

INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DO RIO
GRANDE DO NORTE

SAMARA JÉSSICA SANTOS DE SOUSA

**APLICAÇÃO WEB PARA O GERENCIAMENTO DE ACERVO DE UMA
BIBLIOTECA ESCOLAR**

NATAL – RN

2021

SAMARA JÉSSICA SANTOS DE SOUSA

**APLICAÇÃO WEB PARA O GERENCIAMENTO DE ACERVO DE UMA
BIBLIOTECA ESCOLAR**

Monografia apresentada ao Curso de Tecnologia em Análise e Desenvolvimento de Sistemas do Instituto Federal de Educação, Ciência e Tecnologia do Rio Grande do Norte, em cumprimento às exigências legais como requisito parcial à obtenção do título de Tecnólogo em Análise e Desenvolvimento de Sistemas.

Orientador: Dr. Eduardo Bráulio Wanderley Netto.

NATAL – RN

2021

Sousa, Samara Jéssica Santos de.

S729a Aplicação web para o gerenciamento de acervo de uma biblioteca escolar / Samara Jéssica Santos de Sousa. – Natal, 2021.
33 f : il. Color.

Monografia (Tecnólogo em Análise e Desenvolvimento de Sistemas)
– Instituto Federal de Educação, Ciência e Tecnologia do Rio Grande do Norte. Natal, 2021.

Orientador (a): Dr. Eduardo Bráulio Wanderley Netto.

1. Desenvolvimento de Sistemas. 2. Biblioteca escolar – Gerenciamento. 3. Biblioteca escolar – Automação. 4. Spring Framework (software). 5. PostgreSQL (banco de dados). 6. Thymeleaf (software). I. Wanderley Netto, Eduardo Bráulio. II. Instituto Federal de Educação, Ciência e Tecnologia do Rio Grande do Norte. III. Título.

CDU: 004.41

SAMARA JÉSSICA SANTOS DE SOUSA

**APLICAÇÃO WEB PARA O GERENCIAMENTO DE ACERVO DE UMA
BIBLIOTECA ESCOLAR**

Monografia apresentada ao Curso de Tecnologia em Análise e Desenvolvimento de Sistemas do Instituto Federal de Educação, Ciência e Tecnologia do Rio Grande do Norte, em cumprimento às exigências legais como requisito parcial à obtenção do título de Tecnólogo em Análise e Desenvolvimento de Sistemas.

Monografia apresentada e aprovada em 15 / 04 / 2021, pela seguinte Banca Examinadora:

BANCA EXAMINADORA

Prof. Dr. Eduardo Bráulio Wanderley Netto – Presidente
Instituto Federal de Educação, Ciência e Tecnologia do Rio Grande do Norte

Prof. Dr. Gilbert Azevedo da Silva – Examinador
Instituto Federal de Educação, Ciência e Tecnologia do Rio Grande do Norte

Prof.^a M.^a Danielle Gomes de Freitas Medeiros – Examinador
Instituto Federal de Educação, Ciência e Tecnologia do Rio Grande do Norte

RESUMO

As bibliotecas escolares são fundamentais no incentivo à leitura e escrita entre os seus alunos e no auxílio aos professores em suas aulas. E a tecnologia deve ser utilizada a seu favor como uma ferramenta de auxílio, para que possa alcançar seus objetivos de aprendizagem enquanto um ambiente de promoção à leitura e a pesquisa. Sendo assim, o propósito do trabalho é desenvolver uma solução para o gerenciamento de uma biblioteca de uma escola pública. A metodologia foi baseada em entrevistas com o respectivo levantamento de requisitos. Foi utilizada as tecnologias *Spring Framework*, *Thymeleaf* e *PostgreSQL* para realização deste trabalho. Como resultado temos um sistema web que permite o controle do acervo e dos empréstimos realizados na biblioteca, assim como de seus usuários. O sistema pode assim ser expandido com a implementação de novas funcionalidades a partir de novas necessidades que possam vir a ser identificadas.

Palavras-chave: Biblioteca escolar. Sistema web. Tecnologia.

ABSTRACT

School libraries are fundamental to encourage reading and writing among students and to help teachers in their classes. Technology should be used as an aid tool to achieve learning objectives as an environment to promote reading and research. Therefore, the purpose of the work is to develop a solution for the management of a public school library. The methodology was based on interviews with the respective requirements survey. Technologies *Spring Framework*, *Thymeleaf* and PostgreSQL were used to carry out this work. The methodology and technologies used in the development were presented, the requirements were surveyed and the use cases were defined. As a result, we have a web system that allows control of the collection and loans made in the library, as well as its users. The system can thus be expanded with the implementation of new features based on new needs that may be identified.

Keywords: School library. Web system. Technology.

LISTA DE FIGURAS

Figura 1 – Modelo V	13
Figura 2 – Diagrama de casos de uso.....	22
Figura 3 – Diagrama de classes do projeto	23
Figura 4 – Protótipos de tela	24
Figura 5 – Tela inicial do sistema.....	25
Figura 6 – Tela de login.....	26
Figura 7 – Tela de cadastro de usuário.....	26
Figura 8 – Recorte da tela de listagem de usuários	27
Figura 9 – Recorte da tela de cadastro de livro.....	27
Figura 10 – Recorte da tela de listagem de livros	28
Figura 11 – Recorte da tela de resultado de busca por título.....	28
Figura 12 – Recorte da tela de empréstimo	29
Figura 13 – Recorte da tela de empréstimo por usuário	29

SUMÁRIO

1	INTRODUÇÃO	7
1.1	OBJETIVOS	7
1.1.1	Objetivo geral	7
1.1.2	Objetivos específicos	7
1.2	METODOLOGIA	8
1.3	ORGANIZAÇÃO DO TRABALHO	8
2	FUNDAMENTAÇÃO TEÓRICA	10
2.1	IMPORTÂNCIA DA LEITURA E DAS BIBLIOTECAS ESCOLARES	10
2.2	APRESENTAÇÃO DAS TECNOLOGIAS UTILIZADAS NO PROJETO	12
2.2.1	Metodologia em cascata	12
2.2.2	<i>Spring Framework</i>	13
2.2.2.1	<i>Spring MVC</i>	14
2.2.2.2	<i>Spring Data JPA</i>	14
2.2.2.3	<i>Spring Boot</i>	15
2.2.2.4	<i>Thymeleaf</i>	16
2.2.3	PostgreSQL	16
2.2.4	Eclipse	17
2.3	FERRAMENTAS DE MODELAGEM	17
3	MODELAGEM DO PROJETO	19
3.1	LEVANTAMENTO DE REQUISITOS	19
3.2	DIAGRAMA DE CASOS DE USO	20
3.3	DIAGRAMA DE CLASSES	22
3.4	PROTÓTIPOS DE TELA	24
4	RESULTADOS	25
4.1	APRESENTAÇÃO DO SISTEMA	25
5	CONSIDERAÇÕES FINAIS	31
	REFERÊNCIAS	32

1 INTRODUÇÃO

As bibliotecas escolares são importantes para o incentivo à cultura da leitura e da escrita entre os jovens e o seu material disponível também serve como apoio aos professores que se utilizam dele para dar suas aulas. Porém, a experiência de procurar um determinado livro em uma biblioteca pode ser desafiadora e frustrante. Além da falta de sinalização adequada, ainda há a possibilidade de um título não estar disponível.

Para a escola, também pode ser difícil manter um controle sobre o acervo de sua biblioteca se não tiver disponível algum sistema de organização, podendo ocorrer perdas de livros ou aquisições desnecessárias pela falta de controle adequado.

Sendo assim, este trabalho propõe um sistema de gerenciamento onde é possível realizar o controle de empréstimos do acervo assim como de seus usuários, promovendo um melhor acesso a tudo que ela oferece.

1.1 OBJETIVOS

Para o trabalho proposto foram definidos o objetivo geral e os objetivos específicos.

