



INTEGRAÇÃO E ADAPTAÇÃO DO SOFTWARE EASY2REC COM A PLATAFORMA EDUCAST@FCCN DA FCCN

DIOGO DA SILVA PALHAIS

Outubro de 2016

INTEGRAÇÃO E ADAPTAÇÃO DO SOFTWARE EASY2REC COM A PLATAFORMA EDUCAST@FCCN DA FCCN

Diogo da Silva Palhais

**Dissertação para obtenção do Grau de Mestre em
Engenharia Informática, Área de Especialização em
Sistemas Gráficos e Multimédia**

Orientador: Prof. Doutor António Vieira de Castro

Júri:

Presidente:

[Nome do Presidente, Categoria, Escola]

Vogais:

[Nome do Vogal1, Categoria, Escola]

[Nome do Vogal2, Categoria, Escola] (até 4 vogais)

Porto, outubro, 2016

Resumo

Nos últimos anos temos evidenciado uma enorme alteração nos comportamentos de ensino/aprendizagem que cada vez mais, fruto das constantes evoluções tecnológicas e potencial de partilha pela *Internet*, destacando-se o uso de recursos digitais de aprendizagem com especial destaque para o vídeo educativo (vulgarmente conhecido como videoaula). Ao longo desta dissertação foi analisada e implementada a integração e adaptação de duas soluções que atuam nesta área e que de certa forma se completam.

A primeira solução denomina-se por Easy2rec¹, um *software* para o sistema operativo *Windows* que permite a gravação de vídeo aulas captando vários sinais de vídeo e áudio de forma síncrona.

A segunda solução é a plataforma Educast², um repositório de armazenamento, processamento de vídeos e edição de conteúdos *online*. O Educast está disponível para toda a comunidade servida pelas instituições ligadas à RCTS³ (Rede Ciência, Tecnologia e Sociedade) nomeadamente as instituições de ensino superior portuguesas.

A integração destas duas soluções é uma enorme mais-valia para ambos. Foi analisado e tratado todo o processo de integração dos projetos e também os procedimentos necessários para que os utilizadores possam tirar usufruto do Easy2rec como ferramenta para produzir conteúdos e submetê-los na plataforma para a sua distribuição. Na presente dissertação é também realizado o estudo das soluções já existentes para o problema identificado, de que forma o solucionam, apresentando a nossa proposta e a sua implementação.

A aplicação implementada foi disponibilizada para a realização de testes por parte da equipa técnica do projeto parceiro Educast em duas fases distintas denominadas por fase *alfa* e *beta*. Após o *feedback* recebido de modo iterativo foi possível corrigir os erros detetados de forma a melhorar a solução e a alcançar a versão estável desejada para a posterior disponibilização a toda a comunidade.

Na fase final do projeto a aplicação implementada foi aceite pelo parceiro e encontra-se em ambiente de produção. O *feedback* obtido do parceiro foi muito positivo pelo que nos leva a querer melhorar a solução em trabalho futuro e alargar horizontes integrando a solução a outros parceiros.

Palavras-chave: Easy2rec, Educast, Videoaula, *E-Learning*, *Lecture Recording*, *Windows*

¹ Mais informações disponíveis em <http://easy2rec.com>

² Mais informações disponíveis em <http://educast.fccn.pt>

³ Mais informações disponíveis em <https://www.fccn.pt/pt/rede-academica/a-rede-ciencia-tecnologia-e-sociedade-rcts/>

Abstract

The recent years have shown a dramatic change in teaching/learning behavior that increasingly, the result of constant technological developments and potential of sharing in the Internet, especially the use of digital learning resources with special emphasis on the educational video (commonly known as video lecture). Throughout this dissertation was analyzed and implemented the integration and adaptation of two solutions that work in this area and complement each other.

The first solution is called Easy2rec, and is a software for the Windows operating system that allows to capturing video lessons by recording multiple video and audio signals synchronously.

The second solution is the Educast platform, a storage repository, video processing and editing of online content. The Educast is available to the entire community served by the institutions connected to RCTS (Science Network, Technology and Society) including the Portuguese higher education institutions.

The integration of these two solutions is a huge added value for both. It was analyzed and treated the whole process of integration projects and also the necessary procedures so that users can take use of Easy2rec as a tool to produce content and submit them to the platform for distribution. In this dissertation is also carried out the study of existing solutions to the problem identified, how them solve the problem, also presenting our proposal and its implementation.

The implemented application was made available for testing by the technical team Educast partner project in two phases called for alpha and beta. After the feedback received iteratively was possible to correct the errors detected in order to improve the solution and achieve the desired stable version for later delivery to the community.

In the final phase of the implemented application was accepted by the partner and it is in the production environment. The partner feedback was very positive so it leads us to want to improve the solution in future work and broaden horizons by integrating the solution to other partners.

Keywords: Easy2rec, Educast, Videoaula, *E-Learning*, *Lecture Recording*, *Windows*

Agradecimentos

Eu gostaria de mostrar a minha gratidão às pessoas que de alguma forma me apoiaram durante este trabalho e contribuíram para o seu sucesso.

As minhas primeiras palavras de apreço dirigem-se ao meu orientador, o Prof. Doutor António Castro, pelo apoio e disponibilidade que sempre demonstrou como pelos valiosos conselhos que me foi dando ao longo de todo o percurso. A sua prontidão e atenção contínua contribuíram indiscutivelmente para um melhor trabalho.

Ao meu colega de trabalho, Engenheiro Pedro Carvalho, por me incentivar tantas vezes e por todo o apoio que me proporcionou, mesmo nos momentos mais difíceis.

Aos meus pais e família por todo o apoio e sacrifícios que fizeram e fazem por mim que possibilitaram que tudo isto se tornasse realidade.

À minha namorada, Sara Gaspar, pela sua companhia e apoio com o seu toque característico ao longo de todo o projeto.

Ao Engenheiro Paulo Cunha por ter dado o seu voto de confiança ao projeto e ter acreditado no meu trabalho, pelo seu apoio e disponibilidade que permitiu a continuação do desenvolvimento do Easy2rec e que possibilitou transformar o projeto académico num negócio.

Ao Prof. Doutor João Rocha por acreditar no projeto e por todo o seu apoio e disponibilização dos meios necessários para a continuação do seu desenvolvimento. Sem o seu apoio teria sido muito mais difícil alcançar os objetivos.

Ao Instituto Superior de Engenharia do Porto e a todos que de alguma forma direta ou indiretamente contribuíram para o desenvolvimento do projeto.

À Fundação para a Computação Científica Nacional por acreditar no projeto e no seu potencial percorrendo connosco este caminho. Agradecer em especial ao Engenheiro João Gomes diretor da área de serviços avançados e ao Engenheiro Rui Ribeiro gestor do serviço técnico de vídeo, pela sua enorme disponibilidade, abertura e colaboração em todos os momentos.

A todos eles, o meu mais sincero obrigado.

Índice

1	Introdução	1
1.1	Contexto	1
1.1.1	O Vídeo na Educação.....	2
1.1.2	O Easy2rec.....	3
1.1.3	O Educast	5
1.2	O Problema	6
1.3	Análise de Valor	7
1.4	Abordagens e Objetivos	7
1.5	Estrutura do Documento.....	8
2	Contexto e Estado da Arte	9
2.1	Sistemas de <i>Lecture Recording</i>	9
2.2	Abordagem ao Problema	9
2.2.1	Os Conceitos de Negócio	10
2.2.2	Os Intervenientes.....	11
2.2.3	O Modelo de Casos de Uso	12
2.2.4	Diagrama de Sequencia de Sistema.....	13
2.2.5	Modelo de domínio	14
2.2.6	Os Requisitos Não Funcionais.....	17
2.2.7	As Restrições	18
2.3	Análise de Valor	19
2.4	Estado da Arte em Soluções	21
2.4.1	Capturer Lecture Video Recording System	22
2.4.2	SWITCHcast Video Management System	22
2.4.3	Videotorium	22
2.4.4	L2L (Live to e-Learning)	22
2.4.5	ViPS	23
2.4.6	Opencast.....	23
2.5	Análise de Tecnologias.....	23
2.5.1	Simple Object Access Protocol (SOAP)	23
2.5.2	Web Services Description Language (<i>WSDL</i>).....	24
2.5.3	Universal Description Discovery and Integration (<i>UDDI</i>).....	24

2.5.4	ASP.NET	25
2.5.5	SQL SERVER	25
2.5.6	FRAMEWORK .NET	25
2.5.7	C Sharp (C#)	26
2.5.8	FFMPEG.....	26
2.5.9	Extensible Markup Language (XML)	26
2.5.10	JavaScript Object Notation (JSON)	27
3	Avaliação das Soluções Existentes	29
4	Design.....	35
4.1	Design Conceptual.....	35
4.2	Design Arquitetural.....	36
4.2.1	O Serviço de Licenciamento	36
4.2.2	A Aplicação Easy2rec.....	36
4.3	Design Detalhado	37
4.3.1	O Serviço de Licenciamento	37
4.3.2	A Notificação de Licença	39
4.3.3	A Submissão de Ficheiros.....	40
4.4	A Base de Dados	41
4.5	A Interface Gráfica	42
5	Implementação	47
5.1	A Base de Dados	47
5.2	O Serviço de Licenciamento	48
5.2.1	A Camada de Acesso a Dados	49
5.2.2	O Web Service.....	51
5.2.3	Os Códigos de Resposta.....	53
5.2.4	A Autenticação e Segurança	53
5.3	A Aplicação Easy2rec.....	54
5.3.1	A Autenticação.....	54
5.3.2	A Gravação	58
5.3.3	A Submissão	62
5.3.4	Suporte Multilíngue	65
5.3.5	Incorporação de <i>Interface</i> Externa	65
5.3.6	A Notificação à Aplicação Easy2rec.....	66

5.4	Dados de Utilizador	67
5.4.1	A Estrutura de Ficheiros	67
5.5	Testes de Integração	68
6	Avaliação da Solução	69
6.1	As Grandezas de Teste	70
6.2	As Hipóteses a Testar	70
6.2.1	Hipótese 1	70
6.2.2	Hipótese 2	70
6.3	A Metodologia de Avaliação	71
6.4	Os Resultados Obtidos	72
6.5	Análise de Resultados Obtidos	75
6.5.1	A Versão Alfa	77
6.5.2	A Versão Beta	77
6.6	Conclusão da Avaliação	78
7	Conclusão	79
7.1	Considerações gerais	79
7.2	Trabalhos Futuros	80

Lista de Figuras

Figura 1 – O ambiente de gravação Easy2rec.....	4
Figura 2 – A plataforma Educast	6
Figura 3 - Diagrama de Casos de Uso identificados.....	12
Figura 4 - Diagrama de Sequencia de Sistema da Aplicação Easy2rec	14
Figura 5 - Diagrama de Sequencia de Sistema do <i>Serviço de Licenciamento</i> Easy2rec	14
Figura 6 - Diagrama de modelo de domínio: Aplicação	16
Figura 7 - Diagrama de Modelo de Domínio: Serviço de Licenciamento Easy2rec	17
Figura 8 - Modelo de Negócio <i>Canvas</i>	21
Figura 9 – Exemplo de pedido <i>SOAP</i>	24
Figura 10 – Exemplo de representação de informação em <i>JSON</i>	27
Figura 11 - <i>Workflow</i> de sistema de gravação de vídeo-aulas	30
Figura 12 - Diagrama conceptual de componentes.....	36
Figura 13 - Diagrama de sequência das comunicações entre componentes	39
Figura 14 - Diagrama de sequência das comunicações entre componentes	40
Figura 15 – Descrição do ficheiro estruturado metadata.xml.....	40
Figura 16 - Diagrama de Base de Dados	41
Figura 17 - Janela inicial de Login	42
Figura 18 - Escolha de Parceiro de Login.....	43
Figura 19 - Interface com o sistema de autenticação externo do parceiro.....	43
Figura 20 - Confirmação de login e licença de utilizador válida	44
Figura 21 - Janela de Menu Principal	44
Figura 22 - Lista de gravações realizadas	45
Figura 23 - Formulário externo de submissão de metadados.....	45
Figura 24 - Janela de processamento e <i>upload</i> dos vídeos	46
Figura 25 – SQL de criação da tabela <i>utilizador_partner</i>	48
Figura 26 – Estrutura de componentes no serviço de licenciamento	49
Figura 27 – Estrutura de componentes e classes de acesso a dados	50
Figura 28 – Diagrama de classes de camada de acesso a dados	51
Figura 29 – Estrutura de classes e métodos presentes no <i>Web Service</i>	52

Figura 30 – Exemplo de verificação do parâmetro <i>validate</i>	53
Figura 31 – Exemplo de criação do parâmetro <i>validate</i>	54
Figura 32 – Janela inicial para realizar o login	55
Figura 33 – Janela de seleção do parceiro para login	55
Figura 34 – Janela de solicitação de licença ao parceiro	56
Figura 35 – Informações de autenticação no serviço do parceiro	56
Figura 36 – Ambiente de autenticação federada controlada pelo parceiro	57
Figura 37 – Mensagem de autenticação e licença válida na aplicação	57
Figura 38 – Diagrama de sequência de sistema do processo de autenticação	58
Figura 39 – Janela de menu principal quando autenticado por parceiro	59
Figura 40 – Ambiente de gravação do Easy2rec	59
Figura 41 – Mensagem de gravação realizada com sucesso	60
Figura 42 – Lista de gravações realizadas pelo utilizador	60
Figura 43 – Ficheiro estruturado de metadados sobre as gravações	61
Figura 44 – Janela de metadados de submissão de vídeo para o parceiro	62
Figura 45 – Exemplo do ficheiro de metadados de uma submissão	63
Figura 46 – Janela de submissão de gravação para o parceiro realizada com sucesso	64
Figura 47 – Diagrama de sequência do processo de submissão via <i>FTP</i>	65
Figura 48 – Exemplo código <i>Javascript</i> para comunicação entre parceiro e aplicação	66
Figura 49 – Estrutura de ficheiros da aplicação Easy2rec	67

Lista de Tabelas

Tabela 1 - Conceitos de Domínio: Aplicação Easy2rec	15
Tabela 2 - Conceitos de Domínio: Serviço de Licenciamento	16
Tabela 3 - Comparação entre sistemas de lecture recording.....	30
Tabela 4 - Comparação entre soluções para a plataforma <i>Windows</i>	33
Tabela 5 - Parâmetros do método <i>PartnerLicencerRequest</i>	38
Tabela 6 - Parâmetros do método <i>GetLicencesCreated</i>	38
Tabela 7 - Parâmetros do método <i>SetPartnerPrivatekey</i>	38
Tabela 8 - Parâmetros do método <i>GetLicenceData</i>	39
Tabela 9 – Códigos e descrições de resposta a pedidos ao <i>Web Service</i>	53
Tabela 10 - Exemplo de estrutura do relatório de avaliação e preenchimento	72
Tabela 11 – Resultados obtidos versão <i>alfa</i>	73
Tabela 12 – Resultados obtidos versão <i>beta</i>	74
Tabela 13 – Resultados obtidos ordenados por gravidade na fase <i>alfa</i>	75
Tabela 14 – Análise dos resultados obtidos por frequência de gravidade	76

Acrónimos

API	<i>Application Programming Interface</i>
DNS	<i>Domain Name System</i>
FCCN	Fundação para Computação e Ciência Nacional
FCT	Fundação para a Ciência e a Tecnologia
IES	Instituições de Ensino Superior
ISEP	Instituto Superior de Engenharia do Porto
JSON	<i>JavaScript Object Notation</i>
LMS	<i>Learning management system</i>
MD5	<i>Message-Digest algorithm 5</i> – Algoritmo de encriptação
MOOC	<i>Massive open online course</i>
MOODLE	Ambiente de Aprendizagem Modular Orientado ao Objeto
PESTI	Projeto/Estágio da Licenciatura em Engenharia Informática
RCTS	Rede de Ciência, Tecnologia e Sociedade
SOAP	<i>Simple Object Access Protocol</i>
SQL	<i>Structured Query Language</i>
TIC	Tecnologias da informação e comunicação
UML	Unified Modeling Language
WEB	<i>World Wide Web</i>
WSDL	<i>Web Services Description Language</i>
XML	<i>Extensible Markup Language</i>

1 Introdução

No presente capítulo apresenta-se o contexto da presente dissertação, como também o problema identificado e os projetos envolvidos neste. Os objetivos pretendidos com a realização deste trabalho irão ser detalhados passando pela análise de valor do projeto e de que forma este foi implementado. Finalmente foi descrita uma breve reflexão sobre os resultados alcançados. Termina-se o capítulo com a organização do presente documento.

1.1 Contexto

Os últimos anos têm sido profícuos em novas tecnologias e meios para promover a partilha de conhecimento e auxiliar o processo de ensino-aprendizagem. O acesso à informação e especificamente à formação ficou mais simples com o acesso à mesma via *Web*. Existem inúmeros mecanismos e ferramentas para produzir conteúdos de ensino para apoiar a educação.

A *Internet* constitui atualmente a base tecnológica da forma organizacional que caracteriza a Era da Informação: a rede (Castells, 2007). As tecnologias da informação e comunicação são definidas por Gibson como “formas de fluxo de informação digitalizada, nas quais os dados, sejam texto, som ou imagens em tempo real, são comprimidos em séries de 0 e 1, e transmitidos via ondas, cabo ou redes” (Gibson e Ward, 2000).

O aparecimento da *Web 2.0* veio alterar a forma de como a *Internet* era consumida e utilizada. Num âmbito designado por *Web 1.0* caracterizado pela elaboração de conteúdos por parte de especialistas onde os utilizadores tinham apenas um papel passivo para um âmbito totalmente diferente onde os utilizadores passaram a ter um papel ativo e a poderem criar os mais variados tipos de conteúdos e comunicar entre si. O foco da denominada *Web 2.0* transfere o *focus* da tecnologia para as pessoas e nos serviços que as empresas destas pessoas utilizam (Siony da Silva, 2010). Segundo Gomes (2009) “a adoção crescente de práticas de *e-learning* em complementaridade ao ensino presencial coloca novos desafios e oferece novas alternativas às práticas de avaliação de aprendizagens habitualmente utilizadas”. Os

produtores destes conteúdos, poderão ser mais audazes e torná-los mais motivadores, completos, eficazes e interativos, pautando-se pela originalidade e qualidade dos mesmos. Poderão proporcionar mecanismos de ensino paralelo e mesmo criar novos modelos de ensino que levem a aprender matérias inerentes a conhecimentos específicos.

Os ambientes virtuais de aprendizagem podem ser definidos como sistemas de ensino e aprendizagem integrados e abrangentes capazes de promover o desenvolvimento do aluno. Eles possibilitam criar situações de ensino e aprendizagem nas quais os alunos organizam e definem o próprio ritmo do seu estudo. O próprio estudo não é iniciado e dirigido por eventos expositivos e recetivos ritualizados, mas, sim, por meio de discussão e interação (princípio do estudo por meio de comunicação e interação) (Peters, 2004).

O Moodle⁴ (*Modular Object-Oriented Dynamic Learning*) é uma dessas ferramentas que surgiu neste seguimento e permite a gestão de cursos à distância, desenhado para ajudar educadores a criar, com facilidade, cursos on-line de qualidade. Ferramentas como o Moodle também podem ser chamadas de LMS (*Learning Management Systems*), que são sistemas de gestão de ensino-aprendizagem ou ambientes virtuais de ensino e aprendizagem. O surgimento da internet em particular, que tem vindo a alterar a forma como, nomeadamente professores e alunos colaboram, partilham e interagem em meio académico. Caracteriza-se principalmente por agilizar os fenómenos comunicacionais através de meios digitais e da comunicação através de redes informáticas, normalmente mediadas por computadores no processo de captação, transmissão e distribuição de informação.

Imagine-se agora um produto que possa simular de forma fiel uma aula em ambiente presencial, onde o docente se apresenta perante os alunos com os seus recursos educativos digitais. Um produto que de uma forma facilitada possa auxiliar os docentes a criarem conteúdo para facilitar o processo de ensino-aprendizagem com os seus alunos. Este produto é capaz de captar e representar o orador, a voz do orador e os conteúdos que este apresenta aos seus alunos, tudo isto em simultâneo. Aliando estas funcionalidades a um *layout* simples, intuitivo e prático para que qualquer pessoa, com qualquer nível de conhecimento informático consiga tirar partido desta solução e criar os seus conteúdos. Foi sobre este pressuposto que foi desenvolvido e criado o Easy2rec.

1.1.1 O Vídeo na Educação

O ensino e aprendizagem através de publicação de vídeo aulas de forma organizada é uma nova tendência global. Grandes universidades mundiais como por exemplo o MIT (*Massachusetts Institute of Technology*) lançaram já o seu portal de cursos online⁵. Segundo a publicação *International Trends in Higher Education 2015* esta nova tendência promove outras formas e métodos de ensino e aprendizagem como *flipped classroom* e *blended learning*.

⁴ Mais informação disponível em <http://moodle.org>

⁵ Mais informações disponíveis em <https://ocw.mit.edu/index.htm>

Estes métodos promovem a inversão do paradigma de ensino tradicional onde a aula presencial é usada para adquirir conhecimentos e a componente prática é realizada extra-aula, passando para a partilha de conhecimento antecipadamente através de recursos online como vídeos onde os alunos adquirem os conhecimentos e a aula neste caso é usada para a prática e aplicação dos conhecimentos. Estes novos métodos que vão surgindo têm como objetivo fomentar um papel mais ativo dos alunos e promover a discussão de ideias (ITHE, n.d.).

Segundo Pappas vídeo aulas podem gerar uma sensação de que o professor irá perder a interatividade com os alunos e gerar um entrave numa fase inicial, no entanto as vídeo aulas podem também promover a discussão de ideias e matérias, convidar os alunos a analisar e pensar, ser vistos e revistos quantas vezes foram necessárias, conter matérias subdivididas explicadas mais detalhadamente tornando-se mais fáceis de absorver pelos alunos, incluir as pessoas nas tarefas e assuntos discutidos, como também permite cada aluno aprender ao seu ritmo. De acordo com a *ComScore* em 2014 o número de utilizadores de dispositivos móveis ultrapassou o número de utilizadores de computadores pela primeira vez. Perante este facto enfrentamos uma enorme mudança no paradigma de visualização de conteúdo pois a partir deste momento o local de consumo de conteúdos por defeito é móvel. E que cada vez mais os dispositivos têm são mais poderosos e inteligentes (Pappas, 2013).

Micro-Learning é uma tendência a crescer em popularidade que consiste em focar e minimizar os recursos educativos para uma só temática ou tarefa muito específica. Desta forma a tendência é reduzir o tempo de vídeo para o mínimo possível e de uma forma muito objetiva definir um só objetivo bem claro para que os alunos percebem claramente a informação que se pretende transmitir. Segundo Eades 70% dos *millennials* (pessoas nascidas entre 1977 e 2000) visitam o *Youtube* mensalmente e preferem o vídeo a outro tipo de multimédia. Um vídeo seguindo o padrão de *micro-learning* deve ter no máximo quatro minutos ou menos. (Eades, 2014). O vídeo na internet tem cada vez mais relevância e a educação deve tirar partido disso mesmo para melhorar o processo de ensino aprendizagem e obter melhores resultados.

1.1.2 O Easy2rec

Atualmente a comunicação faz-se cada vez mais através de conteúdos multimédia para a *Web*. Numa rede como o *YouTube*⁶, que têm mais de mil milhões de utilizadores, o que equivale a quase um terço do total de utilizadores da *Internet* e são vistos diariamente centenas de milhões de horas de vídeo (*Youtube Statistics*, 2016). A *Internet* com especial destaque para as redes sociais, está a alterar a forma como comunicamos, apresentamos ideias, lançamos campanhas, fazemos publicidade ou promovemos projetos e organizações.

Ao nível da educação e formação, cada vez mais se assiste à criação e partilha de conteúdos multimédia, seja para apoiar dinâmicas de ensino à distância, *Mooc's* e para sustentar

⁶ Disponível em <http://youtube.com>

repositórios de conteúdos como o *Moodle*. Deste modo a disponibilização de ferramentas intuitivas que permitam inovar e facilitar a criação de conteúdos para atrair a atenção de vários públicos é uma necessidade constante do mercado.

O Easy2rec é um *software* desenvolvido para a plataforma *Windows* de produção de vídeos para criação de apresentações. O desenvolvimento do Easy2rec iniciou-se no âmbito da disciplina PESTI (Projeto/Estágio da Licenciatura em Engenharia Informática) no Instituto Superior de Engenharia do Porto. Nesta fase o projeto era identificado sobre um nome diferente, Edugraal, que mais tarde viria a dar o nome a empresa que iria explorar o produto. O *software* Easy2rec permite capturar simultaneamente e de forma síncrona vídeo e áudio em qualquer computador e diversos tipos de conteúdos pré-selecionados tais com apresentações *powerpoint*, *pdf's*, vídeos, áudios, páginas *Web*, imagens, e até capturar a área de trabalho, juntando num só vídeo com aparência personalizável e de uma forma simples e rápida.

O produto pretende minimizar o tempo, custo e esforço na produção de vídeos, fazendo a própria pessoa no seu computador mesmo que não seja especialista em informática. O objetivo é facilitar a compreensão dos espectadores que, deste modo, têm acesso a vários materiais de uma só vez e a tarefa dos apresentadores que com menos custos e em menos tempo conseguem criar um só vídeo com todos os elementos pretendidos.

