Abner Castanho Cardoso - **RA:** 24214899-2

Gabriel Phelipe Bernardo Coronado - **RA:**

Guilherme Tófoli da Silva - **RA:** 24098313-2

Matheus Henrique Coronado Elias - **RA:** 24154884-2

**SISTEMA DE ESTAGIÁRIO**

LONDRINA

2024

Abner Castanho Cardoso - **RA:** 24214899-2

Gabriel Phelipe Bernardo Coronado - **RA:**

Guilherme Tófoli da Silva - **RA:** 24098313-2

Matheus Henrique Coronado Elias - **RA:** 24154884-2

**SISTEMA DE ESTÁGIOS**

Primeira Etapa de Atividade de Estudo Programado, apresentada ao Curso de **Análise e Desenvolvimento de Sistemas** para obtenção parcial de nota semestral.

LONDRINA

2024

**SUMÁRIO**

| INTRODUÇÃO.................................................................................. | 3 |
| --- | --- |
| 1.DESENVOLVIMENTO...................................................................  1.1. Título..........................................................................................  1.1.1. Subtítulo..................................................................................  1.2. Título.......................................................................................... | 6  7  8  9 |
| 2. CONCLUSÃO............................................................................... | 10 |
| REFERÊNCIAS BIBLIOGRÁFICAS................................................. | 12 |
| ANEXOS/ APÊNDICES.................................................................... | 15 |

**INTRODUÇÃO:**

A crescente demanda por profissionais capacitados na área da saúde, aliada à necessidade de vivência prática durante a formação acadêmica, destaca a importância de estágios supervisionados. No entanto, muitos estudantes enfrentam dificuldades para encontrar vagas, acompanhar processos seletivos e manter atualizadas suas documentações junto às instituições responsáveis.

Neste contexto, este projeto propõe o desenvolvimento de um sistema digital de apoio à gestão de estágios voltado para cursos da área da saúde. O sistema tem como objetivo facilitar a comunicação entre estudantes, instituições de ensino e estabelecimentos de saúde, otimizando processos como cadastro, candidatura a vagas e acompanhamento de estágios em andamento.

A justificativa para este projeto baseia-se na carência de soluções simples e eficazes que atendam, ao mesmo tempo, às necessidades dos alunos e das instituições. Um sistema acessível e intuitivo pode contribuir diretamente para a profissionalização dos estudantes, promovendo maior organização e transparência nos estágios obrigatórios.

A metodologia utilizada para esta primeira etapa do projeto foi baseada em levantamento de requisitos, análise de casos de uso, definição de interfaces e aplicação de princípios de usabilidade e acessibilidade para web. Foi também considerada a realidade de muitos estudantes que acessam os sistemas via dispositivos móveis ou redes de internet instáveis, o que reforça a importância de um protótipo de baixa fidelidade funcional e leve.

**1. DESENVOLVIMENTO:**

#### 1.1 Levantamento de Requisitos

Nesta etapa, foram identificadas as principais funcionalidades que o sistema deverá oferecer:

* Cadastro de alunos e instituições de saúde;
* Login para acesso seguro ao sistema;
* Página de exibição de vagas de estágio;
* Acompanhamento do status de cada vaga;
* Interface amigável com foco na usabilidade e responsividade.

Esses requisitos foram baseados em entrevistas informais com estudantes da área da saúde, além da observação de plataformas existentes que possuem funcionalidades limitadas ou pouco intuitivas.

#### 1.1.2 Requisitos Funcionais

| Nome | RF-001 – Cadastro de Universitário |
| --- | --- |
| Descrição | Cadastrar o aluno (universitário), com isso será necessário alguns campos para preencher informações, por exemplo: Nome completo, CPF, RG, Endereço Completo (rua, número, bairro, cidade,estado,cep), Telefone, E-mail, Data de nascimento, Curso, Instituição de Ensino, Semestre e foto caso for necessári**o** |

| Nome | RF-002 – Cadastro das unidades de saúde |
| --- | --- |
| Descrição | Permitir o vínculo dos universitários nas vagas de estágios nas unidades de saúdes cadastradas, então será necessário alguns campos para preencher informações, por exemplo: Nome da Unidade, Endereço Completo (rua, número, bairro, cidade,estado,cep), Telefone, e-mail de contato, website (caso tenha), website (caso tenha) e horário de funcionamento. |

| Nome | RF-003 – Cadastro dos estágios |
| --- | --- |
| Descrição | Permitir o vínculo dos universitários nas vagas de estágios nas unidades de saúdes cadastradas, então será necessário alguns campos para preencher informações, por exemplo: ID do Universitário, Área do estágio, Data de Início do Estágio e Fim do Estágio, ID da Unidade de Saúde, Supervisor responsável, |

