

Ikasturtea:	2018/2019
Gaia:	Software Ingeniaritza
Ariketak:	Patroiak
Data:	2019ko Otsaila

1. Kotxeko nabigatzailea (2018ko Uztaila)

TAMTOMTUM enpresak bere kotxeko nabigatzaileen softwarea garatzeko kontratatu gaitu. Softwareak mapak kudeatu behar ditu. Adibidez, abiapuntu eta helmuga baten arteko ibilbidea kalkulatzeko aukera emango digu, ibilbidearen GPS puntuez osatutako String kolekzio bat bueltatzen delarik. Enpresak jada badu funtzionalitate hau inplementatzen duen ibilbideak kudeatzeko modulu bat. Kasu honetan, *ibilbidea_kalkulatu(String pAbiapuntua, String pHelmuga)* funtzioa erabiliz egiten da, baina merkatuan dauden beste modulu batzuk erabili nahi dira, Google Maps edo OpenStreetMaps adibidez. Hala ere, kasu hauetan ibilbidea kalkulatzeko era aldatzen da, lehenengo kasuan *calculate_route(String origin, String destination)* egiten delarik eta *osm_calc_path(String o, String d)* bigarreanean. Sistemarekin lana era homogeenoa batetan egin nahi dela eta etorkizunean ibilbideak kalkulatzeko era berriak gehitu nahi direla kontutan izanik,

Eskatzen da:

- Labur arrazoitu zein patroi erabiliko zenukeen eta zergatik ibilbidearen kalkulua diseinatzeko.
- Klase diagrama zehaztu, klase nagusiekin eta garrantzitsuak iruditzen zaizkizun metodoekin.
- Klase horien inplementazioa idatzi.

2. Hotel kateen ordainketa kudeaketa (2018ko Uztaila)

MOLAHOTELS hotel kateak, Europa, Asia eta Amerikan dozenaka hotelekin, bere hotelak kudeatzeko softwarea garatzeko kontratatu gaitu. Kontinente bakoitzeko hotelek ezaugarri bereziak dituztela kontutan hartu behar da. Sprint honetan egonaldiaren ordainketa egingo duen modulua garatzen ari gara. Zehazki gure taldea kreditu txartel bidezko ordainketa funtzionalitatea egiten ari da. European ordainketa EuropeBank-ekin kudeatzen da, Asia eta Amerikan AABank-ekin egiten delarik. Bi kasuetan ordainketa egiteko bezeroaren izena, kreditu txartel zenbakia eta zenbatekoa bidali behar dira, baina banku bakoitzarentzat era desberdinean exekutitzen da. Etorkizunean katea beste kontinenteetara zabaltzeko nahi dela eta zein bankurekin lan egiten duen aldatzea erabaki dezakeela kontutan hartuz,

Eskatzen da:

- Labur arrazoitu zein patroi erabiliko zenukeen eta zergatik hoteletako kreditu txartel bidezko ordainketa diseinatzeko.
- Klase diagrama zehaztu, klase nagusiekin eta garrantzitsuak iruditzen zaizkizun metodoekin.
- Klase horien inplementazioa idatzi.

Ikasturtea:	2018/2019
Gaia:	Software Ingeniaritza
Ariketak:	Patroiak
Data:	2019ko Otsaila

3. Dragamina (2018ko Maiatza)



1. Irudia: Dragamina

Lan egiten dugun enpresan, bezero batek “Dragamina” joko klasikoa inplementatzea eskatu digu (1. Irudia). Sprint honetan tablero eta gelaxka kudeaketarekin zerikusia duten erabiltzaile istorioak inplementatzen ari gara. Espezifiki, orain erabiltzaileak saguarekin gelaxka batetan klik egiteari negozio logikak nola erantzuten dion lantzen ari gara. Erabiltzaileak itxita dagoen gelaxka batetan eskuineko botoiarekin klik egiten badu, gelaxkaren gainean bandera bat agertuko da. Bandera duen gelaxka batetan klik eginez, bandera kenduko da. Irekita dagoen gelaxka batetan klik egiteak ez du ezer egingo, ez eskuineko ez ezkerreko botoiekin.

Eskatzen da:

- Labur arrazoitu zein patroi erabiliko zenukeen eta zergatik eskubiko botoiaren erantzuna tablero eta gelaxketan inplementatzeko.
- Klase diagrama zehaztu, klase nagusiekin eta garrantzitsuak iruditzen zaizkizun metodoekin.
- Klase horien inplementazioa idatzi.

