Universidade Federal de Campina Grande

Centro de Engenharia Elétrica e Informática

Coordenação de Pós-Graduação em Ciência da Computação

Infraestrutura para o Desenvolvimento de Aplicações Pervasivas com Suporte ao Chaveamento Automático de Tecnologia de Comunicação

Arthur Lucio Meneses Farias

Dissertação submetida à Coordenação do Curso de Pós-Graduação em Ciência da Computação da Universidade Federal de Campina Grande - Campus I como parte dos requisitos necessários para obtenção do grau de Mestre em Ciência da Computação.

Área de Concentração: Ciência da Computação

Linha de Pesquisa: Engenharia de Software

Hyggo Almeida (Orientador)

Angelo Perkusich (Orientador)

Campina Grande, Paraíba, Brasil

FICHA CATALOGRÁFICA ELABORADA PELA BIBLIOTECA CENTRAL DA UFCG

F224i Farias, Arthur Lucio Meneses.

Infraestrutura para o desenvolvimento de aplicações pervasivas com suporte ao chaveamento automático de tecnologia de comunicação / Arthur Lucio Meneses Farias. – Campina Grande, 2012.

65 f. : il. color.

Dissertação (Mestrado em Ciência da Computação) – Universidade Federal de Campina Grande, Centro de Engenharia Elétrica e Informática.

Orientadores: Prof. Dr. Hyggo Oliveira de Almeida, Prof. Dr. Angelo Perkusich.

Referências.

1. Aplicação Pervasiva. 2. Chaveamento entre Tecnologias. I. Título.

CDU 004.4'2(043)

"INFRAESTRUTURA PARA O DESENVOLVIMENTO DE APLICAÇÕES PERVASIVAS CIENTES DE CONTEXTO COM SUPORTE AO CHAVEAMENTO AUTOMÁTICO DE TECNOLOGIA DE COMUNICAÇÃO"

ARTHUR LUCIO MENESES FARIAS

DISSERTAÇÃO APROVADA EM 31/08/2012

HYGGO OLIVEIRA DE ALMEIDA, D.Sc
Orientador(a)

ANGELO PERKUSICH, D.Sc
Orientador(a)

MARCOS RICARDO ALCÂNTARA MORAIS, D.Sc
Examinador(a)

LEANDRO DIAS DA SILVA, D.Sc
Examinador(a)

CAMPINA GRANDE - PB

Resumo

Conforme previsto por Weiser há aproximadamente vinte anos, a era da computação ubíqua na qual a tecnologia se integra a vida das pessoas de maneira tão presente que as mesmas nem percebem tem se tornado realidade. Este novo paradigma tem se tornado possível graças à popularização de dispositivos tais como smartphones e tablets que, aliados ao extraordinário avanço das tecnologias de comunicação sem fio de curto e longo alcance, têm viabilizado cenários para aplicações pervasivas. Porém, aplicações pervasivas requerem que o desenvolvedor lide com questões como: (i) extrair do ambiente em que estão inseridos informações de contexto como a presença do usuário no ambiente; (ii) lidar com a sucessiva necessidade de mudança de tecnologias de comunicação de longo e curto alcance devido à alta dinamicidade dos ambientes. Neste trabalho apresenta-se uma infraestrutura para auxiliar o desenvolvedor de aplicações pervasivas, fornecendo uma abstração para a complexidade relacionada à comunicação e ao chaveamento transparente e ciente de contexto entre as tecnologias de comunicação presentes nestes dispositivos. A validação é realizada através de um estudo de caso que utiliza as tecnologias NFC, Bluetooth e Wi-Fi, com gerenciamento da utilização destas tecnologias de acordo com a demanda da aplicação.

Abstract

As predicted by Weiser about twenty years ago, the ubiquitous computing era in which technology integrates people's lives in such way that they do not even realize it has become reality. This new paradigm has become possible thanks to the popularization of devices such as smartphones and tablets that combined with the extraordinary progress of wireless technologies for short and long range, have made possible scenarios for pervasive applications. However, pervasive applications require the developer to deal with issues such as: (i) extract from the environment they are inserted context information as user's presence, (ii) handle the need for successive changing communication technologies for short and long range due to high dynamicity of the environments. This work presents an infrastructure for supporting pervasive applications developers, providing an abstraction for the complexity related to the communication and to the context-aware switching between communication technologies present in these devices. The validation is performed through a case study that utilizes NFC, Bluetooth and Wi-Fi technologies, with management of these technologies according to application needs.

Agradecimentos

Agradeço primeiramente a Deus, o autor da vida, que pela sua graça, me permitiu superar mais um desafio.

A minha esposa, Fernanda, companheira de todas as horas.

A minha família, pai, mãe e irmãs pelo apoio incondicional.

Aos meus amigos campinenses pela companhia nos dias em Campina Grande.

Também, aos amigos e companheiros de laboratório, com os quais eu pude trocar idéias e soluções que contribuíram diretamente no desenvolvimento deste trabalho.

Aos professores e funcionárias da COPIN sempre eficientes.

Aos meus orientadores Hyggo e Angelo, pela contribuição substancial para a realização destre trabalho e orientação recebida. Ao CNPq, pelo apoio financeiro.

Conteúdo

Lista de Símbolos

3G - *3rd Generation*

API - *Application Programming Interface*

CEO - *Chief Executive Officer*

PC - *Personal Computer*

GHz - *Gigahertz*

GPS - *Global Positioning System*

IEEE - *Institute of Electrical and Electronics Engineers*

IP - *Internet Protocol*

ISM - *Industrial, Scientific and Medical*

JRE - *Java Runtime Environment*

JSR - *Java Specification Request*

KBIT - *Kilobit per second*

KBPS - *Kilobyte per second*

LAN - *Local Area Network*

NFC - *Near Field Communication*

MAC - *Media Access Control*

MBPS - *Megabyte per second*

MHz - *Megahertz*

PPT - *Microsoft Power Pointer Presentation*

PDF - *Portable Document Format*

RFCOMM - *Radio-Frequency Communication*

RFID - *Radio-Frequency IDentification*

SSID - *Service Set Identification*

UUID - *Universally Unique Identifier*

UPnP - *Universal Plug and Play* WEP - *Wired Equivalent Privacy*

WLAN - *Wireless Local Area Network*

WPA - *Wi-Fi Protected Access*

WPA2 - *Wi-Fi Protected Access II*

Lista de Figuras

Lista de Tabelas

Lista de Códigos Fonte

Capítulo 1

Introdução

"The most profound technologies are those that disappear." (Mark Weiser, 1991)

Durante a conferência "All Things Digital", em 01 de Junho de 2010, o então CEO*(Chief Executive Officer)* da Apple Inc., Steve Jobs declarou que nós havíamos alcançado a era "pós-PC"[1], a era da computação ubíqua na qual dispositivos tecnológicos se integram a vida das pessoas de maneira tão presente e profunda que as mesmas nem percebem. *Smartphones, tablets* e pequenos dispositivos portáteis são exemplos de dispositivos "pós-PC".

O termo PC (Computador Pessoal, do inglês) se refere a computadores pessoais derivados do modelo da IBM, o IBM PC, que se popularizou no início da década de 90. *Desktops* e *laptops* são os exemplos destes computadores. A Figura 1.1 ilustra como estes computadores têm evoluído e como aparecerão no decorrer dos próximos anos, de acordo com o Forrester Researcher[2].

A competitividade entre os fabricantes e entre as operadoras de telecomunicação tem barateado e popularizado estes dispositivos. Esta popularização pode ser percebida através da Figura 1.2 que mostra um comparativo entre vendas de computadores e dispositivos móveis nas últimas três décadas. Através da figura, é possível visualizar como as vendas de *tablets* e, sobretudo, de *smartphones* - telefone celular com funcionalidades avançadas que podem ser estendidas por meio de programas executados no seu sistema operacional - alcançaram em poucos anos o nível que o PC levou décadas para alcançar.[3] Ademais, as vendas de

[1] http://www.techrepublic.com/blog/hiner/steve-jobs-proclaims-the-post-pc-era-has-arrived/4701

[2] http://www.forrester.com/

[3] http://www.asymco.com/2012/01/17/the-rise-and-fall-of-personal-computing

smartphones no ano de 2011 superou a venda de PC's, e a estimativa é que em poucos anos fará diminuir as vendas de PC's, tomando-lhes seu espaço.

A popularização destes dispositivos aliada ao extraordinário avanço das tecnologias de comunicação sem fio de curto e longo alcance tais como NFC, Bluetooth, Wi-Fi e 3G, levou a criação de uma nova indústria de aplicativos que fatura 10 bilhões de dólares, segundo a consultoria Business Insider.[4] A massificação destes aplicativos é de tal maneira que os dois principais sites de comércio de aplicativos, o *App Store* e o *Google Play* juntos já fornecem, para download, mais de 1 milhão de aplicativos voltados para *smartphones* e *tablets*.[5]

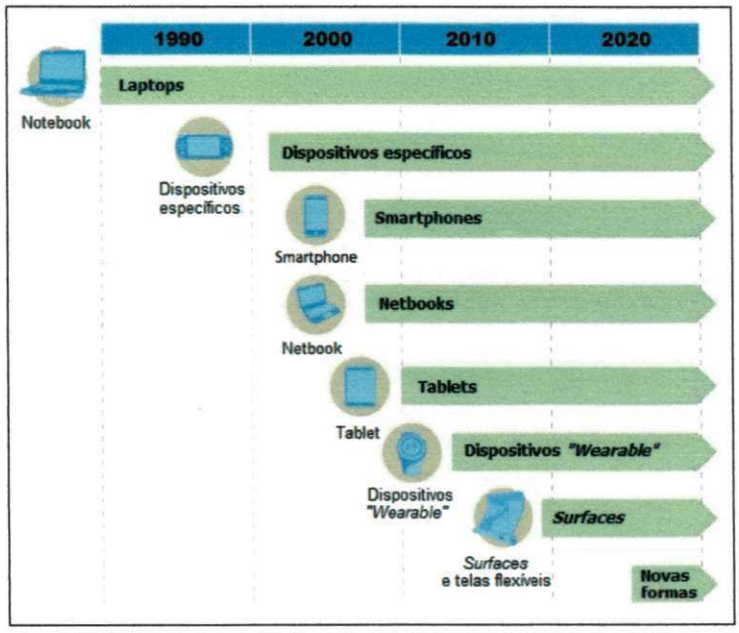


Figura 1.1: Dispositivos da "era pós-PC" de acordo com o Forrester Researcher, Inc.

Neste contexto de evolução tecnológica, a NFC[6] (Comunicação por proximidade de campo) surge como um dos mais recentes avanços nas tecnologias de comunicação. NFC oferece uma comunicação sem fio de curto alcance entre dispositivos eletrônicos não apenas segura mas também simples e intuitiva. Os usuários de dispositivos que suportam o NFC podem simplesmente aproximar ou tocar seus dispositivos em outros elementos NFC no ambiente para se comunicar com eles tornando o uso de aplicações e troca de dados fácil e conveniente.

[4]http://www.businessinsider.com/the-future-of-mobile-deck-2012-3

[5]http://en.wikipedia.org/wiki/Smartphone

[6]http://www.nfc-forum.org

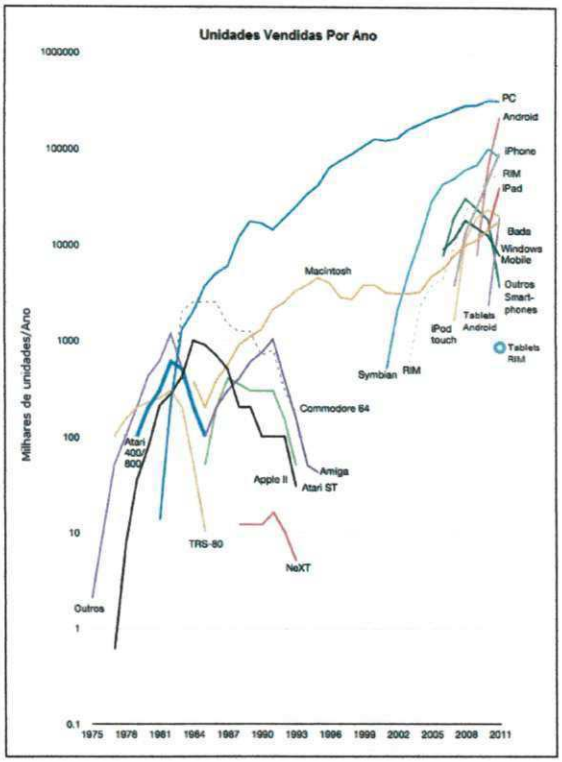


Figura 1.2: Unidades de computadores e dispositivos vendidos por ano

O paradigma da computação pervasiva, descrito pela primeira vez por Mark Weiser em 1991, remete a uma realidade onde dispositivos inteligentes e conectados em rede executam aplicações que tomam decisões e que se integram de forma transparente à vida das pessoas, auxiliando na execução de suas tarefas [24] [36]. Na prática, uma aproximação razoável a este ideal seria a existência de um sistema com o mínimo de interação com o usuário, ou seja, um sistema que requeresse o mínimo de intervenção humana possível [37].

Embora este paradigma ainda não esteja plenamente realizado, a combinação entre NFC e a capacidade de processamento dos dispositivos "pós-PC" pode enfim tornar a idéia de Weiser completamente possível permitindo que seja possível ter cenários intuitivos para aplicações pervasivas [35].

O paradigma da computação pervasiva tem algumas necessidades especiais. A primeira delas é que para que ele funcione perfeitamente é primordial que as aplicações tenham a capacidade de extrair do ambiente em que estão inseridas informações de contexto tais como temperatura e localização. Aplicações que fazem uso de informações de contexto são denominadas *aplicações sensíveis ao contexto* ou *aplicações cientes do contexto* [27] [28].

Ademais, os ambientes da computação pervasiva são, por natureza, muito heterogêneos devido aos diversos dispositivos com diferenças de hardware e software. Além disso, estes ambientes são altamente dinâmicos devido à constante entrada e saída de usuários do ambiente. Assim, faz-se necessária uma infraestrutura de hardware e software que abstraia toda a complexidade de baixo nível tais como a comunicação entre dispositivos com as mais diversas tecnologias de comunicação - a saber, de curto e longo alcance - e critérios de segurança como controle de acesso de usuários [9]. Assim, os *frameworks* - aplicação genérica que fornece uma abstração para que várias aplicações específicas possam utilizar - surgem como soluções que visam suportar o desenvolvimento de software na medida em que eles oferecem um conjunto de ferramentas previamente concebidas e testadas [26].

1.1 Problemática

O principal benefício que a inclusão do NFC trouxe ao *smartphone* foi a simplicidade e o esforço mínimo exigido para aplicações interagirem com o ambiente. Assim, o NFC e as outras tecnologias de comunicação de maior alcance como Wi-Fi e Bluetooth fazem do *smartphone* um forte candidato a interface de interação padrão para aplicações de computação pervasiva [21].

Por isso, inúmeras possibilidades de criação de aplicações se tornaram possíveis, e muitas destas têm sido implementadas. São exemplos: aplicações para controle de acesso [20], saúde [13] [32] [10], transporte [29], trocas de dados [31] e, principalmente, para pagamentos móveis [19] [6].

Entretanto, ainda não existe uma solução para *smartphones* que forneça suporte às tecnologias de comunicação de curto e longo alcance atuando de maneira integrada e ciente de contexto, e que tenha a finalidade de acelerar o desenvolvimento de aplicações pervasivas. O cenário a seguir descreve uma aplicação que necessita desta solução.

