

eman ta zabal zazu



Universidad del País Vasco Euskal Herriko Unibertsitatea

MOODLEN INTEGRATUTAKO MEDIATEKA BATEN KUDEAKETARAKO MODULU BATEN GARAPENA

**SISTEMEN INFORMATIKAN INGENIARITZA TEKNIKOA
KARRERA BUKAERAKO PROIEKTUA**

Ikaslea: Ander Perez Nava
Zuzendaria: Iñaki Alegria Loinaz

ESKERRAK

Lerro hauen bitartez eskerrak eman nahi nizkieke proiektu honetan modu zuzenean edo ahal izan duten bezala lagundu didaten pertsona guztiei.

Lehenik eta behin, familia eta lagunak eskertu nahiko nituzke, haien babesik gabe ez bainuke ezer egingo.

Bestalde, proiektu honen garapenean eta kudeaketan hainbeste lagundu nauten Felix Elejosteri eta Iñaki Alegriari ustekabeen nire proiektuaren zuzendari izatea onartzeagatik.

Beren ideien, laguntza eta motibaziorik gabe proiektua eta garatutako aplikazio hau ez lirateke orain direna izango.

AURKIBIDEA

1. SARRERA.....	1
2. PHD.....	2
2.1 Helburua.....	2
2.2 Atazen zerrenda.....	2
2.3 Arriskuak.....	4
3. AURREKARIAK ETA TEKNOLOGIA.....	6
3.1 Garapenerako tresnak eta teknologia.....	8
3.1.1 Moodle.....	8
3.1.2 Moodle-en programatzeko baliabideak.....	11
3.1.3 phpMyAdmin.....	13
3.1.4 Apache.....	14
3.1.5 Besteak.....	14
4. ANALISIA.....	15
4.1 Erabilpen-kasuen eredia.....	17
4.1.1 Objektuak sartu erabilpen-kasua.....	18
4.1.2 Objektuak bilatu erabilpen-kasua.....	18
4.1.3 Objektuak aldatu erabilpen-kasua.....	18
4.1.4 Objektuak ezabatu erabilpen-kasua.....	19
5. DISEINUA.....	19
5.1 Datu-basearen diseinua.....	19
5.2 Liburutegiaren kontsulta.....	21
5.3 Erregistroaren gehiketa.....	23
5.4 Erregistroaren aldaketa.....	23
5.5 Erregistroaren ezabaketa.....	24
5.6 Katalogoaren kudeaketa.....	26
6. INPLEMENTAZIOA.....	27
6.1 Blokearen definizioa.....	27
6.2 Datu-basearen definizioa.....	30
6.3 Aplikazioen garapena.....	37
6.3.1 Liburutegiaren kontsulta.....	37
6.3.2 Erregistroaren gehiketa.....	40
6.3.3 Erregistroaren aldaketa.....	41
6.3.4 Erregistroaren ezabaketa.....	42
6.3.5 Katalogoaren kudeaketa.....	43
6.3.6 Hizkuntza fitxategiak.....	43
6.3.7 Baimenen definizioa.....	44
6.4 EMAITZAK.....	46
7. ONDORIOAK.....	49
8. BIBLIOGRAFIA.....	50
9. ERANSKINA.....	51

Ilustrazioen indizea

Ilustrazioa 1: proiektuaren Gantt diagrama.....	3
Ilustrazioa 2: LDE diagrama.....	5
Ilustrazioa 3: La Salle Berrozpe ikastetxeko moodle-en pantailaren adibidea.....	7
Ilustrazioa 4: Erabilpen-kasuen Eredua.....	17
Ilustrazioa 5: E/E diagrama.....	20
Ilustrazioa 6: bilaketa sinplea pantailaren diseinua.....	21
Ilustrazioa 7: bilaketa aurreratua pantailaren diseinua.....	22
Ilustrazioa 8: bilaketaren emaitza pantailaren diseinua.....	22
Ilustrazioa 9: erregistroa gehitu pantailaren diseinua.....	23
Ilustrazioa 10: aldatzeko erregistroaren bilaketa pantailaren diseinua.....	24
Ilustrazioa 11: erregistroa aldatu pantailaren diseinua.....	25
Ilustrazioa 12: ezabatzeko erregistroaren bilaketa pantailaren diseinua.....	25
Ilustrazioa 13: erregistroa ezabatu pantailaren diseinua.....	26
Ilustrazioa 14: katalogoa kudeatu pantailaren diseinua.....	26
Ilustrazioa 15: blokea sortzen duen init() funtzioa.....	27
Ilustrazioa 16: blokearen edukia definitzen duen get_content() funtzioa.....	28
Ilustrazioa 17: blokea editatzeko aukera ematen duen allow_config() funtzioa.....	29
Ilustrazioa 18: zer editatu azaltzen duen config_instance.html fitxategiaren kodea.....	29
Ilustrazioa 19: titulua aldatzeko specialization metodoa.....	29
Ilustrazioa 20: liburutegi blokea moodle-eko orri nagusian instalatuta.....	30
Ilustrazioa 21: XMLDB editorearekin sortutako mdl_block_liburutegi_objektua taula.....	31
Ilustrazioa 22: mdl_block_liburutegi_egilea taularen definizioa.....	32
Ilustrazioa 23: mdl_block_liburutegi_egile_obj taularen definizioa.....	32
Ilustrazioa 24: mdl_block_liburutegi_etapa taularen definizioa.....	33
Ilustrazioa 25: mdl_block_liburutegi_etapa_obj taularen definizioa.....	33
Ilustrazioa 26: mdl_block_liburutegi_formatua taularen definizioa.....	34
Ilustrazioa 27: mdl_block_liburutegi_formatu_obj taularen definizioa.....	35
Ilustrazioa 28: block_liburutegi_gaia taularen definizioa.....	35
Ilustrazioa 29: mdl_block_liburutegi_gaia_obj taularen definizioa.....	36
Ilustrazioa 30: XMLDB editorearekin egindako blokearen taulak.....	36
Ilustrazioa 31: bilaketa aurreratua erakutsi eta ezkutatzeko erabiliko diren funtzioak.....	38
Ilustrazioa 32: erregistroak bilatzeko baimenen definizioa.....	44
Ilustrazioa 33: erregistroak gehitzeko baimenen definizioa.....	44
Ilustrazioa 34: erregistroak aldatzeko baimenen definizioa.....	45
Ilustrazioa 35: erregistroak ezabatzeko baimenen definizioa.....	45
Ilustrazioa 36: katalogoa kudeatzeko baimenen definizioa.....	46
Ilustrazioa 37: baimenak aldatzeko pantaila.....	46
Ilustrazioa 38: blokea ondo instalatzerakoan lortutako pantaila.....	47

1. SARRERA

Gaur egun aplikazio informatikoak arlo guztietan aurkitzea guztiz ohikoa da. Gizartean ohituta gaude jada mota guztietako aplikazioak ikustera, testu-editore sinple batetik fisika simuladore konplexu bateraino. Honekin batera, etxetik, fakultatetik edo lantegitik nahi dugun informazio guztia jasotzera ohitzen ari gara, lekuz mugitu gabe edota telefonoz beste inorekin kontaktuan jarri gabe. Irakaskuntzaren mundua ez da atzera geratzen, eta ikasleak teknologia berrietan geroz eta trebatuagoak daudenez, hauentzako ordenagailu-pantaila baten bitartez edozein motatako informazioa jasotzea oso gauza arrunta da.

Hau gauza positiboa izan arren, mundu digital honek eragin zuzena du, geroz eta ahaztuagoak dauden liburu fisikoetan adibidez. Honen ondorioz liburutegiek ere teknologia hauen etorrera pairatzen dute eta ondorioz liburutegien bisitari eta erabiltzaile kopurua gutxitu da. Liburutegi handietan, udal liburutegiak adibidez, efektu hauek ez dira hain nabarmenak, baina bai ikastetxeetako liburutegietan adibidez.

Hau guztiagatik Andoaingo La Salle Berrozpe ikastetxean, liburutegiko materialaren erabilera bultzatzeko bideren bat bilatzeko erabakia hartu zuten.

Teknologia berriak gai desberdinetara zuzenduta daude eta irakaskuntza-mundua hauetariko bat da. Bai irakaskuntza baita ikasketa lanak errazteko asmoz aplikazio desberdin asko garatu dira eta La Salle Berrozpe ikastetxean *Moodle* izeneko ikasteko ingurune birtuala integratu zuten. Ikasleei zuzendutako ikastaro desberdinak eskaintzen dituzte tresna honen bidez.

Ikastetxean bertako liburutegia bultzatzeko ideia eta *Moodle* plataformaren erabilera aprobetxatuz, liburutegiaren kudeaketa bertan integratzea erabaki zuten, *Moodle*-en eskaintzen dituzten ikastaroetara zuzendu ahal izateko. Horrela, bertako liburutegian dauden baliabideez gain, sarean egon daitezkeen eta *Moodle*-en dagoen edozein ikastaroren baterako interesgarriak izan daitezkeen dokumentuak ere eskuragarri izateko.

2. PHD

2.1 Helburua

Karrera amaierako proiektu honen helburu nagusia *Moodle* ikasteko ingurune birtualean liburutegi baten erregistroa kudeatzeko behar diren funtzioak izango dituen aplikazio bat inplementatzea da. Aplikazio honen bidez ikastetxeko liburutegira hurbiltzeko aukera emango zaie *Moodle*-eko erabiltzaileak diren ikasleentzat. Honen ondorioz, materiala era egokiagoan zuzenduko zaie, eta ikasleek material desberdinen eskaintza eskuragarri izango dute. Gainera, plataforma birtuala denez, fisikoki hurbil izan gabe sarean egon daitekeen material desberdina ere liburutegiko erregistroan gorde daiteke, *Moodle*-en dauden ikastaro desberdinetara gomendatuz.

Aplikazio hau garatzeko LAMP azpiegitura erabiliko dugu, hau da *Linux* sistema eragilea, *Apache* web-zerbitzaria, *MySQL* datu-baseak kudeatzeko sistema eta *PHP* programazio lengoaia. *Windows* erabiltzeko aukera ere badago, baina software librea erabili dezakegunez hau erabiliko dugu.

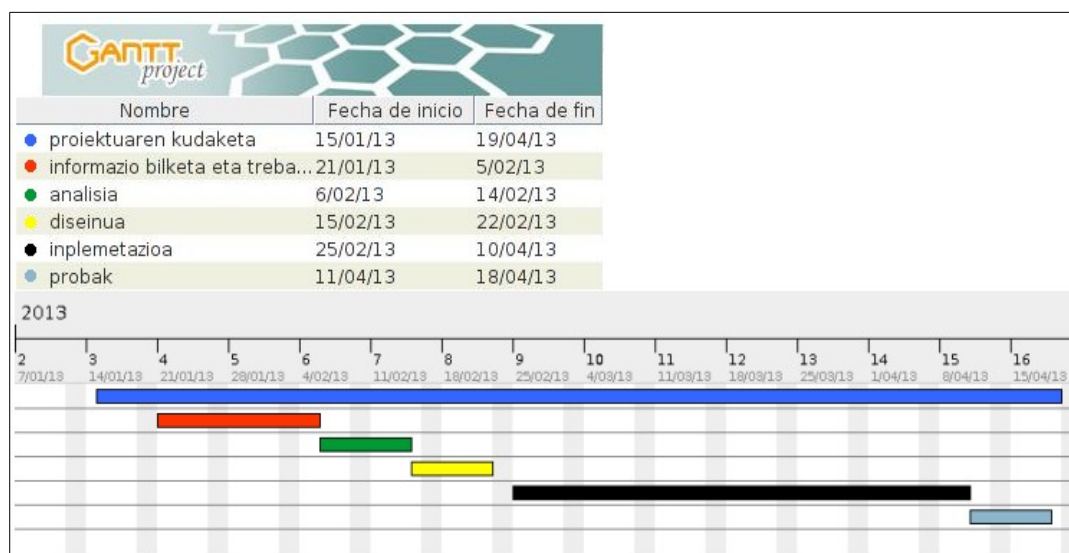
Ikusten den bezala, pertsonalki garrantzitsuak diren zeharkako helburu batzuk ere betetzen dira. Helburu hauen artean informatika munduko proiektu baten garapena nolakoa den ezagutuko dut eta proiektua La Salle Berrozpe ikastetxean garatuko dudanez enpresa mailan dagoen antolaketa ere ikasiko dut.

Honetaz aparte, erabiliko ditudan tresnen erabileran trebatzeko helburua ere beteko dut. *PHP* programazio-lengoaia eta *HTML* lengoaietan sakondu behar izango dudalako eta niretzako berria den *Moodle* plataforma erabiltzaile, administratzaile eta garatzaile moduan ezagutzeko aukera ere emango dit proiektu honek.

2.2 Atazen zerrenda

Proiektu honek ataza nagusi hauek ditu. Ataza hauek proiektuaren bizi-zikloa irudikatzen dute. Eta horiek beste azpiataza batzuetan zabaltzen dira:

- Proiektuaren kudeaketa: La Salle Berrozpeko arduradunari atal desberdinen aurkezpenak, proiektuaren kontrola eta memoria egitea.
- Informazio bilketa eta trebakuntza: Ataza honetan proiektuarekin erlazioa duen informazio-bilaketa oro sartuko da, baina bilketa hauek multzokatuta jarriz gero, honako azpiatazak aurreikus ditzakegu: ikastetxeko liburutegia ezagutzea batetik eta *PHP* eta *HTML* lengoaietan eta *Moodle*-en sakontzea bestetik.
- Analisia: kasu honetan sistemaren analisiaren baitan egindako lanak izango dira kontua, eta hauek izango dira azpiatazak: proiektuaren ezaugarriak finkatzea, arriskuen analisia eta segurtasun ahuleziak aztertzea.
- Diseinua: aplikazioaren interfazearen eta datu-basearen diseinua egingo da hemen.
- Inplementazioa: aurreko atazan diseinatutako elementu guztien inplementazioak osatuko du ataza hau.
- Probak: aplikazioaren erabilgarritasunaren probak egitea da hemen sartuko den lana, hau da, erroreak bilatzea eta emaitzak helburuak betetzen dituen aztertzea.



Ilustrazioa 1: proiektuaren Gantt diagrama

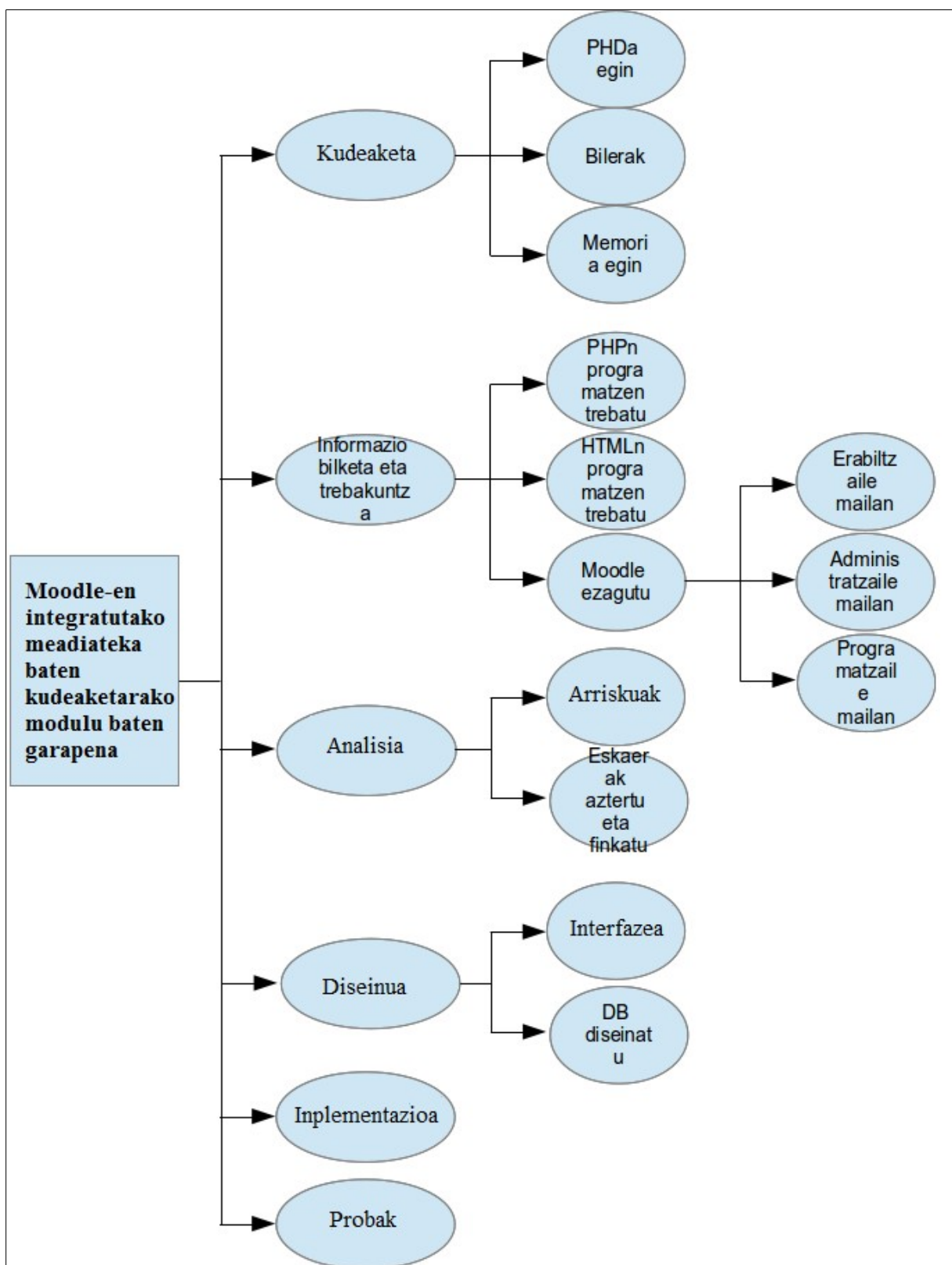
Gantt diagraman ikusten den bezala proiektua urtarrilaren 15ean hasi eta apirilaren 18rako bukatzea espero da, eta ordu gehien hartuko dituen ataza inplementazioa da. Ordu gutxiago baina denboran luzeagoa kudeaketa izango da.

