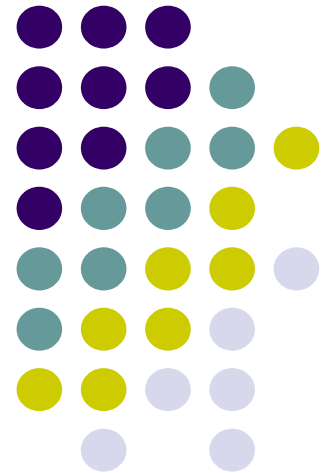


8.- DATUEN MAILA JAVAEEⁿ ESKEMAK

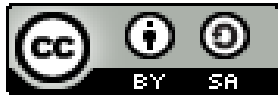
ZERBITZU TELEMATIKO AURRERATUAK
Telekomunikazio Teknologiako Ingeniaritza Gradua, 3. maila

2023-2024

Egileak: Maider Huarte Arrayago, Gorka Prieto Agujeta



2022-2023 ZTA 8 DATUEN MAILA JAVAEEEn.odp



Copyright © 2013-2024 Maider Huarte Arrayago, Gorka Prieto Agujeta



2023-2024 ZTA 8 DATUEN MAILA JAVAEEEn.odp lana, Maider Huartek eta Gorka Prietok egina, Creative Commons-en Attribution-Share Alike 3.0 Unported License baimenaren menpe dago. Baimen horren keria bat ikusteko, <http://creativecommons.org/licenses/by-sa/3.0/> webgunea bisitatu edo gutun bat bidali ondoko helbidera: Creative Commons, 171 2nd Street, Suite 300, San Francisco, California, 94105, USA.

2023-2024 ZTA 8 DATUEN MAILA JAVAEEEn.odp by Maider Huarte and Gorka Prieto is licensed under a Creative Commons Attribution-Share Alike 3.0 Unported License. To view a copy of this license, visit <http://creativecommons.org/licenses/by-sa/3.0/> or, send a letter to Creative Commons, 171 2nd Street, Suite 300, San Francisco, California, 94105, USA.

DATUEN MAILA JAVAEE_n

EDUKIA

1.- SARRERA

2.- JAVA EE

ERREFERENTZIAK

DATUEN MAILA JAVAEE_n

EDUKIA

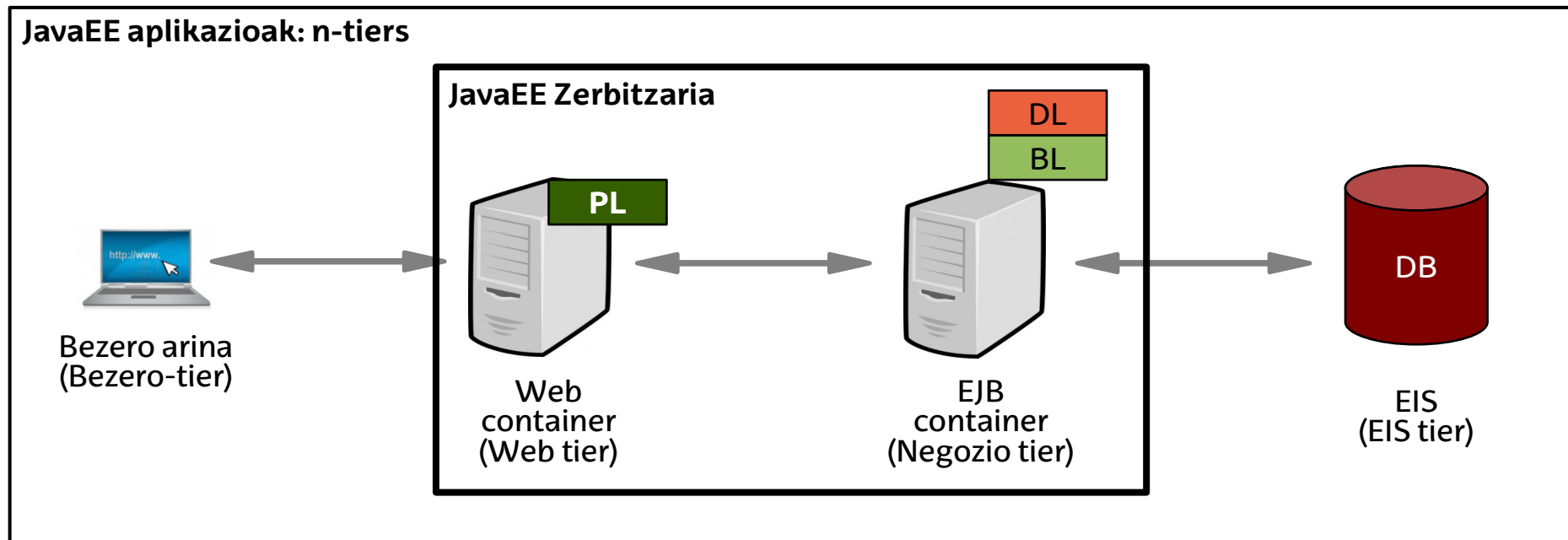
1.- SARRERA

2.- JAVA EE

ERREFERENTZIAK

1.- SARRERA

- Exekuzio eremua
 - Web zerbitzaria+containers
 - Aplikazio zerbitzaria *Wildfly*
- JavaEE aplikazioetarako eredua



DATUEN MAILA JAVAEEⁿ

EDUKIA

1.- SARRERA

2.- JAVA EE

2.1.- Sarrera

2.2.- Bussiness tier

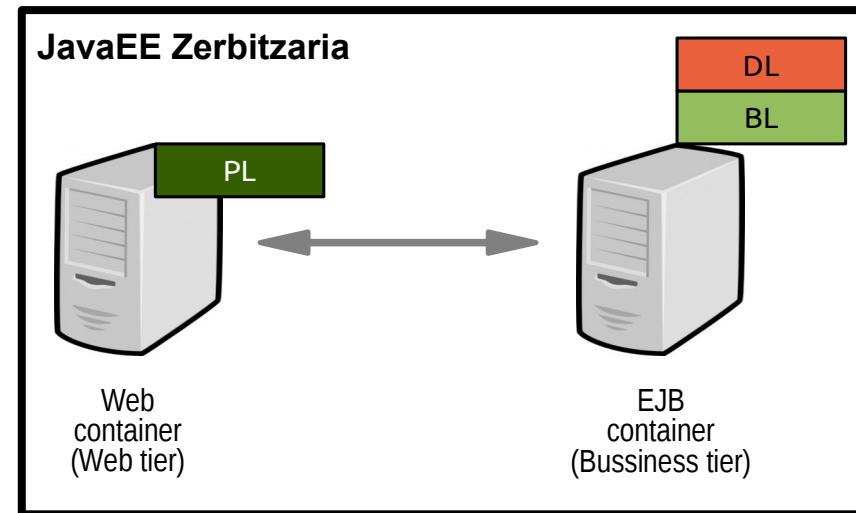
ERREFERENTZIAK

2.- JAVA EE

2.1.- Sarrera

- Bussiness tiereko Java osagaiak
 - Negozio maila
 - Web Zerbitzuak
 - Astunak (SOAP)/Arinak (REST)
 - WS bezeroak
 - EJBak
 - Java klase eta interfazeak
 - EJB bezeroak
 - Managed Beans
 - EJB
 - Datuak atzitzeko maila: datuen iraunkortasunerako entitateak
 - Java klaseak
 - Datuak modu iraunkorrean atzitzeko

- Zerbitzaria itzalean ez "ahazteko"
- "Lasterketa"/"Leiha" arazoak ekiditeko
- Datu Base Sistemak



DATUEN MAILA JAVAEE_n

EDUKIA

1.- SARRERA

2.- JAVA EE

2.1.- Sarrera

2.2.- Bussiness tier

ERREFERENTZIAK

2.- JAVA EE

2.2.- Bussiness tier

- Datuak atzitzeko maila: datuen iraunkortasuna

Persistence

- Kontzeptuak

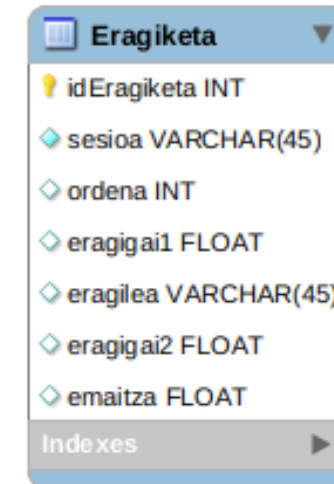
- JPA Datu-baseekin lan egiteko JavaEE API
- Entitate klaseak
 - Datu Baseko taula baten egitura adierazten duen JB klasea
 - JPA oharrak
 - Zutabe bakoitza: atributu bat
- Entitate objektuak
 - JPA oharrak dituen klase baten instantzia
 - Datu Baseko taula baten lerro bateko balioak

- Iraunkortasun egoera: EJB Container

Entitate objektua eta lerroa sinkronizatuta egotea

- Entitate klaseen betebeharrak DERRIGORREZ

- JavaBean: **Serializable**
- **final** ez
- JPA oharrak



2.- JAVA EE

2.2.- Bussiness tier

Hauek gehien bat automatikoki egingo dira, baina errebisatuko ditugu beharbada. Bereziki kontsultak eta atributuak!

- Datuak atzitzeko maila: datuen iraunkortasuna
 - JPA oharrak
 - Derrigorrezkoak
 - **@Entity** Klasean, entitate klasea dela adierazteko
 - **@Id** Primary Key dagokion atribututan
 - Atributuak
 - **@GeneratedValue** Primary Key dagokion atributuan, balioa nola sortuko den adierazteko
 - **@Column**
 - **@Pattern**
 - Kontsultak Klasean, taularekiko kontsulta orokorrak programatzeko
 - **@NamedQuery(name="izend", query="JPQLEskaera")**
 - **@NamedQueries({namedQuery1,...})**
 - Taulen arteko erlazioak Foreign key dagozkien atributuetan
 - **@OneToOne**
 - **@OneToMany**
 - **@ManyToOne**
 - **@ManyToMany**

2.- JAVA EE

2.2.- Bussiness tier

- Datuak atzitzeko maila: datuen iraunkortasuna
 - Iraunkortasunaren kudeaketa
 - Iraunkortasunaren Testuingurua (*persistence context*)
 - Iraunkortutako entitateak Taula bateko lerro batekin sinkronizatuta dauden entitate objektuen multzoa
 - Sinkronizazio gardena Iraunkortutako objektua eta Datu Basean dagokion lerroaren artean
 - Kudeatzailea: **EntityManager**
 - **EntityManager** Testuingurua kudeatuko duen objektuaren mota (klasea)
 - Aplikazioak kudeatuta
 - EJB Container-ak kudeatuta Sinkronizazio gardena + transaction
 - Java Transaction API (JTA) Datu Base eskaerak transaction modura lantzeko JavaEE API-a
 - Erabilpena adierazteko **EJB batean injeztatu: @PersistenceContext**
 - Ohar hori duen atributua deklaratu
 - EZ SORTU new REKIN!!

2.- JAVA EE

2.2.- Bussiness tier

- Datuak atzitzeko maila: datuen iraunkortasuna

- Iraunkortasunaren kudeaketa

- **EntityManager** kudeatzailearen eragiketak (metodoak)

- **find**: `SELECT FROM x WHERE x.pk= ...`

- **createNamedQuery**: kontsulta estatikoak

- **createQuery**: kontsulta dinamikoak

- **persist**: `INSERT`

- **remove**: `DELETE`

- ...

`em.find(primaryKey)`

SOILIK PRIMARY KEY HARTUKO DU

-Ezin du beste zutabeekiko bilatu

-Itzulera

- Entitate-objektu bakarra, iraunkortuta

- null

Datu basearekiko kontsulta orokorrak

- Select from x where...

- Itzulera: entitate-objektuen List iraunkortuta

- Beteta

- Hutsik: `size==0`

persist:

- Datu baseko taula batean lerro BERRIAK sartzeko

- Iraunkortuta ez zegoen entitate-objektuak iraunkortuko ditu

remove:

- Datu baseko taula batean lerro bat ezabatzeke

- Iraunkortuta zegoen entitate-objektua iraunkortuen multzotik atera

2.- JAVA EE

2.2.- Bussiness tier

- Datuak atzitzeko maila: datuen iraunkortasuna
 - Datu-baseekiko kontsultak: aukerak
 - Java Persistence Query Language (JPQL) createNamedQuery/createQuery metodoak konfiguratzeko
 - SQL lengoaiaren antzekoa
 - Emaizten casting beharrezkoa Query kontsulten emaitzak
 - List <Object>
 - Object
 - Irakurri/idazteko erraza
 - Criteria API TOP GEAR TOP TIP: AVOID
 - Zuzenean SQL
 - Ez da beharrezkoa casting egitea
 - Kodea luzeagoa: bitarteko objektu eta eragiketak

2.- JAVA EE

2.2.- Bussiness tier

- Datuak atzitzeko maila: datuen iraunkortasuna
 - Datu-baseekiko kontsultak
 - Kontsulta dinamikoak JPQL

```
public class OrokorraEJB {
    @PersistenceContext private EntityManager em;
    ...
    Casting!
    public List<Eroslea> erosleakIzenaLortuDB(String izena) {
        return (List<Eroslea>)em.createQuery("SELECT e FROM Eroslea e WHERE e.izena= :erosleIzena")
            .setParameter("erosleIzena", izena)
            .getResultList();
    }
    ...
}
```

Emitza orokorrean taulako lerro bat baino gehiago!

Izen ezberdinen arabera bilatzeko

```
@Entity
public class Eroslea implements Serializable {
    private String izena;
    private String helbidea;
    private int adina;
    ...
}
```

- Kontsulta estatikoak JPQL

```
public class OrokorraEJB {
    @PersistenceContext public EntityManager em;
    ...
    Casting!
    public List<Eroslea> erosleakIzenaLortuDB(String izena) {
        return (List<Eroslea>)em.createNamedQuery("Eroslea.findIzena")
            .setParameter("erosleIzena", izena)
            .getResultList();
    }
    ...
}
```

Emitza orokorrean taulako lerro bat baino gehiago!