José é um professor e palestrante e vai dar uma palestra em um simpósio da sua área. Ele carrega consigo seu smartphone equipado com as tecnologias NFC, Wi-Fi e Bluetooth. Ao chegar, pela primeira vez, no palco da palestra, José ao aproximar seu smartphone de um smartpoint, inicia-se uma que troca dados entre ambos, uma autenticação do usuário José e o Smartphone recebe toda as informações de configuração de Wi-Fi e Bluetooth disponíveis

no ambiente do simpósio. Assim, o aplicativo de apresentação de slides é iniciado e o arquivo é automaticamente passado para projetor que prontamente inicia a apresentação. A escolha da tecnologia de comunicação utilizada é feita automaticamente pelo smartphone e é determinada por fatores como tipo de aplicação, preferência do usuário e nível de bateria do aparelho. Tudo isso acontece de maneira automática, transparente para o José e em poucos segundos.

José inicia sua palestra utilizando o próprio smartphone como controle da apresentação, de maneira que ele manipula os slides para frente e para trás com apenas o aparelho. Se durante a apresentação o aparelho mostrar-se com pouca carga de bateria, a conexão é trocada de Wi-Fi para Bluetooth. Em caso de o palestrante decidir se afastar do palco e, por isso, perder o sinal do Bluetooth, a conexão é chaveada de volta para o Wi-Fi.

A apresentação termina, e durante o coffee break José encontra João, o próximo palestrante do dia. José aproxima seu smartphone do smartphone do João e todos os dados de configuração obtidos no palco, são passados para o aparelho de João. Assim, João já pode iniciar o aplicativo de apresentação de slides assim que tiver sinal de Bluetooth ou Wi-Fi.

Para acelerar o desenvolvimento de aplicações como a descrita neste cenário, faz-se necessária uma infraestrutura que auxilie o desenvolvedor de aplicações pervasivas em smartphones fornecendo uma abstração para a complexidade relacionada à comunicação e chaveamento entre as tecnologias de comunicação presentes nestes dispositivos como NFC, Bluetooth e Wi-Fi, e que forneça uma transição transparente e ciente de contexto entre as mesmas.

Mais especificamente, é preciso uma abordagem única que forneça a seguintes funcionalidades: (i) tecnologias de comunicação de curto alcance; (ii) tecnologias de comunicação de longo alcance; e (iii) chaveamento entre tecnologias de comunicação ciente de contexto.

1.2 Objetivos

O objetivo deste trabalho é desenvolver uma infraestrutura que auxilie desenvolvedores de *smartphones* a criarem softwares pervasivos que utilizem diversas tecnologias de comunicação sem fio de curto e de longo alcance. Esta infraestrutura visa fornecer uma abstração do mecanismo de comunicação e de chaveamento entre tecnologias de maneira transparente

para o desenvolvedor de aplicações pervasivas.

Portanto, propõe-se um *framework* que seja capaz de suportar, especialmente: (i) comunicação de curto alcance; (ii) comunicação de longo alcance através do Bluetooth; (iii) comunicação de longo alcance através do Wi-Fi e; (iv) chaveamento automático entre estas as tecnologias de ciente de contexto.

A plataforma de *smartphone* utilizada é o android 2.3, através do aparelho Samsung/Google Nexus S, que tem a capacidade de reunir as três tecnologias de comunicação em um único aparelho.

Para o processo de validação e estudo de caso da infraestrutura foi desenvolvida uma aplicação que utiliza os recursos que a infraestrutura fornece.

1.3 Relevância

Atualmente, os desenvolvedores de aplicações pervasivas para *smartphones* têm de lidar diretamente com as diferenças relacionadas às diferentes formas de conectividade do aparelho, e principalmente, com o processo de chaveamento entre tecnologias. Portanto, existe uma enorme perda de tempo devido à necessidade de que para cada nova aplicação que precise ser desenvolvida seja preciso "reinventar a roda" em relação aos detalhes relacionados ao trato com as tecnologias de comunicação de curto e longo alcance, e sobretudo, ao chaveamento entre estas tecnologias.

Desta forma, o desenvolvimento deste trabalho possui como principal contribuição a criação de uma infraestrutura que torna transparente para o desenvolvedor de aplicações pervasivas o processo de comunicação entre diferentes tecnologias de comunicação de longo e curto alcance, bem como o chaveamento entre elas. Assim, esta solução proporciona uma abstração para ser utilizada durante a criação de aplicações pervasivas em *smartphones*. É importante também fornecer esta solução em uma base tecnológica que possa ser amplamente utilizada pelos programadores do mundo inteiro.

Além disso, com a criação da infraestrutura, pode-se vislumbrar novas pesquisas em diversas área de domínio da computação, tais como em aplicações para: monitoramento de saúde; mínimo de interação em ambientes inteligentes; publicidade móvel; entre outros.

1.4 Estrutura da Dissertação

O restante deste trabalho está organizado da seguinte forma:

- No *Capítulo 2* é apresentada uma fundamentação teórica sobre o paradigma da Computação Pervasiva, sublinhando suas bases, fundamentos e os requisitos necessários para sua realização plena. Em seguida, é apresentada uma fundamentação sobre as tecnologias de comunicação sem fio de curto e longo alcance. Por fim, detalha-se o conceito e os princípios fundamentais do chaveamento entre as tecnologias.
- No *Capítulo 3* são apresentados os trabalhos relacionados;
- No *Capítulo 4* é apresentado o projeto da infraestrutura desenvolvida, descrevendo detalhadamente os componentes de sua arquitetura, bem como, sua implementação. Destacando, principalmente, o processo e o protocolo adotado para o chaveamento entre as tecnologias de comunicação
- No *Capítulo 5* é apresentado o estudo de caso desenvolvido a partir da infraestrutura;
- Por fim, no *Capítulo 6* são apresentadas as conclusões do trabalho, suas contribuições e discussões sobre trabalhos futuros.

Capítulo 2

Fundamentação Teórica

Neste capítulo são apresentadas as bases teóricas que norteiam os principais domínios envolvidos neste trabalho. Os conceitos apresentados neste capítulo são considerados fundamentais para a compreensão do restante do trabalho. Inicialmente, será apresentada uma breve descrição sobre o paradigma da Computação Pervasiva, destacando seus princípios e as necessidades para a sua viabilização. Em seguida, serão detalhadas as tecnologias de comunicação que podem ser aplicadas ao contexto da Computação Pervasiva. Ao final, é apresentado o conceito e detalhes relativos ao chaveamento automático de Tecnologia de Comunicação.

2.1 Computação Pervasiva

Em 1991, Mark Weiser descreveu em seu trabalho um novo paradigma chamado Computação Pervasiva [27]. Weiser vislumbrou uma realidade na qual a computação e as aplicações estariam embutidas nos objetos do dia-a-dia, de forma integrada ao ambiente. Estes objetos inteligentes seriam capazes de se comunicar entre si e interagir com o ambiente sem necessitar da atenção dos usuários. Assim, o paradigma poderia ser entendido como um conjunto recursos computacionais atuando de forma integrada, sem conhecimento dos usuários, como o objetivo de auxiliar na execução de suas tarefas. Este auxílio é exercido através de ações no ambiente e disponibilização de informações relevantes, no local adequado e no momento adequado de acordo com as preferências e necessidades do usuário [15].

Para que o paradigma da Computação Pervasiva pudesse se tornar possível, três elemen-

tos chaves deveriam estar largamente disponíveis: dispositivos de baixo custo e com baixo consumo de energia, infraestrutura de rede sem fio e sistemas que implementassem aplicações pervasivas [15]. Entretanto, no momento da história em que Weiser proclamou os fundamentos para o paradigma da Computação Pervasiva, estes três elementos ainda não se encontravam em um estágio de desenvolvimento avançado, nem tampouco havia a massificação e popularização necessárias para tornar possível as primeiras aplicações pervasivas.

No entanto, nos últimos anos este cenário sofreu uma mudança substancial com a popularização do acesso à Internet, aliado ao extraordinário avanço das tecnologias de comunicação sem fio (e.g., *Bluetooth*, Wi-Fi, NFC, 3G, Wi-Max, entre outros) e popularização dos dispositivos móveis no mercado consumidor, sobretudo, *notebooks*, *smartphones* e *tablets*. Os dois últimos, de maneira especial, estão cada vez populares, com um considerável poder de processamento e com um acesso constante à Internet.

Este novo cenário tem se apresentado favorável a viabilização do paradigma da Computação Pervasiva e para o desenvolvimento de aplicações que tenham a capacidade de serem executadas em ambientes pervasivos.

2.1.1 Ambientes Pervasivos

Segundo Satyanarayanan, os Ambientes Pervasivos podem ser definidos como ambientes saturados com computação e comunicação [28]. As principais particularidades destes ambientes são a heterogeneidade e a dinamicidade.

A primeira particularidade se refere a multiplicidade e diversidade de: (i) dispositivos que se distinguem no poder de computação e na presença do mesmo no ambiente (e.g., *smartphones* e *tablets*, computadores pessoais, celulares, entre outros); (ii) tecnologias de comunicação sem fio (e.g., Bluetooth, Wi-Fi, NFC, 3G, entre outros); e (iii) Sistemas Operacionais (e.g, android, iOS, Windows, entre outros).

A segunda particularidade está relacionada com a: (i) capacidade de mobilidade destes dispositivos permitindo que estes possam sair/entrar do ambiente a qualquer momento; e (ii) capacidade do dispositivo de passar a fornecer ou deixar de fornecer serviços.

A sensibilidade a presença do usuário e sua mobilidade são características importantes dos ambientes pervasivos. Elas permitem que os serviços pervasivos sejam oferecidos no ambiente enquanto o usuário esteja presente no mesmo.

O conceito de dispositivos incorporados transparentemente ao ambiente dos seres humanos junto com a necessidade de lidar com as características de ambientes pervasivos, exige que as aplicações se adaptem em tempo de execução as alterações no ambiente e as necessidades dos usuários. Diante disso, a ciência de contexto surge como solução para a necessidade adaptabilidade necessária em ambientes pervasivos.

2.1.2 Ciência de Contexto

A ciência de contexto pode ser definida conforme foi proposta por Dey, em [7]:

"Contexto é qualquer informação que pode ser utilizada para caracterizar a situação de uma entidade. Uma entidade é uma pessoa, lugar ou objeto que é considerado relevante para a interação entre um usuário e uma aplicação, incluindo o usuário e as aplicações."

Neste sentido, um elemento essencial da Computação Pervasiva é o uso eficiente da informação de contexto para que se atinja completamente a idéia da invisibilidade de Weiser. Para tal, é fundamental obter as informações de contexto no ambiente no qual os usuários estão inseridos a fim de descobrir quais são as reais necessidades e interesses deles.

As aplicações que utilizam informações de contexto para distinguir quais informações são úteis aos usuários, ou mesmo, quais serviços terão utilidade são chamadas de aplicações cientes de contexto [7].

Segundo [18], a obtenção de contexto pode ser classificada em três tipos: (i) sentida: através de sensores físicos dispostos no ambiente tais como sensores de proximidade, ruído, temperatura, entre outros; (ii) derivada: são derivadas da execução de um recurso computacional sob demanda tais como coordenadas geográficas extraídas de um GPS; e por fim (iii) explicitamente fornecida: quando a informação é claramente fornecida pelo usuário como por exemplo as informações selecionadas nas preferências do usuário de uma aplicação. Exemplos de contexto são: contexto do usuário, contexto do dispositivo, contexto do ambiente, entre outros [15].

Após serem adquiridas, as informações de contexto devem ser tornar disponíveis para as entidades interessadas para que sejam interpretadas, e a partir desta interpretação, seja tomada as decisões de quais ações devam ser realizadas pelas aplicações.

2.2 Tecnologias de Comunicação em Dispositivos Móveis

Nesta seção são apresentadas as informações e os principais conceitos envolvendo as tecnologias de comunicação em dispositivos móveis.

2.2.1 NFC

Soluções de comunicação em ambientes pervasivos têm se tornado cada vez mais atraentes na medida em que os dispositivos móveis começam a suportar tecnologias de curto alcance.

NFC é um dos mais recentes e promissores avanços no campo de tecnologias de comunicação sem fio. NFC é uma tecnologia de comunicação de curto alcance sem contato ou *contactless* que é baseada na tecnologia RFID. A RFID utiliza o campo de indução magnética para permitir a comunicação entre dispositivos eletrônicos. Nos útimos anos, o número soluções que utilizam a tecnologia NFC têm crescido bastante sendo elas nas mais diversas aplicabilidades. De maneira especial, a sua utilização em conjunto com os celulares tem possibilitado grandes oportunidades de aplicações [14].

Um dos principais objetivos da tecnologia NFC[1] tem sido a de tornar os benefícios da comunicação de curto alcance disponíveis aos consumidores no mundo todo[2]. A base da tecnologia de frequência de rádio existentes e já utilizada no mundo inteiro tem acelerado o desenvolvimento do NFC para que seja aplicados em soluções de negócios, tais como logística e rastreamento de itens [11]. Além disso, o advento da NFC tem trazido um avanço na mentalidade em relação ao uso da tecnologia, evoluindo na maneira em como ela é utilizada e o que ela oferece para os usuários e consumidores [1].

Apenas com uma aproximação ou com um toque, a tecnologia NFC permite um fácil uso dos dispositivos e aparelhos que são usados no dia-a-dia [3]. Abaixo são descritos exemplos de cenários onde um usuário portando um celular ou *smartphone* com suporte a NFC pode usufruir da tecnologia:

- Baixar música ou vídeo de um *Smart Poster*
- Trocar cartões de visita (*vCards*) com outro telefone ou aparelho

[1] http://www.nfc-forum.org
[2] http://www.nfc-forum.org

Figura 2.1: Aparelho dotado da tecnologia NFC sendo utilizado como um cartão de crédito

- Pagar passagem de trem ou ônibus
- Imprimir imagem em uma impressora
- Utilizar um terminal de ponto de venda para pagar uma compra, da mesma forma que com um cartão de crédito normal (*NFC Payment*). A Figura 2.1 ilustra este cenário.
- Parear dois dispositivos Bluetooth

Especificações

A NFC se comunica através de duas antenas que permitem a criação de um campo magnético utilizado para transmitir os dados. Abaixo segue uma especificação básica da tecnologia:

- Opera dentro da faixa de frequência de 13,56 MHz, com uma largura de banda de quase 2 MHz;
- Taxas de transmissão suportadas: 106, 212, ou 424 kbit/s;
- Existem três modos de operação:
  - **Modo Leitura/Escrita:** o dispositivo NFC é capaz de ler cartões ou *tags* NFC;
  - **Modo *Peer-to-Peer*:** dois dispositivos NFC podem trocar dados conforme ilustrado na Figura 2.2. Neste modo, é possível compartilhar parâmetros de configu-

Figura 2.2: Dois aparelhos dotados de NFC trocando dados em modo *Peer-to-Peer*

ração Bluetooth ou Wi-Fi, ou trocar dados, como cartões de visita ou fotografias digitais;

- **Modo Emulação de Cartões:** o próprio dispositivo NFC age como sendo um cartão ou *tag* NFC, se comportanto para um leitor como sendo um *Smart Card* tradicional.

2.2.2 Wi-Fi

Wi-Fi[3] é uma tecnologia de comunicação de dados sem fio de longo alcance compatível com infraestrutura de rede local a cabo LAN. Mudanças de velocidade, ampliação no alcance do sinal e melhorias em segurança foram algumas das principais novidades que tornaram as tecnologias Wi-Fi tão populares e essenciais. Em lugares tais como empresas, residências, campi universitário, zonas públicas como aeroportos, estações de trem, shoppings, entre outros, é fácil encontrar roteadores e pontos de acesso Wi-Fi disponíveis.

Nos últimos anos ela também tem seguido um ritmo intenso de crescimento e tem se popularizado em dispositivos móveis como *tablets* e, sobretudo, em *smartphones*.

