Relatório de Implementação

do

Gestor Grego

Versão 0.5.0

|  |  |
| --- | --- |
| Fernanda Tamy Ishii | PG 48330 |

|  |  |
| --- | --- |
| Professores: | Edson  Itana |
| Disciplina: | DIN4063 – Engenharia de Software |
|  |  |

|  |  |
| --- | --- |
|  |  |

Sumário

[1. Introdução 4](#_Toc426935462)

[2. Frameworks, padrões de projeto e componentes 4](#_Toc426935463)

[2.1. Hibernate 4](#_Toc426935464)

[2.2. JSF/Primefaces 4](#_Toc426935465)

[2.3. Spring MVC 4](#_Toc426935466)

[2.4. Spring Webflow 4](#_Toc426935467)

[3. Arquitetura detalhada do sistema 5](#_Toc426935468)

[4. Diagrama de instalação final 6](#_Toc426935469)

[5. Estimativa do tamanho do código (LOC) 6](#_Toc426935470)

[6. Funcionalidades implementadas 7](#_Toc426935471)

[7. Avaliação da consistência dos modelos versus implementação 8](#_Toc426935472)

[8. Avaliação do processo de desenvolvimento 8](#_Toc426935473)

[9. Comentários gerais 9](#_Toc426935474)

Revisões

| Versão | Autores | Descrição da Versão | Data |
| --- | --- | --- | --- |
| 0.4.1 | Fernanda | Criação do Documento | 28/07/2015 |
| 0.4.2 | Fernanda | Revisão do Documento | 06/08/2015 |
| 0.4.3 | Fernanda | Edição da Versão Final | 10/08/2015 |

# Introdução

O objetivo deste documento é apresentar o workflow de implementação da aplicação Gestor Grego, a ser utilizada em um restaurante do tipo healthy fast-food. Para isso, ele apresenta os padrões, componentes, frameworks e plataforma utilizados para esta implementação, bem como a estimativa de linhas de código (LOC), as funcionalidades implementadas, a arquitetura final do sistema e o diagrama de implantação.

# Frameworks, padrões de projeto e componentes

Neste projeto, foram utilizados os seguintes frameworks, padrões de projeto e componentes:

## Hibernate

O Hibernate é um framework para o mapeamento objeto-relacional disponível na linguagem Java, que facilita o mapeamento dos atributos entre uma base tradicional de dados relacionais e o modelo objeto de uma aplicação, mediante o uso de arquivos (XML)

## JSF/Primefaces

O Primefaces é uma biblioteca construída para o JSF, que auxilia no desenvolvimento da interface para aplicações Java para Web utilizando componentes visuais pré-definidos, de modo a mitigar os esforços do desenvolvedor.

## Spring MVC

Spring MVC é um framework baseado no container Spring para aplicações *web*, que auxilia na organização do código dentro do padrão MVC e oferece suporte para gerenciar o fluxo entre as páginas da aplicação *web*.

## Spring Webflow

Spring Web Flow é uma extensão do Spring MVC que permite a implementação do fluxo de navegação de uma aplicação web.

## Padrão MVC

MVC é um padrão de arquitetura de software, que separa a aplicação em 3 camadas distintas: A camada de interação do usuário, chamada de view, a camada de manipulação dos dados, designada model, e a camada de controle das requisições do cliente e das operações e serviços internos, denominado controller.

# Arquitetura detalhada do sistema

A figura 1 apresenta a arquitetura final do sistema Gestor Grego, com a organização dos padrões, frameworks e componentes utilizados para sua implementação.

