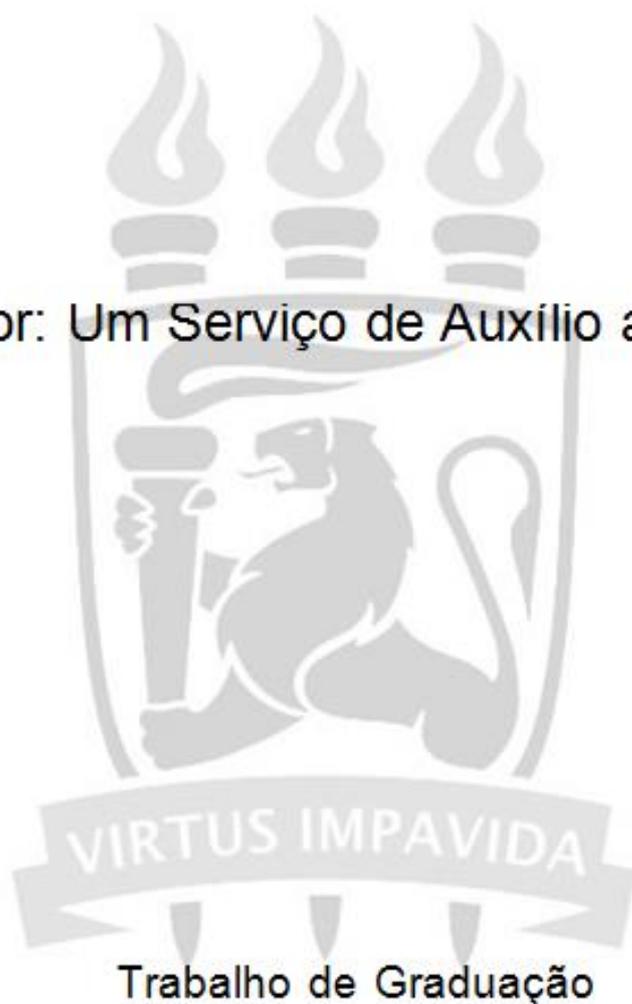


UNIVERSIDADE FEDERAL DE PERNAMBUCO

CENTRO DE INFORMÁTICA

---

TaxiAdvisor: Um Serviço de Auxílio ao Passageiro



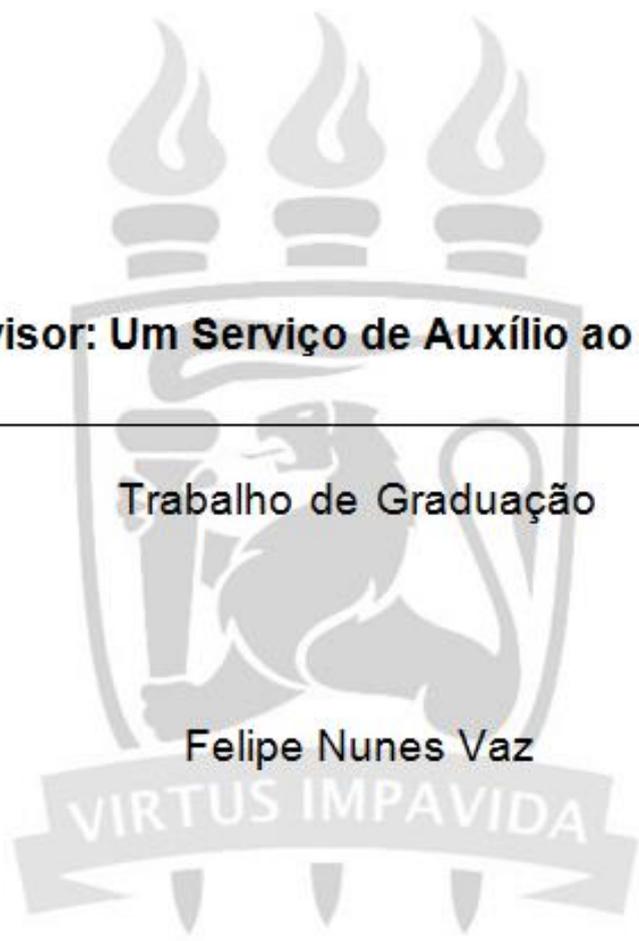
Felipe Nunes Vaz

**Orientador:** Carlos André Guimarães Ferraz

Recife, 2014

**Orientador:** Carlos André Guimarães Ferraz

Recife, 2014



**TaxiAdvisor: Um Serviço de Auxílio ao Passageiro**

---

Trabalho de Graduação

Felipe Nunes Vaz

VIRTUS IMPAVIDA

Projeto de Graduação apresentado no Centro de Informática da Universidade Federal de Pernambuco por Felipe Nunes Vaz, orientado pelo PhD. Carlos André Guimarães Ferraz, como requisito parcial para a obtenção do grau de Bacharel em Ciência da Computação.

## **Agradecimentos**

Agradeço à minha família que sempre me apoiou em todos os momentos, especialmente aos meus pais e avós, que sempre incentivaram meus estudos.

À minha namorada, que esteve ao meu lado durante todo o período de estudos na universidade e me incentivou à fazer o meu melhor.

À todos os meus amigos e colegas que me ajudaram de alguma forma a chegar a este ponto.

À meu orientador, Carlos Ferraz, que além de me guiar durante todo esse processo também fez parte da minha formação ao longo de outras disciplinas no curso.

## **Resumo**

O grande aumento na quantidade de pessoas vivendo na região das cidades está aumentando o problema da mobilidade urbana. Nos maiores centros urbanos transitar já está exigindo dos cidadãos muito tempo, chegando a várias horas por dia.

Nesse contexto estão os Intelligent Transportation Systems, ou ITS. Eles são aplicativos que tem como objetivo prover soluções para problemas na área de transportes. Eles fazem parte da área da computação chamada ubíqua.

A computação ubíqua compreende os sistemas que tem alta mobilidade e tem baixa percepção do usuário, que muitas vezes nem percebem sua existência por completo.

O UbiMid é um middleware que foi desenvolvido com o objetivo de auxiliar o desenvolvimento de aplicações ITS, fornecendo serviços úteis e abstraídos de forma de se torna mais fácil usá-los, entre outras coisas.

O objetivo deste trabalho é desenvolver um ITS, voltado para os passageiros de taxi, e propor um novo serviço a este middleware, que ao ser implementado melhoraria consideravelmente este ITS e outras que possam vir a ser desenvolvidos.

**Palavras-chave: Mobilidade urbana, Intelligent Transportation Systems, computação ubíqua, middleware.**

## **Abstract**

The large increase in the number of people living in the region of the cities is increasing the problem of urban mobility. In major urban centers transit is already requiring citizens much time, reaching several hours a day.

In this context are the Intelligent Transportation Systems, or ITS. They are applications that aim to provide solutions to problems in transportation. They are part of the area called ubiquitous computing.

Ubiquitous computing comprises systems that have high mobility and has low perception of the user, who often do not even realize their existence altogether.

UbiMid is a middleware that was developed with the objective of assisting the development of ITS applications, providing useful services and abstracted in order to make it easy to use them, among other things.

The objective of this work is to develop an ITS, facing the passenger cab, and propose a new service to the middleware, which when implemented would improve this ITS and others that may be developed.

