



INSTITUTO SUPERIOR TÉCNICO
Universidade Técnica de Lisboa

WebC-iTV

Internet TV in Websites

Nuno Miguel Lucas da Silva Magalhães

Dissertação para obtenção do grau de Mestre em
Engenharia Informática e de Computadores

Júri

Presidente: Prof. João António Madeiras Pereira

Orientador: Prof. Alberto Manuel Rodrigues da Silva

Vogal: Prof. José Luís Brinquete Borbinha

Outubro 2010

Acknowledgments

I would like to express my gratefulness to those who helped me directly during the development of this dissertation.

Firstly, I'd like to fully acknowledge my supervisor Professor Alberto Silva for his accompaniment during this whole year. His dedication and patience was crucial for the gradual development of this project both by being available anytime in the year, even during scholar holidays, and by monitoring in regular basis the growth of the work giving pertinent advices and aiding in the development steps.

I thank Researcher and Engineer João Saraiva for his availability in solving many questions that arose throughout the year which allowed me to have a greater comprehension of many topics.

I also express my thanks to Researcher and Engineer David Ferreira for his objective advices and ideas which contribute for the enrichment of the solution.

Lastly, I thank my colleague Pedro Pinto for sharing his knowledge with me from the beginning until the end of this project.

Odivelas, October 13, 2010

Nuno Magalhães

Resumo

O grande desenvolvimento e aceitação mundial da Internet levou as empresas ao redor do globo, numa grande maioria, a depender desta tecnologia de acesso universal para fazerem face aos paradigmas que a sociedade hodierna apresenta.

Um exemplo vívido é o grupo de empresas de canais de televisão. Anteriormente, divulgavam o conteúdo televisivo apenas por aparelhos analógicos como é o caso das televisões pessoais com sinal por antena. Posteriormente, com a tal evolução na tecnologia da Internet, os canais começaram também a ser transmitidos pela Internet. É neste âmbito que surge a televisão na Internet (iTV) que permite qualquer utilizador com acesso à web visualizar um canal em qualquer local sem a restrição de ter de possuir uma antena e uma televisão.

O documento presente consiste numa Dissertação para o grau de Mestre em Engenharia Informática e de Computadores na Faculdade do Instituto Superior Técnico, sediado na Alameda, da Universidade Técnica de Lisboa.

Esta dissertação pretende a construção duma *framework* dentro do paradigma da televisão na Internet. Esta terá como público alvo produtores de canais de televisão que poderão criar canais e gerir os respectivos recursos de modo a, posteriormente, poderem ser transmitidos por via Web em sinal digital.

Serão abordados vários sistemas existentes que usam este paradigma de iTV. Estudar-se-ão o WorldTV e o WoMa que são sistemas voltados exclusivamente para a criação de canais de televisão pelo público em geral e, finalmente, estudar-se-á o caso mais importante do YouTube uma vez que, apesar das limitações que serão devidamente citadas, encontra-se mais adequado ao paradigma da iTV.

É proposta uma solução que será cabalmente analisada explicitando as opções tomadas e que requisitos do paradigma iTV foram devidamente tratados e implementados. Esta solução irá tratar fundamentalmente da parte prática associada a esta dissertação sendo que possuirá uma análise teórica dos problemas que se pretendem tratar e que arquitectura se levou no desenvolvimento deste projecto.

A solução proposta é passiva de uma validação com o intuito de se poder estudar a adequabilidade e praticidade num contexto de iTV. Por isso haverá uma secção responsável pela avaliação da solução em cenários concretos de utilização.

Abstract

The great development and worldwide acceptance of the Internet has led a large majority of the companies around the globe to rely on this technology for universal access to tackle the paradigms that society has today.

A vivid example is the group of companies of television channels. Previously, they only broadcasted television content for analog devices such as personal effects of television signals via antenna. Later, with these developments in Internet technology, the channels also began to be transmitted over the Internet. It is in this context that arises in the Internet television (iTV), which allows any user with access to web to view one channel at any point in the world without the constraint of having to have an antenna and a television.

The present document is a dissertation for the Master's degree in Computer Science Engineering at Instituto Superior Técnico, headquartered in Alameda, that belongs to Universidade Técnica de Lisboa.

This dissertation aims to build a framework within the paradigm of television on the Internet. The main target are the producers of TV channels that can create and manage their resources so that later can be transmitted by a digital signal via the Web.

It will examine several existing systems that use this paradigm of iTV. It will be analyzed WorldTV and Woma which are systems intended exclusively for the creation of television channels by the general public. Finally, it will be studied the most important case of YouTube since, despite the limitations that will be addressed later, it is more appropriate to the paradigm of iTV.

It is proposed a solution that will be fully reviewed explaining the choices made and what requirements of the iTV paradigm were adequately addressed and implemented. This solution will fundamentally address the practical part associated with this dissertation but it contains a theoretical analysis of the problems that are intended to be treated and which architecture that led to the development of this project.

The proposed solution has a validation in order to be able to study its suitability and practicality in the context of iTV. So there will be a section responsible for evaluating the solution in specific scenarios of use.

Palavras-chave

Canal

Lista (playlist)

Televisão na Internet

Gestão de recursos

Keywords

Channel

Playlist

Internet-Television (iTV)

Resource management

List of contents

1 - Introduction.....	1
1.1 Overview.....	2
1.1.1 Synonyms for iTV5.....	2
1.1.2 Interactive TV and Interactive TV.....	2
1.1.3 Overview and landmarks	2
1.1.4 User, producer and viewer.....	3
1.1.5 Technologies.....	4
1.2 Context of the project.....	4
1.2.1 WebComfort.....	4
1.3 iTV issues.....	4
1.4 Objectives.....	5
1.5 Structure of the dissertation.....	5
2 Reference model and desirable features.....	7
2.1 Use cases.....	8
3 Related work.....	13
3.1 Systems' analysis.....	13
3.1.1 WorldTV.....	13
3.1.2 WoMa.....	14
3.1.3 YouTube.....	15
3.2 Overview of the systems' advantages and disadvantages.....	17
3.3 Feature comparison.....	18
3.4 Discussion.....	18
4 WebC-iTV: conception issues.....	19
4.1 Domain Model.....	19
5 WebC-iTV: design issues.....	21
5.1 Supporting technologies.....	21
5.1.1 WebComfort.....	21
5.1.2 SWFObject.....	22
5.1.3 GrannetDotNet.....	22
5.1.4 JW Player.....	22
5.1.5 Media Handler Pro.....	22
5.1.6 Flash FLV Meta Reader.....	23
5.2 3-Tier Architecture.....	23
5.3 Modules' architecture.....	24
5.4 Modules and Domain Classes.....	25
5.5 Initial prototype.....	26
5.5.1 Intermediate solution.....	26
5.6 Final solution.....	28

5.6.1	Authoring Modules/Pages.....	33
5.6.2	Channel Modules.....	39
6	Preliminary validation.....	41
6.1	Business scenarios.....	41
6.2	Execution of the business scenarios.....	42
6.3	Discussion.....	43
7	Conclusions.....	45
7.1	Final conclusions.....	45
7.2	Future work.....	45
8	References.....	47
A1	Feature comparison of some existing systems.....	49

List of figures

Figure 1. Use Case 1.....	8
Figure 2. Use Case 2.....	9
Figure 3. Use Case 3.....	10
Figure 4. Use Case 4.....	11
Figure 5. Domain model.....	20
Figure 6. Three tier architecture.....	23
Figure 7. Authoring modules.....	24
Figure 8. Channel modules.....	24
Figure 9. Playlist selection.....	27
Figure 10. Playlist editor.....	28
Figure 11. Homepage of the solution.....	29
Figure 12. Tab Edição.....	30
Figure 13. Left container.....	30
Figure 14. Middle container.....	31
Figure 15. Right container.....	32
Figure 16. Channel's preferences.....	32
Figure 17. Channel's edition.....	34
Figure 18. Playlist's edition.....	36
Figure 19. Resource's edition.....	38
Figure 20. Live broadcast.....	39
Figure 21. Full playlist.....	40
Figure 22. WebChannel's homepage.....	44

List of tables

Table 1. Systems' summary.....	17
Table 2. Feature comparison.....	18
Table 3. Association between Classes and Modules.....	25

Acronyms

WebC	WebComfort
URL	Uniform Resource Locator
TV	Television
iTV	Internet Television
IPTV	Internet Protocol Television
DTV	Digital Television
GMT	Greenwich Mean Time
VoD	Video-on-demand
API	Application Programming Interface
CMS	Content Management System
PDA	Personal Digital Assistant
CSS	Cascading Style Sheets
SWF	Shockwave Flash
GUI	Graphical User Interface
WCM	Web Content Management System
PHP	PHP: Hypertext Preprocessor
SQL	Structured Query Language
UC	Use Case
IT	Information Technology
XML	eXtensible Markup Language

1 - Introduction

With the exponential growth of web technologies and physical resources, devices that used analog systems became rapidly obsolete giving place to the digital ones. Televisions, radios, cameras, frames, among others, experienced a radical change of its inner technology and hardware in order to adopt this new fundamental requirement, that is, to use digital systems.

Digital TV (DTV) is one of the products that born in consequence of this changing [10] and one of its main effects was that the TV (media) content could be authored and managed by computers prior to its broadcasting¹.

Meanwhile, Internet became popular and widely used resulting also in various transformations in the computer science area. One of these was the fact that private (media) resources could easily become public ones as they could be shared over the web. Also, with the expansion of Internet, some tv companies began to try some experiments by broadcasting television over the Internet [9].

These two new technologies put TV in the next step of evolution: the Internet Television (iTV) [11].

As iTV gained popularity, its technologies growth and it was created an innovative way of seeing and interacting with television content: the interactive video. This new paradigm turned advertising more appealing and easier, giving birth to a new tendency in iTV: the impulsive purchase (or impulsive buying). Although it previously existed in other contexts, this kind of purchase, in which a user (spontaneously) buys a product without planning it², became popular in the iTV context as it allows a user to purchase a product while seeing its broadcasted advertising movie. By clicking on the movie (or a specific object on the movie), an event is triggered provoking the appearance of metadata about the product in focus appears (like the website where it can be bought, its price, etc.).

Wikipedia shows a great list of Internet Television Providers in the whole world³. It's clear that the Continent with more Countries that provide Internet Television is Europe, showing the great adhesion of the multimedia enterprises in this area.

Victor Keegan, writer in the guardian.co.uk magazine, states: "But the really interesting things are happening where users are creating their own programmes and then either uploading them to YouTube and vodpod (the easy option) or choosing one of an exploding number of sites that provide you with your own global channel (usually for free) on mobile, PC or both."⁴. Its commentary states a foreseeable truth: Internet Television tends to grow not to decrease or become obsolete [12].

1 <http://www.oracle.com/technetwork/java/javame/techintertv052101-150048.pdf>

2 <http://www.uie.com/publications/whitepapers/ImpulseBuying.pdf>

3 http://en.wikipedia.org/wiki/List_of_Internet_television_providers

4 <http://www.guardian.co.uk/technology/2008/sep/11/internet.socialnetworking>

1.1 Overview

Firstly, it's mandatory to introduce some concepts around the iTV paradigm, such as to what's it means the acronym iTV, what other concepts are related to iTV and differences between two different paradigms in which iTV is one of them.

This section also introduces the frameworks and tools which were used in the development and deployment of the project.

1.1.1 Synonyms for iTV⁵

iTV stands both for Internet-Television or Interactive-Television. This late definition will be explained later and it refers to another paradigm which is frequently used in Internet-Television. Interactive-Television.

Other terms are used when referring to this subject. They are: ITV, i-TV, WebTV, Online TV, Network TV, Broadband TV, Web-based TV, Web-enabled TVs, Net TV InternetTV, Broadband ITV and IPTV.

