



ThothLib

Gestão inteligente, é com a gente


- Hanan Ortiz
- José Eduardo
- Luis Gustavo
- Lucas Alves
- Lucas Donizete
- Matheus Alencar

Contextualização

A thick yellow line that starts in the upper right, curves downwards and to the left, and then loops back to the right.

- Instituições e suas bibliotecas
- Malefícios do atual método de gestão dos livros

"O excesso de impressões, muitas vezes desnecessárias, leva ao consumo exagerado de papel e, por sua vez, o consumo de árvores e de todos os recursos utilizáveis na produção de papel (RECK,2007)."

Two overlapping light blue circular shapes in the bottom right corner of the slide.

User Stories

1.1

eu como admin da biblioteca quero um sistema que faça controle de estoque para otimizar meu tempo

2.1

eu como usuário da biblioteca quero ter uma forma mais ágil de verificar o acervo para poder ser mais rápido na hora de selecionar um livro

3.1

eu como usuário da biblioteca quero poder realizar a seleção do livro de maneira digital para conseguir ser mais ágil na retirada

5.1

eu como professor da Bandtec quero um sistema para verificar o histórico de retiradas para poder oferecer bonificação para alunos mais ativos

1.2

eu como admin da biblioteca quero poder realizar o cadastro digital dos livros para obter uma maior eficácia na gestão

2.2

eu como usuário da biblioteca quero poder visualizar uma lista com todos os livros disponíveis para saber quais livros já estão prontos para uso

4.1

eu como usuário da biblioteca quero receber alertas com os prazos de entrega para garantir uma entrega no prazo correto

