

FACULDADE DE TECNOLOGIA DE SÃO JOSÉ DOS CAMPOS
FATEC PROFESSOR JESSEN VIDAL

DANIEL HIDEKI SHIBATA CASSI

DESENVOLVIMENTO DE AMBIENTES
RESPONSIVOS UTILIZANDO SENSORES

São José dos Campos

2015

DANIEL HIDEKI SHIBATA CASSI

**DESENVOLVIMENTO DE AMBIENTES
RESPONSIVOS UTILIZANDO SENSORES**

Trabalho de Graduação apresentado à
Faculdade de Tecnologia São José dos
Campos, como parte dos requisitos
necessários para a obtenção do título de
Tecnólogo em Banco de Dados.

Orientador: Me. Giuliano Araujo Bertoti

São José dos Campos
2015

Dados Internacionais de Catalogação-na-Publicação (CIP)
Divisão de Informação e Documentação

CASSI, Daniel Hideki Shibata
Desenvolvimento de Ambientes Responsivos utilizando Sensores.
São José dos Campos, 2015.
53f.

Trabalho de Graduação – Curso de Tecnologia Banco de Dados, FATEC de São José dos Campos: Professor Jessen Vidal, 2015.

Orientador: Mestre Giuliano Araujo Bertoti.

1. Áreas de conhecimento. I. Faculdade de Tecnologia. FATEC de São José dos Campos: Professor Jessen Vidal. Divisão de Informação e Documentação. II. Título

REFERÊNCIA BIBLIOGRÁFICA –

CASSI, Daniel Hideki Shibata. **Desenvolvimento de Ambientes Responsivos utilizando Sensores**. 2015. 53f. Trabalho de Graduação - FATEC de São José dos Campos: Professor Jessen Vidal.

CESSÃO DE DIREITOS –

NOME DO AUTOR: Daniel Hideki Shibata Cassi

TÍTULO DO TRABALHO: Desenvolvimento de Ambientes Responsivos utilizando Sensores

TIPO DO TRABALHO/ANO: Trabalho de Graduação / 2015.

É concedida à FATEC de São José dos Campos: Professor Jessen Vidal permissão para reproduzir cópias deste Trabalho e para emprestar ou vender cópias somente para propósitos acadêmicos e científicos. O autor reserva outros direitos de publicação e nenhuma parte deste Trabalho pode ser reproduzida sem a autorização do autor.

Daniel Hideki Shibata Cassi
Rua Benedito Osvaldo Lecques, Jd. Residencial Aquarius
CEP 12-246-021 – São José dos Campos – São Paulo

DANIEL HIDEKI SHIBATA CASSI

**DESENVOLVIMENTO DE AMBIENTES
RESPONSIVOS UTILIZANDO SENSORES**

Trabalho de Graduação apresentado à
Faculdade de Tecnologia São José dos
Campos, como parte dos requisitos
necessários para a obtenção do título de
Tecnólogo em Banco de Dados.

Composição da Banca

Jean Carlos Lourenço Costa, Especialista, FATEC

Denise Aves de Barros, Tecnólogo, ITA

Giuliano Araujo Bertoti, Me, FATEC

____/____/____
DATA DA APROVAÇÃO

A minha maravilhosa esposa Akemi, aos meus queridos filhos Kevin, Evellyn e Luan e ao meu primo Gabriel Cassi.

AGRADECIMENTOS

Agradeço ao Me. Giuliano Araujo Bertoti pela oportunidade de realizar esse trabalho, pela orientação, apoio e paciência, que dedicou seu tempo para me orientar, sendo imensurável a sua dedicação para a realização do mesmo.

Aos meus amigos de trabalho e de classe, pelas palavras de incentivo e apoio, em especial aos meus amigos Tarcísio Aparecido, Thiago Mendes, Lucas Norberto, Rodrigo Leite e Raphael Egídio que dedicaram o seu tempo para incentivar e compartilhar experiências.

Aos professores da Fatec, que compartilharam do seu conhecimento e amizade.

Aos meus familiares que sempre me apoiaram, com palavras e gestos de encorajamento, para que nunca desistisse de sonhos e desafios.

Quero agradecer também aos incentivos e ensinamentos do Budismo e ao presidente Ikeda, que com poucas palavras nos faz sentirmos fortes e perseverantes, e que em uma de suas frases explica: “O futuro existe, sim, na determinação interior de cada um. Ele não será um mar de rosas se não houver esforços; será somente ilusão.”.

“Quando perguntado sobre qual era a sua maior peça, a resposta de Charles Chaplin era sempre a mesma: ‘A próxima!’ Não há impasse quando se está imbuído de desafio. Não se anda porque existe um caminho; por andar é que se abre o caminho.”

Daisaku Ikeda

RESUMO

As informações recebidas no dia a dia são constantes e sem contexto. Nem sempre é possível recebê-las no momento certo como, por exemplo, estar dentro de uma sala de aula e receber o conteúdo da matéria. Ao contrário disso, recebe-se uma mensagem pelo Whatsapp ou Facebook de amigos ou familiares. Portanto, o objetivo desse projeto foi demonstrar que utilizando sensores, é possível que as informações digitais fossem sincronizadas com seus devidos contextos. Para o desenvolvimento foi utilizado um estudo de casos que mostra que, através dos sensores do *smartphone*, o usuário pode sincronizar os mundos virtuais e reais. Através deste projeto certificou-se que com os sensores já disponíveis nos mais atuais dispositivos móveis com o sistema Android é possível desenvolver uma solução simples e de baixo custo utilizando os sensores já existentes e recursos que temos no dia a dia.

Palavras-chaves: Android; informações digitais; *smartphone*; mundos virtuais e reais; sensores.

ABSTRACT

The information that we receive day-by-day are constant and out of context. It is not always possible to receive these information in their proper contexts, for example, be within a classroom and receive the content of the lecture. Conflicting this we receive a message through Whatsapp and Facebook from friends or family. Therefore, the aim of this project was to demonstrate using sensors it is possible that digital information to be synchronized with their proper contexts. For development this solution, a case study was used to show that through the smartphone sensors, the user can synchronize the virtual and real world. Through this project it was proofed that, using sensors already available in most current mobile devices with the Android system, it is possible to develop a simple and low cost using existing sensors and resources we have in day-by-day.

Keywords: Android; digital information; smartphone; virtual and real worlds; sensors.

LISTA DE FIGURAS

Figura 2. 1 - Evento enviado pelo Usuário Agente	18
Figura 2. 2 - Android e sinal Wi-Fi	22
Figura 2. 3 - Sinais emitidos via satélite.....	23
Figura 2. 4 - Modelo Simplificado Iterativo.....	24
Figura 2. 5 - Visualização que ocorrem no prédio	25
Figura 2. 6 - Visualização do objeto dentro do prédio	26
Figura 2. 7 - Quadro feito com lego	27
Figura 2. 8 - Projetos e seus usuários	28
Figura 2. 9 - Transferindo dados através do celular	28
Figura 3. 1 - Localização <i>Outdoor</i> e localização <i>Indoor</i>	31
Figura 3. 2 - UML mostrando arquitetura MVC do aplicativo	32
Figura 3. 3 - Implementação do GPS e Google Maps	33
Figura 3. 4 - Implementação da classe <code>WifiManipulator</code>	34
Figura 3. 5 - Sensor de Sensação tátil.....	35
Figura 3. 6 - Implementação do método <code>setVibrator</code>	36
Figura 3. 7 - Implementação do método <code>abrirWeb</code>	36
Figura 4. 1 - Localização <i>Outdoor</i> usando Google Maps.....	38
Figura 4. 2 - Localização <i>Indoor</i>	39
Figura 4. 3 - Implementação do sensor de sensação tátil	40
Figura 4. 4 - Captando a face do usuário utilizando a câmera.....	41
Figura 4. 5 - Indicando o local através da localização <i>Outdoor</i>	42
Figura 4. 6 - Tela mostrando o conteúdo para o aluno	43
Figura 4. 7 - Medindo a satisfação do aluno	44

SUMÁRIO

1- INTRODUÇÃO.....	12
1.1- Motivação.....	12
1.2- Objetivo.....	13
1.3- Metodologia	13
1.4- Organização do trabalho.....	14
2- REVISÃO BIBLIOGRÁFICA.....	15
2.1- Ambientes Responsivos	15
2.2- Sensores.....	17
2.2.1- Sensores de Visão	17
2.2.2- Localização.....	21
2.2.3- Sensação Tátil	24
2.3- Exemplos de Ambientes Responsivos Utilizando Sensores	25
3- DESENVOLVIMENTO	30
3.1- Arquitetura do Ambiente Responsivo Utilizando Sensores	30
3.2- Implementação do Sensor de Localização Externa.....	32
3.3- Implementação do Sensor de Localização Interna	34
3.4- Implementação do Sensor de Sensação Tátil	34
3.5- Implementação do Sensor de Visão	36
4- RESULTADOS	38
4.1- Utilização do Sensor de Localização Externa	38
4.2- Utilização do Sensor de Localização Interna	39
4.3- Utilização do Sensor de Sensação Tátil	40
4.4- Utilização do Sensor de Visão	41
4.5- União dos Sensores: Ambientes Responsivos.....	41
5- CONSIDERAÇÕES FINAIS	46
5.1- Contribuições e Conclusões	46
5.1.1- Publicação	47
5.2- Trabalhos Futuros.....	47
6- REFERÊNCIAS BIBLIOGRÁFICAS	48

1- INTRODUÇÃO

1.1- Motivação

Atualmente, pode-se notar uma falta de sincronia entre o mundo real e o mundo virtual (SLATER, 1997). Por exemplo, imagine que ao chegar em casa depois de um dia exaustivo, querendo descansar e curtir o restante do dia com a família e de repente o celular vibra. Pensando que é apenas uma notificação de alguma rede social (BARNES, 1972), o e-mail (GRINTER, 2003) recebido é referente ao trabalho enviado pelo seu chefe querendo algumas explicações sobre como foi à reunião de hoje. Os contextos não estão nos seus lugares corretos.

Outro exemplo de informações fora do contexto é estar dirigindo e de repente o celular toca. Consequentemente o motorista pega o celular para atender uma ligação ou visualizar uma mensagem. Muitos acidentes de trânsito são causados por pessoas que estavam mexendo no celular na hora em que estavam dirigindo (KAREL, 1991).

A tecnologia vem aprimorando e tentando ajustar os contextos para sincronizar os mundos reais e virtuais (BARFIELD, 1993). Existem celulares que ao identificar que o usuário está em movimento, ativa o sistema de voz ditando mensagens ou atendendo as chamadas, entre outros comandos úteis fazendo com as informações não fiquem fora de contexto (MOTOROLA, 2014).

Outra situação muito constante é o contexto dentro da sala de aula. O professor passou horas do dia preparando a matéria, para que na aula possa estar explanando o conteúdo da mesma e o aluno possa entender e absorver da melhor forma possível, mas os alunos no meio da aula recebem mensagens do Whatsapp (KAREN, 2013) ou estão acessando o Facebook. As mensagens com certeza não são sobre a aula, e nem muito menos do professor que está explicando a matéria (BOUHNICK, 2014) e sim, de algum amigo convidando para sair, da namorada querendo alguma coisa ou até mesmo da esposa pedindo para não se esquecer de passar na padaria.

Entretanto, as informações que recebemos hoje são constantes e volumosas sem o devido controle, necessitando filtrá-las a todo instante para evitar uma mistura de contextos como nos exemplos acima citados.

