

Documento de detalhamento do Projeto: Sistema bibliotecário para Registro e Gerência de Impressos

Trabalho desenvolvido no contexto abaixo:

Disciplina: Projeto de Software
Professor: Marcelo R. Quinta
Curso: Ciência da Computação
INF - Instituto de Informática
UFG - Universidade Federal de Goiás
2º semestre/ 2017

Alunos participantes:

201507733 – Arnold Christian Veloso Lima
201514365 – Rafael Alessandro Alves Tomé
201507753 – Rafael Falcão Gomes Jardim

Resumo do documento:

Neste documento descreveremos o sistema evidenciando seus objetivos, as estruturas de dados utilizadas, funcionalidades disponíveis e a organização interna (padrão de projeto).

1. Descrição do sistema

Este software visa a gestão de um sistema de bibliotecário universitário no qual será possível gerir o acervo de livros, periódicos e revistas com a inclusão e exclusão de itens do acervo, empréstimos, reservas e gerar relatórios de suas atividades.

O código entregue deve ser capaz de através de um usuário, realizar pesquisas no acervo, a retirada (empréstimo) de um impresso e sua devolução com êxito. Porém, o **projeto em sua versão final deve ser capaz**, além do já citado, realizar com êxito reservas (de um impresso), cadastro de novos usuários/impressos/exemplares no banco de dados, e a aplicação de multas automatizadas em clientes retardatários em suas devoluções.

2. Definição dos Requisitos

Na versão disponibilizada os seguintes requisitos funcionais deverão ser atingidos com êxito.

Funcionalidade do sistema:

- [RF001] O usuário deverá ser capaz de logar-se no sistema, através de uma interface de comunicação com o usuário.
- [RF002] O usuário deverá ser capaz de realizar buscar de impressos cadastrados no acervo através de uma palavra-chave.
- [RF003] O sistema deverá ser capaz de listar, por meio de uma interface de comunicação com o usuário, os impressos resultantes de uma pesquisa [RF002], contendo: o ID do impresso, título, autores e Editora.
- [RF004] O usuário deverá ser capaz de requisitar ao sistema que liste os exemplares disponíveis de um determinado impresso.
- [RF005] O usuário deverá ser capaz de requisitar ao sistema o empréstimo de um determinado exemplar, através de uma interface de comunicação com o usuário.
- [RF006] O usuário deverá ser capaz de requisitar ao sistema a devolução de um empréstimo realizado anteriormente, através de uma interface de comunicação com o usuário.
- [RF007] O sistema deverá ser capaz guardar informações dos empréstimos realizados pelos usuários [RF005].
- [RF008] O sistema deverá ser capaz diferenciar exemplares retirados de exemplares disponíveis.
- [RF009] O sistema deverá ser capaz de realizar relatórios dos empréstimos realizados pelos usuários [RF007].
- [RF010] O resultado da pesquisa do [RF003] deve conter informações do Título do impresso, seu ID e seu tipo (Livro, periódico ou revista).

