



Universidade Nova de Lisboa  
Faculdade de Ciências e Tecnologia  
Departamento de Informática

## **Anotações Partilhadas Usando Quadros Interactivos**

David Daniel da Silva Dias

*Dissertação apresentada na Faculdade de Ciências e Tecnologia da Universidade Nova de Lisboa, para a obtenção do grau de Mestre em Engenharia Informática.*

Orientador: *Professor Doutor Nuno Manuel Robalo Correia*

Fevereiro de 2010

Nº do aluno: 26888

Nome: David Daniel da Silva Dias

Título da dissertação:

NotesBoard – Anotações Partilhadas Usando Quadros Interactivos.

Palavras-Chave:

- Multimédia
- Anotações
- Vídeo
- Trabalho Colaborativo
- Quadros Interactivos

Keywords:

- Multimedia
- Annotations
- Video
- Collaborative Work
- Whiteboards

## **Agradecimentos**

---

Gostaria de agradecer a todos os que me ajudaram nesta etapa da minha vida, em particular ao Prof. Doutor Nuno Correia por toda a ajuda durante toda a dissertação, sem a sua ajuda, compreensão e disponibilidade, esta dissertação dificilmente chegaria ao patamar ao qual chegou.

Gostaria ainda de agradecer aos meus pais, irmão e namorada por todo o apoio prestado durante este ano conturbado da minha vida. Sem todo o apoio anímico e solidário dado teria sido impossível chegar ao fim desta etapa e realizar esta dissertação de uma forma célere com a qualidade pretendida.

Também gostaria de agradecer a todos os meus colegas e amigos por toda a ajuda e solidariedade dada durante a elaboração desta dissertação. Desde as ajudas mais técnicas até ao facto de serem meus amigos, ajudaram em muito o desenrolar deste trabalho.

## Resumo

---

A evolução do ensino encontra-se fortemente correlacionada com as necessidades da sociedade. Em todas as vertentes desta relação, o ensino tem procurado adaptar-se aos novos tempos e o caso da tecnologia não é excepção. A utilização de tecnologia num contexto de ensino é fomentada pela necessidade de trocar informação de forma simples e concisa.

O trabalho desenvolvido integra uma interface colaborativa de anotação de vídeo, utilizando uma abordagem de partilha de informação. É necessário pensar para além da comunicação professor/aluno e desta forma a aplicação desenvolvida tem como objectivo trazer para os alunos a possibilidade de partilharem informação validada pelo professor, como acontece hoje em dia com a troca de apontamentos. O professor mantém o seu papel clássico de mediador recorrendo a um quadro interactivo que será o centro do sistema, mantendo a tradicional forma de partilhar conhecimento com os alunos e aumentando o leque das soluções tecnológicas disponíveis. No essencial, o projecto centra-se numa solução intuitiva para anotações em vídeo, facilmente operável através de um dispositivo táctil de toque único, em particular nos quadros interactivos.

As anotações permitem a contextualização do vídeo, através das mesmas um determinado vídeo pode ganhar interpretações diferentes perante uma comunidade de utilizadores. Para além da vertente usual das anotações, definidas estaticamente numa posição absoluta no vídeo, este trabalho aborda a utilização de anotações dinâmicas, que permitem que uma dada anotação siga um objecto escolhido pelo utilizador.

## **Abstract**

---

The teaching evolution is strongly correlated with society needs. The teaching process has been changing and adapting and the introduction of technology is not an exception. The requirement to use technologies is encouraged by the need to reach the students' attention through new techniques and cultural approaches.

The project integrates a collaborative interface for video annotations, using recent file sharing approaches. It goes beyond the connections between professor and students, in order to bring for students the possibility to share information validated by the professor. Today this option is guaranteed by physical paper sharing. The teacher maintains the classical mediator role in this system, using an interactive whiteboard that will be the system's central element. This role will enable typical relationships and improves the available technologies solutions range. Essentially, the project is focused on an intuitive solution for video annotations which can be easily operated with a single touch device like a interactive whiteboard.

Annotations represent the video contextualization capacity, and through it a video could acquire different interpretations by a user's community. A usual annotation is defined in an absolute position, but in this work this concept was extended with dynamic annotation usage which is useful to perform video object tracking.

## Acrónimos

---

<b>ASF</b>	Apache Software Foundation
<b>AVI</b>	Audio Video Interleave (multimédia container format)
<b>CVA</b>	Colaborative Vídeo Annotator (tipo de ficheiro das anotações da aplicação)
<b>DNS</b>	Domain Name System
<b>FLV</b>	Flash Video (video container format)
<b>FTP</b>	File Transfer Protocol
<b>GEPE</b>	Gabinete de Estatística e Planeamento da Educação (Ministério da Educação)
<b>GIF</b>	Graphics Interchange Format
<b>HTTP</b>	Hypertext Transfer Protocol
<b>IBM</b>	International Business Machines
<b>IP</b>	Internet Protocol
<b>JMF</b>	Java Media Framework
<b>JNI</b>	Java Native Interface
<b>MIT</b>	Massachusetts Institute of Technology
<b>MS</b>	Microsoft
<b>MP4</b>	MPEG-4 Vídeo File
<b>MPEG</b>	Moving Picture Experts Group
<b>P2P</b>	Peer To Peer
<b>PNG</b>	Portable Network Graphics
<b>PC</b>	Personal Computer
<b>RIAs</b>	Rich Internet Applications
<b>RPC</b>	Remote Procedure Call
<b>SMTPE</b>	Society of Motion Picture and Television Engineer
<b>TV</b>	Television
<b>XML</b>	eXtensible Markup Language

# Índice

---

<b>1</b>	<b>Introdução .....</b>	<b>11</b>
1.1	Motivação .....	11
1.2	Solução Proposta .....	14
1.3	Principais Contribuições Propostas .....	16
<b>2</b>	<b>Trabalho Relacionado .....</b>	<b>17</b>
2.1	Anotações .....	17
2.1.1	Experimental Video Annotator (EVA).....	17
2.1.2	YouTube .....	18
2.1.3	VAnnotator .....	19
2.1.4	VideoStore .....	20
2.1.5	Sumário.....	21
2.2	Sistemas para Apresentações.....	21
2.2.1	ClassRoom Presenter .....	22
2.2.2	ConferenceXP.....	23
2.2.3	DyKnow .....	24
2.2.4	Sumário.....	25
2.3	Quadros Interactivos.....	25
2.3.1	NotateIt .....	26
2.3.2	Mimio Studio.....	27
2.3.3	Sumário.....	27
2.4	Partilha de Ficheiros .....	28
2.4.1	Servidores HTTP .....	28
2.4.2	Redes Peer To Peer.....	29
2.4.3	Resumo .....	30
<b>3</b>	<b>Descrição do Sistema .....</b>	<b>31</b>
3.1	Modelo de Utilização.....	31
3.1.1	Modos de funcionamento .....	32
3.2	Anotações .....	34

3.2.1	Tipos de Anotações .....	35
3.2.2	Representação .....	38
3.2.3	Sincronização.....	39
3.2.4	Anotações Dinâmicas .....	39
3.3	Descrição da Interface .....	40
3.3.1	Áreas de Opções .....	41
3.3.2	Formulários.....	47
3.3.3	Notificações.....	50
3.4	Arquitectura .....	52
3.5	Aspectos de Implementação .....	55
3.5.1	Modelo de Classes .....	55
3.5.2	Suporte Colaborativo.....	60
3.5.3	Anotações Dinâmicas .....	64
3.5.3.1	Possíveis Tecnologias para Implementação .....	64
3.5.3.2	Módulo de Anotações Dinâmicas.....	66
3.5.3.3	Integração com a interface JavaFX .....	68
<b>4</b>	<b>Considerações Finais .....</b>	<b>69</b>
4.1	Conclusões.....	69
4.2	Trabalho Futuro .....	72
<b>5</b>	<b>Bibliografia.....</b>	<b>74</b>
<b>6</b>	<b>Anexos.....</b>	<b>77</b>
6.1	Tipos de Dados Próprios .....	77
6.2	Schema da Árvore do Ficheiro de Anotações .....	78
6.3	Tipos de Aparência.....	79



## Índice de Imagens

---

Figura 1.1 – YouTube usado como método de aprendizagem. ....	12
Figura 1.2 – Publicidade através de vídeo no Metacafe. ....	12
Figura 1.3 - Percentagem de escolas que têm quadros interactivos e número médio de quadros interactivos por escola. ....	14
Figura 1.4 – Infra-estrutura física da solução. ....	15
Figura 2.1 – Anotações do YouTube utilizadas como hiper ligações. ....	19
Figura 2.2 – Anotações de texto no YouTube. ....	19
Figura 2.3 – Excerto da interface do VideoStore. ....	20
Figura 2.4 – Visão geral do Classroom Presenter. ....	22
Figura 2.5 – Acções disponibilizadas no ConferenceXP. ....	24
Figura 2.6 – Notas privadas no programa DyKnow. ....	25
Figura 2.7 – Visão global do NotateIt. ....	26
Figura 2.8 – Visão global do Mimio Studio. ....	27
Figura 2.9 – Mimio Studio <i>Toolbox</i> . ....	27
Figura 2.10 – Interacção entre os elementos do BtFiler. ....	29
Figura 3.1 - Visão global da janela principal. ....	41
Figura 3.2 - Barra de Menu com o menu <i>Annotator</i> seleccionado. ....	41
Figura 3.3 - Tipos de anotações existentes. ....	43
Figura 3.4 - Lista de utilizadores de uma sessão colaborativa. ....	44
Figura 3.5 - Listagem das anotações que estão a ser utilizadas pelo utilizador. ....	44
Figura 3.6 - Panorama temporal das anotações na área de planificação. ....	45
Figura 3.7 - Área de Criação e Visionamento de Conteúdos. ....	46
Figura 3.8 - Formulário de criação de uma aula. ....	48
Figura 3.9 - Formulário de edição de uma anotação do tipo Rectângulo. ....	49
Figura 3.10 - Opções disponíveis na área de preferências. ....	49
Figura 3.11 - Barra de notificação com a listagem de eventos expandida. ....	50
Figura 3.12 - Notificação referente à recepção de um ficheiro originário do coordenador. ....	51
Figura 3.13 - Notificação de recepção de uma anotação. ....	51

Figura 3.14 - Arquitectura global da aplicação. ....	52
Figura 3.15 - Organização das classes que herdam da anotação. ....	55
Figura 3.16 - Organização dos objectos Anotação no Gestor de Anotações. ....	57
Figura 3.17 - Troca de informação entre o Coordenador e um Participante. ....	57
Figura 3.18 - Classes que podem herdar da classe Reprodutor Multimédia. ....	58
Figura 3.19 - Gestão interna de vídeos no Gestor Multimédia. ....	59
Figura 3.20 - Pacotes da aplicação e respectivas interacções. ....	61
Figura 3.21 - Obtenção dos conteúdos por parte do aluno. ....	62
Figura 3.22 - Exemplo de uma anotação partilhada duas vezes com atributos distintos. ....	63
Figura 3.23 - Diagrama representativo do módulo de anotações dinâmicas. ....	66
Figura 3.24 - Output gerado pelo módulo. ....	68

## Índice de Tabelas

---

Tabela 3.1 - Atributos globais da anotação. ....	35
Tabela 3.2 - Atributos do tipo de anotação rectângulo.....	36
Tabela 3.3 - Atributos do tipo de anotação textual.....	36
Tabela 3.4 - Atributos do tipo de anotação elipse. ....	37
Tabela 3.5 – Atributos do tipo de anotação Linha.....	37
Tabela 3.6 - Parâmetros do módulo de anotações dinâmicas.....	67

# 1 Introdução

O tema desta dissertação é a utilização e partilha de anotações de vídeo num contexto educacional e a forma como este mecanismo poderá contribuir para uma maior interacção entre os elementos presentes numa sala de aula. O ponto de partida é uma aula presencial tradicional em que se assume que os alunos e o professor têm consigo um computador pessoal ou equipamento equivalente e estão ligados em rede. Sobre essa infra-estrutura procura-se igualmente integrar a utilização de quadros interactivos, enquanto dispositivos que permitem ao professor, ao expor a matéria na sala de aula, ter com os alunos uma interacção idêntica à que obteria a partir do seu computador.

O objectivo principal da tese passa assim por obter uma solução que forneça uma interface amigável e de fácil interacção para que de uma forma transparente, com a utilização do quadro ou computadores pessoais, esta infra-estrutura permita obter uma maior interactividade sem se tornar intrusiva ou limitadora do funcionamento normal da aula. Pretende-se que sobre os materiais de suporte à aula possam ser feitos e trabalhados diversos tipos de anotações, que dependendo do contexto podem tomar a forma de notas pessoais, aspectos particulares da exposição ou mesmo elementos de avaliação. Outro dos aspectos centrais deste trabalho é a possibilidade de gerir a partilha de informação, seja no que respeita aos materiais de suporte à aula, seja no que respeita à partilha de notas pessoais entre os diferentes participantes, professor e alunos.

## 1.1 Motivação

Uma das principais motivações para esta dissertação prende-se com a noção de que parte dos novos comportamentos de interacção social, decorrentes da utilização de redes sociais e mecanismos de comunicação *online* podem ter um papel importante na forma como se ensina e se aprende. A aposta passa pela junção da interacção social e multimédia[1], procurando assim fazer uso das duas, nas suas diferentes vertentes colaborativa e de mediação.

A interacção multimédia pode ser definida como: “*A measure of a media’s potential ability to let the user exert an influence on the content and/or form of the mediated communication.*” [1], e assume assim o papel de mediadora entre o utilizador e o objecto media alvo da sua utilização.

É neste aspecto que a dissertação se situa, trabalhando o aspecto prático da integração de dispositivos tácteis no trabalho cooperativo como uma base de suporte para a interacção social. A interacção social pode ser definida de diferentes formas, mas no contexto deste trabalho adoptamos a seguinte definição: “*The basic model that the sociological interaction concept stems from is thus the relationship between two or more people who, in a given situation, mutually adapt their behavior and actions to each other.*” [1].

Outra das motivações desta tese prende-se com a utilização do vídeo e a interactividade que permite gerar. Relativamente ao vídeo, é fácil perceber a sua grande aceitação por parte do público, considerando a capacidade de estimulação de dois sentidos fulcrais no ser humano e a contextualização simbólica que fazem com que a sua escolha seja unânime por quem usa actualmente os sistemas multimédia. Desde o aparecimento da TV as pessoas começaram a consumir muito mais informação audiovisual, num processo que seria continuado pela chegada da Internet. Esta veio trazer ao mundo uma forte mudança na utilização do vídeo. É difícil não encontrar um utilizador que não tenha já passado alguns minutos em sites como o YouTube (Figura 1.1) ou Metacafe (Figura 1.2).



**Figura 1.1** – YouTube usado como método de aprendizagem<sup>1</sup>.

**Figura 1.2** – Publicidade através de vídeo no Metacafe<sup>2</sup>.

<sup>1</sup> Fonte: <http://www.youtube.com/watch?v=0f6n1HMHI6>, Junho 2009

<sup>2</sup> Fonte: [http://www.metacafe.com/watch/yt-anhixJsOZJY/mc\\_vs\\_pc\\_vista\\_pr/](http://www.metacafe.com/watch/yt-anhixJsOZJY/mc_vs_pc_vista_pr/), Junho 2009

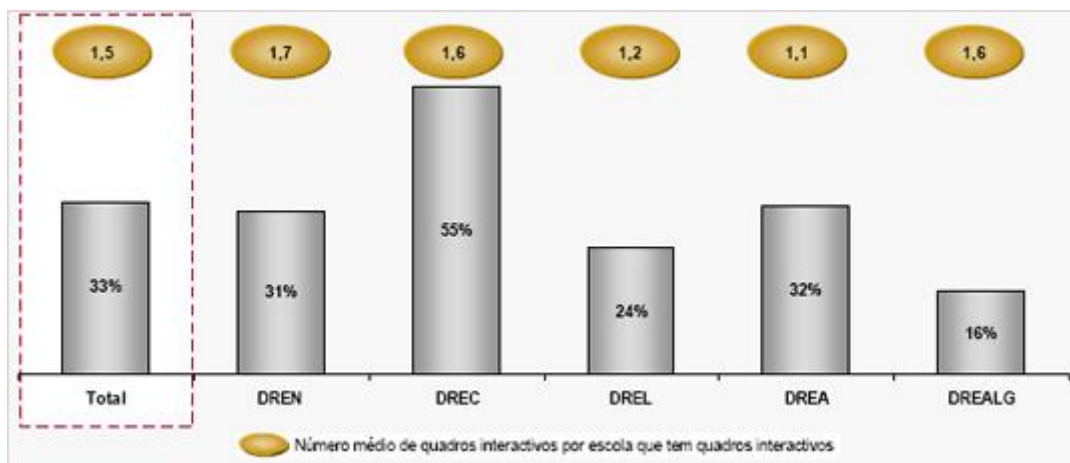
Não se trata apenas de puro entretenimento mas sim de uma fonte de publicidade, com possibilidades de rendimento significativas. As mensagens passam mais facilmente para o público, sem custos adicionais e a um ritmo que o próprio utilizador pode traçar. É com base nestas potencialidades que se propõe uma solução que possa combinar as vantagens do vídeo com as potencialidades das anotações.

As anotações permitem representar as várias interpretações possíveis acerca do mesmo conteúdo multimédia. Anotando, expressa-se uma opinião ou uma interpretação pessoal. Desta forma, cada participante pode tornar público, perante os conteúdos apresentados, diferentes opiniões ou diferentes pontos de vista.

Estas anotações caso sejam registadas permitem ao utilizador analisar posteriormente o sentido que deu ao conteúdo visualizado, bem como partilhar com outros utilizadores as informações registadas. Assim, chega-se a um contexto de trabalho colaborativo, que já fora mencionado anteriormente, em que as tecnologias multimédia passam a ser preponderantes a nível da partilha de informação num determinado grupo de utilizadores.

A solução proposta deve ainda permitir trabalhar vídeo através de diversos tipos de anotações, de forma a criar cenários de contextualização com a colaboração dos utilizadores. Esta não é caracterizada como um sistema centralizado que armazena todo o conteúdo multimédia, pois a sua composição vai no sentido da criação de redes locais, para que um grupo possa formar uma base de partilha instantânea. Todos estes procedimentos serão realizados com base numa arquitectura de coordenador/alunos em que além, dos pressupostos de colaboração, haja mecanismos de filtragem de conteúdos.

Finalmente uma pequena referência à motivação para a utilização de quadros interactivos. Os quadros interactivos [2] têm sido progressivamente introduzidos em ambientes de ensino, segundo um estudo publicado pelo GEPE (Figura 1.3) [3], que pretendia avaliar o impacto dos meios tecnológicos no ensino básico.



**Figura 1.3** - Percentagem de escolas que têm quadros interactivos e número médio de quadros interactivos por escola<sup>3</sup>.

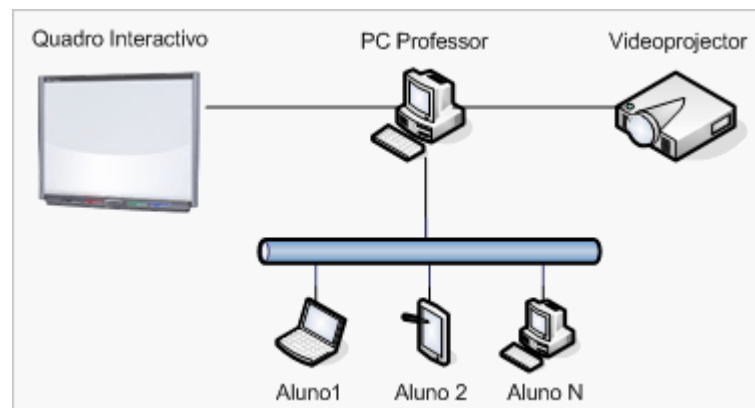
O factor de cobertura presente no estudo [3], onde existe em média um quadro interactivo por escola, é fundamental para que se pensem novas abordagens para o seu uso, tal como é proposto nesta dissertação. Este sistema permite a criação de anotações de vídeo, através de um quadro interactivo de uma forma em tudo idêntica à que é possível efectuar no computador pessoal, do aluno ou do professor.