Eroslea.findIzena beste metodo batzuetan berrerabili daiteke

```
@NamedQuery(
    name="Eroslea.findIzena",
    query="SELECT e FROM Eroslea e WHERE e.izena= :erosleIzena"
)
@Entity
public class Eroslea implements Serializable {
    private String izena;
    private String helbidea;
    private int adina;
    ...
}
```

2.- JAVA EE

2.2.- Bussiness tier

- Datuak atzitzeko maila: datuen iraunkortasuna
 - DBrako konexioaren konfigurazioa
 - JBoss/Wildfly: standalone.xml
 - DBrako konexioaren izena JNDI Formatuan
 - Konexioaren datuak -> Datu basearen URL-a
Erabiltzaile kontu bat: username, password
 - Konexioaren driver-ak
 - JavaEE proiektuan: persistence.xml
 - DBrako konexioaren izena JNDI Formatuan
 - Iraunkortasun entitateen klaseak
 - DB mota eta taulen hasierako kudeaketa: **update, create, validate**

Entity klaseek adierazitako taulak eta Datu Basekoa
bat ez badatoz, proiektua ez da kargatzen

2.- JAVA EE

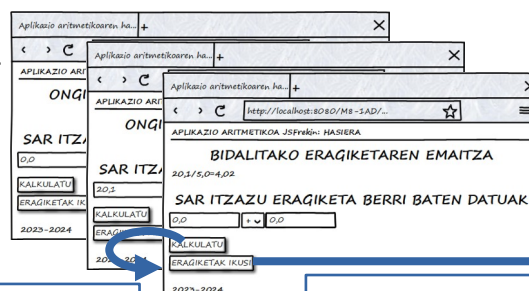
2.2.- Bussiness tier

- JSF+EJB+JP 1. ADIBIDEA: Aplikazio aritmetikoa JSF, EJB eta JPEntities-ekin

1. pantaila: Hasierako egoera

1. pantaila: Erabiltzaileak formularioa beteta

1. pantaila: Emaita + formularioa



Aplikazio aritmetikoaren ha...

APLIKAZIO ARITMETIKOA JSFekin: HASIERA

BIDALITAKO ERAGIKETAREN EMAITZA

20,1/5,0+4,02

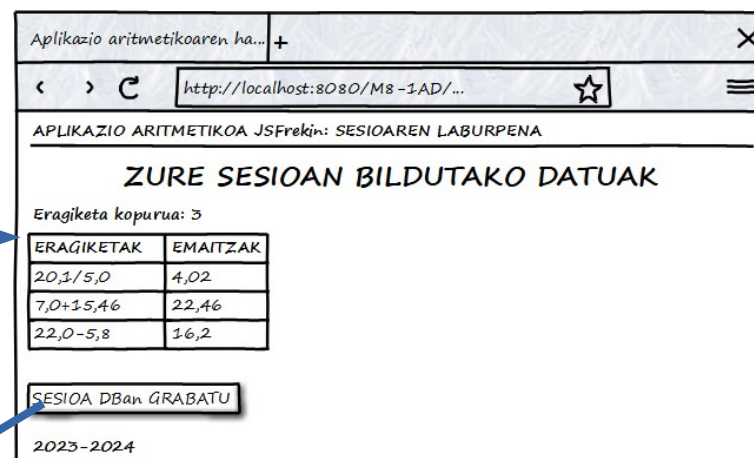
SAR ITZAZU ERAGIKETA BERRI BATEN DATUAK

2023-2024

BESTE ADIBIDEETAN BEZALA:
Kalkulua egin
Eragiketa sesioko EJBko zerrendan sartu
Emaita + formularioa bistaratu

2. pantailara

2. pantaila: Sesioren laburpena



Aplikazio aritmetikoaren ha...

APLIKAZIO ARITMETIKOA JSFekin: SESIOAREN LABURPENA

ZURE SESIOAN BILDUTAKO DATUAK

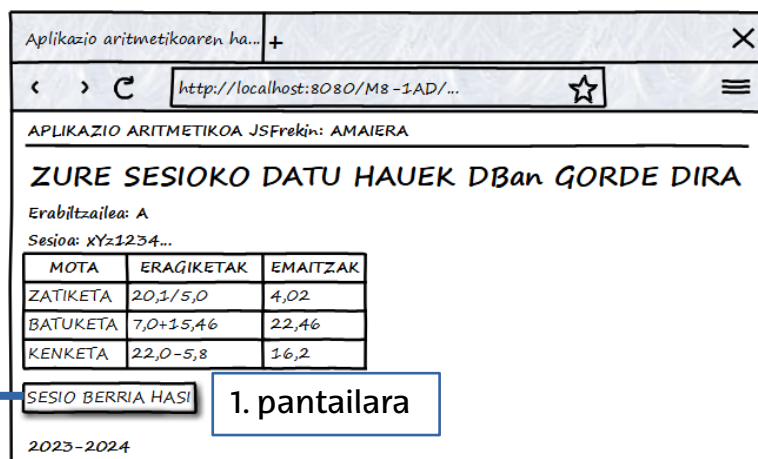
Eragiketa kopurua: 3

ERAGIKETAK	EMAITZAK
20,1/5,0	4,02
7,0+15,46	22,46
22,0-5,8	16,2

SESIOA DBan GRABATU

2023-2024

3. pantaila: Sesioka DBan



Aplikazio aritmetikoaren ha...

APLIKAZIO ARITMETIKOA JSFekin: AMAIERA

ZURE SESIOKO DATU HAUEK DBan GORDE DIRA

Erabiltzailea: A

Sesioa: xYz1234...

MOTA	ERAGIKETAK	EMAITZAK
ZATIKETA	20,1/5,0	4,02
BATUKETA	7,0+15,46	22,46
KENKETA	22,0-5,8	16,2

SESIO BERRIA HASI

2023-2024

EJBko zerrenda DBan sartu
3. pantailara

1. pantailara

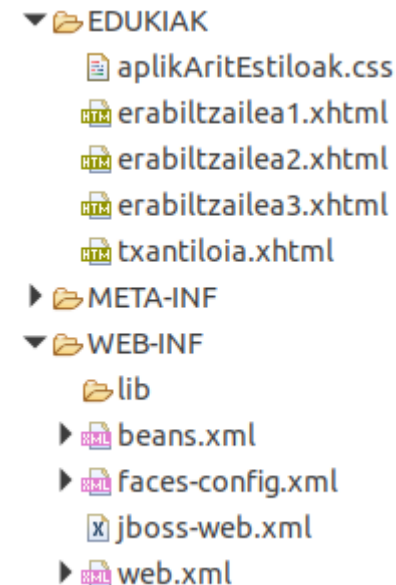
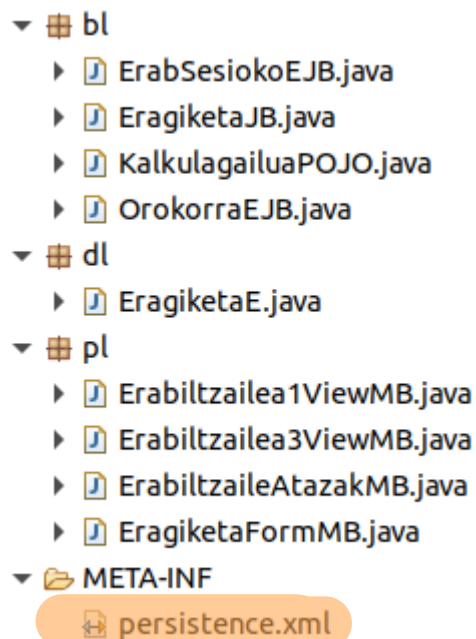
2.- JAVA EE

2.2.- Bussiness tier

- JSF+EJB+JP 1. ADIBIDEA: Aplikazio aritmetikoa JSF, EJB eta JPEntities-ekin

– JAVA osagaiak: **java**

– HTML osagaiak: **webapp**



2.- JAVA EE

2.2.- Bussiness tier

- JSF+EJB+JP 1. ADIBIDEA: Aplikazio aritmetikoa JSF, EJB eta JPEntities-ekin

```
<?xml version="1.0" encoding="UTF-8"?>
<persistence version="2.1" xmlns="http://xmlns.jcp.org/xml/ns/persistence"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://xmlns.jcp.org/xml/ns/persistence
http://xmlns.jcp.org/xml/ns/persistence/persistence_2_1.xsd">
  <persistence-unit name="ProiektuarenIzena" transaction-type="JTA">
    <jta-data-source>java:jboss/datasources/DatuBasearenIzena</jta-data-source>
    <properties>
      <property name="hibernate.dialect" value="org.hibernate.dialect.MySQLDialect" />
      <property name="hibernate.hbm2ddl.auto" value="validate" />
    </properties>
  </persistence-unit>
</persistence>
```

persistence.xml

Gure datu basea zein den

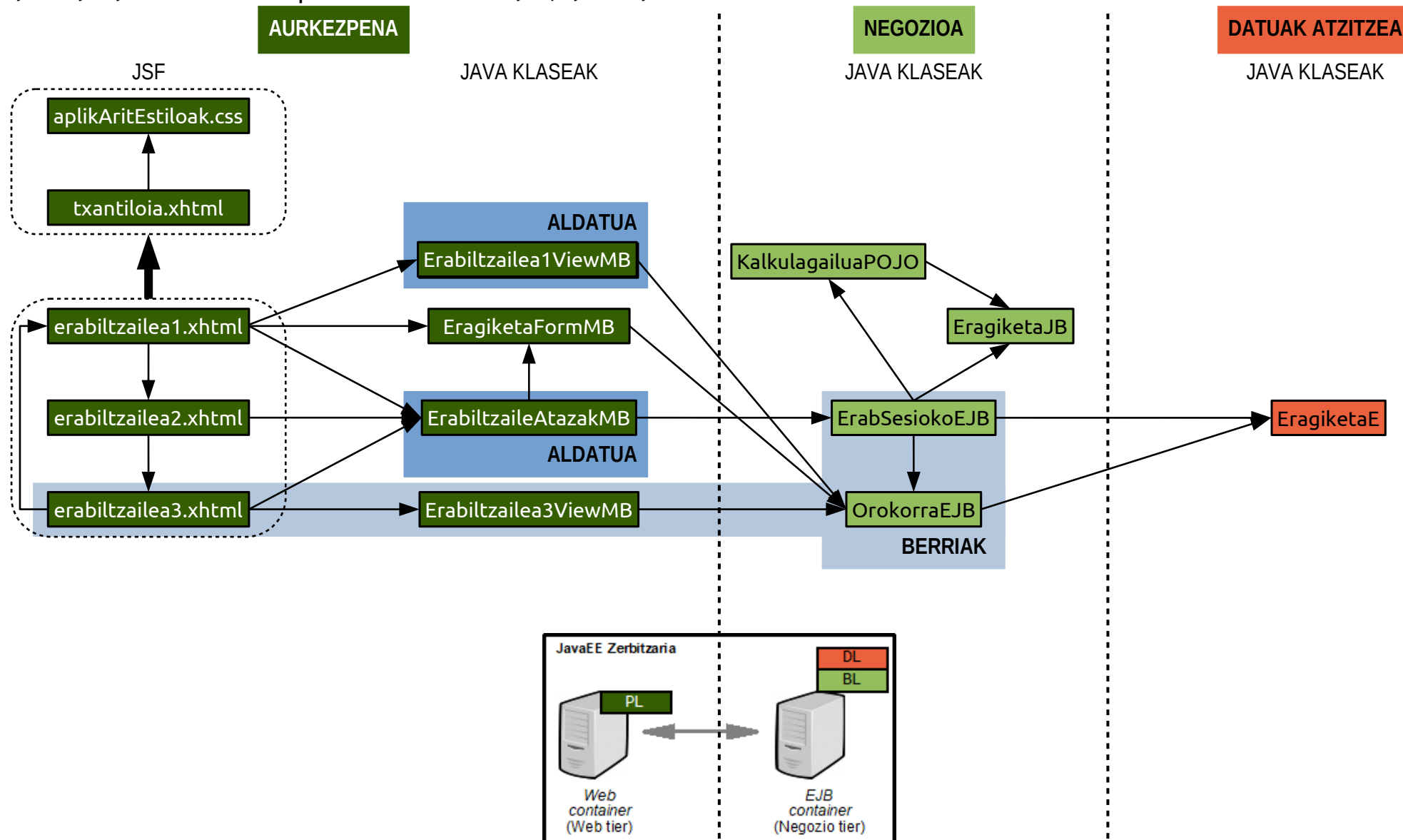
```
CREATE TABLE IF NOT EXISTS `DatuBasearenIzena`.`Eragiketa` (
  `idEragiketa` INT NOT NULL AUTO_INCREMENT,
  `sesioa` VARCHAR(100) NOT NULL,
  `ordena` INT NULL,
  `eragigai1` FLOAT NULL,
  `eragilea` VARCHAR(45) NULL,
  `eragigai2` FLOAT NULL,
  `emaitza` FLOAT NULL,
  PRIMARY KEY (`idEragiketa`))
ENGINE = InnoDB;
```

Eragiketa	
idEragiketa	INT
sesioa	VARCHAR(45)
ordena	INT
eragigai1	FLOAT
eragilea	VARCHAR(45)
eragigai2	FLOAT
emaitza	FLOAT
Indexes	

2.- JAVA EE

2.2.- Bussiness tier

- JSF+EJB+JP 1. ADIBIDEA: Aplikazio aritmetikoa JSF, EJB eta JPEntities-ekin



2.- JAVA EE

2.2.- Bussiness tier

- JSF+EJB+JP 1. ADIBIDEA: Aplikazio aritmetikoa JSF, EJB eta JPEntities-ekin

```
package dl;

import java.io.Serializable;
import javax.persistence.*;

@Entity
@Table(name="Eragiketa")
@NamedQueries ({
    @NamedQuery(name="EragiketaE.findAll", query="SELECT e FROM EragiketaE e"),
    @NamedQuery(name="EragiketaE.findAllSesioa", query="SELECT e FROM EragiketaE e WHERE e.sesioa = :sesioa ORDER BY e.ordena")
})
public class EragiketaE implements Serializable {
    private static final long serialVersionUID = 1L;

    @Id
    @GeneratedValue(strategy=GenerationType.IDENTITY)
    private int idEragiketa;

    @Column(nullable=false)
    private String sesioa;

    //TAULAKO BESTE ZUTABEEI DAGOZKIEN ATRIBUTUAK (null izan daitezke)

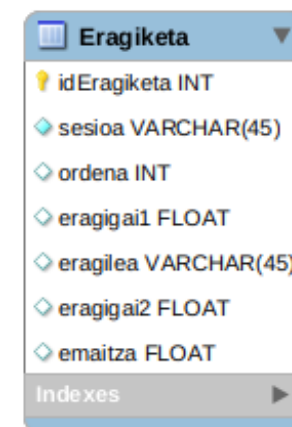
    public EragiketaE() {
    }

    public EragiketaE(String sesioa, int ordena, float eragigai1, String eragilea, float eragigai2, float emaitza) {
        this.sesioa=sesioa;
        this.ordena=ordena;
        this.eragigai1=eragigai1;
        this.eragilea=eragilea;
        this.eragigai2=eragigai2;
        this.emaitza=emaitza;
    }

    //Atributu guztien getter eta setter metodoak

    public String laburpena() {
        return ""+eragigai1+eragilea+eragigai2+"="+emaitza;
    }
}
```

