

# Projecte eEscola de gestió acadèmica

Autor: Ramon Giribet Paucirerol

Consultor: Vicenç Font Sagristà

Data: 15/06/2015

Assignatura: Treball Final de Carrera

Àrea: J2EE

Títol: Enginyeria Tècnica en Informàtica de Gestió

# Introducció

- Es vol treballar en posar en marxa una aplicació web per gestionar l'activitat administrativa i acadèmica d'una escola .
- Es farà usant la tecnologia JEE, una plataforma de programació en llenguatge Java amb arquitectura distribuïda per nivells i estructurada en components, que s'executa en un servidor d'aplicacions.
- S'implementa aquesta tecnologia usant un framework que integra totes les funcionalitats exigides per JEE, Spring sota patró MVC.
- Explorar com fer la integració de la persistència en l'àmbit de la plataforma JEE escollida.

## Objectius

- Treballar per obtenir una aplicació web per a gestionar una escola, el projecte eEscola
- Fer un aprenentatge intensiu del framework Spring que implementa la tecnologia JEE.
- Aquest aprenentatge més un correcte disseny i presa de requeriments, han de permetre als usuaris:
  - Gestionar l'accés i autenticació a l'aplicació
  - Gestió del procés de matriculació
  - Gestió del procés d'assignació i admissió d'alumnes
  - Gestió on-line dels avisos i notificacions
  - Sistema d'avaluació i qualificacions

## Metodologia de treball

S'ha seguit la que marca l'estructura de l'assignatura, que és la de cascada:

- 1) En un primer nivell fa una descripció genèrica de l'aplicació segons les necessitats que en un primer moment transmet el client.
- 2) En la segona fase és fa un anàlisi detallat de les funcionalitats que ha de satisfer l'aplicació. Es detecten les classe i les interrelacions que establiran, es plasma el resultat en els diagrames UML i d'entitat relació.
- 3) La tercera fase inclou la implementació del disseny escollit, el testing de l'aplicació i de forma ideal un anàlisi de la seva qualitat. Malauradament no s'ha pogut implementar tot el disseny i el testing s'ha anat fent conjuntament a la programació.

## Rols

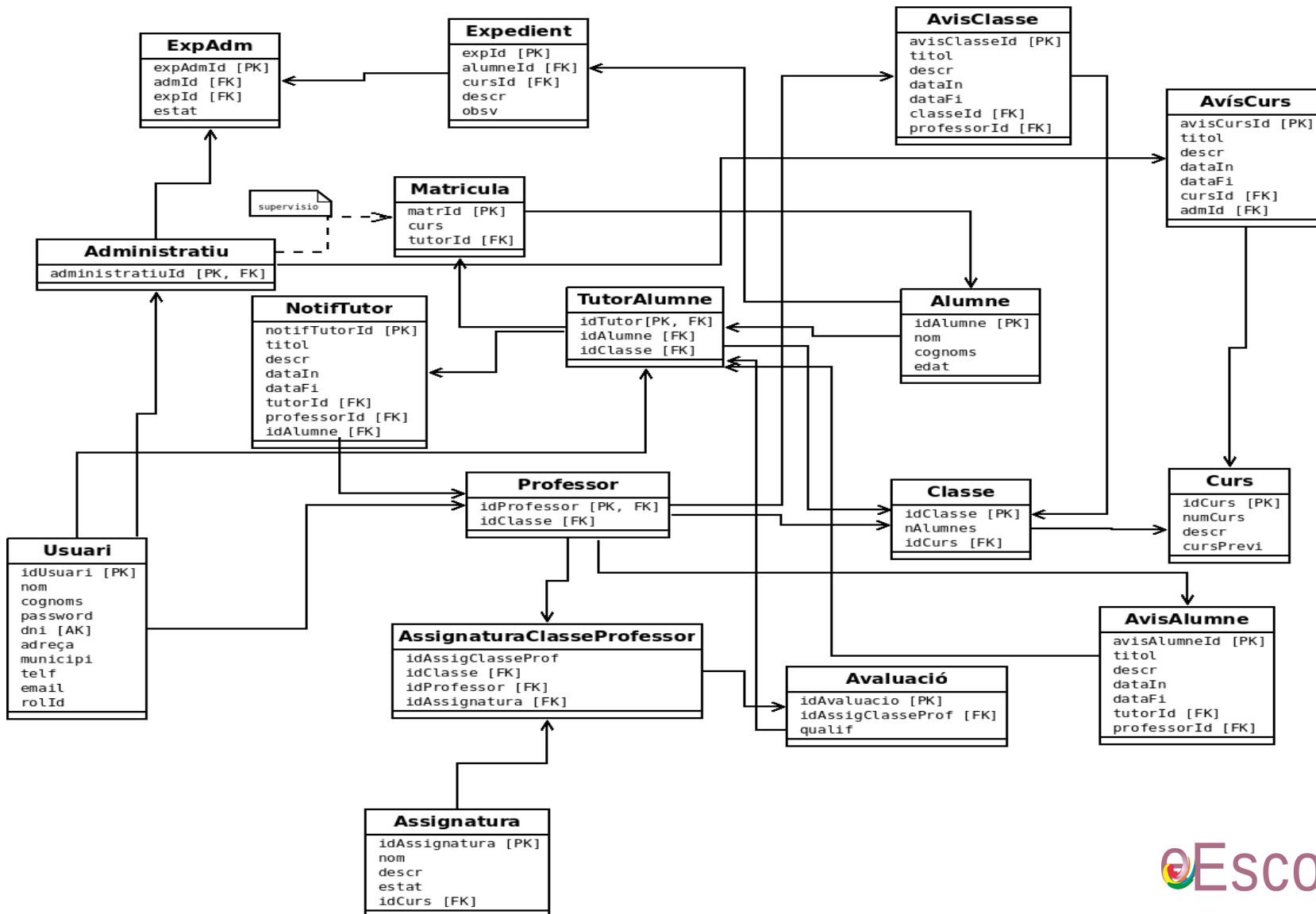
- **Personal administratiu:**
  - Manteniment de les dades del sistema, any acadèmic, professorat, etc.
  - Assignar professors a assignatures i aules
  - Assignar els alumnes a les aules
  - Notificacions i avisos
- **Pares/tutors:**
  - Matricular els alumnes atès que és una escola de primària
  - Modificar i tenir cura de les dades dels seus alumnes
  - Consultar les qualificacions i notificacions
- **Alumnes:**
  - En tractar-se d'un alumne de primària és un subjecte passiu en el procés de gestió acadèmica.
- **Professors:**
  - Avaluar i qualificar als alumnes de les seves aules
  - Emetre les notificacions referents a activitats de les seves aules

