

Você participou de alguma discussão sobre as interações do usuário com o modelo?

No começo tivemos uma discussão sobre a modelagem dos dados, só que muito incipiente, não ligada diretamente à IA. Não rolou uma discussão com todo time, mas sim discussões parciais com o pessoal de produto, desenvolvimento, etc. Nessa fase ainda estávamos construindo o fluxo do sistema e definindo conceitos, cujo significado seria diferente para cada ator.

Nessas discussões, você sentiu dificuldades durante a interação com o time de IA?

Não senti dificuldades porque, devido à configuração do nosso projeto, o time de IA era uma equipe a parte. Era como se fosse uma “caixa preta” que a gente não precisava se importar. Eu imagino que tenham tido discussões sobre isso, mas para gente o que valeu foi o seguinte: perguntávamos quais dados eles queriam, eles respondiam, e aí incluíamos no formulário para que essa informação fosse preenchida pelo usuário. Mas eu sei que eles, internamente, fizeram uma análise mais detalhada sobre quais dados existem e o que poderia ser analisado. Eu não participei dessas discussões, pois era de outra equipe.

Essas reuniões e as definições que saíram delas foram documentadas?

A parte de desenvolvimento foi documentada. Tivemos várias discussões sobre o que era possível e o que não era e aí o pessoal de produto no final produzia uma tela. E aí através da tela a gente conseguiu modelar o banco de dados. Então no final é como se fosse uma linha de produção. A gente participou das discussões iniciais também, mas nem todo mundo de desenvolvimento participava. É muito custoso todo mundo da equipe participar de reunião, se for assim ninguém trabalha. Nas discussões iniciais, para tentar entender o que o cliente quer e o que é possível ou não, a gente participava, e aí o pessoal de produto junto com o time de UX produzia uma tela que chegava até o time de desenvolvimento.

Como foi definido como o modelo seria consumido?

A gente chegou a discutir isso no começo e ficou decidido que existiria um serviço REST “caixa preta” que permitiria a execução da lógica de IA. Não era do interesse do time de IA saber como os dados estavam sendo persistidos, desde que existisse esse serviço. Recebia, executava, retornava, ponto. Ficou para nós a responsabilidade de decidir como é que faríamos para persistir os dados. Discutimos isso somente dentro do time de DEV. Definimos uma interface de comunicação e cada um fez sua parte.

E você encara que ter essa separação fluiu bem?

Eu diria que sim. É verdade que tivemos alguns problemas relacionados à alteração do contrato estabelecido nos serviços, e também no timing em que cada um dos times realizou alterações. Mas acho que os problemas foram mais na linha da gestão das mudanças de

requisitos e das tarefas do que efetivamente na separação das equipes. Se todos estivessem juntos, poderíamos ter evitado esses problemas, mas teríamos um overhead maior já que todos precisariam estar juntos em todas as reuniões.

Você sentiu falta de uma participação maior de alguém ligado a área de requisitos?

Sim, a parte de requisitos para a gente é muito importante, ainda mais quando você começa a entregar coisas para um cliente. É esperado que as definições do sistema mudem ao longo do projeto. É importante ter uma boa gestão dessas mudanças especialmente em relação à definição do que o cliente quer, o que foi entendido e o que foi passado para o restante da equipe.

Como foi a discussão sobre o armazenamento dos artefatos do modelo?

Não teve essa discussão, e isso gerou problemas. Nós disponibilizamos repositórios no Git para esse armazenamento, mas o time de DEV e o time de IA não discutiram o que seria armazenado e nem como seria armazenado. Isso gerou problemas, porque o modelo é algo muito grande (envolve, por exemplo, os scripts de treinamento) e isso não foi separado, fazendo com que muitos arquivos fossem carregados desnecessariamente toda vez que um novo release fosse gerado.

Houve alguma discussão em relação ao transporte dos dados até o modelo?

Sim, foi discutido. A gente não tinha muita opção, estávamos mais passivos do que ativos. Então não tínhamos como ter uma discussão interna. A discussão foi perguntar ao cliente “como é que você consegue me dar os dados que eu preciso? .csv? Arquivo? Então beleza.” Pegamos o dado do cliente do jeito que ele falou e adaptamos para a demanda do time de IA. Não é o modo ideal, mas sim uma adaptação à realidade.

Isso foi um trabalho conjunto entre o time de DEV e o time de IA?

É, foi. Até certo ponto houve uma discussão, mas nós assumimos. Eram necessárias algumas habilidades que fugiam do perfil do time de IA, então foi mais fácil assumirmos essa responsabilidade do que discutir todo o processo com eles

Essas definições foram documentadas?

Não. A documentação que temos é em e-mails, o que poderíamos resgatar em caso de alguma auditoria.

Como foi a discussão em relação a disponibilizar o modelo?

A gente já tinha mais ou menos o padrão de como seria feita a implantação do modelo de antemão. A gente sabia que o time de IA não tinha uma especialização em DevOps, então

deixamos a estrutura pronta para eles.

A manutenção de esteiras também é responsabilidade do time de DEV?

Sim, o que diz respeito à distribuição do algoritmo de IA é nossa responsabilidade.

Houve alguma discussão sobre atualizar o modelo e garantir alguma forma de aprendizado incremental?

Não participei de nenhuma discussão sobre esse assunto. Não sei dizer se foi feito ou não.

Como foi a discussão em relação ao monitoramento do modelo?

Isso é responsabilidade do time de IA. O serviço que foi gerado a partir do modelo é responsabilidade nossa, visto que se torna uma questão de infra.

Como foi definir os passos necessários para conseguir integrar IA ao restante do sistema?

Foi natural. Nas discussões, foram definidos exatamente os dados necessários, e aí partir daí definimos uma API Rest. Cada time depois seguiu do seu próprio lado.

Sobre a integração, ela foi bem documentada?

Eu diria que sim, foi bem documentado. Usamos a própria definição das APIs para documentar os dados de entrada e saída. Nosso maior problema foi com relação às mudanças. O estado inicial foi bem documentado, cada um sabia o que fazer, mas mudanças foram ocorrendo e essas mudanças não foram bem documentadas, dificultando o alinhamento entre as equipes. A gente foi percebendo à medida que funcionalidades foram quebrando.