

UNIVERSIDADE ESTADUAL PAULISTA

Faculdade de Ciências - Bauru

Bacharelado em Ciência da Computação

Pedro Rogério Cavalca Moreira

Thiago Neves Fabre

Gerenciador de Estudos para Dispositivos Móveis

UNESP

2012

Pedro Rogério Cavalca Moreira

Thiago Neves Fabre

Gerenciador de Estudos para Dispositivos Móveis

Orientador: Eduardo Martins Morgado

Monografia apresentada junto à disciplina Projeto e Implementação de Sistemas II, do curso de Bacharelado em Ciência da Computação, Faculdade de Ciências, Unesp, campus de Bauru, como parte do Trabalho de Conclusão de Curso.

UNESP

2012

Pedro Rogério Cavalca Moreira

Thiago Neves Fabre

Gerenciador de Estudos para Dispositivos Móveis

Monografia apresentada junto à disciplina Projeto e Implementação de Sistemas II, do curso de Bacharelado em Ciência da Computação, Faculdade de Ciências, Unesp, campus de Bauru, como parte do Trabalho de Conclusão de Curso.

BANCA EXAMINADORA

Prof. Dr. Eduardo Martins Morgado
Professor Doutor
DCo – FC - UNESP – Bauru
Orientador

Profª. Drª. Simone das Graças
Domingues Prado
Professora Doutora
UNESP - Bauru

Prof. Dr. Marcos Antonio Cavenaghi
Professor Doutor
UNESP - BAURU

Bauru, 29 de Outubro de 2012.

Resumo

O projeto é uma plataforma para a educação que provê uma maneira onde os professores possam criar e compartilhar conteúdos educacionais para suas aulas, e os alunos possam baixar esse conteúdo para consultá-lo a qualquer momento.

A solução é baseada em dois aplicativos para Windows Phone 7: Kookaburra Teachers, onde os professores podem adicionar o conteúdo de suas aulas; e Kookaburra Students, onde os estudantes podem baixar e visualizar esse conteúdo. Para realizar a comunicação entre esses dois aplicativos, usamos um servidor Windows Azure, que também é responsável por armazenar e processar todo o conteúdo que é gerado durante as aulas.

Palavras-chave: Mobilidade, Educação, Compartilhamento, Smartphones.

Abstract

The project is a platform for education that provides a way where teachers can create and share educational content for their classes, and students can download that content to refer to it at any time.

The solution is based on two Windows Phone 7 Apps: Kookaburra Teachers, where teachers can add content to their classes, and Kookaburra Students, where students can download and view the content of classroom. To realize the communication between these two applications, we use a Windows Azure server, that server is also responsible for processing some conversions, enabling real-time translation. In addition to storing all the content that is generated during the lessons.

Keywords: Mobility, Education, Sharing, Smartphones

Lista de Figuras

Figura 1 – Visão Geral da Computação em Nuvem	16
Figura 2 – Arquitetura da Solução.....	27
Figura 3 – Diagrama de Tabelas	29
Figura 4 – Diagrama de Classes da Base de Dados	30
Figura 5 – Visão Geral da Arquitetura de Comunicação do Sistema.....	34
Figura 6 – Diagrama de Classe do Aplicativo Kookaburra Student	35
Figura 7 – Diagrama de Classe do Aplicativo Kookaburra Teacher	35

Sumário

1. INTRODUÇÃO	8
1.1. PROBLEMAS.....	9
1.2. JUSTIFICATIVA	10
1.3. SOLUÇÃO	11
1.4. OJETIVOS GERAIS E ESPECÍFICOS DO PROJETO.....	11
2. FUNDAMENTAÇÃO TEÓRICA.....	13
2.1. TECNOLOGIA DA INFORMAÇÃO E COMUNICAÇÃO (TIC) NAS ESCOLAS	13
2.2. EDUCAÇÃO E TECNOLOGIA NO BRASIL	13
2.3. TENDÊNCIAS TECNOLÓGICAS.....	14
2.4. SMARTPHONES.....	14
2.4.1. <i>WINDOWS PHONE 7</i>	14
2.5. COMPUTAÇÃO EM NUVEM	15
2.5.1. INTRODUÇÃO	15
2.5.2. ARQUITETURA.....	16
2.5.3. CARACTERÍSTICAS ESSENCIAIS.....	18
2.5.4. <i>WINDOWS AZURE</i>	19
2.6. PROJETO HAWAII	20
2.7. EXPERIÊNCIA DO USUÁRIO (UX).....	21
2.8. LIBRAS	21
2.8.1. HISTÓRICO	21
2.8.2. INTRODUÇÃO	23
2.9. WINDOWS LIVE API	24
2.10 BING TRANSLATOR.....	25

3. MÉTODO DE PESQUISA E DIVISÃO DE TAREFAS	26
4. DESENVOLVIMENTO E IMPLEMENTAÇÃO DO PROJETO	27
4.1. CAMADA DE DADOS E SERVIÇO.....	28
4.1.1. FALA-TEXTO.....	30
4.2. CAMADA DE NEGÓCIOS	33
4.2.1. DIAGRAMAS DE CLASSES	34
4.2.2. <i>NAMESPACES</i>	36
4.3. CAMADA DE APRESENTAÇÃO.....	36
5. CONCLUSÃO E TRABALHOS FUTUROS	38
6. REFERÊNCIAS	39
7. GLOSSÁRIO	41

1. INTRODUÇÃO

Com frequência se diz que a tecnologia está transformando profundamente a educação. Ela desafia as definições existentes de conhecimento, oferece novas maneiras de motivar alunos e promete oportunidades de criatividade e inovação. Há uma longa história de afirmações pretensiosas como essas, que existem desde muito antes do advento dos computadores. Os primeiros defensores do uso de filmes e da televisão na educação, por exemplo, fizeram previsões similarmente fantásticas de que esses meios trariam mudanças profundas na natureza da aprendizagem.

Muitos professores resistem ao uso da tecnologia, não por serem antiquados ou ignorantes, mas porque reconhecem que ela não contribui para que eles alcancem seus objetivos. É claro que alguns professores estão usando a tecnologia de modo bastante criterioso e criativo; porém, na maioria dos casos, o uso de tecnologia nas escolas é estreito, sem imaginação e instrumental.

Já as crianças que têm acesso a computadores em casa estão usando-os para jogar, navegar em *sites* de entretenimento na internet, trocar mensagens instantâneas, participar de redes sociais, baixar e editar vídeos e músicas (Buckingham, 2008). Ou seja, no final não se sentem estimuladas por tecnologias ultrapassadas que são utilizadas em algumas escolas.

Hoje a tecnologia que atrai essas crianças são os *smartphones*. *Smartphone* é um celular com tecnologias avançadas, o que inclui programas executados em um sistema operacional equivalente aos computadores. Os *smartphones* possibilitam que qualquer pessoa possa desenvolver programas para eles, os chamados aplicativos, e existem dos mais variados tipos e para os mais variados objetivos.

A cada ano, novos modelos de *Smartphone* chegam ao Brasil e com preços cada vez mais acessíveis, isto está contribuindo para o crescimento do número de usuários desses aparelhos no país. Os *Smartphones* se distinguem quanto ao design e sistema operacional, mas se comportam sempre como indicadores de alta tecnologia. .

1.1. PROBLEMAS

Parece que a discussão da tecnologia pelos filósofos da educação sempre aparece num segundo plano e, normalmente, ela surge limitada por esquemas de interpretação de contextos socioeconômicos mais amplos. (Silva, 2005)

Diante de uma bela demonstração de que a modernização da educação é séria demais para ser tratada somente por técnicos. É um caminho interdisciplinar e a aliança da tecnologia com o humanismo é indispensável para criar uma real transformação. Em síntese, só terá sentido a incorporação de tecnologia na educação como na escola, se forem mantidos os princípios universais que regem a busca do processo de humanização, característico caminho feito pelo homem até então. (José, 1997).

Todos os dias muito conteúdo relevante é gerado em salas de aula ao redor do mundo. Porém o poder de alcance deste conteúdo não é muito grande. O mais comum é que somente os alunos que estiveram presentes na sala de aula é que tenham algum histórico do que foi passado pelo professor, e este histórico consiste normalmente de anotações feitas em caderno pelo próprio estudante.

Para estudantes com necessidades especiais o acesso à educação é ainda mais difícil. São raros os professores plenamente capacitados para transmitir conteúdo para alunos com deficiência auditiva, por exemplo. Além da quantidade ínfima de professores capacitados, as ferramentas de ensino atualmente disponíveis não dão suporte para a criação de um ambiente universal, onde estudantes com ou sem deficiência podem interagir da mesma maneira.

