

Transcrição D2

Função: Programação Web/ Análise de Sistemas

Tempo de experiência : 7 meses

## Seção 1\_

Daniel: Você pode descrever para a gente como é o observatório e o sistema de bolsas?

D2: Eu posso descrever como é o observatório. Bolsas eu acho que eu não sou a pessoa mais indicada para isso. Por que eu só usei o sistema de bolsa como usuário. Só. O observatório funciona com gráficos pré-estabelecidos pela S1, que é a coordenadora do programa de bolsas. Ela é estatística, então ela diz assim: “a gente vai apresentar os dados dessa forma”, a gente vai usar tal gráfico, barras, boxplot... e a arquitetura foi toda F1... tudo que foi decidido como usar, as anotações do Java 8, tudo foi ele que fez. E basicamente ele extrai dados do SIGA, que é o nosso sistema atual de gestão, e é separado em vários tópicos no Observatório. O que foi escolhido primeiro foi o de bolsa, porque é o que estava melhor estruturado. Pelo que D1 fala, essa parte de bolsa estava bem estruturada porque tinha um cara de informática que estava alocado dentro do PROGBOL. Então ele fez do jeitinho que era para fazer. Ele conseguiu estruturar bem, modelar bem essa parte de bolsas, dentro do SIGA, de acordo exatamente, refletindo a realidade do programa de bolsas. Daí, como era a parte melhor estruturada, e foi a que se escolheu primeiro para o Observatório ir pro ar. Aí agora tem projetos, editais, e ainda tem um quarto tópico a ser abordado no Observatório. Mas todos vão seguir aquela mesma linha. De ter em primeiro lugar um gráfico dinâmico com informações gerais, e um dashboard com vários gráficos, que você pode consultar. Esses gráficos já são pré-estabelecidos... tipo, por sexo, por idade ...

Daniel: O que é esse SIGA, que o pessoal fala... como ele funciona?

D2: Olha, a sigla é sistema integrado de gestão. A FAPESB tem um processo. Começa com os editais... tens alguns acordos assim e tal... mas começa com o edital. Com exceção de alguns acordos que geram projetos sem que passem pelo edital. Então você tem que lançar um edital, você tem que gerar um formulário para o pesquisador, submeter propostas de projetos. Eu acho melhor falar assim: a FAPESB está aqui para quê? Para fomentar a pesquisa. É Fundação de Amparo à Pesquisa do Estado da Bahia. Ela está aqui para dar esse suporte. Então começa com o quê? Ah, eu quero submeter um projeto, a primeira coisa que eu preciso esperar é o quê? O lançamento de um edital. Daí eu vejo se a minha linha de pesquisa está contemplada naquele edital. Então a gente precisa disponibilizar formulários para as pessoas poderem cadastrar seus projetos. Aí tem a avaliação que se faz e todo um processo até chegar no pagamento. O SIGA gere tudo isso. O SIGA é um monstro.(risos).Porque hoje ele não atende mais como deveria atender o processo da FAPESB. Aí a gente está reformulando e vai existir a plataforma que vai substituir aos poucos o SIGA.

Daniel: Como acontece aqui com vocês aí o processo de desenvolvimento de software, desde vocês saberem requisitos, a parte de programar, testar... como é feito isso, assim, no geral?

D2: Essa equipe que a gente tem é uma equipe relativamente nova, e pelo que eu tenho conhecimento, (trecho ininteligível), o processo que a gente está utilizando aqui não é o mesmo que se utilizava há dois anos atrás... sei lá... outras pessoas. Aqui sempre se usou o Web Integrator, que é um W.I. ... o SIGA é construído em cima do W.I. Agora a gente está validando um processo diferente, (trecho ininteligível), a gente está pegando do começo ao fim um processo. E mais tudo que a gente está colocando para desenvolver a plataforma Roberto Santos são tecnologias que já foram usadas. Por exemplo no Observatório foi feito por uma pessoa só. Quando eu cheguei o Observatório ‘tava praticamente pronto. Depois, eu assumi depois que F1 saiu, mas uma equipe mesmo em torno ali de um sistema maior, este é o primeiro que a gente está tendo e validando esse processo. Então, no Observatório o F1 já usou tecnologias mais novas, o Java 8, o front-end usou lá o bootstrap; isso não era usado antes. No sistemas de bolsas fizeram com Road(?), fizeram com Angular... tem Angular também no Observatório... então

são sistemas que vieram antes que vieram abrindo portas para que hoje a gente tenha uma equipe saiba lidar com isso... mas agora a gente também está introduzindo coisas novas e métodos novos. Aí agora com o Observatório já começou na primeira reunião de análise de requisitos a gente montou uma sala de guerra sobre a metodologia lá da Inception. Depois vocês podem ver melhor como funciona. Que é um processo ágil de levantar requisitos. A gente saiu das reuniões com um Produto Mínimo Viável nas mãos.

Alex: Isso pro Observatório?

D2: Não. Isso para plataforma Roberto Santos. É por isso que eu estou falando. A gente está validando esse processo agora. Inclusive depois eu quero botar tudo num papel e apresentar lá na SAMUD(?), lá na sede. Porque tem sido um processo, assim, muito ágil, que tem sido uma mão na roda: a gente com uma equipe enxuta, carente de recurso humano, e a gente está conseguindo levar e já está bem adiantado. O primeiro protótipo em menos de dois meses aprovado. O sistema já está começando a ser desenvolvido, e até dezembro é a previsão que a gente tem de entregar o MVP. Então é um processo bem enxuto, enxuto mesmo. Enxuto até de documentação e tal... A gente está usando um framework que é uma mão na roda, que é o JHIPSTER... Na verdade ele é um gerador de código, parecido com o TJFW e com o... (divagação sobre o nome da ferramenta). É parecido, ele gera os CRUDs e é muito “massa”. Depois vocês dão uma brincadinha com ele. Então, nosso processo basicamente sempre procura usar metodologias ágeis e ferramentas que facilitem todo processo e eliminar toda a burocracia envolvida. Tudo que a gente conseguir pelo menos. Uma coisa muito legal que a gente está utilizando também é o TDD porque a gente não tem equipe de teste. A gente não tem alguém alocado só para testar. Confesso que é uma mudança de cultura bem dolorosa sentar para escrever o teste antes de escrever o... mas o benefício que você tem depois é muito grande.

(Entrevistada pede explicações sobre o procedimento da entrevista. Entrevistadores explicam)

D2: Assim, até então, a gente tem um processo consolidado? Não! A gente está com uma equipe nova validando a melhor forma de seguir. Mas acredito que depois, do Roberto Santos... eu vou botar tudo no papel

Daniel: Mas no Observatório, você já pegou pronto, né? Você não chegou a participar de reunião?

D2: Só para a parte de colocar em produção, validar todos os dados, e acrescentar duas melhorias que foram pedidas.

Daniel: E aí já estava usando essa metodologia ou foi...

D2: Da inception? Não usou. Era o F1 e o D1 que iam reuniões, até onde eu tenho conhecimento e... não usou não.

Daniel: Aí vocês faziam por exemplo, para as funcionalidades do Observatório: vocês tinham reuniões...

D2: Sempre havia reunião com [um stakeholder] e S1. Eram as duas pessoas mais envolvidas. [Um stakeholder] era da assessoria de planejamento e S1, do programa de bolsas: A estatística. Elas que validavam todos os dados: o que está certo e o que não está certo, tal gráfico não deveria ser esse de cá.

Daniel: Pra você testar funcionalidade, validar, validava com ela, mas você não tinha esse TDD...

D2: TDD é a metodologia de teste para você escrever o teste antes de codificar. Não. Isso aí não. Eu não sei nem se caberia no Observatório. O Observatório não tem CRUD, você só visualiza dados. Mas para as consultas tem um tipo de retorno... sim. Valeria. (Entrevistada pensa em voz alta consigo). Mas o Roberto Santos é o primeiro sistema que a gente vai usar o TDD.

**\_Fim da Seção 1**

**Seção 2\_**

### **Princípio Users Before Algorithms**

Daniel: A gente está indo agora para uma seção que pergunta mais sobre descoberta de dados e como esses dados são apresentados pro usuário.

Daniel: Então a pergunta é a seguinte: Quando vocês colocam algum filtro de busca, algum algoritmo, vocês já pensam como o usuário vai visualizar aquilo. Tipo, vou implementar um algoritmo. Aí antes de eu implementar eu já penso assim: Tá, o usuário vai ter esse tipo de impacto.

D2: Não entendi em que momento eu pensaria nisso.

Daniel: Por exemplo, vocês têm uma reunião. Aí vocês falam: “A gente achou um método de busca legal”.

D2: Um filtro a mais.

D2: Geralmente isso parte do colaborador daqui da FAPESB. Ele diz: olha, seria interessante que também se pudesse visualizar por tal coisa. Daí, sendo uma requisição dele, o filtro que passa aqui na minha mente é se faz sentido aquilo. Na maioria das vezes faz. E depois senta com o IHC1, que é o designer, ele tem domínio dessa parte de interface, o mestrado dele é nessa área, e vê assim qual a melhor forma de a gente apresentar isso daqui. Eu nunca sei dizer qual é a

melhor forma de apresentar isso aqui. O que eu sei dizer é isso aqui, segundo a minha ótica, não sou especialista nisso. Daí ele fala assim: A gente pode fazer assim, pode fazer assado, a gente dá uns pitacozinhos, e pronto! Eu meto a mão e faço.

Daniel: Quando vocês pensam num filtro lá ou algum algoritmo, depois que você implementar, vai passar pelo designer?

D2: Não. Você tem uma ideia inicial em se fazer. Digamos assim, 90% das vezes parte do cliente, dizer assim: vamos botar esse filtro aqui. (Trecho ininteligível). A gente está falando de acrescentar uma coisa que já está pronta. E aí primeiro vai pro designer. É ele que diz primeiro como fica melhor. Isso aqui na FAPESB, nessa equipe que a gente tem aí. Ele que diz como vai ficar melhor. Se o botão vai ser em cima ou em baixo. Na esquerda ou na direita. Aí eu chego e falo para você: Não, olha, eu preferiria aqui na esquerda, porque eu estou acostumado lá de outros sistemas, para mim sempre foi ali. Não, mas aí tem lá toda uma fundamentação que manda botar na direita. (Entrevistada divaga sobre sua dificuldade em discernir entre esquerda e direita). Se botar aqui do lado de cá, tem uma fundamentação teórica aí de que é melhor para pessoa ver. É melhor um ícone ou é melhor um labelzinho dizendo assim: "excluir"; ou é melhor a lixeirinha? Eu prefiro a lixeirinha. (Entrevistadora circula nos mesmos argumentos).

Alex: Então o fluxo das funcionalidades, requisitos, qualquer coisa do tipo é basicamente cliente, aí sai do cliente e vai para a reunião, vai pro design, e depois vai para a programação de fato.

D2: Aí volta com o cliente. Se tiver mais alguma coisa volta para mim.

Daniel: Acontece alguma coisa, tipo assim, vocês terem que reimplementar um filtro de busca? Por exemplo, vocês já têm um filtro lá. Aí vocês acharam um jeito melhor, mais rápido...

D2: Vocês estão falando bastante de busca, né? Porque para vocês é interessante né?

Alex: É

Daniel: Também.

(Entrevistada confirma pergunta)

Daniel: Por exemplo, vocês... às vezes nem necessidade não... mas às vezes vocês terem um filtro de busca que vocês utilizam um método, algum algoritmo, e aí vocês terem que ter mudado ele com ele já implementado. Já aconteceu isso algumas vezes?

D2: Mas você fala da interface ou o algoritmo que roda por trás?

Daniel: O algoritmo que roda por trás.

D2: Claro! Já. Muitas vezes! As consultas. Muitas. Muitas vezes. Acho que o maior trabalho foi isso, ter que mudar consulta, porque estava errado. Por exemplo, um exemplo esdrúxulo: O total de bolsistas na FAPESB. Buscava por feminino, tipo dava 500, buscava por masculino dava mais 500; o total tinha que dar 1000. Aí dava 1200. Era algum join que estava feito errado, ou porque a gente não conhecia o caminho lá no banco para pegar aquilo da forma correta. Ou porque tava faltando algum parâmetro tal. Porque a base do SIGA, apesar o sistema de bolsa ser a melhor estruturada(sic), (confidencialidade requerida) existem erros de modelagem... coisas que não deveriam ter sido feitas daquele jeito ali e foi... Então fazer consultas e trazer dados coerentes foi um trabalho bem árduo. Então, muitas vezes teve que rever a consulta que estava lendo ali. Porque a S1 dizia assim: "Oxente, a gente não pagou tantas bolsas de doutorado esse ano não. Não, está errado isso aí, pode rever. Volta certa consulta. Agora está certo."

Alex: Enfim, essas mudanças geravam mais por conta de erros mesmo do que de descoberta de novas formas de se fazer...

D2: Por conta de erro mesmo. Mais de mudança aí, mais de 90% dos casos é de coisa de se dizer assim: "Está errado!"

Alex: Vocês descobriram alguma forma de...

D2: Apresentar os dados melhor. Mas aí descobrir novas formas de apresentar, tipo, eu acrescentei um gráfico do tipo sunburst, que eu nunca tinha visto na vida, que é um gráfico redondinho... Mas foi, assim, acrescentar, eu não peguei uma "parada" que já tava pronta, que era para ordenar bolsistas por grande área e área. Aí apareciam as grandes áreas num círculo, dentro do círculo tinham as áreas, e aí o número de bolsistas no meio. Aí você vai brincando: Clica numa grande área e aí ela abre; aí mostra só as áreas pertencentes àquela grande área. É um gráfico bem dinâmico, bem diferente que eu implementei... mas foi acrescentar, não foi mudar uma coisa que estava pronta. Porque tem a estatística lá, que é a S1. Falo bastante dela porque ela que decide as "paradas" todas aqui. Principalmente de bolsa, é ela que manda lá. E aí ela disse assim: "Oh, para esse tipo de dado aqui, o gráfico é esse aqui." Sou eu que vou dizer que não é? Então era assim... Funcionou assim.

## **Princípio Plan for Scale**

Daniel: Agora... você conhece o conceito de escalabilidade? Ou isso é um palavrão?

D2: Não... não é um palavrão. Já escutei bastante e vamos se é a mesma que você está pensando. Que seria assim: você concebe um produto assim... seja o banco, seja o seu software por completo... e você projeta ele para ser escalável. Para você acrescentar coisas novas e ele prever a mudança. Ele vai crescendo, ele está pronto para crescer... basicamente isso. Seria isso?

Daniel: Isso é a ideia. Isso de maneira elegante.



D2: É elegante?

Daniel: Não, mas eu falo assim: Crescer de maneira elegante. Isso que você falou.

D2: É isso. Que ele esteja pronto para receber isso.

Daniel: Beleza. Perfeitamente. Nessa questão dos filtros de busca e tal. Vocês já pensam na escalabilidade? Por exemplo, vocês vão implementar o filtro. Aí você tem esse filtro aqui, será que se você for expandir ele você consegue expandir ele de maneira mais fácil ou o filtro que você coloca ele "morreu" ali? Ele implementou ali e acabou ali.

D2: Vamos lá. Vamos ver se isso responde à sua pergunta. O F1 ele teve um cuidado muito grande de otimizar as consultas. Porque é muito join. Em termos de...(entrevistada pensa consigo o termo desejado)... performance. A gente precisava fazer consultas que fossem otimizadas. Então toda vez que a gente... ele falou assim : "Olha, tenha cuidado com essas consultas porque elas pesam bastante no servidor e aí você vai ter um monte de gente fazendo essa consulta aí, e aí é um monte de consulta lá no servidor. Então vamos sempre pensar na escalabilidade aí em termos de performance". Então ele já fez pensando nisso e passou para mim para que as próximas eu fizesse pensando no mesmo. Daí eu fiz várias consultas de vários gráficos, que foram acrescentados, como keu falei. Pensando assim, dessa forma.

Daniel: Agora no sentido de você poder adaptar elas depois, aí não... necessariamente?

D2: Sim. De poder adaptar. Porque o F1 pensou uma arquitetura toda de uma forma que, por exemplo, acrescentar um novo gráfico no Observatório é tão simples quanto fazer uma nova consulta. Acrescentar um novo gráfico não. porque... acrescentar um novo parâmetro de busca... Não. Naquele gráfico dinâmico... Vocês estão usando como parâmetro só o observatório ou outro sistema?

Alex: O Observatório e o de bolsas.

D2: Mas no Observatório, na página inicial, tem um gráfico dinâmico. Acrescentar um novo parâmetro ali, em termos de desenvolvimento, é tão fácil quanto acrescentar uma nova função. Ele sai, pega todas as consultas, e joga ali. Então, ele pensou de uma forma que fosse muito mais fácil a gente colocar novos parâmetros de busca ali... Foi muito bem concebido(sic) essa arquitetura aí por ele. Inclusive, os novos gráficos que você tem lá no dashboard também são fáceis de implementar. Desde que já seja, tipo, já tem um gráfico lá do tipo boxplot. Você vai implementar um novo... Todo trabalho por trás já está pronto. Você pode acrescentar ali um novo. Não é tão fácil assim quanto fazer uma consulta nova como um gráfico dinâmico, mas é fácil também de desenvolver isso. Mas aí, quando eu tive que acrescentar um tipo de gráfico diferente, que é esse que eu falei para vocês, o sunburst... eu nem como é mesmo o nome desse gráfico, não lembro mais... mas o componente era esse, sunburst da biblioteca NVD3 do Angular. Aí sim eu tive que ver, olhar para a arquitetura inteira e ver como que eu tratava aquele tipo de gráfico. Mas um novo gráfico daquele, de tipos que já são existentes, é muito fácil de colocar.

Daniel: Aí você recebia da S1 e assim, o que você deveria fazer?

D2: Sim. Porque é ela que (trecho ininteligível)

Alex: E você sabe de algum tipo de prática, processo da engenharia de software, para que isso seja capaz de ser feito com essa facilidade que você está dizendo?

D2: Não. (trecho ininteligível). Nomear essa prática toda que F1 adotou? Seria isso?

Alex: Na verdade, é saber quais são as práticas que ele utilizou para...

D2: Pra que fosse feito dessa forma.

Alex: Algum tipo de padrão de projeto...ou coisa assim.

D2: Não. Eu sei que ele (trecho ininteligível) e ficou muito massa lá.

(Entrevistada questiona se F1 já foi entrevistado).

### **Princípio Early Feedback**

Daniel: Tudo que você implementa tem que reportar para S1 lá? (trecho ininteligível)

D2: Tem que reportar não. Ela que passa os filtros, ela que passa os tipos de gráfico. Porque ela já sabe o que vai ser colocado.

Daniel: E aí quando você implementa, já vai verificar com ela?

D2: Sim. Pra ver se está certo.

Daniel: Na mesma hora já.

D2: É... se eu fiz assim na mesma hora é outra coisa né... Tão logo fique pronto eu já ligo para ela: "Ô S1, dá uma olhadinha aí... Nessa parte aí."

Daniel: Tem alguma outra pessoa que olha essa parte também? Quando você implementa...

D2: A S2.

Daniel: Assessoria de planejamento... Ah, então só essas suas pessoas. Mas a S1 está mais próxima, né? Ou você olha com as duas, assim...

D2: Eu lembro que primeiro a gente sempre ia para S1. E depois para S2.

Alex: E era muito grande esse fluxo de ir e vir?

D2: Não. É tão fácil quando ligar para pessoa. Só se a pessoa não estiver disponível.

Alex: Não... Não estou falando de dificuldade. Estou falando de frequência mesmo. De ter várias reuniões. De ter várias "checkagens" delas com...

(Confidencialidade Requerida)

Daniel: Mas é bem informal, né? Por exemplo, não tem uma hora...

D2: A gente tem acesso livre aqui à sala das pessoas né... A menos que a pessoa esteja em reunião ou fazendo alguma outra coisa que não possa. A mesma coisa agora do Roberto Santos. Eu chego aqui (Entrevistada simula possíveis falas):

"Ah, estou detalhando agora aqui aditivo de prazo. O que precisa, o que não precisa ter?" "Aí o que é aditivo de prazo?" "Ah, então tá. Qual ramal eu ligo?". "Posso passar na sua sala?" "Pode." "Não, não pode." "Venha tal hora." "Está bom." "Fui." "Escrevi, gostei." "Vai ter o que no banco?" "Vamos modelar isso aí?" "Beleza. Isso aí vai ser assim, assim." "Está bom." "IHC1, a gente vai ter uma tela com isso e isso aqui, vai ter que fazer isso e isso aqui." Faz o wireframe, volta. "Aqui ó." "E aí? Está bom assim?" "Tá, está bom." Volta. (trecho ininteligível). Assim que eu tenho feito.

Alex: Vocês utilizam bastante prototipação, pacote?

D2: Bastante não. Protótipo, não. O que a gente fez para agilizar o processo? Vai esperar IHC1 fazer um protótipo bonitinho, direitinho? Não dá. Então vamos trabalhar com wireframe, que é um protótipo mais básico, né? Onde só custa lá a informação. Porque nesse momento, por exemplo, gerenciamento de dotação orçamentária é aqui com a S2. E aí eu mostro para ela, ela está interessada em saber qual é a cor que o botão vai ter, se vai estar mais em cima, mais abaixo, qual o pad de uma coisa para outra? De jeito nenhum! Ela quer saber qual dado vai estar lá e se ela vai poder fazer do jeito que ela quer. Do jeito que ela (trecho ininteligível) que se faça. Então a gente apresenta o wireframe. (A entrevistada se oferece para mostrar o wireframe em um momento posterior). E aí eu chego... Aí por exemplo, para gerenciar a dotação: É um cadastro, você tem a listagem da dotação, você tem (trecho ininteligível)... pode fazer um CRUD completo. Eu assumi que ela não precisava validar um wireframe de CRUD. CRUD, um CRUD básico. (trecho ininteligível). Agora, na hora de cadastrar a dotação, que envolvia vários campos, e que precisava ter várias informações que não era simples, não era básico, e precisava validar isso antes, então a gente chegou e fez o wireframe só do cadastro da dotação. A parte de poder excluir, editar, não sei o que... Não precisa validar isso. Entendeu? A gente que mostra uma interface para ela. A gente que dá uma interface para ela. E não é uma interface que é só para ela. É para fundação inteira. Então, não cabe para ela dizer assim: Ah, esse Squee(?) aqui está errado. Não, não, entendeu? É só em termos de validar informação. E é assim que a gente tem seguido.

Alex: É um tipo de prototipação, mas não é detalhado.

D2: Exatamente, só que menos detalhado. É a forma mais ágil que a gente encontrou de fazer isso.

## **Princípio Be Open-Minded**

Daniel: Bacana. Tá, vamos mudar para outro tema agora. Vamos falar que alguém falou para vocês desenvolverem... colocar um filtro novo lá. Aí vocês definem em reunião que um filtro vai ser colocado. Acontece de vocês, por exemplo... mudar como você vai fazer o filtro no meio do caminho, depois que a reunião tiver sido feita. Tipo, para você buscar mais coisa, para você encontrar informação nova. Acontece isso?

D2: Isso partindo de quem? Essa mudança, partindo de quem?

Daniel: Da própria equipe.

D2: Pode acontecer. Não vi acontecendo no Observatório não, mas pode acontecer. A gente depois viu que não fazia muito sentido fazer daquela forma que o cliente pediu e apresentar para ele uma nova forma de se fazer. Pode acontecer sim.

Daniel: Mas aí vai lá e valida com o cliente também.

D2: Exatamente.

Daniel: Mas assim, espontaneamente, isso existe na equipe? Por exemplo, se eventualmente...

D2: Existe. Eu mesma faço isso. No Observatório, não precisei fazer. Inclusive, o gráfico novo que eu acrescentei no... Pronto! Um bom exemplo! O gráfico novo que eu apresentei foi uma sugestão de F1, quando ele nem estava mais aqui. Ele veio para uma reunião para apresentar o Observatório para algumas pessoas, e falou assim: "Ó, o gráfico legal para gente colocar grande área e área de bolsistas é esse daqui." Foi uma sugestão.

Alex: O time tem uma predisposição de...

D2: Propor coisas novas.

Daniel: Isso surge naturalmente assim...

D2: Surge.

Daniel: Ninguém chegou e disse: "Olha, gente, vocês tem que ser criativos, inovadores e tal..."

D2: Não. Eu nunca ouvi isso de ninguém. Mas eu já ouvi um incentivo, né? O D1 incentiva bastante isso. Eu acho que o mais perto que eu cheguei de ouvir isso foi na avaliação de desempenho de (trecho ininteligível)... de incentivar isso daí. Mas nunca ouvi chegar e dizer assim: "Ó, seja criativo."

Alex: Quando tem alguma funcionalidade nova para vocês desenvolverem, a decisão vem lá de cima, e vocês simplesmente usam apresentação de cavalo e desenvolvem ou...

D2: Nãããão! Não vem com esse negócio de por negócio de cavalo aí... não. De jeito nenhum. Não é assim não.

Alex: E vocês tem alguma forma de sistematizar isso ou não? (trecho ininteligível) Quando aparece...

D2: Sistematizar, não.

Alex: Tipo, de vocês... vou tentar dar um exemplo aqui. De vir uma coisa assim e vocês como equipe, a equipe de desenvolvimento sentar, se reunir e tentar pensar em como melhorar aquilo ou outras formas...

D2: Sim. Vamos lá. (Divagações). Isso aí que você está dizendo seria assim: Se aquela forma de apresentar as coisas está errada, assim, não está atendendo, não está sendo a melhor forma, a

gente vê né? E aí volta: "Vamos ver uma forma melhor de fazer isso daqui." Senta... isso pode acontecer. Senta, e vamos ver uma forma. Aí envolve interface, então bota IHC1 no meio. "Oh IHC1, vocês tão com dificuldade assim, assim e assim... Esse negócio de ficar passando página eles não gostam". (Trecho ininteligível). Por exemplo, não que isso tenha de fato acontecido. E aí ele fala: "ah, isso aqui é menor." (Confidencialidade requerida). Basicamente assim. E a gente apresenta um prototipozinho pro cliente, ele validando; a gente faz.

Alex: Vocês tem uma prática de estudar novas tecnologias, novas práticas...

D2: Vocês quem?

Alex: Vocês, equipe de desenvolvimento.

D2: Você está falando equipe de desenvolvimento de um sistema específico ou você está falando da equipe inteira da informática?

Alex: Equipe em geral,né? Porque como os sistemas, enfim, têm...

D2: Algumas pessoas sim, algumas pessoas não. E aí quem tem o hábito de estudar novas coisas, novas tecnologias, estar sempre bem informado são as pessoas que mais têm know-how para propor coisas novas e serem mais criativas.

### **Princípio Avoid Bad Learning**

Daniel: Você falou quem mexe com a parte da estatística é S1 né?

D2: Sim.



Daniel: Tem alguma outra pessoa que mexe com isso?

D2: Não. Não que eu saiba.

Daniel: A validação é toda feita com ela? O D1 também, né?

D2: A parte estatística é ela.

Daniel: Mas, em questão de filtros, quem mais?

D2: A S2. Mas eu acho que a S2 é que mais, assim, para validar dados... E a S2 conhece muito a base do SIGA. Ela tem uma noção de... uma noção lógica muito boa. E de bancos de dados. Ela conhece. Quando ela pede uma consulta, por exemplo; não consulta pro Observatório; consulta assim para ela mesma... pros indicadores dela. Ela é nossa ponte com o Governo. Os indicadores que ela precisa mostrar pro Governo (trecho ininteligível). Daí, o que ela pede, (trecho ininteligível)... Ela sabe até dizer assim, você faz consulta de tal tabela com tal tabela que tem aí no SIGA.

Daniel: Ela não é da área de TI, de informática...

D2: Ela não é, mas ela... (Confidencialidade requerida). Ela sabe. Não nos termos que a gente sabe. Mas ela sabe. A ideia ela tem.

Daniel: Essa validação, acho que você já me respondeu isso, mas ela é quando precisa? Você não tem reuniões estruturadas só para isso...

D2: Teve reunião para dizer assim: "Ó, o sistema vai pro ar, e aí? Está faltando o que aí? Diga." "Assim, para gente, está pronto. Agora, toma aqui e diga aí." Aí, a gente sai anotando... "Olha, isso aqui é assim..." e faz as os pormenores.

Daniel: Mas eu falo no sentido assim, por exemplo, está lá na reunião só com a equipe de desenvolvimento. Aí você fala: "Ó, gente. A gente vai fazer isso daqui, e depois a gente vai fazer isso daqui, e depois a gente faz a validação. Tipo, a gente faz isso aqui hoje, depois a gente vai fazer isso daqui, e depois faz a validação". Você tem esse processo de desenvolvimento de software bem definido e você tem uma fase de validação ou não.

D2: Sim. Tudo que a gente faz tem que ser validado. Tudo que a gente faz as pessoas envolvidas aqui da FAPESB tem que dizer "Ó, está bom assim." "Não não, tá." Entendeu? Mas não está assim definido vai ser assim, vai ser tal dia... Não. A gente terminou uma parte, a gente viu que já dá para mostrar, mostra!" "Ah, melhor só mostrar no fim", a gente vai julgando, se necessário. Então só mostra no fim. Eu mesma faço assim no Roberto Santos. Se foi assim que F1 fez no Observatório, tem que perguntar para ele.

### **Princípio Broad skill set, big toolkit**

Daniel: Os sistemas que vocês têm hoje mostram informações do presente; o que está acontecendo hoje e o que aconteceu no passado.

D2: Não. É informação do passado.

Daniel: É informação do passado.

D2: O Observatório é informação do passado. Inclusive, está escrito assim: Esses dados correspondem de 2012 a 2015. E nossa consulta de importações de dados do SIGA é rodada uma vez por ano. Os dados de 2016 só vão entrar em 2017. É passado.

Daniel: Pro futuro, o sistema faz previsão ou não? Ou ainda não trabalha com isso.

D2: Não.

Alex: Não tem nenhum tipo de indicador...

(Entrevistada confirma pergunta)

D2: Não. Muito bom! Gostei da ideia.

(Entrevistadores pedem para que a entrevistada não comente as respostas com os outros entrevistados até a finalização da entrevista).

(PAUSA NA ENTREVISTA)

**\_Fim da Seção 2**

**Seção 3\_**

**Performance**

Daniel: Desempenho do sistema: Vocês têm algum teste específico para testar desempenho?

D2: A gente vai ter. A gente vai usar, se eu não estou enganada, o nome da ferramenta é (trecho ininteligível, confirmar nome da ferramenta). E o INF1 também. Ele faz uns testes lá de performance que tem no sistema.

Daniel: É INF1?

(A entrevistada dá informações sobre a estrutura física do prédio da FAPESB).

D2: O INF1 é o chefe do sistema. Se você perguntar lá o que ele faz de teste de sistemas. Ele é especialista em segurança da informação. Ele também faz uns testes de segurança. Não que seja o trabalho dele, que ele é o chefe de suporte. Mas ele é bem apaixonado por essa área... Então ele faz o quê, por exemplo? No Observatório, F1 lembra muito bem que tinha deixado umas consultas abertas lá que, se o cara quisesse fazer uma SQL Injection fazia, no Observatório, e, assim, como a base do Observatório é independente da do SIGA, o máximo de trabalho que o cara ia dar para gente é dar um DROP DATABASE lá e a gente fazer toda a rotina de importação novamente. Seria a pior coisa que a pessoa podia fazer, né? Porque esse sistema de consulta aí só tem consulta e pronto. Mas enfim, esses foram os testes aí...

Daniel: Entendi. Você já chegou a alguém, por exemplo, algum dos cabeças aí, pediram para vocês: "Ah, o sistema está muito lento. Dá um jeito aí."

D2: Do Observatório, nunca vi dizendo que tava muito lento não. Mas... Não, nunca vi não. Nunca me lembro de ter ouvido isso aqui na FAPESB. Mas já ouvi em outros lugares. E o que se faz é: reinicia o servidor aí.

Alex: Mas essa preocupação com o desempenho, vocês tiveram no desenvolvimento do sistema?

D2: O F1 teve. Até de ver com o chefe do suporte na época, que era o INF2: "Olha, separe uma máquina aí com 8GB pro Observatório porque tem consultas, consultas pesadas, e tal...".

Alex: E o sistema, enfim, é aberto né?

D2: Exatamente. A gente pode ter um fluxo grande aí de...

Daniel: Mas aí eles saberiam certinho falar quem que pediu para eles algum requisito de desempenho em específico.

D2: Quem saberia falar?

Daniel: Não. Estou perguntando se quem sabe isso é o F1 ou o INF1.

D2: Não. O INF1 hoje acho que não sabe dizer isso não. Mas assim, tipo, quem pediu o requisito e tal de hardware... Estamos falando de provisionamento aqui que vamos ter tantos acessos por dia... seria isso?

Daniel: Sim.

D2: Eu acho que não. Isso aí é uma coisa que partiu só da cabeça de F1 mesmo. Mas hoje já dá para dizer, até porque tem o Google Analytics, né? Hoje já dá para dizer sim. Quanto que a gente tem de acesso por dia...

Alex: Mas vocês, enfim, monitoram isso.

D2: Alguém monitora. O D1 faz isso. (trecho ininteligível). Dá uma olhadinha no Google Analytics. Eu nem tenho acesso. Não me envolvo com isso não. Aliás, não me envolvo com mais nada no Observatório. Agora, é Luana. Inclusive ela está participando aí das reuniões para

segunda parte aí, que parece que vai entra projetos. Ela é ótima aí para falar. Porque ela está pegando os requisitos.

Daniel: Você falou aí um pouco (confirma o nome do INF1)... INF1?

D2: O do suporte? Sim. Ele não estava na época que o Observatório foi implementado. Ele está agora. Mas ele faz os testes dele de performance lá.

Daniel: Hum. Mas aí é da cabeça de F1 mesmo na época. Entendi.

## **Security**

Daniel: Agora, a segurança do sistema. Por exemplo, vocês acessam bancos de dados. Esses bancos de dados têm algum tipo de segurança? Por exemplo, senha... Pra vocês daqui de dentro mesmo.

D2: INF1 que é a melhor pessoa para... Botou a palavra segurança no meio, e ele é o chefe do suporte. Ele que senha de todas essas coisas. Ele é a melhor pessoa para te responder isso: Qual é a segurança. Mas, o Observatório tem uma base só dele. A gente faz aquela importação que eu te falei uma vez no ano. De fora, não dá para acessar a base do SIGA, que seria a mais crucial, que ninguém pode acabar com ela.

Alex: E existe alguma coisa assim de perfil de acesso? Como o sistema é só de leitura.

D2: Perfil de acesso a base de dados. Eu acho que tem. Tem, que eu vejo que tem gente que só tem acesso de leitura... estou falando direto no banco, né? Do SIGA, tem.

Alex: Não, eu falo do Observatório.

D2: Não. Só se for alguma coisa, tipo assim, já do próprio SGBD. Já vier implementado isso. Mas a gente nunca utilizou isso. Não tem nem senha. Estou falando a verdade.

Eu até que tentei colocar senha lá no banco mas... tinha algum empecilho no meio do caminho lá que eu vi com o INF1. E a gente assumiu que não precisava perder tanto tempo com isso porque não era crítico. Só num sistema de consulta não é crítico.

Daniel: Agora, tem como alguém... pelo menos que vocês enxergam; sempre tem um jeito; de atacar o sistema. (Daniel faz piada sobre ataque).

D2: Ainda mais que o Observatório é externo.

Daniel: Vocês já pensaram algo assim, como desenvolvedores... questão de como...

D2: É aquilo... a gente não se preocupar. Eu nunca me preocuparia com esse negócio de segurança até conhecer o INF1 aqui. O F1 eu acho que também não. Por exemplo, a princípio ter deixado uma consulta dessa aberta. Deixar uma vulnerabilidade que é uma das mais fáceis, de SQL Injection. E aí só depois que... O INF1 na época ficava alocado na sala da gente, só depois ele vinha apresentando para a gente que a gente começou a se preocupar mais. Mas não... Nunca me preocupava com isso não. Mas hoje eu me preocupo. Mas é aquilo: como eu não sou especialista, eu pego coisinhas assim, sabe? Pra não deixar acontecer. Tipo agora... Mas com o Observatório, não. Não tem muita coisa com o Observatório. Porque não tem cadastro. Porque não tem manipulação de dados. É só visualização. Entendeu? Como o Observatório é só visualização, então a preocupação é bem mínima. Não é um sistema crítico.

Daniel: Agora, se o cara quiser fazer DDoS... você sabe o que é, né?

D2: Sim, sei.

Daniel: Se o cara quiser fazer um DDoS, só lamento...

D2: Só lamento, porque a gente não tem estrutura para prover... prever um tipo de ataque desse. E se ele quiser fazer, faça. Não é um sistema crítico. Se o Observatório passar um dia fora do ar, quem que vai morrer? Ninguém. Se o SIGA passar um dia fora do ar e o pagamento de "não-sei-quemzinho" não for feito, vai vir todo mundo aqui para porta da FAPESB.

Daniel: Eu também, viu? Eu sou bolsista.

(Entrevistada e Daniel brincam sobre a resposta acima)

## **Availability**

Daniel: Então, seguindo nessa linha também, eu acho que você respondeu um pouco, que vocês não se preocupam se o sistema está online 100% do tempo. Ele pode cair.

D2: Pode. É tanto que, por exemplo: o, uma preocupação que a gente tem; a gente tem pouco recurso de infra-estrutura, de hardware, né? Servidor e tal. Aí, quando pensou, o INF1 falou: "pro Roberto Santos, a gente não tem servidor. A gente tem que comprar um e tal. Se não conseguir, a gente vai tentar alocar ele aqui dentro com servidores que a gente já (trecho ininteligível). Sistemas que a gente já tem aqui rodando, e manda para PRODEB. E manda para rodar lá." A FAPESB tem um contrato lá com a PRODEODEB, eles veem isso melhor do que a gente comprar um servidor novo. E aí, qual o problema a gente tem com o sistema PRODEB? É que se estiver fora do ar, a gente tem que abrir um chamado, eles têm até 24 horas para responder, e aí para resolver um problema de sistema fora do ar é muito mais difícil. Um sistema que é gerencial, da FAPESB toda, não pode esperar isso tudo para voltar pro ar. Vai deixar a



galera toda aí sem trabalhar? Não. O Observatório? Pode. O primeiro sistema que a gente pensa em mandar é o Observatório. Por que, se ficar 24 horas fora do ar, não é crítico. Não é.

Daniel: Agora, por exemplo, assim: a partir de quanto tempo vocês consideram que aí já seria grave deixar o Observatório fora do ar? Por exemplo, 3 dias? Ou tanto faz?

D2: Grave?

Daniel: É. Teria algum período de tempo que vocês não considerariam sendo grave não? Tipo, se já passou uma semana...

D2: Cara, isso é muito relativo. Mas eu acho que uma semana é bem grave, né?

(Entrevistada e Daniel brincam sobre a resposta acima)

D2: 3 dias eu acho que seria crítico.

## **Usability**

Daniel: Você comentou também a questão de usabilidade. De como o usuário vai interagir com o sistema. Vocês têm lá o designer, qual o nome dele?

D2: IHC1.

Daniel: Ele é da área de design? Ele estudou um pouco de interface humano-computador?

D2: O mestrado dele é nessa área.

Daniel: Ah.

D2: E eu acho, no processo todo aí, eu acho que é muito importante ter uma pessoa que nem ele. Eu gosto muito das equipes autogerenciáveis, eu gosto que todo mundo sabe fazer tudo... Eu gosto muito dessa ideia. Mas ter um cara que nem ele, dedicado a isso, eu acho importante. Acho uma "mão na roda". Eu como programadora não ficar me preocupando se o botão vai ser verde ou ser azul. Eu amo isso.

Daniel: Agora, o cliente, quando ele vai pedir para vocês essas questões de interface. Vocês têm reunião para isso ou simplesmente...

D2: Ninguém pede questão de interface não. Uma ou outra pessoa que tem um pouco mais de conhecimento diz assim: "eu queria que tivesse um botãozinho assim que eu seleciono tal coisa", a pessoa explica assim: "um botãozinho eu seleciono isso e apareça isso". Tem alguns colaboradores que têm esse nível de pedir as coisas. Mas não passa disso. No máximo, a pessoa vem e diz assim: "eu quero poder gerar remessa disso aqui. Porque está difícil toda hora ter que ficar digitando nome por nome." Mas em questão de dizer em funcionalidade, né? Aí a gente traduz isso pro sistema.

Alex: Mas, por ser um sistema aberto ao público, vocês tiveram alguma preocupação com a questão da usabilidade?

D2: A preocupação toda da usabilidade vem de IHC1. A preocupação toda da usabilidade vem dele. Aí, a gente, inclusive, identificou quando F1 foi mostrar para... Aconteceu a mesma coisa quando eu fui mostrar para uma prima minha no sistema que eu estava ali naquela hora. Agora eu estou nesse daqui. Aí eu fui mostrar ela, ela entrou no sistema... Não sabia para onde ia. Entrou no Observatório, não sabia para onde ia. Porque não conhece a FAPESB. Não sabe nem o que é que é a FAPESB, nem para que a FAPESB está aqui. Uma pessoa que não sabe nem isso,

entra no Observatório e não sabe para onde que vai Observatório. Foi assim com a namorada de F1. (Circula no argumento). Como a gente faz isso ser uma vitrine, de forma que pessoa que, mesmo que não saiba isso, consiga caminhar ali dentro? A gente se fez essa pergunta... e ficou, assim, no ar...

Daniel: Agora, vocês não precisam formalizar isso não? Por exemplo: a pessoa pediu alguma coisa. Aí você escreve: documento de requisitos e leva para a reunião... Não tem nada disso aqui não, né?

(Daniel comenta com entrevistada experiência pessoal com o SIGA.)

## **Scalability**

(Daniel pede para Alex para fazer uma pergunta de escalabilidade)

Daniel: Agora eu gostaria de perguntar, na questão do Observatório como um todo. Ele está tranquilo para você adicionar funcionalidade nele, né?

D2: Demais. Pergunte aí que macumba F1 fez para pensar aquilo tudo ali que ele pensou, que ele fez uma arquitetura muito escalável.

Alex: Tanto com os outros módulos que vão vir agora, de busca...

D2: Com toda certeza.

Daniel: Isso, você não saberia dizer, se foi pedido para ele que fosse assim não, né?

D2: Não, isso aí é coisa da cabeça dele.

(Daniel faz piada com a resposta da entrevistada).

## **Reliability**

Alex: Vocês tiveram alguma experiência de falhas no sistema até hoje?

D2: Sim. Mas pouca coisa assim. Tipo, tal gráfico não abria no Mozilla. Vamos consertar.

Alex: Compatibilidade...

D2: Incompatibilidade de navegadores. Que eu me lembre assim, foi só.

Alex: E vocês têm planos para mitigar esses problemas, caso aconteça?

D2: A gente vai corrigindo à medida que eles vão aparecendo. Mas como é um sistema que não é crítico, nunca é prioridade.

(Entrevistada pergunta entrevistadores sobre o quão crítico sistemas de Analytics geralmente são.

Alex dá um exemplo de um sistema de Analytics da Microsoft).

Daniel: Agora, o sistema do Observatório, dificilmente ele cai, né? Você falou né? Se cair também, não é problema. Vocês não se preocupam de deixar ele no ar muito tempo não...

## **Interoperability**

Daniel: Você falou que o Observatório conversa com o SIGA, né?

D2: Uma vez por ano.

Daniel: Uma vez por ano.

D2: Eles batem um papo aí.

Daniel: Aí tem o que? API, é, que eles liberam?

D2: Eles te geram uma consulta.

Daniel: Uma consulta SQL?

(Entrevista brinca com a pergunta acima)

D2: Uma importação de dados em SQL.

Daniel: Agora, ele conversa com algum outro sistema?

D2: Não. É uma importação de uma base para outra.

Daniel: Então, os softwares não conversam. É mais o banco de dados.

D2: O software não. É o banco de dados que a gente... e nem é automática essa... Inclusive, a gente pediu isso ao INF1, para botar essa consulta aí automática. Mas, como é uma coisa uma

vez por ano, nunca foi prioridade deixar isso automático, como job lá do banco. No início do ano que vem, a gente roda a consulta de novo.

## **Adaptability**

Alex: E questão de adaptabilidade do sistema: houve algum tipo de requisito para que o sistema fosse compatível com diversas plataformas, diversos navegadores, celulares...

D2: Sim. Que fosse responsivo. Agora, eu não sei dizer se isso foi um requisito do cliente ou se, como o D1 que é entusiasta dessa parte aí, se foi ele disse: "essa 'parada' aí vai ter que ser eu!" Mas tinha que ser, um negócio de acesso externo, para qualquer pessoa, tinha que ser, no mínimo, responsivo.

Alex: E ele, enfim...

D2: Ele é responsivo.

Alex: E roda em qualquer plataforma...

D2: Em qualquer navegador.

Alex: Em qualquer navegador...

D2: Só o IE, que tem que ser o 11 para cima. Acho que do 10 para baixo, nem a Microsoft mais suporte. Então, não é a gente que vai dar.

(Daniel brinca faz piada com resposta acima).

Daniel: Mas aí quem testou isso foi F1? D1?

D2: A gente que vai testando. A gente mesmo.

Alex: Mas não teve nenhum tipo de especificação, de documentos...

D2: Não.

Alex: Você pode dizer quais ferramentas foram utilizadas no desenvolvimento do Observatório?

D2: A linguagem básica é o Java 8, Framework MVC Spring Boot, o Angular pro front-end, Bootstrap, o Eclipse como ferramenta de desenvolvimento, Postgres como banco. Só.

(Entrevistada confirma se há mais algo a se perguntar. Entrevistadores e entrevistados brincam em teor amistoso).

D2: Ah, e o Apache Tomcat no servidor. JBoss. Jet.

### **Fim da Seção 3**

(Finalização da entrevista).