

Histórico de Versões

Data	Versão	Descrição	Autor	Revisor	Aprovado Por
10/12/2008	1.0	Versão Inicial	Heitor		
13/12/2008	1.0	Mudança do nome do framework para Demoiselle	José René		
12/04/2010	2.0	Atualização do documento	Flávio Lisboa		
13/04/2010	2.1	Revisão do documento	Flávio Lisboa	Serge Rehem	Antonio Tiboni

Sumário

Histórico de Versões.....	1
Sumário.....	2
Documento de Arquitetura de Referência.....	3
1. Objetivo.....	3
2. Representação Arquitetural.....	3
3. Objetivo e Restrições Arquiteturais.....	4
4. Visão Lógica.....	4
4.1. Visão Geral.....	4
4.1.1. Camada de Visão e Controle.....	5
4.1.2. Camada de Negócio.....	6
4.1.3. Camada de Persistência.....	7
5. Visão de Processos Concorrentes.....	8
6. Visão de Implantação.....	8
7. Visão de Implementação.....	9
8. Visão de Dados.....	9
9. Tamanho e Performance.....	9
10. Qualidade.....	9
Referências.....	10

Documento de Arquitetura de Referência

1. Objetivo

Este documento é uma referência para os documentos de arquitetura de software (DAS) de aplicações web não distribuídas que utilizam o Demoiselle. Derivações desta arquitetura devem ser feitas sob demanda, de acordo com as necessidades do projeto, pelo arquiteto. Os diagramas da visão lógica identificam as estruturas do sistema de forma genérica através do radical "UseCase".

2. Representação Arquitetural

Para ilustrar a arquitetura de um sistema de software baseado no Demoiselle Framework, este documento faz uso de uma representação na qual as camadas verticais definem a divisão de interesses da aplicação e as camadas transversais são serviços providos pelo framework para as camadas. A figura 1 apresenta uma visão geral desta arquitetura:

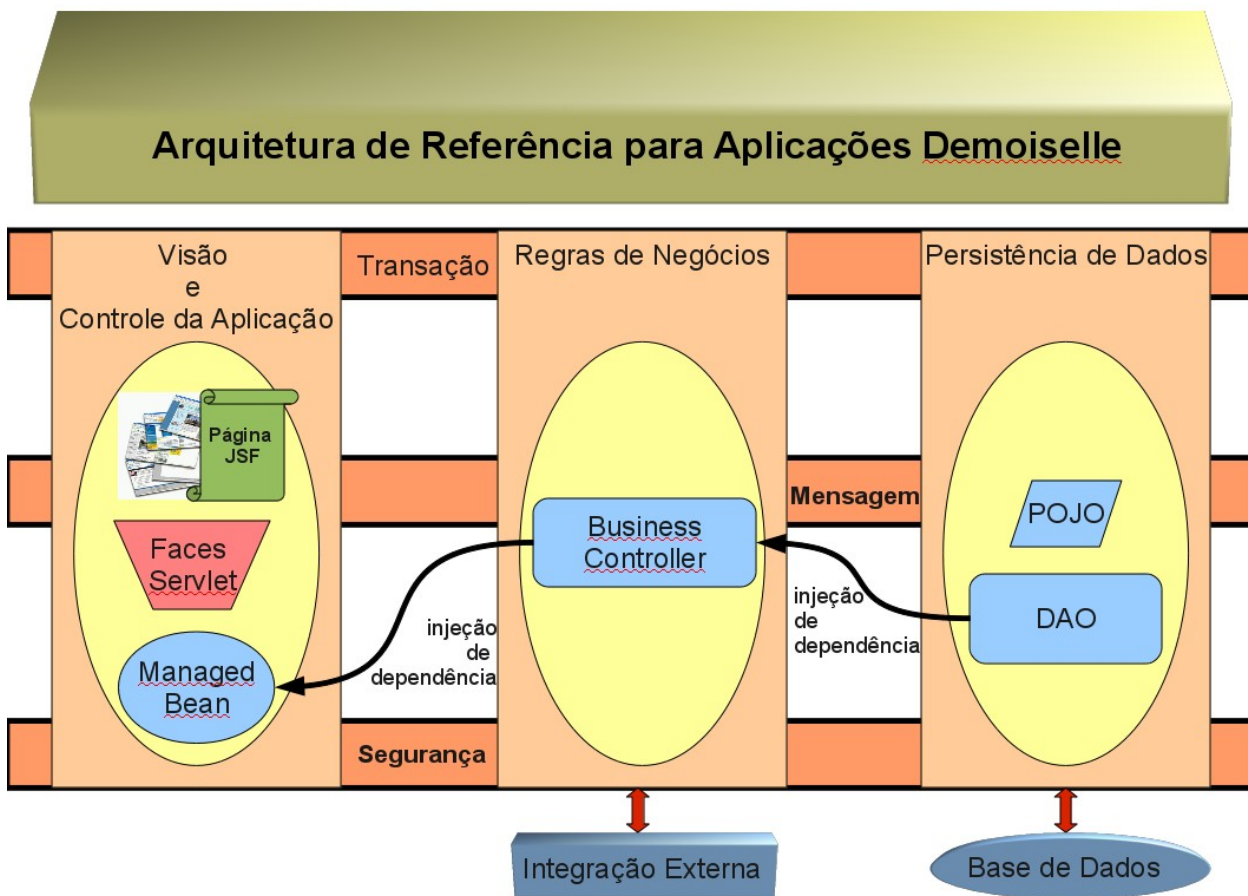


Figura 1: Arquitetura Geral

3. Restrições Arquiteturais

Esta seção descreverá as principais restrições da arquitetura da aplicação de referência para aplicações que utilizam o Demoiselle Framework.

Restrições 1	Aplicações Web
Decisão de Projeto:	Arquitetura desenhada apenas para aplicações web não distribuídas.
Restrições 2	Ambiente de produção
Decisão de Projeto:	É necessário um ambiente de produção com servidor compatível com a especificação JSR-000154 (Servlet 2.5)

4. Visão Lógica

Esta seção apresenta a visão lógica da arquitetura. Esta visão provê uma perspectiva estrutural do sistema, apresentando como este se divide em camadas, quais os principais elementos de projeto destas camadas, suas interfaces e relacionamentos.

4.1. Visão Geral

A utilização de camadas é baseada em Fowler (2006, p. 37-38), que as defende pelo fato de permitirem a fácil substituição de implementações, a minimização de dependências entre partes da aplicação, e a definição de padrões que podem gerar código reutilizável.

Assim sendo, o sistema construído com o framework Demoiselle divide-se em camadas distintas de visão/controlê, negócio e persistência. As próximas subseções apresentarão cada camada e suas respectivas responsabilidades.

Um ponto forte da tecnologia Java para aplicações Web é a facilidade de manter estado entre requisições HTTP. O framework oferece às aplicações serviços relacionados a isso na forma de contextos.

O contexto de transação especifica um mecanismo de controle transacional e atua no início e fim de cada ação. Ele prevê dois tipos de funcionamento: um no qual a aplicação se responsabiliza pelo gerenciamento da transação e outro no qual a aplicação dependerá de uma implementação JTA (Java Transaction API) disponível no container Web.

O contexto de mensagens define uma abstração de mensagens trocadas durante uma requisição entre as camadas do sistema.

O contexto de segurança especifica um mecanismo padrão para acesso a dados de segurança referentes a autenticação e autorização. Ele trata a autorização por meio de papéis e é baseado na especificação JSR 196 (JAAS).

O uso dos contextos é facilitado por um localizador de contextos, implementado pela classe ContextLocator do framework.

4.1.1. Camada de Visão e Controle

Responsável apresentar informações e controlar a interação com os atores externos ao sistema.

