



**UNIVERSIDADE ESTADUAL DA PARAÍBA – UEPB  
CENTRO DE CIÊNCIAS EXATAS E SOCIAIS APLICADAS – CCEA  
CAMPUS VII – GOVERNADOR ANTÔNIO MARIZ**

**OTACÍLIO OLIVEIRA LIMA NETO**

**Responsividade de dados: projetando aplicativos para um  
mundo majoritariamente móvel.**

**Patos  
2017**

OTACÍLIO OLIVEIRA LIMA NETO

**Responsividade de dados: projetando aplicativos para um mundo majoritariamente móvel.**

Monografia apresentada ao Curso de Licenciatura em Computação da Universidade Estadual da Paraíba, em cumprimento à exigência para obtenção do grau de Licenciado em Computação.

Orientador: Prof. Me. Pablo Ribeiro Suárez

Patos  
2017

É expressamente proibida a comercialização deste documento, tanto na forma impressa como eletrônica. Sua reprodução total ou parcial é permitida exclusivamente para fins acadêmicos e científicos, desde que na reprodução figure a identificação do autor, título, instituição e ano da dissertação.

L732r Lima Neto, Otacílio Oliveira  
Responsividade de dados [manuscrito] : projetando aplicativos para um mundo majoritariamente móvel / Otacílio Oliveira Lima Neto. - 2017.  
52 p. : il. color.

Digitado.

Trabalho de Conclusão de Curso (Graduação em Computação) - Universidade Estadual da Paraíba, Centro de Ciências Exatas e Sociais Aplicadas, 2017.

"Orientação: Prof. Me. Pablo Ribeiro Suarez, CCEA".

1. Tecnologia em países emergentes. 2. Desenvolvimento movel. 3. Conectividade. I. Título.

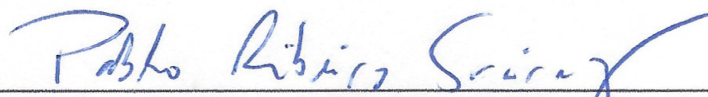
21. ed. CDD 005.1

OTACÍLIO OLIVEIRA LIMA NETO

**Responsividade de dados: projetando aplicativos para um mundo majoritariamente móvel.**

Monografia apresentada ao Curso de Licenciatura em Computação da Universidade Estadual da Paraíba, em cumprimento à exigência para obtenção do grau de Licenciado em Computação.

Aprovado em 27 de julho de 2017.



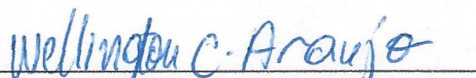
---

Prof. Me. Pablo Ribeiro Suárez  
(Orientador)



---

Prof.<sup>a</sup> Me. Larissa Barbosa Leôncio Pinheiro  
(Examinador)



---

Prof. Dr. Wellington Candeia de Araujo  
(Examinador)



## DEDICATÓRIA

Dedico esse trabalho aos meus pais, que traçaram em mim as oportunidades que lhes foram exclusas.

## **AGRADECIMENTOS**

Preciso iniciar agradecendo à Deus pela possibilidade de exaltá-lo em todas as áreas da minha vida, especialmente nas áreas profissional e acadêmica. À Ele seja dada toda a glória.

Agradeço à minha família, em especial, por ter me apoiado e sempre acreditado em mim, mais do que eu jamais acreditaria. Por todo o carinho, amor e palavras encorajadoras.

Aos meus amigos e colegas de trabalho que sempre estiveram ao meu lado, ouvindo as minhas dúvidas, novidades, angústias e alegrias.

## RESUMO

O número de usuários de internet cresce a cada ano. Como países desenvolvidos já possuem a maioria de sua população conectada à internet, são nos países em desenvolvimento que acontece o maior crescimento de usuários, em sua maioria por meio de *smartphones*. Devido a discrepância entre a qualidade e o alcance do acesso a internet entre esses dois grupos econômicos, usuários de países emergentes pagam mais por uma conexão com menor qualidade. Devido a isso, aplicativos desenvolvidos com mercados desenvolvidos em mente não entregam uma boa experiência de uso em mercados emergentes, causando perda de usuários. Para solucionar isso, esse trabalho faz um levantamento de soluções implantadas em mercados emergentes que solucionam os maiores problemas relacionados à problemas de conexão e performance em mercados emergentes. Por meio dessas soluções, e da análise dessas, é possível avaliar o grau de otimização dado por elas e quais dificuldades estão envolvidas para implementação. Esta pesquisa também traz dados relevantes sobre o cenário atual da internet no mundo, além de estatísticas de crescimento para os próximos anos.

**Palavras chave:** Países emergentes, desenvolvimento móvel, conectividade.

## **ABSTRACT**

The number of internet users around the world is growing every year. Since developed countries already have most of their population already connected to the internet, the biggest amount of growth is coming from developing nations, with most people being mobile first users. Because of the discrepancy of quality and outreach of internet in those two economy groups, users from developing nations tend to pay more for a inferior internet connection. Due to that, application developed focusing on developed nations may not deliver a good user experience in developing nations, causing loss of users. To solve this problem, this paper research for solutions developed for emerging countries that aims to solve the biggest problems related to connection problems and device performances on emerging markets. Along with the solutions and their analysis, it is possible to estimate the level of optimization allowed by them and what difficulties are involved into their implementation. This research also shows relevant information about the current scenario of internet connectivity around the world, plus forecast about growing for the next years.

**Key words:** Emerging markets, mobile development, connectivity.

## LISTA DE ABREVIATURAS E SIGLAS

ISO	Organização Internacional de Normalização
MB	Megabyte
UX	Experiência do Usuário
HTML	Linguagem de Marcação de Hipertexto
HTTP	Protocolo de Transferência de Hipertexto
CSS	Folha de Estilo em Cascata
UIT	União Internacional de Telecomunicações
SaaS	Software como Serviço
API	Interface de Programação de Aplicação
SDK	Kit de desenvolvimento de software
CEI	Comunidade dos Estados Independentes

## LISTA DE GRÁFICOS

<b>GRÁFICO 1</b> – Assinaturas de internet banda larga móvel por 100 habitantes.....	20
<b>GRÁFICO 2</b> – Estimativas de assinaturas de internet móvel por região em 2016 ...	21
<b>GRÁFICO 3</b> – Crescimento de assinaturas únicas de internet móvel, por nível de desenvolvimento econômico, em milhões.....	22
<b>GRÁFICO 4</b> – Crescimento de assinaturas únicas de internet móvel, por geração de tecnologia, em milhões.....	23
<b>GRÁFICO 5</b> – Relação entre tempo gasto com requisição otimizada e o tempo gasto sem otimização, por resolução de tela .....	38
<b>GRÁFICO 6</b> – Resoluções de tela mais usadas em aparelhos Android.....	39
<b>GRÁFICO 7</b> – Tempo médio de carregamento das imagens, em milisegundos .....	40
<b>GRÁFICO 8</b> – Tempo de requisição de imagens em conexões móveis lentas .....	40
<b>GRÁFICO 9</b> – Tempo de requisição de imagens em conexões móveis mais rápidas .....	41

## LISTA DE TABELAS

<b>TABELA 1</b> - Otimizações dos aplicativos para dispositivos e conexões limitadas... 26	26
<b>TABELA 2</b> - Melhores práticas para o desenvolvimento de aplicativos Android ..... 27	27
<b>TABELA 3</b> - Comparação de formatos de imagens ..... 32	32
<b>TABELA 4</b> - Resoluções de tela utilizadas no Android..... 33	33
<b>TABELA 5</b> – Dimensões das resoluções mais utilizadas no Android..... 34	34
<b>TABELA 6</b> - Velocidades de requisições usadas no experimento ..... 38	38
<b>TABELA 7</b> - Relação entre tamanho dos aplicativos regulares e otimizados ..... 43	43

## LISTA DE CÓDIGOS

<b>CÓDIGO 1</b> - Função que realiza requisição das imagens de acordo com a resolução desejada .....	35
<b>CÓDIGO 2</b> - Interceptador que adiciona tempo de início e fim nas requisições HTTP .....	36
<b>CÓDIGO 3</b> - Função que executa a requisição de todas as imagens e retorna uma promessa resolvida em seguida.....	35
<b>CÓDIGO 4</b> - Função recursiva que é executada por todos os itens de uma coleção de resoluções de telas .....	37



## LISTA DE FIGURAS

<b>Figura 1</b> - Skype Lite destacando o monitor de dados .....	44
--	----

## SUMÁRIO

1	Introdução.....	13
1.1	Cenário técnico científico.....	13
1.2	Definição da problemática.....	14
1.3	Objetivo.....	15
1.3.1	Objetivo Geral.....	15
1.3.2	Objetivos Específicos.....	15
1.4	Justificativa.....	15
1.5	Metodologia.....	16
1.6	Estrutura do trabalho.....	18
2	Referencial Teórico.....	19
2.1	Considerações iniciais.....	19
2.2	O cenário atual da internet móvel em países emergentes.....	19
2.3	Impactos na experiência de usuário.....	24
2.4	Aplicativos de referência.....	25
2.4.1	Facebook Lite.....	27
2.4.2	YouTube Go.....	29
2.4.3	Skype Lite.....	29
2.5	Considerações Finais.....	30
3	Análise das Soluções Disponíveis.....	31
3.1	Considerações iniciais.....	31
3.2	Adaptação de imagens.....	31
3.2.1	Impactos do redimensionamento de imagens.....	34
3.2.2	Implementação.....	41
3.3	Tamanho do instalador.....	42
3.4	Transferência de dados.....	43
3.5	Exibição instantânea de páginas web.....	45
3.6	Exibição prévia de imagens.....	46
3.7	Considerações finais.....	47
4	Conclusão.....	48
4.1	Contribuições.....	48
4.2	Limitações.....	48
4.3	Trabalhos futuros.....	49
5	Bibliografia.....	50

## 1 INTRODUÇÃO

### 1.1 CENÁRIO TÉCNICO CIENTÍFICO

*Smartphones* estão cada vez mais ganhando importância na vida das pessoas. Em países desenvolvidos eles funcionam como uma segunda tela, alternativa à *desktops* e *notebooks*, com conexões móveis de alta velocidade, disponíveis de forma ampla. Desse modo, usuários podem acessar a internet mesmo quando estão longe de seus computadores, de forma semelhantemente veloz.

Em economias emergentes, entretanto, *smartphones* tendem a ter um grande impacto no cenário de acesso a internet. Devido aos altos custos, computadores dificilmente estarão presentes nas casas das pessoas, transformando os *smartphones* no principal e muitas vezes único meio de acesso à internet para grande parte da população (INTERNATIONAL TELECOMMUNICATION UNION, 2015).

As diferenças entre países emergentes e desenvolvidos estendem-se a fatores maiores que apenas qual dispositivo é a forma primária de acesso a internet. Vários fatores estão afetando o modo e uso da internet nestes dois lugares. Preço, disponibilidade e qualidade da conexão influenciam fortemente a forma que eles acessarão a internet.

Países emergentes sofrem com caras taxas de acesso a internet, além de limitações na quantidade de dados usados por mês, as chamadas franquias de acesso. Além do mais, os problemas também estão presentes nos dispositivos móveis usados para acessar a internet. *Smartphones*, por exemplo, tendem a ter um poder de processamento menor que os celulares usados em economias desenvolvidas (GOOGLE, INC., 2016a).

Todos esses fatores causam um distanciamento na realidade móvel entre países desenvolvidos e emergentes. Isso também provoca uma preocupação no cenário atual de desenvolvimento de aplicativos móveis, uma vez que uma significativa parte dos aplicativos mais utilizados são projetados com o cenário de países desenvolvidos em mente. Nesses lugares, espera-se encontrar uma conexão a internet rápida, sempre disponível e provavelmente uma grande franquia de dados.

O uso de aplicativos, por sua vez, cresce ao redor de todo o mundo. Sendo assim, é visível que em mercados emergentes existe uma carência no que diz respeito ao uso desses aplicativos. Nestes países, usuários utilizam aplicativos projetados para

conexões rápidas e sempre disponíveis, enquanto que o usuário médio terá uma conexão móvel lenta, limitada, cara e instável (INTERNATIONAL TELECOMMUNICATION UNION, 2016b).

De modo a resolver alguns destes problemas crônicos que a internet móvel enfrenta em países emergentes, grandes empresas de tecnologia implementaram soluções que lidam com cada um desses problemas. Estas soluções, embora distintas, chegam a compartilhar características semelhantes quando analisadas a partir de pontos de vista específicos.

## 1.2 DEFINIÇÃO DA PROBLEMÁTICA

Países emergentes, devido as suas peculiaridades tecnológicas, requerem alguma customização por parte de desenvolvedores para que seus aplicativos funcionem de maneira minimamente adequada. Fatores como custos de conexão, franquias de dados, disponibilidade de acesso e desempenho de seus aparelhos influenciam no modo de uso e também em suas taxas de adoção. Por exemplo, aplicativos que ocupam muito espaço de disco podem ser removidos de aparelhos com limitações de armazenamento.

Grandes empresas encontraram soluções que resolvem problemas críticos de seus aplicativos. Essas empresas foram capazes de identificar quais são as barreiras presentes em regiões emergentes que mais afetavam seus produtos. De acordo com o contexto singular de suas aplicações, elas puderam criar versões adaptadas de seus aplicativos que procuram sanar problemas críticos de performance.

Estas empresas, nesta pesquisa representadas por Facebook, Google e Microsoft, criaram essas soluções alternativas, que coexistem com as versões regulares dos mesmos. Os usuários, então, possuem como fatores de decisão entre uma versão ou outra a qualidade de suas redes de acesso a internet, pois cada aplicativo se adequa de forma única a esses fatores.

Apesar disso, desenvolvedores de aplicativos ainda em desenvolvimento, recém lançados ou com poucos usuários em mercados emergentes apresentam dificuldades em compreender melhor o cenário no qual seus usuários estão inseridos. Muitos deles, por exemplo, são criados em países desenvolvidos, o que acaba gerando uma falsa sensação de alta qualidade e disponibilidade de acesso a internet, como também de dispositivos com ótimo ou razoável poder de processamento.

Além do mais, muitas vezes eles não possuem um número exacerbado de usuários, o que pode dificultar a tomada de decisões sobre quais partes do aplicativo otimizar a princípio, ou quais prioridades ele deve seguir. Portanto, é importante observar o que grandes empresas vêm fazendo nesse campo, a partir da análise de três aplicações chave disponíveis que foram desenvolvidas com foco totalmente em mercados emergentes.

Dado este cenário, se faz necessário um estudo no qual sejam comparadas as soluções existentes para mercados emergentes. Desse modo, será possível promover uma orientação para aqueles que se deparem com pesquisas semelhantes ou que estejam desenvolvendo aplicações que possuem usuários em países emergentes.

### 1.3 OBJETIVO

#### 1.3.1 Objetivo Geral

Promover uma orientação geral sobre soluções tecnológicas de otimização de aplicativos para mercados emergentes.

#### 1.3.2 Objetivos Específicos

- Identificar o cenário atual das conexões móveis em países emergentes e quais são as previsões para os próximos anos;
- Encontrar, na literatura, as soluções existentes para problemas característicos de países emergentes;
- Analisar o impacto na economia de recursos que a otimização de dados poderá trazer;
- Discutir a eficácia que possíveis soluções trazem à performance de rede e processamento.

### 1.4 JUSTIFICATIVA

O número de usuários móveis de internet continua crescendo de forma rápida (INTERNATIONAL TELECOMMUNICATION UNION, 2016b). Enquanto países desenvolvidos já possuem uma quantidade relevante de usuários de *smartphones* já

conectados à internet, são nos países emergentes que esse número vem se ampliando em maior escala. Indicativos apontam que o próximo bilhão de novos usuários de internet será alcançado até 2020, totalizando 5 bilhões de pessoas conectadas à internet pelo mundo (CERWALL, et al., 2015). Apesar disso, características de conexão se diferenciam de acordo com o grau de desenvolvimento econômico de cada região.

A conexão móvel entre os países desenvolvidos e em desenvolvimento não difere apenas em suas perspectivas de crescimento, mas principalmente em velocidade. Países emergentes e subdesenvolvidos sofrem com uma fraca infraestrutura de telecomunicação. Muitas cidades, principalmente aquelas longe de grandes centros urbanos, inclinam-se a possuir uma rede com maior cobertura 2G. Em 2021, mesmo com o crescimento de redes LTE e 5G, ainda existirão cerca de 1,3 bilhão de assinaturas GSM/EDGE pelo mundo, como aponta Burt (2014).

Além disso, usuários móveis de países emergentes também possuem aparelhos com especificações técnicas mais limitadas que os aparelhos mais usados em países desenvolvidos. Empresas costumam lançar aparelhos com especificações de hardware mais baixas para que consigam alcançar preços menores e assim torná-los mais acessíveis para as pessoas.

Tendo em vista a perspectiva de alcance de novos bilhões de usuários de internet móvel, fica clara a necessidade de adaptar o modelo atual de desenvolvimento de aplicativos móveis para o contexto de conexões de baixa velocidade. Não apenas isso trará um impacto positivo em usabilidade e engajamento para os usuários móveis de internet, mas também uma significativa economia nos recursos gastos por clientes móveis e servidores, implicando em um impacto financeiro positivo para ambas as partes envolvidas.

## 1.5 METODOLOGIA

Para a realização desse trabalho foi necessário um levantamento bibliográfico das soluções existentes que permitem a adaptação de aplicativos móveis à *smartphones* com conexões lentas e baixo poder de processamento.

Foram analisadas soluções desenvolvidas por empresas que adaptaram aplicativos já existentes para mercados em desenvolvimento. Como a temática desse trabalho ainda é emergente, poucas soluções estavam aptas à esta pesquisa, pois a

quantidade de material publicado ainda é ínfima. Portanto, este trabalho se ateve ao que empresas usaram como soluções.

Foi realizada uma análise de algoritmos de compressão de mídias com foco em diferentes formatos como vídeos, imagens e áudio, assim como outros algoritmos e técnicas que permitem adaptar esses mesmos conteúdos para diferentes resoluções de tela e velocidades de conexão.

Foram realizadas análises de soluções desenvolvidas por empresas com o foco em mercados emergentes que objetivavam a diminuição de dados consumidos por seus aplicativos e também a adequação dos mesmos às características desses mercados.

O levantamento bibliográfico foi feito por meio de pesquisas em sites da internet, relatórios estatísticos e artigos que fazem referência ao tema tratado nesse trabalho.

Em seguida, foi escrito um algoritmo que realiza requisições HTTP às imagens hospedadas na internet. O objetivo dessa aplicação foi mensurar o impacto que soluções de transformações de imagens teriam sobre a performance do consumo de dados e tempo total de carregamento das imagens.

No experimento, foram usados cenários distintos, com diferentes resoluções de tela e velocidades de conexão. Foram consideradas relevantes as resoluções de tela mais utilizadas segundo estatísticas de uso de *smartphones Android*. Os cenários de velocidade de *download* também foram selecionados de acordo com a tecnologia de acesso e a qualidade da conexão.

Os resultados são apresentados por meio de gráficos e tabelas após serem comparados à cenários semelhantes sem a compressão e adaptação de dados.

Existem diversas formas para classificar uma pesquisa. Segundo Silva e Menezes (2005), do ponto de vista de sua natureza, esse trabalho se caracteriza por uma pesquisa aplicada, pois gera conhecimentos práticos para problemas específicos. Também se classifica como uma pesquisa de abordagem quali-quantitativa, além de também ser uma pesquisa bibliográfica, pois utiliza material publicado por terceiros.

## 1.6 ESTRUTURA DO TRABALHO

Este trabalho está organizado em quatro capítulos. Este primeiro é responsável por trazer as características básicas da estrutura do trabalho, seus objetivos e metodologia usada.

O capítulo dois traz uma análise do cenário da internet móvel em países emergentes, construído por meio de levantamento bibliográfico, relatórios e previsões de instituições, além de uma seção que explica a importância da experiência de usuário (UX). Este capítulo também traz uma análise das aplicações que serão usadas no capítulo seguinte. Por fim, são analisadas aplicações voltadas a mercados emergentes, a partir das características relevantes para essa pesquisa.

No capítulo três é discutido o estado da arte das atuais soluções para o problema proposto para esse trabalho. Detalhes mais específicos sobre as aplicações foram conseguidos a partir de postagens no site de desenvolvedores ou no site de dispositivo de cada aplicação. Também foi desenvolvida uma aplicação para captura de imagens que calcula o tempo de requisição e consumo de dados em vários cenários de *hardware* e conexões diferentes.

O capítulo quatro, por fim, mostra as conclusões encontradas por esse trabalho, seguidas de suas limitações e o que foi deixado para trabalhos futuros.



## 2 REFERENCIAL TEÓRICO

### 2.1 CONSIDERAÇÕES INICIAIS

Esta seção traz uma visão geral do cenário atual da internet móvel em países emergentes, além de uma introdução sobre o critério de classificação entre países emergentes e desenvolvidos. No decorrer do texto, também são mostradas previsões de mudanças do cenário móvel para os próximos anos.

Ao longo deste capítulo, são introduzidos os três aplicativos que servirão como referência para uma análise comparativa entre as soluções por eles implementadas no capítulo seguinte.

### 2.2 O CENÁRIO ATUAL DA INTERNET MÓVEL EM PAÍSES EMERGENTES

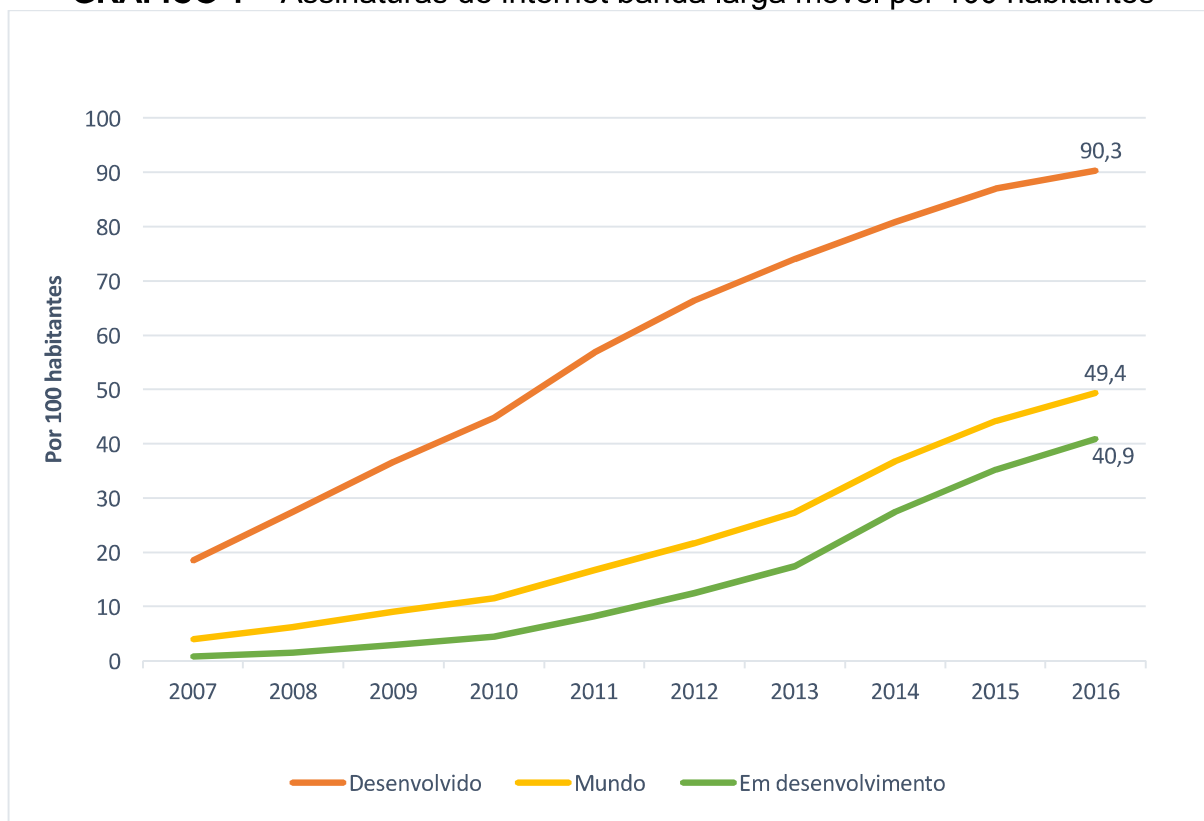
A internet tem apresentado um crescimento estável desde seu surgimento. O primeiro bilhão de usuários foi alcançado em 2005. Desde então, um novo bilhão é adicionado a cada 4 ou 5 anos, em média. Ao final de 2016, diversas projeções indicam que cinco bilhões de pessoas haveriam se conectado a internet em todo o mundo (INTERNATIONAL TELECOMMUNICATION UNION, 2015).

O crescimento da internet, contudo, acontece de forma diferente entre diversas regiões e ao classificarmos os países a partir de suas características econômicas, essa diferença se torna ainda mais aparente. Como visto em International Telecommunication Union (2016a), eles diferem tanto por fatores mais técnicos, como infraestrutura da rede, velocidade média de acesso e cobertura de sinal, como também por fatores mais demográficos, como: principal meio de acesso à internet, taxa da população com acesso a internet, custos de acesso, entre outros.

No decorrer desta pesquisa, são citados grupos de países de diferentes regiões econômicas, por meio de termos como desenvolvidos, emergentes e subdesenvolvidos. Não existe um consenso sobre como classificar países entre essas categorias. Todavia, com o intuito de simplificar esse trabalho, será usada a classificação dada pela United Nations Statistics Division (2017), que chama de regiões desenvolvidas a América do Norte, Europa, Japão, Austrália e Nova Zelândia, enquanto que as regiões em desenvolvimento ou emergentes são África, Américas

(com exceção da América do Norte), Caribe, América Central, América do Sul, Ásia (excluindo Japão) e Oceania (excluindo Austrália e Nova Zelândia).

**GRÁFICO 1 – Assinaturas de internet banda larga móvel por 100 habitantes**



Nota: Período entre 2007-2016. Dados de 2016 são estimativas.

Fonte: Adaptado de (INTERNATIONAL TELECOMMUNICATION UNION, 2016b).

Dados da International Telecommunication Union (2016b) ajudam a compreender melhor o cenário do acesso a internet pelo mundo. Estima-se que em 2016 haviam apenas 40,9 assinaturas de internet banda larga móvel por 100 habitantes em países emergentes, enquanto que em países desenvolvidos, esse número sobe para 90,3 (GRÁFICO 1). Esses dados, porém, não incluem os países menos desenvolvidos na categoria de países emergentes ou em desenvolvimento. Nessas regiões, estima-se que menos de 20% das pessoas possuem acesso à internet móvel banda larga (INTERNATIONAL TELECOMMUNICATION UNION, 2016a).

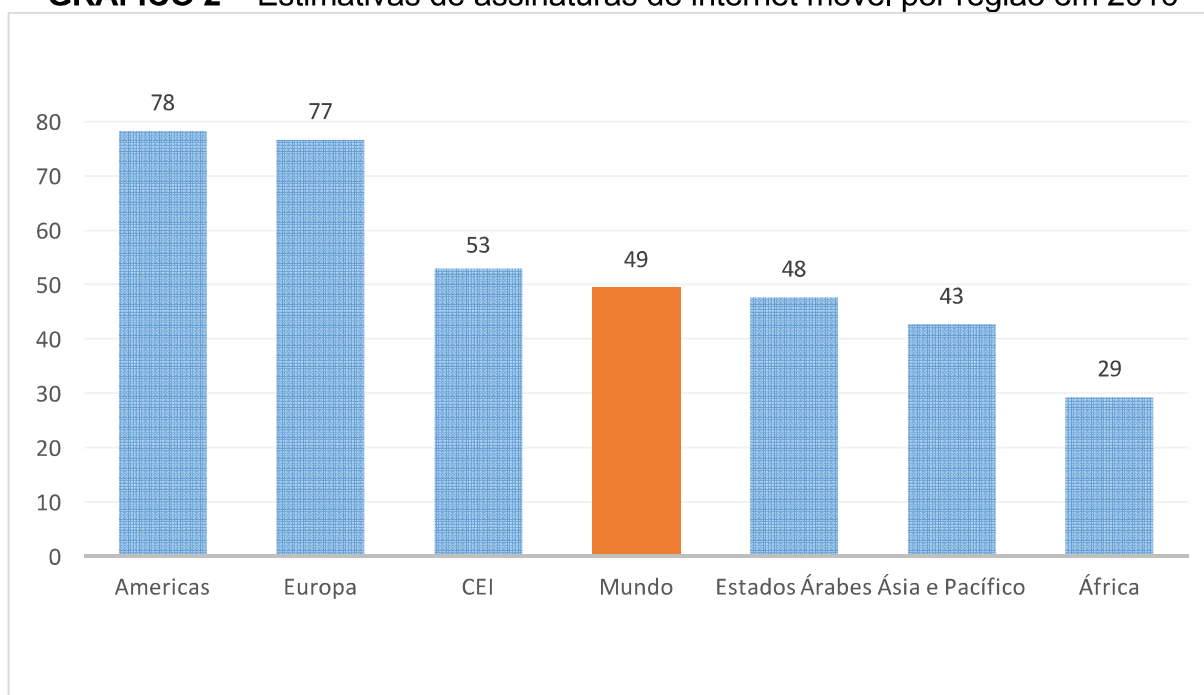
Com relação a internet fixa, países desenvolvidos tendem a possuir um maior número de linhas fixas de internet banda larga, com 30,1 linhas por 100 habitantes. Essa taxa cai consideravelmente em países emergentes e menos desenvolvidos, com

apenas 8,2 e 0,8 assinaturas, respectivamente (INTERNATIONAL TELECOMMUNICATION UNION, 2016a).

Países desenvolvidos também possuem uma maior porcentagem de casas conectadas: mais de 80% dos lares possuem acesso a internet. Esse número cai para apenas 41,1% de lares com acesso à internet em países emergentes. Por sua vez, países menos desenvolvidos possuem apenas 11,1% dos lares conectados. A África se destaca com sua baixa quantidade de lares conectados, com apenas 15,4 em cada 100 lares com acesso a internet, fato que acontece pois boa parte dos países menos desenvolvidos estão presentes nesse continente (UNITED NATIONS STATISTICS DIVISION, 2017).

Embora a taxa de penetração de internet móvel seja bem maior que a de internet banda larga fixa, é preciso observar que esta última permite a utilização de internet por mais pessoas, enquanto que linhas de internet móvel costumam ser individuais (INTERNATIONAL TELECOMMUNICATION UNION, 2015).

**GRÁFICO 2** – Estimativas de assinaturas de internet móvel por região em 2016



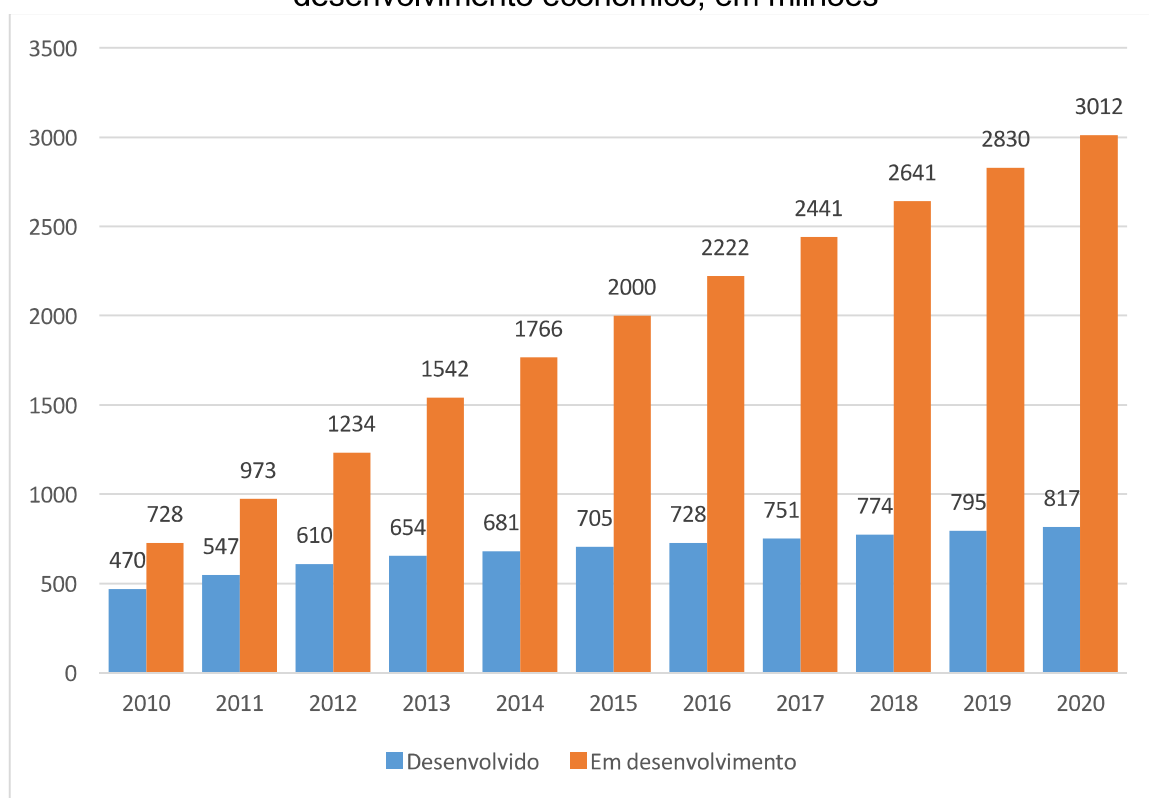
Nota: Dados de 2016 são estimativas. CEI significa Comunidade dos Estados Independentes.  
Fonte: Adaptado de (INTERNATIONAL TELECOMMUNICATION UNION, 2016b)

Separando-se os dados por região, é possível verificar que existe uma disparidade de acesso entre diferentes regiões geográficas, mesmo entre países em desenvolvimento, como visto no GRÁFICO 2. A África destaca-se com apenas 29

assinaturas por 100 habitantes, seguida pela Ásia, Pacífico e Estados Árabes com menos de 50 assinaturas. As onze nações que fazem parte da Comunidade dos Estados Independentes (CEI) têm, em média, 53 assinaturas. Europa e o continente Americano possuem, respectivamente, 77 e 78% da população com uma assinatura ativa de internet banda larga móvel. O alto número de assinaturas nas américas deve-se ao fato de que Estados Unidos e Canadá possuem uma alta taxa de inclusão de internet móvel.

Outros relatórios publicados também trazem índices de velocidade e tecnologias usadas para conexões com a internet entre diferentes países e grupos econômicos. Um relatório da GSMA (2014) aponta que em 2020, devem existir três bilhões de assinaturas de internet móvel únicas em países em desenvolvimento, quatro vezes o número de assinaturas em 2010, como visto no GRÁFICO 3.

**GRÁFICO 3** – Crescimento de assinaturas únicas de internet móvel, por nível de desenvolvimento econômico, em milhões



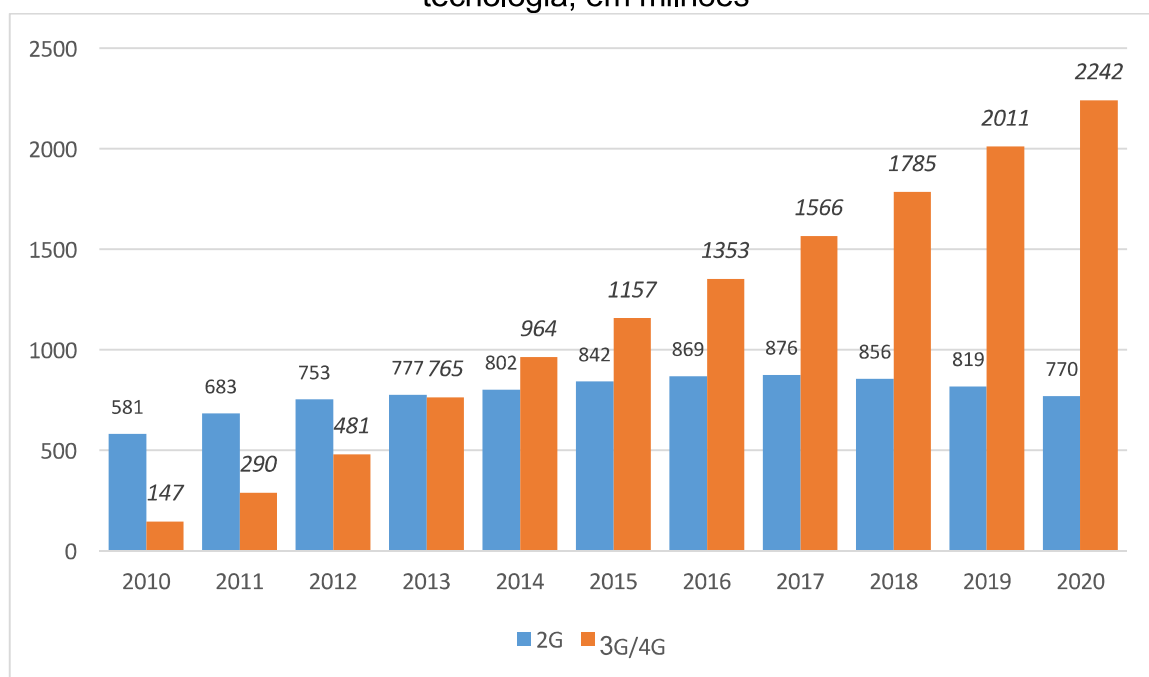
Fonte: adaptado de (GSMA, 2014)

A GSMA prevê também um crescimento expressivo nas velocidades de acesso móvel nesses países. Até 2020 projeta-se que o número de conexões 3G/4G sejam

quase três vezes o número de conexões 2G (GRÁFICO 4). Conexões 2G devem ter uma mudança pequena em seu número de usuários entre 2015 e 2018.

Embora a quantidade de conexões 2G continue sendo menor que de conexões 3G/4G, 770 milhões de pessoas continuarão utilizando-as como forma primária de acesso a internet em países emergentes em 2020. Isso representa uma perda de 32 milhões de assinaturas, comparando-se ao ano de 2014, quando, pela primeira vez, o número de assinaturas 3G/4G deve ter superado o número de assinaturas 2G. Mesmo com menor número de usuários, cerca de 85% da população mundial vive em áreas cobertas por redes 2G, enquanto que apenas 55% das pessoas vivem em regiões com cobertura 3G, que costumam ser concentradas em áreas urbanas (GSMA, 2014).

**GRÁFICO 4** – Crescimento de assinaturas únicas de internet móvel, por geração de tecnologia, em milhões



Fonte: Adaptado de (GSMA, 2014)

Países emergentes e pouco desenvolvidos também sofrem com o custo elevado de conexão. Ainda segundo a (INTERNATIONAL TELECOMMUNICATION UNION, 2016b), países em desenvolvimento pagam em média o dobro que países desenvolvidos pagam por uma conexão banda larga móvel. Em 2015, serviços de banda larga em 83 países em desenvolvimento (incluindo economias menos desenvolvidas), custavam menos de 5% da renda média mensal, que corresponde ao objetivo da Comissão de Banda Larga das Nações Unidas.

Nesse sentido, embora os mercados em desenvolvimento apresentem bons índices de crescimento, eles ainda possuem um número baixo de pessoas conectadas a internet, quando comparados com índices de acesso de países desenvolvidos. O alto custo de conexões móveis mais a dependência em tecnologias antigas e lentas também causam um atraso na qualidade do acesso a internet nessas regiões.

### 2.3 IMPACTOS NA EXPERIÊNCIA DE USUÁRIO

Partindo do pressuposto que a experiência de usuário é um dos fatores que influenciam diretamente no tempo de uso e engajamento dos mesmos, é possível verificar de quais formas o aumento no desempenho da transferência de dados e pré-visualização de imagens e vídeo podem trazer um ganho positivo para aplicativos móveis.

Segundo definição da ISO FDIS 9141-210, a experiência de usuário (UX) é definida como “a percepção e respostas de uma pessoa, como resultado do uso e/ou antecipação do uso de um produto, sistema ou serviço”. Outros também dão uma definição mais ligada aos sentimentos e pessoas do que o próprio produto em si. Desse modo, UX pode ser definida como “um sentimento primariamente avaliatório (bom-ruim) e momentâneo, enquanto se interage com um produto ou serviço” (HASSENZAHN, 2008, tradução nossa)<sup>1</sup>.

Embora a UX inclua a percepção ou sentimento de uma pessoa sobre um produto ou serviço, isso não quer dizer que uma conexão lenta necessariamente implique em uma má experiência de usuário. Segundo Kaasinen et al. (2009), as expectativas criadas por um usuário sobre um determinado produto podem influenciar positiva ou negativamente sua percepção sobre sua experiência de uso:

If the expectations are low, user experience may be good even if the system is not perfect. People also evaluate the value they experienced with the mobile internet, and that may overcome the possible difficulties and thereby make the user experience positive. So, although the mobile internet system might not be perfect yet, it does not mean that the user experience has to be poor. (KAASINEN et al., 2009).

Com isso em mente, é possível compreender que mesmo diante de uma conexão móvel limitada, cenário atual de uma grande parcela da população mundial

---

<sup>1</sup> Original: “momentary, primarily evaluative feeling (good-bad) while interacting with a product or service.”

hoje conectada – ainda é possível entregar uma boa experiência de uso de seus aplicativos, adequada à realidade daquela conexão. Portanto, isso valida o fato de que empresas vem adaptando seus aplicativos para que os mesmos possam se adequar à realidade de diferentes cenários.

Pode-se dizer, então, que a velocidade de carregamento de conteúdo é relevante para a experiência final de uso de um produto. A baixa velocidade de carregamento de dados pode, nesse sentido, diminuir a qualidade da experiência de uso. Segundo Linden (2006), experimentos controlados na Amazon mostraram uma queda de 1% nas vendas a cada 100ms adicionais de tempo de carregamento na página. Resultados semelhantes foram apontados por Kohavi, Henne e Sommerfield (2007) com base em dados técnicos oriundos de um experimento da Google, que apontou uma diminuição da ordem de 20% no tráfego a cada 500ms adicionais no carregamento das páginas.

Embora alguns desses dados não tenham sido obtidos por meio de estudos realizados com foco em dispositivos móveis, é possível observar que a velocidade de carregamento do conteúdo importa para o usuário do sistema.

## 2.4 APLICATIVOS DE REFERÊNCIA

Limitações que *smartphones* possuem em mercados emergentes vão além da velocidade e qualidade de acesso a internet. Espaço e poder de processamento diminutos somam-se aos diversos fatores que usuários móveis e desenvolvedores precisam lidar (GOOGLE, INC., 2016a).

Diante desse problema, empresas surgiram com soluções mais radicais, como o desenvolvimento de aplicativos alternativos, voltados apenas à mercados emergentes. Desse modo, existiria uma segunda versão de um aplicativo, desenvolvido com o propósito de abordar os mesmos problemas de uma forma diferente, enfatizando a instabilidade das conexões e baixo poder de processamento, comum em muitos celulares presentes em países emergentes.

É importante evidenciar que uma solução tão radical pode ser inviável para muitos desenvolvedores de aplicativos (JIN et al., 2016), pois requer grandes mudanças de arquitetura da aplicação e também um novo projeto a ser desenvolvido. Mesmo assim, as técnicas presentes nessa solução podem ser parcialmente aproveitadas, de acordo com o contexto individual de cada aplicação.

Algumas empresas que decidiram desenvolver aplicativos alternativos e adaptados a esses mercados são Facebook, Google e Microsoft. Elas desenvolveram os aplicativos Facebook Lite, YouTube Go e Skype Lite, respectivamente, que trazem soluções adequadas para uso em conexões limitadas e aparelhos com versões antigas do Android, como visto na TABELA 1.

**TABELA 1** - Otimizações dos aplicativos para dispositivos e conexões limitadas

	Tamanho do Instalador (em MBs)	Versão Mínima do Android	Otimização para uso de dados
Facebook	70,66 MB	Android 5.1	Parcial
Facebook Lite	1,46 MB	Android 2.3	Sim
YouTube	21,24 MB	Android 4.1	Parcial
YouTube Go	8,08 MB	Android 4.1	Sim
Skype	35,77 MB	Android 4.0.3	Não
Skype Lite	14,54 MB	Android 4.0.3	Sim

Fonte: Google Play Store e configurações disponíveis nos aplicativos

Mesmo resolvendo problemas distintos, esses aplicativos possuem soluções em comum entre eles, embora todos compartilham do mesmo princípio: trazer uma alternativa ao aplicativo regular que funcione bem no contexto de uma conexão limitada.

Os três aplicativos trazem algumas das soluções descritas na TABELA 2, que lista as melhores práticas para o contexto móvel de mercados emergentes. São levadas em conta, principalmente, características que os diferem da realidade de acesso móvel em países já desenvolvidos.

A seguir, são listados e discutidos alguns dos problemas e quais soluções foram implementadas. Além disso, os aplicativos são comparados sempre que essa solução tiver sido implementada em mais de um dos aplicativos exemplos.



**TABELA 2** - Melhores práticas para o desenvolvimento de aplicativos Android

<b>Melhores práticas</b>	
Reduzir tamanho dos aplicativos	Otimizar recursos gráficos por meio de vetores, usar formatos de imagens com boas taxas de compressão.
	Gerar vários instaladores para diferentes densidades de tela quando os aplicativos possuir muitas imagens.
	Usar ProGuard para diminuir o tamanho dos instaladores.
	Ser seletivo em relação ao incluir Play Services API e bibliotecas de terceiros.
Controlar quantidade e custo de uso de dados	Permitir que o usuário selecione a qualidade de streaming de áudio ou vídeo, baixar imagens de menor resolução ou até mesmo não exibir imagens por padrão.
	Definir qual conteúdo deve ser acessado por Wi-Fi e armazenado em cache para acesso posterior.
	Prover uma tela de preferências de rede que pode ser acessada dentro do aplicativo e também por meio das configurações gerais do <i>smartphone</i> .
Otimizar uso de memória	Usar imagens menores em dispositivos com pouca memória
	Liberar memória quando a interface do aplicativo estiver oculta ou quando memória do sistema estiver limitada.
	Evitar o uso exagerado de serviços pois os mesmos consomem muita memória.
	Usar recursos do Android Studio para identificar problemas de uso ineficiente de memória e os solucionar.
Alcançar maiores audiências com versões populares do Android	Usar Android Support Library para permitir compatibilidade com versões anteriores do Android
	Usar Google Play Services para acessar APIs que são independentes da versão do Android
	Definir o nível de retro compatibilidade do aplicativo para que assim seja possível saber se alguma API usada não é compatível com versões antigas do Android

Fonte: Adaptado de (GOOGLE, INC., 2016a)

#### 2.4.1 Facebook Lite

O Facebook Lite traz soluções relacionadas a muitas das melhores práticas descritas na TABELA 2: seu instalador possui tamanho reduzido; é compatível com versões antigas do sistema; otimiza o download de imagens e dados; armazena dados trafegados em cache para acesso off-line; exige baixo poder de processamento e é otimizado para usar memória RAM de forma eficiente (FACEBOOK INC., 2016).

Dessa forma, ele se diferencia dos outros dois aplicativos citados nesse capítulo pelo fato de ir mais longe em suas soluções, sendo, portanto, o melhor exemplo de boas práticas de adequação à mercados emergentes.

O Facebook Lite é uma versão do aplicativo móvel do Facebook voltado a mercados emergentes, com o foco em usuários que possuem conexão lenta e/ou aparelhos que possuem versões mais antigas do Android. Seu instalador possui em média apenas 1,5MB, contra 68MB do instalador da versão regular. Isso traz duas vantagens principais para os usuários: seu *download* acontece cerca de 45 vezes mais rápido que o aplicativo regular e ele ocupa pouco espaço interno de armazenamento. Isso é um fator muito importante, pois aparelhos Android mais baratos tendem a ter um espaço interno bastante limitado (GOOGLE, INC., 2016b), algumas vezes menores que o próprio tamanho total do instalador normal do Facebook.

O Facebook Lite também suporta aparelhos Android rodando a versão 2.3 ou superior. Dessa forma, é possível atingir um número maior de pessoas, uma vez que o aplicativo regular não suporta versões mais antigas do sistema Android.

O aplicativo também é otimizado para inicializar mais rápido e usar menos dados para carregar dados, seja realizando menos chamadas HTTP, seja carregando fotos em tamanhos menores que o normal, provocando uma economia no uso da franquia de dados.

Embora o Facebook Lite seja projetado inteiramente focando em usuários de países emergentes, ele também pode ser usado em países desenvolvidos em regiões onde o suporte a 2G ainda é o único meio de acesso disponível, como acontece principalmente em áreas rurais.

É importante notar que o Facebook Lite sacrifica várias funcionalidades e animações em prol de uma otimização mais radical. Desse modo, usuários com aparelhos e conexões rápidas e constantes (maioria em países desenvolvidos) podem não tirar muito proveito das otimizações do aplicativo.

Dentre todos, todavia, ele é o de mais difícil implementação, uma vez que sua arquitetura se distancia grandemente da forma que um aplicativo normalmente é desenvolvido. Por exemplo, de acordo com sua equipe de desenvolvimento, suas telas são geradas na nuvem e, então, baixadas e renderizadas localmente. Porém, é importante notar que o Facebook é repleto de pequenas soluções únicas, das quais algumas são analisadas no decorrer do próximo capítulo.

### 2.4.2 YouTube Go

O YouTube Go é uma versão alternativa ao aplicativo YouTube disponível em versão de testes para usuários Android na Índia.

O aplicativo busca resolver os principais problemas relacionados à baixa qualidade do acesso móvel à internet na Índia e outros países emergentes, que dificultam a visualização de vídeos em uma conexão que não seja rápida (GOOGLE, INC., 2016c).

O YouTube Go foi projetado para funcionar sem conexão com a internet na maior parte do tempo, pois permite o *download* de vídeos para posterior acesso *off-line* pelo usuário. Usuários também podem compartilhar, por meio de uma rede local, vídeos já baixados com outros usuários que estejam próximos. Desse modo, o tráfego necessário para o download de um arquivo só precisa ser consumido uma vez.

Quando o usuário clica em algum vídeo, surge uma janela mostrando uma prévia de baixa resolução com alguns quadros de diferentes partes dos vídeos, de modo que seja possível saber de que se trata o vídeo, antes dele ser baixado ou exibido. Essa mesma tela também dá aos usuários controle sobre a qualidade (e, conseqüentemente, o tamanho) do vídeo a ser baixado.

A equipe de desenvolvimento do aplicativo não deu mais detalhes técnicos sobre como as melhorias foram realizadas, portanto, não é possível avaliar de forma eficiente o grau de dificuldade de implementação das soluções presentes. Porém, devido à natureza simples de algumas soluções, como o *download* e o compartilhamento local de vídeos, é possível supor que elas são de fácil implementação.

### 2.4.3 Skype Lite

O Skype Lite é uma versão alternativa ao Skype desenvolvida pela Microsoft com foco em usuários da Índia, com o intuito de “entregar uma melhor performance em condições de rede desafiadoras, usar menos dados e menos energia” (MICROSOFT CORPORATION, 2017, tradução nossa), afim de alcançar mais usuários.

O aplicativo possui um monitor de uso de dados que mostra ao usuário a quantidade de dados que o mesmo utilizou, por Wi-Fi ou rede móvel.

Embora dados mais completos não tenham sido providos pela equipe de desenvolvimento do Skype Lite, o aplicativo foi otimizado de forma a usar menos dados para o envio e recebimento de mensagens e realização de chamadas de áudio e vídeo. O aplicativo também é otimizado de forma a usar menos poder de processamento, aumentando a duração da bateria principalmente em aparelhos mais antigos.

## 2.5 CONSIDERAÇÕES FINAIS

O presente capítulo mostrou uma visão ampla do cenário de uso da internet pelo mundo, principalmente em países emergentes, junto com a forma que grandes empresas lidam com os problemas de conexão presentes nesses mercados.

Dessa forma, será possível analisar mais detalhada as soluções criadas com foco em mercados em desenvolvimento no capítulo a seguir.

### 3 ANÁLISE DAS SOLUÇÕES DISPONÍVEIS

#### 3.1 CONSIDERAÇÕES INICIAIS

Ao decorrer deste capítulo, são analisadas as soluções presentes nos três aplicativos exibidos no capítulo anterior. Além destas, também foram introduzidas tecnologias que implementam soluções similares.

Para esse trabalho, foram usados como referência três aplicativos desenvolvidos pelas empresas Facebook, Google e Microsoft, que desenvolveram o Facebook Lite, YouTube Go e Skype Lite, respectivamente.

Este capítulo foi dividido em tópicos de acordo com soluções, em vez de analisar os aplicativos por completo, pois os mesmos compartilham soluções semelhantes, embora eles, em uma perspectiva maior, possuem propostas diferentes e resolvem problemas distintos. Dessa forma é possível comparar soluções entre diferentes projetos.

Para a melhor compreensão da análise vale a pena salientar que a mesma será conduzida levando-se em conta os seguintes critérios: adaptação de imagens, tamanho de instaladores, transferência de dados, exibição instantânea de páginas *web* e exibição prévia de imagens.

#### 3.2 ADAPTAÇÃO DE IMAGENS

Imagens compõem uma grande parte do conteúdo disponível na internet. Embora seja difícil estimar qual porcentagem do tráfego total de internet seja composto de imagens, alguns estudos trazem estimativas. Agababov et al. (2015) traz dados mais abrangentes sobre quais tipos de dados trafegam pela rede, por meio de análise de uso do *Flywheel*, um proxy HTTP que realiza compressão em respostas entre cliente e servidores. Segundo o artigo, 74% de todos os dados passados pelo *Flywheel* eram imagens. Textos, por sua vez, compreendiam menos de 1% do tráfego. Mesmo otimizando outros tipos de conteúdo, como HTML, JavaScript, CSS e texto, entre outros tipos de mídias, a otimização das imagens por si só correspondiam por 85% de toda a otimização feita pela ferramenta.

Existem diversas possibilidades de se realizar a otimização de imagens. Nesse tópico, são mostradas otimizações diferentes da realizada pela Google. Uma vez que

o *Flywheel* foi projetado para otimizar conteúdo de *websites* sem que os mesmos sejam alterados, ele foge do escopo desse trabalho, pois requer ações do usuário (ativar o recurso de compressão de dados no navegador Chrome) e não dos desenvolvedores (a otimização acontece por meio de um *proxy*, e não nos servidores de onde os dados são originalmente acessados).

Uma das opções de otimização de imagens é por meio de usos de algoritmos de compressão de imagens. WebP é um formato de imagens patenteado pela Google que traz uma melhor compressão de imagens que seus concorrentes como JPEG e PNG. Segundo Nagar et al. (2015, tradução nossa), o WebP “oferece uma compressão 30% maior que que JPEG e PNG, enquanto retém uma qualidade de imagem excelente”<sup>2</sup>. A TABELA 3 mostra mais detalhes sobre as principais vantagens de cada formato de compressão.

**TABELA 3** - Comparação de formatos de imagens

	APNG	PNG	GIF	JPEG	WebP
Compressão	Boa	Boa	Regular	Boa	Melhor
Compressão sem perdas de dados	Sim	Sim	Sim	Sim	Sim
Compressão com perda de dados	Não	Não	Sim	Sim	Sim
Transparência	Sim	Sim	Binária	Não	Sim
Animação	Sim	Não	Sim	Não	Sim
Qualidade	Boa	Boa	Regular	Boa	Boa

Fonte: Adaptado de (NAGAR, MOHAPATRA e CHAKI, 2015)

Embora o problema de compatibilidade limite o uso de WebP em websites, ele pode ser utilizado em aplicações móveis e em servidores, por meio de bibliotecas oficiais publicadas pela Google e por terceiros, sem possuir grandes problemas de compatibilidade.

Uma outra forma de otimizar imagens é por meio de redimensionamento baseado na resolução do dispositivo, que permite transformar imagens de modo que elas sejam otimizadas e adequadas a resolução e densidade de pixel de cada aparelho.

<sup>2</sup> Original: “It offers about 30% more compression than JPEG and PNG and retains an excellent image quality”.

De acordo com dados publicados pela Google, capturados a partir dos dispositivos que acessaram a Google Play Store em um período de 7 dias (dados acessados no dia 23 de março de 2017), 88,7% dos dispositivos possuíam uma resolução de no mínimo 470dp por 320dp e no máximo 640dp por 320dp. Cerca de 37,1 e 34,6% dos aparelhos possuíam uma densidade de tela HDPI (cerca de 240dpi) e XHDPI (cerca de 320dpi), respectivamente (GOOGLE, INC., 2017). A TABELA 4 mostra dados mais específicos sobre cada resolução.

**TABELA 4 - Resoluções de tela utilizadas no Android**

	ldpi	mdpi	tvdpi	hdpi	xhdpi	xxhdpi	Total
Small	1,1%	<0,1%	<0,1%	<0,1%	<0,1%	<0,1%	1,1%
Normal	<0,1%	2,5%	0,2%	36,1%	33,6%	16,3%	88,7%
Large	0,1%	3,7%	1,8%	0,4%	0,4%	<0,1%	6,4%
Xlarge	<0,1%	2,6%	<0,1%	0,6%	0,6%	<0,1%	3,8%
Total	1,2%	8,8%	2,0%	37,1%	34,6%	16,3%	100%

Fonte: Adaptado de (GOOGLE, INC., 2017)

Existem *softwares* que permitem automatizar o redimensionamento de imagens para essas telas. Neste estudo, a Cloudinary foi usada como ferramenta exemplo para isso. A Cloudinary é uma empresa que oferece uma solução SaaS (software como serviço, do inglês *Software as a Service*) que permite o gerenciamento de mídias na nuvem. O serviço permite o upload, armazenamento, transformação, gerenciamento e entrega de imagens por meio de uma interface web, API e integração com diversas plataformas (CLOUDINARY, 2017).

Uma das funcionalidades oferecidas é a transformação de imagens por meio de sua API e SDK disponível para diversas plataformas. As transformações permitem requisição de imagens baseadas em diversos critérios, como dimensionamento, nível de compressão, corte e outras manipulações diversas.

Todo o processamento acontece nos servidores da Cloudinary. Manipulações podem ser feitas por meio de *queries* na URL ou utilizando seus *frameworks* de desenvolvimento, disponíveis para muitas plataformas.

Além do Cloudinary, existem outros *softwares* que podem ser integrados aos servidores de hospedagens de conteúdo que fazem transformações semelhantes, muitos dos quais são gratuitos ou de código e uso livre.

### 3.2.1 Impactos do redimensionamento de imagens

Os dados usados nessa seção foram capturados a partir de várias requisições de imagens, estas redimensionadas levando em consideração diversas resoluções de dispositivos. Foram consideradas como resoluções relevantes aquelas mais utilizadas nos dispositivos Android (TABELA 5). Também foram utilizadas duas resoluções adicionais durante as requisições: a resolução original da imagem baixada e outra com 128px de largura.

**TABELA 5** – Dimensões das resoluções mais utilizadas no Android

nome	mdpi	hdpi	xhdpi	xxhdpi
largura	320	480	720	960
altura	480	800	1280	1600

Fonte: adaptado de (Google, Inc., 2017)

Os dados foram gerados por meio de um algoritmo que faz requisições de imagens hospedadas na Cloudinary. O programa, uma aplicação desenvolvida em Angular 1.5, foi executado no navegador Google Chrome, versão 57, em um MacBook Pro equipado com 8 GB de memória DDR3 1600 MHz e processador Intel Core i5 de 2,6 GHz.

Por meio desse programa, foi possível alterar parâmetros nas URLs das imagens que retornam imagens baseadas em seus valores. Foram realizados apenas transformações de redimensionamento na imagem, embora os serviços da Cloudinary e ferramentas semelhantes permitam que outras modificações, como mudança de formato, corte, compressão, também sejam aplicadas. Isso aconteceu porque esta análise procura mostrar a eficácia que a otimização do tamanho das imagens traz para o consumo de tráfego apenas.



O algoritmo escrito para capturar os dados possui uma função que realiza requisições a sete imagens, de acordo com a largura de cada item presente em um *array* de resoluções (CÓDIGO 1).

**CÓDIGO 1** - Função que realiza requisição das imagens de acordo com a resolução desejada

```

1  function requestImages(id, resolution) {
2      var size = resolution ? ('c_scale,w_' + resolution + '/' + id) : id
3
4      return $http({
5          method: 'GET',
6          url: 'http://res.cloudinary.com/x/image/upload/' + size + '.jpg',
7          responseType: 'arraybuffer'
8      })
9  }

```

Fonte: próprio autor

As imagens utilizadas foram conseguidas na página oficial da Internet.org no Facebook, portanto, são imagens no tamanho real oferecido pelo aplicativo Facebook para Android.

Em seguida, foram feitas requisições dessas imagens, todas de uma única vez (CÓDIGO 2, linha 8), de forma semelhante a como geralmente ocorre em aplicativos, por meio de uma função que resolve uma promessa assim que todas as requisições são cumpridas (CÓDIGO 2, linha 9). Em JavaScript, uma promessa “é um objeto usado para processamento assíncrono [...], (representando) um valor que pode estar disponível agora, no futuro ou nunca” (MOZILLA DEVELOPER NETWORK, 2017).

**CÓDIGO 2** - Função que executa a requisição de todas as imagens e retorna uma promessa resolvida em seguida

```

1  var getImages = (width, times, cb) => {
2      var seconds = [];
3      times = times || 0;
4
5      var makeRequests = (index) => {
6          index = index || 0;
7
8          Promise.all(images.map((value) => requestImages(value, width)))
9              .then(results => {
10                 results.forEach((result, index, array) => {
11                     seconds[index] = seconds[index]
12                         ? seconds[index] + result.duration
13                         : result.duration;
14                 })
15
16                 if (index < times - 1) {
17                     makeRequests(index + 1)
18                 } else {

```

```

19         console.log(index, times, seconds)
20         seconds.forEach((second, index, array) => {
21             array[index] = array[index] / times
22         })
23
24         cb(seconds)
25     }
26 }
27     .catch(err => console.log(err))
28 }
29 makeRequests();
30 }

```

Fonte: próprio autor

Foi escrito um interceptador de requisições HTTP, como visto em CÓDIGO 3, que altera o cabeçalho de cada requisição, adicionando um campo que contém o tempo exato em que a requisição é iniciada (linha 14). Logo em seguida, quando a requisição retorna uma resposta, o tempo exato do fim da requisição também é registrado (linha 20). Após isso, é realizado o cálculo de duração entre a requisição e a resposta, em milissegundos (linha 21).

### CÓDIGO 3 - Interceptador que adiciona tempo de início e fim nas requisições HTTP

```

1  app.config(function (CloudinaryProvider, $httpProvider) {
2      CloudinaryProvider.configure({
3          cloud_name: 'cloudinary_name',
4          api_key: 'cloudinary_api_key'
5      });
6
7      $httpProvider.interceptors.push('imageLogger');
8  })
9
10 app.factory('imageLogger', function () {
11     return {
12         request: function (config) {
13             if (config.url.indexOf('html') == -1) {
14                 config.start = new Date();
15             }
16             return config;
17         },
18         response: function (response) {
19             if (response.config.url.indexOf('html') == -1) {
20                 response.end = new Date()
21                 response.duration = response.end - response.config.start;
22             }
23             return response;
24         }
25     }
26 }

```

Fonte: próprio autor

Quando a promessa é resolvida (nesse contexto, uma promessa resolvida equivale ao momento em que as requisições das sete imagens são finalizadas sem

erros), o tempo de duração de cada requisição imagem é guardado em um *array* (CÓDIGO 2, linha 11).

Ao fim do último grupo de requisições por resolução de tela, a média da duração de todas as requisições para cada imagem é calculada, como indicada na linha 21 do CÓDIGO 2.

Após isso, uma *callback* é retornada (CÓDIGO 2, linha 24), contendo um *array* com todos os tempos médios de todas as requisições relacionadas à resolução atual. A função principal recebe o *array*, projeta seus dados na interface e é novamente executada de forma recursiva (CÓDIGO 4, linha 11).

#### CÓDIGO 4 - Função recursiva que é executada por todos os itens de uma coleção de resoluções de telas

```
1  var runTest = (index) => {
2    index = index || 0
3
4    if (index < screenSize.length) {
5      getImages(screenSize[index].width, 10, (seconds) => {
6        $timeout(() => {
7          this.rows.push({
8            title: screenSize[index].name,
9            seconds: seconds
10         })
11         runTest(index + 1)
12       })
13     })
14   }
15 }
```

Fonte: próprio autor

Assim que esse tempo médio é calculado, é verificado se alguma outra resolução de tela precisa ser testada e, então, a função é executada de forma recursiva novamente (CÓDIGO 4, linha 11). Todo esse processo se repete até que sejam realizadas requisições usando a largura de cada um dos itens da coleção de resoluções.

Todo o processo acontece de forma assíncrona, onde várias requisições acontecem ao mesmo tempo, semelhante ao funcionamento dos aplicativos no mundo real. Não foram feitos testes com apenas uma requisição de cada vez pois isso não reflete o comportamento de aplicativos disponíveis hoje para *smartphones*.

O teste é, dessa forma, repetido em cenários com diferentes velocidades de conexão. Foram usadas as velocidades padrão disponíveis nas ferramentas de desenvolvimento do Google Chrome, que traz um simulador de *Network Throttling*,

que simula limitações de largura de banda. Foram usadas as seguintes velocidades disponíveis na TABELA 6.

**TABELA 6** - Velocidades de requisições usadas no experimento

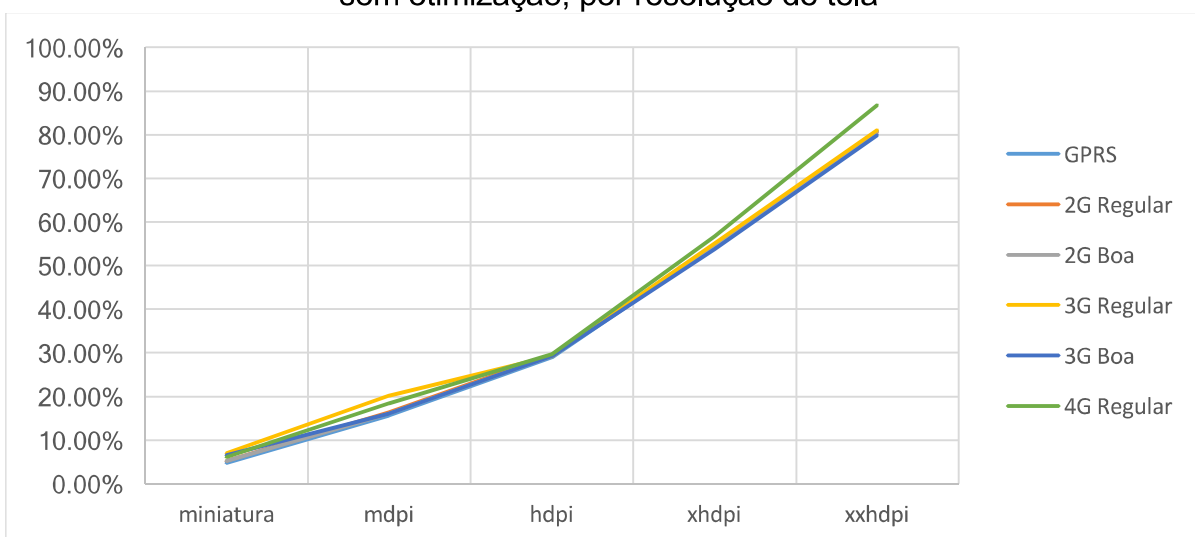
Velocidade	2G regular	2G Bom	3G Regular	3G Bom	4G Bom
Download	250 kb/s	450 kb/s	750 kb/s	1.5Mb/s	4Mb/s
Upload	50 kb/s	150 kb/s	250 kb/s	750 kb/s	3Mb/s
Latência	300ms	150ms	100ms	40ms	20ms

Fonte: Google Chrome Developer Tools, versão 57.

Os resultados foram consistentes em todos os cenários de velocidades aplicados, e mostram que apenas o redimensionamento de imagens traz ganhos consideráveis no carregamento de imagens.

Em todos os testes, como esperado, os resultados mostraram que quanto menor a dimensão da imagem, menor o seu tempo de carregamento, como visto no GRÁFICO 5, que mostra o tempo gasto de requisições em relação à uma requisição sem otimização alguma (o tempo total sem otimização reflete o valor de 100% do tempo de carregamento).

**GRÁFICO 5** – Relação entre tempo gasto com requisição otimizada e o tempo gasto sem otimização, por resolução de tela



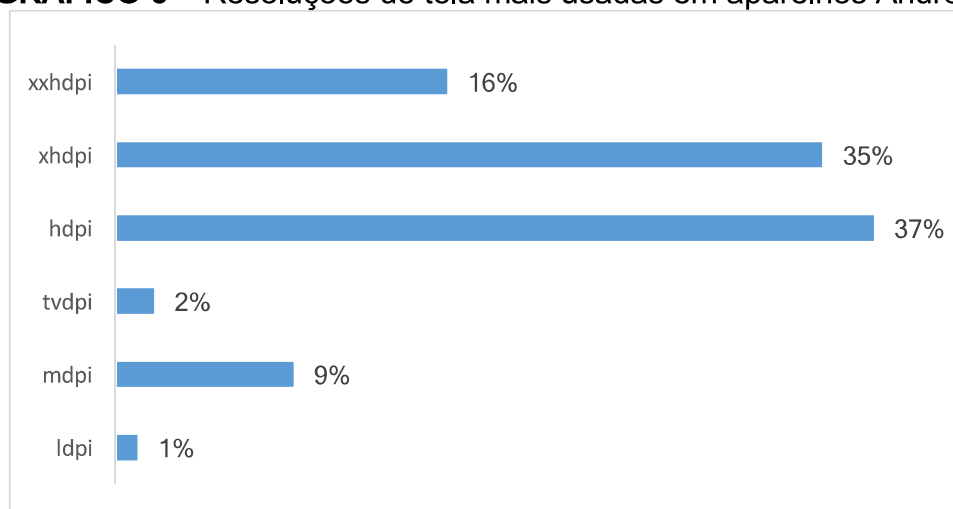
Fonte: próprio autor

Porém, ao analisar os dados capturados junto aos dados de uso por resolução de tela de dispositivos Android, percebe-se o impacto positivo que essa otimização é

capaz de trazer, em número de usuários alcançados. Vale notar, porém, que as estatísticas de uso não apontam qual porcentagem desses aparelhos estão localizadas em países emergentes. Por outro lado, é esperado que países em desenvolvimento possuam mais aparelhos com requisitos de hardware (resolução, processamento, etc.) inferiores aos de países desenvolvidos.

Dados gerados mostram que usuários com a resolução mais utilizada (GRÁFICO 6) no Android, hdpi, presente em 37% dos aparelhos Android, carregariam todas as imagens em um tempo máximo médio de 8,4 segundos, apenas 30% do tempo médio da requisição em resolução normal. Um ganho significativo de performance de carregamento de dados. A segunda resolução mais utilizada, a xhdpi, presente em 36% dos aparelhos, realizaria o carregamento em 15,5 segundos, ou 54% do tempo de requisição normal.

**GRÁFICO 6** – Resoluções de tela mais usadas em aparelhos Android

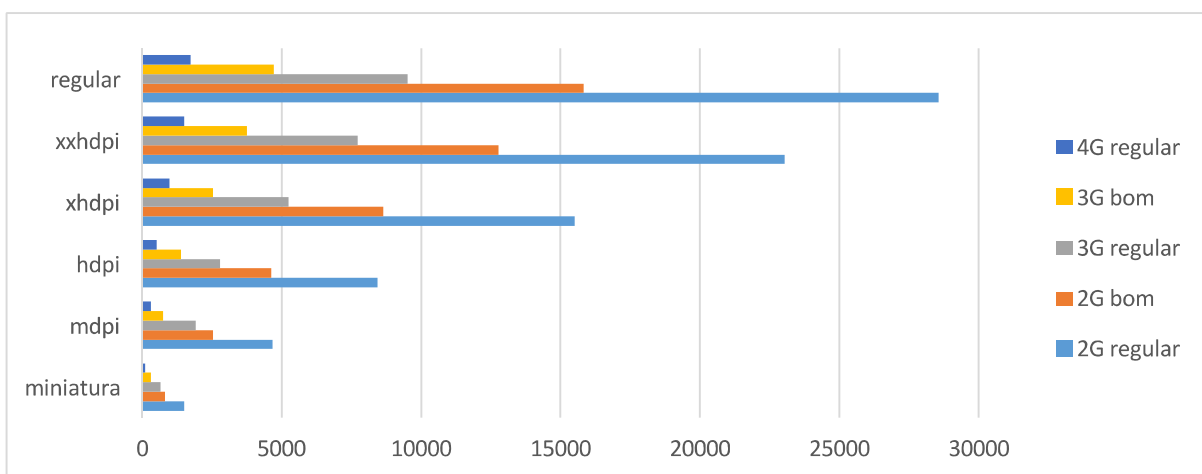


Fonte: Adaptado de (GOOGLE, INC., 2017)

Outras telas, embora possuam uma parcela menor de usuários, também trazem melhorias muito relevantes, conforme mostrado no GRÁFICO 6.

O algoritmo trouxe velocidade semelhante em todos os cenários de velocidades de conexão testados, conforme mostrado no GRÁFICO 7, que mostra que o tempo de carregamento das imagens em resolução regular em uma conexão “3G boa” é semelhante ao tempo de carregamento de imagens otimizadas a um aparelho hdpi com conexão 2G com boa qualidade.

**GRÁFICO 7** – Tempo médio de carregamento das imagens, em milissegundos, por tamanho de tela



Fonte: próprio autor

GRÁFICO 8 e GRÁFICO 9 trazem maiores detalhes do tempo de carregamento de cada imagem em cada resolução de tela.

**GRÁFICO 8** – Tempo de requisição de imagens em conexões móveis lentas

GPRS								
Largura	img1	img2	img3	img4	img5	img6	img7	Tempo total
miniatura	4892	3908	4049	5314	5736	6064	6744	6744
mdpi	19709	11061	17858	21373	17061	19475	22100	22100
hdpi	36147	20796	34249	40734	31461	36077	40922	40922
xhdpi	67639	38998	68717	76920	57865	67521	75303	76920
xxhdpi	100464	60127	103557	114293	86072	99268	111901	114293
Regular	128633	102594	133157	141548	108593	126992	124085	141548
2G Regular								
Largura	img1	img2	img3	img4	img5	img6	img7	Tempo total
miniatura	1169	934	981	1215	1262	1309	1497	1497
mdpi	4071	2337	3686	4399	3527	4000	4670	4670
hdpi	7362	4381	7086	8427	6552	7410	8426	8427
xhdpi	13576	7895	13849	15495	11660	13558	15264	15495
xxhdpi	20220	12161	20876	23023	17374	19961	22568	23023
Regular	25200	19950	26832	28562	22538	26086	24707	28562
2G Boa								
	img1	img2	img3	img4	img5	img6	img7	Tempo total
miniatura	613	499	532	663	681	733	827	827
mdpi	2211	1265	2072	2466	1950	2302	2546	2546
hdpi	4064	2374	3905	4610	3548	4095	4628	4628
xhdpi	7534	4465	7753	8652	6487	7605	8479	8652
xxhdpi	11208	6729	11604	12784	9656	11169	12523	12784
Regular	14188	11185	14881	15834	12456	14451	13747	15834

Fonte: próprio autor.

**GRÁFICO 9** – Tempo de requisição de imagens em conexões móveis mais rápidas

3G Regular								
Largura	img1	img2	img3	img4	img5	img6	img7	Tempo total
miniatura	405	371	369	455	505	554	655	655
mdpi	1448	1090	1395	1625	1426	1737	1913	1913
hdpi	2363	1456	2323	2793	2275	2454	2757	2793
xhdpi	4423	2493	4698	5253	3917	4666	5054	5253
xxhdpi	6663	3994	6984	7726	5916	6875	7563	7726
Regular	8116	6283	8963	9524	7566	8674	7868	9524
3G Boa								
Largura	img1	img2	img3	img4	img5	img6	img7	Tempo total
miniatura	169	145	165	200	231	225	309	309
mdpi	517	338	580	737	630	734	756	756
hdpi	1121	591	1154	1379	1147	1308	1375	1379
xhdpi	2088	1212	2276	2533	2005	2302	2467	2533
xxhdpi	3173	1881	3399	3760	2964	3372	3661	3760
Regular	3818	2907	4301	4710	3867	4476	3730	4710
4G Regular								
Largura	img1	img2	img3	img4	img5	img6	img7	Tempo total
miniatura	64	60	66	81	93	100	107	107
mdpi	162	140	216	242	283	289	320	320
hdpi	355	247	400	504	431	485	515	515
xhdpi	782	421	869	984	721	889	946	984
xxhdpi	1231	644	1316	1507	1086	1261	1461	1507
Regular	1218	1151	1531	1737	1438	1691	1365	1737

Fonte: próprio autor.

Cores vermelhas representam tempos maiores, enquanto que cores verdes representam tempos menores. O tempo total presente nos gráficos representa o tempo completo da requisição e não a soma de todos os tempos.

Assim sendo, a otimização de imagens por meio de redimensionamentos dinâmicos traz ótimos ganhos de performance de carregamento. É importante frisar, também, que ganhos assim trazem economias de dados tanto em clientes como servidores, resultando, então, em economia de dados e, por consequência, financeira, aos usuários e desenvolvedores.

### 3.2.2 Implementação

Essa solução pode ser implementada de maneira simples. Ferramentas que se integram ao ambiente de desenvolvimento podem ser facilmente encontradas com

diversas opções de licença, sejam elas pagas ou gratuitas. Desse modo, desenvolvedores podem trabalhar na implementação desta solução com pequenas mudanças no código dos aplicativos.

Dado o grau de dificuldade de implementação dessa técnica, ela se destaca como uma boa forma de reduzir drasticamente a quantidade de dados em mercados emergentes.

### 3.3 TAMANHO DO INSTALADOR

Todos possuem instaladores menores que o aplicativo regular principal. O instalador do Facebook Lite, por sua vez, consiste em apenas 2% do tamanho do aplicativo padrão do Facebook para Android. É visível que isso promove um impacto positivo no espaço de armazenamento dos aparelhos, que são muitas vezes limitados. Essa solução, sozinha, reduz o tempo de download e, conseqüentemente, o tráfego de dados utilizado para que ocorra a instalação do mesmo.

Essa solução impacta o tamanho do espaço de armazenamento interno do aparelho que será utilizado após a instalação do aplicativo. Também afeta o custo da conexão e tráfego de dados necessário para o download do mesmo, que costuma ser elevado em países não desenvolvidos. Segundo (GOOGLE, INC., 2016b), é recomendável deixar o tamanho do instalador de uma aplicação Android com o menor tamanho possível, pois isso estimula a instalação do mesmo e evita que o ele seja removido quando o aparelho do usuário estiver com pouco espaço de armazenamento disponível.

O Facebook Lite é o que traz a maior diminuição no tamanho do instalador em relação a seu aplicativo regular. Seu instalador possui em média apenas 1,5MB, tamanho muito reduzido quando comparado aos 71MB do instalador normal do Facebook. Isso traz duas vantagens principais para seus usuários: seu download acontece cerca de 45 vezes mais rápido que o aplicativo normal e ele ocupa pouco espaço interno de armazenamento, menos que uma foto gerada pela câmera do celular.

Skype Lite e YouTube Go são mais conservadores no que diz respeito à redução do tamanho do instalador. Embora não possuam aplicativos regulares tão grandes como o Facebook, suas versões otimizadas são em média 60% menores que os mesmos. Ainda assim, resultam em um instalador relativamente maior que o



gerado pelo Facebook Lite. A TABELA 7 mostra o ganho que cada versão otimizada traz.

**TABELA 7** - Relação entre tamanho dos aplicativos regulares e otimizados

Nome	Tamanho (em MB)		Ganho
	Regular	Reduzido	
Facebook	70,66	1,46	98%
YouTube	21,24	8,08	62%
Skype	35,77	14,54	59%

Fonte: Google Play Store.

Como esperado, o tamanho do aplicativo influencia principalmente em duas características dos celulares, que tendem a ser mais limitadas em países emergentes: espaço disponível para armazenamento e custo de acesso à internet móvel. Isso, portanto, torna-se um fator muito importante no momento de decisão entre baixar ou não o aplicativo, pois aparelhos Android mais baratos, que são muito presentes em mercados emergentes, tendem a ter um espaço interno mais limitado (GOOGLE, INC., 2016b).

### 3.4 TRANSFERÊNCIA DE DADOS

A preocupação com uso de espaço continua após a instalação dos aplicativos, pois é sabido que muitos aplicativos realizam transferências de dados pela rede. Todos os três aplicativos abordam de maneira diferente esse problema, pois eles possuem soluções diferentes para problemas individuais.

O Facebook Lite lida com atualizações de *status*, sejam elas texto, vídeos, imagens ou links externos. Por design, ele é eficiente no uso de dados. Segundo informações divulgadas pelo time de desenvolvimento do Facebook Lite, foram realizadas mudanças deveras drásticas: em vez de usar HTTP, foi usado um protocolo TSL diretamente acima da camada de transporte TCP, de forma a estabelecer uma conexão rápida e eficiente, mesmo em conexões 2G.

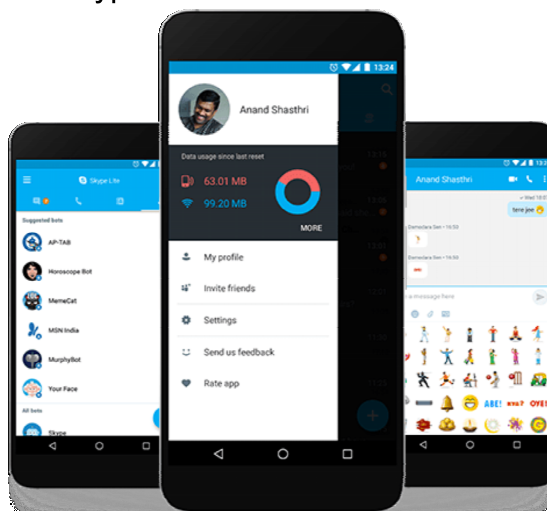
Transformações também são aplicadas as imagens, de forma semelhante à solução proposta na seção 3.2:

Most of the app's data usage comes from images, so this was an area where we wanted to reduce data usage in the app. We control the size of the images to be able to control the amount of data used. The Lite server knows exactly the screen width, height, and screen density of the client. Instead of talking to CDNs directly, the Lite client receives exact-size images over the same TCP connection from the Lite server. The server can adjust the JPEG quality of the images and format images exactly at the size needed, rather than use the few supported sizes in the CDN. The exact size and JPEG quality optimizations make a lot of the images small; when images are large, they are chunked for transport. The Lite image server also provides caching and transcoding of images. (FACEBOOK INC., 2016)

Algumas das soluções propostas pelo Facebook Lite podem se tornar inviáveis para alguns projetos, pois necessitam de mudanças mais drásticas na arquitetura do sistema, o que resulta em altos custos de desenvolvimento e manutenção. Mas outras soluções, como a transformação de imagens, podem ser de fácil implementação, como mostrado anteriormente. Apesar disso, soluções abertas como GraphQL podem ser usadas para diminuir a quantidade e tamanho de requisições HTTP, sem grandes dificuldades de implementação.

Skype Lite, por sua vez, foi otimizado de modo a consumir menos dados no envio e recebimento de imagens e mensagens, além de otimizar chamadas de voz e vídeo para que funcionem bem até mesmo em conexões 2G. Ele também traz um monitor de uso de dados (**Figura 1**) que diferencia tráfego Wi-Fi e dados móveis, com o propósito de mostrar ao usuário de forma distinta como os dados estão sendo consumidos pelo aplicativo (MICROSOFT CORPORATION, 2017).

**Figura 1** - Skype Lite destacando o monitor de dados



Fonte: (MICROSOFT CORPORATION, 2017)

O YouTube Go, por sua vez, aborda o problema de forma diferente. O propósito principal do YouTube é a exibição de vídeos, o que é algo altamente afetado pela velocidade de conexão do usuário. Como é notório, vídeos podem facilmente chegar a dezenas de *megabytes* (MB), o que torna seu consumo difícil em conexões limitadas.

Para solucionar esse problema, a Google desenvolveu este aplicativo alternativo que permite que os vídeos sejam armazenados na memória interna do aparelho, para que possam ser consumidos sem uma conexão à internet. Dessa forma, seus usuários poderão assistir os vídeos sem pausas assim que o download do arquivo terminar. Outro problema resolvido por essa solução é a exibição do vídeo não seja cancelada em caso de problemas de rede, já que o YouTube Go pode recomeçar o download após uma queda de conexão.

Vídeos também podem ser compartilhados por usuários sem a necessidade de um novo download. Usuários que já possuem um vídeo podem compartilhar por uma rede Wi-Fi local seus arquivos com outros usuários. Assim, a franquia de dados só precisará ser consumida em uma conta/aparelho (GOOGLE, INC., 2016c). Esse processo não pode acontecer de maneira totalmente *off-line*, uma vez que conexão a internet é necessária para validar os arquivos de vídeos por meio dos servidores da Google. Essa requisição, entretanto, consome menos de 1KB de tráfego.

### 3.5 EXIBIÇÃO INSTANTÂNEA DE PÁGINAS WEB

O *Accelerated Mobile Pages* (AMP) é um projeto de código aberto criado pela Google que permite que páginas da web sejam carregadas de forma mais rápida do que páginas web feitas em HTML tradicional (BESBRIS, 2015).

Para adaptar uma página existente para o AMP, é preciso importar um script *JavaScript* no site e realizar adaptações no código do mesmo, por meio de elementos padrão do HTML como também específicas do projeto AMP.

Diferente de plataformas semelhantes, como *Facebook Instant Articles* e *Apple News*, o AMP é um projeto de código aberto. Isso permite que ele seja integrado a qualquer aplicativo. Em geral, aplicativos que distribuem conteúdo externos de websites podem tirar grande proveito do AMP, uma vez que ele é focado em distribuir primariamente conteúdo baseado em textos, embora também permita o uso de imagens e vídeos dentro dos artigos.

O projeto é criado em cima de três pilares principais: conteúdo, distribuição e anúncios. Dessa forma, publicadores de conteúdos podem criar páginas ricas em conteúdo que contém mapas, imagens e vídeos, carrosséis, botões de compartilhamento sociais, etc. de forma rápida e disponível para diversos dispositivos e lugares do planeta. Por esta razão, o conteúdo será hospedado em servidores de cache da Google sem custo algum. Ao mesmo tempo que entrega essas facilidades, o AMP permitirá que os publicadores integrem suas páginas a diversos formatos e redes de anúncios, ao mesmo tempo que a velocidade de carregamento que o AMP proporciona continua sendo preservada (AMP, 2016).

### 3.6 EXIBIÇÃO PRÉVIA DE IMAGENS

Além da adaptação de imagens mostrada na seção 3.2, também é possível melhorar a experiência de uso com a introdução de formatos de exibição prévia de imagens.

De acordo com o time de desenvolvimento do aplicativo móvel do Facebook, a imagem de capa presente na página de perfil de usuários demorava muitos segundos para ser carregada em uma conexão 2G (CABRAL e KANDROT, 2015). O tamanho delas costumava passar de 100KB até mesmo quando comprimidos. Além disso, o tempo de carregamento crescia ainda porque era necessário realizar uma segunda requisição HTTP para que as imagens fossem carregadas.

Para solucionar esse problema de tempo de carregamento, a equipe de engenheiros do Facebook decidiu criar um mecanismo de prévia de imagens no aplicativo Facebook. Para tanto, é mostrada uma imagem bastante pequena junto a um filtro borrado na mesma. Desse modo, usuários podem exibir uma prévia da imagem instantaneamente, durante a requisição de rede original à API do Facebook, enquanto a imagem principal está sendo carregada.

O propósito dessa solução é melhorar a experiência de uso dos usuários conectados a conexões lentas ou instáveis. Curiosamente, esta solução não está implementada no Facebook Lite, pois o mesmo lida com imagens de forma diferente, até mesmo por meio de outros protocolos de rede, mas sim no aplicativo regular do Facebook. Desse modo, é possível aprimorar a experiência de uso do aplicativo móvel sem alterar drasticamente a quantidade de dados ou o número de requisições de rede necessários para a exibição de uma imagem.

Essa solução pode ser caracterizada com uma dificuldade média, pois requer mudanças no cliente e servidor, além de alterações no container de imagens JPEG, para que as imagens ocupem menos de 200 bytes. Além disso, também requer transformação que aplique o efeito de desfoque na imagem no cliente.

Essa solução tem se tornado tendência entre grandes empresas de internet. Sites como Globo.com, Medium e aplicativos como Instagram e WhatsApp já mostram uma prévia das imagens antes que as originais sejam carregadas, embora algumas delas usem técnicas diferentes para atingir o mesmo objetivo.

### 3.7 CONSIDERAÇÕES FINAIS

Neste capítulo, foram analisadas as soluções implementadas pelos aplicativos de referência que foram projetados com foco em mercados emergentes.

No próximo capítulo são mostradas as conclusões desse trabalho, junto com suas contribuições, limitações e propostas para trabalhos futuros.

## 4 CONCLUSÃO

O desenvolvimento da presente pesquisa permitiu compreender o cenário da internet móvel em países emergentes e quais os principais problemas enfrentados por seus usuários nessas regiões.

Por meio da análise de soluções já implementadas por terceiros, foi possível identificar quais otimizações impactam positivamente aplicativos voltados a à mercados emergentes. As soluções encontradas neles, portanto, apresentaram diferentes graus de dificuldade para implementação, com diversos fatores influenciando essa avaliação.

O desenvolvimento de um algoritmo também permitiu capturar dados relevantes para a análise quantitativa da eficácia de uma das soluções, por meio do impacto no carregamento de imagens e também do número de aparelhos que seriam alcançados por ela.

### 4.1 CONTRIBUIÇÕES

Acredita-se que a presente pesquisa contribuiu para uma melhor compreensão do cenário da internet móvel em países emergentes, e quais as diferenças e limitações que eles possuem em contraste com países desenvolvidos. Além disso, as soluções aqui analisadas devem servir como guia para futuros trabalhos, uma vez que não existe uma literatura vasta para essa área.

### 4.2 LIMITAÇÕES

Para a análise das soluções para mercados emergentes, foram utilizados artigos e livros publicados pelas empresas, que ausentavam dados estatísticos para melhor compreender a eficácia de algumas de suas soluções. Dados como espaço médio de armazenamento em aparelhos móveis ou versões de Android específicos por regiões ajudariam a compreender melhor a importância de certas soluções.

Além disso, não foram aplicados questionários para analisar a eficácia, de forma qualitativa, para algumas soluções apresentadas, como a exibição prévia de imagens. A tendência de uso da solução tornou-se, portanto, uma das justificativas de adoção da mesma.

### 4.3 TRABALHOS FUTUROS

A presente pesquisa manteve seu foco em características relacionadas à otimização de carregamento de dados em dispositivos móveis. Entretanto, com o propósito de otimizar o uso de aplicativos em mercados emergentes de diversas maneiras, outras soluções e características também devem ser analisadas.

Fatores como alfabetização (SOLMS e KRITZINGER, 2012), acesso a e-mail, localização e experiência de usuário podem afetar taxas de uso e engajamento de aplicativos de forma significativa.

O Twitter Lite, uma solução semelhante às apresentadas nesse trabalho (GALLAGHER, 2017), foi divulgada próximo à conclusão do mesmo, portanto, não pôde ser analisada junto com as outras.

Sugere-se, ainda, uma análise mais detalhada sobre critérios de custo e eficácia de algumas das soluções apresentadas no decorrer desse trabalho.

## 5 BIBLIOGRAFIA

AGABABOV, V. et al. **Flywheel**: Google's Data Compression Proxy for the Mobile Web. In: 12th USENIX Symposium on Networked Systems Design and Implementation (NSDI 2015). Oakland: USENIX Association. 2015.

AMP. Accelerated Mobile Pages Project. **Accelerated Mobile Pages Project**, 2016 fev. 2016. Disponivel em: <<https://www.ampproject.org/>>. Acesso em: 14 jan. 2017.

BESBRIS, D. Introducing the Accelerated Mobile Pages Project, for a faster, open mobile web. **Official Google Blog**, 7 out. 2015. Disponivel em: <<https://googleblog.blogspot.com.br/2015/10/introducing-accelerated-mobile-pages.html>>. Acesso em: 14 jan. 2017.

BURT, D. et al. **Cyberspace 2025: Today's Decisions, Tomorrow's Terrain**. Microsoft. [S.l.], p. 47. 2014.

CABRAL, B. K.; KANDROT, E. Facebook Code. **The technology behind preview photos**, 6 ago. 2015. Disponivel em: <<https://code.facebook.com/posts/991252547593574/the-technology-behind-preview-photos/>>. Acesso em: 2 fev. 2017.

CERWALL, P. et al. **Ericsson Mobility Report - November 2015**. Ericsson. [S.l.], p. 32. 2015. (EAB-15:037849).

CLOUDINARY. Cloudinary. **Cloudinary Features**, 10 jan. 2017. Disponivel em: <<http://cloudinary.com/features>>. Acesso em: 10 jan. 2017.

FACEBOOK INC. How we built Facebook Lite for every Android phone and network. **Facebook Code**, 9 mar. 2016. Disponivel em: <<https://code.facebook.com/posts/1365439333482197/how-we-built-facebook-lite-for-every-android-phone-and-network/>>. Acesso em: 18 jan. 2017.

GALLAGHER, N. How we built Twitter Lite. **Twitter Blogs**, 6 abr. 2017. Disponivel em: <<https://blog.twitter.com/2017/how-we-built-twitter-lite>>. Acesso em: 12 abr. 2017.

GOOGLE, INC. **The Building for Billions Playbook (for developers)**: A Companion Guide to The Secrets to App Success on Google Play. [S.l.]: Google Play Books, 2016a. 99 p.

GOOGLE, INC. **The Going Global Playbook (for developers)**: A Companion to The Secrets to App Success on Google Play. [S.l.]: Google Play Books, 2016b. 128 p.

GOOGLE, INC. YouTube Go: YouTube reimagined for the next generation of YouTube viewers. **YouTube Official Blog**, 27 set. 2016c. Disponivel em: <<https://youtube.googleblog.com/2016/09/youtube-go-youtube-reimagined-for-next.html>>. Acesso em: 12 jan. 2017.



GOOGLE, INC. Android Developers. **Dashboard**, 23 mar. 2017. Disponível em: <<https://developer.android.com/about/dashboards/index.html>>. Acesso em: 23 mar. 2017.

GSMA. **Digital Inclusion Report**. GSMA Intelligence. London, p. 90. 2014.

HASSENZAHN, M. **User Experience (UX): Towards an experiential perspective on product quality**. In: Proceedings of the 20th Conference on l'Interaction Homme-Machine. Metz: ACM. 2008. p. 215.

INTERNATIONAL TELECOMMUNICATION UNION. **Measuring the Information Society Report**. ITU. Geneva, p. 234. 2015. (978-92-61-15791-3).

INTERNATIONAL TELECOMMUNICATION UNION. **ICT Facts and Figures of 2016**. International Telecommunication Union. Geneva, p. 8. 2016a.

INTERNATIONAL TELECOMMUNICATION UNION. 2005-2016 ICT data for the world, 2016b. Disponível em: <[https://www.itu.int/en/ITU-D/Statistics/Documents/statistics/2016/Stat\\_page\\_all\\_charts\\_2016.xls](https://www.itu.int/en/ITU-D/Statistics/Documents/statistics/2016/Stat_page_all_charts_2016.xls)>. Acesso em: 07 jan. 2017.

JIN, X. et al. **NChecker: Saving Mobile App Developers from Network Disruptions**. In: Proceedings of the Eleventh European Conference on Computer Systems. London: ACM. 2016.

KAASINEN, E. et al. User experience of mobile internet: analysis and recommendations. **International Journal of Mobile Human Computer Interaction**, v. 1, n. 4, p. 4-23, 2009.

KOHAVI, R.; HENNE, R. ; SOMMERFIELD, D. **Practical Guide to Controlled Experiments on the Web: Listen to Your Customers not to the HiPPO**. In: Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining. San Jose: ACM. 2007.

LINDEN, G. Make Data Useful, 2006. Disponível em: <<http://www.gduchamp.com/media/StanfordDataMining.2006-11-28.pdf>>. Acesso em: 06 fev. 2017.

MICROSOFT CORPORATION. Microsoft Future Decoded Stories. **Introducing Skype Lite, our fastest and smallest app to date, designed for India**, fev. 2017. Disponível em: <<https://www.microsoft.com/en-in/stories/futuredecoded/skypeLite.aspx>>. Acesso em: 05 mar. 2017.

MOZILLA DEVELOPER NETWORK. Promise. **Mozilla Developer Network**, 2 abr. 2017. Disponível em: <[https://developer.mozilla.org/pt-BR/docs/Web/JavaScript/Reference/Global\\_Objects/Promise](https://developer.mozilla.org/pt-BR/docs/Web/JavaScript/Reference/Global_Objects/Promise)>. Acesso em: 10 fev. 2017.

NAGAR, A.; MOHAPATRA, D. P.; CHAKI, N. **Proceedings of 3rd International Conference on Advanced Computing, Networking and Informatics**. In: ICACNI 2015. [S.l.]: Springer. 2015. p. 653.

SILVA, E. L. D.; MENEZES, E. M. Metodologia da pesquisa e elaboração de dissertação, Florianópolis, 2005. 138.

SOLMS, B. V.; KRITZINGER, E. **Critical Information Infrastructure Protection (CIIP) and Cyber Security in Africa – Has the CIIP and Cyber Security Rubicon Been Crossed?** In: Third International ICST Conference. Zanzibar: Springer. 2012. p. 261.

UIT. 2005-2016 ICT data for the world, 2016. Disponível em:

<[https://www.itu.int/en/ITU-](https://www.itu.int/en/ITU-D/Statistics/Documents/statistics/2016/Stat_page_all_charts_2016.xls)

[D/Statistics/Documents/statistics/2016/Stat\\_page\\_all\\_charts\\_2016.xls](https://www.itu.int/en/ITU-D/Statistics/Documents/statistics/2016/Stat_page_all_charts_2016.xls)>. Acesso em: 07 jan. 2017.

UNITED NATIONS STATISTICS DIVISION. Standard country or area codes for statistical use (M49). **UNSD**, 2017. Disponível em:

<<https://unstats.un.org/unsd/methodology/m49/>>. Acesso em: 14. jan. 2017.