

**Future Tech**

Breno de Souza Silva - RM88332

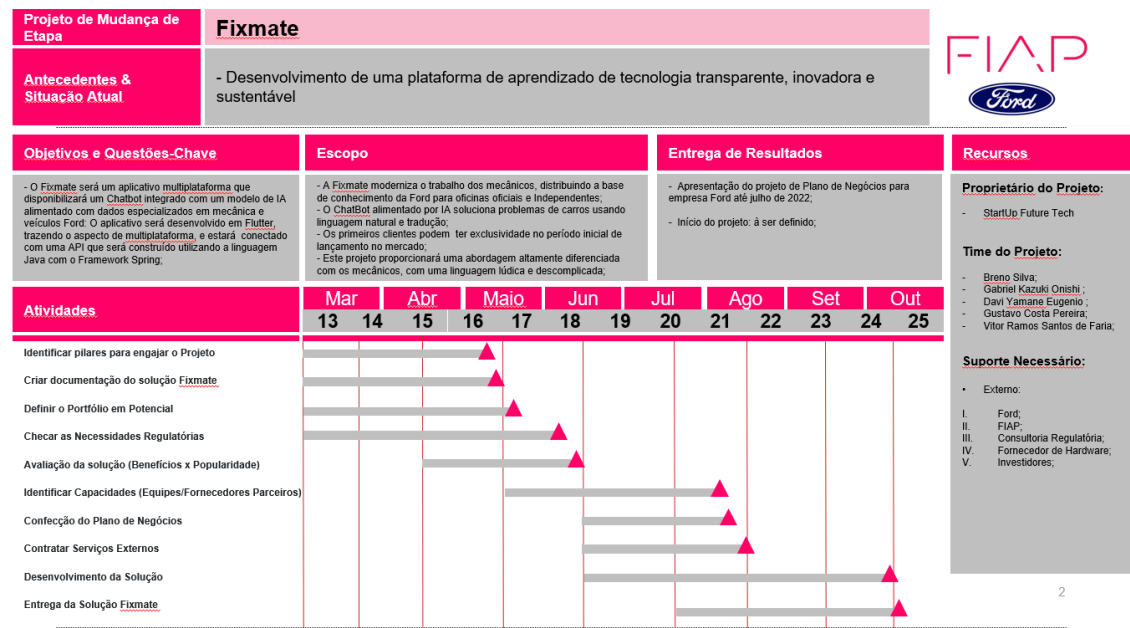
Davi Yamane Eugenio - RM87221

Gabriel Kazuki Onishi - RM87182

Gustavo Costa Pereira - RM87843

Vitor Ramos Santos de Faria - RM88000

Cronograma de atividades planejado



Jornada do projeto

Nos próximos parágrafos será apresentada a jornada que o grupo percorreu durante o desenvolvimento do projeto, comentando as dificuldades passadas, soluções encontradas, planejamentos e pontos de melhoria.

O planejamento do projeto se deu início no dia 08/05, no dia em que o kick-off da Challenge ocorreu, apresentando a empresa Ford, os temas disponíveis para desenvolvimento da solução e outras informações úteis para o prosseguimento das primeiras entregas.

Durante essa primeira semana o grupo se reuniu em diversos momentos para discutir e levantar ideias para o desafio proposto, de forma que foram realizados diversos brainstorms. Na qual dentre os temas disponibilizados: Natureza do trabalho, Mobilidade do Futuro e Gestão de Produtos, houve conversas para decidir qual seria o foco do grupo, nos levando inicialmente à Gestão de Produtos, com uma ideia de Blockchain.

Ela tentaria resolver o problema de transparência e incentivaria a criação de carros utilizando peças nativas dos países alvo das vendas, de forma que os clientes poderiam ter acesso a um portal que seria capaz de exibir as preocupações da empresa com sustentabilidade e a origem das peças. O que incentivaria os clientes a comprar carros Ford no Brasil, já que falta de peças não seria mais um problema.

Porém ao analisarmos melhor a viabilidade, retorno de investimento para a Ford e custos necessários, percebemos que a solução envolveria um valor muito elevado para que chegasse ao ponto de trazer algum retorno, além de que o sistema de rastreamento de peças do mundo automobilístico se apresentava como um sistema extremamente complexo e com soluções já utilizadas por diversas empresas.

