

Marco Aurélio Friesz Junior

Quiz para TV Digital Interativa: Ambiente
completo de execução

Juiz de Fora

8 de dezembro de 2010

Marco Aurélio Freesz Junior

Quiz para TV Digital Interativa: Ambiente completo de execução

Monografia apresentada para obtenção do Grau de Bacharel em Ciência da Computação pela Universidade Federal de Juiz de Fora.

Orientador:
Eduardo Barrére

INSTITUTO DE CIÊNCIAS EXATAS
DEPARTAMENTO DE CIÊNCIA DA COMPUTAÇÃO
UNIVERSIDADE FEDERAL DE JUIZ DE FORA

Juiz de Fora

8 de dezembro de 2010

Monografia de Projeto Final de Graduação sob o título “**Quiz para TV Digital Interativa: Ambiente completo de execução**”, defendida por Marco Aurélio Freesz Junior e aprovada em 8 de dezembro de 2010, em Juiz de Fora, Estado de Minas Gerais, pela banca examinadora constituída pelos professores:

Prof. D. Sc. Eduardo Barrére
Orientador

Prof. M. Sc. Eduardo Pagani Julio
Universidade Federal de Juiz de Fora

Prof^a. D. Sc. Ana Paula Couto da Silva
Universidade Federal de Juiz de Fora

Agradecimentos

Agradeço primeiramente a Deus por estar sempre no meu caminho, me proporcionando tranquilidade para a obtenção do conhecimento exigido ao longo do curso e na elaboração deste projeto.

Ao meu Orientador que foi muito importante ao me ajudar e influenciar para o estudo aqui publicado.

Além de todos os alunos da turma, e os demais que diretamente ou indiretamente me ajudaram na conclusão desse trabalho.

Resumo

Este trabalho consiste no desenvolvimento e implementação de um ambiente completo de execução de um Quiz para TV Digital, que envolve elaboração, envio, execução e a análise das votações de um Quiz. Além das informações básicas relacionadas a quantidade de votos, a emissora poderá avaliar questões sobre regiões e perfis dos usuários. Portanto, a proposta abrange o ambiente como um todo e não somente no desenvolvimento de um Quiz para TV Digital, destacando a arquitetura e componentes técnicos da TV Digital, assim como as linguagens usadas para a aplicação, baseadas no *middleware* padrão para a TV Digital Brasileira, o Ginga.

Palavras-chave: TV Digital, Ginga, Interatividade, Canal de Retorno, QUIZ.

Abstract

This work consists the development and implementation of a complete environment for implementing a Quiz for Digital TV, with development, delivery, execution and analysis of the votes on a Quiz. Besides the basic information regarding the number of votes, the station may evaluate questions about user regions and profiles. Therefore, the proposal covers the environment as a whole and not only the development of a Quiz for Digital TV, highlighting the architecture and technical components of Digital TV, such as languages used for implementation, based on standard *middleware* for the Brazilian Digital TV, Ginga.

Keywords: Digital TV, Interactivity, Return channel, QUIZ.

Sumário

Lista de abreviaturas e siglas	p. 8
Lista de Figuras	p. 9
1 Introdução	p. 11
1.1 Motivação	p. 12
1.2 Objetivos	p. 12
2 TV Digital	p. 13
2.1 Histórico	p. 13
2.2 Sistema Brasileiro de TV Digital - SBTVD	p. 14
2.3 Arquitetura da TV Digital e do <i>middleware</i> GINGA	p. 18
3 Interação na TV Digital	p. 20
3.1 Aplicações em TV Digital	p. 21
3.1.1 Exemplo de Quiz	p. 22
4 Linguagens	p. 23
4.1 Nested Context Language (NCL)	p. 23
4.1.1 Estrutura	p. 25
4.2 NCLua	p. 27
5 Ambiente Completo de Execução do Quiz	p. 28
5.1 Módulo Gerador do Quiz	p. 30
5.1.1 Base de Dados da Emissora	p. 31

5.2	Módulo Aplicação Quiz	p. 32
5.3	Módulo de Análise e Consulta	p. 34
5.3.1	Base de Dados do PSI	p. 37
6	Considerações Finais	p. 38
6.1	Conclusão	p. 38
6.2	Trabalhos Futuros	p. 39
	Referências	p. 40
	Anexo A – Exemplo do <i>script</i> LUA principal do projeto	p. 42
	Anexo B – Exemplo do documento NCL do projeto	p. 44

Lista de abreviaturas e siglas

ABNT - Associação Brasileira de Normas Técnicas

ATSC - Advanced Television Systems Committee

API - Application Programming Interface

DVB - Digital Video Broadcasting

EDTV - Enhanced Definition Television

HDTV - High Definition Television

IP - Internet Protocol

ISDB - Integrated Services Digital Broadcasting

LDTV - Low Definition Television

MPEG - Moving Picture Experts Group

NCL - Nested Context Language

PSI - Provedor de Serviço Interativo

SBTVD - Sistema Brasileiro de TV Digital Terrestre

SDTV - Standard Definition Television

STB - Set Top Box

TV - Televisão

TVDI - TV Digital e Interativa

XHTML - eXtensible Hypertext Markup Language

XML - Extensible Markup Language

Lista de Figuras

1	Cronograma de implantação.	p. 14
2	Algumas possíveis combinações para o uso do canal de 6 MHz em um sistema de TV digital, compressão H.264.	p. 16
3	Componentes fundamentais.	p. 17
4	Processo de envio.	p. 17
5	Arquitetura Set-Top-Box (Middleware GINGA).	p. 18
6	Estrutura do Ginga.	p. 19
7	Aplicações Possíveis: (a) Multi-camêra em partida de futebol. e (b) Compra de automóvel.	p. 21
8	Exemplo de Quiz "Gol da Rodada".	p. 22
9	Estrutura documento NCL.	p. 25
10	Associação entre nó de mídia ("video1"),descritor ("dVideo1") e região ("rgVideo1").	p. 26
11	Arquitetura do Projeto.	p. 29
12	Módulo Gerador do Quiz.	p. 30
13	Layout da aplicação geradora.	p. 31
14	Tabela Enquete.	p. 32
15	Módulo Aplicação Quiz.	p. 32
16	XML de resposta.	p. 33
17	Módulo de Análise e Consulta.	p. 34
18	Consultor sobre a quantidade de votos.	p. 35
19	Filtragem por Regiões.	p. 35

20	Filtragem por Sexo.	p. 36
21	Filtragem por faixa etária.	p. 36
22	Base de dados do PSI.	p. 37
23	<i>script</i> LUA principal - parte 1.	p. 42
24	<i>script</i> LUA principal - parte 2.	p. 43
25	Documento NCL.	p. 44

1 *Introdução*

Atualmente o Brasil está passando por uma fase de transição na questão de digitalização da TV. Pensando na evolução da TV na sociedade brasileira, e em todo o mundo, desde os primeiros aparelhos até essa nova onda de televisores, podemos analisar os diversos níveis de interação. De acordo com lemos, podemos classificar os níveis de interatividade, desde ligar a televisão; trocar de canais; aumentar o volume; usar aparelhos para reprodução de filme; opinar sobre um conteúdo por telefone ou correio; até entrar no nível de interação mais profundo, como na Internet, permitindo ao telespectador realizar escolhas de câmeras, ou navegar pelas informações, podendo até mesmo comprar produtos disponibilizados pela emissora em uma determinada programação. Algumas pessoas já tratam essa questão de maneira até mais profunda, como em [1], onde o usuário poderá enviar seu próprio conteúdo, similar ao que é feito hoje na Internet com a criação de web sites. Dessa forma, podemos perceber que até então a interação do telespectador com a televisão estava se dando de uma maneira 'passiva', já que normalmente o usuário apenas recebia informações, não realizando ações interativas com a mídia [2].

Com o advento da TV digital, tornou-se possível o transporte de dados através de um novo canal, obtido devido a digitalização da transmissão do sinal. Dessa forma, a mídia consegue obter, além do ganho na qualidade de imagem e som, o crescimento em desenvolvimento de aplicações, proporcionando ao telespectador um maior entretenimento, trazendo benefícios de interatividade para ambos os lados.

Através dessa nova tecnologia é possível pensar em diversos aplicativos que os telespectadores poderão usar, como obter informações climáticas, participar de enquetes geradas pela emissora e enviada em conjunto com alguma programação, ou até aplicativos que funcionem de forma independente de um programa de TV. Sendo assim, o novo sistema digital de televisão possibilita aos telespectadores diversas funcionalidades, que se equivalem a uma Internet, pois dentro dessa nova arquitetura é possível a utilização de um canal de retorno, possibilitando ao usuário o envio de dados.

Dessa forma, este trabalho propõe a elaboração de todo o ambiente necessário para a implantação de um sistema de enquetes (Quiz) obedecendo as normas do padrão brasileiro de televisão digital, o *middleware* GINGA-NCL. Para isto, será mostrado as linguagens utilizadas para as aplicações, e também explicado todos os fluxos presentes entre os módulos do ambiente [2].

1.1 Motivação

Hoje em dia muito se fala sobre o potencial da TV Digital quanto a qualidade de imagem e áudio, porém além dessas características devemos destacar a interatividade. Essa se destaca a cada dia por meio de novas aplicações interessantes que são desenvolvidas em NCL e LUA que podem ser encontradas na Internet, como diversos tipos de Quiz.

Porém estes tipos de aplicações, muitas das vezes, não são tratadas de forma completa, ou seja, não é desenvolvido nenhum ambiente completo de produção,divulgação e coleta de informações sobre uma determinada enquete. Para isso nesse trabalho o Quiz é produzido com base na linguagem NCL e NCLua de acordo com os padrões estabelecidos pelo *middleware* Ginga-NCL.