O produto é atualmente explorado pela empresa Edugraal, Innovative Solutions Lda, a primeira empresa *startup* nascida no Instituto Superior de Engenharia do Porto criada pelo mestrando em conjunto com o orientador Dr. António Castro e o Engenheiro Paulo Cunha no papel de investidor e mentor do projeto. O *software* está disponível em <http://easy2rec.com> que qualquer pessoa pode experimentar mediante um registo prévio.

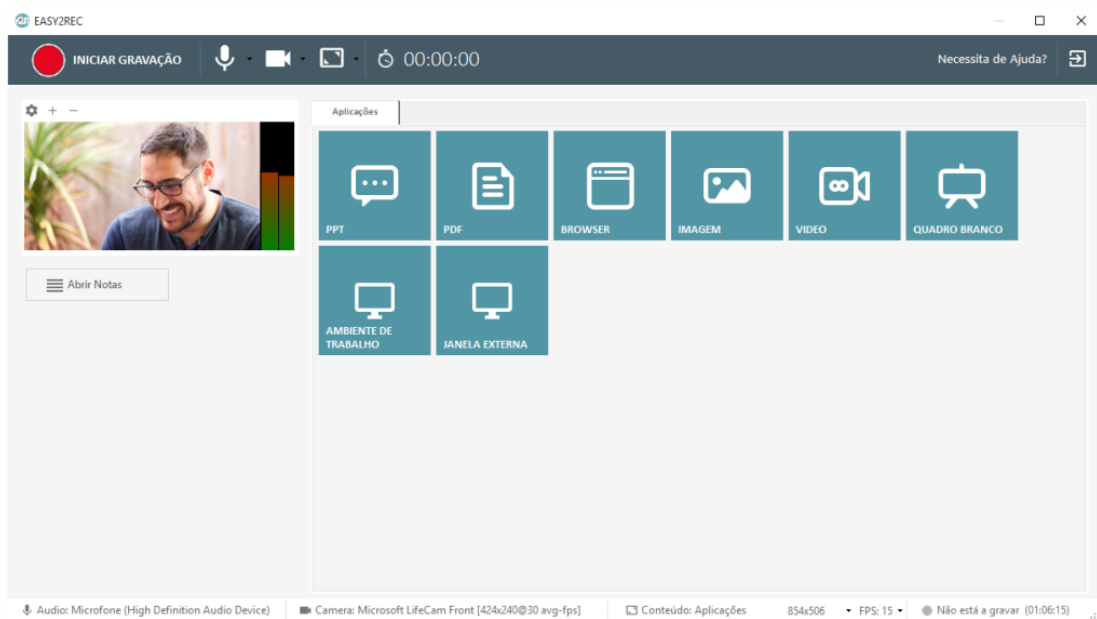


Figura 1 – O ambiente de gravação Easy2rec

1.1.3 O Educast

O Educast⁷ é uma plataforma *online* de gestão de vídeos para registo e distribuição de conteúdos letivos que se destina à comunidade servida pelas instituições ligadas à RCTS. Todos os conteúdos produzidos ficam acessíveis, mediante acesso controlado. O Educast combina funcionalidades de arquivo, armazenamento, gestão e administração de vídeo, pesquisa, processamento e transcodificação em múltiplos formatos, *upload* e edição de vídeo *online*, associação de metadados, segmentação dos conteúdos por canais temáticos, personalização do *branding* dos conteúdos criados e facilidade de integração em ambientes web para sua posterior disponibilização aos alunos. Os conteúdos publicados *online* têm um cariz educativo, didático, lúdico ou cultural e são utilizados em ambiente académico e de investigação, como forma de disseminação de conhecimento (FCCN, 2016).

“O Educast é um serviço de gestão de vídeo para registo e distribuição simples de conteúdos letivos (aulas). O sistema central garante o pós-processamento necessário para que estes fiquem disponíveis aos alunos através da Internet. Este serviço resulta de uma parceria entre a FCCN|FCT e a sua homóloga Suíça, Switch. O Educast surge depois de 5 anos de desenvolvimento da Switch na criação de uma solução completa focada na gravação de aulas e eventos de teor cultural e didático. Esta solução é composta por um repositório central de armazenamento e processamento de vídeos, plataforma de edição de conteúdos online e o software de gravação (Educast Recorder). Esta plataforma permite: gravação de aulas; upload para o servidor central; edição e sua posterior publicação; integração em sistemas de e-learning; portais e websites das Instituições de Ensino Superior. Os vídeos finais ficam disponíveis online ou por download para computadores e dispositivos móveis em três tipos de formatos: Flash, Quicktime e iPod. Após um período de testes de um ano na plataforma com resultados satisfatórios, a FCCN|FCT conseguiu criar uma parceria com a Switch que permitiu trazer toda esta plataforma (Switchcast) e o seu código fonte para Portugal. A instalação e adaptação da plataforma à realidade nacional decorreu em Agosto de 2010, durante um período de seis meses foi ainda implementado um projeto-piloto com duas instituições, a Universidade de Lisboa e a Universidade do Porto. O serviço entrou em produção em Março de 2011, e encontra-se disponível para a todas as instituições pertencentes à RCTS (Rede Ciência, Tecnologia e Sociedade).” (FCCN, 2016)

A FCCN⁸ (Fundação para a Computação Científica Nacional) é uma entidade pública que fornece e gere as plataformas tecnológicas para a rede nacional de instituições do ensino superior em Portugal. É também responsável pela gestão de vários produtos e serviços dentro dos quais se encontra o Educast.

⁷ Mais informações disponíveis em <http://educast.fccn.pt>

⁸ Mais informações disponíveis em <http://fccn.pt>



Figura 2 – A plataforma Educast

1.2 O Problema

Consideramos que ambas as soluções se enquadram na mesma área dado que, a gravação da vídeo aula e a sua difusão e distribuição se interligam. Foram facilmente identificados pontos de convergência que podiam ser trabalhados de uma forma conjunta de modo a tornar o mecanismo de criação e distribuição desse tipo de recurso educativo digital mais eficaz.

O Educast sendo uma plataforma *online* de gestão, armazenamento e distribuição de conteúdo multimédia encontra-se dependente do seu software de gravação e da consequente submissão na plataforma. Atualmente é disponibilizado apenas para computadores baseados no sistema operativo *Mac OS*. O facto de se encontrar apenas disponível para esta plataforma reduz o número de possíveis utilizadores da plataforma e reduz também a aquisição de novos utilizadores pois é nos dias de hoje difícil justificar a compra de equipamento específico para este efeito com os orçamentos disponíveis para as instituições. A necessidade por parte da FCCN em colmatar, desde já, a compatibilidade do *software* de gravação para os utilizadores no serviço Educast que possuíam computadores baseados em sistemas *Windows* foi o ponto de partida para a realização do estudo de uma possível integração dos dois projetos.

O Easy2rec sendo um produto já desenvolvido para a plataforma em questão e que realiza as tarefas pretendidas abriu portas para uma possível parceria. Estando o Easy2rec no mercado necessita todos os dias de captar novos clientes e utilizadores. A integração seria uma enorme mais-valia não só pela exposição a um número elevado de novos utilizadores como também a possibilidade de as equipas técnicas presentes nas instituições de ensino apoiar os docentes e criadores de conteúdo no manuseamento do Easy2rec. Desta forma preconizasse um aumento exponencial de utilizadores na plataforma Educast através da adoção e integração do Easy2rec.

1.3 Análise de Valor

Na análise de valor a descrever numa fase mais avançada da presente dissertação serão abordadas várias questões tais como, o porque da necessidade de apresentar uma proposta de valor bem definida aos clientes, de modo que a mensagem seja transmitida de forma clara e desta forma aumentar a taxa de sucesso do produto ou serviço. Será abordado também os segmentos de mercado a focar, evidenciando as mais-valias para cada um deles. Identificados também os vários tipos de negócio que o produto ou serviço poderá optar e finalmente será apresentado o modelo de negócio sobre a forma de modelo *canvas* e formas de analisar e quantificar a criação e valor obtido pelo negócio realizado identificando as várias variáveis.

1.4 Abordagens e Objetivos

Tento em conta que o público-alvo da plataforma Educast engloba utilizadores com vários níveis de conhecimentos na área informática e que será uma adaptação a uma plataforma já existente, é necessária especial atenção à forma de como será implementada para que não seja um processo completamente novo que obrigue a todos um tempo de aprendizagem demasiado longo hipotecando as hipóteses de sucesso. Deste modo a abordagem adotada será a integração das novas funcionalidades identificadas no *software* Easy2rec já existente, mantendo o módulo de gravação intacto ou com ligeiras melhorias. As principais funcionalidades identificadas a desenvolver são:

- Login de utilizadores Educast;
- Validação da sessão de utilizador;
- Submissão de metadados e associação aos vídeos;
- Submissão de vídeos;
- Visualização de vídeos por submeter;
- Visualização de vídeos já submetidos.

O processo de *login* de utilizadores do parceiro Educast implicará a atribuição de uma licença de utilização que garante o acesso do utilizador ao Easy2rec por um determinado período de tempo finito a definir. A atribuição da licença será realizada através da implementação de um serviço que disponibilizará um serviço ao parceiro que deve ser consultada no momento do *login* do utilizador.

A submissão dos ficheiros de vídeo gerados pelo utilizador no Easy2rec deverá ser enviada para o servidor cumprindo a estrutura e os padrões acordados com o parceiro. As tecnologias a utilizar devem ir de encontro as tecnologias já utilizadas pelos projetos a integrar minimizando deste modo o custo de desenvolvimento facilitando a adaptação. Pretende-se cumprir todos os requisitos identificados colmatando o problema identificado com sucesso

minimizando o esforço de interação com o produto final. A satisfação do utilizador e do parceiro é um aspeto vital para o sucesso da solução.

1.5 Estrutura do Documento

A presente dissertação encontra-se dividida em sete capítulos. Nesta secção será descrito de forma sucinta a estrutura do presente documento.

No primeiro capítulo é abordado o contexto do trabalho, o problema identificado, a análise de valor do projeto e finalmente a abordagem ao problema e os objetivos. Todos eles abordados de uma forma resumida.

No segundo capítulo é apresentado o estado da arte, incluindo toda a descrição teórica subjacente a este trabalho, a análise de valor detalhada, os conceitos e regras de negócio que a implementação deverá respeitar, quais os intervenientes no sistema e finalmente são enumeradas algumas das restrições existentes à partida. Neste capítulo é apresentado mais detalhadamente alguns dos conceitos abordados também no primeiro capítulo.

O capítulo três apresenta o estudo das soluções e abordagens implementadas por várias NREN's europeias ao mesmo problema que nos propomos a resolver. São analisados os resultados obtidos do estudo apresentado e por último apresentadas as conclusões.

No quarto capítulo é abordada a proposta do sistema a implementar, apresentando todo o processo de *design* da solução, as ligações entre os vários componentes como também o modelo de dados associado.

No capítulo cinco é apresentado o mapeamento do *design* para a implementação da solução apresentado no capítulo anterior. É apresentado também a descrição da correspondente fase de testes ao longo da implementação.

No sexto capítulo são apresentadas as experiências realizadas e seus resultados, depois de definidas as hipóteses e os processos de teste a usar sobre a solução implementada. É descrito o grupo de inquiridos e as fases de teste.

Por fim, no capítulo sete são apresentadas as conclusões, limitações da proposta e descrito algum do trabalho futuro a desenvolver.

2 Contexto e Estado da Arte

Neste capítulo será abordado o contexto da presente dissertação de uma forma mais detalhada e aprofundada como também o estado da arte, não só das soluções já existentes como também das tecnologias relevantes para a solução do problema identificado. Neste capítulo também serão detalhados os conceitos de negócio, os intervenientes e as restrições existentes.

2.1 Sistemas de *Lecture Recording*

A gravação de palestras e aulas gravadas pode ser usado como uma maneira de apoiar um método de aprendizagem assíncrona na educação. Um método de aprendizagem assíncrona dá aos alunos uma melhor capacidade de aprender independentemente do tempo disponível e lugar onde se encontram, dando mais controle sobre o seu próprio tempo de aprendizagem. O aumento do uso de palestras de vídeo na educação e seu impacto sobre o processo de aprendizagem tem a tendência de aumentar cada vez mais devido aumento ao acesso e uso de tecnologia no dia-a-dia. Desta forma o conteúdo deve ser fácil de criar, atualizar e os vídeos neste caso devem ser precisos e fáceis de aceder. Para isso é necessário desenvolver sistemas que capacitem todos os intervenientes de o fazer de uma forma fácil e intuitiva. Para a generalidade dos sistemas relevantes seguem um determinado fluxo de atividades são elas o planeamento, a captação, a edição, a transcodificação e a publicação. O desenvolvimento de sistemas que aprimorem este processo separando as atividades por diferentes intervenientes é fundamental para o sucesso deste tipo de sistemas.

2.2 Abordagem ao Problema

Na presente secção será analisado o problema identificado no capítulo anterior de uma forma mais detalhada, onde serão definidos os conceitos de negócio a respeitar na implementação, quais os intervenientes nos processos quais os casos de uso que a solução deverá apresentar.

Serão apresentados também alguns diagramas de sequencia de sistema que explicitam de uma forma simples como os sistemas se interligam e se comunicam. Por fim são apresentados os requisitos não funcionais, o modelo de domínio e as restrições existentes á partida para o desenvolvimento do projeto.

2.2.1 Os Conceitos de Negócio

Pretende-se adaptar e integrar o *software* Easy2rec já existente compatibilizando-o com entidades parceiras nomeadamente o portal Educast. Esta adaptação deve permitir o início de sessão de utilizadores na aplicação através da sua conta de utilizador registada no parceiro, submeter vídeos para a plataforma parceira, associar metadados a cada vídeo submetido e a gestão dos vídeos submetidos e por submeter presentes no sistema local. Analisado o problema e os objetivos são identificados um conjunto de conceitos de negócio que representam restrições, condições e regras que a resolução do problema deve respeitar e refletir. São constituídos por afirmações que definem como o negócio deve se encontrar estruturado e como os processos devem estar relacionados. Os conceitos de negócio provêm tanto de fatores externos bem como internos para que sejam alcançados os objetivos.

- Unicamente utilizadores de instituições associadas à RCTS podem usar o Easy2rec como criador de conteúdos. Utilizadores necessitam de um login válido para iniciar sessão no Easy2rec;
- Todos os utilizadores necessitam de uma licença de utilização Easy2rec válida para iniciar sessão. A licença é fornecida através de um pedido ao serviço de licenciamento pelo parceiro mediante a realização da autenticação com sucesso;
- O atual contrato em vigor entre a Edugraal e a FCCN prevê um número máximo de 3600 licenças mensais a usar num período máximo de dois anos;
- Cada licença tem a duração de 30 dias corridos de calendário a partir do dia em que é gerada;
- Uma licença é única e exclusivamente atribuída a um utilizador;
- Uma licença atribuída ao utilizador pode ser usada em múltiplos equipamentos;
- A licença é disponibilizada aos utilizadores com um login válido após autorização do serviço Educast;
- Deverá ser assegurado a informações de estatística como quantas licenças foram geradas pelo parceiro;
- O Easy2rec deverá verificar se a licença permanece válida ou se já se encontra expirada;
- Em caso de a licença se encontrar expirada o utilizar deverá ser notificado e este encaminhado para proceder à renovação da mesma obtendo uma nova licença por mais 30 dias;

- No processo de autenticação a aplicação deverá encaminhar o utilizador para o mecanismo próprio de autenticação no serviço Educast. Este mecanismo deverá disponibilizar as informações necessárias para o utilizar criar ou renovar das licenças;
- Todos os pedidos efetuados ao serviço de licenciamento só deverão ser respondidos após assegurar que são válidos e provêm de um parceiro autorizado e credenciado para tal;
- Em cada submissão de vídeos para a plataforma do parceiro o utilizador deverá submeter também um formulário com um conjunto de metadados referentes ao vídeo a submeter;
- A estrutura do formulário a submeter é definida pelo parceiro e pode apresentar diferentes estruturas em diferentes ocasiões;
- Toda a interface gráfica com o utilizador deverá ser apresentada nos idiomas Português e Inglês;
- Todos os vídeos a submeter para a plataforma do parceiro devem ser transcodificados para os formatos definidos pelo parceiro de forma a garantir a compatibilidade em todos os pontos do processo;
- Todo o processo de adaptação deve ser implementado de modo a poder ser adicionado mais parceiros com uma carga de desenvolvimento mínima.

2.2.2 Os Intervenientes

No que diz respeito aos intervenientes identificamos à partida dois âmbitos distintos, os intervenientes diretos e indiretos. Com esta designação separamos os intervenientes que interagem diretamente com o produto e os que não interagem. No entanto não interagindo com o produto final tem um papel fundamental para o seu desenvolvimento e *design*. Como intervenientes diretos identificamos os utilizadores provenientes do parceiro que irão usufruir do *software* no papel de criadores de conteúdos para a plataforma Educast. Os utilizadores serão os únicos a ter um contacto direto com o produto final e a satisfação deste grupo é vital para o sucesso do projeto. Como intervenientes indiretos identificamos a equipa da FCCN responsável pela gestão, manutenção e desenvolvimento do produto Educast que será a nossa ligação direta com o parceiro e responsáveis pela definição dos requisitos por parte da FCCN. Em segundo plano identificamos ainda o grupo de técnicos presentes nas instituições de ensino aderentes à RCTS e ao Educast que terão um papel de teste do produto que permite a obtenção de feedback muito importante para o aprimoramento da solução.

2.2.3 O Modelo de Casos de Uso

O padrão que representa as melhores práticas de engenharia de software, o UML (*Unified Modeling Language*) ajuda-nos a definir e perceber a lógica de um sistema, quais as suas responsabilidades, funcionalidades e quais as ações que são disponibilizadas ao utilizador que vai interagir com a solução.

2.2.3.1 Casos de Uso Identificados

- Início de sessão de utilizadores no Easy2rec através do sistema de autenticação do parceiro Educast;
- O parceiro no processo de autenticação de utilizador deverá solicitar uma licença de utilização Easy2rec atribuindo-a ao utilizador em questão;
- Utilizador pode ver a lista de vídeo aulas já gravadas;
- Utilizador pode ver a lista de vídeo aulas gravadas que já foram submetidas para a plataforma parceira Educast;
- Utilizador pode selecionar um ou mais vídeos da lista;
- Utilizador pode submeter uma ou mais vídeo-aulas para a plataforma Educast sequencialmente;
- Para cada vídeo a submeter deverá preencher um formulário de metadados a associar ao vídeo em questão.

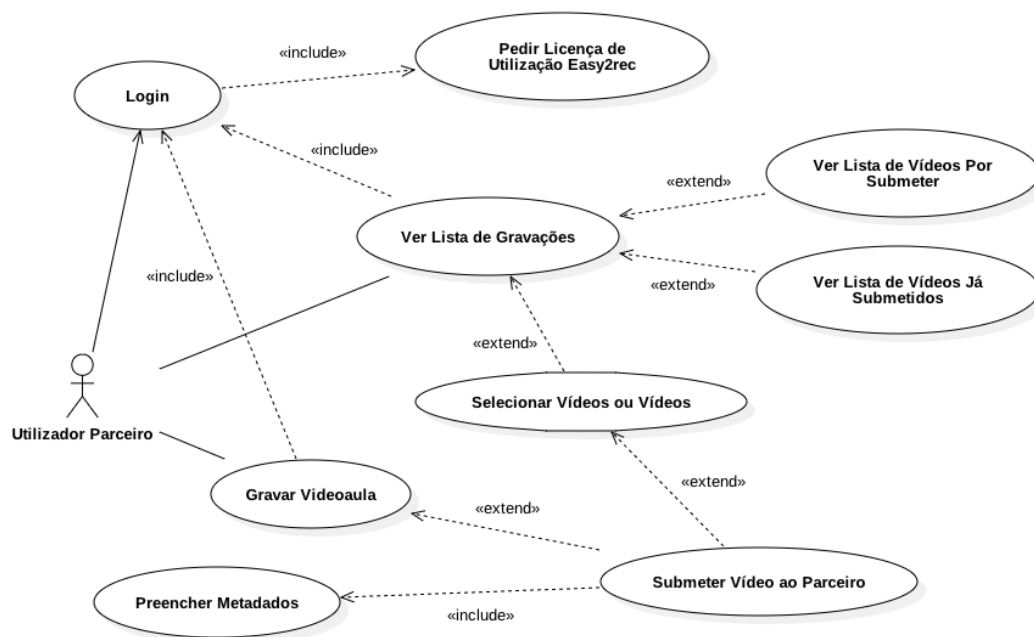


Figura 3 - Diagrama de Casos de Uso identificados

2.2.3.2 Caso de Uso: Login

O utilizador seleciona a opção entrar com um parceiro. Após selecionar da lista de parceiros disponíveis com qual pretende iniciar sessão, este é encaminhado para o sistema de autenticação externo do parceiro. O utilizador insere as suas credencias quando solicitadas para realizar o login. Quando o login for efetuado com sucesso o sistema do parceiro solícita uma licença de utilização Easy2rec a atribuir ao respetivo utilizador. Após o processo de autenticação e licença se encontrar concluído por parte do parceiro este deve notificar a aplicação Easy2rec para iniciar sessão com o utilizador.

2.2.3.3 Caso de Uso: Gravar Videoaula

O utilizador seleciona as fontes que deseja capturar e carrega os conteúdos. Inicia a gravação. Durante a gravação e após o utilizador selecionar terminar a gravação o sistema regista algumas informações referentes à gravação terminada. O utilizador pode optar por prosseguir diretamente para a submissão da respetiva gravação para o portal parceiro ou submeter posteriormente através da listagem de gravações guardadas localmente.

2.2.3.4 Caso de Uso: Ver Lista de Gravações

O sistema mostra em formato de lista todas as gravações realizadas pelo utilizador com sessão iniciada. Cada gravação contém alguns dados associados (e.g. título, data, duração), de modo a facilitar a gestão dos vídeos. O utilizador pode filtrar a lista mostrando apenas vídeos que já foram submetidos ou vídeos ainda por submeter.

2.2.3.5 Caso de Uso: Submeter Vídeo

O utilizador seleciona a gravação que deseja submeter e preenche o formulário de metadados definido pelo parceiro. Após o preenchimento e a submissão do formulário o sistema inicia o processo de transcodificação dos ficheiros. Após a conclusão da transcodificação é iniciada a submissão para o servidor do parceiro. Assim que terminado o utilizador recebe a notificação informando do sucesso ou insucesso no processo bem como algumas informações (e.g. hora de término do processo, duração do processo, tamanho total dos ficheiros submetidos).

2.2.3.6 Caso de Uso: Preencher Metadados

O utilizador para cada vídeo selecionado a submeter para a plataforma do parceiro deverá preencher um formulário de metadados que irá ser associado à submissão. A estrutura e os dados requisitados ao utilizador são únicos e exclusivamente da responsabilidade da entidade parceira.

2.2.4 Diagrama de Sequencia de Sistema

Na figura 4 são representadas as comunicações ou interações realizadas entre o ator utilizador e o sistema nomeadamente a aplicações Easy2rec.

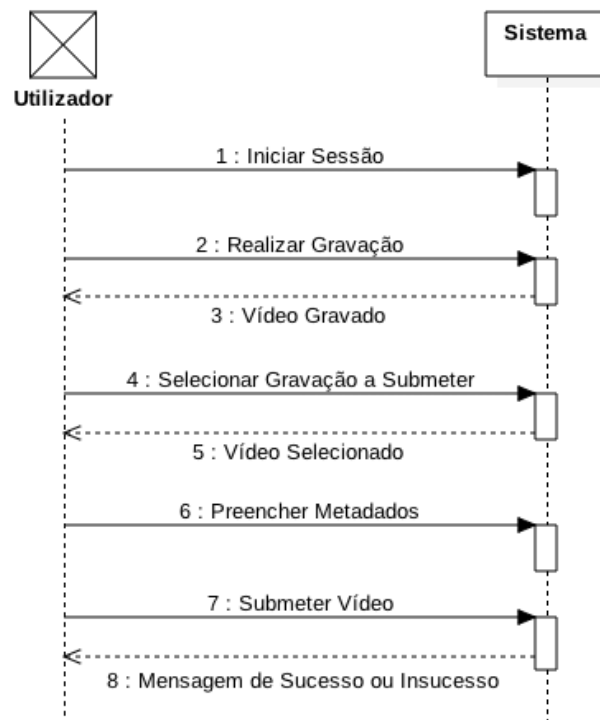


Figura 4 - Diagrama de Sequencia de Sistema da Aplicação Easy2rec

Na figura 5 são representadas as comunicações ou interações realizadas entre o sistema se autenticação do parceiro e o serviço de licenciamento do Easy2rec.

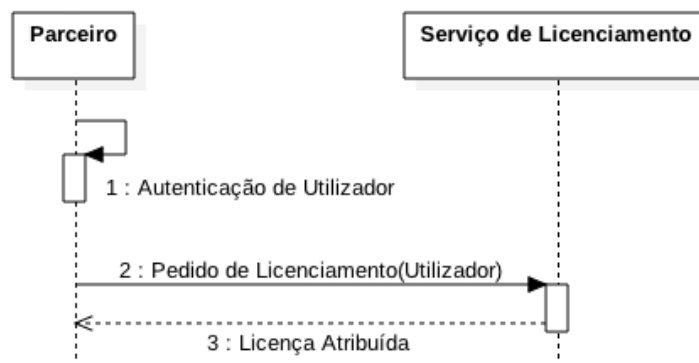


Figura 5 - Diagrama de Sequencia de Sistema do *Serviço de Licenciamento* Easy2rec

2.2.5 Modelo de domínio

2.2.5.1 Conceitos de Domínio: Aplicação Easy2rec

Para a identificação de conceitos de domínio usa-se a lista de categorias sugeridas por Larman no livro *“Applying UML and Patterns”*. Como resultado temos a seguinte tabela nº 1 de conceitos por categoria.

