



Universidade Federal do ABC

Universidade Federal do ABC - UFABC

Trabalho de Graduação III

Desenvolvimento de Software de Interface para Dispositivos Interativos de TVs Inteligentes

Professor Orientador:

Prof. Dr. Celso Setsuo Kurashima

Professor Coorientador:

Prof. Dr. Guiou Kobayashi

Aluno:

Luis Felipe Pellegrini Meloni

SANTO ANDRÉ

Dezembro de 2015

UNIVERSIDADE FEDERAL DO ABC

LUIS FELIPE PELLEGRINI MELONI

**Desenvolvimento de Software de Interface para Dispositivos
Interativos de TVs Inteligentes**

Trabalho de Graduação III da
Universidade Federal do ABC – UFABC

Orientador:

Prof. Dr. Celso Setsuo Kurashima

Coorientador:

Prof. Dr. Guiou Kobayashi

SANTO ANDRÉ

Dezembro de 2015

Glossário

Sigla:	Significado:
<i>Android</i>	Sistema operacional para dispositivos móveis
<i>AppleTV</i>	Reprodutor de conteúdo digital da empresa Apple
<i>broadband</i>	Banda larga
<i>broadcast</i>	Rádiodifusão
<i>Carrossel de dados</i>	Maneira de descrever dados que ficam disponíveis em tempos intercalados para transmissão unidirecional sendo que o receptor aguarda até que o dado desejado seja transmitido
<i>chats</i>	Comunicação através da Internet para transmissão em tempo real de mensagens de texto.
<i>ChromeCast</i>	Reprodutor de conteúdo digital da empresa Google
<i>codecs</i>	Codificador/Decodificador
<i>DTV</i>	Selo de reconhecimento de interatividade para televisores e conversores.
<i>Facebook</i>	Serviço de rede social
<i>Ginga</i>	<i>Middleware que conduz a interatividade da TV Digital brasileira</i>
<i>Global ITV</i>	Projeto de parceria estratégica para desenvolvimento de plataforma inter-operável dos sistemas brasileiro e europeu de TV Digital
<i>Hashtag</i>	Espécie de rotulo utilizado em redes sociais e <i>microblogging</i> que torna a busca por temas mais simples
<i>Java</i>	Linguagem de programação
<i>Media centers</i>	Dispositivo dedicado para execução de diversas formas de mídia
<i>middleware</i>	Camada de <i>software</i> entre a infraestrutura de execução do <i>hardware</i> e o <i>software</i> de aplicação
<i>MPEG2-TS</i>	Formato de mídia para <i>transport stream</i> de vídeo.
<i>Netflix</i>	Site de <i>streaming</i> de filmes por demanda
<i>Nickname</i>	Nome ou apelido utilizado em redes sociais
<i>Parse/parsing</i>	Processo de analisar e converter informações conforme regras semânticas e estruturais de diferentes linguagens.
<i>Playout</i>	Dispositivo que entrega o conteúdo modulado de rádiodifusão ao seu respectivo canal
<i>set-top box</i>	Equipamento conversor que conectado à uma fonte externa de sinal, transforma o conteúdo no formato a ser apresentado em um televisor
<i>SmartTV</i>	TV conectada/integrada à Internet
<i>Socket</i>	Componente de software para conexão com uma rede permitindo acesso a dados entre a rede
<i>Splashscreen</i>	Janela gráfica contendo uma imagem ou logo exibida no início de uma aplicação (<i>software</i>).
<i>streaming</i>	Serviço de entrega de conteúdos multimídia ao usuário final através de um provedor
<i>tablet</i>	Computador móvel operado através do toque

<i>timestamp</i>	Sequencia de caracteres que identificam o exato momento de ocorrência de um determinado evento
<i>Twitter</i>	Serviço de rede social
<i>Videogames</i>	Consoles de entretenimento e execução de jogos digitais
<i>Webservice</i>	Componentes para envio e recebimento de dados em formato XML utilizado por aplicações
<i>YouTube</i>	Site de compartilhamento de vídeo via Internet

Lista de Siglas

<i>AIT (tabela)</i>	<i>Application Information Table</i>
<i>CNPq</i>	Conselho Nacional de Desenvolvimento Científico e Tecnológico
<i>DSM-CC</i>	<i>Digital Storage Media Command and Control</i> – kit de ferramentas para desenvolvimento do canal de controle do <i>stream</i> de MPEG
<i>DVB-T</i>	<i>Digital Video Broadcasting – Terrestrial</i> . Padrão europeu de transmissão terrestre de vídeo digital.
<i>FTP</i>	<i>File Transfer Protocol</i> – protocolo para Internet utilizado para transferência de arquivos.
<i>HbbTV</i>	<i>Hybrid Broadcast Broadband TV</i> – padrão europeu de <i>middleware</i> para TV interativa
ISDB-Tb	<i>Integrated Services Digital Broadcasting Terrestrial</i> – Padrão Brasil
<i>TCP</i>	<i>Transmission Control Protocol</i> – protocolo núcleo para padrão Internet (<i>Internet Protocol</i>) onde transmissão é feita de forma confiável.
<i>XML</i>	<i>Extensible Markup Language</i> – formato de organização de informações digitais que pode ser compreendido por homem e máquina.

***Abstract** – Brazil’s analog television system is expected to start its shutdown in April 2016 and should be replaced in its entirety by the digital system by 2018. This change from analog to digital brings benefits in received media quality and possible interactive applications by use of Ginga middleware. The objective of this work is to develop an interactive digital TV application using the concept of second screen device to send messages that are displayed on the television screen running Ginga application, demonstrating the interactive capabilities of the Brazilian digital TV system.*

***Keywords:** Digital TV, Ginga, interactivity, second screen, interactive television.*

Resumo - O desligamento do sistema analógico de televisão no Brasil está previsto para iniciar em abril de 2016 com previsão de estar substituído em sua totalidade pelo sistema digital até 2018. Esta mudança de sistema analógico para digital traz benefícios na qualidade da mídia recebida e a possibilidade de interação com a TV por uso do *Middleware* Ginga. O objetivo deste trabalho é elaborar uma aplicação interativa para TV digital com uso de dispositivo segunda tela para envio de mensagens que são exibidas na tela na televisão por meio de uma aplicação Ginga, demonstrando a capacidade interativa do sistema brasileiro de TV digital.

Palavras-Chave: TV Digital, Ginga, interatividade, segunda tela, televisão interativa.

Sumário

Introdução	1
Desenvolvimento	3
1. Revisão de Literatura	3
2. Metodologia	8
Descrição:	9
Metas Trimestrais:	9
• Primeiro Trimestre	9
• Segundo Trimestre	9
• Terceiro Trimestre	9
Cronograma:	10
2.1. Descrição da Exibição da Mensagem na Tela da TV	12
2.2. Descrição Detalhada da Aplicação Segunda Tela – Android	15
2.3. Descrição do Servidor e Aplicação PHP	16
3. Resultados	17
3.1 Sobre as Ferramentas e Linguagens Utilizadas no Desenvolvimento	17
3.1.1 Desenvolvimento da Aplicação Ginga:	18
3.1.2 Desenvolvimento do Webservice:	19
3.1.3 Desenvolvimento do Aplicativo Android:	20
3.2 Programação	22
4. Simulação e Teste:	24
4.1 Teste do Aplicativo Android:	24
4.2 Teste do Servidor:	28
4.3 Teste da Aplicação Ginga:	29
Conclusão	31
Referências Bibliográficas:	32

Introdução

Desde o final de 2007 o Brasil busca implantar o Sistema de Televisão Digital em toda extensão de seu território, aos poucos toda a infraestrutura vêm sendo digitalizada para a cobertura dos mais de 5.500 municípios brasileiros, porém devido a atrasos no prazo, vêm sendo adiado para o desligamento do sistema analógico. O sistema brasileiro de TV digital denominado de ISDB-Tb, acrônimo de “*Integrated Services Digital Broadcasting Terrestrial – Padrão Brasileiro*” é uma evolução do padrão pré-existente ISDB-T japonês acrescido de algumas tecnologias como *codecs* de áudio e de vídeo modernos que apresentam melhor eficiência [1]. A TV aberta no Brasil é um meio de comunicação com grande impacto social sendo que a grande maioria dos lares brasileiros possuem pelo menos um aparelho televisivo, conforme dados disponibilizados em pesquisas públicas. O grande desafio do sistema digital de televisão é substituir o sistema analógico anterior sem apresentar ônus à população brasileira. Os benefícios da digitalização são diversos, entre eles melhor qualidade de som e vídeo, melhor cobertura do sinal além de possibilitar múltiplas programações em um único canal permitindo por exemplo a transmissão de uma programação para dispositivos móveis e uma programação em alta definição para aparelhos residenciais. Junto a estas características, o sistema foi concebido para incorporar o padrão interativo denominado de Ginga. Esta padronização é feita através do *middleware* contendo as modalidades declarativas e procedurais. Vem sendo notada a tendência global em se adotar padrões híbridos entre a radiodifusão (*broadcast*) e sistemas de banda-larga (*broadband*) representando uma evolução da interoperabilidade que permitem atingir novas aplicações interativas com dispositivos segunda tela e aplicações semelhantes às encontradas na Internet. Dentro do grupo de pessoas com acesso à Internet, 77% declararam utilizar ou realizar alguma atividade nas redes sociais [2], o que representa um uso intensivo das redes sociais por usuários de Internet e indica uma mudança na forma com que as pessoas interagem e se comunicam umas com as outras e com organizações. O estudo também apresentou uma tendência no uso do telefone celular como principal forma de acesso às redes sociais sendo o meio de acesso para 53% desse público.

No contexto de ampliar a interatividade do usuário que abandona seu papel de mero telespectador para ganhar novas maneiras de interagir com seu aparelho de TV e a programação televisiva, este projeto consiste de um estudo e prova de conceito de uma

das possíveis relações entre o sistema de TV digital e as redes sociais, tendo em vista a importância dessas redes na Internet. O objetivo deste projeto é trazer para o universo da TV digital, com auxílio do sistema de interatividade Ginga, a possibilidade de se enviar mensagens através de um dispositivo segunda tela para que seja exibida em conjunto com a programação televisiva na TV. Para atingir tal objetivo, será desenvolvido programas para diferentes dispositivos (dispositivo móvel e TV digital) com a capacidade de se comunicar através da Internet com uso de um servidor com programa também desenvolvido para este fim.