1.2 Objetivos

Com a difusão dessa tecnologia, a interatividade se destaca como uma característica inovadora nesse tipo de mídia, sendo assim, esse projeto tem como objetivo demonstrar todo o ambiente necessário para a produção, divulgação e coleta de dados de um Quiz. Para isso é estabelecido toda uma arquitetura desde a criação do Quiz de uma maneira simples, sem a necessidade de conhecimento sobre a linguagem adotada no Brasil em TVDI (TV Digital e Interativa), até a consulta dos dados em um servidor online denominado PSI (Provedor de Serviços Interativos) o qual é responsável por guardas as informação de cada voto recebido do telespectador. Consultas estas que podem ser a quantidade de votos que cada opção obteve, bem como levantamento sobre perfis de usuários.

2 TV Digital

Nesse Capítulo é apresentado um breve histórico sobre a TV Digital, tão bem quanto os componentes envolvidos nessa relação emissora x telespectador, citando os padrões utilizados hoje no mundo e detalhando o padrão nacional adotado SBTVD (Sistema Brasileiro de Televisão Digital). Além disso, é mostrado a arquitetura definida por esse padrão, bem como seus módulos.

2.1 Histórico

O desenvolvimento da tecnologia de TV Digital teve início no final da década de 1980 se estabelecendo em meados de 1990 com dois padrões inicialmente definidos: o Europeu (DVB) e o Americano (ATSC). Em seguida, apesar do grande poder em pesquisas relacionadas à TVs de alta definição, os japoneses lançaram comercialmente outro padrão para sistema de transmissão digital em 2003, denominado ISDB [3]. Esses padrões foram concebidos de acordo com as características de cada país e seguindo os seus propósitos, de forma que cada país tivesse inicialmente uma idéia de aplicabilidade. Como alguns exemplos, podemos citar que a idéia inicial do padrão americano era estabelecer a qualidade de imagem. O padrão japonês, além da qualidade de imagem, visava a mobilidade, e por sua vez o europeu visava a multiprogramação e interatividade.

No Brasil, houve a adoção da base do padrão de transmissão japonês ISDB denominado SBTVD em meados de 2005, oferecendo diferenciais em relação aos padrões atuais, presentes em outras nações. Estes diferenciais foram obtidos devido a junção das técnicas de compressão de áudio e vídeo adotados no Brasil com a tecnologia japonesa. Além disso, foi acrescentado tecnologias produzidas pela Pontifícia Universidade Católica do Rio de Janeiro (PUC-Rio) e pela Universidade Federal da Paraíba (UFPB). Dessa forma, o padrão nacional propõe transmissão de conteúdo de altíssima qualidade para os telespectadores, possibilitando ao mesmo tempo a recepção móvel e portátil dos sinais de TV digital nos mais diversos tipos de dispositivos, como celulares, mini-televisores, entre ou-

tros. Além disso é distribuído sem custos, já que o SBTVD é um sistema aberto, livre e gratuito.

O início da transmissão digital no Brasil ocorreu em 02 de Dezembro de 2007, em São Paulo, sendo seguida de outras cidades. A figura 1 apresenta o cronograma de adoção do sinal digital em outras regiões [4].



Figura 1: Cronograma de implantação.

Apesar dos objetivos iniciais de cada país quanto as suas políticas de implantação e tecnologias da transmissão, hoje em dia podemos perceber que os padrões tratados por cada país estão buscando equacionar todas essas funcionalidades da nova era digital através da flexibilização, atendendo assim, aos desejos de todos os telespectadores [5].

2.2 Sistema Brasileiro de TV Digital - SBTVD

O Sistema Brasileiro de TV Digital teve suas normas aprovadas pela Associação Brasileira de Normas Técnicas (ABNT) em fins de 2007, sendo atualmente composta por 13 documentos que formalizam as funcionalidades de aspectos, como os sistemas de transmissão, codificação de vídeo e áudio, além de codificação de dados e interatividade.

Inicialmente, o projeto teve apoio da FINEP e CPqD que ofereceram as universidades e centros de pesquisas a oportunidade de desenvolvimento completo do sistema digital brasileiro. Porém, em meados de 2005, foi adotado como base para o projeto o padrão

japonês ISDB. Sendo assim, aproveitando as pesquisas iniciadas em 2004, foi desenvolvido uma forma híbrida que define a solução atualmente adotada. Essa solução é obtida de forma que nas características de transmissão e recepção são utilizadas as definições propostas pelo ISDB. Já aspectos de software e interatividade fundamentados no *middleware*, assim como controle de conteúdo, são definidos pelas propostas nacionais [6].

O SBTVD apresenta características próprias, como a forma de compressão de vídeo, onde é adotado o H.264, diferente dos outros padrões que fazem uso do MPEG-2. Com essa técnica é possível manter a qualidade de imagem diminuindo a quantidade de bits, aumentando assim, a usabilidade espectral. Outra característica marcante no padrão nacional é o *middleware* adotado, denominado Ginga [7], desenvolvido pela PUC e pela UFPB conforme já mencionado anteriormente e que será um pouco mais detalhado em 2.3.

Devemos entender que um padrão de TV Digital tem por finalidade tratar os seguintes módulos:

- Conteúdo e aplicação;
- Compressão;
- Middleware;
- Multiplexação;
- Transmissão/Recepção;
- Canal de Interatividade.

Além desses módulos, que serão explicados mais a fundo na seção 2.3, devemos ressaltar que o padrão adotado define, também, questões relacionadas a qualidade de imagem. Diferente do que as pessoas pensam, o sinal de TV Digital não é somente difundido com alta definição de imagem, existem outros tipos de qualidade de imagem também utilizados nessa nova tecnologia. Atualmente são quatro tipos adotados de acordo com [8]:

- LDTV, usado geralmente para dispositivos móveis;
- SDTV, equivalente ao analógico, usado geralmente em TV's por assinatura via satélite;
- EDTV, um pouco de ganho na qualidade de imagem em relação ao SDTV;

- HDTV, melhor qualidade de imagem possível hoje, em TV Digital terrestre.

Sendo assim, podemos dividir o canal de TV Digital em até 4 partes, dependendo da qualidade de imagem utilizada para transmissões, em um canal de 6 MHz com utilização do padrão de compressão adotado pelo Brasil, H.264. Para ilustrar as possíveis divisões do canal de acordo com a qualidade de transmissão, a figura 2 é apresentada:

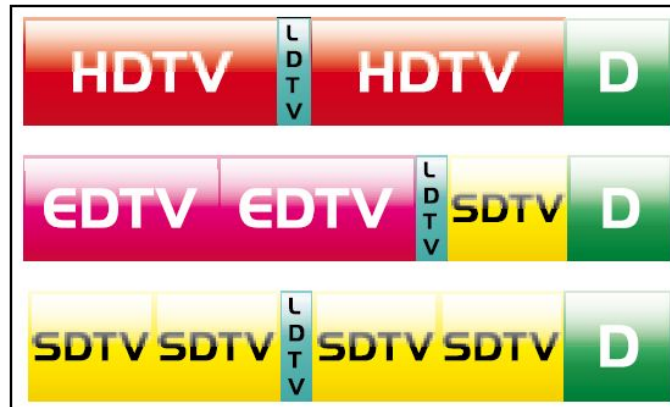


Figura 2: Algumas possíveis combinações para o uso do canal de 6 MHz em um sistema de TV digital, compressão H.264.

Além dos padrões de imagem, um canal de TV Digital apresenta uma faixa destinada a transmissão de dados, referenciado por (D) na figura 2 [8].

A transmissão em TV Digital interativa é realizada em *broadcast* e é composta por três partes principais: Difusor, Receptor e Canal de difusão. O difusor é responsável por prover o conteúdo da programação e/ou aplicação a ser transmitida, o receptor é aonde as informações recebidas são tratadas pelo receptor digital (STB), já o canal de difusão é o meio onde ocorre a troca de informações entre emissora e telespectador. Geralmente a transmissão da emissora é feita via AR em *broadcast* e o retorno do usuário via CABO. Esse retorno do usuário é produzido de acordo com a ação do telespectador quanto a algum item de interatividade disponibilizado em sua TV, seja um Quiz, uma compra, entre outros [3].

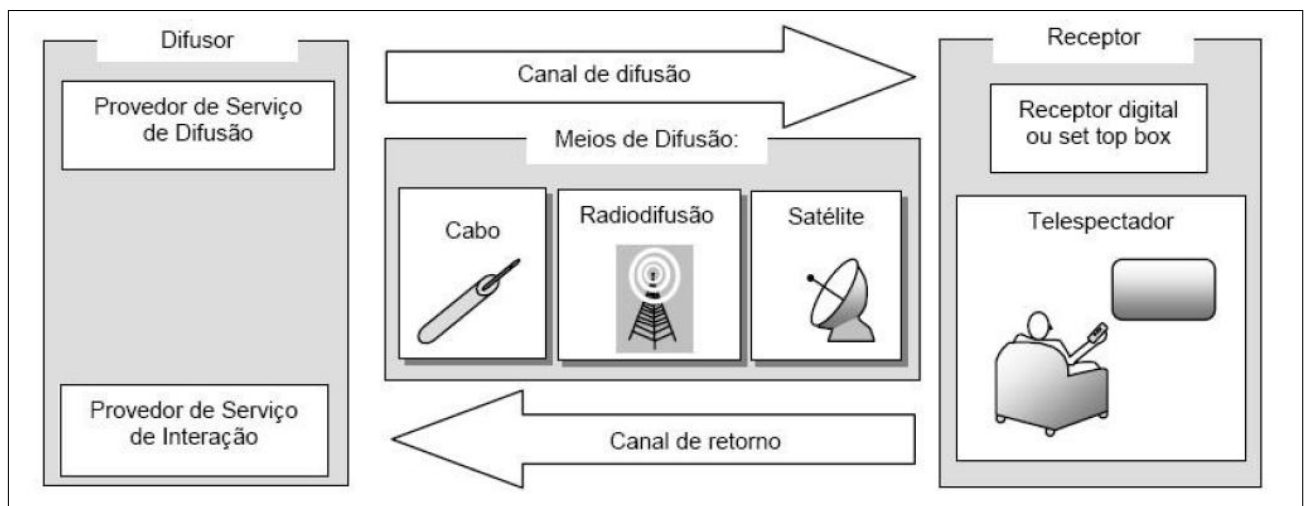


Figura 3: Componentes fundamentais.