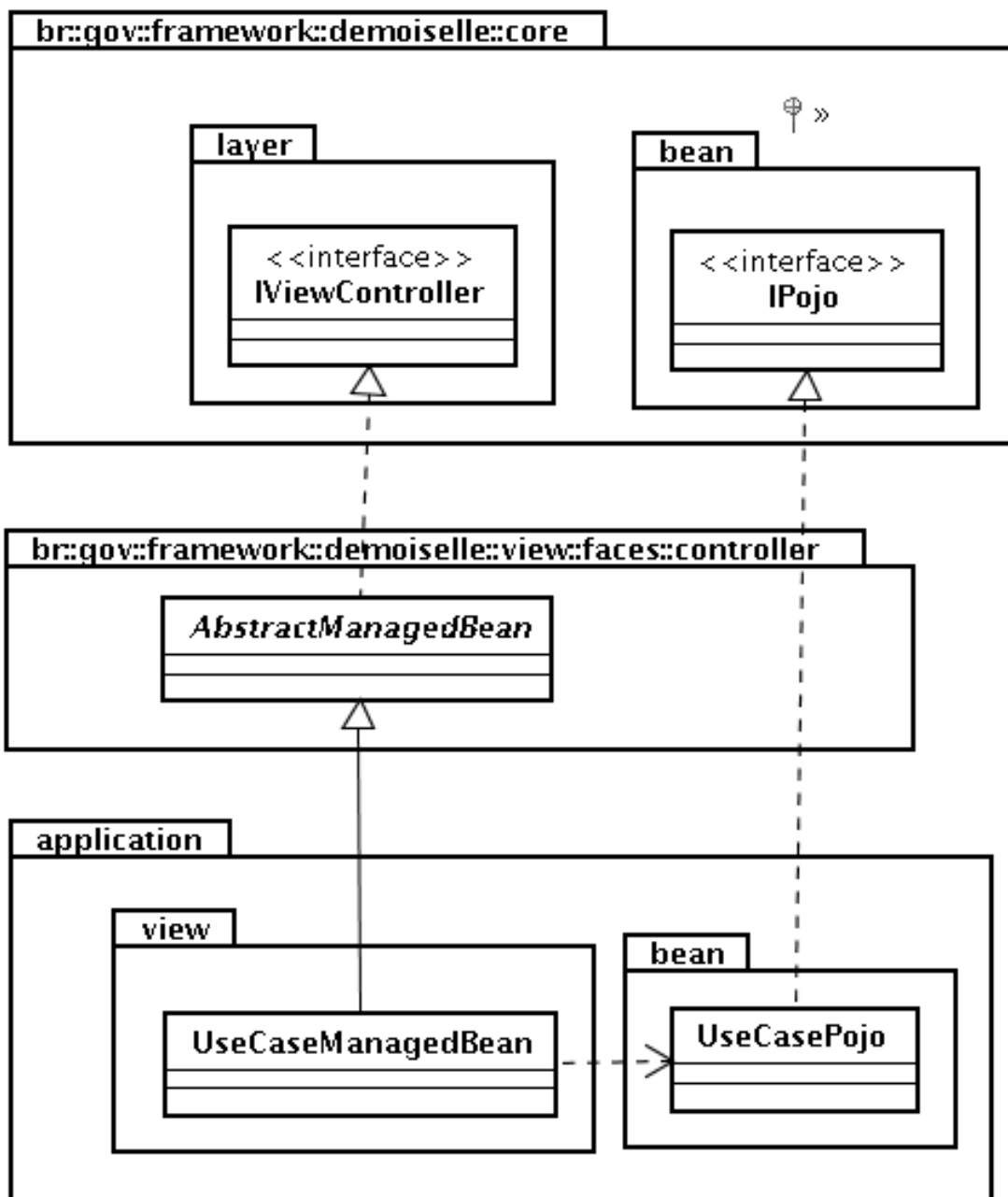


Figura 2: Camada de Visão e Controle

4.1.2. Camada de Negócio

Implementa as regras de negócio definidas para o sistema. Pode definir processos de integração com outros módulos e/ou subsistemas.