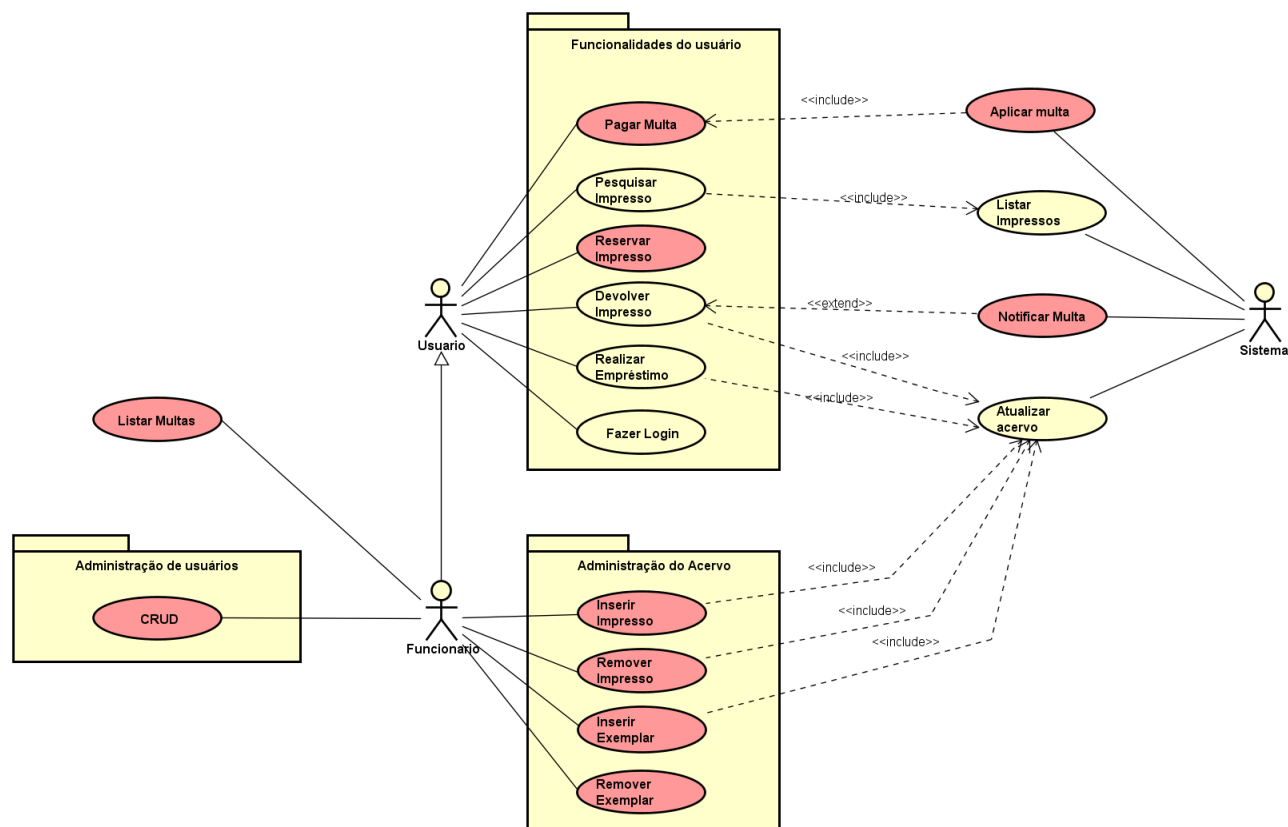
- [RF011] O relatório realizado no [RF009], deve ser capaz de mostrar o usuário que realizou o empréstimo, e o exemplar retirado.
- [RF012] O sistema deverá ser capaz de listar os empréstimos realizados pelo usuário, através de uma interface de comunicação com o usuário.
- [RF013] A listagem realizada no [RF012] deve conter: o exemplar retirado, e a data da retirada.
- [RF014] As informações requisitadas para a realização do login [RF001] serão o número de matrícula do usuário e sua senha vinculada.

O projeto disponibilizado não será capaz de realizar cadastro de usuários, novos impressos e exemplares no banco de dados através de uma interface de comunicação com o usuário, porém será disponibilizado um usuário padrão, um acervo com vários impressos e alguns exemplares. Novos cadastro não será possível no projeto disponibilizado, pois, essa etapa seria normalmente realizada por um usuário com poderes de acesso especiais, como o funcionário (mostrado no diagrama de casos de uso). **Atualmente os cadastros são possíveis apenas se forem realizados diretamente no banco de dados** que compõe o projeto.

Toda as informações do sistema serão guardadas em um banco de dados em tempo real, o projeto do banco pode ser encontrado na secção 3. Outras funcionalidades não disponíveis poderão ser visualizadas no diagrama de Casos de uso disponível na secção 3.