4. Eraikin adimenduen kudeaketa (2018ko Maiatza)

ULTRAKORP enpresak, Berlin, Brasil eta Hong Kong-en egoitzekin, eraikin adimenduetan inbertsio haundi bat egin du bere langileek lan ingurune eroso bat izan dezaten. Hiru eraikinek funtzio desberdinak eskaintzen dituzte, adibidez, nahi dugun tenperatura aukeratu, lan orduak zehaztu, lanegunaren hasieran argiak automatikoki piztu daitezzen eta itzali bukatzen denean, aire berriztapen erritmoa zehaztu, beti aire garbia dagoela bermatzeko, etab. Logikoa den bezala tenperaturaren kontrola eraikin

Ikasturtea:	2018/2019
Gaia:	Software Ingeniaritza
Ariketak:	Patroiak
Data:	2019ko Otsaila

edo baita gela bakoitzeko espezifikoa izan beharko den arren, beste funtzio asko, adibidez argiak piztu eta itzaltzeko ordutegia edota aire berriztapen erritmoa, komunak izango dira eta era zentralizatuan kudeatzeko aukera egon beharko litzateke, komunak dira eta. ULTRAKORP enpresak kudeaketa zentralizatu hau diseinatzeko kontratatu gaitu. Kontutan hartu behar da eraikin bakoitzeko softwarea enpresa desberdin batek sortu duela, aurretik aipatutako funtzioak egiteko era egoitza batetik bestera desberdina izan daitekeelarik.

Eskatzen da:

- a) Labur arrazoitu zein patroi erabiliko zenukeen eta zergatik eraikin guztien funtzio komunak kudeaketa zentralizatu inplementatzeko.
- b) Klase diagrama zehaztu, klase nagusiekin eta garrantzitsuak iruditzen zaizkizun metodoekin.
- c) Klase horien inplementazioa idatzi.

5. Ariketa: Instagram (2018ko Martxoa)

Instagram moduko aplikazio bat garatzen ari gara, bai interfaze grafiko batekin erabiltzaileak, bai programa bidez metodoei deituz erabili daitekeena. Aplikazioak filtro desberdinak sortzeko aukera emango digu. Filtro bakoitzak argazkiaren histograma, kontrastea, distira eta urdin/gorri/berde oreka aldatuko ditu. Filtro bakoitzak zenbaki konkretu bat izango du histograma, kontraste eta distirarentzat eta hiru zenbaki, bat kolore bakoitzeko, urdin/gorri/berde orekarentzat.

Histograma, kontrastea, distira eta urdin/gorri oreka aldatzeko era android bertsio bakoitzean aldatu daiteke. Hau kontutan izanda eta aplikazioak bertsioz aldatzea eta hedatzea erraz baimendu behar duela, eskatzen da:

- a) Labur arrazoitu zein patroi erabiliko zenukeen eta zergatik eskatutakoa inplementatzeko.
- b) Aplikazioaren klase diagrama zehaztu, klase nagusiekin eta garrantzitsuak iruditzen zaizkizun metodoekin.
- c) Klase horien inplementazioa idatzi.

6. Ariketa: eBay (2018ko Martxoa)

eBay moduko aplikazio bat garatzen ari gara. Bertan, erabiltzaileak produktu desberdinetan eskaintzak egin ditzake. Produktu batetan berea baino altuagoa den eskaintza bat egiten denean, sistemak erabiltzaileari abisatuko dio, bere eskaintza igo ahal dezan. Sistemak ere abisatuko dio produktuaren salmenta burutzeko bost minutu gelditzen direnean. Erabiltzaile bat edozein momentutan produktu salmentatik erretiratu daiteke, produktuaren egoeren berri izateari utziz. Eskatzen da:

- a) Labur arrazoitu zein patroi erabiliko zenukeen eta zergatik eskatutakoa inplementatzeko.
- b) Klase diagrama zehaztu, klase nagusiekin eta garrantzitsuak iruditzen zaizkizun metodoekin.

Ikasturtea:	2018/2019
Gaia:	Software Ingeniaritza
Ariketak:	Patroiak
Data:	2019ko Otsaila

- c) Klase horien inplementazioa idatzi.

7. Ariketa: Fitxategien enkriptazioa (2018ko Martxoa)

Banku baten softwarea garatzen ari gara. Software honek modulu desberdinak izango ditu, kontu korronteen kudeaketa, maileguak, bezeroen kudeaketa, etab. Modulu askok fitxategiak enkriptatzeko beharra dute, datu babeserako legediarekin bete ahal izateko. Enkriptaziorako algoritmo desberdinak erabili daitezke, AES edo 3DES esaterako. Modulu bakoitzak, bere beharren arabera, algoritmo bat edo beste erabiliko du. Moduluak eta algoritmoak etorkizunean gehitu eta aldatu daitezkeela jakinda, eskatzen da:

- Labur arrazoitu zein patroi erabiliko zenukeen eta zergatik eskatutakoa inplementatzeko.
- Klase diagrama zehaztu, klase nagusiekin eta garrantzitsuak iruditzen zaizkizun metodoekin.
- Klaseen horien inplementazioa idatzi.

