



**CENTRO DE CIÊNCIAS E TECNOLOGIAS
LABORATÓRIO DE CIÊNCIAS MATEMÁTICAS**

Davi Rodrigues Soares Machado

**Relatório:
Sistema de Rede Escolar**

Campos dos Goytacazes - RJ
26 de novembro de 2025

Sumário

1	Arquitetura Final do Sistema	2
1.1	Visão Geral da Arquitetura	2
1.2	Componentes Principais	2
1.2.1	Frontend	2
1.2.2	Backend	2
1.2.3	Banco de dados	3
1.3	Considerações	3
2	Cronograma Efetivamente Cumprido	4
2.1	Tabela de Cronograma	4
2.2	Resumo Mensal	5
3	Descrição de Desafios Encontrados	7
4	Conclusão	8

Capítulo 1

Arquitetura Final do Sistema

1.1 Visão Geral da Arquitetura

A arquitetura final do sistema foi planejada seguindo o modelo em camadas, integrando a interface desenvolvida com React/Next.js com um backend estruturado e conectado a um banco de dados PostgreSQL.

O objetivo principal era permitir o gerenciamento das informações escolares de alunos, professores e demais usuários dentro da Rede Escolar.

A comunicação entre backend e frontend foi realizada usando API REST.

1.2 Componentes Principais

1.2.1 Frontend

O principal componente do frontend é o next, que serviu para renderizar e otimizar as rotas entre as pages. Já o react, permitiu a criação de componentes reutilizáveis do sistema

Principais Páginas e Módulos A ideia principal consistia em criar uma home page central, com informações da rede de escolas, e uma pagina destinada a cada tipo de usuário (alunos, professores e secretários).

1.2.2 Backend

O backend foi desenvolvido com Python e FastAPI, sendo o local responsável pela parte lógica do sistema, o tratamento de autenticação de usuário e a consistencia dos dados que iriam para o banco de dados.

As requisições vindas dos usuários pelo frontend eram processadas e direcionadas de volta através de respostas em JSON

1.2.3 Banco de dados

O modelo inicial do banco de dados consistia em uma classe geradora Usuário, e suas subclasses como Alunos, Professores e Secretários. No decorrer do projeto, foi visto a necessidade de criar tabelas direcionadas as notas dos alunos.

1.3 Considerações

O andamento do desenvolvimento sofreu mudanças, que serão abordadas na Sessão Resumo Mensal, sendo comparado as atividades feitas com as atividades propostas.

Capítulo 2

Cronograma Efetivamente Cumprido

2.1 Tabela de Cronograma

Calendário de Entrega do Sistema							
Setembro				Outubro			
Semana 2 (07/09 - 13/09)	Semana 3 (14/09 - 20/09)	Semana 4 (21/09 - 27/09)	Semana 5 (28/09 - 04/10)	Semana 1 (05/10 - 11/10)	Semana 2 (12/10 - 18/10)	Semana 3 (19/10 - 25/10)	Semana 4 (26/10 - 01/11)
Planejamento e Levantamento de requisitos	Apresentação da Proposta e Arquitetura inicial	Apresentação da Documentação do Projeto	Ajustes na modelagem e na documentação do sistema	Implementação do Backend da aplicação			
Reunião com Cliente	Semana P1 Pesquisa Operacional		Semana P1 Compiladores				Reunião com Cliente
			Criação do Protótipo				
			Reunião com Cliente				

Figura 2.1: Cronograma do Primeiro Bimestre de Desenvolvimento

Calendário de Entrega do Sistema					
Novembro				Dezembro	
Semana 1 (02/11 - 08/11)	Semana 2 (09/11 - 15/11)	Semana 3 (16/11 - 22/11)	Semana 4 (23/11 - 29/11)	Semana 1 (30/12 - 05/12)	Semana 2 (07/12 - 13/12)
Implementação do Frontend da aplicação				Testes Funcionais e de uso	Demonstração do software e documentação do sistema
<i>Semana Projeto Compiladores</i>				<i>Semana P2 Compiladores</i>	
<i>Semana P2 Pesquisa Operacional</i>		Reunião com Cliente		<i>Semana P3 Pesquisa Operacional</i>	

Figura 2.2: Cronograma do Segundo Bimestre de Desenvolvimento

2.2 Resumo Mensal

Toda a comparação vai se seguir utilizando do github, onde foi registrado cada uma das alterações.

- **SETEMBRO:** No mês de Setembro, o principal foco era elucidar o que seria em si o sistema, quais as funcionalidades e conferir quais os requisitos que seriam essenciais. Na primeira semana, tudo ocorreu como o planejado no cronograma. Houve a primeira reunião e a apresentação do que consistia o projeto, com orçamento e tecnologias que seriam utilizadas. Nesse primeiro mês, o foco principal aconteceu na documentação que precisou passar por mudanças, deixando a quinzena final do mês para a prototipagem do que seriam as classes principais do sistema.
- **OUTUBRO:** O Mês de outubro, segundo o cronograma seria para a implementação do backend e ao final, ter mais uma reunião para retificar o andamento do projeto. No decorrer do mês, houveram problemas pessoais que fizeram com que o desenvolvimento do backend fosse drasticamente afetado, atrasando a entrega do mesmo em 3 semanas.