Tabela 1 - Conceitos de Domínio: Aplicação Easy2rec

Categoria	Classes conceptuais candidatas
Transações	Início de Sessão Gravação Submissão
Linhas de Transações	
Produtos ou Serviços relacionados com transações	Dados de Utilizador Licença de Utilizador Vídeo Gravado Vídeo a Submeter
Registos de transações	Registo de Gravação Registo de Submissão
Papéis de pessoas	Utilizador
Lugares	
Eventos	Início de Sessão Gravação Submissão
Objetos físicos	
Especificações e descrições	Vídeo Metadados Parceiro
Catálogos	Lista de Parceiros
Conjuntos	Lista de vídeos por submeter Lista de vídeos submetidos
Elementos de Conjuntos	Vídeo por submeter Vídeo submetido
Organizações	EduGraal/Easy2rec Parceiro (FCCN)
Outros sistemas	Sistema de Autenticação de Utilizadores do Parceiro
Registos (financeiros), de trabalho, contratos, documentos legais	Contrato de Parceiro Válido
Instrumentos financeiros	
Documentos referidos/para executar tarefas	Conta de utilizador no parceiro válida Licença Utilizador Easy2rec

2.2.5.2 Diagrama de Modelo de Domínio: Aplicação

Com base nos conceitos de domínio identificados anteriormente na figura 6 são identificadas as classes candidatas e as respetivas ligações entre elas.

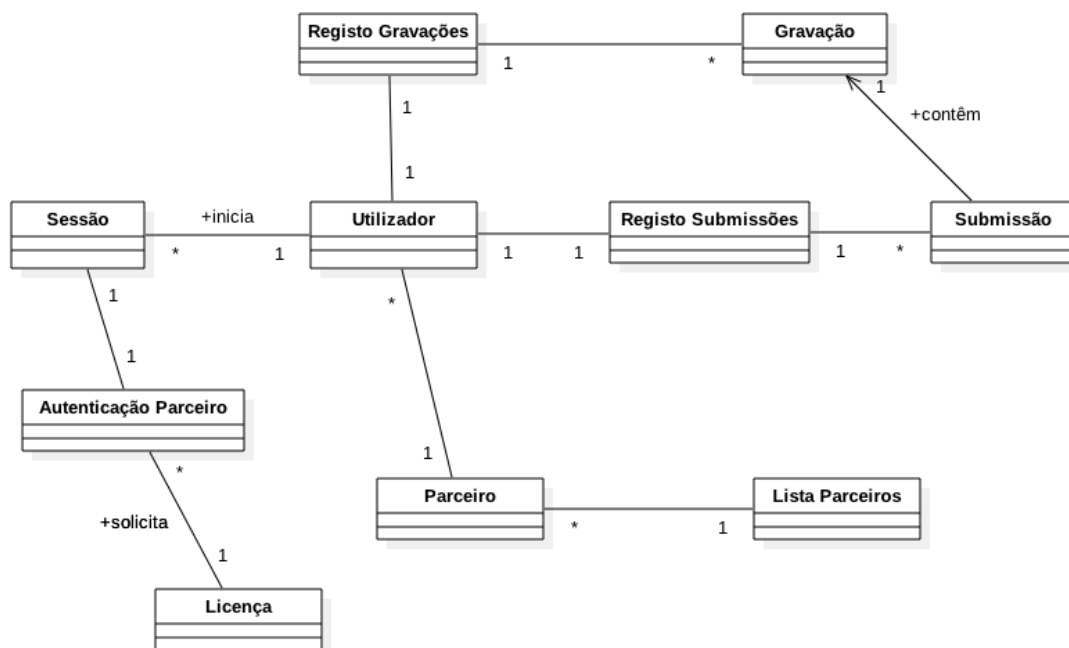


Figura 6 - Diagrama de modelo de domínio: Aplicação

2.2.5.3 Conceitos de domínio: Serviço de Licenciamento

Para a identificação de conceitos de domínio usa-se a lista de categorias sugeridas por Larman no livro “*Applying UML and Patterns*”. Como resultado temos a seguinte tabela nº 2 de conceitos por categoria.

Tabela 2 - Conceitos de Domínio: Serviço de Licenciamento

Categoria	Classes conceptuais candidatas
Transações	Pedido de Licenciamento
Linhas de Transações	
Produtos ou Serviços relacionados com transações	Utilizador
Registos de transações	Registo do Pedido
Papéis de pessoas	Parceiro
Lugares	
Eventos	Pedido de Licenciamento
Objetos físicos	
Especificações e descrições	Dados do Utilizador
Catálogos	
Conjuntos	Lista de Parceiros
Elementos de Conjuntos	Parceiro
Organizações	Edugraal/Easy2rec Parceiro (FCCN)
Outros sistemas	Sistema de Autenticação de Utilizadores do Parceiro
Registos (financeiros), de trabalho, contratos, documentos legais	Contrato de Parceiro Válido

Instrumentos financeiros	
Documentos referidos/para executar tarefas	Conta de Utilizador no Parceiro Válida

2.2.5.4 Diagrama de Modelo de Domínio: Serviço de Licenciamento Easy2rec

Com base nos conceitos de domínio identificados anteriormente na figura 7 são identificadas as classes candidatas e as respetivas ligações entre elas.



Figura 7 - Diagrama de Modelo de Domínio: Serviço de Licenciamento Easy2rec

2.2.6 Os Requisitos Não Funcionais

Os requisitos funcionais são como o nome indica todos os requisitos que não estão diretamente ligados com as funcionalidades disponibilizadas pelo produto. Muitos dos requisitos não funcionais referem-se ao produto como um todo. Desta forma por vezes estes são mais críticos do que os requisitos funcionais, pois se a implementação de um requisito funcional falhar pode desintegrar o produto, mas se falhar um requisito não funcional pode levar a que o produto como um todo possa nem sequer ser válido e deste modo não possa ser usado. De seguida são identificados alguns dos requisitos não funcionais do projeto seguindo as categorias sugeridas por Sommerville no livro *“Software Engineering”*.

2.2.6.1 Requisitos de Produto

- Portabilidade
 - O produto deverá ser implementado para apenas uma plataforma especificamente: a plataforma Windows.
- Usabilidade
 - Um utilizador deverá ser capaz de gravar e submeter uma vídeo-aula após não mais de 1 hora de orientação;

2.2.6.2 Requisitos Organizacionais

- Entregas
 - Durante o desenvolvimento deverá ser reportado à equipa técnica do parceiro a evolução do desenvolvimento ao longo do mesmo;
 - No fim do desenvolvimento deverá ser desenvolvido um tutorial em formato de apresentação para o utilizador seguir o processo de envio para a plataforma;
- *Standards*
 - Todo o texto usado nas comunicações deve seguir o *standard* UTF-8;

2.2.6.3 Requisitos Externos

- Interoperabilidade
 - O produto deverá integrar com a plataforma Educast de uma forma não abusiva e de continuidade com o que os utilizadores habitualmente encontram e usam.
- Segurança
 - Para todas as comunicações entre os intervenientes deverá ser garantido a sua autenticidade e origem bem como todos os dados devem ser verificados;
 - O sistema deve garantir que os dados e acesso às funcionalidades estão protegidos de acessos não autorizados;
 - Todas as comunicações com a entidade parceira devem ser encriptadas;

2.2.7 As Restrições

Por condições internas e externas são identificados determinados fatores capazes de restringir ou impossibilitar a escolha de determinado caminho, restrições essas que nos impõem a optar por escolhas que à partida não seriam as ideais. Deste modo são apresentadas de seguida as restrições identificadas durante a análise do problema:

- As tecnologias previamente adotadas para o desenvolvimento de ambos os produtos a integrar deverão ser tidas em conta no momento de avaliação e escolha das tecnologias para a implementação e resolução do problema.
- O Educast tem implementado o seu sistema de login que assenta sobre uma estrutura de autenticação e autorização federada, possuindo o seu próprio interface e processo não sendo disponibilizado em forma de *API* ou *Web Service*. Desta forma é uma limitação na integração e que irá obrigar a ser integrado o processo de login de uma forma encapsulada no Easy2rec.

2.3 Análise de Valor

A definição de uma proposta de valor é vital para o sucesso de um produto ou serviço, pois é desta forma que é transmitido aos possíveis clientes os benefícios que os clientes terão ao usar o produto. Uma definição de valor bem estruturada é importante que pode ser usada como uma ferramenta para sobressair entre os restantes concorrentes, captando mais e melhor negócio. Para uma proposta de valor bem definida é crucial que seja claro para o cliente os benefícios ao adquirir o produto ou serviço em questão, evitando usar linguagem demasiado técnica. A proposta de valor deve também ser direta e demonstrar quais as suas diferenças comparativamente aos seus concorrentes.

O valor de um mesmo produto ou serviço não é igual para todos os potenciais clientes. Desta forma surge o conceito de valor percebido, que consiste no valor que um produto ou serviço tem para um determinado consumidor. O valor percebido afeta o preço que o potencial cliente se encontra a pagar pelo produto ou serviço. A maior parte dos consumidores não têm a noção de qual os reais preços de custo dos produtos que compram, por isso baseiam-se no valor que eles percebem do produto. Deste modo para conseguir obter um preço maior por um produto os produtores trabalham no sentido de melhorar o valor percebido dos seus produtos por parte dos consumidores. Desta forma é importante ter em conta na definição de valor estes aspetos de modo a maximizar a taxa de sucesso na conversão de possíveis clientes em compradores.

Vejamos o seguinte exemplo, os produtos na área do vestuário tendem a relacionar o valor percebido com a marca que o comercializa. Uma marca muito conhecida do grande público pode de alguma forma aumentar os preços de venda praticados pois os clientes têm esta marca como de qualidade enquanto outras marcas que não possuem essa reputação têm de baixar os preços para poderem competir por uma fatia de mercado.

O *target customer* do produto em questão numa primeira abordagem são as entidades que necessitam de um *software* capaz de gravar vídeos com algumas especificidades bem definidas. Neste caso específico é parceiro FCCN o cliente, para o produto Educast e consequentemente os seus utilizadores. Os utilizadores do Educast que possuam computadores com o sistema operativo *Windows* estão impossibilitados de usar a plataforma Educast na perspetiva de criadores de conteúdo pois a plataforma não apresenta solução para este segmento de utilizadores.

No entanto existem outros segmentos de clientes como os utilizadores não afiliados com nenhuma entidade que podem sempre registar-se como utilizadores do Easy2rec e tirar partido das funcionalidades de gravação e exportação de vídeos. Para este caso vamos nos focar no primeiro segmento.

Como benefícios identificamos:

- Qualidade do produto devido ao ponto de partida assentar num produto já existente e testado;
- Personalização do produto ao longo do tempo e das necessidades identificadas;
- Produto confiável no segmento já desenvolvido;
- Flexibilidade de adaptação e integração;
- Uso gratuito pelos utilizadores.

Como custos identificamos:

- Preço de desenvolvimento atribuído ao parceiro FCCN;
- Tempo de desenvolvimento atribuído ao parceiro FCCN;
- Tempo de aprendizagem de utilização atribuído ao utilizador do parceiro.

Um produto com as características do Easy2rec permite a implementação de vários modelos de negócio dependendo do público-alvo e os objetivos de utilização. Rapidamente identificamos dois tipos de negócio possível. Um para as plataformas de vídeo que necessitam de vários utilizadores em simultâneo e conseqüentemente necessitam de um número elevado de licenças e uma adaptação as suas necessidades e por outro lado os utilizadores singulares com necessidades totalmente diferentes do primeiro caso.

Atualmente o Easy2rec está disponível *online* no website oficial onde qualquer pessoa pode descarregar e experimentar o produto sem qualquer custo. O negócio associado baseia-se essencialmente no modelo de negócio *freemium*. O produto é distribuído gratuitamente *online* no entanto o produto apresenta algumas funcionalidades mais avançadas bloqueadas. Apenas utilizadores que possuam uma licença válida adquirida no *website* podem usufruir de todas as funcionalidades. Para o primeiro caso cada parceiro é um caso específico e deverá ter uma proposta personalizada para as suas necessidades que devem ser discutidas.

De forma a entender melhor como a empresa pode gerar e capturar valor focando no público-alvo identificado anteriormente como parceiros é elaborado o plano de negócios no formato de *canvas* apresentado na figura 8. Esta forma de apresentar o negócio facilita a entender um pouco melhor quem é o cliente, qual o problema que enfrenta e como o produto o pode resolver e finalmente como o produto pode render com esse processo.

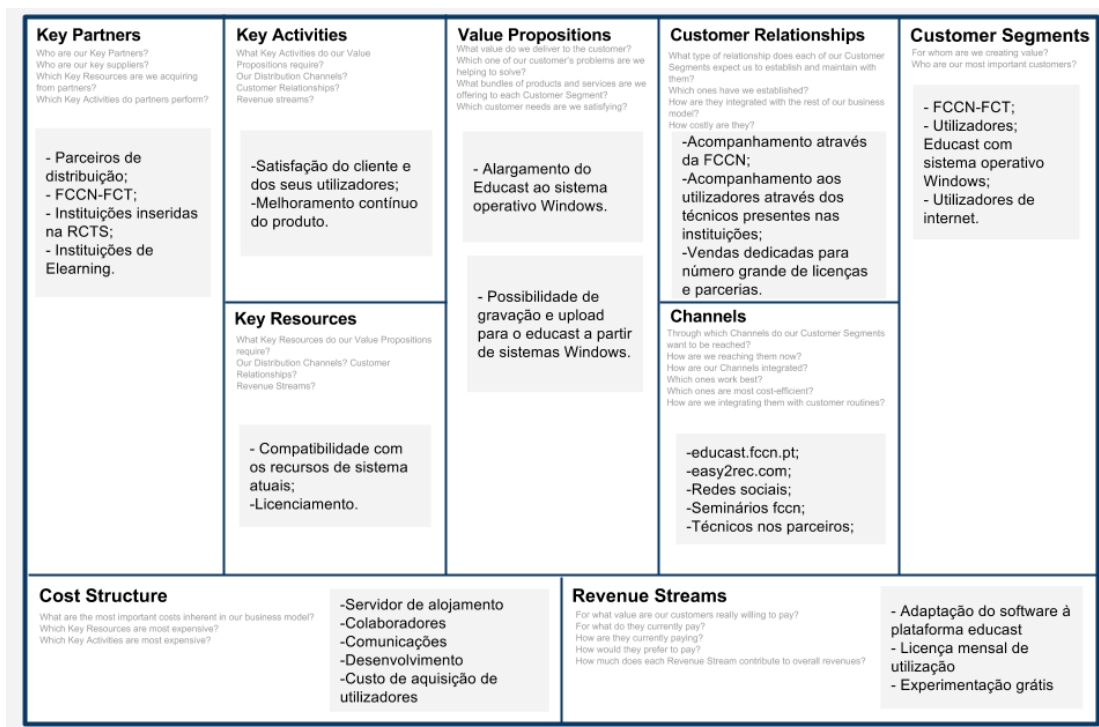


Figura 8 - Modelo de Negócio Canvas

De forma a analisarmos e quantificarmos a possibilidade de criação de valor através da exploração do nosso produto podemos tirar partido de vários métodos de que nos podem auxiliar quando necessitamos de analisar e quantificar a criação de valor. Deste modo iremos estudar o método conceptual apresentado de seguida (Herrala et al, 2011). Para iniciarmos deveremos ter em conta os seguintes conceitos que se iram relacionar no modelo em questão:

- Valor para o cliente;
- Competências Principais;
- Relações;
- Interação.

2.4 Estado da Arte em Soluções

Na análise de estado da arte em soluções iremos essencialmente analisar plataformas que se prossuponham a atender ao mesmo objetivo que o nosso. As diferentes plataformas têm, diferentes abordagens para o mesmo problema. Vamos abordar as diferentes soluções implementadas nas diferentes NREN's (*National Research and Education Network*) europeias. As soluções serão analisadas com base em todo o processo desde a captura dos conteúdos até a sua distribuição.

2.4.1 Capturer Lecture Video Recording System⁹

Sistema desenvolvido pela universidade de Tel Aviv em Israel para a plataforma *Windows* (XP, Win 7) que integra o sinal de vídeo do orador com a apresentação capturada através de sinal VGA dentro de um mesmo *frame*. O output da gravação é um só ficheiro de vídeo que pode ser editável adicionando a marca da instituição. O sistema tira partido da solução externa *Microsoft Windows Media Encoder* ou *VLC* para codificar os sinais de entrada.

2.4.2 SWITCHcast Video Management System¹⁰

Sistema desenvolvido pela *Switch* sediada na Suíça que suporta todo o processo desde a gravação até a distribuição e visualização das vídeo-aulas nos diferentes dispositivos. Desenvolvida para a plataforma *Apple* o seu gravador encontra-se disponível apenas para os sistemas *Mac OS*. Uma das suas principais características é a possibilidade de dividir o conteúdo original em partes. O conteúdo publicado é protegido através de sistema de autenticação federada e uma infraestrutura de autorização (AII). A gravação é realizada através de um computador independente que captura vídeo do orador e a sua apresentação através de um dispositivo como o *Ephiphan* que capta o ecrã do computador que está a apresentar. Os vídeos são depois de terminada a captura enviados de forma independente para a plataforma *online* para serem editados e consequentemente publicados.

2.4.3 Videotorium¹¹

Sistema desenvolvido pela NIFF da Hungria é um produto de *hardware* desenvolvido sobre a plataforma *Linux* usado para a gravação de vídeo-aulas capturando apenas o sinal de vídeo e áudio do orador. Para a gravação da apresentação é necessário um computador externo que capta *snapshots* da apresentação através de um dispositivo como o *Ephiphan*. O sistema é capaz de gravar como também fazer *streaming* em simultâneo dos sinais que está a capturar. O portal web do sistema combina os sinais capturados apenas no momento de visualização sincronizando automaticamente o vídeo capturado como a apresentação.

2.4.4 L2L (Live to e-Learning)¹²

Sistema desenvolvido pela Cineca em Itália consiste em três componentes essenciais, o gravador desenvolvido para a plataforma *Windows*, o portal de conteúdos e metadados *MediaMosa* e o *plugin* para o *Moodle*. O gravador é capaz de gravar até um sinal de áudio e

⁹ Mais informações disponíveis em <http://video.tau.ac.il/>

¹⁰ Mais informações disponíveis em <http://www.switch.ch/cast/>

¹¹ Mais informações disponíveis em <http://videotorium.hu/>

¹² Mais informações disponíveis em <http://www.cineca.it/en>

dois de vídeo sincronizados exportando no formato WMV. A plataforma MediaMosa é open-source baseada em *Web Service* responsável pela pesquisa, reprodução, submissão, e distribuição dos conteúdos. No *Moodle* é possível incorporar conteúdos através de um *applet* a partir de Flash.

2.4.5 ViPS¹³

Sistema desenvolvido pela *Kaunas University of Technology* da Lituânia baseado numa aplicação em Flash que necessita da submissão prévia da apresentação a ser gravada e que capta o vídeo do orador detetando quando este avança de slide de forma a sincronizar para a posterior visualização. A gravação e codificação dos vídeos é realizada através de um serviço de vídeo como o *Red5 Media Server* ou em alternativa o *Wowza Media Server*. O portal de gestão de conteúdos é desenvolvido sobre a plataforma *Drupal*.

2.4.6 Opencast¹⁴

Sistema desenvolvido pela universidade de Vigo em Espanha consiste numa agregação de três componentes distintos são eles: o *Matterhorn*, o *Galiscaster* e o *PuMuKit*. O *Matterhorn* é o principal componente desta solução que consiste na plataforma de gestão de vídeo-aulas. O *Galiscaster* é um *software* desenvolvido para a plataforma *Linux* capaz de capturar e gravar sinais síncronos de vídeo e áudio. Por último o *PuMuKit* é uma plataforma que permite criar um portal para a agregação de conteúdos de vídeo focado nos utilizadores que visualizem os conteúdos.

2.5 Análise de Tecnologias

As tecnologias a utilizar no desenvolvimento da solução proposta são variadas e iremos realizar um breve estudo sobre as principais existentes e possíveis de nos auxiliar no desenvolvimento de uma maneira mais rápida e eficaz comparando umas com as outras sendo desta forma possível escolher as que são mais adequadas para a resolução do problema.

2.5.1 Simple Object Access Protocol (SOAP)

Protocolo padrão de troca de mensagens estruturado em XML que possibilita à comunicação entre serviços. Ele define os componentes essenciais e opcionais das mensagens transmitidas

¹³ Mais informações disponíveis em <https://vips.liedm.lt/en>

¹⁴ Mais informações disponíveis em <http://www.opencast.org/>

entre serviços através do protocolo *HTTP*. O protocolo *SOAP* especifica três diferentes aspectos no processo da mensagem (Bell et al. 2001):

- **Envelope** define uma estrutura *XML* para informação comum que integra qualquer mensagem *SOAP*. A informação contida inclui o endereço e regras de caminho. O envelope contém a mensagem que irá ser enviada de onde para onde;
- **Encoding** define como são codificados os tipos de dados no *XML* (*Strings*, números, datas, etc.);
- **Procedural conventions** define como um método por ser representado num pacote de mensagem *SOAP*.

De seguida na figura 9 é apresentado um exemplo de pedido *SOAP*:

```
POST http://www.websvcicex.com/globalweather.asmx HTTP/1.1
Host: www.websvcicex.com
Content-Length: 341
Content-Type: application/soap+xml

<?xml version="1.0">
<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope">
  <soap:Header></soap:Header>
  <soap:Body>
    <ns1:GetWeather xmlns:ns1='http://www.websvcicex.NET'>
      <ns1:CityName>Montreal</ns1:CityName>
      <ns1:CountryName>Canada</ns1:CountryName>
    </ns1:GetWeather>
  </soap:Body>
</soap:Envelope>
```

Figura 9 – Exemplo de pedido *SOAP*¹⁵

2.5.2 Web Services Description Language (*WSDL*)

Um modelo e formato *XML* para descrever *web services*, possibilitando a separação entre a funcionalidade oferecida da sua descrição abstrata. A descrição do *web service* define a sua interface, ou seja, o conjunto de operações possíveis entre o provedor e o cliente do serviço, bem como a estrutura das mensagens *SOAP* que serão necessárias criar para comunicar com os métodos. A plataforma *ASP.NET* consegue gerar o documento *WSDL* automaticamente para cada *web service* criado (Bell et al. 2001).

2.5.3 Universal Description Discovery and Integration (*UDDI*)

Mecanismo para que os clientes possam encontrar um determinado *Web Service*. O *UDDI* pode ser considerado o *DNS* para os *Web Service's*, contendo informações sobre o provedor

¹⁵ Disponível em <http://bizcoder.com/a-fresh-coat-of-rest-paint-on-a-soap-stack>

do serviço, sua localização, além das descrições do serviço (*WSDL*). Os repositórios de serviços podem ser públicos ou privados, sendo que tanto o registo quanto a consulta do serviço é realizado via de arquivos *XML* (Bell et al. 2001).

2.5.4 ASP.NET

Plataforma constituída por um conjunto de classes cujo principal objetivo é servir pedidos efetuados através do protocolo *HTTP*. Parte integrante da plataforma *.NET* foi desenhada para encapsular os controlos usados mais frequentemente desde um botão a correr do lado do servidor, até *listboxes* e *grids* até aos controlos mais complexos. Temos a possibilidade também de desenvolver os nossos próprios controlos de maneira a serem reutilizados. *ASP.NET*¹⁶ também inclui suporte para desenvolvimento de *Web Service's* fazendo com que de uma forma simples seja possível criar serviços usando *standards* como *SOAP*, *WSDL* e *UDDI* que já foram referenciados anteriormente neste documento (Bell et al. 2001).

2.5.5 SQL SERVER¹⁷

É uma plataforma de gestão de bases de dados relacionais desenvolvida pela *Microsoft*. A sua principal função é de armazenar, gerir e retornar dados solicitadas através de pesquisas utilizando a linguagem *SQL* sejam eles feitos numa rede local como também através da *Internet* (Microsoft, 2016).

2.5.6 FRAMEWORK .NET

É uma plataforma desenvolvida pela *Microsoft* de desenvolvimento e execução de aplicações. Esta plataforma é executada sobre uma *Common Language Runtime (CLR)*, que consegue interpretar mais de vinte linguagens de programação diferentes, fazendo-as interagir entre si como se de uma linguagem apenas se tratasse, permitindo assim a interoperabilidade. A *.NET Framework*¹⁸ contém também um conjunto de bibliotecas pré-codificadas que oferecem soluções para os mais comuns problemas, tais como a acesso a dados e interfaces gráficas, entre outras. (MSDN 2016)

Algumas das vantagens mais relevantes da *.NET Framework* são:

- Interoperabilidade entre plataformas e componentes desenvolvidos em outras linguagens suportadas pela *.NET Framework*;

¹⁶ Mais informação disponível em <https://www.asp.net/>

¹⁷ Mais informação disponível em <https://www.microsoft.com/pt-pt/server-cloud/products/sql-server/overview.aspx>

¹⁸ Mais informação disponível em [https://msdn.microsoft.com/en-us/library/w0x726c2\(v=vs.100\).aspx](https://msdn.microsoft.com/en-us/library/w0x726c2(v=vs.100).aspx)

- Independência de linguagem de programação (CLS);
- Ambiente de execução que minimiza os conflitos de distribuição e consequentes atualizações através de versões atualizadas;
- Compartilhamento de componentes desenvolvidos entre as diversas linguagens de forma transparente para o programador;
- Fim dos problemas com DLL.

