

PONTIFÍCIA UNIVERSIDADE CATÓLICA DE MINAS GERAIS
PUC Minas Virtual
Pós-graduação *Lato Sensu* em Engenharia de *Software*

Trabalho de Conclusão de Curso
Sistema de Gerenciamento de Projetos

Danilo Germiniani Virginio

Belo Horizonte
Fevereiro - 2022

Trabalho de Conclusão de Curso

Sumário

Trabalho de Conclusão de Curso	3
1. Cronograma de trabalho	4
2. Diagrama de casos de uso	5
3. Requisitos não-funcionais	5
4. Protótipo navegável do sistema	7
5. Diagrama de classes de domínio	7
6. Modelo de componentes	9
6.1. Padrão arquitetural	9
6.2. Diagrama de componentes	10
6.3. Descrição dos componentes	11
7. Diagrama de implantação	13
8. Plano de Testes	14
9. Estimativa de pontos de função	21
10. Informações da implementação	22
11. Referências	23

1. Cronograma de trabalho

Datas		Atividade / Tarefa	Produto / Resultado
De	Até		
06/11/2021	07/11/2021	1. Análise do roteiro, organização das atividades e preenchimento do cronograma.	Atividades planejadas e cronograma preenchido.
13/11/2021	14/11/2021	2. Definição e estudo da ferramenta de prototipação.	Ferramenta de prototipação definida.
20/11/2021	21/11/2021	3. Análise e refinamento dos requisitos.	Requisitos analisados, refinados e organizados.
27/11/2021	04/02/2022	4. Criação, atualização e manutenção do protótipo	Versões protótipo de acordo com os requisitos.
04/12/2021	04/12/2021	5. Levantamento dos requisitos não funcionais.	Lista dos requisitos não funcionais do sistema.
05/12/2021	05/12/2021	6. Estudo e desenvolvimento do diagrama de casos de uso.	Diagrama de casos de uso.
11/12/2021	12/12/2021	7. Estudo e desenvolvimento do diagrama de classes de domínio.	Diagrama de classes de domínio.
18/12/2021	19/12/2021	8. Estudo e desenvolvimento do padrão arquitetural.	Padrão arquitetural definido.
08/01/2022	08/01/2022	9. Estudo e desenvolvimento do diagrama de componentes.	Diagrama de componentes.
15/01/2022	16/01/2022	10. Estudo e desenvolvimento das descrições dos componentes	Descrição dos componentes.
22/01/2022	23/01/2022	11. Estudo e desenvolvimento do diagrama de implantação.	Diagrama de implantação.
22/01/2022	23/01/2022	12. Estudo e desenvolvimento do plano de testes.	Plano de testes.
01/02/2022	05/02/2022	13. Criação da estimativa de pontos de função.	Planilha de estimativa de pontos de função definidas.
29/01/2022	30/01/2022	14. Atualização das referências.	Lista das referências utilizadas.
31/01/2022	05/02/2022	15. Revisão e disponibilidade dos arquivos gerados nesse trabalho	Este documento finalizado e os demais arquivos disponibilizados para consulta
06/02/2022	06/02/2022	16. Envio do trabalho de conclusão de curso	Trabalho de conclusão de curso enviado.

