

**CENTRO PAULA SOUZA  
FACULDADE DE TECNOLOGIA DE FRANCA  
“Dr. THOMAZ NOVELINO”**

**TECNOLOGIA EM ANÁLISE E DESENVOLVIMENTO DE SISTEMAS**

**KARLA MARIANA VISCONDI CATARINO  
THABATA MORAES SILVA**

**SISTEMA GERENCIAL DE BIBLIOTECA**

Trabalho de Graduação apresentado à Faculdade de Tecnologia de Franca - “Dr. Thomaz Novelino”, como parte dos requisitos obrigatórios para obtenção do título de Tecnólogo em Análise e Desenvolvimento de Sistemas.

Orientador: Prof. Me. Fausto Gonçalves Cintra

**FRANCA/SP**

**2024**

# SISTEMA GERENCIAL DE BIBLIOTECA

Karla Mariana Viscondi Catarino<sup>1</sup>

Thabata Moraes Silva<sup>2</sup>

Fausto Gonçalves Cintra<sup>3</sup>

## Resumo

Este trabalho tem como objetivo desenvolver a documentação e implementação de um sistema de informação para a Web que ampare a gestão de uma biblioteca, assegurando a integridade e a organização do acervo, bem como as respectivas operacionalização e acessibilidade do usuário. O projeto conta com uma análise de viabilidade, utilizando ferramentas como *Business Model Canvas*, SWOT e o método 5W2H. Foram levantados requisitos detalhados, modelados por meio da notação BPMN, e documentados requisitos funcionais, não funcionais e regras de negócio. O sistema resultante permite aos usuários realizar consultas de disponibilidade de livros, efetuar reservas, empréstimos e devoluções. O projeto não só demonstra a aplicação eficaz de metodologias e ferramentas modernas de desenvolvimento, mas também contribui para a melhoria dos serviços oferecidos pela biblioteca da instituição cliente.

**Palavras-chave:** análise e desenvolvimento de sistemas; engenharia de software; sistema web; tecnologia da informação.

## Abstract

*This project aims to develop the documentation and implementation of a web-based information system to support library management, ensuring the integrity and organization of the collection, as well as user's operability and accessibility. The project includes a feasibility analysis using tools such as Business Model Canvas, SWOT analysis, and the 5W2H method. The detailed requirements of this paper were gathered, modeled using BPMN notation, and documented comprising functional requirements, non-functional requirements, and business rules. The resulting system allows users to query book availability, make reservations, borrow, and return books. The project not only demonstrates the effective application of modern development methodologies and tools but also contributes to improve the services offered by the client institution's library.*

**Keywords:** *information technology; software engineering; systems analysis and development; web system.*

---

<sup>1</sup> Graduanda em Análise e Desenvolvimento de Sistemas pela Fatec Dr Thomaz Novelino – Franca/SP. Endereço eletrônico: karla.viscondi@gmail.com

<sup>2</sup> Graduanda em Análise e Desenvolvimento de Sistemas pela Fatec Dr Thomaz Novelino – Franca/SP. Endereço eletrônico: thabatams16@gmail.com

<sup>3</sup> Docente em Análise e Desenvolvimento de Sistemas pela Fatec Dr Thomaz Novelino – Franca/SP. Endereço eletrônico: fausto.cintra@fatec.sp.gov.br.

## 1 Introdução

À memória biológica, que pertence à espécie, e à memória cerebral, que é do indivíduo, acrescentou-se a biblioteca, como memória coletiva das experiências existenciais, científicas e culturais, seja do indivíduo, seja da sociedade. (Serrai, 1975, p.142).

As bibliotecas tiveram origem há muito tempo, precederam o período cristão. As evidências mais tangíveis de sua existência são as bibliotecas de Nínive (século VI a.C.) e Alexandria (século II a.C.). Elas já apresentavam sistemas de organização de suas coleções, a primeira possuía seções, enquanto a segunda era organizada por classes, com os autores dispostos em ordem alfabética (Serrai, 1975).

Com o passar do tempo, alguns acontecimentos externos como movimentos sociais, políticos e militares, culminaram no fechamento de muitas bibliotecas antigas do ocidente, muitas delas foram substituídas por bibliotecas cristãs. Por outro lado, no oriente, as bibliotecas dos mosteiros foram mais bem sucedidas ao manter seus acervos para gerações futuras (Serrai, 1975).

A partir da Renascença, com a institucionalização da imprensa, as bibliotecas não mais possuíam apenas o objetivo de custodiar e preservar os registros, elas passaram a oferecer informações aos mais diversos públicos, colaborando com a democratização do acesso (Santa Anna, 2015).

Essa breve contextualização nos mostra que a sociedade, desde seus primórdios, se empenha em registrar seus dados, informações e conhecimentos. Nesse contexto é importante refletir sobre o impacto do cenário digital nas bibliotecas.

Moraes (2012, p. 60) aponta que:

A Internet disponibilizou diferentes panoramas para o saber, melhorou o acesso e o tempo ao conhecimento, mudaram os modelos de relação entre autor e leitor, novas formas de ver o conteúdo. Rompemos barreiras culturais e geográficas, de modo que a informação não está mais limitada.

Essas expansões de barreiras não poderiam ser imaginadas nem alcançadas sem a evolução das ferramentas de tecnologia da informação, e esses recursos são considerados “indispensáveis ao avanço científico, social, tecnológico e econômico” (Perfetto; Reis; Paletta, 2023, p.3).

Ao abordar os conceitos, Perfetto, Reis e Paletta (2023, p.3) pontuam que o

conhecimento é gerado a partir da tríade – dado - informação - conhecimento, o que podemos verificar aqui é que a informação começa como um dado, mas,

ela se transforma em conhecimento a partir do momento que o indivíduo transforma em algo útil para si ou para a sociedade.

Sendo assim, a “[...] informação não é conhecimento, mas, sem ela, não há conhecimento, porque ela é a sua base. O conhecimento é construído com informações” (Costa *et al.*, 2018, p. 198). Para além disso, uma boa gestão dos dados e informações se mostra essencial para que a construção de conhecimento seja realizada de forma satisfatória e relevante.

De forma geral, as Tecnologias de Informação e Comunicação (TIC) se mostram essenciais nas organizações, as mantêm ativas e competitivas no mercado e são responsáveis por fundamentar o planejamento organizacional e tomadas de decisão. Dessa forma, elevam o valor da informação e conhecimento e se mostram cruciais para o desenvolvimento das organizações (Silva; Caldas, 2022).

No contexto das bibliotecas o mesmo raciocínio pode ser aplicado. Pois as TIC são indispensáveis para a organização e recuperação de informações, além de tornar o usuário mais independente na busca dessas informações e construção de conhecimento (Souza, 2011).

Em resumo, a tecnologia da informação é uma ferramenta essencial para a gestão de bibliotecas, permitindo uma maior eficiência e eficácia na organização e controle do acervo, além de proporcionar um maior acesso e flexibilidade aos usuários.

Historicamente, a biblioteca teve grande impacto e contribuição no desenvolvimento da sociedade, contribuiu e segue contribuindo na perpetuação da nossa história e conhecimento. Dessa forma, com a expansão das tecnologias, é importante manter o olhar para as bibliotecas e que haja um esforço de adaptação para os contextos atuais tamanho impacto social dessa instituição.

## 1.1 Termo da Abertura do Projeto (TAP)

O Termo da Abertura do Projeto é fundamental no gerenciamento do projeto formalizando seu início e reunindo as informações chaves para a sua execução.

### 1.1.1 Identificação do Projeto

Gestão de empréstimos e devoluções de livros por alunos da biblioteca da Universidade Aberta do Brasil (UAB) - Polo Franca.

### 1.1.2 Justificativa do Projeto

A gestão de uma biblioteca pode ser uma tarefa desafiadora, especialmente quanto à manutenção da organização e acessibilidade do acervo. Com a grande quantidade de livros, periódicos, materiais audiovisuais e outros tipos de mídia que as bibliotecas possuem, é necessário ter um sistema que possa assegurar a integridade e organização do acervo e operacionalização do acesso. Por essa razão, desenvolver uma ferramenta de tecnologia digital que possa auxiliar nessa tarefa é fundamental.

### 1.1.3 Objetivos e Metas do Projeto

- a) Automatizar e agilizar o processo de empréstimo e devolução de livros, reduzindo o tempo de espera dos usuários e melhorando a eficiência do serviço.
- b) Melhorar o controle e a segurança do acervo, garantindo que os livros emprestados estejam registrados no sistema e possam ser facilmente localizados.
- c) Facilitar a gestão dos prazos de empréstimo e devolução, de modo a permitir que os usuários saibam com antecedência quando devem devolver os livros, a fim de evitar atrasos ou punições.
- d) Proporcionar uma experiência positiva para os usuários, com um sistema fácil de usar, intuitivo e que ofereça informações precisas e atualizadas sobre o acervo e os empréstimos.
- e) Possibilitar a integração com outros sistemas e plataformas, como o sistema de gestão de bibliotecas, para uma gestão integrada e mais eficiente da biblioteca como um todo.

### 1.1.4 Descrição do Projeto

O projeto de desenvolvimento de uma ferramenta de tecnologia digital de informação que ampare o gerenciamento de uma biblioteca tem como objetivo principal a melhoria da gestão do acervo e da operacionalização do acesso à biblioteca. O projeto visa desenvolver uma solução tecnológica para automatizar e agilizar o processo de empréstimo e devolução de livros, proporcionando uma experiência positiva para os usuários e garantindo a integridade e organização do acervo.

A ferramenta de tecnologia digital deve permitir a gestão dos empréstimos e devoluções, controlando os prazos, as punições aos usuários em atraso e as reservas de livros. Ela deve ser capaz de localizar os livros emprestados e disponibilizar informações precisas e atualizadas sobre o acervo e os empréstimos. A ferramenta deve ser fácil de usar, intuitiva, além de garantir a privacidade e segurança dos dados dos usuários.

O projeto deve incluir a definição dos requisitos do sistema, a análise e modelagem do processo de empréstimo e devolução de livros.

#### 1.1.5 Premissas

- a) A solução deve ser desenvolvida considerando as necessidades e requisitos específicos da biblioteca em questão, garantindo que a ferramenta atenda às demandas do público atendido pela biblioteca e possa ser utilizada de forma eficiente.
- b) O sistema deve estar em conformidade com as normas e regulamentações vigentes para a gestão de bibliotecas, garantindo a legalidade e segurança da solução.
- c) O projeto deve ser gerenciado de forma eficiente, com acompanhamento das atividades, prazos e metas estabelecidos, para garantir o sucesso da implementação da solução.

#### 1.1.6 Restrições

- a) Restrições de prazo: o projeto deve ser concluído dentro do prazo estabelecido, levando em consideração as datas de entrega e disponibilidade de recursos.
- b) Restrições de escopo: o escopo do projeto deve ser definido de forma clara e concisa, evitando a inclusão de funcionalidades desnecessárias que possam impactar negativamente o sucesso do projeto.

#### 1.1.7 Escopo

- a) Desenvolver uma ferramenta de tecnologia digital que permita o gerenciamento eficiente do acervo da biblioteca, na gestão de empréstimos e devoluções livros.

- b) Desenvolver uma plataforma digital que possibilite o acesso dos usuários ao catálogo da biblioteca, permitindo a busca e reserva de livros, visualização do histórico de empréstimos e devoluções.

#### 1.1.8 Resultados Esperados

- a) Melhoria na gestão do acervo da biblioteca: a ferramenta de gestão de empréstimos e devoluções de livros permitirá um gerenciamento mais eficiente do acervo da biblioteca, de modo a elevar a organização e a integridade do acervo.
- b) Melhoria no acesso dos usuários à biblioteca: a plataforma digital permitirá que os usuários acessem o catálogo da biblioteca de forma fácil e rápida, realizando pesquisas e reservas de livros online.
- c) Melhoria na qualidade dos serviços prestados pela biblioteca: com a implementação da ferramenta, espera-se que a qualidade dos serviços prestados pela biblioteca seja aprimorada, resultando em maior satisfação dos usuários.

#### 1.1.9 Não-escopo

- a) Compra e instalação de equipamentos: este projeto não deve contemplar a compra e instalação de equipamentos de hardware, tais como computadores, servidores e dispositivos móveis.
- b) Construção de infraestrutura física: a construção de uma infraestrutura física, como uma nova sala para a biblioteca ou a instalação de sistemas elétricos e hidráulicos, também não deve ser considerada como escopo deste projeto.
- c) Desenvolvimento de novos softwares: O desenvolvimento de novos softwares que não estejam diretamente relacionados à gestão do acervo da biblioteca ou à operacionalização do acesso também não deve ser considerado como escopo do projeto.

#### 1.1.10 Prazos Previstos

Data de início: Fevereiro/ 2023.

Data de término: Maio/ 2024.

#### 1.1.11 Custo Previsto

Devido à decisão de não implementar o projeto, não serão necessários investimentos financeiros. No entanto, as métricas realizadas neste projeto indicam o valor de R\$ 4.114,80 em investimentos para o desenvolvimento da solução sistêmica.

## **2 Viabilidade do Projeto**

As seções a seguir indicam atividades realizadas para a boa condução do projeto, a citar o levantamento de fraquezas e pontos fortes da organização escolhida para o desenvolvimento da solução sistêmica, bem como o plano de ação e a viabilidade do projeto, considerando-se as fraquezas a serem trabalhadas.

### **2.1 Canvas de Negócio (*Business Model Canvas* - BMC)**

O Business Model Canvas, criado por Alexander Osterwalder e Yves Pigneur, revolucionou a forma como empreendedores e empresas constroem seus negócios. Esta ferramenta visual permite descrever, visualizar, avaliar e alterar modelos de negócios de forma eficaz.

Composto por nove blocos, incluindo segmentos de clientes, proposta de valor e fontes de receita, o Canvas possibilita uma compreensão rápida e completa do negócio. Além disso, estimula a colaboração entre várias partes interessadas, facilitando a análise e o planejamento do modelo. A figura 1 representa o Canvas desse projeto.

Essa ferramenta é amplamente utilizada para explorar novas ideias, inovar em modelos existentes e comunicar de forma clara a estratégia de uma empresa. Em suma, o Business Model Canvas é essencial para entender como uma empresa cria, entrega e captura valor. (Osterwalder; Pigneur, 2013)

Figura 1 - Canvas

# Business Model Canvas



Fonte: Autores

1- Segmentos de Clientes: identifica os diferentes grupos de organizações ou pessoas que uma corporação planeja atender com seu produto ou serviço. Os segmentos de clientes podem ser divididos com base em necessidades, preferências, características de compras, etc.

2- Proposta de Valor: descreve os produtos ou serviços oferecidos para atender às necessidades ou resolver problemas de seus segmentos de cliente, ou seja, destaca os benefícios que diferenciam a corporação das concorrentes.

3- Canais: aborda os meios pelos quais a corporação se relaciona com seus clientes.

4- Relacionamento com Clientes: descreve os tipos de relacionamento e interação que a corporação terá ao longo do ciclo de vida do cliente, incluindo suporte ao cliente, serviços pós-venda, entre outros.

5- Fontes de Receita: descreve maneiras pelas quais a corporação gera receita com base na proposta de valor entregue aos clientes, incluindo vendas diretas, assinaturas, publicidade, etc.

