

eman ta zabal zazu



Universidad  
del País Vasco

Euskal Herriko  
Unibertsitatea

## Informatikan Ingeniaritza

Donostiako Informatika Fakultatea

Karrera Amaierako Proiektua

---

# SAP - METAPOSTA konektorea

---

Egilea

*Ander Bustamante*

Tutorea

*Imanol Usandizaga*



2014



---

## Laburpena

---

SAP ERP sistema eta METAPOSTA interneteko posta zerbitzua erabiltzen duten enpre-  
sen arteko elkarlana errazteko helburuarekin, Oreaka IT enpresak bi sistema hauen arteko  
komunikazioak ahalbidetzen dituen sistema bat sortu nahi izan du.

Helburu honekin, proiektu honen funtsa den **SAP - METAPOSTA konektorea** sortu da,  
nahi duten enpresek baliatu eta SAPen sortzen dituzten bere dokumentuak interneteko  
zerbitzu honen bidez zabaldu ditzaten.

Honela, Konektorea erabiltzen duen edozein enpresak, bere dokumentuak modu automa-  
tikoan sortu eta METAPOSTAra bidaliko ditu bere SAP sistematik at beste ezer erabili  
gabe. Hori egiteko, SAP sistemako ABAP teknologia eta SAP PI moduluak erabili dira.



---

# Gaien aurkibidea

---

<b>Laburpena</b>	<b>i</b>
<b>Gaien aurkibidea</b>	<b>iii</b>
<b>Irudien aurkibidea</b>	<b>ix</b>
<b>Taulen aurkibidea</b>	<b>xiii</b>
<b>1 Sarrera</b>	<b>1</b>
1.1 Oreka IT . . . . .	1
1.2 METAPOSTA . . . . .	2
1.3 Motibazioa . . . . .	2
<b>2 Proiektuaren Helburu Dokumentua</b>	<b>3</b>
2.1 Helburua . . . . .	3
2.2 Lan metodologia . . . . .	4
2.3 Irismena . . . . .	4
2.3.1 Irismen minimoa . . . . .	4
2.3.2 Zabalkuntza plana . . . . .	5
2.3.3 Gogobetetzea bermatzeko argibideak . . . . .	6
2.3.4 LDE diagrama . . . . .	7
	iii

---

2.3.5	Atazen deskribapena . . . . .	8
2.4	Plangintza . . . . .	9
2.4.1	Burutu beharreko atazak . . . . .	9
2.4.2	Kronograma . . . . .	11
2.4.3	Komunikazio plana . . . . .	12
2.4.4	Kalitate plana . . . . .	13
2.4.5	Arriskuen plana . . . . .	14
2.4.6	Aldaketen plana . . . . .	16
<b>3</b>	<b>Oinarri Tekniko-Teknologikoak</b>	<b>17</b>
3.1	SAP . . . . .	17
3.1.1	SAP Arkitektura . . . . .	18
3.1.2	SAP Moduluak . . . . .	18
3.2	ABAP . . . . .	19
3.3	SAP PI . . . . .	20
3.3.1	Zer da SAP PI? . . . . .	20
3.3.2	Arktitektura . . . . .	22
3.3.3	Kronologia . . . . .	24
3.3.4	Integration Process (ccBPM) . . . . .	24
3.3.5	Scenario bat sortzen . . . . .	25
<b>4</b>	<b>Analisia</b>	<b>31</b>
4.1	Sistemaren deskribapena . . . . .	31
4.2	Betekizunen analisia . . . . .	32
4.3	Erabilpen Kasuak . . . . .	35
4.3.1	Erabiltzaileen harpidetza . . . . .	36
4.3.2	Dokumentuen sorketa . . . . .	37
4.3.3	Dokumentuen bidalketa . . . . .	38
4.3.4	Erantzunen prozesamendua . . . . .	39

---

<b>5</b>	<b>Diseinua</b>	<b>41</b>
5.1	Erabiltzaileen harpidetza . . . . .	41
5.2	Dokumentuen sorketa . . . . .	45
5.3	Dokumentuen bidalketa . . . . .	48
5.3.1	Dokumentuak bidali . . . . .	48
5.3.2	Erantzunak jaso . . . . .	50
5.4	Erantzunen prozesamendua . . . . .	51
5.5	Datu Baseko taulak . . . . .	55
5.5.1	Web Service-erako taulak . . . . .	55
5.5.2	Erabiltzaileen harpidetzarako taulak . . . . .	57
5.5.3	Dokumentuen sorketarako taulak . . . . .	58
5.5.4	Bidalketak erregistratzeko taulak . . . . .	59
5.6	LOGa . . . . .	60
<b>6</b>	<b>Garapena</b>	<b>63</b>
6.1	Web Service-a . . . . .	63
6.1.1	Logika komuna . . . . .	64
6.1.2	PeticionTipoUsuario . . . . .	65
6.1.3	PeticionSuscripcion . . . . .	66
6.2	Dokumentuen sorketa . . . . .	68
6.2.1	CL_ABAP_ZIP . . . . .	68
6.2.2	Transformation . . . . .	69
6.3	Dokumentuen bidalketa . . . . .	71
6.3.1	Komunikazio Kanalak . . . . .	71
6.3.2	Sender Agreement . . . . .	72
6.3.3	Receiver Determination . . . . .	72

6.3.4	Interface Determination . . . . .	73
6.3.5	Receiver Agreement . . . . .	73
6.4	Erantzunen prozesamendua . . . . .	74
6.4.1	ZMTP_CL_ACKS_SPLIT . . . . .	74
6.4.2	Operation Mapping . . . . .	76
6.4.3	Integration Process Transformation . . . . .	77
6.4.4	Funtzio Moduluak . . . . .	77
<b>7</b>	<b>Probak</b>	<b>79</b>
7.1	Web Service-aren probak . . . . .	80
7.1.1	Sender-a datu laguntzaile eta email eskaerarik gabe . . . . .	80
7.1.2	Sender-a email eskaerarekin . . . . .	81
7.1.3	Sender-a datu laguntzaileekin . . . . .	82
7.2	Dokumentuen bidalketaren probak . . . . .	83
<b>8</b>	<b>Jarraipena eta Kontrola</b>	<b>85</b>
8.1	Denboraren jarraipena . . . . .	86
8.1.1	Burututako atazak . . . . .	86
8.1.2	Kronograma . . . . .	89
8.2	Komunikazioen jarraipena . . . . .	90
8.2.1	Garatzaile eta enpresaren arteko komunikazioak . . . . .	90
8.2.2	Garatzaile eta METAPOSTAren arteko komunikazioak . . . . .	90
8.2.3	Garatzaile eta tutorearen arteko komunikazioak . . . . .	90
8.3	Kalitatearen jarraipena . . . . .	91
8.3.1	Proiektuaren kalitatea . . . . .	91
8.3.2	Produktuaren kalitatea . . . . .	91
8.4	Arriskuen jarraipena . . . . .	92



---

<b>9 Ondorioak eta etorkizunerako lerroak</b>	<b>93</b>
9.1 Ondorioak . . . . .	93
9.1.1 Proiektuaren inguruko ondorioak . . . . .	94
9.1.2 Ondorio pertsonalak . . . . .	95
9.2 Etorkizunerako lerroak . . . . .	96
<b>Bibliografia</b>	<b>97</b>
<b>Eranskinak</b>	
<b>A Harpidetutako langileen informea</b>	<b>101</b>
A.1 Dictionary . . . . .	102
A.2 Deskribapen teknikoa . . . . .	103
A.3 Prozesuaren logika . . . . .	104
<b>B Harpidetutako bezeroen informea</b>	<b>107</b>
B.1 Dictionary . . . . .	108
B.2 Deskribapen teknikoa . . . . .	109
B.3 Prozesuaren logika . . . . .	110



---

## Irudien aurkibidea

---

2.1	LDE diagrama . . . . .	7
2.2	Kronograma estimatua . . . . .	11
3.1	SAP sistemaren arkitektura . . . . .	18
3.2	SAP moduluak . . . . .	18
3.3	Egoera SAP PI gabe . . . . .	20
3.4	Egoera SAP PIrekin . . . . .	20
3.5	SAP PIko Adaptadore baten adibidea . . . . .	21
3.6	SAP PIko Adaptadore estandarrak . . . . .	22
3.7	SAP PIko arkitektura . . . . .	23
3.8	SAP PIren kronologia . . . . .	24
3.9	SAP PIko hasiera orria . . . . .	25
3.10	Datu fluxua . . . . .	25
3.11	Data Type, Message Type eta Service Interface erlazioa . . . . .	26
3.12	Operation Mapping eta Message Mapping erlazioa . . . . .	26
3.13	Sender eta Receiver Adaptadoreak . . . . .	27
3.14	System Landscape . . . . .	29
3.15	SLren erlazioa ESR eta DIRekin . . . . .	29
5.1	SENDERPeticionTipoUsuarioRequest . . . . .	42

5.2	SENDERPeticionTipoUsuarioResponse . . . . .	42
5.3	SENDERPeticionSuscripcionRequest . . . . .	42
5.4	SENDERPeticionSuscripcionResponse . . . . .	42
5.5	Web Service: Zerbitzu mota . . . . .	43
5.6	Web Service: Funtzio moduluak zehaztu . . . . .	44
5.7	Web Service: Nomenklatura aldaketak . . . . .	44
5.8	Klase laguntzailea . . . . .	45
5.9	Dokumentuen sorketa: ZMTP_REP_NOMINAS . . . . .	46
5.10	Dokumentuen sorketa: generar_nomina() . . . . .	46
5.11	Dokumentuen sorketa: ZMTP_REP_FACTURAS . . . . .	47
5.12	Dokumentuen sorketa: generar_factura() . . . . .	47
5.13	Integration Process . . . . .	52
5.14	IF_MAPPING interfazea . . . . .	53
5.15	ACK fitxategien egitura . . . . .	54
5.16	Datu Baseko taulak . . . . .	55
5.17	Datu Basea: ZMTP_T101 . . . . .	56
5.18	Datu Basea: ZMTP_T201 eta ZMTP_T301 . . . . .	56
5.19	Datu Basea: ZMTP_T203 . . . . .	57
5.20	Datu Basea: ZMTP_T303 . . . . .	58
5.21	Datu Basea: ZMTP_T202 . . . . .	59
5.22	Datu Basea: ZMTP_T501 . . . . .	59
5.23	Datu Basea: ZMTP_T502 . . . . .	60
5.24	LOGa <i>SLG</i> transakzioa . . . . .	60
5.25	LOGa: Mezuak . . . . .	61
6.1	Web Service-ak: parametrizazioa irakurri . . . . .	64

---

6.2	Web Service-ak: Datuak egiaztatu . . . . .	64
6.3	Web Service-ak: <i>PeticionTipoUsuario</i> . . . . .	65
6.4	Web Service-ak: Alta eman . . . . .	66
6.5	Web Service-ak: Baja eman . . . . .	67
6.6	Web Service-ak: Harpidetuen zerrenda . . . . .	67
6.7	Dokumentuen sorketa: <i>CL_ABAP_ZIP</i> . . . . .	68
6.8	Dokumentuen sorketa: Simple Transformation . . . . .	69
6.9	Dokumentuen sorketa: XSLT . . . . .	70
6.10	Dokumentuen sorketa: <i>CALL TRANSFORMATION</i> . . . . .	70
6.11	Dokumentuen bidalketa: <i>Communication Channel</i> . . . . .	71
6.12	Dokumentuen bidalketa: <i>Sender Agreement</i> . . . . .	72
6.13	Dokumentuen bidalketa: <i>Receiver Determination</i> . . . . .	72
6.14	Dokumentuen bidalketa: <i>Interface Determination</i> . . . . .	73
6.15	Dokumentuen bidalketa: <i>Receiver Agreement</i> . . . . .	73
6.16	Erantzunen prozesamendua: Irteerako XMLa definitu . . . . .	74
6.17	Erantzunen prozesamendua: ZIP fitxategia irakurri . . . . .	75
6.18	Erantzunen prozesamendua: ACK fitxategiak irakurri . . . . .	75
6.19	Erantzunen prozesamendua: Emaitzako XMLa sortu . . . . .	76
6.20	Erantzunen prozesamendua: Operation Mapping . . . . .	76
6.21	Erantzunen prozesamendua: Transformation . . . . .	77
6.22	Erantzunen prozesamendua: <i>Function Modules</i> . . . . .	77
6.23	Erantzunen prozesamendua: Funtzio modulu bat . . . . .	78
8.1	Kronograma erreala . . . . .	89
A.1	Langile zerrenda . . . . .	101
A.2	Datu Baseko taulak . . . . .	102

A.3	Datu egiturak eta objektuak . . . . .	102
A.4	Programa nagusia . . . . .	103
A.5	Klasea . . . . .	103
A.6	Datuak lortu . . . . .	104
A.7	Objektuak sortu . . . . .	104
A.8	ALVa sortu . . . . .	105
A.9	Erabiltzailearen datuak . . . . .	105
B.1	Langile zerrenda . . . . .	107
B.2	Datu Baseko taulak . . . . .	108
B.3	Datu egiturak eta objektuak . . . . .	108
B.4	Programa nagusia . . . . .	109
B.5	Klasea . . . . .	109
B.6	Datuak lortu . . . . .	110
B.7	Objektuak sortu . . . . .	110
B.8	ALVa sortu . . . . .	111
B.9	Erabiltzailearen datuak . . . . .	111

---

## Taulen aurkibidea

---

2.1	Atazen estimazioak . . . . .	9
2.2	Arriskuen zerrenda . . . . .	14
4.1	'SENDERPeticionTipoUsuario' erabilpen kasua . . . . .	36
4.2	'SENDERPeticionSuscripcion' erabilpen kasua . . . . .	36
4.3	'Dokumentuak sortu' erabilpen kasua . . . . .	37
4.4	'Dokumentuak bidali' erabilpen kasua . . . . .	38
4.5	'Erantzunak jaso' erabilpen kasua . . . . .	38
4.6	'Erantzunak prozesatu' erabilpen kasua . . . . .	39
7.1	Probak: Datu laguntzaile eta email eskaerarik gabe . . . . .	80
7.2	Probak: Email eskaerarekin . . . . .	81
7.3	Probak: Datu laguntzaileekin . . . . .	82
7.4	Probak: Dokumentuen bidalketak . . . . .	83
8.1	Atazen esfortzu erreala . . . . .	86





# 1. KAPITULUA

---

## Sarrera

---

Hurrengo lerroetan, proiektu honen sustatzaileak diren *Oreka IT* eta *METAPOSTA* enpresei buruzko deskribapenak agertuko dira, hauek izango baitira **SAP-METAPOSTA konektorea** baliatuko dutenak.

### 1.1 Oreka IT

Oreka Information Technologies S.L. SAP aholkularitzan diharduen enpresa da, Gasteizen finkatua eta 2009an sortua. SAPen partner ofiziala den zerbitzu enpresa honetan, giza baliabide (SAP HCM) eta Business Intelligence moduluetan eta aholkularitza teknologikoan (Netweaver plataforma, sistemen kudeaketa, etab.) espeliazitatuta daude eta bertan oinarritzen da bere lan gehiena.

Sortu zenetik bere lan esparrua handituz doa Oreka IT, eta, honela, hasieran 3 bazkideek osatzen zuten enpresa hau, orain, 30 profesionaletik gorako talde batek osatzen du. Handitze horretan, gero eta SAPen baliabide gehiago erabiltzeaz gain, beste enpresekin ere elkarlanean aritu da, proiektu amankomunak sortuz.

Puntu horretan koka daiteke proiektu hau, bi enpresen arteko elkarlanean hain zuzen ere, SAP teknologia beste sistema desberdin batekin konektatzea helburu duena.

## 1.2 METAPOSTA

METAPOSTA "Euskadi Informazioaren gizartean" planaren barruan dagoen proiektua bat da, eta bere funtsa *Interneten postako dokumentu-gordailu elektronikoa* izatea da, non, norberaren intereserako dokumentuak jaso eta biltegitzen diren.

Sistema honen oinarri nagusia herritar zein enpresentzat balio handia duten dokumentuak modu seguruan eta iraunkorrean biltegitzea da, espazio pertsonal elektronikoa batean.

METAPOSTA, beraz, posta klasikoaren alternatiba bat da, eta orain arteko postontzi tradizional baten lana egingo du, baina mundu elektronikoa. Gauzak honela, sistema honen bitartez jasoko diren dokumentuak postontzi fisikoan jasotzen diren berdinak dira: fakturak, banku-laburpenak, frogarriak, polizak, kontratuak, zitazioak, jakinarazpenak, nominak...

Horren osagarri, erabiltzailearen ordenagailutik dokumentuak kargatzeko prestazioak eskainiko ditu METAPOSTAk, dokumentu horiek digitalki eta modu seguruan biltegitzeko.

## 1.3 Motibazioa

Gaur egun, bi zerbitzu hauek (SAP eta METAPOSTA) erabiltzen dituzten enpresek, fakturak, nominak eta abar SAPetik sortu, fitxategietan gorde eta, gero, honekin lan egiten duten bezeroei, postontzi elektronikora bidali behar dizkiete. Beraz, suposa daitekeen bezala, langile askoko enpresetako nominak -edo bezero askoko enpresetako fakturak- METAPOSTAra bidaltzeko lana handia izaten da hilabetero.

Hau ikusita, bi enpresek elkarlanean aritzeko aukera ona ikusi dute, haien eskaintza handituz bezero amankomun gehiago lortzeko helburuarekin, goraka doan interneteko postaren esparru honetan. Elkarrekintza hau aurrera ateratzeko, proiektu honen helburu den **SAP-METAPOSTA konektorea** sortu nahi da.

Konektore hau egiteko, SAP ingurunea erabiliko da, honek baliatzen duen ABAP lengoia erabiliko delarik kodea idazteko. Honetaz gain, eta aurrerago zehaztuko den moduan, beste teknologia zein baliabide desberdinak ere erabiliko dira behar izatekotan.

## 2. KAPITULUA

---

### Proiektuaren Helburu Dokumentua

---

Kapitulu honetan, Proiektuaren Helburu Dokumentua azaltzen da. Hau, proiektuaren beraren zehaztapena da, non, alde batetik, irismena finkatu, eta, bestetik, irismen hori betetzeko jarraituko den plangintza azalduko diren. Plangintza honetan, proiektuaren komunikazio, kalitate eta arriskuen planak azalduko dira.

#### 2.1 Helburua

Proiektu honen helburua, SAP kudeaketarako ERPa erabiltzen duten enpresek sortzen dituzten agiriak (nominak, fakturak, eta abar) zuzenean METAPOSTAra bidaltzeko aukera emango duen lokailu bat garatzea da. Honela, eta ezer gehiago egin behar izan gabe, agiriak sortu eta postontzi elektronikora bidaliko zaizkie zuzenean hauek jaso behar dituzten erabiltzaileei. Gainera, konektorea, ahal bezain orokorra izan behar du, enpresa desberdinetan inplantatu nahi bait da, bakoitzean konfigurazio handiegirik egin gabe. Hau guztia egiteko, SAP sistemak dituen hainbat baliabide erabiliko dira, ABAP programazio lengoia tresna nagusia izango delarik.

Beste alde batetik, eta modu pertsonalean, proiektu hau aurrera ateratzearen nire helburu pertsonal nagusia karreran zehar ikasitakoak mundu profesionaleko proiektu erreal batean praktikan jartzea da. Honelako proiektu konplexua aurrera ateratzeko erronkak ere motibazioa handia suposatzen du.

## 2.2 Lan metodologia

Proiektu hau enpresa batean egingo dela jakinda, honek dakartzan berezitasun guztiakin, eta beste enpresa batekin elkarlanean egingo dela ere kontuan hartuz, lan metodologia zerbait berezia izango da. Alde batetik, lana Oreka IT enpresan egingo delako, METAPOSTA bezero bezala izanda, haiek izango direlarik ezkakizun eta murriztapenak ezarriko dituztenak. Eta, bestetik, konektorearen konplexutasunagatik eta eskainiko dituen aukera guztiengatik, ez baita proiektu honetan guztiz amaituko, etorkizunerako lana ere geratuko delarik.

Gauzak honela, proiektu hau aurrera ateratzeko espiraleko modelo ebolutiboa aukeratu da, zeinetan bezeroen menpe egongo den fase batetik bestera igarotzea. Gainera, aipatutako berezitasunak direla eta, fase desberdinak tartekatzeak aukera ere badago, ezin bait da ziurtatu planifikazio, analisi, diseinu, garapen eta frogak modu guztiz independentean egingo direnik.

## 2.3 Irismena

Atal honetan, aurrekoan aipatutako helburua zehaztuko da, irismen minimoa finkatu eta zabalkuntza planak ere zehaztuz. Gainera, emangarriak, bezeroek zehaztutako murriztapenak eta proiektua bera osatzen duten atazak ere zehaztuko dira.

### 2.3.1 Irismen minimoa

Hauek dira aurretik aipatutako helburua betetzeko sortuko den konektoreak eskaini behar dituen minimoak:

- **Erabiltzaileak kudeatzeko Web Service-a:** METAPOSTA erabili nahi duen enpresa orok, datu-bidaltzaile (SENDER hemendik aurrera) moduan erregistratu behar da, eta Web Service bat eskaini behar du bezeroek posta zerbitzuarekin harpidetza kudeatzeko. Honela, METAPOSTAko erabiltzaile batek enpresa konkretu baten dokumentuak jaso nahi dituzenean, honetara harpidetu beharko da. Harpidetza hau egiteko, konektoreak Web Service bat sortu eta, bertan, bi zerbitzu eskaini beharko ditu, erabiltzaileak identifikatzeko bat, eta altak/bajak emateko, bestea.

- **Dokumentuen sorketa:** Konektoreak, enpresa guztietara moldatutako dokumentuen (faktura, nomina, ...) sorketarako sistema bat izan behar du, zeienek konfigurazio sinple baten ondoren, posta zerbitzuaren erabiltzaile guztien dokumentuak lortu eta konprimitu beharko dituen bidali ahal izateko.
- **Dokumentuen bidalketa:** Dokumentuak sortuta daudela, hauek METAPOSTAra bidali beharko dira, erabiltzaileek bertatik ikus ditzaten. Horretarako, sFTP teknologia erabiliko da bidalketak beti modu seguruan egin ahal izateko.

Ikusten den bezala, konektore honek bi atal argi izango ditu, alde batetik, bezeroak harpidetu ahal izateko sistema bat, eta, bestetik, harpidetutako erabiltzaile hauen dokumentuak sortu eta bidaliko dituen sistema bat.

Bi sistema hauek, aurretik aipatu den bezala, oso konfiguragarriak izan behar dira, konektorea enpresa desberdinetan ezartzeko gai izan behar delako, enpresa bakoitzean aldaketarik egin gabe (edo ahalik eta aldaketa eta moldaketa gutxien eginda).

### 2.3.2 Zabalkuntza plana

Lehenago aipatu den moduan, sortu nahi den konektorea ezin izango da proiektu honetan zehar guztiz bukatu, eta, helburua, funtzionala den eta baldintza minimoak betetzen dituen konektore bat egitea da, hortik aurrerako guztiak, hobekuntzak eta funtzionalitate berriak, etorkizunerako lan bezala zehaztuta geratzen direlarik.

Hala ere, horietako batzuk zehaztuko dira hemen, denbora izatekotan, hoiek izango direlako proiektu honen zabalkuntza planak.

1. **Eskuliburua:** Eskuliburu bat, zeinetan azaltzen den konektorea martxan jarri eta erabiltzen jarraitu beharreko pausu guztiak eta erabili beharreko transakzio, programa, taulak, ...
2. **Harpidetuen informea:** Harpidetutako erabiltzaileen zerrenda ikusteko informe bat, enpresa bakoitzak jakin dezan zeintzuk erabiltzaile dauden harpidetuta eta noiztik.
3. **SAP menua:** SAP sistemarako menu bat, zeinek erabili beharko diren SAPeko transakzio, etab. biltzen dituen, konektorearen erabilpena errazteko.

4. **EDITRAN teknologiaren erabilera:** Fitxategien bidalketa gauzatzeko sFTP erabiltzeaz gain, EDITRAN sistema erabilgarri huztea erabiltzaile bakoitzak erabaki dezan zeinekin lan egin nahi duen.

### 2.3.3 Gogobetetzea bermatzeko argibideak

Atal honetan, bezeroek zehaztu dituzten proiektuan zehar jarraitu beharreko argibide batzuk biltzen dira. Hauek, bi multzotan banatu dira, Oreka IT enpresarenak alde batetik, eta METAPOSTArenak bestetik.

- *Oreka IT:*

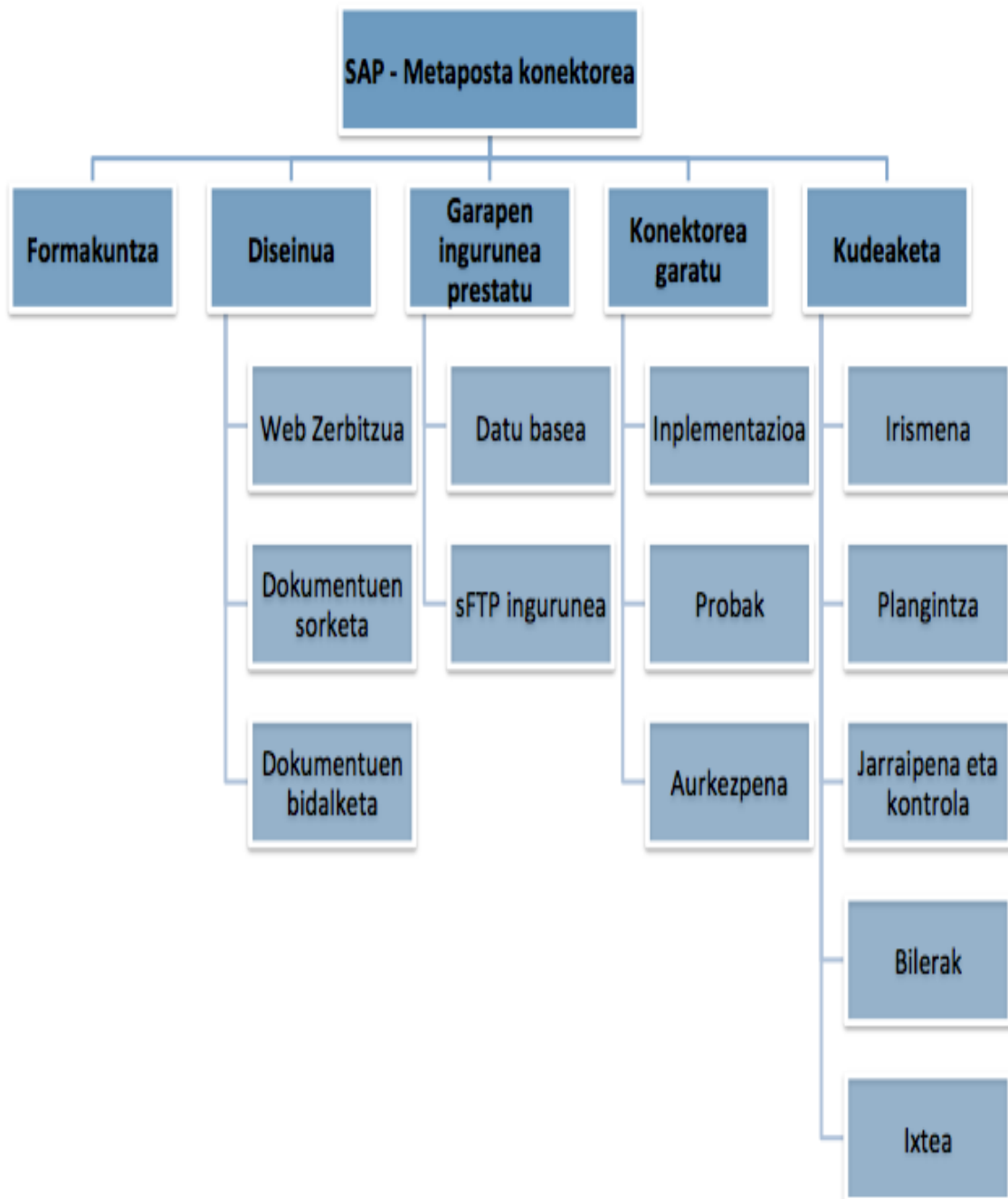
1. Programatzen den guztia ZMTP aurrizkia eramango du (paketeak, taulak, funtsioak, etab.).
2. Log bat gorde behar da konektorean gertatzen den guztiarekin.
3. Beste hizkuntza batera itzultzeko prest egon behar da konektorea, horretarako, testu guztiak testu estandar modura gorde behar dira.
4. Konektorea sistema eragileekiko independentea izan behar du, bai SAP zerbitzari dagokionez, eta, baita, erabiltzaileen ordenagailuei dagokienez era.

- *METAPOSTA:*

1. Segurtasuna bermatu behar da momentu guztietan, bidaliko diren dokumentuak pertsonalak bait dira.
2. Bidalketak kontrolatu behar dira, bidaltzen dena ondo jaso dela zihurtatuz.
3. Konektoreak SENDERek pasatzen dituzten proba guztiak behin pasatu behar ko ditu, enpresa bakoitzean ezartzean, honek, proba gutxi batzuk baino egin behar ez izateko.
4. Terminologia guztia, METAPOSTAk erabiltzen duen estandarra izan behar du. Horretarako, estilo gida bat ere dagoelarik.

### 2.3.4 LDE diagrama

Hurrengo irudian, proiektu honi dagokion lanaren deskonposaketa diagrama aurkitu daiteke (LDE):



**2.1 Irudia:** LDE diagrama

### 2.3.5 Atazen deskribapena

Atal honetan, aurreko atalean definitutako LDE diagrama osatzen duten atazak azalduko dira, bakoitzaren nondik-norakoak aipatuz.

- **Formakuntza**

Ataza honek egin beharreko ikasketa guztiak biltzen ditu. Hauetako gehienak teknologikoak izango dira, SAP, METAPOSTA, eta honelakoekin lotuta, baina, proiektu hau garatzeko ere formakuntza beharko da, memoria edota aurkezpenak egiteko, adibidez.

- **Diseinua**

Konektoreak izango dituen elementu guztiak definituko dira, hauek nolakoak izango diren eta zer egingo duten zehaztuz. Honela, ataza honetan definitu beharko da “Garapen ingurunea prestatu” atazan sortutako elementuak eta “Konektorea garatu” atazan inplementatutako programak nola lan egingo duten.

