



TCCinder - Um sistema de integração e gerenciamento de Trabalho de Conclusão de Curso.

Trabalho de Conclusão de Curso

Amir Santos de Brito

Flávia Maristela Santos Nascimento
Orientadora

Instituto Federal da Bahia – IFBA
Curso de Análise e Desenvolvimento de Sistemas
Campus Salvador

Salvador, Bahia, Brasil
Setembro 2023

SUMÁRIO

1. Visão Geral.....	3
1.1 Declaração do problema.....	3
1.2 Proposta de solução de software.....	4
1.3 Tecnologias adotadas.....	5
1.4 Trabalhos relacionados.....	6
1.4.1 TCC ONLINE: Sistema de gerenciamento de trabalhos de conclusão de curso.....	6
1.4.2 Projeto e Desenvolvimento de um Sistema para Gerenciamento de Trabalhos de Conclusão de Curso.....	7
1.4.3 Asana.....	7
2. Requisitos.....	8
2.1 Requisitos funcionais.....	8
2.2 Requisitos não funcionais.....	8
3. Design.....	9
3.1 Projeto UML.....	9
3.1.1 Diagrama de classe.....	9
3.1.2 Diagrama de Casos de uso.....	10
3.2 Visão Arquitetural.....	29
3.3 Modelagem Banco de Dados.....	30
4 Testes.....	31
5. Implantação.....	34
5.1 Projeto de Implantação.....	34
6. Manual do Usuário.....	35
Agradecimentos.....	39
Referências.....	40

1. Visão Geral

1.1 Declaração do Problema

Conforme [1], o Trabalho de Conclusão de Curso (TCC) é definido como um trabalho escrito, elaborado por formandos, que aborda uma temática específica e, portanto, mostra o conhecimento sobre um assunto. Cada instituição de ensino possui suas regras e normativas que possuem variações e características próprias que definem as premissas básicas do trabalho final, podendo ser realizado em grupo ou individualmente.

O desenvolvimento de um trabalho de conclusão de curso (TCC) é a etapa final e o maior desafio apresentado para um estudante de graduação. Desde a concepção do trabalho até encontrar um orientador envolve planejamento, pesquisa e estudo para organizar informações e produzir um projeto que agregue valor à sociedade. Com o intuito de melhorar a fluidez, as instituições universitárias aproveitaram o avanço da tecnologia e começaram a utilizar sistemas da computação para o gerenciamento, organização e controle do fluxo da informação.

Em [1], apresentam-se alguns exemplos das regras e normativas de trabalho de conclusão de curso de instituições de ensino, sendo elas: a Universidade de São Paulo (USP), Universidade Presbiteriana Mackenzie e Universidade Estadual de Londrina (UEL). O Instituto Federal da Bahia possui semelhanças na estrutura realizada na USP, como: trabalho desenvolvido a partir de uma pesquisa proposta pelo estudante e orientada por um professor do curso. É individual e pode ter a forma de artigo científico, além de possuir disciplinas de acompanhamento relacionadas ao TCC. Na Universidade Presbiteriana Mackenzie, são trabalhos práticos realizados com base na solução de problemas reais enfrentados pela empresa objeto do TCC. A Universidade Estadual de Londrina desenvolve seus trabalhos em grupo de até três estudantes no formato de artigo científico.

O Instituto Federal da Bahia (IFBA), criado em 2008, é uma instituição de excelência na qualidade de ensino, sendo uma referência nacional e por isso tem o compromisso de oferecer educação profissional pública de qualidade. Percebendo a

dificuldade dos estudantes de graduação em encontrar orientadores, ideias e desenvolver seu projeto, identificou-se a necessidade de melhorar a forma como os estudantes e professores da área de tecnologia da informação colaboram e gerenciam seus projetos finais de curso, tornando o processo mais eficiente, rápido e de alta qualidade. Atualmente o Instituto Federal da Bahia (IFBA) não possui um sistema próprio para sua gestão de trabalhos de conclusão de curso.

Diante dessa situação, surgiu a ideia do TCCinder, uma solução tecnológica que permita a apresentação de ideias entre orientandos e orientadores, realize a integração das duas partes e a gestão destes projetos através de uma interface que projete um ambiente de relacionamento permitindo acessibilidade e fluidez aos dados dos trabalhos de conclusão de curso em todo o seu ciclo de vida.

1.2 Proposta de Solução de Software

O software implementado para solucionar o problema proposto tem como objetivo integrar docentes e discentes para fomentar o desenvolvimento de projetos e trabalhos finais de curso através de uma interface web fluida e amigável para sua gestão, garantindo qualidade e velocidade.

Para isso, o sistema dispõe de um ambiente onde tanto os discentes quanto os docentes poderão submeter ideias de propostas com visibilidade controlada conforme seus interesses. Estas propostas de projetos serão publicadas num espaço aberto onde os discentes poderão submeter interesse nos projetos dos professores que mutuamente poderão fazer o mesmo.

No momento que ocorre a confirmação de ambas as partes para uma proposta de projeto, inicia-se o seu gerenciamento onde o discente poderá publicar suas entregas. Tanto o professor da disciplina quanto orientador poderá acompanhar as entregas realizadas pelo estudante portador do projeto, porém o orientador possuirá outras funcionalidades como: adicionar tarefas ligadas às entregas, confirmar, suspender ou recusar as entregas.

Diante deste contexto apresentado, a solução entrega visibilidade e acesso para os usuários envolvidos de forma fácil e prática do projeto em todo o ciclo de vida: concepção da ideia, encontrar um orientador ou orientando, gerenciamento e entregas.

1.3 Tecnologias Adotadas

Front-end:

A interface de usuário foi desenvolvida utilizando as linguagens de programação javascript e typescript, e a biblioteca react.js, já que possuem uma sintaxe amigável e uma baixa curva de aprendizado. Isso pode ser importante em um cenário de desenvolvimento ágil, além de apresentar um desempenho perceptível com o Document Object Model (DOM).

Back-end:

Esta camada foi utilizada para implementar as funcionalidades propriamente ditas, as regras de negócio da aplicação e acesso ao banco de dados pelos usuários. O node.js foi escolhido como plataforma de desenvolvimento de software, uma vez que: (a) possui código aberto e (b) permite criar aplicações web utilizando as linguagens de programação javascript e typescript. Além das características supracitadas, esta tecnologia possui recursos como assincronia, modularização e pacotes, servidor web embutido e mantém o desenvolvedor num mesmo ambiente permitindo reduzir a necessidade de aprender outra tecnologia para implementar a solução. Além do node.js, o framework express.js foi utilizado por ser uma tecnologia flexível e robusta criada a partir do node.js, que fornece recursos para implementar soluções web.

Banco de dados:

A camada de dados foi implementada usando o Postgres, um Sistema Gerenciador de Banco de Dados (SGBD) relacional de código aberto que fornece bom desempenho, escalabilidade, compatibilidade com diversas linguagens de programação, suporte à transações, segurança, suporte aos principais sistemas operacionais e integração com outras tecnologias como servidores Apache.

Tabela 1 - Descrição de Tecnologias		
Nº	Tecnologia	Descrição
1	Javascript	Linguagem de programação
2	Typescript	Linguagem de programação
3	Node.js	Plataforma de desenvolvimento de software
4	Express.js	Framework de aplicações web do Node.js
5	React.js	Biblioteca javascript para Interface de Usuários
6	Postgres	Sistema de gerenciamento de banco de dados

1.4 Trabalhos Relacionados

1.4.1 - TCC ONLINE: Sistema de gerenciamento de trabalhos de conclusão de curso. [2]

Este sistema, desenvolvido por Leonardo Lima Cabral pela Universidade Federal Fluminense, apresenta um projeto de sistema de gerenciamento de trabalhos de conclusão de curso com o objetivo principal de auxiliar na simplificação da interação dos orientandos e seus orientadores, possuindo também foco na comunicação dos envolvidos para agilizar o processo de conclusão. Em contrapartida, o sistema apresenta soluções partindo da premissa que o usuário já possui um projeto definido e um orientador escolhido, o qual não trata um dos principais problemas dos estudantes que possuem dificuldade de encontrar um tema e um orientador adequado ao projeto.

1.4.2 - Projeto e Desenvolvimento de um Sistema para Gerenciamento de Trabalhos de Conclusão de Curso. [3]

Este trabalho, tem como objetivo uma aplicação web destinado ao gerenciamento de trabalhos de conclusão de curso, visa melhorar a comunicação entre alunos e

professores para melhorar na escolha de temas e desenvolvimento dos trabalhos. Em contrapartida, o sistema apresenta soluções partindo da premissa que o usuário já possui um projeto definido e um orientador escolhido, o qual não trata um dos principais problemas dos estudantes que possuem dificuldade de encontrar um tema e um orientador adequado ao projeto.

1.4.3 - Asana. [4]

Asana é um sistema disponível no mercado de software para gerenciamento de projetos no contexto mais amplo sem um escopo definido. Permite que seus usuários visualizem e acompanhem seus projetos, suas tarefas, compartilhe em formato de equipe e controle pelo tempo usando recursos do calendário.

Descrição	Asana	Tcc Online	Projeto Tcc	TCCinder
Se cadastrar no sistema	x	x	x	x
Gerenciar projetos	x	x	x	x
Gerenciar tarefas	x	x		x
Gerenciar permissões	x	x		x
Upload de arquivos	x	x	x	x
Adicionar comentários		x		
Publicar propostas				x
Gerar atas			x	

Conforme Tabela 2, os projetos correlatos com a proposta de solução de software apresentada possuem uma estrutura padrão para gerenciamento de trabalhos com algumas características entre permitir comentários ou atas, porém apenas no TCCinder encontra-se uma estrutura de ambiente para publicação de propostas de

ideias permitindo acessibilidade aos usuários para encontrar seus pares de forma mais ágil e simples.

