

Isabela de Moraes Bertini

**A influência do fator cliente na satisfação dos stakeholders: uma  
análise qualitativa em projetos de software ágil**

São Paulo  
2024

Isabela de Moraes Bertini

**A influência do fator cliente na satisfação dos stakeholders: uma  
análise qualitativa em projetos de software ágil**

**Versão Original**

Monografia apresentada ao PECE – Programa de Educação Continuada em Engenharia da Escola Politécnica da Universidade de São Paulo como parte dos requisitos para a conclusão do curso de MBA em Engenharia de Software.

Área de Concentração: Engenharia de Software

Orientador: Prof. Dr. Fábio Levy Siqueira

São Paulo  
2024

Autorizo a reprodução e divulgação total ou parcial deste trabalho, por qualquer meio convencional e eletrônico, para fins de estudo e pesquisa, desde que citada a fonte.

#### Catálogo-na-publicação

Bertini, Isabela de Moraes

A influência do fator cliente na satisfação dos stakeholders: uma análise qualitativa em projetos de software ágil / I. M. Bertini -- São Paulo, 2024.  
92 p.

Monografia (MBA em Engenharia de Software) - Escola Politécnica da Universidade de São Paulo. PECE – Programa de Educação Continuada em Engenharia.

1.ENGENHARIA DE SOFTWARE 2.MÉTODOS ÁGEIS 3.PESQUISA QUALITATIVA I.Universidade de São Paulo. Escola Politécnica. PECE – Programa de Educação Continuada em Engenharia II.t.

Nome: BERTINI, Isabela de Moraes

Título: A influência do fator cliente na satisfação dos stakeholders: uma análise qualitativa em projetos de software ágil.

Monografia apresentada ao PECE – Programa de Educação Continuada em Engenharia da Escola Politécnica da Universidade de São Paulo como parte dos requisitos para a conclusão do curso de MBA em Engenharia de Software.

Aprovado em:     /     /

Banca Examinadora

Prof(a). Dr(a). \_\_\_\_\_

Instituição: \_\_\_\_\_

Julgamento: \_\_\_\_\_

Prof(a). Dr(a). \_\_\_\_\_

Instituição: \_\_\_\_\_

Julgamento: \_\_\_\_\_

Prof(a). Dr(a). \_\_\_\_\_

Instituição: \_\_\_\_\_

Julgamento: \_\_\_\_\_

## DEDICATÓRIA

*Dedico este trabalho ao meu  
companheiro de vida, Valmir Chitta,  
que há mais de oito anos me  
incentiva a evoluir, mesmo nos  
momentos mais exigentes, quando  
eu achava que não conseguiria  
conciliar trabalho e estudo.*

## **AGRADECIMENTOS**

À Universidade de São Paulo – USP que desde a minha graduação até este MBA sempre me propiciou o contato com conhecimentos sólidos e sempre bem embasados, com um corpo docente de excelência.

À Escola Politécnica da Universidade de São Paulo, EPUSP que propiciou o surgimento e a continuação deste Programa de Educação Continuada, PECE/Poli, permitindo aos alunos ter contato com educadores que são professores doutores da Poli/USP e trazem experiências do mundo real para a sala de aula, promovendo uma aprendizagem contínua.

Ao PECE - Programa de Educação Continuada em Engenharia - pela difusão do conhecimento gerado na Poli através dos Cursos de Extensão Universitária.

Ao meu orientador, Prof. Dr. Fábio Levy Siqueira, minha profunda gratidão pelo cuidado, profissionalismo e pela empatia demonstrada ao longo deste trabalho e das disciplinas ministradas. Sua orientação foi essencial para que eu enfrentasse os desafios e alcançasse este marco, além de ser uma inspiração que me motiva a seguir adiante com os estudos no mestrado.

Ao meu empregador, o banco de investimentos, cujo nome devo omitir, infelizmente. Ao apoio do RH, do meu gestor e do meu time, que me incentivaram e me apoiaram com dicas, ideias e com todo o incentivo que eu precisava.

Aos entrevistados, cujo nome também será necessário omitir, um abraço forte e uma enorme gratidão por me fornecerem seu tempo, compartilharem sua trajetória profissional de forma tão sincera e me fornecerem dados e ideias valiosas para a obtenção dos resultados apresentados a seguir.

Por fim, acima de tudo, expresso meu mais sincero reconhecimento à minha família. A dedicação incansável de vocês, apoio irrestrito, amor e presença constante nos momentos da minha vida foram fundamentais para que eu alcançasse este marco. Nada disso seria possível sem vocês.

## RESUMO

BERTINI, Isabela de Moraes. **A influência do fator cliente na satisfação dos stakeholders: uma análise qualitativa em projetos de software ágil.** São Paulo, 2024. 92 p. Monografia (MBA em Tecnologia de Software). Programa de Educação Continuada em Engenharia da Escola Politécnica da Universidade de São Paulo.

Os fatores críticos de sucesso afetam o desenvolvimento e o resultado de projetos, e são importantes pontos de atenção para o gerenciamento do desenvolvimento de software. Este trabalho explora a influência dos fatores relacionados ao cliente no sucesso de projetos ágeis de software, com foco na satisfação dos *stakeholders*. Baseado no Modelo Ágil de Sucesso de Projetos de Software (BINBOGA e GUMUSSOY, 2024), avaliou-se qualitativamente quatro construtos: suporte do cliente, relacionamento cliente-desenvolvedor, capacitação tecnológica do cliente e *expertise* do cliente em seu domínio. A metodologia incluiu entrevistas semiestruturadas com um desenvolvedor e um cliente da corretora de um banco de investimentos, responsáveis por um software de negociação na bolsa brasileira (B3).

Os resultados indicaram que a proximidade e a troca de conhecimento entre equipes foram fundamentais para o sucesso do projeto, com destaque para a colaboração entre clientes e desenvolvedores, a evolução da capacidade tecnológica dos clientes e a alta experiência dos clientes no mercado financeiro. O software foi avaliado como escalável, intuitivo e seguro, atendendo amplamente às expectativas dos usuários e superando desafios de *performance*. A abordagem qualitativa também evidenciou que a proximidade e o alinhamento entre as equipes de tecnologia - desenvolvedores, redes e infraestrutura - foram cruciais para o resultado satisfatório.

Conclui-se que os fatores do cliente apresentam correlação positiva com a satisfação dos stakeholders, reforçando a importância de aspectos humanos, organizacionais e sociais na implementação de projetos ágeis. O estudo contribui ao evidenciar a relevância da integração cliente-desenvolvedor e sugere o uso de investigações qualitativas para os outros fatores críticos no modelo de sucesso ágil.

**Palavras-chave:** Fatores críticos de sucesso, métodos ágeis, satisfação dos stakeholders, engenharia de software.

## ABSTRACT

BERTINI, Isabela de Moraes. **A influência do fator cliente na satisfação dos stakeholders: uma análise qualitativa em projetos de software ágil.** São Paulo, 2024. 92 p. Monografia (MBA em Tecnologia de Software). Programa de Educação Continuada em Engenharia da Escola Politécnica da Universidade de São Paulo.

Critical success factors affect the development and results of projects and are important points of attention for software development management. The study explores the influence of client-related factors on the success of agile software projects, focusing on stakeholder satisfaction. Based on the Agile Model for Software Project Success (BINBOGA and GUMUSSOY, 2024), a qualitative evaluation was conducted on four constructs: client support, client-developer relationship, client's technological capacity, and client expertise in their domain. The methodology included semi-structured interviews with a developer and a client from a brokerage firm of an investment bank, responsible for a trading software used in the Brazilian stock exchange (B3).

The results indicated that proximity and knowledge exchange between teams were fundamental to the project's success, highlighting collaboration between clients and developers, the evolution of clients' technological capacity, and clients' extensive experience in the financial market. The software was evaluated as scalable, intuitive, and secure, broadly meeting user expectations and overcoming performance challenges. The qualitative approach also revealed that close alignment among technology teams - developers, network, and infrastructure - was crucial to achieving a satisfactory outcome.

It is concluded that client-related factors show a positive correlation with stakeholder satisfaction, reinforcing the importance of human, organizational, and social aspects in implementing agile projects. The study contributes by emphasizing the relevance of client-developer integration and suggests employing qualitative investigations for other critical factors in the agile success model.

**Keywords:** Critical success factors, agile methodologies, stakeholder satisfaction, software engineering.



## LISTA DE ILUSTRAÇÕES

Pág.

Figura 1 – Modelo de sucesso de projetos de software ágil .....	19
---	----

## LISTA DE TABELAS

Pág.

Tabela 1 – Resultado traduzido de Binboga e Gumussoy (2024) .....	15
Tabela 2 – Guia de entrevista a ser utilizado com o cliente .....	26
Tabela 3 – Guia de entrevista a ser utilizado com o desenvolvedor .....	27

## **LISTA DE ABREVIATURAS E SIGLAS**

CSF	Critical Success Factors
BRD	Business Requirement Documents
IA	Inteligência Artificial

## SUMÁRIO

<b>1. INTRODUÇÃO .....</b>	<b>14</b>
1.1 OBJETIVO .....	15
1.2 JUSTIFICATIVAS .....	16
1.3 MÉTODO DE PESQUISA .....	16
1.4 ESTRUTURA DO TRABALHO .....	16
<b>2. REVISÃO BIBLIOGRÁFICA .....</b>	<b>18</b>
2.1 FATORES CRÍTICOS DE SUCESSO DE PROJETO DE SOFTWARE ÁGIL ...	18
2.1.1 Os fatores do cliente .....	20
2.1.2 A satisfação dos stakeholders como métrica de sucesso .....	21
2.2 PESQUISA QUALITATIVA .....	21
2.3 ENTREVISTAS .....	22
2.4 CONSIDERAÇÕES DO CAPÍTULO .....	23
<b>3. ENTREVISTAS .....</b>	<b>24</b>
3.1 ESTRUTURAÇÃO DAS ENTREVISTAS .....	24
3.2 APLICAÇÃO DAS ENTREVISTAS .....	27
3.3 CONSIDERAÇÕES DO CAPÍTULO .....	29
<b>4 ANÁLISE DE RESULTADOS .....</b>	<b>30</b>
4.1 RESULTADOS RECOLHIDOS DA ENTREVISTA DO CLIENTE .....	30
4.1.1 Análise do suporte do cliente ao longo do projeto .....	30
4.1.2 Análise da relação cliente x desenvolvedor .....	31
4.1.3 Análise do domínio do cliente no campo da tecnologia .....	32
4.1.4 Análise do domínio do cliente em seu campo de atuação .....	33
4.1.5 Análise da satisfação do cliente enquanto stakeholder .....	35
4.1.6 Comparação dos resultados do cliente com o Modelo de Sucesso de Projetos de Software Ágil .....	36
4.2 RESULTADOS RECOLHIDOS DA ENTREVISTA DO DESENVOLVEDOR .....	37
4.2.1 Análise do suporte do cliente ao longo do projeto .....	37
4.2.2 Análise da relação desenvolvedor x cliente .....	38
4.2.3 Análise do domínio do cliente no campo da tecnologia .....	39
4.2.4 Análise do domínio do cliente em seu campo de atuação .....	40
4.2.5 Análise da satisfação do desenvolvedor enquanto stakeholder .....	40

<b>4.2.6 Comparação dos resultados do desenvolvedor com o Modelo de Sucesso de Projetos de Software Ágil .....</b>	<b>41</b>
4.3 AMEAÇAS A VALIDADE .....	42
4.4 CONSIDERAÇÕES DO CAPÍTULO .....	42
<b>5. CONSIDERAÇÕES FINAIS .....</b>	<b>43</b>
5.1 CONTRIBUIÇÕES DO TRABALHO .....	44
5.2 TRABALHOS FUTUROS .....	44
<b>REFERÊNCIAS .....</b>	<b>46</b>
<b>APÊNDICE A – TRANSCRIÇÃO DA ENTREVISTA DO CLIENTE .....</b>	<b>47</b>
<b>APÊNDICE B – TRANSCRIÇÃO DA ENTREVISTA DO DESENVOLVEDOR .....</b>	<b>69</b>

## 1. INTRODUÇÃO

Os projetos ágeis possuem taxas de sucesso até três vezes maiores do que os projetos tradicionais (também identificados como projetos cascata), como indicado pelo recente relatório Chaos do Standish Group (JOHNSON, 2021). Por outro lado, as taxas de insucesso dos projetos cascata são duas vezes maiores do que as dos projetos ágeis.

Os dados acima ressaltam a importância de entender os fatores críticos de sucesso (CSFs, do inglês *Critical Success Factors*) para projetos ágeis. Tais fatores afetam o desenvolvimento e o resultado desses projetos, e são importantes pontos de atenção para o gerenciamento do desenvolvimento de software. Em termos técnicos, segundo Milosevic e Patanakul (2005), os CSFs dos projetos ágeis referem-se às características, condições ou variáveis que possam impactar o sucesso do projeto quando adequadamente gerenciados e mantidos.

Partindo desse embasamento, o Modelo Ágil de Sucesso de Projetos de Software (BINBOGA E GUMUSSOY, 2024) se propõe a explicar as relações entre CSFs ágeis e medidas de sucesso de projetos de software. Nesse estudo as autoras investigam como os fatores críticos de sucesso podem afetar o desempenho e o resultado esperado em diversas óticas de um projeto de software ágil.

Os resultados encontrados mostraram que fatores relacionados ao cliente, como o suporte do cliente ao longo do projeto, a boa relação com o cliente, a experiência do cliente em seu domínio de atuação e a familiaridade do cliente no domínio de tecnologia foram destacados como essenciais para o sucesso e para a satisfação dos stakeholders no projeto de desenvolvimento.

Esses resultados mencionados encontram-se na Tabela 1, retirada dos resultados de Binboga e Gumussoy (2024). Nos resultados de estimativas do modelo estrutural apresentados abaixo, os efeitos de cada fator crítico de sucesso - fatores do cliente, fatores do time, fatores organizacionais, fatores do processo ágil, fatores tecnológicos e fatores do projeto - são relacionados às medidas de sucesso do

projeto ágil (eficiência do processo, qualidade e sustentabilidade do software e satisfação do stakeholder).

Nela, pode-se perceber o construto com maior índice de influência na satisfação dos stakeholders são aqueles relacionados aos fatores do cliente. Como ressaltam as autoras: “o fator cliente é um forte preditor da satisfação das partes interessadas” (Binboga e Gumussoy, 2024, p. 95625, traduzido), ou seja, os fatores do cliente representam a maior variância e influência na satisfação dos stakeholders.

	Eficiência do processo	Qualidade e sustentabilidade do software	Satisfação dos stakeholders
$R^2$ (%)	0,51	0,39	0,51
Fatores do cliente	0,27**	0,30**	0,38**
Fatores do time	0,27**	0,09*	0,27**
Fatores organizacionais	0,10*	-0,02	0,10*
Fatores do processo ágil	0,36**	0,41**	0,33**
Fatores tecnológicos	0,21**	0,12**	0,05
Fatores do projeto	-0,27**	-0,16**	-0,19**
*: $p < 0,05$ ;      **: $p < 0,01$ .			

Tabela 1: Resultado de Binboga e Gumussoy (2024), referente às estimativas do modelo estrutural.

## 1.1 Objetivo

O objetivo deste trabalho é avaliar qualitativamente a relação positiva entre o forte suporte do cliente durante o projeto, o bom relacionamento com o cliente, a melhora das capacidades profissionais do cliente no domínio de tecnologia e a experiência do cliente em seu próprio domínio, com a satisfação dos stakeholders.

Isso será feito através de uma abordagem qualitativa, com entrevistas estruturadas, para capturar com profundidade esse resultado. Esse objetivo segue o proposto por Binboga e Gumussoy (2024) a fim de verificar a ocorrência de uma das principais relações fator-sucesso.

## 1.2 Justificativa

Como proposto por Binboga e Gumussoy (2024) na conclusão do artigo sobre modelos de sucesso de projetos de software ágil, as abordagens qualitativas, como entrevistas em profundidade, podem ser usadas nos estudos subsequentes para capturar *insights* que podem não ser totalmente capturados apenas por análise quantitativa.

Embora os dados quantitativos apresentados pelo artigo sejam úteis tal como foram descritos, a investigação qualitativa dispõe-se a leitura desses dados quantitativos de forma crítica. Não se trata de retirar o valor intrínseco dos números, em vez disso, a investigação qualitativa tenta decifrar o processo de compilação. Em vez de confiarem nos dados quantitativos como um caminho para descrever com precisão a realidade, tem-se interesse em como as estatísticas se revelam à luz do cotidiano e dos sujeitos do estudo. Analisar as estatísticas e compará-las com o que os sujeitos relatam verbalmente pode ser uma maneira útil de explorar percepções.

## 1.3 Método de Pesquisa

A metodologia utilizada neste estudo foi a pesquisa qualitativa com utilização de entrevistas semiestruturadas. O contexto de aplicação consistiu em analisar qualitativamente a existência de uma correlação positiva entre os “Fatores do cliente” e a “Satisfação dos stakeholders” considerando um projeto específico. Com isso será possível comparar o resultado obtido por Binboga e Gumussoy (2024) com o que os sujeitos relatam verbalmente e explorar suas percepções práticas.

## 1.4 Estrutura do Trabalho

O Capítulo 2, Revisão bibliográfica, apresenta tanto o embasamento teórico a respeito dos fatores críticos de sucesso de um projeto de software ágil, quanto o embasamento da metodologia de pesquisa qualitativa com a utilização de entrevistas.



O Capítulo 3, Entrevistas, apresenta em detalhes a estruturação dos métodos de coleta de dados por meio de entrevistas, a forma de execução das mesmas e seus artefatos, as transcrições das entrevistas.

O Capítulo 4, Análise dos resultados, apresenta a análise dos dados coletados com as entrevistas. Também foi feita uma comparação desses resultados com o Modelo Ágil de Sucesso de Projetos de Software.

O Capítulo 5, Considerações finais, descreve as conclusões, as contribuições deste trabalho e sugestões de trabalhos futuros.

O Apêndice A apresenta a transcrição completa da entrevista com o cliente.

O Apêndice B apresenta a transcrição completa da entrevista com o desenvolvedor.

## 2. REVISÃO BIBLIOGRÁFICA

Nesta seção é apresentado o conceito de fatores críticos de sucesso das metodologias ágeis. Além disso, são discutidas as metodologias clássicas de estudo qualitativo, com foco principal em entrevistas.

### 2.1 Fatores críticos de sucesso de projetos de software ágil

O Modelo Ágil de Sucesso de Projetos de Software proposto por Binboga e Gumussoy (2024) se propõe a explicar as relações entre fatores críticos de sucesso (CSF, do inglês *Critical Success Factors*) para projetos ágeis e medidas de sucesso de projetos de software. O modelo criado pelas autoras investiga os CSFs que afetam o desempenho de projetos ágeis de software.

