

3 Aurrekariak

Kapitulu honetan proiektua burutu ahal izateko garrantzia izan duten iraganeko elementu esanguratsuak jasotzen dira. ProMeta proiektua ProWF proiektuaren jarraipena denez, aurrekariak antzekoak dira eta aurreko proiektuan oinarrituta daude.

Hasteko, software kalitatea eta gaur egungo bezeroen eskakizun gogorrak aztertuko dira. Ondoren, bezeroen eskakizunengatik agertu diren beharrak azalduko dira: softwarea garatzeko metodologiak eta proiektuak aurkezteko araua. Hori azaldu eta gero, *BPM (Business Process Management)* sistemek eskaintzen dituzten abantailak eta sortzen dituzten menpekotasunak aztertuko dira. Jarraian, inferentzia motorrak eta haien ahalmena komentatuko da, *CLIPS* eta *EHSIS* inguruneak aztertuz. Segidan, *Ereduek Bideratutako Ingeniaritza (MDE)* gauzatzeko softwarea garatzeko metodologia azalduko da eta *Edukiak Kudeatzeko Sistemak (CMS)* softwareak deskribatuko dira. Amaitzeko, ProWF sistemaren ezaugarriak aztertuko dira, aurretik azalduetako kontzeptuak kontuan hartuz.

3.1 Software Kalitatea

Softwarearen kalitatea bezeroen beharrak asetzen dituen softwarearen ezaugarriak kontrolatzean eta ziurtatzean datza, bezero eta erabiltzaileen onurarako. Softwarea produktua ez da fabrikatzen eta ez da fisikoki degradatzen, baina garapen-prozesu bat du. Hala ere, softwarea ez da ia inoiz perfektua izaten, asebate behar den bezeroaren arazoetan ezaugarri berriak agertzen direlako. Proiektu ororen helburua ahalik eta kalitate oneneko softwarea ekoiztea da, bezero eta erabiltzaileen itxaropen eta betekizunak gaindi ditzan. Softwareak akatsak eta gorabeherak izan ditzake, baina ez dira edozein ekipo fisikoren antzekoak, horietan ausazko hutsegiteak eman baitaitezke. Softwarearen kasuan, denak sistematikoak dira eta, ondorioz, zuzendu beharrekoak.

Softwarearen ziurtagiria bere kalitatea ziurtatzeko prozesuaren ondorioa da, baina ez da inoiz azken helburua. Softwarearen kalitatea ez da ziurtatzen, bermatu behar direnak kalitatezko softwarea eraikitzekeko prozedurak dira. Prozedurek, bezeroek eskatutako kontsentsu maila altuko nazioarteko estandarretan oinarritutakoak izan behar dute eta, herrialde bakoitzean, normalizazio agentzia ofizialaren berariazko lan-taldeak egokitutakoak. Prozeduren kalitate-eredu ezberdinak daude, garrantzitsuenak *CMMI* eta *SPICE* dira, helburu berdina dutenak. Kalitate-ereduek, softwarea garatzeko praktika onenak definitzen dituzte, softwarea garatzen duten erakundeek prozesuak hobetzen laguntzeko. Ziurtagiri-emaeleek, praktika on horiek hartu eta beraien emaitza neurgarriak egiaztatzen dituzte garatzaileekin lankidetzan. Ebaluazio-prozesu batean parte hartu ahal izateko inplikatu guztiak ziurtagiri egokiaren jabe izan behar dira.

3.2 Bezeroen Eskakizun Gogorrak

2007ko urriaren 30ean onartutako *Ley de Contratos del Sector Público* legeak, *BOE-261-2007-18874*, bere 69. artikuluan, hornitzaileek kalitate bermeak erakusteko arau europarrei erreferentzia egin eta, dagozkien ziurtagiriak arauarekin bat datozen erakundeek emanak izan behar dutela ezarri zuen.

Ondoren, aurreko legearekin bateratuta, 2011ko azaroaren 14an Estatuko Aldizkari Ofizialean onartu zen legearen ostean, *BOE-A-2011-17887*, bezeroen eskakizunak sendo gogortu ziren, softwarearen kalitateari dagokionez. Lege horren 80. artikulua kalitatea bermatzeko arauak betetzen direla egiaztatzea zuen helburu, horretarako erakunde independenteek emandako ziurtagiriak beharrezkoak ziren merkatuak exijitzen bazituen. Erakunde horiek kalitatea bermatzeko Europako arau jakin batzuei erreferentzia egin

edo baliokideak izan behar ziren. Hortaz, lege horren ostean software garapenean kalitatea bermatzeko *CMMI* edo *SPICE* erakundeei lotutako ziurtagiriak lortzea beharrezkoa bilakatu zen.

CMMI erakunde baten softwarea garatzeko prozesuaren heldutasuna ebaluatzeko eta neurtzeko metodoak erabiltzen dituen prozesua da. Erakundeen heldutasun-maila neurtzeko bost etapa bereizten ditu, 3.1. Taulan ikusten direnak.

