

Informatika Ingeniaritzako Gradua

Software Ingeniaritza

Gradu Amaierako Lana

Web aplikazioen berfaktORIZAZIOA Software Produktu-Lerro bat garatuz

Egilea

Mikel Cantero

2019

Informatika Ingeniaritzako Gradua Software Ingeniaritza

Gradu Amaierako Lana

Web aplikazioen berfaktORIZAZIOA Software Produktu-Lerro bat garatuz

Egilea

Mikel Cantero

Zuzendaria(k)

Arantza Irastorza

Laburpena

Proiektu honen helburu nagusia Haritz Medinak garatutako aplikazioak (*Mark&Go*, *Highlight&Go* eta *Review&Go*) implementatzeko balio duen *Software Produktu-Lerroa* garatzea da.

Hori lortzeko, aplikazio hauen kodea aztertuko da beraien arteko berdintasun eta desberdintasunak (zati amankomunak eta aldakorrak) identifikatzeko asmoarekin. Behin azterketa hori eginda, SPLa definituko duen ezaugarri-diagrama diseinatuko da.

Azkeneko urratsa, behin ezaugarri-diagrama eta murriztapenak zehaztuta eta *pure::variants* tresnaz baliatuz, *Software Produktu-Lerroa* implementatzea izango da.

Gaien aurkibidea

Laburpena	i
Gaien aurkibidea	iii
Irudien aurkibidea	vii
Taulen aurkibidea	xi
1 Sarrera	1
1.1 Software Produktu-Lerroak	1
1.2 Web-anotazioak egiteko aplikazioak	2
2 Proiektuaren plangintza	5
2.1 Helburuak	5
2.2 Irismena	5
2.3 Atazak	6
2.3.1 Atazen deskribapena	7
2.4 Dedikazio plangintza	9
2.4.1 Gantt diagrama	9
2.4.2 Denbora estimazioa	9
2.5 Emangarriak	11

2.6	Kalitate-plana	13
2.6.1	Kudeaketaren kalitatea	13
2.6.2	Produktuaren kalitatea	14
2.6.3	Dokumentazioaren kalitatea	14
2.7	Arriskuak	14
2.8	Informazio eta komunikazio sistemak	16
2.9	Interesatuak	17
3	Proiektuaren birplangintza	19
3.1	Apirilaren 29ra arte dedikatutako orduak	19
3.2	Agertutako arazoak	20
3.3	GrALaren tutorearekin adostutako erabakiak	21
3.4	Helburu berriak	21
3.5	Denbora estimazio berria	21
4	Erabilitako tresnen azterketa	23
4.1	Kodearen analisirako tresnak: <i>DiffMerge</i> eta <i>BUT4Reuse</i>	23
4.1.1	BUT4Reuse	23
4.1.2	DiffMerge	24
4.2	SPL kodearen garapenerako tresna: <i>pure::variants</i>	25
4.3	<i>Feature Model</i>	26
4.4	<i>Family Model</i>	28
4.5	<i>Tagging</i>	28
4.5.1	<i>pvsclxml</i>	29
4.5.2	<i>pvscltext</i>	30
4.6	<i>Variant model</i>	31

5	Iturburuko aplikazioen domeinua eta funtzionalitateak	33
5.1	Highlight&Go	33
5.2	Mark&Go	36
5.3	Review&Go	38
6	WebAnnotationSPLaren ezaugarri-diagrama	41
6.0.1	Ezaugarri-diagrama <i>pure::variants</i> -en	47
7	WebAnnotationSPLaren arkitektura	49
7.1	Iturburuko fitxategiak	49
7.2	<i>pure::variant</i> seko <i>Family Model</i> a	55
8	WebAnnotationSPLtik lortutako produktuak	57
9	Garapenean zehar sortutako arazoak eta egindako lan gehigarria	63
9.1	Arazo eta zailtasunak	63
9.2	Egindako lan gehigarria	65
10	Jarraipen eta kontrola	69
10.1	Helburu eta irismenaren betetze maila	69
10.2	Denbora desbiderapenak	70
10.3	Gantt diagrama eta emangarrien kontrola	70
10.4	Lortutako kalitatea	74
10.4.1	Produktuaren kalitatea	74
10.4.2	Dokumentazioaren kalitatea	74
10.4.3	Kudeaketaren kalitatea	74
11	Ondorioak	77
11.1	Gradu Amaierako Lanari buruzko hausnarketak eta ondorioak	77
11.2	Lanarekin ikasitakoa	78
11.3	Etorkizunera begira	78

Eranskinak

A Funtzionamendua	81
A.1 Diff	81
A.2 Merge	83
B BUT4Reuse <i>plugin</i>-aren instalazioa eta funtzionamendua	87
B.1 <i>Plugin</i> -aren instalazioa Eclipse-n	87
B.2 Funtzionamendua	90
C Ezaugarri diagramaren nomenklatura	95
D Produktuak lortzeko prozesua	97
E Bileren aktak	99
E.1 1. bileraren akta	99
E.2 2. bileraren akta	101
E.3 3. bileraren akta	102
E.4 4. bileraren akata	103
E.5 5. bileraren akata	104
E.6 6. bileraren akata	105
E.7 7. bileraren akata	106
E.8 8. bileraren akata	107
E.9 9. bileraren akata	108
E.10 10. bileraren akata	109
F Egunero sartutako orduen taulak	111
Bibliografia	119

Irudien aurkibidea

2.1	LDE diagrama.	6
2.2	Gantt diagrama.	10
2.3	denbora estimazioaren LDE diagrama.	12
3.1	Apirilaren 29ra arte dedikatutako denbora.	20
3.2	Denbora estimazio berria.	22
4.1	Weather Station SPL-ren <i>Feature Model</i> -a: Ezkerraldean zuhaitz moduan (<i>Tree</i>) ikusi daiteke eta eskuinaldean modu grafikoan (<i>Graph</i>).	27
4.2	<i>pure::variants</i> -eko <i>Family Model</i> baten adibidea.	29
4.3	<i>Variant model</i> baten aukeraketa.	31
5.1	<i>Highlight&Goko Classifying</i> modua. Irudia Medina u. a. (2019a) artikulutik lortu da.	34
5.2	<i>Highlight&Goko Check</i> modua. Irudia Medina u. a. (2019a) artikulutik lortu da.	35
5.3	<i>Mark&Goko Evidencing mode</i> modua. Irudia Medina (2019b) artikulutik lortu da.	37
5.4	<i>Review&Go</i> aplikazioa. Irudia Medina u. a. (2019b) artikulutik lortu da.	39
6.1	Ezaugarri-diagrama.	42
6.2	Ezaugarri-diagrama <i>pure::variants</i> en.	48

8.1	<i>WebAnnotationH</i> eta <i>WebAnnotationHwithComments</i> produktuen arteko alderaketa.	58
8.2	<i>WebAnnotationH</i> eta <i>WebAnnotationHwithToolbarAndStaticGroup</i> produktuen arteko alderaketa.	59
8.3	<i>WebAnnotationR</i> eta <i>WebAnnotationRwithGroupSelector</i> produktuen arteko alderaketa.	60
10.1	Implementatzea lortu diren ezaugarriak adierazten duen ezaugarri diagrama.	71
10.2	GrALeko ataza bakoitzean kontsumitutako denbora adierazten duen taularen irudia.	72
10.3	Proiektuaren prozesuen iraupen erreala adierazten duen Gantt diagrama. .	73
A.1	<i>File</i> erlaintzeko <i>Open file diff...</i> aukera.	82
A.2	<i>Diff</i> egiten hasteko sakatu beharreko ikurra	82
A.3	konparatu nahi diren fitxategiak aukeratzeko menua.	82
A.4	<i>Diff</i> motako konparaketa baten adibidea.	83
A.5	<i>File</i> erlaintzeko <i>Open file merge...</i> aukera.	84
A.6	<i>Merge</i> -ren ikurra.	84
A.7	<i>Merge</i> adibidea.	85
B.1	<i>Help</i> erlaintzeko <i>Install New Software</i> aukera. Iturria: BUT4Reuse.	88
B.2	<i>Name</i> eta <i>Location</i> eremuak dituen leihoa. Iturria: BUT4Reuse.	88
B.3	Instalatu nahi den <i>plugin</i> -aren aukeraketa leihoa. Iturria: BUT4Reuse. . .	89
B.4	<i>ArtefactModel</i> sortzen.	90
B.5	Konparatu nahi den edukia proiektuaren barruan itsatsita.	91
B.6	<i>ArtefactModel</i> -eko <i>Feature Identification</i> aukera.	91
B.7	<i>Adapters selection</i> menua.	92
B.8	A, B eta C direktorioen egituraren arabera blokeen sakabanaketa.	92
B.9	<i>Construct separately</i> aukera.	93

C.1	Ezaugarri moten nomenklatura.	96
F.1	Urtarrilean sartutako orduak.	112
F.2	Otsailean sartutako orduak.	113
F.3	Martxoan sartutako orduak.	114
F.4	Apirilean sartutako orduak.	115
F.5	Maiatzean sartutako orduak.	116
F.6	Ekainean sartutako orduak.	117

Taulen aurkibidea

7.1	Egindako etiketen taula.	55
7.2	<i>Family Modelean</i> ezarritako murriztapenen taula.	56
8.1	Produktu bakoitza lortzeko aukeratu diren ezaugarrien taula.	61

1. KAPITULUA

Sarrera

Proiektuarekin hasi aurretik, egokia da bertan landuko diren kontzeptuak aurkeztea eta azaltzea. Atal honen helburua, irakurlearen burua proiektuan kokatzea da, eta agian eza-gutzen ez zituen kontzeptuak gaintetik ikustea.

1.1 Software Produktu-Lerroak

Bistakoa da gaur egun teknologiaren garaian bizi garela. Hori dela medio, software tres-nen garapen-denborak murriztu behar izan dira, hauen kalitatea jaitsi gabe.

Industriaren iraultzan kateko ekoizpena agertu zen produkzio maila handitzeko eta azkar-tzeko asmoarekin. Softwarearen produkzioan antzeko zerbait lortzea du helburu *Software Produktu-Lerroak*. SPLa software bilduma bat da, non bilduma horretan parte hartzen du-ten aplikazioek ezaugarri amankomunak dituzten eta software guraso berdinetik “jaio” diren (*core asset* berdinetik inplementatu diren aplikazio bilduma)([Wikipedia, 2015](#)).

Software Produktu-Lerroak softwarea garatzeko ezagutzen zen modua guztiz aldatzen du. Amaierako produktuek izango dituzten ezaugarriak aztertzen dira lehendabizi, eta azter-keta horren ondorioz, produktuen alderdi amankomunak (konpartitzen dituzten ezauga-riak) eta berezitasunak (produktuak desberdintzen dituzten ezaugarri bilduma) identi-fikatuko dira. Konparaketa honek, produktu bakoitzeko kode multzo bat eduki beharrean, kode multzo bakarra edukitzea ahalbidetzen digu. Ezaugarri bakoitza kode zati batekin

lotzen da, beraz, bezeroak aukeratzen dituen ezaugarrien arabera kode bat sortuko da edo beste bat. Hiru aplikazioak kode iturri berdinetik sortzen dira, alegia.

Beraz, SPLra orientatzen den enpresa batek, bere software-garaztaile guztiak garatze metodo honetara orientatu beharko ditu; bere alde positiboak eta negatiboak onartuz.

1.2 Web-anoztazioak egiteko aplikazioak

Web-anoztazioak egiteko aplikazioak, web-orrietan azpimarratzeko, iruzkinak jartzeko eta informazioa konpartitzeko erabiltzen diren aplikazioak dira. Mota honetako aplikazioak erabilgarriak dira informazioa partekatzeko. Web-anoztazioen aplikazio ezagunetako batzuk:

- **Hypothes.is:** kode-irekiko softwarea da, artikuluen, liburuen eta albisteen berrikuspena egiteko erabiltzen dena, gehienbat. Aurkitu diren web-anoztazio tresnen artean hau da zabalena, gainera, proiektu honetan **landuko diren aplikazioek ere hau erabiltzen dute oinarri moduan**. Nota pribatuak idazteko, lan talde pribatuak sortzeko, baita anoztazio publikoak egiteko aukera ematen du ([Wikipedia, 2018b](#)).
- **A.nnotate:** tresna hau PDF, Microsoft Office, Open Office eta argazkien formatu ezagunenak onartzen ditu. Web-orriak anoztatu daitezke ere, hori egin ahal izateko, web-orriaren esteka kopia behar da aplikazioan eta honek web-orriaren HTMLa kopiatzen du, bertan anoztazioak egiteko. A.nnotatek ez die uzten bere erabiltzaileei anoztazio publikoak egiten, beti pribatuan egin behar dira. Beste erabiltzaileekin anoztazioak konpartitzeko aukera badago, posta elektronikoz gonbidapena bidaliz ([Wikipedia, 2018a](#)).

Haritz Medinak, *Onekin* ikerketa taldean lan egiten duen ikertzaileak, hiru aplikazio garatu ditu web-anoztazioetan oinarrituz, eta zehazki *Hypothes.is* erabiliz. Hauek izango dira proiektu honetan landuko direnak beren funtzionalitateak eskaintzen dituen SPLa sortzeko:

- **Highlight&Go (Medina, 2018):** aplikazio hau ikerketa taldeko informazio bilketa errazteko sortu zen. Aplikazio honek web-orriak azpimarratzeko balio du. Anoztazioak, zehaztutako irizpide batzuen arabera, kolore batez edo bestez agertuko dira.

- **Mark&Go (Medina, 2019a):** aurreko aplikazioaren hedapena da. Aplikazio hau azterketak ebaluatzeko garatu zuen Medina. Aurretik zehaztutako irizpideen edo errubrikaren arabera azterketak ebaluatzeko aukera ematen du aplikazio honek, PDFn eta Moodle bitartez entregatutako azterketak, alegia. Horrela, irakasleak azterketetan aurkitutako erroreak markatu ditzake, bere azken nota justifikatzeko.
- **Review&Go (Medina u. a., 2019b):** *Mark&Go*ren antza dauka aplikazio honek, baina kasu honetan ikerketa mundura bideratuta. Ikerketa artikuluak markatu daitezke ikerlariak egokiak ikusten dituen irizpideen arabera.

5. kapituluari hiru aplikazio hauen domeinua sakonki azalduko da. Garatu zen lehenengo aplikazioa *Highlight&Go* izan zen. Haritz Medina garatutako hurrengo aplikazioak (*Mark&Go*) lehenengo aplikazio horren kodea hartzen du oinarritzat, eta hirugarrenak (*Review&Go*) bigarrenaren kodea. Beraz, Haritz Medina hiru aplikazio hauek kopiatu eta itsatsi programatze teknikarekin (*copy & paste programming*¹) garatu dituela esan daiteke.

Hori dela eta, hiru aplikazioek kode zati amankomunak dituzte, baita bakoitza beste aplikazioetatik bereizten duten berezitasunak ere. Bereztasun horretatik sortzen da proiektu honen ideia, hiru aplikazioen kodearen azterketan oinarrituz horien funtzionalitateak eskainiko dituen *Software Produktu-Lerro* bat sortzea.

Aipatu behar da hiru aplikazioek *Hypothes.is* erabiltzen dutela anotazioak, iruzkinak, eta abar gordetzeko. Beraz, ezinbestekoa izango da hiru aplikazioak erabiltzeko *Hypothes.is*eko kontu bat edukitzea.

¹*copy & paste programming* buruz informazio gehiago: https://en.wikipedia.org/wiki/Copy_and_paste_programming

2. KAPITULUA

Proiektuaren plangintza

Atal honetan proiektuaren plangintza ikusiko da. Hemen proiektuaren helburuak, irismena, atazak, denbora kudeaketa, emangarriak, arriskuak, kalitatea, informazio-sistema, komunikazio-sistema eta interesatuak aztertuko dira.

2.1 Helburuak

Lan honen helburu nagusia *Highlight&Go*, *Mark&Go* eta *Review&Go* aplikazioak inplementatzeko balioko duen *Software Produktu-Lerroa* garatzea da. Horretarako, lehendabizi aipaturiko aplikazioen kodea aztertuko da antzekotasunak (ezaugarri amankomunak) eta berezitasunak identifikatzeko. Behin kodea aztertu dela, SPLaren ezaugarri-diagrama diseinatuko da. Eta azkenik, SPLaren inplementazioa egingo da *pure::variants* erabilita.

Era berean, tamaina handiko proiektu bat kudeatzen eta SPLa garatzeko erabiliko diren teknologiak (*pure::variants*, *DiffMerge*, *BUT4Reuse*, *JavaScript...*) erabiltzen ikastea bigarren mailako helburuak izango lirateke.

2.2 Irismena

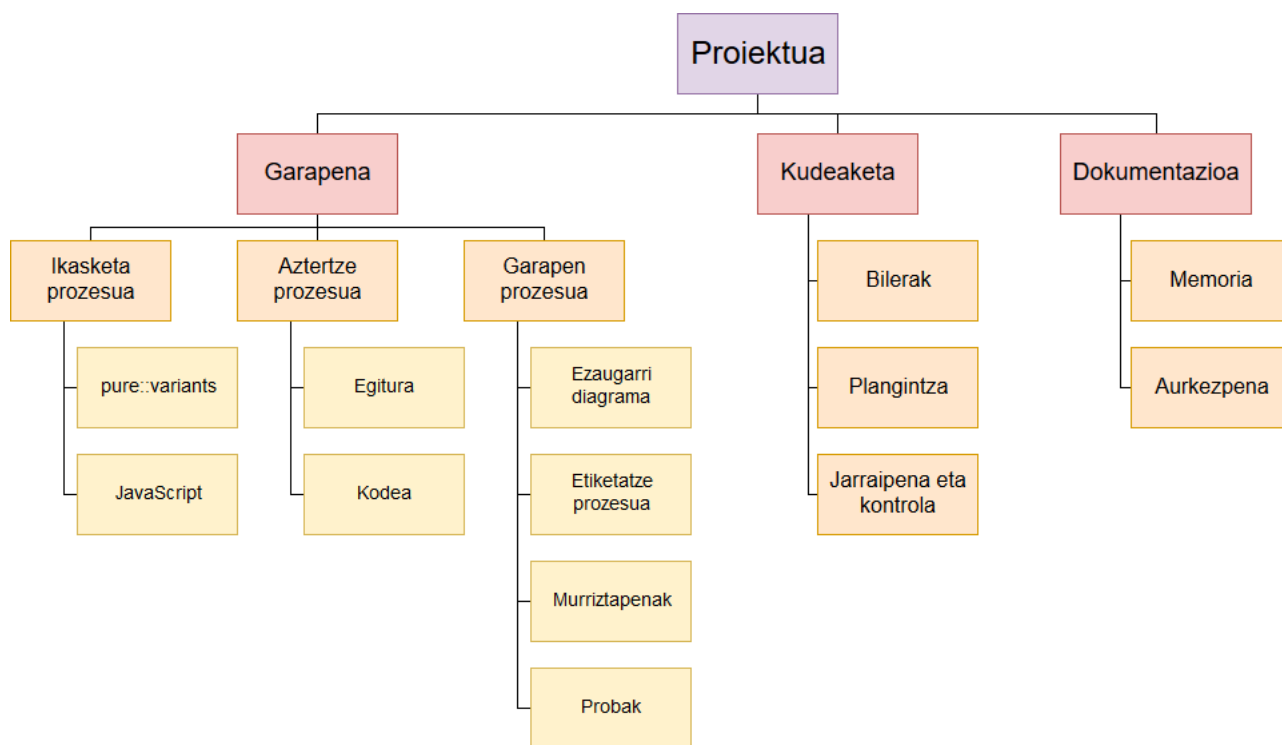
Lan honetan *Highlight&Go*, *Mark&Go* eta *Review&Go* aplikazioen parekoak garatzeko balioko duen SPL bat lortu nahi da. Printzipioz ez dira funtzionalitate berriak edo aplikazio berriak garatuko, baina, aurrerago planifikatutako denbora aurreztu delako edo

funtzionalitate berri bati lehentasuna eman nahi izanez gero, posible da funtzionalitate berriak garatzen hastea. Alabaina, oraingoz lehentasuna SPLan hiru aplikazio horien funtzionalitatea sortzea da.

Lanaren garapena arrakastatsua izan dadin, aurrerago zehaztuko diren kalitate irizpideak, plangintzan zehaztutako pausoak eta denbora balioespena ahalik eta neurri handienez bete beharko dira.

2.3 Atazak

Lanaren garapena hiru adar nagusitan banatu da (ikusi 2.1 irudia): proiektuaren garapena, proiektuaren kudeaketa eta proiektuaren dokumentazioa.



2.1 Irudia: LDE diagrama.

Lehenengo adarrean, **proiektuaren garapena**, proiektua garatzeko egin beharreko atazak identifikatzen dira, beste hiru adarretan banatuta egonik: ikasketa prozesua (proiektua aurrera eramateko jakin beharreko tresnen erabilpena eta ulertu beharreko kontzeptuen ikasketa denbora: *pure::variants*, JavaScript eta bestelakoak), azterte prozesua (aplikazioak aztertze prozesua, hau da, kodea aztertze eta konparatzeko emango den den-

bora) eta garapen prozesua (*pure::variantsen* ezaugarri-diagrama, murriztapenak, probak eta etiketatze prozesua egiteko).

Bigarren adarrean **proiektua kudeatzeko** egingo diren atazak ikusten dira: tutorearekin egingo diren bilerak, proiektuaren plangintza (bigarren atal hau) eta proiektuaren jarraipen eta kontrola, hau da, plangintza noraino betetzen ari garen jakiteko.

Azkeneko adarrean proiektua aurkezteko erabiliko diren **dokumentuak** dauzkagu: proiektuaren memoria eta defentsa egiteko erabiliko den aurkezpena.

2.3.1 Atazen deskribapena

Atal honetan aurrekoan identifikatu diren atazak deskribatuko dira.

- **Garapena:** hiru aplikazioen (*Highlight&Go*, *Mark&Go* eta *Review&Go*) SPLa garatzeko eman beharreko urratsen bilduma. Hiru azpimultzotan banatu da: ikasketa prozesua, aztertze prozesua eta garapen prozesua.
 - **Ikasketa prozesua:** proiektuan zehar erabiliko diren tresnen ikasketa prozesua da hau. Multzo honen barruan proiektua garatzeko erabiliko diren tresnak aurre-identifikatu dira.
 - * **pure::variants:** tresna hau SPLa garatzeko erabiliko den tresna da, beraz, SPL-a garatu ahal izateko tresna hau nola funtzionatzen duen jakin beharko da. Gainera, tresna hau erabiltzen den lehenengo aldia da.
 - * **JavaScript:** Haritz Medinak garatutako aplikazioek lengoia hau erabiltzen dute, beraz, aplikazioen kode zati bakoitzak zer egiten duen jakiteko, lengoia hau menderatu beharko da.
 - * **Bestelakoak:** atal hau libre uzten da aurreikusi ez diren tresnen ikasketa prozesurako.
 - **Aztertze prozesua:** hiru aplikazioen kode azterketa prozesua da hau. SPLa egiteko oso garrantzitsua da kodearen egitura eta kodea bera ulertzea eta amankomunean duten ezaugarriak eta desberdintasunak identifikatzea.
 - * **Egitura:** kodearen egitura aztertze erabiliko den denbora.
 - * **Kodea:** aplikazioen kodea aztertze erabiliko den denbora, ezaugarri amankomunak eta desberdinak identifikatuz.
 - **Garapen prozesua:** SPLa *pure::variantsen* garatzeko erabiliko den prozesua da. Honako azpiatazak bereizten dira:

- * **Ezaugarri-diagrama:** aurretik diseinatutako ezaugarri-diagrama *pure::variants* bitartez idazteko emango den denbora.
 - * **Etiketatzeko-prozesua:** ezaugarri bakoitzak behar duen kodearen etiketatzeko prozesua. Adibidez, A ezaugarriak C kode zatia behar badu, kodean C zatia etiketa bat jarri beharko diogu A ezaugarria aukeratzeko dugunean C kode zatia sortzeko.
 - * **Murritzapenak:** ezaugarrien arteko murritzapenak zehazteko ataza da hau. Batzuetan A ezaugarriak B ezaugarria edukitzea eskatzen du. Mota horretako murritzapenak, hots, mendekotasun murritzapenak inplementatuko dira prozesu honetan.
 - * **Probak:** SPLaren inplementazioa zuzena dela ziurtatzeko egingo den proba multzoa.
- **Kudeaketa:** lanaren nondik-norakoa zehazteko eta kontrolatzeko erabiliko da ataza hau. Hiru azpimultzo identifikatzen ditugu hemen:
 - **Bilerak:** lanean zehar tutorearekin egingo diren bilerak dira. Bilera hauek zailtasunak argitzeko eta lanean zehar egingo diren atazak zuzentzeko erabiliko dira.
 - **Plangintza:** lanaren plangintzan proiektuaren irismena, atazak, emangarriak, denbora estimazioa, kalitatea, lan-metodologia eta arriskuak identifikatu, aurreikusi eta deskribatuko dira. Lanaren arrakasta plangintzan zehazten diren pausoen jarraipenean datza.
 - **Jarraipen eta kontrola:** plangintzan identifikatutako atazak egiteko kontsumitutako denbora kontrola, kalitate irizpideak jarraitzen ari diren eta aurreikusitako irismena (besteak beste) betetzen den kontrolatzeko erabiliko den prozesua da hau.
 - **Dokumentazioa:** lanaren dokumentazioa bi zatitan banatzen da: proiektuaren memoria eta aurkezpena.
 - **Memoria:** lanaren plangintza, jarraipena eta kontrola eta garapena azaltzen duen dokumentua da hau.
 - **Aurkezpena:** lanaren defentsa egiteko erabiliko den materiala; esaterako, gardenkiak edota demostrazio bideoa.

2.4 Dedikazio plangintza

Jarraian lanaren esfortzua zein puntutan enfokatuko dugun eta noiz azalduko da, ataza bakoitzean zenbat denbora kontsumitzea espero den eta gero, kontsumitu den denbora errealarekin konparatuko da.

2.4.1 Gantt diagrama

Atal honetan aipatuko den guztia modu grafikoan Gantt diagraman ikusi daiteke (ikusi [2.2](#) irudia). Gantt diagramak proiektuko bilakaeraren aurreikuspen bat da.

Proiektua 2019ko urtarrilaren 22an hasi zen, Haritz Medinak eta Raul Medeirosen presatu zuten bilerarekin (ikusi [E](#) eranskina bileraren informazio gehiago lortzeko). Lehenak bere aplikazioen funtzionamendua eta kodea ginetik azaldu zuen, eta bigarrenak *pure::variants*en funtzionamendua azaldu zuen. Lanaren garapena maiatzaren amaierarako bukatuta egotea espero da. Alabaina, lana ez da guztiz bukatuko memoria entregatu eta defentsa egin arte, hau da, uztailaren 1 eta 12ren artean.

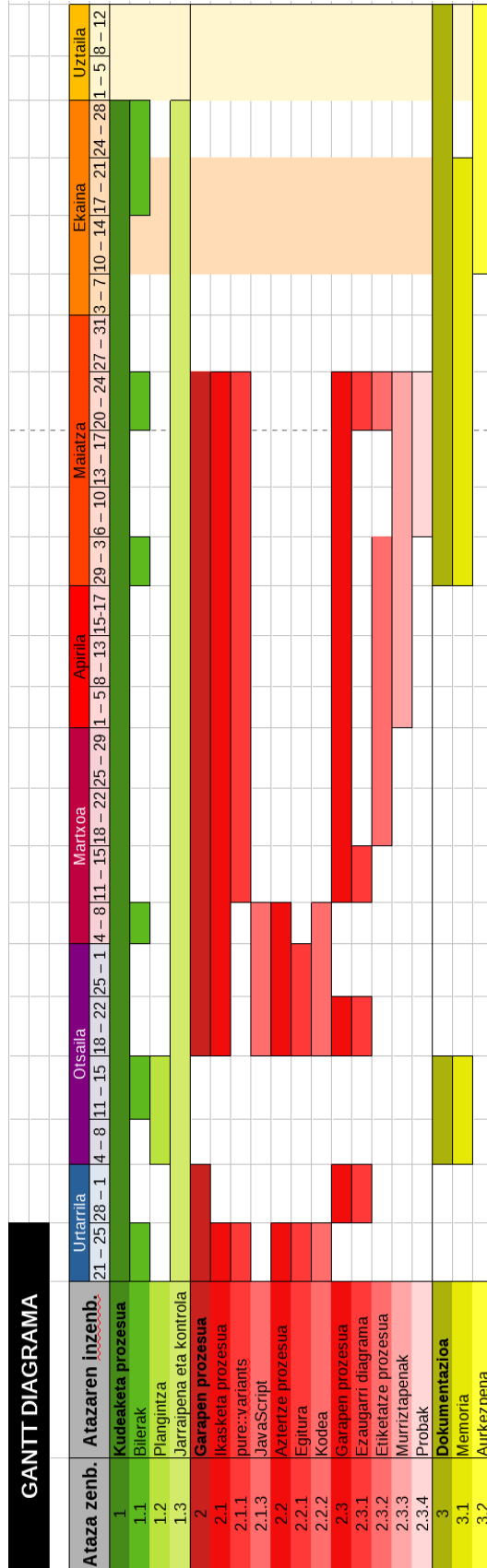
Oinarriko hiru aplikazioen kodea aztertzeko prozesua eta JavaScript ikasketa prozesua paraleloan egingo dira. Bestalde, aztertze prozesua bukatzerakoan, *pure::variants*en ikasketa prozesua, garapen prozesuarekin batera egingo da. Dokumentazioaren fasea, proiektuaren garapenaren amaierako fasean hasiko da.