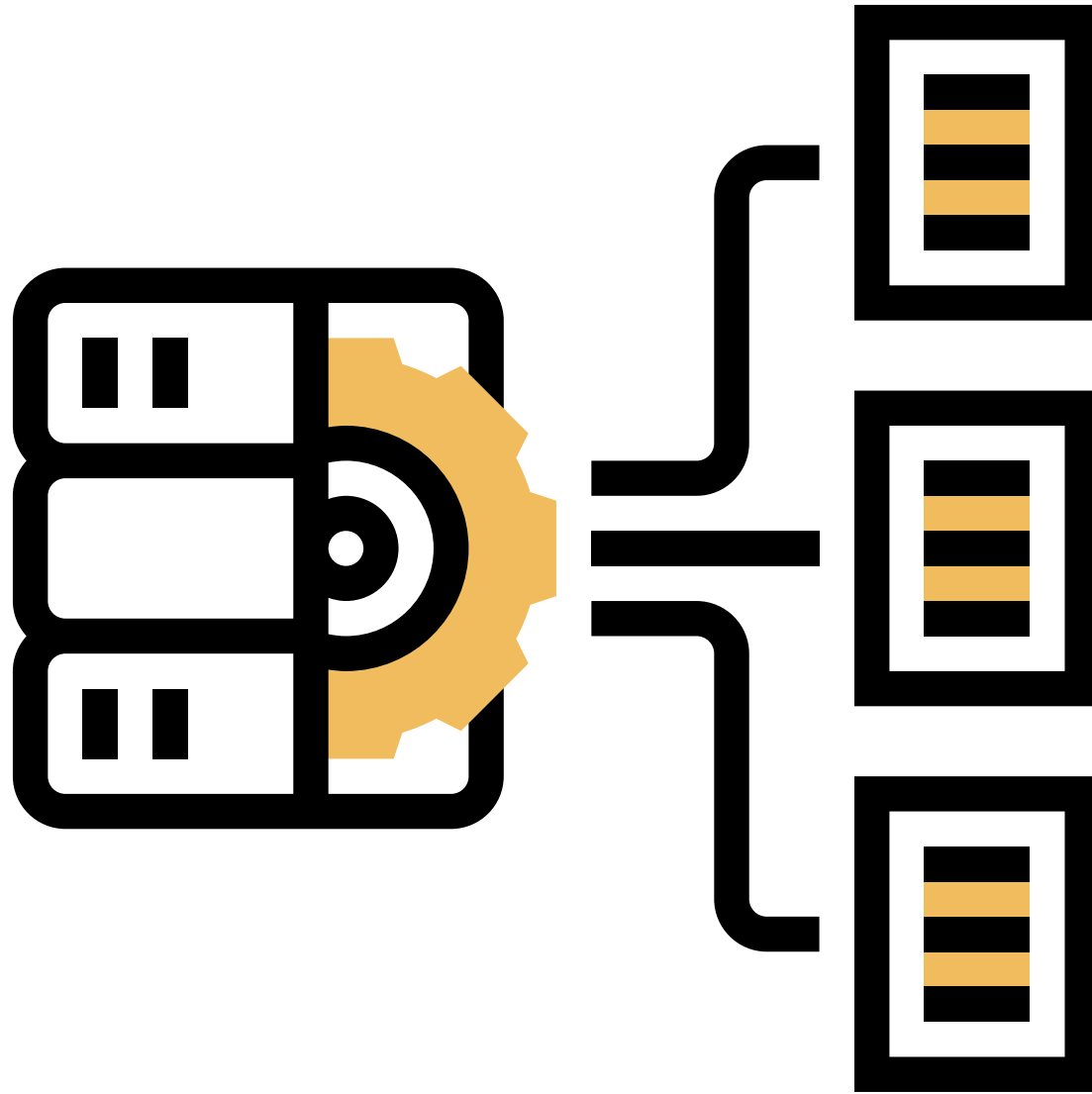
5.2

eu como professor da Bandtec quero poder identificar quais alunos mais ativos na biblioteca para oferecer interações com a aula

