

<Musiverso>

Documento de Requisitos
Versão 1.3

Histórico de Revisões

Data	Versão	Descrição	Autor
	1.0	Versão inicial do documento	Kaio Henrique Cunha
	1.1	Alterações feitas nos casos de uso e suas descrições	Kaio Henrique Cunha
	1.2	Adicionados os diagramas de classe, pacotes e sequência	Kaio Henrique Cunha
	1.3	Alteração nos casos de uso	Kaio Henrique Cunha

Índice

MUSIVERSO

1	INTRODUÇÃO	4
1.1	PROPÓSITO DO DOCUMENTO DE REQUISITOS	4
1.2	ESCOPO DO PRODUTO	4
1.3	CONCEPÇÃO DO SISTEMA	4
1.4	CONVENÇÕES, TERMOS E ABREVIACÕES	4
1.4.1	<i>Identificação dos Requisitos</i>	4
1.4.2	<i>Prioridade dos Requisitos</i>	5
1.5	REFERÊNCIAS	5
1.6	VISÃO GERAL	5
2	DESCRIÇÃO GERAL	6
2.1	USUÁRIOS DO SISTEMA	6
2.2	ABRANGÊNCIA E SISTEMAS SIMILARES	6
	SISTEMAS SIMILARES:	6
2.3	SUPOSIÇÕES E DEPENDÊNCIAS	7
3	REQUISITOS DO SOFTWARE	8
3.1	REQUISITOS FUNCIONAIS	8
3.2	REQUISITOS NÃO-FUNCIONAIS	8
3.2.1	<i>Requisitos de Segurança</i>	8
3.2.2	<i>Requisitos de Interface</i>	8
3.2.3	<i>Requisitos de Operacionais</i>	8
3.2.4	<i>Requisitos de Confiabilidade</i>	8
4	CASOS DE USO	8
4.1	DIAGRAMA DE CASOS DE USO	8
4.2	DESCRIÇÃO DOS CASOS DE USO	9
4.2.1	<i>Efetuar cadastro(create)</i>	9
4.1.2	<i>Efetuar login(doLogin)</i>	9
4.1.4	<i>Autenticar(authentication)</i>	10
4.1.5	<i>Gerenciar usuários(edit e delete)</i>	10
5	DIAGRAMA DE CLASSES	10
5.1	DIAGRAMA DE CASOS DE USO	11
5.1	INTERAÇÃO ENTRE AS CLASSES	11
6	DIAGRAMA DE PACOTES	12
5.1	DIAGRAMA DE PACOTES	12
7	DIAGRAMA DE PACOTES	12
7.1	DIAGRAMA DE SEQUÊNCIA	12

Musiverso

1 Introdução

1.1 Propósito do documento de requisitos

Este documento destina-se aos clientes, engenheiros e gerentes envolvidos no desenvolvimento do sistema, doravante referido apenas como Musiverso. O propósito deste documento é apresentar a descrição dos serviços e funções que o sistema a ser desenvolvido deve prover, bem como as suas restrições de operação e propriedades gerais, a fim de ilustrar uma descrição detalhada do sistema para um auxílio durante as etapas de análise, projeto e testes. O documento especifica todos os requisitos funcionais e não funcionais do sistema e foi preparado levando-se em conta as funcionalidades levantadas durante a fase de concepção do sistema.

1.2 Escopo do produto

O projeto consiste na construção de um sistema (aplicação web) cujo objetivo é gerenciar o cadastro de composições e autores de um site de música chamado Musiverso. Este sistema visa auxiliar o cadastro de composições nos concursos oferecidos pelo site, automatizando o controle sobre as composições, otimizando tempo e custos. Será usado por funcionários da organização.

1.3 Concepção do sistema

Foram usados dois métodos para que pudessem ser obtidos os requisitos do sistema:

- Consulta com especialista:
 - Jocélio Otávio, desenvolvedor da Zup, orientou no levantamento de requisitos devido sua experiência em trabalhar com aplicações Java para a Web.
 - Caine O'Rear, engenheira de software e desenvolvedor do site AmericanSongwriter, orientou a análise de requisitos devido a sua grande experiência em desenvolvimento de software e com sites de música;

1.4 Convenções, termos e abreviações

Para evitar interpretações incorretas deste documento, algumas convenções e termos específicos são descritos a seguir:

CPF	Cadastro Nacional de Pessoa Física
ID	Identificador
JDK	Java Development Kit

1.4.1 Identificação dos Requisitos

Cada requisito será unicamente identificado no formato [tipoRequisito.numero]. Para requisitos funcionais, o código do tipo de requisito será RF, e para requisitos não funcionais, RNF. Um número será assinalado a cada requisito de forma incremental, na ordem que forem mencionados neste documento.

