

---

# ARGAZKI BACKUP PROIEKTUA (DASIAPP)

---

2019.eko abenduaren 17

Ander Duran  
Ekaitz Hurtado  
Urko Gonzalez



## Gaien Aurkibidea

|          |  |           |
|----------|--|-----------|
| <b>1</b> | <b>Proiektuen helburuen dokumentua</b> | <b>2</b>  |
| 1.1      | Sarrera . . . . .                      | 2         |
| 1.2      | Proiektuaren deskribapena . . . . .    | 2         |
| 1.3      | Proiektuaren arkitektura . . . . .     | 3         |
| 1.4      | Plangintza . . . . .                   | 5         |
| <b>2</b> | <b>Analisia</b>                        | <b>6</b>  |
| 2.1      | Domeinuaren eredua . . . . .           | 6         |
| <b>3</b> | <b>Diseinua</b>                        | <b>7</b>  |
| 3.1      | Sekuentzia diagramak . . . . .         | 7         |
| <b>4</b> | <b>Implementazioa</b>                  | <b>8</b>  |
| 4.1      | Ezaugarri nagusiak . . . . .           | 8         |
| <b>5</b> | <b>Izandako arazoak</b>                | <b>9</b>  |
| <b>6</b> | <b>Probak</b>                          | <b>12</b> |
| <b>7</b> | <b>Ondorioak</b>                       | <b>12</b> |
| <b>8</b> | <b>Erabilitako URL-ak</b>              | <b>13</b> |

# 1 Proiektuen helburuen dokumentua

## 1.1 Sarrera

ArgazkiBackup proiektua Informazio Sistemen Analisia eta Diseinua (ISAD) deituriko irakasgaian landu dugu, 2019-2020 urteen arteko ikasturtearen 1. lauhilekoan.

Proiektuaren helburua Flickr web orria kudeatzen duen aplikazio bat sortzea izan da. Honetarako Flickr-en API-a eta SQLite erabiltzea beharrezkoa zen, eta aplikazioa euskaraz egon behar zen, gutxienez. Beste bigarren mailako helburu batzuk API bat nola erabili, MVC patroia ikasi eta interfaze moderno eta erabilgarriak eraikitzea izan dira.

Proiektua burutzeko lauhileko osoa izan dugu, hau da, irailean aurkeztu egin ziguten, eta abenduaren erdira entregatu behar izan dugu.

## 1.2 Proiektuaren deskribapena

Eduki dugun denboran hauek dira inplementatu ditugun atazak:

1. Flickr zerbitzua kautotzea
2. Hizkuntza ezberdinetan aplikazioa ikustaraztea (euskaraz, gazteleraz eta ingelesez)
3. Flickr zerbitzutik irtetea (logout)
4. Argazkiak igotzea Flickr-era
5. Bisualizazio kopurua
6. Bisualizazioa konexioarekin eta konexiorik gabe
7. SQLite.(Zergatien atalean eduki ditugun arazoak agertuko dira)
8. Programaren diseinuan API ezberdinak eta erabiltzaile ezberdinak ahalbideratzea
9. Drag and Drop
10. CSS-en erabilera
11. TableView editagarri egitea argazkientzako (ez bakarrik String motako zutabeak)
12. Bildumak eta argazkiak ezabatzea
13. Datuak ez galtzea sinkronizazio oso landu baten ondorioz

Proiektua azaltzean ataza asko bururatu zitzaizkigun, baina azkenean ezin izan ditugun nahi beste ataza inplementatu. Inplementatu ez ditugunen artean hauek dira esanguratsuenak:

1. Beste API batzuk ere sartu nahi genituen, aplikazioak egiten dituen funtzio guztiak Dropbox eta GooglePhotos edo halako bi API kudeatzeko, horregatik eskatzen digu lehenengo pantailak zein zerbitzari erabili nahi dugun.
2. Erabiltzaile desberdinek aplikazioa erabiltzea ahalbideratu nahi genuen, eta gure DB-a horretarako prestaturik dago, baina ez dugu izan denborarik kodean ideia hau burutzeko.
3. “Atzera” botoi bat jarri nahi genuen, zeinek, non egon garenaren erregistro bat erabiliz gure pausuak atzera bota ahal izango zuena (hasiera baten ez zituen egindako aldaketak kenduko, baina hori ere botoi honen aplikazio oso interesgarri bat zen).
4. Azkenean, sinkronizazioa perfektua egotea, *Exception* guztiak ezin izan ditugulako landu, batzuk bai ordea.

### 1.3 Proiektuaren arkitektura

ArgazkiBackup proiektua Flickr4Java liburutegiaren gainean eraikita dagoela esan genezake. Liburutegi honek Flickr-en API-a eskuragarria usten du, eta gainera gure aplikazioaren eta Flickr-en zerbitzarien arteko lotura ere ezarriko du.