2. Diagrama de casos de uso

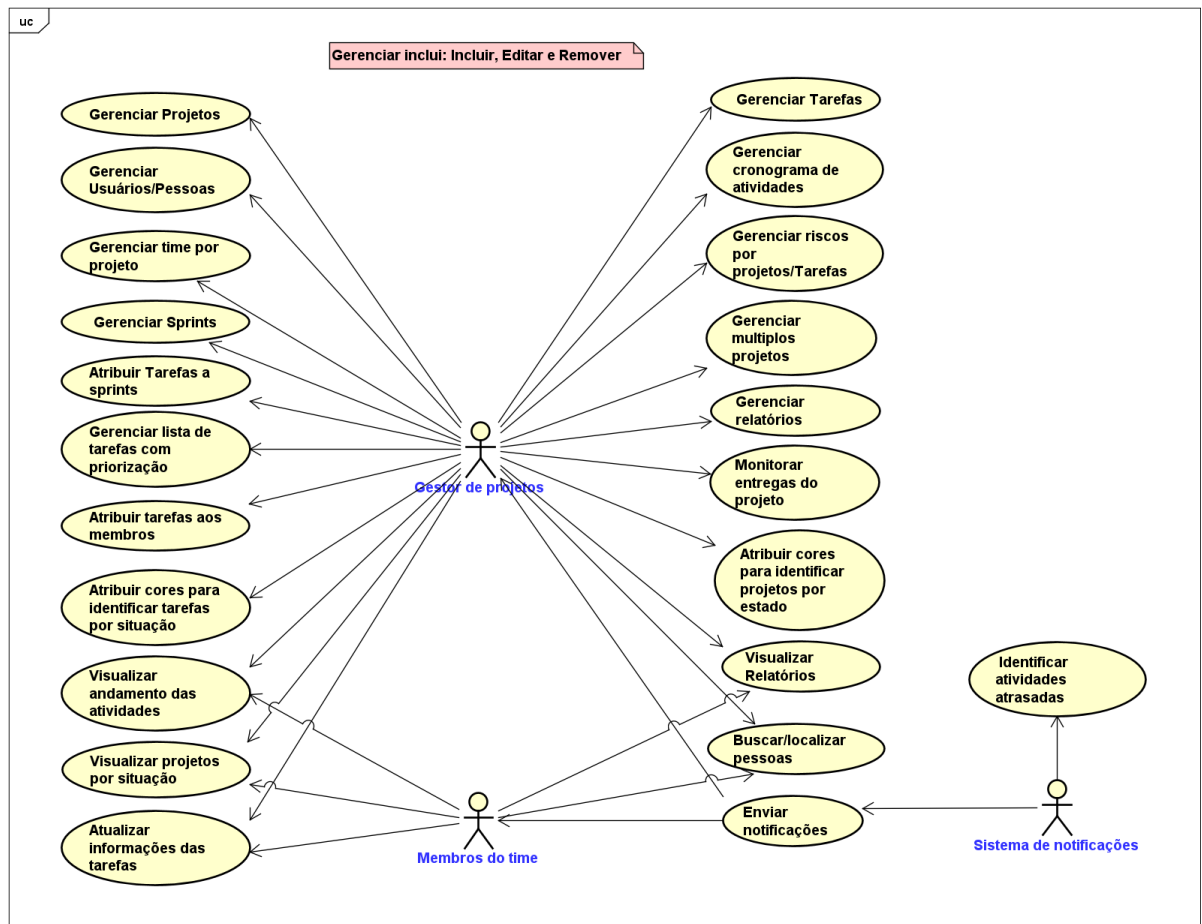


Figura 1 - Diagrama de casos de uso. Disponível em: https://github.com/DaniloGerminianiVirginio/TCC_ENG_SW_2022_PUC/tree/main/Diagramas

3. Requisitos não-funcionais

Ao realizar uma análise sobre o documento de requisitos propostos e de aplicar conceitos importantes sobre confiabilidade, usabilidade, segurança, entre outros, a seguinte lista de requisitos não funcionais foi gerada:

- 1) O sistema deve ficar totalmente disponível; 7 dias por semana e 24 horas por dia.
- 2) O sistema deverá ser acessado via Web e não possuir restrições em relação ao sistema operacional.
- 3) O sistema deverá ser acessado por navegadores web de dispositivos desktop e móvel; as validações deverão ocorrer nos navegadores mais utilizados no mercado: Google Chrome, Safari, Microsoft Edge e Mozilla Firefox.

Sistema de Gerenciamento de Projetos

- 4) O sistema não restringirá o acesso em outro navegador, porém caso o sistema encontre alguma circunstância inesperada, o usuário deverá ser notificado para utilizar alguma das opções do item 3.
- 5) O sistema deve apresentar confiabilidade contínua, mantendo seu funcionamento em circunstância de rotina e apresentar tratativas corretas em circunstâncias inesperadas.
- 6) O sistema não deverá apresentar mais do que 15 defeitos conhecidos; os defeitos não poderão apresentar severidade alta, caso contrário, a correção deve ser iniciada e a solução apresentada o mais rápido possível.
- 7) O sistema deverá possuir coleta e armazenamento de logs em caso de falhas ocorridas.
- 8) O sistema deverá apresentar execução de 100% de seu caderno de testes e nenhum defeito de severidade alta antes de ser disponibilizado.
- 9) O sistema deverá apresentar um plano de contingência, com a restauração de versões e base de dados anteriores, caso ocorra algumas circunstâncias hostis e inesperadas.
- 10) O sistema deverá ser responsivo, devendo apresentar suas funcionalidades independente da resolução ou configurações de acessibilidade do sistema operacional.
- 11) O sistema deverá suportar ao menos 500 usuários simultaneamente.
- 12) O sistema deverá consumir e apresentar as informações ao usuário em menos de 3 segundos, e quando necessário, notificar o usuário em caso de lentidão ou inacessibilidade.
- 13) O sistema deverá autenticar todos os usuários antes de apresentar qualquer informação.
- 14) O sistema deverá criptografar dados sensíveis do usuário.
- 15) O sistema deverá apresentar diretrizes gerais aplicáveis na aparência do software, buscando a melhor usabilidade para o usuário.

4. Protótipo navegável do sistema

O documento abaixo detalha os requisitos e casos de uso que foram utilizados na prototipação. Além disso, também apresenta as imagens da interface geradas pelo protótipo.

A ferramenta utilizada para a prototipação foi a [JustInMind](#).