2.5.7 C Sharp (C#)

A linguagem de programação *C#*¹⁹, também escrito como *C Sharp*, é uma linguagem de programação orientada a objetos, desenvolvida pela Microsoft como parte integrante da *.NET Framework*. Esta linguagem, a par da introdução de inovações ao nível da orientação aos componentes e da programação declarativa com atributos, tira total partido dos serviços de segurança, da compilação *just-in-time*, da gestão automática de memória e de um conjunto vasto de bibliotecas da *.NET Framework*. A sua sintaxe orientada a objetos foi baseada no *C++* mas inclui muitas influências de outras linguagens de programação, como *Object*, *Pascal* e *Java*. Esta linguagem surgiu em 2001 (Marques, 2005).

2.5.8 FFMPEG

O *FFMPEG*²⁰ é uma biblioteca muito completa, desenvolvida para funcionar em várias plataformas e que permite gravar, converter e transmitir vídeo e áudio de uma forma rápida e fiável. Permite criar ficheiros de áudio e de vídeo a partir de vários formatos e converter para um conjunto extenso de formatos. É uma biblioteca desenvolvida sobre *open-source*. (FFMPEG, 2016).

2.5.9 Extensible Markup Language (XML)

A linguagem de marcação também conhecida como *Extensible Markup Language*²¹, é utilizada para criar documentos com dados organizados de forma hierárquica. O seu desenvolvimento começou em meados da década de 90, pela *World Wide Web Consortium (W3C)*, e tem como principal objetivo facilitar a partilha de informação através da *Internet*. Uma vez que o *XML* é na sua essência um ficheiro de texto, que possui uma grande vantagem pelo facto de ser independente de plataformas, podendo por isso ser interpretado por diferentes sistemas. De forma geral um documento *XML* não está dependente de nenhuma plataforma específica e foi

¹⁹ Mais informação disponível em <https://msdn.microsoft.com/en-us/library/67ef8sbd.aspx>

²⁰ Mais informação disponível em <http://ffmpeg.org>

²¹ Mais informação disponível em <https://www.w3.org/XML/>

desenhada para fornecer aos programadores um mecanismo para descrever de uma forma clara e consistente o seu conteúdo e trocar informação na *Web* de maneira fácil e eficiente (Pitts, 1998).

2.5.10 JavaScript Object Notation (*JSON*)

Formato de troca de dados leve, fácil para os humanos de ler e escrever e fácil para as máquinas de gerar e analisar. É baseado num subconjunto da linguagem de programação *Javascript standard* ECMA-262 3ª Edição de Dezembro de 1999. *JSON*²² é um formato de texto completamente independente das plataformas em que é desenvolvido, mas usa convenções familiares aos programadores. A sua estrutura é definida como um conjunto de chaves e valores. Estas propriedades fazem do *JSON* uma linguagem ideal para o intercâmbio de dados. Exemplo de representação de informação de um vídeo estruturado em *JSON* (*JSON*, 2016):

```
{
  "id": 213,
  "titulo": "Marketing 101",
  "canal": "Fundamentais",
  "autor": "José Veiga"
}
```

Figura 10 – Exemplo de representação de informação em *JSON*

²² Mais informação disponível em http://www.w3schools.com/js/js_json_intro.asp

3 Avaliação das Soluções Existentes

De há alguns anos a esta parte o vídeo tomou uma importância cada vez maior no processo de ensino-aprendizagem sobre tudo através da democratização da internet. Neste sentido o desenvolvimento de plataformas que possam mediar este processo desde a gravação de vídeo-aulas até a sua distribuição pelos alunos é um tema em destaque nas várias NREN's europeias. Neste sentido a TERENA (*Trans-European Research and Education Networking Association*) foi responsável pela definição de um objetivo comum, denominado por *TF-Media*, que possibilita um espaço colaborativo com condições para inovar e partilhar conhecimento entre os vários participantes com objetivo de evoluir e apresentar soluções capazes de realizar a tarefa (TERENA, 2012). Neste âmbito têm vindo a ser lançados vários projetos ao nível das instituições de ensino superiores europeias e das várias NREN's, que demonstram uma real necessidade das instituições de modernizar os processos de ensino.

Para proceder à análise das soluções existentes iremos ter como base o estudo realizado às plataformas no evento que ocorreu em 2012, denominado de *TF-Media Recording Contest*, que pôs à prova as capacidades e funcionalidades dos vários sistemas desenvolvidos pelos participantes da *TF-Media*. Os participantes ficaram incumbidos no decorrer do evento de montar o seu sistema de gravação de acordo com alguns requisitos audiovisuais e proceder à gravação e posteriormente à sua publicação. Este evento permitiu com as mesmas condições técnicas comparar uma série de plataformas dedicadas à gravação de aulas à escala europeia (TERENA, 2012).

Os sistemas apresentados foram avaliados e comparados com base na estrutura inicialmente definida pela equipa da TERENA, definindo várias áreas que os sistemas deveriam colmatar e que é detalhada de seguida na figura 11:

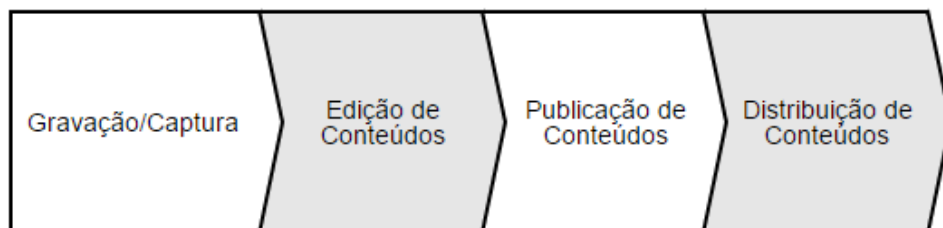


Figura 11 - *Workflow* de sistema de gravação de vídeo-aulas²³

Com base nos resultados apresentados no relatório “*Summary Report*”²⁴ do respetivo teste realizado aos sistemas no evento, temos então a informação necessária para realizar uma análise que de outra forma seria impossível de obter, pois como vamos analisar todas as soluções usam sistemas de autenticação fechados ao público em geral, tal como o Educast. De seguida é apresentada a comparação entre as diferentes plataformas presentes no evento supracitado e identificadas as tecnologias e componentes utilizadas em cada uma delas para posteriormente analisarmos em relação às nossas necessidades e requisitos.

Tabela 3 - Comparação entre sistemas de lecture recording²⁵

Característica	Capturer Lecture Video Recording System	SWITCHcast	Videotorium	L2L	ViPS	Opencast
Plataforma do Gravador	Windows	Mac OS	Linux	Windows	Flash	Linux
Tecnologia do visualizador	Não incluído mas maioria dos visualizadores podem ser usados	Adobe Flash, Quicktime Podcast	Adobe Flash	Adobe Flash	Adobe Flash, HTML5 Vídeo	Adobe Flash, HTML5 Vídeo
Tecnologia de captura de vídeo	Composição de captura de ecrã e apresentação em vídeo	Firewire	Video4Linux	Windows Media Encoding	Adobe Flash Câmara	Video4Linux
Resolução máxima de ecrã	960x540 (@ 20 FPS)	HD	HD	HD	640x480	HD
Tecnologia de	Frame grabber	Dois	Apresenta	Apresenta	Apresenta	Dois

²³ Disponível em <https://www.terena.org/publications/files/TF-Media-Contest-Report-final.pdf> (tradução nossa)

²⁴ Disponível em <https://www.terena.org/activities/media/rc/TF-Media-Contest-Report-final-2.pdf>

²⁵ Disponível em <https://www.terena.org/publications/files/TF-Media-Contest-Report-final.pdf> (tradução nossa)

captura de apresentação	ou VNC	ficheiros de vídeo independentes	ção pré-submetida pelo técnico	ção pré-submetida	ção pré-submetida	ficheiros de vídeo independentes
Conteúdo de apresentação	Integrado com o vídeo principal	Vídeo completo	Snapshots da apresentação	Snapshots da apresentação	Snapshots da apresentação	Vídeo completo
Formato de entrada do conteúdo	Qualquer aplicação	Qualquer aplicação	Qualquer aplicação	Qualquer aplicação	Aplicações suportadas	Qualquer aplicações
Formato de saída de conteúdo	Um vídeo	Dois vídeos	Vídeo e snapshots da apresentação	Vídeo e apresentação	Vídeo e snapshot da apresentação ou janela capturada	Dois vídeos
Plataforma de Publicação	Moodle, Joomla, Drupal	SWITCHcast	Videotorium	MediaMosa	Drupal Content Management System	PuMuKit
Plataforma de distribuição	Windows Media Server, Flash Media Server	Flash Media Server	Wowza Media Server	Flash Media Server	Flash e Wowza Media Servers	Flash Media Server
Capacidades de edição em tempo real	Linha de tempo gráfica	Linha de tempo gráfica	Linha de comandos	Possível	Não possível	Não possível
Paradigma de integração	Conjunto de ferramentas	Integração completa	Integração completa	Integração completa	Integração completa	Integração completa
Capacidades técnicas do Docente ou gravação autónoma	Média	Baixo	Baixo (unicamente para vídeo)	Médio	Baixo	Baixo
Suporte técnico necessário	Não	Não	Depende	Sim	Depende	Não
Transmissão em tempo real	Sim	Não	Sim	Sim	Sim	Sim
Autenticação	TAU LDAP	Federação All	Federação All	Moodle	Drupal	Federação All
Marca	Sim	Sim	Sim	Não aplicável	Não aplicável	Sim
Código fonte e	Grátis	Switch	Vários	Moodle	Vários	Open-

modelo de licenciamento			componentes open-source	plugin é open-source	componentes open-source	source
Requisitos de Hardware	PC e Ephemian (Opcional)	Mac e Ephemian	Linux box e ephiphian	Pc e Ephemian	Um ou dois PC's	PC e hardware compatível com a captura a V4L2
Custo estimado da solução	2000 EUR	2000 EUR + fee por canal no switchcast	1200 EUR para o equipamento	3000-5000 EUR incluindo o hardware	Não aplicável	Não aplicável
Manutenção/ Fornecedor do serviço atualmente	Instituição	NREN	NREN	Instituição	Instituição	Instituição
Uso	3 Universidades	40 Universidades	10 Conferências por ano	Universidades e 2 outras instituições	3000 Gravações : Sistema de educação à distância da Lituânia	60 Horas por semana até 10 salas até 2012

Numa primeira análise às características de cada sistema podemos concluir que existem várias abordagens diferentes para o mesmo problema e diferentes modos de alcançar o objetivo proposto. Cada equipa tomou as suas decisões optando por caminhos diferentes e formas diferentes de resolver o problema adaptando também às suas próprias necessidades e recursos disponíveis.

Em Portugal, o papel de implementação e disseminação de um sistema desta natureza encontra-se a cargo da FCCN, a NREN portuguesa que dá suporte à comunidade académica e de investigação nacional. Foi no âmbito de uma parceria pioneira europeia entre a FCCN e a SWITCH, que foi possível a implementação do Educast em Portugal. Esta parceria permitiu à FCCN o acesso à plataforma *SwitchCast* e a disponibilização do seu código fonte, que possibilitou à FCCN a sua implementação e adaptação à realidade portuguesa.

O serviço Educast foi lançado em Portugal em forma de protótipo em Setembro de 2010, sendo o seu lançamento oficial para toda a rede RCTS em Março de 2011 nas Jornadas FCCN, um evento anual. Deste modo o Educast encontra-se também representado na tabela nº 3 sobre a denominação *SwitchCast* visto que é baseado nesse mesmo sistema e herda todas as características identificadas. De forma a expandir e evoluir o sistema pretende-se alargar a compatibilidade do *software* de gravação para a plataforma *Windows* e esse é o principal objetivo da FCCN neste momento.

Partindo deste ponto inicial, são identificadas dentro de todas as analisadas duas que possuem gravadores compatíveis com a plataforma *Windows* que são elas a *Capturer Lecture Video Recording System* e a *L2L*. No entanto é necessário analisar se os restantes requisitos da plataforma *Educast* são atendidos na implementação destas soluções. Na tabela 4 são descritos os requisitos que a plataforma *Educast* necessita de verificar e se as soluções o implementam ou não.

Tabela 4 - Comparação entre soluções para a plataforma *Windows*

Requisito	Capturer Lecture Video Recording System	L2L
Sinais de vídeo gravados de forma independentes	Não	Sim
Conteúdo em formato de vídeo	Sim	Não
Autenticação Federada	Não	Não

Como podemos verificar nenhum dos sistemas identificados implementa a 100% os requisitos que possibilitaria uma integração direta com a plataforma *Educast*. No entanto a FCCN poderia adotar uma destas soluções e adaptar às suas necessidades, mas envolvia tempo e custos de desenvolvimento para a equipa técnica. Aqui entra o *Easy2rec*. O *Easy2rec* tendo a qualidade comprovada pela equipa técnica da FCCN desde uma fase inicial ainda quando era apenas um projeto académico que poderia facilmente ser adaptado à plataforma *Educast* necessitando apenas de ser incluído o método de autenticação implementado pela nossa equipa.

A vocação do parceiro FCCN para desenvolver os seus serviços e cada vez alargar e melhorar os seus serviços aos seus parceiros/clientes aproveitando também o que de bom se faz em Portugal foi um ponto fulcral para a possibilidade de avançar com este projeto e este desafio. Desta forma potenciar um serviço já existente da parte da FCCN e alargar o serviço a um maior numero de potenciais utilizadores e já que era uma necessidade pois era requisitado por inúmeros utilizadores do serviço *Educast* pertencentes à RTCS até este momento. Os requisitos bem especializados do parceiro condicionam também uma possível adoção a outro tipo de solução não tão adaptável como a nossa e com outro tipo de clientes.

A presente análise foi realizada com o principal foco em soluções semelhantes no contexto em que o *Educast* foi idealizado e desenvolvido dentro do âmbito das IES (Instituições de Ensino Superior) para respeitar determinados requisitos muito próprios. No entanto existem naturalmente soluções capazes de produzir o mesmo efeito passando por outro tipo de soluções como o *Youtube* e outras plataformas abertas que se bem definido um *workflow*

correto podem muito bem ser usadas para o mesmo efeito não necessitando de controlo de acessos.

As tecnologias identificadas à partida vão de encontro às tecnologias previamente estudadas e analisadas no desenvolvimento do produto Easy2rec optando pelas tecnologias Microsoft nomeadamente a plataforma de desenvolvimento *.NET Framework* pois abrangiam todos os requisitos e necessidades para o desenvolvimento. Neste ponto e de forma a adaptar o produto já desenvolvido nessas tecnologias a única opção que nos parece sensata é continuar o desenvolvimento nas mesmas tecnologias.

4 Design

No presente capítulo será apresentado todo o estudo realizado de análise conceptual do projeto descrevendo a sua arquitetura, com também os padrões e regras implementadas. Será apresentado a solução que satisfaz os requisitos funcionais e não funcionais identificados no capítulo 2. Nesta secção será descrito também o modelo de base de dados que suporta a integração entre os projetos envolvidos.

4.1 Design Conceptual

A solução implementada engloba três módulos ou componentes distintos de funcionamento individual, mas que comunicam entre si, são eles a aplicação Easy2rec, o serviço de licenciamento independente também da responsabilidade do Easy2rec e a plataforma do parceiro FCCN. O serviço de licenciamento Easy2rec é responsável por responder aos pedidos de licenciamento provenientes do parceiro e não comunica diretamente com a aplicação. A aplicação Easy2rec responsável pela gravação e submissão de vídeo aulas para a plataforma parceira Educast mas só depois de o utilizador se encontrar com a sessão iniciada e valida comunicada pelo parceiro através de uma notificação. Na figura 12 são representadas as ligações entre os módulos.

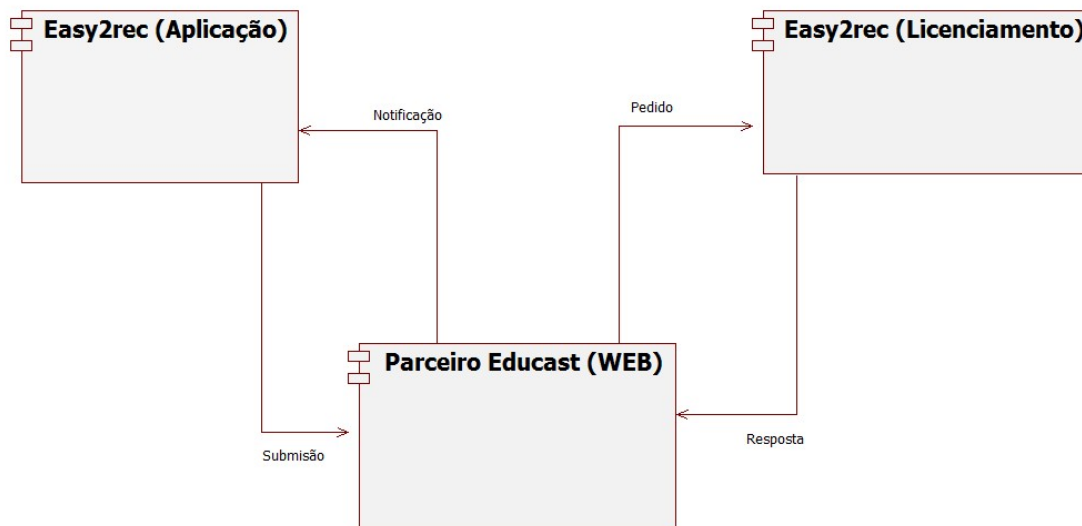


Figura 12 - Diagrama conceitual de componentes

4.2 Design Arquitetural

Os módulos a desenvolver apresentam diferentes necessidades que foram tidos em conta no momento da sua análise e na escolha de quais as arquiteturas mais indicadas a aplicar em cada um deles.

4.2.1 O Serviço de Licenciamento

No processo de licenciamento de utilizadores Educast para a utilização do Easy2rec é necessário implementar um serviço que disponibilize ao Educast uma forma de pedir licença de utilizador após o login por parte dos utilizadores. Para a sua implementação será usada uma arquitetura orientada ao serviço visto se será necessário implementar um *Web Service* para comunicar entre os dois sistemas através de troca de informação estruturada (Sommerville, 2001). Consequentemente para o acesso aos dados iremos implementar uma arquitetura multicamadas separando as responsabilidades por camadas independentes como o acesso aos dados, as regras de negócio e apresentação ao utilizador. A finalidade desta divisão em camadas assenta na separação de responsabilidades permitindo a criação de níveis de abstração entre os vários módulos. Esta opção não só facilita a implementação faseada da solução, como também permite a sua evolução de modo articulado e modular.

4.2.2 A Aplicação Easy2rec

A aplicação Easy2rec funciona num modelo de transformação ao longo do tempo dos dados capturados e sucessivas transformações dos mesmos. O modelo de arquitetura mais adequado a implementar será o *Data Flow Model* dentro dos modelos modulares. Neste

modelo de arquitetura os processos funcionais transformam os *inputs* e produzem os *outputs*. A informação processada circula entre processos de um para outro e é transformada enquanto avança na sequencia. As transformações podem ser executadas como um processo sequencial ou paralelo (Sommerville, 2001).

Algumas das vantagens identificadas pela adoção deste modelo são:

- Suporta a reutilização de transformações;
- É intuitivo de forma que as pessoas pensam o seu trabalho em forma de processamento de *input* e *output*;
- Evoluir o sistema adicionando novas transformações é usualmente um processo simples e rápido;
- É relativamente simples de implementar tanto como um sistema concorrente como sequencial.

4.3 Design Detalhado

Nesta secção será abordado o processo de design de modo mais aprofundado explicitando as alterações previstas na aplicação Easy2rec como também a descrição do funcionamento do serviço responsável pelo licenciamento de utilizadores e de que forma os parceiros o podem consultar. É apresentado também a estrutura de ambos os projetos e de que forma a informação a partilhar entre ambos é estruturada e transmitida.

4.3.1 O Serviço de Licenciamento

O serviço de licenciamento implementado disponibiliza um conjunto de funções identificadas e descritos mais a frente na presente secção que são responsáveis por responder às funcionalidades descritas do serviço de licenciamento. Antes mesmo de definir quais os parâmetros necessários para proceder ao processamento do pedido é de extrema importância definir regras de validação dos dados recebidos de forma a garantir a sua autenticidade. Os atributos *partner*, *request_date* e *validate* deverão estar presentes em todos os parâmetros de entrada de todos os métodos disponibilizados pelo serviço. O parâmetro *partner* representará o identificador único de cada parceiro habilitado a requisitar o *Web Service*. O parâmetro *request_date* como o nome indica deverá conter a data exata do pedido no formato completo. Por fim o campo *validate* deverá conter todos os restantes campos mais a chave privada do parceiro pela mesma ordem que surge no pedido e de seguida todo ele encriptado. Desta forma o atributo *validate* assegura à partida que os pedidos são fidedignos e que provêm de um cliente autorizado para tal. De seguida são descritos os métodos que o serviço deverá disponibilizar e quais as suas responsabilidades.

4.3.1.1 O Método PartnerLicenceRequest

O seguinte método permite ao parceiro/cliente solicitar uma licença com base nas informações do utilizador. Cria uma nova se não existir ou já se a atual se encontrar expirada ou então devolve a licença atual.

Tabela 5 - Parâmetros do método *PartnerLicenceRequest*

Input	Output
partner: string user_mail: string user_name: string request_date: string validate: string	response_code: int response_description: string license_hash: string date_init: string date_end: string

4.3.1.2 O Método GetLicencesCreated

O seguinte método permite ao parceiro solicitar a informação de quantas licenças já foram geradas. É devolvido um número que representa a quantidade de licenças de utilizador requisitadas pelo parceiro em questão.

Tabela 6 - Parâmetros do método *GetLicencesCreated*

Input	Output
partner: string request_date: string validate: string	response_code:int response_description:string number_licences:int

4.3.1.3 O Método SetPartnerPrivateKey

O seguinte método permite ao parceiro modificar a sua chave privada. A chave privada é necessária para garantir a autenticidade dos pedidos efetuados pelo parceiro ao *Web Service*.

Tabela 7 - Parâmetros do método *SetPartnerPrivatekey*

Input	Output
partner:string request_date:string newprivatekey:string validate:string	response_code:int response_description:string

4.3.1.4 O Método GetLicenceData

O seguinte método permite ao parceiro solicitar ao *Web Service* toda a informação referente a uma determinada licença atribuída a um utilizador. Com base no *hash code* da licença é devolvida todos os campos associados a essa licença.

Tabela 8 - Parâmetros do método *GetLicenceData*

Input	Output
partner:string	response_code:int
request_date:string	response_description:string
licence_fingerprint:string	user_mail:string
validate:string	user_name:string
	date_init_string
	date_end:string

4.3.2 A Notificação de Licença

De modo a aplicação Easy2rec receber a licença do utilizador no momento do login o parceiro deverá notificar a aplicação enviando via *JSON* os dados necessários para iniciar sessão na aplicação. Deste modo a forma preconizada será a implementação de um script utilizando a linguagem de programação *Javascript* presente na página do parceiro que envie para a aplicação uma notificação de que o utilizador realizou login com sucesso e que já possui uma licença válida para continuar o processo e iniciar sessão no *software* Easy2rec. Na figura 13 é apresentado um diagrama de sequência simplificado que ilustra as comunicações entre os vários componentes para realizar a autenticação, licenciamento e notificação da licença para um determinado utilizador.

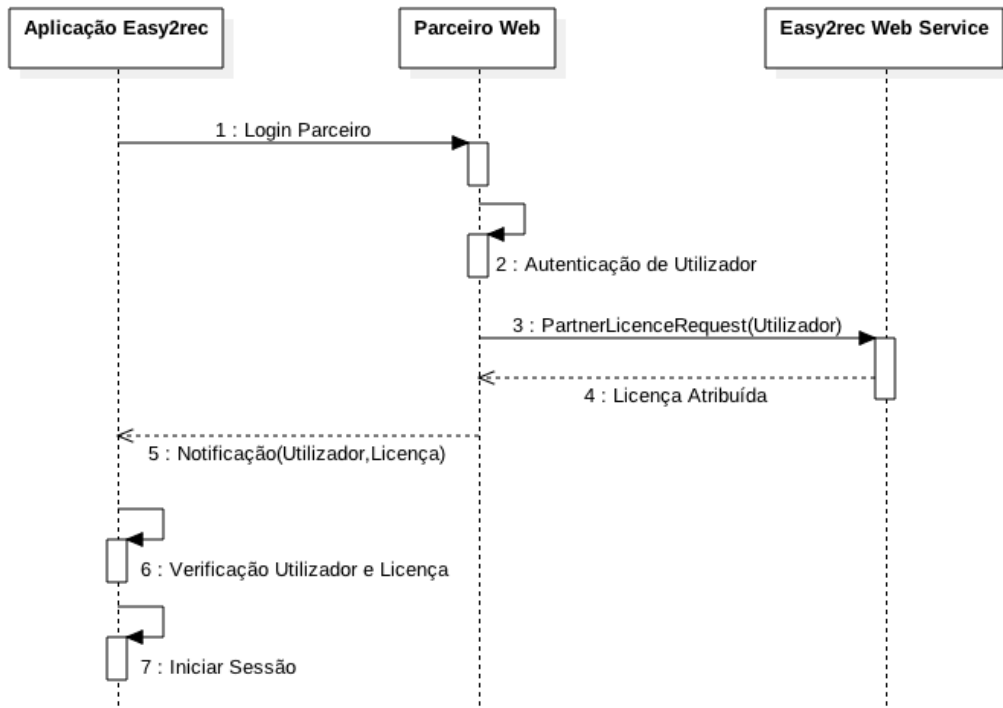


Figura 13 - Diagrama de sequência das comunicações entre componentes

4.3.3 A Submissão de Ficheiros

Na secção que se segue é descrita de que forma os ficheiros de vídeo e de metadados gerados pela aplicação são organizados, onde são guardados e finalmente de que forma são submetidos na plataforma Educast.