Pensando no tema de Mobilidade do Futuro, levantou-se a ideia de criar um aplicativo de compartilhamento de rotinas, um meio de transporte que seria utilizado antes de um Uber ou Taxi, que se baseava em clubes de usuários com destinos e objetivos próximos que poderiam compartilhar uma viagem cobrindo os gastos com manutenção ou combustível do motorista.

Todavia, ao discutir a ideia com terceiros que se encaixavam como público-alvo e com pessoas mais experientes, percebemos que da mesma forma que a ideia anterior não apresentava um retorno muito atrativo se comparado com os custos e desafios envolvidos, também pensando no tamanho do projeto até que o objetivo do aplicativo fosse cumprido, chegamos à conclusão de que precisávamos mudar nossa forma de pensar para alcançar uma solução mais viável e interessante.

Na situação em que o grupo se encontrava, já era dia 25/05 e ainda não tínhamos uma ideia concreta para a entrega do dia 12/06. Em paralelo também havia se iniciado o período de Global Solution na FIAP no dia 24/05, em que os trabalhos semestrais deveriam ser entregues até o dia 07/06. Contexto que dificultou o investimento de tempo no projeto da Challenge.

Então nesse mesmo dia conversamos com o professor José Romualdo da Costa Filho, da matéria de Microservice And Web Engineering, que além de nos ajudar a perceber pontos de melhoria e nos guiar nas implementações, ajudou o grupo a planejar uma próxima ideia, porém com uma base mais concreta e viável, nos indicando dados que poderiam trazer relevância à nossa próxima solução.

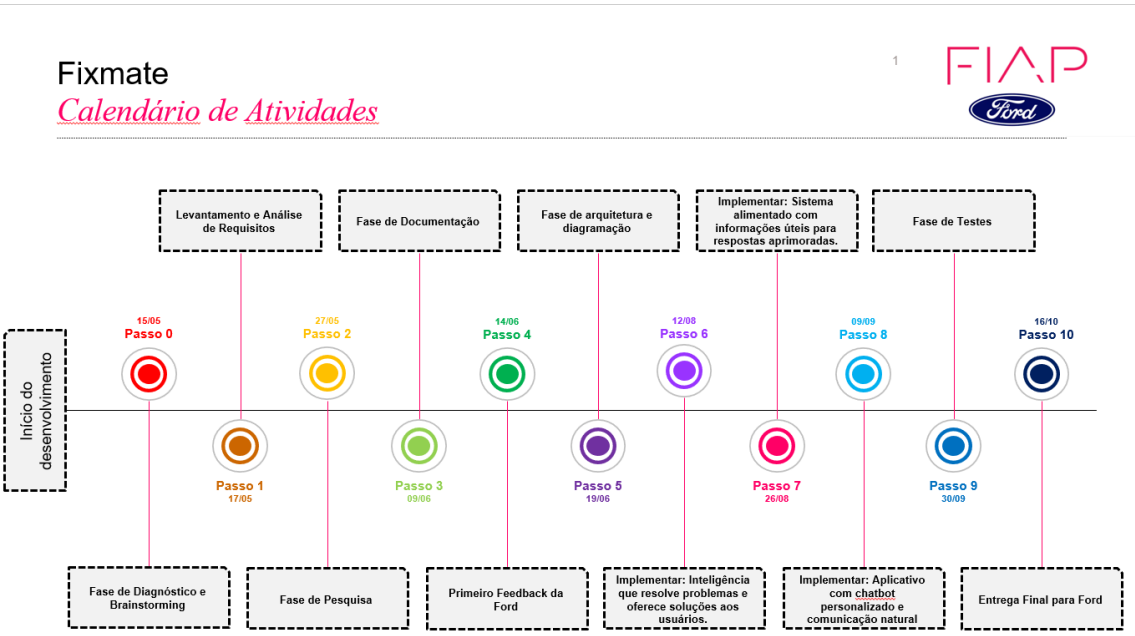
Dessa maneira, no sábado dessa mesma semana nos organizamos com a ajuda do professor para a nossa ideia atual, o Fixmate. Após esse dia basicamente focamos na Global Solution, retornando

ao desenvolvimento da solução no dia 09/06, em que passamos a nos concentrar totalmente no projeto e na entrega da primeira Sprint.

