**Sumário**

[INTRODUÇÃO 4](#_Toc495854461)

[1.1 Contexto 4](#_Toc495854462)

[1.2 Objetivos 4](#_Toc495854463)

[1.3 Justificativa 5](#_Toc495854464)

[1.4 Estrutura do trabalho 5](#_Toc495854465)

[2 COMPUTAÇÃO EM NUVEM 6](#_Toc495854466)

[2.1 Modelos de implantação 7](#_Toc495854467)

[2.1.1 Nuvem Pública 7](#_Toc495854468)

[2.1.2 Nuvem Privada 7](#_Toc495854469)

[2.1.3 Nuvem Comunitária 7](#_Toc495854470)

[2.1.4 Nuvem Híbrida 8](#_Toc495854471)

[2.2 Características de computação em nuvem 8](#_Toc495854472)

[2.2.1 Auto-Serviço Sob Demanda 8](#_Toc495854473)

[2.2.2 Elasticidade Rápida 9](#_Toc495854474)

[2.2.3 *Pool* de Recursos 9](#_Toc495854475)

[2.2.4 Mobilidade 9](#_Toc495854476)

[2.2.5 Monitoramento 9](#_Toc495854477)

[2.3 Modelos de serviços 10](#_Toc495854478)

[*2.3.1* *Software as a Service (SaaS)* 10](#_Toc495854479)

[*2.3.2* *Platform as a Service (PaaS)* 10](#_Toc495854480)

[2.3.3 *Infrastructure as a Service (IaaS)* 11](#_Toc495854481)

[3 MICROSOFT AZURE 12](#_Toc495854482)

[3.1 Uso comercial 12](#_Toc495854483)

[3.2 Serviços 12](#_Toc495854484)

[4 TECNOLOGIAS MÓVEIS 13](#_Toc495854485)

[4.1 Sistemas operacionais mobile 14](#_Toc495854486)

[*4.1.1* *IOS* 15](#_Toc495854487)

[*4.1.2* *WINDOWS PHONE* 15](#_Toc495854488)

[*4.1.3* *ANDROID* 15](#_Toc495854489)

[*4.2* *User experience e user interface* 16](#_Toc495854490)

[*4.2.1* *User Interface* 16](#_Toc495854491)

[*4.2.2* *User Experience* 18](#_Toc495854492)

[*4.3* Desenvolvimento *mobile e cross-plataform* 18](#_Toc495854493)

[5 SONAR DE SERVIÇOS 19](#_Toc495854494)

[5.1 Mini mundo do projeto 19](#_Toc495854495)

[5.2 Análise de requisitos 20](#_Toc495854496)

[5.2.1 Requisitos Funcionais 20](#_Toc495854497)

[5.2.2 Requisitos não Funcionais 21](#_Toc495854498)

[5.2.3 Regras de Negócio 21](#_Toc495854499)

[5.3 Modelo de Casos de Uso 22](#_Toc495854500)

[5.4 Modelo de Classes 22](#_Toc495854501)

[5.5 Modelo Lógico de Banco de Dados 22](#_Toc495854502)

[5.6 Futuro do Projeto 23](#_Toc495854503)

[6 CONCLUSÃO 23](#_Toc495854504)

[Bibliografia 24](#_Toc495854505)

INTRODUÇÃO  
 Pensar em um mundo sem internet hoje seria um exercício mental quase impossível, pois a facilidade que esta tecnologia nos proporciona foi tomando cada vez mais espaço no cenário da sociedade atual, passando de um bem de consumo para uma necessidade diária. Planejar suas viagens, realizar compras, evitar filas bancárias, entre muitas coisas são necessárias graças aos serviços oferecidos através da rede mundial de computadores. Hoje com o advento da computação em nuvem, ficou cada vez mais fácil para empresas e pessoas desenvolverem suas soluções sem se preocupar com infraestrutura e disponibilidade do serviço, delegando esta responsabilidade a empresas que se prontificam a realizar este trabalho. A proposta deste trabalho é conhecer um pouco mais da computação em nuvem, de seus aspectos e desenvolver uma aplicação a partir dela.

## Contexto

O contexto do presente trabalho é a demonstrar a construção de uma aplicação que uma a facilidade proporcionada pelo ambiente em nuvem, neste caso a plataforma *Azure* e o poder de desenvolvimento de aplicações versáteis e multiplataformas proporcionado pela linguagem de programação C#, através da interface de desenvolvimento *Visual Studio.*

## Objetivos

O objetivo deste trabalho é o desenvolvimento de uma aplicação para cadastro de clientes e prestadores de serviços de mecânica e borracharia, focada em computação em nuvem *Microsoft® Azure*, no qual o cliente efetua uma busca baseada em sua localização e a aplicação exibe os prestadores dos serviços cobertos em um raio inicial de cinco quilômetros, permitindo aumento pelo usuário.