A difusão, realizada pela emissora, envolve um processo antes do envio da programação e aplicações ilustrado em 4. O multiplexador é a interface entre o estúdio e o transmissor da emissora, sendo responsável por multiplexar o fluxo de entrada (áudio, vídeo, aplicação) de acordo com a configuração do transmissor.

O fluxo de dados (aplicação) é tratado por um servidor responsável pelo chamado carrossel de dados. Além disso, este servidor recebe informações dos telespectadores via canal de retorno em aplicações interativas que necessitam da troca de dados com a emissora.

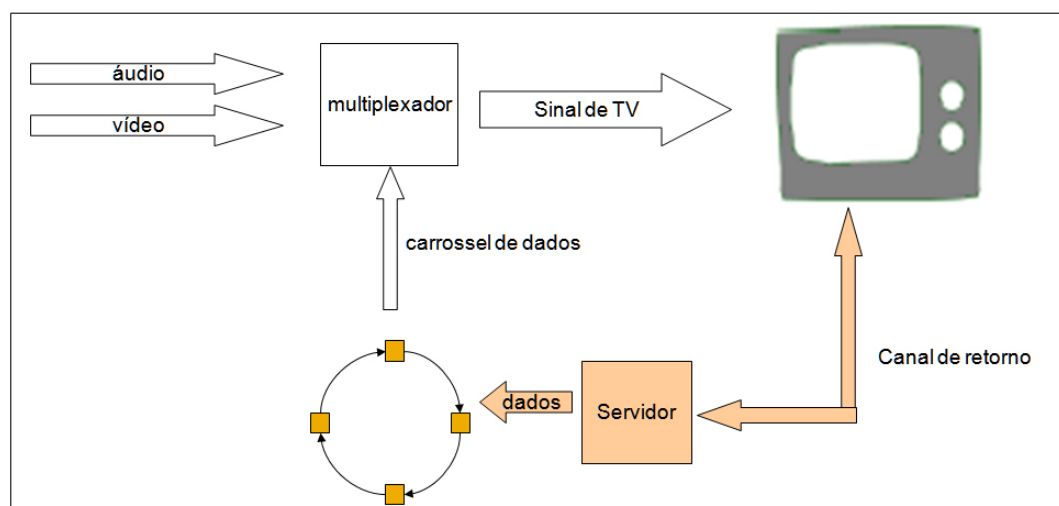


Figura 4: Processo de envio.

2.3 Arquitetura da TV Digital e do *middleware* GINGA

A arquitetura da TV Digital está dividida em diversas camadas (Modulação, Transporte, Compressão, Middleware e Aplicações). Esses módulos estão presentes na STB (*Set-Top-Box*), aparelho que é responsável pela recepção e decodificação de sinais digitais. Cada camada apresenta uma função bem definida, a qual utiliza os serviços prestados da camada inferior e dá suporte a camada superior.

Dentre essas camadas, daremos prioridade ao *middleware*. A função do *middleware* é tornar toda a arquitetura envolvida na STB transparente para as aplicações, de modo que aumente a portabilidade, isto é, qualquer aplicação poderá ser executada sem nenhum problema entre todas as STB que fazem uso dos mesmos *middlewares*. Dessa forma o Quiz tratado nesse projeto poderá ser executado em qualquer STB com o *middleware* nacional Ginga, apenas é necessário que a STB tenha canal de retorno, para a troca de informações.

Na camada de Middleware ilustrada em 5, no caso do Brasil, o sistema adotado é o GINGA que é dividido em 3 subsistemas principais, Ginga-NCL(Declarativo), Ginga-J(Procedural) e Ginga-CC(Common-Core).

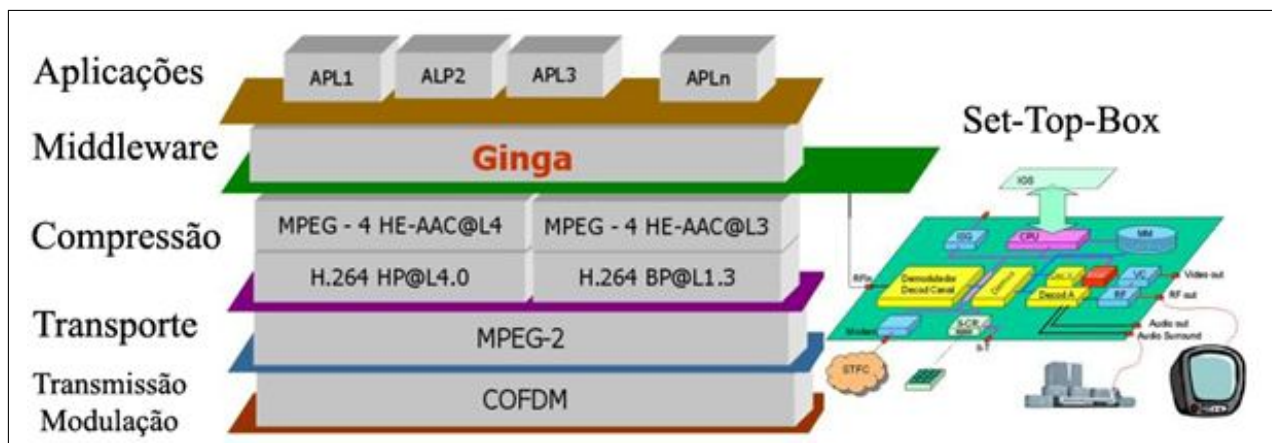


Figura 5: Arquitetura Set-Top-Box (Middleware GINGA).

O GINGA-CC é o núcleo do *middleware* e apresenta uma ponte que associa o GINGA-NCL ao GINGA-J. O primeiro trata-se de uma linguagem estrutural baseada em XML, desenvolvida pela PUC-RIO (Pontifícia Universidade do Rio de Janeiro) que é responsável pela apresentação de mídias, sem qualquer efeito interativo. Para essa funcionalidade relacionada a conteúdos interativos foi criado o GINGA-J, desenvolvido pela UFPB (Universidade Federal da Paraíba). O subsistema detém de um conjunto de APIs em Java que preenchem a falta de dinamismo deixado pelo NCL, ampliando o campo de possibilidades na produção de aplicações para TV Digital.

Porém, é importante frisar que o NCL tem, como suporte na produção de aplicações digitais, a linguagem de *script* LUA. Este permite que as aplicações em NCL também possam ter conteúdos interativos. A figura a seguir representa a estrutura básica do *middleware* GINGA.



Figura 6: Estrutura do GINGA.

3 Interação na TV Digital

Até a década de 90, a interatividade em TV Digital era pouco explorada. Porém com a chegada de novos equipamentos nas televisões por assinatura, pequena parcela da sociedade começou a usufruir das formas iniciais de interação com os telespectadores. Apesar disso, ainda sim essa característica era pouco valorizada, já que as pessoas procuravam os serviços pagos pela quantidade de canais oferecidos e a melhoria na qualidade de imagem e áudio [9].

Essa característica de interatividade herdada de outras culturas não foi muito valorizada no Brasil devido ao fato, também, do analfabetismo funcional estar presente em 68% da população brasileira [2005, IBOPE], sendo assim, isso afetava na interação com os novos serviços prestados pelas emissoras, fazendo com que as pessoas não conseguissem interagir devidamente com as informações hipertextuais disponibilizadas pelas emissoras.

É analisando também essas questões que a TV Digital, no âmbito nacional, busca tratar a interatividade de uma maneira mais satisfatória, seguindo seus novos padrões de modo a adequar as situações relevantes ao país em questão. Essas adequações vão desde os hardwares sendo estes acessíveis economicamente a grande parte da população brasileira, até aplicações desenvolvidas buscando a praticidade, dando ao usuário uma maior comodidade relacionada a usabilidade, evitando a implementação de aplicações inchadas com funções supérfluas, diferente do que é visto e utilizado hoje nos computadores.

Além desses fatores observados, há também trabalhos sendo desenvolvidos que buscam analisar as aplicações a fim de fornecer ao telespectador uma maior usabilidade, apresentado em [10]. No qual analisa as diversas formas de interação do telespectador em um determinado Quiz, observando as opiniões dos usuários de acordo com as maneiras de seleção e confirmação do voto via controle remoto.

3.1 Aplicações em TV Digital

As aplicações para TV Digital são semelhantes as de computadores, porém devido ao *middleware* nacional Ginga, há algumas diferenças quanto as linguagens utilizadas. Outra diferença está relacionada a difusão da aplicação, que é feita por meio de transmissão digital multiplexadas com o áudio, vídeo e legendas, sendo esta recebida e executada na STB do telespectador. Pelo fato das restrições do hardware utilizado, devemos levar em conta as limitações de aplicações comparadas as executados em computadores convencionais [9].

De acordo com o *middleware* Ginga podemos definir atualmente dois tipos de linguagens utilizadas para o desenvolvimento de aplicações, NCL e JAVA. Atualmente o subsistema responsável pelo NCL, Ginga-NCL, é o mais utilizado e se apresenta em um estágio mais evoluído. Esta é uma linguagem declarativa responsável pelo posicionamento das hipermídias¹ na tela, assim como as formas de execução delas. Já os aplicativos em Java são tratados pelo subsistema Ginga-J, porém apenas uma aplicação foi produzida comercialmente e até a corrente data ainda não estavam prontas as normas para testes de conformidade desse subsistema [11].