#### 1.1.3 Requisitos Funcionais

| Nome | RNF-001- Desempenho |
| --- | --- |
| Categoria | Desempenho |
| Descrição | O sistema deve ser capaz de processar até 12.500 cadastros de universitários e unidades de saúde simultaneamente sem queda no desempenho. |

| Nome | RNF-002- Segurança & Privacidade |
| --- | --- |
| Categoria | Segurança |
| Descrição | Encriptar os dados sensíveis, por exemplo: CPF, RG, Endereço e etc), utilizar criptografias modernas por exemplo AES (Advanced Encryption Standard). |

| Nome | RNF-003 - Redefinição de senha segura |
| --- | --- |
| Categoria | Segurança |
| Descrição | O sistema deve implementar autenticação com senha forte (mínimo de 8 caracteres, com números e letras) e permitir a redefinição de senha via e-mail. |

| Nome | RNF-004 - Logs de auditoria |
| --- | --- |
| Categoria | Segurança |
| Descrição | Deve ser feita uma auditoria de acessos e ações no sistema, com registro de logs que possam ser acessados por administradores para monitoramento. |

#### 1.2 Prototipação e Interface

Foi desenvolvida uma interface web composta por quatro páginas principais:

* **Página Inicial**: apresenta o sistema e permite acesso rápido às vagas ou ao login;
* **Página de Vagas**: exibe as oportunidades disponíveis com botões de candidatura;
* **Login**: permite que o aluno acesse o sistema com suas credenciais;
* **Cadastro**: direcionada a novos usuários, com um formulário simples e direto.

A identidade visual do projeto utilizou cores associadas à área da saúde, como azul e verde, garantindo uma apresentação agradável e coerente com o tema. As interfaces foram desenvolvidas em HTML e CSS puros, com foco em baixo custo computacional e compatibilidade com diferentes dispositivos.

#### 1.3 Justificativa Técnica

A opção por um protótipo de baixa fidelidade nesta fase tem como propósito validar a ideia, testar o fluxo de navegação e realizar ajustes com base em feedbacks preliminares. Evita-se, assim, o desperdício de tempo e recursos em fases posteriores de codificação mais robusta.

Além disso, a estrutura modular do código favorece a evolução futura para tecnologias mais avançadas, como banco de dados, autenticação com segurança, e integração com plataformas governamentais.

#### 2. Conclusão

O desenvolvimento do sistema de estágios na área da saúde cumpre o objetivo inicial de apresentar uma solução prática, acessível e funcional para gerenciar processos relacionados à vivência prática dos estudantes. A proposta do sistema visa não apenas facilitar a vida acadêmica dos usuários, mas também estreitar a relação entre instituições de ensino e locais de estágio.

A metodologia adotada — baseada em levantamento de requisitos, prototipação e aplicação de conceitos de design centrado no usuário — mostrou-se adequada para responder ao problema identificado, permitindo a criação de uma interface coerente com as necessidades reais dos estudantes.

A conclusão desta primeira etapa demonstra que o sistema proposto é viável e apresenta grande potencial de impacto social e educacional, especialmente em regiões onde o acesso a sistemas informatizados ainda é limitado. Os próximos passos envolvem a validação com usuários reais e o planejamento da implementação técnica completa do sistema.

#### 3. Referências Bibliográficas:

ChatGPT

#### 4. Descrição da arquitetura de sistemas a ser utilizada

##### 4.1 Visão Geral: Arquitetura em Camadas + Microsserviços

● **Frontend (Camada de Apresentação):** Aplicação Web responsiva (pode ser desenvolvida com React, Angular ou Vue.js).  
  