**Key-words: Urban mobility, Intelligent Transportation Systems, Ubiquitous computing, middleware.**

## Sumário

1	Introdução .....	9
2	Fundamentos Teóricos .....	11
2.1	Computação Ubíqua.....	11
2.2	Middleware .....	12
2.3	Contexto .....	13
2.4	Intelligent Transportation Systems .....	13
2.4.1	Exemplos de ITS .....	14
3	Problema da mobilidade urbana .....	18
3.1	Mobilidade urbana no Brasil.....	18
3.2	Alternativas ao transporte individual.....	19
3.2.1	Ônibus.....	19
3.2.2	Bicicleta.....	20
3.2.3	Metrô .....	20
3.2.4	Taxi.....	21
3.3	Considerações Finais .....	21
4	Tecnologias Usadas .....	22
4.1	Android.....	22
4.2	UbiMid .....	23
4.3	Considerações Finais .....	25
5	TaxiAdvisor .....	27
5.1	Visão geral .....	27
5.2	Implementação.....	29
5.3	Sugestão de um novo serviço ao UbiMid .....	31
5.4	Considerações Finais .....	32

6	Conclusão e trabalhos futuros .....	33
7	Referências.....	35

## Lista de Figuras

Figura 1 - Dimensões da Computação Ubíqua .....	12
Figura 2 – Middleware .....	12
Figura 3 - Aplicativo Waze, mostrando a velocidade média das vias.....	15
Figura 4 - Arquitetura do Ubibus .....	15
Figura 5 - Ubibus Route .....	16
Figura 6 - Frota de veículos automotores no Brasil – 2001 e 2012 .....	18
Figura 7 - Deslocamentos feitos pelas pessoas em cidades com população superior 60 mil habitantes .....	19
Figura 8 - Marketshare do mercado de smartphones.....	22
Figura 9 - Abstração da arquitetura do UbiMid provida através de um barramento de serviços .....	23
Figura 10 - Transformador de contexto .....	24
Figura 11 - Enriquecedor de contexto .....	24
Figura 12 - Componente de QoS .....	25
Figura 13 - Tela inicial de login do TaxiAdvisor .....	26
Figura 14 - Sequência para achar rota.....	27
Figura 15 - Listagem de taxistas .....	28
Figura 16 - Funcionamento de uma tarefa assíncrona.....	29

## 1 Introdução

Com o aumento da concentração de pessoas nas grandes cidades, a mobilidade urbana começou a ser encarada como um grande desafio a ser enfrentado pelos governos. Principalmente em lugares onde o transporte individual é a opção mais escolhida pela população, a locomoção diária costuma ser um desafio.

Os Intelligent Transportation Systems surgiram para auxiliar nessa questão e em outras, como segurança nos veículos e monitoramento das vias. Atualmente eles já se tornaram parte da vida do cidadão comum, seja no seu smartphone ou em grandes sistemas implantados nas cidades.

Nesse cenário surgiu o UbiMid, um middleware de integração e sensível ao contexto voltado para aplicações e sistemas inteligentes de transporte. Ele tem como objetivo ajudar no desenvolvimento de aplicações ITS fornecendo serviços úteis gerenciados por componentes com várias funções, entre elas aumentar a disponibilidade destes serviços.

O objetivo deste trabalho é desenvolver um ITS voltado para o setor de taxis, chamado TaxiAdvisor. Este aplicativo irá auxiliar passageiros tanto na hora de chamar um taxi quanto no momento de achar o melhor caminho, para que ele alcance seu destino de maneira rápida e barata.

Após o desenvolvimento deste software, será proposto um novo serviço ao UbiMid, de forma que TaxiAdvisor e outros aplicativos no futuro poderão usufruir de toda estrutura do middleware.

Os próximos capítulos presentes neste trabalho serão dispostos da seguinte forma:

- No capítulo 2 serão apresentados alguns fundamentos teóricos necessários para uma boa compreensão do texto.
- O capítulo 3 abordará mais profundamente o problema da mobilidade urbana, e serão mostradas algumas alternativas.
- No capítulo 4 serão apresentadas as tecnologias utilizadas no processo de desenvolvimento do TaxiAdvisor.
- No capítulo 5 será apresentado o TaxiAdvisor, suas funcionalidades e seu código. Também será feita a proposta de expansão do UbiMid.

- O capítulo 6 concluirá o trabalho discutindo o resultado obtido e possíveis adições futuras ao software móvel.

## 2 Fundamentos Teóricos

Neste capítulo serão apresentados alguns conceitos necessários para o entendimento deste trabalho. Será mostrado o conceito de computação ubíqua, uma das áreas da computação que mais vem sendo estudada nos últimos anos.

Também será apresentado o conceito de *middleware* e alguns requisitos desejáveis em um sistema que se enquadre nesta categoria. Igualmente importante é o conceito de contexto, que será brevemente explicado. Além disso, haverá uma breve explicação sobre *Intelligent Transportation Systems* (ITS) e algumas aplicações.

### 2.1 Computação Ubíqua

A convergência das tecnologias de rádio, dos microprocessadores e dos dispositivos eletrônicos digitais pessoais está levando ao conceito de ubiquidade no qual dispositivos inteligentes, móveis e estacionários, coordenam-se entre si para prover aos usuários acesso imediato e universal a novos serviços, de forma transparente, que visam aumentar capacidades humanas (Araújo, 2003). Para entendermos melhor a computação ubíqua, primeiros precisamos entender dois outros conceitos: computação móvel e computação pervasiva.

Apesar de estes termos serem, muitas vezes, usados como sinônimos existem algumas diferenças entre eles. A computação móvel é uma junção de sistemas distribuídos em dispositivos distintos que se comunicam entre si através de uma rede sem fio, o que permita a mobilidade (Adelstein, 2005).

Computação pervasiva é um termo que se refere a um computador que está embarcado em algo, de forma transparente para o usuário, e este computador tem a capacidade de obter informações do ambiente no qual ele está embarcado e utilizá-la para dinamicamente construir modelos computacionais, ou seja, controlar, configurar e ajustar a aplicação para melhor atender as necessidades do dispositivo ou usuário (Araújo, 2003).

Desta maneira, a computação ubíqua seria a junção das duas, aliando a alta mobilidade da computação móvel com o alto grau de adaptabilidade e embarcamento da computação pervasiva.

	Computação Pervasiva	Computação Móvel	Computação Ubíqua
Mobilidade	Baixa	Alta	Alta
Grau de "embarcamento"	Alto	Baixo	Alta

Figura 1. Dimensões da Computação Ubíqua (Araújo, 2003).

## 2.2 Middleware

Hoje em dia, as grandes corporações têm uma grande variedade de máquinas, que podem rodar diferentes sistemas operacionais. Estes por muitas vezes não apresentam uma compatibilidade, o que dificulta, por exemplo, que um software acesse serviços oferecidos nestas máquinas ao mesmo tempo. Para tentar resolver este problema, entre outros, surgiu o que chamamos de middleware.

O middleware é um serviço que fica entre as plataformas e a aplicação (Figura 2). Usualmente serviços de middleware rodam de maneira distribuída. Ou seja, o *middleware* normalmente inclui a parte cliente e a parte servidor. Pode haver múltiplas implementações de cada parte (Bernstein, 1993).

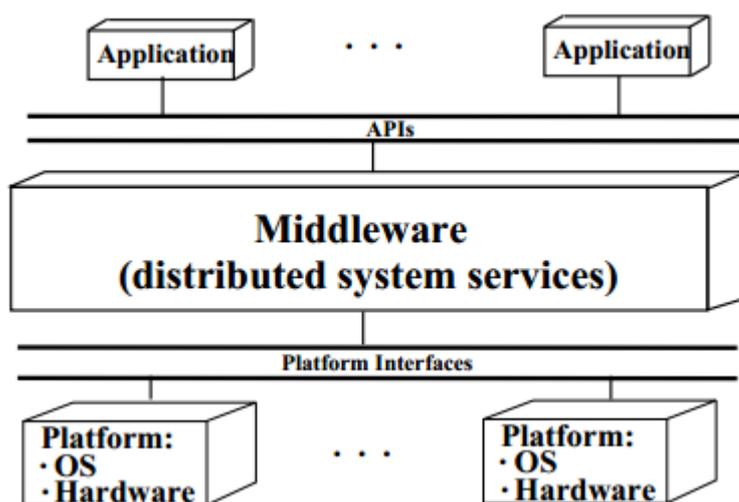


Figura 2. Middleware (Bernstein, 1993).