Desenvolvimento

1. Revisão de Literatura

Recentemente no Brasil os televisores analógicos vêm sendo substituídos por televisores digitais modernos, que por lei incorporam as funcionalidades para operarem de acordo com o ISDB-Tb, compatíveis com a interpretação do Ginga (*middleware* de interatividade do ISDB-Tb) de cada fabricante. O Ginga, por ser definido dentro da norma ABNT NBR 15606-4 [3], possui alguns pontos que permitem diferentes interpretações e implementações capazes de atender ao mesmo quesito, portanto, diferentes fabricantes de TV Digital podem implementar o Ginga de maneiras ligeiramente distintas.

Uma outra solução para compatibilizar os televisores analógicos para o sinal digital, sem a necessidade de substituição do aparelho televisor, é de operar juntamente com um *set top box* que recebe o sinal digital e converte para que seja apresentado em televisores antigos com apenas entradas analógicas. Isto ocorre similarmente no restante do mundo respeitando a compatibilidade de padrão para cada localidade. Com a migração para sistemas digitais, vêm sendo ampliado o uso de sistemas *DTV_i* (televisão digital interativa), desenvolvidos para oferecer aos telespectadores serviços multimídia interativos inovadores. O conjunto de sistemas televisivos, tanto analógico quanto digital, estão presentes em 96,9% dos lares brasileiros de acordo com censo de 2011 [4], porém apenas quem tem acesso ao sistema digital pode se beneficiar da interatividade. Um ponto de destaque é a influência da Internet nos aparelhos de *Smart TVs* que possibilitam a conectividade tanto no uso de aplicativos semelhantes aos *Mobile* para uso residencial quanto a possibilidade de convergência *broadcast-broadband* (radiodifusão e banda larga). A maioria dos aparelhos de TV vendidos atualmente são “inteligentes” (*Smart TVs* também conhecidas como *Connected TVs*) e estão compatíveis com a tendência atual de integrar os aparelhos televisivos com a Internet.

Nota-se esta tendência também nos aparelhos residenciais que são conectados a TV's como por exemplo os *media centers* e a nova geração de *Videogames* que suportam conexão com a Internet tanto para uso em jogos quanto na execução de programação de vídeo por *streaming*. Exemplos de serviço de *streaming* de vídeo – ou vídeo sob demanda - são o *YouTube* e o *Netflix*.

Outros aparelhos eletrônicos que seguem a mesma tendência de obter vídeos através da Internet são o *AppleTV* [5] e o *ChromeCast* [6] que além de estarem conectados à Internet, incentivam o uso do dispositivo de segunda tela que pode ser um *tablet* ou um aparelho celular como forma de interagir com a tela e o conteúdo exibido.

Todos os aparelhos citados fazem uso de aplicativos disponibilizados pelos próprios fornecedores de conteúdo o que indica a tendência de se manter uma plataforma simples para programação e desenvolvimento de conteúdo e com amplo uso da Internet para transmissão de dados e vídeo.

Ao contrário do que vem sendo notado em sistemas proprietários como os citados anteriormente da Google e Apple, que já se encontram numa situação de relativo sucesso no mercado, cada sistema de *DTV* e *SmartTV* vem sendo criado com utilização de tecnologias de cada fabricante, que por serem proprietárias distinguem entre si, e que necessitam atender necessidades e legislações de diferentes nacionalidades. Aplicações desenvolvidas para um sistema e em um país de um modo geral enfrentam muitas dificuldades no momento de ser portadas para um outro sistema. Desta forma, conteúdos interativos encontram dificuldades desde o desenvolvimento até a distribuição o que limitam sua evolução pela dificuldade de realizar uma adequação entre sistemas.

Tendo em vista as dificuldades expostas, verifica-se a necessidade de estabelecer um padrão global sobre a televisão digital. O escopo geral do projeto Global ITV, uma parceria para cooperação internacional firmada entre países da União Europeia e o Brasil, consórcio formado em atendimento à chamada pública conjunta MCTI/CNPq no. 13/2012 e Comunidade Europeia FP7-ICT-2013-EU-Brazil, consiste em apresentar os caminhos para se alcançar a próxima geração de plataformas de TVs Digitais híbridas entre os sistemas brasileiro e europeu definindo padrões e soluções para o futuro. Os centros tecnológicos de pesquisa responsáveis são o CNPq no Brasil e o *Seventh Framework Programme* na Europa. Ao todo são 18 organizações envolvidas, 9 na Europa, dentre elas institutos de renome como o *Fraunhofer* e *IRT (Institut für Rundfunktechnik)* e 9 no Brasil onde 6 destas são universidades, a UFABC, USP, UNICAMP, UNESP, UFPA e a Universidade Católica de Brasília.

O objetivo do projeto GLOBAL ITV como extraído do documento principal do projeto [7]: “*The Project GLOBAL ITV will bring the worlds of interactive television*

together. Findings gained from the exemplar harmonisation approach of the Brazilian ISDB-Tb/Ginga and the European DVB/HbbTV systems will deliver key concepts for interoperability solutions for a future global interactive TV standard worldwide.” – Traduzido como: “O projeto GLOBAL ITV unirá os mundos da televisão interativa. Achados obtidos através da exemplar abordagem harmonizada entre os sistemas brasileiro ISDB-Tb/Ginga e europeu DVB/HbbTV trarão conceitos chave para a solução interoperável de um futuro padrão global de TV digital.”

Uma breve descrição do *background* da interatividade na Europa [8]: A primeira tentativa de executar aplicações para televisores interativos transmitidas em banda larga começou na Europa com o *Multimedia Home Platform* (MHP) com aplicativos executados principalmente em máquinas Java. O sistema não obteve amplo sucesso devido aos royalties associados e linguagens de programação muito complicadas. No Brasil o padrão de TV digital foi concebido com o *middleware* Ginga, que segundo o texto, é um sistema mais avançado e com vantagens técnicas além de não ser proprietário. Com a evolução das TVs digitais e o surgimento das *Smart TV* que podem ser conectadas à Internet houve interesse em criar aplicações, porém, cada fabricante desenvolveu seu próprio sistema proprietário que não permitia a interoperabilidade de aplicações entre as TVs conectadas. Isto motivou a padronização de um vocabulário que pudesse ser compreendido igualmente por todas as TVs independente de seu fabricante. Neste cenário de aplicações utilizando a conexão de banda larga e o sinal das emissoras de TV, surgiu o HbbTV (*Hybrid Broadcast-Broadband Television*).

Os principais focos de estudo no projeto Global ITV que irão possibilitar o entendimento e execução do trabalho proposto são: O HbbTV, padrão aberto europeu de interatividade, entretenimento utilizando a hibridização da radiodifusão e a Internet; O similar brasileiro, o Ginga [9], um *middleware* também aberto que trabalha em cima do padrão nipo-brasileiro de TV digital e se baseia em um conjunto de tecnologias desenvolvidas utilizando inovações e padronizações brasileiras; o padrão de radiodifusão de televisão digital europeu, o DVB-T (*Digital Video Braodcasting – Terrestrial*) e o padrão ISDB-T internacional em que o padrão brasileiro é baseado.

Os *middlewares* utilizados em cada padrão se encontram comparados na Tabela 1 abaixo. Podem ser notadas semelhanças entre eles: ambos utilizam o DSM-CC, sistema de arquivos para aplicativos, que é um carrossel de objetos; as aplicações deste carrossel possuem a mesma maneira de transporte utilizando o MPEG2-TS *broadcast*; a

tabela AIT também é utilizada nos dois sistemas para sinalização de aplicativos com alterações nos padrões devido à localização; a sinalização de controle é realizada da mesma maneira em ambos os sistemas e estes suportam o uso de uma segunda tela. Importante ressaltar que a compatibilidade dos *middlewares* com os sistemas de TV digital ISDB-Tb e DVB-T é possível sem impedimentos devido às semelhanças entre o HbbTV e o Ginga. Isto permite que aplicações desenvolvidas em Ginga sejam portadas para um sistema DVB-T e o mesmo ocorre para o HbbTV executado em sistemas ISDB-Tb se utilizarem *playout* compatível.

Tabela 1: Comparação entre middlewares HbbTV e Ginga

Característica	HbbTV	Ginga
Transporte de Aplicativos	Redes IP e DSM-CC por MPEG2-TS <i>broadcast</i>	DSM-CC por MPEG2-TS <i>broadcast</i>
Sinalização de Aplicativos	Utiliza AIT (<i>Application Information Table</i>) baseado nos padrões ETSI	Utiliza AIT (<i>Application Information Table</i>) baseado nos padrões ABNT
Montagem do Sistema de Arquivos de Aplicativos	Carrossel de objetos DSM-CC	Carrossel de objetos DSM-CC
Execução de Aplicativos	Baseada em CE-HTML	Baseada em NCL/Lua ou Java xlet
Conexão de Banda Larga	Requisito funcional, prioridade para <i>download</i> de aplicativos	Aceita que conexões via <i>broadcast</i> recebam aplicativos e estende a experiência do usuário
Sinalização de Controle de Aplicativos	Mensagens baseadas em GEM (<i>Globally Executable Multimedia Home Platform</i>)	Mensagens baseadas em GEM (<i>Globally Executable Multimedia Home Platform</i>)
Comandos de Edição	Depende do contexto da aplicação CE-HTML	“gingaEditingCommands” para DSM-CC <i>broadcast</i>
Suporte a Segunda Tela	Compartilhamento de sessão da aplicação principal	Estende o conteúdo para uma segunda tela em um mesmo <i>gateway</i>
Compatibilidade DVB-T/ISDB-T	Sim/Possibilidade	Possibilidade/Sim

2. Metodologia

A descrição a seguir detalha a concepção inicial do projeto e as diretrizes tomadas no início para a elaboração bem sucedida de tal. Esta descrição se faz importante para entender as diversas etapas tomadas durante o desenvolvimento e as mudanças realizadas para melhoria, adequação ou para o devido funcionamento do projeto:

A metodologia empregada para atingir o objetivo consiste em um estudo sobre a interatividade em redes sociais, que fornecerá os subsídios necessários para realizar a implementação em software para middleware Ginga. Nesta etapa do projeto procurou-se fazer um levantamento das funções possíveis de serem implementadas em Ginga que atendessem a proposta e a forma como deveriam ser inseridos os textos no servidor. A implementação será desenvolvida em máquina virtual emulando o sistema que será apresentado no aparelho televisivo. Para isto, deverá ser dedicado um tempo para a configuração da máquina virtual em ambiente que permita a conexão e recepção de dados de mensagens pela internet. Em seguida deverá ser programada a aplicação que consiste em blocos do projeto em Ginga como segue:

- Configuração para recepção de mensagens
- Configuração para apresentação das mensagens em tela
- Configurações gerais para exibição ou não do conteúdo

Além desta, deverão estar presentes no projeto as configurações que não são baseadas em Ginga mas que serão necessárias para o projeto:

- Configuração de servidor para recepção das mensagens pelo usuário
- Configuração de serviço baseado em web ou nativo que permita o envio de mensagens ao servidor

Considerados os blocos necessários para a aplicação do projeto, em seguida devem ser estabelecidos a ordem de desenvolvimento destes e as metas para conclusão do projeto no prazo devido.