A aplicação utilizará cadastro de interesses para manter controle de acessos por local, além de contar com uma modalidade de avaliação para que clientes possam avaliar prestadores de serviços e os prestadores de serviços replicarem a avaliação do cliente.

## Justificativa

A ideia surgiu da observação de que, para alguns tipos de trabalhos, ainda falta um ambiente de concentração de seus profissionais. Já existem aplicações que fazem o mesmo com transporte e alimentação, porém basta furar um pneu no meio da estrada, que descobrimos que alguns serviços não carecem desta facilidade.

## Estrutura do trabalho

O presente trabalho está estruturado em introdução, que discorre um pouco sobre o projeto, computação em nuvem, que dá um panorama sobre o conceito e suas formas, *Microsoft® Azure*, que apresenta um panorama sobre a tecnologia utilizada pelo trabalho, desenvolvimento da aplicação, que contará com mini mundo do projeto, modelagem , codificação e testes do software a ser desenvolvido, além de idéias para o futuro do projeto, assim como uma conclusão acerca do trabalho.

# COMPUTAÇÃO EM NUVEM

Computação em nuvem, do inglês *cloud computing* é o termo usado para definir um modelo de compartilhamento de recursos acessíveis de qualquer lugar, que possam ser ajustados conforme a demanda, com mínimo esforço de manutenção por parte do provedor [NIST, 2011]. Plataformas de desenvolvimento de *software* e banco de dados, armazenamento de arquivos, infraestruturas de rede, servidores de impressão, máquinas virtuais, entre outros, são oferecidos ao usuário como serviços, como mostra a figura 1, bastando apenas o usuário acessá-los e operá-los de qualquer computador com um navegador e acesso à internet, sem necessidade de programas ou recursos adicionais.

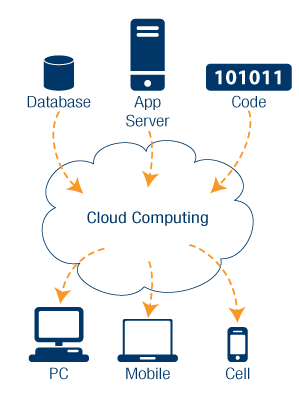


Figura 1: Modelo de computação em nuvem

Fonte: www.infoescola.com

## Modelos de implantação

A computação em nuvem possui modelos de implantação, que predizem a localização dos recursos e quem terá controle e acesso sobre eles, como é mostrado na figura 2. São eles, a nuvem pública, nuvem privada, nuvem híbrida e nuvem comunitária, e serão abordados a seguir.

### Nuvem Pública

Modelo onde os recursos são compartilhados e geralmente são oferecidos por provedores que fornecem soluções para empresas e pessoas, com restrição a recursos de acordo com privilégios de usuário, por isso, geralmente a nuvem pública está em conjunto com um sistema de autorização. A nuvem pública tem como vantagem os recursos compartilhados, o que pode proporcionar um custo menor a empresas com necessidades baixas de recursos.

### Nuvem Privada

Ao contrário da nuvem pública, o modelo de nuvem privada não é compartilhado, ou seja, é constituído pelo uso de servidores dedicados a empresas ou pessoas físicas, situando-se em instalação própria ou provedor de terceiros. A vantagem do modelo de nuvem privada é que o controle do crescimento é do próprio usuário, porém pode significar existência de recursos em demasia, dependendo da demanda da empresa ou pessoa.

### Nuvem Comunitária

No modelo de nuvem comunitária, organizações com interesses em comum se unem para compartilhar a infraestrutura. Sua administração ocorre por meio da própria comunidade ou terceiros existentes, sejam dentro ou fora da comunidade.

### Nuvem Híbrida

No modelo de nuvem híbrida existe a junção de dois ou mais modelos de implantação de nuvem, não sendo somente pública ou privada ou comunitária, mas mesclando características desses modelos. Nuvens híbridas são agrupadas por tecnologia padrão ou proprietária, permitindo portabilidade de aplicativos e dados, porém ainda são vistas como entidades únicas.

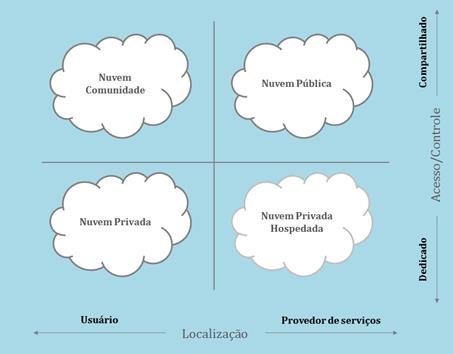


Figura 2: Modelos de implantação de nuvem

Fonte: teleco.com.br

## Características de computação em nuvem

Dentre as características da computação em nuvem, encontram-se o auto-serviço sob demanda, a elasticidade rápida, o *pool* de recursos, a mobilidade o amplo acesso e o monitoramento de serviços.

### Auto-Serviço Sob Demanda

O auto-serviço sob demanda, do inglês *On-Demand Self-Service* é a capacidade dos recursos serem adicionados ou removidos conforme necessidade do cliente, sem para isso a interação humana com o provedor.