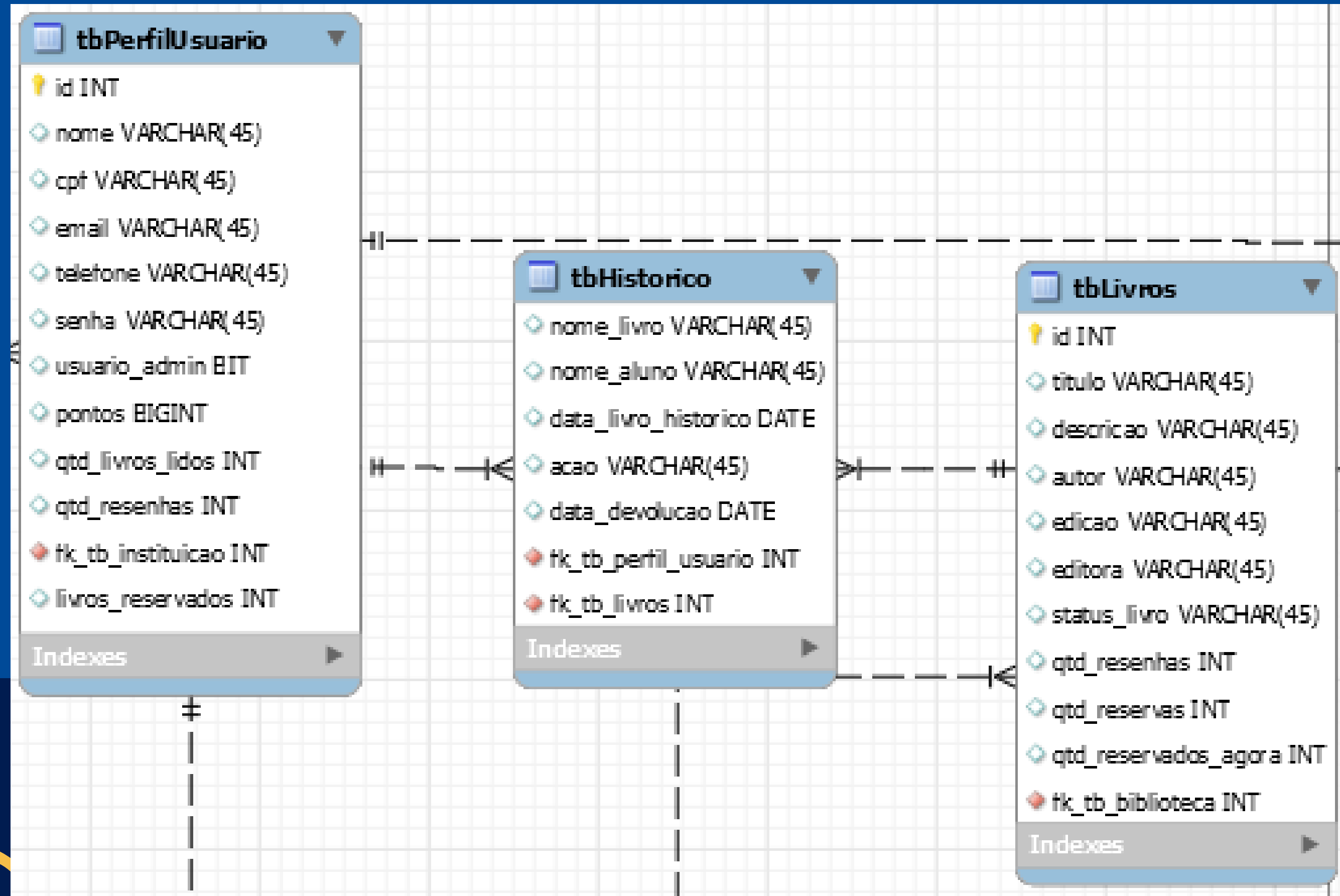
Product Backlog

Estado▼	Funcionalidades▼	Descrição de requisitos▼	Essencia▼	Importante▼	Desejave▼
Concluido	Cadastro<US1.1>	Cadastrar Usuario Admin	X		
Concluido	Cadastro<US1.2>	O sistema deverá cadastrar livros	X		
Concluido	Cadastro<US2.2>	Cadastrar usuario Comum	X		
Planejamento	Notificação<US1.1>	A aplicação deverá notificar caso o prazo esteja acabando			
Concluido	Aplicação<US2.1>	A aplicação deverá ter a relação de livros disponiveis		X	
Concluido	Aplicação<US3.1>	A aplicação deverá reservar livros para o usuario	X		
Execução	Aplicação<US2.3>	aluno publica algo que ele achou relevante sobre o livro			
Concluido	Aplicação<US1.1>	A aplicação deve ter a relação de livros Indisponiveis	X		
Concluido	Aplicação<SB>	A aplicação deverá atrelar o livro ao usuario logo após a retirada	X		
Concluido	Aplicação<US2.1>	A Aplicação deverá ter uma lista de todos livros.		X	
execução	Aplicação<US2.3>	A aplicação deverá rankear os livros mais usados			X