3. Definição da Arquitetura Preliminar

Diagrama de casos de uso:

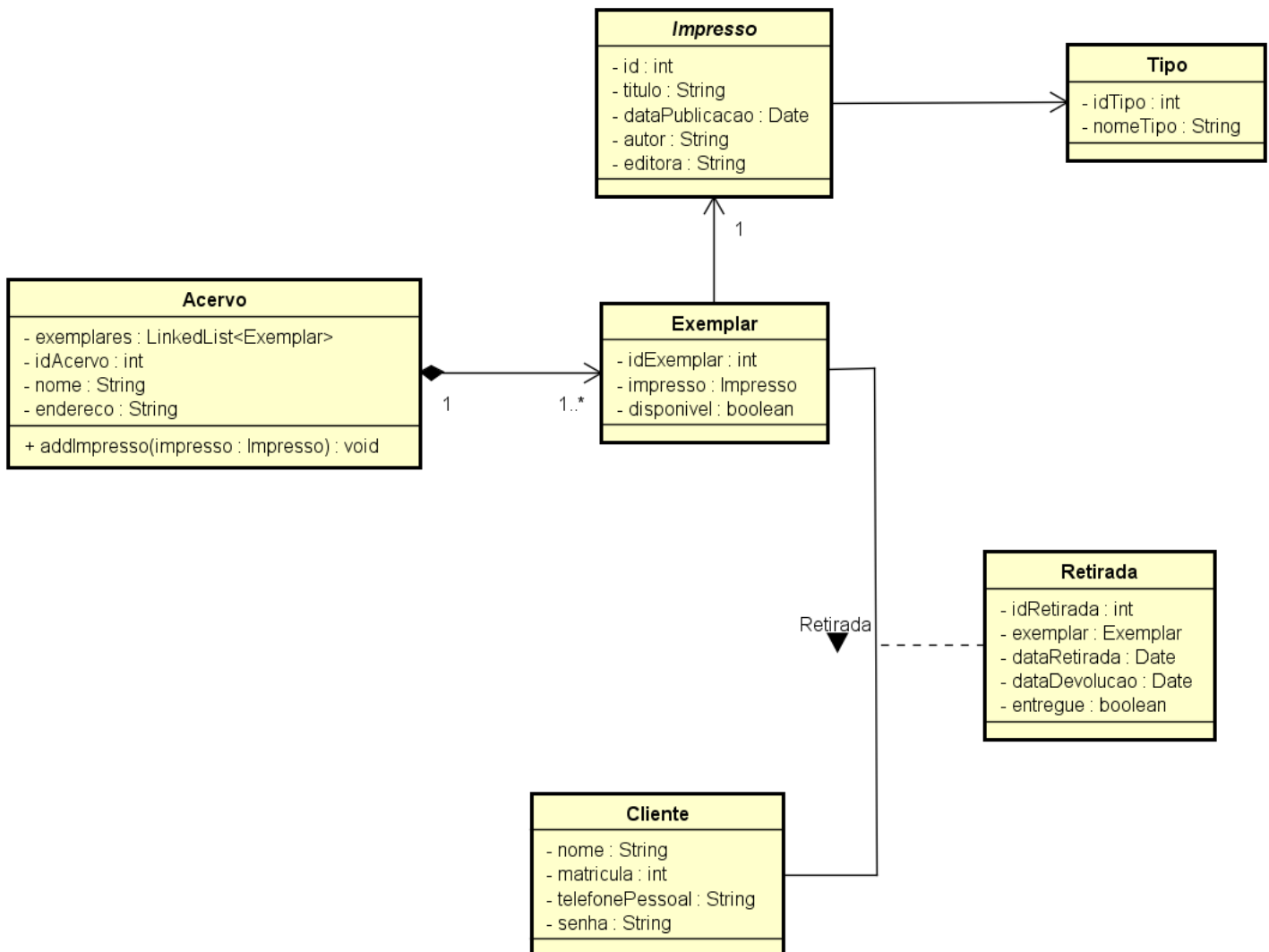


powered by Astah

Através do diagrama de casos de uso é possível ter uma breve visão das funcionalidades que o **usuário** deverá ser capaz de realizar, o **funcionário** (abstração de uma secretária da biblioteca) e o **sistema** e as relações entre eles.