Segundo de Oliveira Júnior (2014), a implementação de um *middleware* deve levar em consideração alguns requisitos, entre os quais:

- Comunicação: O middleware deve garantir a troca de informações entre os sistemas integrantes de forma transparente, dando a impressão ao usuário que o sistema é único e integrado.
- Escalabilidade: Devido a sua natureza distribuída, um middleware deve aguentar aumentos das requisições feitas pelo usuário. Ele deve ser construído de forma que o custo de integrar mais servidores, inclusive com diferentes plataformas, seja mínimo (Bernstein, 1993).
- Segurança: A internet é cheia de ameaças, por isto o middleware deve ser capaz de prover a segurança necessária para que os dados, tanto entre o cliente e o middleware, quanto entre as diferentes partes do middleware, naveguem da maneira mais segura possível.

### **2.3 Contexto**

Contexto pode significar diversas coisas, dependendo da área. Schilit e Theimer (1994) se referem a contexto como lugar, identidades de pessoas e objetos próximos, e mudanças destes objetos.

Uma boa definição, para o caso que será apresentado, foi dada por Dey (2001):

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

Podemos pegar, como exemplo, uma aplicação que mostra informações do clima. Nesse caso, a localização do usuário, a data e o clima em si podem ser considerados contexto.

### **2.4 Intelligent Transportation Systems**

Sistema de transporte inteligente se refere à aplicação de tecnologias da informação e comunicação para o planejamento e operação de sistemas de transporte (McQueen B. e McQueen L., 1998). Atualmente, esse termo é usado para definir praticamente todas as aplicações que buscam mais segurança ou eficiência na área de transportes.

Segundo Silva (2000), podemos dividir os ITS em algumas categorias, entre elas:

- Sistemas Avançados de Transporte Público (APTS) - visam à melhoria do transporte público. Por exemplo, uma aplicação que fornece ao usuário os tempos de chegadas dos próximos ônibus.
- Sistemas Avançados de Gerenciamento de Tráfego (ATMS): - auxiliam no controle do tráfego. Por exemplo, uma aplicação que muda os temporizadores dos sinais baseado no número de carros em cada via.
- Sistemas Avançados de Informação ao Viajante (ATIS): - fornecem informações sobre a via ao viajante. Por exemplo, uma aplicação que mostra a velocidade média de uma via em determinado momento.
- Sistemas Avançados de Controle Veicular (AVCS) – tentam melhorar a segurança do veículo auxiliando os motoristas. Por exemplo, um sistema que detecta um risco de colisão e freia o carro.

#### **2.4.1 Exemplos de ITS**

Devido ao grande desafio que é, atualmente, o transporte público, o desenvolvimento de sistemas inteligentes de transporte vem sendo incentivado pelos governos. A seguir serão apresentados alguns exemplos de ITS.

##### **2.4.1.1 Waze**

O Waze é um aplicativo para *smartphones* que usa GPS para colher dados do trânsito. Usando esses dados ele, além de mostrar para os usuários dados como velocidade média das vias e acidentes, ele consegue indicar qual é a rota mais rápida para determinado destino (que nem sempre é a mais curta).

Além destes serviços, o Waze conta com mais um diferencial: seu mapa. O mapa do aplicativo é desenhado pelos próprios usuários, que podem dar um nível de detalhe do trânsito local que, de outra forma, seria inviável de obter.

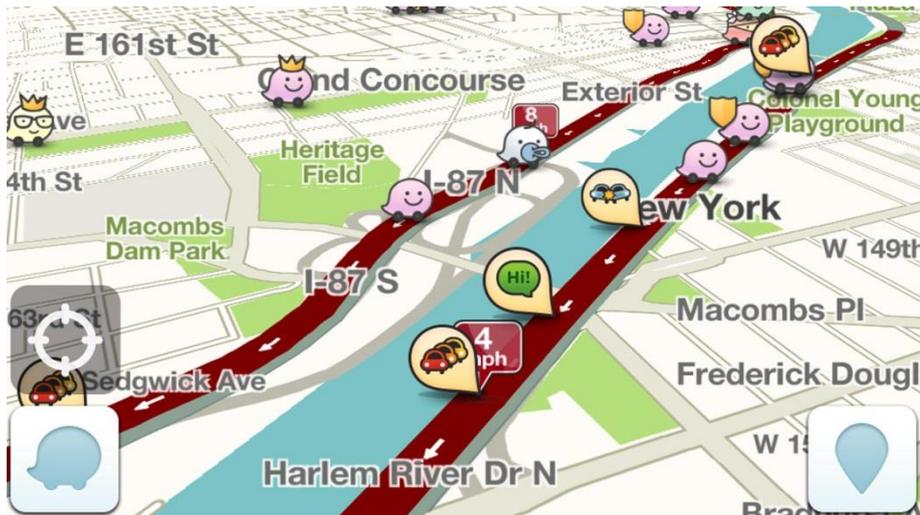


Figura 3. Aplicativo Waze, mostrando a velocidade média das vias.

### 2.4.1.2 Ubibus

O Ubibus é um ITS integrado ubíquo e sensível ao contexto. Ele é um sistema de soluções abertas, com foco no apoio aos passageiros de transporte público, sensível ao contexto (informações dinâmicos, capturadas em tempo real, sobre vias, veículos dispositivos e passageiros), ubíquo e integrado com as redes sociais existentes.



Figura 4. Arquitetura do Ubibus (VIEIRA et al, 2012).

Foram desenvolvidos alguns protótipos de aplicações que fazem uso dos serviços do Ubibus, como o Your City on Time (YCT), que adquire e infere informação de contexto em tempo real e estima o tempo de chegada do ônibus em uma dada parada.

Outra aplicação, a Ubibus Route, extrai informações sobre o transito do Twitter e indica no mapa a melhor rota a seguir, considerando o contexto da rota e do usuário para sugeri-la (Figura 5).



**Figura 5. Ubibus Route (VIEIRA et al, 2012).**

### **2.4.1.3 MASSTR**

Meadowlands Adaptive Signal System for Traffic Reduction é uma Tecnologia Adaptativa de Controle de Sinal (ASCT), ou seja, é uma tecnologia que modifica o tempo dos sinais de trânsito baseado no tráfego ao invés de simplesmente fixar tempos, como é feito no Brasil.

Este projeto irá contar com 128 sinais de trânsito e servir a aproximadamente três milhões de carros por dia. A expectativa é que o projeto consiga reduzir o tempo de viagem destes veículos em mais de um milhão de horas por ano, economizando assim mais de um milhão de galões de gasolina.

#### **2.4.1.4 EasyTaxi**

EasyTaxi é um aplicativo cuja principal função é pedir um táxi. Fundado em abril de 2012, em um sistema de startup, já opera em mais de vinte países e está em constante expansão. Além disso, mais de 200 mil taxistas já estão cadastrados.

Possui um visual bastante simples, com somente um botão: pedir taxi. O fluxo do aplicativo é o seguinte: um usuário pede um taxi; o aplicativo fica responsável por achar o taxista mais perto da localidade; este taxista pode aceitar ou recusar a corrida; se aceitar, o usuário pode acompanhar o motorista em seu trajeto; se rejeitar o aplicativo encontra outro taxista próximo. Segundo seu criador, Tallis Gomes, essa simplicidade é um dos fatores fundamentais para o seu sucesso.

### **2.5 Considerações Finais**

Neste capítulo foram apresentados conceitos necessários para o entendimento deste trabalho. Computação Ubíqua é um termo que engloba sistemas que são ao mesmo tempo bastante móveis e que estão de alguma forma transparentes para o usuário.