1.1.1 Objetivo geral

Desenvolver uma aplicação web que permita o gerenciamento do acervo de uma biblioteca escolar, permitindo o controle de empréstimos de itens aos usuários.

1.1.2 Objetivos específicos

Visando atingir o objetivo geral, os seguintes objetivos específicos foram definidos:

- a) fazer uma explanação do problema proposto;
- b) definir o escopo, os requisitos e os casos de uso do projeto;
- c) modelar o banco de dados;
- d) definir quais tecnologias serão utilizadas no desenvolvimento da aplicação;
- e) desenvolver os protótipos de telas do sistema;

- f) desenvolver a API do sistema web de gerenciamento de biblioteca escolar;
- g) desenvolver os *templates* da aplicação.

1.2 METODOLOGIA

Para a fundamentação teórica do trabalho, foi utilizada como base para o estudo a biblioteca de uma escola estadual. Foram realizadas entrevistas e visitas ao espaço para entender melhor o problema. Também foram utilizados materiais de referência como manuais de orientações básicas para a organização de bibliotecas oferecidos pelo governo do Estado, fichas de cadastro de usuário e de livros e o regulamento da biblioteca para definir os requisitos do sistema, modelar o banco de dados e definir os casos de uso do projeto.

Para o desenvolvimento do sistema, foi utilizada a metodologia em cascata, cujas fases de todo o projeto são sequenciais.

Na fase de planejamento, foram definidos os requisitos e as funcionalidades do sistema e a definição dos casos de uso. Na fase de modelagem, foi pensado em como seria a arquitetura do sistema. Foram realizadas seguindo as informações obtidas na fase anterior, as definições dos diagramas de classes de projeto, diagrama de entidade-relacionamento e o diagrama de casos de usos onde cada caso é considerado uma “unidade” a ser implementada no projeto. Na fase de construção, cada “unidade” do programa é implementada. Ao final da implementação, são feitos os testes unitários para certificar-se de que os casos de uso foram atendidos corretamente. E, para concluir, foi realizado um teste geral para garantir que os requisitos do sistema foram corretamente atendidos, finalizando assim a etapa de desenvolvimento e a entrega do software.

1.3 ORGANIZAÇÃO DO TRABALHO

Esse trabalho está organizado em 6 capítulos. O capítulo 1 abordou a introdução e a metodologia utilizada no projeto. O capítulo 2 trata da fundamentação teórica, falando da importância das bibliotecas nas escolas e das tecnologias utilizadas no desenvolvimento do projeto. O capítulo 3 apresenta a modelagem do projeto, com os diagramas e os protótipos de tela projetados. O capítulo 4 mostra os resultados do projeto, apresentando as principais funcionalidades implementadas e

os templates desenvolvidos. No capítulo 5 temos as considerações finais do trabalho e no capítulo 6 as referências citadas ao longo do documento.

2 FUNDAMENTAÇÃO TEÓRICA

Nesta seção será apresentado uma breve pesquisa sobre a importância da leitura e das bibliotecas escolares e um estudo sobre as tecnologias escolhidas para o desenvolvimento do projeto.

2.1 IMPORTÂNCIA DA LEITURA E DAS BIBLIOTECAS ESCOLARES

A leitura é de fundamental importância na formação educacional das pessoas. Ela é uma importante ferramenta de aprendizagem para uma educação de qualidade. “Seja por prazer, seja para estudar ou para se informar, a prática da leitura aprimora o vocabulário e dinamiza o raciocínio e a interpretação” (BATISTA, c2021, p. 1). É nela também que desenvolvemos nosso senso crítico e ampliamos a nossa visão de mundo.

A produção de leitura consiste no processo de interpretação desenvolvido por um sujeito-leitor que, defrontando-se com um texto, analisa, questiona com o objetivo de processar seu significado projetando sobre ele sua visão de mundo para estabelecer uma interação crítica com o texto (INDURSKY; ZINN, 1985, p. 56 *apud* ARANA; KLEBIS, 2015, p. 4)

Com isso, Magnani (1989 p. 35 *apud* MANYS, [201-?], p. 3) diz que “para ler e escrever é preciso, antes de mais nada, ser alfabetizado, tarefa que, em nossa sociedade, cabe historicamente à escola”. A biblioteca escolar é uma essencial aliada das escolas e dos professores nessa missão. Ela oferece um ambiente favorável ao aprendizado, dando suporte e ferramentas que contribuem para a alfabetização, a educação, a informação e o desenvolvimento econômico, social e cultural dos seus alunos. Sendo assim, as atividades desenvolvidas na biblioteca escolar devem motivar e despertar a curiosidade e o interesse dos estudantes pela leitura.

Ensino e biblioteca não se excluem, completam-se. Uma escola sem biblioteca é instrumento imperfeito. A biblioteca sem ensino, ou seja, sem tentativa de estimular, coordenar e organizar a leitura, será, por seu lado, instrumento vago e incerto (LOURENÇO FILHO, 1946, p. 4 *apud* MANYS, [201-?], p. 7)

Por isso, é preciso garantir condições para o pleno funcionamento desses espaços, voltando-se para a melhoria da qualidade do atendimento nas bibliotecas e

para o alcance dos seus objetivos de aprendizagem enquanto ambiente informacional e de promoção de leitura e pesquisa. Pois, segundo Jesus (2015, p. 3), “o uso inadequado da biblioteca escolar vai contra o objetivo da sua criação, pois ela surgiu para ampliar o ensino formal, sendo de sua responsabilidade parte significativa no desenvolvimento da capacidade de ler”.

Para isso, podemos contar com a ajuda dos recursos tecnológicos que podem proporcionar uma boa gestão do acervo e facilitar o processo de busca e de empréstimos de livros, assim como atender à atual geração de usuários que possui facilidade para usar recursos tecnológicos mais modernos.

A explosão informacional e o aumento da oferta e da busca por publicações eletrônicas mudaram os processos das bibliotecas e o uso dos recursos informacionais por seus usuários finais. Os acervos, compostos por materiais impressos e eletrônicos, envolvem novas formas de aquisição, de registro, de disponibilização e acesso, de controle e de utilização (TONDING; VANZ, 2018, p. 76).

Assim como em qualquer outro, os sistemas para gerenciamento de bibliotecas precisaram evoluir a fim de atender às necessidades atuais das bibliotecas e de seus usuários finais utilizando-se das novas tecnologias para modernizar seus sistemas.

Os serviços tradicionais de organização, armazenamento e tratamento de coleções digitais tiveram desenvolvimento rápido; o fichário foi transformado em catálogo público on-line – esse foi o primeiro serviço tradicional transformado para o digital e, depois, on-line; o empréstimo local – agregado da reserva pela internet e envio do livro ou artigo eletrônico; aplicativos de recuperação de informações foram projetados inicialmente para o acesso à informação e não ao conteúdo. Hoje esses aplicativos incluem gerenciamento de sistemas, serviços de busca on-line (ROWLEY, 2002 *apud* CUENCA, 2009, p. 3).

Podemos citar como benefícios às bibliotecas pela utilização das tecnologias de informação a “racionalidade no trabalho, aumento de produção, melhor controle e uma maior facilidade de armazenamento e disseminação da informação” (RAMOS, 2003 *apud* SILVIA, 2008, p. 3). E de acordo com Tonding e Vanz (2018, p. 75):

O ganho de produtividade se dá através da otimização de fluxos de trabalho que o sistema contempla e também porque, dada sua implementação em nuvem, as equipes de TI ficam eximidas de atividades rotineiras de administração dos recursos de hardware e software. Com esta força de trabalho adicional, as bibliotecas podem centrar esforços em atividades que agregam valor e, assim, recuperar seu papel fundamental no processo de ensino, aprendizagem e pesquisa.

2.2 APRESENTAÇÃO DAS TECNOLOGIAS UTILIZADAS NO PROJETO

Nesta seção será apresentado a metodologia de desenvolvimento escolhida para o projeto assim como as tecnologias utilizadas.