Historicamente a educação de pessoas com deficiências sempre ocupou uma posição marginal na política educacional brasileira. Entretanto, nos últimos anos, seguindo uma tendência mundial, a chamada educação inclusiva passou a fazer parte da agenda das políticas públicas, sobretudo, com a promulgação da Lei de Diretrizes e Bases da Educação Nacional – LDBEN 9394/96, que preconiza o atendimento de pessoas com deficiências no sistema regular de ensino. No Brasil existem aproximadamente 5,8 milhões de surdos, sendo que 30% desse montante não sabem ler o português. O restante (70%) até sabe ler, mas não têm entendimento claro da língua portuguesa. (Ribeiro, 2008)

1.2. JUSTIFICATIVA

As tecnologias evoluem em quatro direções fundamentais: do analógico para o digital (digitalização), do físico para o virtual (virtualização) do fixo para o móvel (mobilidade) e do massivo para o individual (personalização). (Carly Fiorina, ex-presidente da HPackard apud MORAN, 2007)

A digitalização permite registrar, editar, combinar, manipular toda e qualquer informação, por qualquer meio, em qualquer lugar, a qualquer tempo. A digitalização traz a multiplicação de possibilidades de escolha, de interação. A mobilidade e a virtualização nos libertam dos espaços e tempos rígidos, previsíveis, determinados.

As tecnologias que num primeiro momento são utilizadas de forma separada – computador, celular, Internet, mp3, câmera digital – e caminham na direção da convergência, da integração, dos equipamentos multifuncionais que agregam valor.

As metodologias de apoio à aprendizagem, de fato, não podem estar restritas aos procedimentos tradicionais frequentemente adotados pelos professores em salas de aula. O fornecimento de outros meios que permitam aos alunos o acesso ao conhecimento de um modo facilitado, prazeroso e, sobretudo, confortável, deve estar no conjunto dos principais objetivos das metodologias educacionais, em conformidade com as mais novas tendências na área. Estudos evidenciam, p.ex., que a utilização pedagógica das TICs (Tecnologias da Informação e Comunicação) produz resultados bastante animadores e satisfatórios para a educação especial, particularmente quando comparada à educação geral.

1.3. SOLUÇÃO

Kookaburra é uma plataforma que visa melhorar a facilidade de acesso a conteúdos educacionais, digitalizando-os e tornando-os móveis para serem acessados a qualquer momento em qualquer lugar.

Kookaburra se propõe a resolver os problemas apontados promovendo os seguintes tópicos:

- Democratização da informação;
- Conteúdo de longevidade maior;
- Informação portátil;
- Ambiente universal

Com Kookaburra o conteúdo gerado em sala de aula não é perdido, muito pelo contrário, cada palavra dita pelo professor em sala de aula é guardada na base de dados do sistema e fica disponível para qualquer estudante em forma de áudio, texto e linguagem de sinais. Além disso, o professor pode retornar a aula e incluir novos materiais de consulta, mantendo o material atualizado.

Tendo consciência de que o conteúdo de suas aulas estará totalmente disponível ao mundo todo, os professores tomarão atitudes para aumentar a qualidade de seu trabalho e adicionar experiências únicas para as atividades utilizadas em suas aulas.

Como o sistema funciona com um aplicativo móvel, o estudante tem acesso a uma fonte riquíssima de informação e conhecimento aonde quer que vá.

1.4. OBJETIVOS GERAIS E ESPECÍFICOS DO PROJETO

Ajudar os alunos a terem acesso rápido, fácil e organizado aos conteúdos e as informações de suas aulas. Esta solução também ajudará na compreensão do conteúdo em outras línguas, incluindo a linguagem de sinais, como LIBRAS.

O projeto denominado Kookaburra tem por objetivo gerenciar e distribuir conteúdos educativos gerados em aula, de forma globalizada com tradução do conteúdo para várias línguas e também linguagem de sinais. A solução é composta por 2 (dois) aplicativos para *smartphone* com sistema operacional *Windows Phone 7*.

O primeiro aplicativo, Kookaburra Teacher, tem o professor como foco. Com este aplicativo o professor pode gravar, utilizando o microfone de seu dispositivo, suas explicações sobre a matéria que estiver ministrando. Antes de começar uma aula, o professor cria os tópicos que achar necessário que serão abordados durante a aula, assim enquanto o conteúdo é gravado, o professor pode ir dividindo em tópicos com apenas um toque no aparelho. Depois que a aula acaba, o professor pode compartilhar conteúdos de referencia, colocando link de vídeos, textos e artigos da internet que achar interessante para complementar o material do aluno. Também é criada uma agenda compartilhada com os alunos para marcar datas de provas, trabalhos, laboratórios, etc.

O segundo aplicativo, Kookaburra Student, é para que os alunos possam acessar os conteúdos e informações que o professor compartilhar. O aluno utiliza o aplicativo para acompanhar suas próprias aulas ou para pesquisar em conteúdos gravados por outros professores sobre o mesmo assunto. Todo conteúdo gravado pelo professor estará disponível em áudio e texto, este texto também poderá ser traduzido para outras línguas, inclusive LIBRAS. O aluno também possui um calendário compartilhado com o professor que é alimentado de acordo com as atualizações feitas pelos professores das aulas que ele segue. Portanto se, por exemplo, o professor de Biologia marcar uma prova para terça-feira, o aluno com certeza vai ser notificado.

Usando a sessão de pesquisa do aplicativo, é fácil encontrar matérias e aulas de professores do mundo todo, já que o aplicativo estará disponibilizado para todos os países atendidos pelo *Marketplace* do *Windows Phone*.

Para armazenar e compartilhar todo conteúdo gerado, assim como processar as conversões de fala-texto do áudio da aula realizada pelo professor, desenvolvemos um serviço hospedado em um servidor *Windows Azure*.

2. FUNDAMENTAÇÃO TEÓRICA

2.1. TECNOLOGIA DA INFORMAÇÃO E COMUNICAÇÃO (TIC) NAS ESCOLAS

A escola hoje está repleta de alunos com perfil multitarefa, tecnologia e velocidade são comuns a essa geração e está cada vez mais difícil para o professor atrair esses alunos apenas com o quadro negro e giz branco feito de sulfato de cálcio. Nos últimos tempos, novos aparelhos têm surgido no espaço escolar. São os celulares, que em versão mais completa, são chamados de *smartphones* e, de forma ainda tímida, os *tablets*. Reclamações surgem de todos os lados e, o que se dissemina é a nocividade desses aparelhos à sala de aula, por dificultarem a concentração e atrapalharem a aprendizagem. Mas será que não é possível encontrar nenhuma contribuição? (MATEUS & BRITO, 2011).

As tecnologias são introduzidas nas salas de aulas como ferramentas, reproduzindo o modelo tradicional de ensino vigente. No entanto, a inserção de computadores portáteis na sala de aula, pode fazer com que as práticas pedagógicas sejam renovadas, no sentido de favorecer a criação de ambientes de aprendizagens que enfatizem o conhecimento construído entre aluno e professor. (SANTOS, 2009)

2.2. EDUCAÇÃO E TECNOLOGIA NO BRASIL

Na rede pública, professores estão familiarizados com tecnologia e internet, mas estão despreparados para usa-los em sala de aula. 80% do uso pedagógico de computadores são feitos sem alunos, em atividades como preparo de provas e listas de exercícios. A atividade mais realizada com alunos é copiar e colar conteúdo, em programas de edição de texto, visualização de mapas e de apresentação. (Pesquisa IBOPE, 2009) Ou seja, a tecnologia é usada principalmente para automatizar e facilitar tarefas simples.

“O próprio computador não é um projeto (...). É preciso ter um projeto, e ter um projeto não pode ser confundido com ter um software pronto para ser utilizado” - Nilson José Machado.

2.3. TENDÊNCIAS TECNOLÓGICAS

Com o crescimento de ambientes de consumo de conteúdo e com o grande número de dispositivos móveis em uso hoje, o Estado deverá fornecer para a Educação recursos que complementem os já sendo utilizados por professores e alunos, e não que os substituam. Soluções interoperáveis, nas nuvens e sistemas com suporte a toque terão mais sucesso.

2.4. SMARTPHONES

Atualmente os usuários de telefones celulares em todo o mundo estão trocando seus aparelhos com menos funções pelos *smartphones*. Estes que evoluem junto com as tendências em tecnologia da informação e tecnologia móvel. Geralmente um *smartphone* possui características mínimas de *hardware* e *software*, sendo as principais a capacidade de conexão com redes de dados para acesso à internet, a capacidade de sincronização dos dados do organizador com um computador pessoal, e uma agenda de contatos que pode utilizar toda a memória disponível do celular.

Estes dispositivos estão se tornando cada vez mais sofisticados e a última geração de celulares inteligentes agora incorpora diversos e poderosos sensores. Estes sensores incluem sensores de GPS, sensores de visão (ou seja, câmeras), sensores de áudio (ou seja, microfones), sensores de luz, sensores de temperatura, sensores de direção (ou seja, bússolas magnéticas) e acelerômetros. Uma grande fatia desses dispositivos possuem sistemas embarcados mantidos por grandes empresas como *Android* da *Google*, *IOS* da *Apple*, ***Symbian*** da *Nokia* e *Windows Phone* da *Microsoft*.

2.4.1. WINDOWS PHONE 7

Entrando em completa ruptura com as antigas versões da sua plataforma móvel *Windows Mobile*, a *Microsoft* apresenta no *Windows Phone 7* uma nova interface gráfica, denominada "Metro". Os aplicativos para *Windows Phone 7* são criados usando ferramentas conhecidas, como o *Visual Studio 2010*, o

Silverlight e o XNA, permitindo que os desenvolvedores criem aplicativos avançados para *smartphones* com suas habilidades existentes. Como uma plataforma segura, os aplicativos para *Windows Phone 7* conseguem tirar proveito dos vários serviços e recursos disponíveis na nuvem.