Maila	Etap	Prozesuaren egoera
I	Hasierakoa	Kaotikoa da, kontrol zaila du, erreaktiboa da
II	Kudeatuta	Proiektuen ezaugarriak ditu, askotan erreaktiboa da
III	Definituta	Erakundeen ezaugarriak ditu, proaktiboa da
IV	Kuantitatiboki kudeatuta	Erdi-mailako prozesua da, kontrolatuta dago
V	Optimizazioa	Etengabeko hobekuntza dago

3.1. Taula. *CMMI*-ren 5 heldutasun-mailak.

*CMMI*ren 3. heldutasun-maila lortzea ezinbestekoa zen merkatuan mantentzeko, hori dela eta, hainbat enpresa kexatu ziren merkatutik kanpo geratzeagatik, baina alferrikakoa izan zen. Horren adibidea, 2016an *OESIA NETWORKS, S.L.* enpresak jarritako erreklamazioa da, *Recurso 0006-2016*, honek *CMMI*ren 3. heldutasun-maila lortzeaz zegoela erreklamatzeko merkatuan mantentzeko, tramitazioaren faltan zegoen. Hala ere, administrazioaren erantzuna errekurtsua baiestearen aurka egotea izan zen, lehen aipatutako 80. artikulua ez betetzeagatik. Arrazoia enpresak oraindik ez zuela eskatutako kalitate maila bermatzen zuen ziurtagirik izan zen eta gainera, maila hori zuten software enpresa asko zeudela.

Ebidentzia horiek direla eta, gaur egun proiektu informatiko baten bezeroek dute pisu handiena edo agintea, haien exijentzia gogorak betetzea funtsezkoa da merkatuan mantentzeko. Bezeroen behar eta ametsetan fokatzea, 2018ko *CMMI 2.0* bertsioan eta estandar internazionaletan sendotuta geratu da.

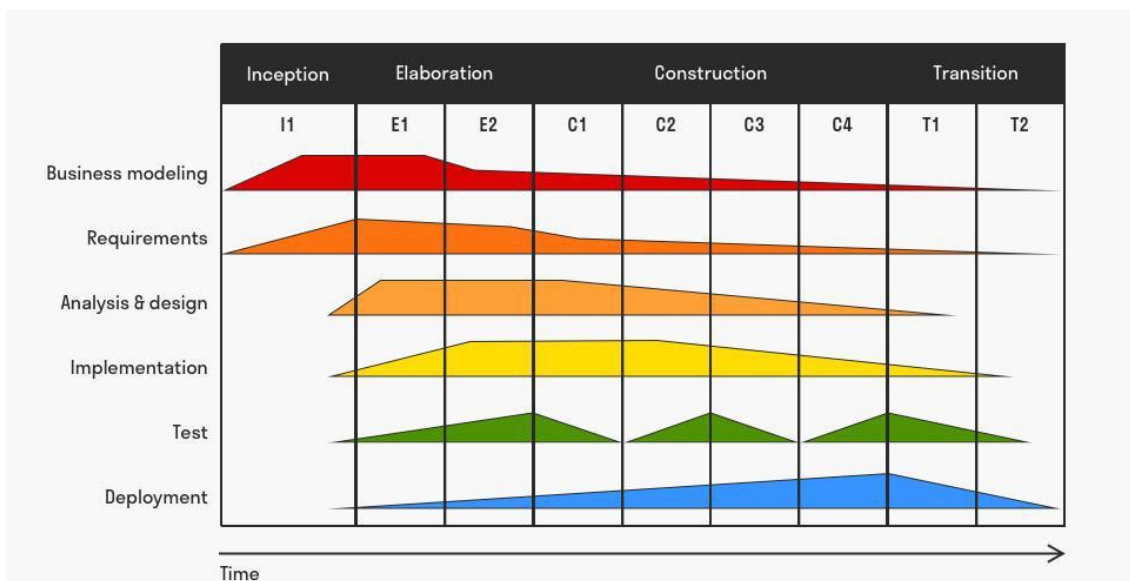
3.3 Garapen Metodologiak

Gaur egun pentsaezina bilakatu da software proiektu bat aurrera ateratzea metodologiarik jarraitu gabe. Software munduan artisautza lanak ez du etorkizunik, metodologia baten ezarpenak hori ekiditeko balio du. Software ingeniariak edozein momentutan jakin behar du zer egin, noiz eta nola, bestela arazoak eta galerak agertzeko probabilitatea handituz joango da proiektuaren garapena luzatzen doan heinean.

Gainera, proiektuak zerotik egitea garestiegia da software garapen enpresentzat. Berrerabilpenean oinarritutako metodologia bat aukeratuz, enpresaren kostu ekonomikoak gutxituko dira epe laburrean.