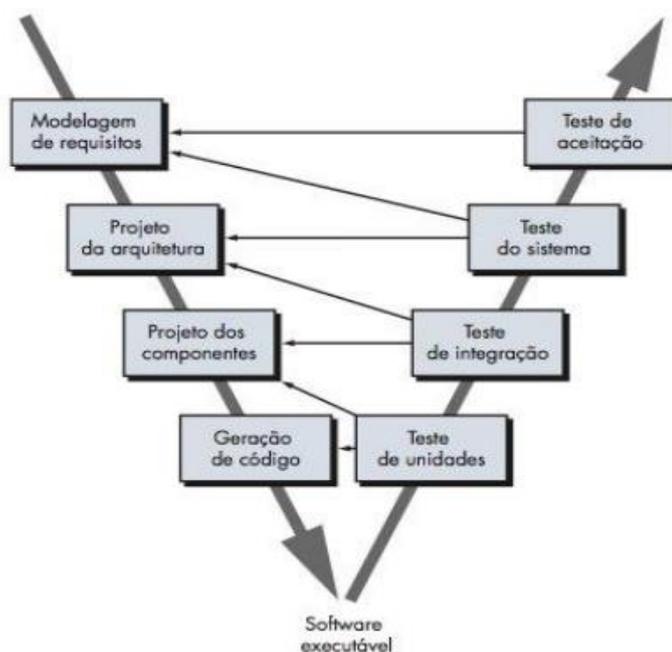
2.2.1 Metodologia em cascata

Essa metodologia de desenvolvimento de software demanda uma longa fase de planejamento e conta com escopo e cronograma definidos. As ações avançam quando uma etapa é finalizada. Isso pode tornar o desenvolvimento mais preso, porém com fases mais previsíveis. De acordo com Pressman (2011, p. 59):

O modelo cascata, algumas vezes chamado ciclo de vida clássico, sugere uma abordagem sequencial e sistemática para o desenvolvimento de software, começando com o levantamento de necessidades por parte do cliente, avançando pelas fases de planejamento, modelagem, construção, emprego e culminando no suporte contínuo do software concluído.

Uma variante do modelo em cascata é denominado modelo V mostrado na Figura 1. À medida que descemos pelo lado esquerdo do V, os requisitos básicos do problema são refinados e o problema é mais detalhado, assim como sua solução. Quando o código for gerado, a equipe se desloca para o lado direito do V, realizando os testes necessários que validem o que foi definido no lado esquerdo. Não há diferenças entre o modelo em cascata e o modelo V. “O modelo V fornece uma forma para visualizar como a verificação e as ações de validação são aplicadas ao trabalho de engenharia anterior” (PRESSMAN, 2011, p. 60).

Figura 1 – Modelo V



Fonte: Pressman (2011).

Para o desenvolvimento do sistema web de gerenciamento de bibliotecas, foram utilizadas as tecnologias descritas a seguir.

2.2.2 Spring Framework

O *Spring* é um conjunto de projetos *Open Source* que visam ajudar a criar aplicações Java com simplicidade e flexibilidade e que surgiu como uma alternativa ao Java EE. O ecossistema *Spring* cobre várias áreas como o *Spring Data*, o *Spring Security* e diversos outros projetos que vão desde *cloud computing* até *big data*.

O *Spring Framework* é um dentre um conjunto de projetos que o Spring possui e que serve como base para todos os outros. Ele foi pensado para que as aplicações pudessem focar mais na regra de negócios e menos na infraestrutura. Dentre suas principais funcionalidades, temos: *Spring Model View Controller* (MVC), suporte *Java Database Connectivity* (JDBC) e *Java Persistence API* (JPA) e injeção de dependências.

O *Spring* torna a construção de aplicativos da web rápida e descomplicada. Ao remover muito do código padrão e da configuração associada ao desenvolvimento web, você obtém um modelo de programação da web moderno que agiliza o desenvolvimento de aplicativos HTML do lado do

servidor, APIs REST e sistemas bidirecionais baseados em eventos (WEB, 2021, p. 1, tradução nossa).

O *Spring MVC* é um *framework* para criação de aplicações web e serviços RESTful. O suporte JDBC e JPA é necessário para a persistência de dados. A injeção de dependências é um tipo de inversão de controle que dá o nome ao processo de prover instâncias de classes de que um objeto precisa para funcionar. Assim, conseguimos programar voltados para interfaces e, com isso, manter o baixo acoplamento entre classes, ou seja, uma baixa dependência de uma classe com outra de um mesmo projeto.

2.2.2.1 *Spring MVC*

O *Spring MVC* é um *framework* que auxilia no desenvolvimento de aplicações web robustas, flexíveis e com uma clara separação de responsabilidades nos papéis do tratamento da requisição.

Ele já tem todas as funcionalidades que precisamos para (1) atender as requisições HTTP, (2) delegar responsabilidades de processamento de dados para outros componentes e (3) preparar a resposta que precisa ser dada. É uma excelente implementação do padrão MVC (AFONSO, 2017, p. 1).

O MVC é acrônimo de *Model*, *View* e *Controller*. O *controller* recebe os dados enviados pelo browser e passa os dados para o *model*, que, por sua vez, executa todas as regras de negócio da aplicação, como validações e acesso ao banco de dados. O resultado, então, é enviado de volta para o *controller*, que retorna o nome da *view* com os dados necessários para renderizar a página. O *framework* encontra a *view* que processa os dados, transformando-os em um *HyperText Markup Language* (HTML).

2.2.2.2 *Spring Data JPA*

O *Spring Data* é um projeto que tem o objetivo de simplificar o acesso a tecnologias de armazenamento de dados, sejam elas relacionais (MySQL, PostgreSQL etc.) ou não.

Ele ajuda os desenvolvedores padronizando o uso de algumas funcionalidades, e isso faz com que seja necessário menos esforço para implementar as mesmas

coisas. Um exemplo disso seria a implementação padrão que ele já nos dá em repositórios, incluindo métodos como *save*, *delete*, *findOne*, entre outros. De acordo com o site oficial, o *Spring Data*:

Facilita o uso de tecnologias de acesso a dados, bancos de dados relacionais e não relacionais, estruturas de redução de mapa e serviços de dados baseados em nuvem. Este é um projeto abrangente que contém muitos subprojetos que são específicos para um determinado banco de dados. Os projetos são desenvolvidos trabalhando em conjunto com muitas das empresas e desenvolvedores que estão por trás dessas tecnologias interessantes (WEB, 2021, p.1, tradução nossa).

2.2.2.3 *Spring Boot*

O *Spring Boot* ajuda o desenvolvedor a focar nas funcionalidades da sua aplicação com o mínimo de configurações. O código não é gerado, ele analisa o projeto automaticamente e o configura, sendo possível a customização das configurações por parte do programador.

O *Spring Boot* transforma a maneira como você aborda as tarefas de programação em Java, otimizando radicalmente sua experiência. O *Spring Boot* combina necessidades, como um contexto de aplicativo e um servidor Web incorporado e configurado automaticamente para tornar o desenvolvimento de microsserviços uma tarefa fácil. Você pode combinar o *Spring Boot* com um rico conjunto de bibliotecas de suporte, servidores, padrões e modelos do *Spring Cloud* para implantar com segurança arquiteturas inteiras baseadas em micro serviços na nuvem em tempo recorde (WEB, 2021, p. 1, tradução nossa).

O *Spring Boot* utiliza o Maven como gerenciador de dependências. O Maven, de acordo com o site oficial:

É uma ferramenta de gerenciamento e compreensão de projetos de software. Com base no conceito de modelo de objeto de projeto (POM), o Maven pode gerenciar a construção, o relatório e a documentação de um projeto a partir de uma informação central (APACHE, 2021, p. 1, tradução nossa).

As dependências de um projeto *Spring Boot* são adicionadas a um arquivo pom.xml, que pode vir a ficar extenso; porém, com o *Spring Boot*, podemos reduzir muito com os starters que ele fornece. Os *starters* são dependências que agrupam outras dependências, assim, se você precisa trabalhar com JPA e Hibernate, por exemplo, basta adicionar uma única entrada no pom.xml que todas as dependências necessárias serão adicionadas ao *classpath*.