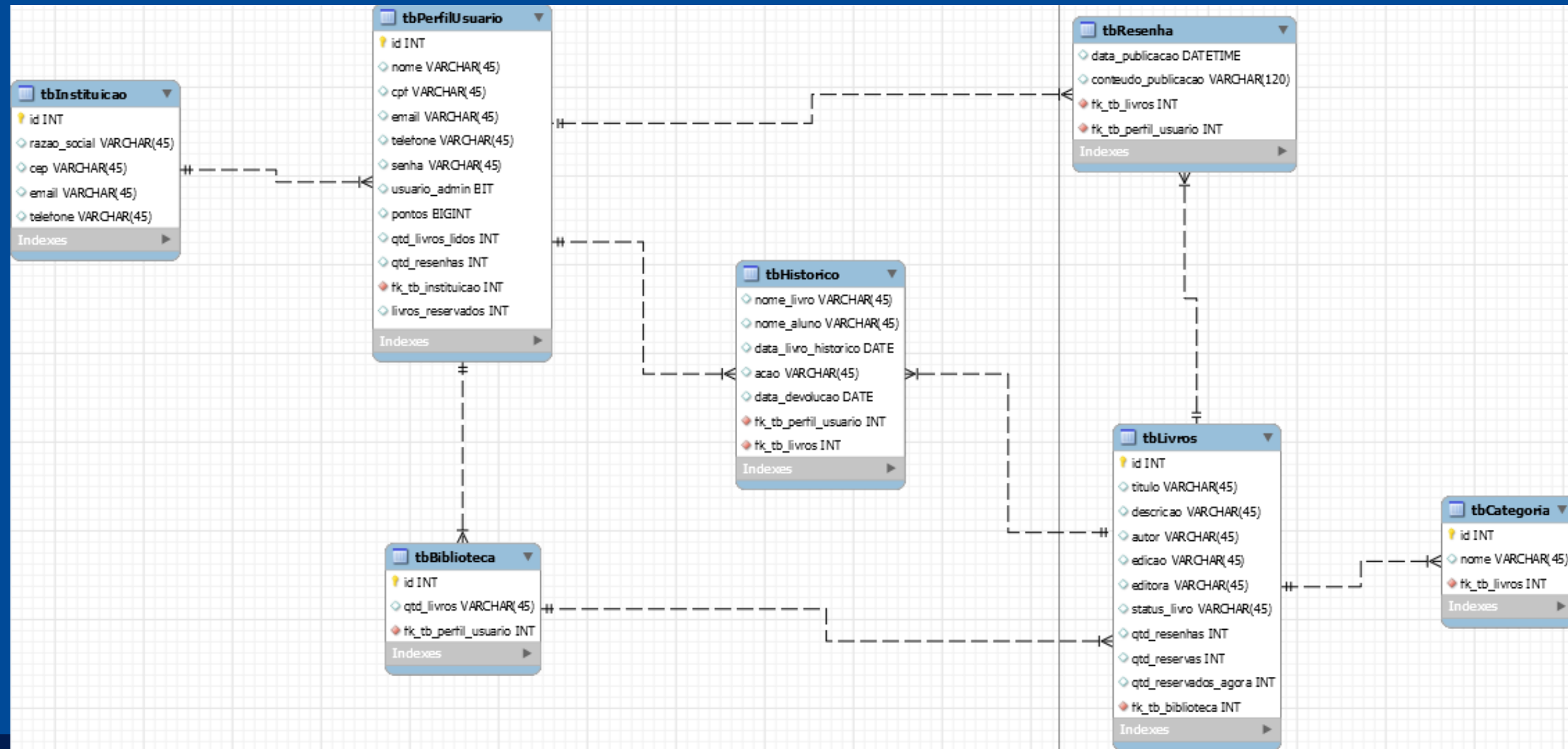
Modelo Lógico



Modelo Lógico



Modelo Lógico



Qualidade

Classes de teste unitario

```
@Before
public void setup() {...}

@Test
public void BuscaLivroPorIdTest() {

    Integer idTeste = 90000;

    Optional<Livros> livroTeste = usuario.buscarLivro(idTeste);

    Assertions.assertEquals( expected: 90000, livroTeste.get().getId());
    Assertions.assertEquals( expected: "TesteJunit", livroTeste.get().getTitulo());
    Assertions.assertEquals( expected: 0, livroTeste.get().getQtdReservadoAgora());
}
```


Qualidade

Classes de teste unitario

```
@Test
public void RetirarTest() {

    UsuarioAdmin teste = Mockito.mock(UsuarioAdmin.class);

    Integer idRegistro = 6666;
    Integer idLivro = 8888;
    Integer idUsu = 9999;
    String tipo = "Retirada";

    teste.criaResgistroComIdRegistro(idRegistro, idLivro, idUsu, tipo);
    Mockito.verify(teste).criaResgistroComIdRegistro(idRegistro, idLivro, idUsu, tipo);
    Mockito.verify(teste, Mockito.times(wantedNumberOfInvocations: 1)).criaResgistroComIdRegistro(idRegistro, idLivro, idUsu, tipo);
}
```