1.1.2 Interactive TV and Interactive TV

Internet Television (also commonly known as IPTV) refers to television content that's placed in an internet repository that can be later broadcasted any device that has Internet access (either if it's a computer with a browser or a set-top box) [1].

Interactive-TV refers to the fact that a user might interact with the broadcasted content in various ways: voting programs, teletext, video-on-demand (VoD) [2]. Internet-TV is a case of Interactive-TV since it (normally) allows some level of interaction in its contents.

The scope of this dissertation, regarding interactivity, is the Interactive Video. In this context, Interactive Video is defined as the clicking act over a video in order to obtain information about the content that is being streamed. Normally, it's used in advertising: a product is being presented and the user might click over the video to view information about that product or even to access a specific website to make the purchase. It can be generalized to other areas such as sports. In this case, a user can click over a sport match (or a player in highlights) in order to see information about it.

1.1.3 Overview and landmarks

In 1994/1995 (sources differ in the date), in the early years of the internet, there was a landmark in the history of television: the first broadcast show over the internet. This broadcast was made by ABC and the show called: "World News Now"⁶. At that time, the number of Internet users was scarce. Statistics shows that in December 1995, only about 16 million people in the world used the internet⁷ (less than 1% of the world population). This fact states that ABC was a pioneer in joining both television and internet. This landmark states the fact that the idea of mixing internet and television isn't a new concept. It has, by now, over 15

⁵ http://www.itvdictionary.com/internet_tv_directory.html

⁶ <http://www.apsc.org/>

⁷ <http://www.internetworldstats.com/emarketing.htm>

years.

In the same year of 1995, videoconference over the Internet was already a reality and it was beginning to be used in specific and private contexts [8].

Only in the beginning of this century iTV began to have developments. December 2000's statistics⁸ showed that the number of Internet users was over 360 millions, corresponding to almost 6% of the world population – an increasing of about 1% per year. This growth led companies to make investments in this area thus generation a new product: internet to private homes. As a direct consequence of this fact, the number of internet users exponentially increased as instant access to the web was at a distance of just a mouse click and the physical requirements were just a simple modem and phone line.

As stated earlier, when Internet gained popularity, more and more content began to be shared in the network not only between the organizations and the clients, but also between internet (common) users. It eventually led to the creation of specific databases, called “portals”, that allowed many users to download and upload content to be shared in the internet. One of the most popular examples of these portals is YouTube (founded in 2005 [3]). These portals' major purpose is to allow the instant sharing of any kind of videos and as they grew rapidly in terms of popularity and subscribers, television companies started to deliver some of their content over these portals. Parallel to this, these companies started also to deliver some media content online, on their own websites and, nowadays, live television can be watched in the internet.

Historically, the study of television technology started at the end of the XIX century [14] and internet technology only begun in the 1970's [15], meaning that TV has a century ahead regarding its discoveries and regarding its worldwide acceptance. However, internet users are now a quarter of the worldwide population.

1.1.4 User, producer and viewer

During this dissertation, the terms “user”, “producer” and “viewer” are going to be oftenly used. Producer is the person that creates and owns a channel and is responsible for its content management. A viewer is the person that accesses the channel to only view the movies its playlists. A user is any person that is using a system disregarding if it is a producer or a viewer since it can be both of them. This last entity is used in systems like YouTube, where a registered user can create a channel but also view content of other users. There's almost no distinction between a producer and a viewer in this perspective. The scope of this solution, however, focus more on the producer, which is also a user. The viewer has not any representation in the implementation of this project but it's present in this document as an abstract entity which can see the final product – a channel.

⁸ <http://www.internetworldstats.com/emarketing.htm>

1.1.5 Technologies

Some of the features of this project were implemented by the aid of some free existing open-source projects. The reason for this importation was to focus more on the solution of the iTV paradigm and less on collateral details.

Each tool is explained later in the dissertation but they can be listed here. They are: SWFObject⁹, GrannetDotNet¹⁰, JW Player¹¹, Media Handler Pro¹², Flash FLV Meta Reader¹³.

1.2 Context of the project

As stressed in the introduction, this document consists in a Master's Thesis or Dissertation which leads to the graduation of Master in Computer Science Engineering at Instituto Superior Técnico of Universidade Técnica de Lisboa.

The whole project was carried out in connection with INESC-ID at the headquarters of Lisbon having as advisor Professor Alberto Silva and, as accompanying teachers, the Researchers and Engineers João Saraiva and David Ferreira.

1.2.1 WebComfort

WebComfort is a CMS developed by SIQuant¹⁴ which is a company created in 2004 and it's specialized in the area of IT.

This framework is explained later in detail but in this section it's important to introduce the fact that the whole solution of this project was designed to be deployed in WebComfort. It is structured in modules which was part of the project's implementation (implementation of WebComfort modules) and each page is separated in WebComfort Tabs.

1.3 iTV issues

iTV issues refers to the most common problems that this paradigm wants to approach and resolve. Although it depends on the Internet as it's deployed over it, iTV has no concerns with aspects like upload time and fault tolerance because these are problems that are not directly related to this paradigm – it concerns networking paradigms. As a consequence, they will not be studied in this document.

The biggest issue concerning iTV is resource management. Over time, the companies' databases tend to grow in resources (which can be movies, playlists, advertisement banners, etc) and if there's no specific order in the addition of these resources, there might be some of them repeated. Also, for a producer to easily create a playlist using existing resources, he must have efficient filters to clearly see what is wanted.

9 <http://code.google.com/p/swfobject/>

10 <http://gdnsfwobject.codeplex.com/> and <http://www.grannet.net/>

11 <http://www.longtailvideo.com/players/>

12 <http://www.mediasoftpro.com/flash-streaming.html>

13 [http://johnnyer.name/post/Flash-FLV-meta-reader-in-NET-\(C\).aspx](http://johnnyer.name/post/Flash-FLV-meta-reader-in-NET-(C).aspx)

14 <http://www.siquant.pt/portal/Contactos@133.aspx>

Other issue concerns the creation of the playlists. Since there are no standards yet, each producer might be confronted with as many different solutions as the systems that he uses. The playlists may be on a XML file, on a database, serialized in a file, etc.

The edition of the channel is also an issue since there are different solutions that can be approached. The simplest solution is to automatically create a channel according to some parameters given by a producer. A more complex solution passes for the manual edition of the channel's logic and visual presentation.

In recent years, interactivity was also inserted as a iTV issue. This concerns to the interaction that a broadcasted content might provide to its viewer. It can be polling on shows or contests that requires an answer, between others.

1.4 Objectives

The objective of this project is to create a framework over WebComfort that will become possible for a company's producer to create an online channel thus allowing a viewer to watch that channel in the web. It will be analyzed the state of the art in the iTV features and elaborated a theoretical solution for this paradigm. This analysis will take into account existing systems and their evaluation. It will be taken into account characteristics such as content management, available features and the GUI.

With this project, a company's producer will be able to: manage channels, playlists and resources and insert hidden advertisement hyperlinks.

As a direct result from this previous objective, a viewer will be able to navigate in the channel, see the playlist, see the live emission and to click in the resource that's being transmitted in order to navigate to another site (which is decided by the producer).

Another objective of this document is to provide a simple solution for the Interactive Video paradigm. As Interactive Video is also a big subject that is passive of a dissertation for its own, the solution will only deal with the mouse interaction over a whole movie. This objective is present in this document since it's one of the issues that iTV should treat.

1.5 Structure of the dissertation

This dissertation consists of six chapters, each with subtopics. Chapter 1 gives an overview of the project, stating the context in which iTV born and where it's used nowadays. It also states the main issues to be addressed and the objectives for the final solution.

Chapter 2 lists the main requirements and some desirable features that are part of the reference model.

Chapter 3 will study the state of the art in the iTV paradigm by studying and comparing some existing systems.

Chapter 4 starts to converge to the proposed solution by detailing some conception issues behind the solution itself. It details the use cases that were the basis for the solution.

Chapter 5 gives the detailed solution. It explains how it was approached in a first prototype passing to the final approach of the final solution.

Chapter 6 validates the implemented project by testing it in some hypothetical cases that happens in real systems of this area.

Chapter 7 concludes the document by summing up the conclusions taken throughout the dissertation and by suggesting some future work.

2 Reference model and desirable features

By studying and comparing various existing systems that implement iTV, a list of features and characteristics was created and they are mandatory to solve the iTV problems. This list is called the reference model. They are:

- **Channel creator:** a channel is more an identity than anything else. If playlists are scattered in websites or web portals, there's no identity since these playlists are not connected between them. For a producer to individualize the company's mission and objectives about iTV, it must have a channel. This way, a viewer can access the playlists, expecting to see a certain type of content that is according to its preferences.
- **Resource management:** a playlist is constituted by various resource entries. These resources must, somehow, be managed so that they can be properly used in the playlist. The resource management requirement also includes other features like playlist management, channel management and everything else that has to be managed by the producer.
- **Playlist creator:** for the iTV paradigm to work as it's supposed, it must allow the creation of playlists (that, normally, are changed in daily basis), so that viewers can access different resources at different times of the day.
- **Resource importation:** the resources can be stored on a local disk or somewhere in a remote disk for example, on a portal like YouTube or MySpace. It's important that a iTV system allows the importation of resources not only from local disks but also from remote places including portals.

From this model, it can be concluded that there are three main entities: channel, playlists and resource. Without any of them the iTV paradigm cannot be solved or approached since it would not allow the creation of an online channel.

Adding to the reference model are some desirable features that are important for the resolution of some of the issues concerning iTV. These features can be grouped by entity:

- **Channel:** manage channel's information, configure the channel's logic and visual.
- **Playlist:** change playlist's date, define the opening time of the playlist, allow the creation of playlist entries with specific associated metadata.
- **Resource:** manage resource's information, have access to resource's metadata, have filters to view only the desirable resources. Apart from these, Video interactivity is one of the most desirable features because the iTV paradigm becomes even more useful when adding some interaction in the broadcasted content. This interaction is important as it changes the role of the viewer from mere spectator to actor or participant.

2.1 Use cases

The given reference model was based on some use cases created for this topic of iTV. They are presented on this section.

The use cases are presented as UML diagrams [7].

