

Capítulo

5

ESPIM: um sistema para coleta de dados de usuários e intervenção programada a distância usando o método ESM e dispositivos móveis

Isabela Zaine¹, Kamila R. H. Rodrigues¹, Bruna C. R. da Cunha¹, Yuri N. Z. G. Magagnatto¹, Alex F. Orlando¹, Caio C. Viel¹, Olibário J. Machado Neto¹, André Carlomagno Rocha¹, Maria da Graça C. Pimentel¹

¹Instituto de Ciências Matemáticas e de Computação da Universidade de São Paulo (USP)

Abstract

The present course aims to introduce the Experience Sampling and Programmed Intervention Method (ESPIM), which combines selected procedures from the field of Psychology, such as Individualized Programmed Teaching and Experience Sampling Method (ESM), and the advantages promoted by Ubiquitous Computing. Our goal is to enhance ubiquitous data collection and interventions in naturalistic settings. The method is supported by a computational system - ESPIM - developed for the Web and mobile platforms. It allows the design of interventions based on both, explicit and pervasive data collection. The system supports multimedia data exchange between the stakeholders involved in planning data collection and/or intervention. We expect ESPIM to enable researchers and professionals from various fields, as health and education, to create and execute experiments or interventions carried out remotely. During the course attendants will learn about the fundamentals of the ESM and relevant results deriving from this method, data collection and interventions in naturalistic settings. Also, they will be presented to the latest version of the system ESPIM, its functionalities and computational development details. Attendants will have the opportunity to interact with the system, evaluating it in terms of usability and user experience, contributing to the improvement of the system.

Resumo

O presente curso tem o objetivo de apresentar o Método de Amostragem de Experiências e Intervenção Programada (ESPIM), que combina procedimentos

selecionados do campo da Psicologia, tais como Programação de Ensino Individualizada e Método de Amostragem de Experiências (ESM), e as vantagens proporcionadas pela Computação Ubíqua. O objetivo é ampliar o alcance da coleta de dados e intervenções a distância em ambientes naturais. O método é suportado por um sistema computacional - ESPIM - desenvolvidos para as plataformas Web e *mobile*. Ele permite que se programem intervenções baseadas em dados coletados de maneira explícita ou pervasiva. O sistema suporta a troca de dados entre as diferentes partes interessadas envolvidas no planejamento da coleta de dados e/ou intervenção. Espera-se que o ESPIM permita a pesquisadores e profissionais de diversas áreas de conhecimento, como educação e saúde, criar e executar experimentos ou intervenções conduzidos a distância. No decorrer do curso os participantes irão aprender sobre os fundamentos do ESM, bem como resultados relevantes decorrentes desse método, da coleta de dados e intervenções em ambiente natural. Será apresentada também a versão mais recente do ESPIM, suas funcionalidades e detalhes de desenvolvimento computacional. Os participantes terão a oportunidade de interagir com o sistema e avaliá-lo em termos de usabilidade e experiência de uso, contribuindo assim, para melhorias no mesmo.

5.1. Contextualização

A obtenção de informações acuradas acerca de comportamentos, sentimentos e atividades de pessoas em seu cotidiano e em ambiente natural tem sido um desafio para pesquisadores e profissionais de diversas áreas de conhecimento. O *Experience Sampling Method* - ESM [Csikszentmihaly *et al.* 1977], [Csikszentmihaly and Larson 1987] é um método da psicologia comumente utilizado para a coleta de dados acerca da experiência das pessoas em seu cotidiano e ambiente natural. Atualmente sistemas computacionais têm sido usados por pessoas de diferentes formas em seu cotidiano, de maneira inconsciente e automática, de modo ajudá-las a cumprir tarefas. Neste contexto, a Computação Ubíqua [Weiser 1991] é uma área que estuda o desenvolvimento de modelos que levam em consideração as atividades realizadas por indivíduos e os contextos em que essas atividades ocorrem [Yin *et al.* 2008], [Kapoor and Horvitz 2008]. As soluções computacionais, no contexto do método ESM, que fazem uso da computação ubíqua, permitem e facilitam o acesso a informações sobre comportamentos de interesse, bem como facilitam o planejamento de ações interventivas.

Do ponto de vista de relevância e aplicabilidade social, quanto mais pervasiva¹, contextualizada e individualizada for uma coleta de dados, potencialmente maior será o alcance de ações interventivas em diferentes domínios, uma vez que os dados previamente coletados podem ser usados para o planejamento de intervenções aplicáveis ao cotidiano e necessidades de diferentes indivíduos. Desta maneira, esta equipe procurou modelar o *Experience Sampling and Programmed Intervention Method* - ESPIM, um método que combina: a) técnicas de ESM; b) procedimentos selecionados de programação de ensino individualizada, e c) Computação Ubíqua; para aumentar a efetividade da coleta de dados. O método é apoiado por um sistema multiplataforma e, além de facilitar a coleta de dados a distância, permite o planejamento, autoria e

¹ O dicionário Cambridge associa a palavra pervasivo ao conceito de onipresente ao defini-la como presente ou visível em todas as partes de uma coisa ou lugar.

aplicação de intervenções também a distância por diferentes áreas de domínio, tais como a educação e cuidados de saúde física e mental. A seção seguinte apresenta em maior detalhes fundamentos teóricos importantes para a concepção do ESPIM.

5.2. Referencial Teórico

Esta seção apresenta o método *Experience Sampling Method* (ESM) para a coleta de dados a distância, em tempo real e em ambiente natural, bem como a sua aplicabilidade nas diversas áreas da saúde e da educação. Também serão discutidos conceitos da abordagem da psicologia da Análise do Comportamento para a programação de condições de ensino que podem ser estendidos ao planejamento e à execução de ações interventivas baseadas em dados previamente coletados.

5.2.1. Método ESM e Programação de Ensino

O *Experience Sampling Method* - ESM [Csikszentmihaly *et al.* 1977], [Csikszentmihaly and Larson 1987], [Heckner *et al.* 2007] é um método para coletar informações sobre a experiência das pessoas, no que se refere ao contexto e conteúdo, em ambiente natural. O foco nas experiências das pessoas pode ser descrito como interesse em ter acesso aos eventos ditos conscientes, tais como pensamentos, sentimentos e sensações. Esse método originou-se na abordagem psicológica da Fenomenologia Sistemática [Csikszentmihaly 2000] que combina esforços da fenomenologia pura em enfatizar a experiência como ela é percebida pelas pessoas com ferramentas de investigação de caráter empírico, tais como o uso de tecnologias, desenhos experimentais, análise estatística de dados, dentre outros [Heckner *et al.* 2007]. O método prevê que participantes recebam lembretes de maneira aleatória ou programada de acordo com intervalos de tempo ou ocorrência de eventos de interesse para que façam um autorrelato sobre o que estão fazendo ou sentindo em um determinado momento.

O método ESM tornou-se mais popular a partir da segunda metade da década de 1970, com a publicação de um dos primeiros trabalhos de ESM por Csikszentmihalyi *et al.* [1977] sobre as atividades diárias e a qualidade das experiências relatadas por adolescentes. Nesse caso específico, aparelhos de *pager* eletrônico sinalizavam aleatoriamente durante o período de uma semana momentos para que os participantes fizessem autorrelatos. Desde então, o ESM tem sido usado para obter informações sobre os mais diversos temas, entre eles: comportamento de adolescentes, depressão, estresse e satisfação no trabalho, qualidade de vida, estado de humor antes e depois de assistir TV, monitoramento de pacientes em serviços de saúde, coleta de dados sobre experiência de aprendizagem em sistemas *online*, entre outros (para uma revisão abrangente, consultar Heckner *et al.* [2007]).

Além de ser um método eficiente para coleta de dados sobre experiências cotidianas, os dados podem ser valiosos para o planejamento de programas de intervenção levando-se em consideração as particularidades da experiência, comportamento, ambiente físico e social de uma amostra de indivíduos ou de um indivíduo em particular, aumentando assim, o alcance e a efetividade da intervenção proposta.

Para ilustrar o escopo de trabalhos da literatura envolvendo ESM, a Tabela 5.1 descreve exemplos de estudos com temas e populações variadas que utilizam o método ESM com uso de algum tipo de recurso presente em dispositivos móveis.

Tabela 5.1. Exemplos de estudos que utilizam o método de ESM e dispositivos móveis.

Autores	n*	Tema da pesquisa	Recursos de captura
Floridou & Müllensiefen (2015)	40	Imaginário musical	Folheto e Celular para lembretes
Fuller-Tyszkiewicz et al. (2015)	144	Avaliação da auto-estima	Dispositivo móvel Palm com PalmOS
Hartley et al. (2014)	27	Estudo de emoções em pacientes diagnosticados com psicose	Dispositivo móvel Palm com PalmOS
Hofmann et al. (2014)	208	Estudo sobre autocontrole e dieta alimentar	Celular para lembretes e registro de respostas
Kackar et al. (2011)	331	Análise do comportamento de adolescentes durante o trabalho de casa	Celular p/ lembretes e questionários
Kramer et al. (2014)	102	Acompanhamento de tratamento de depressão	Dispositivo móvel psymate ¹
López et al. (2012)	**	Análise de atividades de diretores e chefes de escolas	Celular p/ lembretes e captura de log de atividades
Maes et al. (2015)	139	Imaginário musical	Dispositivo móvel psymate
McShane & Zirkel (2008)	12	Análise do comportamento de pessoas que sofrem de bulimia	Celular p/ lembretes e para responder emails
Mujagic et al. (2015)	26	Pacientes com Síndrome do cólon irritável	Dispositivo móvel psymate
Mylykangas et al. (2002)	78	Estudo da atividade de idosos por meio de desafios	PDAs para sinalização e feedback
Nielsen & Cleal (2011)	58	Análise da transformação de seguidores em líderes	PDAs p/ alertas e questionários
Rieh et al. (2010)	333	Avaliação da evolução de credibilidade das pessoas	Celular p/ lembretes e Interface Web para questionário
Swendeman et al. (2015a)	50	Automonitoramento de atividades de pacientes com HIV	Lembretes via email e celular
Wang et al. (2012)	28	U&G em uso de mídias sociais	Celular para lembretes e envio de SMS
Wang & Tchernev (2012)	32	Comportamento multitarefa e gratificações	Celular p/ lembretes e envio de email

* Número de participantes. ** Diretores e chefes de escolas 6 escolas municipais no Chile

A coleta de informações à respeito das experiências diárias no momento em que ocorrem é interessante pois permite que sejam traçados fluxos de eventos e relações entre eventos internos e externos. Combina, portanto, a validade ecológica das observações naturalísticas sobre comportamentos observáveis, com métodos não-intrusivos de autorrelato sobre experiências privadas. O fato dos autorrelatos serem feitos no momento de ocorrência de eventos de interesse também é interessante, pois diminui o viés da memória acerca dos acontecimentos (como esquecimento e distorções).

Outra abordagem da Psicologia, a Análise do Comportamento, também está comprometida com a qualidade da coleta de dados sobre situações de interesse. De maneira sucinta, a Análise do Comportamento é uma ciência do campo da psicologia que se baseia na filosofia do Behaviorismo Radical [Skinner 1953, 1974], que por sua vez tem o objetivo de estudar o comportamento dos organismos, entre eles, o humano. Parte-se da constatação de que existe ordem e regularidade no comportamento. Essa abordagem trata o homem sob uma perspectiva monista, ou seja, sem separação entre mente e comportamento. Os fenômenos ditos mentais por outras abordagens são entendidos como fenômenos comportamentais e dizem respeito às relações de um organismo com seu ambiente físico e social. Assim, as explicações para o comportamento são sempre buscadas nas interações funcionais entre o organismo e o ambiente. O objeto de estudo da Análise do Comportamento são os eventos que

ocorrem dentro de uma realidade natural, que podem ser acessados instrumental e empiricamente. Em decorrência, considera como objeto de estudo os comportamentos públicos observáveis e os comportamentos privados, tais como sentimentos e pensamentos.

A despeito de diferenças do ponto de vista de embasamento filosófico, as duas abordagens anteriormente apresentadas concordam em relação à importância da acuidade na coleta de dados. As informações coletadas em situações cotidianas podem ser usadas para estabelecimento de linha de base de ocorrência sobre comportamentos de interesse, definição de objetivos terapêuticos ou educacionais para a intervenção, além de usar determinadas variáveis de contexto, temporais ou autorrelatos, como gatilho, para iniciar uma intervenção planejada. Essa intervenção pode ser, por exemplo, engajamento em uma atividade educacional suplementar, administração de medicamentos, realização de exercícios de reabilitação fisioterapêutica, dentre outros. Em todos esses casos, considera-se que quanto mais informações forem obtidas sobre o cotidiano da população de interesse e sobre a percepção sobre suas experiências diárias, mais individualizado e potencialmente eficaz será o programa interventivo.

Tomando como base o exemplo de programas de intervenção educativos, psicólogos e pesquisadores da área de Análise do Comportamento têm proposto diretrizes para a programação de ensino dos mais variados comportamentos voltadas à populações com necessidades educativas especiais heterogêneas, desde comportamentos relativamente simples; como lavar as mãos e alimentar-se sozinho, a repertórios complexos; como comportamentos simbólicos de leitura e escrita. Já na década de 1950 Skinner [1958, 1961, 1972] apresenta uma proposta denominada Instrução Programada (PI – *Programmed Instruction*). Para esse autor, instruir significa instalar, alterar e eliminar comportamentos. Dessa maneira, o planejamento de ensino deve garantir ou maximizar as condições de aprendizagem de um comportamento, bem como de manutenção do que foi aprendido. Para tanto, o autor elabora algumas recomendações básicas. As principais são: 1) a programação do ensino deve ser iniciada pelo estabelecimento do repertório de entrada do aprendiz (o que ele já sabe); 2) definição de objetivos claros de ensino (o que ele deve aprender); 3) especificação detalhada do comportamento alvo do aprendiz (condições em que ele deverá ocorrer, consequências que seguem a emissão do comportamento); 4) situações aversivas e que favoreçam o erro devem ser minimizadas; 5) divisão do conteúdo total em pequenas unidades de ensino, com gradual progressão do nível de dificuldade; 6) apresentação de excelência do domínio do conteúdo atual para a progressão entre diferentes unidades de ensino; 7) fornecimento de *feedback* constante de desempenho; 8) garantia de alta densidade de reforços²; e 9) programação de contingências de reforçamento arbitrárias e naturais (passando das arbitrárias para as naturais)³.