2. Requisitos

2.1 Requisitos Funcionais

Nº	Descrição
RF01	O sistema deve permitir que o usuário realize seu autocadastro.
RF02	O sistema deve permitir que o usuário realize o login no sistema.
RF03	O sistema deve permitir que o usuário edite seus dados pessoais.
RF04	O sistema deve permitir que o usuário realize a criação, edição, leitura e remoção de projetos.
RF05	O sistema deve permitir que o usuário controle a visibilidade e permita acesso ou remova de seus projetos.
RF06	O sistema deve permitir que o usuário realize a criação, edição, leitura e remoção de tarefas.
RF07	O sistema deve permitir que o usuário submeta interesse em projetos de outros usuários.

2.2 Requisitos Não-Funcionais

Nº	Descrição	Categoria
RNF01	O sistema deve ser protegido contra o acesso não autorizado	Segurança
RNF02	O sistema deve validar e autenticar os usuários antes de permitir acesso.	Segurança
RNF03	O sistema deve ser capaz de lidar com número mínimo de 500 acessos simultâneos sem perda de desempenho.	Desempenho
RNF04	O sistema deve ser intuitivo, possuir uma interface amigável e de simples acesso.	Usabilidade

RNF05	O sistema deve atender os requisitos funcionais do usuário.	Confiabilidade
-------	---	----------------

3. Design

3.1 Projeto UML

3.1.1 Diagramas de Classe

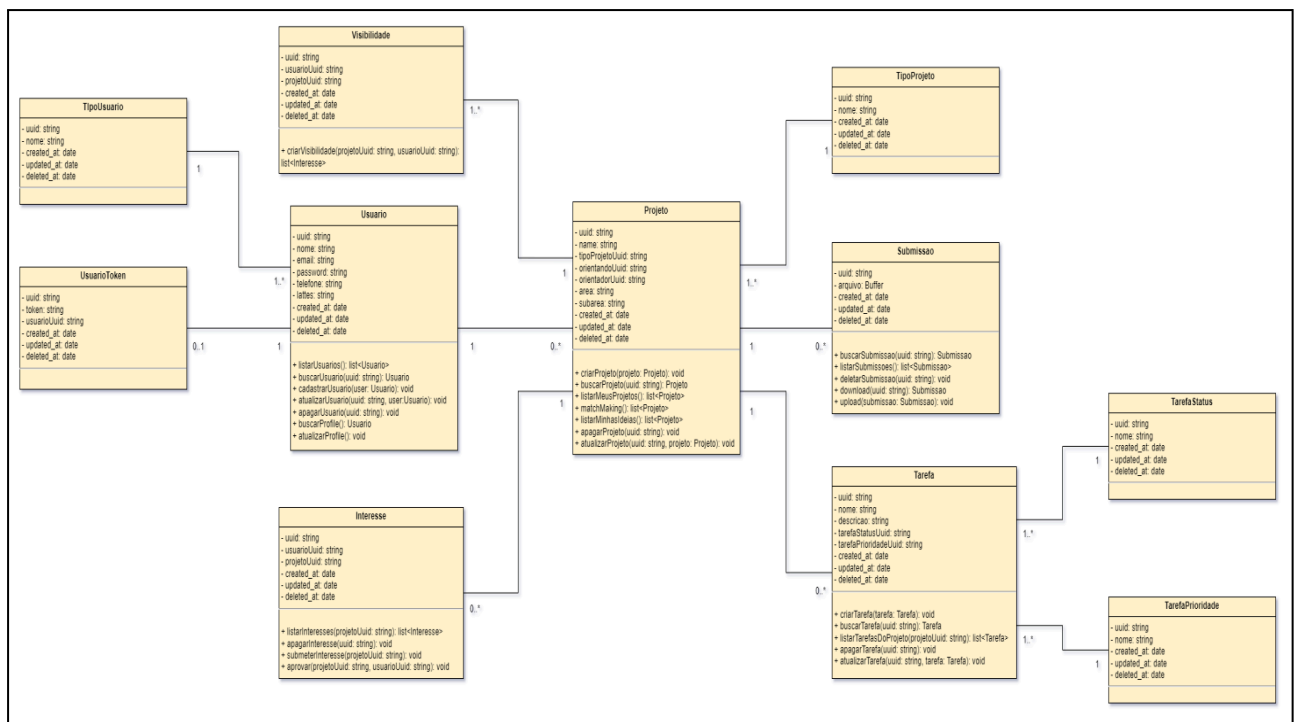


Figura 1 - Diagramas de classe

3.1.2 Diagrama de Casos de Uso

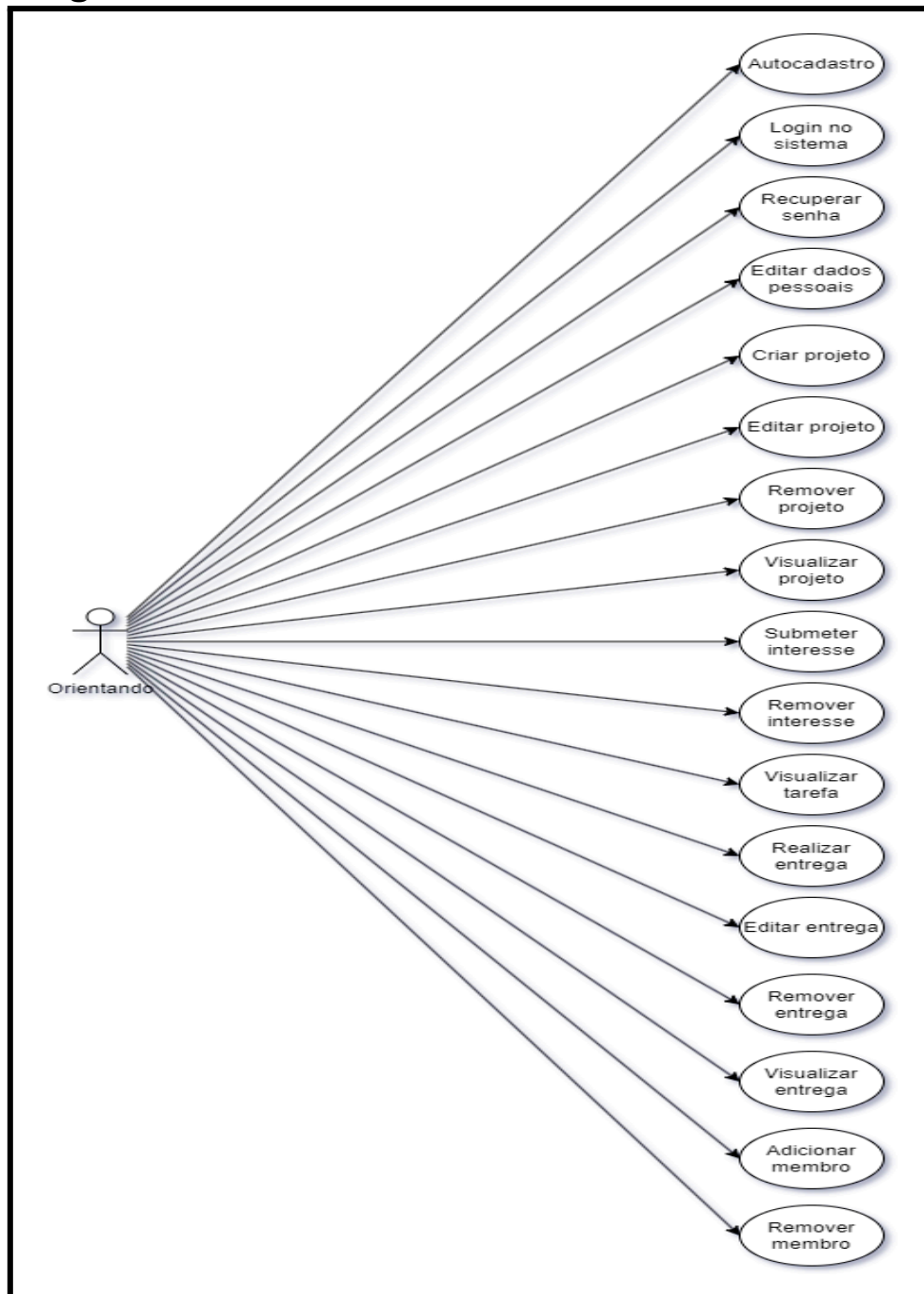


Figura 2 - Diagrama de caso de uso do orientando.