Também foi mostrado o que é middleware: um sistema que fica entre as aplicações e plataformas. Ele tem como uma de suas funções prover serviços para aplicações de forma transparente. Outro tópico foi contexto, que foi definido como algo relevante para alguém em determinada situação.

O último e não menos importante tópico deste capítulo foram os ITS. São sistemas que tem como objetivo prover melhorias relacionadas ao transporte. São exemplos de ITS sistemas que aumentam a segurança de veículos e de controle do tráfego urbano.

No próximo capítulo será apresentado o problema da mobilidade urbana, principalmente no Brasil, e algumas alternativas para suavizá-lo.

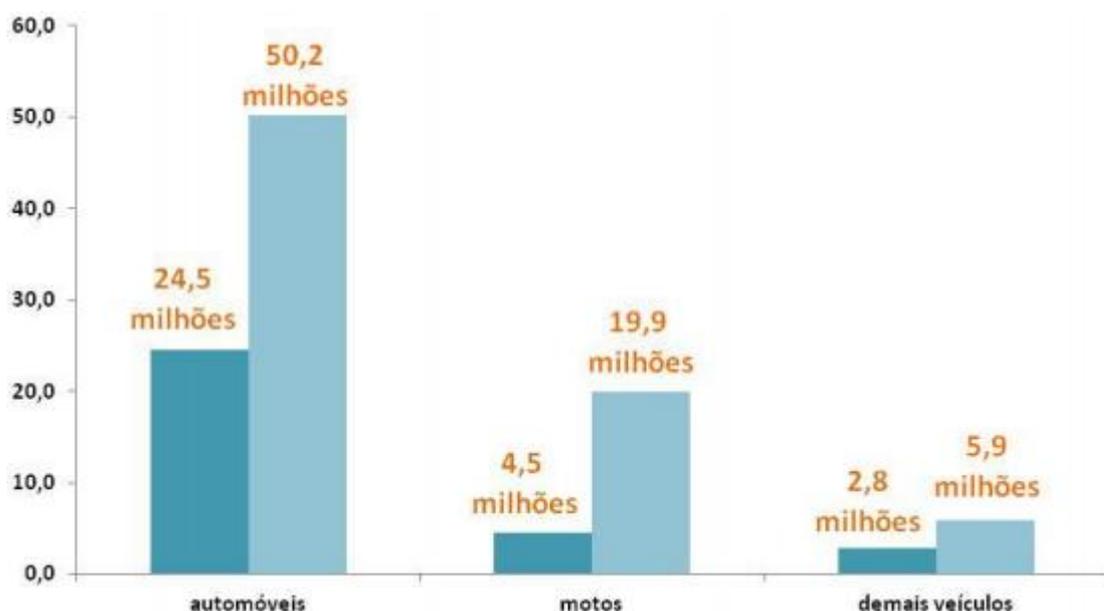
### 3 Problema da mobilidade urbana

Segundo relatório da Organização Mundial da Saúde, em 2010 mais da metade das pessoas viviam em áreas urbanas. Em 2030 serão seis em cada dez, e em 2050 sete em cada dez. E quanto mais pessoas vivendo nas cidades, maior o tamanho destas, e conseqüentemente maior o problema da mobilidade urbana.

#### 3.1 Mobilidade urbana no Brasil

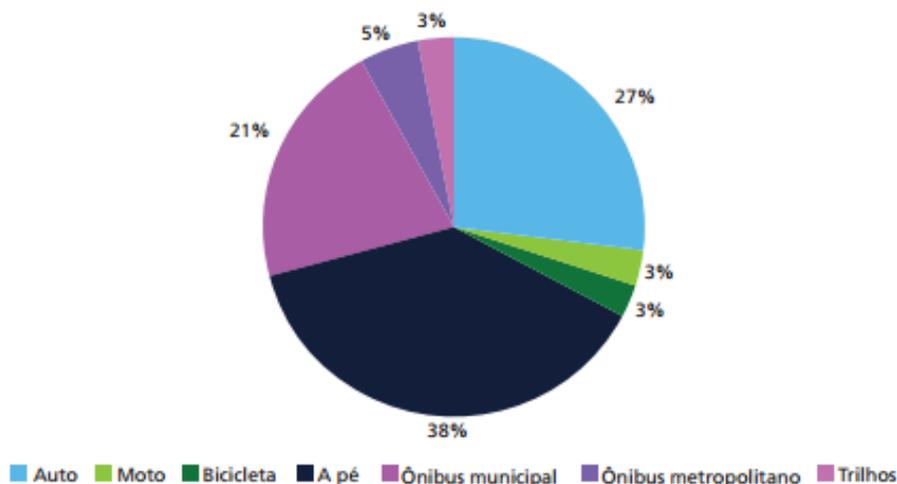
Desde a década de 40, com o governo de Getúlio Vargas, a indústria automobilística tem estado em foco no Brasil, e isto tem um impacto direto na mobilidade urbana brasileira.

Para incentivar a compra de carros, o governo investiu pesado na construção de rodovias. Isto causou um grande aumento na frota de veículos automotivos do país, como mostra a seguir a figura 6.



**Figura 6. Frota de veículos automotores no Brasil – 2001 e 2012 (Elaborado pelo Observatório das Metrôpoles com dados do DENATRAN, 2013).**

Esse aumento causou um problema constante nas principais capitais do país, o engarrafamento. A figura 7 mostra que os brasileiros estão dando prioridade ao transporte individual ao invés do coletivo, principalmente devido às facilidades dadas pelo governo para compra de carros populares e a má qualidade do transporte de massa.



**Figura 7. Deslocamentos feitos pelas pessoas em cidades com população superior 60 mil habitantes (ANTP, 2008).**

### 3.2 Alternativas ao transporte individual

As alternativas para o transporte individual são várias: ônibus, bicicleta, metrô, taxi, barcos. A seguir um detalhe em alguns destes meios de transportes.

#### 3.2.1 Ônibus

Segundo Xavier (2011), os carros geram 7,7 vezes mais poluentes que os ônibus, e ocupam 82% do espaço viário por pessoa, enquanto os ônibus ocupam 18%. Os ônibus tem a grande vantagem de usar a malha viária já existente, o que facilita bastante a sua implantação. Ainda assim, entre as opções a seguir o ônibus é a segunda mais poluente, só perdendo para o taxi.

Uma pesquisa do governo da Escócia, em 2010, listou alguns problemas identificados pela população:

- Comportamento inadequado do motorista

- Comportamento de outros passageiros causando desconforto ou irritação
- Medo das condições dos ônibus os tornarem inseguros, não confiáveis ou inacessíveis (para pessoas com deficiência física), e também receio quanto ao conforto e limpeza.
- Receio quanto à segurança
- Receio quanto aos horários dos ônibus

### **3.2.2 Bicicleta**

Segundo Heinen *et al.* (2012), ao aumentar o número de pessoas que utilizam a bicicleta para se locomover, consegue-se três benefícios principais: diminuição dos congestionamentos, da poluição e melhora da saúde da população.

Para que a população possa aderir a esse meio de transporte, são necessários alguns fatores:

- De acordo com Abraham *et al.* (2002), instalações (como bicicletários, chuveiros e vestiários) no fim do percurso são de extrema importância.
- Ciclovia na cidade. Quando não há estrutura exclusiva para os ciclistas a tendência é que ocorram acidentes, desestimulando novos ciclistas.

### **3.2.3 Metrô**

Para grandes áreas urbanas, o metrô sem dúvida é o meio de transporte mais indispensável. Para se ter uma ideia, o metro de Pequim, o mais movimentado do mundo, somente em um dia transporta cerca de 7,5 milhões de passageiros (China National Radio, 2014). O grande problema desse meio de transporte é o seu alto custo de implementação, já que cada quilômetro de metrô custa, em média, entre 100 e 500 milhões de reais para ser construído.