### Elasticidade Rápida

A elasticidade rápida, do inglês *Rapid Elasticity* é característica de alocar mais ou menos recursos instantaneamente conforme seja necessário, dando a impressão ao cliente de que a nuvem é um espaço infinito de processamento e armazenamento.

### *Pool* de Recursos

O *Pool* de recursos, do inglês *Resource Pooling* é a característica de divisão de recursos computacionais do provedor para múltiplos consumidores, usando um modelo *multi-tenant*, descrito no item 2.3, habilitando e reabilitando de acordo com a demanda do cliente. O cliente não possui controle de onde exatamente estão alocados seus recursos, porém pode especificar o país, estado ou centro de dados em alto nível de abstração.

### Mobilidade

A mobilidade é a capacidade dos serviços da nuvem ser acessados de qualquer lugar, bastando somente uma conexão de banda larga e um navegador de internet. Por não ser necessário um sistema ou aplicação própria, o cliente pode acessar os serviços através de *smartphones*, *tablets, lan houses,* televisores *smart,* entre outros dispositivos que possuam acesso à internet.

### Monitoramento

O Monitoramento de serviços é a capacidade de controlar a utilização dos recursos da nuvem, medindo a utilização de forma automática e controlando-os de forma transparente, ou seja, o fornecedor e o cliente possuem total conhecimento da utilização dos recursos.

## Modelos de serviços

Os modelos de serviços definem a divisão de responsabilidades entre cliente e provedor, delimitando onde iniciam e terminam as responsabilidades de cada um dos envolvidos. Conforme apresentado na figura 3, cada modelo atende a um tipo diferente de propósito, sendo três os modelos de serviços definidos pela *National Institude of Standarts and Technology - NIST* (2011), o PaaS, o Saas e o IaaS, que serão descritos a seguir.

### *Software as a Service (SaaS)*

Software como um serviço é um modelo de disponibilização de sistemas de propósitos específicos armazenados e executados sobre a infraestrutura do provedor, sendo acessíveis através de um navegador *WEB* ou por uma interface de programação da aplicação (API – *Application Program Interface*). No SaaS, o usuário não possui controle sobre a infraestrutura, exceto configurações específicas de domínio do usuário. Sistemas operacionais, armazenamento, rede, entre outros, são de controle do provedor do serviço. Exemplos de serviços neste modelo são o *Google Docs* e o *Office Web Apps.*

### *Platform as a Service (PaaS)*

Plataforma como um serviço é um modelo de disponibilização de ambientes de programação, testes e hospedagem de aplicações desenvolvidas pelo usuário na infraestrutura do provedor, onde o usuário possui um ambiente propício para desenvolver suas aplicações, sem a preocupação com infraestrutura e *softwares de desenvolvimento e gestão de banco de dados. A* plataforma *Microsoft Azure* é um exemplo deste modelo (*Microsoft*, 2012).

### *Infrastructure as a Service (IaaS)*

Infraestrutura como um serviço é um modelo de disponibilização de recursos de infraestrutura fundamentais, tais como servidores, rede, entre outros, onde o usuário pode instalar sistemas operacionais e aplicativos de forma arbitrária. oIaaS possui uma interface única para administração da infraestrutura, API para interação com *hosts, routers e switchs,* adição de novos equipamentos de forma simples. O controle da infraestrutura no modelo IaaS não fica nas mãos do usuário, porém o mesmo possui controle sobre os sistemas operacionais, aplicativos e armazenamento implantados, também a seleção eventual de componentes de rede, porém o usuário é responsável pela escalabilidade no modelo, ao invés do provedor.

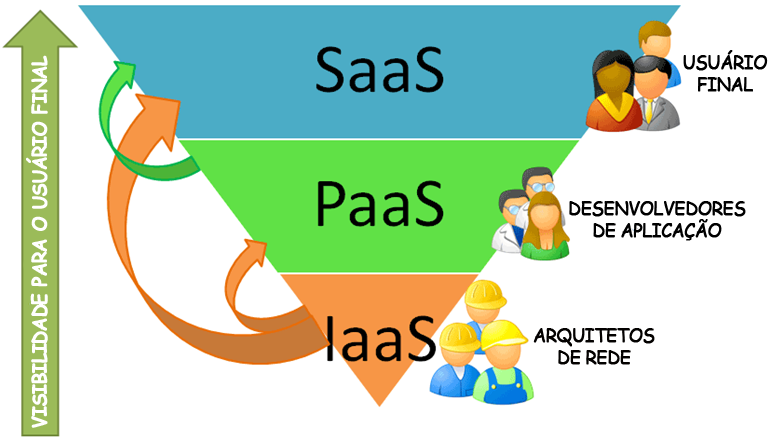


Figura 3: Serviços da nuvem

Fonte: vitormeriat.com.br

# MICROSOFT AZURE

A *Microsoft Azure* é uma plataforma destinada à execução de aplicativos e serviços, baseada nos conceitos da computação em nuvem. Ela trabalha com o *Windows Azure* que é uma espécie de sistema operacional que gerencia todo e qualquer recurso empregado dentro dele, juntamente com seu banco SQL *Azure* e seu *middler* de integração *AppFabric*.

