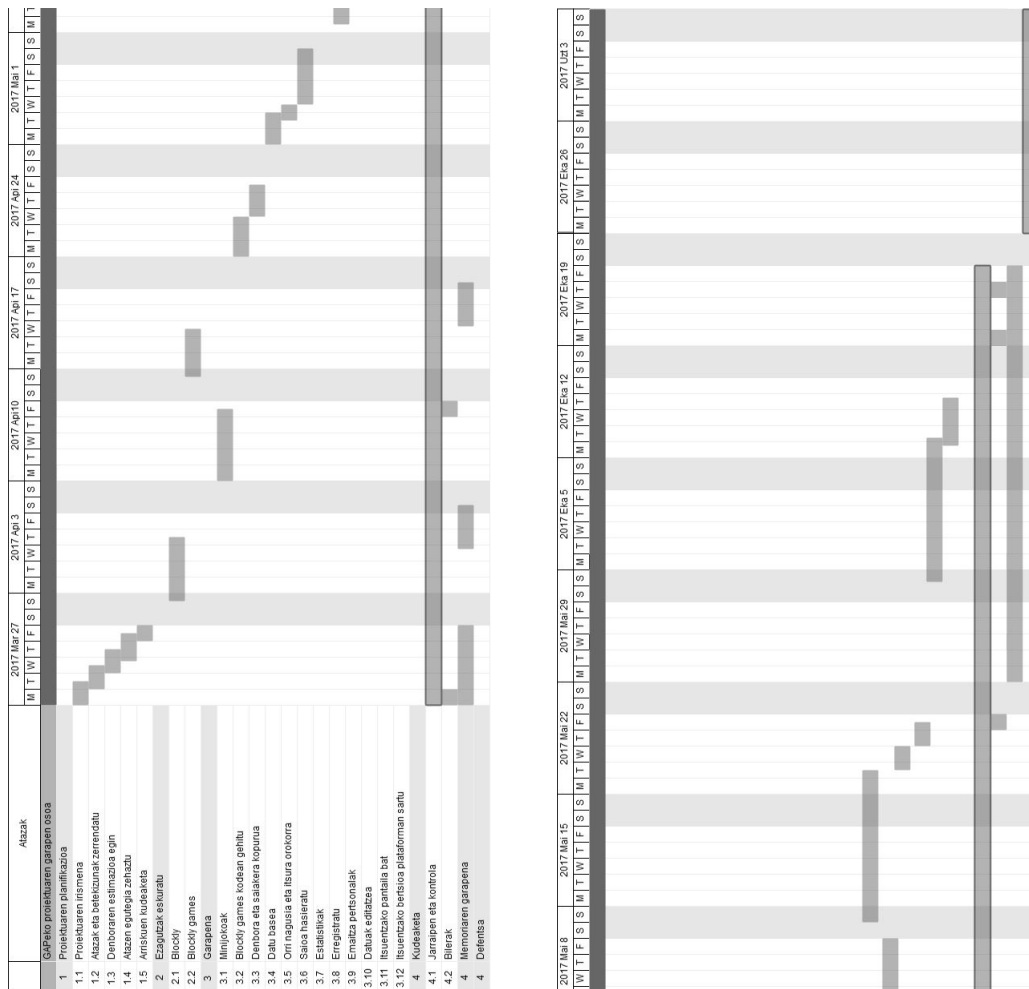
8.1. Irudia: *LDE diagrama*

8.2.2 Ataza bakoitzari emandako denbora

Ataza	Estimatitako orduak	Behar izandako denbora	Desbideratzea (%)
1. Proiektuaren plangintza			
1.1 Proiektuaren irismena	2	2	0
1.2 Atazak eta betekizunak zerrendatu	5	5	0
1.3 Denboraren estimazioa egin	2	2	0
1.4 Atazen egutegia zehaztu	2	3	0
1.5 Arriskuen kudeaketa	3	3	0
Plangintza guztira	14	14	0
2. Ezagutzak eskuratu			
2.1 Blockly	10	15	50
2.2 Blockly games	X	10	X
Ezautzak eskuratu guztira	10	25	150
3. Garapena			
3.1 Minijokoak	15	20	33.3
3.2 Blockly games kodean gehitu	X	7	X
3.3 Denbora eta saiakera kopurua	7	10	42
3.4 Datu-basea			
3.4.1 Erabiltzailea	2	X	-100
3.4.2 Ikaslea	X	2	X
3.4.3 Irakaslea	X	2	X
3.4.4 E-maila	2	2	0
3.5 Web plataformako orri nagusia eta itxura orokorra	10	8	-20
3.6 Saioa hasieratu			
3.6.1 Lehenengo aldiko saio hasiera	10	15	50
3.6.2 Saio hasiera normala	3	2	-33.3
3.7 Estatistikak	25	35	40
3.8 Erregistratu			
3.8.1 Irakasleen erregistroa	3	2	-33.3
3.8.2 Ikasleen erregistroa	7	10	42
3.9 E-maila pertsonalak	8	5	-37.5
3.10 Datuak editatzea	5	5	0
3.11 Itsuentzako pantaila	34	45	32.3
3.12 Pantaila edo minijoko kopurua handitu	12	X	-100
3.13 Sagu mugimendu kopurua kontrolatu	6	X	-100
3.14 Itsuentzako bertsioa plataforman sartu	15	10	-50
Garapena guztira	164	180	9.7
4. Kudeaketa			
4.1 Jarraipena eta kontrola	8	8	0
4.2 Bilerak	7	5	-28.5
Kudeaketa guztira	15	13	-13.3
5. Memoriaren garapena	77	90	16.8
6. Defentsa	20	20	0
Guztira	300	342	14

8.1. Taula: Ataza bakoitzari emandako denbora eta aurreikusikoarekiko desbideratzea

8.2.3 GANTT diagramak jasandako aldaketak

8.2. Irudia: *Bukaerako gantt diagrama*

9. Kapituluia

Ondorioak eta etorkizuneko lana

Proiektuan zehar, hezkuntzara bideratutako web aplikazio bat garatu da, non ikasleek zenbait minijokoen bitartez programatzen ikasiko duten eta irakasleek haien emaitzak aztertu ahal izango dituzten, bere irakaskuntza jardueraren jarraipen egokia egin ahal izateko.

Hasierako proposamena, *blockly* liburutegia erabiliz, minijoko hauek sortzea bazen ere, zenbait arazoren ondorioz ezin izan da helburu hau burutu eta *Blockly games* joko multzoko *maze* jokia erabili da web aplikazioan. *Blockly* liburutegiak, bloke bidezko programazioa irakasteko erabiltzen da, honek eskaintzen dituen blokeen bidez edo bloke berriak sortuz.

Bloke bidezko programazioa, blokeak arrastatuz egiten da. Baina, mugimendu hau ezinezkoa suertatzen da ikusmen eta mugimendu urritasuna duten pertsonentzako. Hasiera batetik, tekla bidezko elkarrekintza onartzen duen interfazea proposatzea izan da proiektuaren helburuetako bat. Azkenean, web aplikazioan *blockly games* joko multzoko *maze* jokia erabili denez, joko honen bertsio irisgarria garatu da.

