



MASTER THESIS

"Time-based Video Annotation and Time-based Video Linking - A concept analysis realized with the Joomla! Framework"

By

Luis Ramon Santana Alvarez

(Matriculation Number: 268615)

First Professor
Prof. Dr. Wolfgang Prinz

Second Professor
Prof. Dr. Reinhard Oppermann

Thesis Advisor
Dipl.-Inform. René Reiners

I hereby affirm that this work was created exclusively by me and that no other sources and auxiliary means other than the ones stated were used. All material or text which was extracted literally or analogously from other published work is explicitly stated.

Date

Luis Ramon Santana Alvarez

Abstract

This work presents a tool for extending annotation capabilities on video media objects on the web using Joomla! Content Management System as the development framework.

Nowadays, the amount of video media on the internet is increasing significantly; there are many web sites that use video media as their main instrument of communication, therefore a need for annotation on video media web objects has increased as well.

However, the existing video formats were not prepared to support such features. In addition, video media is time-based, that means the user will retrieve the information over time by watching the video. Therefore, analyzing and extracting information from video media by human perception is a time demanding task in large video repositories.

For the reasons mentioned above this thesis provides a web tool for annotating video media web objects by three means:

1. Users can extend the information on video media by creating a video article using HTML; a video article can only contains one video.
2. Users can create annotations on video articles using HTML, they can define a start and end time for the annotations, the start and end time will define the time that the annotations point inside the video on the video article.
3. Users can create links to video related information inside video articles and annotations; links to related video define the start and end playing time playing for the related video media.

This thesis argues that by searching user generated annotations that incorporate time parameters will help finding information inside video media faster.

During each phase of the development cycle, user testing was conducted in order to identify usability problems.

Preface

This thesis is the final work of my Master study at the RWTH Aachen in Germany; it was developed at the Fraunhofer Institute for Applied Information Technology in Sankt Augustin, Schloss Birlinghoven.

This thesis consists of two parts:

The first part refers to the theoretical framework, background and related work. It also explains the existing problem scope concerning annotations on video media, the existing solutions and finally it provides a comparison between the analyzed systems.

Chapter 1 presents the theoretical framework, in this chapter it is stated the problem concept. It explains the domain and context that led to the creation of this thesis.

Chapter 2 presents the basis needed to understand this thesis. This chapter gives definitions to the concepts used and it presents the used technologies.

Chapter 3 describes related systems. Publicly and under research system were analyzed.

The second part refers to the development process of the generated system for this thesis. It presents the software life cycle used in interactive system and initial steps like requirements elicitation and prototyping; it also shows how the system improved through the use of user testing and the final system, and finally proposes future work.

Chapter 4 refers to the first DIA iteration. This chapter describes requirements elicitation and prototyping.

Chapter 5 refers to the second DIA iteration. This chapter describes web software prototyping, development and user testing.

Chapter 6 refers to the third DIA iteration. This chapter provides solutions for the usability catastrophes identified on the previous DIA iteration

Chapter 7 presents the Envideo system.

Acknowledgements

First of all, thanks to Guadalupe Alvarez (my mom) and my family for supporting me during this stage of my life in Germany.

I would like to thank my tutor Mr. René Reiners for his great support, patience and encouragement; I would like to extend my gratitude to the first professor in grading my Master thesis. Mr. Wolfgang Prinz, and second grading professors and Head of the FIT department Mr. Reinhard Oppermann for their incredible support. Equally important is Mr. Markus Eisenhauer leader of our working group.

I would also like to extend my gratitude to the following persons: Ferry Pramudianto, Ebenezer Rajakumar and Christian Petersheim, my life time friends.

Special thanks to Miss Mahsa Jenabi, Mr. Alex Kempkens and all the Joomla! community.

Also thanks for all that people that helped in an indirect way or that I forgot to mention.

This work is dedicated to all of you.

Table of Figures

Figure 2.1 - Joomla! Front end	6
Figure 2.2 - The Vibelle Home Page	8
Figure 3.1 - The Annotea Framework	10
Figure 3.2 - The Annozilla Plug-in User Interface.....	11
Figure 3.3 – Annotation options in Viddler Web Service	12
Figure 3.4 – YouTube User Interface	13
Figure 3.5 – YouTube Annotation Tools Interface	14
Figure 3.6 – Hyper-Hitchcock User Interface	15
Figure 3.7 – M4Note User Interface	16
Figure 3.8 – EspiritUs’ User Interface	17
Figure 4.1 - First DIA Cycle	21
Figure 4.2 – The DIA Software Life Cycle	22
Figure 4.3 – Envideo Mind Map.....	25
Figure 4.4 – Watch video with Annotations Use Case Diagram	26
Figure 4.5 – Make Annotation Use Case Diagram	26
Figure 4.6 – List Annotations Use Case Diagram	27
Figure 4.7 – Example of Envideo Paper Prototypes.....	28
Figure 5.1 - Second DIA Cycle	32
Figure 5.2 – Video Articles and Annotations Workflow.....	33
Figure 5.3 - Video article and annotations framework.....	34
Figure 5.4 – Prototyping Approaches	35
Figure 5.5 – Creation of Video Articles 2nd DIA Cycle	36
Figure 5.6 – Creation of Annotations 2nd DIA Cycle.....	37
Figure 5.7 – Parts of the Video Related Information Pattern	37
Figure 5.8 - Envideo Database Schema.....	38
Figure 5.9 – Model View Controller Diagram	40
Figure 5.10 - Video Article Listing	40
Figure 5.11 – Return to the Envideos Listing Link.....	41
Figure 5.12 – Time Input Fields.....	42
Figure 6.1 - Third DIA Cycle.....	43
Figure 6.2 – Help Button on the Front end	45
Figure 6.3 – Tooltip in Submit Button	45
Figure 6.4 – Tooltips on the Back end.....	46
Figure 6.5 - Back Button on the Front end.....	46
Figure 6.6 - Annotations Marquee.....	47
Figure 7.1 – Envideo Framework	48
Figure 7.2 – Envideo Component in Joomla! Back end.....	49
Figure 7.3 – Envideo Administration.....	49

Figure 7.4 – Creating a Video Article	51
Figure 7.5 – Annotations’ User Interface	52
Figure 7.6 – Video Related Information’s Command.....	53
Figure 7.7 – Parts of the Video Related Information Command	53
Figure 7.8 – Video Related Information Command Button	54
Figure 7.9 – Search Field	54

List of Tables

Table 3.1 – Comparison of Annotation Systems	19
Table 6.1 - Usability Problems and Proposed Solutions	44
Table 7.1 – User Mapping	50
Table 7.2 – Comparison between Annotation Systems and Envideo	56

Table of Contents

Abstract.....	I
Preface	II
Acknowledgements.....	III
Table of Figures.....	IV
List of Tables	VI
PART I – THESIS DESCRIPTION.....	1
CHAPTER 1 THEORETICAL FRAMEWORK	2
1.1 Introduction	2
1.2 Motivation.....	2
1.3 Problem statement	3
1.4 Project Context	3
1.5 Conclusions	3
CHAPTER 2 BACKGROUND	4
2.1 Introduction	4
2.2 Metadata.....	4
2.3 Annotations.....	5
2.4 Content Management System (CMS)	5
2.5 Joomla! CMS	5
2.5.1 Joomla! Features.....	6
2.5.2 Extension capabilities.....	7
2.6 JW FLV player	8
2.7 Vibelle	8
2.8 Deaf People and Deaf Culture.....	9
2.9 Conclusions	9
CHAPTER 3 RELATED WORK.....	10
3.1 Introduction	10
3.2 Publicly available systems.....	10
3.2.1 Annotea.....	10
3.2.2 Annozilla.....	11
3.2.3 Viddler.....	12

3.2.4 YouTube	13
3.3 Systems under Research	15
3.3.1 Hyper-Hitchcock.....	15
3.3.2 M4Note – MultiMedia Multimodal Annotation Tool	16
3.3.3 EspiritUs'	17
3.4 Discussion.....	18
3.5 Conclusion.....	19
PART II – SYSTEM DEVELOPMENT.....	20
CHAPTER 4 FIRST DIA ITERATION.....	21
4.1 Introduction	21
4.2 Design, Implement and Analyze - The DIA Software Life Cycle	22
4.3 Design.....	23
4.3.1 Requirements engineering.....	23
4.3.2 Requirements elicitation.....	23
4.3.3 Brainstorming, Mind Mapping, and Scenarios.....	24
4.3.4 UML Use cases	25
4.4 Implementation	27
4.4.1 Paper prototypes storyboard-like.....	28
4.4.1.1 Back end prototyping.....	29
4.4.1.2 Front-end prototyping	29
4.5 Analyze	30
4.6 Conclusions	31
CHAPTER 5 SECOND DIA ITERATION	32
5.1 Introduction	32
5.2 Designing.....	33
5.2.1 Workflow.....	33
5.2.2 Annotations.....	34
5.2.3 Prototyping	34
5.2.4 New User Interface Prototype	36
5.2.5 Database schema	37
5.3 Implementation	38

5.3.1 Video player	38
5.3.2 Delivering video on the internet	39
5.3.3 Envideo development	39
5.4 Analyze	40
5.4.1.1 Backend administration	41
5.4.1.2 Front-end problems	41
5.5 Conclusions	42
CHAPTER 6 THIRD DIA ITERATION	43
6.1 Introduction	43
6.2 Designing.....	44
6.3 Implementation	44
6.3.1 Marquee functionality	47
6.4 Analysis	47
6.5 Conclusions	47
CHAPTER 7 PRESENTATION OF THE ENVIDEO SYSTEM.....	48
7.1 Introduction	48
7.2 Objectives.....	48
7.3 Envideo User Interface.....	49
7.3.1 Envideo Back end	49
7.3.2 Envideo front-end	50
7.4 Envideo User Profiles	50
7.4.1 Authors of video articles	50
7.4.2 Authors of annotations	50
7.4.3 User that search and retrieve information	50
7.4.4 User Mapping.....	50
7.5 Creation and extension of content in the Envideo system	51
7.5.1 Creation of video articles	51
7.5.2 Annotations.....	52
7.5.3 Link to video related information using time parameters.....	53
7.6 Information searching and retrieval	54
7.7 Fast information searching and retrieval inside video media.....	54

7.7.1 Case Scenario	54
7.8 Advantages.....	55
7.9 Disadvantages	55
7.10 Comparison with related systems.....	56
7.11 Conclusions	56
Envideo Quick Start.....	66

PART I – THESIS DESCRIPTION

This part of the thesis provides the theoretical background. It also explains the existing problem scope with annotations on video media, current approaches and the proposed solution.

Chapter 1 answers the question of why extending annotation by incorporating time parameters and what are the benefits of the proposed approach.

Chapter 2 gives an introduction to the basis needed to understand this thesis work.

Chapter 3 analyzed the current solutions for annotation.

CHAPTER 1 THEORETICAL FRAMEWORK

1.1 Introduction

Metadata is the “data about the data”, in the past, metadata was used and created only by professionals, nowadays almost any user can create metadata (and many times without knowing it) for example by using YouTube or Facebook. Annotations are a kind of metadata, people have used annotations in their basic form of paper and pen for long time in order to extend, correct, clarify and search documents.

There has been an extensive use of video media over the internet in the last years; one example is the video streaming service from YouTube, the increase in amount and quality of media on the internet has brought the necessity for annotation in order to be able to search, classify, etc. The video objects, however, the annotation capabilities on video are rather new, this is due to the video formats were not prepared to support annotations. A lot of research has been performed in order to incorporate annotation features, and there are many systems nowadays that provide the possibility of annotation. Some of them are commercial and the companies behind such systems are supporting annotation features even more. Nevertheless there is a lack of attention to time parameters for annotations on commercial systems, this thesis work considers this lack a big problem because video media is time-based, and this means that the user gets the information through time.

This thesis describes a prototype system developed at Fraunhofer FIT, which enables the annotation of video objects on the web with stress on time parameters for the annotations. By using the system, Users can annotate videos on the web asynchronously, collaboratively and they are able to define at what time and for how long each annotation is relevant on the video. The users are also able to define related video media by pointing only to the relevant parts of the related videos.

With the previous features users are able to discuss whole videos, or only certain parts (if the discussed video is too long), and link to other video media without the need of watching completely the related media thus saving time in getting related information.

1.2 Motivation

Users have now an active role in the internet community. They are generating and retrieving information as an everyday part of their lives. Nowadays, users publish information on the internet by using their mobile phones. The impact of the information generated by users is enormous.

User generated information varies from text, pictures and video media. Users are mapping their mental concept from the physical tools like agendas, pen and paper to digital devices in order to help them complete their tasks. One of the most important and frequently used tasks is the annotation of content. Annotations are generated by the users, they are usually small but their content is semantically high.

Also annotations are usually placed near the information they are referring, therefore it is argued in this thesis that annotation on video media must incorporate time parameters to define the position of the annotation on the video and moreover including time parameters when linking video related information.

This master thesis is an approach to map the mental model from the users' point of view of annotation in order to apply it on video media. Its aim is to provide an easy tool for video annotation that lets users include time parameters in order to be usable for fast searching and retrieval of information inside video media.

1.3 Problem statement

The amount of video media used on the web has increased dramatically. As an example the Vibelle project [WR-9] is a web portal for deaf people that use video media as its principal mean of communication. Video portals like Vibelle have the need to include an annotation system for video media in order to gather information from their users. However, there is actually no video annotation system for the web for easy installation and deployment.

Moreover video media information is time based and with the existent systems from other platforms there is no possibility to include annotations that supports HTML in conjunction with time parameters for video.

In addition there is currently no system that supports linking to video related information that incorporates time parameters for controlling playing functionality on the related video. This thesis work argues that by searching inside annotations that point to video related media with time parameters will help user to find information inside video media faster.

1.4 Project Context

The outcome of this Master thesis is a web application designed to work as an installable component for Joomla! CMS. It is developed to be used by the first time on the Vibelle project at Fraunhofer FIT. However, the developed component is aimed to be a general video annotation solution for any web site that uses Joomla! CMS.

1.5 Conclusions

This Master thesis aims to provide a web application that incorporates time parameters in the annotations and links to related information. Therefore users will be able to annotate video media by defining the time of the annotations. Users can position the annotations on the video.

The system uses the generated annotations for fast search and retrieval of information. By searching annotations that incorporate time parameters, users will find semantically relevant information of annotations inside a video.

In the following chapter the technological basis of the system presented in this work will be introduced.

CHAPTER 2 BACKGROUND

2.1 Introduction

This chapter describes the concepts, technologies and existing approaches concerning content annotation. In order to better understand how this thesis is trying to solve the problem, it is important to identify and clarify how the diverse parts interact with each other.

This thesis starts by defining the basic concepts like: Metadata, Annotations and Content Management System. These concepts are widely used through this master thesis therefore a basic understanding is required.

Following is a description of the used technologies, like: Joomla! and JW FLV player, these are the two main technologies used for this work.

Finally, an explanation of the underlying ground work that gave the idea for developing this Master thesis is given.

2.2 Metadata

A generic term for metadata is: “The data about the data” that means metadata is data that “talks” about other data, so, for example if we have a book, the information that is inside the book is the data, the metadata in this case is the name of the book, the ISBN, the number of pages, the author and the annotations made on the book by the reader [Baca, 2008].

In the past, metadata was created only by professionals, however, nowadays with the expansion of the internet and the increasing amount of digital media [Marsden, 2006]; many users are creating metadata, many times without even realizing it. There are several functions of metadata, just to name some: it helps in extending content, fast information searching, information retrieval, metadata harvesting, description of resources, cataloguing and identification. An example of metadata is when users publish their videos in YouTube, they can include a description of the videos, select a predefined set of categories and define tags for the videos, all the information users add to the videos are Metadata.

In this thesis work metadata is used for extending content, fast search and information retrieval. This thesis uses metadata generated by non-expert users in order to search information video data.

Metadata can also have different semantic levels [Goularte, 2004]. For example, in a picture a low semantic level of annotation is the size of the picture, date and level of exposure; on the other hand a high semantic level is the description of states or naming of persons shown on the picture.

There is a lot of debate about the definition of metadata which is beyond the scope of this work, but for a more complete guide on metadata refer to the book from Murtha Baca, “Introduction to metadata” [Baca, 2008]. This thesis will follow the terms used in the mentioned book in order to have clear guideline and a point of reference at the moment of explaining concepts concerning metadata.

2.3 Annotations

Annotations are semantic metadata [Baca, 2008], consisting of free-text notes, and they do not follow a static format for saving information. Annotations are usually used for extending content; these are also generated mainly by non-professional users who want to extend or clarify certain information, a common way of annotation is the written annotation.

In this thesis work a tool was developed that allows the users to make annotations on video media. This means that the user is able to extend or clarify the content that is inside videos. Annotations make use of HTML, thus having the possibility to incorporate text, images, links, links to other videos, links to videos from Google and all the possibilities offered by HTML.

This work provides the users with the necessary tools to create annotations with time parameters for video media. The system uses the annotations made by the users in order to search for specific information inside video media. Searching inside video media is accomplished by using the time parameter of the annotations. Users searching inside annotations will in consequence find also information inside video media due to their time parameters.

2.4 Content Management System (CMS)

CMS is an acronym for “Content Management System”. A CMS supports users to administrate information. This can be a catalog from a bookstore, a bank system or an internet portal; there are several categories for CMS, to give some examples there are: Enterprise Content Management Systems (ECMS), Document Management Systems (DMS) and Web Content Management Systems. Actually it is difficult to define the type of a CMS, because of their variety and functionalities. However, Joomla! CMS is well-defined inside the group of Web Content Management System (WCMS) because it helps to administrate content on the web.

A WCMS (see above) helps users to create, classify, edit, delete and publish electronic content on the web like text, images or video; for that, users usually do not need technical knowledge. A Web CMS generates web pages dynamically so the user does not have to generate static Web pages for the content. Moreover a Web CMS stores the content in text files or more complex information systems like Database Management Systems.

The prototype presented in this thesis makes use of the Joomla! CMS for administration of information and development framework, Joomla! CMS (cf. 2.5) was chosen due to its stability and extension capabilities and open source community for the development framework.

The developed component is part of a set of future requirements for the Vibelle project. The Vibelle project also makes use of Joomla! therefore the development of the prototype will ensure a future reusability.

2.5 Joomla! CMS

Joomla! is a very popular Web Content Management System, it is an open source project licensed under GNU GPL [WR-11], Joomla! is written in PHP programming language and it uses MYSQL as its supported

database system, it is extensible and it is considered one of the top 50 most important open source projects [WR-13].

Joomla! CMS consists of a front end and a back end:

On the front end users interact with the information (figure 2.1), depending on the user privileges, users are able to access, add, edit, delete or publish information.

On the back end is the administration of Joomla!. The back end provides configuration, user administration, content administration, administration of new components and configuration of the navigation for web sites made in Joomla!