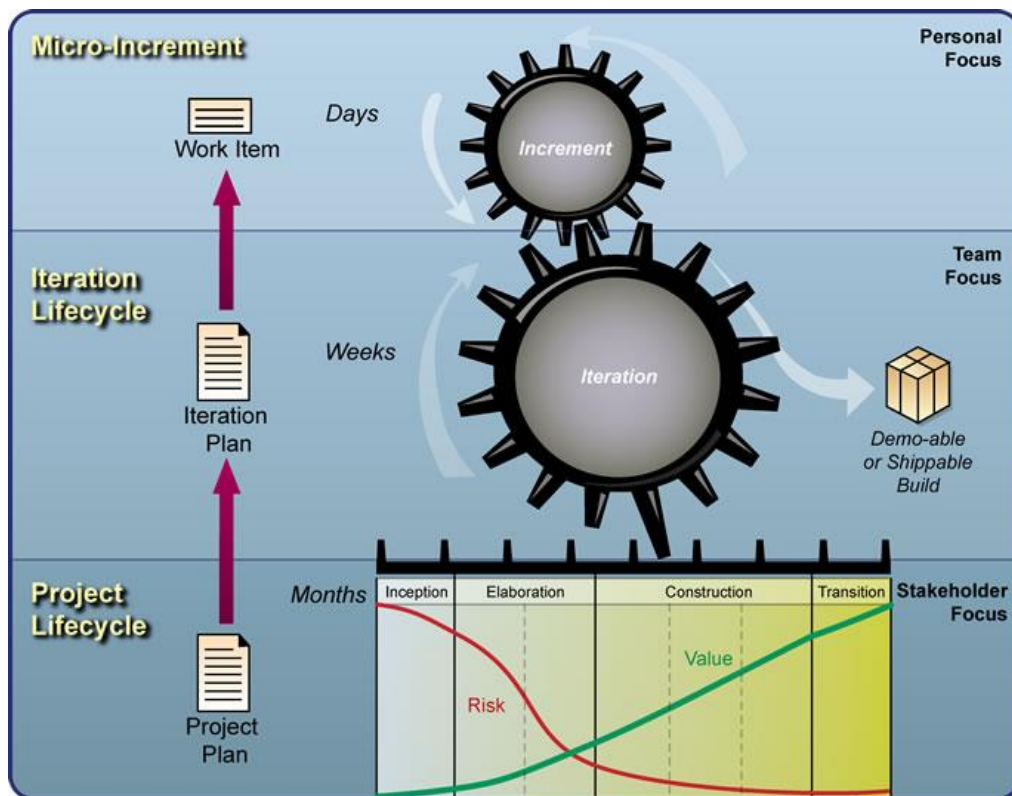
Gehien erabiltzen direnak metodologia arinak edo bizkorak dira. Arina kontzeptua softwarea sortzeko urratsak arindu behar direlako sortzen da, urrats bakoitzeko prototipo bat sortuz. Giza interakzioetan zentratzen da, aldean arteko elkarriketa-fluxuari eutsi ahal izateko, garapen dinamikoagoa eta parte-hartzaileagoa ahalbidetzeko. Metodologia bizkorrek garapen-sistema egokitzaille bat erabiltzen dute, eta ez prediktiboa. Horrek esan nahi du lantaldeak buruan duela nahi duen emaitza, baina ez daki zehatz-mehatz zer produktu mota sor dezakeen.

Software garapenaren metodologia arinen artean RUP aurkitzen da. Softwarea garatzeko prozesu iteratibo bat da, Rational Software Corporation erakundeak sortua, IBM enpresaren dibisio bat. 3.1. Irudian ikus daiteke RUP prozesuaren egitura. Hala ere, RUP ez da zehatz-mehatz jarraitu behar den prozesua, baizik eta prozesu moldagarria da, garapen-erakundeek eta software-proiektuen taldeek egokitzeko asmoarekin, bakoitzaren premietarako egokiak diren elementuak hautatuz.



3.1. Irudia. RUPen prozesu iteratiboa, bizi-zikloaren faseak, iterazioak eta jarduerak.

RUPen oinarritutako metodologia simple eta erabiliena OpenUP da. Metodologia horrek RUPen funtsezko ezaugarriak gordetzen ditu, garapen iteratiboa, erabilpen-kasuak, arriskuen kudeaketa eta arkitekturan oinarritutako ikuspegia bultzatzen duten agertokiak barne. RUPen erabiltzen ez diren aukerako zati gehienak baztertu eta elementu asko bateratzen ditu. Eraitza prozesu askoz sinpleagoa da, eta RUP printzipioekiko leiala izaten jarraitzen du. 3.2. Irudian ikus daiteke OpenUP prozesuaren laburpen bat. Proiektuaren tamainaren arabera OpenUPtik RUPera pasatzea komenigarri ikusiko dugu, informazio zehatzagoa gorde ahal izateko.



3.2. Irudia. OpenUP-en geruzak: mikro-gehikuntzak, iterazio bizi-zikloa eta proiektu bizi-zikloa.

3.4 Proiektuen Aurkezpenerako Arauak

Proiektuen garapenean zehar sortzen diren dokumentazio multzoaren aurkezpenerako ezinbestekoa da ezarrita dagoen araudi ofiziala betetzen dela bermatzea, nazioarteko estandarren oinarritutakoa. Helburu nagusia proiektuan parte hartzen duten alderdi guztien aldeko dokumentazioa osoa eta gardena izatea izanik, bezeroaren gogobetetasuna handituz.