8. Bideojokoa (2017ko Ekaina)

Kontsolarako bideojoko bat garatzen ari gara. Hasiera puntua *World of Warcraft* antzeko zerbait da, non arraza ezberdinetako pertsonaiek zailtasun maila ezberdinetako misioak aurrera eramaten dituzten. Pertsonaia bakoitzak bere izena, sorrera data eta dituen puntu kopurua gordeko ditu. Sprint honetan kontsolako gurutze botoien erantzuna garatzen ari gara:

- Gora botoiari* ematean, sorgin eta iratxoak hegan hasiko dira. Beste arraza guztiek salto bate mango dute.
- Eskubira botoiarekin* mago eta sorginek aztikeria bat botako dute, sendalariak babes ezkutu bat eta elfoek gezi bat jaurtiko dute. Beste arrazek norabide horretan korrika joango dira etsaiaren bila.
- Behera botoiarekin* satorrak tunel bat zulatzen hasiko dira. Beste arrazak makurtu egingo dira.
- Atzera botoiari* ematean elfo eta gizonek mortal bat egingo dute atzera, beste arrazek norabide horretan ihes egingo dutelarik.

Eskatzen da:

- Labur arrazoitu zein patroi erabiliko zenukeen eta zergatik eskatutakoa garatzeko.
- Klase diagrama zehaztu sistemaren diseinua egin ahal izateko, klase nagusiekin eta garrantzitsuak iruditzen zaizkizun metodoekin.
- Klase horien inplementazioa idatzi.

Ikasturtea:	2018/2019
Gaia:	Software Ingeniaritza
Ariketak:	Patroiak
Data:	2019ko Otsaila

9. Fitness tracker (2017ko Ekaina)

Fitbit moduko pultsera bat garatzeko kontratatu gaitezke. Pultsera honek erabiltzailearen forma fisikoa neurtuko du. Sprint honetan gomendioen zatia garatzen ari gara. Dietari dagokionez, gure IMC-a (pisua zati altuera ber bi eginez kalkulatzeko dena) 18,5 baino txikiagoa bada, dieta hiperkaloriko bat gomendatuko digu, 18,5 eta 24,55 artean badago zori onduko gaitu eta 25 baino haundiagoa bada dieta hipokaloriko bat gomendatuko digu. Kirolari dagokionez, gure forma baxua dela adierazten badiogu, ariketa plangintza lasai bat prestatuko digu, normala bada maila horretako bat eta altua bada kirolari onentzat bat. Hau aurrera eraman ahal izateko, sistemak pisua, altuera eta forma maila sartzen utziko du.

Eskatzen da:

- Labur arrazoitu zein patroi erabiliko zenukeen eta zergatik eskatutakoa garatzeko.
- Klase diagrama zehaztu sistemaren diseinua egin ahal izateko, klase nagusiekin eta garrantzitsuak iruditzen zaizkizun metodoekin.
- Klase horien inplementazioa idatzi.

10. Smartwatch-a (2017ko Maiatza)

Smartwatch bat kontrolatzen softwarea garatzen ari gara (*Apple Watch* edo *Motorola 360*-aren antzeko gailu bat). Gailu honek funtzio ezberdinak izango ditu: erlojua, pulsometroa, podometroa, loaren kalitatea neurtzeko sensorea eta altitude sensorea zenbat pisu igo diren neurtzeko Gainera, telefono mugikorraren aplikazio ezberdinen notifikazioak jasoko ditu: dei bat edo whatsapp bat jasotzen duen bakoitzean edo egutegiaren notifikazio bat aktibatzen den bakoitzean adibidez.

Eskatzen da:

- Labur arrazoitu zein patroi erabiliko zenukeen eta zergatik *smartwatch*-aren funtzionamendua inplementatzeko.
- Klase diagrama zehaztu, klase nagusiekin eta garrantzitsuak iruditzen zaizkizun metodoekin.
- Klase horien inplementazioa idatzi.

11. Pilotu Automatikoa (2017ko Maiatza)

Boeing enpresak bere hegazkinen piloto automatikoa inplementatzen duen software modulu bat garatzeko kontratatu gaitu. Modulu honen funtzionamendua honakoa da: behin aktibatuta, modulu honek hegazkinaren gurutzaldi-abiadura eta altuera finkatzeaz arduratuko da, *flap*-ak aktibatu eta desaktibatzeaz eta presioa doitzeaz beharrezkoa izanez gero. Kontutan izan behar da funtzio hauetako bakoitza software modulu batek kontrolatzen duela eta Boeing hegazkin eredu bakoitzak desberdinak izan daitezkeela daukaten hardware-aren arabera (normala denez hegazkin berrienek software modulu azkar eta eraginkorrenak izango dituzte). Proposatutako diseinuak

Ikasturtea:	2018/2019
Gaia:	Software Ingeniaritza
Ariketak:	Patroiak
Data:	2019ko Otsaila

berrerabilpena bermatu behar duela eta software moduluen aldaketen aurrean sendoa izan behar dela kontutan hartuz,

Eskatzen da:

- Labur arrazoitu zein patroia erabiliko zenukeen eta zergatik sistemaren funtzionamendua inplementatzeko.
- Klase diagrama zehaztu, klase nagusiekin eta garrantzitsuak iruditzen zaizkizun metodoekin.
- Klase horien inplementazioa idatzi.