Estatistikak datu errealekin egiteko asmoz, proiektua *hostinger* zerbitzu batera igo eta gertuko artean banatu da. Lagin honetan, lehendik programatu duten jendea eta inoiz programatu ez duten jendea biltzen da, 20-60 adin tartean. Helburua hainbat ezaugarri dituzten jendea biltzea da, irakasle batek atera ditzakeen ondorioak simulatzeko. Lagin honen emaitzak jasota, Bigarren Hezkuntzako irakasle batekin eduki da elkarriketa, lagin honek emandako grafikoak erakutsiz irakasle honek ondorioztatutako emaitzak gorde direlarik (7 kapituluia).

9.1 Etorkizuneko lana

Proiektua proposatzerakoan eta garapenean zehar, proiektu honetan garatu ezin izan diren zenbait ideia irten dira.

Hasteko, proiektuaren proposamenean *blockly* liburutegia erabiliz, minijoko berriak sortzea zen helburuetako bat. Zenbait arazo direla eta, helburu hau ezin izan da bete eta ondorioz, sagu mugimenduaren kalkulua ere ezin izan da burutu. Proiektu honekin erlazionatutako etorkizuneko lanean, *blockly* liburutegiari buruz jasotako informazioan oinarrituz, joko multzo berria sortu beharko da, proiektu honetan izandako arazoari soluzioa bilatuz. Arazo honi eman dakioken soluzio bat, ahots mezuekin egindakoa izan daiteke (6.10.5 atala) . Sagu mugimenduari dagokionez, *blockly* liburutegia erabiliz izango den *workspace* aldagaiari *onchange* gertaera gehituz, *workspace* aldagaia aldaketa kopurua

jaso daiteke. *Workspace* aldagaian, exekutatu nahi den bloke segida egongo da eta beraz, *workspace* aldatzen den bakoitzean, bloke multzo honen gainean aldaketa bat egin dela adieraziko du.

Ikasleen emaitza aztertzerakoan, hauen denbora eta klik kopurua kontuan hartzeaz gain, ikasleak sortutako bloke segida gordetzea interesgarria dela pentsatu da. *Blockly* liburutegiko *code* demoan ¹ bloke bidez sortutako kodea elkarbanatzeko aukera ematen du.

Proiektu honetan, minijoko eta ikasle bakoitzeko emaitza bakarra gordetzen da. Etorkizunera begira, ikasleek beraien emaitza hobetzeko aukera egotea pentsatu da, minijoko batekiko honen hobekuntza aztertuz, ikaslearen eboluzioa aztertu baitaiteke.

Ikasle lagnaren datuak aztertzerako garaian, emaitzekin batera minijokoari buruzko informazioa faltan bota da. Minijoko batetik bestera gertatutako emaitza aldaketak ondorioztatzeko, minijokoaren zailtasun maila, deskribapena, bloke kopurua, agertu diren bloke berriak ... baliagarria den informazioa da. 6.3 atalean azaltzen den bezala, etorkizunean minijokoari buruzko informazioa gordetzeko asmoa dago.

Web plataformaren segurtasunari dagokionez, administratzaile papera beteko duen erabiltzaile berria sortzea pentsatu da. Web aplikazio honetan edonor erregistra daiteke irakasle bezala, eta honek arrisku puntu bat du. Rol berri honek irakasle bezala erregistratzen diren erabiltzaileak onartu beharko ditu. Gainera, gailurreko erabiltzaile berri honek, erabilpen globalagoa ematen dio web aplikazioari. Irakasleek, bere ikasleen emaitzak ikusi eta aztertzeaz gain, administratzaileak irakasleen arteko ezberdintasunak ikusi eta aztertu ahal izango du, ikastetxeen arteko maila aztertu ahal izango duelarik.

Administratzaile erabiltzailea gehituz, irakasleak eta hainbat ikastetxeko ikasleak ere aztertuko dira beraien artean eta horregatik, hauen profila osagarriagoa egitea ondo egongo litzateke. Ikasleen datu gehiago eta lagn handiagoa izanik, estatistikak ondorio zehatzagoak modu automatikoan ateratzen lagundu dezake. Horretarako, metodo aproposa pentsatzeaz gain, ikasleen emaitzetan eragin dezaken datuen azterketa sakonagoa egin beharko da.

Irigarritasunari dagokionez, *maze* jokoaren bertsio irigarria egin da proiektu honetan. Etorkizunerako, beste motatako jokoaren bertsio irigarria egitea proposatzen da.

9.2 Ikasitako lezioak

Proiektu honetan zehar hainbat gauza ikasi dira. Web aplikazioa izanik, *html*, *php* eta *JavaScript* lengoaien trebetasuna garatu da. Besteak beste, *html* orri bat beste *html* orrian kargatzeko *iframe* etiketa dago eta teklatu bidezko lasterbideak egiteko *accesskey* atributua dute `<a>`, `<area>`, `<button>`, `<input>`, `<label>`, `<legend>`, eta `<textarea>` etiketek.

*JavaScript*eko liburutegiei dagokionez, grafikoak egiteko *chart* eta *canvas* liburutegiak ezagutu dira eta *jsPDF* liburutegia *pdf* fitxategiak sortzeko.

Bestalde, denbora luzerako proiektu bat garatzeko antolakuntza eta arazo baten aurrean azkar erantzutea ezinbestekoa da. Edozein motatako aldaketa txikiek proiektuko helburuak ez betetzea eragin bai dezakete. Gainera, proiektuko zuzendariekin uneoro harremanetan egoteak, proiektuko egoera ezagutzea eta ondorioz gaizki ulertuak ekiditea dakar.

¹Code demoaren linka: <https://blockly-demo.appspot.com/static/demos/code/index.html> (2017/06/22 kontsultatuta)