- **Garapen ingurunea prestatu**

Ataza honetan, batez ere, datuen kudeaketarekin lotutako lan guztiak biltzen dira, hala nola, proiektuaren garapenean zehar erabiliko diren taulak, datu elementuak, domeinuak, klaseak, eta abarrak sortu eta mantentzea. Gainera, dokumentuen biddalketarako erabiliko den sistema ere prestatu beharko da.

- **Konektorea garatu**

Konektorea garatzearekin zuzenean loturiko atazak daude hemen. Ataza honetan, garrantzitsuenak, konektorearen beraren inplementazioa eta hau ondo funtzionatzen duela ziurtatzeko egingo diren probak dira. Gainera, aurkezpenean erabiliko diren proiektuaren irudiak/diapositibak ere ataza honetan egingo dira.

- **Kudeaketa**

Proiektuaren prozesu taktikoak dira hauek:

- Irismena: Proiektuan zehar egingo direnak zehaztu.
- Plangintza: Irismenean zehaztutakoak nola egingo diren azaldu.
- Jarraipena eta Kontrola: Planifikatutakoa egiten ari dela zihurtatu eta nola egiten ari den kontrolatu.
- Ixtea: Proiektua amaitzearekin erlazionatutakoak, azken entregak, aurkezpena, ...



## 2.4 Plangintza

Orain, plangintzarekin erlazionatutako alderdiak aztertuko dira: burutu beharreko atazak, hauek burutzeko beharrezko denbora, komunikazio plana, kalitatea bermatzeko plana eta arriskuen deskribapen zein kontingentzia planak.

### 2.4.1 Burutu beharreko atazak

Hurrengo taulan, proiektu hau osatzen duten atazen denbora estimazioak agertzen dira, horretarako, hauen hasiera eta amaiera data estimatuak agertzen direlarik.

<b>ATAZA</b>	<b>HASIERA</b>	<b>AMAIERA</b>
<b>Formakuntza</b>	<b>2013-12-02</b>	<b>2013-12-20</b>
<b>Diseinua</b>	<b>2013-12-02</b>	<b>2013-12-20</b>
Web Zerbitzua	2013-12-02	2013-12-13
Dokumentuen sorketa	2013-12-09	2013-12-20
Dokumentuen bidalketa	2013-12-09	2013-12-20
<b>Garapen ingurunea prestatu</b>	<b>2013-01-07</b>	<b>2014-01-24</b>
Datu basea	2013-01-07	2014-01-24
sFTP ingurunea	2013-01-07	2014-01-24
<b>Konektorea garatu</b>	<b>2014-01-07</b>	<b>2014-02-21</b>
Inplementazioa	2014-01-07	2014-02-14
Probak	2014-01-13	2014-02-14
Arriskueterako denbora	2014-02-17	2014-02-20
Aurkezpena	2014-02-21	2014-02-21
<b>Kudeaketa</b>	<b>2013-12-02</b>	<b>2014-02-21</b>
Irismena	2013-12-02	2013-12-13
Plangintza	2013-12-02	2013-12-13
Jarraipena eta kontrola	2013-12-16	2014-02-21
Bilerak	2013-12-02	2014-02-21
Ixtea	2014-02-17	2014-02-21
<b>TOTALA</b>	<b>2013-12-02</b>	<b>2014-02-21</b>

**2.1 Taula:** Atazen estimazioak

Datu hauek azaltzen hasi baino lehen, aipatu behar da, ikasleak eguneko 7 orduko lana egingo duela enpresan, batez ere garapenarekin erlazionatutako atazak burutzeko, eta, beraz, eguneroko lan-karga handia izango dela. Honekin erlazionatuta, ikaslea ez da Oreka IT enpresara joango 2013-12-23 eta 2014-01-07 data bitartean, egun horietan, beraz, lan gutxiago egingo duelarik.

Aurreko taulan ikusten den bezala, proiektuaren hasieran bertan Formakuntza, Diseinua eta Kudeaketa atazak hasiko dira. Honela, lehenengo hilabetea proiektuaren planifikazio eta diseinurako erabiliko da gehien bat. Garapen ingurunearen prestaketa ere hasiko delarik. Denbora honetan, garrantzitsuena, proiektuaren betekizunak ondo aztertu eta hauek aurrera ateratzeko egin beharrekoak ondo planifikatu/diseinatzea izango da.

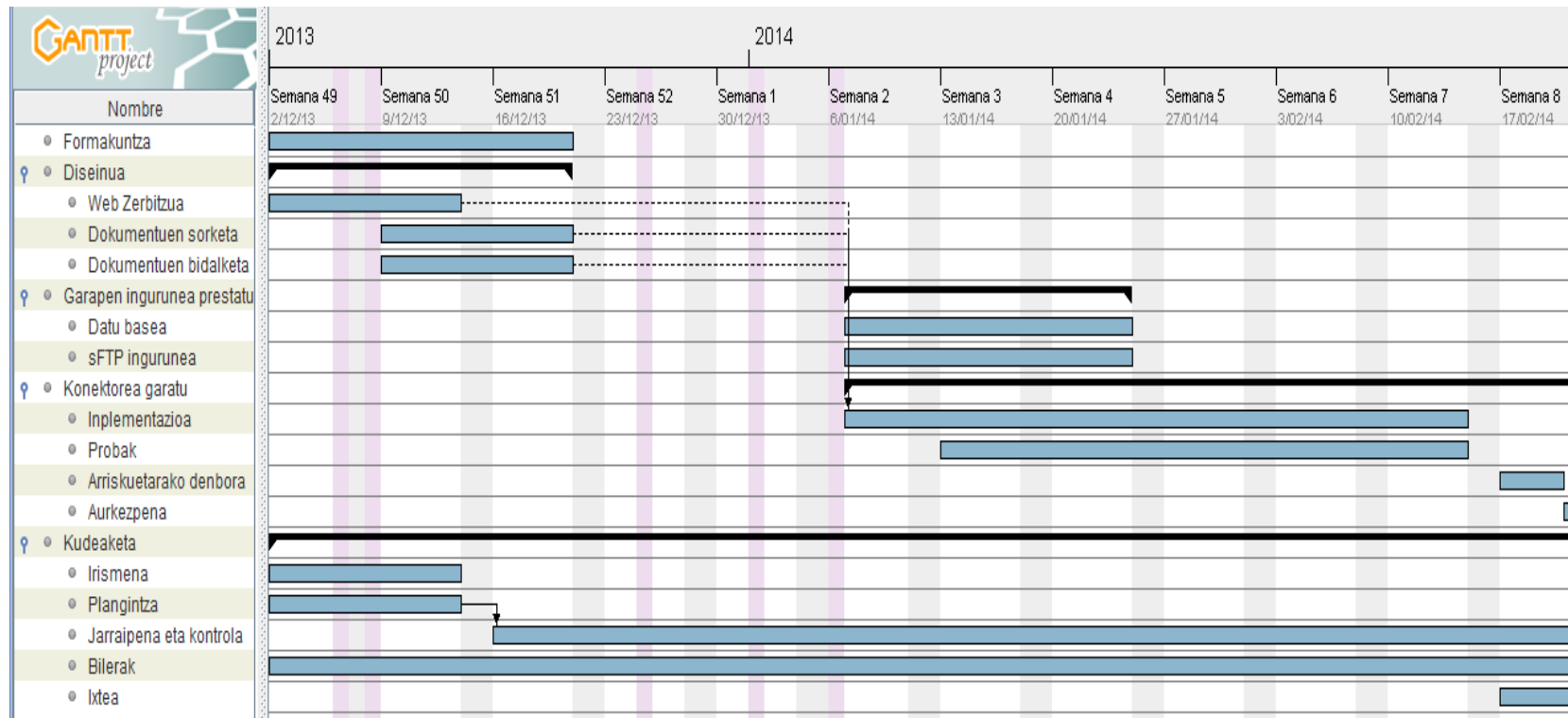
Ataza hauek amaituta, eta, gabonetako oporrak eta gero, proiektuaren garapenari ekingo zaio, diseinatutakoa inplementatuz. Bigarren fase honetan, inplementazioa izango da garrantzitsuena, datu basea zein sFTP ingurunea ere landuko direlarik hasieran. Azken hauetan, prestatutako garapen ingurunea mantentzea izango da lana, inplementatu ahala, datu basean zein fitxategien bidalketarako sisteman aldaketak suertatuko bait dira.

Inplementazioarekin bukatutakoan eta probak ondo ateratzen direla ikusten denean, arrisku-kuetarako denborarekin zer egingo den aztertuko da enpresako zuzendariarekin. Honela, denbora horretan zabalkuntzak edota hobekuntzak egingo dira. Proiekturako denbora gehiago behar izatekotan, ordea, denbora marjina baliatuko da hauek bukatu ahal izateko.

Taulan ikusten den bezala, ere, Kudeaketa lanak proiektu osoan zehar gauzatuko dira, proiektuaren Jarraipen zein Kontrolak, hasieratik amaierara egin behar dira eta.

### 2.4.2 Kronograma

Aurreko atalean, proiektua osatzen duten atazen hasiera eta amaiera datak agertzen dira, hauen iraupena eta deskonposaketak zehaztuz. Orain, ataza horiek Gantt diagrama batean adieraziko dira, modu grafiko batean aztertu eta haien arteko dependentziak hobeto ikusteko.



2.2 Irudia: Kronograma estimatua

### 2.4.3 Komunikazio plana

Komunikazio planean, proiektua aurrera eramateko ezinbestekoak diren komunikazioak nola gauzatuko diren deskribatuko da. Hauek, hiru multzo desberdinetan bananduko dira, hauen jatorri eta helburuen arabera.

Honela, garatzaile eta enpresaren arteko komunikazioak, garatzailearen METAPOSTA-  
rekiko komunikazioak eta garatzaile eta tutorearen artekoak ditugu. Hurrengo lerroetan, hauek nola gauzatuko diren azalduko da.

- **Garatzaile eta enpresaren arteko komunikazioak:** Proiektu hau enpresa batean egingo denez, garatzaileak enpresako bere gainbegiralearekin komunikazio zuzena izan behar du, hura izango bait da, proiektuaren garapenerako argibide eta bete-beharrak emango dizkiona.

Erabaki da lan egiteko modu eraginkorrena, ikaslea enpresan bertan lan egitea dela, honela, gainbegiralearekin erlazio zuzena izan dezake, momentuoro arazoak eta zalantzak komentatu ahal izateko. Honela, hauen bien arteko komunikazio gehienak, zuzenean egingo dira, horretarako bilerak ere egingo direlarik. Gainera, aurrez-aurreko komunikazioa burutu ezin daitekeenerako, enpresaren beraren posta elektronikoa erabiliko da komunikazio guztiak burutzeko.

- **Garatzaile eta METAPOSTAren arteko komunikazioak:** Proiektu honetan, aipatu den moduan, METAPOSTAren arteko komunikazioak lan egingo da eta, beraz, hauekin ere komunikazioa mantendu beharko da proiektuaren garapena egokia izan dadin. Horretarako, posta elektronikoa erabiliko da, hau izango delarik, printzipioz, komunikazio kanal bakarra. Hala ere, behar izatekotan, bilerak egiteko aukera ere egongo da, enpresen arteko beharren/nahien arabera, betiere.
- **Garatzaile eta tutorearen arteko komunikazioak:** Bestalde, proiektu hau, noski, unibertsitateko tutore batek ere jarraituko du, eta, beraz, garatzaileak tutoreari, proiektuaren garapenaren nondik-norakoan berri emango dio. Komunikazio hauen kanala ere posta elektronikoa izango da, garatzaileak tutoreari zalantzak/arazoak komentatu eta baita, proiektuaren beraren bertsio desberdinak erakutsi dizkio, hauek komentatu eta lanean jarraitu ahal izateko. Beharrezkoa izatekotan, interneteko bideo-deiak baliatuko dira(Skype, Google Hangouts), bilerak ere egin daitezkeelarik.

#### 2.4.4 Kalitate plana

Kalitate planean, proiektu zein produktuaren kalitateak bermatzeko jarraituko diren estrategiak azalduko dira. Horretarako, zer neurtuko den azalduko da, nola egingo den ere azalduz.

- **Proiektuaren kalitatea:** Proiektuaren kalitatearen adierazle nagusia, plangintza hau bera da, beraz, hemen definitutako guztiak jarraitu beharko dira proiektuaren kalitatea bermatzeko.

Honela, atal honetan definitu diren Komunikazio, Arrisku eta Aldaketa planak jarraitu beharko dira. Hau guztia, noski, epeak betetzen direla egin behar da, hau, kalitatearen adierazle oso argia baita.

- **Produktuaren kalitatea:** Produktuaren kalitatea bermatzeko, bi neurgailu desberdin erabiliko dira.

Alde batetik, produktua bera software bat denez, honen garapena modu egokian funtzionatzen duela zihurtatuko da probak eginez. Honela, produktuak irismenean definitutakoak betetzen dituela bermatu beharko da, modu eraginkorrean betetzen dituelarik.

Beste alde batetik, produktuaren hartzaileak diren Oreaka IT eta METAPOSTA erabiliko dira kalitate-neurgailu bezala, hauek izango baitira, azkenean, produktua erabiliko dutenak eta honen nondik-norakoak zehaztuko dituztenak. Hau egiteko, beraz, Komunikazio planean definitutakoak baliatu beharko dira, bezeroekin komunikazio on bat mantendu eta haien *feedback*-a jasotzeko.

### 2.4.5 Arriskuen plana

Atal honetan, proiektu honen garapenean gerta litezkeen arriskuak, eta arrisku hauek proiektu zein produktuaren funtzionamenduan eragin ditzaketen ondorioak azaltzen dira. Arrisku bakoitzeko, bere deskribapenarekin batera, aurrezaintza eta kontingentzia planak definituko dira, aipatutako ondorioak ere zehaztuz.

Hurrengo taulan arriskuak zerrendatu dira, hauen probabilitate, eragin eta larritasun mailak zehaztuz.

ID	Arriskua	Prob.	Eragina	Larritasuna
A1	Denbora estimazio okerrak	%50	Negatiboa/positiboa	Ertaina
A2	Lan ingurune ezezaguna	%75	Negatiboa	Txikia
A3	Bezeroen eskakizunen aldaketa	%70	Negatiboa	Ertaina
A4	Behin behineko baja	%20	Negatiboa	Altua
A5	Datuak galtzea	%30	Negatiboa	Ertaina
A6	Epe aldaketak	%20	Negatiboa	Txikia

### 2.2 Taula: Arriskuen zerrenda

- 1. Denbora estimazio okerrak:** Planifikatuta denborak errealitatearekin bat ez etortzea, estimatutakoa baino gehiago edo gutxiago behar delako ataza bat burutzeko.  
Aurrezaintza plana: Esperientzia baliatu eta aurreko proiektuen estimazioak eredu-tzat hartu.  
Kontingentzia plana: Esfortzua handitu beharko da eta beharrezkoa bada, horretarako denbora marjina baliatu.  
Ondorioak: proiektuaren birplanifikazioa egin behar izatea edo epeak ez betetzea.
- 2. Lan ingurune ezezaguna:** Proiektu hau SAP teknologia erabiliz egingo da, eta, garatzaileak honen ezagutza duen arren, ezezagunak dituen baliabide berriak erabiltzera behartuta egoteko arrisku handia dago.  
Aurrezaintza plana: erabili beharko diren baliabideak zehaztu eta ahal diren guztiak prestatu lanean hasi baino lehen.  
Kontingentzia plana: lan ingurunea ezagutzen duen enpresa batean lan eginda, adituak eskura ditu ikasleak, hauei laguntza eskatzeko.  
Ondorioak: ataza bat atzera daiteke, besteak ere atzeratuz eta proiektuaren epeak ez betez.

3. **Bezeroen eskakizunen aldaketa:** Bezeroek ideiaz alda dezakete, proiektuaren garapenean zehar espero ez diren emaitzak edota betekizun berriak suerta daitezkeelarik.  
Aurrezaintza plana: aldaketak proposatu eta aurrera eramateko epeak definitu dira, denbora tarte bat utziz kasu hauetan baliatzeko.  
Kontingentzia plana: hurrengo atalean definitutako aldaketa plana jarraitu.  
Ondorioak: birplanifikazioak gerta daitezke, epeak ez betetzea edo, epeak bete-tzeagatik, produktuaren kalitatea jaistea.
4. **Behin behineko baja:** Proiektu honetako partaide baten baja, proiektutik kanpoko arazoengatik, bere lana ezin betez.  
Aurrezaintza plana: denbora tarte bat utzi da beharrezkoa izatekotan baliatzeko.  
Kontingentzia plana: partaide horrek abisua emango du eta besteez, berarekin erla-zionatuta ez dauden atazak egingo dituzte, denbora ez galtzeko.  
Ondorioak: atzerapenak gerta daitezke, epeak ez betez.
5. **Datuak galtzea:** Proiektuarekin erlazionatutako datuak galtzea, bai proiektuaren kudeaketarekin edota produktuarekin berarekin erlazionatutakoak.  
Aurrezaintza plana: produktuaren segurtasun kopiak egin eta proiektuaren kudea-ketarekin erlazionatutako datuak Google Drive tresna erabiliz gorde.  
Kontingentzia plana: segurtasun kopiak baliatu galdutako datu kopurua minimoa izan dadin.  
Ondorioak: atzerapenak gerta daitezke, datuen arabera.
6. **Epe aldaketak:** Plangintza honetan definitutako entrega epeak aldatzea, aurreratuz zein atzeratuz.  
Aurrezaintza plana: entrega data baino egun batzuk lehenago bukatu ataza guztiak, denbora marjin bat utzi hauetarako.  
Kontingentzia plana: denbora marjina erabili.  
Ondorioak: epe aurreratzeak planifikazioa alda dezakete, atzerapenak ez luke on-dorio handirik eragin behar, ataza eginda geratuko baita nahiz eta ez entregatu.

### 2.4.6 Aldaketen plana

Atal honetan, proiektuan gerta daitezkeen aldaketen aurrean hartu beharreko jarrera azalduko da, hauek nola kudeatuko diren zehaztuz. Garrantzitsua da gogoratzea proiektu hau enpresa batez zuzentzen duela eta, beraz, bertako arduraduna izango dela aldaketa guztien inguruko erabakiak kontrolatuko dituena.

Aldaketa bakoitzeko, honako pausuak jarraituko dira:

1. *Identikatu*: aldaketa bakoitza identifikatu behar da, nondik datorren ikusi (bezero batetik, proiektuaren barnetik, ...) eta honen nondik-norakoak zehaztu. Esfortzuan izango duen eragina ere estimatuko da pausu honetan.
2. *Aztertu*: aldaketa identifikatuta eta haren eragina estimatuta, ondo aztertu behar da zeintzuk izango diren bere eraginak proiektuan, positibo zein negatiboak.
3. *Erabaki*: azterketa eginda, erabaki beharko da aldaketa hori aurrera eramaten den edo ez. Horretarako, bi baldintza nagusi bete behar dira. Alde batetik, ezarritako denbora marjinak bete behar ditu eta epeak errespetatu. Bestetik, proiektu honen partaide guztiek onartu behar dute, enpresako gainbegiralea, tutorea eta garatzailea bera; enpresak azken hitza izango duelarik.
4. *Aurrera eraman*: aldaketa aurrera eramatea erabaki bada, aldaketa inplementatuko da.



## 3. KAPITULUA

---

### Oinarri Tekniko-Teknologikoak

---

Gaur egun, gero eta enpresa gehiagok erabakitzen du bere negozio prozesuak software baten bidez kudeatzea. Software hauei ERP (ingelesezko Enterprise Resource Planning) deritzo, eta, enpresa baten produkzio, logistika, kontabilitate, faktura eta abarrak kontrola ahalbidetzen dute. Sistema hauen artean ospetsuena SAP ERPa da.

Aurretik aipatu den bezala, proiektu honen helburu den Konektorea SAP sistemarako egingo da, horretarako, honek eskaintzen duen ABAP programazio lengoiaia baliatuko delarik. Gainera, SAP sistemaren beste hainbat moduluk ere parte hartuko dute garapenean.

Hurrengo lerroetan, ERP honen ezaugarri nagusiak eta proiektu honetan garrantzitsuak izango diren moduluak aztertuko dira, hurrengo kapituluetan agertuko diren kontzeptuak hobeto uler daitezten.

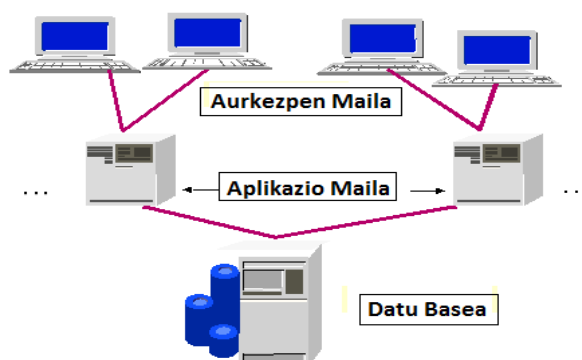
#### 3.1 SAP

SAP (System, Applications and Products), Alemanian 1972an sortua, enpresentzako software hornitzaile handiena da munduan. Bere SAP R/3 ERPa oso arrakastatsua izan da, batez ere, enpresa bakoitzera moldatzeko duen erraztasunagatik.

SAP R/3 sistema, irekia eta moldagarria izateaz gain, etengabeko eguneraketak jasaten ditu momentuko beharretara egokitzeko. Honela, jende asko dago sistema honekin lanean eta eskaintzen dituen aukerak amaigabeak dira.

### 3.1.1 SAP Arkitektura

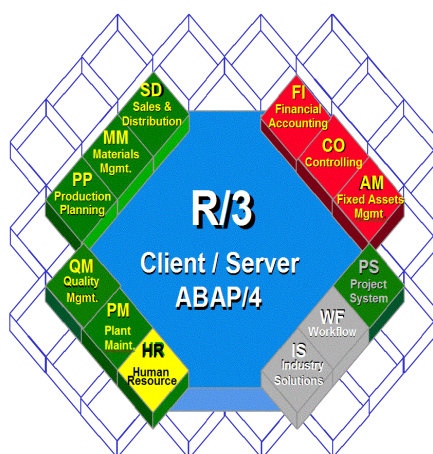
Software honek, bezero/zerbitzari egitura erabiltzen du, 3 mailatako arkitektura baliatuz, hiru maila hauek honela banatzen direlarik: Datu Baseen maila, datuak gordetzeko, Aplikazio maila, gordetako datuekin lan egiten duten programak biltzeko eta Aurkezpen maila, aurreko guztia modu erosoan erabiltzeko.



**3.1 Irudia:** SAP sistemaren arkitektura

### 3.1.2 SAP Moduluak

SAP sistema moduluetan banatuta dago, bakoitza enpresaren arlo desberdin bat modu independentean kudeatzeko erabilia. Honela, lau modulu nagusi desberdintzen ditugu, hurrengo irudian ikusiko ditugunak.



**3.2 Irudia:** SAP moduluak

Irudian ikusten den bezala, lau modulu nagusi ditugu, bakoitza kolore batekin identifikatua: Finantza modulua (gastuak zein etekinak kontrolatzeko), Logistika modulua (produktuen bizi-zikloa kudeatzeko), Giza Baliabideen modulua (nominak eta enpresaren organizazioa kudeatzeko) eta Funtzio multiaplikazioak (plangintza, workflow-a, dokumentuak eta abar kudeatzeko).

## 3.2 ABAP

ABAP (Advanced Business Application Programming) laugarren belaunaldiko programazio lengoia da, SAP enpresak sortua. Lengoaia hau da, Java-rekin batera, SAP sistema osoan erabiltzen dena modulu guztiak garatzeko.

ABAP/4 deritzen lengoia honen bidez, aplikazio estandarrak ulertu/aldatu eta berriak sor daitezke, enpresen beharretara zuzenduta.

Programa berriak sortzeko erabil daitezkeen hainbat tresna desberdin daude, hurrengo lerroetan, proiektu honetan erabiliko direnak agertzen direlarik:

- *ABAP4 editorea (SE38)*: Report-ak sortzeko
- *Funtzioen liburutegia (SE37)*: Funtzio moduluak sortzeko
- *Objektu nabigatzailea (SE80)*: Sortzen den edozer (funtzio, report, modulepool, transakzio, ...) aztertzeko
- *Klase editorea(SE24)*: Objektuera zuzendutako ABAP OO lengoian klaseak sortzeko
- *Datu basearen hiztegia(SE11)*: Taula, egitura eta datu motak definitzeko

Hauetaz gain, ABAP lengoia Open SQL sententziak erabiltzen ditu datu base desberdinekin lan egiteko, eta, RFC (Remote Function Calls) konexioak onartzen ditu, SAP eta beste sistema desberdinen elkarlana ahalbidetzeko. Azken aukera hau oso garrantzitsua da proiektu honetan, **SAP - METAPOSTA Konektorea** teknologia hau baliatuko baitu sistema desberdinek datuak elkar trukatu ahal izateko.

ABAP izango da, beraz, proiektu honetan erabiliko den programazio lengoia inplementazio guztietarako.

### 3.3 SAP PI

#### 3.3.1 Zer da SAP PI?

SAP PI (Process Integrator), sistema desberdinen arteko komunikazioan sortzen den informazioa modu zentralizatu batean kontrolatzen duen erreminta da. Honen bitartez, SAP sistemak elkarrekin, edota beste sistema desberdinekin, informazioa truka dezakete, PI sistema *middleware* moduan erabilia. Tresna honen beharra nabaria izaten da sistema askoko ingurunetan, adibidez, datu basea, ERPa, posta zerbitzaria, etab. duten enpreetan.



**3.3 Irudia:** Egoera SAP PI gabe



**3.4 Irudia:** Egoera SAP PIrekin

Aurreko irudietan ikusten den bezala, konplexutasun handiko erakunde batean, sistema desberdin askorekin lan egiten da, hauetako bakoitzak helburu eta lan egiteko forma desberdinak dituelarik. Gauzak honela, informazio trukea oso konplexua bilakatzen da, kontrolatzeko arazo asko ager daitekeelarik.

Arazo hauei aurre egiteko, eta bigarren irudian agertzen den moduan, SAP PI sistema ezarri eta honen bitartez egiten dira komunikazio guztiak, monitorizazioa erraztuz eta informazio trukeetarako puntu bakarra definituz.

Proiektu honetan, trena hau fitxategien bidalketarako erabiliko da, dokumentuen trukea modu seguruan eta kontrolatuan egin ahal izateko.

## Estandarrak

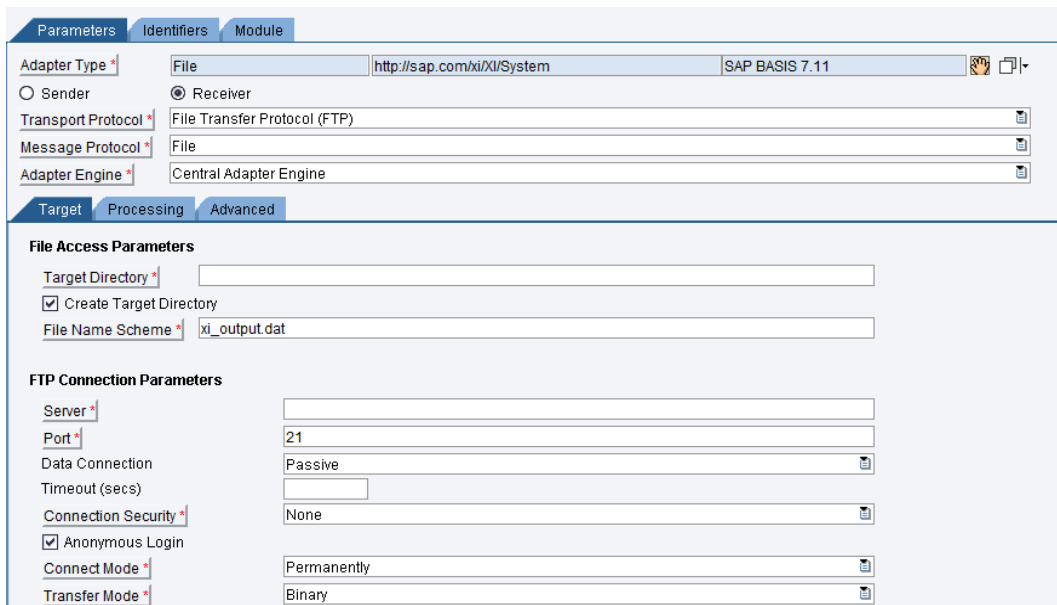
SAP PI sistemaren funtsa, SOAP protokoloaren erabilpena da informazioaren elkartrukerako, informazio guzti hau XML fitxategien bitartez bidaltzen delarik. Gainera, diseinatzaren diren ingurune (Scenario) guztiak, Web Zerbitzu bezala argitaratzen dira, edozein sistemak horik erabiltzeko aukera duelarik, haien WSDLa jakinik.

Honetaz gain, SAP PI sistemak beste hainbat estandar beteten ditu, hala nola, XPath, XSLT, XSD, W3, BPEL, eta abar.

## Adaptadoreak

Erabili nahi den komunikazio teknologia bakoitzeko, adaptadore izeneko gehigarri bat baliatu egin behar da PIin. Honela, fitxategiak trukatzeko 'File Adapter' izeneko adaptadorea erabiltzen da, RFC deiak egiteko 'RFC Adapter' izenekoak, eta abar.

Adaptadore hauetako bakoitzak konfigurazio ezaugarri propioak ditu, adibidez, fitxategi bat FTP bidez idazteko, zerbitzariaren helbidea beharko da, erabiltzaile eta pasahitzak, fitxategiaren izena ...



The screenshot displays the configuration interface for a File Adapter in SAP PI. The interface is divided into several sections:

- Parameters:**
  - Adapter Type: File
  - Transport Protocol: File Transfer Protocol (FTP)
  - Message Protocol: File
  - Adapter Engine: Central Adapter Engine
- Target:**
  - File Access Parameters:**
    - Target Directory: [Empty field]
    - Create Target Directory
    - File Name Scheme: xi\_output.dat
  - FTP Connection Parameters:**
    - Server: [Empty field]
    - Port: 21
    - Data Connection: Passive
    - Timeout (secs): [Empty field]
    - Connection Security: None
    - Anonymous Login
    - Connect Mode: Permanently
    - Transfer Mode: Binary

### 3.5 Irudia: SAP PIko Adaptadore baten adibidea

Defektuz, instalatzen den edozein SAP PI sistemak adaptadore estandar ugari ekartzen ditu, komunikazio gehienetan nahikoak direnak. Hala ere, adaptadore desberdinak gartzten dituzten enpresak ere badaude, teknologia berezietara zuzenduak, EDITRANerako, adibidez.

