

eman ta zabal zazu



Universidad
del País Vasco

Euskal Herriko
Unibertsitatea

Informatika Ingeniaritzako Gradua
Software Ingeniaritza

Gradu Amaierako Proiektua

QR kodeak eta eduki-kudeaketa eleanitza

Egilea

Manex Garaio Mendizabal

informatika
fakultatea



facultad de
informática

2014

Laburpena

Gradu Amaierako Proiektu honek QR kodeak eta eduki-kudeaketa eleanitza jorratzen ditu. Hori egiteko hiru osagai nagusitan banatzen da. Batetik, QR kodeak irakurtzen dituen *Android* Sistema Eragilerako aplikazio bat dago. Bestetik, *Android* aplikazio horrek atzitu duen Eduki Kudeatzailea dago, *WordPress* Edukiak Kudeatzeko Sistemaren bidez garatu dena eta edukiak hizkuntz ezberdinetan izango dituen. Azkenik, Eduki Kudeatzaile hori osatzen duen gehigarria dago, sarean dauden zerbitzuak erabilia webguneari itzulpenak egiteko eta QR kodeak sortzeko gaitasunak emango dizkiona.

Gaien aurkibidea

Laburpena	i
Gaien aurkibidea	iii
Irudien aurkibidea	vii
Taulen aurkibidea	xiii
Esker onak	1
1 Sarrera eta Helburuak	3
1.1 Motibazioa	3
1.2 Helburuak	5
1.3 Proiektuaren irismena	5
1.3.1 Betekizunak	5
1.3.2 Lanaren Deskonposaketa Egitura	6
1.3.3 Emangarriak	9
1.3.4 Mugarriak	10
2 Aurrekariak eta Erabilitako Teknologia	13
2.1 Aurrekariak	13
2.1.1 QRpedia	13

iii

2.1.2	Word Lens	14
2.2	Teknologia posibleak	15
2.2.1	Aplikazioa	15
2.2.2	Eduki Kudeatzailea	17
2.3	Erabilitako teknologia	21
2.3.1	Android Sistema Eragilea	21
2.3.2	QR kodeak	26
2.3.3	WordPress	27
2.3.4	Apertium	29
3	Analisia eta Diseinua	31
3.1	Ohar Eleanitzak aplikazioa	31
3.1.1	Analisia	31
3.1.2	Diseinua	32
3.2	Eduki Kudeatzailea	51
3.2.1	Analisia	51
3.2.2	Diseinua	52
4	Garapena eta Probak	77
4.1	Garapena	77
4.1.1	Ezaugarri Teknikoak	77
4.1.2	Internazionalizazio estrategia	79
4.1.3	Interfaze grafikoa	80
4.2	Probak	82
4.2.1	Probetarako erabilitako baliabideak	82
4.2.2	Garapen zikloak	83
4.2.3	Produktuaren portaera ezegokiak	86
4.2.4	Ondartxo museorako probak	88

5 Ondorioak eta Etorkizuneko Aukerak	89
5.1 Ondorioak	89
5.2 Etorkizuneko Aukerak	90
Bibliografia	93
Eranskinak	
A Proiektuaren Kudeaketa	99
A.1 Plangintza	99
A.1.1 Denboraren planifikazioa	99
A.2 Lan metodologia	101
A.3 Baliabideen esleipena	102
A.4 Kalitate plana	103
A.5 Arriskuen plana	106
A.6 Jarraipen txostena	109
A.6.1 Desbideraketak	109
A.6.2 Kalitatea	111
A.6.3 Arriskuak	113
B Erabilera Gida	115
B.1 Ohar Eleanitzak aplikazioa	115
B.2 Eduki Kudeatzailea	126

Irudien aurkibidea

1.1	Proiektuaren LDEaren irudia.	7
1.2	Proiektuaren mugarriak denboran zehar islatzen dituen irudia.	11
2.1	<i>Android</i> -en arkitektura erakusten duen irudia.	22
2.2	<i>Android</i> jarduera baten bizitza-zikloa irudikatzen duen grafikoa.	24
3.1	Aplikazioaren klase diagrama.	33
3.2	Aplikazioaren datu-basearen diagrama.	43
3.3	Aplikazioaren erabilpen-kasuen diagrama.	44
3.4	QR kodea irakurri erabilpen-kasuaren sekuentzia-diagrama.	46
3.5	QR kodea irakurri (birbideratze automatikoa) erabilpen-kasuaren sekuentzia-diagrama.	47
3.6	Ezarpenak aldatu erabilpen-kasuaren sekuentzia-diagrama.	48
3.7	Irakurketa historiako elementu bat ireki erabilpen-kasuaren sekuentzia-diagrama.	49
3.8	Irakurketa historia ezabatu erabilpen-kasuaren sekuentzia-diagrama.	50
3.9	Hizkuntza aldatu erabilpen-kasuaren sekuentzia-diagrama.	51
3.10	WordPress EKS-aren datu-basearen diagrama.	54
3.11	Eduki Kudeatzailean erabiltzen diren WordPress EKS-aren berezkoak diren erabilpen-kasuen diagrama.	55
3.12	<i>Ohar Itzultzailea</i> gehigarriaren egitura adierazten duen irudia.	62

3.13	<i>Ohar Itzultzailea</i> gehigarriaren erabilpen-kasuen diagrama.	67
3.14	Itzultzailea ireki erabilpen-kasuaren sekuentzia-diagrama.	69
3.15	Itzulpena egin erabilpen-kasuaren sekuentzia-diagrama.	70
3.16	Itzulpena gorde erabilpen-kasuaren sekuentzia-diagrama.	72
3.17	Itzulpena gorde eta itzultzailea itxi erabilpen-kasuaren sekuentzia-diagrama.	73
3.18	Itzultzailea itxi erabilpen-kasuaren sekuentzia-diagrama.	74
3.19	Eduki-elementuaren QR kodea sortu erabilpen-kasuaren sekuentzia-diagrama.	75
A.1	Denboran zehar emango den atazen banaketa erakusten duen Gantt diagrama.	101
B.1	<i>Ohar Eleanitzak</i> aplikazioaren ikonoa <i>Android</i> gailuaren menu nagusian.	116
B.2	<i>Ohar Eleanitzak</i> aplikazioaren hasiera pantaila erakusten duen irudia.	116
B.3	Aplikazioak kode bat irakurtzean erakusten duen konfirmazio mezua.	117
B.4	Kodeak gordetzen zuen helbidearen edukia, gailuaren arakatzailan ikusita.	118
B.5	Kodeak gordetzen zuen helbidearen edukia, gailuaren arakatzailan ikusita.	118
B.6	<i>Ohar Eleanitzak</i> aplikazioaren aukeren barra <i>Ezarpenak</i> aukera irekitzeko.	119
B.7	<i>Birbideratzea Automatikoki Onartu</i> ezarpena ezgaituta.	119
B.8	<i>Birbideratzea Automatikoki Onartu</i> ezarpena gaituta.	120
B.9	<i>Ohar Eleanitzak</i> aplikazioaren aukeren barra <i>Hizkuntzak</i> aukera irekitzeko.	120
B.10	Hizkuntzak aukeratzeko hautatzailea.	121
B.11	Aplikazioak dauzkan hizkuntzen zerrenda.	121
B.12	Aplikazioaren hasierako pantaila, aplikazioaren hizkuntza ingelesera aldatu ostean.	122
B.13	<i>Ohar Eleanitzak</i> aplikazioaren aukeren barra <i>Historia</i> aukera irekitzeko.	122
B.14	Erabiltzailearen irakurketa historia.	123
B.15	<i>Ohar Eleanitzak</i> aplikazioaren aukeren barra <i>Historia</i> aukera irekitzeko.	123
B.16	<i>Historia garbitu</i> aukera erakusten duen aukeren barra.	124

B.17 Historia garbitu aurretik agertzen den oharra.	124
B.18 Erabiltzailearen irakurketa historia garbitu ostean, historia hutsik.	125
B.19 Eduki Kudeatzailean saioa hasteko orrialdea.	126
B.20 Eduki Kudeatzailearen kudeaketarako hasiera orrialdea.	127
B.21 Eduki Kudeatzailean Bidalketa berri bat sortzeko aukera.	127
B.22 Eduki Kudeatzailean Bidalketa berri bat sortzeko editorea.	128
B.23 Eduki Kudeatzailean Bidalketa berri bat publikatzeko sakatu beharreko botoia.	128
B.24 Eduki Kudeatzailean dauden Bidalketak ikusteko sakatu beharreko botoia.	129
B.25 Gunean dauden Bidalketen zerrenda erakusten duen irudia.	129
B.26 Eraldatu nahi den Bidalketa editorean erakusten duen irudia.	130
B.27 Bidalketa eguneratzeko sakatu beharreko botoia.	130
B.28 Eduki Kudeatzailean dauden Bidalketak ikusteko sakatu beharreko botoia.	131
B.29 Bidalketa ezabatzeko sakatu beharreko botoia.	131
B.30 Eduki Kudeatzailean Orrialde berri bat sortzeko aukera.	132
B.31 Eduki Kudeatzailean Orrialde berri bat sortzeko editorea.	132
B.32 Eduki Kudeatzailean Orrialde berri bat publikatzeko sakatu beharreko botoia.	133
B.33 Eduki Kudeatzailean dauden Orrialdeak ikusteko sakatu beharreko botoia.	133
B.34 Gunean dauden Orrialdeen zerrenda erakusten duen irudia.	134
B.35 Eraldatu nahi den Orrialdea editorean erakusten duen irudia.	134
B.36 Orrialdea eguneratzeko sakatu beharreko botoia.	135
B.37 Eduki Kudeatzailean dauden Orrialdeak ikusteko sakatu beharreko botoia.	135
B.38 Orrialdea ezabatzeko sakatu beharreko botoia.	136
B.39 Eduki Kudeatzailean Kategoriak ikusteko atalera joateko.	136
B.40 Kategori berri bat sortzeko erabiltzen inprimakia.	137

B.41 Eduki Kudeatzailean Kategoriak ikusteko atalera joateko.	138
B.42 Gunean dauden Kategorien zerrenda erakusten duen irudia.	138
B.43 Eraldatu nahi den Kategoriaren datuak erakusten dituen orrialdea.	139
B.44 Eduki Kudeatzailean Kategoriak ikusteko atalera joateko.	139
B.45 Kategoria ezabatzeko sakatu beharreko botoia.	140
B.46 Eduki Kudeatzailean egitura sortzeko atalera joateko.	140
B.47 Menu berri bat sortzeko sakatu beharreko esteka.	141
B.48 Menu berriari izena emateko orrialdea.	141
B.49 Menu berriaren edukiak eta kokapena zehazteko orrialdea.	142
B.50 Menu berria, edukia gehitu ondoren.	143
B.51 Eduki Kudeatzailean egitura sortzeko atalera joateko.	144
B.52 Eraldatu nahi den menua aukeratzeko hautatzailea.	144
B.53 Menuaren edukiak erakusten dituen irudia.	145
B.54 Eduki Kudeatzailean hizkuntzen atalera joateko.	145
B.55 Eduki Kudeatzailean hizkuntza berri bat definitzeko atala.	146
B.56 Eduki Kudeatzailean hizkuntzen atalera joateko.	147
B.57 Eduki Kudeatzailean definituta dauden hizkuntzen zerrenda.	147
B.58 Eduki Kudeatzailean gaituta dauden hizkuntzen zerrenda.	148
B.59 Eduki Kudeatzailean hizkuntzen atalera joateko.	148
B.60 Eduki Kudeatzailean definituta dauden hizkuntzen zerrenda.	149
B.61 Editorearen irudia, itzultzailea irekitzeko botoia gorriz nabarmenduta daukalarik.	150
B.62 Itzultzailearen irudia, izenburuak itzultzea aukeratuta dagoenean.	151
B.63 Itzultzailearen irudia, testuak itzultzea aukeratuta dagoenean.	151
B.64 Itzultzailearen irudia, jatorriko hizkuntza gisa gaztelania hautatuta dagoenean.	152

B.65 Itzultzailearen irudia, gaztelaniatik frantsesera itzulpena egin ostean.	153
B.66 Itzulpena gorde dadin editorean sakatu beharreko botoia.	153
B.67 QR sortailea irekitzeko botoia erakusten duen irudia.	154
B.68 Bidalketa/Orrialdearen QR kodea erakusten duen orrialdea.	154

Taulen aurkibidea

2.1	Hizkuntz-bikote batzuen eraginkortasunari buruzko datuak erakusten dituen taula.	30
4.1	Ohar Eleanitzak aplikazioaren ezaugarri teknikoak laburtzen dituen taula.	78
4.2	Eduki Kudeatzailearen ezaugarri teknikoak laburtzen dituen taula.	78
A.1	Atazen denbora estimazioa erakusten duen taula.	100
A.2	Atazen denboraren estimazioen eta dedikatutako denboraren datuak erakusten dituen taula.	110

Esker onak

Proiektuari buruzko xehetasunak azaltzen hasi aurretik proiektu hau aurrera eramateko funtsezkoak izan diren pertsonen nire eskerrik beroena eman nahiko nieke:

- **Iñaki Alegria** eta **Kepa Sarasolari** proiektu honetan lan egiteko aukera emateagatik. Biek osatutako *tandemari* esker proiektuan lan egitea asko erraztu zait, behar nuenean aholkularitza egokia jaso izanagatik. Beraien bultzadarengatik ez balitz proiektu honek ez luke *bizitza errealeko* kasu batean ezarria izateko aukerarik izango.
- **Isaac Saiz**-i *Android* aplikazioa eta webgunea katalanera itzultzen laguntzeagatik, proiektua hobetzeko ideiak emateagatik eta ia egunero *pinta* batzuk edatera joateko prest egoteagatik.
- **Philippe Valensa**-ri *Android* aplikazioa eta webgunea frantsesera itzultzen lagundu izanagatik.
- **Julie Brake**-ri *Android* aplikazioa eta webgunea galeserara itzultzeagatik, bukaeran erabili ez bada ere. Horretaz gain, Galeseko egoera nazional eta kulturalari buruz emandako informazioa eta galeserako irakasle bikaina izana eskertu nahiko nioke. *Diolch yn fawr! Dysgais i llawer o Cymraeg yn y dosbarth, chwarae teg. Dw i eisiau i mynd i Cymru yn fuan.*
- Galesen nengoela ezagutu nituen **lagun** eta **klasekide** guztiei, espero dut berriro ere zuek ikusteko aukera izatea.
- Azkenik, kanpoan nengoela hainbeste faltan bota izan ditudan **lagun** eta **familiari** eskertu nahiko nieke, itsaso batek banatzen gintuen arren gertu izan ditudalako beti.

1. KAPITULUA

Sarrera eta Helburuak

Memoriaren lehen atal honetan proiektuari buruzko sarrera labur bat egingo da. Bertan, proiektuaren garapena hasteko egon zen motibazioari buruz hitz egingo da eta baita proiektuak bete beharko lituzkeen hainbat helbururi buruz. Horretaz gain, proiektuaren irismenari buruz ere hitz egingo da: Lanaren Deskonposaketa Egitura, emangarriak... Proiektuaren plangintzaren beste atal batzuk ikusi nahi izatekotan, jarraipena barne, *Proiektuaren Kudeaketa* eranskinean daude irakurgarri.

1.1 Motibazioa

Gaur egun, herrialde garatuetako biztanle gehienek eskutan edo eskura daude mugikor adimentsu eta tabletak. Horrek gizartearen komunikazio ohituretan aldaketa itzela ekarri du, adibidez, komunikatzeko modurik erabiliena telefono deiak izatetik, berehalako mezularitza izatera pasatu da. Horretaz gain, gailu txiki bezain boteretsu bat beti poltsikoan eramateak erabiltzaileari aukera berri asko sortzen dizkio bestelako gailuren baten beharrik gabe, esate baterako, argazkiak atera, Interneten ibili etab.

Bestetik, hizkuntz gutxitu baten hiztun gisa ohikoa da eguneroko bizitzako beste hainbat arlotan bezala teknologiaren arloan gure hizkuntza erabiltzeko zailtasunak izatea. Horren adibide izan daiteke gustokoen ditugun mugikorrerako aplikazio batzuk euskaraz erabili ahal ez izatea. Teknologiaren alorrean, normalean, eleaniztasuna erabiltzaileen esfortzu autonomotik dator, zerbitzuen egileei eskaerak zuzenduz edota gizabanako baten edota erabiltzaile talde baten ekimenez itzulpen bat prestatuz. Softwarearen garatzai-

le izanik garrantzitsua da lan horren kontzientzia hartu eta guk garatutako produktuetan eleaniztasuna filosofiatzat hartzea erabiltzaileen esfortzuak beste ildo batzuetara fokatu daitezen: produktuaren probak, iradokizunak...

Bestalde, aipatutako gailuek egunerokoan hainbesteko presentzia izan arren, zenbait esparrutan ez zaio behar bezainbesteko zukurik ateratzen. Esaterako, museo gehienek oraindik ere bisitariarekiko elkarrekintza “tradizionala” edukitzen dute eta salbuespenak egon arren, horietan ere ez du eragin itzelik. Museoetan egoten den elkarrekintza garatze-ko ez dago bide bakarrik. Batetik, gailu mugikor adimentsuen arlo soziala bultza liteke, sare sozialetan erakusketan ikusitako gauzak zuzenean baloratzeko. Bestetik, erakusketa-ko edukiei buruzko azalpen osagarriak ere eskaini litezke, beste hainbat gauzaren artean.

Proiektu honen ideia aipatutako bigarren garapen-bidea jorratu nahi izatetik eratorri zen. Izan ere, museo eta bestelako monumentuak turistentzat bideratuta daude, oro har, inguruko zein atzerriko. Askotan gertatu izan zait museo batera joan eta erakusten ziren elementuei buruzko azalpenak nik ezagutzen ez nuen hizkuntza batean egotea. Horren ondorioz, museo bisitatzen duzu, baina testuingururik gabe, ez baitakizkizu aurrean duzun artelanaren nondik-norakoak. Horrek museotik alde egitean izaten duzun sentsazioari eragiten dio, museoaren gaiari buruz oso interesatuta egon arren jakintza berri gutxi barneratu duzu eta. Sarritan, museoaren edukien azalpenak hizkuntza ezberdinetan egote eza erabaki ekonomiko bat izan daiteke (itzultzaile bat kontratatzeak gastu handia ekar dezake), baina baita estetiko-praktikoa ere (ez da batere eroso artelan bakoitzaren ondoan hizkuntz ezberdin ugarritan azalpenak jartzea, toki asko behar du eta).

Hortaz, hau da proiektu honen atzean dagoen helburu nagusia: toki mugatua dagoen lekuetan edukia modu eleanitzean eskaintzeko ahalmena duen sistema bat garatzea. Horretarako, elkarrekintza dinamizatuko duen osagai bat behar da, gure kasuan QR kodea. QR kodeak kamara duen edozein gailu mugikor adimentsuren bidez irakur daitezke eta azken aldi honetan oso erabiliak izaten ari dira. Hala ere, orain arte aipatutakoarekin ez litzateke eduki eleanitzak ez eskaintzeko egoten den arazo ekonomikoa konpontze-ko ezer egingo. Horren aurrean, Interneten dagoenari begiratzea besterik ez dugu. Izan ere, Informatika arloko konpainia handiek (Google eta Microsoften kasua) zein beste talde independente batzuek eduki eleanitza bat-batean sortzeko aukera ematen dute. Hori aprobetxatuta, hizkuntza ezberdinak eremu txikiagoan eskaini ahal izateaz gain, edukia hizkuntza ezberdinetan eskaintzeko aukera izango dugu.

Aurreko paragrafoan esandakoarekin proiektuaren izenburuaren zergatia azaldu dugu jada: QR kodeak eta eduki-kudeaketa eleanitza.

1.2 Helburuak

Hauek dira proiektuan zehar garatuko den sistemak dituen helburu nagusiak:

- Toki mugatua dagoen lekuetan edukia modu eleanitzean eskaintzeko aukera ematea.
- Sarean dagoen zerbitzuren bat erabilia eduki eleanitza bat-batean sortzeko aukera izatea.
- Ahal den heinean, kode irekikoak diren zerbitzuak/tresnak erabilia garatzea.

1.3 Proiektuaren irismena

Azpiatal honetan proiektuaren irismenari buruz hitz egingo dugu. Gehiago zehaztearren, aipatuko diren atalak proiektuaren Betekizunak, Lanaren Deskonposaketa Egitura, Emangarriak eta Mugarriak izango dira.

1.3.1 Betekizunak

Proiektu honek kontutan hartu beharreko hainbat betekizun ditu. Proiektuaren betekizunak bi zati nagusitan banatu ditzakegu, *Android* aplikazioak dauzkanak eta Eduki Kudeatzaileak dituenak. Aplikazioa, argi dago, *Android* sistema eragilean funtzionatzeko gai izan behar da. Horretaz gain, QR kodeak irakurtzeko gaitasuna izan behar du. Irakurritako hori baliatuta Eduki Kudeatzailearekiko harremana gauzatu beharko du. Gainera, erabiltzaileari lana erraztearren aplikazioak irakurri diren kodeen historia gordetzeko gaitasuna izan beharko luke. Horrela, erabiltzaileak ez du zertan dagoeneko irakurri duen QR kode bat berriro irakurri, historiatik atzitzearekin nahikoa luke eta.

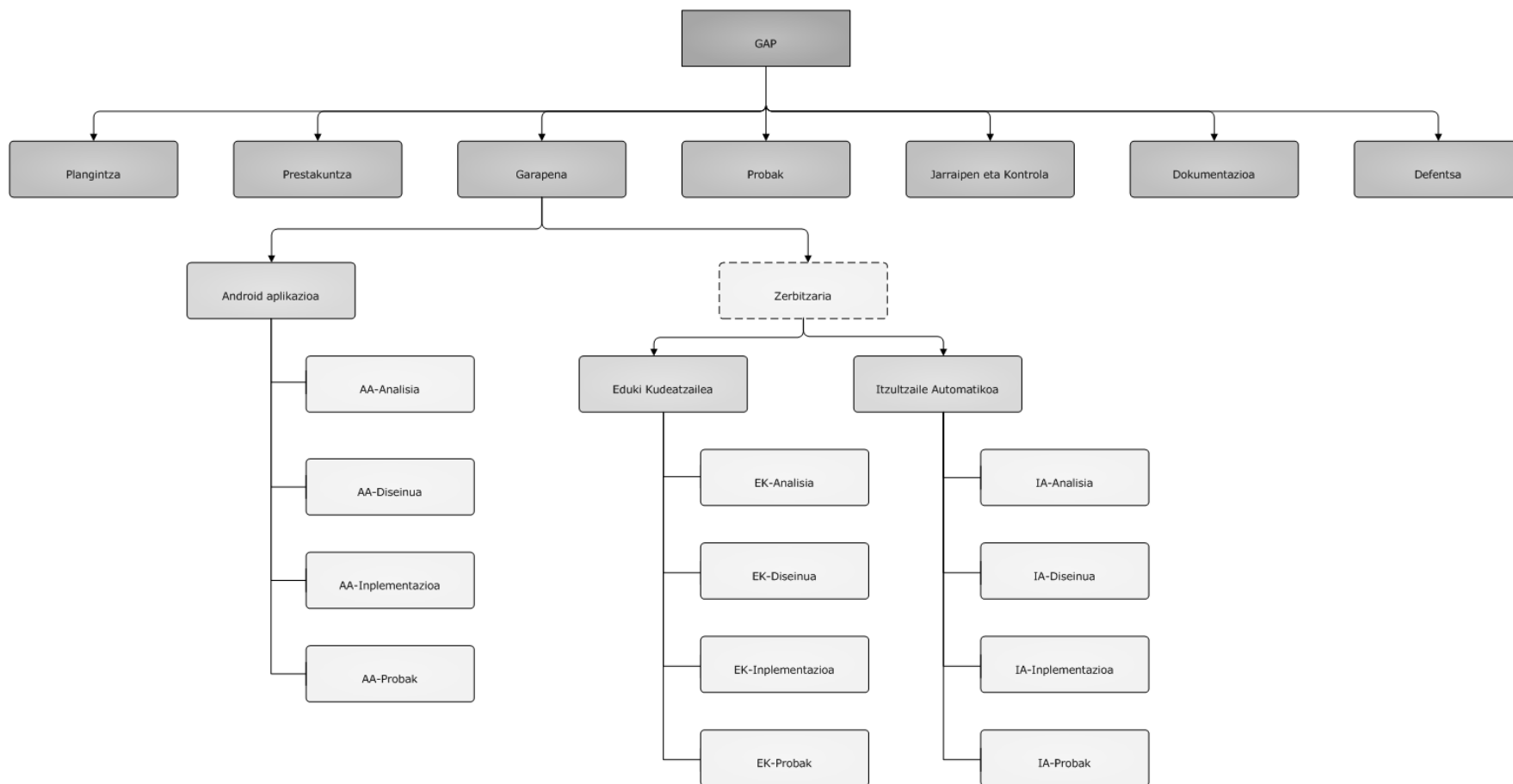
Eduki Kudeatzaileari dagokionean betekizun nagusia informazioa gordetzeko eta erakusteko azpiegitura izatea da. Webgune horrek hainbat hizkuntza izan beharko ditu, eta ziurrenik hizkuntza guztiek ez dute informazio guztia izango. Hor sartuko da berehalako itzulpenen erabilera. Hizkuntza jakin batean informazioa eskura ez dagoenean, Eduki Kudeatzailea gai izan beharko da eduki hori behar den hizkuntzan sortu eta gordetzeko, itzultzaile automatikoki bat erabiliz. Horretaz gain, Eduki Kudeatzaileak atal bat eduki beharko du informazioa gehitzeko, bai zerotik hasitako informazioa bai beste hizkuntza

batzuetan dagoen informazioa hizkuntza berri batera itzulita dagoenean. Informazioa sortu dugunean, informazio horri dagokion QR kodea sortzeko bidea ere eman beharko du, eduki berri horren esteka erabilita QR kodea modu itxuroso batean aurkeztuz.

Aipaturikoaz gain, aplikazioa zein Eduki Kudeatzailea hizkuntza ezberdinen hiztunenentzat bideratuta dagoenez biek hizkuntza ezberdin ugaritan egon beharko dute, erabiltzaileei erabilera errazteko.

1.3.2 Lanaren Deskonposaketa Egitura

Hurrengo orrialdean proiektuaren Lanaren Deskonposaketa Egitura topa daiteke.



1.1 Irudia: Proiektuaren LDEaren irudia.

Aurreko orrialdean erakutsi den LDE-aren atazen deskribapena irakurgai dago jarraian:

- Plangintza: Proiektua arrakastaz aurrera eramateko egin beharrekoaren analisi bat egingo da, eta lanaren garapena antolatu. Egingo diren ekintzen artean, arriskuak identifikatu, betekizunak zehaztu eta horietan oinarritutako kalitate irizpide batzuk ezarri, eta lana denboran zehar antolatu.
- Prestakuntza: Proiektua inplementatzen hasi aurretik beharrezkoak izango ditugun hainbat kontzeptu eta abilezia landu beharko dira. Batetik, landu beharreko kontzeptuak mugikorrerako aplikazioarekin daude loturik *Android* Sistema Eragilerako aplikazioak programatzen ikasi beharagatik. Bestetik, Eduki Kudeatzaileak ere ikasketa eskatuko du, *Wordpress*-en gaitasunak ezagutzeko eta itzultzailea inplementatu ahal izateko.
- Garapena: Atal hau hainbat ataza ezberdinetan banatuko da, bakoitza produktuaren esparru zehatz batean zentratzen delarik.
 - *Android* aplikazioa: aplikazioaren garapen prozesua gauzatuko da hemen eta modu honetan banatuko da:
 - * Analisia
 - * Diseinua
 - * Inplementazioa
 - * Probak
 - Zerbitzaria: Ez da berezko ataza bat. Zerbitzarian kokatuta egongo diren arlo ezberdinez arduratuko diren atazak antolatzeko balio du.
 - * Eduki Kudeatzailea: webgunearen bizkarrezurraren garatze prozesua egingo da hemen eta hauek dira atalak:
 - Analisia
 - Diseinua
 - Inplementazioa
 - Probak
 - * Itzultzaile automatikoa: Eduki Kudeatzaileak erabiliko duen itzulpen automatikoak egiteko modulua garatuko da hemen, eta hauek dira atalak:
 - Analisia

- Diseinua
 - Implementazioa
 - Probak
- Probak: Behin produktuaren atal ezberdinak garatuta daudela, ziurtatu beharko da, banaka funtzionatzeaz gain batera ere portaera egokia dutela.
 - Jarraipen eta Kontrola: Proiektua garatzen dagoen bitartean egingo den ataza honek ezarritako helburuak eta epeak lortutakoekin alderatuko ditu, plangintzaren betetze maila ikuskatzeko eta arazorik egotekotan berandu izan aurretik horiek zuzentzeko.
 - Dokumentazioa: Ataza honetan proiektuaren memoria idatziko da.
 - Defentsa: Ataza honetan proiektuaren defentsa burutzeko prestakuntza egingo da.

1.3.3 Emangarriak

Proiektu honek hainbat emangarri ezberdin ditu, eta horiek bi talde nagusitan bana daitezke. Batetik, produktuarekin modu zuzenean lotuta dauden emangarriak, eta bestetik, proiektuarekin eta honen dokumentazioarekin lotuagoak daudenak.

- Produktuarekin lotutakoak:
 - *Android* aplikazioaren prototipoa.
 - Eduki Kudeatzailearen prototipoa (Itzultzaile automatikoa integratuta izango du).
 - Produktuari buruzko eskuliburua: funtzionamendua azaldu eta erabiltzaileen-tzat laguntza gisa pentsatuta dagoena.
- Proiektuarekin lotutakoak:
 - Proiektuaren helburu dokumentua: proiektuaren hasieran gauzatutako plangintza, proiektuaren dokumentazioaren barnean egongo dena.
 - Memoria: proiektuaren inguruko zenbait informazio ezberdin biltzen dituen dokumentua. Barruan edukiko dituen atalen artean, proiektuaren helburu dokumentua, garapenean zehar biltzen joan den informazioa eta garapena bukatzean landutako informazioa bilduko ditu.
 - Aurkezpena: Proiektuaren defentsan erabiliko den aurkezpen dokumentua.

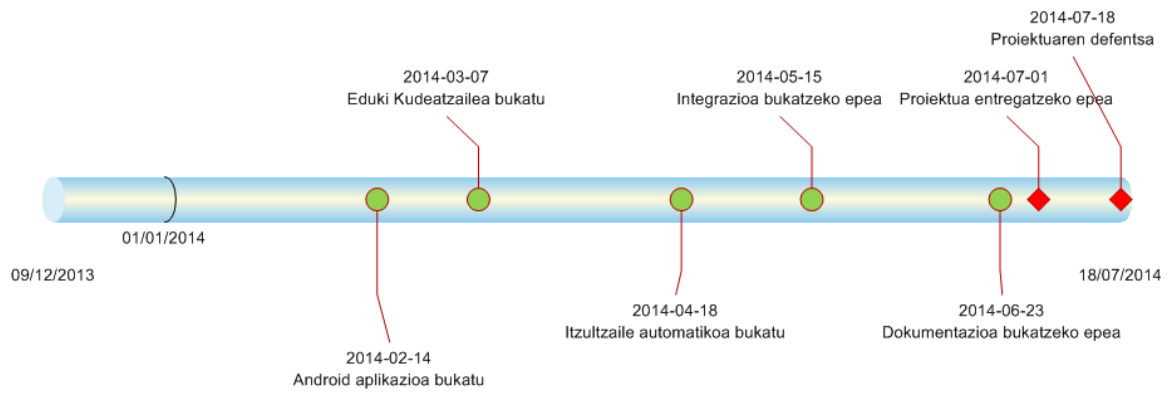
1.3.4 Mugarriak

Proiektuaren garatze prozesuan badaude ezarritako hainbat epe, bete behar direnak, proiektua entregatzeko egunaren kasuan bezala. Horretaz gain, garapenaren egoera kontrolpean izateko, hainbat barne mugarri ere ezarri dira.

- Barne mugarriak:
 - *Android* aplikazioa bukatu (2014/02/14): aplikazioa bukatuta egon beharko da egun honetarako.
 - Eduki kudeatzailea bukatu (2014/03/07): webgunea bukatuta egon beharko da egun honetarako.
 - Itzultzaile automatikoa bukatu (2014/04/18): itzulpenez arduratzen den modula egun honetarako bukatuta egon behar da.
 - Integrazioa bukatzeko epea (2014/05/15): Egun honetarako garapena bukatuta egon behar da eta bi atalek (aplikazioa eta zerbitzaria) funtzionamendu egokia izan behar dute biak batera erabiltzean.
 - Dokumentazioa bukatzeko epea (2014/06/23): egun honetarako dokumentazioa bukatuta egon beharko da, eta aurkezpena prestatzen hasteko unea izango da.
- Kanpo mugarriak:
 - Proiektua entregatzeko epea (2014/07/01): Egun honetarako dokumentazioa eta produktua bukatuta eta entregatuta egon behar dira.
 - Aurkezpena egiteko egunak (2014/07/16-2014/07/18): Egun hauetan proiektuaren defentsa izango da.

Aipatu berri ditugun bi mugarri multzoen artean ezberdintasun nagusi bat dago, kanpo mugarriak ez dira guk ezarriak, eta beraz, bete beharreko epe batzuk dira. Barne mugarriak, berriz, guk ezarri ditugu, hortaz, epeak betetzen saiatu arren lortuko ez balitz ez litzateke ezer gertatuko.

Jarraian daude lehen aipaturiko mugarriak, grafikoki adieraziak.



1.2 Irudia: Proiektuaren mugarriak denboran zehar islatzen dituen irudia.

2. KAPITULUA

Aurrekariak eta Erabilitako Teknologia

Atal honetan proiektu honen aurrekariak aipatzen dira, gure produktua garatzeko baloratu ziren zenbait teknologiaz gain, bukaeran hautatu ziren teknologiak hautatzearen zergatia ere azalduz. Azkenik, erabili ziren teknologien deskribapena ere egiten da.

2.1 Aurrekariak

Produktuaren garapena hasi aurretik antzeko ideia nagusia zuten beste aplikazio batzuk bilatu ziren, horrelako sistema batek bete beharko lituzkeen beharren inguruko ikerketa egiteko. Sarean egindako bilaketan topatu ziren sistemetatik lotura gehien dutenak azaltzen dira jarraian.

2.1.1 *QRpedia* [1]

QRpedia izena QR kodeen eta Wikipediaren arteko loturatik dator. Izan ere, *QRpediak* hori egiten du, QR kodeen bidez hizkuntz ezberdinetako Wikipedia artikulak atzitu. *QRpediaren* funtzionamendua horrela laburtuko litzateke:

1. Daukan QR kode sortaile bereziaren bidez Wikipediako edozein artikulurentzako QR kode bat sortuko da. Kode horrek ez du Wikipediako artikularen helbidea gordeko, baizik eta *QRpediak* erabiltzen duen webgune baterako helbidea. Luzapen bezala Wikipediako artikularen izenburua izango du.

Adibidez, eu.wikipedia.org/wiki/Donostiako_Informatika_Fakultatea helbidea QR sortzailetik pasa ondoren eu.qrwp.org/Donostiako_Informatika_Fakultatea helbidea izatera pasako da.

2. Edozein QR irakurlerekin kode hori irakurtzean *QRpedia*ren webgunera joango da. Bertan, automatikoki mugikorraren hizkuntza detektatu ondoren (arakatzailean ezarrita dagoen hizkuntza hartuz), mugikorraren hizkuntzan dagoen Wikipediako artikulua horretara birbideratuko du erabiltzailea. Wikipediako artikulua erabiltzaileak nahi duen hizkuntzan ez badago, artikulua hori eskuragarri dagoen hizkuntzen zerrenda erakutsiko du.

QRpedia eta proiektu honetan garatutako produktuaren arteko ezberdintasun nagusia hizkuntzaren trataera egiteko kokalekua da. Izan ere, *QRpedia* ez du QR irakurle berezi bat eskatzen kodea irakurtzeko, arakatzailea ezarrita dagoen hizkuntza hartzen baitu kontutuan edukia hizkuntza horretan irekitzeko. Ikuspegi honen bidez edukia zure hizkuntzan irekitzeko derrigorrezkoa da mugikorra hizkuntza horretan egotea.

Proiektu honetan hartutako ikuspegiarekin, ordea, ireki beharreko hizkuntza erabilerara konkritu honetarako egindako aplikazioan erabakitzen da. Honen bidez malgutasuna handiagoa da ireki nahi duzun hizkuntza aukeratzeko. Era berean, aplikazioa gailuan izatea derrigorrezkoa ez den arren (irakurle estandar batekin irakurtzean Eduki Kudeatzailearen hizkuntz lehenetsian irekiko du edukia) erabiltzaileak irekiko den hizkuntza aukeratu ahal izateko aplikazioa bere gailuan eduki beharko du. Kasu horretan, biltegiratze erabilgarri urria duten gailuetan oztopo bat izan daiteke beste aplikazio bat instalatu behar izatea.

2.1.2 Word Lens [2]

Word Lens aplikazioak beste ildo batetik jorratzen du eleaniztasuna, QR kodeak erabili gabe. Aplikazio honek errealitate areagotuaren bidez gailu mugikorraren kamerak hartutako irudia itzultzen du jatorrizko hizkuntzatik aplikazioan ezarritako hizkuntzara. Prozesu horren emaitza kamerak hartutako irudi berdina izango da, baina testua jatorrizko hizkuntzan egon ordez, itzulpenaren hizkuntzan egongo da. Itzulpen hori egiteko aplikazioak ez du Internet konexio baten beharrik, horren ordez mugikorrean instalatuta egon behar diren hizkuntz-paketeak erabiltzen ditu eta. Orain arte eskuragarri dauden hizkuntz-paketeak ingelera eta gaztelania, frantziara, portugalerara, errusiera, italiara edota alemanieraren arteko itzulpenak egiteko gai dira.

Itzulpenak *on-the-fly* edo momentuan egiten dituenek itzulpenen zehaztasuna ez da %100ekoa. Hala ere, aplikazioaren helburua erabiltzaileari aurrean duen testuaren ideia-nagusia ematea denez zuzentasun maila egokia dauka. Zuzentasun hori ahalik eta maila gorenean mantentzeko irakurtzen diren testuak ez dira eskuz idatziak izan behar eta inprimatuak izan diren testuen kasuan estilizatu gabe egon behar dira.

Word Lens-en eta proiektu honetan garatutako produktuaren ikuspegiaren arteko ezberdintasuna argia da. Batek errealitate areagotua erabiltzen duen bitartean edukiak Internet konexiorik gabe hizkuntz ezberdinetan eskaintzeko besteak Interneten kokatutako baliabideak atzitzen ditu. Ikuspegi horrekin produktuaren konplexutasuna jaitsi egiten da baina era berean errazagoa da hizkuntz edukia ezberdin askotan eskaintzea, sarean inplementatuta dauden itzultzaileak erabiltzearekin nahikoa da eta.

2.2 Teknologia posibleak

Proiektuaren garapenean zehar zenbait beharrei aurre egiterako orduan, proiektuan bertan inplementatu behar ez ziren funtzionalitate batzuk lortzeko garapen taldetik kanpo zeuden erakunde batzuen lana aprobeixatu da. Kasu batzuetan behar berdina ase zezaketen hainbat produktu ezberdin topatu ziren. Beste batzuetan, berriz, eztabaida gutxi sortu zuten erabakiak izan ziren. Jarraian erabaki horiek hartzeko zergatia eta planteatu ziren hautabide batzuk aipatuko dira.

2.2.1 Aplikazioa

Sistema Eragilea

Erabaki hau hartzerako orduan ez zen zalantza askorik egon, tutore zein ikaslearen preferentziak kontutan izanda. Lehenik eta behin, *Android* Sistema Eragilea duten gailuak gailu mugikor adimentsuen salmentaren gehiengo zabala osatzen dute, %79-a [3] hain zuzen ere. Garatutako aplikazioa orotariko jendearentzat pentsatuta dagoenez, *Android* SErako aplikazio bat egitea da apusturik seguruenak, gizartearen proportzio handiago bantentzako erabilgarria izango dela ziurtatzen delako. Bestetik, ikasleak Sistema Eragile hori darabilen mugikor bat edukita, *Android* aukeratzea erabaki logiko bat da, probak egitea errazagoa izateaz gain etorkizunean aplikazioa erabiltzeko aukera eskuragarriago egongo

delako. Azkenik, proiektua kode irekikoak diren tresnekin garatu nahi izanak, gainontze-ko plataformak (Blackberry OS eta iOS-ren kasua) baztertzen ditu.

Garapen Ingurune Integratua

Android Sistema Eragilerako programatzeko programatzeko oso lagungarria izaten da Garapen Ingurune Integratu edo *IDE* bat erabiltzea. *IDE*ak programazioan laguntzen duten editoreak dira eta gehienek lengoaiarako espezifikoa den editore bat izateaz gain araztailea eta kodearen konpilatzailea ere izaten dute, beste hainbat aukeren artean. *Android* plataformako aplikazioak garatzeko erabili daitezkeen bi editore aipatzen ditugu jarraian:

- *Android Studio* [7]: *Android* plataformarako espezifikoki pentsatuta dagoen *IDE*a. Hauek dira dituen gaitasun batzuk: Java lengoaiarentzako kode moldaketa eta zuzentze azkarra, interfazea ordenagailuaren saguarekin eraiki daiteke nahi dituzun elementuak nahi duzun kokalekuan jarritz. . . Garapen prozesuan dagoen Garapen Ingurune Integratua da, atzigarri dago baina funtzionalitate batzuk oraindik ez daude erabilgarri edota akatsak dituzte.
- *Android SDK* [8]+*Eclipse* [9]+*ADT* gehigarria [10]: 3 osagaien konbinaketaren birtartez *Eclipse IDE*ari *Android*-erako lagungarriak diren funtzioak gehitzen zaizkio. *Eclipse*-k kodeketa egiteko azpiegitura (editorea, proiektuen antolaketa. . .) eskaintzen du eta *Android SDK* eta *ADT* gehigarriak API liburutegiak eta aplikazioak probatzeko laguntzak eskaintzen dituzte.

Hautatutakoa: *Android SDK+Eclipse+ADT* gehigarria *Android* plataformarako bereziki garatutako *IDE* bat erabiltzeak beste *IDE* “orokorren” aurrean abantailak dituen arren, eraginkortasuna adibidez, garatze prozesuan dagoen ingurune bat erabiltzeak lagundu baino traba gehiago egin dezake. Horregatik erabaki zen *Eclipse* Garapen Ingurune Integratua erabiltzea, beste bi osagaiekin batera.

QR kodeen irakurlea

Behin aplikazioarentzako Sistema Eragilea aukeratuta zegoenean, aplikazioak QR kodeak irakurtzeko gaitasuna lortzeko moduren bat bilatu beharra zegoen. Hauek izan ziren baloratutako aukerak:

- *ZBar bar code reader* [14]: Kode irekikoa den QR kodeak irakurtzeko liburutegia. Sistema Eragile ezberdinetarako bertsioak ditu, *Android* tartean. Memoria gutxi behar duen liburutegia.
- *Zxing* [15]: Kode irekikoa den QR irakurlea. Ez du liburutegirik eskuragarri, zuek bat osatu ezean. Memoria gutxi behar duen irakurlea, funtzionalitate ugariarekin. Liburutegirik ez edukitzeak mugikorraren erabiltzailea *OharEleanitzak* aplikazioa instalatzeaz gain, *Zxing* aplikazioa ere instalatzera behartuko luke.

Hautatutakoa: *ZBar bar code reader* Aurreko deskribapenak ikusita, argi ikusten da *ZBar bar code reader* bezalako liburutegi bat erabiltzeak dituen abantailak. Gainera, ondo aurreikusi zen bezala, liburutegiaren bidez QR kodeak aplikazioaren barnean irakurtzea abantaila handia izango zen, irakurritakoa prozesatu egin behar baita arakatzailarekin ireki aurretik.