### **3.2.4 Taxi**

Apesar do taxi não ser um transporte público de massa, ele existe em todas as grandes cidades do mundo. Mesmo sendo um transporte individual, o taxi auxilia na mobilidade urbana significativamente, pois para pessoas que rodam, em média, poucos quilômetros por dia, ele é mais vantajoso do que ter um automóvel próprio.

Além disso, enquanto uma pessoa normalmente só usa seu carro para migração pendular (trajeto casa-trabalho), um taxi pode ser usado por várias pessoas ao longo do dia.

### **3.3 Considerações Finais**

Neste capítulo foi abordado o problema da mobilidade urbana, que está sendo potencializado pelo grande crescimento das cidades nas últimas décadas. No Brasil esse fenômeno já causa diversos problemas, tendo a cidade de São Paulo como grande exemplo de má qualidade neste quesito.

Também foram mostradas algumas alternativas para o transporte individual: o ônibus tem a grande vantagem de utilizar a malha viária já existente; o taxi tem seu trunfo na comodidade; a bicicleta e o metrô são os transportes ideais para pequenas e grandes distâncias, respectivamente.

No próximo capítulo serão apresentadas as principais tecnologias utilizadas no desenvolvimento deste projeto, o sistema operacional móvel Android e o UbiMid, um middleware voltado para o setor de transportes.

## 4 Tecnologias Usadas

Neste capítulo serão apresentadas as tecnologias principais usadas no projeto TaxiAdvisor. Por ser um aplicativo móvel, a plataforma android foi escolhida pois é o sistema operacional para smartphones mais usado no mundo. Também será apresentado o UbiMid, um middleware de integração e sensível ao contexto.

### 4.1 Android

Android Inc. foi criada, segundo Andy Rubin, um de seus fundadores, para desenvolver “dispositivos móveis mais inteligentes que são mais cientes da localização e preferências do seu dono” (Krajci e Darren, 2013).

Em 2005, a empresa foi comprada pelo Google, mas na época não ficou claro as intenções da gigante das pesquisas. A primeira versão beta e pública foi lançada somente em 2007, e a partir de então vem ganhando atualizações pelo menos uma vez por ano, sendo a última versão a 4.4.4 KitKat.

Android lidera o mercado de OS para smartphones desde 2011, e vem aumentando a sua liderança, conforme a figura 8:

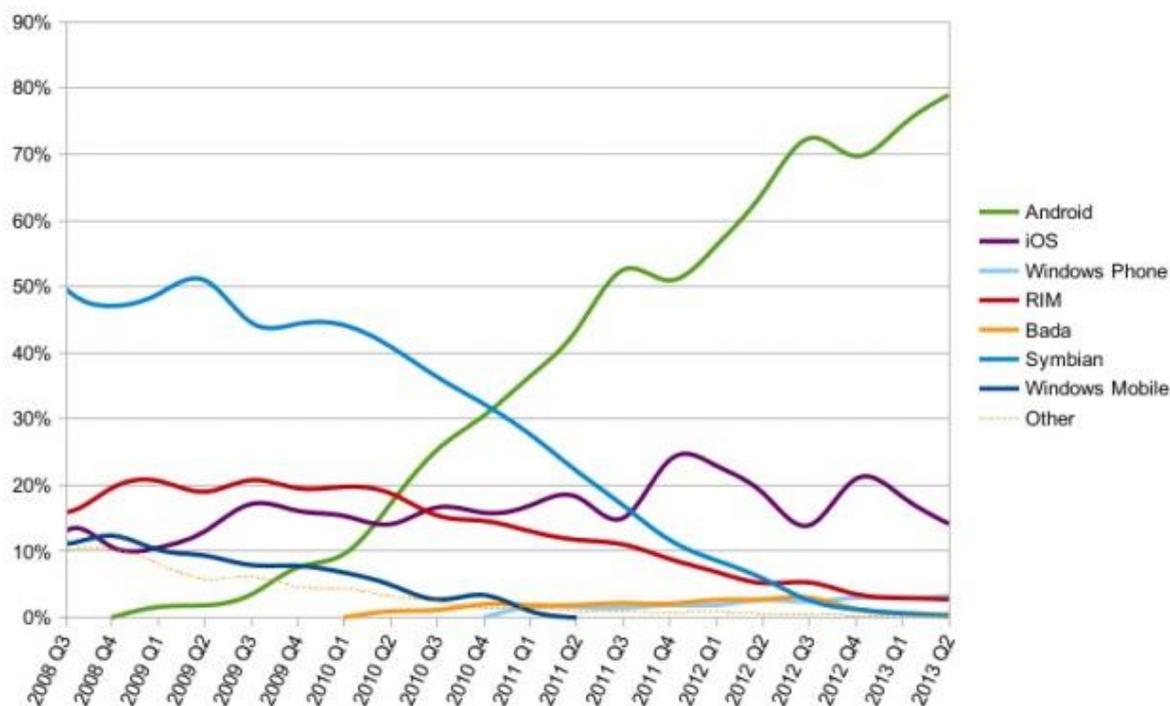


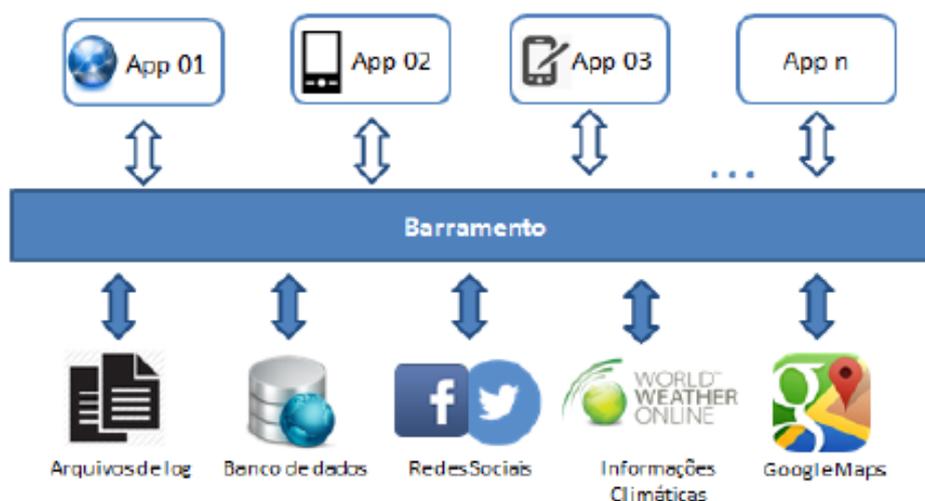
Figura 8. Marketshare do mercado de smartphones (Arstechnica, 2013).

A plataforma Android foi escolhida para o desenvolvimento do aplicativo TaxiAdvisor devido a sua grande gama de aparelhos compatíveis, e a fácil adaptação para seu desenvolvimento, já que usa Java, que é amplamente difundida no mercado, como sua principal linguagem de programação.

Para facilitar o entendimento do código do TaxiAdvisor, é importante entender alguns conceitos básicos de Android. Cada tela é composta por uma activity, e opcionalmente por um xml de layout. Esse xml é opcional pois pode ser substituído por um layout criado programaticamente pela activity.