Naziortean ISO erakundeak ezarritako estandarrak jarraituz, Espainian araurik finkatuenak *UNE* eta *CCII* erakundeek aurkeztutakoak dira: **CCII-N2016-02. Norma Técnica para la realización de la Documentación de Proyectos en Ingeniería Informática**. Zehatz-mehatz deskribatzen du nola egin behar den Informatika Ingeniaritzako proiektu baten dokumentazioa. Proiektuko dokumentazioa egiteko ereduak eta dokumentazioaren antolamendua deskribatzen du ere. Erreferentziako nazioarteko esparru eta estandarrak kontuan hartzen ditu. Hala nola, *UNE 157801:2007* – Informazio-sistemen proiektuak egiteko irizpide orokorrak, *UNE-ISO 21500:2013* – Proiektua zuzentzeko eta kudeatzeko jarraibideak eta PMIren *PMBOK* – Proiektuen Zuzendaritzarako Oinarrien Gida. Proiektu baten bisa edo ikuskaritzak egiteko **CCII-N2016-01** araua ere eskuragarri dago.

Proiektuak aurkezteko araudien gabeziak proiektu informatiko batean gatazkak ekarri ditzake. Hau da, proiektuan esku hartzen duten aldeentzat nahi ez diren ondorioak eragin, bezero, hornitzaile zein interesdunen arteko gatazkak sortuz. Epaitegietan bukatzea oso kaltegarria da edozein software enpresentzako.

3.5 Prozesuetan Oinarritutako Garapena

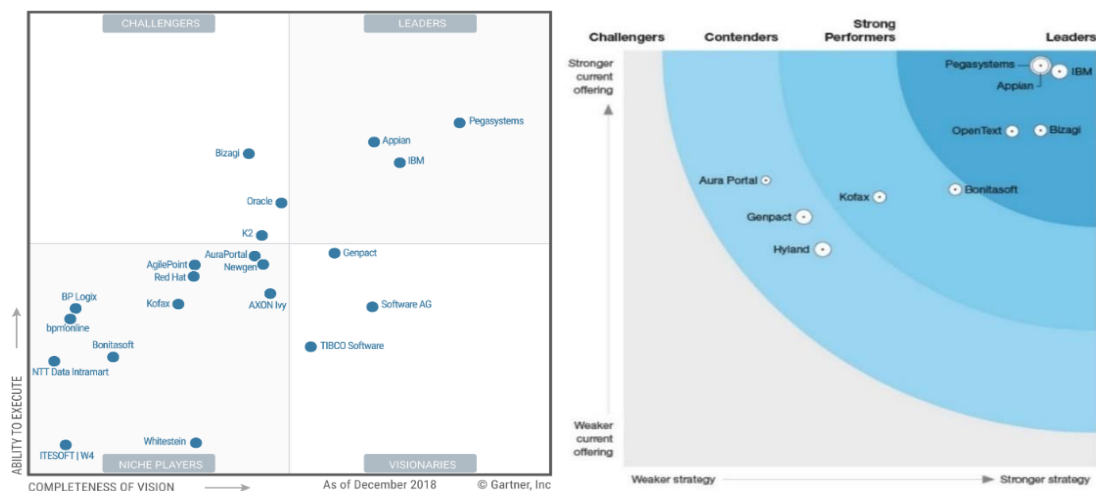
Business Process Management (BPM) negozio-prozesuak definitzera eta aplikatzera bideratutako metodologia edo ikuspegi estrategiko gisa ikus daiteke. Negozio-prozesuak konplexuak eta dinamikoak direnez malguak izan behar dute, negozioa aldatzen joango delako eta etengabe eguneratu behar delako.

Hona hemen BPMrekin lan egitean lortzen diren abantailak:

- Ataza errepikakorrak ezabatzea eta automatizatzea.
- Eraginkortasuna handitzea: prozesuetan akatsak minimizatuz, itxarote-denbora murriztuz, giza esku-hartzeak murriztuz eta lana berregitea saihestuz
- Negozio-arauak betetzen direla ziurtatzea.
- Zerbitzu-maila bermatzea, salbuespenak maneiatuz, egoeren jarraipena eginez, gertakariak mailakatuz, prozesuen sendotasuna eta trazabilitatea bermatuz, etab.
- Lan egiteko modua aldatzeko aukera eskaintzea, inpaktua murriztuz eta etengabe hobetuz.

Gartner eta *Forrester* aholkularitza-enpresek BPM merkatua ikertzeko, urtero, txosten bat argitaratzen dute. Txosten horiek patentatutako datu kualitatiboak aztertzeke metodoetan oinarritzen dira, merkatuaren joerak frogatzeko, hala nola zuzendaritza, heldutasuna eta parte-hartzaileak.