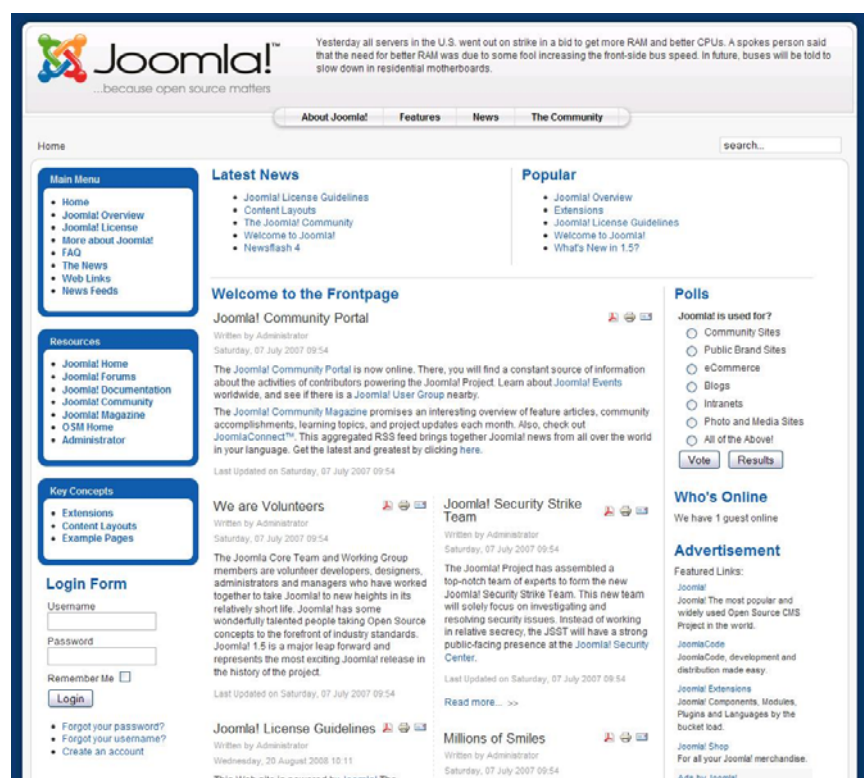


Figure 2.1 - Joomla! Front end

2.5.1 Joomla! Features

Joomla! is a professional and stable platform. It is oriented to people that publish information on the internet, for example: web developers and designers. It is also a solid platform for development. Joomla! provides the following functionalities:

Content Management

Joomla! Uses a fixed hierarchy for content, which consist of section, category, and articles for content items, all the content of a Web site made in Joomla! is handled within this hierarchy.

User Management

Currently Joomla! has a hierarchical structure for user management (it is envisioned to be changed on the new version of Joomla!, refer to [WR-14] for more information). Users are divided by two levels of access: front end and back end (cf. 2.5). Back end users have higher permissions in Joomla! CMS.

2.5.2 Extension capabilities

Joomla! is extensible, users of Joomla! can extend Joomla! to fit their needs by installing new components, modules, plug-ins, templates, languages and tools.

Components

Components are complete systems, Joomla! comes with banners, contacts, news feeds, polls, search and web links components. Moreover the flexibility of Joomla! allows the installation of new components.

Modules

Modules are small pieces of code that deliver content to the front end; an example of a module is the “Latest News” module.

Plug-ins

Plug-ins are small pieces of code that modify the information before is presented, there are different kinds of plug-ins and they usually work with events.

Templates

Templates modify the “look and feel” of a Web site made in Joomla!, with templates users can modify the graphics, colors and layout.

Multilingual support

Joomla! supports the installation of language packages, therefore users of Joomla! are able to create web sites in different languages by installing language packs.

Joomla! Framework

The developers Joomla! 1.5 (the latest version at the moment if this writing) designed a totally new object oriented PHP framework compared with its predecessor Joomla 1.0.x. The Joomla! CMS itself sits on top of the Joomla! Framework.

Web developers can use the Joomla! Framework to generate extensions for Joomla!, it is a fast and stable.

This thesis work use Joomla framework as its development environment and Joomla! CMS as it deploy platform.

2.6 JW FLV player

The Vibelle project (cf. 2.7) uses the JW FLVplayer because its video information is in FLV video format. According to the author of the video player: “The JW FLV Player is the Internet’s most popular and flexible media player [WR-15]”.

The JW FLV player is configurable and extensible; it includes skinning functionalities, plug-in architecture and Javascript API (Application Programming Interface)

This thesis work uses the FLW FLV player, because the video player provides the Javascript API which allows controlling the video functionalities. This thesis work requires control on play, and video-seeking functionalities.

2.7 Vibelle

Vibelle [WR-9] is an acronym for “Visuelles zu Beruf, Leben und Lernen (English translation: Visualization for Job, Life and Learning) [WR-10]”. The leader of the project is the DESIRE (Deaf and Sign Language Research Team) research team [WR-18]. The DESIRE group is established in Aachen Germany. The Vibelle project is being developed by Fraunhofer FIT [WR-19]. Fraunhofer FIT is a sub-partner from DESIRE and its tasks are the development and support of the Vibelle platform.



Figure 2.2 - The Vibelle Home Page

Vibelle uses Joomla! as its content management system. Vibelle is oriented to deaf people. It has three main goals: 1. it provides an E-Learning platform 2. It supports deaf people to incorporate them in the Job Market and 3. It is a medium of communication for the German deaf community.

The Vibelle project is an effort to make smaller the gap between the hearing people culture and the deaf people culture.

Vibelle is a video intensive web portal due to the reading and writing limitations of deaf people (cf. 2.8), in addition there are a lot of media content being generated for the Vibelle project therefore the necessity for annotation arises, the deaf community would like to have the possibility to make annotations on the content that is being presented also searching information inside video media is practically impossible using the current platform.

The tool developed in this thesis aims at being a general annotation solution for any website that uses Joomla CMS; however, it is envisioned that the Vibelle project will be the first real and concrete place where the system will be deployed.

2.8 Deaf People and Deaf Culture

Deaf people communicates with sign language, this means that they communicate using hand gestures, therefore sign language is the first language deaf people usually learn, therefore their main mean of communication, on second place they learn to read and write the language that is being spoken locally in the place they are living, in addition there is no universal sign language so, for example, deaf people in Germany use the DGS (acronym for “Deutsche Gebärdensprache” and means “German Sign Language”) which is different from the ASL (American Sign Language).

Deaf people learn the spoken language like their second language (in reading and writing form), therefore they are not so fluent using it. As a result a 17-18 year old deaf teenager has a reading level of 4th grade (according to the USA system). According to the DESIRE group, German deaf people have also the same problem.

For the previous mentioned reasons a web portal that uses only written language is slow and difficult to understand for deaf people, therefore it is better for the deaf community to watch video media that makes use of sign language instead of reading text on web portals designed for them [Henderson, 2005].

2.9 Conclusions

The Vibelle project makes intensive use of video media because with the help of videos deaf people can communicate in their language (sign language).

The content managers of Vibelle, the DESIRE group (cf. 2.7) have expressed the interest to have an annotation system that let them add information to the created video media; this thesis takes the required annotation capabilities from Vibelle in order to build a new way of annotation. However, the annotation system developed in this thesis work is envisioned to be use as a general solution for hearing and deaf people not only for the deaf community.

The following chapter analyzes commercial and in-research systems that provide annotation capabilities.

CHAPTER 3 RELATED WORK

3.1 Introduction

This chapter analyzes existent annotation systems, it gives an outlook to the most important features, their advantages and disadvantages, and how they work. It makes a division between systems that are public available and the ones that are under development. At the end of this chapter a comparison between the different features provided by the analyzed systems is given.

The goal of this chapter is to find what is still missing in the area of video media annotation and what can be improved.

3.2 Publicly available systems

This section presents three systems, the presented system have the characteristic that are available for public use. These projects have the support of companies; they are continuously improved and widely used.

3.2.1 Annotea

“Annotea enhances collaboration via shared metadata based Web annotations, bookmarks, and their combinations” Annotea [WR-3] is an approach from the W3C [WR-12] for the implementation of annotations, the W3C provides a free test annotation service [WR-6] and there are third party companies that have generated clients that access the Annotea service, examples of these clients are Amaya [WR-4] (being the first client to be developed) and Annozilla (cf. 3.2.2). The annotation is based on the content of the web page, which means that the user can “author” a web page. The W3C also provides the annotation server for download in its web site so the users can generate their own annotation servers; this is the recommended approach by the W3C. This thesis work presents Annozilla as one of the clients.

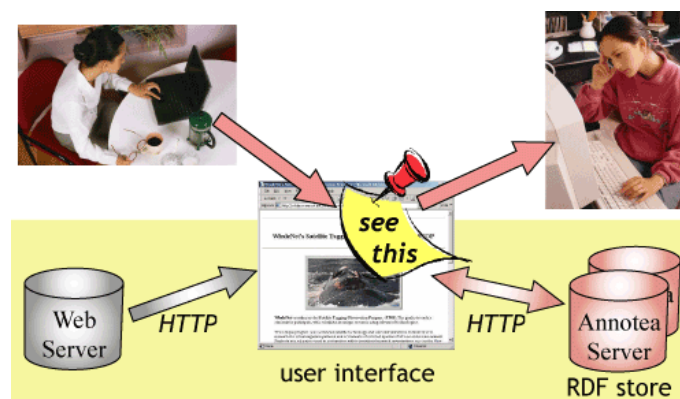


Figure 3.1 - The Annotea Framework

3.2.2 Annozilla

Annozilla [WR-5] is a plug-in for the Firefox Web browser; it works as a client for the annotation service from Annotea server, Annozilla can be connected to any server that has the Annotea (cf. 3.2.1) service installed, however, a typical approach is to connect Annozilla to the test server from Annotea.

The Annozilla client is installed by default like a sidebar on Firefox (the configuration can be changed), the user must be logged in order to start making annotation, if the user wants to make an annotation on a web page, he must select the part he wants to annotate then choose the option “Annotate current selection” from the context menu, all the annotations will be displayed on the sidebar (if Annozilla is configured like a sidebar).

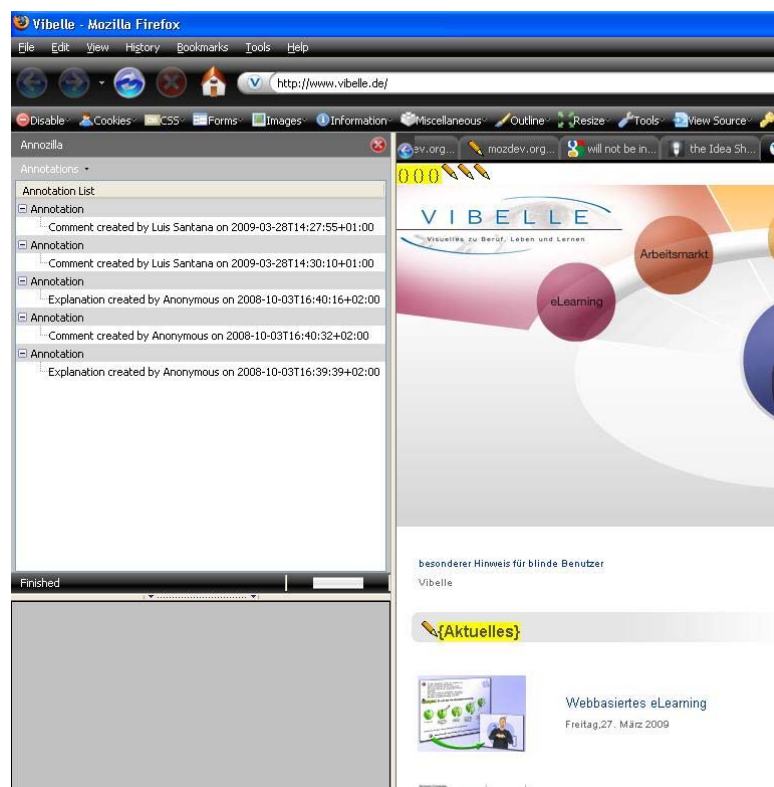


Figure 3.2 - The Annozilla Plug-in User Interface

Annozilla uses the philosophy of “authoring a web page” that means when the user is making an annotation, the annotations will be highlighted and an icon will appear on the side of it, so the user can see exactly where are existent annotations on a web page, unfortunately this kind of annotation changes the layout of a web page.

The main disadvantage of annotations made by Annozilla is that they modify the look and feel of web pages; therefore web designers (that usually spend hours working in the design of a web page) do not like Annozilla [Fernandes, 2005].

Another disadvantage is that the annotation of media objects is not possible and last that users cannot incorporate images on their annotations.

Users cannot incorporate images on their annotations

3.2.3 Viddler

Viddler is a web portal [WR-2]. It is a free and open video service for public use. It is based on the same philosophy like YouTube (cf. 3.2.4) in which the registered users can upload and share their videos on the internet. Viddler also incorporates community functionality, this means that the users from Viddler can create groups or contact other users.

Viddler incorporates annotation capabilities; the user can annotate videos in several ways:

Users can add comments to the videos, only text and Web links are allowed, the comments made by the users will be displayed by the player at the moment the comments were made.

Users can add tags (Metadata) for fast video searching.

Users are able to add video comments; however, users are only allowed to upload recorded videos from a webcam.

Users can also add a response to a video. They are able to annotate with text or the allowed HTML tags ``, ``, `<a>` and `<blackquote>`.

In addition, the developers of Viddler have also made a reduced version of the Viddler API public available (based on request though); therefore it also has extension capabilities.

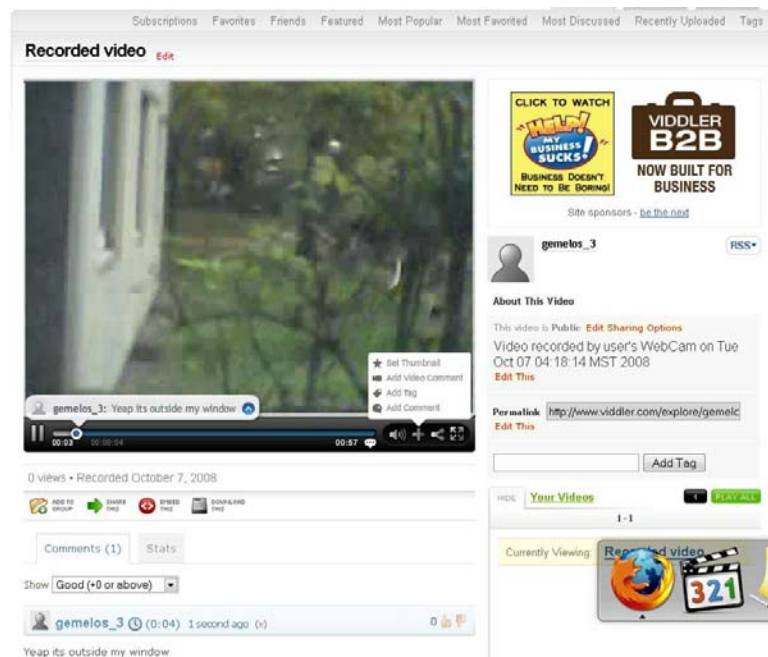


Figure 3.3 – Annotation options in Viddler Web Service

A clear disadvantage is that the only annotation option that incorporates a time parameter is the “add comment” option, and it is setup automatically at the moment of adding a comment.

Another disadvantage of Viddler is that users can add video related information but only video media recorded by a webcam and cannot select another video from the Viddler server. Time parameters are inexistent in this option.

3.2.4 YouTube

The Web service from YouTube [WR-1] is the most well-known public video repository in the world. It is free and open for public use. Alexa [WR-6] is a company that measures the traffic of websites on the web. According to Alexa statistics, YouTube reaches the rank number 3 on the global top sites of the web.

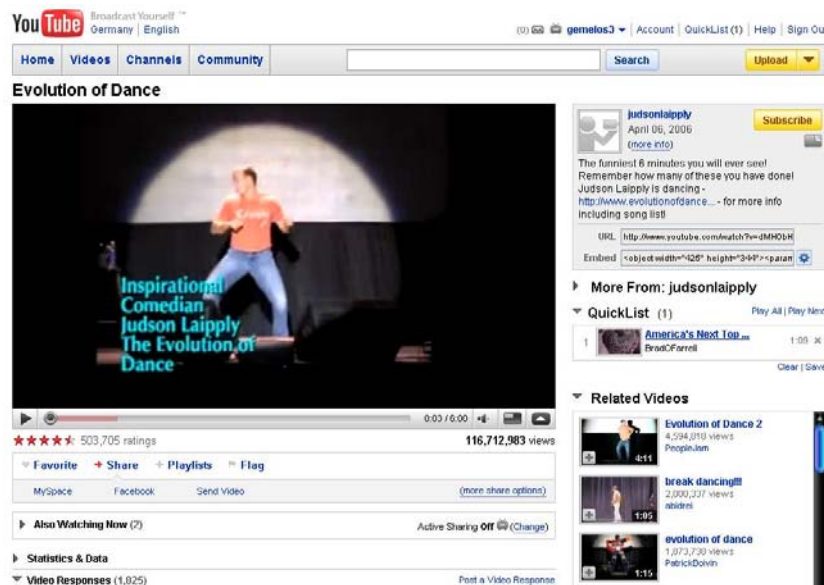


Figure 3.4 – YouTube User Interface

The users have the option to register for free in YouTube, the unregistered users can only see the videos that are published, and on the other hand registered users have the following options:

The most basic option is the YouTube allows users to upload their videos in YouTube. There is also the possibility to make videos public available or private; when users make their videos public then other YouTube users can watch them.

YouTube provides an option for sharing videos on the internet. Users can share their videos by sending an email with the link and a small description of the video.

Users of YouTube are also able to create playlists. Playlists are lists, in which the users of YouTube can place videos for later view. There are three kinds of playlists Normal Playlist, Quick lists and Favorites Lists.

And finally users of YouTube are able to subscribe to other users, and receive automatic notifications when new videos are uploaded.

The users from YouTube can annotate videos in two different ways; one is by annotating videos on the main interface and is available for all users and the other possibility is only available for the users that are publishing their own videos and consists in annotating directly on the videos by using graphical tools.

The annotation on the main interface consist of a small text box in which users type their annotations.

Users that upload their own videos have the possibility to use graphical tools for annotations.

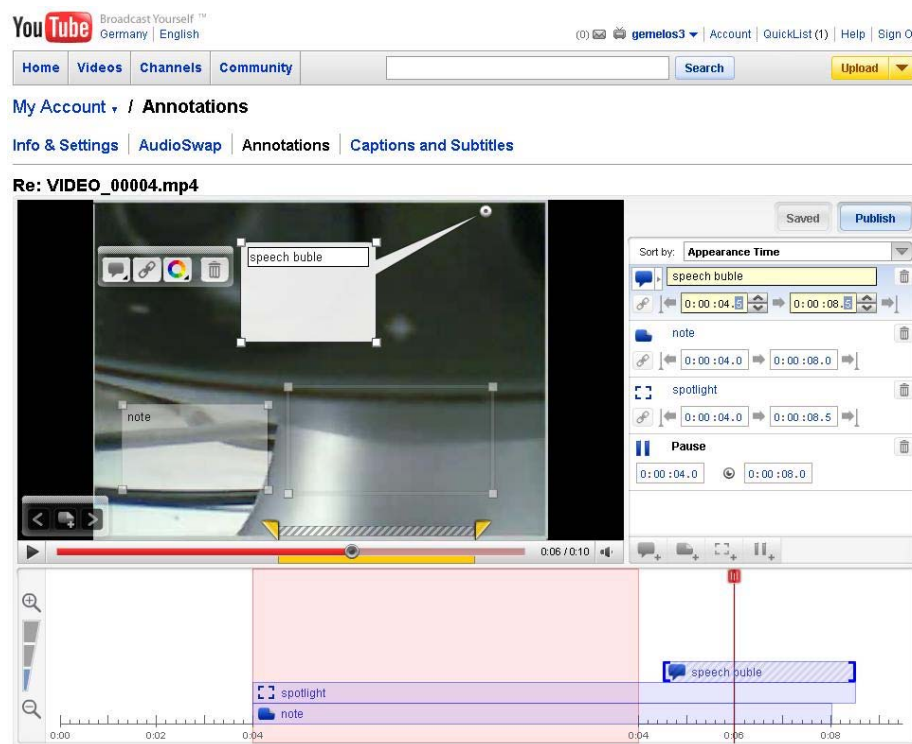


Figure 3.5 – YouTube Annotation Tools Interface

YouTube provides the “speech” graphic tool. Users can insert a speech “bubble” graphics that will appear during the playing of the videos. The “speech” bubble tool incorporates start and end time parameters in order to define its time of appearance.