Vários problemas de informações fora do contexto estão relacionados à rede, sensores de médio acesso, protocolos de transporte para a sua interligação, *softwares*, sistemas e algoritmos para equipamentos, tais como localização e agregação por falta de um filtro mais específico. A forma de acessar estas redes pelos usuários é por meio de dispositivos, como um

assistente digital pessoal (PDA), telefone celular, computador portátil, ou estação de trabalho (STANLEY, 2008).

A técnica de *Cross-reality* (também conhecido como realidade-x) é uma troca de informação ou de mídia entre sistemas do mundo real e virtual (BARGAR, 1994). Especificamente, a realidade afetam ambos os mundos real e virtual, a ponto de suas plataformas terem um efeito significativo e perceptível sobre esse mundo particular (COLEMAN, 2009).

1.2- Objetivo

O objetivo deste trabalho é apresentar o desenvolvimento de ambientes responsivos utilizando sensores que permitirão que informações digitais estejam sincronizadas com seus devidos contextos por conta do grande número de dados disponíveis atualmente e a apresentação de alguns dados fora do contexto.

1.3- Metodologia

A criação de ambientes responsivos é capaz de filtrar cada informação recebida e encaixá-las nos seus devidos contextos. Para a realização desse projeto, foram utilizados os sensores do celular tais como vibração, visão, localização externa e interna.

Para localização *Outdoor* ou externa foi usado o sinal do GPS e para a localização *Indoor* ou interna foi usado o sinal da Wi-Fi. Usa-se a localização *Outdoor* em conjunto com a localização *Indoor* para aumentar a precisão da localização do usuário.

O uso do GPS foi utilizado para indicar a localidade em que o usuário se encontra através das coordenadas de latitude e longitude fornecidas pelo dispositivo do *smartphone* como, por exemplo, a entrada num *campus*.

Para ter uma precisão melhor de localização ao entrar em um prédio ou local fechado foi utilizado a localização *Indoor* utilizando o sinal da Wi-Fi fazendo com que o aplicativo obtenha a localização através do sinal emitido pelo roteador.

Para notificar o usuário foi utilizado sensor de vibração. A vibração é um dos meios mais usados para notificação em aplicativos para *smartphones* e *tablets*.

Foi utilizado câmeras para identificar o usuário e medir a sua satisfação em tempo real ao visualizar o material disponibilizado.

1.4- Organização do trabalho

Este trabalho está organizado nos seguintes Capítulos:

- O Capítulo 2 representa a revisão bibliográfica e as fundamentações teóricas utilizadas no projeto apresentando os conceitos e estudos relacionados a este trabalho.
- O Capítulo 3 representa o desenvolvimento abordando a implementação da aplicação utilizando os sensores existentes no dispositivo móvel.
- O Capítulo 4 representa os resultados obtidos através do desenvolvimento.
- O Capítulo 5 representa as considerações finais e apresenta os trabalhos futuros relacionados a este trabalho.

2- REVISÃO BIBLIOGRÁFICA

Este Capítulo descreve sobre o ambiente responsivo, visão computacional e exemplos de aplicações. A Seção 2.1 explica sobre o que são os ambientes responsivos e a Seção 2.2 explica sobre a visão computacional. Na Seção 2.3, contém exemplos reais de aplicações mostrando como é possível aplicar os conceitos explicados nas Seções 2.1 e 2.2.

2.1- Ambientes Responsivos

Um sistema de ambientes inteligentes aumenta o espaço físico através da tecnologia por meios de comunicação e serviços (LINO, 2010). São ambientes inteligentes que utilizam deste acesso para aumentar a qualidade da experiência que um serviço ou conteúdo está fornecendo.

A história de *interface* e interação do projeto tende a ser uma das simplificações da complexidade das mesmas. A tendência atual é a utilização de tecnologia embarcada, invisível para o usuário que suporta a interação dentro de ambientes.

Os ambientes responsivos contam com a interação natural que é definida em termos de experiência. As pessoas se comunicam naturalmente através de gestos, expressões, movimentos, manipulação de objetos, proporcionando aos usuários a percepção de interação é real e usual (VALLI, 2007).

Um ambiente responsivo é também um espaço que convida certas emoções, ações e estado de espírito (WU, 2009) a partir da interação com o usuário através da visão. A percepção visual pode ser dividida em duas fases: a recepção física do estímulo do mundo exterior e, o tratamento e a interpretação do que é o estímulo.

Pesquisas recentes em psicologia e tecnologia sugere uma visão diferente da relação entre os seres humanos, computadores e emoção, como por exemplo, a Realidade Virtual (TORI, 2006) que está sendo utilizada para auxiliar terapeutas em diversos tratamentos (DE CARVALHO, 2008). Depois de um longo período de dormência e confusão, houve uma explosão de pesquisas sobre a psicologia da emoção (GROSS, 1999).

A emoção não é mais visto como limitado à explosão ocasional de fúria quando um computador trava inexplicavelmente, excitação quando um personagem de videogame pula passando um obstáculo, ou frustração em uma mensagem de erro incompreensível.

Compreende-se agora que uma ampla gama de emoções desempenha um papel crítico em cada atividade dirigida a objetivos relacionados com a informática, desde o

desenvolvimento de um modelo 3D e execução de cálculos em uma planilha, pesquisar na Web, enviar um e-mail para realização de uma compra *online* ou jogar paciência (SHOTTON, 2013). De fato, muitos psicólogos argumentam que é impossível que uma pessoa possa ter um pensamento ou executar uma ação sem se envolver, pelo menos inconscientemente, aos seus sistemas emocionais (COWIE, 2001).

A emoção é um componente fundamental do ser humano. Alegria, ódio, raiva e orgulho, entre a infinidade de outras emoções motiva a ação e adiciona significados e riquezas, para praticamente toda a experiência humana. Tradicionalmente, a Interação Humano Computador (HCI) tem sido vista como uma exceção “final”. Os usuários devem descartar seu eu emocional para trabalhar de forma eficiente e racional com computadores tornando o artefato essencialmente sem emoção.

Interação homem máquina é geralmente limitada a um homem sentado tocando a tela com os dedos ou talvez um toque de uma caneta em um *tablet*. O resultado foi o conceito de um ambiente receptivo em que o computador ou outro dispositivo percebesse as ações de quem interage e responde de forma inteligente através de *displays* visuais e auditivos complexos (PICARD, 1995).

Ambientes responsivos exploram as redes de sensores aumentando a experiência humana e a interação e percepção durante o desenvolvimento de novas modalidades de detecção, permitindo que tecnologias que criam novas formas de experiência interativa e de expressão. Engloba o desenvolvimento de aplicações de vários tipos de redes de sensores, captação de energia e gerenciamento de energia e a base técnica de computação ubíqua. Os ambientes responsivos são destaques em diversas áreas de aplicação que incluam sistemas automotivos, autoestradas inteligentes, instrumentação médica, RFID, computação vestível (BARROS, 2013) e mídia interativa.

Utilizando o meio ambiente como uma plataforma para a interação com o conteúdo de informações ou serviços permite aos usuários interagir com essas informações de uma forma natural, tendo a sensação de que há uma compreensão mútua entre eles e da aplicação (KRUEGER, 1994).

Acessibilidade em geral pode ser universal, em grande área, local ou ponto único dependendo do objetivo que o sistema está tentando alcançar.

O ambiente responsivo tem sido apresentado como a base para um novo meio de estética baseado na interação em tempo real entre homens e máquinas. No longo alcance das experiências humanas, as realidades artificiais que não procuram simular o mundo físico tornam impossível definir relações arbitrárias e abstratas a ação e os resultados. Além disso,

tem sido sugerido que os conceitos e ferramentas dos ambientes de resposta podem ser proveitosamente aplicados em vários campos (KUUTTI, 1996).

O que talvez tenha sido obscurecido é que esses conceitos são o resultado de uma necessidade pessoal de compreender e expressar a essência do computador em termos humanistas (KRUEGER, 1977).

Ambientes responsivos é uma alternativa aos sistemas multimídia e realidade virtual da década passada. Analisando as situações do cotidiano da utilização dos computadores de diferentes usuários, como cientistas, arquitetos, pilotos, médicos e o pessoal de serviço em agências de viagens e balcões de trabalho, é possível reconhecer que quase ninguém quer simulações de seus mundos em um ambiente *desktop*. Geralmente, os usuários querem se concentrar em suas tarefas, em vez de operar o computador. Os sistemas de computadores futuros devem utilizar e adaptar-se a rica vida humana e aos ambientes de trabalho, tornando-se parte de um ambiente mais receptivo (PAVLOVIC, 1997).

2.2- Sensores

Este Capítulo descreve sobre os sensores do sistema e está dividido nas seguintes seções: a Seção 2.2.1 explica sobre o sensor de visão, a Seção 2.2.2 explica sobre o que é a localização e na Seção 2.2.3 explica sobre o sensor tátil.

2.2.1- Sensores de Visão

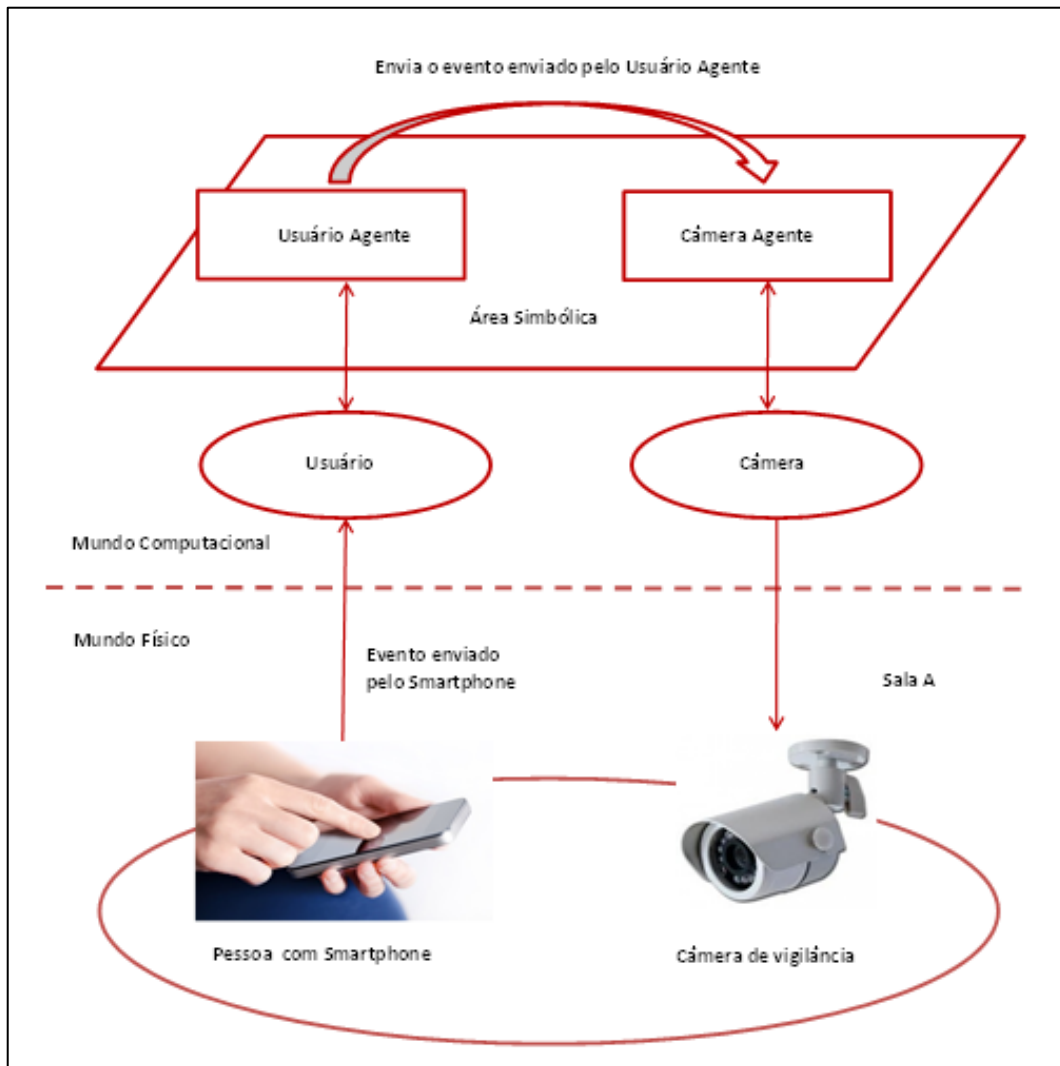
As câmeras modernas são capazes de concentrar automaticamente os rostos das pessoas e acionar o obturador quando sorriem. Sistemas de reconhecimento de texto ajudam a transformar os documentos digitalizados em texto que pode ser analisado ou lido em voz alta por um sintetizador de voz (SONZA, 2003). Sistemas de vigilância por vídeo inteligente desempenham um papel cada vez mais importante na monitorização da segurança de áreas públicas.