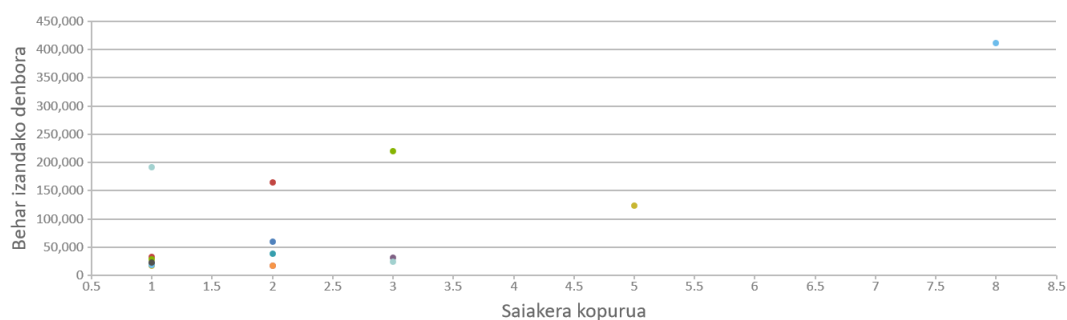
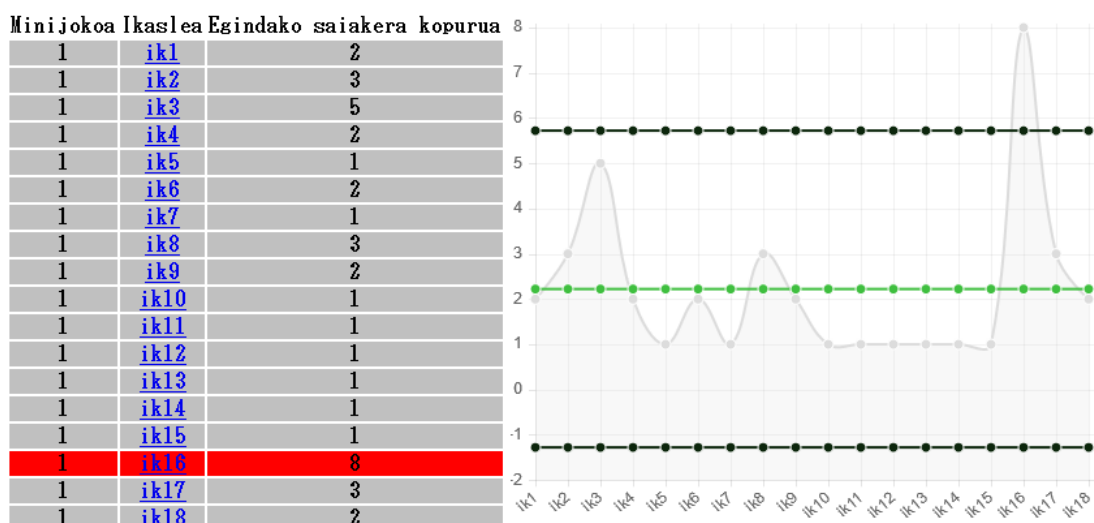
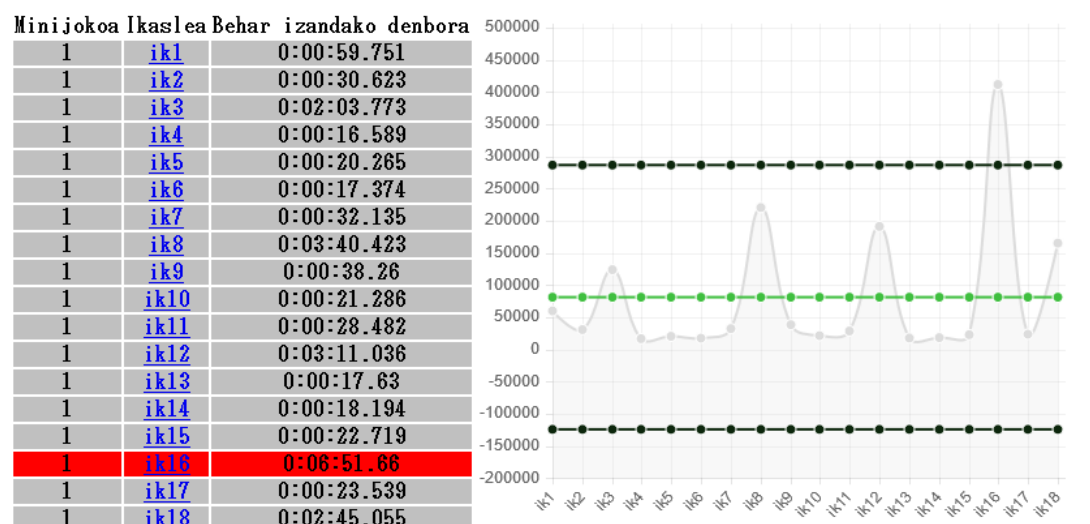
Bibliografia

- [1] JUAN JESÚS VELASCO. *Niños programadores: para qué sirve la enseñanza de programación en las escuelas*. 2014. URL http://www.eldiario.es/turing/Ninos-programadores-ensenanza-programacion-escuelas_0_293970921.html (2017/06/22 konsultatuta)
- [2] GOOGLE. *Google for Education : Blockly*. URL <https://developers.google.com/blockly/> (2017/06/22 konsultatuta)
- [3] GOOGLE. *Blockly*. 2013. URL <https://github.com/google/blockly> (2017/06/22 konsultatuta)
- [4] GOOGLE. *Closure-library*. 2015. URL <https://github.com/google/closure-library> (2017/06/22 konsultatuta)
- [5] GOOGLE. *Blockly games*. 2015. URL <https://github.com/google/blockly-games> (2017/06/22 konsultatuta)
- [6] TUTORIALSPPOINT. *LEARN MYSQL* URL <https://www.tutorialspoint.com/mysql/index.htm> (2017/06/22 konsultatuta)
- [7] *chart.js*. URL <http://www.chartjs.org/> (2017/06/22 konsultatuta)
- [8] *canvas.js*. URL <https://canvasjs.com/> (2017/06/22 konsultatuta)
- [9] *jsPDF*. URL <https://parall.ax/products/jspdf> (2017/06/22 konsultatuta)
- [10] W3SCHOOLS. *AJAX introduction*. URL https://www.w3schools.com/xml/ajax_intro.asp (2017/06/22 konsultatuta)
- [11] *Conociendo GitHub*. URL <http://conociendogithub.readthedocs.io/en/latest/data/introduccion/> (2017/06/22 konsultatuta)
- [12] W3SCHOOLS. *CSS3 Animations*. URL https://www.w3schools.com/css/css3_animations.asp (2017/06/22 konsultatuta)
- [13] JUAN ETXEBERRIA MURGIONDO. *Estatistika Aplikatua: Teoria eta Praktika*. 2011.
- [14] JOSÉ ÁNGEL MARTÍNEZ USERO ETA PABLO LARA NAVARRA. *La accesibilidad de los contenidos web*. Barcelona. 2006.