YouTube also provides Note and Spotlight graphical annotation tools; the only difference between them is the graphic that will be displayed at the moment of playing a video. Note and Spotlight tools also incorporate start and end time parameters.

The last annotation tool for the back end is the pause tool; this tool inserts a pause in the video for a period of time defined by the users.

All the above tools (except the pause tool) share the following functionalities: linking to other videos, to other groups, channel, compose message, group, collaborative annotations, video response upload and YouTube search query.

When the annotations are done, the users (registered and unregistered) will see these graphical tools appearing on the video at the moment defined in the time parameters.

A disadvantage of YouTube is that annotations made on the main interface do not incorporate time parameters like the graphical tools.

Another disadvantage is that time parameters on related video information are inexistent.

3.3 Systems under Research

System under research are usually prototypes; they are results of research projects like thesis work, they are not available to public use and their continuity is uncertain, however, they are interesting to analyze due to their innovation.

3.3.1 Hyper-Hitchcock

This is a web tool for editing videos; it allows the user to create Hypervideos (the resulting videos from this system are called Hypervideos), an Hypervideo is a video that is composed of clips from several videos, and it is not linear, this means that at some point when Hypervideos contain links to other videos, the users can decide continue watching the main video or the clips that have video related information [Shipman, 2005].

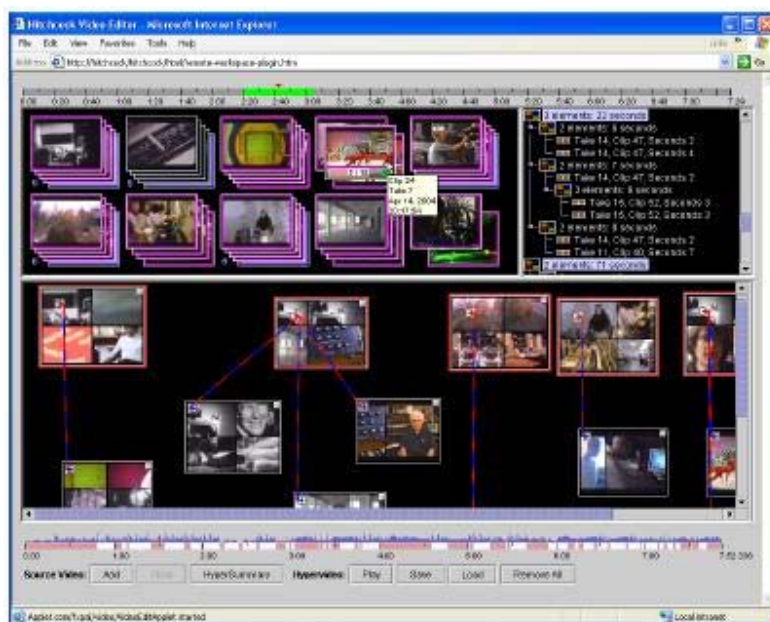


Figure 3.6 – Hyper-Hitchcock User Interface

The Hyper-Hitchcock system User Interface works as follow: it presents on top of the screen several composites, and each composite represents a linear video, or a video decomposed in several clips, so we can think of composites like small linear video clips, linear video clip means a video that plays from start to end in contrast with Hypervideos that can have many directions at the moment of playing not only one. On the bottom of the User Interface are the links between the “composites”, this area is called “the authoring space” in which the user generates the Hypervideo, so for example in this area is where the user makes links between the composites resulting in Hypervideos.

Another example is when there is a Hypervideo about a city tour, then the system presents three direction options to the user at some point (with the help of icons), and depending on the option of the user, the next playing video is different.

The disadvantage of the Hyper-Hitchcock system is that the user can only place descriptions for the links between the composites so it is not suitable for longer descriptions, therefore if a user is watching a Hypervideo and does not know the context of the Hypervideo, jumping between the Hypervideo will be meaningless.

3.3.2 M4Note – MultiMedia Multimodal Annotation Tool

M4Note is a system developed in Java for the TablePC, like its name implies it is designed to allow the user to annotate video media by using a multimodal interface, the interface provides the following tools for annotation: electronic ink, text input, voice input, text recognition (for the electronic ink), symbol recognition and voice recognition. When an annotation is done an XML file is generated, thus giving the flexibility to edit the annotations in the future or playback the video with the annotations [Goularte, 2004].



Figure 3.7 – M4Note User Interface

The M4Note system works with two kinds of annotation:

Inclusion of tags

The user can enter personalized tags for fast searching and indexing.

Extension of content

The user enters information that describes the videos by using any of the techniques above mentioned.

Users have three options available to make annotations: create an annotation from an existent video, edit an existing annotated video or make an annotation when recording a video.

The system works as follow: In order to make an annotation, the user tips the video screen with the electronic pen, this pauses the video recording and makes a snapshot from the video, then the user can make annotations with the electronic pen or via voice, the system uses voice recognition.

The disadvantage of the M4Note system is that users cannot make links inside the video to other time on the same video, this is important in long time recordings, and also the system does not allow the user to make link to other video related information.

3.3.3 EspiritUs'

EspiritUs' is a system that is composed of a Web server that provides annotation services and a plug-in for Internet Explorer browser. The system allows the user to add public and private annotations; it also lists annotations and search inside the annotations [Fernandes, 2005].



Figure 3.8 – EspiritUs' User Interface

The EspiritUs' system works as follow: after the user is authenticated, and a web page is loaded the public annotations and the annotations made by the logged user will appear on the sidebar, the "add", "edit" and "delete" options will be display on the plug-in as well.

The EspiritUs' system will not modify the "look and feel" of web pages; all the processing of the annotations will be using the sidebar.

The disadvantage of EspiritUs' is that the user cannot annotate video web objects, it annotates web pages like a whole, therefore features like video linking, timed annotations are not available.

3.4 Discussion

From the analyzed systems, Annozilla and EspiritUs' work on the same philosophy, this means both consist of a Web server that provides annotation services and a client, Annozilla is a client for the Mozilla Firefox browser and EspiritUs' a client for the Internet Explorer.

The installation of Annozilla client is impossible without searching for help on the internet [WR-8] on the other hand, the developers of EspiritUs' argue that the integration of their system in a web page is simple because no installation is required and that the users may use their favorite browser, this is simply not true, because if their philosophy is to work through a plug-in then an installation with administrator rights is required, moreover the statement that the users can use their favorite browser is partially true because a plug-in must be available for every platform.

Regarding User Interface the system that excels by its simplicity is Viddler, it is the easiest to use from the analyzed system, and after logging in, the users only need one click in order to make an annotation. On the other hand we have more complex options for video annotation with Hypervideo, YouTube and M4Note. A last point concerning annotations on the discussed systems is that they do not let the user specify pictures in their annotation system.

The discussion concerning time parameters show that in Annotea and EspiritUs' time parameters are inexistent, Viddler only allows to define two time parameters: the first one is used to seek on the video, and the second parameter is the time when the annotation will be displayed, this is taken automatically at the moment of adding an annotation, on Hypervideo the time parameter will define at what time we can watch related video information, M4Note is similar to Viddler in a way that the time parameter will only tell the time at which the annotations are going to be display. YouTube has more complete options for defining time of annotations, the users can define a time for seeking video and in the annotation tools the user can define start and stop time of appearance; however, these parameters are only available for the visual annotations not on the text annotations.

The table below compares some of the characteristics from the analyzed systems that are considered as being important for this thesis:

Characteristic	Annotea	Viddler	YouTube	Hyper-Hitchcock	EspiritUs'	M4Note
Annotation of video objects	No	Yes	Yes	Yes	No	Yes
Annotation of web content	Yes	No	No	No	Yes	No
Annotations can point to a certain time on the video	No	Yes	Yes	Yes	No	Yes
Possibility of using HTML on the annotations	No	Partially	Partially	No	No	No
Possibility to setup the start and end time of the annotation	No	No (Only the beginning time)	Partially (Only on graphical annotations)	Yes	No	No
Annotations can point to videos	No	Partially (Only to videos recorded on webcam)	Yes	Yes	No	No
Annotations can point to other videos and define the start and stop time on the linked videos	No	No	No	No	No	No

Table 3.1 – Comparison of Annotation Systems

3.5 Conclusion

After testing several systems, it is clear that all of them lack or have reduce time parameters for their annotations, this is hard to understand due to the nature of videos in which the user get the information through time, moreover time parameters on related video information is inexistent.

The next chapter introduces the system developed in this Master thesis. It will describe how the presented system overcomes the limitations of the actual systems taking care of not involving the user with a complicated User Interface.

PART II – SYSTEM DEVELOPMENT

This part of the thesis explains how the system was developed, the development of the system followed the DIA cycle user-centered design therefore this part is divided in three chapters because the development of the system that is presented here was elaborated in three cycles and each cycle corresponds to one chapter.

Chapter 4 corresponds to how the idea was conceived and the user requirements of the system

Chapter 5 corresponds to definition of technologies, first software prototypes, and small user testing

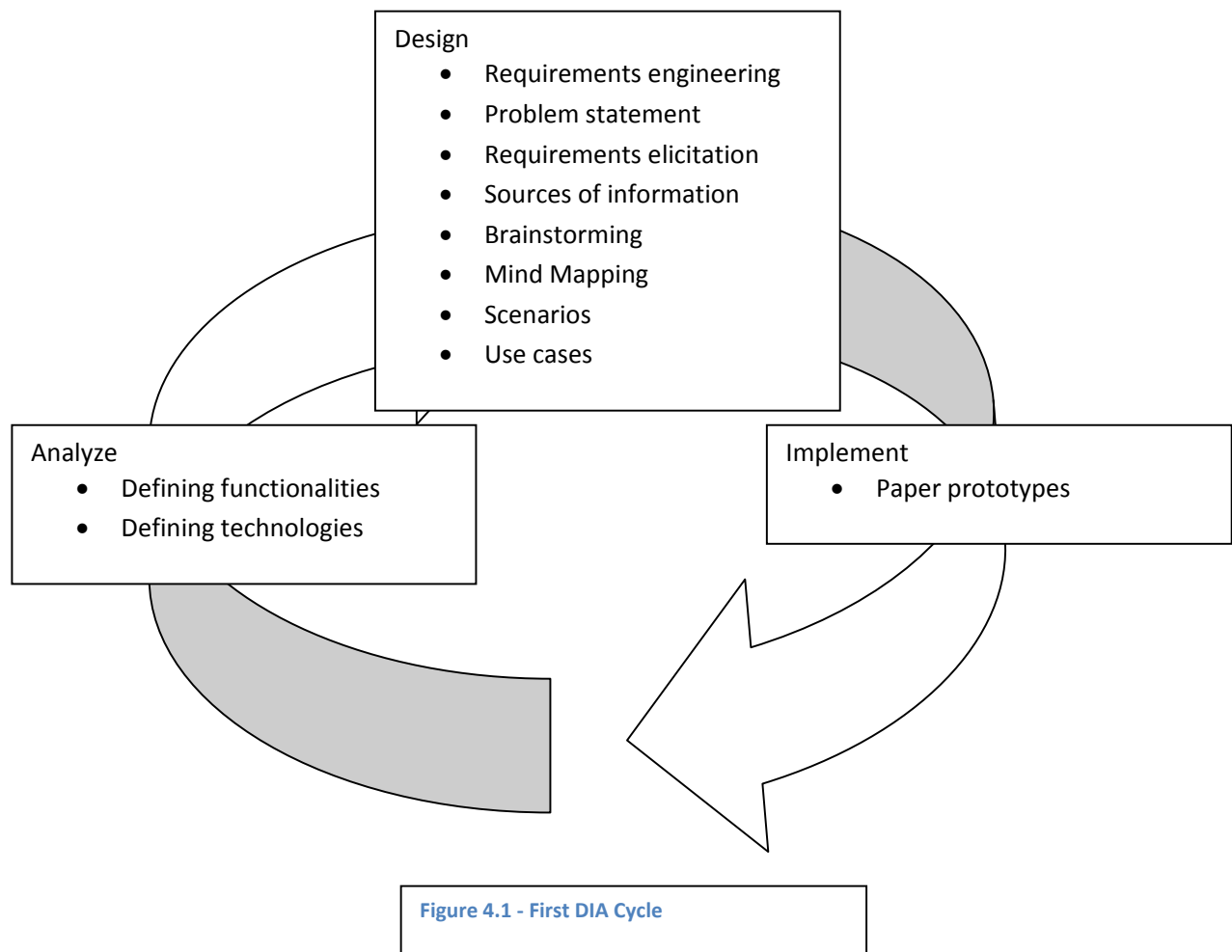
Chapter 6 corresponds to final development of the system, user testing for the usability of the system.

Chapter 7 presents the solution developed in this thesis work.

CHAPTER 4 FIRST DIA ITERATION

4.1 Introduction

This chapter introduces the DIA (Design Implement Analyze) software development cycle. The DIA development approach was used in this thesis. Chapter 5 introduces the first DIA development cycle for this prototype of this work. This thesis developed a system for a proof of a concept; therefore the first iteration was necessary for requirements elicitation and clarification of the research idea.



4.2 Design, Implement and Analyze - The DIA Software Life Cycle

The DIA cycle was presented by Jakob Nielsen [Nielsen, 1993] and Baecker, R., Grudin, J., Buxton, B. and Greenberg, S. [Baecker, 1995]. The DIA cycle stands for “Design, Implement and Analyze”; it is an iterative process for usability and software development for interactive system.

The basic principle of the DIA cycle is the refinement of software to be developed after each cycle; this is accomplished by implementing testing or other analyzing methods. User testing is implemented after every design and implementation phase. The number of iterations depends on the complexity of the project. The DIA cycle does not have a predefined set of tasks for each iteration.

On the early iterations of the DIA cycle the main goal is to identify the major problems (Nielsen calls them usability catastrophes) of a system, the problems can vary from usability to functionality. In later phases of the DIA cycle there are minor improvements concerning usability. However, future enhancement of a system is through evolution.

Following is a graphic from the DIA software life cycle taken from “Readings in Human Computer Interaction: Towards the Year 2000 [Baecker, 1995]”:

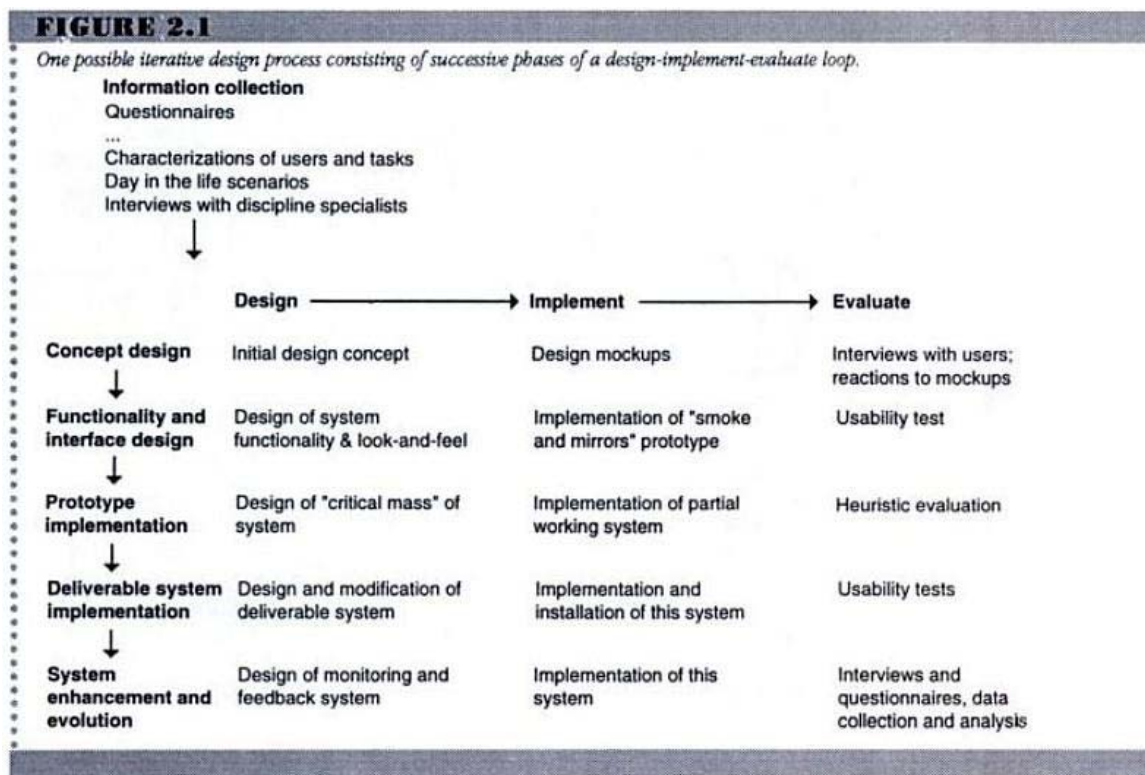


Figure 4.2 – The DIA Software Life Cycle

The software developed in this thesis work required three DIA cycles, the first step was requirements elicitation; the second phase was first software prototype and finally the third phase presents a stable prototype of the system.

4.3 Design

In the first design phase of the DIA cycle for this thesis work, the most important aspect is the user requirements, users make the initial requirements by building the problem statement, after that, the developers have to make a concrete project out of the problem statement. The requirements elicitation for this thesis work was done by using sources of information, brainstorming, mind mapping and scenarios. The previous techniques help to concrete the problem statement.

4.3.1 Requirements engineering

Requirements engineering is the first step in software development [Bruegge, 2004], it helps to answer two main questions at the moment of designing interactive systems:

1. Who are the users?
2. What do they want to do with the system?

Requirements engineering starts with the problem statement, the problem statement is generated by the customer and it defines the problem that the system should solve, but not how it should do it. The problem statement for this thesis work answers both questions:

Problem statement:

“The amount of video media used on the web has increased dramatically. As an example, the Vibelle project is a web portal for deaf people that use video media as its principal mean of communication. Video portals like Vibelle have the need to include an annotation system for video media in order to gather information from their users. However, there is actually no video annotation system for the web for easy installation and deployment.

Moreover video media information is time based and with the existent systems from other platforms there is no possibility to include annotations that supports HTML in conjunction with time parameters for video.

In addition there is currently no system that supports linking to video related information that use time parameters for related video media in order to allow the user to control the playing options. This thesis work argues that by searching inside the annotations that point to video related media with time parameters will help user to find information inside video media faster (cf. 1.2).”

The problem statement suggests developing an annotation system, the annotation system must be able to define time parameters on videos in order to control playing options and the annotations must support the linking of video related information.

However, in order to build the above problem statement, it was first necessary to elicit requirements in order to ensure enough scientific background and to avoid redundancy with other scientific research.