Descrição:

O desenvolvimento do projeto pode ser descrito pelas seguintes etapas na seguinte ordem:

1. Diretrizes (Guidelines) para o desenvolvimento da aplicação
2. Implementação em fase demo da aplicação
3. Configuração de teste com corpo de prova
4. Conclusões e finalizações

Metas Trimestrais:

- ***Primeiro Trimestre***
 - Elaboração de um Plano de Trabalho.
 - Estudo dos padrões do ISDB-Tb e DVB-T, Ginga e HbbTV.
 - Estudos da relação motivo do projeto entre a interatividade da TV Digital e as Redes Sociais.
 - Especificações dos requisitos necessários para alcançar as metas práticas e o desenvolvimento da prova de conceito.
- ***Segundo Trimestre***
 - Compilação e implementação de software em Ginga (Fase 1) que atenda as especificações definidas como prova de conceito.
 - Configuração de servidor e implementação em Ginga de maneira que receba as mensagens.
 - Simulação em máquina virtual do *set top box* com software desenvolvido.
- ***Terceiro Trimestre***
 - A Fase 2 consiste em correção de erros encontrados na fase 1 e aprimoramento do projeto para versão final do sistema como um todo.
 - Finalização do projeto e execução de relatório final.

A metodologia descrita acima foi concebida no início do projeto. Segue a seguir a metodologia utilizada durante o desenvolvimento do trabalho:

No presente trabalho, a abordagem ao problema se deu elaborando a infraestrutura necessária para funcionamento da aplicação. Para funcionamento da aplicação proposta a televisão deve atender ao padrão brasileiro de TV digital e possuir acesso à Internet. O dispositivo segunda tela também deve ter acesso à Internet para enviar a mensagem a um servidor que será utilizado para armazenar todas mensagens recebidas. A aplicação Ginga executada na televisão fica encarregada de buscar no servidor as mensagens para que seja exibida na tela junto à programação televisiva.

O projeto depois de elaborado consiste em uma aplicação para TV Digital onde é implementado uma “sala” de *chat* onde um usuário que assiste á programação televisiva pode compartilhar comentários e opiniões por meio de uma aplicação de segunda tela (*tablet* ou telefone celular) executando um aplicativo Android para inserção de texto e envio para o servidor. O texto é então obtido pela televisão e exibido em tela de acordo com a chegada das mensagens, conforme ilustrado na Figura 1.

O dispositivo segunda tela envia uma requisição HTTP contendo valores, que podem ser interpretados por código PHP, no final da URL contendo as informações de usuário e mensagem ao servidor (1). O servidor por sua vez interpreta a requisição realizando o *parse* dos valores para estrutura XML (será detalhado adiante – vide página 15) e deixando disponível este conteúdo para acesso no servidor. O servidor foi chamado neste trabalho de Webservice (2) por integrar o sistema de mensagem de forma que diferentes aplicações possam trocar dados, compatibilizando a interação entre diferentes plataformas.

Pela norma ABNT que define o Ginga-NCL apenas o protocolo de comunicação padrão TCP está definido como obrigatório. Outros protocolos como HTTP ou SOAP não constam na norma e para serem utilizados devem ser implementados pelo desenvolvedor de aplicações [10].

Neste trabalho não foi utilizada uma arquitetura já definida de Webservice como por exemplo REST ou SOAP. Todas as mensagens recebidas pelo servidor são armazenadas em um mesmo arquivo .txt estruturado em XML adicionado de estampa de tempo (*timestamp*). O televisor que estiver rodando a aplicação interativa faz uma requisição ao webservice (3) via TCP obtendo o conteúdo do arquivo em formato XML.

A solicitação por parte da TV é feita de maneira discreta em intervalos constantes de tempo (a cada instante de tempo determinado no código) e sempre que encontrar uma nova mensagem, a mesma será exibida na tela.



Figura 1: Esquema Geral da Infraestrutura da Aplicação

A aplicação em Ginga (utilizando linguagem NCLua) ficará encarregada de solicitar as mensagens ao servidor e apresentá-las no televisor com uma área separada da tela da programação televisiva. Neste projeto um usuário utilizando dispositivo segunda tela envia mensagens ao servidor. Conceitualmente mais usuários podem estar enviando mensagem à aplicação ao mesmo tempo, porém este cenário não foi testado neste trabalho. O servidor foi configurado para receber as mensagens através de conexões HTTP e assim que uma mensagem é recebido ela a insere no XML junto com estampa de tempo e por este motivo poderia ir acrescentando novas mensagens de diversos usuários ao mesmo tempo.

2.1. Descrição da Exibição da Mensagem na Tela da TV

Diversas maneiras de apresentar o conteúdo ao usuário foram consideradas de maneira otimizar a visualização da tela pelo usuário com objetivo de tornar a leitura fácil. Dentre as possibilidades, foi considerada a exibição de mensagens uma em cima da outra, como demonstra a Figura 2 abaixo, similar a encontrada em mensagens de texto de telefones celular.

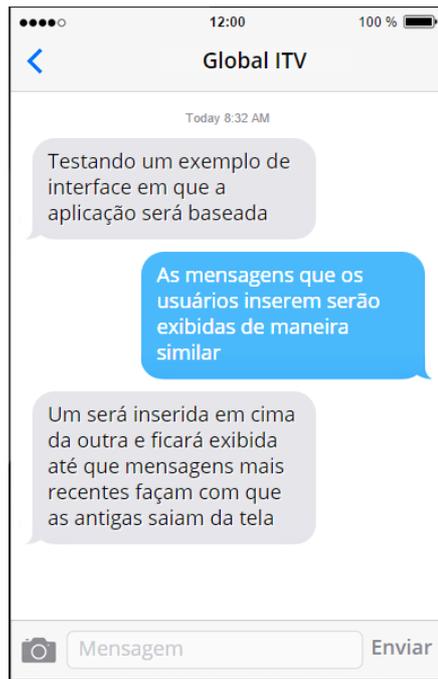


Figura 2: Exemplo que poderia ter servido de base para aplicação interativa em televisores.

A maneira escolhida para exibição das mensagens na tela da TV, representada na Figura 3 é devido sua simplicidade, o que auxilia o usuário a compreender e assimilar o conteúdo sendo exibido na tela de maneira rápida. Uma parte superior da tela é dedicada para exibição de mensagens logo que a aplicação é inicializada e ao realizar a busca por mensagens na Internet e encontrar algum conteúdo, a última mensagem recebida é exibida na tela. Esta maneira pode levar ao caso indesejado em que duas mensagens chegam ao servidor em instantes de tempo muito próximo. Neste caso uma das mensagens não seria lida e por isso foi disponibilizado ao usuário que ele navegue por todas as mensagens já armazenadas utilizando as setas do controle remoto. As mensagens são exibidas em tempo real, levando apenas o tempo das conexões realizadas.

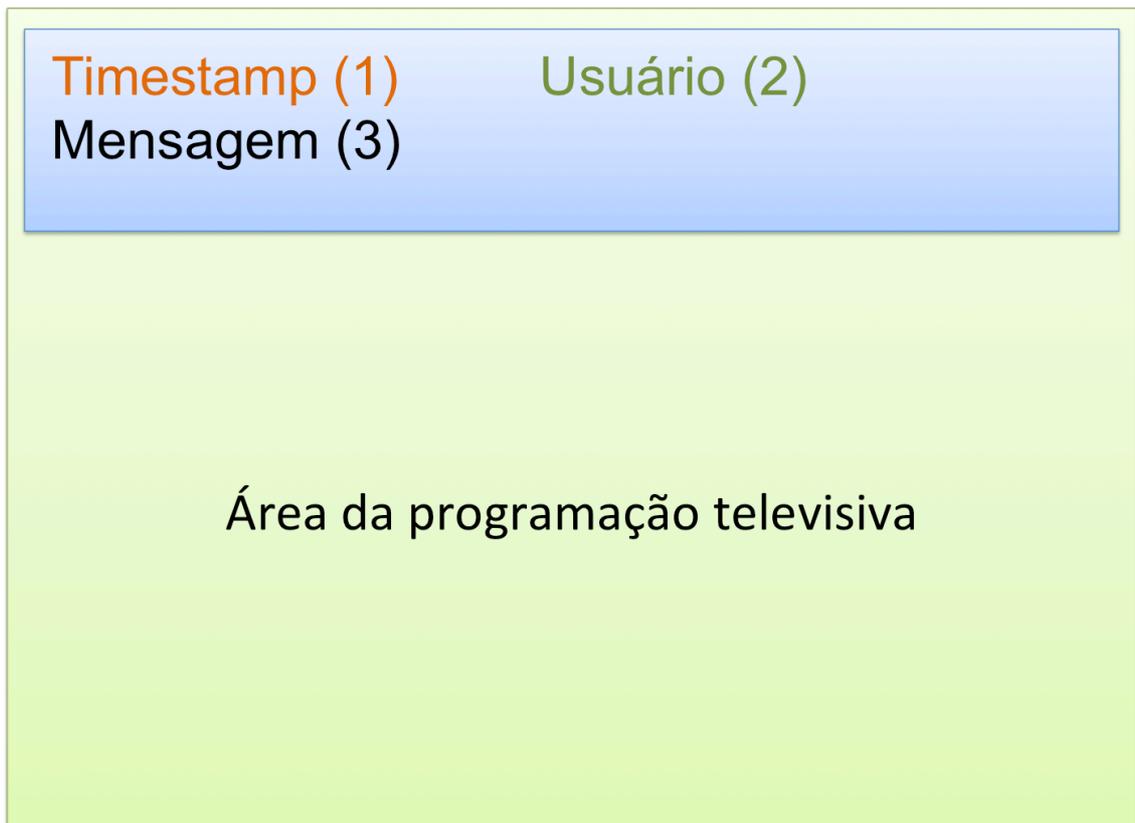


Figura 3: Esquema da exibição de mensagens na televisão escolhido para o projeto.

