

UNIVERSIDADE DE CAXIAS DO SUL
CENTRO DE COMPUTAÇÃO E TECNOLOGIA DA INFORMAÇÃO
BACHARELADO EM SISTEMAS DE INFORMAÇÃO

CARINA PONZONI PILAR

**AVALIAÇÃO DAS ARQUITETURAS DE
DESENVOLVIMENTO PARA DISPOSITIVOS MÓVEIS**

CAXIAS DO SUL

2013

CARINA PONZONI PILAR

**AVALIAÇÃO DAS ARQUITETURAS DE
DESENVOLVIMENTO PARA DISPOSITIVOS MÓVEIS**

Trabalho de Conclusão de Curso para
obtenção do Grau de Bacharel em Sistemas
de Informação da Universidade de Caxias
do Sul.

Daniel Antônio Faccin

Orientador

CAXIAS DO SUL

2013

Dedico este trabalho a todos os professores que fizeram parte desta caminhada, em especial ao Daniel Faccin, que esteve presente nesta jornada desde o curso técnico em desenvolvimento de sistemas no ensino médio, encerrando esta etapa da minha vida como meu orientador neste trabalho. Dedico também à minha família que me deu suporte e ao meu noivo que sempre esteve ao meu lado me motivando e apoiando.

RESUMO

Esta monografia é o resultado de um estudo sobre as tecnologias móveis, onde é abordado o histórico, a evolução, o posicionamento no mercado mundial e a comparação entre as quatro tecnologias mais utilizadas no desenvolvimento de aplicativos móveis. São os aplicativos *web* com HTML 5 e aplicativos nativos para Windows Phone, Android e iOS. Focado em demonstrar a rápida evolução dos dispositivos móveis, exigindo que as empresas estejam sempre atualizadas, o objetivo deste estudo é analisar as diferenças no desenvolvimento de aplicativos móveis através de uma arquitetura *web* e nativa.

Palavras chaves: Desenvolvimento. Tecnologias Móveis. Arquiteturas de Desenvolvimento.

LISTA DE FIGURAS

Figura 1 - Arquitetura do negócio (LEE, 2005)	27
Figura 2 - Arquitetura cliente-servidor (LEE, 2005).....	29
Figura 3 - Arquitetura de desenvolvimento móvel (CAO, 2012).....	30
Figura 4 - Arquitetura HTML 5 (HARRIS, 2011)	36
Figura 5 - Document Object Model (DOM).....	37
Figura 6 - Arquitetura Windows Phone (WHITECHAPEL e MCKENNA, 2013).....	41
Figura 7 - Arquitetura Android (LEE, 2012).....	45
Figura 8 - Arquitetura iOS (RAY, 2011).....	50
Figura 9 - Cartaz "Keep Calm" original	60
Figura 10 - Exemplos de imagens "Keep Calm"	60
Figura 11 - Tela inicial do aplicativo.....	64
Figura 12 - Tela de edição do texto	64
Figura 13 - Tela de captura de imagem	65
Figura 14 - Tela para alteração de imagem de fundo	65
Figura 15 - Tela para alteração da cor do fundo.....	66
Figura 16 - Tela para compartilhar e salvar a imagem	66
Figura 17 - Tela de ícones	67
Figura 18 - Tela de informações do aplicativo	67
Figura 19 - Comparativo da tela inicial do aplicativo	76
Figura 20 - Comparativo da tela de edição do texto.....	77
Figura 21 - Comparativo da tela de captura de imagem.....	77
Figura 22 - Comparativo da tela para alteração de imagem de fundo.....	78

Figura 23 - Comparativo da tela para alteração da cor do fundo	78
Figura 24 - Comparativo da tela para compartilhar e salvar a imagem.....	79
Figura 25 - Comparativo da tela de ícones	79
Figura 26 - Gráfico de horas de desenvolvimento.....	84

LISTA DE TABELAS

Tabela 1 - Evolução dos celulares (FLING, 2009).....	19
Tabela 2 - Comparação entre as tecnologias	54
Tabela 3 - Comparativo de características do aplicativo entre as tecnologias	83

SUMÁRIO

1. INTRODUÇÃO	10
1.1. Motivação	12
1.2. Problema de Pesquisa	13
1.3. Questão de Pesquisa.....	13
2. OBJETIVOS.....	14
2.1. Objetivo Geral.....	14
2.2. Objetivos Específicos	14
3. FUNDAMENTAÇÃO TEÓRICA	16
3.1. A evolução dos dispositivos móveis	16
3.2. Tecnologias móveis	20
3.3. M-Business	21
3.4. A economia móvel	22
4. ANÁLISE TÉCNICA	25
4.1. Características dos dispositivos móveis.....	25
4.2. Desenvolvimento para dispositivos móveis.....	26
4.3. Plataformas Móveis	34
4.3.1. Web: HTML 5.....	34
4.3.2. Windows Phone.....	38
4.3.3. Android.....	42
4.3.4. iOS.....	48
4.4. Comparação entre as tecnologias.....	53
4.5. Comparação entre aplicações móveis <i>web</i> e aplicações nativas.....	55
5. PROPOSTA DE SOLUÇÃO	59
5.1. O Aplicativo.....	59
5.1.1. Aplicativos existentes	61
5.1.2. Funcionalidades	62
5.1.3. Protótipos de telas.....	63
6. ANÁLISE DOS RESULTADOS.....	68

6.1.	Aplicativo <i>web</i>	68
6.1.1.	Análise do processo de desenvolvimento	68
6.1.2.	Execução do aplicativo	70
6.2.	Aplicativo para Windows Phone	72
6.2.1.	Análise do processo de desenvolvimento	72
6.3.	Aplicativo para Android	73
6.3.1.	Análise do processo de desenvolvimento	73
6.4.	Aplicativo para iOS	74
6.4.1.	Análise do processo de desenvolvimento	74
6.5.	Análise dos aplicativos	76
6.5.1.	Comparativo e análise das características	80
7.	CONCLUSÃO	86
8.	REFERÊNCIAS BIBLIOGRÁFICAS	88

1. INTRODUÇÃO

Pense em quarenta, trinta ou vinte anos atrás, onde os dispositivos móveis eram praticamente inexistentes, aqueles poucos que haviam sido fabricados tinham capacidade limitada e utilidade praticamente única. Agora pense atualmente, onde não se consegue imaginar uma vida sem estes aparelhos, sem comunicar-se, carregar arquivos digitais ou levar as músicas favoritas para ouvir em qualquer lugar. Especialmente, sente-se a necessidade de se manter conectado o tempo todo.

Oficialmente, o primeiro telefone móvel foi criado em 1956 pelas empresas suecas de telecomunicações, TeliaSonera e Ericsson. Muitos anos se passaram para que esta tecnologia se tornasse parte da vida das pessoas. A mobilidade é o ponto forte destes aparelhos, sendo esta característica composta pela portabilidade, usabilidade, funcionalidade e conectividade (LEE, 2005). A portabilidade é definida pela capacidade de transportar o dispositivo, uma característica que pode variar a sua percepção ao longo do tempo, o que é portátil hoje poderá se tornar mais ainda amanhã, como se percebeu na evolução do tamanho de celulares e computadores para *tablets*. Outra característica, a usabilidade, leva em consideração a utilização do dispositivo por diferentes tipos de pessoas. Abrange a facilidade física na utilização do aparelho, a sua durabilidade sob condições variadas, rapidez de processamento, manutenção dos dados e interface com o usuário. A funcionalidade refere-se aos múltiplos propósitos que o aparelho pode servir, seguindo seus objetivos e diferenciando-se pelos aplicativos desenvolvidos para cada plataforma. E por fim, a conectividade, funcionalidade primordial dos dispositivos móveis, que deverá ser capaz de conectar pessoas e sistemas, transmitindo e recebendo informações praticamente de forma contínua (LEE, 2005).

Os aparelhos móveis começaram a surgir com as mais diversas funcionalidades e a mobilidade foi um dos principais atrativos para a imensa evolução na comercialização destes aparelhos. Nos anos 80 surgiu o Walkman da

Sony que virou mania mundial, suas vendas ultrapassaram 300 milhões de unidades. Em seguida vieram os videogames, o Gameboy da Nintendo em 1989, abrangendo um mercado mais jovem, com vendas de 120 milhões de unidades. Em 2003 houve a disseminação da utilização dos celulares, com vendas superiores a 200 milhões de unidades. Nesta mesma época surgiu o iPod da Apple, que reinventou o Walkman, conquistando a todos e obtendo vendas na casa dos 240 milhões de unidades.

Em 2007, marcado pelo lançamento do iPhone pela Apple, a indústria dos *smartphones* ganhou força e elevou ainda mais a venda de celulares, contabilizando 1,1 bilhão de aparelhos vendidos no mundo em 2009.

Não só os aparelhos em si que estão envolvidos nesta evolução, mas juntamente com eles vieram os serviços agregados. Utilizando o exemplo do iPod, que foi seguido por diversas gerações de iPhone's e iPad's, têm sua principal sobrevivência no iTunes, uma loja virtual de músicas, aplicativos, livros e etc. Serviço essencial para aqueles proprietários de aparelhos da Apple, com o objetivo de manter-se conectado e interagindo com o resto do mundo. Atualmente este serviço conta com mais de 75 milhões de usuários, 850 mil aplicativos e mais de 50 bilhões de *downloads*, um número realmente expressivo.

Outra grande mudança é na leitura digital, que foi impulsionada pela criação dos dispositivos chamados *e-readers*, pequenos aparelhos que leem livros digitais, os *e-books*. Utilizando uma tecnologia denominada tinta eletrônica, possui uma tela que simula a folha de papel, não forçando a visão dos leitores. Esta tecnologia, criada no final dos anos 90, já atingiu a venda de 50 milhões de unidades nos últimos anos. Empresas editoras de livros e revistas, assim como livrarias, tiveram que se adaptar para entrar neste novo ramo. Agora é possível levar prateleiras cheias de livros em apenas um dispositivo que pesa tanto quanto um pequeno livro, uma mudança de comportamento, uma quebra de paradigma.

Ao observar todos estes dados, percebe-se a grande evolução dos dispositivos móveis nas últimas décadas, sendo que com a crescente aceitação dos

consumidores, esta evolução tende a crescer cada vez mais, impulsionada pela necessidade de comunicação instantânea exigida pela globalização e o mundo dos negócios.

Este trabalho foi dividido em seis capítulos que, após a introdução e objetivos, inicia através de uma fundamentação teórica. Neste capítulo será abordada a evolução dos dispositivos móveis e tecnologias, incluindo também o mercado em que estão inseridos e como são utilizados.

No próximo capítulo será realizada a análise técnica dos dispositivos móveis. Serão expostas as características e meios de desenvolvimento de uma forma geral para implementações móveis. Neste capítulo serão detalhadas as tecnologias que abordadas no trabalho: HTML 5, Windows Phone, Android e iOS. Finalizando, são comparadas as características entre as plataformas e as formas de desenvolvimento *web* e nativa.

Em seguida, no capítulo sobre a proposta de solução, serão detalhadas as características do aplicativo que será desenvolvido. Como suas telas e as funcionalidades que serão analisadas na próxima etapa.

Finalizando o desenvolvimento da pesquisa, no capítulo sobre a análise dos resultados, serão comparados os aplicativos desenvolvidos e analisados os pontos que foram elencados no capítulo anterior.

Em seguida e finalizando o trabalho, serão listadas as referências bibliográficas.

1.1. Motivação

Atualmente vivemos em um mundo cheio de pequenos aparelhos digitais, aparelhos móveis que nos mantém conectados com o mundo, ligados o

tempo inteiro. Dediquei este trabalho para pesquisar mais sobre este fenômeno, explicando de que forma estes aparelhos se tornaram parte da nossa vida e como, do ponto de vista técnico, as empresas de desenvolvimento de *software* podem tirar proveito deste comportamento, entendendo melhor o processo de desenvolvimento de aplicativos móveis.

1.2. Problema de Pesquisa

Observa-se que os dispositivos móveis vêm alcançando cada dia mais avanços tecnológicos, conquistando uma abrangência mundial. Desta forma, é necessário que o processo de desenvolvimento de aplicativos para estes dispositivos tenha portabilidade, onde seja possível a reutilização de processos, garantindo agilidade e foco no seu desenvolvimento e adaptação aos objetivos de cada plataforma, para assim acompanhar a evolução e mudanças das tecnologias móveis.

Para isso, é importante conhecer as diferentes tecnologias e as arquiteturas disponíveis para o desenvolvimento de aplicativos móveis, para saber diferenciá-las e escolhê-las com base nos objetivos e disposições técnicas e financeiras da empresa.

1.3. Questão de Pesquisa

Com base no problema descrito anteriormente, foi criada a seguinte questão de pesquisa:

Como desenvolver um aplicativo móvel em uma arquitetura *web*, portátil para três diferentes tecnologias, sem perder funcionalidades em relação ao desenvolvimento na linguagem nativa de cada uma delas?

2. OBJETIVOS

Os objetivos deste trabalho são divididos em gerais e específicos, conforme explicado a seguir.

2.1. Objetivo Geral

O objetivo principal deste trabalho é desenvolver um aplicativo móvel em uma arquitetura *web*, que seja portátil para três diferentes plataformas móveis, e avaliar as diferenças obtidas em relação ao desenvolvimento na arquitetura nativa de cada uma.

2.2. Objetivos Específicos

Os objetivos específicos foram divididos em tópicos que exemplificam cada etapa da pesquisa a ser realizada.

- Identificar e conceituar dispositivos móveis, sua evolução e arquitetura de desenvolvimento;
- Conceituar e diferenciar o desenvolvimento de aplicativos *web* com HTML 5 e três tecnologias móveis, sendo elas: o Windows Phone da Microsoft, o Android da Google e o iOS da Apple;
- Identificar a arquitetura de desenvolvimento de cada tecnologia que será abordada e preparar os ambientes de desenvolvimento;

- Definir e desenvolver um aplicativo único para as três diferentes plataformas, seguindo uma arquitetura de desenvolvimento *web* e nativa;
- Analisar os resultados obtidos com o aplicativo e comparar as diferentes arquiteturas utilizadas, para identificar as diferenças de cada abordagem.