2.2.2 Eduki Kudeatzailea

Hosting zerbitzua

Proiektuaren parte den Eduki Kudeatzailea sarean atzigarri egon zedin beharrezkoa zen *hosting* zerbitzu bat erabiltzea. Zerbitzu horrek doakoa izan behar zuen. Interneten doakoak diren *hosting* zerbitzu ugari topa daitezke baina zoritxarrez, zerbitzu horien kalitatebermeak urriak dira eta duzun askatasuna ere bai. Doako zerbitzu gehienek ezaugarri berberak zituzten onartzen ziren datu-base kopuruan, zuretzako erabilgarri dagoen memorian... Hala ere, hauek ziren besteen gaineratik zeuden bi zerbitzuak:

- *000WebHost* [18]: Zerbitzu honek domeinuaren izena aukeratzeko askatasuna ematen dio erabiltzaileari, helbideari luzapen bat (*.host56*) gehitzen dion arren. Eduki Kudeatzailea irekitzerakoan pop-up gisako iragarkiak. Iraungitze datarik gabeko doako zerbitzua.
- *Hostinger* [19]: Domeinuaren izena aukeratzeko askatasuna, helbideari luzapen bat (*.esy*) gehitzen dion arren. Iraungitze datarik gabeko doako zerbitzua.

Hautatutakoa: *Hostinger* Azkenean *Hostinger* aukeratu zen batez ere iragarkiak ez zituelako, bestela erabiltzaileen (batez ere mugikorrarekin nabigatzean) erabilera esperien-

tzia asko kaltetuko litzatekeelako. Horretaz gain, zerbitzu honek luzapen bat ere gehitzen duen arren, luzapen hori “irakurgarriagoa” da.

Edukiak Kudeatzeko Sistema [20]

Edukiak Kudeatzeko Sistema bat aukeratzeko orduan oinarrizko ezaugarriak kontutan hartzeaz gain (atzean daukan komunitatea, pertsonalizaziorako aukerak, gehigarriak inplementatzeko laguntza...) hizkuntz ezberdinetan dagoen Eduki Kudeatzailea egiteko dagoen erraztasuna ere kontutan hartu da.

- *Drupal*: Garatzaileentzako lagungarria. Ikasketa prozesu luzea *Drupal* jariatortasunez erabili aurretik. Itxura aldatzeko aukera gutxi eta dauzkan gehigarri gehienengatik ordaindu egin behar da.
- *Joomla*: Gehigarri kopuru handia (horietako asko doakoak) eta atzean dagoen komunitate jendetsua. Erabiltzeko zailtasun handirik ez duen arren jariatortasunez erabili ahal izateko pasatu beharreko ikasketa prozesua dago.
- *WordPress*: Webgune kopuru handiena duen EKS-a. Eraldatzeko erraza eta webgune estilo ezberdinak ezartzeko malgutasuna eskaintzen du. Gehigarri katalogo handia (horietako asko doakoak) eta gehigarrien garapenean laguntzeko komunitate nabarmena. Hizkuntz ezberdin askotarako itzulpenak eskuragarri. Bestalde, oso maiz ateratzen dira eguneraketa berriak, eta ondorioz baliteke gehigarri zaharrek funtzionatzeari uztea.

Hautatutakoa: *WordPress* Proiektuaren garapena hasi aurretik *WordPress*-ekin edukitako esperientzia positiboa eta *WordPress* hizkuntz ezberdin askotan eskuragarri egotea pisuzko arrazoiak izan ziren erabakia hartzerako orduan. Horretaz gain, eskaintzen dituen pertsonalizazio aukerak abantaila esanguratsua dira.

Aipatzekoa da doako *hosting* zerbitzu gehienek, *Hostinger*-ek barne, webguneak eraikitzeko plataforma ezberdinen instalazio automatikoa burutzeko aukera ematen dutela, erabiltzaileari lana erraztuz. Gure kasuan, *WordPress* automatikoki instalatzeko aukera eman arren eskuz egin zen *File Transfer Protocol* edo FTP protokoloa erabilita. Horretarako arrazoi nagusia zuzen-zuzenean euskarazko bertsioa instalatu ahal izatea izan zen. Denbora aurrera joan ahala, ordea, beharrezkoa izan zen eskuz instalatzea *hosting* zerbitzuak *WordPress*-en azken bertsioa soilik instalatzeko aukera ematen zuelako. Izan ere,

qTranslate WordPress-en azken bertsioarekin duen bateragarritasun eza dela eta derri-gorrezkoa izan zen eskuz instalatzea, Eduki Kudeatzaileak funtziona zezan.

Internazionalizazioa [24]

Erabiliko zen Edukiak Kudeatzeko Sistema aukeratu ostean Eduki Kudeatzailearen internazionalizazioa nola gauzatu erabaki beharra zegoen. Internazionalizazioa Eduki Kudeatzailea hizkuntz ezberdinetan jartzeko egin beharrekoa da. *WordPress*-ek ez du era lehenetsian webgune batean hizkuntz ezberdinak ezartzeko aukerarik ematen. Hori lortzeko Eduki Kudeatzailean gehigarri bat instalatu beharra dago. Eduki Kudeatzailea hizkuntza ezberdinetan jartzeko hainbat gehigarri mota daude:

- Bidalketa berdinean hizkuntza ezberdinetako edukia izatea ahalbidetzen dutenak (*qTranslate*): Gehigarri mota honek Bidalketa berdinean hizkuntz ezberdinetako edukia idaztea ahalbidetzen du. Gero, arakatzailerekin hizkuntz ezberdinetako edukia atzitzeko nahikoa da hizkuntzaren bi letra-kodea URL-ean jartzea. Esate baterako, webgunebat.eus/en/sarrera sarreraren ingelesezko bertsioa lortzeko.
- Eleaniztasuna Bidalketa ezberdinak elkarri lotuz lortzen dutenak (*WPML, Polylang...*): Gehigarri mota honek eduki berdinen hizkuntz ezberdinetako bertsioak lortzeko hizkuntz bakoitzerako Bidalketa bat sortzea eskatzen dute. Ondoren, Bidalketa horiek elkarrekin lotu daitezke eduki berdinen bertsio ezberdinak direla adierazteko.
- Eleaniztasuna hizkuntza batean dauden webgune ugari lotuz lortzen dutenak (*Multisite Language Switcher, Multilingual Press...*): Gehigarri mota honek hizkuntz bakoitzeko webgune bat sortzea eskatzen du. Gero, webgune ezberdinetan dauden Bidalketa berdinen bertsio ezberdinak lotzen dira.

Hautatutakoa: *qTranslate* Erosotasuna dela eta, aipatu den azken gehigarri mota azkar baztertu zen hainbat webgune kudeatzeak dakartzan lan biderkatuak direla eta. Beste bi moten artean erabakitzerako orduan *Android* aplikazioak zituen behar teknikoak kontutan hartu ziren. Izan ere, aplikazioak kode bat irakurtzean modu logiko batean egin behar du helbidearen eraldaketa. *WPML* bezalako gehigarriekin, ordea, kasu bakoitzean ezberdina da erlazioa, adibidez, sarreraren ingelesezko bertsioa *introduction* da baina artearena

art da. Hau da, ez dago modu argirik Bidalketa berdinen bertsio ezberdinek zein helbide duten jakiteko. Datu-base batekin lortu daiteke baina horrek behar baino gehiago konplikatu du egin beharreko lana. Gainera, hizkuntz ezberdinetako edukia Bidalketa berdinean idatzi ahal izateak itzulpenak egingo dituen gehigarriaren lana asko errazten du, testua Bidalketa bakarretik hartu behar baitu, eta ez bat baino gehiagotatik.

Itzultzailea

Eduki Kudeatzailearen edukiaren itzulpenak egiteaz arduratzen den gehigarria inplementatzen hastean ere Interneten eskuragarri zeuden hainbat zerbitzu baloratu ziren:

- *Apertium* [30]: Software librea da, *OpenTrad* proiektutik eratorria. Edonork erabil dezake inolako gakorik eskatu behar izanik gabe. Noranzko bakarreko 79 itzulpenbikote ditu erabilgarri (euskararen kasuan, ingelera eta gaztelaniara), horietako kasu batzuetan itzulpenak bi noranzkoetan egiten dituen arren.
- *Google Translate* [27]: Iraganean doako zerbitzu bat izan arren, gaur egun ordaindu beharreko zerbitzua da, 20\$ milioi bat karaktereko, hain zuzen ere. Itzulpenak 69 hizkuntzen artean egin ditzake, bi noranzkoetan.
- *Microsoft Translator* [28]: Gakoa izanez gero dohainik erabili daitekeen zerbitzua, hilean 2 milioi karaktere baino gutxiago itzuliz gero. 45 hizkuntzen arteko bi noranzkotako itzulpenak egiteko gai da, baina euskara ez dago 45 horien artean. Zerbitzua Microsoften zerbitzuen bidez(.NET esaterako) eraiki diren sistemei orientatuta.

Hautatutakoa: *Apertium* Beste bi zerbitzuen izaera murriztailea dela eta, ekonomikoa batetik eta plataformarekiko dependentzia bestetik, *Apertium* taldearen zerbitzua erabiltzea aukeratu zen.

QR kodeen sortzailea

Eduki Kudeatzailearen kudeatzaileei lana errazteko asmoz inplementatutako gehigarriak QR kodeak sortzeko aukera izatea planteatu zen. Helburu horrekin, Interneten eskuragarri dauden zerbitzuak begiratu ziren:

- *QR Code Generator* [16]: Zerbitzu honen bidez Eduki Kudeatzailerako QR kodeak sor daitezke *HTTP GET* deiak erabiliz. QR kodeak sortzeaz gain, kode horiek pertsonalizatzeko aukera ere ematen du: kolorea, tamaina. . .
- *QRickit* [17]: Zerbitzu honek QR kodeak sortzeko aukera ematen du *HTTP GET* dei batekin. Zerbitzu honek ere QR kodearen diseinuan aldaketak egiteko aukera ematen du: kolorea, tamaina, oin-testua. . .

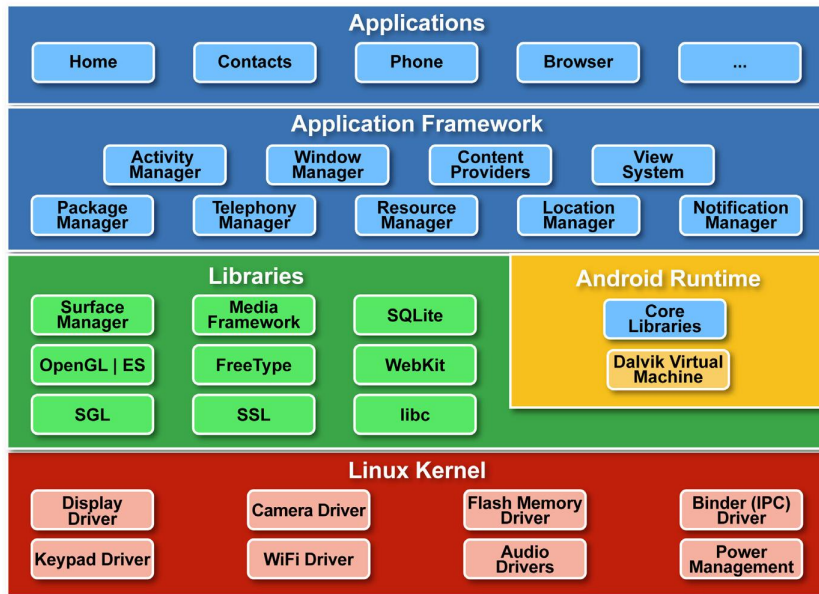
Hautatutakoa: *QRickit* *QR Code Generator*-ek eskaintakoa baina pertsonalizazio aukera gehiago eskaintzen zituenez, gainontzeko espezifikazioa oso antzekoa izanik, *QRickit* aukeratzea erabaki zen.

2.3 Erabilitako teknologia

Atal honetan proiektuan zehar erabili diren teknologien deskribapen zehatzago bat egiten da. Teknologia horiei buruz jakin beharreko kontzeptu batzuk ere azaltzen dira, produktuaren diseinuan azalduko diren hainbat atal ulertzeko komenigarriak izango direnak.

2.3.1 *Android* Sistema Eragilea

Tableta zein telefono mugikor adimentsuetarako garatu den aplikazioa *Android* Sistema Eragilea duten gailuetarako garatu da. *Android* Sistema Eragileak dituen ezaugarri nagusiak honakoak dira [4]: kode irekikoa da (Google konpainiaren jabetzakoa izan arren); *hardware*-arekiko independentea da, hau da, beste SE batzuk ez bezala (Apple-en IOS SEaren kasua, adibidez) ez da gailu konkretu baterako diseinatua izan; aplikazioak Javan garatzen direnez, etorkizuneko edozein PUZ-etan exekutagarriak izango direla ziurtatzen da; zerbitzu ezberdin ugari daude eskuragarri, hala nola, SQL datu baseak, GPS zein sare bidezko kokapena. . . ; segurtasun maila onargarria dauka, Linuxetik eratorria izateagatik eta aplikazioa bakoitzak bere baimenak edukitzeagatik; eta potentzia baxuko zein memoria baxuko gailuentzako optimizatuta dago. Ezaugarri hauek direla eta, *Android* Sistema Eragilea duten mugikor zein tabletek mundu osoan zehar dagoen kopuruaren ia %80a osatzen dute.



2.1 Irudia: *Android*-en arkitektura erakusten duen irudia.

Goiko irudian ikusi daitekeen bezala, *Android*-en arkitektura [5] bost geruzatan banatu daiteke:

- *Linux Kernel*-a: Linux Sistema Eragilearen 2.6 bertsioaz osatua dago. Geruza honek hainbat zerbitzu ematen ditu, esate baterako, segurtasuna, memoriaren kudeaketa, multiprozesua etab. Geruza hau da *hardware*-arekiko menpekota den bakarra eta goiko geruzentzako abstrakzio geruza gisa erabiltzen da.
- *Android* exekuzio unea: Javan erabiltzen den makina birtualaren kontzeptuan oinarritzen da. Memoria eta prozesatze gaitasun eskasa dela eta *Android*-ek bere makina birtual propioa dauka, *Dalvik* izenekoa, murriztapen horiei hobeto erantzuteko asmoz. Horretaz gain, geruza honetan Java lengoia dauden bizkarrezur liburutegi gehienak daude.
- Liburutegi natiboak: *Android*-en zenbait osagaik erabiltzen dituzten hainbat C/C++ liburutegi dauka geruza honek, prozesadorearen kode natiboan konpilatuak izan direnak.
- Aplikazioaren ingurunea: Aplikazioentzako diren berrikuntza ugari dituen plataforma bat eskaintzen du. Horretarako arrazoi nagusia osagaien berrerabilera sinplifikatzea da.

- Aplikazioak: Geruza honetan *Android* gailu batean instalatuta dauden aplikazioak kokatzen dira. Gailuaren segurtasuna babesteko, aplikazio guztiak *Dalvik* makina birtualean exekutatzeko dira.

Android Sistema Eragilearen 19 bertsio ezberdin inguru daude dagoeneko. Atera den azken bertsioa 4.4 edo *Kit-Kat* bertsioa da, 2013ko urrian atera zena (bertsio horren hobekuntza den 4.4.4 bertsioa 2014ko ekainaren 19an atera zen). Hala ere, *Android SE*aren lehen bertsioen erabiltzaileek ez da ia gelditzen. Izan ere, gaur egun, *Android* Sistema Eragilearen erabiltzaileen ia %99a, 10. bertsioa (2.3 *Gingerbread* bertsioa, alegia) edo berriago bat duten erabiltzaileek osatzen dute [3]. Kontutan hartu behar da *Android* aplikazio batek normalean bertsio ugarian funtzionatzeko duela, aldaketarik jasan beharrik gabe. Nahikoa da aplikazioaren manifestuan aplikazioak jasan behar duen bertsio minimoa eta maximoa jartzearekin. Horretarako, ziurtatu beharra dago bertsio horiek aplikazioak erabiltzen dituen osagaiak jasaten dituela.

***Android* jardueren bizitza zikloa [6]**

Android aplikazioak jarduera eta zerbitzu ezberdinez daude osatuta. Jarduera bat, funtsean, aplikazioa erabiltzen ari garela ikusten ditugun interfazeekin elkarrekintza izateaz eta gure ekintzei dagokien erantzuna emateaz arduratzen da. Hortaz, aplikazio baten erabiltzailea jardueraz jarduera joango da, horietako bakoitzak bere funtzio berezia duelarik. Sistema bisitatu ditugun jardueren pila bat osatu eta maneiatzeaz arduratzen da, eta horrela *atzera* botoiari ematean, egon berri garen jarduerara joango da, pilatik elementuak kenduz doan bitartean.

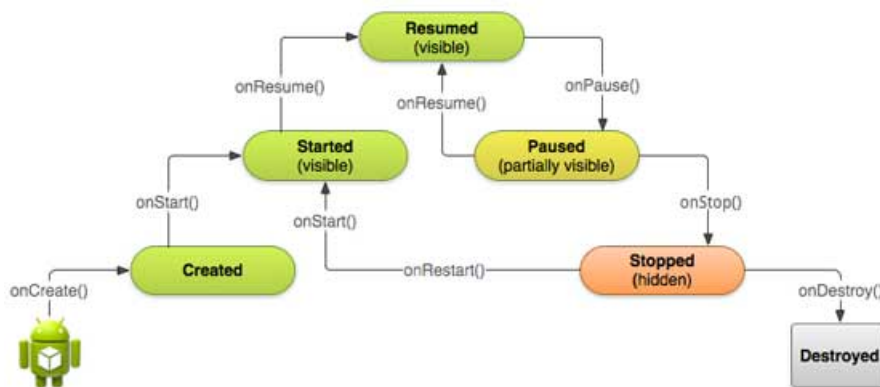
Android aplikazio bakoitzak bere Linux prozesua dauka eta bizirik jarraituko du erabilia izateari utzi eta sistemak deuseztatzea erabakitzen duen arte. Prozesuen suntsitzea martxan dauden jarduera eta zerbitzuen arabera egiten da eta memoria libre gehiago izan nahi duenean suntsitzen du sistemak, erabiltzailearen erabilera esperientzia ona ziurtatzeko. Jarduera baten prozesua itxi ondoren berriro ireki nahi izatekotan aurreko exekuzioko egoera galduko da, aplikazioak hura gorde ezean.

Android jarduera bat honako lau egoeretako batean egon daiteke:

- Exekutatzeko (*Running*): Jarduera pilaren gailurrean dago, beraz, ikusgarri dago eta erabileraren fokua dauka.

- Geratua(*Paused*): jarduera ikusgai dago baina ez dauka fokurik. Egoera honetan beste jarduera bat egoera aktibora pasatu da, baina interfaze garden bat edota pantaila ez-osoko jarduna izateagatik, jarduera ikusgai dago.
- Bukatua(*Stopped*): Jarduera ez dago ikusgai. Ezarpenen eta interfazearen egoera gordetzea komeni da.
- Suntsitua(*Destroyed*): Jardueraren barruan *finish()* metodoari deia egitean edo sistemak prozesua hiltzean gertatzen da, eta prozesuen pilatik kenduko da.

Jarduera bat egoeraz aldatzen den bakoitzean gertaera batzuk sortzen dira, metodo berezi batzuekin harrapatuak izan daitezkeenak, trataera konkretu bat ematearren. Jarraian duzue hori ilustratzen duen irudi bat.



2.2 Irudia: *Android* jarduera baten bizitza-zikloa irudikatzen duen grafikoa.

- *onCreate(Bundle)*: Jardueraren hasieratzea gauzatzen da, esate baterako, interfazearen sorkuntza, gordetako egoeraren berrezarkuntza ...
- *onStart()*: Jarduera erabiltzaileari erakutsi aurreko momentuan exekutatzen da.
- *onResume()*: Jarduera eta erabiltzailearen arteko elkarrekintza hastera doanean exekutatzen da.
- *onPause()*: Jarduera *Geratua* egoerara pasatu behar denean exekutatzen da. Datuak gordetzeko eta kamera gelditzeko une egokia.
- *onStop()*: Jarduera *Bukatua* egoerara pasatuko da. Memoria gutxi badago posiblea da prozesua suntsitua izatea funtzio honi deirik egin gabe.

- *onRestart()*: Jarduera *Bukatua* egoeratik dator eta berriro ikusgai izatera doanean exekutatu da.
- *onDestroy()*: Jarduera *Suntsitua* egoerara pasatu aurretik deitzen da. Memoria gutxi badago baliteke jarduera suntsitzea funtzio honi deitu gabe.

Android aplikazioen garapena

Android aplikazioak garatzeko beharrezkoa da *Android Software Development Kit* edo *Android SDK*-a bezalako tresna bat erabiltzea. Izan ere, horrek *Android*-en garatzeko behar diren liburutegiak eta tresnak jartzen ditu eskuragarri. Hori oinarri hartuta, garapenean laguntzeko *Eclipse* garapen ingurunea erabili da, *ADT gehigarriarekin* batera. Jarraian egiten da osagai horien deskribapena:

- *Android SDK* [8]: Honek garatzaileari API liburutegiak eta garatzaileentzako tresnak eskuragarri jartzen dizkio, *Android* aplikazioak eraiki, probatu eta arazteko lagungarriak direnak.
- *Eclipse* [9]: Garapen Ingurune Integratu edo *IDE* bat da. Lan ingurune bat dauka proiektu ezberdinak lantzeko erabiltzen dena, denak eskuragarri daudelarik. Gehienbat Java programazio lengoaian dago idatzia eta lengoia horretan programatzeko balio du, baina gehigarrien instalazioaren bitartez beste programazio lengoia batzuetan programatzeko ere balio dezake, edota gure kasuan *Android* plataforman programatzeko.
- *ADT* gehigarria [10]: *Android Development Tools* deitutako gehigarria *Eclipse*erako pentsatuta dago, *Android* programaziorako ingurune integratu eta indartsu bat emateko asmoz. Gehigarri honen bidez *Android SDK*-ren gaitasunak *Eclipse* garapen ingurunean integratzen dira.

Aplikazioa garapen prozesuan dagoen bitartean, normala den bezala, aplikazioaren funtzionalitateak probatzea beharrezkoa da. Hori egiteko *Android SDK* osagaiak bi aukera ezberdin ematen ditu. Batetik, garatua izaten ari den aplikazioa mugikorrera pasatu eta mugikorrean bertan probatzeko aukera ematen du. Bestetik, emuladore bat erabiltzeko aukera ere badago. Emuladoreak *Android* Sistema Eragilea daraman gailu bat sortzeko aukera ematen dizu, Sistema Eragilearen bertsioa eta gailu mugikorraren ezaugarri teknikoak (prozesadorearen gaitasuna, pantailaren tamaina. . .) zehazteko aukerarekin. Emuladorean

gailu ezberdinak sartzek *Android* aplikazioa beste *Android* bertsio batzuetan probatzeko aukera ematen dizu, aplikazioa bertsio ezberdin guztietan dabilela ziurtatu ahal izateko.

2.3.2 QR kodeak [12]

Android aplikazioa QR kodeak irakurtzeko gai da. QR edo *Quick Response* kodeak bi dimentsiotako barra-kodeak dira, makinek irakurtzeko modukoak direnak irakurle optiko (kamera bat, adibidez) baten bitartez. QR kode bat lauki bat da, bere barnean itxura karra-tuko puntu beltzak eta zuriuneak dauzkana. Kodea irakurtzeko horizontalki zein bertikalki irakurtzen dira aipaturiko osagaiak.

Hasiera batean autogintzan erabiltzeko sortu bazen ere, bere erabilera asko hedatu da. Guk URL edo *Uniform Resource Locator* direlakoak irakurtzeko duen gaitasuna aprobetxatuko dugu. Horren bitartez, erabiltzailearekiko elkarrekintza asko sinplifikatzen da. Eduki Kudeatzailearen helbidea erabiltzaileak web arakatzailan idatzi behar izan ordez, kodearen gaineratik irakurlea pasatuta Eduki Kudeatzailearen helbidea lortuko da, erabiltzaileari ia esfortzurik gabe Eduki Kudeatzailea atzitzea ahalbidetuz.

Hala eta guztiz ere, QR kodeek badute arrisku bat, mugikorraren segurtasunari eragiten diona. Izan ere, QR kodeak webguneak irakurtzeko erabiltzen direnean, horrek Internetarako konexioa izatea suposatzen du webgunea atzitu ahal izateko. Orokorrean ez dago arriskurik QR kodeez egiten den ohiko erabileran, baina kode hori erabilia izan daiteke mugikor edota tabletan dauden beste aplikazio batzuen ahuldadeez aprobetxatzeko, eta ondorioz, erabiltzaileak nahi ez dituen ekintzak egin. Fenomeno horri *attaging* [13] deitzen zaio eta funtsean, zilegi diren QR kodeen itxura hartzen duten kode maltzurak erabiltzean datza. Kode horiek, erabiltzaileak webgune “zintzo” batean dabilela pentsatzen duen bitartean, aplikazio beraren edo beste aplikazio akastun batzuen baimenak erabiltzen dituzte mezuak bidali, argazkiak atera, datu pertsonalen transmisioa etab egiteko.

ZBar bar code reader liburutegia [14]

ZBar bar code reader izeneko liburutegia erabilia, *Android* aplikazioa QR kodeak irakurtzeko gai da. Liburutegia mota ezberdinetako barra kodeak irakurtzeko gai da baina aplikazioak QR kode bat irakurtzen denerako portaera soilik du definituta, kode mota hori aukeratu baita erabiltzailea Eduki Kudeatzailearen edukietara bideratzeko. Liburutegiak plataforma mugikor ezberdin askotarako balio du, webgunean Sistema Eragile ezberdinentzako bertsioak dituelarik.

QRickit [17]

QRickit QR kodeak doan sortzeko aukera eskaintzen duen webgune bat da. Hainbat aukera ezberdin daude eskuragarri webgunean baina produktuaren garapenerako erabili dena *QRickit* APIa da. API horrek deian jasotako web-helbidearen QR kodea itzultzen du. Gainera, QR kodea zenbait modutan eraldatzeko aukera ere eskaintzen du, hala nola, tamaina, kolorea, oin- testua etab.

2.3.3 *WordPress* [21]

WordPress doakoa eta kode irekikoa den Edukiak Kudeatzeko Sistema (*Content Management System*, ingeleraz) bat da. PHP programazio lengoia oinarrituta dago eta MySQL datu-base batekin funtzionatzen du. Dauden Eduki Kudeatzaileen artean erabiliena da, horrelako sistema bat erabiltzen duten webguneen %60 *WordPress* erabiliz eginda dagoelarik. Ezaugarri aipagarrienak: txantilo ugari ditu eskuragarri eta gehigarriak sostengatzen dituen arkitektura bat du. Lehenengoak administrariei itxuraz aldatzea ahalbidetzen die, webgunearen edukia edota egitura aldatu beharrik gabe. Txantilo horiek PHP, HTML edo CSS bidez eraldatu daitezke eta baita berriak sortu ere.

Gehigarriak, berriz, *WordPress*-ek berezkoak dituen funtzionalitateak hedatzeko erabiltzen dira, webgune osoagoak egitea erraztuz. Gainera, erabiltzaileek askatasuna dute behar dituzten gehigarriak sortu eta beraien webgunera gehitzeko. Horretaz gain, bilatzaileekiko atsegina (*search-engine friendly* esaten dena) da, artikuluentzako hierarkia sortzeko aukera du eta esteka ulergarriak edota irakurgarriak eskaintzen ditu. *WordPress* euskaraz [22] erabiltzeko aukera dago, Librezale taldearen lanari esker.

Gehigarrien garapena [25]

WordPress-en dokumentazioaren arabera *WordPress*-erako gehigarri bat PHP lengoia idatzitako programa edo funtzio multzo bat da, *WordPress* *weblog*-ari zerbitzu edota aukera jakin batzuk gehitzen dizkionak bestelako kudeaketarik gabe integratu daitezkeenak *WordPress*-erako Gehigarrien API-ak eskaintzen dituen funtzioak eta sarbide-puntuak erabilita.

Definizioan esan den bezala gehigarri bat inplementatzeko PHP lengoia erabiltzen da, horrekin batera beste hainbat erabili daitezkeen arren, adibidez, CSS, JavaSc-

ript. . . PHP bidez garatutako funtzioek eragina izan dezaten funtzio horiek APIak eskaintzen dituen kakoei lotu behar dira. Bi kako mota daude, ekintzak eta iragazkiak. Lehenengo bidez gure funtzioa guk aukeratutako ekintza exekutatzeko denean exekutatu da. Dauden ekintza ezberdinen artean gehigarriak kargatzearena, edukia erakusteara eta beste hainbat topatu ditzakegu. Iragazkiek, berriz, erabiltzaileari aukera ematen diote egoera batzuetan erabiliko den aldagaiari aldaketa bat egiteko sistemak erabili aurretik. Esate baterako, izenburuari aldaketa bat egiteko, Bidalketaren edukiari aldaketa egiteko. . .

Horretaz gain, *WordPress*-erako gehigarriek aukera dute eleaniztasunerako azpiegitura sortzeko. Karaktere-kateak kodean idazteko *WordPress*-ek duen funtzio bat erabiltzen badugu zuzenean karaktere-kate horiek irakurri ordez, webguneak gehigarriaren karaktere-kateak .mo fitxategitik, makinentzat irakurgarria den itzulpen fitxategitik irakurtzea ahalbidetzen dugu. Gehigarria hizkuntz berri batera itzultzeko nahikoa da .mo fitxategiarekin batera doan .po fitxategia hizkuntz berri batera itzultzea.

qTranslate [26]

Bidalketa bakarrean hizkuntz ezberdinetako testua izatea ahalbidetzen duen *WordPress*-erako gehigarria. *WordPress*-en itzulpen fitxategiak izatekotan (.mo fitxategiak, alegia) edozein hizkuntza gehitzea onartzen du. Hizkuntz ezberdinetako edukia atzitzeko webhelbideari eraldaketa bat egiten dio, hori egiteko hiru aukera ezberdin dauzkalarik:

- *Query* edo itaun bidez: Helbidearen bukaeran `?lang='hizkuntza'` itxurako luzapen bat gehituko du. Adibidez, *honi-buruz* Orrialdearen ingelerazko bertsioa lortzeko:
<http://ohareleanitzak.esy.es/honi-buruz/?lang=en>
- Aurre-helbide bezala: Eduki Kudeatzailearen oinarrizko helbidearen eta Bidalketarinaino eramaten zaituen helbidearen zatiaren artean hizkuntza-kodea sartzea. Adibidez, *honi-buruz* Orrialdearen ingelesezko bertsioa lortzeko:
<http://ohareleanitzak.esy.es/en/honi-buruz/>
- Aurre-domeinu gisa: Eduki Kudeatzaileak aurre-domeinu bat dauka hizkuntza bakoitzerako eta horren bidez egiten da hizkuntz aldaketa (zerbitzariaren konfigurazte gehigarria eskatzen du). Adibidez, *honi-buruz* Orrialdearen ingelesezko bertsioa lortzeko:
<http://en.ohareleanitzak.esy.es/honi-buruz/>

Hosting zerbitzuaren mugak direla eta aurre-domeinuaren aukera ezin izan zen hautatu. Helbide *txukunago* bat izateko hizkuntza-aldaketa aurre-helbide gisa egiten da.

Lehenago aipatu den bezala, *WordPress*-ek sarri izaten ditu eguneraketak. Horietako batzuek beste batzuek baino eragin handiago izaten dute funtzionamenduan. Kasu hone-tan, *Wordpress*-en 3.9 bertsioa atera zenean (3.9.1 bertsioarekin ere berdin) *qTranslate*-k behar bezala funtzionatzeari utzi zion. Garapen fasea ia bukatuta zegoenez, *qTranslate* *Wordpress*-en bertsio berriarekin bateragarria izan arte eguneraketa ez egitea erabaki zen, horren ordez gehigarriarekin bateragarria den *WordPress* 3.8.3 bertsioa erabiliz.

2.3.4 *Apertium* [30]

Apertium, erregeletan oinarritutako itzulpen automatikoak egiten dituen plataforma bat da. Software librea da, beste hainbat itzultzaile automatiko ez bezala. *Apertium Open-Trad* proiektutik abiatu zen eta hasiera batean antzekotasunak zeuzkaten hizkuntzen artean itzulpenak egiteko diseinatu zen. Denborarekin, ordea, hain antzekoak ez ziren hizkuntzen arteko itzulpenak egiteko hobetu zen. Horren ondorioz, gaur egun, noranzko bakarreko 79 hizkuntz bikoteren arteko itzulpenak [32] egin ditzake. Horietako batzuetan, gaztelania eta katalanaren kasuan adibidez, bi noranzkotan egin dezake itzulpena, hau da, gaztelanietatik katalanera eta katalanetik gaztelaniara. Beste kasu batzuetan, ordea, noranzko bakarrean egiteko gai da, euskara eta ingelesaren kasuan bezala.

Apertium software librea eta hein handi batean bolondres edota esparru akademi-koan lan egiten duten pertsonen esfortzuen ondorioa izanik, ez dauka inbertsio ekonomiko erraldoirik bere garapenaren atzean, beste itzultzaile batzuekin gerta daitekeen kasuan, *Google Translate*-ekin, adibidez. Hori dela eta, sortzen dituen itzulpenak ez dira bikainak eta hizkuntz-bikote ezberdinen emaitzen eraginkortasunaren arteko ezberdintasunak handiak dira. Jarraian erakutsiko da hizkuntz-bikote batzuen eraginkortasunari buruzko informazioa. Horretarako WER edo *Word Error Rate* izeneko ehunekoa erabiliko da, itzulpena egin ondoren 100 hitz bakoitzeko zuzendu beharreko hitz kopurua neurtzen duena. *Apertium*-en *wikian* dauden datuen berritasuna ez dago argi, baina ziurrenik ez dira izango azken bertsioei dagozkienak, *wikian* agertzen diren datei erreparatuz gero.

Hizkuntz-bikotea	WER
Euskara - Gaztelania	% 72.4
Galesera - Ingelera	% 55.7
Gaztelania - Portugalera	% 4.7
Katalana - Portugalera	% 14.1
Portugalera - Gaztelania	% 11.3
Portugalera - Katalana	% 16.6
Suediera - Danesera	% 30.3

2.1 Taula: Hizkuntz-bikote batzuen eraginkortasunari buruzko datuak [31] erakusten dituen taula.

Azkenik, *Apertium* bukatu gabe dagoen proiektu bat denez, zerbitzariaren jardunaren fidagarritasuna batzuetan kaltetua izaten da. Izan ere, noizean behin zerbitzaria erori egiten da eta ondorioz ez dago erabilgarri itzulpen eskaerentzat.

3. KAPITULUA

Analisia eta Diseinua

Atal honetan produktuaren nondik-norakoari buruzko azalpen sakonagoa egiten da, funtzionamendua hobeto ulertzeko asmoz. Produktuaren bi osagai ezberdinen arteko menpekotasuna egon arren bien arteko banaketa argia dago. Hori dela eta, osagai bakoitzaren azalpena egingo da banaka-banaka, produktuaren ikuspegi globala lortu arte. Azalpena bi fase ezberdinetan egingo da. Lehenik eta behin, osagai bakoitzaren analisiari buruz hitz egingo da eta horren ondoren diseinuari buruz.

3.1 Ohar Eleanitzak aplikazioa

Azpiatal honetan *Android* programazio lengoaiari garatuta dagoen *Ohar Eleanitzak* aplikazioaren analisi eta diseinuari buruz hitz egingo dugu.

3.1.1 Analisia

Ohar Eleanitzak izeneko aplikazioa proiekturako garatutako *Android* aplikazioa da. Aplikazio hori proiektuak dauzkan hiru osagai nagusietako bat da eta ziurrenik bukaerako erabiltzaileak produktuarekin (aplikazio zein Eduki Kudeatzaile) izango duen lehen harremana bideratuko du. Aplikazioaren helburu nagusia erabiltzaileari Eduki Kudeatzailearen eduki zehatz baterako sarbidea ematea izango da. Horretarako, erabiltzaileak aplikazioa martxan duelarik mugikor zein tabletaren argazki kamera erabiliko du QR kode bat irakurtzeko. QR kode batek berez informazio mota ezberdin ugari izan ditzakeen arren,

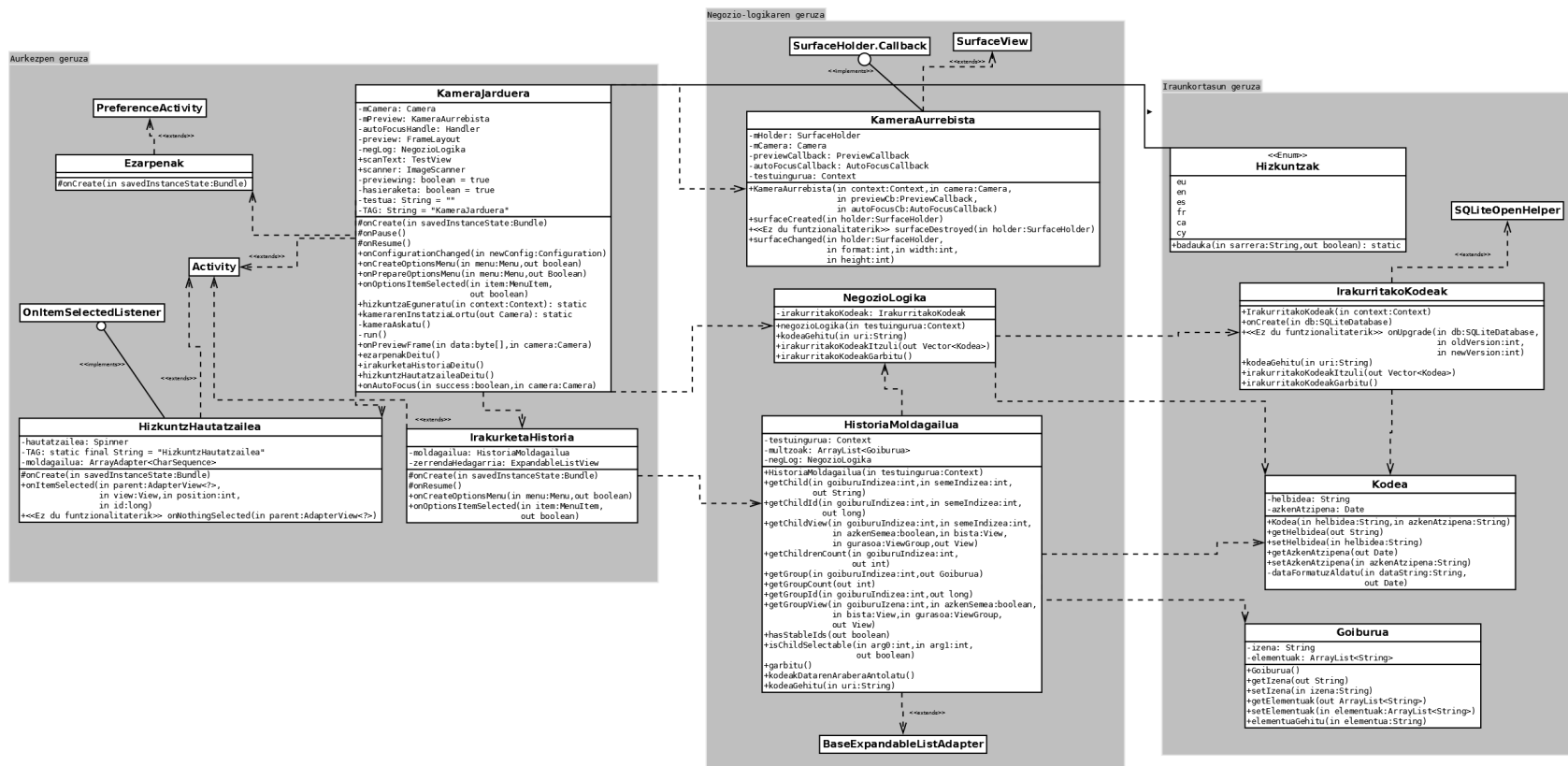
aplikazio honek erabiliko dituenak web-helbide bat gordeko dute. Aplikazioak irakurriko dituen kodeek barnean dituzten web-helbideak bi motatakoak izango dira: Interneten dagoen webgune arrunt batenak edo proiektu honen Eduki Kudeatzailearenak. Lehenengo kasuan, aplikazioak kodea irakurri eta kodeak barnean duen web-helbidea erauzi ostean bere horretan irekiko da arakatzailaren bidez. Bigarren kasuan, ordea, aplikazioak helbideari aldaketa bat egingo dio, aplikazioa ezarrita dagoen hizkuntzaren arabera izango dena. Eraldaketa horren bitartez QR kodeak gordetzen duen helbidea aplikazioa ezarrita dagoen hizkuntzan irekitzea ahalbidetzen da. Aldaketa egin denean aplikazioak web arakatzaila hasieratuko du irakurritako kodea atzitzeko.

Esanda bezala aplikazioa QR kodeak irakurtzeko gai da, baina ez da QR irakurle integral bat. Izan ere, web helbideak gordetzen dituzten QR kodeak soilik irakurriko ditu, ez dago prestatuta beste QR kode motak (SMS, koordenatuak. . .) irakurtzeko, horiei eman beharreko trataera proiektuaren helburuetatik kanpo dagoelako. Web helbide bat barnean daukan edozein QR kode irakurtzeko gai denez lehen aipatu dugun *attaging* delakoa gertatzeko aukerak daude. Horregatik, aplikazioak erabiltzaileari arriskuaz jakinarazten dioten prebentzio neurriak hartzen ditu *attaging*-ari aurre egiteko.

3.1.2 Diseinua

Klase diagrama

Diseinuari dagokionez, jarraian erakusten da aplikazioaren klase diagrama.



3.1 Irudia: Aplikazioaren klase diagrama.

Ikus daitekeen bezala, klase diagrama hiru zatitan banatuta dago, 3-mailako arkiteturari jarraituz. Hori dela eta, aplikazioaren klaseak hiru geruza ezberdinetan daude banatuta: Aurkezpen geruza, Negozio-logikaren geruza eta Iraunkortasun geruza. Lehenengoan, erabiltzailearekin elkarrekintza zuzena duten klaseak daude eta klase bakoitzak duen interfaze grafikoa baliatuta komunikatuko dira aplikazioa eta erabiltzailea. Bigarrena, berriz, interfazearen eta datuak gordetzen diren klaseen arteko bitartekari lanak betetzeaz arduratzen da. Horretaz gain, interfazearen dinamikotasunak behar dituen zenbait ekintza egiteaz arduratzen da, datu gordailuarekin harremanetan jarri gabe. Azkenik, hirugarren geruza datu moten definizioaz eta datu horiek gordetzeaz arduratzen da.

Aplikazioak hainbat klase propio dauzka, eta horiek *Android*-en berezkoak diren beste klase batzuk erabiltzen edo hedatzen dituzte nolabait. Berezko klaseak *Ezarpenak*, *Goiburua*, *HistoriaMoldagailua*, *HizkuntzHautatzailea*, *IrakurketaHistoria*, *IrakurritakoKodeak*, *KameraAurrebista*, *KameraJarduera*, *Kodea* eta *Negoziologika* dira.

Funtzionalitate nagusien pisua eramaten duten klaseak *KameraJarduera*, *KameraAurrebista*, *IrakurritakoKodeak*, *HistoriaMoldagailua* eta *HizkuntzHautatzailea* dira. Lehenengoa, aplikazioaren hasieratzeaz eta kudeatzeaz arduratzen da eta, bigarrena, berriz, kameraren kudeaketa zuzenaz (kameraren instantzia sortu eta kudeatu) arduratzen da soilik. Horretaz gain, *IrakurritakoKodeak* izeneko klasea datu-basea atzitu eta maneiatzeaz arduratzen da, irakurritako kodeen historia erakutsi zein eguneratzeko. Bestalde, *HistoriaMoldagailua* izeneko klasea *IrakurritakoKodeak* klasetik datozen datuak interfazean kokatzeko prest jartzeaz arduratzen da. *HizkuntzHautatzailea* klasearen bidez, berriz, erabiltzaileari aukera ematen zaio aplikazioaren hizkuntza aukeratzeko. Izan ere, *Android* aplikazioek normalean mugikorra ezarrita dagoen hizkuntzan erakusten dira. *HizkuntzHautatzailea* klasearekin portaera hori eraldatu egiten da.