## 4.2 UbiMid

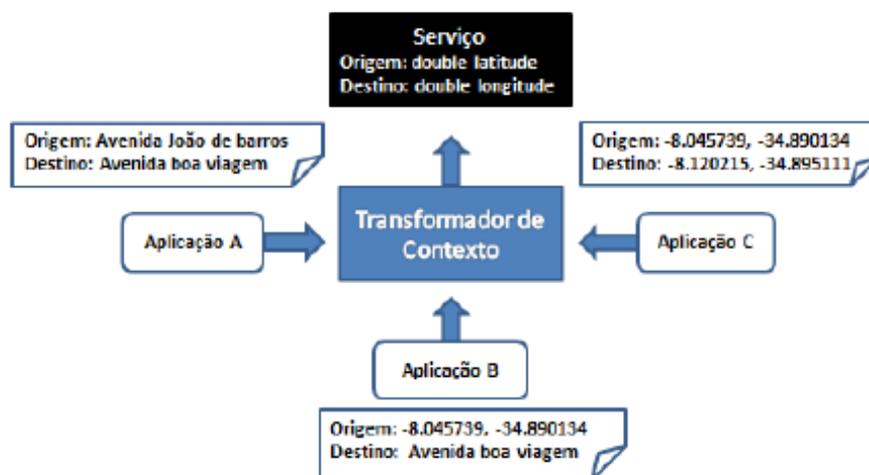
O UbiMid é um middleware sensível ao contexto voltado para aplicações na área de sistemas de transporte público e urbano. Segundo Oliveira Júnior (2014), ele foi desenvolvido para facilitar a comunicação e a coordenação de componentes de software distribuídos através de um barramento de serviços (Figura 9), cujo objetivo é prover modularidade, desacoplamento de sistemas, reuso de código (evitando retrabalho), inferência de contexto lógico, aquisição, armazenamento, enriquecimento, transformação e compartilhamento de informações contextuais entre os diferentes tipos de aplicações e sistemas.



**Figura 9. Abstração da arquitetura do UbiMid provida através de um barramento de serviços (Oliveira Júnior, 2014).**

O UbiMid é composto por vários componentes, alguns deles descritos a seguir:

- Roteador de requisições: é responsável por receber a mensagem e, através de uma análise contextual do conteúdo dela, escolher o destino apropriado.
- Transformador de contexto: é responsável por padronizar a mensagem que será enviada ao serviço final. Na figura 10 é possível ver um exemplo, no qual três mensagens distintas são convertidas para o formato correto do serviço.



**Figura 10. Transformador de contexto (Oliveira Júnior, 2014).**

- Enriquecedor de contexto: é responsável por inferir informações a partir da mensagem do usuário, nos casos em que o serviço precisa de mais informações do que aquelas fornecidas ao UbiMid. Na figura 11 é possível ver um exemplo onde, a partir do CEP fornecido pelo usuário, o enriquecedor de contexto infere várias outras informações.

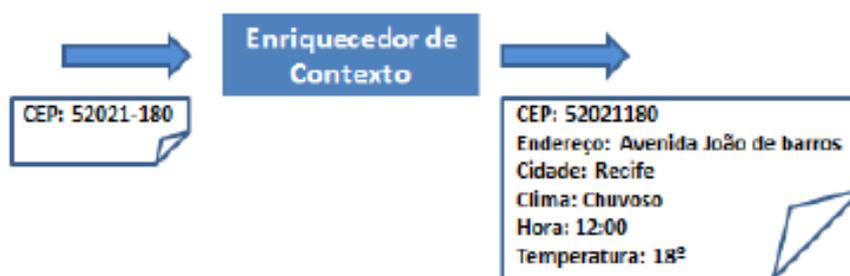


Figura 11. Enriquecedor de contexto (Oliveira Júnior, 2014).

- Componente de QoS: é responsável por identificar qual dos serviços disponíveis, que fornecem o que foi pedido pelo usuário, apresenta o menor tempo de resposta. Após isso, repassa a informação ao roteador para que ele repasse as requisições para o serviço escolhido.

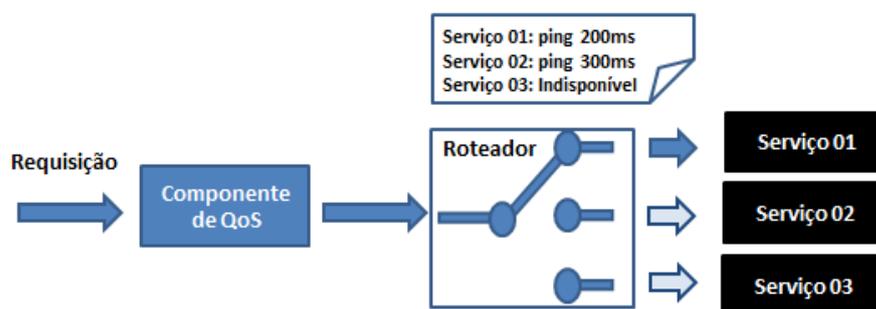


Figura 12. Componente de QoS (Oliveira Júnior, 2014).

### 4.3 Considerações Finais

Neste capítulo foi dada uma breve explicação sobre o sistema operacional móvel Android. Esse OS, que é líder de mercado neste segmento, foi o escolhido para o desenvolvimento do TaxiAdvisor principalmente pela baixa curva de aprendizado e por estar presente em uma variedade grande de aparelhos lançados no mercado.

O UbiMid é um middleware que tem como proposta auxiliar o desenvolvimento de aplicações ITS, provendo serviços úteis gerenciados por componentes que melhoram o desempenho e confiabilidade da aplicação.

No próximo capítulo será apresentado o TaxiAdvisor, aplicativo para a plataforma móvel Android que tem como principal objetivo auxiliar passageiros de taxi na hora de uma corrida.

## 5 TaxiAdvisor

TaxiAdvisor é uma aplicação ITS criada com o intuito de ajudar os passageiros de taxi. Com uma interface bastante simples, ele foi criado de maneira que mesmo um usuário inexperiente não tenha problemas em usá-lo.

Neste capítulo será apresentada a visão geral do aplicativo, explicando suas funções através da análise das telas, também será mostrado como estão implementadas estas funções e por fim será explicado como um novo serviço no UbiMid poderia facilitar e melhorar a implementação do aplicativo.

### 5.1 Visão geral

Com o objetivo de atingir a maior parte da população, a tecnologia escolhida foi Android, e a versão foi a mais antiga compatível com o Google Maps, 2.2 Froyo.

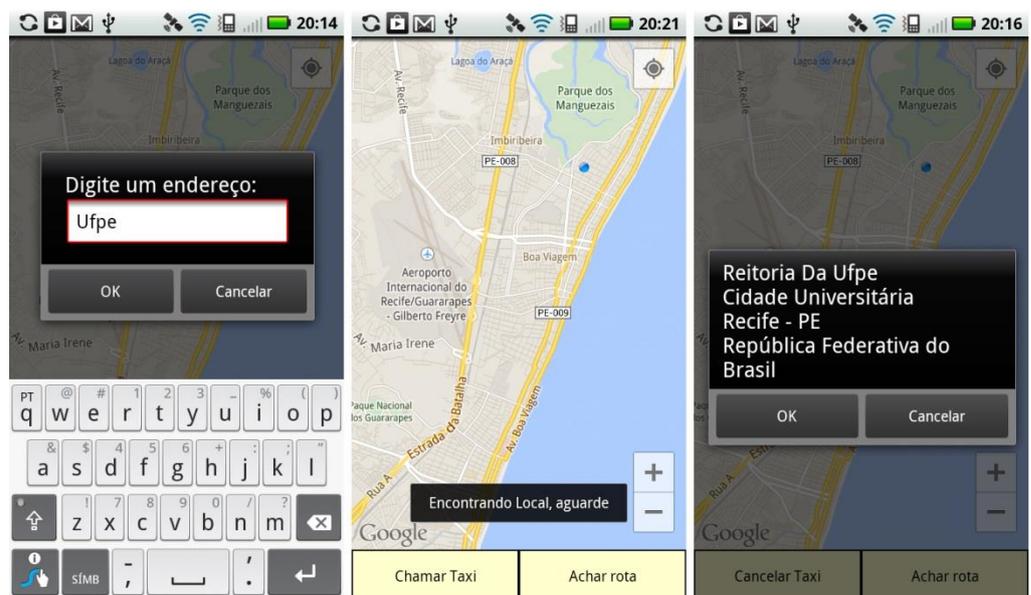
Para o propósito deste trabalho, somente a aplicação que será utilizada pelo cliente foi desenvolvida. Ela conta com três telas:

- Login: é a tela inicial e também a responsável por fazer a autenticação do usuário. Atualmente, devido a falta de um servidor, esta tela representa uma função meramente ilustrativa. É composta pelos campos de texto Nome e Senha, pelo botão Login e pelo logo do TaxiAdvisor.