2.2.2.4 Thymeleaf

Thymeleaf é um mecanismo de template Java moderno, *server-side*, para ambientes da web ou não. É uma biblioteca criada para facilitar a criação da camada de *view* com uma forte integração com o *Spring*, vindo como uma alternativa ao JSP. De acordo com o site oficial do *Thymeleaf*:

O principal objetivo do Thymeleaf é trazer modelos naturais elegantes para o seu fluxo de trabalho de desenvolvimento - HTML que pode ser exibido corretamente em navegadores e também funcionar como protótipos estáticos, permitindo uma colaboração mais forte em equipes de desenvolvimento. Com módulos para Spring Framework, um host de integração com suas ferramentas favoritas e a capacidade de conectar sua própria funcionalidade, o Thymeleaf é ideal para o desenvolvimento da web HTML5 JVM moderno - embora haja muito mais que ele pode fazer (THYMELEAF, 2020, p. 1, tradução nossa).

2.2.3 PostgreSQL

O PostgreSQL é um sistema gerenciador de banco de dados objeto-relacional *open source*.

PostgreSQL é um poderoso sistema de banco de dados relacional de objeto de código aberto que usa e estende a linguagem SQL combinada com muitos recursos que armazenam e escalam com segurança as cargas de trabalho de dados mais complicadas (POSTGRESQL, 2021, p. 1, tradução nossa).

Ele suporta uma grande parte do padrão SQL e oferece muitas características modernas, como é o caso de chaves estrangeiras, *functions*, *triggers*, *Views*, integridades transacionais, *data types*, funções agregadas, operadores, dentre muitos outros.

PostgreSQL vem com muitos recursos destinados a ajudar os desenvolvedores a construir aplicativos, criar ambientes tolerantes a falhas e ajudá-lo a gerenciar seus dados, não importando o tamanho do conjunto de dados. Além de ser gratuito e de código aberto, o PostgreSQL é altamente extensível.

O PostgreSQL tenta se adequar ao padrão SQL, em que tal conformidade não contradiga os recursos tradicionais ou possa levar a decisões arquitetônicas inadequadas. Muitos dos recursos exigidos pelo padrão SQL são suportados, embora, às vezes, com sintaxe ou função ligeiramente

diferente. Outros movimentos em direção à conformidade podem ser esperados ao longo do tempo (POSTGRESQL, 2021, p. 1, tradução nossa).

As origens do PostgreSQL remontam a 1986 como parte do projeto POSTGRES da Universidade da Califórnia em Berkeley e tem mais de 30 anos de desenvolvimento ativo na plataforma central.

2.2.4 Eclipse

Para o desenvolvimento do projeto, foi utilizada a IDE Eclipse. Ela foi criada para o desenvolvimento em Java, porém suporta várias outras linguagens a partir de *plugins* como C/C++, PHP, ColdFusion, Python, Scala e Kotlin. Ele foi feito em Java e segue o modelo *open source* de desenvolvimento de software.

Utiliza como biblioteca gráfica o *Standard Widget Toolkit* (SWT) ao invés do Swing. Possui forte orientação ao desenvolvimento baseado em *plugins* e o amplo suporte ao desenvolvedor com centenas de *plugins* que procuram atender às diferentes necessidades de diferentes programadores.

2.3 FERRAMENTAS DE MODELAGEM

O levantamento de requisitos serve para capturar as necessidades do cliente e que serão a base do desenvolvimento do sistema e de seus componentes futuros. De acordo com Sommerville (2007, p. 79), “os requisitos de um sistema são descrições dos serviços fornecidos pelo sistema e as suas restrições operacionais. Esses requisitos refletem as necessidades dos clientes de um sistema que ajuda a resolver algum problema”.

Os requisitos funcionais são aqueles que dizem respeito aos serviços que o sistema irá oferecer e como ele deve se comportar em determinadas situações, levando em consideração também as entradas fornecidas. Sommerville (2007, p. 81) diz que:

Os requisitos funcionais de um sistema descrevem o que o sistema deve fazer. Esses requisitos dependem do tipo de software que está sendo desenvolvido, dos usuários a que o software se destina e da abordagem geral considerada pela organização ao redigir os requisitos. Quando expressos como requisitos de usuário, eles são geralmente descritos de forma bastante abstrata. No entanto, os requisitos funcionais descrevem a função do sistema detalhadamente, suas entradas e saídas, exceções etc.

Os requisitos não funcionais dizem respeito às restrições sobre os serviços oferecidos pelo sistema, como restrições sobre o tempo de resposta e quais são as regras sobre uma determinada função do sistema, por exemplo. Sommerville (2007, p. 82) diz que:

Os requisitos não funcionais, como o nome sugere, são aqueles não diretamente relacionados às funções específicas fornecidas pelo sistema. Eles podem estar relacionados às propriedades emergentes do sistema, como confiabilidade, tempo de resposta e espaço de armazenamento. [...] Os requisitos não funcionais podem vir de características necessárias do software (requisitos de produto), da organização que desenvolve o software (requisitos organizacionais) ou de fontes externas.

Os diagramas de casos de uso servem para representar visualmente como os casos de uso se relacionam entre si no sistema e como serão utilizadas pelos usuários. Segundo Fowler (2006, p. 104), os casos de uso são: “uma técnica para captar os requisitos funcionais de um sistema. Eles servem para descrever as interações típicas entre os usuários de um sistema e o próprio sistema, fornecendo uma narrativa sobre como o sistema é utilizado”.

Os diagramas de classes são um tipo de diagrama da estrutura porque descrevem o que deve estar presente no sistema a ser modelado. Eles servem para mapear as estruturas do sistema ao modelar as suas classes, seus atributos, operações e as relações entre os objetos. Segundo Fowler (2006, p. 52):

Um diagrama de classes descreve os tipos de objetos presentes no sistema e os vários tipos de relacionamentos estáticos existentes entre eles. Os diagramas de classe também mostram as propriedades e as operações de uma classe e as restrições que se aplicam à maneira como os objetos estão conectados.

3 MODELAGEM DO PROJETO

Para realizar o levantamento de requisitos do projeto, foram realizadas entrevistas com uma professora responsável pela organização da biblioteca escolar. Ela sentiu a necessidade, no decorrer de seu ofício, de um sistema de gerenciamento para cadastrar o acervo e ter maior controle sobre ele, sobre os empréstimos e sobre usuários da biblioteca.

Foram também utilizadas como referências os materiais disponibilizados pela professora, como a ficha de cadastro de um livro indicando todos os campos necessários, a ficha de inscrição de usuários da biblioteca e o regulamento interno.

3.1 LEVANTAMENTO DE REQUISITOS

Neste tópico, serão apresentados os requisitos funcionais e não funcionais que foram definidos para o projeto. No quadro 1, temos os requisitos funcionais que foram definidos.

Quadro 1 – Requisitos funcionais

Cod.	Nome	Descrição
F01	Cadastro de usuarios administrador	O sistema deve ser capaz de manter cadastro de usuários administrador
F02	Cadastro de alunos e funcionários	O sistema deve ser capaz de manter o cadastro de alunos e funcionários
F03	Cadastro de acervo	O sistema deve permitir o cadastro de novos itens no acervo
F04	Atualização de cadastros	O sistema deve permitir mudanças no cadastro dos alunos, professores e funcionários e dos livros do acervo incluindo a desativação dos alunos/funcionário
F05	Autenticação de usuários	O sistema deve autenticar os usuários administrador
F06	Consulta de acervo	O sistema deve permitir a consulta do acervo da biblioteca
F07	Consulta de cadastro de alunos e funcionários	O sistema deve permitir a consulta de cadastro de usuários
F08	Empréstimo de livros	O sistema deve permitir o empréstimo do acervo a alunos, professores e funcionários cadastrados no sistema
F09	Relatório de empréstimos e pendências	O sistema deve permitir a emissão de um relatório de empréstimos por usuário

Fonte: Elaboração própria em 2021.