Gartner-en Koadrante Magikoa industria teknologikoaren azpisektore nagusiei buruz egiten duen merkatu-ikerketari bati erantzuten dio. 3.3. Irudian lau koadrantetan banatutako grafiko gisa aurkezten da. X ardatzak exekutatzeko gaitasuna adierazten du, eta Y ardatzak, berriz, balio-proposamen osoa. Ezkerretik eskuinera eta goitik behera, sektore bakoitzeko enpresak honako koadrante hauetan kokatzen dira: *challengers*, *leaders*, *niche players* eta *visionaries*. Antzeko informazioa aurkitu dezakegu **¡Error! No se encuentra el origen de la referencia.** agertzen den *Forrester Wave* txostenean. Bi txosten ezagun hauek, ondo kokatutako hornitzaileek komertzialki erabiltzen dituzte.



3.3. Irudia. BPM Gartner-en Koadrante Magikoa eta Forrester Wave txostena.

Bizagi bi txostenetan agertzen da, ondo kokatuta, gainerako liderrak tamaina handiko enpresa eta erakundeen prozesu kudeaketan fokuratzeko dute. *Bizagi* bezalako teknologia erabiliz arkitektura konplexuko web-aplikazioa inplementatu daiteke, prozesu, erregela eta *workflow*-ak exekutatzen motorrak integratuak dituen. Apustu hori garestia izan daiteke eta proiektu edo enpresaren menpekotasun teknologikoa handituko sortu. Beste apustu merkeago bat, metodologia eta estandarretara hurbiltzen joateko, teknologia propioa sortzea da, *workflow*-lengoaia propioa sortuz eta lengoaia horretan adierazitako ezagutza exekutatzen inferentzia motorrak erabiliz.

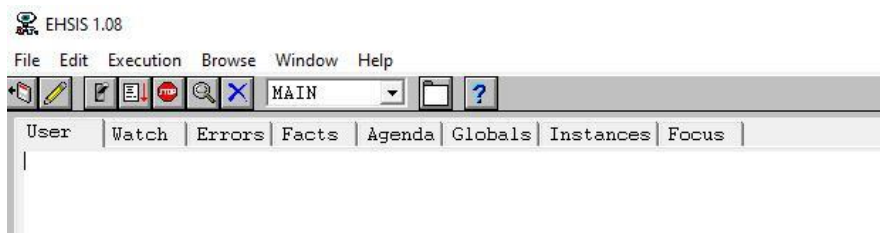
3.6 Ezagutzaren Kudeaketa eta Inferentzia Motorrak

ProWF proiektuan teknologia propioa sortzea erabaki zen, lan-fluxu edo *workflow*-en bidezko lengoaia propioa sortuz eta berekin adierazitako ezagutza inferentzia motorren bidez exekutatuz.

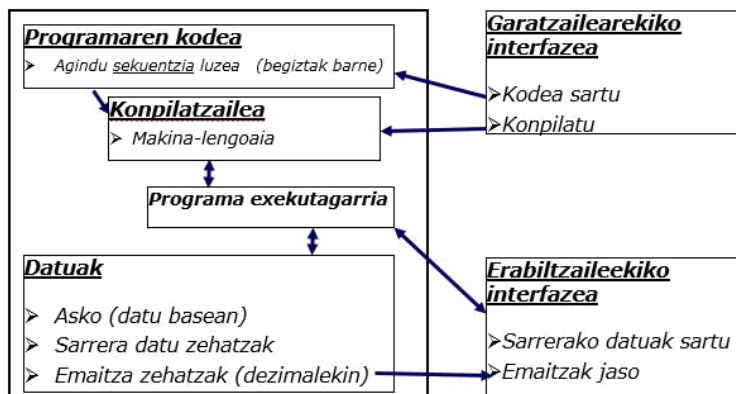
CLIPS softwareak inferentzia motor baten erabilera errazten digu, sistema adituak ekoizteko eta exekutatzen garapen-ingurune bezala erabili daiteke. *NASAK* sortu eta erabilia jabari publikoan utzita dago. Bere lengoia erregelak, objektuetara bideratutako programazioa eta programazio prozedurala erabiltzen du ezagutza adierazteko. Lengoaia sinplea baino oso ahaltsua da, hurrengo ezaugarrietan nabarmenduz:

- **Garraigarritasuna.** C lengoia ematen diona.
- **Integrazioa edo zabalkortasuna.** Programazio prozeduralari esker funtzio berri ahaltsuak sor daitezke. *Service-oriented Architecture* (SOA) estandarra erabilia BPM edo beste edozein sistemekin integragarria da.
- **Interakzio edo diziplinarteko garapena.** Formakuntza ezberdinetako pertsonen ideiak azkar inplementatzeko aukera, objektu eta erregelatan oinarritutako lengoaia erabiliz.
- **Egiaztapen edo balidazio errazagoa.** Horretarako, funtzio bereziak erabiliz.