Gure aplikazioaren barnean SQLite erabiliko dugu, erabiltzailearen argazkiak, bildumak... zeintzuk diren gordetzeko, eta Flickr-era hainbeste dei ez egiteko (gainera, DB-a erabiltzeak aplikazioa konexiorik gabe erabiltzea ere ahalbidertzen du, hori bai, erabilera murriztapenekin). SQLite-ren beste onura bat da, ez duzula inolako zerbitzaririk behar datu-baserrako, MySQL-ek bai, adibidez.

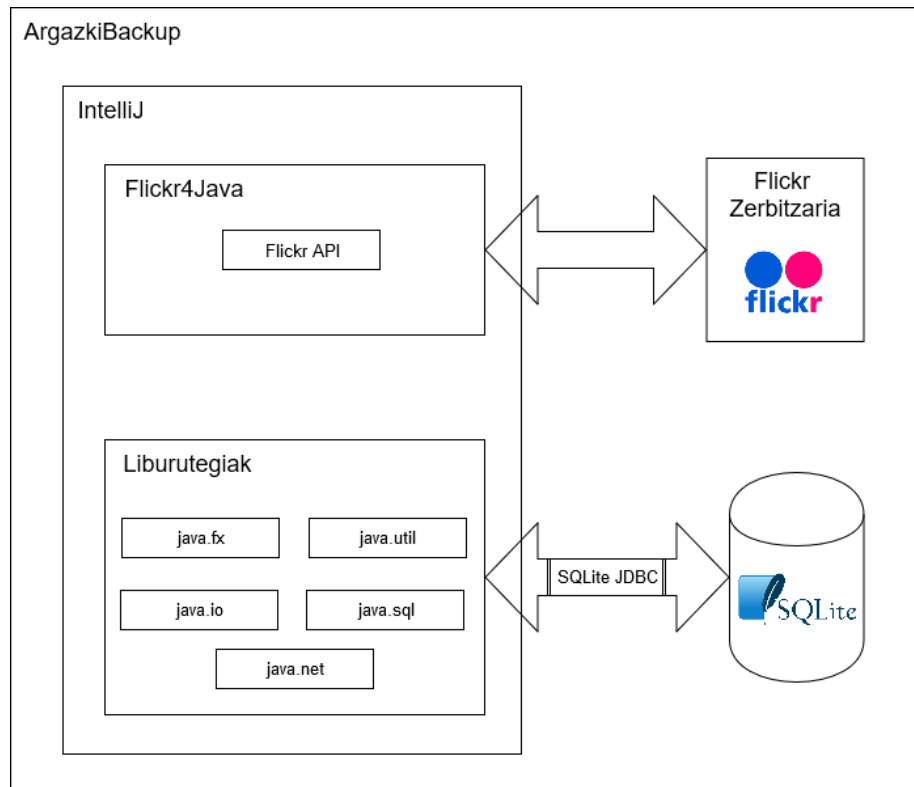
Gainera (eta DB-arekin loturik), proiektuan hainbat direktorio dauzkagu, bata bestearen barruan daudenak. Lehenengoaren barruan erabiltzaile bakoitzak gordeko dugu. Erabiltzaile bakoitza bere direktorioa izango du gure aplikazioaren barruan, eta zein zaren ikusi ondoren, zure fitxategiak zabalduko dizkizu.

Hurrengoko direktorioa Flickr direktorioa izango da, jada gure datuak izango dituen (aplikazioaren barruan dauzkagun argazkiak eta bi .txt, photosToUpload eta photosToDelete). *txt* hauek azkenengo momentuan kendu genituen, bertan zeuden datuak sortu genuen datu-baseko taula berri batzuetan gordeko genituelako.

Idea, behintzat, hau da: erabiltzaile bakoitzak bere direktorioa izango du, eta direktorio bakoitzaren barruan konektatu den web-orriaren karpeta (oraingoz

soilik Flickr-ena, baina beste API batzuk sartzea zen gure lehenengo ideia), eta web-orri bakoitzaren barruan argazkiak, igo behar diren argazkien id-a eta eza-batu behar direnen id-a.

Hemen, erabilitako zerbitzuak eta APIak adieraziko ditugu:



## 1.4 Plangintza

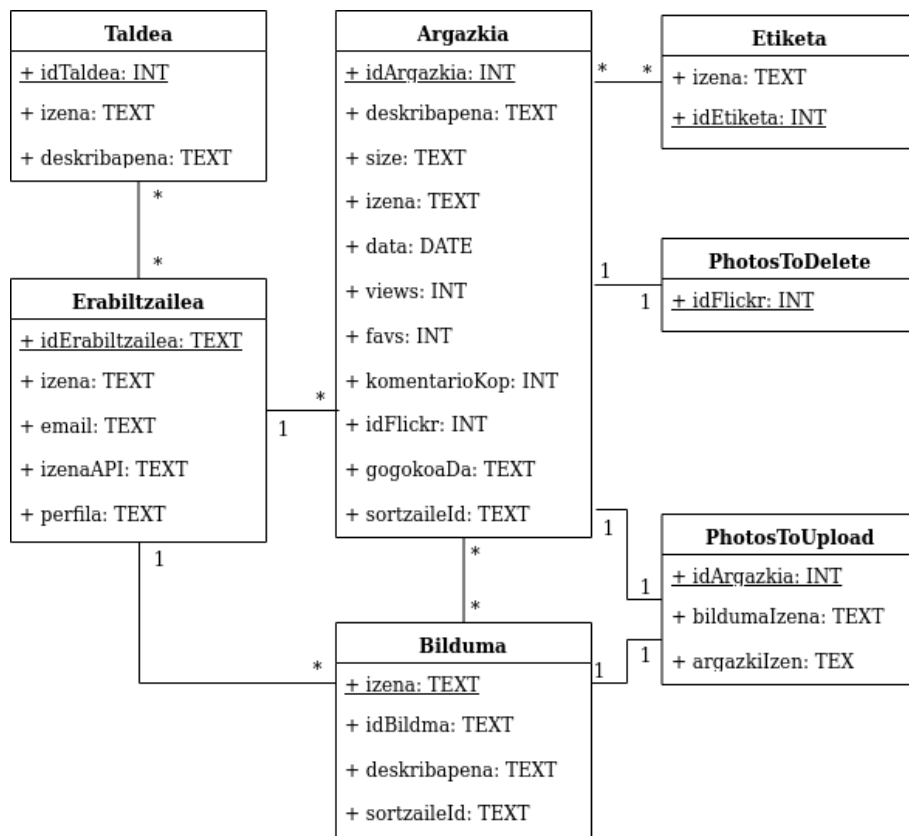
Hau izan da gure proiektuaren plangintza:

| Ataza                                | Pentsatutako orduak | Benetako orduak | Deskribapena  |
|--------------------------------------|---------------------|-----------------|---|
| Flickr konektatu                     | 6                   | 10              | Flickr-erako konexioa eta eskaerak kudeatu, terminalaren bidez.                                   |
| Interfaze grafikoa                   | 8                   | 14              | Interfaze grafikoaren sorrera (comboBoxak, pantailen arteko saltoak...).                          |
| Interfaze grafikoa eta kodea lotu    | 25                  | 50              | Flickr-en konexioa eta datuak pantailaratu, comboBoxak datu errealekin bete, argazkiak hautatu... |
| Domeinuaren eredua eta sek. diagrama | 4                   | 5               | DB-aren Domeinuaren eredua eta metodo nagusi baten sekuentzia diagrama.                           |
| DB                                   | 6                   | 15              | DB sortu, lotu eta erabili.   |

Proiektuan genituen lan helburu asko ez ditugu lortu, denbora asko egon garelako guk egindako erroreak zuzentzen eta gauza sinpleak egiteko. Batez ere aurrerantzean azalduko ditugun arazoak proiektuaren aurrerapena moteldu dute, eta ere bai eragin dute energia gehiago eskaintzea soluzio bat bilatzera interfazea txukuntzea baino. Batez ere hau ikusten da sinkronizazioa prozesuan, oso konplexua baita eta nahi genuena baino askoz denbora gutxi egon gara hau programatzen.

## 2 Analisia

### 2.1 Domeinuaren eredua



Ikus genezakeenez, implementatu dugun domeinu eredua nahiko sinplea da, gure datuak ez baitute erlazio arrarorik elkarren artean. Baliteke daturik arraroenak photosToUpload eta PhotosToDelete izatea.

Bi datu hauek gure aplikazioa “offline“ dagoenean funtziona dezan ezarri dugu, hala erabiltzaileak igo edo ezabatu nahi dituen argazkiak zeintzuk diren gordeko dugu, momenturen batean “online“ egoeran egonda *Sync* sakatzean bere ordenagailuan ezarritako aldaketak Flickr-era eraman ahal izateko. Hasiera batean bi .txt-ren bitartez egiten genuen, baina kodearen garbitasunagatik azkenean DB-an implementatzea erabaki dugu.

