**Análise e Desenvolvimento de Sistemas**  
**Projeto Integrador**   
  
  
  


**São Paulo - SP**

**2025**

**Apresentação da equipe**

* Nome dos participantes:
  + João Vitor Araújo de Matos

**Papéis e responsabilidades de cada participante**

João Vitor - Desenvolvedor / Gerente de Projeto

**Apresentação do projeto**

**Descrição do problema**

* Justificativa / Contexto de negócio:

Biblioteca com um sistema de gestão, que apresenta algumas falhas para fazer o gerenciamento de livros e de salas de estudo. É necessário automatizar e melhorar alguns processos, para facilitar a administração.

* Descrição das personas:

**Maria, Atendente da Biblioteca**

Idade: 32 anos

Perfil: Profissional que trabalha na biblioteca atende vários alunos da biblioteca por dia.

Necessidades: Quer ver os livros em estoque de forma mais ágil e simplificada.

Dores: Visualização de livros de forma muito demorada e manual, gestão de livros de forma desorganizada.

Comportamento digital: Usa planilhas, mas precisa de algo mais automatizado.

**Patrícia, atendente da biblioteca**

Idade: 40 anos

Perfil: Responsável pela administração da biblioteca.

Necessidades: Acompanhar as reservas das salas de estudo, organizar equipe.

Dores: Falta de controle das salas de estudo e dificuldade na gestão de livros.

Comportamento digital: Usa planilhas, mas precisa de algo mais automatizado.

**Macro descrição da proposta de solução**

* Principais funcionalidades previstas
  + Gestão de Livros   
    Permite que os funcionários da biblioteca criem e gerenciem os livros de forma organizada.
  + Gestão de Salas de Estudo  
    Controle de registros das salas de estudo feito pelos alunos.
  + Gestão da Equipe

Controle dos profissionais da biblioteca.

* Escopo do projeto

Plataforma web para acesso via internet.

Funcionalidade de gerenciamento de livros.

**Macro cronograma**

**Fase 1 – Planejamento e Levantamento de Requisitos (1 semana)**

* Definição detalhada das funcionalidades e do escopo do sistema.
* Escolha das tecnologias para front-end e back-end.
* Definição da estrutura do banco de dados.

**Fase 2 – Desenvolvimento Inicial (6 semanas)**

* **Semana 1 e 2 – Configuração do ambiente e primeiros módulos**

Back-end: Criação da estrutura inicial do projeto e Banco de Dados.

Front-end: Configuração do projeto e implementação das telas principais.

* **Semana 3 e 4 – Implementação das funcionalidades principais**

Back-end: CRUD de usuarios, Autenticação

Front-end: Desenvolvimento das telas de login, cadastro e agendamento.

* **Semana 5 e 6 – Integração e ajustes**

Back-end: Implementação da gestão das salas de estudo e livros.

Front-end: Integração com o back-end.

**Fase 3 – Testes e Validação (3 semanas)**

Testes de integração.

Revisão da interface e experiência do usuário.

Correção de bugs e otimizações.

**Fase 4 – Implantação e Avaliação Final (3 semanas)**

Testes finais com o professor.

Ajustes e refinamentos com base no feedback.

Documentação final do projeto.

**Detalhamento da Proposta de solução**

Regras de negócio

**Livros**

* Todos os novos livros adicionados pelo funcionário da livraria devem aparecer na listagem de livros
* Um livro não pode ter dois registros e sim aumentar a quantidade do mesmo em estoque.

**Gestão de Equipe**

* Apenas usuários com permissão de administrador ou gerente podem cadastrar e gerenciar a equipe.

**Gestão Sala de Estudo**

* Todas as salas de estudo devem criadas, devem aparecer em um listagem indicando se a mesma está reservada ou não
* Uma sala de estudo não pode ser reservada quando ela já estiver em uso por um aluno.

**Requisitos Funcionais**

**Autenticação e Controle de Acesso**

* O sistema deve permitir o cadastro e login de funcionários.
* Deve haver diferentes níveis de acesso: profissional e gerente/administrador.

**Gestão de Livros**

* O profissional pode realizar o cadastro de novos livros.
* O profissional pode editar um livro.

**Gestão da Equipe**

* O sistema deve permitir o cadastro de profissionais.

**Gestão Salas de Aula**

* Os profissionais podem criar e editar uma sala de estudo.
* Funcionários podem fazer a reserva da sala de estudo.
* O sistema deve indicar quais salas estão disponíveis.

**Requisitos Não Funcionais**

**Desempenho e Escalabilidade**

* O sistema deve suportar múltiplos acessos simultâneos sem perda de desempenho.

**Segurança**

* Apenas alunos do senac podem reserva uma sala de aula.