Um dos pontos fundamentais da dissertação é garantir que ao adoptar tecnologias no ensino, não se perdem mecanismos tradicionais de colaboração e de trabalho individual.

## 1.2 Solução Proposta

A solução proposta aposta na maior simplicidade possível, que permita garantir os requisitos pretendidos. Desta forma os meios necessários à sua instalação serão mínimos o que permitirá a sua utilização num maior número de situações. A solução centra-se na arquitectura apresentada na Figura 1.4, em que o sistema funciona numa rede local e independente de qualquer outra infra-estrutura externa.

<sup>3</sup> Fonte: GEPE, dados preliminares; análise A. T. Kearney (Escolas públicas EB 2/3 e S)



**Figura 1.4** – Infra-estrutura física da solução.

Na infra-estrutura apresentada é essencial fazer uma divisão em duas grandes áreas, sendo a primeira reservada ao professor e aos seus dispositivos (videoprojector e quadro interactivo), e a segunda à forma como os alunos interagem entre si e com o docente.

O professor é um dos actores deste sistema e interage com o quadro interactivo para explicar anotações que já tenha feito anteriormente ou, caso seja necessário, criar novas anotações. Estas características vão de encontro à postura clássica de um professor numa aula, onde este pode fazer apontamentos no quadro ou trazer material já preparado para apresentar aos alunos. Para isso, o professor terá de usar exclusivamente o quadro interactivo, ou seja, o professor não terá outro dispositivo de interacção, tendo em consideração a utilização do videoprojector e do computador responsável pelo processamento.

Há ainda que ter em conta que o professor é responsável pela coordenação da partilha de anotações, ou seja, o grau de colaboração da aplicação determina que o docente terá de ser mediador das anotações partilhadas. Assim, poderá ser garantida uma filtragem da informação, bem como o controlo contra uso menos apropriado ao contexto da aula.

Os alunos terão um comportamento mais limitado a nível de gestão de anotações. Mesmo assim um aluno poderá criar as suas anotações e partilhar as que desejar com os colegas e com o professor. A visibilidade das anotações, no caso do aluno e do professor permite estabelecer diferentes tipos de trabalho individual e em grupo.

A nível de componentes, o aluno poderá utilizar o sistema com os mais diversos tipos de equipamento (Figura 1.4), sendo que inicialmente será desenvolvida a interface para computador pessoal e para TabletPC.



Estes são, em resumo os principais aspectos que serviram de motivação ao desenvolvimento deste trabalho. Na secção seguinte estão descritas algumas vertentes que pensamos constituem a principal contribuição deste trabalho.

### **1.3 Principais Contribuições Propostas**

O trabalho proposto nesta dissertação tem como objectivo integrar diversas funcionalidades, para que se consiga obter um software colaborativo fiável e de fácil usabilidade de partilha de anotações. O contexto educacional é assumidamente o ponto de partida e a base de referência de todo este trabalho, independentemente do facto da sua utilização poder ser generalizada a outros domínios.

Aspectos principais desta proposta:

- Desenvolvimento de um sistema que permita a partir de uma aula tradicional, rentabilizar a utilização de materiais audiovisuais e a utilização de mecanismos de anotação que promovam a interactividade na aula.
- Desenvolvimento de uma interface simples, que permita ao professor, criar, gerir e disponibilizar conteúdos audiovisuais anotados, para uma turma;
- Desenvolvimento de um mecanismo que permite que um aluno possa utilizar o material audiovisual gerado pelo professor, para criar as suas notas, durante a aula;
- Aproveitar o potencial dos quadros interactivos para criação e exposição de informação (vídeos e anotações), através do desenvolvimento de uma interface de utilização que de forma transparente permite ao professor trabalhar ou no quadro ou no seu computador pessoal;
- Desenvolver uma ferramenta, que pode ser utilizada pelos alunos como um mecanismo para apoio ao estudo;

Obviamente as contribuições não se esgotam na lista de possibilidades enunciadas e serão potenciadas pela criatividade de todos aqueles que tendo por base estes mecanismos adoptem estes aspectos como meros pontos de partida.

## **2 Trabalho Relacionado**

As novas tecnologias têm entrado nos ambientes de aprendizagem de uma forma natural e, por consequência, diversas empresas e entidades procuram desenvolver soluções que acrescentem agilidade de processos e novas opções. Com o constante desenvolvimento das tecnologias multimédia observado nesta área particular, as interfaces tornam-se cada vez mais robustas e ergonómicas, surgindo a necessidade de apresentar soluções com vista a tirar um maior partido da sua utilização em ambientes de ensino.

No que diz respeito ao trabalho existente nesta área, podem ser identificados vários grupos de desenvolvimento sendo de salientar os mais relevantes dentro do âmbito deste trabalho: anotações em vídeo, sistemas para apresentações, quadros interactivos e partilha de ficheiros. As secções seguintes apresentam exemplos de cada um dos grupos anteriormente identificados.

### **2.1 Anotações**

A anotação tem um propósito muito simples, que consiste de adicionar informação sobre um conteúdo audiovisual. As anotações que aqui se referem estão associadas a um contexto preferencialmente educativo, tendo a particularidade de se tratar da adição de informação baseada numa interpretação pessoal, acerca de um determinado tema que é comum a vários utilizadores. Dentro deste contexto, surge um grande leque de aplicações, tais como: Experimental Video Annotator [4], YouTube, VAnnotator [5] e VideoStore [6][7].

#### **2.1.1 Experimental Video Annotator (EVA)**

O Experimental Video Annotator[4] foi proposto em 1988, surgindo na altura em que cada vez mais o vídeo digital ia começando a ser utilizado. Este sistema possuía potencialidades de garantir o contexto e a qualidade da informação armazenada.

O objectivo desta aplicação era possibilitar a criação de anotações de momentos chave em vídeo, de uma forma mais intuitiva do que a que se observava recorrendo ao código temporal SMPTE a servir de *link* para vídeos sem contextualização. Com esta finalidade foi criado um mecanismo de anotações que combina os códigos temporais com informação simbólica.

Na década de oitenta os computadores pessoais eram pouco divulgados, pelo que o projecto se manteve com pouca divulgação pública. Esta aplicação era executada numa *workstation* DEC MicroVax ou IBM PC/RT desenvolvida pelo Visual Computing Group do MIT no projecto Athena[8]. Este projecto incluía também, para além desta capacidade de indexação de informação simbólica a partir de vídeo, uma linguagem cujo objectivo se centrava na criação de aplicações multimédia.

### **2.1.2 YouTube**

Iniciado em 2005, o projecto YouTube constituiu um importante passo a nível da partilha de conteúdos de vídeo. A facilidade com que este se tornou uma aplicação de enorme potencial é transversal a todas as classes/áreas sociais, incluindo política, publicidade e entretenimento, tornando possível a partilha de um vasto leque de vídeos. Qualquer tipo de vídeo pode ser enviado para o YouTube para ser visualizado e comentado por milhares de utilizadores. Constitui assim um ambiente próprio para a partilha de informação em que é oferecido aos utilizadores um meio onde estes podem comentar de forma colaborativa o conteúdo de um dado vídeo.

De entre as várias opções disponíveis no YouTube, as anotações assumem um papel de destaque, sendo possível adicionar/associar aos vídeos anotações sob a forma de texto (Figura 2.2) e/ou de *links* para outros vídeos (Figura 2.1).



**Figura 2.1** – Anotações do YouTube utilizadas como hiper ligações<sup>4</sup>.



**Figura 2.2** – Anotações de texto no YouTube<sup>5</sup>.

Esta inovação no Youtube veio demonstrar o enorme potencial das anotações, oferecendo aos utilizadores uma forma de aumentarem o conteúdo semântico dos vídeos. As anotações são feitas de uma forma muito simples, ao efectuar a transferência do vídeo o utilizador pode adicionar as anotações que pretender, através de um mini editor gráfico fornecido pelo próprio YouTube.

### 2.1.3 VAnnotator

O VAnnotator [5] é um programa de anotação de vídeo que foi desenvolvido no projecto denominado Vizard [9], cujo objectivo era fornecer aos utilizadores um novo paradigma para a navegação, anotação e edição de vídeo.

Esta aplicação é baseada numa implementação do codificador MPEG-7[10] que define como será guardado o vídeo e os metadados a ele associados. Utilizando esta especificação, o VAnnotator demonstra uma grande compatibilidade com outros tipos de aplicações que adoptem este tipo de codificador, dado que a informação gerada no programa é feita obedecendo às normas do MPEG-7.

O sistema de anotações é composto por um mecanismo de camadas denominadas *lenses*. O programa suporta a criação de vários tipos de anotações, onde cada tipo vai ficar anexado numa mesma camada. Ao longo da linha temporal do vídeo as camadas só têm representação caso tenha sido adicionada uma anotação no tempo corrente, caso contrário nada é mostrado.

<sup>4</sup> Fonte: <http://www.youtube.com/watch?v=uZChg1hKjpg>, Junho 2009

<sup>5</sup> Fonte: [http://www.youtube.com/watch?v=Modz5\\_ylk-w](http://www.youtube.com/watch?v=Modz5_ylk-w), Junho 2009

Assim, para o utilizador poder adicionar o seu próprio tipo de anotações terá de adicionar uma nova camada (*lens*) na linha temporal.

#### 2.1.4 VideoStore

Desenvolvido por Diogo Cabral como tema da sua dissertação de Mestrado, o VideoStore [7][6] teve como objectivo desenvolver um sistema de aprendizagem baseado na Internet, onde o utilizador pode aceder ao serviço através de um simples *browser* de Internet. A sua estrutura foi baseada em MS Producer 2003, e consegue obter a sincronização entre os vídeos, apresentações PowerPoint e as anotações.

A possibilidade de sincronização mencionada faz com que ao ser reproduzido um vídeo referente a uma aula, possam ser anexados ficheiros PowerPoint e conjuntos de anotações. Desta forma e à medida que o vídeo vai progredindo ao longo do tempo, as anotações e os slides também vão mudando. Como se pode constatar na Figura 2.3, do lado esquerdo está disponível o vídeo de referência e os capítulos que servem de guia principal ao mesmo, e do lado direito aparecem as anotações e o PowerPoint apresentado durante a aula gravada.

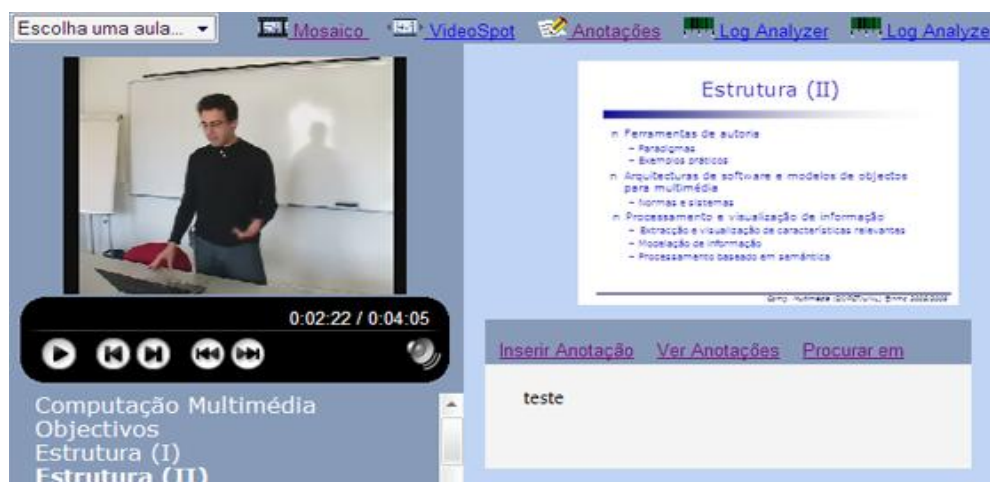


Figura 2.3 – Excerto da interface do VideoStore.

Esta aplicação também inclui a possibilidade de escolha, por parte do utilizador, da interface disponibilizada, nomeadamente através das opções de Mosaico, Log e VideoSpot. Os diversos tipos de interfaces permitem cobrir as necessidades de um universo alargado de utilizadores.

A opção Mosaico permite dar ao utilizador uma representação periódica da imagem (*frame*) registada no vídeo, e assim o utilizador tem uma perspectiva temporal dos

acontecimentos através de uma linha de tempo. Esta apresentação pode ser vista como uma representação discreta do vídeo, onde cada imagem representa o início de um determinado período.

No caso da opção Log, o objectivo passa pela criação de uma linha temporal que identifica os momentos mais vistos do vídeo. À medida que o vídeo vai sendo reproduzido, o servidor vai guardando os segundos reproduzidos por cada utilizador. O resultado final surge do somatório de visualizações individuais de todos os segundos do vídeo.

### **2.1.5 Sumário**

O suporte para anotações e partilha de vídeo tem tido um grande crescimento ao longo do tempo. Este factor está relacionado com os grandes desenvolvimentos tecnológicos, nomeadamente do hardware, cada vez mais potente e barato, que permite cada vez melhores desempenhos na reprodução e acesso a vídeo. Os programas possuem cada vez mais opções, muitas delas directamente provenientes dos constantes avanços tecnológicos.

É notória ainda a diversidade de abordagens dos mecanismos de anotações havendo muitas vezes a divergência entre a adopção de normas para a descrição do vídeo e das anotações MPEG-7[10] e a criação de uma especificação própria para estes dois tipos de informação. Na primeira opção o vídeo e os metadados estão descritos segundo uma norma, o que faz com que uma implementação segundo essa norma tenha compatibilidade com outras aplicações que também a implementem. Por outro lado esta norma é muito complexa, fazendo com que a utilização da mesma nem sempre compense a dificuldade da sua implementação.

## **2.2 Sistemas para Apresentações**

As anotações feitas sobre apresentações (PowerPoint e imagens) são o aspecto mais representativo destes sistemas, ou seja, a possibilidade de realizar as anotações numa base estática seguindo assim as interfaces habituais baseadas em papel e caneta. Para além disso, cada sistema analisado implementou novas ferramentas neste tipo de interface, consoante o público alvo a que se dirige.

## 2.2.1 Classroom Presenter

O Classroom Presenter [11] (Figura 2.4) é um sistema que pretende responder às necessidades do ensino colaborativo em rede. Criado pela Universidade de Washington em 1995 sob a forma de protótipo, suporta anotações partilhadas e publicação de conteúdo.



Figura 2.4 – Visão geral do Classroom Presenter.

Através de uma arquitectura cliente-servidor, o Classroom Presenter suporta uma estruturação hierárquica. Ao definir o professor como o servidor daquela rede e os alunos como clientes, estabelecem-se regras para os alunos, não permitindo o acesso a opções administrativas. Isto evita eventuais abusos e a possível utilização danosa do software em plena aula/apresentação. Para além deste aspecto, contém ainda o modo “Public Display”, onde um participante (nó) do sistema poderá mostrar o que está a ser realizado por ele.

Este programa permite anotações textuais, de imagem e de som. Este factor permite tornar a interacção natural, oferecendo uma facilidade de adaptação à aplicação por parte do utilizador.

O Classroom Presenter permite a extracção do conteúdo trabalhado e respectivas anotações, para publicação na Internet. Desta forma, torna-se mais simples partilhar a informação, dado que quem queira consultar a informação pode fazê-lo através de um explorador de páginas Web com capacidade de visualização de imagens.

Outra característica interessante deste software reside no facto de haver a possibilidade de ser efectuada uma apresentação sincronizada com os alunos, ou seja, à medida que o professor

vai fazendo anotações nos slides o aluno pode sincronizar-se com esse ritmo de trabalho. Esta funcionalidade é uma opção que o aluno pode alterar em qualquer parte da aula/apresentação.

Para finalizar há ainda que referir a possibilidade de importação de apresentações PowerPoint para um DECK (unidade de gravação do Classroom Presenter). Esta opção tem um elevado potencial, pois muitas das apresentações são elaboradas em PowerPoint e a sua utilização facilita a reutilização de materiais previamente produzidos noutros contextos.

### **2.2.2 ConferenceXP**

O ConferenceXP [12] foi igualmente desenvolvido pela universidade de Washington, e integra o Classroom Presenter descrito anteriormente. O Classroom aparece como um módulo do ConferenceXP na área das anotações de texto em imagem e apresentações PowerPoint.

Este sistema foi desenvolvido em parceria com a Microsoft, responsável pelo acesso e gestão de utilizadores através dos seus serviços. Os utilizadores podem ser organizados segundo temas, ou seja, estes podem ser distribuídos por diversas salas temáticas existentes de acordo com as suas preferências.

Neste aspecto, nota-se a opção de aumentar a possibilidade de escalabilidade da solução utilizando os serviços da Microsoft de modo a ter salas a nível global, não considerando a característica interessante de rede local contemplada no Classroom Presenter.

O ConferenceXP resolve os problemas de gestão da rede, deixando os pressupostos de edição e trabalho colaborativo para outros módulos desenvolvidos pela Microsoft e pela Universidade de Washington (Figura 2.5).



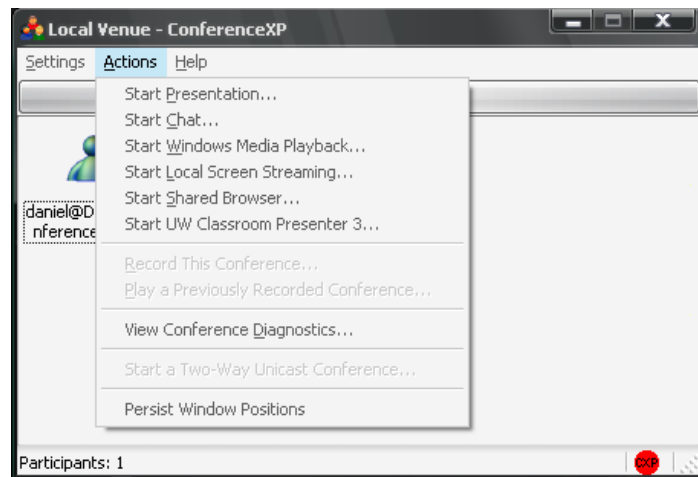


Figura 2.5 – Acções disponibilizadas no ConferenceXP.

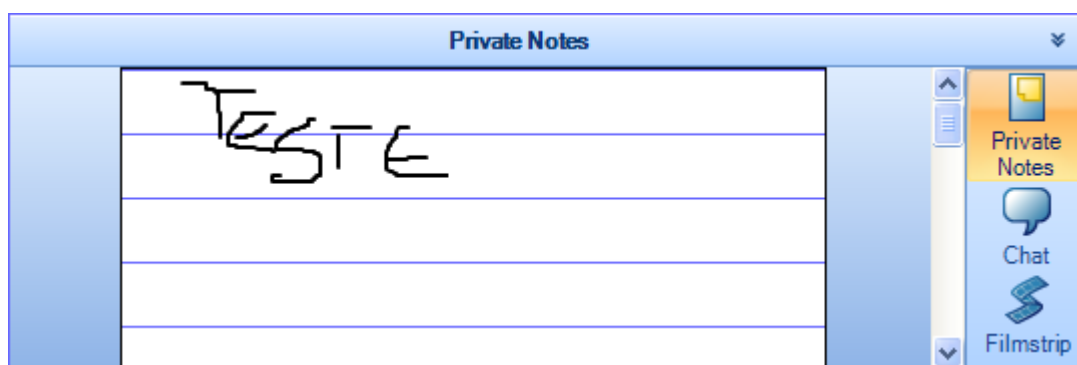
Como se pode constatar pela Figura 2.5, existem diversos tipos de acções, incluindo a possibilidade de utilizar o Classroom Presenter como forma de suporte ao trabalho colaborativo. Para finalizar, há que salientar a forte dependência do sistema operativo derivado desta parceria.

