

UNIVERSIDADE FEDERAL DE UBERLÂNDIA

Aline de Souza Lima Abreu

**Instrumentos Digitais para aplicação do Coping
para crianças e adolescentes hospitalizados**

Uberlândia, Brasil

2021

UNIVERSIDADE FEDERAL DE UBERLÂNDIA

Aline de Souza Lima Abreu

**Instrumentos Digitais para aplicação do Coping para
crianças e adolescentes hospitalizados**

Trabalho de conclusão de curso apresentado à Faculdade de Computação da Universidade Federal de Uberlândia, Minas Gerais, como requisito exigido parcial à obtenção do grau de Bacharel em Ciência da Computação.

Orientador: Maria Adriana Vidigal de Lima

Universidade Federal de Uberlândia – UFU

Faculdade de Computação

Bacharelado em Ciência da Computação

Uberlândia, Brasil

2021

Aline de Souza Lima Abreu

Instrumentos Digitais para aplicação do Coping para crianças e adolescentes hospitalizados

Trabalho de conclusão de curso apresentado à Faculdade de Computação da Universidade Federal de Uberlândia, Minas Gerais, como requisito exigido parcial à obtenção do grau de Bacharel em Ciência da Computação.

Trabalho aprovado. Uberlândia, Brasil, 31 de maio de 2021:

Maria Adriana Vidigal de Lima
Orientador

Silvio Bacala Júnior

William Chaves de Souza Carvalho

Uberlândia, Brasil
2021

*Dedico este trabalho aos meus pais, Lazaro e Marlene, meu irmão Alex e meu esposo
Roberto Júnior.*

Agradecimentos

Agradeço a minha orientadora, Maria Adriana, que me acompanhou durante praticamente toda a graduação, desde o segundo período, me orientando, auxiliando, ensinado e motivando. Obrigada por ser tão paciente e estar sempre atenta a cada detalhe em todos projetos que pude atuar com você, com certeza levarei cada ensinamento comigo.

Agradeço aos meus pais, Lazaro e Marlene, e meu irmão Alex, por sempre me apoiarem e me incentivarem a sempre ser uma pessoa melhor.

Agradeço a meu marido, Roberto Júnior, pelo companheirismo, por sempre me apoiar, incentivar e ajudar de todas as formas para que eu pudesse conciliar minha vida acadêmica, profissional e pessoal.

Aos meus amigos, muito obrigada por fazerem parte da minha jornada e pela mão amiga nos momentos mais difíceis.

Resumo

Este trabalho tem como finalidade a evolução de dois aplicativos Android desenvolvidos com o objetivo de examinar os comportamentos de crianças e adolescentes hospitalizados. Os aplicativos foram desenvolvidos baseados na escala COPHAT - Coping da Hospitalização, Adoecimento e Tratamento que, atualmente, é aplicada através de questionários em papel. As aplicações já existentes foram denominadas como COPHAT-CA e COPHAT-PAs e as evoluções implementadas nestes aplicativos incluem: criação de novas entidades, melhoria na geração e extração dos relatórios, melhoria no gerenciamento dos questionários, reestruturação do código da aplicação e do banco de dados com o objetivo de melhorar a experiência do usuário. Antes da implementação dos aprimoramentos, foi descrita a especificação dos requisitos de toda aplicação, o que fundamentou as decisões de análise e projeto de software.

Palavras-chave: Coping, Enfrentamento, Aplicativo, Android.

Lista de ilustrações

Figura 1 – Arquitetura MVVM e suas camadas.	26
Figura 2 – Processo Scrum	30
Figura 3 – Pacote Pacientes	41
Figura 4 – Pacote Entidades	41
Figura 5 – Pacote Repository	42
Figura 6 – Quadro de Projeto do GitHub	42
Figura 7 – Listagem de pacientes	43
Figura 8 – Cadastro de pacientes - início da tela / campos iniciais	45
Figura 9 – Modal de confirmação de exclusão de pacientes	45
Figura 10 – Tela inicial com todas opções de inicialização	47
Figura 11 – Tela inicial de um novo questionário	47
Figura 12 – Tela de Questionários pendentes	48
Figura 13 – Registro do paciente Teste na tela de Questionários Pendentes	48
Figura 14 – Tela do questionário posicionado na questão de número 29	49
Figura 15 – Modal de confirmação de exclusão de questionário	49
Figura 16 – Aba Questões CA - Versão inicial	50
Figura 17 – Aba Questões P - Versão inicial	51
Figura 18 – Aba Questões CA x P - Versão inicial	51
Figura 19 – Aba Questões CA - Versão nova	52
Figura 20 – Aba Questões P - Versão nova	52
Figura 21 – Aba Questões CA x P - Versão nova	52
Figura 22 – Listagem de questionários - Versão nova	53
Figura 23 – Aba Questões CA - Versão nova por paciente	54
Figura 24 – Aba Questões P - Versão nova por paciente	55
Figura 25 – Aba Questões CA x P - Versão nova por paciente	55
Figura 26 – Desativação do Wi-Fi	57
Figura 27 – Desativação dos Dados Móveis	57
Figura 28 – Telas abertas com dados carregados, mesmo sem conexão com a internet	58
Figura 29 – Inclusão de um novo aplicador, sem conexão com a internet.	59
Figura 30 – Aplicativo exibindo um dado que ainda não foi salvo no banco de dados.	60
Figura 31 – Aplicativo e Banco de dados exibindo a informação inserida enquanto estava sem conexão.	61
Figura 32 – Listagem de hospitais	62
Figura 33 – Cadastro de hospital - início da tela / campos iniciais	63
Figura 34 – Modal de confirmação de exclusão de hospitais	63

Lista de tabelas

Tabela 1 – Tabela com resumo dos requisitos funcionais	33
--	----

Lista de abreviaturas e siglas

IDE	<i>Integrated Development Environment</i> - Ambiente de Desenvolvimento Integrado
JDK	<i>Java Development Kit</i> - Kit de desenvolvimento Java
JVM	<i>Java Virtual Machine</i> - Máquina Virtual Java
SDK	<i>Software development kit</i> - Kit de desenvolvimento de software

Sumário

1	INTRODUÇÃO	15
1.1	Objetivos	16
1.2	Motivação	17
1.3	Métodos	17
1.4	Organização do Trabalho	18
1.5	Trabalhos Relacionados	19
2	COPHAT	21
3	FUNDAMENTAÇÃO TEÓRICA	24
3.1	Android	24
3.2	Android Studio	25
3.3	Arquitetura MVVM	25
3.4	Kotlin	26
3.4.1	SDK	27
3.4.2	JDK	27
3.4.3	Máquina Virtual ART	27
3.5	Firestore	28
3.5.1	Realtime Database	28
3.6	Scrum	29
4	DESENVOLVIMENTO	31
4.1	Estado anterior das aplicações	31
4.2	Apresentação / Requisitos do COPHAT	33
4.3	Reestruturação do BD - Nova Modelagem	35
4.4	Aplicando a arquitetura MVVM	40
4.5	Desenvolvimento usando Scrum	42
4.5.1	Sprint 1 - Pacientes	43
4.5.2	Sprint 2 - Questionários	46
4.5.3	Sprint 3 - Relatórios	50
4.5.3.1	Relatório Total	50
4.5.3.2	Relatório por paciente/responsável	52
4.5.4	Sprint 4 - Offline Capabilities	55
4.5.5	Sprint 5 - Hospitais	61
5	CONCLUSÃO	64

5.1	Considerações Finais	64
5.2	Contribuições	64
5.3	Trabalhos Futuros	64
	REFERÊNCIAS	66
	 APÊNDICES	 69
	APÊNDICE A – ESPECIFICAÇÃO DE REQUISITOS	70
A.1	Introdução	70
A.1.1	Visão Geral	70
A.1.2	Situação Inicial	70
A.1.3	Situação proposta	71
A.1.4	Premissas	71
A.2	Simbologia	72
A.2.1	Objetos do fluxograma	72
A.3	Casos de Uso	73
A.3.1	Identificação dos Atores	73
A.3.2	Identificação dos Casos de Uso	74
A.3.3	Diagrama de Casos de Uso	75
A.3.4	Informações dos Casos de Uso	76
A.4	Requisitos Funcionais	79
A.4.1	RF001: Listar pacientes cadastrados (novo requisito)	79
A.4.1.1	Estória de usuário	79
A.4.1.2	Especificação Funcional	79
A.4.1.3	Diagrama de atividades	79
A.4.1.4	Protótipo de Interface de Usuário	80
A.4.1.5	Pré-condições	80
A.4.1.6	Pós-condições	80
A.4.1.7	Fluxo principal	80
A.4.2	RF002: Cadastrar paciente	81
A.4.2.1	Estória de usuário	81
A.4.2.2	Especificação Funcional	81
A.4.2.3	Diagrama de atividades	84
A.4.2.4	Protótipo de Interface de Usuário	85
A.4.2.5	Pré-condições	87
A.4.2.6	Pós-condições	88
A.4.2.7	Fluxo principal	88
A.4.2.8	Fluxo alternativo 01	88

A.4.3	RF003: Editar paciente	89
A.4.3.1	Estória de usuário	89
A.4.3.2	Especificação Funcional	89
A.4.3.3	Diagrama de atividades	90
A.4.3.4	Pré-condições	90
A.4.3.5	Pós-condições	90
A.4.3.6	Fluxo principal	91
A.4.3.7	Fluxo alternativo 01	91
A.4.4	RF004: Excluir paciente	91
A.4.4.1	Estória de usuário	91
A.4.4.2	Especificação Funcional	92
A.4.4.3	Diagrama de atividades	92
A.4.4.4	Protótipo de Interface de Usuário	93
A.4.4.5	Pré-condições	93
A.4.4.6	Pós-condições	94
A.4.4.7	Fluxo principal	94
A.4.4.8	Fluxo alternativo 01	94
A.4.5	RF005: Responder questionário	95
A.4.5.1	Estória de usuário	95
A.4.5.2	Especificação Funcional	95
A.4.5.3	Diagrama de atividades	97
A.4.5.4	Protótipo de Interface de Usuário	98
A.4.5.5	Pré-condições	101
A.4.5.6	Pós-condições	101
A.4.5.7	Fluxo principal	101
A.4.5.8	Fluxo alternativo 01	101
A.4.5.9	Fluxo alternativo 02	102
A.4.5.10	Fluxo alternativo 03	102
A.4.6	RF006: Listar questionários incompletos	102
A.4.6.1	Estória de usuário	102
A.4.6.2	Especificação Funcional	103
A.4.6.3	Diagrama de atividades	103
A.4.6.4	Protótipo de Interface de Usuário	104
A.4.6.5	Pré-condições	105
A.4.6.6	Pós-condições	105
A.4.6.7	Fluxo principal	105
A.4.7	RF007: Continuar questionário incompleto	105
A.4.7.1	Estória de usuário	105
A.4.7.2	Especificação Funcional	105

A.4.7.3	Diagrama de atividades	106
A.4.7.4	Pré-condições	106
A.4.7.5	Pós-condições	106
A.4.7.6	Fluxo principal	106
A.4.8	RF008: Excluir questionário incompleto	107
A.4.8.1	Estória de usuário	107
A.4.8.2	Especificação Funcional	107
A.4.8.3	Diagrama de atividades	107
A.4.8.4	Protótipo de Interface de Usuário	108
A.4.8.5	Pré-condições	108
A.4.8.6	Pós-condições	108
A.4.8.7	Fluxo principal	108
A.4.9	RF009: Listar questionários completos	109
A.4.9.1	Estória de usuário	109
A.4.9.2	Especificação Funcional	109
A.4.9.3	Diagrama de atividades	110
A.4.9.4	Protótipo de Interface de Usuário	110
A.4.9.5	Pré-condições	110
A.4.9.6	Pós-condições	111
A.4.9.7	Fluxo principal	111
A.4.10	RF010: Extrair relatório de todos questionários	111
A.4.10.1	Estória de usuário	111
A.4.10.2	Especificação Funcional	111
A.4.10.3	Diagrama de atividades	112
A.4.10.4	Protótipo de Interface de Usuário	113
A.4.10.5	Pré-condições	114
A.4.10.6	Pós-condições	114
A.4.10.7	Fluxo principal	114
A.4.10.8	Fluxo Alternativo 001	115
A.4.11	RF011: Extrair relatório por paciente	115
A.4.11.1	Estória de usuário	115
A.4.11.2	Especificação Funcional	115
A.4.11.3	Diagrama de atividades	116
A.4.11.4	Protótipo de Interface de Usuário	117
A.4.11.5	Pré-condições	118
A.4.11.6	Pós-condições	119
A.4.11.7	Fluxo principal	119
A.4.11.8	Fluxo Alternativo 001	120
A.4.12	RF012: Listar hospitais cadastrados (novo requisito)	120

A.4.12.1	Estória de usuário	120
A.4.12.2	Especificação Funcional	120
A.4.12.3	Diagrama de atividades	121
A.4.12.4	Protótipo de Interface de Usuário	122
A.4.12.5	Pré-condições	122
A.4.12.6	Pós-condições	122
A.4.12.7	Fluxo principal	122
A.4.13	RF013: Cadastrar hospital	123
A.4.13.1	Estória de usuário	123
A.4.13.2	Especificação Funcional	123
A.4.13.3	Diagrama de atividades	124
A.4.13.4	Protótipo de Interface de Usuário	125
A.4.13.5	Pré-condições	125
A.4.13.6	Pós-condições	125
A.4.13.7	Fluxo principal	125
A.4.13.8	Fluxo alternativo 01	126
A.4.14	RF014: Editar hospital	126
A.4.14.1	Estória de usuário	126
A.4.14.2	Especificação Funcional	126
A.4.14.3	Diagrama de atividades	127
A.4.14.4	Pré-condições	127
A.4.14.5	Pós-condições	128
A.4.14.6	Fluxo principal	128
A.4.14.7	Fluxo alternativo 01	128
A.4.15	RF015: Excluir hospital	129
A.4.15.1	Estória de usuário	129
A.4.15.2	Especificação Funcional	129
A.4.15.3	Diagrama de atividades	129
A.4.15.4	Protótipo de Interface de Usuário	130
A.4.15.5	Pré-condições	131
A.4.15.6	Pós-condições	131
A.4.15.7	Fluxo principal	131
A.4.15.8	Fluxo alternativo 01	131
A.5	Requisitos Não Funcionais	132
A.5.1	RNF001	132
A.5.2	RNF002	133
A.5.3	RNF003	133
APÊNDICE A – FICHA CADASTRAL DO PACIENTE		134

ANEXO B – QUESTIONÁRIO PARA OS PACIENTES	136
ANEXO C – QUESTIONÁRIO PARA OS PAIS OU RESPONSÁ- VEIS PELO PACIENTE	139

1 Introdução

Quando crianças e adolescentes são hospitalizados por um longo tempo, é possível que aconteçam prejuízos em seus desenvolvimentos emocionais, sociais e cognitivos. A hospitalização e a convivência com doenças graves podem gerar consequências no comportamento da criança ou adolescente, dependendo da duração da doença, da sua visibilidade e ainda dos tipos de intervenções médicas realizados. Algumas destas consequências podem ser o aparecimento de comportamentos agressivos, pesadelos, mudanças na conduta e medo de separar-se dos pais. Sendo assim, é importante examinar os comportamentos observados nas crianças e adolescentes doentes para melhor compreender suas reações ao processo de hospitalização (CASTRO, 2007).

O termo *coping*, que significa em português “enfrentamento” ou “lidar com”, é utilizado na psicologia para o estudo de como as pessoas, independentemente da idade, agem sob estresse psicológico. No caso da hospitalização de crianças e adolescentes, o estudo do *coping* pode ser aplicado para compreender se a percepção dos mesmos e de seus pais sobre os efeitos da doença, do tratamento e da internação hospitalar é concordante. Esta compreensão é de suma importância para que intervenções adequadas possam ser formuladas, considerando crianças e adolescentes internados e seus pais ou responsáveis (AMARAL, 2019).

Amaral e Enumo (2017) propõem duas versões para uma escala de Coping da Hospitalização, Adoecimento e Tratamento (COPHAT): uma versão específica para crianças e adolescentes internados (COPHAT-CA) e outra para os pais ou responsáveis pela criança ou adolescente internado (COPHAT-P). Estes dois instrumentos permitem avaliar o comportamento das crianças e adolescentes, do ponto de vista deles mesmos e de seus pais ou responsáveis em relação às estratégias de enfrentamento. Os instrumentos são compostos por questões em que deve-se informar de 0 (nunca) a 4 (sempre) o quanto a criança ou adolescente tem apresentado um determinado comportamento para enfrentar uma situação específica da hospitalização. Para o processamento e análise dos dados, além do registro dos instrumentos COPHAT-CA e COPHAT-P, devem ser coletadas também informações pessoais, familiares, sociais e econômicas do paciente e sua família. Estas informações são reunidas numa ficha de identificação familiar.

Para iniciar a automatização dos instrumentos COPHAT-CA, COPHAT-P e da ficha de identificação integrante, foram desenvolvidos protótipos de aplicativos no trabalho de conclusão de curso *Desenvolvimento de aplicativos móveis utilizando a escala COPHAT para coleta e análise de dados* de Lima (2019), considerando a substituição dos questionários em papel.

O uso de telefones móveis ou *smartphones* está em pleno processo de universalização e democratização, ampliando seu impacto sobre uma parcela cada vez maior da população. De acordo com uma pesquisa realizada pelo IBGE, no ano de 2018, o telefone móvel celular é o principal meio de acesso à internet em 99,2% dos domicílios. O uso do telefone móvel celular já está próximo da totalidade dos domicílios como o principal meio de acesso à internet (IBGE, 2018). Neste contexto, a captação dos questionários e análise dos dados poderão ter maior alcance e eficiência se realizadas com o auxílio de dispositivos móveis. Pode-se mencionar também a redução no impacto ambiental, já que não será mais necessária a utilização dos questionários em papel.

1.1 Objetivos

Este trabalho tem como objetivo implementar melhorias nos aplicativos móveis iniciados anteriormente por Lima (2019), sendo um implementado para versão COPHAT-CA e outro implementado para versão COPHAT-P, com o intuito de substituir os questionários em papel, tornando-se mais atrativo para quem responde ao questionário. Além disso, a coleta de dados se torna mais eficiente e tem como instrumento facilitador da análise dos resultados planilhas no formato *.xls*, auxiliando no diagnóstico e estudos que possam ser realizados.

A base de dados utilizada pelos aplicativos é não relacional e possui armazenamento *online*. Em consequência, as respostas dos questionários podem ser consultadas para análise a qualquer momento e através de qualquer dispositivo móvel com acesso à internet.

Antes do desenvolvimento das melhorias, foi construído o documento de especificação de requisitos completo com o diagrama de casos de uso, diagramas de atividades, protótipos e requisitos funcionais e não funcionais, o documento foi criado para que exista a documentação completa da aplicação e para auxiliar na decisão das melhorias que poderiam ser implementadas. Esta documentação pode ser encontrada no Apêndice A.

As melhorias implementadas consistem principalmente em correções de falhas em relatórios e telas, modificação na forma de identificar os pacientes e associá-los à seus responsáveis, gerenciamento dos pacientes de forma independente, gerenciamento de hospitais, aprimoramento na forma de exibição de relatórios e possibilidade de exclusão e edição de questionários que não foram respondidos totalmente. Para que estas melhorias pudessem funcionar corretamente, foi realizada uma reestruturação no banco de dados de maneira que os dados ficassem distribuídos de uma forma mais organizada, além da reestruturação do código fonte da aplicação tanto no front-end quanto no back-end. Todas estas melhorias serão discutidas em mais detalhes no capítulo 4.

1.2 Motivação

De acordo com [Malta et al. \(2014\)](#), as doenças crônicas não transmissíveis constituem o problema de saúde de maior magnitude, sendo responsáveis por 72% das mortes, com destaque para os quatro grupos de causas de mortes enfocados pela OMS: cardiovasculares, câncer, respiratórias crônicas e diabetes.

Para as crianças e adolescentes de todas as idades, internações, procedimentos médicos e cirurgias são eventos estressantes que podem causar traumas, sentimento de insegurança, medo e ansiedade. A recorrência destas experiências médicas podem prejudicar o desenvolvimento cognitivo, emocional, físico e social da criança ou do adolescente. Diante disto, é importante a aplicação do coping adaptado às crianças e adolescentes internados, com o intuito de realizar uma ação de autorregulação do comportamento, da cognição e da motivação, através de uma perspectiva desenvolvimentista ([AMARAL, 2019](#)). De acordo com [Linhares e Martins \(2015\)](#), a autorregulação consiste na habilidade de monitorar e modular a emoção, a cognição e o comportamento, para atingir um objetivo e/ou adaptar às demandas cognitivas e sociais para situações específicas.

O uso de aparelhos digitais no contexto da avaliação psicológica pode aumentar a motivação e reduzir a ansiedade, por oferecer tarefas mais dinâmicas e interativas. Tendo em vista esses benefícios e a redução de custo em longo prazo, este trabalho contribui realizando esforços para adaptar os instrumentos COPHAT em meio digital.

1.3 Métodos

O aprimoramento dos aplicativos COPHAT utilizou a metodologia do Ciclo de Engenharia de Usabilidade, proposto por [Mayhew \(1999\)](#). Esta metodologia é composta por um modelo bem detalhado do processo de desenvolvimento, envolvendo as grandes etapas de Análise de Requisitos, Projeto/Testes/Implementação e instalação. Em relação à estas etapas do Ciclo de Engenharia de Usabilidade tem-se:

- Análise de Requisitos: De acordo com [Neto \(2016\)](#) a especificação dos requisitos é a produção de um documento contendo os requisitos levantados, que podem ser sistematicamente revistos, avaliados e aprovados. Este documento estabelece um contrato entre cliente e desenvolvedor. Os requisitos geralmente são divididos em requisitos funcionais que definem a funcionalidade do sistema e requisitos não funcionais que não se referem às funções específicas do sistema, mas sim às propriedades, tais como memória, requisitos de software para funcionamento, requisitos de armazenamento, etc.
- Projeto/Testes/Implementação e instalação: Na visão de Mayhew, os ciclos de projeto, testes e implementação apresentam três subciclos, cada um focado em um nível

diferente do espaço de concepção, sendo eles o modelo conceitual, padrões de telas e acabamento de interfaces (CYBIS; BETIOL; FAUST, 2015).

- Instalação: De acordo com Cybis, Betiol e Faust (2015) esta etapa pode ser entendida como uma atividade de monitoramento da experiência dos usuários. Depois de colocado em produção, o *feedback* do usuário sobre os aspectos do uso do produto será mais fidedigno e extremamente valioso para detectar e eliminar problemas, além de detectar oportunidades de melhoria de novas versões do produto.

Esta metodologia foi escolhida pois, o processo de engenharia de usabilidade está bem estabelecido e se aplica igualmente a todos os *designs* de interface de usuário. Cada projeto é diferente e cada interface de usuário final terá uma aparência diferente, mas as atividades necessárias para chegar a um bom resultado são razoavelmente constantes (NIELSEN, 1993).

Os aplicativos foram desenvolvidos para serem utilizados em Sistemas Operacionais Android, com versão igual ou superior a 5, com API level maior ou igual a 21. O Android é o sistema operacional móvel da Google e atualmente é líder mundial nesse segmento. O sistema operacional do Android é baseado no *kernel* do Linux, que é responsável por gerenciar a memória, os processos, threads, segurança dos arquivos e pastas, além de redes e *drivers*. O *API level* é um número identificador valioso para aplicações Android, uma vez que, ao desenvolver aplicações é necessário definir quais serão os dispositivos alvos (LECHETA, 2015).

A IDE utilizada para desenvolvimento dos aplicativos foi o Android Studio, que é a IDE oficial de desenvolvimento para Android. Esta IDE é baseada no IntelliJ IDEA (KROCHMALSKI, 2014). No Android Studio a compilação é feita pelo Gradle e já possui o SDK instalado. O Android SDK é o software utilizado para desenvolver aplicações no Android, que possui um emulador para simular o dispositivo, ferramentas utilitárias e uma API completa para a linguagem JAVA, com todas as classes necessárias para desenvolver as aplicações (LECHETA, 2015).

Para o controle das atividades a serem realizadas e alterações dos status das mesmas, no intuito de serem acompanhadas por todos os envolvidos no projeto, foi criado um projeto no diretório do gitHub onde o mesmo está armazenado. Cada atividade passou pelos status *To do*, *In progress* e *Done*. O gitHub é uma plataforma de versionamento onde é possível hospedar e revisar código, gerenciar projetos e criar software (GITHUB, 2020).

1.4 Organização do Trabalho

Este trabalho está estruturado conforme as seguintes seções:

1. Breve introdução sobre o termo *coping* e a escala COPHAT, uso de telefone móvel e seus benefícios, além do objetivo do trabalho, o que motivou o desenvolvimento das aplicações e a metodologia utilizada.
2. Apresentação dos conceitos básicos da escala COPHAT para auxílio ao enfrentamento da hospitalização infantil e adolescente.
3. Resumo da teoria básica sobre as tecnologias utilizadas para desenvolvimento do projeto como um todo.
4. Definições e informações gerais sobre como foi realizada a especificação de requisitos, o desenvolvimento das melhorias das aplicações, funcionalidades, reestruturação do banco de dados e do código fonte do aplicativo.
5. Conclusão obtida com o desenvolvimento do projeto e propostas para trabalhos a serem realizados no futuro.

A especificação de requisitos do projeto foi incluída no Apêndice A.

1.5 Trabalhos Relacionados

A utilização de instrumentos digitais tem se tornado viável no campo da elaboração de experimentos em psicologia geral e ciências cognitivas. Algumas das vantagens que podem ser citadas são a padronização de experimentos, redução de erros, otimização do tempo da avaliação, registro e análise de dados com suporte computacional. Existem dezenas de softwares que podem ser utilizados para a elaboração de experimentos. Alguns deles oferecem desde a liberdade criativa de poder escrever código em linguagens como Java, R, Python e Matlab até *ToolBoxes* (caixas de ferramentas) bastante complexas. Um exemplo é o *PsyToolKit*, que foi desenvolvido por [Stoet \(2010\)](#) para criação e aplicação de experimentos comportamentais em ambiente Linux. Este toolkit possui uma interface que permite criar e executar experimentos via web, além de uma estrutura gráfica (Psy-Quest) para criação e aplicação de questionários multi-linguagem online. No campo das ciências cognitivas, estes questionários abrangem testes específicos para avaliar memória de curta duração, discriminação visual, raciocínio espacial, processo de desenvolvimento cognitivo, entre outros. Por ser um recurso não-comercial, é bastante utilizado em estudos acadêmicos.

E-Prime é um pacote de software especialmente projetado para facilitar o desenvolvimento de experimentos psicológicos e também para adquirir e analisar dados em ambiente Windows ([SCHNEIDER et al., 2002](#)). O E-Prime requer licença de uso, e é composto de diversos módulos: E-Studio, E-Basic, E-Merge, E-Recovery e E-DataAid. O E-Prime possui uma interface *drag-and-drop* para organizar e visualizar experimentos, e

sua lógica simula uma linha do tempo, na qual listas de atividades e ações se sucedem. A construção de experimentos no E-Prime utiliza a técnica de programação orientada a objetos. Desta forma, os experimentos se beneficiam de objetos pré-definidos para a composição, interação e validação dos mesmos.