² “Reforço” é uma consequência que segue um comportamento que tem o efeito de aumentar a probabilidade de que o mesmo comportamento (ou outro similar) ocorra novamente no futuro em uma situação semelhante. No contexto leigo, ele pode ser entendido como “recompensa”.

³ Consequências arbitrárias são aquelas que são artificialmente estabelecidas, não havendo uma relação direta com as respostas que as produzem. Já as consequências naturais são aquelas decorrentes da própria atividade em questão, mantendo uma relação direta com as respostas que o produzem. Por exemplo, ler um livro porque ele será matéria de prova trata-se de uma consequência arbitrária, enquanto que ler um livro porque a leitura permite conhecer o conteúdo trata-se de uma consequência natural.

Nesse contexto, Skinner propõe máquinas de ensinar, que seriam dispositivos de apresentação de informações organizadas em uma sequência programada, com a principal característica do reforço imediato de respostas corretas.

De maneira semelhante, Keller, ao final da década de 1960, elabora o Sistema Personalizado de Ensino (PSI – *Personalized System of Instruction*), cujas principais características eram: 1) aprendizagem de acordo com o ritmo de cada aprendiz; 2) a exigência de domínio completo de uma unidade anterior antes de passar para a próxima fase; 3) o uso de demonstrações e palestras como meio de motivação; 4) a ênfase na palavra escrita mediando a comunicação entre professor e o aluno; e 5) o uso de monitores, que permitem a repetição de testes, *feedback* imediato ao aprendiz e tutoria acessível [Keller 1968]. Apesar de existir um foco da área em contextos educativos, a programação de ensino individualizada apresenta diretrizes importantes que podem ser aplicadas a diferentes cenários interventivos, uma vez que quase sempre intervenções envolvem a aprendizagem de algum tipo de repertório novo.

A princípio tanto as pesquisas envolvendo o ESM quanto as diferentes aplicações da área de programação de ensino individualizada eram desenvolvidas manualmente, com materiais impressos. No caso do ESM, comumente se utilizavam formulários ou diários em papel para realização de registros e dispositivos de *paggers* ou mensagens de texto (SMS) em aparelhos celulares para sinalizar o momento em que os autorrelatos deveriam ser feitos [Riddle and Arnold 2007]. Tal prática ainda é comum em estudos recentes, como no trabalho de Udachina *et al.* [2014].

Atualmente, dispositivos móveis como *smartphones* e *tablets* são amplamente populares, acessíveis à população e contam com uma série de recursos como câmeras de alta qualidade, microfone, ferramentas de gravação de voz e editores de texto, GPS para localização e sensores como acelerômetro; giroscópio; detecção de luz ambiente; detecção de proximidade, entre outros. Esse tipo de tecnologia permite grande flexibilização do método de amostragem de experiências [Baxter *et al.* 2015], podendo os usuários serem imediatamente sinalizados para realizar relatos de experiência de diversas formas (como por meio de texto, vídeo, fotografia) e envio de dados imediatamente ao pesquisador. Além disso, o uso de dispositivos móveis possibilita que dados sejam coletados sem a presença direta do experimentador, permitindo a coleta de informações por mais tempo sem a necessidade de deslocamento por parte do experimentador e dos participantes. Outra vantagem no uso de dispositivos móveis é a possibilidade de interferir minimamente nas atividades cotidianas dos participantes e de não provocar, ou pelo menos minimizar, a sensação de estarem sendo observados. Por fim, a adoção dos dispositivos móveis neste contexto aplicado tem potencial para aumentar a probabilidade de fornecimento de informações, bem como a adesão à determinadas ações de intervenções. Tais dispositivos também podem servir como meio para a realização total ou parcial da coleta de dados ou intervenções a distância. Por exemplo, indivíduos acompanhados por profissionais de saúde ou educação podem ser alertados durante o dia para administrar medicamentos, fazer exercícios de reabilitação fisioterapêutica, de estimulação cognitiva ou atividades escolares complementares. O próprio dispositivo pode servir como meio para o registro da atividade (por exemplo, texto, vídeo) e a interação desses usuários com o sistema pode ser imediatamente enviada aos profissionais responsáveis pelas intervenções para que esses façam as análises cabíveis. Novas ações do procedimento interventivo — baseadas nas informações enviadas pelos participantes — podem ser planejadas e implementadas rapidamente e de maneira individualizada.

Para exemplificar, a situação ilustrada a seguir em *storyboards* pela Figura 5.1 descreve um contexto de interação em que informações estão sendo coletadas em tempo real utilizando *smartphones*. O cenário apresentado é o de acompanhamento da rotina e de realização de atividades cotidianas por um indivíduo idoso. A interação ocorre em um ambiente familiar entre o idoso e seu neto. Nas cenas, o idoso recebe uma notificação em seu *smartphone* que desencadeia uma sequência de ações, entre elas: tirar uma *selfie* com o neto, divulgar a foto e responder à questões relacionados ao contexto em que a foto foi tirada.

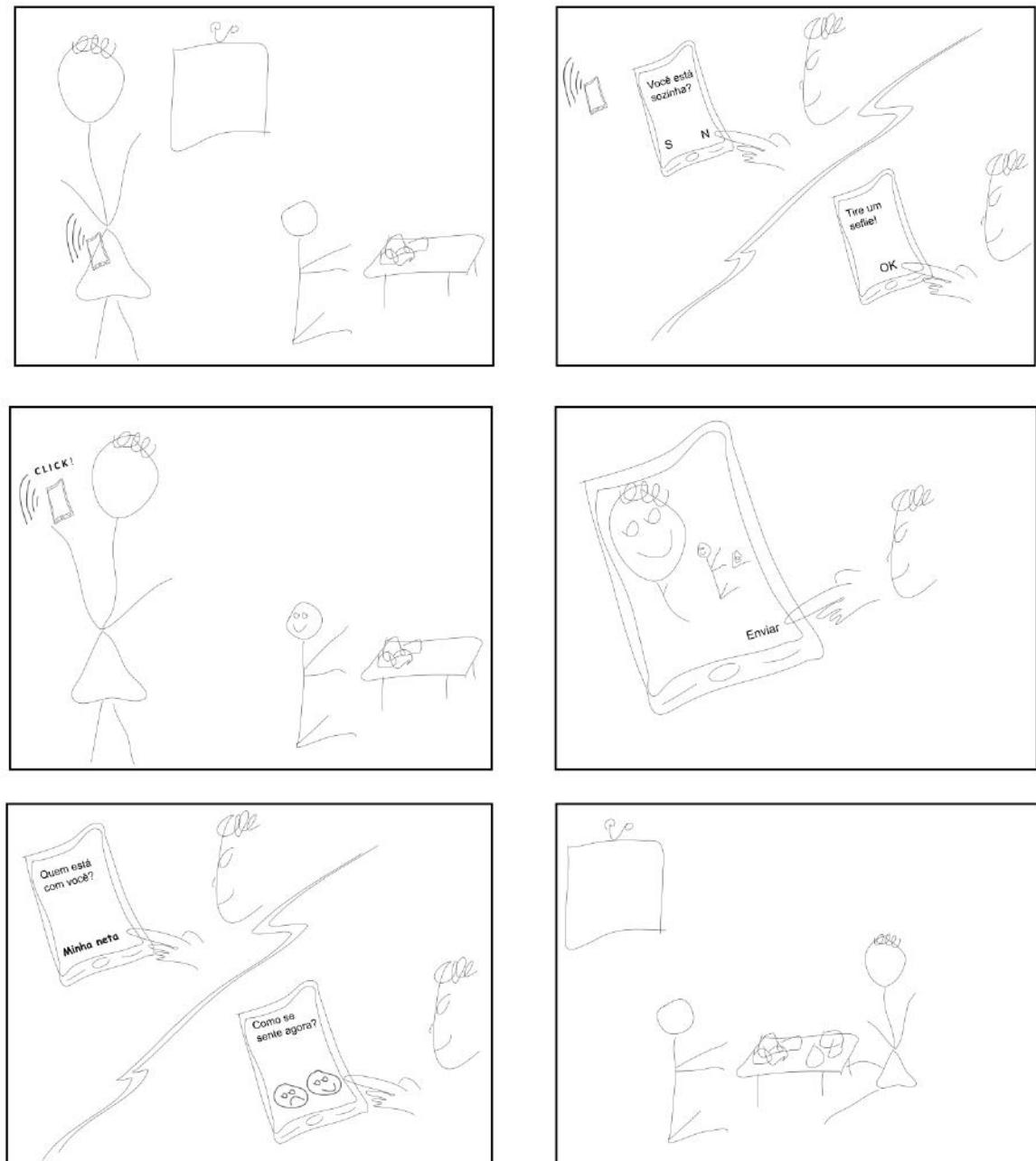


Figura 5.1. *Storyboards* que descrevem um cenário de coleta de dados em tempo real utilizando *smartphones*.

Outras vantagens do uso de Tecnologias da Informação e Comunicação (TICS) aplicáveis a dispositivos móveis para a realização de estudos envolvendo o método de ESM incluem: 1) controle preciso de tempo; 2) acompanhamento da participação efetiva dos usuários; 3) obtenção de dados auxiliares de avaliação de comportamentos e; 4) redução de erro humano. No primeiro caso, o experimentador pode programar o envio de alertas em intervalos de tempo fixo, variável ou restrito e especificar um intervalo de tempo para que o participante responda ao alerta. No segundo caso, pode-se ter um rastreamento de participação efetiva e de alertas que foram “perdidos” ou não atendidos, dessa forma, o pesquisador pode acompanhar a motivação dos participantes em atenderem aos alertas, e modificar a forma ou frequência dos mesmos quando necessário. Com relação à obtenção de dados auxiliares, pode-se programar para que o sistema registre informações complementares ao conteúdo, como a latência de resposta entre diferentes questões. Finalmente, os dados coletados podem ser facilmente registrados em formatos compatíveis com softwares estatísticos automaticamente importados para análise, reduzindo a ocorrência de erros humanos [Barrett and Barrett 2001].

No caso das pesquisas em programação de ensino, os primeiros programas eram impressos em folhas de papel sequencialmente organizadas em pastas do tipo fichário e o registro de desempenho feito à mão, pelo preenchimento de protocolos de registro [De Rose *et al.* 1989]. Da mesma forma que o advento dos dispositivos móveis aumentou o alcance das pesquisas de ESM, a popularização dos computadores pessoais e avanço das TICs permitiu maior controle de variáveis, rapidez e flexibilidade de programas de ensino individualizados, de modo que os computadores passaram a ser utilizados para autoria e aplicação de programas de ensino cada vez mais complexos, bem como para o registro do desempenho dos aprendizes [De Souza *et al.* 2009]. Também a programação de ensino individualizada tem se beneficiado dos dispositivos móveis, sendo comum a criação de aplicativos simples para o ensino de variados repertórios comportamentais, como relações de igualdade ou desigualdade entre estímulos, reconhecimento de expressões faciais e emoções, formação de conceitos, dentre outros.

A presença cada vez mais intrínseca de diferentes soluções de software, desenvolvidas para a computação cada vez mais presente no cotidiano das pessoas, podendo ser elas um sistema, um aplicativo ou um serviço, pode e deve ser aproveitada com objetivos de pesquisa e/ou intervenções clínicas.

5.2.2. Computação Ubíqua

A adoção de sistemas computacionais no cotidiano das pessoas é objeto de estudo da área de Computação Ubíqua, que busca criar soluções para viabilizar o conceito da ubiquidade. Weiser [1991], um dos principais autores e fundadores da área, aponta que as tecnologias mais profundas são aquelas que tendem a “desaparecer”, a ponto de serem indistinguíveis do ambiente. Para definir esse conceito, Weiser utilizou a palavra *ubiquitous* ou ubíquo. Segundo o dicionário Cambridge, a palavra ubíquo significa encontrado ou existente em toda a parte. Logo, ser ubíquo significa ser onipresente. Além da onipresença, Weiser também associou o “desaparecimento” dos computadores ao significado da palavra ubíquo. Dessa maneira, para ser ubíquo, segundo a definição de Weiser, os sistemas devem ser onipresentes e proporcionarem transparência de uso aos usuários.

Atualmente a ubiquidade tem estado cada vez mais enraizada no cotidiano das pessoas, de modo que essas usam naturalmente sistemas computacionais em seu dia-a-dia para ajudarem-nas a cumprir tarefas cotidianas de maneira inconsciente e automática.

Ainda considerando este contexto, são encontrados na literatura trabalhos que fazem referência à Computação Pervasiva. A palavra *pervasivo*, por si só, não define a ubiquidade descrita por Weiser, entretanto, Satyanarayanan [2001] define a computação *pervasiva* como outra forma de se referir à computação *ubíqua*. Araújo [2003] por sua vez, aponta que a computação *ubíqua* e a computação *pervasiva* são definidas pelo grau de “embarcamento”⁴ e de mobilidade de cada uma. De acordo com Araújo, para ser *ubíqua*, a solução de software deve ter um alto nível de embarcamento e de mobilidade, enquanto que o conceito de *pervasivo* envolve apenas um alto nível de embarcamento. Essa definição é apoiada por Lyytinen e Yoo [2002], que se referem à computação *ubíqua* como a possibilidade de qualquer dispositivo configurar os serviços do ambiente em que está, mesmo enquanto se move com o usuário.

Um dos principais objetivos da computação *ubíqua* é o desenvolvimento de modelos que levam em conta as atividades realizadas por indivíduos e os contextos em que ocorrem [Yin *et al.* 2008], [Kapoor and Horvitz 2008], o que torna soluções de computação *ubíqua* grandes facilitadoras não apenas ao acesso de informações sobre comportamentos de interesse, mas também de ações interventivas.