[3] www.wi-fi.org

Especificações

A tecnologia de comunicação Wi-Fi tem suas características padronizadas pela norma IEEE 802.11 Os protocolos mais comuns do Wi-Fi são: IEEE 802.11b, IEEE 802.11g e IEEE 802.11n. Há ainda o mais novo protocolo IEEE 802.11ac. Todos estes baseados no antecessor ao IEEE 802.11[4]. Abaixo segue a descrição da família de protocolos IEEE 802.11:

- **IEEE 802.11:** Foi o primeiro protocolo desenvolvido. Oferece taxa de transmissão de dados de 2 Mbps, operando na frequência de 2.4 Ghz;
- **IEEE 802.11a:** Melhoria do IEEE 802.11 para suportar taxa de transmissão de dados de 54 Mbps operando na frequência de 5 Ghz;
- **IEEE 802.11b:** Oferece taxa de transmissão de dados de 11 Mbps, operando na frequência de 2.4 GHz;
- **IEEE 802.11g:** Permite alcançar taxa de transmissão de dados de 54 Mbps, operando na frequência de 2.4 GHz. É compatível com IEEE 802.11b;
- **IEEE 802.11n:** A mais popular atualmente e oferece uma maior taxa de transmissão de dados atingindo de 108 a 600 Mbps, operando nas frequências de 2.4 GHz e 5 GHz;
- **IEEE 802.11ac:** Mais recentemente desenvolvida e ainda não disseminada. Oferece taxa de transmissão de dados de 1.3 Gbps, operando na frequência de 5 GHz.

A tecnologia Wi-Fi vem sendo cada vez mais utilizada em uma grande variedade de aplicações devido a características como: (i) alta taxa de transmissão de dados; (ii) transmissão de alcance suficiente para ambientes fechados; (iii) compatibilidade com a infraestrutura LAN; e (iv) baixo custo relativo [12]. Na tabela 2.1 são sumarizados os protocolos Wi-Fi e suas características.

2.2.3 Bluetooth

Bluetooth é uma tecnologia de comunicação de dados sem fio de médio/longo alcance. Ela opera através de ondas de rádio e foi projetada para substituir os cabos de conexão com

[4]http://www.wi-fi.org

Protocolos	Faixa de Frequência	Largura de Banda Nominal
802.11a	5 GHz	54 Mbps
802.11b	2.4 GHz	11 Mbps
802.11g	2.4 GHz	54 Mbps
802.11n	2.4 e 5 GHz	600 Mbps
802.11ac	5 GHz	1.3 Gbps

Tabela 2.1: Resumo dos protocolos Wi-Fi

dispositivos portáteis (e.g., câmeras digitas, celulares, notebooks, fones de ouvido, entre outros) e com dispositivos fixos (e.g., teclados, *trackpads*, relógios, entre outros). Bluetooth permite uma transferência de dados confiável e livre de interferências[5].

Com a capacidade de lidar simultaneamente com transmissões de dados e de voz, a Bluetooth se tornou a tecnologia de comunicação padrão para a indústria de computadores e dispositivos periféricos.

Assim, foi adotada pelas principais indústrias e empresas do setor de telecomunicação e computação, da mesma maneira como tem sido adotada também por um conjunto de empresas atuantes em outros setores tais como o setor de entretenimento, automotivo, saúde, automação industrial e brinquedos.

Podemos resumidamente afirmar que a tecnologia Bluetooth se propõe a:

- Eliminar cabos e fios entre dispositivos fixos e moveis separados por pequenas distancias (até 10 metros);
- Facilitar comunicação de dados e voz;
- Ativar Redes ad-hoc e fornecer sincronização automática entre vários dispositivos compatíveis com a tecnologia Bluetooth.

Especificações

A tecnologia Bluetooth opera na faixa de frequência de 2.4 GHz que é chamada de banda ISM, disponível mundialmente. Opera no modo *full duplex* (recebe e transmite ao mesmo

[5]http://www.bluetooth.org

tempo) com taxas de transferência de até 3 Mbps.

As interferências são evitadas através de um processo conhecido como "salto de frequência" (*frequency hop scheme*). Neste processo, o dispositivo procura uma faixa de frequência livre através da execução 1600 saltos por segundo [12].

O consumo de energia é administrado de maneira automática na medida em que identifica a distância entre os equipamentos e muda adequadamente a potência de transmissão conforme o caso. Além disso, o dispositivo Bluetooth pode entrar em um estado baixo consumo de energia caso o tráfego de dados for muito baixo ou ocorrer uma interrupção [17].

O Bluetooth permite dois tipos de comunicação: (i) *peer-to-peer* que é feito diretamente entre dois dispositivos Bluetooth; e (ii) através de um equipamento central que se comunica com até sete outros equipamentos secundários formando uma pequena rede chamada "pico-rede".

Na Tabela 2.2 são mostradas as características do Bluetooth.

Característica	**Valores**
Frequência	2.4 GHz
Alcance	10 metros
Taxa de transferência	1 Mbps
Latência	100 ms
Segurança	Criptografia 64/128 bit
Consumo	2.5 mW

Tabela 2.2: Resumo das características do Bluetooth

Protocolos de Transporte

O protocolo Bluetooth utilizado neste trabalho é o RFCOMM (Comunicação por Rádio frequência) que é um simples conjunto de protocolos de transporte que emulam portas seriais RS-232 e podem ter até 60 conexões simultâneas para um dispositivo Bluetooth. RFCOMM fornece um simples e confiável fluxo de dados de maneira semelhante ao TCP.

Inúmeras aplicações utilizam o protocolo RFCOMM devidoo o seu suporte nas mais diversas API'S da maioria dos sistemas operacionais. Além disso, a grande vantagem de usar o RFCOMM é a possibilidade de se portar aplicações que fazem comunicação utilizando outras portas seriais para o bluetooth RFCOMM devido a ambas utilizarem portas seriais. Por este motivo, este protocolo é utilizado neste trabalho.

Bluetooth Low Energy

Bluetooth Low Energy (Bluetooth de Baixa Energia) é uma característica do Bluetooth versão 4.0 que visa a possibilitar a comunicação com novos e pequenos dispositivos sem fio de baixa potência e baixa latência. A tecnologia facilita e abre caminho para uma ampla gama de aplicações em dispositivos pequenos na área da saúde, *fitness*, segurança e na indústria de entretenimento doméstico.

O *Bluetooth Low Energy* não é considerado neste trabalho devido a ausência da versão 4.0 do bluetooth no *smartphone* e API's utilizados neste trabalho.

2.3 Chaveamento Automático de Tecnologia de Comunicação

A dinamicidade dos dispositivos móveis aliado à diversidade de tecnologias de comunicação geram a necessidade de troca ou chaveamento entre essas tecnologias na medida em que a tecnologia em uso deixa de funcionar ou deixa de atender as necessidades dos usuários e aplicações.

Neste contexto, o processo de manutenção de conexões ativas de um dispositivo móvel durante sua movimentações dentro da rede é chamado de *handover* [2]. Este processo teve origem nas telecomunicações (rede de celulares) onde é necessária a transferência da ligação corrente ou sessão de dados de um canal conectado a rede para outro canal. Este processo especificamente também é chamado de *handover* entre células [34]. Na Figura 2.3 são ilustradas situações em que chaveamento ou *handover* se faz necessário.

Há dois tipos de chaveamento ou *handover*: vertical e horizontal [22].

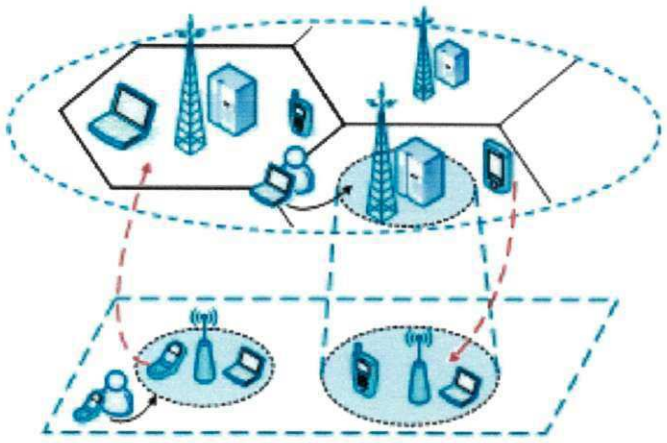


Figura 2.3: Situações que geram necessidade de *handover*

- **Vertical:** Chaveamento entre diferentes tecnologias de comunicação para suportar mobilidade dos dispositivos. Utilizado também como um *fallover* automático a fim de evitar o fim da transmissão. É o foco deste trabalho.

- **Horizontal:** Chaveamento entre canais ou fontes que operam na mesma tecnologias de comunicação. Utilizado largamente nas redes de celulares onde há um alto nível de regulamentações governamentais.

A decisão de chaveamento vertical é gerada por uma necessidade ou uma melhoria da qualidade do serviço [30].

Primeiramente, a necessidade acontece quando uma falha em uma tecnologia de comunicação interrompe o processo de comunicação e o sistema precisa se recuperar e continuar a transmissão. Assim, o sistema deve executar um chaveamento vertical para outra tecnologia de comunicação que esteja em pleno funcionamento a fim de continuar a execução sem interrupção do serviço. Este chaveamento deve acontecer de maneira automática.

A segunda forma de decisão é gerada pela estratégia de melhoria da qualidade de serviço definida por um algoritmo de decisão. A estratégia é definida de acordo com a necessidade do sistema ou do usuário e é tomada através de algoritmos que consideram diversos critérios [39].

Dentre estas estratégias podemos destacar os seguintes critérios para decisão de chaveamento:

- **Métricas:** Este critério considera métricas que estão disponíveis de acordo com o tipo de tecnologia de comunicação. Não são aplicadas a todas elas mas de maneira geral são consideradas as seguintes:
  - Força do Sinal
  - Taxa de Ruído no Sinal
  - *Bit Error Rate*
  - Distância entre as partes
- **Modelagem:** Critério definido pela implementação de uma modelagem do ambiente através de técnicas como: Redes Neurais e Lógica de Fuzzy [33] [16].

2.4 Conclusões do Capítulo

Neste capítulo foram detalhados os fundamentos básicos para o entendimento do trabalho apresentado. A princípio foi apresentada uma breve descrição sobre o paradigma da Computação Pervasiva, destacando seus princípios e as necessidades para a sua viabilização. Em seguida, foram detalhadas as tecnologias de comunicação que podem ser aplicadas ao contexto da Computação Pervasiva. E por fim, foi apresentado o conceito e detalhes relativos ao chaveamento automático de Tecnologia de Comunicação.

Capítulo 3

Trabalhos Relacionados

Neste capítulo são apresentados os principais trabalhos relacionados à infraestrutura proposta neste trabalho para o Desenvolvimento de Aplicações Pervasivas Cientes de Contexto com Suporte ao Chaveamento Automático de Tecnologia de Comunicação. Com base na revisão de literatura realizada, os trabalhos relacionados foram separados em: Sistemas Sem Fio Híbridos e NFC em Ambientes Pervasivos.

3.1 Sistemas Sem Fio Híbridos

3.1.1 CoolSpots

CoolSpots [23] é uma solução que faz o uso de tecnologias de comunicação sem fio a fim de fazer economia de energia de dispositivos móveis e aumentar o tempo de vida de suas baterias. Foi utilizado o conceito de "paginação sob demanda" cuja principal idéia é a utilização de uma interface com baixo consumo (e.g., Bluetooth) para acordar a interface de alto consumo (e.g., Wi-Fi). Esta "paginação sob demanda" é transparente para o dispositivo é disparada somente quando for necessário utilizar a tecnologia de comunicação de alto consumo e alta largura de banda para transferir dados. Nos casos em que não é necessária o Bluetooth é utilizado. Esta abordagem permite que o adaptador da tecnologia de comunicação de alto consumo fique em estado desligado ou hibernado de por longos períodos. Resultados mostraram uma economia de energia de 23% a 48% sem impactos significantes no desempenho.

A CoolSpots utiliza uma abordagem fundamentada em políticas de monitoramento de banda para fazer o chaveamento entre as tecnologias de comunicação sem fio. Esta abordagem apresentou bons resultados sob condições limitadas de uso da banda mas mostrou dificuldades em se adaptar a transferências mais pesadas.

Esta solução está relacionada com a infraestrutura desenvolvida pois se utiliza de políticas para a decisão de chaveamento entre as tecnologias de comunicação.

3.1.2 Ubiquitous Multimedia Environment

Chang-Hong Wu e Chin-Hsien Wu [38] propõem um ambiente de multimídia ubíquo que utiliza as tecnologias Zigbee (IEEE 802.15.4), NFC, Bluetooth e Wi-Fi. Os dispositivos neste ambiente transmitem dados de maneira continua e podem automaticamente chavear entre tecnologias de comunicação sem fio de acordo com as regras estipuladas que levam em consideração a largura de banda e a economia de energia. Os autores definem a política de chaveamento de uma tecnologia de comunicação sem fio para a outra baseada na medição do custo de chaveamento que tem como objetivo atingir o máximo de largura de banda possível.

Nesta solução o chaveamento entre interfaces apenas acontece quando a taxa de transmissão está decrescendo ou quando o dispositivo precisa economizar bateria.

Diferentemente dos objetivos apresentados pela infraestrutura desenvolvida, esta solução não considera as informações de contexto explicitas do usuário e da aplicação para determinar o chaveamento entre as tecnologias de comunicação.

3.2 NFC em Ambientes Pervasivos

3.2.1 IntuiSec

IntuiSec [4] é uma solução que utiliza a NFC para prover uma arquitetura em *middleware* que permite dispositivos móveis fazer uso da tecnologia de comunicação NFC para permitir que os consumidores interajam intuitivamente com os dispositivos de segurança em casas inteligentes. Os autores separaram a infraestrutura em duas camadas: sendo a mais alta delas destinada para interações com usuário para definir configurações de segurança para suas casas; e a camada mais baixa para lidar com a tecnologia NFC.

Essa solução está relacionada com a infraestrutura desenvolvida pois: (i) utiliza a tecnologia NFC para prover interações intuitivas; (ii) provê um tipo de autenticação do usuário dono do aparelho, embora, seja aplicada somente ao domínio específico de casas inteligentes; e (iii) faz a configuração automática da Wi-Fi e Bluetooth.

3.2.2 NCASH

NCASH [5] é uma arquitetura que permite o uso de telefone habilitados com NFC que contendo preferências pessoais predefinidas para controlar dispositivos tais como eletrodomésticos e eletrônicos. Através do NCASH, um telefone NFC não só pode atuar como chave para a entrada de um espaço, mas também o controle personalizado de uma variedade de aparelhos em ambientes inteligentes. O NCASH permite também que o usuário controle estes aparelhos diretamente, sem considerar o contexto.

A arquitetura é composta por quatro partes: front-end (FE), community-end (CE), home-end (HE) e apppliances-end (AE). O FE é implantado na entrada da comunidade para verificar e controlar a entrada das pessoas. O FE é composto por um telefone NFC e um leitor NFC. O CE é responsável por controlar os aparelhos da comunidade. A AE é formado por um conjunto de eletrônicos habilitados para UPnP que são controlados por *gateways* baseados em OSGi.

O NCASH faz relação com a infraestrutura proposta na medida em que: (i) utiliza a tecnologia NFC para prover interações intuitivas com os aparelhos; (ii) utiliza a tecnologia Wi-Fi para comunicação entre os aparelhos; e (iii) faz a ciência de contexto baseada em regras e ontologias.

3.2.3 Solicitação de serviços através de cartões RFID

Salmine e Alakarppa [25] propõem um framework de solicitação de serviços através de toques entre telefone celulares e cartões RFID, de maneira que o celular faz a mediação entre o usuário e os serviços do ambiente local. Neste framework, os cartões RFID estão relacionados com os serviços e, desta maneira, é oferecida uma interação natural entre o usuário e o sistema. Além disso, a informação de contexto como a localização geográfica é utilizada para a correta interpretação da intenção de leitura do cartão RFID e solicitação do serviço.

Essa solução se relaciona com a infraestrutura proposta, pois indica uma alta aceitação da sociedade na utilização da tecnologia de comunicação NFC em conjunto com cartões RFID em ambientes inteligentes.