Gainera, adaptadore propioak sortzeko aukera ere badago, Java zein ABAP programazio lengoaiak erabilia, bakoitzak bere nahietara moldatutako adaptadoreak egin ditzan.

Hurrengo irudian, instalazio estandar batek ekartzen dituen adaptadoreak ikus daitezke.

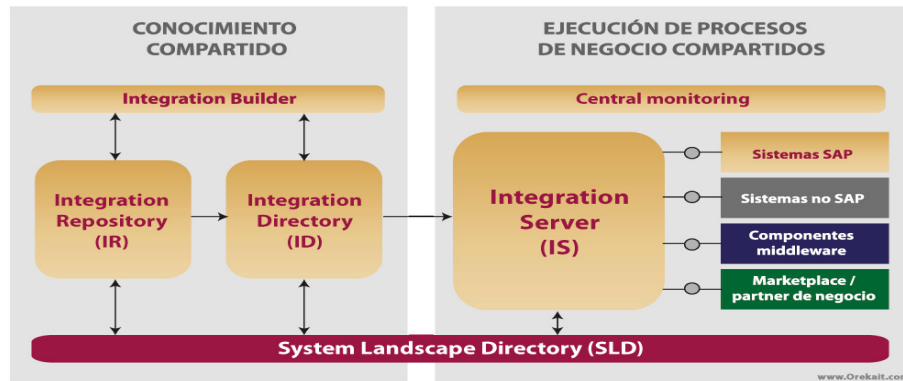
Name ^	Namespace	Software Component Version	Description
BC	http://sap.com/xi/XI/System	SAP BASIS 7.11	Business Connector
CIDX	http://sap.com/xi/XI/System	SAP BASIS 7.11	CIDX AdapterMetadata
File	http://sap.com/xi/XI/System	SAP BASIS 7.11	File
HTTP	http://sap.com/xi/XI/System	SAP BASIS 7.11	HTTP
IDoc	http://sap.com/xi/XI/System	SAP BASIS 7.11	IDoc
JDBC	http://sap.com/xi/XI/System	SAP BASIS 7.11	JDBC
JMS	http://sap.com/xi/XI/System	SAP BASIS 7.11	JMS
Mail	http://sap.com/xi/XI/System	SAP BASIS 7.11	Mail
Marketplace	http://sap.com/xi/XI/System	SAP BASIS 7.11	Marketplace
RFC	http://sap.com/xi/XI/System	SAP BASIS 7.11	RFC Adapter
RNIF	http://sap.com/xi/XI/System	SAP BASIS 7.11	RNIF Adapter
RNIF11	http://sap.com/xi/XI/System	SAP BASIS 7.11	RNIF1.1 AdapterMetadata
SFTP	http://sap.com/xi/XI/SFTP	SFTP ADAPTER 1.0	
SOAP	http://sap.com/xi/XI/System	SAP BASIS 7.11	SOAP
WS	http://sap.com/xi/XI/System	SAP BASIS 7.11	
XI	http://sap.com/xi/XI/System	SAP BASIS 7.11	XI

### 3.6 Irudia: SAP PIko Adaptadore estandarrak

#### 3.3.2 Arktitektura

Atal honetan SAP PIren arkitektura aztertuko da, bere osagaiak deskribatuko direlarik. Hasi baino lehen, argi geratu behar da SAP PI ez dela SAP sistemako osagai simple bat, osagai multzo bat baizik.

Osagai multzo honetan, helburu desberdinak dituzten osagaiak biltzen dira. Honela, badaude diseinu garaian erabiltzen diren osagaiak, beste batzuk konfigurazio garaian erabiltzekoak dira eta, azkenik, exekuzio garaian erabiltzekoak ere badaude.



3.7 Irudia: SAP PIko arkitektura

Hurrengo lerroetan, osagai bakoitzaren eginbeharrak deskribatzen dira.

**System Landscape Directory (SLD):** Software eta datu zentro sistemei buruzko informazio guztia biltzen duen errepositorioa da.

**Integration Builder:** Bezero-zerbitzari egitura duen sistema, Shared Collaboration Knowledge izeneko pilak kudeatzeko.

**Integration Repository (IR):** Scenario-en inplementazioetan erabiliko diren interface, prozesu eta objektu mapeoak diseinatzeko ingurua. Azkenengo bertsioetan, Enterprise Server Repository (ESR) ere deitzen zaio.

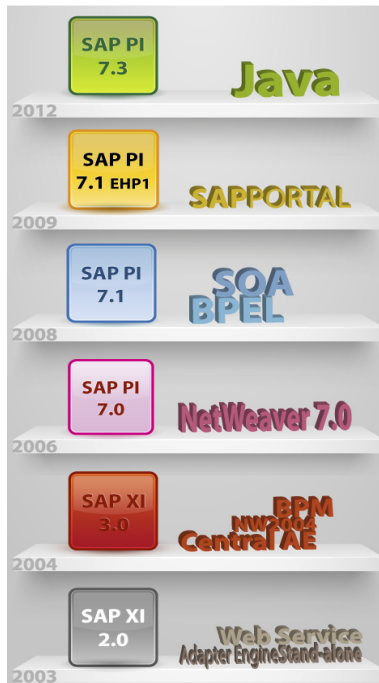
**Integration Directory (ID):** Scenario-en konfigurazioa egiteko erabiltzen da, aurretik, Integration Repository (IR) ingurunean definitutako interface, prozesu eta mapeoak erabiliz.

**Integration Server (IS):** SAP PIko prozesamendu motore zentrala da. Mezu guztiak, SAPekoak izan edo ez, aplikazio zein datu basekoak, modu trinko eta koherente batean prozesatzen dira.

**Central Monitoring:** Exekuzio garaian osagai eta prozesu guztiak monitorizatzeko erabiltzen da.

**Adapter Engine:** Java Conector Architecture (JCA) izeneko Java konektore bat da, SAP PI eta backend sistemak konektatzeko erabilia.

### 3.3.3 Kronologia



**3.8 Irudia:** SAP PIren kronologia

**SAP XI 2.0:** Lehenengo bertsioa izan zen eta jada gaurko PIk dituen zenbait ezaugarri ekartzen zituen, WEB Service protokoloa gehitu zuelarik. Baina, Adapter Engine-a stand-alone aplikazio motakoa zen, honek dakartzan mugekin.

**SAP XI 3.0:** SAP XIko bertsio nagusia. Netweaver 4.0k onartzen zituen sistema eragile eta datu baseetarako soportea ematen hasi zen. Gainera, BPM motorea ezarri eta Adapter Engine-a zentralizatu ziren.

**SAP PI 7.0:** Bertsio honetan, garrantzitsuena, izen aldaketa izan zen, non, SAP Basis 7.0 teknologian oinarritu arren, funtzionalitatea SAP XI 3.0koa zen.

**SAP PI 7.1):** Netweaver plataformako SOA (Service-Oriented Architecture) middleware bezala ezarri zen, honek hobekuntza ugari ekarri zituelarik, BPEL estandarerako soportea, adibidez.

**SAP PI 7.1 EHP1:** Aurreko bertsiorako hobekuntza pakete bat izan zen hau. Nabariena hemen, gestiorako JAVA Administrator kentzea eta SAPPORAL ezartzea izan zen. Hobekuntza hau, hala ere, ez da SAP PIkoa bakarrik, Netweaver plataforma osokoa da.

**SAP PI 7.3:** Gaur egungo bertsioa, JAVArekiko hobekuntza garrantzitsuak dituena.

### 3.3.4 Integration Process (ccBPM)

SAP PI In Integration Process, edo ccBPM, deitzen zaio Integration Repository errepositorioan egiten den eta hainbat mezuren prozesamenduaren sekuentzia definitzen duen sistemari. Honi esker, mezuen trukeak egiteaz gain, logika bat ere definitu ahal zaio SAP PI sistemari, baldintzak ezartzeko, adibidez, mezuen bidalketen artean. Gainera, bidaltzen den mezu ororen monitorizaziorako trena bat dauka, gertatzen den guztia kontrolatzeko.

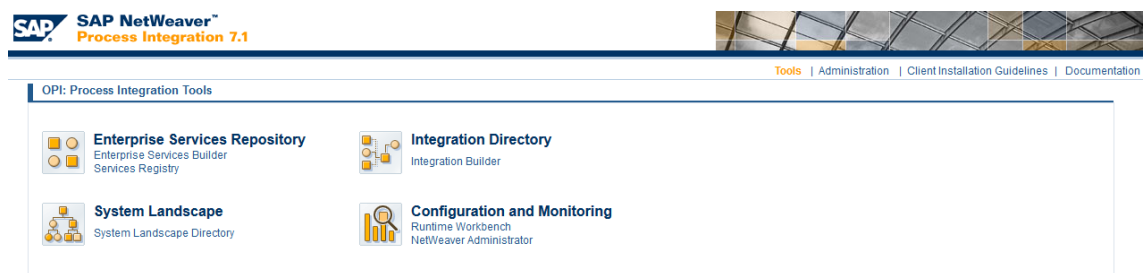
Lan-fluxu honen erabilerak honakoak egiteko aukera ematen du: Mezuak/Kontrolak monitorizatzea, mezual bildu eta elkartzea, mezuak zatitzea, mezuak hainbat hartzaile desberdinetara bidaltzea, alertak bidaltzea, ...



### 3.3.5 Scenario bat sortzen

Hurrengo lerroetan, komunikazio scenario bat sortzeko jarraitu beharreko pausuak azaltzen dira.

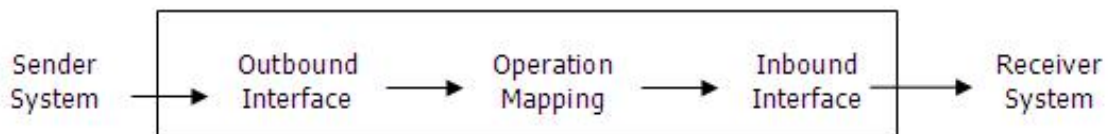
Lehenengo pausua web nabigatzailearen bitartez SAP PIren hasiera orrialdea irekitzea da, bertan, prozesu honetan erabiliko diren lau baliabideei estekak agertzen dira: *Enterprise Services Repository (ESR)*, *Integration Directory (ID)*, *System Landscape (SL)* eta *Configuration and Monitoring (CM)*.



**3.9 Irudia:** SAP PIko hasiera orria

#### Enterprise Services Repository

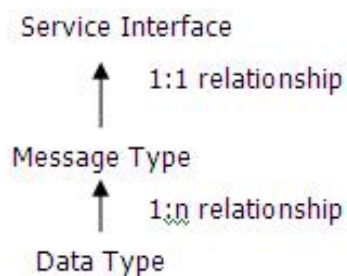
Herreminta honen bitartez, komunikazioan erabiliko diren objektuak diseinatu eta sortzen dira. PIko datuen fluxua, honakoa izaten da, normalean:



**3.10 Irudia:** Datu fluxua

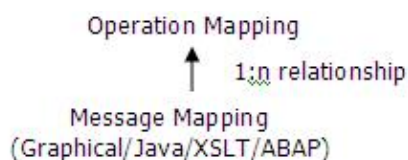
Sor daitezkeen objektuak hiru motakoak izan daitezke.

1. Interface objects: Service Interface, Message Type, Data Type.
2. Mapping objects: Operation Mapping eta Message Mapping izenekoak.
3. Integration Processes: Aurretik azaldutako lan-fluxuak.



**3.11 Irudia:** Data Type, Message Type eta Service Interface erlazioa

Integration Respository tresna sistema igorle eta hartzaileen mezuen egitura diseinatzeko erabiltzen da, gainera, mezu bakoitzeko interface bat sortzen da kanporako interakzio-puntu bezala. Datu eta mezu motak interfaze konplexuak sinplifikatzeko erabiltzen dira.



**3.12 Irudia:** Operation Mapping eta Message Mapping erlazioa

Operation Mapping delakoak, bi egitura desberdinen arteko transformazioak egiteko erabiltzen dira. Interfazeekin bezala, mezuen mapeoak sinplifikatzeko erabiltzen dira. Mezuen mapeoak 4 modu desberdinetan egin daitezke:

1. Mapping grafikoak
2. Java Mapping-ak
3. ABAP Mapping-ak
4. XSLT Mapping-ak

Mapping grafikoak dira erabilienak, modu grafiko batean mapeatu baitaiteke bi egituren arteko atributuak, datuak interfaze batetik bestera pasatzeko. Besteak, kode bitartez egin behar dira eta, beraz, konplexuagoak dira, aukera gehiago ere dituztelarik.

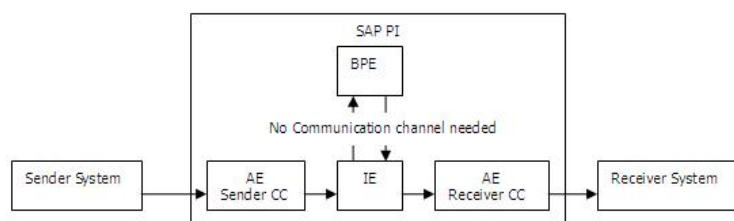
### Integration Directory

Hemen, komunikazioak konfiguratu egiten dira, ESR erremintan sortutako objektuak erabiliz. Hasi baino lehen, bi elementu sortu/inportatu behar dira: Zerbitzua eta Komunikazio kanala.

Zerbitzu bat, azken finean, komunikazioaren parteetako bat da, igorle edo hartzailea, zeinek mezuak jaso/bidaliko dituen. Zerbitzu desberdinak daude erabilpenaren arabera:

- Business System: Business System bat, jada konfiguratutako sistema bat da.
- Business Service: Entitate abstraktua da, ez da inon gehiago definitzen.
- Integration Process Service: Igorle/hartzailea lan-fluxu bat ere izan daiteke, bertatik mezuak atera edo bertara mezuak bidali ahal izateko.

Komunikazio kanalak determinatzen du mezuak sarrerakoak (inbound) edo irteerakoak (outbound) diren. Mezu hauek soap+xml formatutik mezu formatu konkretuetara, eta kontrari, transformatzen dira Adaptadoreen bidez. Bi komunikazio kanal daude, Sender Communication channel (Igorlea) eta Receiver Communication channel (Hartzailea).



**3.13 Irudia:** Sender eta Receiver Adaptadoreak

Komunikazio kanalak zerbitzuei lotuta doaz, ezin zaiolarik komunikazio kanal bat Integration Process zerbitzu bati esleitu. Honela, adibidez, zerbitzu bat fitxategi zerbitzari bat izan daiteke eta, komunikazio kanala, FTP adaptadorea erabiltzen duen komunikazio kanal hartzailea.

Komunikazioak sortzeko, 4 konfigurazio desberdin sortu behar dira: *Sender Agreement*, *Receiver Determination*, *Interface Determination* eta *Receiver Agreement*.

Sender Agreement delakoak, igorle baten mezua nola transformatu behar den definitzen du, Integration Server zerbitzariak prozesatu ahal izateko. Bertan, zerbitzua, interfazea eta komunikazio kanalak zehazten dira. Sender Agreement hau taula baten gakoak bezalakoak dira, ezin dira bi berdin egon sistema berean.

Receiver Agreement delakoak, hartzaileak mezua jasotzeko, honek jasan behar duen transformazioa definitzen du. Bertan, mezuaren igorlea zehazteaz gain, sistema hartzailea zehazten da, honen interfaze eta komunikazio kanalekin batera.

Receiver Determination konfigurazioan, mezu bat nori bidali behar zaion definitzen da. Horretarako, igorlearen sistema eta interfazeaz gain, sistema hartzailea ere zehazten da.

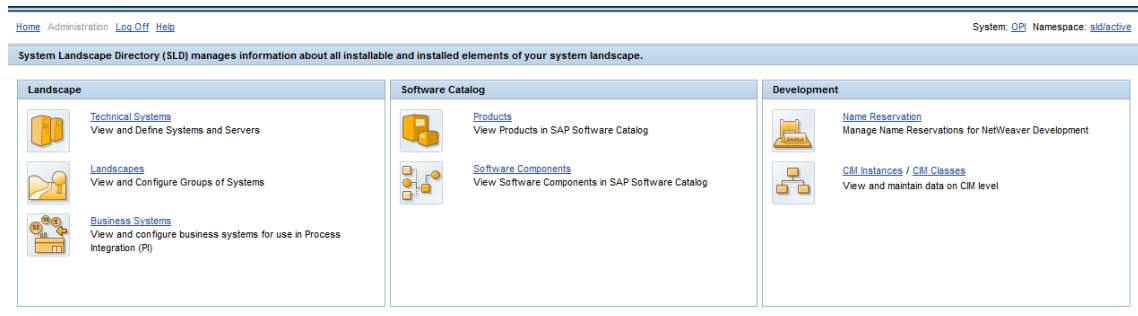
Interface Determination delakoan, mezua zein hartzaile interfazera transformatu behar den zehazten da. Hemen, ESRan definitutako mapeoak ere adierazi daitezke. Hemen zehaztu behar direnak, igorlearen sistema eta interfazeak eta hartzailearen sistema eta interfazeak dira.

## System Landscape

System Landscape Directory (SLD) izenarekin sistemaren azpiegituraren informazio horntzaile zentrala ezagutzen da. Bertan, hurrengo baliabideei estekak aurkitzen dira:

- Technical System: Hauek PIko azpiegitura sisteman instalatuta dauden aplikazio sistemak dira.
- Business System: Hauek sistema logikoak dira, PIn igorle/hartzaile bezala funtzionatzen dutenak. Hauetako bakoitza Technical System sistema bati lotuta dago.
- Products and Components: SAP produktu eta baliabide guztien informazioa biltzen da hemen. Azpiegiturako hirugarrenen produktuak ere hemen daude erregistratuta.

Produktu eta baliabideei, Component Information deitzen zaio normalean, sistemei (Technical eta Business) Landscape Description deitzen zaielarik.

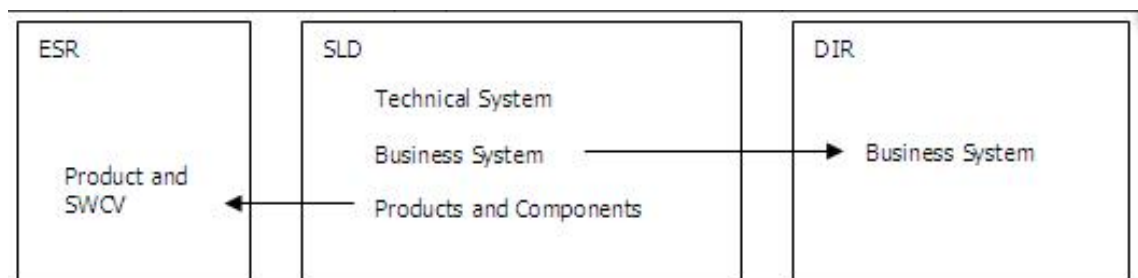


### 3.14 Irudia: System Landscape

Business System bat, Integration server edo Application system bezala konfigura daiteke:

1. Integration Server: Honek, Integration Builder sisteman konfiguratutako logikak bakarrik exekutatzen ditu. Hauek komunikazio bidearen pausu bezala ezar daitezke. Bere lana, XML mezuak jaso, hartzailea identifikatu eta, mapeatu ondoren, XML mezu hauek sistema hartzailera bidaltzea da.
2. Application system: Aplikazio sistemak XML mezuen igorle/hartzaile bezala funtzionatzen dute. Ez dute logikarik exekutatzen, Integration server sistemei deitzen diete logika horiek exekutzeko.

SAP sistemako kliente bakarra konfigura daiteke Integration Server bezala, honen eta, ESR eta DIR sistemekiko erlazioa honakoa delarik:



### 3.15 Irudia: SLren erlazioa ESR eta DIRekin

Honela, Component Information delakoa ESRan erabiltzen da produktuak eta software konponenteak definitzeko, eta, Business System izeneko sistemak, DIRean erabiltzen dira igorle/hartzaileak definitzeko.

## Konfigurazio eta monitorizazioa

Baliabide hau, monitorizazioetarako puntu zentrala da. Bertan, Integration Engine sistemaren monitorizazio funtzioetara nabigatzeko aukera eskaintzen da, Computing Center Management System (CCMS) eta SAPeko Process Monitoring Infrastructure (PMI) baliabideekin integratzeaz gain.

Configuration and Monitoring baliabideak, honako funtzioen monitorizazioa ahalbidetzen du:

- Component monitoring: SAP PIko konponente desberdinen monitorizazioa (ABAP zein Java parteak).
- Message monitoring: Mezuen prozesamenduaren egoeraren jarraipena SAP PI osagaietan, eta baita errore detekzio eta analisisian.
- End-to-end monitoring: Mezu baten bizi-zikloaren monitorizazioa SAP PIren ikuspegitik.
- Performance monitoring: SAP PIren errendimenduaren estatistiken azterketa.
- Index administration: Mezuentzako bilaketa-indizeen kudeaketa, mezuen monitorizazio prozesuan erabiltzeko.
- Alert configuration: PI prozesuan gerta daitezkeen erroreen abisuen konfigurazioa. Alerten bidez, errore edo salbuespenen berri eman daiteke modu erraz eta zuzenean.
- Alert inbox: Alerta biltegitratzea, hemen gordetzen dira erabiltzaile bakoitzaren alertak.
- Cache monitoring: Momentu honetan exekuzio cachean dauden objektuen monitorizazioa.

## 4. KAPITULUA

---

### Analisia

---

Atal honetan sistemaren analisia aztertuko da, hau da, zer egingo den aztertuko da, nola egingo den ere adieraziz. Horretarako, lehenik eta behin sistema bera deskribatuko da, modu orokorrean, eta, ondoren, sistemaren xehetasunak definituko dira, alderdi zehatza-goak azalduz.

Analisia, beraz, hurrengo atalak izango ditu: Sistemaren deskribapena, Betekizunen analisia eta Erabilpen Kasuak.

#### 4.1 Sistemaren deskribapena

Aurretik aipatu den moduan, sistema honen helburua SAP eta METAPOSTA sistemak konektatuko dituen konektore bat garatzea izango da. Konkreterik, SAP sistemaren bitartez sortzen diren dokumentuak METAPOSTAra zuzenean bidaltzeko aukera emango duen tresna bat garatu nahi da, ERP hau erabiltzen duten enpresen lana errazteko. Gainera, eta dokumentu pertsonalak izango direla kontuan hartuz, modu seguru batean egin beharko da bidalketa, ahalik eta kontrol gehien edukita gertatzen den guztian, gerta daitezkeen arazoak modu eraginkor batean konpondu ahal izateko.

Konektore honek egingo duen lana, beraz, aurretik deskribatutako METAPOSTAko Sender orokor bat sortzea izango da. Konektorea, beraz, ez da Sender bat izango, Sender berriak sortzeko tresna baizik, SAP sistema erabiltzen duen edozein enpresa Sender bezala erregistratzea ahalbidetuz.

## 4.2 Betekizunen analisia

Edozein Sender-ek izan behar dituen ezaugarriak aipatu ditugu aurreko atalean, eta, atal honetan, ezaugarri hauek proiektu honetan zehar nola ezarriko den azaldu da, modu sakonago batean.

Lehenengo gauza, **erabiltzaileen harpidetzaren** kudeaketa izango da. Horretarako, Web Service bat sortu beharko da, METAPOSTako Web orrialdetik atzigarri egongo dena, edozein erabiltzailek bere harpidetza kudeatu dezan. Hau da, edozein METAPOSTako erabiltzailek Konektore hau inplementatzen duen edozein Sender-ean alta edo baja eman nahi duenean, Web Service hau kontsumituko du.

Web Service honek, zenbait datu jasoko ditu METAPOSTatik eta, hauek baliatuz, erabiltzaileak identifikatu beharko ditu bere sisteman. Honela, zihurtatu nahi dena da Sender-era harpidetu nahi den METAPOSTako erabiltzailea, benetan, sistema horretarako erabiltzailea ere dela. Honela, enpresa bateko fakturak, adibidez, bere bezeroek bakarrik jaso behar dituzte.

Web Service honek bi zerbitzu eskaini beharko ditu, erabiltzaileak identifikatzeko (*SENDERPetitionTipoUsuario* izenekoa) bat eta harpidetzak kudeatzeko (*SENDERPetitionSuscripcion* izenekoa) bestea. Zerbitzu hauek, esan bezala, METAPOSTako sistematik deituak izango dira, bere Web orrialdeko atal berezi bat erabiltzen denean, eta hurrengo atalean ikusiko dugun fluxua jarraituko dute.

Zerbitzu hauek inplementatzeko ABAP ingurunea erabiliko da, honek eskaintzen duen *Enterprise Service* baliabidea hain zuzen ere. Honek, RFC protokolorako egokitutako *Funtzio Moduluak* SOAP protokoloaren bidez atzigarri uzten ditu URL batean, bertatik kontsumitu ahal izateko. Sistema honen abantaila denborak dira, oso azkarra baita mezuak bidaltzen, eta estandarrak erabiltzen ditu (WSDL, XSD, ...).

Erabiltzaileen harpidetza kontrolatuta dagoela, hurrengo pausua, horientzako dokumentuak sortzea izango da. **Dokumentuen sorketa** prozesu hau, SAP sistemaren modulu eta baliabide estandarrak erabilia egingo da, ahal den eanean. Hau da, langileen nominak sortzeko HR (Giza Baliabideak) modulua erabiliko da, fakturak sortzeko FI (Finantza) modulua, eta abar.

Dokumentuen sorketa hau, ABAP programazio lengoia baliatuta egingo da eta ahalik eta orokorrena izan beharko du, Konektorea konfigurazio desberdineko enpresatan erabili ahal izateko.



Proiektu honi dagokionez, bi dokumentu mota sortzeko programak egingo dira (aurrera-  
go gehiago sortzeko aukera egongo da, baina, hori, proiektu honetatik at geratzen da),  
nominak eta fakturak. Programa hauen bitartez, SAP sistemako datuak eskuratuko dira  
eta, PDF fitxategietan gordeko dira, enpresa bakoitzaren formatua jarraituz.

Berriro ere, ezin denez enpresa bakoitzaren konfigurazio zehatza jakin, parametrizagarria  
egingo da, toki bakoitzeko formatuetara egokitzeko.

Sortzen diren dokumentu hauek bidaltzeko, METAPOSTAko pakete sistema propioa era-  
biliko da. Honen bidez, bidalketak modu egoki eta fidagarrian egiten direla zihurtatzen da,  
dokumentu guztiak ondo prozesatzen direlarik. Pakete hauek guztiek hiru fitxategi nagusi  
izango dituzte beti:

- ZIP fitxategia: Hemen bilduko dira bidali nahi diren PDF dokumentu guztiak, gu-  
txiago okupatu eta fitxategi bakar batean joan daitezten.
- INDEX.XML fitxategia: Fitxategi hau ere ZIP fitxategiaren barruan joango da kon-  
primituta, eta, bidaltzen den dokumentu bakoitzeko sarrera bat izango du, doku-  
mentuaren datuak zehazteko (hartzailearen NIFa/CIFa, jaiotze data, dokumentuaren  
izena, MetaDatuak, ...).
- LOG.XML fitxategia: Fitxategi hau ZIP fitxategitik kanpo egongo da eta kontrole-  
rako erabiliko da. Bertan, bidalketarekin berarekin erlazionatutako datuak zehaztu-  
ko dira (bidalketaren identifikazioa, publikazio data eta ordua, larritasuna, ...).

Bidalketa bakoitza, beraz, ZIP fitxategi batez eta LOG XML fitxategi batez osatuta egon-  
go da eta identifikadore bat izango du beste bidalketa guztietatik desberdintzeko.

Ikusten den bezala, dokumentuak sortzeko prozesuan, PDF dokumentuak berak sortzea-  
rekin batera, paketeen bidaliko diren fitxategiak mantendu beharko dira. Hau da, sortzen  
den dokumentu bakoitzeko, INDEX fitxategia eguneratu beharko da, dokumentu guztiak  
sortuta daudenean, ZIP batean konprimitu beharko direlarik, LOG fitxategia ere sortuz.

**Bidalketa prozesua** sFTP (Secure File Transfer Protocol) protokoloa erabiliz egingo da,  
segurtasuna bermatu ahal izateko, bidaliko diren datuak pertsonalak izango baitira. Hau  
egiteko, Sender bakoitzak bi karpeta edukiko ditu METAPOSTAko zerbitzarietan, pake-  
teak uzteko bat eta erantzunak (ACKk) jasotzeko beste bat.

Bidalketa hauek egiteko bi aukera aztertu dira, alde batetik, ABAP ingurunea bera, eta, bestetik, aurretik azaldutako SAP PI sistema. Lehenengoa sinpleagoa da eta ez du ezarpen prozesu handiegirik behar, baina konfigurazio eta kontrol aukera gutzi eskaintzen ditu, eta, gainera, sistema eragilearen menpe dago, SAPek ez baitu sFTP protokoloa inplementatzen. Bigarrena, ordea, konplexuagoa da eta SAP modulu baten instalazioa behar du, baina askoz konfiguragarriagoa da, eta aukera gehiago eskaintzen ditu bidalketen kudeaketarako.

Abantaila eta desabantaila horiek aztertuta, proiektu honetan SAP PI sistema erabiltzea erabaki da, Konektorearen erabiltzaile potentzialak ikusita, egokiagoa dela ikusi baita. Hala ere, etorkizunean, eta proiektu honetatik kanpo, beste aukera ere garatzeko posibilitatea aztertuko da.

Bidalketa prozesu honen funtsa, paketeak sFTP zertizariko karpeta batean utzi eta konfirmazio dokumentuak beste batetik jasotzea izango da, azken hauek jasotzeaz gain, irakurri eta tratatu beharko direlarik ere. Erantzunak, dokumentuak bezala, paketetan bilduta egongo dira, bidalketa bakoitzeko erantzun pakete bat sortuko delarik.

Erantzunak ZIP fitxategi batean konprimituta egongo dira eta honako hiru fitxategiak edukiko dituzte:

- ACKREC.XML: Paketearen rezeptzioaren emaitza, ondo jaso den edo ez zehaztuz.
- ACKVAL.XML: Paketearen formatuaren balidazioaren emaitza, formatua egokia den edo ez zehaztuz.
- ACKDOCS.XML: Dokumentuen prozesaketaren emaitza, fitxategiren bat errorerik izatekotan, errorearen deskribapena adieraziz.