Figura 13. Tela inicial de login do TaxiAdvisor.

- Mapa: é a tela principal, composta pelo mapa, e por dois botões. Chamar Taxi/Cancelar Taxi e Achar rota. O primeiro é responsável por levar o usuário para a tela de Listagem de taxistas, que será discutida adiante. O segundo botão é responsável pela funcionalidade do TaxiAdvisor de encontrar rotas. Ele abre uma caixa de diálogo onde o usuário deverá digitar o endereço de destino, e pressionar o botão Ok. Após isso uma mensagem será exibida, indicando ao usuário que o aplicativo está buscando o endereço. Quando o endereço é encontrado, uma caixa de diálogo com informações do endereço encontrado é exibida, então o usuário deverá confirmar se o endereço exibido é o correto ou não, caso seja, a rota será exibida no mapa.



**Figura 14. Sequência para achar rota.**

- Listagem de taxistas: é responsável por listar os taxistas disponíveis. Cada taxista tem seu nome informado, sua distância do usuário (dependendo da velocidade de conexão da internet, esse valor pode ser substituído por “carregando...” até que todas as operações necessárias tenham sido realizadas) e seus distintivos.



**Figura 15. Listagem de taxistas.**

O sistema de distintivos é uma maneira de o usuário avaliar os taxistas. No final de cada corrida, será gerado um código para o cliente. Através dele será possível acessar o site do TaxiAdvisor e avaliar o taxista.

O objetivo destes distintivos é fornecer ao cliente informações detalhadas e relevantes a respeito do taxista, para que ele tente achar um que se encaixe no perfil desejado. Afinal, características como dirigir rápido podem ser positivas ou negativas, ficará a cargo do usuário julgar.

## 5.2 Implementação

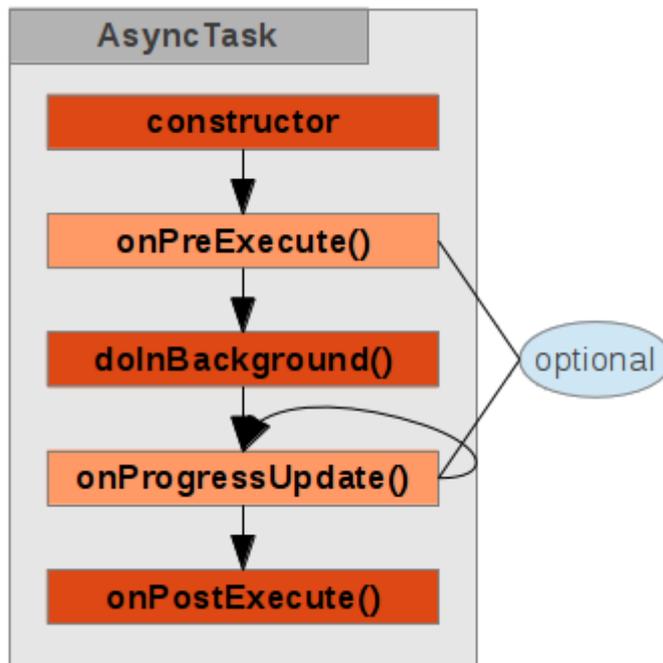
TaxiAdvisor contém aproximadamente 2500 linhas de código, divididas em 52 classes, sendo 3 delas activities. A aplicação necessita de acesso à internet e ao GPS do aparelho.

A activity inicial é a de login, que, como dito anteriormente, devido à falta de um servidor de autenticação está permitindo qualquer usuário utilizar a aplicação.

Na activity do mapa é onde ocorre a maior parte da lógica da aplicação. Inicialmente ela é responsável por habilitar algumas funções no mapa, como o botão de localização do usuário. Esse mapa é fornecido pelo Google Service, uma API do Google fornecida para desenvolvimento em Android a partir da versão 2.2.

A funcionalidade de achar rotas também é responsabilidade dela. Após o usuário digitar o endereço, uma tarefa assíncrona é criada, e ela é responsável por

se comunicar com o serviço do Google e buscar o endereço completo. Esta tarefa deve ser assíncrona, pois em uma requisição na internet não há garantias quanto ao tempo de resposta. Se essa tarefa fosse síncrona ela poderia travar a interface, dando ao usuário uma sensação de o sistema não estaria respondendo, o que é uma prática repudiada por desenvolvedores.



**Figura 16. Funcionamento de uma tarefa assíncrona (Open Sourced, 2013).**

No primeiro momento, a tarefa usa a classe fornecida pela api do GoogleMaps chamada de Geocode. Ela é responsável por fazer a geocodificação, ou seja, encontrar o ponto geográfico a partir de um endereço.

Após a confirmação do endereço pelo usuário, um outro serviço é chamado, o GoogleDirections. Esse serviço recebe os pontos geográficos de início e fim da rota, além de alguns parâmetros de configuração como meio de transporte (bicicleta, a pé, dirigindo), e retorna um XML contendo os passos da rota. Cada passo contém:

- Ponto de início: é composto por latitude e longitude.
- Ponto de término: é composta por latitude e longitude.
- Polyline: é um recurso utilizado para desenhar a rota no mapa, de forma suavizada.

- Duração: a duração que o usuário levará para percorrer aquele passo. Não é cumulativo, ou seja, é o tempo somente daquele passo, não sendo esse tempo somado ao dos próximos passos.
- Instrução: uma instrução no formato html para ser apresentada ao usuário. Por exemplo, “Siga na direção sudeste na Rua Tenente João Cícero em direção à Rua Ministro Nelson Hungria”.
- Distância: a distância daquele passo. Esse valor poderá ser no sistema métrico ou imperial, a escolha deverá ser feita na hora da requisição. O mesmo princípio da duração se aplica aqui, o valor é somente deste passo.

Outra funcionalidade importante executada nesta activity é a de coletar a localização do usuário. Ela é, em condições ideais (local aberto, que facilita o uso do GPS), atualizada a cada 5 segundos e somente a primeira vez que ela é encontrada uma mensagem é mostrada ao usuário. Essa localização é necessária para marcar o início das rotas.

A activity de listagem de taxistas é a única que não tem um XML de layout correspondente, devido aos distintivos, pois . Isso se apresentou como um desafio inicialmente, mas a técnica de construir o layout programaticamente supriu a necessidade.

Construir um layout programaticamente significa fazê-lo em tempo de execução do programa, e não em tempo de compilação. O ponto negativo é que o IDE (ambiente de desenvolvimento integrado, traduzido do inglês integrated development environment) é impossibilitado de exibir uma prévia da tela, dificultando na hora do desenvolvimento.

Esta activity também é responsável por chamar a API GoogleDirections e exibir para o usuário a distância entre os taxistas e o usuário. Esta tarefa também é executada assincronamente.

### **5.3 Sugestão de um novo serviço ao UbiMid**

Como já foi dito anteriormente, o UbiMid possui características que facilitam o desenvolvimento de aplicações ITS. Um novo serviço de rotas para carros poderia

melhorar TaxiAdvisor, pois este faria uso de todos os benefícios que o middleware pode oferecer.

Além de tornar a aplicação mais simples, pois parte do código iria ser substituída somente por uma chamada ao UbiMid, também pouparia tempo de processamento no dispositivo, o que ajudaria a economizar a bateria.

O analisador de QoS iria garantir que o serviço final chamado é o melhor possível, em contrapartida da implementação atual que faz uso exclusivo de somente um serviço independente de suas condições.