Em geral, podemos perceber a formação de um novo conceito em desenvolvimento de softwares, surgindo serviços diversificados como novas formas de propaganda, aplicações com acesso a serviços do governo, marcação de consultas, tele-aulas, entre outros. Serviços estes que podem ser tratados de forma muito melhor no meio televisivo do que em computadores.



Figura 7: Aplicações Possíveis: (a) Multi-camêra em partida de futebol. e (b) Compra de automóvel.

¹Reunião de várias mídias num suporte computacional, suportado por sistemas eletrônicos de comunicação, ou seja, algo que tenha interação com demais pessoas.

3.1.1 Exemplo de Quiz

Quiz é uma espécie de enquete com pergunta e respostas usado, nesse caso, pelas emissoras que têm como elemento fundamental a interação dos telespectadores. Através disso, as emissoras coletam dados dos telespectadores obtendo informações que são de grande utilidade para suas decisões, podendo assim realizar determinadas ações dependendo do contexto da aplicação.

Um exemplo de Quiz seria a utilização de um sistema de votação no qual os telespectadores selecionam o gol mais bonito de uma determinada rodada do campeonato. Essa enquete pode ser enviada ao telespectador pela emissora através do canal de difusão, em broadcast, no momento das repetições (replay) do lances, por exemplo, em um telejornal.



Figura 8: Exemplo de Quiz "Gol da Rodada".

4 *Linguagens*

Devido as restrições de hardware presentes nesse novo conceito de TV Digital, as linguagens adotadas e utilizadas hoje nesse contexto são linguagens que buscam praticidade e leveza. Essas, como mencionado anteriormente, são atualmente duas, Java e NCL.

No projeto foi utilizado apenas NCL em conjunto com *script* LUA, já que o NCL permite utilizar o *script* LUA em forma de mídia, sendo denominados objetos imperativos NCLua, como é mostrado nas próximas seções.

4.1 Nested Context Language (NCL)

A linguagem NCL foi criada pela PUC-RIO com o intuito de atender a demanda da nova tecnologia abordada em televisores. Esta é responsável pela apresentação de hipermídia interativa que dá pelas interconexões entre elementos denominados nó, que vão desde mídias comuns, como imagens e vídeos até contextos da própria linguagem como é feito com os *scripts* LUA [12].

NCL é fundamentado em XHTML, que é uma linguagem declarativa baseada em mídia, o que significa que a estrutura definida pelas relações entre os objetos XHTML é incorporada no conteúdo do documento de mídia. Assim, XHTML pode ser classificada como uma linguagem de marcação: um formalismo que descreve uma classe de documentos que emprega marcação para delinear a estrutura do documento, a aparência, entre outros aspectos [13].

O foco da linguagem declarativa NCL é mais amplo que o XHTML. A linguagem foca a sincronização espaço-temporal, adaptabilidade e o suporte para múltiplos dispositivos de exibição. Estes são definidos pelos elos, descritores e regiões respectivamente, ilustrado em 9. Um documento NCL define como as mídias serão unidas em uma apresentação multimídia, tratando a estrutura e a relação dos objetos de mídia no tempo e no espaço. Nela podemos ter objetos de imagem (GIF, JPEG, etc), objetos de vídeo (MPEG, MOV,

etc), objetos de áudio (MP3, WMA, etc), objetos de texto (TXT, PDF, etc) e objetos de execução (Lua, Xlet, etc). A determinação de quais objetos o NCL pode tratar e apresentar é definido pelo GINGA-CC, já citado anteriormente, e incorporados no player de execução.

4.1.1 Estrutura

A estrutura de um documento NCL segue um padrão XML, e é dividido dessa forma:

- Cabeçalho do arquivo NCL;
- Cabeçalho do programa;
- Corpo do programa;
- Término.

A figura 9 ilustra toda as partes presentes neste documento:

cabeçalho do arquivo NCL	1: <code><?xml version="1.0" encoding="ISO-8859-1"?></code> 2: <code><ncl id="exemplo01" xmlns="http://www.ncl.org.br/NCL3.0/EDTVProfile" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="http://www.ncl.org.br/NCL3.0/EDTVProfile http://www.ncl.org.br/NCL3.0/profiles/NCL30EDTV.xsd"></code>
cabeçalho do programa	3: <code><head></code>
base de regiões	4: <code><regionBase></code> 5: <code><!-- regiões da tela onde as mídias são apresentadas --></code> 6: <code></regionBase></code>
base de descritores	7: <code><descriptorBase></code> 8: <code><!-- descritores que definem como as mídias são apresentadas --></code> 9: <code></descriptorBase></code>
base de conectores	10: <code><connectorBase></code> 11: <code><!-- conectores que definem como os elos são ativados e o que eles disparam --></code> 12: <code></connectorBase></code>
	13: <code></head></code>
corpo do programa	14: <code><body></code>
ponto de entrada no programa	15: <code><port id="plnicio" component="ncPrincipal" interface="ilnicio"/></code>
conteúdo do programa	16: <code><!-- contextos, nós de mídia e suas âncoras, elos e outros elementos --></code>
	17: <code></body></code>
término	18: <code></ncl></code>

Figura 9: Estrutura documento NCL.

Podemos perceber que o documento apresenta diversos elementos, alguns localizadas no cabeçalho delimitadas por `<head>` e algumas presentes no corpo do documento delimitadas por `<body>`.

Na região do cabeçalho podemos identificar três tipos de divisões: `<regionBase>`, `<descriptorBase>` e `<connectorBase>`. Estes componentes são responsáveis por definir onde e como o conteúdo deve ser exibido.

Já o corpo (elemento `<body>`) descreve o que é o conteúdo a ser exibido e quando isso deve ser feito.

Através dos objetos de mídia, definidos pelo elemento `<media>` e denominados nós de mídia em NCL, define-se o que executar, ou seja, qual conteúdo será exibido. Os nós de mídia são definidos dentro de um outro tipo de nó, o nó de composição ou contexto (elemento `<context>`). Os nós de contexto têm a função de encapsular outros nós, podendo ser tanto nós de mídia como nós de composição, tratando-se de uma recursividade. Sendo assim, para acessar um nó interno a um contexto, é necessário definir nesse contexto um ponto de entrada, sendo este rotulado como elemento `<port>`.

Os nós de mídias não apresentam, *a priori*, o local onde eles devem ser exibidos, isso é definido pelos elementos `<region>` presentes no cabeçalho, que define uma área da tela onde uma mídia será apresentada. Essa separação torna-se importante pelo fato de outras mídias poderem utilizar a mesma região, não prendendo assim uma região a uma determinada mídia. As regiões são definidas por seus atributos relacionados a posição (*bottom*, *left* e *right*), sua altura e largura (*height* e *width*). Toda região apresenta um identificador (atributo *id*), usado pelo descritor para associá-lo a uma ou mais mídias.

Sendo assim o descritor (elemento `<descriptor>`) é o responsável por associar uma mídia a uma determinada região, e este define como os nós de mídia são inicialmente apresentados [14].

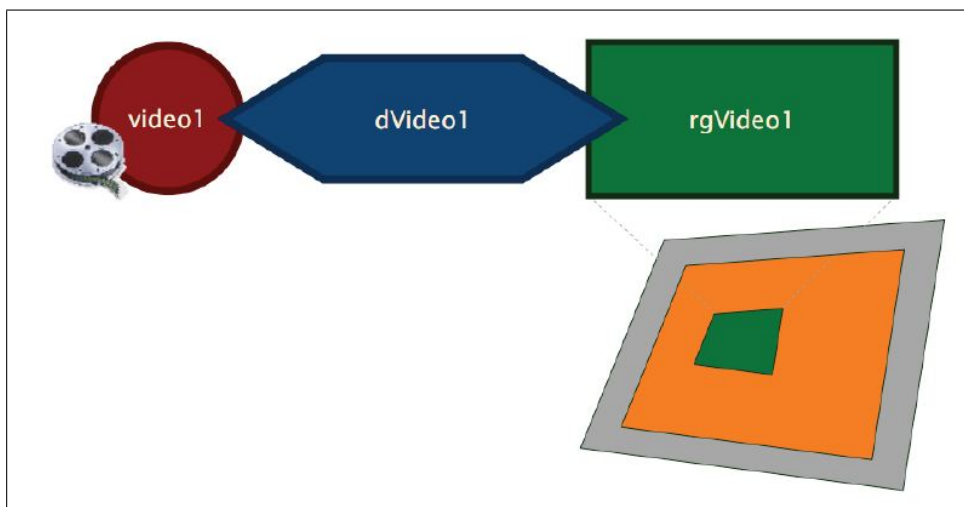


Figura 10: Associação entre nó de mídia ("video1"), descritor ("dVideo1") e região ("rgVideo1").

Depois de definido o que tocar, onde tocar e como tocar, é especificado quando tocar os diferentes nós. Sendo assim, a máquina é responsável por apresentar o documento NCL partindo da execução da mídia (nó) apontada pela porta no documento.

4.2 NCLua

Desde o início dos anos 90, LUA sempre foi utilizado em conjunto com outras linguagens, pois LUA faz com que as aplicações possam ser estendidas ou adaptadas através do uso de seus *scripts*.

LUA é uma linguagem de fácil aprendizado, pois combina sintaxe procedural com declarativa apresentando poucos comandos primitivos. Além destas qualidades apresenta um alto grau de portabilidade, o que favorece em aplicações para TV Digital, já que a tecnologia de TVDI é muito utilizada em dispositivos móveis.

Foi a partir dessa e de outras características que LUA foi escolhida para integrar a linguagem NCL, mas para isso foi preciso produzir um módulo específico com o papel de realizar a comunicação entre o *script* LUA e o documento NCL. Este módulo foi denominado NCLua que apresenta novas funcionalidades, o que permite, por exemplo, a resposta da aplicação a alguma ação efetuada pelo usuário, como o apertado de uma tecla.