Batzutan, ez dira hiru fitxategiak bidaliko, paketearen balidazioan erroreak gertatzekotan, adibidez, ez baita dokumenturik prozesatuko eta, beraz, ez baita dokumentuen prozesaketaren emaitzarik jasoko.

## 4.3 Erabilpen Kasuak

Hurrengo lerroetan, aurreko atalean deskribatutako betekizunak modu zehatzago batean aztertuko dira. Hori egiteko, erabilpen kasuak erabiliko dira Konektorearen eta hura erabiltzen duten aktoreen arteko elkarrekintzak adierazteko. Erabilpen kasu hauetan, gertaera fluxu normalak eta salbuespenak deskribatuko dira, prozesu osoa argi irudikatzeko.

Aurretik aipatu den bezala, proiektu honen helburu den Konektoreak hiru atal nagusi izango ditu; **METAPOSTA**ko erabiltzaileen harpidetzaren kudeaketa, dokumentuen sorketa **SAP**etik eta dokumentu hauen bidalketa **SAP PI** sistemaren bitartez.

Gauzak honela, Konektorearekin elkar eragingo duten hiru aktore desberdin identifika daitezke:

- **METAPOSTA:** Aktorea, METAPOSTAko Web Orrialdea kudeatzen duen sistema izango da. Honek deituko baitu Web Service-a, erabiltzaileen eskaerak kudeatzeko.
- **SAP sistemako erabiltzaileak:** Hauek izango dira dokumentuen sorketa prozesua deituko dutenak, horretarako egingo diren programak erabiliko dituztelarik.
- **SAP PI:** Aktore hau berezia da, sistema bera izango delako elkarrekintza guztiak sortuko dituen, inolako pertsonaren beharrik gabe.

Hurrengo lerroetan betekizunetan azaldutako prozesua osatzen duten 4 pausu nagusien erabilpen kasuak azalduko dira: **Erabiltzaileen harpidetza, dokumentuen sorketa, dokumentuen bidalketa** eta **erantzunen prozesamendua**.

### 4.3.1 Erabiltzaileen harpidetza

Erabiltzaileen harpidetza kudeatzeko, aurretik esan bezala, bi funtzio egongo dira: METAPOSTAko erabiltzaileak SAP sisteman identifikatzeko bat, *SENDERPeticionTipoUsuario* izenekoa, eta identifikatutako erabiltzaile bat Sender-ean alta edo baja emateko bestea, *SENDERPeticionSuscripcion* izenekoa.

Honela, erabiltzaile batek harpidetza eskatzen duenean, bi funtzio hauei deituko zaie jarraian METAPOSTAtik, prozesua, beraz, honakoa izango delarik: METAPOSTAk *SENDERPeticionTipoUsuario* zerbitzua deitzen du erabiltzailea existitzen dela zihurtatzeko. Ondoren, identifikazioa ondo joan bada, *SENDERPeticionSuscripcion* zerbitzua deituko du harpidetzaren alta edo baja den adieraziz.

<b>SENDERPeticionTipoUsuario</b>	
<b>Deskribapena</b>	Erabiltzaile bat Sender-ean existitzen dela zihurtu.
<b>Fluxu normala</b>	<p><b>01</b> METAPOSTAk erabiltzailearen datuak bidaltzen ditu, kontrol-datuekin batera.</p> <p><b>02</b> Sistemak erabiltzailea identifikatu eta bere datuak bueltatzen ditu.</p>
<b>Salbuespenak</b>	<p><b>Daturen bat txarto dago edo falta da</b> Sistemak errorea ematen du eta ez da identifikazioa egiten.</p> <p><b>Ezin da erabiltzailea identifikatu</b> Sistemak erabiltzailea existitzen ez dela bueltatuko du.</p>

4.1 Taula: 'SENDERPeticionTipoUsuario' erabilpen kasua

<b>SENDERPeticionSuscripcion</b>	
<b>Deskribapena</b>	Erabiltzaile bat Sender-ean harpidetu.
<b>Fluxu normala</b>	<p><b>01</b> METAPOSTAk erabiltzailearen Alta/Baja eskaera eta datuak bidaltzen ditu, kontrol-datuekin batera.</p> <p><b>02</b> Sistemak erabiltzailea (des)suskribatu egiten du eta emaitza bueltatzen du.</p>
<b>Salbuespenak</b>	<p><b>Daturen bat txarto dago edo falta da</b> Sistemak errorea ematen du eta ez da identifikazioa egiten.</p> <p><b>Ezin da erabiltzailea identifikatu</b> Sistemak erabiltzailea existitzen ez dela bueltatuko du.</p> <p><b>Erabiltzailea jada alta/baja emanda zegoen</b> Sistemak errorea ematen du eta ez da ezer gehiago egiten.</p>

4.2 Taula: 'SENDERPeticionSuscripcion' erabilpen kasua

### 4.3.2 Dokumentuen sorketa

Dokumentuen sorketa egiteko, esan bezala, bi programa iznago ditugu, bat nominak sortzeko eta beste bat fakturak sortzeko. Hauetan, bietan, hainbat datu sartu beharko dira sortu nahi diren dokumentuak zeintzuk eta norentzat diren adierazteko. Ondoren, eta dokumentuak sortuta daudela, SAP sistemak pakete batean sartuko ditu (aurretik azaldutako egiturarekin) eta hura karpeta batean utziko du bidalketarako prest.

Honela, hainbat pertsonentzako fakturak, adibidez, sortzeko, pertsona horiek identifikatzeko datuak (bezero zenbakia, izena, ...) eta fakturak identifikatzeko datuak (faktura zenbakia, data, emaila, ...) sartu beharko dira programa exekutatzean.

Bi funtzio hauen funtzionamendua oso antzekoa denez, erabilpen kasu bakar batean bilduko dira, hurrengo ataletan gehiago zehaztuko direlarik haien diseinu eta inplementazioak.

Dokumentuak sortu	
<b>Deskribapena</b>	Hainbat erabiltzailerentzako dokumentuak sortu.
<b>Fluxu normala</b>	<p><b>01</b> SAP sistemako erabiltzaileak, Sender-eko bezeroen datuak eta dokumentuen datuak sartzen ditu.</p> <p><b>02</b> Sistemak dokumentuak bilatu eta paketea sortzen du, prozesua ondo joan dela konfirmatuz bukaeran.</p>
<b>Salbuespenak</b>	<p><b>Daturen bat falta da</b> Sistemak derrigorrezkoak diren datuak eskatuko ditu.</p> <p><b>Ez dago dokumenturik sartutako datuentzat</b> Sistemak abisua ematen du, datuak konpondu eta berriro saiatzeko. <b>Dokumenturen bat ez da sortu</b> Sistemak abisua ematen du, baina prozesua jarraitzen da.</p>

**4.3 Taula:** 'Dokumentuak sortu' erabilpen kasua

### 4.3.3 Dokumentuen bidalketa

Dokumentuen bidalketak gauzatzeko, SAP PI sistema erabiliko da, sistema hau, aurreko kapituluaz azaltzen den bezala, datuen trukea ahalbidetzen duen sistema berezi bat delarik, adaptadoreen bidez funtzionatzen duena. Sistema honen funtzionamendua nolakoa den azalduta dagoenez, hurrengo lerroetan, ikusitako hori, proiektu honetan nola erabiliko den deskribatuko da.

Hurrengo tauletan agertzen diren Igorle/Hartzaileak, PI sistemako Komunikazio Kanalak izango dira, haien arteko mezu trukeek osatuko baitute dokumentuen bidalketa prozesua.

<b>Dokumentuak bidali</b>	
<b>Deskribapena</b>	Dokumentuen paketea SAPetik METAPOSTako sFTP zerbitzariara bidali.
<b>Fluxu normala</b>	<ul style="list-style-type: none"> <li><b>01</b> Igorlea SAP sistemako karpeta batean "entzuten"dago pakete berri bat heldu arte.</li> <li><b>02</b> Pakete berria heltzen denean, hartzaileari bidaltzen dio.</li> <li><b>03</b> Hartzaileak paketea jaso eta zerbitzariko karpeta batean uzten du.</li> </ul>
<b>Salbuespenak</b>	<p><b>Paketea jada bidali da</b> Errore bat ikusiko da eta ez da berriro ere prozesatuko.</p> <p><b>Ezin da paketea utzi</b> Mezua kolan geldituko da prozesatu ahal den arte.</p>

**4.4 Taula:** 'Dokumentuak bidali' erabilpen kasua

<b>Erantzunak jaso</b>	
<b>Deskribapena</b>	Erantzunen paketea METAPOSTako sFTP zerbitzaritik SAPera bidali.
<b>Fluxu normala</b>	<ul style="list-style-type: none"> <li><b>01</b> Igorlea zerbitzariko karpeta batean "entzuten"dago pakete berri bat heldu arte.</li> <li><b>02</b> Pakete berria heltzen denean, hartzaileari bidaltzen dio.</li> <li><b>03</b> Hartzaileak paketea jaso eta SAP fitxategi sistemako karpeta batean uzten du.</li> </ul>
<b>Salbuespenak</b>	<p><b>Paketea jada bidali da</b> Errore bat ikusiko da eta ez da berriro ere prozesatuko.</p> <p><b>Ezin da paketea utzi</b> Mezua kolan geldituko da prozesatu ahal den arte.</p>

**4.5 Taula:** 'Erantzunak jaso' erabilpen kasua

SAP PI sistemaren konfigurazioak bi atal desberdin izango ditu. Alde batetik, bi adaptadore mota desberdin erabiliko dira, SFTP adaptadorea, METAPOSTAko zerbitzarian aritzeko, eta Fitxategien adaptadorea, SAP sistemako NFS (Network File System) fitxategi sistemarekin aritzeko. Beste aldetik, adaptadore hauek erabiliko dituzen lau Komunikazio Kanalak erabiliko dira, adaptadore mota bakoitzeko bi, Sender (igorle) eta Receiver (hartzaile) lanak egiteko.

#### 4.3.4 Erantzunen prozesamendua

Erantzunen prozesamendua egiteko, SAP PI sistema ere erabiliko da, haren *Integration Process* baliabidea hain zuzen ere. Honek, aurreko kapituluaz azaldu den moduan, lan fluxu bat definitzeko aukera ematen du, bertan, hainbat prozesu desberdin egin eta beste sistemekin komunikatzeko baliabideak daudelarik.

Lan fluxu honen funtsa, erantzunen paketea deskonprimitu eta lortzen dituen ACK fitxategi desberdinak irakurri eta prozesatzea izango da. Hau da, *Integration Process*-a izango da hiru erantzunen XMLak irakurriko dituen eta, horien arabera, bidalketa bat ondo edo gaizki joan den esango duena.

Kasu honetan aktore bakarra dagoenez, SAP PI sistema bera, ezin da erabilpen kasurik definitu, eta, beraz, hurrengo lerroetan lan fluxuaren pausu desberdinak azalduko dira, hurrengo kapituluetan zehaztuko delarik honen funtzionamendu zehatza.

<b>Erantzunak prozesatu</b>	
<b>Deskribapena</b>	Erantzunak prozesatu bidalketaren arrakasta zihurtatzeko.
<b>Fluxu normala</b>	<ul style="list-style-type: none"> <li><b>01</b> ZIP fitxategia lortu.</li> <li><b>02</b> ACK XML fitxategiak lortu eta irakurri.</li> <li><b>03</b> ACKREC.XML fitxategia egiaztatu, paketea ondo jasotzea zihurtatzeko.</li> <li><b>04</b> ACKVAL.XML fitxategia egiaztatu, paketearen formatua egokia dela zihurtatzeko.</li> <li><b>05</b> ACKDOCS.XML fitxategia egiaztatu, dokumentu guztiak ondo entregatu direla zihurtatzeko.</li> <li><b>06</b> SAP sistema eguneratu gertatutakoaren berri emateko.</li> </ul>
<b>Salbuespenak</b>	<b>ACK fitxategiren bat erroreak ditu</b> Prozesamendua bukatu eta erroreen berri emango da.

**4.6 Taula:** 'Erantzunak prozesatu' erabilpen kasua





## 5. KAPITULUA

---

### Diseinua

---

Kapitulu honetan, aurreko kapituluan deskribatu den sistemaren diseinua azalduko da. Lehendabizi, bertan definitutako Web Service zerbitzuaren diseinua aztertuko da, horretarako inplementatu beharko diren SAP Funtzioak eta interneteko zerbitzu bezala ezartzeko egin beharrekoak deskribatuz. Ondoren, dokumentuen sorketarako garatu beharko diren programa eta klaseak definituko dira, hauek erabiliko dituzten datu baseko taulak ere deskribatuz. Bukatzeko, dokumentuen bidalketan erabiliko den SAP PI sistema aztertuko da.

#### 5.1 Erabiltzaileen harpidetza

Erabiltzaileen harpidetza kudeatzeko Web Service-ak eskainiko dituen bi funtzioak *SENDERPeticionTipoUsuario* eta *SENDERPeticionSuscripcion* SAP Funtzio Modulu bezala definituko dira, Function Group bakar batean bilduta, hauen sarrera eta irteera datuak, METAPOSTak zehaztutakoak izango direlarik.

Hurrengo lerroetan, alde batetik, funtzio hauen definizioa deskribatuko da, eta, bestetik, Web Service bezala sortzeko erabiliko den Enterprise Services baliabidea deskribatuko da, honen Service Provider zehazki.

Irudi honetan, erabiltzaileak identifikatzeko eskaeren (Request) definizioa ikus daiteke.

▼ SENDERPeticionTipoUsuarioRequest	Metodo para saber el tipo de usuario
documento	Indica el numero del documento de identidad seleccionado en tipoDocumentoIdentidad
tipoDocumentoIdentidad	Indica que tipo de documento de identidad es: NIF (valor 1), CIF (valor 2), Tarjeta de residente (valor 3)
oidSender	OID del Sender
oidSeguridad	OID de seguridad generado en el Login por parte de METAPOSTA
firma	Firma (no contemplado en esta version)
fechaNacimiento	Fecha de Nacimiento del usuario (campo en la practica OBLIGATORIO)
auxData	Dato auxiliar enviado al sender para sus comprobaciones
auxData2	Dato auxiliar 2 enviado al sender para sus comprobaciones

### 5.1 Irudia: SENDERPeticionTipoUsuarioRequest

Hurrengo irudikoa, erantzuna (Request) da, Konektoreak bueltatu beharko duena.

▼ SENDERPeticionTipoUsuarioResponse	Respuesta a la petición de tipo de usuario
oidSender	Codigo del sender al que se le envia el paquete de usuarios
oidSeguridad	Codigo de seguridad para el intercambio de usuarios
tipoUsuario	Definición del tipo de usuario. valores posibles 01, 02, 03
documento	Indica el numero del documento de identidad seleccionado en tipoDocumentoIdentidad
tipoDocumentoIdentidad	Indica que tipo de documento de identidad es: NIF (valor 1), CIF (valor 2), Tarjeta de residente (valor 3)
firma	Firma del sender (no contemplada en esta version de servicio)
fNacimiento	Fecha de nacimiento del usuario. Serviria para validar dnis repetidos . Campo a efectos prcticos OBLIGATORIO
codigoRespuestaDeEjecucion	Definicion resultado de la ejecución
descripcionCodigoRespuestaDeEjecucion	Descripcion del codigo de respuesta de la ejecución
recepcionCorreoUsuario	Indica si hay que pedir o no al usuario el email para enviar al sender.

### 5.2 Irudia: SENDERPeticionTipoUsuarioResponse

Konektoreak jasoko dituen harpidetza eskaerak (Request) honelakoak izango dira.

▼ SENDERPeticionSuscripcionRequest	Metodo para suscribir un usuario a un SENDER desde
OIDSender	OID de sender
OIDPeticion	OID de Petición de suscripcion
documento	Indica el numero del documento de identidad seleccionado en tipoDocumentoIdentidad
tipoDocumentoIdentidad	Indica que tipo de documento de identidad es: NIF (valor 1), CIF (valor 2), Tarjeta de residente (valor 3)
firma	Firma. NO se usa en esta version del servicio
email	Email del usuario que envia al sender y no guarda METAPOSTA
fNacimiento	Fecha de nacimiento del usuario. Serviria para validar dnis repetidos
accion	Indica que tipo de accion se realiza, alta, edicion o baja de suscripcion
auxData	Dato auxiliar enviado al sender para sus comprobaciones
auxData2	Dato auxiliar 2 enviado al sender para sus comprobaciones
esAportado	Indica si el usuario ha sido aportado por el sender.

### 5.3 Irudia: SENDERPeticionSuscripcionRequest

Hau Konektorearen erantzuna (Response) izango da.

▼ SENDERPeticionSuscripcionResponse	Metodo para suscribir un usuario a un SENDER desde
OIDSender	identificativo del sender
documento	Indica el numero del documento de identidad seleccionado en tipoDocumentoIdentidad
tipoDocumentoIdentidad	Indica que tipo de documento de identidad es: NIF (valor 1), CIF (valor 2), Tarjeta de residente (valor 3)
firma	Firma. NO se usa en esta version del servicio
fNacimiento	Fecha de nacimiento del usuario. Serviria para validar dnis repetidos. Dato OBLIGATORIO de facto
codigoRespuestaDeEjecucion	Definicion resultado de la ejecución
descripcionCodigoRespuestaDeEjecucion	Descripcion del codigo de respuesta de la ejecución
OIDPeticion	OID de Petición de suscripcion
contratoUsuario	Indica el codigo de contrato del usuario en el sender de banca electronica
tipoUsuario	Definición del tipo de usuario. valores posibles 01, 02, 03
storableData	Dato almacenable. Valores S o N.

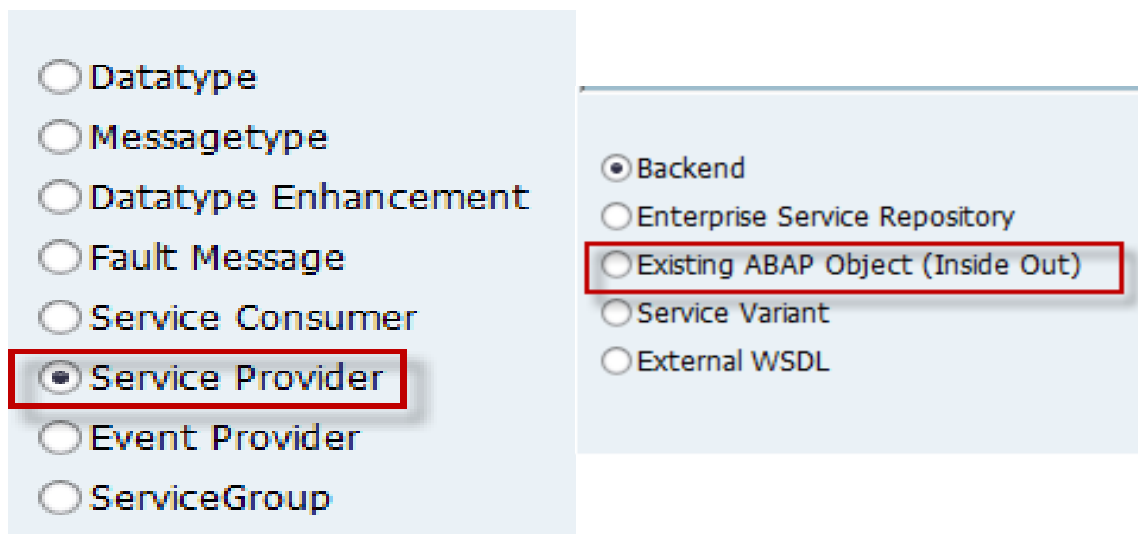
### 5.4 Irudia: SENDERPeticionSuscripcionResponse

Aurreko irudietan ageri diren funtzioak dira erabiltzaileen harpidetzak kudeatu ahal izateko Konektoreak implementatu beharko dituenak. Ikusten den moduan, erabiltzaileak identifikatzeko dokumentu eta jaiotze datak erabiliko dira, hauek izango direlarik Konektoreak gordeko dituen datuak, suskribatutako erabiltzaileak erregistratzeko.

Aurretik aipatu den moduan, funtzio hauek SAPeko funtzio bezala definituko dira, **Function Group** bateko bi **Function Module** bezala, hain zuzen ere. Funtzio hauek, gainera, RFC moduan definitu beharko dira, internetetik atzigarri egon ahal izateko.

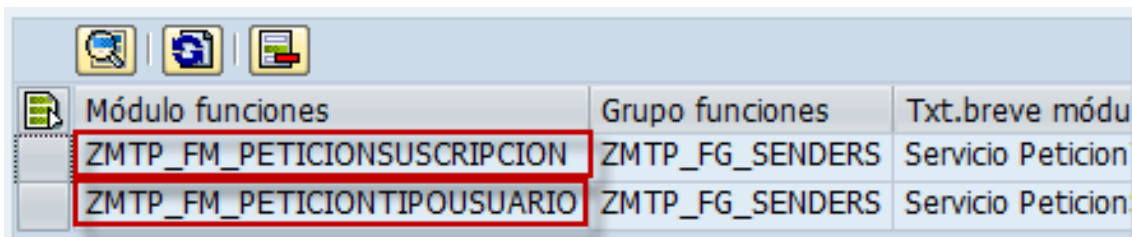
Funtzioak implementatutakoan, Web Service-a sortu beharko da, horretarako, SAPeko **Service Provider** baliabidea erabiliz. Baliabide honek, hurrengo irudietan ikusiko dugun moduan, funtzio batzuk eskatzen ditu eta, horiekin, zerbitzu bat sortzen du, web orrialde konkretu batetik atzigarri uzten duelarik. Irudietan, Konektoreak eskainiko duen 'ZMTP\_WS\_SUSCRIPCION' zerbitzuaren sorketa prozesua ikus daiteke.

Hasieran, zerbitzuaren ezaugarriak finkatzen dira. Proiektu honen kasuan, garrantzitsuenaz izango da 'Service Provider' bat dela eta ABAP objektuak atzigarri utziko dituela zehaztea.



5.5 Irudia: Web Service: Zerbitzu mota

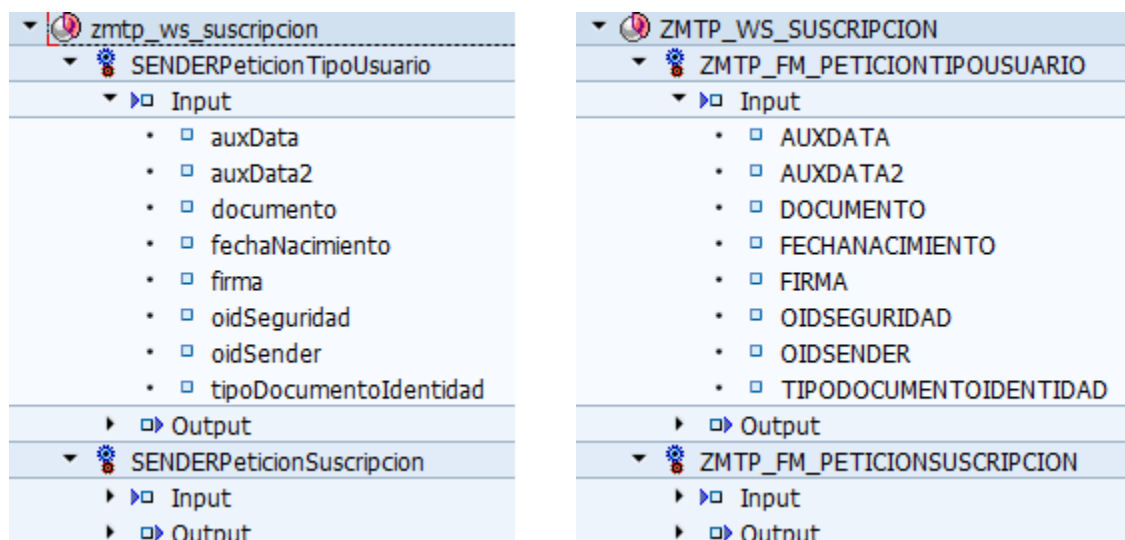
Ondoren, gehitu nahi diren ABAP objektuak aukeratu behar dira, horiek izan daitezzen zerbitzatzen direnak. Kasu honetan, esandako bi Funtzio Moduluak aukeratuko dira, bi metodoak atxigarri uzteko.



Módulo funciones	Grupo funciones	Txt.breve módu
ZMTP_FM_PETICIONSUSCRIPCION	ZMTP_FG_SENDERS	Servicio Peticion
ZMTP_FM_PETICIONTIPOUSUARIO	ZMTP_FG_SENDERS	Servicio Peticion

**5.6 Irudia:** Web Service: Funtzio moduluak zehaztu

Bukatzeko, nomenklatura aldatu behar da, SAP sistemaren barrutik eta bertatik kanpo ez baita berdina izango, adibidez, SAPen *ZMTP\_FM\_PETICIONTIPOUSUARIO* izena duen funtzioak, web zerbitzuan, *SENDERPeticionTipoUsuario* izena izango du. Hau izen guztiekin egin behar da, atxigarri uzten den zerbitzuak METAPOSTAko nomenklatura mantendu dezan. Irudi honetan ikusten da nomenklatura aldaketa hau:



zmtp_ws_suscripcion	ZMTP_WS_SUSCRIPCION
SENDERPeticionTipoUsuario	ZMTP_FM_PETICIONTIPOUSUARIO
Input	Input
<ul style="list-style-type: none"> <li>auxData</li> <li>auxData2</li> <li>documento</li> <li>fechaNacimiento</li> <li>firma</li> <li>oidSeguridad</li> <li>oidSender</li> <li>tipoDocumentoIdentidad</li> </ul>	<ul style="list-style-type: none"> <li>AUXDATA</li> <li>AUXDATA2</li> <li>DOCUMENTO</li> <li>FECHANACIMIENTO</li> <li>FIRMA</li> <li>OIDSEGURIDAD</li> <li>OIDSENDER</li> <li>TIPODOCUMENTOIDENTIDAD</li> </ul>
Output	Output
SENDERPeticionSuscripcion	ZMTP_FM_PETICIONSUSCRIPCION
Input	Input
Output	Output

**5.7 Irudia:** Web Service: Nomenklatura aldaketak

Honekin, Web Service-a konfiguragarria den URL batean sortuta eta atxigarri geratzen da. Gainera, zerbitzua definitzen duen WSDL fitxategia ere sortzen da erabili ahal izateko. Modu honetan sortutako zerbitzuak oso konfiguragarriak dira segurtasun eta pribatutasun arloetan, HTTPS, SSL, eta abar erabiltzeko aukera ematen dutelarik.

## 5.2 Dokumentuen sorketa

Dokumentuak sortzeko bi SAP Report (programa) sortuko dira, bat Nominak sortzeko eta bestea Fakturak. Hauek, bidali nahi diren dokumentuetarako beharrezkoak diren datuak (datak, identifikadoreak, ...) eta dokumentu horiek jasoko dituzten erabiltzaileen datuak eskatuko dituzte.

Bi Report hauekin batera, klase laguntzaile bat sortuko da, objektuetara zuzendutako programazioa erabiliz, programa hauek baliatu dezaten. Honela, dokumentuen sorkuntza eta paketeen kudeaketa klase honen barruan egingo da, dokumentu/erabiltzaileen aukeratzea programen esku geratuko delarik.

Klaseak eskainiko dituen metodoak hurrengo irudian ikus daitezke:

Método	Clase	Visibilidad	T...	Descripción
CONSTRUCTOR	Instanc...	Public	☞	CONSTRUCTOR
GENERAR_FACTURA	Instanc...	Public		Generar factura para cliente
GENERAR_NOMINA	Instanc...	Public		Generar nómina para empleado
GUARDAR_FICHEROS	Instanc...	Public		Generar y archivar los ficheros
NUMERO_FICHEROS	Instanc...	Public		Obtener número de ficheros en el ZIP
CANCELAR_ENVIO	Instanc...	Private		Cancelar el envío y borrar los datos
CREAR_LOG	Instanc...	Private		Crear log
GENERAR_NOMINA_HRFORM	Instanc...	Private		Generar nómina para el formulario con HRForms
GENERAR_NOMINA_SAPSCRIPT	Instanc...	Private		Generar nómina para el formulario con SAPScript
GENERAR_NOMINA_SMARTFORM	Instanc...	Private		Generar nómina para el formulario con SmartForm
GUARDAR_LOG	Instanc...	Private		Guardar el log
MENSAJE	Instanc...	Private		Añadir mensaje al log

### 5.8 Irudia: Klase laguntzailea

Ikusten den bezala, klaseak lau metodo publiko (sortzaileaz gain) eta zazpi pribatu izango ditu. Aipatu diren bi programek, metodo publikoak deituko dituzte haien exekuzioan. Metodo pribatuak, ordea, klase barrutik bakarrik deituko dira, metodo laguntzaileak izango direlarik.

Metodo pribatuen artean, hurrengo sekzio batean deskribatuko den LOGa kudeatzeko metodoak daude, hauek, dokumentu bat sortzen den bakoitzean, horren berri emango dutelarik.

Dokumentuen sorketan, beraz, metodo hauek deituko dituzte Report-ek: *generar\_xxx()* dokumentu bakoitza sortu eta paketera gehitzeko, *numero\_ficheros()* dokumentu guztiak gehitu direla egiaztatzeko, eta *guardar\_ficheros()* paketea sortu eta biltegitatzeko.

## Nominak

Nominak sortzeko, ZMTP\_REP\_NOMINAS programa erabiliko da, bertan, langilearen zenbakia eta nominaren datak sartu behar direlarik. Datuak dituela, Report-ak, langile bakoitzaren nomina bakoitzeko *generar\_nomina()* metodoa deituko du, honek PDF dokumentua sor dezan eta paketera gehitu dezan.

Hurrengo irudian, Report honen interfase grafikoa ikus daiteke, bertan, esan den bezala, langileen zenbakiak eta nominentzako datak sartu behar dira. Gainera, METAPOSTAra harpidetuta dauden erabiltzaile guztiak aukeratzeko *checkbox* bat dago, bidalketen prozesua automatizatu ahal izateko.