### 2.2.3 DyKnow

O DyKnow [13] é um sistema comercial que tem como objectivo possibilitar anotações de diversos tipos, num ficheiro PowerPoint ou num conjunto de imagens. Seguindo uma óptica de interface semelhante às aplicações do Microsoft Office 2007, esta aplicação é uma das melhores em termos de usabilidade e apresentação. Para facilitar a criação desta interface a DyKnow usa a “Framework.Net 3.5”, ferramenta esta que futuramente poderá trazer implicações significativas na portabilidade da aplicação para outros sistemas operativos.

Nesta aplicação, o factor de supervisão é essencial, pois o professor pode ter um controlo total sobre o computador do aluno. O professor pode consultar facilmente o que é que está a aparecer no ecrã do aluno num dado momento, bem como bloquear sites, ou até mesmo aplicações desnecessárias para a aula. Em casos mais extremos o professor pode ter acesso remoto ao computador do aluno.

Existe o conceito de anotações privadas (Figura 2.6) nas apresentações, ou seja, os alunos podem ter as suas próprias notas sem que estas tenham de ser feitas na área reservada aos slides.



**Figura 2.6** – Notas privadas no programa DyKnow.

O sistema de anotações privadas do DyKnow é muito simples, bastando escrever com um dispositivo de entrada que suporte movimento (rato ou caneta electrónica, entre outros). É necessário ter em conta que o rato não é o melhor dispositivo de anotação no DyKnow, isto porque a aplicação foi elaborada para TabletPC e a facilidade de escrita e a fluidez do traço desenhado pelo rato são inferiores quando comparados com os obtidos com uma caneta de um TabletPC.

#### **2.2.4 Sumário**

Ao analisar as aplicações descritas anteriormente constata-se que todos permitem opções semelhantes, nomeadamente criação de anotações (incluindo vídeo, imagem, texto), partilha e publicação. Deste facto, conclui-se que as aplicações chegaram a um ponto de optimização que só será ultrapassado com outro tipo de acções, tal como proposto pelo DyKnow quando entrou no mercado. A empresa que criou este programa criou cursos e esquemas de formação para docentes, criando assim uma cadeia de certificação com base nos seus produtos fomentando e divulgando desta forma o seu produto.

### **2.3 Quadros Interactivos**

Os quadros interactivos são um grupo de dispositivos tácteis, frequentemente utilizados em projectos de realidade aumentada (por exemplo, MIT Sketching [14] e Waterboard [15]). O recurso a este tipo de dispositivos tem tido um elevado grau de aceitação por parte da comunidade escolar, pois assemelham-se muito aos quadros já existentes.

### 2.3.1 NotateIt

O NotateIt [16] é um programa de anotações unicamente vocacionado para os apontamentos em quadros interactivos. Produzido pela empresa Blade Software, disponibiliza um elevado leque de opções ao utilizador, fornecendo ferramentas que melhoram a produtividade/interacção dos utilizadores. A interface (Figura 2.7) é caracterizada pela simplicidade, onde as operações mais importantes são disponibilizadas através de um simples toque. Foi ainda implementada no programa uma interface designada por Dynamic WhiteBoard, que disponibiliza uma vista minimalista, exclusivamente com as opções mais relevantes da aplicação.

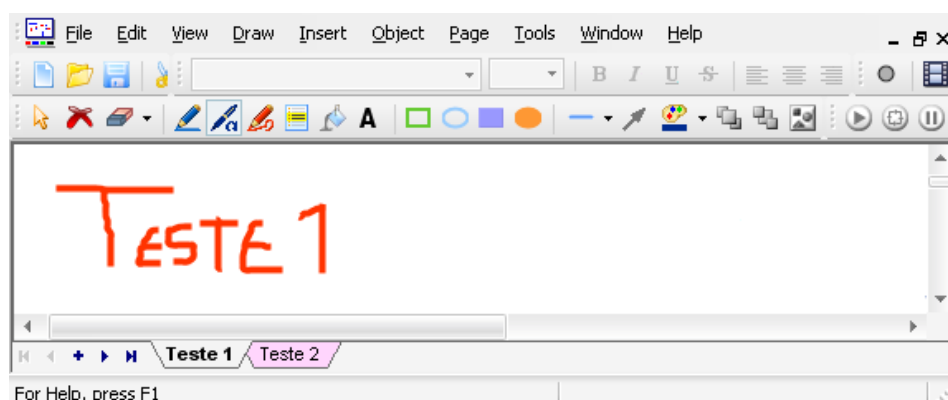


Figura 2.7 – Visão global do NotateIt.

Um aspecto interessante deste projecto é a forma simples como tratam as diversas páginas que existem numa sessão, como se pode constatar na Figura 2.7 onde está patente um mecanismo de separadores. Esta solução simples faz com que seja fácil a navegação entre os vários “quadros” escritos durante uma aula, permitindo a minimização do problema que existia quando os professores apagavam a matéria que estava no quadro e o aluno ainda se encontrava a tirar notas.

O recurso a serviços do Windows permite adicionar novas opções ao programa, entre elas a partilha de ecrã e das anotações no ambiente de trabalho do Windows. Esta segunda opção revela-se muito interessante, pois permite que sejam feitas anotações no quadro em programas que estejam a ser executados no sistema operativo. No entanto, esta opção revela algumas limitações pois é uma simples cópia do ecrã, não sendo possível aproveitá-la de forma dinâmica.

### 2.3.2 Mimio Studio

Esta aplicação [17] segue a aproximação tradicional dos programas para quadros interactivos, suportando vários tipos de ferramentas úteis para quem trabalha com estes dispositivos. Foi desenvolvido pela empresa Mimio formada no MIT e reconhecida pelos produtos que desenvolve na área da realidade aumentada e dos dispositivos tácteis.

Em relação à aplicação foram tidos em conta princípios muito interessantes e igualmente relevantes nas aplicações para quadros interactivos, como por exemplo a separação das ferramentas da aplicação, como se pode constatar na Figura 2.8 e Figura 2.9.

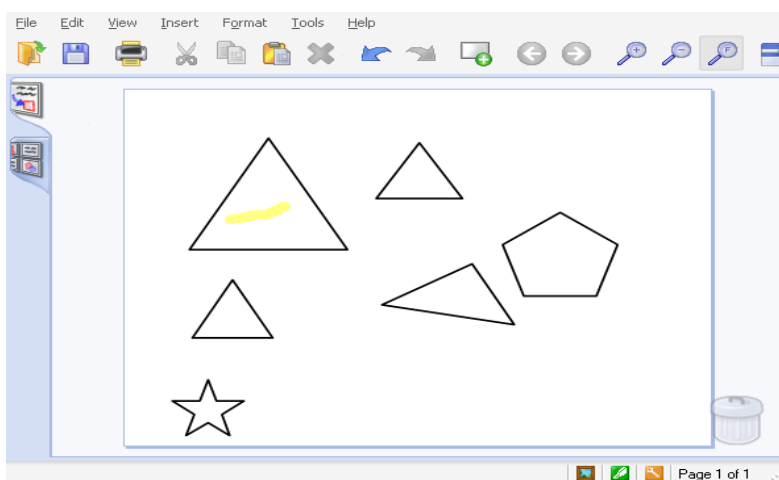


Figura 2.8 – Visão global do Mimio Studio.



Figura 2.9 – Mimio Studio Toolbox.

Quando comparado com o NotateIt da secção 2.3.1, nota-se que o Mimio Studio apresenta opções simplificadas e ícones de grande dimensão. Isto vai de encontro às necessidades para quando se trabalha com dispositivos tácteis, em particular um quadro interactivo. Para além disso, o facto de separar a caixa de ferramentas (*toolbox*) da restante interface, faz com que a utilização da mesma seja preponderante no modo *fullscreen*. Esta opção torna-se muito útil quando se está a trabalhar com múltiplas resoluções, pois possibilita a movimentação da caixa de ferramentas consoante a necessidade específica do utilizador.

### 2.3.3 Sumário

No âmbito dos programas disponíveis para quadros interactivos existem algumas alternativas, todas elas com semelhanças, em que procuram dar um grande número de opções aos utilizadores. Isto surge um pouco do facto de os quadros interactivos terem mecanismos de inserção de dados semelhante aos de um TabletPC, permitindo assim uma generalização

dos programas para os dois tipos de dispositivos. No entanto convém realçar que a utilidade dos quadros interactivos vai um pouco além dos programas por estes facultados, ao oferecerem um suporte digital que permite a flexibilização das interacções.

## **2.4 Partilha de Ficheiros**

A partilha de ficheiros, usualmente caracterizada na aplicação como partilha de conteúdos, é um ponto fulcral da dissertação, pois será através dela que os intervenientes trocaram informação entre si. Tipicamente este factor de partilha poderia ser implementado através de diversas tecnologias, mas nesta dissertação foram considerados dois mecanismos de partilha de informação: Servidores HTTP [18]; e as redes *Peer To Peer* [18].

### **2.4.1 Servidores HTTP**

Os servidores HTTP são um mecanismo padrão de partilha de informação. São caracterizados pela sua componente de centralização de dados, onde um universo de utilizadores pode descarregar sempre que pretenda a informação disponibilizada. Neste âmbito foram seleccionados três servidores HTTP, de forma a serem analisadas as principais características de cada um deles.

O Apache HTTP Server [19] é o servidor HTTP mais utilizado a nível mundial. Este servidor criado pela *Apache Software Foundation (ASF)* é uma das principais referências mundiais no segmento dos servidores HTTP. É caracterizado pela sua grande extensibilidade e transversalidade ao nível dos sistemas operativos sobre os quais opera, sendo que é maioritariamente utilizado sobre sistemas Unix. O Apache é desenvolvido sobre a forma de código aberto e é mantido por uma grande comunidade.

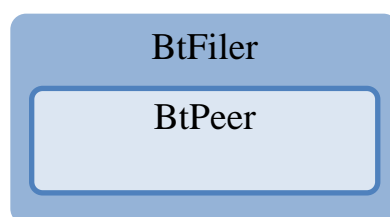
Um outro exemplo de um servidor HTTP é o Small HTTP Server [20], cuja composição não é tão extensível quanto o anterior, ou seja, é menos expansível ao nível das configurações e opções que disponibiliza ao utilizador, e tem um tipo de características mais minimalista. Apesar deste aspecto ainda possui algumas opções interessantes, nomeadamente a implementação de diversos tipos de servidores para além do HTTP, como por exemplo: Mail, FTP, DNS, entre outros. Outro pormenor que demarca este servidor em relação ao anterior é o facto de este ser pago e não é aberto, o que restringe de alguma forma o público-alvo deste tipo de aplicação

Finalmente uma referência para os servidores HTTP que existem enquanto objectos em linguagens de programação, cuja utilização é bastante frequente em aplicações que pretendem ter um servidor HTTP a cargo da mesma. Algumas das linguagens de alto nível já possuem objectos deste tipo, dentro deste contexto temos a linguagem Java, que através de um objecto denominado *HttpServer* existente na sua *Framework* disponibiliza um servidor. Este objecto permite criar um servidor HTTP com algum grau de configuração ao nível da implementação, isto é, o programador poderá definir configurações que, por exemplo, levam à alteração do contexto das respostas do protocolo HTTP.

#### 2.4.2 Redes Peer To Peer

As redes *Peer To Peer*, são outra alternativa à partilha de conteúdos entre utilizadores, com a diferença de não serem baseadas no modelo cliente-servidor descrito anteriormente. As redes *Peer To Peer*, usualmente denominadas por comunicação entre pares, são um tipo de arquitectura onde cada interveniente (nó) tem um comportamento igual aos restantes. Isto é, todos os intervenientes actuam da mesma forma desempenhando em simultâneo o papel de cliente e servidor.

Dentro deste domínio foram analisadas duas aplicações. A primeira é a BtFiler [21] criada em Python que implementa uma *framework* desenvolvida no Berry College, cujo objectivo é suportar a partilha de ficheiros com base numa camada BtPeer que é uma generalização de uma implementação de uma rede Peer To Peer (Figura 2.10). No fundo o BtFiler é uma extensão da Framework BtPeer e é desenvolvida em duas linguagens: Java e Python.



**Figura 2.10** – Interação entre os elementos do BtFiler.

A primeira camada é composta pela classe BtPeer (Figura 2.10), que tem por objectivo tratar dos pontos de baixo nível da rede. Visa garantir a integridade da estrutura de rede, consoante a entrada e saída de utilizadores e fornecer ao utilizador um conjunto de métodos para tratar das listas de nós conhecidos. Esta camada é estendida pela camada BtFiler, que suporta um grau de granularidade mais elevado. Neste nível a gestão de mensagens e

conteúdos, opera com o elemento ficheiro. Isto é, o utilizador dispõe de um leque de mensagens de alto nível (quando comparadas com o BtPeer), que inclui a transferência e requisição de ficheiros a outros utilizadores.

Como segundo exemplo temos uma aplicação denominada TinyP2P [22] criada por Edward W. Felten e Alex Halderman na universidade de Princeton, cuja finalidade passou pelo desenvolvimento de uma aplicação final que permitisse a partilha de ficheiros, ao contrário da anteriormente exemplificada que se centra numa Framework de desenvolvimento. Este programa usa no seu núcleo, para partilha de informação, os protocolos HTTP e XML-RPC e onde cada cliente tem um servidor HTTP para partilha de ficheiros.

### **2.4.3 Resumo**

A partilha de ficheiros é um ponto importante na realização do trabalho apresentado nesta dissertação, nomeadamente no modo de funcionamento colaborativo da aplicação. Essencialmente a escolha de uma das arquitecturas é feita mediante as nuances necessárias na lógica do sistema, sendo a principal, o facto de se pretender uma partilha de ficheiros distribuída ou centralizada. No que diz respeito às redes Peer To Peer, conclui-se facilmente que as mesmas seriam uma mais-valia num sistema de larga escala, sendo que a sua utilização teria de ser acompanhada por uma infra-estrutura paralela que garantisse a comunicação cliente-servidor, para estabelecer uma relação hierárquica entre os intervenientes.

Por outro lado a implementação via servidor HTTP é viável de todas as formas enunciadas (Apache HTTP Server, SmallHTTP Server e objectos de uma linguagem), no entanto o objectivo de reduzir os recursos necessários para instalação e operação do sistema, apontam para que não seja necessária a utilização de um HTTP Server como o Apache ou equivalente. No fundo para se desenvolver uma solução à medida, será sempre conveniente ter um objecto na *Framework* da linguagem de programação escolhida, para haver um maior controlo e configuração dos diferentes módulos da aplicação.

### **3 Descrição do Sistema**

Neste capítulo é feita uma descrição do sistema proposto e da forma como se pressupõe a sua utilização num contexto educativo. A primeira secção descreve de uma forma genérica o modelo de utilização do sistema, ou seja, a forma como se prevê a sua utilização no contexto de uma sala de aula. A secção seguinte aborda o conceito de anotação no contexto deste trabalho. As restantes secções descrevem com maior detalhe alguns aspectos do sistema como a interface de utilização, a arquitectura de suporte e alguns aspectos de implementação mais relevantes.

#### **3.1 Modelo de Utilização**

Ao propor um modelo de utilização para um sistema com estas características é fundamental ter por base uma metáfora e um contexto de utilização que nos sirvam de referência e a partir da qual seja possível desenhar novas forma de utilização, assim como projectar novos meios de interacção entre os utilizadores.

Assim, o ponto de partida para a discussão do modelo de utilização teve por base o conceito de uma aula presencial tradicional onde o professor e os alunos partilham o mesmo espaço, sendo que a tecnologia adoptada permite logo à partida retirar algumas dessas premissas sobre a necessidade de partilha de um mesmo espaço físico.

O desafio que se coloca em termos do modelo de utilização centra-se na forma como, na sala de aula, se pode tirar partido da utilização de vídeo enquanto ferramenta interactiva, ou seja, utilizar o vídeo como um mecanismo colaborativo e de suporte à interacção entre o professor e os alunos.

Esta ideia tem por base a utilização de anotação de vídeo, isto é, a possibilidade de o professor e os alunos poderem exprimir sobre o vídeo as suas opiniões, dúvidas ou pontos de vista através de notas visíveis sobre o vídeo, de uma forma interactiva e em tempo real. No entanto, o problema não se esgota na simples possibilidade técnica de inserir sobre o vídeo,



anotações textuais ou gráficas. Torna-se necessário encontrar uma forma de enquadrar estas possibilidades num modelo de utilização em que a operação sobre os dispositivos não se sobreponha à dinâmica natural de uma aula.

Foi por esta razão que um cuidado particular foi colocado no desenvolvimento das interfaces. Estas foram desenhadas segundo uma lógica minimalista, com o objectivo de permitir ao utilizador focar-se no que pretende transmitir, permitindo-lhe, de uma forma rápida a execução de anotações e a consulta de notas criadas por outros. Pretende-se no fundo criar mecanismos que permitam tornar as notas em vídeo, numa forma natural de trabalhar, com o que poderíamos designar como o fluxo audiovisual resultante da aula.

Globalmente, o modelo de utilização proposto assume que a totalidade da aula, ou parte dela, está suportada na projecção de materiais audiovisuais (vídeo). Note-se que a filmagem em directo da aula, com projecção, é nos dias de hoje, uma solução viável e simples para um suporte audiovisual integral de uma aula.

O material audiovisual, vídeo em directo ou vídeo educativo pré-gravado, é projectado e utilizado como um elemento complementar à exposição do professor. Sobre o vídeo projectado, o professor pode efectuar notas ou solicitar aos alunos que partilhem as suas opiniões e pontos de vista, através de notas (sobre o vídeo). Este modelo pode ser naturalmente estendido para um contexto exterior à sala de aula onde estes mesmos mecanismos servem de suporte ao estudo do aluno ou à realização de trabalhos escolares, neste caso como ferramenta de trabalho individual.

### **3.1.1 Modos de funcionamento**

O modelo de utilização anteriormente referido sugere que a mesma aplicação pode, por parte do aluno e do professor, ser utilizada como instrumento individual no estudo e preparação das aulas e como ferramenta colaborativa, na sala de aula. A aplicação desenvolvida tem por base dois tipos de funcionamento: Autónomo e Colaborativo. Estes dois modos de funcionamento enquadram diferentes métodos de trabalho e pretendem proporcionar uma melhor e mais completa utilização do sistema, em que o utilizador pode utilizar a aplicação de uma forma transparente consoante o contexto em que está inserido. Desta forma o utilizador tem disponível de uma forma integrada, a sua ferramenta de estudo autónomo e a sua aplicação colaborativa.

## **Modo Autónomo**

O funcionamento em modo autónomo é no fundo a utilização da aplicação por um único utilizador, em que este faz as operações típicas no que diz respeito à anotação de vídeo. A principal motivação para a utilização do sistema neste modo reside na possibilidade de criar anotações, dando assim uma caracterização particular do utilizador face a um vídeo previamente definido. Permite que sejam criadas anotações e que as mesmas sejam guardadas para posterior utilização num ambiente colaborativo, ou seja, permite o trabalho que normalmente seria caracterizado como a típica abordagem da preparação de aulas, (do professor) ou realização de trabalhos de casa (do aluno).

Neste modo está igualmente disponível um mecanismo de pesquisa de anotações. Esta funcionalidade facilita a navegação entre anotações e permite uma fácil utilização quando o número de anotações geradas pelo próprio, ou provenientes da partilha com outros utilizadores, tornar inviável uma procura manual.