O documento está disponível em:

1. Github:

https://github.com/DaniloGerminianiVirginio/TCC_ENG_SW_2022_PUC/blob/main/Prot%C3%B3tipo/Rela%C3%A7%C3%A3o%20dos%20Casos%20de%20Uso%20e%20Requisitos%20utilizados%20no%20Prot%C3%B3tipo.pdf

2. Google Drive:

https://drive.google.com/file/d/15LbviPPwzEWwnfxfnvDOMHFKD_RH4AVm/view?usp=sharing

O vídeo de apresentação do protótipo pode ser acessado em:

1. Github:

https://github.com/DaniloGerminianiVirginio/TCC_ENG_SW_2022_PUC/tree/main/Prot%C3%B3tipo/Video

2. Google Drive:

https://drive.google.com/file/d/1s8-Ls9AYpji6b7TM_-mfAZNjDEUnc8qm/view?usp=sharing

5. Diagrama de classes de domínio

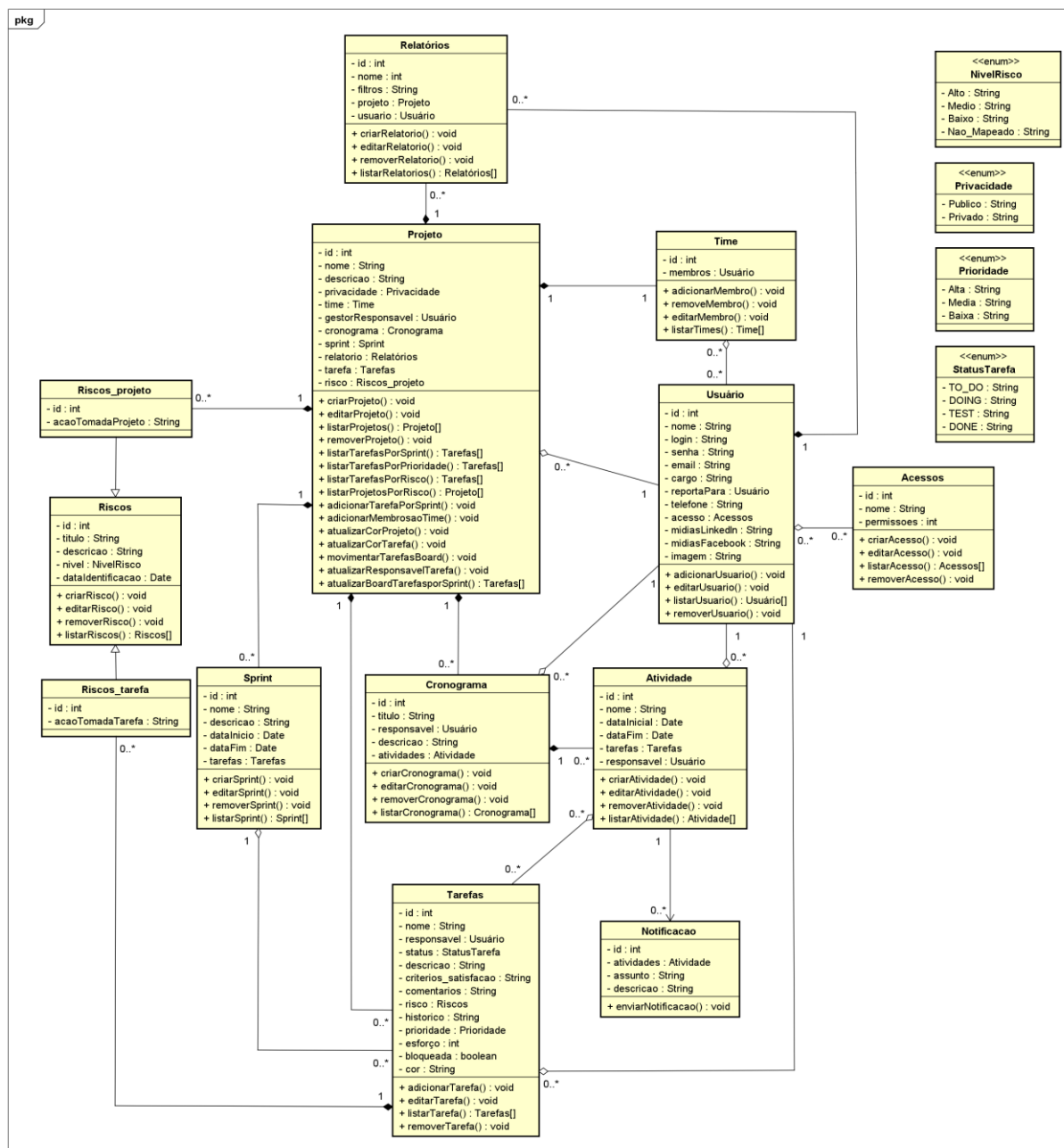


Figura 2 - Diagrama de classes. Disponível

em: https://github.com/DaniloGerminianiVirginio/TCC_ENG_SW_2022_PUC/tree/main/Diagramas

Modelo de componentes

6.1. Padrão arquitetural

Analizando a proposta do trabalho através dos requisitos fornecidos e considerando alguns aspectos como: escalabilidade, flexibilidade, resiliência, entre outros, optou-se pelo desenvolvimento de uma arquitetura híbrida, no qual o *backend* será baseado em microsserviços, enquanto o *frontend* será único, baseado em uma arquitetura monolítica.