No contexto do *coping* da hospitalização infantil, os trabalhos de [Furtado et al. \(2020\)](#) e [Lemos et al. \(2016\)](#) apresentam respectivamente a utilização e a construção de um brinquedo terapêutico digital interativo (BTDI) denominado “Hospital Mirim”. Foi implementado com o motor de jogos Unity para funcionar em sistemas Android, e se passa em um ambiente hospitalar com elementos que caracterizam um quarto infantil. A criança, por meio de seu avatar, pode explorar o ambiente virtual e interagir com os objetos tridimensionais. O objetivo é simular e orientar a criança, passo a passo, no procedimento de coleta de sangue, colaborando na diminuição do medo, estresse, ansiedade e dor. O experimento de [Furtado et al. \(2020\)](#) com o “Hospital Mirim” demonstrou a função psicoeducativa e lúdica do BTDI e permitiu verificar que o acesso à experiência virtual de forma antecipada pode proporcionar à criança uma reação comportamental diferenciada no momento em que é submetida à coleta de sangue.

2 COPHAT

Em 2019, [Amaral \(2019\)](#) escreveu sua tese de doutorado na área da Psicologia Pediátrica, em um contexto hospitalar. Esta área, de acordo com a *American Psychological Association*, inclui, mas não está limitada a: fatores psicossociais, de desenvolvimento e contextuais que contribuem para a etiologia, curso e resultado das condições médicas pediátricas; avaliação e tratamento de concomitantes comportamentais e emocionais de doenças, lesões e distúrbios do desenvolvimento; prevenção de doenças e lesões; promoção da saúde e comportamentos relacionados à saúde; educação, treinamento e orientação de psicólogos e prestadores de cuidados médicos; melhoria dos sistemas de atenção à saúde e defesa de políticas públicas que atendam às necessidades de crianças, adolescentes e suas famílias ([APA, 2021](#)).

É necessário explicitar as faixas etárias e suas categorizações para averiguar diferenças desenvolvimentais, logo, no estudo realizado por [Amaral \(2019\)](#) foi adotada a classificação do Estatuto da Criança e do Adolescente (ECA), onde considera-se criança até 12 anos incompletos, e adolescentes de 12 a 18 anos. É de grande relevância apresentar o impacto da hospitalização em crianças e adolescentes, pois a infância e juventude são fases do ciclo vital que refletem as seguintes etapas da vida.

Autores da área, como [Skinner e Wellborn \(1994\)](#) propuseram que, nos três conjuntos de processos adaptativos relacionados às necessidades básicas de Relacionamento, Competência e Autonomia, os eventos podem ser avaliados como desafio ou ameaça e desencadeariam respostas de coping específicas. Quando o estressor é visto como desafio, a pessoa utilizará de estratégias adaptativas dentro das seis famílias denominadas: Autoconfiança, Busca de Suporte, Resolução de Problemas, Busca de Informações, Acomodação e Negociação. Quando o estressor é visto como ameaça, a pessoa utilizará de estratégias adaptativas dentro das seis famílias denominadas: Delegação, Isolamento, Desamparo, Fuga, Submissão e Oposição.

O trabalho de [Skinner e Zimmer-Gembeck \(2007\)](#), aborda o processo de desenvolvimento do enfrentamento (*coping*) durante a infância e a adolescência, mostrando que revisões anteriores do tema descrevem os estressores neste período, englobando desde traumas, estressores crônicos, aborrecimentos diários e estressores médicos, sociais e acadêmicos. Posteriormente, [Zimmer-Gembeck, Lees e Skinner \(2011\)](#) fizeram uma revisão que trouxe a compreensão de duas tendências em relação à idade da criança para o seu repertório de *coping*: com o avanço da idade, a criança depende menos dos adultos e tem mais autossuficiência e usa o planejamento para a solução de problemas. A outra tendência se refere à escolha de estratégias de enfrentamento específicas que trazem mais eficácia

em relação ao estressor.

Chaves et al. (2000) apresenta o *coping* como um processo dinâmico que modifica-se de acordo com as avaliações e reavaliações feitas continuamente sobre um determinado evento. Por isso faz-se necessária a aplicação dos questionários para estudo com certa periodicidade.

A proposta de Amaral (2019) foi descrever e analisar o processo de elaboração de um instrumento de *coping*: Coping de Hospitalização, Adoecimento e Tratamento, denominando-se COPHAT, nas versões criança/adolescente (COPHAT-CA) e pais/responsáveis (COPHAT-P), baseando-se em variáveis relacionadas ao processo de estresse-*coping* da hospitalização de crianças e adolescentes. A aplicação do COPHAT visa tornar possível a avaliação do risco psicossocial familiar e do *coping* da hospitalização, adoecimento e tratamento, bem como a definição de uma proposta de intervenção no enfrentamento do adoecimento, hospitalização e tratamento. Os questionários correspondentes aos instrumentos COPHAT-CA e COPHAT-P estão apresentados nos Anexos B e C, e a Ficha Cadastral que acompanha o COPHAT-P encontra-se no Anexo A.

Os itens do questionário da escala COPHAT foram semanticamente construídos para detectar a intensidade do enfrentamento mal adaptativo. Foram classificados em cinco categorias para identificar as maiores frequências e constatar a ordem de frequência de cada item em sua categoria (AMARAL, 2019). A seguir, estão listadas as cinco categorias:

- Aspectos ligados à compreensão da doença e tratamento: itens que descrevem o grau de entendimento da doença e tratamento.
- Aspectos ligados à hospitalização: itens que retratam a hospitalização propriamente dita e suas formas de enfrentá-la.
- Aspectos correspondentes à realização e êxito do tratamento: itens que representam como é enfrentado o tratamento, via pensamentos, sentimentos ou comportamentos.
- Aspectos associados aos efeitos colaterais do tratamento: itens que descrevem consequências do tratamento.
- Aspectos relacionados à expectativa do retorno às atividades escolares: itens que explicitem pensamentos ou sentimentos frente ao retorno à escola após o tratamento.

Os itens são pontuados numa escala *Likert* de 0 (nunca) a 4 (sempre). Após o preenchimento da escala, o escore mínimo é de 0 e o máximo de 128 para cada fator a ser analisado. Um escore maior sugere um *coping* pior. Considerando as cinco categorias, quatro fatores foram propostos para as análises das respostas após a aplicação dos instrumentos:

Fator 1 - Compreensão da Doença

Fator 2 - Aspectos da Internação

Fator 3 - Procedimentos e Efeitos Colaterais do tratamento

Fator 4 - Expectativa de Retorno à Escola

As propriedades psicométricas da COPHAT-P foram avaliadas em [Lins et al. \(2021\)](#) a partir de Análise Fatorial Exploratória com o método de extração de Máxima Verossimilhança e rotação Varimax. Os principais resultados indicaram que a COPHAT-P é um instrumento com propriedades psicométricas confiáveis e pode ser utilizado no contexto virtual. Dessa forma, intervenções com o intuito de identificar famílias com alto risco de problemas de saúde mental podem direcionar os serviços de maneira mais apropriada.

A proposição de um instrumento informatizado mostra-se como fator positivo, pois é possível aumentar a interação com o examinando, promove maior segurança e rapidez para armazenamento dos protocolos respondidos, assim como facilita a pontuação e interpretação dos resultados. Além disto, pode-se apontar para a facilidade e praticidade da rotina de um psicólogo hospitalar, que lida com um público diverso em condições de saúde variadas ([LINS et al., 2021](#)).

3 Fundamentação Teórica

Neste capítulo serão detalhadas as tecnologias utilizadas no projeto, desde o gerenciamento das atividades através do gitHub, armazenamento dos dados através do FireBase e implementação utilizando a linguagem de programação Kotlin juntamente com a IDE Android Studio.

3.1 Android

A plataforma Android desfruta de um papel de destaque no mercado, tanto pela quantidade significativa de dispositivos produzidos como também por oferecer uma API rica, disponibilizando fácil acesso a vários recursos de hardware, tais como Wi-Fi e GPS, além de boas ferramentas para o desenvolvedor. A facilidade de desenvolver utilizando linguagens de programação bastante disseminadas (Java e Kotlin), a simplicidade e baixo custo para a publicação de aplicativos na loja Google Play e a quantidade de dispositivos Android em uso no mundo só fazem aumentar a popularidade da plataforma (MONTEIRO, 2013).

A primeira geração de telefones Android foi lançada em outubro de 2008. Em outubro de 2013, um relatório da Strategy Analytics mostrou que o Android tinha 81,3% da fatia de mercado global de *smartphones*, comparados com 13,4% da Apple, 4,1% da Microsoft e 1% do Blackberry (DEITEL; DEITEL; DEITEL, 2015).

De acordo com Deitel, Deitel e Deitel (2015), em abril de 2013 mais de 1,5 milhões de aparelhos Android (incluindo *smartphones*, tablets, etc) estavam sendo ativados diariamente. Atualmente, os dispositivos Android incluem *smartphones*, tablets, e-readers, robôs, motores a jato, satélites da NASA, consoles de jogos, geladeiras, televisões, câmeras, equipamentos voltados à saúde, relógios inteligentes (*smartwatches*), sistemas automotivos de *infotainment* de bordo (para controlar rádio, GPS, ligações telefônicas, termostato, etc) e muitos outros.

O sistema operacional Android tem como base o kernel do Linux, que é responsável pelo gerenciamento de processos, drivers, memória e energia. O middleware, por sua vez, controla a interação entre os aplicativos instalados no aparelho, facilitando a comunicação entre eles. E as aplicações-chave são programas comuns, como discador, navegador, contatos, mensagens, etc (GLAUBER, 2015).

3.2 Android Studio

O Android Studio é a IDE oficial de desenvolvimento para Android, ele foi anunciado no Google I/O 2013 e é baseado no IntelliJ IDEA da JetBrains. O Android Studio apresenta alguns diferenciais se comparado ao Eclipse, que era a ferramenta oficial ([LECHETA, 2015](#)):

- Editor visual mais fluido e com mais opções.
- Sistema de build baseado em Gradle.
- É muito integrado ao Android SDK.
- Templates de projeto para smartphones, tables, relógios, etc.
- Atualizações e melhorias frequentes.

Para automatização do *build* foi utilizado o Gradle. O Gradle é um moderno sistema de builds que segundo a própria documentação oficial é definido com a frase: "Gradle combina o poder e a flexibilidade do Ant com o gerenciamento de dependência e convenções do Maven, em uma maneira mais eficaz.". Ant e Maven são dois sistemas de builds bastante populares, e o Gradle é a evolução que combina o melhor de cada um ([LECHETA, 2015](#)).

3.3 Arquitetura MVVM

O padrão MVVM (Model-View-ViewModel), criado por John Goshmann, ajuda a separar corretamente a lógica de negócios e de apresentação de um aplicativo da interface do usuário. Manter uma separação clara entre a lógica do aplicativo e a interface do usuário ajuda a resolver inúmeros problemas de desenvolvimento e pode facilitar o teste, a manutenção e a evolução de um aplicativo. Ele também pode melhorar consideravelmente as oportunidades de reutilização de código e permite que desenvolvedores e designers de interface do usuário colaborem mais facilmente ao desenvolver suas respectivas partes de um aplicativo ([BRITCH, 2017](#)).

O MVVM mantém uma espécie de Façade (outro padrão de projeto) entre o modelo de negócios e a visão, que é a interface com o usuário. Um Façade, de maneira simples, é uma camada que recebe requisições e as delega para os responsáveis. Há uma ligação (binding) entre a View e a ViewModel, em que a View não tem acesso direto à camada de negócio, ficando isolada. Na camada ViewModel, ainda é possível implementar Commands, que são também ligados e reagem a interações do usuário, o que chamamos de lógica da camada de apresentação ([ARAÚJO, 2018](#)).

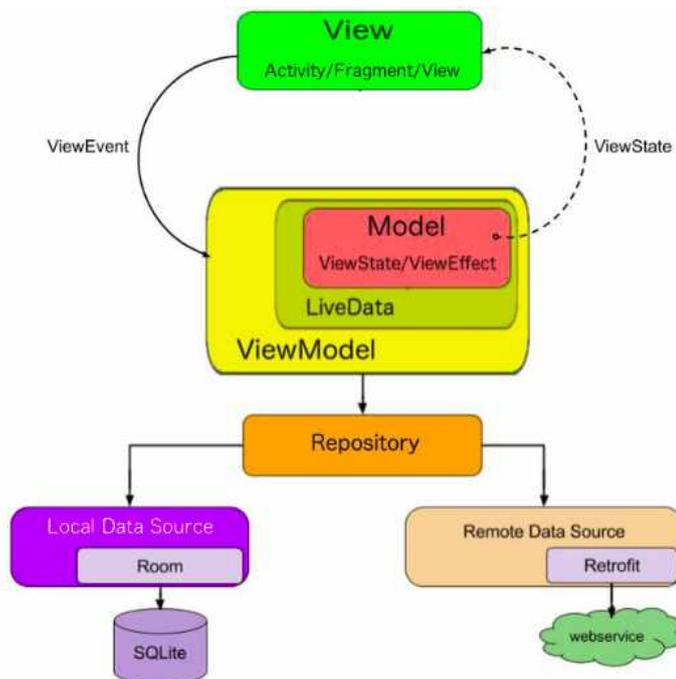


Figura 1 – Arquitetura MVVM e suas camadas.

Fonte: (SURWASE, 2020)

O diagrama da Figura 1 demonstra a ideia central desta arquitetura, em que é incluída uma camada de modelo (Model) imutável e real na arquitetura e a visão (View) depende deste modelo para mudanças de estado. Desta forma, ViewModel terá que modificar e expor este único modelo.

3.4 Kotlin

A linguagem Kotlin foi criada pela JetBrains, a mesma empresa criadora do Android Studio, e veio como uma alternativa ao Java para desenvolvimento Android. Claramente, a intenção do Kotlin não é substituir o Java, mas ser uma linguagem moderna de alta produtividade (RESENDE, 2018).

A linguagem foi primeiramente anunciada em 2011, durante o JVM Language Summit em que a JetBrains revelou que estava trabalhando havia quase um ano no projeto Kotlin. A ideia era criar uma nova linguagem estaticamente tipada para a JVM (RESENDE, 2018).

Kotlin é voltada para a plataforma Java, sendo concisa, segura, pragmática e com enfoque na interoperabilidade com código Java. Pode ser usada em quase todos os projetos em que Java é utilizada como, por exemplo, em desenvolvimento do lado do servidor e aplicações para Android. Kotlin funciona muito bem com todas as bibliotecas e os frameworks Java desenvolvidos e executa apresentando o mesmo nível de desempenho

que Java ([JEMEROV; ISAKOVA, 2017](#)).

De acordo com [Resende \(2018\)](#), esta interoperabilidade com o código Java significa que podem-se utilizar códigos que foram escritos em Java em um código Kotlin. Em consequência, pode-se aproveitar qualquer biblioteca ou classe escrita em Java nos aplicativos feitos em Kotlin.

3.4.1 SDK

Um SDK, ou Kit de Desenvolvimento de Software, é o conjunto das ferramentas e documentação que os desenvolvedores usam para construir aplicativos ([DEITEL; DEITEL, 2017](#)).

De acordo com [Glauber \(2015\)](#) o Android SDK provê as ferramentas e APIs necessárias para iniciar o desenvolvimento de aplicações para a plataforma Android usando a linguagem de programação Java. Como a linguagem Kotlin é executada na JVM, ela também desfruta das facilidades que o Android SDK oferece.

3.4.2 JDK

O JDK - *Java Development Kit*, é um conjunto de utilitários cuja finalidade é promover a criação de jogos e programas para a plataforma Java. Este pacote é disponibilizado pela Oracle, e contém todo o ambiente necessário para a criação e execução dos aplicativos Java ([DIONISIO, 2013](#)).

[Hemrajani \(2007\)](#) afirma que quando o JDK versão 1.0 foi lançado em janeiro de 1996, ele era uma interface de programação de aplicativos relativamente simples e direta. Ao longo dos anos, a linguagem Java amadureceu tornando-se uma plataforma consolidada para aplicações empresariais e científicas. Do JDK 1.0 ao JDK 1.5, muitos recursos novos foram adicionados, como o *Java Collection Framework*, API de registro em log, autoboxing, genéricos e muitos outros.

3.4.3 Máquina Virtual ART

Uma máquina virtual é um software que simula uma máquina física e consegue executar vários programas, gerenciar processos, memória e arquivos. Ou seja, ele constitui-se de uma plataforma, em que a memória, o processador e seus outros recursos, são totalmente virtuais, não dependendo de hardware ([ROMANATO, 2013](#)).

Na versão 4.4 do Android, a máquina ART foi introduzida experimentalmente para desenvolvedores, e muitos destes desenvolvedores observaram uma melhora no desempenho, mesmo em estado de desenvolvimento. A partir de 2014, na versão Android

5.0, ART foi lançada oficialmente como a máquina virtual padrão para sistemas Android ([HERCULES, 2016](#)).

[Hercules \(2016\)](#) afirma que para manter retro-compatibilidade, ART utiliza o mesmo tipo de entrada bytecode que a Dalvik (versão móvel otimizada de uma Máquina Virtual Java utilizada pela Google). ART utiliza o modo de compilação AOT (*Ahead Of Time*) em que o código é compilado quando a aplicação é instalada. A razão para mudar da Dalvik para a ART foi para incrementar a velocidade de execução de código, introduzindo suporte melhorado para processadores multinúcleo e suporte para implementação em 64 bits (uma vez que memória e espaço de armazenamento ficaram cada vez mais baratos e mais acessíveis), possibilitando a utilização de um modo mais eficiente de compilação e otimização.

3.5 Firebase

O Firebase é uma plataforma de desenvolvimento de aplicativos móveis e da web apoiada pelo Google. É uma solução completa para todas as necessidades de desenvolvimento de aplicativos móveis e da web de alta qualidade. Inclui vários produtos, como Realtime Database, Crash reporting, Cloud Firestore, Cloud Storage, Cloud functions, Authentication, Hosting, Test lab for Android e Performance Monitoring for iOS, que podem ser usados para desenvolver e testar aplicativos em tempo real com foco em necessidades do usuário, ao invés das complexidades técnicas. Ele também inclui produtos como Cloud Messaging, Google Analytics, Dynamic Links, Configuração remota, Invites, App Indexing, AdMob e AdWords que ajudam a aumentar a base de usuários e também aumentar o engajamento do público ([SINGH; TANNA, 2018](#)). Neste trabalho, foi utilizado apenas o Realtime Database.

3.5.1 Realtime Database

O Firebase Realtime Database é um banco de dados NoSQL baseado em nuvem que sincroniza dados em todos os clientes em tempo real e oferece possibilidade de funcionamento offline ([MORONEY, 2017](#)). Os dados são armazenados no banco de dados Realtime como objetos JSON (JavaScript Object Notation) e todos os clientes conectados compartilham uma instância, recebendo atualizações automaticamente com os dados mais recentes.

Usar o Firebase Realtime Database do ponto de vista da programação é uma experiência muito diferente se comparada à bancos de dados relacionais tradicionais, como MySQL ou SQL Server. Primeiramente, os dados são estruturados de uma maneira fundamentalmente diferente: usando objetos JSON em vez das tradicionais tabelas e seus relacionamentos ([MORONEY, 2017](#)).

Um recurso importante é o recurso offline (offline capabilities), no qual, o Firebase SDK persiste os dados no disco. Assim, mesmo que o usuário perca a conexão com a Internet, o aplicativo permanecerá responsivo. Ele sincroniza automaticamente os dados assim que a conexão for restabelecida (SINGH; TANNA, 2018).

Um aplicativo que executa em tempo real com o Firebase necessita monitorar a conexão quando os clientes se conectam e desconectam do banco de dados. O Firebase oferece uma solução simples para que os dados sejam armazenados quando um cliente se desconecta dos servidores do Firebase Database. Mesmo nesta situação de desconexão, é possível realizar todas as operações: escrever, configurar, atualizar e remover (SINGH; TANNA, 2018).

3.6 Scrum

A metodologia utilizada para definir as etapas de desenvolvimento foi a Scrum, pois é uma metodologia de gerenciamento de projetos ágeis, em que o software é produzido com melhor qualidade e atende às necessidades do usuário mais rapidamente com um custo menor do que as metodologias tradicionais (COSTA, 2016).

De acordo com Costa (2016), as empresas que se tornam ágeis adotando um processo como o Scrum têm experimentado benefícios significativos, incluindo ganhos de produtividade com diminuições correspondentes em custo. Elas são capazes de levar produtos ao mercado muito mais rapidamente e com maior grau de satisfação do cliente, além de proporcionar uma experiência de maior visibilidade do processo de desenvolvimento, o que leva a maior previsibilidade de resultados.

Com o Scrum, os projetos progridem em uma série de interações mensais, denominadas sprints. Antes de cada sprint, as equipes de desenvolvimento se reúnem com o cliente, ou com o dono do produto (product owner), para priorizar o trabalho a ser realizado e selecionar as atividades que a equipe pode finalizar na próxima sprint. Durante a sprint, a equipe é acompanhada através de reuniões diárias (daily scrum) e, no fim das sprints, a equipe entrega um incremento do produto que pode ser potencialmente utilizado pelo cliente. A Figura 2 é um esqueleto de processo incremental de todas as práticas do Scrum.

O **Product Backlog** é uma lista de requisitos funcionais e não funcionais que, quando transformados em funcionalidades, proporciona a visão geral do produto a ser construído. O Product Backlog tem como princípio a priorização de atividades, para que os itens com maior probabilidade de gerar valor sejam prioridade e estejam divididos em lançamentos propostos.

Já o **Sprint Backlog** consiste das tarefas que uma equipe selecionou do Product

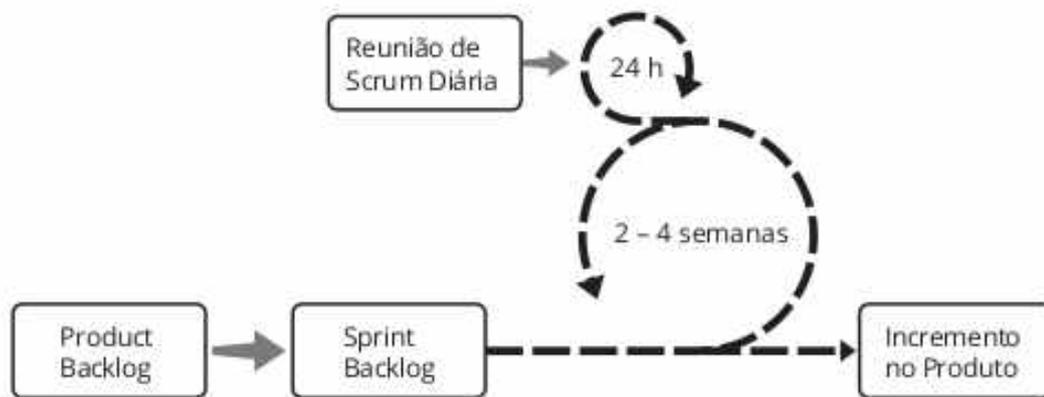


Figura 2 – Processo Scrum

Fonte: (COSTA, 2016)

Backlog para a Sprint, de maneira a transformá-lo em um incremento de funcionalidade do produto potencialmente utilizável. O Sprint Backlog é visível e compartilhado com a equipe, e deve estar organizado em um quadro que demonstra em tempo real o trabalho que a equipe planeja efetuar durante a Sprint.

O círculo inferior da Figura 2 representa uma iteração de atividades de desenvolvimento que ocorrem uma após a outra, e a saída de cada uma delas é um incremento no produto. O círculo superior indica uma atividade de inspeção diária que deve acontecer durante a iteração após um ciclo de 24 horas. A inspeção diária e contínua promove tanto a comunicação entre os membros da equipe quanto a avaliação do andamento dos trabalhos.

4 Desenvolvimento

Neste capítulo serão retratadas as novas funcionalidades e aprimoramentos implementados nos aplicativos COPHAT-P e COPHAT-CA, para melhoria da apresentação e e usabilidade.

4.1 Estado anterior das aplicações

A versão anterior dos aplicativos COPHAT, construída e documentada em [Lima \(2019\)](#), funcionava de acordo com os seguintes pontos:

- Pacientes: o paciente podia responder ao questionário sem ocorrer um cadastro prévio. Para isso, uma identificação era gerada automaticamente consistindo de uma combinação das iniciais do nome do paciente e com a data atual.
- O usuário deveria informar a data de nascimento do paciente e a idade em anos, porém na escala COPHAT há diferença de resultado para crianças que possuem diferença de idade em meses.
- Questionários: o responsável deveria iniciar o questionário na mesma data em que a criança respondeu ao questionário, para que fosse possível relacionar os questionários criança x responsável. Caso não fosse possível que o responsável respondesse o questionário em sua totalidade, ele poderia continuar depois, desde que não tivesse ocorrido nenhum outro preenchimento subsequente ao seu.
- Responsável: só era possível o preenchimento do questionário por um único responsável associado à criança. Caso a criança possuísse mais de um responsável acompanhando-o, era necessário escolher apenas um para responder.
- Relatório: se houvesse algum questionário incompleto, este também era contemplado nos relatórios, algumas vezes quebrando e/ou desconfigurando o layout do relatório.

De acordo com estas funcionalidades implementadas e algumas falhas identificadas em testes, foram apontadas as seguintes propostas de melhorias e correções:

- Criação de tela e implementação da lógica no código fonte do aplicativo para exibição dos questionários pendentes, incluindo a listagem dos questionários incompletos, botão para exclusão de um questionário e botão para dar continuidade à um questionário não finalizado.

- Alteração da regra de criação do identificador do paciente e seu responsável (*familyId*). O *familyId* foi alterado para *identificationCode*, que consiste no código de identificação usado pelos profissionais da saúde no hospital para identificar a criança. Atualmente o uso do *familyId* é constituído pelas iniciais do nome da criança e a data em que a criança respondeu o questionário, o que ocasiona problemas como: se houverem crianças com mesmo nome, o código seria replicado, causando divergências, além disto, como há a data no *familyId*, o responsável é obrigado a responder ao questionário no mesmo dia, já que a função que busca o *familyId* para relacionar os respondentes (criança e responsável) está vinculada à data. Neste item foi necessário uma grande reestruturação do código fonte, já que toda aplicação funcionava em torno do *familyId*.
- Alteração dos relatórios para substituir o *familyId* pelo *identificationCode*.
- Criação de tela e implementação da lógica no código fonte do aplicativo para listagem, cadastro e exclusão de pacientes.
- Alteração na forma de informar e armazenar a idade do paciente, no qual o cálculo passa a ser automático e em meses, à partir da data de nascimento informada até a data do preenchimento do cadastro do paciente.
- Criação de tela e implementação da lógica no código fonte do aplicativo para listagem, cadastro e exclusão de hospitais.
- Retirada das telas iniciais do questionário e alteração na lógica do código fonte que envolviam dados do paciente, já que estes dados agora possuem tela própria.
- Transformação da opção de download dos questionários em botão, pois estava em texto, não intuitivo.
- Correção da exibição dos *Radio-Button* que não estavam sendo exibidos em algumas telas.
- Implementação da lógica no código fonte do aplicativo para inclusão da funcionalidade de vários responsáveis respondentes para a mesma criança/adolescente.
- Alteração da listagem de questionários e implementação da lógica no código fonte do aplicativo para exibição apenas dos questionários que foram respondidos em sua totalidade, já que os questionários incompletos terão uma tela específica para tratativas.
- Retirar obrigatoriedade do campo prontuário.
- Alteração da questão relacionada à profissão dos pais (pai e mãe), pois não necessariamente o responsável pela criança é um dos seus pais.

