



CAMPUS SOLEDADE  
TÉCNICO SUBSEQUENTE EM INFORMÁTICA

ANDRESSA KETHILLY PALMEIRA SOUTO  
CAROLINE DE OLIVEIRA CORDEIRO

SOLEDADE/PB,  
Junho de 2022

## RELATÓRIO DO PROJETO

Projeto apresentado como conclusão da disciplina de Projeto Integrador do curso Técnico Subsequente em Informática do Instituto Federal da Paraíba (IFPB), Campus Soledade.

Orientador: Prof. Dr. Israel Aires Costa Leal

## RESUMO

Este projeto tem como objetivo trabalhar o desenvolvimento de um sistema que automatize as funções realizadas na gestão de uma biblioteca, como empréstimos e devoluções de livros, por meio de um banco de dados que mantém registro de alunos, funcionários e livros, e que possa proporcionar ao estudante uma forma fácil de estar conectado com o acervo da biblioteca em questão.

A problemática aqui trabalhada é uma necessidade real da biblioteca do IFPB - Campus Soledade, que não tem ainda um sistema eletrônico de gerenciamento e que sofre com as consequências desse fato, sendo ela por exemplo, a impossibilidade de realizar empréstimos dos livros. A partir disso, foi feita a análise das necessidades do sistema e planejamento dele a partir de técnicas de Engenharia de Software, como por exemplo o uso de diagramas UML.

Após o planejamento, o desenvolvimento do trabalho foi iniciado e documentado, em que foram utilizadas as linguagens HTML e CSS no *front-end* e PHP e SQL no *back-end*. Uma parte do projeto foi desenvolvida e foram apresentadas propostas de trabalhos futuros para continuidade do projeto.

Esse projeto tem sua importância não somente para a biblioteca usada como objeto de pesquisa, como também para outras instituições de ensino que tenham necessidades semelhantes e queiram usar a base criada aqui. A partir do sistema básico gerado por essa pesquisa, podem ser feitas alterações e melhorias para engrandecimento do sistema.

**Palavras-chave:** Sistema para Biblioteca, Biblioteca, Automação de Serviços.

## **ABSTRACT**

## **SUMÁRIO**

<b>INTRODUÇÃO</b>	<b>6</b>
<b>FUNDAMENTAÇÃO TEÓRICA</b>	<b>7</b>
<b>METODOLOGIA</b>	<b>9</b>
<b>RESULTADOS</b>	<b>18</b>
<b>CONSIDERAÇÕES FINAIS</b>	<b>24</b>
<b>TRABALHOS FUTUROS</b>	<b>24</b>
<b>REFERÊNCIAS</b>	<b>25</b>

## 1. INTRODUÇÃO

As funções de organizar e gerir bibliotecas que recebem dezenas de alunos toda semana, haja vista que nem sempre é uma tarefa simples ou rápida, necessita de comunicação, organização e vigilância. Fazer registros, fichas e cadastros à mão demanda tempo e espaço, visto que deve-se ter lugares específicos e seguros para guardar os dados referente a livros e pessoas. Com um sistema de computador a manipulação, segurança e organização desses dados estaria mais garantida, pois as informações estariam guardadas e organizadas de modo a facilitar a vida dos funcionários da biblioteca, uma vez que apenas eles poderiam consultar e modificar esses dados.

Observando que a biblioteca do Instituto Federal da Paraíba (IFPB) Campus Soledade não utiliza um sistema e vendo as implicações geradas por essa situação, nota-se a demanda de um software que resolva esses problemas. A partir dessa demanda, foram observadas as necessidades básicas tanto dos usuários como dos funcionários da biblioteca, e assim foi estruturado o projeto de um Sistema Bibliotecário. Embora o sistema seja baseado no dia a dia desse campus em específico, ele segue padrões que podem se estender para outras instituições.

O sistema bibliotecário proposto é uma aplicação *web* integrada inicialmente para o IFPB Campus Soledade, em que os procedimentos realizados na interação do sistema são do funcionário e aluno. As funções de cadastrar livros, cadastrar alunos e realizar empréstimo são feitas somente pelo funcionário, tornando a movimentação da biblioteca mais segura, o aluno participa apenas de algumas informações atribuídas a ele, mas não tem a autonomia de realizar o empréstimo sozinho, por exemplo. Já o acervo é uma entidade que o aluno está apto para utilizar sem a intervenção do funcionário, na qual ele pode fazer login e verificar a disponibilidade dos livros e participar de lista de espera, caso o livro não esteja disponível.

Este documento está organizado da seguinte forma: no Capítulo 2 é descrita a fundamentação teórica e o estado da arte da pesquisa em questão; no Capítulo 3 estará descrita a metodologia do projeto e suas etapas de desenvolvimento, desde os diagramas UML até as páginas desenvolvidas até agora; no Capítulo 4 estão os resultados obtidos; e finalmente, as considerações finais deste trabalho são descritas no Capítulo 5.

## 2. FUNDAMENTAÇÃO TEÓRICA

Durante o planejamento do projeto, alguns processos devem ser especificados para um bom desenvolvimento. Sabendo o tipo de trabalho que viria a ser desenvolvido e as necessidades e objetivos do projeto, alguns tópicos foram marcados como de extrema importância para serem estudados, entre eles a Engenharia de Software, a linguagem UML (Unified Modeling Language) e a interação entre desenvolvimento Web e Banco de Dados.

A Engenharia de Software é uma área voltada para a criação de soluções com o desenvolvimento de sistemas, em que é analisado, especificado, implementado e validado uma determinada situação real, de modo a facilitar o trabalho das pessoas. Nesse sentido, existe todo um processo a ser desenvolvido para que o sistema tenha um bom desempenho, aplicando os conceitos e padrões definidos em cada situação nas etapas do desenvolvimento.

De acordo com Sommerville (2007, p.15), um sistema é um conjunto de componentes (hardware, software, processos, organização e operadores) inter relacionados que funcionam juntos para atingir certo objetivo. Seu comportamento não depende apenas da associação que foi atribuída ao sistema, mas sim dos indivíduos e objetivos aos quais vão ser interpretados, então o funcionamento com sucesso de cada componente depende do funcionamento dos outros componentes. Um ponto importante na implementação do sistema é estabelecer os requisitos, para que seja evidente o seu comportamento e direcionamento do que fazer em determinada atividade, como também restringir ações para que suas propriedades de disponibilidade, desempenho e segurança apresentem sinais positivos. A insatisfação do desempenho confiável do sistema pode surgir em diferentes situações, seja o hardware que apresente falhas, softwares com mau processamento ou um indivíduo em um dia com estresse, ocasionará inesperados erros de processamento e dados.