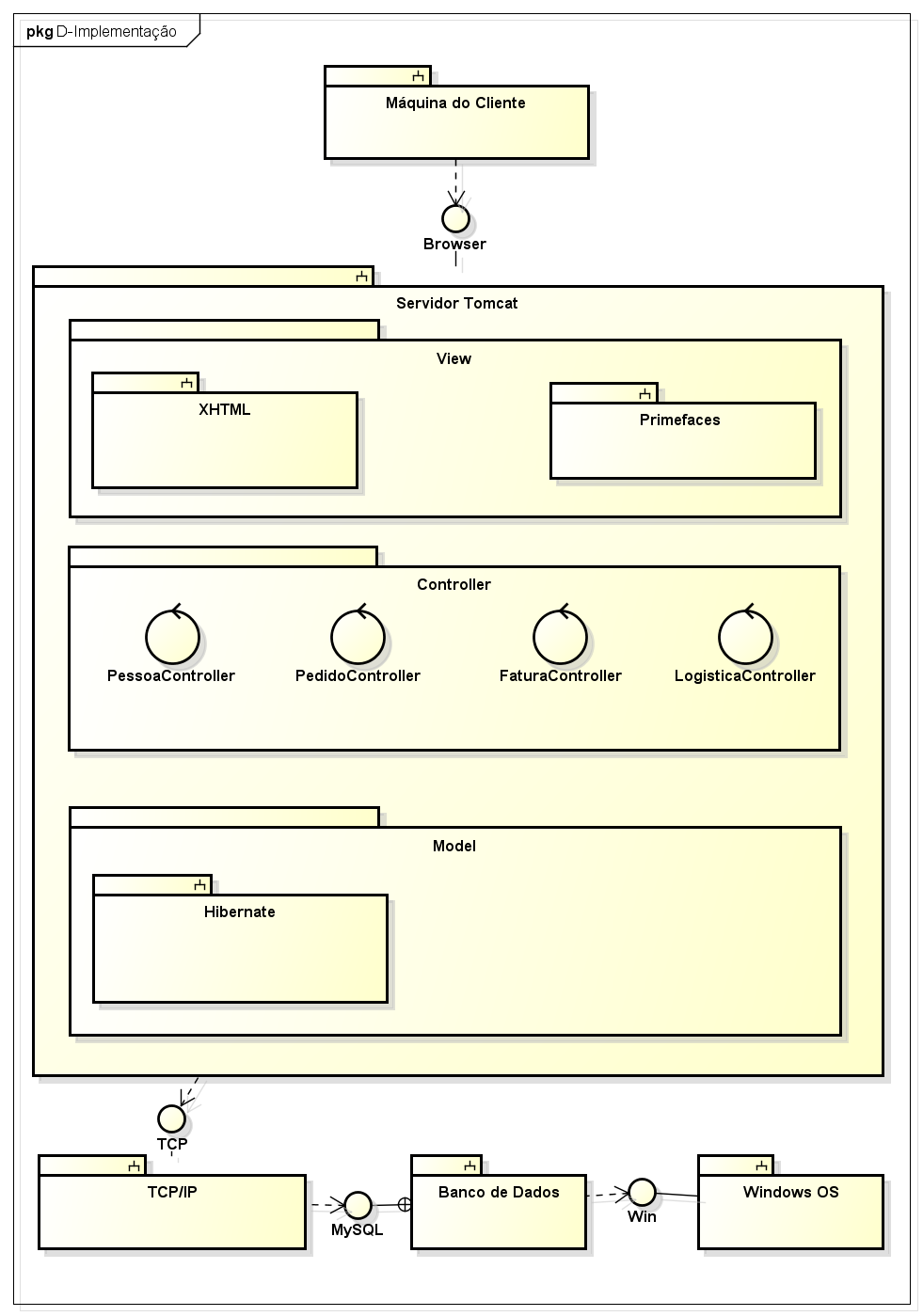


Figura 1 - Arquitetura final do sistema

# Diagrama de instalação final

O diagrama de instalação foi alterado com a remoção do servidor exclusivo para clientes. Este servidor foi unificado ao servidor da operação, como ilustrado na figura 2.

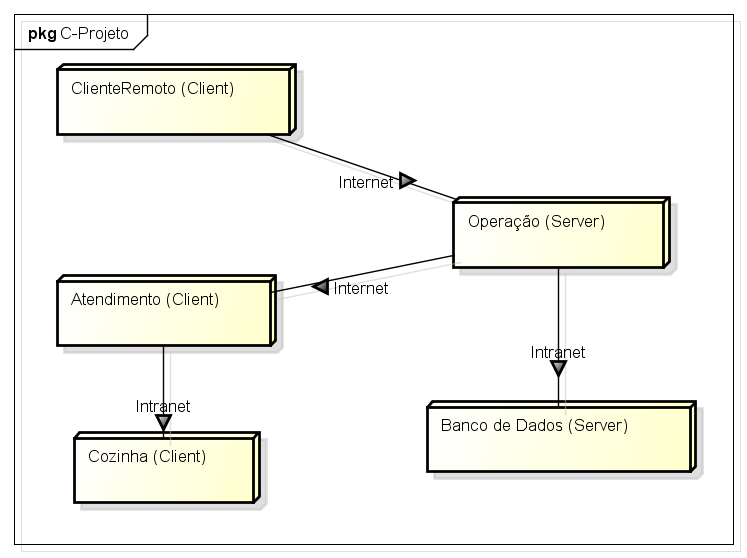


Figura 2 - Diagrama de instalação final

# Estimativa do tamanho do código (LOC)

Como a implementação deste sistema não foi concluído, para a realização da estimativa (LOC), utilizou-se o produto simples do número de funcionalidades (5) e da quantidade média de linhas de código implementadas. Assim, obteve-se os valores da tabela 1.

