

UNIVERSIDADE CATÓLICA DE PELOTAS



Anderson Moreira Passos

Bruno Ribeiro

Felipe Schein Mello

Entrega parcial da disciplina de Projeto Integrador IV - B

Curso: Análise e Desenvolvimento de Sistemas

INSTRUTORES: Giancarlo Lucca

Rogério Albandes

1. Introdução

O presente trabalho é uma parte crucial para avaliação do Projeto Integrador IV-B, que compõem as disciplinas de Arquitetura de Sistemas e Integração de aplicações, a abordagem tomada nesta aplicação se pauta por uma Modelagem ágil para uma implementação de um API REST, desenvolvida em React com Vite, para gerenciamento de uma lista de clientes. O presente sistema desenvolvido deverá permitir adicionar, editar, excluir e pesquisar clientes, além de indicar a quantidade total de clientes cadastrados.

Pautados nestas premissas e os tópicos abordados pelas trilhas de conhecimentos da instituição, o grupo desenvolveu uma aplicação que pudesse representar o solicitado pelos professores da disciplina, esta atividade, tem por intermedio a parte prática do que é aprendido nas duas disciplinas que englobam o Projeto Integrador, através da Arquitetura de sistemas, com a separação e organização do desenvolvimento do trabalho, podemos aplicar uma integração entre as aplicações.

A organização do trabalho ficou por conta de uma metodologia ágil, baseada em *Scrum*, a equipe dividiu no total de 4 *sprint* para a elaboração da primeira entrega do trabalho, a equipe fez uma primeira reunião, à qual reuniu todas as informações necessárias para o desenvolvimento do projeto (Contando com a primeira *Sprint*), ainda nesta reunião foi delimitado a separação de cada etapa e quais os processos e tecnologias a equipe iria utilizar, lembrando que todo o desenvolvimento foi pautado pelas ordens primárias dos professores da disciplina, compete apenas a equipe a escolher as metodologias mais aprofundadas.

Dentro da primeira *Sprint* então foi dado partida no sistema de trabalho, o grupo desenvolveu as funções básicas da aplicação, como Adicionar, Editar, Cadastrar, Pesquisar, Excluir e um contador de Clientes existentes, foi desenvolvida através do banco de dados JSON Serve para armazenagem dos clientes, o que facilita na integração das aplicações e também no gerenciamento, também foi utilizado a integração de vite, para acesso a este banco de dados e gerenciamento da aplicação.

Dentro do Segundo *Sprint*, foi feito uma análise do que já estava desenvolvido, estrutura de código, estrutura de Documentação e a integração das mesmas, assim foi dado partida no restante do desenvolvimento.

Dentro da terceira *Sprint*, foi reunido toda a aplicação, juntamente com a documentação produzida, para análise e entrega parcial, o que deve ocorrer, entre os dias 18/nov até 25/nov, após esta entrega o grupo se reunirá para analisar o feedback gerado pelos

Docentes e assim delimitar os próximos passos a serem desenvolvidos na aplicação. Estima-se que deve ser feito mais dois *Sprint*, até a entrega final, somando um total de 6 *sprints*, onde cada uma determina os próximos passos necessários para o término da aplicação

a. Delimitação dos membros:

Aqui utilizamos uma estrutura simples para definição e organização dos nossos membros, como ponto de partida estabeleceremos nossos **Usuários**, nesta aplicação, dado a parte pedagógica do curso, nossos usuários, serão nossos docentes, o qual irão analisar a aplicação e posteriormente nos darão um Feedback referente ao mesmo, aqui um pouco mais técnica. Hipoteticamente, também podemos determinar que nossos usuários pode ser um cadastro prévio para geração de *leads* comerciais de uma empresa, assim como os próprios discentes o qual devem ter um olhar criterioso para a própria aplicação.

Outro ponto a ser determinado aqui são os **Desenvolvedores**, sendo os Discentes da disciplina, neste caso, os alunos presente neste relatório, o qual, cada um ficou responsável por implementar uma funcionalidade, cabe ressaltar que o colega Felipe estruturou e configurou os principais caminhos da aplicação, ficando a cargo dos outros colegas a formatação e estruturação da documentação, outro ponto a ser ressaltado é que todos os processos foram feitos em acordo e conjunto, apenas cada um ficou responsável por uma parte.

Ainda na organização do trabalho temos os **Testadores**, que aqui adotamos os próprios Discentes, que ao desenvolver a aplicação já iam realizando os testes, ainda como testadores, temos os Docentes da Disciplina, que serão os principais responsáveis por testar e avaliar a aplicação.

b. Requisitos

i. Fundamentais

- O sistema deve permitir o cadastro de novos clientes;
- O sistema deve permitir a edição de clientes já cadastrados;
- O sistema deve permitir a exclusão de clientes cadastrados;
- O sistema deve permitir a busca por clientes já cadastrados;

ii. Secundários

- O sistema deve conter um contador de cadastros;
- O sistema deve mostrar os clientes cadastrados;
- O sistema deve ser intuitivo com os botões de “novo, editar, excluir”;

c. Cronograma

Cronograma		
Outubro	Novembro	Dezembro
Organização e Estruturação do relatório Parcial	Começo da escrita da Documentação	Desenvolvimento da Aplicação
Estruturação da aplicação, delimitação das tecnologias utilizadas	Levantamento das Metodologias	Desenvolvimento do Relatório Final
Levantamento de Informações e Recursos para Desenvolvimento do Software	Levantamento dos Requisitos (Principais, Secundários, Funcionais e Não Funcionais)	Entrega da aplicação
Pesquisa	Entrega do relatório Parcial de Desenvolvimento (Estima-se que seja entregue toda a documentação do Software)	Entrega do Relatório Final