4.3.2 Requirements elicitation

The goals of the requirements elicitation for this thesis work were to find a research area with sufficient scientific background, avoid redundancy with other works and to check what has not yet been

implemented with the actual annotation system. This thesis used the following sources of information to elicit requirements:

- Research papers [WR-16]
- Thesis works
- Journals
- Technical reports on the topic annotation
- Web sites

After having the problem statement and requirements elicited, a solid idea for a thesis topic was needed, brainstorming was the next step taken, and consequently a mind map was built.

4.3.3 Brainstorming, Mind Mapping, and Scenarios

Brainstorming was performed based on the problem statement and requirements elicitation; also meetings with team members of Fraunhofer FIT were hold in order to take part in the brainstorming. The idea of the Envideo system got concrete in this phase. The final idea consisted of making annotations on video media, moreover the annotations should have the possibility to link to other video related media using time parameters for controlling the video playing options (play and stop) on the related media, thus by searching on the annotations that pointed to video related media using time parameters users could find information inside video faster, in contrast of just watching the video media, this idea was attractive and new.

Mind Mapping was made in order to find a linked structure between the requirements and the functionalities that the system should support, the mind map helped to identify the core functionalities of this thesis project.

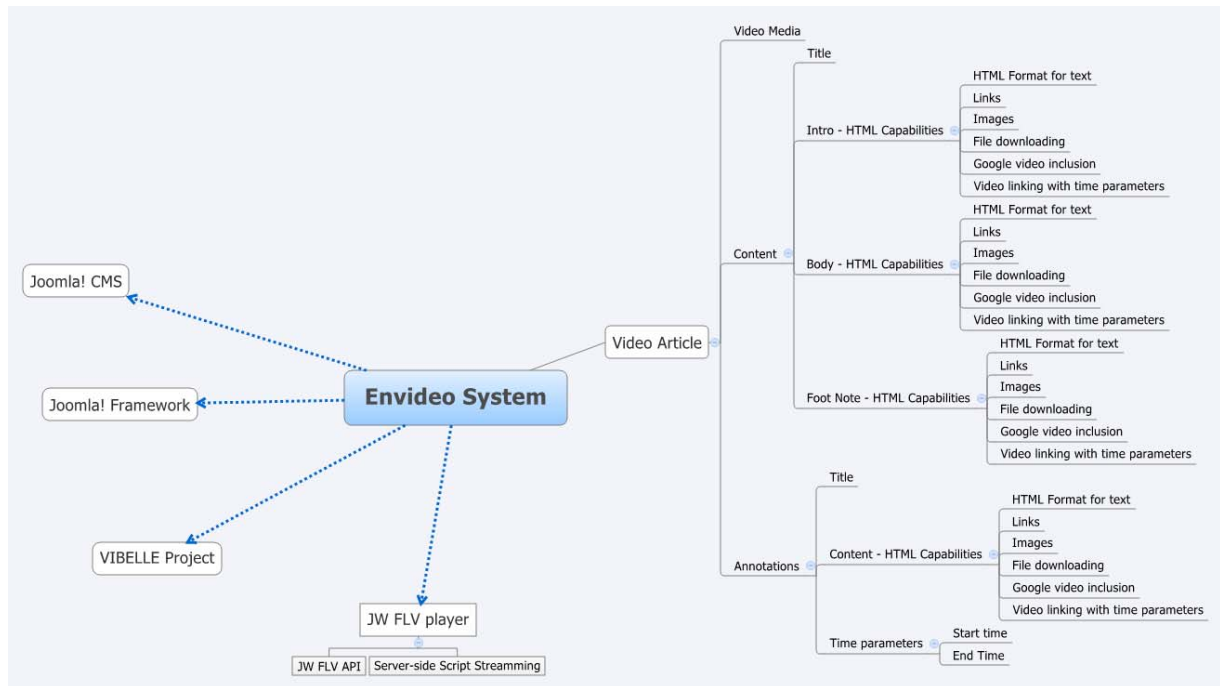


Figure 4.3 – Envideo Mind Map

The next step was to identify requirements elicitation for the system interaction and user interface.

User scenarios [A.2] were written and presented; they showed how users would interact with system, user scenarios were made thinking in how the user could link video related information.

Finally, UML Use Cases were generated in order to elicit the functionalities of the system (cf. 4.3.4).

4.3.4 UML Use cases

This thesis used UML (Unified Modeling Language) in order to have standard diagrams. The UML specification is a well-defined and well-understood specification for system development, and this will ensure understanding and avoid misinterpretation among the readers of this work.

Use case diagrams were used to define the functionality of the system from user's point of view, use cases are part of the requirements elicitation. The following use cases were created:

Watch video with annotations

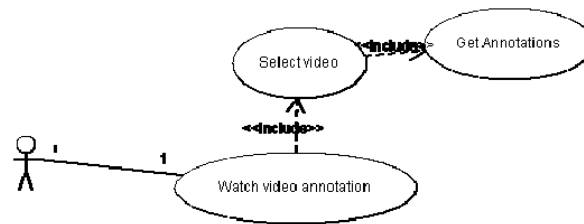


Figure 4.4 – Watch video with Annotations Use Case Diagram

Users must be able to watch video and annotations, and at the moment of selecting the desirable video to watch, all the annotations from that video must be retrieved.

Make annotation

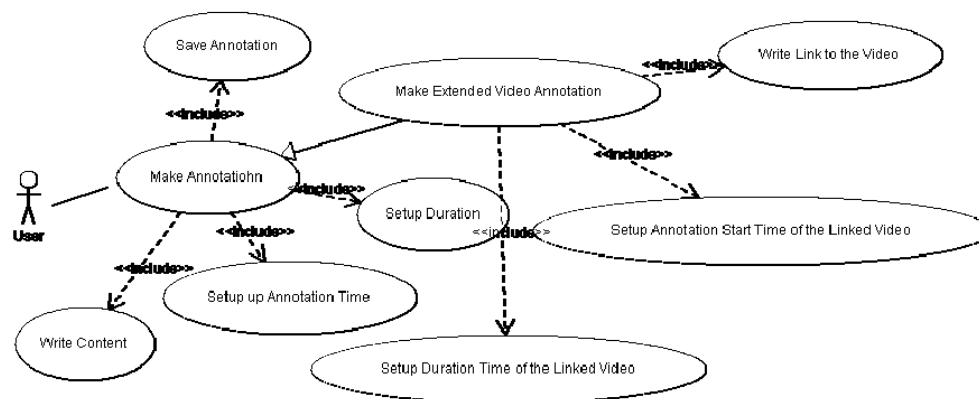


Figure 4.5 – Make Annotation Use Case Diagram

Users must be able to make annotations; the most basic form of an annotation should have content, begin and duration time parameters.

In order to make an extended video annotation, users must setup all the above plus they must provide a link to the related video and setup begin and duration time parameters.

List annotations

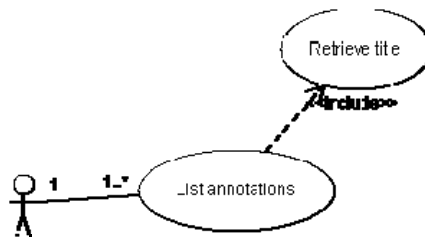


Figure 4.6 – List Annotations Use Case Diagram

Users must be able to see existent annotation on the system.

After eliciting the requirements of the system by using the above methodologies, it was necessary to define the technologies, for that the next phase of the DIA cycle was necessary.

4.4 Implementation

After the design phase, the following functionalities were clear to be included in the implementation:

- Working with video media
- Video player
- Annotations
- Link video information with specific time parameters

The next step was to concrete the functionalities and the first prototype of the user interface. There are several methods for prototyping; at this phase only basic prototyping was used; more advanced prototyping was done on the second DIA cycle. Following are listed the alternatives for early prototyping:

Paper prototypes storyboard-like

Consist of several hand-drawn screens of the future system, they show the interaction between the user and the user interface, they are cheap and easy to create.

Paper prototypes Flipbook-like

Consist of several screens of the user interface of the future system, it emulates the behavior of the system by flipping the pages of the book, its focus is on high level user interface details

Post-it prototype

Consist of screens of the user interface of the future system drawn in post-it notes; it tries to emulate the interaction between the user and the user interface as well. This method is more interactive than the previous methods; it is ideal for dialog boxes, menu interfaces and windows in general.

For this thesis paper prototypes were used for an initial user interface design study. Flipbook-like prototypes were not used because they take more time to be elaborated. Post-it prototypes were also dismissed because the Envideo system was thought initially for the Web and without complicated menus or dialog boxes, therefore paper prototypes storyboard-like were chosen because they are easy, cheap and fast to generate.

4.4.1 Paper prototypes storyboard-like

Several paper prototypes were made at this stage. However, only the final prototypes are published in this thesis work [A.3], Paper prototypes were drawn by hand.

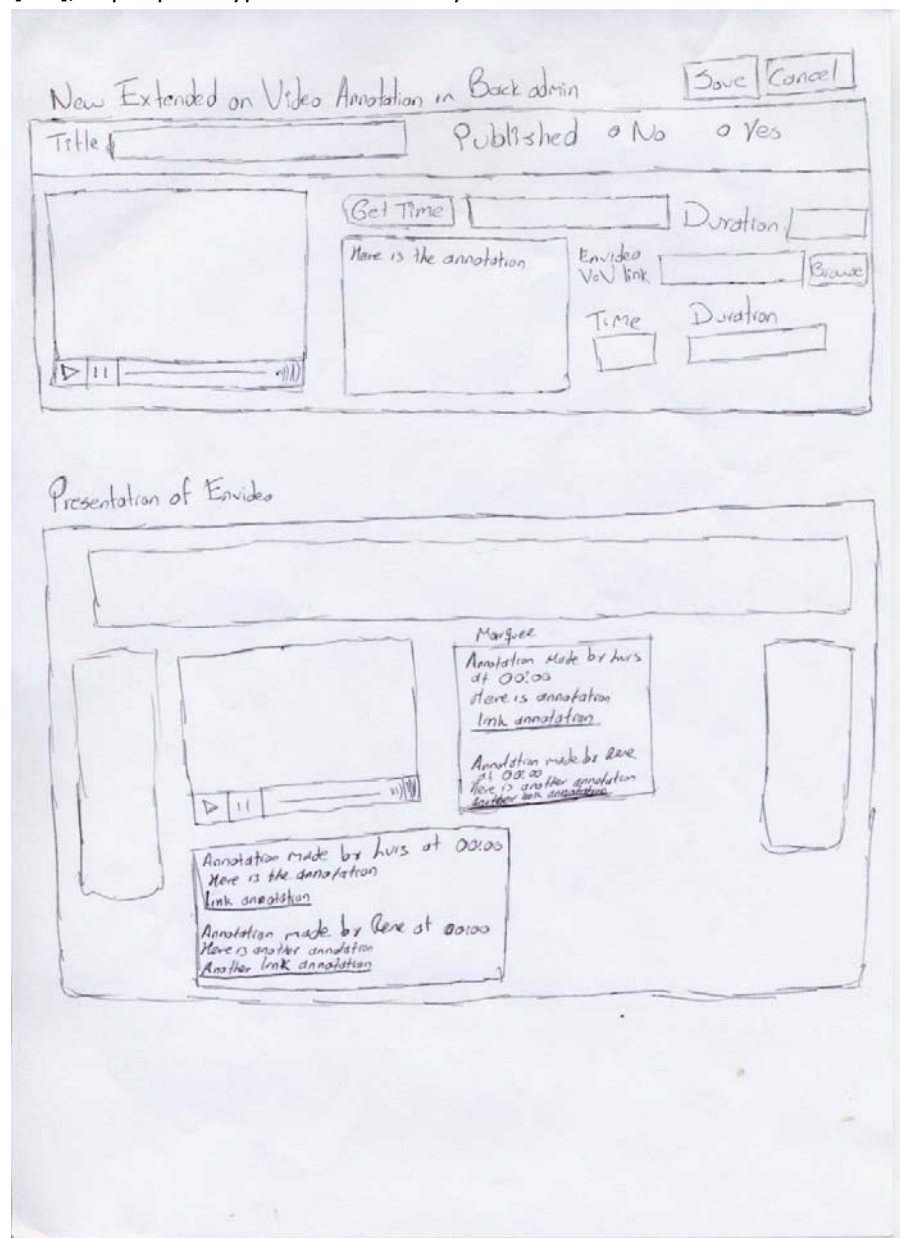


Figure 4.7 – Example of Envideo Paper Prototypes

The paper prototypes helped to define the separation between presentation of information on the back-end and administration on the back-end.

4.4.1.1 Back end prototyping

The back-end initially provided the functionality to create two kinds of annotations:

Video annotations with begin and duration time parameters. The interaction of the back end showed that video annotations let the user choose a video, then making an annotation on it, and finally define the time that the annotation will be displayed on the video by setting up the begin and duration time parameters.

Extended annotation on video, which consisted of the same characteristics as video annotation plus the option to make a link to another video and setting up the play and stop time of the linked video by defining the begin and duration time parameters.

The back end was divided in four areas:

Listing annotations

It was a list with all the annotations made with the system. Authors (cf. 7.4.1) can see all the annotations made with the system and publish them.

New Annotation

Users can make annotations to video media; the video player was included in order to show the video being annotated and a video could contain several annotations. An annotation field was included, and a button that gets the time from the video player was envisioned as well, thus users do not need to type the time manually.

Extended video annotation

The extended video functionality incorporates all the possibilities from the new annotation option, plus fields to link video related media with begin and duration time parameters in order to control the play and stop options of the linked media.

Edit annotation

This option displays a video with a list with all the annotations made on that video, all annotations could be modified at this stage.

4.4.1.2 Front-end prototyping

The Front-end prototyping consisted of three parts:

Video player

It displayed the video that was annotated

Annotations

It displayed a listing of all the annotations made to that video, including normal annotations and extended video annotations, the videos defined on the extended video annotations will be displayed as a link.

Marquee

A dynamic area that will display annotations that incorporate annotation time parameters, it synchronizes the annotations that incorporate begin and duration time parameters with the running time of the video being played on the video player. This means that this area will dynamically change its content according to the time setup on the annotations.

4.5 Analyze

After building the first paper prototypes and use cases, the following points were discussed:

Video Player selection

According to the requirements elicitation a video player was required, the video player should provide an API (Application Programming Interface) to control its video functionalities, for that two video players were analyzed, Viddler video player (cf. 3.2.3) and JW FLV player (cf. 2.6):

Viddler

The video player from Viddler web service provides an API for annotations; however, it does not provide an API for video controlling at the time of this thesis writing. It is designed to work with the web service from Viddler and it can only be used separately but asking directly asking a key to the developers of Viddler.

JW FLV player

The JW FLV player is dual-licensed, it is free to use under certain conditions, and it provides an API for controlling almost every aspect from the video player.

The final decision was made for the JW FLV player, because its API is complete, it provides all the video functionalities required for the proposed system in this work, also the player is continuously under development, it is stable and widely used.

Framework

The decision to develop under the Joomla! Framework and use Joomla! CMS as the deploy environment was also made at this phase, the latest version of the Joomla! Framework is completely Object Oriented and stable, despite that the current version of the Vibelle project is running on the previous version of Joomla! (Joomla! 1.0.x), the decision of building on top of the new version was made in order to ensure that the developed component could be used in future versions of Vibelle.

Framework and User Interface

The front-end and back-end were clear, that implied a web design, therefore a decision to make a web application was made at this stage.

The user interface presented problems due to the lack of defining the development Framework; with the current prototypes the following questions raised:

Which objects are the users going to annotate? Are users going to annotate only on video media files or articles from Joomla! CMS? Is the application going to be a client-server architecture structure? Is the application going to be a standalone? Is the application going to be a Joomla! component?

For the system developed under this thesis, it was decided to use Joomla!'s Framework and Joomla! CMS as the development and deploy environment, because the Joomla! Framework is object oriented, stable and continuously updated and Joomla! CMS as well, moreover Joomla! CMS is used in the Vibelle project, therefore developing a system for Joomla! CMS will provide a low learning curve for the proposed system inside the Vibelle project.

The final decision concerning framework is to use the newest version of Joomla! Framework, at the moment of this writing there are two version of Joomla! being Joomla! 1.0.x the oldest and Joomla! 1.5 The newest. Despite that the current version of the Vibelle project is running on the previous version of Joomla! (Joomla! 1.0.x), the decision of building on top of the new version was made in order to ensure that the developed component could be used in future versions of Vibelle.

4.6 Conclusions

Requirements elicitation took part on the first DIA cycle, for that brainstorming was performed consequently mind mapping, scenarios, paper prototypes and use cases were developed.

The first DIA cycle also helped to establish the ground work for the development; this included video player selection, development and deployment environment.

The functionalities that the system should perform were clear however, the user interface and the workflow were not clear; this was due to the lack of definition for the deployment. However, in the second DIA cycle the user interface was re-designed and the first working version of the system was developed.

CHAPTER 5 SECOND DIA ITERATION

5.1 Introduction

This chapter introduces the second DIA development cycle of this thesis project. The first phase of the second DIA cycle compromises software prototyping for the component being developed, the definition of the database schema and the workflow for the system.

On the second phase the first version of the software was developed. Joomla! framework was used in order to develop the software component. In this phase the time based video related information functionality was also included.

The cycle finishes by doing a user testing in order to identify user interface catastrophes and important errors on the design

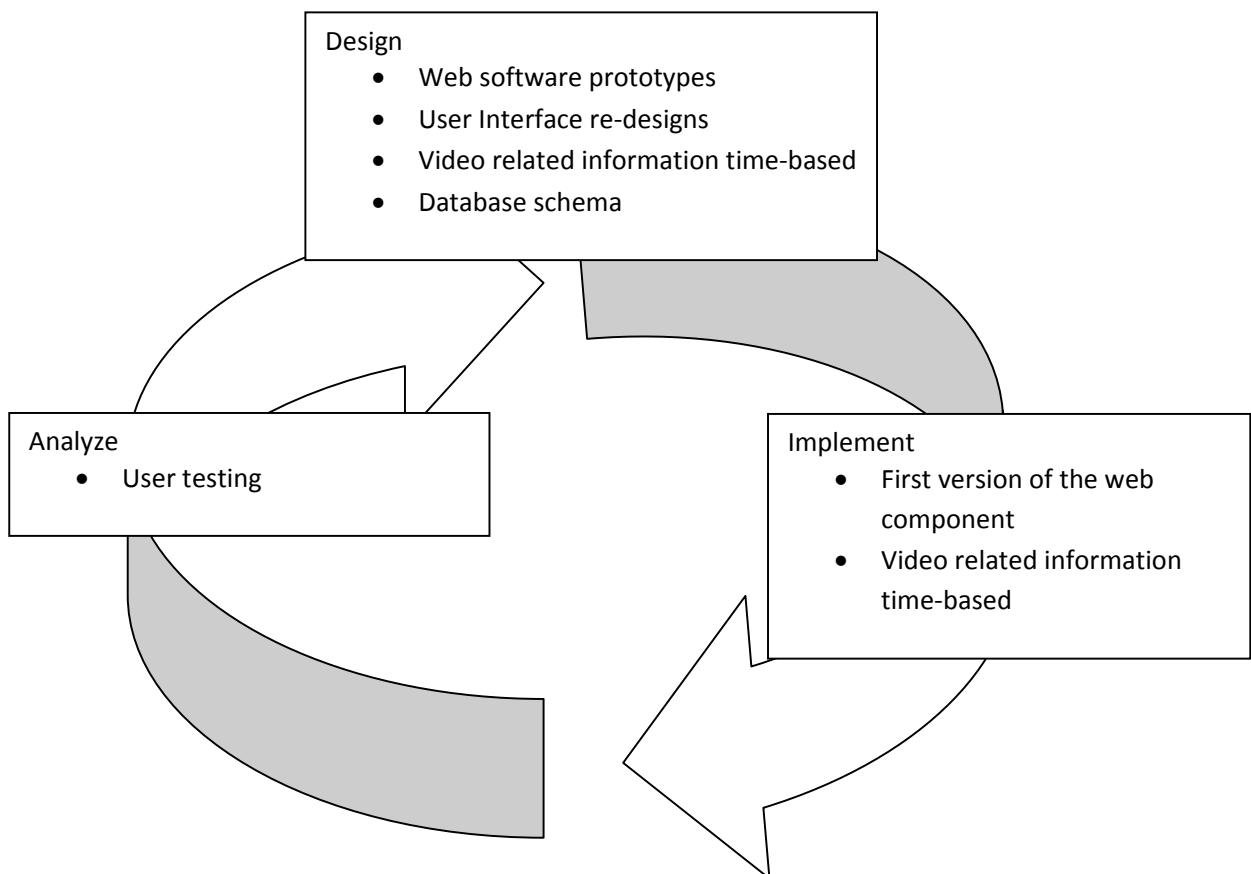


Figure 5.1 - Second DIA Cycle

5.2 Designing

The previous iteration of the DIA cycle suggested a development of a component for the Joomla! CMS. Therefore the second design iteration of the Envideo system followed a similar user interface and workflow like in Joomla!. The Envideo system looks similar like any other component inside Joomla! thus preventing a high learning curve for users already acquaintance with the Joomla! CMS.