Tabela 1 - Estimativa (LOC)

|  |  |  |  |
| --- | --- | --- | --- |
| **Checkpoint Name** | **Baseline Total** | **Valores Estimados** | |
|  |  | **Por Funcionalidade** | **Total** |
| Files | 29,00 | 15,00 | 75,00 |
| Lines | 1.076,00 | 538,00 | 2.690,00 |
| Statements | 492,00 | 492,00 | 492,00 |
| % Branches | 0,40 | 0,40 | 0,40 |
| Calls | 97,00 | 97,00 | 97,00 |
| % Comments | 3,30 | 3,30 | 3,30 |
| Classes | 29,00 | 29,00 | 29,00 |
| Methods/Class | 4,07 | 4,07 | 4,07 |
| Avg Stmts/Method | 1,27 | 1,27 | 1,27 |
| Max Complexity | 3,00 | 3,00 | 3,00 |
| Max Depth | 3,00 | 3,00 | 3,00 |
| Avg Depth | 0,94 | 0,94 | 0,94 |
| Avg Complexity | 1,02 | 1,02 | 1,02 |

# Funcionalidades implementadas

O sistema deveria apresentar as seguintes funcionalidades, conforme mencionado no documento de requisitos deste projeto:

* Estoque: Gestão das entradas, transferências e saídas de produtos entre estoques.
* Financeiro: Visualização do fluxo de caixa e das contas pendentes e agendadas.
* Mix de Produtos: Cadastro e manutenção de produtos, ingredientes e itens.
* Pessoas: Cadastro de Clientes e Colaboradores.
* Relatório: Emissão de relatórios diversos, tais como: Relatório de Vendas (RV), Relatório de Estoque (RE) e Demonstrativo de Resultado do Exercício (DRE).
* Vendas: Gestão do Caixa, Acompanhamento dos pedidos no balcão e dos pedidos de entrega.

A tabela 2 apresenta a situação das funcionalidades previstas para o sistema.

Tabela 2 - Situação das funcionalidades implementadas

|  |  |  |
| --- | --- | --- |
| # | Funcionalidade | Status |
| 1 | Estoque | Iniciado e pendente |
| 2 | Financeiro | Iniciado e pendente |
| 3 | Mix de Produtos | Iniciado e pendente |
| 4 | Pessoas | Iniciado e pendente |
| 5 | Vendas | Iniciado e pendente |
| 6 | Relatórios | Não Iniciado1 |

A funcionalidade de geração de relatórios não foi implementada devido a acordo realizado com a cliente, que optou por definir os parâmetros específicos de cada relatório após a implantação e homologação do sistema.

# Avaliação da consistência dos modelos versus implementação

Dada a adoção do padrão MVC e da utilização dos frameworks supracitados, o modelo de arquitetura do projeto foi o que mais sofreu alterações, uma vez que a representação inicial não considerava a organização dos objetos a serem implementados, tampouco os pacotes aos quais eles seriam posicionados.

A implementação também tornou visível a desnecessidade da separação do servidor em operação e cliente, como definida anteriormente no modelo de implantação. Assim, esse modelo também sofreu alterações.

# Avaliação do processo de desenvolvimento

O processo de desenvolvimento desse projeto, realizada de forma parcialmente sequencial e iterativa, foi realizada com a entrega de atividades e documentos parciais no decorrer do mesmo. Tais entregas serviram como gates para o disparo das atividades seguintes, bem como a retroalimentação de informações para a proposição da melhor solução.

O workflow de requisitos enfatizou a importância de uma boa definição dos requisitos do sistema, independentes de sua classificação (como funcional ou não-funcional) e garantiu o alicerce para a continuidade do proejto, definido com a delimitação dos casos de uso deste sistema. O workflow de análise permitiu uma organização das informações obtidas na fase anterior, garantindo uma melhor tomada de decisão sobre os objetos da aplicação, bem como sobre a coordenação deste projeto. Por fim, o workflow do projeto contribuiu com a delimitação dos métodos, atributos, classes e tecnologias a serem manipulados na implementação do sistema.

# Comentários gerais

A condução deste projeto permitiu confirmar a importância de um bom planejamento no projeto e desenvolvimento de um sistema inédito para uma dada área de negócio. A existência de uma cliente (contratante) real e fornecedora de requisitos, mesmo com uma boa comunicação e um relacionamento anterior com um dos integrantes do projeto, permitiu compreender as dificuldades encontradas na trnsmissão de ideias e necessidades, principalmente causadas pela existência de pontos de vista diferentes.

No que tange os prazos deste projeto, a ocorrência da suspensão das aulas, decorrente em função da greve dos funcionários das IEE’s do Paraná, causou complicações na organização temporal deste projeto, uma vez que os prazos definidos inicialmente para a entrega do projeto tiveram de ser ajustadas ao calendário do programa.

O desconhecimento dos integrantes na tecnologia sugerida pelo professor para a implementação foi um dos fatores que comprometeram a qualidade do projeto como um todo.

Outro fator agravante ao bom desenvolvimento deste projeto foi dada pela desistência de um dos integrantes da equipe, justamente na última etapa. Isso causou a necessidade de readequação de cronograma e negociação de novos prazos com o professor da disciplina.