Esse primeiro documento nos ajudou a formar uma ideia mais concreta de nossa solução, agrupando todos os dados pesquisados anteriormente e trazendo uma visão inicial da arquitetura a ser usada na nossa solução, com as tecnologias envolvidas, diferenciais competitivos e entre outros aspectos.

Com essa primeira etapa concluída no dia 12/06, buscamos então feedbacks com a empresa Ford e professor José, resultando em uma visão positiva e encorajadora para seguir com o Fixmate para a entrega da Sprint 2, em que montamos uma visão mais detalhada e específica dos requisitos necessários, hardware, arquitetura com os diagramas de banco de dados e TOGAF, também um protótipo que será apresentado em um vídeo para a Ford.

Perspectivas até a próxima entrega



Requisitos do projeto

Requisitos Funcionais e Não Funcionais

1 – Lista de Requisitos de Usuários

| Requisitos de Usuários | |
|---|---|
| Requisitos Funcionais | Requisitos Não Funcionais |
| RF001 – O sistema deverá permitir que o mecânico realize seu cadastro. | RNF001 – O sistema deverá ser desenvolvido para dispositivos móveis. |
| RF002 – O sistema deverá permitir que o usuário recupere sua senha | RNF002 - O sistema deverá ser desenvolvido para plataforma Web. |
| RF003 – O sistema deverá permitir que o usuário inicie uma nova conversa | RNF003 – O sistema deverá utilizar criptografia para cifrar as senhas cadastradas. |
| RF004 – O sistema deverá permitir que o usuário visualize o histórico de conversas já realizadas | RNF004 – O sistema deverá validar e autenticar as informações fornecidas no cadastro. |
| RF005 – O sistema deverá permitir que o usuário visualize uma tela com suas informações de cadastro | RNF005 – O sistema deverá ter capacidade de resposta em tempo real. |
| RF006 – O sistema deverá responder perguntas comuns | RNF006 – O sistema deverá ter gerenciamento de contexto. |
| RF007 – O sistema deverá orientar passo a passo diante de uma dúvida do mecânico | RNF007 – O sistema deverá conter monitoramento e controle de conteúdo. |
| RF008 – O sistema deverá identificar problemas descritos pelos mecânicos | RNF008 – O sistema deverá ter processamento de linguagem natural avançado e idiomas. |
| RF009 – O sistema deverá sugerir soluções | RNF009 – O sistema deverá ter integração com outras APIs e serviços. |
| RF010 – O sistema deverá acessar informações de manuais e documentações técnicas | RNF010 – O sistema deverá possuir métricas e análise. |

2 – Diante dos requisitos funcionais e não funcionais, temos os requisitos do sistema:

| Requisitos de Sistema - Funcionais | |
|---|---|
| RF001 | 1- O cadastro do mecânico deverá conter os campos de dados pessoais e dados funcionais. |
| | 2- Na aba de cadastro pessoais deverá ter os seguintes campos: E-mail, Nome, Idioma preferencial. |
| | 3- Na aba de dados funcionais deverá conter os seguintes campos: Nome da Instituição que trabalha, telefone(s) de contato, cargo/função, endereço e CNPJ. |
| | 4- Os dados funcionais e pessoais são obrigatórios. |
| RF002 | 1- O sistema deverá utilizar o e-mail cadastrado para recuperação da senha. |
| | 2- O sistema deverá gerar uma nova senha e enviar ao usuário por e-mail |
| RF003 | 1- Ao iniciar uma nova conversa, a interface deverá permitir que o usuário insira uma nova pergunta. |
| RF004 | 1- A interface do sistema deverá permitir que seja visualizado as conversas anteriores. |
| | 2- Ao clicar na conversa o sistema deverá permitir que seja visualizado todo o histórico da conversa específica. |
| RF005 | 1- Na aba de usuário, a interface deverá disponibilizar as informações de cadastro do usuário. |
| RF006 | 1- O sistema deverá responder a perguntas frequentes dos mecânicos. |
| | 2- O sistema deverá informar procedimentos de manutenção. |
| | 3- O sistema deverá responder perguntas sobre diagnósticos de problemas e especificações técnicas sobre peças. |
| RF007 | 1- O sistema pode fornecer informações e orientação passo a passo para tarefas de reparo ou manutenção. |
| | 2- O sistema deverá conduzir os mecânicos por meio de um processo sequencial, transmitindo informações precisas e claras. |
| RF0008 | 1- O sistema deverá auxiliar os mecânicos na identificação de problemas comuns em peças, equipamentos ou veículos. |
| | 2- O sistema deverá identificar perguntas específicas para entender melhor os problemas relatados e fornecer uma análise precisa do problema. |