O enriquecedor de conteúdo poderia inferir, em caso de múltiplos destinos com um mesmo nome, qual deles o usuário está buscando. Por exemplo, um usuário na cidade do Recife, que digita “Boa Viagem” como destino, provavelmente está se referindo ao bairro que fica localizado na mesma cidade, e não a cidade de Boa Viagem no estado do Ceará.

#### **5.4 Considerações Finais**

Neste capítulo foi apresentado o TaxiAdvisor, que tem como função auxiliar passageiros de taxi, provendo uma maneira simples de chamar um taxista e com um sistema de cálculo de melhor rota. Foi mostrada uma visão geral do aplicativo, mostrando telas e seus comportamentos, e também um pedaço da sua implementação.

Por fim foi sugerido um novo serviço ao UbiMid, que poderia ser usado pelo TaxiAdvisor para melhorar significativamente seu desempenho e manutenção de código, que seria reduzido já que parte das funcionalidades que hoje estão no aplicativo iriam para o middleware.

## 6 Conclusão e trabalhos futuros

Do capítulo de fundamentos teóricos foi possível extrair as informações necessárias para o entendimento de alguns conceitos apresentados durante este trabalho, como computação ubíqua, que trata da computação que está em volta do usuário, muitas vezes sem sequer ser percebida. Também foi apresentado o conceito de middleware, que é um serviço que fica entre as plataformas e as aplicações, o de contexto, que pode significar várias coisas dependendo da situação, e o de ITS.

Foram demonstrados alguns exemplos de Sistemas Inteligentes de Transporte (ou ITS, na sua sigla em inglês), e dessa pequena amostra pode-se concluir que eles poderão prover várias soluções para os problemas no transporte, principalmente urbano, que são enfrentados em todo o mundo.

O capítulo sobre o problema da mobilidade urbana mostrou que, no Brasil, o principal problema é a grande utilização do transporte privado, ou seja, carros e motos. E que a solução para esse problema se encontra na diversificação, pois se a população não começar a utilizar com maior frequência o transporte público os índices de engarrafamento irão aumentar.

Android foi o sistema escolhido para o desenvolvimento devido a sua grande parcela no mercado atual de dispositivos móveis (é a plataforma líder), e por usar Java como base, o que diminuiu bastante a curva de aprendizado.

O UbiMid, middleware sensível ao contexto voltado para aplicações de ITS, teve sua arquitetura resumidamente explicada, com foco na explicação de seus módulos.

Também foi mostrado o TaxiAdvisor, aplicativo desenvolvido para auxiliar os passageiros de taxi, facilitando na hora de pedir um taxi. Também poderá ajudar o passageiro a garantir que o taxista está indo pela rota correta, sem fazer desvios desnecessários.

Foi proposto um novo serviço ao UbiMid, o de rotas para carros, o que possibilitaria que o TaxiAdvisor fizesse uso dos recursos do middleware. Esse uso iria, além de facilitar o desenvolvimento, melhorar seu desempenho e consequentemente reduzir o gasto no consumo de energia. Também aumentaria a

disponibilidade do serviço de rotas, pois o middleware ficaria responsável por enviar a requisição para o serviço disponível.

O aplicativo ainda pode sofrer algumas melhorias, principalmente na questão de economicidade da bateria, devido ao alto uso do GPS. Quanto a novas funcionalidades, poderia ser desenvolvida uma função de agendamento de corrida, onde o usuário poderia agendar com um taxista, o que beneficiaria os dois, pois o cliente teria um maior controle sobre seus horários e o motorista já iria se dirigir para localidade na hora certa, evitando desperdício de combustível.

## 7 Referências

WORLD HEALTH ORGANIZATION. Global Health Observatory. Urban Population Growth. Disponível em:

<[http://www.who.int/gho/urban\\_health/situation\\_trends/urban\\_population\\_growth\\_text/en/](http://www.who.int/gho/urban_health/situation_trends/urban_population_growth_text/en/)>. Acesso em: 20 jul. 2014.

INSTITUTO DE PESQUISA ECONÔMICA APLICADA. Ricos e Pobres perdem cada vez mais tempo no trânsito. Disponível em:

<[http://www.ipea.gov.br/portal/index.php?option=com\\_content&view=article&id=17212](http://www.ipea.gov.br/portal/index.php?option=com_content&view=article&id=17212)>. Acesso em: 20 jul. 2014.

XAVIER, Giselle. O desenvolvimento e a inserção da bicicleta na política de mobilidade urbana brasileira, 2011.

GOVERNO DA ESCÓCIA. Publicações. Understanding Why Some People Do Not Use Buses. Disponível em:

<<http://www.scotland.gov.uk/Publications/2010/04/23115458/2>>. Acesso em: 20 jul. 2014.

HEINEN, Eva; MAAT, Kees; VAN WEE, Bert: The effect of work-related factors on the bicycle commute mode choice in the Netherlands, 2012.

ABRAHAM, J.E.; MCMILLAN, S.; BROWNLEE, A.T.; HUNT, J.D.: Investigation of Cycling Sensitivities. Transportation Research Board, Washington DC, 2002.

CHINA NATIONAL RADIO. Metrô de Pequim transporta mais de 3,2 milhões de passageiros, um aumento de quase 30%. Notícias. Disponível em:

<[http://news.cnr.cn/native/gd/201401/t20140102\\_514555471.shtml](http://news.cnr.cn/native/gd/201401/t20140102_514555471.shtml)>. Acesso em: 20 jul. 2014.

DE ARAÚJO, Regina Borges. Computação Ubíqua: Princípios, Tecnologias e Desafios. Simpósio Brasileiro de Redes de Computadores, XXI. 2003.

ADELSTEIN, F. et al. Fundamentals of Mobile and Pervasive Computing, 2005.

BERNSTEIN, Philip A. Middleware, An Architecture for Distributed System Services, 1993.

DE OLIVEIRA JÚNIOR, Gilson Medeiros. UbiMid: Um middleware de integração e sensível ao contexto voltado para aplicações e sistemas inteligentes de transportes. Dissertação de Mestrado – Universidade Federal de Pernambuco, Recife, 2014.

SILVA, Danyela Moraes da. Sistemas inteligentes no transporte público coletivo por ônibus. Dissertação de Mestrado – Universidade Federal do Rio Grande do Sul, Porto Alegre, 2000.

MCQUEEN, Judy; MCQUEEN, Bob. Intelligent Transportation System Architectures, 1998.

MASSTR, Meadowlands Adaptive Signal System for Traffic Reduction. Disponível em: <<http://masstr.njmeadowlands.gov/>>. Acesso em: 6 ago. 2014.

WAZE. Disponível em: < [www.waze.com](http://www.waze.com)>. Acesso em: 6 ago. 2014.

Ubibus, An inteligente transportation system. Disponível em: < <http://www.cin.ufpe.br/~ubibus/>>. Acesso em: 7 ago. 2014.

VIEIRA, V. et al. The UbiBus Project: Using Context and Ubiquitous Computing to build Advanced Public Transportation Systems to Support Bus Passengers. Anais do VIII Simpósio Brasileiro de Sistemas de Informação, São Paulo, 2012.

DEY, Anind K. Understanding and Using Context, Georgia Institute of Technology, Atlanta, 2001.

SCHILIT B; THEIMER M. Disseminating active map information to mobile hosts, 1994.

KRAJCI, Iggy; CUMMINGS, Darren. Android on x86, An Introduction to Optimizing for Intel Architecture. Capítulo 1: History and Evolution of the Android OS, 2013.

ARSTECHNICA. Gear & Gadgets. Product News & Reviews. Disponível em: <http://arstechnica.com/gadgets/2013/10/googles-iron-grip-on-android-controlling-open-source-by-any-means-necessary/>. Acesso em: 9 ago. 2014.