As seguintes tecnologias foram utilizadas no planejamento e diagramação do projeto:

- [Angular v13](#)
- [HTML5](#)
- [TypeScript](#)
- [C3js v0.7](#)
- [Bootstrap v5](#)
- [JavaScript](#)
- [NodeJs v17](#)
- [Express v4](#)
- [Grafana Loki v2](#)
- [Grafana v8.3](#)
- [Mongo DB v5](#)
- [GitHub Actions](#)
- AWS
 - [Web Application Firewall](#)
 - [Api Gateway](#)
 - [Cognito](#)
 - [Elastic Load Balancing](#)
 - [Amazon MQ](#)
 - [Single Notification Service](#)
 - [Elastic Kubernetes Service](#)

6.2. Diagrama de componentes

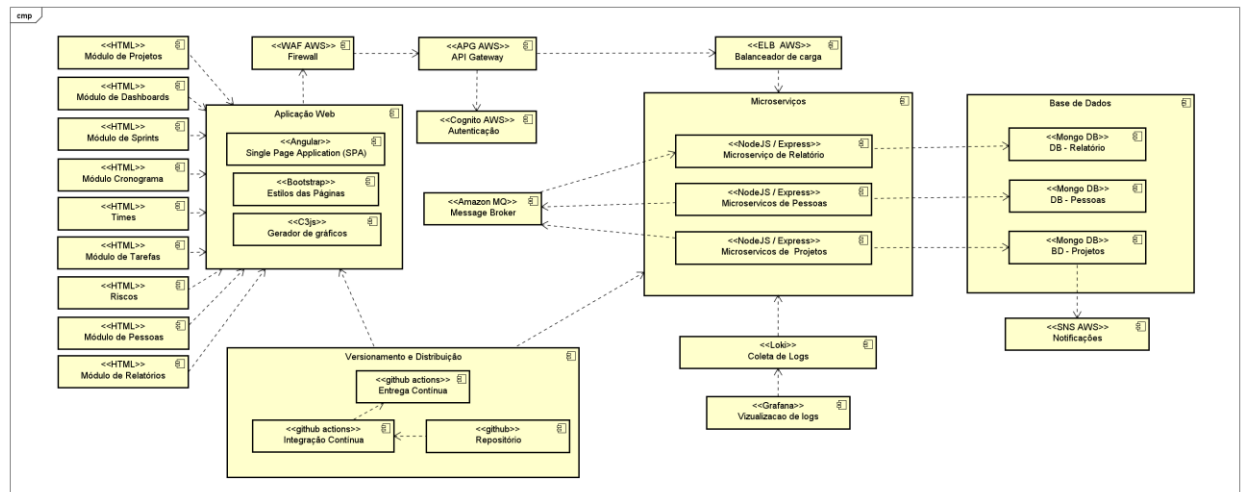


Figura 3 - Diagrama de Componentes. Disponível em: https://github.com/DaniloGerminianiVirginio/TCC_ENG_SW_2022_PUC/tree/main/Diagramas

6.3. Descrição dos componentes

Número	Componente	Descrição
1	Módulos HTML - Projetos, Dashboards, Sprints, Cronograma, Times, Tarefas, Riscos, Pessoas e Relatórios	Componentes responsáveis pela geração de conteúdo do sistema e que será consumido pela SPA.
2	Aplicação Web (SPA, Estilos das páginas e gerador de gráficos e relatórios)	Componente responsável por gerar a interface para o usuário, de maneira responsiva e gerenciar gráficos e relatórios.
3	Firewall (WAF AWS)	Responsável por proteger a aplicação contra <i>bots</i> e <i>exploits</i> comuns na Web que podem afetar a disponibilidade, comprometer a segurança ou consumir recursos em excesso.
4	API Gateway	Fornece um ponto de entrada única para todas as rotas expostas, roteando solicitações de clientes para serviços.
5	Autenticação (Cognito)	Componente responsável por realizar a autenticação dos usuários e a autorização externa de aplicações.
6	Balanceador de Carga (ELB AWS)	Responsável por distribuir automaticamente o tráfego de aplicações, garantindo alta disponibilidade e distribuição de cargas nos serviços.

7	Microserviços (Projetos, Pessoas e Relatórios)	Componentes independentes responsáveis por determinadas partes do projeto. Microserviço de Projeto: Responsável pelas operações relacionadas ao projeto, sprint, cronograma, time, riscos e tarefas. Microserviço de Pessoas: Responsável pelas operações relacionadas aos usuários (Pessoas). Microserviço de Relatório: Responsável pelas operações relacionadas na geração de relatórios sobre projetos e pessoas.
8	Message Broker (Amazon MQ)	Componente responsável por garantir o armazenamento e enfileiramento de mensagens. As mensagens geradas pelos microserviços de projetos e pessoas são consumidas pelo microserviço de relatório.
9	Coleta de logs (Grafana Loki)	Componente responsável pela coleta e agregação de logs da aplicação.
10	Visualizador de logs (Grafana)	Componente responsável por coletar os logs gerados pelo componente acima e apresentá-los de forma organizada e visual.
11	Banco de Dados (Mongo DB)	Componente responsável pela base de dados da aplicação. O projeto contará com 3 bases distintas: Projetos, Pessoas e Relatórios.
12	Versionamento e distribuição – Repositório, Integração Contínua e Entrega Contínua (Github actions)	Componente responsável pelo armazenamento, construção, gerenciamento de dependências, testes e entrega das versões.
13	Notificação (SNS AWS)	Componente responsável pelo envio da notificação ao usuário.