Os dispositivos móveis, como *smartphones* e *tablets*, vêm equipados com câmeras e mais poder de computação, fazendo com que a demanda por aplicações com visão computacional aumentem.

A Figura 2.1 mostra um exemplo de maneira simples à ativação de um ambiente inteligente a partir da interação de dois dispositivos: um dispositivo móvel carregado por uma

pessoa e uma câmera de vigilância com capacidade de se movimentar em diferentes direções sobre seu eixo instalado em um espaço físico.

Figura 2. 1 - Evento enviado pelo Usuário Agente



Fonte: AUTOR (2015)

A especificação seguindo a definição de um ambiente inteligente como organizações de agentes definem os papéis usuário e câmera que estão no mundo computacional e a interação entre estes, como o envio de eventos do primeiro para o segundo papel.

Os agentes associados aos dispositivos que se encontram no mundo físico são: um *smartphone* carregado por uma pessoa e uma câmera de vigilância. O usuário é hospedado no dispositivo móvel e envia um evento. Já a câmera controla a vigilância através de um

componente, que acessa o *driver* do equipamento e permite movimentá-la.

A interação é iniciada toda vez que um alguém é capaz de adotar o papel de usuário quando entra na área simbólica que representa a sala de um prédio onde a câmera está instalada. Esta interação acontece enquanto o agente se encontra dentro da mesma.

O ambiente inteligente é ativado pela execução desta interação. Quando a pessoa que carrega o *smartphone* entra na sala do prédio é gerado um evento. Desta maneira, o usuário envia um evento para a câmera informando sua localização e outras informações. A câmera recebe o evento enviado e se encarrega de movimentá-la na direção da localização do *smartphone*. Este evento é enviado repetidamente até a pessoa sair da sala.

É necessário notar os seguintes pontos neste processo:

- 1) O usuário agente registra os papéis que pode adotar por decisão própria com o propósito de fazer parte deste ambiente inteligente.
- 2) A localização do usuário agente é dada pela localização do *smartphone* dentro da sala e, portanto, da pessoa que o carrega.
- 3) São conhecidos os papéis que podem adotar os dois agentes.
- 4) Mais de um agente pode entrar na área simbólica e adotar o papel usuário agente. Cada um destes agentes inicia a interação descrita.
- 5) A câmera agente é o encarregado de processar e escolher (por decisão própria) como movimentar a câmera de vigilância para rastrear uma pessoa dentro da sala.
- 6) A implementação da câmera agente pode conter mecanismos de detecção de faces para focar a câmera da melhor maneira na pessoa que carrega o *smartphone*.

O modelo é encarregado de efetuar os seguintes pontos neste processo:

- 1) Detecção da entrada do *smartphone* na área simbólica.
- 2) Seleção da interação definida na especificação do ambiente.
- 3) Seleção do usuário agente e câmera agente para fazer parte da interação.
- 4) Envio de eventos entre o usuário agente e câmera agente informando a localização do *smartphone*.

Abaixo segue uma leve explicação sobre mundo físico e mundo computacional descritos no exemplo da Figura 2.1.

Mundo físico: este mundo está regido pelas leis da física como localização, gravidade, tempo, espaço, temperatura, etc. Neste mundo reconhecemos principalmente as seguintes

entidades: pessoas, dispositivos e espaços físicos.

Mundo computacional: neste mundo se encontram quaisquer tipos de sistemas computacionais e consideram-se agentes e suas interações como principais entidades.

Estes mundos estão associados um ao outro no sentido que interações no mundo computacional são refletidas no mundo físico e vice-versa. Esta associação entre os mundos é efetivada exclusivamente por agentes, isto é, os agentes estão associados aos dispositivos no sentido que interações com um agente no mundo computacional podem produzir ações no dispositivo associado ao agente. E no outro sentido, interações com dispositivos no mundo físico são refletidas no agente associado ao dispositivo e podem produzir ações no mundo computacional (VIZZARI, 2004).

Dispositivos com câmeras tornaram-se inteligente o suficiente para fundir várias fotografias em um panorama de alta resolução, para ler um código QR, reconhecê-lo e recuperar informações sobre um produto a partir da *Internet* (VALLÉE, 2005).

A visão computacional é computacionalmente caro, no entanto, mesmo um algoritmo dedicado a resolver um problema muito específico como da foto panorâmica ou a detecção de um sorriso, requer uma grande quantidade de energia e esforço.

Muitos cenários de visão computacional devem ser executados em tempo real, o que implica no tratamento de um único quadro que deve concluir dentro de 30 a 40 milissegundos. Esta é uma das necessidades de recursos muito desafiadores especialmente para arquiteturas de computação móveis e incorporadas.

Por exemplo, o algoritmo de foto panorâmica pode encontrar mais imagens de origem e sintetizar uma imagem de maior qualidade, levando muito mais tempo para ser processado. Para atender às limitações de tempo e orçamento computacional, os desenvolvedores devem escolher em comprometer a qualidade criando algoritmos para processar imagens mais rápido ou investir mais tempo para otimizar o código para arquiteturas de *hardware* específicos.

O conceito de combinar essas duas ideias, isto é, a execução de um ou mais CPUs em conjunto com vários aceleradores é chamado de computação paralela heterogênea (SCHMIDT, 2002).

Tarefas de alto nível de visão computacional muitas vezes contêm sub-tarefas que podem ser executados mais rápidos em arquiteturas de *hardware* de propósito especial do que do processador, enquanto outras sub-tarefas são computados na CPU. Embora as regras do conjunto de instruções x86 da Intel são para computadores de mesa, quase todos os *smartphones* e *tablets* (ADAMS, 1993) são alimentados por processadores da ARM. Processadores ARM seguem a abordagem RISC (conjunto de instruções reduzido), como se

pode deduzir do nome original da ARM, *Advanced RISC Machines*. Enquanto processadores x86, tradicionalmente concebidos para alta potência computacional, processadores ARM foram projetados principalmente para o uso de baixa potência, o que é um claro benefício para dispositivos alimentados por bateria.

2.2.2- Localização

Recentemente as localizações *Indoor* ou interna são feitas através do uso do sinal de comunicação *Wireless*. A maior vantagem de utilizar esse recurso é por não necessitar de recursos ou aparelhos mais sofisticados. Os serviços providos podem ser acessados através dos sinais de *Access Points* de roteadores Wi-Fi e serem captados pelos *smartphones* (CASTRO, 2001).

A rede Wi-Fi consiste em vários sinais emitidos por *Access Points* fazendo uma ponte entre os sinais. Os avanços recentes em sistemas de localização *Indoor* utilizam infraestrutura de comunicação sem fio, por exemplo, 802.11 e GSM para fornecer um serviço em local existente. A grande vantagem para essas abordagens é que o consumidor não tem que comprar qualquer equipamento especializado e pode ainda se beneficiar da localização computacional e os parâmetros operacionais de pontos de acesso Wi-Fi e torres de celular com pouco sinal (EVENNOU, 2006) são facilmente ajustáveis.

Essas mudanças exigem calibração do sistema de localização para introduzir novas infraestruturas distribuindo muitos aparelhos de baixo custo e balizas de curto alcance para o aumento de sinal. O tempo necessário para a instalação e o possível impacto para a estética, no entanto, pode limitar a adoção desses recursos.

A rede *Wireless LAN* (KAMERMAN, 1997) existe na maioria dos locais. O *hardware* é barato para comprar e facilmente disponível nas prateleiras. Mais e mais escritórios, organizações, residências e lugares públicos tem esse tipo de rede instalada por sua facilidade de utilização. O desenvolvimento de serviços de localização *Indoor* usando WLAN (YOUSSEF, 2003), portanto, ganhou enorme interesse nos últimos anos. É, no entanto, ainda um grande desafio para a comunidade de investigação para melhorar a precisão de localização dentro de um edifício a poucos metros por meio de computação, sem perda de caminho para encontrar distâncias com base na intensidade de um sinal recebido pelo roteador.

A precisão oferecida por vários produtos utilizando métodos de intensidade dentro do ambiente interior não é muito bom. Alguns trabalhos apresentam novos métodos para

conseguir uma melhor consistência na intensidade do sinal recebido de um sinal do roteador através de uma eficaz redução do caminho, com efeito sobre a propagação do sinal no interior do edifício, permitindo assim o desenvolvimento de um sistema com melhor estimativa de localização.

A Wi-Fi é cada vez mais usada e o modo preferido de ligação à *Internet* em todo o mundo. Para aceder a este tipo de conexão, é preciso ter um adaptador sem fio em seu computador ou um dispositivo móvel. A Wi-Fi fornece conectividade sem fio através da emissão de frequências entre 2,4 GHz a 5 GHz com base na quantidade de dados na rede. Áreas que são habilitados com conectividade Wi-Fi são conhecidas como *hot spot*. Para iniciar uma conexão sem fio é importante que o roteador *wireless* esteja ligado à *Internet*, e que todas as definições se encontram devidamente instalados (HENRY, 2002).

A rede sem fio é conhecido como Wi-Fi (*Wireless Fidelity*) ou rede 802.11, pois abrangem os protocolos IEEE 802.11. A grande vantagem do Wi-Fi é que ele é compatível com a maioria dos sistemas operacionais, dispositivos móveis e impressoras avançadas.

Em telefones celulares, uma rede Wi-Fi faz uso de ondas de rádio para transmitir informações através de uma rede. Para que um computador receba os sinais Wi-Fi, o mesmo deve incluir um adaptador sem fio que irá traduzir os dados enviados em um sinal de rádio. Este mesmo sinal é transmitido através de uma antena para um decodificador conhecido como roteador conforme ilustra a Figura 2.2.

Figura 2. 2 - Android e sinal Wi-Fi



Fonte: AUTOR (2015)

Uma vez decodificado, os dados são enviados para a *Internet* através de uma conexão *Ethernet* com fio. À medida que a rede sem fio irá funcionar como um tráfego de mão dupla,

os dados recebidos a partir da *Internet* também vão passar através do roteador e ser codificada em um sinal de rádio que será liquidada pelo adaptador sem fio do computador.