Gantt diagramako bi aste koloreztatuta daude: laranja eta horiz. Laranja margotuta daudenak, proiektuaren entrega epea adierazten dute, eta horiz daudenak, defentsa egiteko epea. Aipatu behar da ere, Gantt diagraman ez direla asteburuak agertzen, horrek esan nahi du, ez dela aurreikusten asteburutan lan egitea. Hala ere, denboraz larri ibiliz gero, asteburuak lan egiteko erabili daitezke.

Urtarrilaren 22tik, maiatzaren 24ra, 17 aste daude (aste santuko bi aste horiek kenduta). Guztira 300 ordu inguru sartu behar direla jakinik, astean 18 ordu inguru sartu beharko dira, beraz, 3 ordu eta erdi inguru egunean ($\frac{300}{17} \approx 18$ ordu astean; $\frac{14}{5} = 3,6$).

2.4.2 Denbora estimazioa

Denbora estimazioa egiteko LDE diagrama oinarriztat hartuko da (ikusi [2.3](#) irudia). Proiektu osoa egiteko 300 ordu daude. Pisu gehien duen adarra **garapenaren** adarra da, bertan



2.2 Irudia: Gantt diagrama.

ikasketa prozesua, azterketa prozesua eta garapen prozesua daudelako.

Garapenaren barruan garapen prozesuari garrantzi handiagoa eman zaio, ezaugarri-diagrama eta kodearen etiketatzea eta murriztapenak egin behar direlako, hauen implementazioa egokia dela ziurtatzeko probekin batera, eta horrek denbora eskatzen du (105 ordu aurreikusten dira).

Oinarrizko hiru aplikazioen kodearen azterketan 40 ordu kontsumituko direla aurreikusten da. 10 ordu kodearen egitura (ezaugarri amankomunak eta desberdinak identifikatuz) aztertzen eta kodearen azterketan beste 30 ordu.

Ikasketa prozesuan beste 20 ordu jarri dira, proiektuaren garapenean (*pure::variants*) eta kodearen azterketan (JavaScript). Garapen prozesuko ordu guztiak batzen baditugu, 165 ordu ateratzen dira ataza horretan.

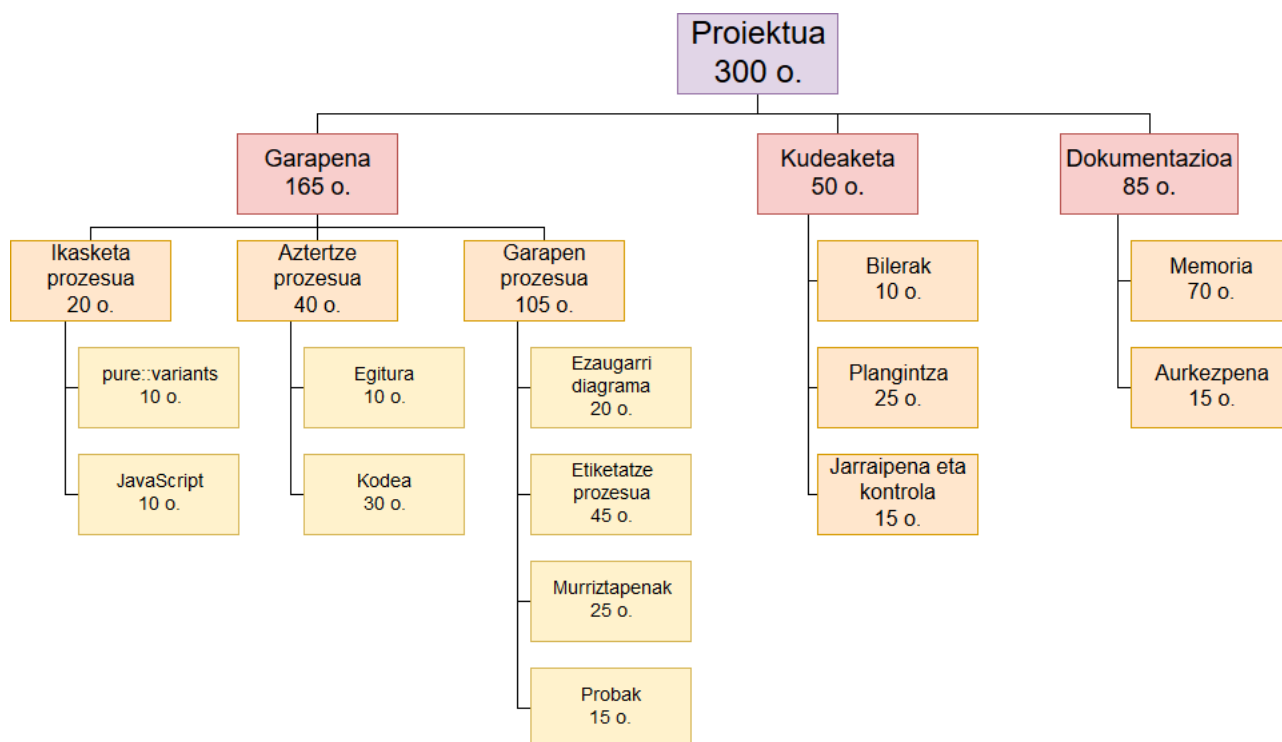
Beste aldetik, **kudeaketa** prozesua dago, bertan (2.3 atalean aipatu den bezala) bilerak, plangintza eta jarraipena eta kontrola sartzen dira. Gantt diagraman 7 bilera aurreikusi dira (ikusi 2.2 irudia) gutxienez, beraz 10 ordu guztira esleitu zaizkio ataza horri. Plangintza egiteko (irismena zehaztea, atazak identifikatzea eta deskribatzea, denbora estimazioa egitea, kalitatea plana egitea, lan metodologia zehaztea eta arriskuak aurreikustea) 25 ordu beharko direla uste da. Jarraipen eta kontrola proiektu osoan zehar egin behar denez beste 15 ordu esleitu zaizkio. Ordu kopurua batuz, 50 ordu lortzen dira kudeaketa atazarako.

Azkenik **dokumentazio** prozesua. Lehen esan den bezala, 2.3 atalean, bertan proiektuaren memoria (70 ordu) eta proiektuaren defentsarako aurkezpena (15 ordu) osatzen dute prozesu hau. Guztira 85 ordu aurreikusten dira.

2.5 Emangarriak

Ondorengoak dira Gradu Amaierako Lanaren emangarri eta muga ofizialak:

- **Web-anotazio aplikazioen *Software Produktu-Lerroa*:** emangarri honi *WebAnnotationSPL* izena eman zaio. Ekainaren 21erako ADDI plataforman entregatu behar da.
- **Proiektuaren memoria:** SPLarekin batera entregatu behar da (ekainaren 21ean).
- **Defentsarako aurkezpena:** proiektuen defentsak uztailaren 1 eta 12 bitartean egingo dira, beraz, data horietarako prest egon beharko da.



2.3 Irudia: denbora estimazioaren LDE diagrama.

Hala ere, beste emangarri ez ofizialak identifikatu daitezke, proiektuaren interesatuekin dauden konpromisoak, alegia:

- **Proiektuaren plangintza eta memoriaren sarrera:** proiektuaren plangintza eta memoriaren sarreraren lehenengo entrega otsailaren 15erako entregatu beharko da proiektuaren tutoreari.
- **Ezaugarri-diagrama:** aplikazioen ezaugarri-diagramaren lehenengo prototipoa martxoaren 4 eta 8 bitartean egingo den bileran erakutsiko da. Martxoaren 15erako, ezaugarri-diagramako ezaugarriak kodean identifikaturik egon beharko dira, eta hauek erakutsi eta eztabaidatuko dira proiektuaren tutorearekin edota aplikazioen garatzailearekin.
- **SPL-ren prototipoak:** SPL maiatzaren 24rako amaituta egon beharko da. Hala ere, apirilaren 29 eta maiatzaren 3 bitartean tutorearekin edo proiektuaren garatzailearekin ikusiko da nola doan proiektuaren garapena, behar diren aldaketak (behar badira) egiteko esfortzua aurrezteko.

- **Memoriaren bertsioak:** memoriaren azken entrega aurretik, tutoreari maiatzaren 24rako entregatuko zaio bertsio bat, aldaketak garaiz egin ahal izateko.
- **Aurkezpenaren prototipoak:** aurkezpenean erabiliko den materiala tutoreari ekainaren 21an erakutsiko zaio.

2.6 Kalitate-plana

Proiektuaren atazen deskonposaketan (ikus 2.1 irudiko LDE diagrama) erakutsi zen proiektua hiru adarretan banatzen dela: garapena, kudeaketa eta dokumentazioa. Kalitate-plana ere hiru zatitan banatuko da: **kudeaketaren** kalitate-plana, **produktuaren** kalitate-plana (garapenarekin lotuta) eta **dokumentazioaren** kalitate-plana.

2.6.1 Kudeaketaren kalitatea

Plangintzan (2. atal honetan) zehaztu diren irizpideak betetzen direla eta aldi berean, kudeaketaren kalitatea ziurtatzeko, hurrengo zerrendako atazak egingo dira (zerrendako betebeharrak *jarraipen eta kontrolaren* barruan sartzen dira):

- Ostiralero, proiektuaren Gantt diagrama berri batean, aste horretan egin diren atazak adieraziko dira.
- Egunero kontsumitutako orduak *Libre Office*ko kalkulu orri batean anotatuko dira.
- Bileraren ondoren, bileraren aktak idatziko dira, bertan landutako gaien informazioa ez galtzeko.
- Ostiraletan, denbora balioespenaren eta kontsumitutako denbora errearen arteko desbiderapenak begiratuko dira, desbiderapen handiak ekiditeko.

Proiektuaren plangintzan aurreikusi den denbora kontsumoa kontsumitu den denbora errearekin alderatuta gertatu diren desbiderapenen balorazioa egiteko, honako neurriak kontuan hartuko dira:

- %50 edo gehiagoko desbiderapena → kalitate baxua.
- %20 - %49 bitarteko desbiderapena → kalitate ertaina.
- %20 baino gutxiagoko desbiderapena → kalitate altua.

2.6.2 Produktuaren kalitatea

Produktuaren kalitate neurriak horrela neurtuko dira:

- SPLak oinarriko hiru aplikazioen funtzionalitateak lortzen baditu eta gainera funtzionalitate berri bat asmatzen bada → kalitate altua.
- SPLak hiru aplikazioen funtzionalitateak lortzen baditu → kalitate ona.
- SPLak *Review&Go* eta *Highlight&Go* aplikazioen funtzionalitateak lortzen baditu → kalitate ertaina.
- SPLak aplikazio bakar baten funtzionalitatea lortzen badu edo ez badu ezta bat lortzen → kalitate baxua.

Produktuaren kalitatea bermatzeko, produktua garatzen den bitartean, probak astero egingo dira.

2.6.3 Dokumentazioaren kalitatea

Dokumentuaren kalitatea bermatzeko Informatika Fakultateak eskaintzen duen eredia erabiliko da. Dokumentazioa idazteko erabiliko den tresna Latex izango da. Tresna hau oso egokia da ikerketa-lanak aurkezteko, eta nola ez, Gradu Amaierako Lanaren memoria egiteko. Dokumentazioan erabiliko den hizkuntza euskara izango da, eta ahal den hizkuntza egokiena eta formalena mantenduko da. Proiektuaren garapena modu argian azaltzea da memoria honen helburu nagusia.

Akats ortografikoak ekiditeko, Latexeko Xuxen *plugin*a erabiliko da. Gainera, dokumentazioa ondo idatzita dagoela (ulergarria dela eta hizkuntz egokia erabiltzen dela) bermatzeko, behin dokumentazioa amaiturik, honen bi irakurketa trinko egingo dira. Bestalde, dokumentazioko atal bakoitza bukatzen denean, proiektuaren tutoreari bidaliko zaio honen zuzentzeko.

2.7 Arriskuak

Atal honetan proiektuaren garapenean zehar gerta daitezkeen arazoak identifikatuko dira, eta ondoren, hauek ez gertatzeko, konponbide bat proposatuko da.

- **Datuak galtzea:** arrisku hau gertatzearen probabilitatea oso txikia da, baina, gerta daiteke. Gordetako datuak (proiektuaren memoria, proiektua bera, informazio baliagarria...) galtzeak plangintzan balioetsitako denbora kontsumoaren desbiderapen handia suposatuko luke. Lan guztia zerotik hasteak denbora asko eskatuko luke eta agian, proiektuaren garatzaileak ez litzateke iritsiko proiektua ekainean entregatze-
ra. Arazo hau ekiditeko, proiektuarekin zerikusia daukan guztia bai ordenagailu pertsonalean eta bai *Google Driven* gordeko da. Kodea ez galtzeko *GitHub* erabiliko da.
- **Komunikazioa moteltzea:** tutorearekin (Arantza Irastorza) eta oinarriko aplikazioen garatzailearekin (Haritz Medina) komunikazioak moteldu daitezke, eta honek proiektuan atzerapenak ekar ditzake. Beraz, bilerak denbora nahikoarekin antolatuko dira posta elektronikoaz baliatuta.
- **Kalitate irizpideak ez betetzea:** gerta daiteke 2.6 atalean zehaztu diren kalitate irizpideak ez lortzea. Hau ez gertatzeko, hilean behin kalitate irizpideak betetzeko plana behar den bezala jarraitzen dela konprobatu beharko da.
- **Garapena atzeratzea:** aurreko arriskuren bat gertatzen bada proiektuaren garapena atzeratu edo moteldu daiteke, eta honek proiektua garaiz ez entregatzea suposa dezake. Hala ere, ez da aurreikusten hau gertatzea plangintzan zehaztutako irizpideak errespetatzen badira.
- ***pure::variants* tresnarekin arazoak edukitzea:** tresna honekin hainbat arazo egon daitezke:
 - **Tresna erabiltzen ez jakitea:** graduan ez da tresna hau irakasten, beraz, GrAL honetan erabiliko den tresnaren funtzionamendua ikastea, proiektua egingo duen ikaslearen esku gelditzen da. Hala eta guztiz ere, ez da aurreikusten arazo handirik edukitzea, baina, edukiz gero, Raul Medeirosekin (proiektuaren hasieratze bileran *pure::variants* aurkeztu zuen pertsonarekin) bilera bat antola daiteke, aurretik interneten kontsultatuta ez bada ezer aurkitu.
 - **Lizentzia garaiz ez lortzea:** tresna hau erabiltzeko lizentzia behar da. Hasierako ebaluazio lizentziak hilabete bat irauten du, beraz, lizentzia bat eskatu beharko da. Lizentzia hori lortzeko arazoak egonez gero, proiektuaren tutoreari abisatu beharko zaio.

- **Lizentziak funtzionalitatea murriztea:** lortuko den lizentzia, ikasle bate-na izanik, funtzionalitatean murriztapenak izan ditzake. Hala bada, ezingo da gauza handirik egin.

- **JavaScripten idatzitako kodea ez ulertzea:** EHUko informatikako graduari ez da *JavaScript* lengoaiari asko ikusten, horregatik, arazoak eta atzerapenak gerta daitez-ke aplikazioen kodea aztertzerakoan. *JavaScript* lengoaiari buruz informazio ugari lortu daiteke interneten, baina, laguntza gehiago behar izanez gero, Haritz Medina-rekin bilera bat antolatu daiteke.

- **Haritz Medinak *Mark&Go* aplikazioa garaiz ez bukatzea:** *Mark&Go* aplikazioa oraindik garatzen ari da Haritz Medina, beraz, gerta daiteke garaiz ez bukatzea. Honek SPLrako bi aplikazio soilik edukitzea suposatuko luke. Ikasleak ezingo du ezer egin arrisku hau ekiditeko, Haritz Medinaren menpe baitago.

2.8 Informazio eta komunikazio sistemak

Haritz Medinak egindako hiru aplikazioen iturburu-kodea *GitHub*etik eskuratu da. *Highlight&Go*, *Mark&Go* eta *Review&Go* aplikazioak *Chrome* nabigatzailearen web-dendatik lortu dira (*Chrome Web Store*), lehenengo biak publikoki argitaratuta daude eta hauek instalatzeak ez du arazorik suposatzen, baina *Review&Go* ez dago publikoki argitaratuta, beraz, hau instalatzeko Haritz Medinaren baimena behar izan da.

Proiektuarekin zerikusia duen guztia, bai ordenagailu pertsonalean, baita *Google Driven* gordeko da. Hainbat karpitetan antolatuko da proiektua: informazio karpeta (proiektuarekin zerikusia duen informazioa gordetzeko erabiliko dena), proiektuaren plangintzarako egindako diagramak gordetzeko erabiliko den karpeta, dokumentazio karpeta (memoria gordetzeko) eta azkenik, proiektuaren garapenarekin zerikusia duen guztia gordetzeko karpeta.

Proiektuaren zuzendariarekin (Arantza Irastorza) eta aplikazioaren garatzailearekin (Haritz Medina) komunikatzeko posta elektronikoa erabiliko da gehien bat, baina, duda zehatzagoak asetzeko bilerak antola daitezke.

2.9 Interesatuak

Proiektu honen interesatu handiena edo printzipala Haritz Medina da, web-anotazio aplikazioen garatzailea. Beste aldetik, proiektuaren zuzendaria eta proiektuaren defentsa ebaluatuko duten irakasleak ditugu. Lehenengoaren kasuan, proiektua bideratu behar duenez, bere interesa argia da; bigarrenen kasuan, proiektua ebaluatu behar dute, beraz, interesa argia da ere.

Azkeneko interesatua lana aurrera daraman ikaslea da, bere esku baitago proiektu honen arrakasta, eta epe-mugak betetzen direla ziurtatzea.

3. KAPITULUA

Proiektuaren birplangintza

Atal honetan momentu honetaraino (apirilaren bukaeraraino) Gradu Amaierako Lanean sartutako denbora eta birplangintzaren beharra sortu duten arazoak aurkeztuko dira, hauen ondorioz erabakitako helburu berriekin eta denbora estimazio berriarekin batera.

3.1 Apirilaren 29ra arte dedikatutako orduak

Apirilaren 29an lanari 244 ordu dedikatu zaizkio. Hasierako plangintzan egindako estimazioa (2. kapituluarena) 300 orduko dedikazio totalarekin egin zen, baina, lanaren puntu honetan argi dago denbora gehiago behar izango dela.

3.1 irudian ikusi daiteke hainbat atazetan estimatutakoa baino denbora gehiago kontsumitu dela: aztertze prozesuko egituran eta kodean, garapen prozesuko etiketatze prozesuan eta probetan, eta azkenik, kudeaketa prozesuko plangintzan. Gainera, ikasketa prozesuan beste ataza bat agertu da ere, *Chrome developer* deiturikoa.

Orain arte kontsumitutako denbora kontuan edukita, eta hurrengo ataletan aurkeztutako arazoak (3.2 atala) eta tutorearekin adostutakoa kontuan edukita (3.3 atala), birplangintza bat egitea erabaki da.

ATAZAK	ESTIMAZIOA	ERREALA	DESBIDERAPENA
Garapena	165	168	3
Ikasketa prozesua	20	14,5	-5,5
pure::variants	10	9	-1
JavaScript	10	2	-8
Chrome developer	0	4,5	4,5
Aztertze prozesua	40	54	14
Egitura	10	11	1
Kodea	30	43	13
Garapena prozesua	105	82,75	-22,25
Ezaugarri diagrama	20	18,5	-1,5
Etiketatzeko prozesua	45	57,25	12,25
Murriztapenak	25	7	-18
Probak	15	16,75	1,75
Kudeaketa	50	52,5	2,5
Bilerak	10	7,5	-2,5
Plangintza	25	35,5	10,5
Jarraipena eta kontrola	15	9,5	-5,5
Dokumentazioa	85	23	-62
Memoria	70	23	-47
Aurkezpena	15	0	-15
GUZTIRA	300	243,5	-56,5

3.1 Irudia: Apirilaren 29ra arte dedikatutako denbora.

3.2 Agertutako arazoak

Lau izan dira birplangintza honen beharra sortu duten arazoak. Atal honetan laburki aipatu-tuko dira, azalpen zabalago bat nahi izanez gero ikusi 9. kapitulua.

- **Beste garatzaile baten kodearekin lan egitearen zailtasuna:** zailtasun honek aztertze prozesuan estimatutakoa baino denbora gehiago behar izatea suposatu du. Aurreikusitakoa baino 14 ordu gehiago guztira.
- **pure::variantseko Family Modelaren konfigurazio arazoak eta informazio falta:** honek garapen prozesuan eta ikasketa prozesuan eragina izan du. *Family Model*arekin gertatzen ari zen arazoa aurkitzeko 15 ordu inguru behar izan dira.
- **Google Chromeko debuggerarekin esperientzia falta eta Haritz Medinaren laguntzaren beharra:** hauek garapen prozesua moteldu dute, horregatik garapen prozesurako denbora gehiago esleituko da birplangintza honetan.

3.3 GrALaren tutorearekin adostutako erabakiak

Gradu Amaierako Lan honi ekaineko deialdian entregatzeko lehentasuna eman zaio ikaslearen beharrak direla medio, beraz, gertatutako arazoak eta orain arte kontsumitutako denbora kontuan edukita lehenengo plangintzan aldaketak egin behar direla erabaki da ikaslearen eta tutorearen artean.

Hasierako 300 orduak baino gehiago behar izango direla argi dago, horregatik birplangintza 380 orduekin egingo da. Gainera, inplementazioaren esfortzua handia dela (ordu asko behar direla) jakinda, GrALaren helburuak birplanteatu dira.

3.4 Helburu berriak

2. kapituluaren SPLarekin hiru aplikazioak lortu nahi ziren, baina, orain arte gertatutako arazoak eta zailtasunak kontuan hartuta *Mark&Go* aplikazioarekin lotuta dauden ezaugarriak alde batera uztea erabaki da.

3.5 Denbora estimazio berria

Denbora estimazio berria egiteko hainbat gauza hartu dira kontuan (ikusi [3.2](#) irudia):

- **Ikasketa prozesuan:** *Chrome developer* ataza ez zen aurreikusi lehenengo plangintzan, beraz, ataza honetan kontsumitutako orduak batu behar dira. Bestalde, *JavaScript*eko ikasketa prozesurako estimatutako denbora baino 5 ordu gutxiago aurreikusten dira orain, eta *pure::variants* ikasketa prozesuari 2 ordu gehiago gehitu zaizkio.
- **Aztertze prozesua:** prozesu hau bukatuta dago dagoeneko, baina, lehenengo plangintzan estimatutako denbora baino gehiago behar izan da. Egituraren azterketarako estimatutakoa baino ordu 1 gehiago eta kodearen azterketarako 13 ordu gehiago.
- **Garapen prozesua:** prozesu honetan gertatu dira aldaketa nagusienak, murriztapentzako espero zen denbora baino gutxiago behar dela erabaki da (10 ordu gutxiago), etiketatze prozesuarentzako eta probentzako, aldiz, espero zena baino gehiago (45 ordu eta 15 ordu hurrenez hurren).

- **Kudeaketa:** Gertatu daitezkeen arazoak direla eta, bilerentzako 3 ordu gehiago esleitu dira eta birplangintza hau egiteko beste 3 ordu gehiago (3 ordu hauei, lehenengo plangintza egiteko gertatutako 7.5 orduko desbiderapena batu behar zaie).

ATAZAK	PLANGINTZA	BIRPLANGINTZA	DESBIDERAPENA	BUKATUTA
Garapena	165	232	67	EZ
Ikasketa prozesua	20	23	3	EZ
<u>pure::variants</u>	10	12	2	EZ
JavaScript	10	5	-5	EZ
Chrome <u>developer</u>	0	6	6	EZ
Aztertze prozesua	40	54	14	BAI
Egitura	10	11	1	BAI
Kodea	30	43	13	BAI
Garapena prozesua	105	155	50	EZ
Ezaugarri diagrama	20	20	0	EZ
Etiketatze prozesua	45	90	45	EZ
Murriztapenak	25	15	-10	EZ
Probak	15	30	15	EZ
Kudeaketa	50	63	13	EZ
Bilerak	10	13	3	EZ
Plangintza	25	35	10	BAI
Jarraipena eta kontrola	15	15	0	EZ
Dokumentazioa	85	85	0	EZ
Memoria	70	70	0	EZ
Aurkezpena	15	15	0	EZ
GUZTIRA	300	380	80	EZ

3.2 Irudia: Denbora estimazio berria.

4. KAPITULUA

Erabilitako tresnen azterketa

Atal honetan Gradu Amaierako Lana garatzeko erabili diren tresnak aurkeztuko dira. Atal hau bi zatitan banatuko da, lehenengo zatian kodearen analisirako erabili diren tresnak aurkeztuko dira eta bigarreanean SPL kodearen garapenerako erabili den tresna.

4.1 Kodearen analisirako tresnak: *DiffMerge* eta *BUT4Reuse*

Kodearen analisisian bi urrats identifikatzen dira, alde batetik hiru aplikazioen (*Highlight&Go*, *Mark&Go* eta *Review&Go*) **arkitektura edo egitura alderatzea**, eta beste aldetik **kode barruko desberdintasunak konparatzea**, hau da, nahiz eta hiru aplikazioek *script* berdina eduki barruko edukia desberdina izan daiteke, eta desberdintasun horiek identifikatu behar dira etiketatze prozesua egiteko.

4.1.1 BUT4Reuse

Tresna honen bitartez jatorrizko hiru aplikazioen kodearen egitura aztertu da, honen alde amankomunak eta bakoitzaren berezitasunak identifikatuz. ***Bottom-Up Technologies For Reuse*** edo ***BUT4Reuse*** aldaki edo produktuen aldakortasuna aztertzeko erabiltzen den iturburu irekiko softwarea da. Mota desberdin askotako artefaktuak soportatzen ditu: Java, C, EMF Models, Textual Files, File structures, JSON, CSV, besteak beste (***BUT4Reuse***).

Software honek aplikazioaren egitura hainbat bloketan banatzen du hiru aplikazioen egi-

turaren desberdintasunaren arabera. Lan honetan software hau *app* direktorio barruko edukiaren egitura alderatzeko erabili da. Guztira 7 bloke desberdin lortu dira:

- **Block 0:** hiru aplikazioek berdina duten egitura.
- **Block 1:** *Highlight&Go* eta *Mark&Go* aplikazioen berdintasunean.
- **Block 2:** *Highlight&Go* eta *Review&Go* berdina duten egiturarekin.
- **Block 3:** *Mark&Go* eta *Review&Go* aplikazioen berdintasunean.
- **Block 4:** *Highlight&Go* berezitasunak.
- **Block 5:** *Mark&Go* berezitasunak.
- **Block 6:** *Review&Go* berezitasunak.

4.1.2 DiffMerge

SourceGear konpainiak sortutako softwarea da hau. *DiffMergeren* helburua bi edo hiru fitxategien edukien arteko konparazioak egitea da. Windows Explorer nabigatzailean integratzeko aukera dago ere ([SourceGear, 2017](#)). Hiru funtzionalitate nagusi bereizten dira:

- **Diff:** bi fitxategien edukien konparaketa.
- **Merge:** hiru fitxategien edukien konparaketa.
- **Folder diff:** bi direktorioen egituren arteko konparaketa.

Tresna hau Haritz Medinaren aplikazioen (*Highlight&Go*, *Mark&Go* eta *Review&Go*) **kodea konparatzeko** erabili da. Alabaina, tresna honek baditu bere **mugak eta arazoak**. Jarraian hauek aztertuko dira:

- **Testuaren ordenak garrantzia du:** fitxategiaren edukiaren ordenak garrantzia du. Adibidez, A eta B direktorioen *scriptak* konparatzen baditugu, nahiz eta biek funtzio berdinak eduki funtzio horiek orden desberdinetan badaude, bien arteko edukia desberdina dela detektatuko du tresna honek, hau da, A direktorioko *scriptak* lehenengo *Funtzio1* badauka eta ondoren *Funtzio2* eta B direktoriokoak *Funtzio2*

eta gero *Funtzio1* badauka orden horretan hurrez hurren, nahiz eta bi *scriptak* funtzio berdinak eduki *DiffMergek* hauek koloreztatuko ditu desberdinak direla esanez. Kasu batzuetarako interesgarria izan daiteke desberdintasun hori, baina lan honetan horrek denbora gehiago eskatzen du, badaezpada desberdintasun horiek orden arazoak diren begiratu behar delako.

- **Folder diffen mugak:** lan honetan hiru aplikazioen karpeten egitura aztertu nahi denez, eta funtzionalitate honek bi karpeten egitura soilik konparatzen uzten duenez, tresna hau ez da egokiena, egokiagoa da aurreko azpiatalean (4.1.1 azpiatalean) aipatzen den *Eclipseko BUT4Reuse* software integratua.