Uma vez que o acesso à informações atualmente é cada vez mais facilitado pela computação *ubíqua*, este grupo de pesquisadores viu a oportunidade de unir o acesso a dados de interesse, por meio de técnicas baseadas em procedimentos selecionados do método de ESM, para implementar planos de intervenções programadas — de modo que tanto a coleta de informações quanto a intervenção planejada possam fazer uso de dispositivos móveis, vestíveis ou integrados a ambientes inteligentes, por exemplo. Acreditamos que quanto mais contextualizadas e individualizadas sejam as ações interventivas, essas serão mais relevantes, adequadas e aplicáveis ao cotidiano e necessidades de diferentes indivíduos, aumentando assim, as chances de sucesso das intervenções propostas.

5.3. O método ESPIM

Como anteriormente mencionado, o ESPIM - *Experience Sampling and Programmed Intervention Method* - combina técnicas de ESM, procedimentos selecionados de programação de ensino individualizada e a computação *ubíqua*, com o objetivo de aumentar a efetividade e otimizar a coleta de dados, bem como o planejamento, autoria e aplicação de intervenções a distância programadas por diferentes áreas de domínio, em especial saúde e educação.

Atualmente o método ESPIM tem o suporte de um sistema para a plataforma Web, que trata da interface com o especialista, e de um aplicativo para a plataforma *mobile*, que permite a coleta de dados e intervenções a distância. Este sistema é uma ampliação a um sistema anterior de uma plataforma intitulada *SmartESM*, destinada em

⁴ O termo “embarcamento” é utilizado por Araújo [2003] para se referir ao grau de inteligência dos computadores, embutidos em um ambiente *pervasivo*, para detectar, explorar e construir dinamicamente modelos de seus ambientes.

especial, ao envio de alertas para responder a questionários, contando com uma aplicação Web de interface gráfica que permite visualizar, adicionar, editar e gerenciar elementos relevantes (e.g. participantes, pesquisadores, experimentos, questionários, lembretes, alertas etc.), e com um aplicativo móvel desenvolvido na plataforma Android, para ser usado em *smartphones* [Pimentel *et al.* 2016].

O sistema ESPIM está sendo desenvolvido utilizando uma abordagem participativa e de coprodução [Schuler and Namioka 1993], em que profissionais de diferentes áreas, entre elas computação, saúde e educação, colaboram na elicitação de requisitos e também na avaliação do sistema.

A adoção da abordagem participativa corrobora com princípios do Design Centrado no Usuário (DCU) [Lowdermilk 2013], cujo foco está nas necessidades, desejos e limitações dos usuários durante todo o projeto, a cada tomada de decisão, desde a concepção até o lançamento do produto. O envolvimento com usuários alvo permite identificar a necessidade real desses. O DCU possui quatro etapas básicas:

1. *Identificar requisitos*: etapa de clarificar problemas, levantar necessidades e entender os pontos de conflitos dos usuários através de pesquisas, observações e entrevistas;
2. *Criar soluções alternativas*: etapa de explorar ideias e sugestões, momento em que são levantadas hipóteses de soluções para as necessidades levantadas;
3. *Construir protótipos testáveis*: etapa em que são criados modelos testáveis do que pode vir a ser o produto final;
4. *Avaliar com usuários*: etapa em que os protótipos são levados para testes com usuários e *feedbacks* sobre o que funciona e o que pode melhorar são coletados.

Este minicurso apresenta para a comunidade do Simpósio Brasileiro de Sistemas Multimídia e Web (*WebMedia*) o método ESM e o ESPIM. O conteúdo será abordado de maneira teórica e também prática. Na etapa teórica serão apresentados o método e a sua aplicabilidade nas diversas áreas de conhecimento. A coleta de dados de maneira ubíqua sobre as experiências dos indivíduos – sendo realizada em tempo real e em ambiente natural – bem como o planejamento e a implementação de intervenções baseada em dados previamente coletados, também serão abordados nessa etapa.

Na sequência, a infraestrutura tecnológica utilizada no desenvolvimento do ESPIM e os seus componentes serão detalhados. Os componentes do ESPIM incluem: um aplicativo móvel a ser utilizado por usuários que estão sendo acompanhados; um sistema Web para o planejamento das coletas e intervenções, a ser utilizado por pesquisadores e profissionais e, um *Web Service* responsável pelo armazenamento das informações. Esses componentes são descritos na Seção 5.4.

Ainda na etapa teórica serão apresentadas as funcionalidades do ESPIM e atividades que cada perfil de usuário pode realizar, entre as atividades: a criação de programas personalizados para coleta de dados e cadastro de participantes.

Na etapa prática do curso cada participante poderá interagir com o ESPIM criando experimentos/intervenções, observando como a coleta de dados pode ser feita e como os participantes são acompanhados. Espera-se que *feedbacks* sobre o sistema possam ser coletados nessa etapa, principalmente no que diz respeito a aspectos relacionados à usabilidade e à experiência de uso.

5.3.1. Exemplo de cenário de uso do ESPIM

Esta seção se dedica à apresentação em detalhes de um possível cenário para o uso do ESPIM, tanto para a coleta de dados e de experiências dos usuários, como para servir de gatilho para o início de atividades de intervenção programadas por especialistas de área. O cenário retomará alguns aspectos apresentados anteriormente pela Figura 1, em um contexto que envolve idosos.

Cenário Exemplo - Acompanhamento e avaliação de desempenhos cognitivos de idosos

O senhor Josias, 83 anos, foi diagnosticado clinicamente com Doença de Alzheimer, apresentando indícios de perdas cognitivas, tais como confusão mental, prejuízos em adquirir e relembrar novas informações e dificuldade em reconhecer pessoas em seu convívio diário. Dessa maneira, o geriatra responsável pelos cuidados do senhor Josias elaborou uma série de atividades para que o mesmo se engajasse, utilizando seu *smartphone*, durante o período de uma semana. As tarefas tinham os objetivos de: 1) Coletar dados acerca da rotina de Josias; 2) Coletar informações acerca dos sentimentos (experiências) de Josias ao longo da semana e, 3) Aumentar a probabilidade de que Josias realizasse atividades de estimulação cognitiva. O fluxo de atividades tanto de coleta de dados quanto de intervenções programadas pelo geriatra é apresentado pela Figura 5.2.

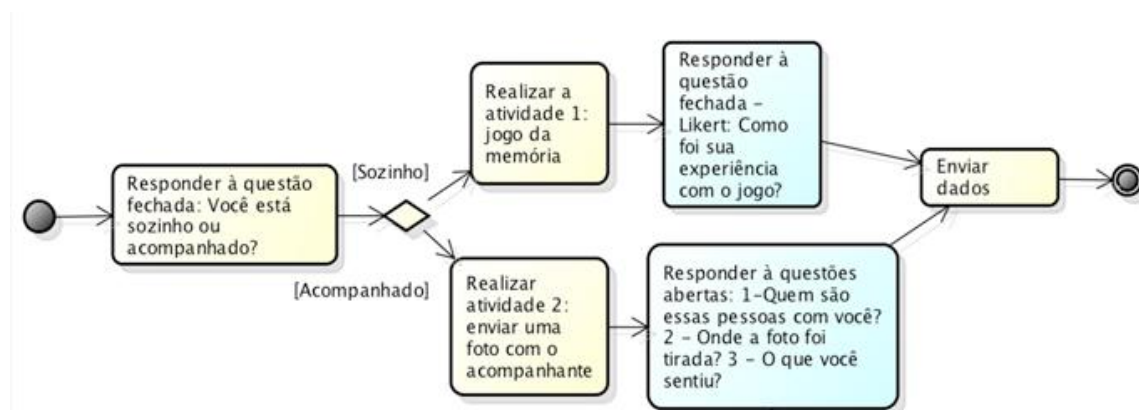


Figura 5.2. Fluxograma de encadeamento das atividades planejadas para o caso de acompanhamento de idosos.

O geriatra considerou apropriado que fossem feitas interações diárias com o sistema, 3 vezes ao dia, uma no período da manhã (das 8h às 12h), uma no período da tarde (das 12:30h às 17:30h) e outra no período da noite (das 18h às 21:30h). Assim, o sistema irá mandar três alertas diários para o início da interação, por um período de 7 dias consecutivos. Observa-se que o gatilho inicial é o recebimento de um alerta no *smartphone* para que o senhor Josias responda à seguinte pergunta de texto: “Você está sozinho ou acompanhado?”. Essa pergunta será respondida por meio da seleção de uma das possibilidades de resposta em uma questão do tipo múltipla escolha com uma opção de escolha, em que as opções são: () sozinho; () acompanhado. O geriatra tem planos distintos de acordo com respostas diferentes, assim, a continuidade de interação com o sistema poderá tomar dois caminhos: se a resposta à questão anterior for “sozinho”, será enviado imediatamente uma solicitação para que o senhor Josias inicie uma atividade de *Jogo de memória*. Uma instrução é exibida na tela do *smartphone*:

“Vamos jogar o jogo da memória!”, seguido de um botão “Início”. No momento em que o idoso seleciona o botão, o jogo (atrelado previamente pelo geriatra a essa intervenção) é iniciado. O sistema mantém o registro automático de informações à respeito da interação do usuário com o jogo. Entre os registros coletados estão: a quantidade de erros/acertos e o tempo gasto na realização da atividade.

Assim que a atividade do *Jogo da Memória* for encerrada, uma mensagem de *feedback* é enviada ao senhor Josias, por exemplo, “Muito bem! Você terminou o jogo!”. Em seguida, o sistema imediatamente inicia uma nova interação, solicitando que o senhor Josias relate o grau de satisfação ao jogar o jogo. O sistema apresenta a seguinte pergunta: “Como foi sua experiência com o jogo?”. O senhor Josias irá selecionar sua resposta, em uma escala analógica do tipo Likert com cinco opções: 0 - *detestei*; 1 - *não gostei*; 2 - *neutro*; 3: *gostei*; e 4: *adorei*. A resposta a essa pergunta encerra o episódio interativo e os dados coletados são enviados ao servidor Web do ESPIM. Uma nova interação com o sistema irá ocorrer novamente (ou não) de acordo com os critérios pré-estabelecidos pelo especialista de área, neste caso, o geriatra.

Agora consideremos novamente a pergunta inicial que foi o gatilho para esta interação. Caso a resposta do senhor Josias seja “acompanhado”, o geriatra programou a interação de outras diferentes atividades. Ele deverá tirar uma foto com o acompanhante e enviá-la. O sistema irá solicitar que Josias faça isso por meio da instrução de voz: “Legal, vamos tirar uma foto com seu acompanhante”. O aplicativo aciona então a câmera do dispositivo, a foto é tirada e enviada ao servidor. Ao final do envio da foto, o aplicativo apresentará uma mensagem de *feedback*, por exemplo: “OK! Foto enviada com sucesso!”. O fim dessa atividade marca o encadeamento para a atividade seguinte em que o Josias irá responder a três perguntas abertas, por meio da gravação de áudio, por exemplo: a) “Quem são essas pessoas com você?”; b) Onde a foto foi tirada?; “O que você sentiu?”. Responder à essas três questões encerra o episódio interativo e os dados coletados são enviados ao servidor Web do ESPIM para análise do especialista.

A seção a seguir descreve como o sistema ESPIM foi desenvolvido considerando questões relacionadas aos seus componentes, bem como as linguagens de programação e *frameworks* utilizados no desenvolvimento desses componentes.

5.4. Desenvolvimento do sistema ESPIM

Conforme exposto, o sistema ESPIM é multiplataforma (Web e *mobile*) e tem sido desenvolvido com o apoio de uma equipe multidisciplinar (profissionais da computação, saúde e educação). O sistema deve apoiar pesquisadores que utilizam o ESM e servir de laboratório para pesquisas associadas ao método.

As subseções a seguir detalham a arquitetura do sistema ESPIM e os seus componentes.

5.4.1. Infraestrutura Tecnológica

A Figura 5.3 ilustra a arquitetura de alto nível proposta para a implementação do sistema ESPIM. As definições de um programa – definido por um especialista – e os dados de resultados, coletados via intervenção com os usuários acompanhados, serão armazenados em um servidor Web. Para armazenar e acessar esses dados uma arquitetura REST é utilizada, de forma que os dados são obtidos e enviados via HTTP.

O cliente, neste caso um aplicativo móvel ou uma página da Web, não tem acesso direto à modelagem ou aos dados do programa.

Para facilitar a implementação de clientes que precisam se comunicar com o serviço Web, foi desenvolvida uma biblioteca em Java que simplifica as chamadas REST para o *Web Service*. A biblioteca de monitoramento simplifica todas as operações e converte os documentos JSON em objetos Java. O uso da biblioteca não é obrigatório, de modo que é possível desenvolver clientes Web e móveis utilizando diferentes linguagens de programação, independente da linguagem utilizada no *Web Service* e na biblioteca.

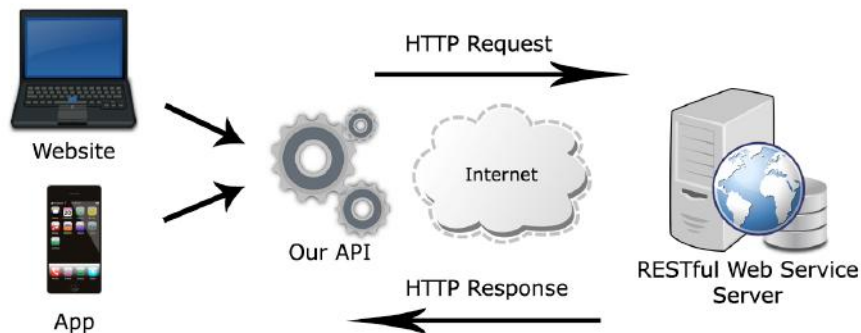


Figura 5.3. Arquitetura de alto nível proposta para o sistema ESPIM.

Os componentes básicos do sistema ESPIM foram desenvolvidos com base na arquitetura da Figura 5.3. Os componentes são:

- 1) Um *Web Service RESTful*, desenvolvido em Java com o *framework Spring Boot*, responsável pelo armazenamento das informações;
- 2) Um sistema Web, desenvolvido com as linguagens HTML5 e JavaScript para o planejamento das coletas e intervenções, a ser utilizado por usuários especialistas e;
- 3) Um aplicativo móvel desenvolvido em Android a ser utilizado pelos usuários que estão sendo acompanhados por especialistas.