1.4.2 Prioridade dos Requisitos

Foram adotadas as seguintes denominações para estabelecer a prioridade dos requisitos: essencial, importante e desejável.

- **Essencial:** é o requisito sem o qual o sistema não entra em funcionamento, ou seja, são requisitos imprescindíveis tendo que ser implementados impreterivelmente.
- **Importante:** é o requisito sem o qual o sistema entra em funcionamento, mas de maneira insatisfatória, ou seja, devem ser implementados, mas se não forem, o sistema poderá ser implantado e usado mesmo assim.
- **Desejável:** é o requisito que não compromete as funcionalidades básicas do sistema, podendo funcionar de forma satisfatória sem ele, ou seja, são requisitos que podem ser deixados para versões posteriores do sistema, caso não haja tempo hábil para implementá-los na versão que

está sendo especificada.

1.5 Referências

Esta subseção apresenta as referências aos documentos que utilizamos no auxílio à construção deste documento de requisitos.

- Engenharia de Software - Uma Abordagem Profissional - 8ª Ed. 2016 - Pressman, Roger S.
- Página da Disciplina Especificação de Requisitos e Validação de Sistemas-
<http://www.cin.ufpe.br/~if716/>
- Material do Moodle da disciplina de Orientação a Objetos e de Projeto de Bloco do INFNET.

1.6 Visão Geral

Este documento está organizado da seguinte forma:

- A seção 1 apresentou uma introdução ao documento de requisitos e ao sistema sendo especificado;
- A seção 2 apresenta uma descrição geral do sistema;
- A seção 3 apresenta as definições dos requisitos funcionais e não-funcionais do sistema;
- A seção 4 apresenta o diagrama de casos de uso do sistema, bem como as descrições dos casos de uso definidos;

2 Descrição geral

2.1 Usuários do sistema

Usuários: Realizam as tarefas comuns, tal como: logar, acessar o conteúdo do site(artigos), atualizar as próprias informações de cadastro, bem como cadastrar composições nos concursos ofertados bi-mensalmente.

Administradores: Responsáveis pelo gerenciamento de usuários, além de terem acesso às suas funcionalidades de usuário.

2.2 Abrangência e sistemas similares

Abrangência:

O sistema irá conter ferramentas para construção de uma aplicação web que esteja de acordo com os objetivos e metodologia de um site de conteúdo(artigos) gratuito e premium para usuários assinantes. Para tal, o sistema consiste em gerenciar o cadastro e assinatura dos usuários, o conteúdo postado, e o material enviado pelos usuários para os concursos. O administrador através de ferramentas como (gráficos estatísticos, documentos, relatórios etc...) poderá realizar atividades que favoreçam a produtividade e análise de custos, prazos, além de gerenciar conteúdo, funcionários, usuários, bem como imprimir relatórios.

Sistemas similares:

Tenho Mais Discos Que Amigos - tenhomaisdiscosqueamigos.com é um site de conteúdo musical com foco em artistas nacionais, mas que também cobre lançamentos, turnês e novidades do cenário musical internacional. O site não possui concursos, sistema de cadastro nem conteúdo premium para assinantes.

American Songwriter - American Songwriter é uma revista bimestral, criada em 1984 cobrindo todos os aspectos do artesanato e da arte da composição. Possui entrevistas, dicas de composição, notícias, críticas e concursos de letras. A revista está sediada em Nashville, Tennessee.

Performing Songwriter - performingsongwriter.com e suas várias iniciativas de mídia se concentram em celebrar a música, iluminando os compositores, reforçando a importância da educação musical e artística e afirmando o valor da música e de seus criadores em todas as nossas vidas. O site também oferece workshops e conteúdo premium para compositores profissionais e aspirantes.

2.3 Suposições e dependências

As seguintes suposições são válidas no decorrer do desenvolvimento do sistema sendo especificado:

- O cliente está responsável pela aquisição de infraestrutura necessária em seu ambiente de produção;
- O cliente será responsável pela disponibilização de recursos de hardware, software, e outros requerimentos destinados à implantação do sistema desenvolvido.