Na Figura 3 acima nota-se a existência de 3 parâmetros sendo exibidos ao usuário. O *Timestamp* (1) contendo a informação de horário é inserido pela aplicação PHP no servidor assim que uma mensagem é recebida e inicialmente se encontra no formato Unix. O usuário (2) é uma informação de nome ou apelido inserido pelo usuário no início do uso do aplicativo Android. A mensagem (3) é também inserida pelo usuário da aplicação no dispositivo segunda tela Android.

A aplicação executada na televisão deve ser compatível com padrão Ginga para ISDB-Tb. Deve realizar uma conexão TCP ao *webservice* obtendo como retorno o arquivo XML em forma de texto, em seguida deve ser realizado um *parsing* para tornar os valores do XML em valores dentro de vetores. O programa deverá tratar de forma separada dois possíveis casos: se ele estiver iniciando sua execução deve obter o XML e exibir a última mensagem; se já estiver em execução, deve verificar se chegou uma mensagem nova e apenas no caso de haver uma nova mensagem deverá exibi-la, caso contrário deverá possibilitar ao usuário visualizar as mensagens armazenadas. Como a comunicação é feita de forma unidirecional (devido à limitações do Ginga), ou seja, a televisão que faz a busca dos dados em vez de o *webservice* ou aplicativo Android

enviarem a mensagem diretamente à TV. Portanto o programa executado na TV deve ficar em *loop* sempre buscando o arquivo e verificando alterações.

2.2. Descrição Detalhada da Aplicação Segunda Tela – Android

Para programação do dispositivo segunda tela foi escolhido utilizar o sistema operacional Android em detrimento a outros sistemas operacionais de dispositivos móveis pela familiaridade prévia de desenvolvimento nesta plataforma.

O aplicativo para dispositivo móvel deve possibilitar ao usuário inserir seu nome ou “*nickname*” (apelido) e em seguida possibilitar a inserção e envio de mensagens ao *webservice*. Foi implementada também uma funcionalidade para que o usuário visualize todas as mensagens enviados anteriormente por ele, em forma de lista.

Esquema geral da aplicação Android segue ilustrada na Figura 4 abaixo:

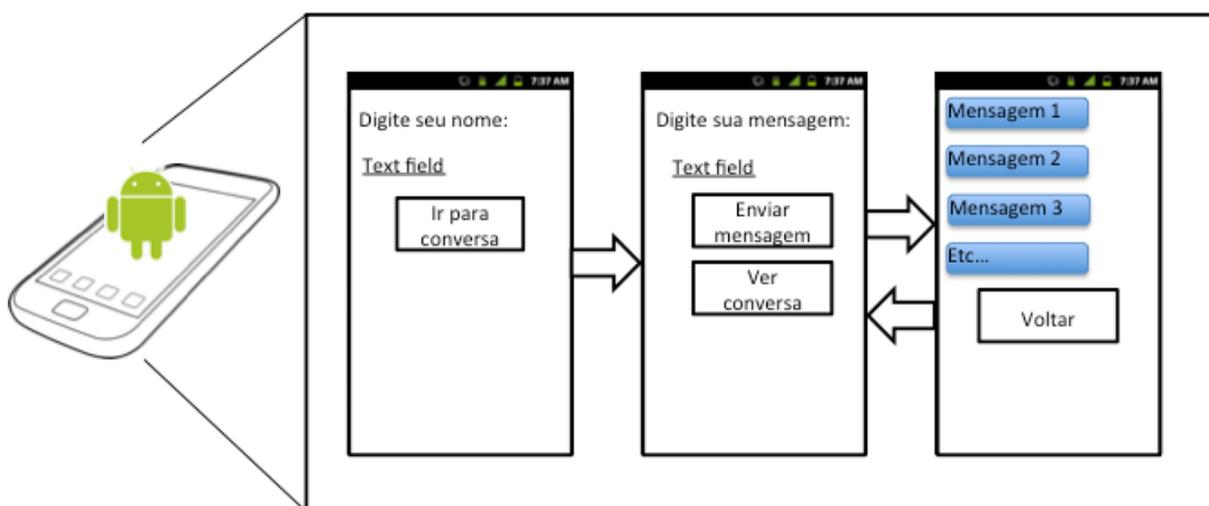


Figura 4: Diagrama de Fluxo de Tela da Aplicação Android

Com o esquema apresentado na Figura 4, todos os requisitos obrigatórios para o funcionamento da aplicação são atendidos, podendo o usuário inserir as informações necessárias e ainda visualizar todas as mensagens enviadas no aparelho utilizado.

O fluxo de telas foi implementado em forma de botões sendo que cada um deles leva a uma outra tela exceto para o caso do botão de envio de mensagem que realiza uma tarefa assíncrona para conexão com a Internet e envio das informações.

2.3. Descrição do Servidor e Aplicação PHP

Toda a inteligência no recebimento da mensagem, conversão para XML e disponibilização dos dados para os televisores se encontra no servidor. O servidor executa um programa escrito em PHP para recebimento dos valores de mensagem e usuário através de comando PHP via URL e executa um código para transformar estes valores em uma estrutura XML e tornar disponível o arquivo.

A estrutura do XML segue um formato semelhante ao apresentado no quadro de texto abaixo, Figura 5, onde diversas mensagens podem ser armazenadas dentro de uma mesma conversa estruturada.

```
<CONVERSA>
  <MENSAGEM>
    <TIMESTAMP>ddMMyyyyhhmmss</TIMESTAMP>
    <USUARIO>nomeDoFulano</USUARIO>
      <MENSAGEM>Diga alo para a TV!</MENSAGEM>
    </MENSAGEM>
    ...
</CONVERSA>
```

Figura 5: Exemplo de estrutura XML definida para o projeto

3. Resultados

3.1 Sobre as Ferramentas e Linguagens Utilizadas no Desenvolvimento

O desenvolvimento total do projeto requereu conhecimento nas linguagens de programação Lua, Java e PHP além de conhecimento da estrutura XML e NCL.

Na Figura 6 abaixo segue a visão estática do sistema, que é um esquema do sistema como um todo com descrições detalhadas para documentação junto com o diagrama de sequencia do fluxo de dados do sistema na Figura 7, que ilustra a sequencia que os dados percorrem uma vez que são enviados pelo usuário do dispositivo segunda tela para o servidor e os passos realizados entre o servidor e TV digital para que seja exibido na tela do televisor.

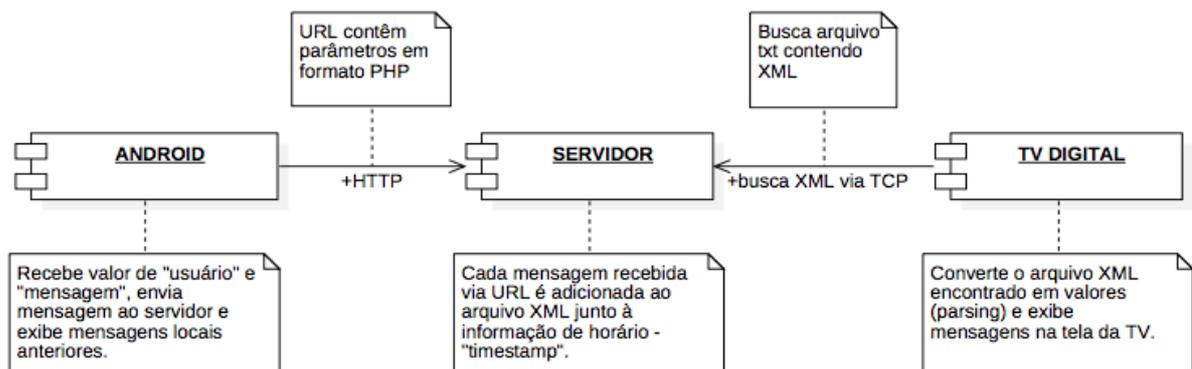


Figura 6: Esquema geral da descrição funcional do sistema como um todo.

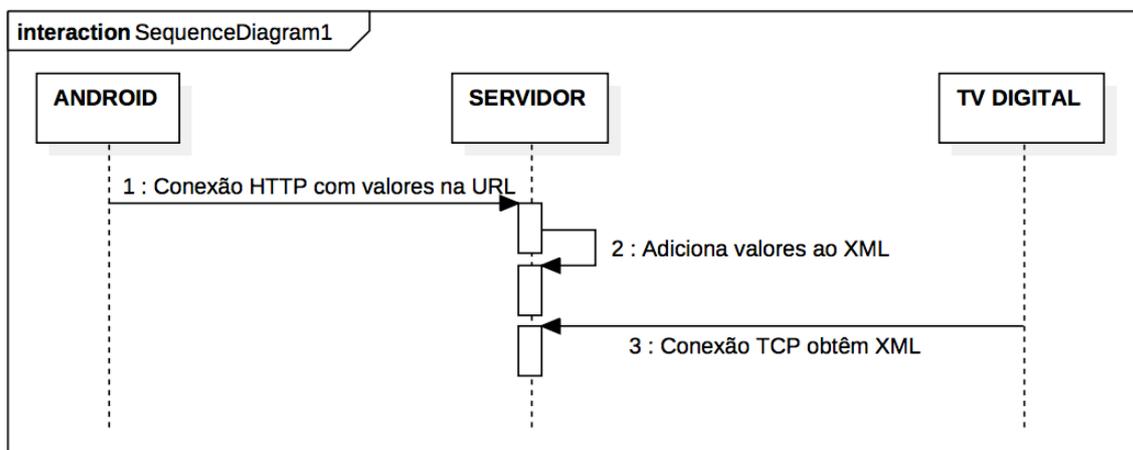


Figura 7: Diagrama de sequencia do fluxo de dados.

3.1.1 Desenvolvimento da Aplicação Ginga:

A linguagem padronizada para Ginga é o NCLua [11], composto por NCL – *Nested Context Language* para declarar e estruturar documentos de hipermídia, dando ordem e descrição entre as mídias sendo executadas no televisor e Lua para programação de fato. As duas linguagens utilizadas juntas possibilitam elaboração de programas baseados em eventos. A ferramenta escolhida para programação e testes foi o AstroBox [12], máquina virtual disponibilizada pela empresa TOTVS capaz de emular a implementação de Ginga de um televisor. A programação foi feita dentro da máquina virtual utilizando o editor de textos “gedit”. As Figuras 8 e 9 abaixo ilustram a plataforma de desenvolvimento para Ginga. Na Figura 8 visualizamos o ambiente de inicialização da máquina virtual exibindo os aspectos e configurações da máquina. A Figura 9 foi obtida com a máquina virtual em execução e demonstrando o ambiente utilizado para desenvolvimento de programas para TV Digital.