Esses componentes são descritos em detalhes nas Seções 5.4.1.1, 5.4.1.2 e 5.4.1.3, respectivamente.

Além dos três componentes apontados acima, foram desenvolvidos componentes para acessar e controlar sensores de dispositivos móveis que podem ser utilizados na construção de intervenções que fazem uso de sensores para coleta de dados específicos, tais como a localização via GPS, sensor de luminosidades, etc. Tais componentes foram organizados em uma arquitetura estrutural que será descrita na Seção 5.4.2.

5.4.1.1. Web Service RESTful

O *Web Service RESTful* contém a modelagem do método ESPIM e é responsável por armazenar e prover dados para o sistema Web e para o aplicativo móvel. A Figura 5.4 apresenta a modelagem de eventos para o método ESPIM.

A modelagem proposta prevê a definição de diferentes tipos de eventos generalizáveis para diferentes cenários no domínio da saúde e educação. Os principais elementos da modelagem são:

- *Programa*: um programa é um conjunto de eventos de monitoramento que podem estar associados a um ou mais participantes monitorados. Eventos podem ser classificados como ativos, que são aqueles que necessitam de uma interação explícita com o dispositivo móvel, ou passivos, em que participantes têm informações coletadas em plano de fundo e não precisam interagir diretamente;
- *Pessoa*: pode ser um participante do programa, ou seja, o indivíduo monitorado, ou um observador/cuidador, responsável pelo participante e seu respectivo programa;
- *Evento*: um evento pode ser do tipo ativo ou passivo. Um evento passivo é aquele que coleta informações de sensores de *smartphones* em intervalos contínuos de tempo. Enquanto um evento ativo possui um conjunto de intervenções. Uma intervenção se caracteriza pela necessidade de interação explícita com o usuário, ou seja, ocorre uma interrupção que deve ser atendida pelo participante. Um evento ativo pode coletar dados de sensores durante a interação do participante com suas intervenções;
- *Gatilho de Evento*: um gatilho de evento pode ter diferentes tipos e condições: manual, temporal e contextual. Um gatilho manual dispara um evento apenas quando requisitado pelo participante. O gatilho temporal dispara eventos agendados em horários e dias específicos. Já o gatilho contextual considera informações de sensores e de contexto para disparar eventos;
- *Sensor*: o objeto sensor define quais informações de sensores serão coletadas e em que condições. Utilizando os sensores do dispositivo, podem ser coletadas informações sobre atividades (via acelerômetro e giroscópio), luz ambiente (via câmera) e áudio ambiente (via microfone), por exemplo;
- *Intervenção*: um evento ativo é composto por uma ou mais intervenções. Uma intervenção representa uma ocorrência em que é necessário interagir com o dispositivo, ou seja, é uma situação em que o usuário é interrompido pelo dispositivo para interagir com lembretes, questionamentos ou requisições de tarefa. Uma intervenção pode apresentar na tela diferentes estímulos: texto, imagem, áudio ou vídeo. O modelo atualmente prevê quatro tipos de intervenção que determinam a forma de interação com o participante: vazia, questão, mídia e tarefa. Uma intervenção vazia se caracteriza por não necessitar da entrada de dados por parte do usuário, sendo adequada para representar lembretes, mensagens ou estímulos. Uma intervenção em forma de questão pode ser de diferentes tipos, como aberta, múltipla escolha com uma única seleção, múltipla escolha com múltipla seleção, escala Likert e diferencial semântico. Uma intervenção de mídia requer que o participante capture uma imagem, um áudio ou um vídeo. Já uma tarefa trata da execução de tarefas externas ao modelo, permitindo que outros aplicativos sejam iniciados pela aplicação móvel, como um jogo. Diferentes tipos de intervenção podem ser combinadas em um único evento. Outro elemento importante em uma intervenção é uma condição. Se a intervenção for uma questão de múltipla escolha, o especialista pode definir um novo caminho dependendo da resposta selecionada pelo participantes. Isto implica na criação de fluxos não sequenciais de execução;
- *Resultado*: o objeto resultado é responsável pela persistência dos dados coletados e seus metadados associados. Um resultado é sempre associado a um participante e à sua intervenção correspondente.

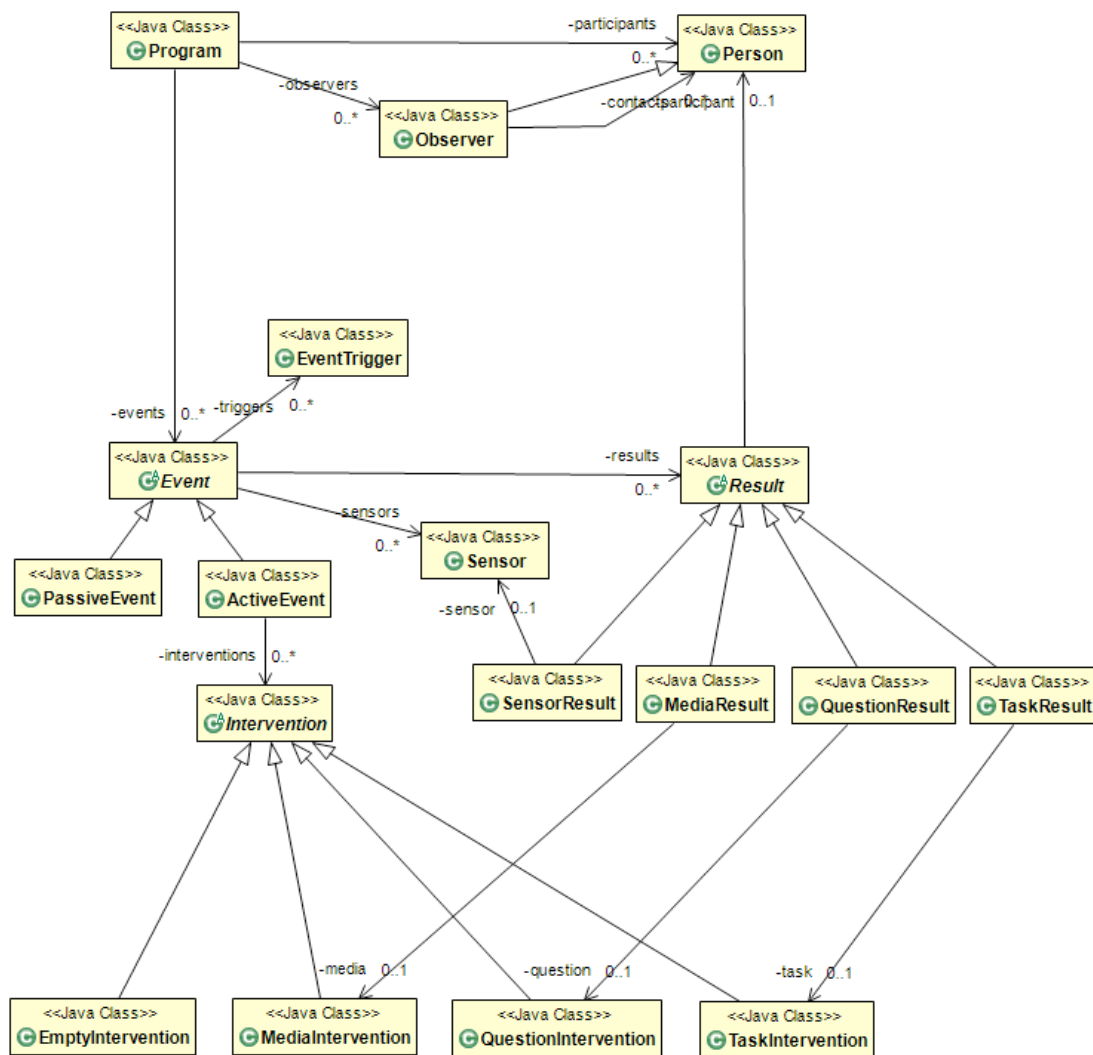


Figura 5.4. Modelagem de um programa no método ESPIM.

5.4.1.2. Sistema Web para Especialistas

A interface Web permite a criação de programas personalizados, cadastro de participantes e visualização dos resultados enviados ao servidor. A interface para criação de programas é baseado no modelo já discutido e apresentado na Figura 5.4.

Um usuário especialista cadastrado no sistema pode criar um programa, composto por eventos que compreendem um conjunto de intervenções. O programa pode ser associado a um ou mais participantes. Atualmente o interface Web possibilita a criação de intervenções com questões do tipo aberta, múltipla escolha com uma opção ou múltiplas escolha com mais de uma opção, mensagens e tarefas (em que um aplicativo externo é iniciado).

A interface foi construída utilizando o *framework* MCV Grails. O *framework* foi escolhido para facilitar a integração do *Web Service* com a biblioteca Java desenvolvida, pois o *framework* também suporta a linguagem Java. As páginas HTML foram construídas utilizando o biblioteca JavaScript jQuery e o *framework* CSS Bootstrap. A interface se comunica com o *Web Service* por meio de requisições HTTP que trocam

informações no formato JSON, de modo que o *Web Service* é o responsável pela persistência dos dados.

O *login* do especialista na interface é feito por meio de seu conta no Google. O cadastro de participantes é feito por meio de um formulário e também é associado a seu endereço de e-mail da conta Google. Esse formato foi utilizado para facilitar a integração e o *login* com sistemas Android. Dessa maneira, o usuário não precisa fornecer *login* e senha para utilizar a aplicação Android. Além disso, o mecanismo de autenticação é de responsabilidade da API do Google. O cadastro de programas, por sua vez, é feito por um formulário que oferece a funcionalidade de associar participantes àquele programa específico. Os eventos são construídos por meio de um formulário dinâmico semelhante a um fluxograma, que permite criar intervenções e conectá-las de maneira flexível, criando assim, fluxos não sequenciais.

A Figura 5.5 ilustra um exemplo de evento que pode ser programado pela interface Web. Nesse exemplo, semelhante àquele ilustrado no Cenário Exemplo com um idoso, uma questão de múltipla escolha (“Você está sozinho ou acompanhado?”) define o fluxo das intervenções. Se o participante estiver sozinho, ele deve jogar um aplicativo de jogo de memória e, no final, é parabenizado por sua atividade. Se estiver acompanhado, o participante deve responder quem o acompanha (questão aberta) e qual o seu humor naquele momento (questão de múltiplas opções) frente à interação realizada.

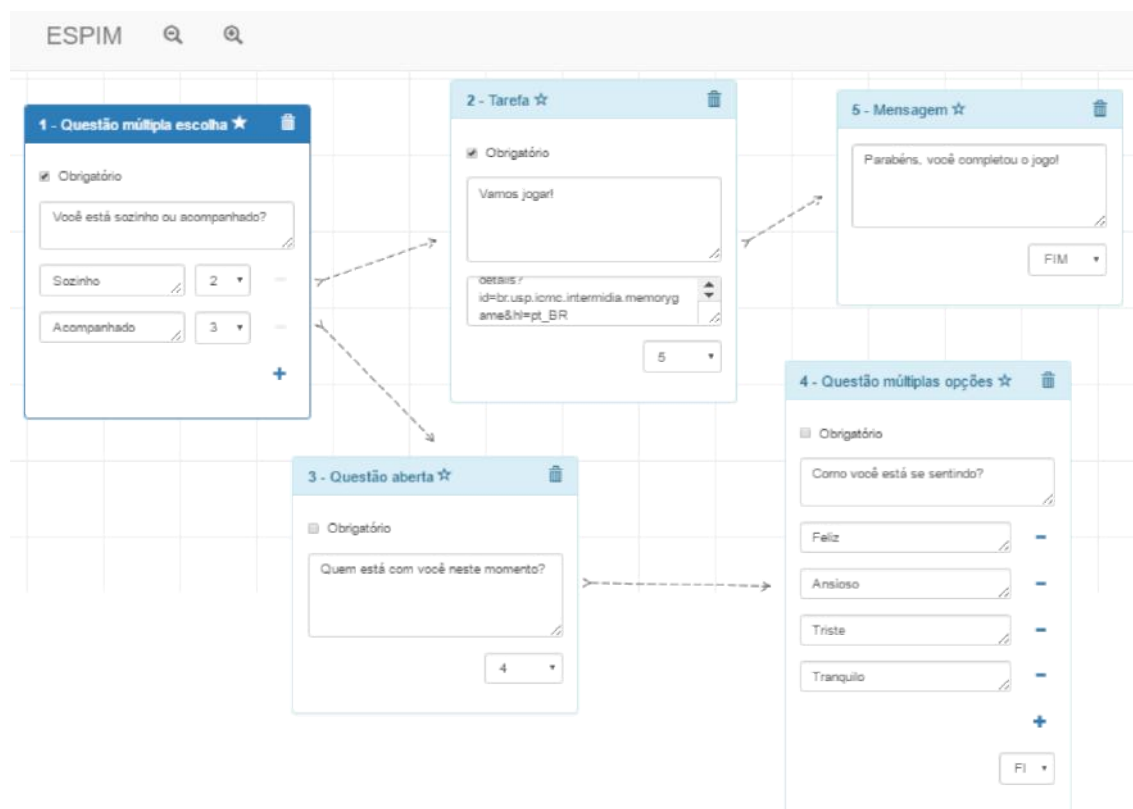


Figura 5.5. Exemplo da interface Web para criação de um conjunto de intervenções com condição.

Após finalizar o bloco de eventos, a interface verifica erros e inconsistências no grafo de intervenções e notifica o usuário ao identificar problemas. Se o evento estiver consistente, no próximo passo o especialista deve definir um gatilho temporal,

delimitando o horário e dias da semana que o evento será disparado no dispositivo do participante, conforme ilustrado na Figura 5.6. Essas informações são codificadas em uma expressão *CRON*⁵.



Figura 5.6. Definindo o gatilho temporal.

Ao finalizar o programa, os dados associados a ele são codificados em JSON e enviados ao *Web Service* para persistência e posterior recuperação por um sistema de acompanhamento (por exemplo, a aplicação móvel descrita na seção a seguir).