3. FUNDAMENTAÇÃO TEÓRICA

Neste capítulo será abordada a questão teórica da pesquisa. Abrangerá a história dos dispositivos móveis e suas tecnologias, assim como o mercado e a economia que utiliza estas ferramentas.

3.1. A evolução dos dispositivos móveis

Toda a tecnologia que conhecemos hoje partiu da necessidade da comunicação, da troca ou armazenamento de informações. Sendo que nos primórdios da história do desenvolvimento humano a tecnologia utilizada eram apenas pedras ou argila com marcas, escritas e números lapidados. Com o tempo, evoluímos para a utilização de papiros e canetas de tinta, desenvolvemos então uma forma de ler e escrever para nos comunicarmos, sendo esta ainda a tecnologia mais antiga e amplamente utilizada. Em uma evolução mais recente, surgiu um dispositivo móvel chamado relógio de bolso, com a sua utilização foi possível sincronizar as ações com outras pessoas em diferentes lugares. Ao mesmo tempo surgiram os livros impressos, com eles a divulgação do conhecimento ocorria de uma forma mais padronizada e de fácil acesso.

Até então os dispositivos móveis eram todos mecânicos e sem interatividade. Com o surgimento dos dispositivos eletrônicos nos anos 60, em meados dos anos 70 surgiram as calculadoras. Com um preço inicial de 300 dólares e um tamanho considerável, em poucos anos a evolução fez seu tamanho diminuir para que coubesse no bolso e com um preço reduzido a apenas 20 dólares. Este exemplo da calculadora é uma imagem predizendo como seria o futuro da tecnologia móvel, avançando velozmente a cada lançamento de uma nova tecnologia.

Em seguida surgiram os relógios digitais, seguido pelas câmeras de vídeo nos anos 80, com vendas superiores a 1 milhão de unidades no mundo todo. Prevendo a necessidade de computadores pessoais, foram criados pequenos aparelhos chamados organizadores pessoais que serviam como uma agenda e ajudavam o usuário a guardar compromissos, telefones e anotações. Em seguida surgiram os celulares, que rapidamente atingiram a marca de mais de 10 milhões de unidades vendidas no mundo em meados dos anos 90.

Os computadores pessoais passaram a se tornar portáteis nos anos 90, quando mais de 1 milhão de *notebooks* foram vendidos no mundo, sendo adotado basicamente por profissionais que precisavam de mobilidade. Na metade dos anos 90 surgiu o chamado PDA (*Personal Digital Assistant*), um dispositivo móvel que tinha como objetivo possuir mais funcionalidades que os antigos organizadores pessoais e ao mesmo tempo ser mais portátil que os *notebooks*, mesmo possuindo capacidade inferior de processamento e armazenamento. Com a adição de funcionalidades devido a internet móvel, câmera integrada e sincronização com o computador *desktop*, o conceito de PDA foi evoluindo para *PalmTops*, *hand-held computers* e os comunicadores, com um conceito de computador portátil e celular. O Nokia 9000 tornou-se o comunicador mais popular, pois contava com inúmeras funcionalidades, como acesso a internet, agenda de endereços, calendário, anotações, calculadora e relógio mundial.

A tecnologia que permitiu este avanço tecnológico já existia desde os anos 90, era a rede sem fio, mas foi graças as evoluções em outros setores que juntos fizeram a tecnologia móvel se tornar o que é hoje: imprescindível. Fatores como os avanços na infraestrutura, em *software*, abundância de capital, o interesse do próprio consumidor e a necessidade dos negócios serem em tempo real, tudo isso passou a exigir mobilidade, o acesso portátil e em tempo real a recursos e ferramentas que até então somente poderiam ser acessados em um computador *desktop*. Esta mudança pode ser dividida em duas fases: a fase um, onde o usuário deveria ir onde o computador está, e a fase dois, o momento em que vivemos, onde o computador está onde o usuário estiver.

Estas evoluções reiteram a tendência da computação ubíqua. A computação móvel, demonstrando um crescimento a cada ano, está se tornando cada vez mais onipresente na vida das pessoas, de uma forma em que os seus usuários integram as interações com o mundo virtual em seus comportamentos naturais, como, por exemplo, acessar a internet em momentos livres, como períodos de folga na pausa para um café ou no deslocamento dentro do ônibus. As tecnologias móveis trazem a facilidade de interação, fazendo com que o usuário nem perceba as informações que envia e recebe, por ter se tornado algo natural (HELAL, 2002).

Ao focarmos na evolução dos celulares, desde o surgimento desta tecnologia até os dias atuais, é possível dividi-la em etapas com características bastante únicas (FLING, 2009) conforme a tabela 1.

	<p>Era do tijolo (19 73–1988)</p>	<p>Pela primeira vez os telefones não precisavam de fios e eram portáteis, mas precisavam de baterias enormes para poderem ter energia suficiente para conectarem com as escassas redes de celulares. Com a proliferação da tecnologia móvel, o que iniciou-se como um item de luxo, tornou-se algo que qualquer um poderia comprar e guardar no bolso.</p>
	<p>Era da Barra de Chocolate (1988–1998)</p>	<p>O termo que denomina esta era vem do formato longo, fino e retangular que os celulares possuíam. Com o surgimento de novas redes de celulares a demanda por energia para se conectar se tornou menor, diminuindo o tamanho do aparelho. Foi nesta era que surgiu o serviço de mensagens, o SMS</p>
	<p>Era dos Recursos (1998–2008)</p>	<p>Nesta era aumentou a variedade de aplicações e serviços para os celulares, como: ouvir músicas, tirar fotos e o início da utilização da internet móvel. Companhias móveis focaram na criação de conteúdo digital como <i>ringtones</i>, <i>wallpapers</i>, jogos e outras aplicações.</p>
	<p>Era dos Smartphones (2002–atualmente)</p>	<p>Os <i>smartphones</i> se distinguem por utilizarem um sistema operacional em comum, terem uma tela maior, um teclado e outra forma de conectividade via <i>wireless</i>. Os <i>smartphones</i> não obtiveram muito sucesso, fazendo com que os telefones da era dos recursos continuassem se desenvolvendo cada vez mais.</p>
	<p>Era Touch (2007-Atualmente)</p>	<p>Era que iniciou com o surgimento do iPhone em 2007. Propondo uma nova interação entre o usuário e seu aparelho, os dispositivos da era <i>touch</i> trouxeram um meio completamente novo de oferecer aos usuários novas formas de interagir e compreender a informação disponível no mundo todo.</p>

Tabela 1 - Evolução dos celulares (FLING, 2009)

3.2. Tecnologias móveis

As tecnologias móveis vão muito além do que carregamos em nossos bolsos o dia todo, os celulares e *smartphones* são apenas alguns exemplos.

Elas estão também presentes nos automóveis através de sistemas de rastreamentos por satélite, dispositivos de comunicação, diagnóstico do estado do veículo, tudo com um aplicativo que pode ser acessado através do celular, no computador do trabalho e em casa.

Outro exemplo é a grande mudança de paradigma ocorrida na última década em relação à leitura de livros, através da criação da mídia digital. Apesar do primeiro livro digital no mundo ter sido lançado em 1971, pelo Projeto Gutenberg, que tornou a declaração da independência dos Estados Unidos digital, a venda de livros digitais somente ganhou força com o início das vendas de conteúdo digital no início do século 21. Em 2007, a loja virtual Amazon lançou o seu *e-reader* Kindle, um dispositivo móvel que facilita a leitura de livros digitais, seguido pela livraria Barnes & Noble com o Nook, com a disponibilidade de quantidade de conteúdo inimaginável, a tendência da leitura digital cresceu cada vez mais.

Enquanto livrarias são fechadas por esta mudança, as que sobrevivem se veem obrigadas a venderem os *e-readers*, ou iniciarem a venda de conteúdo digital, não sendo mais o bastante o antigo livro que ocupa espaço na estante. Pesquisas indicam que enquanto leitores de livros em seu formato tradicional leem uma média de 15 livros por ano, os usuários de *e-readers* estão lendo uma média de 24 livros (RAINIE, 2012). Isto mostra que hoje a tecnologia móvel está presente em tarefas bastante antigas e que não eram nem imaginadas serem feitas de outra forma.

Mas não são somente com os *e-readers* que é possível ler livros digitais, com o lançamento dos *tablets*, como o iPad em 2010, o mercado de leitura digital se expandiu, e com estes aparelhos a abrangência de possibilidades

aumentou enormemente. Além da leitura digital, é possível jogar jogos de diversas modalidades, que vão de cartas, *puzzle*, corrida e ação, superando em qualidade gráfica até mesmo consoles de vídeo games de poucos anos atrás.

A nova era de aparelhos móveis hoje conta com infinitas possibilidades, são compostas por *smartphones*, *tablets* e iPods. Todas estas opções vêm bem abastecidas de opções de personalização, com lojas virtuais como a App Store da Apple, a Marketplace da Microsoft e a Android Market da Google. Elas dispõem de aplicativos gratuitos e pagos para as mais variadas categorias, que vão de jogos, gerenciadores financeiros, tradutores de idiomas, agregadores de notícias, navegadores, e muito mais, são até mesmo disponibilizados aplicativos completamente diferentes, mas com as mesmas funcionalidades, sempre tentando contentar todos os tipos de consumidores.

3.3. M-Business

O *e-business* surgiu juntamente com a evolução da internet, é a denominação para uma estratégia de negócios que visa maximizar o valor do cliente e dos lucros através da utilização da tecnologia (KALAKOTA, 2002).

O *e-commerce*, nomenclatura dada ao comércio digital, faz parte do *e-business*, surgiu na década de 80, mas quando começou a ser amplamente utilizada a internet móvel começou a se desenvolver mais ativamente, atraindo a mídia, o mercado de ações e seus investidores, criando o que é chamado de *m-business*, sendo vista como a próxima etapa da curva de tecnologia (KALAKOTA, 2002).

O *m-business* refere-se à infraestrutura necessária para sustentar as relações de negócios e troca de informações, serviços e mercadorias através dos aparelhos móveis. O *m-commerce*, assim como a relação do *e-commerce* com o *e-business*, é a possibilidade de que qualquer pessoa possa realizar transações eletrônicas através de um dispositivo móvel, como um *smartphone*, funciona como

uma fachada de tudo que acontece por trás, através do *m-business* (KALAKOTA, 2002).

A mudança do *e-business* para o *m-business* não se deu apenas ao fato do comércio passar a ser móvel, a mudança foi radical na forma de como a tecnologia era vista antes do surgimento dos dispositivos móveis. Com esta evolução houve uma migração estrutural, os modelos que antes eram fechados e centrados no computador pessoal, passaram a serem modelos móveis centrados na pessoa. As aplicações de *e-commerce* e *e-business* são focadas em usuários fixos, com uma infraestrutura ligada por fios, mas devido à ampla adoção da rede sem fio, o *m-commerce* está sendo mais amplamente utilizado, onde os negócios podem ser realizados em movimento.

3.4. A economia móvel

Através da consolidação do *m-business*, pode-se visualizar as estruturas que compõe a economia móvel. Estas capturam e organizam todas as atividades que acontecem no mundo destas novas tecnologias, sendo elas (KALAKOTA, 2002):

- Novas plataformas revolucionárias: as empresas de tecnologia fornecem uma plataforma e as ferramentas para que os desenvolvedores escrevam as aplicações, sendo esta a posição mais lucrativa neste mercado;
- Infraestrutura de aplicação móvel: as empresas habilitam, distribuem e gerenciam as aplicações e serviços móveis;
- Novas oportunidades de inovação: as empresas podem utilizar a *web* para desenvolver e distribuir novos produtos e serviços direto aos seus consumidores;

- Foco no consumidor: focar em como a internet móvel irá se integrar ao estilo de vida dos consumidores, quais tarefas, atividades e transações os clientes desejam.
- Foco na cadeia de suprimento: utilização da internet móvel para aprimorar o processo de monitoramento e gerenciamento da cadeia de suprimento, melhorando o fluxo de informações, como: pedidos, produtos e pagamentos;
- Foco operacional: prover aos usuários a possibilidade de acessar as informações de suas atividades de trabalho a qualquer hora e qualquer lugar, desta forma, funcionários, vendedores e parceiros de negócios poderão aumentar sua produtividade.

Através da análise das estruturas citadas anteriormente, pode-se verificar mais atentamente o potencial de cada nova tecnologia móvel, assim como as oportunidades e riscos que podem surgir.

Sabe-se também que as evoluções nas tecnologias móveis são muito velozes, por isso é difícil prever como será o futuro e direcionar as estratégias de negócios para o caminho mais lucrativo. Para determinar a melhor forma de desenvolver soluções móveis propõe-se que sejam analisadas três tendências que influenciam a direção da economia móvel (KALAKOTA, 2002), são elas:

- Sociedade conectada: os consumidores passam a querer acesso contínuo à comunicação e informação, independente de sua localidade;
- Velocidade de serviço: as pessoas não querem mais ter que ficar esperando em filas por algum serviço, por isso desejam aplicações que facilitem estas situações, como serviços de bancos, hotéis e companhias aéreas;

- Simplicidade e conveniência: os clientes querem produtos e serviços fáceis e convenientes de utilizar, esperam melhoras significativas nos serviços de autoatendimento.

4. ANÁLISE TÉCNICA

Neste capítulo serão abordadas as análises técnicas necessárias para a realização deste trabalho.

Serão expostas as características dos dispositivos móveis e do desenvolvimento para seus aplicativos. Também serão detalhadas as tecnologias que serão utilizadas.

