



Distribuição e visualização de recursos educativos interativos

RAFAEL DIOGO MARTINS DE OLIVEIRA

Outubro de 2022

Distribuição e visualização de recursos educativos interativos

Rafael Diogo Martins de Oliveira

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

Orientador: António Vieira de Castro (PhD)

Supervisor: Abel Ferreira

<À minha família e namorada, pelo apoio durante o meu percurso académico e pessoal.>

Resumo

Verifica-se uma crescente procura, utilização e exigência, por parte dos utilizadores, de plataformas digitais de apoio ao estudo. Uma das causas deste aumento deve-se ao crescente acesso a equipamentos digitais, especialmente telemóveis e tablets, por parte das gerações mais novas. Este fenómeno foi drasticamente acelerado pela pandemia Covid-19, com início em 2020, e conseqüentes medidas de confinamento.

Uma das principais funcionalidades das plataformas de *e-Learning* é a distribuição e visualização de conteúdos educativos. Se esta funcionalidade falhar a capacidade de a plataforma cumprir o seu objetivo é dificultada, ou até impossibilitada.

Este documento procura estudar os obstáculos encontrados por estas plataformas, tendo como objetivo encontrar a melhor forma de distribuir e apresentar conteúdos educativos. Para tal, será estudado um caso real, e será projetado e desenvolvido um novo módulo de distribuição de recursos educativos.

Palavras-chave: *e-Learning*, educação, LMS, *media player*, recursos interativos.

Abstract

There is a growing demand for quality digital e-Learning platforms. One of the causes for this growth is the increased access to digital devices, especially mobile phones and tablets, by the younger generations. This phenomenon was drastically accelerated by the Covid-19 pandemic, which began in 2020, and its consequent lockdown measures.

One of the main features of an e-Learning platform is the distribution and visualization of educational content. If this functionality fails, the ability of a platform to fulfill its purpose is severely hampered, or completely impeded.

This document seeks to study the obstacles encountered by these platforms, aiming to find the best way to distribute and present educational content. To this end, a real case will be studied, and a new module for the distribution of educational content will be projected and developed.

Keywords: e-Learning, education, LMS, media player, interactive resources.

Agradecimentos

Não poderia deixar de registar neste documento um agradecimento a todos aqueles que me ajudaram no meu caminho académico e na realização desta dissertação, especialmente ao meu orientador Prof. Doutor António Vieira Castro, pela sua sempre pronta disponibilidade, dedicação e empenho.

Aos meus pais, irmã e à minha namorada que sempre me apoiaram e incentivaram a seguir o meu percurso académico.

Ao grupo de investigação SIIS (Sistemas de Interação e Inovação Social) por me ter associado como colaborador.

Ao ISEP e aos docentes do Departamento de Engenharia Informática (DEI) por todo o conhecimento que me foi transmitido ao longo do meu percurso académico.

À Porto Editora pela oportunidade de ligar o meu caminho profissional e académico, e a toda a equipa de *e-Learning* pelo apoio no desenvolvimento deste projeto, em especial ao Engenheiro Abel Ferreira.

Índice

Resumo	v
Abstract.....	vii
Agradecimentos	ix
Índice.....	xi
Lista de Figuras	xv
Lista de Tabelas	xix
Acrónimos e Símbolos.....	xx
1 Introdução	1
1.1 Contextualização.....	1
1.2 O Problema.....	2
1.3 Objetivos e contributos esperados.....	4
1.4 Metodologia	5
1.5 Motivação	5
1.6 Estrutura do Documento	6
2 Análise de Valor	7
2.1 Modelo de negócio <i>Canvas</i>	7
2.2 Valor para o Cliente e Valor Percebido	9
2.3 Processo de inovação	11
2.3.1 Modelo de Desenvolvimento de Conceito Novo	12
3 Estado da Arte	17
3.1 Conceitos e Tecnologia	17
3.1.1 <i>e-Leaning</i>	17
3.1.2 LMS.....	18
3.1.3 <i>Media Player</i>	18
3.1.4 SCORM	19

3.1.5	Ferramenta de autoria	20
	mAuthor.....	21
	Smart Questions	22
3.1.6	Kaltura.....	23
3.1.7	WebGL	24
3.1.8	TypeScript.....	24
3.1.9	React.....	26
3.1.10	Git.....	27
	Gitflow	27
3.2	Plataformas LMS e distribuição de conteúdo digital	29
3.2.1	Escola Virtual	29
3.2.2	Aula Digital.....	31
	Smart - Aula Digital	32
	Aula Digital (App).....	33
	Infinitas Learning	33
3.2.3	Moodle	33
	Moodle App	34
3.2.4	Blackboard Learn.....	35
3.3	Media Players	37
3.3.1	Player de vídeo	37
	Elemento HTML <video>	37
	Kaltura <i>Player</i>	37
	YouTube Iframe Player API.....	38
3.3.2	icPlayer	38
3.3.3	SQ Player	39
3.3.4	PDF Viewer	39
	PDFObject	39
	Adobe PDF Embed API	40
	React PDF.....	41
	React PDF Viewer	42
3.3.5	Office <i>Viewer</i>	42
	Microsoft Office <i>Viewer</i>	43
	Google Docs Viewer	43

React-doc-viewer	43
3.3.6 <i>Player</i> de áudio	44
Elemento HTML <audio>	44
React Audio Player	44
React H5 Audio Player	45
4 Desenho da Solução.....	47
4.1 Análise do módulo Smart Player da Escola Virtual	47
4.1.1 Incoerência da qualidade da exibição de recursos	47
4.1.2 Limitações de recolha de dados na plataforma	48
4.1.3 Inflexibilidade da plataforma.....	49
4.2 Tecnologias utilizadas	49
4.2.1 Linguagem de programação e <i>framework</i>	49
4.2.2 Controlo de versões e disponibilização.....	50
4.2.3 Ambiente de desenvolvimento integrado	50
4.3 Conceitos principais da biblioteca React	50
4.3.1 Componentes e JSX	51
4.3.2 Estrutura de uma aplicação React	51
4.3.3 Estados e ciclo de vida de um componente	52
4.3.4 Componentes Class e Componentes Function	53
4.3.5 Padrão <i>Container/Presentational</i>	54
4.3.6 Redux	54
4.4 Desenho da aplicação.....	56
4.4.1 JSON da Sequência	57
4.4.2 Componente Sequence	58
4.4.3 Componente Resource	58
4.4.4 Componentes <i>Player</i>	58
4.4.5 Componente <i>Header</i>	59
4.4.6 <i>Store</i> Redux	59
4.4.7 Componente PersistenceManager	59
4.5 Desenho da interface gráfica.....	60
4.6 Autenticação e segurança.....	61
5 Desenvolvimento da solução	63
5.1 Estruturação da aplicação	63

5.2	Criação do estado principal da aplicação.....	64
5.3	Navegação de sequência e componente <i>header</i>	65
5.4	Apresentação de conteúdo	65
5.4.1	<i>Observers</i> e redimensionamento.....	66
5.4.2	Recursos áudio.....	67
5.4.3	Documentos.....	68
5.4.4	Recursos <code>htmlwrap_noscorn</code>	69
5.4.5	Recursos <code>htmlwrap_scorm</code>	71
5.4.6	Imagens.....	72
5.4.7	PDF.....	72
5.4.8	Recursos Smart Questions.....	73
5.4.9	Vídeos.....	74
5.4.10	<i>Weblinks</i>	76
5.5	Submissão de resultados e dados analíticos.....	77
5.6	Integração com a Escola Virtual	78
5.6.1	Componente <i>header</i> na Escola Virtual	79
5.6.2	Compilação, disponibilização e integração.....	80
5.7	Validação da aplicação.....	80
6	Conclusões	83
6.1	Síntese.....	83
6.2	Objetivos realizados	83
6.3	Trabalho futuro.....	84
7	Referências	87

Lista de Figuras

Figura 1- Smart Player Escola Virtual. Retirado do vídeo de Youtube “Escola Virtual Tutorial – Disciplinas”	3
Figura 2 - Modelo de Negócio <i>Canvas</i> da plataforma Escola Virtual.	9
Figura 3 - Modelo de Desenvolvimento de Conceito Novo (NCD). Fonte (Koen, et al., 2004) (adaptado para português)	12
Figura 4 - É possível identificar semelhanças na interface gráfica do VLC Media Player e do Windows Media Player.	19
Figura 5 - Exemplo de conteúdo mAuthor em diversos dispositivos. Fonte (mAuthor, s.d.)	21
Figura 6 - Exemplo de um teste SQ. Retirado do vídeo de Youtube “Escola Virtual Tutorial - Elaboração de um teste: criação de questões”	22
Figura 7 - Jogo QuizEV em contexto sala de aula. Retirado do vídeo de Youtube “Escola Virtual Tutorial - Jogar um QuizEV”	23
Figura 8 - Jogo WebGL "Eu Leio" para alunos do primeiro e segundo ano escolar, disponível na Escola Virtual. Fonte (Escola Virtual - Eu Leio, 2022).	24
Figura 9 – Respostas à pergunta “Na sua opinião, qual dos seguintes cenários quanto ao futuro do TypeScript lhe parece mais provável de acontecer?”. Fonte (The Software House - State of frontend, 2022).	25
Figura 10 - Gráfico resumindo a resposta à pergunta "Ao longo do ano passado, quais das seguintes <i>framework</i> usou e gostou?". Fonte (The Software House - State of frontend, 2022).	26
Figura 11 - Gráfico contendo as respostas à pergunta "Quais os principais sistemas de controlo de versão que utiliza? Selecione todos os que se apliquem.". Fonte (Stack Overflow - Survey, 2022).	27
Figura 12 - Representação gráfica do modelo de trabalho Gitflow. Fonte (Atlassian - Gitflow, s.d.)	28
Figura 13 - EV 360º - Solução à medida. Fonte (Escola Virtual - EV 360º, 2022).	30
Figura 14 - Oferta da Aula Digital para professores (Aula Digital - Recursos, s.d.)	32
Figura 15 – Demonstração da possibilidade de descarregar o curso para uso offline na aplicação moodle. Fonte (Moodle App Features, s.d.)	34
Figura 16 - Ecrã de login da plataforma Blackboard da Universidade do Minho. Retirado de (Universidade do Minho - Blackboard-UMinho, 2022)	35

Figura 17 – Opções de customização do Kaltura Player. Fonte (Kaltura Player, s.d.).....	38
Figura 18 – Exemplo do utilitário PDFObject no navegador Opera. Fonte (PDFObject, s.d.)	40
Figura 19 - Configuração dinâmica do Adobe PDF Embed API, na demonstração disponível no website da API. Fonte (Adobe PDF Embed API Quickstarts, s.d.)	41
Figura 20 - Demonstração do React PDF Viewer. Fonte (React PDF Viewer, s.d.).....	42
Figura 21 - Demonstração do elemento HTML audio. Fonte: (HTML Audio, s.d.)	44
Figura 22 - Demonstração do componente React H5 Audio Player. Fonte (React H5 Audio Player, 2022).....	45
Figura 23 - Exemplo de uma estrutura com três iframes para apresentação de recursos Smart Questions na Escola Virtual.	48
Figura 24 – Diagrama de componentes representativo da estrutura de uma aplicação React.	51
Figura 25 - Ciclo de vida de um componente React. Adaptado da fonte (React lifecycle methods diagram, s.d.).....	52
Figura 26 -Demonstração do padrão <i>container/presentational</i>	54
Figura 27 – Demonstração da propagação de atualizações com e sem Redux. Adaptado de (Lima, 2020).....	55
Figura 28 - Princípios base do Redux. Fonte (Roy, 2022).	56
Figura 29 - Diagrama de componentes da aplicação.	Erro! Marcador não definido.
Figura 30 - Exemplo de um ficheiro JSON com dados necessários ao funcionamento da aplicação projetada.	58
Figura 31 – Design do componente <i>header</i> a incluir no LMS da Escola Virtual. Criado pelos designers da Porto Editora para este projeto.	60
Figura 32 - SequenceState criado para a aplicação.....	64
Figura 33 - Demonstração da class ResourceData, usada para tipificar os dados de cada Recurso no SequenceState.....	64
Figura 34 - Componente <i>header</i> por omissão da aplicação.....	65
Figura 35 - Componentes <i>Player</i> criados no âmbito deste projeto.	66
Figura 36 -Diagrama do funcionamento de um <i>observer</i> . Adaptado de (Javascript Patterns - Javascript Observer, s.d.)	66
Figura 37 – Exemplo da utilização do padrão <i>observer</i> para o redimensionamento de uma iframe de um recurso Smart Questions.	67
Figura 38 – Demonstração da utilização do <i>media player</i> React Audio Player no projeto.....	67

Figura 39 – Diagrama do componente DocumentsPlayer seguindo o padrão <i>Container/Presentational</i> com múltiplos <i>presentational</i>	Erro! Marcador não definido.
Figura 40 - Demonstração da funcionalidade de Download de documentos.....	69
Figura 41 - Demonstração do Microsoft Office Viewer no módulo de distribuição desenvolvido.	69
Figura 42 - Demonstração de um jogo HTML5 para a disciplina e inglês.	70
Figura 43 - Exemplo de <i>observer</i> criado para o redimensionamento de recursos que utilizam o elemento HTML canvas.....	70
Figura 44 - Demonstração do recurso mAuthor “A 1ª Guerra Mundial: as frentes do conflito” apresentado em dois <i>layouts</i> diferentes.	71
Figura 45 – Cálculo da escala baseado na largura disponível e largura do conteúdo.	72
Figura 46 - Demonstração de um PDF apresentado pelo novo módulo de distribuição desenvolvido.	73
Figura 47 - Demonstração de um recurso SQ apresentado no novo módulo de distribuição. .	74
Figura 48 - Diagrama do componente VideoPlayer seguindo o padrão <i>Container/Presentational</i> com múltiplos <i>presentational</i>	75
Figura 49 – Demonstração do <i>Player</i> Kaltura no novo módulo de distribuição de conteúdo...	75
Figura 50 - Demonstração da apresentação de vídeos mp4 no módulo de distribuição de conteúdo.....	76
Figura 51 - Diagrama do componente WebLinksPlayer seguindo o padrão <i>Container/Presentational</i> com dois <i>presentational</i>	77
Figura 52 - Demonstração da página de aviso para recursos <i>weblink</i>	77
Figura 53 - Demonstração da funcionalidade "pré-visualizar" dentro da plataforma Escola Virtual, utilizando a aplicação de distribuição de conteúdo desenvolvida no decorrer deste projeto. Imagem retirada da plataforma Escola Virtual.	78
Figura 54 - Cabeçalho desenvolvido para a Escola Virtual.....	79
Figura 55 – Demonstração do menu “Guardar” do cabeçalho desenvolvido para a Escola Virtual.....	79
Figura 56- Demonstração do esquema de visualização do cabeçalho quando em layout mobile.	80
Figura 57 – Excerto do ficheiro Excel entregue pela equipa de controlo de qualidade com os resultados dos primeiros testes funcionais efetuados num ambiente de pré-produção.....	81

Lista de Tabelas

Tabela 1 – Conceptualização de Valor para o cliente, Benefícios e Sacrifícios. Fonte (Woodall, 2003) (adaptado para português)	10
Tabela 2 - Versões de SCORM disponíveis. Fonte (SCORM.com - Evolution, s.d.).	20
Tabela 3 – Principais tipos de recursos disponibilizados pelo LMS da Escola Virtual.	31
Tabela 4 - Tipos de ficheiros reconhecidos pela plataforma Blackboard Learn. Fonte (Blackboard - Supported File Types, s.d.)(adaptado para português)	36
Tabela 5 - Versões ativas da <i>framework</i> Angular. Adaptado de (Angular versioning and releases, s.d.).....	49
Tabela 6 - Grau de realização dos requisitos funcionais propostos.....	84

Acrónimos e Símbolos

Lista de Acrónimos

API	<i>Application Programming Interface</i>
CMS	<i>Content Management System</i>
DEI	Departamento de Engenharia Informática [do ISEP]
eDAM	<i>e-Learning Digital Asset Management</i>
EV	Escola Virtual
FE	<i>Front-End</i>
FFE	<i>Fuzzy Front End</i>
LMS	<i>Learning Management System</i>
HTML	<i>HyperText Markup Language</i>
icPlayer	<i>Interactive Content Player</i>
IDE	<i>Integrated Development Environment</i> (Ambiente de desenvolvimento integrado)
ISEP	Instituto Superior de Engenharia do Porto
NPD	<i>New Process Development</i>
OS	<i>Operating System</i>
RTP	Rádio e Televisão de Portugal
SaaS	<i>Software as a Service</i>
SCORM	<i>Shareable Content Object Reference Model</i>
SIIS	Sistemas de Interação e Inovação Social
SQ	<i>Smart Questions</i>
SP	<i>Smart Player</i>
VA	<i>Value Analysis</i> (Análise de Valor)
VC	<i>Value for the Customer</i> (Valor para o Cliente)

1 Introdução

No presente capítulo introdutório é apresentada uma contextualização, assim como o problema e os objetivos esperados desta dissertação, a metodologia, a motivação e uma breve descrição da estrutura do documento.