The functionalities were already defined; therefore, the workflow of the system was the first step to establish. The workflow of the system tells how the different functionalities of the system should interact. At this step two information elements were identified: video articles and annotations, this is why the workflow for these two elements was necessary.

User interface was also designed; software prototyping was of great help for the new user interface for video articles and annotations.

5.2.1 Workflow

The decision of developing a component for Joomla! (cf. 2.5.2) was made on the previous DIA cycle; a workflow was designed for the system, the workflow involves two types of sources of information, video articles and annotations.

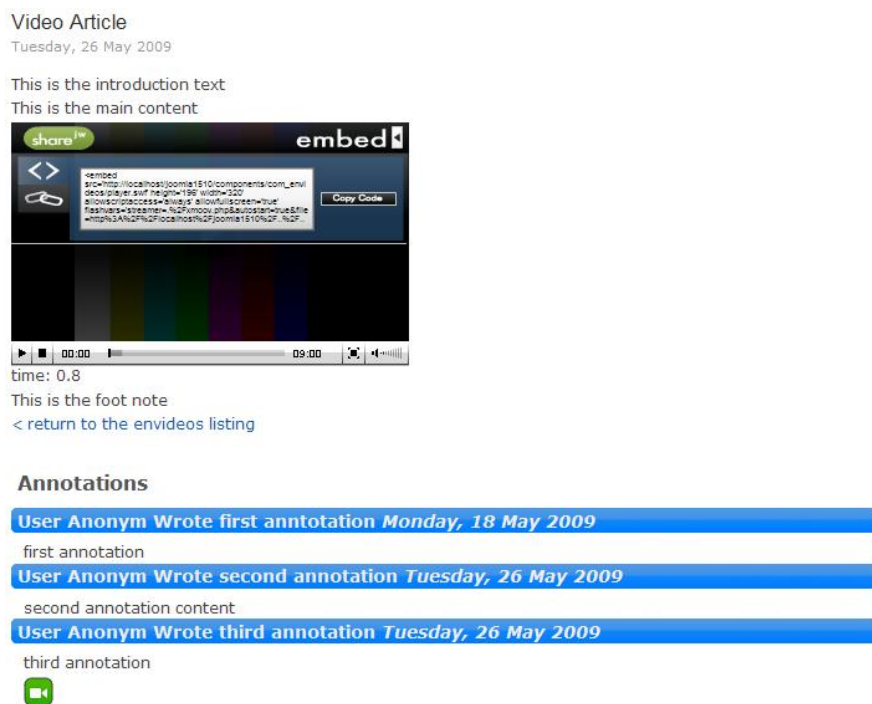


Figure 5.2 – Video Articles and Annotations Workflow

Video articles allow the user to extend the information of one video, Video articles consist of: title, a video link, introduction, main text and foot note. Introduction, main text and foot note may contain HTML markup language.

5.2.2 Annotations

Annotations are made on video articles; they allow the users to extend the information of video articles. Annotations consist of a title and a body for text, the body can contain HTML markup language. Video articles and annotations are a clear separation of roles for content, where video articles are generated by administrators and authors, annotations are generated mainly by users that are consumers of content.

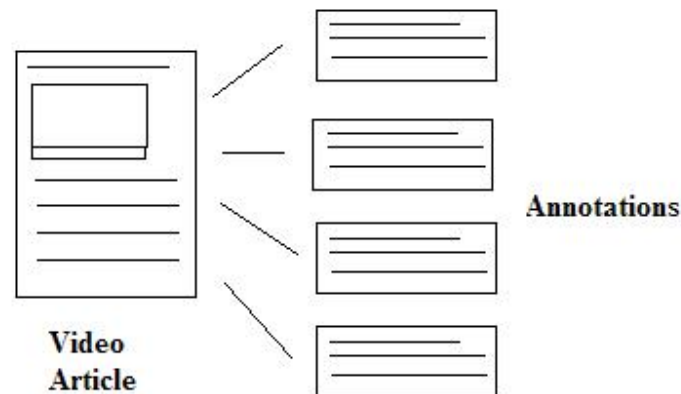


Figure 5.3 - Video article and annotations framework

By making annotations users extend the information on video articles, therefore the design of the user interface was made simple.

5.2.3 Prototyping

The workflow indicated two sources of information, the following step was to develop software prototypes, and software prototyping is the next step after paper prototypes.

Software prototypes provide more details; they show the user interface and also show (depending on the software prototype) the interaction with the future system, there are three ways to define a software prototype: vertical prototype, horizontal prototype and scenario.

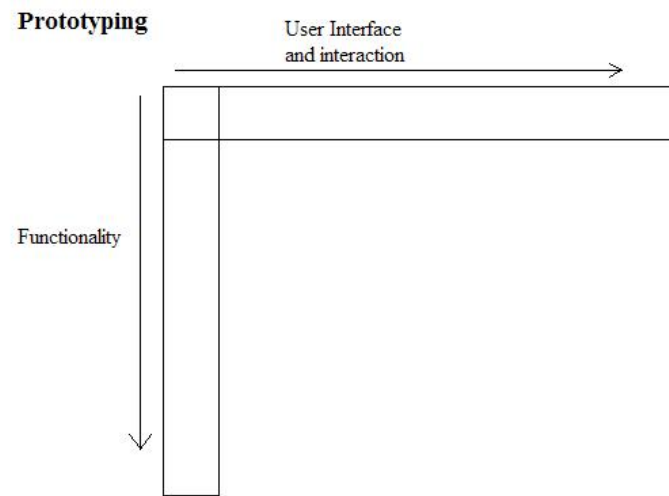


Figure 5.4 – Prototyping Approaches

Vertical prototyping consist of implementing few functions but in detail, horizontal prototyping is implementing the entire user interface and the interaction with it, but no functionality and scenario prototyping is a combination of both.

The author of this thesis work focused more on vertical prototyping because the envisioned system did not show a complicated user interface; this means there were not many screens and message boxes. The most important part of the system was the video player interaction and the inclusion of video related information using time parameters, therefore a deep research on those functions was needed.

There are several approaches for software prototypes; there are screenshots, screen storyboards, animation applications, web, rapid development environment, user interface designers and hardware prototyping.

This thesis used web software prototyping because the envisioned system is a web component, at this stage the Joomla! Framework was used in order to generate the software prototypes thus allowing the development continuity of the generated web software prototypes.

The creation of the software prototypes was a long and time-consuming task, the final software prototypes were highly elaborated, because the generated software prototypes were aimed to become the final product. The Joomla! Framework provides a mechanism for generation of menus and HTML elements; however, it is difficult to use it.

5.2.4 New User Interface Prototype

A new user interface was designed, for the back administration and the front administration, and all the elements of the interface suffered a substantial change from the previous user interface of the first DIA cycle.

The back administration was divided in two parts, video articles and annotations, video articles are the main sources of information, they consist of a title, a video link, an introduction text, a main content text and a foot note area, where introduction, main content and foot note areas can contain HTML code, video articles are used to extend the information of the included video stated on the video link.

The screenshot displays a web-based form for creating video articles. It features the following elements:

- Title:** A single-line text input field.
- Video Link:** A single-line text input field.
- Introtext:** A text area with a rich text editor toolbar above it. The toolbar includes options for bold, italic, underline, text color, background color, bulleted and numbered lists, indentation, link, unlink, image, table, and various alignment and font settings.
- Text:** A larger text area with a similar rich text editor toolbar.
- Foot Text:** A text area with a similar rich text editor toolbar.
- Image, Pagebreak, Readmore:** Three buttons located below each text area, used for inserting media or structural elements.
- Published:** A checkbox at the bottom left, currently unchecked, with radio button options for 'No' and 'Yes'.

Figure 5.5 – Creation of Video Articles 2nd DIA Cycle

The annotations interface consists of a title, a main content, begin time and end time of the annotation. Begin and end time of the annotation states the time that the annotation has impact on the Video inside

the video article. The major change on time-based annotation was the idea of changing the duration time parameter; it is easier and more pervasive for the user to setup the end time, because users actually see time on the video player.

The screenshot shows a web form for creating annotations. At the top is a 'Title:' label followed by a text input field. Below this is a rich text editor with a toolbar containing various icons for text formatting (bold, italic, underline, etc.), alignment, and insertion. Below the editor is an 'Annotation:' label followed by a large text area. At the bottom of the form are three buttons: 'Image', 'Pagebreak', and 'Read more'. Below these are two rows of input fields: 'Begin Time:' and 'End Time:', each followed by a 'Get time' button. At the very bottom is a 'Submit' button.

Figure 5.6 – Creation of Annotations 2nd DIA Cycle

Linking to video related information time-based, this option was incorporated on the second cycle of the DIA, in comparison to the first DIA cycle, the second envisioned more than one link to video related information time-based, thus the ability to increase linking video related information in one annotation. One common way in Joomla! CMS to incorporate more functionality is through the use of patterns, patterns in the Envideo system have the following form:

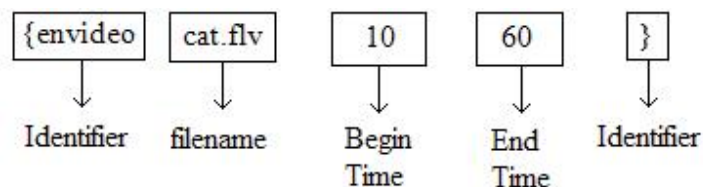


Figure 5.7 – Parts of the Video Related Information Pattern

With the pattern below users are able to relate videos that are inside the same server, defining the beginning time and the end time to which the related video will start playing.

5.2.5 Database schema

The following database schema was designed; the tables are normalized in the first, second and third normal form.

The design also includes foreign key constraint in order to provide referential integrity, it also allows relating video articles with video annotations and it also avoids the use of another table for relating the jos_envideos table with table jos_envideos_annotations. The table Jos_user is a table provided by Joomla!'s core, the Envideo system uses Jos_user table in order to identify the users that make the annotations.

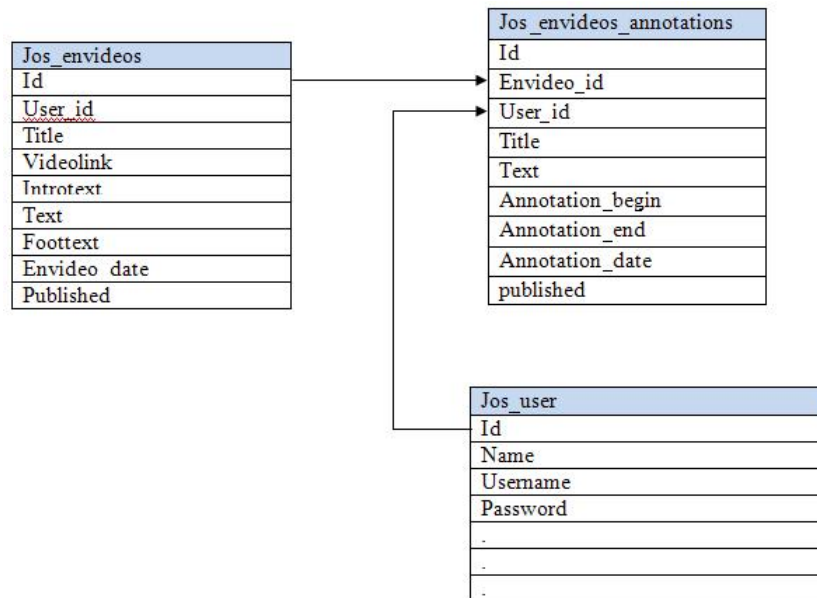


Figure 5.8 - Envideo Database Schema

5.3 Implementation

The implementation of the Envideo system was made using the Joomla! framework. Therefore an installer was built in order to be included into the Joomla! system, an installer is not necessary for the development of a system but it is for the deployment. The installer is also in charge of providing the access mechanisms for the installed system and the creation of the tables inside the database.

The Joomla! framework also provides a mechanism for the inclusion of toolbars inside a component, despite that the creation of the menu system for a component inside the Joomla! backend is not straightforward, it was used because it helps to keep functionality organized. Moreover, by using the menu system administration provided with the Joomla! framework a low learning curve for users and developers that already use the Joomla! CMS and the Joomla! framework is ensured.

5.3.1 Video player

The system developed in this thesis work uses the JW FLV video player for delivering of video on the internet. The JW FLV player provides an API that let developers control the following functionalities: play, stop and seek. These functionalities are mandatory for the software developed in this thesis;

therefore the API (Application Programming Interface) of the JW FLV player was included inside the implementation of the Envideo system.

The API of the JW FLV player is complete and easy to use; it includes listeners for monitoring several aspects of the video player like time and volume. The video player itself has a wide range of parameters with which developers can configure the player to their own needs. Finally the JW FLV video player supports delivery of video through internet with the seek functionality; this is accomplished by using a PHP server side script.

5.3.2 Delivering video on the internet

There are two ways of delivering video over the internet: progressive download and streaming.

Progressive download means that the user watches the video as the video is being downloaded, the user can only jump to different parts on the video (Seek function) only when such parts are already downloaded, but the user cannot jump to parts that have not been downloaded. The seek functionality is essential in this master thesis, therefore progressive download is not suitable for the system developed in this thesis work, progressive download does not offer content protection as well.

Streaming video can be categorized in two: using streaming video servers and HTTP streaming.

Streaming video servers provide seek functionalities and content protection, the drawbacks of video servers are that they have to be installed on a server and sometimes they are expensive, furthermore technical knowledge is required.

HTTP streaming consist of HTTP information delivered by server side scripts (PHP, ASP, etc). HTTP streaming consist of: a video player, a server side script and a FLV video file, the FLV video file must have keyframes data, if FLV files do not have keyframes data, these can be added by using FLVMDI program.

This thesis work used the HTTP streaming method, the HTTP streaming functionality is supported by the JW FLV player, the JW FLV uses the utility xmoov-php, and like its name implies, xmoov-php is a server script developed in PHP for HTTP streaming [reference here].

The HTTP streaming method provided by the JW FLV player and the xmoov-php provides a perfect framework for video delivery that supports the seek functionality because no installation of a video server was needed moreover the inclusion of the xmoov-php script into the JW FLV is well documented and straightforward.

5.3.3 Envideo development

Joomla! Framework is objected oriented and it implements the Model View Controller design pattern (MVC) [reference here], the model view controller designer pattern main goal is to separate the business, data access and user interface. Model is concern with the data access, View is concern with the presentation of the information and Controller is concern with the interaction between Model and View.

The MVC gives the flexibility to add or modify different parts of the implementation without affecting others; it also prevents the duplication of code in the case of views where XHTML is used extensible.

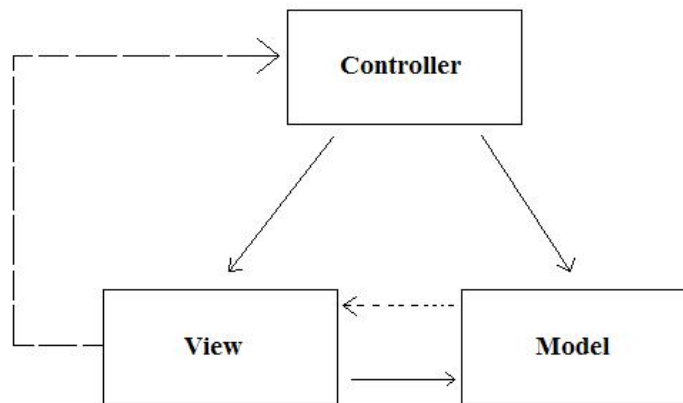


Figure 5.9 – Model View Controller Diagram

This section will not go deep into explaining the MVC design pattern; however, it will present the views and models used on the system.

This thesis work provided three views:

View all for visualizing all the video articles in a list, the model for this view gets the title and the introduction text.

Video Article
This is the introduction text
The history of Videogames by discovery channel
This is a documentary from the Discovery Channel about the history of video games.
Abs in 8 Minutes
This video is about 11 abs exercises in 8 min

Figure 5.10 - Video Article Listing

View single article for visualizing a single article and all its annotations, the model for this view brings all the data for a video article.

View annotation for visualizing related video media with time parameters, the model for this view brings a single annotation data.

5.4 Analyze

At the end of the second DIA cycle a small user testing was envisioned in order to identify the major problems of the system. According to Jakob Nielsen:

“We can gain sufficient insight into the usability problems in a design iteration with only a few test subjects. And it may not be necessary to collect quantitative measurement data. The quantitative measures emphasized in this article may be useful for management in larger projects. But they are not the main goal of usability evaluations. Instead, the principal outcome of a usability evaluation in a practical development project is a list of usability problems and suggestions for interface improvements.”

A user testing was conducted using six users; the goal of the first user testing was to identify problems in functionality and user interface, the participants were asked to perform 4 tasks: make a video article, make an annotation, link video related information and search information using the Envideo system. The conclusion was that all the users did all tasks successfully and all the users were surprise to see the system working at the moment of searching and finding information inside the video faster instead of using the normal video. However, the following problems were found on the user interface:

5.4.1.1 Backend administration

Generation of video articles was not clear because users did not know what to do.

Friendly selection of video file, the system currently uses an input field for selecting the video file used to create the video article; the users testing showed that users would like to have a dialog box for file selection.

There was no helping system; therefore users could not see the purpose of the system.

5.4.1.2 Front-end problems

The identification of the back link (return to the envideos listing) was not clear.

There is a link inside video articles that let the users go back to the listing of the video articles; this link is called “return to the envideos listing”.



Figure 5.11 – Return to the Envideos Listing Link

However, the user test showed that the link “return to the envideos listing” was invisible to users and no descriptive.

Users did not know what to do inside video articles area; this problem was due to the lack of help system.