EHSIS, ordea, Euskal Herriko Unibertsitateko (EHU) *ERABAKI* taldeak hedatutako ingurunea da, *CLIPS* 6.04, *FuzzyCLIPS* 6.04, objektuetara eta gertaeretara bideratutako programazioa, interfazeen garapena, komunikazioa eta leihuetan oinarritutako ingurunea integratzen duena. *EHSIS* inguruneak (ikusi 3.4. Irudia), softwarea garatzeko baliabide tradizionalak oinarritutako aplikazioak sortzeaz gain (ikusi 3.5. Irudia), *COOL* lengoaia erabilia, ezagutzan oinarritutakoak ere sor ditzake sistemaren arkitektura egoki batekin baliatuz.

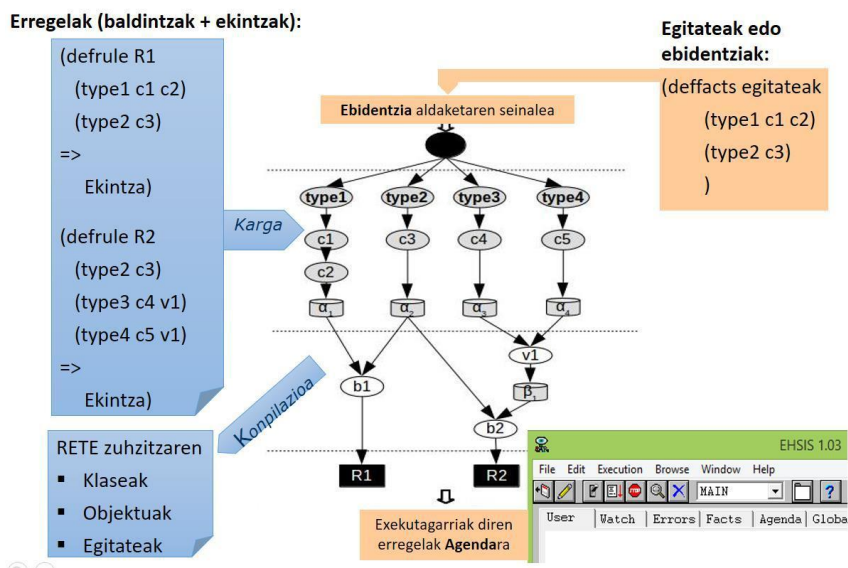


3.4. Irudia. EHSISen garapen ingurunea.



3.5. Irudia. Programazio tradizionalaren eskema.

Erregelak idazteko lengoaia sinplea denez, klase eta objektuak erabili ere, domeinuko ezagutza adierazteko ondo diseinatutako lengoaia bat lortuz gero, ezagutza hori exekutatze kodea automatikoki sortzea ere posible da. Gainera, inferentzia motorrarekin abiadura handiko exekuzio eraginkorra lortu daiteke kode guztia *RETE* sare bezala gordeta (ikusi 3.6. Irudia), hau konpilazio mota bat bezala erabili daiteke. *RETE* sarearen egitura bereziak *RETE* algoritmoaren abiadura, egitate eta erregela askorekin ere, aprobeztatzeko aukera ematen du. Erreminta hau, bere eraginkortasuna eta jabari publikoa dela eta, konplexutasun handiko proiektuetan erabili daiteke produktu mantengarriak lortzeko.



3.6. Irudia. Erregeletan oinarritutako sistemen eskema: klaseak, objektuak, erregelak eta egitateak.

*EHSIS*en garapen inguruneak badu bere produkzio bertsioa, *EHSIS_RT* deitutakoa. Web-zerbitzuetarako bertsioa ere badu, *Mairi* deitutakoa.

3.7 Ereduek Bideratutako Ingeniaritza

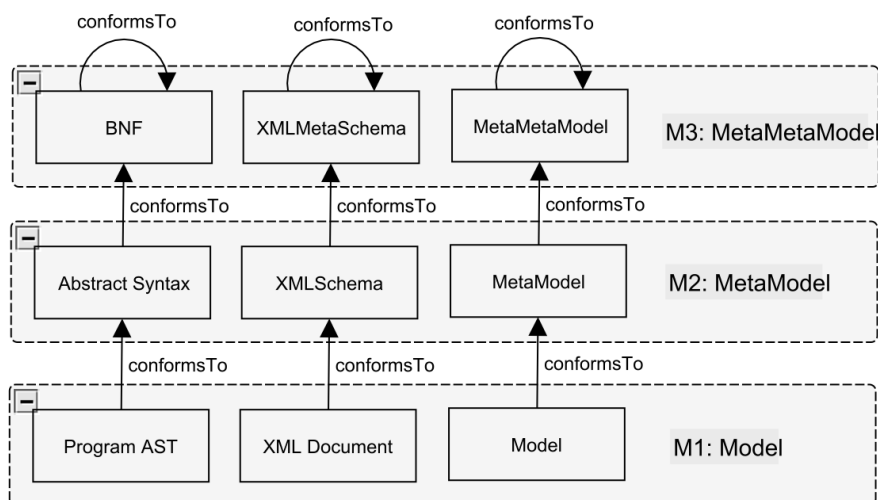
Model Driven Engineering (MDE) edo eredu bidezko ingeniarietza softwarea garatzeko metodologia da. Domeinu ereduak erabiltzen ditu, hau da, arazo zehatz bati lotutako gai guztien eredu kontzeptualak. Hori dela eta, aplikazioen domeinu jakin baten ezagutzaren eta jardueren irudikapen abstraktuak nabarmentzea du helburu, kontzeptu informatikoetan sartu gabe.