Utilizando uma Revisão Sistemática da Literatura (SLR), as pesquisadoras identificaram uma série de CSFs, e os organizaram em seis categorias, são elas: cliente, time, organizacionais, processo ágil, tecnológicos e do projeto. Dentre os materiais utilizados na revisão sistemática temos o manifesto ágil, princípios ágeis, o guia Scrum, e artigos, que serviram de base para o desenvolvimento do modelo proposto. Em seguida, exploraram métricas de sucesso para projetos ágeis, que foram agrupadas como: eficiência dos processos, qualidade e sustentabilidade do software e satisfação dos stakeholders. A relação entre os fatores críticos para o sucesso e as métricas que identificam o sucesso em um projeto pode ser vista na Figura 1, adaptada e traduzida a partir de Binboga e Gumussoy (2024).

A análise foi complementada empiricamente por meio de entrevistas individuais e em grupo com profissionais praticantes de metodologias ágeis e a validação final foi feita através da aplicação de questionários a 596 participantes, usuários dessas metodologias.

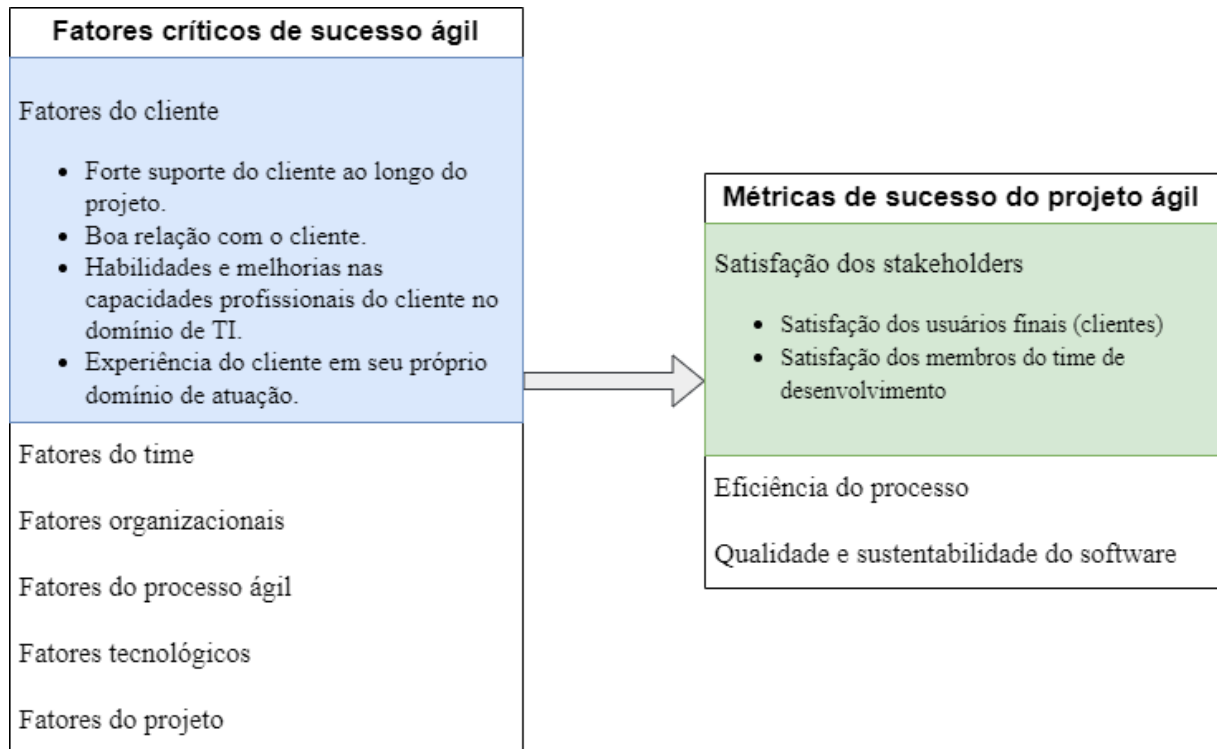


Figura 1: Modelo de sucesso de projetos de software ágil, adaptação de Binboga e Gumussoy (2024).

O modelo desenvolvido permite uma maior compreensão dos desafios e das estratégias de mitigação de riscos em projetos dinâmicos, típicos de metodologias ágeis. No entanto, de acordo com Binboga e Gumussoy (2024), a satisfação dos stakeholders parece estar mais relacionada a melhorias visíveis no produto do que a fatores técnicos isolados, como o uso de tecnologias avançadas.

Fatores de equipe, como motivação, habilidades técnicas e experiência, mostraram uma correlação positiva com a eficiência do processo, a qualidade do produto e a satisfação das partes interessadas. A pesquisa também destacou as diferenças setoriais no uso de metodologias ágeis, com foco particular na indústria financeira, onde o Scrum e o Kanban são amplamente utilizados.

Outro ponto abordado é a importância de uma comunicação eficaz dentro das equipes ágeis, favorecida pelos eventos típicos das metodologias ágeis, como o planejamento de sprints, reuniões diárias e revisões de sprints, cruciais para o alinhamento das expectativas do cliente e para o acompanhamento contínuo do progresso do projeto.

### 2.1.1 Os fatores do cliente

A primeira característica do fator cliente destacado pelas autoras como CSF é o forte suporte do cliente ao longo do projeto. Os integrantes da equipe ágil valorizam um apoio consistente por parte do cliente ao longo de todo o projeto. Stankovic et al. (2013) destacaram que o envolvimento do cliente é um fator que contribui significativamente para o sucesso de projetos que utilizam metodologias ágeis. Essas ideias também são apresentadas por um dos Princípios por trás do Manifesto Ágil, que enfatiza: “Pessoas de negócio e desenvolvedores devem trabalhar diariamente em conjunto por todo o projeto” (BECK et al., 2001b).

Outro princípio altamente correlacionado com o projeto estudado e com a participação do cliente, que será utilizado na análise adiante, nos diz que “O método mais eficiente e eficaz de transmitir informações para e entre uma equipe de desenvolvimento é através de conversa face a face” (BECK et al., 2001b).

O segundo apontamento diz que manter um relacionamento positivo com o cliente é essencial para projetos de desenvolvimento de software ágil. Os Princípios Ágeis destacam a importância dessa colaboração, afirmando que "Pessoas de negócios e desenvolvedores devem trabalhar juntas diariamente ao longo de todo o projeto" (BECK et al., 2001b).

Um terceiro fator importante é aprimorar as habilidades do cliente na área de tecnologia. Segundo Misra et al. (2009), a disposição para compartilhar informações e o compromisso com o aprendizado contínuo, tanto dos membros da equipe do projeto quanto dos clientes, contribui para o sucesso das práticas ágeis no desenvolvimento de software. Além disso, um maior conhecimento técnico por parte dos clientes aumenta seu envolvimento no processo de desenvolvimento e sua motivação para apoiar a equipe de desenvolvimento. Por isso que, de acordo com Binboga e Gumussoy (2024), para garantir o sucesso contínuo dos projetos ágeis, as organizações devem investir em capacitação constante de suas equipes.

O quarto e último ponto verificado pelas duas autoras com relação aos fatores do cliente referem-se a importância da experiência do cliente em seu próprio setor.

Essa *expertise* é crucial para que o projeto seja concluído dentro do prazo e do orçamento previstos. Clientes com *expertise* no domínio são capazes de definir os requisitos do produto com precisão e minimizar mudanças inesperadas. Um dos resultados apresentados por elas mostra que clientes com conhecimento sólido em seu ramo de negócios economiza tempo ao expressar claramente os requisitos de negócios. Além disso, clientes experientes oferecem *feedback* preciso ao longo do processo de desenvolvimento, facilitando a conclusão do projeto dentro do orçamento planejado.

### **2.1.2 A satisfação dos stakeholders como métrica de sucesso**

O Manifesto Ágil deixa claro: “Indivíduos e interações mais que processos e ferramentas” (BECK et al., 2001a), ou seja, a colaboração com o cliente é mais importante que negociação de contratos. Um dos principais objetivos do desenvolvimento ágil de software é alcançar a satisfação do cliente por meio da entrega contínua de soluções funcionais. A satisfação prolongada do cliente está diretamente ligada à utilidade e confiabilidade do sistema entregue. Os resultados apresentados também por Binboga e Gumussoy (2024) apontam a satisfação do cliente como uma das métricas mais relevantes para medir o sucesso de um projeto.

Segundo as autoras, a satisfação dos integrantes da equipe também é considerada uma métrica essencial de sucesso no desenvolvimento ágil. Os Princípios Ágeis reforçam essa ideia ao afirmar: "Construa projetos em torno de indivíduos motivados. Proporcione o ambiente e o suporte necessários, e confie neles para realizar o trabalho" (BECK et al., 2001b).

## **2.2 Pesquisa qualitativa**

A pesquisa qualitativa em Engenharia de Software é uma abordagem investigativa que busca compreender fenômenos relacionados ao desenvolvimento, uso ou impacto de software, explorando aspectos humanos, sociais e organizacionais (SEAMAN, 1999). Em contraste com métodos quantitativos, que se concentram em dados numéricos e análises estatísticas, a abordagem qualitativa utiliza dados contextuais para captar experiências, percepções e interações. Técnicas como

entrevistas, estudos de caso, etnografias e análises de documentos são amplamente empregadas para coletar informações (SEAMAN, 1999).

O método qualitativo é valioso em contextos em que os fatores humanos e organizacionais desempenham um papel central, como na adoção de metodologias ágeis e análise de fatores de sucesso em projetos. Por exemplo, Stol et al. (2016) destacam que a pesquisa qualitativa é essencial para explorar como equipes de desenvolvimento lidam com incertezas e como práticas de engenharia evoluem em ambientes dinâmicos. Essa abordagem é recomendada em áreas onde interações humanas são fundamentais para os processos técnicos (RUNESON e HÖST, 2009).

### **2.3 Entrevistas**

Frequentemente a pesquisa qualitativa é baseada ou corroborada por dados retirados de entrevistas a fim de validar ou ampliar a coleta de informações. Em outros casos as entrevistas podem ser o objeto principal de um estudo. Segundo Bogdan e Biklen (1994, p. 134):

Uma entrevista consiste numa conversa intencional, geralmente entre duas pessoas, embora por vezes possa envolver mais pessoas (MORGAN, 1988 apud BOGDAN e BIKLEN, 1994), dirigida por uma das pessoas, com o objetivo de obter informações.

Em uma entrevista não estruturada, também chamada de entrevista aberta, pode não haver um guia de entrevista ou esse guia se resumir a uma pequena lista de tópicos a serem abordados (SEAMAN, 1999).

Nas entrevistas semiestruturadas é garantida a obtenção de dados comparáveis, porém, perde-se a oportunidade de compreender a interpretação dos sujeitos e como eles estruturam o pensamento sobre o tópico em questão (BOGDAN e BIKLEN, 1994). Essas entrevistas incluem uma mistura de perguntas abertas e específicas, e também algumas notas sobre a direção a ser tomada na entrevista em diferentes circunstâncias, permitindo extrair tanto as informações previstas quanto as inesperadas (SEAMAN, 1999).

Já em uma entrevista estruturada, as perguntas são diretas e podem ser organizadas em uma estrutura "se-então" que levará o entrevistador por um dos vários caminhos, dependendo das respostas anteriores.

Tratando-se da ação de entrevistar propriamente dita é importante que, no início da entrevista, informar com brevidade o objetivo da pesquisa qualitativa em curso e garantir-lhe (caso necessário) que o conteúdo dito na entrevista será tratado confidencialmente.

Durante a sessão, o papel do entrevistador não consiste em modificar os pontos de vista do entrevistado, mas em compreender tais pontos dos sujeitos e suas razões. Ou seja, é importante deixar claro que não existem respostas corretas.

## **2.4 Considerações do capítulo**

Neste capítulo foram discutidos os fatores críticos de sucesso ágil com detalhamento para os fatores do cliente e sua influência na satisfação dos stakeholders, ou seja, todas as partes interessadas no projeto. Essa discussão será importante para a análise dos dados coletados nas entrevistas.

Além disso, foi abordado o embasamento por trás da pesquisa qualitativa em engenharia de software e para o método de entrevistas semiestruturadas escolhido, o qual será utilizado na construção dos artefatos de pesquisa, no próximo capítulo.

### **3. ENTREVISTAS**

Este capítulo constitui o corpo do trabalho, no qual se apresentam a estruturação do método de coleta de dados através de entrevistas, a execução das mesmas e seus artefatos resultantes, ou seja, as transcrições das entrevistas.

#### **3.1 Estruturação das entrevistas**

Essa coleta de dados foi feita dentro da corretora de valores de um banco de investimentos no Brasil, cuja investigação abordou a relação da equipe de tecnologia da corretora com os stakeholders, mais especificamente aqueles que solicitaram e utilizam o sistema produzido pela equipe de desenvolvedores, chamados pela sua função de “Traders”. Por confidencialidade, exigida pelas políticas do banco de investimentos no qual este estudo ocorreu, não foi possível utilizar os nomes dos entrevistados, o nome do banco, das equipes estudadas, nem mesmo o nome oficial da aplicação foco deste estudo.

Tratando-se do software analisado, de forma resumida ele é responsável pelos algoritmos de negociação e pela interface de recebimento de negócios, gerenciamento e envio de ordens para a bolsa de valores brasileira, a B3. Trata-se de um software construído há pouco mais de dez anos, resultado de um longo projeto, de mais de um ano de trabalho das equipes envolvidas. A equipe de desenvolvedores da aplicação propriamente dita era pequena, aproximadamente seis pessoas, todas trabalhando na modalidade presencial, na mesma sala da corretora de valores do banco, ao lado de seus clientes (os traders). Utilizou-se uma linguagem orientada a objetos, com servidores físicos para o ambiente de produção.

Foram escolhidos dois entrevistados para a coleta de dados, um representando o time de desenvolvimento e outro representando os clientes. Infelizmente não foi possível entrevistar todos os membros das duas equipes por dois motivos: primeiramente, pois os membros mais novos das equipes não participaram da construção do projeto (ou participaram apenas em parte); em segundo lugar, pois não havia tempo hábil para entrevistar todos os participantes deste projeto de



software devido ao tempo escasso para a execução deste trabalho. Esse fator já evidencia uma limitação do estudo e também uma possibilidade de trabalho futuro.

A escolha do primeiro entrevistado, representante do time de tecnologia, consistiu no desenvolvedor que mais atuou no desenvolvimento do software foco deste estudo. Oportunamente, também é o desenvolvedor com maior tempo de empresa dentro da equipe. O sucesso de sua trajetória fez com que ele tenha assumido a posição de *tech lead* e não esteja mais atuando no desenvolvimento de novos códigos.

O segundo entrevistado, representante dos clientes, foi um dos três traders e sua escolha se deu por exclusão. Dentre os três, um deles é mais jovem, não estava ao longo de todo o processo de construção do sistema; o segundo se tornou o gestor da corretora, então tem outras atribuições para além do uso do sistema, inclusive burocráticas e políticas dentro da empresa. Então, a terceira opção mostra-se a melhor, uma pessoa com grande tempo de empresa (aproximadamente 20 anos), vasta experiência, estava presente desde o início da vida do software foco deste estudo e tem poucas amarras de gestão.

O cabeçalho das entrevistas consiste no nome fictício da pessoa entrevistada, a data em que a entrevista ocorreu e o local da entrevista. A entrevista foi gravada, dada a permissão prévia do entrevistado, coletada antes do início de cada sessão. As perguntas foram formuladas de forma a evitar respostas monossilábicas, como “sim” e “não”. Notas de campo foram escritas após a entrevista. A gravação foi usada durante a escrita das notas de campo.

As quatro primeiras perguntas de ambos, identificadas em azul nas tabelas abaixo, foram pensadas de forma a abordar explicitamente os quatro parâmetros apontados por Binboga e Gumussoy (2024) dentro do construto “fatores do cliente”. O primeiro deles diz respeito ao forte suporte do cliente ao longo do desenvolvimento e da vida útil da ferramenta. O segundo, a existência de um bom relacionamento entre cliente e desenvolvedor. Terceiro: a existência ou investimento nas habilidades profissionais do cliente no domínio de tecnologia. Por fim, a experiência do cliente em seu próprio domínio de atuação, para o qual ele está solicitando o projeto de software.

As duas últimas perguntas das entrevistas, identificadas em verde nas tabelas abaixo, foram pensadas para avaliar a métrica de sucesso de projeto ágil proposto pelas autoras, especificamente no que diz respeito à satisfação dos stakeholders. No projeto, esse papel de stakeholder pertenceu tanto ao time que recebeu o pedido, participou e entregou a solução, quando aos clientes, os quais também eram os usuários finais do produto (traders). A satisfação da alta gerência do banco não foi avaliada por entrevistas devido dificuldade de disponibilidade por parte dos mesmos.

As Tabelas 2 e 3 apresentam na primeira coluna as perguntas que foram feitas ao entrevistado; na segunda coluna evidencia-se o fator estudado, teorizado por Binboga e Gumussoy (2024) e investigado com a pergunta da mesma linha. Por fim, na última coluna, encontram-se os construtos investigados, a cor azul refere-se ao construto “Fatores críticos de sucesso ágil” e em verde referem-se às “Métricas de sucesso do projeto ágil”.

Pergunta	Fator teórico	Construto
1. Como se dava a presença e a participação da mesa de operações durante a construção do software?	Suporte do cliente ao longo do projeto.	Fatores de sucesso ágil referentes ao construto do “Cliente”.
2. De que forma você descreveria a relação interpessoal da equipe de desenvolvimento da corretora com a equipe da mesa de operações?	Boa relação com o cliente.	
3. Qual a sua experiência e a habilidade quanto à utilização e compreensão do funcionamento do software, da sua modelagem e dos erros já apresentados ao longo do tempo, desde o lançamento em produção?	Habilidade e melhoria das capacidades profissionais do cliente no domínio de TI.	
4. Como você categoriza o seu grau de experiência em seu domínio de atuação?	Experiência do cliente em seu próprio domínio de atuação.	
5. Na posição de cliente da equipe de IT, descreva sua experiência com o uso do software.	Satisfação dos usuários finais (clientes)	Métricas de sucesso do projeto referentes à satisfação dos <i>stakeholders</i>
6. Descreva sua satisfação com o resultado do projeto.		

Tabela 2: Guia de entrevista a ser utilizado com o cliente.

Pergunta	Fator teórico	Construto
1. Como se dava a presença e a participação dos clientes ( <i>traders</i> ) durante a construção do software?	Suporte do cliente ao longo do projeto.	Fatores de sucesso ágil referentes ao construto do “Cliente”.
2. De que forma você descreveria a relação interpessoal da equipe de desenvolvimento da corretora com a equipe da mesa de operações?	Boa relação com o cliente.	
3. Qual a experiência e a habilidade dos clientes quanto à utilização e compreensão do funcionamento do software, da sua modelagem e dos erros já apresentados ao longo do tempo, desde o lançamento em produção?	Habilidade e melhoria das capacidades profissionais do cliente no domínio de TI.	
4. Como você categoriza o grau de experiência dos clientes em seu domínio de atuação?	Experiência do cliente em seu próprio domínio de atuação.	
5. Qual a sua percepção tanto do processo de desenvolvimento quanto do resultado obtido?	Satisfação dos membros do time de desenvolvimento	Métricas de sucesso do projeto referentes à satisfação dos <i>stakeholders</i>
6. Descreva sua satisfação com o projeto, como membro da equipe de desenvolvimento.		

Tabela 3: Guia de entrevista a ser utilizado com o desenvolvedor.

### 3.2 Aplicação das entrevistas

A primeira entrevista aplicada foi feita com o cliente, presencialmente, em um ambiente tranquilo, no dia 11 de novembro de 2024. A gravação teve duração de 38 minutos e 13 segundos.

A segunda entrevista foi feita com o desenvolvedor, feita presencialmente, em um ambiente tranquilo, no dia 25 de novembro de 2024. A gravação teve duração de 43 minutos e 34 segundos.

A primeira etapa das entrevistas não foi gravada e consistiu apenas numa breve explicação deste trabalho, da teoria e da justificativa para sua execução. Entregou-

se uma impressão com as tabelas contendo os respectivos guias de entrevista e aguardou-se alguns minutos para o entrevistado ler e tirar suas dúvidas a respeito da própria pergunta. Em seguida iniciou-se a gravação até o fim da sessão. As transcrições encontram-se no Apêndice A e B.

O processo de transcrição foi dividido em duas etapas, na primeira delas utilizou-se a ferramenta “Galaxy AI”, nativa de celulares Samsung Galaxy, que possuem as funções de inteligência avançada ou artificial (IA). Dentro do ferramental disponível foi utilizado especificamente o “Assistente de transcrição” que transforma as gravações em transcrições de texto. Dentro do assistente foi necessário baixar e instalar o pacote de idioma para transcrição “português (Brasil)”. Das configurações disponíveis pela ferramenta, foram colocadas antes do início da gravação: “autodetectar idioma”, modo de gravação “Normal”, qualidade de gravação alta (256kbps, 48kHz) e canal “mono”. Importante ressaltar que o modo de gravação normal é obrigatório caso queira utilizar a transcrição por IA, uma vez que os demais modos não permitem o uso da ferramenta, segundo as instruções do aplicativo.

A segunda etapa foi a de revisão, que consistiu em ouvir o áudio gravado ao mesmo tempo em que se realizava a conferência do resultado gerado pela IA, palavra por palavra. Isso foi especialmente importante uma vez que numa conversa cotidiana, ainda que em contexto de entrevista, os entrevistados traziam uma série de gírias, jargões e vícios de linguagem que atrapalham o resultado obtido pela ferramenta automática.

Os principais vícios no contexto de um banco internacional foram os estrangeirismos, ou seja, uso de palavra de língua estrangeira, quando existe uma forma equivalente em português. Além disso, a IA também não identifica expressões não verbais como “hummm”, “ahh” e etc. Tudo isso foi transcrito na íntegra, para manter as respostas do entrevistado originais.

Existem também palavras que ficam subentendidas pelo contexto ou por gestos com as mãos ou com a comunicação visual, essas informações foram inseridas pelo entrevistador/transcritor e identificadas por colchetes [ ]. Por fim, nomes de bancos,

ou outras instituições, nomes de softwares e nomes de pessoas foram omitidos pelo entrevistador/transcritor e também identificadas por colchetes [ ].

### **3.3 Considerações do capítulo**

Neste capítulo apresentou-se a construção dos materiais para a coleta de dados e os procedimentos adotados para a entrevista. Detalhou-se a parte teórica por trás do planejamento das entrevistas semiestruturadas, destacando como os CSF foram abordados nas entrevistas, principalmente no que diz respeito à relação entre os fatores do cliente e à satisfação dos stakeholders. Foi descrito com detalhes também o processo das entrevistas e as tecnologias utilizadas para gravação e transcrição.

## **4. ANÁLISE DOS RESULTADOS**

Este capítulo apresenta uma análise crítica dos dados coletados com as entrevistas. Além da análise dos dados retirados de cada entrevista foi feita uma comparação desses resultados com o Modelo de Sucesso de Projetos de Software de Binboga e Gumussoy (2024).

### **4.1 Resultados recolhidos da entrevista com o cliente**

A seguir são apresentadas as análises de cada um dos fatores do cliente destacados pelas autoras como CSF. Cada tópico abaixo relacionará cada resposta do cliente com o fator teórico investigado pela respectiva pergunta, como já correlacionado anteriormente na Tabela 2.

#### **4.1.1 Análise do suporte do cliente ao longo do projeto**

A partir das respostas da entrevista com o cliente, observa-se uma forte presença deles ao longo de diversos momentos de existência do software em estudo. Para a análise, é necessário falar da origem e da idealização deste software: sua idéia inicial surgiu com o intuito de substituir e resolver problemas na vida dos traders (clientes) quando utilizavam as antigas planilhas de Excel e Macros, também chamadas por eles de “maquininhas”.

Esses produtos legados dificultavam e atrasavam a execução das tarefas necessárias dos clientes e já estavam obsoletos. Mas esses produtos também eram “propriedade intelectual” da equipe de desenvolvimento, ou seja, esses desenvolvedores já conheciam o ponto de partida para o software deste estudo e já tinham parte da base conceitual de partida para a criação deste novo software, como mencionado pelo cliente: “ele [o sistema] veio de uma planilha de Excel, então a maioria das estratégias vieram de uma planilha de Excel”.

É possível verificar que existiu uma base para o projeto estudado, e a complexidade do software surgiu a partir de uma evolução do mercado financeiro no Brasil ao longo dos anos. Essa evolução na negociação da bolsa de valores exigiu uma

evolução tecnológica por parte do banco, que foi gradual e dinâmica. Tanto na época de construção, quanto posteriormente (e atualmente) as equipes de negócios e de tecnologia trabalhavam presencialmente, alocadas na mesma sala, ou seja, tudo era verbal, discutido, não havia formalizações por email, não havia grandes documentações de pedidos por escrito, como relatado pelo cliente: “Nunca foi uma coisa assim, por escrito, um documento formal”. O que havia era uma corrida tecnológica dos bancos para sair na frente na execução eletrônica da bolsa.

Havia também o pré-requisito formal das especificações da bolsa de valores para negociação via mercado eletrônico. Esse material, utilizado até hoje, trata-se de um documento com especificações sobre a troca de informações via mensageria eletrônica, divulgado publicamente no site da B3 (bolsa de valores brasileira), entidade que regulamenta o recebimento e a execução de ordens.

Com relação aos pedidos de implementação de novas estratégias de negociação dentro do software estudado, o que observa-se é um processo iterativo-incremental, cuja forte presença dos clientes foi fundamental para a entrega de soluções sempre alinhadas às necessidades deles.

Tinha a entrega do pouquinho e o complemento, mas tinha muito também a entrega do que eu tô pedindo e aí o aperfeiçoamento, que não foi solicitado na primeira vez. Era muito de aperfeiçoar. [...] Rodar a estratégia e ver se precisa de mais alguma coisa. E aí rodava a estratégia por 3, 4, 5 meses. Aí bateu 5 meses [...] começou a ter a seguinte necessidade [...].

Então vamos implementar mais uma funcionalidade nessa estratégia que é a seguinte: se bater um determinado preço que eu colocar na estratégia, ele executa tudo. Se não, ele continua com a ideia original. Então eram várias melhorias. A gente chamava de melhoria [...]. Mas o ponto principal da estratégia, a primeira solicitação, ela sempre é válida.

#### **4.1.2 Análise da relação cliente x desenvolvedor**

A relação interpessoal já havia sido abordada de forma indireta na primeira resposta do cliente, pois ele frequentemente comentava da importância da presença dos

desenvolvedores no dia a dia e no mesmo ambiente, das conversas constantes, dos cafés: “sempre assim na conversa, às vezes no café ou coisa assim”.

Um ponto a ser ressaltado nas respostas do cliente é a valorização dele em compartilhar o espaço e o dia a dia com o time de desenvolvimento, assim os dois times conhecem suas rotinas, entendem mais das tarefas de cada um e das dificuldades de cada um: “(...) a conversa quando a mesa está dentro, quando o pessoal de tecnologia está dentro da corretora e tá vivenciando o dia a dia é muito mais fácil do que quando tá fora da corretora”.

O entrevistado inclusive faz um paralelo de um cenário que ele viveu anos antes, quando o banco dele ainda não havia passado pelo movimento de integração com a atual instituição financeira e como isso tornava os resultados piores na percepção dele:

Eram estruturas apartadas, então eu tinha uma estrutura de tecnologia para a mesa de operações, que não ficava na mesa de operações. E aí, quando eu vim pro [segundo banco] a gente tinha uma estrutura de tecnologia de mesa de operações que ficava dentro da mesa de operações. [...] e isso é muito diferente.

Porque o cara que tá dentro da estrutura da corretora vivendo ali o dia a dia da mesa de operações, ele entende a nossa necessidade. Muitas vezes ele entende o problema e ele sabe muito mais, com muito mais clareza e mais facilidade o que ele tem que fazer para me ajudar com aquilo.

Ele trouxe em sua fala a dificuldade de se comunicar com a equipe de desenvolvimento quando ela estava fora do contexto de negócios: “O oposto era muito ruim. Porque primeiro, o fornecedor no caso, ele não tinha uma ideia de como era mercado. Eu tinha que ensinar o mercado pra ele, pra depois ele pensar como ele desenvolveria aquilo”.

#### **4.1.3 Análise do domínio do cliente no campo da tecnologia**

Essa resposta chamou a atenção pela coincidência, o cliente havia tido uma base de formação em ciência da computação em seu colegial. Segundo ele, isso facilita sua comunicação com o time de desenvolvimento, principalmente no momento de passar os pedidos de novas funcionalidades, uma vez que ele consegue concatenar seus



pensamentos de forma lógica e transmitir as necessidades de execução de código com mais precisão, como constatado na fala: “eu fiz um ano de ciência da computação e parei. Então eu acho que tem algumas coisas que eu consigo ver e pensar [...]. Mas a lógica está um pouco dentro [de mim], então consigo dar idéias”.

Independente dessa informação ficou claro a importância de treinamento dos clientes, pois ele ressalta que o seu conhecimento atual da parte tecnológica e a familiaridade com os conceitos de tecnologia vieram com o aprendizado adquirido e os ensinamentos passados pela equipe de desenvolvimento, pois foi questionado se ele sempre foi capaz de identificar com clareza onde estavam os problemas e os pontos de melhoria na negociação eletrônica e a resposta foi: “Não, não. A gente foi aprendendo com vocês”.

Também existe um conhecimento interdisciplinar de tecnologia e também de processo do negócio que é entender o processo de execução no mercado eletrônico, como ele destacou em:

Aí eu acho que é onde tá a ponta mais importante, a gente sempre tentou entender o que o sistema faz, a gente não se limitou a pedir para vocês, por exemplo, ‘eu quero que o sistema faça isso’. Aí se ele não tá fazendo, pede pra corrigir, porque a gente via isso no passado.

Esse conhecimento apresentado pelo cliente torna a avaliação do resultado do software mais precisa e ajuda na devolutiva aos desenvolvedores, como ele ressalta: “às vezes o tipo de erro que volta a gente sabe em qual parte do caminho uma mensagem parou”.

#### **4.1.4 Análise do domínio do cliente em seu campo de atuação**

O trecho destacado na resposta anterior fornece indícios de que o conhecimento do cliente em seu domínio de atuação tenha uma importância maior para o sucesso do projeto do que o seu conhecimento tecnológico. Isso aparece nos diversos momentos em que o cliente ressalta a importância de entender os processos por trás das ações que ele deseja executar. Como no trecho: “alguns erros a gente já está familiarizado, então a gente acaba tendo esse pré-diagnóstico [...], mas não é por

conhecimento do software. Na verdade, é justamente por saber toda a linha de processo que o comando vai ter”.

Ou seja, esses conhecimentos especializados que ele possui em entender onde seu trabalho se coloca dentro da cadeia de negociação de investimentos na bolsa de valores e o caminho que uma ordem de investimento precisa percorrer para sair desde uma solicitação de compra ou venda, o envio dessa ordem, até chegar na bolsa de valores brasileira.

Sem esse conhecimento de mercado e o conhecimento das obrigações regulatórias, não seria possível entregar um software que atenda as demandas da mesa de operações, como o cliente indica em: “é que assim a gente vem do mercado lá de traz, então a gente cresceu junto com o mercado. O mercado foi evoluindo e a gente foi evoluindo também [...] a gente vai entendendo todo o processo pra ordem chegar onde ela tem que chegar”.

Também vale ressaltar que o cliente entrevistado tem mais de 22 anos de experiência no mercado financeiro, e lembrou em suas respostas todas as áreas pelas quais já passou ao longo de todos esses anos. Isso fornece mais indícios positivos sobre seu amplo conhecimento em seu domínio de atuação. Ele também destacou sobre a qualidade da equipe da mesa de operações de uma forma geral em alguns trechos:

Teve outro ponto muito relevante que a gente sempre teve pessoas dentro da equipe que ‘manja’ do mercado [...].

A gente também, do nosso lado, sempre tentou entender qual é o caminho que o sistema faz, qual é a lógica que ele usa. [...] porque não adianta nada eu pedir pra você faz uma estratégia X se eu não souber como ela tá se comportando.

Por fim ressaltou novamente a importância disso no processo de avaliação e de evolução do software entregue: “Aí eu vou lá ligo aquela estratégia sai fazendo barbaridades no mercado só que eu não sei qual é a barbaridade, porque eu não me preocupei em saber qual era o “pensamento” dela, qual era o raciocínio dela”.

#### 4.1.5 Análise da satisfação do cliente enquanto stakeholder

O cliente avaliou sua satisfação e experiência com o resultado do projeto como positiva, destacou que o software resultando atende às demandas que ele tem necessidade, dando ênfase em suas respostas sobre a interface *user-friendly* e intuitiva, ressaltando que isso facilita e agiliza o processo de entrada de novos membros na mesa de operações, ainda que jovens e inexperientes, como ressaltado em:

A experiência com o uso do software é muito friendly. É muito fácil de usar. É tipo um como se tivesse um iPhone entendeu? [Risos]. É bem diferente do Android, é mais fácil de usar. [...] E quando a gente vai ensinar uma pessoa a usar, ele é muito muito rápido de aprender.

Ressaltou a intuitividade do software e como isso agiliza o processo de desenvolvimento quando novos colaboradores entram na mesa de operações: “ele é muito, como eu posso dizer, ele muito intuitivo. Então é super tranquilo de usar. [...] é muito mais difícil ensinar pro cara novo o que é um spread do que usar o sistema em si. [...] E atende as expectativas que a gente espera”.

O cliente também ressaltou sobre a precisão do projeto e à satisfação com o resultado frente às necessidades da mesa. E quando se questionou sobre os erros da aplicação ele avaliou positivamente, falando que as demandas futuras não eram provenientes de erros da aplicação, mas sim de mais automatizações visando enfrentar a diminuição de mão de obra e os cortes no corpo de funcionários “Isso funciona super bem. [...] a maioria das demandas que a gente tem são demandas de aprimoramento para facilitar nossa vida e nosso dia a dia, porque a gente tá cada vez com menos pessoas na área”.

O único ponto de crítica trazido foi o fato de o software ao longo dos anos ter ganhado muitas funcionalidades que faziam sentido na época e hoje não fazem mais e, por ser uma aplicação *desktop*, ele sente que esse excesso de ferramentas e essa robustez roubam velocidade em microssegundos, quando comparado com um ou outro concorrente, isso é percebido na sua fala em: “eu sei que tem um erro, que ele, às vezes, por ser um sistema pesado, de corretora e robusto, ele perde

velocidade para outros tipos de software”.

Por outro lado, ressaltou em sua fala a segurança ao utilizar o sistema, e a proteção que eles possuem devido aos pedidos de travas de parada automáticas em caso de movimentações bruscas do mercado financeiro ou outras situações, fazendo com que se sintam confortáveis em operar grandes quantidades de dinheiro:

A partir disso que a gente pediu maneiras de se proteger, foi que a gente pediu pra estratégia parar sozinha se de repente, um determinado fato acontecer. Então, isso permitiu pra gente colocar várias travas que deixa a gente mais confortável hoje. [...] Eu me sentia um pouco mais vulnerável. Hoje a gente viu três dias, a última semana inteira na verdade, a gente viu uma volatilidade absurda no dólar assim, muito, muito volátil por conta, por exemplo, de eleição e pacote do governo que não sai, e mesmo assim não fomos uma presa fácil. [...] Então a última semana deixou a gente muito contente e confortável em relação a esse delay.

#### **4.1.6 Comparação dos resultados do cliente com o Modelo de Sucesso de Projetos de Software Ágil**

Os resultados apresentados acima estão em concordância com o modelo de sucesso de projetos de software ágil proposto por Binboga e Gumussoy (2024) no que se refere aos fatores do construto cliente e sua alta correlação com a satisfação dos stakeholders.

Foi possível perceber pela análise da entrevista que no projeto estudado tivemos uma forte participação do cliente ao longo do projeto e uma relação bastante próxima entre o time de desenvolvimento e o time da mesa de operações (denominado cliente).

O ponto que se mostrou fundamental para um bom resultado do software foi o domínio do cliente em seu campo de atuação. Segundo o cliente entrevistado, ele e seus pares possuem um amplo conhecimento do mercado de negociação eletrônica na bolsa de valores brasileira e isso facilitou desde a precisão das solicitações aos desenvolvedores até ao teste e *feedback* do resultado para correções e melhorias, fazendo com que o projeto se torna-se mais ágil e tivesse um resultado mais efetivo.

Afinal, como ele mesmo menciona, é um software que atendeu e atende ainda a todas as necessidades da corretora, de forma simples, intuitiva e segura e que facilita a integração de novos membros.

## **4.2 Resultados recolhidos da entrevista com o desenvolvedor**

A seguir são apresentadas as análises de cada um dos fatores do cliente destacados por Binboga e Gumussoy (2024) como CSF, analisados sob a ótica do desenvolvedor. Cada tópico abaixo relacionará cada resposta desse desenvolvedor com o fator teórico investigado pela respectiva pergunta, como já correlacionado anteriormente na Tabela 3.

### **4.2.1 Análise do suporte do cliente ao longo do projeto**

Corroborando as informações colhidas com o cliente, constatou-se que o software deste estudo foi uma evolução de uma aplicação legada, também de propriedade intelectual do mesmo time de desenvolvimento.

Isso é importante para ressaltar que as ditas “estratégias de negociação” iniciais não foram criadas do zero, foram transportadas e renovadas para uma nova interface e infraestrutura, para suportar o aumento no número de negócios e o avanço na negociação eletrônica da bolsa de valores brasileira, como o desenvolvedor menciona: “as aplicações que a gente tinha, era legado, não suportava, não era escalável, era *single thread*, então não tinha *performance*, ficava limitada à capacidade da máquina do usuário”.

O que ocorreu foi uma proposta do próprio time de desenvolvimento de renovar a plataforma dos algoritmos para sustentar o aumento da demanda. Essa proposta foi uma resposta às crescentes reclamações dos usuários referente à performance: “Ele tinha reclamações, tipo, ele mostrava aqui, olha tá ruim aqui, [...] eles chamavam ‘oh não tá executando’, aí de repente ‘ah executou’. Porque ela [estratégia] ia uma a uma executando”. O desenvolvedor enfatizou que os clientes aceitaram investir no projeto imediatamente.

Então, segundo o desenvolvedor, no início do projeto a presença do time de infraestrutura e de redes do banco era mais importante do que a presença do cliente, como ele menciona:

Esse início não dependia muito do cliente [...]. Não era só a lógica de programação, do algoritmo ou do sistema em si. Estava muito relacionado com o ambiente [...] a gente começou acertar essas coisas junto com o time de Infra, por isso que a gente também teve essa nossa relação com o time de infra, de servidores de redes... a gente precisava de muita ajuda deles.

Depois de estabelecida a versão inicial da nova plataforma que iniciaram as trocas e demandas de novas estratégias dos clientes: “Então eles falaram “caramba, agora eu tenho uma aplicação boa pra caramba aqui, tô executando as ordens dos clientes aqui, tá bombando, preciso de *Market maker* aqui também” [...]. Então eles começaram a demandar”.

Questionado sobre quem estabelecia a prioridade, a complexidade e outros fatores, ele mencionou: “Nós junto com o *Business*. Porque a gente estimava, a gente falava de complexidade”. Ressaltou que eles e clientes trabalhavam como uma unidade com capacidade de decisão.

#### **4.2.2 Análise da relação desenvolvedor x cliente**

Também em consonância com as respostas do cliente, o desenvolvedor descreveu a relação bastante próxima com os clientes (traders), também trouxe a valorização de compartilhar o mesmo espaço que o cliente, ressaltando a importância dessa proximidade para o sucesso do projeto:

Essa interação que a gente tinha com eles era muito próxima mesmo. Por que a gente sempre teve sentado dentro da corretora e cada passo que a gente dava, de progresso que a gente fazia, a gente conversava com eles e mostrava [...] a gente sempre teve essa demanda de ficar no mesmo andar e mesmo ambiente que eles.

Ele também mencionou a importância dessa proximidade nos testes executados pelos usuários e na orientação dos usuários e na troca de informações sobre tecnologia: “a gente colocava uma versão nova e explicava ‘olha, agora testa, vamos fazer assim’. Explicava, eles entendiam, faziam o teste de acordo com o que a gente orientava. E assim, era uma interação constante com eles, desde a primeira fase”.

#### **4.2.3 Análise do domínio do cliente no campo da tecnologia**

O desenvolvedor trouxe a necessidade de entender a demanda por trás das reclamações iniciais acerca de *performance*, inclusive ressaltando que não acredita que o usuário necessite entender desse domínio, como ele analisa neste trecho:

Então a percepção do usuário é o que ele tá vendo na tela, ele não enxerga por trás como a aplicação funciona. Então o nosso trabalho era abstrair essa visão do usuário. De fato, ele não precisa saber o que tá acontecendo por trás, mas ele quer o resultado, ele quer aplicação respondendo.

Observou-se a partir das falas que a experiência dos usuários no domínio de tecnologia era muito baixa no início do projeto, o que trouxe algumas dificuldades do ponto de vista do desenvolvedor principalmente nas primeiras versões do projeto e o projeto piloto, como ele descreve: “Então eles não tinham experiências com tecnologia, eles sabiam ser usuários, então a gente também sofria com isso porque o usuário não lê nem mensagem de erro. A gente colocava umas mensagens na tela eles não liam”.

Mas ele destaca que essa situação passou a ser melhor com as trocas de informação e as explicações por parte do time de tecnologia, corroborando com a ideia de Binboga e Gumussoy (2024) sobre a importância de investir na melhoria das capacidades técnicas do cliente no domínio de TI. O entrevistado trouxe essa ideia em sua fala com algumas ressalvas sobre depender também do interesse dos clientes em aprender:

Quando a gente começou a explicar pra eles um pouco mais como funcionava e tal aí alguma coisa melhorou, mas não era pra todo mundo [...]. Alguma coisa que eles

estão pedindo eles vão querer estar mais por dentro, agora se é só executar uma ordem, quer que funcione e pronto.

#### **4.2.4 Análise do domínio do cliente em seu campo de atuação**

Também alinhado com as respostas do cliente, o desenvolvedor confirmou a grande experiência dos traders em seu domínio de atuação, ele afirmou: “no início, quando a gente começou a desenvolver, nos primeiros anos do sistema. Ali tinham só pessoas mais experientes mesmo”.

O entrevistado avaliou que a junção de pessoas de tecnologia capacitadas, em conjunto com um cliente com grande habilidade no seu mercado favoreceu para o sucesso do projeto:

Talvez isso tenha contribuído muito pro sucesso do início [...] a gente tinha pessoas muito capacitadas para fazer, tecnicamente, mas a gente não conhecia do mercado. Só que esses caras conheciam muito do mercado, entendeu? Então ali a parte técnica que era competente junto com os caras que eram competentes no mercado.

Ressaltou que apesar da entrada constante de novas pessoas, dado uma época de mercado em expansão, as demandas eram solicitadas pelos mais velhos e com mais conhecimento, de forma explícita ele traz: “As demandas sempre vinham mais dos mais experientes”.

Explicou também que os mais novos estavam aprendendo tanto a utilização da ferramenta quanto os conceitos de sua própria área, portanto eles eram gerenciados pelos mais velhos e apenas utilizavam as ferramentas disponíveis e colaboravam apenas apontando pequenos ajustes necessários, isso é observado no seguinte comentário: “ele também não ia te pedir um negócio novo, ele ia usar o que tem entendeu [...]. Mas às vezes eles percebiam coisas que precisavam ser ajustadas”.

#### **4.2.5 Análise da satisfação do desenvolvedor enquanto stakeholder**



O desenvolvedor se mostrou muito orgulhoso e apresentou o que o banco chama de *take ownership* (assumir a propriedade). Trouxe a informação de que foi um dos principais desenvolvedores por trás do projeto e sua satisfação pode ser claramente verificada neste trecho: “a minha sensação, a minha realização tipo pessoal, vendo o negócio funcionar, atendendo tudo que atendeu, funcionando até hoje, bem e o pessoal falando bem, é uma realização pessoal muito boa”.

Relembrou diversos elogios que já recebeu ao longo do tempo pela entrega da aplicação, como por exemplo, nos trechos: “nas primeiras versões, eles de fato ficaram surpreendidos, eles falavam ‘nossa, caramba tá muito rápido’”. Ou em: “não demorou muito tempo, chegou ao ponto de que ‘pô, tá legal, né?’, ‘tá funcionando bem’. E aí, depois começaram a vir os primeiros elogios. Falavam ‘nossa, tá muito rápido’”.

Por fim, o desenvolvedor ressaltou o importante ponto sobre a precisão do projeto e à superação do resultado frente às expectativas da mesa, desde melhoria de *performance* até a posterior criação de novas estratégias proprietárias, sempre atendendo as normas da bolsa de valores, como ele menciona:

Sim, a gente entregou foi pedido. Acho que a gente foi até além um pouco [...] chegamos onde a gente queria. A gente tem um sistema extremamente escalável, que faz o que muita gente paga muito dinheiro no mercado para usar e pra ter. E faz também o que ninguém tem, que são os nossos algoritmos.

#### **4.2.6 Comparação dos resultados do desenvolvedor com o Modelo de Sucesso de Projetos de Software Ágil**

Os resultados obtidos dessa entrevista se mantiveram alinhados com aqueles apresentados por Binboga e Gumussoy (2024). As falas do desenvolvedor corroboraram com as já trazidas pelo cliente acerca de uma intensa colaboração entre ambos e da importância do bom relacionamento entre as duas equipes, como fatores relevantes para o sucesso de um projeto de desenvolvimento de software, como o próprio entrevistado conclui: “No final das contas, sem essa integração de desenvolvedor com o cliente a gente não tinha tido tanto sucesso não”.

#### 4.4 Ameaças a validade

Como mencionado no Capítulo 3, não foi possível entrevistar todos os membros das duas equipes, apenas um cliente e um desenvolvedor foram entrevistados, pois os membros mais novos das equipes não participaram da construção do projeto (ou participaram apenas em parte); em segundo lugar, pelo menor número de pessoas participantes que se mantiveram como funcionários do banco de investimentos após a venda do mesmo para um concorrente; por fim, devido ao tempo escasso para a execução deste trabalho.

Além dessa limitação existe o fato de os questionários não terem sido aplicados com um grupo piloto antes das entrevistas para coleta de dados oficial, devido ao pouco tempo disponível para a execução das entrevistas. Porém, para capturar e corrigir possíveis desentendimentos no significado das perguntas. Para que esse ponto não se impactasse a validade deste estudo optou-se por realizar as entrevistas presencialmente, no formato semi-estruturado e com intermediação do entrevistador. Assim, o entrevistador pode perguntar se a o questionamento está claro, explicar com outras palavras caso necessário e guiar eventuais fugas do assunto por parte do entrevistado, garantindo que o tema das perguntas seja realmente abordado.

#### 4.5 Considerações do Capítulo

Neste capítulo analisou-se as características do “fator cliente” destacadas pelas autoras como CSF e comparou-se essas características com as respostas dos entrevistados sobre cada uma delas. Também foi avaliada a visão dos entrevistados sobre o sucesso do projeto estudado com base na métrica de sucesso de projetos ágeis especificamente para o fator “satisfação dos *stakeholders*”, em acordo com o objetivo proposto no início deste estudo, de avaliar a existência de uma relação positiva entre o fator cliente e a satisfação dos *stakeholders*.

## 5. CONSIDERAÇÕES FINAIS

Partindo dos resultados apresentados Binboga e Gumussoy (2024) sobre a alta influência do “fator cliente” na satisfação dos stakeholders, apresentados na Tabela 1, avaliou-se qualitativamente a relação positiva entre os construtos desse fator com a satisfação dos stakeholders, são eles: forte suporte do cliente durante o projeto, o bom relacionamento com o cliente, a melhoria nas capacidades do cliente no domínio de tecnologia e a experiência do cliente em seu domínio de negócio.

Para a avaliação foi utilizada uma abordagem qualitativa com utilização de entrevistas semiestruturadas, presenciais, gravadas e transcritas integralmente, a fim de abordar os quatro construtos do fator cliente citados acima.

Os resultados obtidos a partir das entrevistas corroboram com os já apresentados por Binboga e Gumussoy (2024), com os Princípios por trás do Manifesto Ágil e com o próprio Manifesto que enfatiza “Indivíduos e interações mais que processos e ferramentas” (BECK et al., 2001a).

A abordagem qualitativa deste estudo permitiu ampliar as conclusões das autoras base deste trabalho e analisar dados contextuais para captar as interações cotidianas. O método escolhido foi importante para atingir o objetivo deste estudo, e fortificar o Modelo Ágil de Sucesso de Projetos de Software, principalmente no que diz respeito aos fatores do cliente, que representam a maior variância na satisfação dos stakeholders, de forma positiva.

Encontraram-se evidências no projeto estudado de uma forte colaboração entre cliente e desenvolvedor, com uma relação estreita entre as duas equipes, com amplo contato e troca de informações. Ademais, os clientes se mostraram altamente capacitados em seu domínio de atuação e mostraram uma evolução em sua capacidade técnica no domínio de tecnologia ao longo do tempo.

Segundo as falas dos entrevistados isso apresenta um fator de influência positiva com o sucesso do projeto, relatado por eles como um software seguro, *user-friendly*, intuitivo e que atendeu amplamente as demandas dos clientes. Do lado do

desenvolvedor também se notou uma satisfação com o projeto e seu resultado com a escalabilidade e precisão, com soluções proprietárias adequadas às necessidades dos usuários. Todas essas ideias foram bem concluídas também pelo próprio entrevistado que ao final da entrevista relatou a seguinte frase:

Acho que quanto mais próximo e mais alinhado, tipo mais conhecimento é trocado ali, as figurinhas, sabe? Um entende melhor o outro e a solução final sai melhor, entende? Se for uma coisa meio que mal conversada vai sair, depois no final não atende a expectativa exatamente como o cliente queria, como era esperado.

## **5.2 Contribuições do Trabalho**

As contribuições deste trabalho estão na abordagem investigativa utilizada que foi qualitativa, capaz de compreender fenômenos relacionados ao desenvolvimento, explorando os seus aspectos humanos, sociais e organizacionais. Essa abordagem qualitativa permitiu coletar e analisar dados contextuais para captar percepções e interações. O método qualitativo escolhido foi importante para o objetivo deste estudo, onde os fatores humanos e sociais desempenharam um papel central no sucesso do projeto.

Como destaque, os dados mostraram a importância de uma relação próxima e a necessidade de colaboração não somente entre cliente e desenvolvedor, mas também entre os desenvolvedores e outros times de tecnologia que suportam os ambientes de desenvolvimento, teste e implantação, como os times de redes e de infraestrutura.

## **5.3 Trabalhos Futuros**

Existem dois possíveis trabalhos futuros. O primeiro é aprofundar o estudo feito neste trabalho por meio de um estudo de caso completo, colhendo dados não só através de entrevistas, mas também de documentações disponíveis, como Business Requirements Documents (BRD), atas de reuniões de solicitação (se houverem) ou observação participante.

O segundo trabalho futuro seria utilizar a pesquisa qualitativa para estudar os outros CSF apontados por Binboga e Gumussoy (2024) e verificar e aprofundar as descobertas das autoras e seu impacto no sucesso de projetos de desenvolvimento de software ágil.

## REFERÊNCIAS

- BECK, K. et al. Manifesto para Desenvolvimento Ágil de Software, 2001. Accessed: Dez. 09, 2024a. [Online]. Available: <https://agilemanifesto.org/iso/ptbr/manifesto.html>
- BECK, K. et al. **Principles Behind the Agile Manifesto**, 2001. Accessed: Dez. 09, 2024b. [Online]. Available: <https://agilemanifesto.org/iso/ptbr/principles.html>
- BINBOGA, B.; GUMUSSOY, C. A. **Factors affecting agile software project success**. IEEE Access, 2024.
- BOGDAN, Robert; BIKLEN, Sari. **Investigação qualitativa em educação: uma introdução à teoria e aos métodos**. Porto editora, 1994.
- JOHNSON, J. **CHAOS 2020: Beyond Infinity Overview**. The Standish Group International: Boston, MA, USA, 2021.
- MENEZES, J.; GUSMÃO, C.; MOURA, H. **Risk factors in software development projects: a systematic literature review**. Software Quality Journal, v. 27, p. 1149-1174, 2019.
- MILOSEVIC, D.; PATANAKUL, P. **Standardized project management may increase development projects success**. International journal of project management, v. 23, n. 3, p. 181-192, 2005.
- MISRA, S. C.; KUMAR, V.; KUMAR, U. **Identifying some important success factors in adopting agile software development practices**. Journal of systems and software, v. 82, n. 11, p. 1869-1890, 2009.
- RUNESON, P.; HÖST, M. **Guidelines for conducting and reporting case study research in software engineering**. Empirical software engineering, v. 14, p. 131-164, 2009.
- SEAMAN, C. **Qualitative methods in empirical studies of software engineering**. IEEE Transactions on software engineering, v. 25, n. 4, p. 557-572, 1999.
- STANKOVIC, Dragan et al. **A survey study of critical success factors in agile software projects in former Yugoslavia IT companies**. Journal of systems and software, v. 86, n. 6, p. 1663-1678, 2013.
- STOL, K.; RALPH, P.; FITZGERALD, B. **Grounded theory in software engineering research: a critical review and guidelines**. In: Proceedings of the 38th International conference on software engineering. 2016. p. 120-131.

## APÊNDICE A – Transcrição da entrevista com o cliente

Entrevista com o cliente, um trader da mesa de operações. Este apêndice foi desenvolvido pela autora.

Entrevistadora (00:07)

Estou entrevistando o cliente da mesa de operações a respeito da construção de um software da corretora de um banco de investimentos. Os dados confidenciais foram omitidos por questões de confidencialidade a pedido do banco.

Entrevistadora (00:36)

A primeira pergunta é: como se dava a presença e a participação da mesa de operações, vulgo clientes, durante a construção do software.

Cliente (00:50)

Tá quem tinha o *know-how* de tudo que tinha que ter e que tem que ter no software é a gente, da mesa de operação. Então assim, a gente que sabe como é a estratégia é como ela tem que se comportar, quais as estratégias a gente precisa para poder atender às demandas dos clientes. Então a gente passa tudo isso para vocês que são os fornecedores do serviço do software e aí vocês veem a maneira a melhor maneira de desenvolver tudo isso, então, por exemplo, a gente tem uma estratégia que a gente precisa só colocar uma ordem no mercado de compra e venda, ou seja, quantidade, compra ou venda e cliente,

Cliente (01:35)

Então vocês vão lá fazem uma telinha e fazem todo o acesso ao mercado para a gente colocar [os parâmetros] cliente, compra ou venda, o preço e o ativo que a gente quer negociar e aí tem umas outras estratégias que têm mais inteligência que, por exemplo, é, eu quero usar uma estratégia *over the day*, então eu passo pra você que eu quero fazer na menor quantidade possível de um determinado lote, então vamos supor que eu vou comprar uma ação X é... 10000 ações e que eu tenho um tempo de mercado de 6 horas. Então você vai dividir aquilo por tempo e vai usar a melhor maneira que você achar no software pra me dar essa estratégia com exatidão.

Entrevistadora (02:25)

Tipo, uma dúvida, quando foi construída a maior parte do software, tipo assim, o *core*, as estratégias como que, como vocês passavam o que queriam?

Cliente (02:37)

A história dele era o seguinte, então vamos supor que eu tinha que fazer uma ordem over the day. Eu ia lá acertar o horário que eu estava começando, num Excel, tendeu? Puxava a célula do Excel para executar, por exemplo, a cada quando era um Excel um era um negócio tão acurado, não era tão perfeito. Por quê? Porque eu ia ver quanto que eu ia gastar menos o meu tempo para fazer.

Cliente (03:01)

Então, por exemplo, no Excel, eu não ia colocar uma ordem de 20, 50 ou 60 mil ações de cem em cem que é o mínimo que eu posso executar. Eu ia colocar de 1000 em 1000 de 2000 em 2000, então ela não saía tão perfeita como é hoje. Hoje o sistema me dá possibilidade de colocar de 100 em 100 porque não tem o meu input, a cada 100.

Cliente (03:24)

Então ele [o sistema] veio de uma planilha de Excel, então a maioria das estratégias vieram de uma planilha de Excel. Então, por exemplo, essa que eu ia lá colocava um horário do Excel, puxava e via de quanto em quanto tempo eu precisava fazer até às 6:00 da tarde, pra dar uma ordem de 50 ou 60 mil over the day. Então eu tinha uma outra ordem que o cliente virava e falava assim: eu quero que você faça 20% do mercado.

Cliente (03:54)

Eu ia lá ficava olhando quanto que saiu e atualizando aquela planilha toda hora pra saber quanto que eu tinha que ter feito. Então, no final, a minha ideia era: se eu tinha que fazer 60000, esperava sair, por exemplo, 600 mil ações, se eu fosse 10% do ser do mercado e aí, eu ia fazendo a cada vez que eu fosse atualizando [a planilha] pra tentar atingir aquele percentual que o cliente pediu.



Entrevistadora (04:20)

Mas como você explicava tudo isso pro desenvolvedor?

Cliente (04:23)

Dessa maneira que eu tô te falando, então tipo, falava “eu preciso de uma ordem que vá ler quanto saiu no mercado e aí ele [sistema] vai executando conforme vai saindo, então se eu preciso, se o lote mínimo é cem, eu preciso que a cada 1000 saiam 100, se eu for fazer uma ordem de 10% do mercado. Preciso que a cada 1000 saiam 200, se eu for fazer uma ordem 20% do mercado”. E aí, a base eu sabia que vocês tinham essas informações publicadas pela bolsa, então é só você ir lendo lá quanto saiu e você ia fazendo o percentual do mercado.

Cliente (05:09)

Então a gente passava pro desenvolvedor exatamente isso: “olha, preciso de uma ordem que execute aqui a cada 100 tickers do horário, de um determinado horário que vou colocar até o horário final de pregão. Então eu preciso de uma ordem que a partir do momento que eu set ela, ela zere tudo como se ela zerasse, tudo que saiu no mercado e daquele momento pra frente, ela fosse fazendo 20, 30, 40 ou 50% do volume, conforme e eu solicitei a cada 100 ações. Então ele entrava no mercado a cada 100 ações.

Cliente (05:31)

Então essa era a maneira com que a gente pedia as estratégias que a gente que a gente tava querendo. Obviamente as estratégias não vinham 100% redondas, voltava do desenvolvedor uma estratégia na qual a gente aperfeiçoando ela, então ela partia de uma base muito simples que era essa.

Entrevistadora (05:53)

Uma dúvida: o desenvolvedor tinha acesso à planilha de Excel?

Cliente (05:56)

Tinha.

Entrevistadora (05:57)

Ele conhecia a planilha de Excel, como ela funcionava e tudo?

Cliente (05:56)

Ele conhecia. Conhecia.

Cliente (06:02)

Então é o que acontece é vamo tomar como base em uma ordem de percentual do volume, então eu pedia uma ordem: “olha, eu quero que essa que essa estratégia, quando eu der um play nela ela vai ler no mercado quanto que tá saindo, a partir daquele momento que eu dei *play*, e vai fazendo o percentual do volume que saiu no mercado, percentual que eu queira.

Cliente (06:26)

Obviamente obedecendo algumas coisas que eram pré definidas, algumas coisas que são do mercado já. Então eu só posso negociar a lotes de cem, então obedecendo tudo isso e obedecendo os limites de preço e tudo mais ela vai executando. Aí vai tendo as necessidades. “Ah... Eu tenho a necessidade agora de por um limite de preço...”

Cliente (06:48)

Porque de repente é um papel líquido que ela vai sair agressivando o papel. Eu não quero, eu quero que execute 20% do volume até um determinado preço. Se tiver acima, se eu tô comprando e tiver acima daquele preço que eu coloquei, eu não quero mais que faça eu quero que ela pare. E fique lá parada.

Cliente (07:08)

Aí veio outra demanda, ela fica parada e ela vai contando o que tá saindo no mercado ou ela fica parada e só começa a contar de novo a partir do momento que entrar no preço? Entendeu? Então aí a gente já teve uma possibilidade. Então, quero os 2 jeitos, eu quero o jeito que ela continue e eu quero jeito que ela fica parada, então já teve mais um desenvolvimento.

Entrevistadora (07:32)

O que me parece é que sempre tinha assim é... era sempre meio que um processo iterativo sabe? Entregava um pouquinho aí complementava, entregava um pouquinho e complementava.

Cliente (07:43)

Tinha, tinha a entrega do pouquinho e o complemento, mas tinha muito também entrega o que eu tô pedindo e aí tinha o aperfeiçoamento, que não foi solicitado na primeira vez. Então era muito de aperfeiçoar. Então eu usava aquela estratégia, por exemplo, colocou a estratégia percentual do volume e colocou o limite de preço, colocou lá como que vai participar se bater aquele limite de preço? Beleza! Roda a estratégia e vê se precisou de mais alguma coisa.

Cliente (08:09)

E aí rodava a estratégia por 3, 4, 5 meses. Aí bateu 5 meses teve alguns clientes que começaram a pedir o seguinte: vamos supor que um papel custe R\$7,00, eu deixei essa ordem de compra com percentual de 20% do volume, pra comprar até R\$7,05. Só que aí começou a ter necessidade o seguinte, se o papel cair do 7 para o R\$6,50, eu quero executar tudo, eu já não quero mais fazer o percentual do volume.

Cliente (08:46)

Eu não quero fazer nada disso, eu quero executar tudo. Então vamos implementar mais uma funcionalidade nessa estratégia que é a seguinte: se bater um determinado preço que eu colocar na estratégia, ele executa tudo. Se não, ele continua com a ideia original que foi colocada. Então assim, existiam várias maneiras de você ficar implementando.

Cliente (09:08)

Então eram várias melhorias. A gente chamava de melhoria. Eram várias melhorias que a gente ia solicitando nas estratégias. Mas o ponto principal da estratégia, a primeira solicitação, ela sempre é válida, ela sempre vai valer. Não é uma questão de entregar um pedacinho e depois ir entregando o resto, era mais uma questão de entregar um pedacinho e surgirem necessidades diferentes.

Cliente (09:32)

Mas a ideia principal do percentual do volume continua. Então eram só mais ideias do tipo: “coloca isso”, “coloca aquilo”, “acho que isso vai ser legal, porque um cliente pediu”.

Cliente (09:45)

Às vezes tinha demanda nossa mesmo, a gente que queria colocar aquilo para oferecer, e muitas vezes tinha demanda de cliente. Só que tinha um pouco de sensibilidade nossa de saber se aquilo ia de fato ser um negócio legal ou se alguma coisa que o cliente tá pedindo e nunca mais ia usar. Então essas coisas do cliente pedir, pô, e eu achar que isso não vai ser legal. Acho que isso aqui não vai ser uma funcionalidade que vai pegar e que outros clientes vão querer, então não vamos pedir. Agora, obviamente, às vezes a gente errava também.

Cliente (10:17)

A gente achava que aquilo ia ser mega demandado por clientes e aí era um outro cliente que pedia e nunca mais pedia.

Entrevistadora (10:23)

Tá entendi. Acho que ficou claro para mim. A última dúvida é, eu imagino como seja mas..., sempre foi presencial né? Os pedidos, tipo assim, na conversa assim.

Cliente (10:37)

Sempre. Sempre presencial

Entrevistadora (10:38)

Nunca foi uma coisa assim, por escrito, um documento formal...

Cliente (10:41)

Não, não. Sempre presencial.

Entrevistadora (10:43)

Sempre assim na conversa, às vezes no café ou coisa assim.

Cliente (10:46)

Isso, exatamente. O máximo que tinha de interação tecnológica era tipo: “vou te mandar a planilha de Excel pra você ver como funciona”.

Entrevistadora (10:55)

Tá sim. boa, beleza. Passando pra segunda pergunta. De que forma você descreveria a relação interpessoal da equipe de desenvolvimento da corretora com a equipe da mesa de operações?

Cliente (11:13)

É legal que a gente até pegou um gancho do que terminou, é exatamente isso. Eu vivi 2 momentos aqui, eu vivi uma corretora que era [Nome do primeiro banco] e vivi um momento corretora [Nome segundo banco, que comprou o primeiro]. Por que eu estou falando isso? Porque eram momentos apartados, assim eram...eram estruturas apartadas, então eu tinha uma estrutura de tecnologia para a mesa de operações, que não ficava na mesa de operações.

Cliente (11:48)

E aí, quando eu vim pro [segundo banco] a gente tinha uma estrutura de tecnologia de mesa de operações que ficava dentro da mesa de operações.

Entrevistadora (11:53)

Igual é hoje?

Cliente (11:54)

Igual é hoje. Então o que acontece, isso é muito diferente. Porque o cara que tá dentro da estrutura da corretora vivendo ali o dia a dia da mesa de operações, ele entende a nossa necessidade. Muitas vezes ele entende o problema e ele sabe muito mais, com muito mais clareza e mais facilidade o que ele tem que fazer para me ajudar com aquilo.

Cliente (12:20)

O oposto era muito ruim. Porque primeiro, o fornecedor no caso, ele não tinha uma

ideia de como era mercado. Eu tinha que ensinar o mercado pra ele, mostrar como aquele mercado funciona pra depois ele pensar como ele desenvolveria aquilo, talvez ele não tivesse alguns vícios e de repente saísse alguma coisa fora da caixinha, mas era muito difícil.

Cliente (12:47)

Era muito difícil, então a conversa quando a conversa quando a mesa está dentro, quando o pessoal de tecnologia está dentro da corretora e tá vivenciando o dia a dia é muito mais fácil do que quando tá fora da corretora. Isso não tem, não tenho nenhuma dúvida. E aí a relação interpessoal que demanda, é o dia a dia, é falar com as pessoas.

Cliente (13:11)

É mostrar um pouco do que a gente faz não só porque a gente quer desenvolver alguma coisa, mas mostrar um pouco do que a gente faz, pra vocês saberem como funciona como que é o dia a dia. E muitas dessas coisas acabam saindo até alguns insights de vocês do tipo “pô, mas não é melhor fazer desse jeito ou de outro jeito?” Porque você tá entendendo um pouco do meu business, você sabe mais ou menos, como é que as coisas funcionam, então esse relacionamento é muito melhor quando tá [todo mundo] aqui um do lado do outro e não separado

Entrevistadora (13:45)

Agora na terceira pergunta, eu acho que é tem um pouco a ver também de, o que eu quis perguntar aqui é: quanto que você também sabe falar nossa linguagem de desenvolvedor? Então a pergunta oficial é: qual a sua experiência e habilidade quanto a utilização e compreensão a respeito do funcionamento do software, da sua modelagem e dos erros já apresentados ao longo do tempo, desde o lançamento em produção? Se você quiser começar a contar, você começou a contar falando que surgiu de uma planilha de Excel, então assim, você, imagino, dominava muito aquela planilha, então partindo daquilo.

Cliente (14:28)

É que a planilha de Excel era um negócio muito simples, tinham planilhas mais complexas tinham planilhas mais simples, então tinham essas 2 partes. O domínio

do Excel é um pouco diferente do domínio aqui da ferramenta que a gente tem hoje. Eu não peguei muito essa parte do Excel, peguei pouco, mas sei que veio do Excel e usava um pouco disso também para poder trabalhar.

Cliente (14:58)

Ah, mas assim conhecer a Excel é um pouco forte, porque eu conhecia as funcionalidades que o Excel me fornecia para eu poder fazer isso aí. Então por exemplo, dividi um horário é você puxar, cê põe um horário na célula de baixo cê põe 2 minutos a mais, desce ali e acabou. Você sabe que de 2 em 2 minutos você tem que fazer aquele trade.

Cliente (15:21)

Ou um cálculo de percentual que você vai fazer, era algo muito simples. Tinha problemas mais complexos? Tinham problemas mais complexos, onde você trocava um papel [ação] pelo outro, sabia que se eu fizer tanto disso, eu tenho que fazer tanto daquilo, e tudo está dentro do sistema hoje. Então, tinham coisas mais complexas.

Cliente (15:41)

Agora falando um pouco assim... Eu acho que a minha compreensão do sistema ela é talvez um pouco maior porque eu vim de uma formação colegial de TI, mas desisti no meio do caminho, cheguei a fazer tipo, sei lá, eu fiz um ano de ciência da computação e parei. Então eu acho que tem algumas coisas que eu consigo ver e consigo pensar às vezes, obviamente a tecnologia de lá para cá voou.

Cliente (16:05)

Mas a lógica está um pouco dentro [de mim], então você consegue [dar idéias]: “pô, acho que era só fazer isso”. Muitas coisas que eu peço pra vocês eu peço no sentido “pô, mas é só colocar alguma coisa assim aqui e funciona”.

E aí vocês voltam e falam “não, não é desse jeito, isso é um puxadinho. Daria pra fazer mas é puxadinho. Tem o que lá no código, que é gigantesco, encontrar aquele pedaço, analisar aquele pedaço e colocar isso que cê tá falando. Você tá certo, mas tá errado em pensar que é só colocar lá e acabou”,

Cliente (16:24)

Então assim, tem muitas coisas que eu penso, que talvez meu pensamento esteja certo, mas a execução dele é completamente errada. Assim, o jeito que eu penso de executar é errado. Quanto ao sistema hoje, ele é muito friendly, então ele é muito amigável, cê consegue é... pra gente que tá no contexto, para a gente que está na mesa de operação e que sabe como as estratégias funcionam, ele é muito fácil de ser usado, entendeu?

Cliente (17:00)

Obviamente tem um monte de funcionalidade ali que, qual é o defeito, não sei nem se seria um defeito, mas... qual é o problema de você começar a colocar muita coisa nele [sistema]?

Cliente (17:15)

Tem coisa ali que, honestamente, eu não sei pra que serve. Então eu uso? Não. De repente se eu tiver alguma demanda, eu olho e faço assim: “ah essa demanda faz sentido pra mim”. Então eu vou usar isso aqui. Mas tem vários campos hoje, por exemplo, que alguém pediu lá atrás que era justamente o que eu estava te falando, que é aqueles campos do tipo que um cliente uma vez pediu, acharam que todos os clientes iriam pedir, mas nenhum cliente pediu mais e aí você acaba deixando lá.

Cliente (17:44)

Ele fica no sistema, mas ninguém usa mais.

Entrevistadora (17:47)

Mas falando assim do domínio do software mesmo. Sei lá, eu já presenciei algumas situações em que, falando de situações de erro, de bug, ou mesmo de novas implementações que ainda não deram certo sabe, estão em processo, ou é alguma entrega nova que você tá testando. Eu vejo e vi situações assim que você ou o próprio [colega de trabalho do cliente] falam “putz não deu certo, mas eu acho que o erro tá aqui”.

Ou em situações de erro de produção de falar não o erro tá aqui ou, muito provavelmente, tá acontecendo isso, isso e isso de errado. Tipo, eu sinto que vocês



têm essa sensibilidade. Não sei se você sente isso também.

Cliente (18:33)

Tem, a gente tem essa sensibilidade até pra saber o tipo. É que às vezes o tipo de erro que volta a gente sabe em qual parte do caminho uma mensagem parou. Então, sei lá, a gente, por exemplo, sabe que se deu um timeout hoje é porque provavelmente caiu algum servidor ou caiu alguma coisa. Então se deu algum erro, deu um erro “*duplicate* alguma coisa” a gente sabe que a estratégia meio que se perdeu ali e ela tá duplicando alguma informação ou aqui [Brasil] ou no mercado lá fora, que é normalmente quando dá essa estratégia de duplicação de ordem.

Cliente (19:05)

Ele [sistema] mandou alguma ordem lá fora que não foi executada, tá tentando mandar de novo e não vai conseguir. Alguns erros a gente já está familiarizado com o erro, então a gente acaba tendo esse pré-diagnóstico do que do tá rolando e do que aconteceu. Ou algum tipo de erro também, que não me vem à cabeça agora, mas que no decorrer da mensagem, eu sei mais ou menos onde vai dar aquele erro, mas talvez não é por conhecimento do software.

Cliente (19:28)

Na verdade, é justamente por saber toda a linha de processo que o comando vai ter. E aí, eu falo pô, esse parece ser um erro, por exemplo, do [nome de uma aplicação], que é o sistema lá de fora, então eu sei que esse erro do sistema lá de fora, por exemplo, bateu lá fora, então é um erro em um ativo gringo.

Cliente (19:49)

Não é um erro no ativo local. Então isso acontece bem. Essa sensibilidade do sistema e, assim, a gente usa o sistema há não sei quanto tempo também, então é bem...

Entrevistadora (20:01)

É eu, por exemplo, eu nunca vi, aliás até já vi, mas assim poucas vezes, vocês gritando “deu erro” aleatoriamente. Vocês geralmente falam “puts deu um erro aqui em tal parte, fazendo tal coisa”. Assim vocês já vão direto ao ponto, não é genérico.

Cliente (20:17)

Isso, e o sistema ele se divide muito bem, então assim eu consigo identificar onde deu erro, por exemplo, eu tô executando 30 estratégias, das 30 estratégias têm 3 estratégias que é over the day, e deu erro é over the day.

Entrevistadora (20:32)

E você sempre, desde o começo, já foi capaz de identificar assim, aonde estavam os problemas e onde podia melhorar?

Cliente (20:40)

Não, não. A gente foi aprendendo com vocês.

Entrevistadora (20:43)

No começo não era assim?

Cliente (20:46)

Não era. A gente foi meio que aprendendo com vocês mesmo da onde vinham esses erros, o que poderia ser e aí, por exemplo, a gente sabe que tem um determinado erro hoje que se der ferrou porque ele [sistema] se perde inteiro e ele não consegue recuperar o passado. Caiu o servidor, caiu o servidor acabou, tipo assim.

Cliente (21:04)

Se cair a gente sabe que ele não vai conseguir recuperar as informações, então a gente já para tudo, sai correndo pra gente resolver tudo manualmente, porque sabe que ele não consegue capturar os históricos do dia. Para poder fazer as atividades futuras que ele [sistema] teria que fazer.

Entrevistadora (21:19)

E aí, você falou uma coisa na sua resposta agora que é totalmente a pergunta quatro que é: como você categoriza ou explica o seu grau de experiência em seu domínio de atuação. Se você quiser falar um pouco a partir da parte que você falou “ah, eu já entendo muito do processo da ordem, já entendo muito pra onde ela vai, se ela é gringa ou não é...”

E isso ajuda muito o desenvolvedor na hora de entregar coisas novas ou resolver problemas.

Cliente (21:48)

É que assim a gente vem do mercado lá de traz, então a gente cresceu junto com o mercado. O mercado foi evoluindo e a gente foi evoluindo também. Então o que aconteceu, você pega todas essas transições, então eu usava um sistema a bolsa mudou, começou a usar outro, aí a regra mudou começou a ser de outro jeito.

Cliente (22:08)

Por exemplo, esses dias eu falei pro [desenvolvedor] não, tá errado, eu posso passar quantos diretos eu quiser, eu tenho que me explicar para a bolsa se eu passar o lote mínimo e o porquê eu estou passando aquele lote, aí ele falou que eu não podia. Depois a gente foi confirmar e viu que eu poderia e eu teria que me explicar depois realmente

Cliente (22:29)

O que acontece, a gente vai evoluindo junto com o sistema, então o sistema vai assim e a gente vai evoluindo.

Cliente (22:54)

O sistema [eletrônico da bolsa, para envio de ordens no Brasil] vai evoluindo e a gente vai entendendo todo o processo que ele que ele faz pra ordem chegar onde ela tem que chegar. Quais os sistemas que eu uso lá fora, por exemplo, agora a gente vai deixar de usar um sistema X para usar um sistema de um outro fornecedor. Aliás, um sistema não, um serviço, então vai mudar a minha rotina.

Cliente (23:13)

Não tenho dúvida de que eu pego os dados de um sistema X hoje e eu vou ter que pegar os dados de um sistema Y, porque vai mudar o fornecedor daquele serviço, consequentemente, os *inputs* dentro do sistema e os *outputs* do sistema vão mudar, porque vai mudar o fornecedor. E eu vou ter que aprender a pegar aquilo de outro jeito. Hoje eu sei como pega, mas eu vou ter que aprender, então assim eu vou ver o caminho de novo como que faz.

Cliente (23:41)

Eu vou entender o sistema e aí eu acho que é onde tá a ponta mais importante, a gente sempre tentou entender o que o sistema faz, a gente não se limitou a pedir para vocês, por exemplo “eu quero que o sistema faça isso”. Aí se ele não tá fazendo, pede pra corrigir, porque a gente via isso no passado, a gente via muita gente pedindo dessa forma. A gente teve outro ponto muito relevante que a gente sempre teve pessoas dentro da equipe que “manja” do mercado.

Cliente (24:13)

E não é “manjar” que nem a gente atualmente, que fica nas mesa de operações 24 horas do horário de trabalho, é manjar que eu digo no sentido de “eu sei como funciona, sei o que vocês fazem e sei das suas necessidades, só não sei como vocês vão colocar isso no sistema e como vocês agressivam o mercado, eu não sei como prever o comportamento de um mercado, isso é com vocês”.

Cliente (24:30)

Mas eles sabiam como que funcionava o dia a dia ali. Isso facilita bastante e a gente também do nosso lado sempre tentou entender qual é o caminho que o sistema faz qual é a lógica que ele usa. Pra gente poder entender o que está acontecendo e como que ele vai se comportar, porque não adianta nada eu pedir pra você faz uma estratégia X e aí se eu não souber como ela tá se comportando.

Cliente (24:58)

Aí eu vou lá ligo aquela estratégia sai fazendo barbaridades no mercado só que eu não sei qual é a barbaridade que ela tá fazendo, porque eu não me preocupei em saber qual era o “pensamento” dela, qual era o raciocínio dela. A gente sempre se preocupou em saber também como que a estratégia vai funcionar o que que ela vai fazer pra atingir o objetivo que eu quero e isso pra gente faz toda a diferença.

Cliente (25:26)

Então tem uma parte nossa ali de querer saber a inteligência da estratégia para poder ver como aquilo vai funcionar e aí, obviamente, no fluxo inteiro do programa a gente consegue ver onde ele parou, onde tem um erro e onde não tem.

Entrevistadora (25:36)

Então me fala de forma meio breve, o que você puder falar também, quantos anos de casa e de mercado você tem? [Risos]

Cliente (25:49)

Assim, eu entrei em 2002 aí eu trabalhei 2 anos, e aí eu vou te contar porque isso faz um pouco de parte de você entender todo o fluxo do mercado. Eu entrei em 2002 trabalhei um ano numa área chamada controle de clientes que nada mais era do que abrir e fechar conta de cliente.

Cliente (26:10)

Mas tinha uma pequena operação de custódia que, quando o cliente deixa as operações aqui com a gente, você fica olhando aquelas operações e acompanhando aquelas operações dos clientes. Um ano e meio depois eu fui pra área de backoffice e middle office de operações de futuros. Por que backoffice e middle office? Porque a corretora era pequena e ela não dividia as operações.

Cliente (26:32)

Então eu cuidava das operações que o cliente estava fazendo um dia, cobrava o cliente no dia seguinte o que ele tinha que pagar, ou o que ele tinha que receber eu pagava, depois cobria a margem daquele cliente. Hoje são funções feitas por 2 áreas distintas, então a gente pegou toda essa parte.

Cliente (26:54)

Quando teve a compra do [Nome primeiro banco] para o [Nome segundo banco] eu ficava na mesa de futuros, e o [Nome segundo banco] não fazia [o mercado de] futuros, então ele [segundo banco] resolveu que iria fazer futuros a partir daquele momento da compra. Então eles levaram 2 pessoas do banco antigo, eu e o operador. Então eu fazia toda a parte de backoffice e middle office e o operador fazia as operações. Dois anos depois o operador resolveu sair e eu assumi o lugar dele, fazendo só operações de [mercado de] futuro.

Cliente (27:20)

Ah, e nesse meio tempo eu fiquei fazendo middle office de operações de futuros e do [mercado] “à vista” junto com o pessoal da corretora, então era tudo uma estrutura só.

Entrevistadora (27:28)

Isso era por volta de?

Cliente (27:30)

2008 ou 2009.

Cliente (27:32)

Depois disso, aí eu já não vou lembrar mais as datas, mas eu fui pra parte de futuros. Nessa parte de futuros eu comecei a agregar mais coisas. Então, assim, eu comecei a atender o private banking nas operações de futuros. Aí como ficava uma pessoa atendendo [mercado] futuros e outra pessoa atendendo o [mercado] à vista, então eu migrei e fiquei atendendo o private banking inteiro.

Cliente (27:54)

Aí peguei toda a parte de exercícios de opções para fazer, peguei toda a parte do private, peguei o Market maker também pra fazer, que é o nosso core business hoje. E aí fiquei com toda essa gama de atividades pra fazer.

Cliente (28:17)

E tudo isso usando o [nome do software deste estudo] como canal de acesso à bolsa. E agora a gente migrou novamente, praticamente a gente não faz mais operações via [participante] institucional, então todas essas estratégias que eu tô te falando a gente usa pouco. Usa mais as estratégias de *market maker* que, aí sim, muito mais do que as outras, são estratégias que a gente tem que tá ali no dia a dia.

Cliente (28:43)

Tanto da execução dela, quanto de acompanhar o que está sendo feito, pedir melhorias... Então essas, mais do que nunca, a gente tá muito mais próximo do que as outras. Porque as outras eram meio play e pronto. Então era muito mais fácil de fazer alguém entender. Já essas de *market maker* são um pouco mais complexas e

os pedidos em cima dessas estratégias são um pouco mais complexos do que as primeiras.

Entrevistadora (29:10)

Então já vai completar 23 anos de experiência no começo do próximo ano?

Cliente (29:19)

No meio do ano que vem fará 23 anos.

Entrevistadora (29:20)

Agora eu queria entender mais, falando da parte de quão satisfeito você está e esteve ao longo desses anos que você usou o software como cliente. Então, na posição de cliente da equipe de IT, como que você descreve sua experiência com o uso do software?

Cliente (29:48)

Tá, eu vou emendar as 2 últimas perguntas.

Entrevistadora (29:50)

Tá bom, e a satisfação com o resultado do projeto de uma forma geral.

Cliente (29:53)

Uhum.

Cliente (29:55)

A experiência com o uso do software foi que eu te falei no começo, ele é muito friendly. É muito fácil de usar. É tipo um como se tivesse um *iPhone* entendeu? [Risos]. É bem diferente do Android, é mais fácil de usar. E pro usuário é muito bom porque ele é fácil de usar e se ele fosse um pouco mais complexo acho que vocês teriam mais problemas também. Acho que vocês nunca precisaram fazer, por exemplo, um curso para ensinar alguém usar entendeu? E quando a gente vai ensinar uma pessoa a usar, ele é muito muito rápido de aprender.

Cliente (30:31)

Assim, ele é muito, como eu posso dizer, ele muito intuitivo. Então é super tranquilo de usar. A gente, por exemplo, numa estratégia de market maker, precisa de quantidade e spread, então é muito mais difícil ensinar pro cara o que é um spread do que usar o sistema em si. Depois que entende o que é spread, você vai no sistema coloca todos esses dados, porque ele sabe o que é um spread agora. Então o sistema mega intuitivo e muito fácil de usar.

Cliente (31:07)

E atende as expectativas que a gente espera. Um outro ponto muito bom dele é ter vocês perto. Então, assim, se [vocês] fossem uma célula que ficasse separada da mesa [de operações] ou em qualquer outro lugar... Se bem que hoje, por exemplo, eu acho que se fosse uma célula que ficasse fora da mesa, não teria tanto problema, por que temos um sistema maturado, você tem um sistema que já está consolidado, eventualmente precisa de uma ou outra melhoria, mas todas as pessoas [IT e mesa] sabem como o sistema funciona.

Cliente (31:38)

A impressão que vocês passam é que vocês sabem como o sistema funciona. Eu imagino que aquilo seja tipo um mapa no Google Maps e aí se eu falo endereço, você sabe onde você vai, então isso facilita bastante. Agora entra um cara novo de IT você fala alguma coisa ele fica tipo, “cara estou perdido aqui no meio de São Paulo e vim da Bahia, eu não sei o que fazer”.

Cliente (32:08)

Então isso facilita bastante quando a gente pede alguma coisa, a gente vê que as demandas hoje saem rápido, porque acho que vocês sabem aonde tem que mexer para fazer, pra resolver aquele problema ou para fazer aquela demanda. Então assim a satisfação do sistema tá bem boa. Ela já teve melhor? Ela até teve por um tempinho, só que tiveram problemas depois, por exemplo, a gente teve os “puxadinhos” né, quando um desenvolvedor antigo queria entregar muito rápido. Então, assim, o atendimento era melhor? Era. Porque você pediu uma coisa ela saía no dia seguinte, só que 2 meses depois, começava a dar uns pau do nada e a gente não sabia do que era.



Cliente (32:47)

E ia ver, era justamente aquela demanda que foi atendida mais ou menos, em um dia. Então, assim era uma falsa alegria ali, era um falso sentimento de que tava dando tudo muito certo e que estava indo tudo muito rápido. Então agora hoje a gente entende que uma demanda nova no sistema pode demorar um pouco e que a demora em si não é no desenvolvimento dela, às vezes é muito mais como você vai encaixar aquele desenvolvimento.

Entrevistadora (33:21)

E em fazer o que precisa ser feito? Assim, em termos de fazer as operações que você precisa que sejam feitas?

Cliente (33:29)

Isso funciona super bem. Eu acho que hoje eu posso te dizer que a maioria das demandas que a gente tem são demandas de aprimoramento para facilitar nossa vida e nosso dia a dia, porque a gente tá cada vez com menos pessoas na área.

Entrevistadora (33:40)

Mas não demandas por causa de erros?

Cliente (33:42)

Não são de erros.

Entrevistadora (33:43)

Ou você vê muitos erros?

Cliente (33:45)

Não, não, não, são poucos. E aí tem um negócio do usuário que é muito bom, a gente soube trabalhar os erros que tinham. Então, por exemplo, eu sei que tem um erro, que ele, às vezes, por ser um sistema pesado, de corretora e robusto, ele perde velocidade para outros tipos de software.

Cliente (34:13)

Então quando tem algum ativo que depende do dólar, o dólar se mexe rápido e a

hora que ele [estratégia do software] vai começar a se mexer, já foi passado pra trás. Então aí tem uma outra atividade nossa, que é o seguinte: vamos escolher ativos ou vamos dar spreads um pouco mais largos, por que esses spreads mais largos ou esses ativos menos demandados ou menos líquidos dá esse tempo que eu preciso..

Cliente (34:39)

Ele dá o tempo do meu sistema se mexer, mesmo que o outro [participante do mercado] se mexa o delay dele [software em estudo] não vai me atrapalhar o delay dele não vai me causar prejuízo. Então tem toda essa parte nossa também, de aumentar um pouco mais esse spread, porque esse aumento de spread é delay no sistema.

Cliente (35:00)

Ou seja, vamos deixar de fazer uma Apple ou uma Microsoft, vamos deixar de fazer uns ativos que são mega líquidos lá fora, para fazer alguns ativos um pouco menos líquidos, porque ele abre um pouco mais o spread e é possível eu me movimentar tempo de me proteger.

Entrevistadora (35:17)

Isso te prejudica? Deixar de fazer essas coisas?

Cliente (35:20)

Ele não prejudica, porque são ativos super líquidos, então sempre vai vir alguém perguntar se eu quero fazer ele ou não. Mas eu não quero ficar sendo o market maker dele especificamente, porque isso me prejudica. Porque todo mercado se mexeu e eu demorei para mexer, e a gente tá falando de milisegundos, então em poucos milisegundos, todo mercado se mexeu e eu não, então eu fui a presa fácil pro mercado. Obviamente, dentro de tudo isso, todos esses acontecimentos foram bons, porque a partir desses acontecimentos, por exemplo, foi que a gente pediu travas pra vocês.

Cliente (35:53)

A partir disso que a gente pediu maneiras de se proteger, foi que a gente pediu pra estratégia parar sozinha se de repente, um determinado fato acontecer. Então, isso

permitiu pra gente colocar várias travas que deixa a gente mais confortável hoje, mas ainda assim se a gente puder evitar dessa maneira, abrindo spread ou trocando os ativos é melhor.

Entrevistadora (36:16)

Você se sente seguro usando [o software]? Ou se sente exposto ou vulnerável?

Cliente (36:21)

Quando a gente tá falando de uma atividade muito relevante no dólar, de volatilidade muito grande no dólar, eu me sentia um pouco mais vulnerável. Hoje em dia a gente viu três dias, os últimos três ou quatro dias, a última semana inteira na verdade, a gente viu uma volatilidade absurda no dólar assim, muito, muito, muito volátil por conta, por exemplo, de eleição e pacote do governo que não sai.

Cliente (36:50)

E mesmo assim a gente viu que não fomos uma presa fácil. A gente saiu perdendo sei lá 60 USD um dia, ganhava 10 dólares no outro, perdia mais 60 UDS no seguinte. E isso tá dentro do nosso nível de perda, porque a gente tem uma estratégia de não passar [a virada do dia] posicionado e essa estratégia custa um preço, porque se alguém me executou.

Cliente (37:12)

Porque se alguém me executou e eu estava cobrando um spread, eu vou pagar esse spread para me zerar [no mercado]. Isso é uma estratégia nossa, então pra mim vale muito mais a pena pagar 60 do que ficar tendo um operacional absurdo durante o dia. Então a última semana deixou a gente muito contente e confortável em relação a esse delay. Obviamente a gente trocou muito papel que a gente tinha.

Cliente (37:39)

Então o que a gente faz hoje, ah esse papel X aqui deu um prejuízo grande; foi a oscilação do dólar? Se sim, então a gente anota esse papel e da próxima vez que tiver uma seleção de papéis e ele aparecer de novo pra ser Market maker, a gente já excluí ele. A gente não entra nele, a gente procura outro. E assim a gente vai selecionando melhor os papéis para evitar esse problema.

Entrevistadora (38:02)

Entendi, obrigada. Quer falar mais alguma coisa?

Cliente (38:06)

Não.

Entrevistadora (38:07)

Não?

Cliente (38:08)

Não, acho que foi isso.

Entrevistadora (38:10)

Obrigada.

## APÊNDICE B – Transcrição da entrevista com o desenvolvedor

Entrevista com o desenvolvedor. Este apêndice foi desenvolvido pela autora.

Entrevistadora (00:07)

Começando a entrevista com o desenvolvedor da equipe de Eletronic Trading. A primeira pergunta é: como se dava a presença e a participação dos clientes, no caso os *traders*, durante a construção do software estudado?

Desenvolvedor (00:28)

Bom, assim falando do [nome da aplicação estudada], a gente já tinha uma plataforma legado... tinha uma aplicação anterior que era legado e os clientes já usavam. Então foi uma evolução tecnológica que a gente estava fazendo. A antiga já não servia mais, ela era limitada, ela ficava rodando na máquina do próprio usuário.

Desenvolvedor (00:57)

Então, a gente tinha uma demanda de fazer uma plataforma que fosse escalável, centralizada, né, tipo, aliás, a execução centralizada, o controle de centralizar.

Desenvolvedor (01:25)

Deixa eu tentar colocar melhor, desde o início do projeto, assim, o que foi o projeto? O projeto foi uma demanda de continuar sendo um dos primeiros no mercado a ter tecnologia para competir no mercado de algoritmos, que é extremamente competitivo. E as aplicações que a gente tinha, que era legado, não suportava, não é escalável, era *single thread*, então não tinha performance, ficava limitada à capacidade da máquina do usuário. Dava um pau era na máquina do usuário parava tudo, então assim, era difícil de debugar, de entender o que tava acontecendo.

Desenvolvedor (02:10)

Então, a demanda de implementar uma coisa nova partiu da nossa própria equipe. Porque já é um exemplo de coisa que o usuário não sabe que ele tem e que ele

precisa, entendeu? Ele não sabe distinguir, por exemplo, se a aplicação tá rodando na máquina dele ou não tá.

Entrevistadora (02:26)

Mas ele tinha alguma reclamação?

Desenvolvedor (02:28)

Ele tinha reclamações, tipo, ele mostrava aqui, olha tá ruim aqui, por exemplo, chegou um dia que eles chamavam “oh não tá executando”, aí de repente “ah executou”. Porque ela [estratégia] ia uma a uma executando. Então, assim, ele chegou ao ponto de ter tanta estratégia colocada na tela que... antes, para quando a aplicação foi feita, era, sei lá, pra 10 [estratégias], mas ele tava com 100 na tela. A primeira voltava executar depois que a última terminasse, então aquilo demorava vários segundos e ele não via a estratégia dele funcionando executando e falava que tava parado.

Desenvolvedor (03:00)

Na verdade tava processando as outras e aí voltava lá em cima, então essa era a percepção que o usuário tinha. Então a percepção do usuário é o que ele tá vendo na tela, ele não enxerga por trás como a aplicação funciona. Então o nosso trabalho era abstrair essa visão do usuário. De fato, ele não precisa saber o que tá acontecendo por trás, mas ele quer o resultado, ele quer aplicação respondendo, ele quer aplicação performando.

Desenvolvedor (03:26)

Ele não quer saber se ele tá com 25 mil planilhas abertas na máquina dele, mais um monte de terminal do *bloomberg*, mais um monte de coisa comendo memória e recurso e a máquina dele tá lenta e a aplicação não tá funcionando direito. Isso ele não quer saber, ele quer que a aplicação funcione. Então a gente sofreu muito com isso, que a gente ficava explicando esse tipo de limitação e tal. E aí nosso próprio time precisou vir com uma ideia de melhorar isso, que era colocar uma plataforma que fosse escalável.

Desenvolvedor (03:54)

Entende? Que conseguisse entregar a performance que os usuários precisavam, com a responsividade que os usuários precisavam, que não importasse o que ele tivesse usando a máquina dele que a aplicação e ia continuar funcionando. Entendeu?

Entrevistadora (04:10)

Mas como que veio esse pedido?

Desenvolvedor (04:14)

Isso foi uma proposta, na verdade foi assim, partiu do nosso time. Na época o [nome do gestor] tava aqui como gerente, ele chamou a gente e falou “olha a gente tá precisando disso”. E aí o time sentou, discutiu e falou “então tá, então a gente precisa desenvolver um sistema de execução de algoritmos, um EMS”. E aí a gente começou a discutir como seria, a gente começou a pensar na arquitetura. Mas uma coisa já era sabido, ele ia ser um servidor executando os algoritmos e não ia ficar mais na máquina do usuário.

Desenvolvedor (04:45)

Isso bastava pra chegar para os usuários, pros stakeholders, né, não só para os usuários, não todos também, o principal, pra gente propor isso. Falar: “Olha, a gente precisa criar um sistema novo, isso aí que está rodando a sua máquina já é legado, a gente vai fazer um novo beleza?”.

Desenvolvedor (05:10)

Aí ele “beleza, manda bala!”. O cliente comprou a ideia, falou “entendi o que a gente precisa”. Nosso time, enquanto IT, identificou a necessidade mostrou as limitações, viu que o que a gente tinha já não atendia mais.

Desenvolvedor (05:26)

Então a gente precisava de um novo, vamos fazer um novo. Porque é o tipo de coisa que o cliente não sabe mesmo.

Entrevistadora (05:32)

O legado era proprietário do time também?

Desenvolvedor (05:35)

Também, também era a gente que tinha desenvolvido, só que já tinha um pouco mais de tempo.

Desenvolvedor (05:42)

Então assim a gente começou a sentar pra pensar na solução... Esse início não dependia muito do cliente, não dependia muito dos *inputs* deles, era muito mais uma coisa técnica mesmo entre nós, entre o time, para desenhar a arquitetura da solução, como que a gente vai fazer uma aplicação que seja escalável.

Desenvolvedor (06:09)

Como que eu vou explorar os recursos que eu vou ter em um servidor de aplicação, como que eu vou fazer para que as estratégias não concorram entre os si. Na verdade elas concorrem, não tem como não concorrer. Mas como que eu vou maximizar a eficiência das estratégias considerando todas essas concorrências que elas precisam ter.

Desenvolvedor (06:26)

Entendeu? Então, foi onde a gente começou a criar o [nome do sistema estudado], pensando na arquitetura de comunicação entre servidor e cliente também, como que a tela vai funcionar como controle remoto, que é o [nome do sistema estudado] hoje. A gente desenhou dentro da aplicação além dessa parte comunicação de processos, como que a gente é persistir em banco de dados, o que a gente ia precisar persistir em banco de dados. Tudo isso é tecnologia ainda, o usuário não precisa saber disso, ele não sabe essas necessidades.

Desenvolvedor (07:00)

Até mesmo sabendo que em algum momento, do lado de cliente, a gente talvez precisasse mostrar pra auditorias, o que usuário inseriu no sistema, quem foi... Tudo



isso veio com a bagagem do que o time já tinha de conhecimento de tudo que é demandado de uma plataforma de electronic trading. Então conta a experiência que a gente tem já de saber o que a gente precisa desenvolver, mesmo sem o cliente saber. Então a gente tinha ali, já sabia que, eu tinha que logar cada comando que o usuário tinha dado, cada *input* de estratégia, manter histórico disso, configuração do cliente.

Desenvolvedor (07:35)

Ele tem lá as estratégias salvas, eu tenho que manter isso em banco dados. Se ele substituir isso eu salvo um histórico, ponho data, usuário que fez essa operação, tudo isso como persistência. Então na camada de persistência, a gente pensou em tudo isso que ia ser necessário também e desenhando a aplicação a gente também desenhou o que precisava da persistência pra aplicação subir sozinha, entendeu?

Desenvolvedor (07:56)

Tipo, derrubei a aplicação ela precisa voltar, precisa voltar de onde parou, então a gente pensou em estados também. Então assim, quando a gente desenhou toda a estrutura da aplicação já tinha lá o *framework* dela também, do funcionamento das classes, objetos, onde ia ficar a instância da estratégia X. Por que a estratégia X é diferente da Y? E aonde elas são semelhantes também?

Desenvolvedor (08:26)

A gente começou a criar as abstrações, os métodos que são as interfaces. Começou a expor só o que é interessante pro *core* do [nome do sistema] entender aquilo como uma estratégia, mas ainda assim deixar essa estratégia flexível o suficiente para ela ser desenvolvida com outra lógica.

Entrevistadora (08:44)

E as novas estratégias que vieram depois?

Desenvolvedor (08:46)

Então essas novas estratégias sempre respeitavam esse contrato, todas elas, elas tinham essas mesmas características padrão, mas ela era flexível suficiente para ela

desenvolver a própria lógica dentro dela. Isso eu já tinha feito, já tinha tido essa experiência numa outra maquininha legado que eu fiz. Eu fiz algo parecido com isso de ter uma estratégia onde ela pode ter tipo subestratégias e dentro dela mesmo ela começar... Então eu reutilizei essa ideia dentro do [nome do sistema], pensando nessa parte de consegui fazer multitarefas, concorrentes, tipo multi thread, e coloquei dentro do sistema dessa forma.

Desenvolvedor (09:26)

E tudo funcionou muito bem, então ficou escalável, a gente conseguir instanciar milhares de estratégia ao mesmo tempo, cada estratégia pode ter mais de uma thread também. Ficou flexível o suficiente, entendeu?

Entrevistadora (09:40)

E quando ia chegando novos pedidos de novas estratégias como eles chegavam?

Desenvolvedor (09:46)

Então, o que a gente fez primeiro. Como a gente já tinha essa maquininha legado, a gente a nossa primeira demanda era migrar esses algoritmos pra dentro do novo sistema, então não foi muito pedido de nova estratégia. A gente importou né, pro modelo novo, a lógica desses mesmos algoritmos uma ficou melhor do que a outra às vezes, mas também o código legado não era o estado da arte também entendeu, assim não era um negócio muito bom.

Desenvolvedor (10:15)

Mas como ela já era um negócio que funcionava daquele jeito como usuário estava acostumado, a gente fez o máximo para não mudar o comportamento de estratégia. Então a gente manteve muita coisa do jeito que tava, só importando o código e adaptando o que era necessário. E assim, nessa fase, que foi quando a gente colocou a primeira versão do [nome do software] em produção, nessa fase os usuários começaram a testar, né, em produção, depois da gente ter testado muito, começaram a testar isso comparando sempre, né, ele não saiu de uma e entra na outra.

Desenvolvedor (10:50)

Ele deixava aberto o antigo ali, colocava a nova e começava a operar. Lógico que no início, nas primeiras versões a gente teve muito problema. Começou a dar erro de alguma coisa, a gente parava, para agora a gente vai voltar pra prancheta para arrumar. Então teve isso. Mas essa interação que a gente tinha com eles era muito próxima mesmo. Por que a gente sempre teve sentado dentro da corretora e cada passo que a gente dava, de progresso que a gente fazia, a gente conversava com eles e mostrava. E aí a gente colocava uma versão nova e explicava “olha, agora testa aí, vamos fazer assim”.

Desenvolvedor (11:32)

Explicava o que é que tinha acontecido, eles entendiam, faziam o teste de acordo com que a gente orientava. E assim, era uma interação constante com eles, desde a primeira fase. Até que chegou ao ponto que, tipo não demorou muito tempo, alguns dias assim, mas chegou ao ponto de que “pô, tá legal, né?”, “tá funcionando bem”. E aí, depois começaram a vir os primeiros elogios. Falavam “nossa, tá muito rápido”, porque o negócio não rodava mais uma de cada vez mais, né, rodava tudo junto.

Desenvolvedor (12:05)

Então eles começaram a colocar mais estratégias, aí um olhava pro outro e falava “olha ali, olha ali um *snipper*, pá, bateu no preço, pá, na hora”. Tipo, eles ficavam impressionados assim, “caramba tá muito bom tal”. Só que teve também uma fase logo desse início que demorou um tempinho pra gente voltar e colocar uma versão mais parruda em produção. Porque a gente tinha a limitação de ambiente, então a gente ainda não tinha um super ambiente para mudar tudo.

Desenvolvedor (12:37)

Igual a escala que a gente precisa. Porque a gente usava máquina virtual, não dava pra pegar, parar, todo mundo parar de usar maquininha legado e partir logo pro sistema novo, que não tava bem dimensionado, entendeu? Então a gente parou um tempo, tirou de produção. Em seguida voltou, até de fato conseguir estruturar, a gente esbarrou em problemas que a gente não pegava em DEV.

Desenvolvedor (13:05)

Como, por exemplo, performance de fila MSMQ, até a gente pegar a manha de como tinha que ser o setup dessas filas, que a gente também usava máquina virtual pra elas. Então assim, a gente também teve um “puta” caminho ali para desenvolver com relação à infraestrutura que o sistema precisava pra rodar bem em produção.

Desenvolvedor (13:22)

Não era só a lógica de programação, do algoritmo ou do sistema em si. Estava muito relacionado com o ambiente, então a gente ia aprendendo isso à medida que a gente avançava, então esbarrava em alguma coisa, precisava arrumar aqui, precisava arrumar ali.

Desenvolvedor (13:47)

Até que a gente começou acertar essas coisas junto com o time de INFRA, por isso que a gente também teve essa a nossa relação com o time de infra, de servidores de redes... Tudo também foi importantíssimo, porque a gente precisava de muita ajuda deles.

Entrevistadora (14:01)

A relação era bem próxima também? Desses outros times?

Desenvolvedor (14:03)

Era. Também era bem próxima. Assim, vamos dizer que não era tão saudável essa relação, por outros fatores, tinha gente no time que... que, ah, que às vezes causava um pouco de discórdia. Ficava tipo, criando uma richazinha, tipo, deu algum problema ou precisa fazer alguma coisa, e não é no tempo, porque o cara tá fazendo outra coisa começava a reclamar e mandar e-mail falando “Olha conforme conversamos...”.

Desenvolvedor (14:37)

Um colocando problema no outro, direcionado o problema, “Pô, a culpa é dele”, “Não, a culpa é dele”, “Não, não, é dele”... Então assim, nesse momento, nessa fase

era um pouco chato tudo aí. Até que as coisas foram melhorando, principalmente quando essa pessoa que era um pouco mais cri cri acabou saindo do time e tal e as relações foram melhorando.

Desenvolvedor (15:02)

Então assim, a gente precisou mesmo dessa ajuda também do time de infra, que também estava aprendendo com isso. Eles nunca tiveram também um ambiente de negociação eletrônica que precisasse dessas características todas que nosso sistema estava demandando, então também foi o aprendizado nessa parte. Até que a gente conseguiu de fato colocar uma versão de um sistema que atendia vários usuários e a gente foi tirando as maquinhas legado.

Desenvolvedor (15:38)

E nessa época, já tinha demanda, tipo já tinha iniciado, já tinha desenvolvido algumas estratégias de market maker em uma dessas outras aplicações legado. Mas quando a gente foi pro novo sistema, o usuário viu isso como uma chance de escalar o business de Market Maker.

Desenvolvedor (15:55)

Então eles falaram “caramba, agora eu tenho uma aplicação boa pra caramba aqui, tô executando as ordens dos clientes aqui, tá bombando, preciso de *Market maker* aqui também”. E também não só *Market maker*, outros algoritmos como *cash and carry*, os caras viram lá “pô, tá batendo rápido pra caramba aqui, vamos fazer outros *Business*”.

Desenvolvedor (16:15)

Então eles começaram a demandar. Uma hora ou outra algum dos usuários falava, eu preciso disso, preciso daquilo. A gente priorizava, acabava tipo meio que entre nós assim, nunca foi uma priorização nível de “ah, vamos ter uma reunião de governança”, do comitê não sei o quê lá. Não tinha isso não, era aqui, era no esquema do pastel de frango.

Entrevistadora (16:48)

Mas era o time de vocês que decidia? Vocês eram uma unidade?

Desenvolvedor (16:51)

Nós junto com o *Business*. Porque a gente estimava, a gente falava de complexidade.

Entrevistadora (16:56)

Então falavam junto com a mesa?

Desenvolvedor (16:57)

É, e a gente tinha 2 mesas de usuário. Era a mesa da corretora e a mesa proprietária. A mesa proprietária também tinha os algoritmos deles que eles usavam. Só que eles ficaram muito tempo ainda usando os nossos legados, porque eles usavam tipo um outro sistema que era pra arbitragem de ADR.

Desenvolvedor (17:14)

Usavam um outro pra... que que era mesmo? Acho que era a mais pra arbitragem de ADR mesmo. Pessoal usava que era uma outra maquininha legado e essa a gente demorou para trazer pra dentro do sistema, mas por que que demorou?

Desenvolvedor (17:34)

Por que o Business deles foi perdendo um pouco de apelo. Por causa de questões de mercado macroeconômico mesmo, tipo IOF, a operação às vezes não compensava. Então, eles usavam meio que, de um jeito lá que a aplicação legado ainda atendia e tal. Então, quando a gente começou a ter novas demandas de novas estratégias... não sei se eu estou me alongando muito e se eu já respondi.

Entrevistadora (18:01)

Na verdade, eu acho que você até acabou passando um pouco pela 2, né?

Desenvolvedor (18:05)

É que eu falo muito.

Entrevistadora (18:07)

Não tem problema, eu vou só o ler segunda pergunta e aí você complementa, porque eu já vi várias coisas da 2 que você falou, porque a pergunta é: de que forma você descreveria a relação interpessoal da equipe desenvolvimento da corretora com a equipe de mesas de operações. Ou seja, você já tá falando, né, pode continuar.

Desenvolvedor (18:24)

Porque a gente era muito próximo mesmo igual eu falei, então a gente sempre teve essa demanda de ficar no mesmo andar e no mesmo ambiente que eles. Assim, o ambiente em si, desde 2010 mudou muito de lá pra cá, mudaram muitas pessoas também da corretora, os nossos usuários mudaram muito. Não são mais os mesmos, muitos deles saíram do banco e muitos deles eram velhos de mercado.

Desenvolvedor (18:52)

Vamos dizer assim, eles tínhamos cacuetes de quem é velho de mercado e daquela época que eles eram “os caras”, “os traders do viva-voz” e tal. E talvez eles tivessem um comportamento que já não condizia muito com os momentos mais atuais de quando o negócio passou a ser ou começou a ser eletrônico. Então assim, a gente tinha ali várias personalidades, então tipo, tinha uma personalidade do cara que era chato pra caramba tinha impressionante do cara que era gente boa pra caramba.

Desenvolvedor (19:25)

Tinha personalidade que o cara não era só chato ele era escroto, entendeu? Que se achava o banbanban, tinha isso... Então com alguns deles a gente tinha mais facilidade do que com outros. Só que esses outros que já geralmente eram os que mais se achavam, os mais chatos né, eles nem eram os que demandavam da gente, os que demandavam mais da gente eram os que a gente acabava tendo uma melhor relação.

Desenvolvedor (19:52)

Talvez uma coisa estivesse ligado com a outra, por exemplo, se o usuário está me pedindo alguma coisa e ele tá vendo que eu tô entregando é interessante pra ele entendeu? Então assim, ele me pede eu ajudo ele e aí a gente tem uma boa relação e eu consigo ajudar ele melhor, fazendo o que ele quer, fazendo o que ele precisa. Porque ele tá usando a minha ferramenta.

Desenvolvedor (20:15)

E eu tô desenvolvendo a ferramenta pra ele gerar o resultado dele, então era mais ou menos essa percepção que eu tinha das relações que a gente tinha com eles, então tinham aqueles que a gente não tinha tanta relação, apesar de também ser o usuário do nosso sistema, mas que não tinha aquela visão de “eu preciso da ferramenta”, era talvez era um usuário que tinha mais aqueles clientes definidos que sempre executava aquelas mesmas ordens daqueles mesmos clientes, então ele não precisa ficar preocupando com outras melhorias, outros negócios.

Desenvolvedor (20:43)

Mas aqueles que tinham mais interesse em novos tipos de execução de estratégia ou de pessoal, tipo chefe da mesa mesmo, que era mais interessado em marketmaker, ele sim dava mais atenção e demandava mais a gente. Então assim eles eram os nossos hard users, os *heavy users*, ou seja, os caras que usavam mais nosso sistema.

Desenvolvedor (21:10)

Então, era assim, a gente acabava tendo uma relação melhor com os *heavy users*. Enfim, acho que é isso sobre a relação.

Entrevistadora (21:24)

Eu vou passar pra terceira então. Qual a experiência/habilidade dos clientes quanto a utilização e compreensão a respeito do funcionamento do software, da sua modelagem, dos erros já apresentados ao longo do tempo, desde o lançamento em produção.

Desenvolvedor (21:43)



Eu acho que entendi.

Desenvolvedor (21:46)

Experiência e habilidade dos clientes. Bom, talvez isso tenha contribuído muito pro sucesso do início assim, da fase inicial do [nome do software]. Principalmente porque a gente sabia como fazer, a gente tinha pessoas muito capacitadas para fazer, tecnicamente, mas a gente não conhecia do mercado. Só que esses caras conheciam muito do mercado, entendeu? Então ali a parte técnica que era competente junto com os caras que eram competentes no mercado, a gente fazia de acordo com que eles iam mostrando pra gente o que eles precisavam.

Desenvolvedor (22:23)

Lógico que chegava ao ponto às vezes, a gente falava “olha isso que não dá pra fazer”. Até porque, às vezes, a gente chegava a tecnicidades ali da parte de mercado eletrônico que a gente percebia como o mercado funcionava pela especificação técnica [da bolsa], a gente via algumas coisas que a gente chegava pros caras depois falavam, “oh isso não dá pra fazer isso, a bolsa não permite, não dá pra fazer desse jeito”.

Desenvolvedor (22:56)

Aí eles voltavam e falavam “verdade tem razão, então a gente vai fazer de outro jeito”. Então tinha esse tipo de troca também, mas quando eles vinham pedindo alguma coisa, por exemplo, vou citar o exemplo de um dos market maker que a gente fez que foi o primeiro que pessoal, assim não o primeiro eatamente, mas foi um que ficou diferenciado, que foi o de ETF que a gente mais tem volume.

Desenvolvedor (23:20)

Que é o da [nome do ETF]. Nesse cara, assim como ele tinha um volume muito alto e a correlação dele com o índice era muito forte, pro usuário controlar o que ele tava executando e “hediar” [do termo hedge] a posição, ficava meio inviável, porque ele tinha que fazer na mão.

Desenvolvedor (23:54)

Então a gente pensou putz, como que eu faço pra implementar um método automático, pra própria estratégia se manter zerada. E aí a gente pensando juntos, a gente começou a estudar. Você quer se zerar aonde?

Entrevistadora (24:17)

Desculpa interromper, mas quem é o “juntos” ?

Desenvolvedor (24:19)

No caso eu, o [chefe da equipe de IT], os desenvolvedores que tavam entendendo, já já tinham é...

Desenvolvedor (24:23)

Tipo, a gente já tinha absorvido uma boa parte do conhecimento do mercado do que era a solicitação do cliente, já conhecia o que tava falando. E o cliente, no caso o trader, ele tava tentando explicar pra gente o quanto isso deixava ele exposto e qual era a dificuldade dele de fazer isso de uma forma eficiente, sendo que tinha que fazer na mão. Porque os outros algoritmos que a gente tem, por exemplo, de ETF, eles não, não de ETF, mas nessa época esses ETFs, eles tinham um ativo objeto pra se zerar, então era mais fácil, mais simples, um pra um.

Desenvolvedor (25:06)

No caso dessa da [nome do ETF] ele queria se zerar no índice futuro, só que os cálculos de correlação de carteira ou do índice futuro, como era um ativo muito líquido, ele se mexia muito, ele tinha que ficar ajustando aonde ele queria jogar o *spread* dele, se era mais pra compra ou mais pra venda, o dia inteiro na mão. E isso não era viável, porque toda hora o mercado começava a andar unilateralmente e ele ia só tendo prejuízo porque ele ficava sempre na ponta que tava sendo agredida, então ele tinha que se proteger.

Desenvolvedor (25:44)

Então ah, o mercado tá comprando muito, ele tinha que sair um pouco da ponta compradora entendeu. Mas ele tinha que continuar respeitando o *spread* dele. Então a gente propôs eu não lembro exatamente como veio essa ideia. Acho que Robson

falou de um bayas, e eu vim com a ideia de criar uma curva para isso. Por que o que é o bayas? *[Aqui o desenvolvedor explicou detalhes de sua solução, que é confidencial por políticas do banco, portanto foi omitido]*, porque eu já tinha feito isso numa outra estratégia, mas esse bayas era acertado manualmente, então eu pensei...

Desenvolvedor (26:19)

Como é que eu posso fazer esse bayas ser automático? Como que eu vou automatizar então essa *[aqui o desenvolvedor explicou detalhes de sua solução, que é confidencial por políticas do banco, portanto foi omitido]* então, no fim ele [sistema] faria isso automático. E isso melhorou pra caramba, foi uma solução que a gente pensou em conjunto.

Desenvolvedor (27:11)

Não foi nem só do cliente nem só de tecnologia. Mas foi uma ideia muito boa que a gente teve conversando juntos e daí pra frente a estratégia começou, tipo a que mais gerava volume na casa, começou a bombar. Foi comentado no mercado todo.

Desenvolvedor (27:28)

O pessoal ficou comentando dessa estratégia, falando que a gente estava usando inteligência artificial e tudo, só que era um negócio mais simples. Essa parte não sei nem se poderia falar, do funcionamento da estratégia.

Entrevistadora (27:41)

Eu posso cortar, fica tranquilo, você me avisa.

Desenvolvedor (27:43)

Tá bom, acabou que eu expliquei muito. Mas assim, de fato, o pessoal começou a achar esse tipo de coisa.

Entrevistadora (27:55)

Ah, mais uma perguntinha só, sobre a experiência deles com tecnologia mesmo.

Desenvolvedor (28:02)

Então eles não tinham experiências com tecnologia, eles sabiam ser usuários, então a gente também sofria com isso porque o usuário não lê nem mensagem de erro. A gente colocava umas mensagens na tela eles não liam. Chegou ao ponto de falar “ah, mas coloca essa mensagem de outra cor, manda aparecer um barulho”...

Desenvolvedor (28:23)

Então a gente começou, pra você ter ideia, mensagem de erro, a gente começou a mudar a cor da mensagem inteira, da janela da mensagem, pra não ficar no padrão da aplicação, pra ficar inteira amarela. Então a gente colocou uma tela amarela gigante com a mensagem, porque era algo que ele precisava ler. Você tinha que falar pro usuário, a gente tinha muito dessas questões com o usuário, como eles eram muito assim “ah, o negócio tem que funcionar sozinho”.

Desenvolvedor (28:46)

Inclusive, quando eles precisavam tomar alguma ação eles queriam que o negócio funcionasse sozinho, então o usuário, a partir do momento que a gente não tenha essa super sensibilidade de como funciona um software, mas eles são usuários.

Desenvolvedor (29:02)

Eles querem ligar e o negócio fazer tudo. Eles têm esse comportamento assim, de querer ligar e fazer tudo sozinho, então a gente sofria com isso porque eles não liam as mensagens, então a gente começou a ter que ficar olhando o log deles. Eu criei uma aplicação que eu baixava o log da máquina deles pra minha. Só pra poder olhar o log e tal, enfim tinha uns negócios desse tipo.

Desenvolvedor (29:29)

Então, assim, quando a gente começou a explicar pra eles um pouco mais como funcionava e tal aí alguma coisa melhorou, mas não era pra todo mundo. Sempre teve e até hoje tem isso de usuário que ser o apertador de botão. Pra algumas coisas eles só querem apertar o botão, pra outras eles querem entender melhor. Alguma coisa mais complexa, alguma coisa que eles estão pedindo eles vão querer

estar mais por dentro, agora se é só executar uma ordem, quer que funcione e pronto, não quer nem saber.

Entrevistadora (30:00)

Você acabou já mencionando na última resposta por causa de como se construíram juntos a estratégia, né, mas é então assim como você categoriza o grau de experiência desses usuários, né dos clientes no domínio de atuação deles, então no mercado em si.

Desenvolvedor (30:19)

É então como eu comentei, alguns deles eram bem experientes, sabiam muito. Outros nem tanto, vamos dizer assim no início quando a gente começou a desenvolver nos primeiros anos do sistema. Ali tinham só pessoas mais experientes mesmo, depois começará a mudar um pouco, né. Algumas pessoas foram saindo, outras foram vindo às vezes vinham umas pessoas vinham um pouco mais novas e sem tanta experiência no mercado. Então esses também não sabiam muito e aprenderam a ser usuários do sistema.

Entrevistadora (31:00)

Mas quem fazia as demandas, as demandas vinham de quem? Dos mais experientes ou dos mais novos?

Desenvolvedor (31:05)

As demandas sempre vinham mais dos mais experientes.

Entrevistadora (31:08)

Dos mais antigos no mercado?

Desenvolvedor (31:10)

É. Mas aí, à medida que esses caras que eram menos experientes, eles começavam a ganhar experiência, também não eram pessoas incompetentes, tipo eram pessoas inteligentes que estudavam bastante e depois tiraram as certificações. Assim, os

caras eram bons, aqueles que eram bons, eles vinham com boas ideias, eles vinham pedindo as coisas pra gente.

Desenvolvedor (31:31)

Então, com o tempo esses caras viravam os usuários mais, assim, além de mais experientes, também mais colaboradores, colaborativos, entendeu? Então a gente começar a fazer coisas pra eles também.

Entrevistadora (31:44)

E isso ajudava? Tipo tinha alguma correlação entre ele ser experiente e ser mais simples de receber as solicitações ou não?

Desenvolvedor (31:54)

Não tanto porque aquele que não é experiente e também não sabia, ele também não ia te pedir um negócio novo, ele ia usar o que tem entendeu?

Entrevistadora (32:05)

Ah então os menos experientes nem pediam?

Desenvolvedor (32:06)

Não porque, assim eles também tão entrando no no game. Eles também tão no começo, também estão aprendendo o ofício. Os caras que são menos experientes eles eram meio que comandados pelos mais experientes, então eles pegavam as atividades mais básicas que os mais experientes passavam pra eles, tipo executar ordens, que é assim assim assim. Então ele aprendia a usar o sistema pra executar aquelas ordens daquela forma.

Desenvolvedor (32:31)

E aos poucos eles iam começando a perceber o resultado de como ele tava executando a ordem e vez ou outra ele mostrava assim, às vezes tinha alguma coisa que não executou legal. Aí ele perguntava “olha essa estratégia aqui não fez o jeito que eu queria e tal”. E aí a gente tentava entender o que aconteceu.

Desenvolvedor (32:51)

Às vezes, na maioria das vezes era só um parâmetro que ele configurou de um jeito, na verdade, precisava ser de outro jeito para executar do jeito que ele queria. Porque a estratégia já meio que já fazia o que ele queria. Mas algumas vezes de fato aconteceu de a gente pegar putz um erro ou bug. Aí precisou melhorar, então assim os menos experientes eles colaboravam dessa forma.

Desenvolvedor (33:12)

Não pedindo tanto coisas novas, mas às vezes eles percebiam coisas que precisavam ser ajustadas e melhorar, mas os mais experientes é que tinham essas sacadas de fazer umas coisas diferentes, mais novas.

Desenvolvedor (33:30)

Até que os menos experientes começavam a ganhar experiência, tipo, não sei se você lembram do [nome de um trader], ele foi um cara que começou bem novo, bem cru aqui na mesa, ele veio como estagiário. Primeiro de middle, aí foi efetivado no middle, depois ele foi pra mesa, depois o cara virou foi até sales trader, tirou a certificação e tal.

Desenvolvedor (33:52)

Enfim, ele contribuiu bastante, porque sempre pedia novas funcionalidades, novos comportamentos. Ele até notava às vezes, por exemplo, “oh tô fazendo essa estratégia aqui, o mercado tá percebendo que eu tô fazendo assim e eu preciso de alguma coisa pra parar de dar esse sinal pro mercado porque o mercado tá me tomando aqui e não deveria”.

Desenvolvedor (34:15)

Entendeu, então muitas coisas assim a gente fez junto com ele. Então, é um cara, um exemplo, de cara que começou sem muita experiência e depois ficou muito bom, né?

Entrevistadora (34:31)

Sim, e se tornou experiente depois.

Entrevistadora (34:33)

Agora, a gente vai acabar mudando um pouco do contexto para falar da sua percepção antes de mudar você quer complementar com mais alguma coisa?

Desenvolvedor (34:42)

Eu acho que assim do ponto de vista que você tá olhando aqui, né? Pra fatores do cliente.

Desenvolvedor (34:50)

Acho que sim, acho que tá bem relacionado com o sucesso mesmo da implementação do sistema. Essa relação que a gente tinha com eles, essa proximidade foi crucial, se a gente tivesse sei lá em outro andar, acho que a coisa não tinha andado tão rápido e a gente tinha pressa.

Desenvolvedor (35:07)

O mercado é tudo pra ontem. A gente não tinha tantos processos de governança, tanta burocracia que também facilitou a gente colocar muita coisa rápida em produção pra testar e tudo mais, porém a gente assumia riscos que a gente não podia e não deveria assumir, mas é uma coisa que a gente acabou fazendo. Talvez nos dias de hoje, se a gente fosse fazer do zero um sistema desse a gente não ter essa velocidade, com a que a gente teve no início desse projeto justamente por mais governança e tudo mais, mas eu acho que essa essa conexão entre o desenvolvedor e o cliente de fato.

Desenvolvedor (35:53)

Acho que quanto mais próximo e mais alinhado, tipo mais conhecimento é trocado ali, as figurinhas, sabe? Tipo um entende melhor o outro e a solução final sai melhor, entendeu? Se for uma coisa meio que mal conversada vai sair depois no final não atende a expectativa exatamente como o cliente queria como era esperado.

Desenvolvedor (36:17)



Então, também tem essa questão da senioridade do cliente, também dessa experiência que ele tem também é importante como eu comentei. E assim, a disposição mesmo, vamos dizer, o comprometimento do time todo fazer funcionar.

Desenvolvedor (36:40)

A gente estava extremamente empenhado em colocar o negócio para funcionar porque a gente queria que desse certo, porque era um projeto ambicioso mesmo. A gente queria ter a melhor ferramenta de execução no mercado. A gente queria ser o principal market maker do mercado, ter as melhores estratégias de execução e a gente tinha pessoas muito boas. Tanto do lado dos clientes, né os traders, eles tinham inputs muito bons pra gente.

Desenvolvedor (37:12)

Eles traziam essas demandas pra gente. E a gente com a ferramenta que a gente desenvolveu a gente tinha a capacidade de execução muito boa até porque a gente acessa vários mercados com essa mesma ferramenta. Então a gente começou a integrar essas funcionalidades todas pra ele receber ordem de cliente e executar ali.

Desenvolvedor (37:28)

Antes ele usava, sei lá, três ou quatro sistemas pra receber uma ordem de um cliente executar e depois devolver a boleto do cliente. No sistema que a gente fez ele fazia isso com 2 cliques e pronto, entendeu? No final das contas, então, sem essa integração de desenvolvedor com o cliente a gente não tinha tido tanto sucesso não.

Entrevistadora (37:53)

E como que você percebe o resultado disso do processo, como você avalia o processo e depois o resultado do sistema?

Desenvolvedor (38:04)

Como assim?

Entrevistadora (38:06)

Assim, sua satisfação com o resultado desde a efetividade do sistema, né, dele fazer o que ele foi proposto, quanto do tempo, do custo... Eu não sei se você têm noção disso tudo.

Desenvolvedor (38:22)

Assim é um sistema que começou a ser desenvolvido, sei lá, 14 anos atrás ou até mais e eu sou um pouco suspeito pra falar porque eu tô desde o início o fui um dos principais. Desenvolvi, inclusive, a parte da arquitetura da aplicação, toda parte do framework dela como que ela funcionaria pra ser escalável, tem essa lógica de estratégias que conseguem instanciar, novas estratégias e tudo mais...

Desenvolvedor (38:49)

Ou gerenciar os tickets de quantidade, tudo isso. Então assim, a minha sensação, a minha realização tipo pessoal, vendo o negócio funcionar, atendendo tudo que atendeu, funcionando até hoje, bem e o pessoal falando bem, é uma realização pessoal muito boa. É um negócio, sempre eu fui um cara sempre muito tech, que gostava de fazer isso, então não tava fazendo ali só por obrigação. Eu tava fazendo porque eu queria ver o negócio dá certo.

Entrevistadora (39:31)

E em questão de efetividade, tipo de entregar o que foi pedido. Aconteceu de fato?

Desenvolvedor (39:36)

Sim, sim, a gente entregou foi pedido. Acho que a gente foi até além um pouco, porque eu acho que ninguém esperava que ia sair um negócio, pelo menos assim nas primeiras versões, né? Depois o pessoal se acostuma e sempre quer mais, né, mas nas primeiras versões, eles de fato ficaram surpreendidos, que nem eu comentei.

Desenvolvedor (39:56)

Tipo, eles falavam “nossa, caramba tá muito rápido” e aí eles começaram até a pensar “nossa iso aqui pra fazer uma cagada e mandar ordem errada”. Então tinha essa criticidade também, sempre pensava em não gerar prejuízo, imagina uma

estratégia dessa que tava executando milhares de ordens por segundo começar a fazer errado. Num mercado que não era tão maduro como é hoje em termos de travas de risco, tudo mais, entendeu, o risco era grande.

Desenvolvedor (40:30)

É por isso que eu falei que a gente assumia muito risco como desenvolvedor. A gente acabava assumindo os riscos de colocar um negócio em produção que, assim, sem brincadeira, podia quebrar o banco, entendeu? Coisas gigantescas, então olhando pra trás, assim, eu não sei se faria isso de novo, mas naquela época era todo mundo muito... a gente tinha muita vontade de fazer dar certo, entendeu? Então assim e a gente, por mais que a gente assumisse esses riscos.

Desenvolvedor (41:08)

Eu pessoalmente assim, nossa, eu sempre tive muito cuidado com tudo que eu tava fazendo, justamente por saber o tamanho do risco. Então eu checava tudo, tinha certeza de tudo, ainda assim, uma coisinha em outra acabava passando porque a gente não é perfeito, mas sempre a gente tinha ali na mão tipo a “tomada pra puxar”, enfim, a gente conseguia sempre ter certeza de que o negócio estava funcionando de de maneira mais estável, o que não estava tão estável a gente fazia devagarzinho.

Entrevistadora (41:38)

Dá pra dizer que você chegou onde você queria com o projeto e que você atingiu o que você esperava no começo?

Desenvolvedor (41:45)

Para esse sistema acho que sim. Lógico que se eu fosse fazer um outro, conhecendo todas as limitações, sabendo de tudo o que aconteceu, como foi esse projeto, talvez eu fizesse algumas coisas diferentes pra fazer melhor.

Entrevistadora (42:01)

Ah mas pensando com a cabeça daquela época, de você construindo aquilo ali.

Desenvolvedor (42:04)

É pensando com a cabeça daquela época, eu acho que sim, chegamos onde a gente queria. A gente tem um sistema extremamente escalável, que faz o que muita gente paga muito dinheiro no mercado para usar e pra ter. E faz também o que ninguém tem, que são os nossos algoritmos de market maker que são só nossos.

Desenvolvedor (42:25)

Entendeu, então sim. E assim e ele ainda traz a possibilidade de desenvolver coisas novas até hoje, então se alguém vier com uma ideia agora a gente consegue desenvolver rapidamente, com um tempo ok de desenvolvimento. A gente não vai depender de, sei lá, pagar alguém no mercado pra fazer pra gente e não ter exclusividade.

Desenvolvedor (42:52)

Então, acho que sim, pra o que foi proposto, a gente conseguiu entregar.

Entrevistadora (42:58)

Muito bom. Acho que era isso. Gostaria de comentar mais alguma coisa?

Desenvolvedor (43:06)

Eu sou bem orgulhoso desse sistema, então eu já tive a oportunidade de largar o osso já, mas não quis largar porque eu não quis deixar pra trás entendeu. Sempre quis ter um negócio que funcionasse.

Entrevistadora (43:23)

Sim, é bem legal.

Desenvolvedor (43:25)

Talvez eu seja até mais apegado do que eu deveria...