É também pela interface Web que o especialista pode acessar os dados coletados pelas intervenções. Nesse caso, a interface acessa o *Web Service* que retorna um documento de texto no formato *Comma-Separated Value* (CSV). Esse documento pode ser aberto em qualquer processador de texto e é compatível com softwares de planilha eletrônica, como o *Microsoft Excel*, contendo os dados associados àquele programa. Esse arquivo fica disponível para *download* pelo especialista.

5.4.1.3. Aplicativo Móvel para Acompanhamento

Os programas ESPIM são entidades totalmente personalizáveis. Ou seja, um profissional ou pesquisador pode definir um programa de acordo com sua área, metodologia e pessoas envolvidas. Isto implica na necessidade de desenvolvimento de aplicações que leiam as definições de um programa ESPIM e se comportem de acordo com os eventos, gatilhos e intervenções definidos.

Neste contexto, foi desenvolvido um aplicativo para o sistema Android que recupera os programas estipulados pelo especialista, para o usuário do dispositivo via REST. O aplicativo então gera as telas de intervenções dinamicamente por meio da leitura de arquivos JSON, os quais definem a estrutura de um programa. Além de gerar as telas, o aplicativo também cria dinamicamente notificações com base nos horários definidos nos gatilhos do arquivo JSON. Para a comunicação com o *Web Service*, o aplicativo desenvolvido utiliza a biblioteca em Java que simplifica as chamadas REST.

As Figuras 5.7 e 5.8 ilustram as telas correspondentes ao programa definido pela Figura 5.6. Tais figuras mostram o fluxo de acordo com a resposta para a primeira questão respondida pelo participante. Intervenções obrigatórias são acompanhadas de um asterisco (*). Se o participante tentar seguir para a próxima intervenção sem responder ou realizar a tarefa estipulada, o aplicativo retorna uma mensagem de alerta. Todas as telas apresentam o botão “Próximo”, para avançar a questão, e o botão

⁵ https://docs.oracle.com/cd/E12058_01/doc/doc.1014/e12030/cron_expressions.htm

“Voltar”, para destacar a possibilidade de voltar à questão anterior. Há um cuidado especial com o fluxo das telas pois, em casos de fluxos condicionais, a ação de voltar deve considerar as escolhas anteriores do participante. Na primeira tela não é apresentado o botão “Voltar”. Na última tela o botão “Terminar” é exibido e, ao clicar no botão, uma mensagem de confirmação é exibida. Para os casos em que uma tarefa deve ser realizada, o participante deve clicar no botão “Iniciar” e, na sequência, é disparado o aplicativo correspondente àquela tarefa.

Já para uma intervenção do tipo *mensagem*, o botão “Continuar” apresenta o mesmo comportamento dos botões “Próximo” e “Terminar”. Após responder às intervenções, o aplicativo envia as respostas das questões em formato JSON para o *Web Service*.

Para prover uma usabilidade adequada a um público mais amplo, a interface do aplicativo seguiu uma série de diretrizes para usuários da terceira idade e foi submetida a uma avaliação heurística com a colaboração de quatro alunos de pós-graduação.



Figura 5.7. Fluxo de intervenções 1 - Se o participante estiver sozinho: realizar a tarefa de jogar.



Figura 5.8. Fluxo de intervenções 2 - Se o participante estiver acompanhado: perguntar quem o acompanha e qual o seu humor.

5.4.2. Componentes para acesso a Sensores

Os aplicativos em Android que utilizam os sensores dos dispositivos móveis para acessarem funcionalidades diversas podem ser implementados utilizando um modelo arquitetural que disponibiliza as principais funcionalidades de cada sensor desses

dispositivos e permite que o desenvolvedor escolha quais sensores utilizar, de forma simplificada. Essa arquitetura foi elaborada para que futuras aplicações, que utilizem um número maior de sensores e outros hardwares (como câmera e microfone), possam ser implementadas de forma mais simples e em tempo reduzido. O modelo segue o padrão de projetos MVC (*Model View Controller*), que separa a definição dos dados a serem armazenados (*Model*) da camada de visualização (*View*) e da entrada de dados (*Controller*). A arquitetura completa está ilustrada na Figura 5.9.

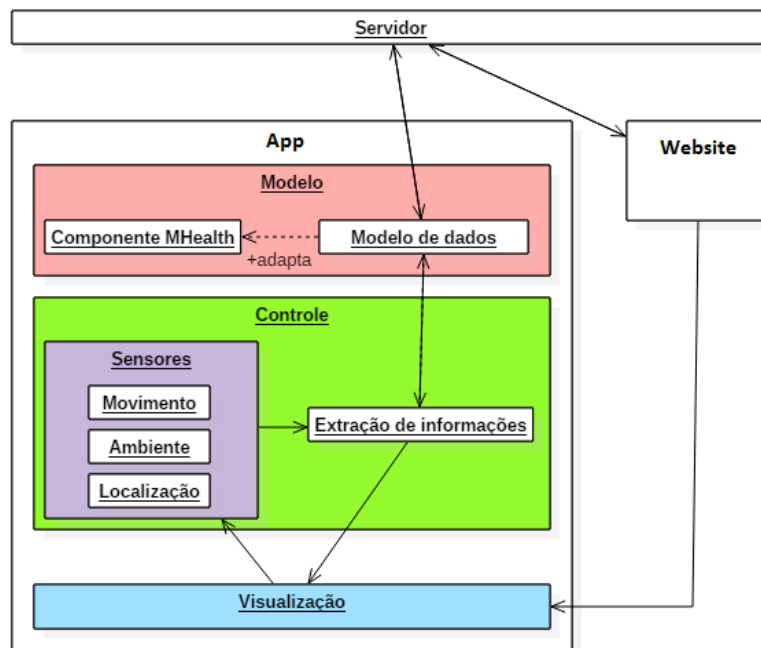


Figura 5.9. Arquitetura para desenvolvimento de aplicativos de dispositivos móveis.

A interface do aplicativo Android é representada pelo componente "Visualização". A partir da interface, a aplicação tem acesso à camada de controle. O desenvolvedor também pode contar com arquivos XML que representam fragmentos de interfaces para a câmera, para o microfone e para os sensores, os quais podem acessar as funcionalidades desses dispositivos por meio da camada de controle.

O componente "Sensores" contém classes de fachada com as funcionalidades mais importantes de cada sensor já implementadas, cabendo ao desenvolvedor escolher quais sensores serão utilizados em sua aplicação, bem como as funcionalidades que ele julgar necessárias.

Apesar dos dados dos sensores disponibilizados no módulo "Sensores" serem suficientes para a implementação de muitas aplicações, o desenvolvedor pode realizar operações sobre os dados obtidos. Esse pós-processamento de informações está ilustrado pelo componente "Extração de informações", o qual é implementado inteiramente pelo desenvolvedor, com base em suas necessidades.

Após o processamento das informações pela aplicação, os dados podem ser armazenados na camada "Dados". Nessa etapa, o desenvolvedor tem a opção de armazenar os dados em JSON no próprio dispositivo, para posterior envio ao servidor. Essa opção é necessária em caso de limitações momentâneas de conexão com a Internet.

A camada "Dados" está associada à esquemas pré-definidos no formato JSON implementados em Java, que definem a estrutura correta de parâmetros médicos específicos para aplicações para a saúde, por exemplo. Tais esquemas são recomendados pela plataforma *openMHealth*⁶, que disponibiliza um repositório contendo recomendações de como armazenar dados médicos referentes a diversos parâmetros, tais como altura, alimentação, batimentos cardíacos, dentre outros. Os esquemas são, portanto, uma compilação de atributos utilizados para descrever os parâmetros de interesse dos profissionais da área médica. Contudo, o uso dos esquemas não é obrigatório para o desenvolvimento de aplicações usando a arquitetura apresentada.

A seção a seguir detalha as funcionalidades do ESPIM Web em sua atual versão.

5.5. Utilizando o ESPIM para criar programas para coleta de dados a distância

As figuras abaixo ilustram telas da interface Web do ESPIM. Por meio delas é possível realizar o cadastro no sistema, criar novos participantes, editar informações de participantes existentes, programar uma coleta de dados, criar eventos de coleta, selecionar usuários para eventos específicos, definir disparos, dentre outras funcionalidades.

A Figura 5.10 exibe a tela em que o especialista realiza o seu cadastro no sistema informando nome, e-mail, senha e o seu papel. O papel define a função do especialista na coleta de dados que ele realizará por meio do ESPIM, por exemplo: terapeuta, professor, geriatra etc.

ESPIM - Interface do Especialista Bem-vindo(a), espim Ações Recentes

Início / Espim / Observadores / Adicionar Observador

Adicionar Observador

Fields in **bold** are required.

Nome:
Fulana de Tal

E-mail:
fulana@gmail.com

Senha:

Papel:
Geriatra

Salvar e adicionar outro(a) Salvar e continuar editando Salvar

Figura 5.10. Tela de cadastro do especialista no ESPIM.

⁶ Documentação disponível em <http://www.openmhealth.org/documentation/#/schema-docs/schema-library>.

Após realizar o seu cadastro no ESPIM, o especialista pode adicionar novos participantes, bem como novos observadores, eventos e programas, conforme ilustrado na Figura 5.11.

The screenshot shows the ESPIM - Interface do Especialista dashboard. At the top, there is a header with the title 'ESPIM - Interface do Especialista', a user greeting 'Bem-vindo(a), espim', and a dropdown menu 'Ações Recentes'. Below the header is a breadcrumb trail 'Início / Espim'. A secondary navigation bar contains 'Espim administration' and an 'Applications' dropdown. The main content area is titled 'Espim' and features a table with four rows: 'Eventos', 'Observadores', 'Pessoas', and 'Programas'. Each row has a '+ Adicionar' button and a 'Modificar' button with a pencil icon.

Category	+ Adicionar	Modificar
Eventos	+ Adicionar	Modificar
Observadores	+ Adicionar	Modificar
Pessoas	+ Adicionar	Modificar
Programas	+ Adicionar	Modificar

Figura 5.11. Tela com possibilidades de cadastros.

O especialista pode cadastrar novos participantes e editar informações de participantes já adicionados, conforme ilustrado na Figura 5.12.

The screenshot shows the 'Adicionar Pessoa' form. The breadcrumb trail is 'Início / Espim / Pessoas / Adicionar Pessoa'. The form title is 'Adicionar Pessoa'. A yellow warning box states 'Fields in bold are required.' The form has two input fields: 'Nome:' with the value 'Josias' and 'E-mail:' with the value 'josias@gmail.com'. At the bottom, there are three buttons: 'Salvar e adicionar outro(a)', 'Salvar e continuar editando', and a blue 'Salvar' button.

Figura 5.12. Tela de cadastro de novos participantes.

O especialista também pode cadastrar um programa interventivo. Ao realizar esse cadastro é preciso definir um título, uma breve descrição, associar participantes e observadores para aquela intervenção (Figura 5.13).

ESPIM - Interface do Especialista Bem-vindo(a), **espim** Ações Recentes

Início / Espim / Programas / Adicionar Programa

Adicionar Programa

Fields in **bold** are required.

Título:
Jogo de Memória para Alzheimer

Descrição:

Observadores:

+ Observadores disponíveis

Filtro

Fulana de Tal

Escolher todos

Observadores escolhido(s)

Remover todos

Mantenha pressionado o "Control", ou "Command" no Mac, para selecionar mais de uma opção.

Participantes:

+

Figura 5.13. Tela de cadastro de programas interventivos.

As Figuras 5.14a e 5.14b ilustram a tela em que o especialista seleciona os participantes que farão parte da coleta de dados.

Previendo a existência de diversos participantes, a interface do ESPIM permite que os especialistas sejam selecionados pelo nome usando também o serviço de auto completar.

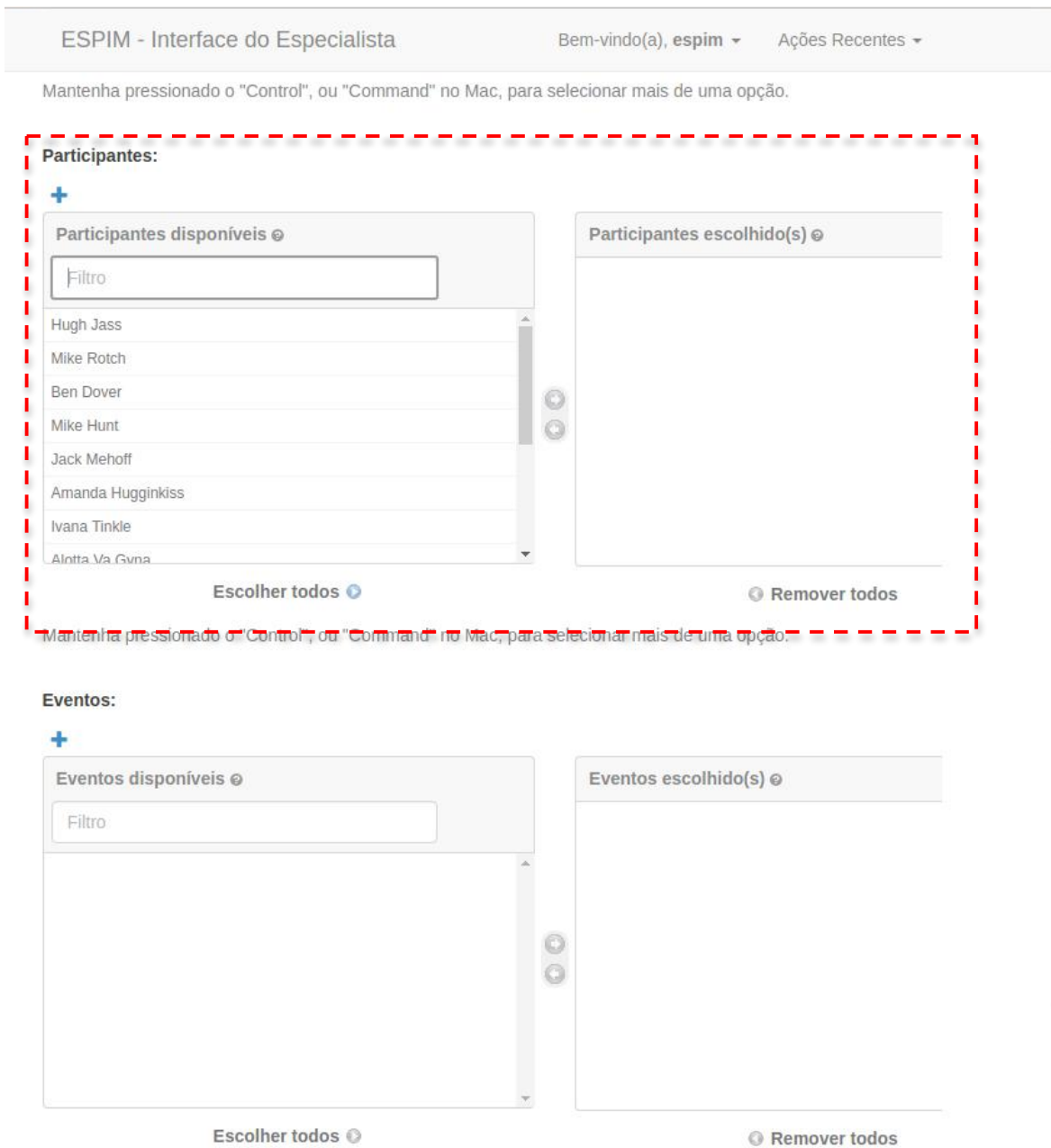


Figura 5.14 a. Tela de seleção de participantes para uma intervenção.