Figure 1. Use Case 1

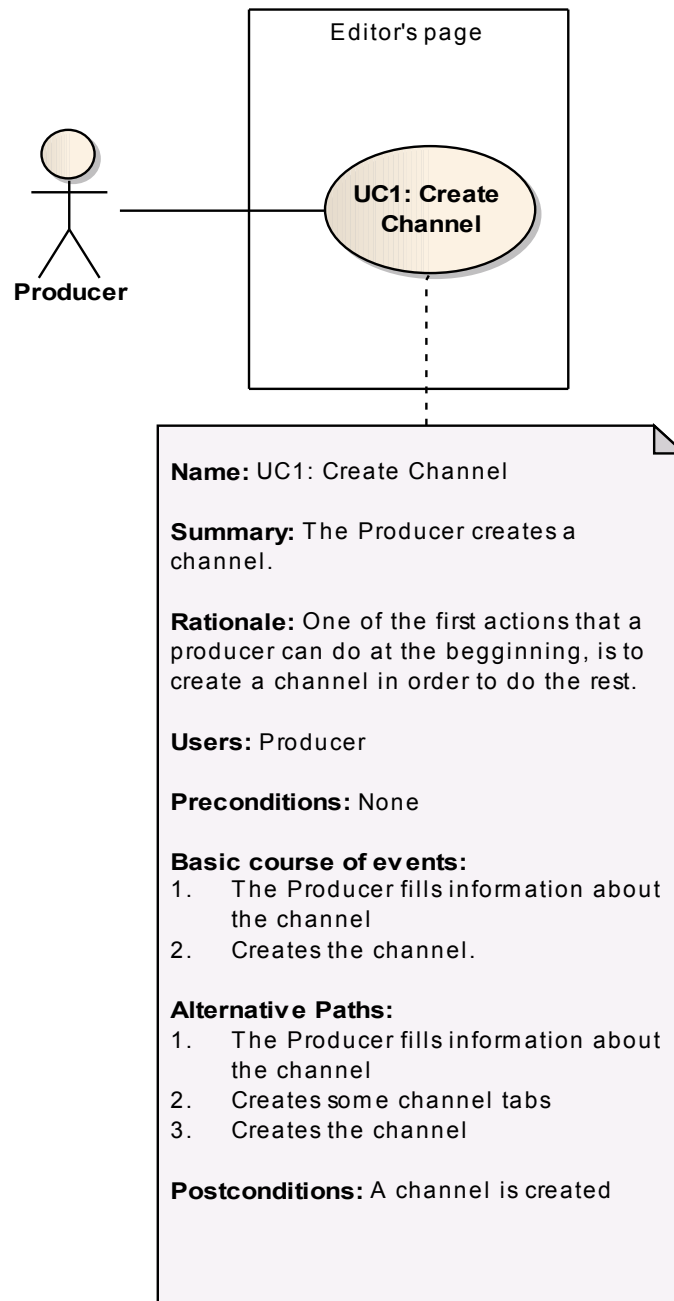


Figure 2. Use Case 2

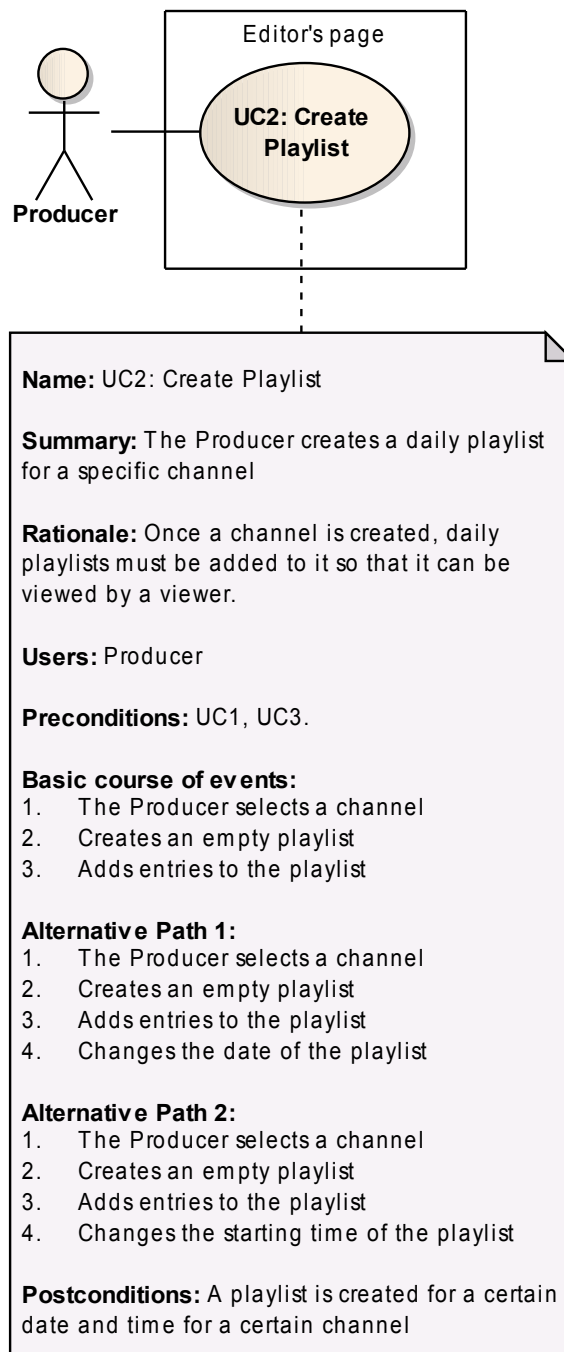


Figure 3. Use Case 3

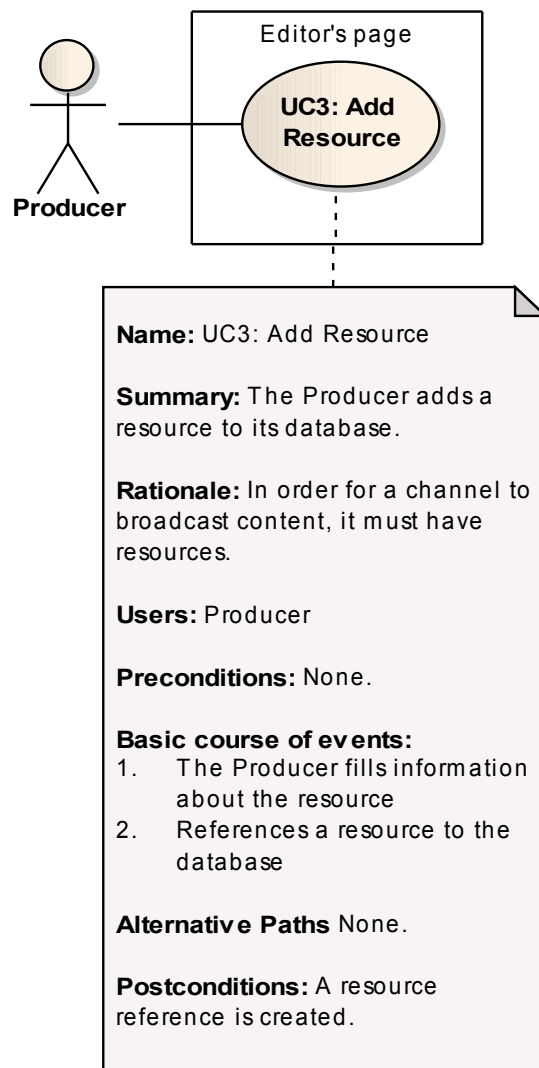
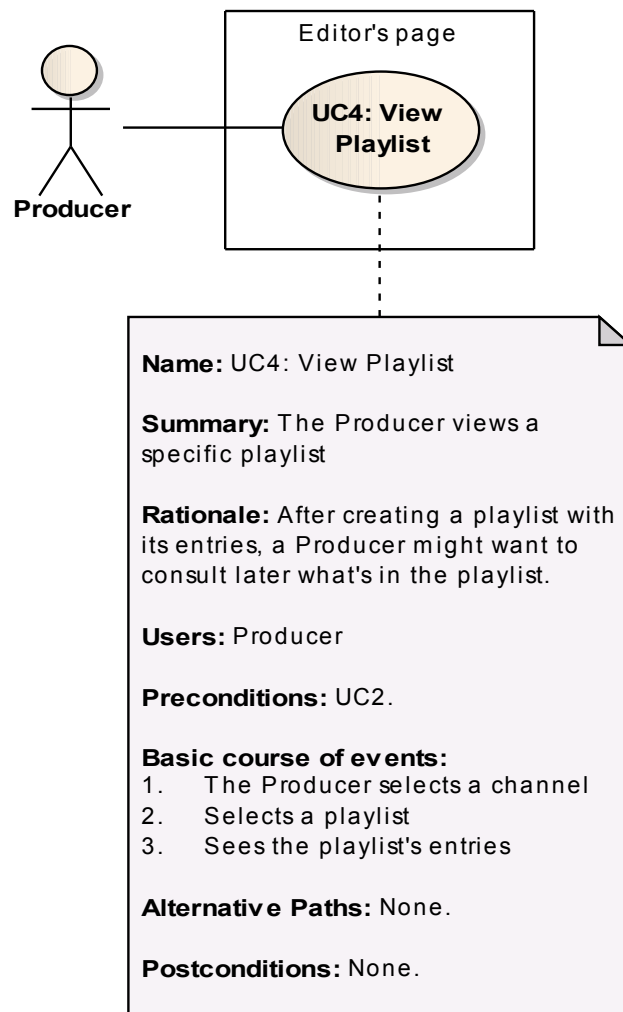


Figure 4. Use Case 4



3 Related work

Nowadays, there are a great number of systems that cover many features of the iTV paradigm. These systems offer a user a large number of different features that goes from TV companies that provide its broadcast content in the Internet for free public access to systems that allow any user to freely create its channel and control its broadcasted content.

Next, it will be analyzed some existing systems that give a solution for the iTV paradigm.

3.1 Systems' analysis

There are many systems that partially or totally fulfill the iTV requirements.

This section will only show a few examples of these systems and some of its features. However, annex A1, at the end of this document, contains a table giving more examples and showing more features, easing the comparison of each system. As for this part, some systems that can be stated are:

- WorldTV: this is one platform that allows the creation of a channel with a playlist. It's a good starting point for the study of the iTV paradigm and will also be studied later on, although it's not a complete option. WorldTV is similar to many other systems that are listed in *Annex A1*.
- YouTube: it's one of the most complete platforms in terms of iTV, as it also provides Interactive Video authoring. This is another system that will be studied in this document and it's the most important one.
- Castter: in Portugal, this is a reference platform for iTV since it allows for a common person to literally become a channel's producer and manger. Many organizations use this platform to create its own channels. Two examples are Querqus and Greenpeace, both environmental organizations.
- Kyte: it's much like WorldTV. It allows the creation of shows (that represents playlists) and the media content is inserted in the channel via hard drive or webcam.
- WoMa: a website dedicated to create channels with videos only about martial arts. This system will be studied in the dissertation.
- Icareus: this is a fully featured platform with several modules that goes from pre-processing media content to the broadcasting feature.

3.1.1 WorldTV

WorldTV is a Brazilian web platform which allows a common internet user to create its own online channel, create a single playlist and find video resources.

The resources can be added in the playlist from two different places: from portals like YouTube or Metacafé or it can be recorded and added from personal videos through the computer's webcam.

This platform is simple though with many features¹⁵. One important feature that is integrated is the statistic's generator. It can show the audience over time and the time spent viewing the channel.

¹⁵ <http://worldtv.com/pages/manual/>

It allows the creation of widgets to promote the site in other websites and it also possesses a connection to Twitter. This allows Twitter user's to follow the channel's updates, for instance, when the playlist has changed.

Its interface is adequate for a newbie user since it uses a language that is most of it common sense. The options are very well distributed along the site and are easy to find.

To create a channel, a user simply enters the WorldTV's homepage and selects the option to create a channel. After a series of questions regarding the channel's and user's identifications, the user is presented with the editor's page which enables it to start managing the channel's content. A big limitation of this platform is the fact that only allows one playlist, which is created by default. Any content to the channel must be inserted and deleted from this single playlist.

To create the playlist entries, a user searches for movies in WorldTV's referred portals and adds them in a temporary section called "My Video Library". Once this is done, a user can start dragging the videos into the "My Playlist" section, creating, therefore, the various entries of this sole playlist.

This platform provides to users the option of recording videos from a cell phone and save these resources directly in the site. Obviously, this is not a feature for worldwide use since not every cell phone is compatible with this option.

In terms of resource management this platforms lacks important features such as the ability of uploading videos from a hard drive. This prevents a user to put directly its media content in its playlist. First, it has to put it in one of the platform's compatible and only then the user can reference it to the playlist. The compatible media content are FLV and MP4 files.

This site doesn't have a registration component and although the editor section of the channel can be for private access, by setting a password, it's, by default, in public domain which means that anyone that accesses the WorldTV website can alter any channel.

One final feature to be stated is the possibility of translating the website into eight different languages. There's an option always present that allows the user to immediately change from one language to another. Although it's a complete option for an average user, it does not provide enough features for a professional use. This platform does not provide a live broadcast features, though there's a beta version being developed.