## Uso comercial

Hoje em dia, a *Microsoft Azure* é utilizada para fins comerciais por empresas que buscam serviços em nuvem tanto para demandas internas, quanto para demandas externas, pelo fato de possuir grande abrangência de sistemas que podem ser desenvolvidos, assim como linguagens que a plataforma aceita em seus servidores. O sistema conta com um gerenciador, que é o próprio *Microsoft Azure*, conforme supracitado, funcionando com máquinas virtuais, onde pode aplicar qualquer sistema operacional. Desta forma permite que o usuário migre para a plataforma Azure, utilizando os meios que ele acha mais seguro para manter seu sistema.

## Serviços

A *Microsoft Azure* trabalha hoje com serviços de *Cloud Computing* e *Data Centers* que conseguem atender os clientes, e, também conta com suporte aos principais *frameworks* de desenvolvimento, assim como os principais bancos de dados para hospedagem em seus servidores. São exemplos de linguagens suportadas pela *Microsoft Azure*: C#, *Visual Basic* .*NET*, *ASP .NET* *Java*, *Phyton*, PHP e *Node*.*js*. O sistema conta também com máquinas virtuais em *Windows* e *Linux*, conforme supracitado para atender as plataformas que forem solicitadas à seu serviço. O sistema também conta com a facilidade da escalabilidade da demanda para facilitar a compra dos serviços de máquinas em *cloud*, por exemplo: Se você possui um servidor com certo limite de memória, e precisar aumentar sua capacidade, você poderá ficar despreocupado que o sistema *Microsoft Azure* faz o ajuste em cima do gasto de processamento que você poderá vir a ter.

A *Microsoft Azure*, hoje em dia, conta com soluções de hospedagem e gerenciamento de software para pequenos e grandes aplicativos, assim como para pequenos aplicativos, em outras palavras, a *Microsoft Azure* consegue lidar com grande parte dos aplicativos que possam ser hospedados em seu serviço. A *Azure*, também possui soluções para serviços com distribuição global, como é o caso do seu BD *Cosmos*. O Serviço também conta com um serviço de *Backup* que oferece hospedagem dos *backups* em servidores online e também, *backups* em fitas conforme necessidade de algumas empresas de guardar dados durante anos.

# TECNOLOGIAS MÓVEIS

Tecnologias móveis são aquelas que permitem a utilização durante a movimentação de seu usuário. O conceito foi apresentado na década de 40, mas devido às limitações da época não teve nenhum avanço prático, limitando-se à criação de teorias. Em 1973, por intermédio das primeiras ligações efetuadas por um dispositivo móvel e o surgimento dos primeiros conceitos sobre o celular, as possibilidades começaram a ser discutidas novamente.

Os primeiros aparelhos de uso pessoal eram limitados, possibilitando somente ligações. Foram as gerações seguintes que começaram a apresentar uma gama de serviços e novas possibilidades, tais como, envio de mensagens de texto (SMS), telas coloridas e a agregação de serviços multimídia.

O próprio conceito de "celular" foi abandonado e por sua vez substituído pela alcunha de *smartphones*. Os aparelhos antes de uso limitado se destacam agora pela robustez de hardware integrado e largo uso de software, vindo a possuir até variações de sistemas operacionais, agregando muito mais possibilidades para os usuários e desenvolvedores de conteúdo.

As tecnologias móveis são abrangentes. Muitos setores já investem no mercado *mobile*, não só na parte de telefonia, mas também em conjunto com eletrodomésticos e até mesmo em carros.

## Sistemas operacionais mobile

O sistema operacional é o mais importante componente de dispositivos móveis. Sendo o responsável pela integração entre as vontades do usuário e os recursos de hardware, para que um determinado aplicativo realize uma tarefa, ou que múltiplos serviços estejam em uso simultâneo eficientemente. Suas diferenças começam já nos layouts iniciais, como mostra a figura 4. Cada sistema possui seus pontos fortes e fracos, podem ser otimizados para uso em hardware específico, possuir uma gama maior e mais versátil de aplicativos ou focar na simplicidade e uso mais ágil. Serão apresentados agora, alguns dos principais sistemas utilizados em *smartphones*.