Fatores humanos, sociais e organizacionais são frequentemente críticos para determinar se o sistema atende aos seus objetivos com sucesso ou não. Todo o conhecimento relevante de uma organização deve ser incluído na especificação de sistema de tal forma que os projetistas de sistema possam levá-lo em consideração. (SOMMERVILLE, 2007)

Buscando por uma forma de modelagem simples e direta que servisse de suporte na hora de planejar o sistema proposto, foram usados diagramas que fazem parte da linguagem visual UML – Unified Modeling Language ou Linguagem de Modelagem Unificada. A UML é uma linguagem que busca auxiliar o engenheiro a fazer toda a preparação inicial do projeto de software. Segundo Guedes(2011) o objetivo da UML é:

[...] auxiliar os engenheiros de software a definirem as características do sistema, tais como seus requisitos, seu comportamento, sua estrutura lógica, a dinâmica de seus processos e até mesmo suas necessidades físicas em relação ao equipamento sobre o qual o sistema deverá ser implantado.(GUEDES, 2011)

Além disso, a modelagem de um sistema é também uma parte da documentação deste, sendo importante para o futuro do produto, onde atualizações e correções de erros serão necessárias e requererão uma documentação detalhada e precisa do sistema. E justamente por prezar o detalhamento do sistema que a linguagem UML tem tantos diagramas diferentes. Cada diagrama visará representar um lado diferente do sistema , mostrando diferentes faces do produto e ajudando a detectar necessidades, erros e características do sistema que antes possam ter passado despercebidas.

A importância também do trabalho colaborativo entre as linguagens PHP e SQL é notável neste e em outros projetos de sistemas para web. Desenvolver o *back-end* do sistema torna-o um site dinâmico, proporcionando diferentes experiências para cada usuário por meio de um mesmo sistema básico, e a escolha da linguagem usada interfere diretamente em várias áreas do projeto, como por exemplo os prazos e o financeiro. Foi ao observar isso que foram escolhidos PHP e MySQL para esse trabalho. Como defendido pelo professor Evaldo Junior Bento (2021), a união de PHP e MySQL é vantajosa pela facilidade com que o PHP conversa com bancos de dados e pelas bibliotecas de acesso ao MySQL já embutidas na linguagem PHP, além de que ambas as ferramentas são tecnologias gratuitas e tem serviços de hospedagem mais baratos se comparados a outros.



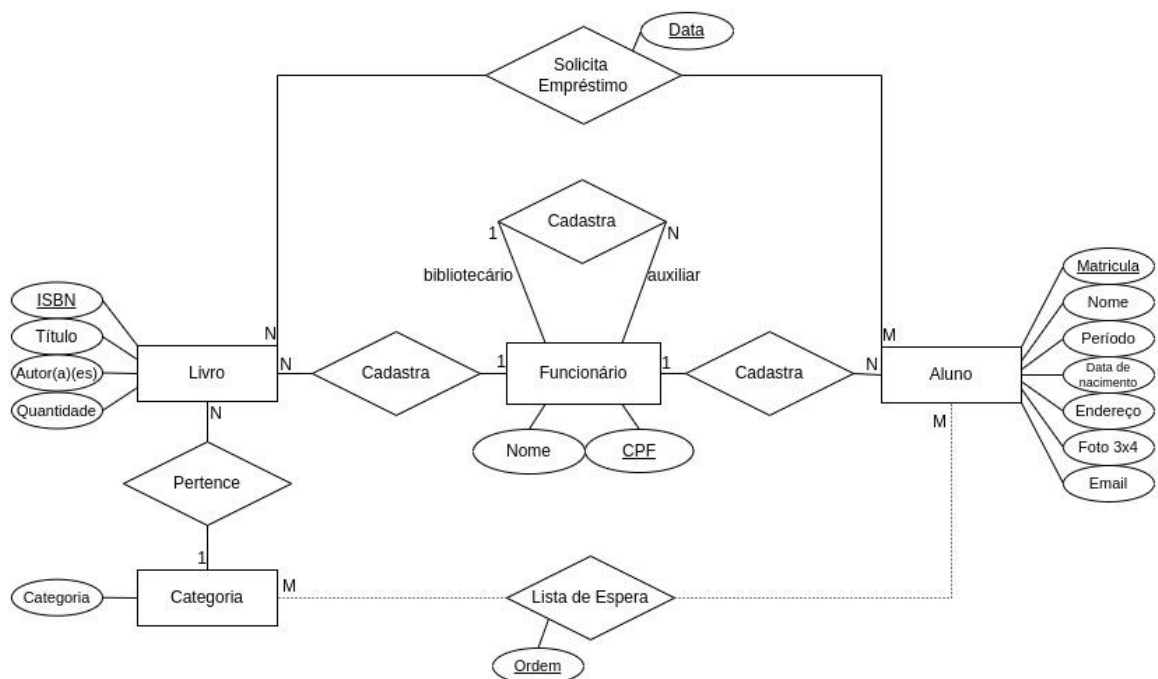
### **3. METODOLOGIA**

Tendo em vista que o projeto é de uma plataforma Web simples e de fácil acesso, foram usadas principalmente duas abordagens ensinadas no decorrer do curso de Informática do IFPB: um Banco de Dados, feito e manipulado por meio de linguagem SQL e do MySQL, e uma estrutura para Web, criada com HTML e PHP, além de CSS na parte visual.

Na página web existem dois usuários principais: o aluno, que acessa a partir de um login e senha próprios, e pode a) visualizar assim os livros existentes na biblioteca, bem como b) ver suas informações pessoais registradas no cadastro e seu histórico como usuário da biblioteca e c) solicitar a entrada na lista de espera para empréstimo de um livro; e os funcionários da biblioteca, que terão as funções de a) administração do acervo online, sendo os responsáveis por cadastrar os livros que chegam, b) cadastramento de novos alunos, c) realização de empréstimos dos livros e também d) cadastramento de outros funcionários secundários, que pode ser realizada pelo funcionário principal da biblioteca.