Foram definidos como requisitos funcionais os cadastros de usuários e livros no sistema, a consulta desses cadastros, a autenticação para usuários administradores, o empréstimo de livros entre outros.

No quadro 2, temos os requisitos não funcionais definidos para o sistema, levando em consideração o regulamento interno da biblioteca em estudo assim como as tecnologias escolhidas para o desenvolvimento.

Quadro 2 – Requisitos não funcionais

Cod.	Nome	Descrição
NF01	Persistência de dados	O sistema deve usar banco de dados para armazenar os dados do mesmo
NF02	SpringBoot Framework	O sistema usará o framework SpringBoot para implementação da API
NF03	Cadastro no sistema	Para realizar o cadastro no sistema deve ser aluno com matrícula ativa, professor ou funcionário
NF04	Realização de empréstimo	Somente poderão retirar livros os leitores devidamente inscritos na biblioteca; Deve existir mais de um exemplar no acervo; Livro deve estar marcado como disponível para empréstimo; E apenas um livro por empréstimo
NF05	Tipo de livro para empréstimo	O sistema deve permitir no cadastro do livro que o mesmo seja marcado como disponível para empréstimo ou não. (Não serão emprestados livros da Coleção de Referência (dicionários, enciclopédias, atlas, etc.)
NF06	Prazo de empréstimo	O prazo de empréstimo é de 15 (quinze) dias, podendo haver a renovação do empréstimo por igual período, caso o livro não esteja reservado para outro leitor;
NF07	Descumprimento do prazo de devolução	Caso o prazo seja descumprido, o usuário não poderá realizar empréstimos pelo período de 7 (sete) dias
NF08	Perda ou extravio do livro	O sistema deve permitir marcar o exemplar do livro como indisponível e retirá-lo da lista caso haja perda ou extravio do mesmo
NF09	Filtro de livros	O sistema deve permitir a filtragem de livros por nome e autor
NF10	Filtro de usuários	O sistema deve permitir a filtragem de usuários por nome e matrícula

Fonte: Elaboração própria em 2021.

Foram definidos como requisitos não funcionais a utilização do *framework Spring* para o desenvolvimento do sistema, o tempo para o prazo de empréstimo, a suspensão caso esse prazo não seja cumprido, regras para a realização de um empréstimo, a filtragem de livros entre outros.

3.2 DIAGRAMA DE CASOS DE USO

No quadro 3, temos os casos de uso definidos para o nosso sistema. Um ator, como citado na tabela, é um papel que um usuário desempenha com relação ao sistema. “Os atores realizam os casos de uso. Um único ator pode realizar muitos casos de uso; inversamente, um caso de uso pode ter vários atores executando-o” (FOWLER, 2006, p.104).

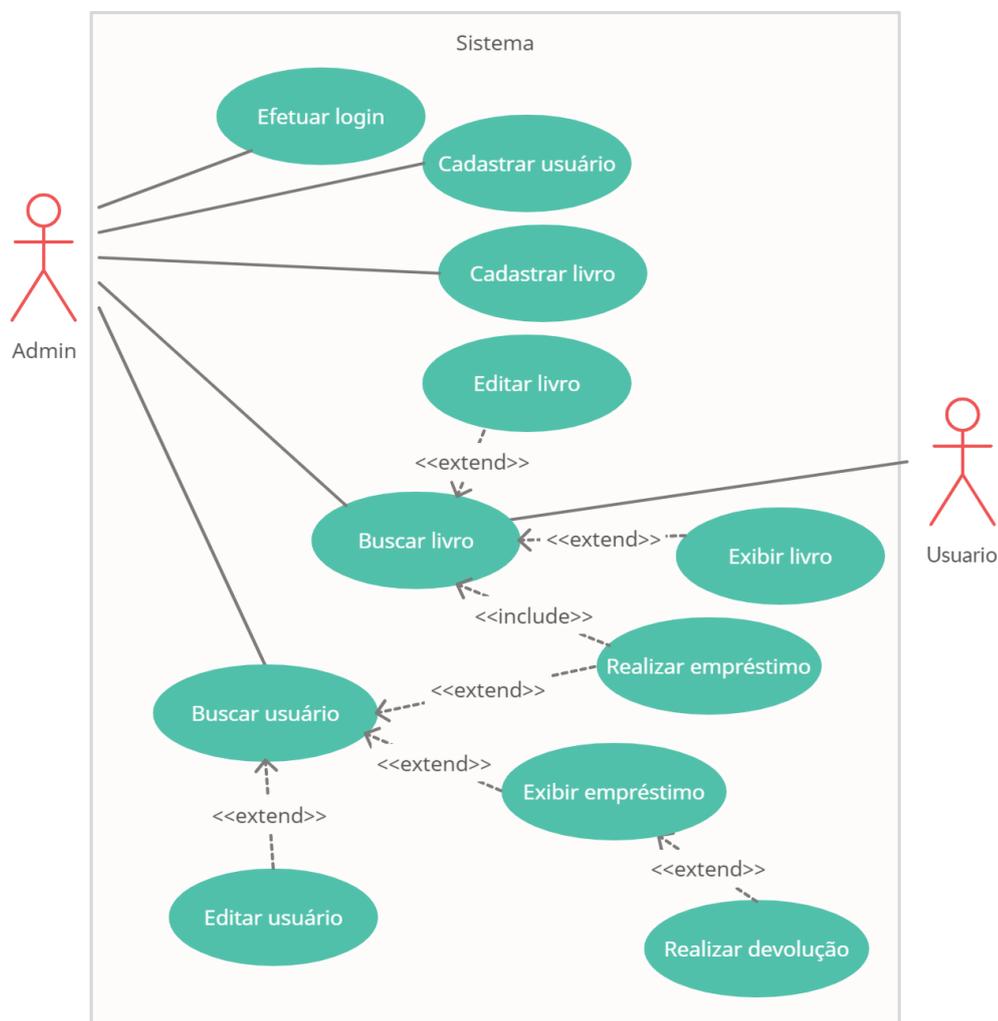
Quadro 3 – Casos de uso

Cod.	Nome	Descrição
UC01	Cadastrar usuários	O ator seleciona a opção "cadastrar novo usuário" fornece os dados e confirma o cadastro
UC02	Editar usuário	O ator modifica os dados de uma conta ativa
UC03	Buscar usuário	O ator decide os parâmetros de busca e recebe os resultados
UC04	Efetuar login	O ator seleciona opção "login" informa os seus dados de administrador e efetua o login
UC05	Cadastrar Livro	O ator seleciona seleciona opção "cadastrar novo livro", informa os dados e confirma
UC06	Buscar Livro	O ator decide os parâmetros de busca e recebe os resultados
UC07	Exibir Livro	Os dados do livro selecionado são exibidos
UC08	Editar livro	O ator seleciona a opção "Editar" e modifica os dados do livro cadastrado
UC09	Realizar Empréstimo	A ator seleciona a opção "Novo Empréstimo" busca o livro e o usuário e confirma o empréstimo
UC10	Realizar Devolução	O ator seleciona a opção "Realizar Devolução" busca na lista de empréstimos por usuário e confirma a devolução
UC11	Exibir empréstimo	O ator seleciona a opção "Exibir empréstimo" para ver os detalhes do empréstimo

Fonte: Elaboração própria em 2021.

Na figura 2 temos o diagrama de casos de uso do sistema. Nele temos representado o ator "Usuário" que tem apenas acesso ao caso de uso "Buscar Livro" e caso deseje ao "Exibir Livro". A relação entre esses dois casos de uso está representada pelo *extend* que significa que o ator pode ou não acessar esse caso de uso que está relacionado ao outro, por exemplo o usuário pode ou não clicar para exibir detalhes de um determinado livro na lista. Já o *include* significa que o caso de uso relacionado ao outro deve ser executado obrigatoriamente, como por exemplo, no caso de uso "Realizar Empréstimo" que depende do caso "Buscar Livro" para ser realizado. Já o ator "Admin" tem acesso a todos os casos de uso.