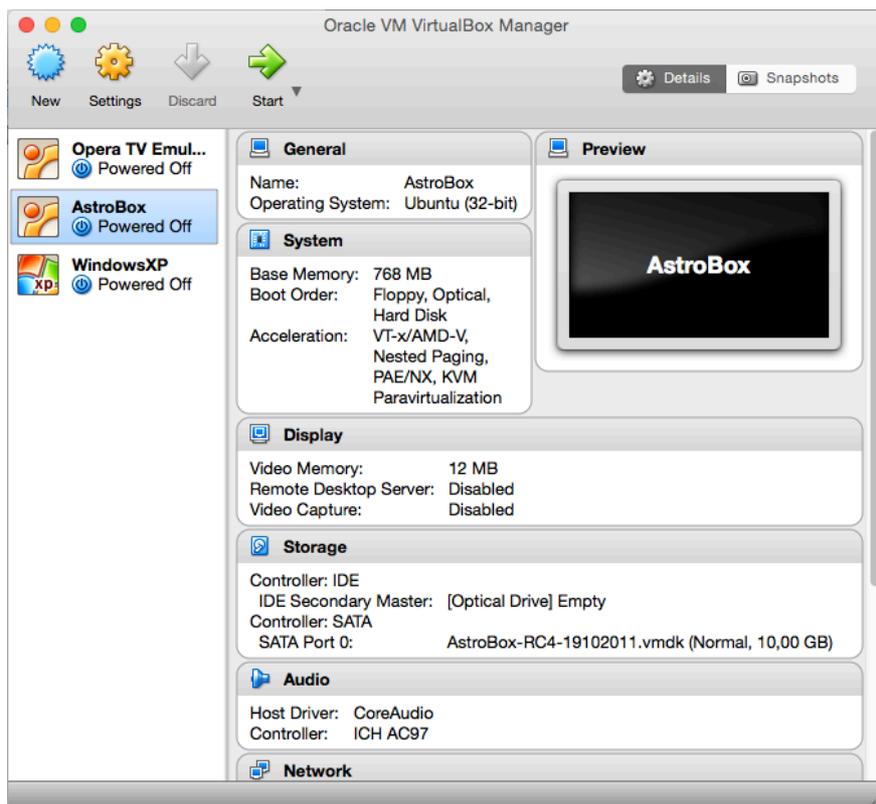


Figura 8: Inicialização da Máquina Virtual AstroBox

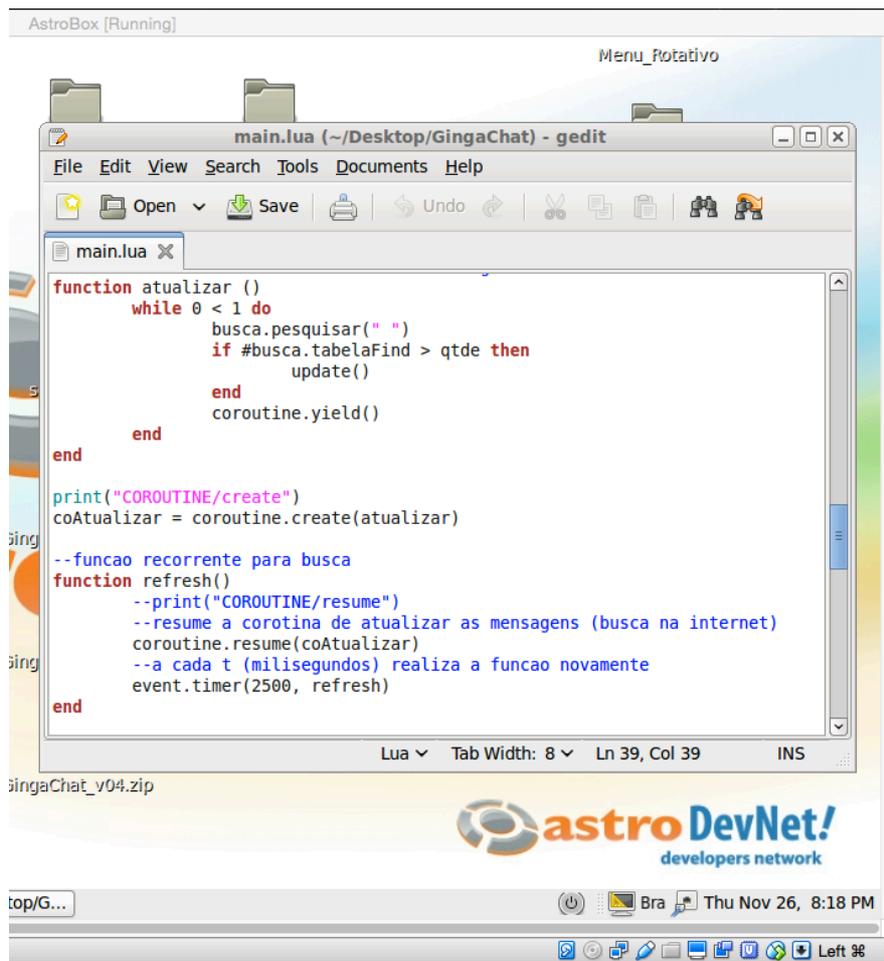
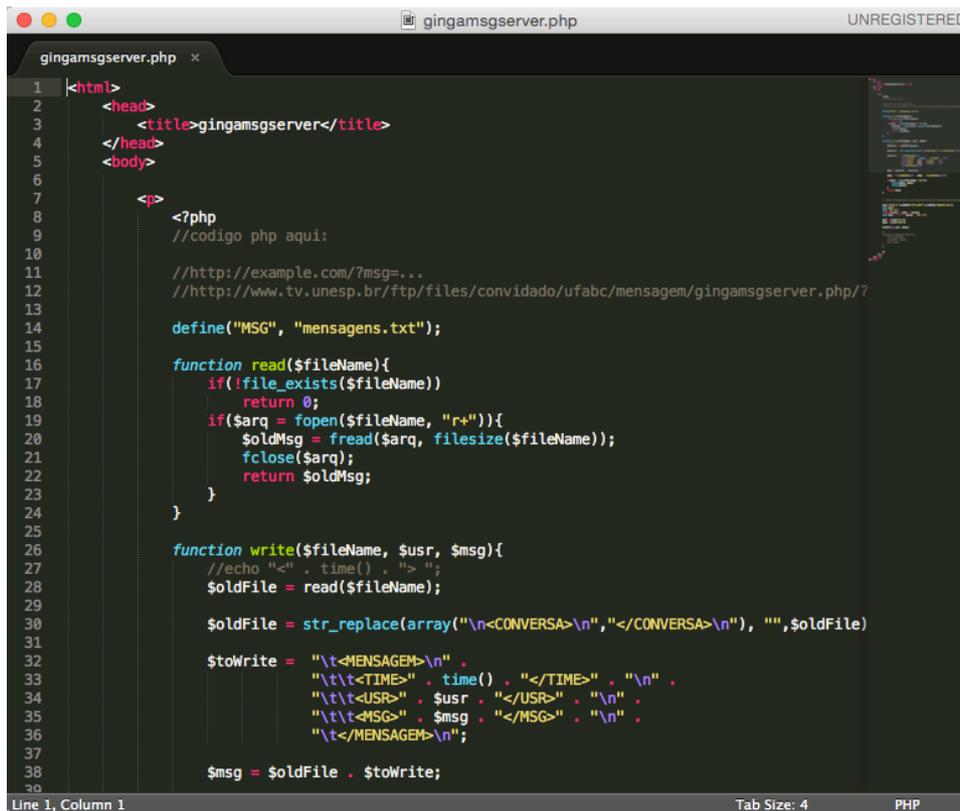


Figura 9: Programação em Lua dentro da Máquina Virtual AstroBox

3.1.2 Desenvolvimento do Webservice:

Foi obtido um domínio com hospedagem gratuita por tempo limitado para testes da aplicação em PHP. O endereço do domínio é felipemeloni.xyz e contém os arquivos utilizados no projeto. O desenvolvimento do programa foi feito pelo editor de textos “Sublime” e utilizado o “FileZilla” para envio do programa ao servidor. O endereço para testes do servidor é: felipemeloni.xyz/gingamsgserver.php e possibilita a inserção de mensagens pela extensão `?msg=”mensagem”&usr=”usuário”`. A Figura 10 ilustra o desenvolvimento do programa PHP para o servidor, utilizando o programa Sublime para edição do texto fonte do programa.



```
gingamsgserver.php x UNREGISTERED
gingamsgserver.php
1 <html>
2 <head>
3 <title>gingamsgserver</title>
4 </head>
5 <body>
6
7 <p>
8 <?php
9 //código php aqui:
10
11 //http://example.com/?msg=..
12 //http://www.tv.unesp.br/ftp/files/convidado/ufabc/mensagem/gingamsgserver.php/?
13
14 define("MSG", "mensagens.txt");
15
16 function read($fileName){
17 if(!file_exists($fileName))
18 return 0;
19 if($arq = fopen($fileName, "r+")){
20 $oldMsg = fread($arq, filesize($fileName));
21 fclose($arq);
22 return $oldMsg;
23 }
24 }
25
26 function write($fileName, $usr, $msg){
27 //echo "<" . time() . "> ";
28 $oldFile = read($fileName);
29
30 $oldFile = str_replace(array("\n<CONVERSA>\n", "</CONVERSA>\n"), "", $oldFile)
31
32 $toWrite = "\t<MENSAGEM>\n" .
33 "\t\t<TIME>" . time() . "</TIME>" . "\n" .
34 "\t\t<USR>" . $usr . "</USR>" . "\n" .
35 "\t\t<MSG>" . $msg . "</MSG>" . "\n" .
36 "\t</MENSAGEM>\n";
37
38 $msg = $oldFile . $toWrite;
39
Line 1, Column 1 Tab Size: 4 PHP
```

Figura 10: Programação PHP utilizando editor de texto Sublime

3.1.3 Desenvolvimento do Aplicativo Android:

Para esta etapa foi escolhido o “AndroidStudio” (Figura 13) para desenvolvimento de aplicativo nativo Android. A programação consiste de estruturação das telas utilizando XML e programação em Java para as tarefas. Foi iniciado um projeto em branco e pelas ferramentas da plataforma de desenvolvimento foram elaborados os *layouts* das telas e a programação das atividades de cada botão. A ferramenta de desenvolvimento possibilita a emulação do aplicativo criado, dentro de um dispositivo móvel virtual emulado ou a transferência do aplicativo executável para um dispositivo físico.