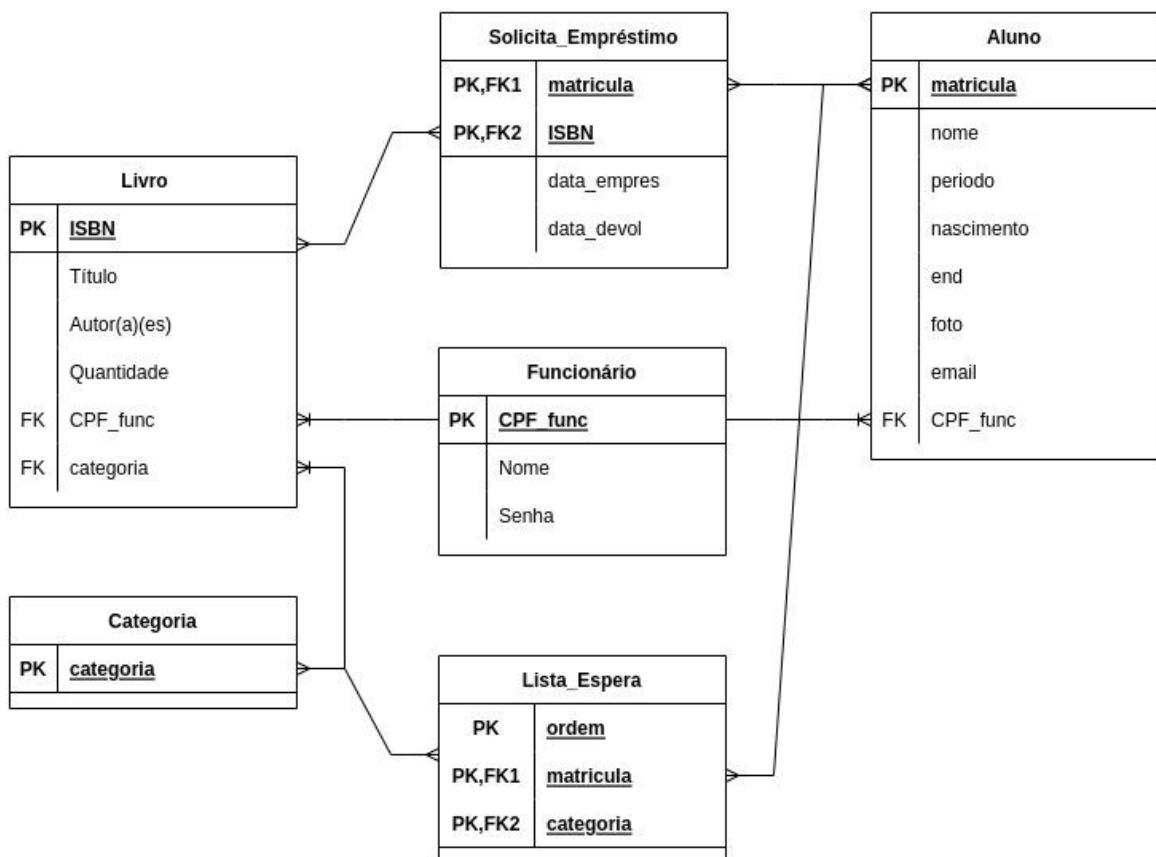
O projeto foi desenvolvido com base em um banco de dados já existente, que foi resultado da disciplina de Banco de Dados ministrada no curso. Ele foi adaptado para as necessidades vistas no campus e a partir dele foi feito todo o estudo para a parte de Desenvolvimento de Aplicações Web do projeto. Na próxima página podemos ver o desenvolvimento do estudo, composto por: Diagrama Entidade Relacionamento (Figura 1); Modelo Relacional (Figura 2); Diagrama de Caso de Uso (Figura 3); Diagrama de Classe (Figura 4); Diagrama de sequência (Figura 5, 5.1 e 5.2).

Na Figura 3.1 está representado o Modelo Entidade Relacionamento, que é o ponto de partida do Banco de Dados (BD), na qual temos a visão geral do sistema com seus respectivos relacionamentos, entidades e atributos descritos da seguinte forma: a entidade Funcionário cadastra vários livros, informando seus atributos, bem como cadastra também vários alunos, informando seus atributos e por fim cadastra outro funcionário, criando um auto-relacionamento. Observando a entidade Aluno, muitos alunos podem solicitar empréstimo de vários livros, como também podem entrar na lista de espera, que são divididas pelas categorias existentes de livros no acervo.



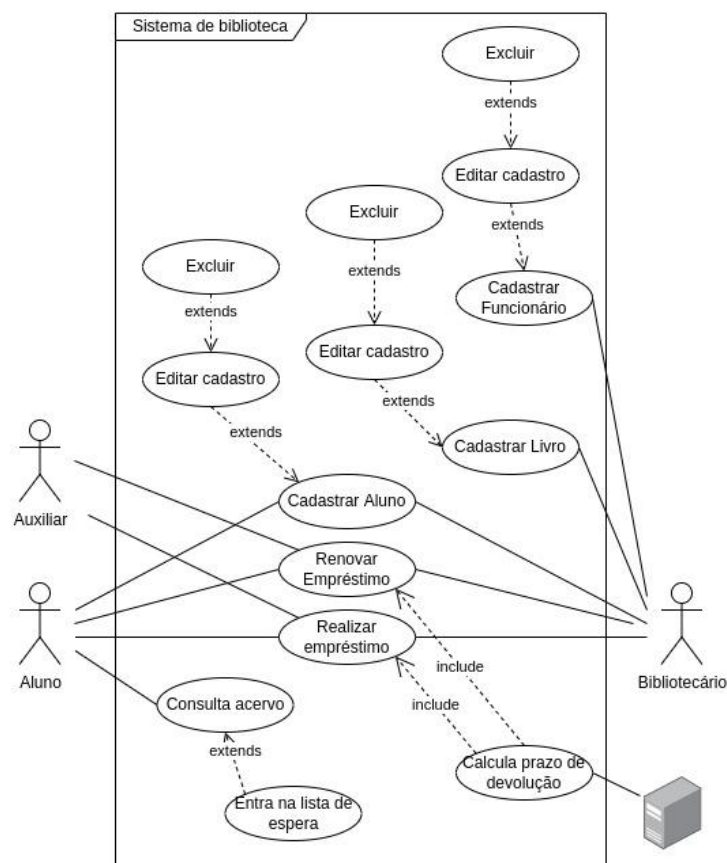
**Figura 3.1: Modelo Entidade Relacionamento**

Na Figura 3.2, é mostrado o Modelo Relacional, que também é parte do projeto do BD. Ele foi feito a partir do Modelo Entidade Relacionamento (Figura 1) e nele pode-se ver a estrutura das tabelas que serão criadas com SQL. Nele temos a identificação das Chaves Primárias (*Primary Key* - PK) e Estrangeiras (*Foreign Key* - FK) de cada tabela, e temos a associação entre as tabelas representadas pelas setas de notação de James Martin, seja com o símbolo de “pé-de-galinha” dos dois lados, representando um relacionamento de muitos-para-muitos, ou apenas de um lado, representando um-para-muitos. Essas setas indicam que alguma FK foi criada dentro de uma dos diagramas. Além de PKs e FKs, pode-se observar os atributos que não são nenhum tipo de chave. Como os atributos dos diagramas provém da Figura 1, não há necessidade de nos estendermos na explicação.