O modo autónomo recorre aos materiais tornados acessíveis durante o trabalho colaborativo e permite preparar materiais (vídeos ou anotações) para partilha numa utilização colaborativa.

## **Modo Colaborativo**

A abordagem implementada no modo colaborativo materializa a extensão à aula tradicional referida anteriormente. Neste contexto o coordenador (professor) poderá adoptar o vídeo como um elemento complementar que acompanha toda a aula ou que é utilizado pontualmente. O vídeo poderá estar presente em permanência se o professor produz imagens em directo a partir de uma filmagem do quadro, ou se possui conteúdos audiovisuais complementares sobre cada um dos temas expostos.

Ao utilizar e disponibilizar esta informação em vídeo, pode criar anotações ou recorrer a anotações que tenha previamente preparado. De igual forma, pode desenvolver trabalhos individuais ou em grupo em que os alunos são motivados a apresentar as suas conclusões sobre a forma de anotações de vídeo.

Operacionalmente, o professor cria a aula e com a aula criada os alunos podem "entrar" na aula ligando-se ao professor e passando a aceder aos conteúdos, por ele partilhados. Como já

foi referido, conteúdos são, neste contexto, vídeos e anotações criadas como suporte à exposição da matéria.

Um aspecto fundamental deste modo colaborativo é a partilha de informação no decorrer da aula, isto é, a capacidade de haver troca de informação entre os diferentes intervenientes no sistema. Esta característica torna possível a partilha de anotações entre os alunos e o professor, vista como avaliação ou troca de ideias, e de igual forma a partilha de anotações entre os alunos. Este último aspecto reflecte o que seguindo a metáfora da aula tradicional seria "tirar fotocópias" do caderno de um colega.

Para além deste mecanismo mais tradicional, pode ser interessante abordar esses apontamentos (anotações) como materiais didácticos que podem ser utilizados ou partilhados como material complementar de suporte à aula, aumentando assim o enriquecimento colectivo da turma.

Os mecanismos e abordagens referidos anteriormente podem também ser vistos para além da sua característica de representação simbólica de um dado momento/acção, nomeadamente através da sua vertente ao nível da avaliação. Isto surge do facto da interactividade poder ser encarada com um mecanismo pergunta resposta, ou seja, se o professor pode apresentar conteúdos e os alunos podem partilhar anotações (respostas, interpretações), poderá ser desencadeado um mecanismo de avaliação que tem por base um conjunto de anotações feitas pelo aluno no decorrer da aula.

Estes mecanismos (anotações) não podem ser vistos como um mecanismo de avaliação analítico processado pelo computador, mas podem perfeitamente servir como material (escrito) de apoio que possui informação essencial no processo de avaliação.

### **3.2 Anotações**

O ponto base desta dissertação assenta na visão normal de um anotador, onde a junção da anotação com o vídeo leva a uma interpretação particular consoante o utilizador. É nesse sentido que vai o conceito de anotação no contexto desta dissertação, ou seja, através das anotações criar um mecanismo que possa ajudar os utilizadores a classificar ou contextualizar um determinado momento num vídeo.

Tecnicamente a anotação pode ser um qualquer objecto, figura ou imagem, onde o único requisito base é o complemento à acção que decorre no vídeo. Assim, na aplicação desenvolvida, criou-se uma anotação de forma abstracta para que a mesma possa ser facilmente estendida no futuro. Isto quer dizer que para além dos tipos de anotações já incluídos, entre eles, Rectângulo, Texto, Elipse e Linha, pode-se criar novos tipos sem comprometer a integridade dos já existentes.

### 3.2.1 Tipos de Anotações

Como já foi referido existem quatro tipos de anotações, Rectângulo, Texto, Elipse e Linha, sendo que a criação destes tipos tem o objectivo de demonstrar a versatilidade dos diferentes tipos de anotações que podem ser desenvolvidos no contexto da aplicação.

A definição de uma forma mais generalista do que é uma anotação permitiu gerar uma classificação de atributos globais, isto é, existe um conjunto de atributos base que são necessários em qualquer tipo de anotação desenvolvido (Tabela 3.1).

<b>Atributo</b>	<b>Descrição</b>
<b>Título</b>	Título da anotação.
<b>Autor</b>	Autor da anotação.
<b>Data de Criação</b>	Data em que a anotação foi criada.
<b>Última Modificação</b>	Data em que o autor da anotação fez a última alteração.
<b>Visibilidade</b>	Estado de visibilidade da anotação.
<b>Opacidade</b>	Opacidade da anotação, cujo valor varia entre 0 e 1.
<b>Endereço WEB</b>	Endereço WEB para o qual a anotação redirecciona quando é dado um clique com a tecla esquerda do rato sobre a mesma.
<b>Pontos de Referencia</b>	Lista de pontos absolutos que as anotações podem assumir durante o seu período de visualização.

**Tabela 3.1** - Atributos globais da anotação.

Para além destes atributos de carácter global cada tipo de anotação apresenta um conjunto de atributos específicos associados à sua natureza, em termos gráficos ou visuais. Nas secções seguintes são descritos os principais atributos referentes a cada um dos tipos de anotação definidos.

## Rectângulo

A anotação do tipo Rectângulo tem como base a representação da figura geométrica com o mesmo nome. Como todas as restantes anotações, implementa um tipo de atributos particulares, que visa descrever a sua aparência, como consta na Tabela 3.2. Este tipo de anotação é utilizado em diversos casos, por exemplo, como delimitador de uma área, através da utilização da cor de fundo transparente e um rebordo superior a zero unidades.

Atributo	Descrição
<b>Largura</b>	Largura do rectângulo.
<b>Altura</b>	Altura do rectângulo
<b>Cor de Fundo</b>	Cor de Fundo do rectângulo.
<b>Rotação</b>	Rotação do rectângulo
<b>Espessura do Rebordo</b>	Espessura do rebordo do rectângulo.
<b>Cor do Rebordo</b>	Cor do rebordo do rectângulo, só aplicada quando a espessura do mesmo for diferente de zero.

Tabela 3.2 - Atributos do tipo de anotação rectângulo.

## Texto

A anotação do tipo Texto representa na integra um texto que se pretenda associar a um vídeo. Para implementar este tipo de anotação foram definidos tipos de atributos particulares, por forma a descrever visualmente o texto da anotação (Tabela 3.3).

Atributo	Descrição
<b>Cor do Texto</b>	Cor do texto que é apresentado na anotação.
<b>Cor de Fundo</b>	Cor de fundo do texto, tipicamente utilizado para aumentar o contraste entre o texto e o vídeo.
<b>Rotação</b>	Rotação do texto.
<b>Texto</b>	Espessura do rebordo do rectângulo.
<b>Tamanho da Fonte</b>	Tamanho do texto que está a ser apresentado.

Tabela 3.3 - Atributos do tipo de anotação textual.

## Elipse

Em relação à Elipse foram considerados factores semelhantes aos já referidos para o Rectângulo e que permitem descrever uma figura geométrica (Tabela 3.4). Inicialmente este tipo de anotação tinha sido idealizada para representar círculos simples, mas ao ser feita uma análise custo benefício desse tipo de abordagem concluiu-se que ao ser definido um círculo implicaria uma grande lacuna ao nível da abrangência e utilização. Este tipo de anotação encontra-se nos metadados que descrevem as anotações, na secção 3.2.2.

Atributo	Descrição
<b>Altura</b>	Altura da elipse.
<b>Largura</b>	Largura da elipse.
<b>Cor</b>	Cor de fundo da elipse.
<b>Espessura do Rebordo</b>	Espessura do rebordo do rectângulo.
<b>Cor do Rebordo</b>	Cor do rebordo da elipse, só aplicada quando a espessura da mesma for diferente de zero.

**Tabela 3.4** - Atributos do tipo de anotação elipse.

## Linha

A Linha é outro tipo de anotação geométrica descrita nos metadados (Tabela 3.5). O objectivo deste tipo de anotação é conseguir caracterizar, por exemplo, movimentos. Este é um tipo de anotação que se integra bem com as interfaces tangíveis, onde a utilização de uma caneta tira preferencialmente partido da utilização de linhas.

Atributo	Descrição
<b>Pontos</b>	Conjunto de pontos que compõe a linha desenhada.
<b>Cor</b>	Cor da linha.
<b>Grossura</b>	Grossura da linha.

**Tabela 3.5** – Atributos do tipo de anotação Linha.

### 3.2.2 Representação

O conceito de representação pretende formalizar a descrição das anotações ao nível dos metadados, isto é, o estabelecimento da linguagem que descreve as anotações de forma a criar um formato intuitivo e de fácil exportação.

Inicialmente foram tidas em conta duas formas de implementar a representação das anotações, uma tendo em conta a utilização de um codificador MPEG21, cuja norma suporta a descrição das anotações, e a segunda passando pela criação de um formato de descrição de anotações próprio. A solução MPEG21 não foi escolhida para esta dissertação, pois a sua adopção iria obrigar a uma implementação desta norma complexa por parte das aplicações que pretendessem ler ou compatibilizar os seus dados com o formato deste sistema. Por outro lado, a utilização de um formato próprio, baseado em XML, demonstrou ser a opção mais simples e versátil para esta tarefa, pois é uma linguagem de marcação frequentemente utilizada para descrever dados, e que ao mesmo tempo se torna um bom mecanismo de exportação do formato desenvolvido.

A premissa anteriormente mencionada não é vinculativa, ou seja, a primeira abordagem passou pela criação de uma descrição dos dados, delegando para trabalho futuro, os aspectos mais profundos da adopção deste formato como padrão em outras aplicações (portabilidade). A utilização do XML também oferece outro tipo de vantagens, nomeadamente ao nível da extensibilidade, que promove a versatilidade do formato criado. Este factor faz com que seja mais fácil implementar alterações ao nível da descrição das anotações, o que em termos de trabalho futuro significa uma maior facilidade e rapidez na actualização do formato existente.

A adopção de um formato pressupõe criação de mecanismos de validação do mesmo, não só por uma questão informativa e de compromisso, mas também para se poder filtrar a informação que é passada para a aplicação. No essencial o protótipo tem de ter capacidade de determinar se um ficheiro de anotações é válido, e o formato para ser aceite tem de estar bem descrito. Para isso é utilizado XML Schema (**Anexos 6.1, 6.2 e 6.3**), que fornece um mecanismo de validação ao XML e por consequência aos metadados definidos na aplicação.

Desta maneira descreve-se o procedimento essencial da representação das anotações, através de um XML Schema (**Anexos 6.1, 6.2 e 6.3**) que descreve todo o tipo de informação, necessário no ficheiro de anotações.

### **3.2.3 Sincronização**

A facilidade com que se processa a sincronização dos dados partilhados entre os diferentes utilizadores e os vídeos que lhes servem de suporte constitui um dos grandes desafios que se coloca a este sistema.

Na sua essência a sincronização é o mecanismo base que garante a junção das anotações com o vídeo, criando assim o conceito de anotador de vídeo. Cada anotação tem associada uma caracterização temporal relativa ao período em que deverá estar visível no vídeo. Este ponto parte do princípio que mediante a progressão temporal do vídeo, são mostradas no ecrã as anotações cujo tempo actual do vídeo se encontre dentro do intervalo de tempo em que as mesmas são definidas como visíveis.

No entanto o problema da sincronização é mais complexo devido à possibilidade de partilhar anotações entre utilizadores e da necessidade de garantir que a actualização de uma anotação deve ser repercutida pelos diversos utilizadores com quem a foi partilhada. De notar que a alteração do conteúdo de uma anotação não se limita ao seu conteúdo textual ou gráfico e pode significar uma alteração da sua janela temporal de visualização.

O mecanismo utilizado na gestão e sincronização temporal das anotações é descrito de uma forma mais detalhada na secção 3.5.2.

Finalmente, há que referir que a problemática da sincronização das anotações deve ser completamente transparente para o utilizador final, que cria e edita as suas anotações.

### **3.2.4 Anotações Dinâmicas**

No contexto deste trabalho, a designação de anotação dinâmica pretende caracterizar apenas a sua localização ao longo de um vídeo. Embora, na essência todas as anotações sejam dinâmicas no que respeita à forma do seu conteúdo, localização e atributos temporais. No âmbito deste sistema, uma anotação dinâmica é uma anotação cuja localização se altera ao longo do vídeo, ou na perspectiva do utilizador, uma anotação que ao longo do tempo se desloca sobre o vídeo. A utilização de anotações dinâmicas está normalmente associada a contextos muito específicos, em que a anotação é relativa a um elemento em movimento.

A solução encontrada pretende que o utilizador possa se expressar facilmente através da interface a intenção de ver a sua nota acompanhar um determinado objecto no vídeo. O que é



efectuado através da associação de um ponto no vídeo a uma anotação dinâmica, sendo assim criada uma anotação que irá automaticamente seguir o objecto representado nesse ponto de acordo com o seu movimento no vídeo.

Esta simplicidade de operação esconde uma complexidade que torna o processamento deste tipo de anotações completamente distinto do das restantes anotações. Uma anotação dinâmica obriga a um processamento adicional do vídeo, para que seja possível identificar o padrão que representa o objecto identificado como a "âncora" da anotação.

Ao nível da representação, uma anotação dinâmica recorre a pontos de referência, um parâmetro mencionado anteriormente quando da descrição dos atributos globais das anotações (Tabela 3.1). Estes pontos visam descrever as posições que a anotação assume no intervalo de tempo em que é desenhada. Uma anotação estática está sempre na mesma posição durante todo o tempo no qual aparece, ou seja, só são definidos os tempos de início e de fim do período de mostra da anotação. Enquanto que nas anotações ditas dinâmicas, existem  $N$  pontos de referência, onde  $N$  representa o número de diferentes posições que a anotação assume no intervalo de tempo em que é desenhada.

Resumindo, o termo anotação dinâmica pressupõe que, como já foi referido, a anotação assume várias posições ou localizações na imagem do vídeo, podendo assim ser utilizada para seguir objectos ao longo do vídeo.

### **3.3 Descrição da Interface**

A interface da aplicação foi desenhada com o objectivo de facilitar a interacção do utilizador, em particular tendo em vista a utilização de interfaces tácteis, como por exemplo quadros interactivos. O intuito global passa pelas operações de toque único onde é necessário que grande parte das operações sejam feitas de forma rápida e intuitiva.

A interface centra-se numa janela principal (Figura 3.1) que se encontra organizada em diversas áreas. Estas áreas têm como objectivo agrupar tematicamente as opções de modo a tornar a interacção do utilizador o mais fácil possível. Para além destas áreas identificadas, também existem mini janelas, cujo objectivo se centra na parametrização de eventuais atributos necessário à tarefa que o utilizador está a desempenhar.

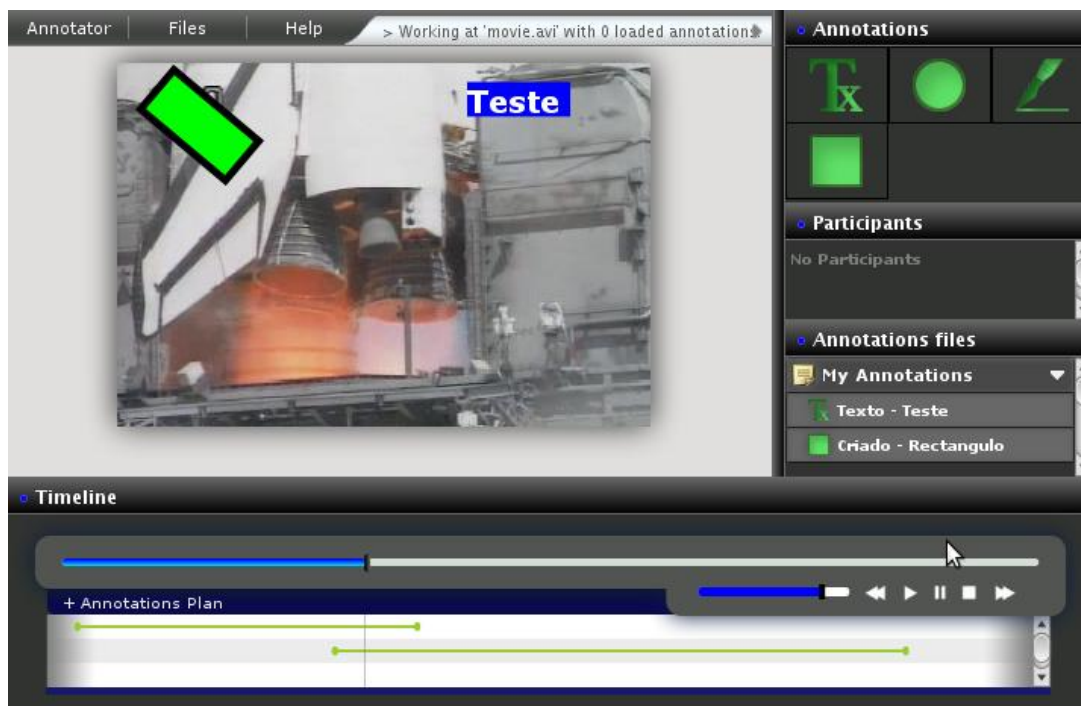


Figura 3.1 - Visão global da janela principal.

### 3.3.1 Áreas de Opções

Na janela principal da aplicação foram identificadas as seguintes áreas: Barra de Menu, Tipos de Anotações, Listagem de Utilizadores, Anotações Carregadas, Planificação das Anotações e Área de Criação e Visionamento de Conteúdos. Estas áreas têm diferentes pressupostos inerentes ao contexto da aplicação, pois através delas o utilizador pode executar tarefas tão diversas como criar anotações partilhar e visionar conteúdos.

#### Barra de Menu

A Barra de Menu oferece ao utilizador um conjunto diversificado de operações e informações. Dentro das opções (Figura 3.2) existem três menus, Annotator, Files e Help.

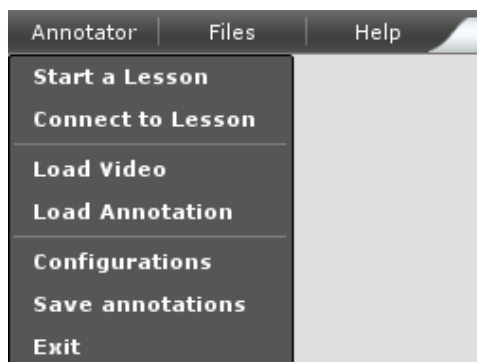


Figura 3.2 - Barra de Menu com o menu *Annotator* seleccionado.

O menu *Annotator* é onde se encontram as operações de configuração e finalização da aplicação. Através deste menu podem ser criadas novas aulas ou entrar numa já existente, possibilitando assim a inicialização do modo colaborativo. Cada uma destas opções pressupõe uma diferente interação por parte do utilizador, no caso da inicialização da aula será mostrado o formulário descrito, mais à frente, na secção 3.3.2, enquanto que no caso de um aluno que pretenda simplesmente ligar-se a uma aula, será apresentada uma janela onde lhe é pedida a informação referente ao endereço IP do professor que fora facultado pelo professor durante a aula.

No menu *Annotator* o utilizador também poderá guardar as anotações que se encontram carregadas. Esta opção tem um cariz importante na forma como as anotações ficam organizadas ao nível do sistema de ficheiros local. Ou seja, o utilizador ao carregar nesta opção irá guardar todas as anotações do vídeo que está a utilizar correntemente, armazenando assim, num só ficheiro, não só as suas anotações como as anotações partilhadas por outros e as suas anotações. Isto faz com que o utilizador tenha uma maior facilidade na importação destas anotações para uso a posteriori.