The annotation system was at the very end, therefore the get time buttons were not clear, because the get time buttons get the time from the video that is inside the video article, but at the moment of using the get time buttons, the video window was not visible.

Identification of introduction, main content, footnote and annotation sections was inexistent. Users could not identify which part was introduction or content or footnote.

System was not appealing; one user complained that the system was ugly.

Users could enter wrong time values because users could edit the input boxes.

A screenshot of a web interface showing two rows of input fields. The first row is labeled 'Begin Time:' followed by a text input box and a blue button labeled 'Get time'. The second row is labeled 'End Time:' followed by a text input box and a blue button labeled 'Get time'.

Figure 5.12 – Time Input Fields

Timed annotations were not taking into account, this was because the marquee was not yet developed, and therefore there was no distinction between timed annotations and non timed annotations.

5.5 Conclusions

At this point of the development the framework of the project was stable and solid, the workflow was defined, the database schema was defined, the functionality was clear. However, the user testing showed usability problems.

The third DIA cycle focuses on analyzing the problems presented on the second DIA cycle, it proposes solutions and release the first stable version of the Envideo system.

CHAPTER 6 THIRD DIA ITERATION

6.1 Introduction

This chapter introduces the third DIA development cycle of this thesis work; this is the last cycle of the system developed for this thesis project.

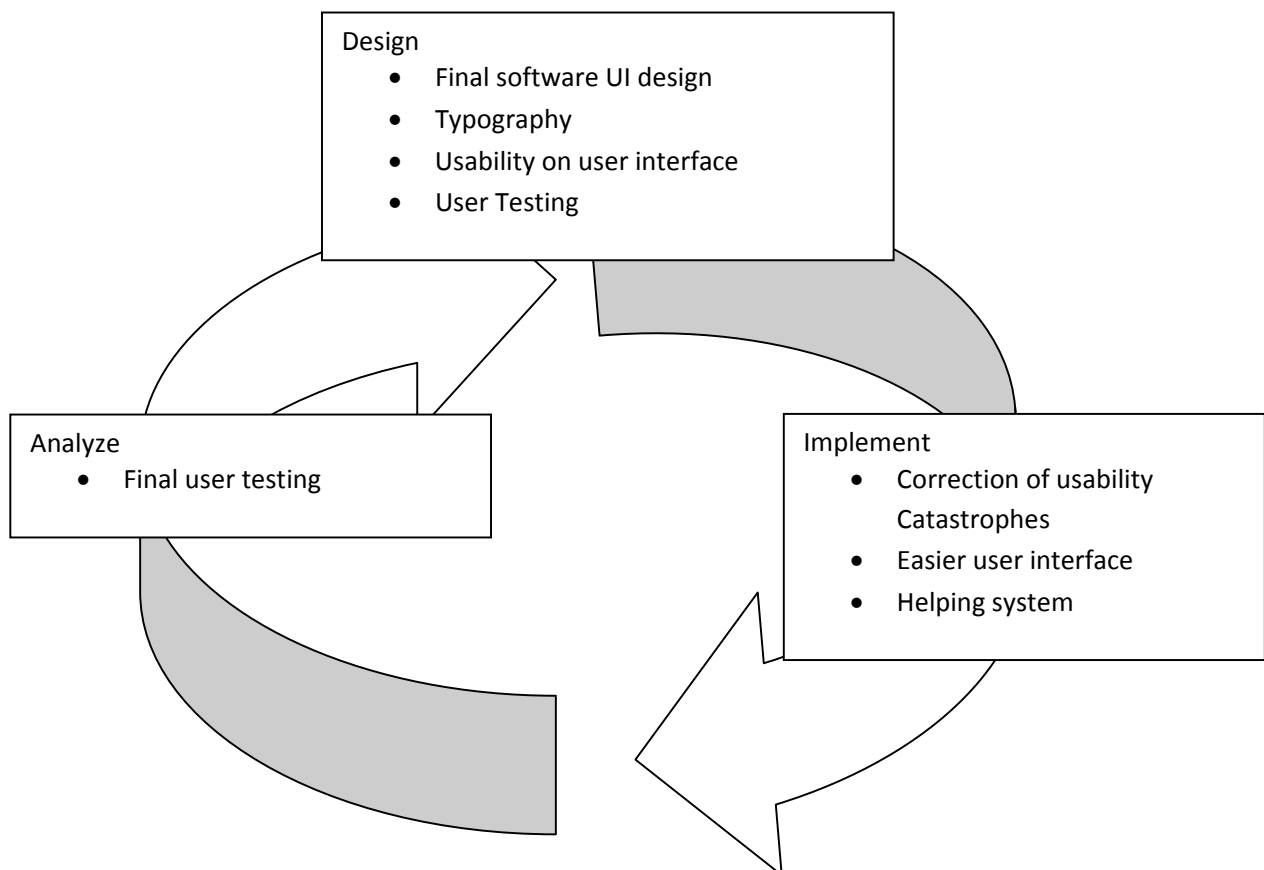


Figure 6.1 - Third DIA Cycle

6.2 Designing

The user testing from the second cycle of DIA (latest chapter) showed usability problems, the designing section of the third DIA cycle propose solutions for the found usability problems.

Usability problems	Proposed Solutions
Back-end usability problems	
Generation of video articles was not clear	Documentation and help system on the backend
Friendly selection of video file	No proposed solution yet
Users did not understand the purpose of the system	Documentation and help system on the backend
Front-end usability problems	
Button “return to envideo listing” was not visible for the users	Use a graphic for a back button
They did not know what to do inside the video articles	Documentation and help system
The annotation system was at the very end therefore the get time buttons were not clear	Relocate the annotation area after the video article
Identification of introduction, main content, footnote and annotation sections was inexistent	Use labels to identify the different sections
System was not appealing	Generate a new look and feel for the system
Timed annotations were not taking into account	Implement marquee functionality
Users could enter wrong time	Disable input text boxes and validation of data at the moment of submitting the form

Table 6.1 - Usability Problems and Proposed Solutions

6.3 Implementation

After designing the third DIA cycle, a big usability problem was identified, and this was a lack of documentation, documentation about the system was inexistent inside and outside the system. A helping system was developed; the helping system consists of two parts: a quick start manual and constant feedback on key parts of the system.

Quick start Manual

A quick start manual was developed; the quick start manual gives a brief introduction to the Envideo system and explains the generation of video articles, annotations, and link video related information, search of information and changing of look and feel. The quick start manual is also available inside the system [Annexes 4].

System Feedback

In some key areas of the system, feedback was provided; feedback is provided in two ways, help buttons and tooltips

Help buttons

Help buttons were implement on the front-end and on the back-end, this to provide users with information about the system at all time.

Help buttons were used in the front end



Figure 6.2 – Help Button on the Front end

And on the back end

Tooltips

Tooltips were used in order to give short descriptions about parts of the system or key functionalities,

Tooltips were used on the front-end and on the back-end.

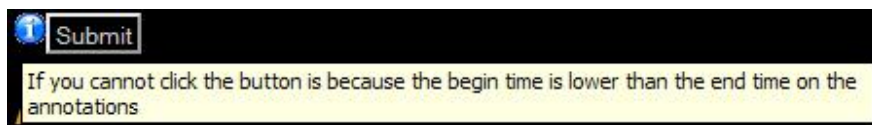


Figure 6.3 – Tooltip in Submit Button

And on the back end

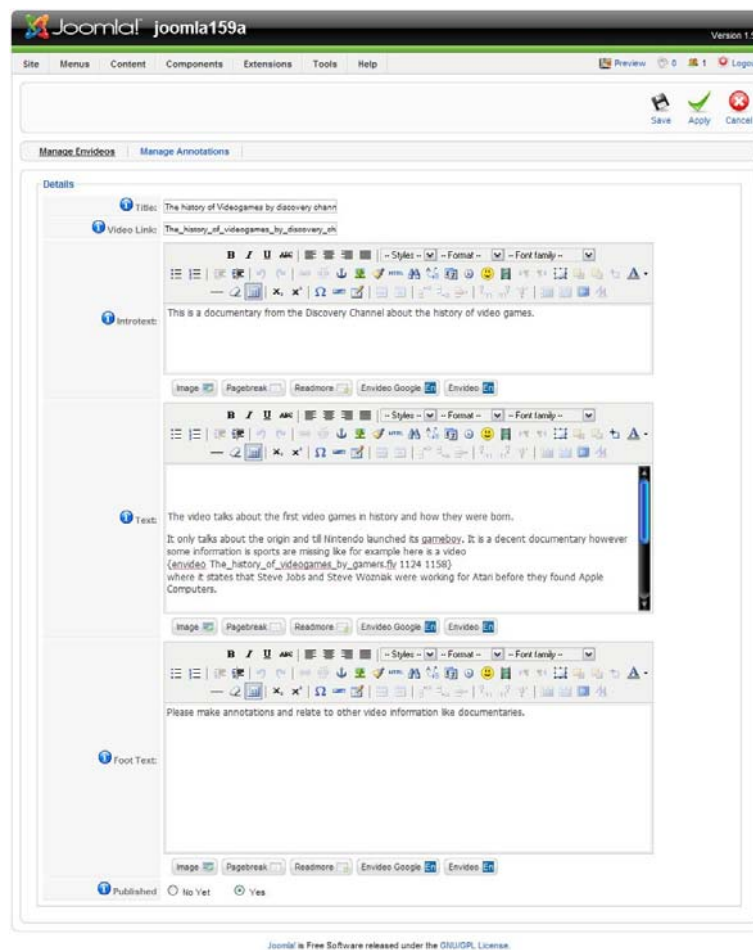


Figure 6.4 – Tooltips on the Back end

According to Nielsen [W-17] the back button is the second most used navigation feature while navigating on the web, the “return to the envideos listing” is the equivalent to the back button of a web browser where users can go back on the navigation, however, this important navigation feature was invisible to the user, therefore improvement of the back button was made by incorporating a button on top of the system.



Figure 6.5 - Back Button on the Front end

6.3.1 Marquee functionality

The marquee functionality is used to display the title of annotations that incorporate time parameters on the right side of the video player (figure 6.6).



Figure 6.6 - Annotations Marquee

The goal of the marquee functionality is to encourage users to use time parameters in their annotations, by display them on the side of the video.

6.4 Analysis

User testing

6.5 Conclusions

Future Work

CHAPTER 7 PRESENTATION OF THE ENVIDEO SYSTEM

7.1 Introduction

This chapter presents the system developed in this work, called *Envideo*.

Envideo is a web component developed for Joomla! CMS. It uses the Joomla! framework, PHP, MySQL and the JW FLV player as its development environment.

The objective of this chapter is to explain the purpose of the technology developed for this thesis work and how it achieves it.

7.2 Objectives

Envideo is the system developed in this thesis work, it is a system that gives the users the possibility to extend information of video media on the web (also locally), it is aimed to web sites, Companies or Individuals that work with videos and would like to have a tool that provides:

- Video functionalities
- Annotation capabilities
- Information searching
- Information retrieval

The Envideo system provides the necessary tools to extend information of video media. It also provides tools to make video annotations that incorporate time parameters, and it also lets the users to relate video information using time parameters in order to control the video functionalities on related videos.

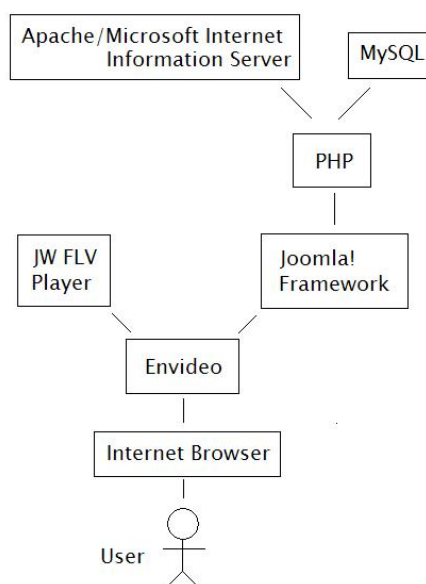


Figure 7.1 – Envideo Framework

The main goal of the Envideo system is fast information searching and retrieval inside video media; this is accomplished by using the annotations made by the users that incorporate time parameters.

The Envideo system is a proof of concept for this thesis work which argues that by using time parameters on the information used to extend the content of the video it will help to find and retrieve information inside video media faster.

7.3 Envideo User Interface

Envideo works inside Joomla! CMS (cf. 2.5) therefore the administration part and the front user interface are divided, each part use a different Urls, users have to login inside the administration part of Joomla! in order to access the administration of the Envideo system. The front user interface is accessible through main web page.

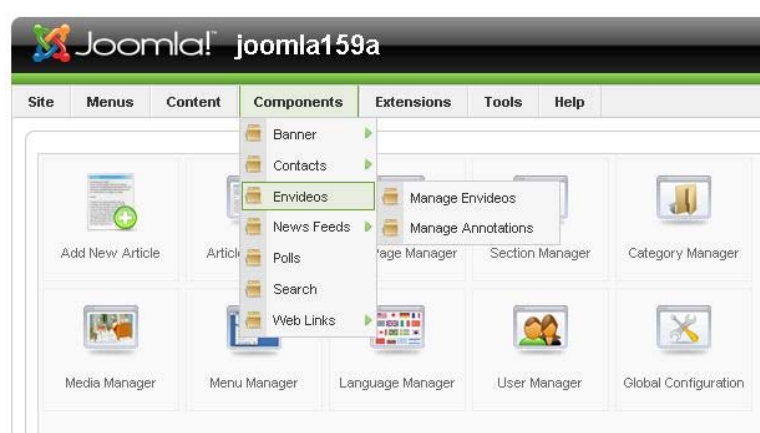


Figure 7.2 – Envideo Component in Joomla! Back end

7.3.1 Envideo Back end

The Envideo back-end is the administration area of the Envideo system, on the back end the Authors of video articles (cf. 7.4.1) can create, edit, delete, publish and unpublish video articles (cf. 7.5.1), create, edit, delete, publish and unpublish annotations (cf. 7.5.2).



Figure 7.3 – Envideo Administration

7.3.2 Envideo front-end

The Envideo front-end is accessed through the main web page; it is the responsibility of the administrators from Joomla! (cf. 2.5) to create a link to the Envideo component by using the Joomla! core functionalities. On the front-end, users are able to see video articles and their annotations (if any), users can also search and retrieve information inside video articles and only registered users are able to make annotations.

7.4 Envideo User Profiles

There are three different user roles during the operation of the Envideo system, the user roles are dictated by their activities inside the system. The Envideo system integrates the user management from Joomla!, therefore the responsibility for assigning user profiles relay on the super administrators from Joomla!.

7.4.1 Authors of video articles

Authors are the users that have all the control over the video articles and they can perform the entire tasks provided by the Envideo system; the authors are the only users that can access the administration of the Envideo system and are the only users that can create video articles.

7.4.2 Authors of annotations

Users that make annotations must be registered in the Envideo system, this user profile is able to annotate video articles, search and retrieve information. The users that make annotations are also able to link video related information and control the video functionalities with the help of time parameters.

7.4.3 User that search and retrieve information

This user profile does not need to be registered; this kind of user is only interested in searching and retrieving information from the video media or from annotations. Users inside this profile are not able to make annotations or video articles.

7.4.4 User Mapping

Following there is table with an equivalent of the user profiles from Joomla! CMS to Envideo system:

Envideo User Roles	Joomla! user management
Author of video articles	Back-end Access
	Super administrator
	Administrator
	Manager
Author of annotations	Front End Access
	Publisher
	Editor
	Author
User that search and retrieve information	Registered
	Public use

Table 7.1 – User Mapping

7.5 Creation and extension of content in the Envideo system

The Envideo system provides a platform to extend content of video media; there are three methods for extending content:

- Creation of video articles
- Annotation on video articles with time parameters
- Link to video related information and control the video functionalities of the linked media with the help of time parameters.

7.5.1 Creation of video articles

Articles can only be created using the administration of the Envideo system which is on the back administration of Joomla!, therefore only users with sufficient privileges (Back end Access cf. 7.3.1) can create video articles.

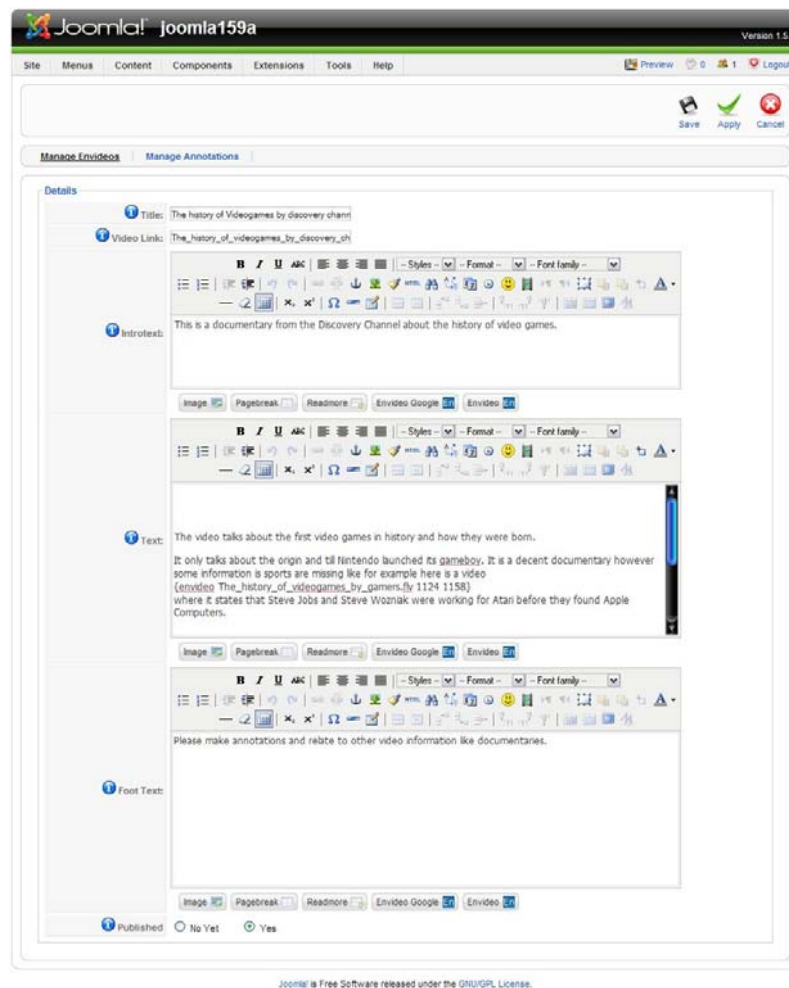


Figure 7.4 – Creating a Video Article

A video article contains one video; users extend the information on the video by generating an article that uses text and HTML code, therefore it is possible to include images and links to other web resources, an article can also contain links to other video related information that incorporate time parameters and control their video functionalities. Articles have the following fixed structure:

- Title
- Video Link
- Introduction
- Content
- Foot Note

Articles can only use HTML code on the introduction, content and foot note sections. Video articles are the main components of the Envideo system.

7.5.2 Annotations

Users can create annotation using the front-end or the back-end of the Envideo system, annotations are created inside video articles, and an annotation has the following fixed structure:

Figure 7.5 – Annotations' User Interface

- Title of annotation
- The content of the annotation
- Begin time of annotation
- End time of annotation

Users are able to use HTML code on the content of the annotation; this feature brings the same advantages like in video articles, begin and end time define the time annotation points to the video on the video article.

