

## SISTEMA PARA TRANSMISSÃO DE VÍDEOS EM SERVIDORES XBMC

Sávio Rennan Menêzes Melo (1); Herlan Assis Pereira da Silva (2); Pedro Avelino Ferreira Nogueira(3); Lucas Hiago de Azevedo Dantas (4)

(Instituto Federal do Rio Grande do Norte - Campus Pau dos Ferros, saviorennan@gmail.com)

### INTRODUÇÃO

Observa-se atualmente um grande desenvolvimento na área das tecnologias móveis, começando estas a assumir um papel preponderante na forma como as pessoas se relacionam, não só com essas tecnologias, mas também com os dispositivos eletrônicos em geral. Notou-se um grande crescimento no desenvolvimento tecnológico para comunicação celular móvel, comunicação via satélite e redes locais sem fio. A popularização dessas tecnologias tem permitido o acesso à informações remotas onde quer que se esteja, abrindo um leque muito grande de facilidades, aplicações e serviços para os usuários. A quantidade de celulares conectados (3.000.000.000) atinge mais que o dobro de computadores com acesso à internet (PEREIRA & SILVA, 2009). Diante dessa ascensão, é notório uma maior adesão da população, em detrimento dos tradicionais computadores pessoais, o que implica também em uma mudança na forma como se utiliza o software. Agora, as aplicações para o público geral também devem ser pensadas para os smartphones ou tablets, etc., ou pelo menos devem ter uma versão que apresente compatibilidade com estes dispositivos.

Por meio de aplicativos móveis, os *Smartphones* podem acessar à internet, capturar fotos, capturar e reproduzir áudio e vídeo, etc (PATHAK; HU; ZHANG, 2012). Na era dos chamados “computadores de mão”, a busca por conteúdos de mídia é cada vez mais frequente, muitos usuários buscam praticidade para assistir sua série, ver seu clipe favorito, *podcast*, dentre outros, e é a partir desta busca que surge a necessidade da aquisição de dispositivos multimídia.

Na literatura encontram-se algumas soluções que permitem o *streaming* de conteúdo multimídia entre dispositivos móveis, através de redes domésticas. Uma dessas tecnologias é o *XBMC Media Center*, ou *Kodi* que é uma central multimídia de código aberto e multiplataforma completa, o XBMC consegue reproduzir a maioria dos formatos de arquivos de áudio e vídeo do mercado, conteúdos de CDs e DVDs, do disco rígido e arquivos compartilhados na rede, vídeos por streaming de YouTube, Vimeo dentre outros, tudo isso oferecido dentro de sua própria interface (KODI, 2017). Uma outra solução proprietária é o *AirServer* desenvolvido pela *App Dynamic* (DYNAMIC, 2017). Ele constitui-se como um receptor *Airplay* (protocolo proprietário desenvolvido pela Apple) e o aplicativo permite que o usuário transmita fluxos de mídia a partir de seus dispositivos iOS para computadores Mac rodando *Moutain Lion* ou *Mavericks*. Destacam-se, ainda, os servidores de mídia que utilizam o protocolo DLNA (*Digital Living Network Alliance*), o *PS3 Media Server*, por exemplo, é um servidor de mídia UPnP compatível com DLNA (MACHADO et al, 2014). Escrito em linguagem Java, suporta os principais sistemas operacionais,

com versões para Windows, Linux e MacOS X. Entretanto, a instalação e configuração destas soluções podem ser um tanto complexa para usuários leigos.

Diante do uso dessas novas tecnologias, a procura por dispositivos de transmissão de vídeos em telas maiores (TV's, projetores, monitores, notebooks) tem se proliferado. Para isso, grandes empresas têm trabalhado na construção de dispositivos multimídia para tal finalidade, um deles muito conhecido é o *Chromecast*, que funciona como um reprodutor de conteúdo multimídia online que se comunica com Smartphones e Tablets via Wi-Fi recebendo comandos dos mesmos e reproduzindo em uma TV ou monitor com suporte à conexão HDMI (MACK, 2013). Entretanto, atrelado à alta qualidade observada no produto está também o seu custo, o que restringe seu acesso. De forma alternativa, as chamadas *SMART TV's* normalmente também contam com recursos nativos para a transmissão de mídia digital, porém, o acesso ao recurso está condicionado à aquisição de um produto novo, solução nem sempre disponível a todos.

Diante de todas as facilidades e possibilidades oriundas da era móvel, tendo em vista o grande uso dessas tecnologias nos dias atuais e visando a integração da computação móvel com dispositivos de hardware, esse trabalho objetiva a proposição de uma solução de baixo custo que possibilite a transmissão de mídia digital entre equipamentos eletrônicos de diferentes plataformas.

## METODOLOGIA

Para o alcance dos objetivos, esse trabalho realizou uma pesquisa exploratória para compreensão dos conceitos relacionados a transmissão de vídeo através de dispositivos portáteis. Em relação aos procedimentos técnicos, a partir de uma pesquisa bibliográfica foram identificados trabalhos relacionados ao *stream* de conteúdos e sobre a utilização do microcomputador *Raspberry*.

A primeira etapa do trabalho foi firmada pelos estudos e revisões na literatura, onde foram feitas investigações e comparações com as tecnologias com o intuito de verificar se essas possibilitavam o *stream* de conteúdos de mídia com qualidade. Esses estudos iniciais também guiaram a escolha das tecnologias que seriam utilizadas na construção de um aplicativo móvel.

A etapa seguinte foi o desenvolvimento de um aplicativo móvel voltado para os sistemas operacionais Android, IOS e Windows Phone utilizando o conjunto de ferramentas de desenvolvimento Ionic, e a plataforma de desenvolvimento *Visual Studio Code*. O sistema de *stream* de mídia é voltado para dispositivos móveis. A integração de aplicativo móvel e *Raspberry PI* oferece transmissão de vídeos diretamente da base de dados do *YouTube* e do *Vimeo*, os vídeos podem ser buscados pelo usuário através do aplicativo, a partir de então uma lista de vídeos é recuperada através das Interfaces de Programação de Aplicativos (API) dos sistemas do *YouTube* e do *Vimeo*. O sistema oferece uma interface que permite o cadastro de dispositivos (notebooks, Smart TV's, Raspberry, etc) que serão utilizados para as transmissões dos vídeos buscados.

O *Ionic Framework* surgiu da necessidade de se ter um conjunto de ferramentas que ajudassem no desenvolvimento de aplicativos móveis usando tecnologias web, com o objetivo de realmente competir com as SDK nativas, este é relativamente novo e está sendo bastante aceito no desenvolvimento móvel atual. O *framework* foi construído do início para utilizar padrões web modernos como HTML5, CSS3 e Javascript, sempre pensando nas versões mais modernas dos navegadores móveis (VIEBRANTZ, 2015). Em suma, o uso do *Ionic Framework* é justificado por

meio da pesquisa bibliográfica com base na revisão da literatura, pois no desenvolvimento de aplicações para dispositivos móveis este permite a criação de aplicações híbridas e de rápido e fácil desenvolvimento.

Definidos os componentes essenciais de software para a implementação do sistema, o passo seguinte foi a escolha dos dispositivos de hardware, inicialmente optando pelo *Raspberry* como tecnologia.

O *Raspberry Pi* foi desenvolvido no Reino Unido pelo ex-professor da Universidade de Cambridge. A ideia inicial era fabricar um computador de baixo custo para ser utilizado na educação e estimular os jovens ao estudo da programação. O sistema operacional do microcomputador funciona basicamente com o sistema operacional gratuito e de código livre LINUX, embora existam também versões do Windows compatíveis (DOS SANTOS, 2016). A infraestrutura de hardware do equipamento utilizado é a seguinte: *Raspberry Pi* modelo B+, com as seguintes especificações: dimensões de 85mm x 56mm, processador Broadcom SoC 700MHz, 512MB de memória RAM, conector de energia microUSB, saída HDMI, saída de áudio A/V, porta Ethernet, 4 portas USB, socket para cartão microSD e 40 pinos de entrada/saída. A pesquisa exploratória foi utilizada aqui com o objetivo de entender os principais conceitos e história do equipamento, que foi viável na pesquisa por utilizar conceitos de software livre e por fornecer a infraestrutura requerida para a solução do problema.

## RESULTADOS E DISCUSSÕES

Com base na pesquisa bibliográfica feita e estudando os protocolos utilizados para tal fim, considerou-se que a implementação de um sistema de *stream* de vídeo para ser executado na plataforma computacional *Raspberry Pi* sanaria o problema do custo dos dispositivos multimídia existentes e o uso de um aplicativo desenvolvido com o *Ionic Framework*, aliado a um serviço XBMC (devido ao seu destaque em relação as demais soluções levantadas), possibilitaria aos usuários uma ferramenta que funcionasse em vários sistemas operacionais, utilizando um sistema de multimídia completo. Por fim, os resultados são de um sistema que transmite conteúdos multimídia, utilizando recursos de baixo custo, de fácil instalação e utilização oferecendo usabilidade e agilidade no *stream* de mídia a partir de um dispositivo móvel.

O desenvolvimento do sistema intitulado *YoutBerry* foi implantado e testado no campus Pau dos Ferros do IFRN e apresentado como prática profissional do curso de Análise e Desenvolvimento de Sistemas. Algumas imagens da versão inicial do sistema móvel podem ser observadas na Figura 1.

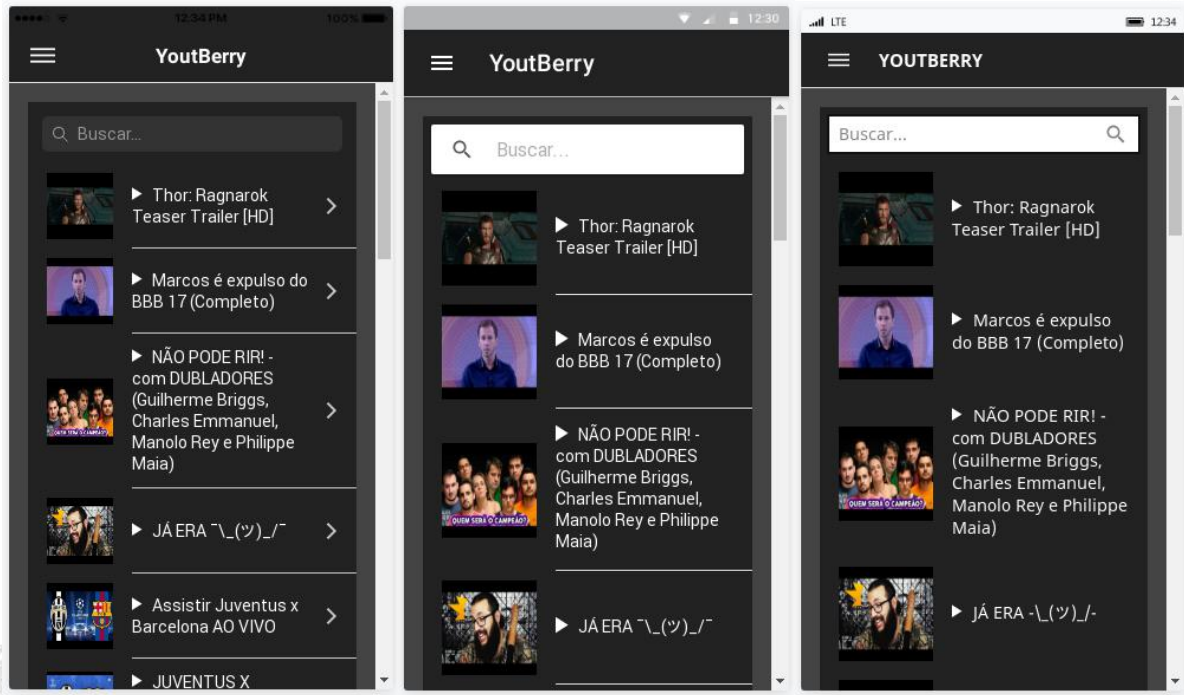


Figura 1 - Tela Inicial, Android, IOS, Windows Phone, respectivamente.

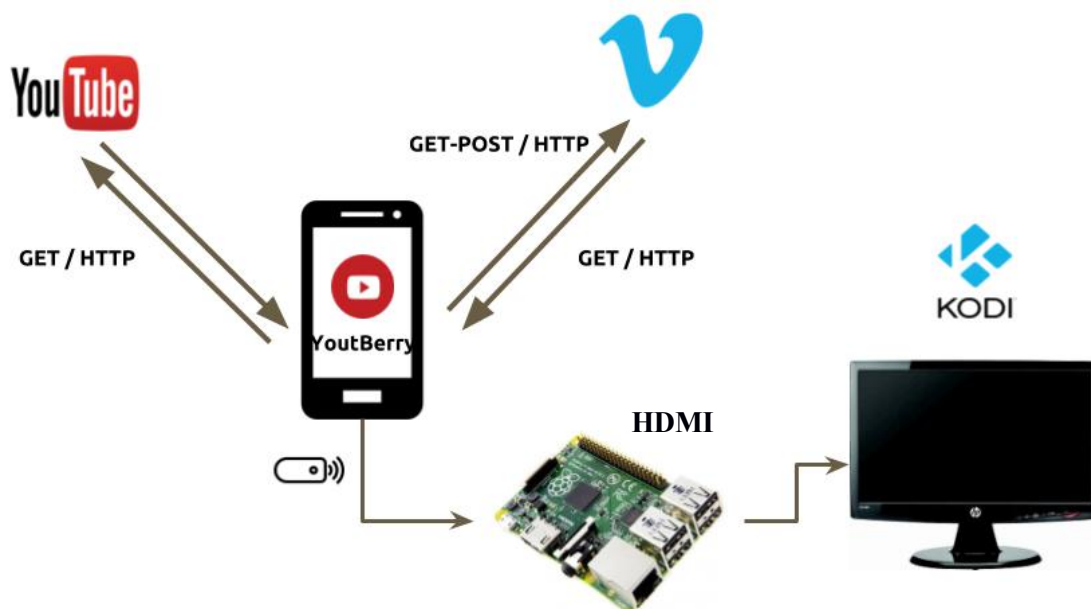


Figura 2 - Funcionamento Geral do Sistema

A Figura 2 mostra o esquema geral de funcionamento do Youtberry, onde a exibição de conteúdo está condicionada a qualquer tela que possua entrada HDMI, no caso do uso do aplicativo com o Raspberry. Vale ressaltar que o aplicativo ilustrado acima é compatível com os três sistemas

operacionais móveis mais usados, portanto a comunicação entre os serviços de vídeo é feita de maneira totalmente multiplataforma.

## CONCLUSÕES

Os resultados iniciais foram satisfatórios, pois estes cumprem com os requisitos dos objetivos propostos pelo trabalho. A ideia de trabalhos futuros é de um desenvolvimento de um software baseado na ideia de centrais multimídia (semelhante ao Kodi) para transmissão dos conteúdos de vídeo, voltado exclusivamente para o aplicativo *YoutBerry*, eliminando a dependência de servidores de terceiros para a correta exibição do conteúdo.

## REFERÊNCIAS BIBLIOGRÁFICAS

DOS SANTOS, Alexandre Henrique. Aprendizagem Musical Na Era Digital: Uma Proposta De Acesso De Baixo Custo a Partir Do Raspberry Pi e Sonic Pi. NICS Reports, v. 1, n. 17, 2016.

DYNAMIC. Dynamic Airserver 5.0. Disponível em: <<https://www.airserver.com/>>. Acesso em: 07 Set. 2017.

KODI. About Kodi. Kodi. 2017. Disponível em: <<https://kodi.tv/about/>>. Acesso em: 07 Set. 2017.

MACHADO, Felipe Alexandre Oliveira; PINTO, Adriano Viana; TEIXEIRA, Mário Meireles. Uma Rede de Compartilhamento de Conteúdo Multimídia em Dispositivos Móveis Baseados na Plataforma Android. 2014.

MACK, E. Going, going... Chromecast sold out online, but not everywhere. 2013. Disponível em: <<https://www.cnet.com/news/going-going-chromecast-sold-out-online-but-not-everywhere/>>. Acessado em: 19 ago 2017.

PATHAK, A.; HU, Y. C.; ZANG, M. Where is the energy spent inside my app?: Fine grained energy accounting on smartphones with eprof. In Proceedings of the 7th ACM European Conference on Computer Systems, EuroSys '12, pages 29-42, New York, NY, USA. ACM.

PEREIRA, L. C. O.; SILVA, M. L. Android para desenvolvedores. Brasport, Rio de Janeiro, 2009.

VIEBRANTZ, Alvaro Fellipe Petry Mendes; DA SILVA CAMPOS, Guilherme Falcão. Construindo Aplicativos Híbridos com Ionic Framework. Tendências e Técnicas em Sistemas Computacionais, p. 60.