4.2 SPL kodearen garapenerako tresna: *pure::variants*

Eclipsen integratutako garapen-ingurune bat da, Software Produktu-Lerroak garatzeko balio duena (*pure::variants*). SPLen garapenean hiru prozesu nagusi identifikatzen dira:

1. **Domeinuaren analisia:** fase honetan produktuen **ezaugarri amankomunak eta berezitasunak identifikatzen dira**. Lan honen kasuan, jatorrizko hiru aplikazioen (*Highlight&Go*, *Mark&Go* eta *Review&Go*) ezaugarri amankomunak eta bereziak (propioak) aztertzea izango litzateke. Prozesu honetan aurreko atalean (ikusi 4.1 atala) azaldutako tresnak erabili dira: *DiffMerge* eta *BUT4Reuse*.
2. **Domeinuaren diseinua:** prozesu honetan SPLaren **ezaugarri-diagrama garatzen da**, aurreko pausoan identifikatu diren ezaugarri amankomun eta bereziekin, dagokien mendekotasunekin. Horretarako *pure::variantsek* eskaintzen duen *Feature Modela* erabili da (ikusi 4.3 atala).
3. **Domeinuaren inplementazioa:** prozesu honetan **jatorrizko fitxategiak ezaugarri-diagramarekin lotzen dira**. Lan honen kasuan, jatorrizko fitxategiak (etiketatutako JavaScript, CSS, HTML eta JSON fitxategiak; ikusi 4.5 azpiatala) *pure::variantsek* eskaintzen duen *Family Modela* erabiliz (ikusi 4.4 atal) ezaugarri diagramarekin (aurreko urratsean lortutako *Feature Modelarekin*) erlazionatuko da.
4. **Aldakien sorkuntza:** SPLtik produktuak lortzeko *pure::variantsko Variant model* erabiliko da lan honetan. *Variant model* horrek, **ezaugarri diagramako aukeratu diren ezaugarrien arabera, produktu bat edo beste bat sortuko du**.

4.3 *Feature Model*

Aurreko atalean aipatu den moduan, *pure::variants*-en ezaugarri-diagramak *Feature Model* bidez inplementatzen dira. Ezaugarri-diagramak Software Produktu-Lerroen aldakortasuna irudikatzeko erabiltzen dira. Honek zuhaitz baten antza du eta zuhaitz horren nodo bakoitza SPLko ezaugarri batekin lotzen da, hau da, ezaugarri bakoitzak SPLko ezaugarri bat irudikatzen du.

Ezaugarriak honako motetan sailkatu daitezke:

- **Derrigorrezko ezaugarriak (*mandatory*):** hauek produktu guztiek berdinak dituzten ezaugarriak dira, hau da, ezaugarri amankomunak. *pure::variants*en ezaugarri hauek harridura ikurrarekin adierazten dira (!).
- **Hautazko ezaugarriak (*optional*):** izena berak esaten duen moduan, hautazkoak diren ezaugarriak dira hauek, hau da, produktu batzuek eta ez beste batzuek izango dituzten ezaugarri bilduma. *pure::variants*en hautazko ezaugarriak galdera ikurrarekin adierazten dira (?).
- **Or-esklusiboa (*alternative*):** mota honetako ezaugarri bat aukeratzen denean, beste guztiak baztertzen dira. Beste modu batera esanda, ezaugarri multzo batetik ezaugarri bat eta soilik bat aukeratzen da. *pure::variants*en bi noranzko gezi batekin adierazten dira (\leftrightarrow).
- **Or-inklusiboa (*or features*):** or-inklusiboa osatzen duten ezaugarrien multzotik ezaugarri bat edo gehiago aukeratu daitezke. Ezaugarri mota hau gurutze batekin irudikatzen da *pure:variants-en* (x).

Gerta daiteke ezaugarri bat egon ahal izateko beste ezaugarri bat egon behar izatea, hau da, **ezaugarrien artean mendekotasunak egon daitezke**. *pure::variants*en mendekotasunak *Relation*ekin definitzen dira.

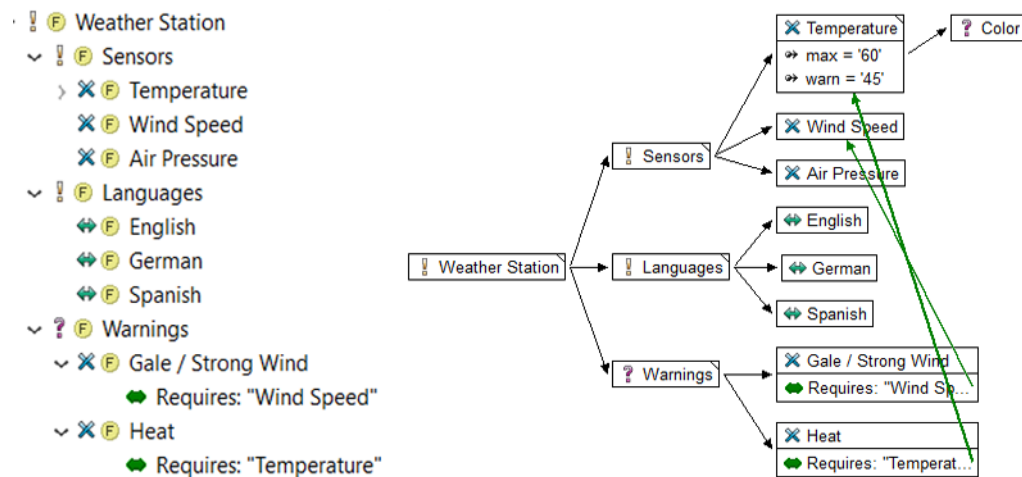
Jarraian erakutsiko den adibidearekin aipatutako kontzeptuak hobeto barneratuko dira. Erabiliko den adibidea, *pure::variants*en 4.0.16 bertsioak eskaintzen duen *Weather Station Example* proiektua izango da. Hau lortzeko *File*→*New*→*Other..* ireki behar da eta bertan, *Examples* karpeta barruko *Variant Management* karpeta *Weather Station* proiektua aukeratu.

Proiektuko “.x fm ” luzapena duen fitxategia *Feature Modela* da. 4.1 irudian ikusi daiteke *Feature Modelak* hartzen duten itxura; ezkerrean zuhaitz moduan eta eskuinaldean modu grafikoan.

Ezaugarri bakoitzak ikur bat du ondoan, horiek dira ezaugarri mota adierazten duten ikurrak, lehen aipatutakoak: *mandatory(!)*, *optional(?)*, *or(x)* eta *alternative(↔)*. Azkeneko bi ezaugarrietan (*Gale / Strong Wind* eta *Heat*) mendekotasunak ezarri dira: *Gale / Strong Wind* ezaugarria soilik *Wind Speed* aukeratu bada aukeratu daiteke, eta *Heat* kasuan, *Temperature* ezaugarria aukeratu bada.

Gradu Amaierako Lan honetan bi mendekotasun (*Relation*) mota erabiliko dira:

- **A Requires B:** A ezaugarria egoteko B ezaugarria egon behar da. Hau da adibidean ikusi den mendekotasuna.
- **A Conflicts B:** A ezaugarria aukeratu nahi bada, B ezaugarria ezin da aukeratu egon.



4.1 Irudia: Weather Station SPL-ren *Feature Model-a*: Ezkerrean zuhaitz moduan (*Tree*) ikusi daiteke eta eskuinaldean modu grafikoan (*Graph*).

4.4 *Family Model*

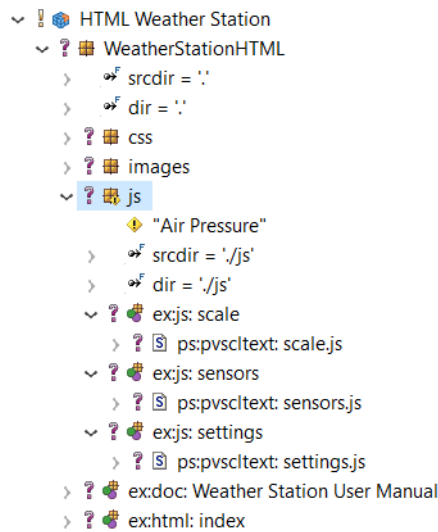
Family Modelak Feature Modelan definitutako ezaugarriak *input* karpetan sartutako eta aurretik etiketatu (ikus 4.5 atala) den iturburu-kodearekin erlazionatzen du. Esan daiteke *Family Modelak* lortu nahi den software produktuaren kodearen egitura definitzen duela, eta estruktura hori, ezaugarriekin lotzen dela murriztapenen bidez (*Restrictions*). Horrela, aukeratzen diren ezaugarrien arabera, kodeak egitura bat edo beste bat edukiko du.

*Family Model*etan bost elementu nagusi identifikatzen dira, lan honetan erabiliko direnak, alegia (ikus 4.2 irudia):

- **Osagaiak (*Components*):** osagaietan atalak multzokatzen dira. 4.2 irudiko adibidean *js* osagai bat da eta bere barruan *JavaScript*eko atal guztiak sartzen dira.
- **Atalak (*Parts*):** kodearen elementuak irudikatzen dituzte. Hauen barruan iturburu-kodeak sartzen dira. 4.2 irudian *scale*, *sensor* eta *settings* atalak dira.
- **Iturburu-kodeak (*Source*):** iturburu-kodeak *input* karpetako kode-fitxategiekin lotzen diren elementuak dira. 4.2 irudian *scale.js*, *sensor.js* eta *settings.js* iturburu-kodeak dira. Bi motako iturburu kode aipatuko dira 4.5 atalean: *pvscltext* eta *pvsclxml*.
- **Atributuak (*Attributes*):** lan honetan atributuak direktorio eta fitxategien kokapena adierazteko erabili dira. 4.2 irudiko adibidean *js* osagaiaren atributuak *dir* eta *srcdir* dira, eta hauek, *js* iturburu-karpetari erreferentzia egiten diote. Iturburu-karpeta horretan iturburu-kodeak edo iturburu-fitxategiak daude, hau da, azkeneko produktuek behar duten fitxategi, *script* eta liburutegi guztiak.
- **Murriztapenak (*Restrictions*):** murriztapenek ezaugarrien aukeraketan arabera osagai eta atalak sortzea ahalbidetzen dute. Adibidez, 4.2 irudian *js* osagaiaren barnean definitutako *Air Pressure* murriztapenarekin adierazten da, *Air Pressure* ezaugarria aukeratzen bada, *js* osagaia sortuko da, kontrako kasuan, ez da sortuko.

4.5 *Tagging*

Etiketak edo *tagak* kode barruko edukia aukeratutako ezaugarrien arabera sortzeko erabiltzen dira. Esaterako, klase bateko funtzio bat A ezaugarriari lotuta badago, etiketa bat



4.2 Irudia: *pure::variants*-eko *Family Model* baten adibidea.

jarri behar da azkeneko produktuan A ezaugarria aukeratzen bada, funtzio hori amaierako produktuaren barruan agertzeko, eta kontrako kasuan, ez agertzeko.

Aurreko atalean (4.4 atalean) aipatu den moduan, bi *tagging* mota desberdintzen dira lan honetan: *pv:scitext* eta *pv:scxml*. Hala ere, beste motako *tagak* badaude ere.

4.5.1 pv:scxml

XML familiako dokumentuetan erabiltzen diren etiketak dira. Adibidearen eta GrALaren kasuan, HTML orrietan erabiltzen diren etiketak izango dira. Etiketa hauek definitzeko, edozein HTML elementu barruan *pv:condition*="baldintza" idatzi behar da. *Baldintzek* 4.4 atalean landu diren murriztapenen (*Restrictions*) antza hartzen dute.

Adibidearen kasua landuz, *input* karpetakako *index.html* orriaren iturburu kodean hiru baldintza definitu dira:

- *pv:condition*="AirPressure": Honekin, *AirPressure* ezaugarria aukeratuz gero, elementu horren barruko elementuak sortuko dira. Kontrako kasuan ez.

```

1 <div class="display pressure" pv:condition="AirPressure">
2   <div id="pressure">
3     ...

```

```

4     </div>
5 </div>

```

Amaierako kodean *pv:condition* atributua duen etiketa mantentzen da ere, hau da, *AirPressure* ezaugarria aukeratuta badago, amaierako produktuan *pv:condition=“AirPressure”* atributua duen *div* elementua eta bere azpi-elementuak sortuko dira.

- *pv:condition=“Temperature”*: Honekin, *Temperature* ezaugarria aukeratzeko bada, HTML elementu berruko elementu guztiak sortuko dira.
- *pv:condition=“WindSpeed”*: Aurreko kasuekin gertatzen den moduan, berdina gertatuko da, baina kasu honetan *WindSpeed* ezaugarria aukeratzeko bada.

4.5.2 pvscitext

Etiketatzeko modu hau gainontzeko iturburu-kodeetan erabili da; CSS, JavaScript eta JSON fitxategietan, alegia. *If* baldintza baten egitura berdina hartzen dute mota honetako *tagek* (kortxeteen barruan dauden klausulak aukerakoak dira, besteak nahitaez agertu behar dira):

```

PVSCSL:IFCOND(baldintza)
// Kodea
[PVSCL:ELSEIFCOND(baldintza2)]
// Kodea
[PVSCL:ELSECOND(baldintza3)]
// Kodea
PVSCL:ENDCOND

```

Weather Station adibidean, *input/js* direktorioko *scale.js* fitxategian honako etiketak aurkitzen dira *initiateScales()* funtzioaren barruan:

```

1 function initiateScales() {
2     // PVSCL:IFCOND(AirPressure)
3     var parent = document.getElementById('p_main');
4     setScale(parent, minPres, maxPres, presScale, 14);
5     // PVSCL:ENDCOND
6

```

```

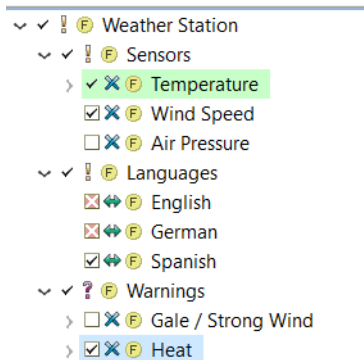
7 // PVSCL:IFCOND(Temperature)
8 var parent = document.getElementById('t_scale');
9 setScale(parent, minTemp, maxTemp, tempScale, 6);
10 // PVSCL:ENDCOND
11
12 // PVSCL:IFCOND(WindSpeed)
13 var parent = document.getElementById('w_main');
14 setScale(parent, minWind, maxWind, windScale, 14);
15 // PVSCL:ENDCOND
16 }

```

AirPressure, *Temperature* edo *WindSpeed* aukeratzeko gero, kode zati bat edo beste bat sortuko da.

4.6 Variant model

Variant modelak ezaugarrien aukeraketak dira, aukeraketa horren ondorioz produktu desberdinak lortzeko asmoarekin. Simple bistaz, *Feature Modelaren* antza du, baina, kasu honetan ezaugarriek ezaugarri bera aukeratzeko lauki bat dute ondoan (*checkbox* bat). Lauki horiek blokeatzen dira *Feature Modelen* definitutako mendekotasunen (*relations*) arabera.



4.3 Irudia: Variant model baten aukeraketa.

4.3 irudiko *Variant modelan* ikusi daiteke *Heat* ezaugarria aukeratzeko bada *Temperature* ezaugarria aukeratuta egon behar dela, erlazio hori berdez agertzen da. Baita ere erreparratu daiteke, nola hizkuntza bat aukeratzeko denean beste hizkuntzak blokeatzen direla, or-esklusiboa dela medio.

Behin ezaugarriak aukeratuta, amaierako produktua lortzeko, bilakatzailera konfiguratu

behar da. Bilakatzailea konfiguratzeke lehenengo urratsa, erabili nahi diren *Feature Model* eta *Family Model*ak aukeratu behar dira. Ondoren, iturburu kodea gordetzen duen aukeratu, eta baita ere, produktua zein karpetan sartu nahi den adierazi. Azkenik, bilakatzailearen transformazio moduluak aukeratu behar dira. Gutxienez hauek bi aukeratu behar dira: *Action List Generator* eta *Action List runner*. Hauek aurreko urratsetan egindako aukeraketen eta proiektuaren arteko lotura egiten duten XML fitxategiak dira.

Bilakatzailea konfiguratuta dagoela, *Variant modelaren* gainean eskuineko botoia sakatu behar da eta *Variant→Transform* menu zabalgarrian konfiguratuta den bilakatzailea aukeratu.

5. KAPITULUA

Iturburuko aplikazioen domeinua eta funtzionalitateak

Kapitulu honetan Haritz Medinak garatutako *Highlight&Go*, *Mark&Go* eta *Review&Go* aplikazioen domeinua eta funtzionalitateak aztertuko dira.

Sarreran aipatu den moduan (ikusi 1. kapitulua), hiru aplikazio hauek Web Anotazioan oinarritutako aplikazioak dira (Wikipedia, 2019b), zehatzago, *Hypothes.is* aplikazioan oinarrituta (Wikipedia, 2018b). Kapitulu horretan aipatutakoa gogoratu, Web Anotazioko aplikazioak web-orrietako edukia azpimarratzeko, markatzeko (iruzkinak jartzeko, adibidez) eta partekatzeko egokiak dira, eta *Hypothes.is* mota horretako aplikazioetatik erabiliena eta ospetsuena da *A.nnote*kin batera.

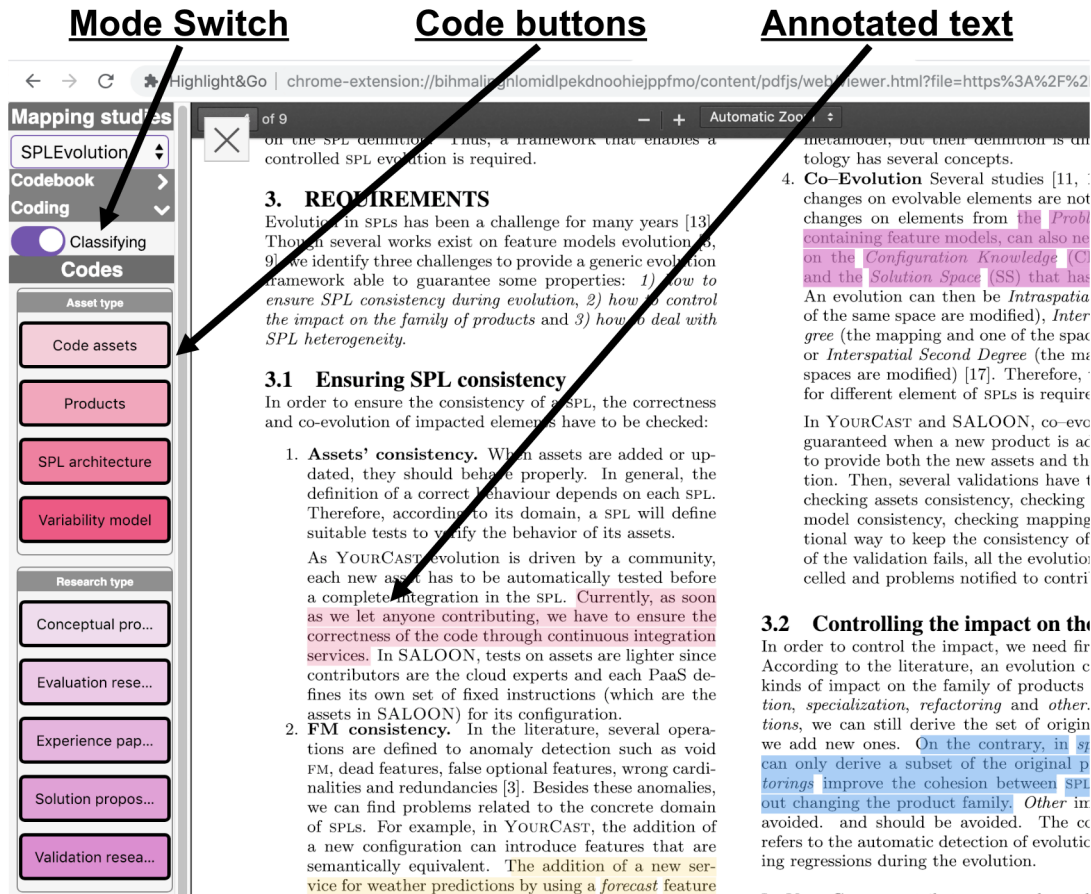
Iturburuko hiru aplikazioek, nahiz eta domeinu berdintsuetan kokatuta egon, badute beraien arteko desberdintasunak funtzionalitateei begira, gehien bat.

5.1 Highlight&Go

Aplikazio honek **ikerketa lanetan egiten den informazio bilketa web anotazioen bidez erraztea du helburu** (Medina u. a., 2019a). Aurredefinitutako irizpideen arabera dokumentuak markatzen dira ikerketa taldeko beste pertsonak hauek berrikusteko, ondoren anotazio horiek balidatzeko.

Aplikazio hau Googleko *Spreadsheet*aren bidez **konfiguratzen** da. Dokumentu horren taulan, ikertu nahi diren dokumentuetara bidaltzen duen esteka bat jarri behar da taulako lehenengo zutabean, eta taulako lehenengo lerroan irizpideak konfiguratzeko daitezke. On-

doren, dokumentuak anotatzen diren bitartean, anotazio horiek taula horretan bistaratuko dira.

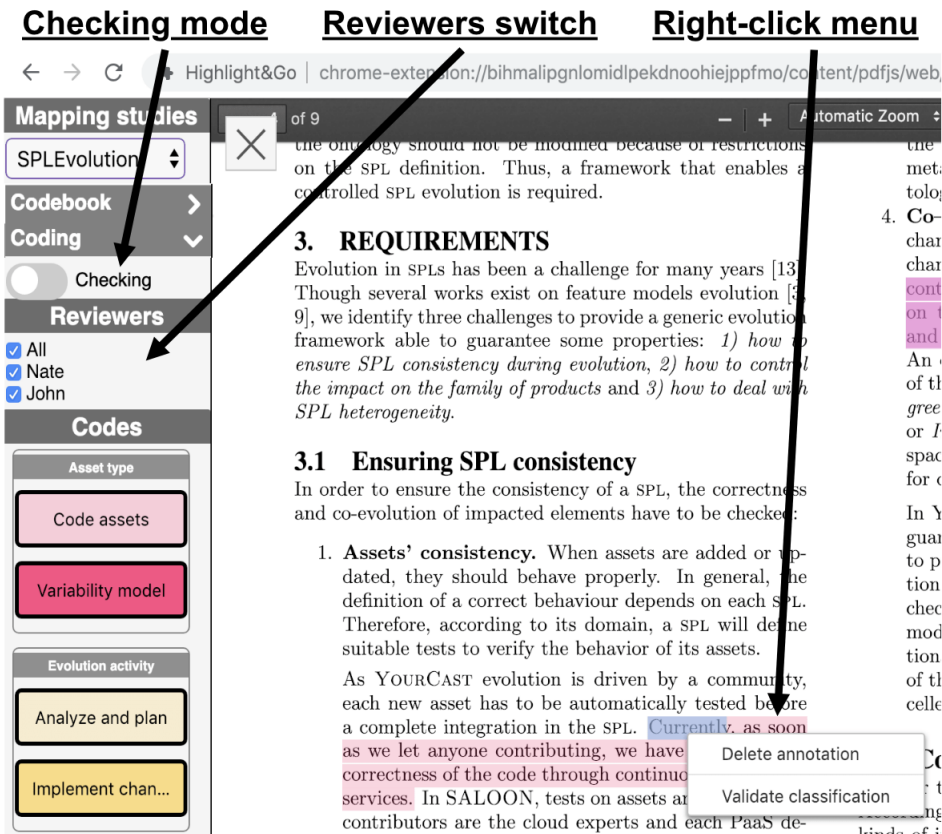


5.1 Irudia: *Highlight&Goko Classifying* modua. Irudia Medina u. a. (2019a) artikulutik lortu da.

Google Chromeko luzapen honen **interfaze grafikoaren** itxura 5.1 irudian ikusi daiteke. Ikusten denez, ikertzen den dokumentua hedatzen da *sidebar* edo alboko menu batekin. Goiko aldetik hasita, menu horretako lehenengo aukera erabiltzailearen *Hypothes.is*eko taldea aukeratzekoarena da. Ondoren, motaz aldatzeko aukera dago, aplikazio honek bi modu desberdin eskaintzen dituelako: lehenengo modua *Classifying* modua da (SPLan *Highlight* izena hartuko du), gehienbat, dokumentua azpimarratzeko balio duena, eta bigarrena, *Checking* modua (SPLan *Index* izena hartuko du), anotazio horiek balidatzeko erabiltzen dena.

Classifying moduan ikerketa irizpideak kolore desberdinez agertzen dira. *Checking* modua zertxobait desberdina da, bertan dokumentua anotatu duten ikertzaileen zerrenda bat agertzen da, hauek egindako anotazioak ezkutatu edo agertzeko, eta azpimarraketen aur-

kibide moduko bat aurkitzen da ere. Modu honen aukerak 5.2 irudian ikusi daitezke.



5.2 Irudia: *Highlight&Goko Check* modua. Irudia Medina u. a. (2019a) artikulutik lortu da.

Aplikazio honen **funtzionalitateak** aztertuko dira jarraian. Bi modu desberdin aurkitzen dira:

- **Mapping mode:** (ikusi 5.1 irudia)
 - *Hypothes.iseko* taldea aukeratu.
 - Google *Spreadsheet* orrira bueltatzeko aukera.
 - Azpimarratu eta anotazioak sailkatu Google *Spreadsheeten* aurredefinitutako irizpideen arabera.
 - *Check* modura aldatzeko aukera.
 - Anotazioak ezabatu.
- **Check mode:** (ikusi 5.2 irudia)

- *Hypothes.is*eko taldea aukeratu.
- Google *Spreadsheet* orrira bueltatzeko aukera.
- *Mapping* modura aldatzeko aukera.
- Web orria azpimarratu duten erabiltzaileen zerrendatik nahi diren azpimarratzaileak aukeratu beraien anotazioak soilik ikusteko.
- Aukeratutako azpimarratzaileek anotazioak egiteko erabili dituzten irizpideen zerrenda agertuko da, eta zerrenda horretako irizpide bakoitzari klik eginez, irizpide horrekin sailkatutako anotazioak ikusi daitezke.
- Anotazioak ezabatu.
- Anotazioak balidatu.

5.2 Mark&Go

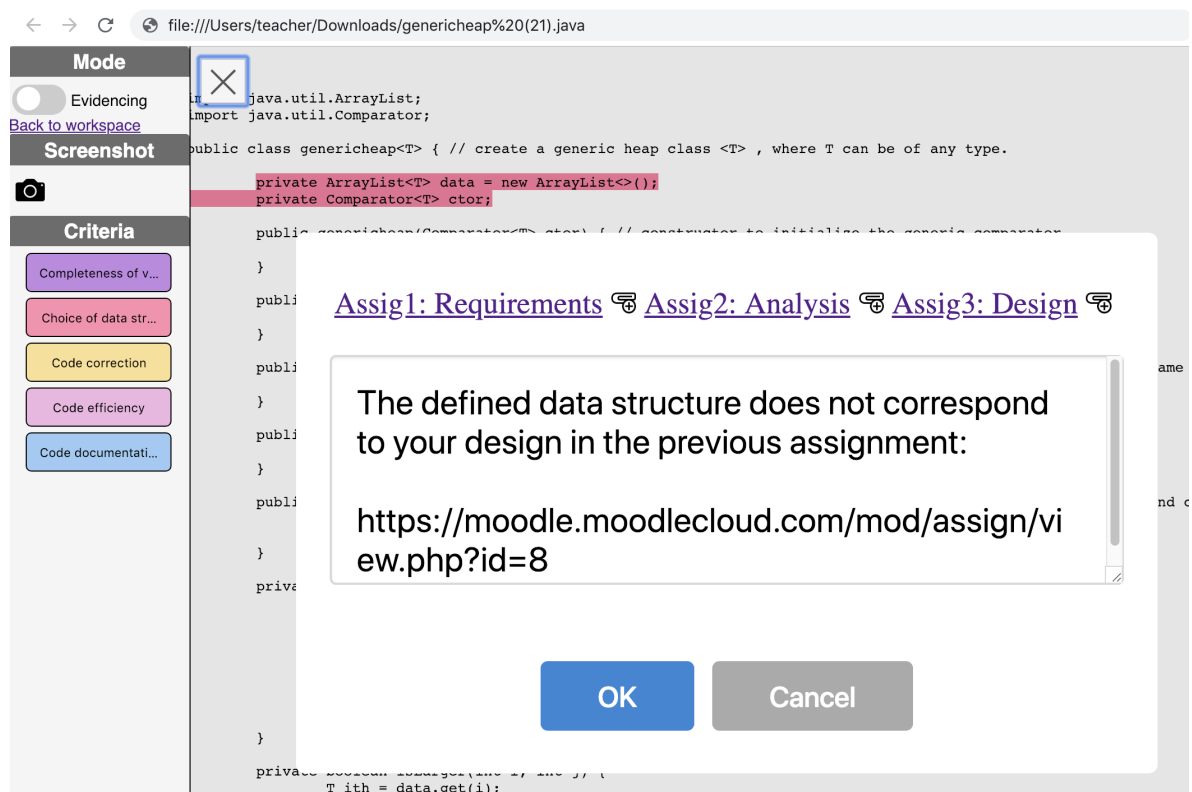
Aplikazio hau ***Moodletik* entregatzen diren azterketak edo ariketak zuzentzeko balio du** (Medina, 2019b). Horretarako, aurretik *Moodlen* zehaztutako ebaluazio irizpideekin dokumentuak azpimarratzen dira, eta irizpide bakoitzari nota bat ezartzen zaio.