6. Diagrama de implantação

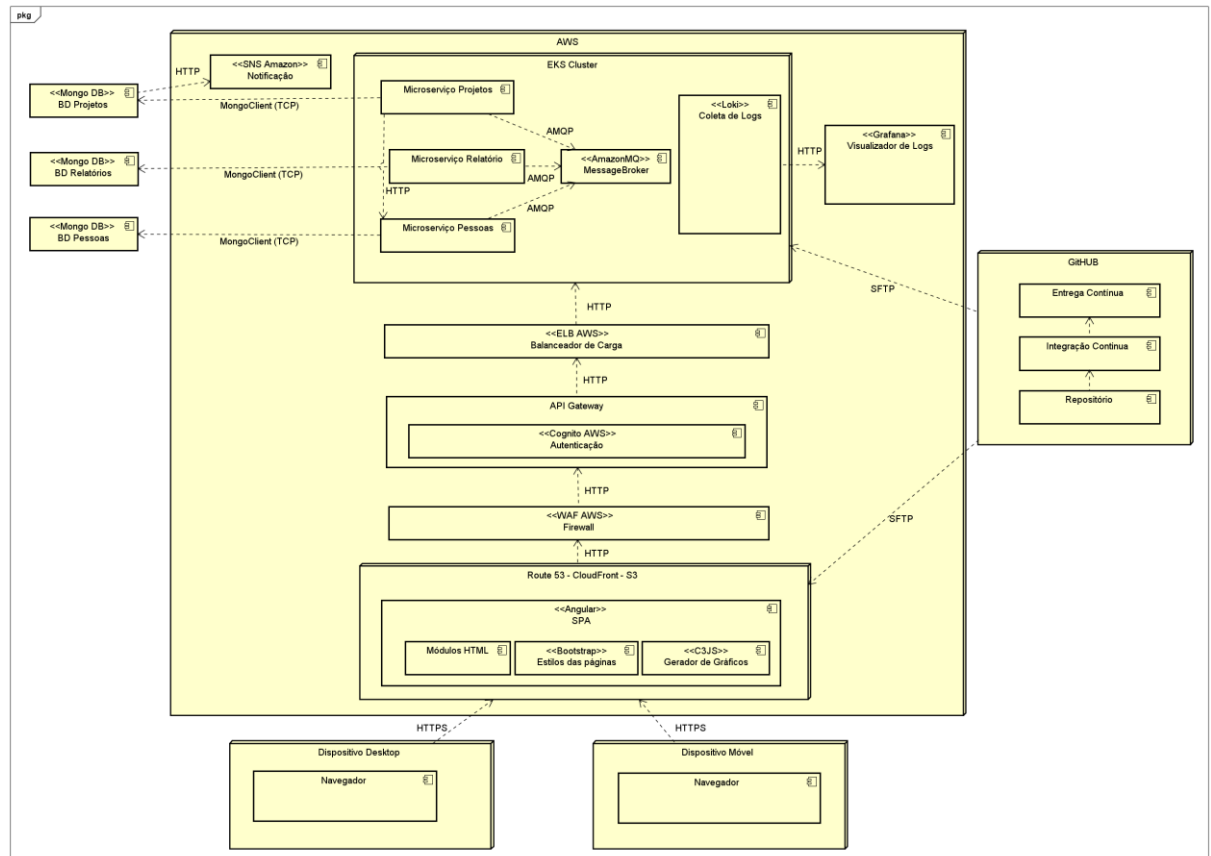


Figura 4 - Diagrama de implantação. Disponível em: https://github.com/DaniloGerminianiVirgino/TCC_ENG_SW_2022_PUC/tree/main/Diagramas

7. *Plano de Testes*

O plano de testes abaixo foi planejado afim de validar alguns cenários, envolvendo os casos de uso: Gerenciar Projetos, Gerenciar Usuários/Pessoas e Gerenciar Sprints.

Como complemento, os testes foram executados no protótipo e as evidências podem ser encontradas nos links abaixo:

1. Github:

https://github.com/DaniloGerminianiVirginio/TCC_ENG_SW_2022_PUC/tree/main/Testes

2. GoogleDrive:

<https://drive.google.com/drive/folders/1jXX2W1IA7c3ShISycYF1SSyTZofJ-VyH?usp=sharing>

Número	Caso de uso	Objetivo do caso de teste	Entradas	Resultados esperados
1	Gerenciar projetos	Validar a inserção de um novo projeto.	<p>Pré-condições:</p> <ol style="list-style-type: none"> 1) Possuir acesso a plataforma. 2) Possuir nível de acesso para cadastrar novo projeto. <p>Passos:</p> <ol style="list-style-type: none"> 1) Acessar a plataforma e realizar o login; 2) Clicar em menu ‘Projetos’ e então em ‘Gerenciar Projetos’; Selecionar a opção: “Criar Projeto”; 3) Preencher os campos com os respectivos dados: Nome do Projeto: Projeto TCC Teste Descrição: Testes criação de projeto com sucesso. Privacidade: Público Time: Selecionar ao menos 1 pessoa da lista. Gestor Responsável: Selecionar ao menos 1 pessoa da lista. 4) Clicar no botão “Criar” 	<ol style="list-style-type: none"> 1) Uma janela com a mensagem: “Projeto criado com sucesso” deve ser mostrada. 2) O projeto criado deve ser listado na lista de projetos.