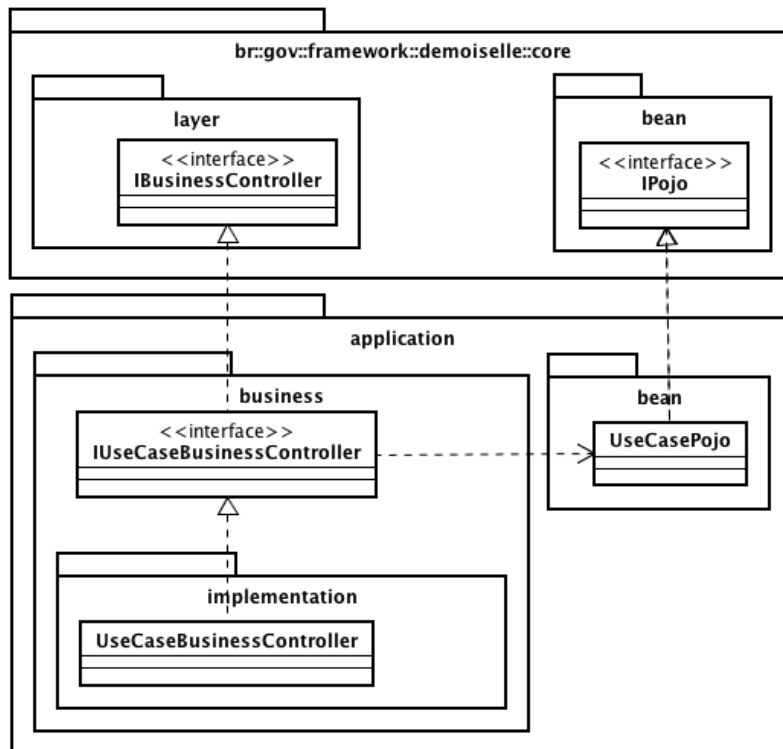


Figura 3: Camada de regras de negócio

4.1.3. Camada de Persistência

Realiza a integração do sistema com outros sistemas gerenciadores de dados para garantir o armazenamento dos estados dos objetos (POJO).