| | |
|---|---|
| RF009 | 1- Com base nas informações fornecidas pelos mecânicos, o sistema deverá sugerir soluções para os problemas mencionados. |
| | 2- O sistema pode incluir recomendações de procedimentos de reparo, encaminhamento para uma agência parceira ou trocas de peças. |
| RF010 | 1- O sistema pode fornecer acesso a manuais de serviços, esquemas elétricos, diagramas técnicos e outras informações relevantes para auxiliar os mecânicos em suas tarefas diárias. |
| Requisitos de Sistema - Não Funcionais | |
| RNF001 | 1- Será utilizado o framework Flutter |
| | 2- O sistema deverá seguir as telas já prototipadas. |
| | 3- O sistema deverá conter os temas “dark e light”. |
| | 4- O sistema deverá fazer permitir que o usuário apague o histórico de perguntas. |
| | 5- O sistema deverá permitir que o usuário exporte os detalhes da conta e conversas para o e-mail que foi utilizado no cadastro da conta em forma de um arquivo para download. |
| RNF002 | 1- Será utilizado o framework Flutter |
| | 2- O sistema deverá seguir as telas já prototipadas. |
| | 3- O sistema deverá conter os temas “dark e light”. |
| | 4- O sistema deverá fazer permitir que o usuário apague o histórico de perguntas. |
| | 5- O sistema deverá permitir que o usuário exporte os detalhes da conta e conversas para o e-mail que foi utilizado no cadastro da conta em forma de um arquivo para download. |
| RNF003 | 1- O sistema deve proteger as senhas dos usuários tornando-as ilegíveis por meio de um processo chamado criptografia. |
| RNF004 | 1- O sistema deve verificar se as informações fornecidas pelos usuários durante o cadastro são corretas e legítimas. |
| RNF005 | 1- O sistema deve ser capaz de responder às consultas dos usuários de forma rápida e eficiente, minimizando qualquer atraso perceptível. |
| RNF006 | 1- O sistema deve ser capaz de lembrar o contexto da conversa, permitindo que os usuários continuem a interação de onde pararam e evitando repetições desnecessárias. |
| RNF007 | 1- O sistema deve ter mecanismos para monitorar e controlar o conteúdo gerado pela |