6- Recursos Principais: lista os ativos, recursos e capacidades essenciais para fornecer sua proposta de valor e exercer o negócio de forma eficaz, sejam recursos físicos, humanos, intelectuais ou financeiros.

7- Atividades Principais: descreve os processos e as ações que a corporação deve executar para gerir eficazmente e fornecer sua proposta de valor aos clientes.

8- Parcerias Principais: identifica as colaborações com outras corporações que ajudem a alcançar seus objetivos.

9- Estrutura de Custos: lista os custos relacionados à operação do negócio e à entrega da proposta de valor aos clientes.

## 2.2 Matriz SWOT

A análise SWOT consiste em avaliar as forças e fraquezas internas da organização, bem como as oportunidades e ameaças externas do ambiente em que atua. Essa avaliação é feita através de um mapeamento dos pontos fortes e fracos da empresa, além das oportunidades e ameaças do mercado ou contexto em que está inserida.

A figura 2 representa o levantamento de todas as forças, fraquezas, oportunidades e ameaças que foram observadas na problemática da instituição para a qual foi proposta a solução sistêmica, ao observar de forma holística a biblioteca da UAB - Polo Franca.

Figura 2 - Matriz SWOT



Fonte: Autores

A partir do levantamento de todas as forças, fraquezas, oportunidades e ameaças, uma fraqueza foi escolhida para o desenvolvimento da solução sistêmica, a fraqueza referente ao processo arcaico de empréstimos.

Na instituição em questão, todos os empréstimos são realizados de forma manual. Esse formato pode representar um procedimento vulnerável no quesito segurança da informação, além de representar um risco no que tange a conservação de integridade do acervo.

### 2.3 Plano de Ação 5W2H do Projeto

É uma técnica utilizada para planejamento e definição de ações, projetos ou tarefas de forma clara e objetiva. O nome é um acrônimo que vem do inglês e representa as sete perguntas que devem ser respondidas ao planejar uma ação: O quê, Por quê, Onde, Quem, Quando, Como e Quanto custa.

Diante da problemática observada na biblioteca da UAB - Polo Franca, quanto à inexistência de um sistema para a biblioteca da instituição, foi proposto um plano de

ação relacionado à fraqueza do processo arcaico de empréstimos, com enfoque no controle de empréstimo e devolução de livros de forma sistemática.

O Quadro 1 representa os detalhes do plano de ação proposto para o desenvolvimento da solução sistêmica para atender a demanda da biblioteca da UAB - Polo Franca.

Quadro 1 - Plano de ação

PLANO DE AÇÃO - 5W2H							
<b>PLANO DE AÇÃO:</b>	5W2H						
<b>DATA PREVISÃO:</b>	Fev 2023 /Maio 2024	<b>DATA REALIZADA:</b>	Maio 2024				
<b>RESPONSÁVEL:</b>	Equipe do projeto						
<b>OBJETIVO:</b>	Controlar empréstimos e devoluções de livros.						
5W					2H		Status
O quê? (What?)	Por quê? (Why?)	Onde? (Where?)	Quem? (Who?)	Quando? (When?)	Como? (How?)	Quanto Custa? (How much?)	
Controlar empréstimos e devoluções dos livros.	Organizar; Assegurar integridade do acervo; Gerenciar o fluxo de empréstimo.	UAB Polo Franca – máquina local	Equipe do projeto	Fevereiro a julho de 2023.	Entrevistar equipe responsável pela manutenção da biblioteca e elaborar a documentação que ampare a melhoria	Devido à decisão de não implementar o projeto, não serão necessários investimentos financeiros.	

Fonte: Autores

O plano de ação teve como objetivo organizar, assegurar a integridade do acervo, além de gerenciar o fluxo de empréstimo. Para isso, foi realizada uma entrevista com a equipe responsável pela manutenção da biblioteca, além da elaboração de documentações que amparam a melhoria. O plano de ação foi pensado e estruturado para se desenvolver entre fevereiro e junho de 2023.

## 2.4 Viabilidade

Baseando-se nas fraquezas encontradas na organização selecionada para a proposição da solução sistêmica, principalmente quanto aos procedimentos arcaicos para empréstimos, o presente projeto propôs uma alternativa para melhor gerenciamento da biblioteca da UAB-Polo Franca, quanto aos processos de empréstimo e devolução de livro.

Para tanto, foram feitas visitas e entrevistas com funcionários envolvidos com a UAB, de modo a facilitar a obtenção das reais necessidades dos usuários diretamente envolvidos com a biblioteca, além de compreender as regras de negócio da instituição.

Assim, foi delineado o processo em vigor na instituição (BPMN processual - Figura 3) e este foi tomado como ponto de partida para a proposição de melhorias, por meio da solução sistêmica, disposta no BPMN (Figura 4).

O procedimento de empréstimo e devolução de livros, que é feito de forma manual na instituição, recebe, então, a proposta de ser realizado por meio de um sistema que conta com a possibilidade de o usuário reservar livros ainda de seu ambiente doméstico, assim como, fazer uso da colaboração de funcionários do polo da biblioteca da UAB para os trâmites. Também na instituição, alguns computadores seriam destinados aos usuários para acesso ao acervo da biblioteca e reservas.

Os livros do acervo são disponibilizados em tela, após *login* no sistema, para que o usuário selecione, no máximo, duas unidades, e a confirmação, tanto de empréstimo, quanto da retirada do item são finalizadas após digitação de senha válida do usuário, além de o sistema notificar o usuário, via e-mail, sobre os procedimentos realizados na biblioteca.

Dessa forma, é possível ter controle sobre a data da reserva, empréstimo, além da data de devolução dos livros, o que garante melhor gerenciamento sobre o fluxo de livros da instituição, e até mesmo da aplicação de sanções ao usuário caso este não siga as regras da instituição quanto aos prazos estabelecidos para devolução de livros e cancelamento de reservas.

Com as melhorias propostas pela solução sistêmica, a biblioteca da UAB-Polo Franca será modernizada, aproveitando as tecnologias da informação para oferecer um acesso seguro, facilitado e intuitivo aos alunos. Isso incentivará o uso da biblioteca e contribuirá para a formação acadêmica dos estudantes.

### 3 Levantamento de Requisitos

A etapa de elicitaco (ou levantamento) de requisitos   bastante importante no desenvolvimento de software, pois   o momento em que so identificadas e documentadas todas as necessidades do cliente, de forma detalhada, al m de suas expectativas quanto ao sistema a ser desenvolvido.

Com o levantamento de requisitos,   poss vel compreender as funcionalidades do sistema que iro atender  s regras do neg cio e oferecer uma boa experi ncia ao usu rio.

#### 3.1 Elicitaco e especificaco dos Requisitos

A elicitaco de requisitos pode ser realizada por diferentes t cnicas, conforme necessidade. A seguir, esto descritas, segundo Brum e Pena (2011), as principais t cnicas de elicitaco de requisitos:

##### 3.1.1 M todos de conversaco

- a) *Entrevista*:   uma t cnica bastante tradicional, simples de utilizar.   uma discusso do projeto a ser desenvolvido, de forma detalhada e planejada, com perguntas diversas que possibilitem a compreenso da real necessidade do entrevistado. As perguntas podem seguir ordens diferentes das que foram programadas; perguntas podem ser inclu das ou exclu das no momento da entrevista, conforme o andamento e esclarecimento de d vidas.
- b) *Wokshop*:   uma t cnica realizada em grupo, com uma reunio estruturada contendo as partes interessadas e os analistas para que os requisitos do sistema a ser desenvolvido sejam obtidos.
- c) *Brainstorm*: geralmente, esta t cnica   utilizada em conjunto ao *workshop* e envolve as opini es, palpites e viv ncias anteriores de diferentes pessoas interessadas no desenvolvimento do sistema.
- d) *Question rio*: utilizado quando   necess rio obter informa es e opini es de um grande n mero de pessoas. As quest es so feitas por escrito e direcionadas aos participantes, de forma padronizada, para obten o de uniformidade nas respostas.

##### 3.1.2 M todos de observa o

- a) *Etnografia*: possibilita um entendimento completo, denso e detalhado sobre as opera es do local, bem como a cultura e os grupos sociais envolvidos na

organização; possibilita a compreensão da política organizacional e a cultura de trabalho.

- b) Observação: envolve a visitação do local para o qual a solução sistêmica será desenvolvida a fim de coletar informações de acordo com o cotidiano das operações e processos realizados.

Para o desenvolvimento do presente trabalho, as técnicas de elicitación de requisitos escolhidas pela dupla foram a entrevista e observação. As 13 perguntas feitas na entrevista foram elaboradas e discutidas, de modo que fosse possível angariar o maior número de detalhes e as reais necessidades da biblioteca da UAB-polo Franca, representada por Daniela, coordenadora da instituição.

No momento da entrevista, a coordenadora da UAB - Polo Franca externalizou, de forma clara, as regras de negócio e os processos essenciais a serem desenvolvidos na solução sistêmica. Alguns ajustes foram necessários ao longo do desenvolvimento do projeto, porém a coordenadora sempre se mostrou solícita e interessada em solucionar quaisquer questionamentos.

Além da entrevista, a observação do local também foi realizada pela dupla e, principalmente, por Karla, que realiza seu estágio no local e, assim, pôde contribuir com detalhes sobre as operações e processos que são executados diariamente na UAB - Polo Franca. As questões da entrevista, bem como as respostas obtidas estão apresentadas no Apêndice B (p. 62).

### 3.2 BPMN

BPMN (*Business Process Model and Notation*) é uma notação gráfica amplamente utilizada para modelagem de processos de negócio. Ela foi projetada para representar visualmente o fluxo de trabalho, as atividades, as decisões, os eventos e as interações envolvidas em um processo de negócio.

A importância do BPMN, segundo Pressman (2021), reside no fato de que ele permite uma comunicação clara e precisa entre diferentes partes interessadas, incluindo analistas de negócio, gerentes de projeto, desenvolvedores e usuários finais. O BPMN oferece uma linguagem comum para descrever e documentar os processos de negócio, facilitando a compreensão e a colaboração entre as equipes envolvidas.

Os principais benefícios do BPMN incluem:

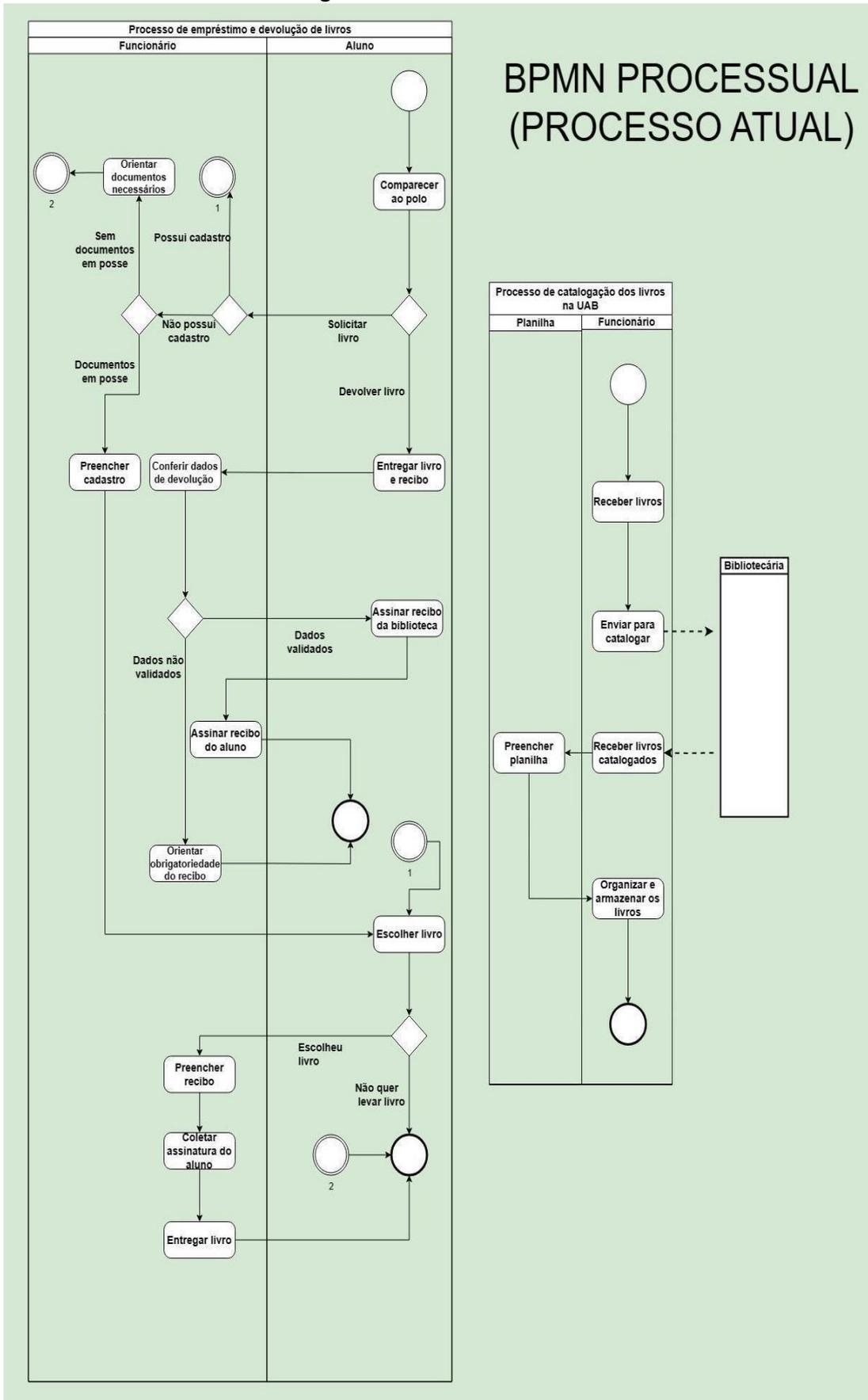
- a) Compreensão e documentação clara dos processos de negócio: o BPMN fornece uma representação visual intuitiva e padronizada dos processos, facilitando a compreensão e a documentação dos fluxos de trabalho.
- b) Colaboração e comunicação efetivas: o BPMN permite que todas as partes interessadas, incluindo usuários finais, analistas e desenvolvedores, se comuniquem de forma clara e eficaz, compartilhando uma linguagem comum para a modelagem de processos de negócio.
- c) Análise e melhoria de processos: o BPMN possibilita a identificação de gargalos, redundâncias, ineficiências e oportunidades de melhoria nos processos de negócio, permitindo que sejam otimizados e aprimorados.
- d) Automação de processos: o BPMN serve como uma base para a automação de processos de negócio, permitindo que as organizações transformem seus modelos em sistemas de software executáveis.

Em resumo, o BPMN desempenha um papel crucial na modelagem e no gerenciamento de processos de negócio. Ele oferece uma notação gráfica padronizada e compreensível que facilita a colaboração, a documentação, a análise e a melhoria dos processos, contribuindo para a eficiência e o sucesso das organizações (Pressman, 2021).

### 3.2.1 BPMN processual da biblioteca da UAB - Polo Franca

O BPMN processual dispõe como o processo de cadastro e empréstimo era realizado no primeiro semestre de 2023 (Figura 3).