3.3 Considerações sobre os Trabalhos Relacionados

Apesar das várias soluções que utilizam a tecnologia NFC focadas em ambientes inteligentes fornecendo uma forma intuitiva de interagir com os mesmos e várias soluções em *middleware* para lidar com diferentes tecnologias de comunicação, ainda não é oferecido todas estas funcionalidades juntas e com suporte de desenvolvimento de aplicações pervasivas. Especificamente, não há uma abordagem isolada que forneça de forma integrada as seguintes funcionalidades: (i) suporte a tecnologias de comunicação de longo e curto alcance; (ii) suporte ao chaveamento automático entre estas tecnologias de modo ciente de contexto; e (iii) forneça uma abstração para o desenvolvedor de aplicações pervasivas a fim de acelerar o desenvolvimento das mesmas.

Capítulo 4

A Infraestrutura

Neste capítulo é apresentado o projeto de uma infraestrutura que permite o desenvolvimento de aplicações pervasivas cientes de contexto que fazem o uso de tecnologias de comunicação sem fio de longo alcance tais como Wi-Fi e Bluetooth e a tecnologia de comunicação de curto alcance NFC, com suporte ao chaveamento automático entre as mesmas. Inicialmente, são descritos os requisitos funcionais e não funcionais da infraestrutura. Em seguida é apresentada sua arquitetura, destacando seus principais componentes, suas respectivas responsabilidades e a forma de interação entre elas. Posteriormente, é apresentada a pilha de módulos detalhando como cada módulo da infraestrutura se encaixa na arquitetura e qual o seu papel. Ao fim do capítulo, é apresentado em detalhes o procedimento de chaveamento entre as tecnologias de comunicação.

4.1 Requisitos

Nesta seção serão explicitados os requisitos que a infraestrutura deve preencher para permitir o desenvolvimento de aplicações pervasivas cientes de contexto com suporte ao chaveamento automático entre tecnologias de comunicação de curto e longo alcance.

4.1.1 Requisitos Funcionais

Com o objetivo de permitir o desenvolvimento de aplicações pervasivas ciente de contexto que tenham chaveamento entre tecnologias de comunicação de curto e longo alcance, a in-

fraestrutura deve permitir que os alguns requisitos funcionais sejam atendidos.

Primeiramente a infraestrutura deve ter suporte a tecnologias de comunicação sem fio de curto alcance para a comunicação entre as partes em situações em que: (i) não seja necessário uma alta largura de banda; (ii) possa ter limitações de energia (bateria); (iii) haja restrições de distância entre as partes.

Outro fator importante é que a infraestrutura suporte tecnologias de comunicação sem fio de longo alcance para comunicação em situações onde seja prioridade uma grande largura de banda ou situações onde haja distância entre as partes comunicantes. No escopo deste trabalho, distâncias consideradas longas são distâncias maiores de dez metros, visto que estamos visando aplicações pervasivas em ambientes fechados como um escritório, uma sala de aula, uma casa ou, no máximo, um ambiente de departamento de uma universidade.

Também é importante destacar a necessidade de suporte ao chaveamento entre as tecnologias de comunicação de curto e longo alcance. De maneira geral, os desenvolvedores de aplicações que utilizem a infraestrutura não devem se preocupar com qual tecnologia de comunicação a sua aplicação esteja utilizando, nem tampouco se preocupar em codificar a lógica de troca entre as mesmas. A infraestrutura deve ser capaz de deixar estas funcionalidades transparentes para os desenvolvedores de aplicações pervasivas.

Além disso, a infraestrutura deve utilizar a ciência de contexto para fazer o chaveamento entre as tecnologias de comunicação, e que este seja feito de forma transparente para a aplicação desenvolvida sobre a mesma. Este ponto é fundamental para a infraestrutura pois ela deve observar diversos fatores, ponderá-los e decidir se deve ou não chavear entre as tecnologias e qual tecnologia deve ser escolhida. Os fatores a serem considerados são relacionados ao ambiente tais como presença do usuário, disponibilidade das tecnologias de comunicação, largura de banda disponíveis de cada uma das tecnologias, necessidades das aplicações desenvolvidas por terceiros e níveis de energia disponível.

Por fim, a infraestrutura também deve prover a disponibilização de ponto de conexão para a ligação com outras aplicações desenvolvidas por terceiros. Este ponto de ligação deve ser fornecido através de uma API que forneça uma interface entre a pilha de módulos da infraestrutura e a camada de aplicação.

Na Tabela 4.1 são sumarizados os requisitos funcionais descritos nesta seção.

Código	Requisitos Funcionais
RF1	Suportar tecnologias de comunicação sem fio de curto alcance
RF2	Suportar a tecnologia de comunicação sem fio de longo alcance
RF3	Prover o chaveamento entre as tecnologias de comunicação de curto e longo alcance
RF4	Utilizar ciência de contexto para fazer este chaveamento
RF5	Disponibilizar ponto de ligação para aplicações terceiras

Tabela 4.1: Requisitos funcionais da infraestrutura

4.1.2 Requisitos Não Funcionais

A infraestrutura proposta também deve atender a uma série de requisitos não funcionais.

Primeiramente a infraestrutura deve ser executada em dispositivos móveis, mais especificamente em *smartphones* que iremos chamar aqui de *Device*. Além disso, a infraestrutura também deve ter uma parte fixa, que chamaremos de *Peer*. O *Peer* pode ser qualquer computador fixo como um *desktop* ou *set-top-box*.

Além disso, a infraestrutura deve utilizar as tecnologia de comunicação sem fio Wi-Fi e Bluetooth como tecnologia de comunicação de longo alcance. A escolha do Bluetooth se deve a sua alta disseminação no mercado de dispositivos móveis e também pelo baixo consumo de energia dos seu adaptador. A escolha da tecnologia Wi-Fi se deve também pela alta aceitação no mercado tornando-se tecnologia padrão para comunicação em ambientes pequenos como casa e escritórios.

Outro requisito não funcional importante é o uso da tecnologia de comunicação NFC para comunicação de curto alcance pois não há outra disponível em *smartphones* disponíveis no mercado. Portanto, o *Device* da infraestrutura precisa dispor de adaptador NFC.

Finalmente, o requisito anterior referente à tecnologia NFC gera a necessidade da escolha do android como plataforma de desenvolvimento para *smartphones*, visto que não há ainda outras plataformas usando o NFC e que esteja amplamente disponível no mercado.

Na Tabela 4.2 são sumarizados os requisitos não funcionais descritos nesta seção.

Código	Requisitos Não Funcionais
RNF1	Executar em *smartphones*
RNF2	Utilizar as tecnologias Wi-Fi e Bluetooth para comunicação de longo alcance
RNF3	Utilizar a tecnologia de comunicação NFC para comunicação de curto alcance
RNF4	Utilizar plataforma android

Tabela 4.2: Requisitos não funcionais da infraestrutura

4.2 Arquitetura

A infraestrutura proposta é composta fisicamente por três partes distintas: um cartão RFID, um *Device* e um *Peer*. Cada uma destas três partes atuam de maneira distinta e se integram através das redes ou tecnologias de comunicação sem fio. Esta integração acontece através de troca de dados entre as partes e obedecem a um protocolo específico desenvolvido especificamente neste trabalho.

Na Figura 4.1 é apresentada uma visão geral da arquitetura da infraestrutura detalhando seus componentes: as três partes físicas e as tecnologias de comunicação. Nas próximas Seções são descritos estes componentes e a forma de interação entre eles.

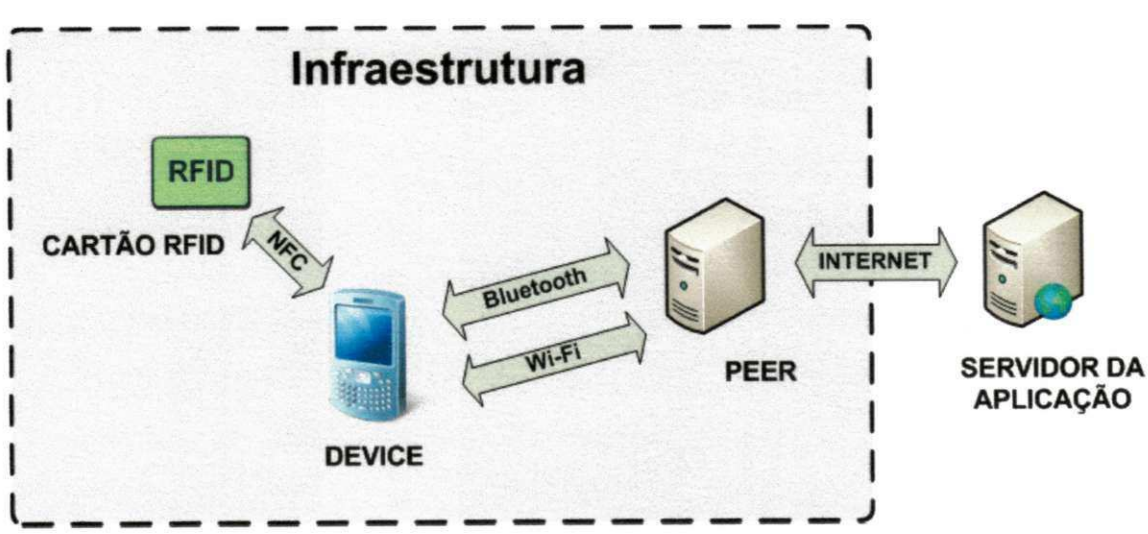


Figura 4.1: Visão geral da Infraestrutura

4.2.1 Cartão NFC

No contexto deste trabalho, o dispositivo NFC é um cartão NFC ou RFID que contém informações tais como dados das tecnologias de comunicação sem fio presentes no ambiente e disponíveis pela infraestrutura e dados de usuário.

Os dados contidos no cartão sobre as tecnologias de comunicação sem fio são informações utilizadas pela infraestrutura para autenticar nas redes Wi-Fi e Bluetooth disponíveis no ambiente. São exemplos de dados sobre a rede Wi-Fi: (i) SSID; (ii) senha *(passphrase)* para autenticação na rede; (iii) mecanismos de segurança como WEP, WPA ou WPA2; e (iv) endereço IP do *Peer* na rede. São exemplos de dados utilizados na rede Bluetooth: (i) UUID; e (ii) endereço MAC do adaptador Bluetooth do *Peer*.

Em relação aos dados do usuário contidos no cartão, são exemplos: (i) nome de usuário; e (ii) número de identificação do usuário. Estes dados são utilizados pela infraestrutura para a autenticação e autorização do usuário na infraestrutura que está sendo executada no ambiente físico onde estão inseridos. A Figura 4.2 ilustra como os dados mencionados são estruturados dentro de um cartão RFID. O mecanismo de autenticação e autorização utilizado neste trabalho é descrito na Seção 4.3.1.

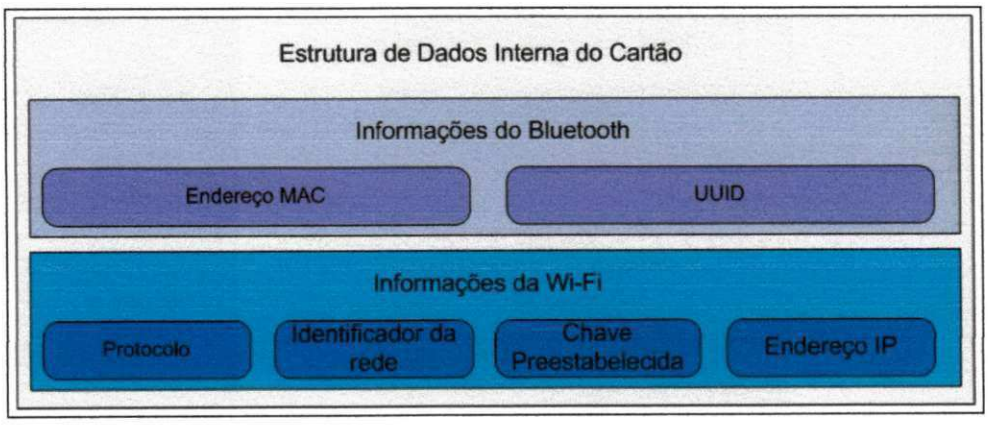


Figura 4.2: Ilustração dos dados contidos em cartão RFID

4.2.2 Device

No contexto deste trabalho, assumimos que o *Device* é dotado de adaptadores Wi-Fi e Bluetooth para comunicação de longo alcance e da tecnologia NFC para comunicação de curto alcance. Assim, o *Device* atua na infraestrutura como um presentante do usuário no ambiente físico no qual a infraestrutura atua. Neste sentido, o simples ato de aproximar ou tocar o aparelho no cartão RFID significa para a infraestrutura que o usuário chegou ao ambiente e que a mesma deve ser iniciada e configurada a partir dos dados lidos no próprio cartão. Os dados lidos e utilizados neste processo são descritos na Seção 4.2.1. O processo é ilustrado na Figura 4.3.

A leitura do cartão RFID pelo *Device* através do NFC faz disparar automaticamente a execução da infraestrutura iniciando o processo de autenticação e autorização do usuário, as conexões das tecnologias de comunicação disponíveis no aparelho, e a conexão com o *Peer*.

É importante ressaltar que o componente da infraestrutura executado no *Device* é responsável por obter as informações de contexto e do ambiente onde está o aparelho e, consecutivamente, as informações do usuário. Mais detalhes dos módulos do *Device* podem ser encontrados na Seção 4.3.

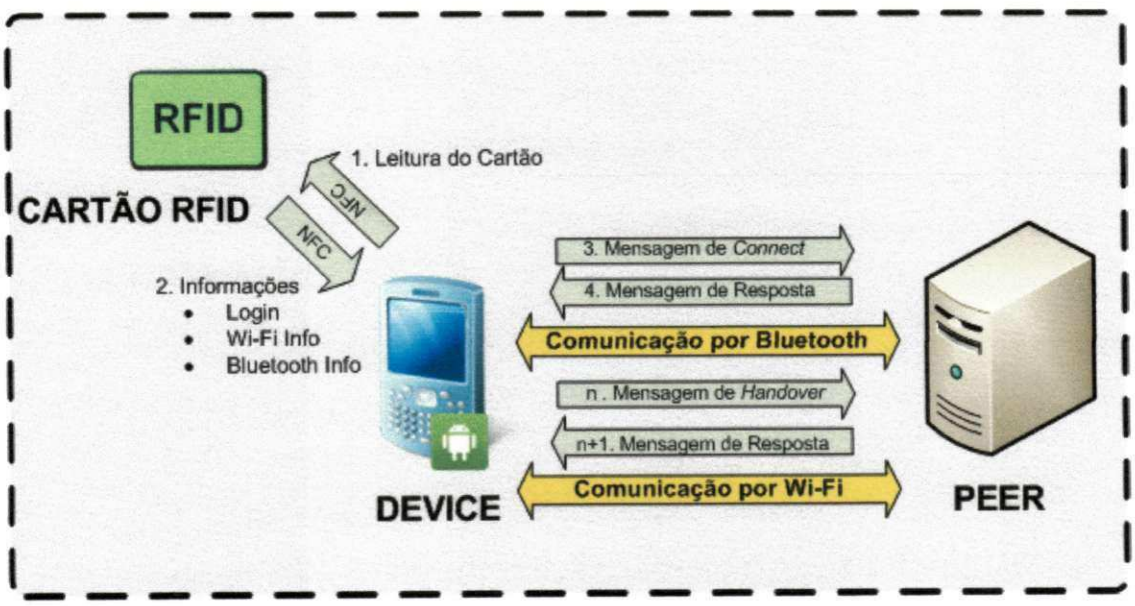


Figura 4.3: Sequência de execução da infraestrutura

4.2.3 Peer

O *Peer* é a parte fixa da infraestrutura que concentra o componente de software da infraestrutura e que se comunica com o *Device*. Assim, o *Peer* não tem responsabilidade de ter ciência de contexto ou ter sensibilidade do ambiente no qual ele está inserido assim como o *Device*. Ele apenas responde as mensagens recebidas do *Device* ou as repassa para sua camada de aplicação. Mais detalhes dos módulos do *Peer* podem ser encontrados na Seção 4.3.