O menu *Files* tem como objectivo permitir o acesso e manuseamento de diversos vídeos da mesma aula. Um dos objectivos da aplicação é a extensão ao modelo das aulas tradicional, o que se traduz na necessidade de se utilizar mais que um conteúdo por aula, por exemplo um conjunto de dez vídeos. Este factor faz com que a aplicação possua uma opção, que permita escolher o ficheiro de vídeo em utilização entre todos os que foram carregados para uma dada aula.

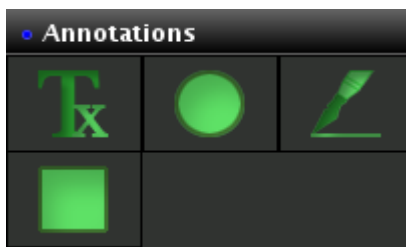
O menu *Help* tem como objectivo fornecer ao utilizador um conjunto de ajudas rápidas, sem que para isso o utilizador tenha que perder tempo a consultar um manual para executar as tarefas mais usuais.

### **Tipos de Anotações**

A área para gestão dos tipos de anotações (**Annotations**), que se situa no canto superior direito, agrupa as funcionalidades para a criação de um diverso leque de anotações por parte do utilizador. Os ícones de anotações correspondem aos tipos de anotações disponíveis na aplicação, tendo cada tipo um ícone respectivo. Analisando o desenho deste bloco observa-se que este apresenta ícones de grande dimensão (Figura 3.3), quando comparados com as restantes opções da aplicação, isto deve-se ao facto da interface ter como requisito a sua utilização em quadros interactivos.

Este conjunto de opções marca o início da criação de anotações, ou seja, o utilizador escolhe o tipo de anotação que pretende criar, e depois basta clicar na **Área de Criação e Visionamento de Conteúdos** para iniciar o processo de criação da anotação.

No processo de criação há ainda que ter em conta o clique que é feito no tipo de anotação pretendido: ao clicar com a tecla esquerda do rato numa das opções é criada uma **anotação estática**, caso seja clicada a tecla direita do rato (ou numa tecla especial, no caso dos quadros interactivos) será criada uma **anotação dinâmica**.



**Figura 3.3** - Tipos de anotações existentes.

### Listagem de Utilizadores

A listagem de utilizadores (**Participants**) (Figura 3.4) está inserida no conjunto de blocos da barra lateral da aplicação, com o objectivo facilitar as actividades colaborativas. Inicialmente quando o utilizador se junta a uma aula que esteja a decorrer, este campo será o primeiro a ser preenchido com nome de todos os participantes actuais. Mantém-se assim constantemente actualizado durante toda a aula, para que todos os intervenientes saibam quem está actualmente a frequentar esta aula.

Para além do aspecto meramente informativo da listagem de utilizadores, através deste bloco é possível proceder à partilha de anotações com um determinado utilizador, recorrendo a um sistema simples de **drag and drop**. O utilizador, para partilhar as anotações, arrasta uma ou mais anotações da sua lista, presente no bloco **Anotações Carregadas**, para cima do nome de um utilizador presente na listagem. Este tipo de interacção vai fazer com que o processo de partilha de informação seja simples e transversal ao mecanismo de interacção, pois quer se utilize o programa num quadro interactivo, quer se utilize num computador pessoal, o mecanismo de partilha será sempre intuitivo e fácil de executar.



**Figura 3.4** - Lista de utilizadores de uma sessão colaborativa.

No campo da partilha de anotações o professor possui uma opção que permite a partilha de uma ou mais anotações para todos os utilizadores de uma aula. Ao arrastar uma anotação para cima do seu nome na lista de utilizadores, a anotação irá ser difundida por todos os utilizadores.

### **Anotações Carregadas**

A área Anotações Carregadas (**Annotations files**) da barra lateral (Figura 3.5) é responsável pela listagem de todas as anotações que foram carregadas, localmente ou remotamente através de partilha, para a aula actual. As anotações encontram-se organizadas hierarquicamente consoante o autor da mesma. Isto faz com que se mantenha a organização quando o número de anotações aumenta. Num nível mais baixo da hierarquia, isto é, dentro de cada item autor, existe uma listagem com as suas anotações. Focando ainda a optimização do espaço reservado a esta listagem quando o número de anotações aumenta, é interessante salientar o facto do utilizador poder esconder as anotações de um determinado autor, através de um clique nesse nó da hierarquia.



**Figura 3.5** - Listagem das anotações que estão a ser utilizadas pelo utilizador.

Cada item de anotação nesta listagem possui um vasto leque de opções:

- Possibilidade de arrastar para cima de um nome de utilizador permitindo a partilha da anotação;
- Ao clicar num elemento da listagem, o vídeo vai para o início da anotação em causa;
- Informação acerca da possibilidade de editar ou não editar a anotação;

Na globalidade são estas as funcionalidades da listagem de anotações, mas a utilização deste bloco de informação também pode ser utilizado em conjunto com a **Planificação das Anotações**, permitindo completar cada um dos tipos de descrição das anotações. Ou seja, um tem um objectivo de fornecer informação global, o outro foca-se mais no enquadramento temporal das anotações.

### Planificação das Anotações

A área de **Planificação das Anotações (Annotation Plan, da Timeline)** (Figura 3.6) encontra-se por baixo da linha temporal do vídeo. O seu objectivo é representar as anotações que o utilizador já carregou de uma forma temporal, isto é, todas as anotações são representadas através de linhas com círculos nos extremos, por forma a que o utilizador possa facilmente aperceber-se das relações e sucessões que as anotações possuem entre si.

Em relação ao espaço reservado para esta área, optou-se por se mostrar apenas três linhas de cada vez de forma a racionalizar o espaço existente. Caso seja necessário mais linhas para representar todas as anotações são criadas novas linhas que podem ser acedidas pela barra de *scroll*. O conceito de linhas existe para evitar sobreposições no planeamento das anotações, causadas pela intersecção das mesmas a nível temporal (Figura 3.6), para isso é utilizado um algoritmo que faz a racionalização do espaço existente para a representação. Assim, o algoritmo vai determinando que anotações podem ser representadas na mesma linha, e caso isso não seja possível cria uma nova linha de representação.

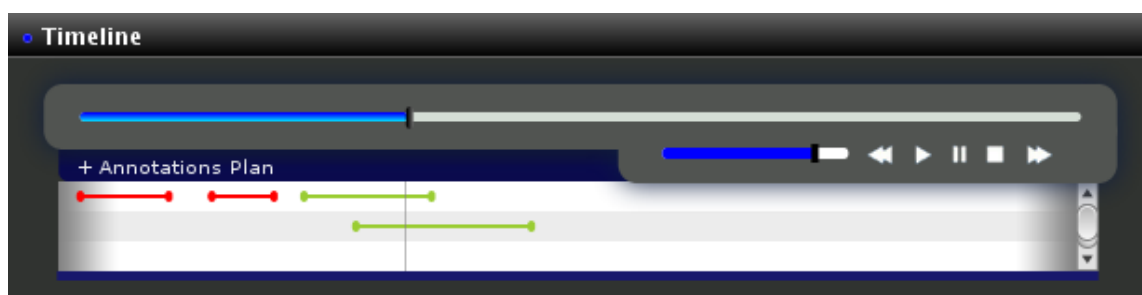


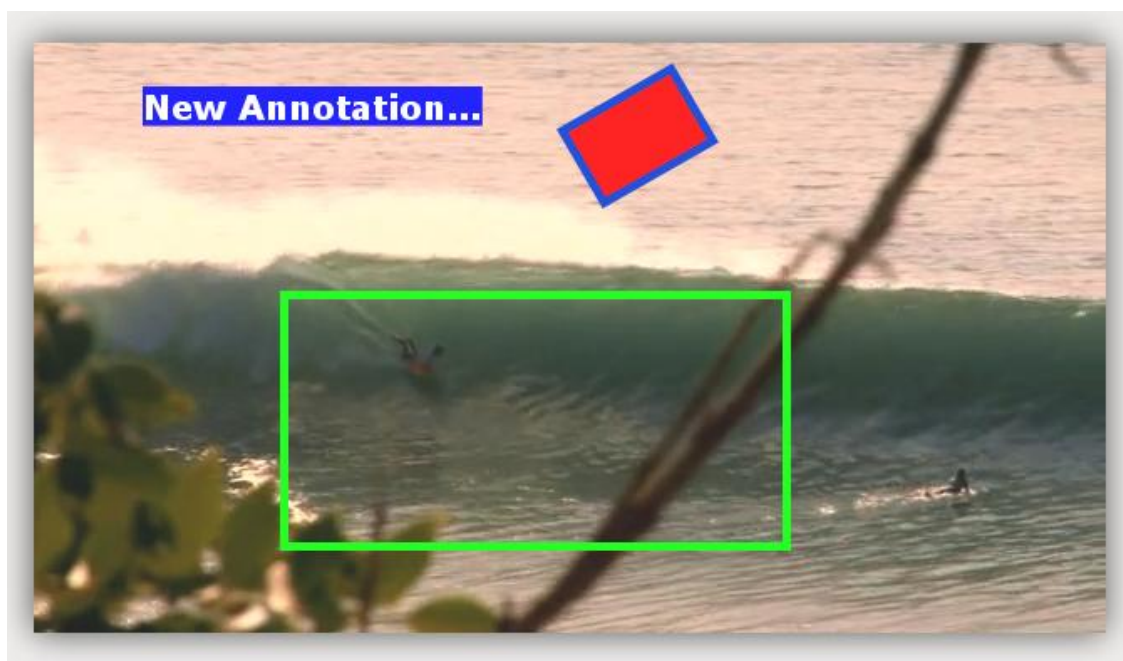
Figura 3.6 - Panorama temporal das anotações na área de planificação.

Como já foi descrito, esta área tem uma componente informativa, mas para além desse teor possui ainda uma característica que permite que a localização temporal de uma dada anotação seja editada. Ao clicar e arrastar uma das extremidades da linha vermelha que representa a anotação (círculos), está a ser executada automaticamente uma edição do ponto de início ou final da anotação.

Esta interface de manipulação temporal de anotações adapta-se às características particulares do quadro interactivo, uma vez que as circunferências que se encontram nos extremos das linhas vermelhas são demasiado pequenas para serem movidas neste dispositivo. Esta adaptação foi executada através da implementação de uma operação, que ao detectar que o rato está a passar por cima de uma linha (anotação) na área de Planificação de Anotações, o raio das circunferências é aumentado para tornar a anotação operável no quadro interactivo.

### Área de Criação e Visionamento de Conteúdos

Esta área é responsável pela visualização de conteúdos, tanto de anotações como de vídeo. À medida que o vídeo progride no tempo é mostrado o *frame* actual através do *player*, e são desenhadas as anotações que têm de ser apresentadas. Assim obtém-se o efeito de sobreposição das anotações em relação ao vídeo (Figura 3.7), que é um factor fundamental na contextualização das mesmas.



**Figura 3.7** - Área de Criação e Visionamento de Conteúdos.

A componente de manuseamento de anotações também se encontra disponível nesta área, onde o utilizador pode criar e editar as anotações do vídeo. No caso de o utilizador pretender criar anotações, terá de seguir o processo já mencionado no tópico Tipos de Anotações da secção 3.3.1, em que ao ser clicado um tipo de anotação o utilizador fica automaticamente habilitado a criar nesta área uma anotação do tipo escolhido.

No caso das anotações já existentes podem ser desencadeadas duas opções: a primeira diz respeito ao clique da tecla esquerda do rato sob a anotação, que faz com que seja aberto um explorador de Internet para uma hiper-ligação definida nas propriedades da anotação; a segunda, é executada através do clique da tecla direita, e faz com que as anotações possam ser alteradas pelo autor das mesmas, utilizando assim a opção de edição de notas presente na aplicação. Este tipo de edição visa a composição visual, e onde o utilizador pode trocar diversos tipos de atributos gráficos da anotação através do formulário **Edição de Anotações** descrito na secção 3.3.2. Assim, a componente de edição temporal das anotações fica delegado para a área de planificação de anotações descrita anteriormente.

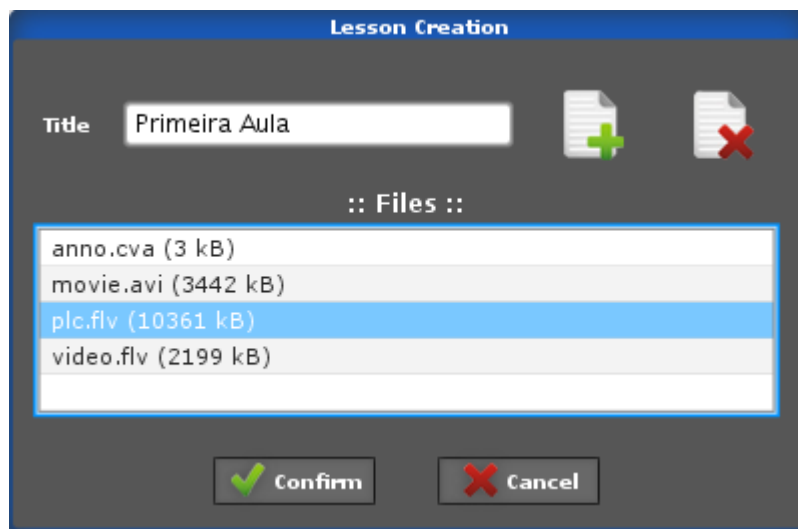
### **3.3.2 Formulários**

Os formulários são mecanismos que visam complementar a execução de certas acções, que necessitem de um número considerável de opções de parametrização. Essencialmente encontram-se entre o início da operação desejada e a fase de finalização, sendo utilizados como intermediário entre os dois, recebendo e tratando as opções.

#### **Criação de Aula**

Quando um professor pretende criar uma aula é fundamental ser recolhida alguma informação acerca da mesma (Figura 3.8). O primeiro passo na criação de uma aula é a definição de um título, para que a aula possa facilmente ser identificada. Seguidamente é efectuado o passo que mais influencia a aula, e que é a definição dos conteúdos que serão carregados. Sempre que o professor quiser começar uma sessão colaborativa através do anotador, terá de seleccionar que conteúdos, anotações e vídeo pretende disponibilizar para a sessão em causa.



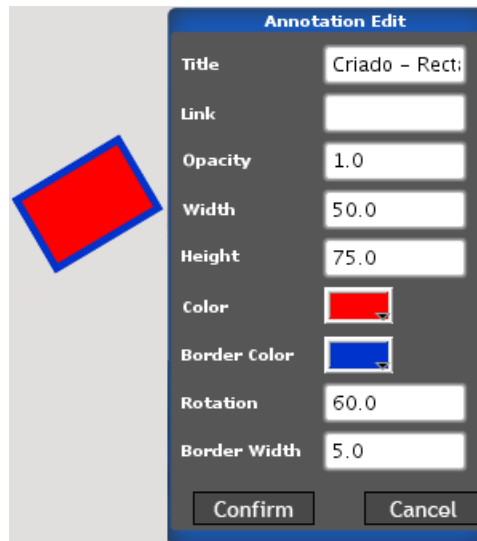


**Figura 3.8** - Formulário de criação de uma aula.

Através da interface disponibilizada o utilizador poderá adicionar e remover conteúdos, de forma a criar uma listagem final que traduza as suas necessidades para a aula. Para finalizar o processo, o coordenador poderá carregar no botão confirmar, para que a sessão comece e os conteúdos fiquem automaticamente disponíveis, para todos os participantes actuais da aula.

### **Edição de Anotações**

O formulário de edição (**Annotation Edit**) tem como objectivo a centralização de diversos atributos referentes às anotações (Figura 3.9). Através dele o utilizador poderá editar diversos tipos de campos, tais como: numéricos, texto e cor. Para além das capacidades ao nível de edição, o formulário permite fazer alterações nos campos sem ter de as submeter, isto é, ao ser modificada uma opção é logo aplicada para que o utilizador tenha uma pré-visualização da edição efectuada.

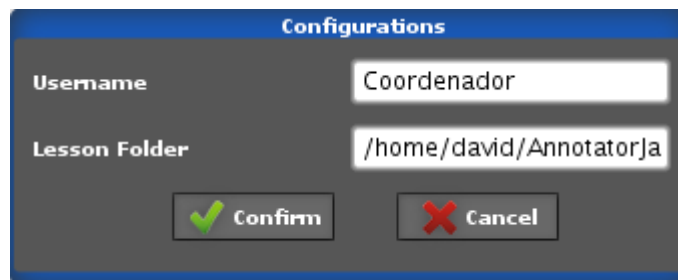


**Figura 3.9** - Formulário de edição de uma anotação do tipo Rectângulo.

Particularmente, os campos do formulário variam de tipo de anotação para tipo de anotação. Cada um dos tipos existentes na aplicação possui um conjunto de características diferentes, o que faz com que anteriormente só tenha sido particularizado o tipo de campos que são suportados pela interface, em vez dos atributos detalhados da anotação como acontece na secção 3.2.1.

### Preferências

No conjunto global da interface existe um determinado tipo de informação cujo preenchimento fica a cargo do utilizador, e é essa informação que se pretende tornar editável no formulário de preferências (Figura 3.10).



**Figura 3.10** - Opções disponíveis na área de preferências.

A primeira opção que pode ser editada é o nome de utilizador. Este nome é utilizado como identificador das suas anotações e serve para identificar o utilizador numa sessão colaborativa do sistema. A outra opção que pode ser editada neste formulário é a pasta de aulas, que tem como objectivo definir a pasta para onde serão guardadas todas as aulas em que o utilizador

participe. Resumidamente é uma pasta que mantém os conteúdos das aulas. Esta opção faz com que exista um desacoplamento ao nível do sistema de ficheiros entre os ficheiros da aplicação e os conteúdos pessoais do utilizador. O utilizador é livre de escolher a pasta para a qual pretende gravar o material da aula.

### 3.3.3 Notificações

Como o nome sugere as notificações têm por base a informação oferecida ao utilizador, acerca de acções que decorrem durante a execução da aplicação. É essencial informar e questionar o utilizador acerca de eventos que ocorrem. Assim as notificações presentes nesta interface encontram-se entre as meramente informativas e as que têm por base questionar o utilizador acerca de determinados eventos.

#### Eventos ocorridos

Este tipo de notificações tem como objectivo fornecer um determinado tipo de informação relevante no desenrolar das operações efectuadas pelo utilizador. Isto surge do facto de ser importante para o utilizador manter um histórico das operações desencadeadas, para que possa definir com maior facilidade as próximas operações a executar. Em termos de interface esta área de notificações exhibe apenas a última notificação gerada e caso o utilizador queira aceder ao histórico terá de carregar na seta (canto superior direito na Figura 3.11) a seguir ao texto da notificação, como demonstrado na Figura 3.11.

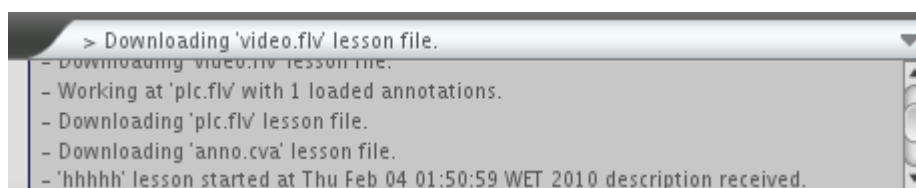
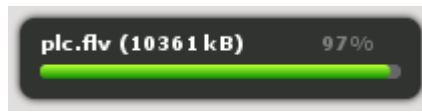


Figura 3.11 - Barra de notificação com a listagem de eventos expandida.