2.3 Arriskuak

Arriskuen plana

Hauek dira proiektu honen garapenean topa ditzakegun eragozpen edo arazo nagusiak eta hauek ekiditeko edota aurre egiteko egin beharreko lanak:

- Informazio galera (digitala): informatika munduan arrisku ohikoa da informazio digitalaren galera. Batzuetan kanpoko arrazoiengatik (makinen arazo teknikoak, hondamen natural baten eraginez, malwarearen ondorioz...) bestetan erabiltzaileen erroreek eraginak (nahigabe borratzea, nahita borratzea geroago erabiliko dela ez ikusiz...). Informazio galera posibleari interneten gordetako segurtasun kopiakin aurre egingo zaio. Segurtasun-kopiak astero egingo dira, azken eguneraketak ongi babestuak izateko. Proiektua garatzean 2 konputagailu erabiliko dira (ikastetxekoa eta etxekoa), beraz askotan bi ordenagailuetan egongo dira lanaren kopiak, baita USB memoria batean ere. Interneten gordetzeko erabilitako web orria bertan behera geratuz gero, beste web zerbitzu bat bilatu beharko da eta bertan egin segurtasun-kopia berriak.
- Ezusteko arazo pertsonalak: proiektua lan indibiduala da, beraz pertsona bakarrak ez badu lan egiten, proiektua gelditu egingo da. Ezusteko arazo pertsonalen bat izanez gero, gaixo jartzea adibidez, galdutako denbora berreskuratu beharko da asteburuetan edo aste tartean planifikatua baino ordu gehiago sartuz.
- Teknologia berrien erabilera: Proiektuan jorratuko diren gai gehienak berriak izango dira niretzat. Horregatik teknologien azterketa eta oinarrizko ikasketak espero baino denbora gehiago har dezake, gainera zaila da hauen ikasketarako beharko den denbora estimatzea. Ikasketa prozesua luzatzen bada, ordu gehiago eman beharko zaizkie ikasketa atazei, eta gainontzeko atazen asteko plangintza berregin beharko da, astean ordu gehiago eskainiz galdutako denbora konpentsatzeko



Ilustrazioa 2: LDE diagrama

3. AURREKARIAK ETA TEKNOLOGIA

Andoaingo La Salle Berrozpe ikastetxean duela urte batzuk hasi ziren teknologia berrien ezarpenarekin. Bai irakasleen lana errazteko baita ikasleek ikasteko baliabide gehiago izateko ere. Prozesu honetan eta denboraren poderioz zerbitzari gela txukun bat lortu arte. Hau pixka bat deskribatzeko dauden zerbitzari guztien azalpen txiki bat emango da.

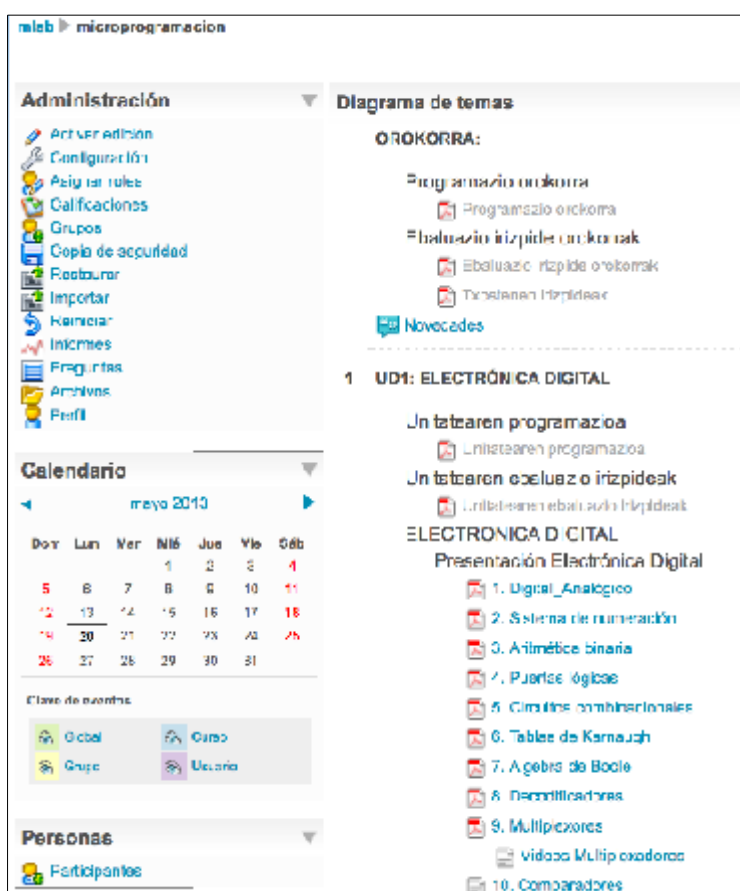
1. Sare guztiaren segurtasuna eta kontrola eramateko eta debekatutako web-orrietan sartzea galarazteko lehenbiziko zerbitzaria firewall-a da.
2. Langileen telefonoak antolatzeko *Asterisk* erabiltzeko beste zerbitzari bat dago, hau IP telefonogunea eratu eta antolatzeko software librea da.
3. Intranetarentzat beste zerbitzari bat dago, lantalde desberdinetan informazioa banatzeko.
4. Diskoentzako beste zerbitzari bat dago, hemen dauden diskoetan eskolaren datuak gordetzeko.
5. Web orria zerbitzatzeko (www.lasalleberrozpe.com) beste zerbitzari desberdin bat erabiltzen da.
6. Posta elektronikoari beste zerbitzari bat dagokio, *zimbra* software libreko posta zerbitzaria erabiliz.
7. Probak egiteko kontroladoreak dituen beste zerbitzari bat dago, hemen probak egiten dira beste zerbitzarietan arazoak ekiditeko.
8. *Hontza* kode irekiko plataformarentzat beste zerbitzari bat dago, zaintza teknologikorako erabiltzen dena.
9. Nahiz eta eskolan erabiltzen den software gehiena librea izan badira lizentzia behar duten aplikazioak ere (*solidworks*, *autocad*...) eta hauek kudeatzeko beste zerbitzari bat dago.
10. Azkeneko zerbitzaria segurtasun-kopiak egiteko erabiltzen dena da.

Ikusten denez teknologia berriak erabiltzen saiatzen dira. Hauen artean *Moodle* plataforma iraultza gehien ekarri duena izan da. Aplikazio honen bidez ikasleei ikastaro ezberdinak eskaintzen hasi ziren eta orain euskarri garrantzitsu bat bihurtu da ikastetxearen egituraren barruan (3. irudia). Irakasleek notak *Moodle*-en bidez argitaratzen dituzte, ikasleei lanak *Moodle*-en bitartez bidaltzen zaizkie...

Honez gain, sarreran aipatu bezala euskarri digitalean dauden baliabideak leku asko

kentzen ari die euskarri fisikoan daudenei. Honen ondorioz ikastetxeko liburutegira doan jende gehiena ez doa han dagoen materiala erabiltzera, bertan lan desberdinak egitera edo ordenagailu eramangarriekin lan egitera doaz bertako wifi konexioa erabiliz.

Moodle-en egindako ikastaro ezberdinak emandako esperientziaren poderioz, ikastetxeko arduradunek, ikasleei material gehiago eskaintzeko aukera falta zela ikusi zuten, eta eskolako liburutegiaren erabilpena bultzatzeko asmoz bertako materialaren eskaintza digital bat egitea erabaki zuten. Gainera, honela, liburutegiko materialaz aparte, sarean dagoen materiala eskaini daiteke Moodle-en eskaintako ikastaroetara zuzenduta. Azken aukera honen bitartez ikasleek ez dute denborarik galdu beharko haien lanetarako erabilgarria den informazioa bilatzen, irakasleek zuzenean gomendatu ahal izango dute interesgarria eta erabilgarria izango den informazioa. Beraz, proiektu honen bidez ikastetxeko liburutegia bultzatuko da, ikasleei lanak errazteko aukera emango zaie, sarean dagoen informazioa eskainiz eta Moodle-en integratuz aplikazio honen erabilera ere bultzatuko da.



Ilustrazioa 3: La Salle Berrozpe ikastetxeko moodle-en pantailaren adibidea

Behar berri hauek asetzeko, ikastetxeak praktiketan ikasleak hartzeko erabakia hartu zuen, horrela ikasleei lanean ikasteko aukera emanaz eta sortzen diren arazo edo aukera berrientzako irtenbideak lortuz. Informatika munduan ikasketak ikastetxean bukatu dituzten ikasleentzako lan-poltsa baten aplikazioa garatu da baita irakasleen zaintzak antolatzeko beste aplikazio bat ere. Elektronikaren hurrengo ikastaroan egingo diren praktikak egiteko ere ikasleak hartu dira baita idazkaritza lanetan laguntzeko ere.

3.1 Garapenerako tresnak eta teknologia

Ondorengo lerroetan proiektuan zehar erabili behar izan ditudan tresnak zerrendatuko ditut eta bakoitzaren ezaugarri gutxi batzuk. Software librean erabili izan da proiektu guztiaren garapenean, honetarako *Ubuntu 12.10* sistema eragilea duen makina erabiliz.

3.1.1 Moodle

Modular Object-Oriented Dynamic Learning Environment delakoaren laburdura da (Objektuetara bideratutako Ikas-Ingurune dinamikoa eta Modularra). Proiektuaren ardatzetako bat da, irakaskuntza kudeatzeko sistema honetara bideratuta dagoelako lan honen emaitza. Online moldeko ikastaroak sortzeko programa-multzoa da, *PHP* dabilen edozein ordenagailutan funtzionatzen du, eta hainbat datu-base mota onartzen ditu (*MySQL*, *PostgreSQL*...).

Moodle Martin Dougiamas australiarrek sortu zuen 2001ean. Harrez geroztik ikaragarriko arrakasta eta zabalkundea izan du mundu osoko hezkuntza-erakundearen artean. 2006ko abenduan *Moodle*-en gune ofizialak 150.000 erabiltzaile zituen, 75 hizkuntzatarako itzulita dago (euskarara ere bai), mundu osoko 160 herritan ari dira erabiltzen, 19.091 gune baino gehiagotan.

Moodle-ek pedagogia konstruktibo soziala bultzatzen du. Bere arkitektura eta aplikazioengatik online klaseentzako egokia da, baina baita ere klasean irakasleak irakatsitakoa hobetzeko eta sakontzeko. Nabigatzaileak duen interfazea erabiltzeko erraza da. Instalazioa ere erraza da, *PHP* plataforma eta datu-base batera sarrera bakarrik eskatzen duelarik. Lehen aipatu den bezala, datu-baseen zerbitzari ezberdinak erabiltzeko aukera ere eskaintzen du. Gainera, plataforma guztian zehar segurtasuna oso kontrolatua dago: inkesta guztiak kontrolatzen dira, cookie-ak zifratzen dira... Testuak sartzeko leku gehienek *HTML* editorea erabiltzen dute,

bertan idaztea erraza bihurtzen duena.

Horrez gain, *Moodle*-en ezaugarri nagusienetako bat modularitatea da, hau da, plataforman sortzen diren ikastaro guztietan hainbat elementu gaineratu, ezabatu, aldatu edo egokitu ahal direla, beti ere tutoreak dituen beharren arabera (gai honetan gehiago sakontzeko [1]).

Hiru modulu mota nagusi ditu:

- Baliabideak

Nolabait esateko, baliabideak edukiak dira: fitxategiak, web orriak, direktorioak, kanpo loturak...

- Jarduera-moduluak

Ikas-materialak dira eta nolabaiteko interaktibitatea gaineratzen dute: foroak, txatak, wikiak, blogak, glosategiak, datu-baseak, zereginak, tailerrak...

- Blokeak

Ikastaroaren orri nagusiak ezkerreko eta eskuineko aldeetan blokeak izaten ditu. Horiek informazioa edo beste funtzionalitate batzuk eskaintzen dituzte irakasleentzat nahiz ikasleentzat: egutegia, kalifikazio-liburua, duela gutxiko aktibitatea...

Blokeak eta moduluak ondo bereizteko era erraz bat dago, orokorrean *Moodle* hiru zutabeetan bereizi daitekeela esan daiteke. Erdiko zutabeetan moduluak kokatzen dira eta alboko zutabeetan berriz blokeak. Moduluak baliabide eta jardueretan banatu daitezke, baliabideak auto-ikasketarako erabiltzen dira gehien bat, ikasketa-prozesu pasiboa (glosategiak adibidez), eta jarduerak aldiz lankidetzan eta interakzioan laguntzen dute (foroak adibidez). Blokeak irakaskuntza-ikasketa prozesuan laguntza emateko dauden baliabideak dira (foro bilatzailea adibidez).

Arkitekturari dagokionez, *Unix*, *Linux*, *FreeBSD*, *Windows*, *Mac OS X*, *Net Ware* eta *PHP*-a erabiltzen duten beste sistemetan inongo aldaketarik gabe erabili daiteke.

Instalazioa ez da beste edukiak kudeatzeko sistemen desberdina. Instalazioa egin nahi den modua kontuan hartu behar da: modu integratuan egin nahi bada (aplikazioen zerbitzariak eta datu-baseak zerbitzari fisiko bakar batean) edota modu sailkatuan egin nahi den (aplikazioen zerbitzaria eta datu-baseak zerbitzari fisiko desberdinetan). Bigarren aukera da gomendagarriena, baina edukiak kudeatzeko sistema honi buruz soilik ikasi egin nahi baduzu lehenengo aukera erabiliko da.

Lehenengo aukera LAMP bidez eta *PHP*, *MySQL* eta *Apache*-k osatutako paketearen bidez egin daiteke, instalatzeko beharrezkoak direnak. Pakete hau bai *Linux* eta bai *Windows*-entzako (LAMP) erabili daiteke. Hau instalatu ondoren, *Moodle* paketea zabaltzea eta instalatzea bakarrik geratzen da.

Programa itzultzea ere posible da. *Moodle*-en hizkuntza-pakete guztiak *lang* direktorioan daude, hizkuntzaren laburdurak adierazitako direktorio ezberdinetan hizkuntza desberdinak gordeta daudelarik. Nahi duzun hizkuntza *Moodle*-en helbidetik jaitsi daiteke. Paketea zabaltzerakoan, *moodledata/lang* zerbitzariko karpeta kopiatu behar da. *Moodle*-en orri nagusira sartzen zaren hurrengoan kopiatu den hizkuntza aukeratzeko aukera agertuko da. Kopiatu den hizkuntza betiko hizkuntza izatea nahi bada honakoa egin behar da. Administrazioaileak hizkuntza menuko hizkuntza ezarpenetan dagoen berezko hizkuntzaren eremuan, instalatu berri den hizkuntza aukeratu behar da.

Esan bezala, *Moodle*-ek planteatzen duena hezkuntza konstruktibo sozial bat da, ikasleek ere hezkuntzako esperientzian ekarpen garrantzitsuak egin ditzaketela azpimarratzen. *Moodle*-en malgutasunari esker irakaskuntza metodo asko erabili daitezke. Material oso espezifikoak edo orokorra sortzeko erabili daiteke edota ebaluatzeko erabili daiteke.

Administrazio orokorra erabiltzailea den administrazioaile batek eramanen du, instalazioan zehar zehaztutakoa. Lekua pertsonalizatu daiteke “gai”ezberdinak erabiliz, eta hauek koloreak, letra motak, distribuzioa... erabakiko dute. Modulu berriak gehitu daitezke. Kodigoa *PHP*-n idatzita dago, GNU GPL lizentziarekin.

Pertsona bakoitzak soilik kontu bat behar du zerbitzari osoan zehar ibiltzeko. Kontu bakoitzak sarbide ezberdinak izan ditzake. Administrazioaile kontu batekin ikastaroen sorrera eta irakasleak kontrola daitezke, ikastaro bakoitzari erabiltzaileak esleituz. Irakasleek beraien ikastaroari sartzeko klabe bat ezarri diezaiekete, honela ikasleak ez direnei sarrera galaraziz. Gako hau pertsonalki adierazi edo korreoz bidali dezakete. Irakasleek ikasle bati sarrera galarazi diezaioke nahi baldin badu. Gainera, denbora jakin batean sartzen ez diren ikasleei sarrera galarazteko metodo automatiko bat ere badago, administrazioaileak ezarritakoa. Erabiltzaile bakoitzak bere ordutegi-zona zein den adierazi dezake, eta *Moodle*-n jarritako data guztiak horretara moldatuko dira. Gainera, hizkuntza ere aukeratu daiteke.

Irakasleak ikastaroaren aukera guztiak kontrola ditzake. Formatu desberdinak aukeratu daitezke ikastaroarentzat: asterokoak, gaien arabekoak eta abar. Orokorrean *Moodle*-k malgutasun handiko aktibitate ezberdinak eskaintzen ditu: foroak, egunerokoak, inkestak,

materialak, kontsultak, galdeketak eta egitekoa. Orri nagusian erabiltzailea azkenekoz sartu zenetik gertatu diren aldaketak adierazi daitezke. Nota guztiak, bai foroenak, egunerokoenak, inkestenak, galdeketenak... orri bakar batean ikusi daitezke eta kalkulu-orri batean jaitsi. Gainera, ikasle bakoitzaren aktibitatearen informeak eskuratu daitezke, grafikoekin eta modulu bakoitzean sartu dena adieraziz (azkenekoz noiz sartu zen, zenbat aldiz irakurri duen), baina baita ere ikasle bakoitzak zenbat parte hartu duen. Foroetara bidalitako mezuak postaz bidali daitezke, baita ere irakasleen komentarioak, ... dena *HTML* formatuan edo testu bat bezala.

Azken bertsio egonkorra *Moodle* 2.4.1 bertsioa da, baina La Salle Berrozpe ikastetxean *Moodle* 1.9 bertsioa erabiltzen denez, hau da proiektu honetan zehar erabiliko den bertsioa.

Gai honetan gehiago sakontzeko bibliografiako [1] eta [7] liburuak kontsultatu.

3.1.2 Moodle-en programatzeko baliabideak

Moodle-ek baliabide desberdinak jartzen dizkio eskuragarri garatzaileei. Programatze-lanak errazteko funtzio desberdinak dituzten liburutegiak eskaintzen ditu. Orain liburutegi nagusien deskribapen bat eta hauen funtzio nagusien aurkezpen bat egingo dut, implementazio garaian erabiliko direnak eta era sakonagoan erakutsiko direnak.