- [15] GOOGLE. *Blockly accessible*. 2014. URL <https://github.com/google/blockly/tree/master/accessible> (2017/06/22 kontsultatuta)

A Laginetik lortutako emaitzak

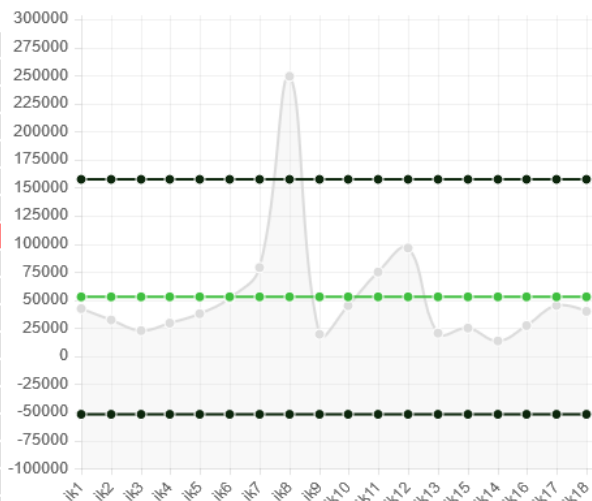
A. A I Minijokoko emaitzak



A. B II Minijokoko emaitzak

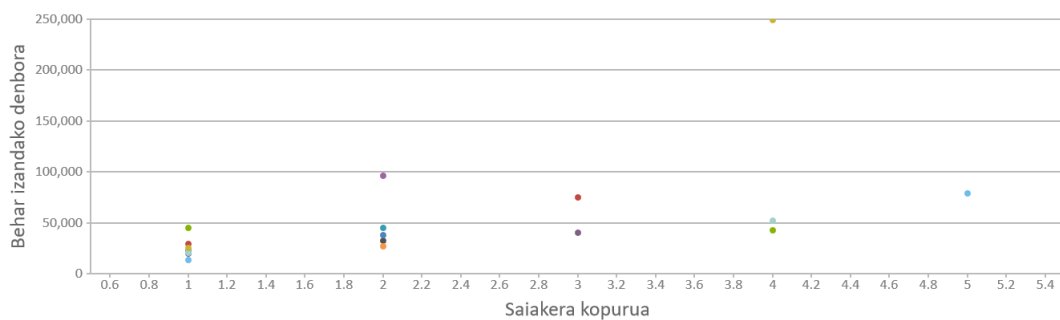
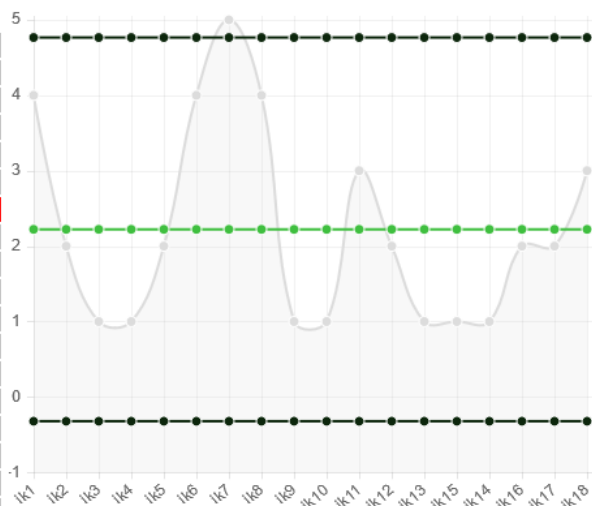
Minijokoa Ikaslea Behar izandako denbora

2	ik1	0:00:42.327
2	ik2	0:00:32.175
2	ik3	0:00:22.729
2	ik4	0:00:29.42
2	ik5	0:00:37.807
2	ik6	0:00:51.664
2	ik7	0:01:18.855
2	ik8	0:04:09.19
2	ik9	0:00:19.478
2	ik10	0:00:44.922
2	ik11	0:01:14.898
2	ik12	0:01:36.27
2	ik13	0:00:20.493
2	ik15	0:00:24.986
2	ik14	0:00:13.521
2	ik16	0:00:27.207
2	ik17	0:00:45.11
2	ik18	0:00:39.879



Minijokoa Ikaslea Egindako saiakera kopurua

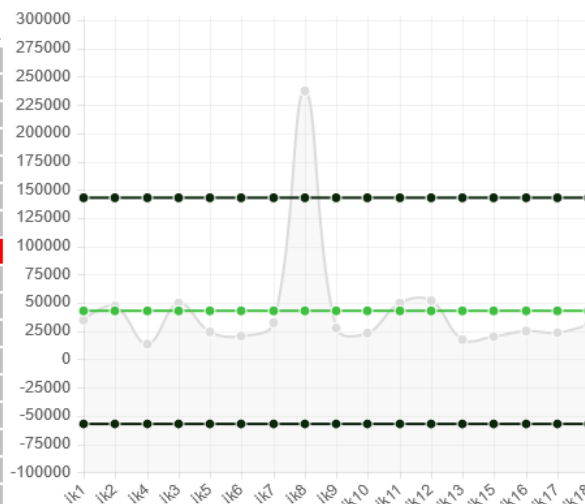
2	ik1	4
2	ik2	2
2	ik3	1
2	ik4	1
2	ik5	2
2	ik6	4
2	ik7	5
2	ik8	4
2	ik9	1
2	ik10	1
2	ik11	3
2	ik12	2
2	ik13	1
2	ik15	1
2	ik14	1
2	ik16	2
2	ik17	2
2	ik18	3



A. C III Minijokoko emaitzak

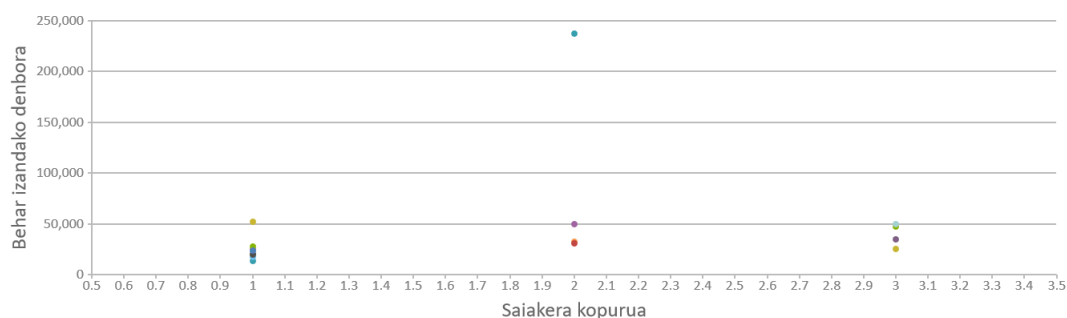
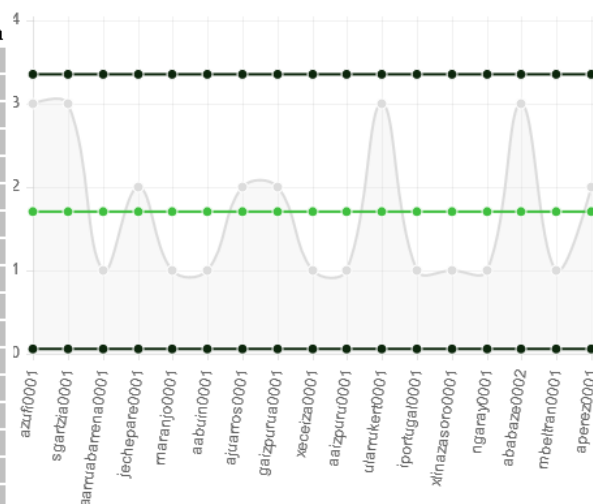
Minijokoa Ikaslea Behar izandako denbora