4.1. Características dos dispositivos móveis

A principal característica do dispositivo móvel é a mobilidade, que é a capacidade da tecnologia realizar múltiplas e diferentes tarefas, assim como conectar-se e trocar informações com outras pessoas, aplicativos e sistemas.

Para ser considerado móvel, um dispositivo deve possuir outras quatro características (LEE, 2005), sendo elas:

- **Portabilidade:** caracteriza-se pela capacidade do dispositivo de se deslocar ou ser deslocado facilmente, onde em geral devem poder ser carregados na mão. Este fator abrange tamanho e peso, incluindo também os seus acessórios;
- **Usabilidade:** é a capacidade do dispositivo de ser utilizado por diferentes pessoas e ambientes. Esta característica abrange o usuário: em relação ao seu tamanho, força, flexibilidade, destreza, conhecimento e capacidade; a natureza: em relação às condições normais e extremas de utilização e; o próprio dispositivo: em relação a características como tempo de inicialização, integridade dos dados, interface com o usuário, robustez e resistência;

- **Funcionalidade:** são as tarefas que o dispositivo pode efetuar através de aplicações móveis que executa. É possível diferenciar os aplicativos em duas categorias: os independentes, que não necessitam de interação do usuário e de outro aplicativo e; os dependentes, que são alimentados com informações do usuário ou de outros sistemas;
- **Conectividade:** dispositivos móveis têm como principal característica a conectividade, pois não são feitos para operarem sozinhos. O seu objetivo é poder conectar pessoas e trocar informações entre diferentes sistemas. A conectividade pode ser dividida em três formas: estar sempre conectado, conexões intermitentes e operações que não necessitam de conexão.

4.2. Desenvolvimento para dispositivos móveis

O desenvolvimento de *softwares* para dispositivos móveis é mais complexo do que *softwares* orientados aos negócios. Devido às suas características, tais como: necessidade de ser em tempo real, memória limitada da tecnologia, canais de entrada e saída limitados, necessidade de ferramentas caras de desenvolvimento, tendo uma forte relação com a dependência de *hardware* e diferentes processadores (LEE, 2005).

As peculiaridades mencionadas anteriormente são pontos vistos como desafios no desenvolvimento móvel, com isto, definiu-se uma arquitetura de negócio específica para o desenvolvimento dos aplicativos, visualizada na figura 1 (LEE, 2005).

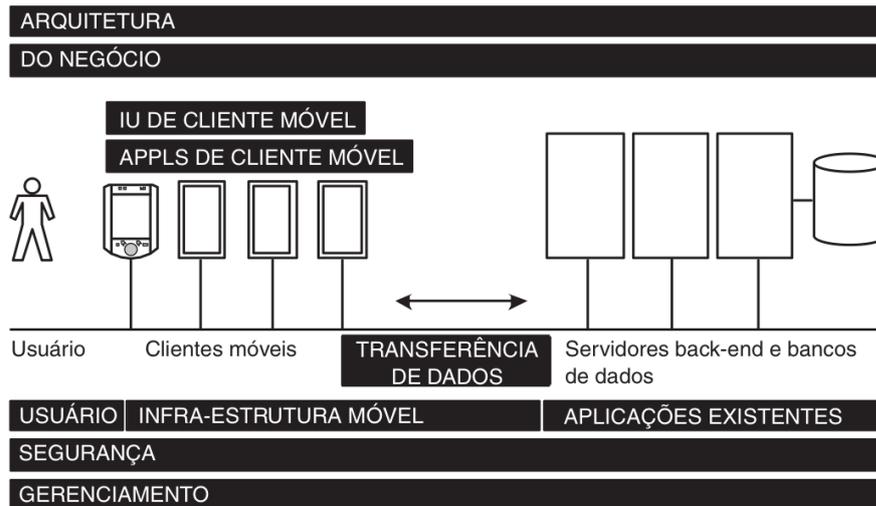


Figura 1 - Arquitetura do negócio (LEE, 2005)

Os itens na figura 1 são divididos e conceituados da seguinte forma (LEE, 2005):

- **Mobilidade:** conforme explicado no capítulo anterior, as aplicações móveis deverão seguir esta principal característica dos dispositivos móveis;
- **Contexto do negócio:** é necessário identificar o contexto de negócio do aplicativo, que deve satisfazer profissionais e consumidores que utilizam estas tecnologias, estes podem possuir como objetivos a utilização para se comunicarem, como ferramenta de trabalho, para entretenimento, para fins educacionais ou para serviços de localização, como um GPS;
- **Arquiteturas de aplicação móvel:** definir os tipos de arquitetura móvel para desenvolvimento que podem ser seguidas, levando em consideração as definições realizadas no item anterior;
- **Infraestrutura móvel:** análise que deve ser realizada ao definir o tipo do dispositivo móvel a ser destinado o aplicativo. Podem se dividir em telefones celulares, *tablets* e *notebooks*, cada um possuindo uma

característica única de *hardware* que deve ser levada em consideração;

- Interface com o usuário de cliente móvel: através do dispositivo a ser destinado o aplicativo, considerar a interface e periféricos que podem ser utilizados pelo usuário. Garantir também que a aplicação seja mais intuitiva e fácil de usar do que um sistema manual já existente;
- Aplicações clientes móveis: onde se deve definir a arquitetura a ser seguida baseando-se na infraestrutura disponibilizada pela tecnologia escolhida;
- Transferência de dados cliente-servidor: definir a forma com que as trocas de informações do usuário entre o aplicativo e o servidor irão ocorrer;
- Tornar móveis as arquiteturas de aplicação existentes: verificar a necessidade de transformar ou atualizar aplicações já existentes para a compatibilidade com os aplicativos móveis;
- Segurança: é necessário o controle de acesso a usuários e a proteção dos dados para casos de perda e roubo do dispositivo. É muito importante manter a privacidade e encriptação dos dados, sempre ficando atento a vulnerabilidades e ameaças;
- Gerenciamento do desenvolvimento de aplicações móveis: seguem as mesmas características do gerenciamento de projetos de desenvolvimento de sistemas para outras tecnologias não móveis. Dividindo-se em fases de requisitos, projeto, codificação, testes e implantação;

- Estudos de caso de aplicações móveis: é considerado importante realizar estudos de casos e análises em aplicativos já existentes, verificando melhores práticas e diferentes resultados.

Pode-se ainda adicionar à lista, um item sobre levantamento de indicadores de utilização (OLIVER, 2008). Para o sucesso nos negócios de um aplicativo móvel, faz-se necessário recolher dados de utilização e atividades realizadas pelos usuários, com o objetivo de prever problemas e planejar a evolução do aplicativo.

Em relação ao desenvolvimento de sistemas para dispositivos móveis, esta etapa geralmente segue uma divisão em camadas em uma arquitetura cliente-servidor, conforme figura 2, facilitando a reutilização de código.

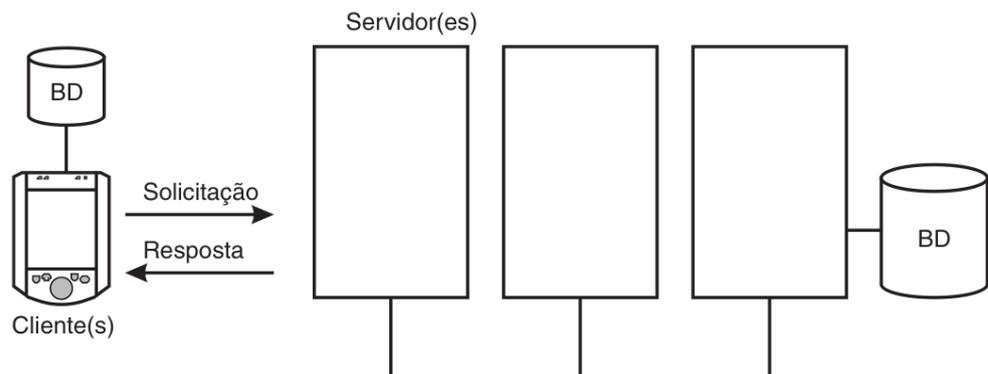


Figura 2 - Arquitetura cliente-servidor (LEE, 2005)

Na camada cliente, existe o aplicativo, também chamado de *embedded systems*, que é executado no dispositivo móvel, este pode ou não possuir códigos de aplicação, por isso podem ser divididos em duas categorias (LEE, 2005):

- Clientes magros: são aqueles clientes que não possuem código personalizado, por isso dependem completamente do servidor para o seu funcionamento. Nestas situações geralmente são utilizados navegadores *web* para a utilização do aplicativo;

- Clientes gordos: são clientes que não necessitam estar conectado o tempo todo com o servidor, sendo útil para situações onde a conexão é instável ou inexistente. Para isto, faz-se necessário o armazenamento de dados localmente no dispositivo.

Quando há a existência de código na camada do cliente, estes geralmente também adotam uma arquitetura separada em camadas, conforme descrito na figura 3 (CAO, 2012).

Camada da Aplicação

- Camada que está no topo do desenvolvimento da hierarquia da arquitetura. Compreende basicamente aplicações independentes para completar determinada tarefa, que podem compreender entrada e saída de dados, processamento e comunicação. Sendo o sistema operacional o responsável por gerenciar estas instruções.

Camada do *Middleware*

- É a camada que suporta diferentes linguagens de programação, como a procedural C, orientada a objetos C++ e Java. Serve como um compilador e interpretador de código para converter em código de máquina.

Camada do Sistema Operacional

- Responsável por cooperar com a aplicação para aperfeiçoar o gerenciamento de processos, procedimentos e tratamento de exceções.

Camada de Abstração do *Hardware*

- É a camada mais próxima do *hardware*, aquela que faz contato diretamente com o dispositivo. Proporciona interfaces para o sistema operacional e aplicações. Geralmente é fornecido pelos fabricantes da tecnologia, principalmente para utilização por desenvolvedores.

Hardware

- Dispositivos móveis como: *smartphones*, celulares, leitores de *e-books*, video games, etc.

Figura 3 - Arquitetura de desenvolvimento móvel (CAO, 2012)

Para que uma arquitetura seja considerada ideal para o desenvolvimento de aplicativos móveis, foram definidos cinco princípios que uma arquitetura deverá dispor:

- Requisitos: deverá levar em consideração os requisitos funcionais, de negócios e do usuário;
- Independência de tecnologia: desenvolver um aplicativo que possa ser executado independente da tecnologia nem sempre é possível, mas o ideal é fazer de uma forma em que seja possível a reutilização de código para desenvolvimento entre diferentes dispositivos;
- Alto desempenho e disponibilidade: criar uma aplicação que mantenha o desempenho estável conforme a demanda dos usuários oscila, assim como atentar para a garantia de disponibilidade de servidores que o aplicativo necessita para seu funcionamento;
- Escalabilidade: deve-se implementar a capacidade do aplicativo e do servidor em suportar o aumento da quantidade de usuários e requisições realizadas;
- Requisitos de sistema e do usuário: analisar a capacidade de utilização dos usuários em relação à execução de determinadas funcionalidades seguindo as especificações de *hardware*, como, por exemplo, qualidade de conectividade. Com o objetivo de obter o melhor desempenho do aplicativo em praticamente todas as execuções.

Além de escolher a arquitetura que será utilizada para o desenvolvimento do aplicativo móvel, faz-se necessário analisar o *hardware* de cada tecnologia que a aplicação será disponibilizada, para isto são elencados os seguintes itens de grande importância (LEE, 2005):

- *CPU*: define a qual velocidade o dispositivo tem capacidade de processar as instruções do aplicativo, influenciando principalmente na velocidade de sua execução. Geralmente é proporcional ao tamanho do dispositivo, onde os celulares possuem menor capacidade de processamento e os *notebooks* maior capacidade;
- Sistema operacional: define a linguagem, ferramentas e tecnologias que deverão ser utilizadas para o desenvolvimento do aplicativo;
- Memória: auxilia na capacidade de processamento da *CPU*, através de armazenamento temporário de dados. Segue também a lógica proporcional ao tamanho do dispositivo;
- Espaço em disco: serve para o armazenamento permanente de dados, deverá ser levada em consideração a quantidade de dados que o aplicativo requer que seja armazenado localmente no dispositivo;
- Baterias e fonte de alimentação: o aplicativo deverá levar em consideração a duração da bateria do dispositivo e a existência ou não de uma conexão a uma fonte de energia;
- Dispositivos de conexão: define de que forma podem ser realizadas as trocas de informações e sincronização do dispositivo através do aplicativo;
- Tela: devem ser consideradas as diferentes dimensões e qualidades existentes, para adaptar o aplicativo a cada necessidade;
- Acessórios como teclado, mouse, caneta e outros periféricos: avaliar quais periféricos é possível utilizar no dispositivo a ser explorado pelo aplicativo. Como, por exemplo, o celular possui câmera digital, mas não possui o mouse.

Sabendo as características dos dispositivos móveis, conhecendo as arquiteturas e suas possibilidades, assim como as restrições e capacidades que são únicas nas tecnologias móveis, será possível realizar comparações com os levantamentos a serem realizados no capítulo seguinte, relacionado às características individuais de cada tecnologia escolhida a ser abordada neste trabalho: HTML 5, Windows Phone, Android e iOS.

4.3. Plataformas Móveis

Nesta etapa serão descritas e detalhadas a história, as características, a arquitetura e o ambiente de desenvolvimento de cada tecnologia móvel abordada neste trabalho.

4.3.1. Web: HTML 5

HTML, sigla para HyperText Markup Language, é a linguagem utilizada pelos conteúdos disponibilizados na internet. É basicamente um sistema de código em textos que possibilitam criar a estrutura que vemos nas páginas da internet.

4.3.1.1. História

Quando o HTML surgiu, não era nada mais que algumas *tags* que determinariam o *layout* do texto a ser disponibilizado na página da internet. Atualmente, além desta funcionalidade principal, a internet também disponibiliza aplicações interativas e dinâmicas. Com as mudanças exigidas pelas novas tecnologias, como celulares e *smartphones*, foi necessário começar a pensar de uma forma diferente na exibição de conteúdo da internet, com isto surgiu o HTML 5, novos padrões de desenvolvimento para suportar as exigências dos dias atuais (HARRIS, 2011).