Figura 3 - BPMN Processual



Fonte: Autores (2023)

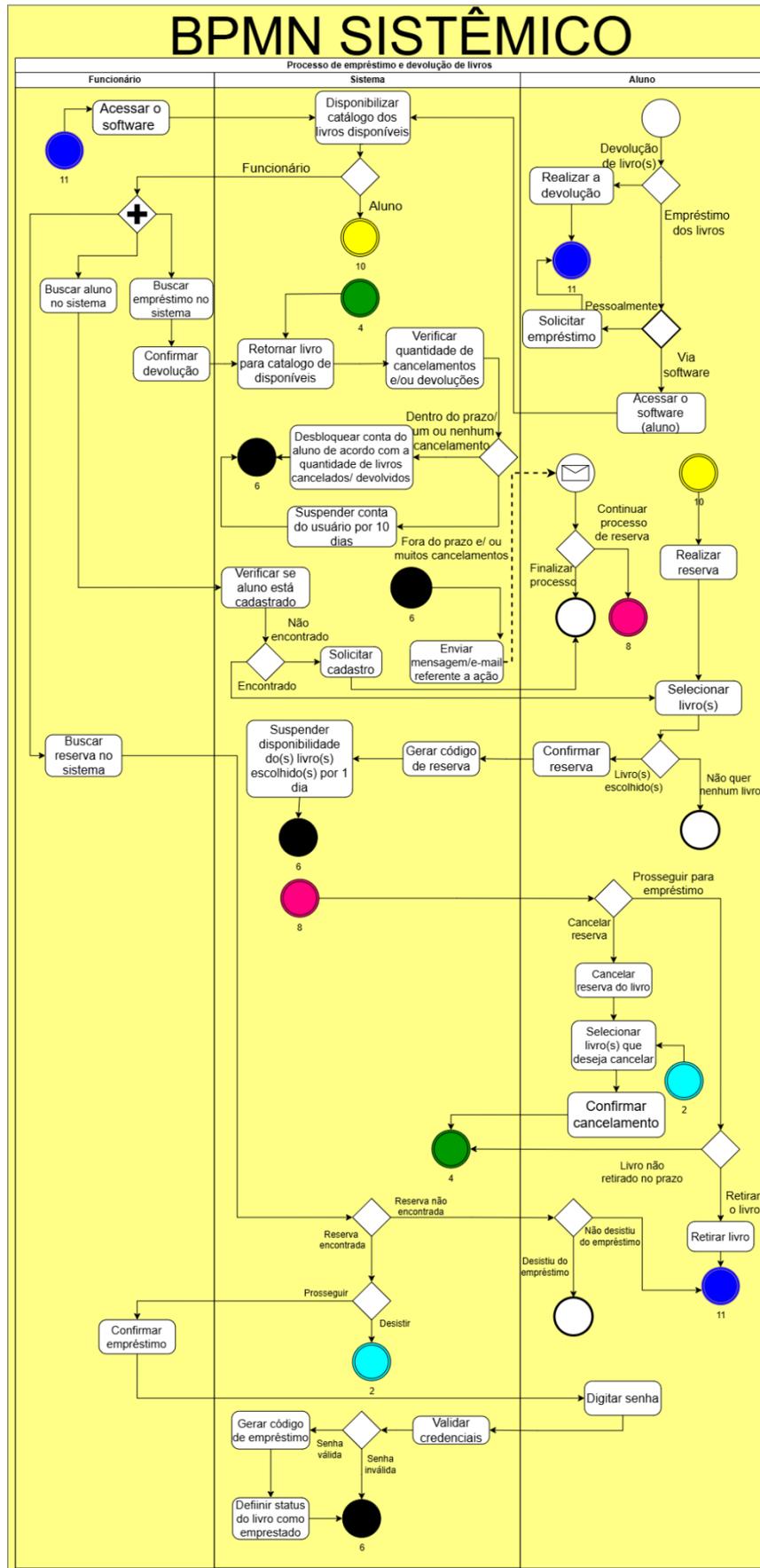
A partir da análise do BPMN processual (Figura 3) é possível notar que todo o processo é realizado de forma arcaica, manualmente. A catalogação dos livros é realizada externamente e, ao retornar, os dados do livro são adicionados a uma planilha.

No que tange o processo de empréstimos, os recibos para empréstimo e devolução são disponibilizados em papel, e a forma de comprovação de identidade é realizada via assinatura.

### 3.2.2 BPMN sistêmico proposto para a biblioteca da UAB - Polo Franca.

O BPMN sistêmico dispõe o processo de empréstimo e devolução dos livros, desenvolvido pela equipe, a partir dos requisitos levantados (Figura 4).

Figura 4 - BPMN Sistêmico



Fonte: Autores

No BPMN proposto (Figura 4), todo o processo é realizado digitalmente. Há a possibilidade de reserva do livro, tanto na instituição, quanto pelo próprio aluno, independentemente do local em que se encontra. A forma de comprovação de identidade, na retirada do livro, é realizada pela entrada de senha do usuário, cadastrada anteriormente.

### 3.3 Requisitos Funcionais

De acordo com Sommerville (2011), os requisitos funcionais são responsáveis por descrever as funções que o software deve executar. Essas declarações representam os serviços que o sistema deve oferecer, especificando as ações que o sistema deve realizar.

Além disso, os requisitos funcionais podem abordar a forma como o sistema deve responder a entradas específicas, como deve se comportar em situações específicas e podem definir o que o sistema não deve fazer.

Esses requisitos funcionais são essenciais para garantir que o *software* atenda às necessidades e expectativas dos usuários, fornecendo as funcionalidades esperadas de forma consistente e eficiente (Sommerville, 2011).

Os requisitos funcionais são: de usuário, de sistema, de domínio, evidentes e ocultos.

Requisitos de usuário são descrições das necessidades, expectativas e restrições dos usuários finais em relação ao sistema a ser desenvolvido. Eles são levantados por meio de interações, entrevistas, pesquisas e outras técnicas de coleta de requisitos com os usuários e partes interessadas envolvidas no projeto.

Os requisitos de usuário desempenham um papel fundamental na definição do escopo e funcionalidades do sistema, pois representam as perspectivas e objetivos dos usuários que irão interagir diretamente com o sistema. Eles ajudam a garantir que o sistema seja desenvolvido para atender às necessidades e expectativas dos usuários finais.

Esses requisitos podem abranger uma ampla variedade de aspectos, como funcionalidades específicas que os usuários desejam que o sistema ofereça, interfaces de usuário intuitivas e fáceis de usar, requisitos de desempenho, requisitos de segurança, requisitos de acessibilidade e requisitos de documentação e suporte (Sommerville, 2011).

Os requisitos de sistema são especificações detalhadas das funcionalidades, características e restrições do sistema a ser desenvolvido. Eles descrevem o que o sistema deve fazer, como deve se comportar e quais são seus limites e restrições.

Esses requisitos são derivados dos requisitos do produto e dos requisitos externos, além de considerar o contexto e o ambiente em que o sistema será implantado. Eles fornecem uma visão abrangente das funcionalidades e características necessárias para atender às necessidades dos usuários e das partes interessadas.

Além disso, os requisitos de sistema podem envolver restrições técnicas, como a plataforma de hardware e software a ser utilizada, interfaces com outros sistemas, protocolos de comunicação e requisitos de integração com sistemas existentes (Sommerville, 2011).

Requisitos de domínio são derivados do próprio domínio do sistema a ser desenvolvido e refletem suas características e restrições específicas. Eles podem influenciar os requisitos funcionais existentes ou estabelecer diretrizes sobre como as regras de negócio devem ser implementadas, refletindo os princípios e fundamentos do domínio do sistema.

Esses requisitos são moldados pela natureza e contexto do domínio em que o sistema será aplicado. Eles podem incluir restrições técnicas, regulamentações específicas, práticas comuns da indústria ou requisitos particulares do setor em que o sistema será utilizado.

Os requisitos de domínio ajudam a garantir que o sistema seja desenvolvido de acordo com as necessidades e exigências específicas do seu ambiente de aplicação. Eles são essenciais para a criação de uma solução que atenda adequadamente aos desafios e requisitos exclusivos do domínio em questão (Sommerville, 2011).

Os requisitos evidentes são requisitos que são explicitamente declarados e facilmente identificáveis. Eles representam funcionalidades, características ou restrições que podem ser facilmente observadas ou percebidas pelos usuários ou partes interessadas do sistema.

Esses requisitos são geralmente tangíveis e mensuráveis, podendo incluir aspectos como funcionalidades específicas que o sistema deve oferecer, desempenho esperado, interfaces de usuário intuitivas, requisitos de segurança, requisitos de confiabilidade, entre outros.

Os requisitos evidentes são aqueles que podem ser prontamente comunicados, compreendidos e validados pelos usuários e pela equipe de desenvolvimento. Eles são fundamentais para o sucesso do projeto, pois representam as expectativas claras e visíveis que os usuários têm em relação ao sistema (Sommerville, 2011).

Por fim, requisitos ocultos são aqueles que não são explicitamente declarados ou facilmente identificáveis pelos usuários ou partes interessadas do sistema. Eles representam funcionalidades, características ou restrições que podem não ser óbvias ou estar fora do escopo das expectativas iniciais dos usuários.

Esses requisitos podem surgir de maneira indireta, como consequência de interações complexas entre os diversos componentes do sistema, ou podem ser impulsionados por regulamentações, padrões de mercado ou restrições técnicas específicas.

Os requisitos ocultos muitas vezes estão relacionados a aspectos de desempenho, segurança, escalabilidade, manutenibilidade ou integração com outros sistemas. Podem envolver a necessidade de processar grandes volumes de dados de forma eficiente, garantir a proteção e privacidade das informações, suportar muitos usuários simultâneos ou permitir a integração com sistemas legados (Sommerville, 2011).

### 3.3.1 Requisitos funcionais para o sistema de gerenciamento de empréstimo e devolução de livros.

Nesta seção, são apresentadas as documentações dos requisitos funcionais para a solução sistêmica, contendo identificação e descrição detalhada, após elicitação de requisitos e elucidação das regras de negócio, bem como o desenho dos processos, por meio de BPMN (Quadro 2).

**Quadro 2** – Documentação de requisitos funcionais

<b>RF001- Acessar o software (funcionário).</b>	Categoria: ( ) Oculto (X) Evidente	Prioridade: (X) Altíssima ( ) Alta ( ) Média ( ) Baixa
<b>Descrição:</b> O sistema será disponibilizado em ambiente web.		

<b>RF002- Acessar o software (aluno).</b>	Categoria: <input type="checkbox"/> Oculto <input checked="" type="checkbox"/> Evidente	Prioridade: <input checked="" type="checkbox"/> Altíssima <input type="checkbox"/> Alta <input type="checkbox"/> Média <input type="checkbox"/> Baixa
<b>Descrição:</b> O sistema será disponibilizado em ambiente web.		
<b>RF003- Gerar catálogo dos livros disponíveis.</b>	Categoria: <input type="checkbox"/> Oculto <input checked="" type="checkbox"/> Evidente	Prioridade: <input checked="" type="checkbox"/> Altíssima <input type="checkbox"/> Alta <input type="checkbox"/> Média <input type="checkbox"/> Baixa
<b>Descrição:</b> O sistema deve gerar um catálogo dos livros com base no status definido como "disponível".		
<b>RF004- Buscar reserva no sistema.</b>	Categoria: <input type="checkbox"/> Oculto <input checked="" type="checkbox"/> Evidente	Prioridade: <input checked="" type="checkbox"/> Altíssima <input type="checkbox"/> Alta <input type="checkbox"/> Média <input type="checkbox"/> Baixa
<b>Descrição:</b> O sistema deve possibilitar que o funcionário verifique se há reserva(s).		
<b>RF005- Buscar empréstimo no sistema.</b>	Categoria: <input type="checkbox"/> Oculto <input checked="" type="checkbox"/> Evidente	Prioridade: <input checked="" type="checkbox"/> Altíssima <input type="checkbox"/> Alta <input type="checkbox"/> Média <input type="checkbox"/> Baixa
<b>Descrição:</b> O sistema deve possibilitar que o funcionário verifique se há empréstimos.		
<b>RF006- Buscar aluno no sistema.</b>	Categoria: <input type="checkbox"/> Oculto <input checked="" type="checkbox"/> Evidente	Prioridade: <input checked="" type="checkbox"/> Altíssima <input type="checkbox"/> Alta <input type="checkbox"/> Média <input type="checkbox"/> Baixa
<b>Descrição:</b> O sistema deve possibilitar que o funcionário busque o aluno no sistema.		
<b>RF007- Verificar se aluno está cadastrado.</b>	Categoria: <input checked="" type="checkbox"/> Oculto <input type="checkbox"/> Evidente	Prioridade: <input checked="" type="checkbox"/> Altíssima <input type="checkbox"/> Alta <input type="checkbox"/> Média <input type="checkbox"/> Baixa
<b>Descrição:</b> O sistema deve verificar se o aluno está cadastrado.		

<b>RF008- Gerar código de reserva.</b>	Categoria: <input checked="" type="checkbox"/> Oculto <input type="checkbox"/> Evidente	Prioridade: <input checked="" type="checkbox"/> Altíssima <input type="checkbox"/> Alta <input type="checkbox"/> Média <input type="checkbox"/> Baixa
<b>Descrição:</b> O sistema deve gerar um código para a reserva solicitada pelo aluno.		
<b>RF009- Gerar código de empréstimo.</b>	Categoria: <input checked="" type="checkbox"/> Oculto <input type="checkbox"/> Evidente	Prioridade: <input checked="" type="checkbox"/> Altíssima <input type="checkbox"/> Alta <input type="checkbox"/> Média <input type="checkbox"/> Baixa
<b>Descrição:</b> O sistema deve gerar um código para o empréstimo efetuado.		
<b>RF010- Confirmar reserva.</b>	Categoria: <input type="checkbox"/> Oculto <input checked="" type="checkbox"/> Evidente	Prioridade: <input checked="" type="checkbox"/> Altíssima <input type="checkbox"/> Alta <input type="checkbox"/> Média <input type="checkbox"/> Baixa
<b>Descrição:</b> O sistema deve permitir que o usuário confirme a reserva de livros.		
<b>RF011- Confirmar empréstimo.</b>	Categoria: <input type="checkbox"/> Oculto <input checked="" type="checkbox"/> Evidente	Prioridade: <input checked="" type="checkbox"/> Altíssima <input type="checkbox"/> Alta <input type="checkbox"/> Média <input type="checkbox"/> Baixa
<b>Descrição:</b> O sistema deve permitir que o funcionário da biblioteca confirme o empréstimo do livro para o aluno.		
<b>RF012- Selecionar livro que deseja cancelar.</b>	Categoria: <input type="checkbox"/> Oculto <input checked="" type="checkbox"/> Evidente	Prioridade: <input checked="" type="checkbox"/> Altíssima <input type="checkbox"/> Alta <input type="checkbox"/> Média <input type="checkbox"/> Baixa
<b>Descrição:</b> O sistema deve possibilitar a seleção do livro que o usuário deseja cancelar.		
<b>RF013- Confirmar cancelamento.</b>	Categoria: <input type="checkbox"/> Oculto <input checked="" type="checkbox"/> Evidente	Prioridade: <input checked="" type="checkbox"/> Altíssima <input type="checkbox"/> Alta <input type="checkbox"/> Média <input type="checkbox"/> Baixa
<b>Descrição:</b> O sistema deve permitir que o usuário seja capaz de confirmar o cancelamento de livros.		
<b>RF014- Confirmar devolução.</b>	Categoria:	Prioridade:

	( ) Oculto (X) Evidente	(X) Altíssima ( ) Alta ( ) Média ( ) Baixa
<b>Descrição:</b> O sistema deve permitir que o funcionário confirme a devolução do(s) livro(s) emprestado(s) pelo aluno.		
<b>RF015- Cancelar reserva do(s) livro(s).</b>	Categoria: ( ) Oculto (X) Evidente	Prioridade: (X) Altíssima ( ) Alta ( ) Média ( ) Baixa
<b>Descrição:</b> O sistema deve possibilitar o cancelamento da reserva efetuada.		
<b>RF016- Selecionar livro(s).</b>	Categoria: ( ) Oculto (X) Evidente	Prioridade: (X) Altíssima ( ) Alta ( ) Média ( ) Baixa
<b>Descrição:</b> O sistema deve permitir que o usuário seja capaz de selecionar os livros disponível para empréstimo.		
<b>RF017- Digitar senha.</b>	Categoria: ( ) Oculto (X) Evidente	Prioridade: (X) Altíssima ( ) Alta ( ) Média ( ) Baixa
<b>Descrição:</b> O sistema deve receber a senha do aluno.		
<b>RF018- Verificar senha.</b>	Categoria: (X) Oculto ( ) Evidente	Prioridade: (X) Altíssima ( ) Alta ( ) Média ( ) Baixa
<b>Descrição:</b> O sistema deve ser capaz de verificar se a senha que o aluno digitou é válida.		
<b>RF019- Retornar livro(s) para catálogo de disponíveis.</b>	Categoria: (X) Oculto ( ) Evidente	Prioridade: (X) Altíssima ( ) Alta ( ) Média ( ) Baixa
<b>Descrição:</b> O sistema deve retornar para o catálogo os livros cancelados.		
<b>RF020- Verificar quantidade de</b>	Categoria:	Prioridade:

<b>cancelamentos e/ou devoluções.</b>	(X) Oculto ( ) Evidente	(X) Altíssima ( ) Alta ( ) Média ( ) Baixa
<b>Descrição:</b> O sistema deve ser capaz de verificar a quantidade de cancelamentos e/ou devoluções de um aluno.		
<b>RF021- Realizar a devolução.</b>	Categoria: (X) Oculto ( ) Evidente	Prioridade: (X) Altíssima ( ) Alta ( ) Média ( ) Baixa
<b>Descrição:</b> O sistema deve permitir que o aluno realize uma devolução.		
<b>RF022- Realizar empréstimo.</b>	Categoria: ( ) Oculto (X) Evidente	Prioridade: (X) Altíssima ( ) Alta ( ) Média ( ) Baixa
<b>Descrição:</b> O sistema deve possibilitar que o aluno realize um empréstimo.		
<b>RF023- Retirar livro(s).</b>	Categoria: ( ) Oculto (X) Evidente	Prioridade: (X) Altíssima ( ) Alta ( ) Média ( ) Baixa
<b>Descrição:</b> O sistema deve possibilitar que o aluno retire um livro.		
<b>RF024- Definir status do livro como emprestado.</b>	Categoria: (X) Oculto ( ) Evidente	Prioridade: (X) Altíssima ( ) Alta ( ) Média ( ) Baixa
<b>Descrição:</b> O sistema deve ser capaz de definir o status do livro como “emprestado” após o registro de um empréstimo.		

Fonte: Autores

### 3.4 Requisitos Não Funcionais

Os requisitos não funcionais são aqueles que impõem restrições a uma solução. De acordo com Sommerville (2011), eles descrevem limitações ou restrições relacionadas aos serviços ou funções fornecidos pelo sistema.

Essas restrições podem surgir a partir das necessidades dos usuários, como restrições de orçamento, políticas organizacionais, requisitos de interoperabilidade com outros sistemas de software ou hardware, bem como fatores externos, como regulamentações e legislações.

Esses requisitos não funcionais são igualmente importantes para o desenvolvimento de um sistema, pois influenciam aspectos críticos, como desempenho, segurança, confiabilidade, usabilidade e outros atributos que afetam a qualidade e o sucesso do sistema em atender às expectativas dos usuários e aos requisitos do ambiente em que será utilizado (Sommerville, 2011).

Os requisitos não funcionais são: de produto, organizacionais e externos.

Requisitos de produto são responsáveis por especificar o comportamento desejado do software. Eles se referem aos atributos de qualidade que o software deve apresentar, como facilidade de uso, eficiência em termos de desempenho e espaço ocupado, confiabilidade e portabilidade.

Esses requisitos estabelecem as características essenciais que o produto de software deve possuir para atender às necessidades dos usuários e garantir uma experiência satisfatória. Eles definem os critérios de qualidade que serão utilizados para avaliar o desempenho do software e determinam se ele está em conformidade com as expectativas estabelecidas.

Portanto, os requisitos de produto são fundamentais para garantir que o software seja desenvolvido de forma a atender aos padrões de qualidade estabelecidos, proporcionando uma experiência de uso eficiente, confiável e compatível com os requisitos de portabilidade.

Requisitos organizacionais são derivados das metas, políticas e procedimentos estabelecidos pelas organizações. Eles englobam uma variedade de requisitos relacionados à entrega do produto, implementação do sistema e conformidade com padrões específicos.

Em suma, os requisitos organizacionais são fundamentais para orientar o desenvolvimento do software, garantindo que ele esteja alinhado com as metas e diretrizes da organização, atendendo aos prazos estabelecidos e seguindo os padrões e procedimentos adequados.

Os requisitos externos são aqueles que se originam de fatores externos ao sistema e ao processo de desenvolvimento. Eles abrangem uma variedade de

requisitos relacionados à interoperabilidade com sistemas de outras organizações, requisitos éticos e requisitos legais (Fernandes, 2017)

3.4.1 Requisitos não funcionais para o sistema de gerenciamento de empréstimo e devolução de livros.

Nesta seção, são apresentadas as documentações dos requisitos não funcionais (Quadro 3), contendo identificação e descrição detalhada.

**Quadro 3** – Documentação de requisitos não funcionais

<b>RNF001- Desbloquear conta do aluno de acordo com a quantidade de livros cancelados/ devolvidos.</b>	O sistema deve desbloquear a conta do aluno de acordo com a quantidade de livros devolvidos.	Oculto.	( ) Desejável (X) Obrigatório	(X) Permanente ( ) Transitório
<b>RNF002- Suspender conta do usuário por 10 dias.</b>	O sistema deve suspender a conta do aluno	Oculto.	( ) Desejável (X) Obrigatório	(X) Permanente ( ) Transitório
<b>RNF003- Enviar mensagem/ e-mail referente à ação.</b>	O sistema deve enviar uma mensagem ou e-mail ao usuário informando sobre a ação realizada pelo sistema.	Evidente.	( ) Desejável (X) Obrigatório	(X) Permanente ( ) Transitório
<b>RNF004- Suspender disponibilidade do(s) livro(s) escolhido(s) por 1 dia.</b>	O sistema deverá, temporariamente, alterar o status do livro como “reservado”.	Oculto.	( ) Desejável (X) Obrigatório	(X) Permanente ( ) Transitório
<b>RNF005- Solicitar cadastro.</b>	O sistema deve exibir quando o usuário não estiver cadastrado no banco de dados.	Evidente.	( ) Desejável (X) Obrigatório	(X) Permanente ( ) Transitório

Fonte: Autores

### 3.5 Regras de Negócio

As regras de negócio são diretrizes ou restrições que definem como uma organização ou sistema deve operar. Elas representam as políticas, procedimentos, práticas e restrições que regem as atividades e processos de uma empresa, elas são

importantes porque fornecem uma estrutura e um conjunto de diretrizes para orientar as atividades de uma organização. Elas ajudam a garantir a consistência, a conformidade e a eficiência nas operações de negócio, ao mesmo tempo em que facilitam a automação e a gestão dos processos.

### 3.5.1 Regras de Negócio da UAB Polo Franca

Nesta seção, são apresentadas e descritas as regras de negócio necessárias para o desenvolvimento da solução sistêmica (Quadro 4).

**Quadro 4** – Documentação das regras de negócio

<b>RN001 - Verificar se aluno está cadastrado.</b>
<b>Descrição:</b> Apenas usuários cadastrados podem utilizar o sistema.
<b>RN002 - Gerar código de reserva.</b>
<b>Descrição:</b> Cada reserva deve possuir um código único e intransferível. O código de reserva será funcional por um dia útil.
<b>RN003 - Gerar código de empréstimo.</b>
<b>Descrição:</b> Cada empréstimo deverá possuir um código único e intransferível.
<b>RN004 - Selecionar livro(s) que deseja cancelar.</b>
<b>Descrição:</b> Existe a possibilidade de cancelamento parcial ou integral da reserva.
<b>RN005 - Cancelar reserva do(s) livro(s).</b>
<b>Descrição:</b> O usuário deve ser notificado sobre o cancelamento da reserva. O livro deve retornar para o catálogo de livros disponíveis.
<b>RN006 - Digitar senha.</b>
<b>Descrição:</b> O aluno deve digitar a senha cadastrada para retirada do(s) livro(s) reservado(s) e a senha deverá ser validada.

<b>RN007 - Verificar senha.</b>
<b>Descrição:</b> Apenas usuários com senhas válidas podem locar livros.
<b>RN008 - Desbloquear a conta do aluno de acordo com a quantidade de livros cancelados/ devolvidos.</b>
<b>Descrição:</b> O limite estabelecido são, no máximo, dois empréstimos por usuário.
<b>RN009 - Enviar mensagem/ e-mail referente à ação.</b>
<b>Descrição:</b> As ações tomadas pelo sistema, que envolvam o usuário, devem ser notificadas a ele.
<b>RN010 - Suspender disponibilidade do(s) livro(s) escolhido(s) por 1 dia.</b>
<b>Descrição:</b> O usuário possui o prazo de um dia útil para retirada do(s) livro(s) reservado(s).
<b>RN011 - Verificar quantidade de cancelamentos.</b>
<b>Descrição:</b> O aluno pode efetuar apenas um cancelamento a cada 15 dias, sem penalidades.
<b>RN012 - Verificar quantidade de devoluções.</b>
<b>Descrição:</b> O aluno pode realizar a devolução do(s) livro(s) no prazo de 15 dias desde a data do empréstimo, sem penalidades.

Fonte: Autores

### 3.6 Casos de Uso

Os diagramas de caso de uso são usados para capturar os requisitos do sistema, ajudando a entender como os atores interagem com o sistema e descrevendo os principais cenários de uso, eles fornecem uma visão clara das funcionalidades do sistema.

Além disso, esses diagramas servem como base para outras atividades de modelagem e projeto, como a definição de casos de teste e a identificação de classes e interfaces do sistema.

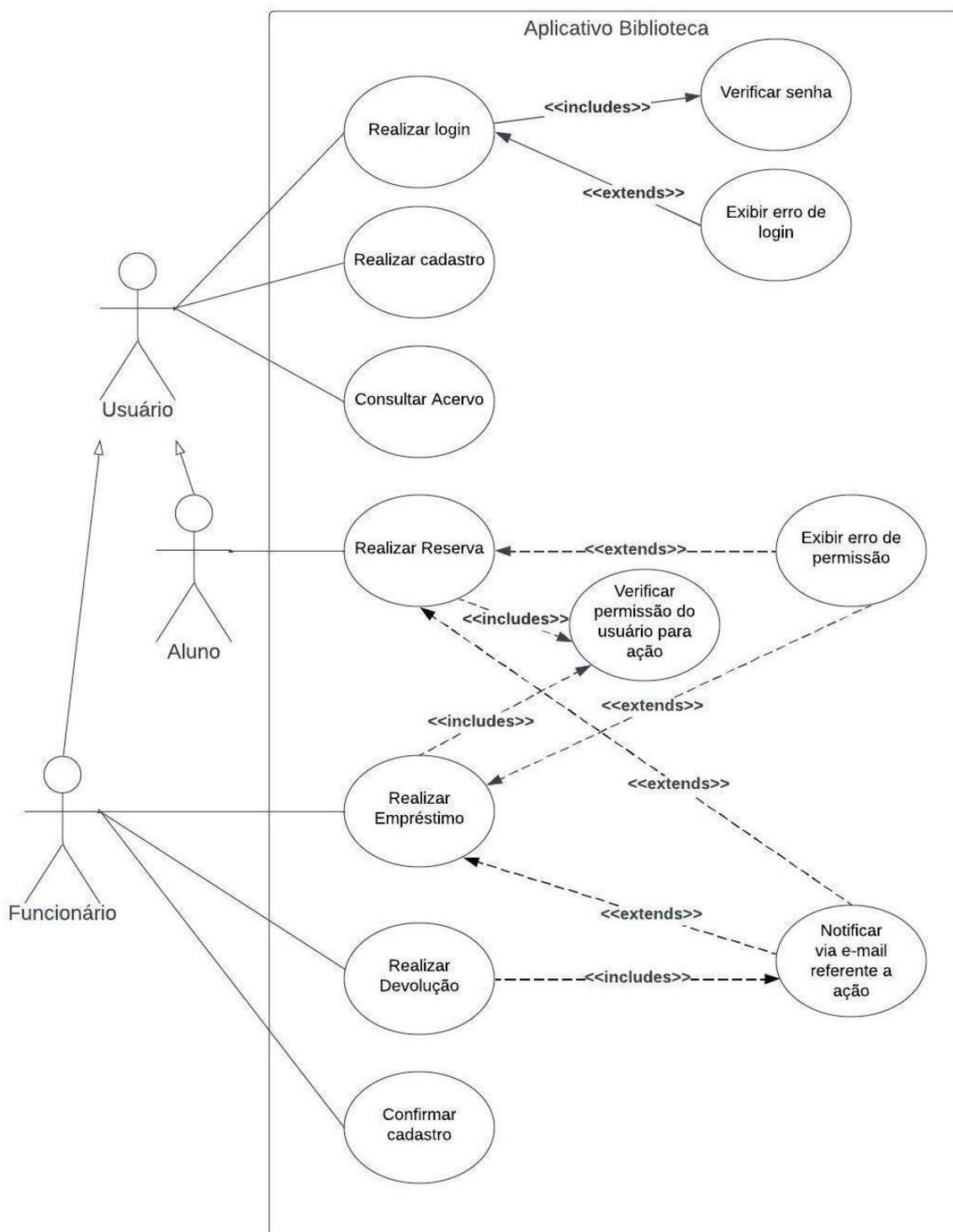
Os elementos principais presentes nos diagramas de caso de uso incluem:

- a) Ator: Representa um usuário externo, sistema externo ou outro componente que interage com o sistema em análise.
- b) Caso de Uso: Descreve uma funcionalidade ou recurso do sistema, mostrando uma sequência de ações e interações entre atores e o sistema.
- c) Relacionamentos: Descrevem as associações e dependências entre atores e casos de uso, indicando como eles interagem.
- d) Inclusão: Indica que um caso de uso (chamado caso de uso base) é incluído em outro caso de uso (chamado caso de uso inclusor) em algum ponto da sua execução.
- e) Extensão: Indica que um caso de uso (chamado caso de uso estendido) pode estender outro caso de uso (chamado caso de uso extensor) adicionando ações opcionais ou alternativas a ele.