4.3.3.1 Estrutura de Ficheiros

Os ficheiros de vídeo gerados pelas gravações dos utilizadores devem ser organizados numa estrutura bem definida entre o *software* de envio e o parceiro. Os vários ficheiros correspondentes a uma gravação devem ser agrupados num só ficheiro compactado antes de serem enviados para a plataforma do parceiro. Devem também ser acompanhados por um ficheiro estruturado de metadados indicando quais os ficheiros que vão ter enviados. De seguida na figura 14 é apresentada a estrutura de ficheiros e indentificadores a adotar na denominação dos ficheiros. Representa-se pelas variáveis `{selected_channel_id}` e `{selected_video_id}` o identificador do canal Educast selecionado pelo utilizador e o identificador do vídeo a submeter.

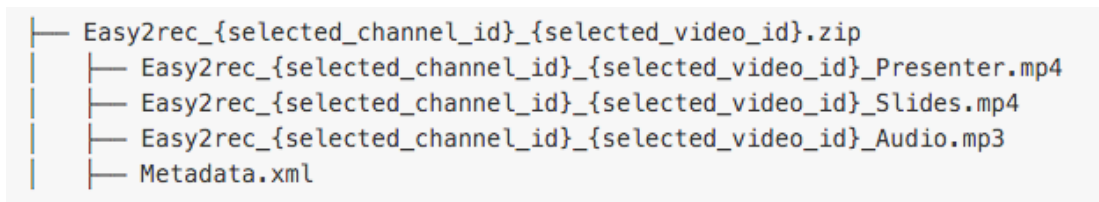


Figura 14 - Diagrama de seqüência das comunicações entre componentes

4.3.3.2 Estrutura de Metadados

O ficheiro de metadados a submeter juntamente com os ficheiros de vídeo capturados deve conter todas as informações disponíveis sobre o utilizador e a gravação em questão. De seguida é apresentada na figura 15 a estrutura XML a incluir no ficheiro de metadados com a descrição dos principais elementos identificados.

```
<video>
  <id>Representa o identificador do vídeo</id>
  <username>Representa o nome de utilizador</username>
  <type>Representa o tipo de vídeo: exemplo Presenter ou Slides</type>
  <filename>Representa o nome do ficheiro</filename>
  <channel_id>Representa o identificador do canal selecionado</channel_id>
  <date_transcoded>Representa a data de processamento do ficheiro</date_transcoded>
  <date_uploaded>Representa a data de submissão</date_uploaded>
  <date_recorded>Representa a data de gravação</date_recorded>
  <duration>Representa a duração do ficheiro</duration>
</video>
```

Figura 15 – Descrição do ficheiro estruturado metadata.xml

4.4 A Base de Dados

A base de dados que é responsável por suportar toda a informação necessária para o correto funcionamento do sistema de licenciamento engloba cinco entidades/tabelas, identificadas na figura 16 apresentada de seguida.

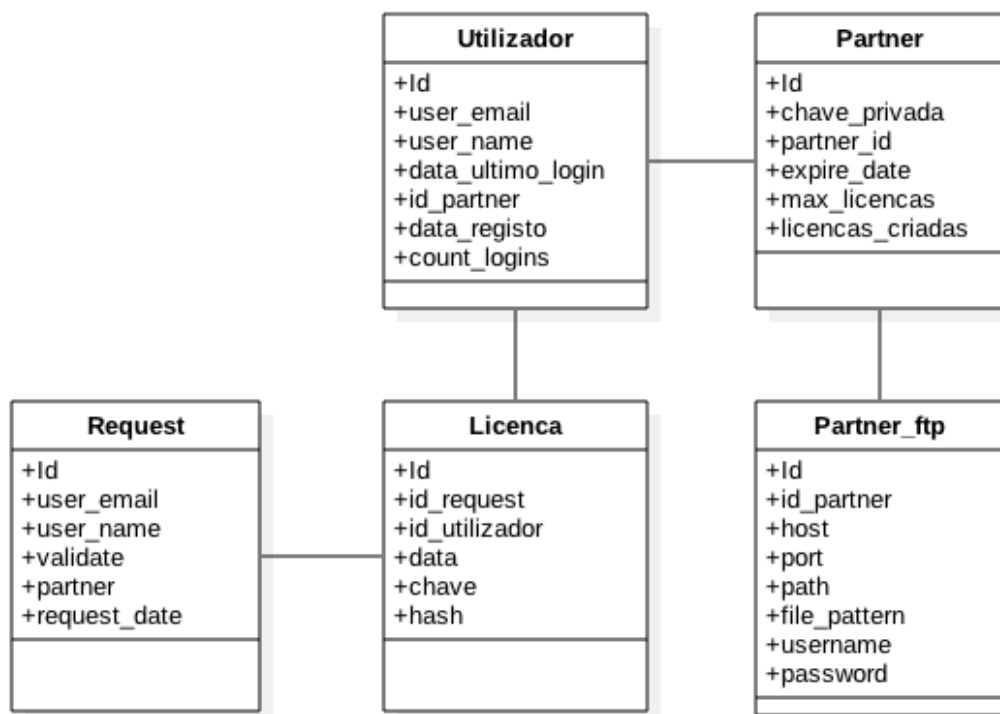


Figura 16 - Diagrama de Base de Dados

A tabela **Request** representa os pedidos de licenciamento recebidos pelo *Web Service*. Todos os pedidos recebidos mesmo que não válidos são guardados na base de dados para deteção de possíveis problemas.

A tabela **Utilizador** representa os utilizadores do parceiro a que vai ser associada a licença a gerar, guardando todas as informações obtidas.

A tabela **Licença** representa a licença de utilização criada. Uma licença quando é gerada é atribuída ao utilizador e ao *request* recebido.

A tabela **Partner** representa o parceiro que efetua os pedidos ao *Web Service*. O *partner* deve ser previamente introduzido na base de dados. A validação dos pedidos depende diretamente se o *partner* existe e se o número de licenças criadas não exceder o número de licenças limite.

A tabela **Partner_ftp** representa os dados de acesso ao servidor FTP do parceiro para envio dos ficheiros de vídeo gerados pelo *software*.

4.5 A Interface Gráfica

De acordo com Somerville (2001, p. 330) toda a interface com o utilizador deve ter em conta as capacidades do público-alvo do produto a desenvolver. Existem alguns conceitos que se aplicados podem facilitar a tarefa dos utilizadores a quando do uso do produto. Alguns dos conceitos mais importantes são:

- **Familiaridade** – A interface deverá usar termos e conceitos que são facilmente identificáveis pela experiência anterior dos utilizadores que irão utilizar mais o produto;
- **Consistência** – A interface deverá ser consistente em todos os pontos, quando possível, operações comparáveis devem ser realizadas da mesma forma;
- **Mínima surpresa** – Os utilizadores não devem ser surpreendidos pela forma como o produto é apresentado;
- **Orientação** – A interface deverá sempre que possível guiar o utilizador na tarefa a realizar e caso ocorram erros deve ser fornecida informação de modo a auxiliar o utilizar na sua resolução;

A interface com o utilizador do *software* Easy2rec sofrerá alterações de modo a suportar as alterações e as novas funcionalidade adicionadas. As principais alterações são definidas de seguida evidenciado os aspetos fundamentais presentes nestas alterações.

A seguinte figura ilustra a primeira janela com que o utilizador se depara quando inicia o *software*. A janela deverá separar o login de utilizador dito normal com uma conta Easy2rec e o login do utilizador através de uma de parceiro. Neste ponto a separação de tipo de login será a principal modificação ao que já se encontra implementado.

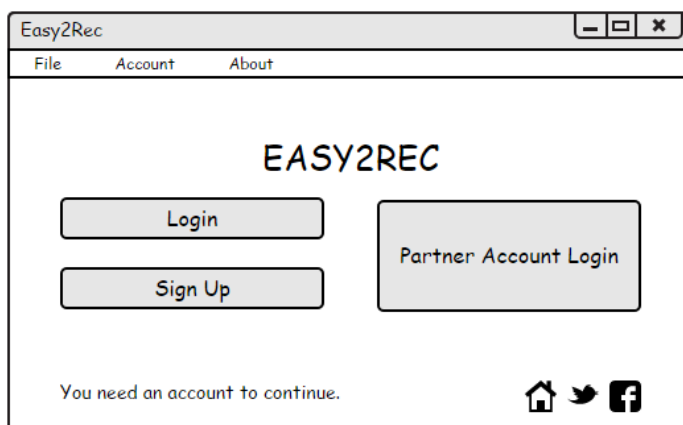


Figura 17 - Janela inicial de Login

Quando o utilizador seleciona a opção “*Partner Account Login*” deverá ser apresentada uma lista de parceiros autorizados como exemplificado de seguida, que este pode efetuar o respetivo login. O utilizador poderá selecionar o parceiro desejado para continuar com o processo.



Figura 18 - Escolha de Parceiro de Login

Depois do utilizador selecionar o parceiro pretendido para efetuar o *login*, este deverá ser encaminhado para o sistema de autenticação do parceiro. Esta integração deverá ser implementada através da incorporação de um *web browser* que acede ao Sistema externo do parceiro para continuar com o processo. O sistema é gerido unicamente pelo parceiro.

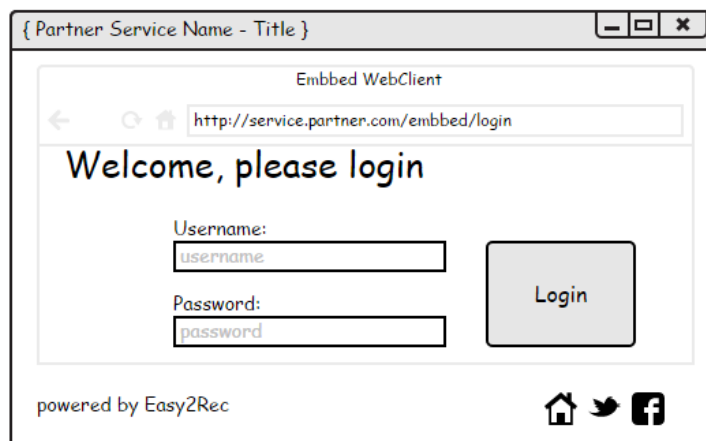


Figura 19 - Interface com o sistema de autenticação externo do parceiro

Assim que o utilizador realizar o seu login com sucesso junto do parceiro e o este solicitar ao *Web Service* de licenciamento uma nova licença, o utilizador deverá receber uma mensagem no software que indique o sucesso ou o insucesso da operação. Esta comunicação deverá ser implementada através do envio de uma notificação para o *software* Easy2rec a partir do parceiro. Este processo irá ser desenvolvido em maior detalhe mais a frente neste documento.

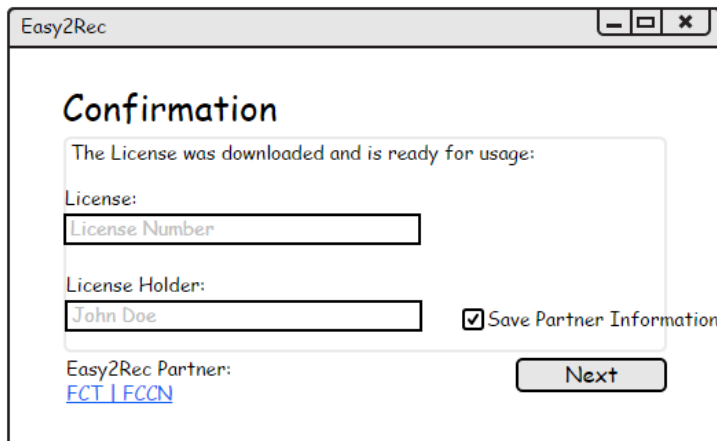


Figura 20 - Confirmação de login e licença de utilizador válida

A janela apresentada de seguida representa o menu principal de um utilizador com a sua sessão iniciada. Identificamos rapidamente as principais funcionalidades como realizar uma nova gravação e visualizar as gravações realizadas previamente. É apresentada um lembrete ao utilizador de quantos vídeos previamente gravados se encontram ainda por submeter no portal do parceiro.

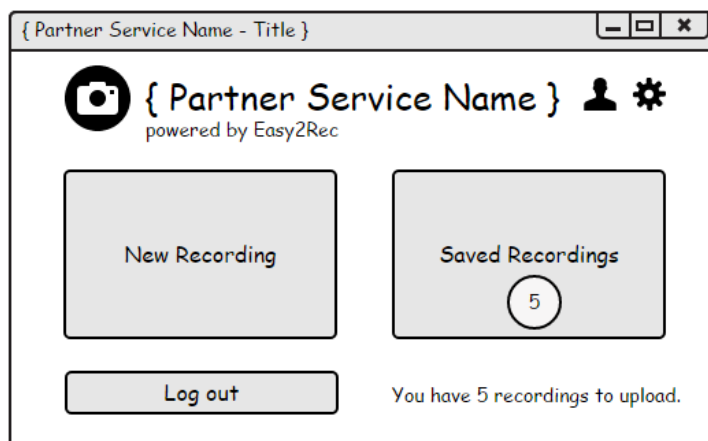


Figura 21 - Janela de Menu Principal

A seguinte janela representa a listagem de todos os vídeos gravados pelo utilizador. Por cada vídeo é apresentado algumas informações como a data de gravação e o nome/título. Para cada vídeo deverá ser possível realizar algumas ações tais como submeter e apagar um determinado vídeo. O utilizador pode também selecionar um conjunto de vídeos e submeter esse conjunto de vídeo em fila. A lista facilitará a tarefa do utilizador na gestão do seu trabalho.

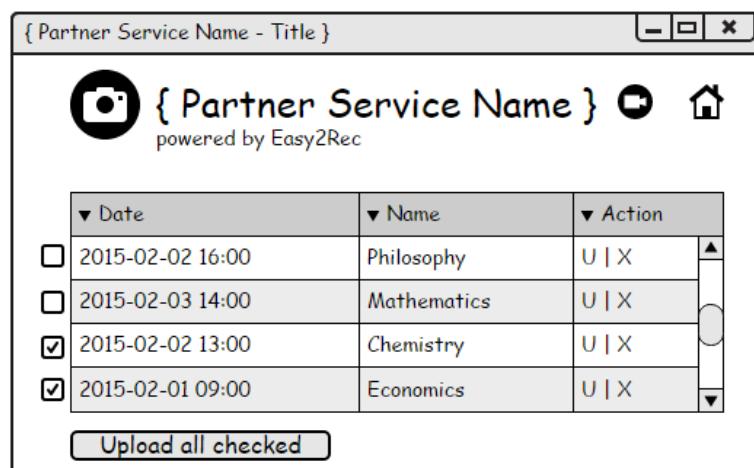


Figura 22 - Lista de gravações realizadas

Por cada vídeo aula a enviar para o portal do parceiro é necessário que o utilizador preencha e submeta os metadados referentes ao vídeo. Metadados esses definidos num formulário pelo parceiro. Os metadados irão ser associados ao vídeo no portal.

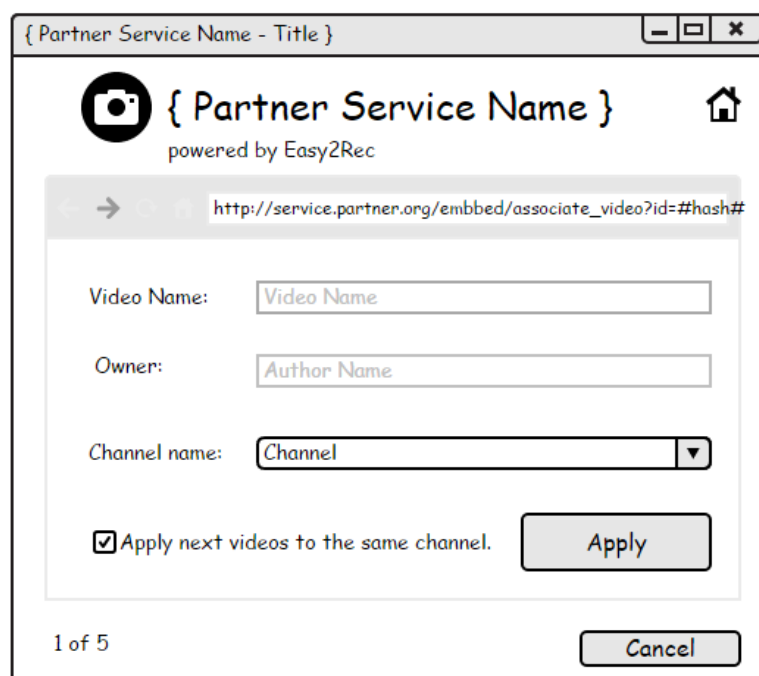


Figura 23 - Formulário externo de submissão de metadados

A seguinte janela representa a interface com o utilizador no momento do processamento e envio dos ficheiros para o portal do parceiro. É apresentado em todos os momentos a percentagem de trabalho realizado. Assim que o processo terminar o utilizador deverá ser notificado do sucesso ou insucesso da tarefa. Caso os vídeos sejam enviados com sucesso o utilizador pode se dirigir ao portal do parceiro e verificar que os vídeos foram submetidos.

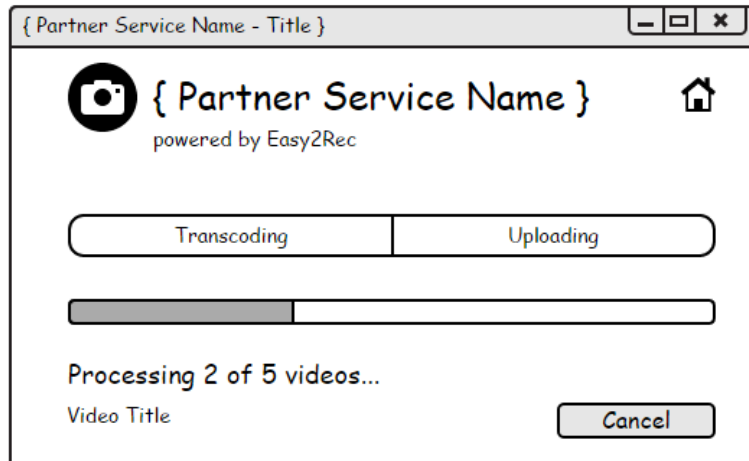


Figura 24 - Janela de processamento e *upload* dos vídeos

5 Implementação

No presente capítulo será abordado todo o processo de desenvolvimento e implementação dos novos componentes a integrar na aplicação Easy2rec já existente e o serviço de licenciamento que foi desenvolvido de raiz. Para mais informações sobre a implementação da aplicação Easy2rec inicial pode ser consultada a documentação no documento apresentado na disciplina de PESTI com o título “EduGraal – Solução para produção e broadcast de vídeo-aulas” apresentado nesta mesma instituição no ano letivo 2011-2012.

5.1 A Base de Dados

A base de dados criada suporta toda a informação relevante para o sistema de licenciamento a utilizadores provenientes dos parceiros. Foi usada a linguagem *SQL* para a criação da estrutura de dados. Este processo engloba a implementação do modelo conceptual, o modelo lógico, físico e dos mecanismos de segurança. No modelo físico a escolha pela tecnologia recaiu sobre *SQL Server* devido à base de dados já existente se encontrar alojada sobre essa tecnologia. A criação da estrutura foi realizada através de *scripts SQL*.

Na figura 25 apresentada de seguida é demonstrado como exemplo o script utilizado para a criação da *tabela utilizador_partner*.

```

CREATE TABLE [dbo].[utilizador_partner] (
  [id] INT IDENTITY ( 1, 1 ) NOT NULL,
  [user_email] VARCHAR( 100 ) NOT NULL,
  [user_name] VARCHAR( 100 ) NOT NULL,
  [data_ultimo_login] DATETIME NOT NULL,
  [id_partner] INT NOT NULL,
  [data_registro] DATETIME NOT NULL,
  [count_logins] INT DEFAULT '((0))' NOT NULL,
  PRIMARY KEY ( [id] ) )
GO

```

Figura 25 – SQL de criação da tabela *utilizador_partner*

Criaram-se não só todas as tabelas como também as relações existentes entre as mesmas, mas também todos os restantes mecanismos essenciais à base de dados, nomeadamente: chaves primárias, chaves estrangeiras, índices únicos e constrangimentos de atributos. Posteriormente será realizado o mapeamento da base de dados no módulo de acesso aos dados no *Web Service* de licenciamento que irá tratar diretamente os dados.

5.2 O Serviço de Licenciamento

As funcionalidades de gestão e de licenciamento de utilizadores ao nível do parceiro são implementadas no serviço *online* que disponibiliza um conjunto de funções que podem ser consumidas pelos mesmos. Desta forma permite que duas aplicações se comuniquem entre si. O serviço foi implementado através de tecnologia de *Web Services* disponibilizada pela Microsoft tirando partido da *Framework.NET*. Na figura 24 é apresentada a estrutura de componentes em que se subdivide o projeto *Web Service*. São três os módulos que o constituem e que tem responsabilidades distintas, são elas o *webservicee2r.dll* que representa o *Web Service* onde são definidos os métodos e a sua implementação em termos de funcionalidades disponibilizadas, a camada de acesso aos dados representada na figura 26 por *DM_API.dll* e os componentes externos onde se destaca o *SKGL.dll* que é a dependência responsável pela criação de chaves de licença.

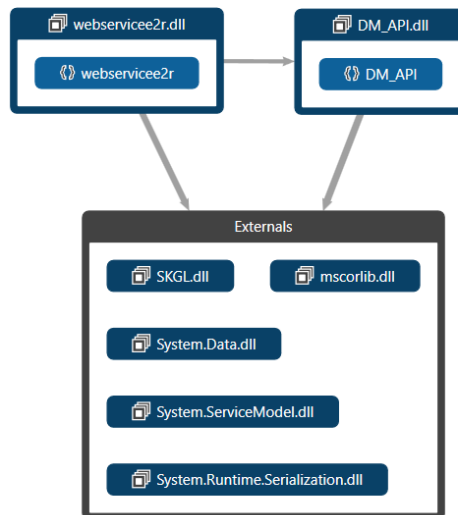


Figura 26 – Estrutura de componentes no serviço de licenciamento

5.2.1 A Camada de Acesso a Dados

A comunicação com a base de dados é realizada através da implementação de uma camada isolada com essa única responsabilidade. Cada camada possui um propósito bem definido e a de acesso a dados inclui os componentes que permite a aplicação ter acesso à base de dados, fornecendo a abstração necessária da estrutura de base de dados à aplicação. Desta forma qualquer alteração realizada na estrutura de base de dados não afeta diretamente a restante aplicação.

Todos os pedidos de acesso aos dados são executados por esta camada e as respostas são também devolvidas pela mesma. A base de dados não é acedida diretamente por mais nenhuma camada da aplicação. Apresenta a principal vantagem de permitir uma maior flexibilidade de evolução e manutenção dos modelos pois o projeto tem uma enorme margem de progressão e a evolução deste pode ser mais facilmente adaptável no futuro.

Na figura 27 é apresentado o diagrama de componentes do serviço de licenciamento focando na secção que implementa a camada de acesso a dados.

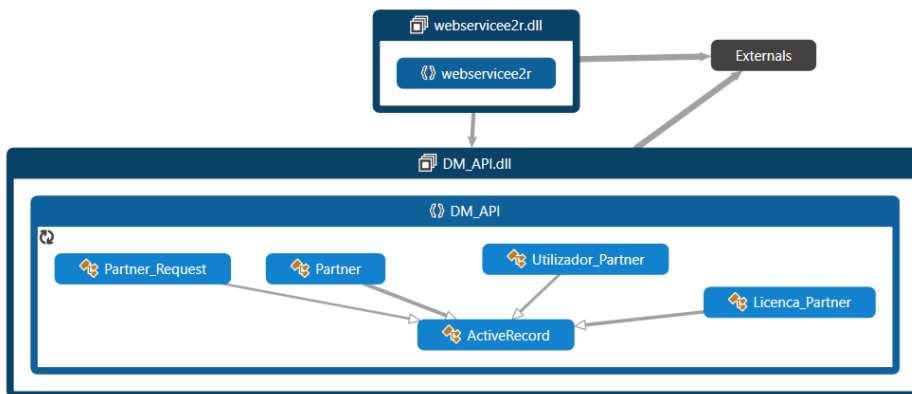


Figura 27 – Estrutura de componentes e classes de acesso a dados

A camada de acesso a dados no diagrama é representada na figura por *DM_API.dll* onde estão presentes as seguintes classes:

- **Partner_Request** que implementa a tabela da base de dados de nome *Request* onde são guardados todos os registos de pedidos efetuados ao serviço;
- **Partner** que implementa as tabelas da base de dados denominadas por *Partner* e *Partner_ftp* onde estão descritos todos os registos de parceiros habilitados a consumir o serviço juntamente com os dados necessários para a submissão de ficheiros via *FTP* posteriormente;
- **Utilizador_Partner** que implementa a tabela da base de dados denominada por *Utilizador* que é responsável por todos os registos de utilizadores com login válido provenientes do parceiro;
- **Licenca_Partner** que implementa a tabela da base de dados de nome *Licenca* que é responsável por todos os registos de licenças de utilização atribuídos a utilizadores provenientes de parceiros;

Na figura 28 é apresentado o diagrama de classes da camada de acesso a dados e neste diagrama é apresentado o mapeamento das tabelas de base de dados para as classes que realizam o seu acesso na aplicação.