A linguagem HTML foi criada em 1989 por Tim Berners-Lee, em uma época em que a internet basicamente era acessada via programas de linhas de comando e para criação de documentos, tendo como objetivo ser simples para que

as pessoas pudessem participar no processo de criação. Com a criação dos *web browsers*, os leitores de HTML, novas funcionalidades foram sendo adicionadas. Em 1993 foi incluída a possibilidade de adicionar imagens no código. Em 1994 o Netscape Navigator surgiu e o grupo *World Wide Web Consortium (W3C)* foi criado para organizar e padronizar o código de HTML. Em 1995 a Microsoft lançou o Internet Explorer, e na competição com o Netscape, ambas as empresas adicionavam funcionalidades aos seus navegadores que não seguiam os padrões definidos pela W3C. Em 1995 o HTML 2 foi determinado como padrão, seguido pelo 3.2 em 1997 e o HTML 4 em 1998. Em 2002 a Microsoft já possuía 95% do mercado de navegadores com o Internet Explorer 6, utilizando basicamente o padrão HTML 4. A partir de 2004, novos navegadores foram criados, primeiro o Firefox, seguido pelo Safari, Opera e o Google Chrome. Com tantas evoluções, a internet passou a não servir mais apenas para leitura de documentos, transformando as páginas *web* em aplicações *web*.

Com o tempo, as funcionalidades do HTML 4 foram ficando ultrapassadas, então em 2002 o grupo W3C introduziu um novo padrão, o XHTML, seguido pela sua versão 2.0, que não foi muito aceita pela dificuldade no desenvolvimento do código, com isto outro grupo, o *WHATWG (Web Hypertext Application Technology Working Group)* criou o padrão HTML 5, que se mostrou mais prático, em seguida também foi adotado pelo W3C, tornando então o próximo padrão da linguagem *web*.

4.3.1.2. Características

As principais características do HTML 5, em relação a sua evolução, são (FREEMAN, 2011):

- Recursos multimídia: suporte a reprodução de vídeos e áudio de forma nativa no navegador *web*;

- Maior poder de desenvolvimento: adição do elemento *canvas*, que possibilita a renderização gráfica e animações de jogos com atualização em tempo real, permitido através da utilização em conjunto com o JavaScript;
- Melhora na semântica: é possível desenvolver utilizando menos código e de forma mais organizada, devido à introdução de novas características e regras que separam o significado dos elementos a serem utilizados.

4.3.1.3. Arquitetura de desenvolvimento

O HTML 5 é formado por mais elementos que o próprio HTML, é a integração de diferentes tecnologias, como: HTML, CSS, JavaScript e outras tecnologias que são executadas na camada do cliente e do servidor (HARRIS, 2011). Desta forma é possível organizar a arquitetura do HTML 5 conforme a figura 4:

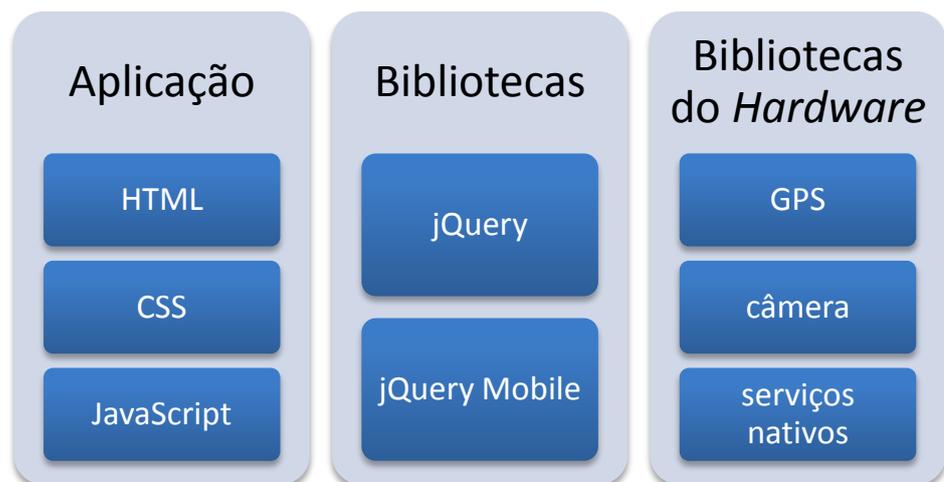


Figura 4 - Arquitetura HTML 5 (HARRIS, 2011)

O HTML 5 possui, além de sua arquitetura, uma estrutura de conteúdo que representa a hierarquia dos elementos do seu documento HTML, chamado de Document Object Model, o DOM, conforme figura 5. Esta estrutura é a chave para o desenvolvimento *web* e proporciona uma ligação entre a estrutura de conteúdo do HTML com o JavaScript (FREEMAN, 2011).

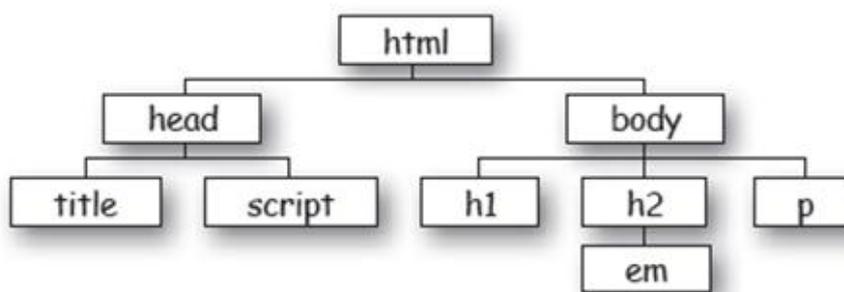


Figura 5 - Document Object Model (DOM)

4.3.1.4. Ambiente de desenvolvimento

Para iniciar o desenvolvimento na linguagem HTML 5 é necessário configurar os seguintes aplicativos (FREEMAN e ROBSON, 2011):

- **Editor de HTML:** é possível utilizar qualquer editor de textos, inclusive o bloco de notas do Windows. Aplicativos editores de HTML possuem a vantagem de identificarem no código fonte diferentes tipos de linguagens *web* e também possuir algumas funcionalidades como auto completar e auto indentação;
- **Navegador de internet:** onde será possível visualizar o aplicativo que está sendo desenvolvido. É recomendada a utilização de no mínimo dois navegadores diferentes, para a comparação dos resultados e realização de ajustes necessários. Para o desenvolvimento de

aplicativos móveis, também é necessário possuir um dispositivo móvel ou um emulador para poder testar o aplicativo;

- Servidor *web*: algumas funcionalidades do HTML 5 e do JavaScript só são executadas em um servidor *web* real. É necessário instalar e configurar localmente um servidor, como o Apache ou o ISS.

4.3.2. Windows Phone

O Windows Phone é o sistema operacional para dispositivos móveis criado pela Microsoft. A sua última versão possui uma interface também presente no Windows 8 para desktop.

4.3.2.1. História

O Windows Phone, sucessor do Windows Mobile, foi divulgado na Mobile World Congress em Barcelona, em fevereiro de 2010. A sua primeira versão foi chamada de Windows Phone 7, seguida pela versão 7.5, lançada no ano seguinte, até a sua versão atual, o Windows Phone 8, lançada no final do ano de 2012.

O mundo do Windows nos dispositivos móveis iniciou com o Windows Mobile em 2000, onde era chamado de Pocket PC 2000, seguido pelo Pocket PC 2002, lançado em 2001, que passou a ser utilizado em *smartphones*, além dos telefones com Pocket PC. A cada ano novas versões foram surgindo, Windows Mobile 2003, Windows Mobile 2003 SE, Windows Mobile 5, Windows Mobile 6, 6.1 e a 6.5, quando foi descontinuado devido ao surgimento do Windows Phone.

No Windows Phone, a grande mudança foi o *design*, que passou a ser denominada Metro, mudando o seu foco de empresarial para o mercado consumidor. Nesta nova interface, o usuário pode personalizar a sua tela principal através de mosaicos de aplicativos que atualizam em tempo real, mostrando diversas informações sem precisar executar nenhum aplicativo. Além da interface, a organização das informações é realizada através de centros, os *hubs*, que reúnem dados de redes sociais com informações do dispositivo, como os contatos e fotos.

4.3.2.2. Características

As principais características do Windows Phone são (MICROSOFT, 2013):

- Multitoque com 4 toques: é possível usar esse recurso para criar jogos para quatro jogadores;
- Tela inicial: totalmente personalizável através de pequenos mosaicos, chamados de Live Tiles, que são atualizados com as informações do aplicativo a que se refere;
- People Hub: é possível conectar a lista de contatos com vários serviços, como Facebook e Skype, e visualizar atualizações e realizar chamadas rapidamente;
- Chat: é possível criar um grupo de conversa, com o objetivo de troca de mensagens e compartilhamento de listas de tarefas, calendários e fotos;
- Segurança: possui proteção para crianças através de senha, onde é possível limitar o acesso a aplicativos, jogos, músicas e vídeos;

- Serviços de nuvem: integração com o serviço de armazenamento de arquivos na nuvem através do SkyDrive;
- Xbox: integração com perfis de usuário do videogame Xbox;
- Tap+Send: ao aproximar o seu telefone com outro com a mesma tecnologia, é possível compartilhar fotos, contatos, músicas e páginas da internet;
- Mapas: possibilidade de *download* de mapas para utilização *offline*;
- Data Sense: os dados trafegados pelo navegador *web* são compactados para evitar a utilização além do contratado com a operadora de celular;
- Outras funcionalidades como comandos por voz, recursos para encontrar o telefone quando perdido ou roubado e aplicativos do pacote Microsoft Office, também estão presentes.

4.3.2.3. Arquitetura de desenvolvimento

Conforme figura 6, a arquitetura do Windows Phone é dividida em dois aplicativos distintos. O primeiro, chamado de TaskHost, é um modelo baseado em XAML, que surgiu no lançamento do Windows Phone 7. O segundo, o CoreApplication, é um novo modelo de aplicação voltado para o Windows Phone 8 (WHITECHAPEL e MCKENNA, 2013).

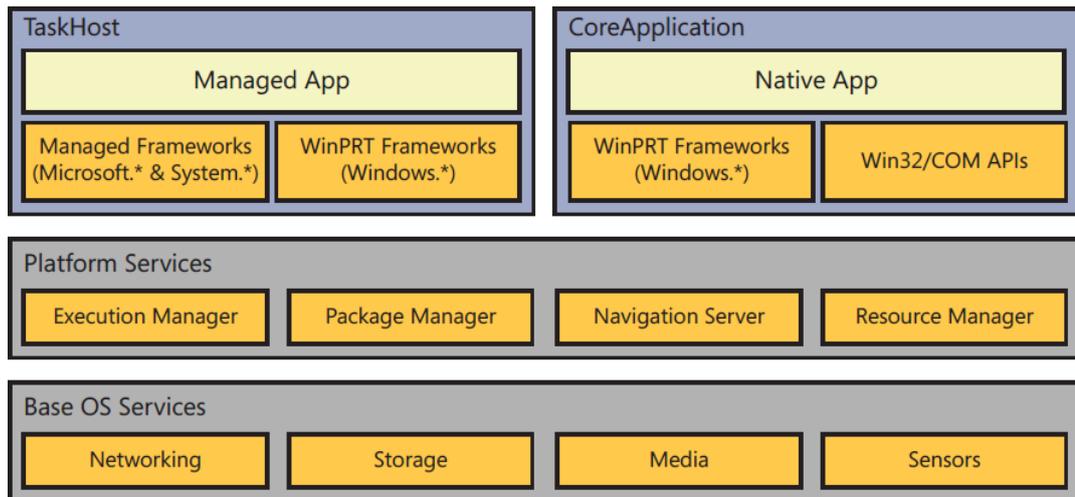


Figura 6 - Arquitetura Windows Phone (WHITECHAPEL e MCKENNA, 2013)

Ambos os modelos compartilham os mesmos serviços da plataforma, sendo eles (WHITECHAPEL e MCKENNA, 2013):

- Package Manager: responsável pela instalação e desinstalação dos aplicativos e pela manutenção dos dados gerados pela sua execução;
- Execution Manager: controla a lógica associada à execução das funcionalidades do aplicativo;
- Navigation Server: gerencia todas as atividades e execuções dos aplicativos;
- Resource Manager: responsável por monitorar e gerenciar os recursos do dispositivo utilizados pelos aplicativos.

Além destes serviços, são compartilhadas as funcionalidades do sistema operacional, que abrangem sensores como GPS, acelerômetro e bússola, disponibilidade de api's para armazenar e gerenciar qualquer tipo de informação e funcionalidades multimídia baseadas no Windows Media Player com integração com o Xbox Music.

4.3.2.4. Ambiente de desenvolvimento

O ambiente de desenvolvimento de aplicativos para Windows Phone, para a última versão, a 8, é chamado de Windows Phone 8 SDK. O SDK (*Software Development Kit*) é composto pelo seguinte conteúdo (WHITECHAPEL e MCKENNA, 2013):

- Microsoft Visual Studio 2012 Express for Windows Phone;
- Microsoft Blend 2012 Express for Windows Phone;
- Emulador do Windows Phone;
- Modelos de projetos, *assemblies* de referência e bibliotecas.

O ambiente é disponibilizado gratuitamente pela Microsoft, que pode ser feito o *download* em <http://dev.windowsphone.com>. Importante salientar que somente pode ser instalado no Windows 8.

4.3.3. Android

O Android é uma plataforma móvel de código livre baseada em Linux desenvolvida pela Google, com o objetivo de operar em dispositivos com toque na tela, como smartphones e *tablets*.