A execução de algumas aplicações em dispositivos móveis tem ainda um custo proibitivo, apesar de o crescente poder computacional que tem vindo a caracterizar cada nova geração de smartphones e da relevância destas aplicações para os usuários. As concretizações atuais destas aplicações delegam o esforço de computação em servidores na Internet, o que pressupõe uma conectividade permanente.

2.5. COMPUTAÇÃO EM NUVEM

2.5.1. INTRODUÇÃO

A Computação em Nuvem é um conceito que começou a se delinear recentemente (últimos quatro anos) e que representa um novo paradigma em relação à infraestrutura, armazenamento e processamento de dados computacionais.

O termo “Computação em Nuvem” surgiu em 1961, a partir de uma ideia de John McCarthy, professor e especialista em Inteligência Artificial do Instituto de Tecnologia de Massachusetts (MIT) que, em formato rudimentar, apresentava um modelo de computação oferecido como um serviço, aos moldes do serviço de distribuição de energia elétrica.

É conceituado como a interligação de todos os sistemas computacionais de uma organização, criando um *pool* de recursos dinâmicos e implementando o conceito de virtualização de computadores, sendo uma maneira bastante eficiente de maximizar e flexibilizar os recursos computacionais. (Uebe Mansur, et al, 2010)

2.5.2. ARQUITETURA

A base estrutural para o conceito de Computação em Nuvens é a *Infrastructure as a Service* (IaaS) ou Infraestrutura em Nuvens, que trata da infraestrutura de servidores; isto minimiza as preocupações das organizações manterem um parque tecnológico para suportar as aplicações, pois estes recursos físicos, ou seja, hardware são serviços remotos onde já existem no mercado global empresas que comercializam, ou seja, alugam repositórios de dados (*Data Center*) que fazem a hospedagem das aplicações. Um exemplo deste cenário é o serviço oferecido pela empresa Locaweb (Figura 1). (Pedrosa et. al., 2010)



Figura 1 – Visão Geral da Computação em Nuvem

A computação em nuvem pode ser dividida em 3 tipos: *Software* como Serviço, Plataforma como um Serviço, Infraestrutura como um Serviço;

2.5.2.1. SOFTWARE COMO SERVIÇO (SaaS)

O software como serviço é muitas vezes caracterizado como: *Software* implementado como um serviço hospedado e acessado pela Internet. Esta afirmação ou definição nos leva a fazer uma reflexão sobre o assunto em questão. Segundo Melo (2007), percebe-se que esta definição não descreve nenhuma arquitetura de aplicativo específica; não descreve nenhuma tecnologia; não

apresenta distinção entre serviços orientados a negócios e/ou serviços orientados a consumidor, além de não exigir modelos de negócios específicos.

De acordo com essa definição, os detalhes principais de distinção do software como serviço se faz de acordo com o local em que o código da aplicação está residente e como ele é desenvolvido e acessado. Analisando de maneira abrangente, ou seja, do geral para o específico, evidencia-se duas categorias principais de software como serviço (Choudhary, 2007):

- **Serviços de linha de negócios:** oferecidos a organizações privadas e públicas, seja de pequeno, médio ou grande porte. Os serviços de linha de negócios geralmente são soluções de negócios personalizáveis direcionados para facilitar processos de negócios como finanças, cadeia de suprimentos e relações com o cliente.
- **Serviços orientados a cliente:** oferecidos ao público em geral. Os serviços orientados a cliente na maioria das vezes são vendidos como assinatura, mas geralmente são fornecidos sem custo e financiados por anúncios.

2.5.2.2. PLATAFORMA COMO SERVIÇO (PaaS)

A PaaS oferece uma infraestrutura para implementação, teste e integração de *softwares* na nuvem. O usuário não gerencia a infraestrutura, incluindo servidores, sistemas operacionais ou armazenamento dentre outros, mas tem total controle sobre as aplicações implantadas nesta infraestrutura. A PaaS fornece um sistema operacional, linguagens de programação e ambientes de desenvolvimento para as aplicações, apoiando o desenvolvimento de sistemas de software (Sousa, 2009).

Em resumo, os desenvolvedores têm a sua disposição um ambiente escalável, contudo eles têm que conviver com algumas restrições sobre o tipo de software que se pode desenvolver, seja limitações do tipo de linguagem de programação, Banco de Dados e etc.

Segundo (Sousa, 2009) do ponto de vista do negócio, a PaaS permitirá aos usuários utilizarem serviços de terceiros, aumentando o uso do modelo de suporte no qual os usuários se inscrevem para solicitações de serviços de TI ou para resoluções de problemas pela Web.

2.5.2.3. INFRAESTRUTURA COMO SERVIÇO (IaaS)

O IaaS é a parte responsável por prover toda a infraestrutura necessária para a PaaS e o SaaS. O principal objetivo do IaaS é tornar mais fácil e acessível o fornecimento de recursos, tais como servidores, rede, armazenamento e outros recursos de computação fundamentais para construir um ambiente sob demanda, que podem incluir sistemas operacionais e aplicativos.

A mesma consiste em entregar infraestrutura tecnológica (basicamente um ambiente com plataforma de virtualização) como serviço. Essa infraestrutura utilizando plataformas de virtualização é exemplo de uma solução inteira como um serviço. Ao invés de comprar supercomputadores, softwares, equipamentos de rede, os clientes podem comprar estes recursos como um serviço terceirizado. Este serviço é tarifado baseando-se na quantidade de recursos consumidos. Diante desta discussão, deste cenário, podemos afirmar que a Computação nas Nuvens ou Cloud Computing como Serviço não é sobre tecnologia, mas sobre um novo modelo de negócios. (Ramos & H. de Farias Junior)

2.5.3. CARACTERÍSTICAS ESSENCIAIS

As características essenciais são as vantagens que as soluções de computação em nuvem oferecem. Algumas destas características, em conjunto, definem exclusivamente a computação em nuvem e fazem a distinção com outros paradigmas. Por exemplo, a elasticidade rápida de recursos, amplo acesso e a medição de serviço são características básicas para compor uma solução de computação em nuvem (Sousa et. al., 2010):

- **Self-service sob demanda:** O usuário pode adquirir unilateralmente recurso computacional, como tempo de processamento no servidor ou armazenamento na rede, na medida em que necessite e sem precisar de interação humana com os provedores de cada serviço.
- **Amplo acesso:** Recursos são disponibilizados por meio da rede e acessados através de mecanismos padronizados que possibilitam o uso por plataformas móveis, tais como smartphones ou *tablets*.

- **Pooling de recursos:** Os recursos computacionais do provedor são organizados em um *pool* para servir múltiplos usuários usando um modelo *multi-tenant* ou multiinquilino, com diferentes recursos físicos e virtuais, dinamicamente atribuídos e ajustados de acordo com a demanda dos usuários. Estes usuários não precisam ter conhecimento da localização física dos recursos computacionais, podendo somente especificar a localização em um nível mais alto de abstração, tais como o país, estado ou centro de dados.
- **Elasticidade rápida:** Recursos podem ser adquiridos de forma rápida e elástica, em alguns casos automaticamente, caso haja a necessidade de escalar com o aumento da demanda, e liberados, na retração dessa demanda. Para os usuários, os recursos disponíveis para uso parecem ser ilimitados e podem ser adquiridos em qualquer quantidade e a qualquer momento.
- **Serviço medido:** Sistemas em nuvem automaticamente controlam e otimizam o uso de recursos por meio de uma capacidade de medição. A automação é realizada em algum nível de abstração apropriado para o tipo de serviço, tais como armazenamento, processamento, largura de banda e contas de usuário ativas. O uso de recursos pode ser monitorado e controlado, possibilitando transparência para o provedor e o usuário do serviço utilizado.

2.5.4. WINDOWS AZURE

A plataforma *Windows Azure* é uma plataforma como serviços (PaaS – platform as a service) oferecendo recurso para computação de processos (*Compute*), armazenamento de objetos e dados (*Storage*), gerenciamento de serviços e recursos contratados (*Management*) (Cambiucci, 2011).

Podemos destacar quatro grandes grupos de capacidades sobre a plataforma Windows Azure, a saber:

- **Windows Azure:** com seus recursos para computação, hospedagem de aplicações e serviços, gerenciamento e armazenamento de dados de alta escalabilidade, via o *Azure Storage*;

- **SQL Azure:** com seus recursos para relatórios, persistência de dados relacional, sincronização de dados com o ambiente *on-premise* e gerenciamento de dados na nuvem;
- **Windows Azure AppFabric:** com seus recursos para gerenciamento e publicação de serviços, através do *Service Bus*, e controle de acesso, através do *Access Control*. Funcionalidades como o *Azure AppFabric Caching* oferecem recursos adicionais para aplicação na nuvem com alta performance.
- **Windows Azure DataMarket:** finalmente, o *Azure DataMarket* oferece mecanismos diversos para *billing*, gerenciamento de pagamentos, subscrição, informações sobre serviços publicados e relatórios para o controle de aplicações oferecidas na nuvem.

2.6. PROJETO HAWAII

O projeto Hawaii é uma solução desenvolvida juntamente com diversas universidades, que visa facilitar o acesso a alguns serviços em nuvem.