#### Recepção de Ficheiros

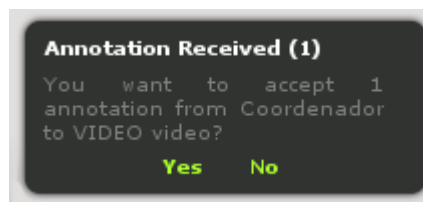
A notificação de recepção de ficheiros é sempre utilizada quando um aluno entra numa sessão colaborativa. Ao entrar, o aluno tem de receber todos os conteúdos que o professor designou para aquela aula. Por cada ficheiro que é carregado do professor (de forma sequencial) é feita uma caixa de notificação que indica o progresso de carregamento do ficheiro, como é demonstrado na Figura 3.12.



**Figura 3.12** - Notificação referente à recepção de um ficheiro originário do coordenador.

### **Anotações Partilhadas**

Para finalizar as funcionalidades associadas às notificações é descrita a caixa de diálogo que pergunta ao utilizador se pretende aceitar uma anotação de um determinado colega. Quando é recebida uma anotação, o utilizador terá de ter um papel interventivo na decisão de aceitação, ou seja, a aplicação não pode por si só aceitar uma anotação sem a autorização do utilizador. Para isso, é utilizada a notificação representada na Figura 3.13, que tem como objectivo mostrar alguma informação relevante da anotação que está a ser partilha, bem como as opções que permitam a aceitação ou rejeição, da mesma.



**Figura 3.13** - Notificação de recepção de uma anotação.

Genericamente, estes são os principais elementos da interface de utilizador do sistema. A utilização em modo autónomo parece-nos intuitiva e de extrema simplicidade, sendo ao nível dos mecanismos colaborativos e de suporte à gestão da aula que se situam os aspectos eventualmente mais complexos para o utilizador. De qualquer forma, os testes informais efectuados com utilizadores que à partida não conheciam o sistema indiciam que para um utilizador típico de computador, habituado a interfaces de janelas, a aplicação é simples de compreender e de utilizar. A criação das aulas e a partilha de anotações já não são tão evidentes, mas é necessário ter em consideração que as acções de partilha nas aplicações também são utilizadas com menor frequência

### 3.4 Arquitectura

A aplicação criada tem por base uma arquitectura global (Figura 3.14) onde estão definidos os principais blocos do sistema e sobre os quais se definem as suas respectivas interacções. Esta arquitectura é comum a todos os tipos de utilizadores, embora nem todos os fluxos de comunicação entre os blocos estejam disponíveis para todos os tipos de utilizadores. A figura seguinte descreve graficamente a arquitectura adoptada:

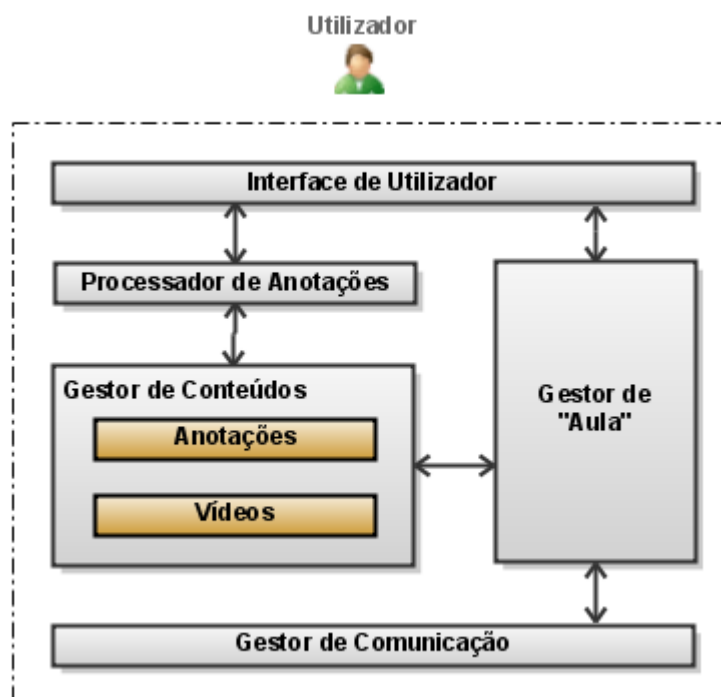


Figura 3.14 - Arquitectura global da aplicação.

O primeiro bloco da arquitectura é a Interface. Este tem como objectivo mediar a interacção do utilizador com todo o sistema, funcionando assim com mecanismo de abstracção de todas as operações internas da aplicação. O conceito de interacção com o utilizador é bastante amplo, pois o seu significado varia entre a forma como o utilizador cria e edita as suas anotações, e partilha informação com outros utilizadores, entre outros. Desta forma a Interface resume-se na íntegra ao grafismo que é apresentado ao utilizador, como foi descrito na secção 3.3. Este bloco da aplicação foi desenvolvido em **JavaFX** [23]. Esta linguagem tem como ponto forte o desenvolvimento de *Rich Internet Applications (RIAs)* e a vasta descrição de interfaces vectoriais.

Objectivamente a Interface só comunica com dois blocos na arquitectura global: o Gestor da Aula e o Processador de Anotações. A comunicação entre a Interface e o Gestor de

Aula centra-se na troca de informação referente a uma aula, quando o utilizador se encontra a trabalhar em modo colaborativo. Este tipo de informação é necessário em diversos cenários, como por exemplo quando um utilizador se liga a uma aula e lhe é passado o ficheiro de descrição da mesma, para que este possa descarregar os conteúdos; ou quando existe uma alteração na lista de utilizadores da aula e este tem de ser informado. O segundo bloco com o qual a Interface comunica é o Processador de Anotações, cujo objectivo principal se centra na mediação entre a Interface e o Gestor de Conteúdos.

A utilização do Processador de Anotações tem como objectivo executar tarefas de processamento nas anotações, antes das mesmas serem apresentadas no ecrã. Este processo é mais evidente na criação de anotações dinâmicas, onde a criação destas é feita de uma forma assíncrona relativamente à sua visualização. Isto faz com que ao ser iniciada a criação da anotação dinâmica, esta não fique logo disponível no gestor de conteúdos. Posteriormente, quando o Processador de Anotações acabar a tarefa de criação da anotação, é que a anotação fica pronta para ser gerida pelo componente de gestão de conteúdos.

O Processador de Anotações foi dos mais exigentes a nível técnico, pois a sua construção é feita através de uma solução híbrida de duas linguagens de programação: **JavaSE** [24] e **C** [25]. O **JavaSE** encarrega-se do trabalho lógico inerente às anotações dinâmicas, enquanto que a linguagem **C** é utilizada como um complemento nas anotações dinâmicas, visto que é através dela que é feito o módulo de anotações dinâmicas descrito nas secções 3.2.4 e 3.5.3, ou seja, é o módulo onde é executado o algoritmo que segue um determinado objecto, através técnicas processamento de vídeo.

O Gestor de Conteúdos tem por base a gestão de todos os conteúdos carregados pela aplicação, ou seja, na sua essência é composto por um conjunto de elementos que tem por objectivo tratar e organizar os conteúdos que se encontram carregados. Neste componente existem dois grandes tipos de conteúdos: as Anotações e os Vídeos. Teoricamente o anotador desenvolvido estabelece uma ligação entre estes dois tipos de elementos, mas na prática, tal ligação não foi estabelecida de uma forma vinculativa, isto porque ambos podem ser utilizados independentemente. Esta afirmação ganha particular relevo quando se partilham conteúdos com outros utilizados. Ao ser feita uma operação de partilha, os conteúdos são transferidos de uma forma independente, para aumentar a granularidade do fluxo de informação que é partilhado na aplicação.

É na óptica de utilização do anotador num cenário colaborativo, que surge a comunicação entre o Gestor de Conteúdos e o Gestor de Aula. Isto significa que quando existe um tipo de conteúdo (anotação ou vídeo) que seja transferido no decorrer da aula, o Gestor de Conteúdos será o primeiro componente a ser envolvido, de forma a gerir a informação recebida. Só depois deste processo ser concluído é que o utilizador é informado através da Interface, acerca do envio e/ou recepção de conteúdos.

O Gestor de Aula é um componente central do sistema, onde o professor gere a comunicação com todos os alunos. Assim, qualquer operação colaborativa estabelecida numa aula é transmitida para o utilizador final através do Gestor de Aula. Este componente pode ser comparado por analogia com um servidor no modelo de comunicação cliente-servidor, como um mecanismo que implementa a troca de informação de alto nível com os utilizadores, à semelhança de um *Web Service*. Por outro lado este bloco também é responsável pela partilha de anotações entre os alunos. Este factor surge da abordagem cliente-servidor adoptada no sistema, que centraliza todas as comunicações do sistema no professor (Gestor da Aula). Desta maneira, sempre que um aluno pretende partilhar uma anotação com um colega, a sua anotação passará pelo professor de forma transparente desempenhando meramente o papel de encaminhador de informação.

Como elemento final desta arquitectura temos o Gestor de Comunicação, que tem por base a tradução dos pedidos de alto nível feitos ao Gestor de Anotações, e o respectivo encaminhamento. Em certos casos até poderia ser considerado um subsistema do componente de gestão de anotações, visto que implementa operações que permitem disseminar a informação recebida por todos os utilizadores que se encontram ligados.

### 3.5 Aspectos de Implementação

Nesta secção descrevem-se alguns aspectos de implementação, que pela sua relevância possuem um impacto preponderante na boa execução da aplicação. Estes encontram-se focados maioritariamente na reutilização de código e na fácil expansibilidade das opções já existentes no protótipo. Estes aspectos de implementação abordam diferentes níveis da aplicação, como o modelo de dados, o suporte colaborativo e as anotações dinâmicas.

#### 3.5.1 Modelo de Classes

No que diz respeito ao modelo de classes pretende-se identificar as classes cujo papel é primordial na aplicação e a forma como internamente as classes que são as principais fontes de abstracção de código, trabalham para que o anotador seja eficiente. Sendo assim, serão descritas as classes **Anotação**, **Gestor de Anotações**, **Coordenador e Participante**, **Reprodutor Multimédia** e **Gestor Multimédia**.

##### Anotação

A classe anotação é uma super classe que é herdada pelos diferentes tipos de anotações (Rectângulo, Texto, Círculo e Linha), cuja implementação reflecte o processamento comum a todos os tipos de anotações (Figura 3.15). Tem um papel fulcral na aplicação, pois é o objecto que simboliza a anotação nas suas diversas formas. Começa inicialmente por processar a informação para a criação da anotação e assegura que a anotação é mostrada quando se encontra no seu tempo de execução.



Figura 3.15 - Organização das classes que herdam da anotação.

Inicialmente a anotação pode ser criada de duas formas, a primeira através de um ficheiro onde é feita a leitura de um excerto de XML através de um método presente na classe, de forma a preencher os atributos com base nessa informação. Por outro lado a anotação pode ser criada pela aplicação quando o utilizador a desenha e neste caso é criada uma anotação padrão que deverá ser personalizada pelo seu criador.



Seguindo a abordagem da leitura do formato XML por parte desta classe, é necessário salientar que a mesma também permite a criação de XML fundamental no processo de gravação de novos ficheiros de anotações. A classe implementa métodos que se aproximam à serialização de objectos Anotação recorrendo a XML.

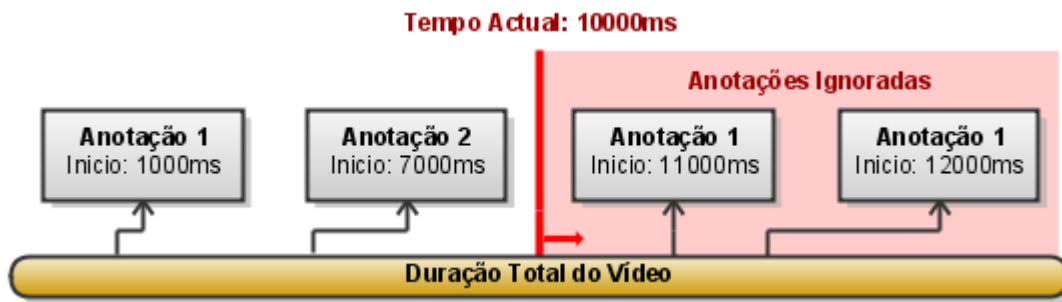
Outra grande vantagem do objecto Anotação é sua morfologia que permite à classe saber quando é que a anotação deve ser mostrada sobre o vídeo. Isto é possível devido a uma interligação entre esta classe e o Gestor de Anotações, que periodicamente executa um método que faz verificação de estado, utilizando como atributo o tempo actual do vídeo (*handleTime*).

Por fim existe ainda o processo de criação da representação visual do objecto, através do mecanismo de herança utilizado. Isto é, a representação gráfica da anotação é gerada pelas classes (Rectângulo, Texto, Elipse e Linha) que herdam da classe abstracta Anotação, de forma a conseguir criar em **JavaFX** o objecto gráfico correspondente.

### **Gestor de Anotações**

A classe de gestão de anotações é o ponto principal da aplicação, pois é o seu funcionamento que permite ao anotador mostrar e gerir todas as anotações carregadas pelo utilizador.

O primeiro ponto base da gestão de anotações é a sua organização interna de objectos do tipo Anotação armazenados numa lista de anotações, cuja ordenação é estabelecida crescentemente pelo tempo de início da Anotação. Com esta lista ordenada é possível criar uma optimização no varrimento das anotações, isto é, quando se está a visualizar um vídeo é feito um varrimento parcial de anotações por cada frame, em vez de um varrimento total da lista. Este princípio parte do pressuposto que ao ser lido o tempo actual do vídeo e estando as anotações ordenadas pelo seu tempo de início, implica que ao ser encontrada uma que possua um tempo de início maior que o tempo actual, o resto das anotações não precisam de ser mostradas (Figura 3.16).



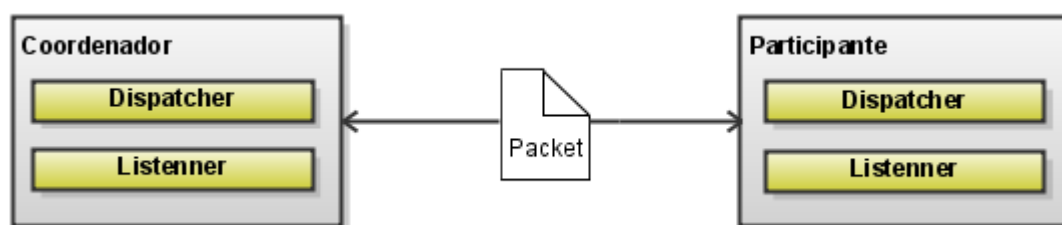
**Figura 3.16** - Organização dos objectos Anotação no Gestor de Anotações.

Ainda no âmbito do trabalho feito pelo gestor em relação às anotações que possui, pode-se constatar que este possui métodos que permitem carregar e guardar no formato XML. Este factor vai de encontro ao já referido na secção anterior, pois ao ser gerado um ficheiro com todas as anotações, vai ter de ser varrida a lista de anotações interna ao objecto Gestor de Anotações (*mediaPlayerProgressionTrigger*), e por consequência o método que permite uma Anotação ser serializada para XML irá ser executado (*handleTime*).

Por outro lado, o Gestor de Anotações também tem uma componente gráfica, que é utilizada quando o utilizador está a criar uma anotação. Isto leva a que ao ser criada uma anotação na interface, o gestor de anotações seja a classe delegada para determinar que tipo de objecto lógico irá ser criado, adicionando-o posteriormente à sua lista de anotações interna.

### Coordenador e Participante

O Coordenador e Participante são classes distintas, mas no âmbito da implementação existem semelhanças que os podem colocar no mesmo patamar. Isto é, apesar de um ser o cliente e o outro representar o servidor, ambos implementam métodos que efectuam o tratamento da troca da informação (Figura 3.17).



**Figura 3.17** - Troca de informação entre o Coordenador e um Participante.

Interiormente ambas as classes possuem dois mecanismos de troca de informação (pacotes). O primeiro é o *listener*, cuja função é ficar à espera de novos pacotes que lhe

sejam enviados. Estes pacotes de informação, descritos na secção 3.5.2, são essenciais para o bom funcionamento da aplicação no modo colaborativo, ou seja, no contexto de uma aula. O segundo mecanismo é o *dispatcher*, cujo objectivo passa por enviar pacotes para um cliente remoto. Este mecanismo induz mais cuidados que o anterior, pois teve de ser elaborado através de uma **Fila** (estrutura de dados) de pacotes. Necessidade esta que surgiu do facto de poderem ser enviados diversos pacotes continuamente. Sem nenhum mecanismo de organização do envio de pacotes poderia surgir uma obstrução do canal de comunicação.

Para além dos aspectos de comunicação, estas classes demonstram a segmentação existente no programa em relação às duas linguagens utilizadas, **JavaFX** e **JavaSE**. **JavaSE** assume um papel de grande relevo no mecanismo de comunicação da aplicação, assim sendo, toda a componente de troca de pacotes foi elaborada em **JavaSE** e em **JavaFX** só é feita a gestão de eventos desencadeados (Secção 3.5.2).

### Reprodutor Multimédia

O Reprodutor Multimédia é essencialmente uma classe abstracta, que visa cobrir o suporte de vários formatos na aplicação. Actualmente o anotador reproduz vídeos, visto ser a única implementação da classe abstracta Reprodutor Multimédia, mas o objectivo é que futuramente consiga reproduzir Apresentações PowerPoint e GIFs Animados entre outros formatos úteis no contexto da aplicação (Figura 3.18).



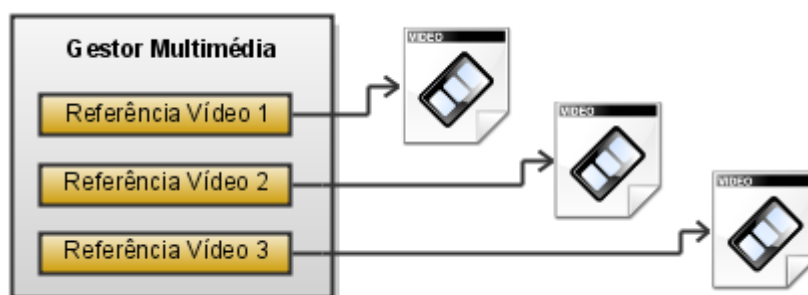
**Figura 3.18** - Classes que podem herdar da classe Reprodutor Multimédia.

Essencialmente esta classe pretende fornecer uma interface comum, que através de um conjunto de métodos emula as principais operações efectuadas num vídeo. Um vídeo suporta as operações iniciar, parar, pausa, passar para a frente, entre outras, e pretende-se assim, que outros formatos possam implementar este tipo de métodos por analogia. Por exemplo, no caso de um vídeo a operação "passar para a frente" determina que a reprodução do vídeo seja mais rápida, logo a analogia para o caso de uma Apresentação PowerPoint poderia ser a troca de diapositivo.

## Gestor Multimédia

O Gestor Multimédia tem por base a necessidade da aplicação trabalhar com diversos elementos multimédia (vídeos, imagens, apresentações PowerPoint, entre outros tipos que possam ser suportados pela classe Reprodutor Multimédia) na mesma aula. Particularizando esta abordagem ao caso do vídeo, deduz-se que durante uma aula ou até mesmo em trabalho autónomo, o utilizador pode ter de trabalhar com mais que um ficheiro de vídeo. Para isso tem de haver um elemento interno no programa que permita armazenar temporariamente os conteúdos carregados. É neste âmbito que o Gestor de Multimédia tem uma especial importância, ou seja, para que a operação de troca de vídeo (Secção 3.3.1) seja rápida e intuitiva, é necessário que este objecto mantenha a referência para os vídeos que estão a ser utilizados na aplicação (Imagem 5).

O aspecto anteriormente mencionado é igualmente notório no caso das anotações, pois para além da troca de vídeo na aplicação, é necessário executar uma procura no Gestor de Anotações para que todas as anotações referentes ao vídeo sejam carregadas para a aplicação. Este facto surge facilmente no contexto colaborativo, pois quando o professor inicia aula escolhe logo à partida um leque de ficheiros de vídeos e anotações, que em muitos dos casos podem não ser correspondentes. Por exemplo, quando são carregados dois vídeos e três anotações significa que as três anotações poderão não pertencer ao mesmo vídeo, logo só têm de ser adicionadas à Lista de anotações do Gestor de Anotações quando o vídeo às quais pertencem é escolhido no Gestor Multimédia.