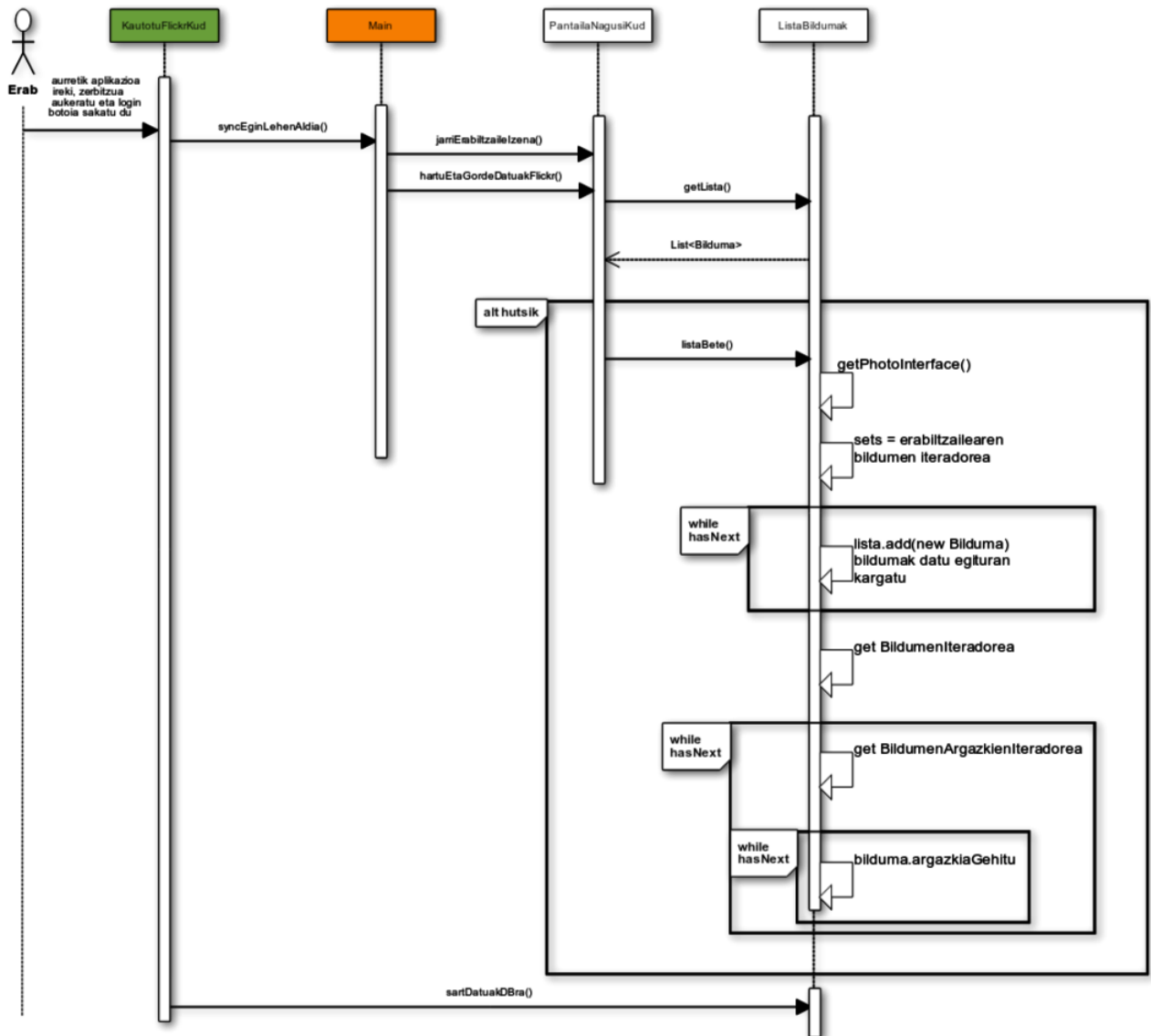
Hauei esker ere konexio erroreak ekidin ditzazkegu, eta modu honetan, argazkiak edo aldaketak ez galdu.