Com o Projeto Hawaii, você pode desenvolver aplicações avançadas *Windows Phone* que acessam um conjunto de serviços em nuvem do *Windows Azure*, tanto para processamento como para armazenamento de dados (Project Hawaii, 2010):

- **Serviço de transmissão:** Este serviço fornece um ponto de transmissão na nuvem que as aplicações móveis podem usar para se comunicar. Ele fornece um esquema de nomenclatura e de armazenamento temporário para mensagens enviadas entre um ou mais dispositivos.
- **Serviço de Reconhecimento ótico de caracteres (OCR):** Este serviço pega uma imagem fotográfica que contém algum texto e retorna o texto. Por exemplo, dada uma imagem JPEG de um sinal de trânsito, o serviço seria capaz de retornar o texto do sinal.
- **Serviço fala-texto:** Utilizado para converter uma frase falada em texto (atualmente apenas em Inglês).

2.7. EXPERIÊNCIA DO USUÁRIO (UX)

Experiência do Usuário (*User Experience*) é como as pessoas se sentem quando interagem com um sistema. Este sistema pode ser um uma aplicação web, para dispositivo móvel, ou para *desktop*.

Ela inclui as emoções, preferências, percepções, reações físicas e psicológicas, comportamentos e realizações do usuário que ocorrem antes, durante e após o uso. Ela é considerada um fator de natureza subjetiva, mas inclui também a percepção em relação aos aspectos práticos, tais como utilidade, facilidade de utilização e eficiência do sistema.

A *UX* é um conjunto de diversas disciplinas, incluindo engenharia de usabilidade, arquitetura de informação, design de interação e interface.

O primeiro requisito de uma *UX* exemplar é atender as necessidades dos usuários através de um processo de *design* centrado no usuário. Design centrado no usuário é um processo e um conjunto de técnicas utilizadas para criar novas soluções, que incluem produtos, serviços, ambientes, organizações e modos de interação. A razão pela qual este processo é chamado de “centrado no usuário” é porque ele começa por analisar as necessidades, objetivos, desejos e comportamentos das pessoas que podem impactar na solução que será projetada.

2.8. LIBRAS

2.8.1. HISTÓRICO

“É de 1873 a publicação do mais importante documento encontrado até hoje sobre a Língua Brasileira de Sinais, o *Iconographia dos Signaes dos Surdos-Mudos*, de autoria do aluno surdo Flausino José da Gama, com ilustrações de sinais separados por categorias (animais, objetos, etc). Como é explicado no prefácio do livro, a inspiração para o trabalho veio de um livro publicado na França e que se encontrava à disposição dos alunos na Biblioteca do INSM. Vale ressaltar que Flausino foi autor das ilustrações e da própria impressão em técnica de litografia.

Em 1911, seguindo os passos internacionais que em 1880 no Congresso de Milão proibira o uso da Língua de Sinais na educação de surdos, estabelece-se que o INSM passaria a adotar o método oralista puro em todas as disciplinas. Mesmo assim, muitos professores e funcionários surdos e os ex-alunos que sempre mantiveram o hábito de freqüentar a escola, propiciaram a formação de um foco de resistência e manutenção da Língua de Sinais.

Somente em 1957, por iniciativa da diretora Ana Rímoli de Faria Doria e por influência da pedagoga Alpia Couto, finalmente a Língua de Sinais foi oficialmente proibida em sala de aula.” (Ramos C. R., 2004)

Medidas como o impedimento do contato de alunos mais velhos com os novatos foram tomadas, mas nunca o êxito foi pleno e a LIBRAS sobreviveu durante esses anos dentro do atual INES.

“Em depoimento informal, uma professora que atuou naquela época de proibições (que durou, aliás, até a década de 1980) contou-nos que os sinais nunca desapareceram da escola, sendo feitos por debaixo da própria roupa das crianças ou embaixo das carteiras escolares ou ainda em espaços em que não havia fiscalização. É evidente, porém, que um tipo de proibição desses gera prejuízos irreversíveis para uma língua e para uma cultura.

Pesquisar as origens da LIBRAS é realmente uma tarefa a ser realizada, pois surpreende a todos aqueles que trabalham com a comunidade surda brasileira (tão espalhada por este imenso país) a homogeneidade lingüística da mesma. Apesar dos "sotaques" regionais, podemos observar apenas algumas variações lexicais que não comprometem em nenhum momento sua unidade estrutural.

Em 1969, foi feita uma primeira tentativa no sentido de tentar registrar a Língua de Sinais falada no Brasil. Eugênio Oates, um missionário americano, publica um pequeno dicionário de sinais, *Linguagem das mãos*, que segundo Ferreira Brito (1993), apresenta um índice de aceitação por parte dos surdos de 50% dos sinais listados.

A partir de 1970, quando a filosofia da Comunicação Total e, em seguida, do Bilingüismo, firmaram raízes na educação dos surdos brasileiros, atividades e pesquisas relativas à LIBRAS.

Em 2001 foi lançado em São Paulo o Dicionário Enciclopédico Ilustrado de LIBRAS, em um projeto coordenado pelo Professor Doutor (Instituto de Psicologia/USP) Fernando Capovilla e em março de 2002 o Dicionário LIBRAS/Português em CD-ROM, trabalho realizado pelo INES/MEC e coordenado pela Professora Doutora Tanya Mara Felipe/UFPernambuco/FENEIS.

Nacionalmente, a LIBRAS foi, recentemente, oficializada através da Lei n.º 4.857 /2002, enquanto língua dos surdos brasileiros, o que, aliada à aceitação da LIBRAS pelo MEC, irá tornar a educação dos surdos e a vida dos surdos cada vez mais fácil.” (Ramos C. R., 2004)

2.8.2. INTRODUÇÃO

“A LIBRAS, como toda Língua de Sinais, é uma língua de modalidade gestual-visual porque utiliza, como canal ou meio de comunicação, movimentos gestuais e expressões faciais que são percebidos pela visão; portanto, diferencia-se da Língua Portuguesa, que é uma língua de modalidade oral-auditiva por utilizar, como canal ou meio de comunicação, sons articulados que são percebidos pelos ouvidos. Mas, as diferenças não estão somente na utilização de canais diferentes, estão também nas estruturas gramaticais de cada língua.” (Revista da FENEIS, número 2:16)

Resumidamente, pode-se afirmar que:

Os sinais são formados a partir da combinação do movimento das mãos com um determinado formato em um determinado lugar, podendo este lugar ser uma parte do corpo ou um espaço em frente ao corpo. Estas articulações das mãos, que podem ser comparadas aos fonemas e às vezes aos morfemas, são chamadas de parâmetros, portanto, nas Línguas de Sinais podem ser encontrados os seguintes parâmetros:

1. **configuração das mãos:** são formas das mãos, que podem ser da datilologia (alfabeto manual) ou outras formas feitas pela mão predominante (mão direita

para os destros), ou pelas duas mãos do emissor ou sinalizador. Os sinais APRENDER, LARANJA e ADORAR têm a mesma configuração de mão;

2. **ponto de articulação:** é o lugar onde incide a mão predominante configurada, podendo esta tocar alguma parte do corpo ou estar em um espaço neutro vertical (do meio do corpo até à cabeça) e horizontal (à frente do emissor). Os sinais TRABALHAR, BRINCAR, CONSERTAR são feitos no espaço neutro e os sinais ESQUECER, APRENDER e PENSAR são feitos na testa;
3. **movimento:** os sinais podem ter um movimento ou não. Os sinais citados acima tem movimento, com exceção de PENSAR que, como os sinais AJOELHAR, EM-PÉ, não tem movimento;
4. **orientação:** os sinais podem ter uma direção e a inversão desta pode significar idéia de oposição, contrário ou concordância número-pessoal, como os sinais QUERER E QUERER-NÃO; IR e VIR;
5. **expressão facial e/ou corporal:** muitos sinais, além dos quatro parâmetros mencionados acima, em sua configuração tem como traço diferenciador também a expressão facial e/ou corporal, como os sinais ALEGRE e TRISTE. Há sinais feitos somente com a bochecha como LADRÃO, ATO-SEXUAL.

Na combinação destes quatro parâmetros, ou cinco, tem-se o sinal. Falar com as mãos é, portanto, combinar estes elementos que formam as palavras e estas formam as frases em um contexto.” (Revista da FENEIS, número 2: 16)

2.9. WINDOWS LIVE API

O *Windows Live* fornece uma série de APIs que permitem que os seus aplicativos operem com informações do *Hotmail*, *Windows Live Messenger*, *Microsoft SkyDrive* e outros serviços compatíveis com o *Live Connect*.

As APIs (*Application Programming Interface*) do *Live Connect* usam protocolos padrão do setor como OAuth 2.0, JSON (*JavaScript Object Notation*) e XMPP (*Extensible Messaging and Presence Protocol*). Para chamar as APIs, use principalmente solicitações de REST (*Representational State Transfer*) que retornam informações formatadas em JSON. Essa arquitetura nos permite suportar várias

plataformas, incluindo aquelas para aplicativos *Web*, *desktop* e móveis, como para sistemas operacionais iOS e Android. Além disso, há SDKs (kits de desenvolvimento de *software*) da Microsoft para o desenvolvimento de aplicativos do Windows Phone e Windows 8.

