**Meeting Music**

Versão 0.2**Controle de Versão**

|  |  |  |  |
| --- | --- | --- | --- |
| **Versão** | **Data** | **Razões para alteração** | **Responsável** |
|  | dd/mm/aaaa |  |  |

**Envolvidos na elaboração do Documento de Arquitetura**

|  |  |
| --- | --- |
| **Nome** | **Área** |
| Carlos Rodrigues | Nome da área a que pertence o profissional |
| Johann Oliveira Rosner |  |

**Índice**

# Introdução

## Objetivo do Documento

Este documento apresenta uma visão geral abrangente da arquitetura do projeto Meeting Music, utilizando uma série de visões arquiteturais diferentes para ilustrar seus diversos aspectos.

Sua intenção é capturar e transmitir as decisões significativas do ponto de vista da arquitetura que foram tomadas em relação ao projeto.

## Escopo

O escopo deste documento se aplica ao projeto como um todo, sua aplicação afeta toda a parte arquiteturial do sistema, como

[Faça uma breve descrição da aplicação do Documento de Arquitetura de Software; o que é afetado ou influenciado por este documento. Deixe claro se o Documento de Arquitetura se aplica ao projeto como todo (Exemplo: desenvolvimento de uma nova aplicação) ou se apenas se restringe a uma parte específica do projeto (Exemplo: uma integração entre sistemas).]

## Siglas, Abreviações e Acrônimos

[Liste as siglas que precisam ser explicadas para o entendimento completo do documento de Arquitetura.]

## Referências

[Esta seção fornece uma lista completa de todos os documentos referenciados em qualquer ponto do Documento de Arquitetura de Software. Identifica cada documento por titulo, número do relatório se aplicável, data, responsável pela publicação e eventual caminho de rede onde a referência se encontra. Especifica as fontes nas quais as referências podem ser obtidas.]

1. Glossário da Arquitetura, versão 99.99 de 99/99/9999 no site da MDS
2. Padrão Java / WebSphere, versão 99.99 de 99/99/9999 no site da MDS
3. Integração Visual Basic / Informix-4GL no site do SQA na antiga Portonet
4. Integração Java / Informix-4GL no site do SQA na antiga Portonet
5. API WebSphere MQ - Manual do projetista no site do SQA na antiga Portonet
6. API WebSphere MQ - Manual do programador no site do SQA na antiga Portonet
7. Documento de Visão, versão 99.99 de 99/99/9999 em //servidor/diretorio
8. Caso de Uso nome\_do\_caso\_de\_uso, versão 99.99 de 99/99/9999 em //servidor/diretorio
9. Glossário, versão 99.99 de 99/99/9999 em //servidor/diretorio
10. Solicitação de Elaboração do Documento de Arquitetura, versão 99.99 de 99/99/9999 em //servidor/diretorio
11. Correio de nome\_do\_remetente em 99/99/9999 sobre assunto\_tratado
12. Reunião com nome\_participante\_1 e nome\_participante\_n em 99/99/9999 sobre assunto\_tratado

# Metas e Restrições da Arquitetura

* Servidor Apache Tomcat 7.0.61
* SQL Server 2014
* Java EE 7
* Junit 3.8.1
* Framework Spring 4.1.6
* Framework Mockito 1.10.19
* Framework Hibernate 4
* Envio de relatório de erros
* Envio de email

# Visão de Casos de Uso

[Esta seção lista os casos de uso ou cenários do modelo de casos de uso se eles representam uma funcionalidade central e significativa do sistema final ou se têm uma ampla cobertura de arquitetura, ou seja, se experimentam muitos elementos arquiteturais ou se enfatizam ou ilustram um determinado ponto frágil da arquitetura. Exemplos: Um caso de uso que requer a transmissão de imagens, um caso de uso que requer a integração com outra aplicação.]

< DIAGRAMA COM CASOS DE USO IMPORTANTES À ARQUITETURA >

## Nome do Caso de Uso

[Descrição breve do caso de uso, enfatizando os pontos críticos à Arquitetura.]