3	ik1	0:00:34.692
3	ik2	0:00:47.022
3	ik4	0:00:13.398
3	ik3	0:00:49.63
3	ik5	0:00:24.254
3	ik6	0:00:20.494
3	ik7	0:00:32.394
3	ik8	0:03:57.34
3	ik9	0:00:27.611
3	ik10	0:00:23.152
3	ik11	0:00:49.711
3	ik12	0:00:51.734
3	ik13	0:00:17.438
3	ik15	0:00:20.049
3	ik16	0:00:24.898
3	ik17	0:00:23.505
3	ik18	0:00:30.662



Minijokoa Ikaslea Egindako saiakera kopurua

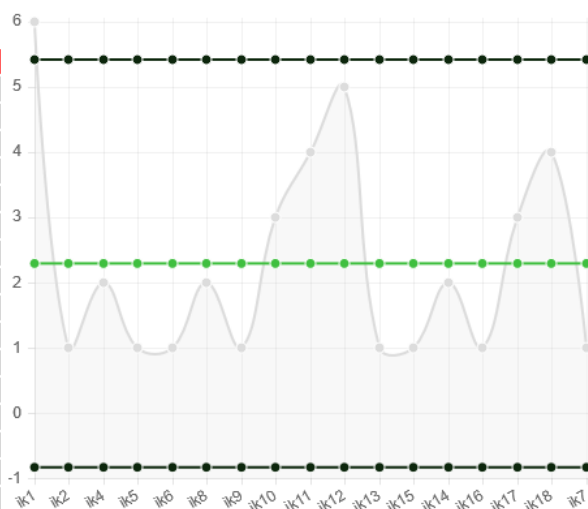
3	azufi0001	3
3	sgartzia0001	3
3	aarruabarrena0001	1
3	jechepare0001	2
3	rnaranjo0001	1
3	aabuin0001	1
3	ajuarros0001	2
3	gaizpuru0001	2
3	xeceiza0001	1
3	aaizpuru0001	1
3	ularrukert0001	3
3	iportugal0001	1
3	xlinazasoro0001	1
3	ngaray0001	1
3	ababaze0002	3
3	mbeltran0001	1
3	aperez0001	2



A. D IV Minijokoko emaitzak

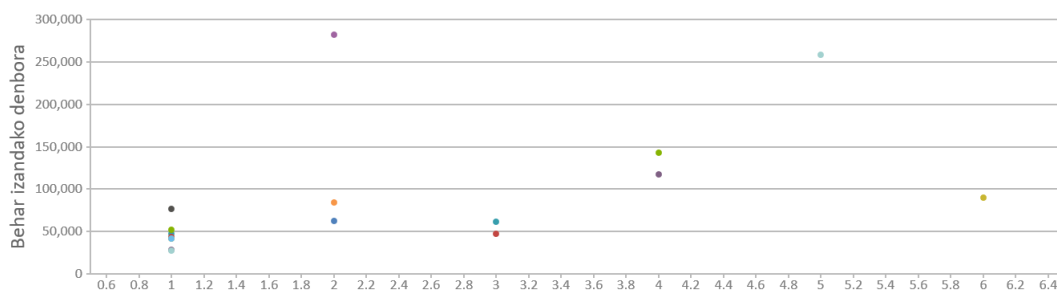
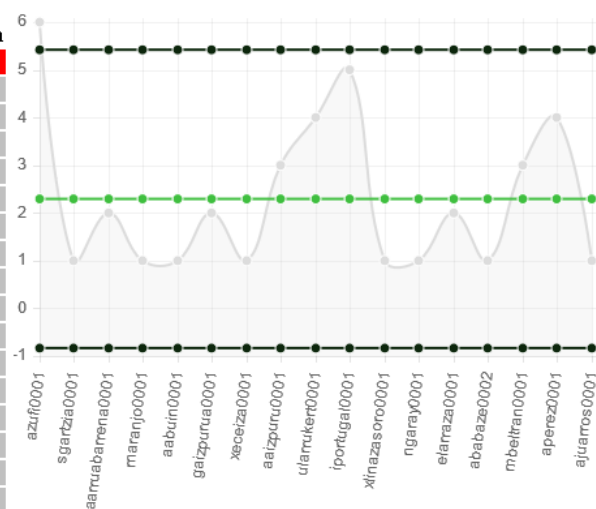
Minijokoa Ikaslea Egindako saiakera kopurua

4	ik1	6
4	ik2	1
4	ik4	2
4	ik5	1
4	ik6	1
4	ik8	2
4	ik9	1
4	ik10	3
4	ik11	4
4	ik12	5
4	ik13	1
4	ik15	1
4	ik14	2
4	ik16	1
4	ik17	3
4	ik18	4
4	ik7	1