O Sistema de Posicionamento Global (GPS) é um dispositivo de navegação composto por uma rede de 24 satélites colocados em órbita pelo Departamento de Defesa dos EUA conforme demonstra a Figura 2.3. Foi originalmente planejado para ser usado em aplicações militares, mas na década de 1980, o governo fez o sistema disponível para uso civil. O dispositivo funciona em quaisquer condições meteorológicas, em qualquer lugar do mundo, 24 horas por dia. Não há taxas de assinatura ou taxa de instalação para usar o equipamento.

Figura 2. 3 - Sinais emitidos via satélite



Fonte: AUTOR (2015)

Os satélites circundam a Terra duas vezes por dia em uma órbita muito precisa e transmite informações para terra. Os receptores de GPS obtém essa informação e usa a triangulação para calcular a localização exata do usuário. Essencialmente, o receptor compara o tempo que um sinal foi transmitido por um satélite com o momento em que foi recebido. A diferença de tempo diz ao receptor o quanto longe o satélite está. Agora, com medidas de distância de mais alguns satélites, o receptor pode determinar a posição do usuário e exibi-lo no mapa eletrônico da unidade (SCOTT, 2001).

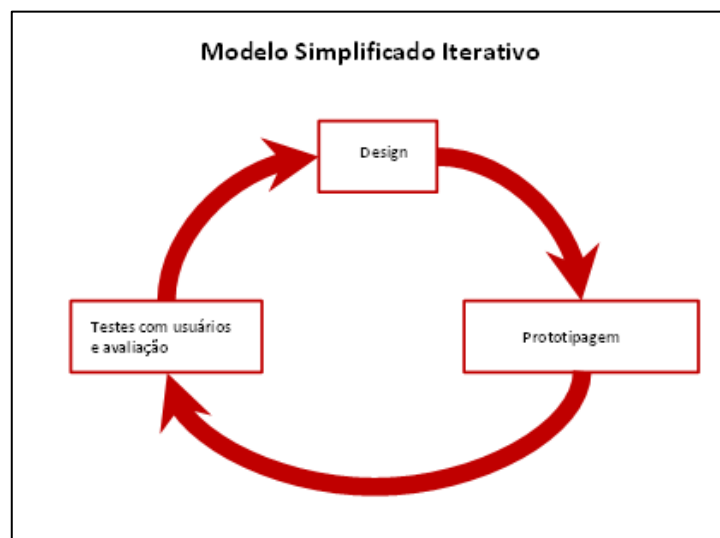
2.2.3- Sensação Tátil

A ferramenta de manipulação de mais importante e natural para ambientes virtuais é a mão do usuário. Nosso ambiente depende da mão (tato) e não de uma representação gerada por computador.

Algoritmos de reconhecimento de gestos e de detecção de colisão calculam a interação do usuário com os objetos do mundo virtual. A metáfora padrão de Interação Humano-Computador surgiu a partir da experiência diária de um trabalhador de escritório de colarinho branco. Nos últimos 20 anos, os sistemas *desktop* foram aprimorados cada vez mais fornecendo ferramentas como linha e rastreadores gráficos, *interfaces* para o usuário usando *mouse* entre outras extensões de multimídia avançados. Com o advento dos ambientes virtuais imersivos o usuário finalmente chegou a um espaço 3D (HOLLAN, 2000).

A manipulação de objetos virtuais e reuniões com colaboradores sintetizados têm sido propostos como *interfaces* de computador e humanidade especiais para o processo de visualização científica conforme mostra a Figura 2.4.

Figura 2. 4 - Modelo Simplificado Iterativo



Fonte: AUTOR, 2015

Outra abordagem para o problema de projeto para futuras *interfaces* homem-computador se baseia nas primeiras ideias de ambientes multimídias interativas e imersivas. Ambientes de visualização orientada para a aplicação têm sido propostos e construídos para suportar um processo de resolução de problemas específicos. O computador atua como um

servidor inteligente em segundo plano fornecendo informações necessárias em todos os canais de interação multissensoriais (MACKENZIE, 1992).

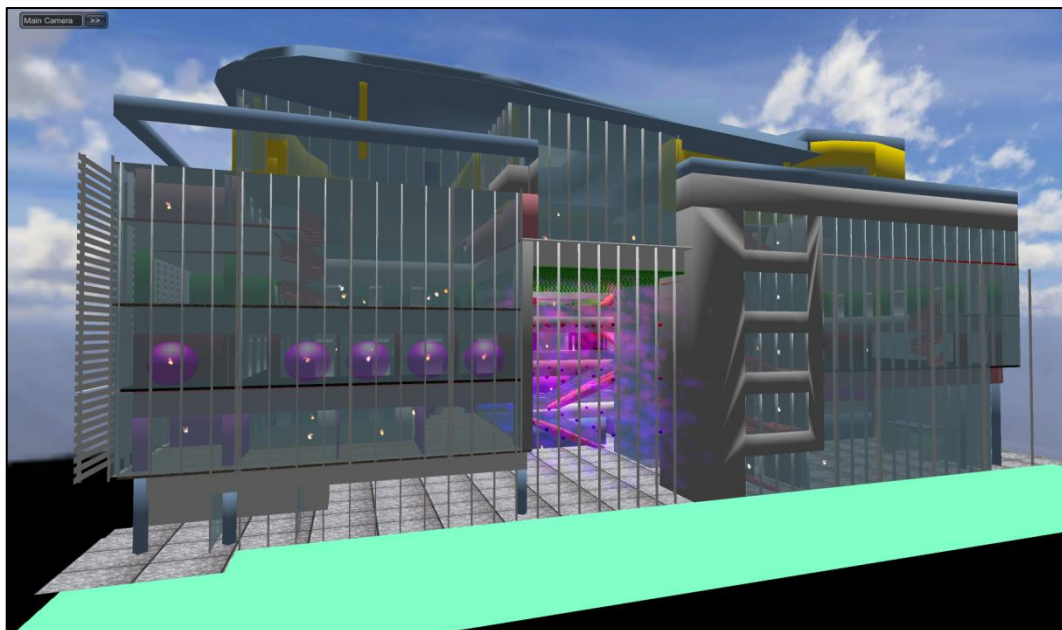
2.3- Exemplos de Ambientes Responsivos Utilizando Sensores

DoppelLab é um ambiente imersivo usando *Cross-reality* virtual que serve como um repositório ativo dos dados do sensor multimodais produzidos por um edifício e de seus habitantes. A plataforma criada pela DoppelLab aproveita a física e iluminação, animações interativas, como o abate da distância para estruturar e encorajar a exploração dos dados (DUBLON, 2011).

As representações visuais de dados assumem a forma de animações metafóricas que expressam magnitudes absolutas e relativas, as taxas de variação, e inferências de alto nível que se relacionam com dados de diferentes modalidades de tempo, de espaço e de sensoriamento.

As visualizações ocorrem dentro das paredes do edifício correspondendo aos locais físicos dos sensores, e as paredes podem ser alternadas em translucidez para expor as relações que se estendem por todo o edifício conforme ilustra a Figura 2.5.

Figura 2. 5 - Visualização que ocorrem no prédio

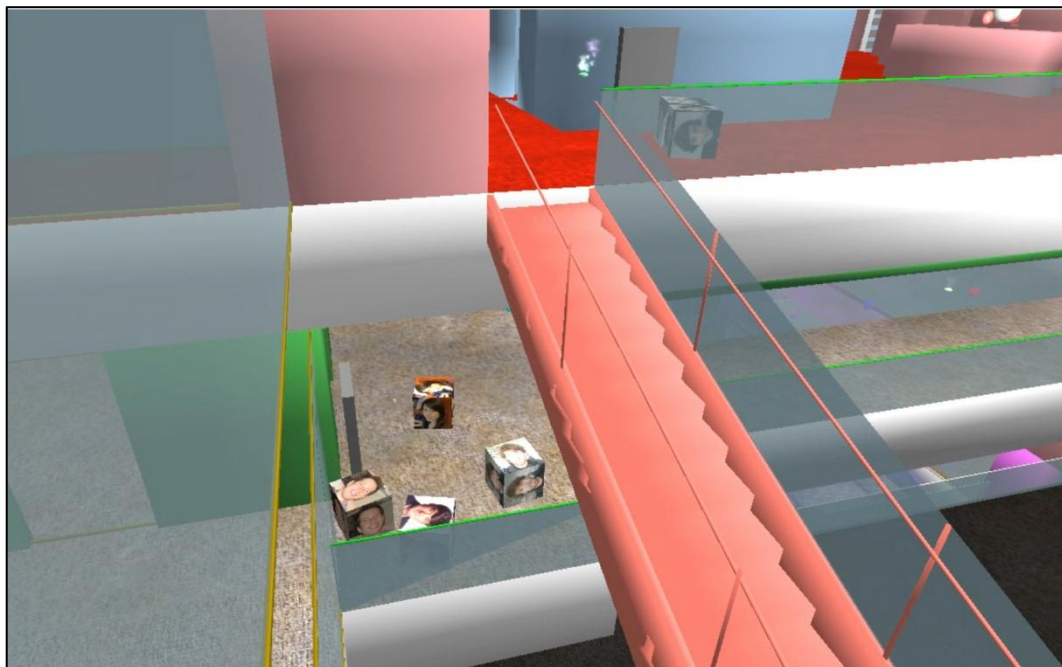


Fonte: DOPPELLAB (2015)

Organizando os dados desta forma fornece uma plataforma amplamente intuitiva que enfatiza a relação entre as pessoas e seus ambientes físicos. Enquanto projeções e transformações para outros espaços podem produzir representações mais eficientes de dados para muitos tipos de consultas, o espaço físico serve como um ponto de partida consistente e relacionável.

O sistema utiliza a comunicação do servidor e analisa as mensagens de visualizações de cada rede. Se nenhum dado for encontrado para um determinado sensor, a animação correspondente está desativada. O sistema relaciona seu sistema de coordenadas para o prédio planta baixa e passa mensagens e dados em conformidade com o nível inferior que gerenciam visualizações e animações usando esses dados para controlar as propriedades de objetos no ambiente. Estes parâmetros são de visualização dependente, e incluem as propriedades do objeto como a cor, forma e tamanho, ou no caso de sistemas de partículas, as propriedades sistêmicas como taxa de emissão e vida representada por quadrados com as ilustrações das pessoas conforme mostra a Figura 2.6.

Figura 2. 6 - Visualização do objeto dentro do prédio

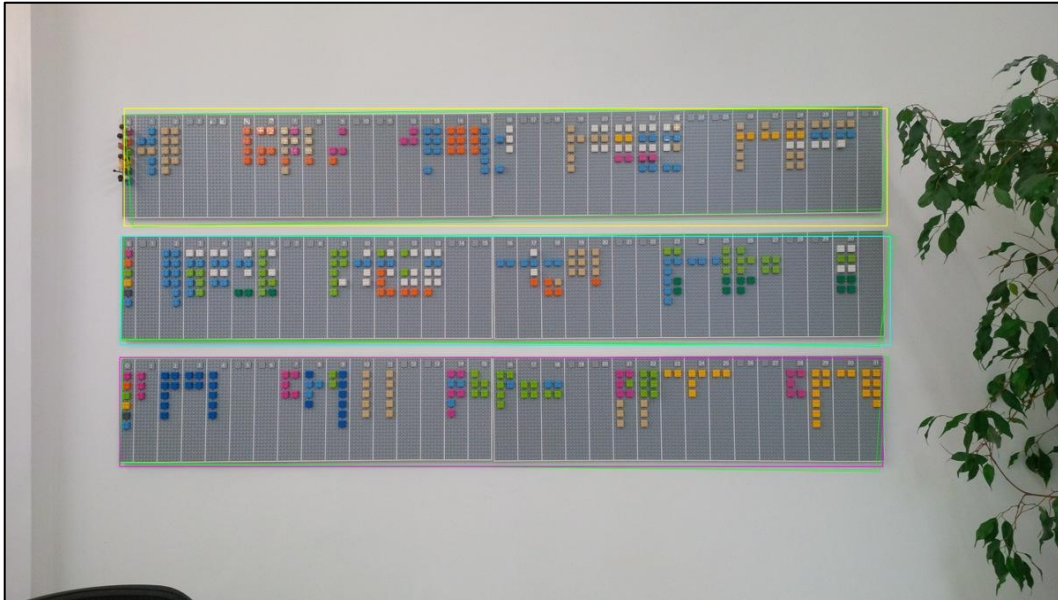


Fonte: DOPPELLAB (2015)

A Figura 2.7 mostra o calendário Lego criado por Vitamins Design que é uma parede do tempo que a equipe inventou para o seu estúdio. É feito inteiramente de Lego e a solução de

Vitamins permite que você tire uma foto dela com um *smartphone* de todos os eventos e horários que são sincronizados para um calendário digital.