Negoziologika klasea, datu-basetik datozen datuak komunikatzeaz arduratzen da, bere egitekoa interfazearen eta iraunkortasun geruzaren arteko banaketa logiko bat egitea baita. *IrakurketaHistoria* eta *Ezarpenak* izeneko klaseak aplikazioaren historia eta ezarpen aldaketa kudeatzeko erabiltzen dira, hurrenez hurren. Azkenik, *Kodea* eta *Goiburua* izeneko klaseak beste klaseek erabiliko dituzten bi objektu definitzeko erabiltzen dira. Horretaz gain, diagraman ere ikus daiteke *enum* motako *Hizkuntzak* objektua, *KameraJarduera* klaseak erabiliko duena eta aplikazioak jasan ditzakeen hizkuntzak gordetzeaz arduratuko dena.

Aurkezpen geruza

KameraJarduera: *KameraJarduera* klaseak, diagraman ikus daitekeen bezala, *Activity* klasea hedatzen du. Hori dela eta *onCreate*, *onPause*, *onResume*, *onConfigurationChanged*, *onCreateOptionsMenu*, *onPrepareOptionsMenu* eta *onOptionsItemSelected* eragiketak berrinplementatzen ditu, bere beharretara moldatzeko.

- *onCreate()*: Funtzio hau aplikazioa abiarazten denean exekutatu da eta hemen hasieran bakarrik ezarri behar diren aldagai edo balioak soilik definituko dira. Esate baterako, interfazea eta ezarpenak kargatu etab.
- *onPause()*: Funtzio hau aplikazioa bigarren plano batera pasatzen denean (aplikazioa pantailan partzialki ikusiko da, baina kontrola beste aplikazioaren batek du) exekutatu da eta bere eginbehar nagusia kamera askatzea izango da.
- *onResume()*: Funtzio hau da aplikazioaren funtzionalitate nagusia kontrolatzen duena, kamera. Funtzio hau arduratuko da kamera kargatzeaz eta aldagaiei eman behar zaizkion balioak emateaz. Funtzio hau aplikazioa abiarazia izan denean edota pausa edo geldialdi egoeratik datorrenean exekutatu da.
- *onConfigurationChanged(Configuration newConfig)*: Funtzio hau aplikazioaren manifestuan definitutako ezarpenetan aldaketaren bat gertatzean exekutatu da. Bere egitekoa aldaketa gertatzen den bakoitzean hizkuntz egokia kargatzea da.
- *onCreateOptionsMenu(Menu menu)*: Funtzio honen bidez erabiltzaileak aplikazioaren aukerak ikusi nahi dituenean menu barra agerrarazteko egin beharreko ekintzak burutuko ditu.
- *onPrepareOptionsMenu(Menu menu)*: Funtzio hau aukeren barra erabiltzaileari erakutsi aurretik exekutatzen da, testuak hizkuntza egokian ezartzeko.
- *onOptionsItemSelected(MenuItem item)*: Funtzio hau erabiltzaileak menu barrako aukera bat aukeratzean, hori irekitzeaz arduratuko da.

Klaseak dituen beste funtzioak:

- *hizkuntzaEguneratu(Context context)*: Funtzio hau arduratzen da aplikazioa hizkuntza egokian erakusten dela ziurtatzeaz. Izan ere, lehenago aipatu den bezala, *Android* aplikazioek hizkuntza bezala mugikorraren hizkuntza aukeratzen dute era lehenetsian. Portaera hori ekiditeko, jarduera bat irekitzen den bakoitzean deitu behar zaio eragiketa honi. Eragiketak ezarpenetan begiratzen du erabiltzaileak hautatu duen hizkuntza eta aplikazioa hizkuntza horretan ezartzen du.

- *kamerarenInstantziaLortu()*: Funtzio hau erabiltzen da gainontzeko funtzioek kameraren instantzia lortzeko. Horrela ziurtatu nahi da kamera ez dela behin baino gehiagotan irekitzen, horrek errorea sortuko luke eta.
- *kameraAskatu()*: Funtzio honekin, berriz, kamerari erabilerarik eman behar ez zaionean (aplikazioa pausa edota geldialdi egoeran dagoelako) kamera askatzen da, beste aplikazio batzuei kamera erabiltzea ahalbidetuz.
- *run()*: Funtzio hau kamerak modu jarraian fokatuko duela zehazteko erabiltzen da.
- *onPreviewFrame(byte[] data, Camera camera)*: Funtzio hau beharrezkoa da *PreviewCallback* motako objektua erabiltzeko, kameraren fokatzeaz arduratzen dena. Funtzio honekin kamera QR kode batera fokatzen dugunean bere kabuz bereiziko du kodea, guk argazkirik atera beharrik gabe, hau da, bere aurrean kode bat duen bakoitzean zuzenean prozesatzea egingo du, erabiltzailearen beste elkarrekintzarik gabe. Kodeak dakarren informazioa erauztean, estekari hizkuntzaren arabera aldaketa egiteaz arduratuko da (beharrezkoa den kasuetan) eta baita web arakatzailerari deitzeaz ere.
- *ezarpenakDeitu()*: honen bidez ezarpenak agertzen diren pantaila irekitzen da.
- *irakurketaHistoriaDeitu()*: eragiketa honen bidez erabiltzaileak irakurri dituen kodeen historia atzitzen da.
- *hizkuntzHautatzaileaDeitu()*: eragiketa honek erabiltzaileari aplikazioaren hizkuntza aukeratzeko aukera ematen dion interfazea irekitzen du.
- *onAutoFocus(boolean success, Camera camera)*: funtzio hau kameraren fokatze automatikoaz arduratuko da, QR kodeak irakurtzeko beharrezkoa dena. Funtzio hau beharrezkoa da *AutoFocusCallback* motako objektua erabiltzeko.

IrakurketaHistoria: Klase honek *Activity* izeneko klasea hedatzen du, eta hori dela eta, *onCreate*, *onResume*, *onCreateOptionsMenu*, *onOptionsItemSelected* eta *onChildClick* eragiketak berrinplementatzen ditu.

- *onCreate(Bundle savedInstanceState)*: Klasea hasieratzen den unean interfazea eta irakurritako kodeen historia kargatu zein erakusteaz arduratzen da, aplikazioan ezarrita dagoen hizkuntza egokia dela ziurtatzeaz gain.

- *onResume()*: Funtzio hau aplikazioa abiarazia izan denean edota pausa edo geldialdi egoeratik datorrenean exekutatu da. Eragiketa honen bidez irakurketa historiaren edukia erakusten da.
- *onCreateOptionsMenu(Menu menu)*: Eragiketa hau aukeren menua hasieratzeaz arduratzen da aukerak erakusteko botoia sakatzen den unean.
- *onOptionsItemSelected(MenuItem item)*: Erabiltzaileak aukera menuko elementu bat sakatzean (bakarra dago, historia garbitzeaz arduratzen dena) alerta mezu bat irekiko du konfirmazioa eskatuz. Baietz esatekotan historia osoa garbituko du, bestela, historia dagoen bezala erakutsiko du.
- *onChildClick(ExpandableListView gurasoa, View vista, int goiburuIndizea, int semeIndizea, long id)*: Goiburu motako objektu bakoitzaren barruko elementuetan gertatzen diren sakatzeak tratatzen ditu, sakatu den kodea arakatzailan irekiz eta datu-basean kode hori atzitu den azken aldia eguneratuz.

HizkuntzHautatzailea: Klase honen bidez erabiltzaileak aukera dauka aplikazioaren hizkuntza aldatzeko. Jarduera bat denez, *Activity* klasea hedatzen du eta erabiltzaileak aukeratu duen hizkuntza jakin ahal izateko *OnItemSelectedListener* interfazea inplementatzen du. Hori dela eta, honako eragiketak birdefinitu edo inplementatzen ditu.

- *onCreate(Bundle savedInstanceState)*: Funtzio hau jarduera hasieratzeaz arduratzen da. Dauzkan egin beharren artean interfazea kargatu, hizkuntza egokia kargatu eta unean hizkuntza hautatzailean aukeratuta dagoen hizkuntza aldeztatik aukeratuta erakustea daude.
- *onItemSelected(AdapterView<?> parent, View view, int position, long id)*: Funtzio honek ezarpenak eta aplikazioaren hizkuntza eguneratuko ditu hizkuntza hautatzailean hizkuntza bat aukeratzen den bakoitzean.
- *onNothingSelected(AdapterView<?> parent)*: Funtzionalitaterik gabeko funtzioa. *OnItemSelectedListener* interfazea inplementatzearen ondorioz klasean definituta egon behar da.

Ezarpenak: Ezarpenak klaseak ere beste klase bat hedatzen du, *PreferenceActivity*, eta bere beharretara moldatzeko, *PreferenceActivity* klaseko *onCreate* eragiketa berrinplementatzen du.

- *onCreate()*: ezarpenak erakutsi zein aldatzeaz arduratzen den jarduera irekitzean egin beharrekoaz arduratzen da. Horretaz gain, barne kudeaketarako erabiltzen den *hautatutakoHizkuntza* ezarpena ezkututzen du.

Negozio-logikaren geruza

KameraAurrebista: *KameraAurrebista* klasea, *SurfaceHolder.Callback* interfazea inplementatzeaz arduratzen da, *surfaceCreated*, *surfaceDestroyed* eta *surfaceChanged* eragiketen bitartez. Horretaz gain, eraikitzaile bat ere badu.

- *KameraAurrebista* (*Context context*, *Camera camera*, *PreviewCallback previewCb*, *AutoFocusCallback autoFocusCb*): funtzio honen bidez *KameraAurrebista* motako objektu bat sortzen da .
- *SurfaceCreated*(*SurfaceHolder holder*): Honen bidez kamera objektuaren hasieratzea tratatzen da eta kameraren bidez lortutako aurrebistak(edo irudiak) non erakutsi zehazten da.
- *SurfaceDestroyed*(*SurfaceHolder holder*): *SurfaceHolder.Callback* izeneko interfazea erabiltzeagatik egon behar den eragiketa. Ez du inolako funtzionalitaterik.
- *SurfaceChanged* (*SurfaceHolder holder*,*int format*, *int width* *int height*): Funtzio hau kameraren aldaketaren bat gertatzen denean egin beharrekoaz arduratzen da, esate baterako, kameraren irudiak mugikorra horizontalean edo paisaia moduan dagoela ondo ikus daitezzen.

NegozioLogika: Klase hau datu-baseaz arduratzen den *IrakurritakoKodeak* klasearen eta datu-baseak dituen edukiak behar dituzten klaseen arteko bitartekaritza egiteaz arduratzen da, banaketa logiko bat sortuz. Klase honen metodoak jarraian azaltzen dira:

- *NegozioLogika*(*Context testuingurua*): Eraikitzailea.
- *kodeaGehitu*(*String uri*): datu-basean kode berri bat gehitzeaz arduratzen den *IrakurritakoKodeak* klasearen *kodeaGehitu* eragiketari deitzen dio, jasotako aldagai berdinarekin.

- *irakurritakoKodeakItzuli()*: *IrakurritakoKodeak* klaseko *irakurritakoKodeakItzuli* metodoari deitzen dio eta jasotako erantzuna itzuliko du, *Kodea* motako objektuez osatutako *Vector* bat alegia.
- *irakurritakoKodeakGarbitu()*: *IrakurritakoKodeak* klasean implementatuta dagoen izen bereko metodoari deitzen dio, datu-basearen edukiak ezabatzeko.

HistoriaMoldagailua: Klase honek *BaseExpandableListAdapter* izeneko klasea hedatzen du. Hori dela eta, aipaturiko klasean definitzen diren hainbat eragiketa abstraktu definitu behar ditu.

- *getChild(int goiburuIndizea, int semeIndizea)*: *goiburuIndizea* indizea dagoen goiburuaren barneko *semeIndizea* posizioan dagoen elementua itzuliko du *String* formatuan.
- *getChildId(int goiburuIndizea, int semeIndizea)*: *goiburuIndizea* indizea dagoen goiburuaren barneko *semeIndizea* posizioan dagoen elementuaren identifikadorea itzuliko du.
- *getChildView(int goiburuIndizea, int semeIndizea, boolean azkenSemea, View bista, ViewGroup gurasoa)*: *goiburuIndizea* indizea dagoen goiburuaren barneko *semeIndizea* posizioan dagoen elementua adierazten duen bista itzuliko du.
- *getChildrenCount(int goiburuIndizea)*: *goiburuIndizea* posizioan dagoen goiburuaren elementu kopurua itzuliko du.
- *getGroup(int goiburuIndizea)*: *goiburuIndizea* posizioan dagoen goiburua itzuliko du.
- *getGroupCount()*: dagoen goiburu kopurua itzuliko du.
- *getGroupId(int goiburuIndizea)*: *goiburuIndizea* posizioan dagoen goiburuaren identifikadorea itzuliko du.
- *getGroupView(int goiburuIndizea, boolean azkenSemea, View bista, ViewGroup gurasoa)*: *goiburuIndizea* posizioan dagoen goiburua adierazten duen bista itzultzen du.

- *hasStableIds()*: elementu bakoitza ezberdintzeko erabiltzen diren identifikadoreak egonkorak diren, hau da, elementuren bat ezabatzean gainontzeko elementuen identifikadoreak aldaketarik jasaten duten ala ez, esaten duen funtzioa. Horretarako, balio *boolear* bat itzultzen du.
- *isChildSelectable(int arg0, int arg1)*: Goiburuen barruan dauden elementuak hautagarriak al diren, hau da, hautatuak direnean portaera bat izango al duten, esaten duen eragiketa, balio *boolear* bat erabilita.

Gainera, berezko eragiketa hauek ere baditu:

- *HistoriaMoldagailua(Context testuingurua)*: Klasearen eraikitzailea. Klasearen aldagaiak hasieratzeaz arduratzen da. Horretaz gain, irakurketa historiaren elementuak formatu egokiaren ordenatzeaz ere arduratzen da. Formatu egoki hori lortzeko, *Goiburu*a motako objektuak erabiltzen dira.
- *Garbitu()*: Datu-basea ezabatu eta moldagailua garbitzeaz arduratzen da.
- *kodeakDatarenAraberaAntolatu()*: *NegozioLogika* klasearen bidez datu basea kudeatzeaz arduratzen den *IrakurritakoKodeak* klasetik lortutako edukiak antolatzen ditu, interfazean erakutsi ahal izateko. Horretarako, *Goiburuak* erabiltzen dira, bakoitzak denbora-espazio bat (Gaur, Atzo, Azken astea. . .) adierazten duelarik. *Goiburu* horien barruan, azken aldiz denbora-espazio horretan irakurri ziren helbideak gehitzen dira.
- *kodeaGehitu(String uri)*: datu-basean kode bat gehitzeaz arduratzen da, *NegozioLogika* bitartekari bezala erabiliz.

Iraunkortasun geruza

IrakurritakoKodeak: *IrakurritakoKodeak* klaseak *SQLiteOpenHelper* izeneko klasea hedatzen du, *onCreate* eta *onUpgrade* eragiketak inplementatuz.

- *onCreate(SQLiteDatabase db)*: Eragiketa hau aplikazioa exekutatzen den lehen al-dian erabiliko dugun datu-basearen taula sortzeaz arduratzen da.

- *onUpgrade (SQLiteDatabase db, int oldVersion, int newVersion)*: *SQLiteOpenHelper* klasea hedatzeagatik eragiketa abstraktu hau inplementatu behar da, datu-basearen bertsio berriren bat ezartzen denerako. Eragiketa honek ez du inolako funtzionalitateirik.

Klaseak dituen berezko funtzioak:

- *IrakurritakoKodeak(Context context)*: Klasearen eraikitzailea da. Lehen exekuzioan datu-basea sortzeaz arduratzen da.
- *kodeaGehitu(String uri)*: Eragiketa honek historian erabiltzaileak irakurri berri duen kodea gehituko du, lehendik ez badago. Kode hori lehendik irakurria izan bada datu-baseko *azkenAtzipena* zutabeko balioa eguneratuko da, uneko momentua jarriz.
- *irakurritakoKodeakItzuli()*: Mugikorraren historian dauden kodeak itzuliko ditu *Vector<String>* motako aldagai batean, azken atzipen dataren arabera antolatuta (berrienetik zaharreneira), historia garbitua izan ez bada.
- *irakurritakoKodeakGarbitu()*: Erabiltzaileak irakurri dituen kodeen historia garbituko du datu-basetik.

Kodea: Klase honetan definitutako objektua erabiltzaileak irakurritako kodeak irudikatzeke erabiltzen da. Horretarako, helbidea eta azken atzipenaren balioa dauzka atributu bezala. Ondorengoak dira inplementatuta dauzkan eragiketak.

- *Kodea(String helbidea, String azkenAtzipena)*: Eraikitzailea.
- *getHelbidea()*: helbidearen balioa itzultzen du, *String* formatuan.
- *setHelbidea(String helbidea)*: Jasotako *String* balioa objektuaren helbidea atributuan ezartzen du.
- *getAzkenAtzipena()*: Objektuak irudikatzen duen kodearen azken atzipen data itzultzen du, *Date* gisa.
- *setAzkenAtzipena(String azkenAtzipena)*: Objektuak irudikatzen duen kodearen azken atzipen data eguneratzen du.
- *dataFormatuzAldatu(String dataString)*: Dataren balioa *String* objektu batetik *Date* motako objektu batera pasatzen duen eragiketa.

Goiburua: Klase honek definitzen duen objektua *HistoriaMoldagailua* klaseak erabiliko du, irakurketa historia formatu egoki batean jartzeko. Bi atributu ditu, denbora-espazioa (Gaur, Atzo, Azken astea. . .) definitzeko erabiltzen den *String* motako objektua eta *String* motako objektuz betetako *ArrayList*-a, azken aldiz goiburuak irudikatzen duen denbora-espazioaren barruan irakurri diren kodeen helbideak gordetzeko erabiltzen dena. Honako hauek dira klase honen eragiketak:

- *Goiburua()*: Eraikitzailea.
- *getIzena()*: *Goiburuaren* izena itzultzen du, *String* objektu baten bidez.
- *setIzena(String izena)*: *Goiburuaren* izena aldatzen du, jasotako balioa izen bezala ezarriz.
- *getElementuak()*: *Goiburuaren* barruan gordeta dauden helbideak itzultzen ditu, *String* motako objektuz betetako *ArrayList* baten bidez.
- *setElementuak(ArrayList<String> elementuak)*: *Goiburuaren* barruan gordeta dauden helbideak aldatzen ditu, jasotako balioa ezarriz.
- *elementuaGehitu(String elementua)*: *Goiburuaren* barruan dauden elementuei beste elementu bat gehitzen die.

Hizkuntzak: Enum motako klase honek, aplikazioa eskuragarri dagoen hizkuntzak gordetzen ditu. Hizkuntza horiek Euskara (eu), Ingelera (en), Gaztelania (es), Frantsesera (fr) eta Katalana (ca) dira. Klase honek ondorengo eragiketa hau dauka:

- *badauka (String sarrera)*: Funtzio honek parametro gisa jasotako balioa *Hizkuntzak enum* klaseak gordeta dauzkan balioekin konparatzen du, eta balio hori topatu al den jakinarazten du boolear motako objektu bat itzulita.

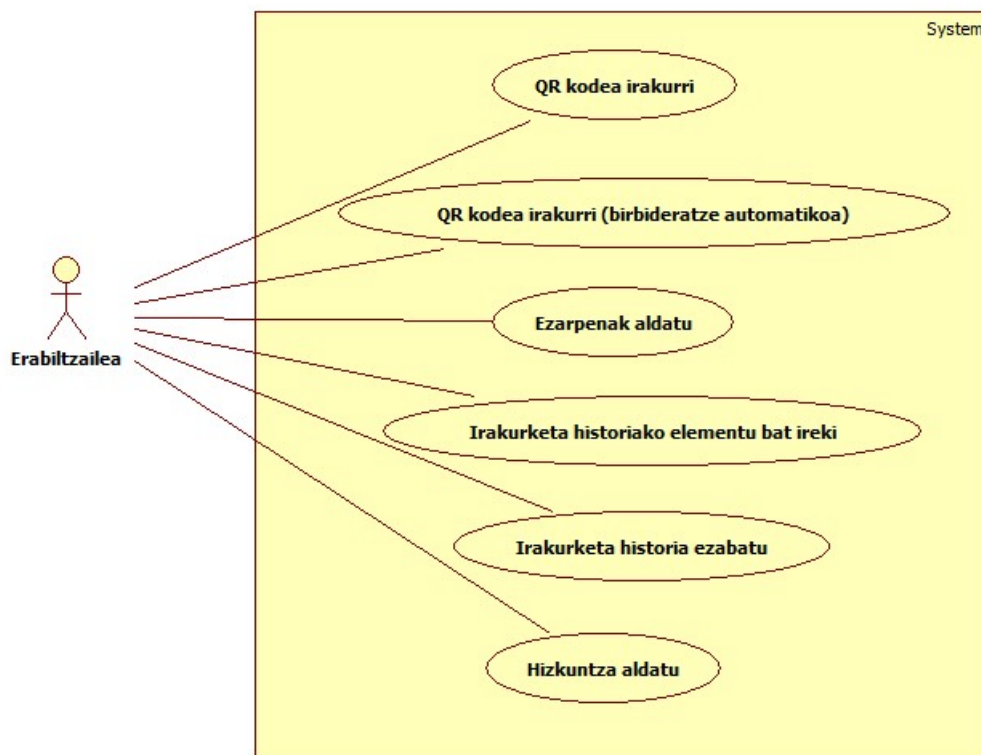
Datu-basea

Kodeak	
PK	<u>helbidea</u>
	azkenAtzipena

3.2 Irudia: Aplikazioaren datu-basearen diagrama.

Goiko irudian ikusi daitekeen bezala, aplikazioak erabiliko duen datu basea oso sinplea da. Hasteko, taula bakarra dauka eta gainera bi atributu dauzka soilik. Datu-basearen taularen izena Kodeak da, eta bertan erabiltzaileak irakurri dituen kodeek gordetzen dituzten web helbideak biltegitratzen dira. Horretarako, *helbidea* izeneko atributua dago, taularen oinarrizko gakoa dena. Atributu horretan erabiltzaileak irakurritako web helbideak gordetzen dira. Bestetik, *azkenAtzipena* izeneko atributua dago, web helbide bakoitza atzitu den azken aldiaren datuak gordetzen dituen (eguna eta ordua).

Erabilpen-kasuak



3.3 Irudia: Aplikazioaren erabilpen-kasuen diagrama.

Goiko diagraman ikusi daitekeen bezala, aplikazioa erabiltzeko aktore mota bakarra dago, Erabiltzailea. Hori horrela da aplikazioa produktuaren eta erabiltzaileen arteko bitartekaria delako. Aplikazioaren funtsa erabiltzaileari Eduki Kudeatzailerako sarrera ematea da, QR kodea irakurri ondoren. Hori egin ahal izateko dagoen aurrebaldintza bakarra, noski, aplikazioa mugikorrean edukitzea da. Administrari funtzioak egiteko (edukiak gehitu, eraldatu...) derrigorrezkoa izango da, beraz, Eduki Kudeatzailera zuzeneko sarbidea izatea, beste bitartekaririk gabe. Erabiltzaileak egin ditzakeen erabilpen-kasuak, hortaz, *QR kodea irakurri*, *QR kodea irakurri (birbideratze automatikoa)*, *Ezarpenak aldatu*, *Irakurketa historiako elementu bat ireki*, *Irakurketa historia ezabatu* eta *Hizkuntza aldatu* dira.

QR kodea irakurri

- Izenburua: QR kodea irakurri

Testuingurua: Ezarpenetan badago aukera *Birbideratzea Automatikoki Onartu* deitzen dena, balio lehenetsi bezala ezgaituta dagoena. Ezgaitua dagoenean erabiltzaileak QR kode bat irakurtzen duen bakoitzean, aplikazioak erabiltzaileari atzitu nahi den esteka erakutsiko dio, eta konfirmazioa eskatu. Hori honela izatearen arrazoia lehendik aipaturiko *attaging* delakoa saihestea da, erabiltzaileari une oro atzitu den webgunea erakutsiz, segurtasuna indartzeko asmoz.

Aktorea: Erabiltzailea

Aurrebaldintzak:

- Mugikorrean aplikazioa izatea.
- Ezarpenetan *Birbideratzea Automatikoki Onartu* aukera ezgaituta izatea.

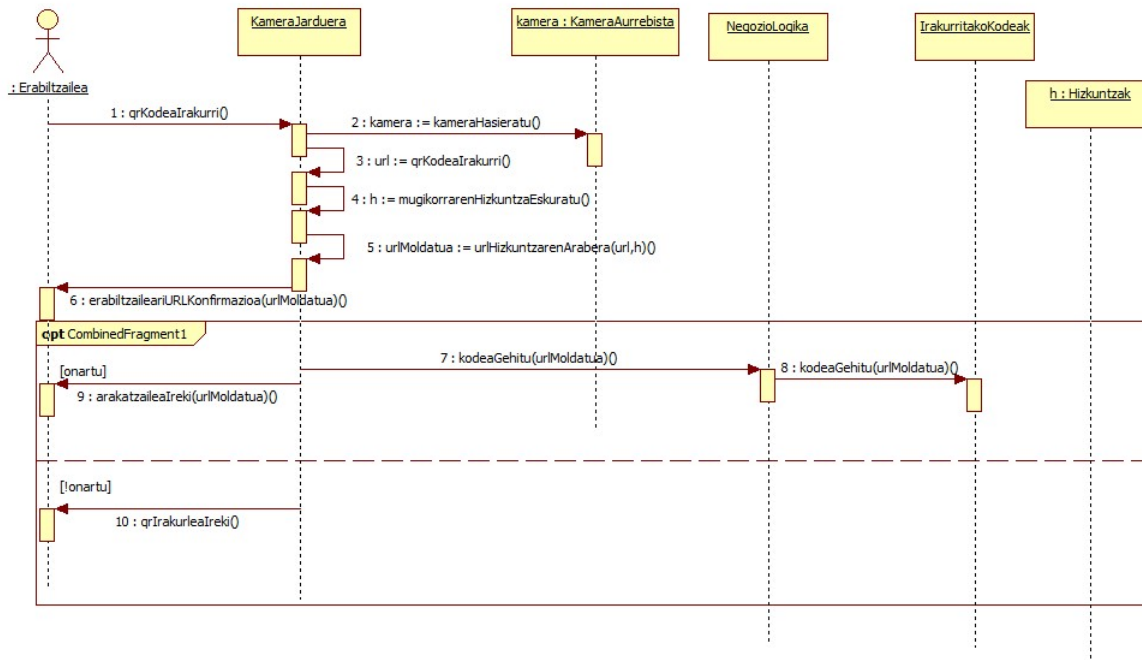
Fluxu normala:

1. Erabiltzailea QR kodeak irakurtzeko gunean dagoela, QR kode bat fokatu.
2. Aplikazioak irakurri egingo du, beharrezkoa bada helbidea aplikazioaren hizkuntzaren arabera eraldatuko du eta helbidea erakutsiko dio erabiltzaileari, konfirmazioa eskatuz.
3. Erabiltzaileak onartuz gero, aplikazioak web arakatzaileri deia egingo dio, web helbidea atzitu dezan. Atzitu aurretik, sistemak web helbidea datu-basean gordeko du.

Ez-ohiko fluxua:

3. Erabiltzaileak konfirmazio mezua ezeztatzeari ematen badio, QR kodeak irakurtzeko orrira joango da.

Sekuentzia-diagrama:



3.4 Irudia: QR kodea irakurri erabilpen-kasuaren sekuentzia-diagrama.

QR kodea irakurri (birbideratze automatikoa)

- Izenburua: QR kodea irakurri (birbideratze automatikoa)

Testuingurua: Ezarpenetan badago aukera *Birbideratzea Automatikoki Onartu* deitzen dena, balio lehenetsi bezala ezgaituta dagoena. Gaituta dagoenean erabiltzaileak QR kode bat irakurtzen duenean zuzenean web arakatzailari deia egingo dio.

Aktorea: Erabiltzailea

Aurrebaldintzak:

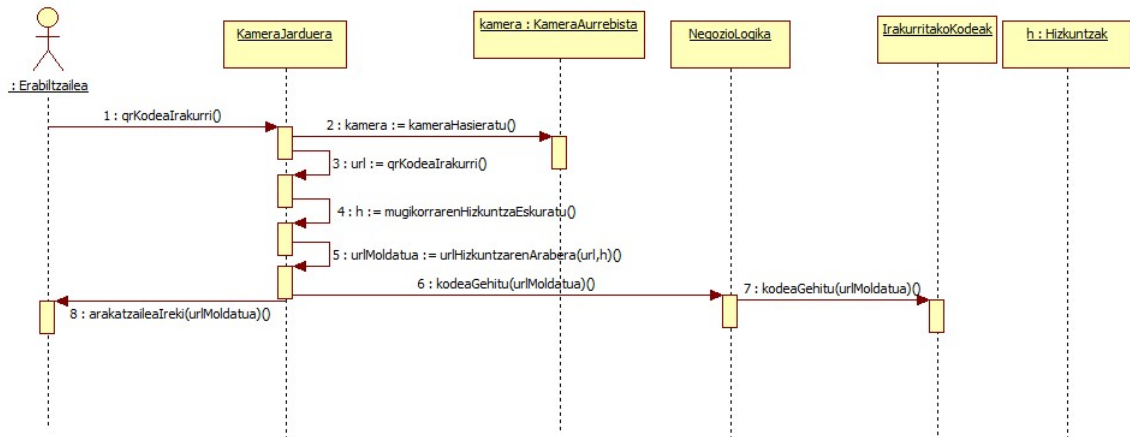
- Mugikorrean aplikazioa izatea.
- Ezarpenetan *Birbideratzea Automatikoki Onartu* aukera gaituta izatea.

Fluxu normala:

1. Erabiltzailea QR kodeak irakurtzeko gunean dagoela, QR kode bat fokatu.

2. Aplikazioak irakurri egingo du eta beharrezkoa den kasuetan mugikorraren hizkuntzaren arabera irakurritako helbidea eraldatu ondoren, web arakatzai-
learen bidez web esteka atzitu du. Arakatzai-leari deitu aurretik, sistemak
web helbidea datu-basean gordeko du.

Sekuentzia-diagrama:



3.5 Irudia: QR kodea irakurri (birbideratze automatikoa) erabilpen-kasuaren sekuentzia-diagrama.

Ezarpenak aldatu

- Izenburua: Ezarpenak aldatu

Testuingurua: Aurreko bi erabilpen-kasuetan aipatu den bezala, aplikazioak *Birbideratzea Automatikoki Onartu* ezarpena dauka. Ezarpen horren balioa eraldatzeko prozedura azaltzen da hemen.

Aktorea: Erabiltzailea.

Aurrebaldintzak:

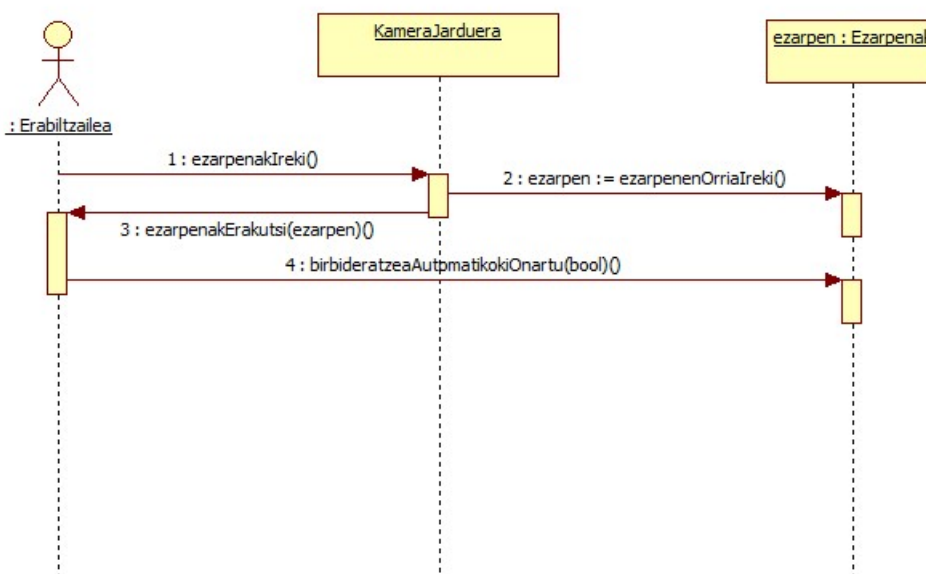
- Mugikorrean aplikazioa izatea.

Fluxu normala:

1. Erabiltzailea aplikazioaren hasiera gunean dagoela, ezarpenak eraldatzeko aukerari eman.

2. Aplikazioak ezarpenak agertzen diren orrira eramango du erabiltzailea.
3. Erabiltzaileak *Birbideratzea Automatikoki Onartu* ezarpenari egoki deritzon balioa emango dio.

Sekuentzia-diagrama:



3.6 Irudia: Ezarpenak aldatu erabilpen-kasuaren sekuentzia-diagrama.

Irakurketa historiako elementu bat ireki

- Izenburua: Irakurketa historiako elementu bat ireki

Testuingurua: Erabiltzailea aplikazioarekin kodeak irakurriz doan heinean, irakurri-tako kode horiek dituzten web-helbideak gordetzen doa datu-base batean. Horrela, erabiltzaileak lehenagotik irakurri duen kode baten web-helbidea atzitu nahi badu historiara joatearekin nahikoa du, berriro ere kodea irakurri behar izanik gabe, bertan atzitu nahi duen helbidea aukeratuz.

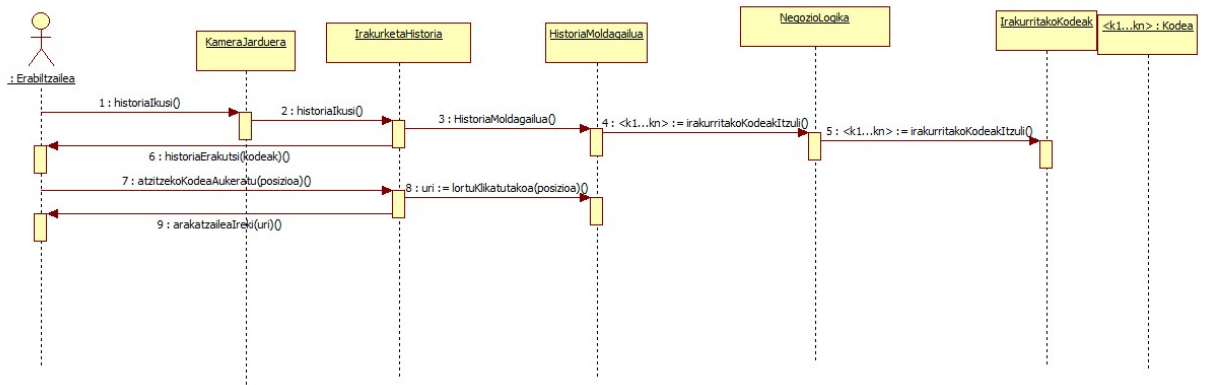
Aktorea: Erabiltzailea

Aurrebaldintzak:

- Mugikorrean aplikazioa izatea.

Fluxu normala:

1. Erabiltzailea aplikazioaren gune nagusian dagoela, historia ikusteko aukerari eman.
2. Sistemak datu-basetik historia atzitu du eta erabiltzaileari erakutsiko dio, azken atzipen dataren arabera ordenatuta.
3. Erabiltzaileak historiatik atzitu nahi duen web helbidea aukeratu du.
4. Sistemak web helbide hori irekiko du web arakatzailarekin.

Sekuentzia-diagrama:

3.7 Irudia: Irakurketa historiako elementu bat ireki erabilpen-kasuaren sekuentzia-diagrama.

Irakurketa historia ezabatu

- Izenburua: Irakurketa historia ezabatu

Testuingurua: *Irakurketa historiako elementu bat ireki* erabilpen-kasuan aipatu bezala, aplikazioak erabiltzaileak irakurritako QR kodeen edukia datu-basean gordetzen du, gero irakurketa historia moduan erakusteko.

Aktorea: Erabiltzailea

Aurrebaldintzak:

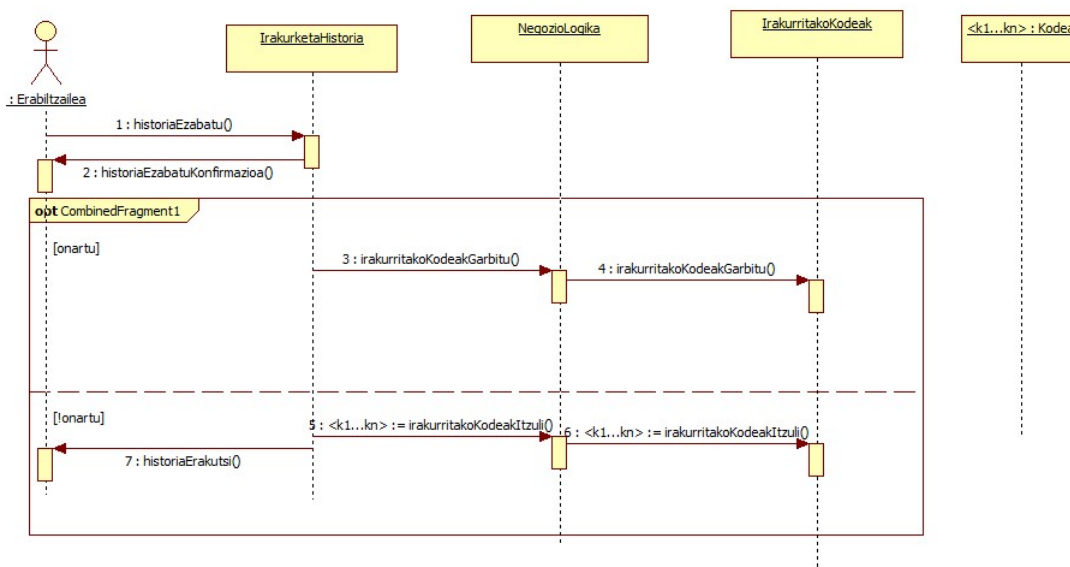
- Mugikorrean aplikazioa izatea.

Fluxu normala:

1. Erabiltzailea irakurketa historia ikusten ari dela historia ezabatzearen aukera sakatu.
2. Sistemak konfirmazio mezu bat erakutsiko du, historia ezabatu nahi al duen galdetuz.
3. Erabiltzaileak onartuz gero, aplikazioak historia ezabatu egingo du.

Ez-ohiko fluxua:

3. Erabiltzaileak konfirmazio mezua ezeztatzeari ematen badio, historia ez da ezabatuko, eta erabiltzaileak historia erakusten duen orrialdean jarraituko du.

Sekuentzia-diagrama:

3.8 Irudia: Irakurketa historia ezabatu erabilpen-kasuaren sekuentzia-diagrama.

Hizkuntza aldatu

- Izenburua: Hizkuntza aldatu

Testuingurua: Aplikazioak erabiltzaileari hizkuntza aldatzeko aukera ematen dio.

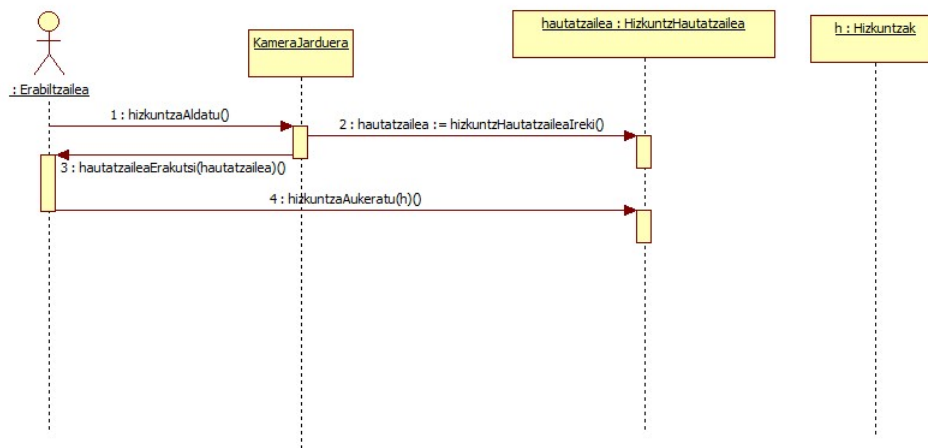
Aktorea: Erabiltzailea.

Aurrebaldintzak:

- Mugikorrean aplikazioa izatea.

Fluxu normala:

1. Erabiltzailea aplikazioaren hasiera gunean dagoela, hizkuntza aldatzeko aukerari eman.
2. Aplikazioak hizkuntza aldatzen den orrira eramango du erabiltzailea.
3. Erabiltzaileak aplikazioarentzat nahi duen hizkuntza aukeratuko du.

Sekuentzia-diagrama:

3.9 Irudia: Hizkuntza aldatu erabilpen-kasuaren sekuentzia-diagrama.

3.2 Eduki Kudeatzailea

Azpiatal honetan, berriz, zerbitzarian kokatuta dagoen Eduki Kudeatzailearen analisi eta diseinuari buruz hitz egingo dugu.

3.2.1 Analisia

Wordpress Edukiak Kudeatzeko Sistemaren bidez garatutako Eduki Kudeatzailea hainbat hizkuntza ezberdinetan dago. Eduki Kudeatzailearentzat bi erabiltzaile-profil nagu-

si identifikatu dira, bakoitzak Eduki Kudeatzailea atzitzeko bere modua duelarik. Bate-tik, *Android* aplikazioaren bitartez QR kode bat irakurri eta Eduki Kudeatzailea atzitu duen erabiltzailea egongo da. Honek, Eduki Kudeatzailea *Ohar Eleanitzak* aplikazioaren bitartez atzitzeko du, QR kode bat irakurri ostean. Bestetik, Eduki Kudeatzailearen bere erabiltzaile-kontuarekin sartu den erabiltzaile erregistratua izango dugu, arakatzailerik normal batekin konektatuko dena. Horretaz gain, maiz gertatuko denik aurreikusten ez den arren, erabiltzailearen batek Eduki Kudeatzailea zuzenean atzitu lezake *Android* aplikazioa erabili beharrik gabe, Eduki Kudeatzailearen helbidea arakatzaileran idatziz.

Hortaz, aipatutako bi erabiltzaile-profil ezberdinek Eduki Kudeatzailea modu ezberdinean erabiliko dute. Aplikazioaren bidez atzitu duenak Eduki Kudeatzailearen edukia ikusteko aukera izango du. Aplikazioaren analisisian esan den bezala, QR kodeak gordetzen duen helbidea aplikazioa ezarrita dagoen hizkuntzan irekiko da. Hala ere, helbidea web-arakatzaileraren bitartez irekitzen denez erabiltzaileak aukera izango du Eduki Kudeatzailean dauden beste eduki batzuk bisitatzeko edota uneko edukiaren hizkuntza aldatzeko.

Eduki Kudeatzailea askotan gailu mugikorretatik atzitzeko denez edukiaren antolamendu egokia izan beharko du. Izan ere, gailu horien pantaila txikia denez ziurtatu egin behar da gailu mugikorretatik Eduki Kudeatzailean nabigatzea erraza dela. Bestela, Eduki Kudeatzailea gailu mugikorretatik atzitzen duten erabiltzaileak ez dira gai izango beraien helburua betetzeko (irakurritako QR kodeak gordetzen duen informazioa irakurtzea) eta ziurrenik erabiltzeari utziko diote, pazientzia galdu ondoren.

Bestetik, erabiltzaile erregistratuak Eduki Kudeatzailea atzitzean kudeaketa lanak egiteko aukera izango du. Beraz, *WordPress*-ek ematen dituen aukerak egiteko gaitasuna izango du, adibidez, eduki berria sortu, Eduki Kudeatzailearen itxura aldatu etab. Gainera, Eduki Kudeatzailearako bereziki garatutako *Ohar Itzultzailea* izeneko gehigarriari esker, Eduki Kudeatzailean dagoen edukia itzultzeko aukera egongo da, baita eduki horientzat QR kodeak sortzeko aukera ere.

3.2.2 Diseinua

Eduki Kudeatzailearen barnean bi osagai ezberdin daudenez, horien diseinua banan banan egingo da. Lehenik eta behin, eduki eta gehigarriarentzako azpiegitura jartzen duen *WordPress* webguneari buruz hitz egingo dugu eta horren ostean, *Ohar Itzultzailea* gehigarriari buruz.