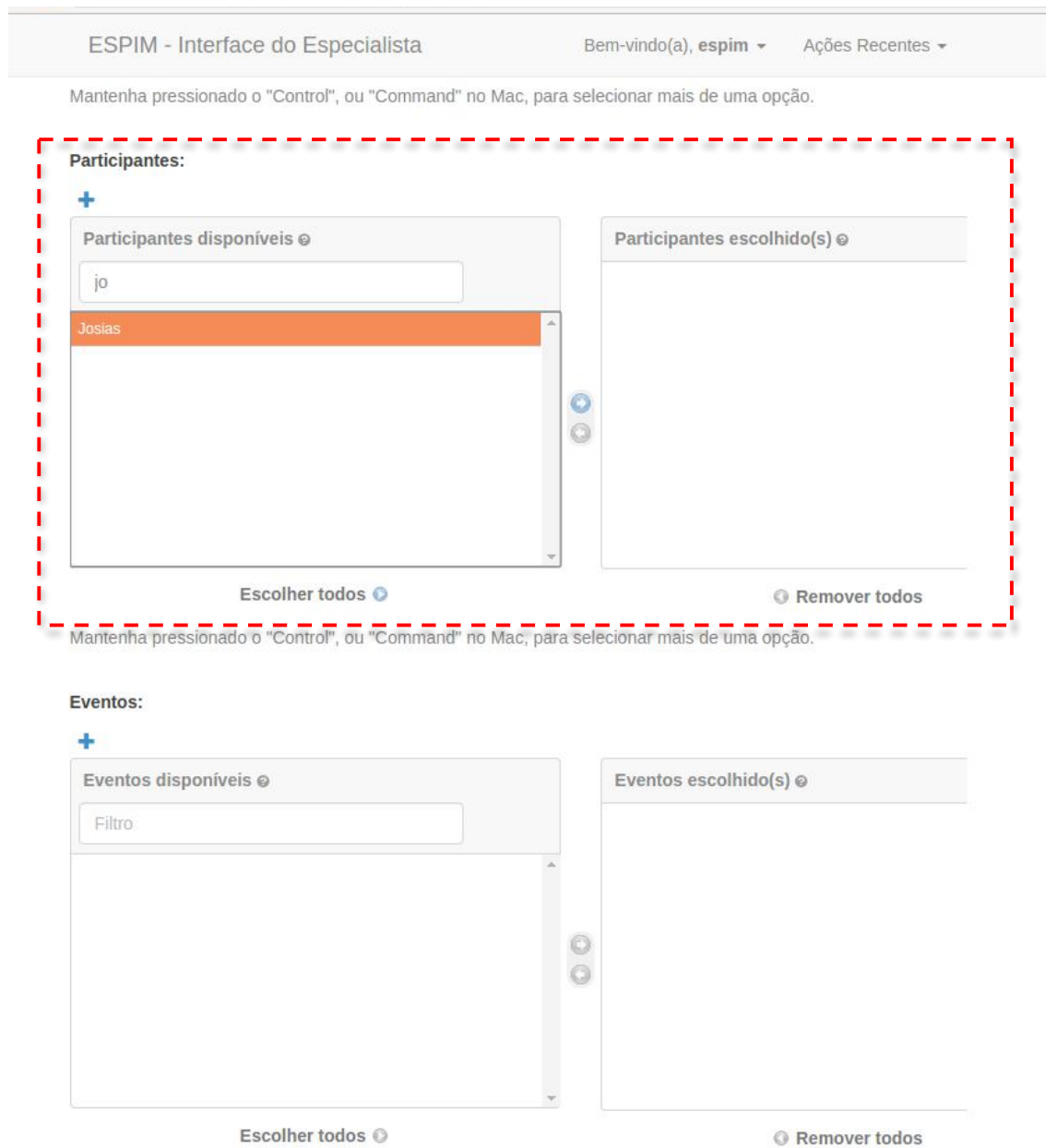


Figura 5.14 b. Tela de seleção de participantes para uma intervenção.

Ao planejar uma coleta de dados o especialista também pode programar eventos (tela ilustrada na Figura 5.15) e associá-los a uma coleta de dados.

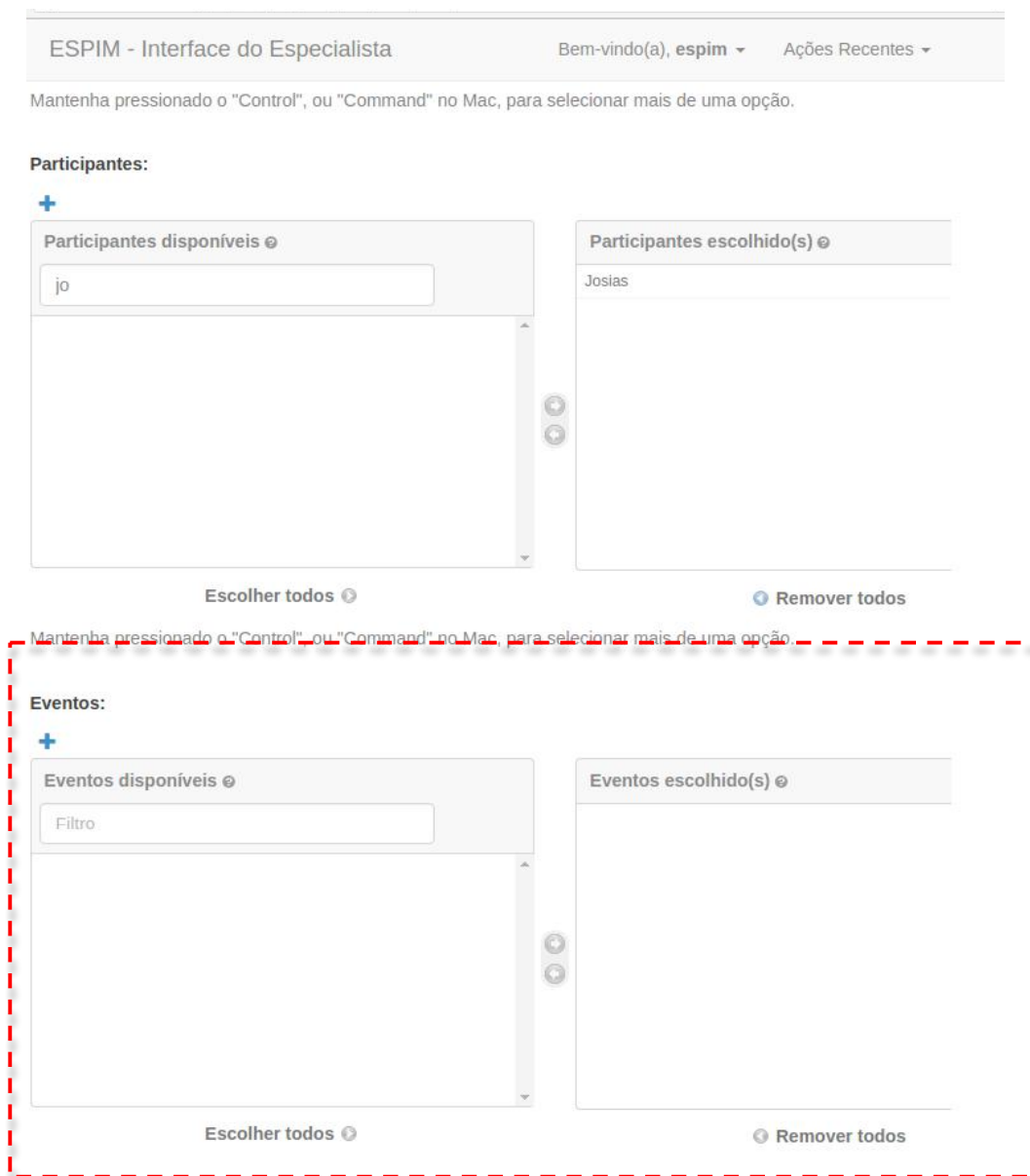


Figura 5.15. Telas de criação de eventos.

Entre as possibilidades de eventos o especialista pode programar questões de múltipla escolha com uma opção, questões com múltiplas opções de escolha, questões abertas, mensagens ou tarefas, conforme ilustrado no trecho em destaque na Figura 5.16.



Figura 5.16. Possibilidades de eventos disponíveis ao especialista.

A Figura 5.17 ilustra a criação de uma questão de múltipla escolha (*radio button*).

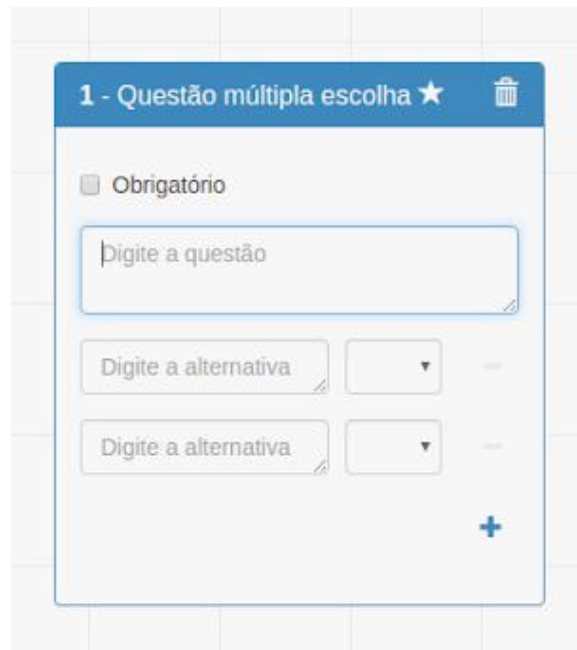


Figura 5.17. Criando eventos com questões de múltipla escolha.

As Figuras 5.18a e 5.18b ilustram um exemplo de evento do tipo múltipla escolha e a criação de um evento do tipo Tarefa associado ao programa interventivo em elaboração.

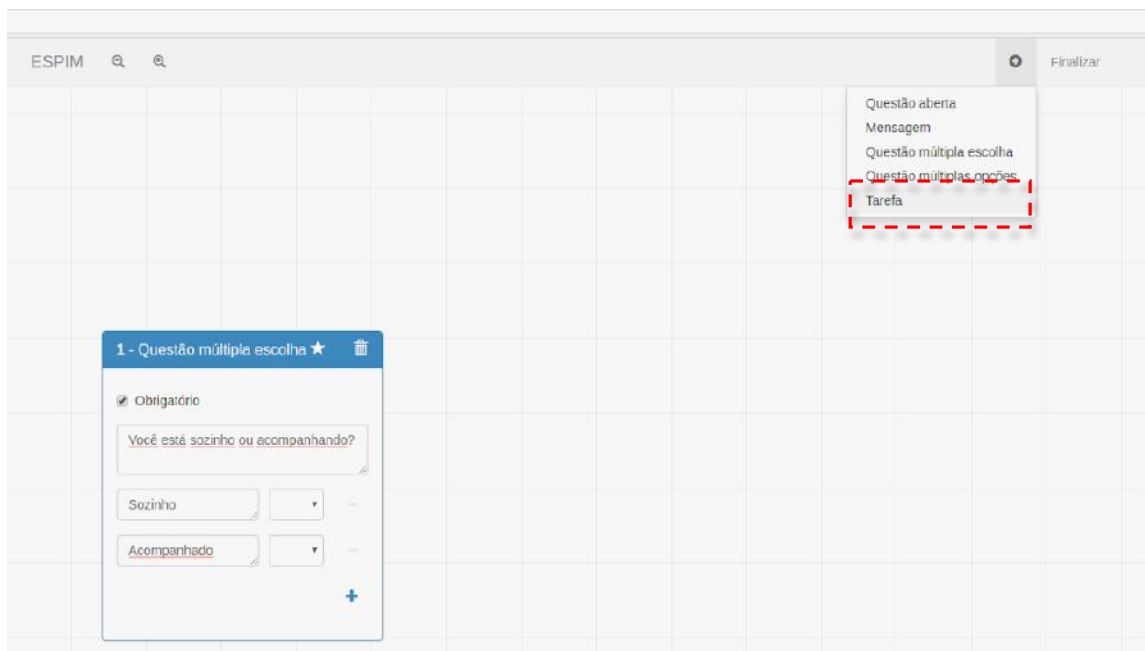


Figura 5.18a. Criando eventos do tipo Tarefa.



Figura 5.18b. Criando eventos do tipo Tarefa.

A caixa com destaque em azul mais escuro na Figura 5.18b representa o primeiro evento do programa interventivo e o ícone em estrela permite alteração desse primeiro evento ao ser clicado.