4.3.3.1. História

Em 2007 a Google anunciou o Android, que no ano seguinte passou a ser comercializado através do *smartphone* HTC Dream na sua versão 1.0. Em seguida foi lançado para o Motorola Droid e o HTC Nexus One. Em 2011 passou a ser comercializado também para *tablets*, iniciando através do Motorola Xoom. Segundo a Google, mais de um milhão de usuários utilizam o sistema operacional, que conta hoje com mais de 600 mil aplicativos.

Curiosamente, as versões do sistema seguem uma ordem alfabética e a nomenclatura de doces, iniciando com o CupCake , Donut e Éclair em 2009, seguido pelo Froyo e Gingerbread em 2010, pelo Honeycomb e Ice Cream em 2011, até a sua versão atual, a Jelly Bean 4.2, cada uma trazendo a sua lista de melhorias.

Por ser desenvolvido pela Google, o Android tem total integração com todos os seus serviços, como o site de pesquisa de mesmo nome, o Google Maps, a rede social Google Plus, o Youtube, o Google Play, o Gmail e o Google Drive. Todas estas ferramentas expandem a utilização do dispositivo móvel para os *desktops*.

4.3.3.2. Características

As principais características do Android são (GOOGLE, 2013):

- Utilização de *widgets*: pequenos ícones que ficam na tela inicial, onde é possível escolher entre seus aplicativos favoritos, tendo acesso a atualizações sem sair da tela principal;
- Tela de notificações: onde é possível visualizar em um só lugar as notificações de novas ligações, mensagens e e-mails;
- Multitarefas: é possível executar diversos aplicativos ao mesmo tempo e alternar a visualização entre eles;
- Escrita por voz: possibilidade de escrever e-mails, SMS e qualquer tipo de texto, através da voz;
- Compartilhamento de imagens: possibilidade de explorar, editar e compartilhar rapidamente nas redes sociais as fotos tiradas direto do celular;
- Compartilhar com o toque: compartilhamento de fotos e vídeos com outros celulares com a mesma tecnologia, apenas encostando-os. Também é possível parear dispositivos via *bluetooth* apenas encostando os aparelhos;
- Fotos panorâmicas de 360°: através do Phot Sphere é possível tirar fotos panorâmicas de todo o seu redor e realizar *upload* do resultado direto para o Google Maps, onde outros usuários poderão visualizá-la;

- Múltiplos usuários: nos *tablets*, é possível criar múltiplos usuários, cada um com sua tela inicial, aplicativos e jogos.

4.3.3.3. Arquitetura de desenvolvimento

A arquitetura do sistema operacional Android é dividida em cinco camadas (LEE, 2012), conforme figura 7.

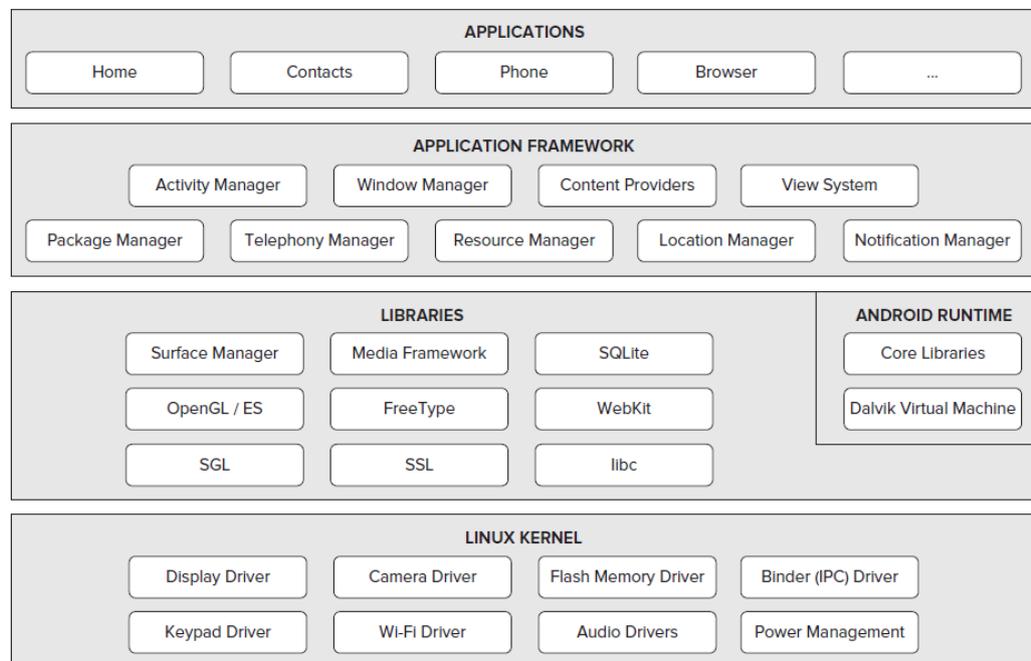


Figura 7 - Arquitetura Android (LEE, 2012)

As camadas descritas na figura 7 possuem as seguintes nomenclaturas e funcionalidades (LEE, 2012):

- **Aplicativos:** é nesta camada onde se encontram todos os aplicativos já instalados no dispositivo e aqueles que são adquiridos pelo usuário através da loja virtual;
- **Framework de aplicação:** é onde são disponibilizadas as funcionalidades para os desenvolvedores utilizarem nos aplicativos;
- **Bibliotecas:** é a camada que contém o código fonte das funcionalidades da tecnologia, como, por exemplo, bibliotecas para suporte ao banco de dados e funcionalidades para navegação *web*;
- **Runtime Android:** situada na mesma camada que as bibliotecas, tem como objetivo fornecer um conjunto de bibliotecas que permitem aos desenvolvedores escrever aplicativos usando a linguagem de programação Java;
- **Núcleo Linux:** é o núcleo que o Android se baseia. Esta camada contém todos os *drivers* de baixo nível para os vários componentes de um dispositivo Android.

4.3.3.4. Ambiente de desenvolvimento

Para o desenvolvimento de aplicativos para o Android são necessárias algumas ferramentas e o SDK. São todas gratuitas. O SDK do Android utiliza o Java SE Development Kit (JDK). Desta forma, antes de iniciar o desenvolvimento, é necessário primeiramente realizar o *download* e a instalação através do link www.oracle.com/technetwork/java/javase/downloads/index.html.

Após a instalação do JDK, pode-se fazer o *download* do Android SDK, que contém as seguintes ferramentas para desenvolvimento (LEE, 2012):

- Depurador;
- Bibliotecas;
- Emulador;
- Documentação;
- Códigos de exemplo;
- Tutoriais.

É possível realizar o download do Android SDK através do link <http://developer.android.com/sdk/index.html>.

Após a instalação do SDK é preciso instalar uma ferramenta de desenvolvimento, chamada de integrated development environment (IDE), que no caso do desenvolvimento para Android a mais recomendada é o Eclipse, na sua versão para desenvolvedores em linguagem Java EE.

Com o Eclipse configurado é necessário instalar as ferramentas de desenvolvimento para o Android, chamadas de Android Development Tools (ADT), um *plug-in* para o Eclipse.

O último passo é a criação de um ambiente de testes para os aplicativos, chamado de Android Virtual Devices (AVDs), são emuladores que simulam diferentes tipos de hardwares para testes.

4.3.4. iOS

O iOS é o sistema operacional da Apple, presente em todos os seus dispositivos móveis, como *tablets*, celulares e reprodutores de músicas.

4.3.4.1. História

O iOS foi lançado em 2007, juntamente com o lançamento do iPhone. Neste momento era ainda chamado de iPhone OS, somente em 2010 teve seu nome alterado para iOS, quando outros dispositivos, como iPod's e iPad's, passaram a utilizar o sistema operacional. Com a evolução e a grande aceitação destes dispositivos, o iOS teve um grande crescimento.

A primeira versão do iOS foi a 1.0 que funcionava na primeira versão do iPhone, o 2G, seguida pela versão 2.0, que também passou a rodar nos iPod's. Um ano depois a versão 3.0 era lançada com novas funcionalidades e com uma versão 3.2 exclusiva para o novo dispositivo da Apple, o iPad. Através da versão 4.0 surgiu o FaceTime, onde seria possível fazer vídeo chamadas entre os dispositivos da Apple. No final de 2011 foi lançada a versão 5.0, que contava com um assistente por voz chamado Siri, a possibilidade de troca de mensagens gratuitamente entre dispositivos da Apple através do iMessage e o serviço de armazenamento de arquivos na nuvem, o iCloud, neste momento todos os dispositivos passaram a utilizar a mesma versão do iOS. Um ano depois, a atual versão foi lançada, a 6.0, que traz integração nativa com o Facebook e a possibilidade de tirar fotos panorâmicas.

O iOS conta com a tecnologia de multitoque, criação de vídeos e fotos, executa músicas, entre outras funcionalidades básicas de outros *smartphones*. Em 2008, através do iTunes, *software* de sincronização da Apple, a App Store foi criada, expandido as possibilidades do dispositivo. Na loja virtual, quando lançada,

contava com apenas 500 aplicativos, hoje estão disponíveis mais de 850 mil, que já contam com mais de 50 bilhões de *downloads*. Soma-se ainda milhares de músicas e vídeos disponíveis, que já ultrapassam 25 bilhões de *downloads*, uma média de 15 mil *downloads* por minuto, aumentando ainda mais as funcionalidades dos dispositivos.

4.3.4.2. Características

As principais características do iOS são (APPLE, 2013):

- Interface intuitiva: fácil de entender como funciona, desenhada pensando na utilização através do toque;
- Assistente Siri: através de comandos de voz é possível perguntar informações e solicitar ações como criar anotações, realizar ligações, escrever e-mails, etc;
- Multitarefas: é possível executar diversos aplicativos ao mesmo tempo e alternar a visualização entre eles;
- iCloud: é o serviço de armazenamento de dados na nuvem. Armazena músicas, fotos, aplicativos, e-mails, contatos, calendários e muito mais, mantendo todos os dispositivos Apple sincronizados;
- Músicas: através da sincronização do iTunes com o iCloud, é possível ouvir músicas em qualquer dispositivo Apple;
- Facetime: é possível realizar vídeo chamadas gratuitas entre dispositivos da Apple;

- iMessage: é possível enviar mensagens para qualquer dispositivo Apple, gratuitamente;
- Safari: o navegador dos dispositivos Apple, salva informações no iCloud e é possível recuperá-las em outro dispositivos. Também possibilita armazenar links em uma lista de leitura;
- AirPlay: através da Apple TV, é possível transmitir conteúdos de outros dispositivos Apple direto para a tela da TV;
- Game Center: é uma central de informações de jogos, possibilita interagir e jogar com amigos diversos aplicativos, armazena também escores e conquistas em um perfil.

4.3.4.3. Arquitetura de desenvolvimento

A arquitetura do sistema operacional iOS é dividida em quatro camadas (RAY, 2011), conforme figura 8.

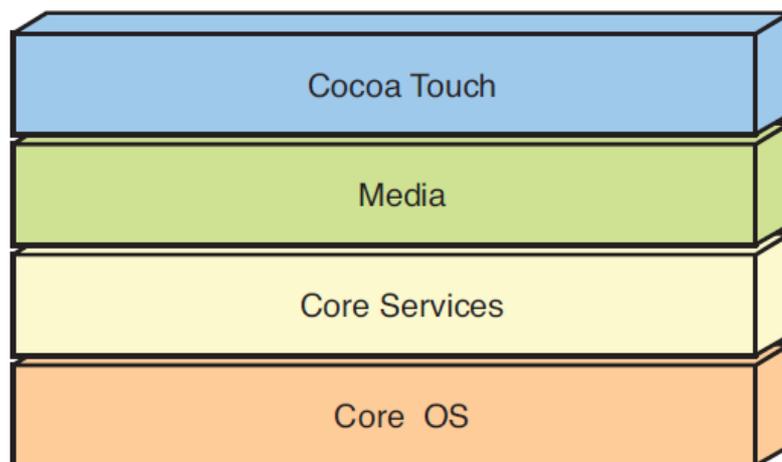


Figura 8 - Arquitetura iOS (RAY, 2011)

Cada camada da arquitetura possui as seguintes funcionalidades (RAY, 2011):

- Cocoa Touch: camada formada pelos *frameworks* que fornecem as principais funcionalidades para o desenvolvimento das aplicações;
- Media Services: possui os *frameworks* relacionados a funcionalidades multimídias, como gráficos complexos, vídeos e criação de gráficos 3D em tempo real;
- Core Services: nesta camada encontram-se as funcionalidades de acesso aos serviços do sistema operacional, como acesso a arquivos e a rede do celular;
- Core OS: é a camada mais próxima ao *hardware* do dispositivo, acessa os serviços de mais baixo nível do sistema operacional, como *threads*, funções matemáticas, acessórios de *hardware* e criptografias. Funcionalidades desta camada são raramente utilizadas em aplicativos.

4.3.4.4. Ambiente de desenvolvimento

Para o desenvolvimento de aplicativos para iOS é necessário primeiro registrar-se gratuitamente como um *Registered Apple Developer* através do link <http://developer.apple.com/programs/register>, utilizando uma Apple ID.

O próximo passo é realizar o *download* do iOS SDK e o Xcode, IDE utilizada no desenvolvimento de aplicativos. Para realizar o *download* das ferramentas de desenvolvimento para a versão 7 do iOS, que será lançada no final de setembro de 2013, é necessário fazer o *download* dos aplicativos através do link <https://developer.apple.com/ios7/>.

Na instalação dos aplicativos necessários no desenvolvimento para iOS é um requisito possuir um Sistema Mac OS X baseado em Intel.

4.4. Comparação entre as tecnologias

Através do estudo realizado sobre as quatro diferentes tecnologias, podem-se identificar as principais funcionalidades de cada uma delas. Com esta informação foi elaborada a tabela 2, onde são listadas as características levantadas e realizada a comparação entre as tecnologias.