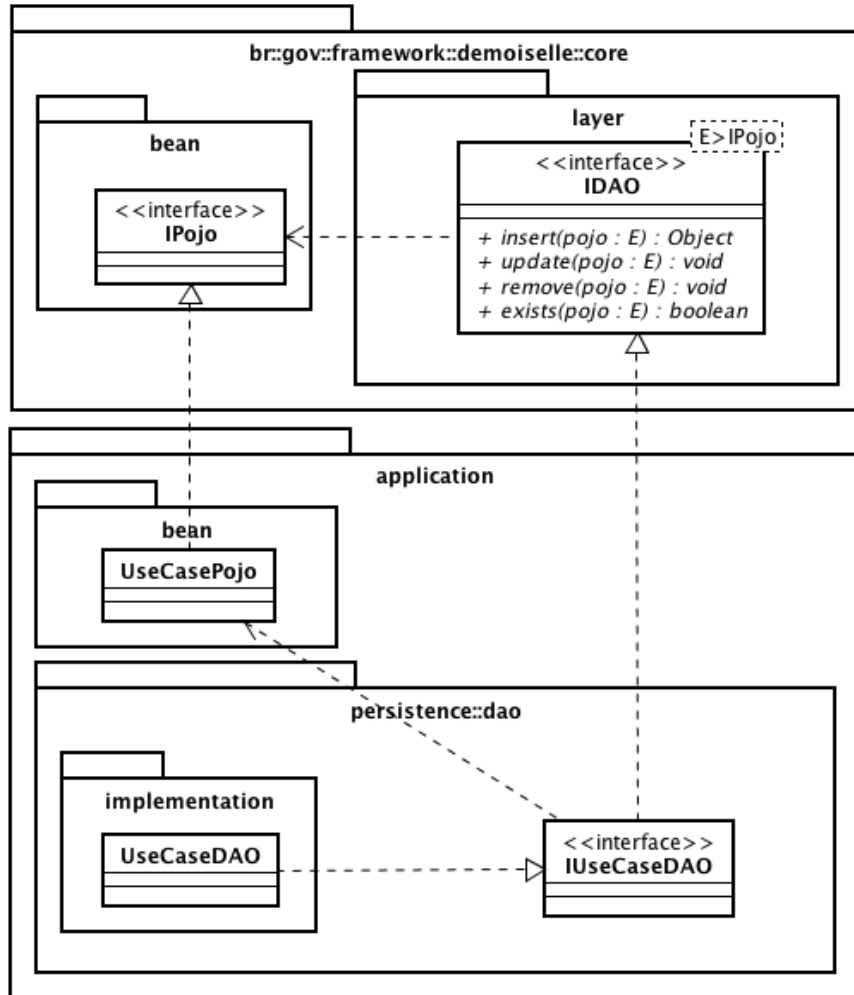


Figura 4: Camada de persistência

5. Visão de Processos Concorrentes

O contexto transacional fornecido pelo Demoiselle Framework permite controlador vários recursos transacionais de forma concorrente.

6. Visão de Implantação

A figura 5 descreve a configuração de distribuição dos componentes do software mais comum para as aplicações que utilizam o framework.

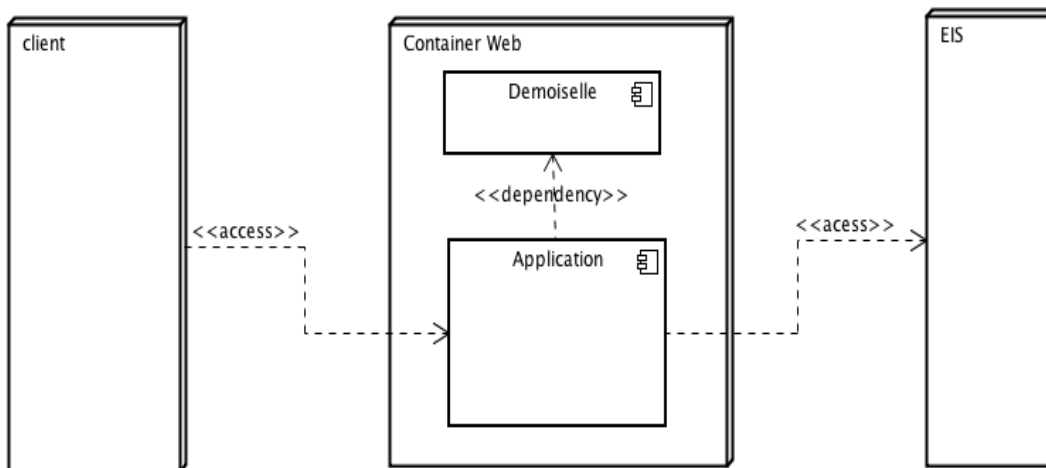


Figura 5: Visão de Implantação

7. Visão de Implementação

A figura 6 demonstra a estrutura de pacotes proposta pelo Framework para as aplicações web.