Entity: dl.EragiketaE.java



2.- JAVA EE

2.2.- Bussiness tier

- JSF+EJB+JP 1. ADIBIDEA: Aplikazio aritmetikoa JSF, EJB eta JPEntities-ekin

EJB: bl.ErabSesiokoEJB.java

```
package bl;

//import lerroak

@SecurityDomain("SecurityDomainIzena")

@DenyAll
@Stateful
@LocalBean
public class ErabSesiokoEJB {
    private KalkulagailuaPOJO kPOJO=new KalkulagailuaPOJO();
    private List<ErariketaE> erariketakE=new ArrayList<ErariketaE>();

    @EJB OrokorraEJB oEJB;

    @RolesAllowed("erabiltzailea")
    public List<ErariketaE> getErariketakE() {
        return erariketakE;
    }

    @RolesAllowed("erabiltzailea")
    public int erariketaEginZerrendaratu(ErariketaE erariketaE) {
        int kodea=1;

        ErariketaJB erariketaJB=new ErariketaJB(erariketaE.getErarikigai1(), erariketaE.getErarikilea(), erariketaE.getErarikigai2(), 0.0f);
        kPOJO.kalkulatu(erariketaJB);

        erariketaE.setEmaitza(erariketaJB.getEmaitza());
        erariketakE.add(erariketaE);

        return kodea;
    }

    @RolesAllowed("erabiltzailea")
    public int zerrendaBiltegitatu(String sesioIzena) {
        return oEJB.erariketakSartuDB(erariketakE, sesioIzena);
    }
}
```

2.- JAVA EE

2.2.- Bussiness tier

- JSF+EJB+JP 1. ADIBIDEA: Aplikazio aritmetikoa JSF, EJB eta JPEntities-ekin

EJB: bl.OrokorraEJB.java

```
package bl;

//import lerroak

@SecurityDomain("SecurityDomainIzena")

@DenyAll
@Stateless           Atributu bakarra injektatua! Stateless!
@LocalBean
public class OrokorraEJB {

    @PersistenceContext private EntityManager em;

    @RolesAllowed("erabiltzailea")
    public List<String> eragileakLortu() {...} //7. GAIKO AZKEN ADIBIDEKO ZerbitzuakeEJB KLASEKO KODE BERDINA

    @RolesAllowed("erabiltzailea")
    public int eragiketakSartuDB(List<EragiketaE> eragiketakE, String sesioaIzena) {
        int kodea=2;

        EragiketaE eE;
        for(int i=0;i<eragiketakE.size();i++) {
            eE = eragiketakE.get(i);
            eE.setOrdena(i+1);
            eE.setSesioa(sesioaIzena);

            em.persist(eE);
        }

        return kodea;
    }

    @RolesAllowed("erabiltzailea")
    public List<EragiketaE> sesiokoEragiketakLortuDB(String sesioIzena) {
        @SuppressWarnings("unchecked")
        List<EragiketaE> eragiketakDB = (List<EragiketaE>)em.createNamedQuery("EragiketaE.findAllSesioa")
            .setParameter("sesioa",sesioIzena).getResultList();

        return eragiketakDB;
    }
}
```

2.- JAVA EE

2.2.- Bussiness tier

- JSF+EJB+JP 1. ADIBIDEA: Aplikazio aritmetikoa JSF, EJB eta JPEntities-ekin

//AURREKO ADIBIDEKO KODE BERDINA

pl.EragiketaFormMB.java

```
package pl;

import java.io.Serializable;
import java.util.List;

import javax.faces.view.ViewScoped;
import javax.inject.Named;

import bl.OrokorraEJB;

@Named
@ViewScoped
public class Erabiltzailea1ViewMB implements Serializable {
    private static final long serialVersionUID = 1L;

    private List<String> eragileakDB;

    @EJB private OrokorraEJB oEJB;

    public void resetView() {
        eragileakDB=null;
        return;
    }

    public List<String> eragileakLortu() {
        if(eragileakDB==null)
            eragileakDB=oEJB.eragileakLortu();

        return eragileakDB;
    }
}
```

pl.Erabiltzailea1View.java

2.- JAVA EE

2.2.- Bussiness tier

- JSF+EJB+JP 1. ADIBIDEA: Aplikazio aritmetikoa JSF, EJB eta JPEntities-ekin

MB: pl.ErabilitzaileAtazakMB.java

```
package pl;

//import lerroak

@Named
@SessionScoped
public class ErabilitzaileAtazakMB implements Serializable {
    private static final long serialVersionUID = 1L;

    private String sesioIzena;
    private int kodea;
    private String mezua;

    @EJB private ErabSesiokoEJB eEJB;

    public ErabilitzaileAtazakMB() {
        String userIP=((HttpServletRequest)FacesContext.getCurrentInstance().getExternalContext().getRequest()).getRemoteAddr();
        HttpSession httpSession=(HttpSession)FacesContext.getCurrentInstance().getExternalContext().getSession(true);
        sesioIzena="IP-BEZEROA:"+userIP+" TIME:"+new Date(httpSession.getCreationTime());
    }

    public String getSesioIzena() {
        return sesioIzena;
    }

    public String getKodea() {...} //AURREKO ADIBIDEKO KODE BERDINA
    public String getMezua() {...} //AURREKO ADIBIDEKO KODE BERDINA

    public void resetSession() {
        sesioIzena="";
        mezua="";
        eragiketakJB=null;
        FacesContext.getCurrentInstance().getExternalContext().invalidateSession();
    }

    //HEMENGOKO KODE GUZTIA HURRENGO ORRIALDEAN DAGO
}
```


2.- JAVA EE

2.2.- Bussiness tier

- JSF+EJB+JP 1. ADIBIDEA: Aplikazio aritmetikoa JSF, EJB eta JPEntities-ekin

//HEMENGO KODE GUZTIA AURREKO ORRIALDEAN DAGO

MB: pl.ErabiltzaileAtazakMB.java

```
public void eragiketaEginZerrendaratu(EragiketaFormMB eragiketaFormMB, Erabiltzailea1ViewMB erabiltzailea1ViewMB) {
    EragiketaE eragiketaE=new EragiketaE("", 0, eragiketaFormMB.getEr1(), ""+eragiketaFormMB.getEr(), eragiketaFormMB.getEr2(), 0.0f);

    kodea=eEJB.eragiketaEginZerrendaratu(eragiketaE);
    mezua=eragiketaE.laburpena();

    eragiketaFormMB.resetForm();
    erabiltzailea1ViewMB.resetView();
}

public List<EragiketaE> eragiketakLortuEJB() {
    return eEJB.getEragiketakE();
}

public void zerrendaBiltegiratu() {
    kodea=eEJB.zerrendaBiltegiratu(sesioIzena);
    mezua="ZURE SESIOKO DATU HAUEK DBan GORDE DIRA";
}
}
```

2.- JAVA EE

2.2.- Bussiness tier

- JSF+EJB+JP 1. ADIBIDEA: Aplikazio aritmetikoa JSF, EJB eta JPEntities-ekin

MB: pl.Erabilitzailea3ViewMB.java

```
package pl;

//import lerroak

@Named
@ViewScoped
public class Erabilitzailea3ViewMB implements Serializable {
    private static final long serialVersionUID = 1L;
    // Datu baseko datuak lortu

    private List<EragiketaE> eragiketakDB;

    @EJB private OrokorraEJB oEJB;

    public void resetView() {
        eragiketakDB=null;
        return;
    }

    public List<EragiketaE> sesiokoEragiketakLortuDB(String sesioIzena) {
        if(eragiketakDB==null)
            eragiketakDB=oEJB.sesiokoEragiketakLortuDB(sesioIzena);
        // Lazy getter

        return eragiketakDB;
    }
}
```

2.- JAVA EE

2.2.- Bussiness tier

- JSF+EJB+JP 1. ADIBIDEA: Aplikazio aritmetikoa JSF, EJB eta JPEntities-ekin

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml"
      xmlns:h="http://java.sun.com/jsf/html"
      xmlns:f="http://java.sun.com/jsf/core"
      xmlns:ui="http://java.sun.com/jsf/facelets">
<ui:composition template="./txantiloia.xhtml">
<ui:define name="headTitle">Erabiltzailea2: Aplikazio aritmetikoaren sesio-laburpenaren orrialdea</ui:define>
<ui:define name="bodyBurua">APLIKAZIO ARITMETIKOA JSF, EJB eta JPArekin: SESIOAREN LABURPENA</ui:define>
<ui:define name="bodyEdukia">
  <h1>ZURE SESIOAN BILDUTAKO DATUAK</h1>
  Eragiketa kopurua: #{erabiltzaileAtazakMB.eragiketakLortuEJB().size()}
  <h:dataTable value="#{erabiltzaileAtazakMB.eragiketakLortuEJB()}" var="eragiketa" border="2">
    <h:column>
      <f:facet name="header">ERAGIKETAK</f:facet>
      #{eragiketa.eragigai1}#{eragiketa.eragilea}#{eragiketa.eragigai2}
    </h:column>
    <h:column>
      <f:facet name="header">EMAITZAK</f:facet>
      #{eragiketa.emaitza}
    </h:column>
  </h:dataTable>
  <p>
    <h:form>
      <h:commandButton action="erabiltzailea3" actionListener="#{erabiltzaileAtazakMB.zerrendaBiltegitatu()}" id="submit"
        value="SESIOA DBan GRABATU" />
    </h:form>
  </p>
</ui:define>
</ui:composition>
</html>
```

XHTML: erabiltzailea2.xhtml

2.- JAVA EE

2.2.- Bussiness tier

- JSF+EJB+JP 1. ADIBIDEA: Aplikazio aritmetikoa JSF, EJB eta JPEntities-ekin

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml"
      xmlns:h="http://java.sun.com/jsf/html"
      xmlns:f="http://java.sun.com/jsf/core"
      xmlns:ui="http://java.sun.com/jsf/facelets">
<ui:composition template="./txantiloia.xhtml">
<ui:define name="headTitle">Erabiltzailea3: Aplikazio aritmetikoaren amaierako orrialdea</ui:define>
<ui:define name="bodyBurua">APLIKAZIO ARITMETIKOA JSF, EJB eta JPArekin: AMAIERA</ui:define>
<ui:define name="bodyEdukia">
  <h1>#{erabiltzaileAtazakMB.mezua}</h1>
  <h:panelGroup rendered="#{erabiltzaileAtazakMB.kodea==2}">
    Sesioa: #{erabiltzaileAtazakMB.sesioIzena}
    <h:dataTable value="#{erabiltzailea3ViewMB.sesiokoEragiketakLortuDB(erabiltzaileAtazakMB.sesioIzena)}" var="e" border="2">
      <h:column>
        <f:facet name="header">ERAGIKETAK</f:facet>
        #{e.eragigai1}#{e.eragilea}#{e.eragigai2}
      </h:column>
      <h:column>
        <f:facet name="header">EMAITZAK</f:facet>
        #{e.emaitza}
      </h:column>
    </h:dataTable>
    <p>
      <h:form>
        <h:commandButton action="erabiltzailea1" actionListener="#{erabiltzaileAtazakMB.resetSession()}"
          id="submit" value="SESIO BERRIA HASI" />
      </h:form>
    </p>
  </h:panelGroup>
</ui:define>
</ui:composition>
</html>
```