Funcionalidades	Windows Phone	Android	iOS	HTML 5
Toques simultâneos	Sim	Sim	Sim	Não
Atualização de aplicativos na tela inicial	Sim	Não	Não	Não
Tela inicial personalizável	Sim	Sim	Sim	Não
Serviços de armazenamento na nuvem	Sim	Sim	Sim	Não
Suporte a Flash	Não	Sim	Não	Sim
Sincronização via <i>wi-fi</i>	Sim	Não	Sim	Não
Ligação com vídeo	Sim	Sim	Sim	Não
Memória expansível	Não	Sim	Não	Não
Screenshot nativo	Não	Sim	Sim	Não
Direções GPS	Sim	Sim	Sim	Não
Navegação com <i>bookmarks</i>	Não	Sim	Sim	Não
Notificações centralizadas	Não	Sim	Sim	Não
Sistema de arquivos	Não	Sim	Não	Não

Controle dos pais	Sim	Não	Sim	Não
Comandos por voz	Não	Sim	Sim	Não
Controle de transferência de dados	Sim	Não	Não	Não
Múltiplos usuários	Não	Sim	Não	Não
Transferência de dados por proximidade	Sim	Sim	Não	Não
Multitarefa	Sim	Sim	Sim	Não
Troca gratuita de mensagens	Sim	Sim	Sim	Não
Integração a serviços de jogos	Sim	Não	Sim	Não
Acesso a dados do celular	Sim	Sim	Sim	Não
Acelerômetro	Sim	Sim	Sim	Não
Câmera	Sim	Sim	Sim	Não
Aplicativo de mapas	Sim	Sim	Sim	Não
Serviços de localização	Sim	Sim	Sim	Não
Utilização <i>offline</i>	Sim	Sim	Sim	Não
Desenvolvimento independente de tecnologia	Não	Não	Não	Sim

Tabela 2 - Comparação entre as tecnologias

4.5. Comparação entre aplicações móveis *web* e aplicações nativas

Historicamente, observou-se que no desenvolvimento para dispositivos móveis, para criar uma experiência completa ao usuário, é necessário desenvolver os aplicativos de forma nativa, seguindo a linguagem e processo de desenvolvimento específico para cada tecnologia (FLING, 2009). Mas o desenvolvimento de aplicativos móveis *web* também têm suas vantagens, e para isto vários fatores deverão ser analisados.

O desenvolvimento de aplicativos móveis *web*, devido a sua maturidade e portabilidade, possui grandes vantagens, tornando-se uma opção bastante viável, são elencados os seguintes motivos (FLING, 2009):

- Fragmentação: atualmente não existem formas economicamente baratas de criar aplicativos nativos que abranjam uma grande fatia de mercado;
- A *web*: evolução dos navegadores *web* móveis, transformando-os de apenas um processador de texto para um avançado mecanismo para suportar aplicações e serviços mais complexos. Estão sendo adicionadas funcionalidades como detectar a localização do usuário, aceleração e meios de utilização do *hardware* do dispositivo, com isso, no futuro, aplicações móveis nativas possivelmente se reduzirão a aplicações específicas que necessitam utilizar toda a capacidade do dispositivo;
- Controle: os aplicativos nativos não são de autoria somente do desenvolvedor, pois necessitam da intermediação da loja de aplicativos disponibilizada pelo fabricante da tecnologia, fazendo com que o retorno financeiro seja dividido. Desta forma não é possível ter controle da forma de disponibilização do aplicativo,

como por exemplo, colocá-lo em lugares de mais destaque, tornando o desenvolvimento um investimento de alto risco;

- Expectativas dos consumidores: os consumidores esperam que os aplicativos funcionem direito e no dispositivo que possui no momento, que seja possível obter o mesmo conteúdo que seus conhecidos utilizam.

A plataforma móvel *web* é a única que funciona independente do dispositivo, sendo também a mais fácil de aprender a desenvolver, mais barata de produzir, mais padronizada, mais disponível para utilização e de fácil distribuição.

Observando por outro ângulo e analisando os objetivos almejados com o aplicativo a ser desenvolvido, o desenvolvimento utilizando a estrutura nativa das tecnologias pode se tornar uma melhor opção, principalmente caso seja necessário alguma funcionalidade do dispositivo que os navegadores *webs* não possuem acesso ou implementados. São listados alguns pontos que devem ser levados em consideração ao escolher o desenvolvimento nativo (FLING, 2009):

- Cobranças: efetuar a cobrança de serviços e aplicativos nativos pode se tornar um ponto fraco, pois exige maior segurança nos dados e serviços de terceiros para realizar a transação. Além disso, sempre existirá um intermediador, a loja virtual ou a operadora, que irá cobrar uma porcentagem dos resultados de vendas;
- Jogos: para a criação de jogos, o desenvolvimento nativo é a melhor opção, pois geralmente necessitam de vários recursos do dispositivo e utilização de funcionalidades únicas;
- Localização: para utilizar a funcionalidade de localização no aplicativo, geralmente é necessária a utilização de recursos próprios do dispositivo. Também somente de forma nativa, é possível que o usuário libere ou não o acesso do aplicativo à sua localização;

- Câmeras: através do desenvolvimento nativo é possível ter acesso a câmera digital do dispositivo, incorporando funcionalidades interessantes, como tirar e armazenar fotos, compartilhar fotos com amigos e até mesmo para processar códigos de barras e textos;
- Acelerômetros: trata-se da tecnologia de medir a aceleração e gravidade exercida no dispositivo, sendo utilizado para ajustar a rotação da tela, alterar o tamanho do texto ao detectar que o usuário está em movimento, aplicativos que contam passos e também na movimentação de personagens em jogos;
- Acesso de arquivos: o aplicativo deverá ser nativo caso seja necessário o acesso de dados do dispositivo, como fotos, agenda, calendário e e-mails. Sendo cabível ao usuário permitir ou não o acesso aos dados, garantindo a sua segurança;
- Falta de conectividade: na aplicação nativa é possível o usuário continuar utilizando-a sem estar conectado a uma rede de celular ou internet sem fio. O aplicativo deverá possuir um modo *offline* e suportar situações em que existe a necessidade de ficar *online*, mas que de repente o sinal possa ser perdido.

É possível perceber em uma visão geral que, a diferença entre um aplicativo nativo e *web* está entre qualidade na experiência do usuário e a facilidade no desenvolvimento (FOWLER,2013). Analisando todos estes pontos, é necessário avaliar os objetivos e funcionalidades que o aplicativo irá possuir e definir qual forma é a mais indicada para desenvolver a aplicação: através de um aplicativo móvel *web* ou nativo.

Este será o tema da segunda parte deste trabalho, definir um aplicativo, desenvolvê-lo em três diferentes linguagens nativas e *web*. Assim será possível

identificar as diferenças durante o processo de desenvolvimento e as funcionalidades do aplicativo final.

5. PROPOSTA DE SOLUÇÃO

A proposta de solução desta pesquisa consiste na criação de um aplicativo móvel utilizando uma arquitetura de desenvolvimento *web* com a linguagem de programação HTML 5. Este aplicativo deverá ser portátil nas três plataformas móveis abordadas no trabalho: Windows Phone, Android e iOS.

Será também desenvolvido o mesmo aplicativo utilizando a arquitetura nativa de cada uma das tecnologias mencionadas anteriormente.

Ao final serão analisadas as diferenças em funcionalidades e interatividade dos aplicativos criados, assim como as dificuldades e facilidades encontradas durante o processo de desenvolvimento.

O objetivo desta análise é exemplificar as vantagens do aplicativo portátil criado através da arquitetura *web* e os benefícios encontrados no desenvolvimento nas arquiteturas nativas das tecnologias escolhidas.

5.1.O Aplicativo

O aplicativo que será desenvolvido terá o nome de Keep Calm Maker. O principal objetivo é a criação de uma imagem personalizada com o slogan “Keep Calm”, onde o usuário pode dar continuação à frase, também podendo adicionar imagens.

O slogan “Keep Calm” tem origem histórica, o original é composto por “Keep Calm and Carry On”, traduzido-se como “mantenha a calma e siga em frente”. Estes dizeres surgiram em um cartaz no ano 1939, criado pelo governo do Reino Unido no início da Segunda Guerra Mundial, como pode ser visualizado na figura

9. Tratava-se de uma frase motivacional, que seria utilizada caso os alemães conseguissem invadir a Grã-Bretanha.



Figura 9 - Cartaz "Keep Calm" original

Com o passar dos anos esta imagem foi redescoberta e passou a ser utilizada amplamente em propagandas, peças de marketing e em imagens humorísticas nas redes sociais, conforme alguns exemplos na figura 10.



Figura 10 - Exemplos de imagens "Keep Calm"

5.1.1. Aplicativos existentes

Aplicativos com a finalidade proposta já são disponibilizados para desktop e dispositivos móveis. Na lista abaixo são expostos alguns exemplos com suas respectivas funcionalidades:

- “Keep Calm and Carry On Creator”: disponibilizado através do site <http://www.keepcalmandcarryon.com/creator/>, possui as seguintes funcionalidades:
 - Alteração do ícone no topo do cartaz;
 - Escolha de uma imagem de fundo;
 - Cor e tamanho da fonte;
 - Cor do fundo;
 - Visibilidade das palavras;
 - Disponibilização em redes sociais.

- “The Keep Calm-O-Matic”: disponibilizado através do site <http://www.keepcalm-o-matic.co.uk/>, possui as seguintes funcionalidades:
 - Alteração do ícone no topo do cartaz;
 - Cor, tipo e tamanho da fonte;
 - Cor do fundo;
 - Tamanho do cartaz;
 - Disponibilização em redes sociais.

- “Keep Calm And ____?”: disponibilizado através da App Store, loja de aplicativos da Apple, possui as seguintes funcionalidades:
 - Alteração do ícone no topo do cartaz
 - Cor e tamanho da fonte;
 - Cor do fundo;

- Utilizar fotos do celular;
- Disponibilização em redes sociais.

5.1.2. Funcionalidades

O principal objetivo do aplicativo é utilizar variadas funcionalidades, onde seja possível abranger diversas características das tecnologias que serão utilizadas no desenvolvimento, para poder realizar a comparação entre eles após a finalização do processo.

Para o desenvolvimento do aplicativo, foram estabelecidas as seguintes características que devem estar presentes, são descritas também o objetivo da escolha de cada uma:

- Alteração do ícone no topo do cartaz: analisar a possibilidade de abrir outra janela através do toque no ícone e alteração da imagem para a escolhida pelo usuário;
- Cor, tamanho e família da fonte: verificar o comportamento e possibilidade da alteração da cor, tamanho e família do texto na tela;
- Cor do fundo: analisar o comportamento e possibilidade da alteração da cor de fundo do aplicativo;
- Escolha de uma imagem de fundo: observar o comportamento e possibilidade da escolha de uma imagem para o fundo do aplicativo;

- Utilização de fotos do celular: verificar a possibilidade e o comportamento na escolha de uma imagem direto do álbum de fotos do dispositivo móvel e da captura de uma imagem em tempo de execução do aplicativo;
- Movimentação dos elementos textuais: observar o comportamento da movimentação dos elementos na tela;
- Salvar a imagem gerada: verificar a possibilidade e o comportamento ao salvar a imagem criada.

5.1.3. Protótipos de telas

O aplicativo irá possuir um total de oito telas, sendo divididas em uma tela principal, cinco funcionalidades, escolha do ícone superior e as informações do aplicativo.

Na tela principal, o usuário poderá alterar a imagem no topo do cartaz através do clique naquela já existente. As palavras em cada linha poderão também ser alteradas através do toque.

Nos botões localizados na parte inferior, será possível acessar as funcionalidades para alterar a fonte e o tamanho do texto; tirar uma foto para usar como fundo; alterar a imagem de fundo; mudar a cor de fundo; salvar no álbum de fotos do dispositivo e enviar por e-mail.

Na parte superior existirá a funcionalidade de bloquear a movimentação do texto existente. Pode-se observar o protótipo desta tela na figura 11.



Figura 11 - Tela inicial do aplicativo

A partir da tela inicial, será possível ter acesso às funcionalidades do aplicativo. Através do botão com a letra “A”, será possível realizar a edição do texto, escolhendo a família da fonte e seu tamanho. Pode-se visualizar o protótipo na figura 12.



Figura 12 - Tela de edição do texto

No botão com o símbolo de uma câmera fotográfica, será possível tirar fotos e inseri-las como imagem de fundo do cartaz. O seu *layout* será conforme figura 13.

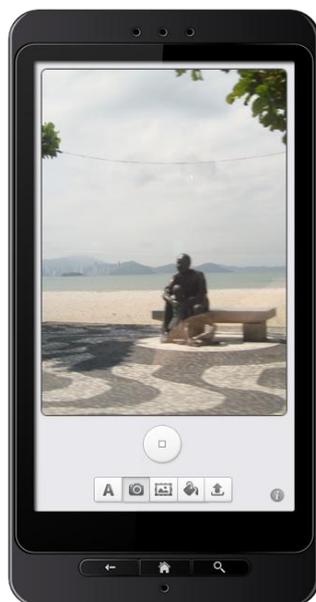


Figura 13 - Tela de captura de imagem

No terceiro botão, será possível escolher uma imagem de fundo entre pré-cadastradas e sugeridas pelo aplicativo. O protótipo pode ser visualizado na figura 14.



Figura 14 - Tela para alteração de imagem de fundo

Através do botão com o símbolo de uma lata de tinta, será possível escolher uma cor para o preenchimento do fundo, sendo elas sugeridas pelo aplicativo. O protótipo desta tela pode ser visualizado na figura 15.



Figura 15 - Tela para alteração da cor do fundo

No último botão de funcionalidades, será possível compartilhar e salvar a imagem gerada pelo aplicativo. Serão disponibilizadas as opções de envio da imagem por e-mail e salvar a imagem no álbum da câmera do dispositivo. O protótipo pode ser visualizado na figura 16.



Figura 16 - Tela para compartilhar e salvar a imagem

Na tela inicial, ao clicar no ícone existente na parte superior da imagem, será carregada uma tela com ícones existentes no aplicativo para o usuário escolher. O protótipo desta tela pode ser observado na figura 17.