2.10 BING TRANSLATOR

Bing Translator (anteriormente *Live Search Translator* e *Windows Live Translator*) é um portal de tradução desenvolvido pela Microsoft como parte de seus serviços Bing para traduzir textos ou páginas inteiras em diferentes idiomas.

Todos os idiomas são fornecidos através do *web service* Microsoft Translator, desenvolvido pela *Microsoft Research*, como seu *software* de tradução *backend*. (Bing Translator, 2009).

3. MÉTODO DE PESQUISA E DIVISÃO DE TAREFAS

A priori foi pesquisado as tecnologias envolvidas para a criação de uma aplicação para *smartphone* com conversão fala-texto e texto para linguagem de sinais apresentados na Fundamentação Teórica deste documento. Também foi necessário pesquisar sobre sistemas distribuídos, pois como o processamento de um celular é restrito, utilizou-se o processamento em nuvem, a fim de aumentar a performance do sistema, para tentar tornar possível uma conversão *real-time*.

Assim foram realizadas pesquisas sobre algumas tecnologias como, *Windows Phone SDK*, *Speech-Text API*, *Windows Azure* e *SQL Azure*, *Bing Translator*, *Windows Live API*.

Como já existem várias bibliotecas de conversão fala-texto consolidadas, como *Google Voice API*, *Microsoft Bing Services* e *Hawaii API*, para decidir qual API atendia os requisitos do sistema, testou-se a performance das 3 (três) APIs de conversão.

O próximo passo era definir uma arquitetura para o sistema. Optou-se por gravar amostras de voz através de um *smartphone*; então essas amostras são enviadas para um servidor na nuvem (*Windows Azure*) onde é processada a conversão fala-texto e enfim de texto para linguagem de sinais.

Com os estudos realizados e arquitetura definida, o desenvolvimento do projeto foi dividido entre os membros da equipe. A solução consiste em 2 aplicativos. Cada membro da equipe desenvolveu um aplicativo. O aluno Pedro Cavalca ficou responsável por desenvolver o aplicativo do aluno, a base de dados da plataforma e toda camada de acesso aos dados armazenados; traduções de texto para outras línguas; e interface. O aluno Thiago Fabre ficou responsável por desenvolver o aplicativo do professor que capta o áudio e transmite para o servidor processar a conversão para texto e linguagem de sinais; também desenvolveu a agenda compartilhada do professor.

4. DESENVOLVIMENTO E IMPLEMENTAÇÃO DO PROJETO

A figura abaixo demonstra a arquitetura de 3 camadas utilizada na implementação do projeto.

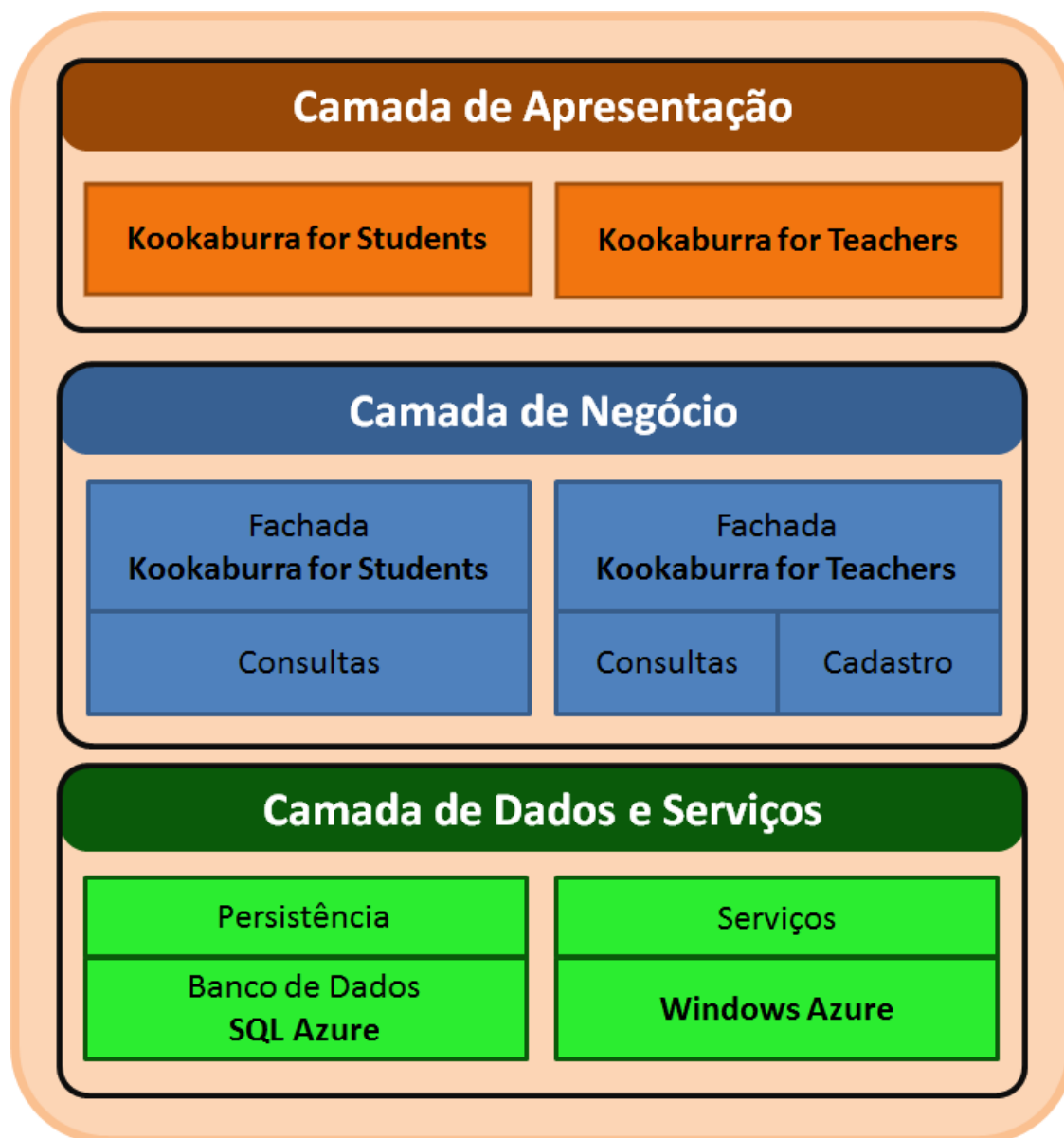


Figura 2 – Arquitetura da Solução

4.1. CAMADA DE DADOS E SERVIÇO

O serviço e base de dados foram implantados na plataforma Windows Azure, pois o Windows Azure provisiona máquinas virtuais para a oferta de serviços online de alta escalabilidade e provisionamento dinâmico, o que oferece a aplicação Kookaburra uma grande capacidade para processar os *streams* de áudio e texto que são transmitidos durante as *Live Lessons*.

A solução de Windows Azure de *Software* como Serviço confere a plataforma um enorme potencial de expansão para ser desenvolvido futuramente aplicativos para outros sistemas operacionais de *smartphones*, um portal *web* ou mesmo uma versão *desktop*.

Para tornar possível a rápida localização de conteúdos e também a tradução para outras línguas incluindo linguagem de sinais, utilizamos 2 (dois) serviços de ponta oferecidos pela *Microsoft*. O primeiro serviço é o *Project Hawaii*, ainda em desenvolvimento acadêmico, com este serviço é possível transcrever o áudio que contém a explicação da matéria feita pelo professor em sala de aula. Este serviço também é hospedado em um servidor *Windows Azure*. O segundo serviço é o Bing Translator que tornou a aplicação Kookaburra mundial, permitindo que conteúdos em outras línguas possam ser traduzidas para a língua nativa do usuário.

A base de dados foi implantada no SQL Azure. Essa base é utilizada para armazenar e compartilhar todas as informações necessárias para a utilização dos aplicativos. A figura 3 abaixo apresenta o Diagrama de Tabelas gerado pela ferramenta de administração do SQL Azure. Este diagrama apresenta as tabelas e seus respectivos relacionamentos:

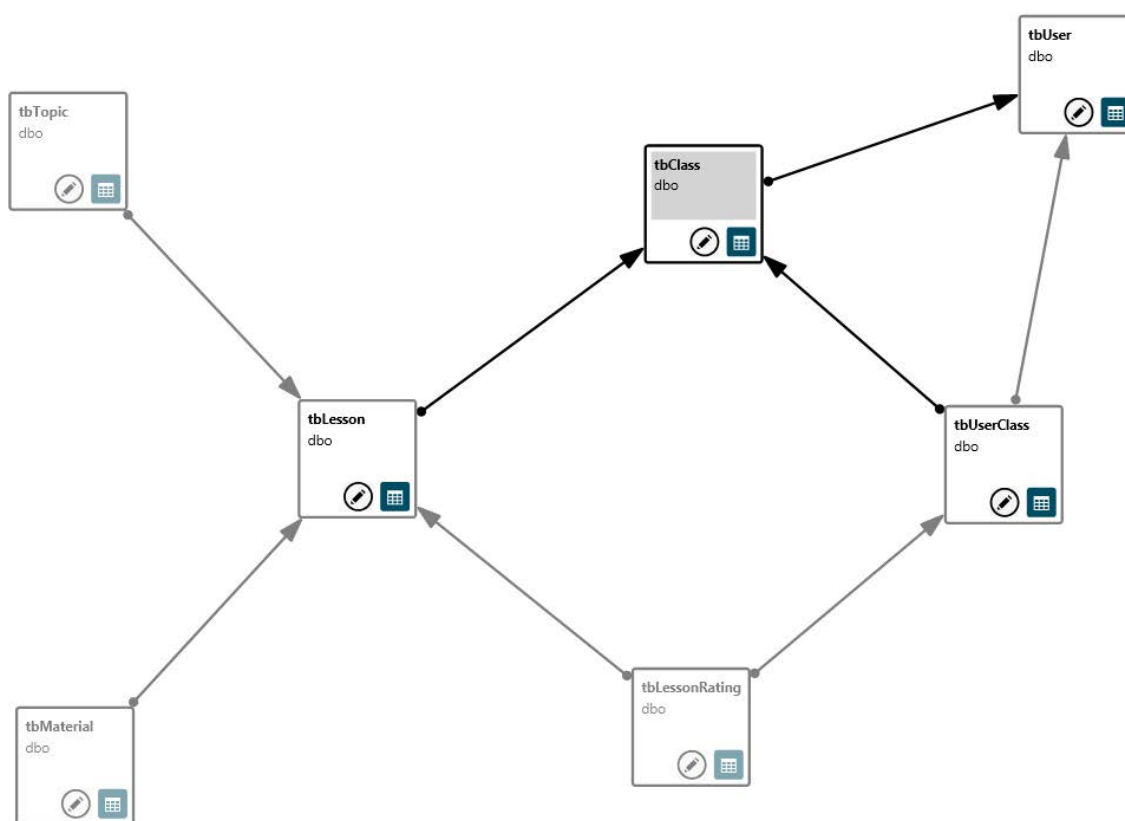


Figura 3 – Diagrama de Tabelas

Na camada de acesso a dados foi aplicada a tecnologia *WCF Data Services*. Windows Azure mais *WCF Data Services* ofereceram um alto grau de interoperabilidade para a plataforma.

O *WCF Data Services* permite a criação e o consumo de serviços *OData* para a *Web*, ou seja, ele mapeia o banco de dados em classes e objetos para serem utilizados na comunicação do aplicativo com o banco de dados. A figura 4 representa o Diagrama de Classes do Banco de Dados mapeado:

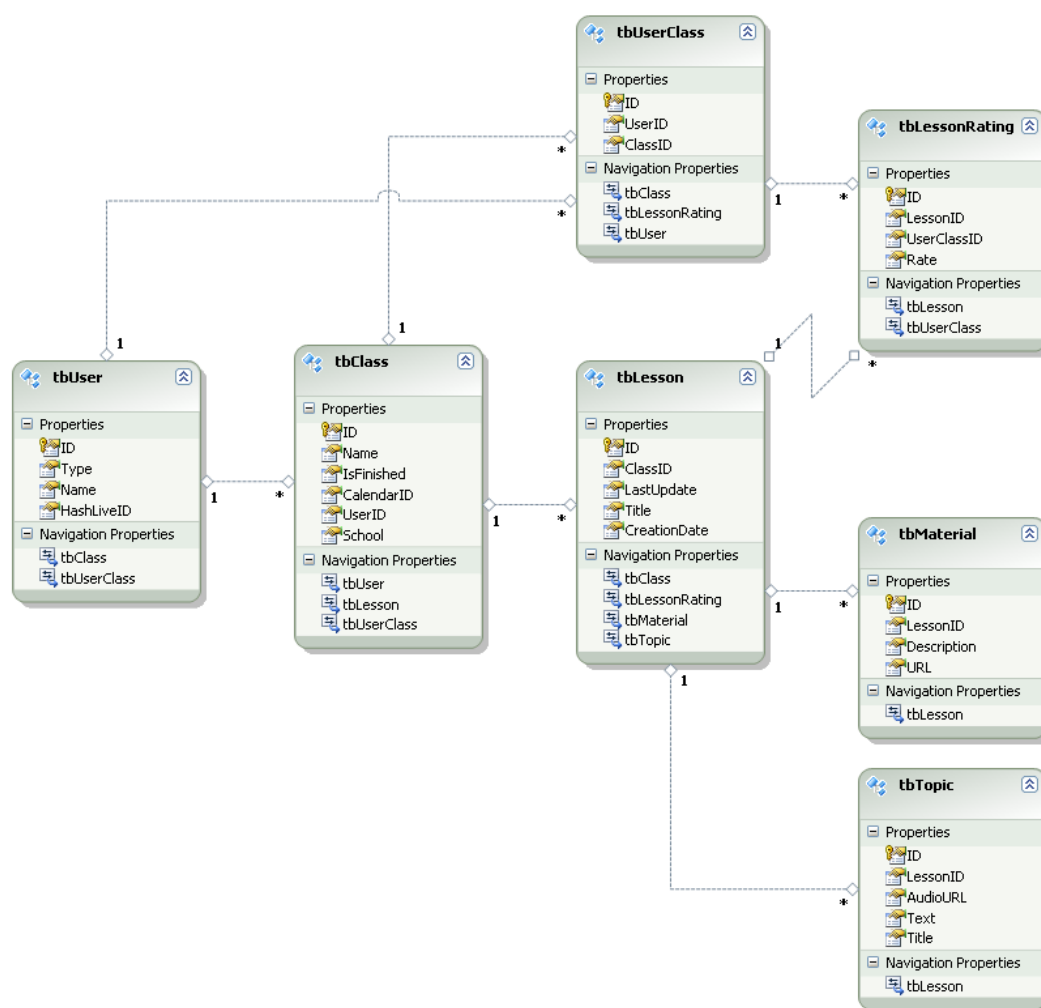


Figura 4 – Diagrama de Classes da Base de Dados

4.1.1.1. FALA-TEXTO

Para a conversão fala-texto, optou-se pelo *Project Hawaii Speech-to-Text* (STT) desenvolvido pela *Microsoft Research*, pois além da facilidade maior para implementação, ela possui boa velocidade de resposta, já que tudo é processado nas nuvens. Para a comunicação com o serviço *Project Hawaii Speech-to-Text* (STT), utilizamos a biblioteca *Speech Recognition Client* que vem com a *Project Hawaii SDK*. Essa biblioteca possui três principais classes:

- **SpeechResult**: Descreve o resultado de uma chamada ao *Hawaii Speech-to-Text*.

- **SpeechService**: Classe de suporte que fornece o acesso ao serviço do Hawaii Speech-to-Text.
- **SpeechServiceResult**: Representa o resultado do processamento de voz para texto.

Exemplo de comunicação com o serviço de tradução fala-texto:

1º Trazer lista de gramaticas do servidor

Para trazer a lista de gramáticas do servidor, utilizou-se o método *GetGrammarsAsync* da *SpeechService*.

```
SpeechService.GetGrammarsAsync(
    HawaiiClient.HawaiiApplicationId,
    (result) =>
    {
        this.Dispatcher.BeginInvoke(() =>
            this.OnSpeechGrammarsReceived(result));
    });
```

Como a função faz um chamada assincrona com o servidor, precisa atribuir um *callback*, e como podemos ver acima foi utilizado o método *OnSpeechGrammarsReceived*, e nesse método verifica-se quais são as gramáticas que serão utilizadas. A classe *SpeechServiceResult* traz o resultado obtido no servidor

```
private void OnSpeechGrammarsReceived(SpeechServiceResult result)
{
    Debug.Assert(result != null, "result is null");

    this.RecognizingProgress.Visibility = Visibility.Collapsed;
    this.RetrievingGrammarsLabel.Visibility = Visibility.Collapsed;
    if (result.Status == Status.Success)
    {
        this.SetButtonStates(true, false, false, false);
        this.SpeechDomainsList.Visibility = Visibility.Visible;

        this.availableGrammars = result.SpeechResult.Items;
```



```

        if (this.availableGrammars == null)
        {
            return;
        }

        this.SpeechDomainsList.Items.Clear();

        if (this.availableGrammars != null)
        {
            this.availableGrammars.ForEach((item) =>
                this.SpeechDomainsList.Items.Add(item));
        }
    }
    else
    {
        MessageBox.Show("Error receiving available speech grammars.",
            "Error", MessageBoxButton.OK);
        this.NoGrammarsLabel.Visibility = Visibility.Visible;
    }
}

```

2º Convertendo de áudio para texto

Para converter o áudio para texto utilizado o método *RecognizeSpeechAsync* da classe *SpeechService*. Esse método envia um *buffer* de áudio para o servidor, que fará a conversão e retornará apenas o texto.

Porém, o buffer a ser convertido precisa de algumas características especiais:

- SamplesPerSecond: 16000
- AudioBitsPerSample: 16
- AudioChannel Mono

```

SpeechService.RecognizeSpeechAsync(
    HawaiiClient.HawaiiApplicationId,
    "Dictation",
    this.AudioStream.ToArray(),
    (result) =>
    {
        this.Dispatcher.BeginInvoke(() =>
            this.OnSpeechRecognitionCompleted(result));
    });

```

Como o método *GetGrammarsAsync*, o *RecognizeSpeechAsync* também realiza chamadas assíncronas, então precisa-se do *callback*.