2.- JAVA EE

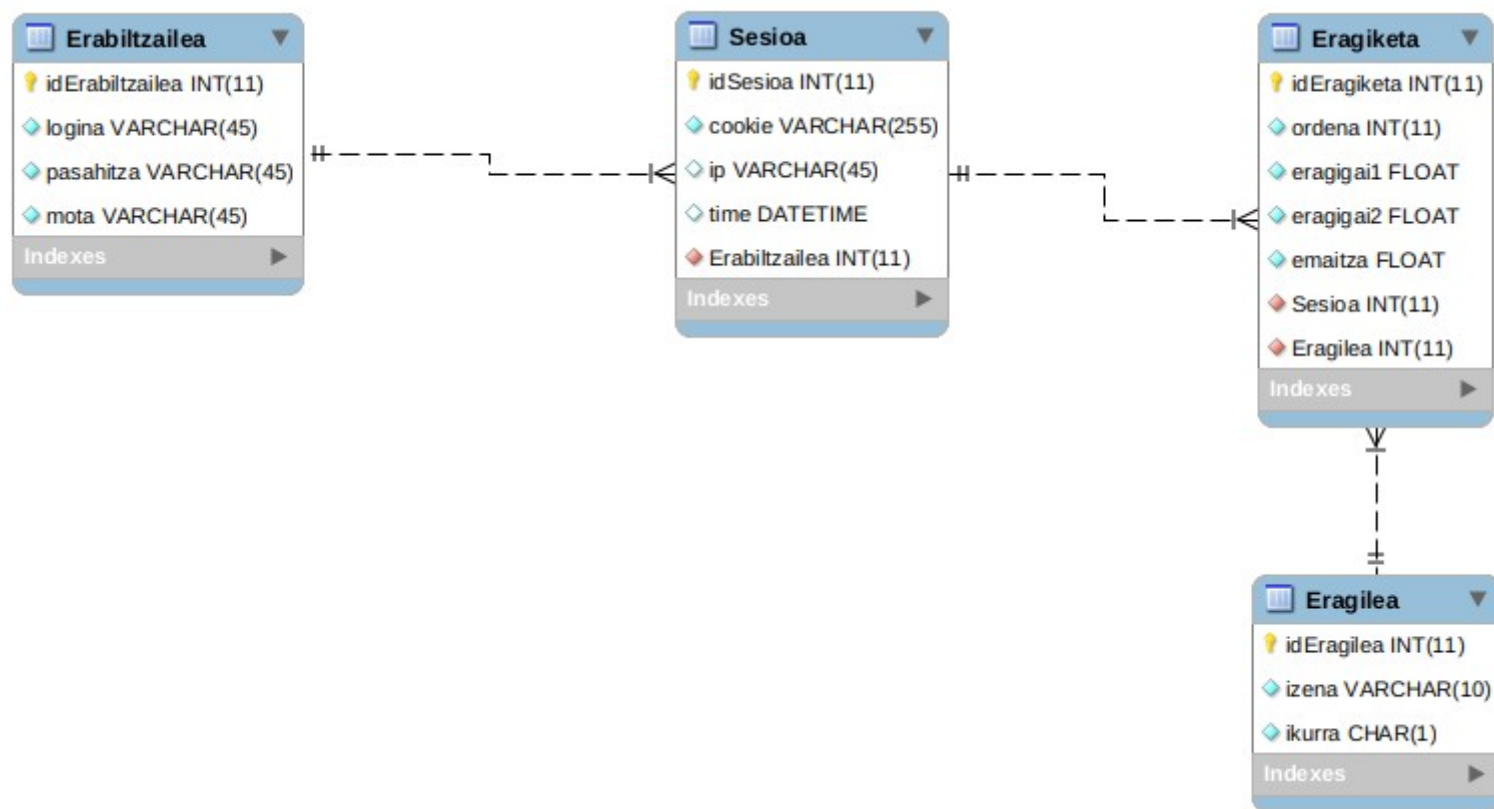
2.2.- Bussiness tier

- JSF+EJB+JPA Aplikazioak
 - Programatu beharreko blokeak
 - Datu-basea
 - JavaEE aplikazioa JSF+EJB+JPA: Eclipse
 - SW maila bakoitzeko osagaiak
 - DL
 - persistence.xml
 - Java klaseak: entitateak (JB),...
 - BL: Java klaseak
 - EJBak
 - Web Zerbitzuak
 - ...
 - PL
 - Java klaseak: MBak,...
 - .xhtml fitxategiak
 - .html, .css,...
 - Proiektuko konfigurazio-fitxategiak
 - web.xml
 - ...

2.- JAVA EE

2.2.- Bussiness tier

- JSF+EJB+JP 2. ADIBIDEA: Aplikazio aritmetikoa JSF, EJB eta ERLAZIONATUTAKO JPEntities-ekin
 - Datu-basea: Datuen eta erlazioen eredua



2.- JAVA EE

2.2.- Bussiness tier

- JSF+EJB+JP 2. ADIBIDEA: Aplikazio aritmetikoa JSF, EJB eta ERLAZIONATUTAKO JPEntities-ekin
 - Datu-basea: Datuen eta erlazioen eredua

```
CREATE TABLE IF NOT EXISTS `ZTA_M8-xAD`.`Erabiltzailea` (  
  `idErabiltzailea` INT(11) NOT NULL AUTO_INCREMENT,  
  `logina` VARCHAR(45) NOT NULL,  
  `pasahitza` VARCHAR(45) NOT NULL,  
  `mota` VARCHAR(45) NOT NULL,  
  PRIMARY KEY (`idErabiltzailea`),  
  UNIQUE INDEX `logina_UNIQUE` (`logina` ASC))
```

Erabiltzailea taula sortu eta betetzeko sententziak

```
ENGINE = InnoDB
```

```
DEFAULT CHARACTER SET = latin1;
```

```
START TRANSACTION;
```

```
USE `ZTA_T08-xAD`;
```

```
INSERT INTO `ZTA_M8-xAD`.`Erabiltzailea` (`idErabiltzailea`, `logina`, `pasahitza`, `mota`) VALUES (1, 'A', 'A', 'erabiltzailea');
```

```
INSERT INTO `ZTA_M8-xAD`.`Erabiltzailea` (`idErabiltzailea`, `logina`, `pasahitza`, `mota`) VALUES (2, 'B', 'B', 'erabiltzailea');
```

```
INSERT INTO `ZTA_M8-xAD`.`Erabiltzailea` (`idErabiltzailea`, `logina`, `pasahitza`, `mota`) VALUES (3, 'C', 'C', 'erabiltzailea');
```

```
COMMIT;
```

```
CREATE TABLE IF NOT EXISTS `ZTA_M8-xAD`.`Eragilea` (  
  `idEragilea` INT(11) NOT NULL AUTO_INCREMENT,  
  `izena` VARCHAR(10) NOT NULL,  
  `ikurra` CHAR(1) NOT NULL,  
  PRIMARY KEY (`idEragilea`))
```

Eragilea taula sortu eta betetzeko sententziak

```
ENGINE = InnoDB
```

```
DEFAULT CHARACTER SET = latin1;
```

```
START TRANSACTION;
```

```
USE `ZTA_T08-xAD`;
```

```
INSERT INTO `ZTA_M8-xAD`.`Eragilea` (`idEragilea`, `izena`, `ikurra`) VALUES (1, 'BATUKETA', '+');
```

```
INSERT INTO `ZTA_M8-xAD`.`Eragilea` (`idEragilea`, `izena`, `ikurra`) VALUES (2, 'KENKETA', '-');
```

```
INSERT INTO `ZTA_M8-xAD`.`Eragilea` (`idEragilea`, `izena`, `ikurra`) VALUES (3, 'BIDEREKETA', '*');
```

```
INSERT INTO `ZTA_M8-xAD`.`Eragilea` (`idEragilea`, `izena`, `ikurra`) VALUES (4, 'ZATIKETA', '/');
```

```
COMMIT;
```

2.- JAVA EE

2.2.- Bussiness tier

- JSF+EJB+JP 2. ADIBIDEA: Aplikazio aritmetikoa JSF, EJB eta ERLAZIONATUTAKO JPEntities-ekin
 - Datu-basea: Datuen eta erlazioen eredua

```
CREATE TABLE IF NOT EXISTS `ZTA_M8-xAD`.`Eragiketa` (  
  `idEragiketa` INT(11) NOT NULL AUTO_INCREMENT,  
  `ordena` INT(11) NOT NULL,  
  `eragigai1` FLOAT NOT NULL,  
  `eragigai2` FLOAT NOT NULL,  
  `emaitza` FLOAT NOT NULL,  
  `Sesioa` INT(11) NOT NULL,  
  `Eragilea` INT(11) NOT NULL,  
  PRIMARY KEY (`idEragiketa`),  
  INDEX `fk_Eragiketa_Sesioa` (`Sesioa` ASC),  
  INDEX `fk_Eragiketa_Eragilea1_idx` (`Eragilea` ASC),  
  CONSTRAINT `fk_Eragiketa_Eragilea`  
    FOREIGN KEY (`Eragilea`)  
    REFERENCES `ZTA_M8-xAD`.`Eragilea` (`idEragilea`)  
    ON DELETE CASCADE  
    ON UPDATE CASCADE,  
  CONSTRAINT `fk_Eragiketa_Sesioa`  
    FOREIGN KEY (`Sesioa`)  
    REFERENCES `ZTA_M8-xAD`.`Sesioa` (`idSesioa`)  
    ON DELETE CASCADE  
    ON UPDATE CASCADE)  
ENGINE = InnoDB  
DEFAULT CHARACTER SET = latin1;      Eragiketa taula sortzeko sententziak
```

```
CREATE TABLE IF NOT EXISTS `ZTA_M8-xAD`.`Sesioa` (  
  `idSesioa` INT(11) NOT NULL AUTO_INCREMENT,  
  `cookie` VARCHAR(255) NOT NULL,  
  `ip` VARCHAR(45) NULL DEFAULT NULL,  
  `time` DATETIME NULL DEFAULT NULL,  
  `Erabiltzailea` INT(11) NOT NULL,  
  PRIMARY KEY (`idSesioa`),  
  INDEX `fk_Sesioa_Erabiltzailea` (`Erabiltzailea` ASC),  
  CONSTRAINT `fk_Sesioa_Erabiltzailea`  
    FOREIGN KEY (`Erabiltzailea`)  
    REFERENCES `ZTA_M8-xAD`.`Erabiltzailea` (`idErabiltzailea`)  
    ON DELETE CASCADE  
    ON UPDATE CASCADE)  
ENGINE = InnoDB  
DEFAULT CHARACTER SET = latin1;      Sesioa taula sortzeko sententziak
```


2.- JAVA EE

2.2.- Bussiness tier

- JSF+EJB+JP 2. ADIBIDEA: Aplikazio aritmetikoa JSF, EJB eta ERLAZIONATUTAKO JPEntities-ekin
 - Wildfly konfigurazioa: *WildflyHome/standalone/configuration/standalone.xml*
 - Datu-basea

```
<datasource jndi-name="java:jboss/datasources/ZTA_M8-xAD" pool-name="ZTA_M8-xAD" enabled="true">
  <connection-url>jdbc:mysql://localhost:3306/ZTA_M8-xAD?sslMode=DISABLED</connection-url>
  <driver>com.mysql</driver>
  <security>
    <user-name>dbErabiltzailea</user-name>
    <password>pasahitza</password>
  </security>
</datasource>
```

- Segurtasun-domeinua



















```
<security-domain name="DBM8" cache-type="default">
  <authentication>
    <login-module code="Database" flag="required">
      <module-option name="password-stacking" value="useFirstPass"/>
      <module-option name="dsJndiName" value="java:jboss/datasources/ZTA_M8-xAD"/>
      <module-option name="principalsQuery" value="SELECT pasahitza FROM Erabiltzailea WHERE logina=?"/>
      <module-option name="rolesQuery" value="SELECT mota, 'Roles' FROM Erabiltzailea WHERE logina=?"/>
    </login-module>
  </authentication>
</security-domain>
```

2.- JAVA EE














2.2.- Bussiness tier

- JSF+EJB+JP 2. ADIBIDEA: Aplikazio aritmetikoa JSF, EJB eta ERLAZIONATUTAKO JPEntities-ekin

– JAVA osagaiak: **java**

- ▼  **bl**
 - ▶  ErabSesiokoEJB.java
 - ▶  EragiketaJB.java
 - ▶  KalkulagailuaPOJO.java
 - ▶  OrokorraEJB.java
- ▼  **dl**
 - ▶  ErabiltzaileaE.java
 - ▶  EragiketaE.java
 - ▶  EragileaE.java
 - ▶  SesioaE.java
- ▼  **pl**
 - ▶  Erabiltzailea1ViewMB.java
 - ▶  Erabiltzailea3ViewMB.java
 - ▶  ErabiltzaileAtazakMB.java
 - ▶  EragiketaFormMB.java
 - ▶  EuskarazPOJO.java
- ▼  **META-INF**
 - ▶  persistence.xml

– HTML osagaiak: **webapp**

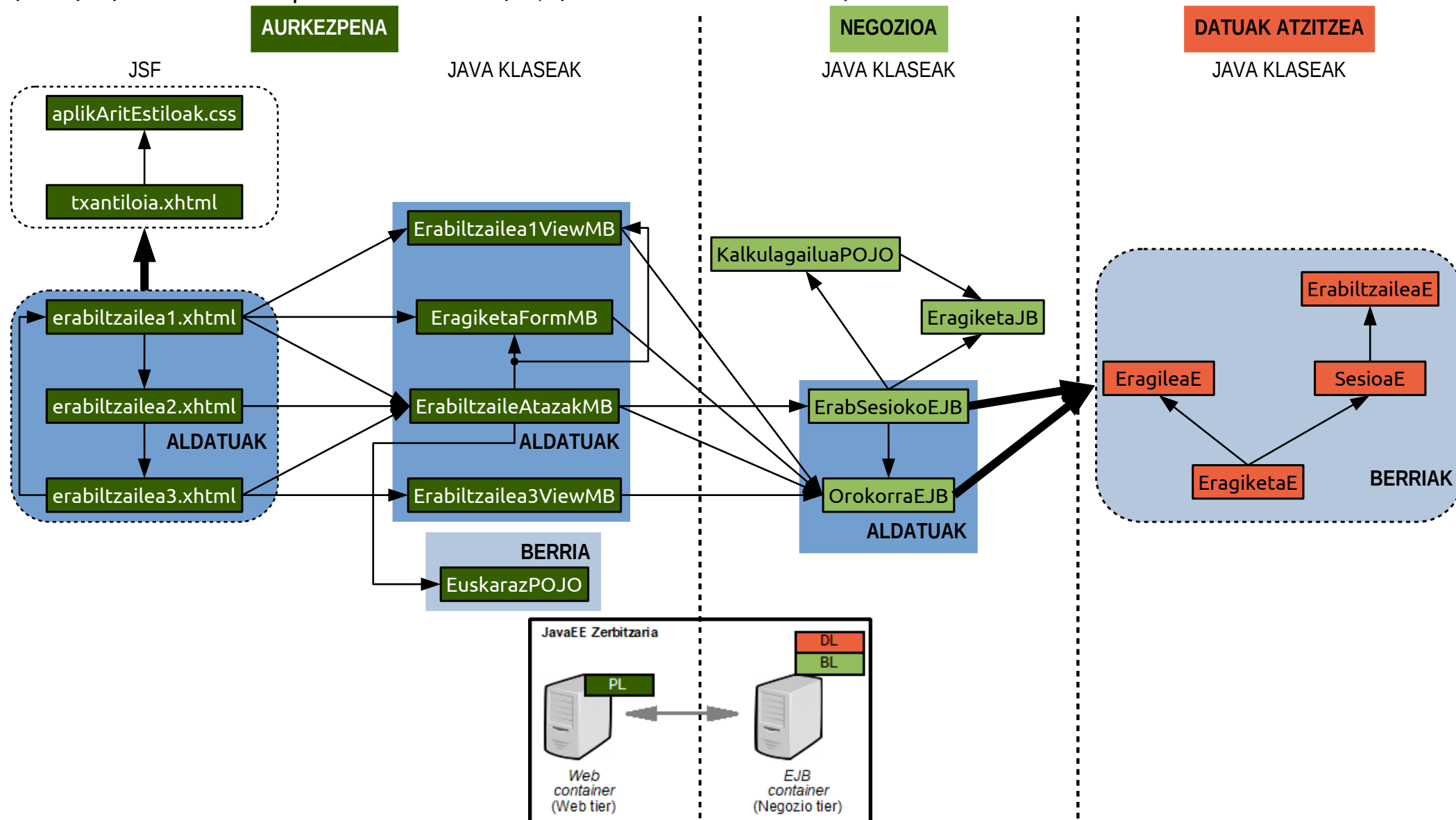
- ▼  **EDUKIAK**
 - ▶  aplikAritEstiloak.css
 - ▶  erabiltzailea1.xhtml
 - ▶  erabiltzailea2.xhtml
 - ▶  erabiltzailea3.xhtml
 - ▶  txantiloia.xhtml
- ▶  **META-INF**
- ▼  **WEB-INF**
 - ▶  **lib**
 - ▶  beans.xml
 - ▶  faces-config.xml
 - ▶  jboss-web.xml
 - ▶  web.xml