### 5.9 Irudia: Dokumentuen sorketa: ZMTP\_REP\_NOMINAS

Cl.	Parámetro	Tipificac.	Descripción
▶□	IM_PERNR	TYPE P_PERNR	Número de personal
▶□	IM_BEGDA	TYPE BEGDA	Fecha Inicio de la validez
▶□	IM_ENDDA	TYPE ENDDA	Fecha Fin de la validez

### 5.10 Irudia: Dokumentuen sorketa: generar\_nomina()

Irudi honetan ikusten den bezala, erabiltzaileak sartzen dituen datuak dira aipatutako metodoak jasotzen dituenak. Honek, era berean, datu hauek egokitzen den nomina modu konkretuari bidaliko dizkiolarik.

Aipatu behar da ere, nominak sortzeko hiru metodo desberdin erabili direla, SAPEk hiru modu desberdinez sor ditzakeelako nominen PDFak. Honela, Konektorea erabiltzen duen enpresa bakoitzak erabakiko du, aurrerago azalduko dugun taulen bitartez, hauetako zein erabili nahi duen.

## Fakturak

Fakturak sortzeko, ZMTP\_REP\_FACTURAS reporta erabiliko da, non, erabiltzaileak, dokumentuen hainbat datu sartuko ditu, hauek SAP sisteman bilatu eta, *generar\_factura()* metodoaren bitartez, paketera gehitzeko.

Hurrengo irudian, dokumentuen datu hauek zeintzuk diren ikus daiteke. Aurrekoan bezala, honetan ere, harpidetutako bezero guztiak aukeratu daitezke *checkbox* baten bidez, haien identifikazio zenbakiak banan-banan sartu behar ez izateko.

### 5.11 Irudia: Dokumentuen sorketa: ZMTP\_REP\_FACTURAS

Cl.	Parámetro	Tipificac.	Descripción
▶□	IM_KUNNR	TYPE KUNNR	Número de deudor
▶□	IM_BELNR	TYPE BELNR_D	Asignación de núm.pos.: documento material-documento compras
▶□	IM_GJAHR	TYPE GJAHR	Ejercicio
▶□	IM_BLDAT	TYPE BLDAT	Fecha de documento en documento

### 5.12 Irudia: Dokumentuen sorketa: generar\_factura()

Irudi hauetan ikusten den bezala, bai programak eta baita metodoak ere, datu berdinak erabiltzen dituzte. Hala ere, hurrengo kapituluan deskribatuko den moduan, programatik metodoa deitzeko zenbait logika exekutatu beharko da, datuak modu egokian pasatzeko.

## 5.3 Dokumentuen bidalketa

Dokumentuen bidalketa prozesu osoa, SAP PI sistemaren bitartez egingo da. Sistema honek, aurreko kapitulu batean azalduta dagoen bezala, bi interfase nagusi ditu komunikazioak konfiguratzeko: *Enterprise Service Builder* eta *Integration Builder*. Lehenengoan, mezu/interface-ak sortzen dira, eta, bigarreanean, hauen trukeak konfiguratu.

Bidalketa prozesua sinplea izango da, fitxategiak karpeta batetik irakurri eta beste batean idatziko baitira. Dokumentuen bidalketak bi pausu izango ditu, lehenengoan, dokumentuak METAPOSTako sFTP zerbitzarian utziko dira eta, bigarreanean, METAPOSTako erantzunen paketea irakurriko da, SAP sistemako karpeta batean utziz.

Hurrengo lerroetan, bi noranzkoetako komunikazio hau gauzatzeko *Enterprise Service Builder* eta *Integration Builder* baliabideetan sortutakoak deskribatuko dira.

### 5.3.1 Dokumentuak bidali

Dokumentuak SOAP mezuen Payload atalean joango direnez, ez da Mapping-ik behariko, ez eta datu/mezu mota desberdinik igorle eta hartzailearen artean.

Enterprise Service Builder

Enterprise Service Builder-ean sortutako mezu, datu eta interfazeak hauek dira:

- **Data Types**

*DT\_Envio* Datu mota sinplea (xsd:string).

- **Message Types**

*MT\_Envio* *DT\_Envio* datu motaren mezua.

- **Service Interfaces**

*SI\_Envio\_Outb\_Async* *MT\_Envio* mezu motaren interface igorlea (Outbound).

*SI\_Envio\_Inb\_Async* *MT\_Envio* mezu motaren interface hartzailea (Inbound).



## Integration Builder

Aurreko atal batean agertzen den moduan, komunikazioa konfiguratu baino lehen, bertan parte hartuko duten sistemak konfiguratu behar dira. Kasu honetan, bi sistema desberdin egongo dira, SAP makina (BS\_OAD100 hemendik aurrera) eta METAPOSTAko sFTP zerbitzaria (BC\_SFTP\_MTP hemendik aurrera). Lehenengoa, Business System bezala definitu da SLDan, bigarrena, ordea, Business Component bezala bakarrik definitu delarik Integration Builder-ean. Hauek biek, bi Komunikazio Kanal izango dituzte, File Adapter adaptadorea inplementatzen duen bat, eta sFTP Adapter adaptadorea inplementatzen duen beste bat.

Gauzak honela, eta sistemak prest daudela, komunikazioaren konfigurazioa honakoa izango da:

- **Sender Agreement**

*BS\_OAD100|SI\_Envio\_Outb\_Async* SAP sistemak interface igorlea bidaltzen duenean, fitxategien Komunikazio Kanal igorlea erabiltzen du.

- **Receiver Determination**

*BS\_OAD100|SI\_Envio\_Outb\_Async* SAP sistemak interface igorlea bidaltzen duenean, METAPOSTAko sFTP zerbitzarira bidaltzen du (BC\_SFTP\_MTP).

- **Interface Determination**

*BS\_OAD100|SI\_Envio\_Outb\_Async|BC\_SFTP\_MTP* SAPek interface igorlea bidaltzen dionean METAPOSTAko zerbitzariari, honek, SI\_Envio\_Inb\_Async interface-a erabilita jasoko du.

- **Receiver Agreement**

*BS\_OAD100|BC\_SFTP\_MTP|SI\_Envio\_Inb\_Async* METAPOSTAko sFTP zerbitzariak, SAP sistematik SI\_Envio\_Inb\_Async interface-a jasotzen duenean, sFTP Komunikazio Kanal hartzailetik jasoko du.

### 5.3.2 Erantzunak jaso

Erantzunak SOAP mezuen Payload atalean joango direnez, ez da Mapping-ik beharko, ez eta datu/mezu mota desberdinik igorle eta hartzailearen artean.

Enterprise Service Builder

Enterprise Service Builder-ean sortutako mezu, datu eta interfazeak hauek dira:

- **Data Types**

*DT\_Respuesta* Datu mota sinplea (xsd:string).

- **Message Types**

*MT\_Respuesta* *DT\_Respuesta* datu motaren mezua.

- **Service Interfaces**

*SI\_Respuesta\_Outb\_Async* *MT\_Respuesta* mezu motaren interface igorlea (Out-bound).

*SI\_Respuesta\_Inb\_Async* *MT\_Respuesta* mezu motaren interface hartzailea (In-bound).

Integration Builder

Erantzunak jasotzearen prozesuan parte hartuko duten sistemak, bidalketan parte hartzen duten berdinak dira, beraz, konfigurazioa bakarra izango da, aurretik definitutako alegia. Beraz, komunikazioaren konfigurazioaren deskribapena aztertuko da zuzenean:

- **Sender Agreement**

*BC\_SFTP\_MTP|SI\_Respuesta\_Outb\_Async* METAPOSTAko sFTP zerbitzariak interface igorlea bidaltzen duenean, sFTP Komunikazio Kanal igorlea erabiltzen du.

- **Receiver Determination**

*BC\_SFTP\_MTP|SI\_Respuesta\_Outb\_Async* sFTP zerbitzariak interface igorlea bidaltzen duenean, SAP sistemara bidaltzen du (BS\_OAD100).

- **Interface Determination**

*BC\_SFTP\_MTP\SI\_Respuesta\_Outb\_Async\BS\_OAD100* METAPOSTAk interface igorlea bidaltzen dionean SAP sistemari, honek, *SI\_Respuesta\_Inb\_Async* interface-a erabilia jasoko du.

- **Receiver Agreement**

*BC\_SFTP\_MTP\BS\_OAD100\SI\_Respuesta\_Inb\_Async* SAP sistemak, METAPOSTAtik *SI\_Respuesta\_Inb\_Async* interface-a jasotzen duenean, fitxategien Komunikazio Kanal hartzailetik jasoko du.

## 5.4 Erantzunen prozesamendua

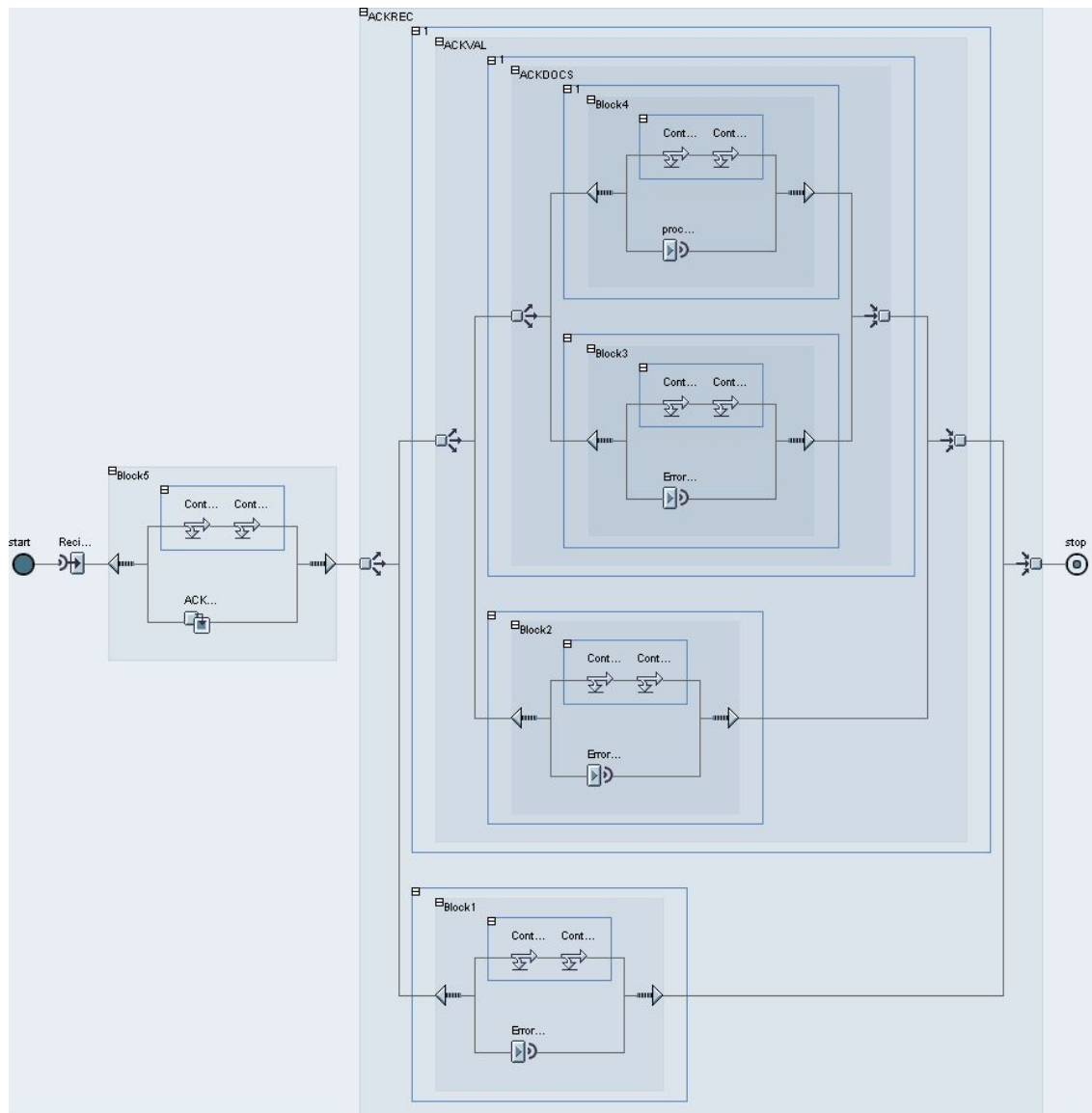
Erantzunen prozesamendua egiteko, Analisisian azaldu bezala, SAP PI sistema erabiliko da, haren *Integration Process* baliabidea, konkretuki. Honek, PIko beste sistema baten antzeko funtzionamendua du, baina igorle/hartzaile bakarra izan beharrean, horietako bat baino gehiago, edo biak batera, izan daiteke.

Baliabide hau erantzunen prozesamendu osoa egiteko baliatuko den arren, SAPen LOGa ere mantendu nahi denez, SAP sistemak ere parte hartuko du prozesu honetan, erantzunaren prozesamenduaren emaitzak gordetzeko. Hau egiteko, eta SAP PI beste SAP sistematik independentea izan dadin, Konektorearen inplementazio guztia duen makinan RFC funtzio multzoa bat sortu da, (*ZMTP\_FG\_ACKS*), PItik SOAP bitartez deitua izateko. Honela, LOGaren kudeaketa guztia toki berean egiten da, eta dokumentuen bidalketekiko independentea da.

Dei hauek egiteko, beste Adaptadore mota berri bat erabiliko da: RFC adaptadorea. Adaptadore honek, RFC deiak egiteko erabiltzen da, hartzailea SAP izanda edo ez. Adaptadore hau, aurretik aipatutako funtzio multzoko Function Module guztiei dei egiteko erabiliko da eta, Komunikazio Kanal bakar batetan inplementatu beharko da, hau, *BS\_OAD100* sisteman gongo delarik.

Gauzak honela, erantzunen prozesamendu honetan, bi sistema desberdinek parte hartuko dute, PI sistema fitxategien trukerako, eta SAP sistema LOGaren kudeaketarako. Hurrengo lerroetan, hauen bien diseinua deskribatuko da.

PI sistemari dagokionez, BPM lan-fluxu bat sortu da, erantzunak prozesatzeko.



**5.13 Irudia:** Integration Process

Irudian ikusten den lan fluxuaren hasieran, erantzunen ZIP fitxategia jaso (Receiver baten bidez) eta ACK desberdinetan banandu egiten da, ABAP Mapping baten bidez. Honekin, ACK bakoitzeko Container bat sortzen da, horren datuak irakurtzeko. Ondoren, hiru baldintza daude jarraian, bat ACK fitxategi bakoitzeko. Bakoitzean, ACK bat irakurri eta horren prozesamendua egokia izan den edo ez egiaztatzen da. Errorerik ez badago, hurrengo fitxategia aztertzen da. Honela, hirurak ondo badaude, RFC dei baten bidez, SAP sistema deitu egiten da LOGa eguneratzeko. Erroreren bat egotekotan, prozesua bertan bukatzen da eta RFC dei baten bidez, SAP sistema deitu egiten da erroreak LOGean gehitu ditzan.

Aurrekoetan bezala, PIren konfigurazio honetan ere, bi baliabide nagusietan sortu diren guztiak deskribatuko dira hurrengo lerroetan:

#### Enterprise Service Builder

Enterprise Service Builder-ean, Integration Process lan-fluxua sortu da (IP\_ACKS izeneko), horrek behar dituen interface eta mapeo guztiakin batera.

- **Operation Mappings**

*OM\_Respuesta* Mapping honek ZIP fitxategia irakurri eta lau RFC deiek behar dituzten datu egiturak sortzen ditu.

- **Service Interfaces**

*SI\_Respuesta\_Abst\_Async* *MT\_Respuesta* mezu motaren interface-a, ZIP fitxategia jasotzeko lan-fluxuaren hasieran.

*SI\_ACKxxx\_Abst\_Async* Interface bat behar da RFC dei bakoitzeko, horrek jaso behar duen datu egiturarekin.

OM\_Respuesta Operation Mapping-a, ABAP Mapping bat denez, hura PI makinan sortu behar da. Horretarako, *IF\_MAPPING* izeneko interfazea implementatzen duen klase bat sortu behar da, honek, hurrengo irudian ikusten den moduan, *EXECUTE* metodoa implementatu behar duelarik.

Método	Clase	Clase	T...	O...	Clase ti...	Tipo ref.	Valor propue...	Descripción
EXECUTE	Instance Method							Transformation ausführen
Parámetros	Clase	T...	O...	Clase ti...	Tipo ref.	Valor propue...		Descripción
SOURCE	Importi...	<input type="checkbox"/>	<input type="checkbox"/>	Type	XSTRING			Quell-XML-Dokument
PARAM	Importi...	<input type="checkbox"/>	<input type="checkbox"/>	Type Re...	IF_MAPPING_PARAM			Header Daten der Laufzeit für ...
TRACE	Importi...	<input type="checkbox"/>	<input type="checkbox"/>	Type Re...	IF_MAPPING_TRACE			Trace-Objekt
PARAMETERS	Importi...	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Type Re...	IF_MAPPING_PARA...			Mapping-Parametrisierung
DYNAMIC_CONFIGURATION	Importi...	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Type Re...	IF_MAPPING_DYNA...			Dynamische Konfiguration (API)
ATTACHMENTS	Importi...	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Type Re...	IF_MAPPING_ATT...			Attachments
RESULT	Exporti...	<input type="checkbox"/>	<input type="checkbox"/>	Type	XSTRING			Ziel-XML-Dokument

5.14 Irudia: IF\_MAPPING interfazea

## Integration Builder

Komunikazio Kanal berri bat sortu da, RFC deiak egiteko, CC\_RFCReceiver\_MTP ize-nekoa. Gainera, prozesua automatizatzeko, erantzunen fitxategia irakurtzerako momentu berean exekutatu da prozesamendua, beraz, aurretik definitutako *Sender Agreement* be-  
ra erabiliko da, *Receiver Determination* konfigurazioan beste hartzaile bat sartu delarik, lan-fluxua gehitzeko. Honetaz gain, hurrengo konfigurazioak sortu dira:

- **Interface Determination**

*IP\_ACKS\SI\_ACKxxx\_Abst\_Async\BS\_OAD100* Integration Process-ak ACK in-  
terface bat bidaltzen duenean, SAP sistemak Function Module batetik jasoko  
du, hau da, funtzio hori, interface-an dauden datuekin exekutatu du.

- **Receiver Agreement**

*IP\_ACKS\BS\_OAD100\ZMTP\_FM\_ACKxx* SAP sistemak, lan-fluxutik interface  
bat jasotzen duenean, hura RFC funtzio baten bidez jasoko du.

SAP PIko diseinua aztertuta, orain, SAP sisteman sortutako Function Group-aren disei-  
nua deskribatuko da. Esan bezala, multzo hau osatzen duten funtzioak RFC deien bidez  
deituak izango dira, ACK fitxategien datuekin. Hurrengo irudian ACK fitxategiek izango  
duten egitura agertzen da, hau izango delarik ere, funtzio moduluak jasoko dituzten da-  
tuen egitura. Prozesua ondo joan delaren funtzio moduluak bidalketaren identifikadorea  
baino ez du jasoko, ACK fitxategi hauek erroreak baino ez baitituzte adierazten.

▼ MP_ACK_DOC_TYPE	Descripcion del formato de comprobacion de envio de
send_ID	Identificador del envio. Formato
send_ReceptionStatus	Estado del envio: P (en_proceso), V (validado),
▼ send_PackageError	Error a nivel de paquete.
send_ErrorCode	Código de error a nivel de paquete
send_Error_Description	Descripción del error a nivel de paquete
▼ send_ErrorCodes	Si hubiera errores notificación de los mismos
▼ send_errors	
send_ErrorCode	Codigo de error: 00019 (Tipo de documento no
send_Error_Description	Descripcion del error
send_Doc_Ref_Sender	Identificador de referencia del Documento del
identificadorDeDocumentoDeUsuario	Indica que número de documento de identidad
send_Signature	Firma del paquete
▼ send_Notification	Datos de la notificación
▼ send_timestamp	Sello del tiempo del paquete
signature	Firma digital con sello de tiempo
hashSignature	Codigo hash del .zip del paquete obtenido
fExactaSignature	Fecha exacta de generacion de sello de tiempo
servidorSignature	Servidor que garantiza ese sello de tiempo
terceroSignature	Identidad del tercero de confianza que ha realizado dicho sellado

### 5.15 Irudia: ACK fitxategien egitura

## 5.5 Datu Baseko taulak

Tabla	Descripción breve
ZMTP_T101	Tabla de parametrización para datos propios del SENDER
ZMTP_T201	Tabla de parametrización para búsqueda de empleados
ZMTP_T202	Tabla de parametrización para recibos de nómina
ZMTP_T203	Tabla de empleados suscritos a METAPOSTA
ZMTP_T301	Tabla de parametrización para búsqueda de clientes
ZMTP_T303	Tabla de clientes suscritos a METAPOSTA
ZMTP_T501	Tabla de datos de cabecera para envíos
ZMTP_T502	Tabla de datos para detalles de envíos a usuarios

### 5.16 Irudia: Datu Baseko taulak

Konektoreak, SAPen garatzen den edozein aplikaziok bezala, datu basea erabiliko du datuak kudeatzeko. Datu baseak, SAP sistemetan, bi atal ditu, defektuzko atala, zeinetan sistemaren defektuzko taulak agertzen diren, sistemaren defektuzko aplikazioek erabiltzen dituztenak, eta bezero atala, edo atal pertsonalizatua, zeinetan erabiltzaile bakoitzak bere beharretara egokitutako taulak sor ditzakeen.

Irudian agertzen diren taulak, Konektoreak erabiliko dituen eta proiektu honen garapenerako sortu diren bezero taulak dira. Hauek, bere erabileraren arabera, lau multzo desberdinetan banatzen dira; honela, Web Service-ean erabiltzeko taulak, erabiltzaileak erregistratzeko taulak, dokumentuen sorketa konfiguratuzko taulak eta bidalketak erregistratzeko taulak ditugu.

Hurrengo lerroetan, taula hauek deskribatuko ditugu, zein motatakoak diren eta zertarako erabiliko diren adieraziz.

#### 5.5.1 Web Service-erako taulak

Web Service-a eskaintzeko hiru taula desberdin erabiliko dira, bat Sender-aren parametrizazio orokorretarako (ZMTP\_T101), beste bat langileen bilaketarako (ZMTP\_T201) eta, bestea, bezeroen bilaketarako (ZMTP\_T301).

## ZMTP\_T101

Taula honetan, METAPOSTako Web orrialdearen eta web zerbitzuaren arteko komunikazioa egiteko beharrezkoak diren Sender-aren datu propioak gordeko dira.

Campo	Descripción breve
<u>MANDT</u>	Mandante
<u>OIDSENDER</u>	Identificador (OID) del Sender
<u>CARPETA</u>	Nombre de carpeta

**5.17 Irudia:** Datu Basea: ZMTP\_T101

Irudian ikusten den bezala, momentuz, Sender-aren segurtasun identifikadorea (OIDSender) eta paketeak biltegitratzeko direktorioa (Carpeta) gordetzen dira. Hala ere, taula hau handitzea espero da, Sender bakoitzak bere konfigurazio propioetara egokitu dezan.

## ZMTP\_T201 eta ZMTP\_T301

Taula hauetan erabiltzaileen datuen bilaketa parametrizatuko da, horretarako, METAPOSTatik jasotzen diren datuak SAPeko zein taula estandarrekin parekatu behar diren zehazten da. Honela, erabiltzaileak sisteman identifikatuko dira, hauek HR moduluko langileak (ZMTP\_T201) zein FI moduluko bezeroak (ZMTP\_T301) izan.

Erabiltzaileen identifikaziorako datu laguntzaileak beharrezkoak diren edo ez eta, suskripzio prozesuan, erabiltzailearen helbide elektronikoa behar den ere zehaztuko dira.

Campo	Descripción breve
<u>MANDT</u>	Mandante
<u>FNACIMIENTO</u>	Fecha de nacimiento del usuario (Formato AAAA-MM-DD)
<u>AUXDATA</u>	Dato auxiliar enviado al sender para sus comprobaciones
<u>AUXDATA2</u>	Dato auxiliar enviado al sender para sus comprobaciones
<u>RECEPCIONCORREOUSUARIO</u>	Indica si hay que pedir o no al usuario el email para enviar

**5.18 Irudia:** Datu Basea: ZMTP\_T201 eta ZMTP\_T301

Taula hauek biak egituran berdinak izango diren arren, datu desberdinak gordeko dituzte, Sender gehienetan, hauetako bakarra baino erabiliko ez delarik.



### 5.5.2 Erabiltzaileen harpidetzarako taulak

Sender-era harpidetutako erabiltzaile guztiak SAP sisteman erregistratuko dira, horretarako diseinatutako tauletan, dokumentuen bidalketan eta harpidetzaren aldaketa prozesuetan atzigarri egon daitezzen.

Taula hauen helburua, erabiltzaileak METAPOSTAn dauzkaten datu berdinekin erregistratzea da, honela, datuak bi tokietan parekatuta gelditzen dira modu unibokoan identifikatuak izateko.

Hurrengo lerroetan ikusiko den bezala, taula hauetan gordeko diren datuen artean ez daude aurretik aipatutako datu laguntzaileak, hauek ez baitira METAPOSTAn gordetzen eta identifikatzeko garaian bakarrik erabiltzen direlako.

Adierazitako datu hauek gordetzeko, bi taula desberdin erabiliko dira, bat langileak erregistratzeko (ZMTP\_T203) eta bestea bezeroak erregistratzeko (ZMTP\_T303).

#### ZMTP\_T203

Taula honetan Sender-ean noizbait harpidetuta egon diren langile motako erabiltzaile guztiak erregistratuko dira, aurretik azaldu den bezala, SAP sistemak dituen hauen datuak METAPOSTAn dituztenekin parekatuta egon daitezzen.

METAPOSTAko datu propioetaz gain, SAPEk erabiltzaileak identifikatzeko erabiltzen duen langile zenbakia (PERNR), harpidetzaren egungo egoera (suskribatuta bai edo ez) eta honen azkeneko aldaketaren data gordeko dira.

Campo	Descripción breve
<u>MANDT</u>	Mandante
<u>PERNR</u>	Número de personal
<u>PERID</u>	Número de identificación personal
<u>GBDAT</u>	Fecha de nacimiento
<u>FECHA</u>	Fecha de suscripción a sender Metaposta
<u>SUSCRITO</u>	Indicador de si está suscrito a Metaposta

**5.19 Irudia:** Datu Basea: ZMTP\_T203

## ZMTP\_T303

Taula honetan Sender-ean noizbait harpidetuta egon diren bezero guztiak bilduko dira, langileekin egiten den bezala.

Gainera, METAPOSTAko datuekin batera, SAPen duten bezero zenbakia (KUNNR), harpidetzaren egoera (suskribatuta/des-suskribatuta) eta honen azkenengo aldaketaren data gordeko dira.

Campo	Descripción breve
<u>MANDT</u>	Mandante
<u>CLIENTE</u>	Número de deudor
<u>STCD1</u>	Número de identificación fiscal 1
<u>FECHA_CONSTITUCION</u>	Fecha
<u>FECHA_SUSCRIPCION</u>	Fecha de suscripción a sender Metaposta
<u>SUSCRITO</u>	Indicador de si está suscrito a Metaposta

**5.20 Irudia:** Datu Basea: ZMTP\_T303

### 5.5.3 Dokumentuen sorketarako taulak

Dokumentuak sortzeko garaian ere, parametrizazio taulak erabiliko dira, datuak nola eta zein formatutan inprimatuko diren zehazteko. Taula honen helburua, Sender-ek kodea aldatzeko beharra murriztea da, honela, errazagoa eta seguruagoa da Sender bakoitzak bere datu propioak taula batean sartzea, kodean aldaketak egitea baino.

Proiektu honetan sortuko diren dokumentuetarako, taula bakarra erabiliko da (ZMTP\_T202) nominen ordainagirientzako. Hala ere, Sender bakoitzak taula hau aldatu, edo berriak sortzeko aukera izango du, bere dokumentuen sorketa sistema propioa konfiguratzeko.

## ZMTP\_T202

Taula honetan, sortu eta METAPOSTAra bidaliko diren nomina agirientzako formulario mota (Tipo) eta izena (Formulario) gordeko dira, hauen sorketa prozesuan erabiltzeko.

Formulario mota deitu zaio SAPEk agiriak sortzeko dituen moduari, honela, hauek dira honek izan ditzakeen balioak: 'HRFORM', 'SAPScript' eta 'SmartForm'. Sender bakoitzak hauetako bat aukeratuko du, bere konfigurazioaren arabera.

Campo	Descripción breve
<u>MANDI</u>	Mandante
<u>TIPO</u>	Tipo de Recibo de nómina
<u>FORMULARIO</u>	Nombre del Formulario para recibo de nómina

#### 5.21 Irudia: Datu Basea: ZMTP\_T202

Esan bezala, taula hau Sender bakoitzaren parametrizazioekin eguneratuko da, taula berriak sortzeko aukera ere edukiko duelarik.

#### 5.5.4 Bidalketak erregistratzeko taulak

Konektorearen bidez egiten diren bidalketa guztiak datu basean erregistratuta geratuko dira, ondorengo lerroetan deskribatuko diren bi taulen bitartez, ZMTP\_T501 bidalketen datu orokorretarako eta ZMTP\_T502 hauen detaileentzako.

#### ZMTP\_T501

Taula honetan bidalketaren datu orokorrak bilduko dira: bidalketaren identifikadorea, data, ordua, data horretako bidalketa zenbakia eta hiru ACK erantzun fitxategien egoera.

Campo	Descripción breve
<u>MANDI</u>	Mandante
<u>IDENTIFICADOR</u>	Identificador del envío a METAPOSTA
<u>FECHA</u>	Fecha
<u>HORA</u>	Hora del día
<u>NUMENVIO</u>	Número de envío a METAPOSTA
<u>ACKREC</u>	Status ACK de recepción
<u>ACKVAL</u>	Status ACK de validación
<u>ACKDOCS</u>	Status de ACK de documentos

#### 5.22 Irudia: Datu Basea: ZMTP\_T501

Erantzun fitxategien egoerak hiru balio posible izan ditzake, 'S' arrakasta (success), 'E' errorea eta 'P' oraindik prozesatzen (processing).

## ZMTP\_T502

Taula hau bidalketen zehetasunak erregistratzeko erabiliko da, honela, bidalketa bakoitze-ko, bidali diren dokumentuak eta hauen hartzaileen datuak gordeko dira. Taula honetan zer bidali den eta nori zehaztuko da, beraz.