**Figura 3.2: Modelo Relacional**

Na Figura 3.3, é apresentado o Diagrama de Casos de Uso que tem como atores: o Aluno, o Bibliotecário (funcionário principal), o funcionário Auxiliar e o Sistema Automático. Começando pelo Sistema: ele é responsável por calcular o prazo de devolução do livro assim que o empréstimo for confirmado; por esse ser um caso do tipo *include*, ele é de execução obrigatória ao se executar os dois casos de uso aos quais ele está ligado: “Realizar Empréstimo” e “Renovar Empréstimo”. Para os dois casos citados anteriormente, temos como atores todos com exceção do sistema; nesses casos, o aluno apenas participa da ação solicitando ao funcionário em questão que exerça o comando, mas ele em si não atua no sistema. Já “Consulta Acervo” é executado pelo aluno através do sistema; esse caso tem uma função *extends*, um caso que é facultativo e que está atrelado a outro. Nessa função *extends* o aluno pode entrar na lista de espera para o empréstimo de um livro que não está disponível no momento. Os casos “Cadastra aluno”, “Cadastra livro” e “Cadastra funcionários” são similares: todos são executados apenas pelo Funcionário principal da biblioteca e tem as funções *extends* “Editar Cadastro” e “Excluir”.



### **Figura 3.3: Diagrama de Casos de Uso**

Na Figura 3.4, é mostrado o diagrama de classe, que é uma representação focada nos métodos de cada atributo que o sistema é capaz de realizar, de acordo com o levantamento de requisitos prévios nos demais diagramas criados. As simbologias denominadas na figura consistem em (#) Protegida, (+) Pública e (-) Privada. Ademais, as comunicações utilizadas entre as classes são: Herança, Composição, Associação e Associação Unitária. No topo está a classe pessoa, que detém os atributos mais gerais das classes herdeiras, aluno e funcionário. Na classe funcionário estão todos os métodos designados para cada função específica das demais classes, assim como na do aluno. A Classe empréstimo é uma composição de livro e funcionário – o empréstimo só irá existir se o funcionário realizar em um livro disponível – e a classe lista de espera é uma composição de livro e aluno – a lista de espera só irá existir se o aluno escolher um livro válido para essa operação. Observe que o livro está associado às duas classes em que ele faz parte na composição, de empréstimo e lista de espera, ou seja, a instância livro existe independente de aluno ou funcionário e vice versa. Por fim, existe uma associação unitária na classe funcionário e cadastra, que pertencem à mesma classe.