2.- JAVA EE

2.2.- Bussiness tier

- JSF+EJB+JP 2. ADIBIDEA: Aplikazio aritmetikoa JSF, EJB eta ERLAZIONATUTAKO JPEntities-ekin



2.- JAVA EE

2.2.- Bussiness tier

- JSF+EJB+JP 2. ADIBIDEA: Aplikazio aritmetikoa JSF, EJB eta ERLAZIONATUTAKO JPEntities-ekin

Entity: dl.EragiketaE.java

```
package dl;

//import lerroak

@Entity
@Table(name="Eragiketa")
@NamedQueries({
    @NamedQuery(name="EragiketaE.findAll", query="SELECT e FROM EragiketaE e"),
    @NamedQuery(name="EragiketaE.findAllSesioa", query="SELECT e FROM EragiketaE e WHERE e.sesioaE.idSesioa = :idSesioa ORDER BY e.ordena")
})
public class EragiketaE implements Serializable {
    private static final long serialVersionUID = 1L;

    @Id
    @GeneratedValue(strategy=GenerationType.IDENTITY)
    private int idEragiketa;

    private float emaitza;
    private float eragigai1;
    private float eragigai2;
    private int ordena;

    //uni-directional many-to-one association to EragileaE
    @ManyToOne
    @JoinColumn(name="Eragilea")
    private EragileaE eragileaE;

    //uni-directional many-to-one association to SesioaE
    @ManyToOne
    @JoinColumn(name="Sesioa")
    private SesioaE sesioaE;

    //PARAMETRORIK GABEKO ERAIKITZAILEA
    //ATRIBUTU GUZTIETARAKO PARAMETROAK DITUEN ERAIKITZAILEA

    //ATRIBUTU GUZTIEN getter eta setter METODOAK

    public String laburdura() {...} //AURREKO ADIBIDEKO KODE BERDINA
}
```



2.- JAVA EE

2.2.- Bussiness tier

- JSF+EJB+JP 2. ADIBIDEA: Aplikazio aritmetikoa JSF, EJB eta ERLAZIONATUTAKO JPEntities-ekin

package dl;

Entity: dl.SesioaE.java

```
import java.io.Serializable;
import javax.persistence.*;
import java.util.Date;

@Entity
@Table(name="Sesioa")
@NamedQuery(name="SesioaE.findAll", query="SELECT s FROM SesioaE s")
public class SesioaE implements Serializable {
    private static final long serialVersionUID = 1L;

    @Id
    @GeneratedValue(strategy=GenerationType.IDENTITY)
    private int idSesioa;

    private String cookie;

    private String ip;

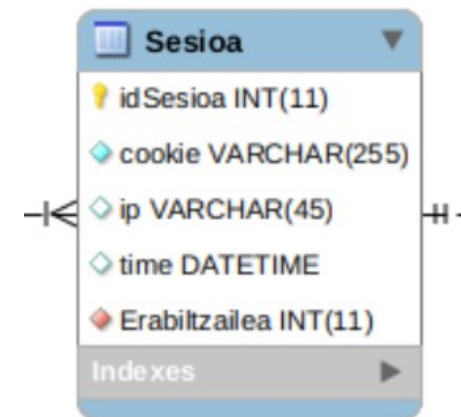
    @Temporal(TemporalType.TIMESTAMP)
    private Date time;

    //uni-directional many-to-one association to ErabiltzaileaE
    @ManyToOne
    @JoinColumn(name="Erabiltzailea")
    private ErabiltzaileaE erabiltzaileaE;

    //PARAMETRORIK GABEKO ERAIKITZAILEA

    //ATRIBUTU GUZTIETARAKO PARAMETROAK DITUEN ERAIKITZAILEA

    //ATRIBUTU GUZTIEN getter eta setter METODOAK
}
```



2.- JAVA EE

2.2.- Bussiness tier

- JSF+EJB+JP 2. ADIBIDEA: Aplikazio aritmetikoa JSF, EJB eta ERLAZIONATUTAKO JPEntities-ekin

package dl;

Entity: dl.EragileaE.java

```
import java.io.Serializable;
import javax.persistence.*;

@Entity
@Table(name="Eragilea")
@NamedQuery(name="EragileaE.findAll", query="SELECT e FROM EragileaE e")
public class EragileaE implements Serializable {
    private static final long serialVersionUID = 1L;

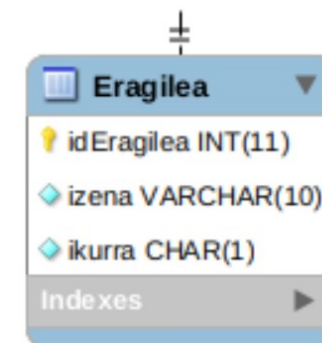
    @Id
    @GeneratedValue(strategy=GenerationType.IDENTITY)
    private int idEragilea;

    private char ikurra;

    private String izena;

    public EragileaE() {
    }

    //ATRIBUTU GUZTIEN getter eta setter METODOAK
}
```



2.- JAVA EE

2.2.- Bussiness tier

- JSF+EJB+JP 2. ADIBIDEA: Aplikazio aritmetikoa JSF, EJB eta ERLAZIONATUTAKO JPEntities-ekin

```
package dl;

//import lerroak

@Entity
@Table(name="Erabiltzailea")
@NamedQueries({
    @NamedQuery(name="ErabiltzaileaE.findAll", query="SELECT e FROM ErabiltzaileaE e"),
    @NamedQuery(name="ErabiltzaileaE.findErabiltzailea", query="SELECT e FROM ErabiltzaileaE e WHERE e.logina = :logina")
})
public class ErabiltzaileaE implements Serializable {
    private static final long serialVersionUID = 1L;

    @Id
    @GeneratedValue(strategy=GenerationType.IDENTITY)
    private int idErabiltzailea;

    private String logina;

    private String mota;

    private String pasahitza;

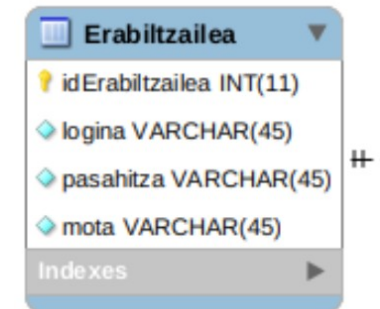
    public ErabiltzaileaE() {

    }

    public ErabiltzaileaE(int idErabiltzailea, String logina, String mota, String pasahitza) {
        this.idErabiltzailea = idErabiltzailea;
        this.logina = logina;
        this.mota = mota;
        this.pasahitza = pasahitza;
    }

    //ATRIBUTU GUZTIEN getter eta setter METODOAK
}
```

Entity: dl.ErabiltzaileaE.java



2.- JAVA EE

2.2.- Bussiness tier

- JSF+EJB+JP 2. ADIBIDEA: Aplikazio aritmetikoa JSF, EJB eta ERLAZIONATUTAKO JPEntities-ekin

EJB: bl.ErabSesiokoEJB.java

```
package bl;

//import lerroak

@SecurityDomain("SecurityDomainIzena")

@DenyAll
@Stateful
@LocalBean
public class ErabSesiokoEJB {
    private KalkulagailuaPOJO kPOJO=new KalkulagailuaPOJO();
    private List<EragiketaE> eragiketakE=new ArrayList<EragiketaE>();

    @EJB private OrokorraEJB oEJB;

    @RolesAllowed("erabiltzailea")
    public List<EragiketaE> getEragiketakE() {
        return eragiketakE;
    }

    @RolesAllowed("erabiltzailea")
    public int eragiketaEginZerrendaratu(EragiketaE eragiketaE) {
        int kodea=1;

        EragiketaJB eragiketaJB=new EragiketaJB(eragiketaE.getEragigai1(), ""+eragiketaE.getEragileakE().getIkurra(),
                                                eragiketaE.getEragigai2(), 0.0f);
        kPOJO.kalkulatu(eragiketaJB);

        eragiketaE.setEmaitza(eragiketaJB.getEmaitza());
        eragiketakE.add(eragiketaE);

        return kodea;
    }

    @RolesAllowed("erabiltzailea")
    public int zerrendaBiltegiratu(int idSesioa) {
        return oEJB.eragiketakSartuDB(eragiketakE, idSesioa);
    }
}
```


2.- JAVA EE

2.2.- Bussiness tier

- JSF+EJB+JP 2. ADIBIDEA: Aplikazio aritmetikoa JSF, EJB eta ERLAZIONATUTAKO JPEntities-ekin

EJB: bl.OrokorraEJB.java

```
package bl;

//import lerroak

@SecurityDomain("SecurityDomainIzena")

@DenyAll
@Stateless
@LocalBean
public class OrokorraEJB {

    @PersistenceContext private EntityManager em;

    @Resource EJBContext ejbContext;

    @RolesAllowed("erabiltzailea")
    @SuppressWarnings("unchecked")
    public List<EragileaE> eragileakLortuDB() {
        List<EragileaE> eragileakDB = (List<EragileaE>)em.createNamedQuery("EragileaE.findAll").getResultList();

        return eragileakDB;
    }

    @RolesAllowed("erabiltzailea")
    public void autSesioaSartuDB(SesioaE sesioaE) {
        String autentikatutakoa=ejbContext.getCallerPrincipal().getName();
        ErabiltzaileaE erabiltzaileaDB=(ErabiltzaileaE)em.createNamedQuery("ErabiltzaileaE.findErabiltzailea")
            .setParameter("logina",autentikatutakoa).getSingleResult();

        sesioaE.setErabiltzaileaE(erabiltzaileaDB);
        em.persist(sesioaE);
    }

    //HEMENGOKO KODE GUZTIA HURRENGO ORRIALDEAN DAGO
}
```

2.- JAVA EE

2.2.- Bussiness tier

- JSF+EJB+JP 2. ADIBIDEA: Aplikazio aritmetikoa JSF, EJB eta ERLAZIONATUTAKO JPEntities-ekin

//HEMENGU KODE GUZTIA AURREKO ORRIALDEAN DAGO

EJB: bl.OrokorraEJB.java

```
@RolesAllowed("erabiltzailea")
public int eragiketakSartuDB(List<EragiketaE> eragiketakE, int idSesioa) {
    int kodea=2;

    SesioaE sesioaDB=em.find(SesioaE.class, idSesioa);
    if(sesioaDB==null)
        kodea=3;
    else {
        EragiketaE eragiketa;
        EragileaE eragileaDB;
        for(int i=0;i<eragiketakE.size();i++) {
            eragiketa = eragiketakE.get(i);
            eragileaDB=em.find(EragileaE.class, eragiketa.getEragileaE().getIdEragilea());
            if(eragileaDB==null)
                kodea=4;
            else {
                eragiketa.setOrdena(i+1);
                eragiketa.setSesioaE(sesioaDB);
                eragiketa.setEragileaE(eragileaDB);

                em.persist(eragiketa);
            }
        }
    }

    return kodea;
}

@RolesAllowed("erabiltzailea")
@SuppressWarnings("unchecked")
public List<EragiketaE> sesiokoEragiketakLortuDB(int idSesioa) {
    List<EragiketaE> eragiketakDB = (List<EragiketaE>)em.createNamedQuery("EragiketaE.findAllSesioa")
        .setParameter("idSesioa",idSesioa).getResultList();

    return eragiketakDB;
}
```

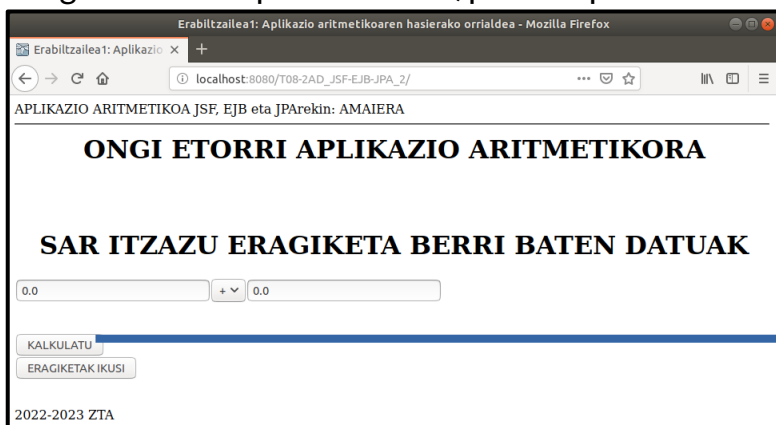
2.- JAVA EE

2.2.- Bussiness tier

- JSF+EJB+JP 2. ADIBIDEA: Aplikazio aritmetikoa JSF, EJB eta ERLAZIONATUTAKO JPEntities-ekin
 - JavaEE aplikazioa: VIEW

DBan sesioa sartu: SessionScoped

Eragileen menua prestatu DBtik, pantaila prestatzeko: ViewScoped



Eragiketa berria eskaera bakoitzean:
RequestScoped
Eragiketa egin+zerrendaratu:
SessionScoped



Sesioko zerrenda bistaratu: SessionScoped



DBan sesioko zerrenda sartu+bistaratzeko gogoratu: SessionScoped
DBtik sesio jakin bateko eragiketak lortu pantaila prestatzeko: ViewScoped