1. *dmlib.php*: Fitxategi honetan datu-basetik erregistroak lortzeko, ezabatzeko, eguneratzeko, kontatzeko eta berriak sartzeko behar diren funtzioak daude [3]. Datu-basea atzitzeko erabiltzen diren funtzio nagusiak hauek dira:
 - *get_record(\$table,\$field1,\$value1...)* *\$table* taulan *\$fieldX* eremuan *\$valueX* balioa duen erregistro bat itzultzen du objektu itxuran. Parametroak: *\$table*: erregistroak bilatzeko taula; *\$fieldX*: konparatzeko lehenengo eremua; *\$valueX*: konparatzeko lehenengo balioa.
 - *get_record_select(\$table,\$select)* *\$table* taulan parametro bezala pasatako WHERE klausula betetzen duen erregistro bat itzultzen du objektu itxuran. Parametroak: *\$table*: erregistroak bilatzeko taula; *\$select*: *SQL* sententziaren WHERE zatia.
 - *get_record_sql(\$sql)* *SQL* sententzia betetzen duen erregistro bat itzuliko du objektu itxuran. Parametroak: *\$sql*: 'SELECT .. FROM ..' erako *SQL* sententzia.
 - *get_records(\$table,\$field1,\$value1...)* *\$table* taulan *\$fieldX* eremuan *\$valueX* balioa duten erregistroak itzultzen ditu objektuak dituen *array* batean.
 - *get_records_select(\$table,\$select)* *\$table* taulan parametro bezala pasatako WHERE

klausula betetzen duten erregistroak itzultzen ditu objektuak dituen *array* batean.

- *get_records_sql(\$sql)* *SQL* sententzia betetzen duten erregistroak itzultzen ditu objektuak dituen *array* batean.
 - *delete_records()*, *get_records*-en parametro berdinekin, pasatako parametroak betetzen dituzten erregistroak ezabatzen ditu.
 - *insert_record(\$table,\$dataobject)* objektu bezala pasatako erregistroa *\$table* taulan sartzen du. *\$dataobject* objektuak erregistroaren eremu bakoitzaren izenarekin atributu bat edukiko du.
2. *weplib.php*: *HTML* elementuak sortzeko erabiltzen diren funtzioak [3]. Hauek dira liburutegi honen funtzio nagusiak:
- *choose_from_menu()* funtzioak hautatzeko menu bat inprimatzen du, *HTML*-ko *<select>*-en lana betetzen du.
 - *print_textfield()* funtzioak testu kutxa bat inprimatzen du, *HTML*-ko *<input type=text>*-en lana betetzen du.
 - *print_textarea()* funtzioak *HTML*-ko *<textarea>*-n lana betetzen du.
 - *print_header()* funtzio honek *Moodle* itxura ematen dion burukoa inprimatzen du.
 - *print_box_start()* eta *print_box_end()* funtzioak marra batez inguratzen ditu funtzio hauen artean dagoen edukia.

Fitxategi hauetako funtzioez aparte *Moodle*-k ere beste baliabide batzuk ere eskeintzen ditu:

3. Blokeak kudeatuko dituen erregistroak gordetzeko, datu-basean taulak sortuko dituen *install.xml* fitxategi bat behar da. Lan hau errazteko XMLDB editorea erabili daiteke. Editore honen bidez modu “grafikoan” blokeak beharko dituen taulen eta hauen eremuen definizioa egingo da. Taulak definitu ondoren editorearen bitartez *install.xml* fitxategia sortuko da, blokea instalatzerakoan *Moodle*-en datu-basean blokearen taulak sortuko dituena.
4. Hizkuntza fitxategiak kudeatzeko *get_string()* eta *print_string()* funtzioak erabiltzen dira. Hauek erabiliz hizkuntza aldatzerakoan pantailaratzen diren hitzak ere hizkuntza berdinerara aldatuko dira. Bloke guztiek duten *lang* karpetan hizkuntza bakoitzerako blokearen izena duten fitxategiak egongo dira. Fitxategi hauetan hitz kate bakoitzaren itzulpenak egongo dira eta *get_string()* funtzioaren bidez hitz kate hauek aldagaietan gordeko dira eta *print_string()* funtzioaren bidez pantailaratuko dira.
5. Blokea sortzeko ere funtzio desberdinak daude definituta. *block_list* klasea definituta

dago eta bloke berriak sortzeko klase hau erabiltzen da. Klase honetan funtzio desberdinak daude baina orain garrantzitsuenak aipatuko dira, funtzio hauetan gehiago sakontzeko *moodledocs*-ak kontsultatu [4]:

- *init()* funtzioak blokea sortzen du. Honen bitartez titulua eta bertsioa definituko dira, baina ez du beste ezer egingo.
- *get_content()* funtzioak blokearen edukia definitzen du. Hemen blokearen beste fitxategietarako estekak definituko dira eta baita blokearen oina ere.

3.1.3 phpMyAdmin

*MySQL*ren kudeaketa web-orrietan barrena erabiltzeko asmoarekin *PHP* lengoaiaren programatutako tresna bat da. Gaur egun, datu-baseak sortu, aldatu eta ezabatzeko gai da, baita taulak sortu, aldatu eta ezabatzeko, eremuak sortu, aldatu eta ezabatzeko, *SQL* sententziak exekutatzeko, eremuetan gakoak kudeatzeko, datuak formatu ezberdinetan esportatzeko... ere. 62 hizkuntzetara itzulita dago eta GNU lizentzia dauka.

Tresna hau 1998 urteaz geroztik indarrean dago. Tresna hau *PHP* eta *MySQL* euskarriak eta web-zerbitzariak dituzten makinetan erabiltzen denez erabilitako teknologia bere garapenean zehar aldatu egin da.

phpMyAdmin dokumentu-sorta zabal batekin dator eta erabiltzaileak ideiak partekatzerara eta eragiketa desberdinentzako tutorialak egitera gonbidatzen ditu wiki orriak eguneratuz. *phpMyAdmin*-eko taldeak edozein arazoren aurrean erabiltzailea laguntzen saiatuko da eskaintzen dituen kanalen bidez.

3.0.0 bertsioaz geroztik *phpMyAdmin GoPHP5* ekimenera elkartu zen, *PHP* eta *MySQL* bertsio zaharretarako kodearen bateragarritasuna eteten du, 3.0.0 bertsioak *PHP 5.2* eta *MySQL 5* bertsioak behar ditu.

Tresna honek sari desberdinak irabazi ditu. Besteak beste *PHP* aplikazio onena bezala aukeratu dute sari desberdinetan eta urtero SourceForge.net Community Choice Awards-etan sistemen administratzaileentzako tresnarik onenaren saria (Best Tool or Utility for SysAdmins) irabazi du. [7]

3.1.4 Apache

Kode irekiko *HTTP* web-zerbitzaria da, *Unix*, *Windows*, *Macintosh* eta beste plataforma batzuetan erabili daitekeena. *Apache*-k beste ezaugarri oso konfiguragarrien artean, autentifikazioko datu-baseak aurkezten ditu, baina bere konfigurazioan lagunduko duen interfaze grafiko baten eza kritikatu zaio.

Hasiera batean, existitzen zen web-zerbitzari bakarra *CERN*-eko *httpd* izan zen. Horren ostean, beste batzuk agertzen hasiko ziren, *NCSA*-eko *httpd* bezala, Illinoiseko unibertsitatekoa. Doakoa zen eta bertsio ezberdinak existitzen ziren plataforma ezberdinentzat. 1995. urtean, erabiliena zen, baina *NCSA*-k utzita zeukan bere garapena. Produktu ona zenez, programatzaile ezberdinak partxeak eta funtzionaltasun berriak sortzen hasi ziren, produktu onari jarraipena emanez. Horrela, partzez egindako web-zerbitzaria sortu zen, “a patchy server”. *Apache* sortu zen. Esan daiteke, *Apache*, software librearen funtzionamenduaren eta onuren adibiderik garbienetariko bat dela.

Aipatu behar da konfigurazio fitxategiak nagusiki hiru direla: *httpd.conf*, *access.conf* eta *srm.conf*. Hiru hauetatik lehenengoa baino ez da erabiltzen, garapenera dedikatzen direnek esaten dutenaren arabera, argiago delako.[7]

3.1.5 Besteak

Aipatutakoaz aparte, arruntagoak diren tresnak ere erabili behar izan dira proiektua garatzeko garaian. Testu editore bezala *LibreOffice Writer* erabili da, *Microsoft Word*-en antzekoa dena baina software librean eta nahiz eta honen lehiakidea izan erabat bateragarriak dira. Kodea garatzeko garaian *Eclipse* erabili da. Kode irekiko software-plataforma bat da. Hasiera batean garapen ingurune integratuak egiteko sortu bazen ere, programatzeko garapen ingurune integratua (IDE) izatea bere erabilera nagusia da. *Java*, *PHP*, *HTML*, *ANSI C...* lengoaiak erabiltzeko aproposa da. Web nabigatzaile bat ere beharrezkoa izan da, eta *Mozilla Firefox* aukeratu da. *Mozilla* fundazioak garatutako web nabigatzaile librea da, edonork lagundu dezakeena bere bilakaeran, eta euskaraz ere eskaintzen dena. Hainbat plataformatan erabil daiteke eta bere iturri kodea librea da.

4. ANALISIA

Lehenbiziko pausoa proiektu bat garatzeko garaian bezeroak eskatzen duen lana ondo finkatzea da, emaitzak entregatzerakoan gaizki ulerturik ez izateko eta bezeroaren helburuak erabat betetzeko. Horregatik lanaren nondik norakoa zehatzago definitzeko egin beharreko lehen pausua ikastetxeko proiektuko tutorearekin (Felix Elejoste) biltzea izan da.

Bilera honetan liburutegia Moodle-en integratzeko modurik onena bilatu ondoren, bloke moduan egitea erabaki da. Aurrekarietan aipatu bezala blokeak irakaskuntza-ikasketa prozesuan laguntzeko tresnak direlako eta hori delako liburutegiaren eginkizuna. Honela, liburutegirako sarbidea Moodle-eko orri nagusietatik eskuragarri egongo da. Administratzaileak erabakiko du blokea pantailako zein aldeetan agertuko den eta zein altueratan. Aplikazioak izan beharko dituen betebeharrak zehaztea izan da hurrengo pausoa.

Moodle 1.9 bertsioa erabiliko da proiektu honetan zehar, hau baita ikastetxean Moodle ezarri zenean zegoen bertsiorik egonkorrena. Bertsio berriagoetan izango duen erantzuna oraingoz ezezaguna denez etorkizunerako lan gisa utziko da.

Liburutegi fisikoan dauden elementuez aparte sarean dauden eta interesgarriak izan daitezkeen baliabideak ere sartu nahi dira datu-basean. Objektu bakoitzak izango dituen ezaugarriak erabakitzeko lehenbiziko pausoa formatu guztiak amankomunean dituzten ezaugarriak adostea da, hau da, metadatuak. Horrela objektu guztiak, bai liburuak, bai DVDak, bai CDak eta baita beste formatuetan dauden objektuak ere izena, hizkuntza, urtea, formatua, gaia, egilea eta etapa izango dutela amankomunean izango dutela erabaki da. Formatu ezberdinek izango dituzten ezaugarrietan erreparaturik, gauza garrantzitsurik ez dagoela jabetzen gara, CD batek zenbat abesti dituen edo DVD baten iraupena adibidez ez dira gauza garrantzitsuegiak. Horregatik, eta datu-basea sinplifikatzeko, objektu guztiei deskribapena eremua gehitzea erabaki da. Objektuari buruzko ezinbestekoa ez den informazioa sartu daiteke bertan, hau da, informazio osagarria. Objektu bakoitzak izango duen informazioa kontuan hartuta datu-basean sortu beharko diren taula eta hauen eremuen diseinu egokiena bilatuko da eta Moodle-eko programatzeko baliabideak puntuan aipatu bezela XMLDB editorearekin diseinatutako taulak sortuko dira.

Ikastetxeko liburutegian maileguak egiteko zerbitzu ofizialik ez dagoenez, aplikazioak ez du objektuen egoeraz berri emango, ez du esango erabilgarri dagoen, mailegatua dagoen edo objektua mailegatzea debekatua dagoen. Hala ere, etorkizunean egiteko aukera irekia uzten da.

Blokearen eginkizunak datuak sartzea, erakustea, aldatzea eta ezabatzea izango dira, hau da, objektuak erregistratzea, liburutegia kontsultatzea, erregistroak aldatzea eta ezabatzea. Hau guztia egiteko *PHP* fitxategiak sortu beharko dira eta ahal denean programatze lanak sinplifikatzeko *Moodle*-ek dituen funtzioak erabiliko dira (datu-basea atzitzeko adibidez).

Proiektuaren ezaugarriak aztertu eta finkatu ondoren, lanaren garapenean egon daitezkeen eta PHD-an aipatu diren arriskuak zehaztea izango da hurrengo pausoa. Proiektu honek ez du arrisku berezirik, hau da, lan honen garapenean gerta daitezkeen gauzak beste edozein lanetan gerta daitezkeenak dira. Ez du arrisku fisikorik, ezta zainketa edo arreta berezia behar duen material edo tresnaren erabilpenik behar.

Gerta daitezkeen arazorik larriena informazioa galtzea da. Hau da, egindako lana galtzea. Atzerapen handia sortuko luke eta gertatuz gero, lan guztia berregin beharko zen hasieratik abiatuta. Faktore ezberdinek eragin dezakete informazio galera, batzuk erabiltzaileak bultzatuta (nahigabezko ezabatzea...) eta beste batzuk kanpo faktoreek bultzatuta (makinen arazo teknikoak, birusen ondorioz, ekaitz baten eraginez...). Nahiz eta arazo hau larrienetakoa izan, aurre egiteko era asko daude eta ez dute zailtasunik. Arrisku hau ekiditeko era arruntena segurtasun-kopiak egitea da. Proiektu honetan segurtasun-kopiak ostiralero egin dira, material guztia eguneratua izateko. Segurtasun-kopiak hodeian gordeko dira, fitxategiak gordetzeko zerbitzu batean, hala ere USB memoria batean beste kopia bat izango da, baita nire etxeko ordenagailuan ere.

Aurreikusi gabeko arazo pertsonalak izan daitezke beste arrisku arrunt bat. Lan hau banakakoa da, beraz gaixotasunen bat edukiz gero lana pausatuta geratuko zen. Arazo hau saihestea ez da erreza, lehen aipatu bezala aurreikusi gabekoa delako, beraz hau gertatuz gero egin daitezkeen gauza bakarra planifikatuta zegoena baina ordu gehiago sartzea da galdutako denbora berreskuratzeko.

Azkenik, teknologia berrien ikasketekin zerikusia duten arazoak sortu daitezke. Proiektu honetan landuko ditudan gai asko niretzat ezezagunak dira. Gerta daiteke ikasketa prozesu honetan emandako denbora nik estimatutakoa baina luzeagoa izatea eta honi aurre egiteko denbora estimazioak egiterakoan saiaturiko naiz motz ez geratzen eta hau gertatuz gero plangintza aldatu beharko da eta aparteko ordu horiek denbora librean sartzen saiaturiko naiz.

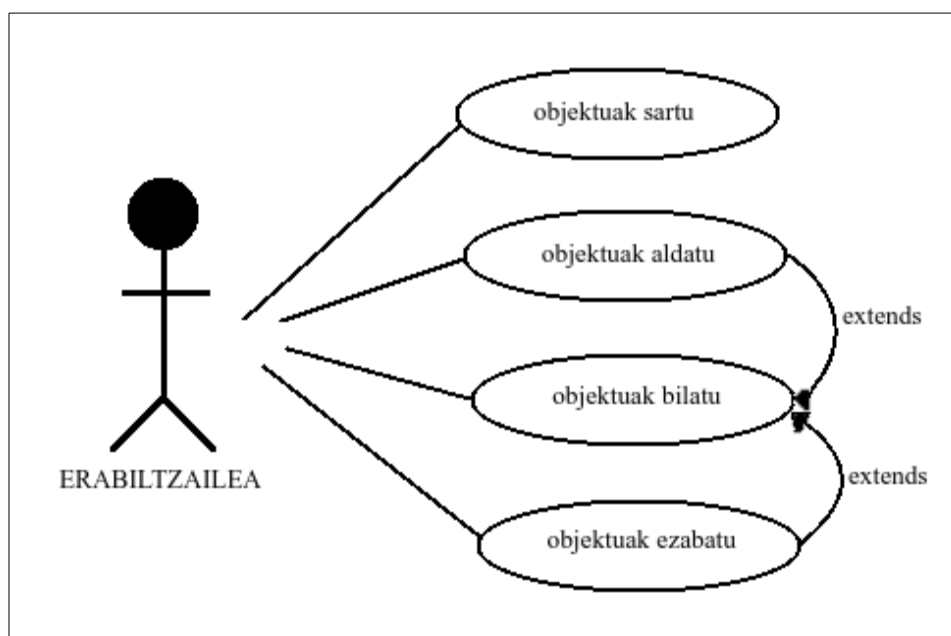
Analisisa bukatzeko, egon daitezkeen segurtasun ahuleziak aztertuko dira. *Moodle*-en administratzaileak erabiltzaile guztien baimenak definitzen ditu rol desberdinen bidez. Segurtasun-ahulezia nabarmenena erabiltzaile batek debekatuta duen eragiketa bat egitea da.

Adibidez ikasle batek datu-baseko erregistroak aldatzea edo ezabatzea lortzen badu, datu-basearen fidagarritasuna arriskuan egon liteke. Horretarako blokea garatzerakoan arreta jarri beharko dugu rol desberdinen erabilpenarekin eta administratzaileak baimenak ondo definitu beharko ditu datu-basearen sendotasuna bermatzeko.

4.1 Erabilpen-kasuen eredua

Erabilpen-kasuak definitu aurretik, aplikazioan parte hartzen duten “aktoreak” identifikatu behar dira. Aktore bat aplikazioarekin elkar-eragiten duen kanpoko entitate bat da. Aplikazio honen kasuan, erabiltzailea da parte hartzen duen aktore bakarra. Erabiltzaileak desberdinak izan daitezkeen arren (administratzailea, irakaslea, ikaslea...) bakoitzaren baimenak edozein momentutan alda daitezkeenez baimen guztiak dituen erabiltzaile baten adibidea aurkeztuko da orain, lehenetsitako moduan administratzailea da baimen guztiak dituen erabiltzailea.

Erabiltzailea aktore bezala hartuta, hurrengo pausoa honi dagozkion erabilpen-kasuak definitzea da. Erabilpen-kasuek aktorearen ikuspuntutik aplikazioak eskaintzen duen portaera definitzen dute, hau da, aplikazioak eskaintzen dituen funtzionalitateen deskribapenak dira.