## 3 Diseinua

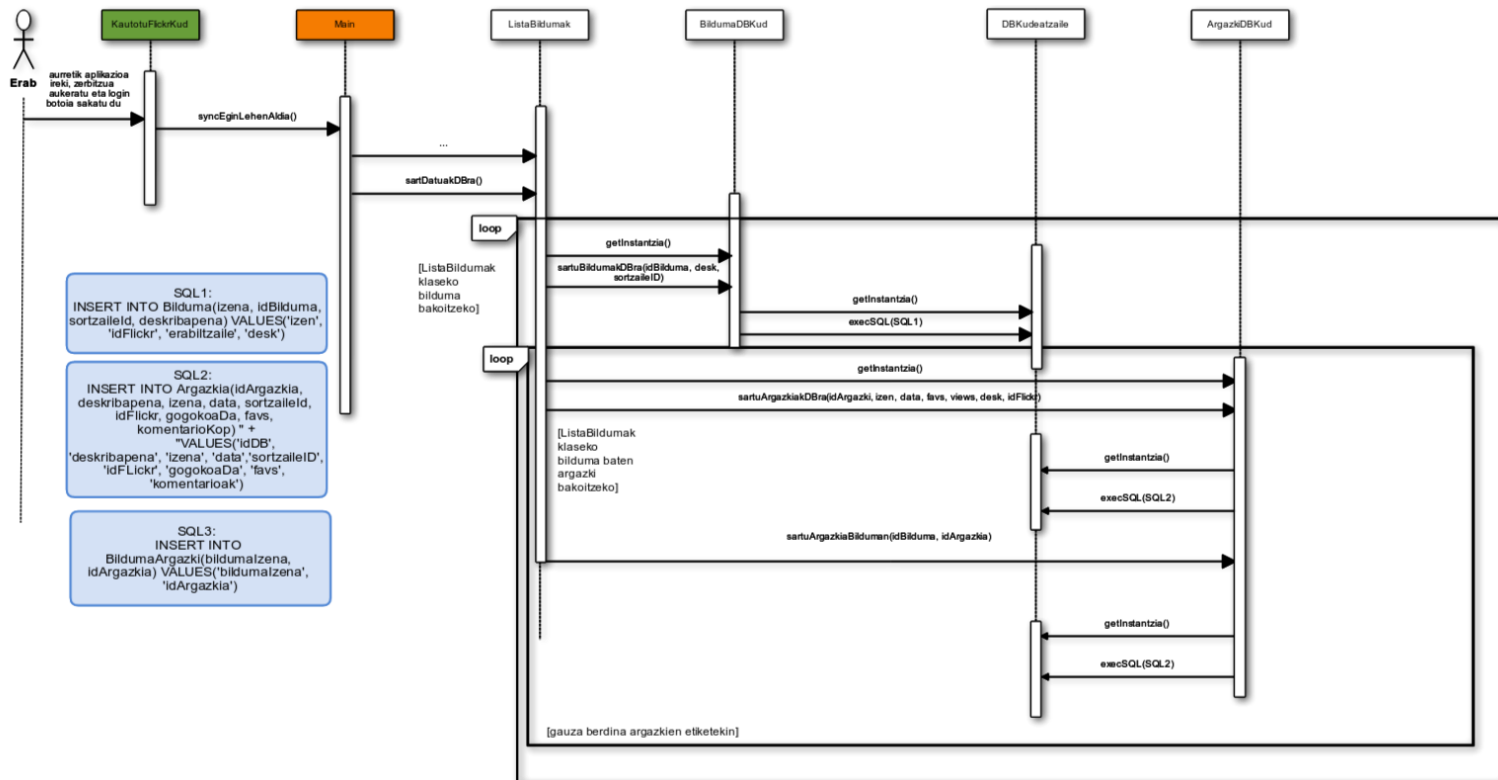
### 3.1 Sekuentzia diagramak

Erabiltzailea lehenengo aldiz kautotzerakoan betetzen den metodoa aurkeztuko dugu. Metodo hau oso handia da sekuentzia diagrama batentzat, beraz, gainera azaldu egin dugu, eta gero, nola sartzen diren datu guztiak datu-basera.



Oraingo hau bildumak, argazkiak, etab datu-basera sartzeko sekuentzia diagrama egin dugu. Tamaina aldetik ere oso handia es geratzeko, etiketen zatia

ez dugu sartu, argazkien ia berdina delako.



## 4 Implementazioa

Programatzen ez dakien erabiltzaileak kontuan hartuz eraiki dugu proiektu hau. Horregatik gure aplikazioa ez du terminaletik datu bat ere hartzen, eta exekuzioa ez da terminaletik hasi ere egin behar. Erabiltzailea testuen bitartez eta botoien bitartez bideratzen dugu, eta hori hurrengo ezaugarriekin lortu dugu.

### 4.1 Ezaugarri nagusiak

- Lehenengo pantailan erabiltzaileari zein zerbitzu erabili nahi duen galde-tzen diogu, eta aplikazioa zein hizkuntzatan erabili nahi duen ere ezarri behar du erabiltzaileak. Behin zerbitzua ezarrita, login prozesua hasten da. Hizkuntza aldatzeko, edozein momentuan erabiltzaileak pantaila Nagusiko “Atzera” botoia saka dezake eta lehenengo pantailara bueltatu.
- Login prozesuan URL bat erakusten dugu, “klikatzen” esaten duena. Han klikatzean webgune bat zabalduko da, zeinek baimen bat eskatzen

duen, eta baimena eman ostean kode bat kopiatu behar da, gero aplikazioan itsatsi behar dena. Ondoren “Jarraitu” botoian sakatu behar da, login prozesuarekin amaitzeko.