# Subsistemes

**El projecte s'ha dividit en cinc subsistemes:**

- Subsistema de connexió i autenticació
- Subsistema de manteniment
  - Manteniment d'usuari
  - Manteniment d'assignatures
  - Manteniment d'aules
  - Manteniment de cursos
- Subsistema de matriculació
- Subsistema d'avisos i notificacions
- Subsistema de gestió d'expedients

# Model lògic relacional



# Patrons i Arquitectura I

- S'usarà una arquitectura de tipus MVC (Model Vista Controlador) on se separen les capes de vista, negoci i dades.
- L'implementació del patró MVC es farà mitjançant el framework Spring i concretament el mòdul Spring MVC. Les seves avantatges són:
  - Clara divisió entre Controllers, Models i Views.
  - Tota l'estructura es pot implementar amb interfaces
  - Spring integra una capa Web (vista) que està directament integrada amb la capa de negoci.
  - Els controllers d'Spring, que conformen la capa de negoci, és configuren mitjançant IoC, fet que implica que ells creen i gestionen els objectes o beans.



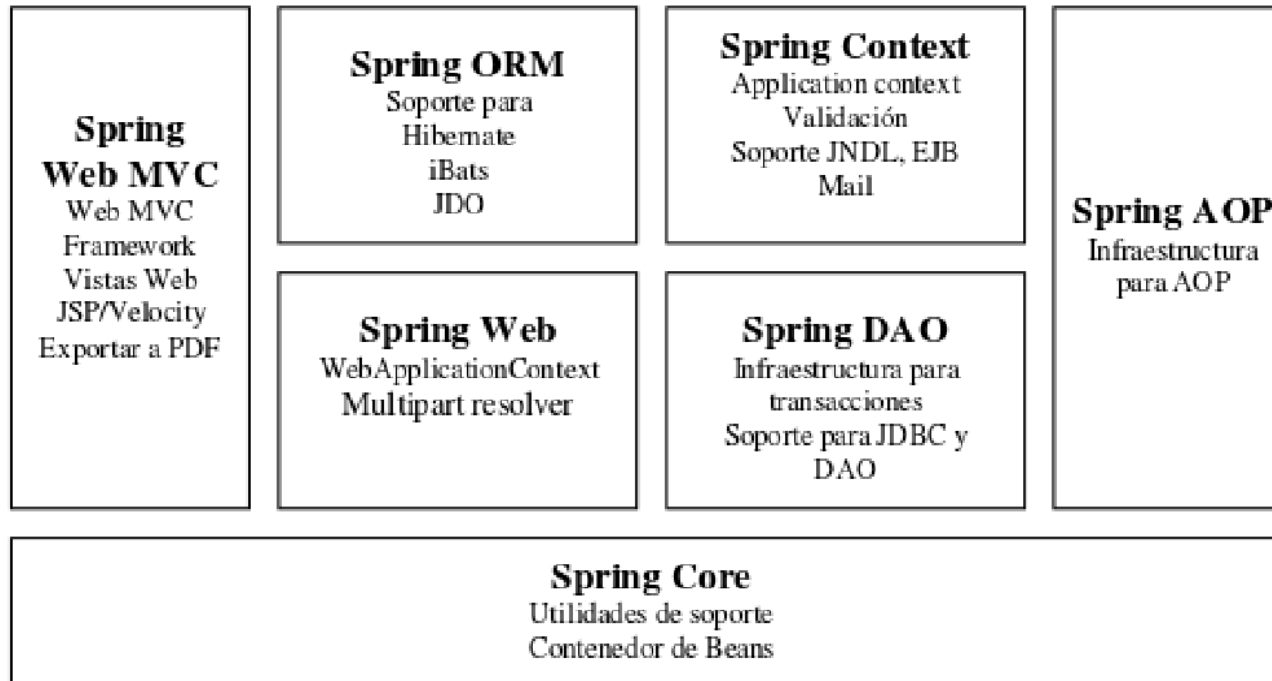
## Patrons i Arquitectura II

- S'usarà el patró DAO, integrat dins el mòdul Spring DAO, per a gestionar i dirigir la connexió amb el SGBD.
- El DAO es caracteritza per dur a terme sempre operacions atòmiques sobre la BBDD mai usa transaccions. Consta de:
  - Business Object: objecte que accedeix a la BBDD
  - Data Access Object: que emmascara els detalls de l'accés
  - Data Source: Implementació de la BBDD
  - Transfer Object: objecte intermedi entre l'objecte accessor i el DAO.
- Dins Spring DAO es troba un submòdul, Spring JDBC, que encara simplifica més les tasques de connexió amb la BBDD mitjançant una eina molt poderosa, el **JdbcTemplate**.

## Patrons i Arquitectura II

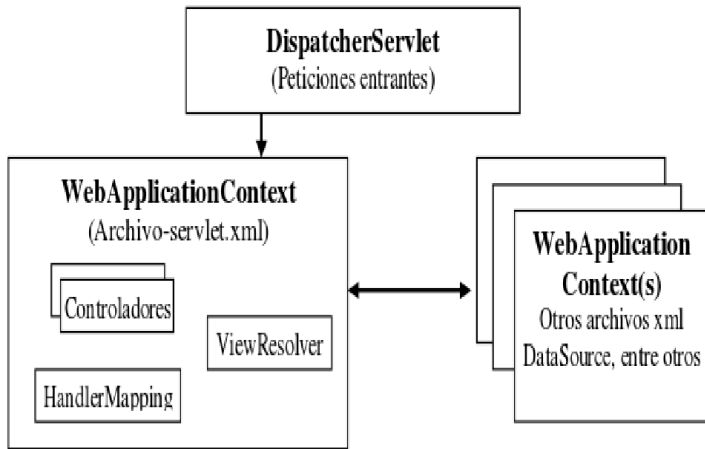
- **JdbcTemplate** és un objecte que gestiona les connexions de l'aplicació amb la BBDD usant l'API de JDBC. Però estalvia molts dels problemes que presenta l'ús de l'API:
  - Estalvia escriure molt de codi abans de l'execució de qualsevol consulta: no cal crear una connexió, ni els statement, ni tancar el result set o la connexió, ell ho gestiona automàticament.
  - No cal escriure el codi de gestió de les excepcions en la lògica de la BBDD.
  - No s'ha de fer una gestió directa de la transacció mentre s'efectua.
  - Per tant s'estalvia repetir molt de codi en cadascuna de les transaccions que s'han d'establir entre la capa de negoci i la de persistència.