4.2 Apresentação / Requisitos do COPHAT

De acordo com o guia PMBOK, coletar os requisitos é o processo de determinar, documentar e gerenciar as necessidades e requisitos das partes interessadas a fim de atender aos objetivos do projeto (PMBOK, 2013). O principal benefício deste processo é o fornecimento da base para definição e gerenciamento do escopo do projeto, incluindo o escopo do produto.

O sucesso do projeto é diretamente influenciado pelo envolvimento ativo das partes interessadas na descoberta e decomposição das necessidades em requisitos, e pelo cuidado tomado na determinação, documentação e gerenciamento dos requisitos do produto, serviço ou resultado do projeto. Os requisitos incluem condições ou capacidades que devem ser atendidas pelo projeto ou estar presentes no produto, serviço ou resultado para cumprir um acordo ou outra especificação formalmente imposta. Os requisitos incluem as necessidades quantificadas e documentadas e as expectativas do patrocinador, cliente e outras partes interessadas. Estes requisitos precisam ser obtidos, analisados e registrados com detalhes suficientes para serem incluídos na linha de base do escopo e medidos uma vez que a execução do projeto se inicie (PMBOK, 2013).

Considerando a importância e complexidade da especificação dos requisitos, que abrange os requisitos funcionais, requisitos não funcionais, diagrama de caso de uso, diagramas de atividades e protótipo, foi confeccionado o documento completo dos requisitos do projeto, que pode ser encontrado no Apêndice A. Este documento é parte fundamental no desenvolvimento do projeto, e um resumo das especificações pode ser encontrado na Tabela 1.

Tabela 1 – Tabela com resumo dos requisitos funcionais

Requisito	Caso de Uso	Status	Resumo
RF001	UC001	Implementar	Criar tela e lógica no código fonte para listar todos os pacientes cadastrados, com possibilidade de inclusão, edição e exclusão de pacientes.
RF002	UC002	Implementar	Criar tela e lógica no código fonte para cadastro de novos pacientes com campos para inclusão de informações e possibilidade de salvar ou cancelar a operação de cadastro de novo paciente.
Continua na próxima página...			

Tabela 1 – continuação da página anterior

Requisito	Caso de Uso	Status	Resumo
RF003	UC003	Implementar	Criar tela e lógica no código fonte para edição de pacientes já cadastrados, com campos para exibição das informações já cadastradas e possibilidade de salvar ou cancelar a operação de edição de cadastro de paciente.
RF004	UC004	Implementar	Implementar lógica no código fonte para exclusão de pacientes já cadastrados, a opção de exclusão se dará à partir da tela de listagem de pacientes (RF001).
RF005	UC005	Alterar	Criar opção e lógica no código fonte para inicialização do questionário com as perguntas relacionadas à escala COPHAT, com possibilidade de armazenamento de questionário de vários responsáveis para uma mesma criança, quando for o caso.
RF006	UC006	Implementar	Criar tela e lógica no código fonte referente à “Questionários Pendentes” para listagem de questionários que não foram respondidos por completo, com possibilidade de continuar ou excluir o questionário pendente.
RF007	UC007	Implementar	Implementar lógica no código fonte para continuar um questionário pendente, à partir de onde parou.
RF008	UC008	Implementar	Implementar lógica no código fonte para excluir um questionário pendente.
RF009	UC009	Alterar	Criar tela e lógica no código fonte para listagem de questionários completos, com informações do paciente e responsáveis e botões para exportar relatórios.
RF010	UC010	Alterar	Criar lógica no código fonte para extração de todos os questionários respondidos por completo.
RF011	UC011	Alterar	Criar lógica no código fonte para extração de todos os questionários respondidos por completo, relacionados à um determinado paciente.
Continua na próxima página...			

Tabela 1 – continuação da página anterior

Requisito	Caso de Uso	Status	Resumo
RF012	UC012	Implementar	Criar tela e lógica no código fonte para listar todos os hospitais cadastrados, com possibilidade de inclusão, edição e exclusão de hospitais.
RF013	UC013	Implementar	Criar tela e lógica no código fonte para cadastro de novos hospitais com campos para inclusão de informações e possibilidade de salvar ou cancelar a operação de cadastro de novo hospital.
RF014	UC014	Implementar	Criar tela e lógica no código fonte para edição de hospitais já cadastrados, com campos para exibição das informações já cadastradas e possibilidade de salvar ou cancelar a operação de edição de cadastro de hospital.
RF015	UC015	Implementar	Implementar lógica no código fonte para exclusão de hospitais já cadastrados, a opção de exclusão se dará à partir da tela de listagem de hospitais (RF001).

4.3 Reestruturação do BD - Nova Modelagem

Para atender às alterações a serem realizadas nas aplicações, houve a necessidade de reorganizar os dados no banco de dados, abaixo temos a estrutura antiga e a nova estrutura proposta:

Listagem 4.1 – Antiga estrutura

```

1 {
2   admins:{
3     id: {
4       contact: "",
5       name: ""
6     },
7   },
8   categories:{
9     id:{
10      description: "",
11      type: ""
12    },
13  },
14  forms:{
15    id:{
16      description: "",

```

```

17         type: "",
18         questions: {
19             id:{
20                 category: "",
21                 id: "",
22                 statement: ""
23             },
24             id:{
25                 category: "",
26                 id: "",
27                 statementFemale: "",
28                 statementMale: ""
29             },
30             id:{
31                 category: "",
32                 id: "",
33                 statement: "",
34                 subQuestions: {
35                     id: {
36                         alternatives: {
37                             id:{
38                                 description: "",
39                                 id: "",
40                                 type: ""
41                             },
42                         },
43                         id: "",
44                         statement: ""
45                     },
46                 }
47             },
48         },
49     },
50 },
51 hospitals:{
52     id: {
53         code: "",
54         name: ""
55     },
56 },
57 questionnaires:{
58     id:{
59         childApplication:{
60             admin: "",
61             date: "",
62             endHour: "",
63             patient: {
64                 gender: "",
65                 patientName: ""
66             },
67             startHour: "",
68             status: "",
69             answers: {
70                 id:{
71                     chosenAnswers: "",
72                     id: ""
73                 }

```

```

74         }
75     },
76     parentApplication: {
77         admin: "",
78         date: "",
79         endHour: "",
80         startHour: "",
81         status: "",
82         patient: {
83             age: "",
84             birthday: "",
85             diagnosis: "",
86             diagnosticTime: "",
87             educationDegree: "",
88             fatherProfession: "",
89             gender: "",
90             home: "",
91             hospitalizations: "",
92             internedDays: "",
93             liveInThisCity: "",
94             maritalStatus: "",
95             medicalRecords: "",
96             monthlyIncome: "",
97             motherProfession: "",
98             patientName: "",
99             religion: "",
100            schoolFrequency: "",
101            schooling: ""
102        },
103        answers: {
104            id: {
105                chosenAnswers: "",
106                id: ""
107            }
108        }
109    },
110    familyId: "",
111    hospital: ""
112 }
113 }
114 }

```

Listagem 4.2 – Nova estrutura

```

1 {
2   admins: {
3     id: {
4       contact: "",
5       name: ""
6     },
7   },
8   categories: {
9     id: {
10      description: "",
11      type: ""
12    },

```

```

13     },
14     forms:{
15         id:{
16             description: "",
17             type: "",
18             questions: {
19                 id:{
20                     category: "",
21                     id: "",
22                     statement: ""
23                 },
24                 id:{
25                     category: "",
26                     id: "",
27                     statementFemale: "",
28                     statementMale: ""
29                 },
30                 id:{
31                     category: "",
32                     id: "",
33                     statement: "",
34                     subQuestions: {
35                         id: {
36                             alternatives: {
37                                 id:{
38                                     description: "",
39                                     id: "",
40                                     type: ""
41                                 },
42                             },
43                             id: "",
44                             statement: ""
45                         },
46                     }
47                 },
48             },
49         },
50     },
51     hospitals:{
52         id: {
53             code: "",
54             name: ""
55         },
56     },
57     patients:{
58         id: {
59             address: "",
60             admin: "",
61             age: "",
62             birthday: "",
63             diagnosis: "",
64             diagnosticTime: "",
65             educationDegree: "",
66             fatherProfession: "",
67             gender: "",
68             hospital: "",
69             hospitalizations: "",

```

```

70         identificationCode: "",
71         internedDays: "",
72         intervieweeName: "",
73         liveInThisCity: "",
74         maritalStatus: "",
75         medicalRecords: "",
76         monthlyIncome: "",
77         motherProfession: "",
78         name: "",
79         relationship: "",
80         religion: "",
81         schoolFrequency: "",
82         schooling: ""
83     },
84 },
85 questionnaires: {
86     id: {
87         identificationCode: "",
88         childApplication: {
89             admin: "",
90             date: "",
91             endHour: "",
92             gender: "",
93             hospital: "",
94             startHour: "",
95             status: "",
96             answers: {
97                 id: {
98                     chosenAnswers: "",
99                     id: ""
100                 }
101             }
102         },
103         parentApplication: {
104             [
105                 0: {
106                     admin: "",
107                     date: "",
108                     endHour: "",
109                     gender: "",
110                     hospital: "",
111                     intervieweeName: "",
112                     relationship: "",
113                     startHour: "",
114                     status: "",
115                     answers: {
116                         id: {
117                             chosenAnswers: "",
118                             id: ""
119                         }
120                     }
121                 },
122                 1: {...},
123                 2: {...}
124             ]
125         }
126     }

```

127 }
128 }

Os documentos *admins*, *categories*, *forms* e *hospitals* não sofreram nenhum tipo de alteração. Já o documento *questionnaires* sofreu alterações, sendo as principais: troca do *familyId* por *identificationCode* e a possibilidade de inclusão de vários responsáveis pela criança para o mesmo questionário, através do *parentApplication*.

No caso da inclusão de vários responsáveis, o documento *parentApplication* foi alterado para que utilize a estratégia de vetor, desta forma ele poderá armazenar várias instâncias do *parentApplication* para um único questionário de determinado paciente.

Também houve a inclusão do documento *patients* para o armazenamento do cadastro prévio da criança. Desta forma os questionários referentes à esta criança poderão ser respondidos em qualquer dia e horário, sem perder a referência entre criança e responsáveis.

Estas alterações realizadas na estrutura do banco de dados, refletiram alterações realizadas também no código fonte da aplicação. A nova estrutura do banco de dados facilita a análise dos dados diretamente no console do Firebase, assim como facilita possíveis melhorias futuras, já que as entidades estão separadas e mais fáceis de serem manipuladas.

4.4 Aplicando a arquitetura MVVM

A implementação utilizando o padrão arquitetural MVVM facilita os testes das funcionalidades implementadas, pois as mesmas são fracionadas de uma forma muito organizada.

Para a inclusão das novas entidades, novas funcionalidades e alterações nas regras de negócios, foi necessário realizar uma grande reestruturação no código fonte da aplicação, assim como inclusão de várias pastas de trabalho, porém o padrão arquitetural MVVM continuou sendo utilizado. A Figura 3 mostra a implementação utilizando o padrão MVVM para o objeto/entidade *patient*, da aplicação.

As pastas de nome *activity* e *fragment* representam a *view*, referente à segunda letra da sigla MVVM, que é responsável pela definição da estrutura, layout e aparência do conteúdo que será exibido na tela.

A pasta de nome *viewmodel*, como o próprio nome já diz é referente às siglas VM, ou seja view model, do padrão MVVM. Esta camada é responsável pelo fornecimento dos dados para a interface do usuário, e também, pela regra de negócio referente à manipulação dos dados para realizar a comunicação com a camada *model*.

A pasta *adapter* é responsável por fazer uma ponte entre os dados do paciente

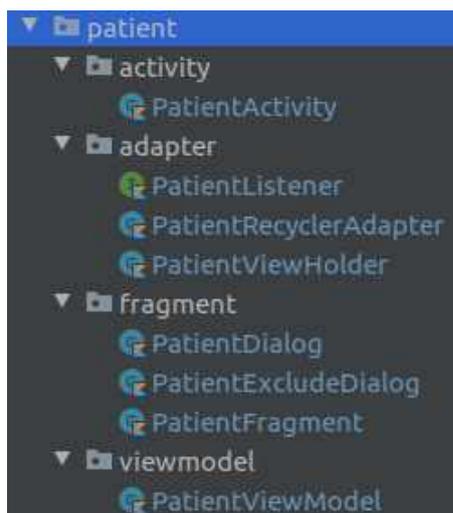


Figura 3 – Pacote Pacientes

contidos na camada *model* com a camada *view*, ou seja, o *adapter* fornece acesso aos itens de dados do *model*. A estrutura principal do modelo está inserida em uma pasta chamada *entity*, que armazena todas as entidades/objetos existentes na aplicação e seus atributos, conforme Figura 4.

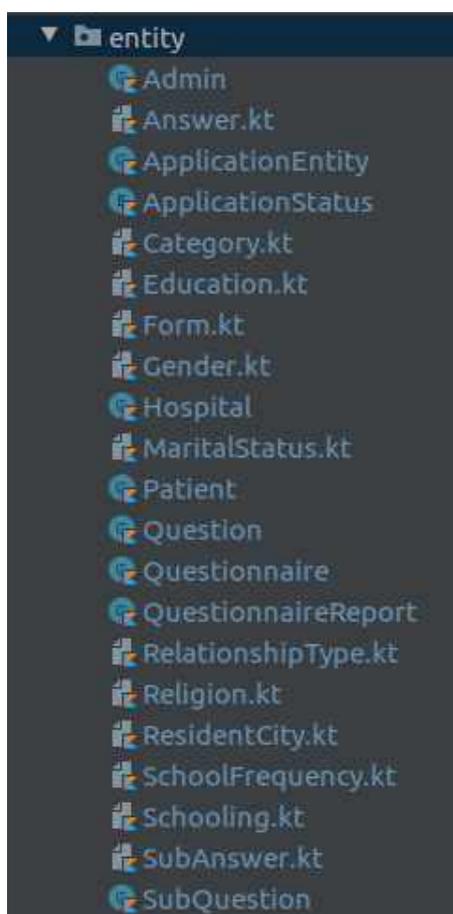


Figura 4 – Pacote Entidades

Foi incluída a pasta *repository*, que também faz parte da camada model. Na pasta *repository* estão contidos todos os componentes responsáveis pela comunicação dos objetos da aplicação com suas respectivas tabelas no banco de dados, conforme mostra a Figura 5.



Figura 5 – Pacote Repository

4.5 Desenvolvimento usando Scrum

Para facilitar a gestão das atividades foi utilizada a metodologia Scrum, conforme mencionado no Capítulo 3, juntamente com o GitHub para exibir as atividades que estão em andamento. Escolheu-se utilizar a funcionalidade de projetos do GitHub, pois nele há um quadro de atividades, que possui o status de cada atividade inserida na Sprint, e além disto, cada atividade pode ser atribuída à um desenvolvedor responsável, como mostra a Figura 6.

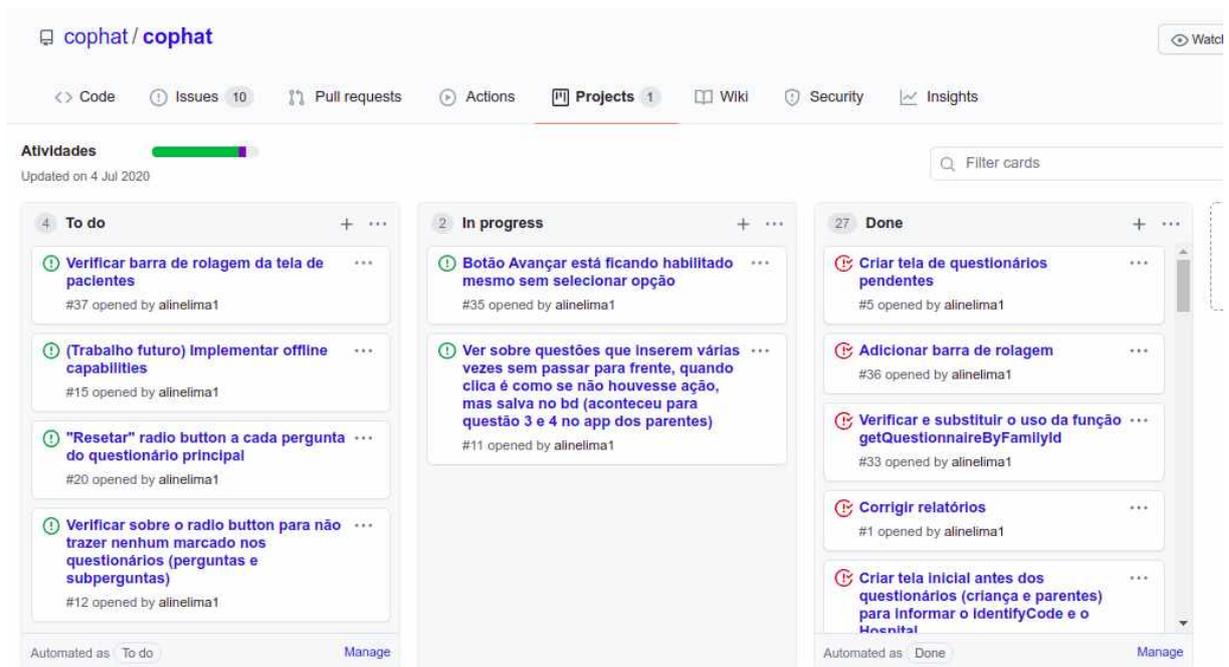


Figura 6 – Quadro de Projeto do GitHub

4.5.1 Sprint 1 - Pacientes

Na primeira Sprint foram implementados os requisitos funcionais relacionados aos pacientes, sendo eles:

- RF001: Criação da tela de listagem de todos os pacientes cadastrados, com opção de inclusão, edição e exclusão de pacientes.
- RF002: Criação de tela de inclusão de novos pacientes, com campos para inclusão de informações e possibilidade de salvar ou cancelar a operação de cadastro.
- RF003: Criação de tela para edição de pacientes cadastrados, com campos para exibição das informações já cadastradas e possibilidade de salvar ou cancelar a operação de edição de cadastro de paciente.
- RF004: Implementação de funcionalidade para exclusão de pacientes cadastrados, no qual a opção de exclusão se dá a partir da tela de listagem de pacientes.

As Figuras 7 e 8 apresentam as telas construídas nesta etapa. Na tela de listagem de pacientes foi incluída uma tabela com as informações nome e identificationCode, e em cada uma das linhas foram incorporados botões de edição e exclusão para o paciente correspondente àquela linha.



Figura 7 – Listagem de pacientes

A tela de cadastro e edição de pacientes é a mesma, o que muda é a forma na qual a tela é carregada. Caso seja escolhida a opção de cadastrar novo paciente, os campos

serão carregados vazios, e no caso de edição os campos aparecem já preenchidos com os dados atuais do paciente.



Figura 8 – Cadastro de pacientes - início da tela / campos iniciais

Também foi implementada uma janela (modal) de confirmação, conforme Figura 9, para que o usuário responda se tem certeza da ação que está executando. Desta forma pode-se evitar que seja realizada uma exclusão por engano de algum paciente.



Figura 9 – Modal de confirmação de exclusão de pacientes

4.5.2 Sprint 2 - Questionários

Na segunda Sprint foram implementados os requisitos funcionais relacionados aos questionários:

- RF005: Alterar a funcionalidade de inicialização do questionário com as perguntas relacionadas à escala COPHAT. Com possibilidade de armazenamento de questionário de vários responsáveis para uma mesma criança, quando for o caso.
- RF006: Criação da tela de questionários pendentes, para listagem de questionários que não foram respondidos por completo, com possibilidade de continuar ou excluir o questionário pendente.
- RF007: Implementação da funcionalidade de continuar um questionário pendente, à partir de onde parou.
- RF008: Implementação da funcionalidade de excluir um questionário pendente.

Anteriormente havia um único botão de “Iniciar questionário” e quando o último questionário respondido estivesse incompleto o usuário tinha opção de continuar ou cancelar para responder a um novo questionário. Porém, se cancelado, o usuário jamais conseguiria voltar para continuar respondendo este questionário que estava incompleto.

Houve uma alteração na forma como os questionários são iniciados. A ideia foi dividir em duas opções: Iniciar questionário e visualizar questionários pendentes. A opção “Iniciar questionário” foi mantida para que o usuário inicie de fato um novo questionário, independentemente de ter deixado um questionário não respondido em sua totalidade anteriormente. Na opção de visualizar “Questionários pendentes” será aberta uma tela com a listagem dos questionários que ficaram pendentes anteriormente. Ainda nesta tela há as opções de dar continuidade ao questionário pendente ou então excluí-lo. As Figuras 10, 11 e 12 apresentam as respectivas telas.



Figura 10 – Tela inicial com todas opções de inicialização

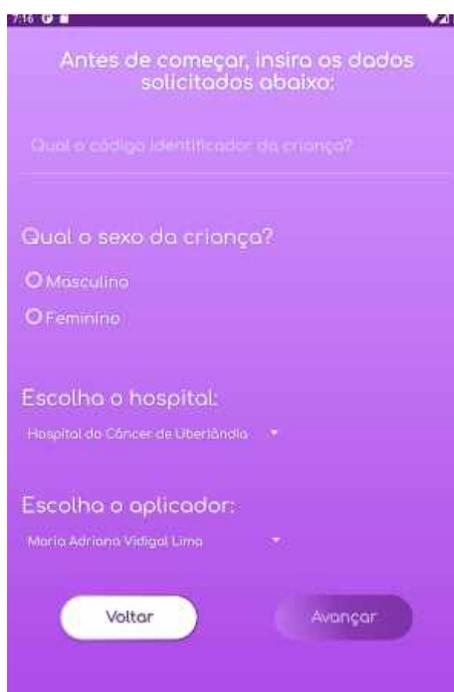


Figura 11 – Tela inicial de um novo questionário



Figura 12 – Tela de Questionários pendentes

O botão para a edição de um questionário pendente leva o usuário ao ponto de parada daquele questionário. Conforme exemplo apresentado na Figura 13, pode-se ver que na lista de questionários pendentes consta um registro para a mãe do paciente “Teste” com *identificationCode* 321.



Figura 13 – Registro do paciente Teste na tela de Questionários Pendentes

Neste exemplo, a opção de edição, simbolizada pelo ícone de um lápis, redireciona o questionário para a questão 29, que é o ponto de parada do preenchimento anterior feito por um usuário. Esta questão pode ser vista na Figura 14.



Figura 14 – Tela do questionário posicionado na questão de número 29

Assim como no cadastro de pacientes, foi incluída uma janela (modal) de confirmação de exclusão dos questionários pendentes (Figura 15), para evitar possíveis exclusões incorretas.



Figura 15 – Modal de confirmação de exclusão de questionário

4.5.3 Sprint 3 - Relatórios

Na terceira Sprint foram implementados os requisitos funcionais relacionados aos relatórios, sendo eles:

- RF009: Alterar a tela de listagem de questionários para que seja exibido apenas os questionários completos, com informações do paciente e responsáveis, além dos botões para exportar relatórios.
- RF010: Alterar a funcionalidade de exportar o relatório com todos os questionários completos, adaptando para possibilidade de múltiplos responsáveis por paciente.
- RF011: Alterar a funcionalidade de exportar o relatório com os questionários completos por paciente, adaptando a tela de listagem de questionários para exibir o quadrante do paciente uma vez para cada responsável que respondeu para o paciente em questão.

4.5.3.1 Relatório Total

Na versão inicial do relatório se algum questionário estivesse incompleto o mesmo estava sendo considerado no relatório, quando não houvesse resposta estava trazendo uma coluna em branco ou então causando quebra na formatação, além disto não havia possibilidade de exibição de mais de um responsável por paciente, conforme Figuras 16 e 17, respectivamente para as abas do paciente e seu responsável, e Figura 18 para a aba de comparação e consolidação das respostas de ambos.

QUESTÕES	AT01052020HCU	GB1052020HCU
1		4
2		4
3		3
4		3
5		4
6		4
7		4
8		0
9		2
10		4
11		4
12		3
13		1
14		2
15		3
16		3
17		4
18		2
19		3
20		2
21		4
22		1
23		1

Figura 16 – Aba Questões CA - Versão inicial

QUESTÕES	AT01052020HCU	GB1052020HCU
1		0
2		0
3		0
4		0
5		0
6		0
7		0
8		0
9		0
10		0
11		0
12		0
13		0
14		0
15		0
16		0
17		0
18		0
19		0
20		0
21		0

Figura 17 – Aba Questões P - Versão inicial

GB1052020HCU		Criança	Pai/Mãe
CATEGORIA			
Compreensão doença			22 0
Internação			21 0
Realização e êxito do tratamento			12 0
Efeitos colaterais			17 0
Expectativa do retorno à escola			17 0
TOTAL			89 0

Figura 18 – Aba Questões CA x P - Versão inicial

Na nova versão, a quantidade e nome das abas dos relatórios permaneceram as mesmas: Questões CA, Subquestões CA, Questões P, Subquestões P, Questões CA x P e Subquestões CA x P. O nome do arquivo do relatório geral, também, permaneceu o mesmo “Cophat.xls”. A identificação do paciente na linha inicial, considerada como cabeçalho do relatório, foi alterada do *familyId* para *identificationCode* mais o nome do paciente, ou o *identificationCode* mais o nome do responsável nas abas específicas das respostas dos responsáveis. Para os pacientes que possuem mais de um responsável, as colunas do paciente foram replicadas sempre fazendo par com um responsável, para facilitar a análise. As Figuras 19, 20 e 21 mostram os novos relatórios.

QUESTÕES	679 - João Silva Neto	123 - Luiza Abreu
1	3	0
2	2	0
3	3	0
4	3	0
5	4	0
6	3	0
7	1	0
8	0	0
9	2	4
10	3	0
11	2	0
12	3	0
13	2	0
14	2	0
15	3	0
16	1	0
17	4	0
18	2	0
19	3	0
20	3	0
21	3	0
22	4	0
23	2	4
24	3	4

Figura 19 – Aba Questões CA - Versão nova

QUESTÕES	679 - Marco Silva	123 - Jorge	123 - Joana
1	2	0	3
2	2	4	1
3	1	4	2
4	3	0	2
5	3	0	0
6	2	0	0
7	1	0	0
8	3	0	0
9	2	0	0
10	2	0	0
11	1	0	3
12	4	0	0
13	3	0	0
14	3	0	0
15	3	0	0
16	1	0	1
17	1	0	0
18	0	0	0

Figura 20 – Aba Questões P - Versão nova

CATEGORIA	679 - João Silva Neto		123 - Luiza Abreu		123 - Luiza Abreu	
	Criança	Responsável	Criança	Responsável	Criança	Responsável
Compreensão doença	15	10	4	0	4	10
Internação	18	12	4	4	4	12
Realização e êxito do tratamento	18	21	12	4	12	14
Efeitos colaterais	18	21	12	0	12	14
Expectativa do retorno à escola	17	14	5	0	5	21
TOTAL	86	78	37	8	37	0

Figura 21 – Aba Questões CA x P - Versão nova

4.5.3.2 Relatório por paciente/responsável

No relatório por paciente os nomes das abas dos relatórios também permaneceram os mesmos: Questões CA, Subquestões CA, Questões P, Subquestões P, Questões CA

x P e Subquestões CA x P. O nome do arquivo do relatório por paciente, mudou do *familyId* para *identificationCode*. A identificação do paciente na linha inicial, considerada como cabeçalho do relatório, também teve o *familyId* alterado para o *identificationCode* associado ao nome do paciente, ou o *identificationCode* associado ao nome do responsável nas abas específicas das respostas dos responsáveis.