Campo	Descripción breve
<u>MANDI</u>	Mandante
<u>IDENTIFICADOR</u>	Identificador del envío a METAPOSTA
<u>FICHERO</u>	Nombre de documento
<u>DOCUMENTO</u>	Número de documento de identidad
<u>FNACIMIENTO</u>	Fecha de nacimiento del usuario (Formato AAAA-MM-DD)


**5.23 Irudia:** Datu Basea: ZMTP\_T502

## 5.6 LOGa

Konektoren gertatzen den guztia monitorizatuta izateko, LOG bat sortu da SAPen, *SLG0* transakzioaren bitartez. Bertan, *ZMTP* objektua sortu da, bi azpiobjektuekin batera, *SENDERS* eta *ENVIOS*. Objektu hauen bitartez, erabiltzaileen harpidetzaren (*SENDERS*) eta bidalketen (*ENVIOS*) berri emango da prozesu bakoitzaren exekuzioan.

LOGa kontsultatu ahal izateko, SAPeko *SLG1* transakzioa erabiliko da.

***Evaluar log aplicación***



Objeto   Log Metaposta

Objeto inferior

Identif.ext.

---

Restricción temporal




de (fecha/hora)

a (fecha/hora)
















**5.24 Irudia:** LOGa *SLG1* transakzioa

Sartu diren objektu eta datetarako LOG sarrerak agertuko dira zerrenda batean. Zerrenda honetatik sarrera bat aukeratzuz gero, horren mezuak bistaratuko dira, gertatutako guztia ikusi ahal izateko.

**Visualizar logs**



 Información técnica
 
 Ayuda

Fecha/Hora/Usuario	Ctd.	Texto obj.	Texto objeto inf.
21.02.2014 08:19:26 ABD	7	Log Metaposta	Log para envíos de documentos a Metaposta
21.02.2014 08:27:34 ABD	6	Log Metaposta	Log para envíos de documentos a Metaposta
21.02.2014 08:30:53 ABD	5	Log Metaposta	Log para envíos de documentos a Metaposta

Tipo: Texto de mensaje

- Proceso de envío iniciado
- Obtener Recibo de nómina: Empleado:00000101 Periodo:2012-12
- Obtener Recibo de nómina: Empleado:00000101 Periodo:2012-12
- Añadir fichero '20180854X\_nómina\_20121201-20121231.PDF' al ZIP
- Envío 'AD1097E707A7F09813085CC1964D3A3420140221000000002' preparado
- Proceso de envío finalizado

### 5.25 Irudia: LOGa: Mezuak



## 6. KAPITULUA

---

### Garapena

---

Kapitulu honetan, aurrekoetan aztertu eta diseinatu den **SAP - METAPOSTA konektorearen** garapena deskribatuko da. Horretarako, konektorearen lau atal nagusien inplementazioak aztertuko dira, ABAP programazio lengoaiari idatzitako kode zatiak aztertuz.

Hurrengo lerroetan, beraz, Web Service-aren, dokumentuen sorketaren, dokumentuen birkalketaren eta erantzunen prozesamenduaren garapenak agertuko dira.

#### 6.1 Web Service-a

Atal honetan, Metapostako erabiltzaileen harpidetza kudeatzeko konektoreak inplementatu eta atzigarri uzten dituen web zerbitzuen inplementazioa aztertzen da.

Azaldu den moduan, bi metodo dira atzigarri utzi direnak, *SENDERPeticionTipoUsuario* eta *SENDERPeticionSuscripcion*, harpidetza prozesua honelakoa izango delarik:

1. Lehendabizi, METAPOSTAK erabiltzaile mota jakiteko metodoa deituko du, erabiltzailea egokia dela zihurtazeko.
2. Ondoren, harpidetza metodoa deituko da, alta/baja eskaera eginez.

Hurrengo lerroetan, metodo hauen bien garapena deskribatuko da.

### 6.1.1 Logika komuna

Bi metodo hauek, hasieran, Sender-aren parametrizazioak irakutzen dituzte, ondorengo guztia modu egokian egiten dela zihurtatzeko.

```

*-----*
*      Obtener los parámetros definidos por el cliente.
*-----*
FORM obtener_parametros.

* Parámetros propios del SENDER
  SELECT SINGLE * INTO gs_t101 FROM zntp_t101.

* Parámetros para la búsqueda de empleados
  SELECT SINGLE * INTO gs_t201 FROM zntp_t201.

* Parámetros para la búsqueda de clientes
  SELECT SINGLE * INTO gs_t301 FROM zntp_t301.

ENDFORM.

```

#### 6.1 Irudia: Web Service-ak: parametrizazioa irakurri

Honen ondoren, jasotako datuak irakurri, egiaztatu eta parametrizazioekin bat datozela zihurtatzen da. Datu hauetakoren bat egokia ez bada, exekuzioa zuzenean bukatu egiten da, aurkitutako errorearekin erantzuten delarik.

```

* Comprobar datos de entrada
PERFORM input_check_tipousuario USING fechanacimiento oidsender
                                     CHANGING codigorespuestadeejecucion descripcioncodigorespuestadeej.

fnacimiento = fechanacimiento.
IF codigorespuestadeejecucion IS NOT INITIAL.
  CONCATENATE codigorespuestadeejecucion ':' descripcioncodigorespuestadeej INTO lv_aux.
  PERFORM mensaje USING 4 'E' 003 lv_aux. "Error en el input
  PERFORM guardar_log USING 'TipoUsuario'.
  RETURN.
ENDIF.

* Comprobar datos de entrada
PERFORM input_check_tipousuario USING fechanacimiento oidsender
                                     CHANGING codigorespuestadeejecucion descripcioncodigorespuestadeej.

fnacimiento = fechanacimiento.
IF codigorespuestadeejecucion IS NOT INITIAL.
  CONCATENATE codigorespuestadeejecucion ':' descripcioncodigorespuestadeej INTO lv_aux.
  PERFORM mensaje USING 4 'E' 003 lv_aux. "Error en el input
  PERFORM guardar_log USING 'TipoUsuario'.
  RETURN.
ENDIF.

```

#### 6.2 Irudia: Web Service-ak: Datuak egiaztatu

Datuak egokiak izatekotan, metodo bakoitzaren prozesamenduarekin hasten da.



### 6.1.2 PeticionTipoUsuario

Funtzio hau erabiltzaileak sisteman bilatzeaz arduratzen da. Horretarako, bi subrutina desberdin daude, bat 1 eta 3 motako erabiltzaileak (langileak, NIF bidez bilatuak) bilatzeko eta, bestea, 2 motako erabiltzaileak (bezeroak, CIF bidez bilatuak) bilatzeko. Bi subrutina hauek, Sender-eko datuen bidez (NIF, CIF, jaiotze data, datu laguntzaileak, ...), erabiltzaile bat bilatzen dute, aurkitzekotan, honen identifikazio zenbakia bueltatzen dutelarik.

```
* Buscar empleados / clientes en el sistema
CASE tipodocumentoidentidad.
  WHEN '1' OR '3'. "NIF / Tarjeta Residente

    " Buscar al empleado en la BD
    PERFORM buscar_empleado USING documento fechanacimiento auxdata auxdata2
      CHANGING lv_pernr codigorespuestadeejecucion descripcioncodigorespuestadeej.

    IF codigorespuestadeejecucion IS INITIAL. "Controlar respuesta
      ...
    ENDIF.

  WHEN '2'. "CIF

    "Buscar cliente en el sistema
    PERFORM buscar_cliente USING documento fechanacimiento auxdata auxdata2
      CHANGING lv_kunnr codigorespuestadeejecucion descripcioncodigorespuestadeej.

    IF lv_kunnr IS NOT INITIAL. "Controlar respuesta
      ...
    ENDIF.

  WHEN OTHERS. "No válido

    codigorespuestadeejecucion = '11'.
    descripcioncodigorespuestadeej = text-011.
    CONCATENATE codigorespuestadeejecucion ':' descripcioncodigorespuestadeej INTO lv_aux.
    PERFORM mensaje USING 4 'E' 003 lv_aux. "Error en el input

ENDCASE.
```

### 6.3 Irudia: Web Service-ak: PeticionTipoUsuario

Irudian ikusten den moduan, aurretik aipatu diren erabiltzaileak bilatzeko subrutinak *buscar\_empleado* eta *buscar\_cliente* dira. Hauek, Web Service-eko bi metodoetan erabiliko da, harpidetzak egiteko ere, erabiltzaileak bilatu behar baitira. Subrutina hauen inplementazioa ez da aztertuko, aurreko kapituluan adierazitako datu baseko taulen irakurketak baino ez direlako.

Metodo honen exekuzioaren amaieran, gertatutakoaren berri ematen da erantzunean. Honela, erabiltzailea aurkitu bada, honen erabiltzaile mota bueltatuko da, sarrerako datuekin batera. Erabiltzailea aurkitu ezean, errore batekin erantzungo da.

### 6.1.3 PeticionSuscripcion

Metodo honetan, aurrekoan jarraitzen den logika berdina jarraitzen da, kasu honetan, erabiltzailea aurkitzen denean bukatu orde, bere harpidetza kudeatzen delarik. Honela, erabiltzailea existitzen dela egiaztatu ondoren, bere Sender-eko alta/baja prozesua exekutatzen da.

Alta prozesuan, lehenabizi, erabiltzailea aurretik harpidetuta ez zegoela egiaztatzen da, egotekotan, errore bat bueltatu eta prozesua bukatzen delarik. Harpidetza berria dela zihuratzen denean, erabiltzaileari alta ematen zaio:

Langile bati alta emateko, ZMTP\_T203 datu baseko taulan erregistro berri bat sortzen da, aurreko atalean deskribatutako datuekin.

```
* Leer datos del empleado
SELECT SINGLE *
  INTO ls_t203
  FROM zmtp_t203
  WHERE pernr = pe_pernr.

IF ls_t203-suscrito <> 'X'. "No está dado de alta

  ls_t203-pernr = pe_pernr.
  ls_t203-fecha = sy-datum.
  ls_t203-perid = pe_documento.
  CONCATENATE pe_fnacimiento(4) pe_fnacimiento+5(2) pe_fnacimiento+8(2)
    INTO ls_t203-gbdat.
  ls_t203-suscrito = 'X'.
  MODIFY zmtp_t203 FROM ls_t203.

  ps_codigorespuesta = '00'.
  ps_descripcioncodigo = text-000.
  PERFORM mensaje USING 4 'I' 009 pe_pernr. "Empleado 'pe_pernr' dado de alta

ELSE. "Ya está dado de alta

  ps_codigorespuesta = '07'.
  ps_descripcioncodigo = text-007.
  DATA: lv_aux TYPE string.
  CONCATENATE ps_codigorespuesta ':' ps_descripcioncodigo INTO lv_aux.
  PERFORM mensaje USING 4 'E' 003 lv_aux. "Error

ENDIF.
```

### 6.4 Irudia: Web Service-ak: Alta eman

Langile bat izan beharrean, erabiltzailea, bezeroa bada, prozesua berdina da, baina, ZMTP\_T303 taula erabilia, honen datu propioekin.

Baja prozesua, alta prozesuaren antzekoa da, kasu honetan, erregistro berriak sortu behar-rean, jada sortutako erregistroetan suskripzioa bukatu dela adierazten delarik *suscrito* aldagaia ezabatuz. Erregistroa ezabatzeko aukera ere aztertu zen, baina, datuak mantentzea erabaki zen, etorkizunean harpidetza berri nahi izatekotan, datuak jada parekatuta egoteko.

Baja prozesua ere erabiltzailearen egoera egiaztatuz hasten da, baja eskaera errepikatuta izatekotan (erabiltzailea jada baja emanda zegoelako), errorea bueltatuko delarik.

Orain ere, prozesua berdina izango da erabiltzaile mota guztientzako, datu baseko taula aldatuko delarik bakarrik.

```
* Leer datos del cliente
SELECT SINGLE *
  INTO ls_t303
  FROM zmtp_t303
  WHERE cliente = pe_kunnr.

IF ls_t303-suscrito = 'X'. "Está dado de alta

  "Dar de baja al cliente
  UPDATE zmtp_t303 SET suscrito = space WHERE cliente = pe_kunnr.

  ps_codigorespuesta = '00'.
  ps_descripcioncodigo = text-000.
  PERFORM mensaje USING 4 'I' 017 pe_kunnr. "Cliente pe_kunnr dado de baja

ELSE. "No está dado de alta

  ps_codigorespuesta = '08'.
  ps_descripcioncodigo = text-008.
  DATA: lv_aux TYPE string.
  CONCATENATE ps_codigorespuesta ':' ps_descripcioncodigo INTO lv_aux.
  PERFORM mensaje USING 4 'E' 003 lv_aux. "Error en el input

ENDIF.
```

### 6.5 Irudia: Web Service-ak: Baja eman

Hurrengo irudian, Web Service honekin lortutako harpidetuen zerrenda baten adibidea ikus daiteke.

Número de personal	NIF	Fecha de nacimiento	Fecha suscripción
101	20180854X	22.08.1970	06.02.2014
102	11111111X	16.02.2011	12.02.2014

### 6.6 Irudia: Web Service-ak: Harpidetuen zerrenda

## 6.2 Dokumentuen sorketa

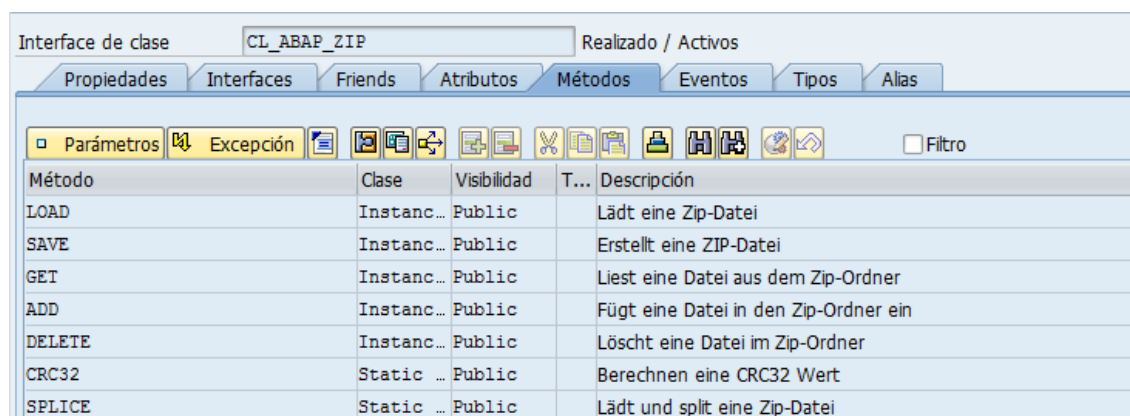
Dokumentuen sorketa prozesuan, Diseinuan deskribatu den moduan, bi programa erabiliko dira Sender-eko SAP sistematik, hauek, era berean, klase nagusi bateko metodoak deituko dituztelarik.

Prozesu honetan, *ZMTP\_REP\_NOMINAS* eta *ZMTP\_REP\_FACTURAS* Report-ak erabiliko dira, hauek, *ZMTP\_CL\_METAPOSTA* klaseko metodoak baliatuko dituztelarik. Memoria honetan jada aipatu den bezala, bidaliko diren dokumentuak sortzeko SAP sistemako baliabide estandarrak erabiliko dira, eta, hori dela eta, ez da hauen inplementazioan gehiegi sakonduko.

Hurrengo lerroetan aztertuko dena, beraz, erabili diren baliabide berezien garapena izango da. Baliabide berezi hauek, ZIP fitxategiak sortzeko erabili den **CL\_ABAP\_ZIP** klasea eta XML fitxategiak sortzeko erabili den **Transformation** tresna izan dira.

### 6.2.1 CL\_ABAP\_ZIP

SAPeko klase estandar hau, ZIP fitxategien kudeaketa egiteko baliatu da. Honek *Xstring* datu motarekin egiten du lan, bere erabilera oso erraza delarik. Hurrengo irudian, klase honek dituen metodoak agertzen dira.



Método	Clase	Visibilidad	T...	Descripción
LOAD	Instanc...	Public		Lädt eine Zip-Datei
SAVE	Instanc...	Public		Erstellt eine ZIP-Datei
GET	Instanc...	Public		Liest eine Datei aus dem Zip-Ordner
ADD	Instanc...	Public		Fügt eine Datei in den Zip-Ordner ein
DELETE	Instanc...	Public		Löscht eine Datei im Zip-Ordner
CRC32	Static ...	Public		Berechnen eine CRC32 Wert
SPLICE	Static ...	Public		Lädt und split eine Zip-Datei

**6.7 Irudia:** Dokumentuen sorketa: *CL\_ABAP\_ZIP*

Proiektu honetan, metodo horietako gehienak erabili dira, garrantzitsuenak *ADD*, dokumentuak gehitzeko, eta *SAVE*, ZIP fitxategia lortzeko, direlarik.

### 6.2.2 Transformation

Transformation Editor izeneko SAP baliabideak, aplikazio zerbitzarian exekutatu diren transformazioak definitzea ahalbidetzen du. Honen bitartez, XML dokumentuak beste XML edo HTML dokumentuetara itzul daitezke, hauen datuak ABAP datu-egituretan gorde daitezkeelarik ere.

Transformation baliabide honen bitartez, bi protokolo desberdin erabil daitezke transformazioak egiteko. Alde batetik, Simple Transformation izeneko transformazioak sor daitezke, ABAP datu-egiturak XMLra bihurtzeko (*serialization* edo XML fitxategiak ABAP datu-egituretan gordetzeko (*deserialization*), eta, bestetik, XSLT (*Extensible Stylesheet Language Transformations*) programak ere sor daitezke XML eta HTML fitxategiekin lan egiteko.

#### Simple Transformation

Protokolo honen bidez, bidaliko diren XML fitxategiak sortu dira. Honela, dokumentuen sorketa prozesuan, ABAP datu egiturekin egiten da lan, beharrezkoak diren datuak gordetzeko, amaieran, egitura hauek XML fitxategietan serializatzen direlarik.

The screenshot displays the SAP Transformation Editor interface, divided into two main panels: 'Data roots' on the left and 'Simple Transformation' on the right.

**Data roots (Left Panel):**

- ROOT
  - MP\_ACK\_DOC\_TYPE : ZMTP\_ST\_ACK
    - SEND\_ID : ZMTP\_DE\_IDENTIFICADOR
    - SEND\_RECEPTIONSTATUS : ZMTP\_DE\_RECEPTIONSTATUS
    - SEND\_PACKAGEERROR : ZMTP\_ST\_PACKAGEERROR
      - SEND\_ERRORCODE : ZMTP\_DE\_ERRORCODE
      - SEND\_ERROR\_DESCRIPTION : ZMTP\_DE\_ERRORDESCRIPTION
    - SEND\_ERRORCODES : ZMTP\_TT\_ERRORCODES
      - \* : ZMTP\_ST\_ERRORS
        - SEND\_ERRORCODE : ZMTP\_DE\_ERRORCODE
        - SEND\_ERROR\_DESCRIPTION : ZMTP\_DE\_ERRORDESCRIPTION
        - SEND\_DOC\_REF\_SENDER : ZMTP\_DE\_DOCREFSENDER
        - IDENTIFICADORDEDOCUMENTODEUSUA : ZMTP\_DE\_DOCUMENTO
    - SEND\_SIGNATURE : ZMTP\_DE\_SIGNATURE
    - SEND\_NOTIFICATION : ZMTP\_TT\_NOTIFICATION
      - \* : ZMTP\_ST\_TIMESTAMP
        - SIGNATURE : ZMTP\_DE\_SIGNATURE
        - HASHSIGNATURE : ZMTP\_DE\_HASHSIGNATURE
        - FEXACTSIGNATURE : ZMTP\_DE\_FEXACTSIGNATURE
        - SERVIDORSIGNATURE : ZMTP\_DE\_SERVIDORSIGNATURE
        - TERCEROSIGNATURE : ZMTP\_DE\_TERCEROSIGNATURE

**Simple Transformation (Right Panel):**

- Transformation templates:
  - xmlns:tt="http://www.sap.com/transformation-templates"
  - xmlns:ddic="http://www.sap.com/abapxml/types/dictionary"
  - xmlns:def="http://www.sap.com/abapxml/types/defined"
- MP\_ACK\_DOC\_TYPE
  - send\_ID
  - send\_ReceptionStatus
  - send\_PackageError
  - send\_ErrorCodes
    - send\_Errors
      - send\_ErrorCode
      - send\_Error\_Description
      - send\_Doc\_Ref\_Sender
      - identificadorDeDocumentoDeUsuario
  - send\_Signature
  - send\_Notification
    - send\_timestamp
      - signature
      - hashSignature
      - fExactSignature
      - servidorSignature
      - terceroSignature

**6.8 Irudia:** Dokumentuen sorketa: Simple Transformation

## XSLT

Protokolo hau, aurrekoaren bidez sortutako XML fitxategiak formatu egokian uzteko erabili da. Honen bidez, utzik dauden XML elementuak ezabatu eta goiburuko egokiak jarri zaizkie bidaltzen diren XML fitxategiei, bidalketetarako formatu egokia izan dezaten.

```
<xsl:transform xmlns:xsl="http://www.w3.org/1999/XSL/Transform" xmlns:sap="http://www.sap.com/sapxsl" version="1.0">
  <xsl:output indent="yes"/>
  <xsl:strip-space elements="*" />
  <xsl:template match="node()|@">
    <xsl:copy>
      <xsl:apply-templates select="node()|@" />
    </xsl:copy>
  </xsl:template>
  <xsl:template match="*[not(//@*) and not(normalize-space())]" priority="3" />
  <xsl:template match="*">
    <xsl:element name="{name()}" namespace="com/metaposta/mp00d/schemas/mp00dIntegracionET">
      <xsl:copy-of select="namespace:*" />
      <xsl:apply-templates select="node()|@" />
    </xsl:element>
  </xsl:template>
</xsl:transform>
```

### 6.9 Irudia: Dokumentuen sorketa: XSLT

Transformation hauek ABAP programa batean erabiltzeko, 'CALL TRANSFORMATION' komandoa erabiltzen da, honek, ABAP datu-egiturak eta Xstring aldagaiak erabiltzen dituelarik parametro bezala.

```
CALL TRANSFORMATION zmtf_tf_log
SOURCE mp_ok_doc_type = ls_log
RESULT XML lv_log_xstring.

CALL TRANSFORMATION zmtf_tf_format
SOURCE XML lv_log_xstring
RESULT XML lv_log_xstring.
```

### 6.10 Irudia: Dokumentuen sorketa: CALL TRANSFORMATION

Proiektu honetan, irudian ikusten den bezala, bi transformazioak elkarrekin erabili dira, honela, dokumentuen sorketa prozesuan zehar mantendutako datu egiturak, XML batean serializatu dira *Simple Transformationen* bidez, ondoren, XSLT bitartez, formatu egokian uzteko.

## 6.3 Dokumentuen bidalketa

Dokumentuen bidalketaren prozesu osoa, SAP PIren bitartez egin da. Honela, Diseinuan azaldutako konfigurazioa erabiliz, Konektoreak dokumentuen paketeak SAP sistematik hartu eta METAPOSTAko zerbitzarian uzten ditu, erantzunen paketeak kontrako bidea egiten dutelarik.

Hurrengo lerroetan, aurreko kapituluaren definitutako hainbat SAP PIko elementu aztertuko dira, hauek nola konfiguratzen diren zehatz ikusteko.

### 6.3.1 Komunikazio Kanalak

Jada azaldu da lau *Communication Channel* behar direla bidalketa prozesua egiteko, hauek bi mota desberdinekoak izango direlarik. Honela, hurrengo irudian, hauetako bat ikus daiteke, hauen konfigurazioa nolakoa den ikusteko.

The screenshot shows the SAP PI Configuration Wizard interface. The 'Parameters' tab is selected, displaying the following configuration:

- Adapter Type:** File
- Transport Protocol:** File System (NFS)
- Message Protocol:** File
- Adapter Engine:** Central Adapter Engine

The 'Source' tab is also visible, showing the following configuration:

- Source Directory:** /tmp
- File Name:** AD1097E707A7F09813085CC1964D3A34\*
- Advanced Selection for Source File
- Additional File(s)

**6.11 Irudia:** Dokumentuen bidalketa: *Communication Channel*

Irudian ikusten den Komunikazio Kanala, prozesu honi guztiari hasiera ematen diona da, paketeak SAP sistemako karpeta batetik (kasu honetan, */tmp*) irakurtzen dituen, hain zuzen ere. 'File Adapter' izeneko adaptadorea baliatzen duenez, fitxategiaren izena eta karpeta adierazi behar zaizkio, beste adaptadoreek, haien konfigurazio propioa izango dutelarik.

Proiektu honetan erabili diren beste Komunikazio Kanalek ere antzeko konfigurazioa daukate, eta, beraz, ez dira memoria dokumentu honetan azalduko.

### 6.3.2 Sender Agreement

Diseinuan azaldu den bezala, lau konfigurazio desberdin sortu behar dira SAP PIn edozein komunikazio egiteko. Hauetako lehenengoa, *Sender Agreement* delakoa da, bertan, irteerako kanala definitzen delarik.

**6.12 Irudia:** Dokumentuen bidalketa: *Sender Agreement*

### 6.3.3 Receiver Determination

Konfigurazio honen bidez, aurrekoan definitutako irteera interfazea zein sistemara bidali behar den esaten zaio SAP PIn.

Rule	Condition	Communication Party	Communication Component *
Local Rule			BC_SFTP_MTP

**6.13 Irudia:** Dokumentuen bidalketa: *Receiver Determination*



### 6.3.4 Interface Determination

Hemen, sistema hartzaileak (kasu honetan, METAPOSTAko sFTP zerbitzaria) irteera interfazea zein sarrera interfazerekin jasoko duen adierazten da. Kasu honetan ez da behar izan, baina, normalean, konfigurazio honetan definitzen dira interfazeen arteko mapping-ak.

**Sender**

Communication Party

Communication Component **BS\_OAD100**

Interface **SI\_Envio\_Outb\_Async**

Namespace **http://com.orekait.receiveSFTP**

**Receiver**

Communication Party

Communication Component **BC\_SFTP\_MTP**

Description

Software Component Version of Sender Interface **METAPOSTA 1.0 of orekait**

Maintain Order at Runtime

Operation

**SI\_Envio\_Outb\_Async**

**Receiver Interfaces**

Condition	Ope...	Name *	Namespace *
		SI_Envio_Inb_Async	http://com.orekait.receiveSFTP

**6.14 Irudia:** Dokumentuen bidalketa: *Interface Determination*

### 6.3.5 Receiver Agreement

Konfigurazio honetan sistema hartzailearen Komunikazio Kanala zehazten da. Irudian agertzen den kasuan, sFTP motako adaptadorea baliatzen duen kanala izango da mezuaren hartzailea.

**Sender**

Communication Party

Communication Component **BS\_OAD100**

**Receiver**

Communication Party

Communication Component **BC\_SFTP\_MTP**

Interface **SI\_Envio\_Inb\_Async**

Namespace **http://com.orekait.receiveSFTP**

Description

Receiver Communication Channel \* **CC\_SFTPReceiver\_MTP**

**6.15 Irudia:** Dokumentuen bidalketa: *Receiver Agreement*

## 6.4 Erantzunen prozesamendua

Erantzunen prozesamenduaren prozesuan, jada Diseinuan deskribatu da nolakoa den erabiliko den *Integration Process* lan-fluxua, beraz, hurrengo lerroetan, bertan erabili den ABAP Mapping-a deskribatuko da, hau izan baita prozesu honen berezitasun aipagarriena.

Hurrengo lerroetan, mapeo hau definitzeko jarraitutako pausuak zehaztuko dira.

### 6.4.1 ZMTP\_CL\_ACKS\_SPLIT

Lehendabizi, aurreko kapituluan adierazi den bezala, *IF\_MAPPING* izeneko interfazea inplementatzen duen klase bat sortu da, honek daukan *EXECUTE* metodoa inplementatu delarik.

Metodo honek, erantzunen ZIP fitxategia *source* izeneko Xstring parametro batean jasotzen du eta, emaitza, *result* izeneko XML batean uzten du. Emaitza honek, hainbat mezu (*message* izango ditu, sarrerako inteface batetik lau interface atera behar direlako.

Honela, prozesu honen hasieran, irteerako XML fitxategia sortuko da.

```
*****
* create iXML factory object
DATA: ixmfactory TYPE REF TO if_ixml.
ixmfactory = cl_ixml=>create( ).

* create streamfactory object
DATA: streamfactory TYPE REF TO if_ixml_stream_factory.
streamfactory = ixmfactory->create_stream_factory( ).

* create output document
odocument = ixmfactory->create_document( ).

** Namespace
DATA: messages TYPE REF TO if_ixml_element.
messages = odocument->create_simple_element_ns( name      = 'Messages'
                                                prefix    = 'ns0'
                                                parent    = odocument ).

** Hay que poner este namespace porque es un fichero multiple
DATA: l_namespace TYPE REF TO if_ixml_namespace_decl.
l_namespace = odocument->create_namespace_decl( name      = 'ns0'
                                                prefix    = 'xmlns'
                                                uri       = 'http://sap.com/xi/XI/SplitAndMerge' ).
messages->set_attribute_node_ns( new_attr = l_namespace ).

** Namespace para los nodos creados
l_namespace = odocument->create_namespace_decl( name = 'ns1'
                                                prefix = 'xmlns'
                                                uri   = 'urn:sap-com:document:sap:rfc:functions' ).
```

### 6.16 Irudia: Erantzunen prozesamendua: Irteerako XMLa definitu

Ondoren, ZIP fitxategia irakurriko da *CL\_ABAP\_ZIP* klasea erabiliz, irakurtzen den fitxategi bakoitzeko, *message* XML elementu bat sortuko delarik.

```
* Obtener el fichero
  DATA: lv_ack_xstring TYPE xstring.
  CALL METHOD lcl_zip->get
    EXPORTING
      name                = lv_name
    IMPORTING
      content             = lv_ack_xstring
    EXCEPTIONS
      zip_index_error     = 1
      zip_decompression_error = 2
      OTHERS              = 3.

* Crear el mensaje
  me->add_file( odocument = odocument
               message   = message
               xstring   = lv_ack_xstring
               ack_name  = lv_name ).
```

#### 6.17 Irudia: Erantzunen prozesamendua: ZIP fitxategia irakurri

Hurrengo irudian, *add\_file* metodoaren muina agertzen da, bertan, ACK fitxategi bakoitza irakurri eta ABAP taula batean gordetzen da, taula honetako datuak direlarik XMLan idazten direnak, *create\_simple\_element\_ns* metodoarekin.

```
***** Leer el fichero de entrada y convertirlo a XML *****

  DATA: lt_xml TYPE TABLE OF smum_xmltb,
        lt_result TYPE TABLE OF bapiret2.

  CLEAR lt_xml[].
  CALL FUNCTION 'SMUM_XML_PARSE'
    EXPORTING
      xml_input = lv_ack_xstring
    TABLES
      xml_table = lt_xml
      return    = lt_result.
```

#### 6.18 Irudia: Erantzunen prozesamendua: ACK fitxategiak irakurri

Bukatzeko, irteerako XML fitxategia sortzen da, prozesuan zehar bete den **odocument** fitxategi laguntzailea renderizatuz.

```
***** Se le da salida al XML*****
* create output stream
  ostream = streamfactory->create_ostream_xstring( result ).

* create renderer
  renderer = ixmlfactory->create_renderer( ostream = ostream
                                          document = odocument ).

renderer->render( ).
```

### 6.19 Irudia: Erantzunen prozesamendua: Emaizako XMLa sortu

## 6.4.2 Operation Mapping

Aurreko lerroetan azaldutako ABAP Mapping-a erabiltzeko, SAP PI sistemaren ESRan **Operation Mapping** mapeo bat sortu behar da. Bertan, sarrer eta irteera interfazeak zehazten dira, zein interfaze jaso eta zeintzuk bueltatuko dituen zehazteko. Honetaz gain, ABAP mapping-a implementatzen duen klasearen izena adierazi behar da.