Figura 2 – Diagrama de casos de uso

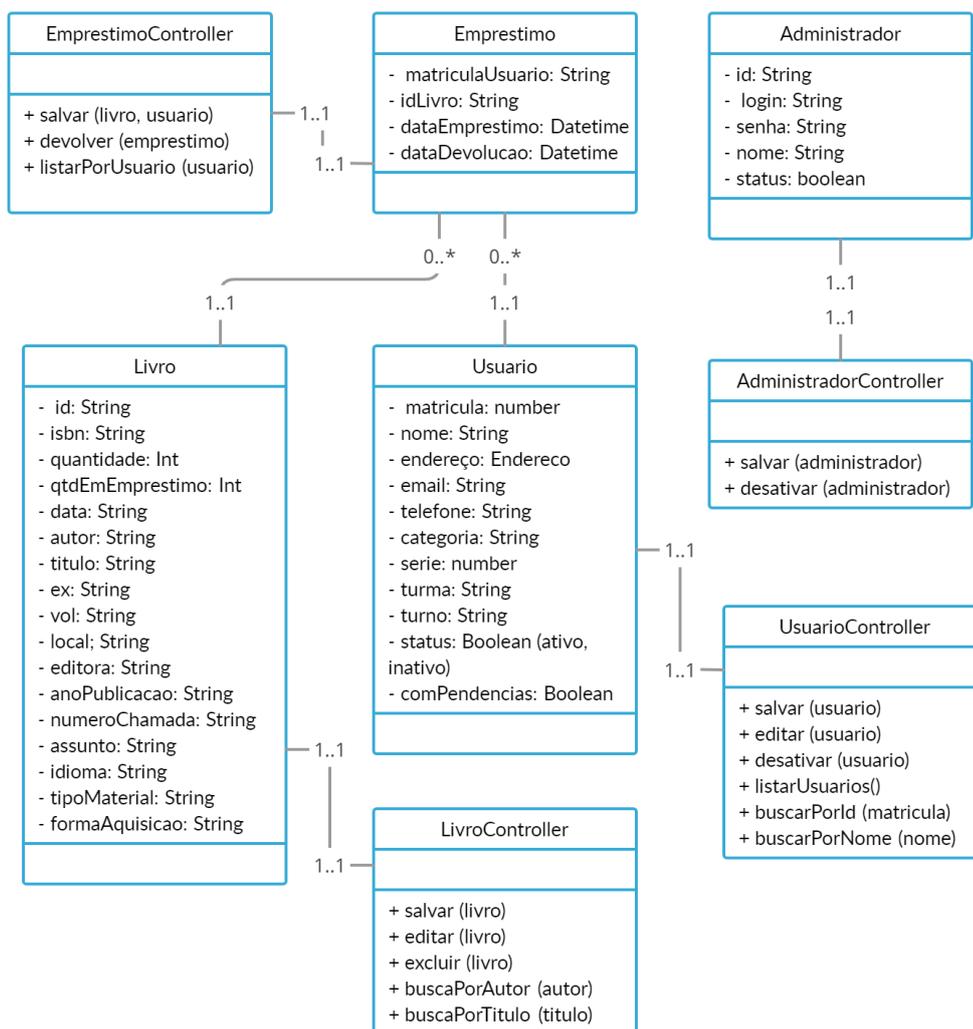


Fonte: Elaboração própria em 2021.

3.3 DIAGRAMA DE CLASSES

Na figura 3, temos o diagrama de classes definido para o sistema.

Figura 3 – Diagrama de classes do projeto



Fonte: Elaboração própria em 2021.

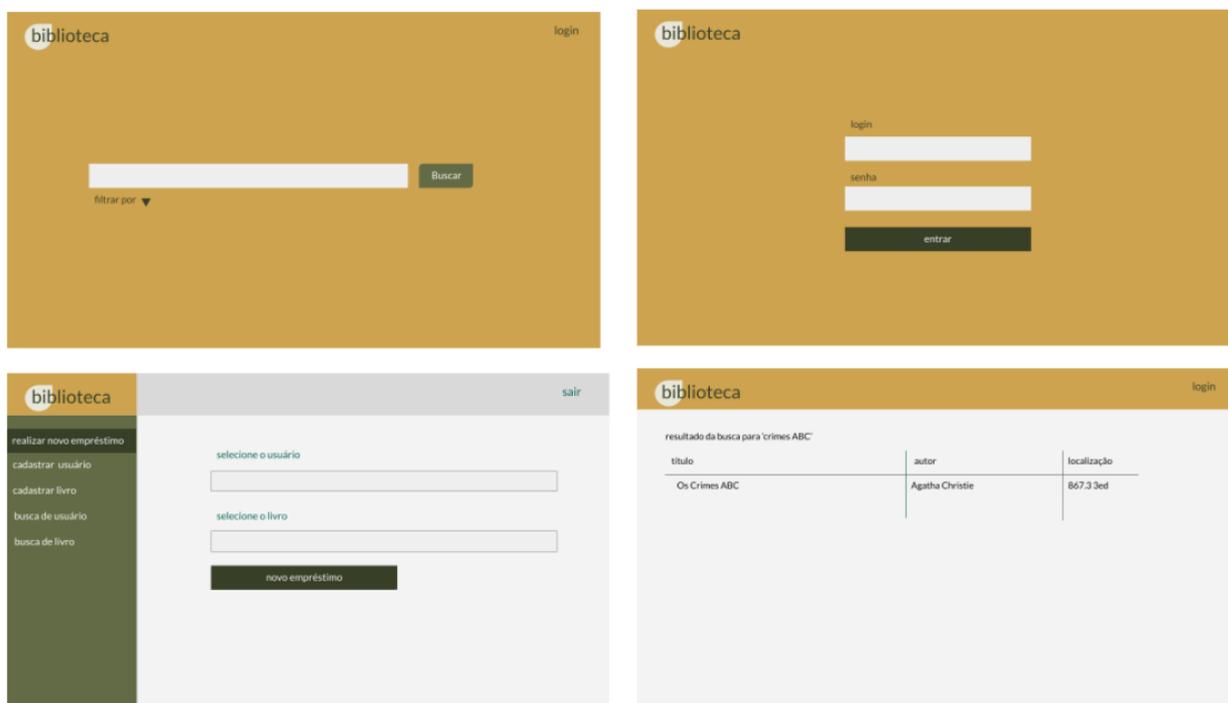
Na figura 3 temos as classes principais do projeto, são elas: usuário, livro e empréstimo, com seus atributos e relacionamentos definidos. Também temos as classes de controle ou *controller* que são as responsáveis por realizar as funções específicas de cada caso de uso. Este diagrama serviu como direcionamento para a criação dos objetos do sistema que foi implementado usando o paradigma de orientação a objetos.

3.4 PROTÓTIPOS DE TELA

Os protótipos de tela servem para dar uma noção de como o sistema ficará visualmente. A ideia é que com essas telas já pensadas previamente podemos assim focar no desenvolvimento das funcionalidades do sistema.

Na figura 4 temos alguns dos protótipos de tela desenvolvidos manualmente, utilizando o programa Microsoft PowerPoint como ferramenta de desenho, para o projeto na fase de modelagem. Eles não necessariamente representam o resultado final da interface do sistema, mas serviram de base para o desenvolvimento dela.

Figura 4 – Protótipos de tela



Fonte: Elaboração própria em 2021.