Esan bezala, aplikazio hau erabiltzeko, *Moodlen* ebaluazio irizpideak zehaztu behar dira, hau da, tresna hau *Moodle* bitartez **konfiguratzen** da. Behin errubrika (ebaluazio irizpideak eta bakoitzaren puntuazioa) definitu dela, Chrome-ko hedapena martxan jarri behar da konfigurazioa amaitzeko.

*Mark&Go*ren **interfaze grafikoa** 5.3 irudian ikusi daiteke. Irudian ikusten den alboko barra *Evidencing* moduari dagokio, baina, aplikazio honetan beste bi modu aurkitu daitezke: aipatutako *Evidencing* modua eta *Marking* modua, irakaslearen rolari lotuta daudenak, eta ikaslearen modua.

*Highlight&Go*kin gertatzen den moduan, aztertzen den dokumentua erdialdean gelditzen da, eta ezker aldean *sidebara* dago aukera desberdinekin. Lehendabizi, motaz aldatzeko *toggle* botoi bat dago, baita *Moodler*a bueltatzeko esteka bat ere. Ondoren, dokumentuari pantaila-irudiak ateratzeko aukera dago. Azkenik, moduaren arabera *tag* lista bat edo beste bat agertuko da. *Evidencing* moduan ebaluazio irizpideak agertuko dira, *Marking* moduan irizpide horiei esleitu beharreko notak, eta azkenik, ikasle batek irakasleak egindako anotazioetatik nabigatzeko aukera izango du.

Anotazio batean iruzkin bat jartzeko, honen gainean eskuineko botoia sakatu behar da eta aukera hori aukeratu. 5.3 irudian ikusi daiteke iruzkin baten adibide bat.



5.3 Irudia: *Mark&Goko Evidencing mode* modua. Irudia [Medina \(2019b\)](#) artikulutik lortu da.

Beraz, hauek izango lirateke aplikazio honen **funtzionalitateak**. Kasu honetan bi rol ditugu; ikasle eta irakaslea. Gainera, azkeneko horrek, bi modu desberdinekin lan egin dezake:

- **Irakaslea:**

- *Evidencing mode:* (ikusi [5.3](#) irudia)

- * *Moodlera* bueltatzeko esteka.
 - * *Checking (Marking mode)* modura pasatzeko aukera.
 - * PDF bat sortu anotazioekin (*screenshots*).
 - * Anotazioak sailkatu ebaluazio irizpideen arabera.
 - * Anotazioak ezabatu.
 - * Anotazioetan iruzkinak jarri.
 - * Ikasleen iruzkinak ikusi.
 - * Ikasleen iruzkinak erantzun.

- *Marking mode:*

- * *Moodlera* bueltatzeko esteka.
 - * *Marking (Evidencing mode)* modura pasatzeko aukera.
 - * PDF bat sortu anotazioekin (*screenshots*).
 - * Irizpide bakoitzarekin lotutako anotazioen arabera kalifikazio bat jarri irizpide edo bloke bakoitzari.
 - * Anotazioak ezabatu.
 - * Anotazioetan iruzkinak jarri.

- **Ikaslea**

- Irakasleak egindako anotazioak ikusi.
 - Irakaslearen anotazioetatik nabigatu.
 - Irakaslearen iruzkinak ikusi.
 - Irakaslearen iruzkinetara erantzun.

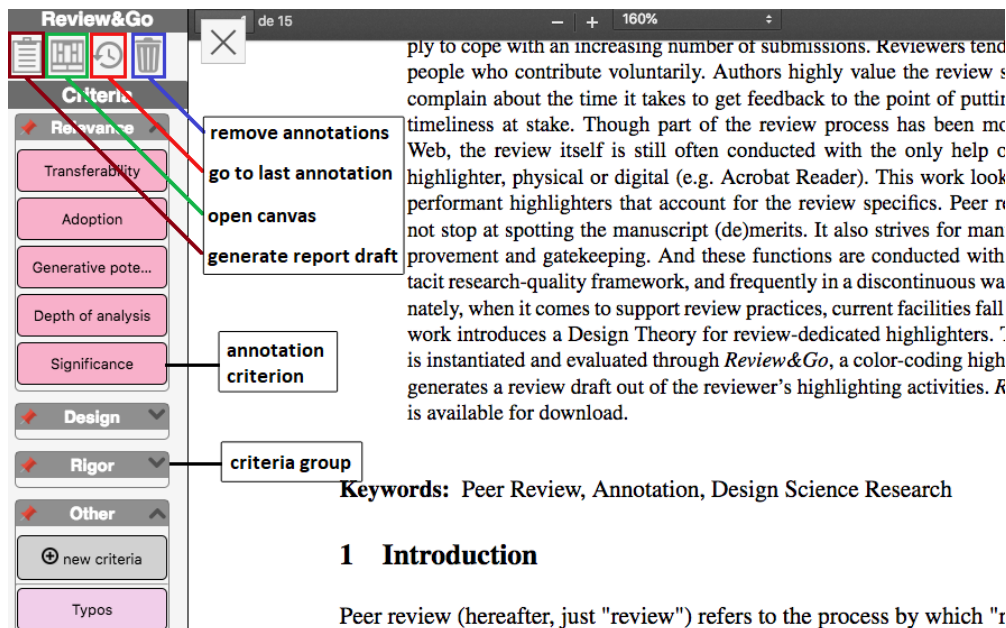
5.3 Review&Go

Gaur egun ikerketa artikuluen berrikuspen batzuk ordenagailuz egiten dira, baina, horretarako tresna gehienak artikulua kolore batekin markatzeko aukera ematen dute soilik.

Review&Go aplikazioa **berrikuspen horiek errazteko sortu du** Haritz Medinak. Hone-tan irizpideak kolore desberdinekin agertzen dira eta kolore horiekin artikulua markatu daitezke ebaluatzen duen pertsonaren lana errazteko, baita ebaluazioa irakurri behar duen autoreak ebaluazio hori errazago ulertzeko (ikusi 5.4 irudia).

Anotazio horiek hiru motetan sailkatu daitezke:

- **Strengths:** Artikuluak dituen atal indartsuenak; beste hitzetan esanda, artikulua dituen puntu positiboak dira.
- **Minor weakness:** Hobetu daitezkeen atalak dira.
- **Major weakness:** Artikuluko puntu ahulenak dira, zuzendu behar diren atalak, alegia (ikusi 5.4 irudia).



5.4 Irudia: *Review&Go* aplikazioa. Irudia Medina u. a. (2019b) artikulutik lortu da.

Tresna honen irizpideak *DefaultCriteria.js* scriptean **konfiguratz**en dira, hau da, ez du beste tresnarik behar konfigurazioa egiteko, beste tresnekin gertatzen ez den bezala. *Script* horretan ondoren erabiliko diren irizpide guztiak zerrendatuta agertzen dira.

5.4 irudian *Review&Go* aplikazioaren **interfaze grafikoa** ikusi daiteke. Beste aplikazioen interfaze grafikoaren antza berdina dauka, hau da, aztertzen den dokumentua erdialdean gelditzen da eta ezkerrean anotazioak egiteko aukerak aurkezten dituen menua dago.

Alboko menuko goialdean hainbat aukera daude: dokumentuan egindako anotazioen txosten bat lortzeko bi formatu desberdinetan (testu fitxategia eta PDF), egindako anotazioen laburpena HTMLko taula batean ikusteko, egindako azkeneko anotaziora nabigatzeko aukera eta anotazio guztiak ezabatzeko aukera. Aukera hauen ondoren ebaluazio irizpideak daude, eta azkenik, irizpide berriak sortzeko aukera.

Tresna honen **funtzionalitateak** honakoak dira:

- Irizpide berriak sortu.
- Anotazioekin txosten bat (*report*) sortzeko aukera. Bi modutako txostena sortu daiteke, testu bidezko txostena edo PDFko pantaila-argazkiak ateratzeko tresnaren bidez.
- Egindako anotazioen laburpen bat ikusi (*Canvas*). Aukera honek HTML orri bat sortzen du egindako anotazioen eta hauen sailkapenen aurkibide moduko laburpen bat erakusten duena.
- Egindako azkeneko anotaziora mugitzeko aukera.
- Anotazio guztiak ezabatzeko aukera.
- Anotazioak banaka ezabatu.
- Anotazioak irizpideen arabera egin.
- Anotazioetan iruzkinak jarri.
- Anotazioak sailkatu (*Strengths, Minor weakness* eta *Major weakness*).
- Anotazioetan gomendioak ezarri.

6. KAPITULUA

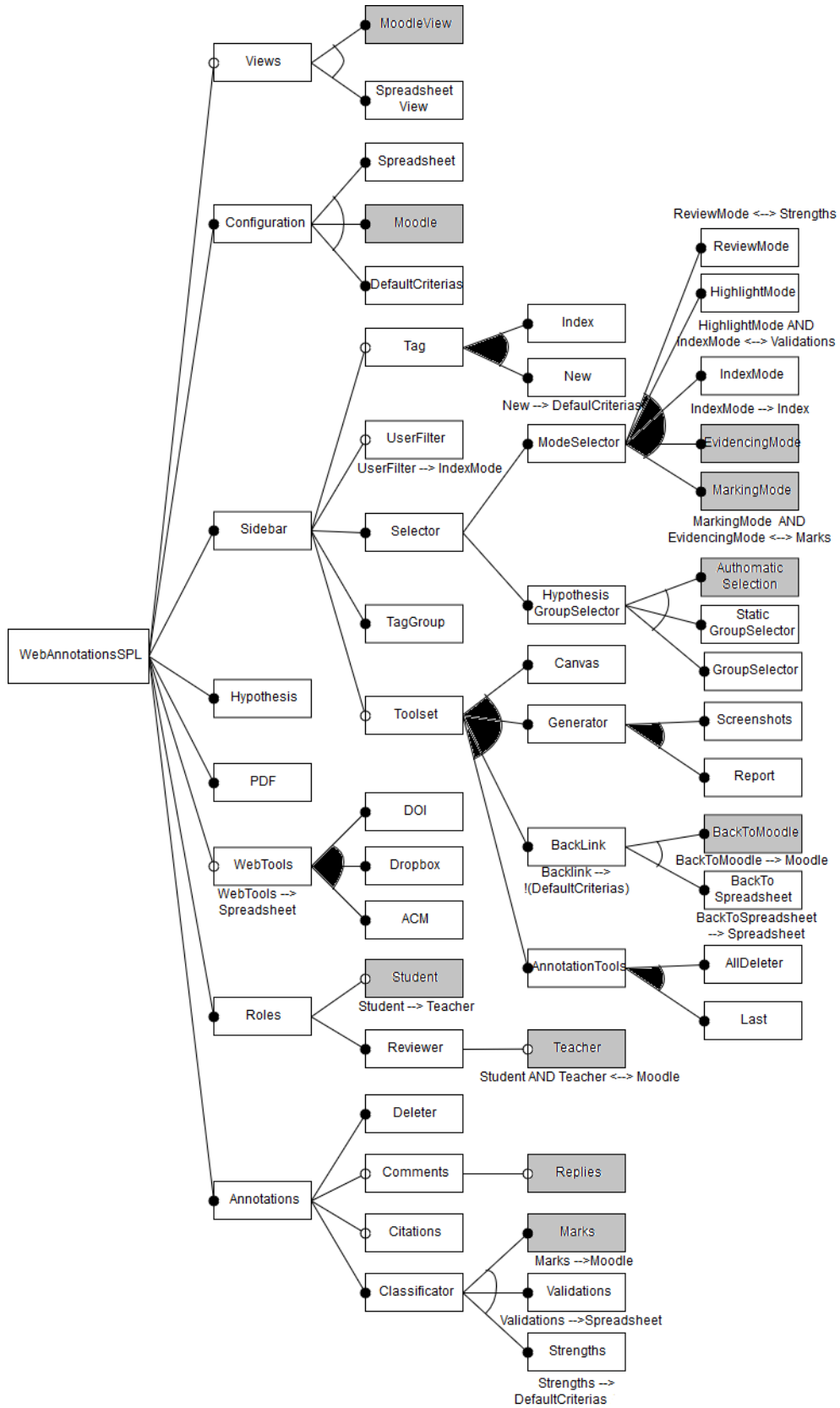
*WebAnnotationSPL*aren ezaugarri-diagrama

5. kapituluaren deskribatzen diren aplikazioen kodearen azterketa sakona eginez 6.1 irudian erakusten den ezaugarri-diagrama osatu da (C eranskinean aurkeztutako notazioa erabiliz idatzia). Diagramako ezaugarrien deskribapena jarraian ematen da. Ezaugarrien deskribapen horiek, diagraman agertzen diren orden berdinean, goitik behera, emango dira.

Ezaugarri-diagrama modu grafikoan 6.1 irudian ikusi daiteke. Grisez markatutako ezaugarriak *Mark&Go* aplikazioarekin bakarrik erlazionatutako ezaugarriak dira, eta birplangintzan (3. kapituluaren) esan bezala, *Mark&Go*ko ezaugarriak garapenaren hurrengo fasetan alde batera uztea erabaki zen.

Ezaugarrien azalpenean **beltzez** adierazitako ezaugarriak derrigorrezkoak izango dira, **beltzez eta letra etzanez** adierazitakoak hautazko ezaugarriak, *letra etzanez* adierazitakoak or-inklusiboko ezaugarriak eta azkenik, azpimarratuta adierazitakoak or-esklusibokoak:

- **Views**: aztertzen den dokumentuaren gainean egindako anotazioen emaitzak *Moodle* edo *Google Spreadsheet*en bistaratzea aukeratu daiteke. SPLtik lortzen diren aplikazioak (hurrengo puntuan ikusiko den moduan) *Moodle* bitartez, *Google Spreadsheet* bitartez edo lokalki konfiguratu daitezke. Konfigurazio hori ez du eragingo bistaratzeko moduari, hau da, nahiz eta produktuaren irizpideak *Moodle* bitartez konfiguratuta egon, emaitzak Googleko kalkulu horri batean bistaratzea aukeratu daiteke, adibidez. Bi aukeretako bat aukeratu behar da (or-esklusiboa), ezingo dira emaitzak bi leku desberdinetan bistaratu. Hauek dira aipatutako aukerak, hurrenez hurren: MoodleView eta SpreadsheetView.



6.1 Irudia: Ezaugarri-diagrama.

- **Configuration:** hiru aplikazioak era desberdinean konfiguratzeko direnez, amaierako produktuen konfigurazioa hiru modu desberdinetan egin daiteke: Google *Spreadsheet* bitartez (*Spreadsheet* ezaugarrian bilduta), *Moodle* bitartez (*Moodle* ezaugarrian) edo lokalki, hau da, *DefaultCriteria.js* scriptan irizpideak konfiguratzeko (*DefaultCriteria* ezaugarrian). Konfigurazio bakarra aukeratu daiteke produktuko, eta konfigurazioa derrigorrezkoa da.
- **Sidebar:** aplikazio guztiek *sidebar* bat erakusten dute, hau da, ikertzen edo ebaluatzen den dokumentua areagotzen dute HTMLez garatutako alboko menu moduko batekin. Menu horretan konfiguratzeko ikerketa irizpideak edo ebaluazio irizpideak agertzen dira, horiekin batera beste aukera batzuk agertu daitezke: anotazio guztiak ezabatzeko aukera, anotazioekin txosten bat sortzeko aukera, anotazioetatik nabigatzeko aukera, besteak beste. Aipatzeko aukera horiek banan-banan ikusiko dira hemen:
 - **Tag:** irizpideekin lotutako ezaugarria da hau. Irizpide berriak sortzeko, ezabatzeko edota editatzeko funtzionalitateak *New* ezaugarrian biltzen dira. *Index* ezaugarria, bestalde, irizpide baten gainean klikatuz irizpide horiekin anotatzeko lehenengo anotaziora nabigatzeko funtzionalitatearekin lotuta dago, dokumentuko testurik aukeratu ez badago, noski. Bi ezaugarri hauek orinklusibo batean daude, hau da, hauetako bat edo biak batera aukeratu egon daitezke, funtzionalitate horiek batera agertu daitezkeelako, edo ez agertzea ere izan daiteke, eta horregatik da **Tag** ezaugarria aukerakoa.
Aipatu behar da *New* ezaugarria aplikazioa *DefaultCriteria* bidez konfiguratu badago aukeratu daitekeela soilik, zeren eta beste *Spreadsheet* eta *Moodle* kasuan, Googleko kalkulu orriko taulan eta Moodleko errubrikan editatu behar dira irizpideak, hurrenez hurren. Beraz, kasu horietan, ezin dira irizpideak *sidebare*tik editatu.
 - **UserFilter:** Aukerako ezaugarria da. Hau aukeratzeko gero, amaierako produktuan anotazioak erabiltzaileen arabera (anotazio horiek egin dituzten erabiltzaileen arabera) sailkatzeko aukera edukiko du. Funtzionalitate horiekin, adibidez, *User*en anotazioak ikusi nahi badira soilik, hau aukeratzeko da *IndexMode* eskaintzen duen erabiltzaileen zerrendatik beste erabiltzaile guztiek egindako anotazioak baztertuz. Funtzionalitate hau *IndexMode* ezaugarriari lotuta dago.
 - **Selector:** Ezaugarri honekin aplikazioen erabat mendekoak diren funtzionali-

tateak biltzen dira, ia funtzionalitate bakoitza aplikazio bati lotuta dago. Hauek dira aukeraketa horiek:

- * **ModeSelector:** aplikazioek hainbat modu desberdin eskaintzen dituzte. Azkeneko produktuan modu bat baino gehiago egonez gero, hauetako batetik bestera aldatzeko aukera ematen du ezaugarri honek. Hauek dira aipatutako moduak:
 - *ReviewMode:* modu honetan ebaluazio irizpideak taldeka multzokatzten dira, hau da, irizpide bakoitza talde bat ordezkatzten du. Honen zergatia, irizpide bakoitzari maila bat (*Strength*, *Minor weakness* eta *Major weakness*) esleitu behar zaiolako da. Irizpide hauek ikerketa artikularen gainean anotazioak egiteko balio dute. Modu hau *Strenths* motako sailkapenari lotuta dago.
 - *HighlightMode:* modu honetan, aldiz, irizpide bakoitza ikerketa irizpide bakarra da, hau da, ez dira taldeka multzokatzten, baizik eta indibidualki. Irizpide hauek ikertzen den dokumentuaren gainean anotazioak ezartzeko erabiltzen dira. Modu hau *Validations* sailkapen motari lotuta dago.
 - *IndexMode:* modu hau *Validations* ezaugarria aukeratu denean agertu daiteke. Honetan beste erabiltzaileek egindako anotazioak ikusi eta erabiltzaile zehatzen anotazioak ikusi daitezke, baita ikerketa edo ebaluazio irizpideen arabera nabigatu anotaziotan zehar. Modu honen funtzionalitaterik garrantzitsuena anotazioen balidazioa da. Modu honek *Index* ezaugarriaren beharra dauka.
 - *MarkingMode:* *Marks* ezaugarriarekin lotuta dago ezaugarri hau. Modu honek hainbat aukera eskaintzen ditu, baina nagusiena edo garrantzitsuena, ebaluazio irizpideen arabera azpimarratutako anotazioei nota jartzearena da.
 - *EvidencingMode:* ezaugarri hau ere *Marks* ezaugarriarekin lotuta dago. Modu honen aukera nagusiena dokumentua ebaluazio irizpideen arabera azpimarratzekoarena da.
- * **HypothesisGroupSelector:** aplikazioen konfigurazioa eta hauekin egindako anotazioen informazioa *Hypothes.is*eko talde batean gordetzen dira (**Hypothesis** ezaugarrian berriro aipatuko da hau). Talde horren eta aplikazioaren arteko lotura hiru modu desberdinetan konfiguratu daiteke. Ezaugarri honek hiru modu desberdin horiek multzokatzten ditu or-

esklusibo batekin; hots, hiru konfigurazioetatik soilik bat har daiteke (rolena *Roles* ezaugarrian azalduko da):

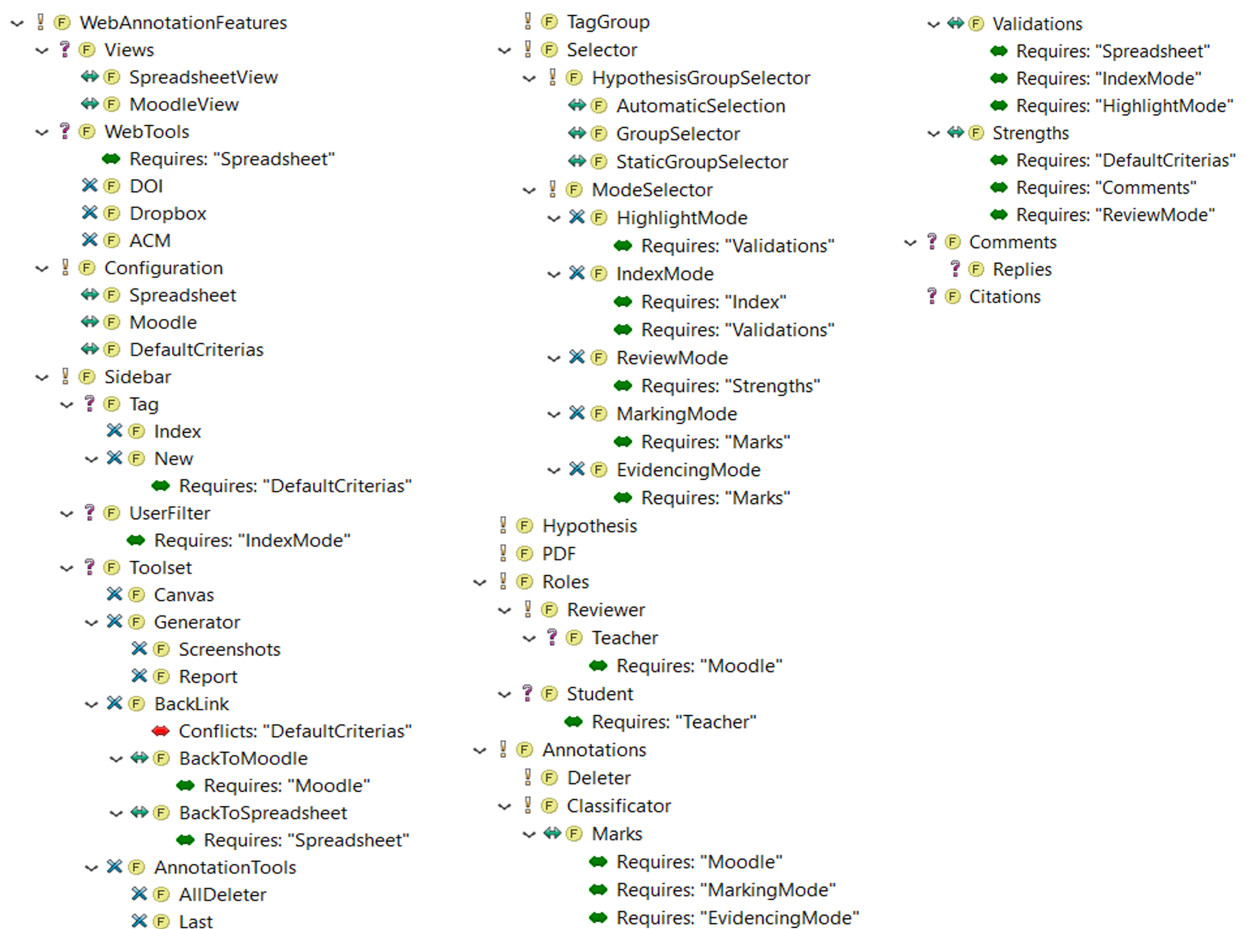
- AuthomaticSelection: konfigurazio hau aukeratzen bada, produktuak automatikoki aukeratuko du *Hypothes.is*eko taldea erabiltzailearen rolaren arabera, adibidez, erabiltzailea irakaslea bada (*Teacher*) aplikazioak talde bat esleituko dio, ikaslearen (*Student*) kasuan beste bat.
 - StaticGroupSelector: konfigurazio honekin taldea hasieran esleitzen da betirako, hau da, nahiz eta rola gero aldatu taldea beti berdina izango da, hasierakoa. Taldearen informazioa *Config.js scriptean* gordezen da.
 - GroupSelector: konfigurazio honek erabiltzaileari taldea edozein momentutan aldatzeko aukera ematen dio, nahiz eta rolez ez aldatu.
- **TagGroup**: *sidebarean* (alboko menuan) ebaluazio edo ikerketa irizpideak zerrendatuta agertzen dira kasu guztietan. Funtzionalitate hori derrigorrezko ezaugarri honetan biltzen da.
- **Toolset**: ezaugarri honek hurrengo ezaugarriak multzokatzen ditu or-inklusibo batekin. Ezaugarri hauek multzokatzea erabaki da, bai interfaze grafikoan, baita kodeketan multzokatuta agertzen direlako (*toolsetBar.js scriptan* multzokatzen dira):
- * *Canvas*: ezaugarri honek azkeneko produktuan botoi bat ezartzen du, non hau sakatuz gero, egindako anotazioen laburpen bat HTMLez garatutako taula batean erakusten duen.
 - * *Generator*: bi motako sorgailu (*Generator*) daude, alde batetik dokumentuaren pantaila-argazkiak ateratzen duen sorgailua (*Screenshots* ezaugarria), eta beste aldetik, testu txosten bat sortzen duen sorgailua (*Report* ezaugarria). Bi ezaugarri horiek batera agertu daitezke amaierako produktuan, edo horietako bat soilik (or-inklusiboa).
 - * *BackLink*: *BackLink* ezaugarria aukeratzea ikertzen edo zuzentzen ari den dokumentutik konfigurazio tresnara bueltatzeko link bat produktuan ezartzea da. Hortaz, *Moodlera* bueltatzeko esteka bat edo Google *Spreadsheetera* bueltatzeko esteka bat txertatuko da (bat edo bestea izan behar da, ezin delako bukaerako aplikazioa aldi berean bi tresna desberdinekin konfiguratu):
- BackToMoodle: ezaugarri hau aktibatzeko *Moodleko* konfigurazioa (*Moodle* ezaugarria) beharrezkoa da.

- BackToSpreadsheet: aurrekoaren berdina, baina Google *Spreadsheet*ekin (*Spreadsheet* ezaugarriarekin).
- * *AnnotationTools*: ezaugarri honek anotazioekin zerikusia du, bere barnean bi ezaugarri aurkitzen dira or-inklusibo batekin: *AllDeleter* eta *Last*. Ezaugarri hau ez da azkeneko produktu guztietan agertuko, aukerakoa baita.
 - *AllDeleter*: ezaugarri hau aukeratuz gero, amaierako produktuak dokumentu bateko anotazio guztiak ezabatzeko aukera eskaini ahalko du.
 - *Last*: ezaugarri honen bidez amaierako produktuan, egindako azkeneko anotaziora nabigatzeko aukera eskaini ahalko da.
- **Hypothesis**: hiru aplikazioek anotazioak, iruzkinak eta anotazioekin zerikusia duten beste ezaugarri batzuk (*citations*, *strenghts...*) *Hypothes.is*en gordetzen dituzte, beraz, ezaugarri hau beharrezkoa izango da produktu guztietan. Bertan *Hypothes.is*eko konexioak eta eragiketak kodetzen dira.
- **PDF**: ezaugarri hau derrigorrezkoa da. Honi esker aplikazioek PDF motako dokumentuak ikertu edo ebaluatu ditzakete. Ezaugarri honetan PDF dokumentuak irakurri eta anotatu ahal izateko behar diren eragiketak kodetzen dira.
- **WebTools**: Ezaugarri honetan zenbait web tresna biltzen dira (*DOI*, *Dropbox* eta *ACM*). Tresna edo zerbitzu hauekin lan egiteko *Spreadsheet* ezaugarria beharrezkoa da, tresna hauek ikerketa taldeen domeinuarekin erlazionatzen direlako.
- **Roles**: aplikazioetan hiru rol desberdin identifikatu daitezke, derrigorrezkoa den rol bakarra **Reviewer** rola da. Beste biak (*Student* eta *Teacher*), hautazkoak dira, soilik azterketak edota lanak ebaluatzeko prestatuta dauden produktuetan agertzen baitira (*Marks* ezaugarria aktibatuta duten produktuetan). *Teacher* ezaugarria *Reviewer*etik zintzilikatu da, *Teacher Reviewer*en hedapen bat delako.
- **Annotations**: azpimarratutako paragrafoen gainean egin daitezkeen funtzionalitateak multzokatzen ditu:
 - **Deleter**: ezaugarri hau derrigorrezkoa da, egindako azpimarraketak ezabatze-ko aukera ematen baitu.
 - **Comments**: hau aukerazko ezaugarri bat da, aktibatuta egonez gero, azpimarratutako paragrafoetan iruzkinak jartzeko aukera ematen du.