1.1 Contextualização

Nos últimos anos, verifica-se a crescente utilização de dispositivos eletrónicos (telemóveis, tablets e computadores) pelas gerações mais jovens em Portugal. Num artigo da Forbes Portugal (Viage, 2020) sobre um estudo da empresa Dashlane, o autor escreve “pelo menos oito em cada dez crianças (80%), com idades compreendidas entre os cinco e os doze anos, usa um dispositivo eletrónico com acesso à internet”.

Acompanhando este crescimento verifica-se também o aumento da procura de plataformas digitais de apoio ao Estudo. Estas plataformas disponibilizam aos seus utilizadores conteúdos de várias áreas letivas, tentando complementar ou até mesmo substituir as aulas dos professores.

Com o início da pandemia Covid-19 em 2020, e as consequentes medidas de confinamento implementadas pelo governo português em março do mesmo ano, as escolas e institutos de educação superior viram-se forçados a encerrar e a rapidamente adotar modelos de ensino totalmente digitais.

Durante o primeiro confinamento em Portugal verificou-se uma rápida adoção de software de videoconferência por parte das escolas, como o Microsoft Teams, o Google Meet e o Zoom

Meetings, procurando uma solução para substituir as aulas presenciais (Facebook, Microsoft e Zoom: quem ganha a guerra da videoconferência?, 2020).

No mesmo período, as editoras Grupo Porto Editora e Grupo LeYa disponibilizaram temporariamente as suas respetivas plataformas de ensino e aprendizagem, Escola Virtual e Aula Digital, de forma gratuita (Porto Editora e LeYa dão acesso gratuito a plataformas virtuais para apoiar alunos sem aulas, 2020). Conteúdos destas plataformas foram ainda exibidos no projeto #EstudoEmCasa, uma parceria entre o Ministério da Educação português e o canal de televisão público RTP (Rádio e Televisão de Portugal), cujo objetivo foi a distribuição de blocos pedagógicos através da televisão, permitindo que estes conteúdos chegassem aos alunos com maior dificuldade de acesso à internet (#EstudoEmCasa 20-21, 2020).

Este cenário anormal provocou um afluente de novos utilizadores nas plataformas de ensino à distância e revelou limites nas soluções disponíveis atualmente. Neste documento estas limitações serão exploradas, um novo módulo de distribuição e visualização de conteúdos educativos será projetado e desenvolvido para o caso em estudo, o LMS da Escola Virtual, procurando explorar as tecnologias e métodos de distribuição utilizados e responder à questão “Qual a melhor forma de distribuir e visualizar recursos educativos interativos numa plataforma de *e-Learning*?”.

1.2 O Problema

Com a disponibilização gratuita de plataformas de *e-Learning*, durante o período de confinamento, a sua utilização atingiu níveis muito superiores ao normal, provocando vários problemas de estabilidade. Numa entrevista à revista Exame Informática publicada a 25 de março de 2020, sobre problemas com a plataforma Escola Virtual, o responsável de comunicação da Porto Editora, Paulo Gonçalves, explica: “O tempo de utilização e número de utilizadores diários é cerca de cinco a seis vezes o que tínhamos antes de março” (Gonçalves, 2020).

Para melhor explorar este problema será estudado o caso do Grupo Porto Editora e da sua plataforma de ensino, a Escola Virtual. Um dos principais módulos desta plataforma de *e-Learning* é o Smart Player, responsável pela distribuição e exibição dos vários tipos de conteúdo disponível no LMS (*Learning Management System*). Desenvolvido com recurso à *framework* de

desenvolvimento Angular, este componente depende de vários *media players* para a distribuição e apresentação de recursos educativos.

Para além das suas responsabilidades no projeto Escola Virtual, o módulo Smart Player tem ainda obrigações para com outras aplicações de *e-Learning* do Grupo Porto Editora, como a aplicação EV Smart Book e a aplicação de gestão de conteúdo da plataforma, o eDAM (*e-Learning Digital Asset Management*).



Figura 1- Smart Player Escola Virtual. Retirado do vídeo de Youtube “Escola Virtual | Tutorial – Disciplinas”¹

Com a afluência de novos utilizadores na plataforma, a Porto Editora a reforçou a sua infraestrutura e analisou potenciais pontos de melhoria. Relevantes para a presente dissertação, foram identificados os seguintes pontos críticos no módulo Smart Player:

- **Incoerência da qualidade da exibição de recursos** – Este ponto foi especialmente notado na utilização da plataforma em dispositivos móveis de menores dimensões, pondo em causa a qualidade do serviço.
- **Limitações de recolha de dados na plataforma** – A plataforma utiliza serviços externos para a captura de dados de utilização. Com a abundância de novos utilizadores, causada pela

¹ Vídeo disponível em: https://youtu.be/_XQfZqKVScM

pandemia, a empresa detetou limites nos dados recolhidos, assim como o aumento dos custos deste serviço.

- **Inflexibilidade da plataforma** – Foram detetadas dificuldades na expansão das funcionalidades disponíveis na plataforma. No contexto da pandemia, esta inflexibilidade limitou a capacidade de resposta da plataforma às necessidades dos milhares de novos utilizadores.

1.3 Objetivos e contributos esperados

Como já referido anteriormente neste documento, o principal objetivo desta dissertação é procurar responder à pergunta “Qual a melhor forma de distribuir e visualizar recursos educativos interativos numa plataforma de *e-Learning*?”. Esta dissertação irá por isso analisar e documentar o desenvolvimento de um módulo de distribuição e apresentação de conteúdos educativos para a plataforma de *e-Learning* Escola Virtual do Grupo Porto Editora, registando as tecnologias e métodos de distribuição utilizados e promovendo boas práticas de engenharia na sua projeção e implementação.

A Porto Editora colocou como objetivos os seguintes requisitos:

- **Atualização do módulo para uma tecnologia mais recente:** o módulo de distribuição de conteúdos, criado com a *framework* Angular 1.2.28, está severamente desatualizado, criando entraves nas suas capacidades. A empresa pretende que a nova aplicação seja desenvolvida com tecnologias mais recentes e com a possibilidade de facilmente ser atualizada.
- **Capacidade de apresentação correta em dispositivos móveis e computadores:** a correta apresentação de todos os tipos de recursos do LMS da Escola Virtual em dispositivos com ecrãs de diversas resoluções, com especial atenção às capacidades de reprodução em dispositivos móveis.
- **Capacidade de expansão:** a empresa pretende com este desenvolvimento adquirir maior flexibilidade de expansão do leque de conteúdos educativos interativos disponíveis nas suas plataformas de aprendizagem.

- **Capacidades de funcionamento *offline*:** o novo módulo deve ser desenvolvido com uma abordagem *offline-first*. Isto é, a aplicação desenvolvida deve estar pronta para funcionamento sem acesso à internet.
- **Potencial para recolha de dados analíticos:** A empresa pretende futuramente ser capaz de capturar dados de utilização dos seus recursos educativos internamente, eliminando assim a atual dependência de serviços externos como o Google Analytics.

1.4 Metodologia

Este projeto segue uma metodologia de estudo de caso, utilizando o desenvolvimento de um novo módulo de distribuição e visualização de conteúdo educativo para a plataforma de ensino-aprendizagem da Porto Editora, a Escola Virtual, para responder à questão colocada.

Havendo um grande foco em *media players*, este projeto requer o estudo detalhado de múltiplos meios e formas de distribuir conteúdo educativo em contexto digital. Inicia-se por isso este estudo com uma pesquisa exploratória, com levantamentos bibliográficos e análise de exemplos de forma a familiarizar o problema e facilitar a compreensão da solução proposta. Esta análise inclui a descrição de conceitos chave e a distinção de diferentes plataformas e *softwares* no contexto da distribuição de conteúdo educativo.

A avaliação do trabalho prático desenvolvido é feita pela equipa de controlo de qualidade da Porto Editora, sendo elaborado uma série de testes funcionais em dispositivos com diversas configurações de *hardware*, sistema operativo e resolução de ecrã.

1.5 Motivação

A escolha deste projeto surgiu do meu interesse profissional na área do ensino digital, fazendo parte da equipa de desenvolvimento *e-Learning* da Porto Editora, há já cinco anos. A entrada nesta equipa foi possibilidade por um projeto de estágio académico para finalização da Licenciatura em Engenharia Informática do ISEP, no âmbito da unidade curricular de PESTI. Desde então desempenho funções de desenvolvimento *front-end* na Escola Virtual, incluindo atualizações e melhorias no módulo de distribuição e visualização de conteúdo da plataforma.

A presente dissertação possibilita, uma vez mais, juntar a minha progressão académica ao meu percurso profissional.

1.6 Estrutura do Documento

O presente documento está dividido em capítulos e segue a seguinte estrutura:

Introdução – O primeiro capítulo contém uma contextualização seguida de uma descrição do problema. Segue uma descrição dos objetivos e contributos esperados, a metodologia e motivação para a escolha deste projeto e, por último, é descrita a estrutura do documento.

Análise de Valor – Neste capítulo é explorado o modelo de negócio da Escola Virtual, avaliando o valor que este projeto traz para os utilizadores da plataforma e o processo de inovação.

Estado da Arte – O terceiro capítulo apresenta conceitos fundamentais para a compreensão do tema deste documento, assim como uma exploração de algumas das plataformas LMS mais relevantes no mercado português da educação à distância. Para finalizar este capítulo, é apresentada uma análise de vários *media players* focados na apresentação de vários tipos de conteúdo multimédia.

Desenho da Solução – No capítulo quatro é explorado o desenho de uma solução para o problema, começando por analisar os problemas detetados no módulo de distribuição de conteúdo da Escola Virtual. Segue uma breve enumeração das tecnologias escolhidas para a solução e a descrição do funcionamento da biblioteca React. O capítulo finaliza apresentando um diagrama de componentes da aplicação e a interface gráfica.

Desenvolvimento da Solução – Este capítulo foca no projeto desenvolvido ao longo desta dissertação. O capítulo começa por descrever o processo de criação do projeto e de alguns dos componentes estruturais implementados. Em seguida é apresentada uma descrição dos vários componentes criados para a apresentação de conteúdo, assim como os *media players* utilizados. O capítulo termina com a descrição do processo de submissão de resultados, a integração da aplicação na Escola Virtual e o processo de validação da mesma por parte da equipa de controlo de qualidade da Porto Editora.

Conclusões – Por último é apresentada uma reflexão sobre o trabalho realizado, o que foi possível atingir e o trabalho que deve ser realizado no futuro.

2 Análise de Valor

“A Análise de Valor é um processo sistemático, formal e organizado de análise e avaliação. É uma atividade que requer planeamento, controlo e coordenação”¹ (Rich & Holweg, 2000).

Este capítulo foca na análise do potencial deste projeto através da análise da capacidade de o desenvolvimento proposto trazer valor para os clientes. Este capítulo inicia com a apresentação do modelo de negócio associado à Escola Virtual. Em seguida é definido o que se pode considerar Valor para o Cliente (VC) e o processo de inovação.

2.1 Modelo de negócio *Canvas*

“Um modelo de negócios descreve a lógica de como uma organização cria, entrega e captura valor”² (Osterwalder & Pigneur, 2013). Um modelo de negócio contempla os seguintes aspetos:

1. **Segmentos de clientes:** Identificação dos clientes que a organização/produto pretende servir. Grupos de clientes representam novos segmentos quando requerem ofertas diferentes, são alcançados por métodos de distribuição diferentes, requerem novos tipos de relacionamento, estão dispostos a pagar por aspetos diferentes na oferta ou apresentam lucros numa escala substancialmente diferente.

¹ Tradução livre do autor. No original “*Value Analysis (and Value Engineering) is a systematic, formal and organized process of analysis and evaluation. It is not haphazard or informal and it is a management activity that requires planning, control and co-ordination*” (Rich & Holweg, 2000).

² Tradução livre do autor. No original “*A business model describes the rationale of how an organization creates, delivers and captures value*” (Osterwalder & Pigneur, 2013)

2. **Proposta de Valor:** O problema ou necessidade do cliente que a organização/produto pretende resolver ou satisfazer. Mais informação sobre o valor gerado para os clientes pode ser encontrada na secção 2.2 deste capítulo.
3. **Canais:** A forma como o valor chega ao cliente. Trata-se de venda direta e pessoal ao cliente, ou venda direta impessoal (através da internet por exemplo). A avaliação de canais de distribuição deve obedecer a uma regra de custo benefício, utilizando tempo de entrega, custo monetário e quantidade de segmentos de clientes que poderá servir.
4. **Relação com o Cliente:** O tipo de relação que se pretende manter com cada segmento de clientes. Este ponto é fundamental para atingir não só a conquista de clientes, mas também a manutenção dos mesmos a longo prazo e potencializar a ampliação das vendas.
5. **Fontes de Lucro:** O valor gerado para a empresa. Pode ter a forma de vendas, taxas de uso, taxas de subscrição, taxas de empréstimo, taxas de licenciamento, publicidade ou taxas de intermediário.
6. **Recursos Chave:** Recursos requeridos para a distribuição, como matérias-primas, acesso a uma frente de loja (física ou virtual), recursos humanos ou infraestrutura.
7. **Atividades Chave:** Atividades que devem ser feitas para alcançar os objetivos do negócio, podendo ser categorizadas como atividades de produção, atividades de resolução de problemas ou atividades de plataforma (quando uma plataforma externa é um recurso chave).
8. **Parcerias Chave:** Parceiros fundamentais para o funcionamento do negócio. Parcerias são formadas para otimizar a economia de escala, redução de risco e incerteza, ou aquisição de recursos essenciais ao negócio.
9. **Estrutura de Custos:** Os custos de operação do modelo de negócio descrito.

É importante para a compreensão deste projeto entender o modelo de negócio da Escola Virtual.