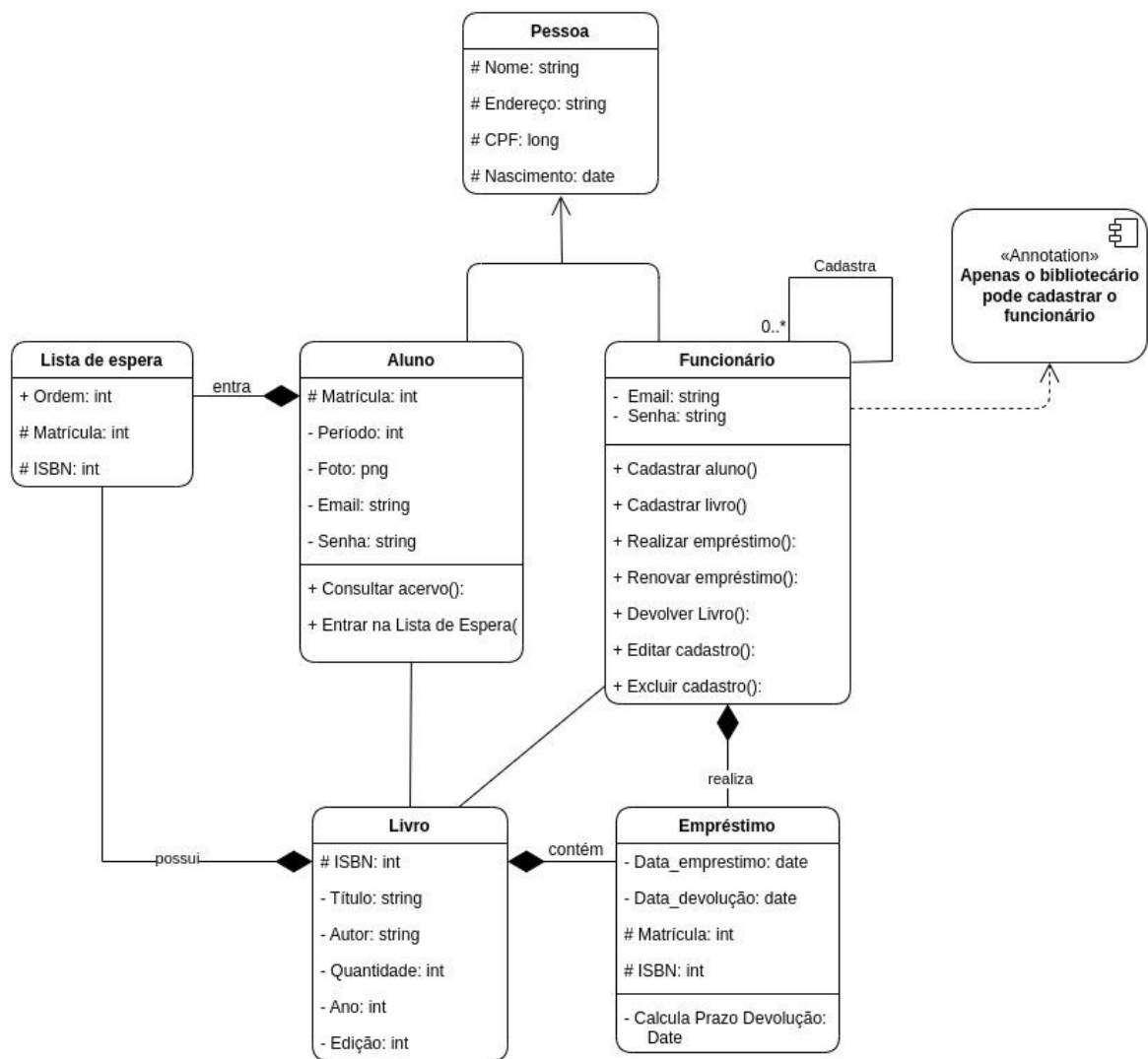
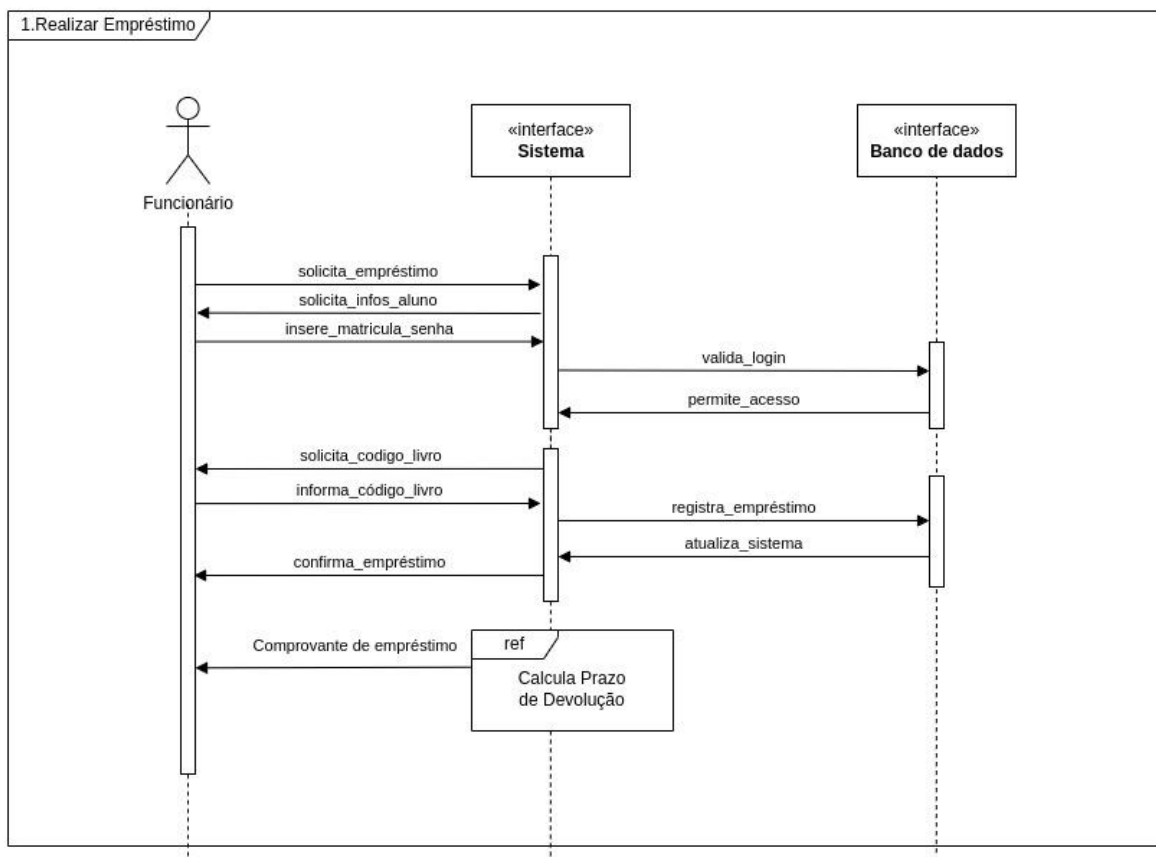


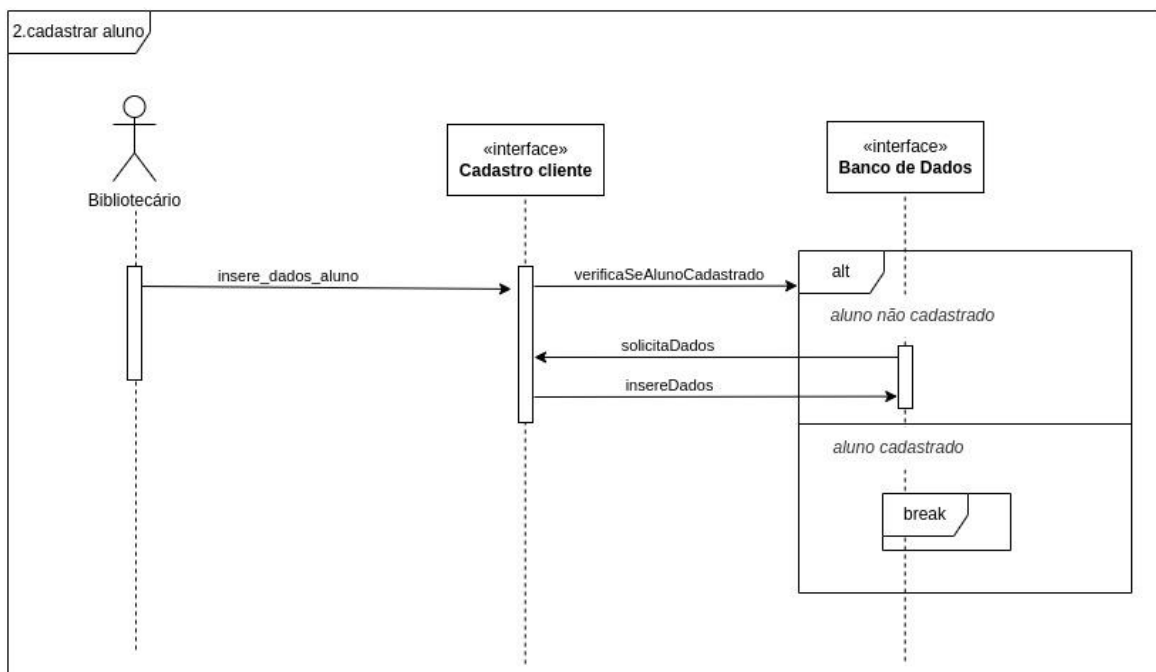
Figura 3.4: Diagrama de Classes

Na Figura 3.5.1 é apresentado o diagrama de sequência referente a Realizar Empréstimos. O Funcionário solicita empréstimo ao Sistema, o Sistema solicita os dados do aluno ao Funcionário. Ele insere os dados e o Sistema valida-os no banco de dados. O Sistema vai então solicitar o código do livro desejado, o Funcionário informa o código e assim o empréstimo será registrado, e quando o funcionário confirmar a ação, é calculado o prazo de devolução e gerado um comprovante com os dados do aluno, livro e datas (de empréstimo e devolução).