3.1.2 WoMa

WoMa¹⁶ is a website dedicated to martial arts¹⁷. All the videos that are allowed in this platform must consist in movies related to any kind of martial art. These media can be tutorials about any movement, philosophy behind a certain type of fight, any documentary about the culture that's the basis of a martial art. Secondly, it allows movie trailers but it's not the website's target.

As it only allows this specific content, there's no possibility for a common user to upload a video. This prevents users from filling the repository with media that's not related to martial arts. In the iTV context this represents a huge limit since it prevents a user to upload its own videos but it also prevents the reference of videos that are from other media portals (such as YouTube). In this platform a user can only make playlists

16 <http://www.woma.tv/docs/your-woma-channel.html>

17 <http://www.woma.tv/docs/what-is-all-this.html>

with movies already inserted by authorized users.

The GUI of this platform is not as friendly as WorldTV and it takes some trial and error to understand how to move the media to the playlist.

The content management of this website, in terms of playlist management, is simple. There's no limit on the amount of playlists that the user can create, and each playlist can have as many movies as the user wants.

For the movies to be accessed and put on the playlists, a user must first find a movie in the WoMa repository (by searching in the existing channels or through the existing users' videos) and add it to a temporary container. Once the user has enough movies in this container, it can start adding the movies to the playlists. One positive aspect is the fact that the insertion of these movies in the playlists uses drag-and-drop technique. Normally, this represents an improvement of the traditional "add buttons" as it saves space on the website and makes the interface friendlier. There's also the possibility to select several movies at the same time to put them in the playlists.

Unfortunately, in addition to the already stated limitation, WoMa only allows the creation of a channel per user. This means that if someone wants more than one channel, it has to register another account. It also means that there's no sharing between movies of each temporary container, that is, if there's a set of movies in a container's account that's going to be used in other account, they have to be re-added. This represents a great loss of time.

Despite of its disadvantages, WoMa fulfill its main targets – to provide a portal for channels dedicated exclusively to martial arts and also a simple content management possibility for the creation of playlists.

3.1.3 YouTube

Although mostly known as an online video sharing portal, YouTube's programmers implemented functions that allow its users to create a channel with all its required features.

The first limitation to be stated is in the creation of a channel. Like WoMa, in YouTube each user has only a single channel. If someone wants to create more than one channel it has to create a new YouTube account in order to do it. This represents a big limitation as it makes channel management almost unfeasible. If a user wants to work with multiple channels it must have each channel opened in different internet browsers as each browser will allow only a single login, hence a single user.

Despite this limitation, YouTube is still a good choice for a producer since its advantages outweigh this disadvantage. For instance, the playlist creator is quite complete as it allows the creation of two different kinds of playlists: various normal playlist and a single fast playlist. The main difference resides on the fact that the fast playlist allows a user to see a movie in YouTube and add it to this temporary playlist so that it can manage the video later. This playlist's main purpose is to allow a user to now waste time thinking of where to put a movie or even to allow a user to save a movie to see it later. The other playlists are normal ones and there's no limit on their number – a user can create as many as it wants.

In terms of resource management, YouTube is a five star choice for any producer. Its interface is easy to use and has many features that make the creation of a channel in a whole easier for any user. The media content can be uploaded from the hard disk and it can be of almost any kind. If it's the will of the user, the

video can also be uploaded via a webcam. A playlist can be composed with both videos already published by other users and videos published the user itself. Playlist management is also quite intuitive: a user can decide, first of all, the order of the movies inside each playlist and, later, the order of the playlists in the channel.

As YouTube is not supposed to be a professional channel platform, it does not allow live content or strong interactive features. The only interactivity that YouTube provides consists in annotations and buttons to links, which is more and less what is wanted to this project. Before publishing a video, a user can add features to the video such as this last described. It can also add information to the movie and subtitles.

Each movie has associated a statistics tool. It provides the normal information such as the number of viewers, but it can also provide much complex queries like filter the viewers by region.

The final feature that YouTube has that's desirable to this project is the Multilingual support. On the bottom of each page, a user can select one from nineteen different languages. One problem with this feature is the fact that is somehow hidden as it only appears in the end of the website and in little letters.

YouTube is presented in this section has a reference example of the project to be developed since it's one of the most complete and free systems. As it was stated, this platform provides features as:

- Creation and management of a single channel;
- Creation and management of various playlists;
- Video authoring (in terms of adding notes and buttons);
- Statistics tools;
- Multilingual support.

One of YouTube's few disadvantages, in this specific context, is that it's not liable to be used by a TV company because of the fact that the site has many users and channels and the creation of one more channel would be a tear drop in a big ocean, not fulfilling most companies' objectives (such as, reach a certain audience). But it's free and fully featured and it comprises the whole required and desired features for this project.

3.2 Overview of the systems' advantages and disadvantages

The table below represents a summary of all the studied systems' advantages and disadvantages.

Table 1. Systems' summary

System	Advantages	Disadvantages
WorldTV	Intuitive and easy GUI; gets popular video portals; solves the iTV paradigm of an average user	Allows only one playlist; doesn't yet provide a live content feature (it's in beta release); doesn't provide a way of uploading a movie; by default, there's no password for the editor of the channel
WoMa	Allows drag-and-drop of many movies on at a time into playlists; perfect for martial arts fans as it only allows movies of this subject;	A common user can't upload a video; doesn't have a live feature (in consequence of the previous topic); allows only one channel per user;
YouTube	Used worldwide; easy to learn how to use; allows the creation of multiple playlists; any user can upload a video from the hard driver or from the webcam; allows Interactive Video	Doesn't allow the creation of multiple channels; not fit for most of TV companies

3.3 Feature comparison

Taking into account the reference model and its desirable features, the following table enables a comparison of each studied system in this context.

Table 2. Feature comparison

Feature	WorldTV	WoMa	YouTube
Number of Channels	Unlimited	1	1
Playlist creator	Yes	Yes	Yes
Number of Playlists	1	Unlimited	Unlimited
Video interactivity	No	No	Yes
Upload videos	No	No	Yes
Video reference	Yes	Yes	Yes
Portals compatible for resource importation	Google, Metacafe, YouTube, AOL, MySpace	WoMa	YouTube

3.4 Discussion

Comparing the three studied systems we can see that YouTube is the reference system for this paradigm as it covers more desirable features than the others. Also, it's more adequate do the given reference model.

Even though it only allows one channel per user, YouTube gives the most complete experience in the iTV paradigm.

Although it allows the massive repetition of its movies, YouTube has a powerful search engine which allows a movie to be searched only by opening movies with similar name or subject.

The interactivity is very complete allowing the user to create multiple links and boxes throughout the movies although it does not provide a way of making the viewer to become a participant of this same movie. An improvement it would be the possibility of a viewer to participate in contests or pools in YouTube as many interaction features are already implemented.

4 WebC-iTV: conception issues

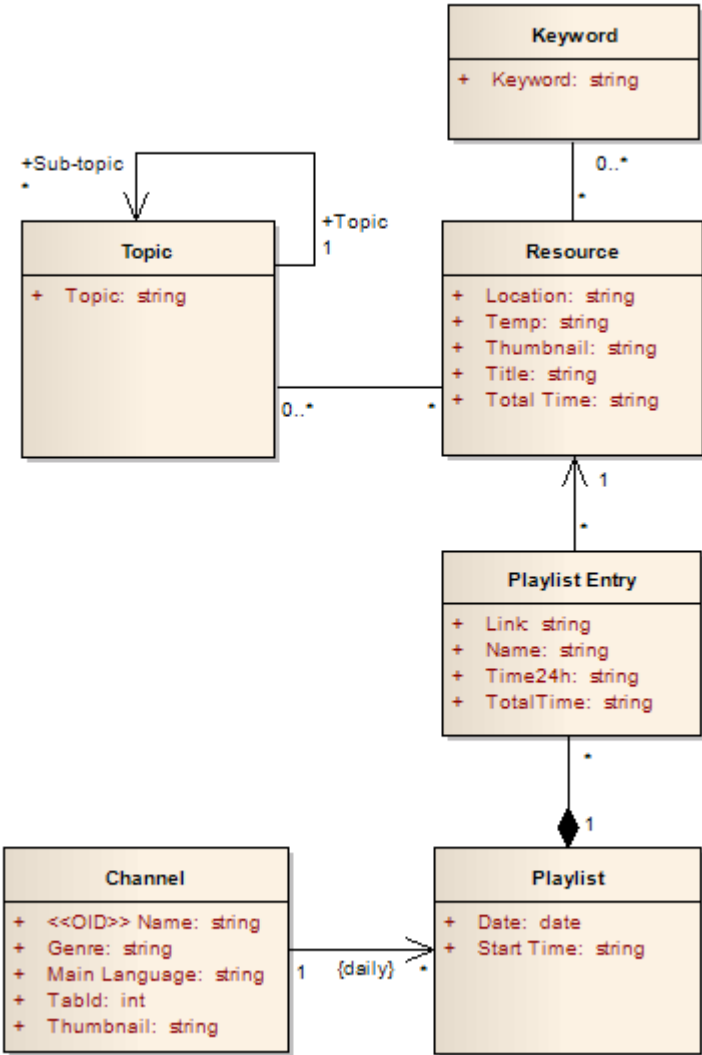
Before passing to the implementation of the solution, some concepts are clarified in this section. By explaining the domain model of the system, each of the most important concepts are introduced and explained so that they can be used in the next section.

4.1 Domain Model

Figure 5, below, represents the system's domain model. The most important class in this project is the Playlist because it's what allows a channel to perform its intended purpose, that is, to publish the resources so that they can be seen, and it's also what makes the resources achieving their own objectives – to be broadcasted. I'll pass to the detailed explanation of each domain's class and their purpose:

- Channel: Entity responsible for hosting the tabs and the playlists. It also contains information about the created channel, such as name and logo.
- Playlist: Its responsibility is high since it has to provide a daily list of resources for the channel to function properly. Without a playlist, a channel won't be able to broadcast. It saves the date of the playlist so that it can be automatically broadcasted in the given day.
- Playlistentry: This entity is responsible for connecting a resource to a playlist. It's important as it saves the duration of the resource and also its relative time in the playlist, that is, the time of the ending of the resource taking into account the starting time in the playlist. For instance, if a 1 minute resource starts at 12:00 in the playlist, the playlistentry saves the information about the "1 minute" and also the ending at "12:01" thus allowing the channel to know what playlist entry to play at a given time.
- Resource: This entity is responsible for saving the path of a resource and to add some metadata such as it's name and duration.
- Keyword: Minor entity that stochastically saves keywords that can be associated to a resource to ease its search.
- Topic: Similar to the previous, but it hierarchically saves topics. This hierarchy it's defined by the producer and it's a parent-child relation, that is, a topic might be sub-topic of another one.

Figure 5. Domain model



5 WebC-iTV: design issues

As already stated, the solution of the iTV paradigm in this project passes for the development of a framework over WebComfort.

For this solution, only two sets of modules are going to be developed: a set for authoring the whole features of the channel and a set for playing the content of the channel.

WebComfort provides multiple containers for deploying modules that can easily be rearranged. Normally, only two containers are used for the sake of the comprehension of the website (too many containers would make a page look rather deranged): one main container which is in the center of the website and occupies most of it, and a secondary container which is in the left side of the website and can be compared to a frame, that is, it only occupies the space of a strip in this side.

The solution uses two and three containers: the first to the channel's viewing page and the second to the authoring page (which is the core of the project) and homepage.

5.1 Supporting technologies

As mentioned in section 1.1.5, this project used some free existing tools in its implementation. These tools' objective were to simplify some features that were not directly related to this paradigm. By using these tools, I could dedicate more time to the effective issues of the iTV paradigm not spending time in the implementation of secondary features that wouldn't contribute for the enrichment of the project.