A Figura 5.19 ilustra a criação da tarefa iniciada na Figura 5.18 e a criação de um novo evento, também do tipo Tarefa, associado ao programa em elaboração. Caso a tarefa seja uma aplicação externa ao ESPIM, um campo específico é disponibilizado para carregar a URL onde a aplicação/aplicativo em questão está disponibilizada.

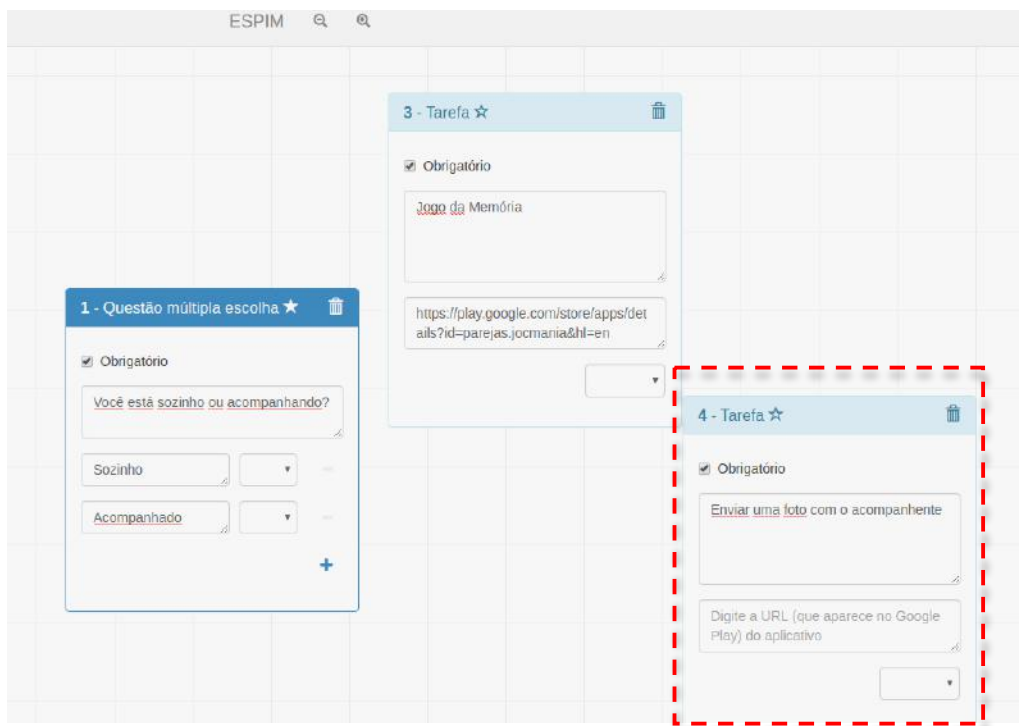


Figura 5.19. Instanciação de eventos do tipo Tarefa.

Após a criação dos eventos, ao escolher uma das opções numéricas na caixa de seleção o especialista faz uma ligação entre o evento e o seu sucessor, conforme o fluxo de tarefas desejado pelo mesmo para o participante. Veja destaque na Figura 5.20.

A seta tracejada indica o caminho apontado. Também por meio da caixa de seleção o especialista pode informar que o evento que ele está manipulando é o evento final.

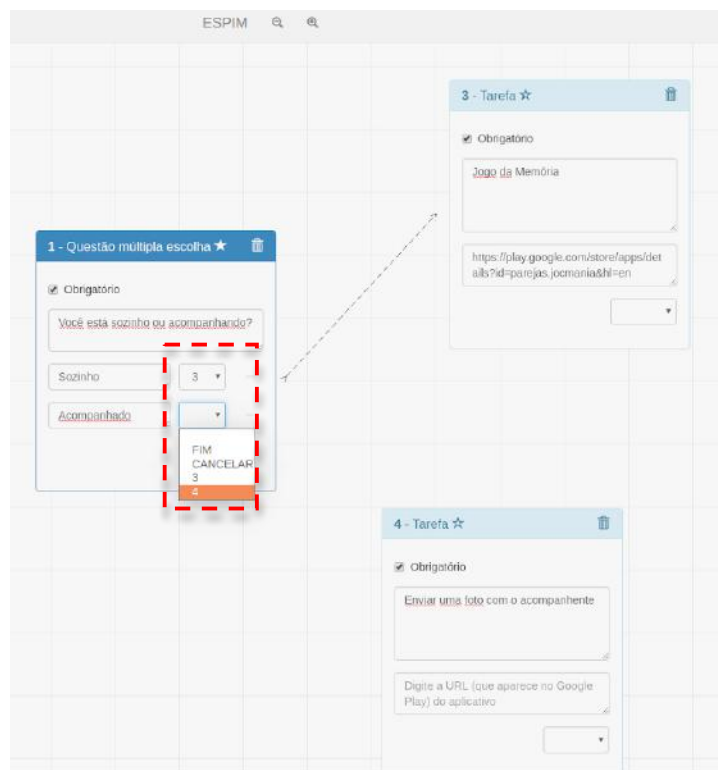


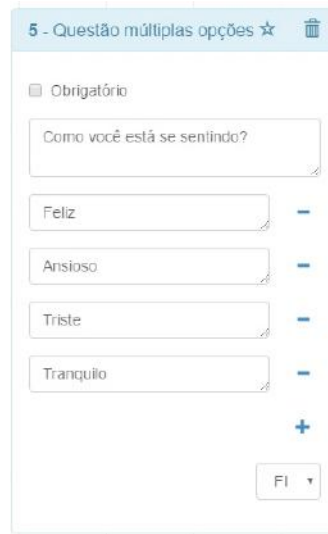
Figura 5.20. Fazendo a ligação entre eventos.

O especialista também pode prever e programar uma questão de múltipla escolha tipo Likert, conforme ilustrado na Figura 5.21.



Figura 5.21. Criando eventos do tipo Questão de múltipla escolha.

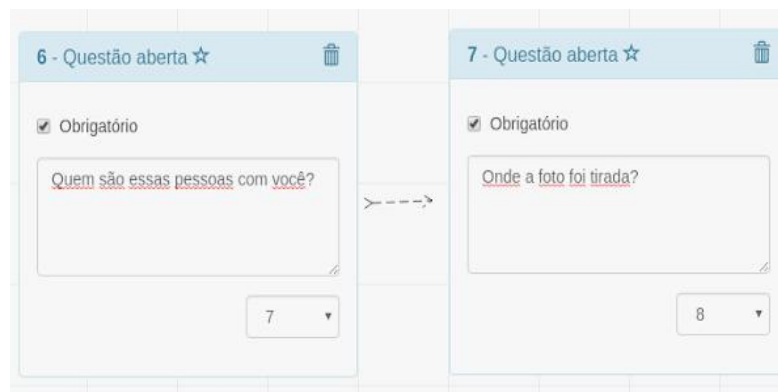
Eventos de múltiplas opções (*checkbox*) também podem ser criados, conforme ilustrado na Figura 5.22.



The screenshot shows a question editor for a multiple-choice question. At the top, it says "5 - Questão múltiplas opções" with a star icon and a trash icon. Below that is a checkbox labeled "Obrigatório" which is checked. The question text is "Como você está se sentindo?". There are four radio button options: "Feliz", "Ansioso", "Triste", and "Tranquilo". Each option has a minus sign to its right. At the bottom right, there is a plus sign and a dropdown menu labeled "FI".

Figura 5.22. Criando eventos com Questões de múltiplas opções.

A Figura 5.23 ilustra a criação de eventos do tipo Questões abertas, em que o participante deve digitar um texto como resposta e, em versões futuras, gravar um áudio.



The screenshot shows two question editors side-by-side. The left one is titled "6 - Questão aberta" and has a checked "Obrigatório" checkbox. The question text is "Quem são essas pessoas com você?". Below the text area is a dropdown menu showing the number "7". The right one is titled "7 - Questão aberta" and also has a checked "Obrigatório" checkbox. The question text is "Onde a foto foi tirada?". Below the text area is a dropdown menu showing the number "8". A dashed arrow points from the left editor to the right one.

Figura 5.23. Criando eventos com Questões abertas.

Ao finalizar o programa interativo e o fluxo de eventos a ser realizado pelo participante, o especialista pode criar um evento do tipo Mensagem com a marcação de fim desse fluxo, acompanhada de uma mensagem a esse participante, conforme ilustrado na Figura 5.24.

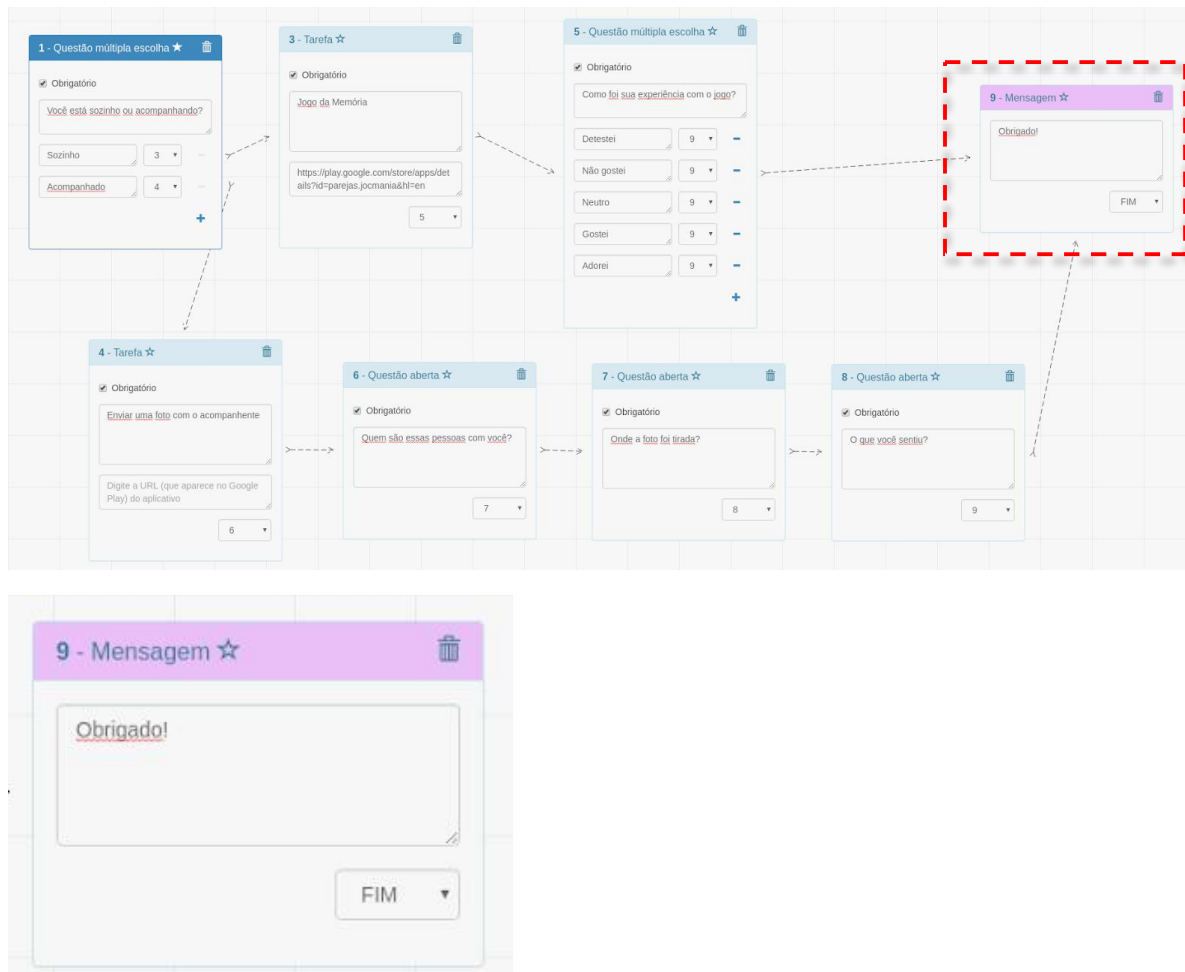


Figura 5.24. Finalizando o fluxo de eventos para um determinado programa com uma mensagem.

A versão atual do sistema ESPIM foi concebida com o apoio de profissionais de diferentes áreas de atuação, entre elas, computação e saúde. Além das funcionalidades implementadas de protótipos anteriores, como o *SmartESM*, outros requisitos foram coletados junto aos profissionais de área e, após ponderações, deverão fazer parte das próximas versões do sistema.

Outros trabalhos existentes na literatura também serviram de inspiração para a concepção de novas funcionalidades do ESPIM.

A Tabela 5.2 exibe uma comparação entre o ESPIM e funcionalidades já implementadas por outros sistemas semelhantes.

Tabela 5.2. Comparação entre o sistema ESPIM e outros sistemas para coleta de dados.

	ESPIM	MyExperience (FROELICH, et al., 2007)	PsychLog (GAGGIOLI et al., 2013)	AndWellness (HICKS et al., 2010)	StudentLife (WANG et al., 2014)	Ohmage (TANGMU- NARUNKIT et al., 2015)	PACO (BAXTER et al., 2015)
Autorrelato via texto (aberto/fechado)	Sim	Sim	Sim	Sim	Sim	Sim	Sim
Autorrelato via multimídia	Previsto: foto, vídeo, áudio.	Não	Não	Foto	Não	Foto	Foto
Coleta de dados via sensores	Previsto	Sim	Sim	Sim	Sim	Sim	Sim
Integração com outros dispositivos	Previsto	Não	Sim	Não	Não	Sim	Não
Recursos de autoria para pesquisador/especialista	Web	XML	TXT	GUI	Não	GUI	Web
Recursos de visualização de informações para profissional/paciente	Previsto	Não	Básicos	Avançados	Não	Avançados	Básicos
Recursos de sistemas de recomendação	Não	Não	Sim	Sim	Não	Não	Não
Recursos de bate-papo ao vivo	Previsto	Não	Não	Não	Não	Não	Não

As versões do ESPIM serão submetidas à avaliações com usuários alvo para que seja observado aspectos relacionados à experiência de uso e à usabilidade da sua interface. Para esta etapa especialistas da computação, da saúde e da educação especial, bem como pacientes e cuidadores serão convidados a interagirem com o sistema e programas interventivos criados, e fornecerem *feedback* sobre o sistema.