Nesse documento é possível ter uma breve visão das funcionalidades disponíveis pelo sistema, onde os casos de uso de cor avermelhada não estarão disponíveis na versão disponibilizada, porém o sistema possuirá a capacidade de adequar à expansão, para projetos futuros.

Diagrama de Classes:

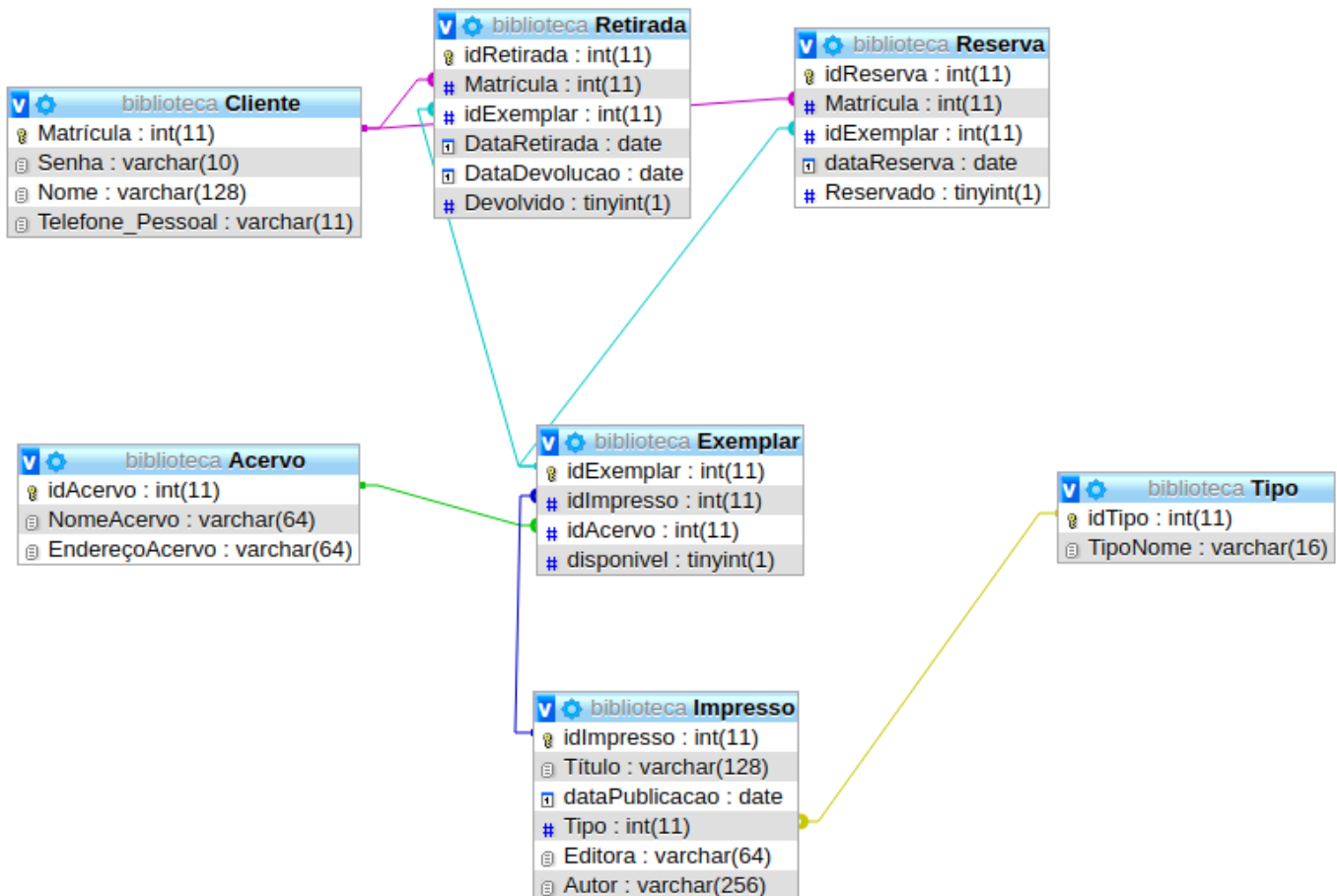


Através do diagrama de classes é possível ter uma breve visão e como são tratados os dados dentro do sistema e no código disponibilizado.