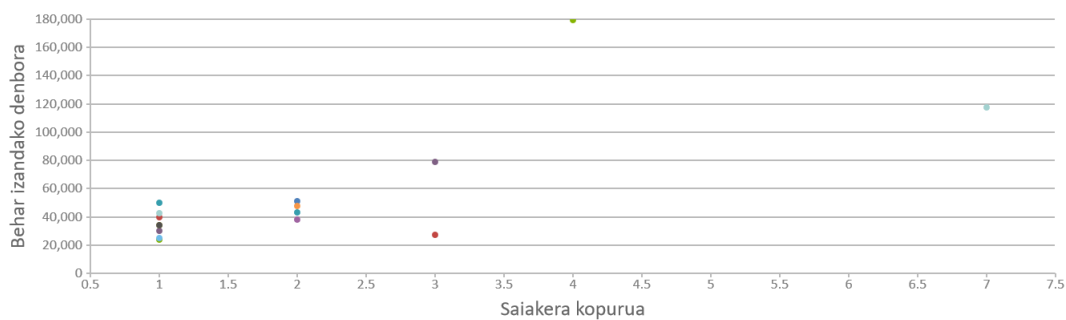
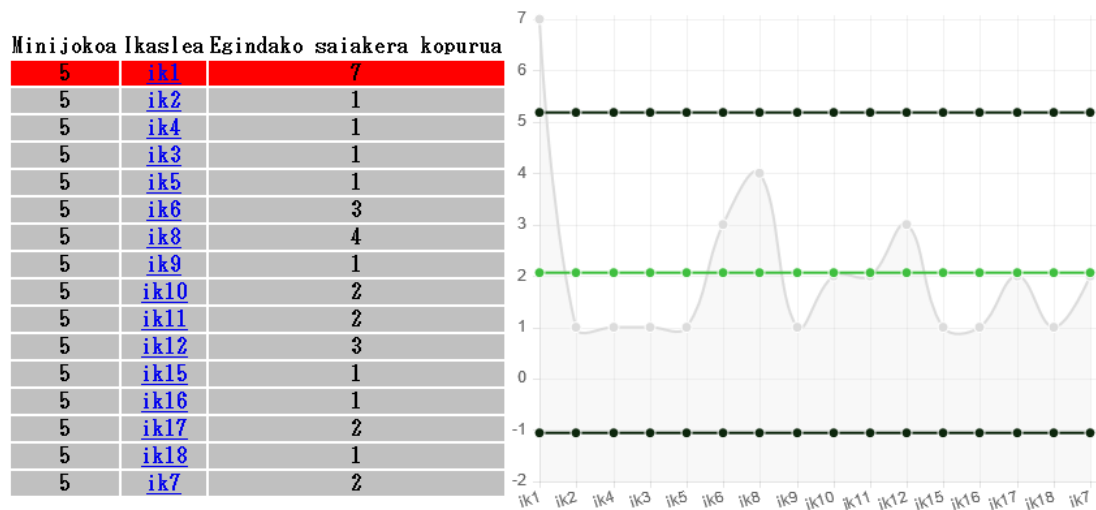
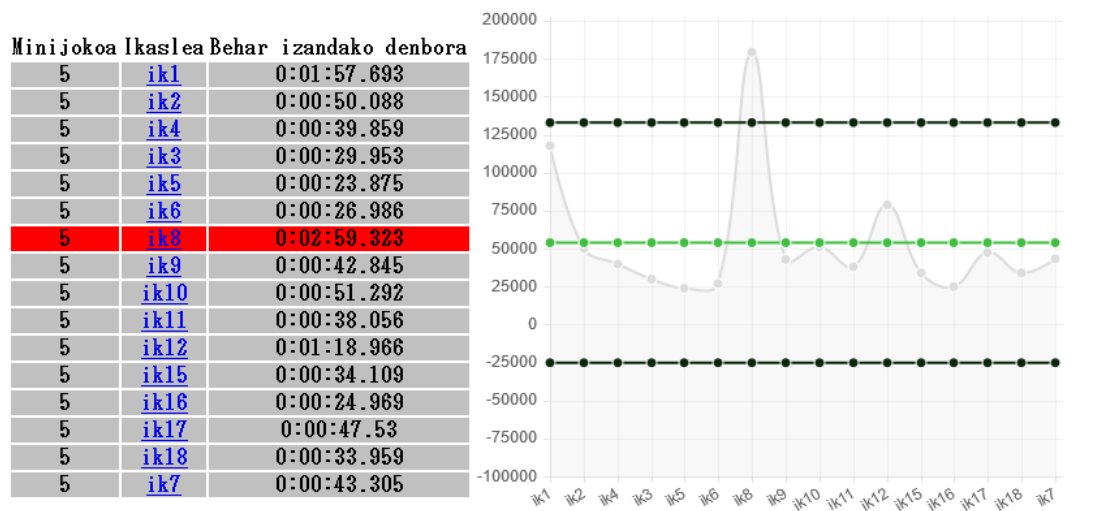


Minijokoa Ikaslea Egindako saiakera kopurua

4	azufi0001	6
4	sgartzia0001	1
4	aarruabarrena0001	2
4	rmaranjo0001	1
4	aabin0001	1
4	gaizpurua0001	2
4	xeceiza0001	1
4	aaiizpuru0001	3
4	ularrukert0001	4
4	iportugal0001	5
4	xlinazasoro0001	1
4	ngaray0001	1
4	elarraza0001	2
4	ababaze0002	1
4	mbeltran0001	3
4	aperez0001	4
4	ajuarros0001	1



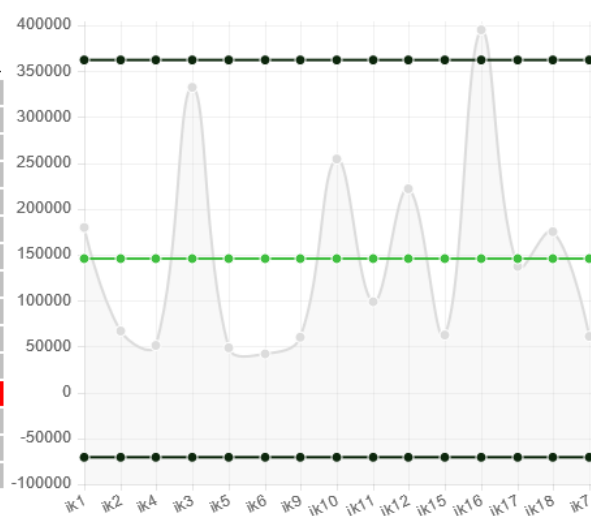
A. E V Minijokoko emaitzak



A. F VI Minijokoko emaitzak

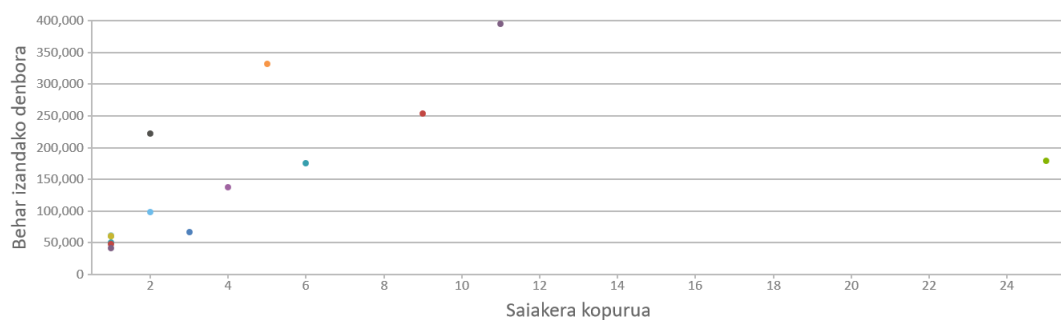
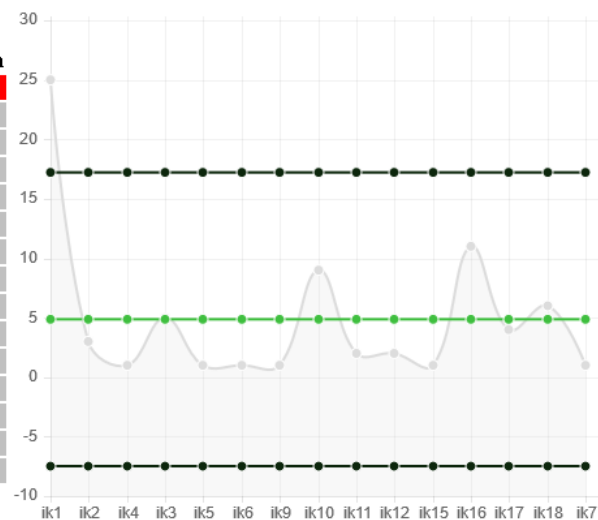
Minijokoa Ikaslea Behar izandako denbora

6	ik1	0:02:59.589
6	ik2	0:01:06.831
6	ik4	0:00:51.101
6	ik3	0:05:32.33
6	ik5	0:00:48.516
6	ik6	0:00:41.87
6	ik9	0:01:00.028
6	ik10	0:04:14.237
6	ik11	0:01:38.683
6	ik12	0:03:41.768
6	ik15	0:01:02.443
6	ik16	0:06:35.006
6	ik17	0:02:17.091
6	ik18	0:02:55.065
6	ik7	0:01:00.973