Permite acesso aos usuários via navegador: alunos, gestores, RH, instituições.  
  
Comunicação com a API por meio de chamadas RESTful ou GraphQL.

● **Backend (Camada de Lógica de Negócio):**API construída em Node.js, Spring Boot (Java), Django (Python) ou ASP.NET Core (C#).

Dividida em **microsserviços**, cada um responsável por um domínio funcional, por exemplo:

* Autenticação e Autorização
* Gestão de Usuários
* Gestão de Instituições
* Gestão de Estágios
* Notificações (e-mail)

Cada serviço se comunica via API Gateway e utiliza mensagens assíncronas quando necessário (por exemplo, via RabbitMQ ou Kafka).

● **Banco de Dados (Camada de Persistência):**Banco de dados relacional (como PostgreSQL ou MySQL) para garantir integridade e consistência dos dados.  
  
Redis ou MongoDB pode ser usado como apoio para cache ou armazenamento temporário de formulários ou logs.

##### 4.1.2 Serviços de Suporte: Serviço de Notificações: envio automatizado de e-mails via fila. Serviço de Logs e Auditoria: para rastrear ações no sistema. Serviço de Segurança: controle de acesso, criptografia, autenticação (JWT + OAuth2).

##### 4.2 Tecnologias Recomendadas

| **Camada** | **Tecnologia Sugerida** |
| --- | --- |
| Frontend | React.js + Tailwind + Axios |
| Backend (API) | Node.js com Express / NestJS |
| Banco de Dados | PostgreSQL (relacional) |
| Autenticação | JWT + OAuth2 |
| Mensageria (opcional) | RabbitMQ / Kafka |
| Cache (opcional) | Redis |
| Hospedagem | Docker + Kubernetes (ou Docker Compose no MVP) |
| Deploy/Infra | AWS / Azure / Render / Railway |

##### 4.3 Segurança

* Criptografia de senhas com bcrypt.
* Controle de acesso por níveis de permissão.
* Autenticação com tokens JWT.
* HTTPS obrigatório.
* Política de acesso restrita aos dados sensíveis (como CPF, e-mail, documentos).

##### 4.4 Escalabilidade e Manutenção

* Microsserviços permitem escalar partes específicas (por exemplo, o serviço de notificações).
* Isolamento de falhas.
* Facilidade para adicionar novas funcionalidades.
* CI/CD pode ser integrado com GitHub Actions ou GitLab CI.