3 Requisitos do Software

3.1 Requisitos Funcionais

<RF01>< Efetuar Login>

O sistema deve permitir que o usuário possa logar com suas credenciais (e-mail e senha).

<RF02>< Cadastrar Usuário>

Deve ser possível cadastrar usuários no sistema.

<RF03>< Editar Cadastro>

O sistema deve possibilitar que os usuários editem as suas próprias informações de cadastro.

<RF04>< Autenticação>

Deve ser permitido o acesso ao sistema por meio de autenticação.

3.2 Requisitos Não-funcionais

Os requisitos que descrevem os aspectos não-funcionais do sistema são apresentados a seguir:

3.2.1 Requisitos de Segurança

Ident.	Descrição	Casos de uso relacionados
RNF/SEG-01	O usuário autorizado deverá efetuar <i>login</i> no sistema para poder ter acesso ao conteúdo premium, se inscrever nos concursos e enviar composições	

3.2.2 Requisitos de Interface

Ident.	Descrição	Casos de uso relacionados
RNF/INT-01	O sistema deve ter uma interface amigável, de fácil utilização, ao seja, auto explicável.	

3.2.3 Requisitos de Operacionais

Ident.	Descrição	Casos de uso relacionados
RNF/OPE-01	O sistema deve ser desenvolvido em <i>Java</i> .	
RNF/OPE-02	O sistema deve ser desenvolvido em uma arquitetura em camadas.	
RNF/OPE-03	A camada de aplicação para <i>web</i> compatível com <i>browsers</i> de mercado (<i>Internet Explorer</i> , <i>Mozilla Firefox</i> e <i>Google Chrome</i>).	

3.2.4 Requisitos de Confiabilidade

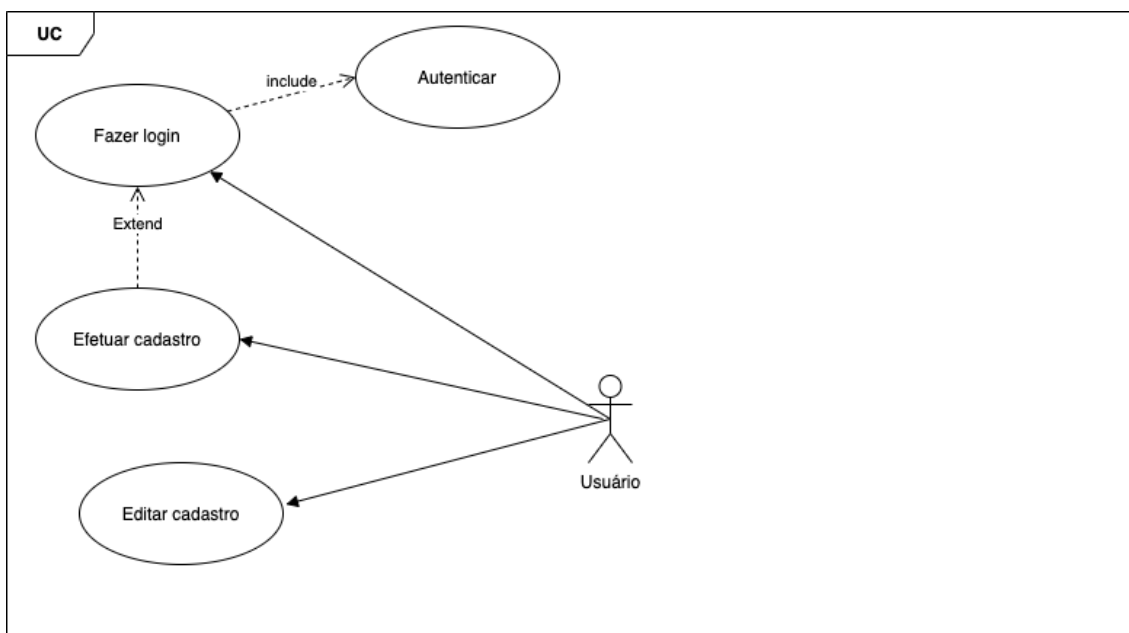
Ident.	Descrição	Casos de uso relacionados
RNF/CON-01	O sistema deve estar disponível 24 horas por dia durante os 7 dias da semana. Por não se tratar de um sistema crítico, poderá ficar fora do ar até que seja corrigida alguma falha que possa ocorrer.	