-
- * **Replies:** hautazkoa da honako ezaugarri hau ere, iruzkinak erantzuteko aukera aktibatzeke balio du, horregatik *Comments*en mendekoa da.
 - **Citations:** egindako anotazioetan gomendioak edo zita bibliografikoak ezartzeko balio du.
 - **Classifier:** aplikazio hauen funtzionalitate nagusienak multzokatzen ditu or-esklusibo batekin:
 - * **Marks:** *Marks* ezaugarria aukeratzen bada, azkeneko produktua azterketak edo ariketak zuzentzeko balioko du, hau da, dokumentuak ebaluazio irizpideen arabera azpimarratuko da eta azpimarraketa hauen arabera nota bat jarriko da amaieran atal bakoitzari; ebaluazio irizpide bakoitzari, alegia.
 - * **Validations:** ezaugarri hau aktibatzen bada, amaierako produktua dokumentuak ikerketa irizpideen arabera azpimarratutako parrafoak balidatzeko balioko du, ikerketa taldeetan baliozkoa den informazioa lortzeko balio du.
 - * **Strengths:** ezaugarri hau hautatzen bada, azkeneko produktua artikulua ebaluatzeko balioko du. Dokumentuak ebaluazio irizpideen arabera markatuko dira eta marka horiek hiru mailetan sailkatuko dira: *Strength* (artikulua duen puntu indartsuak), *Minor weakness* (artikuluko puntu ahul txikiak) eta *Major weakness* (artikuluko puntu ahulak).

6.0.1 Ezaugarri-diagrama *pure::variants*-en

4. kapituluan esan den moduan, *pure::variants*en ezaugarri-diagramek *Feature Models* izena hartzen dute. Deskribaturiko ezaugarri-diagramari dagokion *Feature Model*a [6.2](#) irudian erakusten da.



6.2 Irudia: Ezaugarri-diagrama *pure::variantsen*.

7. KAPITULUA

*WebAnnotationSPL*aren arkitektura

Behin ezaugarri-diagrama definituta, Software Produktu-Lerroaren arkitektura definitu behar da. Bi pauso nagusi identifikatzen dira (ikus 4. kapitulua): iturburuko fitxategiak sortzea (hemen etiketatze prozesua edo *tagging* sartzen da) eta fitxategi horiek ezaugarriekin erlazionatzea *Family Model*aren bitartez. Hurrengo bi ataletan *WebAnnotationSPL*ren inplementazioa azalduko da.

7.1 Iturburuko fitxategiak

Atal honen helburua ez da jatorrizko hiru aplikazioen kodea azaltzea, baizik eta iturburuko fitxategiak gaineratik aipatzea eta egindako etiketak ikustea.

Iturburuko fitxategiak *pure::variant*seko proiektuaren *inputs* karpetan aurkitzen dira. Bertan produktu guztiak martxan jartzeko behar duten fitxategiak aurkitzen dira: *HTML* orriak, *JavaScript* fitxategiak, irudiak, *JSON* fitxategiak aplikazioen konfiguraziorako eta testuarentzako, eta *SCSS*¹ scriptak estiloarentzako.

input karpetaaren barnean hainbat karpeta desberdin aurkitzen dira:

- **app**: fitxategi gehienak karpeta honetan aurkitzen dira. Aplikazioen inplementazioa (kodea) hemen aurkitzen da:

¹Haritz Medinaren aplikazioek Sass (*Syntactically Awesome Stylesheets*) erabiltzen dute estiloa ezartzeko. *SCSS Sasseko* estilo *scriptak* dira.

- **_locales:** produktuetan interfazeaz agertuko den testu guztia karpeta honetako *messages.json* fitxategian aurkitzen da.
 - **content:** karpeta honetan PDFekin lan egiteko behar diren fitxategi eta *script* guztiak multzokatzen dira.
 - **images:** produktuek erabiliko dituzten irudiak gordetzen dira karpeta honetan.
 - **pages:** karpeta honetan HTML fitxategiak aurkitzen dira. Barnean beste bi karpeta ditugu: *sidebar* eta *specific*.
 - **scripts:** hemen barruan JavaScript fitxategi guztiak aurkitzen dira. Lan gehiena hemen egin da. Bi fitxategi pisutsuenak *TextAnnotator.js* (1.117 lerro) eta *TagManager.js* (1.675 lerro) dira. Lehenengoa anotazioetarako erabiltzen da, eta bigarrena, *sidebarean* (alboko barran) irizpideak ezartzeko.
 - **styles:** karpeta honetan SCSSko scriptak aurkitzen dira, HTML orrietan estiloa ezartzeko.
 - **manifest.json:** JSON fitxategi honetan lortzen diren aplikazioen konfigurazioa ezartzen da: aplikazioaren izena, bertsioa eta erabili behar dituen web orrietara atzitzeko baimenak.
- **tasks** eta **gulpfile.babel.js:** *Gulp* atazak automatizatzeko erabiltzen den Node.js²ko script bat da. Ataza horiek *tasks* karpetan aurkitzen dira: build, chromereload, clean...
 - **package.json:** JSON fitxategi honek aplikazioek behar dituzten mendekotasun guztiak (*Node.js* paketeak) gordetzen ditu.
 - **wdio.conf.js** eta **test:** WebdriverIO (*wdio.conf.js* scripta) probak (*test*) egiteko *Node.js*ko tresna bat da (SPLarekin lortutako produktuen probak ez dira tresna honekin egin).

Iturburuko fitxategien aldakortasuna identifikatzeko 4. kapituluaren aipatu diren *DiffMerge* eta *BUT4Reuse* teknologiak erabili dira. Lehena fitxategi bakoitzaren barruko desberdintasunak identifikatzeko eta bigarrena aplikazioen egitura alderatzeko; modu honetan identifikatu dira aplikazio batek dituen eta beste batek ez dituen fitxategiak, adibidez.

*pure::variant*seko etiketak kode barruko edukia aukeratu diren ezaugarrien arabera sortzeko edo ez sortzeko erabiltzen dira, hau da, ezaugarri bat aukeratzekoan ezaugarri horri lotuta dauden funtzioak sortzeko, edo ezaugarri hori ez bada aukeratzen funtzio horiek ez sortzeko balio dute *pure::variant*seko etiketak (ikusi 4. kapituluko *Tagging* atala).

²Node.js interneteko aplikazio eskalagarriak sortzeko programazio ingurune bat da ([Wikipedia, 2019a](#))

Nahiz eta *Mark&Goko* ezaugarriekin erlazioatutako etiketa batzuk aipatu, hauek ez dira kontuan hartuko jarraian aipatuko diren datuetan, hauekin egindako lana ez delako behar bezala egiaztatu. 7.1 taulan ikusi daitezke egindako etiketa guztiak. Taulako lehenengo zutabearen ezaugarri-diagraman (ikusi 6.1 irudia) erakutsi diren ezaugarriak zerrendatu dira; ondorengo zutabearen ezaugarri horiekin erlazioatu diren baldintzak edo etiketak daude; eta azkenik, baldintza horiek zein fitxategitan dauden eta zein lerrotan agertzen diren adierazi da. Beraz, *Highlight&Go* eta *Review&Gokin* erlazioatutako etiketak soilik hartuta, 25 fitxategi etiketatu dira eta guztira 261 etiketa jarri dira.

Ezaugarriak	Etiketak	Agerpena fitxategietan (lerroa)
Spreadsheet	Spreadsheet	<ul style="list-style-type: none"> - TagManager.js (92, 170, 190, 217, 343, 995, 1091). - GroupSelector.js (21). - ContentScriptManager.js (75). - AnnotationBasedInitializer.js (48, 49). - TextAnnotator.js (104, 282, 515, 546, 576, 644, 660, 672, 1204). - ToolsetBar.js (214, 255, 390) - specificContentScript.js (8, 30, 56, 160) - background.js (15, 38) - manifest.json (14, 52, 93, 101)
	NOT(Spreadsheet)	<ul style="list-style-type: none"> - TagManager.js (683, 1023) - TextAnnotator (638, 732)

DefaultCriterias	DefaultCriterias	<ul style="list-style-type: none"> - TagManager.js (17, 145, 165, 199, 234, 275, 288, 430, 442, 703, 743, 748, 972, 1045, 1061, 1065, 1074, 1075, 1078, 1086) - Tag.js (13) - GroupSelector.js (15, 19) - ContentScriptManager.js (68) - TextAnnotator.js (325, 346, 361, 374, 452, 505, 691, 700, 705, 712, 1139, 1149, 1369, 1382, 1386, 1581, 1623) - Level.js (28) - Criteria.js (8, 9, 15, 48) - ToolsetBar.js (20, 212, 241, 277, 307, 334) - Config.js (56) - contentScript.scss (23)
	NOT(DefaultCriterias)	<ul style="list-style-type: none"> - TagManager.js (128, 322) - TextAnnotator.js (1432, 1659, 1664) - specificContentScript (5)
Index	Index	<ul style="list-style-type: none"> - TagManager.js (31, 834, 1391, 1414)
New	New	<ul style="list-style-type: none"> - TagManager.js (777, 805) - TextAnnotator.js (81, 88, 135) - specificContentScirpt.js (25, 50)
UserFilter	UserFilter	<ul style="list-style-type: none"> - TagManager.js (834) - Events.js (14) - ContentScriptManager.js (17, 133, 156, 164, 276, 299) - TextAnnotator.js (73, 96, 172)
ReviewMode	ReviewMode	<ul style="list-style-type: none"> - tagWrapper.html (35) - TagManager.js (40, 182, 560, 946, 964) - ModeManager.js (33, 101, 171)
	NOT(ReviewMode)	<ul style="list-style-type: none"> - TagManager.js (42) - ModeManager.js (162)
	ReviewMode AND IndexMode	<ul style="list-style-type: none"> - ModeManager.js (83)

HighlightMode	HighlightMode	<ul style="list-style-type: none"> - messages.json (71) - tagWrapper.html (33) - TagManager.js (934) - Tag.js (72) - ModeManager.js (30, 99, 116, 167, 191)
	HighlightMode AND IndexMode	<ul style="list-style-type: none"> - TagManager.js (48) - ModeManager.js (70) - TextAnnotator.js (609)
IndexMode	IndexMode	<ul style="list-style-type: none"> - messages.json (77) - tagWrapper.html (34) - TagManager.js (940) - Tag.js (73) - ModeManager.js (21, 126, 136, 175, 200) - TextAnnotator.js (720, 766)
	IndexMode AND (HighlightMode OR ReviewMode)	<ul style="list-style-type: none"> - TagGroup.js (11) - Tag.js (28) - ModeManager.js (53, 97, 162) - ContentScriptManager.js (4, 63, 97) - TextAnnotator.js (15, 70, 99, 209, 220)
StaticGroupSelector	StaticGroupSelector	<ul style="list-style-type: none"> - GroupSelector.js (18) - ContentScriptManager.js (14, 67, 92) - Config.js (15)
GroupSelector	GroupSelector	<ul style="list-style-type: none"> - groupSelection.html (2) - TagManager.js (78) - GroupSelector.js (3, 59, 72, 86, 177, 341) - ContentScriptManager.js (29, 84, 111, 125, 175, 181, 311) - manifest.json (39)
Toolset	Toolset	<ul style="list-style-type: none"> - ContentScriptManager.js (26) - specificContentScript.js (22, 64) - contentScript.scss (13)

	Toolset AND DefaultCriteria	- specificContentScript.js (44)
Canvas	Canvas	- ToolsetBar.js (68, 208, 440) - contentScript.scss (16)
Generator	Generator	- messages.json (134)
Screenshots	Screenshots	- TolsetBar.js (2, 45, 139)
Report	Report	- ToolsetBar.js (33, 145, 440)
BackLink	BackLink	- ToolsetBar.js (5, 92)
BackToSpreadsheet	BackToSpreadsheet	- ToolsetBar.js (113)
AllDeleter	AllDeleter	- messages.json (139) - TagManager.js (63) - Events.js (25) - TextAnnotator.js (77, 92, 117, 1637) - ContentAnnotator.js (9) - ToolsetBar.js (56, 186) - DeleteAnnotationManager.js (22, 50)
Last	Last	- TextAnnotator.js (261) - ToolsetBar.js (80, 434)
DOI	DOI	- background.js (18, 43) - manifest.json (87)
Dropbox	Dropbox	- DoiManager.js (61) - manifest.json (87)
ACM	ACM	- manifest.json (32)
Comments	Comments	- messages.json (106) - Events.js (19) - TextAnnotator.js (836, 847, 875, 1117, 1200)
Citations	Citations	- TextAnnotator.js (1163, 1209, 1236) - contentScript.scss (26)
Validations	Validations	- messages.json (3, 13, 22, 39, 56, 95) - tagWrapper.html (26) - Events.js (5) - TextAnnotator.js (859, 866) - ToolsetBar.js (158) - specificContentScript.js (14)
	NOT(Validations)	- tagWrapper.html (29)

Strengths	Strengths	<ul style="list-style-type: none"> - messages.json (5, 15, 27, 41) - TextAnnotator.js (1117, 1172, 1214) - ToolsetBar.js (14, 147) - textAnnotator.scss (10)
------------------	-----------	--

7.1 **Taula:** Egindako etiketen taula.

7.2 *pure::variantseko Family Modelak*

Behin iturburuko fitxategiak etiketatuta, fitxategi horiek ezaugarri-diagramako ezaugarriekin lotu behar dira, *pure::variantsen Family Modelak* eta murriztapenak erabiliz. Aukeratu diren ezaugarriek behar dituzten fitxategiak sortzeko *Family Modelak* murriztapenak ezartzen dira, horrela fitxategi talde bat soilik ezaugarri bat aukeratzeko denean sortzeko. *Family Model* batean bi ezaugarri nagusi identifikatzen dira: zuhaitza (produktuen fitxategi eta karpeten egitura definitzen duena) eta murriztapenak (zein karpeta eta fitxategi sortu behar diren produktu bakoitzarentzako ezartzen dutenak).

Murriztapen (*restrictions*) guztiak *images*, *pages* eta batez ere *scripts* osagaietan (*components*) aurkitzen dira. 37 murriztapen jarri dira guztira, eta horiek 7.2 taulan bildu dira. Taula horretako lehenengo zutabean ezaugarrien zerrenda dago; bigarrenean, ezaugarri horiekin erabili diren murriztapenak; eta azkeneko zutabean, murriztapenen baldintza horiek betetzen badira sortuko diren fitxategien zerrenda.

Ezaugarriak	Murriztapenak	Inplikaturako fitxategiak
SpreadsheetView	SpreadsheetView	<ul style="list-style-type: none"> - CommonHypersheetManager.js - HypersheetColors.js - PrimaryStudySheetManager.js
Spreadsheet	Spreadsheet	<ul style="list-style-type: none"> - GoogleSheetsManager.js - googleSheets - googleSheetsContentScript - googleSheetsContentScript.js
	NOT(Spreadsheet)	<ul style="list-style-type: none"> - RubricManager.js
DefaultCriterias	DefaultCriterias	<ul style="list-style-type: none"> - DefaultCriterias.js - DefaultHighlighterGenerator.js
New	New	<ul style="list-style-type: none"> - CustomCriteriasManager.js

UserFilter	UserFilter	- userFilterWrapper.html - UserFilter.js
GroupSelector	GroupSelector	- ConfigDecisionHelper.js - hypothesisGroupContentScript.js
Toolset	Toolset	- toolset.html - Toolset.js - ToolsetBar.js
Canvas	Canvas	- overview.png - reviewCanvas.html
Screenshots	Screenshots	- screenshots.png - screenshots.html - Screenshots.js
Report	Report	- generator.png
BackLink	BackLink	- BackLink.js
BackToSpreadsheet	BackToSpreadsheet	- spreadsheet.svg
AllDeleter	AllDeleter	- deleteAnnotations.png
Last	Last	- resume.png
DOI	DOI	- DoiManager.js
ACM	ACM	- acmContentScript.js
Validations	Validations	- validate.png - ValidateAnnotationManager.js
Strengths	Strengths	- majorConcern.png - minorConcern.png - strength.png - exporter - ReviewAssistant.js

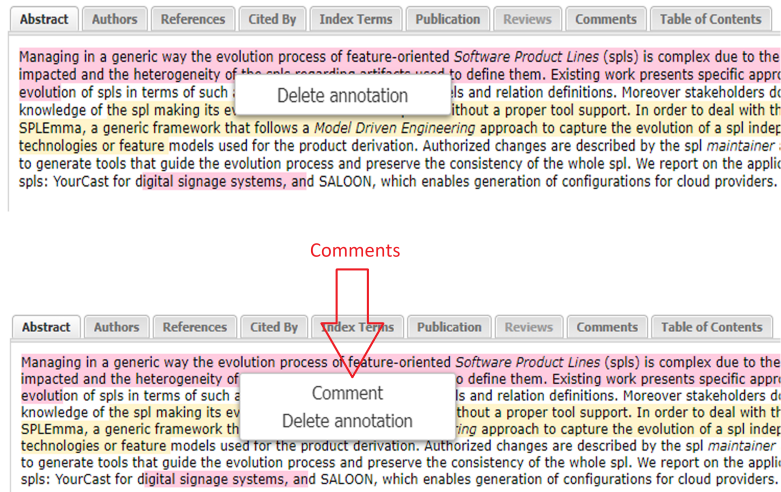
7.2 Taula: *Family Modelean ezarritako murriztapenen taula.*

8. KAPITULUA

*WebAnnotationSPL*tik lortutako produktuak

Atal honetan *WebAnnotationSPL*tik lortu diren produktuak edo aldakiak deskribatuko eta alderatuko dira. 8.1 taulan ikusi daiteke bakoitza lortzeko aukeratu diren ezaugarriak. Jarraian aurkeztuko diren produktuek parentesien artean 8.1 taulan duten izena adierazita daukate. Hauek dira lortutako 10 produktuak:

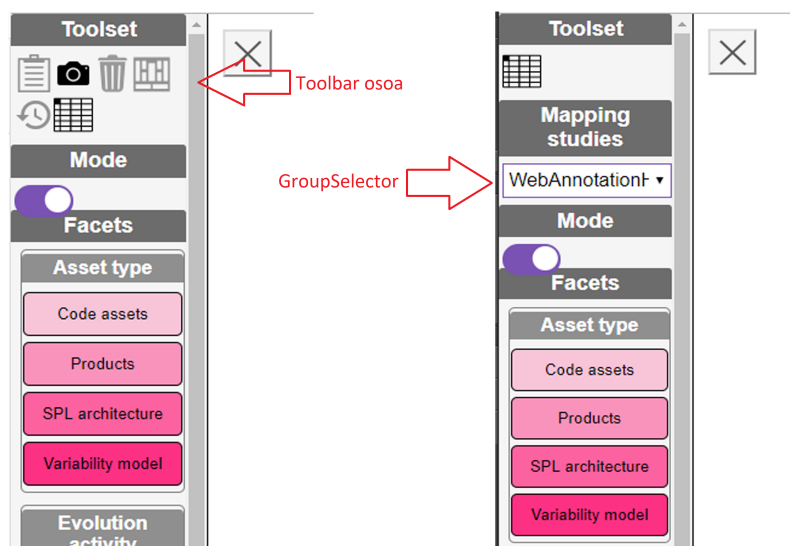
- **WebAnnotationH (H1):** jatorrizko *Highlight&Go* aplikazioak dituen funtzionalitate berdinak ditu.
- **WebAnnotationHwithAllDeleter (H2):** produktu honek aurrekoaren funtzionalitate gehienak ditu. Desberdintasun bakarrak *BackToSpreadsheet* ezaugarria ez dela aukeratu eta *AllDeleter* ezaugarria aukeratu dela dira. *AllDeleter*ri esker egindako anotazio guztiak ezabatu daitezke.
- **WebAnnotationHwithCanvas (H3):** produktu honek **WebAnnotationH** produktuak dituen funtzionalitate berdinak ditu, baina, aplikazio honetan egindako anotazioen laburpen bat HTML taula batean ikusteko aukera dago (*Canvas*).
- **WebAnnotationHwithComments (H4):** lehenengo produktuaren eta produktu honen arteko desberdintasun bakarra, azkeneko honek *Comments* ezaugarria aukeratu daukala da, hau da, aplikazio honetan anotazioetan iruzkinak jartzeko aukera dago. 8.1 irudian ikusi daiteke jatorrizko *Highlight&Go* aplikazioaren eta aldaki honen arteko desberdintasuna.



8.1 Irudia: *WebAnnotationH* eta *WebAnnotationHwithComments* produktuen arteko alderaketa.

- **WebAnnotationHwithoutUserFilter (H5):** produktu hau, **WebAnnotationH** produktuaren ezaugarri berdina dugu, baina, kasu honetan ezingo dira autoreen arabera anotazioetatik nabigatu *UserFilter* ezaugarria ez delako aukeratu.
- **WebAnnotationHwithReport (H6):** produktu honetan, **WebAnnotationH** produktuaren ezaugarriak aukeratu dira (*BackToSpreadsheet* izan ezik), baina, kasu honetan egindako anotazioen testu fitxategi bat deskargatzeko aukera dago *Report* ezaugarria aukeratu delako.
- **WebAnnotationHwithScreenshots (H7):** aurrekoaren berdina, baina kasu honetan *Report* ezaugarria egon beharrean *Screenshots* ezaugarria dago, hau da, anotazioen laburpena testu fitxategi batean gorde beharrean irudiekin sortutako fitxategi batean gordeko da, pantaila-irudiekin, alegia. Gainera, kasu honetan ere Googleko kalkulatu orrira bueltatzeko esteka (*BackToSpreadsheet* ezaugarria) kenduta dago.
- **WebAnnotationHwithToolbarAndStaticGroup (H8):** *Highlight&Goko* domeinuan oinarritutako azkeneko produktua nahiko desberdina da **WebAnnotationH** produktuekin alderatuta. Kasu honetan *GroupSelector* ezaugarria aukeratu beharrean, *StaticGroupSelector* ezaugarria aukeratuta dago. Gainera, *Toolset* ezaugarriak multzokatzen dituen ezaugarri guztiak aukeratuta daude (domeinu honetarako aukeratu daitezkeen ezaugarriak, noski). Produktu honek ez du *UserFilter* ezaugarria.

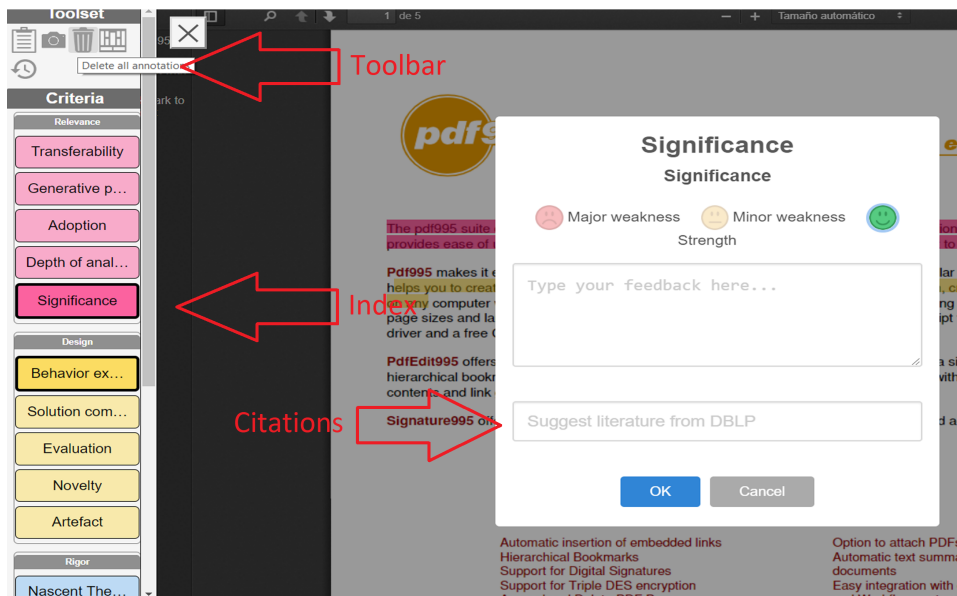
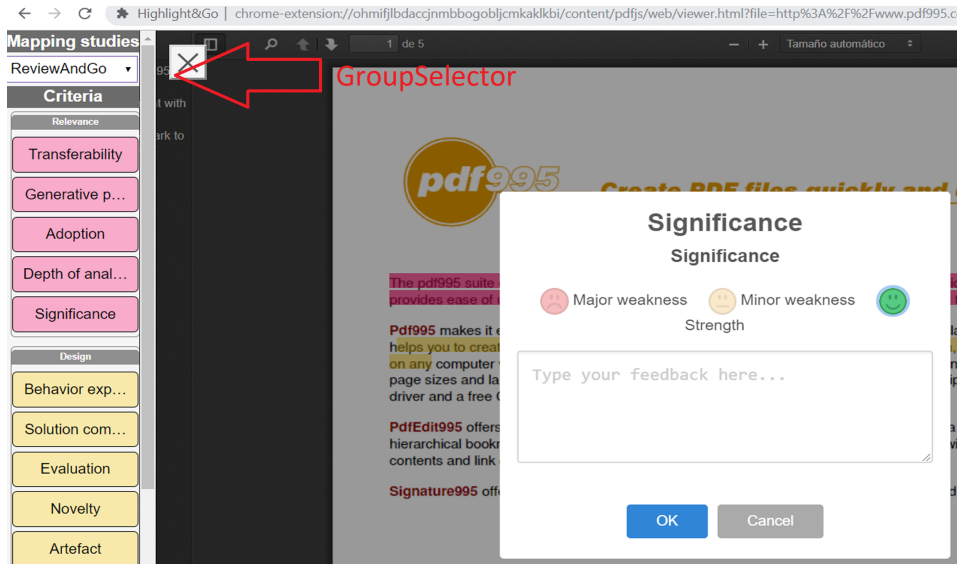
garria, baina, badu iruzkinak jartzeko aukera (*Comments*). H1 eta H8 produktuen arteko konparazio grafikoa 8.2 irudian ikusi daiteke.



8.2 Irudia: *WebAnnotationH* eta *WebAnnotationHwithToolbarAndStaticGroup* produktuen arteko alderaketa.

- **WebAnnotationR (R1):** jatorrizko *Review&Go* aplikazioaren funtzionalitate guztiak ditu.
- **WebAnnotationRwithGroupSelector (R2):** produktu hau, aurrekoarekin alderatuta, nahiko ezaugarri desberdinak ditu: *Hypothes.is*eko taldea modu estatikoan (*StaticGroupSelector*) aukeratu beharrean, *Hypothes.is*eko taldea eskuz aldatzeko aukera dauka (*GroupSelector*), zita bibliografikoak egiteko aukera kenduta dauka (*Citations*), ez dauka *Toolbar*eko erremintarik, ez dauka *Index* ezaugarria aukeratuta eta ezin dira irizpide berriak sortu (*New*). Jatorrizko *Review&Go* aplikazioaren eta aplikazio honen arteko desberdintasunak modu grafikoan 8.3 irudian ikusi daitezke.

Esan bezala, hurrengo taulan (8.1 taulan) aipatutako produktuak lortzeko egindako ezaugarrien aukeraketa ikusi daiteke modu grafikoan: ixa batekin (x) adierazitako ezaugarriak hautatu dira.



8.3 Irudia: WebAnnotationR eta WebAnnotationRwithGroupSelector produktuen arteko alderaketa.

Ezaugarriak	H1	H2	H3	H4	H5	H6	H7	H8	R1	R2
SpreadsheetView	X	X	X	X	X	X	X	X		
Spreadsheet	X	X	X	X	X	X	X	X		
DefaultCriterias									X	X
Index	X	X	X	X	X	X	X	X	X	
New									X	
UserFilter	X	X	X	X		X				
ReviewMode									X	X
HighlightMode	X	X	X	X	X	X	X	X		
IndexMode	X	X	X	X	X	X	X	X		
StaticGroupSelector								X	X	
GroupSelector	X	X	X	X	X	X	X			X
Canvas			X					X	X	
Screenshots							X	X	X	
Report						X		X	X	
BackToSpreadsheet	X		X	X	X		X	X		
AllDeleter		X						X	X	
Last								X	X	
DOI	X	X	X	X	X	X	X	X		
Dropbox		X	X		X	X	X	X		
ACM	X	X	X	X	X	X	X	X		
Comments				X				X	X	X
Citations									X	
Validations	X	X	X	X	X	X	X	X		
Strengths									X	X

8.1 Taula: Produktu bakoitza lortzeko aukeratu diren ezaugarrien taula.

10 produktu esanguratsuenak egitea erabaki da, baina, dezente gehiago sortu daitezke inplementatutako *WebAnnotationSPL*tik, adibidez, *Review&Goko* domeinuko aldaki gehiago sortu daitezke *Highlight&Goko* domeinuan oinarritzen diren produktuekin (8.1 taulako P1 eta P8 bitarteko produktuak, horiek barne) egin den moduan.

9. KAPITULUA

Garapenean zehar sortutako arazoak eta egindako lan gehigarria

Gradu Amaierako Lanaren azkeneko emaitzari erreparatuta pentsa daiteke atzetik dagoen lana sinplea dela, lanaren mamia kodean etiketa batzuk jartzea besterik ez dela, baina, hori ez da horrela izan. GrAL honen benetako esfortzua kodearen azterketan, ezaugarri diagramaren diseinuan eta produktu desberdinetan ezaugarriak integratzeko egin beharreko inplementazioan egon da.