2.- JAVA EE

2.2.- Bussiness tier

- JSF+EJB+JP 2. ADIBIDEA: Aplikazio aritmetikoa JSF, EJB eta ERLAZIONATUTAKO JPEntities-ekin

```
package pl;  
  
//import lerroak  
  
@Named  
@ViewScoped  
public class Erabiltzailea1ViewMB implements Serializable {  
    private static final long serialVersionUID = 1L;  
  
    private List<Erabileak> erabileakDB;  
  
    @EJB private OrokorraEJB oEJB;  
  
    public void resetView() {  
        erabileakDB=null;  
        return;  
    }  
  
    public List<Erabileak> erabileakLortuDB() {  
        if(erabileakDB==null) {  
            erabileakDB=oEJB.erabileakLortuDB();  
        }  
  
        return erabileakDB;  
    }  
}
```

MB: pl.Erabiltzailea1ViewMB.java

2.- JAVA EE

2.2.- Bussiness tier

- JSF+EJB+JP 2. ADIBIDEA: Aplikazio aritmetikoa JSF, EJB eta ERLAZIONATUTAKO JPEntities-ekin

```
package pl;  
  
//import lerroak  
  
@Named  
@RequestScoped  
public class EragiketaFormMB {  
    private float er1;  
    private int valueMenuEragilea;  
    private float er2;  
  
    //ATRIBUTU GUZTIEN getter eta setter metodoak  
  
    public void resetForm() {  
        er1=0.0f;  
        valueEragileMenu=0;  
        er2=0.0f;  
    }  
}
```

MB: pl.EragiketaFormMB.java

2.- JAVA EE

2.2.- Bussiness tier

- JSF+EJB+JP 2. ADIBIDEA: Aplikazio aritmetikoa JSF, EJB eta ERLAZIONATUTAKO JPEntities-ekin

```
package pl;

//import lerroak

@Named
@SessionScoped
public class ErabiltzaileAtazakMB implements Serializable {
    private static final long serialVersionUID = 1L;

    private SesioaE sesioaDB;
    private int kodea;
    private String mezua;

    @EJB private ErabSesiokoEJB eEJB;
    @EJB private OrokorraEJB oEJB;

    @PostConstruct
    private void sesioaSartuDB() {
        String userIP=((HttpServletRequest)FacesContext.getCurrentInstance().getExternalContext().getRequest()).getRemoteAddr();

        HttpSession httpSession=(HttpSession)FacesContext.getCurrentInstance().getExternalContext().getSession(true);
        Date time=new Date(httpSession.getCreationTime());
        String cookie=httpSession.getId();

        sesioaDB=new SesioaE(0,cookie, userIP, time, null);
        oEJB.autSesioaSartuDB(sesioaDB);
    }

    public SesioaE getSesioaDB() {
        return sesioaDB;
    }

    public String getKodea() {...} //AURREKO ADIBIDEKO KODE BERDINA
    public String getMezua() {...} //AURREKO ADIBIDEKO KODE BERDINA

    //HEMENGO KODE GUZTIA HURRENGO ORRIALDEAN DAGO
}
```

MB: pl.ErabiltzaileAtazakMB.java

2.- JAVA EE

2.2.- Bussiness tier

- JSF+EJB+JP 2. ADIBIDEA: Aplikazio aritmetikoa JSF, EJB eta ERLAZIONATUTAKO JPEntities-ekin

//HEMENGOKODEGUZTIAAURREKORRRIALDEANDAGO

MB: pl.ErabiltzaileAtazakMB.java

```
public void resetSession() {
    sesioaDB="";
    mezua="";
    eragiketakJB=null;
    FacesContext.getCurrentInstance().getExternalContext().invalidateSession();
}

public void eragiketaEginZerrendaratu(EragiketaFormMB eragiketaFormMB, Erabiltzailea1ViewMB erabiltzailea1ViewMB) {
    Eragileak eragileakE=erabiltzailea1ViewMB.eragileakLortuDB().get(eragiketaFormMB.getValueEragileMenu());

    EragiketaE eragiketaE=new EragiketaE(0, 0, eragiketaFormMB.getEr1(), eragiketaFormMB.getEragileakE(),
                                         eragiketaFormMB.getEr2(), 0.0f,null);

    kodea=eEJB.eragiketaEginZerrendaratu(eragiketaE);
    mezua=eragileakE.getIzena()+" "+eE.laburdura();

    eragiketaFormMB.resetForm();
    erabiltzailea1ViewMB.resetView();
}

public List<EragiketaE> eragiketakLortuEJB() {
    return eEJB.getEragiketakE();
}

public void zerrendaBiltegitatu() {
    kodea=eEJB.zerrendaBiltegitatu(sesioaEDB.getIdSesio());
    mezua=EuskarazPOJO.erabiltzaileMezuak[kodea];
}
}
```

2.- JAVA EE

2.2.- Bussiness tier

- JSF+EJB+JP 2. ADIBIDEA: Aplikazio aritmetikoa JSF, EJB eta ERLAZIONATUTAKO JPEntities-ekin

```
package pl;

//import lerroak

@Named
@ViewScoped
public class Erabiltzailea3ViewMB implements Serializable {
    private static final long serialVersionUID = 1L;

    private List<EragiketaE> eragiketakDB;

    @EJB private OrokorraEJB oEJB;

    public void resetView() {
        eragiketakDB=null;
        return;
    }

    public List<EragiketaE> sesiokoEragiketakLortuDB(int idSesioa) {
        if(eragiketakDB==null)
            eragiketakDB=oEJB.sesiokoEragiketakLortuDB(idSesioa);

        return eragiketakDB;
    }
}
```

MB: pl.Erabiltzailea3ViewMB.java

```
package pl;

public abstract class EuskarazPOJO {
    public final static String erabiltzaileMezuak[] = {
        "MEZURIK EZ",//0
        "ZERRENDATUTAKO ERAGIKETA",//1
        "ZURE SESIOKO DATU HAUEK DBan GORDE DIRA",//2
        "ERROREA ZURE SESIOA DBan GORDETZEAN: Sesioa jadanik ez dago DBan",//3
        "ERROREA ZURE SESIOA DBan GORDETZEAN: Eragileren bat ez dago DBan eta ez da sesioa gorde"//4
    };
}
```

POJO: pl.EuskarazPOJO.java

2.- JAVA EE

2.2.- Bussiness tier

- JSF+EJB+JP 2. ADIBIDEA: Aplikazio aritmetikoa JSF, EJB eta ERLAZIONATUTAKO JPEntities-ekin

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml"
      xmlns:h="http://java.sun.com/jsf/html"
      xmlns:f="http://java.sun.com/jsf/core"
      xmlns:ui="http://java.sun.com/jsf/facelets">
<ui:composition template="/txantiloia.xhtml">
<ui:define name="headTitle">Erabiltzailea1: Aplikazio aritmetikoaren hasierako orrialdea</ui:define>
<ui:define name="bodyBurua">APLIKAZIO ARITMETIKOA JSFekin: HASIERA</ui:define>
<ui:define name="bodyEdukia">
  <h:panelGroup rendered="#{erabiltzaileAtazakMB.kodea==0}">
    <h1>ONGI ETORRI APLIKAZIO ARITMETIKORA</h1>
  </h:panelGroup>
  <h:panelGroup rendered="#{erabiltzaileAtazakMB.kodea!=0}">
    <h1>BIDALITAKO ERAGIKETAREN EMAITZA</h1>
    #{erabiltzaileAtazakMB.mezua}
  </h:panelGroup>
  <br /><br />
  <h:form>
    <h1>SAR ITZAZU ERAGIKETA BERRI BATEN DATUAK</h1>
    <h:inputText id="er1" value="#{eragiketaFormMB.er1}" required="true" requiredMessage="ERROREA: 1. eragigaia falta da"/>
    <h:selectOneMenu id="er" value="#{eragiketaFormMB.valueMenuEragilea}">
      <f:selectItems value="#{erabiltzailea1ViewMB.eragileakLortuDB()}" var="eragilea"
                    itemValue="#{erabiltzailea1ViewMB.eragileakLortuDB().indexOf(eragilea)}" itemLabel="#{eragilea.ikurra}"/>
    </h:selectOneMenu>
    <h:inputText id="er2" value="#{eragiketaFormMB.er2}" required="true" requiredMessage="ERROREA: 2. eragigaia falta da" /><br />
    <h:message for="er1" style="color:blue" /><br />
    <h:message for="er2" style="color:blue" /><br />
    <h:commandButton actionListener="#{erabiltzaileAtazakMB.eragiketaEginZerrendaratu(eragiketaFormMB, erabiltzailea1ViewMB)}"
                    id="submit" value="KALKULATU" />
  </h:form>
  <h:form>
    <h:commandButton action="erabiltzailea2" id="submit" value="ERAGIKETAK IKUSI" />
  </h:form>
</ui:define>
</ui:composition>
</html>
```

XHTML: erabiltzailea1.xhtml

2.- JAVA EE

2.2.- Bussiness tier

- JSF+EJB+JP 2. ADIBIDEA: Aplikazio aritmetikoa JSF, EJB eta ERLAZIONATUTAKO JPEntities-ekin

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml"
      xmlns:h="http://java.sun.com/jsf/html"
      xmlns:f="http://java.sun.com/jsf/core"
      xmlns:ui="http://java.sun.com/jsf/facelets">
<ui:composition template="/txantiloia.xhtml">
<ui:define name="headTitle">Erabiltzailea2: Aplikazio aritmetikoaren sesio-laburpenaren orrialdea</ui:define>
<ui:define name="bodyBurua">APLIKAZIO ARITMETIKOA JSF, EJB eta JPAREKIN: SESIOAREN LABURPENA</ui:define>
<ui:define name="bodyEdukia">
  <h1>ZURE SESIOAN BILDUTAKO DATUAK</h1>
  Eragiketa kopurua: #{erabiltzaileAtazakMB.eragiketakLortuEJB().size()}
  <h:dataTable value="#{erabiltzaileAtazakMB.eragiketakLortuEJB()}" var="eragiketa" border="2">
    <h:column>
      <f:facet name="header">MOTA</f:facet>
      #{eragiketa.eragileaE.izena}
    </h:column>
    <h:column>
      <f:facet name="header">ERAGIKETAK</f:facet>
      #{eragiketa.eragigai1}#{eragiketa.eragileaE.ikurra}#{eragiketa.eragigai2}
    </h:column>
    <h:column>
      <f:facet name="header">EMAITZAK</f:facet>
      #{eragiketa.emaitza}
    </h:column>
  </h:dataTable>
  <p>
    <h:form>
      <h:commandButton action="erabiltzailea3" actionListener="#{erabiltzaileAtazakMB.zerrendaBiltegitatu()}" id="submit"
        value="SESIOA DBan GRABATU" />
    </h:form>
  </p>
</ui:define>
</ui:composition>
</html>
```

XHTML: erabiltzailea2.xhtml

2.- JAVA EE

2.2.- Bussiness tier

- JSF+EJB+JP 2. ADIBIDEA: Aplikazio aritmetikoa JSF, EJB eta ERLAZIONATUTAKO JPEntities-ekin