4 Casos de uso

4.1 Diagrama de casos de uso

O diagrama de casos de uso, expresso em UML (*Unified Modeling Language*), expressa os requisitos funcionais do sistema na forma de casos de uso. Segundo o RUP (*Rational Unified Process*), para cada requisito funcional tem-se um caso de uso. A descrição textual detalhada dos requisitos funcionais, seus fluxos de atividades e requisitos não funcionais associados pode ser encontrada na próxima seção. Na figura abaixo mostramos a representação gráfica em UML dos casos de uso do sistema.

4.1.1 Administrador e Usuário



4.2 Descrição dos casos de uso

Dentre os casos de uso do sistema mostrados no diagrama de casos de uso, foram escolhidos quatro para serem detalhados e trabalhados nas fases de análise e projeto do sistema.

4.2.1 Efetuar Cadastro

[CDU-01]	
Nome:	Efetuar cadastro
Atores:	Usuário
Prioridade:	Desejável
Requisitos associados:	<ul style="list-style-type: none">• [RF/SEG-02]
Entradas e pré-condições:	<ul style="list-style-type: none">• Não há
Saídas e pós-condições:	<ul style="list-style-type: none">• O usuário fica habilitado a ter acesso a áreas restritas do sistema.
Fluxos de eventos	
Fluxo principal:	<ol style="list-style-type: none">1. O usuário decide se cadastrar no sistema.2. O sistema solicita as informações obrigatórias para o cadastro(nome, canção, link da canção, email e senha).3. O ator informa os dados no formulário de cadastramento.4. O sistema valida os dados(authenticação).

	<ol style="list-style-type: none"> O sistema registra em histórico (log) a autenticação realizada pelo ator. O sistema habilita as ações relacionadas ao grupo de usuários ao qual pertence o ator. O sistema informa que a autenticação foi realizada com sucesso (ao se cadastrar o usuário automaticamente realiza o primeiro login).
Fluxo alternativo:	<ol style="list-style-type: none"> No passo 4 do Fluxo Principal, caso haja algum erro na autenticação relacionado aos dados informados: O sistema informa o erro ao ator. O fluxo retorna ao passo 2 do fluxo principal.

4.2.2 Efetuar Login

[CDU-02]		
Nome:	Efetuar login	
Atores:	Usuário	
Prioridade:	Desejável	
Requisitos associados:		<ul style="list-style-type: none">[RF/SEG-01]
Entradas e pré-condições:		<ul style="list-style-type: none">O usuário deve estar cadastrado no sistema.
Saídas e pós-condições:		<ul style="list-style-type: none">O usuário fica habilitado a ter acesso a áreas restritas do sistema.
Fluxos de eventos		
Fluxo principal:	<ol style="list-style-type: none">O usuário decide se autenticar no sistema.O sistema solicita as informações obrigatórias para a autenticação(login e senha).O ator informa os dados de autenticação.O sistema valida os dados de autenticação.O sistema registra em histórico (log) a autenticação realizada pelo ator.O sistema habilita as ações relacionadas ao grupo de usuário ao qual pertence o ator.O sistema informa que a autenticação foi realizada com sucesso.	
Fluxo alternativo:	<ol style="list-style-type: none">No passo 4 do Fluxo Principal, caso haja algum erro na autenticação relacionado aos dados informados:O sistema informa o erro ao ator.O fluxo retorna ao passo 2 do fluxo principal.	

4.2.3 Editar Cadastro

[CDU-03]		
Nome:	Editar as informações cadastradas	
Atores:	Usuário	
Prioridade:	Desejável	
Requisitos associados:	<ul style="list-style-type: none">[RF/SEG-03]	
Entradas e pré-condições:	<ul style="list-style-type: none">O usuário deve estar logado.	
Saídas e pós-condições:	<ul style="list-style-type: none">O usuário edita as suas informações cadastradas.	
Fluxos de eventos		
Fluxo principal:	<ol style="list-style-type: none">O usuário clica no botão “editar”.O sistema exibe a página de edição.O usuário edita as informações desejadas.O usuário clica no botão “salvar”.O sistema valida as informações.O sistema salva as novas informações no banco de dados.O sistema mostra uma mensagem de que a edição foi feita com sucesso.	
Fluxo alternativo:	<ol style="list-style-type: none">No passo 14 do Fluxo Principal, caso o usuário digite um username repetido, o sistema fará o tratamento da exceção e solicitará que o usuário tente alterar os dados novamente.	