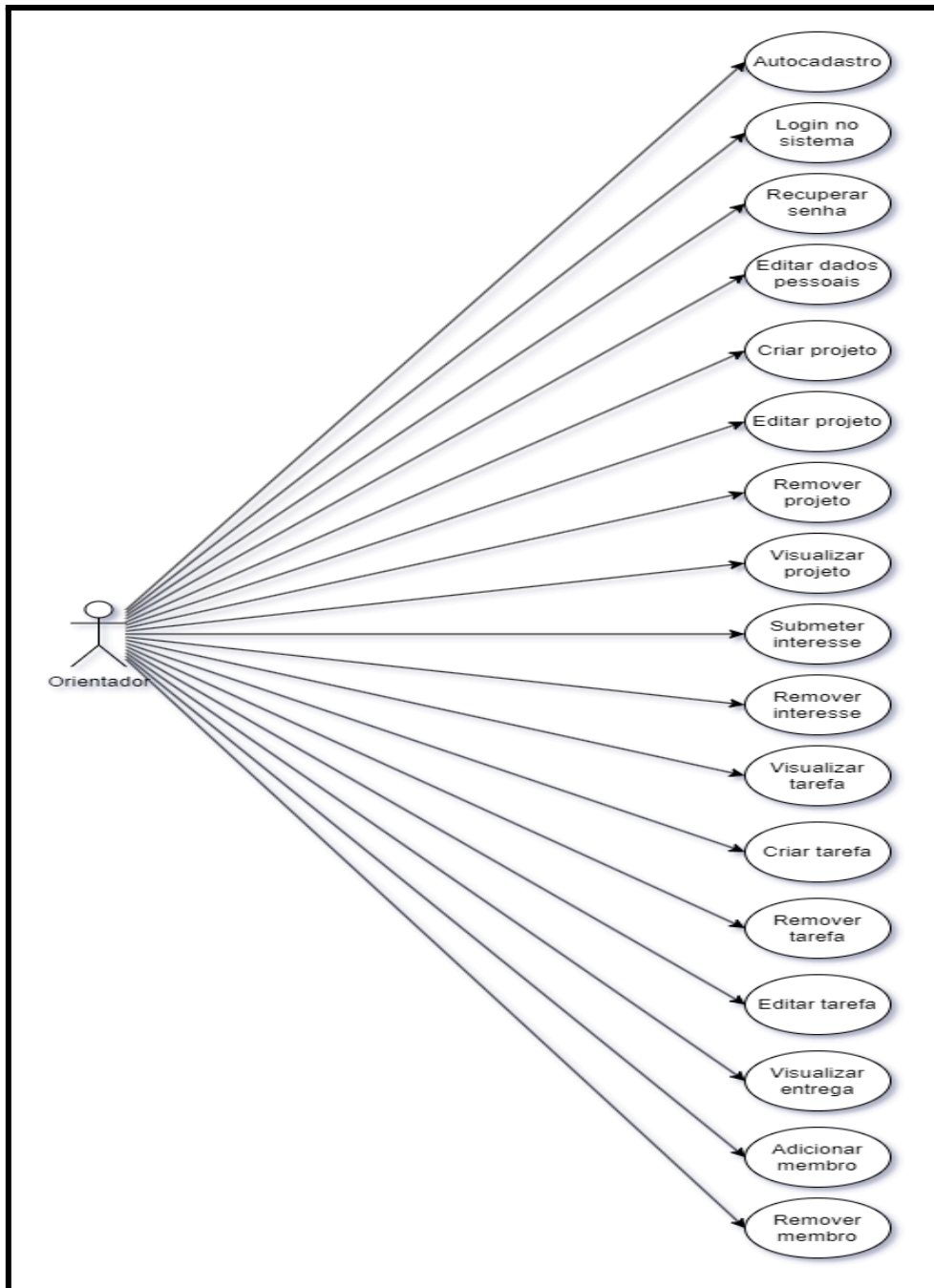


Figura 3 - Diagrama de caso de uso do orientador.

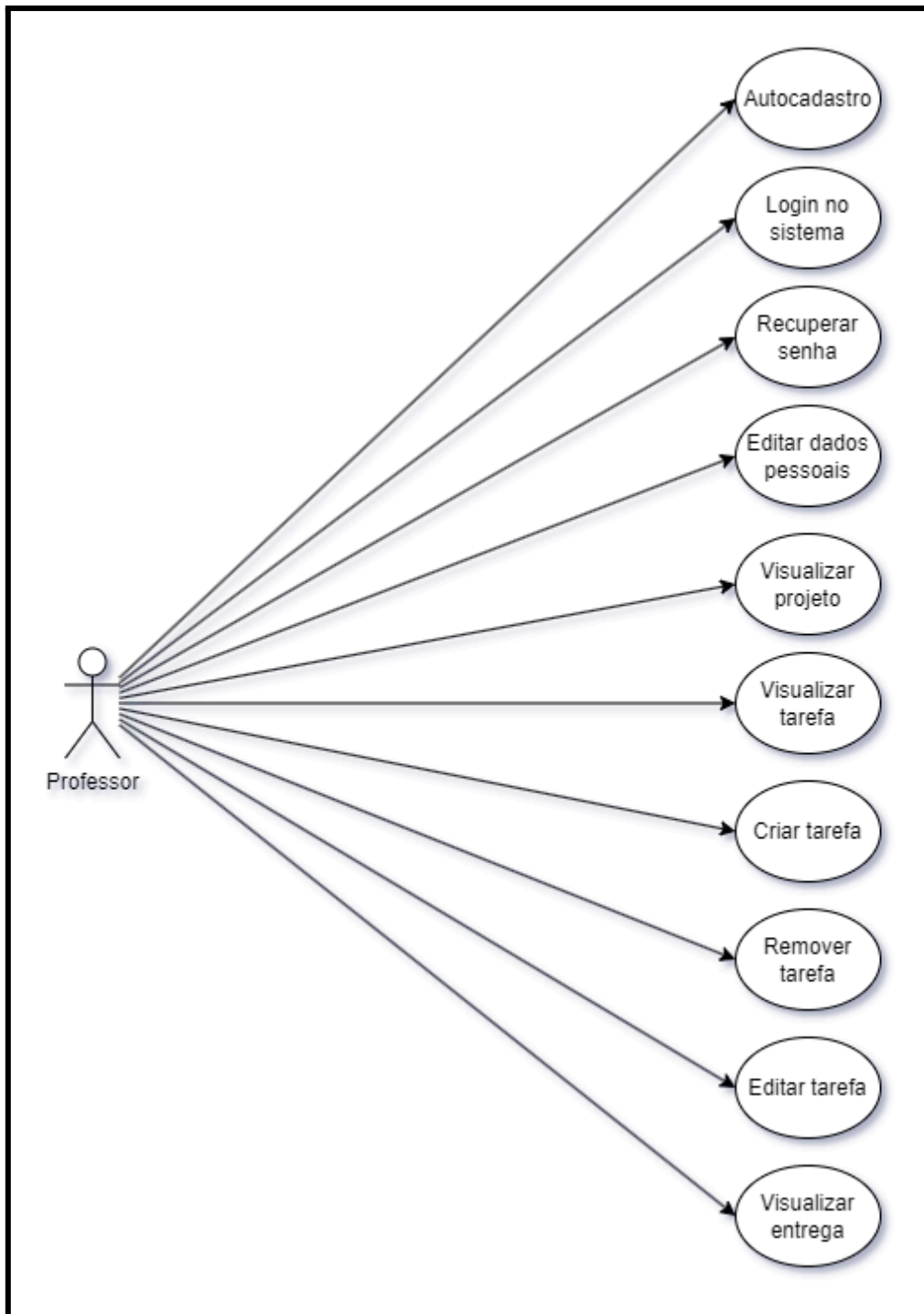


Figura 4 - Diagrama de caso de uso do professor

Com base nos diagramas de caso de uso apresentados nas figuras 2, 3 e 4, apresenta-se uma tabela com cada permissão de usuário:

Nº	Casos de uso	Orientando	Orientador	Professor
01	Autocadastro	sim	sim	sim
02	Login no sistema	sim	sim	sim
03	Recuperar senha	sim	sim	sim
04	Editar dados pessoais	sim	sim	sim
05	Criar projeto	sim	sim	não
06	Editar projeto	sim	sim	não
07	Remover projeto	sim	sim	não
08	Visualizar projeto	sim	sim	sim
09	Submeter interesse	sim	sim	não
10	Remover interesse	sim	sim	não
11	Criar tarefa	não	sim	sim
12	Remover tarefa	não	sim	sim
13	Editar tarefa	não	sim	sim
14	Visualizar tarefa	sim	sim	sim
15	Realizar entrega	sim	não	não
16	Editar entrega	sim	não	não
17	Remover entrega	sim	não	não
18	Visualizar entrega	sim	sim	sim
19	Adicionar membro no projeto	sim	sim	não
20	Remover membro no projeto	sim	sim	não

3.1.2.1 - Caso de uso: Cadastro

Caso de uso	UC01 - Cadastro
Ator	Todos

Resumo	Descreve as etapas para o usuário se auto cadastrar no sistema
Pré-condição	
Pós-condição	
Cenário Principal - Sucesso	
Ações do ator	Ações do sistema
Selecione a opção cadastre-se	
	Exibe o formulário de cadastro
Inserir os dados e submete o formulário	
	Valida os dados submetidos
	Registra o usuário

3.1.2.2 - Caso de uso: Login no sistema

Caso de uso	UC02 - Login no sistema
Ator	Todos
Resumo	Descreve as etapas para o usuário acessar o sistema
Pré-condição	Estar cadastrado
Pós-condição	
Cenário Principal - Sucesso	
Ações do ator	Ações do sistema
Selecione a opção login	
	Exibe o formulário de login
Inserir os dados e submete o formulário	
	Valida os dados submetidos
	Autentica o usuário
Cenário Alternativo - Usuário não cadastrado	
Ações do ator	Ações do sistema

	Caso o usuário não esteja cadastrado realizar o UC01 - Autocadastro
Cenário Alternativo - Dados inválidos ou incorretos	
Ações do ator	Ações do sistema
	Realizar o UC03 - Recuperar senha

3.1.2.3 - Caso de uso: Recuperar senha

Caso de uso	UC03 - Recuperar senha
Ator	Todos
Resumo	Descreve as etapas para o usuário recuperar o acesso
Pré-condição	
Pós-condição	
Cenário Principal - Sucesso	
Ações do ator	Ações do sistema
Selecione a opção Esqueci minha senha	
	Exibe o formulário solicitando e-mail cadastrado
Insere o e-mail e submete o formulário	
	Envia um e-mail para o usuário com link de acesso para recuperar a senha.
Acessa o email e clica no link	
	Exibe formulário de recuperação de senha.
	Solicita uma nova senha
Insere os novos dados e submete	
	Valida os dados
	Atualiza o cadastro

	Redireciona para a tela de login
--	----------------------------------

3.1.2.4 - Caso de uso: Editar dados pessoais

Caso de uso	UC04 - Editar dados pessoais
Ator	Todos
Resumo	Descreve as etapas para editar os dados pessoais
Pré-condição	Estar logado no sistema e na página home
Pós-condição	
Cenário Principal - Sucesso	
Ações do ator	Ações do sistema
Clicar na opção Meus dados	
	Exibe a tela de Dados pessoais
Clica na opção Atualizar dados	
	Exibe formulário para atualizar os dados.
Inserir os dados e submete o formulário	
	Valida os dados
	Atualiza os dados do usuário
	Retorna o usuário para a página de Dados Pessoais.