Parcerias Chave  <ul style="list-style-type: none"> • Porto Editora. • Areal Editora. • Raiz Editora. 	Atividades Chave  <ul style="list-style-type: none"> • Desenvolvimento e manutenção do website; • Criação de novos recursos digitais educativos; • Divulgação da plataforma. 	Proposta de Valor  <ul style="list-style-type: none"> • Acesso a conteúdo educativo de grande qualidade; • Acesso a livros digitais; • Facilitar planeamento de aulas; • Dinamizar a sala de aula; • Acompanhamento remoto do progresso de um aluno; • Promoção de métodos de estudo; 	Relação com o Cliente  <ul style="list-style-type: none"> • Suporte ao cliente. • Eventos de formação de professores. • <i>Webinars</i>. • Boletins de notícias. 	Segmentos de clientes  <ul style="list-style-type: none"> • Instituições de Ensino Primário e Secundário (EPS). • Professores de todas as áreas do EPS. • Alunos do pré-escolar. • Alunos do EPS. • Alunos em preparação para exames nacionais. • Alunos estrangeiros que pretendam aprender Português. • Encarregados de educação.
	Recursos Chave  <ul style="list-style-type: none"> • <i>Hardware</i>; • <i>Software</i>; • Livros digitais. • Recursos digitais educativos. 		Canais  <ul style="list-style-type: none"> • Venda direta. 	
Estrutura de Custos  <ul style="list-style-type: none"> • Aquisição e manutenção de <i>hardware</i>; • Aquisição de <i>software</i>; • Aquisição de recursos digitais educativos; • Recursos humanos. 		Fontes de Lucro  <ul style="list-style-type: none"> • Comercialização do produto. • Publicidade a livros físicos e digitais das Editoras. 		

Figura 2 - Modelo de Negócio *Canvas* da plataforma Escola Virtual.

2.2 Valor para o Cliente e Valor Percebido

"O valor para o cliente (VC) é qualquer percepção pessoal das vantagens provenientes da associação do cliente com a oferta de uma organização, e pode ocorrer na forma de redução no sacrifício; presença de benefício (percebido como atributo ou resultado); o resultante de qualquer combinação ponderada de sacrifício e benefício (determinada e expressada de forma racional ou intuitiva); ou uma agregação, ao longo do tempo, de uma ou de todas estas"¹ (Woodall, 2003). Na secção 2.1, foi explorado o modelo de negócio, incluindo no mesmo a proposta de valor da plataforma Escola Virtual. Nesta secção é explorado o valor que poderá ser acrescentado com os desenvolvimentos deste projeto.

Na Tabela 1 encontram-se exemplos de benefícios e sacrifícios que poderão determinar o valor de um produto para um cliente.

¹ Tradução livre do autor. No original "Value for the customer (VC) is any demand-side, personal perception of advantage arising out of a customer's association with an organisation's offering, and can occur as reduction in sacrifice; presence of benefit (perceived as either attributes or outcomes); the resultant of any weighed combination of sacrifice and benefit (determined and expressed either rationally or intuitively); or an aggregation, over time, of any or all of these." (Woodall, 2003).

Tabela 1 – Conceptualização de Valor para o cliente, Benefícios e Sacrifícios. Fonte (Woodall, 2003) (adaptado para português)

Benefícios		Sacrifícios
Atributos	Resultados	
Qualidade percebida	Benefício funcional	Preço
Qualidade do produto	Utilidade	Preço no mercado
Qualidade	Função de utilização	Custos Monetário
Qualidade de Serviço	Função estética	Financeiro
Qualidade Técnica	Benefício operacional	Custos
Qualidade funcional	Economia	Custo de utilização
Qualidade de desempenho	Benefício logístico	Custos percebidos
Desempenho de serviço	Benefícios do produto	Custos de procura
Serviço	Benefícios estratégicos	Custos de aquisição
Suporte ao Serviço	Benefícios financeiros	Custos de oportunidade
Aspetos especiais do serviço	Resultados para o cliente	Custos de entrega e instalação
Serviços adicionais	Benefícios sociais	Custos de reparação
Solução principal	Segurança	Custos de manutenção e treino
Personalização	Conveniência	Custos não monetários
Confiabilidade	Prazer	Custos não financeiros
Características do Produto	Agradecimento dos utilizadores	Custo de relacionamento
Atributos do Produto	Conhecimento, humor	Custo Psicológico
Recursos	Autoexpressão	Tempo
Desempenho	Benefícios Pessoais	Energia Humana
	Associação a grupos sociais	Esforço
	Excitação afetiva	

O projeto apresentado nesta dissertação analisa o desenvolvimento de um módulo de distribuição de recursos educativos para a Escola Virtual. Podemos considerar um módulo de distribuição como sendo o coração de um LMS, sendo a distribuição de conteúdo um dos principais propósitos de um LMS. No contexto específico da Escola Virtual, trata-se de uma substituição de um módulo existente por um mais atual e de melhor qualidade.

Podemos por isso concluir que o projeto apresentado se traduz em valor para os utilizadores da plataforma na forma de aumento da qualidade, acessibilidade e confiabilidade do serviço prestado pelo LMS. O serviço torna-se ainda mais conveniente, com a possibilidade de aceder ao conteúdo em dispositivos de menores dimensões. Este projeto não irá afetar o preço da licença da Escola Virtual, não trazendo qualquer sacrifício extra para os utilizadores, para além daqueles que já afetavam a plataforma. É ainda relevante o facto deste desenvolvimento potenciar a adição de novos tipos de conteúdo educativo interativo no futuro, o que potencia a adição de ainda mais valor para o cliente.

2.3 Processo de inovação

O processo de inovação, como descrito por Peter Koen (2004) “... pode ser dividido em três áreas: o *front-end* difuso, o processo de desenvolvimento de novos produtos e a comercialização”¹ (Koen, et al., 2004)

- **Front-end difuso:** Do inglês “*Fuzzy Front End*” (FFE), também chamado de “front-end da inovação” por Peter Koen (2004), é o primeiro passo no processo de inovação, não tendo estruturas formais e sendo por isso imprevisível.
- **Desenvolvimento de novos processos:** Do inglês “*New Process Development*” (NPD), marca a passagem para a fase de estruturação formal do processo de inovação.
- **Comercialização:** A fase final e objetivo de um processo de inovação.

¹ Tradução livre do autor. No original “*The innovation process may be divided into three areas: the fuzzy front end (FFE), the new product development (NPD) process, and commercialization...*” (Koen, et al., 2004).

2.3.1 Modelo de Desenvolvimento de Conceito Novo

O modelo NCD (*New Concept Development*) é um modelo relacional com objetivo de “...fornecer uma linguagem e definição comum para os principais componentes do FFE”¹ (Koen, et al., 2004). Este modelo é composto por três elementos.

- **O motor** – Representando a Organização e a sua influência, direta ou indireta no processo (como a liderança, a cultura ou a estratégia de negócio).
- **Elementos da atividade** – As cinco categorias controláveis do processo FFE: identificação de oportunidade, análise da oportunidade, geração de ideias, seleção de ideias e definição de conceitos.
- **Fatores influentes (o ambiente)** – Fatores externos e incontroláveis que influenciam o processo, como atividades da competição ou novas leis.

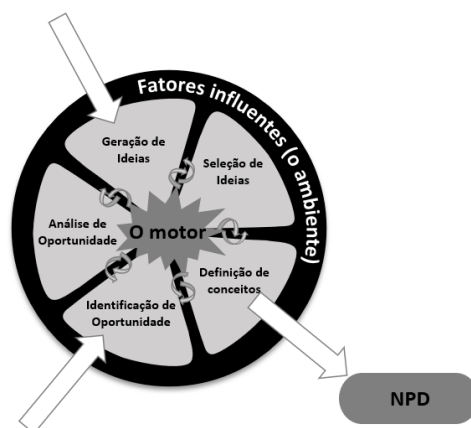


Figura 3 - Modelo de Desenvolvimento de Conceito Novo (NCD). Fonte (Koen, et al., 2004) (adaptado para português)

Expandindo um pouco a definição das cinco categorias controláveis do processo FFE:

1. **Identificação de Oportunidade:** “Este elemento é tipicamente impulsionado pelos objetivos de negócio. Por exemplo, a oportunidade pode ser uma resposta de curto prazo a uma ameaça da concorrência, um avanço que possibilita a captura de uma vantagem competitiva, ou um método de simplificar as operações, acelerá-las ou reduzir os seus custos. Pode ser

¹ Tradução livre do autor. No original “[The NCD] provides a common language and definition of the key components of the fuzzy front end (FFE)” (Koen, et al., 2004)

uma nova direção para o negócio ou uma melhoria a um produto existente.”¹ (Koen, et al., 2004).

2. **Análise de Oportunidade:** Trata-se de um primeiro processo de análise que conclui se uma oportunidade identificada se poderá converter em valor para a organização e deverá ser perseguida.
3. **Geração de Ideias:** “... o nascimento, desenvolvimento e maturação de uma ideia concreta. A geração de ideias é evolucionária. As ideias são construídas, destruídas, combinadas, remodeladas, modificadas e atualizadas. Uma ideia pode passar por múltiplas iterações e mudanças enquanto é examinada, estudada, discutida e desenvolvida em conjunto com outros elementos do modelo NCD. O contato direto com o cliente, utilizadores e ligações a outras equipas multidisciplinares, assim como colaborações com outras companhias e instituições, muitas vezes aprimora esta atividade.”² (Koen, et al., 2004).
4. **Seleção de Ideias:** Uma segunda fase de análise com objetivo de filtrar as ideias que nasceram no passo anterior. Geralmente o processo de geração origina um excedente de ideias, “o problema para a maioria dos negócios é a seleção de que ideias devem seguir de forma a atingirem o maior valor possível para o negócio.”³ (Koen, et al., 2004)
5. **Definição de conceitos:** Desta etapa é possível avançar de FFE para NPD. Para que tal possa acontecer, a ideia deve ser apresentada às entidades competentes da organização, com um bom caso de negócio ou como uma proposta tecnológica com valor à empresa. O caso de negócio deve consistir de informação qualitativa e quantitativa. Em caso de sucesso, a ideia passará para a próxima etapa do processo de Inovação (o NPD). Em caso de insucesso, a ideia

¹ Tradução livre do autor. No original “*This element is typically driven by the business goals. For example, the opportunity may be a near-term response to a competitive threat, a “breakthrough” possibility for capturing competitive advantage, or a means to simplify operations, speed them up, or reduce their cost. It could be an entirely new direction for the business or an upgrade to an existing product.*” (Koen, et al., 2004).

² Tradução livre do autor. No original “*The element of idea generation and enrichment concerns the birth, development, and maturation of a concrete idea. Idea generation is evolutionary. Ideas are built up, torn down, combined, reshaped, modified, and upgraded. An idea may go through many iterations and changes as it is examined, studied, discussed, and developed in conjunction with other elements of the NCD model. Direct contact with customers and users and linkages with other cross-functional teams as well as collaboration with other companies and institutions often enhance this activity.*” (Koen, et al., 2004)

³ Tradução livre do autor. No original “*The problem for most businesses is in selecting which ideas to pursue in order to achieve the most business value.*” (Koen, et al., 2004)

pode ficar dormente, à espera de novas oportunidades para aumentar a sua relevância, ou imediatamente reforçada e rerepresentada.

Aplicando o modelo NCD ao projeto desta dissertação:

1. **Identificação de Oportunidade:** Ocorreu após o início da pandemia Covid-19, provocada pelo vírus SARS-CoV-2, que se espalhou pelo mundo em 2020, e forçou à implementação de medidas de confinamento. O fecho de escolas foi uma destas medidas, aumentando a relevância e procura de plataformas de ensino à distância. A organização Porto Editora permitiu o acesso gratuita à sua plataforma de educação e aprendizagem, a Escola Virtual, registando grandes quantidades de novos utilizadores. Em simultâneo, notou-se uma maior utilização de dispositivos móveis para contexto de e-Learning.
2. **Análise da Oportunidade:** A disponibilização da plataforma gratuitamente a grandes quantidades de novos utilizadores forneceu à Porto Editora um grande volume de novos dados de utilização, permitindo identificar os seguintes problemas no sistema de distribuição de conteúdo da plataforma:
 - a. Exibição de recursos em dispositivos móveis inconsistente, colocando em causa a qualidade e principal missão da plataforma.
 - b. Persistência de dados de recursos (avaliativos e expositivos) incoerente, provocado por falhas nos eventos que gerem a persistência e submissão de dados.
 - c. Dificuldade na expansão das funcionalidades de acordo com as necessidades dos utilizadores face à situação pandémica, o que revelou uma rigidez indesejada nas capacidades de adaptação da plataforma.

Estes problemas inibem o potencial de crescimento da plataforma no mercado de *e-Learning* português, sendo por isso necessária uma solução.

3. **Geração de Ideias:** Várias soluções foram propostas para resolução dos problemas identificados:
 - a. Correção dos problemas no módulo de distribuição atual.

- b. Atualização do *software* do módulo atual e correção dos problemas detetados, permitindo também a adição de novas funcionalidades (com limitações).
 - c. Criação de um novo módulo de distribuição de conteúdo, de raiz, utilizando novas tecnologias e que permita não só a correção dos problemas já discutidos, mas que também facilite a adição de novas funcionalidades e tipos de conteúdo.
4. **Seleção de Ideias:** Ponderando as soluções encontradas, a opção c, criação de um novo módulo de raiz, foi escolhida. A possibilidade de mais facilmente expandir as funcionalidades do Smart Player no futuro foi o principal fator nesta decisão, abrindo portas a novas oportunidades de crescimento para a plataforma.
5. **Definição de conceitos:** O projeto foi apresentado e aprovado pela administração da organização, sendo adicionado ao planeamento da equipa de desenvolvimento *e-Learning* da Porto Editora.

3 Estado da Arte

Este capítulo começa por apresentar conceitos e tecnologias cujo conhecimento é importante para a compreensão do problema e da sua resolução. Segue uma análise de plataformas LMS e distribuição de conteúdo digital relevantes atualmente. Por fim uma análise de vários *media players*, focados na reprodução de vários tipos de recursos.

3.1 Conceitos e Tecnologia

Iniciamos o capítulo com a introdução de alguns conceitos fundamentais no contexto da educação digital, necessários para a total compreensão do projeto exposto neste documento, assim como algumas das tecnologias que serão exploradas na construção de uma solução ao problema.

3.1.1 *e-Learning*

O *e-Learning*, *Electronic Learning* ou educação online, é um processo de aprendizagem com recurso ao computador e à internet (unYLeYa O que é o e-Learning, s.d.). Várias ferramentas foram desenvolvidas para este efeito, nomeadamente LMS (secção 3.1.2 do presente documento). A experiência pode incluir a participação de um professor ou tutor, de forma síncrona ou assíncrona.

Plataformas de e-Learning oferecem várias vantagens quando comparadas a métodos de ensino tradicional (presencial):

- **Flexibilidade:** o conteúdo educativo fica disponível em permanência, permitindo a adoção de horários flexíveis e convenientes para o aluno.
- **Economia de tempo:** Otimização do tempo do aluno, cortando no tempo necessário para deslocamentos a estabelecimentos de ensino.
- **Customização da experiência:** o conteúdo apresentado ao utilizador pode ser personalizado de forma a maximizar a retenção de conhecimento. Para além disso o utilizador ganha total controlo do ritmo de aprendizagem, podendo repetir conteúdo que não compreendeu na totalidade.

O *e-Learning* ganhou ainda mais relevância e destaque como consequência da pandemia Covid-19, com início em 2020, e do impedimento do normal funcionamento de estabelecimentos de ensino presenciais, por todo o mundo.

3.1.2 LMS

Abreviatura de *Learning Management System* (Sistema de gestão de aprendizagem), um LMS é um programa com propósito de ajudar na gestão, criação, organização e distribuição de conteúdo educativo (Encyclopedia - LMS, s.d.).

Este tipo de software está geralmente dividido em pelo menos duas partes, uma interface administrativa e uma interface para utilizadores. A primeira permite a gestão do material educativo disponível, incluindo processos de criação, publicação, remoção e organização dos mesmos por parte de um administrador. A interface de utilizador é responsável pela distribuição desses conteúdos aos participantes (Easy LMS - What is a learning management system, s.d.).