A classe “**Acervo**” conterá informações como o nome, id, endereço e os exemplares associados a ela. Por sua vez, a classe de “**Exemplar**” conterá informações como id, a disponibilidade (se o exemplar pode ser retirado em outro empréstimo) e o impresso associado. A classe “**Impresso**” conterá informações concretas do Exemplar, como seu Título, data de publicação, editora, autores, id e o seu tipo. A classe “**Cliente**” conterá informações sobre o nome do usuário, matrícula, telefone pessoal e sua senha de acesso ao sistema. A classe associada “**Retirada**” corresponde em uma relação entre cliente e exemplar, e é a classe que conterá informações do empréstimo realizado pelo

usuário e conterá informações como o id, o exemplar associado, o cliente associado, as datas de retirada e devolução e a informação no caso do empréstimo já ter sido finalizado (o usuário ter entregue o exemplar de volta na biblioteca).

Esquema Relacional do Banco de Dados:

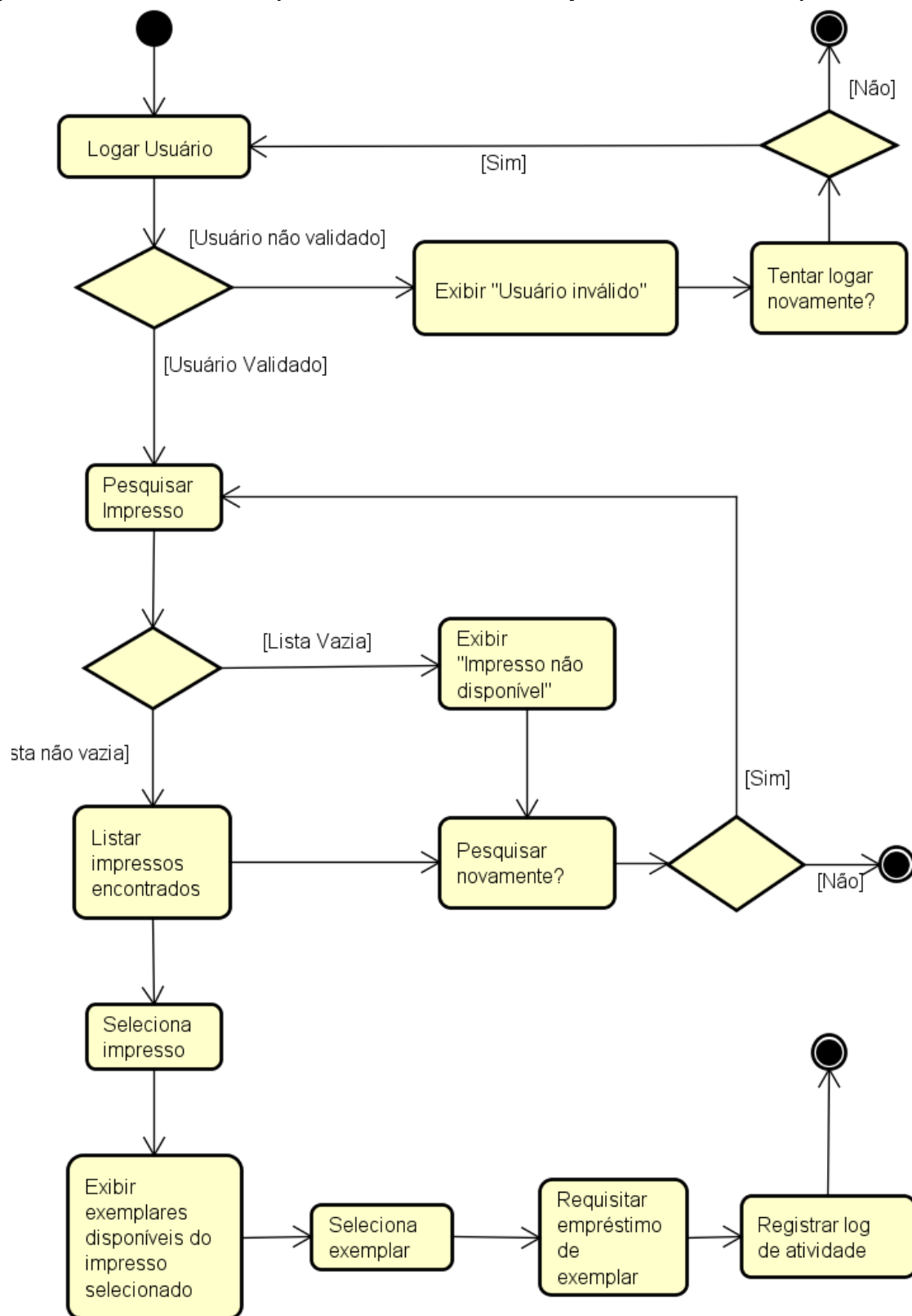