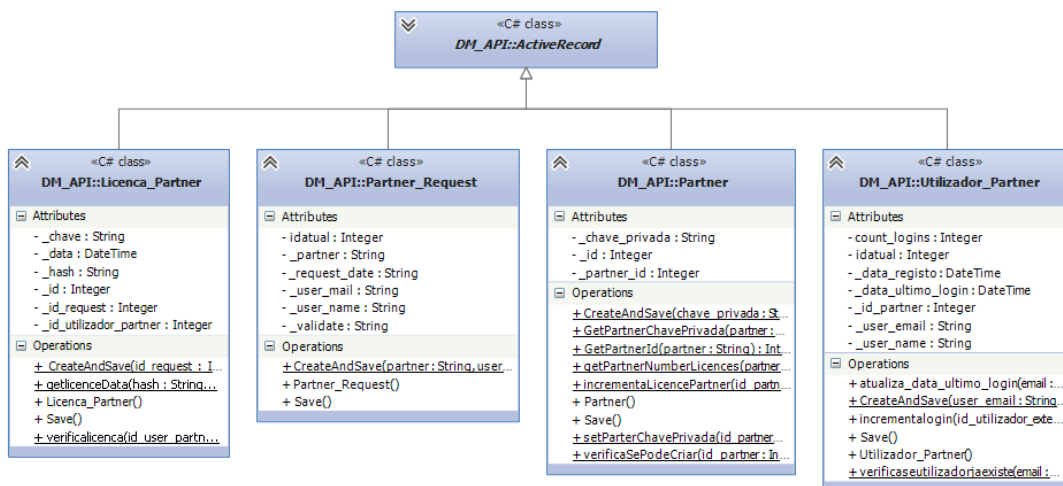


Figura 28 – Diagrama de classes de camada de acesso a dados

Todas as classes implementadas contêm os métodos construtores, de *store* na base de dados e de acesso correspondentes. De forma complementar foram implementadas algumas funções necessárias para dar resposta a um conjunto variado de funcionalidades que a aplicação necessita de responder como por exemplo:

- A função **getLicenceData** na classe **Licenca_Partner** é responsável por retornar toda a informação sobre uma determinada licença através da pesquisa pelo seu *hash* que a identifica publicamente;
- A função **verificaSePodeCriar** na classe **Partner** é responsável por verificar se o parceiro cumpre nesse momento todas as condições para gerar nova licença para um determinado utilizador;
- A função **verificaLicenca** na classe **Licenca_Partner** é responsável por verificar se o utilizador fornecido na chamada de licenciamento, possui uma licença ativa nesse momento e se sim retorna os dados da correspondente licença ativa.

5.2.2 O Web Service

Neste módulo são definidos dois papéis: o cliente e o servidor. O servidor disponibiliza um conjunto de funcionalidades e o cliente faz o seu uso. O cliente envia pedidos ao servidor, este por sua vez ao recebê-las toma a decisão de aceitá-las e processa-las ou não de acordo com as regras definidas no servidor. Neste ambiente o servidor será o *Web Service* implementado e o cliente é o parceiro que irá consumir o serviço. Nesta arquitetura o servidor não armazena nenhuma informação de contexto. Toda a informação necessária para processar um determinado pedido deve estar contida nele mesmo. Desta forma torna o servidor mais simples, pois ele não precisa levar em consideração o contexto atual para tomar decisões, toda informação necessária será enviada em cada pedido (Fielding, 2000).

O documento *XML* de descrição do serviço denominado por *WSDL* pode ser consultado acedendo ao seguinte endereço público <http://api.easy2rec.com/PartnersService.svc?wsdl>. A interface pública descreve todas as funções disponibilizadas pelo serviço, quais os parâmetros de *input* e *output* como também os seus tipos de dados. O cliente acedendo a este documento contém toda a informação necessária para consumir o serviço. O serviço encontra-se publicado no seguinte endereço <http://api.easy2rec.com/partnersservice.svc>. Na figura 29 é apresentada a estrutura de classes e métodos disponíveis do componente *Web Service*.

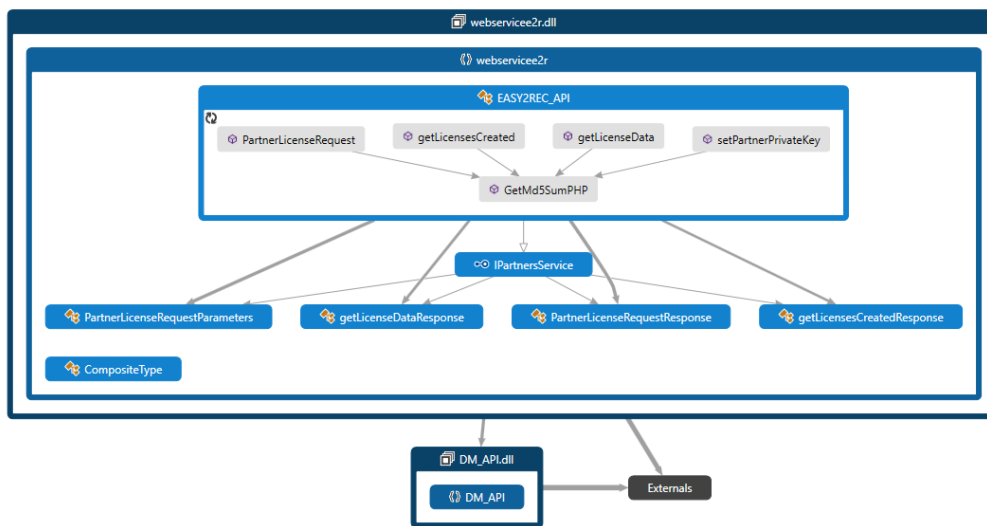


Figura 29 – Estrutura de classes e métodos presentes no *Web Service*

Na figura anterior podemos verificar que dentro da implementação do *EASY2REC_API* são disponíveis os seguintes métodos públicos:

- **PartnerLicenseRequest** permite criar uma nova chave de licença para um determinado utilizador;
- **GetLicencesCreated** permite receber um pedido que devolve ao cliente o número de licenças criadas por um determinado parceiro devidamente identificado;
- **GetLicenceData** permite ao cliente receber todas as informações disponíveis sobre uma determinada licença através do seu identificador *hash*;
- **SetPartnerPrivateKey** permite ao cliente alterar a sua chave privada, necessária para construir os pedidos com sucesso garantindo a sua autenticidade.

Todas as funções disponibilizadas no serviço possuem ligação com a função *GetMd5SumPhp* que implementa o algoritmo de encriptação de mensagens de texto do tipo *String*. De seguida na figura 30 é apresentado um exemplo de uma chamada a função que encripta os parâmetros recebidos na função *PartnerLicenseRequest* e que compara com o parâmetro

validate também recebido no pedido. Unicamente após a verificação ter sido executada com sucesso é que o pedido efetivamente é processado.

```
GetMd5SumPHP(composite.Partner + composite.User_mail  
+ composite.User_name + composite.Request_date +  
DM_API.Partner.GetPartnerChavePrivada(DM_API.Partner.GetPartnerId(composite.Partner)))  
.Equals(composite.Validate)
```

Figura 30 – Exemplo de verificação do parâmetro *validate*

5.2.3 Os Códigos de Resposta

Com base nos padrões de respostas a pedidos web *HTTP* foram adotados e adaptados alguns dos códigos tendo por base as respostas padronizadas²⁶. Desta forma adaptamos os códigos documentados já existentes e adaptamo-los às nossas necessidades. No entanto estes códigos descritos de seguida unicamente são referentes à resolução e processamento da funcionalidade em questão e não à transmissão do pedido *HTTP* efetivo. Todas as descrições aqui definidas foram previamente discutidas com o parceiro e consequentemente aprovadas antes de se proceder à sua implementação.

Tabela 9 – Códigos e descrições de resposta a pedidos ao *Web Service*

Códigos de Resposta	
Código	Descrição
400	"Invalid data"
403	"Partner exceeded quota"
404	"Partner not found"
200	"Licence still active"
200	"Licence data returned successfully"
200	"Number of licences created returned successfully"
201	"New licence created"
201	"New licence created for expired licence"

5.2.4 A Autenticação e Segurança

O acesso aos métodos disponibilizados pelo *Web Service* necessita de ser controlado para evitar que pedidos não autorizados não sejam satisfeitos. Para consumir o serviço o parceiro necessita de obter previamente os dados de acesso constituídos por:

- Identificador do parceiro do tipo de dados *String*;
- Chave privada do tipo de dados *String*.

²⁶ Padrão de respostas HTTP descrito em <https://support.microsoft.com/pt-br/kb/173971>

Ambos necessitam de ser definidos antes do parceiro realizar qualquer chamada ao serviço. Desta forma é controlado o acesso indevido e é controlado de uma forma mais clara toda a atividade realizada por terceiros.

Outra preocupação tida em conta durante o desenvolvimento foi assegurar que a informação não era adulterada durante a comunicação entre as partes. Com esse objetivo em mente foi acordado um método de encriptação com a equipa do cliente. A escolha recaiu sobre o método *MD5* como já foi exemplificado o seu uso na secção anterior. Todos os pedidos efetuados ao serviço devem conter um campo adicional no final do pedido denominado por validade. Este campo é constituído por todos os parâmetros de input na função juntamente com a chave privada do parceiro. A chave nunca é transmitida como parâmetro nas chamadas as funções desta forma é garantida a autenticidade do pedido validando estes parâmetros antes do serviço responder efetivamente.

```
Validate = md5(partner+user_mail+user_name+request_date+private_key)
```

Figura 31 – Exemplo de criação do parâmetro *validate*

5.3 A Aplicação Easy2rec

Sendo o software um produto já existente no mercado todas as principais características do produto como um todo foram tidas em conta no momento da análise ao problema, desenvolvimento e integração das novas funcionalidades, como por exemplo o *design* e *workflow* já existente da aplicação. Deste modo todas as novas funcionalidades foram desenhadas de modo a integrar a aplicação de uma forma não intrusiva e que se misturassem nativamente com a aplicação já existente.

Esta secção serve então para descrever todas as alterações implementadas descrevendo também os seus objetivos não só de um ponto de vista funcional como de utilidade para os utilizadores.

5.3.1 A Autenticação

A figura 32 demonstra a página apresentada a um utilizador não autenticado introduzindo uma nova opção de login “*Entrar com parceiro*”, que permite uma alternativa para iniciar sessão na aplicação. Esta opção irá reencaminhar para uma listagem de parceiros aderentes ao Easy2rec e que permitem os seus utilizadores o usufruto da aplicação.



Figura 32 – Janela inicial para realizar o login

A figura 33 apresenta a listagem de todos os parceiros disponíveis e integrados com a aplicação. Neste momento apenas o parceiro Educast se encontra completamente integrado, no entanto o desenvolvimento realizado para este parceiro específico permite também que posteriormente novos parceiros possam ser introduzidos de uma forma relativamente fácil e com uma taxa de esforço reduzida devido ao trabalho ter sido realizado com esse objetivo a médio, longo prazo bem presente.

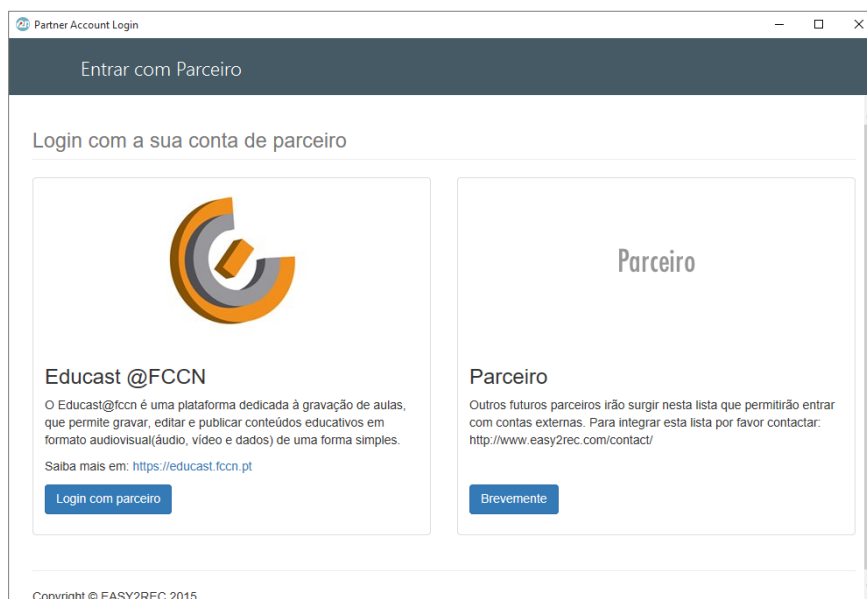


Figura 33 – Janela de seleção do parceiro para login

A figura 34 ,35 e 36 apresentam a sequencia de ecrãs da responsabilidade do parceiro, neste caso o Educast que implementam a autenticação federada dos seus utilizadores antes de se encaminhar novamente para a aplicação de modo a finalizar o processo de login. Todas as janelas são apresentadas ao utilizador dentro da aplicação Easy2rec através da incorporação de um *web browser*, no entanto todo o conteúdo apresentado é inteiramente da responsabilidade do parceiro.

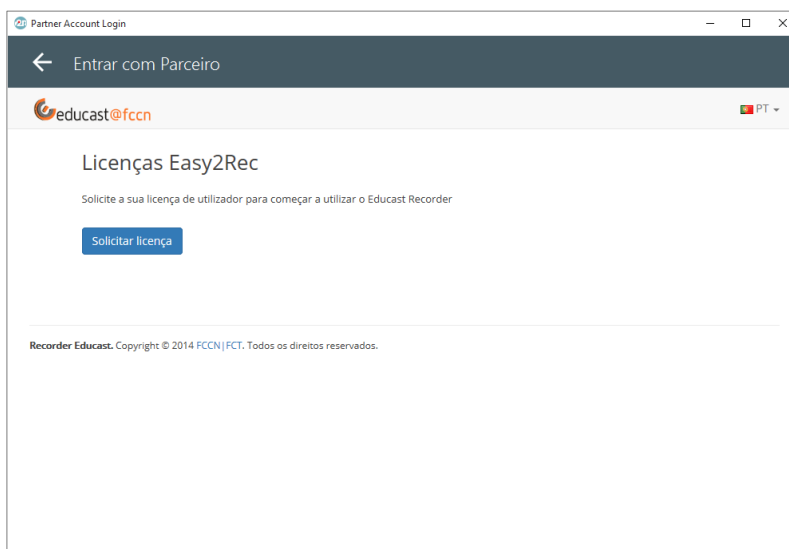


Figura 34 – Janela de solicitação de licença ao parceiro

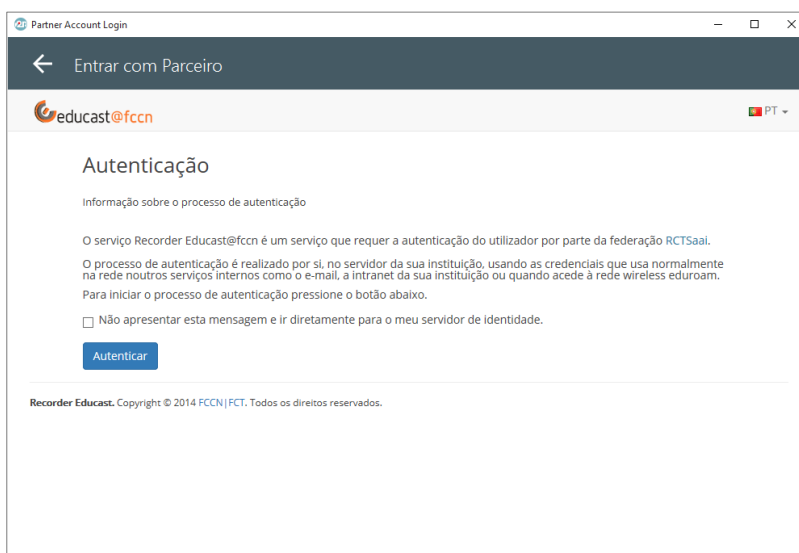


Figura 35 – Informações de autenticação no serviço do parceiro



Figura 36 – Ambiente de autenticação federada controlada pelo parceiro

Após todo o processo de autenticação ter sido concluído pelo utilizador junto do parceiro e este ter solicitado a respetiva licença ao serviço de licenciamento do Easy2rec, encontra-se agora em condições de voltar à aplicação de modo a continuar e finalizar o processo de *login*.

O parceiro neste momento deverá notificar a aplicação que a autenticação foi realizada com sucesso, enviando os dados do utilizador e a licença atribuída para a aplicação, de modo a esta verificar os dados recebidos. Se validados com sucesso o utilizador é autenticado localmente e pode a partir deste momento usufruir de todas as funcionalidades do *software*.

O processo de notificação e os dados recebidos irão ser descritos mais a frente neste documento na secção dedicada a este processo. Na figura 37 é apresentada a mensagem de autenticação mostrada ao utilizador após a validação com sucesso dos dados de autenticação recebidos pela aplicação.

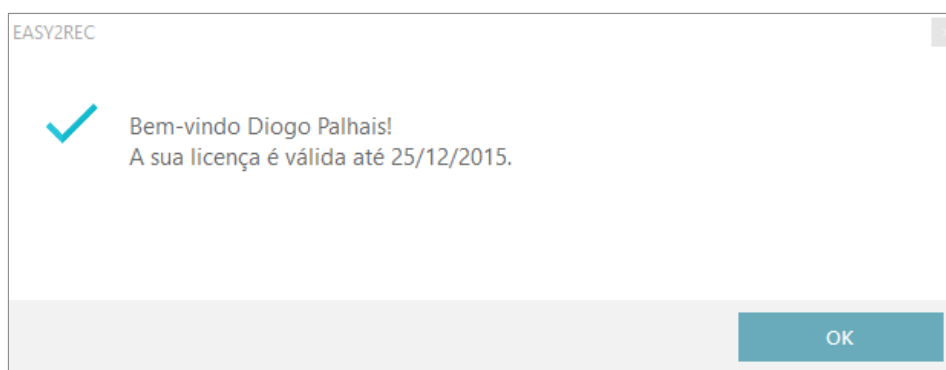


Figura 37 – Mensagem de autenticação e licença válida na aplicação

De seguida na figura 38 é apresentado o diagrama de sequência do processo de autenticação desde o momento em que o utilizador seleciona a opção “*entrar com parceiro*” até receber a notificação de autenticação com sucesso. É possível perceber um pouco melhor quais os papéis desempenhados por cada entidade e quais as suas responsabilidades. Mais a frente neste documento irá ser descrito a forma como a aplicação Easy2rec é notificada pelo parceiro.

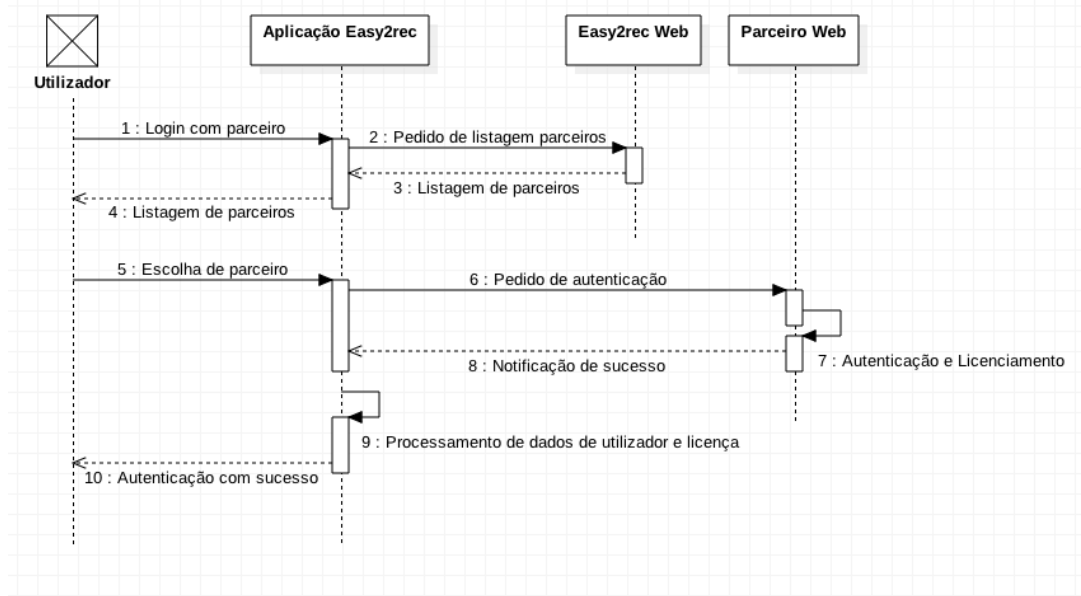


Figura 38 – Diagrama de sequência de sistema do processo de autenticação

5.3.2 A Gravação

Ultrapassado o processo de autenticação externo e já com toda a informação necessária carregada na aplicação o utilizador pode a partir deste momento iniciar as suas gravações. Na figura 39 é apresentado o ecrã principal após efetuado o login com sucesso. Neste ecrã pode iniciar uma nova gravação, ver a listagem de gravações já efetuadas, configurar alguns aspetos do software e realizar o *logout* da sua conta de utilizador. São apresentadas também informações sobre parceiro como por exemplo o logotipo e indicação do nome de utilizador com a conta iniciada nesse momento.



Figura 39 – Janela de menu principal quando autenticado por parceiro

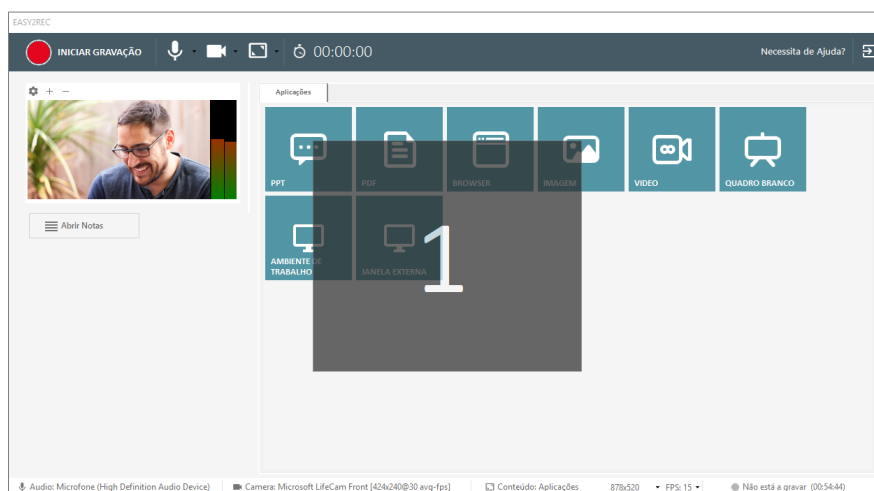


Figura 40 – Ambiente de gravação do Easy2rec

Na figura 40 é apresentado o ambiente de gravação onde os utilizadores realizam as suas gravações. A interface é bastante intuitiva, selecionando na barra de ferramentas superior quais os dispositivos que desejam capturar. O utilizador pode selecionar de entre várias combinações, tais como capturar a câmara de vídeo do apresentador, vídeo da área de conteúdos e também o áudio. Todos eles capturados de forma sincronizada.

Após o termino de uma gravação é apresentado ao utilizador a mensagem presente na figura 41, que lhe permite encaminhar imediatamente para o processo de submissão ou pode também escolher “OK, continuar” que encaminhará para a listagem de gravações onde pode realizar a submissão posteriormente quando assim o desejar. Todas as gravações são guardadas localmente e podem ser acedidas posteriormente.

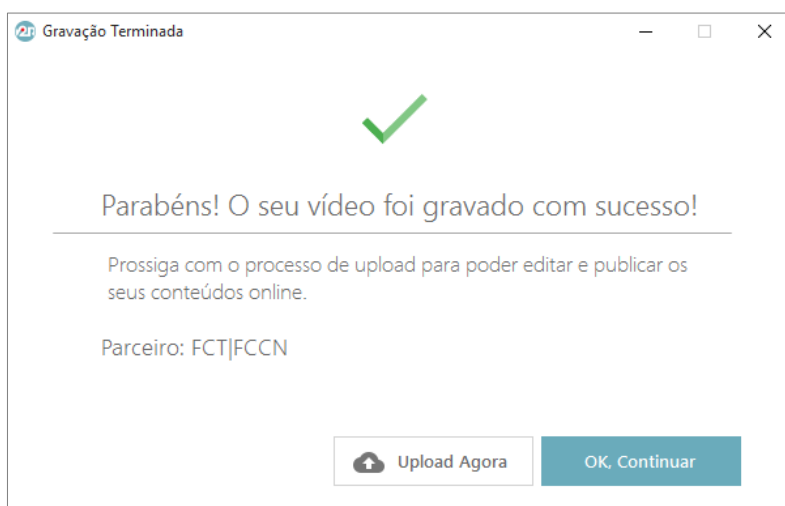


Figura 41 – Mensagem de gravação realizada com sucesso

O utilizador pode visualizar uma listagem de todas as gravações efetuadas. Existem dois separadores onde agrupam os tipos de ficheiros existentes:

- **Vídeos por carregar** apresenta ao utilizador todas as gravações que se encontram por submeter para o servidor do parceiro;
- **Vídeos já carregados** apresenta a utilizador todas as gravações que, até à data, já foram submetidas no portal do parceiro, no entanto podem ser enviados novamente substituindo os ficheiros enviados anteriormente.