Em algumas versões de LMS, dados de utilização são guardados pela plataforma e utilizados para criação de perfis cognitivos. Estes perfis podem ser utilizados para melhor orientar os utilizadores da plataforma, sugerindo conteúdo mais relevantes perante as suas necessidades, gostos e dificuldades.

3.1.3 Media Player

Um reproduzidor de média, do inglês *media player*, é um programa capaz de reproduzir ficheiros digitais de média ou multimédia. A maioria dos sistemas operativos tem incluído pelo menos um

media player, como é exemplo o Windows Media Player em computadores com Microsoft Windows.

Quando focados na reprodução do mesmo estilo de recursos, a maioria dos *media players* apresentam características semelhantes nas suas interfaces gráficas. Por exemplo, na interface gráfica de *players* focados na reprodução de áudio ou vídeo, é comum encontrar um botão para iniciar/parar a reprodução, uma representação gráfica de progresso, geralmente, em forma de barra e um controlador de volume.

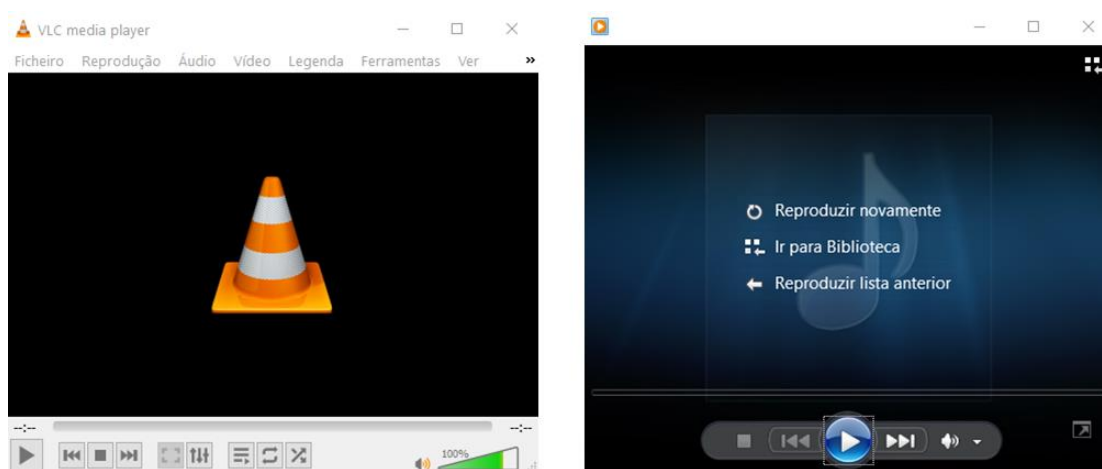


Figura 4 - É possível identificar semelhanças na interface gráfica do VLC Media Player e do Windows Media Player.

Em contexto de educação, a inclusão de pelo menos um *media player* num LMS é prática comum, de modo a facilitar a distribuição do conteúdo digital disponível, apresentando-o diretamente na plataforma e sem necessidade de instalar *software* adicional nos dispositivos dos utilizadores. Alguns destes *players* permitem interações entre o LMS e o recurso a ser apresentado, possibilitando a criação de recursos interativos e a submissão e armazenamento de dados na plataforma LMS.

3.1.4 SCORM

SCORM, abreviatura de *Sharable Content Object Reference Model* (Objeto Modelo de Referência para Conteúdo Partilhável), é um conjunto de padrões técnicos para a criação de conteúdos

partilháveis no contexto educacional. Este não é o único padrão técnico disponível neste contexto, mas é o mais amplamente utilizado (SCORM.com - Evolution, s.d.).

Conteúdos que seguem este padrão trazem como principal vantagem para o autor a fácil disponibilização do seu conteúdo em diversas plataformas LMS. Simultaneamente, o trabalho dos vários LMS é facilitado, conseguindo acomodar conteúdos interativos de diversas fontes de forma mais eficiente e evitando problemas de compatibilidade (SCORM.com, s.d.).

Para garantir este nível de compatibilidade o padrão SCORM é composto por três componentes: a especificação do formato a seguir, a especificação do método de execução e por último a especificação da sequência (SCORM.com - Explained, s.d.).

- **Especificação do formato a seguir:** este elemento dita como o conteúdo deve ser empacotado, garantindo que o pacote gerado é facilmente reconhecido por qualquer LMS.
- **Especificação do método de execução:** o segundo elemento define regras de execução, comunicação com LMS e o formato dos pacotes de dados a serem comunicados.
- **Especificação de sequência:** o último elemento descreve a navegação entre as diversas partes do recurso.

Tabela 2 - Versões de SCORM disponíveis. Fonte (SCORM.com - Evolution, s.d.).

Versão	Data de lançamento
SCORM 1.1	janeiro 2001
SCORM 1.2	outubro 2001
SCORM 2004 <i>2nd Edition</i>	julho 2004
SCORM 2004 <i>3rd Edition</i>	outubro 2006
SCORM 2004 <i>4th Edition</i>	março 2009

3.1.5 Ferramenta de autoria

Uma ferramenta de autoria é um *software* que “permite a criação de conteúdo digital. Pode ser algo tão simples como um documento Word ou tão complexo como uma ferramenta de design

gráfico. Quando aplicado ao *e-Learning*, o *software* permite ao utilizador gerar e manipular objetos multimédia para a finalidade pretendida para o conteúdo”¹ (Penfold, 2022).

mAuthor

Desenvolvido pela empresa Learnetic SA, o mAuthor é um *software* para criação de conteúdo digital interativo para fins educativos (ferramenta de autoria). Para a criação de aulas interativas mAuthor, o autor não necessita de conhecimentos de programação e a plataforma oferece vários tutoriais para facilitar o processo de formação de novos autores. A plataforma disponibiliza ainda um catálogo de amostras para demonstrar as suas capacidades e promover a utilização da totalidade das suas funcionalidades.

O conteúdo criado no mAuthor pode ser exibido em dispositivos com diferentes resoluções e tamanhos de ecrã. É possível adicionar vários tipos de multimédia aos recursos, como vídeos, áudios, jogos ou animações. O mAuthor disponibiliza ainda funcionalidades estatísticas que fornecem informações úteis para a adaptação dos conteúdos às necessidades e preferências dos utilizadores. O conteúdo pode ser exportado para um LMS seguindo os padrões SCORM para maior compatibilidade com a plataforma.



Figura 5 - Exemplo de conteúdo mAuthor em diversos dispositivos. Fonte (mAuthor, s.d.)

O mAuthor é uma das ferramentas utilizadas para a criação dos conteúdos educativos disponíveis no LMS Escola Virtual, representando um dos principais tipos de recurso expositivo da plataforma. Todo o conteúdo criado é exportado seguindo os padrões SCORM, existindo, no entanto, algum conteúdo mais antigo que não obedece a esta regra.

¹ Tradução livre do autor. No original “An authoring tool is a piece of software that enables the creation of digital content. This could be as simple as creating a Microsoft Word document, or as complex as a graphic design tool. Elearning authoring software allows the user to generate and manipulate multimedia objects for the content’s intended purpose.” (Penfold, 2022)

Smart Questions

Smart Questions ou SQ, é um módulo da Escola Virtual que permite a criação e apresentação de recursos avaliativos para uso dentro da plataforma. Pode por isso ser considerado uma ferramenta de autoria interna à plataforma.



Figura 6 - Exemplo de um teste SQ. Retirado do vídeo de Youtube “Escola Virtual | Tutorial - Elaboração de um teste: criação de questões” ¹.

Este módulo permite a administradores e utilizadores da Escola Virtual com conta de perfil Professor, a criação de testes interativos com vários tipos de questões e recursos multimédia associados. Estes conteúdos podem depois ser associados a hierarquias de aprendizagem, distribuídos como tarefa aos alunos das turmas de um professor, ou exportados para impressão, onde o professor recebe um ZIP contendo dois ficheiros Word, o teste, uma folha de soluções, e um documento Excel com uma grelha para correção e avaliação.

Para além de testes, esta ferramenta permite ainda a criação de perguntas para utilização em jogos *Quiz*, criados para contexto sala de aula, presencial ou à distância. Estes jogos têm como objetivo dinamizar o processo de aprendizagem em sala de aula.

¹ Vídeo disponível em: <https://youtu.be/NNGIUSKTyIE>

Os testes SQ são apresentados no módulo de distribuição da plataforma, sendo este responsável pela submissão dos resultados obtidos. Os jogos *Quiz* por outro lado, abrem numa aplicação React diretamente no LMS da Escola Virtual.



Figura 7 - Jogo QuizEV em contexto sala de aula. Retirado do vídeo de Youtube “Escola Virtual | Tutorial - Jogar um QuizEV”¹.

3.1.6 Kaltura

Kaltura é uma empresa de *software* norte-americana, com a missão de “... potencializar qualquer experiência de vídeo, para qualquer organização”² (Kaltura Everything Video, s.d.). A Kaltura oferece vários produtos em formato SaaS (*Software as a Service*), todos relacionados com reprodução ou gestão de ficheiros vídeo, incluindo um *player*, e um CMS (*Content Management System*).

O Kaltura *Video Content Management System* é uma plataforma de gestão de conteúdo fornecendo aos seus utilizadores várias funcionalidades, como a edição de vídeos, controlo de acesso, adição de legendas e acesso a dados de utilização.

O Kaltura é utilizado pela Porto Editora para a gestão, publicação e apresentação de conteúdo vídeo na plataforma Escola Virtual, sendo por isso uma importante tecnologia para incluir na solução desta dissertação.

¹ Vídeo disponível em: <https://youtu.be/ODTjnm1BOPk>.

² Tradução livre do autor. No original “*Our mission is to power any video experience, for any organization*” (Kaltura Everything Video, s.d.).

3.1.7 WebGL

WebGL é uma tecnologia baseado em OpenGL para apresentação de conteúdo gráfico 2D e 3D em navegadores de internet. É gratuito e multiplataforma, sendo suportado pela maioria dos navegadores. Existem duas versões desta API: WebGL 1.0, baseado em OpenGL ES 2.0, e WebGL 2.0, baseado em OpenGL ES 3.0. A Apple (Safari), Google (Chrome), Microsoft (Edge) e a Mozilla (Firefox) são membros do grupo de trabalho WebGL, garantindo assim a compatibilidade desta tecnologia nos navegadores de internet mais populares (Khronos Group, s.d.).

Uma das aplicações para a tecnologia WebGL é a criação e disponibilização de Jogos sérios diretamente no navegador do utilizador. Estes jogos podem ter carácter educativo, como é exemplo o jogo “Eu Leio” para aprendizagem de leitura da Porto Editora (Escola Virtual - Eu Leio, 2022).



Figura 8 - Jogo WebGL "Eu Leio" para alunos do primeiro e segundo ano escolar, disponível na Escola Virtual. Fonte (Escola Virtual - Eu Leio, 2022).

3.1.8 TypeScript

“TypeScript é uma linguagem de programação fortemente tipificada, baseada em JavaScript, oferecendo melhores ferramentas para todas as escalas”¹. Desenvolvida pela Microsoft, em regime “Código Aberto”, esta linguagem grátis tem como principal objetivo oferecer uma integração mais rigorosa com IDEs (*Integrated Development Environment*), facilitando o processo de programação, a descoberta de erros e gestão de projetos de grandes dimensões e

¹ Tradução livre do autor. No original “TypeScript is a strongly typed programming language that builds on JavaScript, giving you better tooling at any scale.” (Typescript, s.d.)

complexidade. Uma vez que se baseia em JavaScript, e quando compilada é convertida em JavaScript simples, esta linguagem partilha grande parte da sua sintaxe e pode ser utilizada em qualquer ambiente compatível (Typescript, s.d.).

Em 2020, num questionário realizado pelo popular *website* de programação, Stack Overflow, TypeScript foi considerado a segunda linguagem de programação mais adorada pelos programadores que a usam, com 67.1% de votos positivos. No mesmo questionário JavaScript ocupa o décimo lugar, com 58.3% dos inquiridos a expressar o seu gosto pela linguagem (Stack Overflow - Survey, 2020). Em 2022, num questionário semelhante, no mesmo *website*, TypeScript desce para quarta linguagem mais adorada pelos programadores. Ainda assim, a sua popularidade cresceu, tendo recebido 73.46% de votos positivos (Stack Overflow - Survey, 2022).

Segundo um relatório do *website* The Software House, em 2022 esta linguagem tem-se tornado dominante no desenvolvimento de *front-end*, sendo já utilizada por mais de 84% dos profissionais. No mesmo *website*, um questionário sobre o futuro desta linguagem indica que a grande maioria dos inquiridos considera que TypeScript se tornará o padrão de desenvolvimento em *front-end*, ultrapassando o JavaScript. O relatório refere inclusivamente que novos projetos comerciais já raramente são iniciados sem TypeScript (The Software House - State of frontend, 2022).

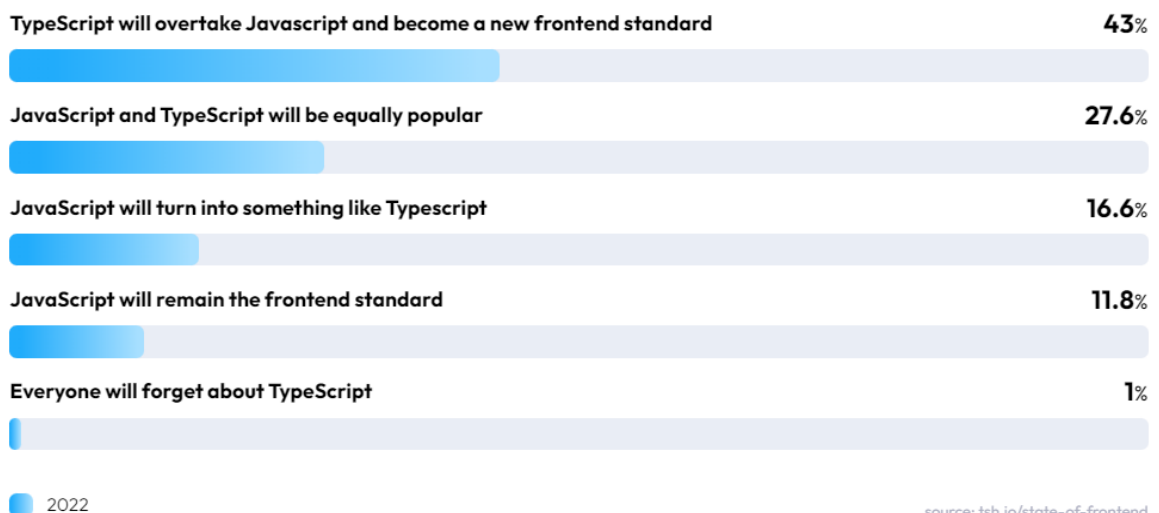


Figura 9 – Respostas à pergunta “Na sua opinião, qual dos seguintes cenários quanto ao futuro do TypeScript lhe parece mais provável de acontecer?”. Fonte (The Software House - State of frontend, 2022).

3.1.9 React

React é “uma biblioteca de JavaScript para a construção de interfaces de utilizador”¹ (React, 2022). Criada por Jordan Walke, esta biblioteca é atualmente desenvolvida pela Meta (antiga Facebook) seguindo o princípio de “Código Aberto”, contando assim com contributos de uma vasta comunidade online (React - Team, 2022). De acordo com dados recolhidos pelo website The Software House, React é a “*framework*” mais usada e apreciada por profissionais de *front-end* (The Software House - State of frontend, 2022).