O NCLua diferencia de um programa LUA puro, pois o NCLua é controlado pelo documento NCL no qual está inserido. Assim, além de toda a funcionalidade que o LUA dispõe através de sua biblioteca nativa, os seguintes módulos estão disponíveis para *scripts* NCLua:

- Módulo *event*: permite que objetos NCLua se comuniquem com o documento NCL e outras entidades externas (controle remoto e canal de interatividade);
- Módulo *canvas*: oferece funcionalidades que permitem desenhar objetos gráficos na região definida para o NCLua. Nesse trabalho foi utilizado para desenhar a pergunta e as respostas na região definida para a aplicação na TV do telespectador;
- Módulo *settings*: oferece acesso às variáveis definidas no objeto *settings* do documento NCL (objeto do tipo "*application/x-ncl-settings*");
- Módulo *persistent*: exporta uma tabela com variáveis persistentes entre execuções de objetos imperativos.

5 *Ambiente Completo de Execução do Quiz*

Atualmente a televisão é um dos meios mais importantes na difusão da informação na sociedade, representando uma forte ferramenta na interação com os telespectadores de modo a trocar informações importantes, seja para o autobenefício ou para a comercialização de produtos. Nesse novo cenário de TVDI (Televisão Digital e Interativa) a maneira de se trabalhar com as aplicações para esse contexto é um pouco diferente no que diz respeito a difusão e coleta de dados.

Assim, de acordo com a arquitetura de transmissão ilustrada na figura 3 foi criado todo um ambiente para a elaboração do Quiz por meio de uma interface simples, o que facilita a rapidez da emissora em produzir a aplicação Quiz sem ter conhecimento nas linguagens adotadas pelo *middleware* brasileiro.

Além disso, é abordado a coleta dos dados feita pelo PSI (Provedor de Serviços Interativos) com base nos XML's recebidos dos telespectadores que efetuaram o voto, bem como a consulta de dados no PSI (Provedor de Serviços Interativos) feita pela emissora, sendo que a coleta e consulta são realizadas com a utilização de um Web Service.

Sendo assim, o ambiente é dividido em três módulos:

- Módulo Gerador do Quiz;
- Módulo Aplicação Quiz;
- Módulo de Análise e Consulta.

A figura 11 ilustra todo o ambiente:

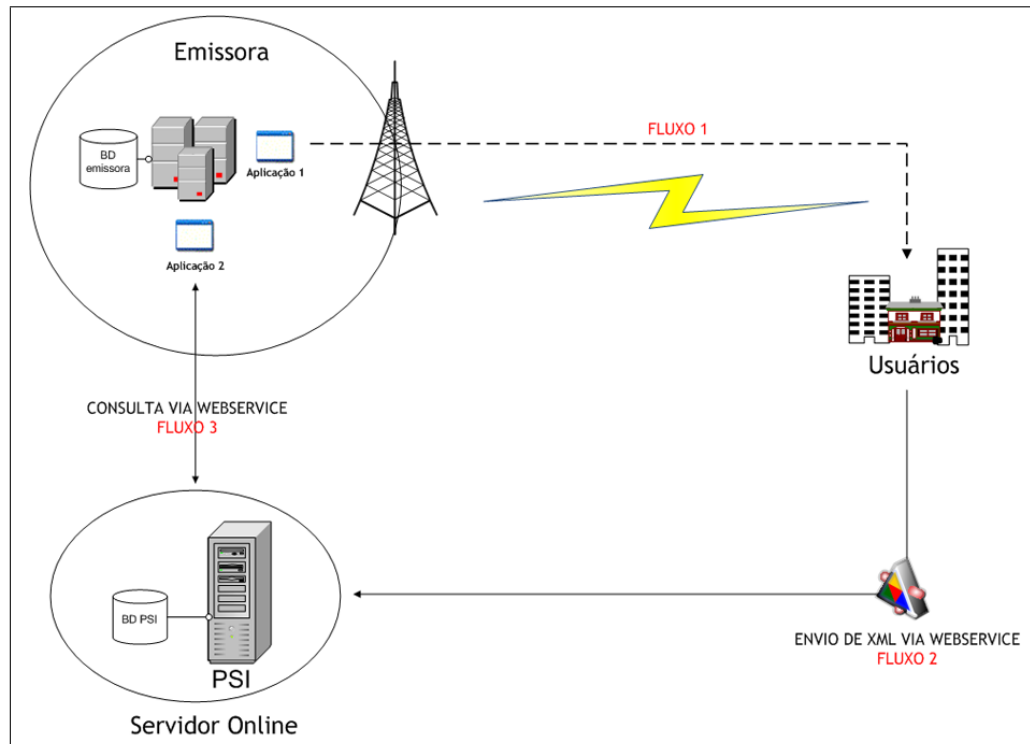


Figura 11: Arquitetura do Projeto.

5.1 Módulo Gerador do Quiz

Este módulo é responsável pelo desenvolvimento de todo o código que será enviado via FLUXO 1 para o telespectador, conforme destacado na figura 12.

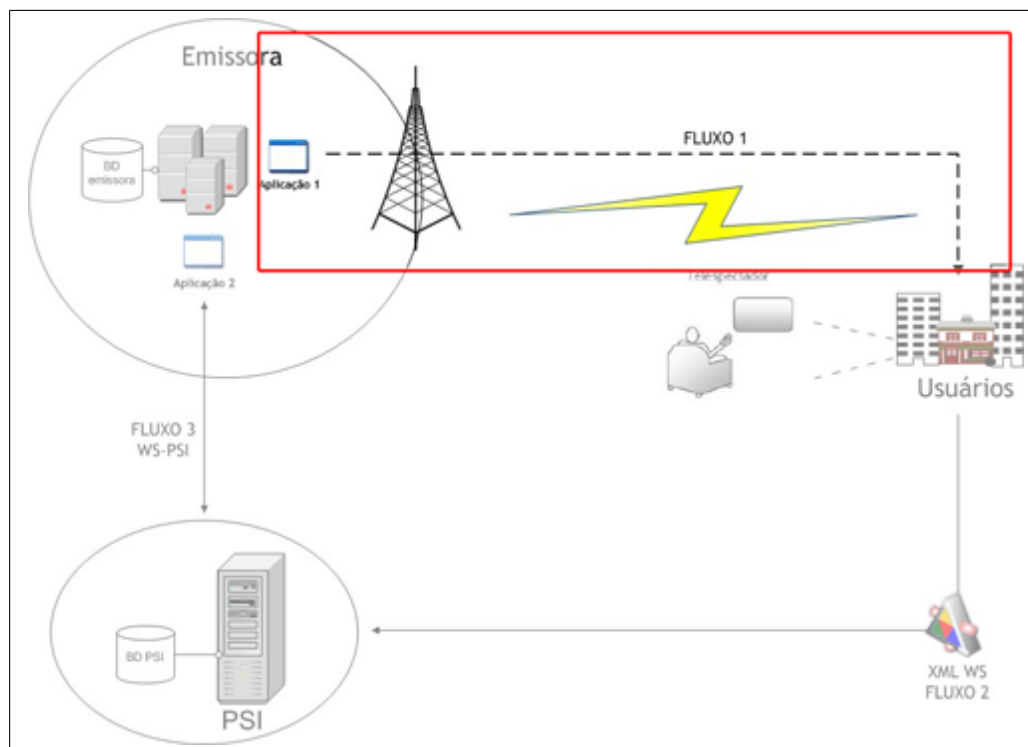


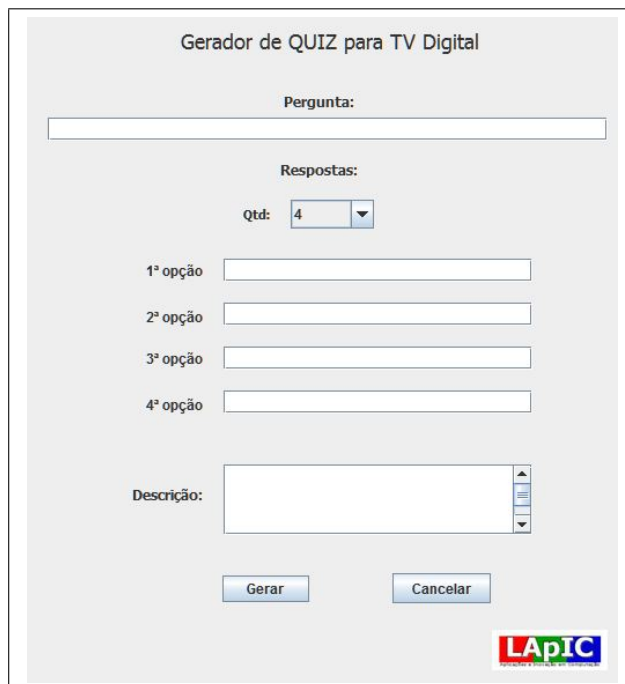
Figura 12: Módulo Gerador do Quiz.

A APLICAÇÃO 1, citada na figura 12, é responsável por gerar os arquivos necessários para a execução do aplicativo na STB do telespectador e por guardar as informações do Quiz em um banco de dados da emissora destinado a essa função, definindo um identificador (id) para este Quiz para referenciá-lo no banco de dados do PSI. Estes arquivos são compostos por *scripts* LUA e documento NCL. Com a associação dessas duas linguagens podemos exercer todas as funcionalidades necessárias para o funcionamento correto de nossa aplicação.

Entre os arquivos gerados pela APLICAÇÃO 1 temos:

- Módulo NCLua SOAP para comunicação com Web Service no PSI [15];
- *script* LUA principal responsável pela interface da aplicação (Anexo A);
- Documento NCL responsável por tratar o *script* LUA principal (Anexo B).