Figura 17 - Tela de ícones

A última tela contém as informações do aplicativo. O seu protótipo pode ser visualizado na figura 18.



Figura 18 - Tela de informações do aplicativo

6. ANÁLISE DOS RESULTADOS

Neste capítulo serão comparadas as versões finais dos aplicativos para cada plataforma. Em seguida serão analisadas as características e comparadas as funcionalidades.

6.1. Aplicativo *web*

Para o desenvolvimento do aplicativo *web* foram utilizadas as seguintes linguagens de programação: HTML, PHP, a linguagem web script Javascript e a biblioteca JavaScript jQuery.

Para algumas funcionalidades utilizou-se bibliotecas de terceiros, como na interface de toque, no armazenamento de informações na memória local do dispositivo e o componente de paleta de cores, pois não havia a possibilidade de desenvolver de forma nativa.

O desenvolvimento do aplicativo foi realizado com o editor Dreamweaver da Adobe e a execução e testes foram realizados através da transferência dos arquivos necessários para o *FTP* do domínio www.carinapilar.com, utilizando o aplicativo FileZilla. O endereço para acessar o aplicativo é www.carinapilar.com/KeepCalmFiles/1KeepCalmMaker.html.

6.1.1. Análise do processo de desenvolvimento

Foram necessárias 50 horas para o desenvolvimento do aplicativo *web*. Durante o processo foram encontrados os seguintes pontos positivos:

- Facilidade em encontrar material, fóruns e livros sobre o assunto;
- Preparação do ambiente de desenvolvimento de forma ágil: instalação do editor de código e aplicativo para transferência de arquivos para o *FTP*;
- Agilidade no desenvolvimento e execução do código.

E também foram encontradas as seguintes dificuldades:

- Falta da possibilidade de depuração do código;
- Necessária a utilização de bibliotecas de terceiros;
- Limitações do *browser*, como: acesso as fotos do celular e armazenamento de variáveis muito extensas no endereço da página;
- Utilização de diferentes linguagens para abranger todas as funcionalidades: HTML, Javascript, jQuery e PHP;
- Particularidades de cada dispositivo e *browsers* que alteram o comportamento do aplicativo;
- Funcionalidades que não executam em dispositivos móveis, como: *mailto*, para envio de e-mails, função *drag and drop*, para arrastar o conteúdo do pôster, foi necessária a utilização de uma biblioteca de um terceiro;
- Ao ocorrer oscilações na conexão com a internet, algumas funcionalidades e componentes do aplicativo ficam comprometidas, sendo necessária a atualização da página.

6.1.2. Execução do aplicativo

O aplicativo desenvolvido na plataforma *web* tem como objetivo ter sua execução em totalidade em todos os dispositivos móveis que têm a capacidade de acessar a internet através de um navegador. Com a finalidade de satisfazer esta premissa, foram realizados testes em variados dispositivos e versões, os resultados obtidos são descritos a seguir:

- Windows Phone 7.8: executado com o Internet Explorer, apenas o botão para alterar o texto funcionou, sendo que ao setar diferentes configurações, nada acontece. A funcionalidade de arrastar o texto do pôster não funciona, assim como o botão para bloquear/liberar a sua movimentação;
- Windows Phone 8: executado com o Internet Explorer, todas as funcionalidades funcionam corretamente. A única dificuldade encontrada foi com a movimentação do conteúdo do pôster, que é permitida apenas em um curto espaço, pois a seleção do conteúdo é perdida ao arrastar;
- Android 4.1.2: executado com o navegador padrão do dispositivo utilizado, um Samsung S3, todas as funcionalidades foram executadas corretamente. A única questão observada foi a utilização de uma função que servia para obter o tamanho disponível na tela do navegador, onde nesta situação não estava sendo obtido o tamanho correto, sendo necessário a alteração para a utilização de um valor fixo;
- iOS 6.1.3: executado com o navegador padrão dos dispositivos da Apple, o Safari, todas as funcionalidades foram executadas corretamente.

- iOS 7.0.2: ao executar com o navegador padrão dos dispositivos da Apple, o Safari, algumas particularidades foram encontradas: é necessário executar duas vezes a função de desenhar as letras do pôster, pois na primeira vez a fonte utilizada no pôster não é carregada e; a geração da imagem, para salvar no dispositivo ou enviar por e-mail, não é possível realizar;
- Opera Mini: executado em dispositivos com iOS, nenhuma funcionalidade funcionou neste navegador. A tela inicial é carregada normalmente, mas a fonte utilizada no texto do pôster não é apresentada e nenhum botão funciona;
- Chrome: executado em dispositivos com iOS e Android, no navegador Google Chrome foi possível executar com êxito todas as funcionalidades do aplicativo.

Foi possível medir a quantidade de dados trafegados durante a execução do aplicativo *web*. Foram executadas as funcionalidades de alteração do ícone e texto do pôster e também a alteração da cor de fundo, após finalizar, foi feito o envio do pôster por e-mail. Em um teste com o iPhone 5, no navegador Chrome, utilizando o aplicativo Navita T.E.M. Personal a medição da quantidade de *kbytes* enviados ficou por volta de 50 e recebidos 100.

6.2. Aplicativo para Windows Phone

O desenvolvimento do aplicativo para o Windows Phone foi realizado em um computador com o sistema operacional Windows versão 8.1. Foi utilizada a ferramenta de desenvolvimento Visual Studio Ultimate 2012 com o *framework* de desenvolvimento .Net 4.5 e o aplicativo desenvolvido é compatível com a versão Windows Phone 8, ou superior.

6.2.1. Análise do processo de desenvolvimento

Foram necessárias 30 horas para concluir o aplicativo para o Windows Phone. Durante o processo de desenvolvimento foram elencadas as seguintes características:

- Emulador é bastante lento na primeira execução, mas depois funciona perfeitamente;
- Não permite enviar e-mails com anexos, porém há uma opção paga que habilita esta funcionalidade. No aplicativo foi utilizada a opção Compartilhar, nativo através do ShareMedia;
- O Visual Studio é uma excelente ferramenta de desenvolvimento, dando maior suporte ao desenvolvedor ao longo do processo;
- É permitido usar a câmera no emulador, mas a imagem é simulada;
- O componente de paleta de cores não é nativo, sendo necessário obtê-la de terceiros;
- Não é permitido usar o teclado para digitar no emulador, como no iPhone e no Android, dificultando o desenvolvimento.

6.3. Aplicativo para Android

O desenvolvimento do aplicativo para a plataforma do Android foi realizado em um computador com sistema operacional Windows versão 7. Foi utilizado o ambiente de desenvolvimento Eclipse versão 4.2.1 com o Android Developer Tools build v22.0.5 instalado. O aplicativo desenvolvido é compatível com a versão do Android 2.3.3 Gingerbread, com a API level 10, ou superior.

6.3.1. Análise do processo de desenvolvimento

Foram necessárias 45 horas para a conclusão do aplicativo para Android. Durante o processo de desenvolvimento foram observadas as seguintes características:

- O emulador é bastante lento, principalmente na primeira execução, tornando cansativo o desenvolvimento;
- O componente de paleta de cores não é nativo, sendo necessário obtê-la de terceiros;
- É permitido usar a câmera no emulador, mas a imagem é simulada.

Foram encontradas dificuldades na utilização da ferramenta de desenvolvimento, em um determinado momento não atualizava mais o aplicativo no emulador e parou de executar o aplicativo, sendo necessária a criação de um novo projeto e a transcrição de todo o código anterior para os novos arquivos.

6.4. Aplicativo para iOS

O desenvolvimento do aplicativo para a plataforma do sistema operacional iOS foi realizado em um computador Mac Mini com MacOS Lion versão 10.8.5. Foi utilizado o ambiente de desenvolvimento xCode 5 e o aplicativo desenvolvido é compatível com a versão iOS 6.1, ou superior.

6.4.1. Análise do processo de desenvolvimento

Foram necessárias 40 horas para finalizar a criação do aplicativo para iOS. Durante o processo de desenvolvimento foram observadas as seguintes características:

- O código, objective-C, possui mais complexidade se comparado ao desenvolvimento *web*, Java e C#;
- Funcionalidades que poderiam ser automatizadas devem ser programadas, como, por exemplo, a rolagem da tela, onde é necessário a programação do tamanho da tela e o deslocamento da rolagem;
- Os tamanhos da fonte disponibilizados para o usuário precisaram possuir uma variação menor em relação ao aplicativo *web*, pois o pôster deve caber na tela, sem que haja a necessidade de rolar a tela;
- Algumas fontes para as letras não existem de forma nativa, como, por exemplo, a Calibri;
- Não foi possível realizar os testes com a câmera do dispositivo, pois a utilização do emulador para esta funcionalidade não é confiável,

forçando a aquisição de uma conta desenvolvedor junto à Apple para poder realizar os testes direto no dispositivo;

- Também não é possível enviar e-mail via emulador, forçando, mais uma vez, a aquisição da conta desenvolvedor;
- O componente de paleta de cores não é nativo, sendo necessário obtê-la de terceiros.

6.5. Análise dos aplicativos

Nesta etapa serão comparados e analisados os resultados de telas e características do aplicativo e do desenvolvimento em cada tecnologia.

A análise será dividida em tópicos agrupados pelos protótipos de telas que foram definidos no item 5.1.3 deste trabalho e na comparação das tecnologias realizada no capítulo 4.4. As telas apresentadas do aplicativo desenvolvido em HTML foram capturadas acessando o endereço *web* onde está hospedado através do iPhone 5 com o iOS 7.0.3, utilizando o *browser* Chrome. As telas do aplicativo nas outras plataformas foram capturadas através do emulador de cada uma delas. Ao final deste capítulo é exibido um gráfico comparando o tempo de desenvolvimento de cada tecnologia.

Conforme figura 19, na tela inicial do aplicativo, é possível perceber algumas diferenças em relação aos componentes da tela, como a funcionalidade para bloquear e liberar a movimentação do texto e ícone da imagem e o *layout* dos botões.



Figura 19 - Comparativo da tela inicial do aplicativo

Na figura 20, na tela de edição de texto, é possível perceber algumas diferenças em relação ao *layout* e estilo dos componentes da tela, ficando bem visíveis as particularidades de cada tecnologia.



Figura 20 - Comparativo da tela de edição do texto

Na tela de captura de imagem, visualizada na figura 21, as alterações mais visíveis são apenas nos *layouts* dos botões. O que mais se destaca é a forma da escolha da imagem em HTML, que não sendo nativo ficou com uma apresentação pouco amigável.

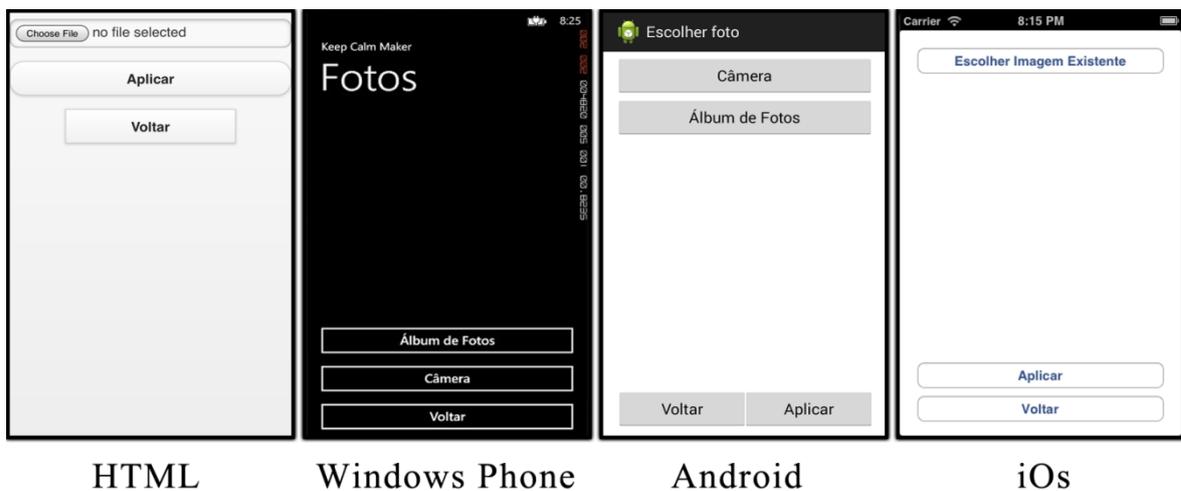


Figura 21 - Comparativo da tela de captura de imagem

Na figura 22, que mostra a tela para alteração de imagem de fundo, é possível perceber que manteve-se um padrão de *layout*. A diferença de maior impacto é o fundo preto e o título da tela no Windows Phone.



Figura 22 - Comparativo da tela para alteração de imagem de fundo

Na tela para alteração da cor do fundo, visualizada na figura 23, segue o mesmo comportamento da tela anterior, seguindo um padrão entre as tecnologias.



Figura 23 - Comparativo da tela para alteração da cor do fundo

Na figura 24, que mostra a tela para compartilhar e salvar a imagem que foi gerada, a diferença maior é no *layout* dos botões.



Figura 24 - Comparativo da tela para compartilhar e salvar a imagem

Na tela de ícones, visualizada na figura 25, é possível perceber a diferença no *layout* dos componentes de entrada de texto e nos botões, que variam bastante entre as tecnologias. As imagens dos ícones disponíveis seguiram o mesmo padrão de apresentação.



Figura 25 - Comparativo da tela de ícones

De uma forma geral, observa-se que a maior diferença nos aplicativos é na versão para Windows Phone, que possui *layout* preto, com botões na mesma cor e textos de títulos em tamanho grande. Em componentes mais comuns, como botões, texto e imagens, o *layout* segue um padrão. Já em outros componentes, como os *combo boxes*, editores de texto e seletores (como o que bloqueia a movimentação do texto) o *layout* muda bastante de uma tecnologia para a outra.