### 3.6.1 Diagrama de Caso de Uso

O diagrama de caso de uso foi elaborado baseando-se no BPMN sistêmico proposto no desenvolvimento da solução sistêmica. Nele constam os atores aluno, funcionário e o sistema, bem como suas ações e as interações entre os diferentes atores (Figura 5).

Figura 5 - Diagrama de Caso de Uso



Fonte: Autores

A figura 5 representa o caso de uso do sistema. Nele temos o fluxo do sistema representado acerca dos processos de solicitação de reserva, empréstimo, retirada e devolução de livros, além do cancelamento da reserva.

### 3.6.2 Documentação de Caso de Uso

Nesta seção, são apresentadas as documentações dos casos de uso (Quadro 5), contendo identificação e descrição detalhada do cenário em que a ação acontece.

**Quadro 5** – Documentação dos casos de uso

<b>Caso de Uso - Consultar Acervo</b>	
<b>ID:</b>	UC 001
<b>Descrição:</b>	Este caso de uso tem por objetivo permitir que o usuário consulte o acervo da biblioteca.
<b>Primário:</b>	<b>Ator</b> Usuário.
<b>Pré-condição:</b>	Não existe.
<b>Cenário Principal:</b>	1- O aluno seleciona a opção de consultar o acervo. 2- O sistema exibe o acervo da biblioteca ao aluno.
<b>Pós-condição:</b>	O aluno visualiza o acervo da biblioteca.
<b>Cenário Alternativo:</b>	Não existe cenário alternativo.
<b>Caso de Uso - Realizar cadastro</b>	
<b>ID:</b>	UC 002
<b>Descrição:</b>	Este caso de uso tem por objetivo permitir que o usuário realize cadastro no sistema.
<b>Primário:</b>	<b>Ator</b> Usuário.
<b>Pré-condição:</b>	Não existe.
<b>Cenário Principal:</b>	1- O usuário seleciona a opção de realizar cadastro. 2- O usuário preenche o cadastro.
<b>Pós-condição:</b>	O usuário poderá fazer login.
<b>Cenário Alternativo:</b>	Não existe cenário alternativo.
<b>Caso de Uso - Confirmar cadastro</b>	
<b>ID:</b>	UC 003

<b>Descrição:</b>	<b>Descr</b>	Este caso de uso tem por objetivo permitir que o funcionário confirme o cadastro do usuário.
<b>Primário:</b>	<b>Ator</b>	Funcionário.
<b>Condição:</b>	<b>Pré-</b>	O usuário solicitar o cadastro.
<b>Cenário Principal:</b>	<b>Cená</b>	O cadastro é confirmado.
<b>Condição:</b>	<b>Pós-</b>	O usuário poderá fazer login.
<b>Cenário Alternativo:</b>	<b>Cená</b>	O cadastro não é confirmado.
<b>Caso de Uso - Realizar login</b>		
	<b>ID:</b>	UC 004
<b>Descrição:</b>	<b>Descr</b>	Este caso de uso tem por objetivo permitir que o usuário realize login no sistema.
<b>Primário:</b>	<b>Ator</b>	Usuário.
<b>Condição:</b>	<b>Pré-</b>	O usuário estar cadastrado.
<b>Cenário Principal:</b>	<b>Cená</b>	1- O usuário seleciona a opção de realizar login. 2- O sistema verifica a senha do usuário. 3- Se o usuário estiver com a senha correta, o sistema prossegue com o login.
<b>Condição:</b>	<b>Pós-</b>	O usuário estará logado.
<b>Cenário Alternativo:</b>	<b>Cená</b>	Exibe erro de login.
<b>Caso de Uso - Verificar senha</b>		
	<b>ID:</b>	UC 005
<b>Descrição:</b>	<b>Descr</b>	Este caso de uso tem por objetivo verificar a senha de login do usuário.
<b>Primário:</b>	<b>Ator</b>	Sistema.
<b>Condição:</b>	<b>Pré-</b>	O usuário fazer login.
<b>Cenário Principal:</b>	<b>Cená</b>	1- O sistema verifica a senha do usuário. 2- Se o usuário estiver com a senha correta, o sistema prossegue com o login.
<b>Condição:</b>	<b>Pós-</b>	O usuário estará logado.
<b>Cenário Alternativo:</b>		Exibe erro de login.
<b>Caso de Uso - Exibir erro de login</b>		
	<b>ID:</b>	UC 006

<b>Descrição:</b>	<b>Descr</b>	Este caso de uso tem por objetivo exibir erro de login.
<b>Primário:</b>	<b>Ator</b>	Sistema.
<b>condição:</b>	<b>Pré-</b>	O usuário fazer login.
<b>rio Principal:</b>	<b>Cená</b>	1- O sistema exibe erro de login.
<b>condição:</b>	<b>Pós-</b>	O usuário não estará logado.
<b>rio Alternativo:</b>	<b>Cená</b>	Não existe.
<b>Caso de Uso - Realizar Reserva</b>		
	<b>ID:</b>	UC 007
<b>Descrição:</b>	<b>Descr</b>	Este caso de uso tem por objetivo permitir que o aluno realize uma reserva de itens do acervo da biblioteca.
<b>Primário:</b>	<b>Ator</b>	Aluno.
<b>condição:</b>	<b>Pré-</b>	O aluno estar logado no sistema.
<b>rio Principal:</b>	<b>Cená</b>	1- O aluno seleciona a opção de realizar reserva. 2- O sistema verifica a permissão do usuário para a ação. 3- Se o usuário tiver permissão, o sistema prossegue com a reserva. 4 -O sistema notifica via e-mail referente à ação.
<b>condição:</b>	<b>Pós-</b>	O aluno realiza uma reserva de itens do acervo da biblioteca.
<b>rio Alternativo:</b>	<b>Cená</b>	O sistema exibe uma mensagem de erro.
<b>Caso de Uso - Realizar empréstimo</b>		
	<b>ID:</b>	UC 008
<b>Descrição:</b>	<b>Descr</b>	Este caso de uso tem por objetivo permitir que o aluno realize um empréstimo atendendo o RF022.
<b>Primário:</b>	<b>Ator</b>	Usuário.
<b>condição:</b>	<b>Pré-</b>	O aluno estar logado no sistema.
<b>rio Principal:</b>	<b>Cená</b>	1- O usuário seleciona a opção de realizar empréstimo. 2- O sistema verifica a permissão do usuário para a ação. 3- Se o usuário tiver permissão, o sistema prossegue com a reserva. 4 -O sistema notifica via e-mail referente à ação.
<b>condição:</b>	<b>Pós-</b>	O empréstimo está em andamento.
<b>rio Alternativo:</b>	<b>Cená</b>	O sistema exibe uma mensagem de erro.
<b>Caso de Uso - Realizar devolução.</b>		

<b>ID:</b>	UC 009
<b>Descrição:</b>	Este caso de uso tem por objetivo permitir que o aluno realize uma devolução atendendo o RF021.
<b>Primário:</b>	<b>Ator</b> Funcionário.
<b>Pré-condição:</b>	Existir um empréstimo ativo no nome do aluno.
<b>Cenário Principal:</b>	1- O funcionário seleciona a opção de realizar devolução. 2- O sistema verifica a permissão do usuário para a ação. 3- Se o usuário tiver permissão, o sistema prossegue com a devolução. 4 -O sistema notifica via e-mail referente à ação.
<b>Pós-condição:</b>	A devolução está em andamento.
<b>Cenário Alternativo:</b>	O sistema exibe uma mensagem de erro.
<b>Caso de Uso - Verificar Permissão do Usuário para Ação</b>	
<b>ID:</b>	UC 010
<b>Descrição:</b>	Este caso de uso tem por objetivo verificar se o usuário tem permissão para realizar determinada ação no sistema.
<b>Primário:</b>	<b>Ator</b> Sistema
<b>Pré-condição:</b>	O sistema recebe uma solicitação de ação por parte do usuário.
<b>Cenário Principal:</b>	1- O sistema verifica as permissões do usuário para a ação solicitada. 2- Se o usuário tem permissão, a ação é realizada.
<b>Pós-condição:</b>	O sistema realiza ou não a ação solicitada com base nas permissões do usuário.
<b>Cenário Alternativo:</b>	Se o usuário não tem permissão, o sistema exibe uma mensagem de erro informando a falta de permissão.
<b>Caso de Uso - Exibir Erro de Permissão</b>	
<b>ID:</b>	UC 011
<b>Descrição:</b>	Este caso de uso tem por objetivo exibir uma mensagem de erro quando o usuário tenta realizar uma ação para a qual não tem permissão.
<b>Primário:</b>	<b>Ator</b> Sistema
<b>Pré-condição:</b>	O sistema identifica que o usuário não tem permissão para a ação solicitada.
<b>Cenário Principal:</b>	1- O sistema exibe uma mensagem de erro informando a falta de permissão.
<b>Pós-condição:</b>	2- O usuário é informado sobre a falta de permissão para a ação solicitada.

<b>Cenário Alternativo:</b>	<b>Cená</b>	Não existe cenário alternativo.
<b>Caso de Uso - Notificar via E-mail Referente à ação</b>		
<b>ID:</b>		UC 012
<b>Descrição:</b>	<b>Descr</b>	Este caso de uso tem por objetivo enviar uma notificação via e-mail referente a uma ação realizada no sistema.
<b>Primário:</b>	<b>Ator</b>	Sistema
<b>condição:</b>	<b>Pré-</b>	O sistema realiza uma ação no sistema que requer notificação por e-mail.
<b>Cenário Principal:</b>	<b>Cená</b>	1- O sistema envia uma notificação por e-mail para o usuário informando sobre a ação realizada.
<b>condição:</b>	<b>Pós-</b>	O usuário recebe uma notificação por e-mail sobre a ação realizada.
<b>Cenário Alternativo:</b>	<b>Cená</b>	Não existe cenário alternativo.

Fonte: Autores

### 3.7 Matriz de Rastreabilidade

A matriz de rastreabilidade é uma ferramenta usada no contexto de gerenciamento de requisitos e engenharia de software para rastrear e documentar a relação entre diferentes elementos do sistema ao longo de seu ciclo de vida.

Essa matriz permite estabelecer a rastreabilidade dos requisitos, ou seja, acompanhar a origem, o relacionamento e o status dos requisitos em todo o processo de desenvolvimento de software. Ela mostra as interdependências entre os requisitos, casos de uso, testes, componentes do sistema e outros artefatos relacionados.

A matriz de rastreabilidade também é útil para a análise de impacto, permitindo entender como as mudanças em um requisito ou componente afetam outros elementos do sistema. Ela facilita a identificação de dependências e ajuda a planejar a execução de testes e a validação dos requisitos.

#### 3.7.1 Matriz de Rastreabilidade - Requisitos x Regras de negócio

O Quadro 6 representa a matriz de rastreabilidade considerando os requisitos (funcionais e não funcionais) e as regras de negócio envolvidas para tal requisito.

**Quadro 6** - Matriz de Rastreabilidade - Requisitos x Regras de negócio

Requisitos x Regra de negócio												
	RF 007	RF 008	RF 009	RF 012	RF 015	RF 017	RF 018	RF 020	RNF 001	RNF 002	RNF 003	RNF 004
RN 001	X											
RN 002		X										
RN 003			X									
RN 004				X								
RN 005					X							
RN 006						X						
RN 007							X					
RN 008									X			
RN 009											X	
RN 010												X
RN 011								X		X		
RN 012								X		X		

Fonte: Autores

### 3.7.2 Matriz de Rastreabilidade - Requisitos x Regras sistêmicas

O Quadro 7 representa a matriz de rastreabilidade considerando os requisitos (funcionais e não funcionais) e as regras sistêmicas envolvidas para tal requisito.

**Quadro 7** - Matriz de Rastreabilidade - Requisitos x Regras sistêmicas

Requisitos x Regras sistêmicas									
	RS 001	RS 002	RS 003	RS 004	RS 005	RS 006	RS 007	RS 008	RS 009
RF 003	X								
RF 004		X							
RF 005			X						
RF 006				X					
RF 010					X				
RF 011						X			
RF 016							X		
RF 019								X	
RF 024									X

Fonte: Autores

## 4 Ferramentas e Métodos

As ferramentas, bem como os métodos utilizados para a elaboração desta solução sistêmica estão descritos a seguir.

### 4.1 FERRAMENTAS

As ferramentas escolhidas para o projeto foram selecionadas com base em sua eficiência, escalabilidade, suporte à comunidade e pela preferência e experiência pessoal da equipe. Essas ferramentas têm documentação abrangente, tutoriais e recursos disponíveis na comunidade de desenvolvedores, o que torna mais fácil para os desenvolvedores aprenderem e implementarem as soluções, além de serem

ferramentas de código aberto, ou seja, são gratuitas e podem ser usadas para fins comerciais e pessoais.

#### 4.1.1 SWOT - Canva

O Canva é uma ferramenta de design gráfico que oferece uma série de vantagens para a criação de análises SWOT. Com uma interface intuitiva, proporciona a criação rápida e fácil de gráficos e diagramas, mesmo para aqueles sem habilidades avançadas de design, permitindo que você adapte o design do seu gráfico com cores, fontes, imagens e muito mais. O Canva também facilita a colaboração em equipe, armazenamento em nuvem, exportação e compartilhamento de projetos. (Canva, 2024)

#### 4.1.2 5W2H - Google Planilhas

O Google Planilhas é uma ferramenta versátil e acessível para criar um plano 5W2H. A colaboração em tempo real e a integração com outros serviços do Google permitem que equipes trabalhem de forma eficiente e compartilhem informações instantaneamente. Além disso, o armazenamento na nuvem garante a segurança dos seus dados e o acesso a eles em qualquer lugar. (Google planilhas, 2024)

#### 4.1.3 Diagramas - Draw.io

O Draw.io é uma ferramenta de desenho e diagramação online gratuita, que permite criar e compartilhar diagramas de forma fácil e rápida, é um mecanismo baseado na web, com uma interface que facilita a criação. Com colaboração em tempo real e vários formatos para exportação, possui integração com outras ferramentas, como Google Drive, Dropbox e OneDrive. (Draw.io, 2024)

#### 4.1.4 Documentação - Documentos Google

O Documento Google é uma ferramenta de processamento de texto baseada na web, ou seja, permite o acesso da documentação de qualquer lugar e dispositivo com conexão à internet, possibilitando edições simultâneas, e armazenamento automático na nuvem. Oferece mecanismo de revisão, como comentários e sugestões, para facilitar o feedback e a colaboração, também mantém um histórico de todas as alterações feitas no documento. (Google documentos, 2024)

#### 4.1.5 Desenvolvimento

##### 4.1.5.1 Visual Studio Code

O VS Code (Visual Studio Code) é uma ferramenta de desenvolvimento altamente popular e amplamente utilizada, gratuita, de código aberto e multiplataforma, ou seja, é compatível com Windows, macOS e Linux, com suporte para múltiplas linguagens de programação. Possui uma alta extensibilidade, isto é, possibilitando adicionar funcionalidades adicionais e personalizar sua experiência de desenvolvimento. O VS Code tem integração com o Git, o que permite o gerenciamento de versões e colaboração em projetos, inclui um terminal integrado que permite a execução de comandos diretamente do editor, facilitando as tarefas de desenvolvimento. (Visual Studio Code, 2024)

