

## PRE-MAAP

Aelson Carvalho Fernandes<sup>1</sup>, Edmilson Almeida Gandra<sup>2</sup>, Kaike Marcelo da Cruz Silva<sup>3</sup>, Vitor Carlos Loureiro Santos<sup>4</sup>, Vinicius Oliveira Nepomuceno<sup>5</sup>

<sup>1</sup> á <sup>4</sup> Discentes do curso de ADS, IFBA (Campus Eunápolis) Av. David Jonas Fadini, S/N, Rosa Neto – CEP 45823-431 – Eunápolis – BA

<sup>5</sup>Professor do curso de ADS, orientador do projeto interdisciplinar, IFBA (Campus Eunápolis) Av. David Jonas Fadini, S/N, Rosa Neto – CEP 45823-431 – Eunápolis – BA

**Resumo:** *O objetivo deste projeto é oferecer um software de prontuário eletrônico em nuvem para consultórios e clínicas médicas, com uma proposta de valor que se baseia na praticidade e segurança na gestão de dados dos pacientes. Além da venda do software, a empresa oferecerá serviços de treinamento e capacitação para os clientes e cobrará uma assinatura mensal que inclui manutenção e suporte técnico. O público-alvo são consultórios e clínicas médicas de pequeno e médio porte, e as principais fontes de receita são a venda do software e a cobrança de assinaturas mensais. Para isso, a empresa contará com recursos como tecnologia de ponta, parcerias com empresas de hospedagem em nuvem e uma equipe qualificada para o desenvolvimento e suporte do produto.*

## Relatório de Condução

Após a primeira apresentação do Projeto Integrador (P.I.) em 25 de setembro de 2023, o grupo realizou uma reunião estratégica para planejar a reestruturação do projeto, visando atender às sugestões dos avaliadores. Durante esse processo, identificamos algumas problemáticas fundamentais que necessitavam de abordagem:

- **Classe Models:** As classes ``UserModel`` representavam objetos de dados, mas estavam sobrecarregadas com atributos generalizados para diferentes tipos de usuários. Por exemplo, um usuário designado como paciente possuía atributos como "CRM", que é um registro associado exclusivamente aos médicos.
- **Permissões versus Tipos de Usuário:** Inicialmente, estávamos criando apenas permissões e utilizando o atributo `'role'` para distinguir os usuários. Optamos por migrar para uma abordagem de tipos de usuário, dividindo as classes de usuário em cinco categorias: ``UserModel``, ``PatientModel``, ``DoctorModel``, ``CollaboratorModel`` e ``AdminModel``. A classe ``UserModel`` continha atributos comuns a todos os tipos de usuários e era referenciada pelas subclasses.
- **Mudança de API para MVC:** Recebemos a recomendação de migrar de uma abordagem inicial baseada em *API* para adotar o padrão *Model-View-Controller* (MVC). Essa mudança significativa demandou semanas de estudo do framework *SpringBoot* para realizar as alterações necessárias.

Durante o processo de reestruturação, enfrentamos desafios significativos ao separar os tipos de usuário. A autenticação de usuários, agora em entidades distintas, tornou-se uma tarefa complexa. Após concluir os métodos de autenticação e garantir seu funcionamento adequado, passamos a desenvolver as demais funcionalidades do sistema.

Kaike Marcelo focou nos métodos relacionados a atestados, receitas e laudos médicos do paciente, enquanto Vitor Carlos testava o sistema em andamento para identificar erros e realizar correções. Aelson Carvalho, responsável pelo desenvolvimento dos métodos relacionados aos usuários, ofereceu suporte constante a Kaike, que enfrentou desafios na implementação de partes lógicas do código.

Edmilson Almeida concentrou-se no desenvolvimento do *Frontend*, utilizando inicialmente a ferramenta *NodeJS*. No entanto, após a recomendação de um avaliador, optamos por utilizar apenas *HTML* puro para simplificar o desenvolvimento. A transição de *API* para *MVC* introduziu complicações na interação entre *Frontend* e *Backend*, resultando na decisão de envolver Kaike tanto no *Front* quanto no *Back*.

Antes da entrega, o *Backend* estava funcional e atendia a todos os requisitos estabelecidos para a primeira fase do projeto. Contudo, problemas surgiram quando enfrentamos dificuldades para obter informações do usuário autenticado no *Frontend*. A utilização do *Thymeleaf* para fazer essas requisições não estavam atendendo às expectativas, desencadeando uma série de desafios técnicos.

Apesar dos esforços da equipe para contornar os contratempos, incluindo a implementação de alternativas no *Backend*, a autenticação dos tipos de usuários e o controle de acesso não foram totalmente eficazes até a segunda apresentação do P.I. em 13 de novembro de 2023. Embora tenhamos alcançado sucesso em aspectos como a inserção e atualização de usuários no banco de dados, reconhecemos que alguns requisitos não foram atendidos conforme inicialmente planejado. Este cenário motivou a equipe a buscar soluções adicionais visando uma entrega mais completa e eficiente.

O aluno Caio Almeida Rodrigues, que era um dos componentes da equipe do PREMAAP, foi demitido por não realizar nenhuma das tarefas proposta pela equipe. Tentamos dar muitas oportunidades em realocá-lo de posição para evitar essa decisão, mas ele não cumpriu nenhuma das atividades.