

**Informatika Fakultatea**

**Software Ingeniaritza 2**

# ***RIDES\_25***

## ***Hibernaterekin***

**Egilea:** Iker Mayordomo

**Data:** 14/12/2025

**Github-esteka:** <https://github.com/imayordomo/RidesJSF>

**Bideoaren esteka:**

[https://drive.google.com/file/d/1-S7WAezId\\_lraOJTN5pkg\\_1o5c6gL0bC/view?usp=sharing](https://drive.google.com/file/d/1-S7WAezId_lraOJTN5pkg_1o5c6gL0bC/view?usp=sharing)

# Aurkibidea

Zailtasunak eta Konponbideak.....	2
Ordu Kopurua.....	4
Ondorioak.....	5

# Zailtasunak eta Konponbideak

Proiektuaren garapenean zehar, hainbat arazoekin topatu naiz. Hona hemen garrantzitsuenak, aurkitu dudak konponbideekin:

- 2 EAGER ALDI BEREAN

Driver domeinuko klasean bi lista ditut: cars (kotxe zerrenda) eta rides (bidai zerrenda). Biak, @oneToMany izanda, fetch type bezala EAGER jarrita nuen; programa txikia baita eta errazago zelako dena ekartzea DBtik. Tamalez, ezin dira mi EAGER izan aldi berean ejekutatzeke garain errorea ematen duelako. Eta, ondorioz, LAZYra aldatu dut rides zerrenda, biak aldi berean kargatzea eta errorea sahiestuz.

- BEAN BAKOIZTA JSFrako

Proiektua modu ordenatu batean egiteko, erabilpen eta JSF bakoitzeko, Bean desberdin bat erabili dut. Honen ondorioz, nahiz eta garbiago eta estrukturatuago geratu proiektua, zailtasunak sortzen ditu informazioa batetik bestera pasatzeko.

Adibidez, logeatutako erabiltzailea create rideeko Beanean erabiltzeko. Arazo hori konpontzeko, interneten honako funtzio hauek aurkitu nituen:

```
FacesContext.getCurrentInstance().getExternalContext().getSessionMap().put("loginUser", d);
```

Honekin lortzen dut Login eta Register etik lortutako erabiltzaile-Driverra beste Beanetatik lortu ahal izatea sesio berdina erabiltzen den bitartean (hemen gordeta dagoelako) honako funtzio honekin:

```
private Driver getDriver(){  
    return (Driver) FacesContext.getCurrentInstance()  
        .getExternalContext()  
        .getSessionMap()  
        .get("loginUser");}
```

- VIEW SCOPED Y SESSION SCOPED

Aurreko puntuarekin zerikusia duen arazo bat izan da Bean batzuetan VIEW SCOPED nuela jarrita probak egin genuenetik. Horren ondorioz, beharrezko informazioa ezin zen pasa Beanen artean, inplementatutako metodoak sesioarekin funtzionatzen duelako. Konpontzeko, viewscoped nituen kasuetan, sessionscoped bezala ezarri ditut.

- CAR PKren KONPROBAKETA

addCar erabilpena gehitzean, konprobatzen nuen ea sortutako kotxeak zuen matrikula (Car-raren primary key-a) ez zela existitzen erabiltzaile horrentzat. Arazo izan zen, beste pertsonak zituzten matrikulak ez nituela konprobatzen, eta behin probak egitean jadanik existitzen matrikula bat sartu nuenean beste Driver batean zenez errore bat suertatu zen. Konpontzeko, erabiltzailearen matrikulak konprobatu ordez, Car DBko taulan find batekin konprobatzen dut orain.

- PERSISTen ORDENAREN GARRANTZIA

Car domeinua gehitzean, DB initalizatzeko metodoan pare bat gehitu nituen frogak egiteko. Lehengo aldiz exekutatzen ikusi nuen elementuak ez zirela ondo gehitu DBra; izan ere, Car bat zegoen lotuta persistitu gabeko erabiltzaile bati, alegia,

oraindikan existitzen ez zena. Konpontzeko, domeinu bakoitza sortu ondoren persistitu egiten det eta ondoren hurrengo domeinua sortzen eta lotzen hasten naiz.

## Ordu Kopurua

PROIEKTUKO ATALA	ORDUAK (h) [Aprox.]
Hasierako dokumentazioaren irakurketa	1.5h
Diseinua	1h
JSFko implementazioa	3h
Hibernateko implementazioa	4.5h
Dokumentazioa	1h
Besteak (Probak era errorean zuzenketa)	3.5h
<b>TOTALA</b>	<b>14.5h</b>

## Ondorioak

Proiektua amaitu ondoren, berriro ere eskertzekoa iruditzen zait Software Ingeniaritza (1 eta 2) hain ikasgai praktikoak izatea, modu egokienean ahalbidetzen baitu ezagutza teorikoak praktikan integratzea. Hibernatek eskaintzen dituen abantailak ondo ikasi ditut, baita JSF interfazeak erabiltzen ere, eta gainera, datu-base berri bat erabiltzeko aukera izan dut, kasu honetan MariaDB.

Nire ustez, proiektuaren alderdirik azpimarragarriena izan da pixkanaka-pixkanaka proiektu handi bat garatzeko eta beste teknologietara aldatzeko gai garela ikustea. Gainera, berriro ere baieztatu dut, Jon Iturriozek aurrean, zein iaz, esan duen bezala, debugak eta oro har erroreak zuzentzeko lanak, denbora asko eskatzen duela.