A primeira mudança necessária foi na tela de listagem dos questionários, já que na nova versão um paciente pode possuir mais de um responsável relacionado, no caso o mesmo paciente poderá aparecer mais de uma vez na tela de listagem, de acordo com a quantidade de responsáveis respondentes. A imagem da Figura 22 ilustra a situação em que o paciente com *identificationCode* “123” aparece duas vezes, pois possui dois responsáveis que seriam o pai e a avó materna. As Figuras 23, 24 e 25 mostram exemplos de relatório por paciente/responsável com as respostas do paciente, do responsável e a comparação dos resultados respectivamente.



Figura 22 – Listagem de questionários - Versão nova

QUESTÕES	679 - João Silva Neto
1	3
2	2
3	3
4	3
5	4
6	3
7	1
8	0
9	2
10	3
11	2
12	3
13	2
14	2
15	3
16	1
17	4
18	2
19	3
20	3
21	3
22	4
23	2
24	3
25	4

Figura 23 – Aba Questões CA - Versão nova por paciente

QUESTÕES	679 - Marco Silva
1	2
2	2
3	1
4	3
5	3
6	2
7	1
8	3
9	2
10	2
11	1
12	4
13	3
14	3
15	3
16	1
17	1
18	0
19	3
20	2
21	2
22	1
23	1
24	2
25	2
26	2
27	2

Figura 24 – Aba Questões P - Versão nova por paciente

CATEGORIA	679 - João Silva Neto	
	Criança	Responsável
Compreensão doença	15	10
Internação	18	12
Realização e êxito do tratamento	18	21
Efeitos colaterais	18	21
Expectativa do retorno à escola	17	14
TOTAL	86	78

Figura 25 – Aba Questões CA x P - Versão nova por paciente

4.5.4 Sprint 4 - Offline Capabilities

Nesta etapa, foi implementada a funcionalidade de Offline Capabilities no sistema, para que as aplicações possam ser utilizadas mesmo após perda da conexão com a internet, por parte do dispositivo que o usuário está utilizando.

Os aplicativos do Firebase administram automaticamente interrupções temporárias de rede. Os dados em cache estão disponíveis off-line, e o Firebase reenvia quaisquer gravações quando a conectividade de rede é restaurada (GOOGLE, 2021).

A implementação da capacidade de funcionamento off-line é bastante simples, com uma única linha de código é possível ativar a persistência dos dados em disco. A Listagem 4.3 mostra a implementação da classe CophatApplication, com a ativação da persistência de dados definida na linha 21.

Listagem 4.3 – CophatApplication.kt, basicstyle = scriptsize,

```
1 package com.jodi.cophat
2
3 import android.app.Application
4 import com.google.firebase.database.FirebaseDatabase
5 import com.jodi.cophat.di.*
6 import org.koin.android.ext.koin.androidContext
7 import org.koin.android.ext.koin.androidFileProperties
8 import org.koin.android.ext.koin.androidLogger
9 import org.koin.core.context.startKoin
10
11 class CophatApplication : Application() {
12
13     override fun onCreate() {
14         super.onCreate()
15
16         startKoin {
17             androidLogger()
18             androidContext(this@CophatApplication)
19             androidFileProperties()
20             modules(listOf(networkModule, dataModule, appModule,
21                 viewModelModule, repositoryModule))
22             FirebaseDatabase.getInstance().setPersistenceEnabled(true
23                 )
24             val ref = FirebaseDatabase.getInstance().reference
25             ref.keepSynced(true)
26         }
27     }
28 }
```

Nas linhas 22 e 23 o RealTime Database sincroniza e armazena uma cópia local dos dados para os listeners ativos. Além disso, mantém as instâncias locais em sincronia. O cliente do Firebase Realtime Database faz o download automático dos dados nesses locais e os mantém sincronizados, mesmo que a referência não tenha listeners ativos.

Nas Figuras 26 a 31 podem-se ver imagens geradas durante os testes do funcionamento do offline capabilities. As Figuras 26 e 27 mostram a desativação das redes que poderiam gerar a conexão com a internet para a aplicação realizar a sincronização de dados.

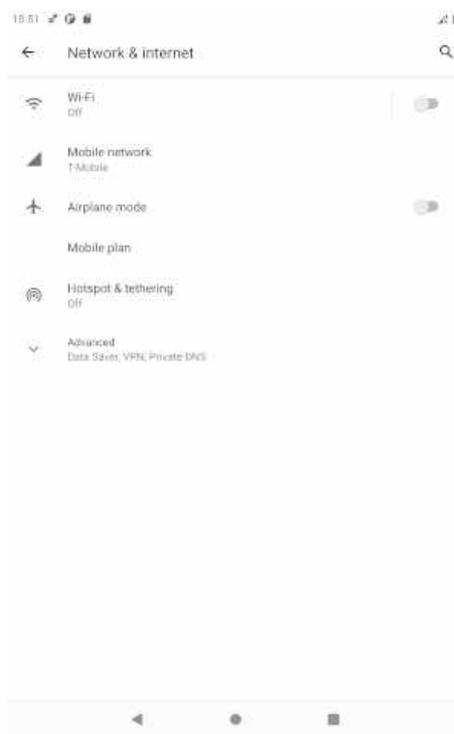


Figura 26 – Desativação do Wi-Fi

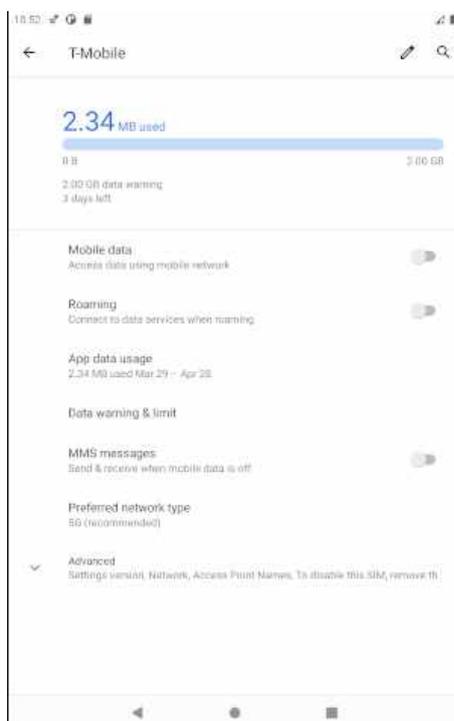


Figura 27 – Desativação dos Dados Móveis

A Figura 28 apresenta as telas Questionários Pendentes, Pacientes, Lista de questionários e Aplicadores carregadas corretamente com os dados da aplicação recuperados, mesmo sem conexão com a internet.

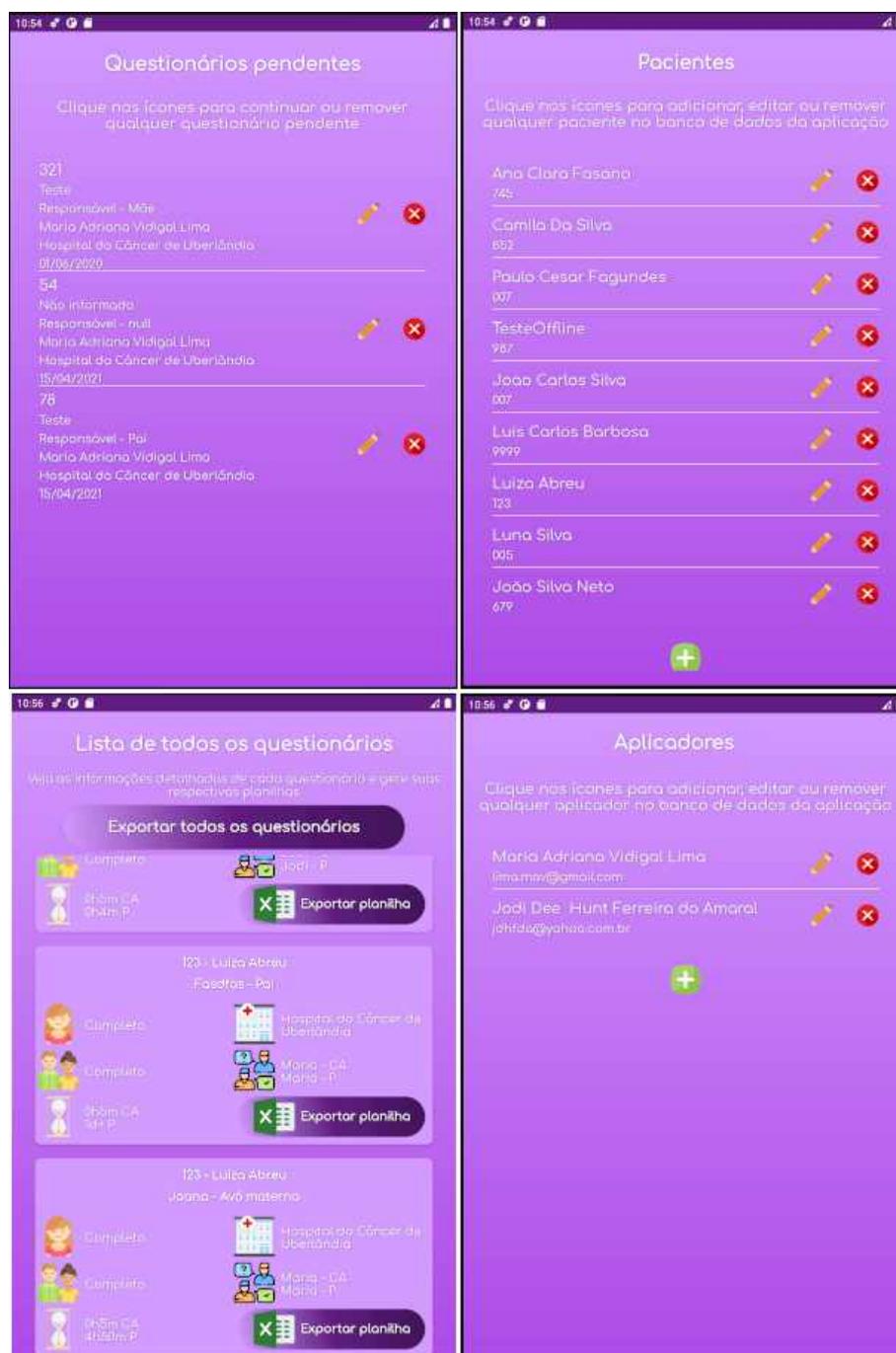


Figura 28 – Telas abertas com dados carregados, mesmo sem conexão com a internet

Na Figura 29 o usuário está com a tela de cadastro de novo aplicador aberta e com os dados do novo aplicador já inseridos na tela. Em seguida, foi realizada a ação de salvar o formulário de cadastro de aplicador.



Figura 29 – Inclusão de um novo aplicador, sem conexão com a internet.

Na Figura 30 é possível perceber que ainda não há conexão com a internet, pois existem três aplicativos cadastrados no aplicativo, porém no banco de dados apenas dois aplicativos estão registrados, ou seja, ainda não houve a sincronização.

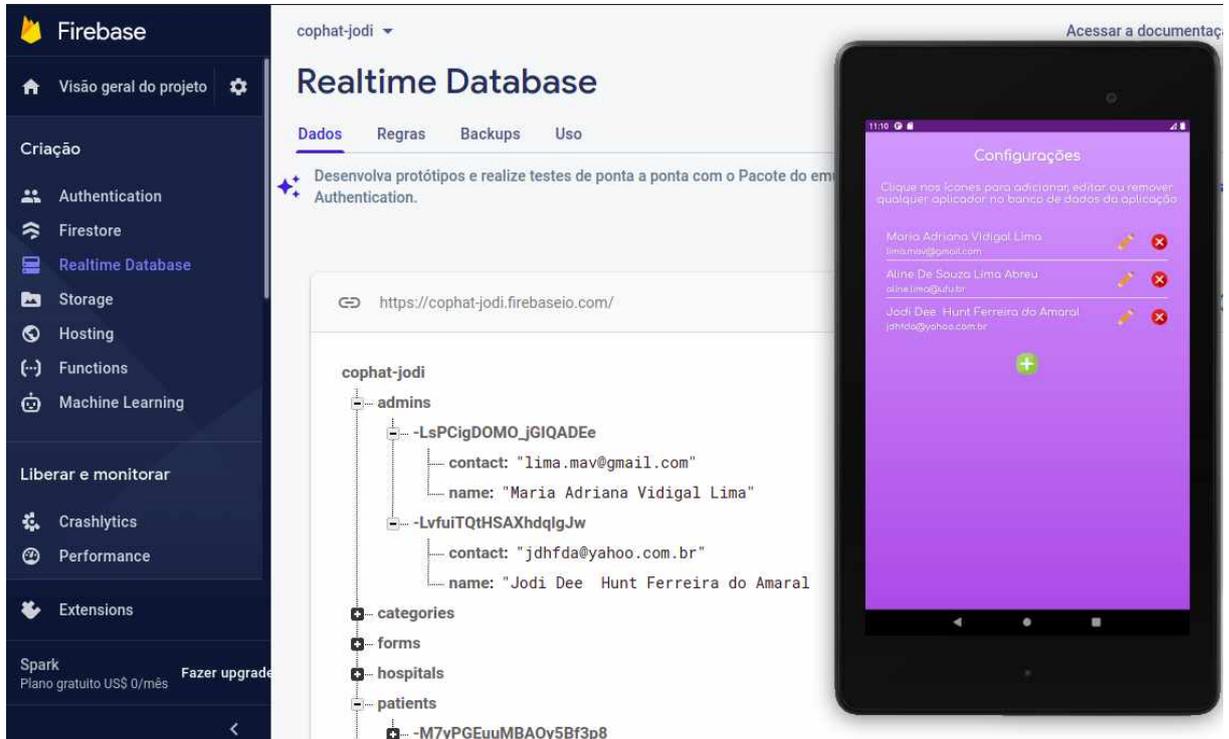


Figura 30 – Aplicativo exibindo um dado que ainda não foi salvo no banco de dados.

Por fim, a Figura 31 mostra que com o restabelecimento da conexão com a internet houve a sincronização dos dados do aplicativo para o banco de dados, e em consequência, o novo aplicador já consta no console do Firebase.

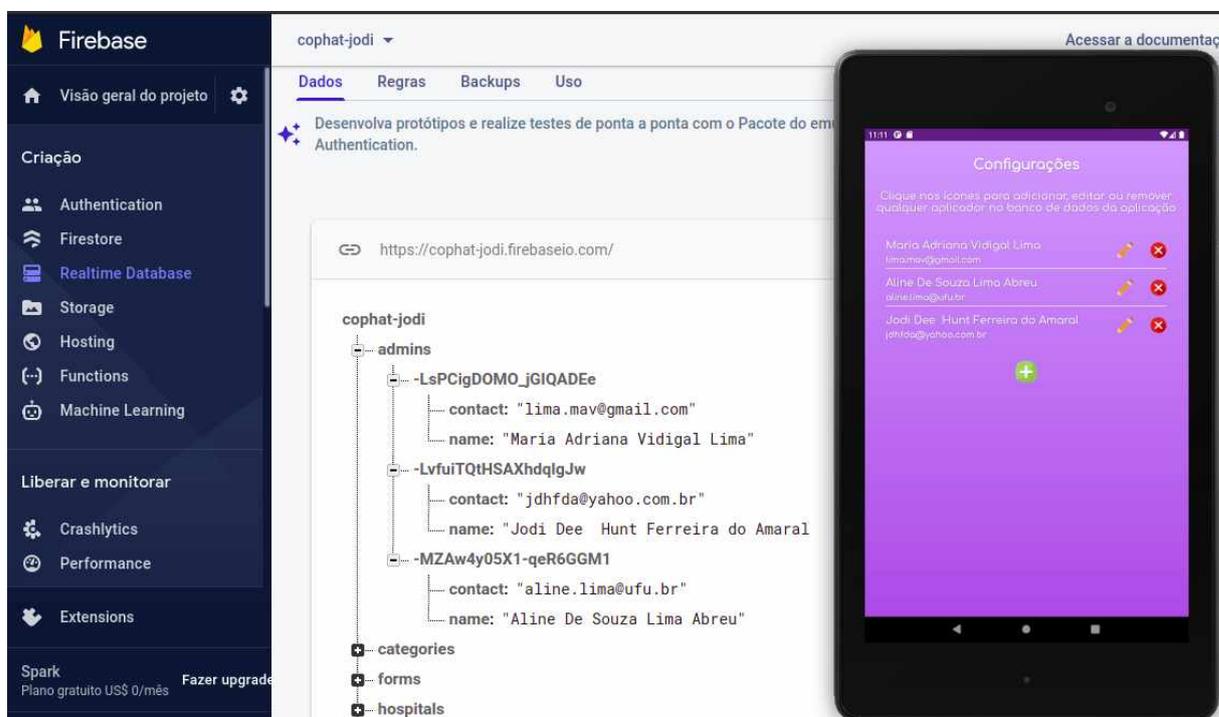


Figura 31 – Aplicativo e Banco de dados exibindo a informação inserida enquanto estava sem conexão.

4.5.5 Sprint 5 - Hospitais

Nesta Sprint foram implementados os requisitos funcionais relacionados aos hospitais, sendo eles:

- RF012: Criação da tela de listagem de todos os hospitais cadastrados, com opção de inclusão, edição e exclusão de hospitais.
- RF013: Criação de tela de inclusão de novos hospitais, com campos para inclusão de informações e possibilidade de salvar ou cancelar a operação de cadastro.
- RF014: Criação de tela para edição de hospitais cadastrados, com campos para exibição das informações já cadastradas e possibilidade de salvar ou cancelar a operação de edição de cadastro de hospital.
- RF015: Implementação de funcionalidade para exclusão de hospitais cadastrados, no qual a opção de exclusão se dá a partir da tela de listagem de hospitais.

As Figuras 32 e 33 apresentam as telas construídas nesta etapa. Na tela de listagem de hospitais foi incluída uma tabela com as informações nome e código, e em cada uma

das linhas foram incorporados botões de edição e exclusão para o hospital correspondente àquela linha.

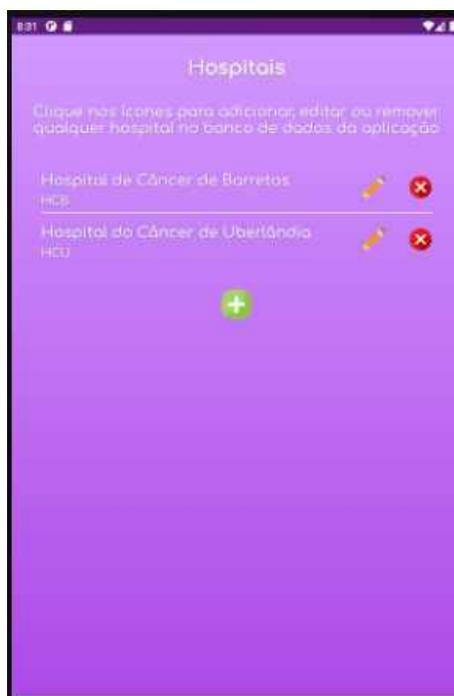


Figura 32 – Listagem de hospitais

A tela de cadastro e edição de hospitais é a mesma, o que muda é a forma na qual a tela é carregada. Caso seja escolhida a opção de cadastrar novo hospital, os campos serão carregados vazios, e no caso de edição os campos aparecem já preenchidos com os dados atuais do hospital.



Figura 33 – Cadastro de hospital - início da tela / campos iniciais

Também foi implementada uma janela (modal) de confirmação, conforme Figura 34, para que o usuário responda se tem certeza da ação que está executando. Desta forma pode-se evitar que seja realizada uma exclusão por engano de algum hospital.



Figura 34 – Modal de confirmação de exclusão de hospitais

5 Conclusão

5.1 Considerações Finais

Com este projeto foi possível observar a crescente evolução das tecnologias relacionadas ao desenvolvimento para dispositivos móveis. Cada vez mais os aplicativos, sejam eles móveis ou web, estão tomando conta dos ambientes corporativos, educacionais e hospitalares. Em relação aos aplicativos móveis tem-se, em sua grande maioria, utilização com sistema operacional Android, com público de todas idades, inclusive crianças e adolescentes. Desta forma sabemos que a aplicação dos formulários através de dispositivos móveis será uma experiência mais agradável aos pacientes e seus responsáveis, do que o preenchimento de formulários em papel.

5.2 Contribuições

Através do desenvolvimento deste projeto e evolução dos aplicativos COPHAT-CA e COPHAT-P, observa-se que é possível fazer integrações entre o ambiente hospitalar e o ambiente virtual, diminuindo a utilização de papel, otimizando o tempo dos aplicadores e usuários respondentes, além de ser uma forma muito mais atrativa para quem está respondendo o questionário, principalmente para os pacientes/crianças.

As evoluções realizadas no aplicativo vão contribuir com a gestão das informações por parte dos psicólogos, tanto em relação aos dados dos pacientes que já responderam quanto aos questionários que foram respondidos totalmente ou não, podendo ser excluídos a qualquer momento, em caso de desistência do preenchimento. Além de ter sido facilitada a análise dos relatórios, devido aos questionários serem disponibilizados para análise apenas se estiverem completos.

5.3 Trabalhos Futuros

Após a implementação dos aprimoramentos apontados neste projeto, foram levantadas outras melhorias à serem implementadas futuramente, para que os aplicativos possam ficar ainda mais funcionais e relevantes para os profissionais da psicologia pediátrica. Abaixo estão apontadas estas novas melhorias:

- Edição de questionários respondidos por completo, para possíveis correções.
- Geração de gráficos para facilitar as análises.

- Inclusão de filtros para busca de pacientes, questionários, aplicadores, etc.
- Implementação dos aplicativos COPHAT-CA e COPHAT-P para sistema operacional IOS.
- Geração de relatórios psicossociais, utilizando dados contidos nas fichas dos pacientes.
- Envio dos arquivos correspondentes aos relatórios gerados, por e-mail.

Referências

- AMARAL, J. D. H. F. do. *Risco psicossocial familiar e coping da hospitalização em crianças e adolescentes: Avaliação e intervenção*. Tese (Doutorado) — Pontifícia Universidade Católica de Campinas - Programa de Pós-Graduação em Psicologia, 2019. Citado 5 vezes nas páginas 15, 17, 21, 22 e 70.
- AMARAL, J. D. H. F. do; ENUMO, S. F. *Coping of hospitalization, ailment, and treatment (COPHAT): Design and assessment of an instrument for parents and hospitalized children and adolescents*. Tese (Doutorado) — Universidade Federal de Uberlândia; Pontifícia Universidade Católica Campinas - PUCCAMP, 2017. Citado na página 15.
- APA, A. P. A. *Society of Pediatric Psychology*. 2021. Disponível em: <<https://www.apa.org/about/division/div54>>. Acesso em: 04 apr. 2021. Citado na página 21.
- ARAÚJO, E. C. de. *Xamarin Forms e MVVM - Persistência local com Entity Framework Core*. [S.l.]: Casa do Código, 2018. v. 1. Citado na página 25.
- BRITCH, D. *Enterprise Application Patterns using Xamarin.Forms*. Redmond, Washington: DevDiv, .NET and Visual Studio produc teams - A division of Microsoft Corporation, 2017. v. 1. Citado na página 25.
- CASTRO, E. K. d. Psicologia pediátrica: a atenção à criança e ao adolescente com problemas de saúde. *Psicologia: ciência e profissão*, SciELO Brasil, v. 27, n. 3, p. 396–405, 2007. Citado na página 15.
- CHAVES, E. C. et al. Coping: significados, interferência no processo saúde-doença e relevância para a enfermagem. *SciELO Brasil*, 2000. Disponível em: <https://www.scielo.br/scielo.php?script=sci_arttext&pid=S0080-62342000000400008&lng=pt&tlng=pt>. Citado na página 22.
- COSTA, R. A. *Fundamentos do Scrum*. Rio de Janeiro - RJ: Escola Superior de Redes, 2016. Citado 2 vezes nas páginas 29 e 30.
- CYBIS, W.; BETIOL, A. H.; FAUST, R. *Ergonomia e Usabilidade: Conhecimentos, Métodos e Aplicações*. São Paulo - SP: Novatec Editora Ltda., 2015. v. 3. Citado na página 18.
- DEITEL, P.; DEITEL, H. *Java: como programar*. São Paulo: Pearson Education do Brasil, 2017. v. 10. Citado na página 27.
- DEITEL, P.; DEITEL, H.; DEITEL, A. *Android para programadores: uma abordagem baseada em aplicativos*. Porto Alegre: Bookman, 2015. v. 2. Citado na página 24.
- DIONISIO, E. J. *Introdução ao Java JDK*. [S.l.], 2013. Disponível em: <<https://www.devmedia.com.br/introducao-ao-java-jdk/28896>>. Acesso em: 22 nov. 2020. Citado na página 27.

- FURTADO, K. R. et al. O Uso do Jogo Digital "Hospital Mirim" como Estratégia de Enfrentamento à Procedimento Invasivo. *Estudos e Pesquisas em Psicologia*, scielopepsic, v. 20, p. 251 – 267, 04 2020. ISSN 1808-4281. Disponível em: <http://pepsic.bvsalud.org/scielo.php?script=sci_arttext&pid=S1808-42812020000100013&nrm=iso>. Citado na página 20.
- GITHUB. *GitHub*. 2020. Disponível em: <<https://github.com>>. Acesso em: 15 nov. 2020. Citado na página 18.
- GLAUBER, N. *Dominando o Android do básico ao avançado*. [S.l.]: Novotec, 2015. v. 1. Citado 2 vezes nas páginas 24 e 27.
- GOOGLE, F. *Como ativar recursos off-line no Android*. 2021. Disponível em: <<https://firebase.google.com/docs/database/android/offline-capabilities?hl=pt-br>>. Acesso em: 25 apr. 2021. Citado na página 55.
- HEMRAJANI, A. *Desenvolvimento ágil em Java com Spring, Hibernate e Eclipse*. São Paulo - SP: Pearson Prentice Hall, 2007. v. 1. Citado na página 27.
- HERCULES, J. R. L. *Análise de Impacto da Máquina Virtual ART em Sistemas Android*. 2016. Trabalho de Conclusão de Curso (TCC) - Engenharia de Computação. Universidade Federal do Rio Grande do Sul. Citado na página 28.
- IBGE. *Uso de Internet, televisão e celular no Brasil*. 2018. Disponível em: <<https://educa.ibge.gov.br/jovens/materias-especiais/20787>>. Acesso em: 15 nov. 2020. Citado na página 16.
- JEMEROV, D.; ISAKOVA, S. *Kotlin em Ação*. São Paulo - SP: Novatec Editora Ltda, 2017. v. 1. Citado na página 27.
- KROCHMALSKI, J. *IntelliJ IDEA Essentials*. [S.l.]: Packt Publishing Ltd, 2014. Citado na página 18.
- LECHETA, R. R. *Google Android: Aprenda a criar aplicações para dispositivos móveis com Android SDK*. São Paulo - SP: Novatec Editora Ltda., 2015. v. 4. Citado 2 vezes nas páginas 18 e 25.
- LEMOS, E. et al. Hospital mirim: Um serious game como brinquedo terapêutico digital interativo no apoio a procedimentos médicos. *Anais do SBGames*, 2016. Citado na página 20.
- LIMA, M. R. Desenvolvimento de aplicativos móveis utilizando a escala cophat para coleta e análise de dados. 2019. Citado 4 vezes nas páginas 15, 16, 31 e 70.
- LINHARES, M. B. M.; MARTINS, C. B. S. O processo da autorregulação no desenvolvimento de crianças. v. 1, n. 1, p. 3, 2015. Citado na página 17.
- LINS, C. d. A. et al. *Avaliação das Propriedades Psicométricas da Escala Coping da Hospitalização, Adoecimento e Tratamento Versão para Pais (COPHAT-P)*. 2021. Citado na página 23.
- MALTA, D. C. et al. Mortalidade por doenças crônicas não transmissíveis no Brasil e suas regiões, 2000 a 2011. *Epidemiol. Serv. Saúde*, v. 1, n. 1, p. 2, 2014. Citado na página 17.