Hurrengo irudian, aurretik definitutako mapeoa erabiltzen duen Operation Mapping-a ikus daiteke, haren sarrera eta irteera interfazeekin.

### 6.20 Irudia: Erantzunen prozesamendua: Operation Mapping

### 6.4.3 Integration Process Transformation

Operation Mapping bat Integration Process lan-fluxu batean erabiltzeko, **Transformation** elementu bat sartu behar, mapeoa eta interfazeak zeintzuk diren zehaztuz.

Name	Value
Step Name	Split
Description	
Operation Mapping	OM_Respuesta
Create New Transaction	<input type="checkbox"/>
▼ <b>Exceptions</b>	
System Error	ErrorMapping
▼ <b>Source Messages</b>	
SI_Respuesta_Abst_Async   http://com.orekait.sendSFTP	source
▼	
SI_ACK_OK_Abst_Async   http://com.orekait.sendSFTP	ack_ok
SI_ACKREC_Abst_Async.SI_ACK_Abst_Async   http://com.orekait.sendSFTP	ackrec
SI_ACKVAL_Abst_Async.SI_ACK_Abst_Async   http://com.orekait.sendSFTP	ackval
SI_ACKDOCS_Abst_Async.SI_ACK_Abst_Async   http://com.orekait.sendSFTP	ackdocs

#### 6.21 Irudia: Erantzunen prozesamendua: Transformation

Aurreko irudian, konektore honen *IP\_ACKS* lan-fluxuan definitutako Transformation-a ikus daiteke. Bertan, hurrengo lerroetan zehaztuko diren funtzioen deiak egiteko beharrezkoak diren datuak lortuko dira, ACK fitxategiak irakurrita.

### 6.4.4 Funtzio Moduluak

Erantzunen prozesamendua bukatzeko, jasotako emaitzak SAP sisteman gordeko dira. Horretarako, SAPeko *Function Group* bat sortu da, hurrengo lerroetan definituko diren **Function Module** izeneko funtzioez osatuta.

Lau funtzio modulu sortu dira, suerta daitekeen egoera bakoitzeko bat.

ZMTP_FG_ACKS	Recepción de ACKS de Metaposta
ZMTP_FM_ACKDOCS	Comprobar respuesta en ACKDOCS
ZMTP_FM_ACKREC	Comprobar respuesta en ACKREC
ZMTP_FM_ACKVAL	Comprobar respuesta en ACKVAL
ZMTP_FM_ACK_OK	Envío realizado correctamente

#### 6.22 Irudia: Erantzunen prozesamendua: Function Modules

Function Module hauek PI sistematik deituak izango dira RFC deien bitartez, horretarako sortu diren lau interface eta hauen konfigurazioak erabiliz. Dei hauek, Integration Process lan-fluxuaren barrutik egingo dira, honela, erantzunak jaso bezain laster, hauek irakurri eta emaitzen arabera, funtzio bat edo beste deituko da.

```

FUNCTION zmtplib_fm_ackrec.
  *-----
  *""Interfase local
  *  IMPORTING
  *    VALUE(SEND_ID) TYPE  ZMTP_DE_IDENTIFICADOR
  *    VALUE(SEND_RECEPTIONSTATUS) TYPE  ZMTP_DE_RECEPTIONSTATUS
  *    VALUE(SEND_PACKAGEERROR) TYPE  ZMTP_ST_PACKAGEERROR OPTIONAL
  *    VALUE(SEND_ERRORCODES) TYPE  ZMTP_TT_ERRORCODES OPTIONAL
  *    VALUE(SEND_SIGNATURE) TYPE  ZMTP_DE_SIGNATURE OPTIONAL
  *    VALUE(SEND_NOTIFICATION) TYPE  ZMTP_TT_NOTIFICATION OPTIONAL
  *-----

  DATA: lv_ack TYPE string.
  CONCATENATE send_id '_ACKREC.XML' INTO lv_ack.

  * Inicializar el log
  PERFORM crear_log.

  * Mensaje de ACK recibido
  PERFORM mensaje USING 4 'I' 030 lv_ack.

  * Añadir mensaje de error al log
  DATA: lv_error TYPE string.
  lv_error = 'El fichero no se recibió correctamente en Metaposta'.
  PERFORM mensaje USING 4 'E' 031 lv_error.

  * Actualizar la BD
  UPDATE zmtplib_t501 SET ackrec = 'E' ackval = 'P' ackdocs = 'P'
    WHERE identificador = send_id.

  * Guardar el log
  PERFORM guardar_log.

```

### 6.23 Irudia: Erantzunen prozesamendua: Funtzio modulu bat

Irudian ikusten den bezala, funtzio hauek, Diseinuan definitutako datuak jasoko dituzte, eta ZMTP\_T501 taula eguneratuko dute, LOGean ere idatziko dutelarik.

## 7. KAPITULUA

---

### Probak

---

Kapitulu honetan, METAPOSTAk definitutako proba plana zehazten da. Proba plan honetan, Konektorearen funtzionamendu egokia egiaztatu egin da, horretarako, atal honetan deskribatuko diren proba guztietan emaitza positiboak lortu egin dira. Hurrengo lerroetan zehaztuko diren guztiek, beraz, Konektorea onargarriztat ematen dute eta Sender batean ezar daitekeela zihurtatzen dute.

Proba bakoitzeko, hauen osotasuna bermatzeko, emaitza tekniko zein funtzionalak egiaztatu dira, modu honetan, Konektorearen funtzionamendua egokia dela zihurtatuz.

Proba guzti hauek hasi baino lehen, honako baldintzak betetzen direla egiaztatu da:

- Proba unitarioak egin dira METAPOSTAtik
- Sender-a (kasu honetan, Konektorea) METAPOSTAko sistemetan sortuta dago
- Proba guztiak egiteko erabiltzaileak sortu dira

Egindako proba guztiak bi multzo desberdinetan banatzen dira, alde batetik, **Web Service**-a testeatzeko egindakoak eta, bestetik, **dokumentuen bidalketarekin** lotutako guztiak testeatzekoak. Hurrengo lerroetan, taula batzuen bidez, egindako proba guztiak zerrendatuko dira.

## 7.1 Web Service-aren probak

Hurrengo tauletan deskribatuko diren proben bitartez, Web Service-a modu egokian funtzionatzen duela egiaztatuko da. Horretarako, eta Konektorea konfigurazio desberdinetako Sender-etan ezar daitekeela kontuan harturik, Sender-ek izan ditzaketan konfigurazio desberdin horiek guztiak probatuko dira.

### 7.1.1 Sender-a datu laguntzaile eta email eskaerarik gabe

Sender-ak ez du informazio gehigarririk behar erabiltzaileak identifikatzeko, eta, jasoko dituen datuak, beraz, dokumentu-mota, dokumentu-zenbakia eta jaiotze-data izango dira.

Proba	Deskribapena	Emaitza
1	Existitzen ez den erabiltzaile batekin <i>SENDERPeticionTipoUsuario</i> metodoa deitu.	Errore bat bueltatzen da eta erabiltzaileari abisua ematen zaio.
2	Existitzen den baina jaiotze-data desberdinak dituen erabiltzaile batekin suskripzio eskaera egin.	Errore bat bueltatzen da eta erabiltzaileari abisua ematen zaio.
3	Existitzen den erabiltzaile batekin lehenengo suskripzio eskaera egin.	Erabiltzailea Sender-ean harpidetuta geratzen da.
4	Existitzen den erabiltzaile baten hurrengo suskripzio eskaerak.	Sender-ak errore bat bueltatzen du eta METAPOSTak erabiltzailea suskribatuta bezala markatzen du.
5	Existitzen den erabiltzaile batekin lehenengo baja eskaera egin.	Erabiltzailearen harpidetza ezabatzen da.
6	Data desberdinak dituen erabiltzaile baten baja eskaera.	Harpidetza mantentzen da eta erabiltzaileari abisua ematen zaio.
7	Existitzen ez den erabiltzaile baten baja eskaera.	METAPOSTak erabiltzailearen harpidetza ezabatzen du.
8	Harpidetuta gabeko erabiltzaile baten baja eskaera.	Erabiltzailearen harpidetza METAPOSTatik ezabatzen da.

**7.1 Taula:** Probak: Datu laguntzaile eta email eskaerarik gabe



### 7.1.2 Sender-a email eskaerarekin

Hurrengo probetan, Sender-ak harpidetzak egiteko posta elektronikoko helbide bat behar izteko konfiguratu da. Horretarako, Web Service-ak *repcionCorreoUsuario='S'* aldagaiarekin erantzungo du *SENDERPeticionTipoUsuario* deiei.

Funtzionamendua aurreko ataleko antzekoa izango da kasu gehienetan, METAPOSTAk emaila bidaltzen ez duen kasuetan izan ezik.

Proba	Deskribapena	Emaitza
1	Harpidetza eskaera emailik eman gabe.	Errore bat eman eta emaila eskatzen zaio erabiltzaileari.
2	Existitzen ez den erabiltzaile baten suskripzio eskaera.	Errore bat bueltatzen da eta erabiltzaileari abisua ematen zaio.
3	Existitzen den baina jaiotze-data desberdinak dituen erabiltzaile baten suskripzio eskaera.	Errore bat bueltatzen da eta erabiltzaileari abisua ematen zaio.
4	Existitzen den erabiltzaile baten lehenengo suskripzio eskaera.	Erabiltzailea harpidetuta geratzen da Sender-ean.
5	Existitzen den erabiltzaile baten hurrengo suskripzio eskaerak.	Sender-ak errore bat bueltatzen du eta METAPOSTAk erabiltzailea suskribatuta bezala markatzen du.
6	Existitzen den erabiltzaile batekin lehenengo baja eskaera egin.	Erabiltzailearen harpidetza ezabatzen da.
7	Jaiotze-data desberdinak dituen erabiltzaile baten baja eskaera.	Harpidetza mantentzen da eta erabiltzaileari abisua ematen zaio.
8	Existitzen ez den erabiltzaile baten baja eskaera.	METAPOSTAk erabiltzailearen harpidetza ezabatzen du.
9	Sender-ean harpidetu gabeko erabiltzaile baten baja eskaera.	METAPOSTAk erabiltzailearen harpidetza ezabatzen du.

**7.2 Taula:** Probak: Email eskaerarekin

### 7.1.3 Sender-a datu laguntzaileekin

Sender-ak identifikazio datu estandarretaz gain, beste batzuk behar ditu erabiltzaileak identifikatzeko. Horretarako, METAPOSTAk, suskripzioa egiteko garaian, datu laguntzaile bat edo bi bidaltzen ditu erabiltzaileen identifikazioa gauzatu ahal izateko.

Sender-a, beraz, jasotzen dituen datu hauek bere sistemako zeintzuekin egiaztatu behar dituen konfiguraturata izan behar du.

Proba	Deskribapena	Emaitza
1	Harpidetza eskaera datu laguntzaile-lerik eman gabe.	Errore bat eman eta datuak eskatzen zaizkio erabiltzaileari.
2	Existitzen ez den erabiltzaile baten suskripzio eskaera.	Errore bat bueltatzen da eta erabiltzaileari abisua ematen zaio.
3	Existitzen den erabiltzaile baten lehenengo suskripzio eskaera.	Erabiltzailea harpidetuta geratzen da Sender-ean.
4	Existitzen den erabiltzaile baten hurrengo suskripzio eskaerak.	Sender-ak errore bat bueltatzen du eta METAPOSTAk erabiltzailea suskribatuta bezala markatzen du.
5	Existitzen den erabiltzaile batekin lehenengo baja eskaera egin.	Erabiltzailearen harpidetza ezabatzen da.
6	Existitzen ez den erabiltzaile baten baja eskaera.	METAPOSTAk erabiltzailearen harpidetza ezabatzen du.
7	Sender-ean harpidetu gabeko erabiltzaile baten baja eskaera.	METAPOSTAk erabiltzailearen harpidetza ezabatzen du.

**7.3 Taula:** Probak: Datu laguntzaileekin

## 7.2 Dokumentuen bidalketaren probak

Atal honetan, fitxategien bidalketak modu egokian egiten direla zihurtatzeko egindako probak zerrendatzen dira.

Proba	Deskribapena	Emaitza
1	Dokumentu egokiez bakarrik osatutako pakete baten bidalketa.	Dokumentu guztiak modu egokian banatzen dira.
2	Dokumentu egokiez eta okerrekin osatutako pakete baten bidalketa.	Dokumentu egoki guztiak modu egokian banatzen dira, okerren berri ematen delarik erantzunen paketeetan.
3	Pakete baliogabe baten bidalketa.	Ez da dokumenturik banatzen METAPOSTAn eta ACKVAL fitxategiak erroreak ditu.
4	Onartutakoak baino errore gehiagoko paketea.	Ez da dokumenturik banatzen METAPOSTAn eta ACKDOCS fitxategiak erroreak ditu.
5	Publikazio data atzeratutako pakete baten bidalketa.	Dokumentuak publikazio data ailegatzen denean banatzen dira.
6	Publikatu gabeko pakete baten zuzenketa paketearen bidalketa.	Zuzenketa paketeko dokumentuak bakarrik banatzen dira.
7	Publikatutako pakete baten zuzenketa paketearen bidalketa.	Ez da zuzenketa paketeko dokumenturik banatzen eta ACKVAL fitxategiak erroreak ditu.
8	Errorerik gabeko publizitadedun paketearen bidalketa.	Dokumentu eta publizitateak modu egokian banatzen dira.
9	Erroreak dituen publizitadedun paketearen bidalketa.	Erroredun publizitatea duten dokumentuak ez dira banatuko eta ACK fitxategiek erroreak dituzte.

**7.4 Taula:** Probak: Dokumentuen bidalketak



## 8. KAPITULUA

---

### Jarraipena eta Kontrola

---

Proiektuen kudeaketaren helburu nagusia proiektu bakoitzak dituen helburu berekiak lortzea da, proiektuaren murriztapenak kontuan hartuta. Kudeaketa, beraz, edozein proiektuaren atalik garrantzitsuenetakoa da, ondo eginten bada, proiektuaren arrakasta bermatzen baitu.

Kudeaketa egokia izateko, plangintza sendo eta zehatz bat izan behar du abiapuntu bezala, horren jarraipen eta kontrol sakonak egiten direlarik ondoren. Kudeaketa prozesuan zehar, planifikazioan definitutako guztiak jarraitu behar dira, horiek egokiak badira, gertatzen den guztiari erantzun egokia ere emango zaiolarik.

Aurretik aipatutakoak kontuan hartuta, eta horren garrantziaren jakitun, proiektu honetan ere Kudeaketari indar handia eman zaio, *Proiektuaren Helburu Dokumentuan* adierazitako guztia jarraitzen saiatu delarik.

Kapitulu honetan, beraz, proiektuaren kudeaketan zehar jasotako emaitzak biltzen dira, non, hasieran planifikatu zena eta, gero, errealitatean egin denaren arteko desberdintasunak ikusiko diren.

Hau guztia azaltzeko, hurrengo lerroetan, honako atal hauek deskribatuko dira: **Denboraren jarraipena**, **Komunikazioen jarraipena**, **Kalitatearen jarraipena** eta **Arriskuen jarraipena**.

## 8.1 Denboraren jarraipena

Atal honetan, proiektuaren atazen esfortzuen analisia egingo da, hau da, ataza bakoitzari eskaini zaion denbora aztertuko da. Horretarako, estmitatutako denborak eta denbora errealak alderatuko dira. Honekin batera, proiektuan zehar gertatutakoa azalduko da, arazoak eta hauen soluzioak zehaztuz eta hartutako erabakiak adieraziz.

### 8.1.1 Burututako atazak

Hurrengo taulan, proiektu hau osatzen duten ataza guztien denborak agertzen dira, hauen konparaketa egin ahal izateko, hasiera eta amaiera data estimatu eta errealak agertzen direlarik.

<b>ATAZA</b>	<b>HASIERA</b>	<b>AMAIERA</b>	<b>HASIERA</b>	<b>AMAIERA</b>
<b>Formakuntza</b>	<b>2013-12-02</b>	<b>2013-12-20</b>	<b>2013-12-02</b>	<b>2013-12-20</b>
<b>Diseinua</b>	<b>2013-12-02</b>	<b>2013-12-20</b>	<b>2013-12-02</b>	<b>2013-12-20</b>
Web Zerbitzua	2013-12-02	2013-12-13	2013-12-02	2013-12-13
Dokumentuen sorketa	2013-12-09	2013-12-20	2013-12-02	2013-12-20
Dokumentuen bidalketa	2013-12-09	2013-12-20	2013-12-02	2013-12-20
<b>Garapen ingurunea prestatu</b>	<b>2013-01-07</b>	<b>2014-01-24</b>	<b>2013-01-07</b>	<b>2014-01-31</b>
Datu basea	2013-01-07	2014-01-24	2013-01-07	2014-01-31
sFTP ingurunea	2013-01-07	2014-01-24	2013-01-07	2014-01-31
<b>Konektorea garatu</b>	<b>2014-01-07</b>	<b>2014-02-21</b>	<b>2014-01-07</b>	<b>2014-03-07</b>
Inplementazioa	2014-01-07	2014-02-14	2014-01-07	2014-02-14
Probak	2014-01-13	2014-02-14	2014-01-13	2014-02-21
Arriskueterako denbora	2014-02-17	2014-02-20	2014-02-17	2014-02-21
Dokumentazioa	- -	- -	2014-02-24	2014-03-07
<b>Kudeaketa</b>	<b>2013-12-02</b>	<b>2014-02-21</b>	<b>2013-12-02</b>	<b>2014-02-28</b>
Irismena	2013-12-02	2013-12-13	2013-12-02	2013-12-13
Plangintza	2013-12-02	2013-12-13	2013-12-02	2013-12-13
Jarraipena eta kontrola	2013-12-16	2014-02-21	2013-12-16	2014-02-28
Bilerak	2013-12-02	2014-02-21	2013-12-02	2014-02-28
Ixtea	2014-02-17	2014-02-21	2014-02-24	2014-02-28
<b>TOTALA</b>	<b>2013-12-02</b>	<b>2014-02-21</b>	<b>2013-12-02</b>	<b>2014-03-07</b>

**8.1 Taula:** Atazen esfortzu errealak

Irudian ikusten den moduan, egunak erabili dira data-unitate bezala, orduak erabili beharrean. Hau, ordu bakarra, proiektuaren osotasunean, kontuan hartzeko denbora gutxiegi dela iritzi delako egin da. Gainera, *Proiektuaren Helburu Dokumentuan* azaldu bezala, egun bakoitzeko lana 7 ordukoa izan dela kontuan hartuz, egun guztietako lan karga berdina izan da.

Hurrengo lerroetan, taulan agertzen diren atazak azalduko dira, bakoitzean agertutako arazoak edota aldaketak deskribatuz. Gogoratu behar da, hori bai, Plangintzan adierazita dagoen moduan, 2013-12-23 eta 2014-01-07 data bitartean ez zela lanik egin, egun horietan ikasleak oporrak hartu zituelarik.

**Formakuntza** ataza izan da aldaketarik jasan ez duen bakarra, hasieratik definitu baitzen horrek izango zuen denbora zehatza, hortik aurrera ikasitako guztia modu sekundario batean egin delarik. Honela, proiektuaren lehenengo asteetan formakuntzari lan-karga handia dedikatu zitzaion, ahalik eta gauza gehien ikasi, eta hurrengo atazak modu errazagoan egin ahal izateko.

**Diseinua** ataza, proiektuarekin batera hasi zen, ondorengo guztiak errazteko asmoz, ahalik eta zehatzena izan zedin. Honela, bere iraupena luzatu den arren, lan gehigarri honek, etekinak ekarri ditu hurrengo atazen lan kargak gutxituz. Gainera, METAPOSTatik jasotako dokumentazio zehatza baliatuta, Konektorearen elementuen diseinua errazagoa izan da, oso garrantzitsuak izan direlarik ere Oreka IT enpresatik jasotako jarraibideak.

**Garapen ingurunea prestatu** atazari dagokionez, aste batez luzatu egin da, batez ere, SAP PI sistemarekin erlazionatutako konfigurazioak direla eta, hauek, inplementazioaren azken faseetan ere aldatu behar izan baitira. Hala ere, espero zen moduan, ataza honen lan-karga gutxituz joan da denborarekin, azkeneko asteetan, ingurunearen mantentze azkarrak bakarrik egin direlarik.

**Konektorearen Garapena** izan da aldaketa gehien jaso dituen ataza, bere iraupena 3 astez luzatu baita, hurrengo lerroetan azalduko diren arrazoiengatik.

Konektorearen funtzionalitateen inplementazioa, esperotako denboran egin da, estimatutako epean bukatu delarik. Probak, ordea, aste batez luzatu dira, inplementazio osoa bukatutakoan, METAPOSTAk, hainbat proba gehigarri egitea erabaki zuelako, Konektorearen funtzionamendua modu zehatzagoan egiaztatzeko. Proba hauek, hala ere, kontingentzia moduan utzitako tartearen barruan egin dira, ataza hauen bukaera parekatu delarik.

Probak egin eta hauen emaitzak jaso bitartean, eta, *Arriskueta* denboraren barruan, harpidetutako erabiltzaileak (langileak eta bezeroak) ikusteko bi programa egin dira. Honela, aurretik definitutako zabalkuntza bat egiteko ere erabili da denbora hau.

Proiektuaren iraupena luzatu duena, eta planifikazio garaian erabakitako lan metodologia aldatu duena, *Dokumentazioa* izan da. Hasiera batean, ideia, dokumentazioa proiektuaren garapena egin bitartean egitea izan zen. Baina, memoria hau L<sup>A</sup>T<sub>E</sub>X bitartez egitea erabaki ondoren, dokumentazioa bukaeran egitea erabaki zen. Honela, egunero egindakoa apuntatu da, dokumentazio guztia bukaeran egin delarik, horretarako ataza berri bat sortuz.

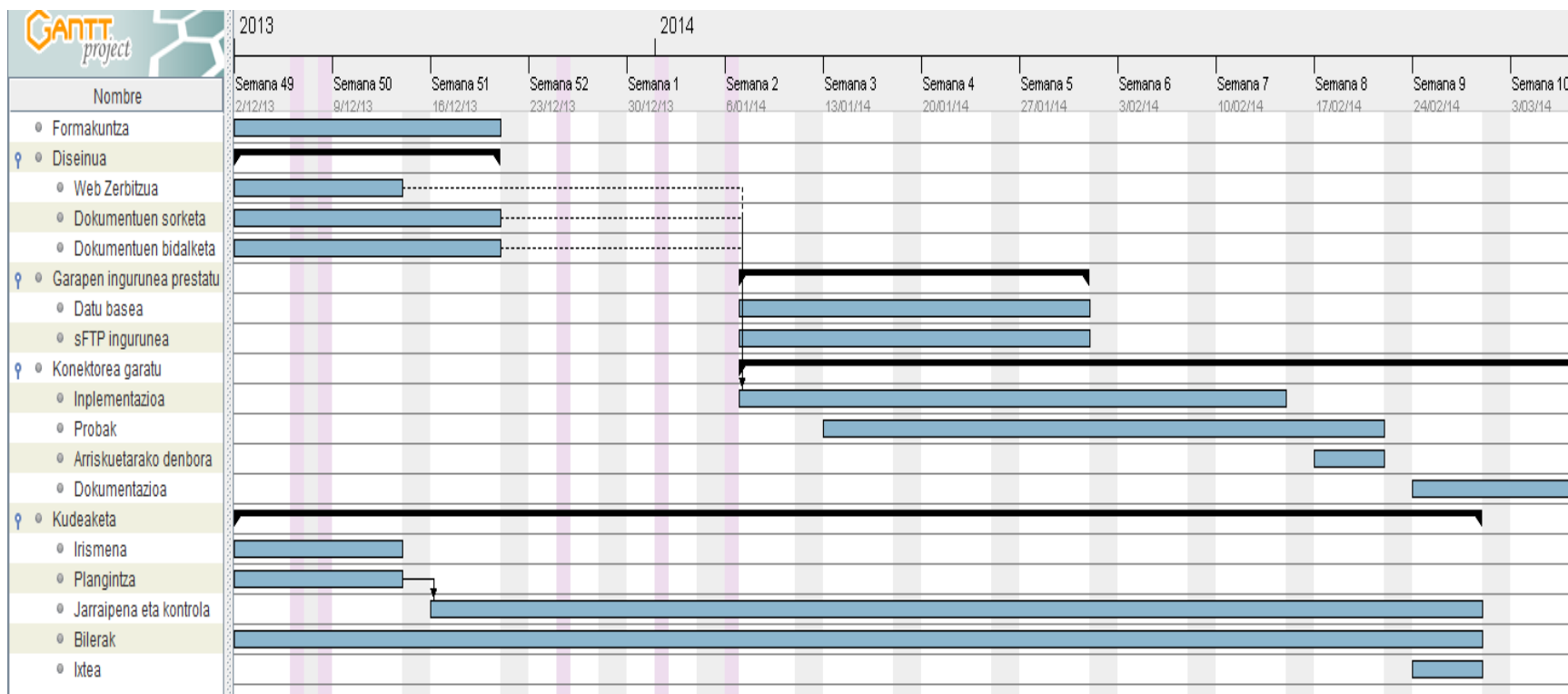
Honekin erlazionatuta dago ere *Aurkezpena* atazaren desagertzea, sortutako ataza berri biltzeko baita lan hau ere.

**Kudeaketa** proiektu osoan zehar egiten denez, eta aipatutako aldaketaren ondorioz, bi astez luzatu egin da. Honela, proiektuaren bukaeraz egin behar diren atazak, noski, luzatu egin dira, hasierakoak, estimatu bezala egin direlarik.



## 8.1.2 Kronograma

Aurreko lerroetan, proiektua osatu duten atazen hasiera eta amaiera datak azaldu dira. Orain, ataza horiek denboran kokatuko dira, ataza bakoitza zein egunen artean egin den adierazteko. Hurrengo irudia, beraz, proiektu honen Kronograma erreala da:



8.1 Irudia: Kronograma erreala

## 8.2 Komunikazioen jarraipena

Komunikazio planean definitu bezala, garatzaileak hiru komunikazio desberdin izan ditu proiektu honetan: Oreka IT enpresako arduradunarekin, METAPOSTAren eta unibertsitateko tutorearekin.

Atal honetan, proiektuan zehar jarraitutako hiru protokolo hauek aztertuko dira, nolakoak izan diren zehaztuz, eta gertatutakoak deskribatuz.

### 8.2.1 Garatzaile eta enpresaren arteko komunikazioak

Komunikazio hau izan da hiruretan landuena, proiektu hau enpresa batean egin denez, garatzaileak enpresako bere gainbegiralearekin komunikazio zuzena behar izan duelako, oso gertuko jarraipena mantenduz.

Gauzak honela, eta garatzailea enpresan bertan lan egin duela baliatuz, komunikazio gehiena presentziala izan da, kontaktu zuzenak, komunikazioa hobetu duelarik. Hone-la, proiektuaren garapenaren inguruan zerbait eztabaidatu behar izan denean, momentuan egin da, denborarik galdu gabe. Komunikazio hau, laburbilduz, oso positiboa eta arrakastatsua izan da, neurri handian, Oreka IT enpresak jarri dituen erraztasun guztiei esker.

### 8.2.2 Garatzaile eta METAPOSTAren arteko komunikazioak

Komunikazio hau modu egokian egin da proiektuaren hasieratik bukaerara. Momentuoro egon da posta elektronikoko bidezko komunikazioa, Konektorearen inguruko aldatzea, hobekuntza, argibideak, eta abarrak eztabaidatu ahal izateko. Bi noranzkoetan izan da komunikazioa egokia, garatzaileak zein METAPOSTAk erantzun azkar eta egokiak eman dituztelarik behar izan denean.

### 8.2.3 Garatzaile eta tutorearen arteko komunikazioak

Unibertsitateko tutorearekin egondako komunikazioak, planifikatu bezala joan dira, proiektuaren nondik norakoen berri eman zaio, zalantzak eta arazoak posta elektronikoko bidez adierazi zaizkiolarik. Oro har, komunikazio hau egokia izan da, ez da gertuko jarraipena izan, baina behar izan denean posta elektronikoko bidez komunikatu da.

## 8.3 Kalitatearen jarraipena

Hurrengo lerroetan, Proiektu eta Produktuaren kalitatearen jarraipena aztertuko da.

### 8.3.1 Proiektuaren kalitatea

Proiektuaren kalitatea, *Kalitate planean* agertzen den moduan, plangintzan zehaztutakoak jarraituz neurtzen da. Honela, zenbat eta bertan definitutako alderdi gehiago errespetatu, orduan eta kalitate hobea lortuko da proiektuan.

Honela, aurreko ataletan azaldu direnak kontuan hartuz, ezin da esan proiektuaren beraren kalitatea oso altua izan denik, epeak guztiz bete ez direlako, eta, komunikazioa, batez ere tutorearerin, hobea izan zitekeelako. Hala eta guztiz ere, proiektua gehiegizko arazorik gabe aurrera atera da, Plangintzan zehaztutako guztiak modu egokian betez, beraz, proiektuaren kalitate orokorra ona izan dela zihurtatu daiteke.