MDEren helburua produktibitatea handitzea da. Horretarako, sistemen arteko bateragarritasuna maximizatzen du eredu estandarizatuak berrerabiliz, diseinu prozesua sinplifikatu aplikazioaren domeinuko diseinu patroierrepikakorren eredu bidez eta sisteman lan egiten duten pertsonen eta taldeen arteko komunikazioa sustatu praktika onen estandarizazioaren bidez.

MDEren modelatze paradigma bat eraginkorra dela deritzo bere ereduak domeinua ezagutzen duen erabiltzaile baten ikuspuntutik zentzua badute eta sistemak ezartzeko oinarri gisa balio badute. Ereduak produktuen kudeatzaileen, diseinatzaileen, garatzaileen eta aplikazioaren domeinuko erabiltzaileen arteko kolaborazioarekin garatzen dira. Ereduak amaitzen doazen heinean, softwarea eta sistemak garatzea ahalbidetzen dute.

MDEn 4 modelatze maila daudela esan dezakegu. Maila handiagoa den heinean abstrakzio-maila igotzen doa. 3.7. Irudian ikus daitezke 3 abstrakzio-maila altuenak: eredu, metaeredua eta meta-metaeredua.

- **M0. Instantziak.** Maila hau exekuzio sistemari dagokio. Maila honetan negozioko elementuak daude, edo mundu errealeko elementuen errepresentazioak (software errepresentazioak).
- **M1. Eredua.** Eredu honek software sistemen itemak adierazten dituzten kontzeptuak dauzka. M1 mailan dauden kontzeptuek M0 mailan dauden instantziak kategorizatu edo sailkatzen dituzte.
- **M2. Metaeredua.** M1 mailan dauden kontzeptuen inguruan arrazoitze beharrezkoak diren kontzeptuak dauzka. M2 mailako elementu batek M1-eko elementuak espezifikatzen ditu. Ereduen eredu honi metaeredu esan ohi zaio.
- **M3. Meta-metaeredua.** M2 mailan dauden kontzeptuen inguruan arrazoitze beharrezkoak diren kontzeptuak dauzka. M3 mailako elementuek M2-ko elementuak kategorizatzen ditu. Meta-metaeredu esan ohi zaio.



3.7. Irudia. MDEren 3 abstrakzio maila: eredu, metaeredua eta meta-metaeredua.

3.8 Edukiak Kudeatzeko Sistemak

Content Management System (CMS) izeneko softwareak dokumentuak eta bestelako edukiak antolatu eta kudeatzeko erremintak dira, webguneak eta web edukiak kudeatzeko balio dutenak. Gaur egun sistema asko daude arlo honetan, bai kode irekikoak eta baita jabetunak ere.

Edukiak kudeatzeko sistema zerbitzarian kokaturik egon ohi da. CMS batera atzipena erabiltzaile motaren arabera maila ezberdinetan egin daiteke. Adibidez, edukiaren sortzaileek sistema osatuko duten dokumentuak sortuko dituzte. Argitalpen-teknikariek dokumentu hauek aipatu, onartu edo baztertuko dituzte. Editoreen burua gure web orrian eduki hori argitaratzeaz arduratuko da. Dena interfaze grafiko baten bidez kontrolatuko da, honen erabilera erraztu ahal izateko.

ProWF eta ProMeta proiektuen IO-System sistema garatzeko Drupal CMSa erabili da.

3.9 ProWF Sistema

Esan bezala, ezinbestekoa da ProWF sistema [1] ulertzea proiektu hau horren jarraipena delako. Horretarako, bertan zehazten diren rolak, sistemaren azpisistemak eta osagaiak deskribatuko dira.

ProWF sistemaren erabiltzaileek hurrengo rol hauekin lan egin behar dute:

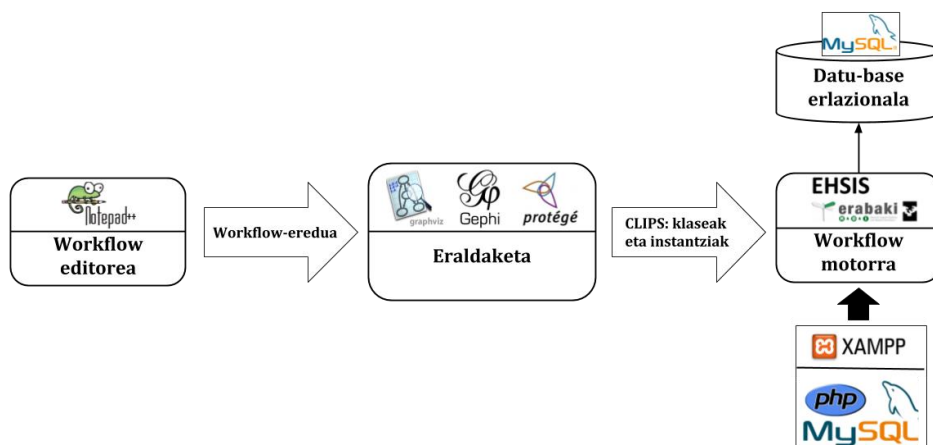
- **Prozesu Sortzailea:** sortutako workflow-lengoaia baliatuz, softwareen garapenerako bizi-zikloa ezartzen duen metodologia definituko duena.
- **Analista:** sortutako workflow-ereduak adierazten duen bizi-zikloa exekutatzeaz eta aplikatzeaz arduratuko da.
- **Proiektuko Zuzendaria:** proiektuak sortu eta proiektuko partaideen rolak esleitu ditu.
- **Kalitate Arduraduna:** workflowaren exekuzioan zehar sortuko diren artefaktuen kalitatea bermatzea du helburu, balorazioak emanez.