- MAYHEW, D. J. *The Usability Engineering Lifecycle: A practitioner's handbook for user interface design*. San Diego: Morgan Kaufmann Publishers, 1999. Citado na página 17.
- MONTEIRO, J. B. *Google Android: Crie suas aplicações para celulares e tablets*. [S.l.]: Casa do Código, 2013. v. 1. Citado na página 24.
- MORONEY, L. *The Definitive Guide to Firebase - Build Android Apps on Google's Mobile Platform*. Seattle, Washington, USA: Springer Science+Business Media New York, 2017. v. 1. Citado na página 28.
- NETO, R. M. *Engenharia de Software*. Londrina - Paraná: Editora e Distribuidora Educacional S.A., 2016. Citado na página 17.
- NIELSEN, J. *Usability Engineering*. Mountain View - Califórnia: SunSoft, 1993. Citado na página 18.
- PMBOK. *Um Guia do Conhecimento em Gerenciamento de Projetos - Guia PMBOK*. [S.l.]: Project Management Institute, Inc., 2013. v. 5. Citado na página 33.
- RESENDE, K. *Kotlin com Android: Crie aplicativos de maneira fácil e divertida*. [S.l.]: Casa do Código, 2018. v. 1. Citado 2 vezes nas páginas 26 e 27.
- ROMANATO, A. *Introdução ao Java Virtual Machine*. [S.l.], 2013. Disponível em: <<https://www.devmedia.com.br/introducao-ao-java-virtual-machine-jvm/27624>>. Acesso em: 22 nov. 2020. Citado na página 27.
- SCHNEIDER, W. et al. E-prime. *Pittsburgh, PA: Psychology Software Tools*, 2002. Citado na página 19.
- SINGH, H.; TANNA, M. *Serverless Web Applications with React and Firebase*. RBirmingham - UK: Packt Publishing, 2018. v. 1. Citado 2 vezes nas páginas 28 e 29.
- SKINNER, E. A.; WELLBORN, J. G. *Coping during childhood and adolescence: A motivational perspective*. Hillsdale, NJ: Erlbaum: Life-span development and behavior, 1994. Citado na página 21.
- SKINNER, E. A.; ZIMMER-GEMBECK, M. J. The development of coping. *Annual Review of Psychology*, Annual Reviews, p. 119–144, 2007. Disponível em: <<https://www.annualreviews.org/doi/10.1146/annurev.psych.58.110405.085705>>. Citado na página 21.
- STOET, G. Psytoolkit: A software package for programming psychological experiments using linux. *Behavior research methods*, Springer, v. 42, n. 4, p. 1096–1104, 2010. Citado na página 19.
- SURWASE, R. *Best Architecture For Android : MVI + LiveData + ViewModel*. [S.l.], 2020. Disponível em: <<https://proandroiddev.com/best-architecture-for-android-mvi-livedata-viewmodel-71a3a5ac7ee3>>. Acesso em: 08 dec. 2020. Citado na página 26.
- ZIMMER-GEMBECK, M. J.; LEES, D. C.; SKINNER, E. A. *Children's emotions and coping with interpersonal stress as correlates of social competence*. Hillsdale, NJ: Erlbaum: Australian Journal of Psychology, 2011. Citado na página 21.

Apêndices

APÊNDICE A – Especificação de requisitos

A.1 Introdução

A.1.1 Visão Geral

No final do ano de 2019 foram implementados dois aplicativos com o objetivo de coletar dados sobre o impacto da hospitalização infanto-juvenil (LIMA, 2019). Estes aplicativos devem permitir que crianças e adolescentes, na faixa etária de 6 a 14 anos, em tratamento de saúde e seus responsáveis possam responder, de forma interativa, a uma série de questões que viabilizam a avaliação do enfrentamento da hospitalização, baseada na escala COPHAT (Coping da Hospitalização, Adoecimento e Tratamento) elaborada para crianças e adolescentes internadas e seus respectivos pais ou responsáveis num Hospital Geral (AMARAL, 2019).

A.1.2 Situação Inicial

Como ainda existem ajustes e melhorias à serem feitas nos aplicativos, estes ainda não estão sendo utilizados. Logo, os questionários estão sendo aplicados para as crianças e seus responsáveis através de formulário impresso.

Foi definido um código por família constituído das iniciais da criança, data de aplicação e sigla que identifica o hospital, porém a forma que foi implementado obriga o aplicador à realizar a aplicação do questionário da criança e do responsável no mesmo dia.

Outro problema identificado é que apenas um responsável por criança pode responder ao questionário, ou seja, se houver mais de um responsável acompanhando a criança, apenas um destes poderia responder, à menos que a criança responda novamente a um novo questionário.

Além disto, não há opção de cadastro da criança, com possibilidade de visualização, edição ou exclusão do cadastro.

A data de nascimento e idade da criança devem ser informadas manualmente.

Não há possibilidade de cadastro de hospitais através do aplicativo, se necessário incluir novo hospital é preciso adicionar através do banco de dados.

Em relação aos questionários, apenas o último questionário pendente que pode ter continuidade, não há a possibilidade de visualizar os demais questionários pendentes para dar continuidade ou realizar a exclusão. E quando há a solicitação de gerar relatórios, é

possível gerar para os questionários incompletos, gerando métricas inconsistentes.

A.1.3 Situação proposta

À princípio será realizada a implementação do cadastro da criança no qual será referido como paciente. Este cadastro será independente do questionário, ou seja, não é obrigatório iniciar o questionário para informar os dados do paciente. Além disto, ao incluir a data de nascimento da criança a idade será calculada, em meses, automaticamente.

Após implementar o novo cadastro de pacientes, será necessário alterar a lógica referente ao *familyId* para utilizar o código identificador, que é atribuído à todos os pacientes, este código não terá relação com a data e sim com o paciente cadastrado.

Será implementada a opção de mais de um responsável por paciente responder aos questionários, desta forma poderá haver a comparação entre as respostas de outros responsáveis, com as respostas das crianças.

Será implementada a opção de cadastro, listagem, edição e exclusão de hospitais.

Por último, será realizada a inclusão da possibilidade de excluir ou dar continuidade aos questionários pendentes, independente de quando o mesmo foi interrompido.

Nos relatórios, apenas os questionários completos serão listados para geração, e estes relatórios deverão sofrer ajustes para contemplar os múltiplos responsáveis por questionário.

A.1.4 Premissas

- Usuário deve possuir dispositivo móvel, com sistema operacional Android com versão maior ou igual a 5 (API nível maior ou igual a 21).
- Para que os dados sejam enviados ou recebidos do banco de dados, deverá haver conexão com a internet.
- As questões e subquestões devem estar cadastradas no banco de dados.

A.2 Simbologia

A.2.1 Objetos do fluxograma

Identificação	Nome	Descrição
	Ator	Ator que realizará alguma ação de um determinado caso de uso.
	Caso de Uso	<p>Elemento que referencia os casos de uso, geralmente está ligado à algum ator.</p> <p>Obs.:</p> <ul style="list-style-type: none">- Utilizamos a cor azul para identificar os casos de uso já implementados que não irão sofrer alteração.- A cor verde representa os casos de uso que ainda não foram implementados.- A cor rosa representa os casos de uso que já existem, mas sofrerão alterações.

A.3 Casos de Uso

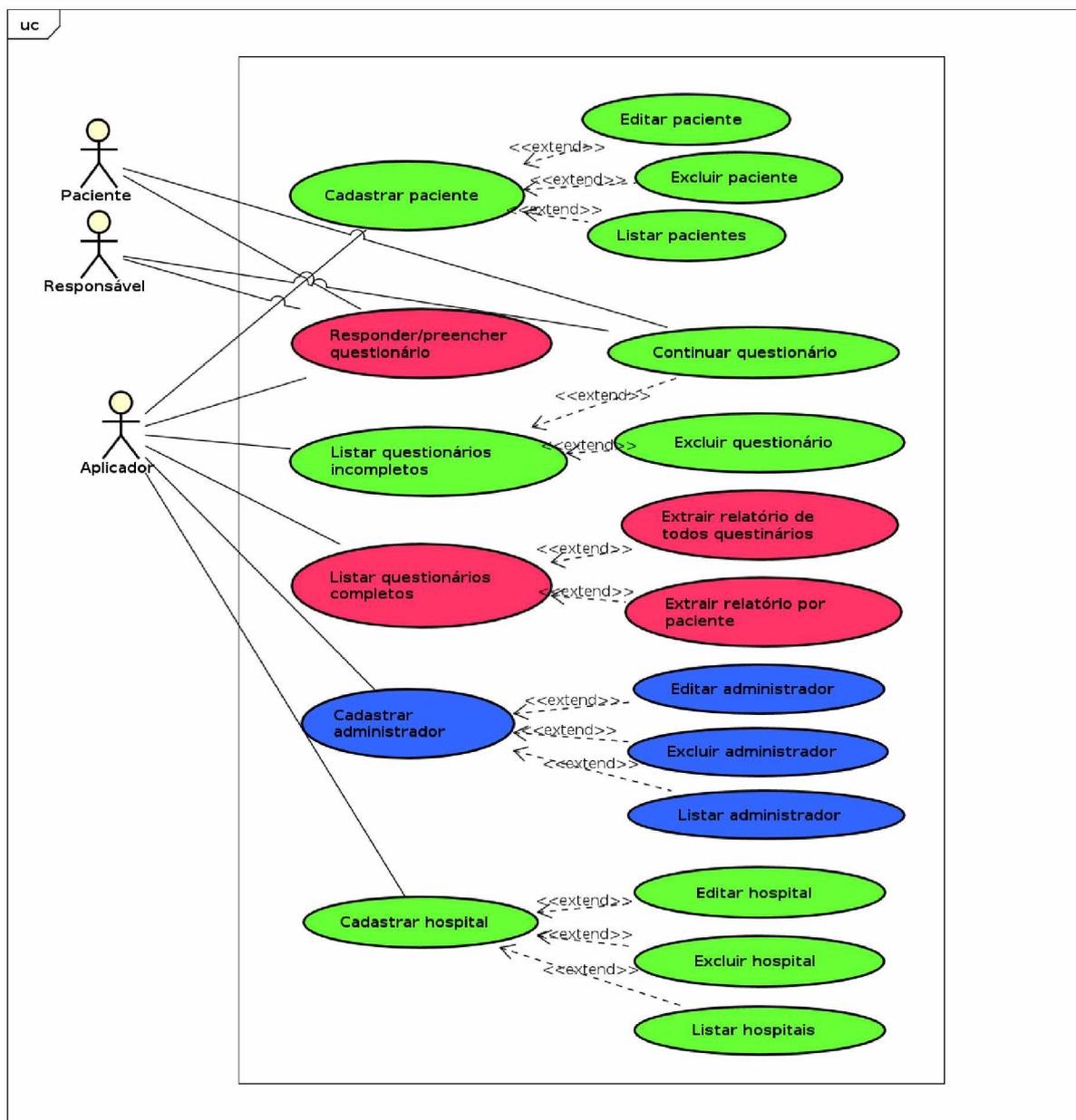
A.3.1 Identificação dos Atores

Atores	Descrição
Aplicador (psicólogo)	Usuário que irá entrevistar os atores paciente e responsável
Paciente	Criança que está internada e irá responder ao questionário
Responsável	Responsável pela criança que está internada, poderá haver mais de um responsável

A.3.2 Identificação dos Casos de Uso

ID	Caso de Uso	Descrição
UC001	Listar pacientes cadastrados	Usuário poderá visualizar todos os pacientes cadastrados no BD
UC002	Cadastrar paciente	Usuário terá a possibilidade de incluir novos pacientes
UC003	Editar paciente	Usuário terá a possibilidade de editar pacientes, quando necessário
UC004	Excluir paciente	Usuário terá a possibilidade de excluir pacientes, quando necessário
UC005	Responder questionário	Os usuários poderão responder aos questionários
UC006	Listar questionários incompletos	Caso exista questionários que não foram respondidos em sua totalidade, o usuário terá a possibilidade de visualizar quais são os questionários pendentes.
UC007	Continuar questionário incompleto	O usuário poderá selecionar um questionário incompleto para dar continuidade
UC008	Excluir questionário incompleto	O usuário poderá excluir um questionário incompleto
UC009	Listar questionários completos	O usuário poderá visualizar a lista de todos os questionários completos
UC010	Extrair relatório de todos questionários	O usuário poderá extrair o relatório que contém todos os questionários completos.
UC011	Extrair relatório por paciente	O usuário poderá extrair o relatório que contém os questionários referente à um determinado paciente.
UC012	Listar hospitais cadastrados	Usuário poderá visualizar todos os hospitais cadastrados no BD
UC013	Cadastrar hospital	Usuário terá a possibilidade de incluir novos hospitais
UC014	Editar hospital	Usuário terá a possibilidade de editar hospitais, quando necessário
UC015	Excluir hospital	Usuário terá a possibilidade de excluir hospitais, quando necessário

A.3.3 Diagrama de Casos de Uso



A.3.4 Informações dos Casos de Uso

Caso de uso:	UC001 - Listar pacientes cadastrados
Objetivo:	Exibir a lista de todos pacientes cadastrados na base de dados
Requisito(s):	RF001
Ator(es):	Aplicador
Condições de entrada:	O usuário aplicador deverá abrir o aplicativo e clicar em Pacientes.
Caso de uso	UC002 - Cadastrar paciente
Objetivo:	Cadastrar novos pacientes na base de dados
Requisito(s):	RF002
Ator(es):	Aplicador
Condições de entrada:	O usuário aplicador deverá abrir o aplicativo, clicar em Pacientes e clicar no botão 'Novo', representado pelo ícone de '+'.
Caso de uso	UC003 - Editar paciente
Objetivo:	Editar pacientes cadastrados na base de dados
Requisito(s):	RF003
Ator(es):	Aplicador
Condições de entrada:	O usuário aplicador deverá abrir o aplicativo, clicar em Pacientes e clicar no botão 'Editar', representado pelo ícone de lápis.
Caso de uso	UC004 - Excluir paciente
Objetivo:	Excluir pacientes cadastrados na base de dados
Requisito(s):	RF004
Ator(es):	Aplicador
Condições de entrada:	O usuário aplicador deverá abrir o aplicativo, clicar em Pacientes e clicar no botão 'Excluir', representado pelo ícone de 'x'.
Caso de uso	UC005 - Responder questionário
Objetivo:	Usuários paciente e responsáveis podem responder à seus respectivos questionários.
Requisito(s):	RF005
Ator(es):	Paciente e Responsáveis.
Condições de entrada:	Os usuários devem abrir o aplicativo e clicar em Iniciar questionário.

Caso de uso	UC006 - Listar questionários incompletos
Objetivo:	Listar os questionários que não foram respondidos por completo.
Requisito(s):	RF006
Ator(es):	Aplicador
Condições de entrada:	O usuário aplicador deverá abrir o aplicativo e clicar em Questionários pendentes.
Caso de uso	UC007 - Continuar questionário incompleto
Objetivo:	Continuar um questionário que não foi respondido por completo.
Requisito(s):	RF007
Ator(es):	Paciente e Responsáveis
Condições de entrada:	Os usuários devem abrir o aplicativo, clicar em Questionários pendentes e clicar em Continuar, simbolizado pelo ícone de lápis.
Caso de uso	UC008 - Excluir questionário incompleto
Objetivo:	Excluir um questionário que não foi respondido por completo.
Requisito(s):	RF008
Ator(es):	Aplicador
Condições de entrada:	O usuário devem abrir o aplicativo, clicar em Questionários pendentes e clicar em Excluir, simbolizado pelo ícone 'x'.
Caso de uso	UC009 - Listar questionários completos
Objetivo:	Listar todos os questionários que foram preenchidos até o final, por seus respectivos usuários.
Requisito(s):	RF009
Ator(es):	Aplicador
Condições de entrada:	O usuário deve abrir o aplicativo e clicar em Relatórios.
Caso de uso	UC010 - Extrair relatório de todos questionários
Objetivo:	Extrair relatório de todos questionários respondidos por completo.
Requisito(s):	RF010
Ator(es):	Aplicador
Condições de entrada:	O usuário deve abrir o aplicativo, clicar em Relatórios e por fim clicar em Exportar todos os questionários.

Caso de uso	UC011 - Extrair relatório por paciente
Objetivo:	Extrair relatório por paciente, referente aos questionários que foram respondidos por completo.
Requisito(s):	RF011
Ator(es):	Aplicador
Condições de entrada:	O usuário deve abrir o aplicativo, clicar em Relatórios e por fim clicar em Exportar planilha, de acordo com a relação paciente x responsável desejado.
Caso de uso:	UC012 - Listar hospitais cadastrados
Objetivo:	Exibir a lista de todos hospitais cadastrados na base de dados
Requisito(s):	RF012
Ator(es):	Aplicador
Condições de entrada:	O usuário aplicador deverá abrir o aplicativo e clicar em Hospitais.
Caso de uso	UC013 - Cadastrar hospital
Objetivo:	Cadastrar novos hospitais na base de dados
Requisito(s):	RF013
Ator(es):	Aplicador
Condições de entrada:	O usuário aplicador deverá abrir o aplicativo, clicar em Hospitais e clicar no botão 'Novo', representado pelo ícone de '+'.
Caso de uso	UC014 - Editar hospital
Objetivo:	Editar hospitais cadastrados na base de dados
Requisito(s):	RF014
Ator(es):	Aplicador
Condições de entrada:	O usuário aplicador deverá abrir o aplicativo, clicar em Hospitais e clicar no botão 'Editar', representado pelo ícone de lápis.
Caso de uso	UC015 - Excluir hospital
Objetivo:	Excluir hospitais cadastrados na base de dados
Requisito(s):	RF015
Ator(es):	Aplicador
Condições de entrada:	O usuário aplicador deverá abrir o aplicativo, clicar em Hospitais e clicar no botão 'Excluir', representado pelo ícone de 'x'.

A.4 Requisitos Funcionais

A.4.1 RF001: Listar pacientes cadastrados (novo requisito)

A.4.1.1 Estória de usuário

Caso de uso relacionado: UC001.

Como: Aplicador.

Desejo/Preciso: Que o aplicativo liste todos os pacientes cadastrados na base de dados.

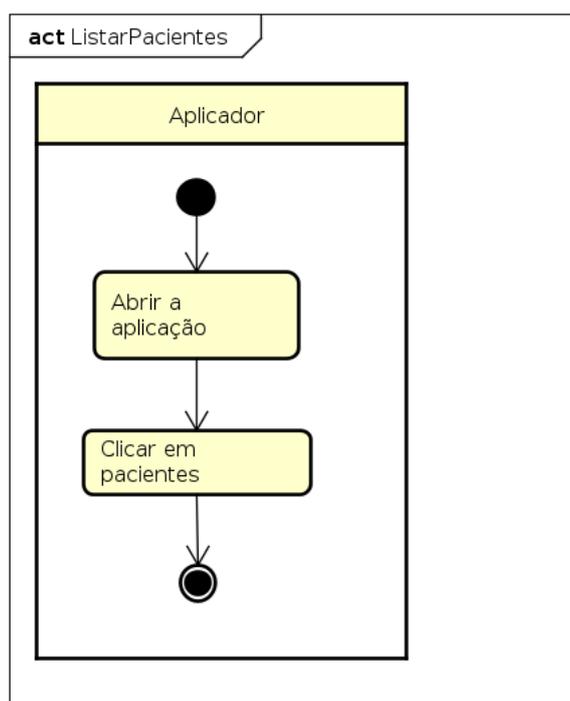
Para que/Porque: Possa realizar a conferência dos pacientes que estão cadastrados e saber quais ainda faltam ser cadastrados.

A.4.1.2 Especificação Funcional

Será necessária a implementação de uma tela que irá listar todos os pacientes cadastrados na base de dados da aplicação, nesta tela irá conter um título, um texto de orientação de uso da tela e um grid com os dados dos pacientes. Em cada linha do grid irá conter o nome e o código identificador do paciente, ao final de cada linha do grid haverá dois ícones, sendo um deles o ícone de um lápis representando a opção de editar o cadastro do paciente e o outro ícone um 'X', representando a opção de excluir um paciente.

Além disto, ao final da página, após o grid com os pacientes haverá um botão '+' para que o usuário aplicador possa incluir novos pacientes.

A.4.1.3 Diagrama de atividades



A.4.1.4 Protótipo de Interface de Usuário



A.4.1.5 Pré-condições

- Usuário aplicador deverá ter o aplicativo instalado em um dispositivo móvel, com sistema operacional Android com versão maior ou igual a 5 (API nível maior ou igual a 21).
- Usuário aplicador deverá possuir acesso à internet, para que a aplicação possa buscar as informações do banco de dados.
- Usuário deve abrir a aplicação em clicar em Pacientes.

A.4.1.6 Pós-condições

- Aplicação deverá ter aberto a tela Pacientes e, se houver, deverá ter carregado as informações dos pacientes.

A.4.1.7 Fluxo principal

1. Usuário aplicador abre a aplicação.

2. Aplicação carrega a tela inicial.
3. Usuário aplicador clica no botão Pacientes.
4. Aplicação busca as informação dos pacientes no banco de dados.
5. Aplicação exibe a tela de pacientes com as informações já cadastradas, se houver.

A.4.2 RF002: Cadastrar paciente

A.4.2.1 Estória de usuário

Caso de uso relacionado: UC002.

Como: Aplicador.

Desejo/Preciso: Cadastrar novos pacientes.

Para que/Porque: A aplicação consiga gerar os relatórios corretos referente aos questionários destes pacientes.

A.4.2.2 Especificação Funcional

Será necessária uma tela para que o usuário aplicador consiga cadastrar novos pacientes, esta tela deve possuir os seguintes campos:

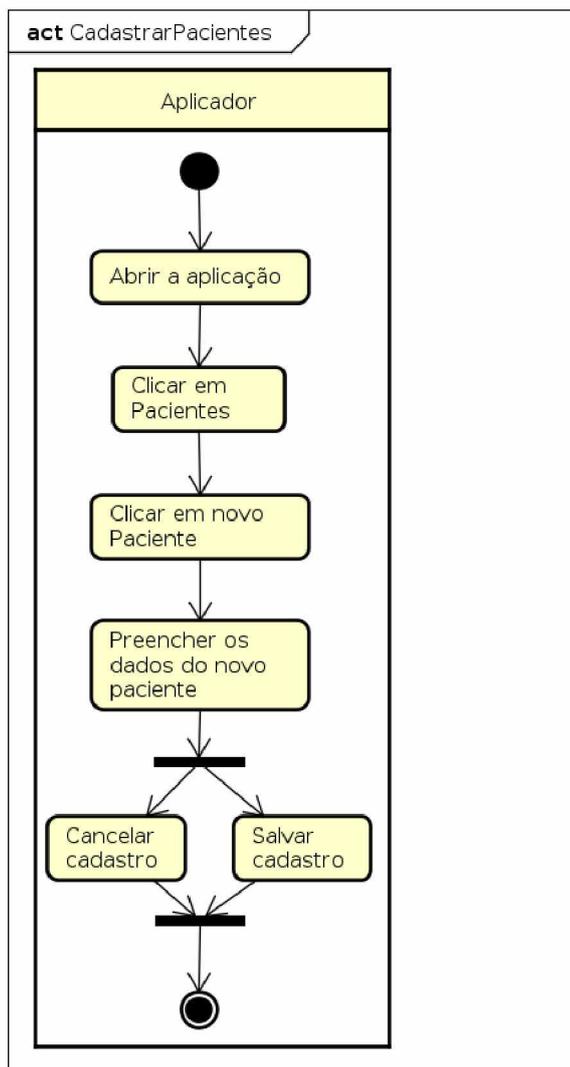
- Nome do paciente [campo de texto]
- Código de identificação [campo de texto]
- Prontuário [campo de texto]
- Nome do entrevistado [campo de texto]
- Qual o grau de parentesco com a criança? [Lista de opções - escolha única]
 - Pai [Opção - Radio Button]
 - Mãe [Opção - Radio Button]
 - Avó paterna [Opção - Radio Button]
 - Avô paterno [Opção - Radio Button]
 - Avó materna [Opção - Radio Button]
 - Avô paterno [Opção - Radio Button]
 - Outro [Opção - Radio Button]
 - Outro grau de parentesco [Campo de texto que ficará ativo apenas se a opção Outro for selecionada]
- Qual a profissão da mãe? [campo de texto]

- Qual a profissão do pai? [campo de texto]
- Qual o estado civil dos pais? [Lista de opções - escolha única]
 - Casado [Opção - Radio Button]
 - Amasiado [Opção - Radio Button]
 - Divorciado/Separado [Opção - Radio Button]
 - Solteiro [Opção - Radio Button]
 - Viúvo [Opção - Radio Button]
- Renda de todos que contribuem com as despesas [campo numérico]
- Qual é o grau de instrução do chefe da família? (Considere como chefe da família a pessoa que contribui com a maior parte da renda familiar) [Lista de opções - escolha única]
 - Analfabeto / Fundamental I incompleto [Opção - Radio Button]
 - Fundamental I completo / Fundamental II incompleto [Opção - Radio Button]
 - Fundamental II completo / Médio incompleto [Opção - Radio Button]
 - Médio completo / Superior incompleto [Opção - Radio Button]
 - Superior completo [Opção - Radio Button]
- Qual a religião da família? [Lista de opções - escolha única]
 - Católica [Opção - Radio Button]
 - Evangélica [Opção - Radio Button]
 - Espírita [Opção - Radio Button]
 - Outra [Opção - Radio Button]
 - Outra religião [Campo de texto que ficará ativo apenas se a opção Outra for selecionada]
 - Nenhuma [Opção - Radio Button]
- Nascimento [Campo de data, padrão dd/mm/aaaa]
- Idade [Campo numérico]
- Escolaridade [Lista de opções - escolha única]
 - Pré [Opção - Radio Button]
 - 1º ano [Opção - Radio Button]
 - 2º ano [Opção - Radio Button]

- 3º ano [Opção - Radio Button]
- 4º ano [Opção - Radio Button]
- 5º ano [Opção - Radio Button]
- 6º ano [Opção - Radio Button]
- 7º ano [Opção - Radio Button]
- 8º ano [Opção - Radio Button]
- 9º ano [Opção - Radio Button]
- Frequenta a escola fora da internação? [Lista de opções - escolha única]
 - Sim [Opção - Radio Button]
 - Não [Opção - Radio Button]
- Residente nesta cidade? [Lista de opções - escolha única]
 - Sim [Opção - Radio Button]
 - Não [Opção - Radio Button]
- Moradia (Bairro / Cidade / Estado) [Campo de texto]
- Qual o sexo da criança? [Lista de opções - escolha única]
 - Masculino [Opção - Radio Button]
 - Feminino [Opção - Radio Button]
- Diagnóstico [Campo de texto]
- Tempo de diagnóstico (meses) [Campo numérico]
- Dias internados [Campo numérico]
- Internações [Campo numérico]
- Escolha o hospital: [Lista de opções - Campo drop down]
 - Hospital do Câncer de Uberlândia [Opção do drop down]
 - Hospital do Câncer de Barretos [Opção do drop down]
- Escolha o aplicador: [Lista de opções - Campo drop down]
 - A aplicação deverá listar como opções todos os aplicadores cadastrados na opção "Aplicadores".