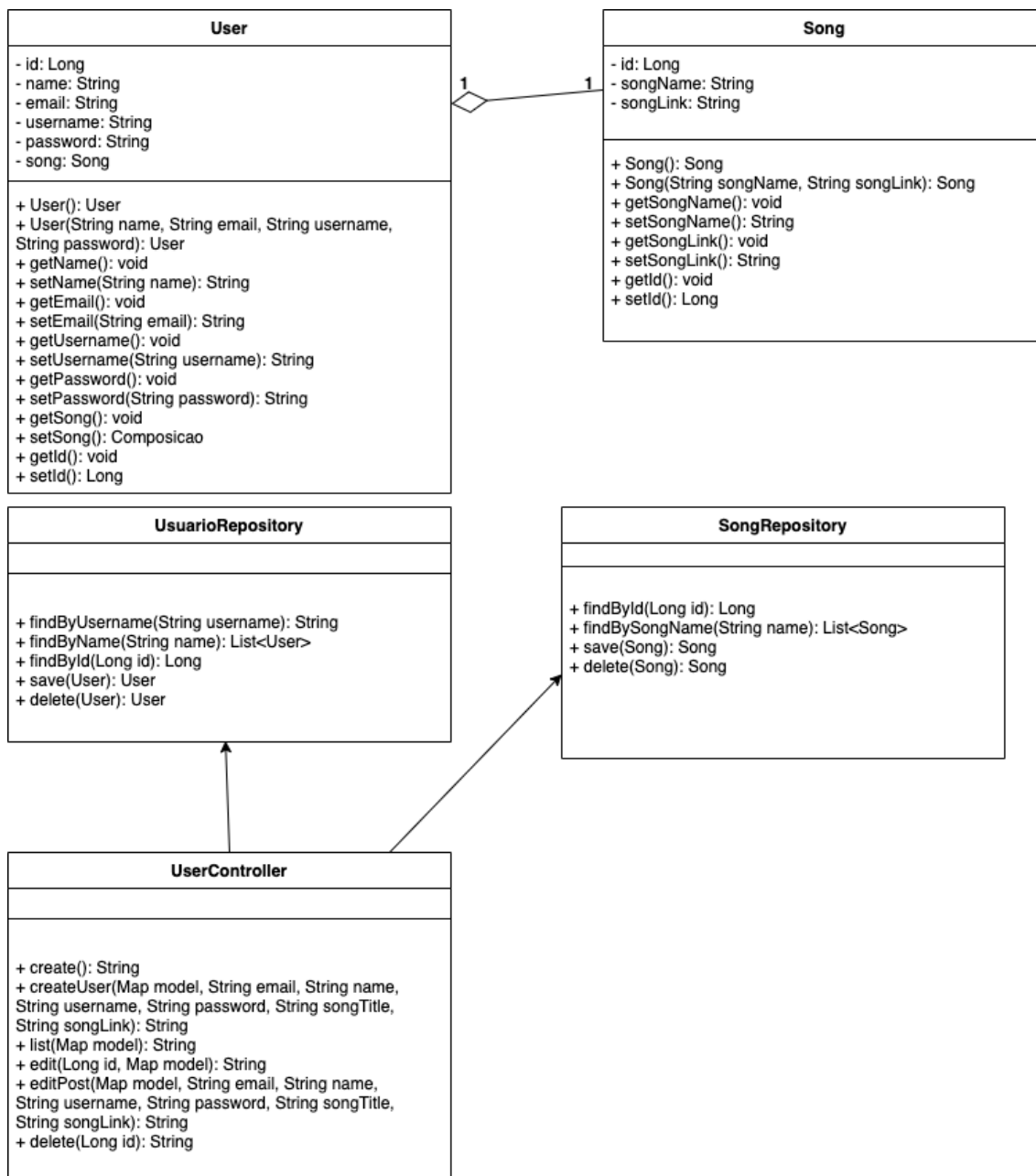
4.2.4 Autenticação

[CDU-04]

Nome:	Cadastrar composições em concursos		
Atores:	Usuário		
Prioridade:	Desejável		
Requisitos associados:	<ul style="list-style-type: none">• [RF/SEG-04]		
Entradas e pré-condições:	<ul style="list-style-type: none">• O usuário tenta realizar um login.		
Saídas e pós-condições:	<ul style="list-style-type: none">• O usuário é autenticado e tem acesso liberado ao sistema.		
Fluxos de eventos			
Fluxo principal:	<ol style="list-style-type: none">1. O usuário entra na página de login.2. O usuário preenche os campos necessários para autenticação.3. O sistema realiza a validação dos dados informados.4. O usuário é redirecionado para a página main do sistema.		
Fluxo alternativo:	<ol style="list-style-type: none">1. O usuário entra na página de login.2. O usuário preenche os campos necessários para autenticação.3. O sistema informa que os dados informados são inválidos.		