Através do esquema relacional do banco de dados, é possível ter uma visão clara de como são tratados e armazenados os dados do sistema no banco de dados e as relações entre as tabelas do esquema relacional.

Os relatórios usarão basicamente utilizarão informações da tabela “**Retirada**”, que conterá informações de todos os empréstimos realizados pelos usuários. A tabela “**Cliente**” conterá todos os usuários cadastrados no sistema. A tabela “**Exemplar**” conterá informações sobre todos os exemplares cadastrados no sistema, ela também contém informações sobre o impresso associado. A tabela “**Impresso**” por sua vez, conterá todos os impressos cadastrados no sistema, e cada impresso por sua vez está relacionado com um **Tipo**, os tipos cadastrados no sistema são: Livro, periódico e revista.

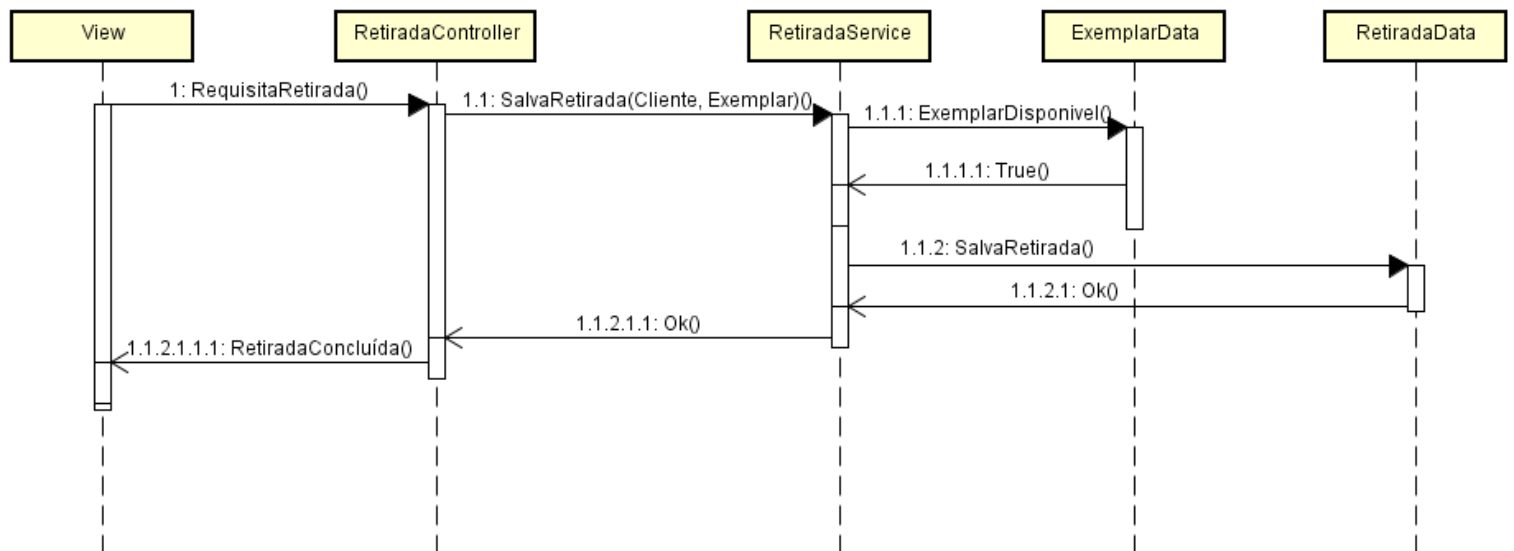
O sistema virá com um **usuário padrão cadastrado**, com a finalidade de possíveis testes, sua matrícula e senha de acesso são 201700001 e aluno123, respectivamente. E além disso, alguns livros e exemplares também estarão cadastrados no sistema.