Ao final destes campos o usuário terá duas opções de botões, sendo eles "Cancelar" e "Salvar".

A.4.2.3 Diagrama de atividades



A.4.2.4 Protótipo de Interface de Usuário

12:41

Adicionar paciente

Insira abaixo os dados do paciente que deseja cadastrar

Nome do paciente

Código de Identificação

Endereço

Qual o nome do entrevistado?

Qual o grau de parentesco com a criança?

Pai

Mãe

12:42

Qual o grau de parentesco com a criança?

Pai

Mãe

Avó paterna

Avô paterno

Avó materna

Avô materno

Outro

Outro grau de parentesco:

Qual a profissão da mãe?

Qual a profissão do pai?

Qual o estado civil dos pais?

12:42

Qual o estado civil dos pais?

Casado

Amasiado

Divorciado/Separado

Solteiro

Viúvo

Renda de todos que contribuem com as despesas...

Qual é o grau de instrução do chefe da família? (Considere como chefe da família a pessoa que contribui com a maior parte da renda do domicílio)

Analfabeto / Fundamental I incompleto

Fundamental I completo / Fundamental II incompleto

Fundamental II completo / Médio incompleto

Médio completo / Superior incompleto

Superior completo

12:43

Qual a religião da família?

Católica

Evangélica

Espírita

Outra

Outra religião

Nenhuma

Nascimento

Idade

Escolaridade

Pré

1º ano

2º ano

5º ano

6º ano

7º ano

Frequenta a escola fora da internação?

Sim

Não

Residente nesta cidade?

Sim

Não

Moradia (Bairro / Cidade / Estado)

Qual o sexo da criança?

Masculino

Feminino

Diagnóstico

Diagnóstico

Tempo de diagnóstico (meses)

Dias internado

Internações

Escolha o hospital:

Hospital do Câncer de Uberlândia

Escolha o optificador:

Maria Adriana Vidigal Lima

Cancelar

Salvar

A.4.2.5 Pré-condições

- Usuário aplicador deverá ter o aplicativo instalado em um dispositivo móvel, com sistema operacional Android com versão maior ou igual a 5 (API nível maior ou igual a 21).

- Usuário aplicador deverá possuir acesso à internet, para que a aplicação possa salvar as informações no banco de dados.
- Usuário deve abrir a aplicação > Clicar em Pacientes > Clicar em Novo Paciente > Preencher os dados do paciente > Clicar em Salvar.

A.4.2.6 Pós-condições

- Aplicação deverá ter salvo as informações cadastradas pelo usuário Aplicador.

A.4.2.7 Fluxo principal

1. Usuário aplicador abre a aplicação.
2. Aplicação carrega a tela inicial.
3. Usuário aplicador clica no botão Pacientes.
4. Aplicação busca as informação dos pacientes no banco de dados.
5. Aplicação exibe a tela de pacientes com as informações já cadastradas, se houver.
6. Usuário aplicador clica no botão novo paciente, simbolizado pelo ícone com '+?.
7. Aplicação redireciona o usuário aplicador para tela de cadastro de novo paciente.
8. Usuário aplicador preenche os dados cadastrais.
9. Usuário aplicador clica no botão Salvar.

A.4.2.8 Fluxo alternativo 01

1. Usuário aplicador abre a aplicação.
2. Aplicação carrega a tela inicial.
3. Usuário aplicador clica no botão Pacientes.
4. Aplicação busca as informação dos pacientes no banco de dados.
5. Aplicação exibe a tela de pacientes com as informações já cadastradas, se houver.
6. Usuário aplicador clica no botão novo paciente, simbolizado pelo ícone com '+?.
7. Aplicação redireciona o usuário aplicador para tela de cadastro de novo paciente.
8. Usuário aplicador clica no botão Cancelar.

A.4.3 RF003: Editar paciente

A.4.3.1 Estória de usuário

Caso de uso relacionado: UC003.

Como: Aplicador.

Desejo/Preciso: Que o aplicativo dê a opção de editar pacientes já cadastrados.

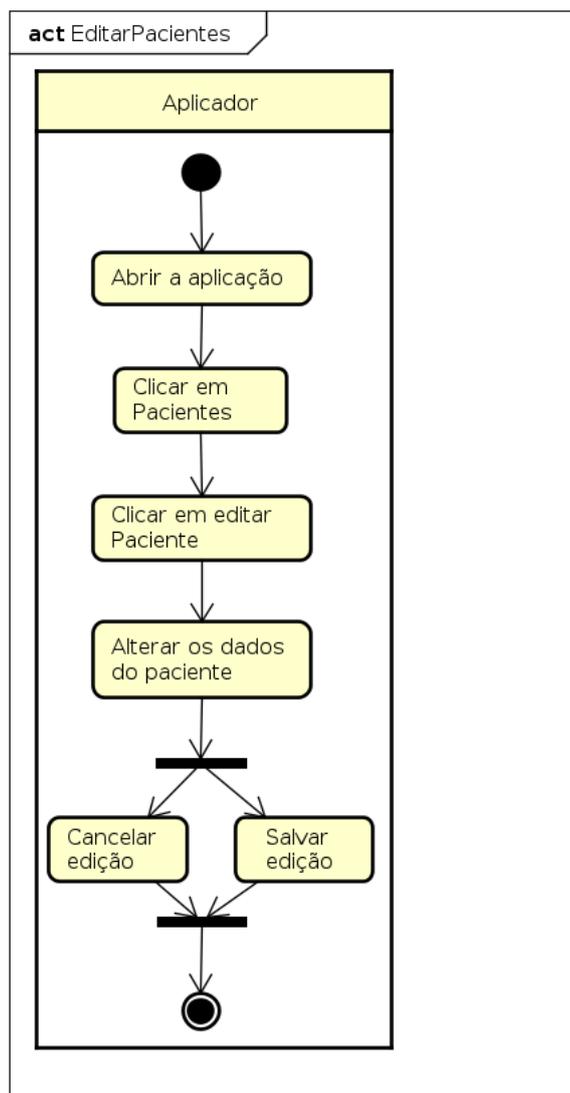
Para que/Porque: O usuário aplicador possa fazer possíveis correções cadastrais.

A.4.3.2 Especificação Funcional

Será utilizada a mesma tela que foi especificada em RF002: Cadastrar paciente, apenas o comportamento da aplicação que será diferente. Quando o usuário aplicador clicar na opção de editar cliente, na tela de Listagem de pacientes, a aplicação abrirá a tela de cadastro de paciente já preenchida com os dados do paciente selecionado anteriormente para edição.

Lembrando que o ícone para edição que irá aparecer na tela de Listagem de Pacientes é simbolizada por um lápis, conforme especificado em RF001: Listar Pacientes.

A.4.3.3 Diagrama de atividades



A.4.3.4 Pré-condições

- Usuário aplicador deverá ter o aplicativo instalado em um dispositivo móvel, com sistema operacional Android com versão maior ou igual a 5 (API nível maior ou igual a 21).
- Usuário aplicador deverá possuir acesso à internet, para que a aplicação possa buscar e salvar as informações no banco de dados.
- Usuário deve abrir a aplicação > Clicar em Pacientes > Clicar em Editar Paciente > Alterar os dados do paciente > Clicar em Salvar.

A.4.3.5 Pós-condições

- Aplicação deverá ter salvo as informações alteradas pelo usuário Aplicador.

A.4.3.6 Fluxo principal

1. Usuário aplicador abre a aplicação.
2. Aplicação carrega a tela inicial.
3. Usuário aplicador clica no botão Pacientes.
4. Aplicação busca as informação dos pacientes no banco de dados.
5. Aplicação exibe a tela de pacientes com as informações já cadastradas, se houver.
6. Usuário aplicador clica no botão editar paciente, simbolizado pelo ícone de lápis.
7. Aplicação redireciona o usuário aplicador para tela de edição de paciente.
8. Usuário aplicador altera os dados cadastrais conforme necessidade.
9. Usuário aplicador clica no botão Salvar.

A.4.3.7 Fluxo alternativo 01

1. Usuário aplicador abre a aplicação.
2. Aplicação carrega a tela inicial.
3. Usuário aplicador clica no botão Pacientes.
4. Aplicação busca as informação dos pacientes no banco de dados.
5. Aplicação exibe a tela de pacientes com as informações já cadastradas, se houver.
6. Usuário aplicador clica no botão editar paciente, simbolizado pelo ícone de lápis.
7. Aplicação redireciona o usuário aplicador para tela de edição de paciente.
8. Usuário aplicador clica no botão Cancelar.

A.4.4 RF004: Excluir paciente

A.4.4.1 Estória de usuário

Caso de uso relacionado: UC004.

Como: Aplicador.

Desejo/Preciso: Que o aplicativo dê a opção de excluir pacientes já cadastrados.

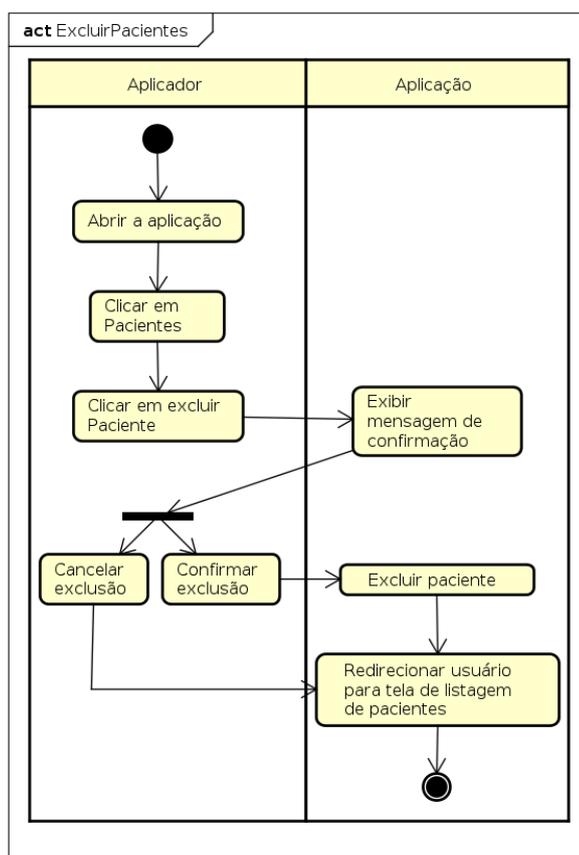
Para que/Porque: O usuário aplicador possa fazer possíveis exclusões de pacientes que possam ter sido cadastrados indevidamente.

A.4.4.2 Especificação Funcional

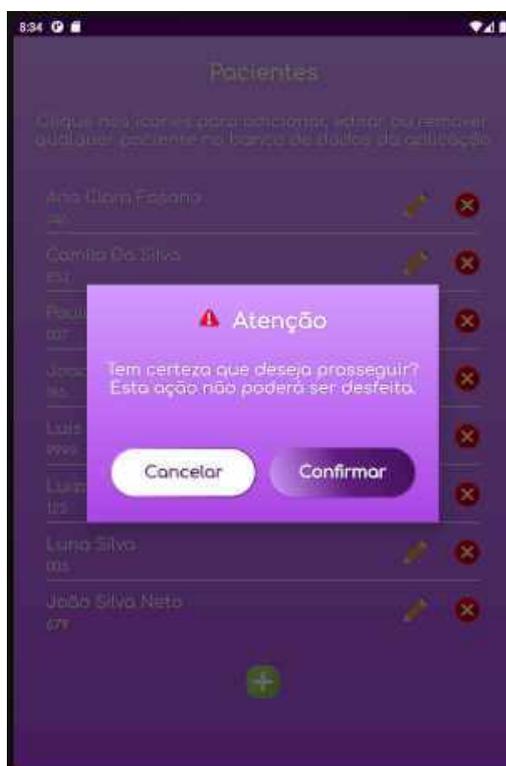
O ícone para exclusão de pacientes irá aparecer na tela de Listagem de Pacientes na frente de cada paciente, este símbolo será representado por um X envolto por um círculo vermelho, conforme especificado em RF001: Listar Pacientes.

Ao clicar no botão de exclusão a aplicação irá exibir a mensagem *"Tem certeza que deseja prosseguir? Esta ação não poderá ser desfeita."*, através de um modal, com os botões *Cancelar* e *Confirmar*.

A.4.4.3 Diagrama de atividades



A.4.4.4 Protótipo de Interface de Usuário



A.4.4.5 Pré-condições

- Usuário aplicador deverá ter o aplicativo instalado em um dispositivo móvel, com sistema operacional Android com versão maior ou igual a 5 (API nível maior ou

igual a 21).

- Usuário aplicador deverá possuir acesso à internet, para que a aplicação possa buscar e salvar as informações no banco de dados.
- Usuário deve abrir a aplicação > Clicar em Pacientes > Clicar em Excluir Paciente > Confirmar a exclusão.

A.4.4.6 Pós-condições

- Aplicação deverá ter excluído o paciente conforme solicitado pelo usuário aplicador.

A.4.4.7 Fluxo principal

1. Usuário aplicador abre a aplicação.
2. Aplicação carrega a tela inicial.
3. Usuário aplicador clica no botão Pacientes.
4. Aplicação busca as informação dos pacientes no banco de dados.
5. Aplicação exibe a tela de pacientes com as informações já cadastradas, se houver.
6. Usuário aplicador clica no botão excluir paciente, simbolizado pelo ícone de X.
7. Aplicação questiona *"Tem certeza que deseja prosseguir? Esta ação não poderá ser desfeita."*
8. Usuário aplicador clica no botão Confirmar.
9. Aplicação exclui o paciente.
10. Aplicação redireciona o usuário aplicador para tela de listagem de pacientes.

A.4.4.8 Fluxo alternativo 01

1. Usuário aplicador abre a aplicação.
2. Aplicação carrega a tela inicial.
3. Usuário aplicador clica no botão Pacientes.
4. Aplicação busca as informação dos pacientes no banco de dados.
5. Aplicação exibe a tela de pacientes com as informações já cadastradas, se houver.
6. Usuário aplicador clica no botão excluir paciente, simbolizado pelo ícone de X.

7. Aplicação questiona *"Tem certeza que deseja prosseguir? Esta ação não poderá ser desfeita."*
8. Usuário aplicador clica no botão Cancelar.
9. Aplicação redireciona o usuário aplicador para tela de listagem de pacientes.

A.4.5 RF005: Responder questionário

A.4.5.1 Estória de usuário

Caso de uso relacionado: UC005.

Como: Paciente ou Responsável.

Desejo/Preciso: Responder ao questionário com as perguntas de acordo com meu perfil (paciente ou responsável).

Para que/Porque: Possa ser extraído posteriormente os relatórios.

A.4.5.2 Especificação Funcional

Na tela inicial do aplicativo há um botão *"Iniciar questionário"* que o usuário paciente ou responsável irá clicar para ser redirecionado para as perguntas de seu respectivo questionário. Lembrando que há duas aplicações, sendo uma específica para criança e outra específica para o responsável, o aplicador deverá abrir o aplicativo de acordo com quem será entrevistado.

Caso o usuário não conclua o questionário, este ficará pendente podendo ser concluído ou excluído posteriormente (ver RF06, RF07 e RF08).

As perguntas estão cadastradas no banco de dados, podendo ser alteradas através do próprio banco de dados, posteriormente poderá ser feita uma tela para inclusão e edição de questões.

As perguntas são respondidas através da escolha de uma das seguintes opções: Sempre, Quase sempre, Às vezes, Quase nunca ou Nunca. Em alguns casos se o usuário escolher uma opção diferente de Nunca, a aplicação deve abrir um modal para que o usuário responda sub-perguntas, estas subperguntas podem ser múltipla escolha ou então abertas.

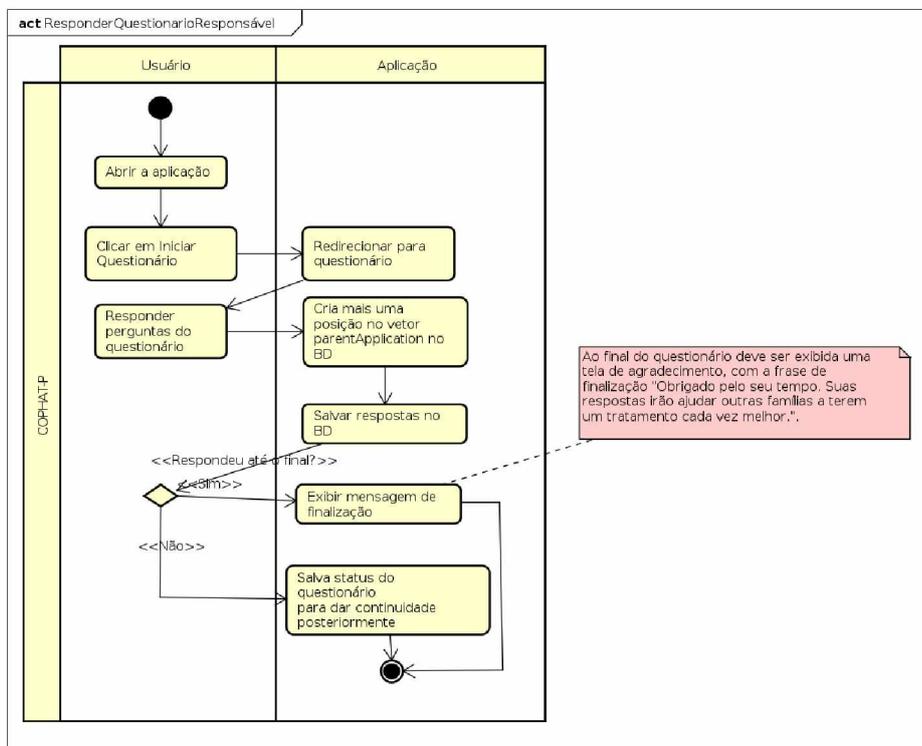
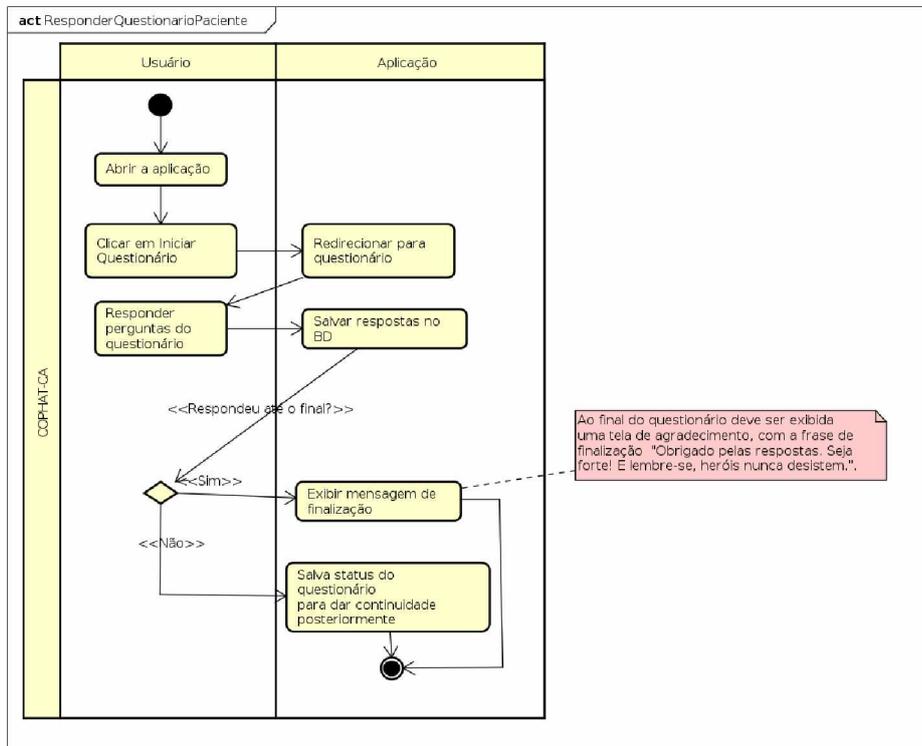
Quando o usuário respondente for o responsável a aplicação deverá armazenar no banco de dados em formato de vetor, para que seja possível o armazenamento de vários responsáveis para uma mesma criança/paciente.

Ao final do questionário deve ser exibida uma tela de agradecimento, que irá variar de acordo com o aplicativo do usuário paciente ou responsável. No caso do paciente a frase de finalização será *"Obrigado pelas respostas. Seja forte! E lembre-se, heróis nunca"*

desistem”. No caso do responsável, a frase de finalização será *“Obrigado pelo seu tempo. Suas respostas irão ajudar outras famílias a terem um tratamento cada vez melhor”*.

No topo das telas do questionário há uma barra de progresso, para que o usuário saiba em que posição do questionário ele está atualmente.

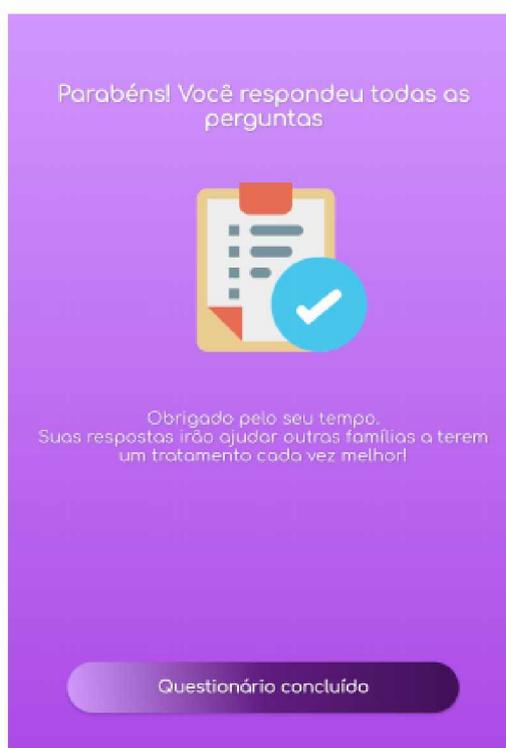
A.4.5.3 Diagrama de atividades



A.4.5.4 Protótipo de Interface de Usuário







A.4.5.5 Pré-condições

- Usuário paciente ou responsável deverá ter o aplicativo instalado em um dispositivo móvel, com sistema operacional Android com versão maior ou igual a 5 (API nível maior ou igual a 21).
- Usuário paciente ou responsável deverá possuir acesso à internet, para que a aplicação possa buscar e salvar as informações no banco de dados.
- Usuário deve abrir a aplicação > Clicar em Iniciar questionário > Responder as perguntas do questionário.

A.4.5.6 Pós-condições

- Aplicação deverá armazenar as questões respondidas pelo usuário e o status do questionário (Completo ou Incompleto).

A.4.5.7 Fluxo principal

1. Usuário paciente abre a aplicação.
2. Usuário paciente clica em "Iniciar Questionário".
3. Aplicação carrega o questionário.
4. Usuário paciente responde todo o questionário.
5. Aplicação armazena as respostas a cada pergunta respondida.
6. Aplicação exibe mensagem de finalização "Obrigado pelas respostas. Seja forte! E lembre-se, heróis nunca desistem."
7. Aplicação atualiza o status do questionário para *completed*.

A.4.5.8 Fluxo alternativo 01

1. Usuário paciente abre a aplicação.
2. Usuário paciente clica em "Iniciar Questionário".
3. Aplicação carrega o questionário.
4. Usuário paciente inicia o questionário, mas não conclui.
5. Aplicação armazena as respostas para serem concluídas posteriormente.
6. Aplicação mantém o status do questionário como *started*.

A.4.5.9 Fluxo alternativo 02

1. Usuário responsável abre a aplicação.
2. Usuário responsável clica em "Iniciar Questionário".
3. Aplicação carrega o questionário.
4. Usuário responsável responde todo o questionário.
5. Aplicação cria uma posição no vetor do *parentApplication* referente ao paciente informado.
6. Aplicação armazena as respostas a cada pergunta respondida.
7. Aplicação exibe mensagem de finalização "Obrigado pelo seu tempo. Suas respostas irão ajudar outras famílias a terem um tratamento cada vez melhor".
8. Aplicação atualiza o status do questionário para *completed*.

A.4.5.10 Fluxo alternativo 03

1. Usuário responsável abre a aplicação.
2. Usuário responsável clica em "Iniciar Questionário".
3. Aplicação carrega o questionário.
4. Usuário responsável inicia o questionário, mas não conclui.
5. Aplicação cria uma posição no vetor do *parentApplication* referente ao paciente informado.
6. Aplicação armazena as respostas para serem concluídas posteriormente.
7. Aplicação mantém o status do questionário como *started*.

A.4.6 RF006: Listar questionários incompletos

A.4.6.1 Estória de usuário

Caso de uso relacionado: UC006.

Como: Aplicador.

Desejo/Preciso: Listar os questionários incompletos.

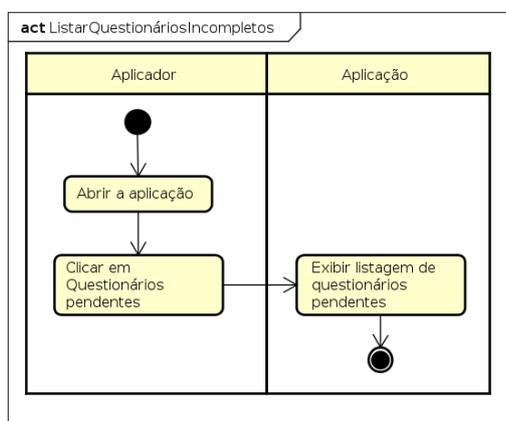
Para que/Porque: O usuário aplicador possa visualizar a lista dos questionários que não foram respondidos em sua totalidade.

A.4.6.2 Especificação Funcional

Na tela inicial do aplicativo há um botão "*Questionários pendentes*" que o usuário aplicador irá clicar para ser redirecionado para listagem dos questionários pendentes.