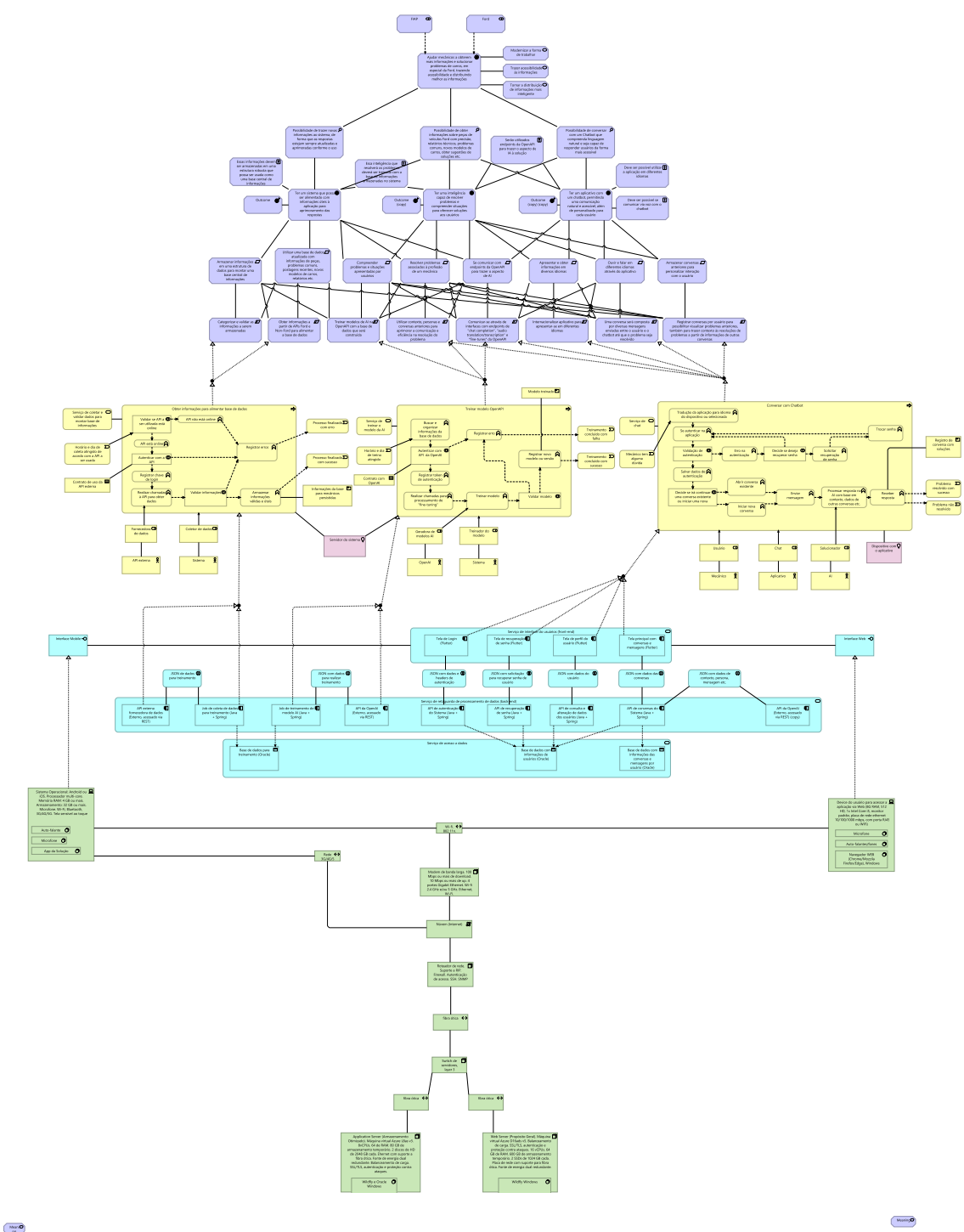
| | |
|--------|--|
| | IA garantindo que ele não produza respostas inadequadas, ofensivas ou enganosas. |
| RNF008 | 1- O sistema deve ser capaz de compreender e interpretar a linguagem natural dos usuários e outros idiomas de forma precisa e contextual, permitindo uma interação mais natural e fluente. |
| RNF009 | 1- O sistema deve permitir a integração com APIs e serviços externos, possibilitando a obtenção de informações atualizadas e a execução de ações específicas com base nas solicitações dos usuários. |
| RNF010 | 1- O sistema deve ser capaz de coletar métricas e dados de análise sobre o desempenho da IA permitindo avaliar sua eficácia e identificar áreas de melhoria. |

Planejamento de uso de software

Para o funcionamento do projeto, planejamos o uso dos seguintes componentes:

- Linguagens de Programação
 - Java – Será utilizado no back-end da aplicação para as APIs do sistema.
 - Dart – Será utilizado no front-end da aplicação.
- Frameworks
 - Spring – Será utilizado juntamente do Java para facilitar a construção das APIs e suas configurações.
 - Flutter – Será utilizado juntamente do Dart para trazer o aspecto de multiplataforma e facilitar a construção da interface do usuário.
- Cloud
 - Azure – Dado o suporte e máquinas virtuais disponíveis que a Azure poderá fornecer para a implantação da aplicação, além da parceria entre a FIAP e a Microsoft, planejamos usar a Azure.
 - Para a máquina de aplicação que irá coletar informações para o treinamento do modelo AI - L8as v3: Windows. 8vCPUs. 64 Gb RAM. 80 Gb de armazenamento temporário. 2 discos de HD de 2048 Gb cada.
 - Para a máquina de Web Server que resolverá o Chatbot - D16ads v5: Windows. 16vCPUs. 64 Gb de RAM. 600 Gb de armazenamento temporário. 2 SSDs de 1024 Gb cada.
- Sistema Gerenciador de Banco de Dados
 - Oracle – A mais reconhecida do mercado de bancos relacionais, sendo atualizada e recebendo suporte a todo momento, de forma que outros também passaram a suportar comandos dele e a compatibilizar linguagens, sendo uma escolha segura para a solução.
- APIs
 - OpenAI – Será um componente essencial para o funcionamento da solução, trazendo o aspecto de NLP, resolução de problemas, interpretação de diversos idiomas, além de possibilitar o treinamento de modelos especializados nos dados técnicos para mecânicos.
 - APIs externas para obtenção de dados para treinamento a aprimoramento do modelo:
 - <https://www.carmd.com/api/technical-service-bulletins-tsb-carmd-api/>
 - <https://api.auto-data.net/get-a-quote>
 - <https://api-ninjas.com/api/cars>
 - <https://rapidapi.com/collection/car-api>