Figura 2. 7 - Quadro feito com lego



Fonte: VITAMINS (2015)

No calendário, cada linha representa um mês e cada coluna representa um dia da semana. Cada pessoa no estúdio tem sua própria linha e cada projeto tem sua própria cor conforme mostra a Figura 2.8.

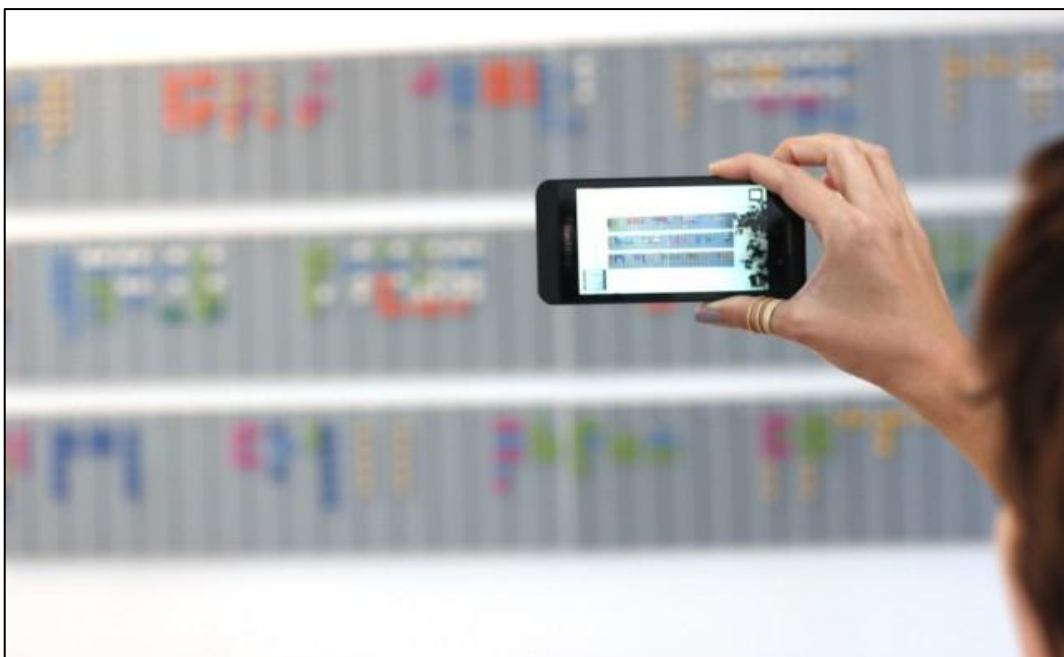
Figura 2. 8 - Projetos e seus usuários



Fonte: VITAMINS (2015)

Cada tijolo representa metade de um dia passado e qual projeto em particular. Toda vez que alguém muda algo no calendário, tudo que eles precisam fazer é tirar uma foto do quadro com qualquer *smartphone* como mostra a Figura 2.9.

Figura 2. 9 - Transferindo dados através do celular



Fonte: VITAMINS (2015)

Depois, eles enviam a foto por e-mail para um endereço que eles criaram. Alguns segundos depois, o remetente recebe um e-mail confirmando que todas as informações sobre o calendário de lego foi sincronizada ao seu calendário digital. O sistema utiliza atualmente o Google Calendar, mas isso poderia funcionar com qualquer calendário baseado em nuvem.

3- DESENVOLVIMENTO

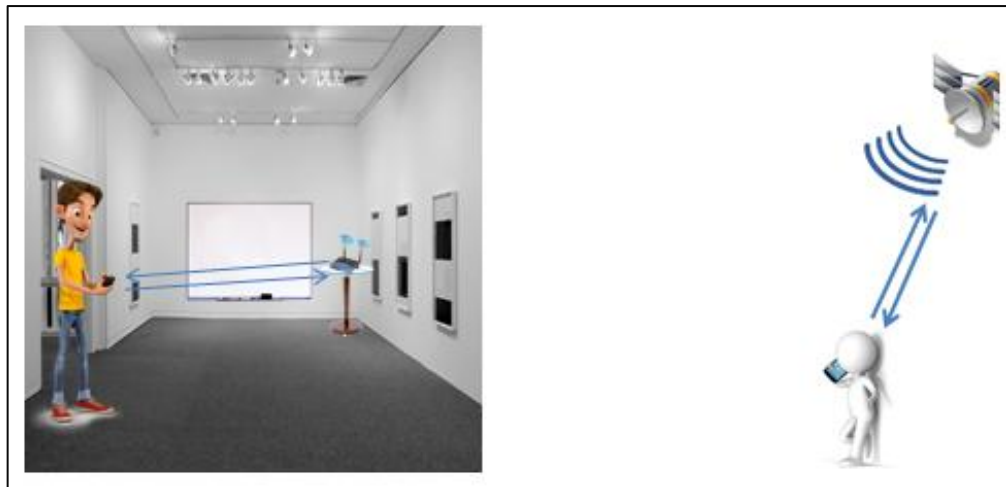
Este Capítulo descreve sobre o desenvolvimento do sistema e está dividido nas seguintes seções: a Seção 3.1 explica sobre a arquitetura do sistema, a Seção 3.2 explica sobre o que é a localização *Outdoor*, na Seção 3.3 explica sobre a localização *Indoor*, na Seção 3.4 explica sobre o sensor Tátil e na Seção 3.5 explica sobre o sensor de visão.

3.1- Arquitetura do Ambiente Responsivo Utilizando Sensores

O aplicativo desenvolvido utiliza os sensores de um *smartphone* ou *tablet* para oferecer ao usuário uma experiência relacionada ao seu contexto. Quando a pessoa chega a um local como, por exemplo, o prédio da Fatec de São José dos Campos, o sensor de GPS identifica sua posição externa. Para que a identificação de localização interna seja reconhecida, é necessário realizar o mapeamento do roteador para que, ao entrar no recinto, sua posição seja identificada pelo sensor de Wi-Fi, de acordo com o nível de proximidade com o ponto de acesso do roteador permitindo assim a localização nas salas e corredores do prédio. Ao entrar na sala, o sensor de vibração alertará o usuário que informações estarão disponíveis para aquele contexto. A pessoa poderá então verificar na tela do seu *smartphone* o conteúdo e nela serão exibidos os materiais relativos àquela aula específica.

O sistema reconhece a localização do dispositivo de duas formas, a *Indoor* e a *Outdoor*, que serão explicadas nas seções 3.2 e 3.3. A Figura 3.1 ilustra como o dispositivo móvel recebe as informações de localização através de um roteador e através de sinais via satélite. Ao receber o sinal através de uma interação ao toque, a tela notifica usando o sensor de vibração.

Figura 3. 1 - Localização *Outdoor* e localização *Indoor*



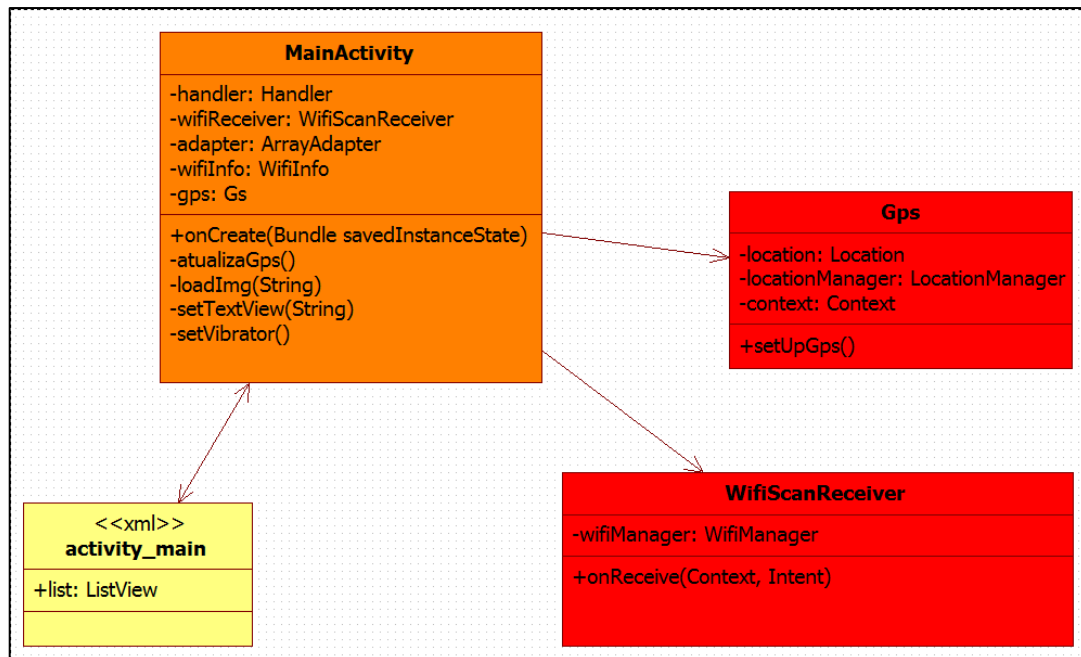
Fonte: AUTOR (2015)

A Figura 3.2 descreve como foram modeladas as classes da aplicação, seguindo a arquitetura MVC (GAMMA, 1994), que visa separar as camadas para melhorar a estrutura facilitando a manutenção, devido o aumento da complexidade do projeto.

Esse padrão é dividido em três camadas:

- Model: camada responsável pelos objetos representativos de entidades e armazenamento.
- View: camada responsável pela renderização do modelo para a interface do usuário.
- Controller: camada responsável por responder a processos e eventos vindos da View.

Figura 3. 2 - UML mostrando arquitetura MVC do aplicativo



Fonte: AUTOR (2015)

O xml `activity_main` é a View do Android onde irá aparecer todas as informações para o aluno. A classe `MainActivity` é a classe controladora, camada responsável por controlar as ações da aplicação. As classes `Gps` e `WifiScanReceiver` são os modelos responsáveis por informar a localização.

3.2- Implementação do Sensor de Localização Externa

A Figura 3.3 mostra o uso do GPS indicando onde o dispositivo está usando a API do Google Maps (GOOGLE, 2015).