Diagrama de Atividades (Realização de um empréstimo/retirada):



O diagrama de atividades acima, exemplifica os passos seguidos pelo algoritmo durante a requisição de um empréstimo pelo usuário.

Diagrama de Sequência:



Através do diagrama de sequência mostrado acima, é possível ter uma visão geral de como ocorre a transição de mensagens dentro do sistema para a realização de uma requisição de retirada (ou empréstimo) de um exemplar.

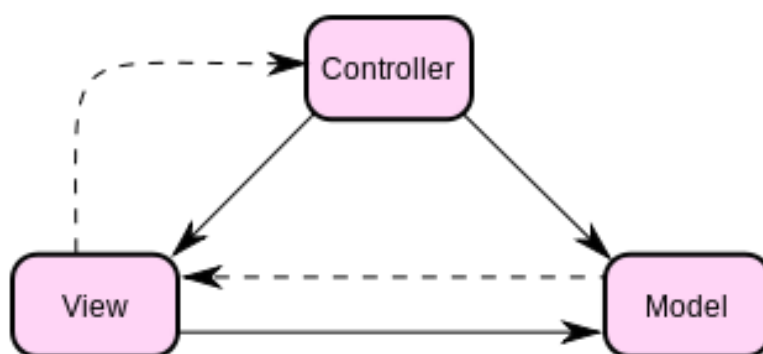
A **View** deve ser capaz de comunicar com o **RetiradaController**, avisando que uma requisição de retirada foi realizada pelo usuário. O **RetiradaController** por sua vez, envia os dados referentes a retirada para o **RetiradaService**, que por sua vez realiza consultas no banco de dados se comunicando com o **ExemplarData**, que consultará se o exemplar requerido está disponível, e o **RetiradaData**, que irá propriamente dito salvar as informações da retirada no banco de dados.

4. Descrição do padrão de arquitetura utilizado:

O projeto utilizará o padrão arquitetural conhecido como **Model-view-controller (MVC)**.

É um padrão de arquitetura de software que separa a representação da informação da interação do usuário com ele. O **modelo** (model) consiste nos dados da aplicação, regras de negócios, lógica e funções. Uma **visão** (view) pode ser qualquer saída de representação dos dados, como uma tabela ou um diagrama. O **controlador** (controller) faz a mediação da entrada, convertendo-a em comandos para o modelo ou visão.

As ideias centrais por trás do MVC são a reusabilidade de código e separação de conceitos.



5. Estimativas de horas de desenvolvimento baseado em horas:

Projeto do sistema (diagramas) → 8 horas

Projeto do banco de dados -> 8 horas

Projeto de interface de comunicação com o usuário -> 8 horas

Pesquisar e Listar impresso -> 2 horas

Listar empréstimos realizados e Devolver impresso -> 2 horas

Realizar empréstimo -> 2 horas

Fazer login -> 2 horas

(Também contabilizadas o tempo de adequação com a interface e o banco de dados)

Inserir impresso -> *Tempo faz parte do projeto do banco de dados

Remover impresso -> *Tempo faz parte do projeto do banco de dados

Tempo total estimado → 32 horas.