# Visão Lógica

## Camadas da Aplicação

**Camada de Apresentação**

A camada de apresentação web (GUI) será responsável por prover as funcionalidades de iteração para os atores.

**Camada de Segurança**

A camada de segurança será responsável pela autenticação e autorização do usuário (caso o mesmo tenha se cadastrado e queira efetuar login). Também restringe as permissões do usuário com o que ele poderá interagir.

**Camada de Controle**

Será responsável por fazer o processamento entre a camada de modelo e a camada de apresentação. Controla o fluxo de informações da aplicação.

**Camada de Modelo**

Responsável pela regra de negócio e acesso a base de dados.

## Nomenclatura da Aplicação

[Esta seção é pertinente apenas no desenvolvimento de novas aplicações.]

A aplicação a ser desenvolvida terá as seguintes identificações:

|  |  |
| --- | --- |
| Macrosistema | Meeting Music |
| Nome da aplicação | Meeting Music |
| Pacotes | br.com.meetingmusic.frontend.views  br.com.meetingmusic.frontend.scripts  br.com.meetingmusic.frontend.styles  br.com.meetingmusic.backend.models  br.com.meetingmusic.backend.controllers  br.com.meetingmusic.persistence.daos  br.com.meetingmusic.common.factories |

Visão de Implementação

[Descreva nesta seção os componentes e frameworks que sejam significativos para Arquitetura, independentemente de pertencerem à Infraestrutura, a terceiros (Apache, por exemplo)ou à aplicação; de já estarem prontos ou precisarem ser construídos.]

## Componentes e frameworks a serem reutilizados

| Identificação | Responsável | Descrição |
| --- | --- | --- |
| Spring-core-4.1.6.RELEASE.jar | Terceiros | Biblioteca base para utilização do Framework Spring |
| Spring-orm-3.1.2.RELEASE.jar | Terceiros | Biblioteca para integração entre o spring e frameworks ORMs |
| Guava-18.0.jar | Terceiros | Biblioteca que contem diversas classes utilitárias que serão usadas no projeto |
| Junit-3.8.1.jar | Terceiros | Framework de testes unitários |
| Mockito-core.1.10.19.jar | Terceiros | Framework de criação de testes baseados em mocks de objetos |

## Tratamento de Erros e Exceções

Os erros serão que ocorrerem no sistema serão capturados e apresentados ao usuário (em caso de exceções, apresentar ao usuário a mensagem da mesma) e será proposta uma opção de reportar o erro.

# Visão de Implantação

[O diagrama deve exibir todos os nodes (clientes e servidores) envolvidos na solução, seus tipos e nomes (Exemplo: <<Servidor de Aplicações>> li07:WebSphere, <<Banco de Dados>> u01:Informix), a forma de comunicação entre os nodes (BUS – mesma máquina, LAN – rede interna, INTERNET – rede externa) e a distribuição dos principais componentes entre eles.]

< DIAGRAMA DE IMPLANTAÇÃO >

## Servidor de Aplicações

| Datasource | Provider | Versão | Servidor | Banco | Login |
| --- | --- | --- | --- | --- | --- |
| Jdbc/sqlserver | SQLServer |  | mm01 | meetingmusic | developer |

## Servidor de Banco de Dados

Nome do Banco de Dados: meetingmusic

Endereço do servidor: 127.0.0.1:1433

Dados de Acesso:

User: developer

Senha: Dev01

Collation: SQL\_Latin1\_General\_CP1\_CI\_AS

# Tamanho e Desempenho

[Uma descrição das principais características de dimensionamento do software que têm um impacto na arquitetura, bem como as restrições do desempenho desejado. Corresponde a requisitos não funcionais descritos no Documento de Visão.]

* tempo de resposta de uma realização de caso de uso
* quantidade de usuários simultâneos da aplicação
* disponibilidade da aplicação
* frequência de mensagens trafegadas em uma integração (qtde / tempo)
* tamanho das mensagens trafegadas em uma integração (parâmetros ou bytes)
* frequência de imagens enviadas para o sistema de imagens (qtde / tempo)
* tamanho das imagens enviadas para o sistema de imagens (bytes)