12. Ariketa: Scrum (2017ko Martxoa)

Klasean ikusi dugun bezala, Scrum metodologia talde txikitan aplikatzeko pentsatu zen hasiera batean. Hala ere, bere arrakasta dela eta, talde handiagotan aplikatu ahal izateko hedapenak bilatu dira. Hau aurrera eraman ahal izateko, behar adina Scrum talde definitzen dira, bakoitzak 4/6 pertsona inguru dituelarik. Hauetako 4/6 talde bakoitzeko super-talde bat definitzen da, talde horiek kudeatuko dituenak. Talde gehiago sartu nahi izanez gero, 4/6 bakoitzeko super-talde desberdin bat sortuko da, eta ondoren goiko mailan beste super-talde bat definituko da azpiko super-taldeak kudeatzen dituenak. Horrela, hierarkian igoz, milaka pertsonako taldeak Scrum bidez kudeatzera iritsi gaitezke. Zeregin bat esleitzeko garaian, taldeari edo pertsonari esleitzea dago. Zeregin batzuk (dokumentatu, adibidez), hierarkiako talde eta pertsona guztiei esleitu beharko zaizkie. Daukagun garapen taldearen tamainaren arabera hierarkiak maila desberdinak izan ditzakeela jakinda, eskatzen da:

- Labur arrazoitu zein patroia erabiliko zenukeen eta zergatik eskatutakoa inplementatzeko.
- Klase diagrama zehaztu Scrum garapen taldearen kudeaketa egin ahal izateko, klase nagusiekin eta garrantzitsuak iruditzen zaizkizun metodoekin.
- Klase horien inplementazioa idatzi.

13. Ariketa: Web Aplikazioak (2017ko Martxoa)

Web aplikazioak sortzen laguntzeko sistema bat garatzen ari gara. Bertan mota desberdinetako aplikazioak sortzeko aukera egongo da. Denak denda modukoak izango dira, zerbait saltzeko aukera emango dutelarik: liburuak, arropa, hotel gauak, ikuskizunak... Bezeroak bilaketa bat egiten duenean, emaitzak irizpide desberdinen arabera ordenatzeko aukera izango du: prezioa, sistemak gomendatuak, bezeroen balorazioak, etab. Denda mota bakoitzak interesatzen zaizkion irizpideak aukeratu ditzakela jakinda, eskatzen da:

- Labur arrazoitu zein patroia erabiliko zenukeen eta zergatik eskatutakoa inplementatzeko.
- Klase diagrama zehaztu sistemak denda motak garatu ahal izateko, klase nagusiekin eta garrantzitsuak iruditzen zaizkizun metodoekin.
- Klase horien inplementazioa idatzi.

Ikasturtea:	2018/2019
Gaia:	Software Ingeniaritza
Ariketak:	Patroiak
Data:	2019ko Otsaila

14. Ariketa: Bankua (2017ko Martxoa)

Banku bateko sistema informatikoa diseinatu eta inplementatzeko eskatu digute. Scrum bidez garatzen ari gara eta sprint honetan kontu korronteak kudeatzen dituen modulua garatzen ari gara. Kontu korronte normal batek diru sarrerak egiteko garaian ez du komisiorik baina transferentzia bat egiteak %1eko komisioa du. Bezero batek kontuan 100.000 € edo gehiagoko saldoa badu transferentzien komisioak barkatuko zaizkio. Aldiz, kontua zenbaki gorrietan badago, transferentzien komisioa %2ra igoko da eta diru sarrerak egiteagatik %1 kobratuko zaio. Interesei dagokionez, positiboan dauden kontuek %0,5eko interesa emango dute urtero. Zenbaki gorrietan dauden kontuek, zenbaki bera emango dute, baina negatiboan, hau da, %0,5 kenduko zaie. Kontu korronteen funtzionamendua ezagututa, eskatzen da:

- Labur arrazoitu zein patroi erabiliko zenukeen eta zergatik eskatutakoa inplementatzeko.
- Klase diagrama zehaztu kontu korrontearen kudeaketa egiteko klase nagusiekin eta garrantzitsuak iruditzen zaizkizun metodoekin.
- Klaseen horien inplementazioa idatzi.

15. Ariketa (2016ko Ekaina)

Bideojoko baten pertsonaien kudeaketa inplementatzea eskatu digute. Joko honetan hasieran jokalaria zein avatar mota erabili nahi duen aukeratu ahal izango du hiru aukeren artean: magoa, elfoa edo ipotxa. Avatar mota bakoitzak suntsipen ezaugarri desberdinak dituen arma bat du aurkitzen dituen oztopoak deuseztatu ahal izateko eta bere bizi puntuak era desberdinean berreskuratuko ditu. Behin avatar mota aukeratu duenean, jokalaria izena, generoa eta begi eta ile kolorea adieraziz pertsonalizatu ahal izango du. Jokoak avatar mota berriak era errazean gehitu ahal izateko diseinatuta egon behar da.