O *Peer* necessariamente deve ser provido de tecnologias de comunicação sem fio de longo alcance que sejam utilizadas na infraestrutura. No entanto, não é necessário que o *Peer* tenha conectividade de curto alcance como a NFC. O *Peer* tem disponível as tecnologias Wi-Fi e Bluetooth, apenas.

4.3 Pilha de Módulos

Nesta seção, iremos detalhar os módulos da infraestrutura que formam o componente executado no *Device* e também os relativos ao componente executado no *Peer*.

4.3.1 Módulos do Device

O componente de software que executa no *Device* é composto por sete módulos que funcionam de forma integrada e, dessa maneira, formam a parte principal da infraestrutura. Os nomes dos módulos em questão são: Device Manager, Handover Manager, Context Aware Manager, Device Monitor, WiFi Service, Bluetooth Service e Nfc Manager. Abaixo são descritos cada um destes módulos. A Figura 4.4 ilustra os módulos que fazem parte da infraestrutura no *Device*.

1. **Device Manager:** Módulo responsável por gerenciar a infraestrutura do dispositivo integrando os outros módulos e fornecendo uma única forma de acesso aos serviços da infraestrutura. Ademais, funciona como uma fachada para a API da infraestrutura do *Device*, um ponto de ligação ou *hot spot* (segundo a nomenclatura utilizada por [8, p. 222]) que é utilizado pelas aplicações para acessar a infraestrutura.

2. **Handover Manager:** Módulo responsável por centralizar e gerenciar as informações necessárias para a decisão do chaveamento entre as tecnologias de comunicação. É a

partir deste módulo que é disparada a requisição de chaveamento (*handover request*) e onde é esperada a sua resposta (*handover response*), do *Peer*. Mais detalhes sobre o chaveamento na Seção 4.4.

3. **Context Aware Manager:** Módulo encarregado de monitorar o contexto no qual o aparelho se encontra. As informações que este módulo processa são dados que indicam a necessidade de economia de energia, alteração do sinal das tecnologias de comunicação (Wi-Fi e Bluetooth) e o cálculo do custo de cada umas destas tecnologias. Mais detalhes sobre o cálculo de custo na Seção 4.4.2.

4. **Device Monitor:** Módulo responsável por monitorar os detalhes do dispositivo e notificar ao *Context Aware Manager*. Este módulo é executado a cada dez segundos.

5. **WiFi Service:** Este módulo gerencia o serviço de Wi-Fi para o dispositivo fornecendo uma interface única de acesso para a infraestrutura acessar os recursos da API Wi-Fi do dispositivo.

6. **Bluetooth Service:** Módulo que faz a gerência do serviço de Bluetooth para o dispositivo fornecendo uma interface única de acesso para a infraestrutura acessar os recursos da API Bluetooth do dispositivo.

7. **NFC Manager:** Módulo responsável por centralizar e fornecer uma interface única para o adaptador NFC para a infraestrutura. Ademais, lê os dados citados na Seção 4.2.1 e os fornece de forma estruturada para o uso dos outros módulos da infraestrutura.

4.3.2 Módulos do Peer

De maneira análoga, o componente de software executado no *Peer* é composto por cinco módulos que executam de forma integrada provendo e fornecendo serviços ao *Device*. Estes módulos são nomeados da seguinte maneira: Peer Manager, Handover Manager, Authentication Manager, WiFi Service e Bluetooth Service. Abaixo são descritos cada um destes módulos. A Figura 4.5 ilustra os módulos que fazem parte da infraestrutura no *Peer*.

1. **Peer Manager:** Módulo responsável por gerenciar a infraestrutura do *Peer* integrando os outros módulos e fornecendo uma única forma de acesso aos serviços da infra-

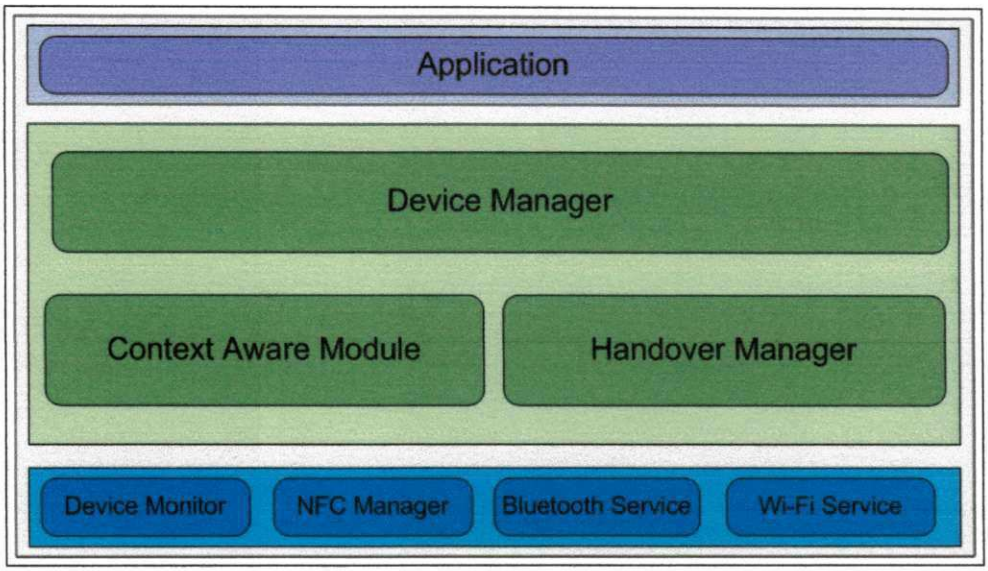


Figura 4.4: Pilha de módulos da infraestrutura no *Device*

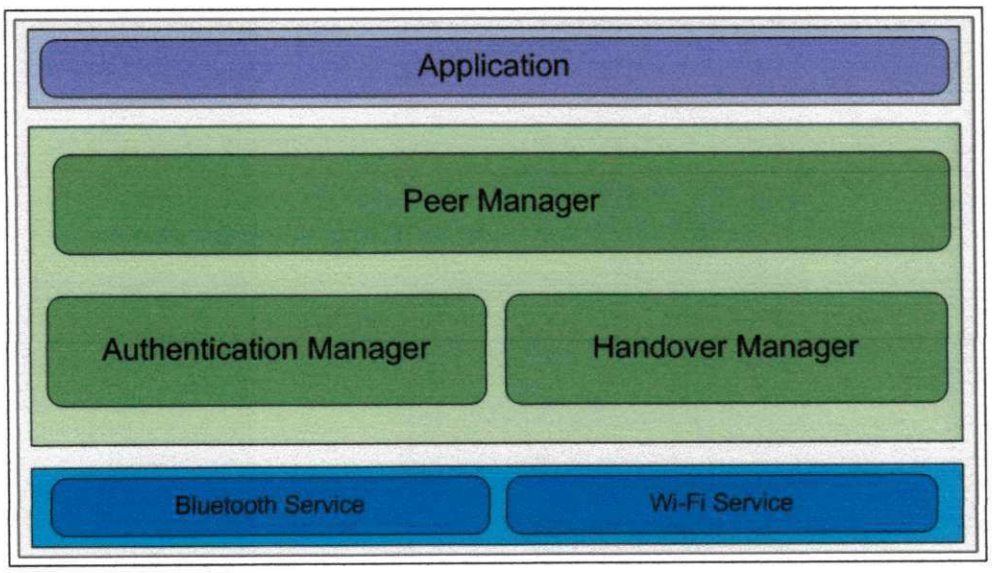


Figura 4.5: Pilha de módulos da infraestrutura no *Peer*

estrutura. Ademais, funciona como uma fachada para a API da infraestrutura *Peer*, funcionando como um hot spot, para que a infraestrutura possa ser usada e acessada pela aplicações.

2. **Handover Manager:** Módulo responsável por centralizar e gerenciar as informações necessárias para a execução do chaveamento entre as tecnologias de comunicação. Este módulo recebe mensagens de *handover request* e responde com mensagem de *handover response*.

3. **Authentication Manager:** Módulo encarregado de monitorar o contexto no qual o aparelho se encontra. As informações que este módulo processa são dados que indicam a necessidade de economia de energia, alteração do sinal das tecnologias de comunicação(Wi-Fi e Bluetooth) e o cálculo do custo de cada umas destas tecnologias. Mais detalhes sobre o cálculo de custo na Seção 4.4.2.

4. **WiFi Service:** Análogo ao módulo homônimo do *Device*.

5. **Bluetooth Service:** Análogo ao módulo homônimo do *Device*.

4.4 Chaveamento

Nesta Seção serão detalhados os aspectos relacionados ao chaveamento entre tecnologias de comunicação. Mais especificamente, serão detalhados o protocolo, política de tomada de decisão e cálculo de custo de uma tecnologia de comunicação.

O chaveamento é o processo de troca de um canal de comunicação para outro durante uma transferência de dados sem que haja interrupção do serviço. Além disso, é um processo que deve acontecer de forma transparente para o usuário do serviço.

No escopo deste trabalho, o chaveamento é aplicado entre as tecnologias de comunicação disponíveis tanto no *Device* como no *Peer* e sua execução acontece de maneira sincronizada e baseada na troca de mensagens entre ambos. Esta troca de mensagens é definida pelo protocolo da infraestrutura desenvolvido neste trabalho e detalhado na Seção 4.4.1.

4.4.1 Protocolo

Nesta seção será detalhado o funcionamento do protocolo de comunicação utilizado na infraestrutura. O protocolo é a convenção ou padrão que controla e possibilita a conexão, comunicação e transferência de dados entre o *Device* e *Peer*.

Na infraestrutura proposta, o protocolo de comunicação especifica um padrão que tem como propriedades: (i) o estabelecimento de ligação entre as partes, o *handshaking*; (ii) autenticação de usuário; (iii) estabelecimento do chaveamento entre tecnologias de comunicação; e (iv) término da sessão ou conexão.

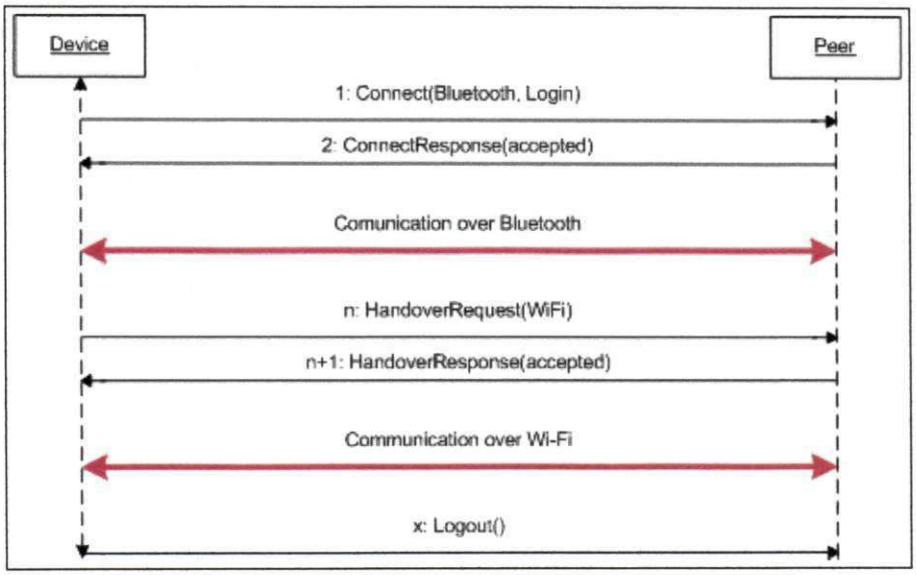


Figura 4.6: Protocolo de comunicação da infraestrutura

A Figura 4.6 ilustra a sequência de mensagens trocadas entre o *Device* e o *Peer* durante a execução da infraestrutura.

Estas mensagens podem ser categorizadas em dois tipos: **mensagem de protocolo** e **mensagem de dados**. A mensagem de dados é uma mensagem provinda da camada de aplicação do *Peer* ou *Device* e apenas é repassada para a camada de aplicação da outra parte. Assim, a infraestrutura recebe a mensagem da aplicação e adiciona apenas um byte de sinal para marcar a mensagem como sendo mensagem de dados e envia através da infraestrutura. A parte destino recebe a mensagem e envia para a camada de aplicação.

A mensagem de protocolo é uma mensagem que tem origem da própria infraestrutura e é utilizada para controle interno, não sendo recebida e percebida pelas aplicações que executam sobre a infraestrutura. Este tipo de mensagem é utilizada para comunicação entre as partes no que diz respeito a implementação das propriedades do protocolo. A Figura 4.7 ilustra a diferença nos caminhos que as mensagem de dados e de protocolo fazem dentro da infraestrutura.

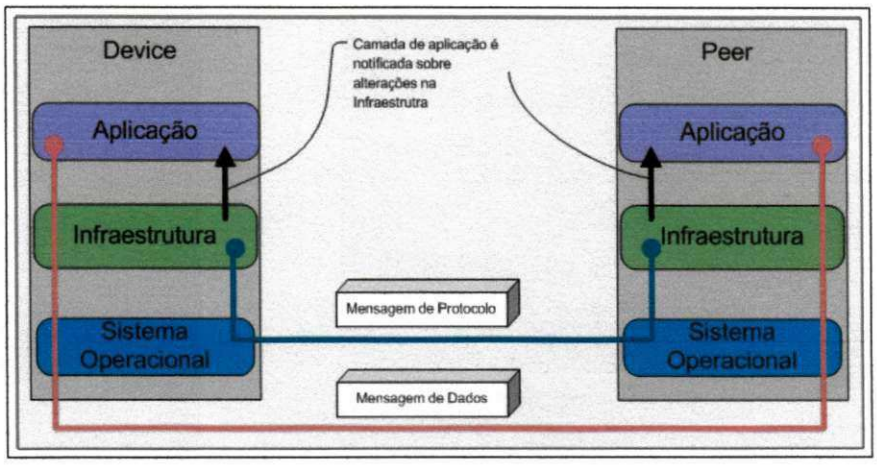


Figura 4.7: Caminho que os dois tipos de mensagens percorrem na pilha da infraestrutura

Abaixo são listadas as mensagens de protocolo utilizadas pela infraestrutura bem como sua descrição e parâmetros.

- **Connect Request:** É a primeira mensagem entre as partes e é enviada a partir do aparelho *Device* com destino para o *Peer*. Os parâmetros desta mensagem são:
  - *NetworkTarget*: Referencia a tecnologia de comunicação que o emissor da mensagem deseja se conectar com o receptor da mesma.
  - *Username*: Parâmetro que identifica o usuário que deseja se autenticar na infraestrutura.
  - *Password*: Parâmetro que identifica a senha para o usuário que deseja se autenticar na infraestrutura.
- **Connect Response:** Mensagem de resposta ao *Connect Request* enviada do *Peer* para o *Device*. Esta mensagem contém os seguintes parâmetros:

  - *Response*: Parâmetro que identifica a resposta para o *Connect Request* enviado. O valor deste é binário sendo *true* para resposta positiva e *false* para resposta negativa.
  - *NetworkTarget*: Referencia a tecnologia de comunicação que o *Device* foi autenticado e será aberta a conexão. Caso a resposta da autenticação seja negativa, este parâmetro será vazio.

- **Handover Request:** É a mensagem principal do protocolo que representa uma requisição de chaveamento. Neste trabalho, apenas o *Device* envia esta mensagem com o seguinte parâmetro:
  - *NetworkTarget*: Referencia a tecnologia de comunicação na qual o emissor da mensagem (o *Device*) deseja se comunicar com o receptor (o *Peer*).

- **Handover Response:** Mensagem de resposta ao *Handover Request* enviada do *Peer* para o *Device*.
  - *Response*: Parâmetro que identifica a resposta para o *Handover Request* enviado. O valor deste é binário sendo *true* para resposta positiva e *false* para resposta negativa.
  - *NetworkTarget*: Referencia a tecnologia de comunicação para a qual o *Peer* vai chavear.