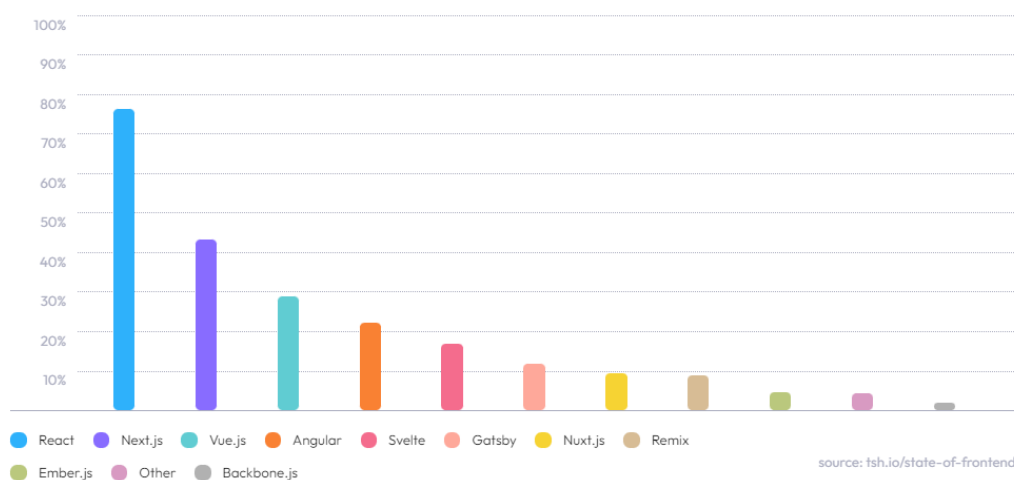


Figura 10 - Gráfico resumindo a resposta à pergunta "Ao longo do ano passado, quais das seguintes *framework* usou e gostou?". Fonte (The Software House - State of frontend, 2022).

Apesar de React ser uma biblioteca de JavaScript, é habitualmente referida como sendo uma *framework* de desenvolvimento. Existem múltiplas diferenças entre uma biblioteca e uma *framework*, sendo um dos principais fatores diferenciais o facto de uma *framework* controlar o fluxo de uma aplicação e ser responsável por chamadas a código. Por outro lado, uma biblioteca é um conjunto de funções chamadas dentro do fluxo de uma aplicação. No caso específico da biblioteca React, as funções disponibilizadas têm tamanho impacto na estruturação da aplicação que forçam a um ajuste do fluxo da mesma. Pode por isso dizer-se que a biblioteca React obtém um controlo indireto do fluxo da aplicação, sendo por isso referida como sendo uma *framework*.

A biblioteca React é compatível com TypeScript permitindo uma melhor gestão da complexidade da mesma (Create React App - Adding Typescript, s.d.), no entanto, alguns dos pacotes de

¹ Tradução livre do autor. No original “A JavaScript library for building user interfaces” (React, 2022).

funcionalidades opcionais poderão não conter suporte para esta linguagem. Nestes casos é possível misturar componentes JavaScript a uma aplicação React TypeScript, não sendo uma prática muito elegante para a consistência do código da aplicação.

Tratando-se da principal e mais versátil tecnologia para desenvolvimento *front-end* de momento, React será utilizado na solução proposta por esta dissertação.

3.1.10 Git

“Git é um sistema de controlo de versão gratuito e de código aberto, projetado para lidar com projetos de todos os tamanhos com velocidade e eficiência”¹ (Git, s.d.). Projetado por Linus Torvalds, criador do sistema operativo Linux, Git foi inicialmente desenvolvido com o objetivo de distribuir os ficheiros do sistema Linux, após o sistema de controlo de versões utilizado, BitKeeper, revocar ao projeto Linux o seu estatuto de uso gratuito (A Git Origin Story, 2018).

Git é o sistema de controlo de versões mais utilizado atualmente. No questionário de 2022, sobre tecnologias de controlo de versão, do *website* Stack Overflow, 93.87% dos 71,379 inqueridos, selecionou Git como o sistema de controlo de versão que utiliza (Stack Overflow - Survey, 2022).



Figura 11 - Gráfico contendo as respostas à pergunta "Quais os principais sistemas de controlo de versão que utiliza? Seleccione todos os que se apliquem.". Fonte (Stack Overflow - Survey, 2022).

Gitflow

“Giflow é um modelo alternativo para a gestão de *branches* Git que envolve o uso de *branches* para novos desenvolvimentos e múltiplas *branches* primárias”². Publicado e tornado popular por

¹ Tradução livre do autor. No original “Git is a free and open-source distributed version control system designed to handle everything from small to very large projects with speed and efficiency.” (Git, s.d.).

² Tradução livre do autor. No original “Giflow is an alternative Git branching model that involves the use of feature branches and multiple primary branches.” (Atlassian - Gitflow, s.d.).

Vincent Driessen no website nvie.com (nvie.com - A successful git branching model, 2010), este modelo tem como principal característica a utilização de múltiplas *branches* de longa duração, utilizadas para desenvolvimento de uma funcionalidade, e que só voltam a juntar-se à *branch* principal quando a funcionalidade é concluída.

Utilizando este modelo são criadas algumas *branches* primárias, com responsabilidades específicas. São estas *branches* chamadas de *Release*, *Development* e *Hotfix*.

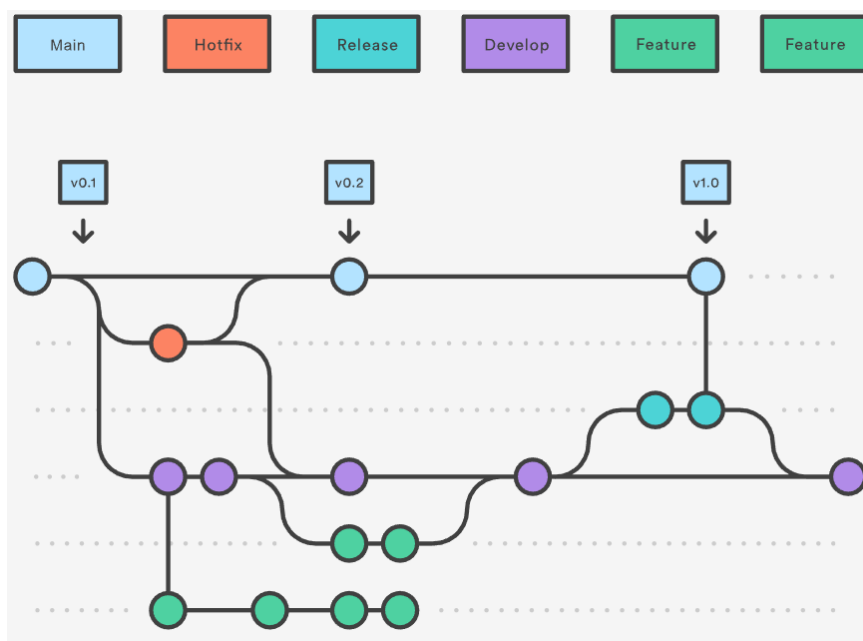


Figura 12 - Representação gráfica do modelo de trabalho Gitflow. Fonte (Atlassian - Gitflow, s.d.).

- **Development** – a principal *branch* de desenvolvimento. É a partir desta *branch* que se criam as *branches* de funcionalidade, e é também nesta *branch* que as novas funcionalidades são inicialmente integradas na aplicação. Esta *branch* é permanente.
- **Release** – esta *branch* criada a partir da *branch* de desenvolvimento, tem por objetivo isolar as funcionalidades que serão disponibilizadas no próximo ciclo. A partir do momento em que esta *branch* é criada, apenas documentação e correções devem ser permitidas até que esta se volta a juntar à *main*. Alterações feitas nesta *branch* devem também ser refletidas na *branch development*.

- **Hotfix** – usada para manutenção, esta *branch* é criada a partir da *main*, com objetivo de corrigir alguma funcionalidade com urgência. As alterações feitas em *hotfix* são fundidas diretamente com a *main*, e também com a *development*.

3.2 Plataformas LMS e distribuição de conteúdo digital

Nesta secção serão analisadas algumas das plataformas LMS mais utilizadas, explorando as suas capacidades de distribuição de conteúdo e os serviços que disponibilizam.

3.2.1 Escola Virtual

A Escola Virtual é a plataforma de ensino-aprendizagem do Grupo Porto Editora focada no apoio ao estudo de alunos do pré-escolar ao 12º ano de escolaridade. Lançada a 25 de janeiro de 2005 foi a primeira plataforma deste género em Portugal (Porto Editora a nossa história, s.d.).

A plataforma tem três tipos de conta, Aluno, Professor e Encarregado de Educação:

- **Aluno** – Esta conta pode ser grátis ou *premium*. A versão grátis é oferecida através de um acordo do Estado Português com a editora, e garante o acesso aos recursos complementares dos livros do Grupo Porto Editora, adotados pela escola (Escola Virtual - Subscrição Aluno e Pais, 2022). A conta *premium* de aluno, por outro lado, é adquirida individualmente e confere acesso integral ao conteúdo para o ano de escolaridade escolhido (Escola Virtual - Subscrição Premium Aluno, 2022).
- **Professor** – Grátis para professores que lecionam em escolas que tenham adotado livros do Grupo Porto Editora (Escola Virtual - Professor Adotante., 2022). Para os restantes professores é possível adquirir uma licença individual *premium* (Escola Virtual - Licença Individual Professor, 2022).
- **Encarregado de Educação (EE)** – Uma conta totalmente grátis com o objetivo de promover o envolvimento dos encarregados de educação no progresso escolar dos seus educandos através da disponibilização de relatórios de desempenho (Escola Virtual - Subscrição Aluno e Pais, 2022).

Esta plataforma de aprendizagem tem ainda soluções digitais completas para instituições (municípios / escolas / bibliotecas / centros de estudo), com o projeto EV 360º, que inclui acessos à plataforma Escola Virtual, formação, acesso a manuais digitais, fornecimento de equipamentos, entre outras vantagens (Escola Virtual - EV 360º, 2022).



Figura 13 - EV 360º - Solução à medida. Fonte (Escola Virtual - EV 360º, 2022).

O *front-end* da Escola Virtual foi desenvolvido em PHP utilizando a framework Phalcon, tendo sido criado como um conjunto de websites interligados. Quer isto dizer que, por exemplo, o LMS da plataforma existia no domínio www.lms-escolavirtual.pt, enquanto o módulo responsável pela apresentação de manuais digitais, a biblioteca da plataforma, se encontrava no domínio www.library-escolavirtual.pt. Em 2020 a plataforma uniu a maioria destes módulos dentro do mesmo projeto e domínio, www.app-escolavirtual.pt, adotando uma estrutura Phalcon *multi-module*. Esta estrutura permite manter uma separação de responsabilidades, continuando a existir um módulo LMS e biblioteca, por exemplo, enquanto simultaneamente centraliza o código, o processo de disponibilização e o domínio da plataforma. O módulo de distribuição e apresentação de conteúdo, Smart Player, não foi integrado nesta reestruturação, mantendo o seu domínio próprio, www.smartplayer.escolavirtual.pt.

Tabela 3 – Principais tipos de recursos disponibilizados pelo LMS da Escola Virtual.

Tipo de Recurso	Breve descrição
Vídeo	Conteúdo de vídeo Kaltura e mp4.
Vídeo Interativo	Conteúdo de vídeo Kaltura com interatividades.
Imagens	Conteúdo visual.
Áudio	Conteúdo áudio.
Aulas mAuthor (SCORM)	Conteúdo interativo criado com recurso à ferramenta mAuthor.
Jogos sérios HTML5	Jogos 2D com conteúdo educativo.
Jogos sérios WebGL	Jogos 2D criados no motor de jogo Unity, e disponibilizados em formato WebGL.
Testes SmartQuestions	Conteúdo avaliativo criado no SmartQuestions, um módulo da Escola Virtual.
Web Links	Links para conteúdo digital fora da plataforma.
Youtube Web Links	Links para conteúdo disponível no Youtube, apresentados na plataforma.
PDF	Documentos PDF.
Documentos Office	Documentos Word, PowerPoint e Excel, apresentado diretamente na plataforma.
Outros Documentos	Documentos em formato ZIP, disponíveis para download na plataforma.

3.2.2 Aula Digital

Desenvolvido pelo grupo LeYa, a Aula Digital é uma plataforma de ensino digital com conteúdos educativos do 1º ao 12º ano de escolaridade, e é o principal competidor da Escola Virtual no mercado de ensino digital português.

A Aula Digital tem dois tipos principais de conta, Aluno e Professor, tendo múltiplos tipos de licença com vários níveis de acesso à plataforma. Uma conta do tipo aluno completa tem acesso a manuais digitais (versão aluno) das editoras do grupo LeYa (ASA, Gailivro, TEXTO e Sebenta), assim como acesso a recursos e ferramentas de aprendizagem direcionados ao ano de escolaridade escolhido e avaliações interativas para preparação para testes e exames (Aula Digital - Oferta escolar para Alunos, s.d.). Uma conta de tipo professor completa inclui todos os

benefícios da conta aluno, tendo um pouco mais liberdade no acesso a livros digitais e acesso a ferramentas próprias de avaliação, aulas interativas para projeção em sala de aula e gestão de grupos/turmas virtuais onde um professor pode distribuir conteúdo de estudo, atribuir trabalhos e receber relatórios de desempenho dos seus alunos (Aula Digital - Oferta Professores, s.d.).

Ao entrar na Aula Digital, professores e alunos têm acesso às seguintes áreas:






 Biblioteca de manuais digitais adotados, com recursos digitais complementares e materiais exclusivos para o professor. Também disponível em versão offline.	 Banco de Recursos com recursos digitais das disciplinas do 1.º ao 12.º ano, organizados pelos temas dos programas, que podem ser partilhados com os alunos através da área As minhas salas. Ver vídeo
 Os meus testes para a edição/criação de testes de avaliação interativa, que podem ser enviados aos alunos através da área As minhas salas. Área exclusiva do professor	 As minhas salas para partilhar recursos e orientar o estudo dos alunos, através do envio de trabalhos e testes interativos. Ver vídeo
 Smart para aceder a vídeos e quizzes rápidos, com explicações imediatas, que ajudam a esclarecer dúvidas nas disciplinas em que o aluno estuda com manuais da LeYa Educação. Ver vídeo	*Quando o aluno não tem acesso a um computador, pode também fazer estes quizzes no seu telemóvel, descarregando o app Smart Aula Digital .

Figura 14 - Oferta da Aula Digital para professores (Aula Digital - Recursos, s.d.).

O banco de recursos desta plataforma inclui material em forma de animações, apresentações PowerPoint, simuladores, fichas para treino, exercícios com soluções, vídeos, infográficos interativos, testes interativos com correção automática e jogos didáticos (Aula Digital - Banco de Recursos, s.d.).

Smart - Aula Digital

A aplicação Smart tem como objetivo expandir as capacidades do serviço aula digital em smartphones, estando o acesso à mesma incluído na oferta do serviço para Alunos. Esta aplicação está disponível nas principais lojas de aplicações digitais, o Google Play, a App Store e a AppGallery (Aula Digital - Oferta escolar para Alunos, s.d.). No Google Play, loja digital com maior número de transferências e críticas da aplicação, a mesma é avaliada pelos utilizadores com 3.0 estrelas num possível máximo de 5 (Google Play - Smart - Aula digital, s.d.).

Aula Digital (App)

A aplicação Aula Digital, permite aos utilizadores aceder às principais funcionalidades do serviço num tablet e sem necessidade de internet. Disponibilizada também nas principais lojas digitais, a aplicação conta com uma avaliação de 3.2 estrelas, em 5, na Google play (Google Play - Aula Digital, s.d.).