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml"
      xmlns:h="http://java.sun.com/jsf/html"
      xmlns:f="http://java.sun.com/jsf/core"
      xmlns:ui="http://java.sun.com/jsf/facelets">
<ui:composition template="/txantiloia.xhtml">
<ui:define name="headTitle">Erabiltzailea3: Aplikazio aritmetikoaren amaierako orrialdea</ui:define>
<ui:define name="bodyBurua">APLIKAZIO ARITMETIKOA JSF, EJB eta JPArekin: AMAIERA</ui:define>
<ui:define name="bodyEdukia">
  <h1>#{erabiltzaileAtazakMB.mezua}</h1>
  Erabiltzailea: #{erabiltzaileAtazakMB.sesioaDB.erabiltzaileaE.logina}<br />
  <h:panelGroup rendered="#{erabiltzaileAtazakMB.kodea==2}">
    Sesioa: #{erabiltzaileAtazakMB.sesioaDB.cookie}
    <h:dataTable value="#{erabiltzailea3ViewMB.sesiokoEragiketakLortuDB(erabiltzaileAtazakMB.sesioaDB.idSesioa)}" var="eragiketa"
                  border="2">
      <h:column>
        <f:facet name="header">MOTA</f:facet>
        #{eragiketa.eragileaE.izena}
      </h:column>
      <h:column>
        <f:facet name="header">ERAGIKETAK</f:facet>
        #{eragiketa.eragigai1}#{eragiketa.eragileaE.ikurra}#{eragiketa.eragigai2}
      </h:column>
      <h:column>
        <f:facet name="header">EMAITZAK</f:facet>
        #{eragiketa.emaitza}
      </h:column>
    </h:dataTable>
  </h:panelGroup>
</p>
<h:form>
  <h:commandButton action="erabiltzailea1" actionListener="#{erabiltzaileAtazakMB.resetSession()}" id="submit" value="SESIO BERRIA
    HASI" />
</h:form>
</p>
</ui:define>
</ui:composition>
</html>
```

XHTML: erabiltzailea3.xhtml

2.- JAVA EE

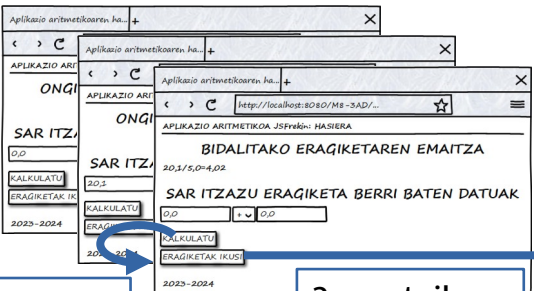
2.2.- Bussiness tier

- JSF+EJB+JP 3. ADIBIDEA: Aplikazio aritmetiko OSOA

1. pantaila: Hasierako egoera

1. pantaila: Erabiltzaileak formularioa beteta

1. pantaila: Emaiza + formularioa



APLIKAZIO ARITMETIKOAREN HASIERA

BIDALITAKO ERAGIKETAREN EMAITZA

20,1/5,0+4,02

SAR ITZAZU ERAGIKETA BERRI BATEN DATUAK

KALKULATU

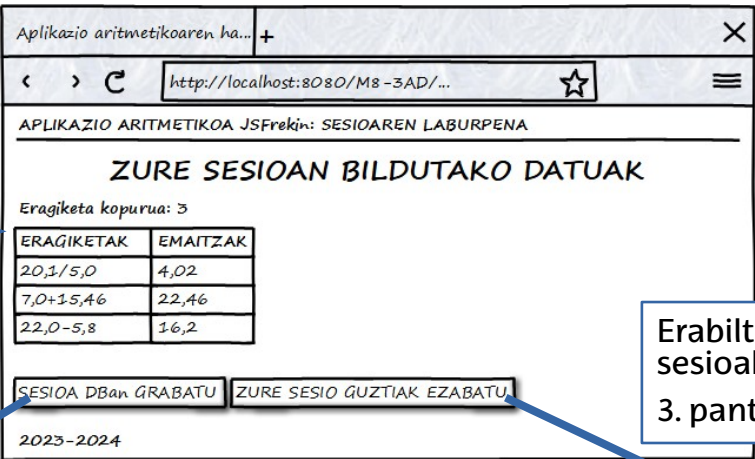
ERAGIKETA IKUSI

2023-2024

BESTE ADIBIDEETAN BEZALA:
Kalkulua egin
Eragiketa sesioko EJBko zerrendan sartu
Emaiza + formularioa bistaratu

2. pantailara

2. pantaila: Sesioren laburpena



APLIKAZIO ARITMETIKOAREN LABURPENAREN HASIERA

APLIKAZIO ARITMETIKOAK JSFekin: SESIOAREN LABURPENAK

ZURE SESIOAN BILDUTAKO DATUAK

Eragiketa kopurua: 3

ERAGIKETAK	EMAITZAK
20,1/5,0	4,02
7,0+15,46	22,46
22,0-5,8	16,2

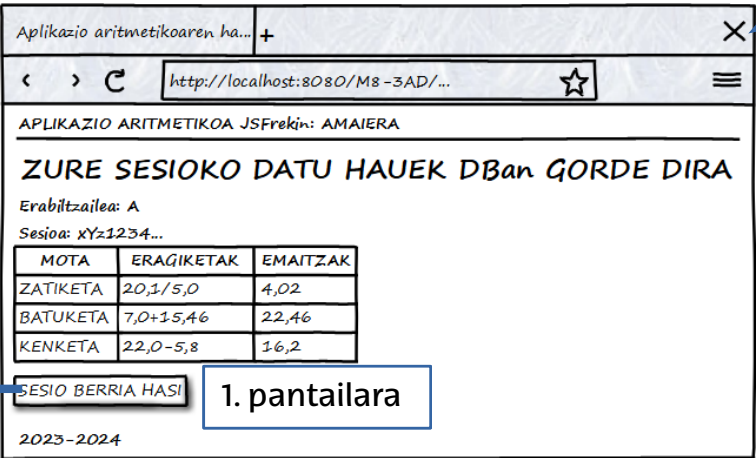
SESIOA DBan GRABATU

ZURE SESIO GUZTIAK EZABATU

2023-2024

Erabiltzailearen
sesioak DBtik ezabatu
3. pantailara

3. pantaila: Sesioko DBan



APLIKAZIO ARITMETIKOAREN LABURPENAREN HASIERA

APLIKAZIO ARITMETIKOAK JSFekin: AMAIERA

ZURE SESIOKO DATU HAUEK DBan GORDE DIRA

Erabiltzailea: A

Sesioa: xYz1234...

MOTA	ERAGIKETAK	EMAITZAK
ZATIKETA	20,1/5,0	4,02
BATUKETA	7,0+15,46	22,46
KENKETA	22,0-5,8	16,2

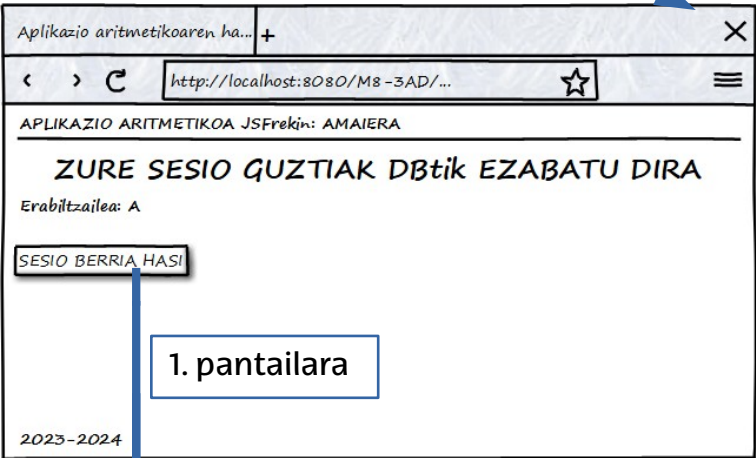
SESIO BERRIA HASI

2023-2024

1. pantailara

EJBko zerrenda
DBan sartu
3. pantailara

3. pantaila: Sesioko guztiak ezabatuta DBtik?



APLIKAZIO ARITMETIKOAREN LABURPENAREN HASIERA

APLIKAZIO ARITMETIKOAK JSFekin: AMAIERA

ZURE SESIO GUZTIAK DBtik EZABATU DIRA

Erabiltzailea: A

SESIO BERRIA HASI

2023-2024

1. pantailara

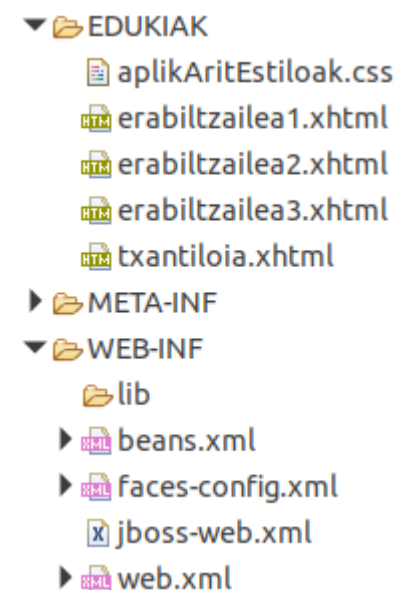
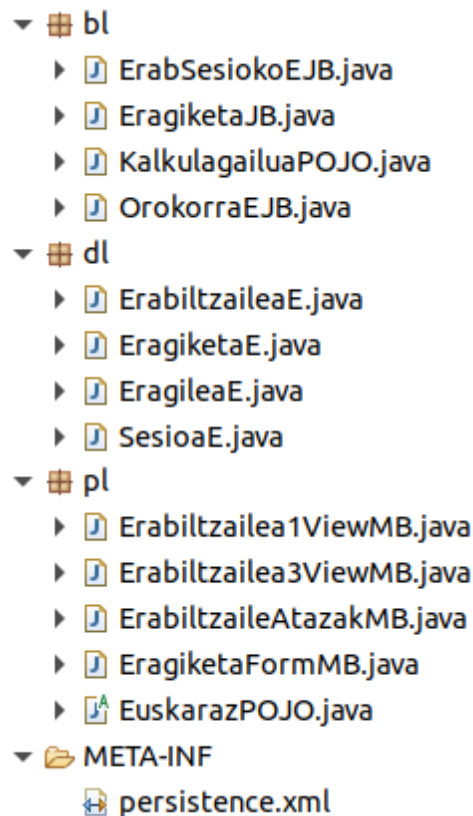
2.- JAVA EE

2.2.- Bussiness tier

- JSF+EJB+JP 3. ADIBIDEA: Aplikazio aritmetiko OSOA

– JAVA osagaiak: **java**

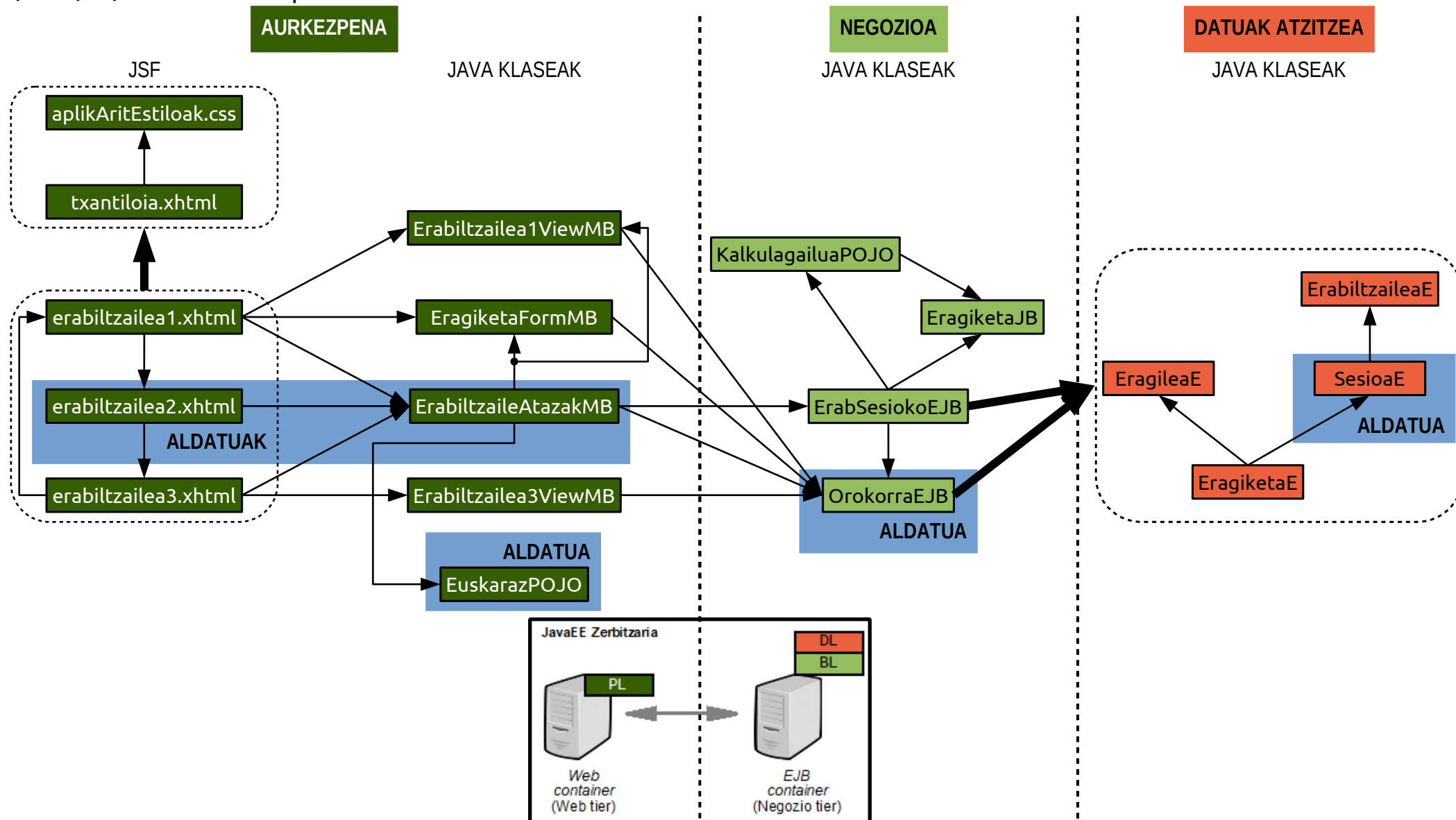
– HTML osagaiak: **webapp**



2.- JAVA EE

2.2.- Bussiness tier

- JSF+EJB+JP 3. ADIBIDEA: Aplikazio aritmetiko OSOA



2.- JAVA EE

2.2.- Bussiness tier

- JSF+EJB+JP 3. ADIBIDEA: Aplikazio aritmetiko OSOA

```
package dl;

//import lerroak

@Entity
@Table(name="Sesioa")
@NamedQueries({
    @NamedQuery(name="SesioaE.findAll", query="SELECT s FROM SesioaE s"),
    @NamedQuery(name="SesioaE.findAllErabiltzailea", query="SELECT s FROM SesioaE s WHERE s.erabiltzaileaE.logina= :logina")
})
public class SesioaE implements Serializable {...} //AURREKO ADIBIDEKO KODE BERDINA
```

Entity: dl.SesioaE.java

2.- JAVA EE

2.2.- Bussiness tier

- JSF+EJB+JP 3. ADIBIDEA: Aplikazio aritmetiko OSOA

EJB: bl.OrokorraEJB.java

```
package bl;

//import lerroak

@SecurityDomain("SecurityDomainIzena")

@DenyAll
@Stateless
@LocalBean
public class OrokorraEJB {

    //AURREKO ADIBIDEKO KODE BERDINA

    @RolesAllowed("erabiltzailea")
    @SuppressWarnings("unchecked")
    public int autSesioakEzabatuDB() {
        int kodea=5;

        String autentikatutakoa=ejbContext.getCallerPrincipal().getName();