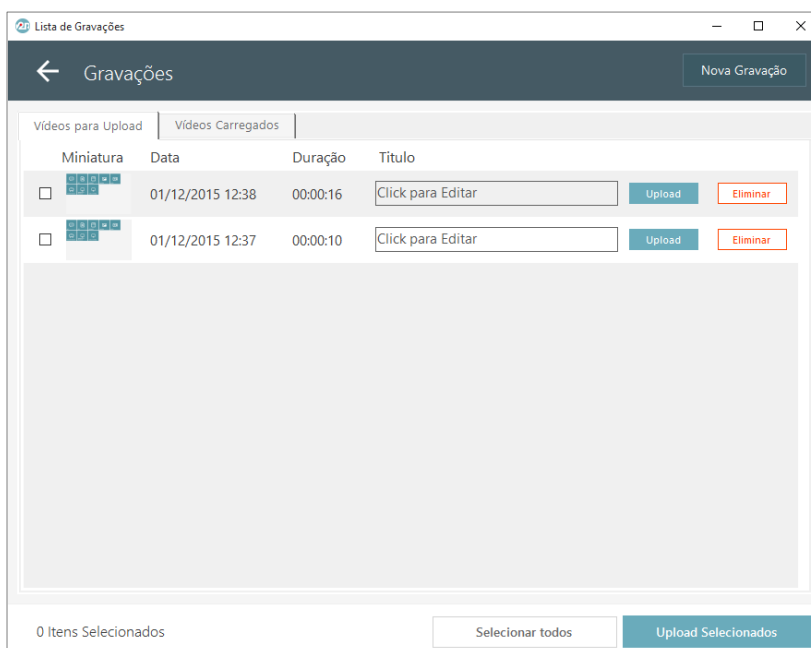


Figura 42 – Lista de gravações realizadas pelo utilizador

A informação e o estado das gravações apresentadas na listagem da figura 42 são carregadas do ficheiro *videodata.xml* referente ao utilizador com sessão iniciada na aplicação. O ficheiro é

responsável por guardar toda a informação das gravações. O ficheiro é estruturado no formato XML e segue a seguinte estrutura:

- **<Videos>** representa a raiz que contem todos os vídeos do utilizador;
 - **<Video>** representa uma gravação por submeter para a plataforma parceiro;
 - **<ID>** Identificador da gravação;
 - **<User>** Localização física do ficheiro de gravação do apresentador;
 - **<Content>** Localização física do ficheiro de gravação do conteúdo;
 - **<Audio>** Localização física do ficheiro de gravação áudio;
 - **<Date>** Data e hora do término da gravação;
 - **<Thumbnail>** Localização física do ficheiro de imagem *thumbnail*;
 - **<Duration>** Duração da gravação no formato hh:mm:ss;
 - **<Title>** Título da gravação;
 - **<VideoUploaded>** representa uma gravação que já se encontra submetida com sucesso na plataforma do parceiro;
 - Todos os atributos definidos na *tag video* encontram-se também presentes na *tag VideoUploaded* e apresentam a mesma descrição.
 - **<UploadedDate>** Data e hora do término da submissão;

```
<?xml version="1.0" encoding="utf-8"?>
<Videos>
  <Video>
    <ID>924654e9-42d2-4b32-aad0-a6d687a2ad42</ID>
    <User>{userpath}\temp\56d05eb8-fc04-4524-b99e-5d178b18e62b.avi</User>
    <Content>{userpath}\temp\e4281245-4bfb-4f1b-a4c4-1728b86ba699.avi</Content>
    <Audio>{userpath}\temp\b0c6a880-ffcb-4e4c-9a4b-b5983b9f5a74.wav</Audio>
    <Date>19/08/2016 21:08</Date>
    <Thumbnail>{userpath}\temp\cbfc177b-0472-4584-8232-6b12be2b2e8b.PNG</Thumbnail>
    <Duration>00:06:23</Duration>
    <Title>Titulo Aula 2</Title>
  </Video>
  <VideoUploaded>
    <ID>a16531bf-6858-4f18-987b-6b47bedd4aba</ID>
    <User>{userpath}\temp\a613a1ca-9188-461e-afb9-b52e02f65e69.avi</User>
    <Content>{userpath}\temp\4736b1b8-c523-46ac-af41-b7cce2da7df5.avi</Content>
    <Audio>{userpath}\35295bc0-b8d1-4a30-901e-ad087b57bd44.wav</Audio>
    <Date>19/08/2016 19:39</Date>
    <Thumbnail>{userpath}\temp\fbfeab43-afc1-412d-8ef8-97e316e062b8.PNG</Thumbnail>
    <Duration>00:12:08</Duration>
    <Title>Titulo Aula</Title>
    <UploadDate>19/08/2016 19:41</UploadDate>
  </VideoUploaded>
</Videos>
```

Figura 43 – Ficheiro estruturado de metadados sobre as gravações

5.3.3 A Submissão

Antes de cada submissão de cada gravação para a plataforma do parceiro, é necessário preencher um formulário de metadados que contém por exemplo o título do vídeo e o canal para onde deseja publicar a gravação. O formulário é da responsabilidade do parceiro e implementado pelo mesmo. É apresentado ao utilizador na aplicação como ilustrado na figura 44, através de um componente *web browser* de forma semelhante ao que acontece a quando da autenticação.

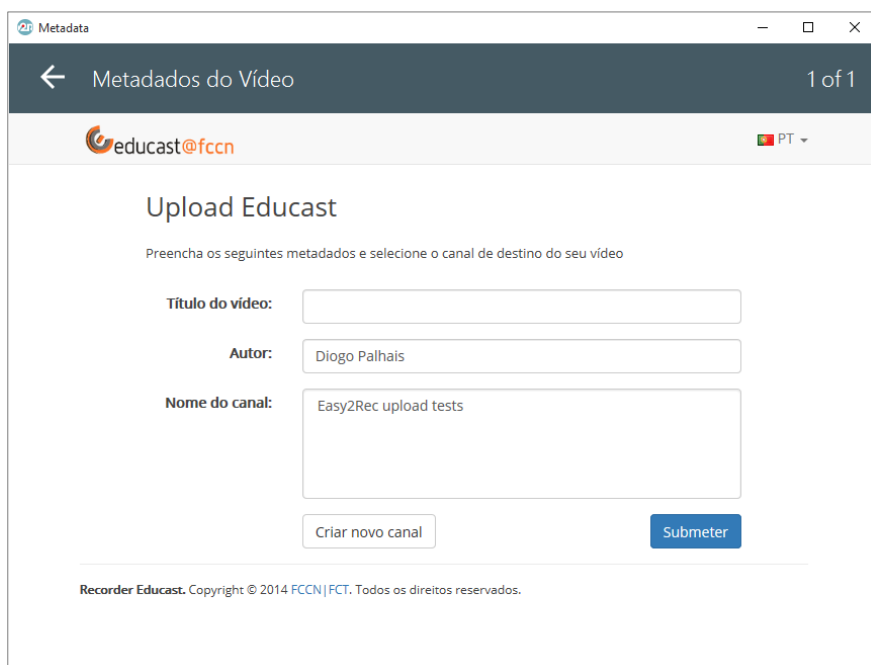


Figura 44 – Janela de metadados de submissão de vídeo para o parceiro

Quando todos os campos estiverem preenchidos, o utilizador selecciona o botão “*Submeter*” para iniciar o processo de submissão enviando o formulário para o parceiro. Neste momento é gerado um identificador para a gravação, que será o ponto de ligação entre os metadados enviados pelo formulário e os ficheiros que serão enviados posteriormente.

Ao iniciar efetivamente o processo de submissão os ficheiros de vídeo e áudio a enviar são processados e transcodificados de forma a garantir que a plataforma do parceiro é capaz de compreender e reproduzir o seu conteúdo. Durante esse processamento é criado um ficheiro *XML* estruturado que contém a especificação e algumas informações sobre os ficheiros a submeter. Na figura 45 é apresentado um exemplo do ficheiro gerado.

```

<?xml version="1.0" encoding="utf-8"?>
<Videos>
<Video>
  <ID>a16531bf-6858-4f18-987b-6b47bedd4aba</ID>
  <Username>Diogo Palhais</Username>
  <User_email>diogopalhais@easy2rec.com</User_email>
  <Type>Presenter And Audio</Type>
  <Filename>Easy2rec_2e9ifp4db0_a16531bf-6858-4f18-987b-6b47bedd4aba_Presenter.mp4</Filename>
  <Channel_id>2e9ifp4db0</Channel_id>
  <Date_trascoded>19/8/2016 19:39:48</Date_trascoded>
  <Date_uploaded>19/8/2016 19:39:48</Date_uploaded>
  <Date_recorded>19/8/2016 19:39</Date_recorded>
  <Duration>00:012:08</Duration>
</Video>
<Video>
  <ID>a16531bf-6858-4f18-987b-6b47bedd4aba</ID>
  <Username>Diogo Palhais</Username>
  <User_email>diogopalhais@easy2rec.com</User_email>
  <Type>Slides</Type>
  <Filename>Easy2rec_2e9ifp4db0_a16531bf-6858-4f18-987b-6b47bedd4aba_Slides.mp4</Filename>
  <Channel_id>2e9ifp4db0</Channel_id>
  <Date_trascoded>19/8/2016 19:39:48</Date_trascoded>
  <Date_uploaded>19/8/2016 19:39:48</Date_uploaded>
  <Date_recorded>19/8/2016 19:39</Date_recorded>
  <Duration>00:012:08</Duration>
</Video>
</Videos>

```

Figura 45 – Exemplo do ficheiro de metadados de uma submissão

O parceiro de forma a receber os ficheiros disponibiliza uma ligação ao seu servidor através do protocolo *FTP*. A aplicação implementa um cliente que se liga ao servidor aplicando as credencias de acesso que identificam a aplicação Easy2rec e que permite o envio dos ficheiros.

Os ficheiros enviados respeitam algumas regras definidas pelo parceiro de modo a serem devidamente processados do seu lado para a consequente publicação dos vídeos sem erros, após a sua submissão com sucesso por parte da aplicação Easy2rec.

As regras são as seguintes: os ficheiros a enviar devem ser compactados num só ficheiro criado no formato *ZIP*. No interior desse ficheiro no formato *ZIP* deve conter todos os ficheiros correspondentes a cada captura de forma independente e o ficheiro estruturado *XML* descrito anteriormente com toda a informação e metadados disponíveis para a submissão.

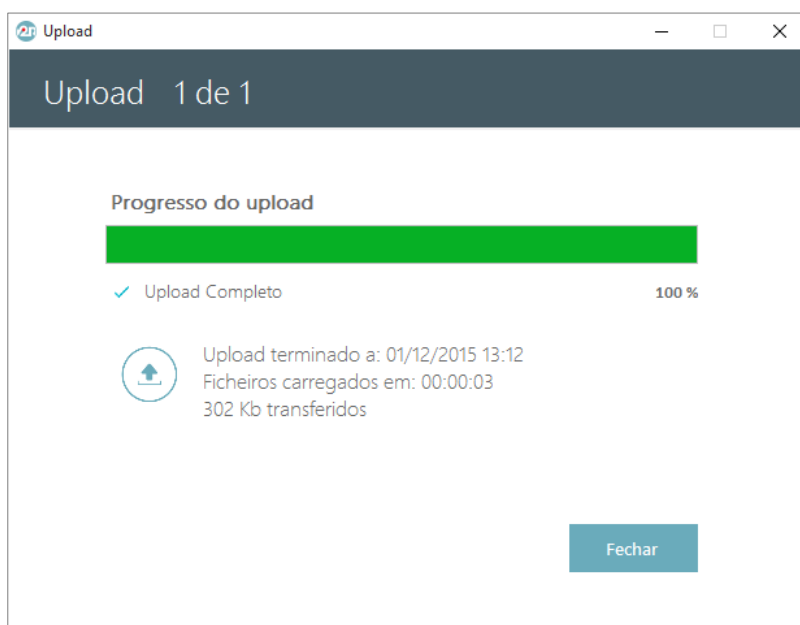


Figura 46 – Janela de submissão de gravação para o parceiro realizada com sucesso

Na figura 46 é demonstrada a informação apresentada ao utilizador no processo de submissão de uma gravação. Nesta janela é apresentado ao utilizador durante todo o processo o estado atual da submissão, indicando qual a percentagem de dados que já foram submetidos e quanto falta para terminar. O utilizador pode realizar a submissão de mais do que uma gravação ao mesmo tempo sem que tenha de aguardar pelo término da primeira. Para isso o utilizador submete os vários formulários de metadados para cada gravação a submeter respetivamente e só após esse passo o processo se iniciará submetendo os vários vídeos sem necessidade de intervenção por parte do utilizador. No final será apresentada a mensagem de sucesso ou insucesso da operação.

Na figura 47 é apresentado o diagrama de sequência que representa o processo de submissão de uma ou mais gravações.

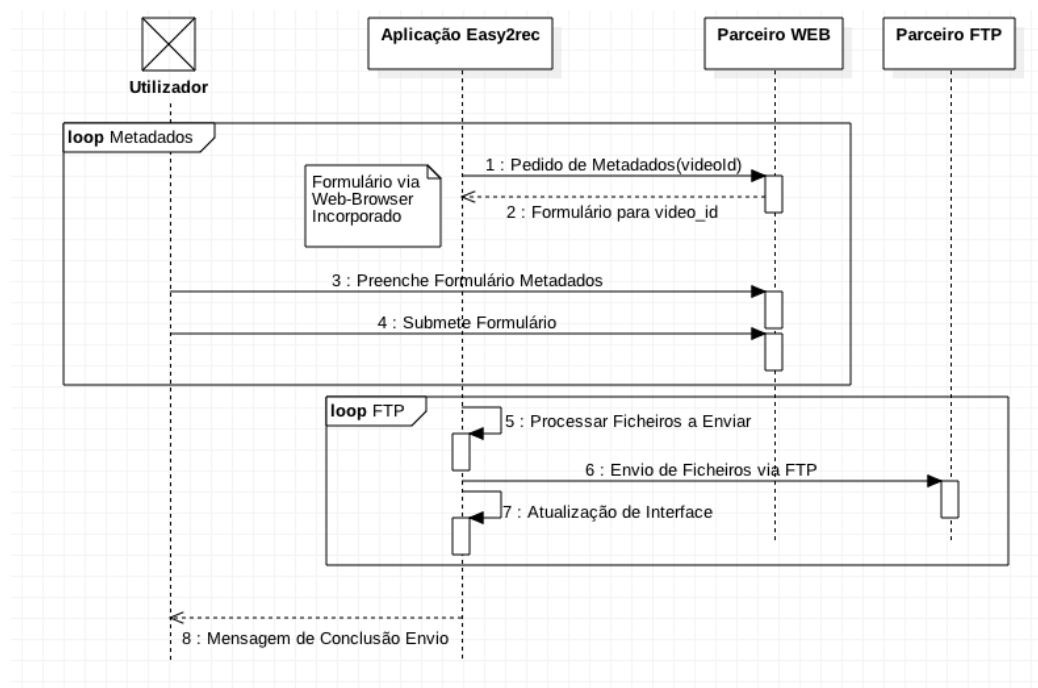


Figura 47 – Diagrama de sequência do processo de submissão via *FTP*

5.3.4 Suporte Multilíngue

A aplicação Easy2rec e a sua interface com o utilizador foi desenvolvida de modo a suportar multilíngue, e neste momento encontram-se implementadas os idiomas Português e Inglês.

Os utilizadores podem seleccionar o idioma desejado na primeira utilização da aplicação como também na secção de configurações da aplicação e a partir desse momento a aplicação é apresentada no idioma seleccionado, sem necessidade de reiniciar a aplicação ou carregar ficheiros de língua adicionais.

5.3.5 Incorporação de *Interface Externa*

Todos os componentes que incluem *interfaces*, informação ou formulários externos da responsabilidade do parceiro e que deverão ser apresentadas ao utilizador como parte integrante do *software* Easy2rec são incorporadas através de um componente *web browser* que fará a ponte entre a aplicação e a chamada à respetiva página a mostrar ao utilizador num determinado momento. As páginas da responsabilidade do parceiro são:

- Autenticação de utilizador via parceiro;
- Pedido de licenciamento de um utilizador ao realizar autenticação via parceiro;
- Formulário de metadados na submissão de cada gravação.

5.3.6 A Notificação à Aplicação Easy2rec

No momento em que um utilizador completa o processo de autenticação junto do parceiro e lhe é atribuída uma licença esta necessita de ser comunicada pelo parceiro à aplicação para que possa concluir o processo do lado da aplicação Easy2rec.

Deste modo foi necessário implementar uma forma de comunicação entre a aplicação Easy2rec e a página do parceiro carregada no *web browser*. De seguida será apresentada a implementação que permitiu essa notificação. Através da linguagem de programação *Javascript* que é executada do lado do cliente, permite notificar a aplicação quando esta é gerada com sucesso do lado do parceiro e se encontra pronta a descarregar e utilizar na aplicação.

Na figura 48 são apresentados os métodos *Javascript* que permitem a leitura e escrita de informação entre a aplicação e a incorporação do parceiro. São eles o método *receivejs* e o *sendjs*. O primeiro é responsável por receber a informação e o segundo por enviar a informação.

Na figura é possível verificar um exemplo de como seria enviada a notificação à aplicação através da criação de um objeto *JSON* que contem toda a informação necessária para completar a autenticação.

```
<script type="text/javascript"/>

function receivejs(message) {
    alert(message); //process the data received
}

function sendjs() {
    var obj = {
        id_hash: "xxxxxxxxxxxxxxxx", // license_hash recebido
        partner: "PARTNER",
        user_email: "user@domain.com",
        user_name: "username",
        date_request: "date",
        validate: partner + user_email + user_name + date_request + "chaveprivada"
    };
    var string = JSON.stringify(obj);
    return window.external.sendcsharp(string);
}

</script>
```

Figura 48 – Exemplo código *Javascript* para comunicação entre parceiro e aplicação

5.4 Dados de Utilizador

Na aplicação Easy2rec é pretendido que seja utilizada por vários utilizadores no mesmo dispositivo, desta forma é necessário a separação da informação referente a cada um. Para cada utilizador é iniciada uma pasta que irá ser o local de armazenamento de todos os ficheiros referentes as gravações. Para além das gravações é iniciado um ficheiro estruturado com a listagem de todas as gravações efetuadas como também das suas submissões para o parceiro.

5.4.1 A Estrutura de Ficheiros

Sendo o objetivo da aplicação a gravação de vídeo aulas e a sua conseqüente submissão para o servidor parceiro é necessário que a aplicação disponha da informação necessária para saber quais os vídeos que já foram submetidos, os que ainda não foram, a sua localização e toda a informação adicional relevante para o processo. Desta forma definiu-se uma estrutura de ficheiros que irá suportar a aplicação e que deve ser respeitada para o correto funcionamento da aplicação. Na figura 49 é apresentada a estrutura de ficheiros criada para cada utilizador da aplicação, sendo que num mesmo dispositivo pode haver mais que um utilizador, dessa forma a pasta raiz é identificada pelo email único de registo.



Figura 49 – Estrutura de ficheiros da aplicação Easy2rec

A estrutura é inicializada no momento em que o utilizador realiza a autenticação pela primeira vez num determinado dispositivo. É criada a pasta raiz e contendo duas pastas denominadas por *temp* e *videos*.

Na pasta *temp* são guardados todos os ficheiros criados na realização de uma gravação, sendo eles os ficheiros de vídeo, áudio como também o *thumbnail*. É atribuído o nome de cada ficheiro através da criação um identificador único pela função *Guid.NewGuid()* disponibilizado na plataforma *.NET*.

A pasta *videos* irá conter todos os ficheiros de cada gravação que são processados no processo da submissão. É criada uma subpasta para cada gravação submetida atribuindo um *video_id* gerado aquando da preparação da submissão e do envio do formulário de metadados pelo utilizador. Após os ficheiros serem processados com sucesso e o ficheiro de *metadata.xml* criado é comprimido todo o conteúdo da pasta para um ficheiro em formato *ZIP* pronto para iniciar a submissão.

5.5 Testes de Integração

Os testes à aplicação e em específico a cada módulo de forma independente ocorreu sempre paralelamente ao processo de desenvolvimento dos mesmos.

Para cada funcionalidade implementada era imediatamente testada a sua funcionalidade e a sua integração na aplicação já existente. No que diz respeito ao serviço de licenciamento foi também testado através da realização de casos de teste positivos e negativos, de forma a verificar as respostas obtidas caso os dados fornecidos se encontrassem válidos ou inválidos. Este módulo acusa uma maior sensibilidade na fase de testes pois é um módulo muito importante para o conceito de negócio da solução. Caso este se encontrasse mal desenvolvido ou com falhas iria representar uma grande perda para a empresa Edugraal.

O processo de comunicação e experimentação com a equipa do parceiro desenrolou-se sempre de uma forma iterativa que permitiu ao longo de todo o desenvolvimento e à medida que novas funcionalidades eram concluídas desenrolar-se em simultâneo o processo de teste das mesmas tanto de um lado como do outro. Desta forma o processo foi agilizado encontrando sempre alternativas para todos os contratempos encontrados ou mesmo algumas incompatibilidades.

A equipa técnica do parceiro/cliente mostrou-se sempre muito aberta e disponível para ajudar o que claramente contribuiu de forma muito importante para o sucesso do presente trabalho.

6 Avaliação da Solução

Para iniciar o capítulo de avaliação da solução citamos (Saxena e Singh, 2014) que definem o processo de teste de *software* como:

O teste de software é um processo de verificação e validação de que um aplicativo ou programa de software atende o negócio e requisitos técnicos que orientaram a sua concepção e desenvolvimento, e funciona como esperado. O teste de software também identifica defeitos importantes, falhas ou erros no código do aplicativo que devem ser corrigidos. (tradução nossa).

Existem diversas metodologias de teste possíveis de aplicar das quais vamos destacar a *Black Box*, *White Box* e *Gray Box*. A metodologia *Black Box* consiste no teste de um sistema sem o conhecimento dos componentes internos. A equipa de testes não tem acesso ao código fonte e se abstrai da arquitetura do sistema. Normalmente o indivíduo que testa interage com um sistema apenas através da sua interface com o utilizador, fornecendo os parâmetros de entrada e examinando as suas saídas sem saber onde e como os parâmetros de entrada foram processados. Por outro lado, na metodologia *White Box* a equipa de testes têm acesso ao código fonte do sistema e conhecimento de todos os componentes que o integram. Esta metodologia de teste está mais vocacionada para análise de código testando a lógica da implementação e aumentar a tolerância ao erro. Por último temos a metodologia *Gray Box* que não é nada menos que uma junção das duas anteriores. A equipa de testes nesta metodologia tem acesso limitado aos componentes internos do sistema, não tendo acesso ao código fonte.

Neste capítulo é definido todo o processo de teste da solução implementada, o grupo de pessoas que o realizou, como também apresentados os resultados obtidos e a sua análise. O processo desenrolou-se de uma forma iterativa, sendo repartido por várias fases. Finalmente

são apresentadas as conclusões avaliando se a solução implementada responde efetivamente ao problema definido inicialmente bem como a todos os requisitos identificados.

6.1 As Grandezas de Teste

Para o estudo a realizar são tidas em conta grandezas como a facilidade de aprendizagem dos utilizadores a manusear o *software* e a satisfação do utilizador. A avaliação será desenvolvida no sentido de responder as seguintes questões (hipóteses):

- O grupo de teste identificou erros críticos que impossibilitem a conclusão do processo?
- O grupo de teste identificou erros graves que não invalidem a conclusão do processo, mas sejam prioritárias na sua resolução?

No entanto outras grandezas que não serão introduzidas no estudo a realizar serão também testadas. Mais especificamente no componente de licenciamento como será para a utilização unicamente das entidades parceiras devidamente credenciadas, é necessário testar grandezas como a segurança e performance.

6.2 As Hipóteses a Testar

Com a definição das hipóteses a testar pretendemos aferir se o produto desenvolvido atende todos requisitos do cliente e se após a execução do experimento se encontra em condições de avançar para a fase de aceitação por parte do cliente. Para que isso seja possível o produto terá de verificar as hipóteses definidas de seguida:

6.2.1 Hipótese 1

Hipótese Nula (H_0): Não foram encontrados erros ou falhas consideradas catastróficas que impossibilitam a conclusão do processo e invalidam o correto funcionamento do produto.

H_0 : Número de erros/falhas detetadas de nível 4 = 0

Hipótese Alternativa (H_1): Foram encontrados erros ou falhas consideradas catastróficas que impossibilitam a conclusão do processo e invalidam o correto funcionamento do produto.

H_1 : Número de erros/falhas detetadas de nível 4 > 0

6.2.2 Hipótese 2

Hipótese Nula (H_0): Não foram encontrados erros ou falhas consideradas prioritárias mas que não impossibilitam a conclusão do processo e não invalidam o correto funcionamento do produto.