##### 4.1.5.2 JavaScript

O JavaScript é uma das linguagens de programação mais populares e amplamente utilizadas no mundo do desenvolvimento web e mobile, relativamente fácil de aprender, permite criar funcionalidades interativas e dinâmicas de forma rápida, conta com uma vasta seleção de bibliotecas e frameworks, pode ser facilmente integrado com outras tecnologias, como APIs. Com o uso de Node.js, pode ser utilizados tanto para desenvolvimento de front-end quanto de back-end. A evolução do JavaScript, conhecida como TypeScript, permite definir tipos de dados para variáveis, argumentos de função e valores de retorno, ajudando a detectar erros de compilação e reduzi-los, é altamente integrado com ferramentas de desenvolvimento modernas, com tipagem estática que tornam o código mais legível e fácil de entender para outros membros da equipe. (JavaScript, 2024; TypeScript, 2024)

##### 4.1.5.4 React

O React é uma biblioteca JavaScript utilizada para a construção de interfaces de usuário (UI) em aplicações web. Por ser componentizado (dividir a UI em pequenas partes reutilizáveis) facilita a manutenção e o desenvolvimento de aplicações complexas, a utilização do Virtual DOM, permite que o React atualize apenas os componentes que foram alterados, tornando a aplicação mais rápida e eficiente. (React, 2024)

#### 4.1.5.5 Tailwind CSS

O Tailwind CSS é uma estrutura de design de código aberto que facilita a criação de interfaces de usuário (UI) elegantes e responsivas, altamente flexível e permite a criação de estilos únicos, utiliza uma abordagem chamada Atomic CSS, que divide as classes CSS em pequenos utilitários reutilizáveis tornando a escrita de CSS mais eficiente e facilita a manutenção. (Tailwind CSS, 2024)

#### 4.1.5.6 DaisyUI

A DaisyUI é uma biblioteca de componentes para Tailwind CSS, que oferece uma variedade de componentes, como botões, cartões, formulários, é altamente customizável e permite personalizar os estilos dos componentes, ou até mesmo criar seus próprios, de acordo com as necessidades do projeto. Otimizado para desempenho e com uma abordagem de carregamento de CSS apenas necessário. (DaisyUI, 2024)

#### 4.1.5.7 Next.js

O Next.js é um framework de React que suporta a renderização do lado do servidor (SSR), o que significa que as páginas são renderizadas no servidor antes de serem enviadas para o navegador o que melhora a performance e a experiência do usuário, especialmente em aplicações com muitos dados dinâmicos. Além do SSR, Next.js suporta a pré-renderização estática, o que significa que as páginas podem ser geradas no momento da construção e servidas como arquivos estáticos para o navegador. Tem um sistema de roteamento dinâmico que permite criar rotas dinâmicas com facilidade, suporta hot reloading, o que significa que as alterações no código são refletidas automaticamente no navegador, sem a necessidade de recarregar a página. (Next.js, 2024)

#### 4.1.5.8 PostgreSQL

O PostgreSQL oferece um desempenho robusto e escalabilidade, permitindo o manuseio eficiente de grandes volumes de dados e transações complexas. A conformidade com padrões SQL garante portabilidade e compatibilidade de consultas e esquemas. A extensibilidade do PostgreSQL permite a adição de novos tipos de dados, operadores e linguagens de programação, enquanto suas avançadas capacidades de suporte a dados complexos e conformidade com ACID e transações

garantem confiabilidade e integridade dos dados. Além disso, o PostgreSQL possui uma forte comunidade de código aberto, documentação abrangente e suporte ativo, tornando-o uma solução confiável e econômica para o desenvolvimento de projetos. (PostgreSQL, 2024)

## 4.2 MÉTODOS

O processo de desenvolvimento do projeto abrangeu a criação de diversos artefatos essenciais para garantir o sucesso e a eficiência do resultado.

- a) SWOT (*Strengths, Weaknesses, Opportunities, Threats*): A análise SWOT foi realizada para identificar e avaliar os pontos fortes, pontos fracos, oportunidades e ameaças relacionadas ao projeto. Esse artefato permitiu uma compreensão mais profunda do ambiente do projeto, auxiliando na definição de estratégias e tomada de decisões.
- b) 5W2H: Utilizando o 5W2H, foi possível estabelecer um plano de ação detalhado, atribuir responsabilidades claras, definir prazos e recursos necessários, e acompanhar o progresso do projeto de forma mais eficiente.
- c) TAP (Termo de Abertura do Projeto): O TAP é um documento que descreve o objetivo, escopo, entregas, prazos, recursos e demais informações relevantes do projeto. Ele serviu como uma referência central para alinhar as expectativas e definir os parâmetros do projeto.
- d) BPMN (*Business Process Model and Notation*): O BPMN foi utilizado para modelar e documentar os processos de negócio relacionados ao projeto. Esses diagramas forneceram uma representação visual clara das atividades, decisões, fluxos de trabalho e interações, facilitando a compreensão e a comunicação entre os membros da equipe.
- e) Requisitos e Casos de Uso: Os requisitos foram identificados e documentados para definir as funcionalidades e as características da solução. Os casos de uso foram desenvolvidos para descrever os diferentes cenários e interações entre usuários e o sistema. Esses artefatos foram fundamentais para alinhar as necessidades dos usuários e direcionar o desenvolvimento da solução. (Sommerville, 2011).
- f) Desenvolvimento: O software foi construído para fornecer funcionalidades específicas aos usuários. Cada etapa do processo de desenvolvimento foi

planejada e executada para garantir que o software seja eficiente, confiável e seguro. O código foi escrito de forma a permitir futuras expansões e modificações, garantindo uma base sólida e escalável para o sistema. Além disso, foram realizados testes para garantir que o software funcione conforme o esperado e atenda aos requisitos e expectativas dos usuários.

- g) **Matriz de Rastreabilidade:** A matriz de rastreabilidade foi elaborada para relacionar os requisitos com os artefatos desenvolvidos, garantindo a rastreabilidade e a cobertura dos requisitos ao longo do projeto.
- h) **Portabilidade e Métricas:** A portabilidade do sistema foi considerada durante o desenvolvimento, visando garantir que a solução possa ser executada em diferentes plataformas e ambientes. Métricas foram definidas para medir e monitorar aspectos como desempenho, usabilidade, confiabilidade e segurança do sistema.

Todos esses artefatos desempenharam um papel fundamental no processo de desenvolvimento do projeto, fornecendo direcionamento, estrutura e documentação para garantir a entrega de uma solução de qualidade, alinhada com as necessidades dos stakeholders e com os objetivos do projeto.

## **5 Desenvolvimento**

O desenvolvimento de software é um processo complexo e multifacetado que envolve a criação, implementação e manutenção de sistemas de software, que está em constante evolução, com novas tecnologias, metodologias e práticas, abrange uma ampla gama de atividades, desde a concepção inicial de um projeto até a entrega final do produto.

### **5.1 Componentes React**

#### **5.1.1 Componente "Results"**

O componente `Results` é responsável por exibir os resultados de uma pesquisa, permitindo ao usuário filtrar os resultados e realizar ações relacionadas aos resultados.

Figura 16 - Componente “Results”

```

Results.tsx X
front-end > src > components > Results.tsx > Results > options
1  import React, { useEffect, useState } from 'react'
2  import SearchBox from './common/Searchbox';
3  import { bookOptions, operationOptions, userOptions } from '@/data/options';
4  import BookModal from './common/BookModal';
5  import OperationModal from './operationModal/OperationModal';
6  import UserModal from './common/UserModal';
7  import ResultBookBox from './ResultBookBox';
8  import ResultUserBox from './ResultUserBox';
9  import ResultOperationBox from './ResultOperationBox';
10 import search from './utils/search';
11 import myfetch from './utils/myfetch';
12
13 interface IResultProps {
14   type: string;
15 }
16
17 export default function Results({type}:IResultProps) {
18   const [selectedOption, setSelectedOption] = useState<string>('');
19   const [searchValue, setSearchValue] = useState<string>('');
20   const [doSearch, setDoSearch] = useState<boolean>(false);
21   const [result, setResult] = useState();
22
23   const options = () => {
24     if(type === 'book') return bookOptions
25     if(type === 'user') return userOptions
26     if(type === 'operation') return operationOptions
27     return [{value: '', description: ''}]
28   }
29
30   const registerOption = () => {
31     if(type === 'book') return 'Cadastrar novo livro'
32     if(type === 'user') return 'Cadastrar novo usuário'
33     if(type === 'operation') return 'Cadastrar nova operação'
34     return ''
35   }
36

```

Fonte: Autores

1. Importações: O componente importa os módulos React, “useState”, “useEffect”, bem como alguns componentes e funções de utilitário de outros arquivos.
2. Interface “IResultProps”: Define uma interface TypeScript chamada “IResultProps” que especifica a propriedade “type” como uma string.
3. Componente “Results”: Define um componente de função React chamado “Results” que recebe “type” como uma propriedade. O componente possui vários estados, incluindo “selectedOption”, “searchValue”, “doSearch”, “result”, “bookModalOpen”, “userModalOpen”, e “operationModalOpen”.

4. Funções auxiliares “options” e “registerOption”: Retorna as opções disponíveis com base no tipo de resultado e retorna o texto apropriado para o botão de cadastro.

Figura 17 - Componente “Results”

```

37     const handleSelectChange = (value: string) => {
38         setSelectedOption(value);
39     };
40     const handleSearchChange = (value: string) => {
41         setSearchValue(value);
42     };
43
44     async function handleSearchClick() {
45         setDoSearch(true)
46         const url = search(type, selectedOption, searchValue)
47         await getResult(url as string)
48         .then(() => {
49             setDoSearch(false)
50         })
51     };
52
53     async function getResult(url:string){
54         try {
55             var response = await myfetch.get(url)
56             setResult(response)
57             return response
58         } catch (error) {
59             console.error('Erro na pesquisa:', error);
60             return 'Erro na Pesquisa'
61         }
62     }
63
64     const [bookModalOpen, setBookModalOpen] = useState(false);
65     const [userModalOpen, setUserModalOpen] = useState(false);
66     const [operationModalOpen, setOperationModalOpen] = useState(false);

```

Fonte: Autores

5. Funções de manipulação de eventos “handleSelectChange” e “handleSearchChange”: Atualiza o estado “selectedOption” e “searchValue” quando o usuário seleciona ou digita algo nos campos de seleção e pesquisa.

6. Função “handleSearchClick”: Define “doSearch” como “true”, constrói a URL de pesquisa com base no tipo de resultado, opção selecionada e valor de pesquisa, e então chama “getResult” para obter os resultados da API.

7. Função “getResult”: Usa o utilitário “myfetch” para fazer uma solicitação GET à API com a URL fornecida. Se a solicitação for bem-sucedida, atualiza o estado “result”

com os dados retornados. Se ocorrer um erro, exibe uma mensagem de erro no console e retorna uma string indicando um erro na pesquisa.

Figura 18 - Componente “Results”

```

67  const openModal = (entityType: string) => {
68      if (entityType === 'book') {
69          setBookModalOpen(true);
70      } else if (entityType === 'user') {
71          setUserModalOpen(true);
72      } else if (entityType === 'operation') {
73          setOperationModalOpen(true);
74      }
75  }
76
77  return (
78      <>
79          <SearchBox options={options()} onSelectChange={handleSelectChange} onSearchChange={handleSearchChange} className='mt-5' onSearchClick={handleSearchClick}/>
80          <div className='mx-2 border border-gray-300 mt-6 h-auto rounded-lg md:mx-auto relative'>
81              <div className='flex py-2 justify-end border-b border-gray-300 items-center'>
82                  <button className='btn' onClick={() => openModal(type)}>
83                      {registerOption()}
84                  </button>
85              </div>
86              {bookModalOpen && <BookModal closeModal={() => setBookModalOpen(false)} />}
87              {userModalOpen && <UserModal closeModal={() => setUserModalOpen(false)} />}
88              {operationModalOpen && <OperationModal closeModal={() => setOperationModalOpen(false)} />}
89
90
91              {type === 'book'
92                  ? <ResultBookBox result={result}/>
93                  : type === 'user'
94                    ? <ResultUserBox result={result}/>
95                    : <ResultOperationBox result={result}/>
96              }
97          </div>
98      </>
99  )
100 }
101

```

Fonte: Autores

8. Função “openModal”: Abre o modal apropriado com base no tipo de entidade (livro, usuário ou operação).

9. Retorno JSX: Retorna um elemento JSX que contém um componente “SearchBox”, um botão para adicionar um novo registro, e um componente de resultado (“ResultBookBox”, “ResultUserBox” ou “ResultOperationBox”) com base no tipo de resultado.

### 5.1.2 Componente “OpModalContent”

O “OpModalContent” é responsável por renderizar o conteúdo de um modal de operação, onde o usuário pode selecionar um tipo de operação (reserva ou empréstimo), um aluno e um ou mais livros para a operação.

Figura 19 - Componente “OpModalContent”

```

1  import React, { useEffect, useState } from 'react';
2  import SearchBox from '../common/Searchbox';
3  import { bookOptions, operationOptions, userOptions } from '@/data/options';
4  import ResultUserBox from '../ResultUserBox';
5  import ResultBookBox from '../ResultBookBox';
6  import search from '@/src/utills/search';
7  import myfetch from '@/src/utills/myfetch';
8  import { IBookProps, IUserProps } from './OperationUpdateModal';
9
10 interface ISelectedUserProps {
11   cpf: string | undefined;
12   isChecked: boolean;
13 }
14 interface ISelectedBookProps {
15   bookCode: string | undefined;
16   isChecked: boolean;
17 }
18
19 export default function OpModalContent(){
20   const options = (type:string) => {
21     if(type === 'book') return bookOptions
22     if(type === 'user') return userOptions
23     return [{value: '', description: ''}]
24   }
25
26   const [selectedTypeOption, setSelectedTypeOption] = useState<string>('');
27   const [selectedBookOption, setSelectedBookOption] = useState<string>('');
28   const [selectedUserOption, setSelectedUserOption] = useState<string>('');
29   const [searchUserValue, setSearchUserValue] = useState<string>('');
30   const [searchBookValue, setSearchBookValue] = useState<string>('');
31   const [doUserSearch, setDoUserSearch] = useState<boolean>(false);
32   const [userResult, setUserResult] = useState();
33   const [doBookSearch, setDoBookSearch] = useState<boolean>(false);
34   const [bookResult, setBookResult] = useState();
35   const [book, setBook] = useState<IBookProps>();
36   const [user, setUser] = useState<IUserProps>();
37   const [selectedUser, setSelectedUser] = useState<ISelectedUserProps>({
38     cpf: undefined,
39     isChecked: false,
40   });

```

Fonte: Autores

1. Importações: O componente importa o React, o hook “useState”, o componente “SearchBox”, algumas opções de dados, os componentes “ResultUserBox” e “ResultBookBox”, a função “search” e o utilitário “myfetch”.

2. Interfaces TypeScript: O componente define duas interfaces “ISelectedUserProps” e “ISelectedBookProps” que descrevem os tipos de dados esperados para os estados “selectedUser” e “selectedBook”.