Infinitas Learning

“A Infinitas Learning é uma empresa de recursos educativos dinâmicos e uma das principais fornecedoras da Europa de produtos e serviços inovadores e cativantes para ensino e aprendizagem”¹ (Infinitas Learning - About us, s.d.).

Em fevereiro de 2022, foi anunciado a venda do Grupo LeYa à empresa holandesa Infinitas Learning (Queirós, 2022). Numa notícia do jornal Observador, publicada a 15 março de 2022, é possível ler uma parte do comunicado oficial divulgado pelo grupo, dizendo que o mesmo irá manter a sua autonomia para o desenvolvimento de produtos e serviços para o sistema educativo português, focando ainda nos benefícios que esta operação irá trazer para o crescimento e partilha de conhecimentos entre os colaboradores (Concorrência notificada da venda da editora LeYa a grupo holandês, 2022).

3.2.3 Moodle

Modular Object-Oriented Dynamic Learning Environment, mais conhecido pelo seu acrónimo Moodle, é um LMS disponibilizado gratuitamente que permite a criação de websites com conteúdo dinâmico no contexto da educação.

Segundo estatísticas disponibilizadas pelo *website* oficial de estatísticas da plataforma (Moodle *Statistics*, s.d.), existem perto de 180 mil *websites* moodle em todo o mundo, servindo mais de 300 milhões de utilizadores. Em Portugal o moodle regista mais de 2 mil *websites*, sendo a sua adoção mais comum entre instituições de ensino superior, como as escolas do Politécnico do Porto (P.Porto).

¹ Tradução livre do autor. No original “Infinitas Learning is a dynamic educational resources company, and one of Europe's leading providers of innovative and engaging products and services for teaching and learning.” (Infinitas Learning - About us, s.d.).

Para a distribuição de conteúdo o moodle inclui *players* para a reprodução de alguns tipos de recurso, como PDFs e pacotes SCORM. Existem também *plugins* opcionais para suporte a outros tipos de conteúdo, como é exemplo o JW Player dedicado à exibição de vídeos (Moodle *Plugins – Media Player*, s.d.). Os administradores da plataforma podem ainda permitir que o conteúdo seja descarregado pelos utilizadores, possibilitando assim a distribuição de virtualmente todo o tipo de conteúdo comprometendo, no entanto, a exclusividade desse conteúdo à plataforma.

O moodle foi desenvolvido em PHP abraçando o princípio de “Código Aberto”, aceitando contributos de participantes de todo o mundo para a criação de novas funcionalidades e correção de problemas. Estes contribuidores são também responsáveis pelos múltiplos *plugins* que ajudam a fortalecer a oferta da plataforma.

Moodle App

A aplicação para dispositivos móveis Moodle App é um dos métodos de ligação ao conteúdo fornecido pelo Moodle. Um dos principais benefícios desta aplicação é a possibilidade de aceder a conteúdo offline, desde que previamente descarregado, dando maior flexibilidade aos seus utilizadores.

Esta app não está disponível em nenhuma loja digital, nem para todos os utilizadores das plataformas moodle, sendo um recurso que as organizações que gerem o LMS necessitam optar e subscrever, registando as suas páginas moodle no *website* Moodle Apps. Atualmente existem quatro planos de subscrição: grátis, profissional, *premium* e *branded*. Cada um destes planos oferece maior número de funcionalidades e vantagens, com um aumento progressivo do custo anual do serviço (Moodle App, s.d.).

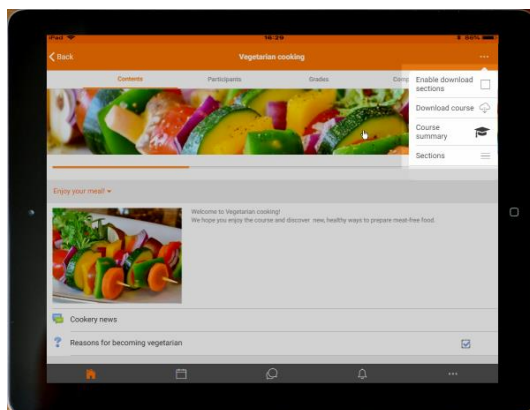


Figura 15 – Demonstração da possibilidade de descarregar o curso para uso offline na aplicação moodle. Fonte (Moodle App Features, s.d.).

3.2.4 Blackboard Learn

Blackboard Learn é um LMS desenvolvido e disponibilizado pela empresa Blackboard Inc, com o propósito de “ajudar a resolver os desafios mais críticos na educação e impulsionar os estudantes ao sucesso”¹ (Blackboard - Our purpose, s.d.).

Trata-se de uma plataforma comercial, não tendo uma versão completa gratuita, apenas um modo de demonstração com duração máxima de 30 dias. O preço da versão completa do programa começa nos 9500\$, cerca de 8400€, por ano (Capterra Blackboard Learn, s.d.). Em Portugal, este LMS foi adotado em algumas instituições de ensino superior, como é exemplo a Universidade do Minho (Universidade do Minho - Blackboard-UMinho, 2022).



Figura 16 - Ecrã de login da plataforma Blackboard da Universidade do Minho. Retirado de (Universidade do Minho - Blackboard-UMinho, 2022)

Em termos de distribuição de conteúdo o BlackBoard reconhece vários tipos de ficheiros, permitindo que estes sejam demonstrados diretamente no *website*, ou numa aplicação associada. Quando o sistema não reconhece o tipo de ficheiro, o mesmo pode ser descarregado pelos utilizadores e aberto com *software* apropriado instalado nos seus computadores (Blackboard - Supported File Types, s.d.).

¹ Tradução livre do autor. No original “We Help You Solve Your Most Critical Challenges in Education to Drive Student Success” (Blackboard - Our purpose, s.d.).

Tabela 4 - Tipos de ficheiros reconhecidos pela plataforma Blackboard Learn. Fonte (Blackboard - Supported File Types, s.d.)(adaptado para português)

Extensões	Tipo de Ficheiro	Programas associados
AAM	Multimédia	Macromedia® Authorware® <i>plug-in</i> .
AIFF	Áudio	Áudio não comprimido.
ASF	Multimédia	Microsoft® .NET™ Show
AU	Áudio	Real Audio Player™
AVI	Vídeo	Video player – Apenas em Windows
DOC, DOCX	Texto	Microsoft® Word
EXE	Executável	Executáveis podem ser descarregados.
GIF	Imagem	Programa gráfico ou diretamente no browser
HTML, HTM	Página Web	Editor HTML ou browser
JPG, JPEG	Imagem	Programa gráfico ou diretamente no browser
JIF	Imagem	Programa gráfico ou diretamente no browser
MP3	Áudio	Programa de Áudio
MP4	Vídeo	<i>Player</i> de vídeo
MPE	Áudio/Vídeo	Programa de Áudio
MPG, MPEG	Vídeo	<i>Player</i> de vídeo
MOOV, MOVIE	Filme	QuickTime®
MOV	Vídeo	<i>Player</i> de vídeo ou filme
NUMBERS	<i>Spreadsheet</i>	Apple Numbers®
PDF	Texto	Adobe® Acrobat® Reader®
PNG	Imagem	Programa gráfico ou diretamente no browser
PPT, PPTX, PPS	Apresentação de slides	Microsoft® PowerPoint®, PowerPoint Player®
QT	Filme	QuickTime®
RA	Áudio	Real Audio Player™
RAM	Vídeo	Real Audio Movie™
RM	Áudio	Programa de Áudio
RTF	Texto	Word processor
SWF	Multimédia	Macromedia® Shockwave® <i>plug-in</i>
TIFF, TIF	Imagem	Programa gráfico ou diretamente no browser
TXT	Texto	Editor de Texto ou HTML
WAV	Áudio	Programa de Áudio
WMA	Áudio	Programa de Áudio

WMF	Gráfico	Microsoft® Windows®
XLS, XLSX	<i>Spreadsheet</i>	Microsoft® Excel®
ZIP	Pacote Comprimido	WinZip®

3.3 Media Players

Nesta secção são analisados vários *media players* focando o tipo de ficheiro multimédia que suportam, as funcionalidades que disponibilizam e é feita uma análise da sua compatibilidade com os requisitos identificados para este projeto.

3.3.1 *Player* de vídeo

Um dos tipos de multimédia mais utilizado nos dias de hoje, vídeos são um tipo de conteúdo indispensável a uma plataforma de aprendizagem. No contexto deste projeto serão utilizados múltiplos *media players* para este tipo de conteúdo.

Elemento HTML <video>

A solução para a apresentação de vídeos disponível em todos os navegadores mais utilizados, a *tag* HTML video é a forma mais simples e direta de adicionar conteúdo vídeo a um *website* ou aplicação *front-end*. Este elemento suporta os formatos de vídeo MP4, WebM e Ogg (HTML Video, s.d.)

Kaltura *Player*

“Kaltura é a primeira plataforma de vídeo *online* de código aberto, fornecendo *software* e serviços comerciais de nível empresarial, totalmente suportados e mantidos pela Kaltura, bem como soluções gratuitas de código aberto suportadas pela comunidade, para publicação, gestão, distribuição e monetização de vídeos”¹ (Kaltura Player ToolKit, s.d.).

¹ Tradução livre do autor. No original “Kaltura is the world's first Open-Source Online Video Platform, providing both enterprise level commercial software and services, fully supported and maintained by Kaltura, as well as free open-source community supported solutions, for video publishing, management, syndication and monetization” (Kaltura Player ToolKit, s.d.).

Este *player* de vídeo oferece suporte multiplataforma, várias opções de customização, tanto de aparência como de funcionalidades, e ainda plugins para expansão das funcionalidades base da plataforma.



Figura 17 – Opções de customização do Kaltura Player. Fonte (Kaltura Player, s.d.)

YouTube IFrame Player API

A IFrame Player API do YouTube é uma ferramenta que permite embutir um vídeo YouTube num *website*, controlando o *player* usando JavaScript. As funções disponibilizadas pela API permitem criar listas de reprodução, ajuste de volume, iniciar, parar ou colocar em pausa e recolher informações sobre o vídeo tocado (Youtube - IFrame Player API, s.d.).

Sendo o YouTube a plataforma de vídeos online mais popular, a inclusão de um *player* dedicado à reprodução do conteúdo da plataforma é imprescindível, uma vez que confere a possibilidade de usufruir do vasto catálogo de vídeos disponível.

3.3.2 icPlayer

O *Interactive Content Player*, ou icPlayer, é um *media player* de “código aberto” baseado em HTML-5. Este *media player* é ativamente recomendado pela empresa Learnetic S.A. para a exibição de conteúdo interativo da sua ferramenta de autoria, o mAuthor, sendo essa a principal função do mesmo. O *player* tem capacidade de exibir o conteúdo mAuthor mesmo quando este não é exportado seguindo o padrão SCORM (IcPlayer - mAuthor Open Source Player, s.d.).

Não existe muita informação pública sobre este *player*, no entanto a sua relação com o mAuthor, utilizado para a criação de conteúdo educativo para a Escola Virtual, torna-o relevante para atingir os requisitos propostos pela Porto Editora.

3.3.3 SQ Player

Smart Questions Player, é um *player* criado para a reprodução dos recursos criados no módulo *Smart Questions* da Escola Virtual. Este *player* foi criado em JavaScript e é responsável pela gestão das regras funcionais de um teste, sendo a única forma de apresentar este tipo de conteúdo.

Este *player* está integrado no módulo Smart Questions (SQ) da plataforma Escola Virtual.

3.3.4 PDF Viewer

Acrónimo de *Portable Document Format*, o PDF é um formato de arquivo criado pela Adobe para a apresentação de documentos de texto, em múltiplas plataformas e sistemas operativos.

PDF *Viewers* ou PDF *Readers*, é o nome genérico dado a *software* com objetivo de apresentação de ficheiros de formato PDF. Podemos por isso considerar estes *softwares* como um tipo de *media player*. Os navegadores mais populares como o Chrome, Opera ou Firefox, já incluem suporte para leitura de ficheiros deste tipo, no entanto requerem a abertura do documento numa nova aba ou página e utilizam plugins, muitas vezes pré incluídos no *software*, para desempenhar esta função.

No contexto deste projeto é necessário que o ficheiro PDF seja apresentado diretamente na página HTML online ou offline, garantindo controlo da qualidade de apresentação do conteúdo. Não é requisito a inclusão de funcionalidades extra, como a capacidade de fazer download do ficheiro.

PDFObject

“Um utilitário JavaScript compatível com padrões de código aberto para incorporar arquivos PDF em documentos HTML”¹ (PDFObject, s.d.). Este utilitário facilita a incorporação dos arquivos em páginas HTML, fornecendo tratamento de erros e oferecendo opções alternativas quando o browser do utilizador não suporta esta funcionalidade.

¹ Tradução livre do autor. No original “An open-source standards-friendly JavaScript utility for embedding PDF files into HTML documents.” (PDFObject, s.d.).

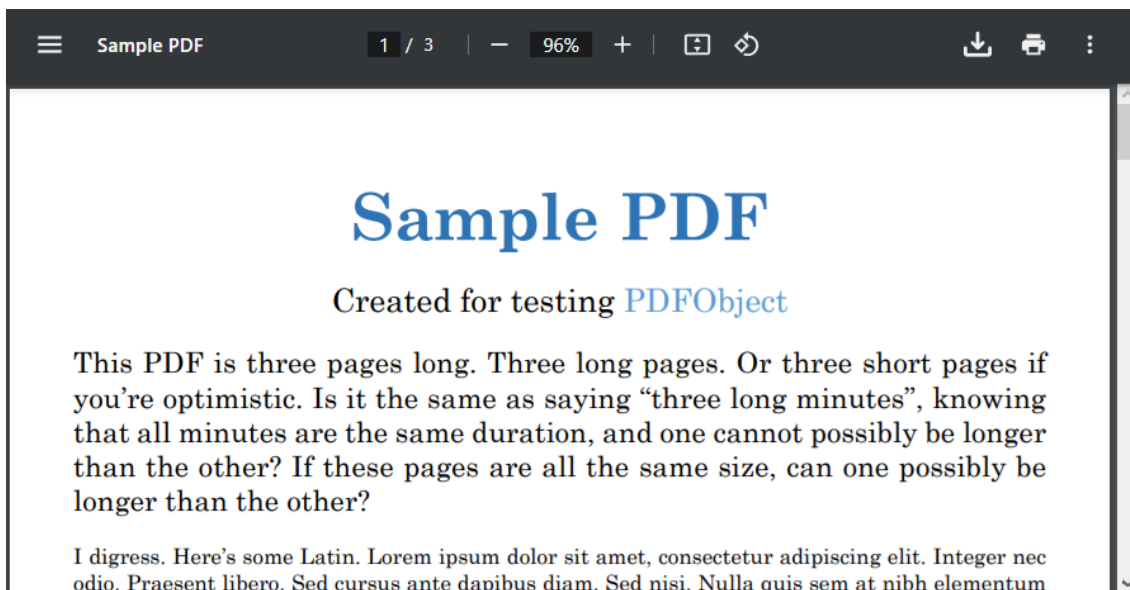


Figura 18 – Exemplo do utilitário PDFObject no navegador Opera. Fonte (PDFObject, s.d.)

É um *player* simples de integrar, mas não resolve a totalidade do problema, uma vez que depende do PDF *viewer* do navegador, não garantindo o controlo da apresentação e das funcionalidades incluídas, assim como dificultar a expansão das mesmas.

Adobe PDF Embed API

Adobe, empresa criadora do formato PDF, disponibiliza o Adobe PDF Embed API como solução gratuita para a inclusão de PDFs em páginas HTML. Esta solução garante a correta apresentação do conteúdo em qualquer navegador, assim como opções de customização e facilidade de integração com o serviço Adobe Analytics, não sendo claro no *website* se esta integração resultaria em custos monetários (Adobe PDF Embed API, s.d.).