ProWF sistema bi azpisistema ezberdinetan bananduta dago:

- **Workflow Editor:** sortutako workflow-lengoaia erabiliz workflow-ereduak sortzeko balio du, ondoren, eredu horiek workflow motorrak exekutatzeke eraldatuko ditu. Testu-editore bat izango da. Rolak: Prozesu sortzailea.
- **IO-System:** workflow kudeatzaileraren sarrera/irteerak maneiatzea du helburu. CMS baten bitartez kudeatutako web-aplikazioa izango da. Rolak: Proiektuko zuzendaria, analista eta kalitate arduraduna.

3.8. Irudian ikusten den moduan, hauek dira **Workflow Editor** azpisistemaren arkitekturaren osagaiak:

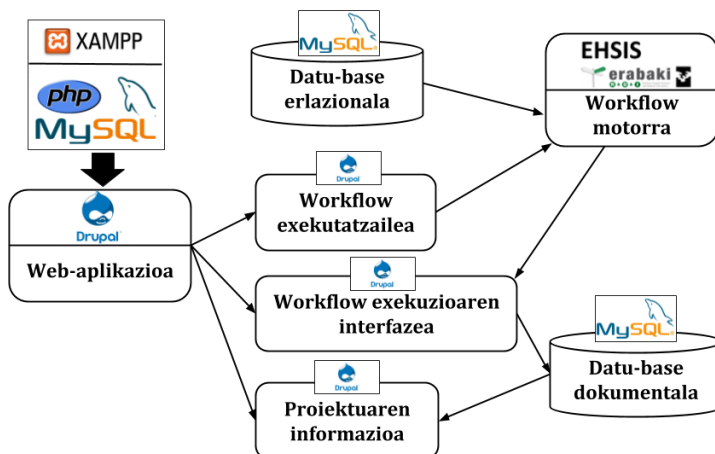
- **Workflow editorea:** definitutako lengoaia grafiko bat erabilita, testu editore batekin, DOT lengoiaren bitartez workflow-ereduak sortzeko balio du.
- **Eraldaketa-prozesua:** DOT lengoian idatzitako workflow-eredua CLIPS lengoiako klase eta instantzietan eraldatzea du helburu.
- **Datu-base erlazionala:** sortutako workflow-ereduen informazio gordetzeaz arduratzen da.
- **Workflow motorra:** CLIPS lengoiara eraldatutako workflow-ereduaren klaseak eta instantziak baliatuz, workflowa exekutatzeke erregela batzuen bidez, informazio guztia datu-base erlazionalan gordeko du.



3.8. Irudia. ProWF Workflow Editor azpisistemaren arkitektura.

3.9. Irudian ikusten den bezala, hauek dira **IO-System** azpisistemaren arkitekturaren osagaiak:

- **Web-aplikazioa:** CMS baten bitartez, lokalean erabiltzeko (zerbitzari batean ere kokatu daiteke) pentsatuta dagoen sistema kudeatzailea. Hurrengo osagai nagusiak integratuko ditu:
 - **Workflow exekutatzailea:** workflow motorrari aginduak emango dizkio eta ondoren, bere irteerak jasoko ditu.
 - **Workflow exekuzioaren interfazea:** erabiltzaileak exekuzioan zehar ikusiko duena da, workflow motorraren irteeran adierazitako ekintzak betetzeko interfazea da.
 - **Proiektuaren informazioa:** CCII-2016N-02 araua jarraituz, erabiltzailearen dokumentuak antolatuta dituen webgune antzeko bat da.
- **Workflow motorra:** Datu-base erlazionalari kontsultak eginez, workflow-eredua prozesatuko du eta jardueraz-jarduera joango da egin beharreko ekintzak irteera lez bueltatuz.
- **Datu-base erlazionala:** sortutako workflow-ereduen informazioa gordetzeaz aparte, erabiltzaileen proiektu eta artefaktuen informazioa biltegitratuko du.
- **Datu-base dokumentala:** erabiltzaileek idatzitako dokumentuen kokapena eta sekzio bakoitzaren edukia gordeko du.



3.9. Irudia. ProWF IO-System azpisistemaren arkitektura.