3. Componente “OpModalContent”: Este componente possui vários estados, incluindo “selectedTypeOption”, “selectedBookOption”, “selectedUserOption”, “searchUserValue”, “searchBookValue”, “doUserSearch”, “userResult”,

“doBookSearch”, “bookResul”, “book”, “user”, “selectedUser”, “selectedBook”, “userCheckedCPF”, e “userCheckedBookCode”.

Figura 20 - Componente “OpModalContent”

```

41   const [selectedBook, setSelectedBook] = useState<ISelectedBookProps>({
42     bookCode: undefined,
43     isChecked: false,
44   });
45   const [userCheckedCPF, setUserCheckedCPF] = useState<undefined | string>()
46   const [userCheckedBookCode, setUserCheckedBookCode] = useState<undefined | string>()
47   const handleUserChange = (cpf:string | undefined) => {
48     selectedUser.isChecked === true? cpf = undefined : cpf = cpf
49     selectedUser.isChecked === true? setUserCheckedCPF(undefined) : setUserCheckedCPF(cpf)
50     setSelectedUser({ cpf, isChecked: !selectedUser.isChecked });
51   };
52
53   const handleBookChange = (bookCode:string | undefined) => {
54     selectedBook.isChecked === true? bookCode = undefined : bookCode = bookCode
55     selectedBook.isChecked === true? setUserCheckedBookCode(undefined) : setUserCheckedBookCode(bookCode)
56     setSelectedBook({ bookCode, isChecked: !selectedBook.isChecked });
57   };
58
59   useEffect(()=>{
60     if (selectedUser.cpf != undefined){
61       getUserResult(selectedUser.cpf);
62     }
63   },[selectedUser])
64
65   useEffect(()=>{
66     if (selectedBook.bookCode != undefined){
67       getBookResult(selectedBook.bookCode);
68     }
69   },[selectedBook])
70
71   async function getUserResult(CPF: string) {
72     try {
73       var response = await myfetch.get(`/user/cpf/${CPF}`)
74       response = response.shift()
75       setUser(response)
76       return response
77     } catch (error) {
78       console.error('Erro na pesquisa:', error);
79       return 'Erro na Pesquisa'
80     }
81   }

```

Fonte: Autores

4. Hooks de Efeito: Usa os hooks “useEffect” para chamar as funções “getUserResult” e “getBookResult” sempre que os estados “selectedUser” e “selectedBook” forem atualizados, respectivamente.

5. Funções “getUserResult” e “getBookResult”: Fazem solicitações GET à API para obter os detalhes do usuário e do livro associados à operação.

Figura 21 - Componente "OpModalContent"

```
82
83  async function getBookResult(bookCode: string) {
84    try {
85      var response = await myfetch.get(`/book/code/${bookCode}`)
86      response = response.shift()
87      setBook(response)
88      return response
89    } catch (error) {
90      console.error('Erro na pesquisa:', error);
91      return 'Erro na Pesquisa'
92    }
93  }
94
95  async function handleSearchClick(type: string) {
96    let url
97    if(type === "user"){
98      url = search(type, selectedUserOption, searchUserValue)
99      setDoUserSearch(true)
100  }
101  if(type === "book"){
102      url = search(type, selectedBookOption, searchBookValue)
103      setDoBookSearch(true)
104  }
105  await getResult(url as string, type)
106  .then(() => {
107    if(type === "user"){
108      setDoUserSearch(false)
109    }
110    if(type === "book"){
111      setDoBookSearch(false)
112    }
113  })
114  };
```

Fonte: Autores

Figura 22 - Componente “OpModalContent”

```
115
116     function handleSearchUserClick() {
117         handleSearchClick("user")
118     }
119
120     function handleSearchBookClick() {
121         handleSearchClick("book")
122     }
123
124     async function getResult(url:string, type: string){
125         try {
126             var response = await myfetch.get(url)
127             if (type === "user"){
128                 setUserResult(response)
129             }
130             if (type === "book"){
131                 setBookResult(response)
132             }
133             return response
134         } catch (error) {
135             console.error('Erro na pesquisa:', error);
136             return 'Erro na Pesquisa'
137         }
138     }
139
```

Fonte: Autores

6. Funções “handleSearchUserClick” e “handleSearchBookClick”: Disparam as funções de pesquisa quando o usuário clica no botão de pesquisa.

Figura 23 - Componente “OpModalContent”

```

140 async function handleCreateOperation() {
141   if(selectedBook.bookCode != undefined && selectedUser.cpf != undefined && selectedTypeOption){
142     try {
143       console.log(selectedTypeOption)
144       var body = {
145         type: selectedTypeOption,
146         bookCode: selectedBook.bookCode,
147         studentCPF: selectedUser.cpf
148       }
149       await myfetch.post('/operation', body)
150       alert("Operação criada com sucesso!")
151       return
152     } catch (error) {
153       console.error('Erro na criação:', error);
154       return 'Erro na criação'
155     }
156   }
157   alert("Todas as informações precisam ser preenchidas")
158 }
159 }
160

```

Fonte: Autores

7. Função “handleCreateOperation”: Verifica se todas as informações necessárias foram preenchidas e, em seguida, faz uma solicitação POST à API para criar a operação.

Figura 24 - Componente “OpModalContent”

```

161 return (
162   <div className='flex-col text-black'>
163     <h2 className='mb-3 mt-3'>Selecionar operação</h2>
164     <select value={selectedTypeOption} onChange={e => setSelectedTypeOption(e.target.value)} className='rounded-md border bg-white border-gray-300'>
165       <option value=""></option>
166       <option value="RESERVE">Reserva</option>
167       <option value="BORROW">Empréstimo</option>
168     </select>
169     <h2 className='mb-3 mt-3'>Selecionar aluno</h2>
170     <SearchBox options={options('user')} onSelectChange={({value: string}) => setSelectedUserOption(value)} onSearchChange={({value: string}) =>
171       setSearchUserValue(value)} className='mt-5' onSearchClick={handleSearchUserClick}/>
172     {
173       userResult?
174       <ResultUserBox result={userResult} input={true} handleUserChange={handleUserChange} checkedUser={userCheckedCPF}/>
175       :
176       <</>
177     }
178     <h2 className='mb-3 mt-3'>Selecionar livro(s)</h2>
179     <SearchBox options={options('book')} onSelectChange={({value: string}) => setSelectedBookOption(value)} onSearchChange={({value: string}) =>
180       setSearchBookValue(value)} className='mt-5' onSearchClick={handleSearchBookClick}/>
181     {
182       bookResult?
183       <ResultBookBox result={bookResult} input={true} handleBookChange={handleBookChange} checkedBook={userCheckedBookCode}/>
184       :
185       <</>
186     }
187     <div className='flex flex-col mt-8 space-y-4'>
188       <p className='text-xl text-center pb-4'>Resumo da operação</p>
189       <p>Nome do aluno: {user?.name}</p>
190       <p>CPF do aluno: {user?.cpf}</p>
191       <p>Título do livro: {book?.title}</p>
192       <p>Autor do livro: {book?.author}</p>
193       <p>Tipo de operação: {
194         selectedTypeOption === "reserve"? "Reserva" :
195         selectedTypeOption === "borrow"? "Empréstimo" : ""
196       }</p>
197     </div>
198     <div className='flex flex-row space-x-8 justify-end mt-4'>
199       <button className='btn' onClick={handleCreateOperation}>Confirmar operação</button>
200     </div>
201   </div>
202 )
203 }
204

```

Fonte: Autores

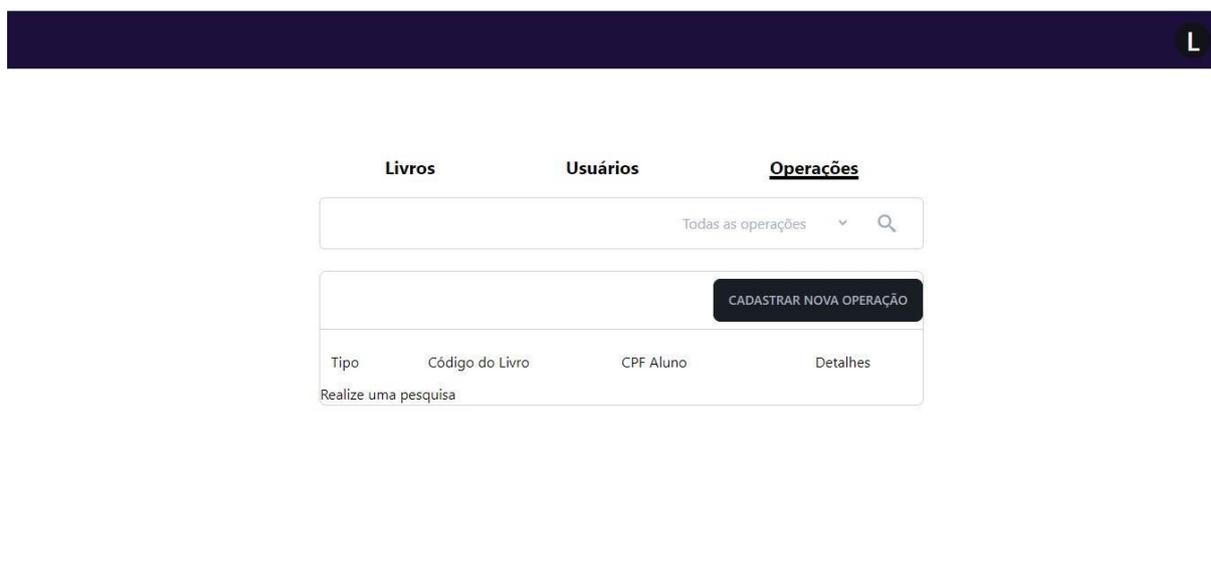
8. Retorno JSX: Retorna um elemento JSX que exibe um dropdown para selecionar o tipo de operação, um componente “SearchBox” para selecionar um usuário, um componente “ResultUserBox” para exibir os resultados da pesquisa de usuário, um componente “SearchBox” para selecionar um livro, um componente “ResultBookBox” para exibir os resultados da pesquisa de livro, e um botão para confirmar a operação.

Em suma, este componente React é responsável por renderizar o conteúdo de um modal de operação, onde o usuário pode selecionar um tipo de operação, um usuário e um ou mais livros para a operação. Ele usa estados para gerenciar o comportamento e os dados do componente e funções auxiliares para lidar com as solicitações à API.

## 5.2 Tela

A tela de operações é uma parte fundamental do sistema de gerenciamento de biblioteca desenvolvido para esse projeto. Ela permite que os usuários realizem operações de reserva e empréstimo de livros de forma eficiente e conveniente simplificando o processo e proporcionando uma experiência fluida e intuitiva para os usuários da biblioteca.

**Figura 25** – Tela de “operações”



Fonte: Autores

## 6 Resultados e Discussão

### 6.1 PORTABILIDADE

A infraestrutura de uma empresa ou instituição é essencial para garantir que as atividades possam ser realizadas com eficiência e segurança. Nesse sentido, a rede de dados é um dos principais elementos da infraestrutura de TI e deve ser projetada para atender às necessidades do negócio.

6.1.1 Configuração mínima para o sistema de gerenciamento de empréstimo e devoluções de livros.

As configurações mínimas para um sistema web de gerenciamento de livros podem variar dependendo do número de usuários simultâneos ou se houver muitas operações sendo executadas ao mesmo tempo. No entanto, aqui estão algumas configurações mínimas de CPU recomendadas para esse sistema:

- a) Processador: Intel Core i3 ou equivalente
- b) Memória RAM: 4 GB
- c) Armazenamento: pelo menos 500 GB de armazenamento em disco rígido
- d) Sistema operacional: Windows 10 ou Linux
- e) Navegador: as versões mais recentes do Google Chrome, Mozilla Firefox ou Microsoft Edge
- f) Banco de dados: MySQL ou PostgreSQL

### 6.2 MÉTRICAS

Para realizar o cálculo das métricas, para estabelecer o valor de investimento para a solução sistêmica, foi tomado por base o diagrama de classes e seus atributos.

Cada classe foi denominada aqui em tabela, então, temos:

- a) Tabela Biblioteca: 4 atributos
- b) Tabela Pessoa: 6 atributos
- c) Tabela Transação Livro: 3 atributos
- d) Tabela Livro: 8 atributos
- e) Tabela Funcionário: 1 atributo
- f) Tabela Alunos: 2 atributos
- g) Tabela Reserva: 3 atributos

- h) Tabela Empréstimo: 2 atributos
- i) Tabela Geral: 29 atributos (somatória de todos os atributos das tabelas)

Para estabelecer o nível de complexidade de entradas, saídas, consulta, arquivos e interface, foram utilizadas como parâmetro as tabelas sobre métricas orientadas à função, baseado em Albrecht (1979). Assim, temos:

- a) Entrada: 8 tabelas de nível de complexidade MÉDIO (desconsidera tabela Geral)
- b) Saída: 9 tabelas de nível de complexidade MÉDIO (computa-se tabela Geral)
- c) Consulta: 9 tabelas de nível de complexidade MÉDIO (computa-se tabela Geral)
- d) Arquivo: Entrada: 8 tabelas de nível de complexidade MÉDIO (desconsidera tabela Geral)
- e) Interface: 1 tabela de nível de complexidade SIMPLES e 8 tabelas de nível de complexidade MÉDIO (computa-se tabela Geral)

Então, foi possível utilizar uma planilha de Excel para o cálculo do valor do projeto, baseado na complexidade dos requisitos, linguagem utilizada (JavaScript), o tipo de sistema (Sistema Web), dias e horas trabalhadas (22 dias e 6h/dia), também o nível de influência do sistema. Como resultado, foi obtido o valor de R\$ 4.114,80 como investimento para o desenvolvimento da solução sistêmica, conforme figura 25.