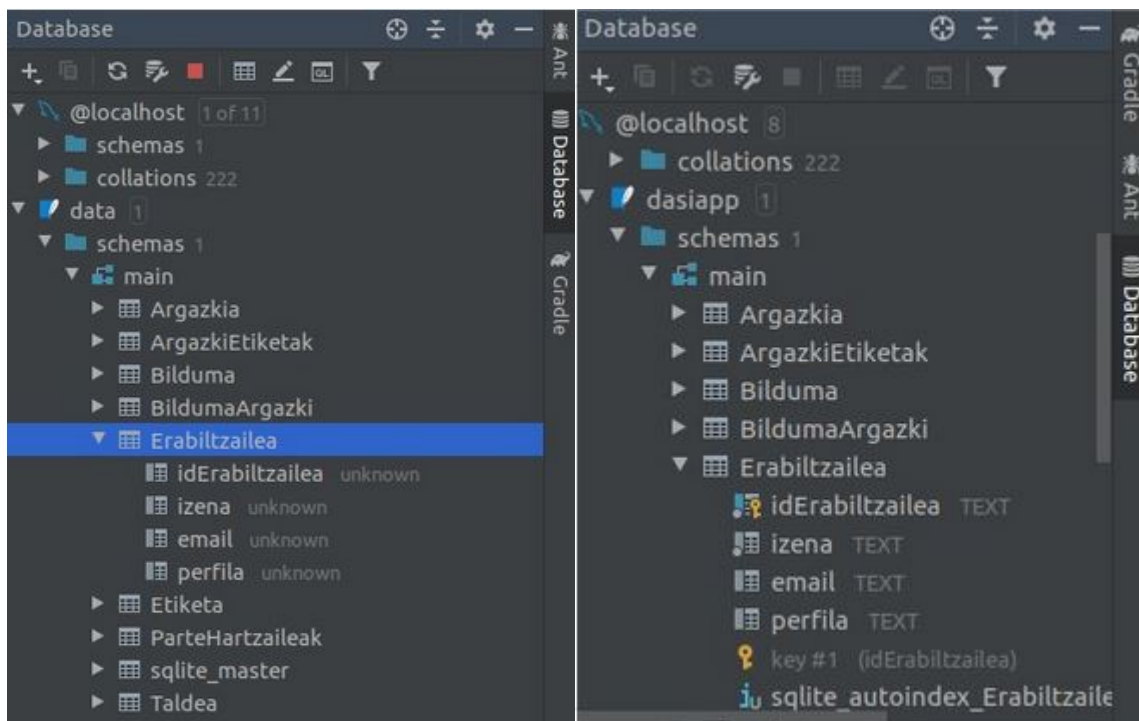
- Hurrengo pantaila gure pantaila nagusia da. Horretan bildumak eta argazkiak ikusi ahal izango ditugu, eta hortik logout prozesua, argazkiak igotzeko prozesuak, argazkiak ezabatzeko prozesuak... egikaritu ahal izango dira.
  - Argazkien datu batzuk editagarriak dira, hala nola: izena, etiketa, data, deskribapena eta baita komentario bat jartzeko aukera dago. Nahi duzuna edita dezakezu eta ondoren “Gorde” botoiari eman. Bildumaz aldatu nahi izango bazenu eta zerbait editatu egin duzu, aplikazioak abisatu egingo dizu datuak gordetzeko edo deuseztatzeko.
  - Argazkiak ezabatzeko, azken zutabean *CheckBox* bat jarri dugu. Horrela, nahi dituzunak aukera ditzakezu eta “Ezabatu Argazkia” botoiari eman. Edizioarekin bezala, zerbait aukeratuta badaukazu eta bildumaz aldatu nahi baduzu, abisu bat pantailaratuko da.
  - Bildumak sortzeko eta ezabatzeko aukera dago baita ere, “Bilduma Berria” eta “Bilduma Ezabatu” botoiak sakatuz.
- Argazkia igotzeko pantailan *Drag and Drop* inplementatu dugu, hau da, pantaila horretan argazki bat arrastatu egin ezkerreko, argazki horren datuak hartuko ditugu, gero igo botoiari ematean argazkia erabaki dugun zerbitzura igoko da.

## 5 Izandako arazoak

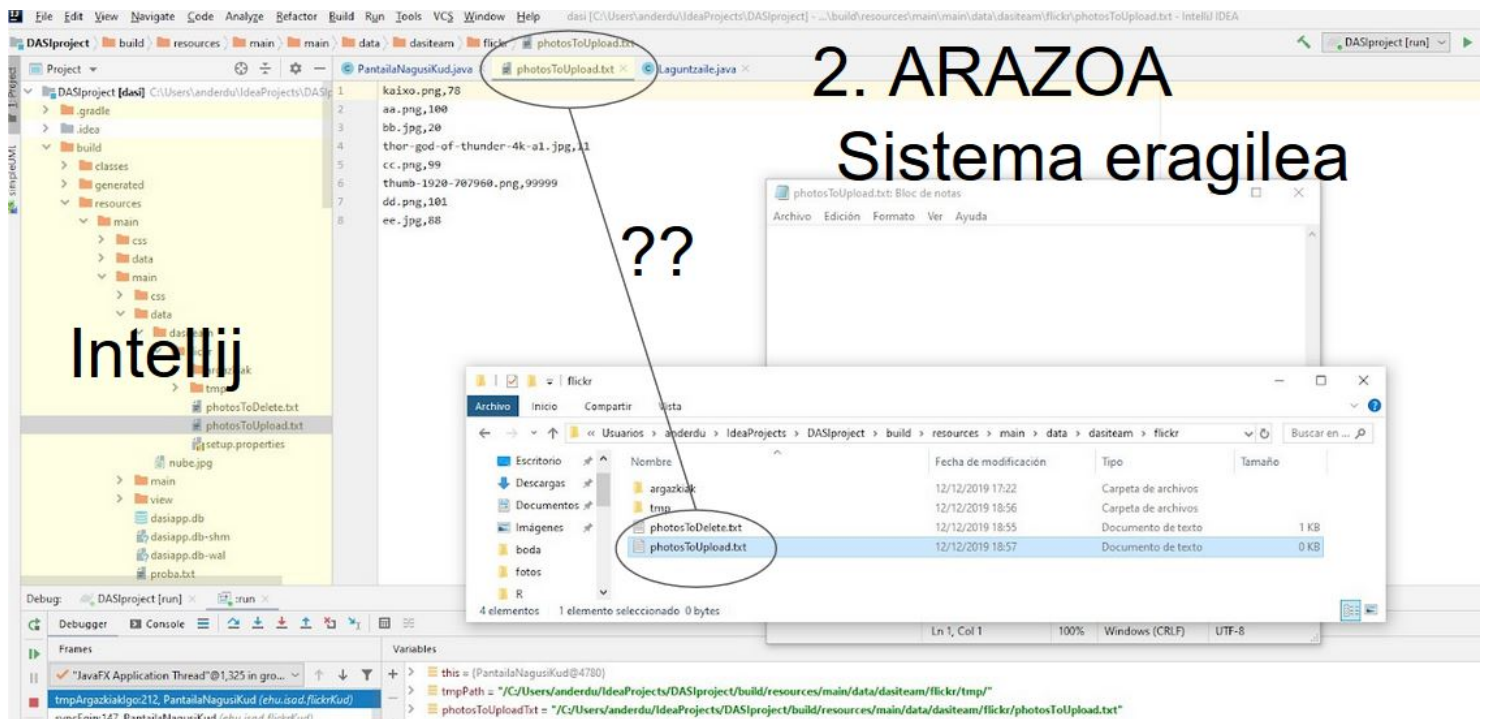
Proiektua garatu ahala arazoekin egin dugu topo, batzuk momentuan konpondu egin ditugu, baina hemen denbora gehien kendu diguten arazoei emango diegu ahotsa:

- Izan dugun arazorik nagusia denbora izan da. Irakasgaian lan anitz eduki izan ditugu, guztiak proiektuan aplikatu beharrekoak, baina lan horiek ez zeuden proiektura bideratuta, baliabide berri bat lantzerantz bideratuta zeudelako. Hala eta guztiz ere, ikusi dugun beste arazo bat zera izan da, etxean proiektua aurrera eramatea pentsa genuen, eta ordu asko arazo bat konpontzen egon eta gero, irakasleari galdetu eta klasean arazoa landuko dugula esan zigun, eta hau birritan gertatu ostean teoria guztia amaitzean proiektuarekin jarraitzea pentsa genuen (hau pentsatu bagenuen ere egin genituen gauza nahiko aldatu egin behar izan genituen beste baliabide bat erabili behar zelako).
- SQLite-rekin arazo anitz izan ditugu. Lehenengoz, *mySQL* -i *SQLite* konbertsioa egiteko erabili genuen baliabideak arazoak sortzen zigun, naiz eta klase-kideei arazorik ez sortu... Eta horrek hainbat ordu xahutzera eraman

gintuen, eta proiektua geldirik edukitzera hori konpondu arte, momentu horretan DB-aren inplementazioa egin behar baikenuen.



- Bestetik, gure proiektuak fitxategien aldaketa asko egitea behar du, eta dena *resources* karpetan gorde genuen. Karpeta hau estatikoa da, oso ondo balio du fitxategiak gordetzeko, baina ez bertan daudenak editatzeko. Horregatik, guk arazo izugarriak izan genituen honekin (datuen desagerpena, datu bikoizketa, fitxategi berriak bakarrik sortzea...). Proiektuaren aurkezpena baina 3 egun lehenago, dena *resources* karpetatik kanpo atera genuen, eta hortik aurrera dena arazo barik joan zen.



- FlickrAPIrekin ere erroreak izan genituen datuak hartzerakoan. Hasieran bakarrik izena, id eta horrelako datu simple gutxi batzuk ematen zigun. Gero, konturatu ginen “extras” sartu behar genuela bildumak eta argazkiak lortzeko momentuan. Hala ere, honekin bakarrik bilduma batean dauden argazkien datuak ondo bueltatzen zigun. *NotInASet*-eko argazkien datu edo deskribapen datuak ez genituen jada lortu, eta oraindik ez dakigu zergatik.

## NotInASet

```
setTitle = "NotInASet"
currentSet = {ArrayList@5951} size = 25
setIterator = {ArrayList$Itr@5954}
p = {Photo@5955}
pTitle = "WhatsApp_Image_2019-09-16_at_13.29.16"
pDescription = null
pDate = null
pDatePosted = null
pid = "48742475908"
pFavourite = {Boolean@5959} false
favs = {Integer@5960} 6
comments = {Integer@5961} 0
views = {Integer@5962} -1
etiketenLista = {ArrayList@5963} size = 0
etiketak = {ArrayList@5964} size = 0
```

## Beste bildumak

```
setTitle = "Ajo"
currentSet = {PhotoList@5974} size = 11
setIterator = {ArrayList$Itr@5975}
p = {Photo@5976}
pTitle = "Ajo"
pDescription = "Fotos sacadas finde cuadrilla ajo"
pUrl = "https://farm66.staticflickr.com/65535/48742598851_0c4ea2a6a7_m.jpg"
pDate = null
pDatePosted = {Date@5980} "Mon Sep 16 12:29:16 CEST 2019"
pid = "48742598851"
pFavourite = {Boolean@6003} false
favs = {Integer@6007} 1
comments = {Integer@6009} 0
views = {Integer@6007} 1
etiketenLista = {ArrayList@6012} size = 4
etiketak = {ArrayList@6015} size = 4
```

## 6 Probak

Ez dugu denborarik izan probak inplementatzeko. Proiektua garatu ahala sysout-en bitartez proba anitz egin ditugu, noski, datuak era egokian bueltatzen eta lantzen genituela ikusteko, baina, esan bezala, ez dugu denborarik izan proba automatizatuak egiteko.