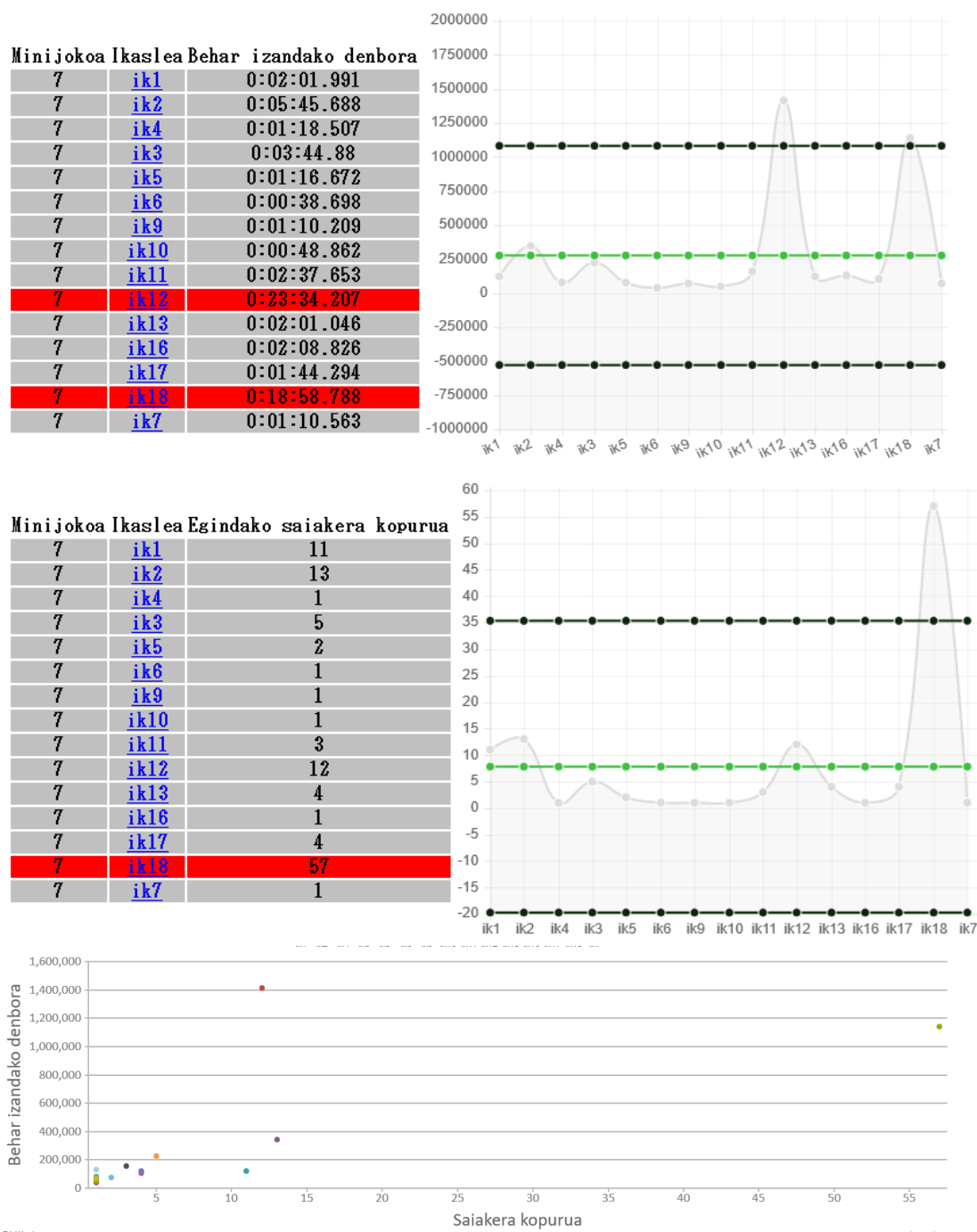


Minijokoa Ikaslea Egindako saiakera kopurua

6	ik1	25
6	ik2	3
6	ik4	1
6	ik3	5
6	ik5	1
6	ik6	1
6	ik9	1
6	ik10	9
6	ik11	2
6	ik12	2
6	ik15	1
6	ik16	11
6	ik17	4
6	ik18	6
6	ik7	1



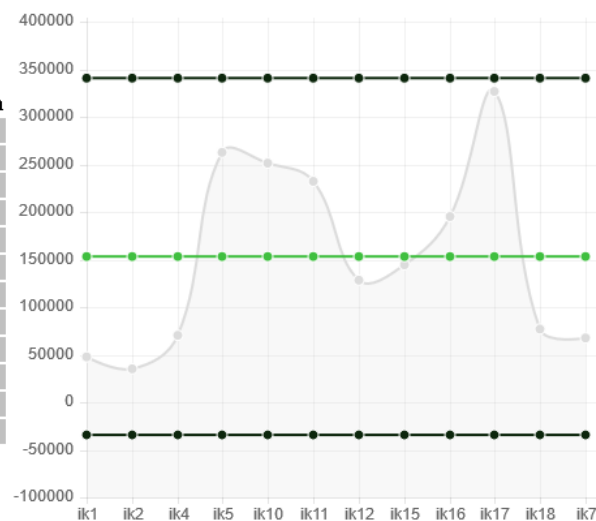
A. G VII Minijokoko emaitzak



A. H VIII Minijokoko emaitzak

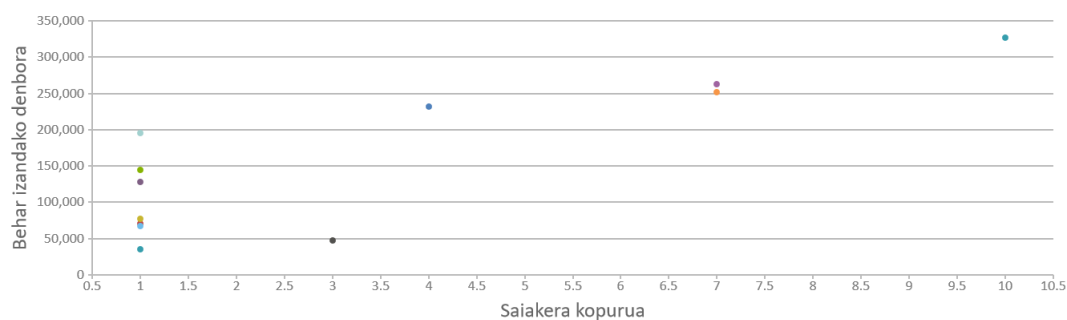
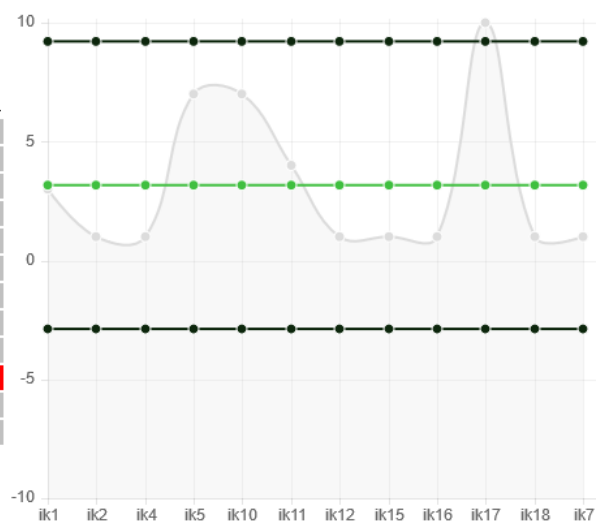
Minijokoa Ikaslea Behar izandako denbora