2	Gerenciar projetos	Validar a tentativa de inserção de um novo projeto com um nome já existente.	<p>Pré-condições:</p> <ol style="list-style-type: none"> 1) Possuir acesso a plataforma. 2) Possuir nível de acesso para cadastrar novo projeto. 3) Possuir um projeto cadastrado com o nome “Projeto TCC Teste” <p>Passos:</p> <ol style="list-style-type: none"> 1) Acessar a plataforma e realizar o login; 2) Clicar em menu ‘Projetos’ e então em ‘Gerenciar Projetos’; Selecionar a opção: “Criar Projeto”; 3) Preencher os campos com os respectivos dados: <p>Nome do Projeto: Projeto TCC Teste</p> <p>Descrição: Testes criação de projeto com sucesso.</p> <p>Privacidade: Público</p> <p>Time: Selecionar ao menos 1 pessoa da lista.</p> <p>Gestor Responsável: Não selecionar nenhuma pessoa da lista.</p> <ol style="list-style-type: none"> 4) Clicar no botão “Criar” 	<ol style="list-style-type: none"> 1) Uma janela com a mensagem: “Já existe um projeto com esse nome. Por favor, verifique as informações fornecidas e tente novamente.” deve ser mostrada. 2) Um botão “Ok” deve ser mostrado e ao clicar nele o usuário é redirecionado para a tela de criação de projeto.
---	--------------------	--	--	--

3	Gerenciar sprints	Validar a inserção de uma nova sprint.	<p>Pré-condições:</p> <ol style="list-style-type: none"> 1) Possuir acesso a plataforma. 2) Possuir nível de acesso para criar uma nova sprint. 3) Possuir um projeto cadastrado com o nome “Projeto I”. 4) Possuir ao menos uma tarefa cadastrada. <p>Passos:</p> <ol style="list-style-type: none"> 1) Acessar a plataforma e realizar o login; 2) Clicar em menu ‘Projetos’ e então selecione o projeto “Projeto I”. 3) Selecionar a opção sprint no menu lateral esquerdo, clique em ‘Gerenciar Sprints’ e então em ‘Criar Sprint’. 4) Preencher os campos com os respectivos dados: <p>Nome da Sprint: Sprint TCC Teste</p> <p>Descrição: Teste de criação de sprint com sucesso.</p> <p>Data Início: 03/01/2022</p> <p>Data Fim: 17/01/2022</p> <p>Tarefas do Backlog: Selecionar ao menos 1 tarefa da lista.</p> <ol style="list-style-type: none"> 5) Clicar no botão “Criar” 	<ol style="list-style-type: none"> 1) Uma janela com a mensagem: “Sprint criada com sucesso” deve ser mostrada. 2) A sprint criada deve ser listada na lista de sprints.
---	-------------------	--	---	--

4	Gerenciar Sprints	Validar a tentativa de inserção de uma nova sprint com a data final anterior à data inicial.	<p>Pré-condições:</p> <ol style="list-style-type: none"> 1) Possuir acesso a plataforma. 2) Possuir nível de acesso para criar uma nova sprint. 3) Possuir um projeto cadastrado com o nome “Projeto I”. 4) Possuir ao menos uma tarefa cadastrada. <p>Passos:</p> <ol style="list-style-type: none"> 1) Acessar a plataforma e realizar o login; 2) Clicar em menu ‘Projetos’ e então selecione o projeto “Projeto I”. 3) Selecionar a opção sprint no menu lateral esquerdo, clique em ‘Gerenciar Sprints’ e então em ‘Criar Sprint’. 4) Preencher os campos com os respectivos dados: <p>Nome da Sprint: Sprint TCC Teste</p> <p>Descrição: Teste de criação de sprint com sucesso.</p> <p>Data Início: 17/01/2022</p> <p>Data Fim: 03/01/2022</p> <p>Tarefas do Backlog: Selecionar ao menos 1 tarefa da lista.</p> <ol style="list-style-type: none"> 5) Clicar no botão “Criar” 	<ol style="list-style-type: none"> 1) Uma janela com a mensagem: “A data inicial não pode ser posterior a data final. Por favor, verifique as informações fornecidas e tente novamente.” deve ser mostrada. 2) Um botão “Ok” deve ser mostrado e ao clicar nele o usuário é redirecionado a tela de criação de sprint.
---	-------------------	--	---	--

5	Gerenciar Usuários/ Pessoas	Validar a inserção de uma nova pessoa.	<p>Pré-condições:</p> <ol style="list-style-type: none"> 1) Possuir acesso a plataforma. 2) Possuir nível de acesso para cadastrar uma nova pessoa. <p>Passos:</p> <ol style="list-style-type: none"> 1) Acessar a plataforma e realizar o login; 2) Selecionar a opção ‘Pessoas’ no menu lateral esquerdo, clique em ‘Gerenciar Pessoas’ e então em ‘Adicionar Pessoa’. 3) Preencher os campos com os respectivos dados: <p>Nome da Pessoa: Pessoa Teste</p> <p>Email: teste@tcc.com</p> <p>Cargo: Tester</p> <p>Reporta para: Diretor A</p> <p>Telefone: (35)99211896</p> <p>Mídias Sociais:</p> <p>LinkedIn - https://www.linkedin.com/in/pessoa-teste</p> <p>Facebook - https://www.facebook.com/pessoa-teste</p> <p>Acessos: Visualizar pessoas; Visualizar projetos;</p> <p>Imagem: Selecionar uma imagem;</p> <p>Login: pessoa_teste;</p> <p>Senha: pessoa_teste_123</p> <p>4) Clicar no botão “Criar”</p>	<p>1) Uma janela com a mensagem: “Pessoa cadastrada com sucesso” deve ser mostrada.</p> <p>2) A pessoas deve ser listada na lista de pessoas;</p>
---	-----------------------------	--	---	---