Como demonstrado na figura abaixo, é possível customizar esta API na demonstração que se encontra no *website* da plataforma de forma intuitiva e, posteriormente, gerar o código de configuração para facilmente incluir no projeto (Adobe PDF Embed API Quickstarts, s.d.).

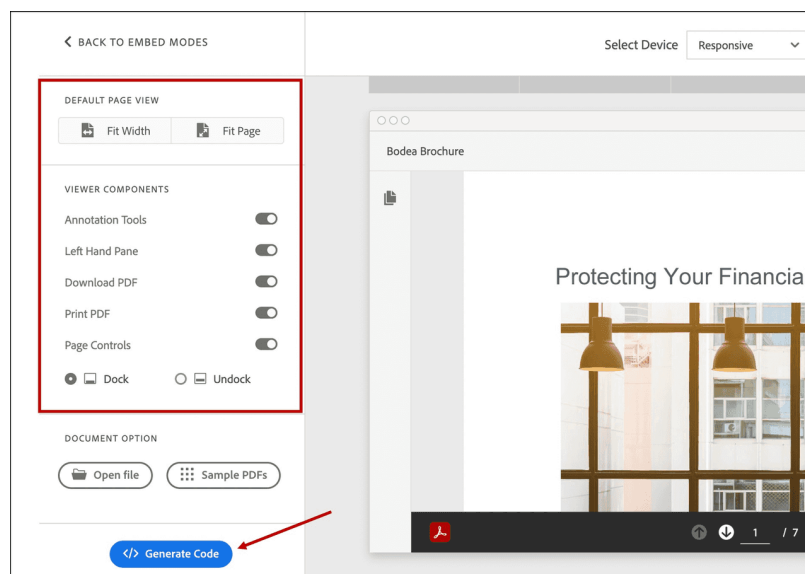


Figura 19 - Configuração dinâmica do Adobe PDF Embed API, na demonstração disponível no website da API. Fonte (Adobe PDF Embed API Quickstarts, s.d.)

Apesar de todas as capacidades e vantagens apresentadas, existem múltiplas limitações na apresentação e nas funcionalidades disponibilizadas por esta API. Em computadores, por exemplo, o navegador Firefox não permite a impressão do ficheiro diretamente a partir do leitor. Em dispositivos móveis esta limitação estende-se a todos os navegadores. A função de anotação é limitada em telemóveis, estando, no entanto, disponível em tablets. Quanto à função para descarregar o PDF, não é suportada em dispositivos iOS. Para além disto, esta solução requer uma conexão à internet para funcionar (Adobe PDF Embed API Quickstarts, s.d.).

Em geral o Adobe PDF Embed API é uma boa solução disponível para a incorporação de ficheiros PDF em *websites*. No entanto as suas limitações, incapacidade de funcionamento *offline* e incoerências de apresentação em diferentes dispositivos torna-a insuficiente.

React PDF

React PDF é um componente para a apresentação de documentos PDF em aplicações React. Trata-se de um componente simples que não contém nenhuma funcionalidade extra incluída, fornecendo, no entanto, uma série de funções para que essas funcionalidades possam ser implementadas (React PDF, 2022).

React PDF Viewer

O React PDF Viewer é outro componente para a apresentação de documentos PDF em aplicações React (React PDF Viewer, s.d.). A principal vantagem desta solução em relação ao React PDF apresentado anteriormente é a biblioteca de *plugins* opcionais que permitem a adição de múltiplas funcionalidades ao leitor de PDF, como zoom, marcador de destaques, anotações ou a escolha de temas (React PDF Viewer - Plugins, s.d.). No entanto, esta solução tem um custo associado de 199\$ (cerca de 185€) (React PDF Viewer - Purchase, s.d.).

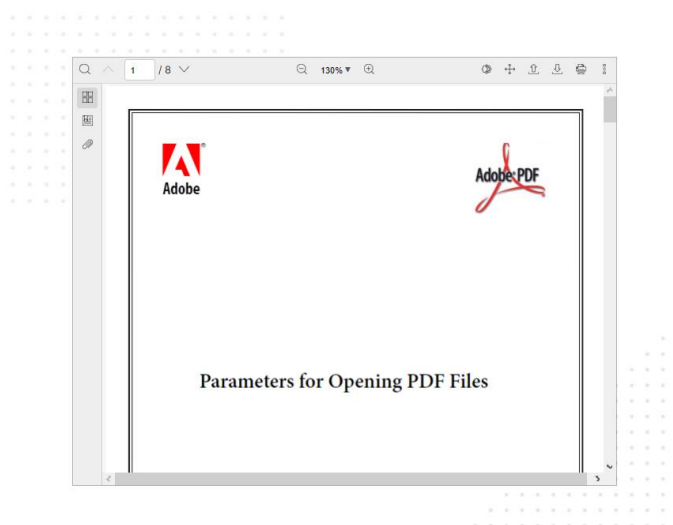


Figura 20 - Demonstração do React PDF Viewer. Fonte (React PDF Viewer, s.d.).

3.3.5 Office Viewer

Quando se refere o termo “documentos *Office*”, refere-se a documentos cujo formato é associado à família de *software* Microsoft Office. Este termo inclui documentos Word (geralmente com extensão .doc e .docx), documentos PowerPoint (.ppt, .pptx ou .pps) e documentos Excel (.xls e .xlsx). A um *Software* utilizado para a visualização de documentos Office é dado o nome de *Office Viewer*.

Para o presente projeto, o *office viewer* escolhido deve apresentar o conteúdo corretamente em diferentes resoluções, não permitindo que o mesmo seja alterado dentro da plataforma, mas dando a possibilidade de o utilizador descarregar uma copia para edição.

Microsoft Office Viewer

Utilizando a funcionalidade de pré-visualização do Microsoft Office é possível utilizar o Microsoft Office Viewer para incluir documentos em páginas HTML. Este visualizador é focado na visualização dos três tipos de documentos referidos acima, oferecendo opções de impressão e conversão para PDF e até funcionalidades de tradução e de acessibilidade como a leitura avançada, onde um ficheiro de texto pode ser lido para o utilizador.

A principal limitação deste visualizador é o limite de tamanho do ficheiro que permite pré-visualizar, que não deve ultrapassar os 10MB.

Google Docs Viewer

Criado pela Google, este visualizador faz parte das funcionalidades do Google Docs, um editor de documentos online. Tal como acontece com o Microsoft Office Viewer, o Google Docs Viewer permite a abertura de documentos diretamente na página HTML através da funcionalidade de pré-visualização.

Este *player* não se limita à apresentação de documentos Office, podendo ser utilizado para a pré-visualização de ficheiros PDF, ficheiros de texto básico (.txt) ou até mesmo vídeos em múltiplos formatos (WebM, .MPEG4, .MOV, entre outros).

Esta funcionalidade apresenta também uma limitação em termos do tamanho de ficheiro a visualizar, não podendo exceder os 25MB, e não disponibiliza nenhuma funcionalidade extra na apresentação dos documentos.

React-doc-viewer

Um componente React criado com o objetivo de apresentar vários tipos de documentos, incluído todos os ficheiros Office. Este componente utiliza o Microsoft Office viewer apresentado anteriormente, mantendo todas as suas capacidades e limitações (react-doc-viewer, 2020).

Este projeto aparenta ter sido abandonado, não tendo sofrido atualizações por mais de dois anos, no entanto uma *fork* do projeto, @cyntler/react-doc-viewer, continuou o seu desenvolvimento e atualização, sendo a versão recomendada deste componente (@cyntler/react-doc-viewer, 2022).

3.3.6 *Player* de áudio

Áudio é um dos elementos de mídia mais utilizados, existindo inúmeros *players* e plataformas comerciais inteiras dedicadas à apresentação deste tipo de conteúdo. Para o componente de distribuição desenvolvido ao longo deste projeto são apenas necessárias as funcionalidades básicas de reprodução deste tipo de recurso: controlo de reprodução (iniciar/parar), controlo de áudio e barra de reprodução interativa.

Elemento HTML <audio>

A solução mais simples e direta de implementar um *player* de áudio numa página ou aplicação web é a utilização da *tag* HTML5 audio. Esta opção responde a todos os requisitos listados acima, é suportada por todos os principais navegadores e suporta três formatos de ficheiro áudio: MP3, WAV e OGG (HTML Audio, s.d.).

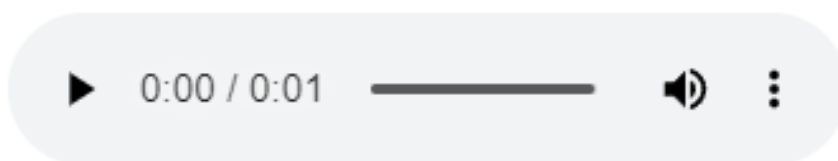


Figura 21 - Demonstração do elemento HTML audio. Fonte: (HTML Audio, s.d.)

React Audio Player

O React Audio Player é um componente React que “... envolve a *tag* HTML5 audio. (o componente) providencia a habilidade de manipular o *player* e subscrever eventos através de uma agradável interface React” ¹ (React Audio Player, 2021). Este componente oferece vantagens de performance e compatibilidade com a biblioteca React em comparação com a utilização direta do elemento HTML5 audio.

¹ Tradução livre do autor. No original “*This is a light React wrapper around the HTML5 audio tag. It provides the ability to manipulate the player and listen to events through a nice React interface.*” (React Audio Player, 2021).

React H5 Audio Player

Este *player* de áudio para React, é também baseado na *tag* HTML5 audio, acrescentando algumas funcionalidades, como a possibilidade de reproduzir o áudio em ciclo e reproduzir lista de áudios. Trata-se de um *player* mais pesado e que excede os requisitos deste projeto (React H5 Audio Player, 2022).

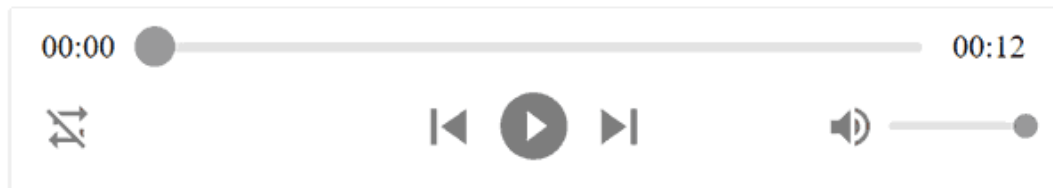


Figura 22 - Demonstração do componente React H5 Audio Player. Fonte (React H5 Audio Player, 2022).

4 Desenho da Solução

Neste capítulo é apresentada a solução proposta para a criação de um módulo de distribuição de recursos educativos, focando no caso em estudo. O capítulo será iniciado por uma análise aos problemas encontrados pela Porto Editora no seu módulo de distribuição de conteúdo, seguido de uma lista das tecnologias utilizadas para a criação do novo módulo. Segue uma breve explicação sobre o funcionamento da *framework* de desenvolvimento escolhida e, para finalizar, é apresentado o desenho da aplicação e da interface gráfica.

4.1 Análise do módulo Smart Player da Escola Virtual

Nesta secção será analisado o módulo de distribuição da Escola Virtual, o Smart Player, explorando em maior detalhe as causas das limitações listadas no capítulo 1.2 deste documento, os requisitos esperados pela Porto Editora listados no capítulo 1.3 e as soluções propostas para a nova aplicação.

4.1.1 Incoerência da qualidade da exibição de recursos

O principal obstáculo para atingir coerência e qualidade na exibição dos recursos é a utilização de iframes ou webviews, com diferentes domínios. Isto acontece porque o módulo de distribuição está publicado num domínio diferente do LMS, como explicado no capítulo 3.2.1. Existem também alguns tipos de recursos que são apresentados recorrendo a iframes, criando estruturas com três níveis de iframes sem domínio partilhado.

Quando uma iframe não partilha domínio com a página que a contém, o acesso a informação do seu DOM é impedido pelo mecanismo de proteção web CORS (*Cross-Origin Resource Sharing*), tornando difícil o processo de ajuste de tamanho dos recursos ou o cálculo de escalas.



Figura 23 - Exemplo de uma estrutura com três iframes para apresentação de recursos Smart Questions na Escola Virtual.

Esta arquitetura é consequência de anos de iterações e desenvolvimentos na plataforma Escola Virtual, sem acompanhamento suficiente de desenvolvimento e atualização do módulo de distribuição de conteúdo.

Para a resolução deste problema é primeiro importante incluir o módulo de distribuição diretamente no LMS, removendo assim a necessidade de utilizar iframes ou webviews. Em segundo lugar é necessário garantir que todos os conteúdos que, por requisito dos seus *players*, necessitem ser apresentados dentro de uma iframe, tenham já garantido um comportamento responsivo ou que sejam disponibilizados dentro do mesmo domínio que o LMS, permitindo assim acesso direto ao seu DOM para recolha de dados de visualização (largura e altura).

4.1.2 Limitações de recolha de dados na plataforma

Como referido no capítulo 1.2 e 1.3 a Escola Virtual depende de serviços externos para a recolha de dados analíticos. Para além dos custos associados, esta dependência traz algumas limitações quanto ao tipo de dados recolhidos e à sua utilidade.

De forma a reduzir os custos associados, assim como permitir maior liberdade de captura e utilização dos dados recolhidos, é necessário criar uma estrutura no módulo de distribuição capaz

de recolher os dados requeridos, e em simultâneo criar uma estrutura de back-end para os guardar.

4.1.3 Inflexibilidade da plataforma

Parte deste problema tem origem na estrutura descrita anteriormente na secção 4.1.1, mas, tal como referido no capítulo 1.3, o maior desafio à expansão, atualização e desenvolvimento deste módulo é causado pela dependência de uma tecnologia severamente desatualizada, a *framework* Angular versão 1.2.28, lançada a 15 de Dezembro de 2014 (angular.js v1.2.28, 2014).

Tabela 5 - Versões ativas da *framework* Angular. Adaptado de (Angular versioning and releases, s.d.).

Versão	Estado	Lançamento
^14.0.0	Ativo	2022-06-02
^13.0.0	<i>Long-term support</i>	2021-11-04
^12.0.0	<i>Long-term support</i>	2021-05-12

A tabela acima ajuda a compreender o quão desatualizada é a versão utilizada pelo módulo de distribuição da Escola Virtual, assim como a dificuldade em proceder com a sua atualização dado o número total de versões principais (*major versions*) em atraso.

A resolução deste problema passa pela construção do novo módulo com recurso a tecnologias mais modernas, assim como o planeamento de atualizações de versão frequentes.

4.2 Tecnologias utilizadas

Nesta secção são exploradas as principais tecnologias escolhidas para o desenvolvimento do projeto, assim como uma breve explicação do motivo para a sua escolha.

4.2.1 Linguagem de programação e *framework*

Para desenvolver o projeto, foi escolhida a linguagem de programação TypeScript, pela sua utilidade em manter a coesão em projetos de grande escala/complexidade, e a biblioteca React, versão 18.0.0, foi escolhida e utilizada como principal *framework* do projeto.

Como explicado brevemente no capítulo 3.1.9, a biblioteca React permite a criação de aplicações que podem facilmente ser integradas num website, cumprindo assim um dos requisitos identificados anteriormente: a inclusão do módulo de distribuição diretamente no LMS. A versão escolhida, 18.1.0, é a versão mais recente da biblioteca à data da escrita do presente documento (React Versions, 2022). É também importante lembrar que a principal função deste projeto é a distribuição de recursos, e a biblioteca React tem à sua disposição uma grande variedade de *media players* customizáveis para múltiplos tipos de conteúdo, como demonstrado no capítulo 3.3 deste documento.

Para fins de estilização da aplicação foi usada a linguagem CSS (*Cascading Style Sheets*).