Qualidade

Classes de teste unitario

```
@Autowired
MockMvc mockMvc;

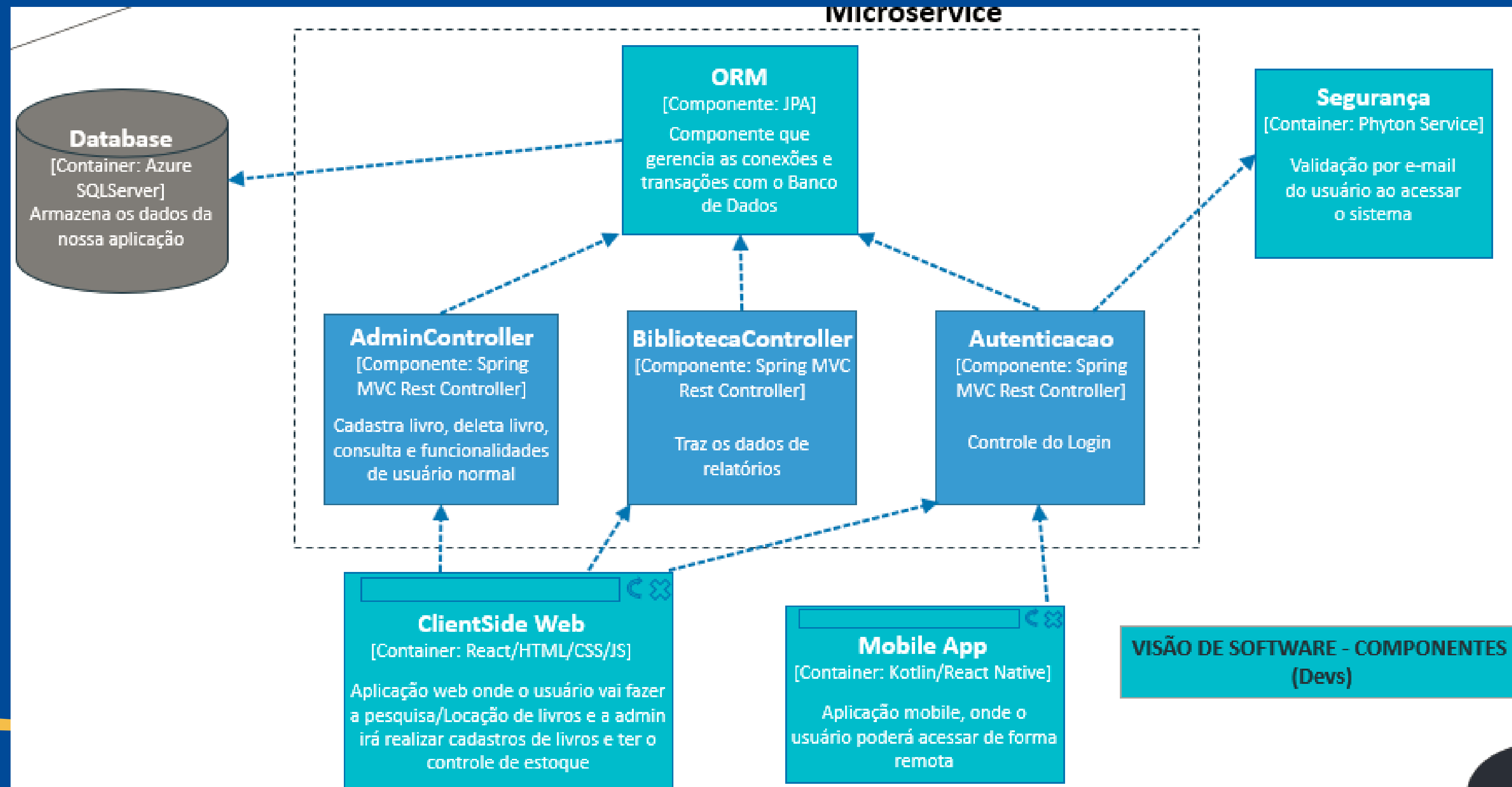
@Test
public void adminControllerTestListUsuariosAdmin() throws Exception {
    mockMvc.perform(get(urlTemplate: "/admin")).
        andExpect(status().isOk());
}
```

Qualidade

Planilha UAT

Cenário	Condição/Iteração com sistema	Massa de Teste	Esperado / Valor esperado	Status – Desenvolvimento
Pesquisar Livro	Logar no sistema, entrar na tela de listagem de livros e digitar no input o nome do livro	Nome do Livro: Java	Trazer A lista de livro com o titulo: "JAVA".	OK
Pesquisar Livro	Logar no sistema, entrar na tela de listagem de livros e digitar no input o nome do livro	Nome do Livro: ""	O sistema ira exibir um pop-up de erro: "Digite no minimo 3 letras"	OK
Pesquisar Livro	Logar no sistema, entrar na tela de listagem de livros e digitar no input o nome do livro	Nome do Livro: "fenfiawnfiawfawf"	O sistema ira exibir um pop-up de erro: "Não existe um livro com esse nome"	OK
Filtrar Livro	Logar no sistema, entrar na tela de listagem de livros e clicar na opção de "filtrar por", escolher categoria e digitar a categoria desejada	Filtrar por : "Categoria" e digitar : "Tecnologia".	O sistema deve exibir apenas livros da categoria "Tecnologia".	OK
Filtrar Livro	Logar no sistema, entrar na tela de listagem de livros e clicar na opção de "filtrar por", escolher categoria e digitar a categoria desejada	Filtrar por : "Categoria" e digitar : "ijaoinfaoiwnfiaw".	O sistema ira exibir uma mensagem de erro: "Categoria não encontrada"	Ok

Diagrama de componentes



Aplicação web

- Fila
- Pilha
- API Google Books
- Exportação/Importação de arquivo de layout
- API Python
- Fluxos de locação dos livros
- Gameificação



Swagger

Tothlibs API REST 1.0.0

[Base URL: tothlibs.azurewebsites.net/]