2. Modelagem

Neste capítulo iremos determinar quais as tecnologias que iremos utilizar, vamos começar pelo ambiente de desenvolvimento, num primeiro momento iremos utilizar um editor de código, aqui pode ser qualquer um disponível no mercado, porém a equipe utilizou o **Visual Studio Code**, salientamos sempre para utilizar a última versão disponível, visto que o processo de aprimoramento das aplicações estão em constante evolução.

Definido o ambiente de edição do código, buscamos algumas tecnologias mais aprofundadas, para podermos modelar nossos API, de forma a ter responsividade e organização, não só no código, mas também na apresentação final, tecnologias como **Node**

JS, para trabalhar com o banco de dados de cadastros e o **JSON**, Utilizado para simular uma API REST com operações CRUD, assim como foi utilizado, a biblioteca **React**, como base principal para a interface do usuário, em conjunto também foi utilizado a biblioteca **React DOM**, para manipulação do DOM com React, foi também configurado a Ferramenta de desenvolvimento para criação de projetos React rápida e eficientemente **Vite**. Assim como os plugins **ESLint** e **Plugins**, uma ferramenta de linting para manter o código limpo e consistente.

3. Análise de Requisitos

a. Funcionais

Identificador	RF001	Categoria:	Cadastro
Nome:	Cadastro de cliente		
Data de Criação:	17/11/2024	Autor:	Felipe Mello
Data Alteração:	N/A	Autor:	N/A
Versão:	1	Prioridade:	Essencial
Descrição:	O sistema deve permitir que o usuário cadastre um novo cliente		

Identificador	RF002	Categoria:	Cadastro
Nome:	Editar Cliente		
Data de Criação:	17/11/2024	Autor:	Felipe Mello
Data Alteração:	N/A	Autor:	N/A
Versão:	1	Prioridade:	Essencial
Descrição:	O sistema deve permitir que o usuário edite um cliente cadastrado		

Identificador	RF003	Categoria:	Cadastro
Nome:	Excluir Cliente		
Data de Criação:	17/11/2024	Autor:	Felipe Mello

Data Alteração:	N/A	Autor:	N/A
Versão:	1	Prioridade:	Essencial
Descrição:	O sistema deve permitir que o usuário exclua um cliente cadastrado		

Identificador	RF003	Categoria:	Interatividade
Nome:	Buscar Cliente		
Data de Criação:	17/11/2024	Autor:	Felipe Mello
Data Alteração:	N/A	Autor:	N/A
Versão:	1	Prioridade:	Essencial
Descrição:	O sistema deve permitir que o usuário busque um cliente cadastrado		

4. Não Funcionais

Identificador	RNF001	Categoria:	Funcionalidade
Nome:	Cadastro de Cliente		
Data de Criação:	17/11/2024	Autor:	Felipe Mello
Data Alteração:	N/A	Autor:	N/A
Versão:	1	Prioridade:	Essencial
Descrição:	O sistema deve estar conectado a um Banco de Dados		

Identificador	RNF002	Categoria:	Funcionalidade
Nome:	Cadastro		
Data de Criação:	17/11/2024	Autor:	Felipe Mello
Data Alteração:	N/A	Autor:	N/A
Versão:	1	Prioridade:	Essencial

Descrição:	O sistema deve possuir um formulário de cadastro de cliente
-------------------	---

Identificador	RNF003	Categoria:	Funcionalidade
Nome:	Tempo de Resposta		
Data de Criação:	17/11/2024	Autor:	Felipe Mello
Data Alteração:	N/A	Autor:	N/A
Versão:	1	Prioridade:	Essencial
Descrição:	O sistema deverá ter um tempo de resposta entre 0,1 a 1 segundo, de maneira a dar agilidade para a aplicação		

Identificador	RNF004	Categoria:	Funcionalidade
Nome:	Proteção		
Data de Criação:	17/11/2024	Autor:	Felipe Mello
Data Alteração:	N/A	Autor:	N/A
Versão:	1	Prioridade:	Essencial
Descrição:	O sistema sempre que possível deverá ter reuso de código		

Identificador	RNF005	Categoria:	Funcionalidade
Nome:	Proteção		
Data de Criação:	17/11/2024	Autor:	Felipe Mello
Data Alteração:	N/A	Autor:	N/A
Versão:	1	Prioridade:	Essencial
Descrição:	O código deve ter tecnologia atualizada para manter a segurança dos dados informados		

5. Bibliografia

- PRESSMAN, R.S.; MAXIM, B.R. Engenharia de software: uma abordagem profissional. 8. ed. Porto Alegre: McGraw Hill, 2016. 968 p. ISBN 9788580555332.
- HERRINGTON, J.D. PHP Hacks: Dicas e ferramentas para criação de websites dinâmicos. Porto Alegre: Bookman, 2007. 424 p. ISBN: 9788577802135.
- ALBUQUERQUE, Rafael, Arquitetura de Sistemas, Modelagem Ágil