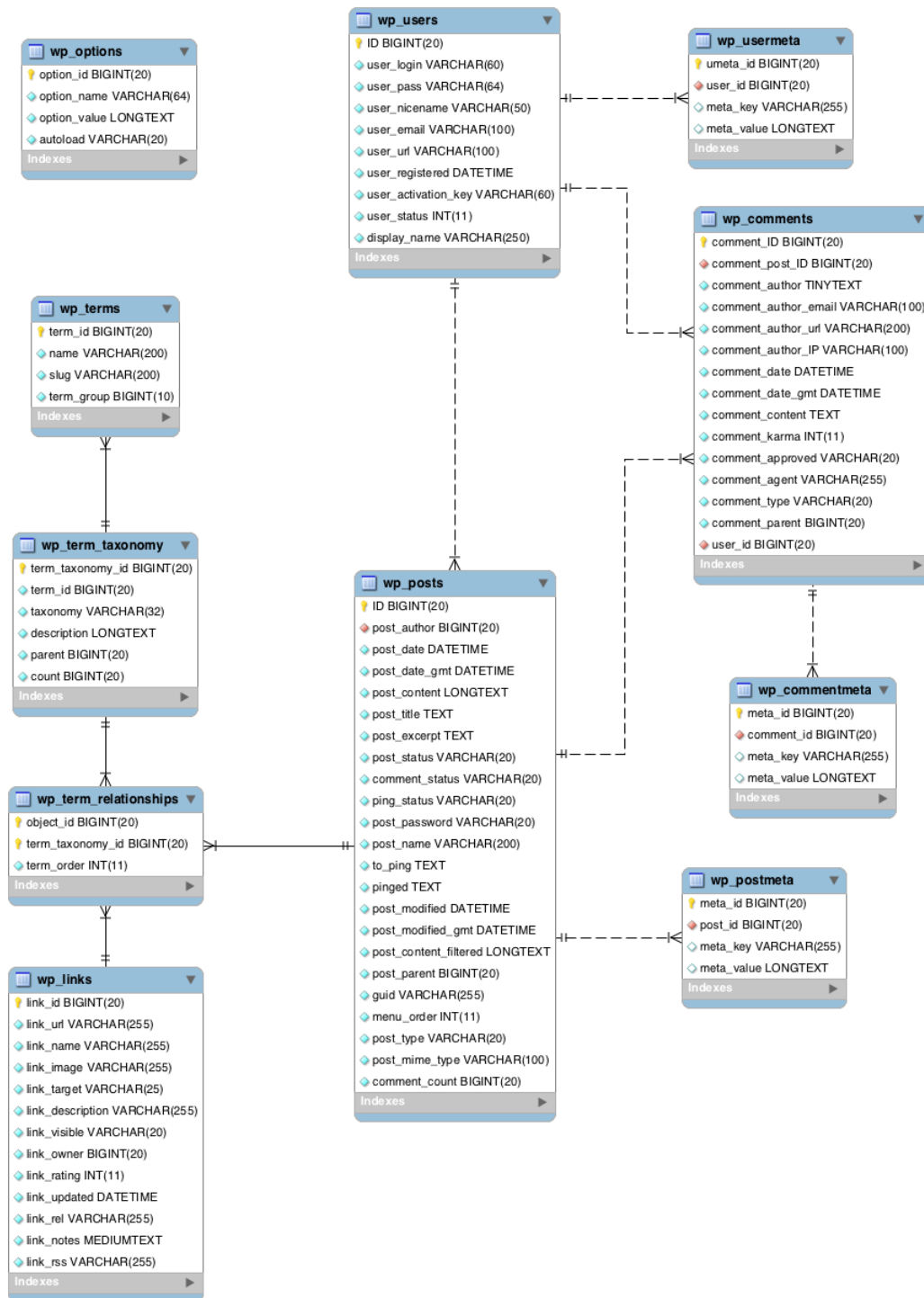
WordPress webgunea

WordPress Edukiak Kudeatzeko Sistemak hamar urtetik gora daramatza modu jarrai batean garatua izaten. Hori dela eta, eskaintzen dituen aukerak oso ugariak dira, esate baterako, *e-commerce* delakoa inplementatzea, erabiltzaile-kudeatzaile arteko harremana sustatzea... Hala ere, gure Eduki Kudeatzailea garatzerako orduan eskaintzen dituen funtzionalitate horietatik guztietatik minimo bat soilik erabiltzen da. Izan ere, Eduki Kudeatzailea ez da diseinatua izan webgune arrunt batek egiten dituen gauza guztiak egiteko, edukia *Android* aplikaziotik datozen erabiltzaileei erakusteko baizik, kudeatzaileek eduki hori maneiatzeko gaitasuna izateaz gain. EKS sistemak berez dituen gaitasunak murriztu ez direnez etorkizun batean sistemak eskaintzen dituen beste gaitasun batzuk erabiltzeko aukera dago, noski.

Datu-basea

WordPress erabilia egindako webgune orok MySQL datu-base bat erabiltzen du eduki eta ezarpenen iraunkortasuna bermatzeko. Datu-basearen egitura irudikatzen duen diagrama [23] bat erakusten da hurrengo orrialdean.

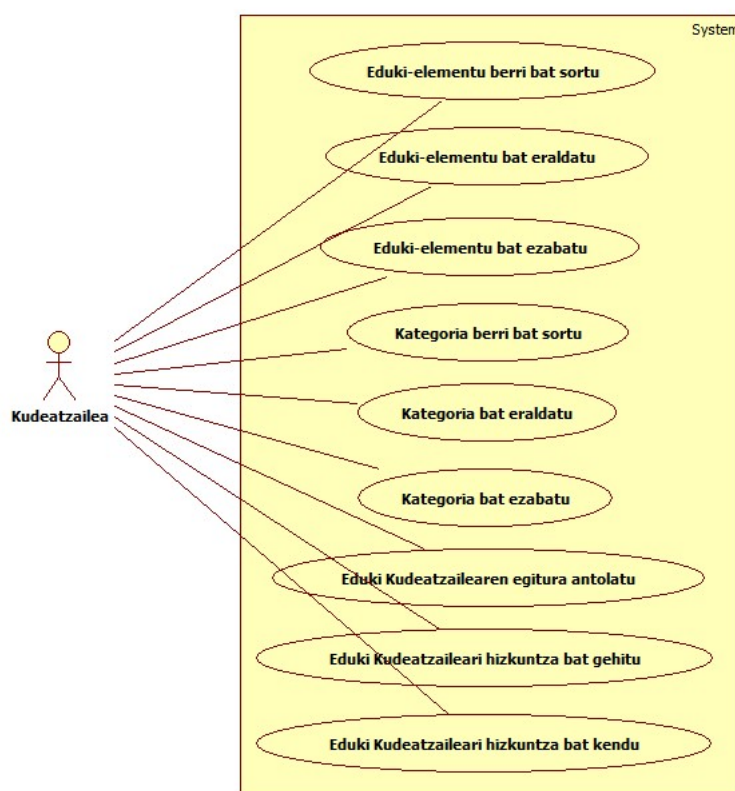
Hurrengo orrialdean agertzen den datu basea da *WordPress*-en edozein osagaik datuak gordetzeko erabiltzen duen azpiegitura nagusia. *WordPress* plataformarako zenbait gehigarrik datu-basearen diagrama eraldatu dezakete, taula berriak sortuz edota taula bati zutabe berriak gehituz. Hala ere, Eduki Kudeatzaile honek erabiltzen duen datu-basearen egitura *WordPress* plataformak era lehenetsian duena da. Datu-baseak dituen taula aipagarrienak *wp_posts*, *wp_users* eta *wp_options* dira. Lehenengo taula Eduki Kudeatzailean dauden Bidalketa guztiak gordetzeko erabiltzen da, bakoitzaren egilea, testua eta argitaratze-data gordetzen dituelarik, besteak beste. Bigarren taulak, berriz, Eduki Kudeatzailean kontu bat daukaten erabiltzaile guztien informazioa (erabiltzaile-izena, pasahitza, posta-kontua ...) gordetzen du. Azkenik, *wp_options* taula Eduki Kudeatzailearen ezarpenen balioak gordetzeko erabiltzen da.



3.10 Irudia: WordPress EKS-aren datu-basearen diagrama.

Erabilpen-kasuak

Esanda bezala proiektuan zehar garatutako Eduki Kudeatzaileak ez ditu *WordPress* EKS-aren funtzionalitate guztiak erabiltzen. Horretaz gain, Eduki Kudeatzaileari *qTranslate* izeneko gehigarria gehitu zaio, eleanitza izan dadin. Jarraian datorren diagraman ikus daiteke Eduki Kudeatzaile hau garatzeko aprobetxatu diren *WordPress*-en berezko erabilpen-kasuak, horietako batzuk *qTranslate*-ren gaitasunekin konbinatuta daudelarik.



3.11 Irudia: Eduki Kudeatzailean erabiltzen diren *WordPress* EKS-aren berezkoak diren erabilpen-kasuen diagrama.

Erabilpen-kasuen diagraman agertzen diren erabilpen-kasuak aktore bakar batek egiten ditu, Kudeatzaileak, hain zuzen ere. Kudeatzailea diogunean Eduki Kudeatzailearen egitura eta edukiak maneiatu eta eraldatzen dituen pertsona orori buruz ari gara. Hori egiteko, noski, pertsona horrek Eduki Kudeatzailearen kudeaketa-panelerako sarbidea eduki beharko du, bere erabiltzaile-kontuaren kredentzialak erabilita. Kudeatzaileak egin ditzakeen erabilpen-kasuak modu labur batean azalduko ditugu orain. Erabilpen-kasuen diagraman eduki-elementua aipatzen da. Eduki-elementua bi elementu ezberdinen (Bidalaketak eta Orrialdeak) orokortze bat da, bi elementu horietarako definitu diren erabilpen-

kasuak oso antzekoak dira eta. Ondorioz, bien erabilpen-kasuak bateratu egin dira prozesu berdina behin eta berriz azaldu behar ez izateko.

- Erabilpen-kasua: Eduki-elementua berri bat sortu

Testuingurua: *WordPress*-ek eduki-elementuak erabiltzen ditu artikulu gisako informazio gehiena erakusteko. Batetik, Bidalketak blog bateko sarrerak bezala izateko pentsatuta daude, baina ez da zertan portaera hori jarraitu. Bestetik, Orrialdeak aldaketa askorik gabe mantenduko den informazioa erakusteko erabiltzen dira, orrialde estatikoak adibidez. Eduki Kudeatzaile honen kasuan eduki-elementuak izango dira *Ohar Eleanitzak* aplikazioaren bidez atzitu direnak.

Aktorea: Kudeatzailea.

Aurrealdintzak:

- Eduki Kudeatzailean saioa hasita izatea.

Fluxu normala:

1. Kudeatzaileak, kudeaketa-panelean kokatuta dagoelarik, sortu nahi duen eduki-elementu motaren (Bidalketa zein Orrialde) elementu berri bat sortzeari emango dio.
2. Sistemak eduki-elementu berri bat sortzeko orrialdea erakutsiko dio.
3. Kudeatzaileak eduki-elementua osatuko du nahi duen testuarekin eta Eduki Kudeatzailean ezarrita daudenetatik nahi bezain beste hizkuntzatan. Horretaz gain, Bidalketen kasuan, nahi izatekotan Kategoriaren batetara ere lotuko du. Horren ondoren, gordetzeari emango dio.
4. Sistemak eduki-elementu hori gorde eta Eduki Kudeatzailean argitaratuko du.

- Erabilpen-kasua: Eduki-elementu bat eraldatu

Testuingurua: Eduki-elementu bat sortua izan denean Eduki Kudeatzailearen edozein bisitarik ikus dezake. Hala ere, eduki-elementua argitaratuta egon arren baliteke kudeatzaileak edukia eraldatu nahi izatea.

Aktorea: Kudeatzailea.

Aurrealdintzak:

- Eduki Kudeatzailean saioa hasita izatea.

Fluxu normala:

1. Kudeatzaileak, kudeaketa-panelean kokatuta dagoelarik, eraldatu nahi duen eduki-elementu motako elementuen zerrenda ikusteari emango dio.
2. Sistemak Eduki Kudeatzailean dauden mota horretako elementu guztien zerrenda erakutsiko du.
3. Kudeatzaileak eraldatu nahi duen eduki-elementua irekitzeari emango dio.
4. Sistemak eduki-elementu hori editorean irekiko du.
5. Kudeatzaileak eduki-elementuaren edukia eraldatuko du, lehendik zegoen zerbait ezabatuz, lehengoari zerbait gehituz, hizkuntz berri batean idatziz, Kategoria eraldatuz. . . Ondoren, gordetzeari emango dio.
6. Sistemak eduki-elementua eguneratuko du.

- Erabilpen-kasua: Eduki-elementu bat ezabatu

Testuingurua: Eduki-elementu bat sortua izan denean Eduki Kudeatzailearen edozein bisitarik ikus dezake. Hala ere, eduki-elementua argitaratuta egon arren baliteke kudeatzaileak ezabatu nahi izatea.

Aktorea: Kudeatzailea.

Aurrealdintzak:

- Eduki Kudeatzailean saioa hasita izatea.

Fluxu normala:

1. Kudeatzaileak, kudeaketa-panelean kokatuta dagoelarik, ezabatu nahi duen eduki-elementu motako elementuen zerrenda ikusteari emango dio.
2. Sistemak Eduki Kudeatzailean dauden mota horretarako eduki-elementu guztien zerrenda erakutsiko du.
3. Kudeatzaileak ezabatu nahi duen eduki-elementua ezabatzeari emango dio.
4. Sistemak eduki-elementu hori ezabatu egingo du eta hortik aurrera ez da atzigarri egongo.

- Erabilpen-kasua: Kategoria berri bat sortu

Testuingurua: *WordPress*-en bidez egindako Eduki Kudeatzailearen edukiak antolatzeke Kategoria berriak erabil daitezke. Kategorien bidez hainbat Bidalketa multzokatu

daitezke Eduki Kudeatzailearen bisitariak batera ikus ditzaten. Kategoriak beste Kategoria batzuen barruan ere sar daitezke hierarkia bat eratzeko.

Aktorea: Kudeatzailea.

Aurrebaldintzak:

- Eduki Kudeatzailean saioa hasita izatea.

Fluxu normala:

1. Kudeatzaileak, kudeaketa-panelean kokatuta dagoelarik, Kategoria berri bat sortzeari emango dio.
 2. Sistemak Kategoria berri bat sortzeko orrialdea erakutsiko dio.
 3. Kudeatzaileak Kategoriaren datuak beteko ditu eta ondoren Kategoria gordetzeari emango dio.
 4. Sistemak eduki-elementu hori gorde eta Eduki Kudeatzailean argitaratuko du.
- Erabilpen-kasua: Kategoria bat eraldatu

Testuingurua: Kategoria bat sortu den momentutik Bidalketak multzokatzeko erabili daiteke. Hala ere, gerora baliteke Kategoriaren informazioa aldatu nahi izatea (izena, gurasoa. . .).

Aktorea: Kudeatzailea.

Aurrebaldintzak:

- Eduki Kudeatzailean saioa hasita izatea.

Fluxu normala:

1. Kudeatzaileak, kudeaketa-panelean kokatuta dagoelarik, Kategorien zerrenda ikusteari emango dio.
2. Sistemak Eduki Kudeatzailean dauden Kategoria guztien zerrenda erakutsiko du.
3. Kudeatzaileak eraldatu nahi duen Kategoria irekitzeari emango dio.
4. Sistemak Kategoria hori editorean irekiko du.
5. Kudeatzaileak Kategoriaren edukia eraldatuko du eta ondoren, gordetzeari emango dio.

6. Sistemak Kategoria eguneratuko du.

- Erabilpen-kasua: Kategoria bat ezabatu

Testuingurua: Kategoriak Bidalketak antolatzeko erabiltzen dira baina Eduki Kudeatzailearen egitura aldaketa bat ematen bada baliteke Kategoria egitura berrian beharrezkoa ez izatea.

Aktorea: Kudeatzailea.

Aurrebaldintzak:

- Eduki Kudeatzailean saioa hasita izatea.

Fluxu normala:

1. Kudeatzaileak, kudeaketa-panelean kokatuta dagoelarik, Kategorien zerrenda ikusteari emango dio.
2. Sistemak Eduki Kudeatzailean dauden Kategoria guztien zerrenda erakutsiko du.
3. Kudeatzaileak ezabatu nahi duen Kategoria ezabatzeari emango dio.
4. Sistemak Kategoria hori ezabatu egingo du eta hortik aurrera ez da erabilgarri egongo.

- Erabilpen-kasua: Eduki Kudeatzailearen egitura antolatu

Testuingurua: Eduki Kudeatzailearen edukia modu ulergarri eta antolatu batean egon dadin *WordPress*-ek egitura pertsonalizatzeko aukera ematen du, Kategoria eta Orrialdeen bitartez. Lehenengoak, lotura duten Bidalketak multzo berdinean antolatzeko erabiltzen dira eta bigarrenak, berriz, webgunearen orrialde estatikoak (sarrera orria, honi buruz atala...) egiteko. Gero, menuen bitartez aurreko bi elementu motak modu hierarkiko batean antolatu daitezke, menu hori webgunearen leku ezberdinetan (goiburutik hasita oinera) kokatzeko aukera emanaz.

Aktorea: Kudeatzailea.

Aurrebaldintzak:

- Eduki Kudeatzailean saioa hasita izatea.

Fluxu normala:

1. Kudeatzaileak Eduki Kudeatzailearen egitura zehazteko atala irekitzeari emango dio.
 2. Sistemak Eduki Kudeatzailearen egiturari dagokion atala irekiko du, menua sortzeko aukera ematen duena.
 3. Kudeatzaileak menua sortzeari ekingo dio, bere edukiak (Orrialde eta Kategoriak) modu hierarkiko batean antolatuz. Ondoren, menu horri Eduki Kudeatzailean emango zaion kokapena zehaztuko du eta gordetzeari eman.
 4. Sistemak menu hori gordeko du eta momentu horretatik aurrera menua definitutako tokian agertuko da.
- Erabilpen-kasua: Eduki Kudeatzaileari hizkuntza bat gehitu

Testuingurua: *qTranslate* gehigarriaren bidez Eduki Kudeatzailea eleanitza izatea ahalbidetzen da. *qTranslate*-rekin Eduki Kudeatzaileari edozein hizkuntza gehitu dakioke.

Aktorea: Kudeatzailea.

Aurrebaldintzak:

- Eduki Kudeatzailean saioa hasita izatea.

Fluxu normala:

1. Kudeatzaileak, kudeaketa-panelean kokatuta dagoelarik, Eduki Kudeatzailearen hizkuntzak zehazten diren atalari emango dio.
2. Sistemak atal hori irekiko du, dagoeneko ezarrita daudenak erakutsiz.
3. Kudeatzaileak gehitu nahi duen hizkuntza definituta badago (gaituta egon ez arren) zuzenean gaitzeari emango dio.
4. Sistemak hizkuntz berria gaituko du eta bisitariek zein kudeatzaileek ikusi ahalko dute.

Ez-ohiko fluxua:

3. Hizkuntza definituta ez badago kudeatzaileak gehitu nahi duen hizkuntzaren datuak idatziko ditu, zerbitzarira hizkuntzaren .mo fitxategiak igotzearekin batera, eta hizkuntz hori sortzeari emango dio.

4. Sistemak hizkuntz hori sortuko du eta definitutako hizkuntzekin batera agertuko da.
5. Kudeatzaileak hizkuntza gaitzeari emango dio.
6. Sistemak hizkuntz berria gaituko du eta bisitariak zein kudeatzaileek ikusi ahalko dute.

- Erabilpen-kasua: Eduki Kudeatzaileari hizkuntza bat kendu

Testuingurua: Aurreko erabilpen-kasuen aipatu den bezala *qTranslate* gehigarriarekin hizkuntz berriak gehi dakizkioke Eduki Kudeatzaileari, baina, era berean, dagoeneko ezarrita dagoen hizkuntza bat ezabatzeko aukera ere ematen du.

Aktorea: Kudeatzailea.

Aurrebaldintzak:

- Eduki Kudeatzailean saioa hasita izatea.

Fluxu normala:

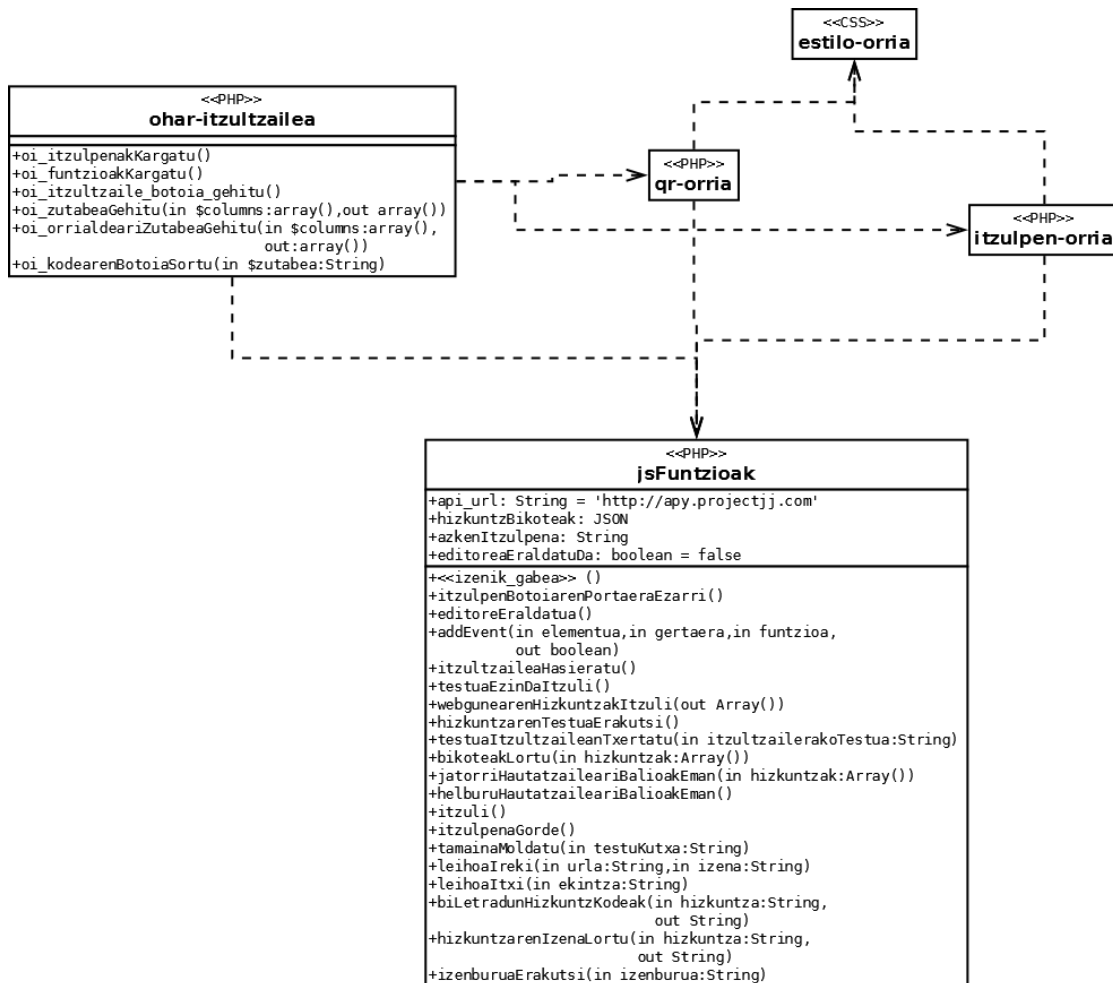
1. Kudeatzaileak, kudeaketa-panelean kokatuta dagoelarik, Eduki Kudeatzailearen hizkuntzak zehazten diren atalari emango dio.
2. Sistemak atal hori irekiko du, dagoeneko ezarrita daudenak erakutsiz.
3. Kudeatzaileak kendu nahi duen hizkuntza ezgaitzeari emango dio.
4. Sistemak hizkuntz berria ezgaituko du eta bisitariak zein kudeatzaileek ezingo dute ikusi (definituta egoten jarraituko duen arren).

Ohar Itzultzailea gehigarria

Proiektuan garatutako Eduki Kudeatzailearako garatu den *Ohar Itzultzailea* izeneko gehigarriak Eduki Kudeatzailearen baitan dauden edukiak itzultzeko aukera eskaintzen dio kudeatzaileei, baita eduki horien QR kodeak sortzeko, gero Ohar Eleanitzak aplikazioaren bitartez irakurri daitezzen.

Gehigarriaren egitura

Hurrengo orrialdean agertzen den irudian erakusten da gehigarriak duen egitura.



3.12 Irudia: *Ohar Itzultzailea* gehigarriaren egitura adierazten duen irudia.

Goiko irudian agertzen den elementu bakoitza fitxategi ezberdin bat da. Fitxategi guztiak (estilo-orria izan ezik) PHP lengoaian daude idatzita, osoki ez bada ere. Gehigarriaren funtzionalitateak martxan jartzen dituen osagaia *ohar-itzultzailea* da. Horren bidez *WordPress*-ek dituen kakoekin lotura egiten da. Kakoak (ekintzak zein iragazkiak), *WordPress*-erako gehigarrien garapena azaltzen zen atalean esanda bezala, *WordPress* webgune baten gertaeretan gure koda txertatu eta exekutatzeko ahalbidetzen duten exekuzio-puntu batzuk dira, esate baterako, gehigarriak kargatzen bukatu den unea, edukia kargatu den unea...

Horretaz gain, *itzulpen-orria* eta *qr-orria* osagaiak erabiltzaileak erabiliko duen interfazea definitzeko erabiltzen dira. *JsFuntzioak* osagaia berriz beste osagaiak erabiliko dituzten JavaScript funtzioak definitzeko erabiltzen da. Azkenik, *estilo-orria* interfazearen itxura zehazten ditu. Jarraian duzue osagai bakoitzaren deskribapen zehatzago bat:

ohar-itzultzailea: Gehigarriaren osagai nagusia, beste funtzionalitateak martxan jarri ahal izateko beharrezkoa dena. Hauek dira definituta dituen funtzioak:

- *oi_itzulpenakKargatu()*: *plugins_loaded* ekintzari lotzen zaion funtzioa, gehigarri guztiak kargatu direnean exekutatu dena. Funtzio honek gehigarriaren eleaniztasuna ahalbidetzen du, gehigarria Eduki Kudeatzailearen hizkuntzan jarriaz eskuragarri egotekotan. Horretarako hizkuntz horretarako .mo fitxategiak definituta egon beharko dira.
- *oi_funtzioakKargatu()*: *init* ekintzari lotzen zaion funtzioa, orrialdea hasieratzen denean exekutatu dena. Funtzio honen bidez gehigarriak itzulpenetarako eta QR kodeen sorkuntzarako behar dituen *JavaScript* funtzioak kargatzen ditu, funtzio horiek exekutatzeke behar dituzten liburutegiak kargatzeaz gain. Bestetik, gehigarriak sortzen dituen orrialdeek erabiltzen duten estilo-orria ere kargatu egiten du.
- *oi_itzultzaileBotoiaGehitu()*: *media_buttons* ekintzari lotzen zaion funtzioa, edukien editorearen multimediamiako botoiak kargatzen direnean exekutatu dena. Funtzio honek editoreari botoi berri bat gehituko dio, multimedia botoiak egoten diren eremuan, itzultzailea martxan jartzen duena.
- *oi_zutabeaGehitu(\$columns)*: *manage_posts_columns* iragazkiari lotzen zaion funtzioa, Bidalketa guztiak erakusten diren orrialdea kargatzean exekutatu dena. Bidalketa guztiak erakusten diren orrialdean Bidalketa bakoitzaren informazioa (izenburua, egilea...) erakusten duen taula itxurako egitura bat dago. Funtzio honek taulari zutabe berri bat gehitzen dio, Bidalketa bakoitzaren QR kodeak sortzeko erabiliko dena.
- *oi_orrialdeariZutabeaGehitu(\$columns)*: Funtzio hau *oi_zutabeaGehitu* funtzioaren oso antzekoa da. *manage_pages_columns* iragazkiari lotzen zaion funtzioa, Orrialde guztiak erakusten diren orrialdea kargatzean exekutatu da. Orrialde guztiak erakusten diren orrialdean Orrialde bakoitzaren informazioa (izenburua, egilea...) erakusten duen taula itxurako egitura bat dago. Funtzio honek taulari zutabe berri bat gehitzen dio, Orrialde bakoitzaren QR kodeak sortzeko erabiliko dena.
- *oi_kodearenBotoiaSortu(\$zutabea)*: *manage_posts_custom_column* eta *manage_pages_custom_column* ekintzei lotzen zaion funtzioa, Bidalketa edota Orrialde guztiak erakusten diren orrialdea kargatzean exekutatu dena. Funtzio hau elementu (Bidalketa eta Orrialde) horien informazioa erakusten duen taularen zutabeak kargatu ondoren exekutatu da, eta sortu berri dugun zutabearen errenkada

bakoitzari errenkada horretako elementuaren QR kodea sortzeko botoi bat gehituko dio.

jsFuntzioak: Osagai honetan gainontzeko osagaiak erabiliko dituzten JavaScript-eko funtzioak definitzen dira. Hauek dira definitzen dituen funtzioak:

- *()*: Funtzio honen bidez edukiaren editorea kargatzen denean itzultzailea abiarazten duen botoiak izan behar duen portaera definitzen duen funtzioari deitzen zaio.
- *itzultzailearenPortaeraEzarri()*: Funtzio honek itzultzailearen portaera zehazten du. Izan ere, erabiltzaileak edukiaren editoreak dituen fitxen artean nabigatzean itzultzailea ez da guztiz funtzionala (garapenari buruz diharduen atalean azalduko da portaera hori).
- *editoreEraldatua()*: Funtzio honen bidez erabiltzaileak editoreko fitxaz aldatu duen kontrolatzen da.
- *addEvent(elementua, gertaera, funtzioa)*: Funtzio honekin HTML objektuei gertaera jakin batzuei erantzuteko funtzioak esleitzen zaizkie.
- *itzultzaileaHasieratu()*: Funtzio hau itzultzailea hasieratzen denean behar diren datuak eskuratzeaz arduratzen da, hala nola, Apertium web-zerbitzuak eskaintzen dituen hizkuntz-bikoteak.
- *testuaEzinDaItzuli()*: Eragiketa honek erabiltzaileari jakinarazten dio Bidalketa edota Orrialdearen testua ezin dela itzuli, editoreko fitxen artean nabigatu duelako. Horren ordez izenburua itzultzeko aukera ere ematen du.
- *webgunearenHizkuntzakItzuli()*: Funtzio honek Eduki Kudeatzailean gaituta dauden hizkuntzak lortzen ditu.
- *hizkuntzarenTestuaErakutsi()*: Itzultzailearen interfazeak bi hautatzaile dauzka itzulpenaren jatorri hizkuntza eta helburu hizkuntza erabakitzeko. Jatorri hizkuntzaren hautatzaileak eskaintzen dituen balioak Eduki Kudeatzaileak gaituta dituen hizkuntzak dira, web-zerbitzuak hizkuntz hori eskaintzen badu. Jatorri hizkuntzaren hautatzailean hizkuntz bat hautatzean funtzio hau exekutatu da editoretik hizkuntza horren testua lortu eta jatorri hizkuntzaren testu-kutxan testu hori erakusten duen funtzioari deitzeko. Testu hori testu-kutxan gorde ostean testuaren luzeraren araberrako tamaina emango dio testu-kutxari.

- *testuItzultzaileanTxertatu(itzultzaileakoTestua)*: Funtzio honek edukiaren editorean dagoen testutik jatorri hizkuntzaren hautatzailean hautatuta dagoen hizkuntzaren testua erauzi eta jatorri hizkuntzaren testu-kutxan erakusten du.
- *bikoteakLortu(hizkuntzak)*: Funtzio hau *Apertium* web-zerbitzuari deia egiteaz arduratzen da, eskaintzen dituen hizkuntz-bikoteak lortzeko. Lortzen duenean aldagai batean gordeko du erantzuna, web-zerbitzuari berriro ere berdina galdetu behar ez izateko. Azkenik, informazio hori kudeatzeaz arduratzen den *jatorriHautatzaileariBalioakEman* funtzioari deituko dio.
- *jatorriHautatzaileariBalioakEman(hizkuntzak)*: Funtzio honek *Apertium*-etik jasotako hizkuntz-bikoteak prozesatu egingo ditu eta jatorrizko hizkuntzaren hautatzailea beteko du bikote horien jatorri hizkuntzekin. Hizkuntz bat hautatzailean gehitu aurretik funtzio honek jasotzen duen *hizkuntzak* bektorearekin alderatzen da, Eduki Kudeatzailean gaituta dauden hizkuntzak gordetzen dituena, Eduki Kudeatzailea hizkuntz horretan ere ezarrita al dagoen egiaztatzeko. Hautatzailearen portaera ere definitzen da, balio berri bat klikatzen den bakoitzean helburu hizkuntzaren hautatzaileari balioak emango dizkion *helburuHautatzaileariBalioakEman* eragiketari deitu diezaion.
- *helburuHautatzaileariBalioakEman()*: Funtzio honek helburu hizkuntzaren hautatzailea bete egiten du *Apertium* web-zerbitzuak dituen hizkuntz-bikoteen helburu hizkuntzekin. Hala ere, hori egin aurretik egiaztatu egiten du ea helburu hizkuntza hori Eduki Kudeatzailean gaituta al dagoen. Funtzio honetan jatorri hizkuntzaren testu-kutxaren balioa eguneratu egiten da, funtzio hau deitua izateak jatorri hizkuntz berri bat hautatu dela esan nahi baitu.
- *itzuli()*: Funtzio honek itzultzailean itzulpena egiteaz arduratzen den botoiari portaera ematen dio. Botoia sakatua denean *Apertium* web-zerbitzuari itzulpena egiteko eskatu eta helburu hizkuntzaren testu-kutxan itzulpena idatziko du, testu-kutxari itzulpenaren luzeraren arabera tamaina emanez.
- *itzulpenaGorde()*: Funtzio hau egindako itzulpena editorean gordetzeaz arduratuko da, itzulpena gordetzeaz arduratzen den botoia sakatua denean. Itzulpena Bidalketa edota Orrialdearen testuarena edota izenburuarena izan daitekeenez, horren arabera gordeko da behar den tokian.
- *tamainaMoldatu(testuKutxa)*: Funtzio honek itzultzaileak dauzkan bi testu-kutxen

tamaina moldatzen du irakurgarriagoa izan dadin, barruan duten testuaren arabera. Jasotzen duen aldagaiak moldatu beharreko testu-kutxa adierazten du.

- *leioaIreki(urla,izena)*: Funtzio honek leiho berri bat irekiko du url eta leihoaren identifikaziorako erabiliko den izen bat jasota. Funtzio hau itzultzaile edota QR sortailearen leihoa irekitzeko erabiltzen da.
- *leioaItxi(ekintza)*: Funtzio honek itzultzailearentzat edota QR sortailearentzat irekitako leihoa itxiko du. Jasotzen duen aldagaiaren arabera portaera ezberdina izango du: zuzenean itxiko du edo itxi aurretik itzulpena gordetzeaz arduratuko da (itzulpena gorde eta leihoa ixteaz arduratzen den botoia sakatua izan bada).
- *biLetradunHizkuntzKodeak(hizkuntza)*: *Apertium* web-zerbitzuak hiru letrako hizkuntz-kodeak (ISO-639-2/639-5) erabiltzen ditu baina *WordPress* eta *qTranslate*-k bi letrakoak (ISO-639-1) erabiltzen dituzte. Funtzio honek sarrera gisa hiru letrako hizkuntz-kodea jasoko du eta horren baliokidea den bi letratakoa itzuliko du. Hizkuntz guztiek ez dute bi letratako koderik, beraz, kasu horietan ez da koderik itzuliko. Beste batzuetan, hizkuntz baten aldaerak (en_US adibidez, hau da, AEBetako ingelesaren aldaera) jasotzen ditu. Hizkuntz horien “estandarra” itzultzaileak ere eskaintzen duenez, aldaeraren bat jasotzekotan ez du koderik itzuliko.
- *hizkuntzarenIzenaLortu(hizkuntza)*: Funtzio honen bidez hiru letrako hizkuntz-kode baten izena itzultzen da, adibidez, *eu* jasota *Euskara* itzuliko luke.
- *izenburuaErakutsi(izenburua)*: Funtzio hau QR sortailean erabiltzen da QR kodearen gainean Bidalketa edota Orrialdearen izenburua erakusteko, eskuragarri dagoen hizkuntz ezberdin guztietan.

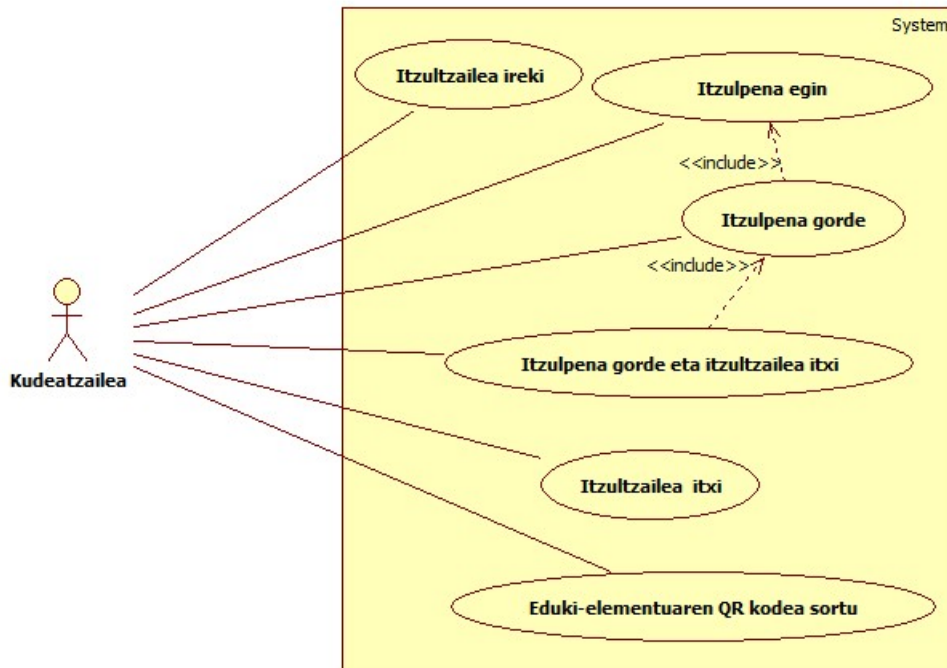
itzulpen-orria: Osagai honek erabiltzaileak Bidalketa edota Orrialdeen edukia itzultzeko erabiliko duen interfazea definitzen du. Osagai honek editoretik itzuli beharreko testua atzitu eta itzuli ondoren editorean txertatu egiten du itzulpena.

qr-orria: Osagai honek erabiltzaileak Bidalketa edota Orrialdeen QR kodea sortzeko erabiliko duen interfazea definitzen du.

estilo-orria: Osagai honek gehigarriak erabiltzailearekin elkarrekintza burutzeko erabiltzen dituen bi interfazeen (itzulpen-orria eta qr-orria) itxura definitzen du.

Erabilpen-kasuak

Hurrengo orrialdean agertzen den irudian *Ohar Itzultzailea* gehigarriak dituen erabilpen-kasuak ikus daitezke.



3.13 Irudia: *Ohar Itzultzailea* gehigarriaren erabilpen-kasuen diagrama.

WordPress-ek dituen berezko erabilpen-kasuak deskribatzen zituen irudian bezala, *Ohar Itzultzailea* gehigarriak dituen erabilpen-kasuak aktore gisa Kudeatzailea dute. Kudeatzailea Eduki Kudeatzailearen kudeaketa-panelerako sarbidea duen eta Eduki Kudeatzailearen edukia eraldatzeko eskumena duen pertsona oro litzateke. Kudeatzaileak gehigarriaren bidez egin ditzakeen ekintzak, beraz, hauek dira: *Itzultzailea ireki*, *Itzulpena egin*, *Itzulpena gorde*, *Itzulpena gorde eta itzultzailea itxi*, *Itzultzailea itxi* eta *Eduki-elementuaren QR kodea sortu*. Eduki-elementua esaten denean Orrialde edota Bidalketari erreferentzia egin diezaioke. Izan ere, bi elementu motak berdinak ez diren arren gehigarriaren ikuspuntutik berdinak dira, bi kasuetan urrats berdinak egin behar dira eta ekintza berdina burutzeko.

- Erabilpen-kasua: *Itzultzailea ireki*

Testuingurua: Lehenago aipatu bezala gehigarri honek *WordPress*-eko eduki-elementuak

itzultzeko aukera ematen du. Horretarako, itzuli nahi den eduki-elementuaren editorerara joan behar da eta bertan itzultzailea irekitzeko gehitu den botoia sakatu.

Aktorea: Kudeatzailea.

Aurrebaldintzak:

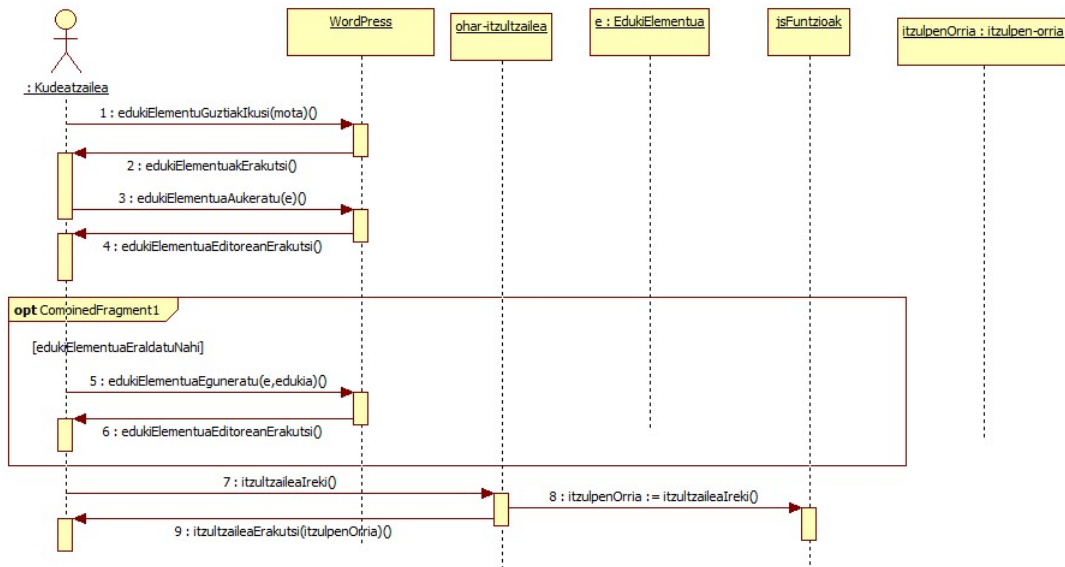
- Eduki Kudeatzailean saioa hasita izatea.

Fluxu normala:

1. Kudeatzaileak, kudeaketa-panelaren sarreran dagoela, itzuli nahi dituen eduki-elementu motako elementuak ikusteari emango dio.
2. Sistemak Kudeatzaileari Eduki Kudeatzailean dauden mota horretako eduki-elementuak erakutsiko dizkio.
3. Kudeatzaileak itzuli nahi duen eduki-elementua aukeratuko du (eduki-elementua berri bat ere sortu dezake).
4. Sistemak aukeratutako eduki-elementuaren editorea irekiko du.
5. Kudeatzaileak eduki-elementuaren orain arteko testua eraldatu nahi ez badu zuzenean itzultzailea irekitzeari emango dio.
6. Sistemak itzultzailea leiho berri batean irekiko du.

Ez-ohiko fluxua:

5. Kudeatzaileak eduki-elementuaren edukia eraldatu nahi badu eraldaketa egingo du eta horren ondoren eduki-elementua eguneratzeari emango dio.
6. Sistemak eduki-elementua eguneratuko du.
7. Kudeatzaileak itzultzailea irekitzen duen botoia sakatuko du.
8. Sistemak itzultzailea leiho berri batean irekiko du.