Prozesu horietan edukitako arazo edo zailtasunak ugariak izan dira. Hainbeste izan dira non apirilaren amaieran birplangintza bat egin behar izan dela irismena jaisteko (*Mark&Go*-ekin zerikusia zuten ezaugarriak baztertzea erabaki zen; ikusi 3. kapitulua informazio gehiagorako).

9.1 Arazo eta zailtasunak

Jarraian aurkeztuko dira edukitako **zailtasun eta arazoak**:

- **Beste pertsona baten kodean lan egitearen zailtasunak:** Gradu Amaierako Lan honen zailtasun handiena izan da hau. Hiru produktuen (*Highlight&Go*, *Mark&Go* eta *Review&Go*) kodea sakonki aztertu eta ulertu behar izan da, berdinak eta desberdinak dituzten ezaugarriak identifikatu ahal izateko.

Hiru aplikazio horien inplementazioaren zatirik garrantzitsuenak *JavaScript*ez programatuta daude. Lengoia hau ulertzeko konplexua izan daiteke, gainera Informatikako Graduan ez da asko lantzen, beraz, beste pertsona batek inplementatutako kodearekin lan egitearen zailtasunari ondo ezagutzen ez den eta ulertzeko konplexua den lengoia batekin lan egitearen zailtasuna batu behar zaio.

Gainera, kodearen dokumentazioa oso eskasa da (kodean zehar iruzkin batzuk eta aplikazio bakoitzeko testu fitxategi bat *GitHub*en), baina, Haritz Medinaren laguntza eduki da (7 bilera guztira; 14 ordu inguru). Bilera hauetan Haritz Medinak garapenean sortu ziren errore batzuk kodean detektatzen lagundu du, *script*en arteko erlazio ugari baitaude eta hauek ez daude sekuentzia diagrama batean dokumentatuta. Beraz, Haritz Medinaren laguntza ezinbestekoa izan da errorearen detekzioa arintzeko.

- **Software Produktu-Lerroak garatzeko behar den denbora:** Software Produktu-Lerroak garatzeko prozesua luzea da:

1. Jatorrizko aplikazio guztiak aztertu ezaugarriak identifikatzeko.
2. Ezaugarri diagrama garatu eta hau *pure::variant*sera pasa (*Feature Model*).
3. Kode fitxategiak etiketatu/refaktorizatu.
4. Iturburuko kodea eta *Feature Model*a lotzeko *Family Model*a garatu.
5. Produktuak lortu eta probak egin, eta horien arabera zuzenketak planteatu prozesu guztia errepikatuz.

Nahiz eta Gradu Amaierako Lan honetako hiru software produktuak txikiak izan (*Highlight&Go* eta *Review&Go* aplikazioek 55 *script* dituzte eta *Mark&Go* aplikazioak 76 *script* ditu), prozesua luzea izan da, zeren eta ezaugarri bakoitza inplementatzeko prozesuko urrats bat baino gehiago behin eta berriro errepikatu behar izan da (batez ere 3 eta 5 urratsak). Esaterako, ezaugarri-diagramaren 6 bertsio aurkeztu dira azkeneko bertsioa lortu aurretik. Horrek esan nahi du prozesu guztia hasiera-hasieratik 7 aldiz errepikatu dela gutxienez.

- **Google Chromeko *debugger*arekin esperientzia falta:** beste pertsona baten kodearekin lan egitearen zailtasunari Google Chromeko *debugger*aren funtzionamenduaren ezagupen falta batu behar zaio. Garapen prozesuan edukitako arazorik handiena (denboraren kontsumoaren arabera) *TextAnnotator.js script*ean gertatu zen begizta infinitu bat konpontzea izan zen.

Arazo hau konpontzeko Chromeko *debuggera* erabili zen, baina ezin izan zen identifikatu arazoaren jatorria, beraz, arazo hau konpontzeko denbora asko kontsumitu da (30 ordu inguru), zeren eta *TextAnnotator scripta* oso luzea da (etiketak jarri ondoren *script* honek 1675 lerro ditu) eta beste hainbat *scriptekin* komunikatzen da (11 eta 13 inguru, amaierako produktuan aukeratzaren diren ezaugarrien arabera; hauek dira *script* horiek: *ContentAnnotator*, *ContentTypeManager*, *Tag*, *TagGroup*, *TagManager*, *Events*, *DOMTextUtils*, *LanguageUtils*, *PDFTextUtils*, *Alerts*, *Config*, *ModeManager* eta *RolesManager*). Arazo hau konpontzeko Haritz Medinaren laguntza behar izan da. Azkenean, *Events.js scriptean* zegoen arazoa.

- ***pure::variantsen* transformazioarekin edukitako arazoak eta informazio falta:** Nahiz eta Software Produktu-Lerroak garatzeko tresna erabilienetako bat izan, zaila da *pure::variants* inguruko informazioa aurkitzea. *pure::variantseko* web-orri ofizialean tutorial bat eta manual bat baino gehiago eskuragarri daude (<https://www.pure-systems.com/>), baina, zaila da hauetan arazo espezifikoaren konponbideak aurkitzea.

pure::variantsekin edukitako arazo nagusia *Family Modelarekin* izan da. Ingurune honek behin iturburuko kodea egituratuta eta etiketatuta edukita, *Family Modelak* automatiko sortzeko aukera ematen du, baina, sorkuntza automatikoki horretan iturburuko fitxategi guztiak *file* motarekin jartzen ditu. Mota honetako fitxategietan ezarritako etiketak ez dira interpretatzen, hau da, fitxategiak dauden moduan (etiketekin eta guzti) itzuliko ditu amaierako produktuan.

Transformazioa ondo egiteko iturburuko fitxategiak *pvsclxml* motakoak edo *pvscltext* motakoak izan behar dira (gai hau 4.2 atalean lantzen da), mota hauetako etiketak erabili baitira.

Nahiz eta *file* motatik *pvsclxml* edo *pvscltext* motetara aldatzeko prozesua erraza izan (fitxategi bakoitzaren *properties* atalean mota adierazteko aukera dago), arazoa zein zen aurkitzeko denbora asko behar izan da (15 ordu inguru) informazio eta adibideen faltagatik. Arazo hau konpontzeko GrAL honen *pure::variantseko* proiektua tresnak berak eskaintzen duen *Weather Station* adibidearekin alderatu behar izan da.

9.2 Egindako lan gehigarria

Atal honen sarreran aipatu den moduan, domeinu bateko funtzionalitateak beste domeinu batean integratzeko kodea refaktorizatu eta inplementatu behar izan da. Jarraian **egindako**

lan gehigarria ikusiko da:

- ***Canvas, Report eta AllDeleter* ezaugarriak integratzeko egindako lana:** hiru ezaugarri hauek *Review&Go* aplikazioaren domeinuari estu lotuta zeuden, baina, *Highlight&Go* domeinuan antzeko funtzionalitateak egotea interesgarria zen, beraz, hauen funtzionamendua inplementatzea erabaki da.

Canvas eta *Report*en kasuan, aplikazioak itzultzen dituen emaitzak aldatu behar izan dira, zeren eta *Review&Go*ko domeinuan bi ezaugarri hauei lotutako funtzionalitateak anotazioen laburpen bat sortzen zuten. Horrela, honakoak erakusten zituzten: anotazio guztiak eta bakoitzari dagokion maila (*Strength, Minor weakness* eta *Major weakness*) HTML taula bat (*Canvas* ezaugarriari lotutako funtzionalitatea) edo testu fitxategi bat (*Report* ezaugarriari lotutakoa) sortuz. Eta *Highlight&Go*ko kasuan, aldiz, anotazioak ez dira mailetan sailkatzen, balidatuta baizik. Ez da lortu anotazioak balidatuta dauden edo ez agertzea, horren ordez, anotazioen laburpen soil bat lortuko da.

AllDeleter ezaugarria *Highlight&Go*ko domeinuan integratzeko anotazio guztiak *Spreadsheet*etik ezabatzea falta zen. Hau izan da inplementatu behar izan dena, anotazio guztiak dokumentutik ezabatu ondoren Google *Spreadsheet*eko taulatik ezabatzea ere.

- ***specificContentScript.js* eta *toolsetBar.js* scripten berfaktORIZAZIOA:** bi kasu haue-tan kodea berfaktORIZATU behar izan da hainbat funtzionalitate domeinu desberdine-tan integratu ahal izateko.

Jatorrizko aplikazioen *scripts/specific* karpetak alderatuz, ikusi daiteke bakoitzak karpeta horretan funtzionalitate desberdinak dituela. Kasu bakoitzean funtzionalitate guzti horiek batzen dituen *script* nagusia (*main script*) desberdina da: *Highlight&Go*ren kasuan *script* horrek *SLRDataExtractionContentScript.js* izena dauka, *Mark&Go*ren kasuan *ExamDataExtractionContentScript.js* eta *Review&Go*ren kasuan *ReviewContentScript.js*.

*WebAnnotationSPL*rekin sortutako aldakietan jatorrizko hiru aplikazioen *specific* karpeta barruko funtzionalitateak nahasita agertu daitezke, adibidez, *Highlight&Go*ko aldaki batean, jatorrizko aplikazio horren funtzionalitateekin batera, pantaila-irudiak ateratzeko funtzionalitatea lortu daiteke, nahiz eta funtzionalitate hori hasiera batean *Highlight&Go*koa ez izan. Hori lortu ahal izateko jatorrizko hiru aplikazioen *specific* karpetako *script* nagusiak aztertu dira hauen nahasketa bat sortzeko. Nahasketa horren ondorioa *specificContentScript.js* scripta izan da. *Script* horren bitartez

jatorrizko hiru aplikazioen *specific* karpeta barneko funtzionalitateak hiru aplikazioen aldakietara integratzea lortzen da.

Bestalde, jatorrizko *Mark&Go* eta *Review&Go* aplikazioetan *Toolset* ezaugarriak multzokatzen dituen funtzionalitateak era desberdinetan inplementatuta daude. *specificContentScript.js* scriptarekin egin den moduan, bi inplementazio modu horiek aztertu behar izan dira kasu guztietarako balio duen kodea lortzeko. Inplementazio berri hori *toolsetBar.js* scriptan egin da.

- **Produktuen *Node.js*ko mendekotasunak instalatzeko prozesua azkartzeko *Batch scripta*:** D eranskinean aipatu dira produktu bakoitza lortzeko jarraitu beharreko urratsak. Bertan ikusten denez, *Node.js*-ko mendekotasunak instalatzeko 15 minutu inguru itxaron behar da. Exekuzioetan gertatzen ziren eten luze horiek ekiditeko, produktu bat finko utzi da bertan mendekotasunak instalatuta mantentzeko. Ondoren sortuko diren produktueta mendekotasunak instalatu beharrean, finko mantendu den produktu horretako mendekotasunetara esteka bat sortuko da. Horretarako *Batch script* bat sortu da. *Script* horrek, sortu den produktuaren izena eskatuko du, eta produktuaren izena jakinda, sortuta mantendu den produktuko mendekotasunetara esteka bat sortuko du *mklink* komandoarekin adierazi den produktuan. Modu horretan 15 minutuko etenaldi horiek ekiditen dira.

10. KAPITULUA

Jarraipen eta kontrola

Kapitulu honetan Gradu Amaierako Lanean lortutako produktua, egindako kudeaketa eta idatzitako dokumentazioa planifikatutakoarekin alderatuko da lortu den emaitzaren kalitatea bermatzeko.

10.1 Helburu eta irismenaren betetze maila

Hasierako plangintzan ezarritako helburua jatorrizko hiru aplikazioak (*Highlight&Go*, *Mark&Go* eta *Review&Go*) garatzeko balio zuen SPL bat sortzea zen, baina, birplangintzan (ikusi 3. kapitulua) *Mark&Go*ren ezaugarriak alde batera uztea erabaki zen 9. kapituluan aipatu diren arazoengatik. Beraz, birplangintzaren arabera, GrAL honekin *Highlight&Go* eta *Review&Go* aplikazioak automatikoki sortzeko balioko zuen Software Produktu-Lerro bat implementatu nahi zen.

8. kapituluan ikusi den moduan, bi aplikazio horiek lortu dira, eta ez hori bakarrik, horien beste 8 aldaki desberdin ere lortu dira ere. Ondorioz, esan daiteke irismenaren aldetik motz gelditu dela GrALa, *Mark&Go* aplikazioa alde batera utzi delako, baina, beste aldetik hasierako muga gainditu da, beste bi aplikazioen aldaki asko lortu direlako. 10.1 irudian berdez ikusi daitezke implementatu diren ezaugarriak, eta txuriz egin gabe gelditu direnak. Ez da behar den bezala konprobatu zer gertatzen den *SpreadsheetView* ezaugarria aukeratzen ez denean, eta ez da lortu ere *Review&Go*ko aldaki batekin egindako anotazioak *Spreadsheet*eko taula batean bistaratzea. *Citations* ezaugarriaren atzean dagoen funtzionalitatea ez da *Highlight&Go*ko aldaki batean konprobatu.

Bestalde, 9. kapituluan aipatu den moduan, aplikazio batzuen funtzionalitateak beste aplikazioen aldaketan integratzeko lan gehigarria egin behar izan da kode zatiak berfaktortuz, eta kasu batzuetan (*Canvas*, *Report* eta *AllDeleter* ezaugarrien kasuan) inplementatuz ere.

10.2 Denbora desbiderapenak

3. kapituluko birplangintzan ikusi zen jada nola luzatu ziren garapeneko prozesuak. 10.2 irudian ikusi daiteke nola nahiz eta ataza gehienak burutzeko erabilitako denbora balioetsi zen denbora tartean egon, garapen prozesua hainbeste luzatu da non garapen osoaren ordu kopurua (175 ordu) estimatuarena bainoa altuagoa izan dela (247 ordu, estimatutakoa baino 15 ordu gehiago).

Luzatu den garapen prozesu horretan denbora gehien eraman duten atazak *etiketatze prozesua* eta *probak* izan dira. Hauen luzapenaren arrazoia produktu bakoitza lortu aurretik horien funtzionalitate bakoitza banaka probatu behar izan dela da, egindako aldaketa bakoitzarekin produktua berriro exekutatzuz (9. kapituluan aipatu da arazo hau: **Software Produktu-Lerroak garatzeko behar den denbora**).

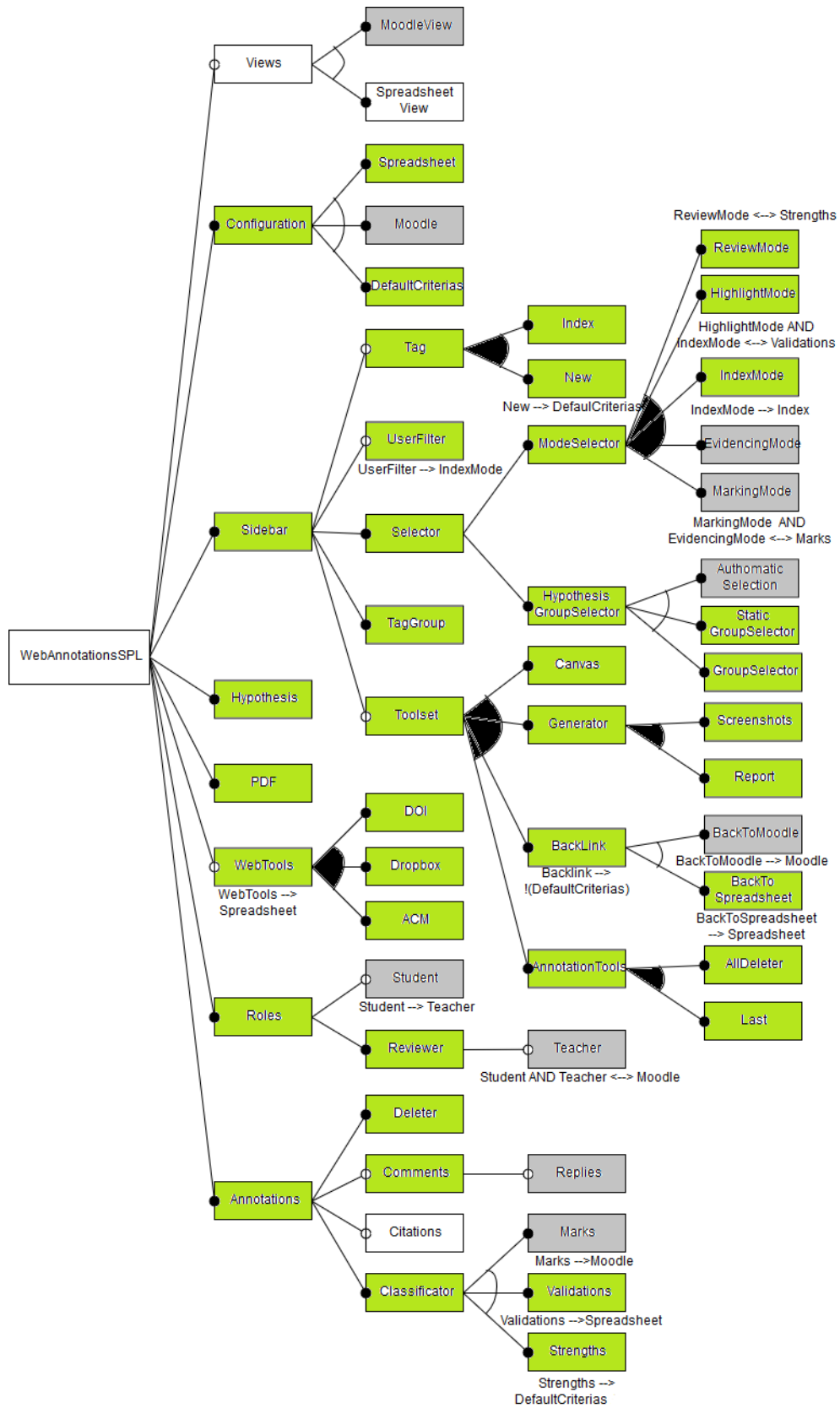
Luzatu den beste ataza bat memoriaren idazketa izan da, honi denbora gehiago eskaini nahi izan zaiolako zuzenketak eginez eta azalpenak hobetuz memoriaren kalitate ona lortzeko asmoz. Desbiderapenak kontuan edukita, GrALaren ordu kopuru totala 414 ordu ingurukoa izan da (birplangintzan estimatutakoa baino 34 ordu gehiago).

10.3 Gantt diagrama eta emangarrien kontrola

2. kapituluan erakutsi zen Gantt diagrama eta errealitatearen Gantt diagramaren arteko (10.3 irudian ikusi daitekeen diagrama) desberdintasunak oso txikiak dira. Nahiko ondo kontrolatu da prozesuen iraupena. Aipatu daitekeen desberdintasun nabariak dira memoriaren garapena pentsatu zena baino lehenago hasi dela, ikasketa prozesua hasieran batez ere egin dela eta probak garapen osoan zehar egin direla.

Emangarrien entrega data ofizialak honakoak izan dira:

- **Proiektuaren plangintza eta memoriaren sarrera:** lehenengo bertsioa otsailaren 14ean entregatu zen (aurreikusi zen data baino egun bat lehenago), baina, hainbat



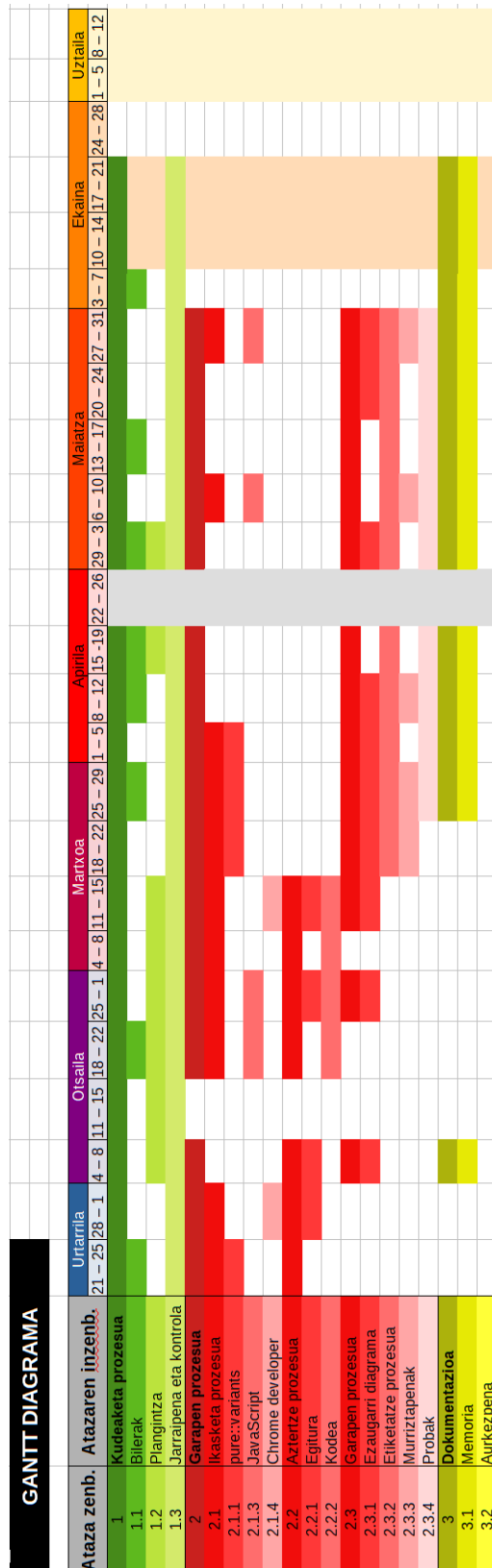
10.1 Irudia: Implementatzea lortu diren ezaugarriak adierazten duen ezaugarri diagrama.

ATAZAK	ESTIMAZIOA	ERREALA	DESBIDERAPENA
Garapena	232	247	15
Ikasketa prozesua	23	18	-5
pure::variants	12	9	-3
JavaScript	5	2	-3
Chrome developer	6	4,5	-1,5
Aztertze prozesua	54	54	0
Egitura	11	11	0
Kodea	43	43	0
Garapena prozesua	155	175	20
Ezaugarri diagrama	20	20,5	0,5
Etiketatzeko prozesua	90	105,75	15,75
Murriztapenak	15	12	-3
Probak	30	36,75	6,75
Kudeaketa	63	66,75	3,75
Bilerak	13	16,5	3,5
Plangintza	35	35	0
Jarraipen eta kontrola	15	15,25	0,25
Dokumentazioa	85	100,5	15,5
Memoria	70	80,5	10,5
Aurkezpena	15	20	5
GUZTIRA	380	414,25	34,25

10.2 Irudia: GrALeko ataza bakoitzean kontsumitutako denbora adierazten duen taularen irudia.

zuzenketa direla medio, emangarri honekin martxoaren 14ean amaitu zen, aurreikusitako zena baino hilabete bat beranduago.

- **Ezaugarri-diagrama:** ezaugarri diagramaren lehenengo bertsioa martxoaren 7an bidali zen (balioetsitako daten barnean), baina, azkeneko bertsioa martxoaren 26an bidali zen. 2. kapituluak aipatu zen martxoaren 15erako ezaugarri guztiak kodean identifikatuta egon behar zirela, baina, errealitatea oso desberdina izan da, kodeko etiketak etengabe zuzendu dira, SPLaren garapena amaitutzat eman arte, alegia.
- **SPLren prototipoak:** ekainaren 5ean implementazioari bukaera eman zitzaion, espero zena baino 12 egun beranduago. Hala ere, aipatu behar da implementazioaren nondik norakoa etengabe erakutsi zaiela proiektuen interesatuei, batez ere jatorrizko aplikazioen garatzailea den Haritz Medinari, honen laguntzarekin hainbat arazo konpondu direlako (arazoen informazio gehiagorako 9. kapituluak irakurri).
- **Memoriaren bertsioak:** memoria atalka entregatzen joan da, memoriako 3, 4 eta 5. kapituluak apirilaren 15ean entregatu ziren, 6 eta 7 kapituluak maiatzaren 8an eta 8, 9 eta 10. kapituluak ekainaren 11n.



10.3 Irudia: Proiektuaren prozesuen iraupen erreala adierazten duen Gantt diagrama.

10.4 Lortutako kalitatea

Kalitatearen ebaluazioa egiteko birplangintzan (3. kapitulua) hartutako erabakia kontuan hartuko da, hau da, *Mark&Gore*kin zerikusia duen guztia ez da kontuan hartuko emaitzak baloratzeko. Kalitatea 2. kapituluan hiru zatitan banatu zen: produktuaren kalitatea, dokumentazioaren kalitatea eta kudeaketaren kalitatea. Hiru zati horiek banaka aztertuko dira jarraian.

10.4.1 Produktuaren kalitatea

Nahiz eta SPLtik hiru aplikazioak ez lortu, jatorrizko *Highlight&Go* eta *Review&Go* aplikazioak sortzea lortu da, eta ez hori bakarrik, horien aldaki desberdinak lortu dira ere. Egia da ez dela funtzionalitate berririk implementatu, baina kodea berfaktORIZATU behar izan da, eta beste kasu batzuetan funtzionalitateak domeinu desberdinetan integratzeko, hauek implementatu behar izan dira berriro.

Hori guztia eta 2. kapituluren kalitate-plana kontuan hartuta, lortutako Software Produktu-Lerroaren kalitatea *ona* dela esan daiteke.

10.4.2 Dokumentazioaren kalitatea

2. kapituluan esan bezala, memorian akats ortografikoak ekiditeko Xuxeneko *plugina* erabili da. Gainera, Gradu Amaierako Lanaren zuzendariari kapituluka bidali zaio memoria honek zuzenketak gomendatzeko.

Horrekin batera, behin memoriaren garapena amaitu dela, bi irakurketa trinko egin dira azalpenak hobetzeko asmoarekin. Esandako guztia kontuan hartuta, esan daiteke memoriaren kalitatea altua dela.

10.4.3 Kudeaketaren kalitatea

Planifikatu zen moduan (ikusi 2. kapitulua), ostiralero Gantt diagrama berri batean jorratutako atazak markatu dira. Ostiralero ere kontrolatu da zein atazetan kontsumitu den planifikatu zen denbora baino gehiago. Bestalde, egunero anotatu dira kontsumitutako orduak dagokion atazan (ikusi F eranskina) eta bilera baten ondoren aktak idatzi dira (ikusi E eranskina).

Birplangintzan guztira 380 ordu sartuko zirela aurreikusi zen, baina, errealitatean 414 ordu sartu dira, horrek esan nahi du %11ko desbiderapena egon dela, hau da, kudeaketaren kalitatea altua izan da.

11. KAPITULUA

Ondorioak

Kapitulu honetan GrAL-aren inguruko hausnarketak eta ondorioak azalduko dira, baita horren garapenean ikasitakoa eta etorkizunera begira emandako aholkuak ere.

11.1 Gradu Amaierako Lanari buruzko hausnarketak eta ondorioak

Gradu Amaierako Lan honen emaitzak positiboak direla esan daiteke aurreko kapituluan aipatutako arazoak, lortutako emaitzak eta egindako lan gehigarria kontuan hartuta. Hala eta guztiz ere, hainbat alditan aipatu den moduan, proiektuaren hasierako irismena jaitsi zen apirilaren bukaeran 9. kapituluan aipatu diren arazoengatik.

WebAnnotationSPLa egiteko erabili diren jatorrizko bi aplikazioak (*Highlight&Go* eta *Review&Go*) tamaina txikikoak diren arren (55 *script* bakoitza), GrAL honekin ikusi da zenbat denbora behar den SPL batekin emaitzak ikusten hasteko. Lan honen mamia aplikazioen kodearen azterketan dago, kodea guztiz ulertuz gero ezaugarriak erraz detektatu daitezkeelako eta errazago detektatzen dira ezaugarriak kodean etiketak azkarrago ezarriz.

Gainera, GrAL honi esker, teknologia berriak ezagutu dira: *pure::variants*, *Chrome developer*, *DiffMerge* eta *BUT4Reuse*. Informatikako Graduan asko lantzen ez den lengoia batekin lan egin da ere, *JavaScript* lengoiairekin, alegia.

Arazoak sortu diren momentuan profesionaltasunez jokatu da, lehendabizi arazoa inde-

pendenteki konpontzen saiatuz, eta ezin izan denean arazoa konpondu, laguntza eskatuz. Hemen Haritz Medinaren eta Arantza Irastorzaren lana aitortu behar da, lehenengoari esker garapenean sortu diren hainbat errore konpontzea lortu delako, eta bigarrenari esker memoriaren kalitatea handitu delako.

11.2 Lanarekin ikasitakoa

Gradu Amaierako Lan honen ikasketa nagusia proiektu bat guztiz kontrolatzea izan da, proiektuaren plangintza ahal den neurrian betetzea, eta hori ez bada bete, desbiderapenak justifikatzea. Horrekin batera, proiektuan zehar interesatuekin bilerak egin behar izan dira proiektuaren helburuak lortzen direla kontrolatzeko eta gauzak zuzentzeko.

Software Produktu-Lerroak garatzeko prozesua landu da, horretarako, nahiko erabilia den tresna batean oinarrituta (*pure::variants*). Bestalde, beste pertsona baten kodea aztertzea, ulertzea eta horren gainean lan egitea ikasi da. Gainera, asko kontrolatzen ez zen lengoia batean (*JavaScript*) idatzitako kodea interpretatzen eta kodetzen ikasi da ere.