7.5.3 Link to video related information using time parameters.

Users can relate video related information in video articles and in annotations and define video functionalities.

User can link to video related information in the content of video articles and in the content of annotations, users are able to control the start and end time of video related information.

Users need to generate a command (pattern) in order to link video related information, the command as the following form:

```
{envideo path_of_video_file begin_time end_time}
```

Following is an example:

```
{envideo cat.flv 10 60}
```

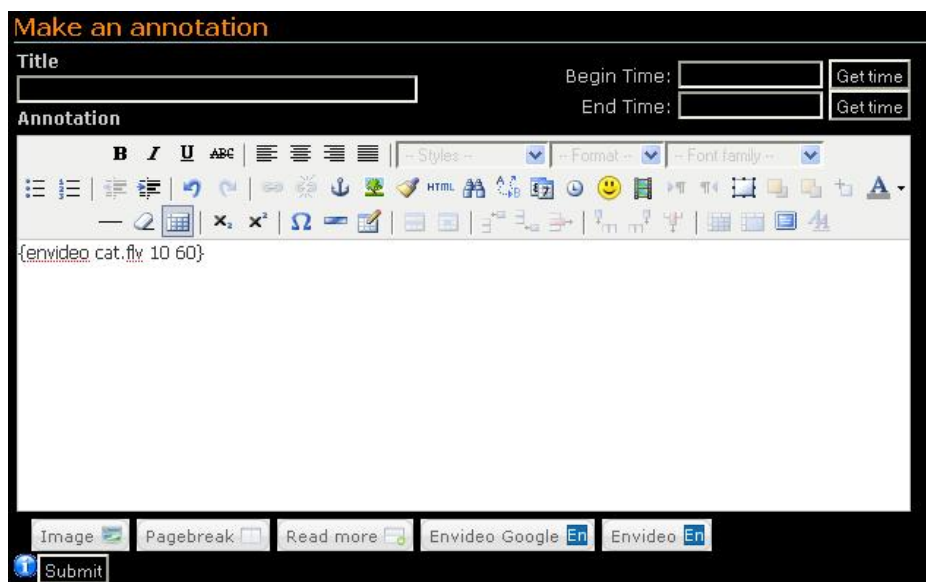


Figure 7.6 – Video Related Information's Command

Following is a description of the several parts that compose the link video command:

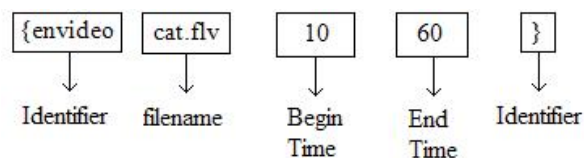


Figure 7.7 – Parts of the Video Related Information Command

Identifiers

The identifiers tell the system that the user wants to link video related information; the identifiers must be included always.

Filename

The name of the video that the user wants to link.

Begin Time

At what time the related video should start playing, it must be in seconds.

End Time

At what time the related video should stop playing, it must be in seconds.

In order to avoid the user to remember the command structure a plug-in was built, when the button is pressed a template of the command is inserted, this button is included in order to give a better user interface to the user:



Figure 7.8 – Video Related Information Command Button

7.6 Information searching and retrieval

User can search for information inside annotations, this is accomplished by using the search field on the front end, and users only need to type the information they want to search.

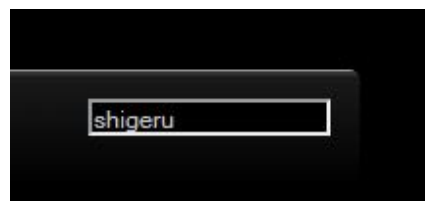


Figure 7.9 – Search Field

7.7 Fast information searching and retrieval inside video media

The Envideo system lets users search and retrieve information faster by using annotations that implement time parameters. Users that find annotations that incorporate time parameters will also find the position inside the video. Moreover users will be able to watch the video at the time stated in the annotations. This also applies to video related information.

7.7.1 Case Scenario

There is a video article with one video (history of videogames made by Discovery Channel) of 46 minutes approximately, and inside the video article there is also a link to another video (history of videogames

made by fans) of 110 minutes approximately, that makes in total almost two hours of video media to watch in video article.

There are already some annotations made by several users inside the video article.

A user that is interested in the topic, wants to find the answer to the question who is Shigeru Miyamoto (a world famous game developer)? For homework at University, however, he discovers that all the video information makes in total almost two hours.

He then types the string "Shigeru Miyamoto" (without quotes) in the search field on the website, and he sees an annotation related to Shigeru Miyamoto, he opens it.

He sees on the Annotation (made by other user) the answer, moreover he can see that there are links that points to several parts in both videos where Shigeru Miyamoto is speaking.

In the end, the user found an annotation and watch only the parts in both video where Shigeru Miyamoto is talking, in comparison of watching both videos and taking two hours.

7.8 Advantages

The Envideo system lets the users extend the content of video media, it provides the mechanisms to related video media with the possibility to incorporate time parameters, this allow Envideo to find and retrieve information inside and outside media faster.

Users normally (using existing solutions) have to watch hours of video media in order to get information from video because video media information is time-based (cf. 1), and in order to search inside video media, they use scenes (in case of DVD) or the SEEK function available in almost every video player.

Users using Envideo system can read an article about a video (if they desire), watch the video, read annotations, search inside related information and moreover when the extended information implement time parameters, users can jump directly to the video interval the annotations are pointing.

7.9 Disadvantages

The search and retrieval of inside video media is dictated by the quality of the video articles and annotations, poor annotations will lead to poor results in search.

However, inside a company, where there is quality control of content (for example in newspapers), this disadvantage can be overcome.

Another disadvantage is that extended information must have time parameters and these are introduced by the users. A solution for that is to encourage users to apply time parameters in their annotations by offering a novel and good looking user interface. The approach used in this work was to provide a user interface for easy introduction of time parameters in annotations and also the inclusion of a marquee to display only the annotations that include time parameters on the time specified inside the annotations.

7.10 Comparison with related systems

Following there is comparison between Envideo and the related systems:

Characteristic	Annota a	Viddler	YouTube	Hyper- Hitchcock	EspiritUs'	M4Note	Envideo
Annotation of video objects	No	Yes	Yes	Yes	No	Yes	Yes
Annotation of web content	Yes	No	No	No	Yes	No	No
Annotations can point to a certain time on the video	No	Yes	Yes	Yes	No	Yes	Yes
Possibility of using HTML on the annotations	No	Partially	Partially	No	No	No	Yes
Possibility to setup the start and end time of the annotation	No	No (Only the beginning time)	Partially (Only on graphical annotations)	Yes	No	No	Yes
Annotations can point to videos	No	Partially (Only to videos recorded on webcam)	Yes	Yes	No	No	Yes
Annotations can point to other videos and define the start and stop time on the linked videos	No	No	No	No	No	No	Yes

Table 7.2 – Comparison between Annotation Systems and Envideo

The Envideo system is a web tool; it is design to provide a user with an easy user interface but also with the necessary tools to allow the user create complicated extension of information on video media.

7.11 Conclusions

The system that was developed for this thesis work was presented, it has the following goals:

- Extend video media information by generating video articles and annotations
- Link related information

- Search and retrieval inside video media information by searching inside annotations

The author of this system argues that by searching inside the annotations and video related information that have time parameters will allow the user to find and retrieve information inside video media faster.

The following chapter will present the first development phase of this thesis project; it will show how the idea of searching inside video media annotation was conceived and the first prototype.

References

Akkana Peck, "Beginning GIMP: From Novice to Professional", Apress, New York, 2006

Alan Marsden, Harriet Nock, Adrian Mackenzie, Adam Lindsay, John Coleman and Greg Kochanski, "ICT Tools for searching, annotation and Analysis of audiovisual Media", Lancaster Institute for the contemporaray Arts, and Institute for cultural research, Lancaster University, Phonetics Laboratory, University of Oxford, October 2006

Baecker, R., Grudin, J., Buxton, B. and Greenberg, S. "Readings in Human Computer Interaction: Towards the Year 2000. Second edition", Morgan-Kaufman, 1995.

Bernd Bruegge, Allen H. Dutoit, „Object-Oriented Software Engineering Using UML, Patterns and Java 2nd Edition", Pearson Education/ Prentice Hall, Upper Saddle River, 2004

Dan Rahmel, "Professional Joomla!", Wiley Publishing, Inc, Indianapolis, Indiana, 2007

Frank Shipman, Andreas Girgensohn, Lynn Wilcox, "Hypervideo Expression: Experiences with Hyper-Hitchcock", Department of Computer Science & Center for the study of Digital Libraries Texas A&M University College Station – FX Palo Alto Laboratory, Inc. USA, 2005

Frank Shipman, Andreas Girgensohn, Lynn Wilcox, "Hyper-Hitchcock: Authoring Interactive Videos and Generating Interactive Summaries", Department of Computer Science & Center for the study of Digital Libraries Texas A&M University College Station – FX Palo Alto Laboratory, Inc. USA, 2005

Hagen Graf, "Building Websites with Joomla! 1.5 Beta 1", Packt Publishing Ltd., Birmingham, February 2007

Jakob Nielsen, "Iterative user-interface design." Computer, 26 (11):32–41, 1993

James Kennard, "Mastering Joomla! 1.5 Extension and Framework Development", Packt Publishing Ltd., Birmingham, November 2007

Jason Beaird, "The principles of Beautiful Web Design", SitePoint Pty. Ltd., VIC, January 2007

Jay Greenspan, Brad Bulger, "MySQL/PHP Database Applications", M&T Books, Foster City, CA, 2001

Jim Keogh, "Javascript Demystified", McGraw-Hill/Osborne, New York, 2005

Joseph LeBlanc, "Learning Joomla! 1.5 Extension Development Creating Modules, Components, and Plug-ins with PHP", Packt Publishing Ltd., Birmingham, May 2007

Larry Ullman, "PHP 5 Advanced", Peachpit Press, Berkely, CA, 2007

Marco Fernandes, Miguel Alho, Joaquim Arnaldo Martins, Joaquin Sousa Pinto, Pedro Almeida, "Web Annotation System Based on Web Services", University of Aveiro, Portugal, 2005

References

Matt Zandstra, "PHP Objects, Patterns, and Practice, second edition", Apress, New York, 2008

Matthias R. Brust and Steffen Rothkugel, Faculty of Science, Technology and Communication, University of Luxemburg, 2007

Murtha Baca, "Introduction to metadata", Getty publications, November 11, 2008

Oliver Aubert, Yannick Prié, "Creating and Sharing Hypervideos with Advene", Lyon 1 University, 2005

Oliver Aubert, Yannick Prié, "Advene: an open-source framework for integrating and visualizing audiovisual metadata", Lyon 1 University, 2007

Ronald Schroeter, Jane Hunter and Douglas Kosovic. "FilmEd – Collaborative Video Indexing, Annotation and Discussion Tools Over Broadband Networks", DSTC, The University of Queensland, 2004

Rudinei Goularte, Jose A. Camacho-Guerrero, Valter R. Inacio Jr, Renan G. Cattelan, Maria da Graca C. Pimentel. "M4Note: A Multimodal Tool for Multimedia Annotations", Mathematics and Computing Institute – University of Sao Paulo, 2004

Simon Collison, "Beginning CSS Web Development: From Novice to Professional", Apress, New York, 2006

Steve Suehring, "MySQL Bible", Wiley Publishing , Inc, New York 2002

Takeshi Sannomiya, Toshiyuki Amagasa, Masatoshi Yoshikawa, and Shunsuke Uemura, "A framework for sharing Personal Annotations on Web Resources using XML", Graduate School of Information Science Nara Institute of Science and Technology Japan, 2001

Valerie Henderson, Rebecca E. Gringter, and Thad Starner, "Electronic Communication by Deaf Teenagers", Georgia Institute of Technology Gvu Center, College of Computing Atlanta, 2005

Web References

- [WR-1] YouTube, "YouTube Annotation tool", <http://www.youtube.com>
- [WR-2] Viddler, "Annotation tool of viddler", <http://www.viddler.com>
- [WR-3] Annotea Project, <http://www.w3.org/2001/Annotea/>
- [WR-4] Amaya Project, <http://www.w3.org/Amaya/Overview.html>
- [WR-5] Annozilla Project, <http://annozilla.mozdev.org/>
- [WR-6] Annotea Test Server, <http://annotest.w3.org/annotations>
- [WR-7] Alexa The web information company, <http://www.alexa.com>
- [WR-8] The idea shower, <http://www.ideashower.com/learned/how-to-disable-will-not-be-installed-because-it-does-not-provide-secure-updates-warning-in-firefox-3/>
- [WR-9] Vibelle Web Portal <http://www.vibelle.de>
- [WR-10] What is Vibelle <http://www.vibelle.de/content/blogcategory/193/256/>
- [WR-11] GNU General Public License <http://www.gnu.org/licenses/gpl.html>
- [WR-12] World Wide Web Consortium <http://www.w3.org/>
- [WR-13] Top 50 most important open source projects <http://www.uversainc.com/download/top50.pdf>
- [WR-14] Joomla! 1.6 Future user management options
<http://community.joomla.org/blogs/community/739-has-it-been-worth-the-wait.html>
- [WR-15] JW FL player <http://www.longtailvideo.com/players/jw-flv-player/>
- [WR-16] The ACM digital library <http://portal.acm.org/portal.cfm>
- [WR-17] Top Ten Web Design Mistakes of 1999 <http://www.useit.com/alertbox/990530.html>
- [WR-18] DESIRE Research Team <http://desire.isk.rwth-aachen.de/>
- [WR-19] Fraunhofer FIT <http://fit.fraunhofer.de/>

Appendix A

Scenarios

A.1 Scenario 1

Rene is 28 years old, he is informatics, he is acquaintance with technology and he is an extreme user, he works in Fraunhofer institute in FIT department concerning technology projects.

Rene sometimes discuss videos concerning projects relevant to his work with his team members, the way he has already done it has been the following way:

He sees an interesting video of a conference of about one hour duration, he sends the video link to his colleagues by using the emailing list of the working group.

After some time he receives responses from several team members by using the same mean: the emailing list, it is hard for Rene to organize the information that comes within the emails, moreover when a colleague makes a comment pointing at a specific time on video, Rene must go back to the link and manually go to the specific part his colleague is referring to, this is obviously because Rene cannot remember the whole hour of video second by second.

However, Rene has just read about the new extend video annotation tool and he wants to give it a try, so this time, he sends another video link to all his colleagues as well as access to the system, but this time the video is on the extended video system and the video is two hours long, he also makes some comments on the video. After some hours, he checks the extended video system and he finds out that some colleagues have already made annotations on the video, during the incoming days, Rene can see that there are many annotations of his colleagues, he is also satisfied to see that when a working colleague points out at a specific time on the video, Rene can click on it and go directly where his team member has made his comment, but not only that, he also realizes that his colleagues have established the duration of their comments.

After some days he sees many comments on the video, however, he wants to make one more annotation, that is very important, so he makes it, then when he is done with it, he sends the link to all the members, so his working colleagues can see the comment of Rene and the video at the specific time and duration.

Rene is very happy about using the system.

A.2 Scenario 2

Hörst is 32 years old, he is deaf, he is not acquaintance with technology and his knowledge in computers is very limited.

Hörst is a vibelle moderator; he generates content for the vibelle portal

Hörst has a new video concerning the deaf Olympics 2008 of 15 minutes long, and he wants to include it on the vibelle, he does it through the use of the Joomla CMS:

However, after doing his task, he realizes that there is a part on the video that shows an email address in order to get in contact with the organizers of the Deaf Olympics, and he remembers that there is already a video concerning the editing and sending of an email on the vibelle portal. So he looks for this video and finds it, then he sees that the video is 45 minutes long due to the introduction of the email, how it works, etc, but he is only interested on the editing and sending process, therefore he checks at what time starts the part that he is interested, and for how long, then he goes to the video concerning the deaf Olympics, uses the extended video tool, and at the time concerning the contact information, he writes a comment like following: "click this link if you do not know how to make an email" and then he makes a link to the video concerning the email editing process, on the process he sets up:

The duration of his comment.

The link to the video.

At what time the other video should start playing.

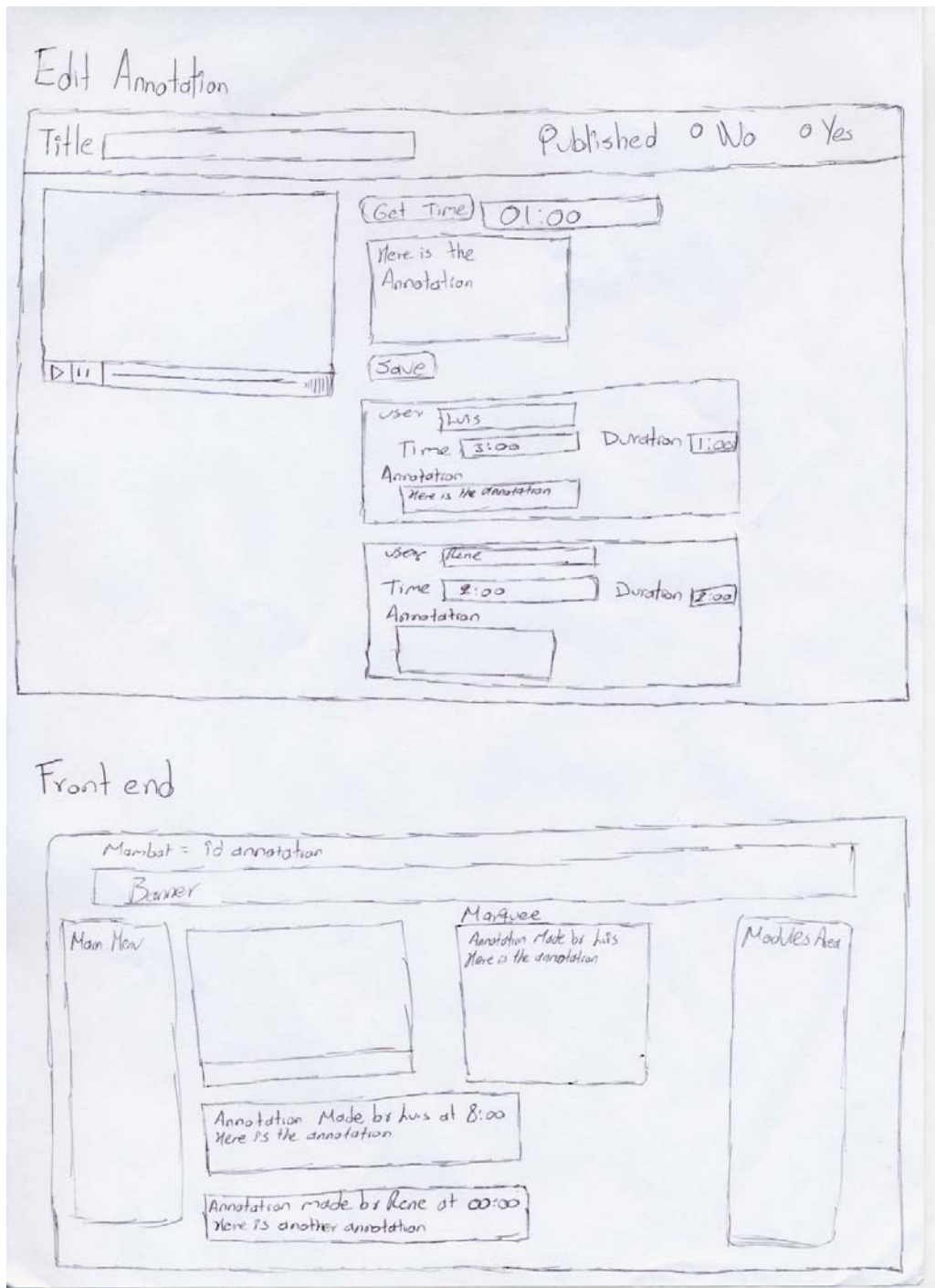
For how long the other video should play.