It was also introduced in section 1.2.1 the WebComfort which will be explained in this section along with the other tools that were used.

5.1.1 WebComfort

WebComfort it's a Content Management System (CMS) developed in Microsoft ASP.NET 2.0 and it's available for computer clients (such as internet browsers) and also for mobile clients (cell phones and PDAs, for example).

One of its main purposes is to allow management of web content – not only content in a web application but also content regarding cooperation between various web applications.

All the WebComfort's features are divided in Modules [4] which can be deployed (and removed) in specific sections. This allows a WebComfort's webpage to have multiple Modules, or features, running in the same page without having to open a page for each one.

Lastly, this platform provides a visual interface that is normalized. When a project is deployed, it must import the master page and CSS classes already defined [5], so that it can be viewed properly and accordingly to the rest of the WebComfort's native Modules.

In WebComfort, all pages are divided into tabs which are created and managed by the user. These tabs allows a programmer to deploy various modules into different containers according to its needs.

WebComfort's library has a great amount of interfaces that can be easily used. Just as examples, it's provided a Language API which allows any module (and pages) to have multiple languages easily implemented; there's also Tabs API and Modules API which allows to manipulate the modules in a page and also manage the Tabs.

This project is will be developed using WebComfort's existing features and according to its normalized interface and visual appearance.

5.1.2 SWFObject

Although it was not directly used, this open-source project aims to provide an alternative way of embed media other than Flash Player. It provides a Javascript API so that it can be easily used by any web developer.

5.1.3 GrannetDotNet

As stated in the SWFObject site, GrannetDotNet is "An ASP.NET server control that uses SWFObject to render Flash movies using either dynamic or static publishing methods, as well as templates for alternate content.". This project works as a wrap so that SWFObject can be used in C# as an object class with the features already described.

It was used in the player module in order to programmatically create an html embed object that could have parameters passed at runtime. In this case, the runtime parameter was the location of the resource to be played in the player.

5.1.4 JW Player

This is a Flash Player that can be used in web pages. It has many features that allows a developer to decide the behavior of the player and also provides a JavaScript API. This project is open-source.

As the player did not met all the requirements that were needed for this project, the player was changed by altering its source-code which is given in its official site.

The main changes were two: disable the interaction with the controlbar as in real TV channels it wouldn't be possible to go forward in the broadcasted content; disable the "pause" when clicking over the movie, replacing that behavior to one that would access a JavaScript function which would be responsible for opening the link associated with that specific resource (Interactive Video paradigm).

5.1.5 Media Handler Pro

This project allows the conversion and encoding of flash movies. In this dissertation, Media Handler was used to create thumbnails for the resources that were referenced in the database.

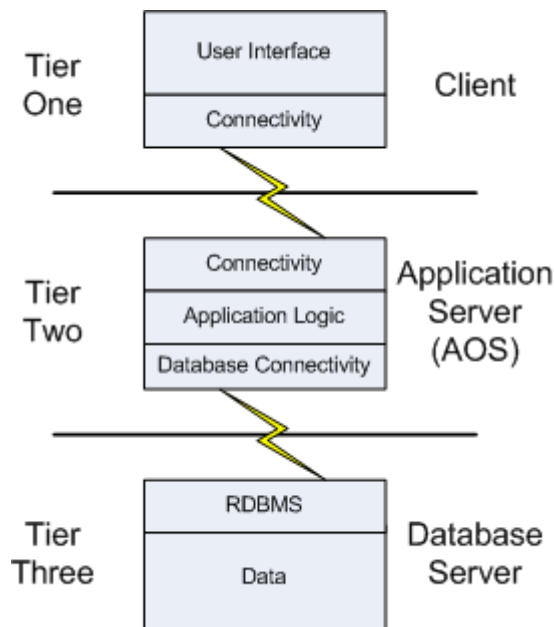
5.1.6 Flash FLV Meta Reader

Acquired in its official site, this simple project consists in a C# class that provides metadata information about FLV files. It was used to get each resource's duration automatically.

5.2 3-Tier Architecture¹⁸

Many enterprises around the world adhere to the three tiered architecture. As stated in its name, this architecture comprises the development of products, in this case software, dividing three main groups: presentation logic, business rules, business data. The first layer is related to the visual settings that are going to be presented to the final user; the second layer can be seen as the “core” of the product as it comprises the whole logic that responds to the interaction made by the user; the last layer is related to the way information is persisted so that it can be accessed after. Figure 6 shows this architecture in a scheme.

Figure 6. Three tier architecture¹⁹



This architecture allows an abstraction between different paradigms thus allowing three different entities, with different knowledge, the simultaneous development of each part.

The dissertation followed this architecture. For the presentation layer, it was used ASPX and ASCX file types; for the business rules it was used CS (C#, C-Sharp) files; for the last layer it was also used CS files but automatically generated from a Freeware program called MyGeneration which created an abstraction between the coding and the querying of the database. MyGeneration uses a dOODads which is “an elegant .NET architecture available in C# and VB.NET and capable of supporting any .NET managed data provider”²⁰ as stated in the official website.

18 <http://msdn.microsoft.com/en-us/library/ee658109.aspx> <http://www.linuxjournal.com/article/3508>

19 <http://msdn.microsoft.com/en-us/library/aa660629.aspx>

20 <http://www.mygenerationsoftware.com/portal/dOODads/Overview/tabid/63/Default.aspx>

5.3 Modules' architecture

The modules architecture consists in two main sets of modules: authoring modules and channel modules. Authoring modules consist in the universe of modules that allows a producer to manage and create a channel (and all its resources). Channel modules consist in the modules responsible for the visualization of the channel itself. These don't alter the project's data – in a certain sense, they are “read-only” modules.

Figure 7 and Figure 8 show the set of modules (UserControl files and ASPX Pages) separately.

Figure 7. Authoring modules

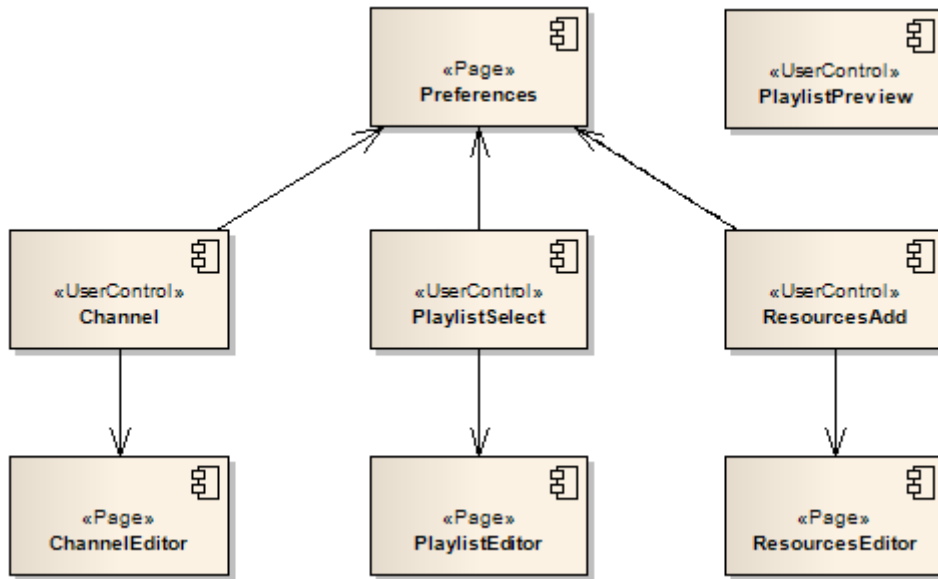


Figure 8. Channel modules

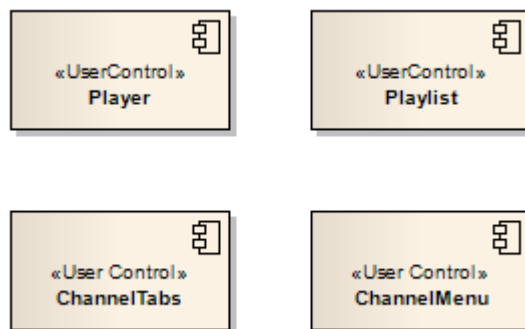


Figure 7 shows three sets of modules of the fundamental entities of this project: playlist, resource and channel. Each of them have the module Preferences associated in which each module can show/hide table columns according to its preferences defined in this module.

Figure 8 consists on a set of separated modules that are to be inserted in the containers of the channel's page. Playlist is self-explanatory, Player consists on the module that contains the player to broadcast the resources, ChannelTabs is the module that contains each tab of the channel previously created and ChannelMenu is the set of links that always appear on the channel in order to navigate through it. Later, it will be better explained with images of the project.

5.4 Modules and Domain Classes

Sections 4.1 and 5.3 showed the project's architecture in two different views: Domain Model and Modules' Architecture. These two are connected as each class in the domain model has a set of modules associated that makes use of those classes. Table 3 shows this connection.

Table 3. Association between Classes and Modules

Class	Modules/Pages
Channel	Channel, ChannelEditor
Playlist	PlaylistSelection, PlaylistEditor, PlaylistPreview, Playlist, Player
Resources	ResourcesAdd, ResourcesEditor
Tab	ChannelEditor, ChannelTabs, ChannelMenu
Playlistentry	PlaylistEditor
Topic	PlaylistEditor, ResourcesEditor
Keyword	PlaylistEditor, ResourcesEditor

This table means that the domain classes (on the left) were used on the modules (on the right).

5.5 Initial prototype

A prototype had been created (before the writing of this document) which allowed to study in practice various problems and solutions on the iTV paradigm and also provided an intermediate solution that would be used to develop the final project.

This section concerns the prototype that was supposed to represent a shadow of the final solution. The core ideas were maintained until the end of the project but some of them were modified or erased according to the requirements.

A note of caution: this section does not represent the final solution. It represents an intermediate solution taking into account some initial requirements. The description will reference some concepts that were abandoned later.

5.5.1 Intermediate solution

The initial idea was to start the producer's module by inserting (or choosing if it was already inserted) the company's name, after that the channel's name and, finally, the authoring of the playlists. But as this framework's target is to be used by a single company, it was neither efficient nor needed to do this choosing every time. Instead of that, a company inserts its name once, which was saved in a database. Whenever the company restarted the producer's module, there was no need to choose the company's name as it was automatically loaded from the database. This solution assumed that this platform was not going to be shared among different companies (otherwise it would have to be a registration component and an online database from which every producer would have to access to use the framework).

Once passed this small registration step, the producer would be presented with the list of channels that the company already possesses (if it was the first time, this list would be void). A producer could create as many channels as wanted – it was just a matter of adding a new channel in this menu. The identification of a channel consisted in filling some information about it: name, type of channel (sports, movies, general), access (public or premium), main language, GMT, URL of the channel and if there was a need for registering the channel in order to view its content or not. This same menu for channel selection, also showed the playlists associated with the selected channel (by default the first from the list was already selected). These playlists could be deleted and other playlists could be added. This meant that the management of both channels and playlists was made in a single web-page, saving time for the producer. The playlist's entries management were made by clicking on the desired playlist. By doing so, other web-page was presented which allowed the management of both the playlist entries and the resources (media content). A producer could add and remove entries in any part of the playlist. Each playlist entry was represented by a resource and it could have a specific title. This meant that a playlist entry was composed by a media resource, a title, the time of the resource and a timeline of the whole playlist, that is, the time when each resource started in the real time of the day. As for the resources, they also could be added and removed, although this last option was only available if no playlist used the resource in question.

The viewer module would have a search engine for the viewer to choose a channel according to its preferences. The homepage would contain a list of the most viewed channels, providing the viewer a way of accessing a channel without having to do any kind of search or navigation.