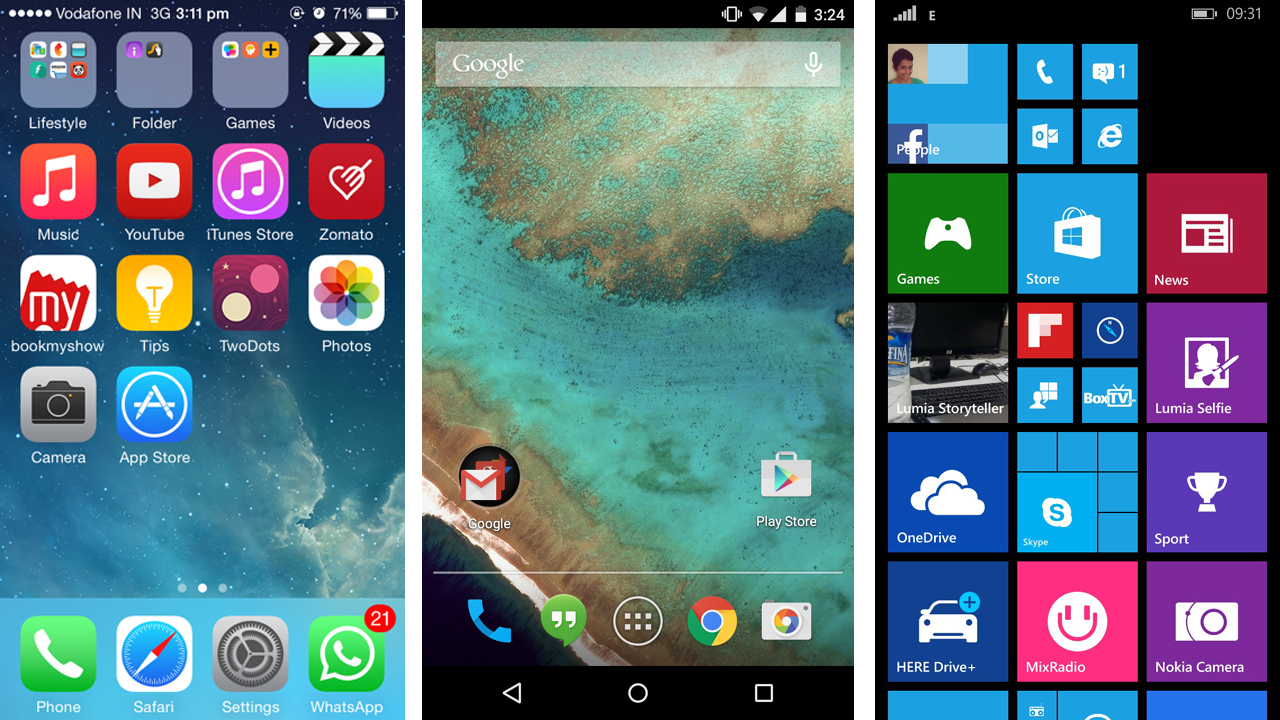


Figura 4 - Os Layouts de tela dos três principais concorrentes do mercado *Mobile*: IOS, Windows Phone e Android. Fonte: firstpost.com

### *IOS*

Lançado em 2007 pela Apple, sendo criado para os aparelhos da linha *iPhone*, integrando os *tablets* *iPad* posteriormente. Possui muita fluidez, fruto do aproveitamento de *hardware* dedicado. Seus aplicativos nativos e de terceiros são baixados e instalados somente pela loja oficial, a *Apple Store*, conferindo certa segurança aos mesmos. O visual é semelhante às versões *desktop* do OS. Atualmente está se encaminhando para a versão 11.

### *WINDOWS PHONE*

Sistema idealizado pela Microsoft e baseado no *Kernel* do *Windows* CE6. Foi lançado em 2010, sucedendo o *Windows Mobile*. Possui alta compatibilidade com os computadores *Windows,* sendo empregado em sua maioria em aparelhos da própria Microsoft. Também possui loja oficial para aplicativos, atualmente na versão 10.

### *ANDROID*

Criado pela *Android Inc* e baseado no *Kernel Linux*, com diversos componentes e bibliotecas, sendo usado e posteriormente adquirido pela Google. É disponibilizado sobre licença de código aberto, mas devido à facilidade de personalização é geralmente combinado com *software* privado, possibilitando interfaces diferentes, algumas vistas na figura 5. Possui a maior gama de aplicativos em sua loja e possibilita também a instalação de programas de terceiros. Atualmente é o mais difundido entre usuários e desenvolvedores, chegando à versão 8.



Figura 5 - Diferentes personalizações de tela do Android. Fonte: techtudo.com.br

## *User experience e user interface*

Estes termos se aplicam também a dispositivos móveis e tratam da experiência do usuário com a plataforma em questão.

### *User Interface*

A interface com o usuário basicamente inclui a aparência do sistema, seus elementos visuais, botões e ícones, ou seja, tudo o que é utilizado para interagir com um dispositivo. A figura 6 mostra as diferentes interfaces de acesso inicial dos três principais sistemas.