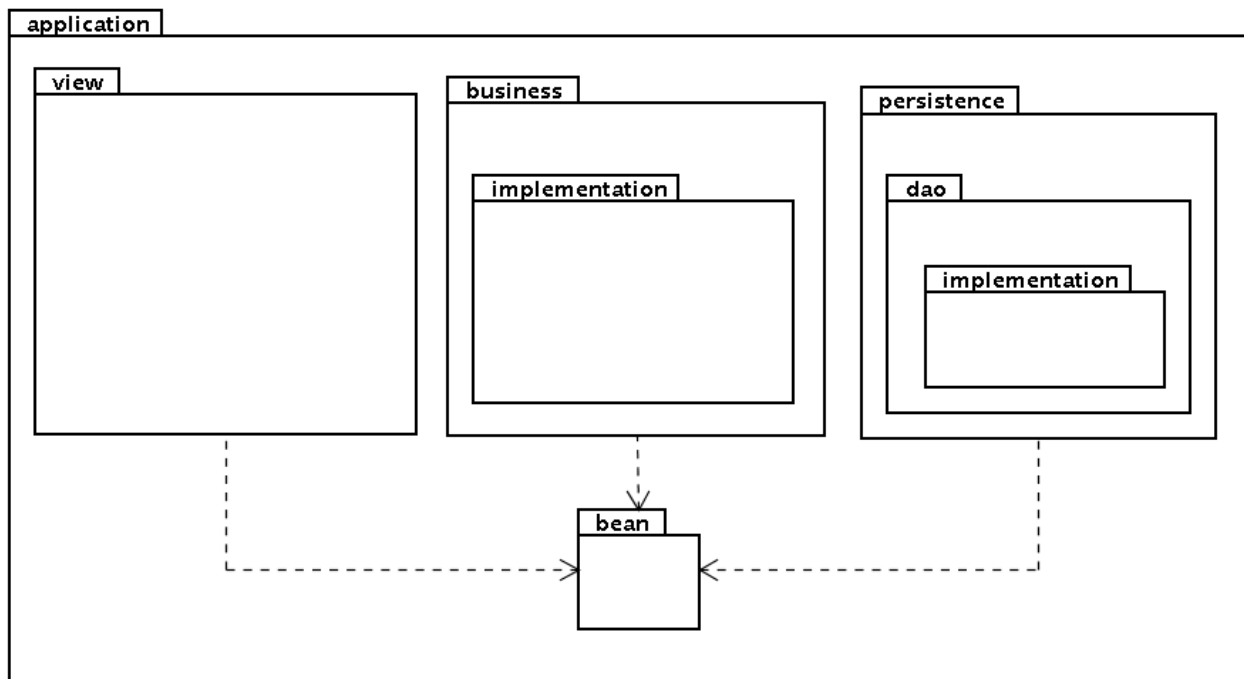


Figura 6: Visão de Implementação

O componente `demoiselle-archetype-webapp-sample` do subprojeto Demoiselle Components dispõe de um esqueleto de aplicação, que por meio da ferramenta Maven constrói automaticamente essa estrutura.

8. Visão de Dados

A versão mínima recomendada do Demoiselle Framework é a 1.1, que faz uso de JPA, pois ela permite configurar o acesso a mais de uma fonte de dados com escolha de um provedor específico.

9. Tamanho e Performance

O framework não possui restrições conhecidas de tamanho e performance.

10. Qualidade

Os critérios de qualidade devem estar explícitos na arquitetura final da aplicação, que contempla a arquitetura de referência e o domínio do problema.

Referências

ALLUR, Deepak, CRUPI, John e MALKS, Dan. *Core J2EE Patterns: As melhores práticas e estratégias de design*. Rio de Janeiro. Campus, 2002.

BAUER, Christian e King, Gavin. *Java Persistence com Hibernate*. Rio de Janeiro. Ciência Moderna, 2007.

DEMOISELLE FRAMEWORK. *Demoiselle Quickstart*. Disponível em <<http://demoiselle.sourceforge.net/1.1.0/quickstart.html>>. Acesso em 13/04/2010.

FOWLER, Martin. *Padrões de Arquitetura de Aplicações Corporativas*. Porto Alegre. Bookman, 2006.

GAMMA, Erich et al. *Padrões de Projeto: Soluções Reutilizáveis de Software Orientado a Objetos*. Porto Alegre. Bookman, 2000.

GEARY, David e HORSTMANN, Cay. *JavaServer Faces Fundamentos*. 2.ed. Rio de Janeiro. Alta Books, 2007.

JAVA COMMUNITY PROCESS. *JSR-000154 Java Servlet 2.5 Specification*. Disponível em <<http://jcp.org/aboutJava/communityprocess/mrel/jsr154/index.html>>. Acesso em 13/04/2010.

HORSTMANN, Cay. *Padrões e Projeto Orientados a Objetos*. 2.ed. Porto Alegre. Bookman, 2007.

METSKER, Steven John. *Padrões de Projeto em Java*. Porto Alegre. Bookman, 2004.