Além do diagrama de telas da Figura 4, a documentação de *software* para a aplicação *Android* consiste do diagrama de sequencia das interações entre usuário e atividades do aplicativo (Figura 11) e diagrama de estados (Figura 12). Atividades no contexto de programação *Android* são elementos compostos por tela mais código que definem os comandos que o usuário pode inserir na aplicação e a maneira como o programa se comporta. Neste diagrama de atividades, está demonstrado o comportamento das atividades e o que elas realizam. Logo que o programa é aberto, a atividade “*Main Activity*” é

inicializada, quando o usuário aperta o botão “IR PARA CHAT” a atividade “Main Activity” inicia a atividade “Chat Activity” da mesma forma que se for apertado o botão “VISUALIZAR MENSAGEM” ou “VOLTAR” uma atividade leva a outra e assim as telas da aplicação se alternam.

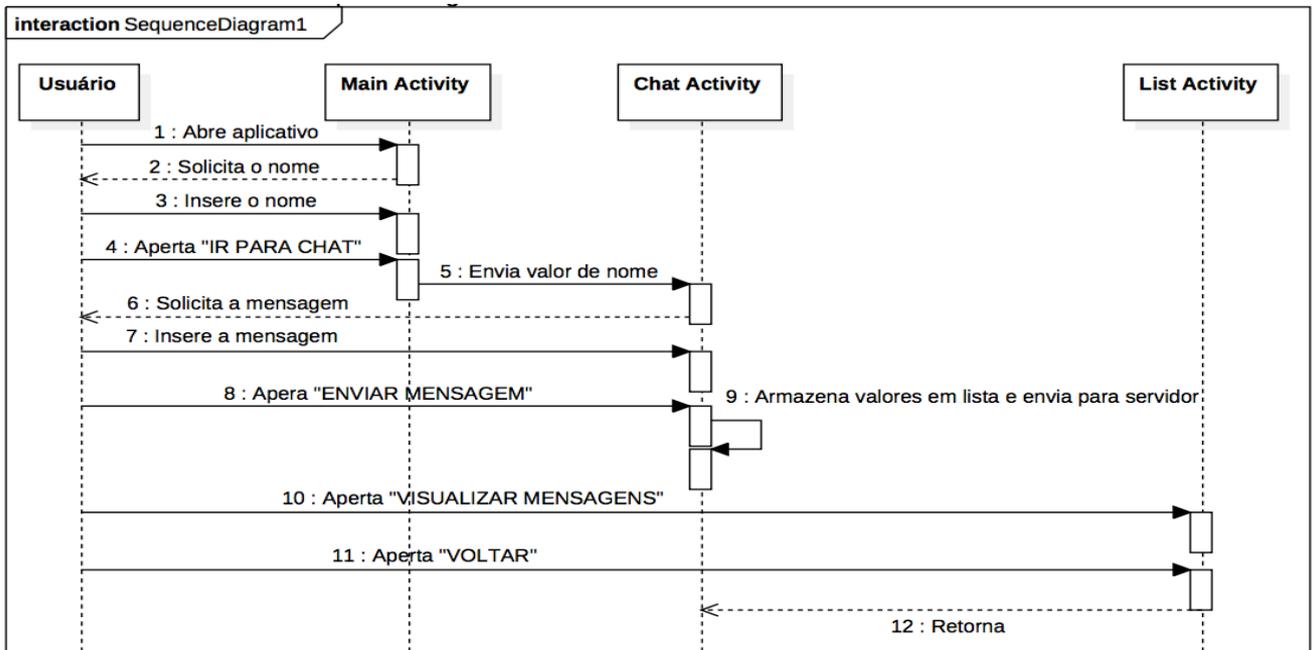


Figura 11: Diagrama dinâmico da sequencia de atividades da aplicação Android

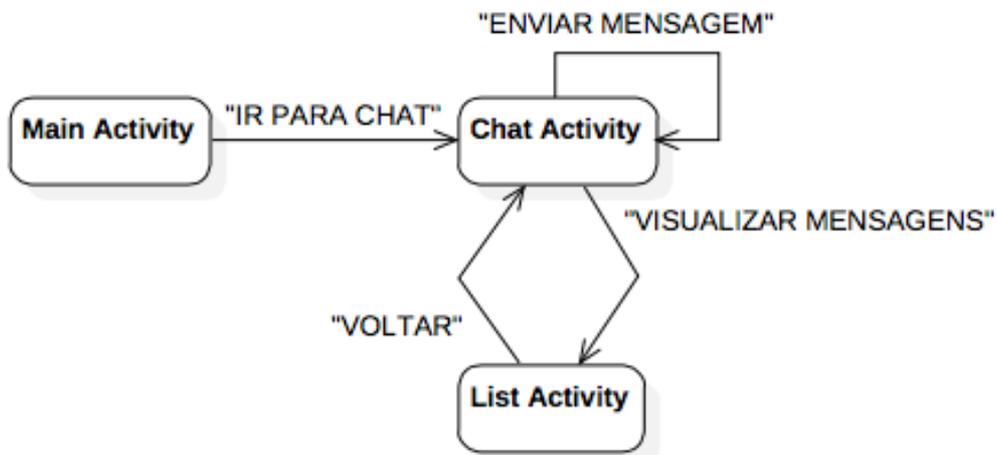


Figura 12: Representação do fluxo de atividades por maquinas de estado ativados por botões.

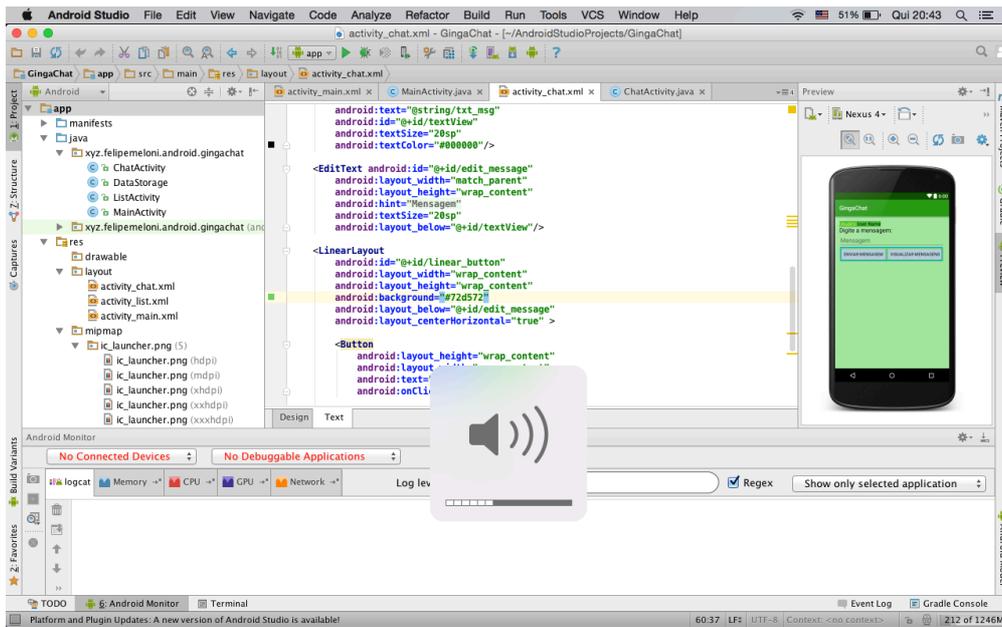


Figura 13: Desenvolvimento do Aplicativo dentro do AndroidStudio

3.2 Programação

Duas versões do aplicativo Android foram desenvolvidas ao longo da execução do trabalho. A diferença entre as duas consiste em qual *software* manipula o arquivo XML, se é o aplicativo Android ou se é o servidor PHP. Na primeira versão, o aplicativo Android obtém o XML do servidor, insere uma nova mensagem dentro da estrutura e sobrescreve via FTP o arquivo XML no servidor com uma nova versão atualizada. Na primeira versão existe a possibilidade de perda de mensagens quando mais de um usuário tenta realizar o envio de mensagem em um curto período de tempo. Para contornar este defeito foi desenvolvido a arquitetura com servidor onde o aplicativo Android envia uma mensagem por meio de URL ao servidor PHP em vez de manipular diretamente o arquivo XML no servidor. Neste trabalho será descrito o desenvolvimento da segunda versão apenas.

O software desenvolvido para ser executado na TV foi escrito nas linguagens NCL e Lua. A parte de NCL contém o mínimo necessário de código seguindo as especificações do padrão Ginga e inicia a execução do código Lua e define as áreas da tela que será destinado para a aplicação ou programação. A parte Lua executa códigos para acesso à Internet em períodos definidos buscando modificações no arquivo do servidor e realiza a exibição de novas mensagens na tela da TV no caso em que encontra novas mensagens. Enquanto não chega novas mensagens o usuário pode visualizar mensagens anteriores. Existe uma parte do código destinado para realizar o *parsing* do arquivo

XML para utilizar os dados como valores em tabelas. O *software* para Ginga também realiza rotinas para apresentar os dados na tela.

O servidor, escrito em PHP, recebe os valores de usuário e mensagem através da URL de conexão HTTP e para cada conexão recebida executa um código que insere os valores dentro de um arquivo TXT estruturado em XML e torna este arquivo disponível para acesso via TCP. A programação PHP foi feita para receber valores através da URL e estruturá-los conforme projetado para XML. Foram definidas funções para leitura e escrita em um arquivo de texto disponível para a TV executando Ginga.

A arquitetura detalhada do sistema com todas as funcionalidades e possíveis interações entre usuário e sistema estão detalhadas na Figura 14 abaixo. Os dois usuários descritos na arquitetura podem ser um mesmo usuário interagindo com o dispositivo segunda tela e assistindo à programação ou usuários distintos onde um apenas envia uma mensagem e o outro apenas a visualiza na televisão. O restante da figura descreve do lado esquerdo a interação entre usuário e aplicativo Android através do uso de botões e inserção de texto enquanto o meio e lado direito da figura representam os métodos executados para envio de mensagem conforme descrito no item 3.1 deste trabalho.