**Figura 3.19** - Gestão interna de vídeos no Gestor Multimédia.

### 3.5.2 Suporte Colaborativo

Esta secção tem como objectivo descrever alguns pormenores relacionados com o funcionamento da aplicação em modo colaborativo. Em termos de implementação, os blocos definidos na Arquitectura (Secção 3.4), foram implementados em **JavaSE**, pois o **JavaFX** não possui nenhum mecanismo interno na sua *framework* que permita fazê-lo.

Como já foi referido anteriormente (Secção 3.4), o **JavaFX** foi desenvolvido com o intuito de criar *Rich Internet Applications* e por consequência houve um maior foco nos mecanismos de suporte à comunicação, de alto nível através de *Web Services*. No caso do suporte colaborativo as necessidades de comunicação residiam a um nível mais baixo, pois para a implementação do mecanismo de gestão de pacotes e troca de informação era necessário recorrer a *Sockets*.

#### Interacção entre Utilizadores

O modo de funcionamento colaborativo, definido no contexto desta dissertação como "aula", prevê a troca de informação entre todos os tipos de utilizadores. De forma a concretizar este modelo de comunicação entre os utilizadores, foi criado um mecanismo de troca de dados onde a unidade de troca é denominada pacote (objecto serializável). A definição de pacote reside no facto do bloco de informação estar pré-definido e por sua vez representar um conjunto de informação contextualizada.

Assim, uma primeira análise passou pela identificação dos tipos de pacotes que seriam necessários para garantir uma boa comunicação entre os utilizadores da aplicação. Estes pacotes tinham de representar toda a informação que se pretendesse trocar, bem como, quando necessário, tornarem-se abstractos para ser possível garantir que podem ser transferidos entre todos os tipos de utilizadores. No total foram criados seis tipos de pacotes (Figura 3.20), mas para além deste tipo de troca de informação, há ainda a considerar a informação fornecida pelo professor através do HTTP Server, descrita no tópico seguinte.

Pacote	Conteúdo	Actores
<b>Lista_Utilizadores</b>	Lista de utilizadores actuais na aula.	Professor → Aluno
<b>Identificação_Utilizador</b>	Identificação do utilizador que pretende se ligar à aula.	Aluno → Professor
<b>Descrição_Aula</b>	Título e informação dos conteúdos que precisarão de ser descarregados.	Professor → Aluno
<b>Anotações</b>	Lista de uma ou mais anotações.	Professor ⇔ Aluno Aluno ⇔ Aluno
<b>Ligação_Terminada</b>	Razão pela qual foi terminada a ligação.	Professor ⇔ Aluno
<b>Erro_Identificação</b>	Erro ocorrido durante o processo de identificação do aluno.	Professor → Aluno

**Figura 3.20** - Pacotes da aplicação e respectivas interacções.

Como se observa na Figura 3.20, os diferentes tipos de pacote enquadram as principais necessidades do utilizador, no que respeita aos tipos de interacção que tem com a aplicação. Um dos tipos de pacote mais relevante, no âmbito deste trabalho é o pacote **Anotações**. A sua importância deve-se ao facto da sua utilização ser transversal a todos os tipos de utilizadores e de, no fundo, este pacote ser o principal suporte do mecanismos de partilha de anotações entre os utilizadores.

### Servidor HTTP

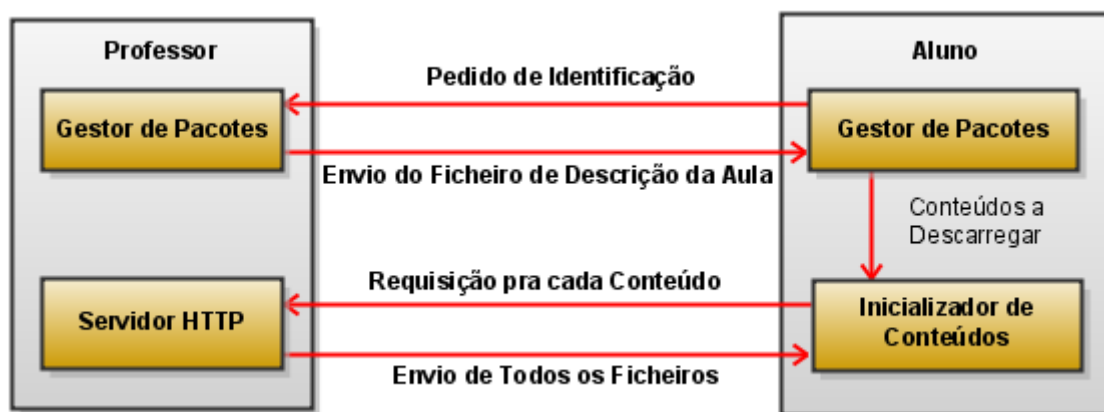
O Servidor HTTP tem um papel preponderante na acção do professor enquanto coordenador da aula, isto porque é através do Servidor HTTP que são partilhados todos os conteúdos considerados necessários, no âmbito de uma aula. Com esta solução o professor pode facilmente disponibilizar o material da aula sem que para isso seja necessário criar uma solução mais dispendiosa. Pois graças ao objecto *HTTPServer* presente na *Framework* do **JavaSE** não foi necessário implementar de raiz uma solução de partilha de conteúdos.

A inicialização do servidor HTTP surge quando o professor inicia uma sessão em modo de funcionamento colaborativo, isto é, quando inicia uma aula na aplicação. Em primeiro lugar o professor tem de escolher todos os ficheiros (conteúdos) que pretende usar na aula, como descrito no tópico “Criação de Aula” da secção 3.3.2 Quando os ficheiros se encontram todos escolhidos e o professor inicia a aula, é executado um método que tem como objectivo copiar

todos os conteúdos escolhidos para a pasta raiz (*sharing*) do Servidor HTTP. Desta maneira, quando a cópia de ficheiros terminar e o servidor se iniciar, os conteúdos passam a estar disponíveis para serem descarregados por parte dos alunos.

Com a implementação de um Servidor HTTP surgiram alguns detalhes que necessitaram de ser ajustados, para que o processo de partilha decorresse de uma forma transparente para o professor e para os alunos. Em primeiro lugar, era necessário encontrar uma forma de garantir que o aluno conseguiria saber qual a matéria necessária para a aula, e consequentemente, quais as hiperligações no Servidor HTTP para os respectivos conteúdos.

A solução adoptada teve por base a utilização de um pacote com a informação da aula, que é enviado para o aluno aquando da sua entrada na aula (*connect*). Este pacote contém à partida as hiperligações de cada ficheiro necessário para a aula. A figura seguinte descreve visualmente este processo.



**Figura 3.21** - Obtenção dos conteúdos por parte do aluno.

O aluno liga-se à aula e em resposta é lhe enviado um Ficheiro de Descrição da Aula. Quando o aluno recebe o Ficheiro de Descrição da Aula pede ao Inicializador de Conteúdos, para descarregar todos os ficheiros mencionados (Figura 3.21). Após a obtenção de conteúdos, o aluno não necessita de voltar a obter informações/conteúdos do professor através do Servidor HTTP. No fundo este servidor serve apenas para facilitar o envio de grandes quantidades de informação para um diverso número de utilizadores. As trocas de informação subsequentes (troca de anotações, envio da lista de alunos actuais) entre o professor e os alunos, são feitas através do mecanismo de Gestão de Pacotes.

## Sincronização de Anotações

A sincronização de anotações tem como objectivo garantir que não há duplicação de anotações durante o processo de partilha. Isto é, ao ser partilhada uma anotação com um dado utilizador, se passado um intervalo de tempo essa anotação for novamente partilhada com alterações, o utilizador que a recebe não passa a ter duas anotações distintas, mas sim uma só, actualizada. De notar que esta anotação partilhada é propriedade do seu autor e como tal o conteúdo da nota é da total responsabilidade de quem a criou. Consequentemente pretende-se que as actualizações à anotação, efectuadas pelo seu autor, sejam difundidas por todos aqueles com que a anotação foi partilhada.

Para garantir este pressuposto foi necessário distinguir univocamente as anotações, um aspecto essencial na detecção de anotações duplicadas. Esta distinção foi feita através de um identificador único, composto por dois atributos da anotação concatenados: a data de criação da anotação e o seu autor. Desta forma não é permitido a um utilizador criar duas anotações em simultâneo, sendo esta restrição aceitável no contexto deste protótipo.

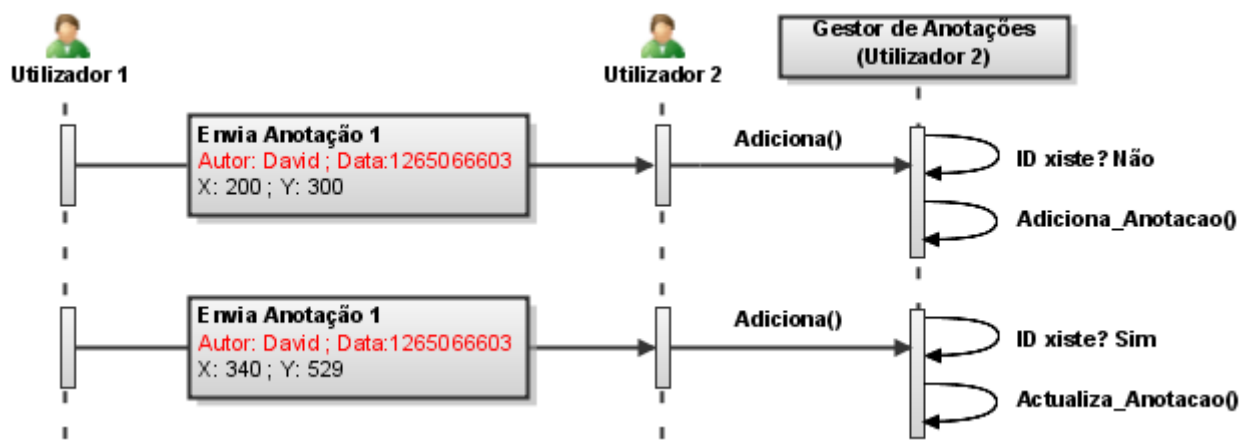


Figura 3.22 - Exemplo de uma anotação partilhada duas vezes com atributos distintos.

Criado um identificador único para a anotação, a identificação individual das notas tornou-se mais simples, mas continuava a ser necessária a criação de um mecanismo que permitisse identificar a existência da actualização de uma anotação. Perante este cenário foram tidas em conta duas alternativas: a actualização automática da anotação; ou a actualização explícita da anotação por parte do seu autor, através de uma nova partilha da mesma.



A primeira foi abandonada por se achar que retirava ao autor, controlo sobre informação que lhe é privada, as suas notas. Isto é, ao partilhar uma anotação, o utilizador está a dizer que naquele contexto a anotação é pertinente para ser partilhada com aquele conteúdo, o que não implica que quando a anotação sofra algum tipo de alteração, o autor a queira ver obrigatoriamente partilhada.

A segunda opção, de efectuar a actualização através de uma acção explícita do autor, pretende dar ao utilizador a possibilidade de ser ele a decidir se anotação deve ou não ser novamente partilhada. Desta forma, quem recebe a anotação é simplesmente informado acerca da partilha de uma anotação já existente, e esta é feita sem haver nenhuma duplicação de notas (Figura 3.22). De notar que independentemente da nova partilha gerada pelo autor da anotação, cabe ao utilizador decidir se aceita ou não a actualização da anotação.

### **3.5.3 Anotações Dinâmicas**

As anotações dinâmicas são, no contexto deste trabalho e como já foi referido, um tipo de anotação cuja localização se altera ao longo do tempo, com o objectivo de assim permitir o seguimento de um determinado objecto. Numa perspectiva mais técnica, estas anotações pretendem analisar o movimento de um objecto escolhido pelo utilizador e por sua vez calcular a rota que o mesmo seguirá após um terminado momento temporal. Para implementar este tipo de opção foi utilizada a biblioteca OpenCV 2.0 [26], cuja estrutura já implementa alguns algoritmos de análise de movimento. Desta maneira foi necessário desenvolver um módulo para a aplicação desenvolvido na linguagem C como será descrito nesta secção.

#### **3.5.3.1 Possíveis Tecnologias para Implementação**

O conceito de anotação dinâmica pressupõe logo à partida processamento de vídeo, pois a necessidade de analisar movimento implica o processamento contínuo dos *frames* existentes no vídeo, de forma a determinar a trajectória do objecto. Esta necessidade faz com que a tecnologia escolhida para criar as anotações dinâmicas possua algoritmos rápidos e eficazes de processamento de vídeo.

## JavaFX

Inicialmente pretendia-se que fosse tudo implementado nesta plataforma, o que rapidamente se demonstrou impossível. **JavaFX** [23] é uma plataforma muito recente e com objectivos bem traçados na área do vídeo, baseados na criação de **Rich Internet Applications** (RIAs). Esta caracterização esconde algumas limitações que levaram a que na sua utilização só tenha sido explorada a vertente de reprodução de vídeo, colocando de parte a sua utilização para edição e processamento.

## JavaSE

Como já fora mencionado, a tecnologia **JavaSE** [24] pode ser facilmente utilizada através de **JavaFX** (linguagem utilizada na interface). Esta hipótese demonstrou alguma pertinência devido à componente de processamento de vídeo disponibilizada através da **Java Media Framework** (JMF) [27]. Desta maneira seria possível analisar os *frames* para calcular a trajectória de um objecto, mas esta escolha traria uma considerável limitação ao nível dos formatos de vídeo suportados pela aplicação, visto que a JMF está limitada ao nível dos formatos de codificação. No fundo seria uma abordagem algo contraditória: a interface feita em **JavaFX**, que impulsiona o uso dos novos formatos e dos novos *containers* liderados pelos formatos FLV, MP4, AVI, entre outros; por outro lado havia uma abordagem limitada inerente à utilização da **JMF**.

## C com OpenCV

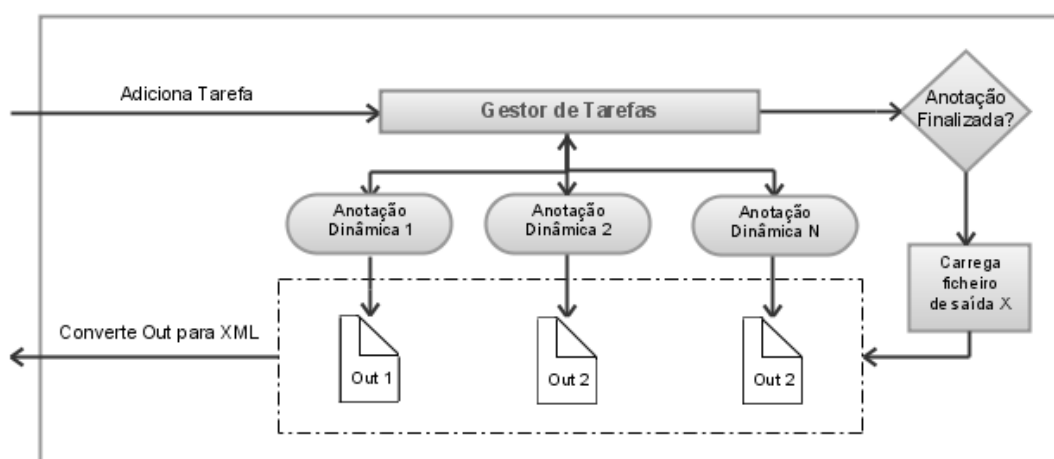
**OpenCV** [26] é uma biblioteca criada pela Intel sob uma licença de código livre, que tem como objectivo fornecer um conjunto de ferramentas e algoritmos relacionados com a visão computacional. Funcionalidade essa que é útil na criação de anotações dinâmicas, em grande parte por causa da implementação do algoritmo Lucas-Kanade[28][26], cuja característica base se centra na análise de movimento de um dado objecto. O **OpenCV** possibilita ainda a utilização da biblioteca *ffmpeg* para descodificação de vídeo, isto é, ao ser compilado com *ffmpeg* fica habilitado a interpretar grande parte dos formatos de vídeo existentes, o que representa uma grande vantagem para aplicação.

Assim sendo, a escolha para elaborar o módulo de anotações dinâmicas recaiu sob o **OpenCV**, garantindo assim uma maior compatibilidade no que diz respeito às ferramentas e aos formatos utilizados. Esta escolha também tem algumas desvantagens, nomeadamente a

necessidade do módulo ser criado na linguagem C, o que leva a um aumento do tempo despendido neste módulo e a integração com o que já fora feito em **JavaFX**. Elevando assim para três o número de linguagens utilizadas no protótipo, o que normalmente requer alguma atenção para evitar problemas de interligação entre componentes.

### 3.5.3.2 Módulo de Anotações Dinâmicas

O módulo de anotações dinâmicas é essencialmente uma aplicação feita em C, cujo funcionamento deriva da utilização do algoritmo Lucas-Kanade[28][26]. A rapidez é uma das vantagens desta implementação, que executa assincronamente em relação à execução dos outros módulos, isto é, quando é criada uma anotação dinâmica é adicionada uma nova tarefa, cujo objectivo é controlar o fluxo da anotação criada. Isto significa que a anotação é adicionada a uma lista de anotações em espera e que quando a mesma finaliza é disparado um evento a informar a aplicação que esta terminou, para que possa ser lido o ficheiro de saída que descreve a anotação dinâmica (Figura 3.23).



**Figura 3.23** - Diagrama representativo do módulo de anotações dinâmicas.

### Inicialização

Ao nível de execução, este módulo funciona exactamente como uma aplicação independente. Tipicamente para ser executado é necessário criar outro processo no **JavaSE** para este módulo. Esta abordagem também poderia ser feita através do **Java Native Interface** (JNI) [29] presente na *Framework* da **Sun**, mas no caso deste protótipo a sua utilização não

traria nenhum benefício. Assim, o módulo será executado através da classe **Runtime** do **JavaSE**, uma abordagem simples e facilmente implementável.

Em primeiro lugar é preciso definir os parâmetros (Tabela 3.6), que são utilizados nesta execução, muitos deles são parâmetros para a inicialização da análise de imagem, mas também existem parâmetros que visam otimizar o tempo despendido na execução da tarefa. Um exemplo desta otimização é o **startTime**, este parâmetro é formalmente utilizado na indicação do início da anotação, mas para além disso indica onde é que vai ser inicializado o processamento de *frames* do vídeo. Considerando, por exemplo, que a anotação iria ser criada aos vinte mil milissegundos e o vídeo tinha uma propriedade de trinta *frames* por segundo, caso não houvesse esta otimização seriam processados seiscentos *frames* inúteis para o processo de criação da anotação dinâmica.

Parâmetro	Descrição
<b>videoPath</b>	Caminho absoluto do ficheiro de vídeo a anotar, no sistema de ficheiros.
<b>startTime</b>	Instante de tempo (em milissegundos), no qual vai ser inicializado o processo de criação da anotação.
<b>xPos</b>	Coordenada X onde a anotação vai começar a ser inicializada.
<b>yPos</b>	Coordenada Y onde a anotação vai começar a ser inicializada.
<b>jobId</b>	Identificação unívoca da tarefa criada, na classe que gere a criação de anotações dinâmicas.
<b>debug</b>	Variável binária que quando é verdadeira renderiza o processamento da anotação numa janela, de forma a tornar possível o visionamento do algoritmo.