Ilustrazioa 4: Erabilpen-kasuen Eredua

4.1.1 Objektuak sartu erabilpen-kasua

Deskribapena: erabilpen-kasu honen bitartez liburutegiko erregistrora, hau da, datu-basera, objektuak gehituko dira. Lehenetsitako moduan administratzaileek, ikastaro sortzaileek eta editatzeko baimena duten irakasleek izango dute hau egiteko baimena.

Gertaera-fluxu normala:

- 1- Objektu baten eremu guztiak hutsak agertuko dira baimendutako erabiltzaileak betetzeko.
- 2- Sartu botoia sakatzerakoan idatzitako informazioa datu-basean gordeko da.

Gertaera-fluxu alternatiboak:

1-Beharrezko eremuren bat hutsa uzten bada errore mezu bat eta atzera joateko botoi bat pantailaratuko da. Beste erroreren bat gertatuz gero ere errore mezu bat pantailaratuko da

4.1.2 Objektuak bilatu erabilpen-kasua

Deskribapena: erabilpen-kasu honen bitartez liburutegian dauden erregistroak bilatuko dira erabiltzaileak emandako irizpide batzuen arabera. Erabiltzaile guztiek izango dute hau egiteko baimena

Gertaera-fluxu normala:

- 1- Erabiltzaileak eremu desberdinen bidez bilaketa irizpideak ematen ditu.
- 2- Bilatu botoia sakatzerakoan bilaketa irizpideak betetzen dituzten objektuen informazioa pantailaratuko da.

Gertaera-fluxu alternatiboak:

- 1- Prozesuan zehar erroreren bat gertatuz gero errore mezu bat pantailaratuko da.

4.1.3 Objektuak aldatu erabilpen-kasua

Deskribapena: erabilpen-kasu honen bitartez liburutegian erregistratuta dauden objektuen informazioa aldatzeko aukera eskainiko da. Lehenetsitako moduan administratzaileek, ikastaro sortzaileek eta editatzeko baimena duten irakasleek izango dute hau egiteko baimena.

Gertaera-fluxu normala:

1- Hautatutako objektuaren eremu guztiak pantailaratuko dira hauek aldatzeko aukera emanaz.

2- Gorde botoia sakatzerakoan idatzitako informazioa datu-basean gordeko da.

Gertaera-fluxu alternatiboak:

1-Beharrezko eremuren bat hutsa uzten bada errore mezu bat eta atzera joateko botoi bat pantailaratuko da. Beste erroreren bat gertatuz gero ere errore mezu bat pantailaratuko da.

4.1.4 Objektuak ezabatu erabilpen-kasua

Deskribapena: erabilpen-kasu honen bitartez liburutegian erregistratuta dauden objektuak ezabatzeko aukera eskainiko da. Lehenetsitako moduan administratzaileek, ikastaro sortzaileek eta editatzeko baimena duten irakasleek izango dute hau egiteko baimena.

Gertaera-fluxu normala:

1- Hautatutako objektua ezabatu nahi duen edo ez galdetuko zaio erabiltzaileari.

2- Bai botoia sakatzerakoan objektua datu-basetik ezabatuko da eta ezezko botoia sakatuz aurreko orrira itzuliko da.

Gertaera-fluxu alternatiboak:

1-Prozesuan erroreren bat gertatuz gero errore mezu bat pantailaratuko da.

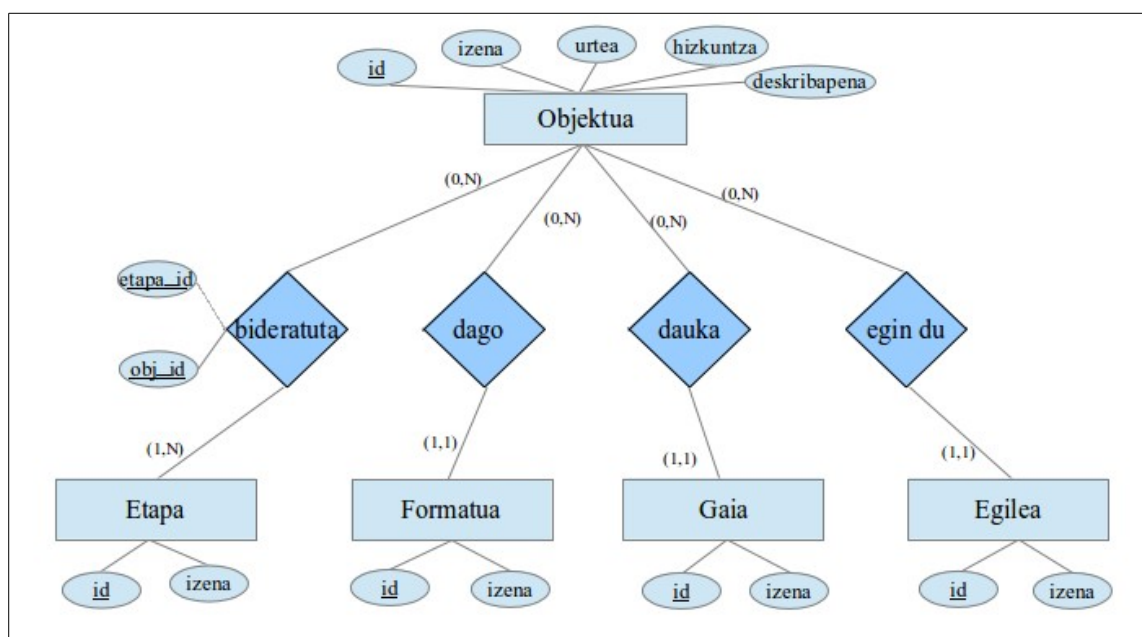
5. DISEINUA

5.1 Datu-basearen diseinua

Diseinatu beharreko lehenengo gauza datu-basea da, hau baita egitura guztia antolatuko duena. Analisisian zehaztu da zeintzuk izango diren erregistratu beharreko objektu bakoitzaren ezaugarriak, baina orain ezaugarri horiei gorputza eman beharko zaie, egitura aproposa aukeratuz. Datu-basearen diseinuan Moodle-ek dituen zenbait arau hartu beharko ditugu kontuan[6] eta 3. kapituluari aipatu diren baliabideak erabiliko dira inplementatzeko garaian (hau da, XMLDB editorea). Datu-baseko taula guztiek *id* izeneko eremu bat izan beharko dute *INT10* motakoa, autoinkrementala eta gako nagusia izango dena, eta blokeen taulen izenak egitura hau eramango dute:

"mdl_"+"block_" +taula sortzen duen blokearen izena+"_" +gehigarriak (non mdl, datu-basearen aurrizkia da, \$CFG->prefix aldagaian gordeko dena eta gehigarriak, taulari jarri nahi diogun izena).

Analisian objektu bakoitzaren izena, hizkuntza, urtea, formatua, gaia, egilea, etapa eta deskribapena gordeko direla finkatu da. Orain informazio hori datu-basera nola pasa pentsatu behar da. Horretarako E/E diagrama egingo dugu (5. irudia).



Ilustrazioa 5: E/E diagrama

Hasiera batean gaia eta formatua, objektuaren propietateak izan behar zirela erabaki zen. Baina ikastetxetik formatu berriak eransteko aukera eskatu zen, teknologia berriek aurrera egin ahala formatu berriak ateratzen direlako (eBook adibidez). Entitate berri bat eginez era errazean egiteko aukera eskaintzen da, administratzaileak datu-basean nahi dituen formatuak gehitu ditzakeelako, eta nahi duen unean. Gaiaren kasuan ere, gaiak ondo antolatuak egoteko, bakarrik administratzaileak finkatzen dituen gaietatik aukeratzea eskatzen da, baita administratzaileak edozein momentutan gai hauek aldatzea ere. E/E eredutik eredu erlazionalera taula hauek ateratzen dira:

mdl_block_liburutegi_objektua (id, izena, hizkuntza, urtea, deskribapena)

mdl_block_liburutegi_etapa (id, izena)

mdl_block_liburutegi_etapa_obj (etapa_id, obj_id)

mdl_block_liburutegi_egilea (id, izena, obj_id)

mdl_block_liburutegi_gaia (id, izena, obj_id)

mdl_block_liburutegi_formatua (id, izena, obj_id)

Taulen izenetan ikusten den bezala, gure blokearen izena liburutegi izango da eta *mdl_block_liburutegi_etapa_obj* taulan nahiz eta beharrezkoa ez izan *id* eremu bat jarri beharko da gako nagusi bezela Moodle-en arauak horrela behartzen dutelako.

Aplikazioaren interfazearen diseinua egitea izango da hurrengo pausoa. Moodle-eko bloke bat izango da aplikazioaren sarrera, beraz, hemengo diseinua Moodle-ek instalatuta daukan itxuraren arabera izango da. Sarrera honetan egongo diren aukeretak loturak egongo dira bakarrik, blokea sinplea izatea nahi delako, ulergarritasuna eta garbitasuna bermatzeko. Lotura hauen bidez liburutegia kontsultatzeko aukera, erregistroak gehitzeko aukera, erregistroak aldatzeko aukera, erregistroak ezabatzeko aukera eta liburutegia administratzeko aukera izango dugu. Azken aukera honen bidez gaiak, formatuak eta etapak definitzeko eta ezabatzeko aukera izango dugu. Blokeko aukeretako batean klikatuz gero orri berri bat irekiko da aukeratu dugun eskakizunekin. Orri berri hauetan Moodle-ek duen egitura eta itxura mantentzen saiatuko gera.

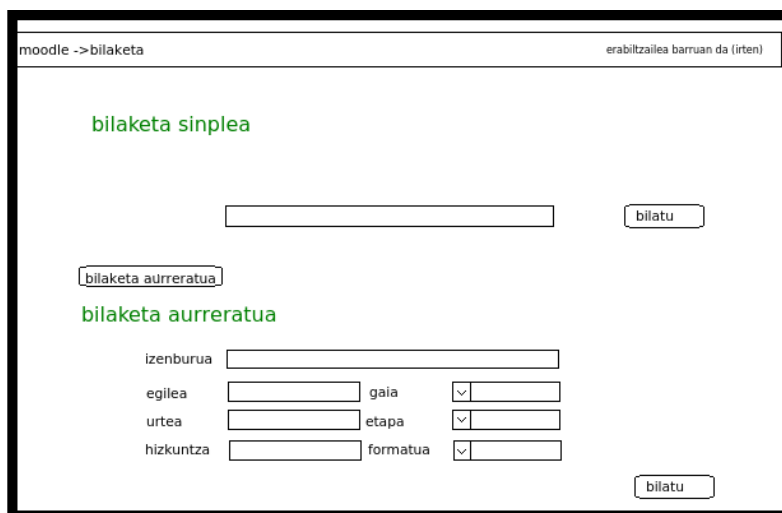
5.2 Liburutegiaren kontsulta

Hemen bilaketa egiteko aukera eskainiko da. Bi bilaketa desberdin egiteko aukera egongo da, bilaketa sinplea eta bilaketa aurreratua. Bilaketa sinplean testu-kutxa bakarra egongo da eta objektuaren eremu guztietan bilatuko da idatzitako testua. Bilaketa aurreratuan eremu desberdinen arabera bilaketa definitzeko aukera eskainiko da. Defektuz bilaketa sinplea aukeratuko da (6. irudia) eta bilaketa aurreratua egiteko botoi bat egongo da eremukako bilaketa erakutsiko duena. Bilaketa irizpideak jarri eta bilaketa egiteko botoia sakatu ondoren irizpide hori betetzen duten erregistroen informazioa pantailaratuko da.



Ilustrazioa 6: bilaketa sinplea pantailaren diseinua

Bilaketa aurreratua botoia sakatuta, eremuz eremuko bilaketa egiteko aukera agertuko da (7.irudia) bilaketa zehatzagoak egiteko aukera eskainiz. Objektuak dituzten eremu guztiak agertuko dira deskribapena izan ezik, azken eremu hau ez delako batere zehatza, informazio osagarria besterik ez duelako. URL eremua ez agertzea ere erabaki da honen informazioa ez delako lagungarria bilaketa egiteko eta informazio hau osagarria delako ere.



moodle -> bilaketa erabiltzailea barruan da (irten)

bilaketa sinplea

bilaketa aurreratua

izenburua

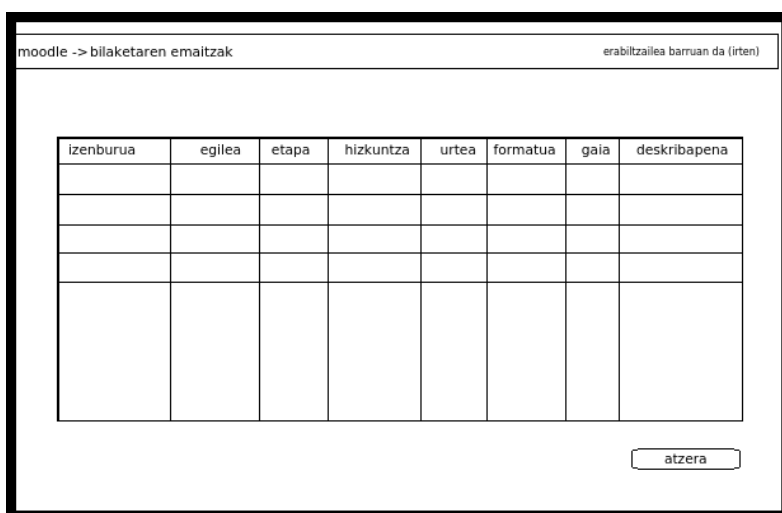
egilea gaia

urtea etapa

hizkuntza formatua

Ilustrazioa 7: bilaketa aurreratua pantailaren diseinua

Bai bilaketa aurreratua eta baita bilaketa sinplean ere, emaitzak formatu berdinean pantailaratuko dira. Hemen bilaketa irizpideak betetzen dituzten objektuak agertuko dira zutabeetan antolatuta 8. irudian agertzen den bezala. Zutabe bakoitza erregistroaren eremu bati egingo dio erreferentzia.



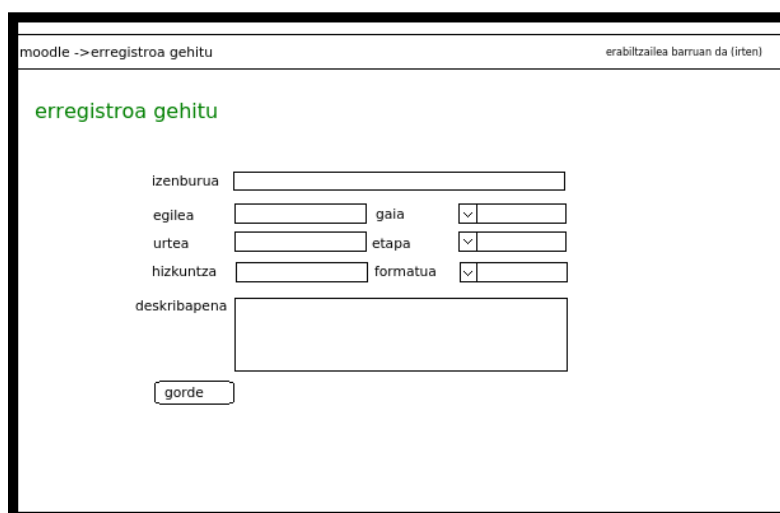
moodle -> bilaketaren emaitzak erabiltzailea barruan da (irten)

izenburua	egilea	etapa	hizkuntza	urtea	formatua	gaia	deskribapena

Ilustrazioa 8: bilaketaren emaitza pantailaren diseinua

5.3 Erregistroaren gehiketa

Hemen erregistro baten eremu bakoitzerako testu-koadro bat agertuko da eta gordetzeko botoi bat (9. irudia). Botoia sakatzerakoan dena ondo joan bada erregistroa datu-basean gordeko du eta mezu bat pantailaratuko du dena ondo joan dela adieraziz. Erregistroa gordetzen ez bada errore mezu bat pantailaratuko da.



moodle -> erregistroa gehitu erabiltzailea barruan da (irten)

erregistroa gehitu

izenburua

egilea gaia

urtea etapa

hizkuntza formatua

deskribapena

Ilustrazioa 9: erregistroa gehitu pantailaren diseinua

5.4 Erregistroaren aldaketa

Hemen datu-basean gordetako erregistroak aldatzeko aukera izango da. Bilaketa aurreratua bezalako pantaila izango da, lehenik eta bilaketaren emaitzak pantailaratzerakoan erregistro bakoitzak hau aldatzeko aukera ematen duen botoi bat izango du (10. irudia). Botoi hau sakatzerakoan erregistroaren eremu guztiak pantailaratuko dira erregistro berria sartzean agertzen den bezala, baina aurretik zeuden datuekin. Datuak aldatu ondoren, gorde botoia sakatuta aldaketak konfirmatzeko eskatuko da eta konfirmatuz gero erregistroa datu-basean

gorde eta mezu bat pantailaratuko da dena ondo joan dela esanez eta horrela ez bada errore mezu bat pantailan agertuko da.

moodle ->erregistroa aldatu erabiltzailea barruan da (irten)

bilatu aldatu nahi duzun erregistroa

izenburua

egilea gaia

urtea etapa

hizkuntza formatua

izenburua	egilea	etapa	hizkuntza	urtea	formatua	gaia	deskribapena	
								<input type="button" value="aldatu"/>
								<input type="button" value="aldatu"/>
								<input type="button" value="aldatu"/>
								<input type="button" value="aldatu"/>

Ilustrazioa 10: aldatzeko erregistroaren bilaketa pantailaren diseinua

11. irudian ikusten den bezala, erregistroa gehitzerakoan pantailaratzen diren testu kutxak pantailaratuko dira, baina kasu honetan objektuak dituen datuekin eta kurtsorea sartzeko aukerarekin, hau da, datuak aldatzeko aukerarekin.

5.5 Erregistroaren ezabaketa

Kasu honetan, erregistroa aldatzen denean bezala, bilaketa eremuka egingo da, eta lortutako emaitzak pantailaratzerakoan erregistro bakoitzaren ondoan ezabatzeko aukera ematen duen botoi bat egongo da (12. irudia). Botoi hau sakatzerakoan erregistroaren ezabaketaren berrespena eskatuko da eta konfirmatuz gero erregistroa deuseztatu eta mezu bat pantailaratuko da dena ondo joan dela esanez.