Figura 3.3 - Implementação do GPS e Google Maps

```

1 public class Gps implements LocationListener{
2
3     private Location location;
4     private LocationManager locationManager;
5     private Context context;
6     private GoogleMap googleMap;
7
8     public Gps(Context c, GoogleMap mMap){
9         location = null;
10        context = c;
11        googleMap = mMap;
12    }
13
14    public void setUpMap() {
15        // Enable MyLocation Layer of Google Map
16        googleMap.setMyLocationEnabled(true);
17
18        // Get LocationManager object from System Service LOCATION_SERVICE
19        locationManager = (LocationManager) context.getSystemService(Context.LOCATION_SERVICE);
20
21        // Create a criteria object to retrieve provider
22        Criteria criteria = new Criteria();
23
24        // Get the name of the best provider
25        String provider = locationManager.getBestProvider(criteria, true);
26
27        // Get Current Location
28        location = locationManager.getLastKnownLocation(provider);
29
30        // request that the provider send this activity GPS updates every 0 seconds
31        locationManager.requestLocationUpdates(LocationManager.GPS_PROVIDER, 0, 0, this);
32
33        // set map type
34        googleMap.setMapType(GoogleMap.MAP_TYPE_NORMAL);
35
36        // Get current position to get camera position
37        getCameraPosition(getCurrentPosition());
38    }

```

Fonte: AUTOR (2015)

Para implementar o GPS no android, é necessário usar três classes: MyLocationOverlay (MYLOCATIONOVERLAY, 2015), LocationManager (LOCATIONMANAGER, 2015) e a LocationListener (LOCATIONLISTENER, 2015).

A interface LocationListener obriga a classe implementada a escrever o método onLocationChanged, método que atualiza o atributo Location toda vez que o dispositivo de localização sofre uma alteração. A classe MyLocationOverlay é a classe responsável para desenhar o ponto no mapa, recebendo as informações necessárias como latitude e longitude através do atributo Location, que é preenchido pela classe LocationManager.

O método setUpMap ativa o myLocationEnable para que o mapa possa ser desenhado a cada alteração de localização. Recebe um serviço através do getSystemService, recebendo como parâmetro o tipo de serviço. O método

`requestLocationUpdates` é o método que atualiza a localização no mapa, recebendo como parâmetro o tempo em segundos.

3.3- Implementação do Sensor de Localização Interna

A Figura 3.4 mostra como o aplicativo lista as posições e os sinais possíveis captados dos roteadores.

Figura 3. 4 - Implementação da classe WifiManipulator

```

1 public class WifiManipulator {
2
3     WifiManager wifiManager;
4
5     public List<ScanResult> getConections(Context context){
6         wifiManager = (WifiManager)context.getSystemService(Context.WIFI_SERVICE);
7         wifiManager.startScan();
8         List<android.net.wifi.ScanResult> connections = wifiManager.getScanResults();
9         return connections;
10    }
11 }
12 }
```

Fonte: AUTOR (2015)

O reconhecimento *Indoor*, usa-se a classe `WifiManager` (WIFIMANAGER, 2005), que é responsável por manipular os sinais captados através do Wi-Fi. Com ela é possível listar, verificar e reconhecer redes Wi-Fi ativas, resultados de exames de pontos de acesso que contêm informações suficientes para tomar decisões sobre o *access point* para conexão entre outras funcionalidades.

3.4- Implementação do Sensor de Sensação Tátil

Mesmo quando o volume está baixo, ativar a funcionalidade de vibração é uma excelente forma para interação com os usuários. Isso pode garantir uma melhor experiência com o uso do aplicativo, pois, aumenta a interação com o *smartphone*.

A classe `Vibrator` (VIBRATOR, 2005) é responsável por manipular esse evento e descreve a *interface* para acionar o sensor de vibração do *smartphone*. Com ela, é possível fazer com que um dispositivo com o sistema operacional Android vibre em uma das seguintes maneiras:

- 1) Vibrar durante um determinado período de tempo.

- 2) Vibrar em um determinado padrão.
- 3) Vibrar repetidamente até ser cancelada.

Ao tocar na tela, o *smartphone* identifica o evento e vibra para notificar o usuário de sua ação conforme ilustra a Figura 3.5.

Figura 3. 5 - Sensor de Sensação tátil



Fonte: AUTOR (2015)

O modo de toque é um estado que depende unicamente da interação do usuário com o telefone. Por si só, esse modo é algo muito fácil de entender por simplesmente indicar se a última interação do usuário foi realizada com o toque (TOUCH, 2015).

Neste modo não há foco e nenhuma seleção. Qualquer item selecionado numa lista torna-se desmarcada ou marcada, logo que o usuário toca na tela. Ou quando há um botão para ser clicado e, que após esse evento ser chamado, o mesmo emite uma notificação na tela, por exemplo.

A Figura 3.6 mostra a implementação do código usando o método `Context.VIBRATOR_SERVICE` (CONTEXT, 2015) como o parâmetro da função `getSystemService` na linha 3.

Figura 3. 6 - Implementação do método setVibrator

```

1 private void setVibrator(){
2     // vibration for 300 milliseconds
3     ((Vibrator)context.getSystemService(Context.VIBRATOR_SERVICE))
4         .vibrate(300);
5 }

```

Fonte: AUTOR (2015)

O método `setVibrator` é acionado toda vez que a aplicação dispara algum evento que seja necessário dar um *feedback* a pessoa.

3.5- Implementação do Sensor de Visão

O *software* utilizou a detecção facial para medir a satisfação da pessoa em tempo real que é capaz de detectar a expressão (CLMTRACKR, 2015) do usuário, verificando se o mesmo ficou contente, com raiva, triste ou assustado a partir da resposta obtida através do aplicativo conforme a captura da face através da câmera.

Ao clicar no botão Ver Material que irá aparecer na tela, o método `abrirWeb` é chamado conforme mostra a Figura 3.7.

Figura 3. 7 - Implementação do método abrirWeb

```

1 public void abrirWeb(View view){
2
3     String url = URL;
4     Intent i = new Intent(Intent.ACTION_VIEW);
5     i.setData(Uri.parse(url));
6     startActivity(i);
7 }
8

```

Fonte: AUTOR (2015)

Para que a aplicação apresente outros comandos, o Android utiliza o recurso da `Intent` (INTENT, 2015), permite que o aplicativo Android defina comandos a serem executados. Enquanto a `Intent` declara o que deve ser feito, o componente que a recebe é o

responsável por definir como a ação será executada. Ou seja, para uma mesma ação, podem-se ter comportamentos distintos quando a ação for executada por componentes diferentes. Por exemplo, a `ACTION_VIEW` pode ser utilizada tanto para indicar que desejamos abrir uma página da Internet quanto para abrir informações de um contato armazenado no telefone.

Com a finalização do desenvolvimento, no próximo Capítulo serão apresentados os resultados de sua execução.

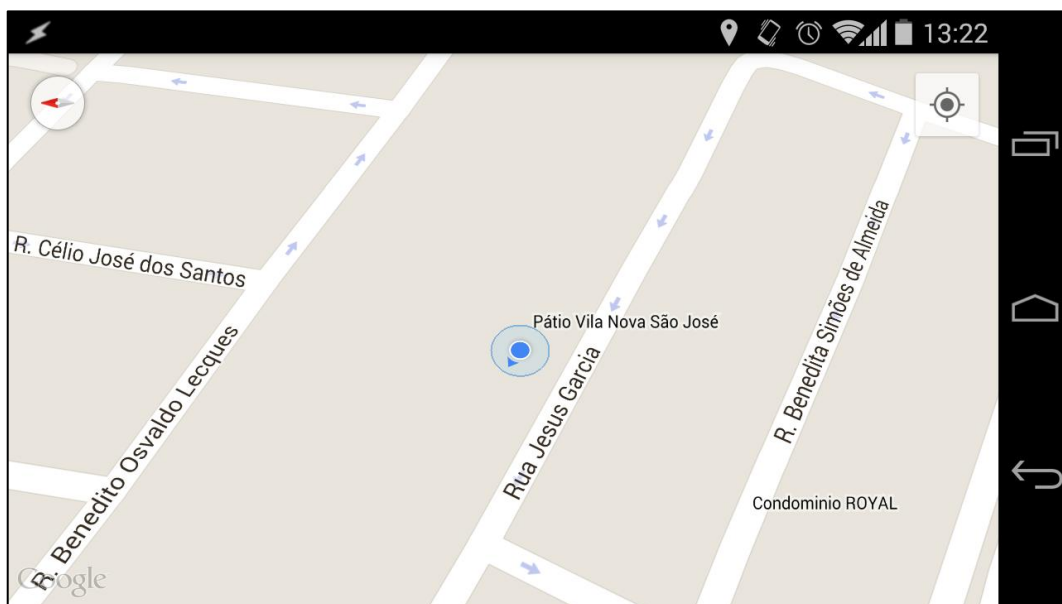
4- RESULTADOS

Este Capítulo descreve os resultados obtidos e está dividido nas seguintes seções: a Seção 4.1 explica sobre a utilização do sensor de localização externa, a Seção 4.2 explica sobre a utilização do sensor de localização interna, na Seção 4.3 explica sobre a utilização do sensor de sensação tátil, na Seção 4.4 explica sobre a utilização do sensor de visão e na Seção 4.5 explica sobre a integração desses sensores.

4.1- Utilização do Sensor de Localização Externa

Para demonstrar o uso do GPS, foi utilizado o mapa do Google, que desenha na tela exatamente o ponto onde o usuário está, através das coordenadas latitude e longitude, como demonstra a Figura 4.1.

Figura 4. 1 - Localização *Outdoor* usando Google Maps



Fonte: AUTOR (2015)

O aplicativo detecta as coordenadas enviadas através do GPS do dispositivo móvel e desenha um ponto no mapa fornecido pelo Google, indicando onde o usuário se encontra naquele momento. A configuração de atualização de localização no *smartphone* é facilmente configurável programaticamente, caso o usuário se mova, o GPS é atualizado e sua localização redesenhada na tela. Isso faz com que a localização seja mais precisa, porém, o

aplicativo consome mais energia, pois a cada momento que a localização é alterada, o aplicativo redesenha a posição na tela.

4.2- Utilização do Sensor de Localização Interna

Para demonstrar o uso do sensor de localização *Indoor* ou interna, foi desenvolvido uma tela para exibir a captura do sinal da Wi-Fi ao clicar no botão que lista todos os sinais capturados, informando o *mac address*, o SSID e o *level*, conforme a Figura 4.2.

Figura 4. 2 - Localização Indoor



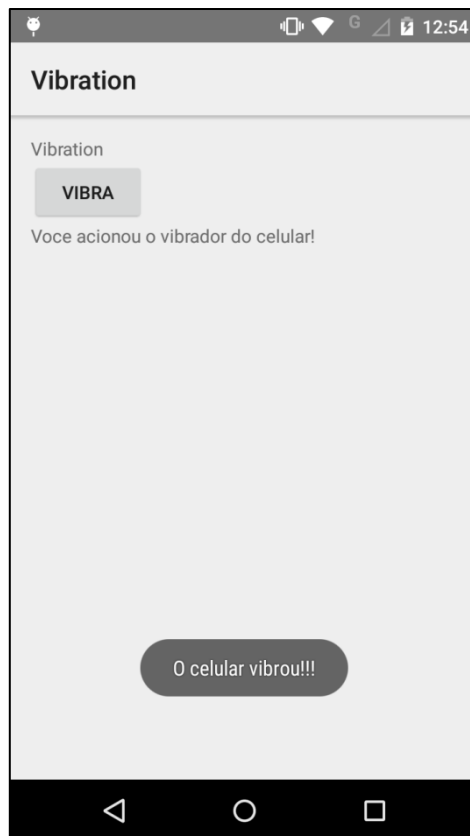
Fonte: AUTOR (2015)

O aplicativo utiliza a classe `WifiManager` para obter essas informações, facilitando o tratamento das mesmas. A listagem do aplicativo foi apenas para demonstrar de maneira simples e prática e, com a utilização desses dados para obter a localização do *smartphone* com maior precisão.