#### 5. ANEXOS ou APÊNDICES:

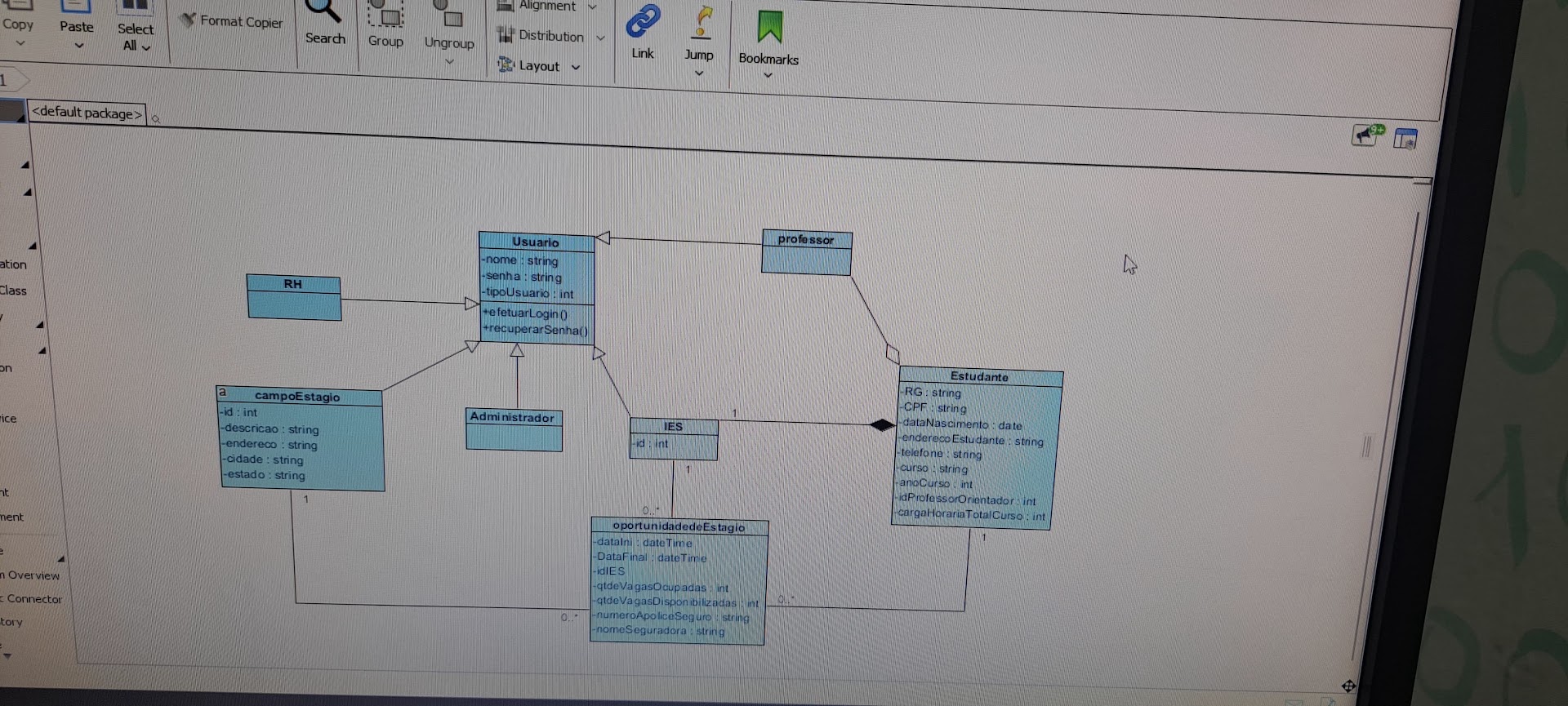
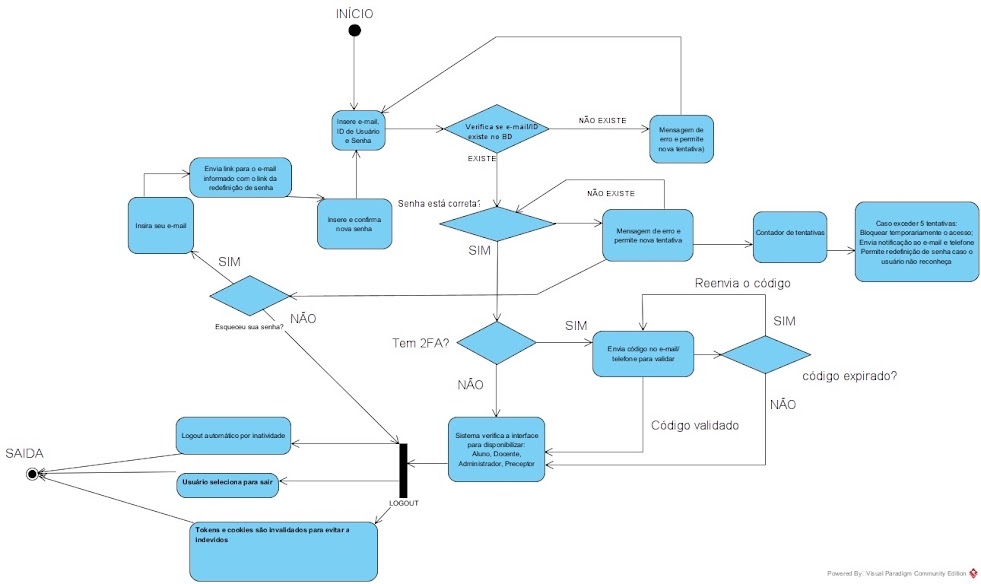
Diagrama de Classes

Diagrama de Atividade

Repositório no GitHub para restantes dos anexos: