

## Segunda entrevista

- Data da entrevista: 06/02/2023
- Duração da entrevista: 49:22
- Quem entrevistou: Isaque Alves, Leonardo Gomes

**Isaque Alves:** Tá bom, a gente tem um time box (nome do entrevistado) com você de no máximo 60 minutos, então estourando esse tempo a gente já finaliza beleza?

**Entrevistado:** perfeito

**Isaque Alves:** Como a Carla mencionou esse trabalho começou quando a gente identificou que a UPM e a USP com o trabalho do Léo alcançaram um resultado muito parecidos o resultado da UPM talvez você não conheça ela identificou também quatro padrões de times esse interdepartamental DevOps de cada um tá em um departamento diferente, eles têm essa dificuldade na colaboração, eles identificaram que pode ter uma plataforma DevOps que auxilia, mas isso não é comum por outro lado a gente continua ou segundo a gente continua tendo os Dev Ops separados por departamento, só que agora já tem um DevOps atuando mais firme, né para garantir essa comunicação o segundo padrão que eles identificaram que são uma equipe DevOps Cross funcional que eles são como um centro de excelência que auxilia as equipe dev e eles fazem rotação, ou seja, essa primeira equipe tá precisando muito de melhorias e uma parte X então seleciona algumas pessoas para auxiliar. Por fim que a gente acredita que seja o mais, o resultado melhor possível, os times já são times de Dev e Ops, então ele já tem tudo que quer necessário ele já atendem em end to end do produto, só que mesmo assim ainda tem esse centro de Excelência em DevOps que auxiliares em outras partes. O trabalho do Léo provavelmente você já conhece essa imagem, né? Ele também identificou quatro times diferentes, então eu não vou me estender muito nesse daqui, e como a Carla falou o meu trabalho que é um artigo que a gente já submeteu harmonizando todas essas teorias sobre as taxonomias que existem tanto pegando do ponto de vista do mercado quanto da academia. Para construir essa minha teoria, eu usei alguns construtores que eu levo em consideração a cultura automatização, os Silos, a plataforma que ela é muito presente em vários times, a comunicação, a colaboração, o gerenciamento normal e principalmente o time, com isso a gente chega nesse diagrama UML, que a gente tem um sistema de software que é definido por várias atividades e que um ator ou N atores performances atividades desses atores a gente tem os times que possuem estilos que pode ter conflitos pode ter a colaboração entre os times a comunicação e algumas especificações desses times como o product team o time normal que a gente conhece de desenvolvimento de operação e esse time horizontal que aqui a gente vê um pouco mais do DevOps que ele vai guiar ou aplicar a cultura e boas práticas vai auxiliar o time de operação também dá suporte no time de desenvolvimento e também tem duas especificações que é o enabler team e o bridge team. A gente tendo da parte de gerenciamento que, é como uma atividade, é um dos tipos de atividade. O gerenciamento de produto que ele vai estar ligado ao time de produto e o gerenciamento de projeto que vai estar diretamente ligado ao time de desenvolvimento e cada um desses tem uma especificidade ou algo que nos tornam mais únicos para a gente trazer do outro lado a gente tem as atividades mais ligadas à tecnologia. Então eu tenho automação da infraestrutura o automação do ciclo de vida desse meu sistema tudo isso conectado a uma plataforma que eu posso disponibilizar essa plataforma através de um serviço para ele minha equipe de produto. E por fim o nosso objetivo aqui como a Carla mencionou a gente tem hipóteses e a gente tem preposições a gente através dessas hipóteses e preposições nessa conversa com você, a gente quer validar essa nossa teoria colocar ela em teste identificar as fraquezas refinar e evoluir. Então já dando início a nossa série de perguntas, a gente tem algumas perguntas dividido em blocos são no total cinco blocos, a gente vai fazer quantos blocos são necessários de acordo com a nossa conversa aqui. Só que primeiro eu gostaria, se você poderia contar um pouco mais sobre a estrutura dos times do local que você trabalha e qual o papel você desempenha.

**Entrevistado:** Então, eu sou um diretor de engenharia, então a gente é um platform team, do team topologies ai, ai quem usa a nossa plataforma são os times das áreas de produto então, sei lá: Gmail ou workspace Maps, onde tem user generated content pode ter abuso, hate speech, malware, spam. E aí então a gente é especialista em abuso em detectar esse tipo de conteúdo esses times de produto usam a nossa plataforma para classificar. Né? Sei lá quando você aperta lá isso aqui é spam você tá treinando ajudando treinar um modelo aqui que é específico para spam no e-mail, mas que roda dentro dessa plataforma que também atende outras áreas de produto, né? Então a gente é um platform team.

**Isaque Alves:** E qual o tamanho dessa equipe da sua equipe de TI ou da? Que você tem.

**Entrevistado:** Então putz cara é que tem vários times aí, então não sei se você quer saber especificamente a minha parte que é uma parte de server platforms, tem 50 pessoas vai tem um pouquinho mais mas acho que a gente pode falar 50 pessoas, mas tem uma outra parte aí que, deixa eu ver aqui, umas 80 pessoas incluindo aí outras partes que tem uma parte que é no cliente, né? Então nem chega no servidor se é um SMS que você recebe com fishing, a gente não precisa mandar para o servidor de falar cara, não clica nesse link aí direto no seu Android, né? Não tem outros isso tem alguns outros times que trabalham com a parte de manual reaview. Deixa eu pegar aqui só. Vou tentar achar que eu vou validar se são esses 83 aí mesmo. Desculpa cara, então 150 no total essa parte que cuida de tudo aí de counter bios independente né? Então eu tenho um terço dessa organização o meu chefe que é um outro Senior Director tem mais 100 pessoas aí, tá?

**Isaque Alves:** Não, entendi beleza bom.

**Entrevistado:** Esses 80 acho que entra mais outras pessoas que são de ux,

produto, tech writers, né? que não reportam para mim aqui é uma estrutura matricial, então só engenharia de software direta para mim, então tenham um par de produto a gente tem uns program managers, né? Então dentro de counter bios platform tem mais pessoas nessas outras funções 50 são software engineers ou managers?

**Isaque Alves:** Não, perfeito, uma última pergunta mais geral, então é como você vê essa capacidade do time de fazer entrega contínuo, isso é algo forte aí é algo que precisa...

**Entrevistado:** é hoje em dia a gente tem a capacidade de fazer deploy diários, né na plataforma claro que a plataforma por si, só não entrega valor pode entregar novas capacidades, mas aí os clientes ali tem que ir lá e configurar ou começar a usar, um negócio aqui de permitir com que as regras de abusos elas são configuradas em runtime, né? E aí pode ter um analista lá dentro na área de produto ou um programador que vai escrever uma regra nova. Sei lá, vai que tem uma campanha de fishing nova, ele pode ir lá e “putz meu se tiver essa essa palavra chave”, né? O veredito deve ser deletar e a gente quando dá novas capacidades, a gente dá, trabalhando num exemplo agora onde a gente vai poder fazer gradual rollout das regras em si dentro da plataforma, então a gente já faz o nosso continuos deployment com gradual rollout, a pessoa vai poder configurar essa regra aí em runtime e falar não, agora eu quero testar com 1 por cento da população essa nova regra, deu tudo certo vamos para 5, deu tudo certo vamos para 10, vamos para 50, vamos para 100, mas eles têm que começar a usar, né? Então tem essa dependência aí, de que tipo as nossas capacidades vão para o ar não quer necessariamente dizer que elas estão sendo usadas ali na hora.

**Isaque Alves:** Não, beleza, e uma coisa que eu fiquei em dúvida agora, curioso que eu já identifiquei em outras empresas que o time de plataforma ele também consome serviços de outro time de plataforma, isso já aconteceu ou acontece com vocês.

**Entrevistado:** Acontece então, a gente mora dentro de core, né? Então a nossa plataforma de Deploy continua a nossa plataforma de experimentos e feature flags, né ao próprio go ou spam, banco de dados são todos os outros serviços ou produtos aí que a parte de core oferece, não só para minha área, mas para o (empresa do entrevistado) como um todo né? A gente não tá no Cloud do (empresa do entrevistado) ainda porque a plataforma tem sete anos de idade, mas está em um outro exemplo, né tem o time de plataforma de cloud e vários produtos usam cloud, isso acontece bastante sim. A minha é uma plataforma de negócio horizontal, né? Tem essas outras plataformas, mas de infraestrutura de engenharia que a gente constrói em cima.

**Isaque Alves:** E a sua?

**Entrevistado:** Mas tô dando de um banco de dados que é só meu aí é open source, um banco dados de grafo também. Esse não é do Core ali a gente tem uma duas pessoas que cuidam, mas a tendência tentar consolidar essa infraestrutura.

**Isaque Alves:** Não, entendi, bom eu ficaria fazendo várias perguntas, mas eu tenho que seguir os meus blocos aqui e o primeiro bloco é sobre automação. Nos estudos, eu acho que isso já é um algo de

conhecimento geral. É que a gente sabe que a automação ela desempenha um papel fundamental quando a gente fala em DevOps. Então além de possibilitar automatizar a automação de um pipeline, provisionamento, monitoramento ela auxilia na organização das equipes do departamento também. Então para você, quais fatores ele promovem essa automação e quais aspectos são impactados por mais ou por menos quando a gente decide automatizar o ciclo de vida de um software.

**Entrevistado:** Eu acho que é conseguir trabalhar na nossa escala é o primeiro fator, né? Então tudo que automatizado cara, tipo a gente tem ali chegou uma nova revisão de código a partir do momento que alguém fala “beleza tranquilo” tanto rodar todos os testes, quanto fazer esses deploys no canário, fazer tipo é tudo tudo tudo tudo automatizado a gente automatiza processos também que são não necessariamente repetíveis, né? Sei lá como migrando de uma estrutura de dados para outra então deprecando um negócio temos que migrar uma configuração, muita gente automatiza isso também. Inclusive tem grupos lá em core pensando de usar modelos de machine learning para acelerar esse tipo de imigração, a gente teria super interesse, sei lá, um exemplo, a gente tem uma ferramenta de backend-for-frontend onde você pode escrever ali uma descrição de como que você quer o seu formuláriozinho para acessar um dado e aí você não precisa saber o frontend, ele já faz o HTML javascript para você e a gente tá migrando de um estilo de widget para material ui versão 2.0. Hoje em dia os clientes precisam lá e tipo eu tenho isso daqui na minha versão antiga, eu vou migrar para os vídeos da versão nova, né? Então a gente gostaria de automatizar isso para enfim poder entregar esse valor mais rápido. A gente não quer trazer uma nova funcionalidade com o caved disso aí agora você tem que trabalhar seis meses para conseguir concretizar né? Esse valor, então acho que automação tem muito a ver com a velocidade na qual a gente consegue entregar e também a lidar com as coisas na nossa escala, né? Então é enfático para um engenheiro ter que colocar o seu binário em todo o servidores em todos os data centers, isso daí é tudo tudo automatizado.

**Isaque Alves:** Então agora ainda falando sobre automação e com certeza, vocês devem ter métricas ou compartilhar alguma forma de compartilhar informações. Eu queria saber qual, na sua opinião, qual o papel que essas métricas e compartilhar informação ela auxilia ou atrapalha na automação?

**Entrevistado:** Eu acho que você não pode simplesmente confiar cegamente na automação e métricas são ali meio que o pulso que vai determinar, se de fato a coisa tá funcionando ou não, né? Então a gente tem é o feedback desse processo e a gente tem um overside que é humano ainda é o meu deploy e eu vou querer ver quanto da nossa frota tá com a versão nova e eu vou poder ver tipo quando está tendo de erro no stacktrace nos meus logs e eu vou querer poder ver as minhas métricas de negócio também, né? Então sem essas métricas eu tô meio que tipo de maneira muito rápida podendo fazer muita merda. Sei lá eu posso degradar o meu modelo de repente. Ele não tá mais pegando spam nenhum como eu vou saber eu tenho lá automação de uma metrica de procedure recall, que se eu faço alguma coisa de preferência, eu vou pegar isso num ambiente mais controlado ali, né? Não em produção, mas tenho essas métricas e tem alertas, né? Então inclusive quem tá em oncall aí consegue perceber que precisa fazer algo antes dos nossos clientes por causa dessas métricas. Claro que elas não são perfeitas, elas não cobrem tudo mas as coisas mais importantes são cobertas. E aí como a gente é um time de plataforma para gente algumas métricas são muito importantes porque a gente assina contratos com os nossos clientes ali, que são os opções desse service level object, né? Então como o dono da plataforma eu tenho aí um Senior Staff Engineer que fala olha esse daqui é um infrastructure service agreement da minha plataforma você cliente quer entrar saiba que a gente tem umas metricas aí mais sei lá... qualitativas como C7 a gente vai falar quantos nossos clientes estão satisfeitos, né? Se você quiser entrar, mas aí a gente também tem os nossos SLO e vai está lá throughput, latência de resposta, algumas coisas que são métricas aí que a gente não só acompanha a gente se coloca ali como um objetivo, e os nossos clientes sabem que eles podem contar com aquilo, né? Então tipo eu falo eu dou conta de tantos milhões de requests por segundo e a minha latência vai ser não sei quantos minutos de segundos na média, esse é meu contrato com você, né? E aí a gente tem que ter essas métricas sendo monitoradas também de forma automatizada para poder reagir quando alguma coisa, se desvia aí do que a gente espera do que a gente tem de acordo com os nossos clientes.

**Isaque Alves:** Aqui a gente também coloca o compartilhamento de informações junto com as métricas. Para você isso faz sentido?

**Entrevistado:** Olha assim só com essas poucas palavras que você usou para escrever não.

Compartilhamento de informação junto com métrica tipo a quantidade de pessoas que leem um documento compartilhado ali com design um design de sistema os e-mails que a gente troca não, quantas reuniões a gente faz não sei né? Não fez tanto sentido não.

**Isaque Alves:** Tá tudo bem, porque onde eu queria chegar é porque a gente tem uma hipótese que quando a gente automatiza alguma coisa é vamos supor automatizar o pipeline de vida do software. Isso acaba impactando no compartilhamento de conhecimento entre essas equipes eu tenho uma equipe de desenvolvimento que sabe tudo sobre o desenvolvimento e uma equipe de DevOps que sabe tudo da equipe de DevOps quando a gente opta por automatizar esse processo, essas equipes têm que compartilhar conhecimento, isso acaba criando um time DevOps, e acaba evoluindo essa estrutura que eles têm.

**Entrevistado:** Não sei se faz tanto sentido para mim ainda mesmo assim colocar isso como métricas. Acho que é compartilhar o conhecimento e estabelecer relações de confiança. Aí é muito importante, mas a gente tem uma, essa parte da organização, eu não falei né, mas a gente não tem, não chama DevOps aqui, chama SRE né? E a organização de SRE ela reporta para um Senior VP diferente da minha. Então a minha dessa plataforma tá dentro da área de primer (inaldivel) security que tem o VP, que tem o Senior VP lá de core que é a (Nome de uma pessoa inaldivel), a gente tem um time de ABIOS SRE essa aí que são 15 pessoas estão divididas em duas localizações ali na Suíça e Estados Unidos em Nova York e eles apoiam a minha plataforma as plataformas de counter ABIOS, mas ele responde ali pro Senior VP de Cloud. A hierarquia só se encontra ali mesmo no nível do CEO, são separadas, mas aí que a gente tem, tem um ritual mensal que eu tô falando com o Lead de SRE, a gente se comunica através de OKR's, então, eles têm ali objetivos os resultados chave. A gente também é tipo eu sou um dos stakeholders primários que falam legal esses objetivos aí de vocês, dos projetos de automação que vocês estão fazendo de fato são importantes para nós e eles fazem algumas coisas com total independência tem automação engenharia por lá também. Sei lá um exemplo concreto, a gente tá regionalizando armazenamento de dados, né? Para parte do meu aplicativo, isso é muitas vezes uma configuração que é transparente. Como de fato aquele dado vai cair no banco de dados no disco no datacenter que o cara escolheu, é um problema que esse time de SRE está resolvendo, a gente combina uma api, fala beleza? Então eu vou configurar assim, eu vou rotear assado e a partir daí né? Esse é o nosso contrato. Vocês vão fazer essa coisa acontecer como eles vão fazer acontecer, aí é o trabalho deles mas eu sou o stakeholder. Falou cara assim, né? Porque a gente precisa entregar para o negócio, esse é um projeto que a gente tem que colaborar se tem que fazer parte de vocês, a gente tem que fazer a nossa que bom que estamos alinhados aí então é uma série de rituais e ferramentas de colaboração e comunicação. Como os OKR's que a gente usa para estar alinhados?

**Isaque Alves:** Tá perfeito e uma última pergunta, para você essas plataformas, provavelmente você faz parte de uma equipe de plataforma e tem vários outros, quando a gente aplica essas plataformas facilita no processo de automação, vamos supor uma empresa que tá querendo automação então se eu aplico e crio plataformas eu vou acabar organizando e gerenciando o conhecimento das pessoas e isso tem um impacto na automação?

**Entrevistado:** Cara, se não te pergunta, eu acho que esse é o sonho, essa é a promessa, né? Então a gente só cria time de plataformas que a gente vai ter, que a gente aposta que a gente vai ter essa alavanca, mas a realidade não necessariamente chega no primeiro dia, né? Então tem um custo maior você tem que criar uma dependência com um time de plataforma para resolver um problema para você e talvez os primeiros passos que você devam ser um pouco mais devagar do que simplesmente falar não vou resolver tudo aqui dentro de cara, né? Então um dos meus grandes objetivos do negócio é ir atrás de mais produtos e clientes e falar cara, porque que vocês não usam a nossa plataforma porque eles começaram ou eram outras empresas que foram adquiridas, né? Ou tipo não tinham como, era um time muito pequeno ou não sabiam da nossa existência e saíram fazendo E aí a parte de abuso ficou como algo que eles fizeram puxadinho ali tipo custo de migrar da solução deles para nossa plataforma. Inicialmente um custo ali que precisa ser considerado. E aí a promessa é putz. Mas eu sou especialista em abuso. Então você vai ter ganhos de escala de entrar na minha plataforma se eu acho que um novo ataque num cliente. Você pode ter essa informação antes do que se você tivesse isolado aí. Mas é sempre um traidor ser considerada, não é Preto no Branco assim usa plataforma você ganhando um monte de coisa. Se você não tinha nada no começo, né? Fica um pouquinho mais fácil. mas cria essa dependência muitas pessoas mesmo que eu não tô começando do zero muitos times, eles optam por

Não vamos fazer do nosso jeito aqui porque eles querem ter uma velocidade maior que não vai criar dependência e a plataforma sempre é aquilo, né? Eu tô olhando para um pull de vários clientes esses produtos, que eu não vou fazer necessariamente o que você precisa, eu vou fazer o que a maioria dos meus clientes precisam então tem também essa questão tipo necessariamente a plataforma vai me dar todos os corner cases que eu preciso ter e esse é um trade off aí também. Overall no final eu acho que a conta fecha e é bom para o negócio sim investir nesse tipo de time. Mas é Um Desafio, eu tô aqui inclusive tipo estimulado por esse desafio de conseguir vender as vantagens dessa plataforma e conseguir ter mais velocidade e entregar cada vez mais valor para esses clientes.

**Isaque Alves:** Eu acho que o Léo tem algumas outras perguntas sobre outros blocos cross functionality, share responsibility

**Entrevistado:** valeu

**Leonardo Gomes:** Eu tenho assim. Bora lá (nome do entrevistado), então já pegando esse spoiler que acabou de dar em relação a cross functionality, eu vou pegar uma fala que você disse quando a gente estava falando de automação. De estabelecer relações de confiança, né entre os times, né? E no final das contas a gente vai pegar esse aspecto porque no nosso estudo a gente... a gente mostra que as organizações, gerar aspecto de multidisciplinaridade entre a equipe entre as equipes, ou seja, o que significa né, as equipes que ela tem habilidade de controlar todo o ciclo de vida da construção do produto isso impacta diretamente nas estruturas organizacionais. E aí eu gostaria de saber qual é a sua visão em relação a isso se uma equipe multidisciplinar ela tem um impacto direto nessa estrutura que de confiança que você citou e como é organizado no final das contas a estrutura da do time?

**Entrevistado:** Eu não sei se entendi totalmente a pergunta, né? Mas assim Acho que eu posso dizer que é muito importante você ter uma equipe multidisiplinar porque senão você tem um bando de engenheiro de software ali, eles não são times, né? Eles são só um bando de pessoas que sabem fazer a mesma coisa então na hora que eu estou falando com cliente, eu gosto de ter ali um quarteto de liderança que representam algumas das habilidades. Então tem um enginner manager que alguém entende a visão a necessidade do negócio e tá pensando nas melhores organizações aí de times para isso, tem um product manager que tá pensando do ponto de vista do produto e dos jobs to be done e das dores do usuário dos ganhos que ele precisa, né? Aí tem um técnico Líder, um Staff, um Senior Staff Engineer que tá pensando nas questões técnicas nas limitações sistêmicas e tal, e um programmer manager que alguém tá pensando de como eu amarro todas as pontas e garanto que essa execução chega no final.

Então no nosso caso em uma plataforma super técnica e que os clientes são engenheiros e outras áreas essa estrutura que faz sentido tem algumas partes que são interfaces, aí vai ter alguém de UX também, né? Mas na hora de conversar com o nosso cliente tem uma conversa que é PM com PM, né? Produto com produto, tem outra conversa que é tipo engineer manager com engineer manager, business com business, tem uma conversa que é staff enginner com staff enginner, uma coisa mais técnica e tal, então acho que sim é importante a gente considerar essas habilidades complementares como uma estrutura que ajuda você estabelecer essas relações de confiança. Não sei se era por aí que a pergunta tava indo.

**Leonardo Gomes:** Não tá tá perfeito. E aí eu vou puxar isso que você acabou de falar, para falar exatamente de equipes multifuncionais. Então a equipes que por exemplo tem a parte de UX, a equipe que tem por exemplo o Senior de desenvolvimento, de engenharia, né? E aí a gente tem uma hipótese que a gente caracteriza essas equipes multifuncionais com alto e Médio nível de compartilhamento de habilidades. Então dentro dessas equipes as pessoas elas conseguem fazer esse compartilhamento de conhecimento, eu queria saber sua opinião em relação a isso se você acredita que no final das contas dessas equipes e acaba que dentro dessas equipes, eles conseguem ter um compartilhamento alto, de médio a alto compartilhamento de conhecimento.

**Entrevistado:** Cara acreditava eu acredito, mas é tipo aquele unicórnio, né? Você vai ver uma vez na vida equipe de fato consegue ter 100% de cobertura e entre papéis, né? Então eu diria que a gente tá no nível médio de compartilhamento de habilidades aí porque são né, trabalhos diferentes mesmo, né? Então eu gostaria de chegar em um nível mais alto, mas aí eu tenho que prestar atenção. Quais são as habilidades que eu tenho ali na minha equipe e com tratar para isso então um exemplo, né? A gente tem poucas pessoas de produto. Porque o produto em si é uma plataforma super técnica, né? É uma api que

se tá vendendo por outro desenvolvedor de software ali de outra área e não tem como ter cobertura como tudo então tipo na parte de storage, né? A gente tem alguns complicated subsystems que tem todo o time que que toma conta daquilo, mas ali eu tenho um Manager um Tech Lead, alguém um pouco mais senior e né? e engenheiros, então é um time muito focado em geral engenharia de software tecnologia, não tem ninguém muito de produto ali É nesse sentido que que eu tô fazendo, eu tô tentando forçar esse médio para alto contratando novos managers que não só manjem da parte tecnologia, tem esse track de ter desenvolvido software, mas também tem nome de produto. Venham de alguma área mais né? Startups mesmo, tiveram que cobrir essa parte, eu mesmo na minha carreira me desenvolvi muito em produto para poder dar essa cobertura. Mas assim, se eu for te falar quantas vezes eu vi um time onde essa cobertura era alta e uma mesma pessoa podia fazer produto, management, engenharia de back, front, sistemas distribuídos é raríssimo.

**Leonardo Gomes:** Realmente. E aí a gente parte para uma visão mesmo nessa visão rara da pessoa, né? Que realmente, a gente vem para uma outra hipótese que é nessa perspectiva de equipes multidisciplinares, né? A gente pensa, a gente levantou que em uma hipótese, que equipes dessa estrutura acaba no final das contas automatizando o ciclo de vida de software, você corroboraria com essa afirmação?

**Entrevistado:** Em questão de equipes multidisciplinares que automatizam? **Leonardo Gomes:** Eles tendem, elas tendem automatizar o ciclo de software.

**Entrevistado:** Putz, Eu acho que posso concordar mas eu não vejo correlação, só porque uma equipe não é multidisciplinar não viria porque elas não automatizam também.

**Leonardo Gomes:** Só tava fazendo algumas anotações que você estava dizendo. Mas então, bora lá, a gente vai para um outro bloco agora agora partindo para questão de compartilhamento de responsabilidade. A gente pega dentro da nossa pesquisa também que a compartilhamento de responsabilidade entre equipes e indivíduos ele afeta diretamente a estrutura da equipe, ou seja, que que isso quer dizer, né? Então pessoas que é que compartilham de um mesmo produto por parte da organização, então essa gestão de responsabilidade compartilhada dentro do produto, no final das contas ele cria indivíduos que afeta diretamente a estrutura da equipe. Aí a gente pega nessa parte, né? Qual é a sua opinião em relação a promover o compartilhamento dentro dos indivíduos, né dentro das equipes, se isso no final das contas afeta a estrutura afeta a equipe no geral se afeta o desenvolvimento e afeta no final das contas voltando para essa parte de DevOps, né? Afeta o ciclo de vida.

**Entrevistado:** Sim, muito 100% acho que uma das coisas que mais afeta, né? Então a parti do momento que você tem objetivos compartilhados e cada um tem que fazer a sua parte, você tem um time mesmo e você consegue ter impacto maior, conquistar objetivos maiores, né? Que que a gente faz que é maior do que cada um de nós nenhum. Ninguém pode falar que tipo. Ah, eu eu fiz a minha parte eu entreguei né? E todos no final tem que colaborar para um negócio sair, né? Então é um Shift para você ter um time versus você ter um grupo de indivíduos cada um fazendo a sua partezinha ali, eu acho que é muito importante, inclusive é um trabalho contínuo de gestão você tentar amarrar essas coisas, né? Para que uma pessoa não acha que tipo só a minha parte tudo bem? Tipo até para conseguir entregar um impacto um pouco maior onde não seja um mais um seja igual a três ali em vez de dois na hora que você tá trabalhando junto, né? Então eu tenho a tentar fazer isso através do processo de setar os objetivos e metas ali os nossos OKR's, e aí quando a gente tá falando expectativas de cada indivíduo um exemplo, né? Um tech lead vai ter expectativa de olha então para você impactar, ter esse impacto no negócio esse (inaldivel) que a gente quer trazer você vai liderar esse projeto vai conseguir fatiar e fazer mentoring desses seus quatro, cinco engenheiros aí junto com seu manager, aí o seu manager vai ter uma expectativa mais organizacional, você vai ter uma expectativa de roadmap técnico, de visão tecnológica e cada um dos engenheiros vai ter uma expectativa de executar o seu pedaço, mas todos alinhados com esse objetivo de negócio maior que a gente faz, né? Então esse é um trabalho contínuo, e que traz, traz um diferencial porque senão você tem ali projetinhos que são muito contidos e um impacto muito menor né? Tipo tem um engenheiro uma engenheira que tá fazendo um negocinho menor ali, você acaba ficando preso o as óticas locais, versus quando você tem essas expectativas compartilhadas, você consegue achar uma máxima global.

**Isaque Alves:** E quando a gente expande um pouco mais esse compartilhamento de responsabilidade,

não sei duas equipes tem que somar esforços para resolver um problema entre elas. E a gente passa a compartilhar mais essa responsabilidade, isso é capaz de quebrar os silos culturais ou organizacionais.

**Entrevistado:** Depende do tamanho cara, a partir do momento, acho que você vai ter demanagins returns (inaldivel) aí, né? Então se você tem sei lá se essas duas equipes estão em times zones muito diferentes aí, é tipo, se tiver mais do que 6 horas de diferença, aí você vai criar um overhead de comunicação, é melhor separar e dar mais autonomia, vocês têm objetivos, vocês têm objetivos (gesticulação de duas coisas separadas), do que tem que ter essa dependência, né? Se já tá mais próxima num time zone até dá para fazer alguma coisa um pouquinho melhor, eu acho que o problema tem a ver com também, claro como uma organização a gente quer tá ligado aos objetivos da companhia, né? Então tem alguma coisa que amarra tudo isso, mas a partir do momento que você tem muitas dependências aí e vai ter um objetivo só, fica difícil, né? Você vai criar gargalos de comunicação e você vai tirar autonomia do pessoal poder rodar mais rápido dentro do seu contexto, “a (onomatopeia de surpresa) que que a galera lá vai achar, né?” Acho que depende um pouco do tamanho dessa organização que você tá falando.

**Isaque Alves:** Entendi e quando a gente pega essas equipes de plataforma. Elas têm as suas próprias responsabilidades. Elas têm um limite ali que é quando ela fornece um serviço a gente pode dizer que essa equipe de plataforma acaba sendo um silo também.

**Entrevistado:** Acho que sim, né, porque eles cuidam da plataforma e se ninguém usar plataforma eles entregaram zero valor, né podem estar ali numa Torre de Marfim pirando em utilizar um negócio que tá desconectado com as necessidades dos clientes e os clientes não usam, se otimizou algo que ninguém tá usando e né você não entrega o valor. Acho que tem esse risco sim. E por isso que é importante ter aí uma proximidade e esses stakeholders management e essa visão de produto mesmo para garantir que você está investindo em algo que vai ser valioso para a maioria dos seus clientes.

**Isaque Alves:** Entendi, obrigado, desculpa te interromper Léo.

**Leonardo Gomes:** Não, mas perfeito os seus apontamentos Isaque, obrigado. E aí a gente continua ainda falando essa parte de estabilidade, né? E aí a gente pega a questão cultural, né? A gente vê bastante empresas criando e compar tilhando culturas, culturas entre as equipes né então com outras positivas, às vezes até de, é de reuniões que precisam compartilhar conhecimento entre outras coisas. E aí a gente pega nessa parte de compartilhamento de responsabilidade. Você acredita que empresas que desenvolvem essas culturas, né? Desenvolve uma cultura mais plural. Elas acabam também fazendo esse compartilhamento de responsabilidade ser mais frequente dentro das equipes.

**Entrevistado:** Sim sim, acredito, acho que são coisas que vão juntas, né? Então a partir do momento que você valoriza muito a colaboração, você vai fazer com que as equipes conversem com seus vizinhos e queiram compartilhar a responsabilidade por algo maior.

**Leonardo Gomes:** Perfeito e voltando um pouquinho nessa parte do que vocês falaram por silos, né? A gente tem uma, a gente tem uma coisinha muito impor tante ali no meio que é no final das contas aí enfim. Você acredita que o desen volvedor né que trabalha dentro dessa parte do, trabalhando com tecnologias que oferecem essa parte de infraestrutura as a code estão fazendo um pouquinho esse gerenciamento automatizado. ele pode executar no desenvolvimento essa tarefa de desenvolvimento e aprimorar o compartilhamento da responsabilidade quebrando um pouquinho essa perspectiva dos silos dentro daquele estilo de infraestrutura?

**Entrevistado:** Então, é que você cria uma camada de abstração, eu não sei dizer se isso é um silo no sentido de tipo, nem tem o acesso, você tem que enten der um pouquinho de um nível abaixo, né? Então quando eu estou configurando aqui o meu serviço eu falo que eu preciso de não sei quantas instâncias, qual que é o tipo de banco de dados qual que é meu data retention, data replication plan, quanto de redundância eu preciso, eu tenho que ter um mínimo de entendi mento do que isso significa em como infraestrutura vai ser de fato provisionada, mas eu não preciso entender de configuração, de nuvem, de máquina virtual, de rede, de alocação, de dados, nem nada; isso daí é responsabilidade do time que me prover essa capacidade; então acho que você mantém aí o silo nesse sentido, mas para eu fazer um bom trabalho como desenvolvedor, se eu não entendo o que é aquela configuração significa é meio que tentativa e erro ali, né? Eu pus o negócio lá funcionou, beleza, ele vai dar um problema, mas e aí, você tem backup? “Ah não, configurei essa parte porque não parei para pensar que o banco da dados

precisava de backup” Então tem um risco que você introduz porque né, agora a interface é essa configuração, mas hoje em dia também com complexidade das nuvens e tal, é algo que não tem como você cobrar de uma pessoa se especialista em todas essas partes de uma estaca tão complexa, né?

**Leonardo Gomes:** Então, você acredita que? Desculpa, pode falar.

**Entrevistado:** Acho que a barra mínima você conseguiu entender e conseguir escrever pelo menos em alto nível, o que que esse outro nível né? O que que está abaixo dessa camada de abstração de fato está acontecendo.

**Leonardo Gomes:** Perfeito perfeito, Isaque você quer comentar alguma coisa, porque agora eu vou passar para o próximo bloco.

**Isaque Alves:** Então tá tudo certo, pode ir acho que dessas perguntas e dos

comentários do Alez a gente já possui por várias dessas hipóteses que a gente tem.

**Leonardo Gomes:** Eu acredito que sim eu iria no final das contas, só nem que seja um pouquinho redundante, mas só para Ver a questão do último bloco que você já passou por essa nas suas falas você já passou um pouquinho por esse último bloco. Que é o bloco de colaboração, então é você, acredita que no final das contas colaboração entre imagina que a gente vou dar um exemplo. Imaginário aqui a gente tem um design System a gente tem várias plataformas no final das contas que colaboram para aquele desenvolvimento, né daqueles anicista sendo que eles trabalham em times diferentes trabalham em plataforma diferente, você acredita que a colaboração agora na perspectiva de DevOps ela seja crucial para para o desenvolvimento evolução e para e que seja frequente dentro das equipes.

**Entrevistado:** Sim sim, acredito muito acho que é aquilo, né? Se você tem um design system e você não tá com colaborando com os designers e com os engenheiros de front-end para entender, o que que eles precisam para você determinar quais são os próximos passos, capacidades que você precisa colocar nesse design system? Você tá perdendo uma oportunidade aí de entregar mais valor, então acho que na sua essência essa colaboração é que permite com que a gente tem os ganhos de escala de criar esse silos de maneira deliberada, quando a gente cria esses times de plataforma, a gente tá criando uma dependência, o essencial seria você poder ter um time ali, só com né, os unicórnios que conseguem fazer tudo de ponta a ponta, mas isso não funciona em escala, pelo menos talvez funcione ou eu nunca achei, né? Se vocês acharem um dia me mandei um exemplo a par tir do momento que você está numa certa escala, você vai começar a criar essas dependências; e aí? O ruim de você criar uma dependência é? Se essas pessoas não se falarem, né? E aí você mitiga esse risco criando laços de colaboração aí entre elas garantindo que existe a confiança e muito alinhamento aí para você não pegar só o ônus de ter introduzido essa dependência.

**Isaque Alves:** O (nome do entrevistado) e quando essas pessoas, esses times eles têm papéis muito bem definidos, a gente tem o desenvolvedor, a gente tem o PO; Você acha que isso atrapalha na colaboração entre essas pessoas? No mesmo time?

**Entrevistado:** Essa pergunta tá todo um outro doutorado, né Isaac? Acho que sim. Acho que atrapalha, agora ao mesmo tempo as pessoas querem muito saber qual que é a caixinha delas, então muitas vezes você precisa deixar explícito, desse papel a gente espera essas responsabilidades e a partir daí qual que é o risco, eu só vou fazer o que tá no meu quadrado. Né? “Ah não, não isso daqui não é comigo isso daí é com a pessoa de programa ou com pessoa de produto, eu sou só um engenheiro”, né “lavo as mãos do que não tá ali no meu papel de responsabilidade”. Então por mais que eu acho importante, você dá clareza do que que se espera e qual a diferença entre cada papel, eu acho que isso daí é só meio que o começo e o guideline tem que estar atrelado com, a gente espera que você possa trabalhar e ter influência fora da sua caixa cobrir pelos seus colegas, então várias vezes o conflito que eu tô querendo resolver é uma pessoa que falou, “não isso daqui não é meu trabalho, é da pessoa de produto ali” eu chego e putz, mas não tá sem pessoa de produto agora porque o mundo não é perfeito e a gente não tem os recursos infinitos, e agora você e fulano não podem cobrir essa parte ou não tem como, tipo entendo que você não vai trazer a mesma qualidade do que alguém que se especializou nisso daí né? Mas você topa aprender e investir nisso daí, na sua carreira para poder tipo ir um pouquinho fora do seu caderno que é porque é isso né, os nossos times nunca vão ser perfeitos, né mesmo no (empresa do



entrevistado) onde a gente pode contratar, ou enfim podia, tantas pessoas quanto quanto necessárias, você nunca vai ter ali o time que tá ali redondinho, né? Então cara, não sei, acho que é um jeito longo de dizer que eu acho que é importante ter isso, mas acho que também tem uma pegadinha muito grande aí, no sentido que se você faz isso e as pessoas se limitam; o framework ele nunca é 100% exaustivo, né? Então quando você lista as responsabilidades do papel, você nunca vai conseguir ter ali o território, você vai ter um guia, né? Um mapa só e aí as coisas vão começar a cair entre as festas; então o que eu gosto é de tipo deixar explícito para o pessoal, olha; beleza, então você como engenheiro você como um produto tem essas responsabilidades primárias, mas eu eu preciso que exista aí um gran diagram com um overlap e que vocês possam também; eu quero alguém de produto opinando no design técnico, eu quero alguém técnico opinando nas histórias e nas especificações de produto porque você vai trazer uma perspectiva diferente, não é um manda aqui, outro manda ali porque senão você mata a colaboração que é o que a gente falou ali que também é essa liga, né? Se eu tivesse Silos entre as funções, aí vai só dar briga, E aí você vai fazer o que vai acabar escalando, e aí vai ter um VP de produto brigando com um VP de engenharia com outro de operações; tipo, não, se chegou até lá em cima, a gente já perdeu, né?

**Isaque Alves:** Eu ouvi em outras entrevistas que o ideal para a gente resolver isso, é que todo mundo se sinta responsável pelo produto então; ah, tá dando algum erro em tal parte; ah, mas quem é responsável por isso é a outra pessoa, só que essa outra pessoa não tá aqui então eu posso ajudar eu vou fazer alguma coisa porque eu me sinto responsável por aquilo; você vê isso como uma forma de resolver isso então a gente, é uma linha tênue entre a colaboração e se sentir responsável pelo time como um todo.

**Entrevistado:** Sim, acho que você tem que desenhar essa responsabilidade compartilhada, né? Então quando você tá citando as expectativas por mais quem vai estar editando a visão do produto aquela pessoa, você também é responsável por influenciar; olha tem um tech lead aqui o staff enginner, ele quer ta ditando rubymap técnico, você como produto, você como engenheiro de software, vocês também são responsáveis por influenciar, perguntar, participar, eu quero seu bainha ali, não é? Ele falou e ai se der ruim o buraco tá do lado dele do barco, né? Não é bem por aí.

**Isaque Alves:** Entendi bom a gente já tá se aproximando falta 5 minutos, eu acho que dá tempo de fazer só mais uma pergunta. Você tem Léo? Alguma pergunta que você queira. Muito fazer

**Leonardo Gomes:** Pode mandar bala Isaque

**Isaque Alves:** mim é isso. Eu já conversei bastante. Eu queria muito agradecer (nome do entrevistado) ou (apelido do entrevistado), eu não sei como você prefere que te chame.

**Entrevistado:** Coloquei porque os gringos não conseguem pronunciar (nome do entrevistado), eu tentei (nome do entrevistado) na minha primeira semana e eu fiquei por último numa reunião que estavam chamando as pessoas vão chamar o nome do outro; eu falei “Putz não vai dar certo”, deixa eu colocar (apelido do entrevistado) ali

**Isaque Alves:** Não então perfeito, eu queria te agradecer por participar, assim que a gente tiver a transcrição eu te mando para você dar um cheque para gente, se a gente vai ver e esqueceu ou citou o seu nome ou citou a sua empresa, mas a gente não pretende levar isso, eu acho que nem para os outros para a equipe da Universidade de Madrid.

**Entrevistado:** Perfeito, eu agradeço aí, acho que tipo condição não deu poder participar era essa a normalização, né? Então apagar ali qualquer coisa (empresa do entrevistado) e tal, e parabéns aí pessoal pela pesquisa, acho que vai ser bem legal aí poder entender melhor dessas taxonomias, né? Até influencia muito como eu faço meu trabalho como gestor, né? Porque eu vou lá identifico alguns padrões; nossa vocês conseguiram sintetizar aí essa parada aí, eu posso usar essa mesma linguagem como grupo de managers ali e acelerar o processo de designer organizacional então. Veja o valor na pesquisa que vocês estão fazendo. Obrigado por embarcar nela.

**Isaque Alves:** Valeu por enquanto a gente não vai ter esses resultados, a gente está numa fase de operacionalizar, testar; então acho que é o fim do meu doutorado, acho que no ano que vem, eu consigo ter esse essa teoria um pouco mais concreta, eu posso compartilhar por e-mail outras formas de contato, tá bom?

**Entrevistado:** Tem sempre essa pesquisa aí, mas só o seu ml ali já “Opa. Nossa que legal.”

**Isaque Alves:** Aquilo foi mais de anos um ano, eu acho fazendo Mas então beleza? Muito obrigado, viu?

**Entrevistado:** Obrigado pessoal o resto da pesquisa

**Leonardo Gomes:** Obrigado (apelido do entrevistado)

**Entrevistado:** qualquer coisa estiverem em um flow up. dúvida, estou aqui à disposição.

**Isaque Alves:** Tá bom. Obrigado. Tchau. Tchau.