Nesta tela há um grid com os questionários pendentes, na linha de cada grid há um botão "Continuar", simbolizado por um lápis, e há um botão de excluir, simbolizado por um X.

A.4.6.3 Diagrama de atividades



A.4.6.4 Protótipo de Interface de Usuário



A.4.6.5 Pré-condições

- Usuário aplicador deverá ter o aplicativo instalado em um dispositivo móvel, com sistema operacional Android com versão maior ou igual a 5 (API nível maior ou igual a 21).
- Usuário aplicador deverá possuir acesso à internet, para que a aplicação possa buscar as informações no banco de dados.
- Usuário deve abrir a aplicação > Clicar em Questionários pendentes.

A.4.6.6 Pós-condições

- Aplicação deverá exibir a listagem dos questionários pendentes/incompletos.

A.4.6.7 Fluxo principal

1. Usuário aplicador abre a aplicação.
2. Usuário aplicador clica em "Questionários Pendentes".
3. Aplicação redireciona o usuário aplicador para tela de listagem de questionários pendentes.
4. Aplicação carrega os dados na tela.
5. Usuário aplicador clica no botão "Continuar" do usuário paciente/responsável correspondente.
6. Usuário paciente ou responsável continua o questionário de onde parou.

A.4.7 RF007: Continuar questionário incompleto

A.4.7.1 Estória de usuário

Caso de uso relacionado: UC007.

Como: Paciente e Responsável.

Desejo/Preciso: Continuar questionário incompleto.

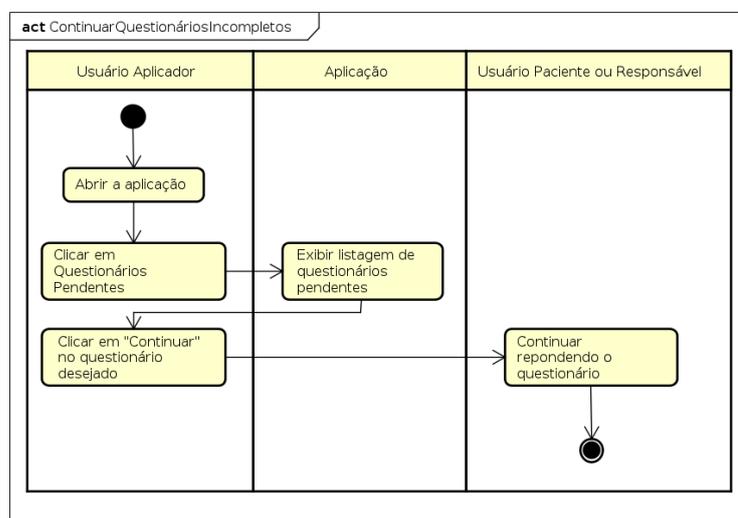
Para que/Porque: Os usuários (paciente ou responsável) possam dar continuidade à um questionário incompleto.

A.4.7.2 Especificação Funcional

Na tela de listagem de questionários pendentes, acessível através do botão "*Questionários pendentes*", disponível na tela inicial do aplicativo, o usuário clica no botão

"Continuar", simbolizado por um lápis, na sequência o usuário é redirecionado para tela do questionário posicionado na pergunta que seria a próxima a ser respondida.

A.4.7.3 Diagrama de atividades



A.4.7.4 Pré-condições

- Usuário paciente ou responsável deverá ter o aplicativo instalado em um dispositivo móvel, com sistema operacional Android com versão maior ou igual a 5 (API nível maior ou igual a 21).
- Usuário paciente ou responsável deverá possuir acesso à internet, para que a aplicação possa buscar as informações no banco de dados.
- Usuário deve abrir a aplicação > Clicar em Questionários pendentes > Clicar em continuar no questionário desejado.

A.4.7.5 Pós-condições

- Aplicação deverá exibir a tela do questionário pendente na posição que estava antes do usuário fechar a aplicação / desistir de responder ao questionário.

A.4.7.6 Fluxo principal

1. Usuário aplicador abre a aplicação.
2. Usuário aplicador clica em "Questionários Pendentes".
3. Aplicação redireciona o usuário aplicador para tela de listagem de questionários pendentes.

4. Aplicação carrega os dados na tela.

A.4.8 RF008: Excluir questionário incompleto

A.4.8.1 Estória de usuário

Caso de uso relacionado: UC008.

Como: Aplicador.

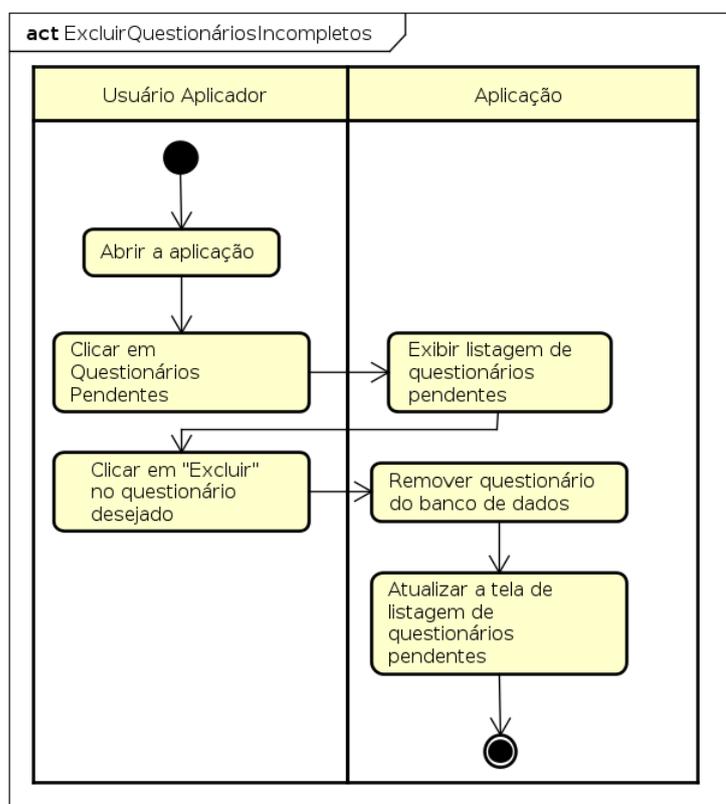
Desejo/Preciso: Excluir questionário incompleto.

Para que/Porque: O usuário aplicador possa excluir questionários que foram iniciados e não serão mais necessários.

A.4.8.2 Especificação Funcional

Na tela de listagem de questionários pendentes, acessível através do botão "Questionários pendentes", disponível na tela inicial do aplicativo, o usuário aplicador clica no botão "Excluir", simbolizado por um X.

A.4.8.3 Diagrama de atividades



A.4.8.4 Protótipo de Interface de Usuário



A.4.8.5 Pré-condições

- Usuário aplicador deverá ter o aplicativo instalado em um dispositivo móvel, com sistema operacional Android com versão maior ou igual a 5 (API nível maior ou igual a 21).
- Usuário aplicador deverá possuir acesso à internet, para que a aplicação possa buscar e salvar as informações no banco de dados.
- Usuário deve abrir a aplicação > Clicar em Questionários pendentes > Clicar em excluir no questionário desejado > Confirmar exclusão.

A.4.8.6 Pós-condições

- Aplicação deverá exibir a listagem dos questionários pendentes/incompletos atualizada, sem o questionário que foi excluído.

A.4.8.7 Fluxo principal

1. Usuário aplicador abre a aplicação.
2. Usuário aplicador clica em "Questionários Pendentes".

3. Aplicação redireciona o usuário aplicador para tela de listagem de questionários pendentes.
4. Aplicação carrega os dados na tela.
5. Usuário aplicador clica no botão "Excluir" do usuário paciente/responsável correspondente.
6. Aplicação exclui o questionário do banco de dados.
7. Aplicação recarrega a tela de listagem de questionários pendentes atualizada.

A.4.9 RF009: Listar questionários completos

A.4.9.1 Estória de usuário

Caso de uso relacionado: UC009.

Como: Aplicador.

Desejo/Preciso: Listar questionários completos.

Para que/Porque: O usuário aplicador possa visualizar todos os questionários que foram iniciados e respondidos até o final.

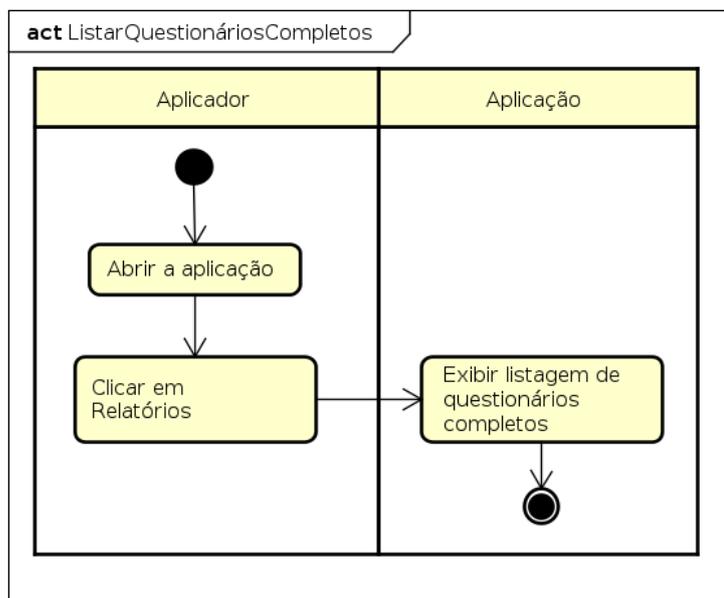
A.4.9.2 Especificação Funcional

Na tela inicial do aplicativo há um botão "*Relatórios*" que o usuário aplicador irá clicar para ser redirecionado para listagem dos questionários completos.

Nesta tela há um grid com quadrantes com as informações relacionadas ao tempo gasto para responder o questionário, desde a primeira tentativa até a conclusão, tanto da criança quanto do responsável. Quando há mais de um responsável pela criança, aparecem mais de um quadrante para aquela criança. Além disso, nos quadrantes das informações temos o hospital que a criança está internada, o nome dos aplicadores da criança e do responsável e um botão para exportar a planilha no formato *.xls* para análise do questionário.

No topo do grid há um botão *Exportar todos os questionários*, onde o usuário poderá clicar e exportar uma planilha no formato *.xls* para análise conjunta de todos os questionários respondidos por completo.

A.4.9.3 Diagrama de atividades



A.4.9.4 Protótipo de Interface de Usuário



A.4.9.5 Pré-condições

- Usuário aplicador deverá ter o aplicativo instalado em um dispositivo móvel, com sistema operacional Android com versão maior ou igual a 5 (API nível maior ou

igual a 21).

- Usuário aplicador deverá possuir acesso à internet, para que a aplicação possa buscar as informações no banco de dados.
- Usuário deve abrir a aplicação > Clicar em Relatórios.

A.4.9.6 Pós-condições

- Aplicação deverá exibir a listagem atualizada dos questionários completos.

A.4.9.7 Fluxo principal

1. Usuário aplicador abre a aplicação.
2. Usuário aplicador clica em "Relatórios".
3. Aplicação redireciona o usuário aplicador para tela de listagem de questionários completos/concluídos.
4. Aplicação carrega os dados na tela.

A.4.10 RF010: Extrair relatório de todos questionários

A.4.10.1 Estória de usuário

Caso de uso relacionado: UC010.

Como: Aplicador.

Desejo/Preciso: Extrair relatório de todos os questionários completos.

Para que/Porque: O usuário aplicador possa analisar os dados de acordo com a escala COPHAT.

A.4.10.2 Especificação Funcional

Na tela de listagem dos questionários completos, acessada através do botão "*Relatórios*" da tela inicial o usuário clica no botão *Exportar todos os questionários*, a aplicação pergunta se o usuário deseja salvar a planilha, no formato *.xls* na pasta *Android/data/-com.cophat/files/*, o usuário tem a possibilidade de escolher entre a opção de confirmar para salvar, ou cancelar para desistir do download da planilha.

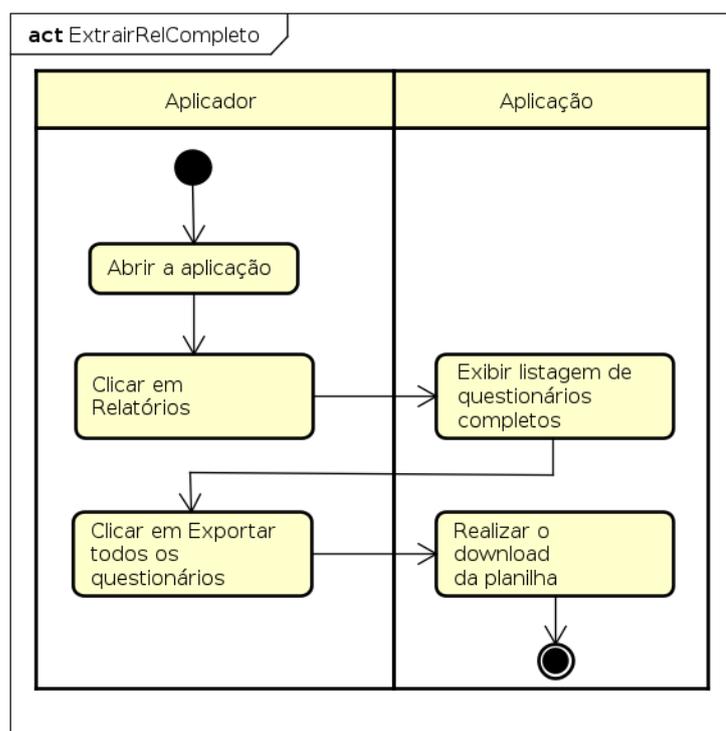
A planilha é composta por 6 abas, sendo elas:

- Questão CA: questões principais que a criança responde.
- Subquestões CA: subquestões das questões principais, que a criança responde.

- Questões P: questões principais que o responsável pela criança respondeu.
- Subquestões P: subquestões das questões principais que o responsável pela criança respondeu.
- Questões CA x P: comparação entre as respostas das questões da criança e dos pais de acordo com a categoria da questão.
- Subquestões CA x P: comparação entre as respostas das subquestões da criança e dos pais de acordo com a categoria da questão.

A criança é identificada através do código do prontuário, no cabeçalho das abas das planilhas, há o código do prontuário e o nome do responsável por preencher o questionário, seja ele a criança ou os responsáveis.

A.4.10.3 Diagrama de atividades



A.4.10.4 Protótipo de Interface de Usuário



QUESTÕES	679 - João Silva Neto	123 - Luiza Abreu
1	3	0
2	2	0
3	3	0
4	3	0
5	4	0
6	3	0
7	1	0
8	0	0
9	2	4
10	3	0
11	2	0
12	3	0
13	2	0
14	2	0
15	3	0
16	1	0
17	4	0
18	2	0
19	3	0
20	3	0
21	3	0
22	4	0

CATEGORIA	679 - João Silva Neto		123 - Luiza Abreu		123 - L
	Criança	Responsável	Criança	Responsável	Criança
Compreensão doença	0	0	0	0	0
Internação	0	0	0	0	0
Realização e êxito do tratamento	28	17	3	0	3
Efeitos colaterais	28	3	13	0	13
Expectativa do retorno à escola	0	0	0	0	0
TOTAL	56	20	16	0	16

A.4.10.5 Pré-condições

- Usuário aplicador deverá ter o aplicativo instalado em um dispositivo móvel, com sistema operacional Android com versão maior ou igual a 5 (API nível maior ou igual a 21).
- Usuário aplicador deverá possuir acesso à internet, para que a aplicação possa buscar as informações no banco de dados.
- Usuário deve abrir a aplicação > Clicar em Relatórios > Clicar no botão Exportar todos os questionários.

A.4.10.6 Pós-condições

- Aplicação deverá realizar o download da planilha dos questionários completos no dispositivo móvel.

A.4.10.7 Fluxo principal

1. Usuário aplicador abre a aplicação.
2. Usuário aplicador clica em "Relatórios".
3. Aplicação redireciona o usuário aplicador para tela de listagem de questionários completos/concluídos.
4. Aplicação carrega os dados na tela.
5. Usuário aplicador clica em Exportar todos os questionários.

6. Aplicação questiona se o usuário aplicador deseja prosseguir com o download ou cancelar.
7. Usuário aplicador clica em "Confirmar". [A001]
8. Aplicação realiza o download do arquivo no dispositivo móvel.

A.4.10.8 Fluxo Alternativo 001

1. Usuário aplicador abre a aplicação.
2. Usuário aplicador clica em "Relatórios".
3. Aplicação redireciona o usuário aplicador para tela de listagem de questionários completos/concluídos.
4. Aplicação carrega os dados na tela.
5. Usuário aplicador clica em Exportar todos os questionários.
6. Aplicação questiona se o usuário aplicador deseja prosseguir com o download ou cancelar.
7. Usuário aplicador clica em "Cancelar".
8. Aplicação volta para tela de listagem de questionários completos/concluídos.

A.4.11 RF011: Extrair relatório por paciente

A.4.11.1 Estória de usuário

Caso de uso relacionado: UC011.

Como: Aplicador.

Desejo/Preciso: Extrair relatório individual de acordo com a relação paciente x responsável escolhida.

Para que/Porque: O usuário aplicador possa analisar os dados de acordo com a escala COPHAT.

A.4.11.2 Especificação Funcional

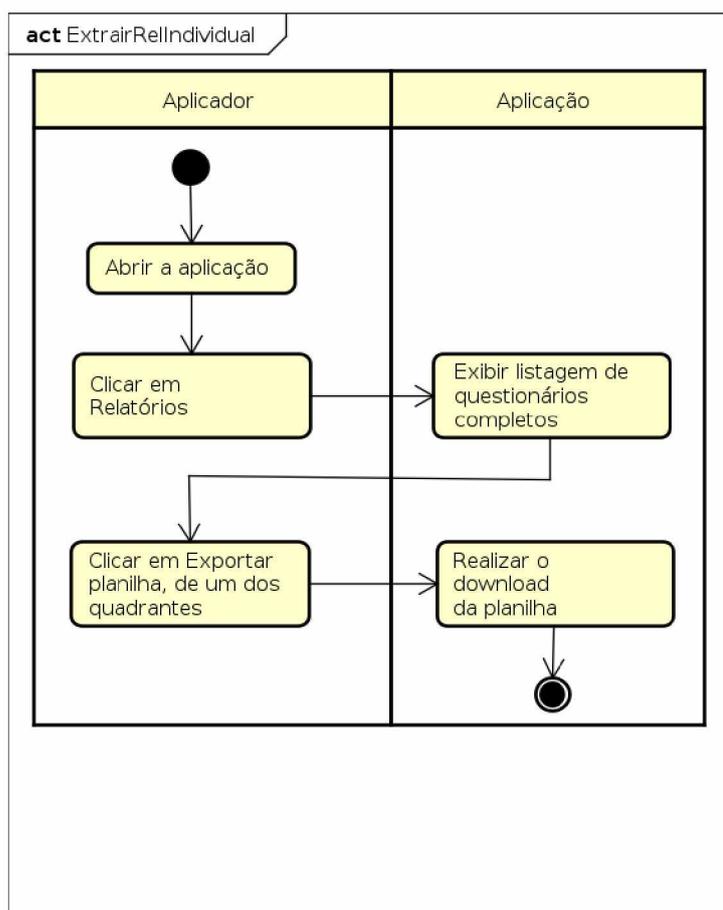
Na tela de listagem dos questionários completos, acessada através do botão "*Relatórios*" da tela inicial o usuário escolhe um quadrante de dados que deseja analisar e clica no botão *Exportar planilha*, a aplicação pergunta se o usuário deseja salvar a planilha, no formato *.xls* na pasta *Android/data/com.cophat/files/*, o usuário tem a possibilidade de escolher entre a opção de confirmar para salvar, ou cancelar para desistir do download da planilha.

A planilha é composta por 6 abas, sendo elas:

- Questão CA: questões principais que a criança responde.
- Subquestões CA: subquestões das questões principais, que a criança responde.
- Questões P: questões principais que o responsável pela criança respondeu.
- Subquestões P: subquestões das questões principais que o responsável pela criança respondeu.
- Questões CA x P: comparação entre as respostas das questões da criança e dos pais de acordo com a categoria da questão.
- Subquestões CA x P: comparação entre as respostas das subquestões da criança e dos pais de acordo com a categoria da questão.

A criança é identificada através do código do prontuário, no cabeçalho das abas das planilhas, há o código do prontuário e o nome do responsável por preencher o questionário, seja ele a criança ou os responsáveis.

A.4.11.3 Diagrama de atividades



A.4.11.4 Protótipo de Interface de Usuário



A.4.11.5 Pré-condições

- Usuário aplicador deverá ter o aplicativo instalado em um dispositivo móvel, com sistema operacional Android com versão maior ou igual a 5 (API nível maior ou igual a 21).

QUESTÕES	679 - João Silva Neto
1	3
2	2
3	3
4	3
5	4
6	3
7	1
8	0
9	2
10	3
11	2
12	3
13	2
14	2
15	3
16	1
17	4
18	2
19	3
20	3
21	3
22	4
23	2
24	3
25	4

CATEGORIA	679 - João Silva Neto	
	Criança	Responsável
Compreensão doença	15	10
Internação	18	12
Realização e êxito do tratamento	18	21
Efeitos colaterais	18	21
Expectativa do retorno à escola	17	14
TOTAL	86	78

- Usuário aplicador deverá possuir acesso à internet, para que a aplicação possa buscar as informações no banco de dados.
- Usuário deve abrir a aplicação > Clicar em Relatórios > Clicar no botão Exportar Planilha do questionário desejado.

A.4.11.6 Pós-condições

- Aplicação deverá realizar o download da planilha relacionada ao questionário escolhido, no dispositivo móvel.

A.4.11.7 Fluxo principal

1. Usuário aplicador abre a aplicação.
2. Usuário aplicador clica em "Relatórios".
3. Aplicação redireciona o usuário aplicador para tela de listagem de questionários completos/concluídos.
4. Aplicação carrega os dados na tela.
5. Usuário aplicador clica em Exportar planilha do quadrante desejado.

6. Aplicação questiona se o usuário aplicador deseja prosseguir com o download ou cancelar.
7. Usuário aplicador clica em "Confirmar". [A001]
8. Aplicação realiza o download do arquivo no dispositivo móvel.

A.4.11.8 Fluxo Alternativo 001

1. Usuário aplicador abre a aplicação.
2. Usuário aplicador clica em "Relatórios".
3. Aplicação redireciona o usuário aplicador para tela de listagem de questionários completos/concluídos.
4. Aplicação carrega os dados na tela.
5. Usuário aplicador clica em Exportar planilha do quadrante desejado.
6. Aplicação questiona se o usuário aplicador deseja prosseguir com o download ou cancelar.
7. Usuário aplicador clica em "Cancelar".
8. Aplicação volta para tela de listagem de questionários completos/concluídos.

A.4.12 RF012: Listar hospitais cadastrados (novo requisito)

A.4.12.1 Estória de usuário

Caso de uso relacionado: UC012.

Como: Aplicador.

Desejo/Preciso: Que o aplicativo liste todos os hospitais cadastrados na base de dados.

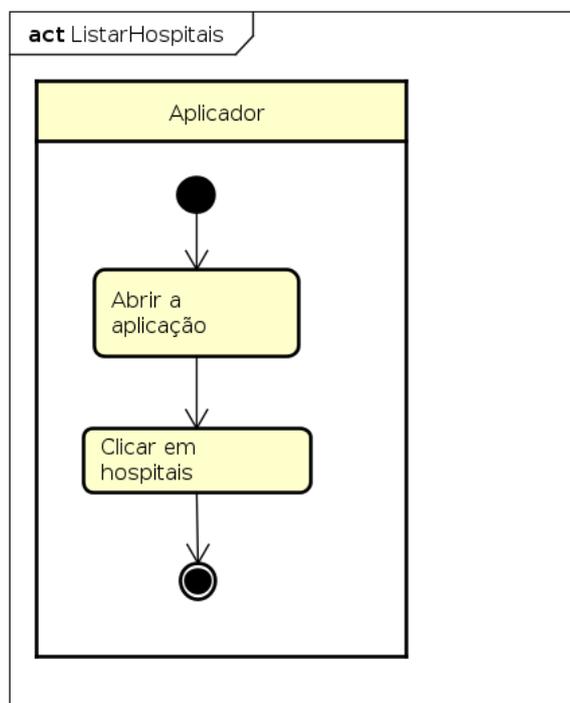
Para que/Porque: Possa realizar a conferência dos hospitais que estão cadastrados e saber quais ainda faltam ser cadastrados.

A.4.12.2 Especificação Funcional

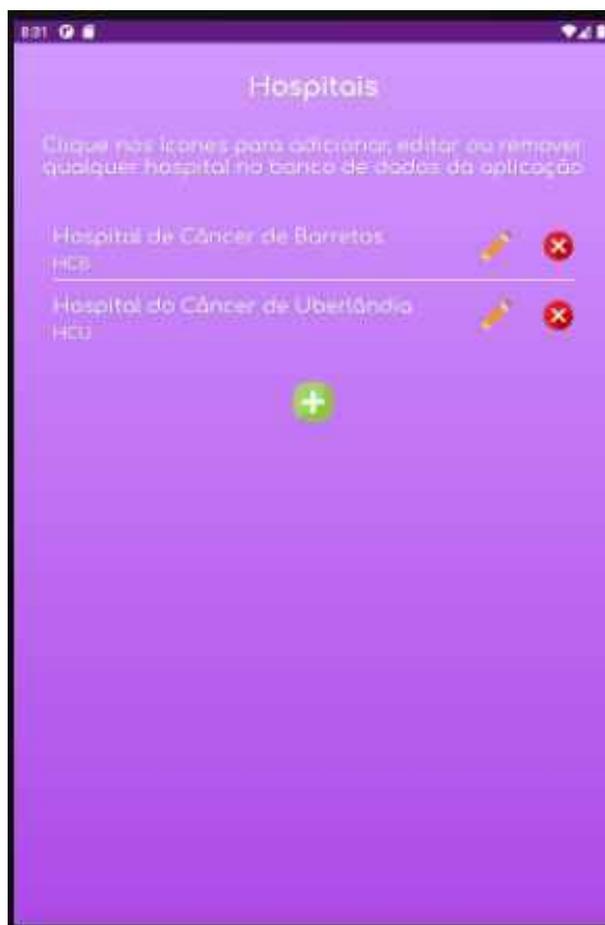
Será necessária a implementação de uma tela que irá listar todos os hospitais cadastrados na base de dados da aplicação, nesta tela irá conter um título, um texto de orientação de uso da tela e um grid com os dados dos hospitais. Em cada linha do grid irá conter o nome e o código do hospital, ao final de cada linha do grid haverá dois ícones, sendo um deles o ícone de um lápis representando a opção de editar o cadastro do hospital e o outro ícone um 'X', representando a opção de excluir um hospital.

Além disto, ao final da página, após o grid com os hospitais haverá um botão '+' para que o usuário aplicador possa incluir novos hospitais.

A.4.12.3 Diagrama de atividades



A.4.12.4 Protótipo de Interface de Usuário



A.4.12.5 Pré-condições

- Usuário aplicador deverá ter o aplicativo instalado em um dispositivo móvel, com sistema operacional Android com versão maior ou igual a 5 (API nível maior ou igual a 21).
- Usuário aplicador deverá possuir acesso à internet, para que a aplicação possa buscar as informações do banco de dados.
- Usuário deve abrir a aplicação em clicar em Hospitais.