Sekuentzia-diagrama:**3.14 Irudia:** Itzultzailea ireki erabilpen-kasuaren sekuentzia-diagrama.

- Erabilpen-kasua: Itzulpena egin

Testuingurua: Itzultzaileak eduki-elementuaren editorean dagoen edukia atzitzen du itzulpenak egiteko. Itzulpenak egiteko bi aukera daude: eduki-elementuaren izenburua itzultzea edota eduki-elementuaren gorputza itzultzea. Eduki hori hizkuntz ezberdinetan dagoenez jatorri hizkuntzaren hautatzailean hautatuta dagoen hizkuntzaren testua jarriko du irakurgarri jatorri hizkuntzaren testu-kutxan. *Apertium* web-zerbitzuak jatorri hizkuntza horrekiko dituen itzulpen aukeren arabera balioak izango ditu helburu hizkuntzaren hautatzaileak, beti ere Eduki Kudeatzailea hizkuntza horietan gaituta badago. Izenburuaren itzulpena modu egokian gauzatzen da baina kasu batzuetan (editoreko fitxen artean nabigatu denean, adibidez) eduki-elementuaren gorputza ezin izaten da itzuli.

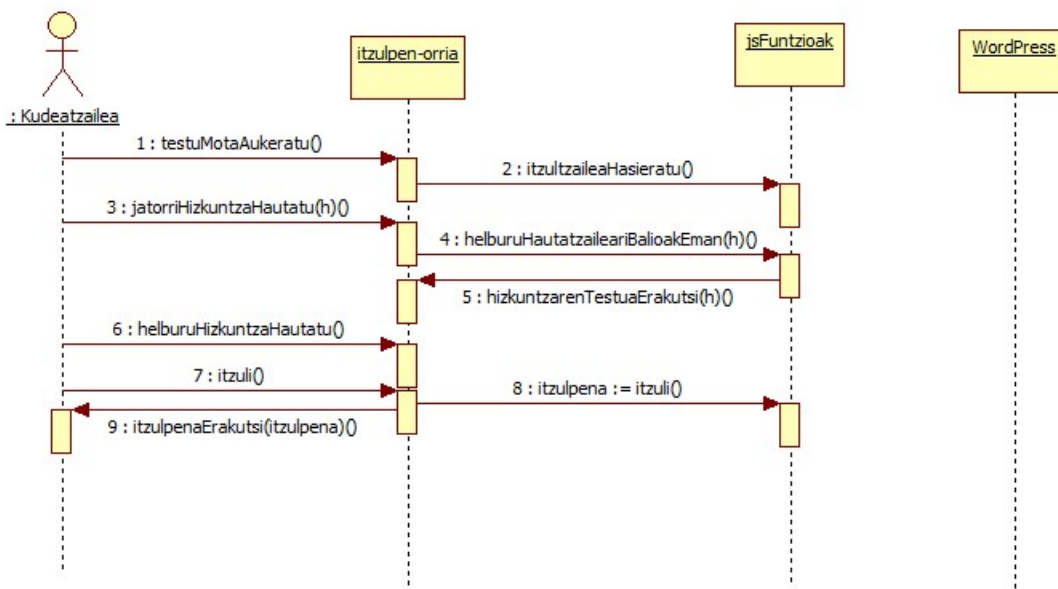
Aktorea: Kudeatzailea.

Aurrebaldintzak:

- Eduki Kudeatzailean saioa hasita izatea.
- Itzultzailea irekita egotea.

Fluxu normala:

1. Kudeatzaileak itzuli nahi duen testu mota (izenburua edota eduki-elementuaren gorputza) aukeratu du.
2. Kudeatzaileak gorputza itzultzea aukeratzen badu eta sistemak detektatzen badu editoreko fitxen arteko nabigazioa gertatu dela, kudeatzaileari jakinaraziko dio unean ezin dela hori burutu, arazoa konpontzeko argibideak emanez. Izenburuak itzuli ditzakeela oroitaraziko dio.
3. Kudeatzaileak aurrera jarraitu nahi badu erabili nahi duen jatorri hizkuntza hautatuko du.
4. Sistemak jatorri hizkuntza horren testua erakutsiko du.
5. Kudeatzaileak testu hori eraldatu ahalko du nahi izatekotan eta gero itzulpen-erako helburu hizkuntza hautatuko du. Ondoren, itzulpena martxan jartzen duen botoia sakatuko du.
6. Sistemak itzulpena egingo du *Apertium* web-zerbitzua erabilia eta erabiltzaileari erakutsiko dio.

Sekuentzia-diagrama:

3.15 Irudia: Itzulpena egin erabilpen-kasuaren sekuentzia-diagrama.

- Erabilpen-kasua: Itzulpena gorde

Testuingurua: Itzulpena egin denean itzultzaileak aukera ematen du itzulpen hori modu automatiko batean eduki-elementuaren editorean txertatzeko.

Aktorea: Kudeatzailea.

Aurrebaldintzak:

- Eduki Kudeatzailean saioa hasita izatea.
- Itzultzailea irekita egotea.

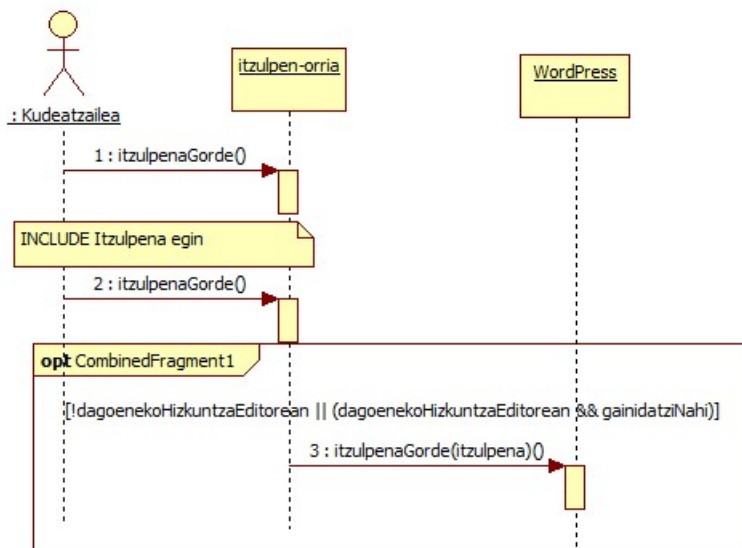
Fluxu normala:

1. Kudeatzaileak *Itzulpena egin* erabilpen-kasua burutu ondoren itzulpena gordetzeari emango dio.
2. Sistema itzulpen hori gordetzen saiatuko da, itzulpen horren helburu hizkuntzan. Dagoeneko editorean hizkuntz horri dagokion testua badago Kudeatzaileari galdetuko dio ea testu hori gainidatzi nahi al duen. Kudeatzaileak baiezkoa erantzuten badu sistemak aurreko testuaren gainean idatziko du.

Ez-ohiko fluxua:

2. Kudeatzaileak ezezkoa erantzuten badu itzulpena ez da gordeko.

Sekuentzia-diagrama:



3.16 Irudia: Itzulpena gorde erabilpen-kasuaren sekuentzia-diagrama.

- Erabilpen-kasua: Itzulpena gorde eta itzultzailea itxi

Testuingurua: Aurreko erabilpen-kasuak itzulpena editorean automatikoki gordetzen zuen, horren ondoren beste itzulpen batzuk egitea ahalbidetuz. Erabilpen-kasu honen bidez, berriz, itzulpena gorde ondoren leihoa itxi egingo da.

Aktorea: Kudeatzailea.

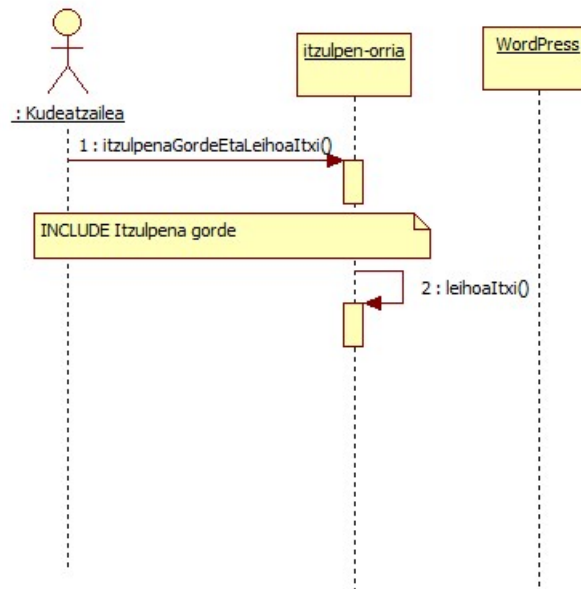
Aurrealdintzak:

- Eduki Kudeatzailean saioa hasita izatea.
- Itzultzailea irekita egotea.

Fluxu normala:

1. Kudeatzaileak *Itzulpena egin* erabilpen-kasua burutu ondoren itzulpena gorde eta leihoa ixteari emango dio.
2. Horren ondorioz, *Itzulpena gorde* erabilpen-kasua burutuko da. Bukatzen duenean sistemak itzultzailearen leihoa itxi egingo du.

Sekuentzia-diagrama:



3.17 Irudia: Itzulpena gorde eta itzultzailea itxi erabilpen-kasuaren sekuentzia-diagrama.

- Erabilpen-kasua: Itzultzailea itxi

Testuingurua: Erabilpen kasu honen bidez itzultzailea ixteko aukera dago, exekuzioaren edozein unetan.

Aktorea: Kudeatzailea.

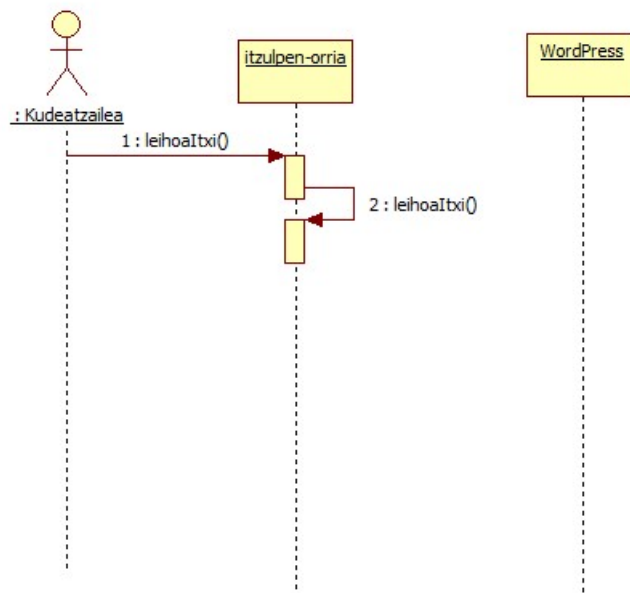
Aurrebaldintzak:

- Eduki Kudeatzailean saioa hasita izatea.
- Itzultzailea irekita egotea.

Fluxu normala:

1. Kudeatzaileak itzultzailea ixteari emango dio.
2. Sistemak itzultzailea itxi egingo du.

Sekuentzia-diagrama:



3.18 Irudia: Itzultzailea itxi erabilpen-kasuaren sekuentzia-diagrama.

- Erabilpen-kasua: Eduki-elementuaren QR kodea sortu

Testuingurua: Gehigarriak Kudeatzaileari aukera ematen dio Eduki Kudeatzaileko eduki-elementuei dagokien QR kodeak sortzeko. Horretarako *QRicket* web-zerbitzua erabiltzen da.

Aktorea: Kudeatzailea.

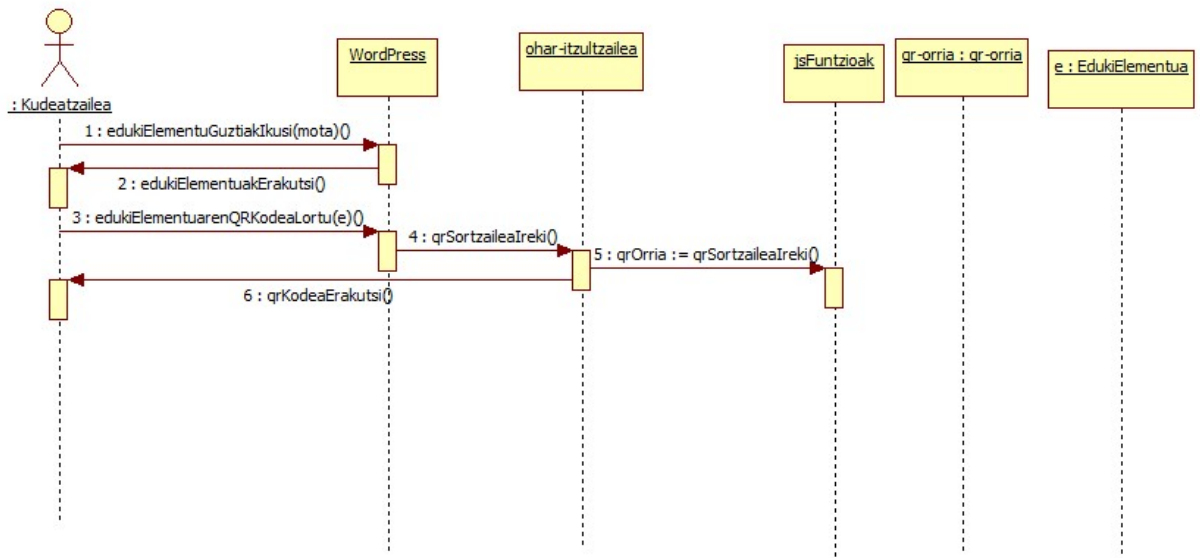
Aurrebaldintzak:

- Eduki Kudeatzailean saioa hasita izatea.

Fluxu normala:

1. Kudeatzaileak, kudeaketa-panelaren sarreran dagoela, ikusi nahi duen eduki-elementu motako elementu guztiak ikusteari emango dio.
2. Sistemak Kudeatzaileari Eduki Kudeatzailean dauden mota horretako eduki-elementuak erakutsiko dizkio.
3. Kudeatzaileak eduki-elementua bat aukeratuko du QR kodea sortzeko.
4. Sistemak leiho berri bat irekiko du eduki-elementua horren QR kodea erakusteko.

Sekuentzia-diagrama:



3.19 Irudia: Eduki-elementuaren QR kodea sortu erabilpen-kasuaren sekuentzia-diagrama.

4. KAPITULUA

Garapena eta Probak

Atal honetan produktuaren garapen eta probei buruz hitz egingo dugu. Garapena eta proba faseak normalean batera gertatzen diren arren bakoitza bere aldetik azalduko dugu.

4.1 Garapena

Azpiatal honetan proiektuaren garapenari buruzko zenbait xehetasun ematen dira, esate baterako, garatu den produktuaren ezaugarri teknikoak, erabili den internazionalizazio estrategia, interfaze grafikoa. . . Garatutako produktuaren erabileraren berri jakin nahi izatekotan *Erabilera gida* eranskinean dituzue xehetasunak.

4.1.1 Ezaugarri Teknikoak

Ohar Eleanitzak aplikazioa

Ezaugarria	Balioa
Aplikazioaren izena	Ohar Eleanitzak
Aplikazioaren bertsioa	1.0
Hizkuntzak	Euskara Ingelera Gaztelania Frantsesa Katalana
Jasaten duen <i>Android</i> bertsio minimoa	2.3 <i>Gingerbread</i>
Jasaten duen <i>Android</i> bertsio maximoa	4.4 <i>KitKat</i>
Erabiltzen dituen kanpo liburutegiak	<i>ZBar bar code reader 0.1</i>
Garatzeko erabilitako lengoiaia	Java, SQL, XML
Behar duen biltegitratze gaitasun minimoa	5.67 MB

4.1 Taula: Ohar Eleanitzak aplikazioaren ezaugarri teknikoak laburtzen dituen taula.

Eduki Kudeatzailea

Ezaugarria	Balioa
Helbidea	http://ohareleanitzak.esy.es/
<i>WordPress</i>	
<i>WordPress</i> -en bertsioa	3.8.3
Hizkuntzak	Euskara Ingelera Gaztelania Frantsesa Katalana
Erabiltzen dituen gehigarriak	<i>qTranslate 2.5.39</i> <i>ohar-itzultzailea 1.0</i>
Erabiltzen duen txantiloia	<i>Tiny Forge 1.5.6</i>
Erabiltzen duen hosting zerbitzua	http://www.hostinger.co.uk/
<i>ohar-itzultzailea</i> gehigarria	
Gehigarriaren bertsioa	1.0
Erabiltzen dituen web-zerbitzuak	<i>Apertium</i> <i>QRickit</i>
Itzulgarria al da?	Bai
Zein hizkuntzatan dago eskuragarri?	Euskara Ingelera
Garapenean erabilitako lengoaiak	PHP, JavaScript, HTML, CSS, JSONP, JQuery

4.2 Taula: Eduki Kudeatzailearen ezaugarri teknikoak laburtzen dituen taula.

4.1.2 Internazionalizazio estrategia

Internazionalizazio estrategia esatean produktua hizkuntz ezberdinetan egon dadin eman beharreko pausoak esan nahi da. Beste era batera esanda, zertan oinarritzen den eta zein aldagai erabiltzen diren aldaketa hori aurrera eramateko.

Ohar Eleanitzak aplikazioa

Diseinuaren atalean azaldu den bezala aplikazioak aukera menuaren baitan aplikazioaren hizkuntza hautatzeko aukera ematen du. Horren bidez, *Android* aplikazioek duten lehentasitako portaera eraldatzen da, aplikazioan zure mugikorraren hizkuntzaren ezberdina den hizkuntz bat hautatzea ahalbidetuz. Aplikazioan aukeratuta dagoen hizkuntz hori izango da QR kode bat irakurtzen denean kontutan hartuko dena.

Eduki Kudeatzailean hizkuntza aurre-helbide bezala adierazten denez, irakurritako helbidean trataera egingo beharko da. Hori egiteko aplikazioak irakurritako kodeak gordetzen duen helbidean Eduki Kudeatzailearen erroa bilatuko du, ohareleanitzak.esy.es, alegia. Erroa bukatzen den indizea lortuta mugikorra ezarrita dagoen hizkuntzaren bi le-tretako kodea (ISO 639-1 estandarrari dagokiona) txertatuko du erroaren eta erroaren os-tean dagoen helbide zatiaren artean. Horrela, helbidea prest egongo da aplikazioa ezarrita dagoen hizkuntzan irekia izateko.

Eduki Kudeatzailea

Eduki Kudeatzaileari dagokionez, eduki berdinen hizkuntz bertsio ezberdinak atzitze-ko aurre-helbidearen estrategia erabiltzen da. Aurre-helbideak zera esan nahi du, Eduki Kudeatzailearen erroaren (Eduki Kudeatzailearen osagai ezberdin guztiek duten zati ko-muna) eta osagai bakoitzaren bereizgarri den helbidearen artean egongo dela hizkuntzaren bereizgarria. Bereizmen hori, aurreko atalean ikusi den bezala (*Ohar Eleanitzak* aplika-zioaren internazionalizazio estrategia) hizkuntzaren ISO 639-1 kodearen bidez egiten da.

Jarraian azaltzen da Eduki Kudeatzailearen internazionalizazio prozesua kasu prak-tiko batekin. Horretarako, Josep M. Sert margolariak marraztutako Elezahar-herria mihi-seari dagokion Eduki Kudeatzaileko Bidalketa erabiliko dugu. Adibiderako hizkuntza gisa euskara erabiliko dugu.

- **Helbidea:** *ohareleanitzak.esy.es/arte-ederrak/elezahar-herria/*
 - Erroa: *ohareleanitzak.esy.es/*
 - Osagaiaren helbidea: *arte-ederrak/elezahar-herria/*
 - Hizkuntz-kodea: *eu*
 - Helbidearen euskarazko bertsioa:
ohareleanitzak.esy.es/eu/arte-ederrak/elezahar-herria/

4.1.3 Interfaze grafikoa

Azpiatal honetan produktuaren interfaze grafikoaren nondik-norakoak azalduko dira.

Ohar Eleanitzak aplikazioa

Interfaze grafikoari dagokionez, aplikazioak berezko lau pantaila dituela esan daiteke. Batetik, jarduera nagusiak duen interfazea dago (*KameraJarduera* jarduerarena, alegia), eta bestetik, irakurritako kodeen historia ikusteko (*IrakurketaHistoria* jarduerarena), ezarpenak aldatzeko (*Ezarpenak* jarduerarena) eta hizkuntza aldatzeko (*HizkuntzHautatzailea* jarduerarena) erabiltzen diren interfazeak. Jarduera nagusiak duen interfazea erabiltzailiak ikusiko duen lehenengoa izango da eta bertan kamerak jasotzen dituen irudiak erakutsiko dira, modu jarrai batean.

Horretaz gain, interfaze nagusi honetan ezarpenetara, hizkuntza aldatzera edota historiara joateko aukera egongo da aukeren barran *Aukerak*, *Hizkuntza* eta *Historia* botoiak sakatuz gero, hurrenez hurren. Ezarpenak aldatzeko erabiltzen den interfazeak nahiko funtzionalitate murrizta du, *Birbideratzea Automatikoki Onartu* ezarpenaren balioak aldatzeko soilik erabiliko baita. Ezarpen honetan balioaren araberako mezu bat agertzen da. Erabiltzaileri askatasuna ematen zaio nahi duen bezala erabiltzeko. Hala ere, ezarpena gaituta badago, hau da, *attaging* delakoa gertatzeko aukera gehiago dagoen egoera, erabiltzaileri funtzionamendua azaldu eta jakinaraziko zaio hori ez dela gomendatutako ezarpena, segurtasuna arrazoiak direla medio. Ezgaituta dagoenean, berriz, erabiltzaileri funtzionamendua azalduko zaio, ezarpenarentzako balio gomendatua ezarrita daukala esanez.

Bestetik, hizkuntza aldatzeko erabiltzen den interfazean egin daitekeen gauza bakarra aplikazioaren hizkuntza aldatzea da. Horretarako, *Spinner* edo hautatzaile bat dago,

aplikazioa eskuragarri dagoen hizkuntza ezberdin guztiak erakusten dituena. Azkenik, irakurketa historia erakusten duen interfazean bi ekintza ezberdin egin daitezke. Lehenik eta behin, irakurketa historia kontsultatzeko aukera dago eta bertan agertzen den elementuetako bat ere atzitu daiteke, bere gainean sakatuz gero. Horretaz gain, erabiltzailearen irakurketa historia garbitzeko aukera dago.

Eduki Kudeatzailea

WordPress webgunea

Diseinuaren atalean esan dugun bezala *WordPress* Edukiak Kudeatzeko Sistemak eskaintzen dituen aukera guztietatik gutxi batzuk erabiliko ditugu. Hori kontutan hartuta, Kudeatzailearen ikuspuntutik sei interfaze nagusi daudela esan daiteke: hasierako orria, Bidalketena, hizkuntzena, Kategoriena, Orrialdeena eta webgunearen antolamenduarena.

Lehenik eta behin, hasierako orrian guneari buruzko hainbat xehetasun agertzen dira. Bertan, Eduki Kudeatzaileak dauzkan Bidalketa kopurua, instalatzeko dauden eguneraketak edota antzeko beste ohar batzuk ikus daitezke. Horretaz gain, Orrialde honetatik atzituko dira Eduki Kudeatzailea kudeatzeko dauden aukera guztiak: Bidalketak kudeatu, hizkuntzak kudeatu...

Bidalketen eta Orrialdeen kudeaketa eremuak oso antzekoak dira. Bietan, Eduki Kudeatzailean dauden Bidalketa/Orrialdeen zerrenda (hautatutako aukeraren arabera) ikus daiteke, horiek ezabatu edota eraldatzeko aukerarekin. Zerrenda horretako elementu bakoitzeko elementu horren QR kodea sortzen duen botoi bat dago, *Ohar Itzultzailea* gehigarriaren bidez gehitzen dena. Horretaz gain, Bidalketa/Orrialde berri bat sortzeko aukera ere badago.

Bestetik, Kategoriak kudeatzeko orrialdean Eduki Kudeatzailean sortuta dauden Kategorien zerrenda bat erakusten da, Kategoria berriak sortzeko aukera emateaz gain. Bestalde, hizkuntzak kudeatzeko orrialdean unean gaituta dauden hizkuntzak ageri dira. Era berean, definituta dauden hizkuntzak ere erakusten ditu, gaituta egon ez arren, horietako edozein gaitu edota ezgaitzeko aukerarekin. Horretaz gain, hizkuntz berri bat definitzeko aukera ere badauka.

Azkenik, webgunearen egitura kudeatzeko erabiltzen den atalean webgunea antolatzeko erabiltzen diren menuak sortu eta antolatzeko aukera dago. Antolaketaren kasuan, adibidez, menua osatuko duten elementuak aukeratu eta menu hierarkiko batean kokatze-

ko aukera dago. Bestalde, menuak webgunean izango duen kokapena zehazteko aukera ere badago.

Webgunea *Ohar Eleanitzak* aplikazioaren bidez edota erabiltzaile soil bat bezala bisitatzekotan ez dira hainbeste aukera egongo bisitariaren eskura. Beste edozein webgunea arruntetan bezala, atzigarri dauden edukien artean nabigatzeko askatasun osoa izango du erabiltzaileak eta erakusten den edukiaren hizkuntza aukeratzeko gaitasuna ere izango du.

Ohar Itzultzailea gehigarria

Ohar Itzultzailea gehigarriak *WordPress* erabilia sortutako webguneari bi funtzionalitate gehitzen dizkio: itzulpenak egiteko gaitasuna eta Bidalketen edota Orrialdeen QR kodea sortzearena. Funtzionalitate berri bakoitzak interfaze bat gehitzen dio Eduki Kudeatzailerari. Interfaze horiek irekitzeko bi toki ezberdinetara joan beharra dago. Itzultzailea ireki nahi izatekotan, Bidalketa edota Orrialdeen editorera joan beharko da Kudeatzailea. QR sortzailea ireki nahi izatekotan, berriz, Bidalketen edota Orrialdeen zerrenda erakusten den orrialdera joan beharko da Kudeatzailea. Editoretik atzitu daitekeen itzultzaileak testua edota izenburuak itzultzeko gaitasuna dauka. Bestetik, QR sortzailea Bidalketen eta Orrialdeen QR kodea sortzeko erabiliko da.

4.2 Probak

Atal honetan garatutako produktuaren proba prozesua nola egin den azalduko da: probetarako erabilitako baliabideak, garapen zikloak...

4.2.1 Probetarako erabilitako baliabideak

Azpiatal honetan produktuaren probak egiteko erabili diren gailu eta softwarearen aipamen bat egiten da.

Ohar Eleanitzak aplikazioa

Android aplikazioaren probak egiteko bi bide ezberdin jarraitu dira. Hasteko, proba gehienak egiteko *Android* Sistema Eragilea duten gailu fisikoak erabili dira baina kasu jakin batzuetan *Eclipse* Garapen Ingurune Integratuak daukan emuladorea ere erabili da, esate

baterako, dimentsio ezberdina duten pantailetan interfazea nola ikusiko litzatekeen jakiteko.

Gailu fisikoa erabili denean *Android 2.3.4 Gingerbread* bertsio daukan *Sony Ericsson Xperia Mini Pro* telefono mugikorra erabili da, 3 hazbeteko pantaila daukana. Aplikazioaren garapen prozesuaren bukaeran *Android 4.3 Jelly Bean* bertsioa daukan *Samsung Galaxy S II* eta *Android 4.4 KitKat* bertsio duen *Nexus 5* gailua ere erabili dira, 4.3 hazbeteko eta 5 hazbeteko pantaila darabiltena, hurrenez hurren.

Eclipse IDEa erabili izan denean, berriz, *Android 4.2.2 Jelly Bean* bertsioa duten 7 eta 10 hazbeteko alegiazko gailuak erabili dira.

Eduki Kudeatzailea

Eduki Kudeatzailearekin probak egitean hainbat arakatzaille ezberdin erabili dira, gailu mugikorretakoak batetik eta ordenagailurako direnak bestetik. Multzo bakoitzak egin beharreko probak ezberdinak dira, gailu mugikorretatik atzitzen denean lehentasun osoa izango du edukiak ondo ikusteak, hau da, kudeatzaile lanak egiteko erraztasuna ez da kontutan hartu, hori *WordPress* plataformaren esku dagoen zerbait da, beraz, proiektuaren helburuetatik kanpo dagoen zerbait. Ordenagailutik Eduki Kudeatzailea atzitzean garrantzi handiagoa izango du kudeatzaile lanak egiteko erraztasunak, batez ere *Ohar Itzultzailea* gehigarriak eskaintzen dituen funtzionalitateenak. Hala ere, ordenagailutik webgunearen edukia atzitzean daukan erabilgarritasuna ere kontutan hartuko da. Probak egiteko erabili diren arakatzailleak jarraian aipatzen dira.

- Gailu mugikorretan:
 - *Google Chrome* 35.0.1916.141.
 - *Android* gailuekin berez datorren arakatzaillea.
- Ordenagailuan:
 - *Mozilla Firefox* 27tik 30 bertsiora.
 - *Google Chrome* 35.0.1916.141.

4.2.2 Garapen zikloak

Produktuaren osagai ezberdinen garapena burutzeko garapenaren ikuspegi mailakatuaren ideia jarraitu da. Beste era batera esanda, osagai bakoitzaren garapena jarritako helburuen

lorpen konstante batean oinarritu da, bukaerako produktua hasieratik lortzeko asmorik gabe. Hori horrela, osagai bakoitzaren funtzionalitate nagusiak hobetsi ziren garapenerako lehen pauso bezala eta hori lortu ondoren funtzionalitate nagusi hori oinarritzat hartuta horren inguruan eraikitzeari ekin zitzaion. Zikloka eman zen garapen prozesu horretan laguntzeko *Git* [34] bertsio kontrolerako softwarea baliatu zen.

Ohar Eleanitzak aplikazioa

Android aplikazioaren garapen zikloak zerrendatzen dira jarraian:

1. Aplikazioaren QR kodeak irakurtzeko funtzionalitatea landu zen. Une honetan aplikazioa gai zen web-helbide bat barruan gordetzen zuen edozein QR kode irakurtzeko.
2. *Attaging*-az babesteko neurri bezala *Birbideratzea Automatikoki Onartu* ezarpena gehitu zitzaion aplikazioari. Ezarpen hori gaituta dagoenean aplikazioak erabiltzaileak irakurritako kodeen web-helbidea zuzenean atzitu du. Ezgaituta dagoenean, ordea, aplikazioak erabiltzaileari baimena eskatuko dio helbidea irakurtzeko, erabiltzaileari irakurtzen denaren gaineko kontrol handiagoa emanez.
3. Aplikazioaren hizkuntza aldatzeko aukera garatu zen, mugikorren hizkuntza dektatuz. Horrela, aplikazioa mugikorra ezarrita dagoen hizkuntzan ezarriko da, hizkuntza horretan eskuragarri badago. Horrekin batera, aplikazioa hizkuntza horretan ez egotekotan erabili beharreko lehenetsitako hizkuntza (ingeleza) ere zehaztu zen. Bestetik, hizkuntza hori aprobetxatuta *ohareleanitzak.esy.es* webgunera zihoazen kode guztietan edukia aplikazioaren hizkuntzan atzitzeko gai zen. Era berean, aplikazioari euskara, ingeleza eta gaztelaniazko testuak gehitu zitzaizkion. Azkenik, jarduera nagusiaren interfazearen txukunketa egin zen.
4. Erabiltzailearen irakurketa historia gordetzen zuen funtzionalitatea garatu zen, irakurritako kodeen zerrenda bat erakusten zuena. Zerrenda hori kronologikoki antolatuta zegoen, kode berrienak zerrendaren hasieran agertzen zirelarik.
5. Irakurketa historia modu antolatuago batean erakusteko asmoz, irakurritako kodeak irakurketa dataren arabera antolatu ziren, multzo ezberdinetan banatuz: Gaur, Atzo, Azken astea... Karaktere-kateen eguneraketa egin zen, aplikazioaren beharretara moldatzeko.

6. Aplikazioari mugikorraren hizkuntza ez den hizkuntza bat aukeratzeko funtzionalitatea gehitu zitzaion. Funtzionalitate honekin *Android* aplikazioen lehenetsitako portaera aldatu zen, erabiltzaileari aukera emanez aplikazioaren hizkuntza hauta-tzaile batekin aukeratzeko. Horretaz gain, aplikazioaren karaktere-kateak eguneratu eta katalan eta frantsesezko testuak gehitu zitzaizkion.

Eduki Kudeatzailea

WordPress webgunea Jarraian zerrendatzen dira *ohareleanitzak.esy.es* helbidean dagoen webgunea garatzeko erabilitako garapen-zikloak:

1. Webgunea zerbitzarian kokatu eta webguneari itxura eman zitzaion txantilo bat gehituz. Webgunea pantaila txikia duten gailuetara bideratuta zegoenez informazio funtsezkoa soilik erakusten zuen webgunea diseinatu zen. Horrekin batera, *qTranslate* gehigarria instalatu zen eta webgunea euskaraz, ingelesez eta gaztelaniaz ezarri zen.
2. Webgunearen edukia modu antolatu batean erakusteko asmoz, Kategoria eta Orrialdeetan oinarritzen zen egitura gehitu zitzaion webguneari. Era berean, goiburua azpian kokatzen zen menua gehitu zitzaion, Kategoriak eta Orrialdeak erakusten zituena.
3. Webgunearentzako Bidalketak sortu ziren sarean atzigarri zeuden museo batzuen edukiak eta Donostiako Informatika Fakultateari dagozkion zenbait informazio aprobetxatuz. Horretaz gain, Pasai San Pedron kokatuta dagoen Ondartxo museoarentzat egin ziren proben ondorioz museo horri dagozkion hainbat Bidalketa gehitu ziren. Bestalde, webguneari sarrera Orrialde bat eta webgunearen helburuen berri ematen zuen beste Orrialde bat gehitu zitzaizkion. Aipatu berri dugun eduki berri guztia Kategoria eta Orrialde ezberdinak erabiliz antolatu zen.
4. Webguneari katalan eta frantses hizkuntzak gehitu zitzaizkion, *Ohar Itzultzailea* gehigarriaren garapena bukatzearekin batera, itzulpen aukera gehiago izateko.

Ohar Itzultzailea gehigarria Jarraian aipatzen dira *WordPress*-erako *Ohar Itzultzailea* gehigarriaren garapen-zikloak:

1. Gehigarriaren itzulpenak egiteko gaitasuna garatu zen, *Apertium* zerbitzuarekiko beharrezkoa zen komunikazioa implementatuz. Une honetan gehigarria ez zen gai

Bidalketen edukia automatikoki atzitu edo gordetzeko, itzulpenak egiteaz bakarrik arduratzen zen. Era berean, gehigarria hizkuntz ezberdinetara modu erraz batean itzulgarria izan zedin gehigarriaren lokalizazioarentzako oinarria landu zen. Lokalizazioa hurrengo garapen zikloetan ere landu zen.

2. Bidalketen edukia atzitu eta gordetzeko aukera inplementatu zen. Horrekin batera, itzultzaileak itzulpenetarako eskaini zitzakeen aukerak mugatu ziren, webgunean ezarrita zeuden hizkuntzen arteko itzulpenak egiteko aukera soilik eman zezan.
3. Gehigarria Bidalketen edukia itzultzeaz gain Orrialdeen edukia itzultzeko gai zen. Horretaz gain, Bidalketa eta Orrialdeen testua itzultzeaz gain izenburuak itzultzeko aukera gehitu zitzaion.
4. *QRickit* zerbitzua erabilia Bidalketa eta Orrialdeen QR kodeak lortzeko aukera inplementatu zen.
5. Itzultzaileak dituen portaera okerrak mugatzeko prebentzio neurriak hartu ziren, erabiltzaileari akatsa gertatu aurretik abisatuz egoera horretan ez dela prozesua ondo garatuko. Horretaz gain, garapen prozesuan egindako lokalizazio lana aprobe-txatuz gehigarria ingelesera ere itzuli zen.

4.2.3 Produktuaren portaera ezegokiak

Azpiatal honetan produktuak dituen zenbait portaera ezegoki azalduko ditugu.

Eduki Kudeatzailearen atzitzea

Eduki Kudeatzailea Interneten atzigarri egoteko *hosting* zerbitzu bat erabili da, *Hostinger* izena duena. Zerbitzu horrek plan ezberdinak dauzka, doakoak eta ordaintzekoak. Proiektu honetarako doako zerbitzua hautatu zen, garapenaren kostu ekonomikoa ahalik eta baxuen mantentzeko asmoz. Erabaki hori hartzean jakitun ginen doako *hosting* zerbitzu bat hautatzeak zeukan berme urriaz. Garapenaren hasieran doako zerbitzu bat erabiltzeak sortzen duen ziurgabetasuna eduki arren zerbitzuak ez zuen arazo nabarmenik eman. Noizean behin zerbitzaria ez zegoen atzigarri baina denbora labur bat itxaron ondoren berriro atzigarri zegoen, bestelako eraginik gabe.

Hala ere, Eduki Kudeatzailean egin behar zen garapena bukatu zenean beste ataza batzuetan lan egin beharrak Eduki Kudeatzailea maiztasun txikiagoarekin atzitzea suposatu

zuen. Hortik aurrera, Eduki Kudeatzailea atzitzerako orduan arazoak maizago gertatzen zirela nabaritu zen. Batzuk aipatzearren, Eduki Kudeatzailea atzitzeko itxaron denbora luzeagoak, abiaduraren moteltzea, baita zerbitzariko eduki guztien ezabatzea ere. Azken kasu hau konpontzeak ez zuen galerarik ekarri eduki horren babes-kopia bat zegoelako eta ondorioz denbora gutxian berrezarri zelako bere egoera arruntera. Dena dela, gertakari horrek zerbitzariaren ikuskaritza handiagoa suposatu zuen, berriro gertatzekotan azkar erantzun ahal izateko.

Itzultzailearen portaera

Lehenago esan den bezala, itzultzaileak ez du beti eraginkortasunez funtzionatzen. Hobeto esanda, itzultzaileak itzulpenak ondo egiten ditu beti baina kasu batzuetan itzulpen hori ez da editorean gordetzen. Lehenik eta behin, komenigarria da aipatzea funtzionamendu okerra Bidalketa/Orrialdearen testua itzultzean soilik gertatzen dela, ez izenburua itzultzean. Funtzionamendu ezegokia hiru egoera ezberdinetan gertatzen da: itzultzailea unean gehitutako edukia gorde gabe dagoela irekitzean, itzultzailea ireki aurretik editoreak daukan fitxen (hizkuntz bakoitzarentzat dagoena zein editorearen estiloa ezartzeko erabiltzen direnak) artean nabigatzean eta itzultzailea editorea *Visual* aukeran dagoela irekitzean. Zoritxarrez, ez da arazo horiek modu automatiko batean konpontzeko biderik topatu.

Hala ere, portaera ezegokia ekiditeko egin beharrekoa ez da oso zaila. Hasteko, Bidalketa/Orrialdeari aldaketa bat egiten zaionean ondo funtziona dezan beharrekoa da eguneratzeari ematea itzulpenik egin nahi izatekotan. Horri jarraituz, edukiaren sorrerarako erabiltzen diren editoreko fitxen artean nabigatu ondoren ere beharrekoa da bidalketa eguneratzea. Neurri batean aurreko bi soluzioak lotuta daude, editoreko fitxak hein handi batean Bidalketa/Orrialdearen edukia eraldatzeko erabiltzen dira eta. Fitxen artean nabigatu ostean itzultzailea zuzenean irekitzeari ematen bazaio Kudeatzaileari ohar bat erakutsiko zaio, testua itzuli ezingo duela esanez eta izenburuak itzultzea iradokiz. Oharrean testua itzultzeko egin beharrekoa (Bidalketa/Orrialdea eguneratzeari eman) ere azalduko zaio. Azkenik, ez da mekanismorik topatu editorea *Visual* aukeran ote dagoen kontrolatzeko. Era lehenetsian editorea beste aukeran dago, *Text* aukeran, alegia, baina orrialdea eguneratzeari ematean editorea *Visual* aukeran badago birkargatzean ere horrela egongo da. Horretaz jakinarazteko itzulpena gordetzeari ematen zaionean ohar bat ematen da, *Visual* aukera ez hautatzeko aholkatuz.

4.2.4 Ondartxo museorako probak

Produktuaren garapen fasea bukatzeaz zegoela Pasai San Pedron eraikitzen ari ziren Ondartxo museoaren arduradunen aplikazioarekiko interesaren berri izan genuen. Ondartxo museoak Kanadako uretan hondoratu zen San Juan izeneko baleontziaren inguruko tematika dauka. Baleontzia 1565. urtean hondoratu zen arren 1978. urtean baleontzia topatzean kontserbazio egoera onean topatu zuten. Ondartxo museoan, baleontzi horren aztarnak ardatz hartuta baleontzia eraikitzeari ekin diote garai hartako teknikak erabiliz, Donostia 2016 ekimenaren parte bezala.

Museoa ekainaren 25ean irekitzekoa zenez aplikazioarekin proba batzuk egin ziren, Eduki Kudeatzailean museoan erabili behar zituzten testu batzuk sartuaz. Testu horiek lau hizkuntz ezberdinetan zeuden (euskara, ingelesa, gaztelania eta frantsesa) eta museoan egingo zen balizko implementazio baten lehen proba bezala egin zen. Museoaren arduradunek beste zenbait eskaera ere bazituzten aplikazioak izan beharreko funtzionalitateen inguruan baina inaugurazio eguna eta proiektuaren entrega egunaren arteko gertutasuna zela eta funtzionalitate horiek proiektua bukatu osterako uztea erabaki zen. Azkenean, museoaren inaugurazio egunerako egin beharreko gauza guztiak direla eta aplikazioa museora etorkizun hurbil batean gehitzea hobetsi zen, prozesua arrakastaz burutuko zela ziurtatzeko.

5. KAPITULUA

Ondorioak eta Etorkizuneko Aukerak

Atal honetan proiektuan zehar ateratako ondorioen berri emango dugu, baita etorkizuneko aurreikusten diren aukerei buruz ere.

5.1 Ondorioak

Jarraian aipatzen dira proiektuan zehar egindako lanaren bidez atera diren ondorioak:

- Lehenik eta behin, esan beharra dago proiektuaren hasieran ezarri ziren helburuak bete egin direla. Batetik, garatutako produktuak aukera eskaintzen du toki mugatu batean eduki eleanitza eskaintzeko, QR kodeak erabiltzeari esker. Bestetik, garatutako produktua gai da sarean eskuragarri dauden zerbitzuak erabiliz eduki eleanitza bat-batean sortzeko itzulpenen bidez. Azkenik, produktua kode irekikoak diren tresnak erabiliz garatu da, *Apertium* [30] itzulpenetarako, *QRicket* [17] QR kodeak sortzeko, *WordPress* [21] Eduki Kudeatzailea eraikitzeko, *Android* [3] gailu mugikorretarako aplikazioa garatzeko. . .
- Orokorrean, proiektuaren irismenean zehaztutakoa bete egin da. Ezarri ziren bete-kizunak bete egin dira garapenaren bukaeran. Mugarrietan ezarritako epeak ez dira guztiz bete, itzultzaile automatikoari dagokion mugarrria eta dokumentazioari dagokion epeetan atzerapenak egon ziren eta. Aldez aurretik utzitako denbora marjinari esker ez zuen bestelako eraginik izan.