**Usabilidade e Interface**

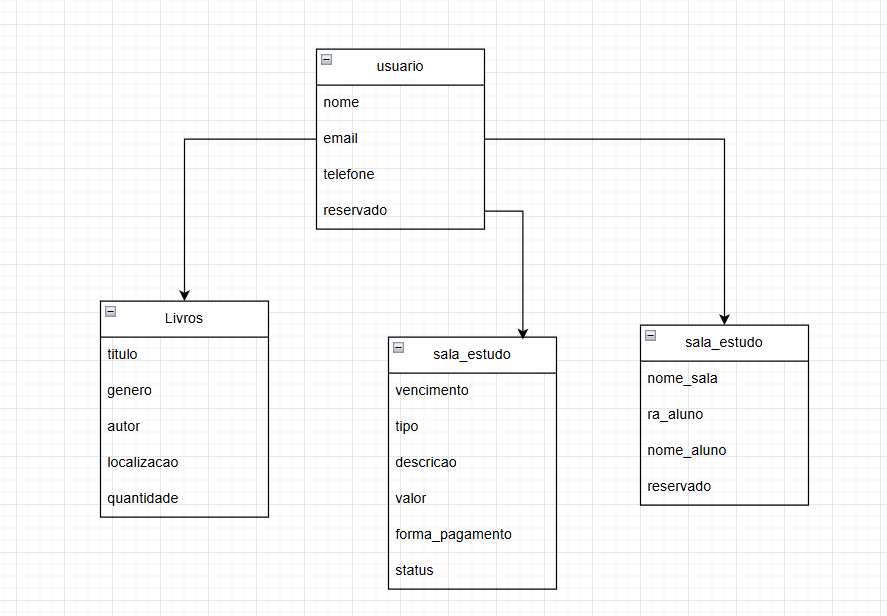
* O sistema deve ser responsivo, funcionando bem em desktop e tablets

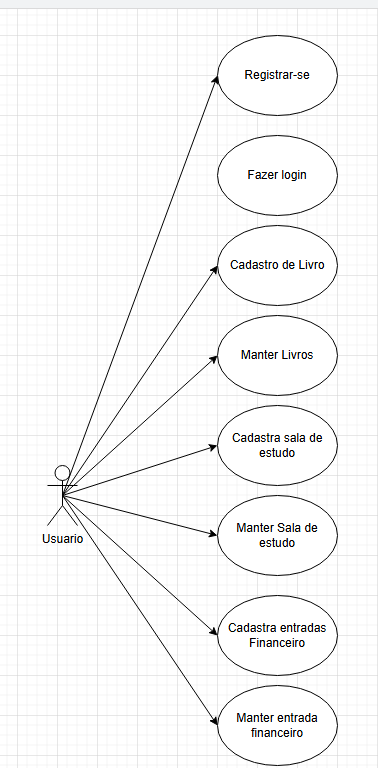
**Manutenção e Evolução**

* O código deve ser bem estruturado para facilitar futuras manutenções.
* A documentação do projeto deve estar atualizada para facilitar melhorias futuras.

**Diagramas**

Diagrama de classes

  
Diagrama de caso de uso



**Modelos**

* Banco de dados
  + MER/DER - enviado separadamente

**Especificação da arquitetura de software**

Arquitetura padrão MVC (Model, View e Controller).

* **Divisão por módulos** → Cada funcionalidade do sistema tem seu próprio módulo (Usuários, Livros, etc.).
* **Camadas organizadas** → Controllers lidam com as requisições, e processam a lógica de negócio.

Ecossistema da solução

* *Refere-se ao conjunto de ferramentas, tecnologias, plataformas e serviços que são utilizados para desenvolver, implementar e operar a solução de software. Isso inclui tanto os componentes internos do sistema quanto as integrações com sistemas externos e serviços de terceiros. O ecossistema da solução abrange todo o ambiente em que o software opera. Neste sentido devemos representar o usuário no ecossistema da solução. O usuário é uma parte fundamental do sistema, e sua interção com a solução deve ser considerando design e na implementação. No ecossistema da solução, o usuário é representado como uma entidade que interage com o sistema atraves de diferentes interfaces e dispositivos.*

**Tipos de Usuários e Acessos**

Usuários → Controlam livros, profissionais, Salas de Estudo.

**Interfaces Disponíveis**

Web App (React.js) → Interface responsiva acessível por desktop e tablet.

API REST (Laravel) → Camada intermediária entre o front-end e o banco de dados, garantindo segurança e performance.

**Definição das tecnologias a serem empregadas na solução**

Linguagem

* Front-end: NextJs (React.js)
* Back-end: Laravel (PHP)

Banco de dados

* MySql

Segurança

* JWT

Repositórios

* Controle de versão: Git
* Plataforma: GitHub (para versionamento e colaboração da equipe)

Provedores em Nuvem

* Railway
* Netlify

Recursos utilizados no projeto

* VS Code: Editor de código principal
* Postman: Testes de API durante o desenvolvimento
* PHPunit: Framework de testes para validação do back-end

**Frameworks de desenvolvimento**

* Back-end: Laravel
* Front-end: Next.JS
* Testes: PHPunit

**Softwares de apoio**

* Postman: Para testar requisição à API
* Testes unitários: PHPunit