If the viewer wanted to access a premium channel, it firstly would have to register in the channel and provide personal information such as full name, address, credit card number (or other kind of payment). This solution didn't deal with the transactions between the viewer and the channel. It only provided the possibility of inserting a way of payment, not a way of transferring the money.

This prototype only allowed movies in flash, that is, in FLV extension as it used a flash object to read movies called "SWFObject".

The figures above are taken from the prototype which follows the WebComfort's standards. The final solution follows the same pattern.

Figure 9. Playlist selection



Figure 9 shows a cropped image of the appearance of the Playlist selection module. There's a drop down list of the existing channels (in the company). Just below, there's a link button that allows the creation of a new playlist for the selected channel. Finally, there's a list of the existing playlist and the option of edition and deletion.

Figure 10. Playlist editor

Location

Duration (HH:MM:SS) [Insert new resource](#)

Playlist's date: 06-01-2010

Insert new date: (DD-MM-YYYY) [Change date](#)

	LOCATION	TOTALTIME
Seleccionar	http://localhost/wbc/1.flv	00:04:05
Seleccionar	http://localhost/wbc/2.flv	00:03:23
Seleccionar	http://localhost/wbc/3.flv	00:03:38
Seleccionar	http://localhost/wbc/4.flv	00:04:38

Insert title:

	ID	NAME	RESOURCES	PLAYLIST	TOTALTIME	TIME_24H
Delete	1		http://localhost/wbc/1.flv	1	00:04:05	00:04:05
Delete	2		http://localhost/wbc/2.flv	1	00:03:23	00:07:28
Delete	3		http://localhost/wbc/4.flv	1	00:04:38	00:12:06

Figure 10 shows the Playlist editor which works as a playlist entries and resource manager. At the top of the web-page, there are fields to be filled in order to insert a new resource.

After that, there's the playlist date. By default, when a playlist is created, it has the current date, which is not desirable for a producer. To solve this problem, there's a field to insert the desired date.

The table next to the date is the resources list in which a producer can click in the link button next to them in order to add it to the end of the playlist, thus creating a new list entry. Before inserting a resource, a title can be added in the proper field.

Lastly, there's the playlist entries list which have the time of each entry and the when it will be played (in real time).

5.6 Final solution

The theme followed by the prototype was early abandoned replacing blue based colors to gray based colors in memory of the early cinema resources which followed this pattern. The image below shows the homepage which gives the general idea of the altered theme used in the solution.

Figure 11. Homepage of the solution

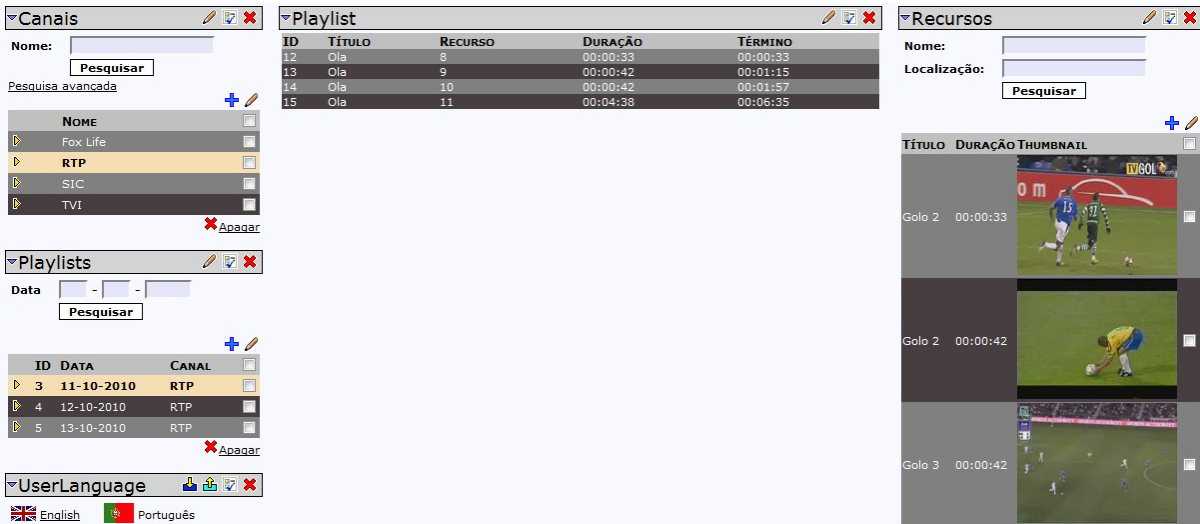


As previously stated, this page has modules deployed in three containers. All of these modules are part of WebComfort meaning that none of them needed an implementation during the development of this project. The choices of the modules were clear: in the left container, there's a treemap of the website to rapidly navigate to the desired page and the language selection below to support the multilingual paradigm; in the center container, there's the login (and registration) module in order to the producer to use the administrative modules; in the right container, there's a calendar to help the producer to remember its present date as it might be important in the configuration of the playlists.

Three (WebComfort) tabs exist by default: the main tab, called "Principal" in the image, the authoring tab, called "Edição" and the WebComfort's administration tab called "Administração do Portal". The other existing tabs (in this case "SIC" tab) consist on the created channels. A viewer can only access to the main and channels tabs as they are the only ones with public access.

The first tab was already explained. The second (and more important) tab, "Edição", which means "Edition", contains the core modules and pages of this solution. Figure 12 shows this tab's presentation.

Figure 12. Tab Edição



The images presented from this point are split as they occupy a large space and become small to fit into the sheet (as observed in the two previous images). These tree containers are going to be explained one at a time and each of them has an enlarged image, which corresponds to crops in the Figure 12.

Figure 13. Left container

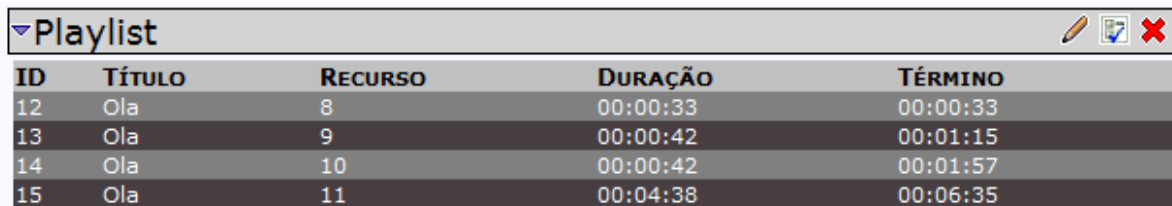


The left container consists in the modules that manage channels, playlists and language support. With the exception of the last module (which was already implemented in WebComfort), the first two have the search option whenever there's data saved in the project's database. This searching option allows the producer to filter the content thus providing a faster way of navigating in large amounts of different data.

In the channel's module, two kinds of search are provided: simple search by channel's name and advanced search by channel's features. The later provides a search by name, genre and main language.

The playlist's module only provides a search by date in which the producer can search a playlist according to its date. This search has numerous combinations as it can filter a whole date or filters dates by days, months, years or combinations of these three, according to what it's typed in the boxes. It's assumed that a null box corresponds to any value (thus, the default null date shows every playlist in the corresponding channel).

Figure 14. Middle container



ID	TÍTULO	RECURSO	DURAÇÃO	TÉRMINO
12	Ola	8	00:00:33	00:00:33
13	Ola	9	00:00:42	00:01:15
14	Ola	10	00:00:42	00:01:57
15	Ola	11	00:04:38	00:06:35

Immediately above is Figure 14, which corresponds to the middle container, and it shows the entries of a selected playlist. It's merely informative as nothing can be changed in this module called PlaylistPreview. This table contains two different durations: the first corresponds to the resource's total time, the second corresponds to when the resource will finish in real time (as explained before). Another table has these similar columns and its meaning is the same as the one presented here.

The final container is in Figure 15, shown in the next page, which is reserved to the resources' entity. Although the image shows the resource's thumbnails, that can be changed by the producer if it's not desirable to be visible. Its search engine only allows to search by name of by location because in this context the main purpose of showing the resources is to allow the producer to visualize which ones are already in the database. A more complete search engine is provided in other context where the producer creates a playlist and needs more information as the duration of the resource, for instance.

Figure 15. Right container



The pencil at the right side of the module's banner corresponds to the preferences of each module. These preferences consist in letting the producer to choose which columns of each table are shown or hidden. For instance, module Channel in Figure 13 only shows the name of the channel. This happens because its preferences are to hide all other columns. In this specific example, the preferences web-page has the aspect shown in the next image.

Figure 16. Channel's preferences

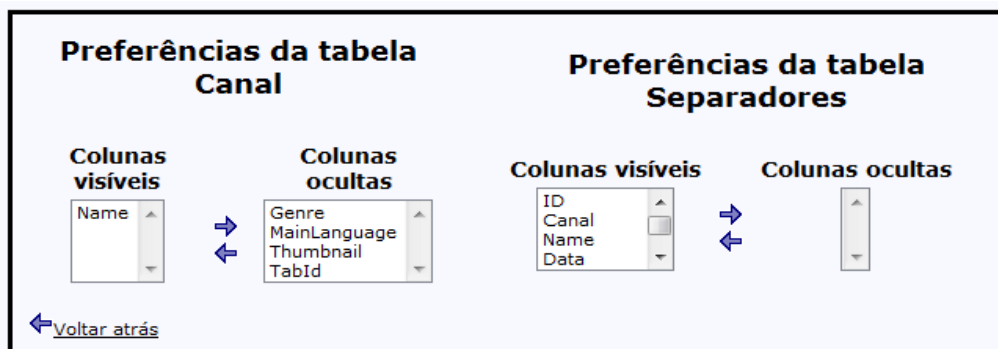


Figure 16 shows the preferences of two tables associated with the Channel module: the channels and its tabs. These are not WebComfort tabs. Instead of it, these are the tabs that are going to be available at the channel's WebComfort Tab menu (explained later).

In the list of visible columns (left list), under the channel's table preferences, only one column is available which means it's the only one to be shown in every channel's table throughout the solution. The hidden list (right list) contains the remaining columns that are not to be shown.

The preferences option helps the producer to hide unimportant details that would turn the tables more confusing and of difficult handling.

5.6.1 Authoring Modules/Pages

The previous section described the common features of all modules, with the exception of module PlaylistPreview (the reason is explained later). In this section and in the next, it will be explained in detail how each module and page work.

The modules have a common function: to show and delete the entries of each table (with the same exception of PlaylistPreview). Add and edit options are available but are just serve to pass arguments that are treated in the modules' associated pages. For this reason, each module has a page which contains all the important operations of creating the entities that are available in the same module.

- **Channel Module/Page**

The project's channels are created in Channel (module) and ChannelEditor (page). The module shows every existing channel according to the given filter (by default every channel is shown). If a channel is created or edited, the associated page opens providing the desired editing features. Figure 17 shows an example of the options available when editing a channel.

The name of the channel cannot be changed and cannot be repeated. Otherwise it would be difficult to distinguish between two different channels with the same name and in real world there is no company with two channels of the same name.

Although not probable nor common, a channel might change its genre or language. These features are also available for changing.

The logotype of a channel is changed from time to time. Therefore it is possible, at any time, to change channel's logo according to necessity. It's also provided a preview feature to help the producer to see the final result of the appliance of the selected image.

Lastly, the channel's tabs edition allows the producer to make navigation tabs that are automatically created when the channel is created. The names of each tab are also unique inside a channel and their content is variable. The producer decides whether the tab appears in the homepage or not.

When the producer selects the option of creating a new channel, a new WebComfort tab is automatically created. This tab has the channel's name and the common modules are automatically loaded. All tabs that were appointed to appear in the homepage are loaded. Later, this topic will be explained in its appropriate context with more detail.

At the bottom of this web-page, there's a list of all existing channels. This allows the producer to edit another channel without leaving this context.