- **Logout:** Mensagem de *logout* enviada do *Peer* para o *Device*.

As mensagens de protocolo e dados são estruturadas pela infraestrutura em um array de bytes sendo cada posição ou elemento do array uma informação que deve ser passada para o destino. No caso da mensagem de dados é adicionado ao array de dados apenas um elemento que representa o byte de sinal sinalizando que a mensagem contém dados. Assim, uma mensagem de dados tem 1 byte de sinal e 1023 bytes de dados totalizando um array de tamanho de 1024 bytes.

No caso de mensagem de protocolo, o empacotamento é feito de maneira diferente pois o mesmo é estruturalmente maior devido ao maior número de elementos de controle do

protocolo. Uma mensagem de protocolo tem 1 byte de sinal, 1 byte para o Argumento 1, 1 byte para o Argumento 2 e 29 bytes para o Array de Bytes totalizando 32 bytes.

Na lista abaixo são descritos os elementos do array utilizados para formar as mensagem do protocolo.

- *Byte de sinal:* Byte utilizado para sinalizar se a mensagem é uma mensagem que contém dados ou uma mensagem de protocolo. Tamanho de 1 byte.
- *Array de bytes:* Array utilizado para carregar os dados entre o remetente e o destinatário. É a parte útil da mensagem e dependendo do tipo de mensagem de protocolo pode ser vazio ou não. Tamanho de 1023 bytes e 29 bytes para mensagem de dados e mensagem de protocolo, respectivamente.
- *Argumento 1:* Byte utilizado para distinguir o tipo de mensagem de protocolo. Utilizado apenas em mensagens de protocolo. Tamanho de 1 byte.
- *Argumento 2:* Byte auxiliar utilizado para passar parâmetros para o destinatário. O parâmetro depende do tipo da mensagem de protocolo podendo guardar um valor binário ou um inteiro que referencia a tecnologia de comunicação (Bluetooth ou Wi-Fi). Também utilizado apenas em mensagens de protocolo. Tamanho de 1 byte.

A Figura 4.8 ilustra os tipos de mensagem utilizados no protocolo da infraestrutura.

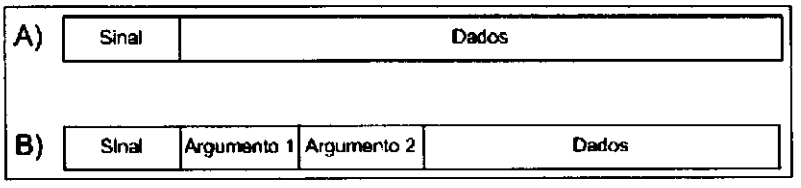


Figura 4.8: Estrutura da mensagem utilizada na infraestrutura: A) Mensagem de Dados; B) Mensagem de Protocolo

A listagem de código fonte 4.1 mostra o trecho de código no qual é montada a estrutura das mensagens de protocolo e de dados utilizada pela infraestrutura. Este trecho é chamado pelo segundo trecho de código 4.2. Observe que em 4.2 o argumento byte de sinal é o que diferencia o tipo de mensagem a ser criado.

Finalmente, a listagem de código fonte 4.3 mostra como são criados os diferentes tipos de mensagem de protocolos e quais os argumentos são utilizados.

Código Fonte 4.1: Trecho de código onde a mensagem é formatada

```
        Integer signalInt = new Integer(sinal);
        Integer arg1Int = new Integer(arg1);
        Integer arg2Int = new Integer(arg2);
        byte[] header = {signalInt.byteValue(), arg1Int.byteValue(),
            arg2Int.byteValue()};
        byte[] message = Util.concat(header , data);
        return message;
}
```

Código Fonte 4.2: Trecho de código onde há a diferenciação entre os tipos de mensagens

```
byte[] formatDataMessage(byte[] data){
        return formatMessage(DATA_CONTENT, data , -1, -1);
}

byte[] formatProtocolMessage(byte[] data , int arg1 , int arg2){
        return formatMessage(PROTOCOL_CONTENT, data , arg1 , arg2);
}
```

Código Fonte 4.3: Trecho de código onde os parâmetros da mensagem de protocolo são definidos de acordo com o tipo específico

```
byte[] createConnectMessage(Network networkToBeUsed){
        return MessageCenter.formatProtocolMessage(DeviceManager.
            getInstance().getLoginInfo(), DeviceMessageCenter.CONNECT,
            networkToBeUsed.getValue());
}

byte[] createConnectResponseMessage(int accepted){
        return DeviceMessageCenter.formatProtocolMessage(null ,
        MessageCenter.CONNECTED_RESPONSE, accepted);
}
```

```
byte[] createHandoverRequestMessage(Network NetworkTarget) {
        return MessageCenter.formatProtocolMessage(null,
            DeviceMessageCenter.HANDOVER_REQUEST, NetworkTarget.getValue()
            );
}

byte[] createHandoverResponseMessage(int accepted, Network NetworkTarget)
    {
        return MessageCenter.formatProtocolMessage(null, MessageCenter.
            HANDOVER_RESPONSE, accepted);
}
```

4.4.2 Política de Decisão

Nesta seção será mostrado em detalhes o processo de decisão e as regras que disparam uma mensagem de *Handover Request* na infraestrutura. Estes detalhes contemplam as regras definidas pela infraestrutura, as políticas e como estes diversos fatores são ponderados no algorítmo de cálculo de custo para uma determinada tecnologia de comunicação.

Durante a execução da infraestrutura existem duas formas de se chegar a um *Handover Request*: por regras de disparo ou por cálculo de função de custo.

Regras

O disparo através de regras acontece necessariamente quando situações específicas geram bloqueio na comunicação entre as partes da infraestrutura. Mais especificamente, qualquer situação que possa acontecer durante o uso de uma determinada tecnologia de comunicação que possa interromper o seu funcionamento gera a necessidade da infraestrutura de disparar um *Handover Request* para se recuperar e continuar a comunicação de maneira que o usuário não precise tomar alguma ação. É uma ação de *fallover*.

Os casos mais comuns para este tipo de interrupção são situações quando ou o Bluetooth ou a Wi-Fi fica inacessível, fora de alcance ou simplesmente deixa de funcionar. Neste caso, a infraestrutura obrigatoriamente envia um *Handover Request* para se recuperar e manter a comunicação.

Quando ambos, Bluetooth e Wi-Fi, ficam incapazes de prover comunicação a infraestrutura inevitavelmente cessa o provimento da comunicação e notifica a camada de aplicação sobre o estado em que se encontra. Neste caso, mesmo assim a infraestrutura continua tentando periodicamente a cada 10 segundos se comunicar através das duas tecnologias de comunicação até que seja possível se conectar novamente através de alguma tecnologia. Uma vez conseguida a conexão, a infraestrutura notifica a camada de aplicação e continua o seu funcionamento normalmente.

Regras	**Descrição**
Regra 1	Wi-Fi se torna indisponível
Regra 2	Bluetooth fora de alcance

Tabela 4.3: Regras de Disparo de um *Handover Request*

A infraestrutura se mantém atualizada sobre o estado das tecnologias de comunicação através de mensagens de *Ping* que são trocadas entre o *Device* e o *Peer*. Quando o *Device* identifica alguma das regras descritas na Tabela 4.3 ele dispara um *Handover Request* destinado ao *Peer* e segue o protocolo descrito na Seção 4.4.1. Nestes casos, quando as mensagens de *Ping* não são respondidas por uma tecnologia de comunicação o *Handover Request* é enviado através da outra tecnologia de comunicação. Por exemplo, se a comunicação com o *Peer* deixar de funcionar através do Wi-Fi o *Device* irá enviar uma *Handover Request* através do Bluetooth.

É importante ressaltar que estas regras fazem parte do código da infraestrutura e não podem ser alteradas pelas aplicações.

Cálculo de Custo

O cálculo de custo considera dois tipos de fatores na sua função: variáveis mensuradas durante a execução da infraestrutura e políticas definidas pela aplicação e/ou usuário. O módulo responsável pela medição é o *DeviceMonitor* e módulo responsável pela inferência do ambiente é o *Context Aware Module*, sendo ambos executados no *Device*.

Variáveis Mensuradas

As variáveis mensuradas durante a execução da infraestrutura são métricas que indicam como está o funcionamento da infraestrutura em um certo instante. Estas métricas dizem respeito ao desempenho de cada uma das tecnologias de comunicação bem como ao desempenho específico do *Device* e são executadas periodicamente. O intervalo de tempo entre duas execuções sucessivas deve ser definido pela camada de aplicação pois deve respeitar as características da aplicação.

A utilização de métricas propicia a infraestrutura um acompanhamento do seu desempenho e, aliado a isso, a possibilidade de analisar qual tecnologia de comunicação é a melhor opção no determinado instante a partir da função de custo. As variáveis mensuradas são detalhadas abaixo:

- *Nível de Bateria*: Variável que referencia o nível de carga que a bateria do *Device* possui no determinado instante. Valores são na faixa de 0 a 100.
- *Força do Sinal do Wi-Fi*: Referencia a força do sinal do Wi-Fi especialmente. Não há informação equivalente para Bluetooth devido a inexistência dessa métrica na API do *Device* (sistema operacional android). Valores são na faixa de 0 a 100.
- *Largura de Banda*: Referencia a largura de banda que a infraestrutura utiliza em um determinado instante. A medição é feita a cada 10 segundos, e o valor é atualizado. Medida em Kbps.

Políticas

As políticas são definidas pela camada de aplicação (ou usuário) e são diretrizes de funcionamento que são tomadas pela infraestrutura e influenciam na decisão do *handover* na medida em que atua no cálculo de custo das tecnologias de comunicação envolvidas. Esta atuação acontece através de ponderações de acordo com a política definida. Mais adiante será detalhada a função de cálculo de custo e a maneira na qual estas políticas influenciam o cálculo do custo.

A importância destas políticas se dá pela possibilidade dos desenvolvedores de aplicações customizarem a infraestrutura definindo políticas de uso de acordo com as necessidades es-

pecíficas de cada aplicação. As políticas podem ser definidas em três tipos conforme abaixo:

- *Mínima Largura de Banda Requerida*: Política que define mínima largura de banda requerida pela aplicação. Largura de banda medida em Kbps.
- *Estratégia de Consumo de Bateria*: Política que define a estratégia de consumo de bateria. Pode assumir três valores: Modo Econômico, Moderado, Alto Desempenho.
- *Preferência do Usuário*: Política que define qual a tecnologia de comunicação de preferência do usuário ou da aplicação. Pode assumir dois valores: Bluetooth ou Wi-Fi.

Função de Cálculo de Custo

Para definir qual tecnologia de comunicação deve ser utilizada pela infraestrutura neste trabalho, utilizamos a lógica de cálculo de custo apresentado em [39] como base. Desta maneira, o cálculo é realizado por uma função, que chamaremos de função de custo.

A função de custo (Equação 4.1) define o custo C^n para uma dada tecnologia de comunicação n como sendo a divisão da sua eliminação E^n por sua qualidade de serviço Q^n.

$$C^n = \frac{E^n}{Q^n} \quad \text{onde} \quad C^n = [0,1] \tag{4.1}$$

Por sua vez, a eliminação E^n (Equação 4.2) para uma tecnologia de comunicação n é definida como sendo o produtório das eliminações individuais Ei. Onde i representa uma restrição definida por uma política.

$$E^n = \prod_i E_i^n \tag{4.2}$$

A restrição Ei (Equação 4.3) é definida como sendo a constante 1 (um) dividida por Ii, onde novamente i representa uma restrição definida por uma política . Assim Ii referencia individualmente uma única restrição que deve ser atendida de acordo o com o que for definido pela aplicação para a tecnologia de comunicação n.

Quando a restrição for satisfeita *Ei* ficará com valor 1 (um), quando a restrição não for satisfeita *Ei* assume um valor muito próximo de zero. Neste último caso, o valor de ε*(epsilon)* é definido como o menor valor que sistema operacional pode representar (e.g., $1e^{-7}$). Desta maneira, é possível que as restrições impossibilitem a escolha de uma tecnologia de comunicação n pois o valor de E^n aumentaria o seu custo C^n.

$$E_i = \frac{1}{I_i} \quad \text{onde} \quad I_i = \begin{cases} \epsilon, & \text{restrição } i \text{ não satisfeita} \\ 1, & \text{restrição } i \text{ satisfeita} \end{cases} \tag{4.3}$$

Finalmente, a qualidade do serviço Q^n (Equação 4.4) de uma tecnologia de comunicação n, é definida como sendo o somatório do produto do peso w pelo quantificador de qualidade q para cada variável mensurada j. Ou seja, o resultado de Q^n mensura a qualidade do serviço de uma tecnologia de comunicação n ponderando a qualidade de cada um dos parâmetros.

$$Q^n = \sum_j w_j^n q_j^n \tag{4.4}$$

Categoria	**Peso(*w*)**
Econômico	9
Moderado	5
Alta Performance	0

Tabela 4.4: Pesos utilizados na função de custo para a Política *"Estratégia de consumo de bateria"*

A ponderação é feita através do peso, dado por w, que por sua vez é extraído das políticas definidas pela a camada de aplicação. A qualidade q é medida em tempo de execução. Cada parâmetro j de q referencia uma variável mensurada para a tecnologia de comunicação n. As políticas definidas pela camada de aplicação bem como as variáveis mensuradas são apresentadas e detalhadas na Seção 4.4.2.

A Tabela 4.4 mostra os pesos w adotados para a política "Estratégia de Consumo de Bateria" definida na Seção 4.4.2. Estes pesos foram definidos empiricamente e seguem a idéia de que quanto mais severa e crítica for a política maior deve ser o peso da bateria - w(bateria) - na função de cálculo de custo.

Variável Mensurada	**Restrição**	**Parâmetro**	
		Peso(w)	**Qualidade(q)**
Nível de Bateria	Maior que 10%	Ver Tabela 4.4	Nível mensurado em %
Largura de Banda	Maior que Requerida	1 - w(bateria)	Requerida - Mensurada

Tabela 4.5: Variável Mensurada x Restrição/Parâmetro

Portanto, a tecnologia de comunicação escolhida em um determinado instante será a que tiver o menor valor C^n dentre todas as n tecnologias de comunicação disponíveis, conforme a Equação 4.5.

$$Min(C^1, C^2, \ldots, C^n) \tag{4.5}$$

É importante ressaltar que quando é impossível calcular a qualidade(q) da Wi-Fi, no caso quando ainda não está em uso, a infraestrutura utiliza a política "Força de Sinal" em substituição para realizar o cálculo do custo da tecnologia de comunicação Wi-Fi.

4.5 Conclusões do Capítulo

Neste capítulo foram discutidos os principais aspectos relacionados ao projeto da infraestrutura para desenvolvimento de aplicações pervasivas cientes de contexto com suporte ao chaveamento automático entre tecnologias de comunicação sem fio de longo e curto alcance. Inicialmente foram apresentados os requisitos funcionais e não funcionais da infraestrutura. Com base nesses requisitos, foi especificada e mostrada a arquitetura da infraestrutura. Em

seguida, foram descritos os detalhes do chaveamento automático como: (i) o protocolo utilizado; (ii) detalhamento de cada um dos módulos que fazem parte da infraestrutura; e (iii) a forma na qual os módulos interagem entre si para prover as funcionalidades especificadas. Finalmente, foi mostrado como é calculado o custo de uma tecnologias de comunicação e como é possível definir o chaveamento a partir do custos das tecnologias de comunicação calculados.

Capítulo 5

Estudo de Caso

Neste capítulo é apresentado o estudo de caso para demonstrar o suporte da infraestrutura à criação de aplicações pervasivas com chaveamento automático e ciente de contexto entre tecnologias de curto e longo alcance. A aplicação desenvolvida no estudo de caso apresenta funcionalidades que fazem uso de todas as características da infraestrutura para prover uma interação intuitiva com o usuário em um ambiente pervasivo.