### 8.3.2 Produktuaren kalitatea

Produktuaren *Kalitate plana* zehaztean, honen kalitatearen bi neurgailu identifikatu ziren. Alde batetik, garatutako guztiaren funtzionamendua egokia dela zihurtatzea eta, beste aldetik, garatutako guztia produktuaren hartzailleak diren Oreka IT eta METAPOSTA enpresen nahietara egokitzea.

Produktuaren kalitatea aztertzeko, beraz, definitutako neurgailu bi hauen emaitzak aztertu behar dira:

- Aurreko atalean azter daitekeen moduan, proba plan sendo baten bidez zihurtatu da produktuaren funtzionamendua egokia dela, Irismenean definitutako *Irismen minimo* guztiak betetzen dituelarik.
- Komunikazioaren jarraipenean azaltzen den moduan, produktuaren hartzailleekiko komunikazioa oso ona izan da, produktuaren beraren inguruko iritzia momentuoro jaso delarik. Honela, eguneroko onarpena izanda, eta *Gogobetetzea bermatzeko argibideak* jarraituz, Oreka IT eta METAPOSTA enpresetatik jasotako *feedback*-a oso ona izan dela zihurtatu daiteke.

Aurrekoak kontuan hartuta, beraz, produktuaren kalitatea oso altua izan dela ondoriozta daiteke.

## 8.4 Arriskuen jarraipena

*Arriskuen planean*, proiektu honetan gerta zitezkeen arrisku guztiak daude zerrendatuta, gertatzeko zuten probabilitate, eragin eta larritasun estimazioekin. Bertatik, hiru arrisku izan dira proiektu honetan gertatu direnak, hurrengo lerroetan gertatutakoa eta hartutako neurriak deskribatuko direlarik. Orokorrean, hala ere, Arrisku eta Aldaketa planetan zehaztutakoak jarraitu dira, eta, arrisku hauen eraginak baxuak izan dira.

### Denbora estimazio okerrak

Denboraren jarraipenean azaldu den moduan, estimazio batzuk okerrak izan diren arren, ez dute, orokorrean, eragin handirik izan, eragin positibo eta negatiboak konpentsatu direlarik. Hala ere, *Dokumentazioaren* inguruan hartutako erabakiak, hasieran zehaztuako epeak ez betetzea ekarri du, honen eragina bai altuagoa izan delarik.

Denbora estimazio oker hauek ez dira, dena den, oso larriak izan, produktua epean bukatu delako, atzerapenak, proiektuaren bukaera epean baino eragin dutelarik.

### Lan ingurune ezezaguna

Aurreikusi bezala, garatzaileak, ezagutzen ez zituen hainbat baliabide erabili behar izan ditu. Hala ere, arrisku honek, aurretik hartutako prestakuntza eta adituengandik jasotako laguntzari esker, ez du eragin larririk izan.

Modu pertsonalean arrisku honek ondorio positiboak ere izan ditu garatzailearentzat, proiektuan eragin negatiborik izan gabe, ezagutzen ez zituen hainbat SAPen baliabide desberdin ikasi dituelarik.

### Bezeroen eskakizunen aldaketa

Enpresa batean egindako proiektua izanda, beste enpresa batekin elkarlanean, arrisku hau gertatuko zela estimatu zen, horrela izan delarik. Arrisku honek izan dituen ondorioak, ordea, ez dira estimatu bezain larriak izan eta agertu diren aldaketek ez dute proiektuan arazo handirik suposatu.

Gertatutako aldaketa gehienak txikiak izan dira eta ez dira proiektuaren irismen minimoaren inguruan izan, beraz, lan karga pixka bat baino ez diote gehitu proiektuari.

## 9. KAPITULUA

---

### Ondorioak eta etorkizunerako lerroak

---

Azken kapitulu honetan, Karrera Amaierako Proiektu honi bukaera emateko, honen garapenean zehar gertatutako guztiak aztertu eta ondorioak adieraziko dira. Horretarako, proiektuan zehar bildutako ondorioekin batera, ikuspegi globala aztertuta lortutako ondorio berriak ere deskribatuko dira.

Ondorioetaz gain, proiektuaren etorkizunaren inguruko hainbat ausnarketa ere adieraziko dira kapitulu honetan. Aurretik zenbaitetan aipatu den moduan, proiektu honen helburua den **SAP - METAPOSTA konektorea** ez baita Karrera Bukaerako Proiektu honekin bukatzen.

#### 9.1 Ondorioak

Hurrengo lerroetan azalduko diren ondorio guztiak, bi multzo desberdinetan bananduko dira. Alde batetik, proiektuarekin berarekin erlazionatutako ondorioak deskribatuko dira, honen irismen, kudeaketa eta garapenarekin erlazionatutakoak, batez ere. Bestetik, ondorio pertsonalak bilduko dira, ikasleak, modu pertsonalean, denbora guzti honetan jasotakoak. Azken hauek, beraz, ikaslearen Ikasitako Lezioak bezala ere interpreta daitezke.

### 9.1.1 Proiektuaren inguruko ondorioak

Proiektu honek, SAP ERP sistema eta METAPOSTA interneteko posta zerbitzua konektatzen dituen **Konektore** bat egitea izan du helburu. Ez da aplikazio izugarri konplexua izan, baina, sistema desberdinekin lan egin eta hauek elkarrekin lan egitea lortu da, hau erraza izan ez delarik. Gainera, enpresa batean egindako proiektua izan da, honek izaten dituen abantaila eta desabantaila guztiak.

Aztertu behar den lehendabizikoa, proiektuaren beraren helburua da, bere **Irismena**, alegia. Honela, eta txosten honetan jada adierazi diren kontuan hartuz, hau da ondoriozta daitekeena: proiektuak bere helburua modu egoki batean lortu du, egin-beharreko guztiak egin eta aurretik ezarritako baldintza guztiak betetzen dituelarik.

Irismenaren inguruan ere, oso garrantzitsua izan da proiektu osoan zehar Konektorearen hartzaileak diren bi enpresen gogobetetzeak jarraitzea, hau emaitza egokiak lortzeko ezinbestekoa baita modu honetako proiektu batean.

Proiektuaren **Kudeaketa** funtsezko alderdi bat dela argi izanda, garrantzi handia eman zaio hasieratik amaierararte. Hala eta guztiz ere, eta aurreko atalean azalduta dagoen bezala, plangintza ez zen guztiz egokia izan, zenbait aldaketa egin behar izan direlarik proiektuan zehar. Dena den, bertan zehaztutakoaren gehiengoa modu egokian egin eta jarraitu da, gertatutako arazoak, larriak izan ez direlarik. Egondako arazo hauek, aurretik azaldu den moduan, ez dira larriak izan eta tamaina honetako proiektu batean espero daitekeenaren barruan daude, gerta daitezkeen gauza onargarrien artean. Honela, beraz, plangintza on batetik abiatuta, jarraipen egoki bat egin dela ondoriozta daiteke.

**Garapenaren** inguruan, garrantzitsua izan da analisi eta diseinu landuak izatea hasieratik, egin-beharrekoa ondo definitzen bada, lana askoz errazago egiten delako. Honela, proiektu honen garapenaren gakoetako bat, hasieratik erabakitako lan metodologia izan da, oso onuragarria izan baita ikaslea Oreka enpresaren barruan lan egitea Metapostako langileekin konexio zuzena mantenduz. Honi esker, garapenean zehar agertutako zalantza zein arazo guztiei erantzun azkar bat eman zaiolarik.

Proiektu hau, laburbilduz, bere helburua zen **Konektorea** egitea lortu du, arazo handiegirik izan gabe, eta modu antolatu batean, momentu guztietan egungo egoera eta egin beharreko hurrengoa argi egon delarik.

### 9.1.2 Ondorio pertsonalak

Proiektu honek, hasieratik, karreraren zehar ikasitakoak praktikan jartzea izan du helburu nagusietako bat, Karrera Amaierako Proiektuak, azken finean, horretarako egiten baitira. Gainera, enpresa batean eta proiektu errealean lan egiteko aukera ere izan dut, horrek balio erantsi handia duelarik.

Honela, egindako lan guztia esperientzia pertsonal zein profesional positibo izugarria izan da. Alde batetik, pertsona gehiagorekin lan eginda, hauek ezagutu eta haiengandik ikasteko aukera eman didalako. Bestetik, enpresa batean egonda, profesional bezala hazten lagundu didalako, unibertsitatekanpo daudenak hobeto ulertzeko aukera emanez. Hori guztiaz gain, oso eskertzekoa izan da **Oreka IT** enpresak niregan jarri duen konfiantza, ezinbestekoa izan baita proiektu honetan zehar.

Modu pertsonalean ere, karreraren zehar egindakoak baino askoz konplexuagoa izan den proiektu hau aurrera ateratzea erronka handia izan da niretzat, horretarako gai naizela erakutsi dudalarik.

Alderdi pertsonal hauetaz gain, alderdi tekniko asko ikasteko ere balio izan dit proiektu honek, guzti honekin hasi baino lehen ezagutzen ez nuen SAP sistemaren inguruan gauza asko ikasi baititut. Orain, SAP ERPa zer den badakit, eta horren erreminta desberdinekin lan egiteko gai ikusten dut nire burua. Honela, ABAP programazio lengoaiara, SAPen funtsezkoak diren *Nomina* eta *Finantza* atalak, eta, batez ere, *SAP PI* integrazio sistema indartsua ezagutu ditut.

Proiektu hau, nire Ingenieritza Informatikoko ikasketei bukaera ematen die, eta oso modu onean eman gainera, bide berri bat irekiz, aurrekoa amaitzen den tokian.

## 9.2 Etorkizunerako lerroak

Hasieratik azaldu den moduan, proiektu honek hasiera ematen dio bi enpresen arteko elkarlanari, horretarako baliabide berri bat sortuz. Orain, beraz, garatutako Konektore hau erabili nahi duten enpresak bilatu behar dira. Honela, etorkizunean egin beharreko lehen-dabiziko gauza, **SAP - METAPOSTA konektorea** erabiliko duten enpresak aurkitzea izango da.

Konektorea eraabili nahi duen enpresa bat aurkitzen denean, bertan ezarri beharko da, horretarako, bertara egokitu beharko delarik Konektorea. Adibidez, dokumentuen sorketa bertan erabiltzen duten nomina/faktura berezietara egokitzeko, bidalketak egiten dituen SAP PI konfigurazioa bertakoari gehitzeko, dokumentu mota berriak bidaltzeko, ...

Argi dago, beraz, Konektore honek eboluzio handiak jasango dituela hemendik aurrera, eta, beraz, etorkizunean berarekin egingo dena, erabiltzen dutenen araberakoa izango da, beharretara moldatu beharko delarik.

Hala ere, eta Konektorearen erabilerak berak ekarriko dituen aldaketak alde batera utziz, hurrengo lerroetan, Konektoreari berari gehitu ahal zaizkion zenbait ezaugarri zerrendatzen dira, hauek izan daitezkeelarik gehitu ahal zaizkion hobekuntzak:

- Kodearen parametrizazioa: Aurretik aipatu da jada, Konektorea parametrizagarria izan behar duela. Eta, ahal izan denean, horrela egin da, baina, honetan, asko dago oraindik hobetzeko.
- EDITRANen erabilera: Konektoreak sFTP teknologia baliatzen du bidalketak egiteko, baina, METAPOSTAk EDITRAN teknologia ere onartzen du bidalketarako, hau izan daitekeelarik egin beharreko beste gauza bat.
- SAP PI kentzea: Konektoreak SAP PI sistema bat behar du bidalketak egiteko, eta, hau, bidalketak ABAP teknologiaz egitea baino hobea den arren, sinpleagoa den beste modua ere bideragarria izan daiteke.
- Publizitate automatikoa: METAPOSTAk publizitatea bidaltzeko aukera ematen du dokumentuen bidalketarekin batera, baina Konektoreak funtzionalitate hau ez du automatikoki egiten, eskuz egin behar delarik orain.



---

## Bibliografia

---

- [1] *Oreka IT Bloga*  
<http://orekait.com/blog/>
- [2] *Blog de SAP*  
<http://www.blogdesap.com/>
- [3] *SAP PI for Beginners*  
<https://scn.sap.com/docs/DOC-41766>
- [4] *SAP Community Network Wiki*  
<http://wiki.scn.sap.com>
- [5] *SAP Community Network*  
<http://scn.sap.com>
- [6] *SAP Help Portal*  
<http://help.sap.com/>



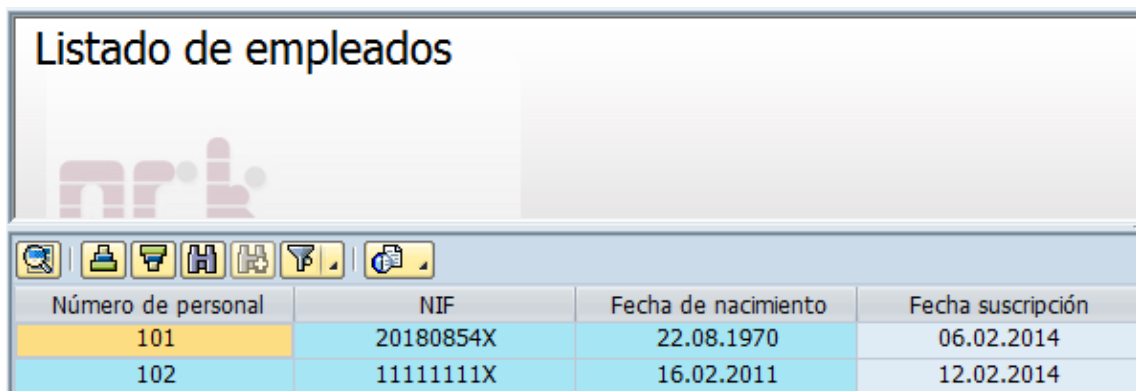
# **Eranskinak**



## A. ERANSKINA

### Harpidetutako langileen informea

Report honen bitartez, SAP sisteman langileak diren, eta METAPOSTako Sender-era harpidetuta dauden erabiltzaileen zerrenda ikus daiteke. Langile zerrenda ikusteaz gain, hauetako bat aukeratuta, bere informazio pertsonal orokorra ere ikus daiteke.

The image shows a screenshot of a SAP web interface. At the top, the title "Listado de empleados" is displayed. Below the title is a toolbar with several icons for actions like search, print, and refresh. The main content is a table with four columns: "Número de personal", "NIF", "Fecha de nacimiento", and "Fecha suscripción". The table contains two rows of data. The first row has a yellow background, and the second row has a light blue background.

Número de personal	NIF	Fecha de nacimiento	Fecha suscripción
101	20180854X	22.08.1970	06.02.2014
102	11111111X	16.02.2011	12.02.2014

**A.1 Irudia:** Langile zerrenda

## A.1 Dictionary

Hurrengo irudian agertzen diren Datu Baseko taulak erabili dira bilaketak egiteko. Taula hauek, SAPeko HR (Giza Baliabideen) modulua baliabide estandarrekin erlazionatutako datuak erabiltzen dituzte.

```
*****
*Declaración de las tablas de BD
*****
TABLES: zmtp_t203.
```

### A.2 Irudia: Datu Baseko taulak

Zerrenda ALV baten bidez erakusteko, objektuetara orientatutako programazioa erabiltzen da. Horretarako, hurrengo datu egiturak, taulak, aldagai eta klaseak erabiltzen dira.

```

| *****
| *Declaración de variables
| *****
| * Campos relacionados con el ALV
| DATA: go_container TYPE REF TO cl_gui_custom_container, "Objeto para Container
|        go_splitter TYPE REF TO cl_gui_splitter_container, "Objeto splitter
|        go_container1 TYPE REF TO cl_gui_container, "Container para el header
|        go_container2 TYPE REF TO cl_gui_container, "Container para el ALV
|        go_document TYPE REF TO cl_dd_document, "Documento para TOP_OF_PAGE
|        go_alv TYPE REF TO cl_gui_alv_grid, "Objeto para ALV
|        gt_fieldcat TYPE lvc_t_fcat, "Cátalogo de campos
|        gs_layout TYPE lvc_s_layo, "Layout
|        gt_botones_excluir TYPE ui_functions,
|        ok_code TYPE sy-ucomm. "Código del usuario
|
| * Tabla para datos a mostrar
| DATA: gs_empleado TYPE zmtp_t203,
|        gt_empleados LIKE TABLE OF gs_empleado.
|
| * Clases para eventos
| CLASS lc_eventos DEFINITION DEFERRED.
| DATA: go_handler TYPE REF TO lc_eventos.
```

### A.3 Irudia: Datu egiturak eta objektuak

## A.2 Deskribapen teknikoa

Dynpro baten bidez, eta aurretik definitutako aldagai eta klaseak erabiliz, harpidetutako langileen zerrenda ALV baten bidez erakutsiko duen 'ZMTP\_REP\_EMPLEADOS' Report-a sortu da. Report honek, 5 INCLUDE desberdin ditu, definizioetarako (TOP), klaseetarako, subrutinentzako eta Module Pool-eko PBO eta PAI moduluentzako.

```

INCLUDE zmtplib_employeestop. " global Data
INCLUDE zmtplib_employeesc01. " Clases
INCLUDE zmtplib_employeesc01. " PBO-Modules
INCLUDE zmtplib_employeesc01. " PAI-Modules
INCLUDE zmtplib_employeesc01. " FORM-Routines
* Llamar al programa
START-OF-SELECTION.
  CALL SCREEN 0001.

```

### A.4 Irudia: Programa nagusia

Klase bat sortu da ere, erabiltzaileen interakzioak jasotzeko. Kasu honetan, *handle\_double\_click* ebentua kontrolatuko da, erabiltzaile batek click bikoitza egitean jasotzen dena.

```

METHOD handle_double_click.

  " Obtener empleado seleccionado
  CLEAR gs_empleado.
  READ TABLE gt_empleados INTO gs_empleado INDEX e_row-index.
  CHECK gs_empleado IS NOT INITIAL.

  " Mostrar el infotipo
  CALL FUNCTION 'HR_MASTERDATA_DIALOG'
    EXPORTING
      p_pernr      = gs_empleado-pernr
      p_infty      = '0002'
      p_activity   = 'DIS'
      p_skip       = 'X'
    EXCEPTIONS
      wrong_activity = 1
      no_authorization = 2
      OTHERS = 3.
  IF sy-subrc <> 0.
  ENDIF.

ENDMETHOD.

```

### A.5 Irudia: Klasea

### A.3 Prozesuaren logika

Lehendabizi, erakutsi behar diren harpidetutako langile guztiak lortzen dira eta *GT\_EMPLEADOS* taulan gordetzen dira, geroago erakusteko.

```
FORM obtener_datos.
  SELECT *
    INTO CORRESPONDING FIELDS OF TABLE gt_empleados
  FROM zmtpt203
  WHERE suscrito = 'X'.
ENDFORM.
```

#### A.6 Irudia: Datuak lortu

ALVaren eta bere kontenedorearen objektuak sortzen dira, eta aurretik sortutako klasea ebentuen *handler* bezala definitzen da.

```
* Crear container
CREATE OBJECT go_container
EXPORTING
  container_name = 'C_CONTAINER'.

* Crear el Splitter
CREATE OBJECT go_splitter
EXPORTING
  parent   = go_container
  rows     = 2
  columns  = 1.

* Crear los containers para el ALV y la cabecera
go_container1 = go_splitter->get_container( row = 1 column = 1 ).
go_splitter->set_row_height( id = 1 height = 20 ).
go_container2 = go_splitter->get_container( row = 2 column = 1 ).

* Crear el documento para el TOP_OF_PAGE
CREATE OBJECT go_document
EXPORTING
  style = 'ALV_GRID'.

* Crear el alv
CREATE OBJECT go_alv
EXPORTING
  i_parent = go_container2.

* Asignar clase como manejadora de los eventos
CREATE OBJECT go_handler.
SET HANDLER go_handler->handle_double_click FOR go_alv.
SET HANDLER go_handler->top_of_page FOR go_alv.
```

#### A.7 Irudia: Objektuak sortu



ALVaren elementu desberdinak informatzen dira (eremu katalogoa, layout-a, baztertu beharreko botoiak, etab.), eta jada ALVa erakuts daiteke.

```

* Configurar ALV
PERFORM informar_layout.
PERFORM informar_fieldcat.
PERFORM excluir_botones.

* Preparar ALV
CALL METHOD go_alv->set_table_for_first_display
EXPORTING
    i_structure_name      = 'GT_EMPLEADOS'
    i_save                 = 'A'
    is_layout              = gs_layout
    it_toolbar_excluding  = gt_botones_excluir
CHANGING
    it_outtab              = gt_empleados[]
    it_fieldcatalog        = gt_fieldcat[]
EXCEPTIONS
    invalid_parameter_combination = 1
    program_error               = 2
    too_many_lines               = 3
    OTHERS                       = 4.
IF sy-subrc <> 0.
ENDIF.

```

### A.8 Irudia: ALVa sortu

Orain, zerrendako langile batean click bikoitza egitean, SAPen 'PA20' transakzioaren bitartez, bere informazio zehatza agertuko da.

Nombre			
Tratamiento	Sra.		
Apellido	Posta	2º apellido	
Nombre	Meta		
Nº documento	EST11111111X	Tp.doc.	1 D.N.I (N.I.F.)
Datos nacimiento			
Fecha nacim.	16.02.2011	Lugar nacim.	Vitoria
Provincia	01 Álava	País nacim.	ES España
Idioma	ES Español	Nacionalidad	ES española
Otros datos			
Estado civil		Desde	
		Nº de hijos	0

### A.9 Irudia: Erabiltzailearen datuak



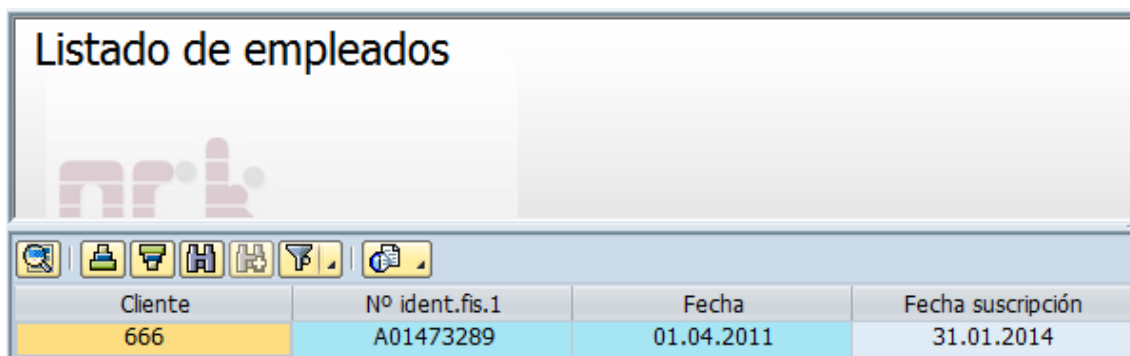
## B. ERANSKINA

---

### Harpidetutako bezeroen informea

---

Report honen bitartez, SAP sisteman bezeroak diren, eta METAPOSTako Sender-era harpidetuta dauden erabiltzaileen zerrenda ikus daiteke. Bezeroak zerrenda ikusteaz gain, hauetako bat aukeratuta, bere informazio orokorra ere ikus daiteke.



Cliente	Nº ident.fis.1	Fecha	Fecha suscripción
666	A01473289	01.04.2011	31.01.2014

**B.1 Irudia:** Langile zerrenda

## B.1 Dictionary

Hurrengo irudian agertzen diren Datu Baseko taulak erabili dira bilaketak egiteko. Taula hauek, SAPeko 'KNAI' taula estandarrekin erlazionatutako datuak erabiltzen dituzte.

```
*****:
*Declaración de las tablas de BD
*****:
TABLES: zntp_t303.
```

### B.2 Irudia: Datu Baseko taulak

Zerrenda ALV baten bidez erakusteko, objektuetara orientatutako programazioa erabiltzen da. Horretarako, hurrengo datu egiturak, taulak, aldagai eta klaseak erabiltzen dira.

```
☐ *****
| *Declaración de variables
| *****
| * Campos relacionados con el ALV
| DATA: go_container TYPE REF TO cl_gui_custom_container, "Objeto para Container
| go_splitter TYPE REF TO cl_gui_splitter_container, "Objeto splitter
| go_container1 TYPE REF TO cl_gui_container, "Container para el header
| go_container2 TYPE REF TO cl_gui_container, "Container para el ALV
| go_document TYPE REF TO cl_dd_document, "Documento para TOP_OF_PAGE
| go_alv TYPE REF TO cl_gui_alv_grid, "Objeto para ALV
| gt_fieldcat TYPE lvc_t_fcát, "Cátalogo de campos
| gs_layout TYPE lvc_s_layo, "Layout
| gt_botones_excluir TYPE ui_functions,
| ok_code TYPE sy-ucomm. "Código del usuario
|
| * Tabla para datos a mostrar
| DATA: gs_cliente TYPE zntp_t303,
| gt_clientes LIKE TABLE OF gs_cliente.
|
| * Clases para eventos
| CLASS lc_eventos DEFINITION DEFERRED.
| DATA: go_handler TYPE REF TO lc_eventos.
```

### B.3 Irudia: Datu egiturak eta objektuak

## B.2 Deskribapen tekniko

Dynpro baten bidez, eta aurretik definitutako aldagai eta klaseak erabiliz, harpidetutako langileen zerrenda ALV baten bidez erakutsiko duen 'ZMTP\_REP\_CLIEN TES' Report-a sortu da. Report honek, 5 INCLUDE desberdin ditu, definizioetarako (TOP), klaseetarako, subrutinentzako eta Module Pool-eko PBO eta PAI moduluentzako.

```
INCLUDE zntp_rep_clientestop. " global Data
INCLUDE zntp_rep_clientesc01. " Clases
INCLUDE zntp_rep_clienteso01. " PBO-Modules
INCLUDE zntp_rep_clientesi01. " PAI-Modules
INCLUDE zntp_rep_clientesf01. " FORM-Routines

* Llamar al programa
START-OF-SELECTION.
  CALL SCREEN 0001.
```

### B.4 Irudia: Programa nagusia

Klase bat sortu da ere, erabiltzaileen interakzioak jasotzeko. Kasu honetan, *handle\_double\_click* ebentua kontrolatuko da, erabiltzaile batek click bikoitza egitean jasotzen dena.

```
CLASS lc_eventos IMPLEMENTATION.
METHOD handle_double_click.
* Obtener cliente seleccionado
  CLEAR gs_cliente.
  READ TABLE gt_clientes INTO gs_cliente INDEX e_row-index.
  CHECK gs_cliente IS NOT INITIAL.

  SET PARAMETER ID 'KUN' FIELD gs_cliente-kunnr.
  CALL TRANSACTION 'XD03' AND SKIP FIRST SCREEN.

ENDMETHOD.
ENDCLASS.
```

### B.5 Irudia: Klasea

## B.3 Prozesuaren logika

Lehendabizi, erakutsi behar diren harpidetutako bezero guztiak lortzen dira eta *GT\_CLIENTES* taulan gordetzen dira, geroago erakusteko.

```
| FORM obtener_datos.
  SELECT *
    INTO CORRESPONDING FIELDS OF TABLE gt_clientes
    FROM zmtp_t303
    WHERE suscrito = 'X'.
ENDFORM.
```

### B.6 Irudia: Datuak lortu

ALVaren eta bere kontenedorearen objektuak sortzen dira, eta aurretik sortutako klasea ebentuen *handler* bezala definitzen da.

```
* Crear container
CREATE OBJECT go_container
EXPORTING
  container_name = 'C_CONTAINER'.

* Crear el Splitter
CREATE OBJECT go_splitter
EXPORTING
  parent   = go_container
  rows    = 2
  columns = 1.

* Crear los containers para el ALV y la cabecera
go_container1 = go_splitter->get_container( row = 1 column = 1 ).
go_splitter->set_row_height( id = 1 height = 20 ).
go_container2 = go_splitter->get_container( row = 2 column = 1 ).

* Crear el documento para el TOP_OF_PAGE
CREATE OBJECT go_document
EXPORTING
  style = 'ALV_GRID'.

* Crear el alv
CREATE OBJECT go_alv
EXPORTING
  i_parent = go_container2.

* Asignar clase como manejadora de los eventos
CREATE OBJECT go_handler
```

### B.7 Irudia: Objektuak sortu

ALVaren elementu desberdinak informatzen dira (eremu katalogoa, layout-a, baztertu beharreko botoiak, etab.), eta jada ALVa erakuts daiteke.

```
* Configurar ALV
  PERFORM informar_layout.
  PERFORM informar_fieldcat.
  PERFORM excluir_botones.

* Preparar ALV
  CALL METHOD go_alv->set_table_for_first_display
    EXPORTING
      i_structure_name      = 'GT_CLIENTES'
      i_save                = 'A'
      is_layout             = gs_layout
      it_toolbar_excluding = gt_botones_excluir
    CHANGING
      it_outtab            = gt_clientes[]
      it_fieldcatalog      = gt_fieldcat[]
    EXCEPTIONS
      invalid_parameter_combination = 1
      program_error                 = 2
      too_many_lines                 = 3
      OTHERS                         = 4.
  IF sy-subrc <> 0.
  ENDIF.
```

### B.8 Irudia: ALVa sortu

Orain, zerrendako bezero batean click bikoitza egitean, SAPen 'XD03' transakzioaren bitartez, bere informazioa agertuko da.

The screenshot shows the SAP 'XD03' transaction interface. At the top, the debtor name 'Amvisa' is displayed with a red box around the '66' in the account number field. Below this, there are several tabs: 'Dirección', 'Datos de control', 'Pagos', and 'Marketing'. The 'Dirección' tab is active, showing a form with the following fields:

- Nombre:**
  - Tratamiento: Empresa
  - Nombre: Amvisa
  - 20110401
- Dirección calle:**
  - Calle/Número: [Empty]
  - Código postal/Pobl.: [Empty]
  - País: ES (España)
  - Región: [Empty]
- Comunicación:**
  - Idioma: ES Español
  - Teléfono: [Empty]
  - Teléfono móvil: [Empty]
  - Fax: [Empty]
  - E-mail: [Empty]
  - Extensión: [Empty]
  - Otras comunicaciones...: [Empty]

### B.9 Irudia: Erabiltzailearen datuak