Diagrama de arquitetura



O diagrama apresentado acima foi feito utilizando TOGAF no aplicativo Archimate, em que cada cor de bloco, símbolo e conector possui seu significado.

- Os blocos roxos são a visão da arquitetura, em que são apresentados desde os stakeholders, diretrizes e valores até os objetivos, requisitos e restrições de

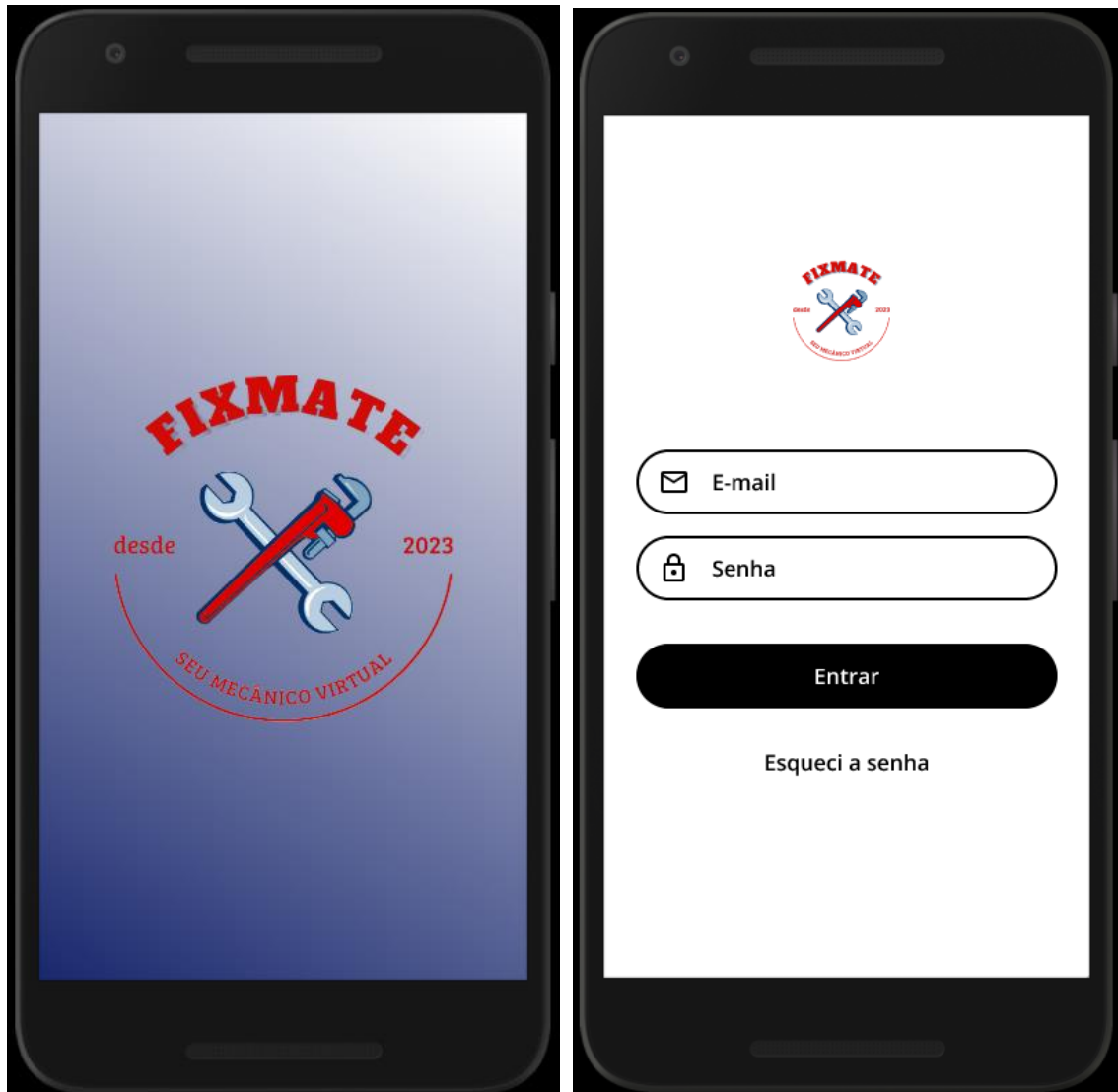
negócio. De forma que a Governança do projeto é garantida, tendo em vista que tudo será desenvolvido a partir dos stakeholders (FIAP e Ford) e seus desejos.