A.4.12.6 Pós-condições

- Aplicação deverá ter aberto a tela Hospitais e, se houver, deverá ter carregado e listados as informações dos hospitais cadastrados.

A.4.12.7 Fluxo principal

1. Usuário aplicador abre a aplicação.

2. Aplicação carrega a tela inicial.
3. Usuário aplicador clica no botão Hospitais.
4. Aplicação busca as informações dos hospitais no banco de dados.
5. Aplicação exibe a tela de hospitais com as informações já cadastradas, se houver.

A.4.13 RF013: Cadastrar hospital

A.4.13.1 Estória de usuário

Caso de uso relacionado: UC013.

Como: Aplicador.

Desejo/Preciso: Cadastrar novos hospitais.

Para que/Porque: A aplicação consiga gerar os relatórios corretos referente aos questionários destes hospitais.

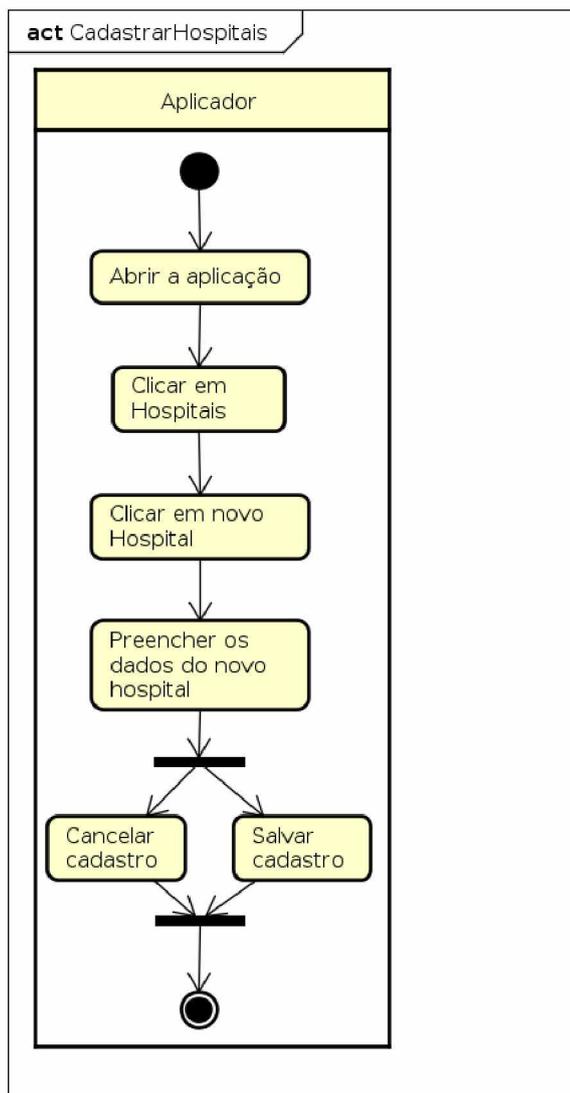
A.4.13.2 Especificação Funcional

Será necessária uma tela para que o usuário aplicador consiga cadastrar novos hospitais, esta tela deve possuir os seguintes campos:

- Nome do hospital [campo de texto]
- Código [campo de texto]

Ao final destes campos o usuário terá duas opções de botões, sendo eles "Cancelar" e "Salvar".

A.4.13.3 Diagrama de atividades



A.4.13.4 Protótipo de Interface de Usuário



A.4.13.5 Pré-condições

- Usuário aplicador deverá ter o aplicativo instalado em um dispositivo móvel, com sistema operacional Android com versão maior ou igual a 5 (API nível maior ou igual a 21).
- Usuário aplicador deverá possuir acesso à internet, para que a aplicação possa salvar as informações no banco de dados.
- Usuário deve abrir a aplicação > Clicar em Hospitais > Clicar em Novo Hospital > Preencher os dados do hospital > Clicar em Salvar.

A.4.13.6 Pós-condições

- Aplicação deverá ter salvo as informações cadastradas pelo usuário Aplicador.

A.4.13.7 Fluxo principal

1. Usuário aplicador abre a aplicação.
2. Aplicação carrega a tela inicial.
3. Usuário aplicador clica no botão Hospitais.

4. Aplicação busca as informação dos hospitais no banco de dados.
5. Aplicação exibe a tela de hospitais com as informações já cadastradas, se houver.
6. Usuário aplicador clica no botão novo hospital, simbolizado pelo ícone com '+'.
7. Aplicação redireciona o usuário aplicador para tela de cadastro de novo hospital.
8. Usuário aplicador preenche os dados cadastrais.
9. Usuário aplicador clica no botão Salvar.

A.4.13.8 Fluxo alternativo 01

1. Usuário aplicador abre a aplicação.
2. Aplicação carrega a tela inicial.
3. Usuário aplicador clica no botão Hospitais.
4. Aplicação busca as informação dos hospitais no banco de dados.
5. Aplicação exibe a tela de hospitais com as informações já cadastradas, se houver.
6. Usuário aplicador clica no botão novo hospital, simbolizado pelo ícone com '+'.
7. Aplicação redireciona o usuário aplicador para tela de cadastro de novo hospital.
8. Usuário aplicador clica no botão Cancelar.

A.4.14 RF014: Editar hospital

A.4.14.1 Estória de usuário

Caso de uso relacionado: UC014.

Como: Aplicador.

Desejo/Preciso: Que o aplicativo dê a opção de editar hospitais já cadastrados.

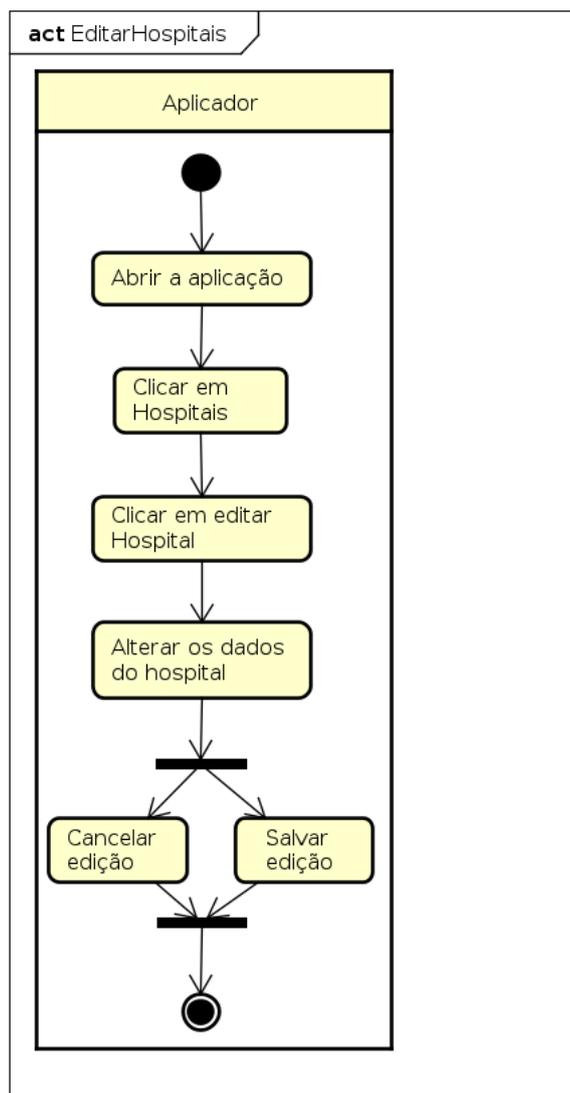
Para que/Porque: O usuário aplicador possa fazer possíveis correções cadastrais.

A.4.14.2 Especificação Funcional

Será utilizada a mesma tela que foi especificada em RF013: Cadastrar hospital, apenas o comportamento da aplicação que será diferente. Quando o usuário aplicador clicar na opção de editar hospital, na tela de Listagem de hospitais, a aplicação abrirá a tela de cadastro de hospital já preenchida com os dados do hospital selecionado anteriormente para edição.

Lembrando que o ícone para edição que irá aparecer na tela de Listagem de Hospitais é simbolizada por um lápis, conforme especificado em RF012: Listar Hospitais.

A.4.14.3 Diagrama de atividades



A.4.14.4 Pré-condições

- Usuário aplicador deverá ter o aplicativo instalado em um dispositivo móvel, com sistema operacional Android com versão maior ou igual a 5 (API nível maior ou igual a 21).
- Usuário aplicador deverá possuir acesso à internet, para que a aplicação possa buscar e salvar as informações no banco de dados.
- Usuário deve abrir a aplicação > Clicar em Hospitais > Clicar em Editar Hospital > Alterar os dados do hospital > Clicar em Salvar.

A.4.14.5 Pós-condições

- Aplicação deverá ter salvo as informações do hospital alteradas pelo usuário Aplicador.

A.4.14.6 Fluxo principal

1. Usuário aplicador abre a aplicação.
2. Aplicação carrega a tela inicial.
3. Usuário aplicador clica no botão Hospitais.
4. Aplicação busca as informação dos hospitais no banco de dados.
5. Aplicação exibe a tela de hospitais com as informações já cadastradas, se houver.
6. Usuário aplicador clica no botão editar hospital, simbolizado pelo ícone de lápis.
7. Aplicação redireciona o usuário aplicador para tela de edição de hospital.
8. Usuário aplicador altera os dados cadastrais conforme necessidade.
9. Usuário aplicador clica no botão Salvar.

A.4.14.7 Fluxo alternativo 01

1. Usuário aplicador abre a aplicação.
2. Aplicação carrega a tela inicial.
3. Usuário aplicador clica no botão Hospitais.
4. Aplicação busca as informação dos hospitais no banco de dados.
5. Aplicação exibe a tela de hospitais com as informações já cadastradas, se houver.
6. Usuário aplicador clica no botão editar hospital, simbolizado pelo ícone de lápis.
7. Aplicação redireciona o usuário aplicador para tela de edição de hospital.
8. Usuário aplicador clica no botão Cancelar.

A.4.15 RF015: Excluir hospital

A.4.15.1 Estória de usuário

Caso de uso relacionado: UC015.

Como: Aplicador.

Desejo/Preciso: Que o aplicativo dê a opção de excluir hospitais já cadastrados.

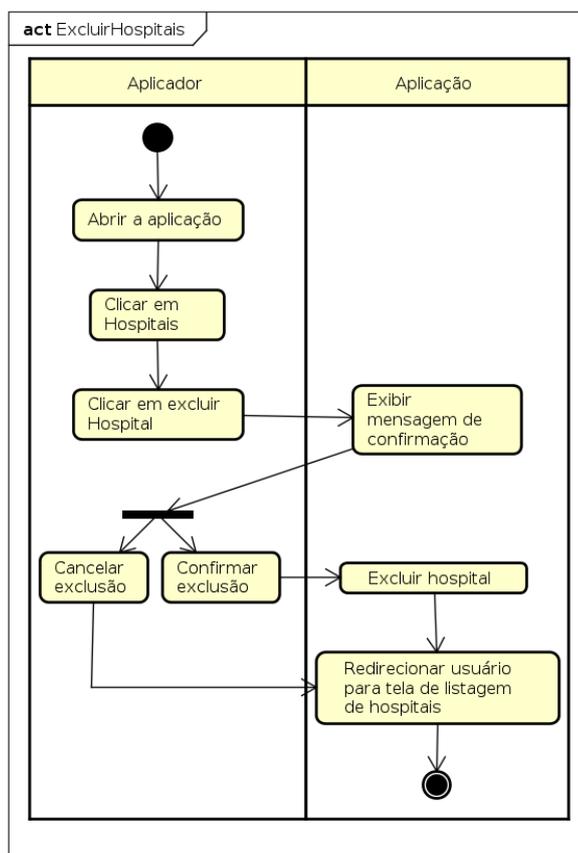
Para que/Porque: O usuário aplicador possa fazer possíveis exclusões de hospitais que possam ter sido cadastrados indevidamente.

A.4.15.2 Especificação Funcional

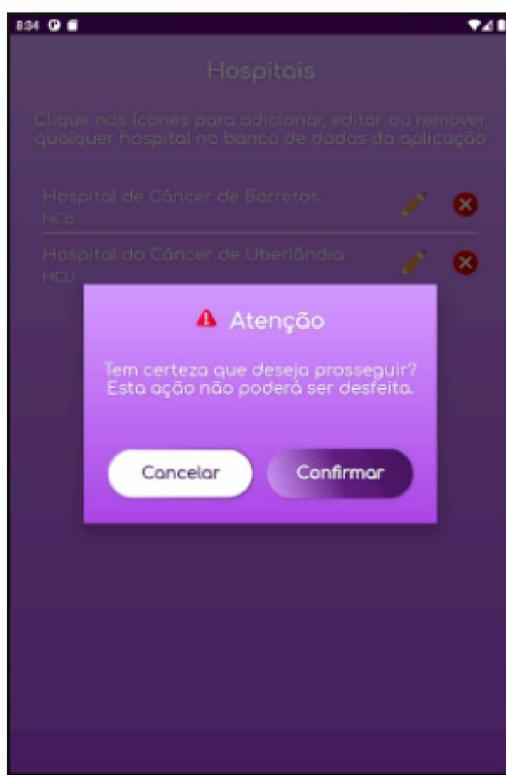
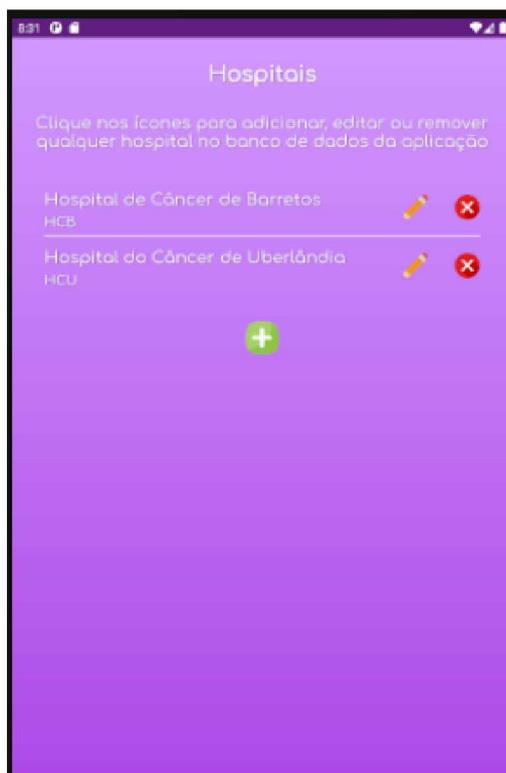
O ícone para exclusão de hospitais irá aparecer na tela de Listagem de hospitais na frente de cada hospital, este símbolo será representado por um X envolto por um círculo vermelho, conforme especificado em RF012: Listar hospitais.

Ao clicar no botão de exclusão a aplicação irá exibir a mensagem *"Tem certeza que deseja prosseguir? Esta ação não poderá ser desfeita."*, através de um modal, com os botões *Cancelar* e *Confirmar*.

A.4.15.3 Diagrama de atividades



A.4.15.4 Protótipo de Interface de Usuário



A.4.15.5 Pré-condições

- Usuário aplicador deverá ter o aplicativo instalado em um dispositivo móvel, com sistema operacional Android com versão maior ou igual a 5 (API nível maior ou igual a 21).
- Usuário aplicador deverá possuir acesso à internet, para que a aplicação possa buscar e salvar as informações no banco de dados.
- Usuário deve abrir a aplicação > Clicar em hospitais > Clicar em Excluir hospital > Confirmar a exclusão.

A.4.15.6 Pós-condições

- Aplicação deverá ter excluído o hospital conforme solicitado pelo usuário aplicador.

A.4.15.7 Fluxo principal

1. Usuário aplicador abre a aplicação.
2. Aplicação carrega a tela inicial.
3. Usuário aplicador clica no botão Hospitais.
4. Aplicação busca as informação dos hospitais no banco de dados.
5. Aplicação exibe a tela de hospitais com as informações já cadastradas, se houver.
6. Usuário aplicador clica no botão excluir hospital, simbolizado pelo ícone de X.
7. Aplicação questiona "*Tem certeza que deseja prosseguir? Esta ação não poderá ser desfeita.*".
8. Usuário aplicador clica no botão Confirmar.
9. Aplicação exclui o hospital.
10. Aplicação redireciona o usuário aplicador para tela de listagem de hospitais.

A.4.15.8 Fluxo alternativo 01

1. Usuário aplicador abre a aplicação.
2. Aplicação carrega a tela inicial.
3. Usuário aplicador clica no botão hospitais.

4. Aplicação busca as informações dos hospitais no banco de dados.
5. Aplicação exibe a tela de hospitais com as informações já cadastradas, se houver.
6. Usuário aplicador clica no botão excluir hospital, simbolizado pelo ícone de X.
7. Aplicação questiona "*Tem certeza que deseja prosseguir? Esta ação não poderá ser desfeita.*".
8. Usuário aplicador clica no botão Cancelar.
9. Aplicação redireciona o usuário aplicador para tela de listagem de hospitais.

A.5 Requisitos Não Funcionais

A.5.1 RNF001

Identificador	RNF001
Categoria	Persistência
Descrição	O banco de dados a ser utilizado para armazenamento das informações é o <i>Firebase Realtime Database</i> , que é um banco de dados não relacional, que proporciona maior flexibilidade no armazenamento dos dados.

A.5.2 RNF002

Identificador	RNF002
Categoria	Compatibilidade
Descrição	A aplicação será construída para ser executada em dispositivo móvel, com versão do sistema operacional Android a partir da versão 5 (API nível maior ou igual a 21). A interface da aplicação deve se comportar adequadamente em tablets.

A.5.3 RNF003

Identificador	RNF003
Categoria	Padrões
Descrição	A aplicação será desenvolvida utilizando o padrão de projetos arquitetural MVVM (Model View ViewModel). A linguagem a ser utilizada é o kotlin.

ANEXO A – Ficha cadastral do paciente

O presente anexo apresenta as questões que compõem a Ficha Cadastral do Paciente, necessária para a coleta das informações pessoais, familiares, sociais e econômicas do paciente. Esta ficha é parte integrante do estudo de *coping* de hospitalização para crianças e adolescentes.

1. Nome pais e profissão:
 - a) Mãe:
 - b) Pai:
2. Estado civil dos pais:
 - a) Casado
 - b) Amasiado
 - c) Divorciado/Separado
 - d) Solteiro
3. Religião da família:
 - a) Católica
 - b) Evangélica
 - c) Espírita
 - d) Outra:
4. Nome do paciente:
5. Nascimento: / /
6. Idade no dia da entrevista:
7. Sexo:
 - a) Masculino
 - b) Feminino
8. Diagnóstico:
9. Tempo de diagnóstico:
10. Número de dias internado:

-
11. Quantas internações:
12. Escolaridade:
- a) Primeiro ano
 - b) Segundo ano
 - c) Terceiro ano
 - d) Quarto ano
 - e) Quinto ano
 - f) Sexto ano
 - g) Sétimo ano
 - h) Oitavo ano
 - i) Nono ano
13. Está frequentando a escola durante a internação:
- a) Sim
 - b) Não
14. Residente nesta cidade?
- a) Sim
 - b) Não
15. Moradia (Bairro/Cidade/Estado):
16. Qual é a renda mensal de todos que contribuem com as despesas da casa?
17. Qual é o grau de instrução do chefe da família? (Considere como chefe da família a pessoa que contribui com a maior parte da renda do domicílio.)
- a) Analfabeto / Fundamental I incompleto
 - b) Fundamental I completo / Fundamental II incompleto
 - c) Fundamental completo/Médio incompleto
 - d) Médio completo/Superior incompleto
 - e) Superior completo

ANEXO B – Questionário para os pacientes

Este anexo contém o instrumento COPHAT-CA. Todas as perguntas do questionário tem como alternativas: **Nunca, Quase nunca, Às vezes, Quase sempre e Sempre.**

1. Quando eu pergunto sobre a minha doença, as respostas ainda me deixam com dúvidas.
2. Eu fico preocupado(a) quando fico sabendo que vou ser internado(a).
3. É difícil para mim receber resultados ruins de exames.

(Sub-Pergunta) - Parece que sinto:

- a) Raiva
- b) Medo
- c) Tristeza

4. Eu sinto mal estar com o tratamento.

(Sub-Pergunta) - Quais?

- a) Cansaço
- b) Falta de apetite
- c) Enjoo
- d) Dor
- e) Tontura
- f) Fraqueza
- g) Sono
- h) Irritação
- i) Dificuldade de atenção
- j) Suor
- k) Dificuldade motora
- l) Outros:

5. Eu acho que será difícil acompanhar as atividades na escola.

6. Eu tenho dúvidas sobre como funciona meu tratamento.

7. Eu tenho raiva de ficar internado(a).
8. Eu atrapalho o tratamento.
9. É difícil para mim aceitar as mudanças no meu corpo por causa do tratamento.
(Sub-Pergunta) - Quais?
 - a) Inchaço
 - b) Perda de peso
 - c) Ganho de peso
 - d) Queda de cabelo
 - e) Marcas de injeção/agulha
 - f) Manchas
 - g) Perdas de partes do corpo
 - h) Palidez
 - i) Olho amarelo
 - j) "Caroços"
 - k) Ter cateter ou fístula
 - l) Outros:
10. Eu tenho medo de que os colegas de escola riam de mim.
11. Eu não sei por que tenho que tomar remédios.
12. É insuportável ficar internado (a), mesmo podendo brincar no hospital.
13. É difícil para mim quando “furam a minha veia”.
14. Eu tenho dificuldades para dormir por causa do tratamento.
15. Eu tenho medo de me machucar na escola.
16. A equipe do hospital esconde de mim coisas sobre meu tratamento.
17. É difícil para mim ficar internado (a).
18. É difícil para mim, quando tiram meu sangue.
19. É difícil para mim não poder fazer as coisas que fazia antes do tratamento.
20. Eu tenho medo de voltar à escola.
21. Eu tenho dúvidas sobre o nome da doença que tenho.

-
22. Ficar internado (a) é ruim para mim, pois sinto falta da minha casa e das minhas coisas.
23. Eu me mexo quando vou fazer exames com aparelhos.
24. Eu tenho dificuldades em depender dos outros.
25. Eu acho que vai ser difícil conseguir fazer tudo o que o (a) professor (a) pedir.
26. Eu tenho dúvidas por que preciso fazer este tratamento.
27. Eu fico triste de ficar internado(a).
28. Eu acho que vou ficar doente para sempre.
29. Eu me sinto diferente dos outros.
- (Sub-Pergunta) - Parece que sinto:
- a) Raiva
 - b) Medo
 - c) Tristeza por causa disso
30. Eu tenho medo que meu professor (professora) não me aceite.
31. É difícil para mim pensar no futuro.
- (Sub-Pergunta) - Parece que sinto:
- a) Raiva
 - b) Medo
 - c) Tristeza
32. Eu acho ruim quando as outras pessoas perguntam sobre a minha doença/tratamento.
- (Sub-Pergunta) - Parece que sinto:
- a) Raiva
 - b) Medo
 - c) Tristeza
33. Eu fico com medo de estar (ficar) internado(a).
34. É difícil para mim ver a família preocupada comigo e a minha doença.

ANEXO C – Questionário para os pais ou responsáveis pelo paciente

O presente anexo contém o instrumento COPHAT-P. Todas as perguntas do questionário tem como alternativas: **Nunca, Quase nunca, Às vezes, Quase sempre e Sempre.**

1. Quando meu filho (a) pergunta algo sobre a doença, ele (ela) ainda permanece com dúvidas.
2. Meu filho(a) fica ansioso (a) quando vai ser internado (a).
3. É difícil para meu filho (a) receber resultados ruins de exames.

(Sub-Pergunta) - Ele (a) parece sentir:

- a) Raiva
- b) Medo
- c) Tristeza

4. Meu filho (a) se sente mal com o tratamento.

(Sub-Pergunta) - Marque os sintomas:

- a) Cansaço
- b) Falta de apetite
- c) Enjoo
- d) Dor
- e) Tontura
- f) Fraqueza
- g) Sono
- h) Irritação
- i) Dificuldade de atenção
- j) Suor
- k) Dificuldade motora
- l) Outros:

5. Meu filho (a) acha que será difícil acompanhar as atividades na escola.

6. Meu filho (a) tem dúvidas sobre como funciona seu tratamento.
7. Meu filho (a) tem raiva de ficar internado(a).
8. Meu filho (a) atrapalha o tratamento.
9. É difícil para meu filho (a) aceitar as mudanças que ocorram no seu corpo por causa do tratamento.
(Sub-Pergunta) - Quais?
 - a) Inchaço
 - b) Perda de peso
 - c) Ganho de peso
 - d) Queda de cabelo
 - e) Marcas de injeção/agulha
 - f) Manchas
 - g) Perdas de partes do corpo
 - h) Palidez
 - i) Olho amarelo
 - j) "Caroços"
 - k) Ter cateter ou fístula
 - l) Outros:
10. Meu filho (a) tem medo de que os colegas de escola riam dele (dela).
11. Meu filho (a) não sabe por que tem que tomar os remédios.
12. É insuportável para meu filho(a) suportar a internação, mesmo que ele(ela) possa brincar no hospital.
13. É difícil para meu filho (a) quando o profissional “pega a sua veia”.
14. Meu filho (a) tem dificuldades para dormir por causa do tratamento.
15. Meu filho (a) tem medo de se machucar na escola.
16. A equipe esconde do meu filho (a) o motivo do tratamento dele(a).
17. É difícil para o meu filho (a) ficar internado (a).
18. É difícil para o meu filho (a) que tirem sangue dele (a).
19. É difícil para meu filho (a) não poder fazer as coisas que fazia antes do tratamento.

20. Meu filho (a) tem medo de voltar à escola.
21. Meu filho (a) desconhece o nome de sua doença.
22. Ficar internado (a) é ruim para o meu filho (a), pois ele (ela) sente falta de casa e de suas coisas.
23. Meu filho (a) se mexe na hora de fazer os exames feitos com aparelhos.
24. Meu filho (a) tem dificuldades em depender dos outros.
25. Meu filho(a) pensa que será difícil conseguir fazer tudo que o professor (a) pedir.
26. Meu filho (a) tem dúvidas sobre o motivo de seu tratamento.
27. Meu filho (filha) fica triste por estar internado(a).
28. Meu filho (a) tem medo de continuar doente.
29. Meu filho (a) sente-se diferente dos outros.
(Sub-Pergunta) - Ele (a) parece sentir:
 - a) Raiva
 - b) Medo
 - c) Tristeza por causa disso
30. Meu filho (a) acha que seu professor (sua professora) terá dificuldades em aceitá-lo (la).
31. É difícil para meu filho (a) pensar no futuro.
(Sub-Pergunta) - Ele (a) parece sentir:
 - a) Raiva
 - b) Medo
 - c) Tristeza
32. Meu filho (a) acha ruim quando as outras pessoas perguntam sobre sua doença/tratamento.
(Sub-Pergunta) - Ele (a) parece sentir:
 - a) Raiva
 - b) Medo
 - c) Tristeza
33. Meu filho (filha) fica com medo por estar internado(a).
34. É difícil para meu filho (a) ver a família preocupada com ele (ela).