Bilaketa egin ondoren eta zein erregistro ezabatu nahi den aukeratu ondoren 13. irudian agertzen den pantailaren antzeko bat agertuko da ezabaketa konfirmatzeko eskatuz.

moodle ->erregistroa aldatzen erabiltzailea barruan da (irten)

erregistroa aldatzen

izenburua

egilea gaia

urtea etapa

hizkuntza formatua

deskribapena

Ilustrazioa 11: erregistroa aldatu pantailaren diseinua

moodle ->erregistroa ezabatu erabiltzailea barruan da (irten)

bilatu ezabatu nahi duzun erregistroa

izenburua

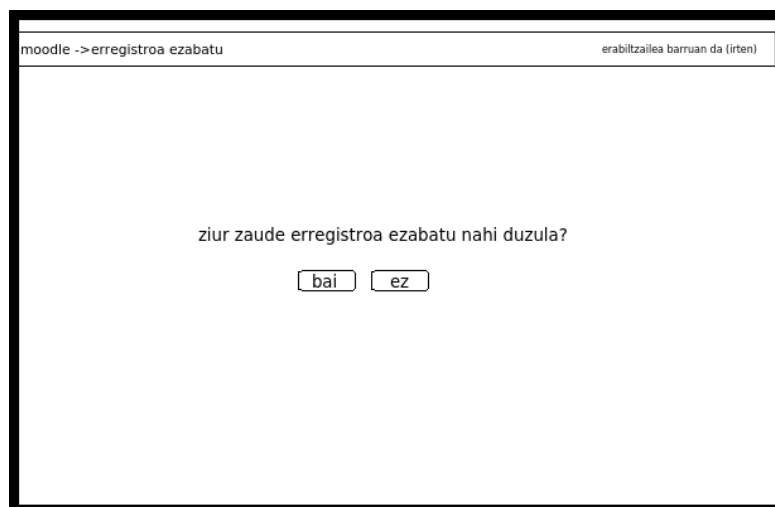
egilea gaia

urtea etapa

hizkuntza formatua

izenburua	egilea	etapa	hizkuntza	urtea	formatua	gaia	deskribapena	
								<input type="button" value="ezabatu"/>
								<input type="button" value="ezabatu"/>
								<input type="button" value="ezabatu"/>
								<input type="button" value="ezabatu"/>

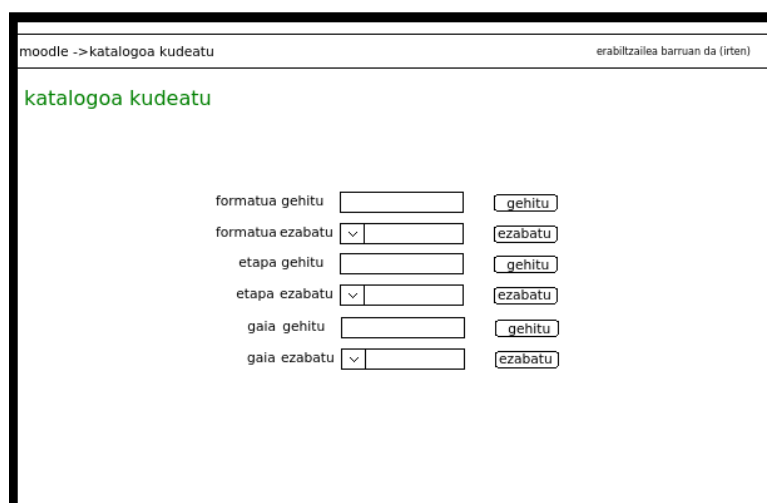
Ilustrazioa 12: ezabatzeko erregistroaren bilaketa pantailaren diseinua



Ilustrazioa 13: erregistroa ezabatu pantailaren diseinua

5.6 Katalogoaren kudeaketa

Ikastetxeko arduradunek egindako azken eskakizunen ondorioz (gaiak, formatuak eta etapak administratzaileak definitu behar ditu) beste aukera berri bat sortu behar izan da. Hemen administratzaileak liburutegiko objektuek izango dituzten etapa, formatu eta gaiak definitu beharko ditu.



Ilustrazioa 14: katalogoa kudeatu pantailaren diseinua

6. INPLEMENTAZIOA

Diseinua bukatu ondoren, lan zailenarekin hasiko gara, blokearen garapenarekin. Diseinuan erabaki dugu aplikazioa bloke batean egitea modulu batean egitea baztertuz. Inplementazioak hartuko du ordu kopuru handiena eta aurreko atalak ongi egin diren hemen ikusiko da. Blokea sortzeko *PHP* eta *HTML* lengoaietz gain, bloke eta moduluen garapenean laguntzeko *Moodle*-ek dituen liburutegi desberdinetan dauden funtzioak erabiliko dira. Egin beharreko lehen gauza blokea sortzea izango da, hau burutzeko 3.kapituluan aipatutako *Moodle*-ek dituen klase eta funtzioak erabiliko dira, zehatzago izateko *block_list* klasea eta bere metodoak erabiliko ditugu kasu honetan. Gero diseinatu den datu-basea sortu beharko da, hau egiteko *Moodle*-ek XMLDB editore bat dauka datu-basea sortuko duen *XML* fitxategia egiteko. Fitxategi honen bidez blokea instalatzerakoan *Moodle*-en datu-basean gure blokearen taulak sortuko dira. Honen ondoren aplikazioaren garapenarekin bukatuko da inplementazioa.

6.1 Blokearen definizioa

Egin beharreko lehenabiziko pausoa liburutegi blokea definitzea da [4]. Hasteko */blocks/liburutegi* karpeta sortuko dugu, slash (/) aipatzen dugunean *Moodle*-en *home* direktorioaz arituko gara, eta fitxategi eta direktorio guztiak direktorio honetatik hasiko dira. Blokeak era estandarrean egiten dira *Moodle*-en. Horretarako */blocks/liburutegi/block_liburutegi.php* fitxategia sortuko dugu. Lehenengo lerroan gure blokea definituko dugu (15. irudia). Gero klasera metodo bat ematen zaio, *init()* metodoa. Hau funtsezkoa da bloke guztietarako eta bere helburua *\$this->title* eta *\$this->version* aldagaiak definitzea da. *\$this->title* blokearen burukoan agertuko den titulua da eta *\$this->version* gure blokearen bertsioa.

```
class block_liburutegi extends block_list {
    function init() {
        $this->title = get_string('liburutegia', 'block_liburutegi');
        $this->version = 2013013000;
    }
}
```

Ilustrazioa 15: blokea sortzen duen *init()* funtzioa

Liburutegi blokeak zerbait pantailaratzeko beste metodo bat gehitu beharko da, `get_content()` metodoa. Honek hiru propietate izango ditu, `$this->content->items`, `$this->content->icons` eta `$this->content->footer`. Lehenbizikoa liburutegian izango diren eragiketa bakoitzeko esteka bat izango duen `array` bat izango da. Bigarrena ere `array` bat izango da, eta eragiketa bakoitzeko irudi bat izango du. Eta azkenak blokearen oinean agertuko den testua gordeko du.

```
function get_content() {
    if ($this->content !== NULL) {
        return $this->content;
    }

    $this->content = new stdClass;
    $this->content->items = array();
    $this->content->icons = array();
    $this->content->footer = '';

    $this->content->items[] = '<br><a href="'. $CFG->wwwroot.'
    /liburutegi/bilaketa.php">&nbsp;'.get_string('liburutegia_kontsultatu','block_liburutegi').'</a>';

    $this->content->icons[] = '<br>';

    $this->content->items[] = '<br><a href="'. $CFG->wwwroot.'
    /liburutegi/add_record.php">&nbsp;'.get_string('erregistroa_gehitu','block_liburutegi').'</a>';

    $this->content->icons[] = '<br>';

    $this->content->items[] = '<br><a href="'. $CFG->wwwroot.'
    /liburutegi/modify_record.php">&nbsp;'.get_string('erregistroa_aldatu','block_liburutegi').'</a>';

    $this->content->icons[] = '<br>';

    $this->content->items[] = '<br><a href="'. $CFG->wwwroot.'
    /liburutegi/remove_record.php">&nbsp;'.get_string('erregistroa_ezabatu','block_liburutegi').'</a>';

    $this->content->icons[] = '<br>';

    $this->content->items[] = '<br><a href="'. $CFG->wwwroot.'
    /liburutegi/administratu.php">&nbsp;'.get_string('administratu','block_liburutegi').'</a>';

    $this->content->icons[] = '<br>';

    return $this->content;
}
```

Ilustrazioa 16: blokearen edukia definitzen duen `get_content()` funtzioa

Orain blokearen titulua aldatzeko aukera ere jarriko dugu. Honetarako 17. irudian ikusten den funtzio sinplea gehituko dugu gure kodeari.


```
function instance_allow_config() {
return true;
}
```

Ilustrazioa 17: blokea editatzeko aukera ematen duen allow_config() funtzioa

Hau egin ondoren Moodle-ek blokea editatzeko aukera emango digu editatzeko irudian klikatuz, baina ez du ezer egingo /blocks/liburutegi/config_instance.html fitxategia (18. irudia) sortzea falta delako. Hemen, zer konfiguratu nahi den azaldu behar da. Beraz aipatutako fitxategia sortu eta bertan nahi dugun titulua sartzeko aldagaia garatuko da.

```
<table align="center" cellpadding="9" cellspacing="0">
<tr>
<td align="center"><p><?php print_string('configtitle', 'block_liburutegi'); ?></td>
<td><input type="text" name="title" size="30" value="<?php echo $this->config->title; ?>" /></td>
</tr>
<tr>
<td colspan="2" align="center">
<input type="submit" value="<?php print_string('savechanges') ?>" />
</td>
</tr>
</table>
<?php use_html_editor(); ?>
```

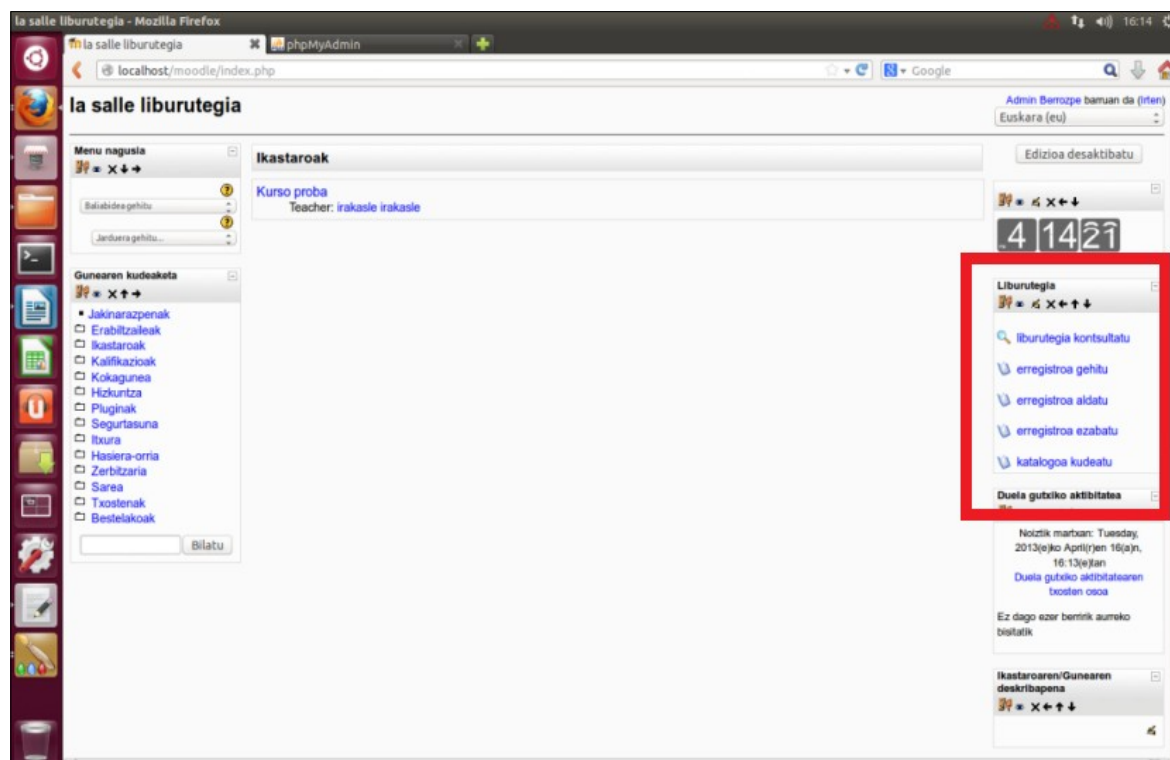
Ilustrazioa 18: zer editatu azaltzen duen config_instance.html fitxategiaren kodea

Orain, blokea editatzeko irudian sakatuta testu-kutxa bat agertuko da, eta bertan nahi dugun titulua jartzeko aukera emango da. Idatzitakoa *this->config->title* aldagaiari gordeko da, orain *\$this->title* aldagaiari balio hori esleitu behar zaio. Hau burutzeko, titulua blokean agerrarazteko, azken pauso hau falta da, baina *\$this->config* metodo guztietan erabil daiteke *init()*-en ezik, eta *\$this->title* metodoaren barruan dago. Hau konpontzeko metodo berri bat gehituko dugu. *Specialization()* metodoaren bitartez (19. irudia), barruan agindutakoa *init()* metodoa kargatu eta berehala kargatzea lortzen da. Gure kasuan tituluari guk idatzitako balioa esleituko zaio.

```
function specialization() {
$this->title = $this->config->title;
}
```

Ilustrazioa 19: titulua aldatzeko specialization metodoa

Hau egin ondoren liburutegiko blokea gehitzeko aukera dago, honetarako administratzaileak edizioa aktibatu eta blokeak gehitu sakatu eta liburutegiko blokea aukeratu beharko da.



Ilustrazioa 20: liburutegi blokea moodle-eko orri nagusian instalatuta

6.2 Datu-basearen definizioa

Gure Moodle sisteman blokea sortu eta ezarri dugu, baina oraindik ez du ezer eginen. Orain gure liburutegia kudeatuko duen sistema sortu beharko da. Aurreneko pausoa diseinatu den datu-basea sortzea da. Hau Moodle-ek duen XMLDB editorearen bitartez egingo dugu. Honetarako administratzailearen menuan bestelakoak aukeratu eta XMLDB editorea topatuko dugu. Hemen `blocks/liburutegi/db` bilatu eta sortzea aukeratu dugu. Horrela `/blocks/liburutegi/db/install.xml` fitxategia sortuko da, blokea instalatzerakoan datu-basean taula berriak sortuko dituena. Orain fitxategi hau editatuko dugu XMLDB editorearen bitartez.

Taula berrian klikatu eta defektuz `id` izeneko eremu bat agertzen da, gako nagusia izango

dena. Hemen taularen izena idatziko da eta eremu berriak sartu beharko dira. Adibidez, *mdl_block_liburutegi_objektua* taula definitzeko izena eremu berria gehituko dugu char motakoa, 60 byteko luzerakoa eta ezingo da *null* izan, liburutegiko objektu guztiek izenen bat izan beharko dutelako. Hizkuntza eremua *char* mota eta 10 bytekoa izango da, baina eremu honen balioa mugatuta egongo da, estandarizaturia, hizkuntzaren menpe ez egoteko. Euskara, castellano, ingelesa eta frantsesaren artean hautatu beharko da. Urtea 4 byteko *int* motako zenbaki bat izango da eta ez du *null* izateko aukerarik izango. Azkenik deskribapena eremua sortuko da, *char* mota eta 255 bitekoa. 21. irudian ikusten den bezala geratuko da egin berri dugun taula.

Egin behar den gauza desberdin bakarra, *mdl_block_liburutegi_formatu_obj*, *mdl_block_liburutegi_etapa_obj* eta *mdl_block_liburutegi_gaia_obj* gakoak definitzerakoan foreign key gisa definitu beharko dira bi gako, bat *mdl_block_liburutegi_objektua* taularen *id* eremuarekiko, eta bestea formatu, etapa, gaia edo egilearen *id* eremuarekiko.

Editatu taula

Name:

Comment:

[\[Ikusi jatorrizkoa\]](#)
[\[Ikusi editatutakoa\]](#)
[\[Eremu berria\]](#)
[\[Giltza berria\]](#)
[\[Aurkibide berria\]](#)
[\[Aizera\]](#)

[\[Ikusi SQL kodea\]](#)
[\[Ikusi PHP kodea\]](#)

Eremuak

id	[Editatu] [Gora] [Behera] [Ezabatu]	int (10) unsigned not null auto-numbered
izena	[Editatu] [Gora] [Behera] [Ezabatu]	char (60) not null
hizkuntza	[Editatu] [Gora] [Behera] [Ezabatu]	char (10) enum('Euskera', 'Castellano', 'Ingelesa', 'Frantsesa') not null default 'Euskera'
urtea	[Editatu] [Gora] [Behera] [Ezabatu]	int (4) unsigned not null
deskribapena	[Editatu] [Gora] [Behera] [Ezabatu]	char (255)

Giltzak

[primary](#) [Editatu] [Gora] [Behera] [Ezabatu] primary (id)

Ilustrazioa 21: XMLDB editorearekin sortutako *mdl_block_liburutegi_objektua* taula

Orain, taula bakoitzerako, egin berri den prozesua errepikatuko da XMLDB editorearekin. 22. irudian ikusten da definituko den hurrengo taula.

Editatu taula

Name:

Comment:

[\[Ikusi jatorrizkoa\]](#)
[\[Ikusi editatutakoa\]](#)
[\[Eremu berria\]](#)
[\[Giltza berria\]](#)
[\[Aurkibide berria\]](#)
[\[Atzera\]](#)

[\[Ikusi SQL kodea\]](#)
[\[Ikusi PHP kodea\]](#)

Eremuak

id	[Editatu]	[Gora]	[Behera]	[Ezabatu]	int (10) unsigned not null auto-numbered
izena	[Editatu]	[Gora]	[Behera]	[Ezabatu]	char (30) not null

Giltzak

primary	[Editatu]	[Gora]	[Behera]	[Ezabatu]	primary (id)
----------------	---------------------------	------------------------	--------------------------	---------------------------	--------------

Ilustrazioa 22: mdl_block_liburutegi_egilea taularen definizioa

23. irudian *mdl_block_liburutegi_egile_obj* taula ikusten da. Taula honetan objektuen eta egileen arteko erlazioa definituko da. Honetarako objektuaren *id* eremua eta egilearena ere foreign key gisa definitu beharko dira.