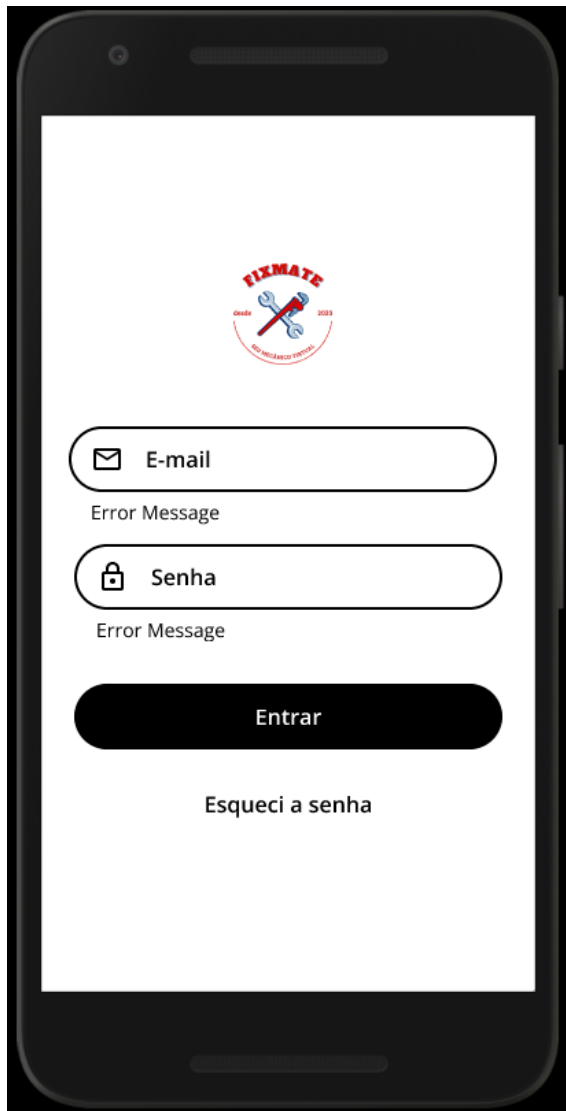
- Os amarelos representam a arquitetura de negócios, representando os processos envolvidos no uso da solução, mostrando os atores, papéis, eventos, funções, produtos e entre outras indicações relacionadas a como o negócio funcionaria.
- A cor azul é utilizada na Arquitetura de Sistemas, que contém visualmente os componentes se comunicando para o funcionamento da solução, exibindo as APIs, Telas, Interfaces, Bancos de dados e entre outras informações.
- Por fim a cor verde representa a Arquitetura de Tecnologia, exibindo os dispositivos, softwares, equipamentos de rede, conectores, redes, servidores e entre outras tecnologias necessárias.

Dessa forma, a partir do diagrama é possível extrair os requisitos, o funcionamento dos processos, os componentes do projeto e até mesmo o hardware necessário, assim representando em detalhes como o aplicativo e seus serviços funcionariam, além de mostrar o tamanho da solução, permitindo estimar custos e tempo com maior precisão.