Proiektuaren gorabeheretatik ere ikasi da, nola jokatu arazoen aurrean konponbideak aurkitzeko, eta konponbiderik ez badago amore ematen jakitea (birplangintzarekin gertatu zen moduan, ikusi 3. kapitulua).

11.3 Etorkizunera begira

*WebAnnotationSPL*aren garapena amaitzeko *Mark&Go* ezaugarriak inplementatu behar dira, baita egin gabe gelditu diren ezaugarri batzuk garatzea: *Spreadsheet view* ezaugarria, *Citations Highlight&Goko* domeinuan funtzionatzen duela konprobatzea.

Prozesu hori errazteko, GrAL honen esperientziaz baliatuta, hona hemen gomendio batzuk: lehenengoa, produktuen exekuzioa astunegia ez izateko *script* bat sortu prozesua azkartzeko, bigarrena, aldaketak produktuetan probatu eta funtzionatzen badute, iturburu-kodean hauek ezarri, etengabe produktuak sortzen ez egoteko eta azkenik, Chromeko *debugger*arekin esperientzia edukitzeak asko lagunduko du erroreak azkar detektatzeko eta hauek azkar konpontzeko (gomendio hauen arrazoia 9. kapituluan aipatutako arazoak dira).

Eranskinak

A. ERANSKINA

Funtzionamendua

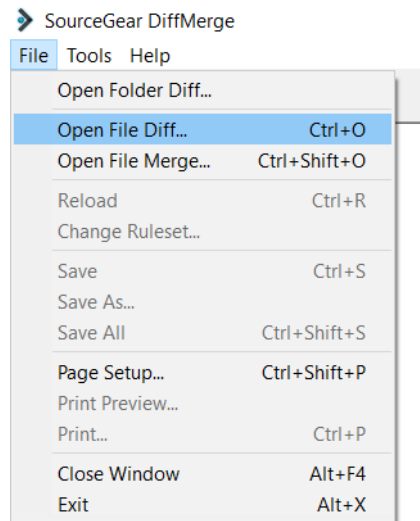
Eranskin honetan *Diff* eta *Merge* funtzionalitateen funtzionamendua aztertuko da. *Folder diff*-ena ez da aztertuko lan honetan funtzionalitate hori ez delako erabiltzen, ?? azpiatalean erabaki horren zergatia aipatuko da.

A.1 Diff

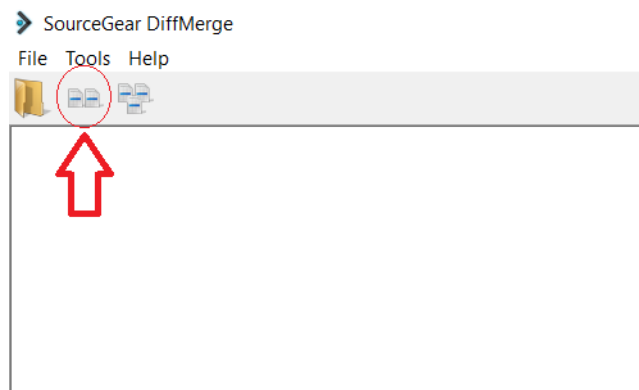
Bi fitxategiren egitura konparatzeko egin beharreko lehenengo urratsa teklatuko Ctrl+O sakatu, goialdeko menuan aurkitzen den bi orriko ikurra sakatu (A.2 irudia) edo *File* erlitzeko *Open file diff...* aukeratu behar da (A.1 irudia).

Ondoren A.3 irudiko menua agertuko da, bertan konparatu nahi diren fitxategiak aukeratu behar dira. Adibide honetan A direktorioko C1 eta B direktorioko C1 fitxategiak konparatuko dira.

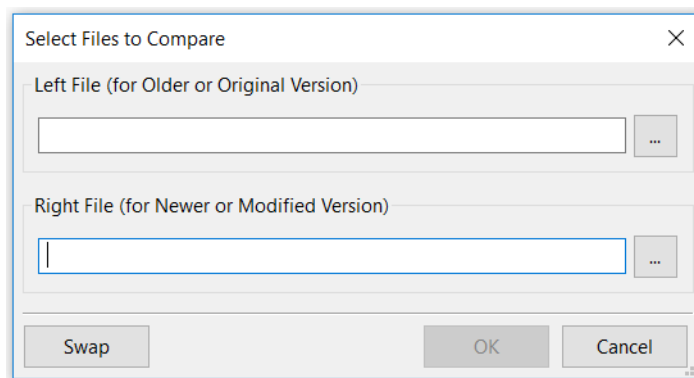
Behin fitxategiak aukeratu direla, A.4 irudiko erantzunaren antzeko zerbait agertuko da pantailan. Bi fitxategiek berdina duten edukia ez da markatzen, eduki desberdina, aldiz, kolore batekin azpimarratuko da, kasu honetan gorriz.



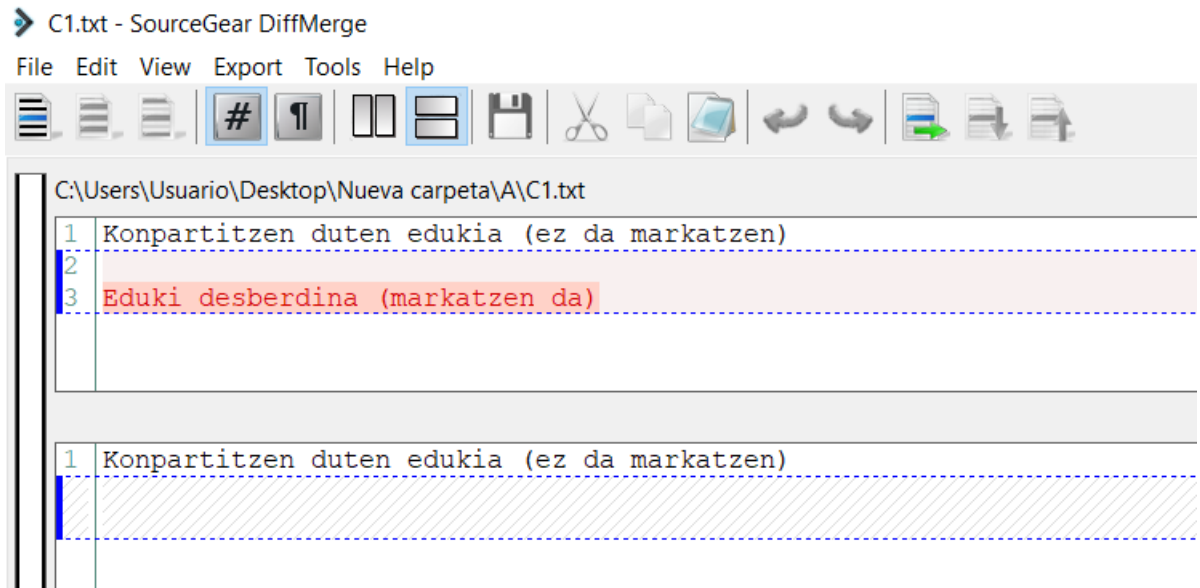
A.1 Irudia: File erlaintzeko *Open file diff...* aukera.



A.2 Irudia: Diff egiten hasteko sakatu beharreko ikurra



A.3 Irudia: konparatu nahi diren fitxategiak aukeratzeko menua.



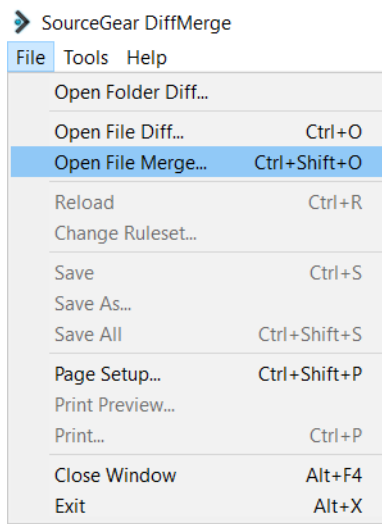
A.4 Irudia: *Diff* motako konparaketa baten adibidea.

A.2 Merge

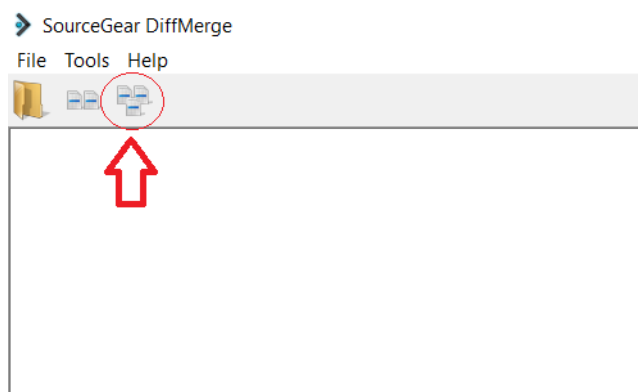
Merge funtzionalitateak hiru fitxategiren arteko konparaketak egitea ahalbidetzen du. Honekin hasteko **Ctrl+Shift+O** sakatu, goialdeko menuko hiru orrien ikurra sakatu (A.6 irudia) edo *File* erlaintzeko *Open file merge...* aukeratu (A.5 irudia).

Merge funtzionalitatea aukeratzekoan A.3 irudian ikusi daitekeen antzeko menu bat agertuko da, kasu honetan, hiru fitxategi aukeratzeko aukerarekin. Adibide honetan, A, B eta C direktorioen C2 fitxategia konparatuko da.

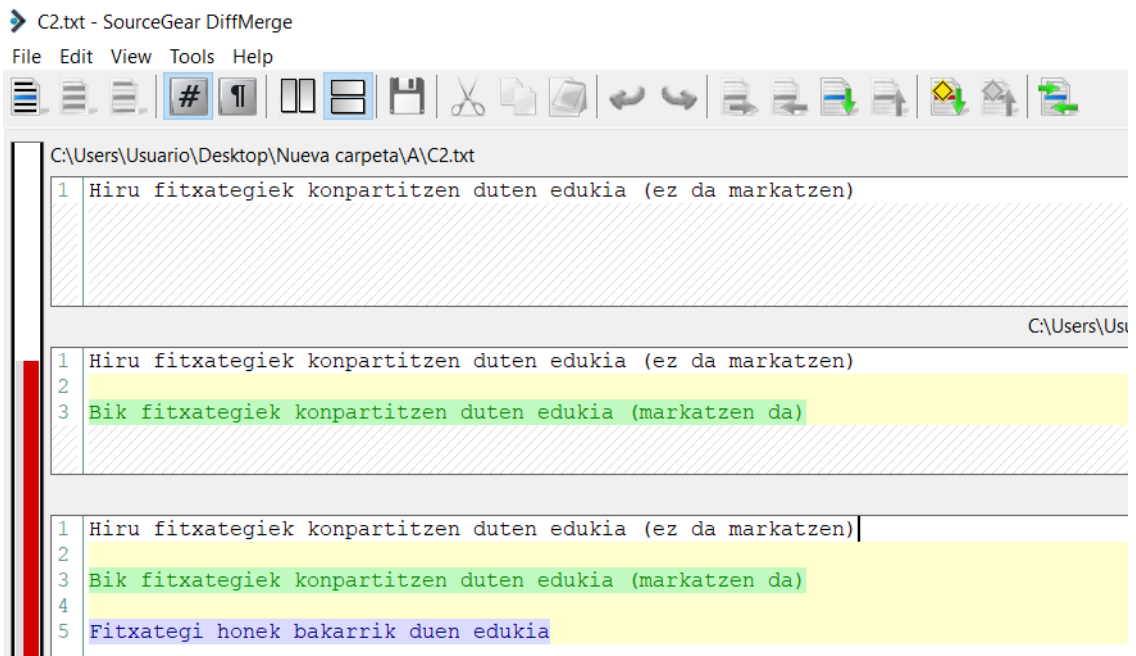
A.7 irudiko *Merge* konparaketan hiru fitxategiek berdina duten edukia ez da markatzen, bi fitxategik berdina duten edukia bi fitxategitan kolore berdinekin markatuko da (irudian berdez markatu da), azkenik, fitxategi bakarrak duen edukia beste kolore batez agertuko da (irudian morez).



A.5 Irudia: File erlaintzeko *Open file merge...* aukera.



A.6 Irudia: Merge-ren ikurra.



A.7 Irudia: Merge adibidea.

B. ERANSKINA

BUT4Reuse *plugin*-aren instalazioa eta funtzionamendua

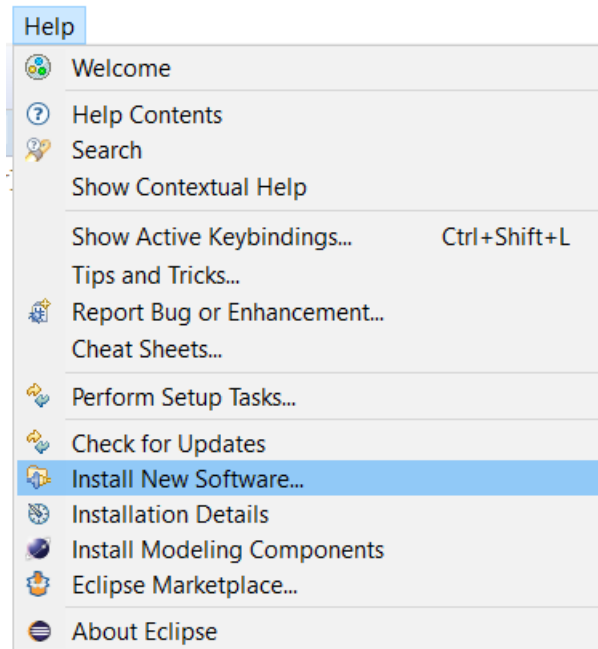
Eranskin honetan **BUT4Reuse** *plugin*-aren instalazioa eta funtzionamendua aztertuko dira:

B.1 *Plugin*-aren instalazioa Eclipse-n

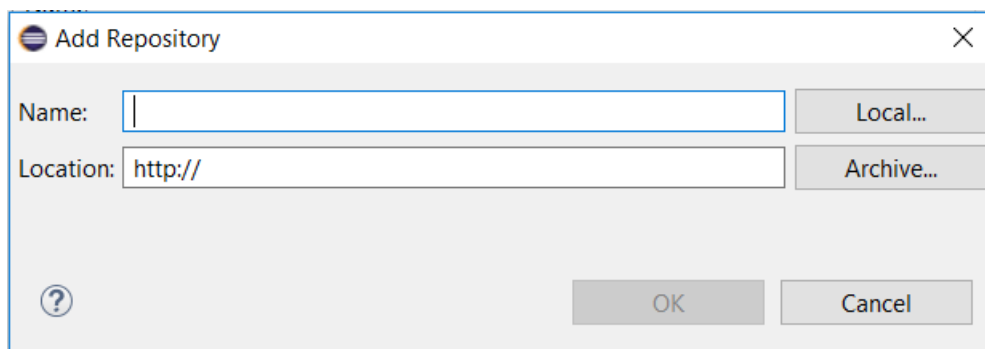
Instalazioarekin hasteko *Eclipse* ireki behar da, ondoren, *Help* erlaintzeko *Install New Software* aukeratu (ikusi B.1 irudia).

Agertuko den leihoan *Add* botoia sakatu behar da, hori egiterakoan beste leiho bat pantailaratuko da, *Name* eta *Location* eremuak dituen. *Name* eremuan edozein izen jarri daiteke, eremu hori ez delako garrantzitsua. *Location* eremuan, aldiz, honako link-a itsatsi behar da: <https://raw.githubusercontent.com/but4reuse/but4reuse/master/release/org.but4reuse.site/site.xml> (ikusi B.2 irudia). Urrats honekin amaitzeko, *Ok* botoia sakatu behar da.

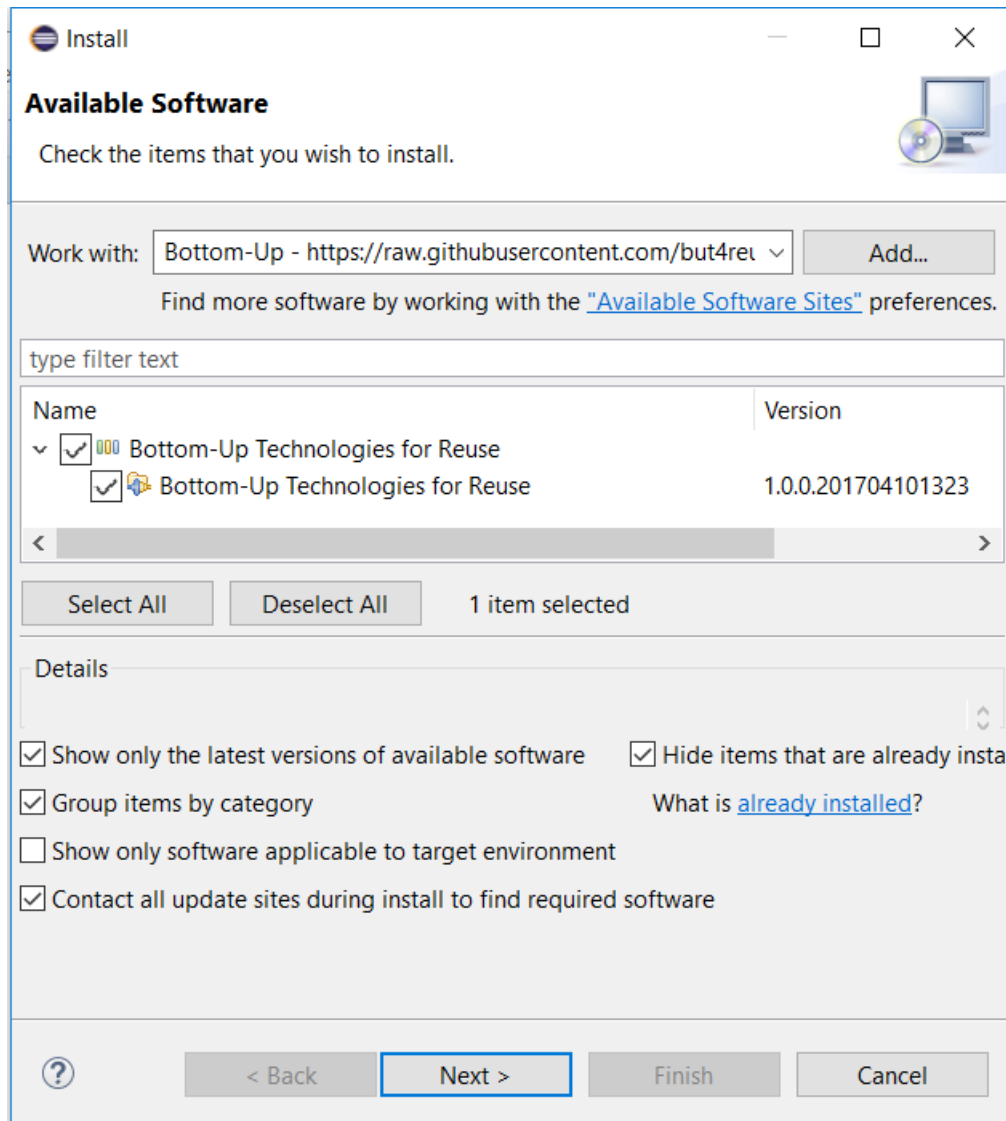
Instalazioa amaitzeko, leiho berrian erakusten den *plugin*-a aukeratu eta urratsak jarraituko dira (leiho horretan emandako izenarekin eta sartutako URL-arekin agertuko da *plugin*-a; B.3 irudiaren kasuan *plugin*-a *Bottom-up* izena du). Instalazio urratsak BUT4Reuse-ko *GitHub*-eko orrialde ofizialetik lortu dira.



B.1 Irudia: *Help* erlaitzeko *Install New Software* aukera. Iturria: [BUT4Reuse](#).



B.2 Irudia: *Name* eta *Location* eremuak dituen leihoa. Iturria: [BUT4Reuse](#).

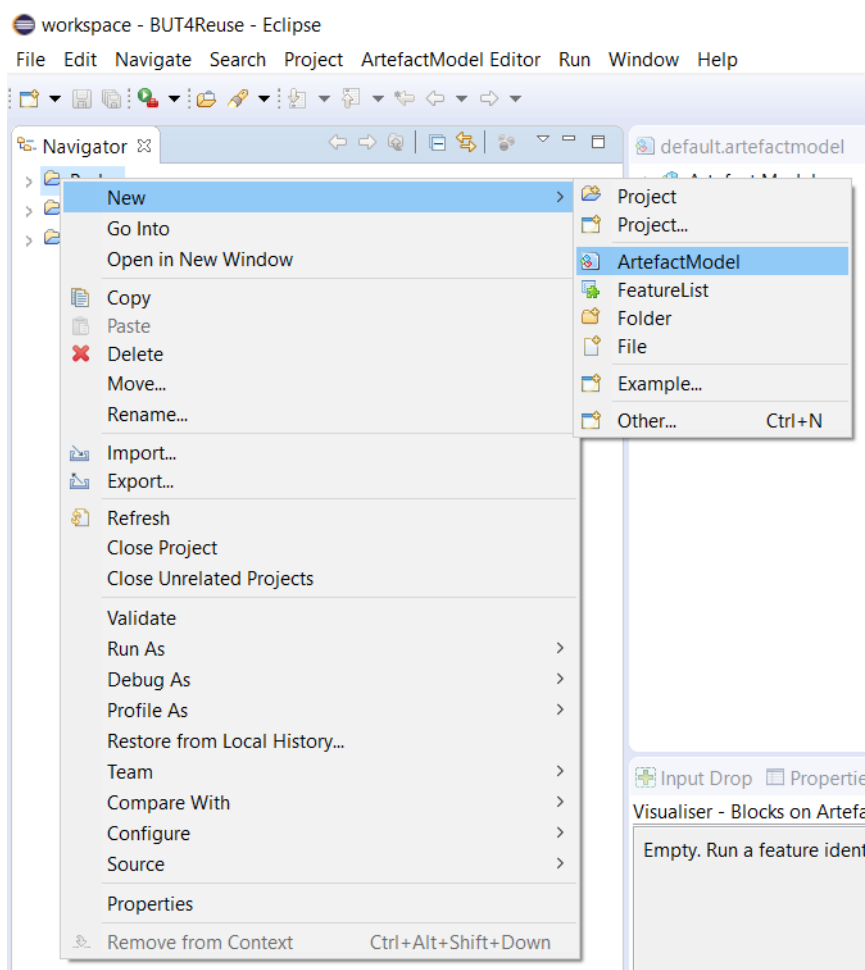


B.3 Irudia: Instalatu nahi den *plugin*-aren aukeraketa leihoa. Iturria: BUT4Reuse.

B.2 Funtzionamendua

Egin beharreko lehenengo gauza *Eclipse*-ren perspektiba aldatzea da, horretarako *Window* → *Select perspective* → *Other...* → *BUT4Reuse* sakatu behar da.

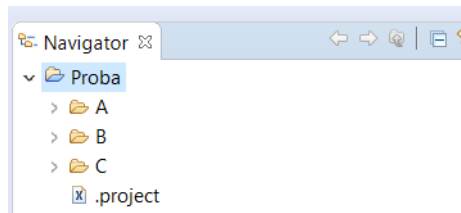
Proiektu berri bat sortzeko *File* → *New* → *Project* aukeratu behar da. Sortu ondoren, honen gainean saguaren eskuineko botoia sakatu eta menuko *New* → *ArtefactModel* aukeratu behar da, honek, egituraren arabera produktuak bloke desberdinetan banatuko ditu (ikusi [B.4](#) irudia).



B.4 Irudia: *ArtefactModel* sortzen.

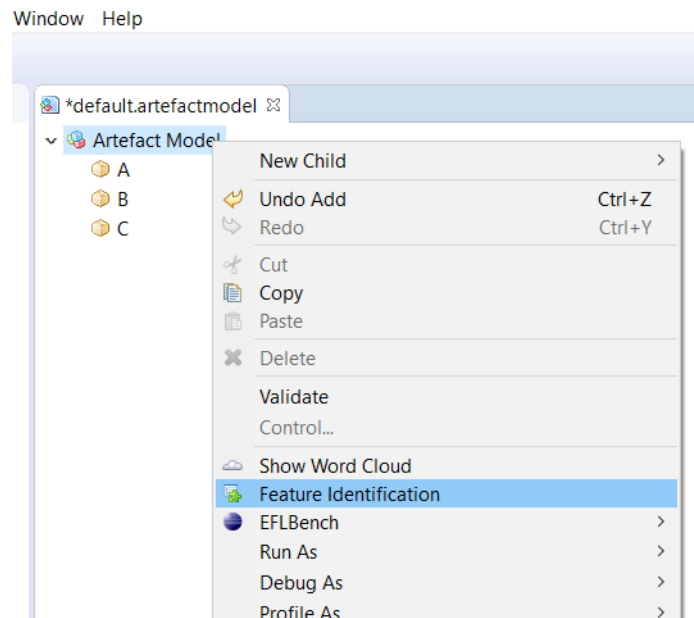
Hori egin ahal izateko, lehenengo konparatu nahi diren fitxategiak proiektu barruan itsatsi behar dira. Tutorial honetan A, B eta C direktorioak erabiliko dira ([B.5](#) irudia).

Ondoren, konparatu nahi den eduki guztia aurreko urratsean sortu den *ArtefactModel*-ra



B.5 Irudia: Konparatu nahi den edukia proiektuaren barruan itsatsita.

arrastatu behar da, eta *ArtefactModel* gainean saguko eskuineko botoia sakatuz *Feature Identification* aukeratu behar da (ikusi B.6 irudia).

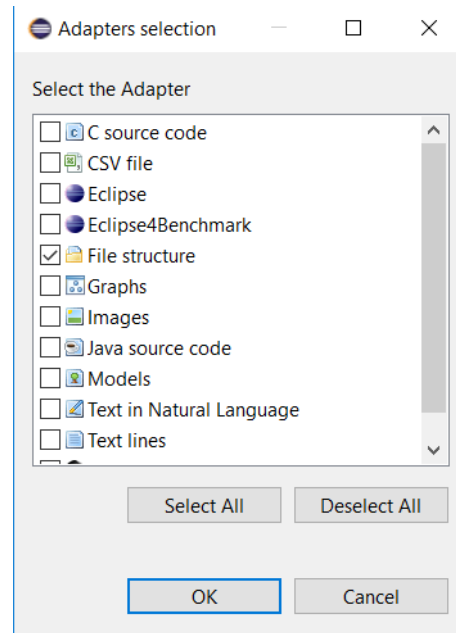


B.6 Irudia: *ArtefactModel*-eko *Feature Identification* aukera.

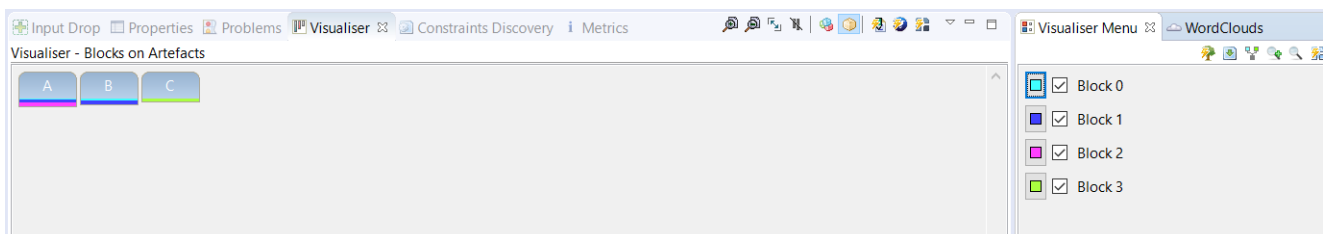
Lan honetan direktorioen egitura konparatu nahi da, beraz, aurreko urratsa jarraitu ondoren agertzen den leihoan *File structure* aukeratuko da (ikusi B.7 irudia). Hala ere, errepertatu daiteke hainbat aukera desberdin eskaintzen dituela *plugin* honek: *C source code*, *CSV file*, *Eclipse*, *Eclipse4Benchmark*, ***File structure***, *Graphs*, *Images*, *Java source code*, *Models*, *Text Natural Language*, *Text lines* eta *json*.

Behin konpilazioa amaitu dela, azpiko bi ataleko *Visualizer* erlaintzak interesgarrienak dira lan honetan egingo den ezaugarrien identifikaziorako, bertan A, B eta C direktorioen egituraren arabera blokeen sakabanaketa ikusi daitekeelako (B.8 irudia).

Nahi izanez gero, blokeen arabera karpetak sortu daitezke *Construct separately* auke-

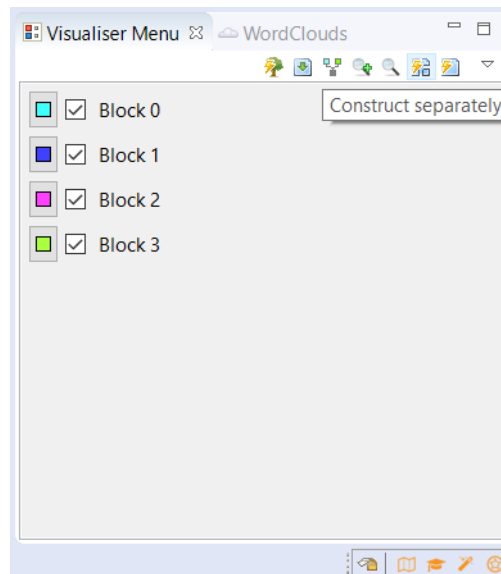


B.7 Irudia: *Adapters selection* menua.