## 7 Ondorioak

Hainbat ondorio atera ditugu proiektu osoan zehar. Gauza oso on bat egin duguna izan da dena ondo pentsatzea ezer egin aurretik. Errore kasu guztiak pentsatu ditugu erabiltzaileak ezer ez gal zezan.

Beste alde batetik egia da, antolakuntza aldetik ez dugu hain lan ona egin. Irailean aurkeztu ziguten proiektua eta hasieratik gelditu gabe egin beharrean, hiruhilekoaren erdian aste batzuk ia ezer egin gabe eman genituen. Hau beste irakasgai batzuen lan karga handiaren ondorioz, eta proiektu honen hasieran motibazioa galdu egin genuelako, gauzak egiten baikenituen, eta gero ez zutelako proiekturako balio, beste modu batzuetan egin behar zirelako.

Baina, egia da, asko ikasi dugula proiektu honetan. Nahiko denbora pasa dugu erroreekin, eta horrek ez digu utzi proiektu on egitea. Hala ere, erroreekin ikasten duzu, eta asko gainera, eta ia guztiekin topatu garenez, alde orretik ezagutza askorekin amaitu dugu proiektua.

Azkenik esan beharra dago, proiektu profesional bati begira ikaragarri ikasi dugula. Nola antolatu eta baita tresna oso baliagarriak erabiltzen, adibidez,

*Gradle* edo *JavaFX*. Alde honetatik oso pozik amaitu egin dugu lana, izugarri handitu dugulako gure ezagutza.

Ondorio nagusi gisa, oso proiektu polita iruditzen zaigu, ideia on bat zuena eta denbora sartzea kostatzen ez zaizuna (erroreak ematen ez dituenean).

## 8 Erabilitako URL-ak

<https://www.journaldev.com/878/java-write-to-file>  
Fitxategietan (.txt) testua idazteko

<https://www.geeksforgeeks.org/different-ways-reading-text-file-java/>  
Fitxategietan (.txt) testua irakurtzeko

<https://www.youtube.com/watch?v=ttD35jEo-f0>  
XML dinamikoa sortzeko

<https://coderanch.com/t/689645/gradle/build-tools/install-Gradle-Windows>  
Gradle windows 10-ean erabiltzeko

<https://coderanch.com/t/404575/java/static-final-final-static>  
Atributu estatikoen eta final-en arteko desberdintasuna ulertu

<https://stackoverflow.com/questions/7357969/how-to-use-java-code-to-open-windows-file-explorer-and-highlight-the-specified-f/39445156>  
Igotako argazkiak fitxategi tenporal batera igo

<https://www.youtube.com/watch?v=f7KGXUraH0g>  
Fitxategiak hartzeko UI-an artxiboak arrastatzen

<https://www.journaldev.com/861/java-copy-file>  
Argazkiak temp en kopiatu

<http://lineadecodigo.com/java/borrar-un-directorio-con-java/>  
Ezabatu karpeta batean dagoena

<https://www.flickr.com/groups/51035612836@N01/discuss/72157629352276563/>  
Argazki bat flickr-era igotzeko

<https://www.youtube.com/watch?v=W2SjWBMRCgE>  
Table View-en botoia jarri

<https://howtodoinjava.com/java/io/read-file-from-resources-folder/>  
Resources-eko fitxategiak irakurtzeko arazoaren zuzenketa

<https://www.flickr.com/services/api/flickr.photos.getFavorites.html>

Argazki baten gustuko kantitatea (favs) lortzeko

<https://stackoverflow.com/questions/33305037/javafx-images-path>  
new Image() JavaFX-ekin nola sortu

<https://stackoverflow.com/questions/33305037/javafx-images-path>  
new Image() JavaFX-ekin nola sortu

[https://www.simba.com/docs/SDK/SimbaEngine\\_Java\\_API\\_Reference/com/simba/dsi/utilities/StringConverter](https://www.simba.com/docs/SDK/SimbaEngine_Java_API_Reference/com/simba/dsi/utilities/StringConverter)  
SQL Date mota bat den taula zutabe batean defaultTextField bat jartzeko DateToStringConverter

<https://stackoverflow.com/questions/30800345/javafx-tableview-edit-triggers-keypress-events>  
Supr ukitzean taulatik datuak kentzeko 1

<https://stackoverflow.com/questions/17318015/keyeventlistener-recognising-up-and-delete-key-simultaneously-in-java>  
Supr ukitzean taulatik datuak kentzeko 2

<https://stackoverflow.com/questions/20413419/javafx-2-how-to-focus-a-table-row-programmatically>  
Supr ukitzean taulatik datuak kentzeko 3