O capítulo está organizado da seguinte maneira: inicialmente, é feita uma sucinta descrição da aplicação, mostrando as suas principais funcionalidades e descrevendo como elas interagem com a infraestrutura. Após isto, serão detalhados aspectos relacionados ao desenvolvimento e configuração da aplicação bem como seu funcionamento e execução.

5.1 Requisitos

IntuitivePresenter é uma aplicação pervasiva que utiliza a infraestrutura proposta neste trabalho e tem como objetivo prover aos seus usuários uma forma rápida, fácil e intuitiva de configurar, iniciar e apresentar seus slides em conferências e salas de aula. Esta aplicação pode ser utilizada, por exemplo, em cenários de ambientes pervasivos como o descrito a seguir:

João é um professor e foi convidado por outra universidade para dar uma aula especial. Ele carrega consigo seu smartphone equipado com as tecnologias NFC, Wi-Fi e Bluetooth e que contém o arquivo com a apresentação. Ao chegar na sala de aula, João aproxima seu smartphone de um Interaction Point(um cartão RFID), que contém dados das redes - Wi-Fi

e Bluetooth - da sala de aula e que são lidos pelo smartphone. Com isso, o usuário João através do smartphone, é autenticado no ambiente pervasivo, o aplicativo de apresentação de slides é iniciado e o arquivo é automaticamente passado para projetor que prontamente inicia a apresentação.

A escolha da tecnologia de comunicação utilizada é feita automaticamente pelo smartphone e é determinada por fatores como tipo de aplicação, preferência do usuário e nível de bateria do aparelho. Tudo isso acontece de maneira automática, transparente para o João e em poucos segundos. João inicia sua palestra utilizando o próprio smartphone como controle da apresentação, de maneira que ele manipula os slides para frente e para trás com apenas o aparelho. Se durante a apresentação o aparelho mostrar-se com pouca carga de bateria, a conexão é trocada de Wi-Fi para Bluetooth. Em caso de o palestrante decidir se afastar do palco e, por isso, perder o sinal do Bluetooth, a conexão é chaveada de volta para o Wi-Fi. Por fim, a aplicação é fechada e usuário João faz o logoff do ambiente.

Código	**Funcionalidade**
F1	Autenticar o usuário no ambiente através da infraestrutura
F2	Transferir arquivos de apresentação como PPT e PDF para a projeção
F3	Iniciar, controlar e finalizar a apresentação através da aplicação
F4	Fazer logoff no ambiente através da infraestrutura
F5	Notificar o usuário os estados da infraestrutura

Tabela 5.1: Funcionalidades da aplicação IntuitivePresenter

De maneira específica, a aplicação IntuitivePresenter utiliza a infraestrutura proposta para autenticar o usuário nas redes disponíveis no ambiente. A aplicação também define configurações que serão levadas em conta pela infraestrutura para que ela possa decidir qual tecnologia deve ser utilizada no ambiente pervasivo bem como decidir fazer o chaveamento entre estas tecnologias de comunicação quando necessário.

Na Tabela 5.1 está sumarizado o conjunto de funcionalidades providas pela aplicação IntuitivePresenter.

5.2 Desenvolvimento da Aplicação

Nesta seção é descrito como foi realizado o desenvolvimento da aplicação IntuitivePresenter, a partir dos requisitos descritos na Tabela 5.1.

A aplicação IntuitivePresenter foi desenvolvida para ser executada em dois dispositivos: no *Device* (*smartphone*) e no *Peer*. A parte móvel da aplicação IntuitivePresenter foi desenvolvida para ser executada na plataforma de dispositivos android, versão 2.3, batizada de *Gingerbread*, com *API level 10*. A aplicação requer necessariamente que o *smartphone* tenha o adaptador de NFC em sua especificação de hardware. Com base nestas especificações citadas, para a execução da aplicação IntuitivePresenter, foi utilizado o *smartphone* Samsung Google Nexus S cujas características são mostradas na Tabela 5.2.

Característica	**Descrição**
Fabricante	Samsung
Sistema Operacional	Android 2.3
Processador	Hummingbird 1 GHz Cortex-A8
Memória	512 MB RAM
Armazenamento	16GB
Wi-Fi	802.11 b/g/n
Bluetooth	Bluetooth 2.1
Outras Conexões	NFC e GPS

Tabela 5.2: Especificação do dispositivo Google Nexus S

A parte do *Peer* foi desenvolvida para ser executada sob a plataforma Java, mais especificamente Java versão 1.6. A aplicação requer necessariamente que o *Peer* tenha o adaptador de Bluetooth e Wi-Fi em sua especificação de hardware. Com base nestas especificações citadas, para a execução da aplicação IntuitivePresenter, foi utilizado o notebook Dell Inspiron 1525 cujas características são mostradas na Tabela 5.3.

Característica	Descrição
Fabricante	Dell
Sistema Operacional	Windows 7
Processador	Intel Core 2 Duo T5450, 1.66 GHz
Memória	4 GB RAM
Armazenamento	320Gb
Wi-Fi	802.11 b/g/n
Bluetooth	USB Bluetooth Adapter("Dongle")

Tabela 5.3: Especificação do *Peer* Dell Inspiron 1525

5.2.1 Implementação das Funcionalidades da Aplicação

Nesta seção são descritos detalhes de como as funcionalidades da aplicação IntuitivePresenter foram implementadas a partir da interface definida pela infraestrutura. A descrição abaixo é dividida entre as duas versões da IntuitivePresenter: *Device* e *Peer*.

Device

Nesta seção será descrito como a infraestrutura e aplicação IntuitivePresenter do *Device* interagem entre si. Esta interação acontece de três maneiras: (i) notificações da infraestrutura; (ii) envio de bytes para o *Peer*; e (iii) definição de políticas. Abaixo são detalhadas estas interações.

- **Notificações da Infraestrutura:** A aplicação IntuitivePresenter recebe notificações da infraestrutura sobre suas alterações de estado e notifica o usuário da aplicação através de uma mensagem de diálogo. Em android estas são chamadas de *Toast*. Os tipos de notificação são:
  - *Recebimento de bytes*: Enviada pela infraestrutura para a aplicação quando a mesma recebe mensagem de Bytes endereçadas a aplicação. Esta notificação é acompanhada dos próprios bytes recebidos do *Peer*.
  - *Notificação de Handover*: Enviada pela infraestrutura notificando o que o chaveamento foi feito. Detalha qual a tecnologia que está em uso no momento.

- *Notificação de Login*: Emitida pela infraestrutura significando que o usuário foi autenticado na infraestrutura.
- *Notificação de Logoff*: Enviada pela infraestrutura notificando que o usuário saiu do ambiente da infraestrutura.
- *Notificação de Conexão*: Emitida pela infraestrutura notificando que o *Device* e a aplicação estão conectados a mesma.
- *Notificação de Perda de Conexão*: Enviado pela infraestrutura quando não há conexão com o *Peer* através das todas as tecnologias de comunicação.

- **Envio de bytes para o *Peer*:** A aplicação IntuitivePresenter envia bytes para o *Peer* através do método *write* do *Device Manager*.
- **Definição de Políticas:** A aplicação IntuitivePresenter define as políticas mencionadas na Seção 4.4.2 através do método *setPolicies()* do módulo *Device Manager* no início da aplicação e quando estas políticas forem alteradas.

Peer

Nesta seção será descrito como a infraestrutura e aplicação IntuitivePresenter do *Peer* interagem entre si. Esta interação acontece por dois tipos de interações: notificações da infraestrutura e envio de bytes para o *Peer*. Abaixo são detalhadas estas interações.

- **Notificações da Infraestrutura:** A aplicação IntuitivePresenter recebe notificações da infraestrutura sobre suas alterações de estado. Os tipos de notificação são:
  - *Recebimento de bytes*: Enviada pela infraestrutura quando a mesma recebe mensagem de Bytes endereçadas a aplicação. Esta notificação é acompanhada dos próprios bytes recebidos do *Device*.
  - *Notificação de Chaveamento*: Enviada pela infraestrutura notificando o que o chaveamento foi feito. Detalha qual a tecnologia que está em uso no momento. Nesta notificação não há exibição mensagem para o usuário.
  - *Notificação de Login*: Emitida pela infraestrutura significando que o usuário foi autenticado na infraestrutura. Especialmente na atual implementação da IntuitivePresenter não há exibição mensagem para o usuário.

- *Notificação de Logoff*: Enviada pela infraestrutura notificando que o usuário saiu do ambiente da infraestrutura. Caso esteja em apresentação, a apresentação é finalizada. Nesta notificação não há exibição mensagem para o usuário.
- *Notificação de Conexão*: Emitida pela infraestrutura notificando que o *Device* e a aplicação estão conectados a mesma. Especialmente na atual implementação da IntuitivePresenter não há exibição mensagem para o usuário. Nesta notificação não há exibição mensagem para o usuário.
- *Notificação de Perda de Conexão*: Enviado pela infraestrutura quando não há conexão com o *Peer* através das todas as tecnologias de comunicação. A aplicação ficará esperando até que o *Device* se conecte novamente a infraestrutura e continue a interação do ponto onde parou.

- **Envio de bytes para o *Device*:** A aplicação IntuitivePresenter envia bytes para o *Device* através do método *write* do módulo *Peer Manager*.

5.3 Configuração da Aplicação

5.3.1 Configuração no *Device*

Para que a infraestrutura possa funcionar no *Smartphone* Google Nexus S é preciso instalá-la no sistema operacional android do aparelho como um Serviço android. Além disso, é preciso garantir ao serviço o direito de usar as seguintes permissões do android:

- android.permission.READ_CONTACTS
- android.permission.ACCESS_NETWORK_STATE
- android.permission.GET_ACCOUNTS
- android.permission.NFC
- android.permission.BLUETOOTH_ADMIN
- android.permission.BLUETOOTH
- android.permission.ACCESS_WIFI_STATE

- android.permission.CHANGE_WIFI_STATE

Além disso, é preciso garantir ao serviço o direito de usar as seguintes permissões do Android:

- android.permission.INTERNET
- android.permission.WRITE_EXTERNAL_STORAGE

A aplicação IntuitivePresenter deve ser instalada no sistema operacional do aparelho necessariamente. É importante ressaltar que as políticas definidas para serem consideradas pela infraestrutura devem ser definidas no momento do desenvolvimento da aplicação, e são são utilizadas pela infraestrutura no momento em que a mesma iniciar a aplicação.

Em referência as medições sucessivas das métricas descritas na Seção 4.4.2, o tempo definido pela a aplicação entre duas medições do desempenho da infraestrutura foi de 10 segundos. A escolha por este valor foi feita com bases empíricas observadas durante o desenvolvimento da aplicação.

Em referência as políticas descritas na Seção 4.4.2, foram definidas as seguintes políticas e valores:

- **Estratégia de Consumo de Bateria:** Modo Econômico;
- **Mínima Largura de Banda Requerida:** 1000 Kbps;
- **Preferência do Usuário:** Wi-Fi.

5.3.2 Configuração no *Peer*

Para que a infraestrutura possa funcionar no *Peer* Dell Inspiron 1525 é preciso ter instalado no sistema operacional o *Java Runtime Environment* (JRE) versão 1.6. Além disso, é preciso ter configurada nas bibliotecas do IntuitivePresenter a biblioteca BlueCove[1] que é a implementação para Java do Bluetooth (especificação JSR-82).

[1] http://bluecove.org/

5.4 Resultados

Nesta seção serão descritos os resultados da aplicação IntuitivePresenter quando posta em funcionamento. Conforme pode ser observado na Figura 5.1a o *Device* é o *smartphone* e o *Peer* tem a sua saída de vídeo ligada ao monitor do fundo da imagem. A infraestrutura é executada por uma instância de um *android service*.

(a) Leitura do cartão através do NFC

(b) Infraestrutura inicia a aplicação IntuitivePresenter

Figura 5.1: Iniciação da infraestrutura e da aplicação IntuitivePresenter pela leitura do cartão através do adaptador NFC

O funcionamento inicia com a aproximação entre *smartphone* o cartão RFID que faz com que o adaptador NFC leia as informações do cartão e inicie a infraestrutura, conforme ilustrado na Figura 5.1a.

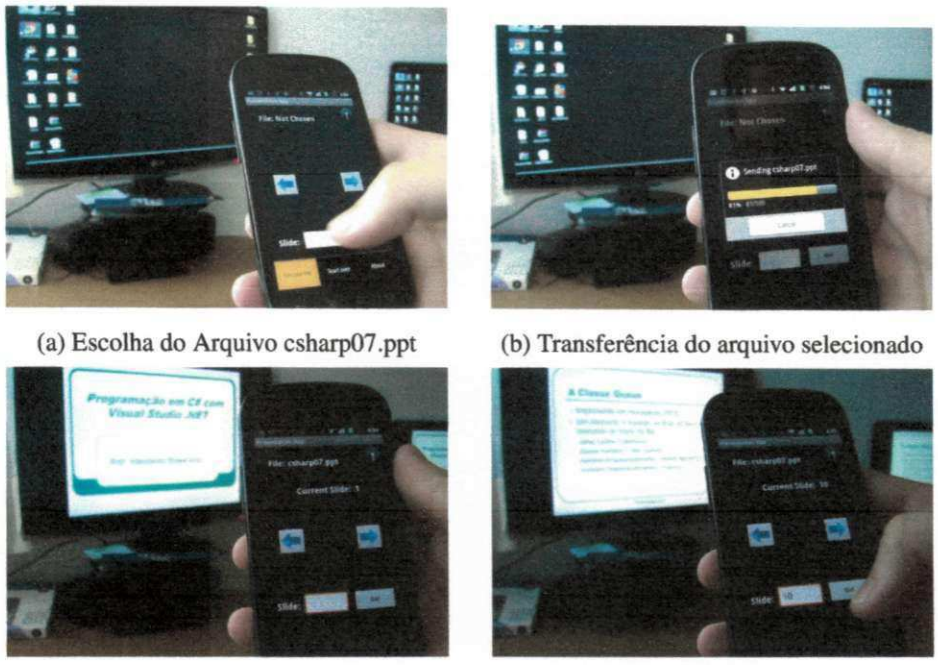
(a) Notificação de conexão via Bluetooth

(b) Notificação de chaveamento para Wi-Fi

Figura 5.2: Notificações de conexão e chaveamento sendo mostradas na aplicação

Após isso, a infraestrutura inicia as conexões referenciadas pelos cartão através da

chamada do método *start()* do módulo *Device Manager*. Após isso, aplicação é iniciada através de um *android Intent* disparado pela infraestrutura, conforme ilustrado na Figura 5.1b. Neste momento da execução as políticas definidas pela aplicação são configuradas na infraestrutura.



(a) Escolha do Arquivo csharp07.ppt

(b) Transferência do arquivo selecionado

(c) Apresentacao do arquivo em formato PPT

(d) Manipulando os slides

Figura 5.3: Escolha, transferência e apresentação de uma apresentação em formato PPT

A partir deste momento a aplicação espera a notificação de autenticação e de conexão da infraestrutura com o *Peer* e através de alguma tecnologia de comunicação. A Figura 5.2a mostra o momento a aplicação é notificada sobre conexão da infraestrutura através do Bluetooth. As notificações são recebidas pela aplicação através de objetos *Message* do pacote *android.os*.

Em um dado instante a infraestrutura decide, através da função de custo, por fazer o chaveamento para o Wi-Fi e a notificação é então mostrada na tela da aplicação, conforme ilustrado na Figura 5.2b.