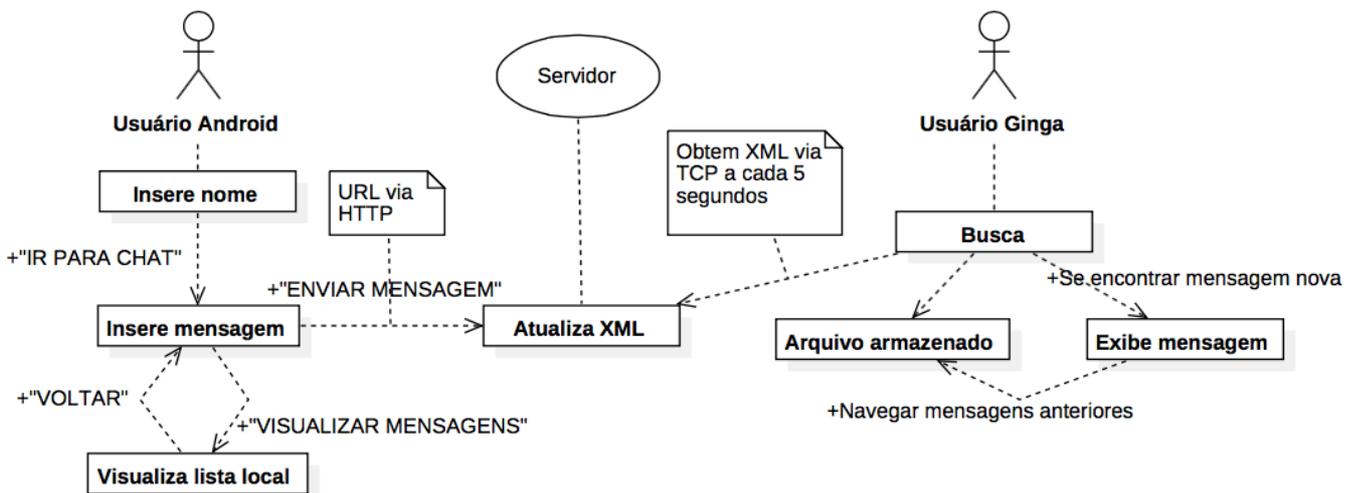


Figura 14: Diagrama de Módulo/Relação dos componentes do sistema

4. Simulação e Teste:

4.1 Teste do Aplicativo Android:

Os testes da aplicação Android foram realizados em um celular da fabricante LG modelo Nexus 4 com sistema operacional Android versão 5.0.1. O aplicativo Android foi denominado de GingaChat por ser onde o usuário realiza a inserção de mensagens na "sala" de chat. A sua aparência e funcionamento estão ilustradas nas Figuras 15 a 21 abaixo obtidas diretamente do aparelho celular onde foram realizados os testes. Nas Figuras se nota a sequência de telas onde inicialmente, ao abrir o aplicativo, é possibilitado a inserção do nome de usuário, Figuras 15 e 16. Em seguida ao apertar o botão "IR PARA CHAT" o aplicativo apresenta a tela de inserção de mensagem, Figuras 17, 18. Ao ser pressionado o botão "ENVIAR MENSAGEM" o aplicativo executa uma tarefa assíncrona para estabelecer conexão TCP com o servidor e enviar a mensagem através da URL, assim que o programa é executado com sucesso uma mensagem é exibida ao usuário em forma de *token*, Figura 19. Caso o usuário pressione o botão "VISUALIZAR MENSAGENS" o aplicativo exibe a lista de mensagens enviadas, Figuras 20 e 21.

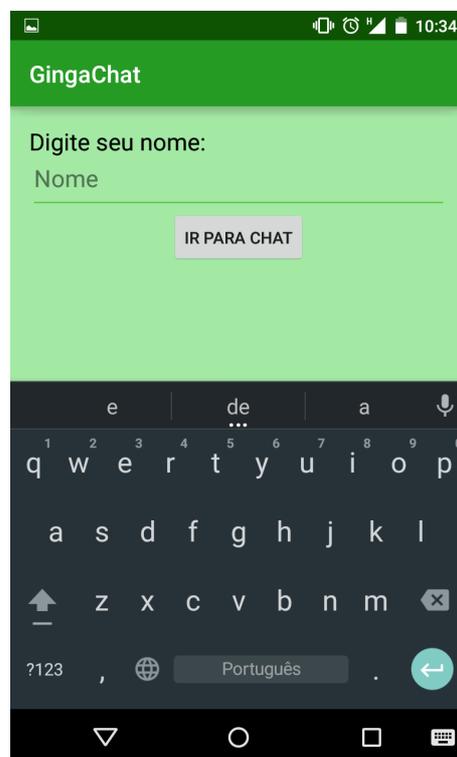


Figura 15: Tela inicial do aplicativo possibilitando a inserção do nome do usuário.

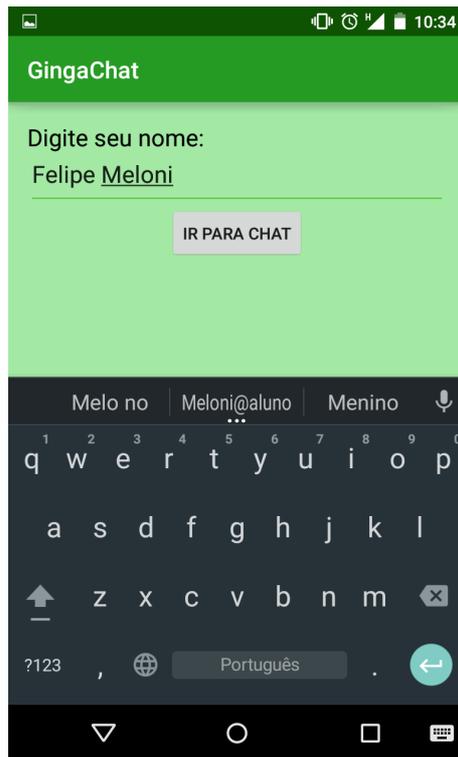


Figura 16: Tela inicial com nome de usuário inserido.



Figura 17: Tela para inserção de mensagem logo após apertar o botão “IR PARA CHAT” da janela anterior.

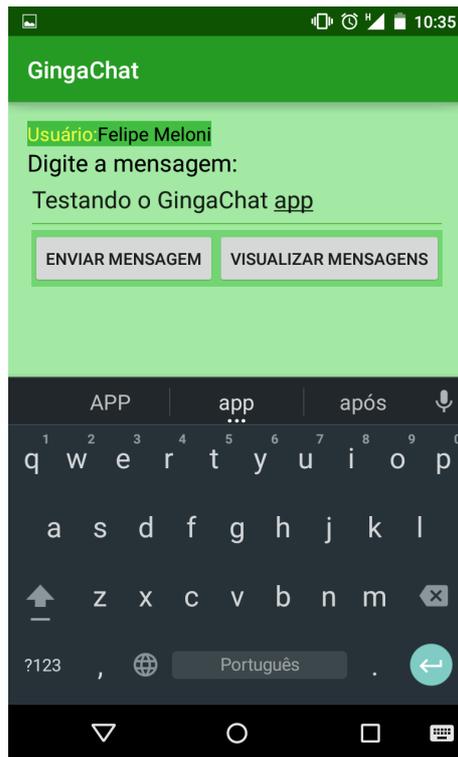


Figura 18: Tela de inserção de mensagem com texto inserido. Em seguida deverá ser pressionado o botão “ENVIAR MENSAGEM”.

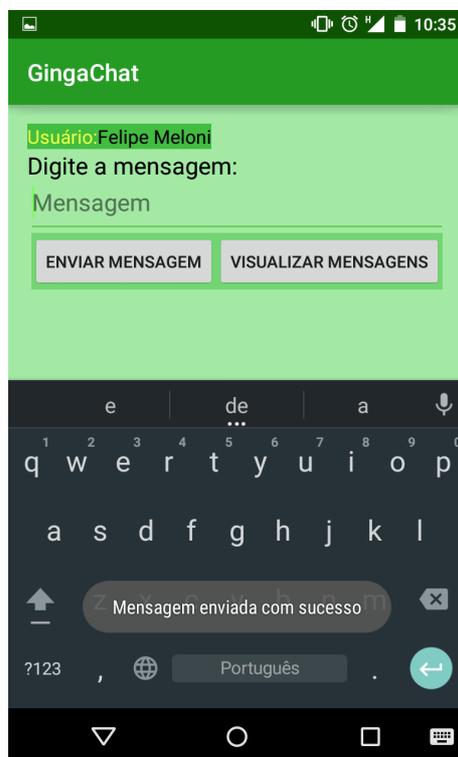


Figura 19: Tela de inserção de mensagem logo após o envio de mensagem. Um “token” é exibido após o envio bem sucedido da mensagem ao servidor.

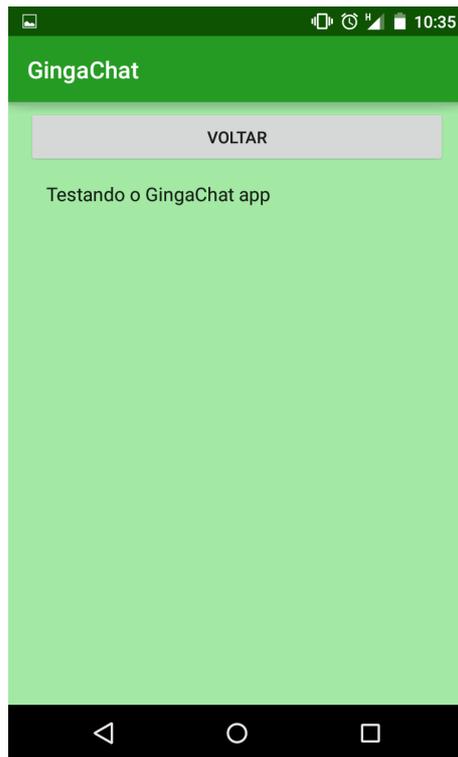


Figura 20: Tela de exibição das mensagens enviadas logo após a primeira mensagem ter sido enviada.



Figura 21: Tela de visualização de mensagens após o envio de algumas mensagens para demonstrar a estrutura da lista.

4.2 Teste do Servidor:

O código do servidor foi elaborado de maneira a possibilitar a visualização de seu funcionamento. Para realizar este teste foi utilizado um computador da Apple modelo MacBook Pro modelo “*early 2011*” com sistema operacional OS X Yosemite e browser Safari versão 8.0.8. É possível realizar a inserção de mensagens mesmo sem o uso do aplicativo, utilizando o endereço URL do servidor em um browser como será demonstrado nas Figuras 22 a 25 abaixo. Utilizando o endereço www.tv.unesp.br/ftp/files/convidado/ufabc/mensagem/ é exibido o conteúdo da página mostrando os arquivos encontrados, Figura 22. Pode ser inserido mensagens pela URL através da endereçamento `/?urs=usuário%msg=mensagem`, demonstrado na Figura 23 onde após a requisição uma tela é exibida mostrando o sucesso da operação e o resultado, Figura 24. O código PHP trabalha em conjunto com um arquivo .txt onde fica armazenado o conteúdo das mensagens, o conteúdo deste arquivo se encontra ilustrado na Figura 25. O arquivo PHP do servidor foi denominado de `GingaMsgServer` por realizar a tarefa de servidor do projeto.