Na função de *callback* *OnSpeechRecognitionCompleted*, verifica-se a classe *SpeechServiceResult* para saber quais resultados foram obtidos, que é o texto convertido.

```
private void OnSpeechRecognitionCompleted(SpeechServiceResult
speechResult)
{
    Debug.Assert(speechResult != null, "speechResult is null");

    this.RecognizingProgress.Visibility = Visibility.Collapsed;

    if (speechResult.Status == Status.Success)
    {
        this.SetRecognizedTextListBox(speechResult.SpeechResult.Items);
    }
    else
    {
        if (speechResult.Exception == null)
        {
            MessageBox.Show("Error recognizing the speech.", "Error",
                MessageBoxButton.OK);
        }
        else
        {
            MessageBox.Show(speechResult.Exception.Message, "Error",
                MessageBoxButton.OK);
        }
    }
}
```

4.2. CAMADA DE NEGÓCIOS

Nesta camada estão contidas todas as regras e protocolos de transferência das informações. A figura 5 apresenta uma visão geral da arquitetura de comunicação do sistema:

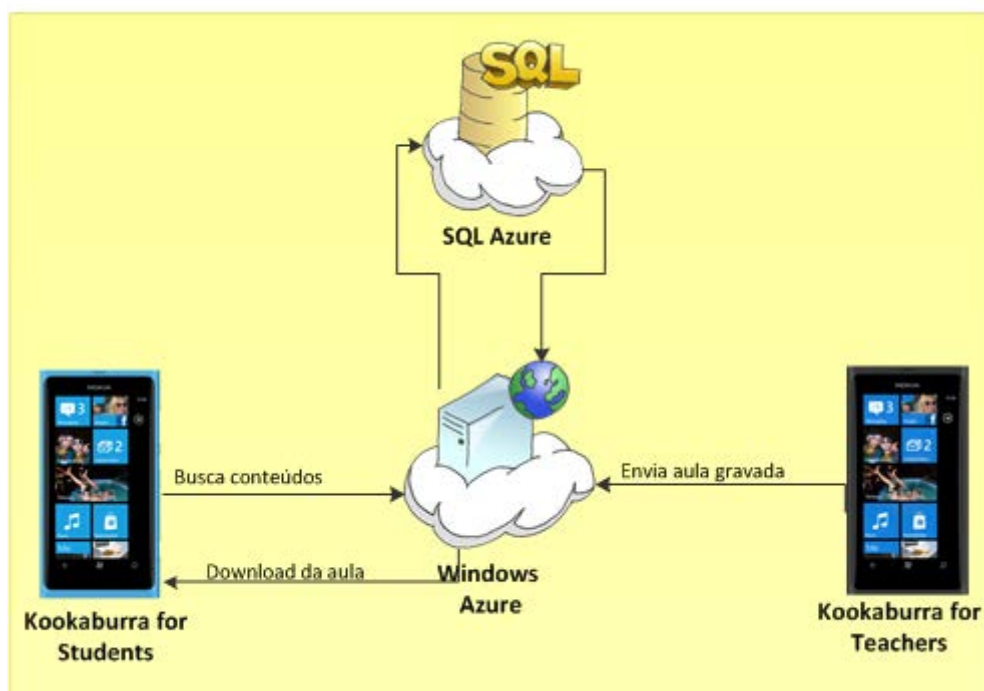


Figura 5 – Visão Geral da Arquitetura de Comunicação do Sistema

Utilizando seu *Live ID* o usuário poderá se autenticar e ter acesso aos recursos da plataforma. Quando o usuário loga, o sistema verifica se é a primeira vez que este usuário é autenticado, caso verdadeiro, ele é cadastrado automaticamente com os dados providos do ser perfil do *Live ID*.

4.2.1. DIAGRAMAS DE CLASSES

Esta seção provê um maior detalhamento dos componentes da aplicação, apresentando suas classes através de diagramas:

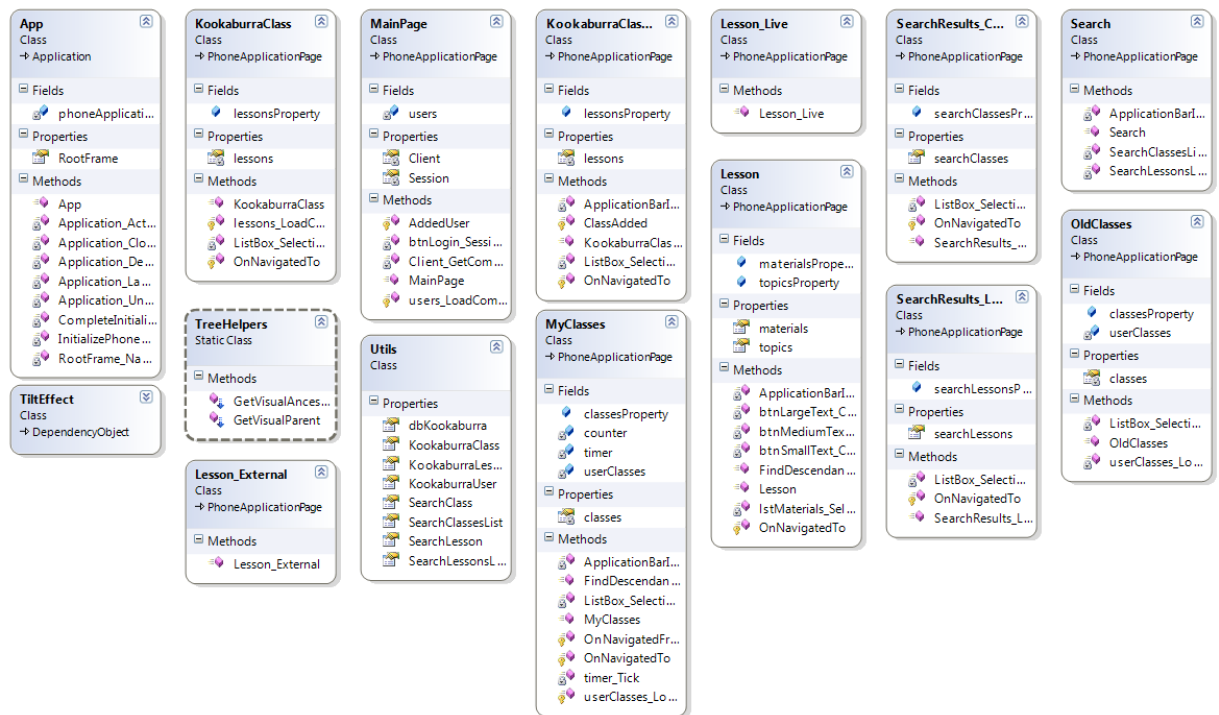


Figura 6 – Diagrama de Classe do Aplicativo Kookaburra Student

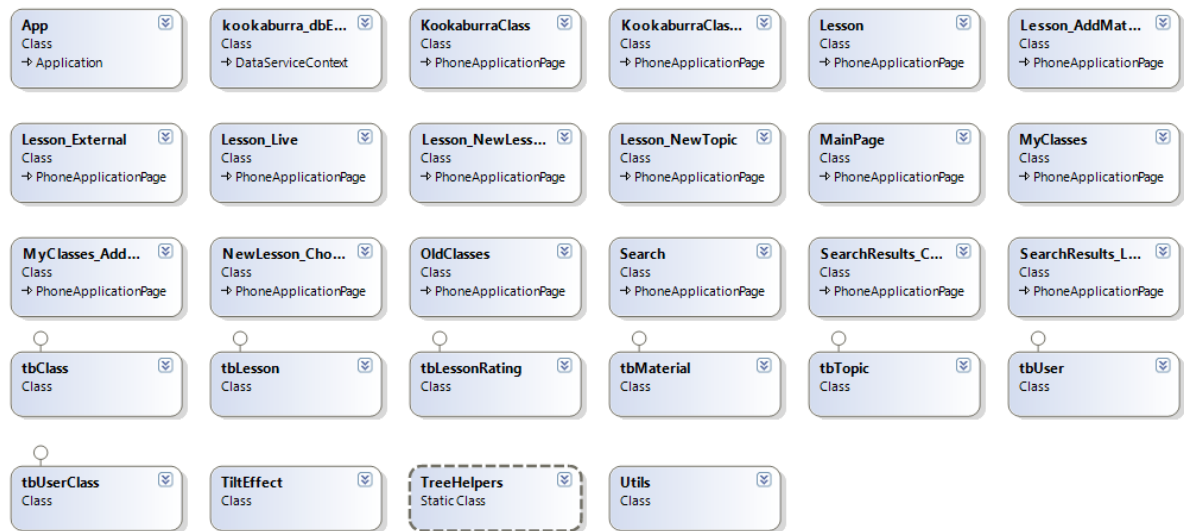


Figura 7 – Diagrama de Classe do Aplicativo Kookaburra Teacher

4.2.2. NAMESPACES

Esta seção apresenta os *namespaces* definidos para a implementação da aplicação.