3.1.2.5 - Caso de uso: Criar projeto

Caso de uso	UC05 - Criar projeto
Ator	Orientando e Orientador
Resumo	Descreve as etapas para criar um projeto.
Pré-condição	Estar logado no sistema e na página home

Pós-condição	
Cenário Principal - Sucesso	
Ações do ator	Ações do sistema
Clicar na opção Projetos	
	Exibe a tela de Projetos
Clica na opção Criar projeto	
	Exibe formulário para criar o projeto.
Inserir os dados e submete o formulário	
	Valida os dados
	Registra o projeto
	Retorna o usuário para a página de Projetos
	Exibe os projetos com visibilidade para este usuário

3.1.2.6 - Caso de uso: Editar projeto

Caso de uso	UC06 - Editar projeto
Ator	Orientando e Orientador
Resumo	Descreve as etapas para editar um projeto.
Pré-condição	Estar logado no sistema e na página home
Pós-condição	
Cenário Principal - Sucesso	
Ações do ator	Ações do sistema
Clicar na opção Projetos	
	Exibe a tela de Projetos com projetos visíveis para este usuário.
Seleciona um projeto	

	Exibe a opção de Editar Projeto
Clica na opção Editar projeto	
	Exibe formulário para atualizar os dados do projeto.
Inserir os dados e submete formulário	
	Valida os dados
	Atualiza o Projeto
	Retorna o usuário para a página de Projetos.

3.1.2.7 - Caso de uso: Remover Projeto

Caso de uso	UC07 - Remover projeto
Ator	Orientando e Orientador
Resumo	Descreve as etapas para remover um projeto.
Pré-condição	Estar logado no sistema e na página home
Pós-condição	
Cenário Principal - Sucesso	
Ações do ator	Ações do sistema
Clicar na opção Projetos	
	Exibe a tela de Projetos com projetos visíveis para este usuário.
Seleciona um projeto.	
	Exibe a opção de remover o projeto.
Clica na opção de remover	
	Exibe uma tela solicitando confirmação para remover o projeto.
Clica na opção de sim e submete	
	Remove o projeto.

	Retorna o usuário para a tela de Projetos.
--	--

3.1.2.8 - Caso de uso: Visualizar projeto

Caso de uso	UC08 - Visualizar projeto
Ator	Todos
Resumo	Descreve as etapas para visualizar um projeto.
Pré-condição	Estar logado no sistema e na página home
Pós-condição	
Cenário Principal - Sucesso	
Ações do ator	Ações do sistema
Clicar na opção Projetos	
	Exibe a tela de Projetos com projetos visíveis para este usuário.
Seleciona um projeto.	
	Exibe uma tela com os dados do projeto.

3.1.2.9 - Caso de uso: Submeter interesse

Caso de uso	UC09 - Submeter interesse
Ator	Orientando e Orientador
Resumo	Descreve as etapas para submeter interesse em um projeto
Pré-condição	Estar logado no sistema e na página home
Pós-condição	
Cenário Principal - Sucesso	

Ações do ator	Ações do sistema
Clicar na opção Projetos	
	Exibe a tela de Projetos com projetos visíveis para este usuário.
Seleciona um projeto.	
	Exibe a opção Submeter interesse
Clica na opção Submeter interesse	
	Registra o interesse no projeto
	Retorna o usuário a página Projetos

3.1.2.10 - Caso de uso: Remover interesse

Caso de uso	UC10 - Remover interesse
Ator	Orientando e Orientador
Resumo	Descreve as etapas para remover interesse em um projeto
Pré-condição	Estar logado no sistema e na página home
Pós-condição	
Cenário Principal - Sucesso	
Ações do ator	Ações do sistema
Clicar na opção Projetos	
	Exibe a tela de Projetos com projetos visíveis para este usuário.
Seleciona um projeto marcado com interesse.	
	Exibe a opção Remover interesse
Clica na opção Remover interesse	
	Registra o interesse no projeto
	Retorna o usuário a página Projetos

3.1.2.11 - Caso de uso: Criar tarefa

Caso de uso	UC11 - Criar tarefa
Ator	Orientador
Resumo	Descreve as etapas para criar uma tarefa para um projeto
Pré-condição	Estar logado no sistema e na página home
Pós-condição	
Cenário Principal - Sucesso	
Ações do ator	Ações do sistema
Clicar na opção Projetos	
	Exibe a tela de Projetos com projetos visíveis para este usuário.
Seleciona um projeto	
	Exibe a opção Adicionar tarefa
Clica na opção Adicionar tarefa	
	Exibe formulário
Insere os dados e submete o formulário	
	Valida os dados
	Registra a tarefa
	Retorna o usuário para a página de Projetos.

3.1.2.12 - Caso de uso: Remover tarefa

Caso de uso	UC12 - Remover tarefa
Ator	Orientador
Resumo	Descreve as etapas para remover uma tarefa para um projeto
Pré-condição	Estar logado no sistema e na página

	home
Pós-condição	
Cenário Principal - Sucesso	
Ações do ator	Ações do sistema
Clicar na opção Projetos	
	Exibe a tela de Projetos com projetos visíveis para este usuário.
Seleciona um projeto	
	Exibe a lista com as tarefas
Clica em uma tarefa	
	Exibe opção de remover tarefa
Clica na opção Remover tarefa	
	Registra a remoção
	Retorna o usuário para a página de Projetos.

3.1.2.13 - Caso de uso: Editar tarefa

Caso de uso	UC13 - Editar tarefa
Ator	Orientador
Resumo	Descreve as etapas para editar uma tarefa para um projeto
Pré-condição	Estar logado no sistema e na página home
Pós-condição	
Cenário Principal - Sucesso	
Ações do ator	Ações do sistema
Clicar na opção Projetos	
	Exibe a tela de Projetos com projetos visíveis para este usuário.

Seleciona um projeto	
	Exibe a lista com as tarefas
Clica em uma tarefa	
	Exibe opção de editar tarefa
Clica na opção editar tarefa	
	Exibe o formulário para editar
Inserir os dados e submete o formulário	
	Valida os dados
	Registra a edição
	Retorna o usuário para a página de Projetos.

3.1.2.14 - Caso de uso: Visualizar tarefa

Caso de uso	UC14 - Visualizar tarefa
Ator	Orientando e Orientador
Resumo	Descreve as etapas para visualizar uma tarefa para um projeto
Pré-condição	Estar logado no sistema e na página home
Pós-condição	
Cenário Principal - Sucesso	
Ações do ator	Ações do sistema
Clicar na opção Projetos	
	Exibe a tela de Projetos com projetos visíveis para este usuário.
Seleciona um projeto	
	Exibe a lista com as tarefas
Clica em uma tarefa	
	Exibe uma tela com a tarefa

3.1.2.15 - Caso de uso: Realizar entrega

Caso de uso	UC15 - Realizar entrega
Ator	Orientando
Resumo	Descreve as etapas para realizar uma entrega
Pré-condição	Estar logado no sistema e na página home
Pós-condição	
Cenário Principal - Sucesso	
Ações do ator	Ações do sistema
Clicar na opção Projetos	
	Exibe a tela de Projetos com projetos visíveis para este usuário.
Seleciona um projeto	
	Exibe a opção de realizar entrega
Clica na opção realizar entrega	
	Exibe formulário
Insere os dados no formulário	
Faz o upload do arquivo no formato .pdf	
	Valida os dados
	Registra a entrega
	Retorna o usuário a página de Projetos.

3.1.2.16 - Caso de uso: Editar entrega

Caso de uso	UC16 - Editar entrega
Ator	Orientando
Resumo	Descreve as etapas para editar uma entrega

Pré-condição	Estar logado no sistema e na página home
Pós-condição	
Cenário Principal - Sucesso	
Ações do ator	Ações do sistema
Clicar na opção Projetos	
	Exibe a tela de Projetos com projetos visíveis para este usuário.
Seleciona um projeto	
	Exibe a opção de realizar entrega
Clica na opção realizar entrega	
	Exibe formulário
Insere os dados no formulário	
Faz o upload do arquivo no formato .pdf	
	Valida os dados
	Registra a entrega
	Retorna o usuário a página de Projetos.

3.1.2.17 - Caso de uso: Remover entrega

Caso de uso	UC17 - Remover entrega
Ator	Orientando
Resumo	Descreve as etapas para remover uma entrega
Pré-condição	Estar logado no sistema e na página home
Pós-condição	
Cenário Principal - Sucesso	
Ações do ator	Ações do sistema
Clicar na opção Projetos	

	Exibe a tela de Projetos com projetos visíveis para este usuário.
Seleciona um projeto	
	Exibe a opção de remover entrega
Clica na opção realizar entrega	
	Registra a remoção
	Retorna o usuário a página de Projetos.

3.1.2.18 - Caso de uso: Visualizar entrega

Caso de uso	UC18 - Visualizar entrega
Ator	Todos
Resumo	Descreve as etapas para visualizar uma entrega
Pré-condição	Estar logado no sistema e na página home
Pós-condição	
Cenário Principal - Sucesso	
Ações do ator	Ações do sistema
Clicar na opção Projetos	
	Exibe a tela de Projetos com projetos visíveis para este usuário.
Seleciona um projeto	
	Exibe a opção de visualizar entrega
Clica na opção visualizar entrega	
	Exibe uma tela com as informações da entrega e o botão de download do arquivo.pdf
Clica em download arquivo	
	Envia o arquivo para o usuário

3.1.2.19 - Caso de uso: Adicionar membro no projeto

Caso de uso	UC19 - Adicionar membro no projeto
Ator	Orientando e Orientador
Resumo	Descreve as etapas para adicionar um membro no projeto.
Pré-condição	Estar logado no sistema e na página home
Pós-condição	
Cenário Principal - Sucesso	
Ações do ator	Ações do sistema
Clicar na opção Projetos	
	Exibe a tela de Projetos com projetos visíveis para este usuário.
Seleciona um projeto	
	Exibe a opção de adicionar membro
Clica na opção adicionar membro	
	Exibe um formulário para adicionar membro.
Insere o e-mail do novo membro e submete formulário	
	Valida os dados
	Registra o novo membro
	Retorna o usuário a página de projetos.