**Figura 3.5.1: Diagrama de Sequência referente a Realizar Empréstimos**

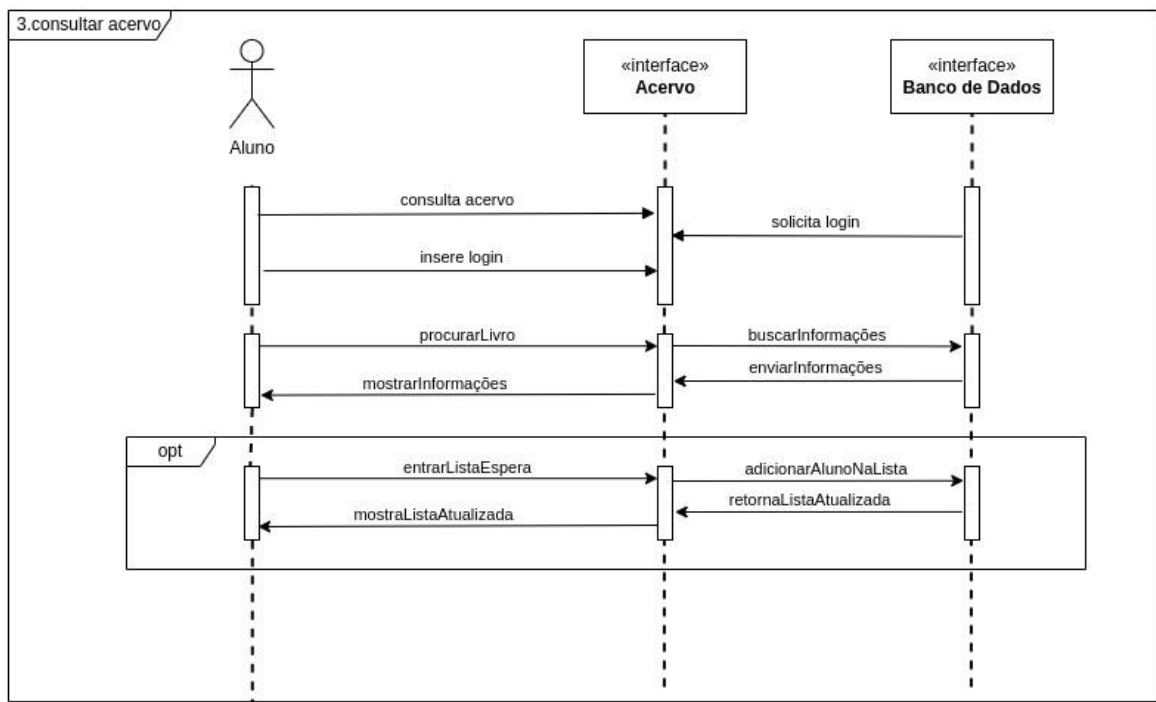
Na Figura 3.5.2 é exibido o diagrama de sequência referente a Cadastrar Alunos, que tem como atores o Bibliotecário e as interfaces Cadastro Cliente e o Banco de Dados do sistema. Para realizar um cadastramento, o bibliotecário deve inserir os dados do aluno na tela de cadastramento e aguardar enquanto o sistema verifica se o aluno já está cadastrado ou não. Se não, os dados são inseridos automaticamente no Banco de Dados e o cadastro é feito.



**Figura 3.5.2: Diagrama de Sequência referente a Cadastrar Alunos**



Na Figura 3.5.3 é mostrado o Diagrama de Sequência referente a Consulta Acervo. Nele temos como ator e usuário do sistema o Aluno, que pode solicitar consulta ao acervo. O Sistema irá solicitar que ele insira os dados de login. Após isso, o aluno deve inserir uma chave de pesquisa para procurar o livro desejado; o sistema vai procurar informações relacionadas a essa chave no banco de dados e informar ao aluno se o livro solicitado existe na biblioteca e suas informações básicas. Após isso, podemos ver ações não-obrigatórias, que são referentes ao caso de uso Entrar na Lista de Espera: o aluno deve solicitar entrar na lista de espera, e o sistema vai adicioná-lo à lista referente que está no Banco de Dados e mostrar a lista atualizada ao aluno para que ele possa verificar sua posição.



**Figura 3.5.3: Diagrama de Sequência referente a Consulta Acervo**

Os diagramas aqui mostrados serviram como a base para o desenvolvimento do sistema com as linguagens de programação propriamente ditas, ajudando a estruturar e organizar o processo de construção do trabalho, como mostrado no Capítulo 4.

#### 4. RESULTADOS

Após a finalização do processo de planejamento do sistema, foi iniciada a criação das páginas do projeto, utilizando as linguagens HTML, PHP e CSS. Observando que existia um prazo curto para criação do maior número de páginas até a entrega do projeto, foi possível programar as seguintes funcionalidades do sistema:

- As telas principais de interação com o usuário, sendo elas:
  - a tela de escolha do tipo de usuário a acessar o sistema (Aluno, ou Funcionário) mostrada na Figura 4.1, que direciona o ator a sua tela de login;