8	ik1	0:00:47.692
8	ik2	0:00:35.087
8	ik4	0:01:10.332
8	ik5	0:04:22.577
8	ik10	0:04:11.3
8	ik11	0:03:52.165
8	ik12	0:02:08.288
8	ik15	0:02:24.423
8	ik16	0:03:15.14
8	ik17	0:05:26.647
8	ik18	0:01:16.925
8	ik7	0:01:07.53



Minijokoa Ikaslea Egindako saiakera kopurua

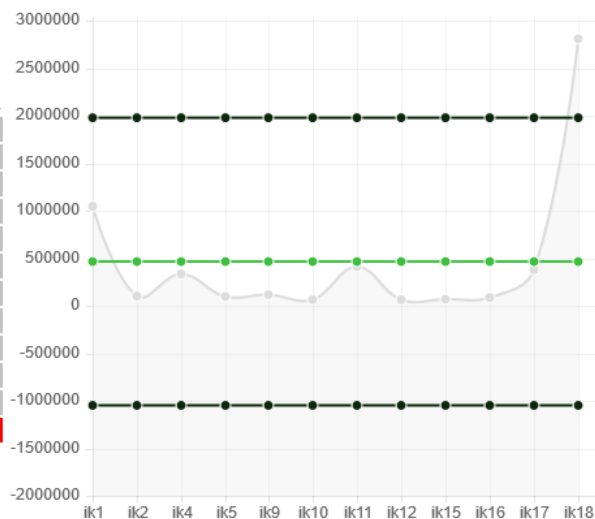
8	ik1	3
8	ik2	1
8	ik4	1
8	ik5	7
8	ik10	7
8	ik11	4
8	ik12	1
8	ik15	1
8	ik16	1
8	ik17	10
8	ik18	1
8	ik7	1



A. I IX Minijokoko emaitzak

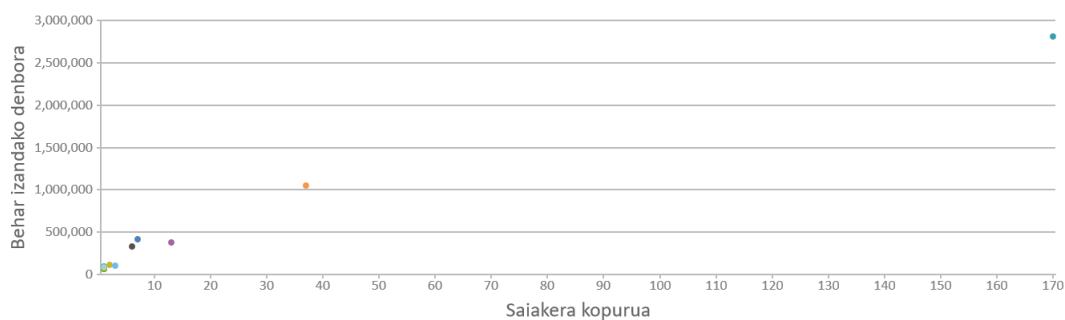
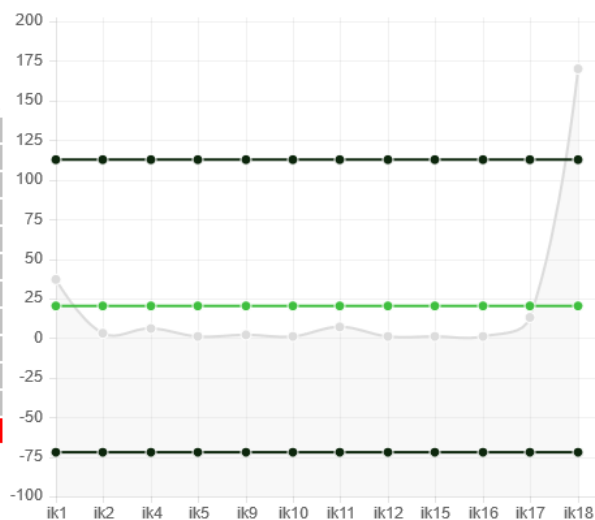
Minijokoa Ikaslea Behar izandako denbora

9	ik1	0:17:27.001
9	ik2	0:01:41.508
9	ik4	0:05:33.661
9	ik5	0:01:37.317
9	ik9	0:01:56.076
9	ik10	0:01:04.607
9	ik11	0:06:51.997
9	ik12	0:01:04.064
9	ik15	0:01:08.898
9	ik16	0:01:26.204
9	ik17	0:06:20.996
9	ik18	0:46:48.247



Minijokoa Ikaslea Egindako saiakera kopurua

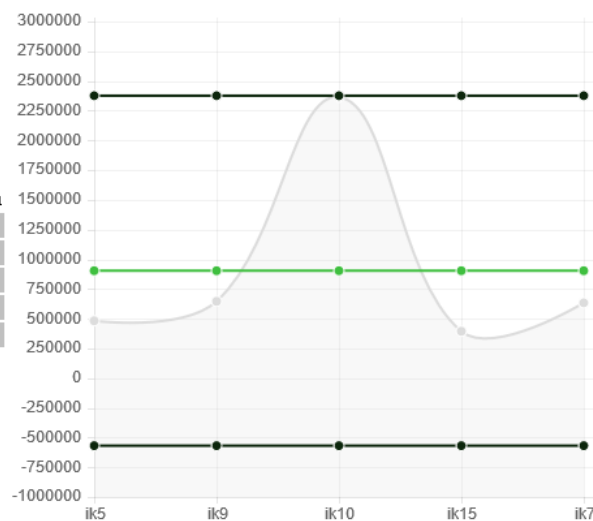
9	ik1	37
9	ik2	3
9	ik4	6
9	ik5	1
9	ik9	2
9	ik10	1
9	ik11	7
9	ik12	1
9	ik15	1
9	ik16	1
9	ik17	13
9	ik18	170



A. J X Minijokoko emaitzak

Minijokoa Ikaslea Behar izandako denbora

10	ik5	0:08:02.362
10	ik9	0:10:46.002
10	ik10	0:39:24.036
10	ik15	0:06:34.086
10	ik7	0:10:33.636



Minijokoa Ikaslea Egindako saiakera kopurua

10	ik5	6
10	ik9	8
10	ik10	25
10	ik15	1
10	ik7	5