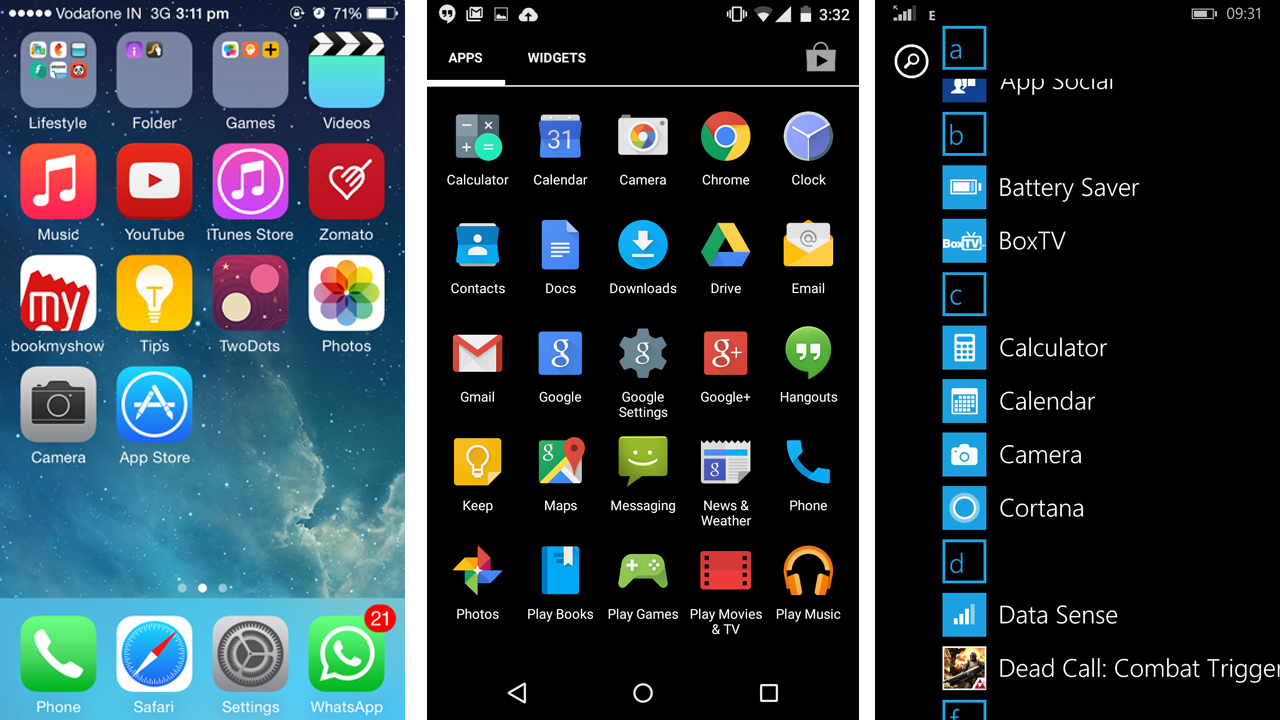


Figura 6 - As interfaces de acesso e uso dos apps nos sistemas IOS, Android e Windows Phone. Fonte: firstpost.com

### *User Experience*

A experiência do usuário trata dos aspectos da interação a um nível de satisfação com os serviços oferecidos. Foca o processo de resolução de um problema ou solicitação. A figura 7 da uma idéia inicial dos recursos de interação com as opções de configuração iniciais dos sistemas mobile, fator primordial para melhor aproveitamento no uso.