- Produktuaren erabilera esparru bezala museoak izango zirela aurreikusi zen hasiera hasieratik eta garapenean ere ideia horrekin jardun da. Hala ere, garatutako produktuak dituen ezaugarriak kontutan hartuta esan daiteke museoetaz harago doan erabilera esparru bat duela. Izan ere, erabiltzaileak produktua erabiltzeko duen behar bakarra Internet konexio bat izatea da, irakurritako QR kodeak atzitu ahal izateko. Gaur egun, gailu mugikor gehienetan dago Internet konexioa, haririk gabeko sareen beharrik gabe. Hori horrela izanda, produktuaren eleaniztasunerako gaitasunak oso erabilgarriak izan daitezke beste esparru askotan. Produktuak izan ditzakeen beste erabilera esparru batzuk ondorengoak lirateke: hiri eta herrietako kartelen edukia zure hizkuntzan ikusteko, denda batean dauden produktuen deskribapen zehatza-goak ikusteko. . .
- Azkenik, proiektuaren garapenean doakoak diren zerbitzuak erabiltzearen ondorioak sufritu dira garapenean zehar. Batetik, *hosting*-arekin izandako arazoak pairatu ziren eta bestetik, *qTranslate* gehigarria eta *WordPress* Edukiak Kudeatzeko Sistemaren bertsio berrienaren arteko bateragarritasun eza gertatu zen. Zorionez, garapenean ez du eragin negatibo handirik izan baina kontutan hartu beharreko gauza da, zalantzarik gabe, batez ere produktua merkaturatzea edota produktua eguneroko bizitzan erabili nahi izatekotan.

5.2 Etorkizuneko Aukerak

Produktuak proiekturako ezarritako helburuak bete dituen arren oraindik ere garapen aukera asko dago produktua hobetu nahi izatekotan:

- **Ondartxo museoan produktua inplementatzea:** Etorkizun hurbilean gertatzeko probabilitate gehien duen aukera da. Honek museoako arduradunen zenbait betekizun osagarri inplementatzea ere suposatuko luke, tartean, Eduki Kudeatzailean museoaren edukia testuz agertzeaz gain ahoz ere esatea. Hala ere, Ondartxo museotik egindako betekizunak inplementatzeak produktua beste museo, denda batzuetara zabaltzea erraztuko luke.
- **Eduki Kudeatzailearen atzigarritasuna hobetu:** Doako *hosting* zerbitzu bat erabiltzeak, lehenago aipatu den bezala, eragina dauka Eduki Kudeatzailearen atzigarritasunean. Esparru hori hobetzeko pausorik ziurrena zerbitzuz aldatzea litzateke,

ordaindu beharreko beste zerbitzu batera pasatuz. Horrek ez luke ziurtatuko Eduki Kudeatzailearen atzigarritasuna une oro baina behintzat berme batzuk eskainiko lituzke.

- **Eduki Kudeatzaileak zein aplikazioak eskaintzen dituen hizkuntz kopurua hedatu:** Produktuaren izaeran atal garrantzitsu bat hartzen du eleaniztasunak eta horren presentzia handitzeko komenigarria litzateke hizkuntz gehiago gehitzea. Azken finean, Eduki Kudeatzaileari hizkuntz berri bat gehitzeko sarean *WordPress*-en hizkuntz horretako fitxategiak topatzearekin nahikoa da. *Ohar Itzultzailea* gehigarria eta *Ohar Eleanitzak* aplikazioa hizkuntz berrira gehitzeko, berriz, nahikoa da karaktere-kateak hizkuntz berri horretara itzultzearekin. Azkenik, Eduki Kudeatzailearen edukia hizkuntz berri batean eskaintzeko nahiko da gehigarriaren itzulpen funtzionalitateak erabiltzea.
- **Itzultzailearen itzulpen gaitasunak hedatu:** Apertium [30] itzultzaileak itzulpenetarako erabilgarri dauzkan hizkuntz-bikoteak mugatuak dira eta horietako zenbait ez dira produktuaren merkaturatze fase batean erabiltzeko modukoak [31], dituzten itzulpen mugak direla eta. Gainera, euskararekiko dituen itzulpen gaitasunak gutxi dira, euskaratik ingeles eta gaztelaniara itzul daiteke, baina ezin da euskarara itzulpenik egin. Hori oztopo bat izan daiteke euskara jakin ez eta euskarara itzuli nahiko luketenentzat. Aipatutako arazoei aurre egiteko, bi aukera daude: itzulpen gaitasunak hedatzea Apertium kode irekiko beste zerbitzu batzuekin konbinatuta (gaztelania eta euskararen arteko itzulpenak egiten dituen *Matxin* [29], adibidez), edo itzulpen aukera gehiago dituen itzultzaile bat (*Google Translate* [27], adibidez) erabiltzea. Lehenengo aukeraren bidez kode irekikoak diren zerbitzuak erabiltzearen filosofiaren jarraipen bat egingo litzateke (aipatutako eraginkortasun maila izanda ere). Bigarrena erabiltzeak ziurrenik produktuaren merkaturatze ahalmena hobetuko luke, itzulpen gaitasunak hedatzeaz gain itzulpen horien maila hobetuko luke eta.

Bibliografia

Produktuaren aurrekariak

- [1] *QRpedia* Wikipedian (Ingelesez)
<https://en.wikipedia.org/wiki/QRpedia>
- [2] *Word Lens* Wikipedian (Ingelesez)
https://en.wikipedia.org/wiki/Word_Lens

Android

- [3] *Android* Wikipedian (Ingelesez)
[https://en.wikipedia.org/wiki/Android_\(operating_system\)](https://en.wikipedia.org/wiki/Android_(operating_system))
- [4] *Android* Sistema Eragilearen ezaugarriak (Gaztelaniaz)
<http://www.androidcurso.com/index.php/tutoriales-android-fundamentos/31-unidad-1-vision-general-y-entorno-de-desarrollo/97-que-hace-a-android-especial>
- [5] *Android* Sistema Eragilearen arkitektura (Gaztelaniaz)
<http://www.androidcurso.com/index.php/tutoriales-android-fundamentos/31-unidad-1-vision-general-y-entorno-de-desarrollo/99-arquitectura-de-android>
- [6] *Android* jarduera baten bizitza-zikloa (Gaztelaniaz)
<http://www.androidcurso.com/index.php/tutoriales-android-fundamentos/37-unidad-6-multimedia-y-ciclo-de-vida/158-ciclo-de-vida-de-una-actividad>

Android-erako garapena

- [7] *Android Studio* (Ingelesez)
<https://developer.android.com/sdk/installing/studio.html>

- [8] *Eclipserako Androiden Software Development Kit-a(SDK)*

<https://developer.android.com/sdk/index.html>

- [9] *Eclipse* Wikipedian (Ingelese)

https://en.wikipedia.org/wiki/Eclipse_%28software%29

- [10] *Eclipserako ADT gehigarria*

<https://developer.android.com/tools/sdk/eclipse-adt.html>

- [11] *Android-erako programazioa ikasteko (Gaztelaniaz)*

<http://www.androidcurso.com/>

QR kodeak

- [12] QR kodeak Wikipedian (Ingelese)

https://en.wikipedia.org/wiki/QR_code

- [13] *Attaging-a* (Ingelese)

<http://rbach.net/blog/index.php/qr-codes-can-put-users-at-risk/>

QR kodeen erabilera

- [14] *ZBar bar code reader liburutegia*

<http://sourceforge.net/projects/zbar/>

- [15] *Zxing irakurlea*

<https://github.com/zxing/zxing>

- [16] *QR Code Generator web-zerbitzua*

<http://goqr.me/api/>

- [17] *QRicket web-zerbitzua*

http://qricket.com/qricket_apps/qricket_api.php

Hosting zerbitzuak

- [18] *000WebHost hosting zerbitzua*

<http://www.000webhost.com/>

- [19] *Hostinger hosting zerbitzua*

<http://www.hostinger.co.uk/>

Edukiak Kudeatzeko Sistemak

[20] EKS ezberdinen konparazioa (Ingelesez)

<https://www.udemy.com/blog/drupal-vs-joomla-vs-wordpress/>

WordPress

[21] *WordPress* Wikipedian (Ingelesez)

<https://en.wikipedia.org/wiki/WordPress>

[22] *WordPress* euskaraz

<http://eu.wordpress.org/>

[23] *WordPress* EKS-aren datu-basea

http://codex.wordpress.org/Database_Description

[24] *WordPress*-en internazionalizazio aukerak (Ingelesez)

http://codex.wordpress.org/Multilingual_WordPress

[25] *WordPress*-erako gehigarrien garapena (Ingelesez)

http://codex.wordpress.org/Writing_a_Plugin

[26] *WordPress*-erako *qTranslate* gehigarria

<http://www.qianqin.de/qtranslate/>

Itzultzaile automatikoak

[27] *Google Translate* itzultzailea

<https://developers.google.com/translate/>

[28] *Microsoft Translator* itzultzailea

<http://www.microsoft.com/en-us/translator/developers.aspx>

[29] *Matxin* itzultzailea

<http://ixa2.si.ehu.es/ixazerbitzuak/>

Apertium

[30] *Apertium* Wikipedian (Ingelesez)

<https://en.wikipedia.org/wiki/Apertium>

[31] *Apertium* itzultzailearen eraginkortasunaren datuak (Ingelesez)

http://wiki.apertium.org/wiki/Translation_quality_statistics

[32] *Apertium*-en hizkuntz bikoteak
<http://apy.projectjj.com/listPairs>

[33] *Apertium* web-zerbitzua
http://wiki.apertium.org/wiki/Apertium_web_service#JSONP_REST_API

Bestelakoak

[34] *Git* Wikipedian (Ingelesez)
[https://en.wikipedia.org/wiki/Git_\(software\)](https://en.wikipedia.org/wiki/Git_(software))

Eranskinak

A. ERANSKINA

Proiektuaren Kudeaketa

Eranskin honetan Proiektuaren Kudeaketari buruzko xehetasun gehiago aipatzen dira. *Sarrera eta Helburuak* atalean proiektuaren irismenari buruzko informazioa zegoen. Bada, proiektuaren hasieran egin zen plangintzako beste hainbat esparru aipatzeaz gain, egindako aurreikuspenen betetze mailaz jardungo gara orain.

A.1 Plangintza

Atal honetan, denboraren planifikazioa, lan metodologia, baliabideen esleipena, kalitate eta arriskuen planaz jardungo gara.

A.1.1 Denboraren planifikazioa

Atal honetan proiektuan zehar gauzatuko diren atazen zehaztapen handiagoa egiten da. Horretarako, atazei estimatzen zaien iraupena eta ataza horiek gauzatuko diren denbora-tartea definitzen dira.

Atazen denbora estimazioa

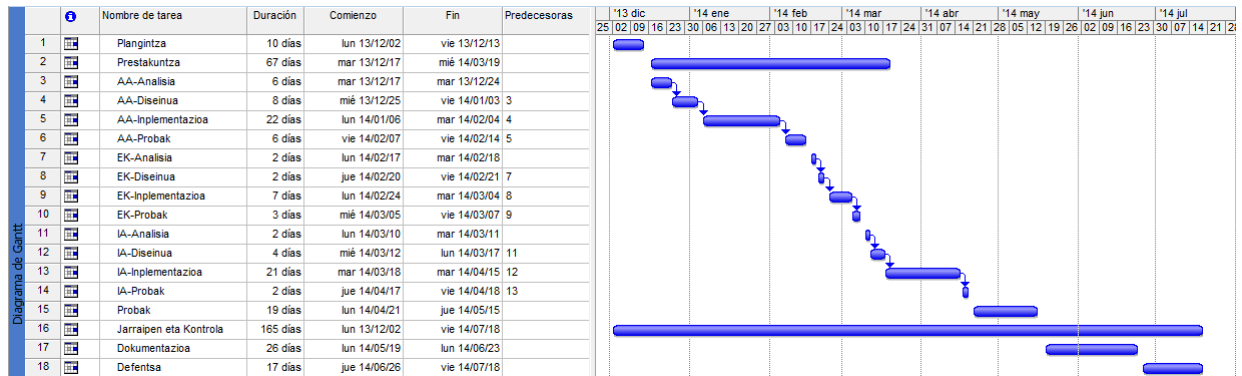
Jarraian datorren taulan proiektuaren ataza bakoitzarentzat estimatu den gauzatze denbora azaltzen da.

Ataza	Estimatutako denbora
Plangintza	30 ordu
Prestakuntza	30 ordu
Garapena	210 ordu
<i>Android Aplikazioa</i>	96 ordu
Analisia	10 ordu
Diseinua	14 ordu
Inplementazioa	62 ordu
Probak	10 ordu
Eduki Kudeatzailea	35 ordu
Analisia	4 ordu
Diseinua	6 ordu
Inplementazioa	20 ordu
Probak	5 ordu
Itzultzaile automatikoa	79 ordu
Analisia	8 ordu
Diseinua	12 ordu
Inplementazioa	51 ordu
Probak	8 ordu
Probak	15 ordu
Jarraipen eta Kontrola	20 ordu
Dokumentazioa	80 ordu
Defentsa	20 ordu
GUZTIRA	405 ordu

A.1 Taula: Atazen denbora estimazioa erakusten duen taula.

Gantt diagrama

Jarraian proiektu honen Gantt diagrama dago, ataza bakoitzaren garapena adierazten duena, denboran zehar. Erabilitako datuak estimazioetan oinarrituta daudenez, ez dira guztiz zuzenak izango, baina proiektuaren atazen banaketa nola izango den erakusteko balio du.



A.1 Irudia: Denboran zehar emango den atazen banaketa erakusten duen Gantt diagrama.

A.2 Lan metodologia

Proiektuaren garapenean hainbat esparru ezberdin daude, eta horiek nolabait antolatu eta horien funtzionamendua ahalik eta garbien edukitze aldera, metodologia bat ezarri dugu horietako bakoitzeko.

- **Bilerak:** Proiektu ia osoan zehar, proiektuaren tutorearen eta garatzen ari den ikaslearen arteko sakabanaketa geografiko handia egongo da, kurtsoan zehar ikaslea Galesen ikasten egongo da eta. Hortaz, normalean ezingo dira bilera fisikoak egin. Horren ordean, posta elektronikoko bidezko komunikazioa sustatuko da, edozein zalantza edota proiektuaren inguruko bestelako gai batzuk jorrazteko. Komunikabide honek azkartasunez igorri dezake mezua bi alderdien artean, baina horretarako posta helbidea maiz kontsultatuko dela ziurtatu beharko da. Posta mezu bidez zalantzak argitu ezin badira, sarearen bitartez egindako bideo-deiak ere erabiliko dira, aurrez aurrekoak (pantaila baten aurrean bada ere) ematen dituen abantailak aprobetxatzeko asmoz. Ikaslea Euskal Herrian dagoen bitartean, berriz, oso komenigarria izango da aurrez-aurreko bilerak izatea, lanaren jarraipen intentsiboago bat egiteko balioko baitu. Hori baino gauza txikiagoentzat, posta mezuen bidez ere bideratu ahalko da komunikazioa.

- **Ordutegiak:** Proiektuaren garatzearen zati handi bat kurtsoarekin batera gertatuko da, beste modu batera esanda, proiektuaz gain atenzioa behar duten beste ikasgai batzuk ere egongo dira. Kurtsoa aurrera doan heinean, bi ildoetara, proiektua batetik eta ikasgaiak bestetik, eutsi beharko zaie, biak une berean aurrera eramanez. Horrek lanak azken unerako ez uztea suposatzen du, horrela bi ildoen arteko denbora oreka bat lortuko da eta, oso komenigarria dena lanaren aurreratzea ez mozteko. Azterketen garaian berriz, kurtsoko ikasgaiak lehentasuna izango dute, baina aldi berean proiektuarekin seriotasunez jokatu da. Era berean, proiektuaren mugarrak ez betetzeko arriskua badago, esfortzu berezi bat egin beharko da, estimatutako epeetatik ez urruntzeko.
- **Inplementazioa:** Proiektuaren atal ezberdinen garapena aurrera doan bitartean probak egingo dira, bai moduluaren funtzionalitateak aurrera doazen heinean bai moduluak bukatu denean. Horrela, ziurtasunez jakin ahalko dugu gure implementazioa bide onetik doala. Gainera, proiektuaren atal ezberdinak bateratzean beste proba fase bat ere egongo da, elementu guztiak batera ondo dabiltzala ziurtatzeko eta hobetu daitezkeen atalak identifikatzeko.
- **Prestakuntza:** Prestakuntza proiektuaren zati gehienean zehar modu jarrai batean emango da, intentsitate ezberdinekin izango bada ere. Atal bakoitzaren analisi garaian prestakuntzak garrantzi handiagoa hartuko du eta horrela mantenduko da atal horren implementazioa bukatzen den arte. Helburu honetarako erabiliko diren baliabide gehienak saretik hartutakoak izango dira, libururen baten beharra ere suertatu daitezkeen arren.

A.3 Baliabideen esleipena

Proiektua hau garatzeko baliabide ezberdin ugari erabiliko dira, bai hardware mailan bai software mailan.

- **Erabiliko den hardwarea:**
 - **Ordenagailu eramangarria:** proiektua garatzeko erabiliko da, eta baita dokumentazioa idatzi eta aurkezpena prestatzeko. Ordenagailu horrek *Windows 7* sistema eragilea erabiliko du.
 - **USB memoria:** segurtasun neurri bezala egingo diren babes-kopien gordailu gisara erabiliko da.

- Zerbitzaria: Eduki Kudeatzailea kokatzeko erabiliko da. Proiektuaren izaera ez-komertziala dela eta, doako hosting aukerak baliatuko dira.
- Erabiliko den softwarea:
 - *Eclipse* programazio ingurunea, *Android*-en programatzeko *ADT (Android Developer Tools)* pluginarekin: *Android* aplikazioa garatzeko erabiliko da.
 - Eduki Kudeatzailea garatzeko erabiliko den Edukiak Kudeatzeko Sistema.
 - Itzultzaile automatikoa: itzulpen automatikoak egiten dituen zerbitzua, gure Eduki Kudeatzailean txertatuko den moduluak erabiliko duena.
 - QR kodeen irakurketa egiteaz arduratzen den kodea.
 - QR kodeak sortzeaz arduratzen den zerbitzua.
 - FTP bezeroa: Eduki Kudeatzailearen zerbitzariarekin fitxategien komunikazioa burutzeko.
 - *Skype*: tutore eta ikaslearen arteko bideo-deiak burutzeko erabiliko da, behar izanez gero.
 - *Texmaker*: proiektuan zehar behar diren dokumentuak idazteko erabiliko da, memoria barne.
 - *Microsoft Power Point*: proiektuaren aurkezpena garatzeko erabiliko da.
 - *Microsoft Visio*: Plangintzan erabiliko diren hainbat grafiko egiteko erabiliko da, esate baterako, LDEa edota mugarren diagrama.
 - *Microsoft Project*: Plangintzan sortuko den *Gantt* diagrama egiteko erabiliko da.
 - *StarUML*: Software hau Erabilpen Kasuen diagramak eta Sekuentzia Diagramak egiteko erabiliko da.
 - *Dia*: Software honekin kodearen Klase diagrama egingo da.

A.4 Kalitate plana

Proiektuaren egikaritzea neurtzeko modu garbiena, zalantzarik gabe, prototipoa ikustea da. Hala ere, hasierako helburu horien betetze minimoa ziurtatzeaz gain urrats bat harago joan beharrean gaude, produktuak zein proiektuak kalitate maila minimo bat betetzen dutela ziurtatzeko.

Proiektuaren kudeaketaren kalitatea

- **Kalitate elementua:** mugarriekiko atzerapena
Azalpena: Atzerapenen bat gertatzea ulergarria da, proiektuan zehar ezbeharrak gerta daitezke eta, baina kudeaketa kalitate egokiz egin dela ziurtatzeko plangintza eta errealitatearen arteko konparaketa bat egingo da.
Neurtzeko modua: plangintzan ezarritako barne mugarriekiko atzerapena neurtuko da.
Maximo onargarria: 8 eguneko atzerapena guztira.
Lortu nahi den neurria: 0-5 eguneko atzerapena.

Aplikazioaren kalitatea

- **Kalitate elementua:** *Android* aplikazioak jariakortasunez funtzionatzea.
Azalpena: Aplikazioa laguntza bat izan behar da erabiltzaileentzat, eta ez oztopoa. Beraz, aplikazioak modu 'azkar' batean lan egin beharko du, eta hasieratu edota martxan hasteko ez luke denbora asko hartu behar.
Neurtzeko modua: Aplikazioaren funtzionalitateak erabiltzen hasi aurretik itxaron behar den denbora neurtu.
Maximo onargarria: 7 segundo.
Lortu nahi den neurria: 1-3 segundo baino gutxiago behar izatea.

Eduki Kudeatzailearen kalitatea

- **Kalitate elementua:** Eduki Kudeatzailearen bizkarrezurra hizkuntza ezberdinetan egotea.
Azalpena: Proiektuaren helburua informazioa hizkuntza ezberdinetan erakusteko gaitasuna izatea da, hortaz, Eduki Kudeatzailearen atalak hizkuntza ezberdinetan egon beharko dira, edukiak gehitu edota itzulpenak egin ahal izateko.
Neurtzeko modua: Eduki Kudeatzailea zenbat hizkuntza ezberdinetan dagoen neurtu.
Minimo onargarria: 2-4 hizkuntza.
Lortu nahi den neurria: 5-8 hizkuntza.

Itzultzaile Automatikoaren kalitatea

- **Kalitate elementua:** Itzultzaile Automatikoak itzulpen aukera ugari edukitzea.
Azalpena: Itzultzaile Automatikoa Eduki Kudeatzailearen eleaniztasuna sustatzeko pentsatuta dagoenez komenigarria da itzultzaileak Eduki Kudeatzaileak dituen hizkuntzetatik beste batzuetara itzultzeko aukera eskaintzea.
Neurtzeko modua: Eduki Kudeatzailean ezarrita dagoen hizkuntza bakoitzak ezarrita dauden beste hizkuntzetara itzultzeko duten aukera kopurua.
Minimo onargarria: Hizkuntza bakoitzak 1-2 hizkuntzatarara itzultzeko aukera izatea.
Lortu nahi den neurria: Hizkuntza bakoitzak webgunean ezarrita dauden beste hizkuntza guztietara itzultzeko aukera izatea.

Produktu osoaren kalitatea

- **Kalitate elementua:** Eduki bat atzitzeko behar den denbora.
Azalpena: Produktu osoak hainbat etapa pasatu behar ditu, lehenik eta behin *Android* aplikazioa ireki eta QR kodea irakurri, gero informazio hori atzitu eta behar izatekotan mugikorren hizkuntzara itzuli. Prozesu horiek guztiak exekutatzeko denbora txikia izatea komeni da, erabiltzailea itxaroten egon ez dadin.
Neurtzeko modua: Aplikazioa martxan jartzen den unetik edukia erakusten duen arteko denbora neurtzea. Ziurtasun handiagoa izatearren, gailu ezberdinekin probatuko da, modu orokor batean ondo funtzionatzen al duen ikusteko.
Maximo onargarria: 15 segundo.
Lortu nahi den neurria: 5-10 segundo behar izatea.
- **Kalitate elementua:** Produktuaren erabilera argia da.
Azalpena: Produktua testuinguru ezberdin askotatik datozen erabiltzaileek erabiltzeko aukera legoke. Ziurtatu behar da produktua inoiz erabili ez dutenentzat erraza dela produktuaren nondik norakoak ulertzea eta horiek erabiltzea.
Neurtzeko modua: Produktua erabiltzaile ezberdinei emango zaio eta erabiltzen duten bitartean izandako erreakzioak aztertuko dira. Prozesu hau iteratiboa izango da ziurrenik, hobekuntzak egiten joan ahala ziurtatu beharko baita.
- **Kalitate elementua:** Produktuaren erabileraren inguruko esku liburua.
Azalpena: Erabilerraztasun probak egin arren, aztertutako lagin txikia izango da

eta beraz, ez ditugu erabiltzaile mota ezberdin guztiak aztertu ziurrenik. Horregatik, esku liburu argi bat beharko da, zalantzak izan ditzaketen erabiltzaileek kontsulta dezaten.

Neurtzeko modua: Produktuak eskaintzen dituen erabilpen kasu guztiak ilustratu beharko ditu, bai testu bidez eta baita irudien laguntzaz.

Memoriaren kalitatea

- **Kalitate elementua:** Memoriaren luzera.

Azalpena: Proiektuaren memorian hilabeteak iraun dituen prozesu baten dokumentatzea egingo da, eta hain luzaroan iraun duen zerbaitek xehetasun asko ditu eskaintzeko. Horregatik, memorian horien berri eman beharko da ahalik eta modurik argienean, behar diren azalpen guztiak eskainiz. Argi dago memoria luzea izateak ez duela esan nahi memoria ona denik, baina hasiera puntu on bat da.

Neurtzeko modua: Memoriak duen orrialde kopurua neurtzea.

Minimo onargarria: 75 orrialde.

Lortu nahi den neurria: 100 orrialdetik gorako memoria.

A.5 Arriskuen plana

Proiektuaren garapen prozesuan zehar hainbat ezusteko gerta daitezke, batzuk ezbeharrak sortutakoak, eta besteak, berriz, jarrera ezegoki batzuek lagundutakoak. Jarraian identifikatu diren hainbat arrisku azaltzen dira, eta horientzako kontingentzia planak edo erreakzioak ere aurreikusten dira.

Baliabide propioen kaltetzea

- **Arriskua:** Datuen galera gertatzea.

Azalpena: Baliabideen atalean aipatu den bezala, hainbat azpiegitura erabiliko dira proiektu honetan. Horrek, informazioa galtzeko aukerak handitzen ditu, esate baterako, proiektu honetan Eduki Kudeatzailea dagoen zerbitzariko datuak gal daitezke edota ordenagailuan bertan gordetakoak.

Eragina: Proiektua aurrera doan heinean handitzen doa, baina orokorrean eragina oso handia da. Datuen galerak lana berriz egitea suposatzen du, eta hori edozein

unetan gerta daitekeen zerbait da.

Gertatzeko probabilitatea: Txikia. Gauzak ohiko egoera batean (ordenagailua edota zerbitzaria ez dira apurtu) egongo balira, datuen galera ez litzateke ohiko gauza bat.

Ekiditeko estrategia: Fitxategien babes-kopiak egin, horien erredundantzia sustatuz. Prozesu hori periodikoki egingo da, epe laburretan (ez aste bat baino gehiago kopia batetik bestera). Erabiliko diren gordailuak USB memoria bat eta software bertsio kontrolaz arduratzen den Git programa (fitxategiak zerbitzari batean gordeko dituen).

- **Arriskua:** Ordenagailuak funtzionatzeari uztea.

Azalpena: Gailu elektronikoek zoritxarrez ez dute betirako irauten eta gutxien esperotako momentuan puskatu daitezke.

Eragina: Ertaina. Babes-kopiak egitea dela eta ez luke proiektuan egindako lanean eraginik izan behar baina noski, beste ordenagailu bat bilatu beharrak, eta horretan behar bezainbeste programa jartzeak denbora kopuru handi bat beharko du.

Gertatzeko probabilitatea: Txikia. Aurreikusi ezin daitekeen gertakari bat da, normalean gertatuko ez dena, baina gerta daiteke.

Ekiditeko estrategia: Ordezko ordenagailu bat bilatzea erraza litzateke, erosiko ez bada (normalena erostea litzateke) fakultatean dauden ordenagailuen erabilera egin daiteke eta (ordutegiek mugatuko lukete horien erabilera, klaseak gela horietan ematen dira eta). Horretaz gain, komenigarria da proiektuan zehar behar diren programen zerrenda bat edukitzea eta baita horiek jaitsi edo erabiltzeko behar den informazioa dokumentu batean gordetzea.

- **Arriskua:** Zerbitzaria atzitu ezina.

Azalpena: Eduki Kudeatzailea zerbitzari batean kokatuta dago eta hori atzitzeko Internet konexio bat behar da. Baliteke komunikazioaren bi muturretakoren batean arazoren bat egotea, Eduki Kudeatzailea atzitzea ezinezko egiten duena.

Eragina: Handia. Eduki Kudeatzailearen beharra dagoenean erabiltzeko ezintasunak, garapenari oztopo bat jartzen dio.

Gertatzeko probabilitatea: Txikia.

Ekiditeko estrategia: Babes-kopia erabiliz gero, posiblea da Eduki Kudeatzailea modu lokal batean garatzea eta gauzak bere onera itzultzean zerbitzarira bertsio berria igotzea. Eduki Kudeatzaileak sarean dauden osagaiekin komunikatu behar badu, baliteke ezin izatea, arazoa gure Internet zerbitzuan badago. Azken egoera

honetan, alternatiba bat unibertsitatera joatea litzateke hango internet konexioarekin edota hango ordenagailuekin lan egitera.

Produktuaren kanpoko zerbitzuekin erlazionatuak

- **Arriskua:** Erabiliko diren elementuen eguneratzeak gertatzea.

Azalpena: Garapena gertatzen ari den bitartean (edo bukatu ondoren) baliteke erabiltzen dugun osagai baten (*Android* sistema eragilea, Eduki Kudeatzailea...) bertsio berri bat ateratzea.

Eragina: Handia. Egoera batzuetan baliteke guk inplementatutako osagaiek ez funtzionatzea, bertsio berriak dituen ezaugarriak direla eta.

Gertatzeko probabilitatea: Ertaina.

Ekiditeko estrategia: Inplementazioa egiten ari garenean, erabiltzen ari garen elementuei buruz dokumentazioan kontsulta egitea, hurrengo bertsioetan ezabatzeko asmoa al duten jakiteko. Kasu horretan osagai hori ez litzateke erabiliko. Eguneraketaren batek produktuaren funtzionalitateari modu negatiboan eragiten badio proiektuaren garapen fasearen bukaeratik gertu, gatazka sortzen duen eguneraketa ez instalatzea ere kontutan hartu liteke.
- **Arriskua:** Erabiliko diren elementuak ordaintzeko bertsioetara pasatzea.

Azalpena: Garapena gertatzen ari den bitartean (edo bukatu ondoren) baliteke erabiltzen dugun osagai bat (QR irakurlea, itzulketaz arduratzen zen web-zerbitzua...) aldatzea eta egileek zerbitzu hori erabiltzeko ordaindu egin beharko dela erabakitzea.

Eragina: Handia. Gure funtzionalitate ugarik ez lukete funtzionatuko zerbitzu horiek gabe.

Gertatzeko probabilitatea: Txikia.

Ekiditeko estrategia: Lehenik eta behin, egileei mezu bat idatzi, ikasleentzako doako zerbitzu bat (mugatua bada ere) eskaintzen al duten jakiteko. Erantzuna ezezkoa izatekotan edo emandakoa baino zerbait gehiago behar izatekotan, beste zerbitzu bat bilatu beharko da, doakoa dena. Topatzen ez bada, tutorearekin hitz egin beharko da, zerbitzuarengatik ordaintzeko aukeraz.

Proiektuaren mugarriekiko eragina dutenak

- **Arriskua:** Proiektuaren barneko epeak ez betetzea.
Azalpena: Gaixotasunak, unibertitateko lan eskerga handia edota modu okerrean estimatutako denbora direla eta baliteke proiekturako ezarritako barne mugarriak ez betetzea.
Eragina: Proiektuaren hasieran eragina ez da oso handia izango, errekupeartzeko denbora egongo delako, birplanifikatuz gero. Bukaeratik gertu gaudenean, ordea, denbora marjina txikiagoa egongo da, eta horrek baliteke bukaerako proiektuan eragin handia izatea.
Gertatzeko probabilitatea: Ertaina.
Ekiditeko estrategia: Birplanifikazio bat egingo da, barne mugarriak eguneratuz, proiektua arrakastaz bukatzen dela ziurtatzeko.
- **Arriskua:** Aurkezpenaren eguna aldatzea.
Azalpena: Arrazoi ezberdinak direla eta, baliteke ezinezkoa izatea aurkezpena aurreikusitako egunetan egitea.
Eragina: Dataren aldaketaren noranzkoaren arabera izango da. Aurkezpenaren epea aurreratzeotan, eragin handia izango du, prestakuntza denbora txikitu egingo duelako. Epea atzeratzeotan, berriz, eragina ez da hain handia izango planifikatutakoa burutzeko denbora gehiago egongo baita.
Gertatzeko probabilitatea: Oso txikia.
Ekiditeko estrategia: Birplanifikazio bat egingo da, aurkezpenaren prestakuntza egokia egingo dela ziurtatzeko.

A.6 Jarraipen txostena

Atal honetan proiektuaren plangintzan aurreikusitakoa bukaeran gertatu denarekin alderatuko dugu.

A.6.1 Desbideraketak

Proiektuaren hasieran plangintza egin zenean proiektuan orotara 405 ordu lan egingo zirela aurreikusi zen. Hurrengo orrian dagoen taulan aurreikusitako denboraren eta dedikatutako denboraren arteko alderaketa egingo dugu.

Ataza	Estimazioa	Dedikatua	Desbideratzea
Plangintza	30 ordu	25 ordu	-%16,7
Prestakuntza	30 ordu	24 ordu	-%20
Garapena	210 ordu	204 ordu	-%2,9
<i>Android Aplikazioa</i>	96 ordu	87 ordu	-%9,4
Analisia	10 ordu	8 ordu	-%20
Diseinua	14 ordu	12 ordu	-%14,3
Inplementazioa	62 ordu	57 ordu	-%8,1
Probak	10 ordu	10 ordu	%0
Eduki Kudeatzailea	35 ordu	34 ordu	-%2,9
Analisia	4 ordu	4 ordu	%0
Diseinua	6 ordu	6 ordu	%0
Inplementazioa	20 ordu	19 ordu	-%5
Probak	5 ordu	5 ordu	%0
Itzultzaile automatikoa	79 ordu	83 ordu	%5,1
Analisia	8 ordu	7 ordu	-%12,5
Diseinua	12 ordu	13 ordu	%8,3
Inplementazioa	51 ordu	52 ordu	%2
Probak	8 ordu	11 ordu	%37,5
Probak	15 ordu	11 ordu	-%26,7
Jarraipen eta Kontrola	20 ordu	17 ordu	-%15
Dokumentazioa	80 ordu	86 ordu	%7,5
Defentsa	20 ordu	18 ordu	-%10
GUZTIRA	405 ordu	385 ordu	-%4,9

A.2 Taula: Atazen denboraren estimazioen eta dedikatutako denboraren datuak erakusten dituen taula.

Taulari begiratuta ikus daiteke proiektuaren desbideratzea guztira ez dela oso handia izan, -%4,9-koa alegia. Proiektuaren ataza multzo nagusiei dagokienean desbideratze handia dago Probak (-%26,7), Prestakuntza (-%20) eta Plangintza (-%16,7) atazetan. Hala ere, ataza horiek duten ordu kopuruari erreparatzen badiogu konturatuko gara proiektuan duten eragina ez dela hain handia, eta duten eragina ez dela negatiboa, denbora gutxiagoan gauzatu dira eta. Izan ere, hiru ataza horiek proiekturako aurreikusitako 405 orduetatik 75 ordu edota dedikatutako 385 orduetatik 60 ordu osatzen dituzte, orduen %18.5 eta %15.5-a, hurrenez hurren.

Ordu kopuru gehien behar duten atazei erreparatuz gero aipagarria da Dokumentazioa (%7,5), *Android Aplikazioa* (-%9,4) eta Itzultzaile Automatikoa (%5,1) ataza multzoetan gertatzen diren desbideratzeak. Portzentaian fijatuz gero argi dago ez dela desbideratze handia baina ataza horiek duten ordu kopurua dela eta bertan gertatzen diren

desbideratzeak eragin handia dute proiektuaren gauzatzean. Izan ere, aurreikusitako 405 orduetatik 255 ordu edota dedikatutako 385 orduetatik 256 ordu osatzen dute, %62,3 eta %66,5, hurrenez hurren. Orain, desbideratze horien arrazoia espekulazioa egingo da. Hasteko, Dokumentazioa atazan izan den desbideratzea azaltzeko ikaslearen jardunean erreparatu beharra dago. Informatika Ingeniaritza Graduaren lau urteetan zehar Gradu Amaierako Proiekturako gaitasunak landu dira, GAP-ean egin beharreko atazak eskala txikian eta bata bestetik era independentean eginez. Plangintza egiterako orduan tamaina horretarako aurrekaririk ez zegoenez dokumentazioa idazteko denbora estimatzea zaila izan zen.

Bestetik, *Android* aplikazioaren garapena aurreikusitakoa baina %9,4 lehenago bukatu zen. Hori azaltzeko ikasleak *Android* Sistema Eragilerako programazioan zuen eskarmentu falta hartu behar da kontutan, *Android* ezagutu aurretik ezezagunak baitziren eskaintzen zituen laguntza guztiak. Azkenik, Itzultzailea Automatikoan izandako %5,1-eko atzerapena azaltzeko ataza horien garapenean jazotako arazoei begiratu behar zaie. Izan ere, ataza horren garapenean *qTranslate* eta *WordPress*-en arteko bateraezintasuna eman zen, gehigarriaren garapena oztopatu zuena. Bestetik, erabiltzaileak editorearekin duen portaerak (fitxen artean nabigatzea, *Text* edo *Visual* fitxa hautatuta izatea...) garapena asko zaildu zuen, funtzionamendu okerraren arrazoa azaltzeko hipotesia sortzera derrigortu zuena, lehenengo saiakeretan asmatu ez zen zerbait.

A.6.2 Kalitatea

Plangintzan ezarri ziren kalitate elementuen betetze mailari buruz hitz egingo dugu orain. Lehenik eta behin, proiektuaren kudeaketaren kalitateri buruz hitz egingo dugu. Esparru horren kalitatea neurtzeko erreferentzia gisa mugarriekiko izandako atzerapena hartu da. Proiektuaren garapenean zehar bi barne mugarriekiko atzerapena egon da: Itzultzaile automatikoa bukatzeko eta dokumentazioa bukatzeko mugarrietan, hain zuzen ere. Atzerapen hori 2 eta 5 egunekoa izan zen, hurrenez hurren. Kalitate elementuan esaten da proiektuko mugarrien artean guztira izandako atzerapena gehienez 8 egunekoa izan behar zela eta gure kasuan 7 egunekoa da, beraz, ozta-ozta bada ere muga barruan dagoen balioa da.

Aplikazioaren kalitatea neurtzeko aplikazioaren funtzionalitateak erabili aurretik itxaron beharreko denbora hartzen da kontutan kalitatea neurtzeko, hau da, aplikazioa irekitzeko itxaron beharreko denbora. Aplikazioarekin gailu ezberdinetan egindako proben

emaitzak neurtu ondoren batazbestekoa 2 eta 3 segundo artean dago, hasiera batean lortu nahi zen balio tartearen (1 eta 3 segundoren tartean) barruan, beraz.

Bestetik, Eduki Kudeatzailearen kalitatea neurtzeko ezarrita zeuden hizkuntz kopurua ezarri zen. Eduki Kudeatzaileak 5 hizkuntz dauzka ezarrita, beraz, hasiera batean lortu nahi zen balio tartearen barnean dago, 5 eta 8 hizkuntza eduki beharko lituzkeela erabaki baitzen.

Itzultzaile Automatikoaren kalitatea neurtzeko Eduki Kudeatzailean ezarrita dagoen hizkuntz bakoitzak dituen itzulpen aukera kopurua ezarri zen. Balio minimo bezala hizkuntz bakoitzak gutxienez 1-2 hizkuntzara itzultzeko gai izan behar zela erabaki zen. Itzultzailearen itzulpen aukerei erreparatuz gero ikusiko dugu hizkuntz bakoitzak gutxienez bi hizkuntzatarara itzuli dezakeela.

Horretaz gain, produktu osoaren kalitatea neurtzeko hiru neurgailu ezarri ziren: aplikazioa irekitzen denetik Eduki Kudeatzaileko eduki bat atzitzeko behar den denbora, produktuaren erabiltzeko argitasuna eta produktuaren erabilera gidaren osotasuna. Lehenago aipatu den bezala, *hosting* zerbitzuak ez du errendimendu uniforme bat, batzuetan ez baita atzigarri egoten. Neurketak egiteko salbuespen egoera hori alde batera utziko dugu, zerbitzaria atzigarri dagoen probak soilik kontutan hartuz eta horretaz gain Internet konexio egoki bat dauden probak kontutan hartuko dira soilik. Izan ere, produktua erabiltzeko Internet konexioa beharrezkoa da, konexio horren abiaduraren arabera abiadura izango du produktuak. Aipatutako baldintzetan egindako proben batazbestekoa 7 eta 8 segundoren tartean dabil, plangintzan ezarritako 5-10 segundoen barnean dagoena. Bestalde, gainontzeko beste bi kalitate elementuak neurri batean lotuta doaz erabilera gida oso baten produktuaren erabilera argi bat errazten du eta. Garapenean zehar egin diren proben bitartez ikusi da erabiltzaile ezberdinek ez dutela arazo askorik topatu produktua erabiltzean eta topatu dutenean erabilera-gidaren bitartez zalantzak argitu dituztela. Beraz, esan daiteke bi kalitate elementu horiek ere bete direla.

Memoriaren kalitateari dagokionez, kalitate elementu gisa memoriaren luzera hartu zen, balio minimo gisa 75 orrialde eta lortu nahi zen luzera gisa 100 orrialdetik gorako memoria izatea ezarriz. Memoriak 100 orrialdetik gorako luzera duenez, kalitate elementua bete egin dela esan daiteke.

A.6.3 Arriskuak

Proiektuaren hasieran identifikatu ziren arrisku ezberdinen artean badira zenbait proiektuan zehar gertatu izan direnak. Hasteko, datuen galera gertatu da, zerbitzarian hain zuzen ere. Horri aurre egiteko, aurretik egindako babes-kopiak erabili ziren, zerbitzaria bere egoera normalera itzuliz bestelako eraginik gabe.

Horretaz gain, zerbitzaria atzitu ezin izatea ere gertatu da behin baino gehiagotan. Normalean, minutu batzuetako kontua izan zen, beraz, ez zen bestelako ekintzarik egin behar izan. Hala ere, kasu bakanen batean denbora tarte luzeagoan egon da eskuragaitz. Horrelako egoerak Eduki Kudeatzailea zerbitzarian probatua izaten ari zenean gertatu ziren, beraz, beharrezkoa zen zerbitzarian lan egitea (modu lokalean probatzeak ez zuen balio). Kasu horiek proiektuaren beste esparru batzuetan aurreratzeko aprobeixatu ziren: dokumentazioa, jarraipena. . .

Proiektuan gertatu den beste arrisku bat produktuan erabiltzen diren elementuen eguneratzea izan da. Eguneraketak egon diren arren normalean ez dute eraginik izan aurretik egindako lanean. Hala ere, *WordPress 3.9* bertsioa atera zenean *qTranslate* bateraezina zen eta horrek eragina izan zuen bai itzultzailean bai Eduki Kudeatzailea arakatzaitetik ikustean. Horren aurrean, *WordPress*-en aurreko bertsio bateragarrira itzultzea erabaki zen, 3.8.3 bertsiora, alegia.

Azkenik, proiektuaren barne-mugarriak ez betetzea ere gertatu da. Lehen aipatu bezala atzerapen horiek ez dira oso handiak izan, 2 egun Itzultzaile Automatikoa bukatzeko mugarrin eta 5 egun dokumentazioa bukatzeko mugarrin. Lehenengo kasuan nahikoa izan zen hurrengo mugarrirako lan gehigarria egitea, atzerapen hori neutralizatzeko. Bigarren kasuan, berriz, proiektua entregatzeko epea hain gertu egonda planifikatutakoa baina bost egun gehiago lan egitea besterik ez da geratu.

B. ERANSKINA

Erabilera Gida

Eranskin honetan produktuaren osagaien erabilera azaltzen da, gida baten bitartez. Horretarako, osagai bakoitza banaka banaka azalduko da.

B.1 Ohar Eleanitzak aplikazioa

Atal honetan *Android* aplikazioaren erabilera azalduko dugu, egin daitezkeen ekintza ezberdinak deskribatuz. Prozesu ezberdinen azalpenak laguntzeko gailuaren pantaila kapturak erabiliko dira. Hala ere, *Android* gailuetan kamera erabili bitartean kapturak egiten badira, irudi horietan ez da kamerak hartzen dituen irudirik agertuko, horren ordez dena beltz erakutsiz.