Figure 17. Channel's edition

The screenshot shows a web application interface for channel management, divided into three main sections:

- Edição do canal:** This section contains form fields for 'Nome*', 'Género*', 'Idioma*', and 'Logótipo*'. The 'RTP' radio button is selected. The 'Idioma' dropdown is set to 'Português'. There are 'Procurar...' and 'Pré-visualizar' buttons. Below the fields is a 'Campos obrigatórios' section with a red grid icon and the text 'A dimensão deverá ser 150x150'. At the bottom of this section are two green checkmark buttons: 'Aplicar alterações' and 'Criar novo canal'.
- Edição dos separadores:** This section has a 'Nome do separador' text input, a 'Conteúdo' text area, and a 'Mostrar na página inicial' checkbox. A large blue square represents the content area. At the bottom is a green checkmark button labeled 'Criar separador'.
- Canais:** This section displays a table of existing channels:

	NOME	
▶	Fox Life	✖
▶	RTP	✖
▶	SIC	✖
▶	TVI	✖

A red 'Anegar' button is located at the bottom right of the table. A 'Voltar atrás' button with a left arrow is at the bottom left of the entire interface.

- **Playlist Module/Page**

After the creation of a channel, a producer creates daily playlists to broadcast shows, movies, news, etc. The PlaylistSelect module is responsible for the creation of sequential playlists. If no playlist exists, it creates a playlist with the current date, else it creates a playlist with the date in the next day of the bigger date existing in the playlists of the company's database. This is the only module that creates entities (in this case, playlists) although it creates empty instances, that is, playlists without entries.

In order to edit the playlists, the producer selects the desired playlist and edits it in the corresponding associated page. Figure 18 shows the GUI of this page responsible for the edition of the playlists. This page represents almost all of the effort in this project as it occupied more than 50% of the work done in the whole solution. It provides multiple filters and features for the resources to help the producer to select the desirable ones.

Each entry of a playlist consists of a resource associated to a specific (and mandatory) name. It has a time duration, and ends at a specific hour. If a set of entries has the same name, the solution interprets it as a program. In almost every channel, some programs are repeated a certain number of times. To avoid repeating the creation of the same program, which would waste time, and to prevent the producer to commit mistakes in this repetition, the feature of importing programs is available. It allows the importation of any program saved in the database regardless where it's located. As there can be thousands of programs in an universe of channels and playlists, the producer must specify the channel, the playlist and the name of the program, by this order, to view the resources that were used in this program.

It was focused before that more complete filters for the resources were introduced in this page. This page also allows the filtering of resources by name and location but, as a playlist is being created and time is an invaluable attribute to decide when a resource its played and how much time spends, there's a filter for the minimum and maximum duration of a resource.

Sometimes, a producer might want to create a playlist with resources of a certain theme. To help in the decision of the appropriated resources, there's a filter by topic and keyword, which allows the user to see all resources related with this criteria.

Each set of resources can be insert on top of the existing list or after a selected resource. When this insertion is made all resources update their ending times (with the exception of an insertion at the end of the existing playlist). The same update happens whenever a resource (that's not the last one) is deleted. Figure 18 shows, at the bottom, on the left side, a list of resources and on the right side the list of resources already added in the playlists.

Figure 18. Playlist's edition

Data da playlist: 11-10-2010
Nova data: : : (DD-MM-YYYY) [Alterar data](#)
Título: (Obrigatório)
Link associado:
Hora de começo: 00:00:00
Nova hora: : : [Alterar hora](#)

Importar recursos

Nome **Duração mínima** : :
Localização **Duração máxima** : :
Tópicos
Keywords

Recursos

TÍTULO	DURAÇÃO	IMAGEM
Golo 2	00:00:33	
Golo 2	00:00:42	
Golo 3	00:00:42	
Telerural 1	00:04:38	

[← Voltar atrás](#)

Playlist

ID	TÍTULO	DURAÇÃO	TÉRMINO
12	Ola	00:00:33	00:00:33
13	Ola	00:00:42	00:01:15
14	Ola	00:00:42	00:01:57
15	Ola	00:04:38	00:06:35

Nowadays, almost every channel broadcasts twenty four over twenty four hours and seven days a week, continuously. But some channels with specific themes, such as channels that broadcast debates at a parliament, have a limited schedule. In this example, the playlist doesn't start at 0:00:00 of the current day as it's not business hours. To solve problems similar to this, there's an option that allows a producer to change the starting time of the playlist.

As already stated, the playlist has a date. If the producer wants to change the current playlist's date, it can do that in the appropriate field but it must not match an existing date as it would not be logical to have two playlists broadcasting different programs at the same time in the same channel.

Lastly, an entry might have a link associated. This represents the simple video interactivity which allows a viewer to access an outside website related to the resources that's being broadcast. It's useful specially during the commercials because it helps the viewer to have more information about a certain product or, if it's possible to do it, buy it online.

- **Resource Module/Page**

The last of the Module-Page pair regards the last entity of the solution – resource. Similar to the Channel module, this module doesn't allow creation of any entities. It shows the existing resources but doesn't creates instances of them. For the creation of a new resource the producer selects the appropriate option (the blue plus sign).

Figure 19 shows the presentation of the resource associated edition page. A resource is defined by its location and, therefore, can't be added twice. Any resource, either from the local disk or from a remote location, is saved on a specific folder in the local disk for the extraction of metadata concerning the resource. If the resource is remote, the producer might not want to keep it on its local disk after acquiring the necessary information. In that case, the checkbox that asks about saving the temporary file remains unchecked. Otherwise, the box is can be checked and the resource remains on the local disk.

Any relevant keyword can stochastically be associated to a resource to ease the filtering (already explained). These keywords can be about anything in the resource or related to it. But to define the keyword in an ordered manner, the producer adds topics to it. As previously detailed, these topics have a specific hierarchy defined by the producer.

- **Playlist Preview Module**

This is the only module that only shows information and does not allow any modification in the database. Figure 14, some pages above, shows this information.

The module aims to help the producer to keep track of what the playlists have without the need to open the page regarding the edition of these. In order to alter its entries, the producer must open the playlist editor.

Figure 19. Resource's edition

Título: (Campo obrigatório)

URL do recurso:
 Guardar ficheiro temporário?
 ou
 Recurso no disco:

Aplicar alterações
 Adicionar novo recurso

Edição de keywords e tópicos

Tópicos

0-Raiz

ID	TÓPICO	TÓPICO-PAI	TOPICID	<input type="checkbox"/>	<input type="button" value="+"/>
35	Aventura			<input type="checkbox"/>	<input checked="" type="button" value="Apagar"/>

Keywords

ID	KEYWORD	<input type="checkbox"/>	<input type="button" value="+"/>
1	Pistola	<input type="checkbox"/>	<input checked="" type="button" value="Apagar"/>

Tópicos do filme

ID	ID TÓPICO	TÓPICO	RECURSO	<input type="checkbox"/>	<input type="button" value="Apagar"/>
40	34	Ação	8	<input type="checkbox"/>	
41	36	Violência	8	<input type="checkbox"/>	
42	37	Sangue	8	<input type="checkbox"/>	<input checked="" type="button" value="Apagar"/>

Keywords do filme

ID	ID KEYWORD	KEYWORD	RECURSO	<input type="checkbox"/>	<input type="button" value="Apagar"/>
3	2	Fogo	8	<input type="checkbox"/>	<input checked="" type="button" value="Apagar"/>

Recursos disponíveis

ID	DURAÇÃO	LOCALIZAÇÃO TEMPORÁRIA	<input type="checkbox"/>
<input type="button" value="▶"/>	Golo 2	00:00:33	<input type="checkbox"/>
<input type="button" value="▶"/>	Golo 2	00:00:42	<input type="checkbox"/>
<input type="button" value="▶"/>	Golo 3	00:00:42	<input type="checkbox"/>
<input type="button" value="▶"/>	Telerural 1	00:04:38	<input type="checkbox"/>

5.6.2 Channel Modules

While the previous section gave an explanation about the modules responsible for the creation of the entities, this section covers the modules that appear in the created channels. These are not authoring modules – they just provide information and data. They can be seen as a validation of the previous modules because these show in action the final product: a functional channel with resources to be broadcast.

The following image shows two containers with different modules. On the left it's located the ChannelMenu module which loads the logotype of the channel and its common and created tabs. By selecting a tab, a viewer can see its content on the container in the right side. However, the Playlist module is always present as a viewer might be navigating on the channel and, at the same time, be informed about the current broadcast emission.

The left container hosts all the modules that show the information about the tabs on the right. In this image, there's an example of a live broadcast which allows the viewer to actually see online television. On the top is the playlist shown in its simple view – a view that contains only three records: the previous resource, the resource on air and the next resource. But the playlist can be expanded as shown in Figure 21. While showing the whole playlist in the current day, it also colors the resource that's being broadcasted.

Figure 20. Live broadcast

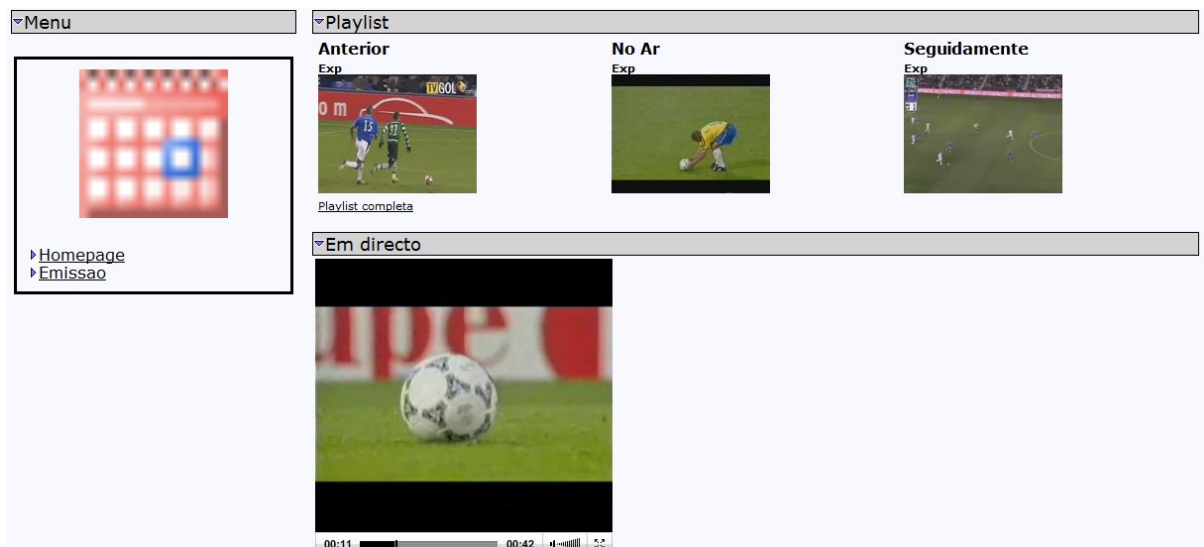






Figure 21. Full playlist

Menu		Playlist	
NAME	THUMBNAIL	TIME24H	
Exp		18:48:03	
Exp		18:48:45	
Exp		18:49:27	
Exp		18:54:05	

6 Preliminary validation

The use cases presented on section 2.1 raised the requirements of the system in theory and section 5 displayed an implementation of a possible solution that fulfills those same requirements. However, theoretical results must be applied in practice to put liability in them.

This section can be seen as a practical conclusion of the project. It will show if the requirements were, in fact, taken into account and solved in the presented implementation. However, by exposing the implementation, some validation was already shown. Therefore, some of the snapshots already presented can be seen as part of the validation process. Those that fit in this section will be referenced throughout the explanation.

Taking the use cases into account, this section will show a set business scenarios that validate this solution. These business scenarios converge to one final objective: to have a functional channel broadcasting resources.