Eskatzen da:

- Labur arrazoitu zein patroi erabiliko zenukeen eta zergatik pertsonaien kudeaketa inplementatzeko, etorkizunean avatar mota berriak gehitzea eskatu diezazkigutela kontutan hartuz.
- Klase diagrama zehaztu, klase nagusiekin eta garrantzitsuak iruditzen zaizkizun metodoekin.
- Klase horien inplementazioa idatzi.

16. Ariketa (2016ko Ekaina)

Denda elektronikoak kudeatzeko software bat sortzen ari gara. Konkreteriki erosketa orga kontrolatzen duen zatia diseinatu ari gara. Orga honek ordainketa egiteko metodo bat izango du. Aplikazioak hiru ordainketa modu baimenduko ditu: Paypal, kreditu txartela eta transferentzia. Denda bakoitzak ordainketa modu bakarra ezarriko du bertan egindako erosketentzat. Modu hau aldatzeko aukera egongo da, baina dendako bezero guztientzat berdina izango da. Orga software hau erabiliko duten 20 dendentzat espezializatuko da, bakoitzarentzat portaera espezifikoak baimenduz.

Ikasturtea:	2018/2019
Gaia:	Software Ingeniaritza
Ariketak:	Patroiak
Data:	2019ko Otsaila

Eskatzen da:

- Labur arrazoitu zein patroi erabiliko zenukeen eta zergatik sistemaren funtzionamendua inplementatzeko, etorkizunean ordainketa modu berriak gehitu daitezkeela kontutan hartuz.
- Klase diagrama zehaztu, klase nagusiekin eta garrantzitsuak iruditzen zaizkizun metodoekin.
- Klase horien inplementazioa idatzi.

17. Ariketa (2016ko Maiatza)

Gure enpresa mp3 single baten softwarea garatzen ari da (*ipod shuffle*-aren antzeko zerbait). Reproductore honek 4 botoi bakarrik ditu: piztu/itzali, play/pause, abestia saltatu (>>) eta abestia berriz hasieratik hasi (<<). Azken bi botoi hauek musika dagoenean bakarrik funtzionatzen dutela kontutan hartu behar da.

Eskatzen da:

- Labur arrazoitu zein patroi erabiliko zenukeen eta zergatik lau botoien funtzionamendua inplementatzeko.
- Klase diagrama zehaztu, klase nagusiekin eta garrantzitsuak iruditzen zaizkizun metodoekin.
- Klase horien inplementazioa idatzi.

18. Ariketa (2016ko Maiatza)

Sukaldeko robot desberdinak fabrikatzen dituen enpresa batetan lan egiten dugu. Robot bakoitzak osagai desberdinak ditu: marraza kontroladorea, tenperaturarena eta pauso baten denbora kontrolatzen duena. Errezeta desberdinak dituen modulu bat programatu bat da, bakoitzak robotean egin behar diren pausu zerrenda bat dituelarik. Pausu bakoitzak marraza biraketa abiadura, tenperatura eta denbora konkretu bat ditu. Modulua robot eredu guztiekin erabili nahi dugu, ereduaren arteko osagai aldaketak kontutan hartu gabe.

Eskatzen da:

- Labur arrazoitu zein patroi erabiliko zenukeen eta zergatik sistemaren funtzionamendua inplementatzeko.
- Klase diagrama zehaztu, klase nagusiekin eta garrantzitsuak iruditzen zaizkizun metodoekin.
- Klase horien inplementazioa idatzi.

19. Ariketa (2016ko Martxoa)

Robot batek osagai desberdinak ditu (gurpilak, besoa, sentsoreak, etab.), zati bakoitzak bere kontroladorea duelarik. Robota zeregin sinpleak egiteko gai izatea nahi dugu. Adibidez:

Ikasturtea:	2018/2019
Gaia:	Software Ingeniaritza
Ariketak:	Patroiak
Data:	2019ko Otsaila

- *Pilota urdina jo:* Aurrera zuzen joan 50 zentimetro baino gutxiagora elementu bat aurkitu arte. Elementuaren kolorea konprobatu eta urdina bada besoarekin jo.

Hau egin ahal izateko, osagai desberdinen kontroladoreak erabiliko dira:

- Gorpilen kontroladoreak gorpilen errotazio bat aurrera egiteko aukera ematen du.
- Distantzia kontroladoreak objektu bat 50 zentimetrora baino gutxiagora dagoen esango digu.
- Besoaren kontroladoreak besoa mugitzeko aukera ematen du.
- Kolore sentsoarearen kontroladoreak aurrean dugun pilotaren kolorea emango digu.

Etorkizunean operazio gehiago definitu daitezkeela jakinda edo daudenak aldatu (besoa luzatzen badugu robota 70 zentimetrotara gelditu adibidez), eskatzen da:

- a) Labur arrazoitu zein patroia erabiliko zenukeen eta zergatik eskatutakoa inplementatzeko.
- b) Klase diagrama zehaztu robotaren kudeaketa egin ahal izateko, klase nagusiekin eta garrantzitsuak iruditzen zaizkizun metodoekin.
- c) Klase horien inplementazioa idatzi.