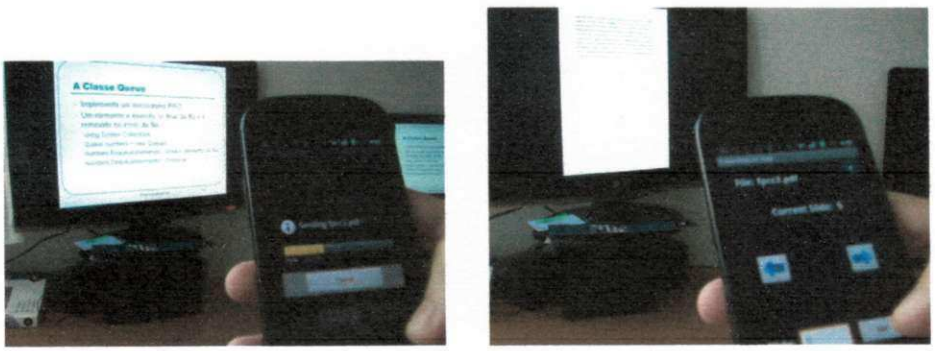
É importante ressaltar que neste ponto não há alteração alguma para a aplicação no que diz respeito a forma de envio de dados para o *Peer*. Para o envio de dados da aplicação o cha-

veamento é transparente, ou seja, a aplicação continua chamando o método *write()* do *Device Manager* independentemente de qual tecnologia está sendo utilizada pela infraestrutura.

O usuário pode escolher uma apresentação em formato PPT que é então enviada e apresentada no *Peer*. Esta sequência é ilustrada nas Figuras 5.3a, 5.3b e 5.3c, respectivamente. Além disso, a aplicação permite manipular passando e voltando os slides da apresentação conforme ilustrado na figura 5.3d.

É importante notar também que para o envio de dados do *Peer* para o *Device* a tecnologia que está sendo utilizada pela infraestrutura e o chaveamento também são transparentes. Ou seja, de mesma maneira do *Device*, a aplicação IntuitivePresenter que executa no *Peer* continua chamando o método *write()* do *Peer Manager* independentemente da tecnologia de comunicação em uso.

A aplicação IntuitivePresenter também suporta apresentações em formato PDF, conforme mostrado nas Figuras 5.4a e 5.4b.



(a) Transferência do arquivo PDF (b) Apresentação e manipulação do PDF

Figura 5.4: IntuitivePresenter suporta apresentações em formato PDF

5.5 Conclusões do Capítulo

Neste capítulo foi apresentado o estudo de caso utilizado como mecanismo de validação da infraestrutura proposta. Esta validação foi feita por meio do desenvolvimento da aplicação IntuitivePresenter que utilizou as funcionalidades da infraestrutura. O objetivo deste estudo de caso foi demonstrar como a infraestrutura pode ser utilizada para abstrair complexida-

des referente a ambientes pervasivos de várias conexões auxiliando o desenvolvimento de aplicações pervasivas cientes de contexto.

Capítulo 6

Considerações Finais

Nos últimos anos, têm ocorrido consideráveis avanços nas tecnologias de comunicação e computação móvel, em especial, os relacionados ao surgimento e a massificação de dispositivos "pós-PC". Este fato tem viabilizado o paradigma da Computação Pervasiva na medida em que tem sido possível presenciar o surgimento de aplicações e cenários do mesmo. Neste contexto, diversas soluções para computação pervasiva têm sido propostas com as mais variadas finalidades.

Apesar das várias soluções com diferentes focos, não havia uma solução que contemplasse junção de tecnologias de curto e longo alcance que fosse construída para ser executada em dispositivos móveis e que fosse aplicado a ambientes pervasivos.

Neste trabalho foi apresentada uma infraestrutura para o desenvolvimento de aplicações pervasivas cientes de contexto com suporte a tecnologias de longo e curto alcance e chaveamento automático entre as mesmas. A infraestrutura desenvolvida é composta por três tecnologias de comunicação: Bluetooth e Wi-Fi para longo alcance e NFC para curto alcance. Ela permite ler cartões RFID com informações sobre as tecnologias de comunicação disponíveis em ambientes pervasivos e se conectar através das mesmas. Além disso a infraestrutura se conecta e faz o chaveamento automático entre estas tecnologias através informações definidas pelas aplicações e medidas em tempo de execução.

Ademais, a infraestrutura fornece uma interface para que aplicações de terceiros possam utilizá-la. Para isso, o desenvolvedor precisa implementar sua solução em um modelo que separa em dois componentes (móvel e fixo), sendo interligados através das tecnologias de comunicação implementadas pela infraestrutura: Bluetooth e Wi-Fi.

Como forma de validação da infraestrutura, foi desenvolvido um estudo de caso: uma aplicação pervasiva chamada IntuitivePresenter que utiliza a infraestrutura e que tem como objetivo prover aos seus usuários uma forma rápida, fácil e intuitiva de configurar, iniciar e apresentar seus slides em conferências e salas de aula.

Observou-se que a infraestrutura atendeu as expectativas no escopo da aplicação na medida em que: (i) abstraiu complexidades relativa a detalhes de comunicação utilizando três tecnologias de comunicação de longo e curto alcance; (ii) proveu um serviço de chaveamento automático entre estas tecnologias baseado em regras e políticas definidas pela aplicação; e (iii) permitiu um desenvolvimento mais rápido da aplicação facilitando o trabalho do programador.

Portanto, tendo em vista o objetivo que foi proposto, a infraestrutura desenvolvida neste trabalho pode ser utilizada de forma satisfatória contribuindo positivamente para o desenvolvimento de aplicações pervasivas.

6.1 Contribuições

As principais contribuições deste trabalho são enumeradas a seguir:

- Especificação de uma infraestrutura que possui capacidade de integrar tecnologias de comunicação de curto e longo alcance e fornecer chaveamento automático.
- Desenvolvimento de uma infraestrutura focada em aplicações para ambientes pervasivos baseado na interação mínima com o usuário.
- Definição de um protocolo de comunicação para ambientes pervasivos de comunicação híbrida com suporte a login, logoff, conexão e chaveamento.
- A partir da utilização da infraestrutura desenvolvida neste trabalho, o processo de desenvolvimento de aplicações pervasivas se tornará menos complexo tendo em vista que os desenvolvedores não serão obrigados a ter que lidar com detalhes já atendidos pela infraestrutura.

6.2 Limitações e Trabalhos Futuros

Existem alguns pontos que não foram o foco no desenvolvimento deste trabalho, mas que necessitam ser considerados em trabalhos futuros. Estes são: a privacidade e a segurança das informações que trafegam nas tecnologias de comunicação (Bluetooth e Wi-Fi) bem como as informações guardadas no cartão RFID.

Dado que estas informações podem ser muito sensíveis ou até críticas dependendo da natureza da aplicação implementada, faz-se necessário implementar políticas, técnicas e mecanismos de segurança que forneçam a segurança, proteção e sigilo dessas informações.

Outra limitação relevante é que a infraestrutura, atualmente, só permite que uma aplicação cadastrada seja executada por vez na infraestrutura.

Para trabalhos futuros, pode-se adicionar um suporte para várias aplicações permitindo a infraestrutura executar mais de uma aplicação por vez. Pretende-se também realizar um estudo e análise sobre como a infraestrutura se comporta em situações de transferência ou *streaming* de vídeo e áudio.

Ademais, pretende-se adicionar a funcionalidade de trocar informações relativas as conexões disponíveis no ambiente entre dois *Devices* NFC a fim de que seja possível compartilhar estes dados apenas aproximando dois aparelhos. Esta funcionalidade certamente abrirá um vasta gama de possibilidades de aplicações em ambientes pervasivos.

Além disso, pode-se realizar um estudo de novas tecnologias e bibliotecas para substituir a biblioteca Bluecove no *Peer* devido ao fato de haver a falta de documentação e, sobretudo, pela mesma não ser mantida com regularidade por seus patrocinadores e colaboradores.

Por fim, pretende-se adicionar ao protocolo utilizado na infraestrutura mecanismos para prover uma transferência confiável ponto a ponto entre o *Device* e o *Peer*. Este ponto se torna crítico principalmente no uso do Bluetooth e sobretudo nos momentos de chaveamento entre o Bluetooth e Wi-Fi. No estágio atual, esta implementação está sendo feita na camada de aplicação e, para futuras aplicações seria de fundamental importância a implementação da mesma no protocolo nativo da infraestrutura.

Bibliografia

[1] P. Agrawal and S. Bhuraria. Near field communication. *IT Matters*, page 67.

[2] I.F. Akyildiz, J. Mcnair, J.S.M. Ho, H. Uzunalioglu, and W. Wang. Mobility management in next-generation wireless systems. *Proceedings of the IEEE*, 87(8):1347–1384, 1999.

[3] Y. Anokwa, G. Borriello, T. Pering, and R. Want. A user interaction model for nfc enabled applications. In *Pervasive Computing and Communications Workshops, 2007. PerCom Workshops' 07. Fifth Annual IEEE International Conference on*, pages 357–361. IEEE, 2007.

[4] Z. Antoniou and D.N. Kalofonos. Nfc-based mobile middleware for intuitive user interaction with security in smart homes. In *Proc. of*, 2006.

[5] Y.S. Chang, Y.S. Hung, C.L. Chang, and T.Y. Juang. Toward a nfc phone-driven context awareness smart environment. In *Ubiquitous, Autonomic and Trusted Computing, 2009. UIC-ATC'09. Symposia and Workshops on*, pages 298–303. IEEE, 2009.

[6] W.D. Chen, K.E. Mayes, Y.H. Lien, and J.H. Chiu. Nfc mobile payment with citizen digital certificate. In *Next Generation Information Technology. ICNIT. The 2nd International Conference on*, pages 120–126. IEEE, 2011.

[7] A.K. Dey. Understanding and using context. *Personal and ubiquitous computing*, 5(1):4–7, 2001.

[8] M.E. Fayad, R.E. Johnson, and D.C. Schmidt. Building application frameworks: object-oriented foundations of framework design. *John Wiley & Sons*, 1999.

[9] K. Henricksen, J. Indulska, and A. Rakotonirainy. Infrastructure for pervasive computing: Challenges. In *Workshop on Pervasive computing INFORMATIK*, volume 1, pages 214–222, 2001.

[10] R. Iglesias, J. Parra, C. Cruces, and N.G. De Segura. Experiencing nfc-based touch for home healthcare. In *Proceedings of the 2nd International Conference on PErvasive Technologies Related to Assistive Environments*, page 27. ACM, 2009.

[11] M. Koskela, J. Ylinen, and P. Loula. A framework for integration of radio frequency identification and rich internet applications. In *Information Technology Interfaces, 2007. ITI 2007. 29th International Conference on*, pages 691–695. IEEE, 2007.

[12] H. Labiod, A. Hossam, and C. De Santis. Wi-fi, bluetooth, zigbee and wimax. *Springer-Verlag GmbH*, 2007.

[13] A. Lahtela, M. Hassinen, and V. Jylha. Rfid and nfc in healthcare: Safety of hospitals medication care. In *Pervasive Computing Technologies for Healthcare, 2008. PervasiveHealth 2008. Second International Conference on*, pages 241–244. IEEE, 2008.

[14] CY Leong, KC Ong, KK Tan, and OP Gan. Near field communication and bluetooth bridge system for mobile commerce. In *Industrial Informatics, 2006 IEEE International Conference on*, pages 50–55. IEEE, 2006.

[15] E. Loureiro, G. Ferreira, H. Almeida, and A. Perkusich. Pervasive computing: What is it anyway? *Ubiquitous and Pervasive Knowledge and Learning Management: Semantics, Social Networking and New Media to Their Full Potential. Hershey, PA, EUA: Idea Group.(Aceito para publicação)*, 2006.

[16] A. Majlesi and B.H. Khalaj. An adaptive fuzzy logic based handoff algorithm for hybrid networks. In *Signal Processing, 2002 6th International Conference on*, volume 2, pages 1223–1228. IEEE, 2002.

[17] M. Miller. *Descobrindo bluetooth*. Campus, 2001.

[18] G.K. Mostefaoui, J. Pasquier-Rocha, and P. Brezillon. Context-aware computing: a guide for the pervasive computing community. In *Pervasive Services, 2004. ICPS 2004. IEEE/ACS International Conference on*, pages 39–48. IEEE, 2004.

[19] J. Ondrus and Y. Pigneur. Near field communication: an assessment for future payment systems. *Information Systems and E-Business Management*, 7(3):347–361, 2009.

[20] E. O'Neill, P. Thompson, S. Garzonis, and A. Warr. Reach out and touch: using nfc and 2d barcodes for service discovery and interaction with mobile devices. *Pervasive Computing*, pages 19–36, 2007.

[21] A. Oulasvirta, T. Rattenbury, L. Ma, and E. Raita. Habits make smartphone use more pervasive. *Personal and Ubiquitous Computing*, 16(1):105–114, 2012.

[22] K. Pahlavan, P. Krishnamurthy, A. Hatami, M. Ylianttila, J.P. Makela, R. Pichna, and J. Vallstron. Handoff in hybrid mobile data networks. *Personal Communications, IEEE*, 7(2):34–47, 2000.

[23] T. Pering, Y. Agarwal, R. Gupta, and R. Want. Coolspots: Reducing the power consumption of wireless mobile devices with multiple radio interfaces. In *Proceedings of the 4th international conference on Mobile systems, applications and services*, pages 220–232. ACM, 2006.

[24] L. Press. Personal computing: the post-pc era. *Communications of the ACM*, 42(10):21–24, 1999.

[25] J. Riekki, T. Salminen, and I. Alakarppa. Requesting pervasive services by touching rfid tags. *Pervasive Computing, IEEE*, 5(1):40–46, 2006.

[26] O. Riva and J. Kangasharju. Challenges and lessons in developing middleware on smart phones. *Computer*, 41(10):23–31, 2008.

[27] D. Saha and A. Mukherjee. Pervasive computing: a paradigm for the 21st century. *Computer*, 36(3):25–31, 2003.

[28] M. Satyanarayanan. Pervasive computing: Vision and challenges. *Personal Communications, IEEE*, 8(4):10–17, 2001.

[29] R. Steffen, J. Preißinger, T. Schollermann, A. Muller, and I. Schnabel. Near field communication (nfc) in an automotive environment. In *Near Field Communication (NFC), 2010 Second International Workshop on*, pages 15–20. IEEE, 2010.

[30] M. Stemm and R.H. Katz. Vertical handoffs in wireless overlay networks. *Mobile Networks and applications*, 3(4):335–350, 1998.

[31] E. Strömmer, M. Hillukkala, and A. Ylisaukko-oja. Ultra-low power sensors with near field communication for mobile applications. *Wireless Sensor and Actor Networks*, pages 131–142, 2007.

[32] E. Strommer, J. Kaartinen, J. Parkka, A. Ylisaukko-oja, and I. Korhonen. Application of near field communication for health monitoring in daily life. In *Engineering in Medicine and Biology Society, 2006. EMBS'06. 28th Annual International Conference of the IEEE*, pages 3246–3249. IEEE, 2006.

[33] N.D. Tripathi. *Generic adaptive handoff algorithms using fuzzy logic and neural networks*. PhD thesis, Virginia Polytechnic Institute and State University, 1997.

[34] N.D. Tripathi, J.H. Reed, and H.F. VanLandinoham. Handoff in cellular systems. *Personal Communications, IEEE*, 5(6):26–37, 1998.

[35] R. Want. Near field communication. *Pervasive Computing, IEEE*, 10(3):4–7, july-september 2011.

[36] M. Weiser. The computer for the 21st century. *Scientific American*, 265(3):94–104, 1991.

[37] M. Weiser and J.S. Brown. The coming age of calm technology [1]. *Xerox Parc*, 8:2007, 1996.

[38] C.H. Wu and C.H. Wu. An ubiquitous data delivery system in hybrid wireless environments. In *P2P, Parallel, Grid, Cloud and Internet Computing (3PGCIC), 2010 International Conference on*, pages 230–234. IEEE, 2010.

[39] F. Zhu and J. McNair. Optimizations for vertical handoff decision algorithms. In *Wireless Communications and Networking Conference, 2004. WCNC. 2004 IEEE*, volume 2, pages 867–872. IEEE, 2004.