5.6. Agradecimentos

Agradecemos o apoio financeiro das agências de fomento à pesquisa - FAPESP, CNPq, CAPES - que têm possibilitado o desenvolvimento deste projeto.

Referências Bibliográficas

- Araújo, R. B. (2003). Computação ubíqua: Princípios, tecnologias e desafios. In: XXI Simpósio Brasileiro de Redes de Computadores. 2003. p. 11-13.
- Barrett, L. F., & Barrett, D. J. (2001). An introduction to computerized experience sampling in psychology. *Social Science Computer Review*, 19(2): 175–185.
- Baxter, K. K., Avrekh, A., & Evans, B. (2015). Using experience sampling methodology to collect deep data about your users. In *Proceedings of the 33rd Annual ACM Conference Extended Abstracts on Human Factors in Computing Systems, CHI EA'15*, pages 2489–2490, New York, NY, USA. ACM.
- Csikszentmihaly, M. (2000). *Beyond boredom and anxiety* (2nd ed.). San Francisco: Jossey-Bass.

- Csikszentmihaly, M., Larson, R., & Prescott, S. (1977). The ecology of adolescent activity and experience. *Journal of Youth and Adolescence*, 6(3): 281–294.
- Csikszentmihaly, M., & Larson, R. (1987). Validity and reliability of the experience-sampling method. *The Journal of Nervous and Mental Disease*, 175(9), 526–536.
- De Rose, J., Souza, D., Rossito, A., & De Rose, T. (1989). Equivalência de estímulos e generalização na aquisição de leitura após história de fracasso escolar. *Psicologia: Teoria e Pesquisa*, 5(2): 325–346.
- De Souza, D. G., De Rose, J. C., Faleiros, T. C., Bortoloti, R., Hanna, E. S., & Mcilvane, W. J. (2009). Teaching generative reading via recombination of minimal textual units: A legacy of verbal behavior to children in Brazil. *International Journal of Psychology and Psychological Therapy*, 9(1): 19.
- Floridou, G. A. & Mullensiefen, D. (2015). Environmental and mental conditions predicting the experience of involuntary musical imagery: An experience sampling method study. *Consciousness and Cognition*, 33(0): 472 – 486.
- Froehlich, J., Chen, M. Y., Consolvo, S., Harrison, B., & Landay, J. A. (2007). Myexperience: A system for in situ tracing and capturing of user feedback on mobile phones. In *Proceedings of the 5th International Conference on Mobile Systems, Applications and Services, MobiSys'07*, pages 57–70, New York, NY, USA.
- Fuller-Tyszkiewicz, M., McCabe, M., Skouteris, H., Richardson, B., Nihill, K., Watson, B., & Solomon, D. (2015). Does body satisfaction influence self-esteem in adolescents' daily lives? An experience sampling study. *Journal of adolescence*, 45: 11–19.
- Gaggioli, A., Pioggia, G., Tartarisco, G., Baldus, G., Corda, D., Cipresso, P., & Riva, G. (2013). A mobile data collection platform for mental health research. *Personal and Ubiquitous Computing*, 17(2): 241–251.
- Hartley, S., Haddock, G., Vasconcelos, Sa, D., Emsley, R., & Barrowclough, C. (2014). An experience sampling study of worry and rumination in psychosis. *Psychological medicine*, 44(08): 1605–1614.
- Hektner, J. M., Schmidt, J. A., & Csikszentmihaly, M. (2007). *Experience sampling method: Measuring the quality of everyday life*. Sage.
- Hicks, J., Ramanathan, N., Kim, D., Monibi, M., Selsky, J., Hansen, M., & Estrin, D. (2010). Andwellness: an open mobile system for activity and experience sampling. In *Wireless Health 2010*, pages 34–43. ACM.
- Hofmann, W., Adriaanse, M., Vohs, K. D., & Baumeister, R. F. (2014). Dieting and the self-control of eating in everyday environments: An experience sampling study. *British journal of health psychology*, 19(3): 523–539.
- Kackar, H. Z., Shumow, L., Schmidt, J. A., & Grzetich, J. (2011). Age and gender differences in adolescents' homework experiences. *Journal of Applied Developmental Psychology*, 32(2): 70 – 77.
- Kapoor, A. & Horvitz, E. (2008). Experience sampling for building predictive user models: A comparative study. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI'08*, pp. 657–666, New York, NY, USA. ACM.

- Keller, F. S. (1968). Good-bye, teacher... *Journal of Applied Behavior Analysis*, 1(1): 79.
- Kramer, I., Simons, C. J., Hartmann, J. A., Menne-Lothmann, C., Viechtbauer, W., Peeters, F., Schruers, K., Bommel, A. L., Myin-Germeys, I., Delespaul, P., *et al.* (2014). A therapeutic application of the experience sampling method in the treatment of depression: a randomized controlled trial. *World Psychiatry*, 13(1): 68–77.
- Lowdermilk, T. (2013). *Design Centrado no Usuário*. Editora: Novatec Editora, 184p.
- Lyytinen, K; Yoo. & Y. (2002). Issues and Challenges in Ubiquitous Computing. *Communications Of The ACM*, p.63-65, 2002.
- Lopez, V., Ahumada, L., Galdames, S., & Madrid, R. (2012). School principals at their lonely work: Recording workday practices through {ESM} logs. *Computers & Education*, 58(1): 413 – 422.
- Mcshane, J. M. & Zirkel, S. (2008). Dissociation in the binge-purge cycle of bulimia nervosa. *Journal of Trauma & Dissociation*, 9(4): 463–479.
- Maes, I. H., Delespaul, P. A., Peters, M. L., White, M. P., Van Horn, Y., Schruers, K., Anteunis, L., & Joore, M. (2015). Measuring health-related quality of life by experiences: The experience sampling method. *Value in Health*, 18(1): 44–51.
- Mujagic, Z., Leue, C., Vork, L., Lousberg, R., Jonkers, D., Keszthelyi, D., Hesselink, M., Schagen, T., Os, J., Masclee, A., *et al.* (2015). The experience sampling method—a new digital tool for momentary symptom assessment in IBS: an exploratory study. *Neurogastroenterology & Motility*, 27(9): 1295–1302.
- Myllykangas, S. A., Gosselink, C. A., Foose, A. K., & Gaede, D. B. (2002). Meaningful activity in older adults: Being in flow. *World Leisure Journal*, 44(3): 24–34.
- Nielsen, K. & Cleal, B. (2011). Under which conditions do middle managers exhibit transformational leadership behaviors? — An experience sampling method study on the predictors of transformational leadership behaviors. *The Leadership Quarterly*, 22(2): 344 – 352.
- Pimentel, M. G., Rocha, A. C., Cunha, B. C. R., Orlando, A. F., Machado Neto, O., Viel, C., Antunes, E., & Zaine, I. (2016). Apoio ao envelhecimento no lugar por meio de amostragem de experiências e de intervenção programada. *Revista da Faculdade de Medicina de Ribeirão Preto e do Hospital das Clínicas da Universidade de São Paulo*, 49 (Supl 2), 11.
- Riddle, M. M. D. & Arnold, D. M. V. (2007). *The day experience method: A resource kit*.
- Rieh, S. Y., Kim, Y. M., Yang, J. Y., & St. Jean, B. (2010). A diary study of credibility assessment in everyday life information activities on the web: Preliminary findings. *Proceedings of the American Society for Information Science and Technology*, 47(1): 1–10.
- Satyanarayanan, M. (2001). *Pervasive Computing: Vision and Challenges*. *Personal Communication, IEEE*, v. 8, n.4, p.10-17, 2001.
- Schuler, D., & Namioka, A. (1993). *Participatory Design: Principles and Practices*. Hillsdale, NJ: Lawrence Erlbaum Associates.

- Skinner, B. F. (1974). *About behaviorism*. New York: Appleton-Century-Crofts.
- Skinner, B. F. (1953). *Science and human behavior*. New York: Simon and Schuster.
- Skinner, B. F. (1958). Reinforcement today. *American Psychologist*, 13(3): 94.
- Skinner, B. F. (1961). Teaching machines. *Scientific American*, 205(11): 90–102.
- Skinner, B. F. (1972). Humanism and behaviorism. *The Humanist*, 32: 18–20.
- Swendeman, D., Comulada, W., Ramanathan, N., Lazar, M., & Estrin, D. (2015). Reliability and validity of daily self-monitoring by smartphone application for health-related quality-of-life, antiretroviral adherence, substance use, and sexual behaviors among people living with HIV. *AIDS and Behavior*, 19(2): 330–340.
- Tangmunarunkit, H., Hsieh, C., Jenkins, J., Ketcham, C., Selsky, J., & Alquaddoomi, F. (2015). Ohmage: A general and extensible end-to-end participatory sensing platform. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 6(3), 38.
- Udachina, A., Varese, F., Myin-Germeys, I., & Bentall, R. P. (2014). The role of experiential avoidance in paranoid delusions: An experience sampling study. *British Journal of Clinical Psychology*, 53(4): 422–432.
- Wang, R., Chen, F., Chen, Z., Li, T., Harari, G., Tignor, S., Zhou, X., Ben-Zeev, D., & Campbell, A. T. (2014). Studentlife: Assessing mental health, academic performance and behavioral trends of college students using smartphones. In *Proceedings of the 2014. ACM International Joint Conference on Pervasive and Ubiquitous Computing, UbiComp'14*, pages 3–14, New York, NY, USA.
- Wang, Z. & Tchernev, J. M. (2012). The “myth” of media multitasking: Reciprocal dynamics of media multitasking, personal needs, and gratifications. *Journal of Communication*, 62(3): 493–513.
- Wang, Z., Tchernev, J. M., & Solloway, T. (2012). A dynamic longitudinal examination of social media use, needs, and gratifications among college students. *Computers in Human Behavior*, 28(5): 1829 – 1839.
- Weiser, M. (1991). The computer for the 21st century. *Scientific American*, 265(3), 94–104.
- Yin, J., Yang, Q., & Pan, J. J. (2008). Sensor-based abnormal human-activity detection. *Knowledge and Data Engineering, IEEE Transactions on*, 20(8): 1082–1090.

Biografia Resumida dos Autores

Isabela Zaine - Psicóloga e Doutora em Psicologia pela UFSCar, Especialista em Psicoterapia Analítico-Comportamental, Docente de Ensino Superior do curso de Psicologia. Atua em análise do comportamento aplicada e experimental, programação de ensino, processos básicos de aprendizagem e tecnologia de ensino. Atualmente atua como pós-doutoranda na Universidade de São Paulo - Instituto de Ciências Matemáticas e de Computação, realizando pesquisas relacionadas a tecnologias interativas aplicadas a contextos educacionais e de saúde.



Kamila R. H. Rodrigues - Professora de Ensino Superior, Doutora em Ciência da Computação. Tem interesse nas áreas de Sistemas Multimídia e Interação Humano-Computador. Atua em pesquisas com o foco em acessibilidade e respostas emocionais de usuários, bem como em suas experiências durante a interação com sistemas multimídia interativos. Atualmente é pós-doutoranda na Universidade de São Paulo - Instituto de Ciências Matemáticas e de Computação, realizando pesquisas com documentos interativos para coleta especializada de experiências cotidianas de indivíduos.



Bruna C. R. da Cunha - Doutoranda em Ciências da Computação e Matemática Computacional pela Universidade de São Paulo. É Bacharel em Ciências de Computação pela Universidade de São Paulo (2009) e Mestre em Ciências da Computação e Matemática Computacional pela Universidade de São Paulo (2014). Realiza pesquisas nas áreas de Interação Usuário-Computador, User Experience, Tecnologia Assistiva e Computação Ubíqua.



Yuri Nehase Zuliani Goulart Magagnatto - Mestrando em Ciências da Computação e Matemática Computacional ICMC/USP. Tecnólogo em Sistemas para Internet na Faculdade de Tecnologia de Jaú (FATEC JAHU). Desenvolve pesquisas sobre utilização de aplicativos móveis na área da saúde, mais precisamente para a área de terapia ocupacional. Trabalhou como desenvolvedor web em empresas de e-commerce. Possui formação técnica em informática.



Alex F. Orlando - Graduado em Ciência da Computação e Mestre em Engenharia de Software pelo PPG-CC, ambos pela UFSCar. Atualmente é doutorando no PPG-CCMC do ICMC-USP, gerente executivo do Projeto SACI@Ipê, gerente do projeto GEIC e empresário na bluedotsoft. Tem experiência na área de Ciência da Computação, com ênfase em Software Básico, atuando principalmente nos seguintes temas: tecnologia, tendências, mercado, informação e multimídia.



Caio C. Viel - Aluno de Doutorado no programa de Computação e Matemática Computação na Universidade de São Paulo - Instituto de Ciências Matemáticas e de Computação. Possui graduação em Engenharia da Computação e mestrado em Computação pela Universidade Federal de São Carlos. Realiza pesquisas nas áreas de computação ubíqua e em sistemas multimídia interativos com foco em aplicações de Captura e Acesso.



André Luiz Carlomagno Rocha - Mestre em Ciência da Computação pela Universidade Federal da Bahia - UFBA (2014), atuando na área de Sistemas Web. Bacharel em Ciência da Computação pela Universidade Salvador - UNIFACS (2010), com ênfase em Sistemas de Computação e Redes de Computadores. Está envolvido também com estudos na área de coleta de dados usando métodos de pesquisa para capturar, por meio de estratégias usando dispositivos móveis e vestíveis, a experiência do usuário em tempo-real, bem como, o contexto em que ele se encontra.



Maria da Graça C. Pimentel - Professora Titular da Universidade de São Paulo, Doutora em Ciências da Computação pela University of Kent at Canterbury (1994), e Livre-docente pela Universidade de São Paulo (2001). Atua nas áreas de Engenharia de Documentos, Computação Ubíqua, Web Semântica, Interação Usuário-computador e CSCW, com pesquisas aplicadas à Tecnologias Assistivas, TV Digital e à Educação. Coordena projetos de pesquisa financiados por CNPq, CAPES e FAPESP, e coordenou projetos financiados pela FAPESP, CNPq, CAPES, FINEP e MCT/CITIC.