4.3- Utilização do Sensor de Sensação Tátil

Para o uso do sensor de sensação tátil, foi desenvolvida uma aplicação que ao clicar no botão Vibra, a aplicação dispara o evento de vibração ao smartphone através do método `Context.VIBRATOR_SERVICE`. Para confirmar o acionamento do botão, foi inserido no final do método uma mensagem como demonstra a Figura 4.3.

Figura 4. 3 - Implementação do sensor de sensação tátil



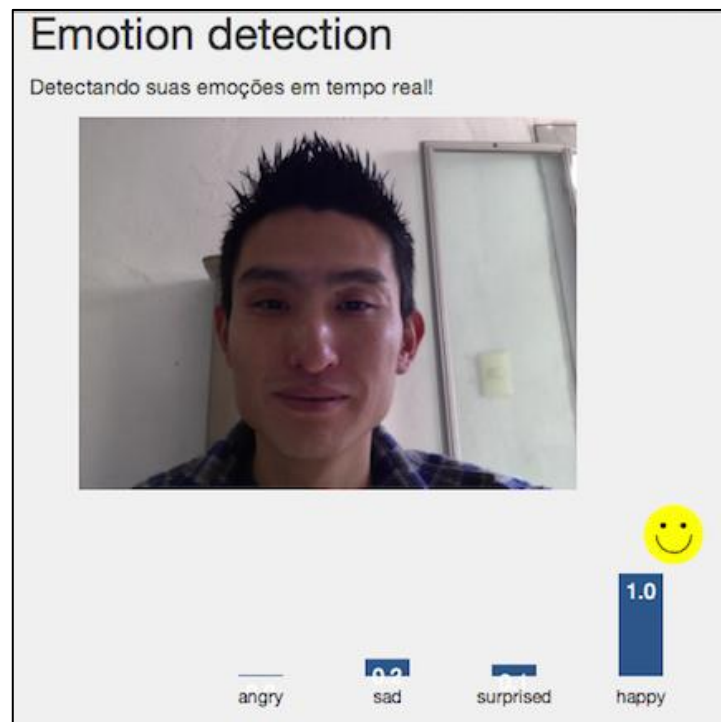
Fonte: AUTOR (2015)

Dessa forma, podemos verificar que é possível interagir com o usuário através do sensor de vibração do *smartphone* para dar um *feedback*. Muitas aplicações utilizam essa estratégia de interação com o usuário por se tratar de uma *interface* natural (WIGDOR, 2011).

4.4- Utilização do Sensor de Visão

Foi desenvolvida uma aplicação (CASSI, 2014) que utiliza o uso da câmera para medir a satisfação do usuário conforme ilustra a Figura 4.4.

Figura 4. 4 - Captando a face do usuário utilizando a câmera



Fonte: AUTOR (2015)

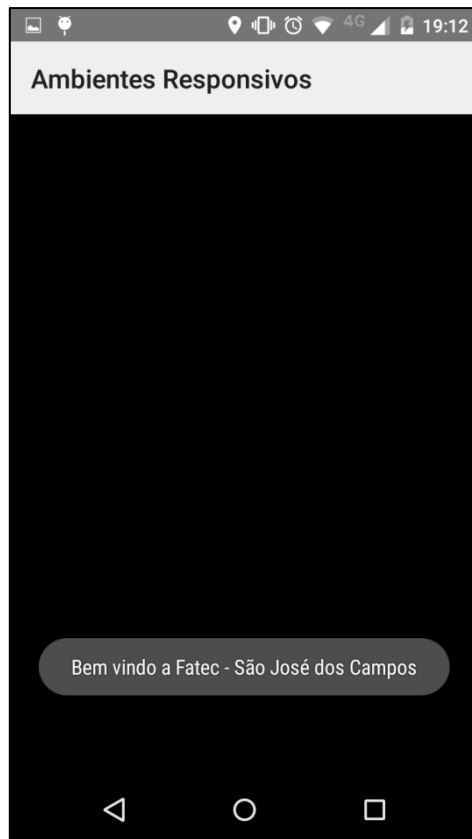
A aplicação detecta a face do usuário e mede em tempo real o que ele está sentindo, mostrando a porcentagem juntamente com a imagem do *smile*. As medidas para o reconhecimento facial são: bravo, triste, surpreso ou feliz.

4.5- União dos Sensores: Ambientes Responsivos

A integração dos sensores acima descritos foi criado um aplicativo que utiliza desses recursos para inserir as informações em seus devidos contextos, de uma forma simples, sem telas de configuração e de fácil interação.

O aplicativo vibra e mostra uma notificação ao aluno informando ao seu destino que chegou como, por exemplo, a Fatec de São José dos Campos conforme ilustrado na Figura 4.5.

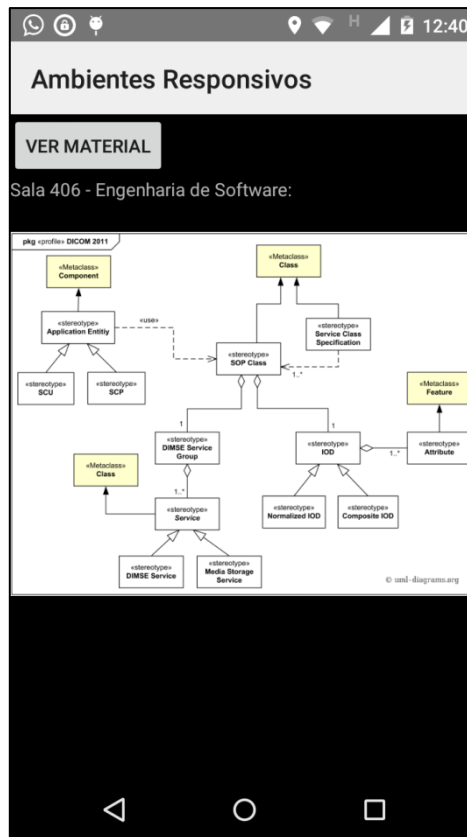
Figura 4. 5 - Indicando o local através da localização *Outdoor*



Fonte: AUTOR (2015)

O aplicativo indica ao aluno que, ao entrar na sala, por exemplo, a sala 406 – Engenharia de *Software*, o mesmo vibra mostrando na tela um *label* com o nome e o número da sala, um botão para ver o material da aula e em seguida mostra na tela a imagem referente à aula proposta naquele local, fazendo com que o aluno, ao invés de usar o *smartphone* para outros fins como, por exemplo, ficar conectado em aplicativos como Facebook, Whatsapp entre outros, receba a informação no contexto correto conforme mostra a Figura 4.6.

Figura 4. 6 - Tela mostrando o conteúdo para o aluno



Fonte: AUTOR (2015)

Ao clicar no botão Ver Material, o aplicativo redireciona para a página contendo o material específico para a aula proposta. O diferencial neste momento é que o *software* é capaz de medir a satisfação do aluno em tempo real sobre o conteúdo recebido pelo *smartphone*, fazendo com que o professor da matéria tenha um *feedback* sobre o material postado, conforme mostra a Figura 4.7.

Figura 4. 7 - Medindo a satisfação do aluno



Fonte: AUTOR (2015)

O desenvolvimento desta aplicação somente foi possível através do uso dos sensores presentes nos atuais *smartphones*, como os usados no aplicativo. O uso do GPS com o auxílio do Wi-Fi indica a posição do aluno com maior precisão, porém, devido esse recurso consumir muita bateria do aparelho foi utilizado em conjunto com a localização interna. Para evitar esse desperdício de consumo de bateria, o aplicativo ao detectar o sinal da Wi-Fi, muda o recurso de localização externa, para a localização interna economizando recursos e energia do aparelho.

O uso do sensor tátil ajuda a notificar o aluno, para que o mesmo não necessite olhar a tela o tempo todo para verificar se alguma notificação chegou a seu dispositivo. Com isso, o aplicativo informa de maneira interativa ao aluno os seus devidos contextos, utilizando recursos que qualquer *smartphone* contém, sem maiores custos.

5- CONSIDERAÇÕES FINAIS

Este Capítulo está organizado como segue: a Seção 5.1 apresenta as contribuições e conclusões, a Seção 5.1.1 apresenta a publicação de artigo e a Seção 5.2 propõe trabalhos futuros.

5.1- Contribuições e Conclusões

As contribuições deste trabalho são:

- a) Desenvolvimento de um ambiente responsivo capaz de inserir as informações em seus devidos contextos.
- b) Utilização de vários sensores disponíveis em dispositivos móveis.
- c) Capacidade de localização com alta precisão através da utilização da localização *Indoor* e *Outdoor*.
- d) Notificação através da vibração do aparelho.
- e) Notificação de imagem na tela facilitando a identificação do conteúdo da informação.
- f) Medir a satisfação em tempo real.
- g) Desenvolvimento de uma aplicação com baixo custo.
- h) Aplicar o desenvolvimento no ambiente educacional.

A partir destas contribuições pode-se concluir que:

- a) É possível capturar as informações e colocá-las em seus devidos contextos.
- b) Fazer a união de vários sensores já existentes em dispositivos móveis, facilitando o uso da informação.
- c) Alternar entre modos diferentes de localização para economizar bateria do dispositivo móvel.
- d) Interagir com o usuário através de notificações e vibrações.
- e) Analisar o conteúdo mostrado ao aluno, podendo receber um *feedback* em tempo real.
- f) Fazer com que o dispositivo tenha uma interação mais humana.
- g) Desenvolver sem necessidade de adquirir equipamentos adicionais, pois todos os sensores estão presentes nos dispositivos móveis.

- h) Inserir as informações educacionais em seus devidos contextos para ter um melhor aproveitamento em sala de aula.

5.1.1- Publicação

Foi publicado o seguinte artigo em periódico relacionado ao TG, que mostra o uso do sensor de visão em outro estudo de caso, desta vez no transporte público, demonstrando assim que este trabalho pode ser usado em diferentes contextos.

CASSI, Daniel Hideki Shibata; BERTOTI, Giuliano Araujo. APLICATIVO MÓVEL PARA O TRANSPORTE PÚBLICO QUE MEDE A SATISFAÇÃO DO USUÁRIO. Boletim Técnico da Faculdade de Tecnologia de São Paulo, v. 38, p. 99, 2014.

5.2- Trabalhos Futuros

As contribuições alcançadas com este trabalho não encerram as pesquisas relacionadas ao desenvolvimento de ambientes responsivos utilizando apenas os sensores de localização, visão, tátil entre outros sensores, mas abrem oportunidades para alguns trabalhos futuros:

- a) Utilização da câmera para detectar algum elemento como, por exemplo, a lousa e enviar informações mais detalhadas.
- b) Utilizar a realidade aumentada para melhorar o contexto e ilustrar melhor o conteúdo da informação.
- c) Utilizar um servidor *web* para coletar mais informações do aluno para melhorar a comunicação e os contextos.
- d) Inserir cadastro de *login* e senha para ter um controle melhor de segurança.
- e) Realizar cadastros de matérias em um servidor e disponibilizar para o aluno em tempo real, através de um *WebService*.
- f) Usar RFID para autenticação e identificação do aluno.
- g) Fazer a integração de ambientes educacionais com a aplicação.
- h) Utilizar outros sensores do dispositivo móvel para melhor interação.
- i) Fazer com que a aplicação bloqueie outros aplicativos de comunicação enquanto o aluno esteja acessando o aplicativo.