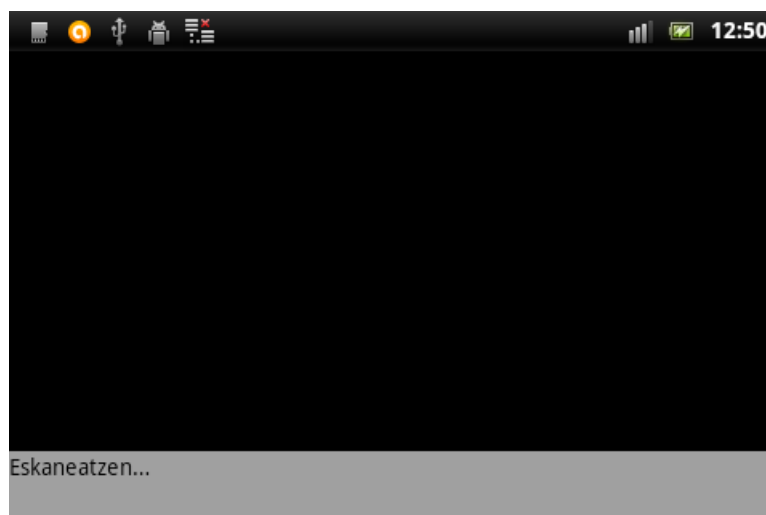
Aplikazioa ireki

1. Joan zaitetz zure *Android* gailuaren menu nagusira, bilatu *Ohar Eleanitzak* aplikazioa eta ikonoa sakatu.



B.1 Irudia: *Ohar Eleanitzak* aplikazioaren ikonoa *Android* gailuaren menu nagusian.

2. Gailuaren pantailan aplikazioa irekiko zaizu, QR kodeak irakurtzeko prest.



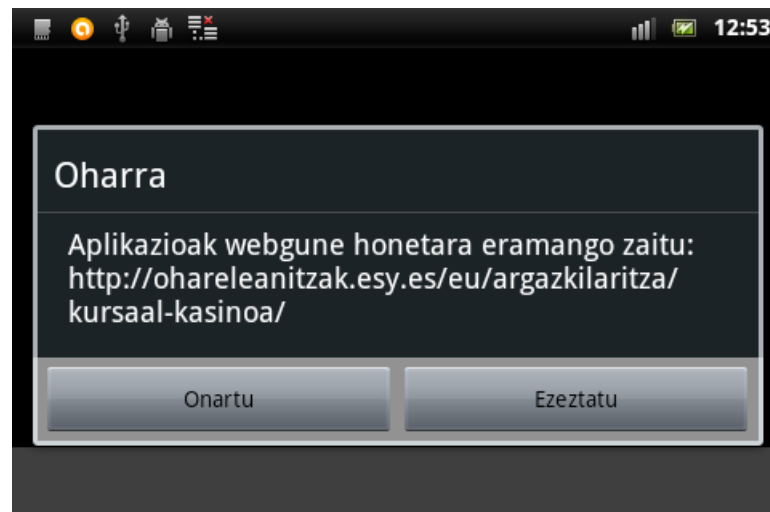
B.2 Irudia: *Ohar Eleanitzak* aplikazioaren hasiera pantaila erakusten duen irudia.

QR kodea irakurri

Ekintza hau egiteko bi modu ezberdin dauzkazu, *Birbideratzea Automatikoki Onartu* ezarpenaren balioaren arabera izango dena. Era lehenetsian, ezarpena ezgaituta egongo da.

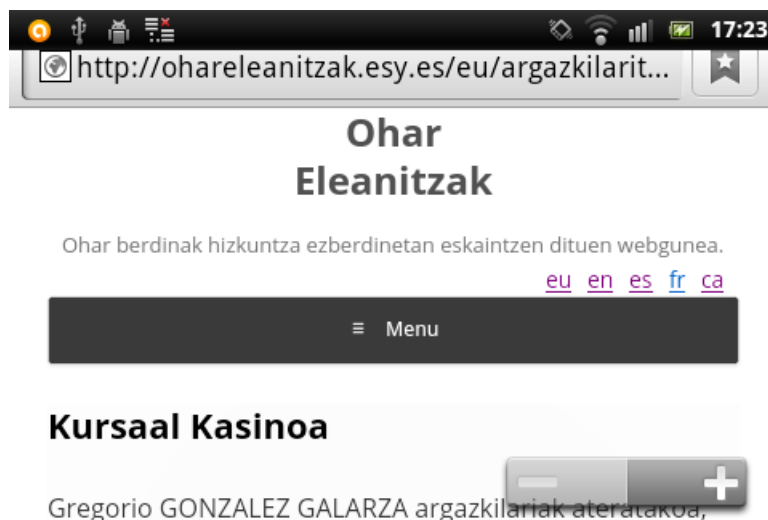
***Birbideratzea Automatikoki Onartu* ezarpena ezgaituta**

1. Enfoka ezazu QR kode bat zure gailuaren kamerarekin (erabili pantailak erakusten dizkizun irudiak laguntza bezala). Kode bat fokatzen duzunean aplikazioak konfirmazioa eskatuko dizu, ezetz esaten baduzu aplikazioaren hasierako pantailara itzuli zara.



B.3 Irudia: Aplikazioak kode bat irakurtzean erakusten duen konfirmazio mezua.

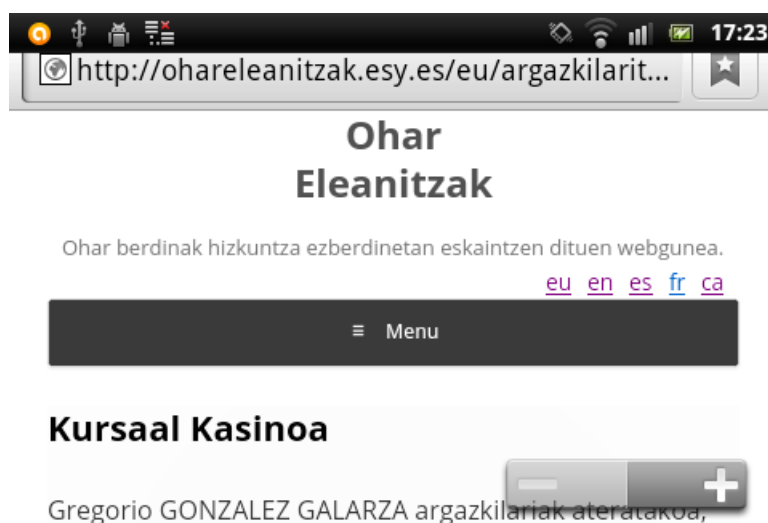
2. Baietz esaten baduzu aplikazioak kode horren edukia arakatzailan irekiko du.



B.4 Irudia: Kodeak gordetzen zuen helbidearen edukia, gailuaren arakatzailan ikusita.

Birbideratzea Automatikoki Onartu ezarpena gaituta

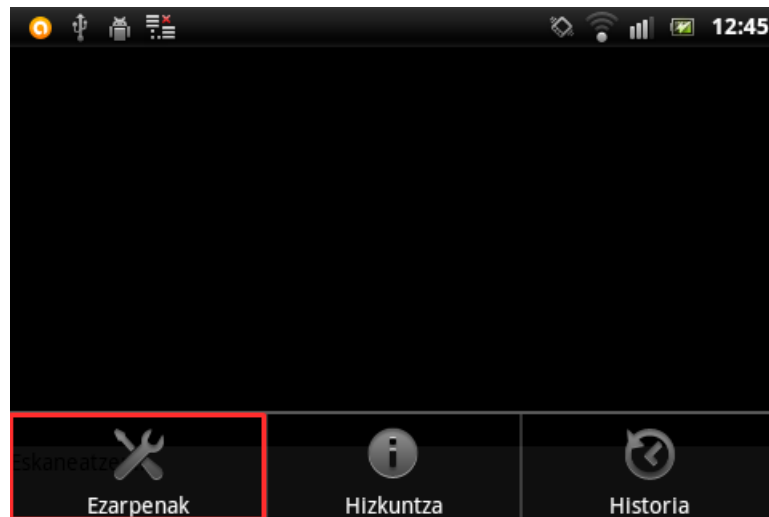
Enfoka ezazu QR kode bat zure gailuaren kamerarekin (erabili pantailak erakusten dizkizun irudiak laguntza bezala). Kode bat fokatzen duzunean aplikazioak edukia arakatzailan irekiko du.



B.5 Irudia: Kodeak gordetzen zuen helbidearen edukia, gailuaren arakatzailan ikusita.

Birbideratzea Automatikoki Onartu ezarpenaren balioa aldatu

1. Aplikazioaren hasierako pantailan zaudela sakatu aukerak ikusteari. Irekiko zaizun menuan aukeratu ezazu *Ezarpenak* aukera.

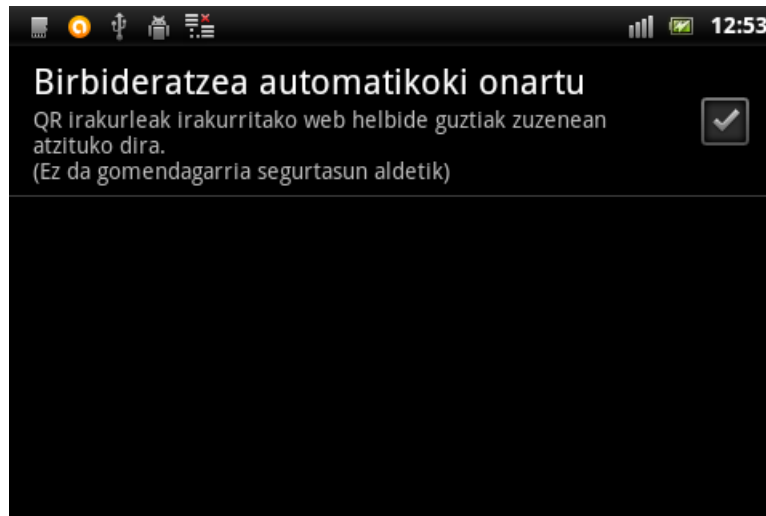


B.6 Irudia: *Ohar Eleanitzak* aplikazioaren aukeren barra *Ezarpenak* aukera irekitzeko.

2. Irekiko zaizun pantailan emaiozu *Birbideratzea Automatikoki Onartu* ezarpenari zuk nahi duzun balioa, ezarpenaren gainean sakatuz.



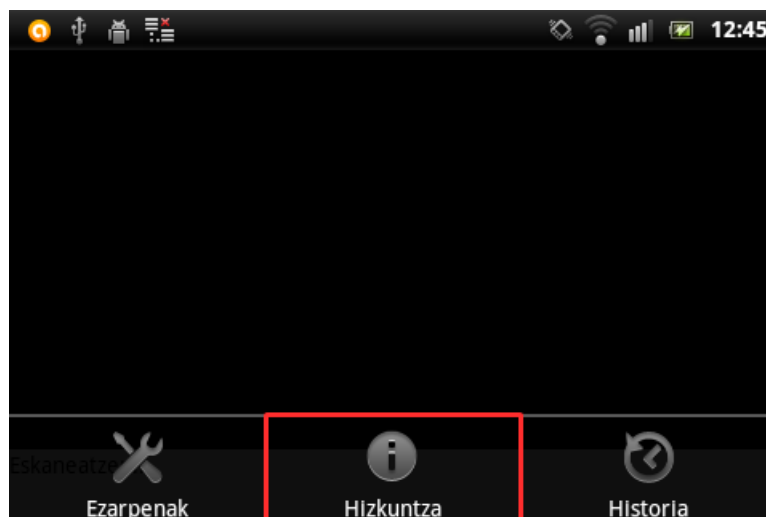
B.7 Irudia: *Birbideratzea Automatikoki Onartu* ezarpena ezgaituta.



B.8 Irudia: *Birbideratzea Automatikoki Onartu* ezarpena gaituta.

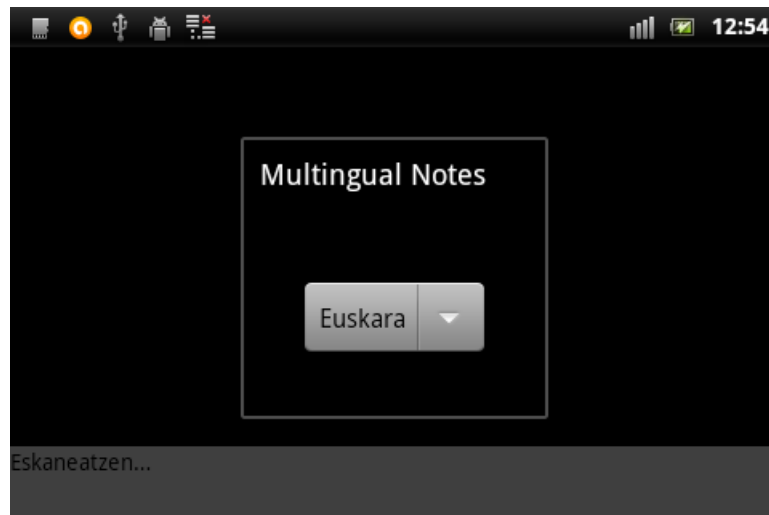
Aplikazioaren hizkuntza aldatu

1. Aplikazioaren hasierako pantailan zaudela sakatu aukerak ikusteari. Irekiko zaizun menuan aukeratu ezazu *Hizkuntzak* aukera.



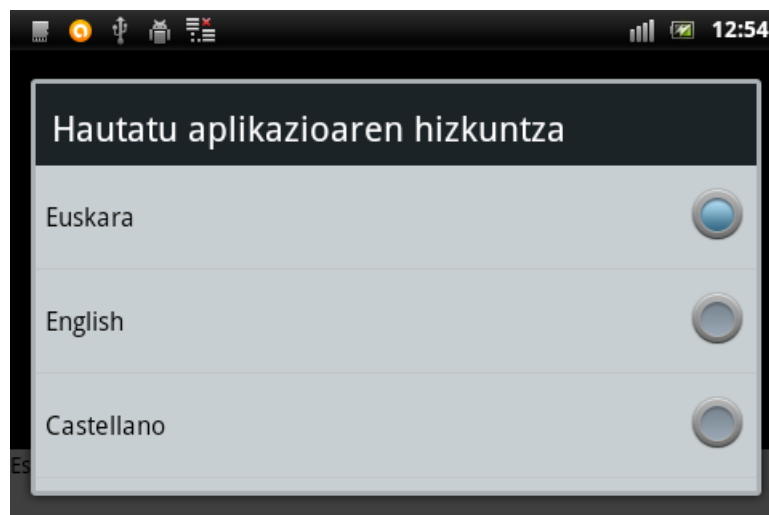
B.9 Irudia: *Ohar Eleanitzak* aplikazioaren aukeren barra *Hizkuntzak* aukera irekitzeko.

2. Irekiko zaizun pantailan hautatzaile bat agertuko zaizu.



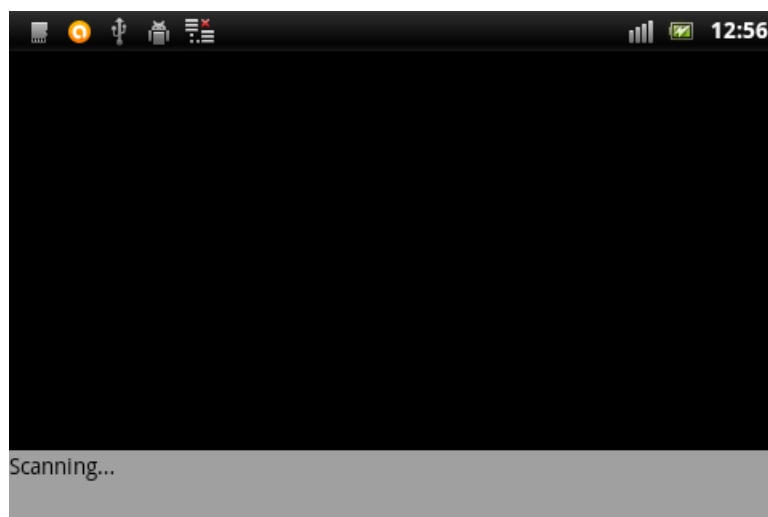
B.10 Irudia: Hizkuntzak aukeratzeko hautatzailea.

3. Sakatu hautatzailean eta aplikazioaren hizkuntzak zerrendatuko zaizkizu. Hautatu bertan nahi duzun hizkuntza.



B.11 Irudia: Aplikazioak dauzkan hizkuntzen zerrenda.

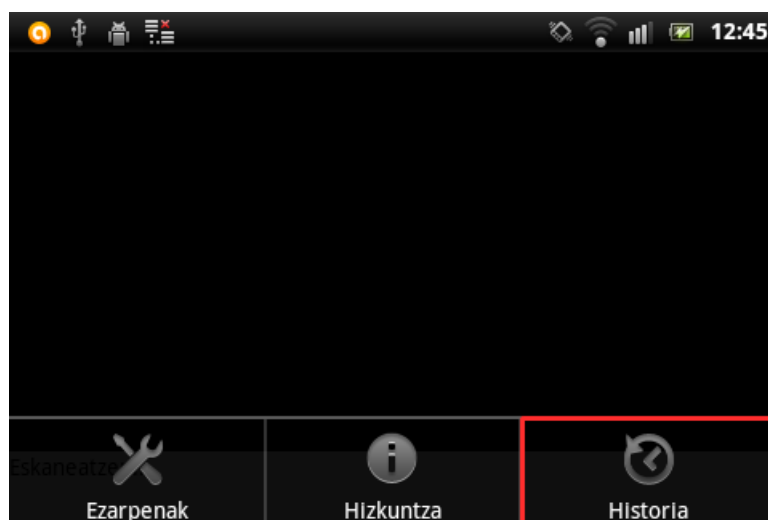
4. Aplikazioaren hasierako pantailara joaten zarenean aplikazioa zuk hautatutako hizkuntzan egongo da.



B.12 Irudia: Aplikazioaren hasierako pantaila, aplikazioaren hizkuntza ingelesera aldatu ostean.

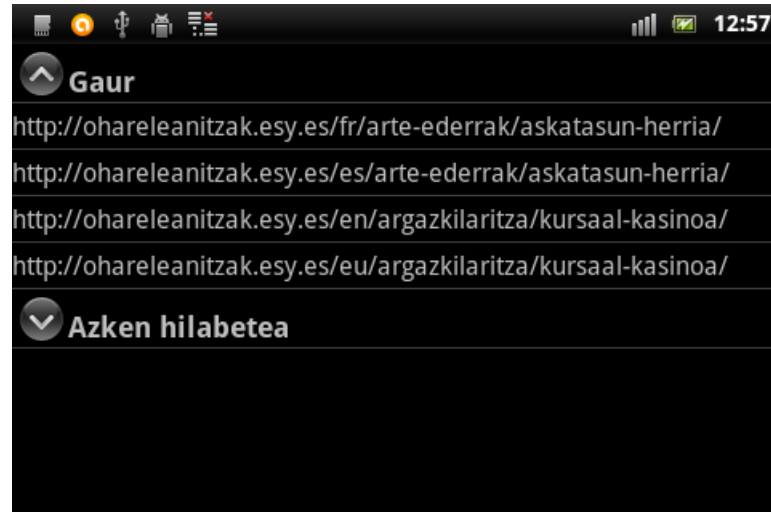
Irakurketa historia ikusi

1. Aplikazioaren hasierako pantailan zaudela sakatu aukerak ikusteari. Irekiko zaizun menuan aukeratu ezazu *Historia* aukera.



B.13 Irudia: Ohar Eleanitzak aplikazioaren aukeren barra *Historia* aukera irekitzeko.

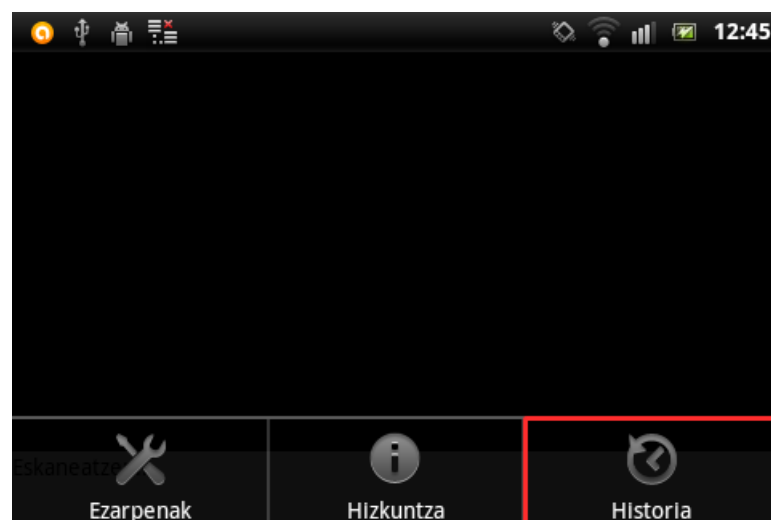
2. Irekiko zaizun pantailan zure irakurketa historia azalduko zaizu. Historiako elementu bat arakatzailan ikusi nahi baduzu sakatu ezazu elementu horren gainean.



B.14 Irudia: Erabiltzailearen irakurketa historia.

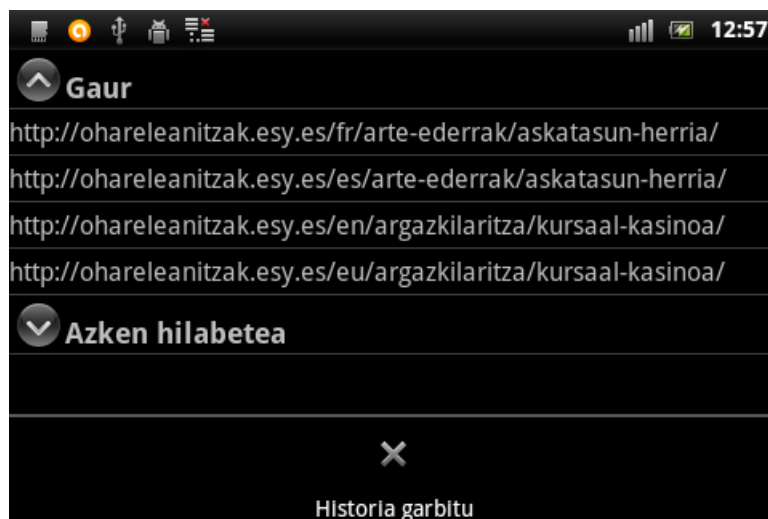
Irakurketa historia garbitu

1. Aplikazioaren hasierako pantailan zaudela sakatu aukerak ikusteari. Irekiko zaizun menuan aukeratu ezazu *Historia* aukera.



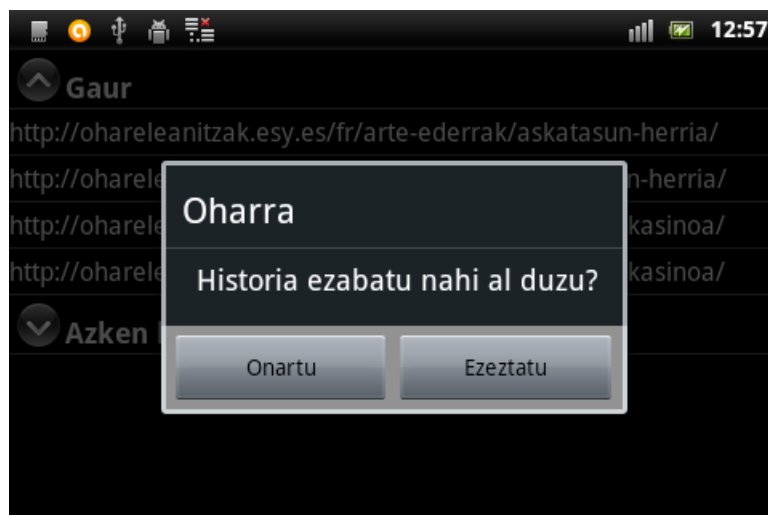
B.15 Irudia: *Ohar Eleanitzak* aplikazioaren aukeren barra *Historia* aukera irekitzeko.

2. Irekiko zaizun pantaila berrian eman aukerak ikusteari. Bertan agertzen den *Historia garbitu* aukerari eman historia garbitu nahi baduzu.



B.16 Irudia: *Historia garbitu* aukera erakusten duen aukeren barra.

3. Konfirmazio mezu bat agertuko zaizu. Ezezkoa esatekotan historia erakusten duen pantailara itzuliko zara. Baiezkoa esatekotan, berriz, historia garbitu egingo da.



B.17 Irudia: Historia garbitu aurretik agertzen den oharra.



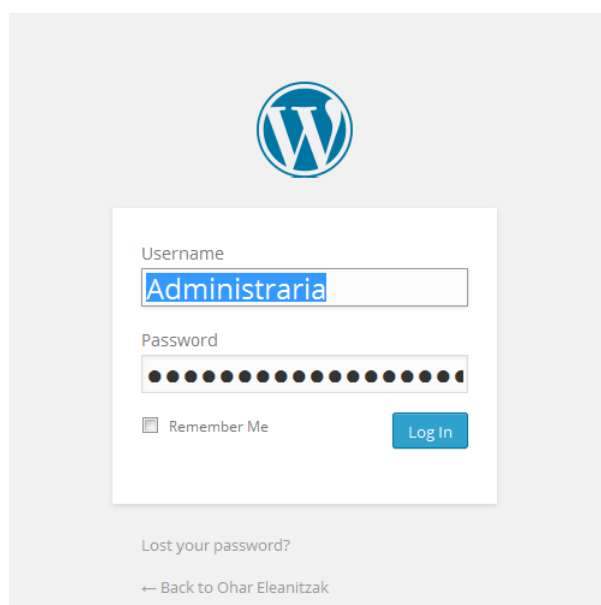
B.18 Irudia: Erabiltzailearen irakurketa historia garbitu ostean, historia hutsik.

B.2 Eduki Kudeatzailea

Atal honetan *WordPress* bidez garatutako Eduki Kudeatzailearen erabilera azalduko da. *WordPress* Edukiak Kudeatzeko Sistemak funtzionalitate ezberdin asko dauzkanez gida honetan Eduki Kudeatzailearen funtzioetarako erabili direnak soilik azalduko ditugu. Horretaz gain, gida honetan *Ohar Itzultzailea* gehigarriaren bitartez lortutako funtzionalitateak ere azalduko dira.

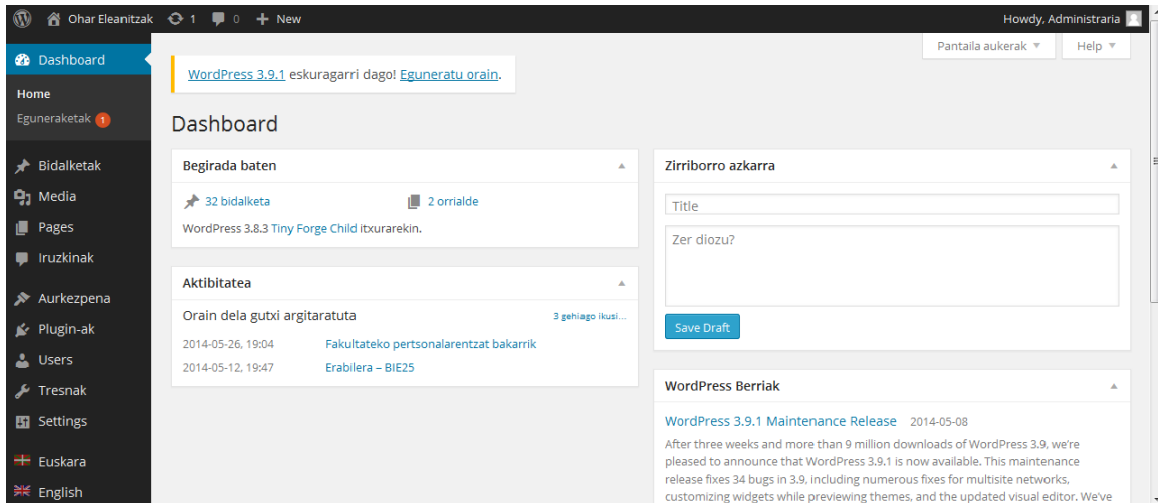
Eduki Kudeatzailean saioa hasi

1. Arakatzaillean zaudela idatz ezazu <http://ohareleanitzak.esy.es/wp-admin/> arakatzaillean helbide eremuan. Saioa hasteko orrialdea irekiko zaizu. Sartu zure erabiltzaile kontuaren izena eta pasahitza eta ondoren eman saioa hasteari.



B.19 Irudia: Eduki Kudeatzailean saioa hasteko orrialdea.

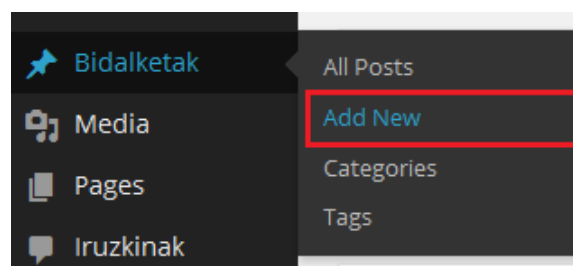
2. Sartu dituzun erabiltzaile kontuaren datuak egokiak badira, gunearen kudeaketa egi-teko orri nagusira eramango zaitu.



B.20 Irudia: Eduki Kudeatzailearen kudeaketarako hasiera orrialdea.

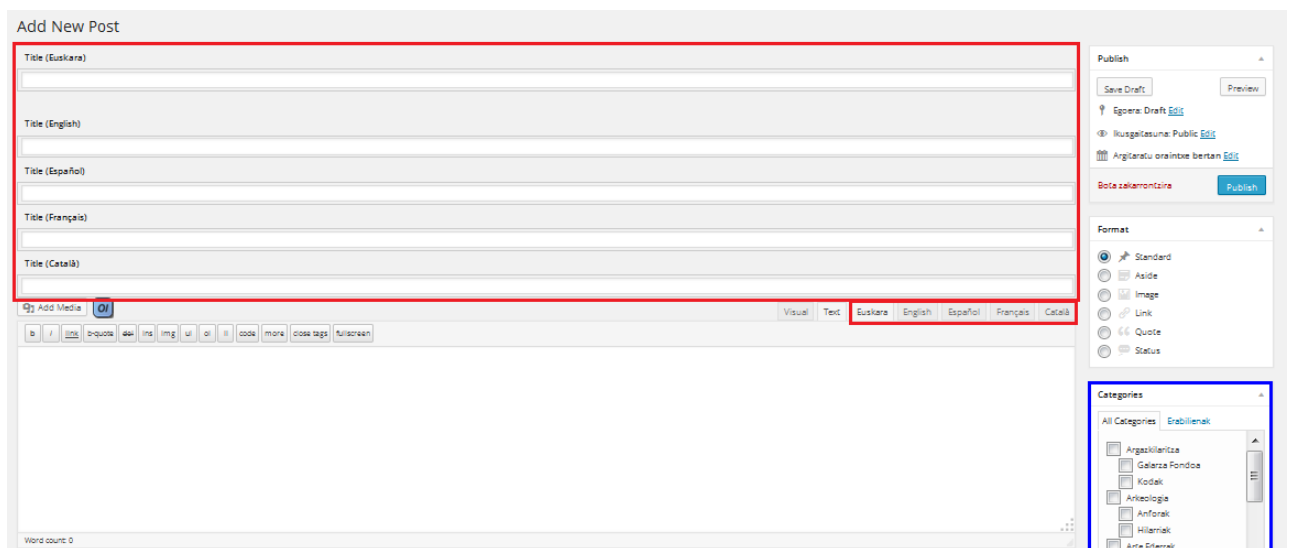
Bidalketa berri bat sortu

1. Eduki Kudeatzailearen hasiera orrian zaudela jarri sagua ezkerreko zutabearen *Bidalketak* aukeran eta bertatik sortuko den menuan klikatu *Add new* esaten duen tokian.



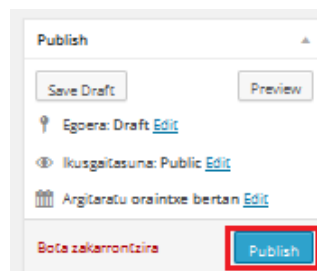
B.21 Irudia: Eduki Kudeatzailean Bidalketa berri bat sortzeko aukera.

2. Bidalketa berri bat sortzeko editorea irekiko zaizu. Bertan idatzi nahi duzun edukia, nahi duzun hizkuntzatan. Hizkuntz ezberdinetan idazteko, hizkuntz bakoitzaren izenburuak testu-kutxa bat dauka orrialdearen goikaldean (gorriz inguratuta agertzen da), *Add New Post* jartzen duen tokiaren azpian. Bidalketaren edukia beste hizkuntza batean idazteko editorearen goikaldean hizkuntz bakoitzeko fitxa bat dago (gorriz inguratuta agertzen da). Azkenik, aukeratu zein *Kategoria* esleitu nahi dizkion Bidalketari, eskuinean eta beheko partean azaltzen den *Categories* ataleko *Kategoriak* hautatuz (urdinez inguratuta dago).



B.22 Irudia: Eduki Kudeatzailean Bidalketa berri bat sortzeko editorea.

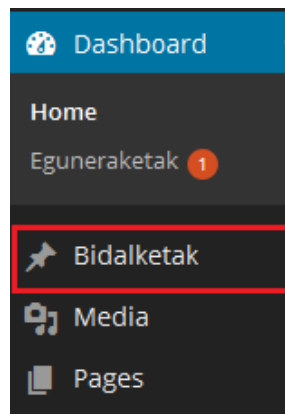
3. Edukia eta *Kategoria* gehitzeaz bukatzean eskuinean eta goikaldean dagoen *Publish* ataleko *Publish* botoia sakatu. Hori egitean Bidalketa berria sortuko da.



B.23 Irudia: Eduki Kudeatzailean Bidalketa berri bat publikatzeko sakatu beharreko botoia.

Bidalketa bat eraldatu

1. Eduki Kudeatzailearen hasiera orrian zaudela klikatu ezkerreko zutabean dagoen *Bidalketak* aukera Eduki Kudeatzailean dauden Bidalketen zerrenda ikusteko.



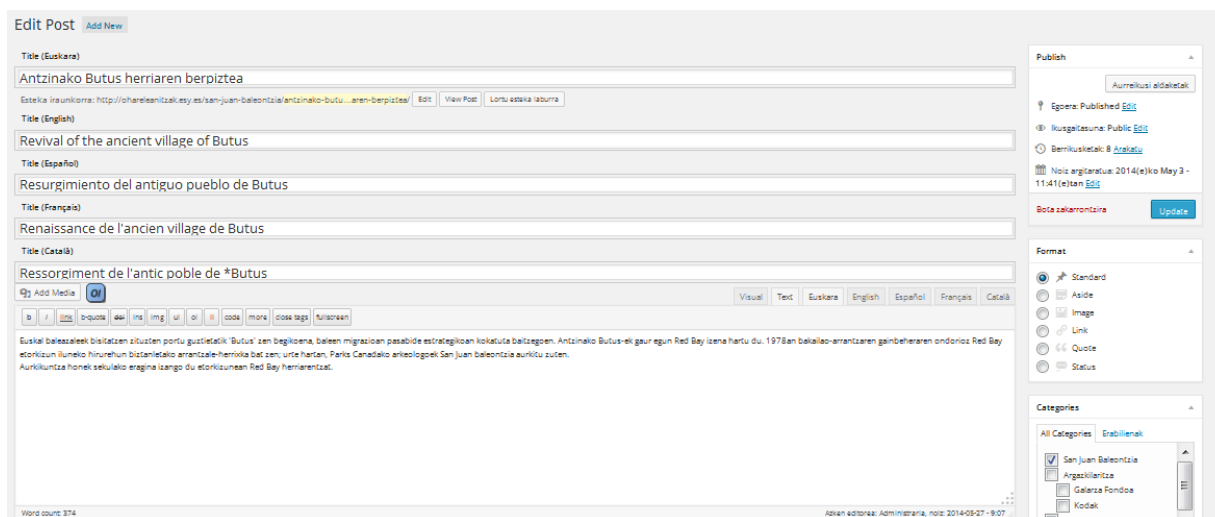
B.24 Irudia: Eduki Kudeatzailean dauden Bidalketak ikusteko sakatu beharreko botoia.

2. Eduki Kudeatzailean dauden Bidalketa guztien zerrenda agertuko zaizu. Eraldatu nahi duzun Bidalketaren izenburua sakatu (gorriz inguratuta daude) Bidalketa eraldatzeko.

izenburua	Author	Categories	Tags	Languages	QR Kodea	Date
Fakultateko personalarentzat sakarrak	Ipea Sarasola	DIF	—	English, Español, Français, Català		2014/05/26 Published
Erabilera - BIE25	Ipea Sarasola	DIF	—	Euskara, English, Español, Français, Català		2014/05/12 Published
Informatika Fakultatearen ordutegia	Ipea Sarasola	DIF	—	Euskara, English, Español, Français, Català		2014/05/12 Published
Mazkaritzako Oharra:	Ipea Sarasola	DIF	—	Euskara, English, Español, Français, Català		2014/05/12 Published
Baleontziaren indusketa prozesua	Administratzailea	San Juan Baleontzia	—	Euskara, English, Español, Français, Català		2014/05/03 Published
Antzinako Butus herriaren berripitua	Administratzailea	San Juan Baleontzia	—	Euskara, English, Español, Français, Català		2014/05/03 Published
Kargaren berreskuratzeari eta sortutako estabaidak	Administratzailea	San Juan Baleontzia	—	Euskara, English, Español, Français		2014/05/03 Published
Baleontziaren eraikitze eta hondoratzea	Administratzailea	San Juan Baleontzia	—	Euskara, English, Español, Français		2014/05/03 Published

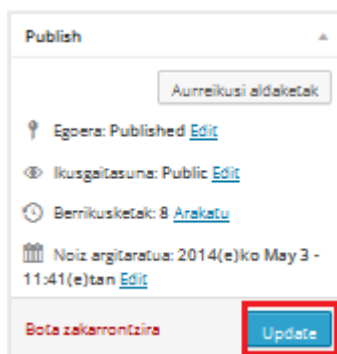
B.25 Irudia: Gunean dauden Bidalketen zerrenda erakusten duen irudia.

3. Hautatutako Bidalketa editorean irekiko zaizu. Orain Bidalketari nahi bezain beste aldaketak egin dizkiokezu: edukia gehitu/kendu, Kategoriak gehitu/kendu. . .



B.26 Irudia: Eraldatu nahi den Bidalketa editorean erakusten duen irudia.

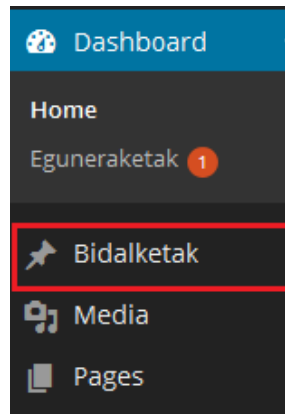
4. Aldaketak bukatzen dituzunean eman *Publish* ataleko *Update* atalari. Hori egitean Bidalketa eguneratu egingo da.



B.27 Irudia: Bidalketa eguneratzeko sakatu beharreko botoia.

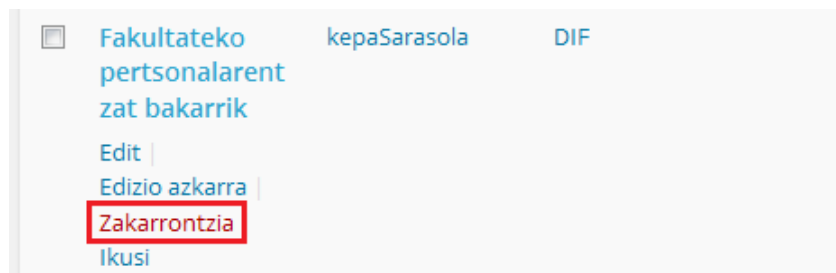
Bidalketa bat ezabatu

1. Eduki Kudeatzailearen hasiera orrian zaudela klikatu ezkerreko zutabean dagoen *Bidalketak* aukera Eduki Kudeatzailean dauden Bidalketen zerrenda ikusteko.



B.28 Irudia: Eduki Kudeatzailean dauden Bidalketak ikusteko sakatu beharreko botoia.

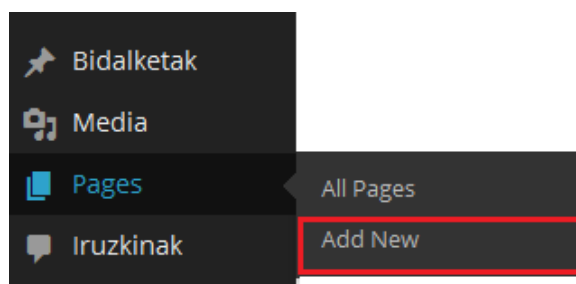
2. Agertzen zaizun zerrendan bilatu ezabatu nahiko zenukeen Bidalketa. Bidalketa ezabatzeko kokatu sagua Bidalketaren izenburuaren gainean eta agertuko zaizkizun aukeretatik sakatu *Zakarrontzia* esaten duena. Horrela Bidalketa zakarrontzira bidaliko da eta ez da atzigarri egongo.



B.29 Irudia: Bidalketa ezabatzeko sakatu beharreko botoia.

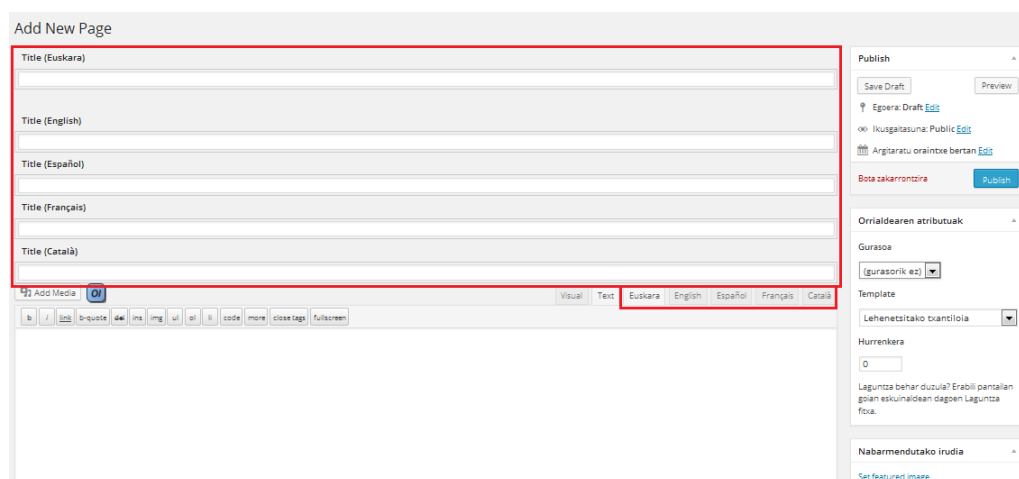
Orrialde berri bat sortu

1. Eduki Kudeatzailearen hasiera orrian zaudela jarri sagua ezkerreko zutabearen *Pages* aukeran eta bertatik sortuko den menuan klikatu *Add new* esaten duen tokian.



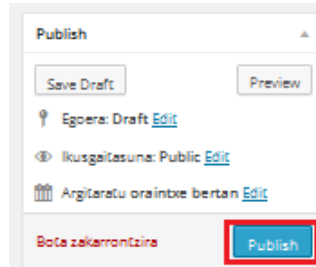
B.30 Irudia: Eduki Kudeatzailean Orrialde berri bat sortzeko aukera.

2. Orrialde berri bat sortzeko editorea irekiko zaizu. Bertan idatzi nahi duzun edukia, nahi duzun hizkuntzatan. Hizkuntz ezberdinetan idazteko, hizkuntz bakoitzaren izenburuak testu-kutxa bat dauka orrialdearen goikaldean (gorriz inguratuta agertzen da), *Add New Page* jartzen duen tokiaren azpian. Bidalketaren edukia beste hizkuntza batean idazteko editorearen goikaldean hizkuntz bakoitzeko fitxa bat dago (gorriz inguratuta agertzen da).



B.31 Irudia: Eduki Kudeatzailean Orrialde berri bat sortzeko editorea.

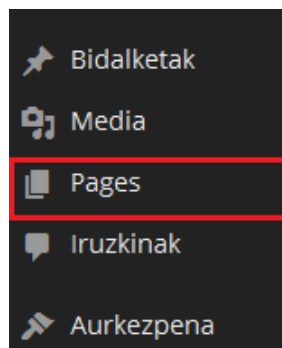
3. Edukia gehitzeaz bukatzean eskuinean eta goikaldean dagoen *Publish* ataleko *Publish* botoia sakatu. Hori egitean Orrialde berria sortuko da.



B.32 Irudia: Eduki Kudeatzailean Orrialde berri bat publikatzeko sakatu beharreko botoia.