6	Gerenciar Usuários/ Pessoas	Validar a tentativa de inserção de uma nova pessoa sem preencher o campo obrigatório nome.	<p>Pré-condições:</p> <ol style="list-style-type: none"> 1) Possuir acesso a plataforma. 2) Possuir nível de acesso para cadastrar uma nova pessoa. <p>Passos:</p> <ol style="list-style-type: none"> 1) Acessar a plataforma e realizar o login; 2) Selecionar a opção ‘Pessoas’ no menu lateral esquerdo, clique em ‘Gerenciar Pessoas’ e então em ‘Adicionar Pessoa’. 3) Preencher os campos com os respectivos dados: <p>Nome da Pessoa: [Não preencher este campo.]</p> <p>Email: pessoa_tst@tcc.com</p> <p>Cargo: Tester Sr</p> <p>Reporta para: Diretor A</p> <p>Telefone: (35)99211896</p> <p>Mídias Sociais:</p> <p>LinkedIn - https://www.linkedin.com/in/pessoa-teste</p> <p>Facebook - https://www.facebook.com/pessoa-teste</p> <p>Acessos: Visualizar pessoas; Visualizar projetos;</p> <p>Imagem: Selecionar uma imagem;</p> <p>Login: pessoa_teste_2</p> <p>Senha: pessoa_teste_123</p> <p>4) Clicar no botão “Criar”</p>	<p>1) Uma janela com a mensagem: “Não foi possível cadastrar a pessoa. Preencha o campo nome.” deve ser mostrada.</p> <p>2) Um botão “Ok” deve ser mostrado e ao clicar nele o usuário é redirecionado a tela de cadastro da pessoa.</p>
---	-----------------------------	--	---	--

8. Estimativa de pontos de função

A seguinte planilha de estimativa de pontos de função foi gerada:

Planilha de contagem de ponto de função - Versão 2.4												
Aplicação : Sistema de Gerenciamento de Projetos			Projeto : Sistema de Gerenciamento de Projetos			PF IFPUG			242,00			
Responsável : Danilo Germiniani Virgínio			Revisor : Danilo Germiniani Virgínio			PF Local do EM			242,00			
Empresa : --			Tipo da Contagem : Projeto de Desenvolvimento			PF Local da FS			0,00			
Nome da Função	Tip	Maintenance	AR	U	comple	PF IFPUG	PF Local do EM	PF Local da FS	Peso	Referência	Observações	
Usuário (Usuário e Acesso)	ALI		12	2	Baixa	7						7,00
Criar Usuário (Usuário e Acesso)	EE		12	2	Média	4						4,00
Criar níveis de acesso para usuário (Acesso)	EE		3	1	Baixa	3						3,00
Editar Usuário (Usuário e Acesso)	EE		11	2	Média	4						4,00
Remover Usuário (Usuário)	EE		12	1	Baixa	3						3,00
Listar Usuários (Usuário)	CE		10	1	Baixa	3						3,00
Projeto (Projeto, Usuário, Time, Risco, Sprint, cronograma, Tarefa, Relatório)	ALI		11	8	Média	10						10,00
Criar Projeto (Projeto, Time, Usuário)	EE		6	3	Alta	6						6,00
Editar Projeto (Projeto, Time)	EE		5	2	Média	4						4,00
Remover Projeto (Projeto, Time, Risco, Sprint, cronograma, Tarefa, Relatório)	EE		11	7	Alta	6						6,00
Listar Projetos (Projetos)	CE		5	1	Baixa	3						3,00
Time (Time, Usuário)	ALI		2	2	Baixa	7						7,00
Adicionar Membro do Time (Time, Usuário)	EE		2	2	Baixa	3						3,00
Editar Membros do Time (Time, Usuário)	EE		1	2	Baixa	3						3,00
Listar Membros do Time (Time)	CE		1	1	Baixa	3						3,00
Remover Membro do Time (Time)	EE		1	1	Baixa	3						3,00
Sprints (Sprint, Tarefa)	ALI		6	2	Baixa	7						7,00
Criar Sprints (Sprint, Tarefa)	EE		6	2	Média	4						4,00
Editar Sprints (Sprint, Tarefa)	EE		5	2	Média	4						4,00
Remover Sprints (Sprint)	EE		6	1	Baixa	3						3,00
consultar andamento das sprints (Sprint)	SE		5	1	Baixa	4						4,00
Listar Sprints (Sprint)	CE		5	1	Baixa	3						3,00
Tarefas (Tarefas, Usuário, Riscos)	ALI		13	3	Baixa	7						7,00
Adicionar Tarefas (Tarefas, Usuário, Riscos)	EE		13	3	Alta	6						6,00
Remover Tarefas (Tarefas, Riscos)	EE		13	2	Média	4						4,00
Editar Tarefas (Tarefas, Usuário, Riscos)	EE		11	3	Alta	6						6,00
Listar Tarefas (Tarefas)	CE		12	1	Baixa	3						3,00
consultar andamento das tarefas (Tarefas)	SE		12	1	Baixa	4						4,00
Listar tarefas por cores (Tarefas)	SE		12	1	Baixa	4						4,00
cronograma (cronograma, Atividade, Usuário)	ALI		5	3	Baixa	7						7,00
Criar cronograma (cronograma, Atividade, Usuário)	EE		5	3	Alta	6						6,00
Editar cronograma (cronograma, Atividade, Usuário)	EE		4	3	Média	4						4,00
Remover cronograma (cronograma, Atividade)	EE		5	2	Média	4						4,00
Listar cronogramas (cronograma)	CE		4	1	Baixa	3						3,00
Atividades (Atividades, Tarefas)	ALI		5	2	Baixa	7						7,00
Criar Atividades (Atividades, Tarefas)	EE		5	2	Média	4						4,00
Editar Atividades (Atividades, Tarefas)	EE		4	2	Baixa	3						3,00
Remover Atividades (Atividades)	EE		5	1	Baixa	3						3,00
Listar Atividades (Atividades)	CE		4	1	Baixa	3						3,00
Riscos (Riscos)	ALI		6	1	Baixa	7						7,00
Adicionar riscos de projetos (Riscos)	EE		6	1	Baixa	3						3,00
Editar riscos de projeto (Riscos)	EE		5	1	Baixa	3						3,00
Remover riscos de projetos (Riscos)	EE		6	1	Baixa	3						3,00
Listar riscos de projetos (Riscos)	CE		5	1	Baixa	3						3,00
Editar riscos de tarefas (Riscos)	CE		5	1	Baixa	3						3,00
Adicionar riscos de tarefa (Riscos)	EE		6	1	Baixa	3						3,00
Remover riscos de tarefa (Riscos)	EE		6	1	Baixa	3						3,00
Listar riscos dos tarefa (Riscos)	CE		5	1	Baixa	3						3,00
consultar atividades em atraso (Atividades)	SE		3	1	Baixa	4						4,00
Enviar notificações (notificacao e Atividades)	SE		4	2	Baixa	4						4,00
Relatórios (Relatórios, Projetos, Usuário)	ALI		3	3	Baixa	7						7,00
Criar relatório (Relatórios, Projetos, Usuário)	EE		5	3	Alta	6						6,00
Editar relatório (Relatórios, Projetos, Usuário)	EE		4	3	Média	4						4,00
Remover relatório (Relatórios)	EE		5	1	Baixa	3						3,00
Listar relatórios (Relatórios)	SE		3	1	Baixa	4						4,00
Fazer login	SE		2	1	Baixa	4						4,00

Figura 5- Planilha de pontos de função

A planilha pode ser encontrada em:

1. Github:

https://github.com/DaniloGerminianiVirgínio/TCC_ENG_SW_2022_PUC/blob/main/Documentos/Planilha%20APF%20-%20Danilo%20Germiniani.xls

2. GoogleDrive:

https://docs.google.com/spreadsheets/d/1uyQwOVQA27AGx-wjfb7Ua4_754gMoHkOj2xPLDoP2z0/edit?usp=sharing

9. Informações da implementação

Item não implementado.

10. Referências

FAKHROUTDINOV, Kirill. UML 2.5 Diagrams Overview. *uml-diagrams*, 2016. Disponível em: <<https://www.uml-diagrams.org/uml-25-diagrams.html>>. Acesso em: 05 de dez. de 2021.

ÖZKAYA, Mehmet. Microservices Architecture. *Medium*, 2021. Disponível em: <<https://medium.com/design-microservices-architecture-with-patterns/microservices-architecture-2bec9da7d42a>>. Acesso em: 18 de jan. de 2022.

MORGAN, Andrew. The Modern Application Stack – Part 1: Introducing The MEAN Stack. *MongoDB*, 2017. Disponível em: <<https://www.mongodb.com/blog/post/the-modern-application-stack-part-1-introducing-the-mean-stack>>. Acesso em: 18 de jan. de 2022.

Frameworks for Developing Single Page Applications in 2022. *Toobler*, 2022. Disponível em: <<https://www.toobler.com/blog/frameworks-for-developing-single-page-applications>>. Acesso em: 18 de jan. de 2022.

RESELMAN, Bob. Synchronous vs. asynchronous microservices communication patterns. *The Server Side*, 2021. Disponível em: <<https://www.theserverside.com/answer/Synchronous-vs-asynchronous-microservices-communication-patterns#:~:text=A%20synchronous%20microservice%20is%20one,seen%20in%20Figure%201%20below.>>. Acesso em: 22 de jan. de 2022.

LOPES, Jhoney da Silva. Guia Prático em Análise de Ponto de Função. *Fattocs*, 2011. Disponível em: <<https://www.fattocs.com/wp-content/uploads/2020/04/JhoneySLopes-JoseLBraga-2011.pdf>>. Acesso em: 30 de jan. de 2022.