20. Ariketa (2016ko Martxoa)

Web-aren bidez erabili daitekeen testu editore oso sinple bat inplementatzeko eskatu digute (*Google Docs* antzeko zerbait). Web-ean sartzean “*Dokumentu berria*” botoia agertuko zaigu. Bertan klik egitean dokumentua azalduko da eta operazio ezberdinak egiteko aukera izango dugu (idatzi, gorde eta itxi). Gauza da editore hau nabigatzailearen arabera desberdina izango dela (momentuz *Chrome* edo *Firefox*). Hau da, operazio hauek desberdin inplementatuko dira nabigatzaile bakoitzean. Etorkizunean nabigatzaile gehiago gehitu daitezkeela kontutan hartuz, eskatzen da:

- a) Labur arrazoitu zein patroia erabiliko zenukeen eta zergatik eskatutakoa inplementatzeko.
- b) Klase diagrama zehaztu sistemak dokumentuen kudeaketa egin ahal izateko klase nagusiekin eta garrantzitsuak iruditzen zaizkizun metodoekin.
- c) Klase horien inplementazioa idatzi.

21. Ariketa (2016ko Martxoa)

Autoeskola batetik gidatzen ikasteko simuladore bat egiteko eskatu digute. Simuladoreak kotxearen elementu nagusiak ditu: bolantea, martxak, pedalak, etab. Sprint honetan azeleradore eta freno pedalen funtzionamendua inplementatuko dugu. Bat eta bost martxen arteko bat sartuta dagoenean azeleradorea zapaltzean pantailako kotxeak aurrera egingo du eta freno zapaltzean moteldu egingo da. Atzerako martxa sartuta dagoenean aldiz, azeleradorea zapaltzean kotxeak atzera egingo du eta freno zapaltzean atzerako ibilera motelduko da. Kotxea itzalita edo lokan (“punto muerto”)

Ikasturtea:	2018/2019
Gaia:	Software Ingeniaritza
Ariketak:	Patroiak
Data:	2019ko Otsaila

badago, pedalak zapaltzean ez da mugituko. Sprint honetako erabiltzaile istorio hauek ezagututa, eskatzen da:

- Labur arrazoitu zein patroi erabiliko zenukeen eta zergatik eskatutakoa inplementatzeko.
- Klase diagrama zehaztu sistemak pedalen kudeaketa egin ahal izateko klase nagusiekin eta garrantzitsuak iruditzen zaizkizun metodoekin.
- Klaseen horien inplementazioa idatzi.

22. Ariketa (2015ko Ekaina)

Psikologiako Fakultateko ikerlari talde batek populazioaren atentzio zuzendua neurtzen duen joko bat garatzeko laguntza eskatu digu. Aplikazioak honako funtzionamendua izatea eskatzen digute:

- Jokoa hasi baino lehen erabiltzaileari bere adina eta ama hizkuntza eskatuko zaizkio. Onartutako hizkuntzak momentuz euskara eta gaztelania izango dira, baina etorkizunean hizkuntza gehiagotara zabaltzea espero da.
- Ondoren, bere ezaugarrien arabera joko espezifikoak sortuko da. 18 urtetik beherako erabiltzaileek piktogrametan oinarritutako joko batetara jolastuko dute, helduek estrategia joko batetara jolasten duten bitartean. Bi jokoek funtzionamendu desberdina izango dute.
- Jokoa bukaeran, adingabekoek zorientzen dituen animazio bat ikusiko dute, helduek aurretik aukeratutako hizkuntzan galdetegi txiki bat erantzuten duten bitartean.

Eskatzen da:

- Zein patroi erabiliko zen(it)u(z)ke esandako softwarea garatzeko? Erantzuna arrazoitu.
- UML klase diagrama zehaztu klase nagusiekin eta garrantzitsuak iruditzen zaizkizun metodoekin.
- Aurreko klaseen inplementazioa idatzi.

23. Ariketa (2015ko Ekaina)

Fabrika automatizatu baten softwarea garatzea eskatzen digute. Espero den funtzionamendua honakoa da:

- Martxan jartzen denean softwareak argiak piztuko ditu, hiru produkzio lerroak aktibatuko ditu, biltegitik beharrezko materiala aterako du eta, fabrikazio plantak tenperatura asko igoarazten duenez, bulegoen aire egokitua piztuko du.
- Matxura elektriko bat egonez gero, sistemak argi guztiak itzaliko ditu eta larrialdikoak piztuko ditu, aire egokitua eta biltegi modulua itzaliko ditu eta hiru lerroak larrialdi moduan jarriko ditu (hau da, larrialdietarako sorgailu batek emango die energia).
- Produkzioa gelditu behar izanez gero, sistemak lerroak geldituko ditu eta biltegia, argiak eta aire egokitua itzaliko ditu.

Eskatzen da:

- Zein patroi erabiliko zen(it)u(z)ke esandako softwarea garatzeko? Erantzuna arrazoitu.