Editatu taula

Name:

Comment:

[\[Ikusi jatorrizkoa\]](#)
[\[Ikusi editatutakoa\]](#)
[\[Eremu berria\]](#)
[\[Giltza berria\]](#)
[\[Aurkibide berria\]](#)
[\[Atzera\]](#)

[\[Ikusi SQL kodea\]](#)
[\[Ikusi PHP kodea\]](#)

Eremuak

id	[Editatu]	[Gora]	[Behera]	[Ezabatu]	int (10) unsigned not null auto-numbered
egile_id	[Editatu]	[Gora]	[Behera]	[Ezabatu]	int (10) unsigned not null
obj_id	[Editatu]	[Gora]	[Behera]	[Ezabatu]	int (10) unsigned not null

Giltzak

primary	[Editatu]	[Gora]	[Behera]	[Ezabatu]	primary (id)
egile_id	[Editatu]	[Gora]	[Behera]	[Ezabatu]	foreign (egile_id) references block_liburutegi_egilea (id)
obj_id	[Editatu]	[Gora]	[Behera]	[Ezabatu]	foreign (obj_id) references block_liburutegi_objektua (id)

Ilustrazioa 23: mdl_block_liburutegi_egile_obj taularen definizioa

24. irudian *mdl_block_liburutegi_etapa* taularen definizioa ikusten da.

Editatu taula

Name:

Comment:

[\[Ikusi jatorrizkoa\]](#) [\[Ikusi editatutakoa\]](#) [\[Eremu berria\]](#) [\[Giltza berria\]](#) [\[Aurkibide berria\]](#) [\[Atzera\]](#)
[\[Ikusi SQL kodea\]](#) [\[Ikusi PHP kodea\]](#)

Eremuak

[\[Editatu\]](#) [\[Gora\]](#) [\[Behera\]](#) [\[Ezabatu\]](#) int (10) unsigned not null auto-numbered
 [\[Editatu\]](#) [\[Gora\]](#) [\[Behera\]](#) [\[Ezabatu\]](#) char (20) not null

Giltzak

[\[Editatu\]](#) [\[Gora\]](#) [\[Behera\]](#) [\[Ezabatu\]](#) primary (id)

Ilustrazioa 24: mdl_block_liburutegi_etapa taularen definizioa

Editatu taula

Name:

Comment:

[\[Ikusi jatorrizkoa\]](#) [\[Ikusi editatutakoa\]](#) [\[Eremu berria\]](#) [\[Giltza berria\]](#) [\[Aurkibide berria\]](#) [\[Atzera\]](#)
[\[Ikusi SQL kodea\]](#) [\[Ikusi PHP kodea\]](#)

Eremuak

[\[Editatu\]](#) [\[Gora\]](#) [\[Behera\]](#) [\[Ezabatu\]](#) int (10) unsigned not null auto-numbered
 [\[Editatu\]](#) [\[Gora\]](#) [\[Behera\]](#) [\[Ezabatu\]](#) int (10) unsigned not null
 [\[Editatu\]](#) [\[Gora\]](#) [\[Behera\]](#) [\[Ezabatu\]](#) int (10) unsigned not null

Giltzak

[\[Editatu\]](#) [\[Gora\]](#) [\[Behera\]](#) [\[Ezabatu\]](#) primary (id)
 [\[Editatu\]](#) [\[Gora\]](#) [\[Behera\]](#) [\[Ezabatu\]](#) foreign (obj_id) references block_liburutegi_objektua (id)
 [\[Editatu\]](#) [\[Gora\]](#) [\[Behera\]](#) [\[Ezabatu\]](#) foreign (etapa_id) references block_liburutegi_etapa (id)

Ilustrazioa 25: mdl_block_liburutegi_etapa_obj taularen definizioa

25. irudian objektuen etapak eta objektuak erlazionatzen dira bi taula hauen gako nagusiak taula honetan kanpoko gako gisa jarritz.

Editatu taula

Name: **block_liburutegi_formatua**

Comment:

[\[\[kusi jatorrizkoa\]\]](#)
[\[\[kusi editatutakoa\]\]](#)
[\[Eremu berria\]](#)
[\[Giltza berria\]](#)
[\[Aurkibide berria\]](#)
[\[Atzera\]](#)

[\[\[kusi SQL kodea\]\]](#)
[\[\[kusi PHP kodea\]\]](#)

Eremuak

id	[Editatu]	[Gora]	[Behera]	[Ezabatu]	int (10) unsigned not null auto-numbered
izena	[Editatu]	[Gora]	[Behera]	[Ezabatu]	char (20) not null

Giltzak

primary	[Editatu]	[Gora]	[Behera]	[Ezabatu]	primary (id)
----------------	---------------------------	------------------------	--------------------------	---------------------------	--------------

Ilustrazioa 26: mdl_block_liburutegi_formatua taularen definizioa

Nahiz eta hasiera batean formatua objektuaren taularen eremu bat bezala definitzea espero zen, formatuak gehitu edo ezabatzeko aukera eskatu zenez, taula berri bat sortzea erabaki zen (26. irudia). Horrela egile eta etaparen era berdinean erlazionatuko dira formatua eta objektua, hau da hauen gakoak beste taula batean erlazionatuz (27. irudia).

XMLDB editorearekin bukatzeko azken bi taulak sortu beharko dira, *mdl_block_liburutegi_gaia* taula eta *mdl_block_liburutegi_gaia_obj* taula. Hauek ere oso sinpleak izango dira, objektuak izan ditzakeen taulan gaien izenak gordeko dira eta hauek objektuekin erlazionatu ahal izateko id zenbaki bat, gako nagusia izango dena.

Editatu taula

Name:

Comment:

[\[Ikusi jatorrizkoa\]](#) [\[Ikusi editatutakoa\]](#) [\[Eremu berria\]](#) [\[Giltza berria\]](#) [\[Aurkibide berria\]](#) [\[Atzera\]](#)
[\[Ikusi SQL kodea\]](#) [\[Ikusi PHP kodea\]](#)

Eremuak

id	[Editatu]	[Gora]	[Behera]	[Ezabatu]	int (10) unsigned not null auto-numbered
formatu_id	[Editatu]	[Gora]	[Behera]	[Ezabatu]	int (10) unsigned not null
obj_id	[Editatu]	[Gora]	[Behera]	[Ezabatu]	int (10) unsigned not null

Giltzak

primary	[Editatu]	[Gora]	[Behera]	[Ezabatu]	primary (id)
formatu_id	[Editatu]	[Gora]	[Behera]	[Ezabatu]	foreign (formatu_id) references block_liburutegi_formatua (id)
obj_id	[Editatu]	[Gora]	[Behera]	[Ezabatu]	foreign (obj_id) references block_liburutegi_objektua (id)

Ilustrazioa 27: mdl_block_liburutegi_formatu_obj taularen definizioa

Editatu taula

Name:

Comment:

[\[Ikusi jatorrizkoa\]](#) [\[Ikusi editatutakoa\]](#) [\[Eremu berria\]](#) [\[Giltza berria\]](#) [\[Aurkibide berria\]](#) [\[Atzera\]](#)
[\[Ikusi SQL kodea\]](#) [\[Ikusi PHP kodea\]](#)

Eremuak

id	[Editatu]	[Gora]	[Behera]	[Ezabatu]	int (10) unsigned not null auto-numbered
izena	[Editatu]	[Gora]	[Behera]	[Ezabatu]	char (20) not null

Giltzak

primary	[Editatu]	[Gora]	[Behera]	[Ezabatu]	primary (id)
-------------------------	---------------------------	------------------------	--------------------------	---------------------------	--------------

Ilustrazioa 28: block_liburutegi_gaia taularen definizioa

29. irudian definitutako azken taula ikusten da *mdl_block_liburutegi_gaia_obj* taula, eta aurreko kasuetan bezala hau ere zubi lanak egiten ditu liburutegi objektuaren eta honi dagokion gaiarekin.

Editatu taula

Name:

Comment:

[\[\[kusi jatorrizkoa\]\]](#)
[\[\[kusi editatutakoa\]\]](#)
[\[Ereму berria\]](#)
[\[Giltza berria\]](#)
[\[Aurkibide berria\]](#)
[\[Atzera\]](#)
[\[\[kusi SQL kodea\]\]](#)
[\[\[kusi PHP kodea\]\]](#)

Eremuak

id	[Editatu]	[Gora]	[Behera]	[Ezabatu]	int (10) unsigned not null auto-numbered
gai_id	[Editatu]	[Gora]	[Behera]	[Ezabatu]	int (10) unsigned not null
obj_id	[Editatu]	[Gora]	[Behera]	[Ezabatu]	int (10) unsigned not null

Giltzak

primary	[Editatu]	[Gora]	[Behera]	[Ezabatu]	primary (id)
gai_id	[Editatu]	[Gora]	[Behera]	[Ezabatu]	foreign (gai_id) references block_liburutegi_gaia (id)
obj_id	[Editatu]	[Gora]	[Behera]	[Ezabatu]	foreign (obj_id) references block_liburutegi_objektua (id)

Ilustrazioa 29: mdl_block_liburutegi_gaia_obj taularen definizioa

XMLDB editorean *blocks/liburutegi/db* fitxategia 30. irudian bezala geratuko da blokeak behar dituen taula guztiak definitu ondoren. Hau gorde ondoren datu-basearen definizioarekin bukatuko da eta aplikazioaren garapenarekin jarraitu beharko da.

Editatu XML fitxategia

Path:

Version:

Comment:

[\[\[kusi jatorrizkoa\]\]](#)
[\[\[kusi editatutakoa\]\]](#)
[\[Taula berria\]](#)
[\[Taula berria MySQL-etik\]](#)
[\[Sententzia berria\]](#)
[\[Ori nagusira itzuli\]](#)
[\[\[kusi SQL kodea\]\]](#)
[\[\[kusi PHP kodea\]\]](#)

Taulak

block_liburutegi_objektua	[Editatu]	[Gora]	[Behera]	[Ezabatu]
block_liburutegi_egilea	[Editatu]	[Gora]	[Behera]	[Ezabatu]
block_liburutegi_egile_obj	[Editatu]	[Gora]	[Behera]	[Ezabatu]
block_liburutegi_etapa	[Editatu]	[Gora]	[Behera]	[Ezabatu]
block_liburutegi_etapa_obj	[Editatu]	[Gora]	[Behera]	[Ezabatu]
block_liburutegi_formatua	[Editatu]	[Gora]	[Behera]	[Ezabatu]
block_liburutegi_formatu_obj	[Editatu]	[Gora]	[Behera]	[Ezabatu]
block_liburutegi_gaia	[Editatu]	[Gora]	[Behera]	[Ezabatu]
block_liburutegi_gaia_obj	[Editatu]	[Gora]	[Behera]	[Ezabatu]

Ilustrazioa 30: XMLDB editorearekin egindako blokearen taulak

6.3 Aplikazioen garapena

Orain, bloke nagusiko aukera bakoitzerako aplikazioa garatu beharko da, eta horretarako *PHP* eta *HTML* lengoiaiez gain *Moodle*-ek dituen liburutegi desberdinen funtzioak erabiliko dira. Honez gain *Moodle*-ek arau orokor batzuk betetzea eskatzen du [6] fitxategi berriak garatzeko orduan. Arau nagusiak hauek dira:

- kode fitxategi guztiak *.php* luzapena izan behar dute.
- *PHP* etiketa guztiak 'osoak' izan behar dira, `<?php ?>` bezalakoak bai eta ez `<? ?>` motakoak.
- fitxategi guztiak `require_once("../config.php")` kodea izan behar dute.
- beste edozein include/require guztiek `$CFG->dirroot` edo `$CFG->libdir`-ekin hasitako bide absolutua erabili beharko dute, inoiz ez erlatiboak, hauek era arraroan funtzionatu dezaketelako.
- fitxategi guztiek frogatu beharko dute erabiltzailea kautotuta dagoela `require_login()`, `isadmin()`, `isteacher()`, `iscreator()` edo `isstudent()` funtzioak erabiliz.
- datu-basera egindako atzipen guztiak `/lib/datalib.php`-en definitutako funtzioen bidez egiten saiatu behar da. Funtzio hauek erabiliz ia dena egitea posible da.
- hitz kate guztiak beste hizkuntzetara itzulgarriak izan behar dira. `/blocks/liburutegi/lang` karpetan hitz kate guztiak hizkuntza desberdinetan dituzten fitxategiak egongo dira eta `get_string()` edo `print_string()` funtzioen bidez berreskuratuko dira.
- `$_POST` eta `$_GET` funtzioak ez erabili, hauen ordez `required_param()` edo `optional_param()` funtzioak erabili.
- ahal denean `required_param()` eta `optional_param()` funtzioen dei guztiak elkartu hauek aurkitzea errazagoa izateko.
- *SQL* kodea idazten bada zuzena dela frogatu. Bereziki, aldagaietan komatxoak erabilera frogatu.

6.3.1 Liburutegiaren kontsulta

Diseinuan aipatu den bezala, hemen guk emandako irizpide bat betetzen duten erregistroak bistaratu beharko dira. Irizpide hori emateko bi aukera eskainiko dira, bilaketa

sinplea (guk idatzitako bilagaia erregistroen eremu guztietan bilatuko dira) eta bilaketa aurreratua (guk erabakiko dugu zein eremutan egingo dugun bilaketa).

Egingo den lehenbiziko gauza Moodle-eko karpeta nagusian liburutegi karpeta bat sortzea izango da, egingo diren fitxategi guztiak gordetzeko. Lehenbiziko fitxategia */liburutegi/bilaketa.php* izango da.

Moodle-en arauak esaten duten bezala aurreneko pausoa **require_once("../config.php")** idaztea da, hemen *\$CFG* aldagai globala definitzen da. Honen ondoren *require_login()* funtzioa deituko dugu erabiltzailea kautotuta dagoela bermatzeko. Orri hau Moodle-en formatua izateko *print_header()* funtzioa erabiliko da. Funtzio honek Moodle-eko goiburu bat sortuko eta Moodle-ek duen temaren estiloa emango dio. Goiburuaren ezker aldean nabigazio bide bat agertuko da Moodle-en orri nagusirako lotura batekin eta eskuin aldean erabiltzaile izena agertuko da eta saioa ixteko botoia. Hau fitxategi berri guztietan egingo dugu.

Orain bilaketa sinplea egiteko testu-kutxa eta botoia definituko dira. Honetarako 3. kapituluan aipatutako */lib/weblib.php* liburutegia erabiliko da. Hemen testu-kutxa, select, checkbox... era errazagoan egiteko funtzioak aurkituko dira. *print_textfield()* funtzioaren bitartez testu-kutxa bat sortuko dugu eta HTML-z botoi bat. Testu-kutxan erabiltzaileak nahi duen bilagaia sartuko du eta botoian klikatuz bilaketaren emaitza pantailaratuko du.

Bilaketa aurreratua bistaratzeko botoi bat jarriko da bilaketa aurreratuaren eremuak erakutsi eta ezkutatzeko. Hau lortzeko bilaketa eremuak *div* motako edukiontzian sartuko dira eta Java script baten bidez *style.display* propietatea aldatuko da eremuak bistaratzeko edo ezkutatzeko.

```
<script type="text/javascript">
function ocultar(id){
    var elDiv = document.getElementById(id); //se define la variable "elDiv" igual a nuestro div
    elDiv.style.display='none'; //damos un atributo display:none que oculta el div
}
function muestra(id){
    var elDiv = document.getElementById(id); //se define la variable "elDiv" igual a nuestro div
    elDiv.style.display='block';//damos un atributo display:block que el div
}
window.onload = function(){/*hace que se cargue la función */
/* "Mandamos como parametro el nombre de la Div para ocultar" */
    ocultar('bilaurre');
}
</script>
```

Ilustrazioa 31: bilaketa aurreratua erakutsi eta ezkutatzeko erabiliko diren funtzioak

Esan bezala bilaketa eremuak *div* batean gordeko dira izenburua, egilea eta urterako *print_textfield()* funtzioa erabiliz testu-kutxak jarriko dira eta etapa, formatua, hizkuntza eta gairako *choose_from_menu()* funtzioa erabiliko da hautatzeko eremuak sortzeko. Hautatzeko menuen aukerak administratzaileak definitzen ditu katalogoa administratzeko menuaren bitartez eta aukerak zeintzuk diren jakiteko datu-basea atzitu beharko da. Hau egiteko */lib/datalib.php*-en dauden funtzioak erabiliko dira, *get_records_sql()* funtzioa zehatzagoa izateko. Funtzio honen bidez *SQL* bitartez idatzitako *select* agindua betetzen duten erregistroak itzuliko dira, atzitutako taularen eremuak dituen objektuz osatutako array batean. Bukatzeko botoi bat gehituko da hau klikatzerakoan bilaketaren emaitzak pantailaratzeko.

Bai bilaketa sinplearen botoia, bai bilaketa aurreratuaren botoia klikatzerakoan */liburutegi/result.php* fitxategi berria kargatuko da *HTML*-z egindako formularioen bitartez. Fitxategi honek bilaketaren emaitzak pantailaratu beharko ditu. Hemen egin beharreko lehen gauza fitxategi guztietan bezala `require_once("../config.php")`, *require_login()* eta *print_header()* funtzioak deitzea izango da. Ondoren bilaketa aurreratua edo sinplea egin den jakin behar da, datu-basea atzitzeko *SQL* galderak desberdinak direlako kasu bakoitzean. Hau jakiteko *optional_param()* funtzioaren bidez bilaketa sinplearen botoia edo bilaketa aurreratuarena klikatu den aztertuko da.

Moodle-en arau orokorrak ahal denean *optional_param()* eta *required_param()* funtzioen deiak elkartzea eskatzen duenez, bilaketa aurreraturako erabiltzaileak idatzitako parametroak jarraian jasoko dira *optional_param()* funtzioaren bidez. Hau egin ondoren *SQL* sententziak definituko dira bai bilaketa sinplerako baita bilaketa aurreraturako. Bilaketa sinplea egiteko taula guztiak elkartuko dira *inner join* batzuen bidez, gero bilagaia eremu guztiekin konparatzeko. Bilaketa aurreratuan *SQL* dinamikoa erabiliko da, eremu guztiak ez direlako beti aztertu behar, bakarrik zerbait idatzita edo aukeraren bat hautatua duten eremuak aztertu behar direlako. Kasu honetan ere taula guztiak *inner join* batzuen bidez elkartuko dira, baina *SQL* sententziako *WHERE* atala dinamikoki sortuko da erabili gabeko eremuak baztertuz. Zein bilaketa mota hautatu den *if* baten bidez jakin ondoren emaitzak agertuko diren taularen lehendabiziko errenkada idatziko da, eta gero *SQL* emaitzak dituen *array*-ak elementuak dituen bitartean elementu bakoitzeko eremu guztiak pantailaratu dira atzera itzultzeko botoi batekin batera.