Figura 26 – Métricas

MÉTRICAS				
Função	Nº de ocorrência	Complexidade	Peso	Resultado
entradas	0	Simple	3	0
	8	Médio	4	32
	0	complexo	6	0
saídas	0	Simple	4	0
	9	Médio	5	45
	0	complexo	7	0
consultas	0	Simple	3	0
	9	Médio	4	36
	0	complexo	6	0
arquivos	0	Simple	7	0
	8	Médio	10	80
	0	complexo	15	0
interfaces	1	Simple	5	5
	8	Médio	7	56
	0	complexo	10	0
<b>Total de FP'b</b>				<b>254</b>

Estimativas do número médio de LOC por FP	
Cobol	100
Pascal	90
JavaScript/TypeScript	60
Linguagens Orientadas a Objeto (C++)	30
Java / Delphi / Visual Basic	20

Tipo de Sistema	Produtividade - Kloc/Loc /mês
Sistema Comercial	2.500
Comércio Eletrônico	3.600
Sistema Web	3.300

PRAZO	
DIAS/MÊS	22
HORAS/DIA	6
MINUTOS/HORA	60

Nível de Influência do Sistema (0 a 5)	
Comunicação de dados	5
Performance	5
Volume de transações	5
Eficiência do usuário final	5
Processamento complexo	5
Facilidade de implantação	5
Múltiplos locais	5
Processamento distribuído	5
Utilização de equipamento	5
Entrada de dados on-line	5
Atualização on-line	5
Reutilização de código	5
Facilidade operacional	5
Facilidade de mudanças	5
<b>Total de NI</b>	<b>70</b>
FA = multiplicar o NI pela taxa real = $0,65 + (0,01 * 119)$	<b>1,35</b>
FP'r = multiplicar o FP'b pelo FA	<b>343</b>
<b>Informe o nº de LOC da Linguagem</b>	<b>60</b>
KLOC = Multiplicar o FP'r pelo tipo de linguagem	
<b>Total de KLOC</b>	<b>20574</b>
<b>Informe o tipo de sistema</b>	<b>3.300</b>
<b>PRAZO (dividir o KLOC pelo tipo de sistema)</b>	
(RESULTADO DA DIVISÃO) QTDD DE MESES	<b>6,23</b>
QTDD DE DIAS	
QTDD DE HORAS	
QTDD DE MINUTOS	
<b>CUSTO - INVESTIMENTO</b>	
<b>Informe o valor da hora de trabalho</b>	<b>R\$ 20,00</b>
<b>ISO (NORMA INTERNACIONAL) HORAS/MÊS =</b>	<b>132</b>
FÓRMULA = $132 * \text{RESULTADO DA DIVISÃO} * \text{VALOR DA HORA}$	
<b>VALOR TOTAL DO PROJETO =</b>	<b>R\$ 16.459,20</b>
PROVA DE CONFIRMAÇÃO (DIAS * HORAS * VALOR DA HORA)	

Fonte: Autores

### 6.3 PROPOSTA COMERCIAL

A gestão de uma biblioteca pode ser uma tarefa desafiadora, especialmente quando se trata da manutenção da organização e acessibilidade do acervo. Com a grande quantidade de livros, periódicos, materiais audiovisuais e outros tipos de mídia que as bibliotecas possuem, é fundamental ter um sistema que possa assegurar a integridade, organização e operacionalização do acesso ao acervo.

No âmbito do projeto, foram definidos os requisitos do sistema, realizada a análise e modelagem do processo de empréstimo e devolução de livros, com o objetivo de alcançar uma gestão eficiente, a ferramenta busca automatizar e agilizar o processo de empréstimo e devolução, reduzindo o tempo de espera dos usuários e

aprimorando a eficiência do serviço. Além disso, visa melhorar o controle e a segurança do acervo, garantindo que os livros emprestados estejam registrados no sistema e possam ser facilmente localizados. Também tem o propósito de facilitar a gestão dos prazos de empréstimo e devolução, proporcionando aos usuários informações antecipadas sobre as datas de devolução, a fim de evitar atrasos ou penalidades. A ferramenta busca oferecer uma experiência positiva aos usuários, com um sistema fácil de usar, intuitivo e que forneça informações precisas e atualizadas sobre o acervo e os empréstimos.

Com essa ferramenta tecnológica, a biblioteca poderá enfrentar os desafios da gestão do acervo de forma mais eficiente, garantindo a integridade do acervo e facilitando o acesso dos usuários, além de aprimorar a qualidade dos serviços oferecidos.

### 6.3.1 Solução proposta

A solução proposta é um Sistema de Gestão de Empréstimos e Devoluções de Livros de Bibliotecas, uma plataforma tecnológica abrangente desenvolvida para auxiliar na gestão eficiente do acervo, empréstimos, devoluções e acesso dos usuários à biblioteca. Com base nos desafios enfrentados pelas bibliotecas na manutenção da organização e acessibilidade do acervo, nossa solução busca automatizar e aprimorar os processos, garantindo uma experiência satisfatória para os usuários.

#### 6.3.1.1 Descrição da Solução

- a) Empréstimos e Devoluções Simplificados: Nossa solução automatiza e agiliza o processo de empréstimo e devolução de livros. Por meio de um sistema intuitivo, os usuários podem realizar essas operações de forma rápida e eficiente, reduzindo o tempo de espera e proporcionando uma experiência positiva.
- b) Controle de Prazos e Penalidades: A plataforma oferece recursos para o controle de prazos de empréstimo e devolução, proporcionando aos usuários informações antecipadas sobre as datas de devolução. Isso ajuda a evitar atrasos e penalidades, garantindo uma gestão mais eficiente.

- c) Acesso Online ao Catálogo: Nossa solução inclui uma plataforma digital que permite aos usuários acessarem o catálogo da biblioteca de forma fácil e rápida. Eles podem realizar pesquisas, visualizar informações detalhadas sobre os livros e fazer reservas online, proporcionando conveniência e agilidade.

Melhorias significativas esperadas:

- a) Melhoria na gestão do acervo, elevando a organização e a integridade do mesmo. - Acesso fácil e rápido ao catálogo da biblioteca, permitindo pesquisas e reservas online.
- b) Aprimoramento na qualidade dos serviços prestados, resultando em maior satisfação dos usuários.

### 6.3.2 Prazos e investimentos

Prazo: 6 meses.

Investimento: R\$ 16.459,20.

### **Considerações finais**

A proposta do trabalho era desenvolver uma solução sistêmica para solucionar a problemática de empréstimo de livros e manutenção de acervo da biblioteca da UAB Polo Franca.

Apesar da não intenção de implementar a solução sistêmica, o projeto foi desenvolvido de modo a cumprir todas as etapas de documentação, desde o termo de abertura de projeto, compreensão das regras de negócio, requisitos básicos e essenciais ao sistema, métricas para cálculo de investimentos para desenvolver o sistema e a proposta comercial.

Ao longo do processo, foram utilizadas diversas ferramentas e metodologias para garantir o seu sucesso. A integração de ferramentas como BPMN, Canvas, SWOT, 5W2H e outras foi fundamental para o planejamento e execução eficiente do sistema. O uso do Draw.io e DaisyUI para a criação de diagramas e interfaces gráficas, respectivamente, proporcionou uma visualização clara e concisa das etapas e componentes do sistema.

Além disso, o uso do Next.js e Tailwind CSS permitiu uma fácil e rápida implementação das interfaces, garantindo um design moderno e responsivo. A escolha do JavaScript e React como tecnologias principais para o desenvolvimento

do sistema mostrou-se acertada, pois ambas proporcionaram uma experiência de desenvolvimento ágil e eficiente. O uso do TypeScript também foi uma decisão acertada, pois trouxe benefícios significativos em termos de segurança e manutenibilidade do código.

Em relação aos objetivos futuros, dois pontos se destacam como metas a serem alcançadas: facilitar a gestão dos prazos de empréstimo e devolução e possibilitar a integração com outros sistemas e plataformas. Ambos os objetivos têm o potencial de melhorar significativamente a eficiência e a eficácia do sistema.

Para atingir o primeiro objetivo, é necessário desenvolver uma funcionalidade que permita aos usuários visualizar os prazos de empréstimo e devolução de forma clara e intuitiva. Isso pode ser feito por meio da criação de um calendário ou lista de tarefas, que mostre os livros emprestados e suas datas de devolução.

Já o segundo objetivo envolve a criação de uma API que permita a integração do sistema de gerenciamento de empréstimos e devoluções com outros sistemas e plataformas, como o sistema de gestão de bibliotecas. Isso possibilitará uma gestão mais integrada e eficiente da biblioteca como um todo, facilitando o acesso e a atualização de informações entre os diferentes sistemas.

Este projeto trouxe grande aprendizado, pois envolveu diversas etapas que requereram bastante organização quanto à divisão de tarefas, cumprimento de prazos e trabalho em equipe, situação bastante similar ao que é encontrado no mercado de trabalho. O manejo do tempo, estudo, consistência na execução das tarefas, atenção, trabalho em equipe, maturidade na discussão de soluções para os desafios do projeto e conciliação das diversas atividades foram essenciais para a boa condução do projeto.

## Referências

ALBRECHT, A.J. (1979) Measuring Application Development Productivity. Proceedings of IBM Applications Development Symposium, Monterey, 14-17 October 1979, 83.

BRUM, Bruno Conde Perez e PENA, Leandro. Principais Técnicas de Levantamento de Requisitos. Reflectz's Blog. Disponível em: <<https://brunobrum.wordpress.com/2011/04/27/principais-tecnicas-de-levantamentode-requisitos-de-sistemas/>>. Acesso em: 04/06/2023

CANVA. Disponível em: <<https://www.canva.com/>>. Acesso em: 12/03/2024.

COSTA, L. F. da; GOMES, J. de O.; DUARTE, E. N.; VALENTIM, M. L. P. Reflexões sobre o estado da arte da gestão da informação e do conhecimento por Emeide Nóbrega Duarte. **Perspectivas em Gestão & Conhecimento**, [S. l.], v. 8, n. 2, p. 194–204, 2018.

DAISYUI. Disponível em: <<https://daisyui.com/>>. Acesso em: 12/03/2024.

DRAW.IO. Disponível em: <[draw.io \(diagrams.net\)](https://draw.io)>. Acesso em: 12/03/2024.

FERNANDES, R. Engenharia de software. 1ª ed. Curitiba: Fael, 2017.  
Google documentos. Disponível em: <<https://docs.google.com/document/u/0/>>. Acesso em: 12/03/2024.

GOOGLE PLANILHA. Disponível em: <<https://docs.google.com/spreadsheets/u/0/>>. Acesso em: 12/03/2024.

JAVASCRIPT. Disponível em: <<https://www.javascript.com/>>. Acesso em: 12/03/2024.

MORAES, M. H. M. de. As tecnologias de informação e comunicação contribuindo para a disseminação da produção científica. **BIBLOS - Revista do Instituto de Ciências Humanas e da Informação**, Rio Grande, v. 26, n. 1, 2012.

NEXT.JS. Disponível em: <<https://nextjs.org/>>. Acesso em: 12/03/2024.

OSTERWALDER, A. PIGNEUR, Y. Business Model Generation: Inovação em Modelos de Negócios. 5º ed. Rio de Janeiro: ALTA BOOKS, 2013.

PERFETTO, F. V.; REIS, S. G. de O.; PALETTA, F. C. Gestão da informação digital: caminhos possíveis. **RDBCI: Revista Digital De Biblioteconomia E Ciência Da Informação**, v. 21, 2023.

POSTGRESSQL. Disponível em: <<https://www.postgresql.org/about/>>. Acesso em: 11/06/2024.

PRESSMAN, R. “Engenharia de Software: Uma Abordagem Profissional”. 9ª ed. Porto Alegre: AMGH, 2021.

REACT. Disponível em: <<https://pt-br.legacy.reactjs.org/>>. Acesso em: 12/03/2024.

SANTA ANNA, J. Trajetória histórica das bibliotecas e o desenvolvimento dos serviços bibliotecários: da guarda informacional ao acesso. **RDBCI: Revista Digital de Biblioteconomia e Ciência da Informação**, v. 13, n. 1, p. 138, 2015.

SERRAI, A. História da biblioteca como evolução de uma ideia e de um sistema. **Revista da Escola de Biblioteconomia da UFMG**. UFMG, Belo Horizonte, 4(2) : 141-61, set. 1975.

SILVA, R. C. d.; CALDAS, R. F. Bibliotecas híbridas e o desenvolvimento das sociedades. **RDBCI: Revista Digital de Biblioteconomia e Ciência da Informação**, v. 20, 2022.

SOMMERVILLE, I. Engenharia de software. 9ª ed. São Paulo: Pearson Prentice Hall, 2011.

SOUZA, F. P. d. **Impacto das tecnologias da informação e comunicação no tratamento da informação**. 2011. 52 f. Trabalho de Conclusão de Curso (Graduação em Biblioteconomia) – Centro de Ciências da Educação, Universidade Federal de Santa Catarina, Florianópolis, 2011.

TAILWIND CSS. Disponível em: <<https://tailwindcss.com/>>. Acesso em: 12/03/2024.

TYPESCRIPT. Disponível em: <<https://www.typescriptlang.org/>>. Acesso em: 12/03/2024.

VISUAL STUDIO CODE. Disponível em: <<https://code.visualstudio.com/>>. Acesso em: 12/03/2024.

## **APÊNDICE A – DESCRIÇÃO UAB**

### **1. UNIVERSIDADE ABERTA DO BRASIL (UAB) POLO FRANCA**

A UAB é um sistema integrado por universidades públicas que oferece cursos superiores a distância para pessoas que têm dificuldade de acesso à formação universitária, com prioridade para professores e trabalhadores da educação básica. O sistema funciona como articulador entre as instituições de ensino superior e os governos locais, visando atender às demandas locais por educação superior. O Polo UAB Franca oferece cursos gratuitos, à distância, em graduação e pós-graduação para qualquer pessoa interessada.

### **2.1 MISSÃO, VISÃO E VALOR**

#### **2.1.2 Missão**

Promover o acesso à educação superior pública, gratuita e de qualidade a distância, contribuindo para a formação de profissionais qualificados e para o desenvolvimento social e econômico do país.

#### **2.2.2 Visão**

Ser referência nacional e internacional em educação a distância, oferecendo cursos superiores de excelência que atendam às necessidades da sociedade e do mercado de trabalho.

## 2.2.3 Valor

- a) Democratização do acesso à educação superior;
- b) Qualidade e excelência acadêmica;
- c) Inovação e atualização constante;
- d) Respeito à diversidade cultural e social;
- e) Comprometimento com o desenvolvimento sustentável e com a transformação social.

**APÊNDICE B – ENTREVISTA**

PERGUNTAS	RESPOSTA DA ENTREVISTA
1-Quem pode solicitar empréstimo?	Apenas alunos.
2-Quem pode realizar empréstimo?	O aluno pode realizar uma pré-reserva do livro e ir retirar no polo. O funcionário pode emprestar diretamente. Porém existem algumas formas de confirmar o empréstimo como: aluno e funcionário confirmarem no sistema ou aluno confirmar com senha/pin.
3-Quem pode cadastrar usuários? E como é realizado?	O próprio usuário e funcionário. Porém o funcionário no primeiro empréstimo precisa confirmar a identidade do aluno verificando um documento com foto.

<p>4-Quantos dias para empréstimo? Quantas vezes seguida? Reservar mais de uma vez ao mesmo tempo é possível? Quantos dias de tolerância para buscar o livro reservado?</p>	<p>15 dias, não tem máximo de vezes seguidas, desde que respeite a fila. Não é possível reservar mais de uma vez ao mesmo tempo, e tem 1 dia útil para buscar o/os livro(s) reservado(s)</p>
<p>5-Quem gerencia o acervo?</p>	<p>Os empréstimos e as devoluções de livros podem ser feitas por qualquer funcionário do Polo UAB Franca</p>
<p>6-Como funciona a mobilidade intra universidade?</p>	<p>Um aluno de uma universidade A pode pegar emprestado os livros de outra universidade B, porém apenas de livros em posse do Polo UAB Franca.</p>
<p>7-Como retirar o(s) livro(s)?</p>	<p>No próprio polo das 13h às 21h de segunda à sexta.</p>
<p>8-Limite de livros por empréstimo?</p>	<p>Dois por vez.</p>
<p>9- Existe algum tipo de penalidade caso um aluno atrase a entrega do livro e/ou não o devolva?</p>	<p>Não se cobra multa, infrações serão aplicadas após atraso.</p>
<p>10-Aba de pedidos para aquisição no acervo?</p>	<p>Não.</p>

11-Cadastrar patrimônio?	Não.
12-Aceita doação?	Sim.
13-Como saber se a matrícula do aluno está ativa?	São checadas semestralmente, e atualizadas no banco de dados.