Orrialde bat eraldatu

1. Eduki Kudeatzailearen hasiera orrian zaudela klikatu ezkerreko zutabearen dagoen *Pages* aukera Eduki Kudeatzailean dauden Orrialdeen zerrenda ikusteko.



B.33 Irudia: Eduki Kudeatzailean dauden Orrialdeak ikusteko sakatu beharreko botoia.

2. Eduki Kudeatzailean dauden Orrialde guztien zerrenda agertuko zaizu. Eraldatu nahi duzun Orrialdearen izenburua sakatu (gorriz inguratuta daude) Orrialdea eraldatzeko.

<input type="checkbox"/>	Izenburua	Author	Languages	QR Kodea	Data
<input type="checkbox"/>	Honi Buruz	Administraria	Euskara, English, Español, Français, Català		2014/04/03 Published
<input type="checkbox"/>	Ongi etorri!	Administraria	Euskara, English, Español, Français, Català		2014/04/01 Published
<input type="checkbox"/>	Izenburua	Author	Languages	QR Kodea	Data

B.34 Irudia: Gunean dauden Orrialdeen zerrenda erakusten duen irudia.

3. Hautatutako Orrialdea editorean irekiko zaizu. Orain Orrialdeari nahi bezain beste aldaketak egin ahal dizkiokezu.

Edit Page Add New

Title (Euskara)
Honi Buruz

Esteka irauinkorra: <http://ohare/eanitzakesy.es/honi-buruz/> Edit View Page Lortu esteka laburra

Title (English)
About

Title (Español)
Acerca de

Title (Français)
Sur

Title (Català)
Sobre

Add Media

Visual Text Euskara English Español Français Català

Manex Garaiola Mendizabal dut izena eta webgune hau nire Gradu Amaierako Proiektuaren zati bat da. Nire GAP-aren helburua honakoa da:

 <i></i>
 <i></i>
 <i></i>

 Espero dut gustoko izateal :D

Publish Aurreikusitako aldaketak

Egoera: Published [Edit](#)

Ikusgaitasuna: Public [Edit](#)

Berrikusketak: 11 [Arakatu](#)

Noiz argitaratua: 2014(e)ko Apr 3 - 17:25(etan) [Edit](#)

Bota zakarrontzira [Update](#)

Orrialdearen atributuak

Gurasoa
(gurasorik ez)

Template
Lehenetsitako txantiloia

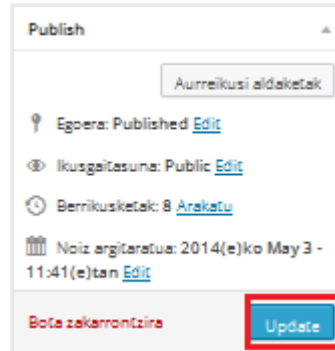
Hurrenkera
0

Laguntza behar duzula? Erabili pantailan golan eskuinaldean dagoen Laguntza fitxa.

Nabarmendutako irudia

B.35 Irudia: Eraldatu nahi den Orrialdea editorean erakusten duen irudia.

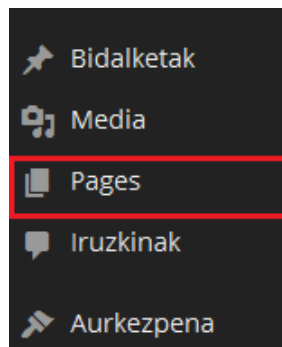
4. Aldaketak bukatzen dituzunean eman *Publish* ataleko *Update* atalari. Hori egitean Orrialdea eguneratu egingo da.



B.36 Irudia: Orrialdea eguneratzeko sakatu beharreko botoia.

Orrialde bat ezabatu

1. Eduki Kudeatzailearen hasiera orrian zaudela klikatu ezkerreko zutabearen dagoen *Pages* aukera Eduki Kudeatzailean dauden Orrialdeen zerrenda ikusteko.



B.37 Irudia: Eduki Kudeatzailean dauden Orrialdeak ikusteko sakatu beharreko botoia.

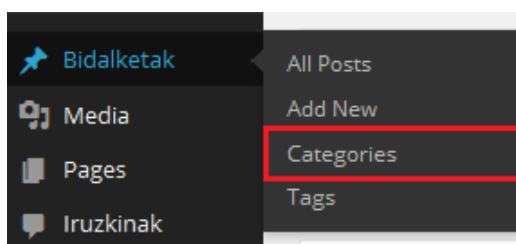
2. Agertzen zaizun zerrendan bilatu ezabatu nahiko zenukeen Orrialdea. Orrialdea ezabatzeko kokatu sagua Orrialdearen izenburuaren gainean eta agertuko zaizkizun aukeretatik sakatu *Zakarrontzia* esaten duena. Horrela Orrialdea zakarrontzira bidaliko da eta ez da atzigarri egongo.



B.38 Irudia: Orrialdea ezabatzeko sakatu beharreko botoia.

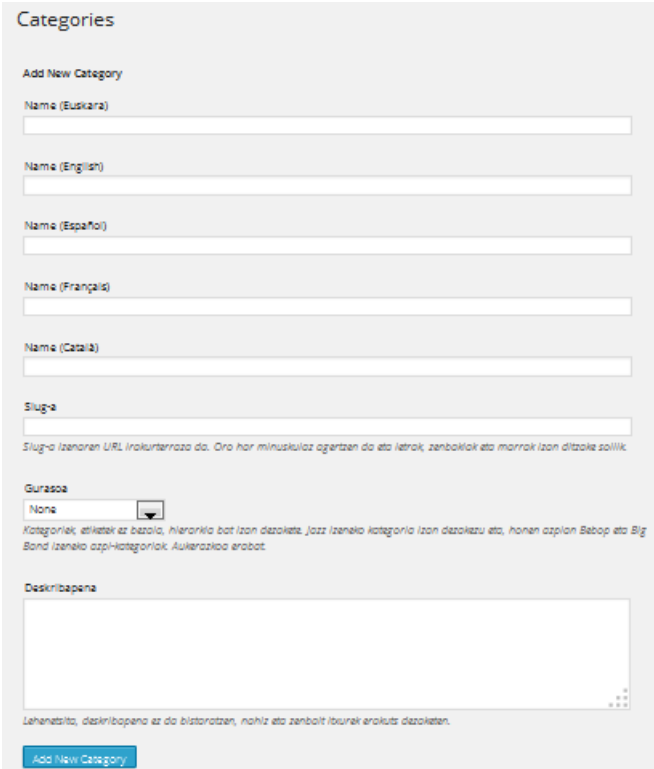
Kategoria sortu

1. Eduki Kudeatzailearen hasiera orrian zaudela jarri sagua ezkerreko zutabearen *Bidalketak* aukeran eta bertatik sortuko den menuan klikatu *Categories* esaten duen tokian.



B.39 Irudia: Eduki Kudeatzailean Kategoriak ikusteko atalera joateko.

2. Irekiko zaizun orrialdearen ezker aldean inprimaki bat azalduko zaizu. Bertan, jarri sortu nahi duzun Kategoriaren izena hizkuntz ezberdinetan. Horretaz gain, *slug-a* jartzen duen tokian idatzi Kategoriaren izena, URL-ean agertzea nahiko zenukeen bezala. Azkenik, *Gurasoa* atalean hautatu *Kategoria* bat, zuk sortu nahi duzun *Kategoria* beste horren menpekoa bada. Datuak betetzean klikatu *Add New Category* *Kategoria* gordetzeko.



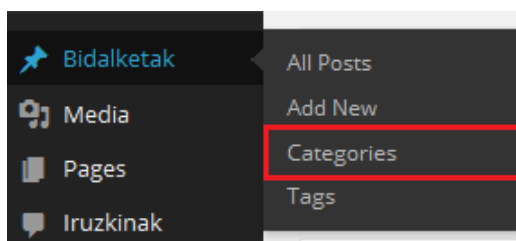
The image shows a web form titled "Categories" for adding a new category. It includes the following elements:

- Add New Category** header.
- Five text input fields for the category name in different languages: Name (Euskara), Name (English), Name (Español), Name (Français), and Name (Català).
- A text input field for the **Slug**, with a note below it: "Slug-a izenaren URL irakurterraz da. Ora har minuskulaz agertzen da eta letrak, zenbakiak eta marrikak izan ditzake soilik."
- A **Gurasoa** (Parent) dropdown menu currently set to "None", with a note: "Kategoriak etiketak ez bezala, hierarkia bat izan dezakete. Jazz izeneko kategoria izan dezakete eta, honen azpian Sebop eta Big Band izeneko azpi-kategoriak aukeratzeko erabot."
- A large text area for the **Deskribapena** (Description), with a note: "Lehenetsita, deskribapena ez da bistarazten, nahiz eta zenbait itxurek erakutsi dezaketen."
- A blue **Add New Category** button at the bottom.

B.40 Irudia: Kategori berri bat sortzeko erabiltzen inprimakia.

Kategoria eraldatu

1. Eduki Kudeatzailearen hasiera orrian zaudela jarri sagua ezkerreko zutabearen *Bidalketak* aukeran eta bertatik sortuko den menuan klikatu *Categories* esaten duen tokian.



B.41 Irudia: Eduki Kudeatzailean Kategoriak ikusteko atalera joateko.

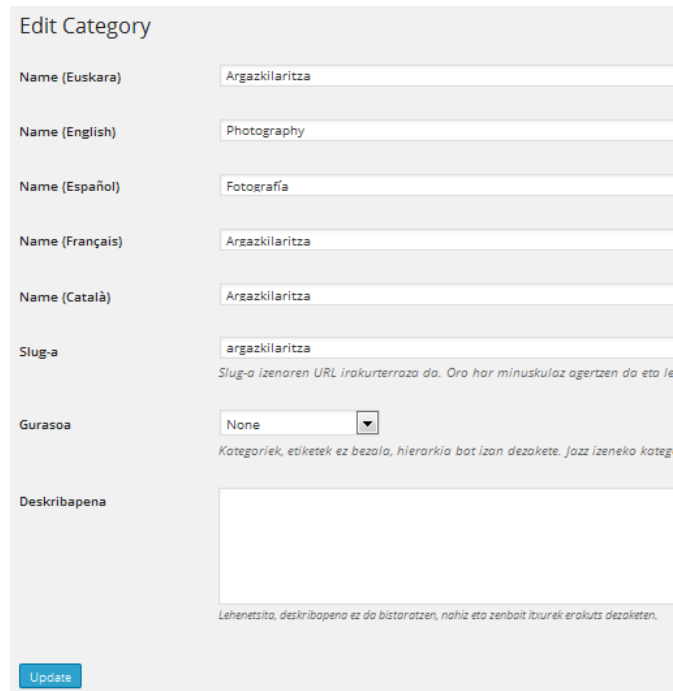
2. Eduki Kudeatzailean dauden Kategoria guztien zerrenda agertuko zaizu orrialdearen eskuinaldean. Eraldatu nahi duzun Kategoriaren izenburua sakatu (gorriz inguratuta daude) Kategoria eraldatzeko.

 A screenshot of the WordPress 'Categories' management screen. At the top, there is a search bar labeled 'Search Categories' and a '2 elementu' indicator. Below is a table with columns: 'Izena', 'Description', 'Sluga', and 'Posts'. The table contains 15 rows of categories. A red rectangular box highlights the 'Izena' column. At the bottom, there is a 'Multitokako elementuak' section with an 'Apply' button and another '2 elementu' indicator.

Izena	Description	Sluga	Posts
Argazkilaritza		argazkilaritza	8
Galeria Fonda		galeria-fonda	5
Kodak		kodak	3
Arkeologia		arkeologia	5
Anforak		anforak	2
Hilarrak		hilarrak	3
Arte Ederrak		arte-ederrak	11
Euskal Artea		euskal-arte	6
Sert-en Mihiseak		sert-en-mihiseak	5
DIF		dif	4
Salikatu gabea		salikatugabeak	0
San Juan Baiantzak		san-juan-baiantzak	4

B.42 Irudia: Gunean dauden Kategorien zerrenda erakusten duen irudia.

3. Hautatu duzun Kategoriaren datuak azalduko zaizkizu irekitako orrialdean. Egin interesatzen zaizkizun aldaketak eta bukatzean sakatu *Update* Kategoria eguneratzeko.



Edit Category

Name (Euskara) Argazkilaritza

Name (English) Photography

Name (Español) Fotografía

Name (Français) Argazkilaritza

Name (Català) Argazkilaritza

Slug-a argazkilaritza
Slug-a izenaren URL irakurterraza da. Oro har minuskulaz agertzen da eta letrak zuriak dira.

Gurasoa None

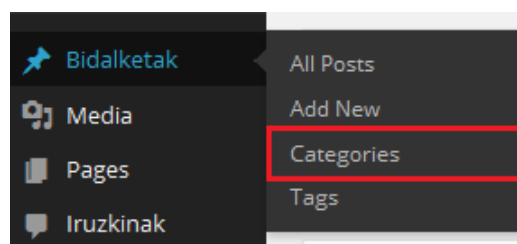
Deskribapena

[Update](#)

B.43 Irudia: Eraldatu nahi den Kategoriaren datuak erakusten dituen orrialdea.

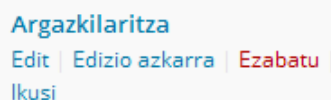
Kategoria ezabatu

1. Eduki Kudeatzailearen hasiera orrian zaudela jarri sagua ezkerreko zutabearen *Bidalketak* aukeran eta bertatik sortuko den menuan klikatu *Categories* esaten duen tokian.



B.44 Irudia: Eduki Kudeatzailean Kategoriak ikusteko atalera joateko.

2. Agertzen zaizun zerrendan bilatu ezabatu nahiko zenukeen Kategoria. Kategoria ezabatzeko kokatu sagua Kategoriaren izenburuaren gainean eta agertuko zaizkizun aukeretatik sakatu *Ezabatu* esaten duena. Horrela Kategoria ezabatu egingo da eta ez da atzigarri egongo.



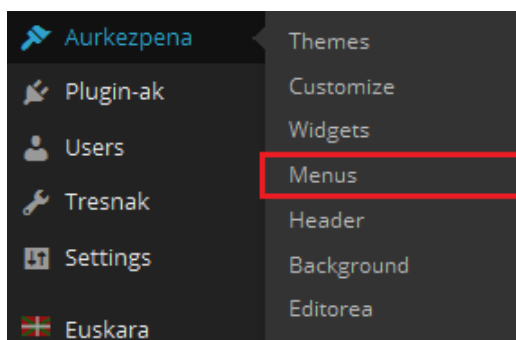
Argazkilaritza
Edit | Edizio azkarra | Ezabatu
Ikusi

B.45 Irudia: Kategoria ezabatzeko sakatu beharreko botoia.

Eduki Kudeatzailearen egitura sortu

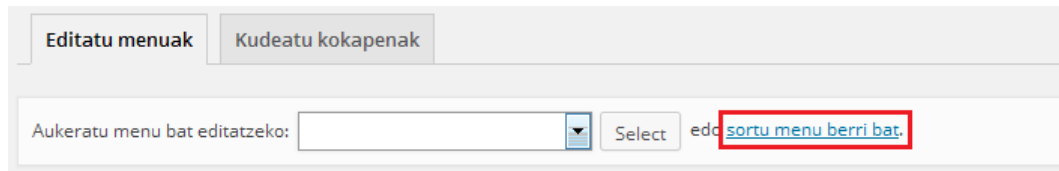
Eduki Kudeatzaileak aukera dauka goiburuaren azpian aukera barra bat jartzeko, edukietara modu azkarragoan iritsi ahal izateko. Aukera barra horretan Kategoriak eta Orrialdeak kokatu daitezke. Orrialdeen bidez informazio estatikoa erakutsiko da (sarrera orrialdea, honi buruz atala. . .) eta Kategoriekin berriz Kategoria horietara esleituta dauden Bidalaketak erakutsiko dira.

1. Eduki Kudeatzailearen hasiera orrian zaudela jarri sagua ezkerreko zutabearen *Aurkezpena* aukeran eta bertatik sortuko den menuan klikatu *Menus* esaten duen tokian.



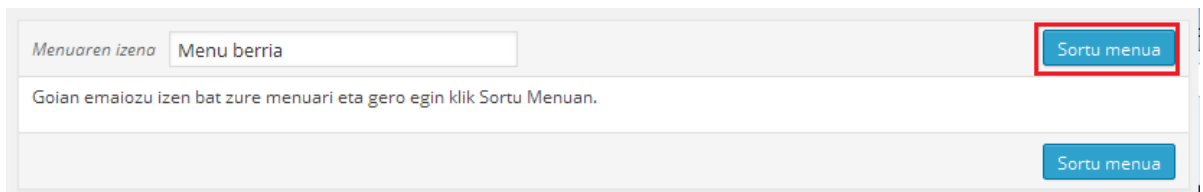
B.46 Irudia: Eduki Kudeatzailean egitura sortzeko atalera joateko.

- Irekiko zaizun orrialdearen goikaldean *sortu menu berri bat* deituriko esteka daukazu. Sakatu ezazu menu berria sortzeko.



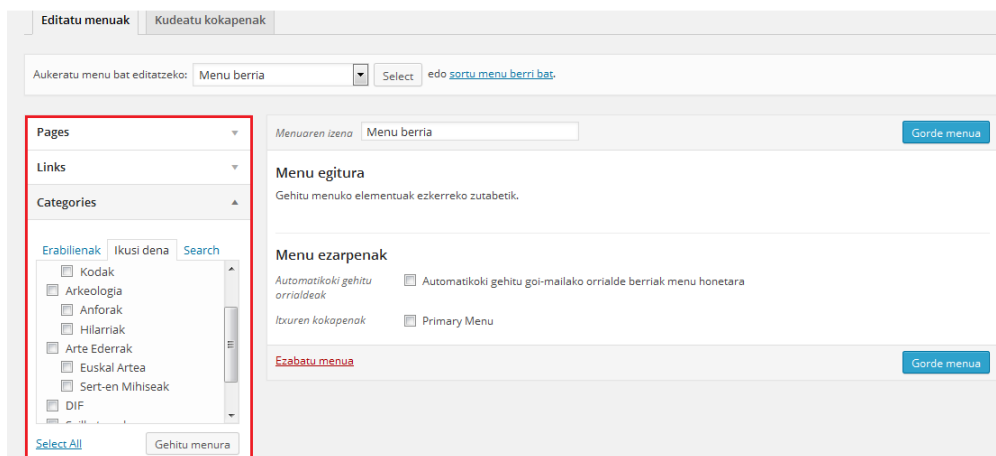
B.47 Irudia: Menu berri bat sortzeko sakatu beharreko esteka.

- Agertuko zaizun orrialdean menu berriari izena eman beharko diozu *Menuaren izena* atalean. Hori egin ostean, emaitza *Sortu menua* aukerari.



B.48 Irudia: Menu berriari izena emateko orrialdea.

- Orrialdea osatuko zaizu eta bertan menuaren edukia eta kokapena zehazteko aukera izango duzu. Hautatu gorritz inguratuta dagoen ataletik menuari gehitu nahi dizkionzun Orrialde eta Kategoriak. Hautatu ondoren eman atal horretan bertan dagoen *Gehitu menura* aukerari.



B.49 Irudia: Menu berriaren edukia eta kokapena zehazteko orrialdea.

5. Hautatu dituzun edukiak *Menu egitura* atalean agertuko zaizkizu, zerrendatuta. Horiek modu hierarkiko batean antolatzeko klikatu saguarekin mugitu nahi duzun elementua eta kokatu beste elementuen menpeko bezala, gorago, beherago... Horrekin bukatzean hautatu *Itxuren kokapenak* ataleko *Primary menu* kutxa menua Eduki Kudeatzaileari gehitzeko. Azkenik, aldaketak gordetzeko sakatu *Gorde menua* aukera.

Menu egitura

Arrastatu elementuak nahi dituzun ordenean. Egin klik elementuaren eskuineko gezia ezarpen gehigarriak ikusteko.

Arte Ederrak	Category ▼
Euskal Artea <i>elementu atala</i>	Category ▼
Sert-en Mihiseak <i>elementu atala</i>	Category ▼
Argazkilaritza	Category ▼
Galarza Fondoa <i>elementu atala</i>	Category ▼
Kodak <i>elementu atala</i>	Category ▼
San Juan Baleontzia	Category ▼

Menu ezarpenak

Automatikoki gehitu orrialdeak Automatikoki gehitu goi-mailako orrialde berriak menu honetara

Itxuren kokapenak Primary Menu (Unekoa Goleko barra da)

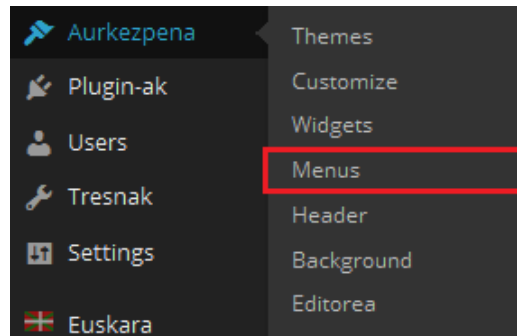
[Ezabatu menua](#)

[Gorde menua](#)

B.50 Irudia: Menu berria, edukia gehitu ondoren.

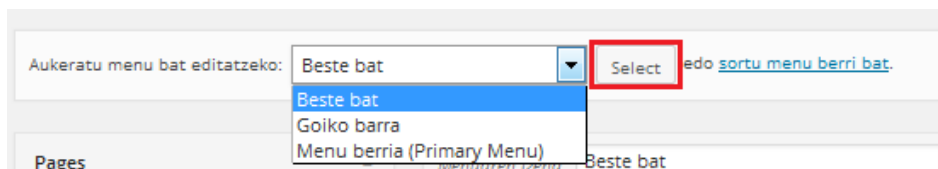
Eduki Kudeatzailearen egitura eraldatu

1. Eduki Kudeatzailearen hasiera orrian zaudela jarri sagua ezkerreko zutabearen *Aurkezpena* aukeran eta bertatik sortuko den menuan klikatu *Menus* esaten duen tokian.



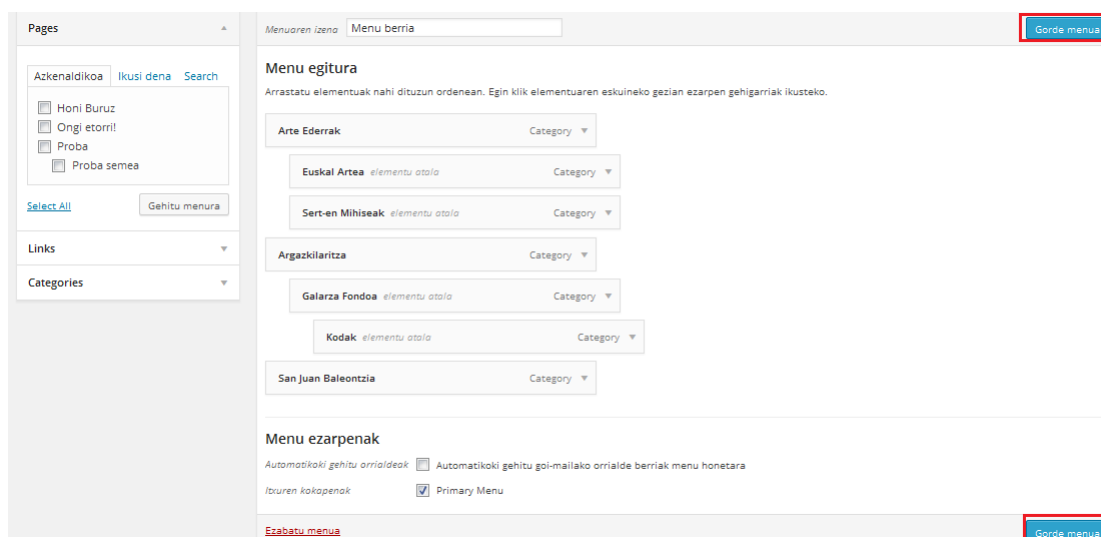
B.51 Irudia: Eduki Kudeatzailean egitura sortzeko atalera joateko.

2. Irekiko zaizun orrialdearen goikaldean hautatzaile bat izango duzu. Bertatik hautatu eraldatu nahi duzun menua eta sakatu *Select*.



B.52 Irudia: Eraldatu nahi den menua aukeratzeko hautatzailea.

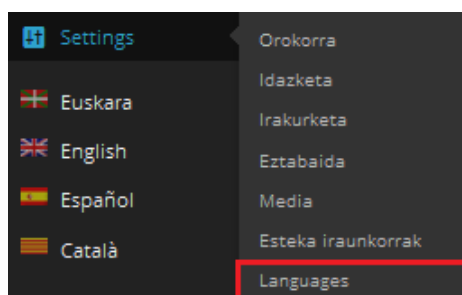
3. Hautatu duzun menuaren edukia eta informazioa azalduko zaizkizu. Egin nahi bezain beste aldaketa menuari (edukia gehitu/kendu, izena aldatu...) eta ondoren sakatu *Gorde menua* botoia menua eguneratzeko.



B.53 Irudia: Menuaren edukiak erakusten dituen irudia.

Hizkuntz bat definitu

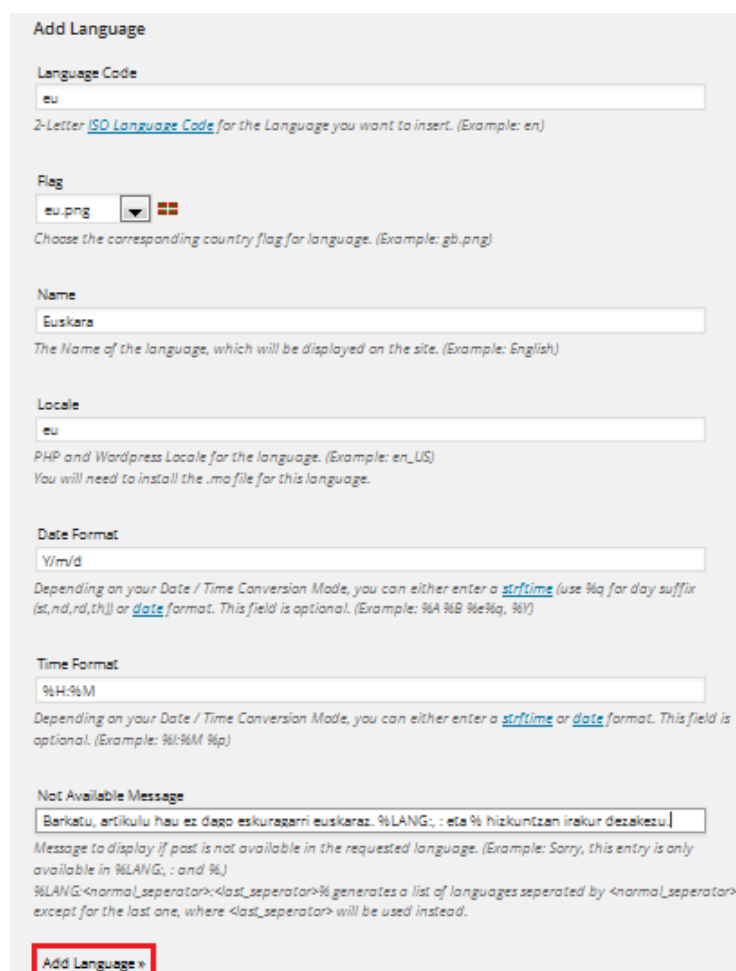
1. Eduki Kudeatzailearen hasiera orrian zaudela jarri sagua ezkerreko zutabearen *Settings* aukeran eta bertatik sortuko den menuan klikatu *Languages* esaten duen tokiari.



B.54 Irudia: Eduki Kudeatzailean hizkuntzen atalera joateko.


2. Irekiko zaizun orrialdean behera joaten bazara *Add language* esaten duen atal bat ikusiko duzu. Atal horretan dagoen informazioa bete behar da hizkuntz berri bat

definitzeko. *Language code* esaten duen tokian definitu nahi duzun hizkuntzaren bi letrako hizkuntz kodea (ISO-639-1 estandarrari dagokiona) idatzi beharko duzu. *Flag* jartzen duen tokian zuk gehitu nahi duzun hizkuntzari dagokion bandera aukeratu beharko duzu. *Name* eta *Locale* jartzen duen tokian, berriz, hizkuntzaren izena eta *WordPress*-ek hizkuntz horretarako erabiltzen duen hizkuntz kodea idatzi beharko dituzu, hurrenez hurren. Azken honetan ere hizkuntz-kode horri dagokion *WordPress*-en .mo fitxategiak igo beharko dituzu zerbitzarira. *Date format* eta *Time format* ataletan definitu nahi duzun hizkuntzan data eta ordua nola esaten diren definitu beharko duzu. Azkenik, *Not available message* atalean eduki bat hizkuntza horretan ez dagoenean azaldu beharreko mezua definitzen da. Datuak sartu dituzunean sakatu *Add language* atala hizkuntz berria gorde dadin.



Add Language

Language Code
eu
2-Letter [ISO Language Code](#) for the Language you want to insert. (Example: en)

Flag
eu.png 
Choose the corresponding country/flag for language. (Example: gb.png)

Name
Euskara
The Name of the language, which will be displayed on the site. (Example: English)

Locale
eu
PHP and Wordpress Locale for the language. (Example: en_US)
You will need to install the .mo file for this language.

Date Format
Y/m/d
Depending on your Date / Time Conversion Mode, you can either enter a [strftime](#) (use %a for day suffix (st,nd,rd,th)) or [date](#) format. This field is optional. (Example: %A %B %e%g, %Y)

Time Format
%H:%M
Depending on your Date / Time Conversion Mode, you can either enter a [strftime](#) or [date](#) format. This field is optional. (Example: %l:%M %p)

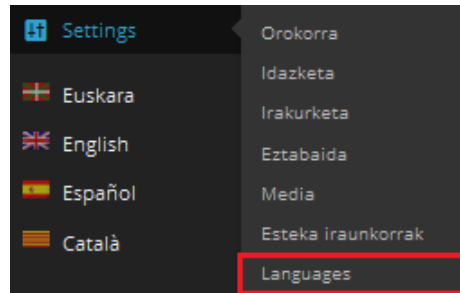
Not Available Message
Barkatu, artikulu hau ez dago eskurazemi euskaraz. %LANG,: eta % hizkuntzan irakur dezakezu.
Message to display if post is not available in the requested language. (Example: Sorry, this entry is only available in %LANG,: and %.)
%LANG:<normal_seperator><last_seperator>% generates a list of languages separated by <normal_seperator> except for the last one, where <last_seperator> will be used instead.

Add Language

B.55 Irudia: Eduki Kudeatzailean hizkuntza berri bat definitzeko atala.

Hizkuntz bat gaitu

1. Eduki Kudeatzailearen hasiera orrian zaudela jarri sagua ezkerreko zutabearen *Settings* aukeran eta bertatik sortuko den menuan klikatu *Languages* esaten duen to- kian.



B.56 Irudia: Eduki Kudeatzailean hizkuntzen atalera joateko.

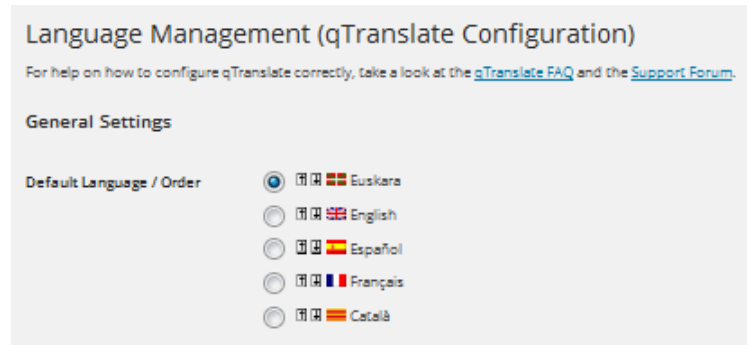
2. Irekiko zaizun orrialdean beheraka joaten bazara taula bat ikusiko duzu Eduki Ku- deatzailean definituta dauden hizkuntz guztiekin. Bilatu zuk gaitu nahi duzuna eta *Action* zutabeko *Enable* aukera sakatu hizkuntza gaitzeko (*Enable* ordez *Disable* jartzen badu hizkuntz hori gaituta dago jadanik).

Flag	Name	Action		
	Deutsch	Enable	Edit	Delete
	中文	Enable	Edit	Delete
	suomi	Enable	Edit	Delete
	Français	Enable	Edit	Delete
	Nederlands	Enable	Edit	Delete
	Svenska	Enable	Edit	Delete
	Italiano	Enable	Edit	Delete
	Română	Enable	Edit	Delete
	Magyar	Enable	Edit	Delete
	日本語	Enable	Edit	Delete
	Tiếng Việt	Enable	Edit	Delete
	العربية	Enable	Edit	Delete
	Português	Enable	Edit	Delete
	Polski	Enable	Edit	Delete
	galego	Enable	Edit	Delete
	Euskara	Disable	Edit	Delete
	English	Disable	Edit	Default
	Español	Disable	Edit	Delete
	Català	Disable	Edit	Delete
Flag	Name	Action		

Enabling a language will cause qTranslate to update the Gettext-Database for the language, which can take a while depending on your server's connection speed.

B.57 Irudia: Eduki Kudeatzailean definituta dauden hizkuntzen zerrenda.

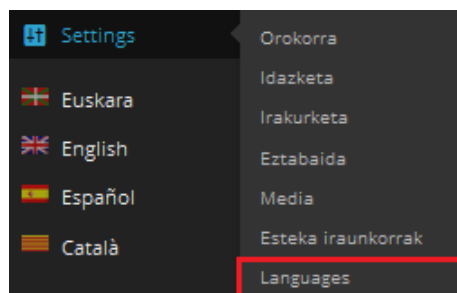
3. Orrialdea birkargatuko zaizu. Bukatzean, orrialdean goikaldean gaituta dauden hizkuntzen zerrenda ikusiko duzu. Gaitutako hizkuntzen ordena aldatu ahalko duzu hizkuntz bakoitzaren ondoan dauden geziak bitartez. Horretaz gain, gunearen hizkuntz lehenetsia hautatu ahalko duzu hizkuntzaren ondoko irati-botoia hautatuz.



B.58 Irudia: Eduki Kudeatzailean gaituta dauden hizkuntzen zerrenda.




















Hizkuntz bat ezgaitu

1. Eduki Kudeatzailearen hasiera orrian zaudela jarri sagua ezkerreko zutabearen *Settings* aukeran eta bertatik sortuko den menuan klikatu *Languages* esaten duen tokian.



B.59 Irudia: Eduki Kudeatzailean hizkuntzen atalera joateko.

2. Irekiko zaizun orrialdean beheraka joaten bazara taula bat ikusiko duzu Eduki Kudeatzailean definituta dauden hizkuntz guztiekin. Bilatu zuk ezgaitu nahi duzuna eta *Action* zutabeko *Disable* aukera sakatu hizkuntza ezgaitzeko (*Disable* ordez *Enable* jartzen badu hizkuntz hori ezgaituta dago jadanik).

Flag	Name	Action		
	Deutsch	Enable	Edit	Delete
	中文	Enable	Edit	Delete
	suomi	Enable	Edit	Delete
	Français	Enable	Edit	Delete
	Nederlands	Enable	Edit	Delete
	Svenska	Enable	Edit	Delete
	Italiano	Enable	Edit	Delete
	Română	Enable	Edit	Delete
	Magyar	Enable	Edit	Delete
	日本語	Enable	Edit	Delete
	Tiếng Việt	Enable	Edit	Delete
	العربية	Enable	Edit	Delete
	Português	Enable	Edit	Delete
	Polski	Enable	Edit	Delete
	galego	Enable	Edit	Delete
	Euskara	Disable	Edit	Delete
	English	Disable	Edit	Default
	Español	Disable	Edit	Delete
	Català	Disable	Edit	Delete

Enabling a language will cause qTranslate to update the Gettext-Database for the language, which can take a while depending on your server's connection speed.

B.60 Irudia: Eduki Kudeatzailean definituta dauden hizkuntzen zerrenda.

Itzulpenak egin

1. Orrialdeen zein Bidalketen editorean zaudela sakatu itzultzailea irekitzeko botoia. Gogora ezazu itzulpenek funtzionamendu optimoa izan dezaten ondorengo baldintzak behar direla: edukia eguneratuta egotea *Update* botoia erabilia, fitxen artean nabigaziorik ez gertatzea eta editoreko Text aukera hautatuta egotea.



B.61 Irudia: Editorearen irudia, itzultzailea irekitzeko botoia gorri nabarmenduta daukalarik.

2. Itzultzailea irekiko zaizu leiho berri batean. Lehenik eta behin, hautatu zer itzuli nahi duzun goikaldeko irati-botoiak erabiliz. Horren arabera testu-kutxan azalduko den testua aldatu egingo da, bidalketa/orrialdearen izenburua edota testua erakutsiz.

OharEleanitzak-Apertium itzultzailea

Testua itzuli
 Izenburua itzuli

Euskara ▼

Gaztelania ▼


Leihoa itxi

Antzinako Butus hemiaren berpiztas

Itzuli

Itzulpena gorde eta leihoa itxi

Itzulpenetarako erabilgaitako web-zerbitzua:



B.62 Irudia: Itzultzailearen irudia, izenburuak itzultzea aukeratuta dagoenean.

OharEleanitzak-Apertium itzultzailea

Testua itzuli
 Izenburua itzuli

Euskara ▼

Gaztelania ▼


Leihoa itxi

Euskal baleazaleek bisitatzen zituzten portu guztietatik "Butus" zen begikoena, baleen migrazioan pasabide estrategikoan kokatuta baitzegoen. Antzinako Butus-ek gaur egun Red Bay izena hartu du. 1978an bakailao-arrantzaren gainbeheraren ondorioz Red Bay etorkizun iluneko hirurehun biztanletako arrantzaile-herrikoa bat zen; urte hartan, Parks Canadako arkeologoek San Juan baleontzia aurkitu zuten. Aurkikuntza honek sekulako eragina izango du etorkizunean Red Bay hemiarentzat.

Itzuli

Itzulpena gorde eta leihoa itxi

Itzulpenetarako erabilgaitako web-zerbitzua:



B.63 Irudia: Itzultzailearen irudia, testuak itzultzea aukeratuta dagoenean.

- Ondoren, aukera ezazu zure itzulpenaren jatorri hizkuntza. Hizkuntzaz aldatzean testu-kutxan agertuko den testua eguneratu egingo da, hizkuntz horretako testua erakutsiz.

OharEleanitzak-Apertium itzultzailea

Testua itzuli Izenburua itzuli

Gaztelania ▼

De todos los puertos que visitaban los balleneros vascos 'Butus' era el preferido, ya que se situaba en un paso estratégico en la migración de la ballenas. El antiguo Butus es conocido hoy en día como Red Bay. En 1978 era una pequeña población de trescientos habitantes, de futuro incierto debido a la decadencia de la pesca del bacalao, fue ese año cuando los arqueólogos de Parks Canada encontraron el San Juan. Este descubrimiento tendría un impacto considerable en el futuro de Red Bay.

Itzuli

Frantsiera ▼

Leihoa itxi

Itzulpena gorde

Itzulpena gorde eta leihoa itxi

Itzulpenetarako erabilitako web-zerbitzua:



B.64 Irudia: Itzultzailearen irudia, jatorriko hizkuntza gisa gaztelania hautatuta daukanean.

4. Aukeratu ezazu zure itzulpenarentzako helburu hizkuntza eta eman *Itzuli* botoiari. Beheko testu-kutxan itzulpena agertuko zaizu. Itzulpena gorde nahi baduzu eman *Itzulpena gorde* (itzulpen gehiago egin nahi badituzu) edota *Itzulpena gorde eta leihoa itxi* (itzultzailea itxi nahi baldin baduzu) aukerei. Itzulpena gorde nahi ez baduzu, berriz, egin beste itzulpen bat hizkuntzak edo itzuli behar den testu-mota aldatuta edota eman *Leihoa itxi* aukerari itzultzailea ixteko.

OharEleanitzak-Apertium itzultzailea

Testua itzuli Izenburua itzuli

Gaztelania ▼

A pesar de las duras condiciones de trabajo a las que los submarinistas se veían sometidos, el trabajo de excavación del San Juan fue extraordinariamente minucioso. Se utilizaron técnicas que no habían sido aplicadas anteriormente en ninguna otra excavación. Junto con el hallazgo del pecio del San Juan por parte de los investigadores de Parks Canada, los arqueólogos de la Memorial University of Newfoundland, dirigidos por el profesor James A. Tuck, encontraron numerosos yacimientos en tierra firme, entre ellos unas 140 sepulturas de marineros vascos fallecidos allí a lo largo de los años. Debido a este conjunto de restos, Red Bay se ha convertido en un lugar de importancia mundial en el campo de la arqueología.

Itzuli

Frantziara ▼

Malgré les dures conditions de travail à celles que les *submarinistas se voyaient soumis, le travail d'excavation du San Juan il a été *extraordinairement *minucioso. Ils s'ont utilisés des techniques qu'ils n'avaient pas été appliquées antérieurement dans aucune autre excavation. Je joins avec la trouvaille du *pecio du San Juan par part des chercheurs de *Parks *Canada, les *arqueólogos de la Mémorial *University *of *Newfoundland, dirigés par le professeur James A. *Tuck. Ont trouvé des nombreux gisements en terroir ferme, entre ils quelques 140 sépultures de marins basques morts là tout au long des ans. En raison de cet ensemble de restos, Réseau *Bay s'est converti dans un lieu d'importance mondiale dans le champ de la *arqueología.

Leihoa itxi **Itzulpena gorde** **Itzulpena gorde eta leihoa itxi**

B.65 Irudia: Itzultzailearen irudia, gaztelaniatik frantsesera itzulpena egin ostean.

5. Editorera itzultzen zarenean itzulpena hor egongo da. Hala ere, ondo gorde dadin sakatu *Publish* ataleko *Update* botoia.

Publish ▲

Aurrekusi aldaketak

Egoera: Published [Edit](#)

Ikusgaitasuna: Public [Edit](#)

Berrikusketak: 8 [Ara](#)[katu](#)

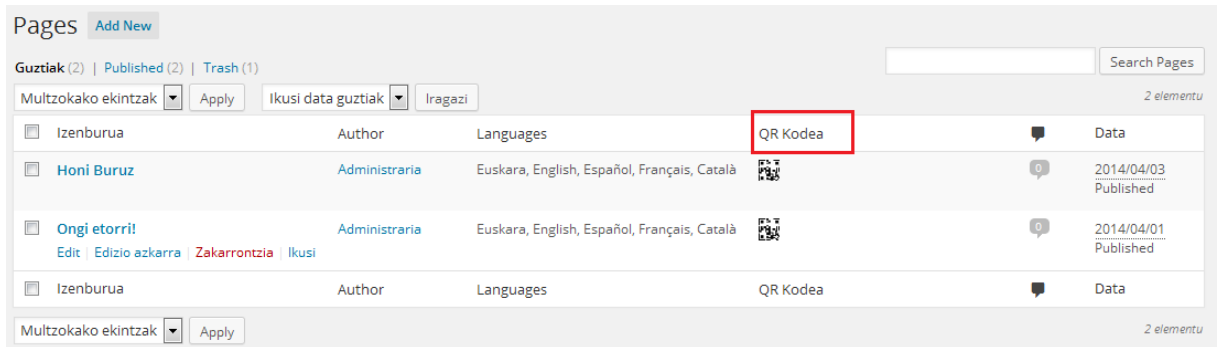
Noiz argitaratua: 2014(e)ko May 3 - 11:41(e)tan [Edit](#)

Bota zakezarrontzira **Update**

B.66 Irudia: Itzulpena gorde dadin editorean sakatu beharreko botoia.

QR kodea sortu

1. Bidalketak edota Orrialdeak zerrendatzen diren orrialdean zaudela bilatu QR kodea lortu nahi diozun elementua eta *QR Kodea* deituriko zutabeko botoia sakatu.



B.67 Irudia: QR sortzailea irekitzeko botoia erakusten duen irudia.

2. QR sortzailea irekiko zaizu eta bertan Orrialde/Bidalketaren QR kodea agertuko da, izenburuarekin batera.



B.68 Irudia: Bidalketa/Orrialdearen QR kodea erakusten duen orrialdea.