- **Kookaburra For Students**
 - **KookaburraStudent.KookaburraStudent:** contém classes e métodos que definem o aplicativo para Windows Phone 7
 - **KookaburraStudent.KookaburraServiceReference:** contém classes para conexão com Serviços Web.
 - **KookaburraStudent.Pages:** contém classes e métodos que representam cada uma das telas do aplicativo
- **Kookaburra For Teachers**
 - **KookaburraTeacher.KookaburraTeacher:** contém classes e métodos que definem o aplicativo para Windows Phone 7
 - **KookaburraTeacher.KookaburraServiceReference:** contém classes para conexão com Serviços Web.
 - **KookaburraTeacher.Pages:** contém classes e métodos que representam cada uma das telas do aplicativo
- **Bing**
 - **BingTranslateLibrary.BingTranslateLibrary:** contém as classes e métodos necessários para a tradução.
- **WFC Service**
 - **KookaburraWCFService.KookaburraDataService:** contém as classes e métodos necessários para implementar a camada de acesso ao banco

4.3. CAMADA DE APRESENTAÇÃO

Na camada de interface, utilizamos os Data Bindings do Silverlight. O Data Binding fornece uma maneira simples de exibir e interagir com os dados, separando a gestão dos dados da forma como eles são apresentados. Essa tecnologia ajudou muita na criação de interfaces mais ricas.

O *design* de interface dos aplicativos foi baseado em 3 (três) princípios básicos:

1. Sobriedade;
2. Clareza;
3. Facilidade de uso

Procurou-se explorar as possibilidades oferecidas pelo sistema operacional, para criar uma experiência clara e eficiente para os mais diferentes tipos de usuários, a fim de universalizar a utilização da aplicação.

O sistema usa de features como *Live Tiles*, *Push Notifications* e Calendários integrados para manter o estudante informado sobre o que está acontecendo nas aulas que segue.

A interface Metro oferece uma sobriedade e clareza fundamentais para o Kookaburra e o uso das *Accent Colors* permite que o usuário tenha um nível de customização sobre o sistema.

5. CONCLUSÃO E TRABALHOS FUTUROS

Durante o desenvolvimento do projeto a equipe teve a oportunidade de pesquisar e aprender mais sobre as tecnologias utilizadas, pois são as tecnologias que pretendem ter *expertise* na conclusão deste projeto.

A plataforma *Windows Azure* se mostrou eficaz na questão de tornar o aplicativo interoperável, ou seja, futuramente pode-se ter versões para outras plataformas utilizando o mesmo serviço. Teria o esforço de adaptar e implementar somente a camada de interface para outros dispositivos.

Algumas funcionalidades não ficaram na qualidade esperada devido a falta de tempo. *Features* como o aluno poder dar *feedback* sobre as aulas. Transmissão *real-time* do audio de uma aula nas chamadas *Live Lessons*.

Também poderia aperfeiçoar alguns pontos como utilizar um serviço de transcrição de audio de melhor qualidade, ou na interface do tradutor de Libras, apresentar a letra correspondente ao sinal apresentado.

6. REFERÊNCIAS

- Project Hawaii. (2010). Acesso em 15 de Junho de 2012, disponível em Microsoft Research: <http://research.microsoft.com/en-us/projects/hawaii/default.aspx>
- Cambiucci, W. (Maio de 2011). Introdução sobre o Windows Azure. Acesso em 14 de Junho de 2012, disponível em Site do MSDN: <http://msdn.microsoft.com/pt-br/library/hh150078.aspx#ref>
- Choudhary, V. (Janeiro de 2007). Software as a Service: Implications for Investment in Software Development, IEEE IT Professional. HICSS, p. 209.
- Melo, C. e. (2007). Software como Serviço: Um Modelo de Negócio Emergente. UFPE.
- Pedrosa, Paulo H. C., & Nogueira, T. (2010). Computação em Nuvem. Enacomp.
- R. C. Sousa, F., O. Moreira, L., F. de Macêdo, J. A., & C. Machado, J. (2010). Gerenciamento de Dados em Nuvem: Conceitos, Sistemas e Desafios. SWIB.
- Ramos, P. H., & H. de Farias Junior, I. (s.d.). Os Desafios da Computação em Nuvem como Serviço. Faculdade Joaquim Nabuco.
- Sousa, F. R. (2009). Computação em Nuvem: Conceitos, Tecnologias, Aplicações e Desafios. ERCEMAPI.
- Uebe Mansur, A. F., Silva Gomes, S., Moraes de Almeida Lopes, A., & Villanova Biazus, M. C. (2010). Novos rumos para a Informática na Educação pelo uso da Computação em Nuvem (Cloud Education): Um estudo de Caso do Google Apps. Rio de Janeiro: UFRGS.
- MATEUS, M., & BRITO, G. (7 de Novembro de 2011). CELULARES, SMARTPHONES E TABLETS NA SALA DE AULA: COMPLICAÇÕES OU CONTRIBUIÇÕES? X Congresso Nacional de Educação - Educere.
- Pesquisa IBOPE. (2009). O USO DOS COMPUTADORES E DA INTERNET NAS ESCOLAS PÚBLICAS DE CAPITALS BRASILEIRAS. Estudo realizado pelo IBOPE Inteligência e pelo LSI-Tec sob encomenda da Fundação Victor Civita.
- SANTOS, J. (2009). AS POTENCIALIDADES DO COMPUTADOR PORTÁTIL PARA A EDUCAÇÃO: PROGRAMA UM COMPUTADOR POR ALUNO (UCA). UNIVERSIDADE FEDERAL DA BAHIA.
- Ribeiro, S. L. (2008). A interface acessibilidade e educação inclusiva.

- MORAN, J. M. A integração das tecnologias na educação, 2007 Acesso em 07 de Junho de 2012, disponível em: <http://www.eca.usp.br/moran/integracao.htm>
- Alisandra, F., Castro Filho, J., Freire, R., & Lima, L. L. (2008). Objetos de Aprendizagem na Escola: Estudo de um Modelo de Implementação.
- Buckingham, D. (2008). Aprendizagem e Cultura Digital.
- Garofolo, J., Auzanne, C., & Voorhess, E. (2000). The TREC Spoken Document Retrieval Track: A Success Story.
- José, R. E. (1997). Informática e educação.
- Ramos, C. R. (2004). LIBRAS: A Língua de Sinais dos Surdos Brasileiros
- Silva, G. (2005). A tecnologia como um problema para a teoria da educação.
- Bing Translator. (2009). Acesso em 26 de Outubro de 2012, disponível em Bing Translator <http://blogs.msdn.com/b/translation/archive/2008/09/08/translation-powered-by-microsoft-translator.aspx>

7. GLOSSÁRIO

Data Bindings

É uma tecnologia do Silverlight que fornece uma maneira simples de exibir e interagir com os dados, separando a gestão dos dados da forma como eles são apresentados. Essa tecnologia ajudou muita na criação de interfaces mais ricas.

Desktop

São os computadores de mesa, estes que tem tudo separado (mouse, teclado, gabinete, monitor, impressora...), e são os de menor custo.

Live ID

É um serviço unificado de autenticação de usuários criado e mantido pela Microsoft que permite aos usuários se "logarem" a vários websites usando uma única conta.

Live Tiles

A tela inicial do Windows Phone 7 é composta por "mosaicos dinâmicos" que são atalhos para aplicações, funções, recursos e itens individuais (como contactos, páginas de Internet, aplicações ou itens de multimédia), que os utilizadores podem adicionar, rearranjar ou remover. Eles são dinâmicos e atualizam-se em tempo real – por exemplo, o mosaico de uma conta de email mostra o número de mensagens por ler ou um mosaico mostra uma atualização em tempo real do estado do tempo.

Portal Web

Um portal é um site na internet que funciona como centro aglomerador e distribuidor de conteúdo para uma série de outros sites ou subsites dentro, e também fora, do domínio ou subdomínio da empresa gestora do portal.

Push Notifications

Dão ao usuário final vantagens como poder verificar as novidades sobre seu aplicativo sem que precise estar o utilizando.

Silverlight

O Silverlight é um plug-in gratuito da plataforma .NET Framework compatível com múltiplos navegadores, dispositivos e sistemas operacionais, trazendo um novo nível de interatividade em qualquer lugar onde funcione a Web.

Smartphone

Aparelho com funcionalidades avançadas que podem ser estendidas por meio de programas executados no seu Sistema Operacional. É um aparelho celular que engloba muitas das principais tecnologias de comunicação e serviços que temos num computador ou em outros dispositivos, como acesso à E-mails, mensagens instantâneas (como o MSN), internet, GPS, entre outros. Os Smartphones possuem diversos meios de se comunicar com outros dispositivos através de Bluetooth, infravermelho e internet, com a finalidade de transferir arquivos de vídeo, fotos, música, planilhas, ou até mesmo sincronizar informações com outros programas no computador ou em outros Smartphones.

Stream

Pode ser definido como um fluxo de dados em um sistema computacional.

WCF Data Services

Permitem a criação e o consumo de serviços OData para a Web (anteriormente conhecido como ADO.NET Data Services).

Web Service

Solução utilizada na integração de sistemas e na comunicação entre aplicações diferentes. Com esta tecnologia é possível que novas aplicações possam interagir com aquelas que já existem e que sistemas desenvolvidos em plataformas diferentes sejam compatíveis. Os *Web services* são componentes que permitem às aplicações enviar e receber dados em formato XML. Cada aplicação pode ter a sua própria "linguagem", que é traduzida para uma linguagem universal, o formato *XML*.