H_0 : Número de erros/falhas detetadas de nível 3 = 0

Hipótese Alternativa (H_1): Foram encontrados erros ou falhas consideradas prioritárias mas que não que impossibilitam a conclusão do processo e não invalidam o correto funcionamento do produto.

H_1 : Número de erros/falhas detetadas de nível 3 > 0

6.3 A Metodologia de Avaliação

Para realizar a avaliação da solução aplicámos o método de análise heurística baseada no trabalho de Nielsen (1993), embora seja um trabalho que remonta há já alguns anos atrás é uma técnica muito usada ainda nos dias de hoje e que nos permite de uma forma simples aferir várias conclusões de forma a perceber a atual situação da nossa solução e em que medida esta responde com sucesso as necessidades dos utilizadores e cliente. Este método consiste em submeter a interface e funcionalidades de um determinado *software* a avaliação de um grupo de pessoas, com base num conjunto de heurísticas previamente definidas a que esse grupo terá de responder. As heurísticas mais comuns e que iremos ter como base são (Nielsen, 1995):

- H1 - Visibilidade do estado do sistema;
- H2 - Correspondência entre o sistema e o mundo real;
- H3 - Controlo e liberdade do utilizador;
- H4 - Consistência e padronização;
- H5 - Prevenção de erros;
- H6 - Reconhecimento em vez de lembrança;
- H7 - Flexibilidade e eficiência de uso;
- H8 - Estética e *design* minimalista;
- H9 - Reconhecimento, diagnóstico e recuperação de erros;
- H10 - Ajuda e documentação.

Uma parte importante deste método é a definição da escala de erros que nos permite diferenciar erros ou falhas muito graves que impedem de continuar o processo com sucesso de outras falhas que não são tão importantes, mas que o utilizador considera que devem ser tratadas. A escala a ter em conta a quando do preenchimento do relatório é a seguinte:

- Nível 0 - Não é encarado necessariamente como um problema de usabilidade;
- Nível 1 - Problema estético que não tem necessidade de ser corrigido, a menos que haja tempo e recursos disponíveis;
- Nível 2 - Pequeno problema com baixa prioridade na correção;

- Nível 3 - Problema com alta prioridade de correção;
- Nível 4 - Catástrofe, ou seja, o produto só será aceite se a correção for implementada.

Aplicando os conceitos apresentados e descritos anteriormente o formulário a ser preenchido pelos inqueridos será uma tabela com a estrutura bem definida como podemos verificar na tabela 9 apresentada de seguida.

Tabela 10 - Exemplo de estrutura do relatório de avaliação e preenchimento

Heurística Violada	Erro	Local	Gravidade	Solução Sugerida
H8	Menu de navegação nos PPT é muito transparente o que dificulta a visualização	Aplicação PowerPoint	3	Criar uma barra de navegação ou aumentar o contraste dos botões (setas para avançar e retroceder)

O processo de avaliação foi levado a cabo por um grupo de inqueridos constituído por elementos da equipa técnica do cliente/parceiro FCCN juntamente com um pequeno grupo de técnicos de informática seleccionados pelo primeiro grupo pertencentes as instituições de ensino superior membros da rede RCTS e que serão responsáveis pela formação dos futuros utilizadores da solução numa fase posterior.

Todos os elementos inqueridos tiveram conhecimento prévio da solução e qual o seu objetivo. Possuem um nível de conhecimentos em informática bastante avançados como também conhecimento do processo de criação de conteúdos multimédia em específico vídeo e áudio.

O método de teste usado foi o *Black Box*, pois os inqueridos não possuem acesso ao código fonte da solução. Estes testaram através do uso da aplicação unicamente pela sua interface gráfica avaliando e reportando os resultados esperados com os efetivamente obtidos.

6.4 Os Resultados Obtidos

Os dados apresentados de seguida foram obtidos através da ferramenta de colaboração online *Google Sheets*, onde foi criado e disponibilizado um documento ao grupo de teste contendo a estrutura definida e apresentada anteriormente na tabela 10.

A avaliação realizou-se em dois momentos distintos. Num primeiro momento lançou-se a versão *alfa* da solução para a realização de testes unicamente pela equipa técnica do parceiro FCCN. Pretendeu-se testar a aplicação apenas internamente nesta fase, de modo a detetar a possibilidade de ocorrer possíveis erros de um nível de gravidade maior. No segundo momento foi lançada a versão *beta* para testes pelos dois grupos. Só após a aceitação das alterações e correções realizadas à versão alfa é que o segundo grupo teve acesso à solução. Desta forma foi obtido o feedback necessário de uma forma iterativa que possibilitou corrigir rapidamente a grande maioria dos erros detetados.

A tabela 11 apresentada de seguida contém a informação obtida através do preenchimento do documento pelo grupo de teste do cliente, referente à fase *alfa*.

Tabela 11 – Resultados obtidos versão *alfa*

Id	Heurística Violada	Erro	Local	Gravidade	Solução Sugerida
1	H1	Quando não é selecionado nenhum modo de captura de conteúdos, não há qualquer alteração na interface de acordo com a ação realizada.	Ambiente de Gravação	2	Assim como quando se seleciona qualquer outra opção deveria de surgir na zona das <i>apps</i> informação clara de que essa parte não vai ser gravada.
2	H5	Modo de redimensionamento livre da área de gravação não é consistente e pode levar a resultados não esperados.	Ambiente de gravação	3	Adicionar <i>presets</i> de geometria para a captura de conteúdos (640x480, 800x600, 1024x768)
3	H5	O ficheiro de informação referente ao utilizador quando fica corrompido ou mal estruturado é impossível usar a aplicação	Sistema de informação	4	Tentar eliminar o erro ocorrido na leitura do ficheiro e recuperar o conteúdo ou realizar um <i>reset</i> ao ficheiro
4	H8	A informação da licença não necessita de indicar as horas minutos e segundos de expiração da licença recebida	Dispensa de chaves de licença	1	Mostrar apenas a data de validade da licença recebida
5	H5	Seleção de resoluções não compatíveis com o sistema não deveriam ser disponibilizadas	Ambiente de Gravação	3	Mostrar a informação ao utilizador a indicar que a resolução foi impossível de aplicar.
6	H1	Mostrar tempo disponível em disco durante a gravação ou tempo médio gasto por hora	Ambiente de Gravação	4	Mostrar ao utilizador a informação necessária para evitar que este fique sem espaço no disco durante a gravação
7	H2	Texto do botão para login deverá passar de "Login as <i>Partner</i> " para " <i>Partner</i> Login"	Início	0	Texto do botão para login deverá passar de "Login as <i>Partner</i> " para " <i>Partner</i> Login"
8	H8	Deve ser solicitada ao utilizador a seleção do idioma na primeira execução	Início	3	Mostrar ao utilizador os idiomas disponíveis no primeiro ecrã da aplicação

9	H5	Depois de terminada a gravação a uma janela de uma aplicação, quando voltamos ao gravador é esse o modo que está pré-selecionado, mas não existe indicação de qual a janela que está a ser gravada.	Ambiente de Gravação	3	No início de cada gravação deve ser feito o "reset" ao modo selecionado de gravação.
10	H4	A caixa de confirmação de "As configurações foram gravadas com sucesso" é desnecessária	Configurações	2	Remover caixas de confirmação de ações simples
11	H3	Quando terminada a gravação a janela fica bloqueada em termos de geometria. Antes da gravação dá para redimensionar, durante a gravação está fixa depois de terminada continua fixa	Ambiente de Gravação	3	Desbloquear a geometria da janela depois de terminar a gravação
12	H10	O manual de utilização deverá conter imagens demonstrativas dos passos que o utilizador deverá realizar	Documentação	2	Inserir imagens no manual juntamente com as descrição dos passos a realizar

Após ter sido finalizado o momento de testes referente à fase *alfa* analisamos todos os problemas reportados partindo para a sua resolução. Após a resolução de todos os erros detetados e reportados pela equipa técnica do cliente foi compilada uma nova versão da solução com as alterações implementadas. Publicada a versão *beta* iniciou-se uma nova fase de testes. A tabela 12 apresentada de seguida contém a informação recebida através do preenchimento do documento pelo grupo completo de testes durante a fase *beta*.

Tabela 12 – Resultados obtidos versão *beta*

Id	Heurística Violada	Erro	Local	Gravidade	Solução Sugerida
1	H4	Botão de "back" deve voltar para a "lista de recordings" e não para o menu principal	Metadados	2	Alterar o destino do link da seta "para trás"
2	H4	Thumbnails não respeitam a geometria original. A listagem estende as imagens para formato 4:3	Lista de Vídeos	1	Apresentar os thumbnails de acordo com a geometria original das imagens
3	H2	O browser não	Aplicação	2	Adicionar a

		permite pesquisa diretamente da barra de endereços	Browser		funcionalidade de pesquisa quando utilizador coloca um <i>url</i> não válido na barra de endereços
4	H8	A data de gravação não necessita de ter o segundo em que foi realizada	Lista de Gravações	1	Retirar os segundos da data ganhando mais espaço na tabela para outros atributos

6.5 Análise de Resultados Obtidos

Começamos por apresentar a análise aos resultados obtidos referentes à fase *alfa* devido ao seu maior número de erros reportados e por ser a primeira versão da solução disponibilizada para teste. Posteriormente analisa-se as diferenças dos resultados obtidos na fase *alfa* com os obtidos na fase *beta*. Por fim são verificadas as hipóteses definidas e apresentadas na secção 6.2, verificando se a solução responde as necessidades e requisitos do cliente e se pode e deve ser aceite pelo mesmo.

Para a análise estatística dos resultados obtidos, estes foram ordenados por grau de gravidade de modo a identificar quais os principais problemas a serem corrigidos em primeiro lugar. Aplicou-se uma distribuição de frequências agrupando os dados por classes de ocorrências, resumindo a análise do conjunto de relatórios dos intervenientes da experiencia. Foram processados os relatórios de teste preenchidos pelos intervenientes do teste de modo a analisar e aferir quais os erros que ocorrem com mais frequência tendo em conta os níveis mais altos em primeiro lugar.

Foi realizada depois da análise de prioridades a verificação da validade dos erros e falhas identificadas implementando a sua correção. A tabela 13 apresenta de seguida os resultados obtidos durante a fase *alfa* ordenados por ordem decrescente de gravidade e é descrita também a solução implementada para cada uma delas. Todos os erros reportados foram devidamente corrigidos.

Tabela 13 – Resultados obtidos ordenados por gravidade na fase *alfa*

Id	Solução Implementada	Gravidade
5	Adicionada opção nas configurações de <i>reset</i> do ficheiro de lista de gravações. Quando ocorre algum erro na leitura do ficheiro é realizado um <i>debug</i> para solucionar o problema sem ser necessário o <i>reset</i>	4
9	Com base nas configurações selecionadas no momento do inicio da gravação é calculado e mostrado ao utilizador o tempo restante que é possível guardar no disco	4
2	Adicionada <i>combobox</i> na <i>statusbar</i> que permite	3

	escolher <i>presets</i> de geometria a ser gravada para a área de conteúdos.	
7	Apenas as resoluções aplicáveis ao sistema em questão são apresentadas na interface	3
11	Inserida uma <i>selectionbox</i> na introdução à aplicação que é mostrada na primeira execução	3
12	No modo de gravação de janela de aplicação quando volta à janela de gravação o modo selecionado é o de gravação de <i>apps</i>	3
14	Ao abrir o gravador a geometria pode ser alterada. Ao iniciar gravação a geometria é bloqueada. Ao parar de gravar surge a mensagem “repetir gravação”, se responder sim, volta ao gravador com a geometria desbloqueada	3
1	Informação na área de conteúdos do que está a ser gravado quando não existe <i>preview</i> .	2
13	Removida a caixa de confirmação identificada	2
16	Manual de utilização atualizado com imagens e lista de passos a realizar para o bom uso da aplicação e as suas funcionalidades	2
6	A informação disponibilizada no momento do registo da licença recebida foi atualizada	1
10	Texto apresentado no referido botão foi atualizado para o sugerido	0

A tabela 14 apresenta de seguida os resultados obtidos durante a fase *alfa* por ordem decrescente de gravidade lado a lado com frequência acumulada. Esta tabela ajuda-nos a compreender um pouco melhor o ponto de situação da aplicação no final desta importante fase. Analisando os dados presentes na tabela 14 é possível verificar que foram identificados dois erros ou falhas de nível 4 que são considerados catastróficos e que impossibilitam ou invalidam o correto funcionamento do produto. Desta forma foi prioritária a resolução dos mesmos antes de avançar para outros erros detetados com um grau de severidade menor.

Tabela 14 – Análise dos resultados obtidos por frequência de gravidade

Gravidade	Frequência por gravidade
4	2
3	5
2	3
1	1
0	1

Na segunda fase de testes denominada por *beta* foram registadas apenas quatro falhas/erros de níveis de gravidade reduzidos, o que representam apenas pequenos problemas com baixa prioridade na correção. No entanto todos os erros e falhas reportados foram devidamente tratados não tendo restado nenhum erro identificado e reportando por resolver na versão final.

A melhoria de uma fase para a outra é bastante notória e positiva o que nos deixa bastante satisfeitos com o trabalho desenvolvido, pois verificamos que a equipa de testes não reportou nenhuma falha grave no final da fase *beta*. De seguida são apresentados os resultados obtidos após a análise das hipóteses a testar para ambas as fases de teste.

6.5.1 A Versão Alfa

Perante os dados apresentados podemos retirar a conclusão de que para a hipótese 1 a testar o número de falhas ou erros reportados do nível 4 foram dois. Logo como temos $x = 2$ desta forma terá de ser rejeitado H_0 e consequentemente aceite H_1 devido ao número de erros/falhas detetadas de nível 4 ser maior que 0. No que diz respeito à hipótese 2 a testar temos $x = 5$ devido ao número de falhas ou erros reportados de nível 3. Desta forma o resultado é similar ao obtido no teste da hipótese 1 rejeitando H_0 e aceitando H_1 .

Perante a análise aos resultados e respondendo às questões definidas no início deste capítulo, a solução nesta fase não poderia ser aceite pelo cliente, uma vez que contém erros ou falhas consideradas catastróficas e que impossibilitam o correto uso da solução pelos utilizadores e não resolvem o problema definido.

No entanto esta não é a decisão final tendo ainda a possibilidade de correção dos erros reportados nas fases seguintes, tentando tomar o melhor caminho possível a percorrer até alcançar à versão final.

6.5.2 A Versão Beta

Após realizada toda a análise e correção dos erros e falhas reportadas na fase anterior, foi facultada uma nova versão da solução. Esta nova versão foi testada pelos dois grupos identificados anteriormente. Esta forma iterativa de teste à solução permitiu resolver as falhas mais graves internamente numa primeira fase e agora nesta segunda analisar internamente como também através de um conjunto selecionado de inquiridos externos. Desta forma testamos uma última vez antes de realizar a avaliação da solução final.

Perante a análise dos resultados obtidos nesta fase apresentados na tabela 12 é possível concluir que foram reportados zero erros de nível 3 ou 4, desta forma temos $x = 0$ seja na hipótese 1 ou na hipótese 2. Tendo em conta estes resultados obtidos estamos em condições de poder aceitar em ambas as hipóteses a hipótese nula H_0 o que nos permite olhar para a solução e dizer que construímos a solução corretamente.

6.6 Conclusão da Avaliação

Após a análise aos resultados obtidos nos dois momentos distintos podemos verificar que houve uma importante alteração na decisão.

Na primeira fase foram detetados vários erros ou falhas graves tanto de nível 3 como de nível 4. Erros esses que impossibilitavam o correto funcionamento da aplicação. Sem a sua devida correção não seria possível avançar para a aceitação da solução.

Nesse momento após a análise dos problemas reportados houve um grande avanço no desenvolvimento com o objetivo de finalizar a aplicação sem qualquer problema identificado de forma a que o cliente seja inteiramente satisfeito com a solução obtida. Todos os problemas identificados nesta fase foram devidamente resolvidos.

Na segunda fase foram detetados e reportados quatro erros, no entanto já não foram detetados nenhuns erros de nível 3 ou 4. Unicamente foram detetados erros menores que não influenciam no processo. Desta forma o caminho para a aceitação da solução ficou nesta fase bastante mais próximo. Todos os erros detetados também nesta fase mesmo que menores foram corrigidos.

A aceitação da solução foi celebrada entre as partes envolvidas declarando que o produto satisfaz todos os requisitos definidos no ponto inicial do projeto e cumpre todas as funcionalidades descritas e acordadas.

7 Conclusão

Neste capítulo são apresentadas as conclusões relativas a todo o desenvolvimento levado a cabo na presente dissertação, abordando resumidamente o contexto, o trabalho desenvolvido e os resultados obtidos. São apresentadas também algumas das possíveis melhorias à solução implementada a trabalhar no futuro. Finalmente apresentam-se as principais limitações encontradas no decorrer da realização deste trabalho.

7.1 Considerações gerais

A presente dissertação centrou-se essencialmente na integração da aplicação Easy2rec que possibilita a gravação de vídeo aulas na plataforma Windows com o serviço *online* de distribuição de vídeo aulas Educast. Este processo envolveu implementar bastantes alterações ao software Easy2rec como também o desenvolvimento de um serviço *online* de raiz para o licenciamento de utilizadores externos provenientes dos parceiros/clientes.

Iniciou-se pela contextualização do problema e pela constatação que as TIC são cada vez mais utilizadas em ambiente educativo por alunos, professores e equipas técnicas, como ferramentas mediadoras de suporte à produção, distribuição e consumo de conteúdos audiovisuais no apoio ao processo de ensino e aprendizagem.

O *workflow* da aplicação Easy2rec por ser bastante simplificado numa tarefa por vezes complexa para a grande maioria das pessoas, aliado às suas possibilidades de integração com outras plataformas e adaptação às atuais metodologias de ensino das instituições aderentes, tem apresentado um grau de acolhimento bastante positivo por parte da comunidade que têm contacto com a solução mesmo antes da sua integração com o Educast.

O problema foi analisado cuidadosamente juntamente com o contexto que o engloba, tendo em contas os dois projetos a integrar e as suas especificidades. A análise de valor foi também um tema abordado de forma a compreendermos melhor as potencialidades da solução a implementar e optar pelo melhor caminho possível.

Analisaram-se um conjunto de soluções e diferentes abordagens desenvolvidas por várias NREN's europeias para o mesmo problema, no entanto após um estudo de todas elas apresentado no capítulo 3 concluímos que nenhuma das soluções apresentadas satisfazia as necessidades e requisitos do projeto.

Do ponto de vista da Edugraal, empresa que explora comercialmente a solução Easy2rec, é uma enorme mais valia associar-se a entidades como a FCCN que permite acrescentar valor, validar os seus produtos e crescer como empresa. Do lado da FCCN, permitiu colmatar uma das suas necessidades mais evidentes nesta fase permitindo que o serviço Educast possa escalar alargando o suporte a outras plataformas. Foi uma parceria sem dúvida muito

vantajosa para ambas as partes envolvidas no projeto e que com certeza não terminará por aqui.

A integração e o desenvolvimento do *software* foram desenvolvidos no período de tempo espetável e esperado sem qualquer tipo de atrasos significativos ou contratempos dignos de assinalar. A solução foi avaliada por uma equipa constituída para o efeito em várias fases do seu desenvolvimento. Os resultados obtidos foram muito positivos, culminando na aceitação da solução pelo parceiro/cliente. Encontra-se neste momento em ambiente de produção e está disponível para toda a comunidade.

As principais dificuldades sentidas no projeto foram principalmente, numa fase inicial, a definição dos métodos e formas de comunicação entre os projetos nos vários processos. O processo de autenticação é um excelente exemplo em como foi necessária bastante flexibilidade no diálogo para permitir a adaptação dos dois projetos com sucesso.

7.2 Trabalhos Futuros

Os principais objetivos definidos para o projeto foram cumpridos, no entanto existe um longo caminho a percorrer para melhorar a solução. Neste momento o principal trabalho a desenvolver é promover junto de todos os membros da RCTS o uso da solução Easy2rec que agora se encontra devidamente integrada com a plataforma Educast. Esta tarefa será o principal foco após a publicação da aplicação e permitirá também criar uma relação de proximidade com o cliente de forma a identificar possíveis dificuldades que possam surgir com o uso frequente por parte dos utilizadores e desta forma melhorar continuamente o produto.

A publicação de escrita científica é também um dos objetivos do autor após o termino da presente dissertação. Foram já escritas e aceites duas publicações relacionadas o presente estudo pela conferencia ICERI 2016 que serão apresentadas em novembro deste ano em Sevilha, Espanha.

A replicação do processo de integração com outros parceiros da mesma área é também um trabalho muito importante a desenvolver no futuro, de modo a alargar a rede de parceiros e o número de utilizadores a utilizar a solução. Do ponto de vista da empresa é necessário trabalhar sobre este objetivo para garantir a sustentabilidade do projeto e garantir novas formas de retorno.

Por último, mas não menos importante gostaríamos de contribuir para o melhoramento do processo de ensino e aprendizagem tirando partido das novas tecnologias, facilitando o processo de criação de vídeo aulas em formato digital de maneira fácil, rápida e intuitiva.

Referências

- (Bell et al. 2001) Bell, J, Clark, M, Elmhorst, A, Gibbs, M, Homer, A, Lee, B, Milner, M, Narkiewicz, J, Rehan, A, Schenken, J, ASP:NET Programmer's Reference, 2001, Wrox Press, Birmingham, UK
- (Castels, 2007) Castells, M ,2007, A Galáxia Internet – Reflexões sobre a Internet, Negócios e Sociedade. Fundação Calouste Gulbenkian.
- (Eades, 2014) Eades, John, 2014 , Why Microlearning is HUGE and how to be a part of it, <https://elearningindustry.com/why-microlearning-is-huge>
- (EDUCAST@FCCN 2016) Educast@fccn. Visualizado a 12 de dezembro de 2015, <http://www.educast.fccn.pt>
- (FFMPEG, 2015) FFmpeg: A complete, cross-platform solution to record, convert and stream audio and vídeo, visualizado em 17 de dezembro de 2015, <https://www.ffmpeg.org/>
- (Fielding, 2000) Fielding, R. Architectural Styles and the Design of Network-based Software Architectures. UNIVERSITY OF CALIFORNIA, IRVINE, 2000.
- (Gibson e Ward, 2000) Gibson, R., Ward, S. (2000). Reinvigorating Democracy? British Politic and Internet, Ashgate, Aldershot.
- (Gomes, 2009) Gomes, M. J. (2009). Problemáticas da avaliação em educação online. In Paulo Dias & António Osório (orgs.), Actas da VI Conferência Internacional de TIC na Educação – Challenges 2009; Braga: Centro de Competência da Universidade do Minho;
- (Herrala et al, 2011) Herrala, M, Pakkala, P, Haapasalo, H, Value-creating networks – A conceptual model and analysis, 2011, University of Oulo
<http://herkules oulu.fi/isbn9789514293863/isbn9789514293863.pdf>
- (ITHE, n.d.) International Trends in Higher Education 2015, University of Oxford, visualizado em 7 de setembro de 2016, <https://www.ox.ac.uk/sites/files/oxford/International%20Trends%20in%20Higher%20Education%202015.pdf>
- (JSON, 2016) Introducing JSON, visualizado em 11 de Janeiro de 2016, <http://www.json.org/>
- (Marques e Pedroso, 2005) Marques, 2005, C# 2.0 , FCA Editora de Informática, Lisboa
- (Microsoft, 2016) Why Microsoft for your operational database management system?, visualizado em 11 de janeiro de 2016, <https://www.microsoft.com/en-us/server-cloud/products/sql-server/Features.aspx>
- (MSDN 2016) Overview of the .NET Framework, MSDN, Microsoft, visualizado em 14 janeiro 2016, <https://msdn.microsoft.com/en-us/library/zw4w595w.aspx>
- (Pappas, 2013) Pappas, Chistopher, 2013, Future eLearning Trends and Technologies in the Global eLearning Industry.
<https://elearningindustry.com/future-elearning-trends-and->

- technologies-in-the-global-elearning-industry
- (Peters, 2004) PETERS, Otto. A educação a distância em transição: tendências e desafios. Trad. Leila Ferreira de Souza Mendes. São Leopoldo, RS: Ed. Unisinos, 2004. Resenha de João Mattar.
- (Pitts ,1998) Natanya Pitts, 1998, XML In Record Time: Teach Yourself the Essential XML Skills, Sybex
- (Saxena e Singh, 2014) Saxena e Singh, 2014, Gray Box Testing: Proactive Methodology for the Future Design of Test Cases to Reduce Overall System Cost
http://www.krishisanskriti.org/vol_image/03Jul201510071017.pdf
- (Siony da Silva, 2010) REFLEXÕES SOBRE WEB 1.0, WEB 2.0 E WEB SEMÂNTICA, 2010, Singergia, São Paulo,
http://www.cefetsp.br/edu/prp/sinergia/complemento/sinergia_2010_n2/pdf_s/segmentos/artigo_01_v11_n2.pdf
- (Sommerville, 2001) Sommerville, 2001, Software Engineering, Sixth Edition, Pearson Education, Harlow ,England
- (TERENA, 2012) TNC Recording Contest 2012, visualizado a 3 de fevereiro de 2016,
<https://www.terena.org/activities/media/rc/rc.html>
- (Youtube Statistics, 2016) Estatísticas, visualizado em 18 de janeiro de 2016,
<https://www.youtube.com/yt/press/pt-PT/statistics.html>