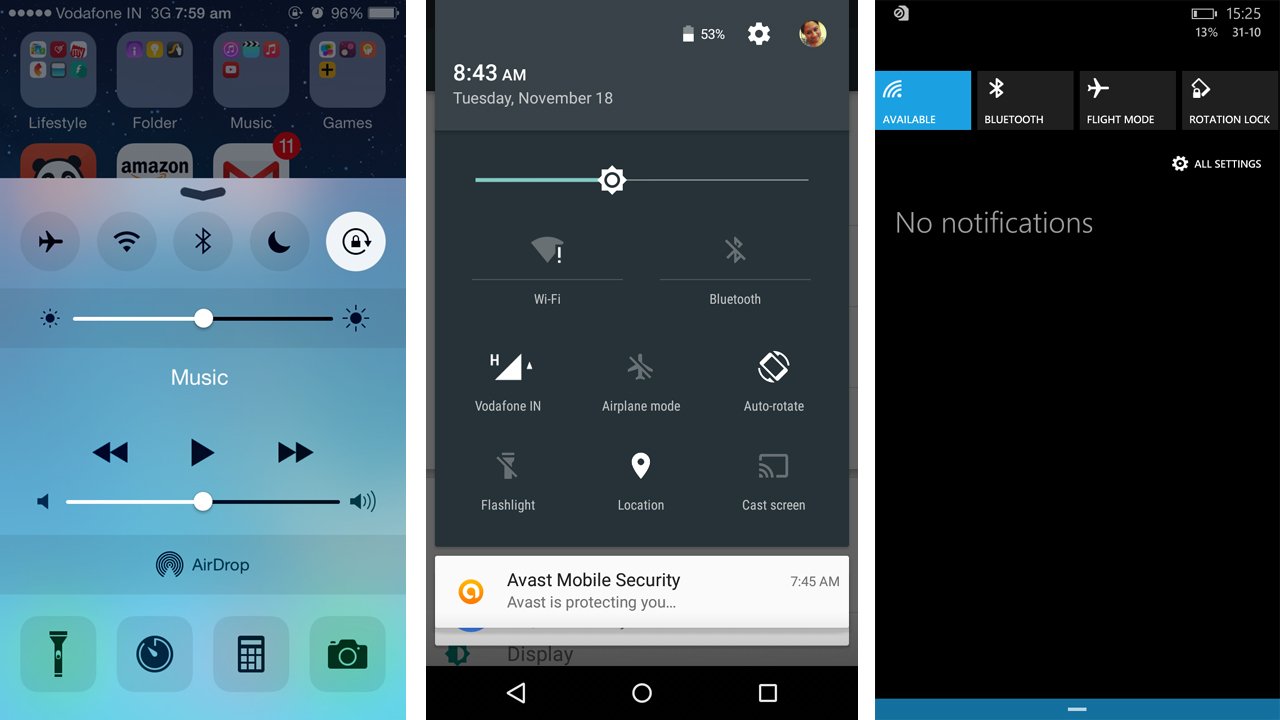


Figura 7 - Apresentação dos acessos básicos de configuração dos sistemas IOS, Android e Windows Phone. Fonte: firstpost.com

## Desenvolvimento *mobile e cross-plataform*

Dadas as estruturas de concepção, a construção e arquitetura de um OS móvel, possui diferentes recursos para o desenvolvimento de aplicações. Com tantas particularidades o ambiente móvel se mostrou promissor para a integração de *Cross-Plataform*. Á compatibilidade gerada através dessas ferramentas permite a flexibilidade no desenvolvimento de aplicativos para várias plataformas ao mesmo tempo. Uma aplicação pode ter versões para várias plataformas, como mostra a figura 8 e o desenvolvimento pode ocorrer ao mesmo tempo.

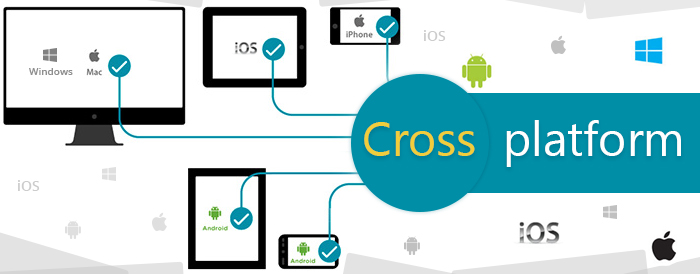


Figura 8 - Ilustração do conceitro de Cross Plataform. Fonte: purelogics.net

# SONAR DE SERVIÇOS

Sonar de serviços será uma aplicação ambientada em nuvem, que realizará o cadastro de prestadores de serviços, que poderão ser buscados por clientes cadastrados no sistema ou não.

## Mini mundo do projeto

A proposta é a criação de um aplicativo de execução em ambiente de nuvem que permita que prestadores de serviços de mecânica e borracharia se cadastrem como pessoa física ou jurídica, permitindo que clientes que busquem pelos serviços oferecidos nas proximidades os encontrem e os contatem.

O sistema deve contar com um formulário para cadastro de prestadores de serviços, que colete dados como Nome ou Razão Social, CPF ou CNPJ, endereço residencial / comercial, Telefones, e-mail, valor do orçamento para clientes nas proximidades, serviços prestados, informações complementares, além de login e senha do sistema.

O sistema também deve contar com um formulário para cadastro de clientes, que colete dados como Nome ou razão social, CPF / CNPJ, endereço, telefones, e-mail, login e senha.

Deverão ser aceitos clientes sem cadastro, porém um usuário cadastrado só poderá fazer pesquisas em um raio de 5 (cinco) kilômetros, sendo necessário um registro para alterar o raio de busca.

O cliente poderá realizar uma busca digitando um endereço ou utilizando coordenadas de GPS do *smartphone*.

Deverá haver uma funcionalidade que permita que clientes avaliem serviços feitos por prestadores, além de réplica opcional pelo prestador de serviços. O sistema de avaliação contará com uma nota de 1 (hum) até 5 (cinco), além de uma descrição da avaliação, opcional pelo cliente.

O telefone e endereço do prestador de serviço buscado deverão ficar ocultos, até que o cliente clique em um botão que revele os dados, que crie um registro de interesse do cliente ao prestador, para estatísticas do aplicativo, que poderá ser desenvolvido em versões futuras.

O sistema deve permitir ao cliente visualizar prestadores de serviço por pontos no mapa ou por uma lista, em ordem de proximidade.

## Análise de requisitos

A Análise de requisitos do sistema contará com requisitos funcionais, requisitos não funcionais e regras de negócios, demonstrados a seguir.

### Requisitos Funcionais

RF01 – O cliente pode realizar o cadastro no sistema

RF02 – O cliente pode logar-se no sistema

RF03 - O cliente poderá avaliar o serviço executado por um prestador que demonstrou interesse.