B.8 Irudia: A, B eta C direktorioen egituraren arabera blokeen sakabanaketa.

rarekin (ikusi [B.9](#) irudia), hau da, kasu honetan *Block 0* (hiru direktorioek berdina duten egitura), *Block 1* (A eta B direktorioek konpartitzen duten egiura), *Block 2* (A direktorioak bakarrik duen egitura) eta *Block 3* (C direktorioak bakarrik duen egitura).



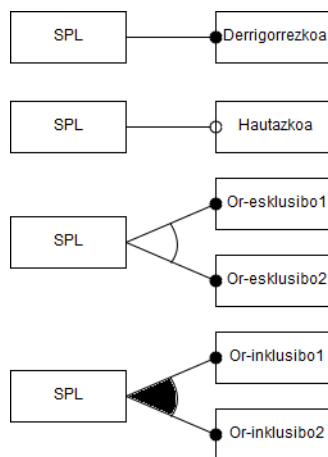
B.9 Irudia: *Construct separately* aukera.

C. ERANSKINA

Ezaugarri diagramaren nomenklatura

Atal honetan amaierako ezaugarri diagrama azalduko da. Ezaugarriak banan-banan azalduko dira. Hauen azalpenekin hasi aurretik ezaugarri diagraman agertu daitezkeen ezaugarrien arteko lotura motak gogoratuko dira:

- **Derrigorrezko ezaugarriak:** produktu guztietan agertu behar diren ezaugarriak. [C.1](#) irudiko lehenengo kasuan ikusten da nola adierazten dira derrigorrezko ezaugarriak.
- **Hautazko ezaugarriak:** aukerako ezaugarriak dira hauek, hau da, produktu batzuetan agertu daitezke eta beste batzuetan ez. [C.1](#) irudiko bigarren kasuan ikusten da nola adierazten dira hautazko ezaugarriak erabilitako nomenklaturan.
- **Or-esklusiboa:** or-esklusibo batean dauden ezaugarri multzoko ezaugarri bakarria agertu daiteke azkeneko produktuan. [C.1](#) irudiko hirugarren kasua da or-esklusibo batena.
- **Or-inklusiboa:** or-inklusibo bateko multzotik ezaugarri bat edo gehiago (guztiak aukeratu daitezke) aukeratu daitezke. Or-inklusibo baten adibidea bat [C.1](#) irudiko laugarren kasua da.



C.1 Irudia: Ezaugarri moten nomenklatura.

D. ERANSKINA

Produktuak lortzeko prozesua

4. kapituluaren aipatu da nola lortzen diren produktuak *pure::variants*-en *Variant Model*-en bitartez, baina, produktuen dependentziak (*packages.json* fitxategian multzokaturik agertzen diren dependentziak) instalatzeko eta aplikazioa Chrome-n integratzeko urratsak azaltzea falta da. Hauek dira jorratu beharreko urratsak:

1. *PowerShell* terminala ireki behar da, eta handik lortu den produktuaren *input* karpeta mugitu.
2. *npm install* komandoa exekutatu behar da. Exekuzio hau nahiko luzea da, 15 minutu inguru iraun dezake.
3. *gulp default* komandoa exekutatu. Hau ere 5 minutu iraun dezake, batez ere produktua exekutatzen den lehen aldian. Aldaketa bakoitzarekin aipatutako komandoa exekutatzea aurrezteko *-watch* aukera erabili daiteke, horrela proiektuaren aldaketak automatikoki ikusten eta exekutatzen ditu amaierako Chrome-ko hedapena zuzenean sortuz.
4. Aplikazioa Chrome-n integratzeko <chrome://extensions>-en garatzaile modua aukeratu behar da (goialdeko ezkerreko *toggle* botoia desaktibatuz).
5. *Deskonprimatuta kargatu* aukeratu. Agertuko den leihoan sortutako produktuaren *input/dist/chrome* aukeratuz aplikazioa Chrome-n integratuko da.

E. ERANSKINA

Bileren aktak

E.1 1. bileraren akta

- **Data:** 2019ko urtarrilaren 22a.
- **Tokia:** Informatika fakultateko 3.2 mintegia.
- **Hasiera ordua:** 15:15.
- **Partaideak:**
 - Raul Medeiros.
 - Haritz Medina.
 - Xabier Garmendia.
 - Arantza Irastorza.
 - Mikel Cantero.
- **Bileraren gaia edo arrazoia:** Gradu Amaierako Lanari hasiera ematea.
- **Bileraren helburuak:** *pure::variants*-en funtzionamendua gaineratik ikustea, Haritz Medinak bere aplikazioak aurkeztea eta kodearen egitura gaineratik azaltzea.
- **Bileraren laburpena:** hasieran partaide bakoitza bere burua aurkeztu du. Ondoren Raul Medeirosek *pure::variants*-en funtzionamendua azaldu du adibide batekin.

Azkenik, Haritz Medinak *Highlight&Go* eta *Review&Go* aplikazioak aurkeztu ditu, bakoitzaren kodearen egitura gaineratik aztertuz.

- **Bileraren ondorioak eta eginbeharrekoak:** *Mark&Go* aurrerago ikustea erabaki da, dena bilera bakar batean ez pilatzeko.

Bi ikasleek kodea aztertzen hasi beharko dira, ondoren, SPL-ren ezaugarri diagrama diseinatzeko eta aplikazioen funtzionamendua ulertzeko.

Ez da bigarren bileraren data zehaztu.

E.2 2. bileraren akta

- **Data:** 2019ko otsailaren 18a.
- **Tokia:** Informatika fakultateko 3.2 mintegia.
- **Hasiera ordua:** 10:30.
- **Partaideak:**
 - Haritz Medina.
 - Xabier Garmendia.
 - Mikel Cantero.
- **Bileraren gaia edo arrazoia:** Falta zen aplikazioaren (*Mark&Go*) azalpena.
- **Bileraren helburuak:** *Mark&Go*-ren funtzionamendua ulertzea eta orain arte sortu diren zalantzak argitzea.
- **Bileraren laburpena:** Lehenengo bileran ikusitakoaren laburpen txiki batekin hasi da bilera. Ondoren, aplikazio bakoitzaren kodea *GitHub*-etik nola lortu eta kode hori nola konpilatu azaldu du Haritz Medinak. Azkenik, *Mark&Go* aplikazioa nola konfiguratu eta bere funtzionamendua aztertu da.
- **Bileraren ondorioak eta eginbeharrekoak:** Ikasleek kodearen funtzionamendua aztertzen hasi beharko dira, horretarako JavaScript erreparatu beharko dute, aplikazioak lengoaia horretan programatuta baitaude.

Ez da hirugarren bileraren data zehaztu.

E.3 3. bileraren akta

- **Data:** 2019ko martxoaren 12a.
- **Tokia:** Informatika fakultateko 3.2 mintegia.
- **Hasiera ordua:** 10:15.
- **Partaideak:**
 - Haritz Medina.
 - Mikel Cantero.
- **Bileraren gaia edo arrazoia:** Kodearen inguruko zalantzak eta lehenengo ezaugarri diagramaren bertsioa erakutsi.
- **Bileraren helburuak:** Aplikazioen kodeen inguruko zalantzak argitzea eta ezaugarri diagrama zuzentzea.
- **Bileraren laburpena:** Hasieran Mikel Canterok kodean ulertzen ez zituen funtzioak aipatu ditu eta banaka Haritz Medinak funtzio hauen helburuak azaldu ditu. Behin kodearen zalantzak argitu direla, Mikel Canterok egindako ezaugarri diagramaren lehenengo bertsioa erakutsi dio Haritz Medinari. Bion artean zuzenketak proposatu dira.
- **Bileraren ondorioak eta eginbeharrekoak:** Mikel Canterok ezaugarri diagrama zuzendu beharko du eta kodearen funtzionamendua hobetu ulertu behar du Google Chrome-ko *debuger*-a erabiliz. *Mark&Go*-ren kodea aztertzen hasi beharko da ere. Laugarren bilera ezaugarri diagramaren bertsio berria egiterakoan egingo da.

E.4 4. bileraren akata

- **Data:** 2019ko martxoaren 26a.
- **Tokia:** Informatika fakultateko 3.2 mintegia.
- **Hasiera ordua:** 10:15.
- **Partaideak:**
 - Haritz Medina.
 - Mikel Cantero.
- **Bileraren gaia edo arrazoia:** Ezaugarri diagramaren eztabaidatzea eta kodearen etiketatze prozesuaren dudak argitzea.
- **Bileraren helburuak:** Ezaugarri diagrama zuzentzea eta etiketatze prozesuari ofizialki hasiera ematea.
- **Bileraren laburpena:** Bilera ezaugarri diagramaren azterketarekin hasi da. Ondoren, Mikel Canterok egindako fitxategi eta ezaugarrien arteko erlazioa aztertu da, baita etiketatze prozesua *pure::variants*-ekin edo *Review&Go* erabiliz egin behar den eztabaidatu da. *pure::variants*-i buruzko duda batzuk ere planteatu ditu Mikel Canterok.
- **Bileraren ondorioak eta eginbeharrekoak:** Mikel Canterok adostutako aldaketak egin beharko ditu ezaugarri diagramari, baita kodearen etiketatze prozesuari hasiera eman beharko dio. Horretarako, zuzenean *pure::variants* erabiltzea erabaki da.
Ez da bosgarren bileraren data finkatu.

E.5 5. bileraren akata

- **Data:** 2019ko apirilaren 11a.
- **Tokia:** Informatika fakultateko 3.2 mintegia.
- **Hasiera ordua:** 10:15.
- **Partaideak:**
 - Haritz Medina.
 - Mikel Cantero.
- **Bileraren gaia edo arrazoia:** Erroreak konpontzeko bilera.
- **Bileraren helburuak:** SPLko produktu bat konpilatzerako garaian sortzen diren erroreak kentzea edo murriztea.
- **Bileraren laburpena:** Mikel Canterok gertatuko erroreak erakutsi ditu eta bion artean erroreak murrizten joan dira.
- **Bileraren ondorioak eta eginbeharrekoak:** Medinak Cantero bideratu du berak erroreak bere kabuz konpontzen jarraitzeko.

Seigarren bilera hurrengo astean egitea pentsatu da, baina ez da eguna finkatu.

E.6 6. bileraren akata

- **Data:** 2019ko apirilaren 15a.
- **Tokia:** Informatika fakultateko Arantza Irastorzaren bulegoan.
- **Hasiera ordua:** 14:30.
- **Partaideak:**
 - Arantza Irastorza.
 - Mikel Cantero.
- **Bileraren gaia edo arrazoia:** Lanarekin edukitako arazoak aurkeztea.
- **Bileraren helburuak:** Arazoei irtenbide bat aurkitzea.
- **Bileraren laburpena:** Mikel Canterok orain arte edukitako arazoak azaldu ditu, iturburu kodearekin izandako arazoak. Arazoa *TextAnnotator.js script*-ean SPL-ko produktuetako bat etengabe bueltaka gelditzen dela da, hau da, amaiera gabeko iterazio batean. Produktua nahiz eta iturburuko aplikazioaren kode berdina eduki, arazoa ez da konpontzen. Arantza Irastorzak arazoaren jatorria aurkitzen saiatu da. Gradu Amaierako Lan honen zailtasunak kontuan edukita, birplangintza bat egitea erabaki da helburu errealagoekin.
- **Bileraren ondorioak eta eginbeharrekoak:** Mikel Canterok birplangintza bat egin beharko du arazoak eta zailtasunak azalduz eta helburu berriak finkatuz.

E.7 7. bileraren akata

- **Data:** 2019ko apirilaren 15a.
- **Tokia:** Informatika fakultateko 3.2 mintegian.
- **Hasiera ordua:** 10:30.
- **Partaideak:**
 - Haritz Medina.
 - Mikel Cantero.
- **Bileraren gaia edo arrazoia:** *TextAnnotator.js script*-an gertatzen zen iterazio infinitua konpontzeko bilera.
- **Bileraren helburuak:** Lehenengo produktuarekin azpimarraketak egin ahal izatea.
- **Bileraren laburpena:** Chrome-eko *debugger*-a erabiliz urratsez urrats exekuzioa begiratu da bion artean eta arazoa ordu bat *debuggeatzen* egon ondoren aurkitu da *Events.js script* barruan. *Events.js* fitxategian falta zen kode lerroa jarri ondoren bileraren helburua lortu da.
- **Bileraren ondorioak eta eginbeharrekoak:** Mikel Canterok inplementazioarekin jarraitu behar du eta Haritz Medinak proposatutako hainbat aldaketa egin ere.

E.8 8. bileraren akata

- **Data:** 2019ko maiatzaren 2a.
- **Tokia:** Informatika fakultateko 3.2 mintegian.
- **Hasiera ordua:** 15:15.
- **Partaideak:**
 - Haritz Medina.
 - Mikel Cantero.
- **Bileraren gaia edo arrazoia:** *Comments* eta *Strengths* ezaugarriak implementatzeko arazoak izan ditu Mikel Canterok, hauek konpontzeko bilera antolatu da.
- **Bileraren helburuak:** Bai bi ezaugarri horien implementazioa aurreratzea edo bukatzea, baita *Canvas* ezaugarria aurreratu ere.
- **Bileraren laburpena:** *Comments* eta *Strengths* ezaugarrien zailtasuna, *sweatalert2* zuzenean erabili beharrean *Alert.js script*-eko *multipleInserts* funtzioa erabili behar dela da. Mikel Canterok gauza asko aurreratu ditu, baina, azkeneko urratsean, *Hypothes.is*-en anotazioak gordetzerakoan arazoak gertatzen ziren. *Chrome*-ko *debugger*-a erabiliz eta Haritz Medinaren laguntzarekin hauek implementatzea lortu da, baita *Canvas* ezaugarria ere.

Azkeneko honekin arazo gehiago sortu dira, bi dokumentu desberdin berdina izango balira bezala interpretatzen zuen *Hypothes.is*-ek. Arazo hau ez da konpontzea lortu, baina beste dokumentuekin ez da arazo hori gertatzen.
- **Bileraren ondorioak eta eginbeharrekoak:** Mikel Canterok egindako aldaketak iturburuko kodean egin beharko ditu.

E.9 9. bileraren akata

- **Data:** 2019ko maiatzaren 14a.
- **Tokia:** Informatika fakultateko 3.2 mintegian.
- **Hasiera ordua:** 10:15.
- **Partaideak:**
 - Haritz Medina.
 - Mikel Cantero.
- **Bileraren gaia edo arrazoia:** Haritz Medinak Google Chrome-n eguneraketaren ondorioz gertatutako arazoak konpontzeko aldaketak aurkezteko antolatu da bilera hau.
- **Bileraren helburuak:** Haritz Medinak egindako aldaketak aurkeztea.
- **Bileraren laburpena:** Haritz Medinak egindako aldaketak aurkeztu dizkio Mikel Canterori. Azkeneko honek zituen bi duda argitu dira ere bilera honetan.
- **Bileraren ondorioak eta eginbeharrekoak:** Mikel Canterok Hihglight&Go-eko funtzionalitateak inplementatu beharko ditu.

E.10 10. bileraren akata

- **Data:** 2019ko ekainaren 5a.
- **Tokia:** Informatika fakultateko Arantza Irastorzaren mintegian.
- **Hasiera ordua:** 11:30.
- **Partaideak:**
 - Arantza Irastorza.
 - Mikel Cantero.
- **Bileraren gaia edo arrazoia:** Bilera inplementazioari itxiera emateko antolatu da.
- **Bileraren helburuak:** Inplementazioari bukaera ematea eta memoriako azkeneko kapituluak planifikatzea eta zuzentzea.
- **Bileraren laburpena:** Bilera memoriaren aholkuekin hasi da, ondoren Mikel Canterok zituen zalantzak argitu dira, eta azkenik produktu bat sortu da garatutako SPL-arekin Arantza Irastorzak ikusteko.
- **Bileraren ondorioak eta eginbeharrekoak:** Mikel Canterok memoria idatzi behar du. Ez da beste bilera baten data zehaztu.

F. ERANSKINA

Egunero sartutako orduen taulak

Jarrain Gradu Amaierako Lana jorrazeko sartutako orduak adieraziko dira. Horretarako hilabete bakoitzeko taula baten irudia erakusten da, taula horretako zutabe bakoitza hilabete horren egun bat izanik eta lerro bakoitza proiektuaren ataza bat izanik.

ATAZAK/EGUNAK	19/01/21	19/01/22	19/01/23	19/01/24	19/01/25	19/01/28	19/01/29	19/01/30	19/01/31
Kudeaketa prozesua	66,75	0	2,5	0	0	0	0	0	1
Bilerak	16,5	2							
Plangintza	35								
Jarraitzea eta kontrola	15,25		0,5						1
Garapen prozesua	247	1,5	0,5	0	0	0	3	2	1
Ikasketa prozesua	18	1,5	0,5	0	0	0	2	1	0
pure::variants	9	1,5	0,5						
JavaScript	4,5								
Chome developer	4,5						2	1	
Aztartze prozesua	54	0	0	0	0	0	0	1	1
Egitura	11						1	1	1
Kodea	43								
Garapen prozesua	175	0	0	0	0	0	0	0	0
Ezaugarri diagrama	20,5								
Etiketatzeko prozesua	105,75								
Murriztapenak	12								
Probak	36,75								
Dokumentazioa	80,5	0	0	0	0	0	0	0	0
Memoria	80,5								
Aurkezpena	0								
GUZTIRA	394,25	1,5	3	0	0	0	0	3	2

F.1 Irudia: Urtarrilean sartutako orduak.

ATAZAK/EGUNAK	19/02/01	19/02/04	19/02/05	19/02/06	19/02/07	19/02/08	19/02/11	19/02/12	19/02/13	19/02/14	19/02/15	19/02/18	19/02/19	19/02/20	19/02/21	19/02/22	19/02/25	19/02/26	19/02/27	19/02/28
Kudeaketa prozesua	0	1	2	2	3	3	4	3	3	2	0	0	0	0	0	0	0	0	0	0
Bilerak	66,75																			
Plangintza	16,5																			
Jarraipena eta kontrola	35	1	2	3	3	3	3	3	2											
15,25																				
Garapen prozesua	247	4	3	0,5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Ikasketa prozesua	18	0	0	0	0	0	0	0	0	0	0	0,5	0	0,5	0	0	0	0	0	0
pure::variants	9																			
JavaScript	4,5											0,5								
Chome developer	4,5																			
Azietze prozesua	54	2	1	0	0	0	0	0	0	0	0	1	2	1	2	2	0	2,5	3	3
Egitura	11	2	1																	
Kodea	43											1	2	1	2	2		2,5	3	3
Garapen prozesua	175	0	2	2	0,5	0	0	0	0	0	0	0	0	0	0	0	0	1,5	0	0
Ezaugarri diagrama	20,5	2	2																	
Etiketatzeko prozesua	105,75																			
Murriztapenak	12																			
Probak	36,75																			
Dokumentazioa	80,5	2	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Memoria	80,5	2	2																	
Aurkezpena	0																			
GUZTIRA	394,25	0	7	7	3,5	3	4	3	3	2	0	1	4	2	1,5	2	5,5	4	4	3,5

F.2 Irudia: Otsailean sartutako orduak.

ATAZAK/EGUNAK	19/03/01	19/03/04	19/03/05	19/03/06	19/03/07	19/03/08	19/03/11	19/03/12	19/03/13	19/03/14	19/03/15	19/03/18	19/03/19	19/03/20	19/03/21	19/03/22	19/03/25	19/03/26	19/03/27	19/03/28	19/03/29	
Kudeaketa prozesua	66,75	0	0	0	0	0	0	1,75	0	0	2,25	0	0	0	0	0,25	0	2,25	0	0	0,25	
Bilerak	16,5						1,5				2							2				
Plangintza	35						2															
Jarraipena eta kontrola	15,25						0,5	0,25			0,25					0,25		0,25			0,25	
Garapen prozesua	247	1,5	3,5	4,5	4	3	0,5	2,5	3,5	5	3	2,5	6	3	6	5,5	4	3	3	3	6	6
Ikasketa prozesua	18	0	0	0	0	0	0	0	0	0	1,5	0	1	0,5	0	0	0	2	3	0	0	0
pure::variants	9												1	0,5					2		3	
JavaScript	4,5																					
Chome developer	4,5										1,5											
Azteritze prozesua	54	1,5	3,5	4,5	4	3	0,5	1,5	3	3	3	0	2	0	0	0	0	0	0	0	0	0
Egitura	11	0,5						2,5														
Kodea	43	1	3,5	4,5	4	3	0,5	1,5	0,5	3	3	2	2									
Garapen prozesua	175	0	0	0	0	0	0	1	0,5	2	0	1	4	2	5,5	5,5	4	3	1	3	6	6
Ezaugarri diagrama	20,5												2	0,5	0,5				1	1		
Etiketatzeko prozesua	105,75							1	0,5	2		1	4	2	5	3	4	3		2	5	5
Murriztapenak	12																					
Probak	36,75																				2	1
Dokumentazioa	80,5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3	1,5	0	0	0
Memoria	80,5																	3	1,5			
Aurkezpena	0																					
GUZTIRA	394,25	1,5	3,5	4,5	4	3	3	2,5	5,25	5	3	4,75	6	3	6	5,5	4,25	6	6,75	6	6	6,25

E.3 Irudia: Martxoan sartutako orduak.

ATAZAK/EGUNAK	19/04/01	19/04/02	19/04/03	19/04/04	19/04/05	19/04/08	19/04/09	19/04/10	19/04/11	19/04/12	19/04/15	19/04/16	19/04/17	19/04/29	19/04/30
Kudeaketa prozesua	66,75														
Bilerak	16,5										1,75	2	0	2,5	0
Plangintza	35								2		1,5	1,5			2,5
Jarraitzea eta kontrola	15,25				0,25				0,25		0,25	0,5			
Garapen prozesua	247	4	6	2,5	3	2,5	4,5	7,5	4,5	4,5	1,5	4	0	3,5	5,5
Ikasketako prozesua	18	0	0	0,5	0	0	0	0	0	0	0	0	0	0	0
pure::variants	9			0,5											
JavaScript	4,5														
Chome developer	4,5														
Aztertze prozesua	54	0	1	0	0	1	0	0	0	0	0	0	0	0	0
Egitura	11	1			1										
Kodea	43														
Garapen prozesua	175	4	5	2,5	1,5	4,5	7,5	4,5	5	6,5	1,5	4	0	3,5	5,5
Ezaugarri diagrama	20,5	1				1			0,5	0,5					0,5
Etiketate prozesua	105,75	3	3,5	2,5	1,5	2,5	1,5	2	2	2,5	1,25	3,5		2,5	4
Murriztapenak	12						2		0,5	0,5					
Probak	36,75	1	0,5			2	3	2,5	2	3	0,25	0,5		1	1
Dokumentazioa	80,5	0	0	2	2	0	0,5	2,5	0	1	4	0	0,5	1	0
Memoria	80,5			2	2		0,5	2,5		1	4		0,5	1	
Aurkezpena	0														
GUZTIRA	394,25	4	6	4,5	5	4,75	4,5	8	7	7,25	7,5	6	0,5	7	5,5

F.4 Irudia: Apirilean sartutako orduak.

ATAZAK/EGUNAK	Guztira	19/05/01	19/05/02	19/05/03	19/05/06	19/05/07	19/05/08	19/05/09	19/05/10	19/05/13	19/05/14	19/05/15	19/05/16	19/05/17	19/05/20	19/05/21	19/05/22	19/05/23	19/05/24	19/05/27	19/05/28	19/05/29	19/05/30	
Kudeaketa prozesua	66,75	0	3	0,5	0,5	0	0	0	0	0	2	0	0	0,5	0	0	0	0	0,5	0	0	0	0	0
Bilerak	16,5		3								2													
Piangintza	35																							
Jarraipena eta kontrola	15,25			0,5	0,5					0,5														
Garapen prozesua	247	0,5	5	1,5	0	4,5	5	2,5	1	3,5	4	6	4	1,5	2,5	5	1,5	3	2	6	5	4	1	1
Ikasketa prozesua	18	0	0	0	0	0	0	1,5	1	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
pure::variants	9																							
JavaScript	4,5							1,5	1													1		
Chome developer	4,5																							
Azteritze prozesua	54	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Egitura	11																							
Kodea	43																							
Garapen prozesua	175	0,5	5	1,5	0	4,5	5	1	0	3,5	4	6	4	1,5	2,5	5	1,5	3	2	6	5	3	1	1
Ezaugarri diagrama	20,5																							
Etiketaze prozesua	105,75	0,5	4	1		4	4	1		3	3	4	3	1	2	4,5	1	2	1,5	3	0,5	2	1	1
Murriztapenak	12																							
Probak	36,75		1	0,5		0,5	1			0,5	1	2	1	0,5	0,5	0,5	1	0,5	1	0,5	3	3	1	1
Dokumentazioa	80,5	3,5	0	2,5	2,5	1	0	1	0	0	0	0	0	2,5	1	1	2	2	1	0	0	1	2	2
Memoria	80,5	3,5		2,5	2,5	1								2,5	1	1	2	2	1			1	2	2
Aurkezpena	0																							
GUZTIRA	394,25	4	8	4,5	3	5,5	5	3,5	1	3,5	6	6	4	4,5	3,5	6	3,5	5	3,5	6	5	5	3	3

F5 Irudia: Maiatzean sartutako orduak.

ATAZAK/EGUNAK	19/05/31	19/06/03	19/06/04	19/06/05	19/06/06	19/06/07	19/06/10	19/06/11	19/06/12	19/06/13	19/06/14	19/06/17	19/06/18	19/06/19	19/06/20	19/06/21
Kudeaketa prozesua	66,75	0	0	1	0	2	1,5	0	0	0	0	0	0	0	0	0
Bilerak	16,5			1												
Plangintza	35															
Jarrajpena eta kontrola	15,25	0,25				2	1,5									
Garapen prozesua	247	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Ikasketa prozesua	18	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
pure::variants	9															
JavaScript	4,5															
Chome developer	4,5															
Aztertze prozesua	54	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Egitura	11															
Kodea	43															
Garapen prozesua	175	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Ezaugarri diagrama	20,5															
Etiketatzeko prozesua	105,75															
Murriztapenak	12	1														
Probak	36,75															
Dokumentazioa	80,5	2	4,5	6	4	3	1	2	3	2	2	0	0	0	0	0
Memoria	80,5	2	4,5	6	4	3	1	2	3	2	2					
Aurkezpena	0															
GUZTIRA	394,25	3,25	4,5	6	4	5	2,5	2	3	2	2	0	0	0	0	0

F.6 Irudia: Ekainean sartutako orduak.

Bibliografia

- [BUT4Reuse] BUT4REUSE: *BUT4Reuse installation*. – URL <https://github.com/but4reuse/but4reuse/wiki/Installation>
- [Medina 2018] MEDINA, Haritz: *Highlight&Go*. sep 2018. – URL <https://chrome.google.com/webstore/detail/highlightgo/bihmalipgnlomidlpekdnoohiejppfmo?hl=es>
- [Medina 2019a] MEDINA, Haritz: *Mark&Go*. mar 2019. – URL <https://chrome.google.com/webstore/detail/markgo/kjedcndgienemldgjpjjnhjdhfoacfa?hl=es>
- [Medina 2019b] MEDINA, Haritz: Scalability & Quality Tradeoffs in Student Feedback: Web Annotations to the rescue. In: *ITICSE* (2019)
- [Medina u. a. 2019a] MEDINA, Haritz ; DIAZ, Oscar ; ANFURRUTIA, Felipe I.: Coding-Data Portability in Systematic Literature Reviews: a W3C's Open Annotation Approach. In: *EASE* (2019)
- [Medina u. a. 2019b] MEDINA, Haritz ; DIAZ, Oscar ; CONTELL, Jeremias P.: Performant Peer Review for Design Science Manuscripts: A Pilot Study on Dedicated Highlighters. In: *ONEKIN* (2019)
- [pure::variants] PURE::VARIANTS: *pure::variants*. – URL <https://www.pure-systems.com/products/pure-variants-9.html>
- [SourceGear 2017] SOURCEGEAR: *DiffMerge*. 2017. – URL <http://www.sourcegear.com/diffmerge/>
- [Wikipedia 2015] WIKIPEDIA: *Software produktu-lerro*. apr 2015. – URL https://eu.wikipedia.org/wiki/Software_producto-lerro

[Wikipedia 2018a] WIKIPEDIA: *A.nnotate*. jan 2018. – URL <https://en.wikipedia.org/wiki/A.nnotate>

[Wikipedia 2018b] WIKIPEDIA: *Hypothes.is*. oct 2018. – URL <https://en.wikipedia.org/wiki/Hypothes.is>

[Wikipedia 2019a] WIKIPEDIA: *Node.js*. mar 2019. – URL <https://eu.wikipedia.org/wiki/Node.js>

[Wikipedia 2019b] WIKIPEDIA: *Web annotation*. mar 2019. – URL https://en.wikipedia.org/wiki/Web_annotation