3.1.2.20 - Caso de uso: Remover membro no projeto

Caso de uso	UC20 - Remover membro no projeto
Ator	Orientando e Orientador
Resumo	Descreve as etapas para remover um membro no projeto.

Pré-condição	Estar logado no sistema e na página home
Pós-condição	
Cenário Principal - Sucesso	
Ações do ator	Ações do sistema
Clicar na opção Projetos	
	Exibe a tela de Projetos com projetos visíveis para este usuário.
Seleciona um projeto	
	Exibe a lista com membros do projeto
Clica no respectivo membro	
	Exibe a opção remover membro.
Clica na opção remover membro	
	Registra a remoção
	Retorna o usuário a página de projetos.

3.2 Visão Arquitetural

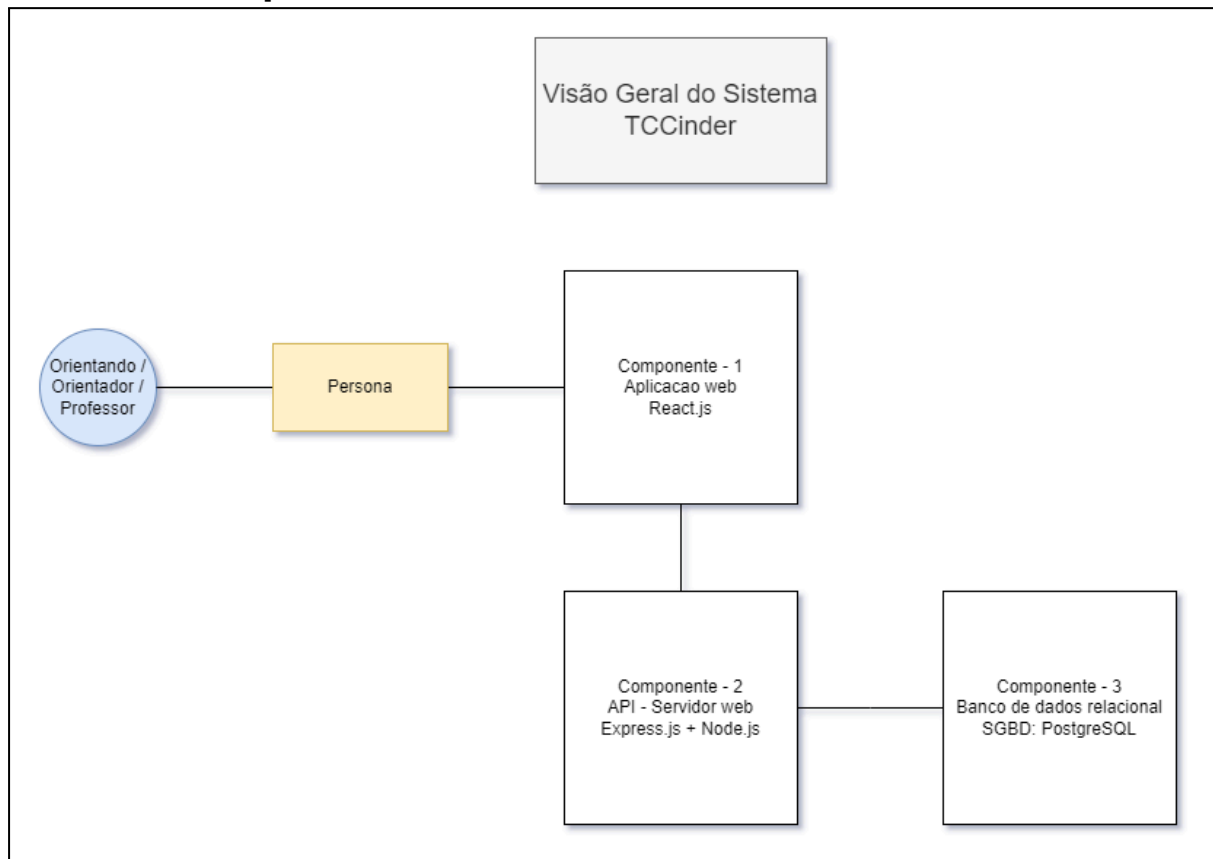


Figura 5 - Visão geral da arquitetura baseada em componentes

A figura 5 oferece uma visão geral do sistema, destacando os componentes principais da aplicação e os fluxos de interação com o usuário. Essa visão permite uma compreensão clara das interações fundamentais e do funcionamento geral da aplicação.

No Componente 1, a aplicação interage diretamente com o usuário através do protocolo HTTP, apresentando interfaces e componentes visuais de maneira amigável e responsiva. Este componente é responsável por renderizar a interface do usuário, proporcionando uma experiência de uso fluida e intuitiva. Para alcançar isso, são utilizadas bibliotecas como React.js, Redux, e Sass, que facilitam a construção de interfaces dinâmicas e gerenciadas de forma eficiente. Além disso, o Componente 1 é encarregado de capturar e enviar os dados do usuário para o Componente 2.

O Componente 2 consiste em uma API REST que funciona como o servidor web, sendo responsável pelo processamento e tratamento dos dados recebidos do Componente 1. Nesta fase, os dados são validados, transformados e processados de acordo com as regras de negócio definidas para a aplicação. Este componente desempenha um papel crucial no fluxo de dados, garantindo que as informações recebidas sejam tratadas de forma adequada antes de serem encaminhadas.

Após o processamento, o Componente 2 se comunica com o Componente 3, que é o sistema de gerenciamento de banco de dados. Este componente é responsável pela persistência dos dados, assegurando sua integridade e autenticidade. A confiabilidade do armazenamento de dados é garantida por meio deste sistema, que mantém os dados de forma segura e acessível para futuras operações.

3.3 Modelo de Banco de Dados

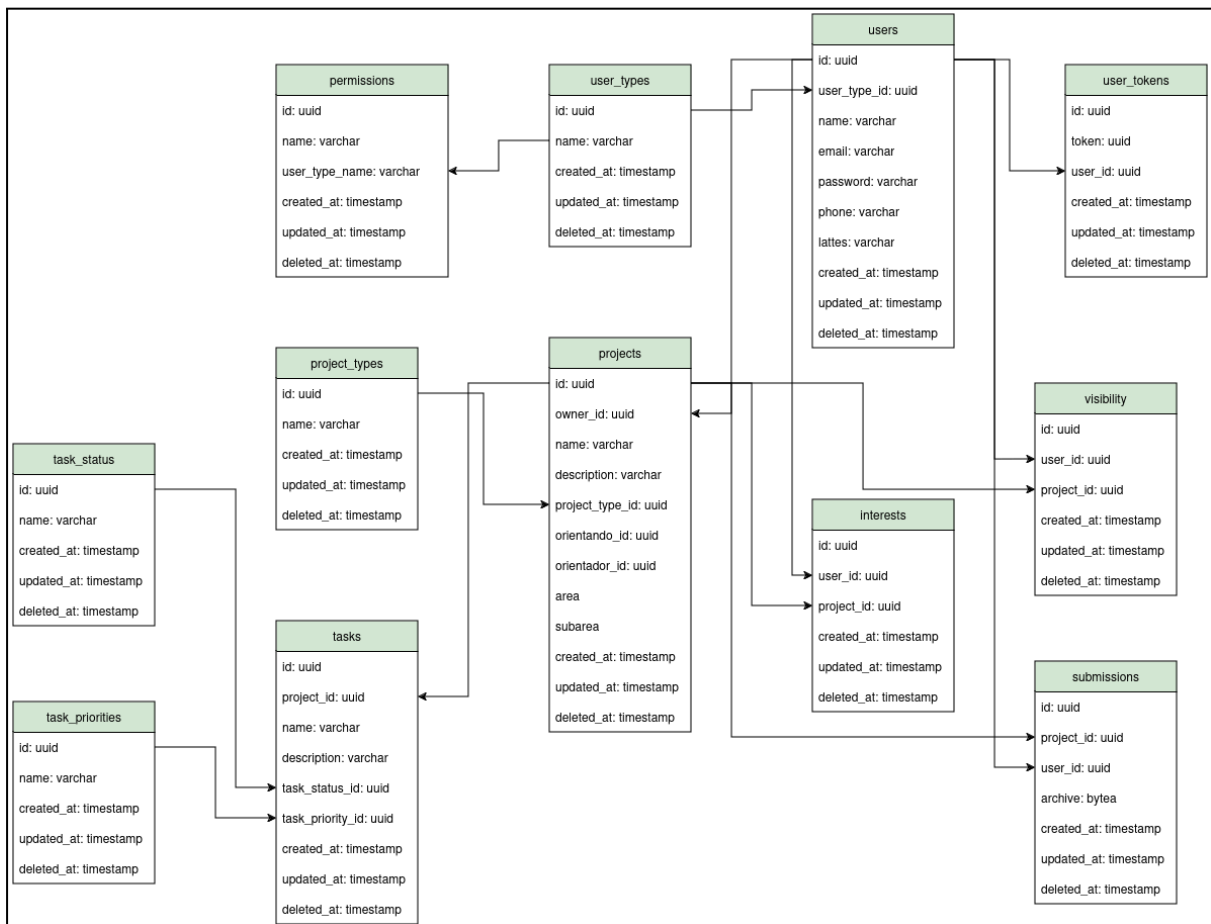


Figura 6 - Modelo lógico banco de dados

A figura 6 ilustra as entidades e seus respectivos relacionamentos no banco de dados da aplicação. As entidades principais incluem: usuários, projetos, tarefas, e entregas. Nesta modelagem:

Projetos estão vinculados a um proprietário, que é um usuário, além de outros usuários que participam como membros. Esse relacionamento é de um para muitos (1-n), onde um usuário pode ser proprietário de vários projetos e um projeto pode ter múltiplos membros.

Cada tarefa e entrega estão associados a um projeto, formando também um relacionamento de um para muitos (1-n). Isso significa que um único projeto pode conter várias tarefas e várias submissões, reforçando a hierarquia e a dependência entre essas entidades.