A APLICAÇÃO 1 é feita em Java apresentando o seguinte layout:



The image shows a Java Swing window titled "Gerador de QUIZ para TV Digital". The window contains the following elements:

- A text input field labeled "Pergunta:".
- A section labeled "Respostas:" containing a dropdown menu labeled "Qtd:" with the value "4" selected.
- Four text input fields labeled "1ª opção", "2ª opção", "3ª opção", and "4ª opção" stacked vertically.
- A text area labeled "Descrição:" with a vertical scrollbar on the right side.
- Two buttons at the bottom: "Gerar" and "Cancelar".
- A logo for "LApIC" in the bottom right corner.

Figura 13: Layout da aplicação geradora.

Com esta aplicação (figura 13) é definido, a pergunta do Quiz, a quantidade de opções e os textos correspondentes a cada opção, bem como uma breve descrição sobre o Quiz.

Posteriormente os dados da aplicação são encaminhados para um servidor definido pela emissora, que é responsável pela transmissão dos dados necessários para a execução do Quiz em conjunto com a programação da emissora através de um carrossel de dados. Essa transmissão é feita em *broadcast*, por uma antena, a todos os telespectadores. Porém como não contamos atualmente com um playout (equipamento para a geração do sinal), não foi testado nesse projeto a questão de transmissão, sendo esta simulada através da cópia direta dos arquivos na STB.

5.1.1 Base de Dados da Emissora

A base de dados da emissora é formada por apenas uma tabela, responsável por guardar as informações do Quiz atribuindo um identificador (id), sendo estas informações utilizadas pela APLICAÇÃO 2 para consultas relacionadas ao módulo de análise e consulta (seção 5.3) para obter as perguntas criadas até a corrente consulta.

enquete	
quiz_ID	INT(11)
pergunta	VARCHAR(40)
opcao1	VARCHAR(40)
opcao2	VARCHAR(40)
opcao3	VARCHAR(40)
opcao4	VARCHAR(40)
descricao	VARCHAR(100)
Indexes	

Figura 14: Tabela Enquete.

5.2 Módulo Aplicação Quiz

Este módulo corresponde a aplicação Quiz onde ocorre a interação com o usuário, na qual o telespectador irá realizar sua escolha pressionando no controle remoto o número correspondente a opção desejada, destacado na figura 15. Porém existem diversas técnicas sendo abordadas a fim de aumentar a usabilidade da aplicação [10].

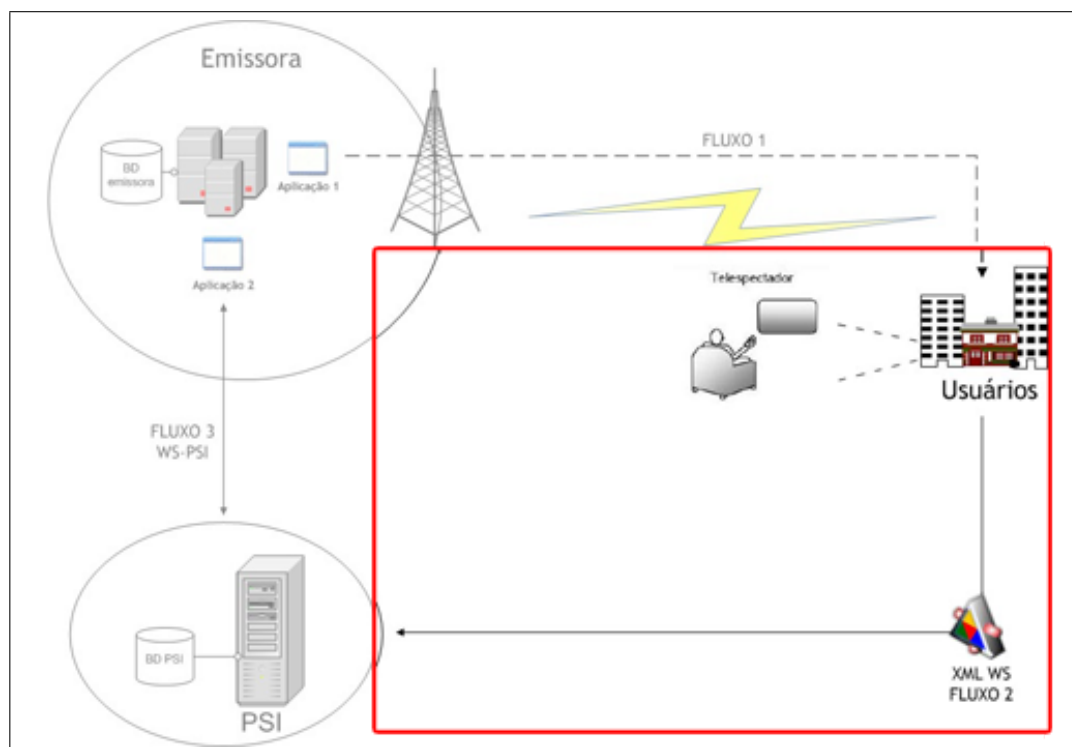


Figura 15: Módulo Aplicação Quiz.

Como citado na seção anterior, uma série de *scripts* LUA e um arquivo NCL são necessários para a execução desta aplicação.

Neste contexto o arquivo NCL (Anexo B) é padrão para qualquer Quiz gerado, sendo este apenas responsável por tratar a mídia NCLua, isto é, o NCL define uma região e seu descritor para o *script* LUA principal mostrar a pergunta e as respostas através da função *drawText* presente no módulo *canvas* citado em 4.2.

O módulo NCLua SOAP [15] trata-se de um conjunto de *scripts* LUA que são responsáveis pela conexão da STB do telespectador ao PSI via protocolo SOAP onde as informações trocadas se baseiam em XML e são enviadas via HTTP, o que facilita a comunicação possibilitando a heterogeneidade dos sistemas envolvidos.

O *script* principal é o único que sofre alterações quando criado um novo Quiz, pois é nele que ficam as informações sobre a pergunta e as opções de respostas definidas pela APLICAÇÃO 1. Nele também é tratado os eventos realizados pelo telespectador via controle remoto, isto é, de acordo com a opção selecionada pelo telespectador um diferente XML (figura 16) é encaminhado ao PSI (FLUXO 2) contendo informações importantes para a emissora. Além disso, é esse *script* que faz uso do módulo NCLua SOAP, acessando o método do Web Service presente no PSI. Para isso, a conexão ao PSI é definida neste *script* principal (Anexo A) onde é informado o IP, método e o parâmetro a ser consumido. O parâmetro trata-se do XML de resposta ilustrado em 16.

```
<?xml version="1.0" encoding="utf-8"?>
<quizstb>
  <head>
    <device serialnumber="123456xx"/>
  </head>
  <quiz id="607">
    <option value="0"/>
    <location zip="13566770"/>
    <user birth="01-02-1972" genre="male">Marco Aurelio Freesz Junior</user>
  </quiz>
</quizstb>
```

Figura 16: XML de resposta.

Neste XML podemos identificar: Número do dispositivo do usuário, identificação (id) correspondente ao Quiz, opção selecionada pelo usuário, além das informações pessoais: CEP, data de nascimento, sexo e nome. Essas informações são usadas pela emissora para realizar não só consultas relacionadas a quantidade de votos, como também levantar perfis dos telespectadores.

5.3 Módulo de Análise e Consulta

Este módulo é responsável pela interação emissora x PSI. Essa interação é feita através do acesso ao Web Service presente no PSI que é responsável por servir a APLICAÇÃO 2, conforme destacado na figura 17.

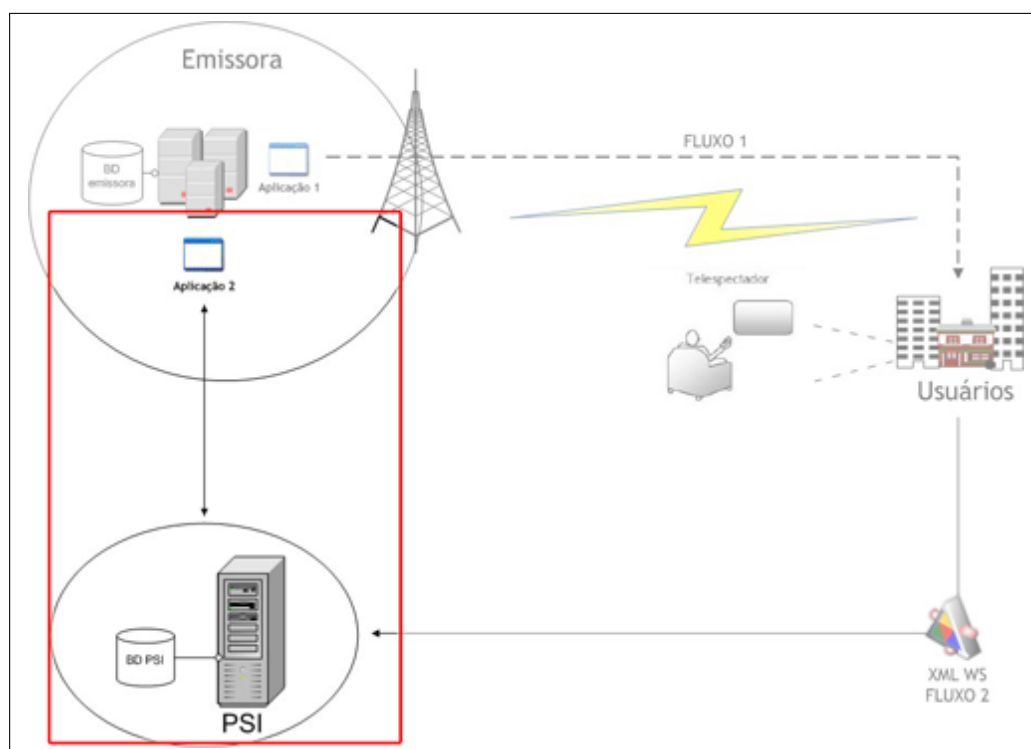


Figura 17: Módulo de Análise e Consulta.