When he is done, he checks the comment he just made, and he can see that on the deaf Olympics video on the part concerning the contact information, the link to the another video appears, he clicks it and the other video starts playing at the specific time he setup before, he is amazed at this new feature and happy with it.

Appendix B

Paper prototypes

B.1



B.2

Envideo

Annotations

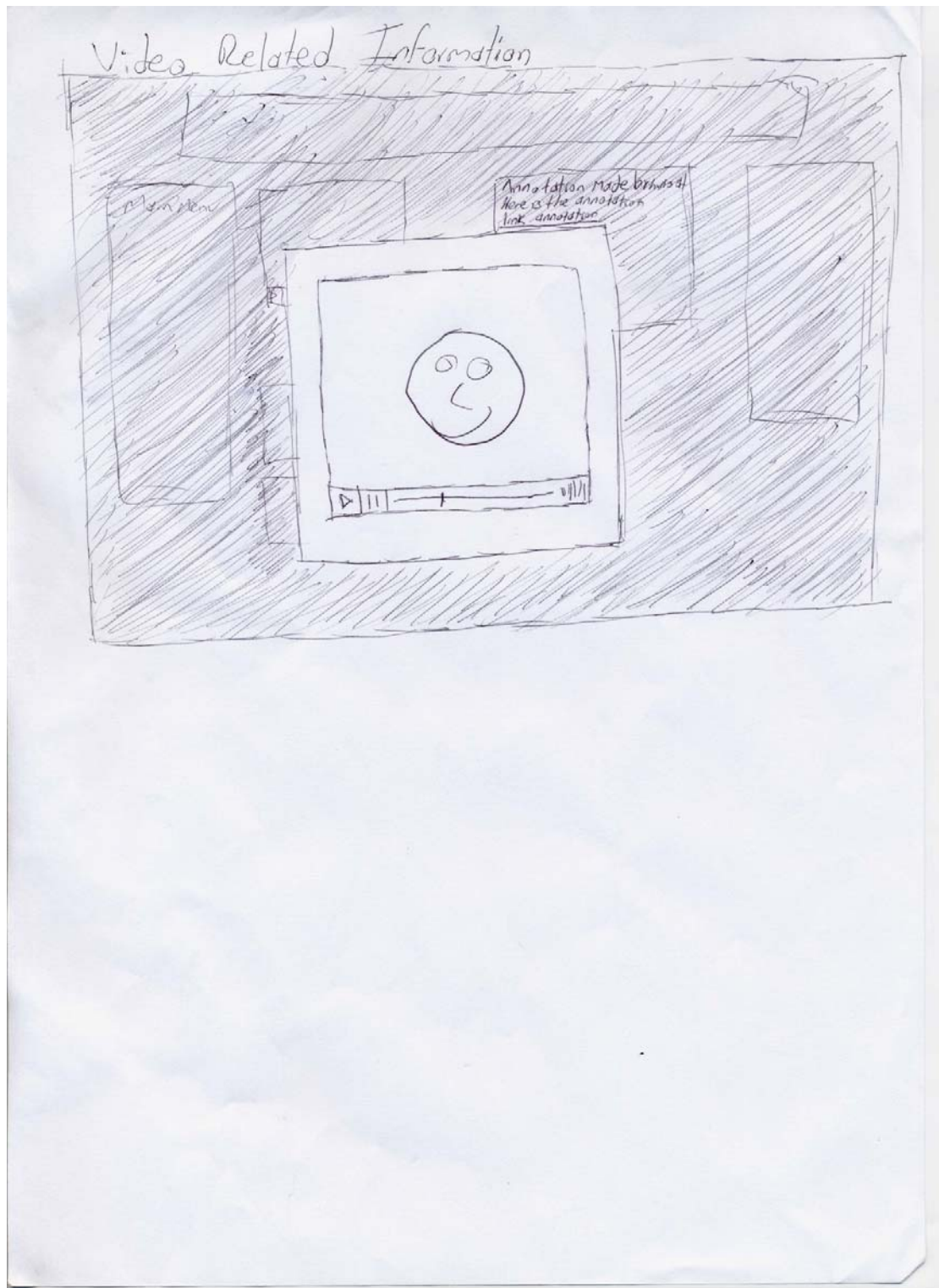
<input type="checkbox"/>	Title	Author	Date	video link	Publish

New Annotation

Title Published ☐ No ☐ Yes

Here is the annotation

B.3



Appendix C

C.1 Envideo Quick Start – User Manual

Envideo Quick Start

Luis Ramon Santana Alvarez

Index

1. Introduction	2
2. First steps with the Envideo System	2
3. Creation of Video Articles	3
4. Creation of Annotations	5
5. Creation of Video Related Media	6
6. Searching for information	8

1. Introduction

Hello there, this quick start manual that will help you to understand and setup the Envideo system quick and easy.

Envideo is a component developed for Joomla! CMS. The Envideo system provides a platform to extend content of video media; there are three methods for extending content:

- Creation of video articles
- Annotation on video articles with time parameters
- Link to video related information and control the video functionalities of the linked media with the help of time parameters.

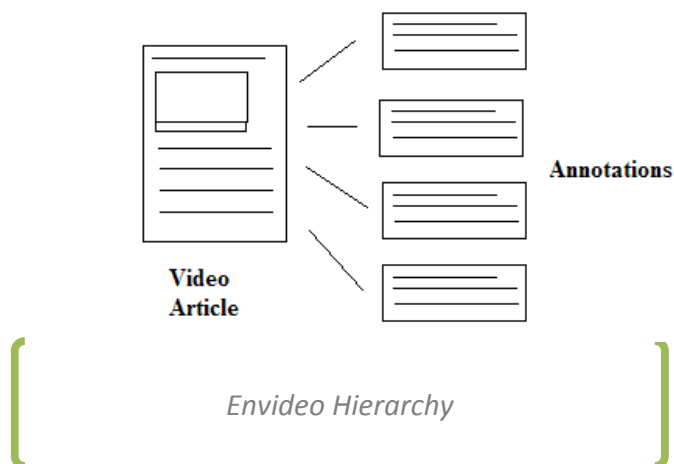
Users normally (using existing solutions) have to watch hours of video media in order to get information from video because video media information is time-based and in order to search inside video media, they use scenes (in case of DVD) or the SEEK function available in almost every video player.

Users using Envideo system can read an article about a video, watch the video, read annotations, and search inside related information.

The real power of Envideo comes when users include time parameters in their annotations (video articles or annotations), then Envideo can take the users to the video part that is specified on the annotations, otherwise the user will have to seek or watch entire hours of video in order to search for video information. Enough talk; let's start using the Envideo system.

2. First steps with the Envideo system

There are two main components that compose envideo system, all the functionality is inside these two components.



Annotations are made on video articles; therefore the main step is to create a video article then annotations.

3. Creation of video articles

Articles can only be created using the administration of the Envideo system which is on the back administration of Joomla!, therefore only users with sufficient privileges can create video articles. A video article contains one video; users extend the information on the video by generating an article that uses text and HTML code, therefore it is possible to include images and links to other web resources, an article can also contain links to other video related information that incorporate time parameters and control their video functionalities.

Users must follow the next steps in order to generate video articles:

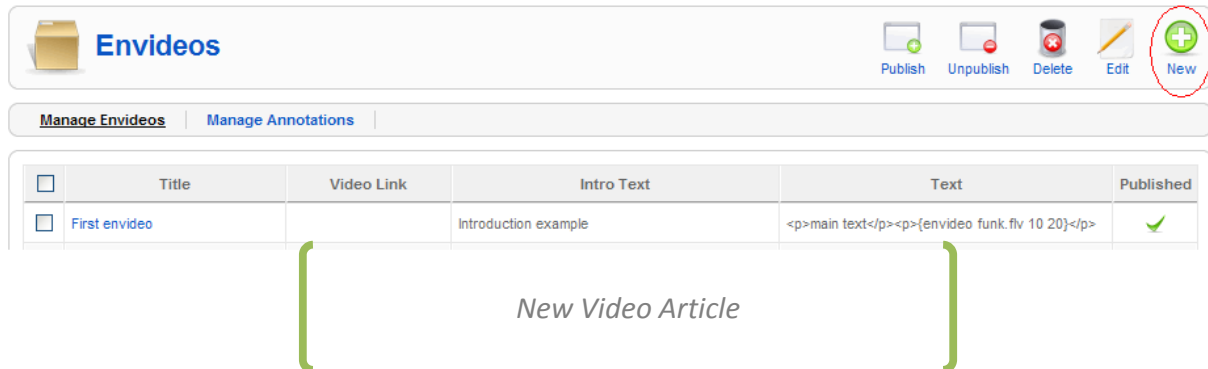
1. Login in Joomla! back administration



2. Click on Envideos from the components menu



3. Click on Manage Envideo > New



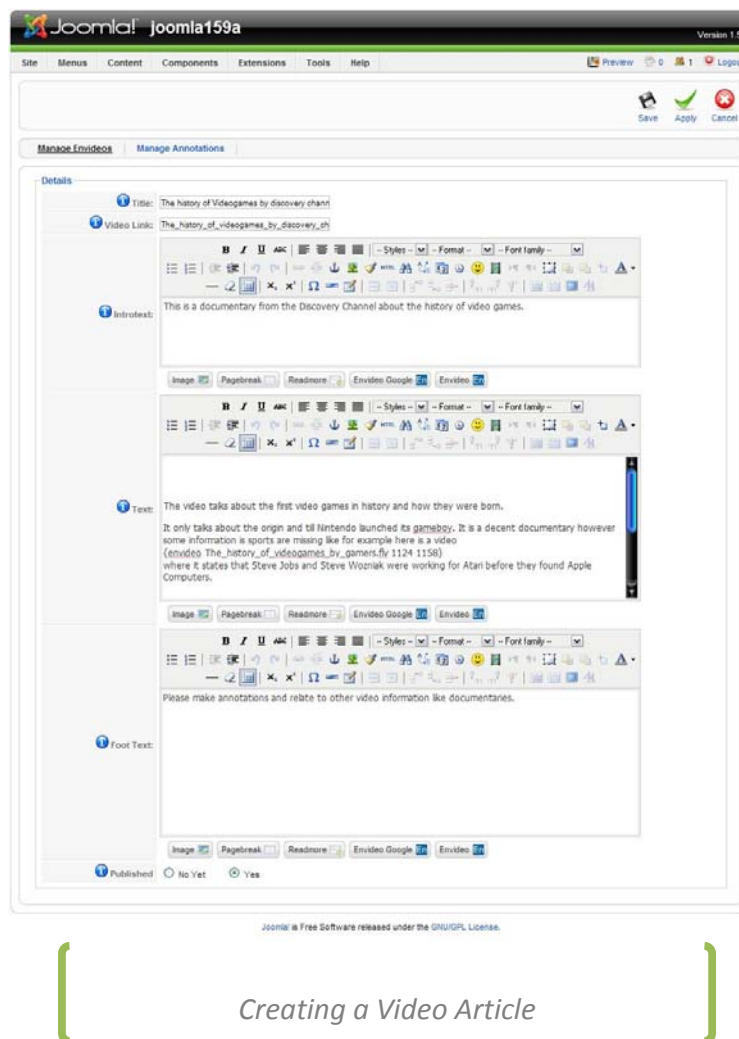
Envideos

Manage Envideos | Manage Annotations

	Title	Video Link	Intro Text	Text	Published
<input type="checkbox"/>	First envideo		Introduction example	<p>main text</p><p>{envideo funk.flv 10 20}</p>	✓

New Video Article

4. Fill in the fields and make sure that the “published” field is selected to “yes”



Joomla! 1.5.9

Site | Menus | Content | Components | Extensions | Tools | Help

Save | Apply | Cancel

Manage Envideos | Manage Annotations

Details

Title: The history of Videogames by discovery chann

Video Link: The_history_of_videogames_by_discovery_ch

Introtext: This is a documentary from the Discovery Channel about the history of video games.

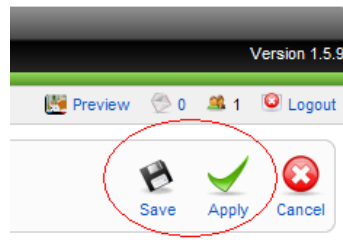
Text: The video talks about the first video games in history and how they were born. It only talks about the origin and till Nintendo launched its gameboy. It is a decent documentary however some information is sports are missing like for example here is a video (envideo The_history_of_videogames_by_gamers.flv 1124 1158) where it states that Steve Jobs and Steve Wozniak were working for Atari before they found Apple Computers.

Foot Text: Please make annotations and relate to other video information like documentaries.

Published: ☐ No Yet ☒ Yes

Creating a Video Article

- Click on save or apply if you still want to continue with changes.



Save or Apply

4. Creation of Annotations

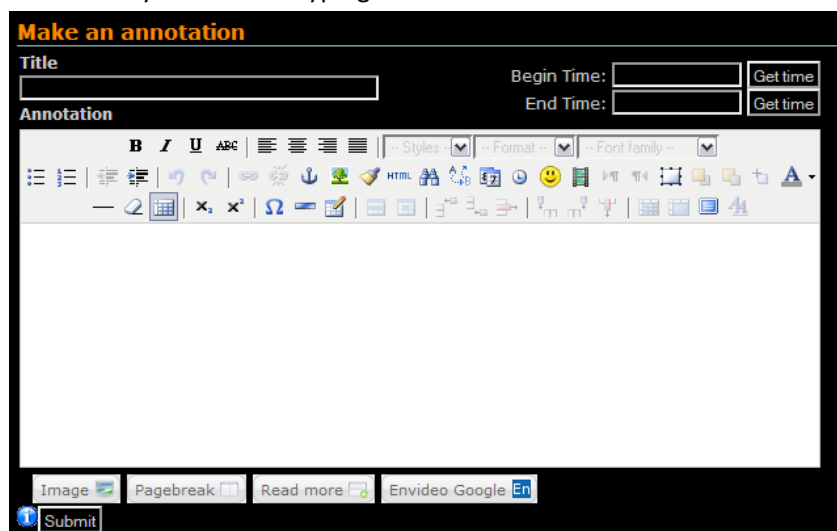
Annotations are used to extend the information inside video articles. Annotations can be created on the front-end and of the back-end, they are normally created on the front-end therefore I am going to explain the creation on annotations on the front-end.

- Annotations are created inside video articles, for that you must be inside a video article by clicking one from the main list:

First envideo
Introduction example
The history of Videogames by discovery channel
This is a documentary from the Discovery Channel about the history of video games.
abs in 8 min
this video is about 11 abs exercises in 8 min

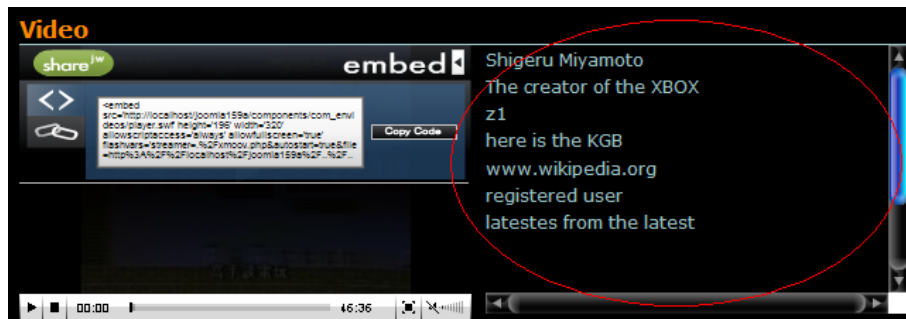
Listing of video articles

- Inside a video article you can start typing on the annotation area



Making an annotation

3. One remark to take into account is at the moment of using the “get time” buttons, the begin time must always be lower than the end time, otherwise “submit” button will be disabled, if you do not setup a time for your annotation, this will be ignored on the marquee option.



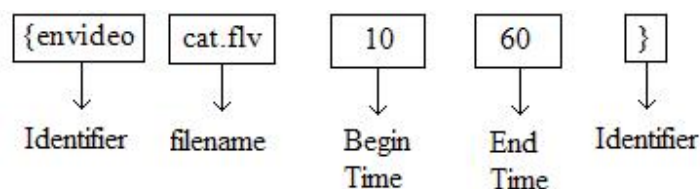
Annotations with time parameters on the marquee

5. Creation a video related media

Users can relate video related information in video articles and in annotations and define video functionalities.

User can link to video related information in the content of video articles and in the content of annotations, users are able to control the start and end time of video related information.

Users need to generate a command (pattern) in order to link video related information, the command as the following form:



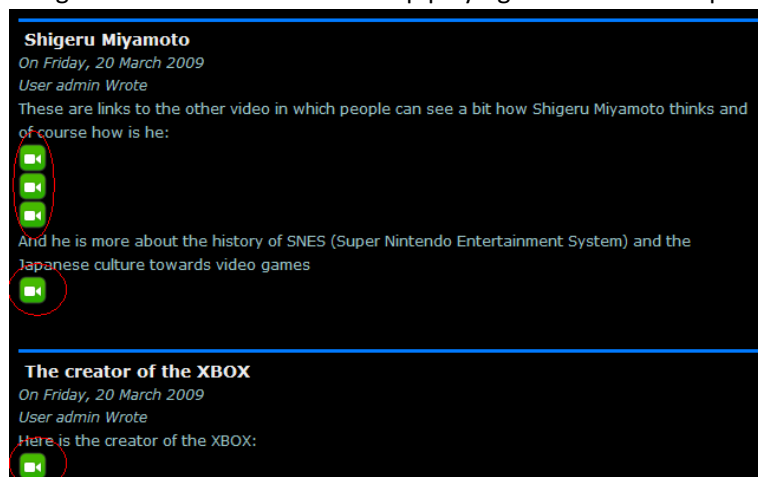
Command for including video related information

Following is an example:

Writing command for including video related information

This means that we want to link a video that is called “cat.flv” and it will start playing at the second 10 and will stop at the second 60.

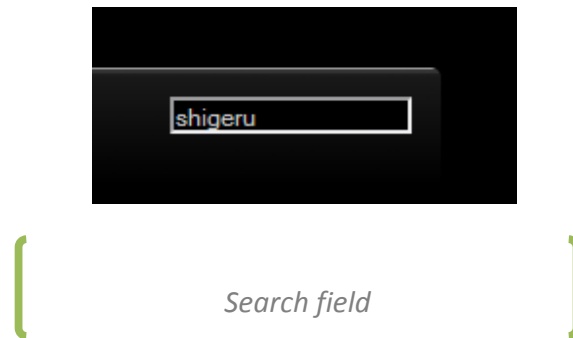
At the moment of saving the video article or the annotation an icon of a camera will be displayed and at the moment of clicking it the video will start and stop playing at the moment specified on the command.



Icons of video related information

6. Searching of information

Searching of video articles or annotations is an easy task, you just need to type on the search field the information that you are looking for:



The results will be displayed using the standard format from Joomla!