Independente destas alterações visíveis de *layout*, o comportamento do aplicativo em todas as plataformas se manteve o mesmo.

6.5.1. Comparativo e análise das características

Baseado nos conceitos estudados nos primeiros capítulos, a comparação das características de cada tecnologia realizada no capítulo 5, e os resultados obtidos no desenvolvimento do aplicativo, foi criada a tabela 3. Nela é organizada uma comparação entre as características analisadas durante o desenvolvimento e execução do aplicativo nas quatro diferentes plataformas.

Os itens apresentados na tabela são detalhados nos tópicos a seguir:

- Ambiente de desenvolvimento gratuito: os ambientes de desenvolvimento para as tecnologias escolhidas são todos gratuitos. Apesar de ter sido utilizado uma ferramenta paga, o Dreamweaver da Adobe, para o desenvolvimento em HTML, existem diversas opções sem custo;
- Ambiente de desenvolvimento próprio: o Windows Phone e o iOS possuem uma ferramenta de desenvolvimento própria. O Android possui um SDK que deve ser instalado em uma ferramenta genérica

de desenvolvimento na linguagem Java e para o HTML pode ser utilizado qualquer editor de texto;

- Disponibilização do aplicativo de forma gratuita: apenas o aplicativo desenvolvido na arquitetura *web* pode ser disponibilizado gratuitamente. Para iOS e Windows Phone é necessário adquirir uma conta para disponibilizar o aplicativo nas lojas e a loja do Android exige um certificado aplicado no aplicativo, que possui custo;
- Necessidade de internet no dispositivo: somente o aplicativo *web* necessita de internet para a sua execução total, nas outras plataformas pode ser executado *offline*, necessitando da internet apenas na funcionalidade de envio de email. No aplicativo *web* isto é um problema, pois qualquer falha na conexão as telas não são carregadas corretamente;
- Acesso aos arquivos do dispositivo: apenas nas arquiteturas nativas foi possível ter acesso aos arquivos do dispositivo. No desenvolvimento *web* o usuário deve realizar manualmente a funcionalidade de salvar a imagem gerada para o álbum de fotos do dispositivo;
- Acesso à câmera na execução do aplicativo: somente no aplicativo *web* foi possível tirar uma foto e utilizá-la durante execução. Nas outras tecnologias foi utilizado seus respectivos emuladores, que não possibilitavam tirar fotos, nem mesmo utilizando uma *webcam*;
- Necessidade de utilização de componentes de terceiros: em todas as tecnologias foi necessária a utilização de código desenvolvido por terceiros, pois nem todas as funcionalidades propostas no aplicativo foram possíveis de serem implementadas de forma nativa;

- Fontes de texto nativas: para a funcionalidade de edição do texto da imagem, somente no Windows Phone e no HTML não foi necessária a importação das fontes de texto disponibilizadas;
- Envio de email durante execução: somente o iOS não permite o envio de emails na execução do aplicativo. No Windows Phone foi utilizada uma funcionalidade própria de compartilhamento chamada ShareMedia;
- Utilização de *hardware* durante execução: somente o Windows Phone não permite a utilização do teclado físico para digitação de texto no emulador. Também somente no aplicativo HTML foi possível utilizar a câmera para a execução da funcionalidade de captura de imagens;
- Execução conforme programado: todas as tecnologias nativas são executadas no emulador conforme foi implementado. No desenvolvimento *web*, o comportamento do aplicativo varia conforme o *hardware* e o *browser* utilizados na execução;
- Captura da tela nativa: somente no aplicativo *web* foi necessária a reconstrução da imagem criada para a geração de um arquivo de imagem. Nas outras tecnologias esta funcionalidade, de captura do que foi criado, é tratada de forma nativa.

Características	Windows Phone	Android	iOS	HTML 5
Ambiente de desenvolvimento gratuito	Sim	Sim	Sim	Sim
Ambiente de desenvolvimento próprio	Sim	Não	Sim	Não
Disponibilização do aplicativo de forma gratuita	Não	Não	Não	Sim
Necessidade de internet no dispositivo	Não	Não	Não	Sim
Acesso aos arquivos do dispositivo	Sim	Sim	Sim	Não
Acesso à câmera na execução do aplicativo	Não	Não	Não	Sim
Necessidade de utilização de componentes de terceiros	Sim	Sim	Sim	Sim
Fontes de texto nativas	Sim	Não	Não	Sim
Envio de email durante execução	Sim	Sim	Não	Sim
Utilização de <i>hardware</i> durante execução	Não	Não	Não	Sim
Execução conforme programado	Sim	Sim	Sim	Não
Captura da tela nativa	Sim	Sim	Sim	Não

Tabela 3 - Comparativo de características do aplicativo entre as tecnologias

No gráfico da figura 26 é possível observar e comparar a diferença na quantidade de horas que foram necessárias para o desenvolvimento do aplicativo para as diferentes tecnologias.

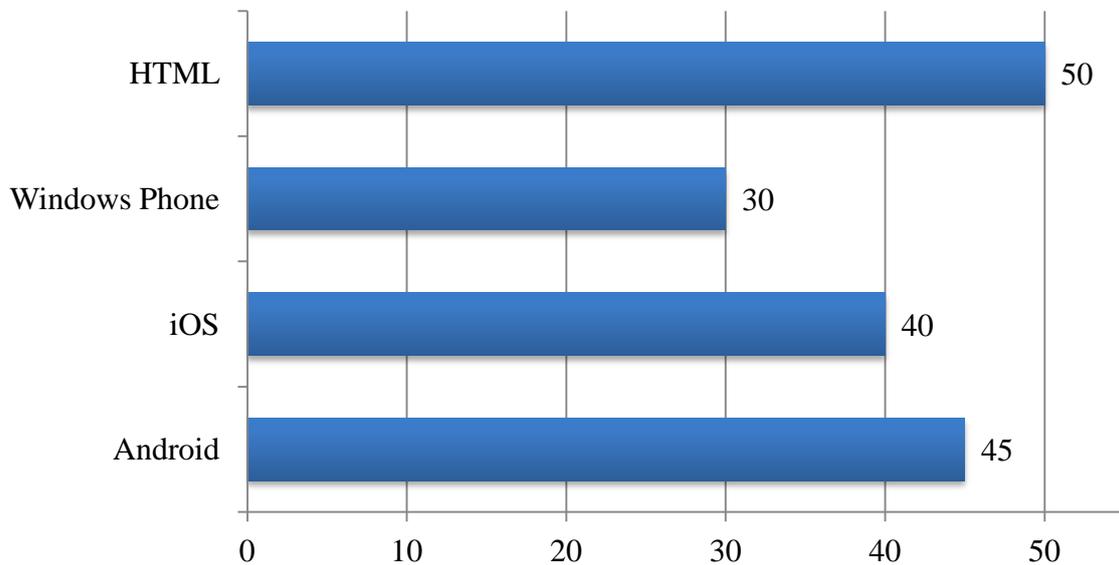


Figura 26 - Gráfico de horas de desenvolvimento

Diante destes dados percebe-se que o desenvolvimento para o Windows Phone é mais rápido que as outras tecnologias, seguida pelo desenvolvimento para iOS, depois para Android e finalizando com o HTML.

Durante o processo de desenvolvimento nas quatro diferentes tecnologias, percebeu-se que a criação do aplicativo para Windows Phone foi facilitado devido a ferramenta de desenvolvimento, o Visual Studio, possuir funcionalidades para tornar a escrita do código mais ágil. O iOS apareceu em segundo lugar devido a questão de possuir uma linguagem de programação mais complexa. Em seguida ficou o Android, que consumiu mais tempo devido a alguns problemas encontrados com a ferramenta de desenvolvimento e a lentidão apresentada na execução do aplicativo no emulador. Por último, apesar do HTML possuir uma linguagem simples e com bastante conteúdo disponível para consulta, a necessidade de pesquisas e correções para garantir a compatibilidade entre as diferentes plataformas exigiu bastante tempo de desenvolvimento.

Através do trabalho realizado, dos resultados obtidos e das informações apresentadas e analisadas, observou-se que é possível desenvolver o aplicativo proposto em uma plataforma *web*, sem que haja a perda de funcionalidades ou grandes dificuldades no processo de desenvolvimento. No comparativo das telas dos aplicativos de cada tecnologia, ficou visível que as características propostas para o desenvolvimento foram seguidas em todas as plataformas.

7. CONCLUSÃO

De um ponto de vista de desenvolvedor, diante dos resultados e percepções apresentados, conclui-se que o desenvolvimento do aplicativo em uma plataforma *web* é possível. Também se percebeu que para garantir a compatibilidade em diferentes plataformas é exigida muita pesquisa, mais horas de desenvolvimento e testes em diferentes tecnologias, tornando o processo mais custoso e estressante. Adicionando também o fato de que as tecnologias móveis avançam com uma grande rapidez, o esforço para manter o aplicativo sempre disponível para todas as tecnologias pode resultar em mais investimento. Desta forma, pode-se concluir que a melhor opção seria: realizar o desenvolvimento nas duas maiores tecnologias utilizadas no mercado de dispositivos móveis: o iOS e o Android. Disponibilizando aplicativos com mais aderência em uma fatia grande dos usuários destes aparelhos.

Analisando os resultados do ponto de vista de uma empresa de desenvolvimento de softwares para dispositivos móveis, optar pela arquitetura *web* pode se tornar uma opção mais viável financeiramente, pois será necessária apenas uma equipe de desenvolvimento para abranger uma grande gama de dispositivos móveis. Como se percebeu, as funcionalidades básicas foram mantidas e ficaram acessíveis em todas as plataformas móveis utilizadas.

De acordo com estes dois pontos de vista e os resultados obtidos, conclui-se que, para um aplicativo onde não haja muita complexidade ou utilização de recursos nativos do dispositivo móvel, é possível utilizar o desenvolvimento *web*. Através desta arquitetura pode-se desenvolver uma única vez, para todas as tecnologias disponíveis no mercado. Desta forma o aplicativo é portátil, sendo desenvolvido com reutilização de código, de uma forma ágil e prática.

Ao mesmo tempo, conclui-se que, ao aumentar a complexidade do aplicativo em relação as suas funcionalidades, partir para o desenvolvimento nativo é mais produtivo. Leva-se em consideração que, desde a sua criação até a manutenção, horas de desenvolvimento são investidas em novas funcionalidades,

ao contrário do aplicativo *web*, onde horas são gastas para garantir a compatibilidade multiplataforma.

É importante destacar que todo o aplicativo a ser desenvolvido deverá ser analisado e decidido qual arquitetura, *web* ou nativa, deverá ser utilizada. Isto deverá levar em consideração os objetivos da empresa e do próprio aplicativo.

Com o objetivo de dar continuidade na pesquisa realizada neste trabalho, são listadas algumas sugestões: agregar novas funcionalidades nos aplicativos e implementá-las no aplicativo *web*, garantindo a portabilidade nas diferentes tecnologias, onde seja possível comparar mais características entre a arquitetura de desenvolvimento *web* e nativa; disponibilizar os aplicativos em suas respectivas lojas e; divulgar os aplicativos e realizar uma pesquisa de opinião entre os usuários, comparando a percepção deles em relação às diferenças apresentadas em cada versão do aplicativo.

8. REFERÊNCIAS BIBLIOGRÁFICAS

APPLE. **iPhone iOS**. Disponível em <<http://www.apple.com/iphone/ios/>>, acesso em 21/05/2013.

CAO, Guitao. **Advances and Applications in Mobile Computing: Software Testing Strategy for Mobile Phone**. 1ª Edição, 2012. InTech, Croatia.

FLING, Brian. **Mobile Design and Development**. 1ª Edição, 2009. O'Reilly Media, Inc., Sebastopol, CA.

FOWLER, Martin. **Developing software for multiple mobile devices**. Disponível em <<http://martinfowler.com/articles/multiMobile/>>, acesso em 20/03/2013.

FREEMAN, Adam.. **The Definitive Guide to HTML5**, 1ª Edição, 2011. Springer Science+Business Media, LLC., New York, NY.

FREEMAN, Eric. ROBSON, Elisabeth. **Head First HTML5 Programming**. 1ª Edição, 2011, O'Reilly Media, Inc., Sebastopol, CA.

GOOGLE. **Discover Android**. Disponível em <<http://www.android.com/about/>>, acesso em 20/05/2013.

HARRIS, Andy. **HTML5 For Dummies**, 1ª Edição, 2011. Wiley Publishing, Inc., Indianapolis, Indiana.

HELAL, Abdelsalam. **Anytime, anywhere computing: Mobile Computing Concepts and Technology**. 1ª Edição, 2002. Kluwer Academic Publishers. New York.

KALAKOTA, Ravi. **M-Business : tecnologia móvel e estratégia de negócios**. 1ª Edição, 2002, Bookman.

LEE, Valentino. **Aplicações móveis: arquitetura, projeto e desenvolvimento**. 1ª edição, 2005, Pearson Education do Brasil, São Paulo.

LEE, Wei-Meng. **Beginning Android 4 Application Development**. 1ª Edição. 2012, John Wiley & Sons, Inc., Indianapolis, Indiana.

MICROSOFT. **Windows Phone Features**. Disponível em <<http://www.windowsphone.com/en-US/features>>, acesso em 17/05/2013.

OLIVER, Mike. **Mobile Device Management for Dummies**, 1ª Edição, 2008. John Wiley & Sons Ltd, Chichester, West Sussex, England.

RAINIE, Lee. **The rise of e-reading**. Disponível em <http://libraries.pewinternet.org/2012/04/04/the-rise-of-e-reading/>, acesso em 22/03/2013.

RAY, John. **Sams Teach Yourself iPhone Application Development in 24 Hours**. 2ª Edição, 2011, Pearson Education, Inc.

WHITECHAPEL, Andrew. MCKENNA, Sean. **Windows Phone 8 Development Internals**. 1ª Edição, 2013. Microsoft Press.