Nesta interação a APLICAÇÃO 2 (figura 17) acessa, com utilização do protocolo SOAP, a base de dados do PSI, que contém informações das enquetes que já receberam votos. Essas informações são obtidas e salvas no banco de dados do PSI através do tratamento do XML recebido das STB's, após cada voto realizado pelos usuários.

Assim, na APLICAÇÃO 2 a emissora escolhe o Quiz que deseja realizar a consulta sendo exibido as opções correspondentes ao Quiz seguidas de suas porcentagens de votos.

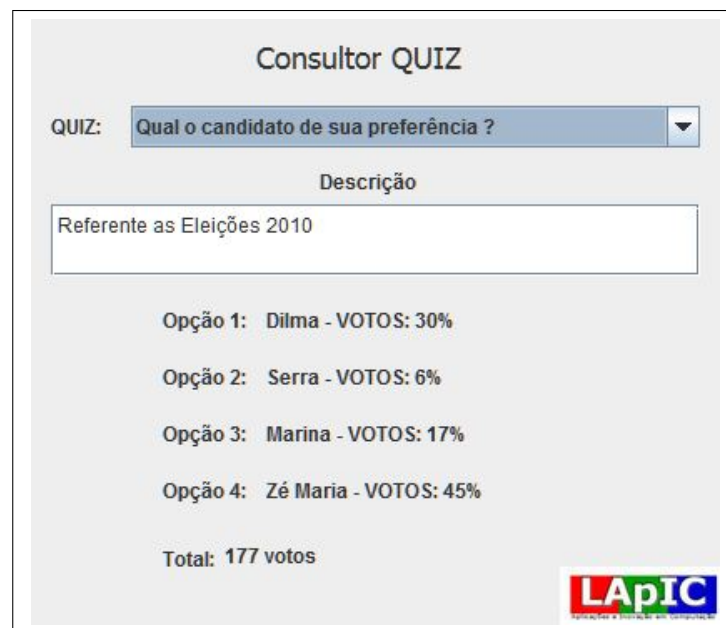


Figura 18: Consultor sobre a quantidade de votos.

Além disso, é possível na APLICAÇÃO 2 fazer determinadas análises quanto a filtragem dos votos, agrupando os votos de acordo com o perfil do usuário, relacionando-os em região, sexo e faixa etária.

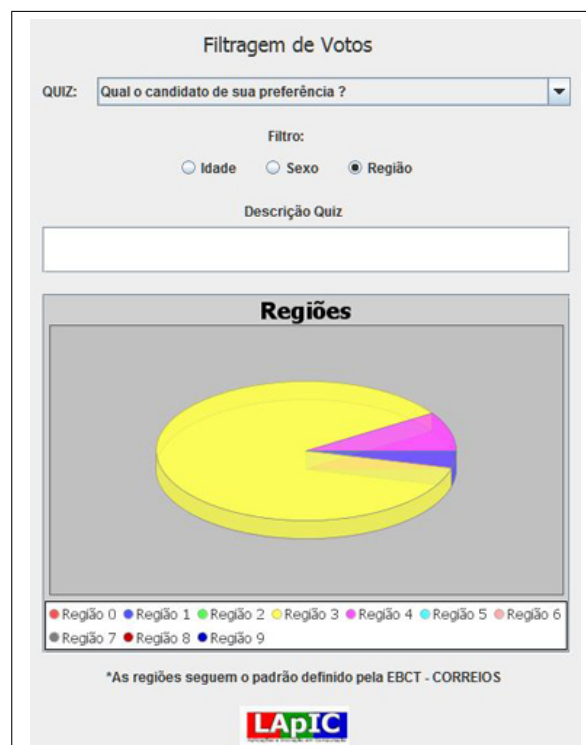


Figura 19: Filtragem por Regiões.

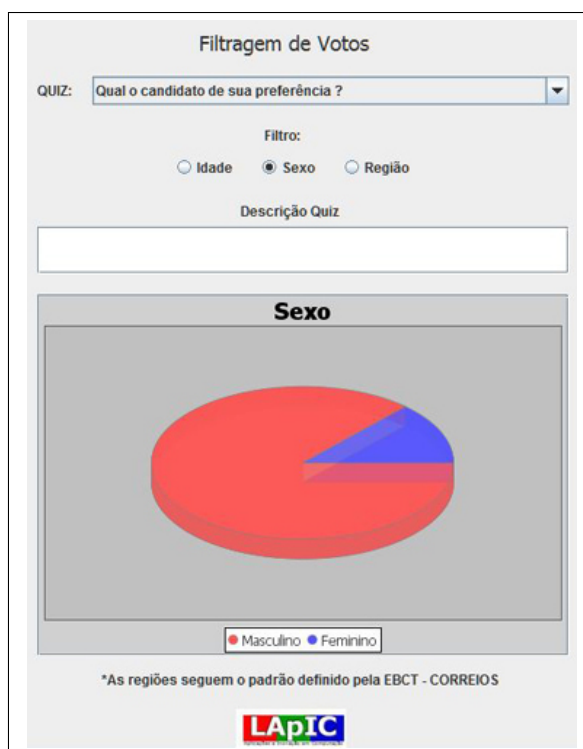


Figura 20: Filtragem por Sexo.



Figura 21: Filtragem por faixa etária.

Essa aplicação exemplifica um dos grandes objetivos de uma emissora, que é a busca de informação sobre os telespectadores, para poder oferecer conteúdos ou divulgar produtos conforme o perfil de cada um.

Com isso, essa aplicação apenas mostra uma das diversas pesquisas possíveis nesse novo contexto de TVDI, divulgando e coletando os dados por um meio tão difundido quanto a Internet.

5.3.1 Base de Dados do PSI

O banco de dados do PSI conta apenas com três tabelas. A tabela *enquete* guarda informações das enquetes que já receberam votos, isto é, quando uma enquete é criada no lado da emissora ela ainda é ausente na base de dados do PSI até que ocorra um primeiro voto. Ocorrendo o primeiro voto o registro dessa enquete é criado com uma identificação (id) idêntica ao registro presente na base de dados da emissora.

A tabela *usuario_enquete* é responsável por guardar cada evento de voto recebido, definindo o usuário, enquete e opção que o telespectador votou, assim como a hora do voto. O PSI ao receber o voto verifica, também, se o usuário já se encontra em sua base de dados, em caso negativo o usuário é criado com as informações contidas no XML de resposta do telespectador, sendo a tabela usuários responsável por guardar os dados dos telespectadores.

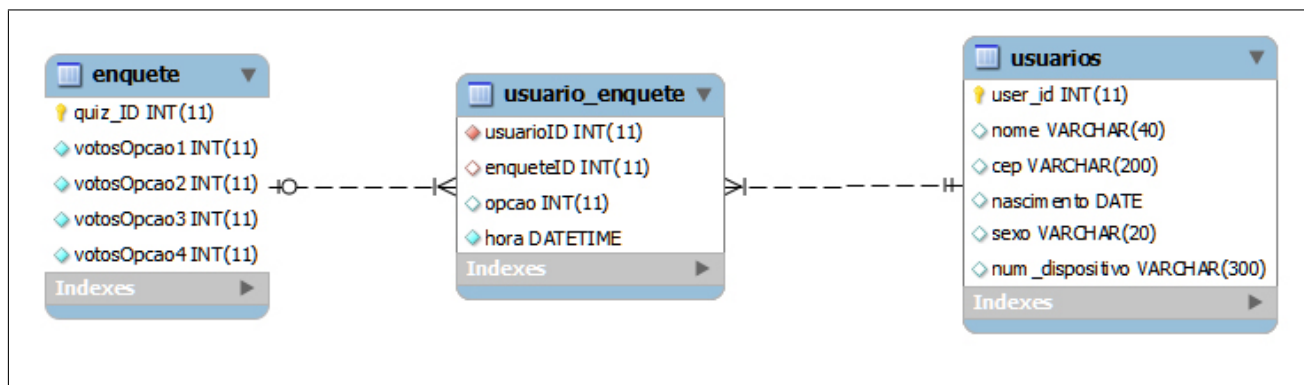


Figura 22: Base de dados do PSI.

Neste trabalho o PSI foi tratado por uma máquina virtual a fim de testar toda a aplicação.

6 *Considerações Finais*

Fazer compras, realizar votos, marcar consulta médica são algumas das diversas possibilidades de aplicações para TV Digital, o que dá ao telespectador comodidade, interagindo através do seu controle remoto. Porém ainda é um grande desafio garantir a interação segura e privativa entre as partes envolvidas, já que os estudos ainda não estão totalmente voltados para a segurança e sim para o desenvolvimento de aplicações.

Sendo assim, podemos perceber que esta é uma tecnologia que ainda está em fase de desenvolvimento e por isso exige muitos cuidados quanto as aplicações desenvolvidas.

6.1 Conclusão

Esse trabalho mostrou como pode ser feito a divulgação e coleta de informações por parte da emissora, de acordo com sua proposta. Para o trabalho, foi feito o uso de um Web Service localizado no PSI, por ser mais vantajoso, já que trata-se de uma arquitetura descentralizada, onde qualquer emissora poderá fazer uso deste provedor para gerenciar os dados referentes as enquetes produzidas. Lembrando que, para a utilização do PSI por parte de outra emissora, seria necessário a inserção de um novo campo na base de dados do PSI (figura 22) para referenciar a qual emissora um determinado Quiz está ligado. Assim, seria necessário também, que a consulta ao PSI fosse feita restringindo o acesso ao Quiz das outras emissoras, isto é, fazer o uso do novo campo que define a emissora produtora do Quiz.

O trabalho aqui desenvolvido, define um ambiente apropriado para criação de um Quiz de maneira prática, não precisando conhecer sobre as linguagens envolvidas em uma aplicação para TVDI no Brasil. Além da criação do Quiz é feito a coleta dos resultados quantitativos dos votos, bem como uma breve análise dos perfis dos usuários. Fato este que pode ser de grande importância para uma emissora no que diz respeito a divulgação de produtos de acordo com a características dos telespectadores.