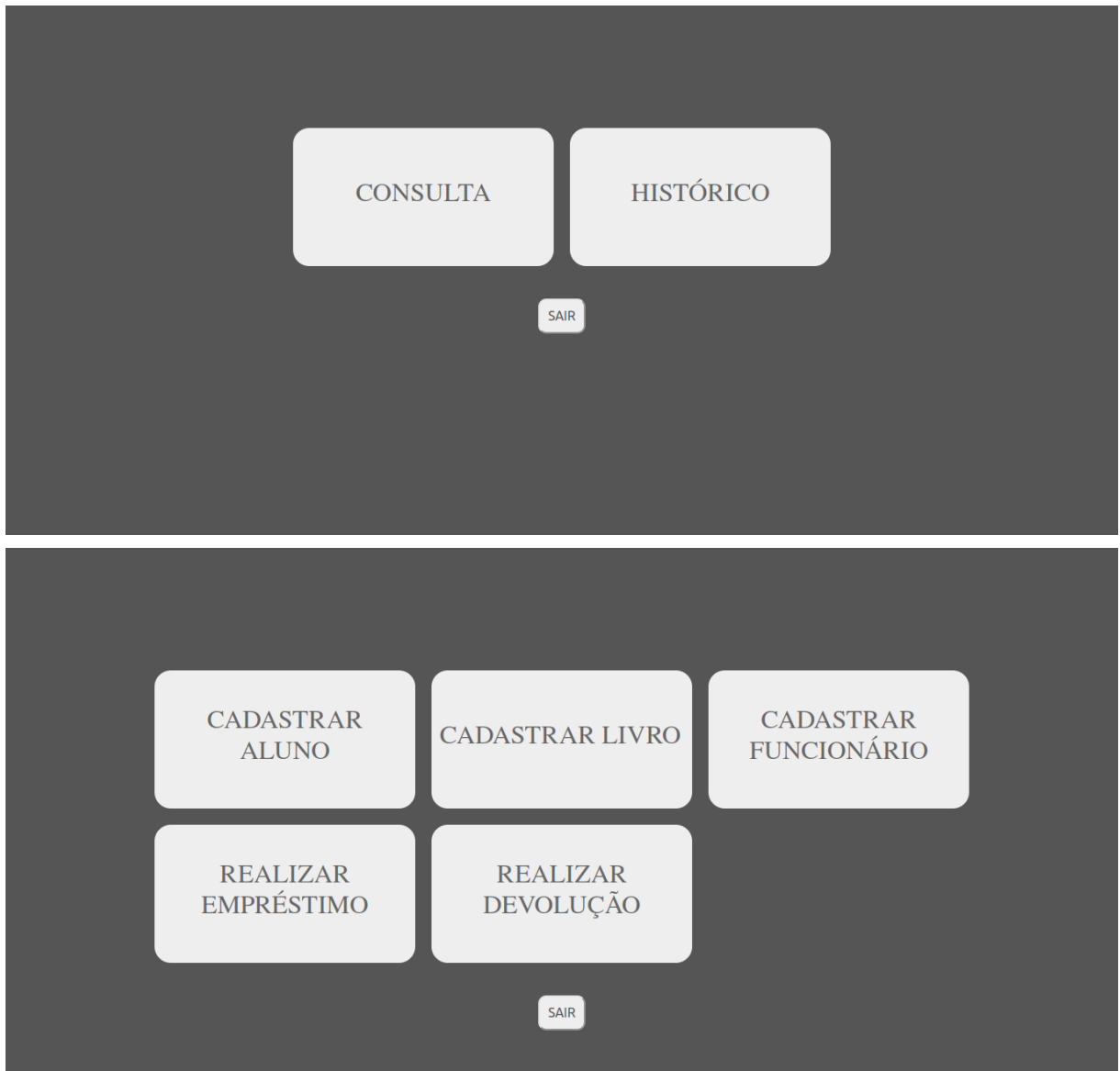


- as telas de login dos usuários, que são separadas e pedem as informações de identificação (CPF ou Matrícula, e senha), nas Figuras 4.2 para o aluno e 4.3 para o funcionário;

A screenshot of a login form titled "LOGIN ALUNO" centered on a dark gray background. The form is a light gray rounded rectangle. It contains two input fields: "Matrícula:" and "Senha:". Below the inputs are two green buttons: "ENTRAR" and "VOLTAR".

A screenshot of a login form titled "LOGIN FUNCIONÁRIO" centered on a dark gray background. The form is a light gray rounded rectangle. It contains two input fields: "CPF:" and "Senha:". Below the inputs are two green buttons: "ENTRAR" and "VOLTAR".

- e as telas iniciais para cada usuário, contendo as principais funcionalidades tanto do aluno como do funcionário (Figura 4.4 do aluno e Figura 4.5 do funcionário);



- a tela de acervo, que mostra os livros disponíveis, e que é uma funcionalidade para o aluno (Figura 4.6);

Acervo				
ISBN	Título	Autor	Quantidade	Edição
978-8516035468	Estrelas Tortas	Walcyr Carrasco	2	
978-8525056009	Admirável Mundo Novo	Aldous Huxley	2	1ª
978-8535909555	A revolução dos bichos	George Orwell	2	1ª
978-8580570083	O Herói Perdido	Rick Riordan	3	1ª
978-8580575392	O Ladrão de Raios	Rick Riordan	2	Não Informado
978-8580576337	Deuses Gregos	Rick Riordan	3	1ª

VOLTAR

- as telas de cadastro de alunos, livros ou funcionários, funções do funcionário, onde são pedidas as informações necessárias para cadastro de cada item ou pessoa (Figura 4.7 para cadastro de alunos, 4.8 para cadastro de livros e 4.9 para cadastro de funcionários); e

### CADASTRAR ALUNO

Nome:

Matrícula:

Período:

Data de Nascimento:

Endereço:

Email:

Senha:

CPF do funcionário:

CADASTRAR

VOLTAR

### CADASTRAR LIVRO

ISBN:

Título:

Autor:

Quantidade:

Edição:

CPF:

CADASTRAR

VOLTAR

### CADASTRAR FUNCIONARIO

Nome:

CPF:

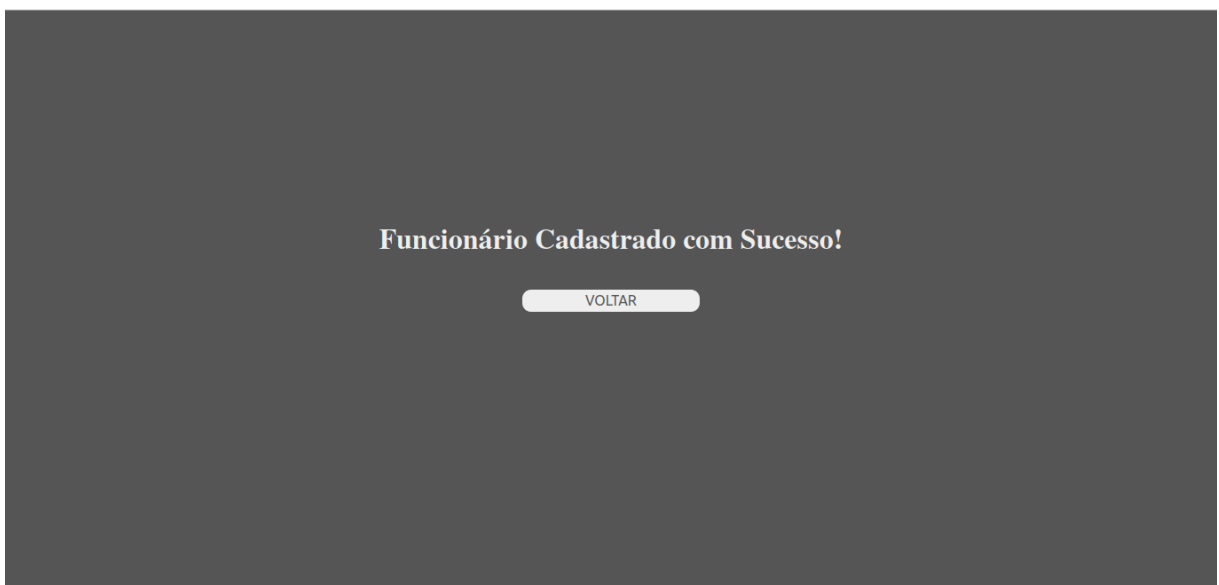
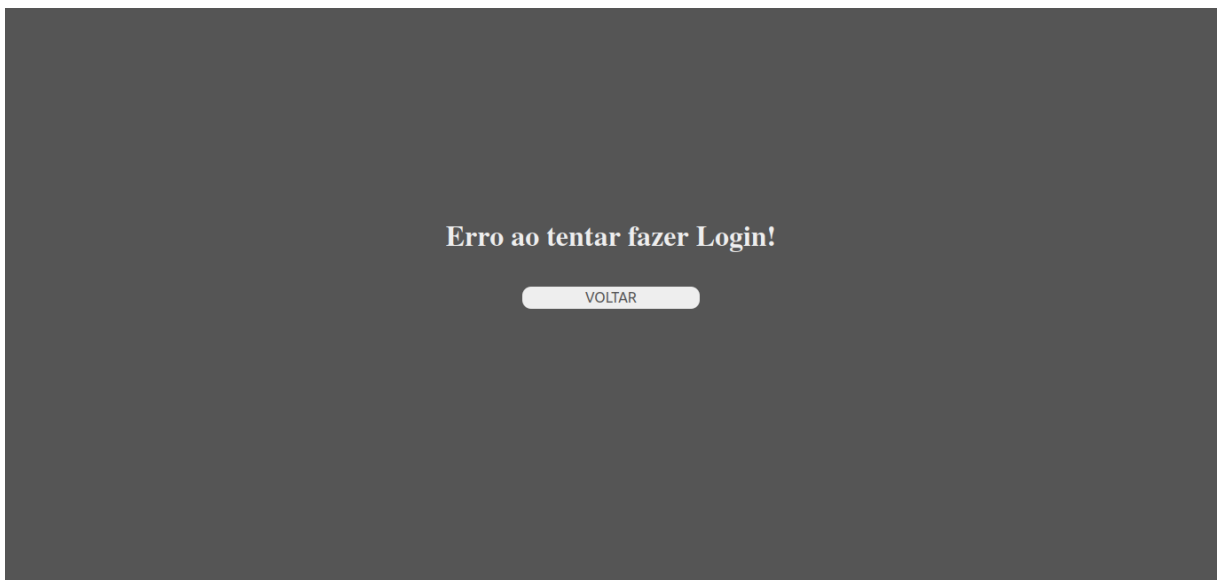
Senha:

CADASTRAR

VOLTAR

- as telas informativas, que informam o status da solicitação que você fez. São elas:
  - erro ao tentar realizar o *login* (Figura 4.10), tela usada em caso de erro do *login* tanto do funcionário como do aluno; e
  - cadastro realizado com sucesso (Figura 4.11), que é válida para os três tipos de cadastro previamente citados neste capítulo.

O desenvolvimento das telas aqui ilustradas foi de fundamental importância para a conclusão do projeto e no futuro será a base utilizada para continuidade do trabalho, onde será buscada a finalização do produto e aprimoramento dele através das fases de teste com os usuários.



## **5. CONSIDERAÇÕES FINAIS**

O objetivo deste trabalho era mostrar o projeto e início do desenvolvimento de uma aplicação Web de um sistema de gerenciamento de biblioteca, onde encontra-se de forma simples e prática a execução automatizada das principais funções que uma biblioteca necessita executar no dia a dia. Esse sistema propõe auxiliar os funcionários unindo todas as informações necessárias para execução de suas tarefas em uma plataforma simples e funcional, facilitando a documentação das atividades ocorridas, sejam elas empréstimos ou chegada e perda de documentos da biblioteca, onde também é mostrada formas de apresentar o ambiente e suas possibilidades para o aluno.

### **5.1. TRABALHOS FUTUROS**

Tendo em vista que a execução do projeto foi interrompida devida ao prazo de entrega, é colocado como próximas etapas deste trabalho a continuidade da programação do sistema, onde devem ser desenvolvidas as funcionalidades de:

- realização de empréstimos;
- renovação de empréstimos;
- devoluções de livros;
- inserção do aluno na lista de espera de um livro já emprestado;
- visão geral e possibilidade de edição dos dados dos usuários;
- visão geral do histórico de empréstimos realizados pelo aluno e
- busca personalizada de livros no acervo (útil para acervos com muitos livros).

Além de outras funcionalidades que podem surgir ao longo do desenvolvimento do projeto. Embora a ideia geral do sistema tenha sido delimitada ao longo do planejamento, é comum que outros desafios surjam com o andamento do projeto, e a capacidade de adaptação a essas necessidades é uma das características que deve estar presente durante o desenvolvimento de um projeto desse porte.



## REFERÊNCIAS

- SOMMERVILLE, Ian. **Engenharia de Software**. 8ª ed. Editora: Addison Wesley, 2007.
- GUEDES, Gilleanes T. A. **UML 2 : uma abordagem prática**. 2ª ed. São Paulo: Novatec Editora, 2011.
- BENTO, Evaldo Júnior. **Desenvolvimento web com PHP e MySQL**. São Paulo: Editora Casa do Código, 2021.
- PHP5 Tutorial – w3schools.com. Disponível em <https://www.w3schools.com/php/>. Último acesso em 20 de Junho de 2022.
- PHP: Manual do PHP - Manual – php.net. Disponível em [https://www.php.net/manual/pt\\_BR/](https://www.php.net/manual/pt_BR/). Último acesso em 20 de Junho de 2022.
- HTML: Linguagem de Marcação de Hiper-Texto | MDN – developer.mozilla.org. Disponível em <https://developer.mozilla.org/pt-BR/docs/Web/HTML>. Último acesso em 20 de Junho de 2022.
- CSS | MDN – developer.mozilla.org. Disponível em <https://developer.mozilla.org/pt-BR/docs/Web/CSS>. Último acesso em 20 de Junho de 2022.