Protótipos de telas

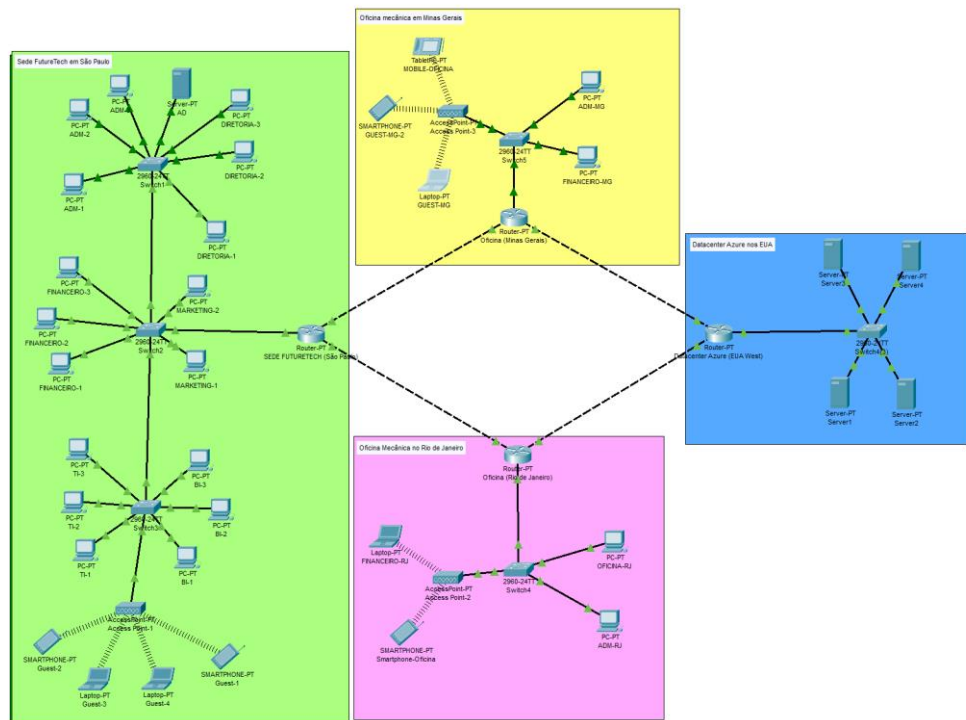
Link do vídeo com as telas.







Infraestrutura de redes



Para a implementação do projeto, é necessário contar com uma infraestrutura robusta e adequada. A área azul destaca a necessidade de um servidor de alta capacidade de processamento, conectado à internet. Esse servidor seria responsável por receber as requisições dos usuários, processá-las e retornar as respostas. É importante que o servidor possua capacidade suficiente para lidar com o volume de solicitações e garantir um desempenho adequado.

Na área verde, refere-se à infraestrutura local da empresa FutureTech, que seria responsável pela administração e desenvolvimento da solução. Nessa área, é essencial contar com os recursos necessários para gerenciar e aprimorar o sistema. Isso inclui servidores adicionais, equipamentos de rede, sistemas de armazenamento e outros recursos tecnológicos relevantes.

A aplicação do protocolo RIP (Routing Information Protocol) nos roteadores de todas as unidades é uma medida importante para permitir a comunicação entre os dispositivos em diferentes locais. O RIP é um protocolo de roteamento dinâmico amplamente utilizado para trocar informações de roteamento entre os dispositivos em uma rede. Ele permite que os roteadores compartilhem informações sobre as redes às quais estão conectados, facilitando o roteamento eficiente de pacotes de dados.

A implantação de ACLs (Access Control Lists) é outra medida relevante para garantir a segurança e restringir o acesso aos dispositivos e recursos. Por exemplo, o servidor AD (Active Directory)

pode ser configurado para permitir o acesso apenas a partir da sede da FutureTech (área verde), garantindo que somente os dispositivos autorizados possam se comunicar com ele.

Essas medidas de segurança e segmentação de rede ajudam a proteger os recursos críticos, restringindo o acesso não autorizado e garantindo um ambiente de trabalho seguro e eficiente. Além disso, a aplicação de VLANs permite separar os departamentos em redes virtuais distintas, facilitando a gestão e o controle de tráfego na rede.

Em resumo, a configuração proposta envolve a implementação de um servidor poderoso para processar as requisições dos usuários, uma infraestrutura local para a FutureTech administrar e desenvolver a solução, a aplicação do protocolo RIP nos roteadores para comunicação entre dispositivos, a utilização de ACLs para segurança e restrição de acesso, e a segmentação da rede em VLANs para uma melhor organização e desempenho. Essas medidas contribuem para a implementação de um sistema eficiente, seguro e escalável.