6.2 Trabalhos Futuros

A idéia é tornar o teste mais realista, pois até o presente momento o sinal digital e toda a infra-estrutura necessária para a difusão da informação não estava disponível no ambiente de teste, sendo assim, seria interessante realizar a simulação com a utilização de um servidor responsável pelo carrossel de dados conforme ilustrado em 4, fazendo com que toda informação seja enviada ao telespectador de forma correta.

Outra melhoria seria a criação de um novo módulo responsável pela divulgação pública do resultado de um determinado Quiz, já que neste trabalho o resultado é acessado apenas pela emissora criadora do Quiz.

Além disso, com base no estudo em [10] é possível fazer uma aplicação geradora considerando o tipo de interação, ou seja, a emissora poderá criar o Quiz definindo como será feito o voto por parte do telespectador, isto é, definir a maneira de se utilizar o controle remoto para efetuar o voto. Neste trabalho a forma adotada foi a numérica, onde o usuário aperta o número correspondente a opção do Quiz. Assim, em [10] se encontra alguns tipos de interação que podem ser adotados a fim de divulgar aplicativos de acordo com a facilidade de uso do público alvo.

Referências

- 1 MONTEZ, C.; BECKER, V. *TV Digital Interativa: Conceitos, Desafios e Perspectivas para o Brasil*. 2nd. ed. Florianópolis SC: Editora da UFSC, 2005.
- 2 PICCOLO, I.; BARANAUSKAS, M. Desafios de design para a tv digital interativa. *IHC - VII Simpósio Brasileiro sobre fatores humanos em sistemas computacionais*, p. 10, 2006.
- 3 MARQUES, V. C. Graduação, *Desenvolvimento de uma aplicação para TV Digital utilizando o Middleware nacional Ginga-NCL voltado ao aprendizado de crianças*. Pelotas: [s.n.], 2008. 42 f.
- 4 DTV. *História do SBTVD*. 2010. Disponível em: <http://www.dtv.org.br/index.php/entenda-a-tv-digital/historia-do-sbtvd/>. Acesso em: 04 de Novembro de 2010.
- 5 CPQD. *Histórico TV Digital*. 2010. Disponível em: http://www.cpqd.com.br/img/historico_tv_digital.pdf. Acesso em: 04 de Novembro de 2010.
- 6 Santos Junior J. B. et al. A platform for diffusion interactive multimedia content: An approach focused on iptv system and broadcasting digital television system. *EATIS 2008 - Euro American Conference on Telematics and Information Systems*, 2008.
- 7 Ginga. *Ginga site*. 2010. Disponível em: <http://www.ginga.org.br/>. Acesso em: 04 de Novembro de 2010.
- 8 MENDES, L. Uma visão sobre a tv digital no brasil. *T&C Amazônia*, p. 48–59, 2007.
- 9 Jucá, Paulyne; Lucena, Ubirajara. Experiências no desenvolvimento de aplicações para televisão digital interativa. *TVDI'2005*, 2005.
- 10 Costa, João; Lemos, Alexandre; Stein, Mônica. Avaliação de usabilidade em aplicativos interativos para a tvdigital terrestre. *Web Media '10*, 2010.
- 11 UOL. *Entrevista*. 2010. Disponível em: <http://idgnow.uol.com.br/blog/circuito/2010/07/29/criador-do-ginga-anda-preocupado-com-os-rumos-de-sua-criacao/>. Acesso em: 04 de Novembro de 2010.
- 12 MELO, J. C. P.; ARAÚJO, R. M. Relatório Estágio Supervisionado, *Os módulos NCL e NCLua do middleware GINGA para aplicações em TV Digital Interativa*. Natal: [s.n.], 2008. 74 p.
- 13 SOARES, L. F. G; RODRIGUES, Rogério Ferreira, M. Ginga-ncl: the declarative environment of the brazilian digital tv system. *Journal of the Brazilian Computer Society*, p. 37–46, 2007.

14 NETO C. S. ; AZEVEDO, R. G. A. . B. S. D. J. F. S. . S. *Capítulo 1 - Desenvolvimento de Aplicações Híbridas para TV Digital Interativa no Middleware Ginga*. Fortaleza,Ceará: [s.n.], 2009. 1-40 p.

15 Manoel Campos. *NCLua SOAP*. 2010. Disponível em: <http://manoelcampos.com/tvd/nclua-soap/>. Acesso em: 04 de Novembro de 2010.

ANEXO A – Exemplo do script LUA principal do projeto

```

require "ncluassoap"
function sendWebService(voteOption, serialNumber, locZip, birth, userGenre, userName, idQuiz, ipServer)
  --Retirado da página http://mancelcampos.com/tvd/nclua-soap/, utilizando conhecimentos fornecidos
  --por Manoel Campos
  b = string.len("<?xml version='1.0' encoding='utf-8'?>\r\n" .. "<quizstb>\r\n" .. "<head>\r\n" .. "<tvserver
ip=" .. ipServer .. "'>\r\n" .. "<device serialnumber=" .. serialNumber .. "'>\r\n" .. "</head>\r\n"
  .. "<quiz id=" .. idQuiz .. "'>\r\n" .. "<option value=" .. voteOption .. "'>\r\n" .. "<location zip=" ..
locZip .. "'>\r\n" .. "<user birth=" .. birth .. "' genre=" .. userGenre .. "'>" .. userName .. "</user>\r\n"
  .. "</quiz>\r\n" .. "</quizstb>\r\n")
  texto = ""
  for i=1,b do
    a = string.byte("<?xml version='1.0' encoding='utf-8'?>\r\n" .. "<quizstb>\r\n" .. "<head>\r\n" ..
"<tvserver ip=" .. ipServer .. "'>\r\n" .. "<device serialnumber=" .. serialNumber .. "'>\r\n"
  .. "</head>\r\n" .. "<quiz id=" .. idQuiz .. "'>\r\n" .. "<option value=" .. voteOption .. "'>\r\n" ..
"<location zip=" .. locZip .. "'>\r\n" .. "<user birth=" .. birth .. "' genre=" .. userGenre .. "'>"
  .. userName .. "</user>\r\n" .. "</quiz>\r\n" .. "</quizstb>\r\n",i,i);
    texto = texto .. a .. '-'
  end
  local msgTable = {
    address = "http://10.1.1.2/MyWebService/MyWsService", --Endereco do Web Service
    namespace = "http://lapic/", --Namespace, encontrado no arquivo WSDL referente ao web service
    operationName = "computarVoto", --Nome do metodo a ser acessado
    --Abaixo, os parametros do metodo a ser acessado
    params = {
      xml = texto
    }
  }
  ncluassoap.call(msgTable, getResponse, "1.1", "80", true)
  print("-----Chamou ncluassoap.call")
end
function showVote()
  local cw, ch = canvas:attrSize()
  canvas:attrColor(255, 255, 255, 255)
  canvas:drawRect("fill", 0, 0, cw, ch)
  canvas:attrColor(0, 0, 0, 255)
  canvas:attrFont("vera", 18)
  --Titulo
  canvas:drawText(30,20,'Qual o melhor candidato ?')
  --opcao 1
  canvas:drawText(30,40,'1 - Dilma')
  --opcao 2
  canvas:drawText(30,60,'2 - Serra')
  --opcao 3
  canvas:drawText(30,80,'3 - José Maria')
  --opcao 4
  canvas:drawText(30,100,'4 - Ricardo')
  canvas:flush()
end

```

Figura 23: *script* LUA principal - parte 1.

```
function handler(evt)
  if (evt.class == 'key' and evt.type == 'press') then
    print("key:", evt.key)
    local ok = false
    --Condicional para ver qual tecla foi apertada pelo usuario.
    if evt.key == "1" then
      ok = true
      sendWebService("1", "123456xx", "13566770", "18-03-88", "Masculino", "Marco Aurelio Freesz", "617")
    elseif evt.key == "2" then
      ok = true
      sendWebService("2", "123456xx", "13566770", "18-03-88", "Masculino", "Marco Aurelio Freesz", "617")
    elseif evt.key == "4" then
      ok = true
      sendWebService("4", "123456xx", "13566770", "18-03-88", "Masculino", "Marco Aurelio Freesz", "617")
    elseif evt.key == "3" then
      ok = true
      sendWebService("3", "123456xx", "13566770", "18-03-88", "Masculino", "Marco Aurelio Freesz", "617")
    end
    if ok then
      showVote()
    end
  end
  if (evt.class == "ncl" and evt.type=="presentation"and evt.action == "start") == false then
    return
  end
  showVote()
end
event.register(handler)
```

Figura 24: *script* LUA principal - parte 2.

ANEXO B – Exemplo do documento NCL do projeto

```

<?xml version="1.0" encoding="ISO-8859-1"?>
<ncl id="main" xmlns="http://www.ncl.org.br/NCL3.0/EDTVProfile">
  <head>
    <regionBase>
      <region id="rgLua" width="100%" height="100%" zIndex="1">
      </region>
    </regionBase>

    <descriptorBase>
      <descriptor id="dLua" region="rgLua" focusIndex="luaIdx"/>
    </descriptorBase>

    <connectorBase>
      <causalConnector id="onEndStart">
        <simpleCondition role="onEnd"/>
        <simpleAction role="start"/>
      </causalConnector>

      <causalConnector id="onBeginStop">
        <simpleCondition role="onBegin"/>
        <simpleAction role="stop"/>
      </causalConnector>
    </connectorBase>
  </head>
  <body>
    <port id="pLua" component="lua"/>

    <media type="application/x-ginga-settings" id="programSettings">
      <property name="service.currentKeyMaster" value="luaIdx"/>
    </media>

    <media id="lua" descriptor="dLua" src="lua/principal.lua">
    </media>

  </body>
</ncl>

```

Figura 25: Documento NCL.