Ikasturtea:	2018/2019
Gaia:	Software Ingeniaritza
Ariketak:	Patroiak
Data:	2019ko Otsaila

- b) UML klase diagrama zehaztu klase nagusiekin eta garrantzitsuak iruditzen zaizkizun metodoekin.
- c) Aurreko klaseen inplementazioa idatzi.

24. Ariketa (2015ko Maiatza)

Musika-kateak egiten dituen enpresa batean lanean gabiltza eta urrutiko aginteen softwarea garatzea eskatu digute. Aginteek ondorengo botoiak dituzte: *power*, *dvd*, *usb*, *irratia* eta *bolumena* (+ eta - botoiak). Botoi hauetan *sakatu* egitean espero den portaera ondorengoa da:

- *power*: Katea itzalita badago piztu egiten du eta piztuta badago itzali egiten du.
- *dvd*: Katea piztuta badago *dvd* moduan jartzen du. Itzalita badago ez du ezer egiten.
- *usb*: Katea piztuta badago *usb* moduan jartzen du. Itzalita badago ez du ezer egiten.
- *irratia*: Katea piztuta badago *irratia* moduan jartzen du. Itzalita badago ez du ezer egiten.
- *bolumena* (+ eta - botoiak): Katea piztuta badago aurreko edozein moduren bolumena kontrolatzen dute. Itzalita badago ez dute ezer egiten.

Eskatzen da:

- a) ¿Zein patroi erabiliko zen(it)u(z)ke esandako softwarea garatzeko? Erantzuna arrazoitu.
- b) UML klase diagrama zehaztu klase nagusiekin eta garrantzitsuak iruditzen zaizkizun metodoekin.
- c) Aurreko klaseen inplementazioa idatzi.

25. Ariketa (2015ko Maiatza)

Utilitateak izeneko klase bat dugu. Bere *filtratu()* metodoak emandako irizpide bat betetzen ez duten elementuak array batetik kentzen ditu. Metodo honek parametro bezala jasotzen duen array-aren elementuek **Filtrable** interfazea inplementatzen dutela suposatzen du. Filtrable interfazeko *irizpideaBetetzenDu()* metodoak objektu batek ematen diogun irizpide bat betetzen duen edo ez bueltatzen digu. Utilitate klasea erabili nahi dugu **Erregistro** motako array batetik elementu baliogabeak kentzeko. Erregistro klaseak *baliagarriaDa()* metodo bat du, objektuaren baliagarritasuna bueltatzen diguna, baina ezin dugu klase horren kodea aldatu. Nola erabili dezakegu Utilitate klasea Erregistro array baten elementuak filtratzeko?

Eskatzen da:

- a) ¿Zein patroi erabiliko zen(it)u(z)ke esandako softwarea garatzeko? Erantzuna arrazoitu.
- b) UML klase diagrama zehaztu klase nagusiekin eta garrantzitsuak iruditzen zaizkizun metodoekin.

Ikasturtea:	2018/2019
Gaia:	Software Ingeniaritza
Ariketak:	Patroiak
Data:	2019ko Otsaila

c) Aurreko klaseen implementazioa idatzi.

26. Ariketa (2014ko Uztaila)

Ondorengo kodean:

- Erabili diren patroiak identifikatu.
- Zein da patroi bakoitzaren helburua?
- Zein abantaila ematen digute?
- Zein desabantaila izan dezake sistemak eboluzionatzen badu?
- UML klase diagrama irudikatu, patroiak non dauden azalduz.

```
import java.util.ArrayList;
import java.util.Iterator;
public class Liburutegia {
    private static Liburutegia nLiburutegia = null;
    private static ArrayList<Liburua> liburuak;
    private Liburutegia() {};
    public static Liburutegia getInstance(){
        if (nLiburutegia == null) {
            nLiburutegia = new Liburutegia ();
        }
        return nLiburutegia;
    }
    public static void liburuakErakutsi (){
        Iterator<Liburua> itLiburuak = liburuak.iterator();
        while (itLiburuak.hasNext()){
            itLiburuak.next().liburuaErakutsi();
        }
    }
}
public class Liburua {
    private String izenburua;
    private String egilea;
    public Liburua (String pIzenburua, String pEgilea){
        izenburua = pIzenburua;
        egilea = pEgilea;
    }
    public void liburuaErakutsi(){
        System.out.println("Izenburua: " + izenburua);
    }
}
```

Ikasturtea:	2018/2019
Gaia:	Software Ingeniaritza
Ariketak:	Patroiak
Data:	2019ko Otsaila

```

        System.out.println("Egilea: " + egilea);
    }
}