6.3.2 Erregistroaren gehiketa

/liburutegi/add_record.php fitxategia sortuko dugu erregistroak gehitzeko. Hemen erregistro berriak datu-basean sartzeko eremu bakoitzeko testu-kutxa bat egongo da eta botoi bat eragiketa konfirmatzeko. Egin beharreko lehendabiziko pausua aurreko fitxategietan bezala **require_once**("../config.php"), *require_login()* eta *print_header()* funtzioak deitzea izango da. Kasu honetan gerta daiteke datu-basean objekturik ez egotea gordeta, eta hau gertatuz gero ez da hizkuntza hautatzeko aukerarik egingo, hau ez gertatzeko hizkuntzak datu-basetik hartu ordez kodean idatziz eskainiko dira. Formatuen kasuan ere defektuz DVD, CD eta liburua formatuekin hasieratuko da, nahiz eta gero administratzaileak aukerak gehitu eta ezaba ditzakeen. Hau egin ondoren datu-basea atzitu dugu dauden gai eta etapak array batean gordetzeko gero hautatzeko eremutan eskaintzeko. Gero *HTML*-z formulario bat sortuko da liburutegiko objektuen eremuekin */lib/weblib.php* liburutegiko funtzioen laguntzaz (*print_string()*, *print_textfield()*, *choose_from_menu()*...), eta bukatzeko erregistroa datu-basean gordetzeko botoi bat jarriko da.

Botoi hau sakatzerakoan *required_param()* eta *optional_param()* funtzioen bidez erabiltzaileak idatzitako parametroak jaso eta gero parametroak hutsak utzi diren edo ez zehaztuko da. Azkeneko pausu hau beharrezkoa da liburutegiko erregistro guztiek eremu minimo batzuk beteta derrigorrezkoa delako, kasu honetan deskribapena eta urtea izango dira hutsak utzi daitezkeen eremuak. Beharrezko eremuren bat hutsa baldin badago mezu baten bidez jakinaraziko zaio erabiltzaileari, bestela *mdl_block_liburutegi_objektua* taulako eremuak dituen elementu bat sortuko da eta *insert_record()* funtzioaren bitartez datu-basean sartuko da. Datu-baseko beste tauletan datuak sartzeko ere taula bakoitzeko elementuak dituen objektuak sortuko dira eta *insert_record()* funtzioaren bitartez datu-basean sartuko dira. Egilearen kasuan, datu-basean sartu baino lehen, erabiltzaileak emandako izena aurretik gordeta zegoen edo ez frogatuko du, erregistroak ez bikoizteko. Erregistroa datu-basean ondo gordetzen bada */lib/weblib.php* liburutegiko *notify()* funtzioaren bidez dena ondo joan dela jakinaraziko da, bestela funtzio berdina erabiliz zein parametro falta den jakinaraziko da.

6.3.3 Erregistroaren aldaketa

Hemen bi pauso definitu ditugu, lehenbizikoa aldatu nahi dugun erregistroa bilatzea eta bigarrena erregistroa pantailaratu eta aldaketak egiteko aukera ematea. Bi pauso hauekin hasi aurretik `/liburutegi/modify_record.php` fitxategia sortu eta `require_once("../config.php")`, `require_login()` eta `print_header()` funtzioak beti bezala erabiliko ditugu. Orain lehendabiziko pausoarekin hasiko gara, aldatu nahi den erregistroa bilatzen.

Hau egiteko bilaketa aurreraturako erabilitako kodea kopiatu eta aldaketa txiki batzuk egitearekin nahikoa izango dugu. Bilaketa aurreraturako formularioaren kodea kopiatuko dugu, baina emaitzak beste orri batean pantailaratu beharrean, orri berean pantailaratuko dira. `/liburutegi/result.php` fitxategiko kodea ere kopiatuko da emaitzak pantailaratzeko, baina zutabe berri bat gehituko emaitzen tulari. Zutabe berri honetan botoi bat erantsiko da eta `hidden` motako `HTML input` elementu bat errenkada bakoitzeko objektuaren `id` zenbakiarekin (`mdl_block_liburutegi_objektua` taularen gako, aldatu nahi den erregistroa identifikatzeko). Aldatzeko botoia sakatzerakoan `popup` leiho bat irekiko da erregistroa aldatu nahi den berresteko.

Bigarren pausoa erregistroa aldatzeko botoia sakatzerakoan hasiko da. Hau egiterakoan `/liburutegi/modify_record2.php` fitxategia irekiko da eta hautatutako erregistroen datuak pantailaratuko dira baina hauek aldatzeko aukera egongo da. Honetarako `/liburutegi/modify_record2.php` fitxategia sortu eta `require_once("../config.php")`, `require_login()` eta `print_header()` funtzioak idatziko dira. Hautatutako erregistroa zein den jakiteko `optional_param()` funtzioaren bitartez aurretik bidalitako objektuaren `id` parametroa jasoko da, eta gero `get_record_sql()` funtzioaren bitartez objektuaren datu guztiak dituen elementu bat. `SQL` sententzian `inner join` bidez taula guztiak elkartuko dira eta guk nahi dugun objektuaren `id` zenbakia duen errenkada itzuliko da. Ondoren, hautatutako objektuaren datu hauek pantailaratu behar dira eta horretarako erregistroa gehitzeko erabili den formularioaren kodea kopiatuko da, baina testu-kutxak hutsak agertu beharrean aukeratutako objektuak duen balioa pantailaratuko da. Hau egiteko `print_textfield()`, `print_textarea()` eta `choose_from_menu()` funtzioetan datu-basetik lortu dugun emaitzaren balioa parametro gisa ezarriko da. Orain erabiltzaileak datuak aldatzeko aukera izango du eta aldaketak burutu

ondoren gordetzeko botoia klikatu beharko du. Botoia sakatu ondoren, hemen ere *popup* leiho bat irekiko da aldaketak gorde nahi direla ziurtatzeko. Aldaketak egin nahi direla berretsi ondoren, erregistroa gordetzeko pantailan bezala, *insert record()* funtzioaren bidez erregistroak datu-basean gordeko dira eta *notify()* funtzioaren bidez mezu bat pantailaratuko da dena ondo joan dela adieraziz edo beharrezko eremu bat hutsa geratu dela adieraziz.

6.3.4 Erregistroaren ezabaketa

Erregistroa ezabatzeko, erregistroa aldatzeko bezala bi pauso egingo dira, erregistroa bilatu lehenengoa eta kasu honetan aukeratutako erregistroa ezabatzea. Beraz, lehenbiziko pausoa erregistroa aldatzeko egin duguna izango da, aldatuko den gauza bakarra pantailan agertuko diren izenak izango dira. */liburutegi/remove_record.php* fitxategia sortu eta *modify_record.php* fitxategiko kodea kopiatuko da bertan. Bilaketaren emaitzak agertzen diren taularen azken zutabea agertzen den aukera aldatzen da hemen, aldatu beharrean erregistroa ezabatu beharko delako, eta botoia sakatzerakoan *modify_record2.php* fitxategira jo beharrean, */liburutegi/confirm_remove.php* fitxategira jo beharko du.

/liburutegi/confirm_remove.php fitxategia sortu eta **require_once("../config.php")**, *require_login()* eta *print_header()* funtzioak idatziko dira. Hemen, *optional_param()* bidez bidalitako id parametroa jasoko da eta erabiltzaileari erregistroa ezabatzeaz ziur dagoela galdetu. Erantzuna jasotzeko bi botoi pantailaratuko dira erregistroa ezabatzeko edo ez ezabatzeko. Baiezkoan *HTML* formularioari erregistroa ezabatzeko aginduko zaio eta bestean klikatuz atzeko orrira itzuliko da. Erregistroa ezabatzeko agindua jasotzerakoan */lib/datalib.php* liburutegiko *delete_records()* funtzioa erabiliko da, atzitu nahi den taularen izena, eremuaren izena eta ezabatu nahi den erregistroaren eremu horren balioa parametro bezala pasatuz. Erregistroa datu-basetik ezabatu ondoren *notify()* funtzioaren bidez mezu bat pantailaratuko da dena ondo joan dela adieraziz. Baiezko botoia sakatu beharrean, erregistroa ez dela ezabatu nahi esanez, atzeko orrira itzuliko da.

6.3.5 Katalogoaren kudeaketa

Pantaila honetan, administratzaileak liburutegiko objektuak izango dituzten formatu, etapa eta gaiak definituko dira. */liburutegi/administratu.php* fitxategia sortu eta `require_once("../config.php")`, `require_login()` eta `print_header()` funtzioak idatziko dira. Gero `get_records_sql()` funtzioaren bitartez datu-basean dauden formatu, gai eta etapa desberdinak jasoko dira. *HTML*-z taula bat egingo da eta `print_textfield()` eta `choose_from_menu()` funtzioekin testu-kutxa eta hautatzeko eremuak jarriko dira, izen berriak gehitzeko eta daudenak ezabatu ahal izateko. Hauen ondoan botoiak jarriko dira, haiek sakatzerakoan orri berdina deitu eta zein botoi sakatu den identifikatuko da zein eragiketa egin behar den jakiteko. Bai formatu, bai etapa eta bai gai bat gehitu nahi denean, aurretik izen hori ez dagoela kontrolatu behar da izen hori datu-basean bilatuz. Dagoeneko datu-basean baldin badago ez da erregistroa datu-basean gordeko eta mezu bat pantailaratuko du erregistroa dagoeneko datu-basean dagoela esanez. Aukera hutsa sartu nahi bada ere mezu bat pantailaratuko du testua sartzeko eskatuz. Dena ondo baldin badago erregistroa ondo gorde dela jakinaraziko da eta erregistroa ezabatu nahi bada, ondo egin dela ere pantailaratuko da. Aukera guztietan, */liburutegi/administratu.php* orrira itzultzeko botoi bat ere pantailaratuko da.

6.3.6 Hizkuntza fitxategiak

Hitz kate guztiak beste hizkuntzetara itzulgarriak izan behar dira, honetarako `get_string()` eta `print_string()` funtzioak erabiltzen dira. Funtzio hauek blokeek duten *lang* karpetan dauden fitxategiak aztertzen ditu eta aukeratutako hizkuntzan pantailaratzen ditu hitz kateak. Kasu honetan */blocks/liburutegi/lang* karpetan hiru karpeta egin dira, *en_utf8* ingeleserako, *es_utf8* gaztelararako eta *eu_utf8* euskararako. Karpeta hauetan *block_liburutegi.php* izeneko fitxategia sortuko dugu eta aplikazioen kodean `get_string()` eta `print_string()` bidez jarri ditugun hitz kateak hemen itzuliko dira. Adibidez, izenburua hitz katea gaztelarara itzuli ahal izateko hau idatziko da *es_utf8* karpetako *block_liburutegi.php* fitxategian `$string['izenburua']="titulo";` ingelesara itzultzeko *en_utf8* karpetako fitxategian `$string['izenburua']="title";` eta euskarara itzultzeko *eu_utf8* karpetako fitxategian `$string['izenburua']="izenburua";` idatziko da.

6.3.7 Baimenen definizioa

Moodle-ek erabiltzaile desberdinak ditu, eta hauek rol desberdinetan multzokatzen dira [5]. Lehenetsitako rola administratzailea, ikastaro sortzailea, irakaslea, editatzeko baimenik gabeko irakaslea, ikaslea eta gonbidatua dira. Liburutegi blokearen kasuan, segurtasun kontuak direla eta, erabiltzaile guztiek ezingo dute baimen berdinak izan, adibidez ikasle batek ezin izango du datu-basetik erregistrorik ezabatu. Baimen hauek definitzeko */blocks/liburutegi/db* karpetan *access.php* fitxategia sortuko da, eta bertan definituko da erabiltzaile bakoitzak blokearen zein aukera izango dituen ikusgai. Erregistroak bilatzeko aukera edozein erabiltzailek izango du. Hau horrela definituko da:

```
'block/liburutegi:search' => array(
    'capytype' => 'write',
    'contextlevel' => CONTEXT_BLOCK,
    'legacy' => array(
        'guest' => CAP_ALLOW,
        'student' => CAP_ALLOW,
        'editingteacher' => CAP_ALLOW,
        'teacher' => CAP_ALLOW,
        'coursecreator' => CAP_ALLOW,
        'admin' => CAP_ALLOW
    )
)
```

Ilustrazioa 32: erregistroak bilatzeko baimenen definizioa

Datu-basean erregistroak gehitzeko aukera administratzaileak, ikastaro sortzaileak eta editatzeko baimena duten irakasleek izango dute.

```
'block/liburutegi:add' => array(
    'capytype' => 'write',
    'contextlevel' => CONTEXT_BLOCK,
    'legacy' => array(
        'guest' => CAP_PROHIBIT,
        'student' => CAP_PROHIBIT,
        'editingteacher' => CAP_ALLOW,
        'teacher' => CAP_PROHIBIT,
        'coursecreator' => CAP_ALLOW,
        'admin' => CAP_ALLOW
    )
)
```

Ilustrazioa 33: erregistroak gehitzeko baimenen definizioa

Datu-baseko erregistroak aldatzeko aukera ere administratzaileak, ikastaro sortzaileak eta editatzeko baimena duten irakasleek izango dute.

```
'block/liburutegi:modify' => array(  
    'capytype' => 'write',  
    'contextlevel' => CONTEXT_BLOCK,  
    'legacy' => array(  
        'guest' => CAP_PROHIBIT,  
        'student' => CAP_PROHIBIT,  
        'editingteacher' => CAP_ALLOW,  
        'teacher' => CAP_PROHIBIT,  
        'coursecreator' => CAP_ALLOW,  
        'admin' => CAP_ALLOW  
    )  
)
```

Ilustrazioa 34: erregistroak aldatzeko baimenen definizioa

Datu-baseko erregistroak ezabatzeko baimena izango dituzten erabiltzaileak erregistroak aldatzeko eta gehitzeko baimenak dituzten erabiltzaile berdinak izango dira, baina aukera hauek lehenetsitakoak dira, horregatik ez dira baimen hauek bateratzen, banatuak egiten dira administratzaileak rolak definitzeko garaian baimen hauek definitu ditzakeelako.

```
'block/liburutegi:delete' => array(  
    'capytype' => 'write',  
    'contextlevel' => CONTEXT_BLOCK,  
    'legacy' => array(  
        'guest' => CAP_PROHIBIT,  
        'student' => CAP_PROHIBIT,  
        'editingteacher' => CAP_ALLOW,  
        'teacher' => CAP_PROHIBIT,  
        'coursecreator' => CAP_ALLOW,  
        'admin' => CAP_ALLOW  
    )  
)
```

Ilustrazioa 35: erregistroak ezabatzeko baimenen definizioa

Azken baimena katalogoa kudeatzeko baimena izango du, hau administratzaileak eta ikastaro sortzaileak soilik egin ahal izango dute.

```
'block/liburutegi:manage' => array(
    'capytype' => 'write',
    'contextlevel' => CONTEXT_BLOCK,
    'legacy' => array(
        'guest' => CAP_PROHIBIT,
        'student' => CAP_PROHIBIT,
        'editingteacher' => CAP_PROHIBIT,
        'teacher' => CAP_PROHIBIT,
        'coursecreator' => CAP_ALLOW,
        'admin' => CAP_ALLOW
    )
)
```

Ilustrazioa 36: katalogoa kudeatzeko baimenen definizioa

Baimen guztiak definitu ondoren administratzaileak erabiltzaileak menuan, baimenak topatuko ditu. Hemen rola definitzeko aukera emango da rola editatuz.

Baimenak ?				
Gaitasuna	Ezarri gabea Baimendu Debekatu Ukatu Arriskuak			
Liburutegia				
Liburutegian erregistroak gehitu <small>block/liburutegi:add</small>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>
Liburutegitik erregistroak ezabatu <small>block/liburutegi:delete</small>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>
Liburutegian erregistroak aldatu <small>block/liburutegi:modify</small>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>
Liburutegian erregistroak bilatu <small>block/liburutegi:search</small>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>

Ilustrazioa 37: baimenak aldatzeko pantaila

6.4 EMAITZAK

Emaitza egokia izan den edo ez erabakitzeko aplikazioan hutsuneak bilatuko dira. Erroreak topatuz gero hauentzako irtenbideak sortuko dira eta errorik ez badago ikastetxeko arduradunei egindako lana aurkeztuko zaie.

Erroreak bilatzeko baimenak ondo definitu diren aztertuko da, hau da, erabiltzaile desberdinekin baimenduta ez dituen eragiketen saiakerak egingo dira. Gero *phpMyAdmin*

erabiliz datu-baserako atzipenak ondo egiten diren arakatuko da. Honetarako garatutako aplikazioaren bitartez erregistroak gehituko ditugu eta nabigatzailearen leiho batean *phpMyAdmin* ireki eta *Moodle*-eko datu-basean blokeko taulak bilatu eta bertan sartu ditugun datuak ondo barneratu diren frogatuko da. Gauza berdina egingo da erregistroak aldatuz, erregistroak ezabatuz eta katalogoa kudeatu aukerarekin. Honetaz aparte aukera guztietan botoiak sakatuko ditugu hauen eraginkortasuna probatzeko eta erregistroak gehitzeko eta aldatzeko orrietan proba ezberdinak egingo dira eremu batzuk hutsak utziz eta aplikazioaren emaitza aztertuz bai pantailaratzen denarekin baita *phpMyAdmin*-ekin datu-basean gordetzen denarekin.