5 Classes do Sistema

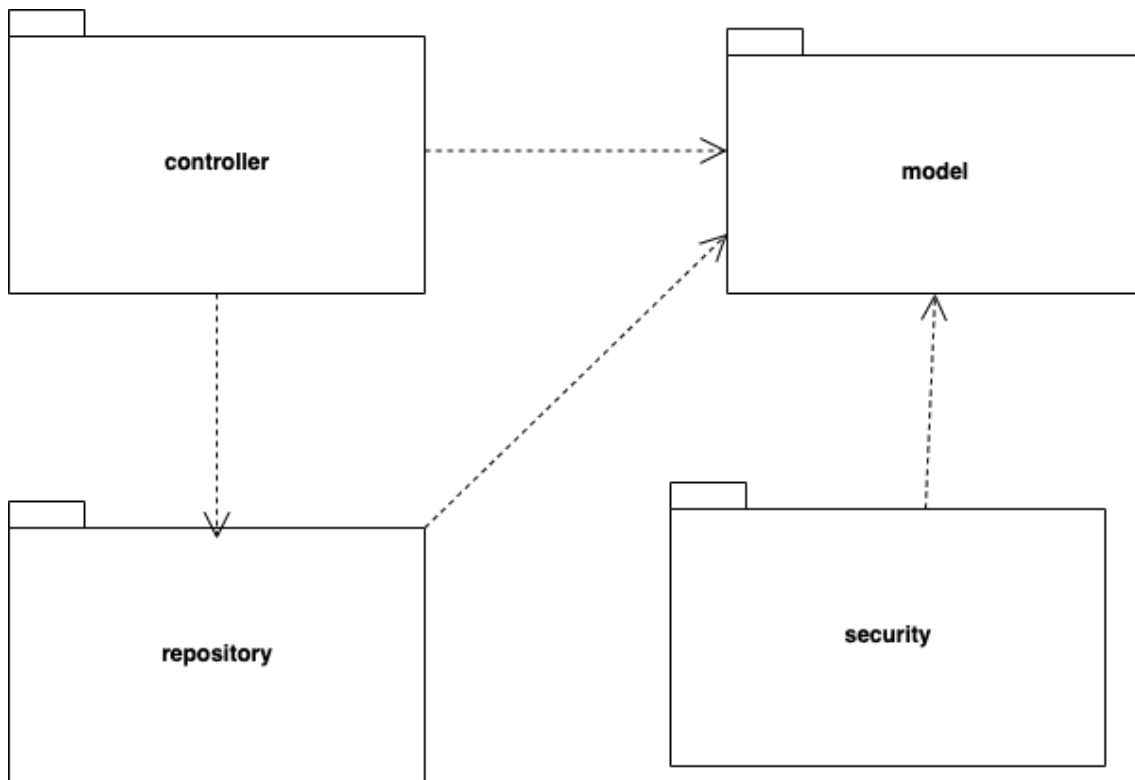
5.1 Diagrama de classes



5.2 Interação entre as classes

Ao iniciar a aplicação, a página index.jsp fornece um link através do botão "Enter Now" para a página de inscrição /user/create. É nessa página que a maior parte da lógica acontece. A classe UserController recebe os dados passados pelo formulário, instancia um objeto User e outro Song com seus respectivos atributos, e depois os salva nos repositórios UserRepository e SongRepository. Feita a criação do primeiro objeto, já é possível realizar o login e acessar as áreas restritas do sistema. Para realizar o login, a classe LoginController colhe os dados na página de login e os envia para o LoginInterceptor, que trata de verificar se há um usuário válido na sessão. Se houver, é dada a permissão para acessar as páginas de listagem e edição dos cadastros.

6 Diagrama de Pacotes



7 Interação Entre Objetos

7.1 Diagrama de sequência