```

27. Ariketa (2014ko Uztaila)

Zainketa Intentsiboetako Unitatean (ZIU) ospitaleratutako gaixoen bihotz taupaden maiztasuna kontrolatzeko aplikazio bat inplementatu nahi da. Honetarako **MaiztasunKontrolatzailea** izeneko klase bat diseinatu da, gaixoaren bihotz taupaden maiztasuna kontrolatuko dueña eskumuturrean daraman eskumuturrekoa erabiliz. Gainera, aplikazioak **MaiztasunPantaila** klasea erabiliz gaixoaren bihotz taupaden maiztasuna erakutsiko du gaixoaren ohe alboan dagoen pantailan. Balio hau maiztasuna aldatzen den bakoitzean eguneratuko da. Bukatzeko, momenturen batean maiztasuna minutuko 120 taupadatik igotzen bada, **MaiztasunAltuAbisua** klaseak soinu bat egingo du osasun langileei abisatzeko. Eskatzen da:

- Zein patroi erabiliko zenituzke aplikazio hau inplementatzeko? Zergatik? Bere funtzionamendua azaldu zure aukera justifikatuz.
- Aplikazio honen klase diagrama irudikatu, metodo garrantzitsuenak adieraziz.
- Zure hitzekin eta laburki azaldu klaseen funtzionamendua eta noiz deitzen zaion metodo bakoitzari.

28. Ariketa (2014ko Maiatza)

Ondorengo kodean:

- Erabili diren patroiak identifikatu.
- Zein da patroi bakoitzaren helburua?
- UML klase diagrama irudikatu, patroiak non dauden azalduz.

```

public interface TagOperator {
    void invokeEnd();
}

```

```

public class XMLParser extends DefaultHandler {
    private static XMLParser mXMLParser = new XMLParser();
    private String texto = null;
    private Character letra = null;
    private Definicion definicionActual = null;
    private TagOperatorS operatorFab;

    private XMLParser() {
        operatorFab = new TagOperatorS();
    }

    public static XMLParser getPDF2XMLParser() {
        return mXMLParser;
    }

    public void parseXmlFile(String pFile)
        throws XmlParsingException {
        SAXParserFactory saxParserFactory =

```

Ikasturtea:	2018/2019
Gaia:	Software Ingeniaritza
Ariketak:	Patroiak
Data:	2019ko Otsaila

```

SAXParserFactory.newInstance();
    try {
        SAXParser saxParser =
saxParserFactory.newSAXParser();
        saxParser.parse(new FileInputStream(pFile), this);
    } catch (Exception e) {
        throw new XmlParsingException(e);
    }
}

public void characters(char[] pCh, int pStart, int pLength)
    throws SAXException {
    texto = new String(pCh, pStart, pLength).trim();
}

public void endDocument() throws SAXException {
    System.out.println("Finished processing the file");
}

public void endElement(String pUri, String pLocalName, String
pName)
    throws SAXException {
    TagOperator op = operatorFab.getTagOperator(pName);
    if (op == null) {
        return;
    }
    op.invokeEnd();
}

public void startDocument() throws SAXException {
    System.out.println("Init parsing");
}

private class TagOperatorS {
    private Map<String, TagOperator> operators = null;

    public TagOperatorS() {
        operators = new HashMap<String, TagOperator>();
        operators.put("Definicion",
DefinicionTagOperator());
        operators.put("Letra", new LetraTagOperator());
        operators.put("Enunciado",
EnunciadoTagOperator());
        operators.put("Respuesta",
RespuestaTagOperator());
    }

    public TagOperator getTagOperator(String pName) {
        return operators.get(pName);
    }
}

// Tag operators
private class DefinicionTagOperator implements TagOperator {
    public void invokeEnd() {
        // Kodea
    }
}

private class LetraTagOperator implements TagOperator {
    public void invokeEnd() {
        // Kodea
    }
}

```

Ikasturtea:	2018/2019
Gaia:	Software Ingeniaritza
Ariketak:	Patroiak
Data:	2019ko Otsaila

```

    }
}

private class EnunciadoTagOperator implements TagOperator {
    public void invokeEnd() {
        // Kodea
    }
}

private class RespuestaTagOperator implements TagOperator {
    public void invokeEnd() {
        // Kodea
    }
}
}

```

29. Ariketa (2014ko Maiatza)

Sakeleko telefono baten bateria kontrolatuko duen aplikazio bat garatu nahi dugu. Aplikazio honek **Bateria** klasea izango du, bateria fisikoaren egoera kontrolatuko duena, bere karga maximoa eta momentuko karga dituelarik. Honetaz gain, aplikazioak **BateriaWidget** klase bat izango du, pantailan uneko karga erakutsiko duena, bateria %1 igo edo jaisten denean eguneratu behar delarik. Bukatzeko, **EnergiaKudeatzaileak** energia aurrezteko modua jarriko du martxan karga %30tik jaisten denean (pantailaren distira jaitsiko du, egunerapenak desgaituko ditu, etab. bateriaren gastua txikiagotzeko). Ondorengoa eskatzen da:

- Zein patroi erabiliko zenituzke aplikazio hau inplementatzeko? Zergatik? Bere funtzionamendua azaldu zure aukera justifikatuz.
- Aplikazio honen klase diagrama irudikatu, metodo garrantzitsuenak adieraziz.
- Zure hitzekin eta laburki azaldu klaseen funtzionamendua eta noiz deitzen zaion metodo bakoitzari.