As tabelas que terminam com o sufixo `_tipo` não são tratadas como entidades independentes, mas sim como tabelas auxiliares. Elas servem para facilitar o gerenciamento e manutenção dos registros no banco de dados, fornecendo um contexto adicional na modelagem principal.

As tabelas `permissões` e `usuário_tokens` também não são classificadas como entidades. Elas são usadas para o gerenciamento de permissões dentro do sistema e para oferecer serviços como a geração de tokens de acesso e a recuperação de senhas. Essas tabelas desempenham um papel na segurança e na administração do sistema.

As tabelas `visibilidade` e `interesse` são utilizadas para controlar o acesso e a visibilidade dos projetos dos usuários. Elas estabelecem um relacionamento de um para muitos (1-n) com projeto, permitindo um controle sobre quais usuários têm acesso a quais projetos.

4. Testes

4.1 Projeto de testes

Para garantir a qualidade do software, foram adotadas as seguintes estratégias de teste:

1. Testes de API

- **Objetivo:** Garantir que a API REST forneça dados corretos e processem as requisições de forma eficiente, mantendo a integridade das regras de negócio.
- **Cenário:** A API REST serve como intermediária entre a interface do usuário e o banco de dados. Testes devem verificar se as rotas para manipulação de projetos, submissões e tarefas respondem de acordo com os requisitos funcionais, validando os cenários de sucesso e falha.
- **Execução:**

Criação de Usuários e Login: Usar ferramentas como Postman para testar a rota de criação de novos usuários e a autenticação, garantindo que um token de autenticação válido é retornado para logins corretos e erros apropriados são tratados para tentativas de login inválidas.

Gerenciamento de Projetos: Enviar requisições para criação, listagem e remoção de projetos, assegurando que um usuário autenticado só possa acessar e gerenciar seus próprios projetos. Validar a consistência do fluxo CRUD (Create, Read, Update, Delete).

Submissões: Testar a API de submissão de entregas e o gerenciamento de submissões, verificando se os dados são corretamente persistidos e retornados.

Respostas de Erro: Verificar se erros são retornados corretamente em cenários de dados inválidos, permissões inadequadas ou falhas de autenticação.

2. Testes Funcionais

- **Objetivo:** Verificar se o sistema cumpre os requisitos funcionais especificados e se o comportamento é o esperado para o usuário final.
- **Cenário:** Cobrir os principais fluxos do sistema, como cadastro de usuários, submissão de interesse em projetos, gerenciamento de tarefas e permissões de acesso.
- **Execução:**

Cadastro e Login de Usuários: Validar se o processo de cadastro de novos usuários (aluno, orientador) funciona corretamente, assegurando que as validações de e-mail, CPF e senha são aplicadas. Testar o login com credenciais corretas e incorretas.

Submissão de Interesse em Projetos: Testar o fluxo de submissão de interesse por alunos em projetos disponíveis, verificando se as submissões são corretamente vinculadas ao aluno e se as permissões de orientadores estão sendo respeitadas.

Criação e Gerenciamento de Tarefas: Testar se o orientador pode criar, atualizar e excluir tarefas em seus projetos. Verificar se as tarefas aparecem corretamente para os alunos, conforme as permissões configuradas.

Visibilidade de Projetos: Certificar que os usuários só conseguem visualizar e editar os projetos para os quais possuem permissões explícitas, testando diferentes níveis de acesso para orientadores e alunos.

3. Testes de Interface de Usuário (UI)

- **Objetivo:** Avaliar a usabilidade, acessibilidade e a experiência do usuário ao interagir com a interface.

- Cenário: O sistema web deve apresentar uma interface clara e amigável, implementada com React.js, Redux e Sass. A interface deve ser responsiva e intuitiva, funcionando corretamente em diferentes dispositivos e navegadores.
- Execução:

Verificação de Layout e Responsividade: Usar ferramentas como Chrome DevTools para testar a responsividade do layout em diferentes tamanhos de tela, desde dispositivos móveis até desktops. Verificar se a interface ajusta os componentes e mantém a usabilidade em diferentes resoluções.

Interatividade e Navegação: Testar todos os componentes interativos, como botões, menus e formulários, garantindo que a navegação seja fluida e intuitiva. Verificar se feedbacks visuais (por exemplo, mensagens de sucesso ou erro) aparecem corretamente ao realizar ações.

4. Testes de Performance

- Objetivo: Garantir que o sistema mantenha uma boa performance sob diferentes condições de carga, proporcionando uma experiência fluida para um grande número de usuários simultâneos.
- Cenário: O sistema será acessado por múltiplos alunos, orientadores e professores ao mesmo tempo, especialmente em períodos críticos, como submissões de TCCs. Os testes de performance devem garantir que o sistema se mantenha estável e responda em tempos aceitáveis sob diferentes cargas.
- Execução:

Teste de Carga: Usar ferramentas como JMeter ou Locust para simular centenas de usuários acessando o sistema simultaneamente, submetendo projetos, tarefas e entregas. Verificar se o tempo de resposta da API e da interface permanece dentro dos limites aceitáveis.

Teste de Stress: Aumentar gradativamente a quantidade de usuários e requisições até que o sistema atinja seu ponto de falha. Registrar quando e como o sistema começa a apresentar degradação no desempenho.

Escalabilidade: Avaliar a capacidade do sistema de escalar horizontalmente ou verticalmente para suportar mais usuários, verificando se o sistema se recupera corretamente após atingir o limite de capacidade.

5. Implantação

5.1 Projeto de Implantação

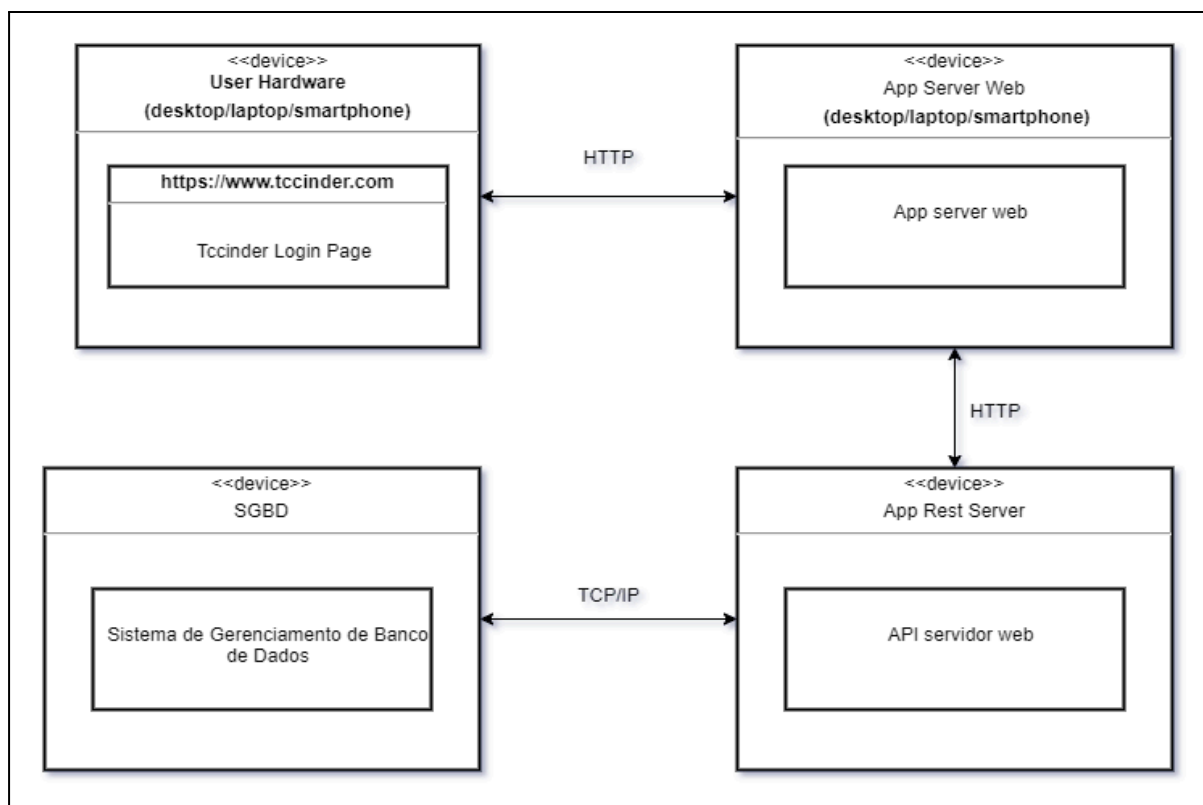


Figura 7 - Diagrama de implantação

A aplicação foi desenvolvida para ser acessada por um navegador browser prioritariamente por computadores desktop ou laptop, contudo possui flexibilidade para ser utilizado também em smartphones.

O Fluxo apresentado pela figura 7 aborda o contexto das possíveis interações que um usuário pode ter no sistema, sendo elas: autocadastro, gerenciamento do perfil, gerenciamento de projetos, tarefas e entregas, publicar e buscar projetos.