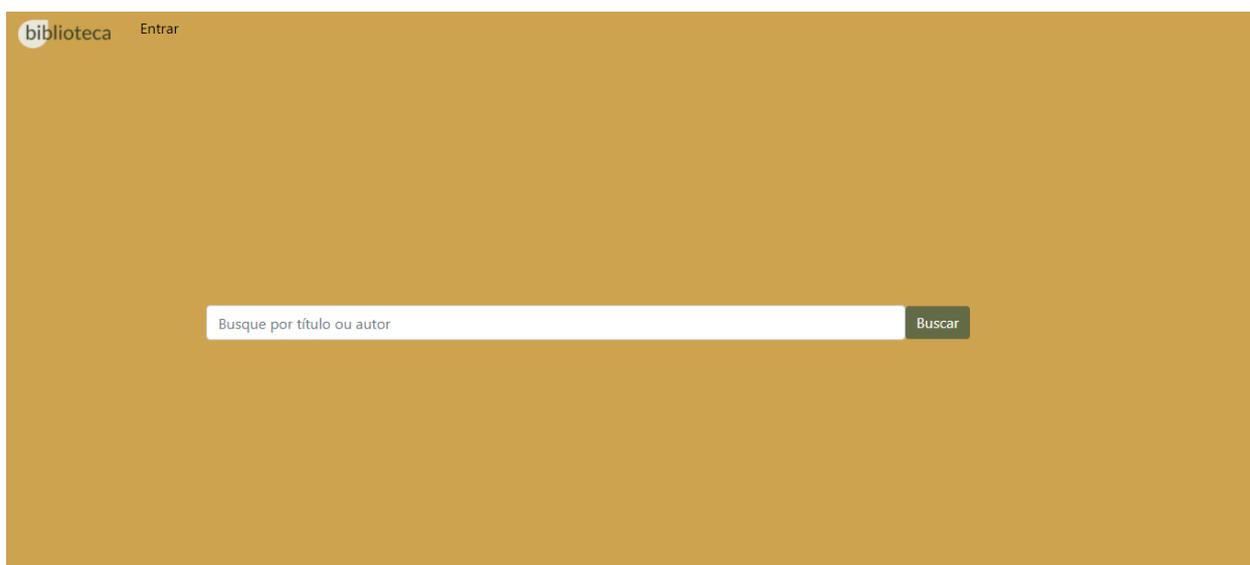
4 RESULTADOS

Nesta seção será apresentado o resultado final do sistema, apresentando as telas principais desenvolvidas e a explicação de como é o funcionamento básico do sistema.

4.1 APRESENTAÇÃO DO SISTEMA

Como resultado final, temos o sistema web desenvolvido para o gerenciamento de uma biblioteca escolar. Na figura 5, temos a tela inicial do sistema.

Figura 5 – Tela inicial do sistema



Fonte: Elaboração própria em 2021.

O usuário comum tem acesso à busca de livros, podendo pesquisar um livro cadastrado no sistema pelo seu nome ou pelo autor da obra. Para as funcionalidades de cadastro de usuários e livros ou de realização de empréstimo e devolução, é necessário possuir um cadastro como administrador e fazer o login, informando login e senha, no sistema. A autenticação é feita utilizando as variáveis de sessão dentro do projeto Spring. Na figura 6, temos a tela de login.

Figura 6 – Tela de login

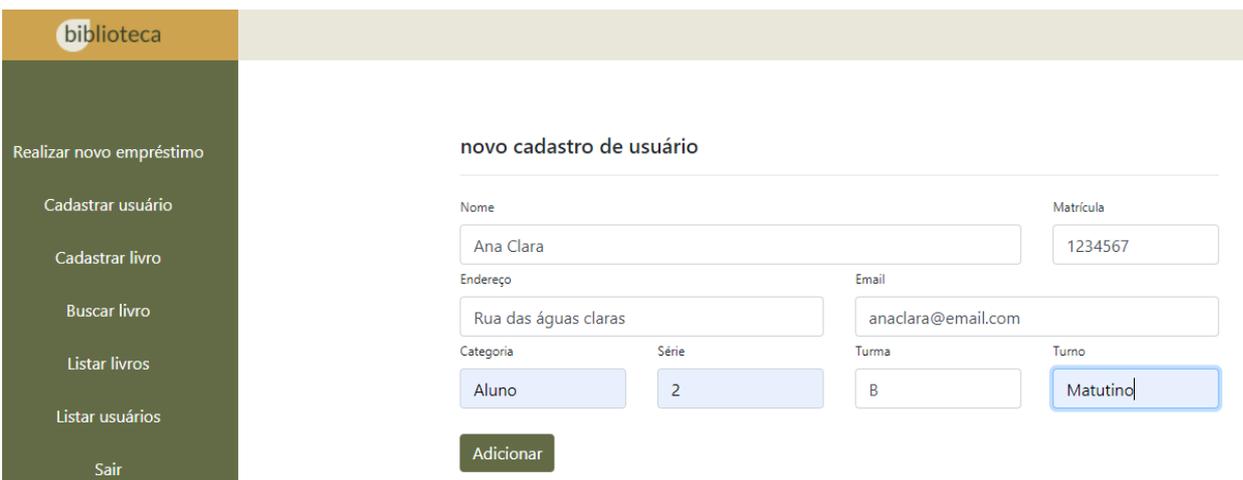


A tela de login da biblioteca apresenta um formulário centralizado em um fundo amarelo-escuro. No canto superior esquerdo, há o logotipo 'biblioteca'. O formulário contém dois campos de entrada: 'Login' e 'Senha', ambos com bordas arredondadas e fundo branco. Abaixo dos campos, há um botão verde com o texto 'Entrar' em branco.

Fonte: Elaboração própria em 2021.

Ao fazer o login, o administrador agora tem acesso a todas as funcionalidades. Na figura 7, temos a tela de cadastro de usuário. Esses usuários cadastrados podem pegar livros emprestados na biblioteca. Informações como nome, matrícula, a classe do usuário, série, turma e turno são solicitadas.

Figura 7 – Tela de cadastro de usuário



A tela de cadastro de usuário da biblioteca possui um cabeçalho com o logotipo 'biblioteca' e uma barra lateral esquerda com um menu de navegação. O formulário principal, intitulado 'novo cadastro de usuário', contém os seguintes campos:

- Nome: Ana Clara
- Matrícula: 1234567
- Endereço: Rua das águas claras
- Email: anaclara@email.com
- Categoria: Aluno
- Série: 2
- Turma: B
- Turno: Matutino

Um botão verde 'Adicionar' está localizado abaixo dos campos.

Fonte: Elaboração própria em 2021.

Na figura 8, temos a listagem dos usuários já cadastrados.

Figura 8 – Recorte da tela de listagem de usuários

Nome	Matrícula	E-mail	Categoria	Série	Turma	Turno	Status	Com Pendências	
Ana Paula	1234	anapaula@email.com	Aluno	1	A	Matutino	true	false	Ver empréstimos
João Paulo	12345	joapaulo@email.com	Professor				true	false	Ver empréstimos
Elis Regina dos Santos	123456	elisregina@email.com	Aluno	3	A	Vespertino	true	false	Ver empréstimos

Fonte: Elaboração própria em 2021.

Na figura 9, temos a tela de cadastro de livros. Os livros, para ficarem disponíveis para empréstimo, precisam ser previamente cadastrados no sistema. Informações como título, autor, ISBN, chamada, assunto e tipo de material são solicitadas.

Figura 9 – Recorte da tela de cadastro de livro

estimo

o

novο cadastro de livro

Título: Os Crimes ABC

Autor: Agatha Christie

Assunto: Romance policial

ISBN: 09876

Exemplar: 1

Volume: 1

Local: RJ

Editora: Record

Ano de publicação: 1970

Número da chamada: 891.73 D724m

Idioma: Português

Tipo de material: Livro

Forma de aquisição: Doação

Data de aquisição: 28/10/2020

Quantidade: 3

Adicionar

Fonte: Elaboração própria em 2021.

Na figura 10, temos a listagem dos livros já cadastrados.

Figura 10 – Recorte da tela de listagem de livros

Título	Autor	Assunto	Tipo de material	Nº chamada	
Os Crimes ABC	Agatha Christie	Romance policial	Livro	891.73 D724m	<input type="button" value="Novo empréstimo"/>
Hora Zero	Agatha Christie	Romance policial	Livro	880.12 C12	<input type="button" value="Novo empréstimo"/>
A Potência Segundo Nietzsche	Maite Larrauri	Filosofia	Livro	108.35	<input type="button" value="Novo empréstimo"/>
Um Dose Mortal	Agatha Christie	Romance policial	Livro	891.73 C103m	<input type="button" value="Novo empréstimo"/>
A Mulher que Roubou a Minha Vida	Marian Keyes	Romance	Livro	821.111(41)-3	<input type="button" value="Novo empréstimo"/>

Fonte: Elaboração própria em 2021.

Para realizar um empréstimo para um usuário, o administrador pode buscar o livro através da listagem ou utilizar a página de busca. Na figura 11, temos o resultado de uma busca realizada por título do livro onde foi retornado apenas um resultado.

Figura 11 – Recorte da tela de resultado de busca por título

Título	Autor	Assunto	Tipo de material	Nº chamada	
Os Crimes ABC	Agatha Christie	Romance policial	Livro	891.73 D724m	<input type="button" value="Novo empréstimo"/>

Fonte: Elaboração própria em 2021.

Ao clicar em “Novo Empréstimo”, o sistema, então, solicita a matrícula do usuário que está solicitando o livro emprestado, como mostrado na figura 12, e, para finalizar a operação, é só clicar no botão “Realizar Empréstimo”.

Figura 12 – Recorte da tela de empréstimo

The screenshot shows a web interface with a header bar (orange and grey) and a dark green sidebar on the left. The main content area contains a form with the label "Informe a matrícula" above a text input field. To the right of the input field is a dark green button with the text "Realizar empréstimo".

Fonte: Elaboração própria em 2021.

Para visualizar os empréstimos de algum usuário, é necessário ir à lista de usuários (ver figura 8) e clicar no botão “Ver empréstimos”. Será mostrada uma lista dos empréstimos do usuário selecionado, como na figura 13.

Figura 13 – Recorte da tela de empréstimo por usuário

The screenshot shows a web interface with a header bar (orange and grey) and a dark green sidebar on the left. The main content area displays a table with the following columns: Matrícula, Livro, Autor, Data do Empréstimo, Data Prevista da Devolução, and Data de Devolução. A single row of data is visible, and a button labeled "Realizar devolução" is positioned to the right of the row.

Matrícula	Livro	Autor	Data do Empréstimo	Data Prevista da Devolução	Data de Devolução
1234	Os Crimes ABC	Agatha Christie	01/04/2021	16/04/2021	

Fonte: Elaboração própria em 2021.

Caso o administrador queira marcar como devolvido o livro emprestado, basta clicar no botão “Realizar Devolução”. Caso o usuário esteja realizando a devolução após a data prevista, ele sofrerá uma suspensão, não podendo fazer nenhum empréstimo dentro de um prazo de 7 dias. Após esse período, o sistema retira o usuário da suspensão automaticamente.

Após finalizar o uso do sistema, o administrador pode clicar em “Sair” e a sua sessão será encerrada. Assim, temos o funcionamento básico do sistema desenvolvido.

5 CONSIDERAÇÕES FINAIS

O desenvolvimento do presente trabalho permitiu uma análise de como uma solução tecnológica pode ser utilizada como uma ferramenta para a melhoria no uso da biblioteca por parte de seus usuários e ajudá-la a alcançar seus objetivos de aprendizagem enquanto ambiente informacional e de promoção de leitura e pesquisa.

Para isso, foi realizado o levantamento dos requisitos e foi feita a análise das tecnologias e, assim, definido que a criação de um sistema web que permitisse o controle do acervo e de empréstimos aos usuários poderia ser uma solução ideal e que atenderia às necessidades da biblioteca. Assim foi iniciado o desenvolvimento do sistema.

A solução final conta com uma área pública de consulta de acervo e uma área restrita para as funções de registro de livros e de usuários, assim como a área de controle de empréstimos. Já contando com as funcionalidades que suprem as necessidades básicas da biblioteca, o projeto possui margem para melhorias e adições de novas funcionalidades, como por exemplo a adição de uma área para recomendação de livros e uma área de acesso para os usuários, e que possam vir a ser identificadas futuramente.

REFERÊNCIAS

- AFONSO, Alexandre. O que é Spring MVC? *In*: ALGAWORKS. [S. l.], 2017. Disponível em: <https://blog.algaworks.com/spring-mvc/>. Acesso em: 8 mar. 2021.
- APACHE Maven Software. *In*: MAVEN. [S. l.], c2021. Disponível em: <https://maven.apache.org/>. Acesso em: 8 mar. 2021.
- ARANA, Alba Regina de Azevedo; KLEBIS, Augusta Boa Sorte Oliveira. A importância do incentivo à leitura para o processo de formação do aluno. *In*: CONGRESSO NACIONAL DE EDUCAÇÃO, 12., 2015, Paraná. **Anais [...]**. Paraná: PUCPR, 2015. p. 26669-26686. Disponível em: https://educere.bruc.com.br/arquivo/pdf2015/17264_7813.pdf. Acesso em: 10 fev. 2021.
- BATISTA, Rafael. Importância da leitura. *In*: BRASIL Escola. [S. l.], c2021. Disponível em: <https://brasilecola.uol.com.br/ferias/a-importancia-leitura.htm>. Acesso em: 10 fev. 2021.
- CUENCA, Angela Maria Belloni *et al.* Biblioteca virtual e o acesso às informações científicas e acadêmicas. **Revista USP**, São Paulo, n. 80, p. 72-83, dez./fev. 2008-2009. Disponível em: <http://www.revistas.usp.br/revusp/article/download/13717/15535/>. Acesso em: 10 fev. 2021.
- FOWLER, Martin. **UML Essencial**: um breve guia para a linguagem-padrão de modelagem de objetos. 3. ed. São Paulo: Bookman, 2006.
- JESUS, Paulo Sérgio de. O papel da biblioteca escolar na formação do leitor crítico. **Educação, Gestão e Sociedade**: Revista da Faculdade Eça de Queirós, São Paulo, ano 5, n. 17, fev. 2015. Disponível em: http://uniesp.edu.br/sites/_biblioteca/revistas/20170509161303.pdf. Acesso em: 10 fev. 2021.
- MANYS, Heliton. **A importância da leitura para o aprimoramento da escrita no ensino médio**. [S. l.], [201-?]. Disponível em: http://uniesp.edu.br/sites/_biblioteca/revistas/20170601131201.pdf. Acesso em: 10 fev. 2021.
- POSTGRESQL. [S. l.], c2021. Disponível em: <https://www.postgresql.org/about/>. Acesso em: 8 mar. 2021
- PRESSMAN, Roger S. **Engenharia de software**: uma abordagem profissional. 7. ed. Porto Alegre: AMGH, 2011.
- SILVIA, Patrícia Maria. O comportamento dos usuários de bibliotecas em sistemas de informação. **Transinformação**, Campinas, v. 20, n. 3, p. 255-263, set./dez. 2008. Disponível em: <https://www.scielo.br/pdf/tinf/v20n3/04.pdf>. Acesso em: 11 fev. 2021.

SOMMERVILLE, Ian. **Engenharia de software**. 8. ed. São Paulo: Pearson, 2007.

THYMELEAF. [S. l.], 2020. Disponível em: <https://www.thymeleaf.org/>. Acesso em: 8 mar. 2021.

TONDING, Fabiana John; VANZ, Samile Andréa de Souza. Plataformas de Serviços de Bibliotecas: a evolução dos sistemas para gerenciamento de bibliotecas. **Perspectivas em Ciência da Informação**, Belo Horizonte, v. 23, n. 4, p. 73-96, out./dez. 2018. Disponível em: <https://www.scielo.br/pdf/pci/v23n4/1413-9936-pci-23-04-00073.pdf>. Acesso em: 10 fev. 2021.

WEB Applications. *In*: SPRING. [S. l.], c2021. Disponível em: <https://spring.io/web-applications>. Acesso em: 8 mar. 2021.