# Components del framework Spring

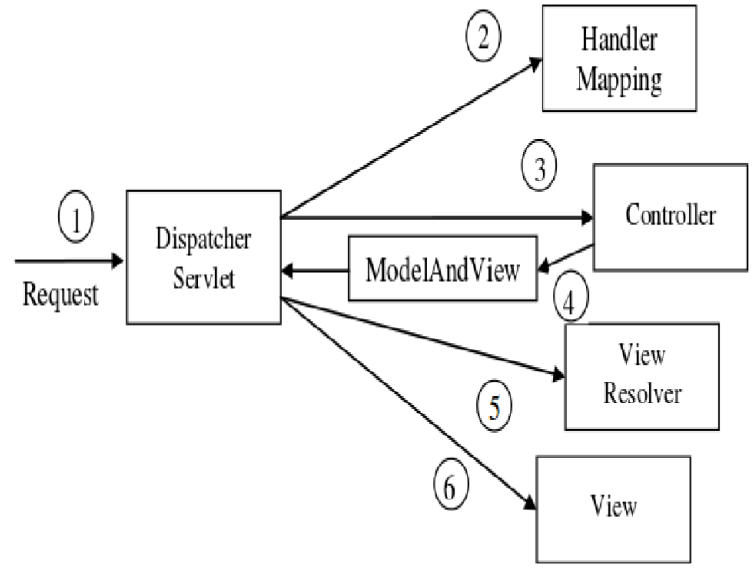


# Arquitectura i cicle de Spring MVC

## Arquitectura de Spring MVC



## Cicle d'una petició en Spring MVC



## Desplegament del projecte I

- Casos d'ús i funcionalitats implementades efectivament en el projecte eEscola:
  - **Connexió:** s'han implementat les funcionalitats de login, logout i tancament de sessió i també canvi de paraula clau.
  - **Matriculació:** s'ha implementat la funcionalitat de matriculació d'un alumne per part del seu tutor en tractar-se d'una escola de primària. També pot modificar, consultar i eliminar una matrícula. Des del punt de vista del personal administratiu, pot assignar els alumnes matriculats i acceptar la matrícula.
  - **Notificacions i avisos:** s'ha implementat la notificació automàtica d'avís d'assignació d'un alumne, la notificació apareix automàticament a la pantalla inicial del tutor/pare. Així com els avisos de modificació o eliminació d'una assignació.

## Desplegament del projecte II

- Hi ha tres pantalles principals d'on deriven la resta per oferir la funcionalitat sencera de cada subsistema.
- Pantalla de login: s'entra fent correr l'executable en format war, o amb el servidor en marxa, corrent l'adreça <http://localhost:8084/rgiribet.uoc.oop2.pro19.login/index>; la pantalla de login ens demana usuari i password
- S'identifica segons usuari el rol, i s'ofereix la pantalla de presentació de funcionalitats adient:
  - **Pantalla tutor/pare:** pot veure notificacions, accedir a fer una matricula, modificar-la o eliminar-la, consultar les assignacions i esborrar el llistat de notificacions. També pot modificar la paraula clau d'accés.

## Desplegament del projecte III

- **Pantalla PAS:** pot assignar els alumnes matriculats, sempre té un llistat obert amb les matriculacions actives, un cop feta l'assignació la matriculació es tanca i desapareix de la taula de matricules. Pot modificar i esborrar les assignacions de qualsevol alumne i automàticament s'envia una notificació al pare o tutor involucrat. També pot canviar la seva paraula clau per a l'accés.
- Val a dir que ha estat força laboriós i ha requerit un esforç de documentació ostensible poder aconseguir que la connexió d'un usuari es tanqui de forma correcta. Es a dir que un cop es tanqui la sessió no pugui accedir qualsevol altra persona a través d'alguna funcionalitat del navegador.

## Conclusions

- Malgrat no haver pogut implementar totes les funcionalitats definides en el projecte he pogut copsar l'essència i potencial que amaga la tecnologia JEE.
- Hi ha una total integració amb la resta de l'API de Java i per tant fa ús de la seva portabilitat i usabilitat.
- El producte obtingut té una alta escalabilitat i modularitat a banda de ser una solució totalment lliure.
- L'aprenentatge del funcionament del framework Spring, que implementa la tecnologia JEE, ha estat molt profitós si bé el procés ha estat lent i ha requerit molt esforç.
- Spring facilita molt la feina en fer-se càrrec mitjançant la propietat d'IoC i els mòduls específics, de gran part de les tasques rutinaries i repetitives, com les de creació de beans o connexió amb la BBDD.