6.1 Business scenarios

- **Scenario A: Create a channel**

In this scenario the producer wants to start using the framework. He wants to create channel “WebChannel” which is a Portuguese channel of general content.

He wants the channel to have two specific tabs: one tab that contains the objective of the channel and it has to appear in the homepage, another tab that contains the location of the company's headquarters.

- **Scenario B: Insert resources**

The producer starts adding resources to the companies database. These resources are both from local disk and referenced from remote machines over the web.

- **Scenario C: Create program “Experience”**

This scenario comprises the creation of a program called “Experience” which starts at the end of one day and ends in the day after. For that to happen, the producer creates two playlists. The first playlist will only pass this program which starts at 23:50:00 and the second playlist has the ending of the program at 00:10:00 on the next day.

- **Scenario D: Visualize program “Experience”**

After creating everything that's needed to broadcast a program, the producer wants to see that program in

WebChannel.

6.2 Execution of the business scenarios

- **Scenario A: Create a channel**

Figure 17 shows the channel edition page which helps in explaining the steps taken to create the channel.

Firstly, the producer clicks on the blue plus link shown in Figure 13, concerning the Channel module. After that channel's edition page appears and he can fill all the fields: name, genre and language. After choosing an appropriate logo, the producer clicks on the corresponding link to create a channel. Next, he goes to the tabs edition section (in the same page) and creates the two wanted tabs and, of the tab with the objectives, checks the checkbox which says for the tab to show in homepage.

Finally, he clicks on the back button to return to the edition's (WebComfort) tab.

- **Scenario B: Insert resources**

Using the same manner as before, the producer clicks on the plus sign in the ResourceAdd module, shown in Figure 15. It will open the resource's edition page as shown in Figure 19.

The producer fills the box concerning the title of the resource and browses an FLV movie from his local disk. After that, he selects the option to create a resource. Below appears two new options which allow to associate to the resource both topics and keywords. He adds some topics and keywords to this resources and does the same to the next ones (he gives different topics and keywords accordingly to the resources content).

In order to add other resources without leaving this page, the producer clicks on the links which says to add a new resource. Again, he puts a name to the resource but, this time, he writes a remote link of an FLV movie and clicks on the option to add resource.

Now the database has two topics both from the local disk and from a remote location.

For the database to be more complete, these steps are repeated so that there are, at least, ten different resources in the database.

At the end, the producer goes to the previous page.

- **Scenario C: Create program "Experience"**

In order to create two playlists, the producer clicks two times in the blue plus sign link on the PlaylistSelect module, also shown in Figure 13. This will create two sequential playlists.

After that, he edits the first playlist by checking the first playlist and by clicking on the pencil next to the blue plus sign. This will open the playlist's edition page as shown in Figure 18.

The producer starts by putting the title "Experience" in the adequate box. He changes the start time to 23:50:00 and filters the list of resources so that only resources with topic "Documentary" and keywords

“Egypt” and “Pyramid” appear.

He starts by adding the resources one by one, adding also a hyperlink to each entry of the playlist. When the last resource reaches 23:59:59 in its ending time, the producer goes back and edits the second playlist.

The same is done but, this time, the producer wants to repeat the resources already added in the first playlist. For that to happen he has to import the resources from program “Experience” from the first playlist. So, selects the channel, playlist and program, with this order, in the section of importing program, and clicks on the import button so that it appears all the resources already added in the beginning of the program in the previous day.

Finally the producer adds again the same resources and goes back to the initial page.

- **Scenario D: Visualize program “Experience”**

This final scenario consist in clicking on the WebComfort tab which is called “WebChannel”. It was automatically created by the time the producer created a channel with that name.

To test the channel, the producer waits (or changes his computer's clock) until 23:50:00.

When the emission starts, he watches the broadcasted resources, clicking in each one to open the correspondent associated hyperlink. Meanwhile he also navigates in the channel to see the tab that shows information about the location of the company's headquarters.

6.3 Discussion

These four business scenarios tested the whole project's features and all four tests were successful. This means that the implemented solution was adequate to the given reference model.

This validation did not take into account some implemented aspects, such as the ability of changing language, because it was not part of the reference model, although it was present in the studied cases. The aim in this execution was to see if the proposed solution fulfilled the main requirements of the iTV paradigm.

The implementation of each entity – channel, playlist and resource – wasn't enough to prove the system's usefulness and correctness as each module had to cooperate with the other. That's why the validation section is of the utmost importance.

Should the validation have failed, the solution would not be prove to be useful in the iTV paradigm.

At the end of the execution, the created channel had the visual shown in Figure 22.

Figure 22. WebChannel's homepage

Principal Edição SIC RTP TVI Fox Life WebChannel Administração do Portal

Menu

Playlist

Anterior Experience



Playlist completa

No Ar Experience



Seguidamente Experience



Mission

The purpose of this channel is to make a validation of the Thesis.

Homepage
Emissao
Mission
Location

7 Conclusions

Despite being an “invention” of this century, the iTV paradigm already has a great number of systems that meet the requirements of a producer/viewer.

The next step would be the creation of standards for the iTV, specifically in terms of playlist creation and resource management. Several online sources (including organizations and companies related to technology and telecommunications) stated that the standards are being discussed, but there are still no published articles about them.

Throughout this study, it became evident that iTV comprises the possibility of a person to publish and visualize television content on a personal computer considering there's an internet access point. It's the attempt to transform the viewer into the producer. This tendency already exists as many of existing systems stimulate the user to create its own channel and create its own resources.

It was also lightly studied how the video interactivity paradigm is growing and creating new possibilities both for the producers and users/viewers.

7.1 Final conclusions

This dissertation shown the state of the art in the iTV paradigm. It studied three cases – WorldTV, WoMa and YouTube – and showed what practical cases already comprised the iTV requirements. It was also concluded that YouTube was the most complete of the three as it fulfilled more requirement than the other systems.

It was seen that when talking in iTV, three entities are fundamental: channel, playlist and resource. These three are cooperate for the creation of an online channel and they are alike in terms of importance.

By developing a solution using WebComfort modules, it was clear that a CMS was a good choice in terms of organization as it possessed containers for the modules to be deployed and interact.

The stated objectives were all met and the proof of that was given in the validation section which assessed the implemented solution in terms of the reference model and desirable features.

7.2 Future work

This project can be used for further work. Some iTV features could be completed and other paradigm could be fully developed over this solution: Interactive TV.

An example of a possible feature to be inserted is the ability to search videos from portals and insert them into the playlists (like WorldTV). Other possible feature to be developed could be the ability to record movies from devices and put them online directly from this framework. These devices could be not only a webcam, as it already exists, but also a cell phone (that would have the ability of sending its videos through SMS to the website [6]; WorldTV already allows it). A direct consequence of this implementation is the possibility of sending live content through these devices (instead of only storing them, it would be better to be allowed to broadcast the content immediately).

In this project, the Interactive TV was almost cast aside, only developing a simple case of Interactive

Video. But to take advantage of the full potential of the iTV, it should be possible to have other kinds of interaction. Examples of other interacting features could be the creation questionnaires in which people could vote and see the results until that moment, the ability to choose the winner of a contest or to participate in contests and the possibility of seeing some events in different angles or perspectives (such as sports events) [13].

Finally, it could be implemented an encoder that would allow a producer to add other kinds of movies other than FLV flash movies.

8 References

- [1] Gilbert Held, *Understanding IPTV*, 2007, Taylor & Francis Group
- [2] Gustavo Cardoso, *Trends and Contradictions in the Broadcasting System: From Interactive to Networked Television*, 2004, Vita e Pensiero
- [3] *7 Things You Should Know About... YouTube*, 2006, EDUCAUSE Learning Initiative
- [4] *SIQuant WebComfort Gestor de Conteúdos e Aplicações Web – Manual do Programador*, 2008, SIQuant
<http://www.webcomfort.org/porta/Home@1.aspx> (Retrieved 23-11-2010)
- [5] *SIQuant WebComfort Gestor de Conteúdos e Aplicações Web – Tutorial para Definição de Css Styles*, 2008, SIQuant
<http://www.webcomfort.org/porta/Home@1.aspx> (Retrieved 23-11-2010)
- [6] Petros Kavassalis et al, *SMS Interactive TV: The Convergence of Television and Mobile Networks*, in *Proceedings. 2004 International Conference on Information and Communication Technologies: From Theory to Applications*, (Greece, April 2004)
- [7] Alberto Rodrigues da Silva, Carlos Videira, *UML, Metodologias e Ferramentas CASE, 2ª Edição, Volume 1*, Maio 2005, Centro Atlântico Editora
- [8] Michael Sattler, *Internet TV with CU-SeeMe*, October 2006, Sams Publishing
<http://sattlers.org/mickey/CU-SeeMe/internetTVwithCUSeeMe/index.html> (Retrieved 23-11-2010)
- [9] Brian C. Fenton, *PC/TV Convergence Will Fracture the Internet*, Popular Mechanics – June 1997
- [10] Yiyang Wu, Shuji Hirakawa, Ulrich H. Reimers, Jerry Whitaker, *Overview of Digital Television Development Worldwide*, Commun. Res. Centre Canada, Ottawa, Ont., Canada, January 2006
- [11] David Waterman, *Internet TV: Business Models and Program Content*, in *Paper for presentation to the 29 th Annual TPRC Research Conference on Information, Communication, and Internet Policy, Washington, D.C.*, October 27-29, 2001
- [12] Rune Hjelsvold, Subu Vdaygiri, Yves Léauté, *Web-based Personalization and Management of Interactive Video*, paper in *WWW10*, May 1-5, 2001, Hong Kong, ACM 1-58113-348-0/01/0005.

- [13] T.D.C Little, D. Venkatesh, ***Prospects for Interactive Video-on-Demand***, in *IEEE Multimedia*, Vol. 1, No. 3, Fall 1994, pp. 14-24
- [14] George Shiers, ***Early Television: A Bibliographic Guide to 1940***, Taylor & Francis, pp. 13, 22., 1997
- [15] Ronda Hauben (2004). ***The Internet: On its International Origins and Collaborative Vision***, Amateur Computerist 12 n° 2, 2004
<http://www.ais.org/~jrh/acn/ACn12-2.a03.txt> (Retreived 23-11-2010)

A1 Feature comparison of some existing systems

Feature	VideoClix	WorldTV	WoMa	YouTube	Livestream	Ustream	Kyle	Justin.tv
Number of Channels	Unlimited	Unlimited	1	1	Unlimited	Unlimited	Unlimited	1
Number of Playlists	None	1	Unlimited	Unlimited	Unlimited	1	1	Unlimited
Video interactivity	Yes	No	No	Yes	No	No	No	No
Video authoring	Yes	No	No	Yes	No	No	Yes	No
Insert subtitles	No	No	No	Yes	No	No	No	No
Upload videos	Yes	No	No	Yes	Yes	Yes	Yes	No
Video reference	No	Yes	Yes	Yes	Yes	Yes	No	Yes
Video portals	N/A	Google, Metacafe, YouTube, Aol, MySpace	WoMa	YouTube	YouTube, Web Server	YouTube	N/A	YouTube
Webcam videos	No	Yes	No	Yes	Yes	Yes	Yes	No
Live content	No	Beta	No	No	Yes	Yes	No	No
Statistics	Yes	Yes	No	Yes	No	No	No	No
Number of Languages	1	8	1	19	1	1	7	20
Target audience	Companies, multimedia professionals	Average users	Martial arts lovers	Average users	Both average and professional users	Companies and users that invest on live content	Average users	Novice users
Access	Paid	Free	Free	Free	Free and Paid	Free	Free	Free
Social networks references	None	Twitter	None	None	Twitter, Facebook	Facebook	Twitter, Facebook, MySpace	MySpace