<https://tothlibs.azurewebsites.net/v2/api-docs>

Essa é uma API para gerenciamento de bibliotecas

[Terms of service](#)

[@Tothlibs - Website](#)

[Send email to @Tothlibs](#)

[Apache License Version 2.0](#)

admin-controller Admin Controller



GET /admin Retorna uma lista de usuarios administradores

POST /admin Realiza o cadastro de um administrador

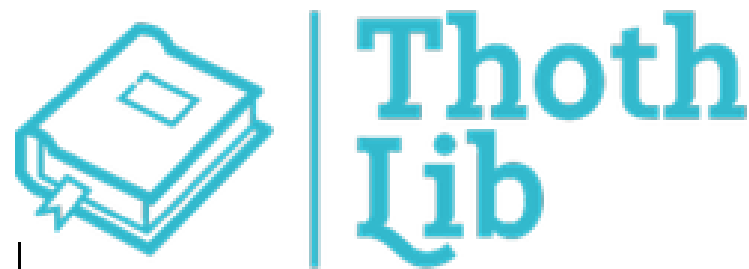
GET /admin/{idAdmin} Retorna um administrador com ID especifico



Conclusão

- Problemas com o banco de dados
- Muita demanda em pouco tempo
- Desafios superados
- Obrigado pela atenção

White Paper



Autores: Hanan ortiz, José Eduardo, Lucas Alves, Lucas Donizete, Luis Gustavo, Matheus Alencar

Alunos do Curso de Tecnologia em Análise e Desenvolvimento de Sistemas – 3º Semestre 2021-2

Orientador: Professor Alex Barreira
SPTech– São Paulo – SP

Resumo

Este projeto trata-se do desenvolvimento de um aplicação para utilização em computadores (desktop e laptop), otimizando a gestão e controle de estoque de livros nas Instituições de Ensino.

Os grupos de usuários que farão uso da aplicação serão: Administradores responsáveis pela gestão e controle de estoque da biblioteca; Alunos da instituição,

Os maiores motivadores para o desenvolvimento desse projeto foram: Garantir maior segurança em processos de retirada e devolução no estoque;

Agilizar o trabalho do administrador em validar as pendências para com a biblioteca;
Sustentabilidade na exclusão da utilização de papel para relatórios, controle de entrada e saída dos livros;

Introdução

Em instituições de ensino pelo Brasil ainda utilizam a forma tradicional de gestão de entrada e saída de livros, ou seja, de forma escrita em planilhas de papel físico é uma realidade presente em diversos estados

Nessa atual forma de gestão podem ocorrer diversos problemas como: falta de controle de estoque, perda(perda) de livros, rasura do papel, pouca agilidade, pouca sustentabilidade. E

esses problemas desencadeiam diversos outros problemas, De acordo com o Site

<http://aems.edubr/>

“Segundo o Instituto Akatu, instituição que estimula o consumo consciente, a produção de papel está entre os processos industriais que mais consomem água. Para entender o sistema industrial do setor necessita 540 litros de água para produzir um quilo de papel.”

“...!” para fazer uma folha de papel A4 são necessárias 10 litros de água. ”

E isso acaba não sendo sustentável visto que temos outras formas de resolver essa mesma questão sem utilizar papel, além da água também devemos observar todo o desmatamento causado para utilizar todo esse papel

Solução

Planejamento

O nosso projeto passou por uma fase de planejamento bem estruturada, nela passamos por diversas ferramentas, técnicas e metodologias como user journey map, mapa de empatia, LeanUX, Story board, Gantt e Proto Persona. Essa fase foi muito importante para visualizarmos onde nosso projeto se encaixaria diante ao mercado, e para termos um ponto focal para trabalharmos, e assim criarmos nossa aplicação para o cliente final ideal.

ProtoPersona

Baseado em pesquisas com alunos da faculdade e pessoas que tem contato com a administração da biblioteca da SpTech chegamos em um paralelo de dois usuários comuns do nosso sistema, o usuário Administrador e o usuário comum/aluno.

Para termos em mente qual era o perfil esperado preferimos separar os dois tipos de usuários, pois cada qual teria os seus próprios seus objetivos, motivações, seus ideais, suas histórias e estilo de vida.

Ter esse perfil do nosso cliente administrador separado do cliente usuário/aluno foi ideal e muito importante para projetarmos e construirmos as próximas fases do nosso projeto, por exemplo, a construção do protótipo das telas do sistema.