

DISSENY I IMPLEMENTACIÓ D'UN MARC DE TREBALL (FRAMEWORK) DE PRESENTACIÓ PER APLICACIONS J2EE

Projecte Fi de Carrera - J2EE

Alumne: Daniel Clemente Marcè

Consultor: Josep Maria Camps Riba

Índex

- Estudi de frameworks
 - Java EE
 - Patró de disseny
 - Frameworks
 - Diferents alternatives de frameworks comercials
- Implementació d'un framework
 - Coneixements previs
 - Característiques
 - Funcionament
 - Restricció d'usuaris
 - Ajuda al desenvolupador – “mode debug”
- Conclusions

Estudi de frameworks (I)

Java EE

- Plataforma de programació escrita amb Java
- Arquitectura distribuïda amb nivells
- Basat en components de programari
- Executat en servidors d'aplicacions
- Independència de la plataforma
- Reutilitzable

Estudi de frameworks (II)

Patró de disseny

- ⦿ Solució general a un problema comú
- ⦿ Descripció o plantilla per resoldre un problema
 - ⦿ No és un disseny acabat que acaba sent un codi.
- ⦿ A partir del patró sorgeix el concepte d'arquitectura

Estudi de frameworks (III)

Frameworks

- ⦿ Implementa un o diversos patrons
- ⦿ Desenvolupament organitzat d'un projecte
 - ⦿ Metodologia
 - ⦿ Interfuncionalitat
 - ⦿ Recursos
 - Programari de suport
 - Llibreries
 - Llenguatges
 - Programari extra

Estudi de frameworks (IV)

Alternativa 1: JavaServer Faces JSF

- Orientat a la interfície gràfica
- Servlet com a controlador
- Model basat en components
- Similar al GUI standalone com Swing

Estudi de frameworks (V)

Alternativa 2: Spring MVC

- Principal característica: modularitat
- No és un framework MVC, té un mòdul MVC
- Codi obert
- Funció de connector entre APIs

Estudi de frameworks (VI)

Alternativa 3: Struts

- Dels més utilitzats
- Programari lliure
- Ús de servlet
- JavaServerPages(JSP) opcionalment
- Configuració mitjançant XML
- Workflow separat de la lògica de Negoci
- Fàcil d'utilitzar

Estudi de frameworks (VII)

Alternativa 4: WebWork 2.2/Struts 2.0

- Fusió entre dos frameworks separats inicialment
- Robust
- Interfícies reutilitzables

Implementació d'un framework (I)

Coneixements previs (punts anteriors)

- Tecnologia Java
- Patrons
- Marcs de treballs
- Altres frameworks
 - Funcionament
 - Característiques
 - Ús

A partir d'aquest punt ja podem dissenyar un nou framework

Implementació d'un framework (II)

Característiques

- ⦿ Java
- ⦿ Implementa el model MVC
- ⦿ Priorització de la facilitat d'ús
 - ⦿ Ajuda en el desenvolupament
- ⦿ Mòdul de restricció d'usuaris
- ⦿ Configuració mitjançant fitxers properties (tant el workflow com el propi framework)

Implementació d'un framework (III)

Funcionament

- Controlador implementat mitjançant servlets
- Workflow de l'aplicació utilitzant fitxer properties
- White-box: totes les accions deriven d'una interfície comuna (hot-spots)
- Vistes mitjançant jsp

Implementació d'un framework (IV)

Restricció usuaris

- Opcionalment podem restringir l'accés a certes pàgines segons el rol de l'usuari
- Integrat al framework i molt senzill d'utilitzar
- Inclou un simulador de login (per utilitzar mentre no es té accés a la B.DD.)

Implementació d'un framework (V)

Ajuda al desenvolupador - “mode debug”

- Identificació d'errors en l'ús del framework
- Informació detallada dels errors comesos
- Proposta de solució a l'error
- Activació/desactivació del mode debug segons programador o usuari final
- Facilitat al màxim del seu ús: prioritització de la corba d'aprenentatge

Conclusions

- ⦿ Paradigma actual: programació en frameworks
- ⦿ Coneixement de les diferents alternatives per dissenyar i construir un nou framework
- ⦿ Creació d'un framework prioritzant
 - ⦿ Simplicitat
 - ⦿ Corba d'aprenentatge
- ⦿ Adoptar una nova visió de programador a arquitecte