

UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL  
INSTITUTO DE INFORMÁTICA  
CURSO DE CIÊNCIA DA COMPUTAÇÃO

ATHOS BERNARDO LAGEMANN

**Entrevista Guiada com base em Estrutura  
de Perguntas para Modelagem de Processos  
de Negócio**

Monografia apresentada como requisito parcial  
para a obtenção do grau de Bacharel em Ciência  
da Computação

Orientador: Prof<sup>a</sup>. Dr<sup>a</sup>. Lucineia Heloisa Thom

Porto Alegre  
2019

UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL

Reitor: Prof. Rui Vicente Oppermann

Vice-Reitora: Prof<sup>a</sup>. Jane Fraga Tutikian

Pró-Reitor de Graduação: Prof. Vladimir Pinheiro do Nascimento

Diretora do Instituto de Informática: Prof<sup>a</sup>. Carla Maria Dal Sasso Freitas

Coordenador do Curso de Ciência de Computação: Prof. Sérgio Luis Cechin

Bibliotecária-chefe do Instituto de Informática: Beatriz Regina Bastos Haro

“When you’re the author of your own fate,  
you don’t want to write a tragedy.”

— *Chris Hadfield*

## **AGRADECIMENTOS**

Agradeço aos meus pais, Vladimir e Giovana, que sempre estavam dispostos a me ajudar, qualquer que fosse o momento de necessidade. Sem vocês eu jamais teria chegado onde cheguei.

À Maria Eduarda Balsan, pelo apoio, amor, cuidado e carinho incondicional nos momentos difíceis. Sem você a jornada teria sido muito mais difícil e pesadosa. Ao meu irmão Thales, pelo apoio e por sempre estar disposto a conversar, seja sobre o que for. Obrigado por me ensinar a ser um irmão mais velho.

Ao Gabriel Conte e ao Lucas Biff, pois sem vocês, nada disso teria valido a pena.

Ao Rafael Ruwer pela ajuda e mentoria com as dúvidas técnicas, e por sempre estar disposto a ajudar. À Juliana Cardoso pelas dicas de design, e pelo apoio que me destes ao longo deste ano. À Alice Wiener pela compreensão e apoio em momentos difíceis. Ao Renan Trévia, pela disposição a ajudar e pelas dicas sobre assuntos acadêmicos.

Ao Luis Otávio Soares, pela solicitude e disposição em ajudar. E por fim, agradeço à Lucinéia, pela paciência e profissionalismo, e por ter aceitado me orientar mesmo do outro lado do oceano.

## RESUMO

Gerenciamento de Processos de Negócio (*Business Process Management* - BPM) é a ciência que estuda como o trabalho é feito dentro de uma organização para garantir resultados consistentes e obter vantagem onde possibilidades de aperfeiçoamento forem identificadas. Neste contexto, o termo aperfeiçoamento pode ter diferentes significados, dependendo dos objetivos da organização. Exemplos comuns de objetivos de aperfeiçoamento incluem a redução de gastos, a redução do tempo de execução e a redução da taxa de erros. BPM atua em toda a cadeia de eventos, atividades e decisões, que por sua vez agregam valor à organização e seus clientes. Estas cadeias de eventos, atividades e decisões são chamadas de processos. A documentação dos processos é feita através de descrições textuais e modelos de processo. As descrições textuais são obtidas através de entrevistas com o responsável pelo processo, e a partir delas é possível gerar os modelos de representação do processo. Modelos de representação consistem em uma representação gráfica do processo e seguem uma notação bem definida, o Modelo e Notação de Processos de Negócio. O problema muitas vezes encontrado na etapa de descoberta do processo consiste na ineficiência, por parte do entrevistador, para obter informações importantes a respeito do processo. Quando não devidamente estruturadas, as entrevistas podem se mostrar ineficientes na descoberta de detalhes importantes sobre um processo. Para evitar a perda de detalhes relevantes durante uma entrevista, este trabalho propõe uma estrutura de entrevista guiada, extraída da literatura. Esta estrutura pode tanto guiar o analista novato durante uma entrevista para modelagem de processos, quanto auxiliar o analista mais experiente a abordar detalhes do processo que possam passar despercebidos.

**Palavras-chave:** Business Process Management. desenvolvimento móvel. engenharia de software. entrevista guiada. iOS.

## Question-based Structure for Business Process Modeling Guided Interview

### ABSTRACT

Business Process Management (BPM) is the science that analyzes how the work inside an organization is done, to ensure consistent results and to obtain advantages where improvement possibilities are identified. In this context, depending on the organization's objectives, the term improvement can be associated with different meanings. Usual examples of improvement objectives are cost saving, and the reduction of execution time and error rate. BPM is present in the whole chain of events, activities and decisions, which in turn adds value to both the organization and its clients. These chains of events, activities and decisions are called processes. The process documentation is achieved through textual descriptions and process models. The textual descriptions are obtained through one or more interviews with the process owner, and from that it is possible to generate models that represent the process. Representation models consist of a graphical representation of the process that complies to a well-defined notation, the Business Process Model and Notation. A common issue in the process discovery stage consists in the inefficiency, by the interviewer, to obtain important information about the process. When not properly structured, the interviews can prove inefficient in the detection of relevant details about a process. To avoid the loss of relevant information during an interview, this paper proposes a guided interview structure, extracted from the literature. This structure may guide both the inexperienced analyst through a process model interview, and the experienced analyst to find details about the process that otherwise would not be acknowledged.

**Keywords:** Business Process Management, guided interview, mobile development, software engineering, iOS.

## LISTA DE FIGURAS

Figura 3.1	Aceitação das distribuições <i>iOS</i> e <i>Android</i> .....	22
Figura 3.2	MVC do modo como é comumente implementado.....	26
Figura 3.3	MVC com o <i>Coordinator</i> aplicado e camadas bem definidas.....	26
Figura 3.4	The Interviewer, no TestFlight.....	28
Figura 3.5	<i>Interview Assistant</i> , da <i>INTERSOG LLC</i> - Entrevista .....	29
Figura 3.6	<i>Interview Assistant</i> , da <i>INTERSOG LLC</i> - Gravação de voz.....	30
Figura 4.1	Diagrama de frequência de contribuições.....	32
Figura 4.2	Distribuição dos sistemas operacionais de dispositivos móveis .....	33
Figura 4.3	Diagrama de classes.....	38
Figura 4.4	Menu principal do aplicativo, <i>iPhones</i> 6s e 11 Pro Max .....	39
Figura 4.5	Menu principal do aplicativo, <i>iPad Pro</i> 12.9 polegadas .....	40
Figura 4.6	Perguntas de resposta longa.....	41
Figura 4.7	Perguntas de resposta curta e <i>compliance</i> .....	42
Figura 4.8	Aviso de sucesso e alerta de deleção .....	43
Figura 4.9	Revisão de entrevistas e <i>empty state</i> .....	44
Figura 4.10	Exemplo de compartilhamento .....	45
Figura 5.1	Conhecimento em BPM.....	47
Figura 5.2	Familiaridade com produtos da Apple.....	48
Figura 5.3	Satisfação com produtos da Apple .....	48
Figura 5.4	Avaliação da interface.....	50
Figura 5.5	Cobertura das perguntas da estrutura.....	52
Figura 5.6	Elementos da notação BPMN.....	54
Figura 5.7	Modelagem do processo pelo analista 1 .....	55
Figura 5.8	Modelagem do processo pelo analista 2 .....	55

## LISTA DE TABELAS

Tabela 3.1	Comparativo das aplicações analisadas .....	31
Tabela 5.1	Avaliação do sistema - Feedback dos analistas .....	49
Tabela 5.2	Parecer dos analistas .....	51
Tabela 5.3	Utilidade do aplicativo.....	52

## LISTA DE QUADROS

2.1 Estrutura da entrevista guiada - Perguntas quantitativas.....	19
2.2 Estrutura da entrevista guiada - Perguntas qualitativas.....	21
C.1 Perguntas quantitativas .....	64
C.2 Perguntas qualitativas .....	65

## LISTA DE ABREVIATURAS E SIGLAS

BPM	<i>Business Process Management</i>
BPMN	<i>Business Process Model and Notation</i>
MVC	<i>Model-View-Controller</i>
MVVM	<i>Model-View-ViewModel</i>
MVP	<i>Minimum Viable Product</i>
PMI	<i>Process Management Institute</i>
UI	<i>User Interface</i>
UX	<i>User Experience</i>
XML	<i>Extensible Markup Language</i>
WPM	<i>Workflow Process Model</i>

## SUMÁRIO

<b>1 INTRODUÇÃO</b> .....	<b>12</b>
1.1 Objetivo.....	13
1.2 Principais contribuições .....	14
1.3 Organização do trabalho .....	14
<b>2 FUNDAMENTOS DA ENTREVISTA BPM</b> .....	<b>15</b>
2.1 A entrevista .....	16
2.2 A estrutura de entrevista guiada .....	18
<b>3 FUNDAMENTOS DO DESENVOLVIMENTO MÓVEL</b> .....	<b>22</b>
3.1 Linguagem .....	23
3.2 Arquitetura e decisões técnicas.....	23
3.2.1 MVVM.....	24
3.2.2 <i>Coordinator</i> .....	25
3.2.3 CocoaPods.....	26
3.3 A distribuição do aplicativo .....	27
3.4 Trabalhos Relacionados.....	28
<b>4 IMPLEMENTAÇÃO DO PROTÓTIPO PARA ENTREVISTA GUIADA</b> .....	<b>32</b>
4.1 Análise de requisitos .....	32
4.2 O Protótipo .....	33
4.2.1 Modo entrevista .....	34
4.2.2 Modo revisão .....	34
4.2.3 Compartilhamento de resultados.....	35
4.3 Modelagem dos casos de uso e diagrama de classes.....	35
4.4 Interface .....	37
<b>5 AVALIAÇÃO DO PROTÓTIPO</b> .....	<b>46</b>
5.1 Teste de usabilidade .....	47
5.1.1 Sobre os analistas .....	47
5.1.2 <i>Feedback</i> dos analistas .....	49
5.2 Teste de completude .....	53
<b>6 CONCLUSÃO</b> .....	<b>56</b>
6.1 Limitações.....	58
6.2 Perspectivas de trabalho futuro.....	58
<b>REFERÊNCIAS</b> .....	<b>60</b>
<b>APÊNDICE A — MODELAGEM DO ANALISTA 1</b> .....	<b>62</b>
<b>APÊNDICE B — MODELAGEM DO ANALISTA 2</b> .....	<b>63</b>
<b>APÊNDICE C — ESTRUTURA DE ENTREVISTA GUIADA PROPOSTA</b> .....	<b>64</b>

## 1 INTRODUÇÃO

Tendo como objetivos a garantia de resultados consistentes e a obtenção de vantagens a partir de aperfeiçoamentos no fluxo de trabalho, o Gerenciamento de Processos de Negócio (*Business Process Management - BPM*) estuda e analisa como o trabalho é feito em uma organização (DUMAS et al., 2013). O objeto de estudo do BPM é um processo, neste caso definido como uma cadeia de eventos, atividades e decisões. Exemplos de possíveis vantagens desejadas incluem o corte de gastos desnecessários, a redução do tempo de execução de um processo, e a redução da taxa de erros em um processo (DUMAS et al., 2013).

Ainda segundo Dumas et al. (2013), o ciclo de vida em BPM se divide em seis etapas cíclicas, conhecidas como identificação, descoberta, análise, redesenho, implementação e monitoramento. A etapa de descoberta é a fase em que os processos são modelados na forma *as-is*, ou seja, da forma como são feitos na organização. Esta será a etapa abordada neste trabalho.

Para iniciar a análise de um processo, etapa de descoberta, é necessário que o analista de BPM tome conhecimento de qual é o passo a passo do processo em questão. A partir de uma entrevista entre o analista e o responsável pelo processo, é obtida uma descrição textual de como as coisas ocorrem no processo. Este é o primeiro artefato adquirido pelo analista. Tendo a descrição textual do processo, é possível gerar uma representação gráfica do processo, um *modelo de processo*. Este modelo gráfico se baseia em uma Linguagem Extensiva de Marcação (*Extensible Markup Language - XML*) específica para a modelagem de processos, conhecida como Modelo e Notação de Processos de Negócio (*Business Process Model and Notation - BPMN*) (OMG, 2019).

Entrevistas com o responsável por um processo muitas vezes são difíceis de serem feitas de forma eficiente, ainda mais se o analista estiver nervoso ou sob pressão. Durante qualquer entrevista de processos, detalhes importantes podem ser perdidos. Como Sommerville (2011) exemplifica, o entrevistado pode supor que certas coisas são triviais, evitando mencioná-las, bem como o analista pode não pensar em certas possibilidades, não inquirindo sobre o assunto. Para evitar a perda de informações durante sessões de modelagem de processos, como exemplificado por Sharp e McDermott (SHARP; MCDERMOTT, 2009), muitas vezes o analista precisará preparar, de antemão, uma lista de tópicos que devem ser cobertos. Mesmo cobrindo os tópicos necessários, certos detalhes podem perder-se por falta de experiência do analista durante a condução da entrevista.

Buscando diminuir a perda de informações durante a fase de descoberta do processo, este trabalho propõe uma estrutura de perguntas para a entrevista, extraídas da literatura. As perguntas seguem o princípio apresentado por Sharp e McDermott (SHARP; MCDERMOTT, 2009), de que uma abordagem mista de autores de Gerenciamento de Processos de Negócio e autores de Engenharia de Software seja mais eficiente no levantamento do processo. Esta estrutura visa cobrir todo o processo, com enfoque em partes que podem passar despercebidas aos analistas.

Dispositivos móveis trazem inúmeras vantagens ao usuário, sendo a maior delas a portabilidade. Além de serem facilmente transportados, possibilitam a gravação de áudio e o fácil compartilhamento de informações. Uma entrevista guiada em uma aplicação móvel ajuda tanto na hora de seguir o passo a passo, guiando a entrevista, quanto na hora de revisar e compartilhar os resultados obtidos na entrevista. Esses motivos deixam clara a utilidade de um protótipo que funcione em praticamente qualquer lugar, melhorando a experiência do analista e do entrevistado. A partir dessas premissas, um protótipo foi desenvolvido, seguindo a estrutura guiada proposta, com o objetivo de facilitar ao máximo a entrevista, a captação de informações e o compartilhamento das respostas obtidas.

## **1.1 Objetivo**

O principal objetivo deste trabalho é propor aos profissionais de BPM uma estrutura de perguntas que os guie ao longo de uma entrevista durante a fase de descoberta de processo. Os resultados da entrevista guiada devem possibilitar uma modelagem do processo, levando em conta o procedimento normal e as possíveis exceções que possam ser geradas. Acompanhando a estrutura de entrevista guiada proposta, o protótipo desenvolvido deve possibilitar a captura de informações durante a entrevista e o acesso às mesmas para a análise posterior. O protótipo deve ser portátil, podendo ser facilmente transportado até o especialista do domínio. Também deve possibilitar o seu uso com o mínimo de informações prévias a respeito do seu funcionamento, de forma que apresente uma interface intuitiva e de fácil acesso.

A estrutura proposta e o protótipo desenvolvido devem ser testados na prática em uma entrevista entre analista de BPM e o especialista do domínio. Os testes devem validar se a estrutura da entrevista gera artefatos que possibilitem a criação de um modelo gráfico, que represente o processo. Os testes também devem verificar a usabilidade do protótipo,

tanto em termos da interface do usuário (*User Interface* - UI) quanto na experiência do usuário (*User Experience* - UX).

## 1.2 Principais contribuições

Este trabalho propõe, a partir da literatura, uma estrutura de entrevista guiada para a modelagem de processos de negócio. A estrutura foi concebida com base em trabalhos de autores da área de BPM e de Engenharia de Software, como sugerido por Sharp e McDermott (SHARP; MCDERMOTT, 2009). O objetivo dessa estrutura é auxiliar a descoberta de processos, revelando detalhes que muitas vezes passam despercebidos durante a etapa de modelagem do processo.

Para tornar esta estrutura prática e testável, foi desenvolvido um protótipo para dispositivos móveis *iOS*, baseado na entrevista estruturada proposta. O protótipo funciona em dispositivos *iPhone* e *iPad*, e permite a navegação do usuário pela entrevista, a gravação de áudio, e a revisão e compartilhamento das informações obtidas durante a entrevista. No momento da escrita deste trabalho, o aplicativo está disponível apenas em modo de testes, sendo necessário acessar um link de testes<sup>1</sup> para obter uma versão testável.

## 1.3 Organização do trabalho

O trabalho está organizado da seguinte maneira. O Capítulo 2 traz os principais conceitos de entrevistas na área de BPM, aborda o embasamento teórico utilizado como base para a fundamentação da estrutura da entrevista guiada e apresenta a estrutura de entrevista guiada obtida. O Capítulo 3 apresenta algumas considerações a respeito do desenvolvimento de aplicativos móveis, as decisões de projeto tomadas, reúne os demais detalhes técnicos sobre a ferramenta desenvolvida e apresenta os principais trabalhos relacionados. O Capítulo 4 apresenta informações sobre a implementação do protótipo, as funcionalidades e o diagrama de classes gerado. O Capítulo 5 demonstra os testes feitos e os resultados obtidos. Finalmente, o Capítulo 6 traz as conclusões e considerações finais deste trabalho.

---

<sup>1</sup>Disponível em <https://testflight.apple.com/join/nX4uoSZ7>

## 2 FUNDAMENTOS DA ENTREVISTA BPM

Este capítulo tem como objetivo apresentar os conceitos básicos da dinâmica de uma entrevista na área de Engenharia de Software, as suas diferenças para uma entrevista na área de BPM, a metodologia utilizada para obter a estrutura de entrevista guiada e a estrutura de entrevista guiada resultante.

Independente se a abordagem for orientada a sistemas, como nos levantamentos de requisitos comuns em engenharia de software, ou orientada a negócios, como geralmente é a área de BPM, sistemas de informação e processos de negócio compõem contextos similares (SHARP; MCDERMOTT, 2009). Sharp faz referência à modelagem de processos em fluxo, *Workflow Process Modeling* (WPM), uma generalização que inclui *Business Process Modeling*.

A partir disso, pode-se utilizar as vantagens das duas abordagens para obter a maior cobertura possível durante uma entrevista, tornando a modelagem resultante da entrevista o mais rica quanto possível. Mesmo a literatura da área de desenvolvimento de sistemas não sendo específica para modelagem de processos, ainda pode trazer benefícios para a estruturação da entrevista guiada.

Ainda de acordo com Sharp e McDermott (SHARP; MCDERMOTT, 2009), a maioria da literatura da área de processos de negócio cobre conceitos, uma visão global, e a abordagem BPM, mas geralmente não entram nos detalhes da identificação, análise e modelagem dos processos em si, normalmente deixando a desejar. Por outro lado, a literatura da área de engenharia de software dá maior destaque às técnicas utilizadas para fazer a análise e modelagem do sistema, mas sua abordagem muitas vezes se limita ao sistema e suas regras, negligenciando o processo de negócio (neste trabalho também referido apenas por *processo*) em si.

É natural, portanto, que uma abordagem mista seja mais eficaz na cobertura de um processo de negócio. Podemos utilizar parte do conhecimento de levantamentos de requisitos da engenharia de software, focada no sistema, e parte do conhecimento de análise de processos de negócio, focada no processo como um todo. Como apontado pelo Instituto de Gerenciamento de Processos (*Project Management Institute - PMI*) (PMI, 2013), para um projeto ser bem-sucedido, um dos aspectos que a equipe do projeto deve considerar é a adoção de uma abordagem bem definida, para atender os requisitos.

Em abordagens baseadas em entrevistas, o entrevistado normalmente coloca todo o seu foco apenas no fluxo principal do processo, deixando completamente de lado os

casos de exceção (DUMAS et al., 2013). Para evitar esse tipo de erro, Dumas sugere que se considere também o caso *rainy day*, que explora o que pode dar errado ao longo do processo. A partir disso, como o nosso foco é aumentar ao máximo possível a cobertura das perguntas na estrutura da entrevista guiada, quando a entrevista chegar no processo em si, ambos os casos devem ser considerados: *sunny days* e *rainy days*.

*Sunny days* se refere ao caminho feliz de um processo, onde tudo acontece de acordo com o previsto e não há problemas durante a execução do mesmo. Considere o processo de compra em uma loja virtual, por um usuário. Os *sunny days* desde exemplo consistem em o usuário acessar a loja virtual, selecionar os produtos desejados, escolher a forma de pagamento e efetuar o pagamento. Em seguida, o processo é terminado com a confirmação do pagamento.

*Rainy days* se refere às exceções que podem surgir no decorrer do processo. Baseado no exemplo anterior, os *rainy days* englobam exceções como os produtos selecionados não serem incluídos no pedido, a forma de pagamento desejada estar indisponível, o pagamento não ser aceito pelo estabelecimento, entre outras situações. As soluções para os respectivos problemas também estão inclusas nos *rainy days*, como contatar o suporte da loja, escolher outra forma de pagamento, ou verificar as informações de pagamento.

Muitas vezes durante o processo de entrevista, o analista e o entrevistado perdem-se nos detalhes de como algum aspecto do sistema funciona (PRESSMAN, 2005). Posteriormente, estes detalhes podem não ser relevantes para a modelagem em si. Pressman sugere manter o nível de abstração adequadamente elevado, sem entrar em muitos detalhes do funcionamento de algo em específico. Portanto, ao longo da entrevista espera-se que o entrevistado dê detalhes gerais do funcionamento do processo, sem entrar em detalhes técnicos específicos das ferramentas utilizadas ao longo do mesmo.

## **2.1 A entrevista**

A estrutura de entrevista guiada proposta neste trabalho foi extraída da literatura com base em livros de autores da área de Engenharia de Software e da área de BPM. Esta estrutura foi concebida a partir do conteúdo sobre a literatura do levantamento de requisitos, no caso de Engenharia de Software, e sobre modelagem de processos *as-is*, no caso de BPM. No caso da literatura de requisitos, da Engenharia de Software, dentre as técnicas apresentadas, foi estudado o conteúdo apresentado sobre entrevistas com o cliente. Estes autores apontam diferentes aspectos a se analisar em uma entrevista, sendo

alguns deles aplicáveis a entrevistas com estrutura bem definida, como é o caso deste trabalho.

As definições dos autores analisados são aplicadas à estrutura quando trazem alguma vantagem para a entrevista e não entram em conflito ou contradição com as perguntas já contidas na estrutura. Autores de Engenharia de Software tratam do planejamento e estruturação da entrevista para o levantamento de requisitos, e trazem aspectos importantes a serem observados.

Sommerville (2011) define dois tipos de entrevistas, as abertas e as fechadas. Entrevistas fechadas possuem um conjunto predefinido de perguntas que serão feitas. Entrevistas abertas, por sua vez, não possuem um roteiro em específico, sendo explorada de acordo com o bom senso do engenheiro de requisitos. Ainda segundo Sommerville, na prática, as entrevistas costumam ser uma mistura de ambos os tipos. Mesmo recebendo a resposta para uma pergunta predefinida, pode ser que esta resposta leve a novas perguntas, que não poderiam ter sido planejadas anteriormente.

O tipo de entrevista abordado neste trabalho é a entrevista fechada, pois é a opção em que uma estrutura guiada melhor se encaixa. Entrevistas fechadas trazem a necessidade de um "espaço para improviso", dentro de uma estrutura guiada bem definida. Assim, chegamos na opção de incluir, no presente trabalho, um campo pergunta-resposta chamado "observações", de preenchimento opcional, ao final de cada seção em que tal possibilidade se aplica.

Ainda segundo Sommerville (2011), existem dois lados a serem considerados durante uma entrevista, para o levantamento de requisitos: o quão adequado o entrevistado é para dar informações acerca do que será analisado, e o objeto de análise em si. Este pensamento também pode ser estendido ao levantamento de processos.

Para tal, estes dois pontos podem ser divididos em duas análises, dentro de uma entrevista: a análise quantitativa e a análise qualitativa (DUMAS et al., 2013). A análise quantitativa busca obter o máximo de informações a respeito do interlocutor e suas especialidades, para entender seu contexto e a propriedade com que as informações são passadas. A análise qualitativa, por sua vez, busca o máximo de informações sobre o processo que será analisado.

## 2.2 A estrutura de entrevista guiada

A metodologia adotada neste trabalho consiste em buscar abordagens, ideias e perguntas que se apliquem à fase de descoberta do processo e sejam compatíveis com a estrutura. Em alguns casos, a pergunta não se aplica ao contexto na forma como é apresentada na literatura. Por exemplo, na análise de requisitos voltada para a utilização de um sistema de *software*, como muitas vezes ocorre no contexto de Engenharia de Software (SHARP; MCDERMOTT, 2009). Para estes casos, adaptações são necessárias para melhorar o entendimento, como por exemplo a reformulação da pergunta. Assim, algumas perguntas inclusas na estrutura possuem a observação *adaptado*, pois a sua transcrição literal prejudicaria o seu entendimento.

Pressman (2005) sugere a elaboração de um plano de entrevista, em antecedência à entrevista. Para tal, algumas diretrizes foram levantadas e seguidas neste trabalho. Duas destas diretrizes incluem o planejamento prévio do que será abordado na entrevista, neste trabalho sendo representado pela estrutura guiada, e a utilização de ferramentas automatizadas, neste trabalho representado pelo protótipo desenvolvido.

Primeiro, foram buscadas heurísticas gerais de como as entrevistas são organizadas, para então as perguntas e observações granulares serem analisadas. Como mencionado anteriormente, Sommerville divide as entrevistas em abertas e fechadas (SOMMERVILLE, 2011). Dumas separa as análises em quantitativa e qualitativa (DUMAS et al., 2013). Como estas duas definições não são conflitantes, ambas podem se aplicar à estrutura guiada. Assim, pode-se definir a estrutura da entrevista como *fechada*, pois ela segue um padrão de perguntas bem definido. Em seguida, a estrutura pode ser dividida em duas partes, a primeira visando efetuar uma análise quantitativa da organização, e a segunda buscando analisar qualitativamente o processo de negócio.

A partir da definição da metodologia utilizada, podemos agora introduzir a estrutura guiada resultante. A entrevista é estruturada em duas partes, que são compostas por diversas seções, que contém os pares pergunta-resposta propriamente ditos. Para facilitar a indexação das perguntas, definimos que o índice é composto por três partes, no seguinte formato: *x.y.z.*, onde *x* se refere à parte, *y* se refere à seção e *z* se refere ao índice do par pergunta-resposta.

O objetivo da primeira parte (tabela 2.1) é captar as informações quantitativas para a entrevista. As informações quantitativas tratam de respostas curtas e pontuais, para obter o contexto da organização e do interlocutor. A partir destas informações, pode-se ter no-

ção da exatidão das informações apresentadas pelo interlocutor (SOMMERVILLE, 2011). Por serem informações pontuais e não muito extensas, esta parte é quase exclusivamente por *input* de texto curto, prevendo uma única linha para a entrada de texto. A primeira seção da primeira parte se refere ao negócio que está sendo avaliado, isto é, a empresa ou instituição que o interlocutor representa. A segunda seção foca no interlocutor, buscando informações sobre o tempo de experiência e a expertise do mesmo no assunto.

Quadro 2.1: Estrutura da entrevista guiada - Perguntas quantitativas

<i>Assunto / Pergunta</i>	<i>Formato da resposta</i>
<b>1. Análise quantitativa</b>	
<b>1.1. Sobre o negócio em questão</b>	
1.1.1. Nome da empresa	Texto curto
1.1.2. Área de atuação	Texto curto
1.1.3. Quantidade de funcionários	Número / Texto curto
1.1.4. Contato	Texto curto
1.1.5. Observações	Texto longo / voz
<b>1.2. Sobre o interlocutor</b>	
1.2.1. Nome do entrevistado	Texto curto
1.2.2. Cargo do entrevistado	Texto curto
1.2.3. Formação do entrevistado	Texto curto
1.2.4. Tempo de empresa (meses / ano)	Número / Texto curto
1.2.5. Contato do entrevistado	Texto curto
<b>1.3. Sobre a entrevista</b>	
1.3.1. Qual ou quais fluxos serão cobertos nesta entrevista?	Texto longo / voz
1.3.2. Qual a relação do entrevistado com cada fluxo?	Texto longo / voz

Fonte: O Autor

O objetivo da segunda parte (tabela 2.2) é coletar as informações sobre o processo que está sendo analisado. Essa parte irá explorar o passo a passo do processo e obter informações sobre alguns aspectos que possam vir a ser relevantes para a posterior modelagem.

Pressman divide esta fase da entrevista em três conjuntos de perguntas (PRESSMAN, 2005), sendo elas as *iniciais*, *intermediárias* e  *finais*. O conjunto de perguntas inicial deve situar o interlocutor e definir o projeto (processo, no contexto de BPM). As perguntas intermediárias devem garantir ao entrevistador um melhor entendimento do problema que está sendo analisado. O conjunto de perguntas final deve verificar a eficácia da comunicação entre o interlocutor e o entrevistador.

Como definido por Sommerville (SOMMERVILLE, 2011), não raras vezes o interlocutor dá pouca importância aos casos de exceção, durante a entrevista. Para evitar o esquecimento dos casos de exceção, Sommerville sugere a divisão do processo em *sunny days* e *rainy days*. Assim, o conjunto de perguntas intermediário, que tratam do processo,

pode ser dividido em duas seções, uma tratando do fluxo principal do processo, e a outra cobrindo os casos de exceção. A partir disso, esta parte da entrevista foi dividida em quatro seções, cada uma com um objetivo bem definido.

A primeira seção tem como objetivos situar-se dentro do processo, entender a motivação do fluxo e entender como se inicia o processo. A segunda seção se refere aos *sunny days*, ou seja, o caso em que tudo ocorre de acordo com o previsto. Seus principais objetivos consistem em explorar o caso ideal do processo e coletar o seu passo-a-passo. A terceira seção, chamada *rainy days*, se refere aos casos de exceção dentro do processo. Isso visa explorar o que pode acontecer de errado e quais são as alternativas de solução para cada caso, o que muitas vezes passa despercebido pelo analista (SOMMERVILLE, 2011). A quarta e última seção se refere ao fechamento do processo. Ela tem como objetivos garantir que a comunicação entre o interlocutor e o entrevistador foi eficaz, e coletar informações de fechamento, atividades relacionadas que devem acontecer e demais observações que possam vir a ser relevantes.

O campo *observações* está presente ao final de diversas seções, para a conveniência do entrevistador, e sempre possui o campo de resposta no formato de texto longo e gravação de áudio. Este campo garante um espaço para anotações que podem não ter sido tocadas ao longo da seção, e que tanto o interlocutor quanto o entrevistador podem ter percebido a necessidade de observar, bem como perguntas alternativas que possam vir a surgir, a partir de alguma resposta obtida (SOMMERVILLE, 2011).

As perguntas da estrutura não são de resposta obrigatória, mas apenas uma sugestão de roteiro, de modo que fugas e desvios são permitidos, caso o analista perceba que algum caminho alternativo seria mais interessante. Para estes casos, o analista pode utilizar o campo *observação* para registrar outros assuntos e perguntas. No protótipo, o campo *observação* pode ser acessado tanto pelo menu da entrevista, quanto "pulando" as perguntas da estrutura.

Existem casos em que há mais de um especialista do domínio a serem entrevistados, que discordam de algum aspecto do processo, sem entrar em consenso. Para estes casos, o analista não deve tomar decisões, deixando para que, na impossibilidade de um consenso, uma autoridade superior tome a decisão de qual é o procedimento correto dentro do processo. Esses casos não são previstos na estrutura da entrevista e devem ser lidados de acordo com o bom senso do analista e dos especialistas do domínio.

Quadro 2.2: Estrutura da entrevista guiada - Perguntas qualitativas

<i>Assunto / Pergunta</i>	<i>Fonte da pergunta</i>
<b>2. Análise qualitativa</b>	
<b>2.1. Início</b>	
2.1.1. Você é a pessoa correta para esclarecer o fluxo que será analisado?	(PRESSMAN, 2005)
2.1.2. As suas respostas são <i>oficiais</i> ?	(PRESSMAN, 2005)
2.1.3. Qual é o objetivo deste fluxo?	(DUMAS et al., 2013)
2.1.4. Quem são os atores envolvidos neste processo?	(PRESSMAN, 2005)
2.1.5. Quais são os objetivos do ator dentro deste fluxo, para cada ator?	(PRESSMAN, 2005)
2.1.6. Suposição inicial (montar o cenário)	(SOMMERVILLE, 2011)
<b>2.2. Sunny Days (caso esperado / situação normal)</b>	
2.2.1. Pré-requisitos: o que precisa acontecer para que o fluxo em questão ocorra?	(PRESSMAN, 2005)
2.2.2. Caso normal: quais são os processos deste fluxo, dentro da situação esperada?	(SOMMERVILLE, 2011)
2.2.3. Quais são as principais tarefas ou funções realizadas por cada ator?	(PRESSMAN, 2005)*
2.2.4. Para cada atividade, qual é a pré-condição, qual é o objetivo e quem é o ator envolvido?	(PRESSMAN, 2005)*
2.2.5. Observações	
<b>2.3. Rainy Days (exceções)</b>	
2.3.1. O que pode dar errado? Que procedimento será tomado, neste caso?	(SOMMERVILLE, 2011) (DUMAS et al., 2013)
2.3.2. Descreva novamente o fluxo. Que exceções deveriam ser consideradas, à medida em que o fluxo é descrito?	(PRESSMAN, 2005)
2.3.3. Como os casos inesperados são resolvidos?	(DUMAS et al., 2013)
2.3.4. Observações	
<b>2.4. Final / Conclusão</b>	
2.4.1. Qual é o estado esperado do sistema, quando este fluxo terminar? O que mudou?	(SOMMERVILLE, 2011)*
2.4.2. Quais são as atividades relacionadas a este processo?	(SOMMERVILLE, 2011)
2.4.3. Alguma outra pessoa poderia me prestar informações adicionais?	(PRESSMAN, 2005)
2.4.4. Deveria eu perguntar-lhe algo mais?	(PRESSMAN, 2005)
2.4.5. Observações	

\* adaptado.

Fonte: O Autor

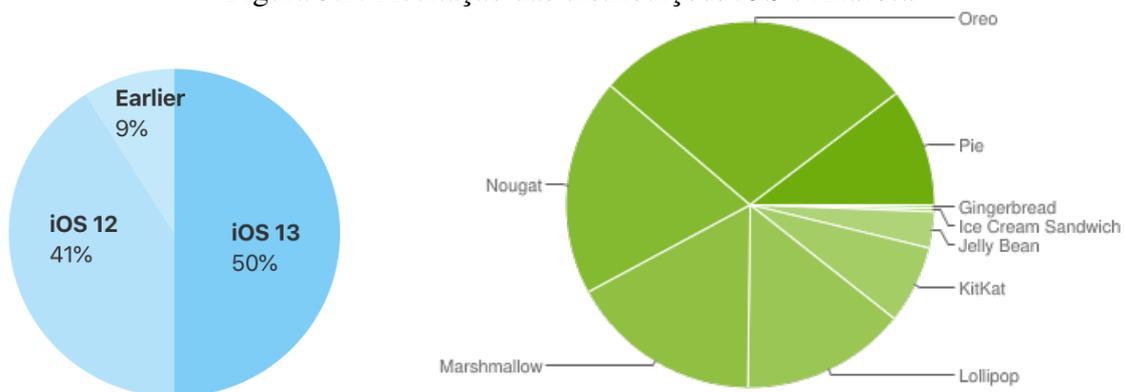
### 3 FUNDAMENTOS DO DESENVOLVIMENTO MÓVEL

Diversos aspectos devem ser levados em consideração durante o desenvolvimento de um aplicativo móvel. Os tópicos discutidos neste capítulo abordam itens como a plataforma utilizada, a linguagem, a IDE, as decisões arquiteturais e as considerações a respeito da distribuição do aplicativo.

A plataforma escolhida para o desenvolvimento do protótipo foi o *iOS*, sendo a sua principal razão a experiência prévia do autor com o ambiente, levando-se em consideração o tempo limitado para a implementação do protótipo. Posteriormente, pode haver também uma implementação nativa em *Android* ou uma implementação híbrida que suporte as duas plataformas.

Como visto na figura 3.1, comparando a aderência às duas versões mais recentes do *iOS* (12 e 13) e do *Android* (*Oreo* e *Pie*), fica claro que o *iOS* (APPLE, 2019a) obteve muito mais aderência do que o *Android* (GOOGLE, 2019). Isso evidencia uma menor diversidade de versionamento no ambiente *iOS*, em relação ao *Android*. Apesar dessas estatísticas de adoção, pode-se verificar que ainda existem mais usuários nas duas versões mais recentes do *Android* do que nas duas mais recentes do *iOS*.

Figura 3.1: Aceitação das distribuições *iOS* e *Android*



Fonte: Apple (esq.) e Google (dir.), 2019

Para uma solução compatível com ambos *Android* e *iOS*, poderia haver uma implementação em alguma tecnologia híbrida, como por exemplo *Flutter* ou *Ionic*. Contudo, aplicações híbridas não apresentam a mesma performance de aplicações nativas, tornando o aplicativo mais lento e com a usabilidade prejudicada. Estes aspectos devem ser levados em conta quando a decisão de implementação final do protótipo for tomada.

### 3.1 Linguagem

Para o desenvolvimento do protótipo, foi utilizada a linguagem *Swift 5*, compilada no *XCode 11.1*. Esta linguagem está documentada (APPLE, 2018) e ainda em fase de expansão, com cada grande versão trazendo novas funcionalidades. Uma grande vantagem do *Swift* é a possibilidade de migração automática da versão da linguagem, oferecida a cada nova grande versão, de modo que é bastante simples manter o código na versão mais recente da linguagem.

Para a definição da interface, foram utilizados arquivos no formato XIB, uma abstração em XML para facilitar a criação de elementos visuais. O projeto foi concebido para rodar em versões do sistema operacional a partir do *iOS 12*, que era o *iOS* mais recente no momento da criação do projeto, e boa quase todos os dispositivos *iOS* ainda em atividade suportam este sistema.

O projeto funciona normalmente em *iPhones* e em *iPads*, com pequenas variações e adaptações de interface. É particularmente importante que o projeto rode em *iPhone*, por motivos de praticidade e acessibilidade dos dispositivos. Contudo, o seu uso em *iPads* também é encorajado, pois uma tela maior oferece uma melhor disposição dos elementos na tela.

### 3.2 Arquitetura e decisões técnicas

Foi optado por uma implementação baseada no modelo de arquitetura *Model-View-ViewModel (MVVM)* (GOSSMAN, 2005), com o auxílio de *Coordinators* para o controle do fluxo do aplicativo. A principal vantagem desta arquitetura, neste caso, é a possibilidade da reutilização de uma mesma instância da *ViewModel* ao longo do fluxo, uma vez que as informações agregadas são atribuídas a uma única instância que contém todos os dados necessários.

O *Coordinator* é responsável por controlar o fluxo do aplicativo, tirando esta responsabilidade da camada *View*. Isso é desejável neste contexto, visto que as *ViewControllers* não serão responsáveis pelo gerenciamento dos dados. Desta forma, o *Coordinator* ficará responsável pela instanciação da *ViewModel* que será utilizada em cada fluxo, bem como a sua atribuição nas respectivas *ViewControllers*.

Nesta proposta, o gerenciamento do fluxo é abstraído das *ViewControllers* para o *Coordinator*, e o gerenciamento de dados é abstraído das *ViewControllers* para a *View-*

*Model*. Isto torna as *ViewControllers* muito menos interdependentes, e reforça o conceito de que as mesmas pertencem à camada da *View*.

A camada *ViewModel* pode ser vista como um agregador dos dados inseridos pelo usuário. Tudo o que é coletado pela *ViewController* é repassado para a *ViewModel*, de modo que toda a informação fica centralizada. A *ViewModel* pode ser passada de uma *ViewController* para outra através do *Coordinator*, de modo que só é necessária uma instância de *ViewModel* por fluxo.

No padrão MVVM, as *ViewControllers*, pertencentes à camada *View*, são agnósticas tanto a regras de negócio quanto aos modelos. Elas lidam apenas com tipos primitivos e se comunicam apenas com as *ViewModels*. É responsabilidade da *ViewModel* passar todos os dados, já processados, para a camada *View*.

Como as *ViewControllers* são agnósticas quanto às regras de negócio e ao modelo, cabe às *ViewModels* receberem os dados do modelo, processá-los com as devidas regras de negócio, e repassá-los como tipos primitivos para as *Views* necessárias.

### 3.2.1 MVVM

Apesar da arquitetura mais utilizada em *iOS* ser *Model-View-Controller (MVC)* (APPLE, 2012), ela não necessariamente é a melhor opção para todos os tipos de projeto. No contexto *iOS*, a Apple disponibilizou o framework *UIKit*, onde temos as *UIViewControllers*, que correspondem a uma tela do aplicativo. Nela, habitualmente configura-se o comportamento da *view*, os dados pelos quais ela será responsável, suas interações com usuário e o fluxo do qual ela faz parte, recebendo e repassando dados sensíveis do contexto e da aplicação como um todo.

A partir disso, surgiu uma discussão sobre o uso do MVC, sob o argumento de que as *ViewControllers* possuem responsabilidade demais. Projetos MVC grandes e complexos não raramente terminam com *ViewControllers* massivas. Um exemplo disso é o questionamento sobre a qual camada do MVC as *ViewControllers* pertencem. Por serem responsáveis pelo gerenciamento da interface, pertenceriam à camada *View*, mas como também acabam gerenciando dados e possuindo regras de negócio, podem ser igualmente colocadas na camada *Controller*.

Com isso em mente, o padrão MVVM cresceu muito em popularidade. Inicialmente proposto por Ken Cooper e Ted Peters, e anunciado por Gossman (2005). O MVVM abstrai toda a parte de gerenciamento de dados e de regras de negócio para uma

camada específica: a *ViewModel*. Assim, as *ViewControllers* não mais precisam se preocupar com o gerenciamento de dados, e podem se dedicar exclusivamente ao seu propósito original: o ciclo de vida da interface e o gerenciamento de interações com o usuário. Isso deixa bem definido que, na arquitetura MVVM, as *ViewControllers* pertencem à camada *View*, exclusivamente.

A *ViewModel* é responsável pelo gerenciamento dos dados da aplicação. É ela quem faz a ponte entre o modelo e a interface, abstraindo os dados para tipos simples. Seu principal objetivo é tirar as regras de negócio e estruturas mais complexas de dados das *Views*. Com isso, um dos ganhos é uma maior centralização do gerenciamento dos dados coletados pela camada *View*.

Não obstante, no MVVM as *ViewControllers* ainda são responsáveis pelo fluxo: devem saber de onde vieram e para onde irão, seja na pilha de navegação, seja sobre as *ViewModels* que serão recebidas ou passadas adiante. Para resolver este problema, surgiu um padrão: o *Coordinator*. Na arquitetura com fluxo controlado pelo *Coordinator*, este passará uma instância da *ViewModel* de uma *ViewController* à próxima *ViewController*, coletando as informações específicas de cada caso.

### 3.2.2 *Coordinator*

Apresentado no contexto *iOS* em 2015 por Khanlou (2015a) e posteriormente justificado e aprimorado em (KHANLOU, 2015b), o padrão *Coordinator* é responsável por controlar o fluxo do aplicativo, tirando esta responsabilidade das *ViewControllers*. Isso é desejável neste contexto, visto que as *ViewControllers* não serão responsáveis pelo gerenciamento dos dados. Desta forma, o *Coordinator* ficará responsável pela instanciação da *ViewModel* que será utilizada em cada fluxo, bem como a sua atribuição nas respectivas *ViewControllers*.

No exemplo demonstrado por KHANLOU, na figura 3.3 pode-se ver a diferença da utilização ideal do padrão MVC aliado com o *Coordinator*, em contraste com a implementação mais comumente encontrada do MVC (figura 3.2). Em sua proposta, Khanlou (2015b) abstrai o gerenciamento do fluxo das *ViewControllers* para o *Coordinator*, e o gerenciamento de dados das *ViewControllers* para a *ViewModel*. Isto torna as *ViewControllers* menos interdependentes e divide as suas responsabilidades entre as camadas *Model*, *View* e *Controller*.

Figura 3.2: MVC do modo como é comumente implementado



Fonte: Khanlou (2015b)

Figura 3.3: MVC com o *Coordinator* aplicado e camadas bem definidas

Fonte: Khanlou (2015b)

### 3.2.3 CocoaPods

*CocoaPods* é um gerenciador de bibliotecas externas, mais conhecidas como "*Pods*", desenvolvidas por terceiros e importadas no projeto para fins específicos e pontuais (CO-

COAPODS, 2019). Em pequena quantidade, facilitam e agilizam a construção do projeto. Em grande quantidade, prejudicam a clareza do projeto, aumentam o tempo de compilação do projeto, e criam uma dependência de terceiros. A manutenibilidade do projeto também é prejudicada, visto que os próprios *Pods* podem se tornar obsoletos devido a atualizações na linguagem ou no compilador. Tendo essas premissas em vista, é desejável balancear performance e praticidade. Quatro *Pods* foram utilizados na implementação do protótipo: *SkyFloatingLabelTextField*, *Reusable*, *RealmSwift* e *Zip*.

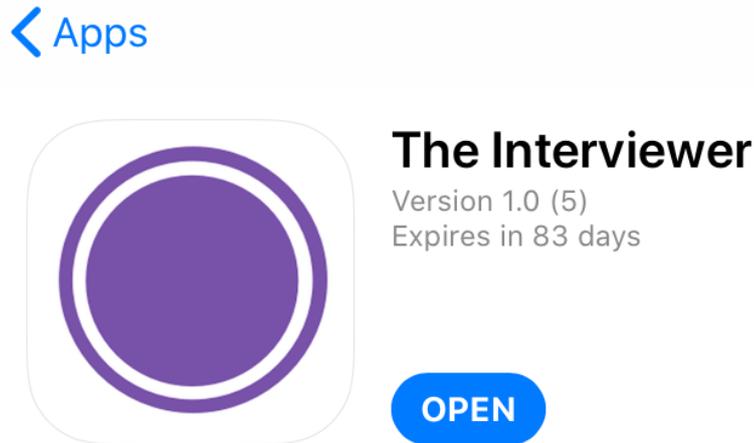
*SkyFloatingLabelTextField* (Langh Gergely Orosz (2019)) oferece um campo de texto bonito, versátil e de fácil utilização. *Reusable* (Halligon (2019)) auxilia na instanciamento e reutilização das XIBs, otimizando a vida útil dos elementos gráficos e aumentando a clareza do código. *RealmSwift* (Realm (2019)) é a base de dados local utilizada para a armazenagem dos dados já coletados pelo protótipo. É de fácil utilização e está disponível tanto em *iOS* quanto em *Android*, unificando a lógica do modelo de dados para ambos os ambientes, caso necessário. *Zip* (Marmelstein (2017)) é a biblioteca utilizada para a compressão dos dados coletados pelo protótipo, de modo que possibilita o compartilhamento de um único arquivo *zip* ao invés de inúmeros arquivos volumosos.

### 3.3 A distribuição do aplicativo

As duas plataformas mais populares para dispositivos móveis são o *Android* e o *iOS* (IDC, 2019). Estas plataformas baseiam-se pesadamente na distribuição de seus aplicativos via *PlayStore* e *AppStore*, respectivamente. A partir disso, como a plataforma escolhida foi o *iOS*, o aplicativo deve ser distribuído na *AppStore*.

Como ainda está em fase de testes, a Apple disponibiliza uma plataforma para testes, chamada *TestFlight* (APPLE, 2014). Esta plataforma permite uma atualização mais rápida do aplicativo, a separação entre grupos de *testers* e um sistema de gerenciamento das diversas versões do aplicativo. Para ter acesso à aplicação, o usuário deve fornecer o seu email ao distribuidor do aplicativo ou acessar um link disponibilizado pelo *TestFlight*. Para fins de teste, uma versão do protótipo está disponível em *TestFlight* (2019). Atualmente, o protótipo encontra-se na versão 1.0, *build 5*, que tem validade de 90 dias.

Figura 3.4: The Interviewer, no TestFlight



## What to Test

Testar os fluxos disponíveis, a gravação de áudio e o compartilhamento dos resultados finais.

Fonte: TestFlight (2019)

### 3.4 Trabalhos Relacionados

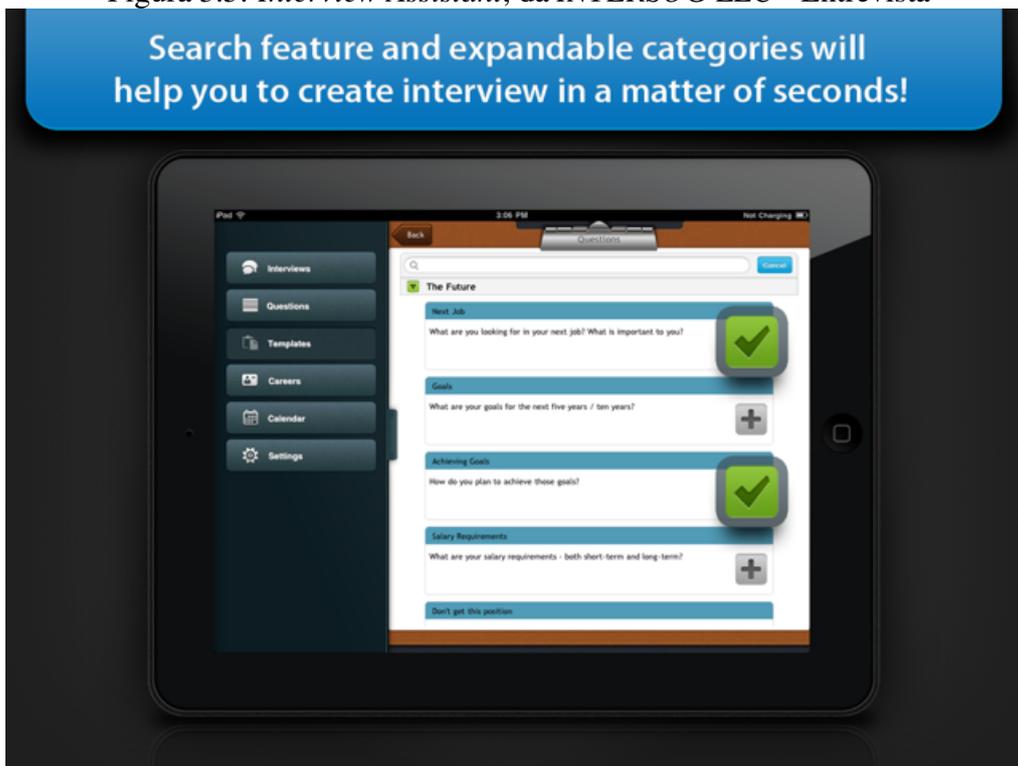
No que diz respeito a aplicações móveis, *Android* e *iOS* são os dois sistemas operacionais mais utilizados. De acordo com a estimativa da IDC (2019) para este ano, 100% dos *smartphones* utilizam ou *Android* ou *iOS*. A partir desta informação, aplicativos que apresentem funcionalidades similares à do protótipo implementado neste trabalho serão encontrados ou na *Play Store*, no caso do *Android*, ou na *AppStore*, no caso do *iOS*.

Buscando por aplicativos com foco em entrevista em ambas as lojas, a vasta maioria se refere a entrevistas de emprego. Dentre estes, a maioria não é relacionada à condução de uma entrevista. Ainda assim, foi possível localizar ao menos uma aplicação de cada loja, relacionada ao registro de uma entrevista, ainda que o foco seja uma entrevista de emprego.

Uma aplicação encontrada na *AppStore* é o *Interview Assistant*, da *INTERSOG* (INTERSOG, 2011), que permite a elaboração de uma entrevista, porém não possui o foco em descoberta de processos. O aplicativo oferece suporte tanto ao *iPhone* quanto ao *iPad*, e suporta a gravação de áudio para as respostas (figura 3.6).

Entre as funcionalidades mostradas na página do aplicativo, na *AppStore*, estão o uso de templates de entrevista, a criação de roteiros de entrevista, a definição de perfis para entrevistados, a possibilidade de vincular fotos e arquivos à entrevista, a digitação e gravação de áudio como resposta, e integração com o *Dropbox*, provavelmente a respeito dos resultados da entrevista.

Figura 3.5: *Interview Assistant*, da *INTERSOG LLC* - Entrevista



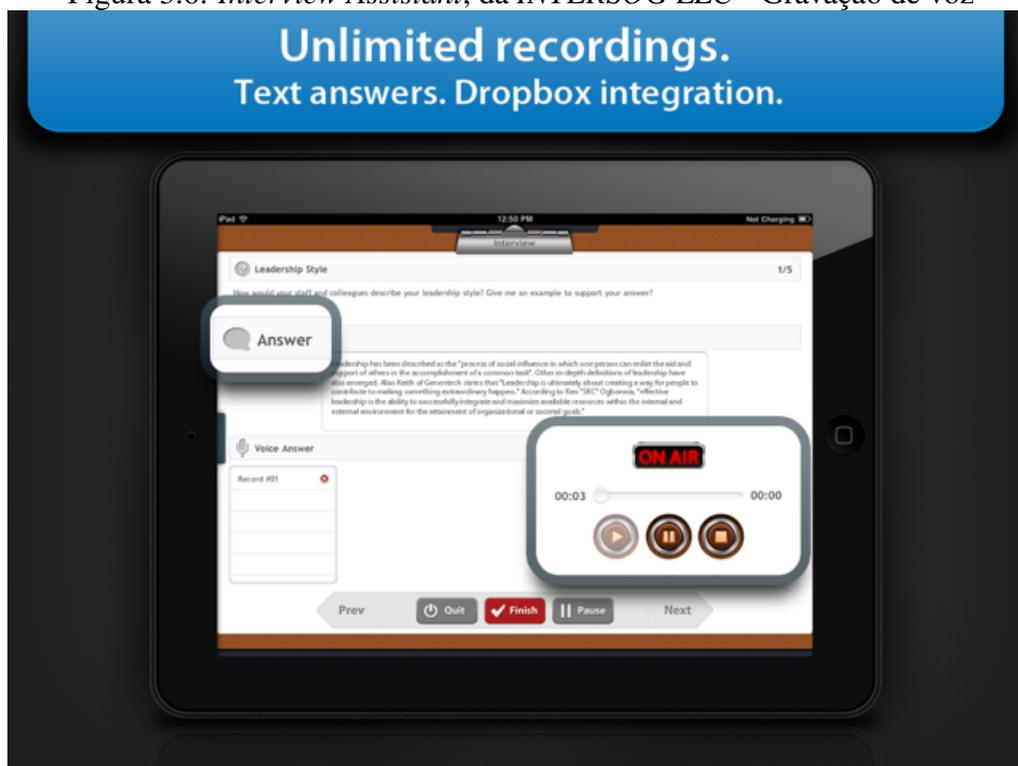
Fonte: Intersog (2011) (AppStore)

Há também outras funcionalidades, mas seu principal objetivo é a entrevista de pessoas. É importante salientar que esta solução requer compras dentro do aplicativo (*in-app purchases*) para liberar uma quantia ilimitada de entrevistas, que são inicialmente limitadas. Este modelo de negócios é conhecido como *Freemium*, pois mistura elementos gratuitos com elementos pagos.

Contudo, este aplicativo não recebe atualizações há vários anos. A primeira versão deste aplicativo registrada na *AppStore* é de 1 de novembro de 2011, e a versão mais recente do aplicativo é de 22 de março de 2012. Ainda em 2012, o sistema operacional corrente era o *iOS 5*, e a interface gráfica dos aparelhos da Apple eram completamente diferentes, como na figura 3.5.

A partir disso, é impossível obter o aplicativo nos *iPhones* atuais, pois a implementação está obsoleta e não é mais suportada pelo *iOS*. O website da companhia está

Figura 3.6: *Interview Assistant*, da *INTERSOG LLC* - Gravação de voz



Fonte: Intersog (2011) (AppStore)

fora do ar, e não há indícios de atividades recentes da mesma. O *Interview Assistant* pode ser encontrado em Intersog (2011).

Outra aplicação relacionada, encontrada na *Play Store*, é o *Interview Assistant* da Tech (2018). Este aplicativo apresenta mais similaridades com a ideia de uma entrevista guiada, mas ainda se limita à pessoa em si, e não ao processo. Ele é mais recente, sendo a última atualização até o momento deste trabalho de fevereiro de 2018, e suporta três tipos diferentes de perfis de entrevista, permite transcrição de áudio e compartilhamento dos resultados via e-mail. Um ponto importante a ser notado é que este aplicativo permite apenas a transcrição de áudio para texto, e não a gravação do áudio em um formato convencional.

Como pode ser verificado na tabela 3.1, pode-se afirmar que o protótipo implementado neste trabalho é o mais completo dos três aplicativos comparados. O *Interview Assistant* da *Intersog*, apesar de atualmente indisponível, é o segundo mais próximo do proposto.

Quanto a aplicativos de gravação de áudio, temos as opções nativas nas duas plataformas, bem como inúmeros aplicativos de gravação de voz. Com certa organização por parte do entrevistador, é possível se preparar para conduzir uma entrevista apenas

Tabela 3.1: Comparativo das aplicações analisadas

<i>Funcionalidade</i>	<i>Interview Assistant (Intersog)</i>	<i>Interview Assistant (Turtle Tech)</i>	<i>The Interviewer (o autor)</i>
Entrevista guiada	Presente	Ausente	Presente
Foco em descoberta de processos	Ausente	Ausente	Presente
Uso de templates de entrevista	Presente	Presente	Presente
Elaboração da entrevista	Presente	Ausente	Ausente
Anotação das respostas	Presente	Presente	Presente
Gravação de áudio	Presente	Ausente	Presente
Compartilhamento	Dropbox	E-mail	Serviços em nuvem, e-mail, mensageiros, <i>AirDrop</i> e outros
Modelo de negócios	Limitado ( <i>Freemium</i> )	Gratuito	Gratuito
Execução em dispositivos atuais	Indisponível	Disponível	Disponível

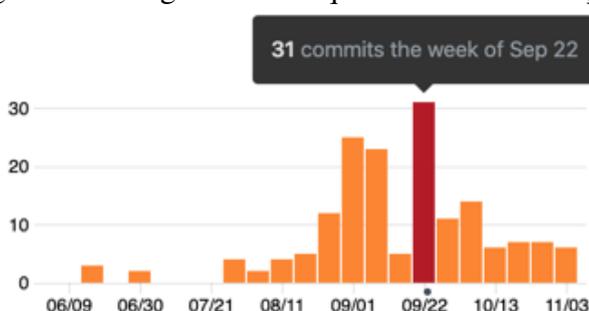
Fonte: O Autor

utilizando um aplicativo de gravação, mas a entrevista em si precisará ser montada com antecedência, caso a estrutura proposta não seja adotada.

## 4 IMPLEMENTAÇÃO DO PROTÓTIPO PARA ENTREVISTA GUIADA

O protótipo para a validação da estrutura de entrevista guiada foi implementado de 19 de junho a 5 de novembro, tendo a semana de 22 a 28 de setembro o maior número de contribuições (*commits*). Uma contribuição consiste em um *git commit*, que deve ser registrado toda vez que uma mudança substancial que não quebre o código é feita. O projeto teve, no total, 178 contribuições. Na figura 4.1 pode ser verificada a distribuição da frequência de contribuições, dividida em semanas.

Figura 4.1: Diagrama de frequência de contribuições



Fonte: O autor, 2019

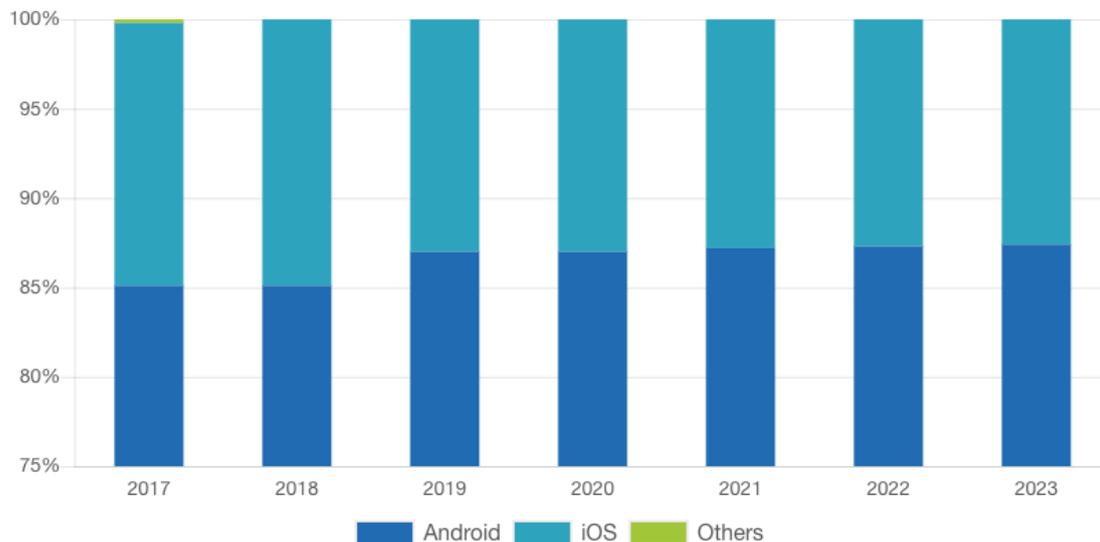
### 4.1 Análise de requisitos

Para que a estrutura guiada seja corretamente testada, o protótipo deve proporcionar um *Minimum Viable Product* (MVP), isto é, uma aplicação que cumpra os requisitos mínimos para demonstrar a viabilidade de uma ideia. Neste caso, a modelagem de processos de negócio a partir da estrutura de entrevista proposta.

Para tal, é necessário que o aplicativo grave e reproduza áudios, armazene dados em um banco de dados local e compartilhe os resultados da entrevista preferencialmente de forma fácil e prática. É desejável também que o aplicativo seja executável tanto em dispositivos de pequeno porte, como *smartphones*, quanto em dispositivos de maior porte, como *tablets*.

Entre os dois ambientes móveis mais populares, optamos pelo *iOS* principalmente pela familiaridade do autor com o ambiente, mas também por 91% dos dispositivos apresentarem compatibilidade com o *iOS 12* (figura 3.1), apesar da menor base de usuários (figura 4.2), e por haver poucas soluções nesta área de atuação, não apenas no contexto de BPM, mas também quanto a entrevistas em geral.

Figura 4.2: Distribuição dos sistemas operacionais de dispositivos móveis  
**Worldwide Smartphone Shipment OS Market Share Forecast**



Fonte: IDC (2019)

A utilização em ambas as plataformas pode ser suportada através de uma implementação complementar nativa para *Android* ou de uma implementação híbrida, que suporte ambas as plataformas. A principal desvantagem de uma implementação híbrida é, no momento, a *performance* inferior à de soluções nativas.

Os arquivos de áudio após uma gravação no *iOS* resultam em um arquivo *.m4a*, uma variação do *.mp4*. Este fato torna o compartilhamento vantajoso, pois o formato é comportado não apenas nativamente no ambiente *iOS*, mas também nos sistemas operacionais mais populares, como *Android*, *Windows*, *Mac OS* e *Linux*, de forma que a necessidade de conversões de formato é evitada.

## 4.2 O Protótipo

O protótipo tem como principal objetivo apresentar uma estrutura de entrevista guiada que facilite a coleta de informações de um processo em uma entrevista. Com isso, dois fluxos principais são obtidos: a criação da entrevista propriamente dita, e a revisão de entrevistas feitas anteriormente. Ao final de ambos os fluxos, o usuário tem a possibilidade de compartilhar a entrevista analisada.

A partir dos artefatos compartilhados pelo aplicativo, ao final da entrevista, deve ser possível revisar a mesma, da melhor forma possível. Para tal, deve ser possível acessar as perguntas e respostas, em um arquivo JSON, e deve-se também ter acesso aos áudios

gravados durante a entrevista. As respostas estruturadas em um formato JSON permitem uma automatização dos resultados, por algum outro programa posteriormente.

Idealmente, tanto o arquivo compartilhado quanto os áudios devem estar em um formato acessível nas plataformas mais comuns. Para tal, escolheu-se o formato *m4a* para os áudios, e o formato *zip* para o pacote de arquivos. O formato *m4a* é uma variação do *mp4*, e pode ser acessado nos mais diversos sistemas operacionais.

#### **4.2.1 Modo entrevista**

Este modo foi projetado para auxiliar o usuário durante a entrevista com o profissional da área. O usuário pode navegar pelas perguntas da entrevista, anotando as respostas nos campos de texto ou gravando-as no modo gravação. Nenhuma pergunta é obrigatória, visto que a estrutura é apenas uma sugestão. Assim, as perguntas podem ser puladas sem uma resposta registrada. Ao final do fluxo, o usuário tem a possibilidade de compartilhar um arquivo compactado contendo as respostas digitadas em um arquivo JSON, e os áudios gravados em formato *m4a* (no padrão da Apple). Além disso, existem as duas possibilidades de fechamento: descartar a entrevista ou salvá-la localmente.

#### **4.2.2 Modo revisão**

Este fluxo inicia-se com uma lista das entrevistas já salvas no aplicativo. Ao selecionar uma entrevista, o fluxo apresentado é análogo ao modo entrevista, mas sem a possibilidade de modificar ou adicionar novas informações. Ao final deste fluxo também há a possibilidade de compartilhar os dados coletados durante a entrevista.

A revisão não permite a alteração dos dados da entrevista, visto que qualquer alteração posterior ao momento da entrevista poderia adulterar o documento. Todas as informações necessárias devem ser obtidas durante a entrevista.

Ainda pode ser considerada parte da revisão a utilização dos artefatos gerados pelo aplicativo após o compartilhamento. O pacote de artefatos deve possibilitar a revisão de todas as perguntas e respostas, sejam elas anotadas ou gravadas, e deve também ser acessível nos sistemas operacionais mais utilizados.

### 4.2.3 Compartilhamento de resultados

Ao final de uma entrevista ou revisão, o usuário tem a possibilidade de compartilhar a entrevista. O material compartilhado consiste em um *json* com as anotações textuais da entrevista e os áudios gravados no formato *m4a*, compactados em um único arquivo *zip*. Ao selecionar a opção de compartilhar, uma lista de opções no padrão do *iOS* é apresentada. Esta lista deve variar de acordo com os aplicativos instalados pelo usuário no dispositivo em uso, mas sempre possibilitará o compartilhamento para as opções padrão do *iOS*, que incluem o *AirDrop* e o *Save to Files*.

O *AirDrop* permite que o usuário envie o arquivo de forma sem fio para outros dispositivos da Apple, como *iPhones*, *iPads* e *Macs*. A opção *Save to Files* salva o arquivo no *iCloud* do usuário, podendo ser acessado por outros dispositivos, e sincronizado automaticamente ao computador do usuário, caso o mesmo possua um sistema *Mac OS*. As outras opções possíveis de se utilizar variam de acordo com a preferência do usuário, mas incluem serviço de e-mail, serviços de armazenamento de dados na nuvem, e mensageiros como *Whatsapp* e *Telegram*, entre outros.

### 4.3 Modelagem dos casos de uso e diagrama de classes

A modelagem dos casos de uso se baseou nos dois principais fluxos do aplicativo: entrevista e revisão. Exceções também foram consideradas, para ambos os casos. Os atores aqui considerados são o Analista, que está efetivamente utilizando o aplicativo, e o entrevistado, que está colaborando com o analista.

- **Caso de Uso 1: Realizar uma entrevista**

**Atores: Analista e entrevistado.**

1. O analista seleciona a opção de "Iniciar Entrevista", a partir da tela inicial do aplicativo.
2. O analista e o entrevistado iniciam, então, a entrevista.
3. O aplicativo perguntará se o entrevistado permite a gravação do áudio.
  1. Caso o entrevistado permita, a opção "gravar" estará disponível nas repostas de texto longo.
  2. Caso contrário, a opção "gravar" estará indisponível, e a entrada por texto

se fará necessária.

3. O analista preenche as informações iniciais do entrevistado.
4. Como a entrevista guiada já tem as perguntas previamente definidas, o analista apenas as lê, uma por vez, ao entrevistado.
5. O analista registra no aplicativo a resposta do entrevistado (por voz ou texto).
6. Os passos 4 e 5 são repetidos até o final da entrevista guiada.
7. Ao final da entrevista, o analista pode compartilhar a entrevista, salvar os resultados localmente ou descartá-los.

- **Caso de Uso 2: Revisar Resultados**

**Ator: Analista**

1. O analista seleciona a opção de "Revisar entrevistas anteriores", a partir da tela inicial do aplicativo.
2. O analista pode, então, revisar cada pergunta e resposta feita durante a entrevista, mas não pode adicionar nem alterar os dados existentes.
3. Ao final da revisão, o analista tem a opção de compartilhar a entrevista revisada ou voltar ao menu anterior.

- **Caso de Uso 3: Compartilhar Resultados**

**Ator: Analista**

1. Ao final de uma entrevista ou revisão, surge a opção de compartilhar a entrevista.
2. Clicando no botão de compartilhamento, é apresentada a *sharesheet* do *iOS*.
3. O analista escolhe a opção de compartilhamento de sua preferência.
4. Ao final do compartilhamento, a *sharesheet* é dispensada.

- **Caso de Uso 4: A entrevista deve ser encerrada prematuramente**

**Atores: Analista e entrevistado**

1. Se por algum motivo, durante uma entrevista, seja necessário encerrar prematuramente, o analista acessa o menu da entrevista.
2. Acessando este menu, o analista então escolhe a opção de finalizar a entrevista, e a tela final é apresentada.
3. O analista deve escolher entre salvar o que foi feito até então, ou descartar tudo. Alterações e adições posteriores não poderão ser feitas.

A modelagem do diagrama de classes segue a arquitetura definida, baseada em MVVM e com o auxílio de *Coordinators*. Também foi necessário incluir a camada de *Serviços*, que é responsável pelo gerenciamento do banco de dados local e pela compressão dos dados em um arquivo *zip* para o compartilhamento.

Para manter o diagrama mais legível, os métodos das *ViewControllers* foram omitidos do diagrama, pois não influenciam a modelagem final. Alguns métodos padrão sequer são implementados pelas *ViewControllers*. É importante salientar que a comunicação entre a camada *View* e *Coordinator* precisa ser feita bilateralmente. No protótipo, foi feita de forma que um *Coordinator* instancia e apresenta uma *ViewController*, que por sua vez possui uma referência fraca de um protocolo implementado pelo *Coordinator*, chamada *delegate*. A *ViewController* aciona o *delegate* quando necessário, que trata-se de um protocolo implementado pelo *Coordinator*. Assim, o *Coordinator* recebe os diversos *triggers* de navegação que a *ViewController* pode necessitar.

Ainda sobre o diagrama de classes, note que a linguagem *Swift* suporta variáveis computadas, ou seja, variáveis que não possuem um valor próprio, mas sim são referências a propriedades de outras classes relacionadas. Estas variáveis computadas funcionam analogamente a *getters* e, portanto, na modelagem, foram atribuídas no campo de funções da classe. Isso pode ser verificado na classe *InterviewViewModel*, pertencente à camada *ViewModel*.

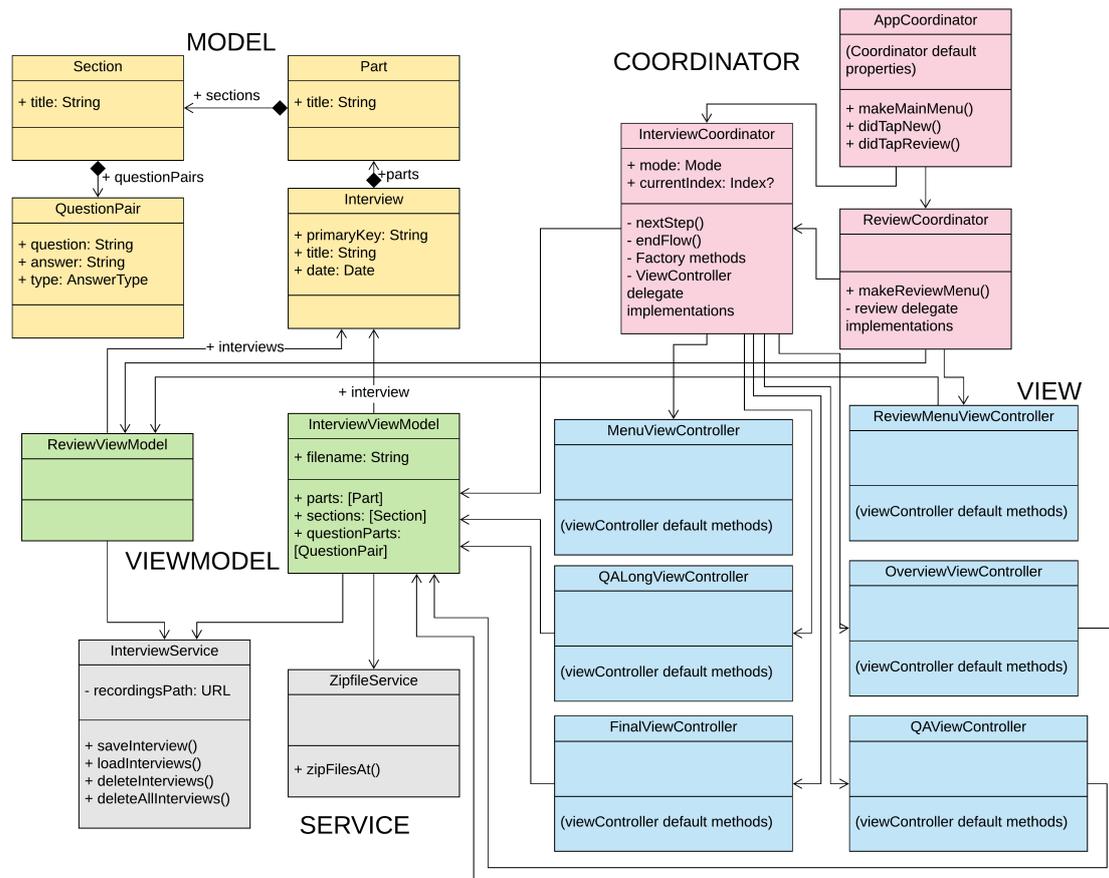
#### 4.4 Interface

Os princípios das guidelines para interface humano-computador da Apple (APPLE, 2019b) foram seguidos sempre que possível, para melhorar a experiência de uso do aplicativo. O aplicativo conforma, por exemplo, com os seis princípios de design propostos em Apple (2019b): Integridade estética, consistência, manipulação direta, *feedback*, metáforas e manter o usuário no controle.

A integridade estética foi seguida no sentido de que o aplicativo não apresenta elementos visuais que distraiam o usuário durante o uso. A consistência garante que os botões sejam posicionados de forma padronizada, na mesma área independente da tela ou contexto. A manipulação direta é respeitada nas ações sempre tendo um impacto imediato na interface, seja ao navegar pela lista de perguntas, interagir com o compartilhamento de entrevistas, ou mesmo na navegação de algum fluxo.

O *feedback* garante que as ações destrutivas sempre apresentem uma tela de con-

Figura 4.3: Diagrama de classes



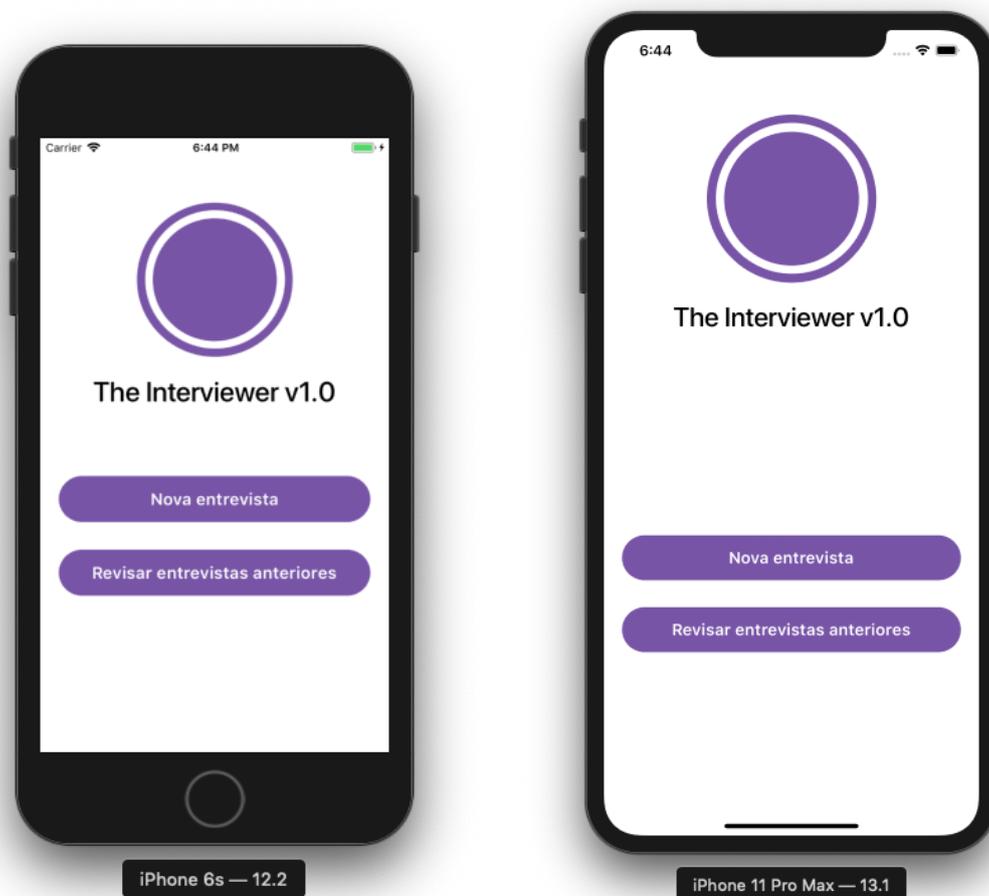
Fonte: O autor, 2019

firmação, a gravação de dados confirma o salvamento, a deleção de entrevistas é interativa e o compartilhamento é dispensado automaticamente ao término da ação. Metáforas são respeitadas, como o *empty state* da lista de entrevistas e as listas de perguntas ou seções. O controle do usuário é total, onde é realmente ele que define o que será armazenado, o que será gravado, e o que será compartilhado, como, quando, e com quem.

A interface do aplicativo foi projetada para rodar em telas mais compactas, como a do *iPhone 6s*, por exemplo, em telas maiores, como a do *iPhone 11 Pro Max* e em telas grandes, como a do *iPad Pro* de 12.9 polegadas. Abaixo, pode-se conferir o exemplo da tela do menu inicial do aplicativo nos tamanhos de tela citados. A partir deste ponto, a tela utilizada nas demais demonstrações de interface será a do *iPhone 11 Pro Max*, por ser uma tela de tamanho médio.

Pode-se conferir os conceitos de interface explicados anteriormente, bem como o design resultante do aplicativo. A integridade estética pode ser percebida na figura 4.6, onde a interface não apresenta elementos visuais distrativos, liberando o foco do usuário para o que realmente interessa no momento. Ainda na figura 4.6, o botão de gravação

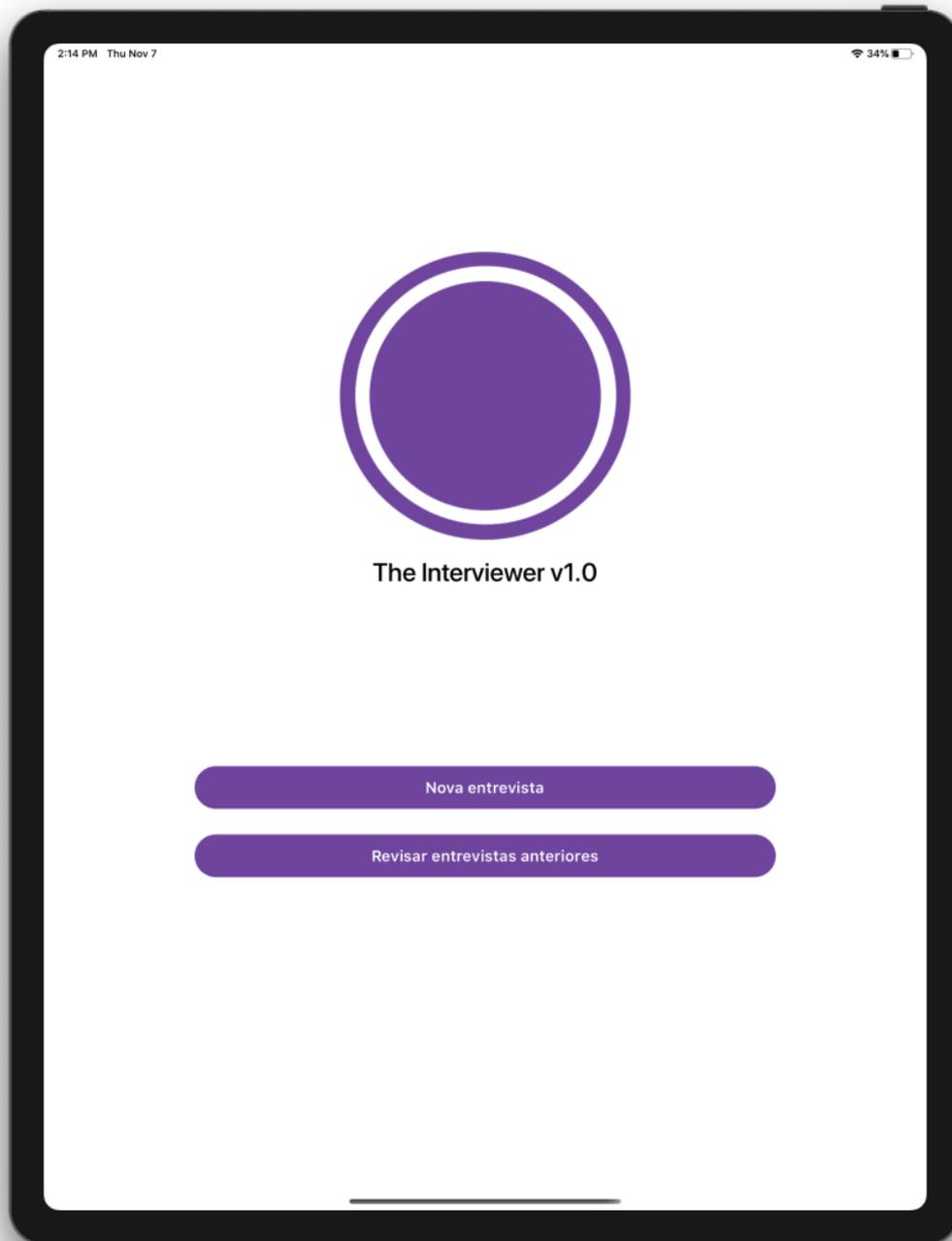
Figura 4.4: Menu principal do aplicativo, *iphones* 6s e 11 Pro Max



Fonte: O autor, 2019

de entrevista, quando habilitado, tem maior enfoque por ser um elemento importante na interface.

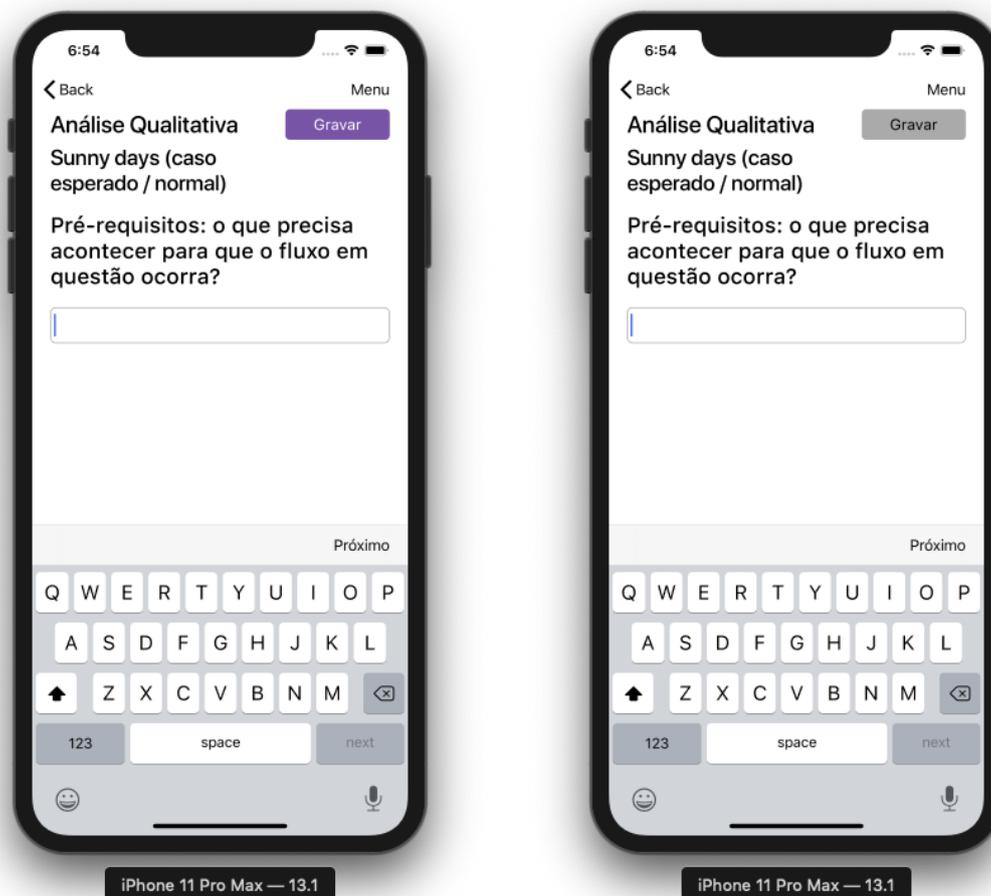
Figura 4.5: Menu principal do aplicativo, *iPad Pro* 12.9 polegadas



iPad Pro (12.9-inch) (3rd generation) — 13.1

Fonte: O autor, 2019

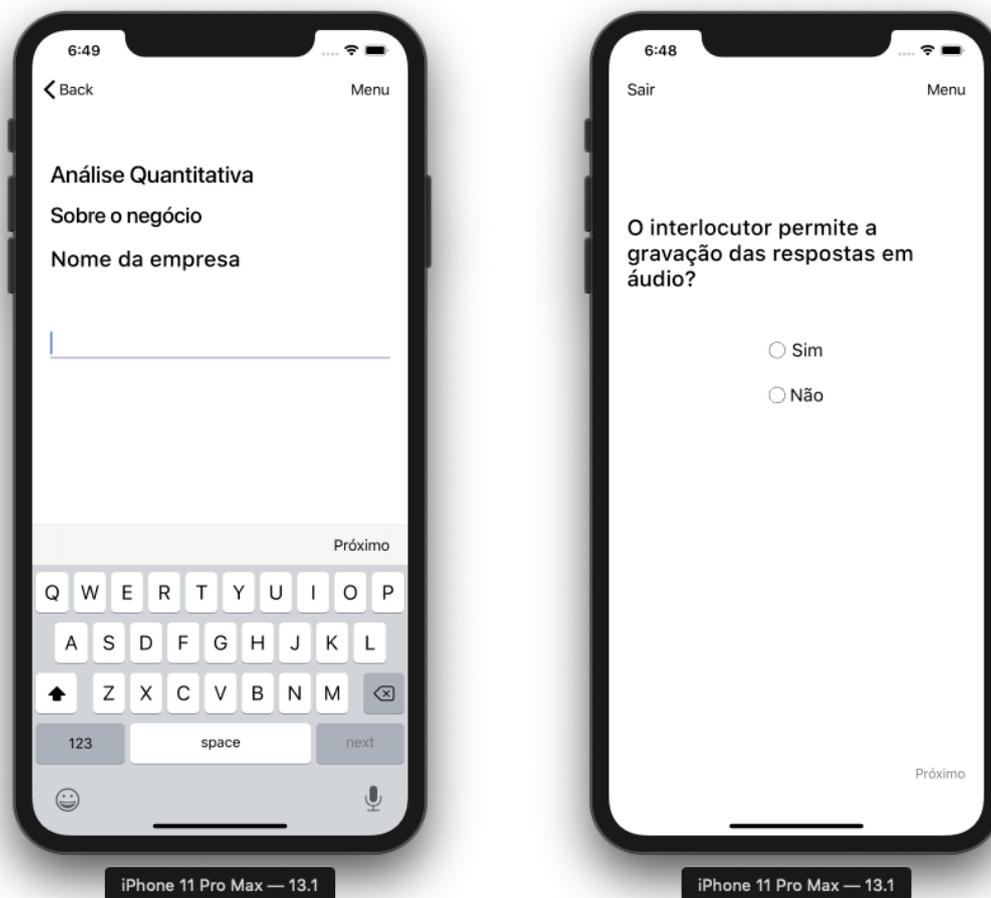
Figura 4.6: Perguntas de resposta longa



Fonte: O autor, 2019

A consistência também pode ser percebida tanto na figura 4.6 quanto na 4.7, onde as utilidades de interface seguem o mesmo posicionamento. O *menu* está sempre acessível no canto superior direito da tela, o botão *próximo*, utilizado para continuar o fluxo, está sempre disponível no canto inferior direito ou nas utilidades do teclado, e as ações *voltar* e *sair* podem sempre ser encontradas no canto superior direito, por possuírem uma semântica similar.

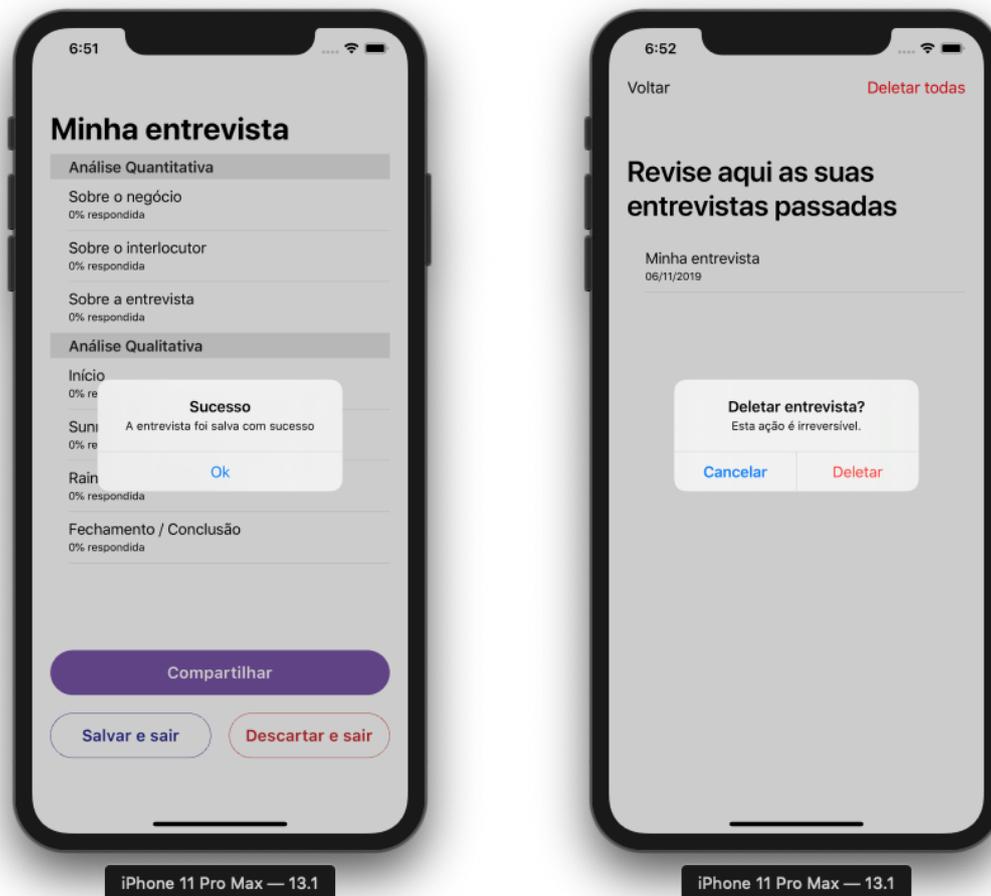
A manipulação direta pode ser depreendida a partir das figuras 4.6 e 4.7, pois os elementos de interface podem ser facilmente acessados e manipulados, bem como a navegação das questões da entrevista. Ainda em relação à navegação, o usuário pode retornar à página anterior do aplicativo utilizando o *swipe* horizontal, a partir da extremidade esquerda da tela. Este tipo de navegação pertence ao design nativo do *iOS* e é recomendado pela Apple.

Figura 4.7: Perguntas de resposta curta e *compliance*

Fonte: O autor, 2019

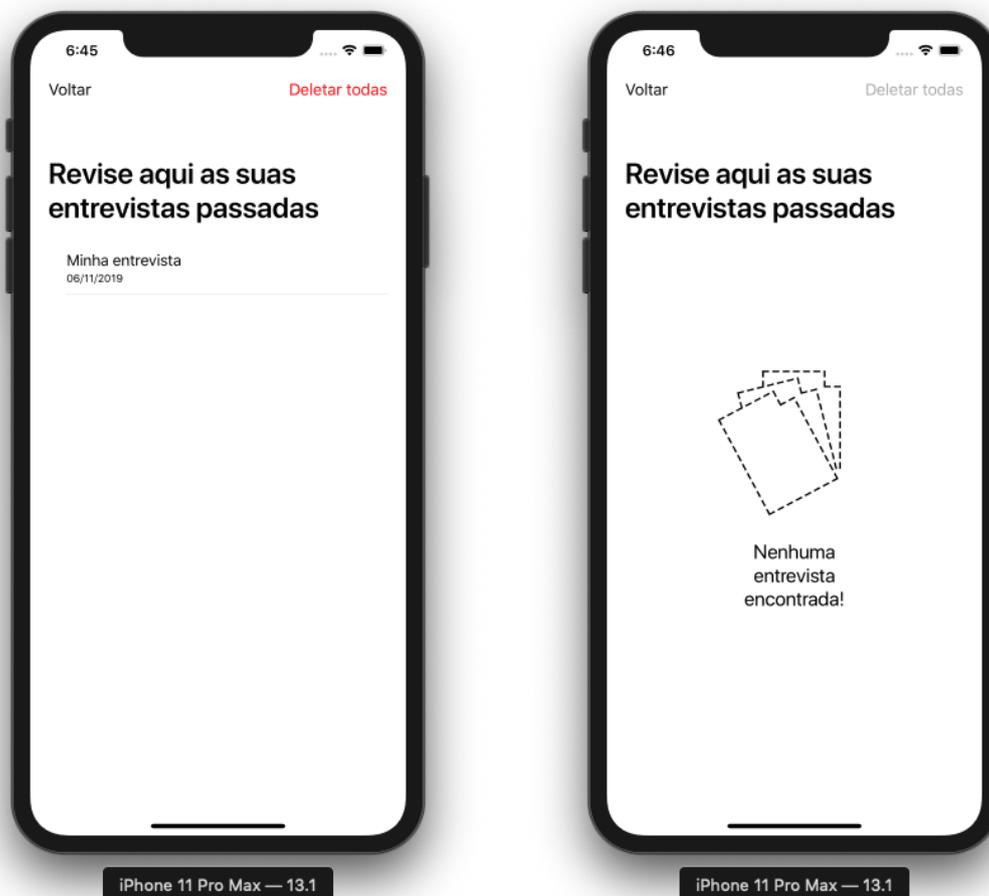
O *feedback* está presente como alertas em diferentes pontos do protótipo. Na figura 4.8, pode-se perceber o alerta apresentado ao terminar uma operação, como na imagem da esquerda, ou quando uma ação destrutiva pode ser tomada, como na imagem da direita. Ao tentar sair durante a entrevista, alertas são apresentados ao usuário, que exigem a confirmação do mesmo, pois o progresso da entrevista pode ser perdido.

Figura 4.8: Aviso de sucesso e alerta de deleção



Fonte: O autor, 2019

As metáforas estão presentes na figura 4.9, onde as entrevistas salvas estão organizadas em forma de lista, e quando não há entrevista salva, o *empty state* de lista vazia é mostrado. Além disso, a lista de seções no *overview* e a lista de questões dentro de uma seção também poderiam ser considerados como metáforas, pois estão dispostos em forma de lista.

Figura 4.9: Revisão de entrevistas e *empty state*

Fonte: O autor, 2019

O usuário no controle é respeitado, pois pode ser verificado na possibilidade de salvar ou deletar a entrevista (figura 4.8), gravar o áudio das respostas ou não (figuras 4.6 e 4.7), e na possibilidade de compartilhar os resultados de uma forma segura (figura 4.10). No *iPhone 11 Pro Max* da figura 4.10, há poucas opções de compartilhamento pelo motivo de que a captura de tela foi feita em um simulador, de forma que não é possível compartilhar dados.

Figura 4.10: Exemplo de compartilhamento



Fonte: O autor, 2019

## 5 AVALIAÇÃO DO PROTÓTIPO

Para a avaliação do protótipo, foram feitos dois tipos de teste. Um teste de usabilidade, para avaliar a usabilidade do protótipo e coletar o *feedback* sobre a sua utilização, e um teste de completude, para avaliar o desempenho da estrutura guiada em uma entrevista.

O teste de usabilidade consiste em um questionário na plataforma *Google Forms*, aplicado após a entrevista, e contém perguntas sobre o conhecimento do *tester* no contexto de BPM, sua familiaridade com tecnologias da Apple, a qualidade das perguntas levantadas, e a usabilidade geral do protótipo. Os objetivos deste teste consistem em coletar informações relevantes sobre o analista, e coletar as opiniões do analista acerca da usabilidade e da interface gráfica do protótipo.

O teste de completude contempla, para cada um dos analistas, a utilização da estrutura guiada em uma entrevista na prática, a modelagem de um processo, seguindo a notação BPMN, a partir dos dados levantados durante a entrevista e coletados pelo protótipo, e a validação da modelagem com o responsável entrevistado pelo analista. Os principais objetivos deste teste consistem em validar a completude da entrevista levantada e avaliar a eficácia do conjunto de perguntas levantadas em uma situação de entrevista.

O processo de negócio escolhido para o teste foi o procedimento de instalação de imagens nos laboratórios, fornecido pela administração da rede do Instituto de Informática da UFRGS. Este processo foi escolhido por apresentar uma complexidade relativamente baixa e por ser pouco extenso, uma vez que o foco do teste não é a modelagem em si, e sim o processo de entrevista.

O teste consiste em três etapas. Primeiro, é realizada a entrevista entre o analista e o entrevistado, utilizando o aplicativo como guia para a entrevista. A seguir, as respostas são compartilhadas via e-mail para o computador do analista, que então inicia a elaboração da modelagem em BPMN. Uma representação gráfica do processo, seguindo a notação BPMN, é importante para obter uma representação objetiva das informações coletadas durante a entrevista. Esta modelagem será utilizada para comparar os resultados obtidos, durante o teste de completude. Finalizada a modelagem, o analista então responde as perguntas do formulário, fornecendo o seu *feedback* sobre a entrevista e a experiência de uso do protótipo. Este roteiro de teste foi realizado com dois analistas, ambos experientes na área de BPM.

## 5.1 Teste de usabilidade

Os resultados do teste de usabilidade consistem no *feedback* coletado pelo formulário, preenchido ao final do roteiro de testes. Este teste pode ser dividido na fase de coleta de informações sobre os analistas, e na fase de coleta de informações sobre o uso do protótipo. As informações sobre os analistas são importantes para situar o contexto dos mesmos dentro da utilização do aplicativo. O *feedback* sobre a usabilidade e a interface gráfica do protótipo são importantes para mensurar a eficácia das técnicas utilizadas no protótipo desenvolvido.

### 5.1.1 Sobre os analistas

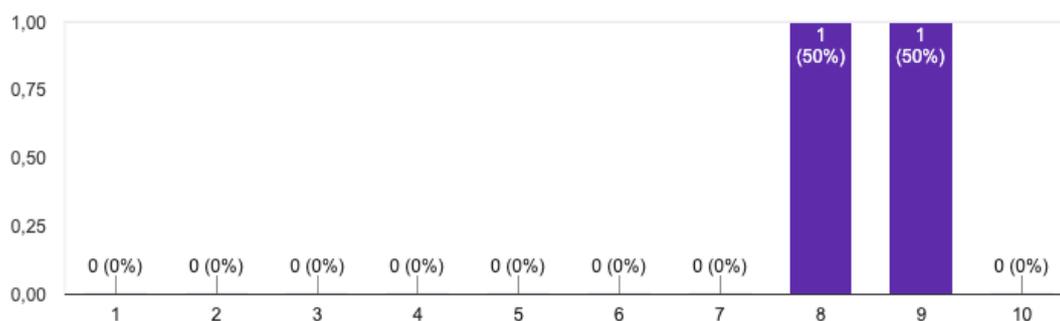
As duas primeiras perguntas do formulário se referem ao conhecimento em BPM. A primeira é sobre o tempo de experiência, a qual ambos os analistas possuem mais de três anos de experiência na área.

A próxima pergunta se refere ao conhecimento em BPM dos analistas, em uma escala de 0 a 10 (figura 5.1). As respostas obtidas foram 8 e 9. Assim, por ambos possuírem mais de três anos de experiência e um conhecimento supostamente avançado, eles podem ser considerados experientes no assunto.

Figura 5.1: Conhecimento em BPM

Como você avaliaria seu conhecimento em BPM?

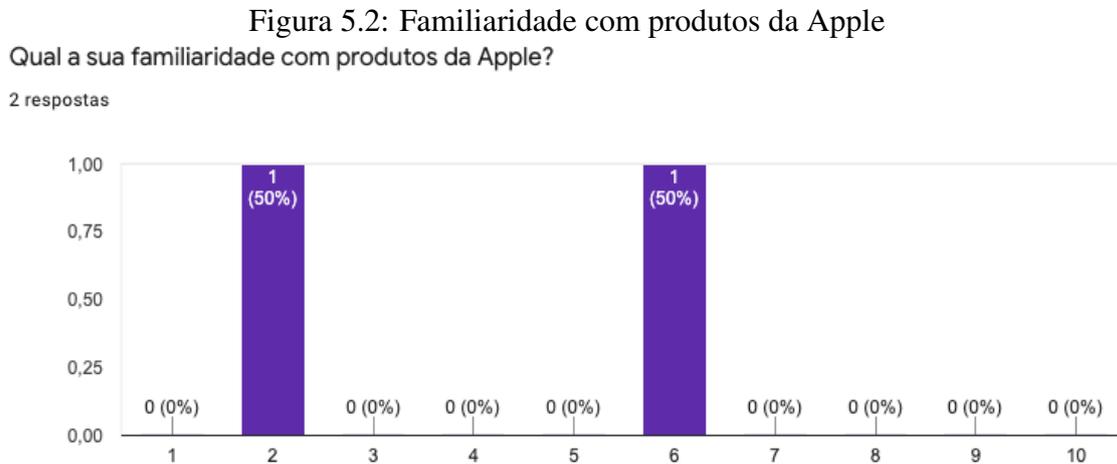
2 respostas



Fonte: O autor, 2019

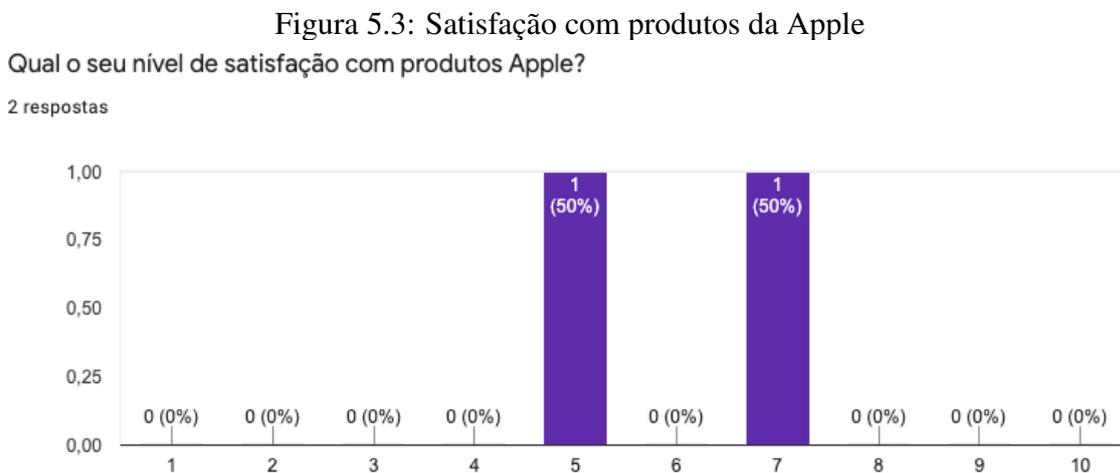
As próximas perguntas (figuras 5.2 e 5.3) buscam esclarecer a familiaridade dos analistas no contexto do dispositivo móvel. Quanto mais familiarizado o usuário estiver, mais acostumado ele estará com o funcionamento das interfaces da Apple. A pergunta

"Qual a sua familiaridade com produtos da Apple?" espera uma resposta entre 0 e 10. As respostas obtidas, 2 e 6, podem ser consideradas discrepantes (figura 5.2).



Fonte: O autor, 2019

A respeito da satisfação com produtos da Apple (figura 5.3), não houve um forte sentimento a respeito, e portanto pode-se considerar que esse aspecto não terá muito impacto no *feedback*. Assim sendo, pode-se afirmar, sobre o perfil dos analistas, que ambos são experientes no contexto de BPM, sendo um inexperiente com produtos da Apple e o outro possuindo alguma experiência. Nenhum dos dois possui uma opinião forte sobre satisfação ou insatisfação com a plataforma Apple. Este ponto pode conferir ao teste uma abordagem mais imparcial, por parte dos analistas, a respeito do sistema operacional escolhido.



Fonte: O autor, 2019

### 5.1.2 Feedback dos analistas

Nenhum dos analistas possuem *iPhone* ou *iPad*. Por este motivo, o protótipo foi utilizado a partir do *iPhone 6s* de um dos autores deste trabalho, de forma que as partes do teste que requeriam a utilização do protótipo precisaram ser sequenciais. Assim, a revisão dos resultados obtidos da entrevista foi feita unicamente a partir dos dados compartilhados para cada um dos analistas, desconsiderando o fluxo de entrevista do protótipo. Portanto, os campos chamados *revisão* a seguir referem-se apenas à revisão feita com os dados compartilhados, e não propriamente ao fluxo de revisão implementado no protótipo. Quando perguntados sobre a interface do aplicativo, os analistas forneceram a sua avaliação sobre diversos aspectos de interface e usabilidade que eles presenciaram durante a entrevista (tabela 5.1).

Tabela 5.1: Avaliação do sistema - Feedback dos analistas

<b>1. Sobre a interface do aplicativo, avalie a sua experiência com os seguintes fluxos</b>	<b>Respostas</b>
Criação de entrevista	Razoável (1) e Boa (1)
Revisão de entrevista	Péssima (1) e Boa (1)
Compartilhamento da entrevista	Boa (2)
<b>2. Para a criação e revisão de entrevista, qual foi a sua percepção a respeito dos seguintes aspectos</b>	<b>Respostas</b>
Facilidade de uso	Boa (2)
Facilidade de entender a interface	Razoável (1) e Excelente (1)
Facilidade de preencher as respostas	Ruim (1) e Excelente (1)
Clareza de saber o que está acontecendo	Razoável (1) e Boa (1)
Clareza para entender o passo a passo	Boa (2)
Facilidade de compartilhar os resultados	Boa (2)

Fonte: O Autor

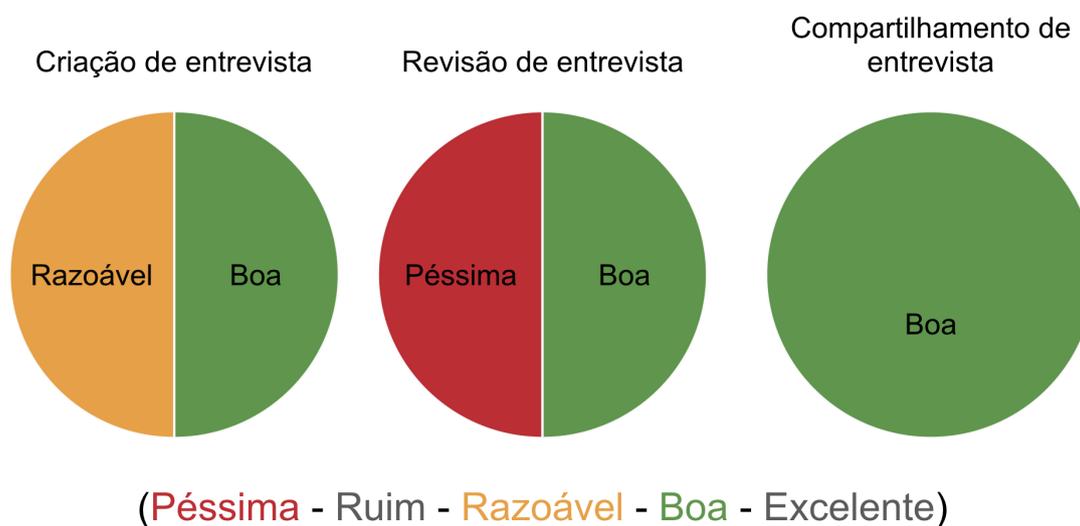
As perguntas contidas na tabela 5.2 referem-se a alguns aspectos de interface do aplicativo. A pergunta 1 da tabela 5.2 também está representada na figura 5.4, oferecendo uma visualização alternativa das respostas da tabela. As possíveis respostas são *péssima*, *ruim*, *regular*, *boa* e *excelente*.

A respeito dos resultados obtidos nesta avaliação, referentes à questão 1 da tabela 5.2, pode-se dizer que o aplicativo foi bem sucedido quanto à criação e o compartilhamento da entrevista. O fluxo de revisão da entrevista, como explicado antes, não foi testado, e obteve esta avaliação devido ao fato de os artefatos compartilhados com os analistas não apresentarem boa usabilidade. Para melhorar a usabilidade da revisão baseada nos artefatos compartilhados, algumas melhorias são sugeridas no presente trabalho.

Os resultados obtidos pela questão 2 da tabela 5.2 indicam boa facilidade de uso,

interface clara e fácil de entender, fluxo claro e boa facilidade para compartilhar os resultados. O preenchimento das respostas obteve uma avaliação discrepante porque um dos analistas não considerou satisfatória a utilização de *smartphones* para o registro das respostas textuais, enquanto o outro analista não encontrou problemas neste ponto.

Figura 5.4: Avaliação da interface



Fonte: O autor, 2019

A tabela 5.2, encontrada a seguir, traz alguns pontos levantados pelos usuários. A primeira questão se refere aos pontos de melhoria, e a segunda se refere às opiniões levantadas pelos mesmos.

Alguns pontos da primeira questão serão justificados. A interface pequena advém do aparelho pequeno que foi utilizado para testes, um *iPhone 6s*. Houve, de fato, uma falha no design ao situar o botão *gravação* próximo ao botão *menu*, visto que a gravação é utilizada bastante durante a entrevista, e a proximidade dos dois botões realmente atrapalhou o uso. Uma solução para este problema seria centralizar o botão de gravação na borda inferior da tela, separando-o dos outros botões.

As instruções foram propositalmente vagas, para simular uma entrevista real. A gravação de áudio foi projetada para ser a funcionalidade de captação de respostas longas primária. A digitação de textos ajuda a anotar possíveis detalhes durante a gravação, ou como entrada principal caso não seja permitida a gravação de voz.

A preferência por um arquivo JSON não foi a melhor opção, visto que é mais apropriado para APIs e processamento automatizado. A melhor opção neste aspecto talvez fosse um arquivo PDF, disponibilizado juntamente com o arquivo JSON. No que diz respeito aos áudios, seria uma boa ideia renomear os arquivos para conter o enunciado das

perguntas no título, dependendo do tamanho do mesmo, ou referenciar o nome do arquivo de áudio correspondente nas respostas das perguntas, dentro do JSON ou do arquivo PDF sugerido.

Tabela 5.2: Parecer dos analistas

<b>1. Pontos de melhoria levantados pelos usuários</b>
Interface pequena, foi fácil tocar no botão errado.
Instruções estavam vagas. O usuário não sabia que era possível pular perguntas, por exemplo.
Escrever texto em um <i>smartphone</i> é inconveniente. O usuário deu preferência à utilização de gravações.
Arquivo JSON com as respostas escritas não possuem quebra de linha para cada resposta. Foi difícil encontrar a resposta desejada.
Os arquivos das gravações de áudio não apresentaram indicativo claro de qual foi a pergunta realizada. É preciso escutar o áudio.
E-mail enviado com as respostas não possui um texto explicativo. É necessário localizar e baixar os arquivos em anexo.
<b>2. Opiniões levantadas no parecer dos usuários</b>
O aplicativo ajuda a organizar um pouco a entrevista, mas não apresentou muito potencial original.
Utilizar <i>smartphones</i> é bastante limitante. Com organização prévia, poderia-se fazer a mesma coisa com um aplicativo de gravação e um caderno de anotações.
O protocolo de entrevista pareceu muito granular. Várias respostas foram captadas antes da pergunta ser feita.
A revisão dos resultados está bastante deficiente.
Aplicativo de fácil uso. Interface objetiva, disponibilizando o necessário para realização da entrevista.

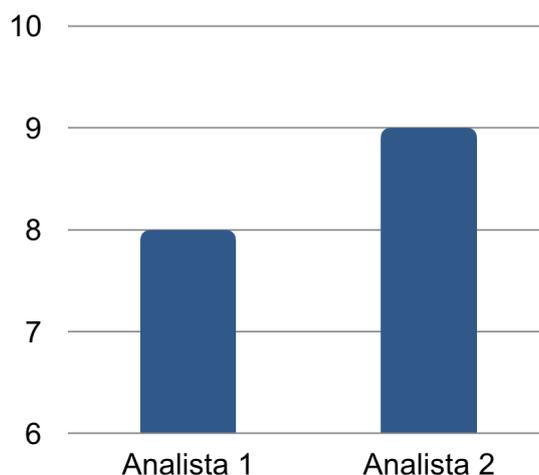
Fonte: O Autor

A figura 5.5 se refere à cobertura que as perguntas alcançam sobre os aspectos da entrevista. Ambos acharam o alcance relativamente alto, o que aponta um possível sucesso na estratégia de estruturação da entrevista, apresentada e discutida anteriormente. A avaliação da cobertura indica a percepção do analista a respeito dos pontos importantes do processo que foram cobertos durante a entrevista.

Quanto à utilidade do aplicativo (tabela 5.3), um dos analistas gostou bastante e utilizaria apenas o aplicativo durante as entrevistas. O outro analista achou que as gravações foram a funcionalidade de maior utilidade do aplicativo. Ele gostou dos momentos propostos para a sumarização dos *sunny days* e *rainy days*, e que seria bom para analistas novatos na área. De acordo com o seu *feedback*, o protocolo de entrevista não foi muito útil por ser muito granular. Ele acredita que um conjunto de tópicos seria mais útil do que levantar as questões prontas para a leitura, pois assim ele poderia interpretar e adaptar a pergunta ao tópico desejado.

A entrevista guiada pode ser útil para ambos os tipos de usuário, sejam eles inex-

Figura 5.5: Cobertura das perguntas da estrutura  
Cobertura das perguntas



Fonte: O autor, 2019

Tabela 5.3: Utilidade do aplicativo

O uso do aplicativo ajudou?
Ajudou, utilizaria somente o aplicativo.
A maior ajuda foi as gravações. O protocolo de entrevista parece não ser muito útil.
Foi bom ter momentos para sumarizar os <i>sunny days</i> e os <i>rainy days</i> . Seria bom para novatos lembrar dos casos de exceção.

Fonte: O Autor

perientes ou experientes. Para os usuários inexperientes, a entrevista guiada serve como um guia para auxiliar no passo a passo da realização das entrevistas. Como usuários inexperientes têm maior tendência a esquecer os casos de exceção (DUMAS et al., 2013), a entrevista guiada garante uma abordagem detalhada para estes casos.

Para os usuários experientes, o uso da entrevista guiada pode auxiliar a não deixar aspectos que possam vir a ser relevantes passarem despercebidos. A estrutura guiada pode auxiliá-los a verificar que os pontos importantes da entrevista foram abordados, bem como averiguar se há algum ponto importante que não está sendo abordado (PRESSMAN, 2005).

Pode-se afirmar que a preferência de abordagem de cada analista teve impacto na sua impressão final sobre a estrutura da entrevista. Disponibilizar uma estrutura com tópicos auxiliaria abordagens onde o analista tem completo controle da situação e prefere elaborar as perguntas ao seu modo, mas a cobertura do processo poderia ser comprometida.

## 5.2 Teste de completude

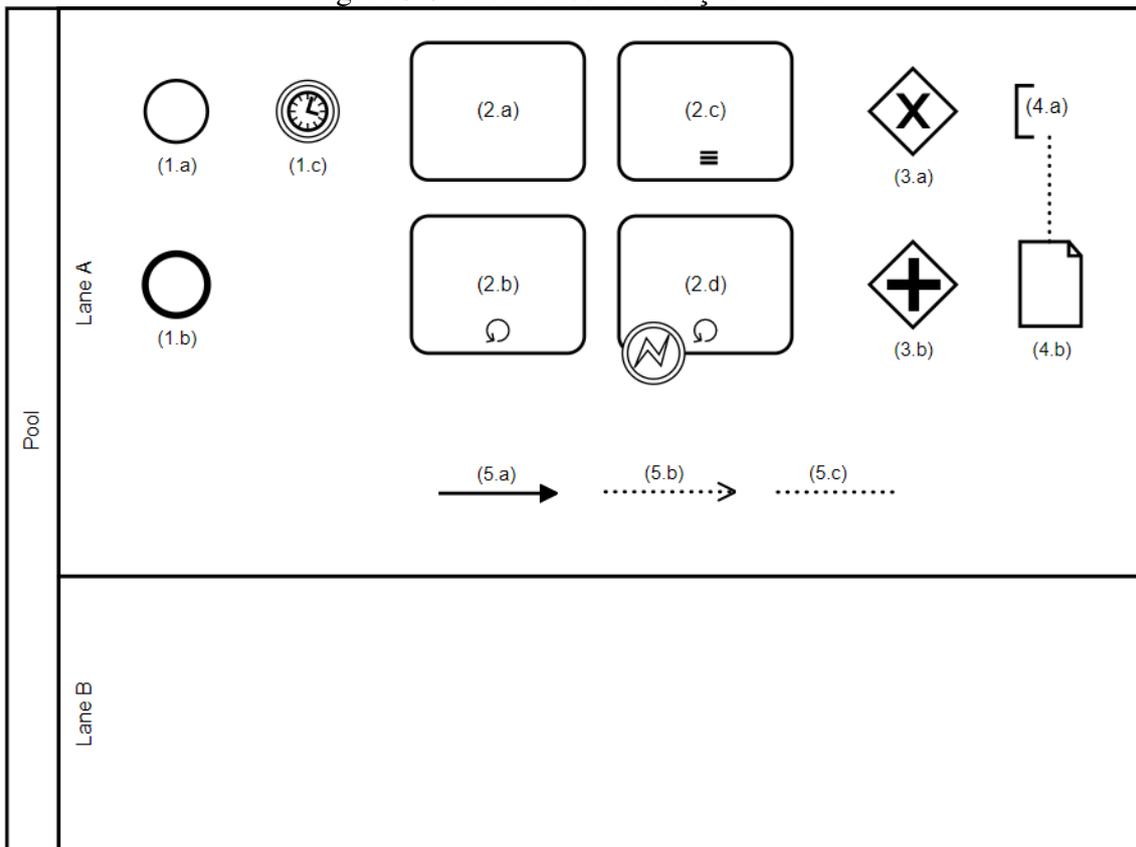
O teste de completude se baseia na modelagem feita por cada um dos analistas, a partir dos resultados do uso do aplicativo durante a entrevista. Nas figuras 5.7 e 5.8, pode-se conferir uma visão geral de como ficou cada modelagem, mas o tamanho da imagem provavelmente não permitirá a leitura das atividades. Para uma visão mais detalhada, confira as modelagens em tamanho real, nos apêndices A e B deste trabalho.

As modelagens geradas pelos analistas (figuras 5.7 e 5.8) seguem a notação BPMN. Para um melhor entendimento das modelagens, os elementos necessários da notação BPMN serão explicados, com base na figura 5.6. O processo é contido em uma piscina, na figura, chamada de *pool*. A piscina é dividida em raias, conhecidas como *lanes*, representando as ações de cada um dos atores envolvidos no processo. Neste caso, há dois atores envolvidos, o professor e a equipe de TI (ou *admRede*).

Seguindo os elementos da tabela 5.6, temos os seguintes itens. Eventos são representados por círculos. O elemento *1.a* indica o início do processo, e o elemento *1.b* indica seu fim. O *1.c* representa um evento *timer*, que acontece após um determinado período de tempo. O item *2.a* indica uma atividade granular que ocorre dentro do processo, enquanto os itens *2.b*, *2.c* e *2.d* representam variações da atividade. *2.b* indica que pode ser necessário repetir a atividade uma ou mais vezes. *2.c* representa uma atividade que pode possuir múltiplas instâncias sequenciais, tendo a sua execução repetida por um determinado número de vezes. *2.d* indica que a atividade pode lançar exceções durante a sua execução.

Os itens *3.a* e *3.b* são chamados *gateways*, e representam possíveis desvios no fluxo de execução do fluxo do processo. O item *3.a* representa um desvio exclusivo, onde o fluxo deve tomar apenas um caminho. O item *3.b* representa um desvio paralelo, onde ambos os fluxos devem ser executados. Os itens *4.a* e *4.b* são conhecidos como artefatos. O elemento *4.a* constitui em uma anotação, e seu propósito é auxiliar no entendimento do fluxo, fornecendo informações relevantes ao contexto. O elemento *4.b* representa um objeto de dados. Pode constituir um arquivo digital, um documento físico, entre outros. Os elementos *5.a*, *5.b* e *5.c* representam, respectivamente, o caminho usual do fluxo, um envio de mensagem ou documento, e uma referência de anotações.

Figura 5.6: Elementos da notação BPMN

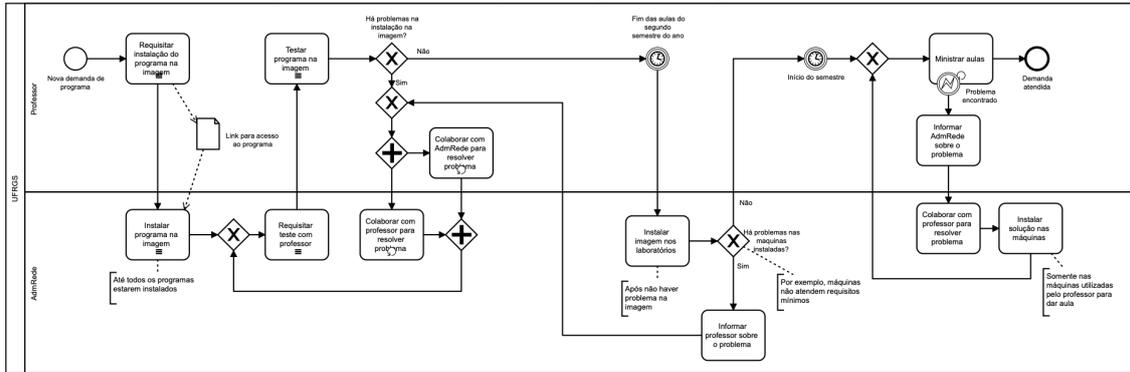


Fonte: O autor, 2019

Como pode-se perceber comparando os diagramas (figuras 5.7 e 5.8), há uma diferença de complexidade e detalhamento entre as duas modelagens. Estas diferenças de detalhamento se devem, em parte, pela forma como foram conduzidas as entrevistas, pelos analistas.

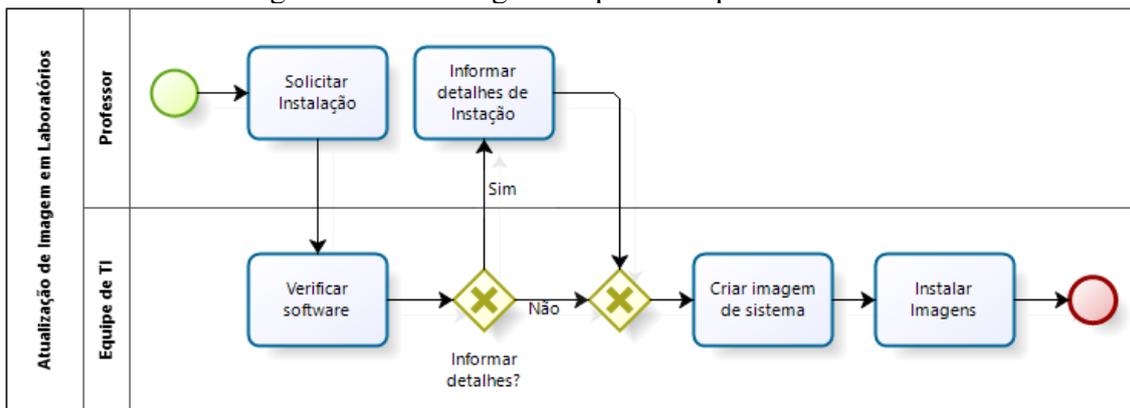
Cada uma das modelagens foi verificada com o especialista do domínio, que executa o processo modelado. De acordo com o especialista, a primeira modelagem (figura 5.7) possui um dos casos de exceção duplicado, onde o professor e a equipe de TI colaboraram para resolver o problema encontrado em apenas um momento. A segunda modelagem (figura 5.8) omite um dos casos de exceção, onde problemas na instalação da imagem podem ser encontrados. Conforme o especialista, apesar destas observações, ambas as modelagens representam o processo descrito e poderiam ser consideradas aprovadas.

Figura 5.7: Modelagem do processo pelo analista 1



Fonte: Analista 1, 2019

Figura 5.8: Modelagem do processo pelo analista 2



Fonte: Analista 2, 2019

## 6 CONCLUSÃO

Neste trabalho foi proposta uma estrutura para entrevista guiada, a partir de informações extraídas da literatura. A estrutura guiada busca cobrir os aspectos relevantes de uma entrevista, para possibilitar uma modelagem do processo de negócio o mais completa possível, buscando ser útil tanto para o analista mais inexperiente quanto para o mais experiente.

Este trabalho também cobre os requisitos funcionais, definições técnicas e a implementação de um protótipo, executável no ambiente *iOS*, que segue a estrutura de entrevista guiada proposta. O código-fonte do protótipo pode ser acessado na plataforma de versionamento *Github*<sup>1</sup> (LAGEMANN, 2019). Os requisitos funcionais do protótipo contemplam os fluxos de entrevista e revisão, e o compartilhamento dos resultados obtidos durante a entrevista. As definições técnicas incluem requisitos de implementação, como o sistema operacional escolhido, a linguagem de programação na qual o protótipo foi implementado, as tecnologias utilizadas na implementação e detalhes de arquitetura.

As contribuições deste trabalho incluem a estrutura guiada proposta, que pode ser encontrada nos apêndices C e D, e o protótipo implementado, que está disponível<sup>2</sup>, de forma temporária, através da plataforma *TestFlight* (TESTFLIGHT, 2019). A estrutura guiada independe do protótipo, podendo ser utilizada também de forma manual ou através de outros meios, de acordo com a preferência do analista. O protótipo, contudo, limita-se apenas à estrutura proposta, não possibilitando a utilização de outra estrutura. De acordo com o que foi proposto, o protótipo está completo e oferece o correto funcionamento das funcionalidades apresentadas.

A partir dos resultados obtidos, pode-se perceber que o analista 1 buscou um maior grau de detalhamento na modelagem (figura 5.7) em relação ao analista 2 (figura 5.8). Não apenas isso, mas o modo como as entrevistas foram conduzidas foi diferente, entre os dois. O analista 1 lia a pergunta para si mesmo, a interpretava, e então perguntava ao entrevistado a sua interpretação da pergunta, ao passo em que o analista 2 apenas lia a pergunta para o entrevistado. Essa diferença em relação à forma de como a entrevista foi feita com certeza teve um impacto no resultado final, demonstrado tanto pelo *feedback* individual dos analistas quanto pela modelagem resultante das duas entrevistas.

Uma conclusão que se pode tirar após uma análise dos resultados é de que, apesar de ambos os analistas serem experientes em BPM, ambos consideraram que a cobertura

---

<sup>1</sup>código-fonte disponível em <https://github.com/athoslag/TheInterviewer>

<sup>2</sup>versão testável disponível em <https://testflight.apple.com/join/nX4uoSZ7>

das perguntas feitas na entrevista foi bastante ampla. Quanto ao modo de condução da entrevista, o analista que interpretou as perguntas antes de fazê-las terminou com uma modelagem mais complexa e detalhada, porém encontrou perguntas redundantes, tendo a impressão de se repetir em alguns momentos. O analista que seguiu a entrevista no modo mais literal não teve problema com redundância de questões, mas terminou em uma modelagem menos detalhada. Assim, a partir dos resultados obtidos, pode-se afirmar que a estrutura guiada funcionou muito bem para modelagens mais simplificadas. Talvez o mais beneficiado desta estrutura seja o analista mais literal, que busca uma modelagem mais pontual.

Como foi indicado no *feedback*, a separação da entrevista do fluxo em *sunny days* e *rainy days* foi útil durante a entrevista e, de acordo com um dos analistas, também seria útil para novatos, que apresentam uma maior tendência a esquecer os *rainy days*. Como a cobertura das perguntas foi considerada alta, podemos dizer que a estrutura da entrevista foi, em linhas gerais, satisfatória. Visto que ambas as modelagens foram aprovadas pelo especialista do domínio, o teste de completude se mostrou satisfatório. Portanto, pode-se afirmar que a estrutura de entrevista guiada proposta funcionou para o propósito de conduzir a entrevista e possibilitar a sua posterior modelagem.

Acreditamos que o aplicativo tenha atingido parcialmente os seus objetivos. Para analistas menos experientes, há indícios de que a estrutura proposta será apropriada para a modelagem e servirá como um guia para a realização da entrevista com alta cobertura do processo. Para analistas mais experientes no contexto, a estrutura auxilia a captação das respostas e os ajuda a lembrar de cobrir pontos que poderiam passar despercebidos.

Por se basear em um misto entre a literatura de modelagem de processos e engenharia de *software*, a cobertura da parte referente às exceções que podem ocorrer no processo foi importante para a modelagem testada. A estrutura poderia ser abstraída para o analista que busca um maior controle sobre a entrevista, separando-a em tópicos a serem cobertos. Para este caso, atualmente, apesar de apresentar uma boa cobertura do processo, as perguntas acabam se tornando eventualmente redundantes, visto que o analista que apresenta uma interpretação das perguntas pode cobrir de antemão pontos que seriam cobertos posteriormente.

## 6.1 Limitações

As limitações encontradas na estrutura da entrevista guiada situam-se no fato de que as perguntas, no modo como foram elaboradas, prendem o analista à estrutura proposta. Nestes casos, ao buscar interpretar as questões, o analista pode obter uma entrevista redundante e repetitiva, pois as mesmas podem englobar mais do que a sua proposição inicial. Como analistas experientes podem preferir manter o controle sobre as perguntas da entrevista, nestes casos uma modelagem voltada a tópicos talvez fosse mais eficiente.

Sobre a interface do protótipo, algumas mudanças serão necessárias. O compartilhamento da transcrição da entrevista em um arquivo JSON não se mostrou uma opção satisfatória, dado que o analista teve dificuldades em encontrar as informações dentro do arquivo. Como abordado no Capítulo 5, Subseção 5.1.2, a vantagem do compartilhamento em JSON é uma maior facilidade para posterior automatização dos resultados. Uma melhor opção para a revisão da transcrição seria o compartilhamento da entrevista em um formato PDF, com as perguntas e respectivas respostas, mas ainda mantendo a opção de gerar o JSON para os eventuais casos de automatização desejada.

## 6.2 Perspectivas de trabalho futuro

A partir da observação dos testes e do *feedback* recebido pelos analistas, será necessário efetuar algumas mudanças tanto na interface quanto no compartilhamento dos resultados do protótipo. Algumas perguntas possuem respostas em formato binário como, por exemplo, sim e não. Uma opção de resposta binária seria conveniente neste caso, pois o analista não precisaria digitar toda a palavra, podendo apenas escolher a opção. Como não ficou claro para os usuários que as perguntas não são obrigatórias, seria também interessante alterar o texto do botão *próximo*, nas telas de respostas, para uma forma dinâmica, que mostre *pular* enquanto não houver resposta registrada.

Um maior distanciamento dos botões *menu* e *gravar* melhoraria a experiência do usuário. Um exemplo seria mover o botão de gravação para o canto inferior da tela, centralizado ou à esquerda. Também a respeito da interface, seria interessante colocar uma barra de progresso acima da pergunta, para que o analista saiba o quantas perguntas faltam para terminar a seção atual.

Sobre os itens compartilhados, a adição de um arquivo PDF melhoraria a experiência de revisão. O PDF poderia conter a pergunta, a resposta e uma referência ao

nome do arquivo de áudio relacionado. Além disso, a adição de uma mensagem padrão no corpo do meio de compartilhamento traria mais contexto da aplicação ao compartilhar os resultados, diminuindo o esforço necessário para revisar algo a partir dos artefatos gerados.

Melhorias nos elementos gráficos do protótipo também trariam uma maior usabilidade. Por exemplo, o botão de gravação mostrar o tempo decorrido desde o início da gravação, indicadores animados de que a gravação está em andamento e, para o fluxo de revisão, o *replay* da gravação mostrar o tempo de áudio que foi gravado e o tempo de reprodução atual do áudio.

Para os casos em que o analista não dispõe de um *iPhone* ou *iPad*, seria necessário utilizar a estrutura da entrevista guiada manualmente. O protótipo pode ser implementado na plataforma *Android* utilizando-se o mesmo banco de dados local, pois a base de dados *Realm* mantém a mesma estrutura do *Modelo* da aplicação e é compatível com ambas as plataformas. A maior dificuldade estaria, portanto, na implementação do controle de fluxos e apresentação da interface gráfica do protótipo.

## REFERÊNCIAS

- APPLE. **Model-View-Controller**. 2012. Available from Internet: <<https://developer.apple.com/library/archive/documentation/General/Conceptual/CocoaEncyclopedia/Model-View-Controller/Model-View-Controller.html>>.
- APPLE. **Beta Testing Made Simple with TestFlight**. 2014. Available from Internet: <<https://developer.apple.com/testflight/>>.
- APPLE. **Swift documentation**. 2018. Available from Internet: <<https://swift.org/documentation/>>.
- APPLE. **Apple Developer Support**. 2019. Available from Internet: <<https://developer.apple.com/support/app-store/>>.
- APPLE. **Human Interface Guidelines**. 2019. Available from Internet: <<https://developer.apple.com/design/human-interface-guidelines/ios/overview/interface-essentials/>>.
- COCOAPODS. **About CocoaPods**. 2019. Available from Internet: <<https://cocoapods.org/about>>.
- DUMAS, M. et al. **Fundamentals of business process management**. [S.l.]: Springer, 2013.
- GOOGLE. **Android Distribution dashboard**. 2019. Available from Internet: <<https://developer.android.com/about/dashboards/index.html>>.
- GOSSMAN, J. **Introduction to Model/View/ViewModel pattern for building WPF apps**. 2005. Available from Internet: <<https://blogs.msdn.microsoft.com/johngossman/2005/10/08/introduction-to-modelviewviewmodel-pattern-for-building-wpf-apps/>>.
- HALLIGON, O. **Reusable**. 2019. Available from Internet: <<https://cocoapods.org/pods/Reusable>>.
- IDC. **Smartphone Market Share**. 2019. Available from Internet: <<https://www.idc.com/promo/smartphone-market-share/os>>.
- INTERSOG, L. **Interview Assistant**. 2011. Available from Internet: <<https://apps.apple.com/us/app/interview-assistant/id475659902>>.
- KHANLOU, S. **The Coordinator**. 2015. Available from Internet: <<http://khanlou.com/2015/01/the-coordinator/>>.
- KHANLOU, S. **Coordinators Redux**. 2015. Available from Internet: <<http://khanlou.com/2015/10/coordinators-redux/>>.
- LAGEMANN, A. **The Interviewer**. 2019. Available from Internet: <<https://github.com/athoslag/TheInterviewer>>.
- LANGH GERGELY OROSZ, R. L. D. **SkyFloatingLabelTextField**. 2019. Available from Internet: <<https://cocoapods.org/pods/SkyFloatingLabelTextField>>.

MARMELSTEIN, R. **Zip**. 2017. Available from Internet: <<https://cocoapods.org/pods/Zip>>.

OMG. **BPMN Specification - Business Process Model and Notation**. 2019. Available from Internet: <<https://www.bpmn.org>>.

PMI, A. guide to the project management body of knowledge (pmbok guide). In: **Project Management Institute**. [S.l.: s.n.], 2013. v. 5.

PRESSMAN, R. S. **Software engineering: a practitioner's approach**. [S.l.]: Palgrave Macmillan, 2005.

REALM. **RealmSwift**. 2019. Available from Internet: <<https://cocoapods.org/pods/RealmSwift>>.

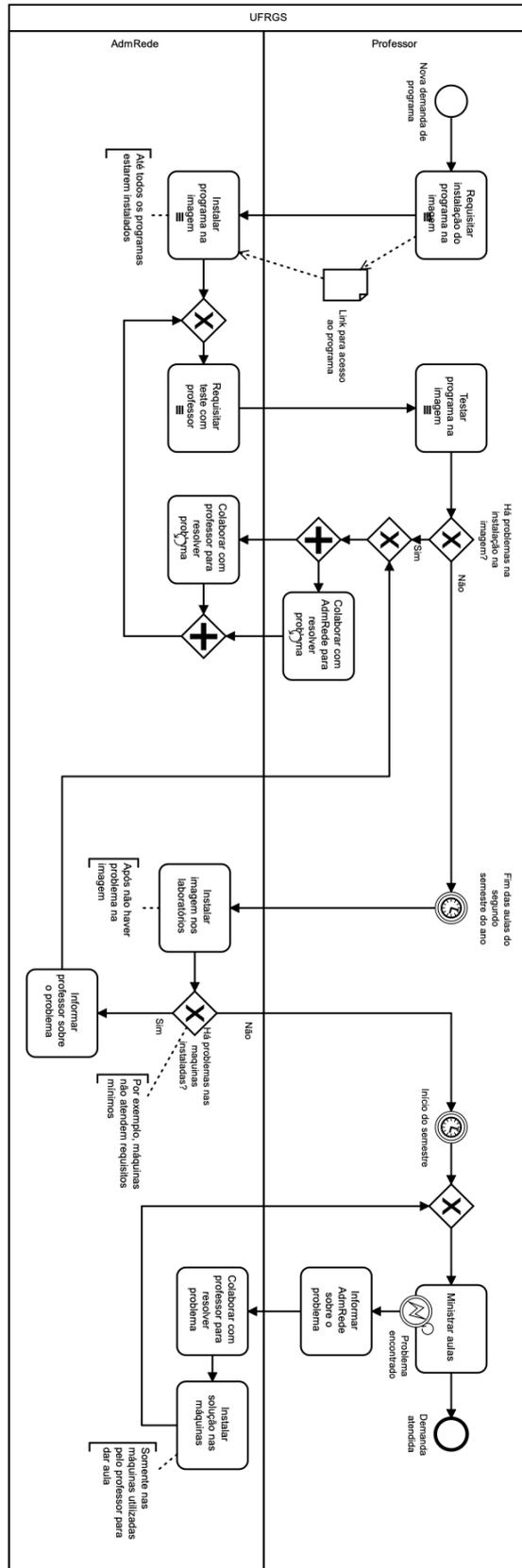
SHARP, A.; MCDERMOTT, P. **Workflow modeling: tools for process improvement and applications development**. [S.l.]: Artech House, 2009.

SOMMERVILLE, I. **Software engineering**. [S.l.]: Addison-Wesley/Pearson, 2011.

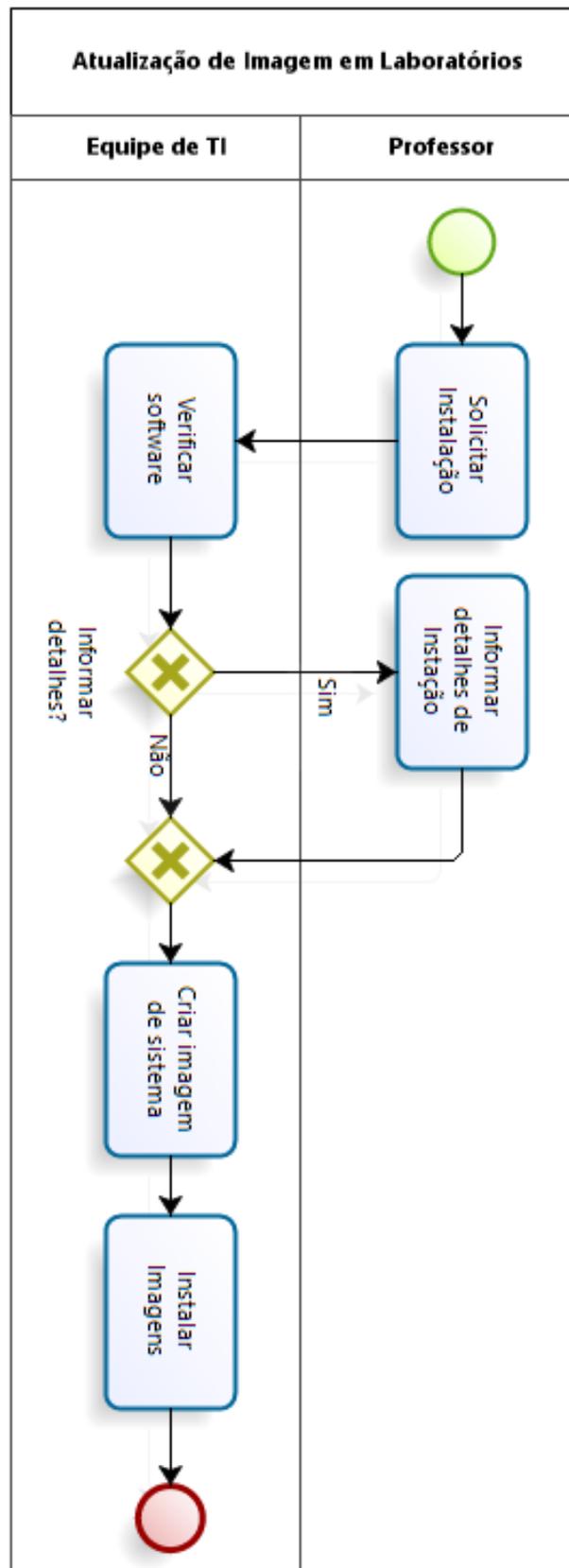
TECH, T. **Interview Assistant**. 2018. Available from Internet: <<https://play.google.com/store/apps/details?id=com.ekb.presentation.presentation>>.

TESTFLIGHT. **The Interviewer**. 2019. Available from Internet: <<https://testflight.apple.com/join/nX4uoSZ7>>.

# APÊNDICE A — MODELAGEM DO ANALISTA 1



## APÊNDICE B — MODELAGEM DO ANALISTA 2



## APÊNDICE C — ESTRUTURA DE ENTREVISTA GUIADA PROPOSTA

Quadro C.1: Perguntas quantitativas

<i>Assunto / Pergunta</i>	<i>Formato da resposta</i>
<b>1. Análise quantitativa</b>	
<b>1.1. Sobre o negócio em questão</b>	
1.1.1. Nome da empresa	Texto curto
1.1.2. Área de atuação	Texto curto
1.1.3. Quantidade de funcionários	Número / Texto curto
1.1.4. Contato	Texto curto
1.1.5. Observações	Texto longo / voz
<b>1.2. Sobre o interlocutor</b>	
1.2.1. Nome do entrevistado	Texto curto
1.2.2. Cargo do entrevistado	Texto curto
1.2.3. Formação do entrevistado	Texto curto
1.2.4. Tempo de empresa (meses / ano)	Número / Texto curto
1.2.5. Contato do entrevistado	Texto curto
<b>1.3. Sobre a entrevista</b>	
1.3.1. Qual ou quais fluxos serão cobertos nesta entrevista?	Texto longo / voz
1.3.2. Qual a relação do entrevistado com cada fluxo?	Texto longo / voz

Fonte: O Autor

Quadro C.2: Perguntas qualitativas

<i>Assunto / Pergunta</i>	<i>Fonte da pergunta</i>
<b>2. Análise qualitativa</b>	
<b>2.1. Início</b>	
2.1.1. Você é a pessoa correta para esclarecer o fluxo que será analisado?	(PRESSMAN, 2005)
2.1.2. As suas respostas são <i>oficiais</i> ?	(PRESSMAN, 2005)
2.1.3. Qual é o objetivo deste fluxo?	(DUMAS et al., 2013)
2.1.4. Quem são os atores envolvidos neste processo?	(PRESSMAN, 2005)
2.1.5. Quais são os objetivos do ator dentro deste fluxo, para cada ator?	(PRESSMAN, 2005)
2.1.6. Suposição inicial (montar o cenário)	(SOMMERVILLE, 2011)
<b>2.2. Sunny Days (caso esperado / situação normal)</b>	
2.2.1. Pré-requisitos: o que precisa acontecer para que o fluxo em questão ocorra?	(PRESSMAN, 2005)
2.2.2. Caso normal: quais são os processos deste fluxo, dentro da situação esperada?	(SOMMERVILLE, 2011)
2.2.3. Quais são as principais tarefas ou funções realizadas por cada ator?	(PRESSMAN, 2005)*
2.2.4. Para cada atividade, qual é a pré-condição, qual é o objetivo e quem é o ator envolvido?	(PRESSMAN, 2005)*
2.2.5. Observações	
<b>2.3. Rainy Days (exceções)</b>	
2.3.1. O que pode dar errado? Que procedimento será tomado, neste caso?	(SOMMERVILLE, 2011) (DUMAS et al., 2013)
2.3.2. Descreva novamente o fluxo. Que exceções deveriam ser consideradas, à medida em que o fluxo é descrito?	(PRESSMAN, 2005)
2.3.3. Como os casos inesperados são resolvidos?	(DUMAS et al., 2013)
2.3.4. Observações	
<b>2.4. Final / Conclusão</b>	
2.4.1. Qual é o estado esperado do sistema, quando este fluxo terminar? O que mudou?	(SOMMERVILLE, 2011)*
2.4.2. Quais são as atividades relacionadas a este processo?	(SOMMERVILLE, 2011)
2.4.3. Alguma outra pessoa poderia me prestar informações adicionais?	(PRESSMAN, 2005)
2.4.4. Deveria eu perguntar-lhe algo mais?	(PRESSMAN, 2005)
2.4.5. Observações	

\* adaptado.

Fonte: O Autor