RF04 – O cliente que possui cadastro como prestador de serviço pode alterar entre modos sem a necessidade de sair do sistema

RF05 – O Usuário deve poder ver os prestadores como uma lista

RF06 – O Usuário deve poder visualizar prestadores de serviço como pontos no mapa

RF07 – O prestador de serviço poderá responder a uma avaliação feita por um cliente

RF08 – O GPS deve poder obter a localização do usuário

RF09 – O usuário poderá digitar sua localização para uma busca

### Requisitos não Funcionais

RNF01 – O sistema será hospedado em nuvem *Microsoftr® azure*

RNF02 – O banco de dados será hospedado em nuvem *Microsoftr® azure.*

RNF03 – O banco de dados será desenvolvido a partir da plataforma *SQL Server*

RNF04 – O sistema será desenvolvido a partir da linguagem de programação C#.

RNF05 – A interface com o sistema será desenvolvida para dispositivos *Android*

RNF06 – Os dispositivos de interface deverão armazenar uma cópia do banco de dados da nuvem pelo internamente pelo *SQLite.*

### Regras de Negócio

RN01 – Serão permitidos somente cadastros com CPF ou CNPJ válidos

RN02 – O raio inicial de pesquisa é de 5 kilômetros

RN03 – Perfis distintos de um mesmo cadastro possuirão ids distintos

RN04 – Somente clientes cadastrados podem aumentar um raio de pesquisa

RN05 – Somente clientes cadastrados podem avaliar prestadores de serviços

## Modelo de Casos de Uso

## Modelo de Classes

## Modelo Lógico de Banco de Dados

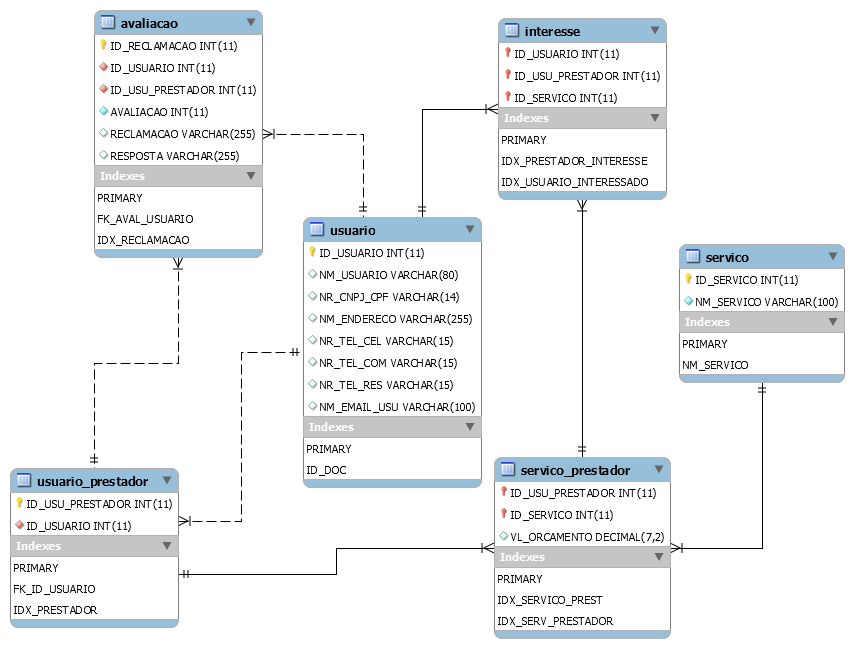


Figura 9 - Modelo lógico do banco de dados

## Futuro do Projeto

# CONCLUSÃO

# Bibliografia

DXC Technology*O Que é O Cloud Computing? | DXC Technology*

HAUTSCH, O. (13 de Novembro de 2008). *Windows Azure: seu computador vai às nuvens - TecMundo*. Acesso em 24 de Agosto de 2017, disponível em TechMundo: https://www.tecmundo.com.br/web/943-windows-azure-seu-computador-vai-as-nuvens.htm

Imamura, D. (19 de Maio de 2011). *Windows Azure: o que é?* Acesso em 24 de Agosto de 2017, disponível em Oficina da Net: https://www.oficinadanet.com.br/artigo/windows/windows-azure-o-que-e

Microsoft. (01 de Janeiro de 2017 ). *Microsoft Azure: Plataforma e serviços de computação em nuvem*. Acesso em 24 de Agosto de 2017, disponível em Microsoft: https://azure.microsoft.com/pt-br/

Pinheiro, F. (2010). *Cloud Computing.* Acesso em 21 de 08 de 2017, disponível em Universidade Federal do Rio de Janeiro: https://www.gta.ufrj.br/ensino/eel879/trabalhos\_vf\_2010\_2/fernando/index.html

The NIST Definition of Cloud Computing. (10 de 2011). *National Institute of Standards and Technology Special Publication 800-145* , p. 7.