Figura 22: Arquivos encontrados no endereçamento da localização do arquivo PHP do servidor.



Figura 23: Exemplo de inserção de mensagens pela URL.



Figura 24: Resultado da operação de envio de mensagem pela URL de um browser.

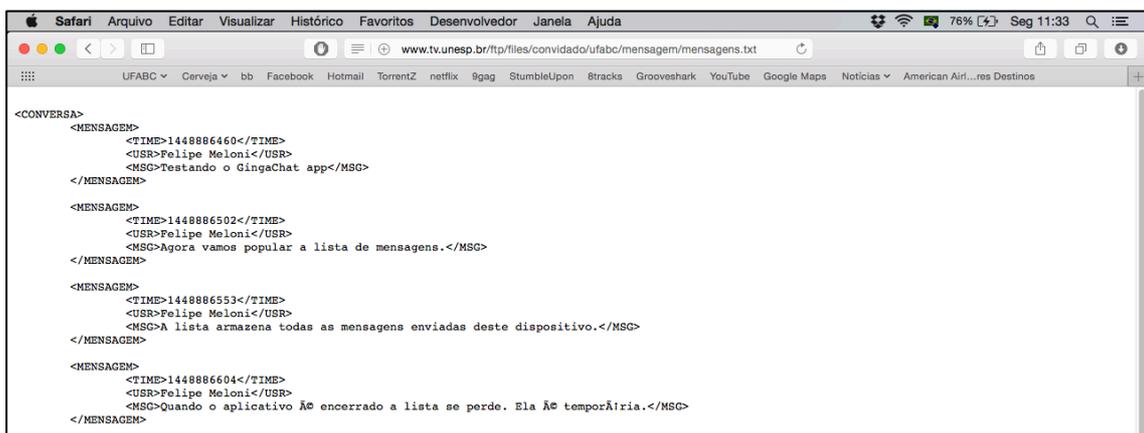


Figura 25: Conteúdo do arquivo .txt que é utilizado pelo código PHP.

4.3 Teste da Aplicação Ginga:

Para esta parte de testes foi utilizada máquina virtual dentro do VirtualBox da Oracle utilizando AstroBox fornecido pela Totvs, uma implementação de Ginga baseado em Ubuntu. Nas Figuras 20 e 21 abaixo são demonstrados o funcionamento do programa na tela da TV emulada. Na parte preta seria exibida a programação televisiva recebida diretamente da emissora sendo que o aplicativo se utiliza apenas da parte em cinza para exibição das mensagens.

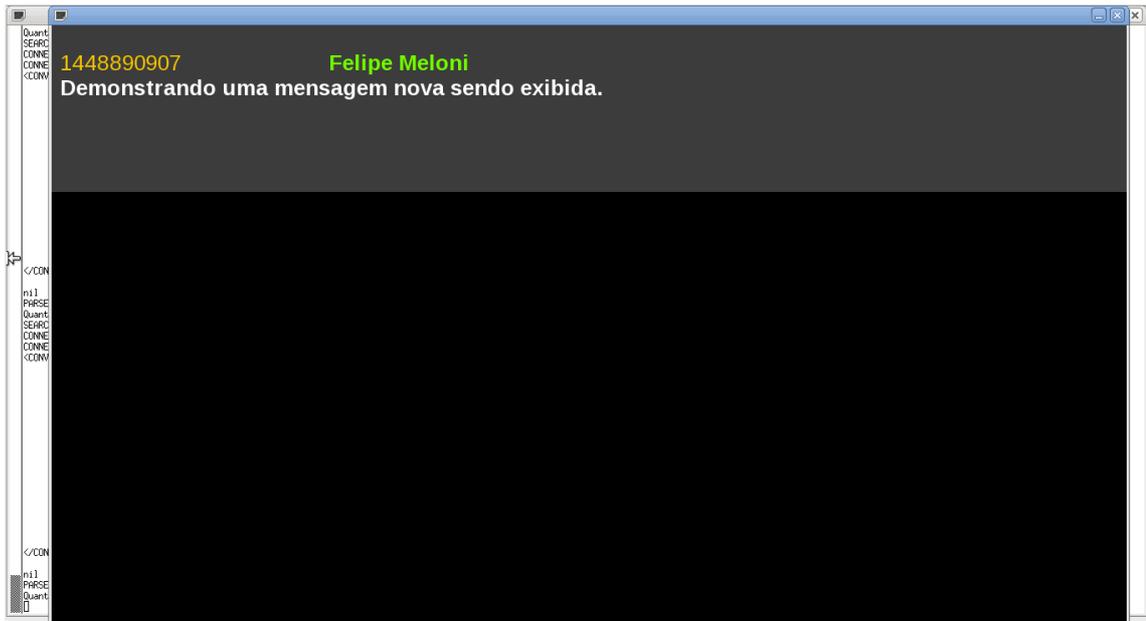


Figura 26: Teste da exibição de mensagens na TV emulada utilizando Ginga. Inicialmente a mensagem exibida foi apresentada automaticamente logo que foi recebida do servidor.

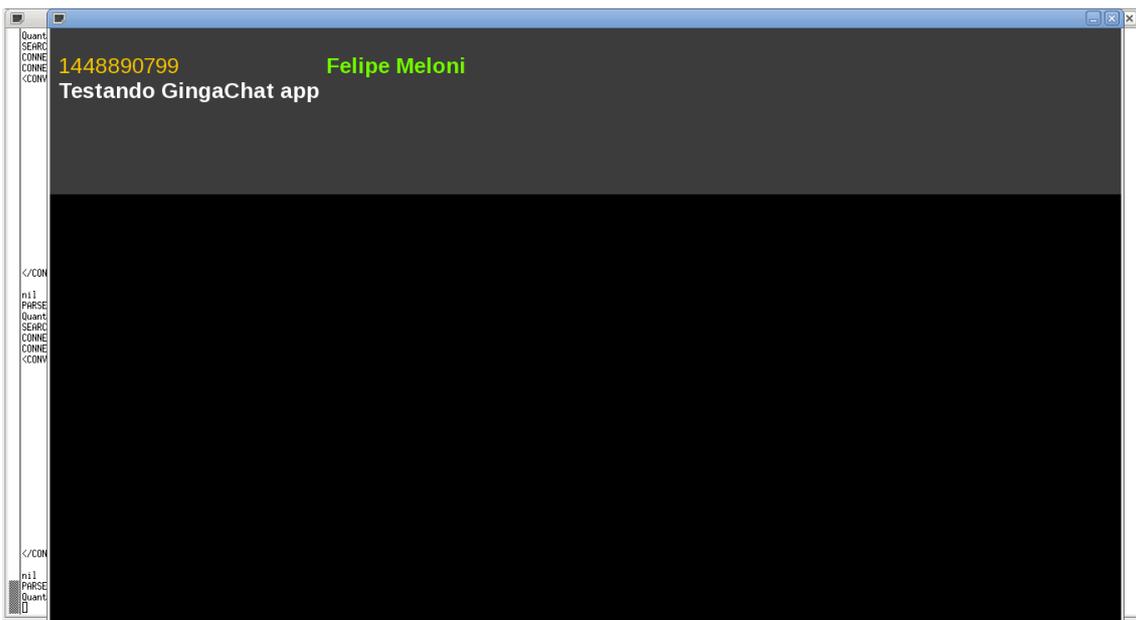


Figura 27: A mensagem exibida não é a ultima recebida e foi visualizada através da navegação pelas setas direcionais do controle remoto.

Conclusão

Com este projeto, é possível concluir que é viável desenvolver novas aplicações interativas e funcionais para TV digital utilizando dispositivo segunda tela. A prova de conceito obteve sucesso no envio de mensagens de um celular Android para a TV Ginga sendo emulada em uma máquina virtual com utilização de um servidor para armazenamento das mensagens.

Novos cenários podem ser deslumbrados com as possibilidades de se interagir com a televisão através de um dispositivo móvel transferindo dados para que sejam exibidos na TV. Com ampliação deste projeto seria possível a criação de um chat onde mais de um usuário (em diferentes dispositivos) enviam mensagens para o mesmo servidor e assim podem estar socialmente conectados pela TV digital.

Referências Bibliográficas:

- [1] Luciano Leonel Mendes, “SBTVD – Uma Visão Sobre a TV Digital no Brasil”, T&C Amazônia, Ano V, Número 12, Outubro de 2007.
- [2] Pesquisa TIC Governo Eletronico, Disponível em <<http://cgi.br/publicacoes/indice/pesquisas/>>, Acesso em 26/03/2015
- [3] Digital terrestrial television — Data coding and transmission specification for digital broadcasting, Disponível em <<http://www.abnt.org.br/tvdigital>>, Acesso em 30/12/2015
- [4] IBGE PNAD 2011, Pesquisa Nacional por Amostragem de Domicílios 2011, Disponível em: <www.ibge.gov.br>, Acesso em 09/03/2015
- [5] AppleTV, Disponível em < <https://www.apple.com/br/appletv/>>, Acesso em: 23/03/2015
- [6] Chromecast, Disponível em < <https://www.google.com.br/chrome/devices/chromecast/>>, Acesso em 23/03/2015
- [7] (FP7-ICT-2013-EU-Brazil): Work Programme for the 2nd Coordinated Call Brazil-Europe, Point 4: Hybrid broadcast-broadband application and services.
- [8] Site Oficial do DVB – História, Disponível em<<http://www.dvb.org/about/history>>, Acesso em 25/03/2015).
- [9] Site oficial do Ginga, Disponível em <<http://www.ginga.org.br/pt-br/>>, Acesso em 23/03/2015
- [10] Silva Filho, M. C. da, Gondim, P. R. de L. (2011) “NCLua SOAP: Acesso a Web Services em aplicações de TVDi.” Em: WCGE, 2011, Florianópolis.
- [11] Ginga-NCL – Declarative DTV Middleware. Disponível em <<http://www.gingancl.org.br>>, Acesso em 23/03/2015.
- [12] astroDevNet – AstroBox. Disponível em <www.astrodevnet.com.br>, Acesso em 25/08/2015.