Dena ondo joan denean, blokea *Moodle*-etik ezabatu eta *phpMyAdmin*-ekin liburutegi blokeko taulak datu-basetik ezabatuko dira. Honen ondoren blokearen instalazioa ondo dabilen ikertuko da, blokea berriz instalatuz. Hau egiteko administratzaileak jakinarazpenak klikatzearekin nahikoa izango da. Dena ondo baldin badoa, instalazio prozesua pantailaratuko da 38. irudian erakusten den bezela. Bidean erroreren bat gertatzen bada *Moodle*-ek jakinaraziko du, eta dena ondo badoa prozesua arrakastaz burutu dela esango du.

```
(mysql): CREATE INDEX mdl_blockliburtegiobj_obj_ix ON mdl_block_liburtegi_formatu_obj (obj_id)
Ondoi
(mysql): CREATE TABLE mdl_block_liburtegi_gaia ( id BIGINT(10) unsigned NOT NULL auto_increment, izena VARCHAR(20) NOT NULL DEFAULT '', CONSTRAINT PRIMARY KEY (id) )
Ondoi
(mysql): ALTER TABLE mdl_block_liburtegi_gaia COMMENT='liburtegiako objektuen gaien zerrenda'
Ondoi
(mysql): CREATE TABLE mdl_block_liburtegi_gaia_obj ( id BIGINT(10) unsigned NOT NULL auto_increment, gai_id BIGINT(10) unsigned NOT NULL, obj_id BIGINT(10) unsigned NOT NULL, CONSTRAINT PRIMARY KEY (id) )
Ondoi
(mysql): ALTER TABLE mdl_block_liburtegi_gaia_obj COMMENT='liburtegiako objektuen eta gaien arteko erlazioak'
Ondoi
(mysql): CREATE INDEX mdl_blockliburtegiobj_gai_ix ON mdl_block_liburtegi_gaia_obj (gai_id)
Ondoi
(mysql): CREATE INDEX mdl_blockliburtegiobj_obj_ix ON mdl_block_liburtegi_gaia_obj (obj_id)
Ondoi
liburtegia taulak egoki sortu dira.
Jarraila
```

Ilustrazioa 38: blokea ondo instalatzerakoan lortutako pantaila

Hau guztia ondo joan bada La Salle Berrozpeko arduradunei garatutako lana aurkeztuko zaie. Hauek aplikazioa ikusi eta probatu ondoren azkeneko eskaera bat egin dute, datu-baseko objektuei beste eremu batzuk gehitzea. URL izeneko eremu bat sarean egon daitezkeen objektuentzako, deskribapen didaktikoa eremua izenak berak esaten duen bezala objektuaren informazio pedagogikoa eransteko eta UDC eremu bat. UDC (*Universal Decimal Classification*) sailkapen hamartar unibertsala euskaraz, liburutegietan lanak sailkatu eta kategorizatze beharrarengatik sortu zen jakinduriaren sailkapen sistema bat da [7]. Hiru eremu hauek sartzea ez dauka zailtasunik, XMLDB editorearekin *mdl_block_liburutegi_objektua* taula editatuko da hiru eremu hauek gehituz, hiruak char motakoak izango dira baina luzera desberdinekoak deskribapen didaktikoa eta URL 255 bitekoak izango dira eta UDC 20 bitekoa. Azkenik erregistroa bilatzeko pantailan UDC testu-kutxa gehituko da eta bilaketaren emaitzaren pantailako taulan zutabe bana gehituko da eremu berri bakoitzeko. Hau erregistroa aldatzeko eta ezabatzeko pantailatan ere aldatuko da.

Emaitzen irudi gehiago ikusteko La Salle Berrozpeko Moodle-en irudiak bukaeran dauden eranskinean ikusi daitezke.

Proiektuan zehar egin den plangintzari begira esan behar da planifikatutako egutegia nahiko ondo bete dela. Ataza guztiak bete dira espero zen egutegiaren barruan. Nahiz eta ordu kopuru zehatza ez den kontatu proiektuaren kudeaketan eta probak egiteko garaian espero zena baina ordu gehiago sartu behar izan dira. Proiektuaren kudeaketaren kasuan memoriaren idazketa espero baino luzeagoa izan delako eta proben garaian, ikastetxeko lan taldeko arduradunak azken momentuko eskakizunak egin dituelako. Trebakuntzan aldiz, espero baino ordu gutxiago eman dira Moodle-ek dituen funtzioen erabilpena oso sinplea delako. Hala ere, ordu batzuk etxean sartu behar izan badut ere planifikatutako egutegia bete egin da.

7. ONDORIOAK

Proiektuaren inplementazioari amaiera emanda eta lortutako emaitzak aztertuta finkatutako helburuak bete diren edo ez aztertu daiteke. Lehenbiziko helburua liburutegi bateko erregistroa eramateko gai den aplikazio bat diseinatzea eta garatzea zen eta aplikazio hau *Moodle* plataforman integratzea. Aplikazio honen bidez ikastetxeko liburutegian nahiz sarean dauden edozein formatuko erregistroak bilatzeko aukera desberdina eskaini behar zen, bilaketa simple bat eginez baita metadatura bilaketa zehatzagoa egiteko aukera eman ere. Erregistratutako katalogoa eguneratzeko aukera ere eskaini behar zen. Hau da, erregistratutako objektuak aldatzeko aukera eman behar zen, baina baita objektu berriak erregistratzeko eta erregistratuta dauden objektuak ezabatzeko aukera ere eskaini behar zen. Hau dena *Moodle* plataforman integratu behar zen eta plataforma honen erabiltzaile desberdinentzat baimen desberdinak kudeatzeko aukera ere eman beharra zegoen katalogoaren segurtasuna bermatuz, erabiltzaile bakoitzari aplikazioaren eragiketa batzuk bakarrik esleituz. Hau guztia eta lortutako emaitzak kontutan hartuta esan daiteke aplikazio honen eraginkortasunari begira zeuden helburuak bete egin direla.

Proiektu hau egin ondoren ateratako ondorioak ez dira bakarrik garatu den aplikaziora murrizten, garatzailearen ezagutza aldetik ere aurrerapenak lortu dira. Helburu pertsonal hauei begira, pozik geratzen naiz *PHP* eta *HTML* lengoaietan programatzen trebatu naizelako eta nire burua gai ikusten dudalako proiektu berriak garatzeko. Honetaz aparte *Moodle* bezalako plataforma batean sakontzeko aukera izan dut bai erabiltzaile moduan, bai administratzaile moduan baita garatzaile moduan ere. Gainera, informatika munduko proiektu baten garapena nolakoa den ezagutu dut eta proiektua La Salle Berrozpe ikastetxean egin denez enpresa mailan nola antolatu beharra dagoen ere ikasi dut.

Ikastetxearen aldetik zeuden helburua bete den a la ez esatea zailagoa da, erabilpen gutxi zuen ikastetxeko liburutegia ikasleei hurbiltzea zen helburu hau, baina hau guztiz bete den edo ez jakiteko denbora beharko da, horrela bakarrik jakingo delako liburutegian dauden objektuak gehiago erabiltzea bultzatu den. Baina eraginkortasunari begira, blokea probatu duten irakasleek gustura bukatu dute lortutako emaitzarekin eta erabilgarria izango dela esan dute hurrengo ikastaroari begira. Batez ere sarean dauden baliabideak eskaintzeko aukera interesgarria dagoela ondorioztatu dute.

Etorkizunera begira, hobekuntza desberdinak planteatu daitezke. Oraingo proiektuan ez da maileguen erabilpena planteatu, hau da, liburutegiko liburuen maileguak *Moodle*-en bitartez

kudeatzea. Baina liburutegiko objektuen kontsulta espero den bezala handitzen bada, liburutegiko mailegu zerbitzua *Moodle*-en bitartez kudeatzea planteatu daiteke egin beharreko tramiteak errazteko eta eskuragarriago izateko. Beste hobekuntza bat sarean dauden erregistroak *Moodle*-en irekitako leihoetan ikusteko aukera ematea izango zen eta objektuak bilatzen direnean bilaketa emaitzak pantailaratzen diren taularen beste zutabe batean objektuak aurreikusarazteko aukera ematea. Etorkizunera begira ere, garatutako blokearen funtzionamendu aztertu beharko da *Moodle*-en bertsio berrietan. Aplikazioa garatzeko garaian *Moodle*-ek dituen liburutegi eta funtzio desberdinak erabili dira, horregatik baliteke aldaketaren bat egin behar izatea bertsio berrietan liburutegiak aldatzen badira.

8. BIBLIOGRAFIA

Liburuak

[1] Moodle 1.9 Extension Development
Jonathan Moore, Michael Churchward, 2010

[2]Diseño de paginas web
ediciones cae, 2011

Dokumentuak

[3]Guia de apoyo para el uso de moodle 1.9.4 Usuario desarrollador
Ana Teresa Gonzalez de Felipe
http://download.moodle.org/docs/es/1.9.4_usuario_desarrollador.pdf

Webguneak

[4]blokeen garapena
http://docs.moodle.org/all/es/Desarrollo_de_bloques#Variables_de_clase:

[5]rolen definizioa
<http://docs.moodle.org/dev/Roles#Capabilities>

[6]kodearen estilorako manuala
http://docs.moodle.org/all/es/Manual_de_Estilo_de_C%C3%B3digo

[7]UDC-ari buruzko informazioa
<http://udcdata.info/066027>

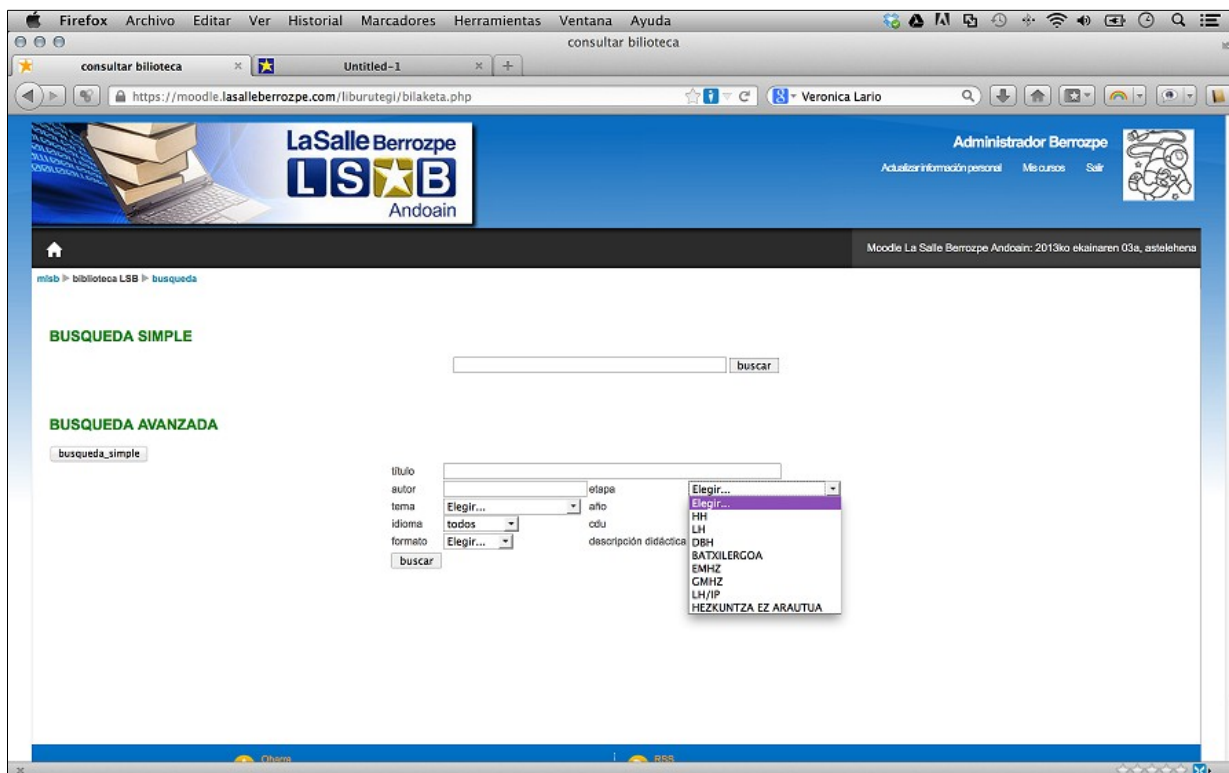
[8]wikipedia
<http://eu.wikipedia.org/wiki/Azala>

9. ERANSKINA

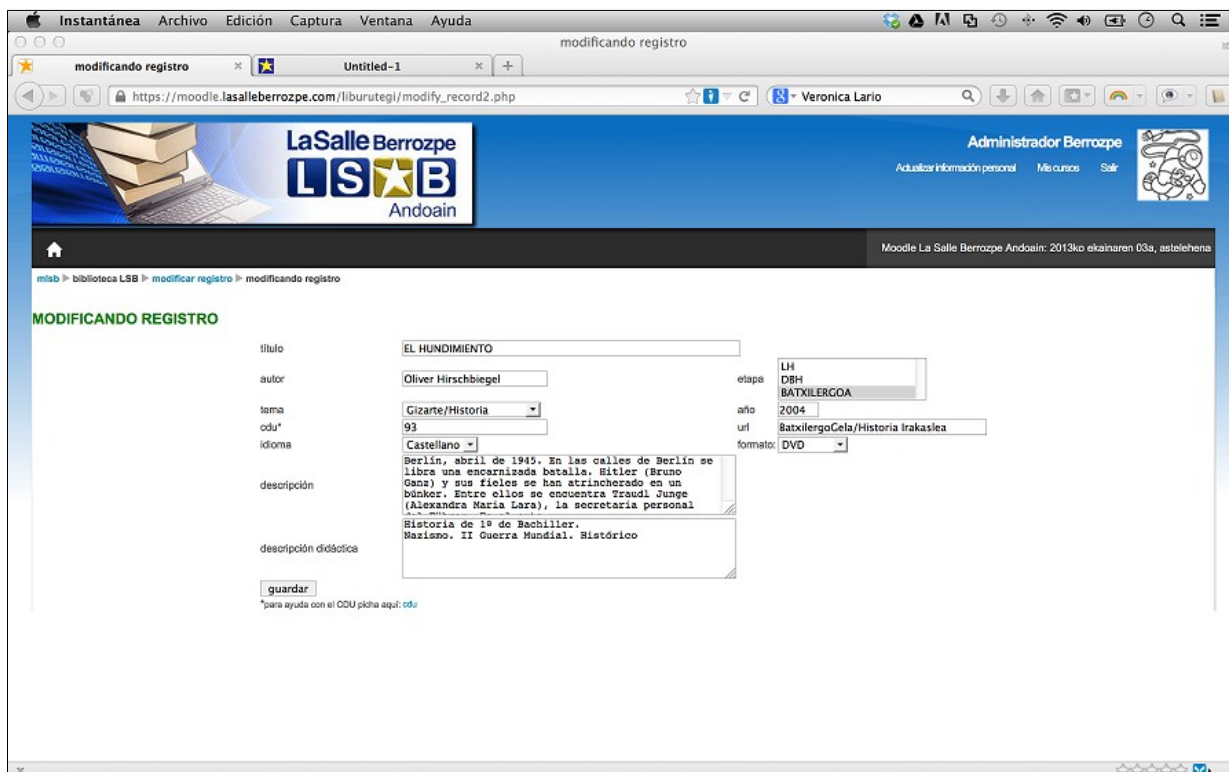
Eranskin honetan La Salle Berrozpe ikastetxean dagoen Moodle-ean liburutegi blokea instalatu eta erabiltzen hasi ondoren ateratako irudi batzuk aurkezten dira.

Titulua	Autorea	Generoa	Idazlea	Urtea	Formatua	Bolumena	Denbora	URLa	Deskribapena	Erantsitako irudiak
EKIPO LANA/TRABAJO EN EQUIPO	Anonomia	DBH BATXILERGOA EMHZ LMIIP HEZKUNTZA EZ ARAUTUA	Ingelesa	2010	YOUTUBE	Balioak	3303	http://www.youtube.com/watch?v=mEDpU1PpI	Bi adibide ekipo lana adierazteko, humore puntuko batekin	Ekipoan lana egiten denean lor daitezkeen emaitzak.
ELIKADURA SUBIROTASUNA	Hegoari Begira	DBH BATXILERGOA	Euskera	2004	DVD	PRO-PROCEGA-Solidar	314.1	BatxilergoGela4, Armerioa/1go Apela	Mundua mailan elikaduran buruz dagoen egoera, etorkizuna	DBH 2. zikloan, Zehar lerroak eta Gizarte zientziak. BATXILERGOA: Zehar lerroak: Filosofia, Historia, Ekonomia, geografia
EMIGRAZIOA	Hegoari Begira	DBH BATXILERGOA	Euskera	2003	DVD	Gizarte/Historia	314.1	BatxilergoGela4, Armerioa/1go Apela	Munduko Hegoaldetik abiatuta emigrazioaren deskribapena eta egungo egoera	DBH 2. Zikloko gara: Zehar lerroak eta Gizarte Zientziak. BATXILERGOA: Lanitzeko aukerak: Filosofia, Historia, Ekonomia, Historia Garaikidea, Filosofaren Historia
LA PASION DE CRISTO	Mel Gibson	DBH BATXILERGOA EMHZ GMHZ	Castellano	2004	DVD	Erljioa	2232	Btxilego gela/4, armeriuan/1go arasan	Nazaretoko Jesusen azken egunetako historia. Oso oinarria	Teologia ikuspuntutik gaia aldez aurretik azaldu behar da.
LA TIERRA, PLANETA VIVO: FÓSILES A TRAVÉS DEL TIEMPO	Inst. Geologiko	DBH	Castellano	2003	DVD	Natur Zientziak	553.2	BatxilergoGela4, Armerioa/1go Apela	Transmitir unos conocimientos básicos en geología y paleontología.	Uso de los registros paleontológicos más importantes de la Península Ibérica realizando un viaje desde el pasado. Más detalle en la contraportada del DVD.
NI UNO MENOS	Zhang Yimou	DBH BATXILERGOA EMHZ GMHZ	Castellano	1999	DVD	Balioak	336	Pastotaltza Anduraduna	Una joven maestra de 13 años realiza un viaje a la gran ciudad para leer de vuelta a un niño que, obligado por las circunstancias económicas de su hogar, ha dejado la escuela para buscar trabajo. Si no regresa la maestra no recibirá su salario.	Valores que se pueden trabajar: Pobreza, Responsabilidad, Autenticidad. Desde ESO 4 en adelante.
THE MASK ZORRO	xxxxxxxxxx	DBH BATXILERGOA	Ingelesa	1998	DVD	Balioak	8811	BatxilergoGela/Ingelesezko Irakaslea	Kaliskoa den Zorro pertsonariari buruz egindako pelikua. Protagonista:	DBH eta BATXILERGOAN Ingelesaren entzumenaren lantzeko. Ondo ulertzen den

La Salle Berrozpeko Moodle-eko liburutegiko blokean bilaketaren emaitza



La Salle Berrozpeko Moodle-eko liburutegiko bilaketaren pantaila



La Salle Berrozpeko Moodle-eko liburutegiko erregistro baten aldaketaren pantaila