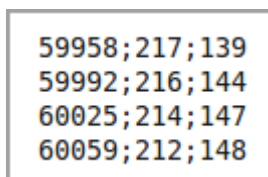
**Tabela 3.6** - Parâmetros do módulo de anotações dinâmicas.

### Ficheiro de Saída

O ficheiro de saída é responsável por guardar o resultado da anotação dinâmica num formato previamente definido, sendo o seu nome igual ao **jobId** definido como parâmetro do módulo. Esta característica surge da necessidade de fazer a comunicação de valores entre o módulo e a interface **JavaFX**, como será constatado posteriormente.

A criação deste ficheiro dá-se no início do processo de criação, e estende-se ao longo de todo o período de análise do vídeo, para que possa armazenar todos os passos da anotação em

questão. O ficheiro de saída é dado por um conjunto de linhas de texto, onde cada uma representa uma posição num determinado momento. Simbolicamente esta informação representa-se por *TEMPO\_MILISSEGUNDOS;POS\_X; POS\_Y*, como sugere a Figura 3.24, ou seja, uma metodologia básica e de fácil implementação feita à base de separadores (ponto e vírgula), que é consumida quando termina a criação da anotação.



```
59958;217;139
59992;216;144
60025;214;147
60059;212;148
```

Figura 3.24 - Output gerado pelo módulo.

### 3.5.3.3 Integração com a interface JavaFX

O processo de criação de anotações dinâmicas é desencadeado na interface **JavaFX**, através de um clique com a tecla direita do rato no tipo de anotação que se pretende criar. A partir deste momento o utilizador só terá de começar a desenhar a anotação dinâmica, para que o módulo seja disparado.

Este processo será inicialmente processado pelo **JavaSE**, ambiente no qual foi implementada a classe que gere as tarefas em execução, pois poderá haver mais que uma anotação dinâmica em processo de criação. A nova tarefa a ser criada vai ter um identificador único gerado pelo gestor de tarefas, de forma a garantir que esta possa ser identificada em qualquer um dos passos do processo, ou seja, a tarefa tem de ser identificada desde a criação, passando pelo processo de composição, até à parte de finalização. Assim, torna-se possível seguir a execução de uma anotação dinâmica no momento em que informação da mesma é transferida entre processos diferentes.

Como já foi descrito, este processo é assíncrono em relação à execução da interface, logo quando uma tarefa finaliza e é disparado o evento no **JavaSE**, o gestor de tarefas tem de ser capaz de identificar e obter o ficheiro de saída (resultado) da tarefa que finalizou.

A partir de momento que a aplicação sabe qual foi a tarefa que acabou é feita uma consulta de forma a obter o resultado do processamento, onde o ficheiro de saída terá de ser analisado e traduzido para XML sob o formato **CVA** (*Collaborative Video Annotator*) suportado pelo anotador criado.

## **4 Considerações Finais**

Neste capítulo consideraram-se duas vertentes, a primeira aborda as conclusões deduzidas ao longo da dissertação e a segunda faz um resumo das linhas gerais quanto ao trabalho futuro desta dissertação. Pretende-se assim considerar o trabalho desenvolvido, através dos pontos que ainda necessitam de ser ajustados, apontando caminhos e possíveis extensões da aplicação a implementar futuramente.

### **4.1 Conclusões**

As conclusões são naturalmente um reflexo dos objectivos anteriormente enunciados, pelo que se justifica que sejam expressas a vários níveis.

Em primeiro lugar é interessante abordar os aspectos relacionados com o desenvolvimento de uma solução que permita, de uma forma simples, implementar numa sala de aula tradicional um mecanismo de partilha de conteúdos audiovisuais sobre o qual é possível efectuar anotações. Penso que a este nível a solução apresentada permite atingir os objectivos propostos e abre caminho à possibilidade de com uma infra-estrutura mínima, ser possível:

- Um professor, partilhar conteúdos audiovisuais e anotações;
- Permitir que se efectuem anotações no decorrer da aula (professor e alunos);
- Que alunos partilhem anotações entre si (e com o professor);
- Que as anotações possam ser alteradas e estas alterações disseminadas pelos utilizadores que as partilhavam;

A utilização de quadros interactivos era de igual forma uma vertente importante a explorar e nesse sentido os resultados obtidos foram igualmente satisfatórios. A interface desenvolvida é de simples utilização e em tudo idêntica à existente para os computadores pessoais.

Ao nível da utilização dos quadros interactivos é importante realçar os seguintes aspectos:

- Ficou disponível uma aplicação que permite sobre um quadro interactivo, gerir os vídeos da aula, efectuar anotações e partilhar anotações com os alunos;
- A aplicação para o quadro interactivo permite gerir e manipular os conteúdos e as anotações criadas previamente pelo professor no seu portátil;
- A aplicação permite ao professor, ao expor a matéria, criar e partilhar anotações com os alunos;

A criação de uma aplicação que fosse simples de utilizar sobre um quadro interactivo obrigou a cuidados acrescidos ao nível da interface. Como se pode verificar ao longo da descrição da interface (Secção 3.3), todos os mecanismos de interacção são dispostos com um razoável grau de separação e as operações centram-se essencialmente em interacções de toque únicos. Para além destes detalhes, ainda foram implementados pormenores que permitem aumentar a operacionalidade de elementos de reduzida dimensão, como por exemplo a barra de planificação. Quando o rato passa por cima da linha que representa temporalmente a anotação, as circunferências que se encontram nos seus extremos aumentam.

Finalmente, existem um conjunto de conclusões relacionadas com aspectos de implementação e com as soluções tecnológicas adoptadas.

A primeira escolha determinante a nível técnico foi a implementação em **JavaFX**, onde a pouca maturidade da plataforma foi determinante em diversos aspectos. Apesar de já ter algum tempo no mercado ainda apresenta falhas e incoerências ao nível da implementação e principalmente contém classes e métodos descritos em documentos oficiais, que não se encontram implementados ao nível da *Framework*. Este aspecto é crítico e gerou problemas ao nível da implementação, pois algumas destas lacunas só foram detectadas no momento da implementação. A ausência de funcionalidades descritas nos manuais como existentes, obrigaram ao desenvolvimento de mecanismos que foram adoptados como soluções de recurso.

Para além destas falhas o **JavaFX** possui uma *Framework* que ainda está bastante incompleta e pouco adaptada ao trabalho colaborativo. Este aspecto, não tão inesperado quanto o anterior, tinha sido pesado na escolha da plataforma e foi compensado pela sua forte compatibilidade com **JavaSE**. Assim, o **JavaSE** foi utilizado como segunda plataforma para colmatar as falhas existentes em **JavaFX**. **JavaSE** foi principalmente utilizado na

implementação do modo de funcionamento colaborativo e nas operações relativas à gestão dos conteúdos.

Ao nível da implementação é ainda relevante focar o impacto da utilização da linguagem **C** na aplicação. Como foi mencionado anteriormente, as anotações dinâmicas levantam um conjunto de problemas complexos de processamento que tornam impossível a sua implementação em **JavaFX**. Concluiu-se por isso que, para as tarefas envolvendo processamento de vídeo, seria necessário recorrer a uma linguagem suficientemente ágil e rápida para executar esta tarefa. A escolha da linguagem **C** revelou-se uma boa opção, no entanto há a considerar o impacto que representa ao nível do desenvolvimento, a utilização de linguagens de programação distintas.

Para além das anotações textuais e rectângulos, que estão integralmente implementadas, existem dois tipos de Anotações que apesar de terem sido descritas ao nível dos metadados, não se encontram implementadas na interface. As Elipses e as Linhas, podem ser interessantes particularmente ao utilizar quadros interactivos. A não implementação destas anotações ao nível da interface não representa uma limitação das funcionalidades do protótipo pois os tipos de anotações já criadas permitem ilustrar a forma como estas seriam utilizadas.

No que diz respeito ao funcionamento em modo colaborativo, uma referência à abordagem seguida e à proposta inicialmente ponderada de adoptar comunicação ponto a ponto. A ideia inicial, integrava dois mecanismos de disseminação de conteúdos: cliente-servidor cujo objectivo se centrava nas interacções entre o professor e os alunos e uma comunicação ponto a ponto para troca de conteúdos entre utilizadores. No entanto, o facto das aulas decorrerem em contextos de baixa participação associado a um reduzido número de utilizadores em simultâneo, levou a que a implementação tivesse apenas por base uma abordagem cliente-servidor.

A disseminação de conteúdos do professor para os alunos, tem por base um servidor HTTP, que suporta a partilha de conteúdos de uma forma centralizada.

Para terminar, uma referência ao menu de ajuda, ou mais concretamente à ausência deste. A utilização deste sistema num contexto real, requer a existência deste tipo de ajuda contextual que informa acerca de tarefas que são menos intuitivas no contexto da aplicação, nomeadamente no que concerne à utilização da aplicação em modo colaborativo. Não sendo relevante para o protótipo é um aspecto fundamental para a utilização do sistema de uma



forma mais alargada e por utilizadores com menos experiência neste tipo de aplicações, pelo que a sua implementação não pode ser ignorada.

Este e outros aspectos relacionados com o alargamento das funcionalidades do sistema são abordados na secção seguinte, como trabalho futuro.

## **4.2 Trabalho Futuro**

O trabalho desenvolvido originou um protótipo estável que pode ser testado em condições reais, no contexto de uma aula. No entanto, existem um conjunto de aspectos que necessitam ser trabalhados para que este protótipo ganhe a robustez necessária e seja simples e intuitivo ao nível da utilização.

Um primeiro ponto a ser focado como trabalho futuro passa pela implementação ao nível da interface de mais tipos de anotações, nomeadamente os tipos Elipse e Linha. Como foi mencionado durante a dissertação, estes dois tipos de anotações já se encontram definidos ao nível dos metadados e, por consequência, será interessante poderem ser utilizados na interface, de forma a aumentarem o leque de anotações para os utilizadores.

Por outro lado existem outros tipos de anotações cuja implementação seria interessante e que podem potenciar significativamente a utilidade do sistema. As Anotações do tipo imagem, onde a anotação é uma imagem definida pelo utilizador, vêm permitir um novo conjunto de oportunidades onde a situação é ilustrada através de uma imagem. Apesar deste tipo de anotações se encontrar teoricamente idealizado, ainda não possui nenhuma definição concreta ao nível dos metadados.

Estes aspectos são na sua essência o complementar de elementos que não se encontram completamente desenvolvidos no protótipo, no entanto e face à experiência adquirida ao longo deste trabalho é interessante perspectivar outro tipo de desenvolvimentos que não estavam originalmente previstos.

Dentro das possíveis expansões da aplicação, uma das mais relevantes será a possibilidade do anotador trabalhar com vários tipos de ficheiros, sendo que o objectivo essencial seria estender o conceito de anotador para além do vídeo. Isto é, a aplicação actualmente desenvolvida suporta anotações sobre um ficheiro de vídeo, mas seria particularmente interessante estender este suporte a outro tipos de formato, como os ficheiros PowerPoint,

GIF Animados, ou PNG. Esta necessidade surge do facto de formatos como o PowerPoint serem frequentemente utilizados pelos utilizadores no meio educacional, e desta forma, a sua compatibilização com a aplicação seria uma mais-valia em todas as vertentes.

Finalmente, muito do trabalho futuro prende-se com aspectos relacionados com a usabilidade da interface e com a melhoria da experiência do utilizador. A este nível há em primeiro lugar que efectuar testes eficácia e usabilidade à interface, do professor, do aluno e da interface para o quadro interactivo. No entanto há que ter em consideração que esta aplicação funciona num contexto em que a interacção com os utilizadores é fundamental. Assim, os testes com utilizadores não se esgotam na avaliação da interface e é necessário avaliar o funcionamento do sistema num contexto real, ou seja em aula. Esta avaliação é essencial para a obtenção de resultados práticos sobre a forma como se processa a utilização do sistema, de que forma esta pode ser melhorada e eventualmente poderá permitir descobrir outras abordagens à sua utilização.

Em resumo, acreditamos que o trabalho apresentado nesta dissertação, que se encontra materializa na aplicação desenvolvida constitui uma proposta válida sobre a utilização de vídeo e de anotação de vídeo no contexto de uma aula. A estabilidade do protótipo não esconde a frustração que está sempre presente quando nos apercebemos que mais do que um fim, o que se atingiu foi um novo ponto de partida.

Esta secção define um caminho, uma visão do desenvolvimento futuro, no entanto irá ser a utilização do sistema e a resposta dos seus utilizadores que marcará o verdadeiro trabalho futuro.

## 5 Bibliografia

- [1] J. F. Jensen, *Interactivity: Tracing a new concept in media and communication studies*. Nordicom Review, 1998.
- [2] D. W. E. a. W. B. Painter, "The Use of an Interactive Whiteboard in promoting interactive teaching and learning," 2005.
- [3] GEPE, "Estudo de Diagnóstico: A modernização tecnológica do sistema de ensino em Portugal," 2007.
- [4] W. E. MACKAY, "Eva: an experimental video annotator for symbolic analysis of video," in *SIGCHI Bulletin*. ACM Press, 1989, pp. 68-71.
- [5] C. N. ., G. N. Costa M., "Annotations as multiple perspectives of video content.," in *In MULTIMEDIA '02: Proceedings of the 10th ACM international conference on Multimedia*, ACM Press, 2002, p. 283–286.
- [6] C. N. Cabral D, "Mobile and web tools for participative learning," in *IADIS International Conference APPLIED COMPUTING*, Salamanca, Spain, 2007.
- [7] C. D. Correia N, "Interfaces for Video Based Web Lectures," in *The 6th IEEE International Conference on Advanced Learning Technologies*, Kerkrade, The Netherlands, 2006.
- [8] M. E. Hodges, *Multimedia Computing : Case Studies from MIT Project Athena*. Addison-Wesley, 1993.
- [9] K. G. Rehatschek H., "Vizard - an innovative tool for video navigation, retrieval, annotation and editing.," in , Vienna, Austria, 2001.
- [10] C. L. J. R. S. B. L. Tseng, *Using MPEG-7 and MPEG-21 for Personalizing Video*. IEEE Multimedia, 2004.

- [11] G. D. Abowd, "Classroom 2000: an experiment with the instrumentation of a living educational environment.," *IBM Systems Journal*, vol. 38, pp. 508--530, 2000.
- [12] T. Needham, "ConferenceXP and Advanced Collaborative Scenarios," in *Proceedings of the International Symposium on Collaborative Technologies and Systems (CTS'06)*, 2006, p. 392.
- [13] D. Berque, "An evaluation of a broad deployment of DyKnow software to support note taking and interaction using pen-based computers," *Journal of Computing Sciences in Colleges*, vol. 21, no. 6, pp. 204-216, Junho.
- [14] R. Davis, "Magic Paper: Sketch-Understanding Research," *Computer*, vol. 40, no. 9, pp. 34-41, Aug. 2007.
- [15] M. Burton. (2008, Jul.) Waterboard - Interactive water. [Online]. <http://www.interactiondesignblog.com/2008/07/interactive-water/>
- [16] Blade Software. (2009, Jul.) NotateIt - Whiteboard Software for Tablets, desktop PCs and Interactive Whiteboards. [Online]. <http://www.notateit.com>
- [17] Mimio, *Mimio Studio: User's Guide for Windows*. Sanford, L.P., 2008.
- [18] J. D. T. K. G. Coulouris, *Distributed Systems - Concepts and Design*. Addison-Wesley, 2005.
- [19] B. Laurie and P. Laurie, *Apache: The Definitive Guide*, 3rd ed. O'Reilly Media, 2002.
- [20] M. Feoktistov. (2010, Jan.) Small HTTP Server. [Online]. <http://www.smallsrv.com/>
- [21] Berry College. (2010, Jun.) P2P Programming Framework - Python. [Online]. <http://cs.berry.edu/~nhamid/p2p/framework-python.html>
- [22] E. W. Felton. (2009, May) TinyP2P. [Online]. <http://www.exonsoft.com/~kochin/TinyP2P/tinyp2p.html>
- [23] J. L. Weaver, W. Gao, S. Chin, and D. Iverson, *Pro JavaFX Platform: Script, Desktop and Mobile RIA with Java Technology*, 1st ed. APRESS, 2009.
- [24] J. Bloch, *Effective Java*, 2nd ed. Prentice Hall PTR, 2008.

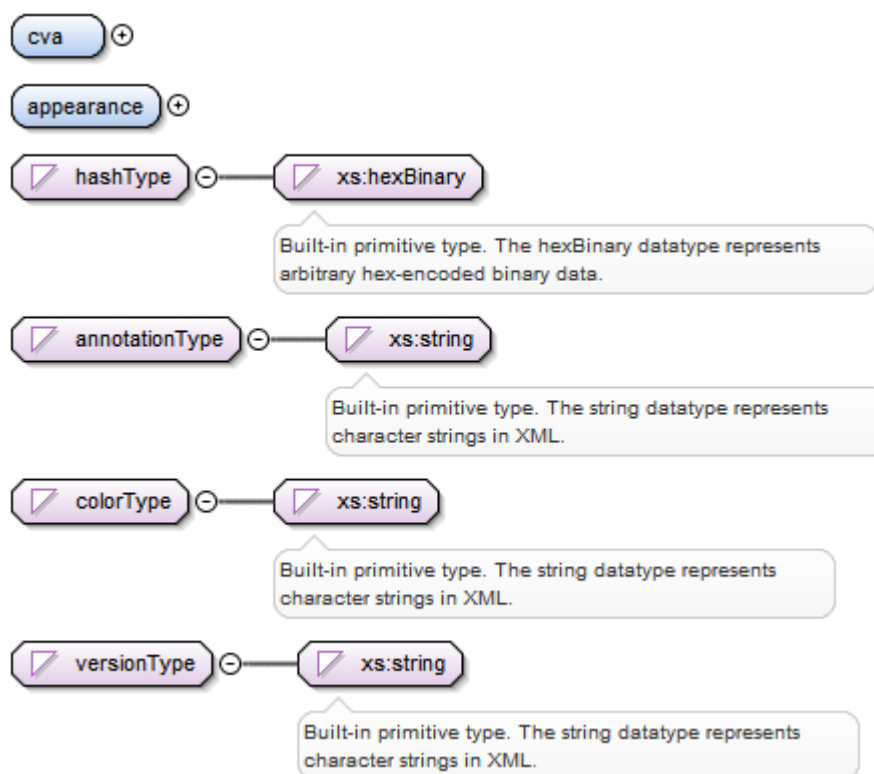
- [25] K. N. King, *C Programming: A Modern Approach*, 2nd ed. W. W. Norton & Company, 2008.
- [26] G. Bradski and A. Kaehler, *Learning OpenCV*, 1st ed. O'Reilly Media, 2008.
- [27] A. Terrazas, *Java Media APIs: Cross-Platform Imaging, Media and Visualization*. Sams, 2002.
- [28] B. D. Lucas and T. Kanade, "An iterative image registration technique with an application to stereo vision," in *Proceedings of the 1981 DARPA Imaging Understanding Workshop*, 1981, pp. 121-130.
- [29] R. Gordon, *Essential JNI: Java Native Interface*. Prentice Hall PTR, 1998.

## 6 Anexos

O capítulo de anexos contempla uma explicação mais detalhada de alguns pontos referidos na dissertação, em particular a estrutura dos metadados que descrevem as anotações utilizadas na aplicação.

### 6.1 Tipos de Dados Próprios

Este diagrama representa os elementos internos ao ficheiro de anotações, isto é, mostra a raiz do ficheiro, a raiz de todos os nós que descrevem a aparência dos objectos anotação (Anexo 6.3) e os tipos de dados complexos utilizados na descrição da informação.





### 6.3 Tipos de Aparência

Este diagrama pretende mostrar os diversos tipos de aparência visual que uma anotação pode assumir. Pretende-se assim focar neste nó da estrutura XML os diferentes atributos que cada anotação tem inerente à sua composição visual.