6. Manual do Usuário

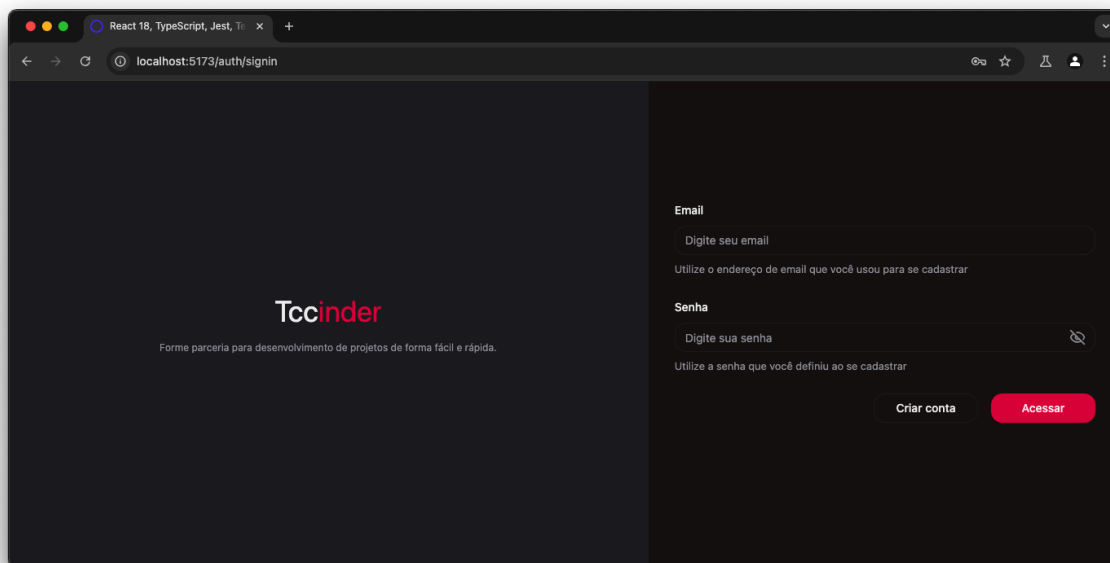


Figura 8 - Tela de Signin

Como primeiro passo para utilizar o sistema, deve-se realizar o login inserindo seu e-mail de cadastro e senha. Caso não possua um cadastro realizado, deve-se seguir a opção de criar uma conta.

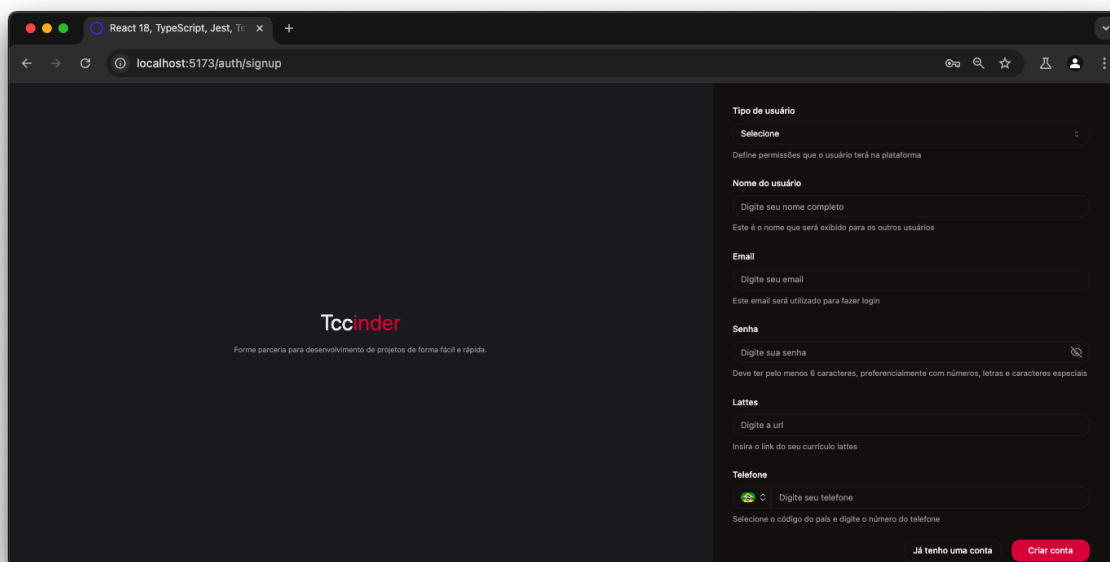


Figura 9 - Tela Cadastro do usuário

Após realizar seu cadastro, será redirecionado para a página principal do sistema para gerenciar projetos.

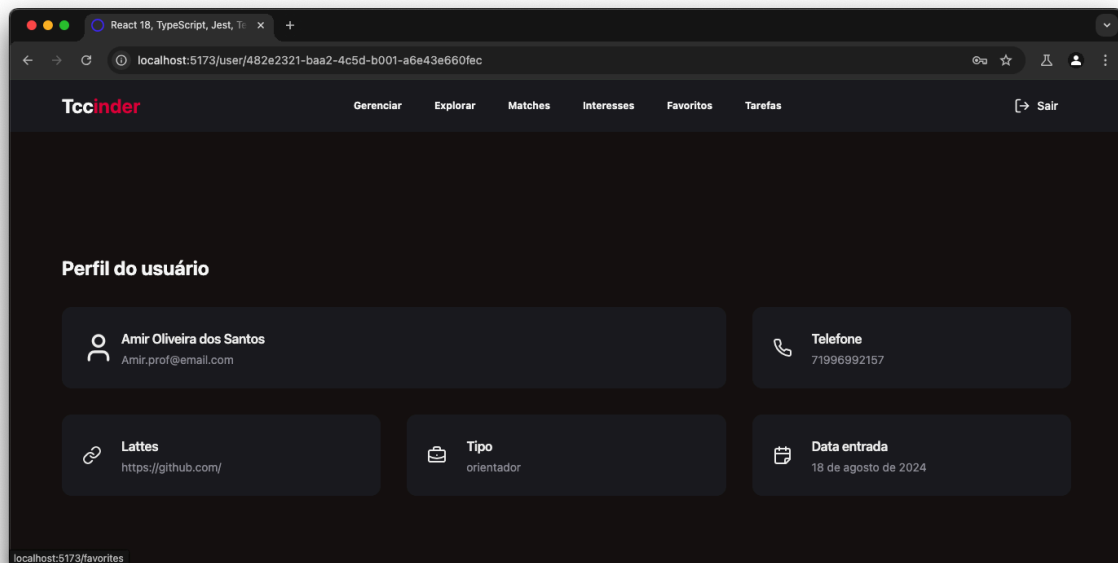


Figura 10 - Editar perfil

Caso necessário, o usuário pode editar o seu perfil e modificar seus dados.

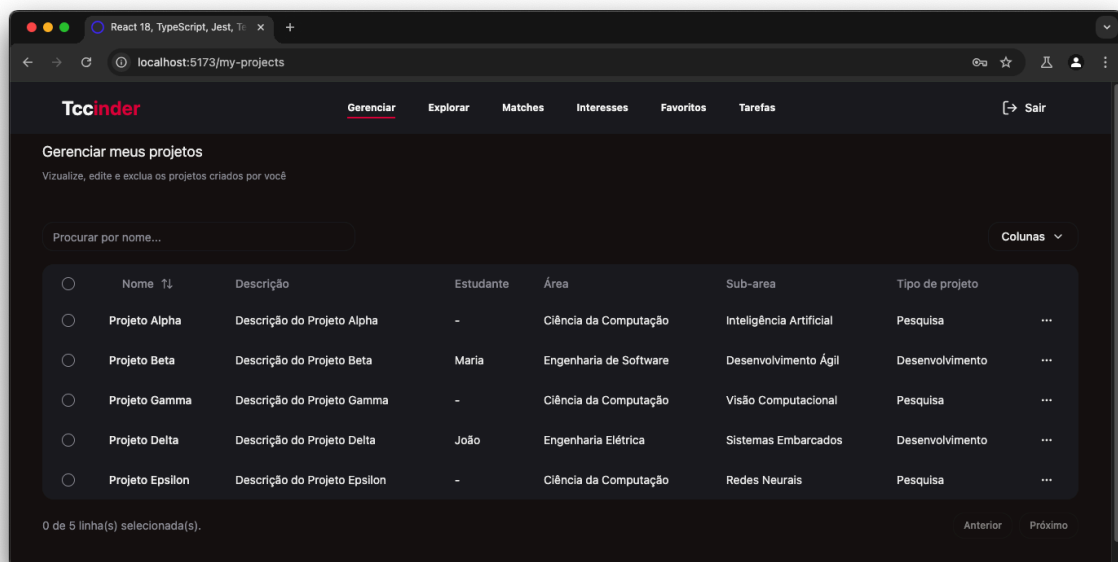


Figura 11 - Tela de gerenciar meus projetos

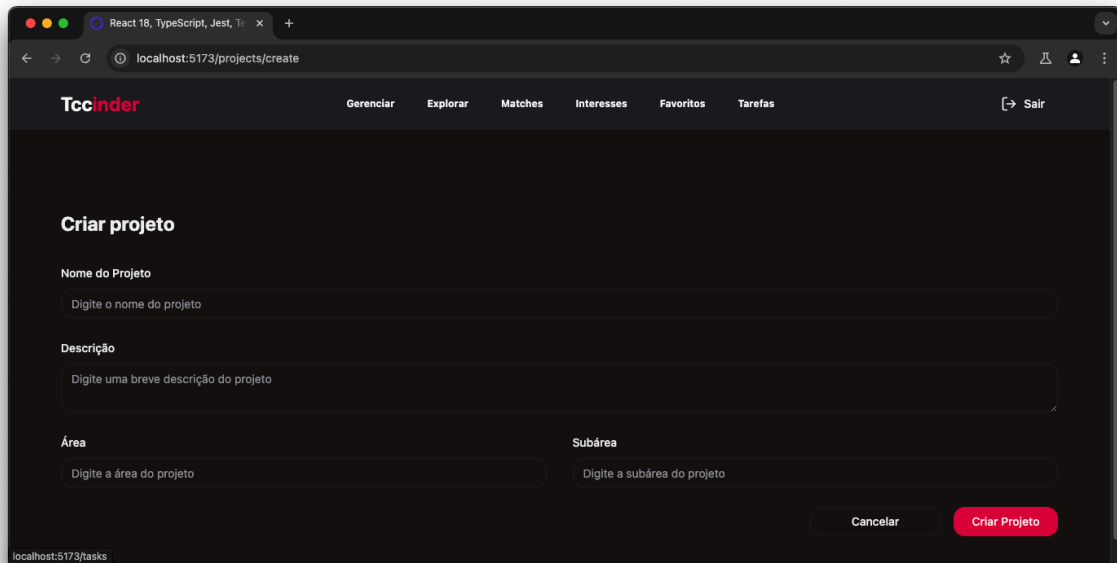


Figura 12 - Tela de criar um projeto

Nesta página, deve-se incluir as informações para divulgar seu projeto em busca de interessados (caso seja estudante, buscar um orientador, caso seja um professor, divulgar para encontrar estudantes interessados em projetos, por exemplo projetos do Labrasoft, Gsort ou outros ambientes de pesquisa).