Tendo uma mudança significativa somente ao fim do mês, onde o início do frontend começou a ser desenvolvido, na esperança de retomar ao cronograma, algumas dificuldades foram aparecendo. O resultado por trás do atraso no backend foi postergar o desenvolvimento dos principais métodos e modulos das classes. A decisão de migrar pro frontend e começar um rascunho do que seria o projeto final fez com que o desenvolvimento dos módulos fosse deixado em segundo plano. Por fim, a reunião que deveria acontecer ao final do mês não aconteceu, deixando ainda mais o desenvolvimento do sistema, distante do requerido pelo cliente.

- **NOVEMBRO:** Novembro tinha como objetivo terminar o frontend e começar os testes funcionais do sistema, na ultima semana. Novamente, o desenvolvimento foi prejudicado, dessa vez em vista das provas e de entregas de trabalhos, principalmente matérias com conteúdos pesados, como pesquisa operacional e compiladores. No decorrer do mes de novembro, integrarmos o render no projeto, ferramenta que estabiliza e conecta a comunicação entre back e frontend.

Os principais commits desse mes foram voltados para a parte de credenciais de usuários, mas a comunicação entre o banco de dados estava sendo quebrada dado ao mau uso da API de comunicação. A alternativa foi voltar a comunicação em um ambiente interno virtual, onde atualmente acontece a validação das credenciais dos usuários.

A reunião prevista com o cliente não aconteceu da forma planejada, sendo demonstrado o estado atual do sistema somente no dia 26 de novembro, onde pude ver que colocar as funções e módulos do sistema em segundo plano, deixasse o sistema somente no esboço, dado que o foco principal foi colocado de lado. As Páginas dos usuários foram visualmente feitas, mas como os módulos nao estavam implementados, ficaram sem funcionalidades.

- **DEZEMBRO:** Dezembro seria o mês em que o projeto seria entregue completo, sendo o dia 03 o dia final da apresentação do sistema. Como dito anteriormente, o sistema ainda não esta finalizado e depende dessa ultima semana, que seria antes para correção de bugs e testes de uso, mudando para novamente os primórdios do backend: os módulos e funções de cada classe. Essencialmente o sistema é oco, visualmente completo mas internamente vazio. A pretenção é fazer o máximo proposto ate o dia final, para que as expectativas do cliente sejam minimamente supridas.

Capítulo 3

Descrição de Desafios Encontrados

- Integração entre Backend,Frontend e Banco de dados: Acredito que pela falta de experiência, uma das maiores dificuldades foi a comunicação entre esses 3 componentes. A solução foi pesquisar ferramentas que permitiriam facilitar essa comunicação. Primariamente, escolhi usar o render mas não tive total familiaridade, então optei por fazer um ambiente virtual usando python, que através do unicorn, consigo gerar uma comunicação local sem muitos problemas.
- Dificuldades com tempo de desenvolvimento: Sem dúvidas, o tempo foi o maior fator dificultador. O cronograma foi feito baseado nas semanas, levando em consideração entregas e provas de outras disciplinas, mas mesmo com isso, não foi como previsto. O acúmulo de tarefas que foram postergadas acabou impedindo que o projeto andasse da forma com que foi previsto. O resultado foi o não cumprimento do cronograma e o atraso do desenvolvimento dos módulos.
- Conhecimento prévio: A construção do sistema não levava em conta o desconhecimento ou a falta de experiência em tecnologias como o github, a programação orientada à objetos, que além de precisar revisar os conceitos e usos, nunca tinha sido feita por mim na linguagem python. Acredito que boas horas foram dedicadas ao estudo e preparo antes mesmo do desenvolvimento, para a familiarização dessas tecnologias.
- Má elaboração do banco de dados: Ao criar o banco de dados, percebi que o diagrama feito não levava em consideração o armazenamento de dados que seriam usados dentro das funções e módulos, como notas dos alunos e como isso seria projetado. Ainda estamos vendo a viabilidade da implementação no sistema e das alterações nos diagramas.

Capítulo 4

Conclusão

O desenvolvimento, mesmo que parcial, do sistema foi e tem sido um bom desafio. Unir o conhecimento de várias disciplinas da faculdade, que por vezes não temos o tempo de por em prática cria uma bagagem que futuramente acredito ser essencial para o desenvolvimento de futuros sistemas.

Em grande parte da faculdade vemos o desenvolvimento dos sistemas baseado em metodologias pouco práticas no dia a dia, como o desenvolvimento cascata. Ná prática, essa metodologia por mais segura que seja, impede que o desenvolvimento dinâmico aconteça, tornando tarefas que poderiam ser feitas em horas, levarem dias, dado a necessidade da espera da etapa anterior ter sido concluída.

O estudo dos requisitos de sistema foi crucial para que a ideia que o cliente tinha fosse estruturada da forma mais real, colocando ambos os lados (cliente e desenvolvedor) em consentimento.

Por fim, acredito que essa tenha sido uma experiência marcante na universidade, definitivamente a complexidade de um sistema vai muito alem do que imaginamos, e a necessidade de novos componentes vem de acordo com a demanda do cliente, mostrando a importancia que a comunicação entre ambos os lados tem para o desenvolvimento.