        List<SesioaE> sesioakDB=(List<SesioaE>)em.createNamedQuery("SesioaE.findAllErabiltzailea")
            .setParameter("logina",autentikatutakoa).getResultList();

        if(sesioakDB.size()==0)
            kodea=6; //Sesiorik ez dago DBan: EZ DA EZER EZABATU
        else
            for(SesioaE sesioaDB: sesioakDB)
                em.remove(sesioaDB);

        return kodea;
    }
}
```


2.- JAVA EE

2.2.- Bussiness tier

- JSF+EJB+JP 3. ADIBIDEA: Aplikazio aritmetiko OSOA
 - JavaEE aplikazioa: VIEW

Aurreko pantailak ez dira aldatzen

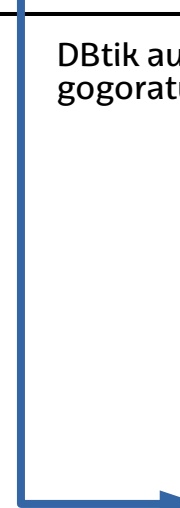


DBan Sesioa eta sesioko zerrenda grabatu+bistaratzeko gogoratu:
SessionScoped

DBtik sesio jakin bateko eragiketak lortu pantaila prestatzeko:
ViewScoped



DBtik autentikatutako erabiltzailearen sesioak ezabatu+bistaratzeko gogoratu eskaera bakoitzean: RequestScoped/SessionScoped



2.- JAVA EE

2.2.- Bussiness tier

- JSF+EJB+JP 3. ADIBIDEA: Aplikazio aritmetiko OSOA

package pl;

//import lerroak

@Named

@SessionScoped

public class ErabiltzaileAtazakMB implements Serializable {

private static final long serialVersionUID = 1L;

private SesioaE sesioaDB;

private int kodea;

private String mezua;

@EJB private ErabSesiokoEJB eEJB;

@EJB private OrokorraEJB oEJB;

@PostConstruct

private void sesioaSartuDB() {...} //AURREKO ADIBIDEKO KODE BERDINA

public SesioaE getSesioaDB() {...} //AURREKO ADIBIDEKO KODE BERDINA

public int getKodea() {...} //AURREKO ADIBIDEKO KODE BERDINA

public String getMezua() {...} //AURREKO ADIBIDEKO KODE BERDINA

public void resetSession() {...} //AURREKO ADIBIDEKO KODE BERDINA

public void eragiketaEginZerrendaratu(EragiketaFormMB eragiketaFormMB, ...) {...} //AURREKO ADIBIDEKO KODE BERDINA

public List<EragiketaE> eragiketakLortuEJB() {...} //AURREKO ADIBIDEKO KODE BERDINA

public void zerrendaBiltegitatu() {...} //AURREKO ADIBIDEKO KODE BERDINA

public void autSesioakEzabatuDB() {

kodea=oEJB.autSesioakEzabatuDB();

mezua=EuskarazPOJO.erabiltzaileMezuak[kodea];

}

}

MB: pl.ErabiltzaileAtazakMB.java

2.- JAVA EE

2.2.- Bussiness tier

- JSF+EJB+JP 3. ADIBIDEA: Aplikazio aritmetiko OSOA

```
package pl;

public abstract class EuskarazPOJO {
    public final static String erabiltzaileMezuak[] = {
        "MEZURIK EZ",//0
        "ZERRENDATUTAKO ERAGIKETA",//1
        "ZURE SESIOKO DATU HAUEK DBan GORDE DIRA",//2
        "ERRORREA ZURE SESIOA DBan GORDETZEAN: Sesioa jadanik ez dago DBan",//3
        "ERRORREA ZURE SESIOA DBan GORDETZEAN: Eragileren bat ez dago DBan eta ez da sesioa gorde",//4
        "ZURE SESIO GUZTIAK DBtik EZABATU DIRA",//5
        "ERRORREA ZURE SESIOA DBtik EZABATZEAN: Zure sesiorik ez dago DBan",//6
    };
}
```

POJO: pl.EuskarazPOJO.java

2.- JAVA EE

2.2.- Bussiness tier

- JSF+EJB+JP 3. ADIBIDEA: Aplikazio aritmetiko OSOA

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml"
      xmlns:h="http://java.sun.com/jsf/html"
      xmlns:f="http://java.sun.com/jsf/core"
      xmlns:ui="http://java.sun.com/jsf/facelets">
<ui:composition template="./txantiloia.xhtml">
<ui:define name="headTitle">Erabiltzailea2: Aplikazio aritmetikoaren sesio-laburpenaren orrialdea</ui:define>
<ui:define name="bodyBurua">APLIKAZIO ARITMETIKOA JSF, EJB eta JPArekin: SESIOAREN LABURPENA</ui:define>
<ui:define name="bodyEdukia">
  <h1>ZURE SESIOAN BILDUTAKO DATUAK</h1>
  Eragiketa kopurua: #{erabiltzaileAtazakMB.eragiketakLortuEJB().size()}
  <h:dataTable value="#{erabiltzaileAtazakMB.eragiketakLortuEJB()}" var="eragiketa" border="2">
    <h:column>
      <f:facet name="header">MOTA</f:facet>
      #{eragiketa.eragileaE.izena}
    </h:column>
    <h:column>
      <f:facet name="header">ERAGIKETAK</f:facet>
      #{eragiketa.eragigai1}#{eragiketa.eragileaE.ikurra}#{eragiketa.eragigai2}
    </h:column>
    <h:column>
      <f:facet name="header">EMAITZAK</f:facet>
      #{eragiketa.emaitza}
    </h:column>
  </h:dataTable>
  <p>
    <h:form>
      <h:commandButton action="erabiltzailea3" actionListener="#{erabiltzaileAtazakMB.zerrendaBiltegitatu()}" id="submitAdd"
        value="SESIOA DBan GRABATU" />
      <h:commandButton action="erabiltzailea3" actionListener="#{erabiltzaileAtazakMB.autSesioakEzabatuDB()}" id="submitRemove"
        value="ZURE SESIO GUZTIAK EZABATU" />
    </h:form>
  </p>
</ui:define>
</ui:composition>
</html>
```

XHTML: erabiltzailea2.xhtml

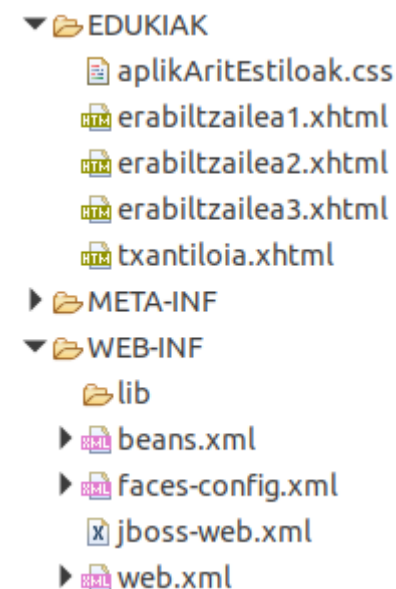
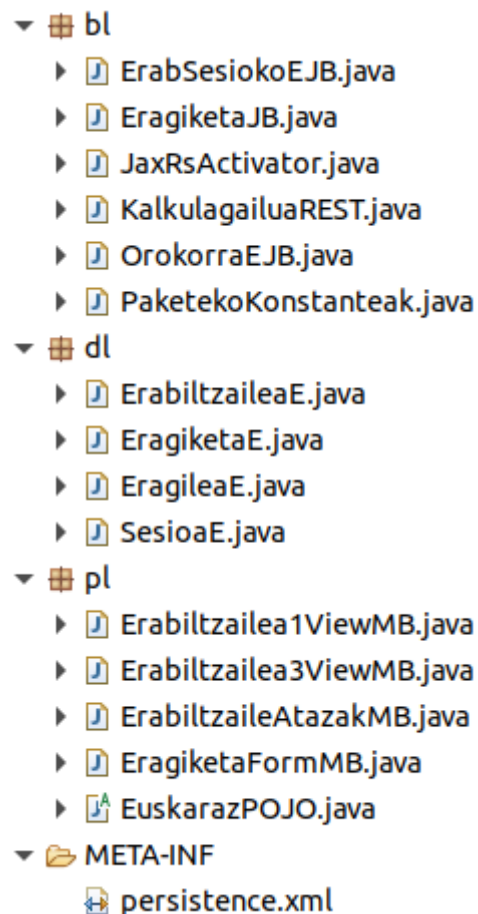
2.- JAVA EE

2.2.- Bussiness tier

- JSF+EJB+JP 4. ADIBIDEA: Aplikazio aritmetiko OSOA + WS

– JAVA osagaiak: **java**

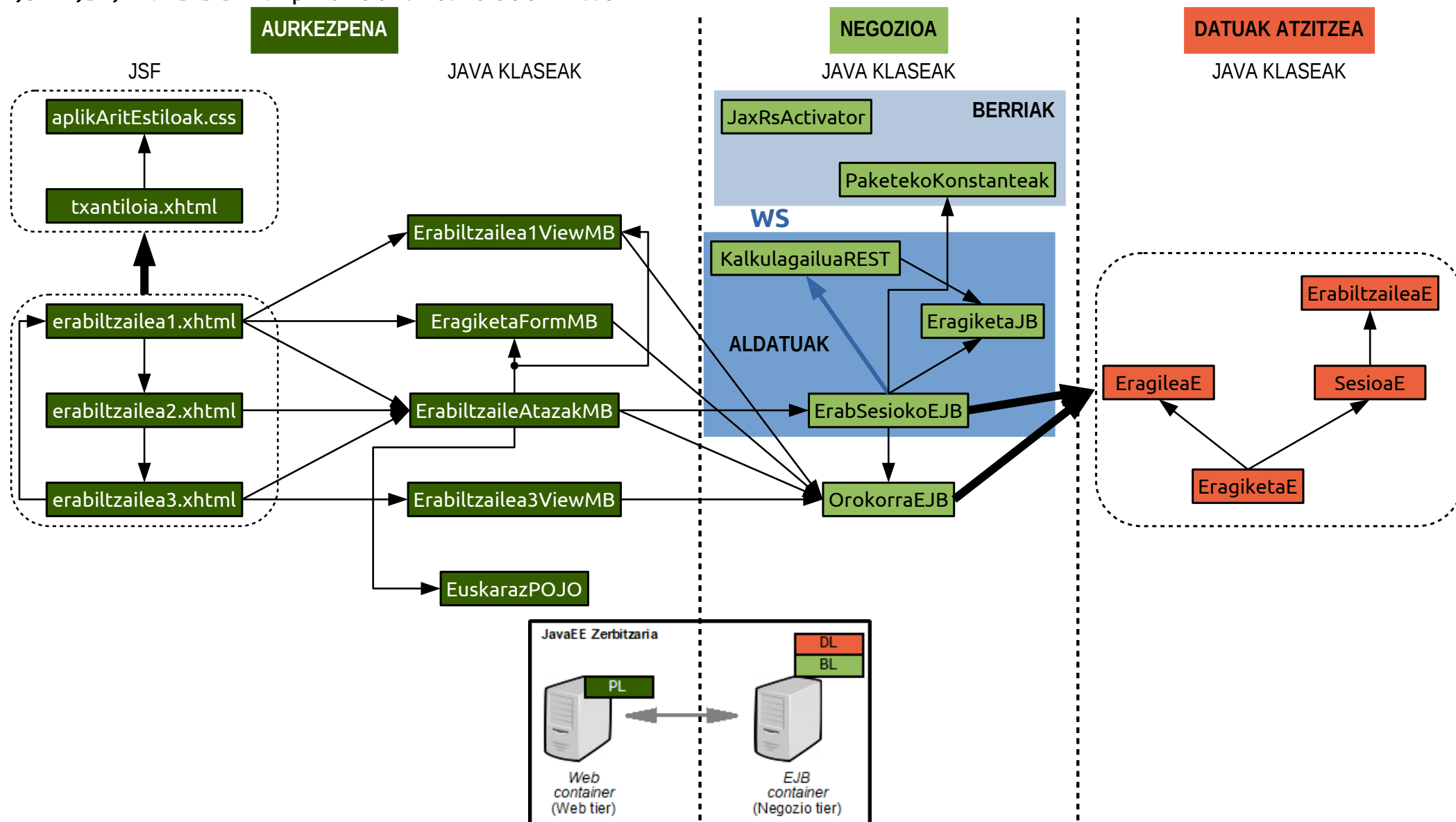
– HTML osagaiak: **webapp**



2.- JAVA EE

2.2.- Bussiness tier

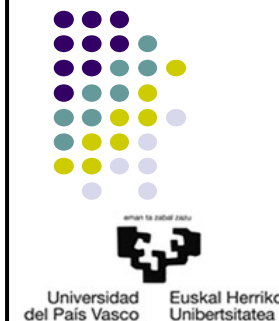
- JSF+EJB+JP 4. ADIBIDEA: Aplikazio aritmetiko OSOA + WS



2.- JAVA EE

2.2.- Bussiness tier

- JSF+EJB+JP 4. ADIBIDEA: Aplikazio aritmetiko OSOA + WS



//5. GAIKO AZKEN ADIBIDEKO KODE BERDINA	bl.EragiketaJB.java
-----------------------------------------	---------------------

//5. GAIKO AZKEN ADIBIDEKO KODE BERDINA	bl.KalkulagailuaREST.java
-----------------------------------------	---------------------------

<pre>package bl; class PaketekoKonstanteak { public static final String kalkulatuURL="http://localhost:8080/ProiektuarenIzena/rest/kalkulagailua/kalkulatuURL"; }</pre>	bl.PaketekoKonstanteak.java
------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	-----------------------------

2.- JAVA EE

2.2.- Bussiness tier

- JSF+EJB+JP 4. ADIBIDEA: Aplikazio aritmetiko OSOA + WS

EJB: bl.ErabSesiokoEJB.java

```
package bl;

//import lerroak

@SecurityDomain("DBM8")

@DenyAll
@Stateful
@LocalBean
public class ErabSesiokoEJB {
    private List<EragiketaE> eragiketakE;

    @EJB OrokorraEJB oEJB;

    public ErabSesiokoEJB() {...} //AURREKO ADIBIDEKO KODE BERDINA
    public List<EragiketaE> getEragiketakE() {...} //AURREKO ADIBIDEKO KODE BERDINA

    @RolesAllowed("erabiltzailea")
    public int eragiketaEginZerrendaratu(EragiketaE eragiketaE) {
        int kodea=1;

        EragiketaJB eragiketaJB=new EragiketaJB(eragiketaE.getEragigai1(), ""+eragiketaE.getEragigai2(), 0.0f);
        eragiketaJB=ClientBuilder.newClient()
            .target(PaketekoKonstanteak.kalkulatuURL)
            .request(MediaType.APPLICATION_JSON)
            .post(Entity.entity(eragiketaJB, MediaType.APPLICATION_JSON), EragiketaJB.class);

        eragiketaE.setEmaitza(eragiketaJB.getEmaitza());

        eragiketakE.add(eragiketaE);

        return kodea;
    }

    @RolesAllowed("erabiltzailea")
    public int zerrendaBiltegitatu(int idSesioa) {...} //AURREKO ADIBIDEKO KODE BERDINA
}
```


DATUEN MAILA JAVAEE_n

ERREFERENTZIAK

- [1] <https://docs.oracle.com/javaee/7/tutorial/>
- [2] <https://docs.oracle.com/javaee/7/firstcup/index.html>
- [3] <https://docs.oracle.com/javaee/7/api/>
- [4] <https://docs.oracle.com/javaee/6/jaserverfaces/2.1/docs/vdldocs/facelets/>