O usuário possui alguns recursos que estão subdivididos entre:

- Gerenciar
- Explorar
- Matches
- Interesses
- Favoritos
- Tarefas

Na aba de gerenciar, abrange os meus projetos que foram criados pelo usuário mas que ainda não encontraram um colaborador para iniciar o projeto (seja estudante ou professor).

Na aba de Explorar, abrange os projetos criados por outros usuários que estão em busca de um colaborador, caso não possua um projeto ou tema para iniciar, o usuário deve buscar nesta aba um projeto para submeter interesse em participar.

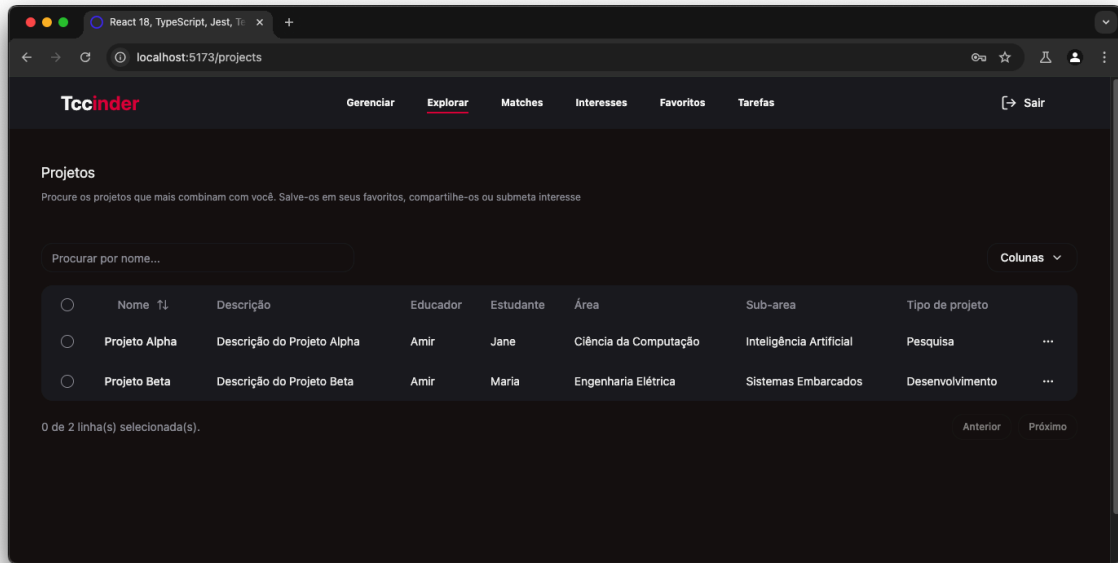


Figura 13 - Tela de explorar para buscar um projeto.

Na aba de Matches, abrange os projetos iniciados os quais o usuário participa ou é proprietário e encontrou um colaborador. Quando o usuário é um orientador, será comum ter mais de um registro pois pode acompanhar mais de um projeto simultaneamente.

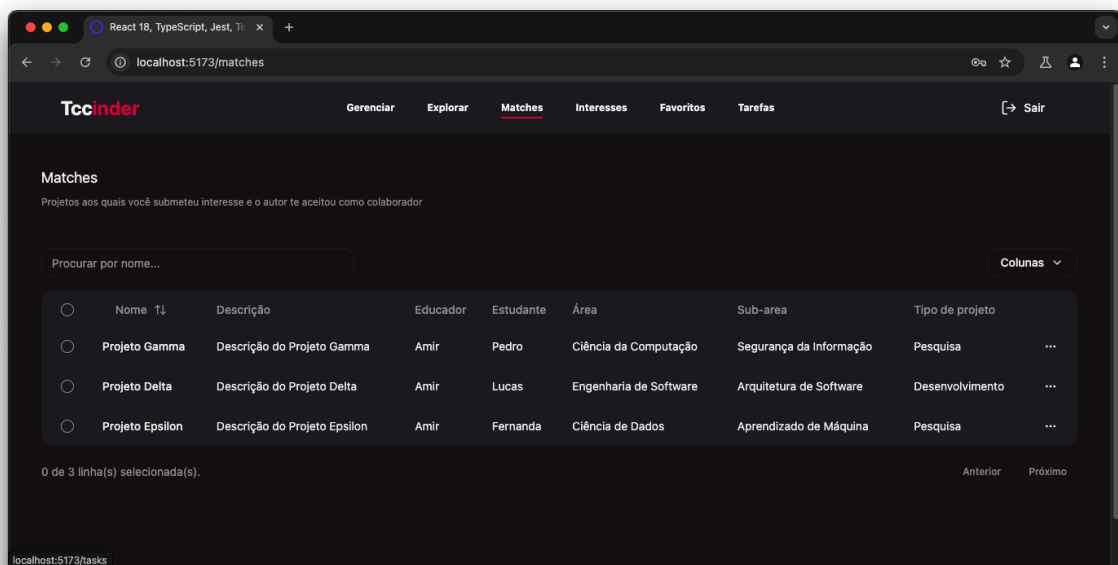


Figura 14 - Tela de Matches.

Na aba de Interesses, abrange a visualização de usuários interessados em participar nos seus projetos.

Na aba de Favoritos, abrange uma marcação filtrada dos projetos para facilitar busca e usabilidade tratando com prioridade.

Na aba de tarefas, abrange os recursos para criar tarefas para os projetos os quais o usuário participa.

Em cada aba disponibilizada, a tabela de dados possui o recurso de ações conforme exemplo que permite ações básicas de criar editar e deletar seus recursos. Por exemplo:

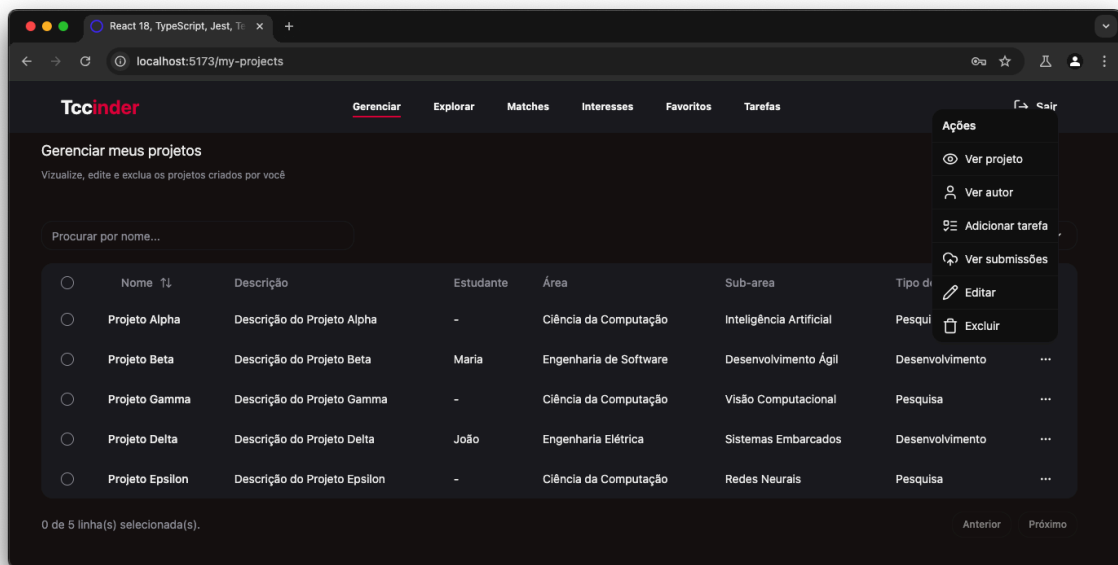


Figura 15 - Tela de gerenciar recursos apresentando ações disponíveis.

Agradecimentos

Gostaria de expressar minha profunda gratidão, primeiramente à minha família, que foi meu alicerce durante toda essa jornada acadêmica. Seu apoio incondicional foi fundamental para que eu pudesse chegar até aqui. Agradeço também a todos os docentes do Departamento de Computação, com quem tive o privilégio de aprender, interagir e compartilhar experiências enriquecedoras. Aos meus colegas, sou grato pela contribuição significativa que tiveram no meu crescimento e desenvolvimento ao longo dessa caminhada.

Por fim, um agradecimento especial à professora Flávia Maristela, que, além de ser uma excelente docente, foi minha orientadora, conselheira e amiga. Sua orientação constante, apoio e incentivo foram essenciais para que eu superasse os momentos difíceis e não desistisse.

Referências

[1] CASTRO, Sílvia Pereira. TCC Trabalho de conclusão de curso: uma Abordagem Leve, Divertida e Prática. Editora Saraiva, 2019.

[2] CABRAL, Leonardo Lima. TCC ONLINE: SISTEMA DE GERENCIAMENTO DE TRABALHOS DE CONCLUSÃO DE CURSO. Niterói, 2017. Disponível em:

<https://app.uff.br/riuff/handle/1/13104>. Acesso em: 01 de setembro de 2023.

[3] SILVA, Matheus Rodrigues Rosado da. Projeto e Desenvolvimento de um Sistema para Gerenciamento de Trabalhos de Conclusão de Curso. Uberlândia, 2017.

Disponível em:

<https://repositorio.ufu.br/bitstream/123456789/19409/1/ProjetoDesenvolvimentoSistema.pdf>. Acesso em: 01 de setembro de 2023.

[4] ASANA. Asana - Software de gerenciamento de projetos e tarefas para equipes.

Disponível em: <https://asana.com/pt>. Acesso em: 01 de setembro de 2023.