REFERÊNCIAS BIBLIOGRÁFICAS

ABOUT. CONTEXT, Available
<http://developer.android.com/reference/android/content/Context.html> Accessed in 03/08/2015.

ABOUT. CLMTRACKR, Available <https://github.com/auduno/clmtrackr> Accessed in 03/10/2015.

ABOUT. DOPPELLAB, Available at <http://doppellab.media.mit.edu/data/atrium.jpg> Accessed in 29/03/2015.

ABOUT. GOOGLE MAPS, Available <https://developers.google.com/maps/?hl=pt-br> Accessed in 23/05/2015.

ABOUT. INTENT, Available
<http://developer.android.com/reference/android/content/Intent.html> Accessed in 27/10/2015.

ABOUT. LOCATIONMANAGER, Available
<http://developer.android.com/reference/android/location/LocationManager.html> Accessed in 23/05/2015.

ABOUT. LOCATIONLISTENER, Available
<http://developer.android.com/reference/android/location/LocationListener.html> Accessed in 23/05/2015.

ABOUT. MEDIA PLAYER, Available
<http://developer.android.com/reference/android/media/MediaPlayer.html> Accessed in 23/05/2015.

ABOUT. MOTOROLA. Moto X, Available at <http://www.motorola.com.br/moto-X--da-Motorola/moto-x-brpt.html> Accessed in 04/10/2014.

ABOUT. MYLOCATIONOVERLAY, Available
<https://developer.mapquest.com/content/mobile/android/documentation/api/com/mapquest/android/maps/MyLocationOverlay.html> Accessed in 23/05/2015.

ABOUT. ROS (MIT). Ros implementation of hand tracking. Available at http://www.ros.org/wiki/kinect_tools Accessed in 04/10/2014.

ABOUT. SOUNDPOOL, Available at
<http://developer.android.com/reference/android/media/SoundPool.html> Accessed in 23/05/2015.

ABOUT. VIBRATOR, Available at <http://developer.android.com/reference/android/os/Vibrator.html> Accessed in 23/05/2015.

ABOUT. VITAMINS, Available at <http://www.creativeapplications.net/objects/lego-calendar-by-vitamins-design-syncs-with-google-calendar/> Accessed in 23/03/2015.

ABOUT. WIFIMANAGER, Available at <http://developer.android.com/reference/android/net/wifi/WifiManager.html> Accessed in 23/03/2015.

ABOUT. TOUCH DEVICES, Available at <https://source.android.com/devices/input/touch-devices.html> Accessed in 23/03/2015.

ADAMS, Norman et al. An infrared network for mobile computers. In: Proceedings USENIX Symposium on Mobile & Location-independent Computing. 1993.

BARFIELD, W.; WEGHORST, S.; “The Sense of Presence Within Virtual Environments: A Conceptual Framework, in Human-Computer Interaction: Software and Hardware Interfaces“, Vol B, edited by G. Salvendy and M. Smith, Elsevier Publisher, 699-704, 1993.

BARGAR, R.; CHOI, I.; DAS S.; “Model based interactive sound for an immersive virtual environment,” in Proceedings of the International Computer Music Conference (ICMC '94), 1994.

BARNES, J. A.; “Social networks. “ Addison-Wesley Module in Anthropology, v.26, p.1-29, 1972.

BARROS, Denise Alves de. **Utilização da Internet das Coisas no Desenvolvimento de Interfaces Naturais**. 2013.49f. Trabalho de Graduação - FATEC de Sao Jose dos Campos: Professor Jessen Vidal.

BOUHNİK, D.; DESHEN, M.; “WhatsApp Goes to School: Mobile Instant Messaging between Teachers and Students“, Journal of Information Technology Education: Research, vol.13, 2014.

CASSI, Daniel Hideki Shibata ; BERTOTI, Giuliano Araujo . APLICATIVO MÓVEL PARA O TRANSPORTE PÚBLICO QUE MEDE A SATISFAÇÃO DO USUÁRIO. Boletim Técnico da Faculdade de Tecnologia de São Paulo, v. 38, p. 99, 2014.

- CASTRO, Paul et al. A probabilistic room location service for wireless networked environments. In: Ubicomp 2001: Ubiquitous Computing. Springer Berlin Heidelberg, 2001. p. 18-34.
- COLEMAN, B.; “Using Sensor Inputs to Affect Virtual and Real Environments,” IEEE Pervasive Computing, vol. 8, no. 3, pp. 16–23, 2009.
- COWIE, R., Douglas-Cowie, E., Tsapatsoulis, N., Votsis, G., Kollias, S., Fellenz, W., & Taylor, J. G. (2001). Emotion recognition in human-computer interaction. *Signal Processing Magazine, IEEE*, 18(1), 32-80.
- DE CARVALHO, Marcele Regine; FREIRE, Rafael C.; NARDI, Antonio Egidio. Realidade virtual no tratamento do transtorno de pânico. *J Bras Psiquiatr*, v. 57, n. 1, p. 64-69, 2008.
- DUBLON, Gershon et al. “Doppellab: Tools for exploring and harnessing multimodal sensor network data. *Sensors*, p. 1612-1615, 2011.
- EVENNOU, Frédéric; MARX, François. Advanced integration of WiFi and inertial navigation systems for indoor mobile positioning. *Eurasip journal on applied signal processing*, v. 2006, p. 164-164, 2006.
- GAMMA, Erich et al. *Design patterns: elements of reusable object-oriented software*. Pearson Education, 1994.
- GILES, Jim. Inside the race to hack the Kinect. *New Scientist*, v. 208, n. 2789, p. 22-23, 2010.
- GRINTER, R.; ELDRIDGE, M.; “Wan2tlk?: everyday text messaging. In *Proceedings of CHI '03* “, ACM (2003), 441–448, 2003.
- GROSS, J. J. (1999). Emotion and emotion regulation. *Handbook of personality: Theory and research*, 2, 525-552.
- HENRY, Paul; LUO, Hui. WiFi: what's next?. *Communications Magazine, IEEE*, v. 40, n. 12, p. 66-72, 2002.
- HERMANN, Thomas. *Sonification for exploratory data analysis*. 2002.
- HERMANN, Thomas; HUNT, Andy. An introduction to interactive sonification. *IEEE multimedia*, p. 20-24, 2005.

- HERMANN, Thomas; KRAUSE, Jan; RITTER, Helge. Real-time control of sonification models with a haptic interface. In: Proceedings of the international conference on auditory display (ICAD). 2002. p. 82-86.
- HINTON, Andrew. Understanding Context: Environment, Language, and Information Architecture. O'Reilly Media, 2014.
- HOLLAN, James, Edwin Hutchins, and David Kirsh. "Distributed cognition: toward a new foundation for human-computer interaction research." *ACM Transactions on Computer-Human Interaction (TOCHI)* 7.2 (2000): 174-196.
- KAMERMAN, Ad; MONTEBAN, Leo. WaveLAN®-II: a high-performance wireless LAN for the unlicensed band. *Bell Labs technical journal*, v. 2, n. 3, p. 118-133, 1997.
- KAREL, A., De Vries, G., & De Waard, D.; "The effects of mobile telephoning on driving performance. *Accident Analysis and Prevention*", 23, 309–316, 1991.
- KAREN, C.; OLIVEIRA, R.; "What's up with WhatsApp? Comparing Mobile Instant Messaging Behaviors with Traditional SMS", *Mobile HCI 2013 – Collaboration and Communication*, 2013.
- KRAMER, Gregory. Auditory display: Sonification, audification, and auditory interfaces. Perseus Publishing, 1993.
- KRUEGER, Myron W. Responsive environments. In: Proceedings of the June 13-16, 1977, national computer conference. ACM, 1977. p. 423-433.
- KRUEGER, Wolfgang; FROEHLICH, Bernd. Responsive Workbench. In: *Virtual Reality'94*. Springer Berlin Heidelberg, 1994. p. 73-80.
- KUUTTI, Kari. "Activity theory as a potential framework for human-computer interaction research." *Context and consciousness: Activity theory and human-computer interaction* (1996): 17-44.
- LINO, Jorge Alves; SALEM, Benjamin; RAUTERBERG, Matthias. Responsive environments: User experiences for ambient intelligence. *Journal of ambient intelligence and smart environments*, v. 2, n. 4, p. 347-367, 2010.
- MACKENZIE, I. Scott. "Fitts' law as a research and design tool in human-computer interaction." *Human-computer interaction* 7.1 (1992): 91-139.

- NI, Lionel M. et al. LANDMARC: indoor location sensing using active RFID. *Wireless networks*, v. 10, n. 6, p. 701-710, 2004.
- PAVLOVIC, Vladimir I., Rajeev Sharma, and Thomas S. Huang. "Visual interpretation of hand gestures for human-computer interaction: A review." *Pattern Analysis and Machine Intelligence*, IEEE Transactions on 19.7 (1997): 677-695.
- PICARD, Rosalind Wright. "Affective computing." (1995).
- SCHMIDT, Heiko A. et al. TREE-PUZZLE: maximum likelihood phylogenetic analysis using quartets and parallel computing. *Bioinformatics*, v. 18, n. 3, p. 502-504, 2002.
- SCOTT, Anthony. Eliciting GPs' preferences for pecuniary and non-pecuniary job characteristics. *Journal of health economics*, v. 20, n. 3, p. 329-347, 2001.
- SHOTTON, J., Sharp, T., Kipman, A., Fitzgibbon, A., Finocchio, M., Blake, A., ... & Moore, R. (2013). Real-time human pose recognition in parts from single depth images. *Communications of the ACM*, 56(1), 116-124.
- SLATER, M.; WILBUR, S.; "A Framework for Immersive Virtual Environments (FIVE): Speculations on the Role of Presence in Virtual Environments", *Presence: Teleoperators and Virtual Environments*, MIT Press, 6(6), 603-616, 1997.
- SONZA, Andréa Polletto; SANTAROSA, Lucila Maria Costi. Ambientes digitais virtuais: acessibilidade aos deficientes visuais. *RENTE*, v. 1, n. 1, 2003.
- STANLEY-MARBELL, P.; "A mobile client platform for sensor network," Eindhoven University of Technology Department of Electrical Engineering Electronic Systems, January-2008.
- TORI, Romero; KIRNER, Claudio; SISCOOTTO, Robson Augusto. *Fundamentos e tecnologia de realidade virtual e aumentada*. Editora SBC, 2006.
- VALLÉE, M., Ramparany, F., & Vercouter, L. (2005). Flexible composition of smart device services. *PSC*, 5, 165-171.
- VALLI, Alessandro. Natural interaction white paper. Recuperado de <http://www.naturalinteraction.org/images/whitepaper.pdf>, 2007.
- VIZZARI, G. (2004). Dynamic interaction spaces and situated multi-agent systems: from a multi-layered model to a distributed architecture. Ph. D. Thesis.

WIGDOR, Daniel; WIXON, Dennis. Brave NUI world: designing natural user interfaces for touch and gesture. Elsevier, 2011.

WU, C., & Aghajan, H. (2009, November). Using context with statistical relational models: object recognition from observing user activity in home environment. In Proceedings of the Workshop on Use of Context in Vision Processing (p. 5). ACM.

YOUSSEF, Moustafa A.; AGRAWALA, Ashok; UDAYA SHANKAR, A. WLAN location determination via clustering and probability distributions. In: Pervasive Computing and Communications, 2003.(PerCom 2003). Proceedings of the First IEEE International Conference on. IEEE, 2003. p. 143-150.