4.2.2 Controlo de versões e disponibilização

Para gestão e distribuição do projeto foi escolhido o sistema de controlo de versões Git (referido no capítulo 3.1.10), adotando Gitflow como estratégia para controlo do fluxo de trabalho. De forma a facilitar a interação com o repositório, foi usada a interface gráfica SourceTree.

Para o processo de compilação, disponibilização e integração da aplicação com o LMS da Escola Virtual foi utilizada uma Pipeline Jenkins.

4.2.3 Ambiente de desenvolvimento integrado

O IDE escolhido para o desenvolvimento foi o Visual Studio Code (VS Code), juntamente com algumas extensões de integração com React, TypeScript e Git, como o compilador de TypeScript, o GitLens, para mais informações sobre o repositório (Visual Studio Marketplace - GitLens, 2022), e o Git Graph para melhor visão geral e gestão das *branches* do projeto (Visual Studio Marketplace - Git Graph, 2021).

4.3 Conceitos principais da biblioteca React

Para total compreensão da solução proposta é necessário entender o funcionamento de uma aplicação React. Nesta secção será explorado o conceito de componentes, JSX, estados React, assim como a estrutura de uma aplicação, o funcionamento do ciclo de um componente e os padrões de desenvolvimento adotados.

4.3.1 Componentes e JSX

“Conceitualmente, os componentes são idênticos a funções JavaScript. Aceitam entradas arbitrárias (propriedades) e retornam elementos React descrevendo o que deve aparecer no ecrã”¹ (React Docs - Components and props, s.d.). Um componente React devolve um elemento JSX, uma extensão da sintaxe JavaScript que representa *tags* HTML (React Docs - Introducing JSX, s.d.).

4.3.2 Estrutura de uma aplicação React

Uma aplicação React pode ser facilmente representada num diagrama de componentes, iniciada pelo ficheiro App, que contém o elemento *root* DOM node (nó raiz da aplicação React). Todos os elementos apresentados dentro do DOM virtual desta raiz são controlados pela biblioteca, sendo estes elementos, objetos simples que requerem poucos recursos (memória e processamento) para manter e alterar (React Docs - Rendering elements, s.d.).

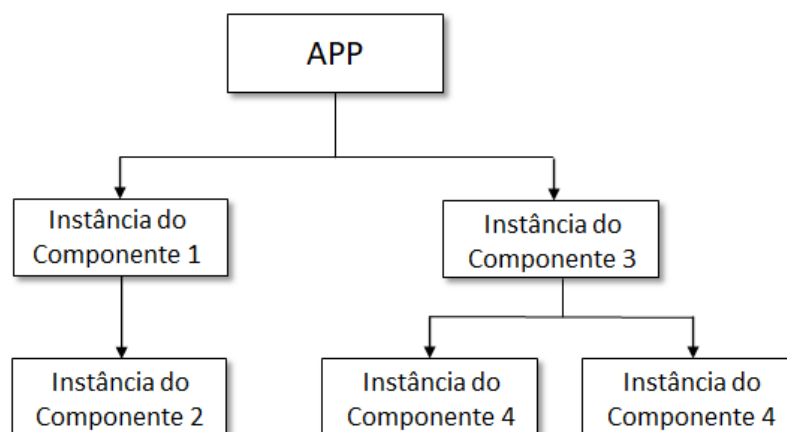


Figura 24 – Diagrama de componentes representativo da estrutura de uma aplicação React.

Uma das principais regras na estruturação de uma aplicação utilizando a biblioteca React diz respeito à propagação de dados. Esta propagação dá-se apenas na vertical, de cima para baixo, ou seja, da raiz (pai) para os ramos/folhas (filhos). Interação vertical invertida (de baixo para cima)

¹ Tradução livre do autor. No original “Conceptually, components are like JavaScript functions. They accept arbitrary inputs (called “props”) and return React elements describing what should appear on the screen.” (React Docs - Components and props, s.d.).

é possível, propagando funções do elemento pai para o filho, permitindo assim ao elemento filho chamar funções que despoletem alterações nos estados do elemento pai.

Por outro lado, a interação horizontal direta entre componentes não é possível, sendo necessário propagar funções e propriedades de um componente pai comum para conseguir essa interação (React Docs - Thinking in React, s.d.). Existem, no entanto, ferramentas de gestão de estados como o Redux, que criam um armazém virtual (*store*), capaz de ser acedido e modificado por qualquer componente da aplicação e despoletar atualizações. Esta ferramenta será explorada em detalhe mais à frente neste capítulo.

4.3.3 Estados e ciclo de vida de um componente

Estados são uma das características mais interessantes em componentes React. Esta propriedade está intimamente relacionada com o ciclo de vida de um componente. Um estado armazena informação, existindo funções capazes de detetar quando estes estados foram atualizados, despoletando uma atualização do conteúdo apresentado pela aplicação (React Docs - State and Lifecycle, s.d.).

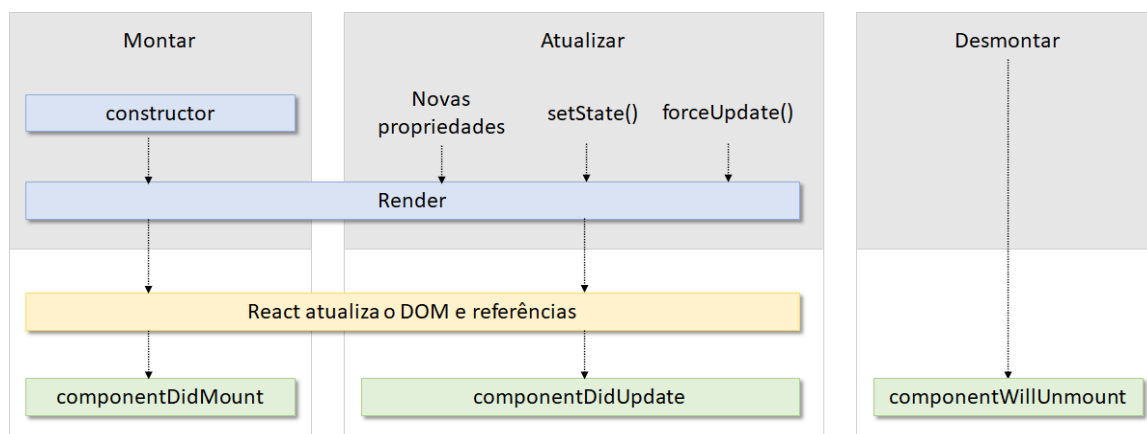


Figura 25 - Ciclo de vida de um componente React. Adaptado da fonte (React lifecycle methods diagram, s.d.)

Como demonstrado na figura acima, existem três momentos principais na vida de uma instância de um componente: montar, atualizar e desmontar.

- **Montar** – momento em que o componente é instanciado. O componente começa por executar o código presente no seu construtor, e em seguida executa a função *render*. O

conteúdo desenhado é então apresentado no ecrã, atualizando o DOM (*Document Object Model*). Por último a função “componentDidMount” é despoletada, oferecendo a oportunidade de iniciar eventos nos novos elementos HTML do DOM virtual.

- **Atualizar** – a atualização da instância de um componente ocorre sempre que um dos seus estados, ou uma propriedade passada pelos elementos acima na hierarquia é alterada. Após atualizar o conteúdo apresentado, a função “componentDidUpdate” é executada, permitindo introduzir código que reaja às alterações. A função “componentDidUpdate” pode executar alterações nos estados do componente, despoletando uma nova atualização.
- **Desmontar** – executado quando a instância de um componente é eliminada. A função componentWillUnmount é chamada antes da instância ser apagada, permitindo terminar funções ou eventos, e assim evitar potenciais erros ou poluição do DOM.

4.3.4 Componentes Class e Componentes Function

Existe uma discussão na comunidade React sobre a utilização de componentes Class e/ou componentes Function. Classes são componentes com o ciclo de vida React completo, como demonstrado na secção anterior, enquanto componentes Function são funções simples de JavaScript/TypeScript.

A discussão entre estes dois tipos de componentes iniciou-se com a versão 16.8 do React, e a introdução de Hooks na biblioteca (React Docs - Introducing Hooks, s.d.). Esta nova funcionalidade criou a possibilidade de utilizar funções de um ciclo React em funções JavaScript (ou TypeScript) comuns, transformando-as assim em componentes Function. Para além da consequente alteração da sintaxe, Hooks permitem ao programador escolher que elementos do ciclo de vida React pretende utilizar (React Docs - Hooks at a Glance, s.d.).

Devido à maior complexidade base de um componente Class, existe um ganho teórico de desempenho ao utilizar componentes Function mais simples. No entanto o impacto é difícil de medir e dependendo do tamanho da aplicação totalmente desprezível.

Para o projeto apresentado nesta dissertação, foi adotado a utilização de componentes Class.

4.3.5 Padrão *Container/Presentational*

Para melhor estruturar os componentes Class, foi decidida a utilização do padrão *Container/Presentational*. Este padrão impõe o princípio da separação de conceitos, separando os componentes Class em 2 partes:

- ***Container***: Responsável pela aquisição e tratamento de dados, assim como escolha dos dados a apresentar ao utilizador.
- ***Presentational***: Responsável por mostrar, e decidir como mostrar, a informação ao utilizador.

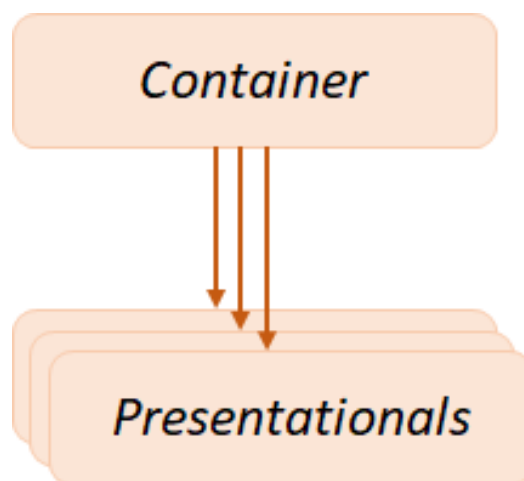


Figura 26 -Demonstração do padrão *container/presentational*.

Componentes criados seguindo este padrão podem ter múltiplos *presentational*, mas deverá instanciar apenas um de cada vez, cabendo ao *container* decidir qual *presentational* utilizar. (Patterns - Container/Presentational Pattern, s.d.).

4.3.6 Redux

Redux é uma ferramenta utilizada para centralizar estados de uma aplicação React. Qualquer componente dentro do DOM virtual React tem assim a possibilidade de subscrever a estes estados, despoletando atualizações sempre que o estado for alterado (Redux, s.d.).

Utilizando esta ferramenta é possível contornar as regras de propagação de dados estabelecidas pelo React (descritas na secção 4.3.2), permitindo assim comunicação horizontal direta e sem necessidade de despoletar atualizações em vários componentes colaterais.

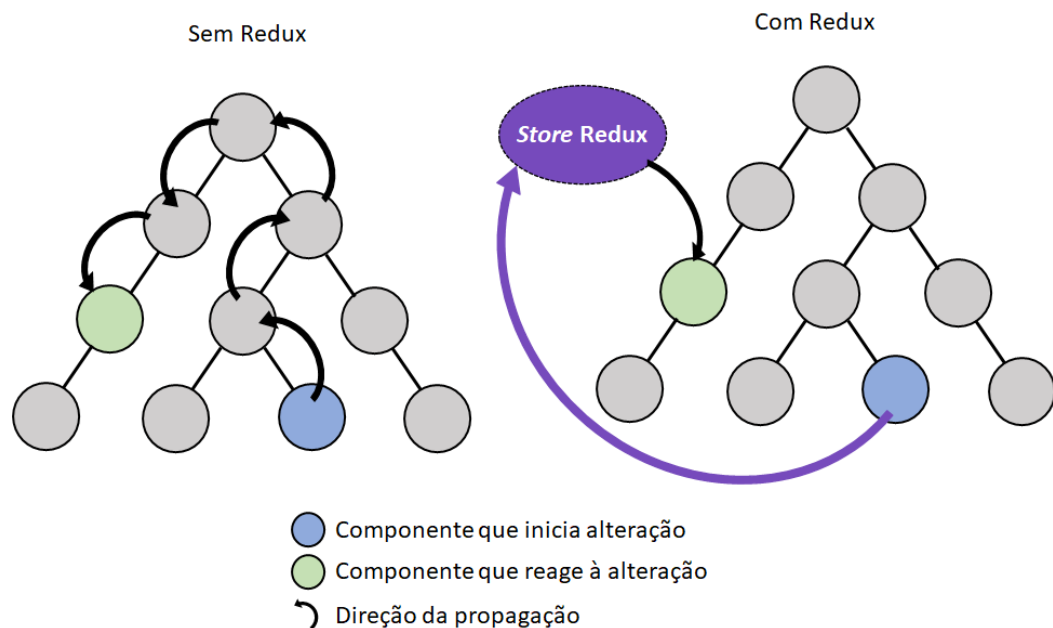


Figura 27 – Demonstração da propagação de atualizações com e sem Redux.
Adaptado de (Lima, 2020).

O Redux é composto por 3 elementos: a *store*, as *actions* e os *reducers*.

- **Store** – O armazém único de estados do Redux. Qualquer componente pode aceder a um estado contido na *store*, procedendo com a subscrição ao mesmo. Para modificar estes estados é necessário utilizar uma *action*.
- **Actions** – *Actions* permitem a alteração dos estados dentro da *store* Redux, servindo de intermediário entre os componentes da aplicação e os *reducers*.
- **Reducers** – Um *reducer* é despoletado por uma *action* e é responsável por atualizar um estado da *store*, despoletando um novo ciclo React em todos os componentes que o subscrevem.

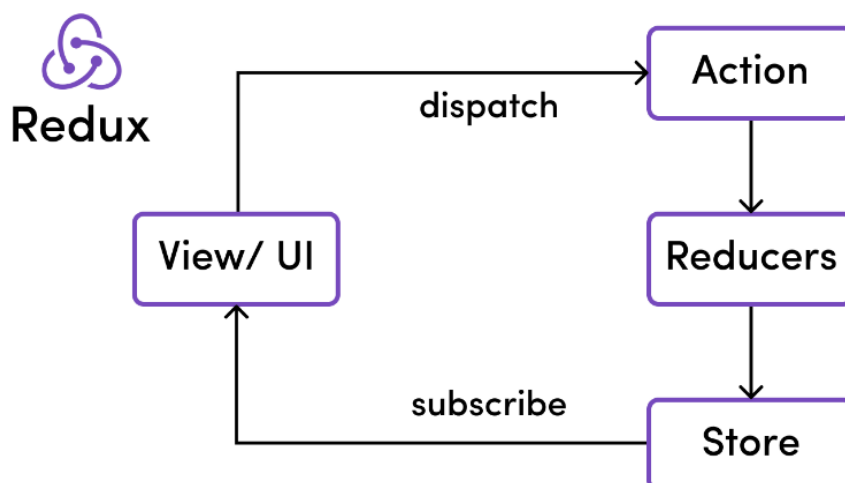


Figura 28 - Princípios base do Redux. Fonte (Roy, 2022).

A principal desvantagem da utilização desta ferramenta de gestão de estados é o aumento de complexidade associada à necessidade de utilizar estes elementos para a atualização e subscrição de estados, requerendo vários passos para implementar qualquer atualização de estados, mesmo que relativamente simples.

4.4 Desenho da aplicação

Nesta secção é apresentado o desenho da aplicação React para distribuição de conteúdo educativo projetada para o caso em estudo nesta dissertação. Para iniciar será apresentado um diagrama de componentes, descritivo da estrutura de toda a aplicação, seguido de uma análise mais detalhada de cada componente proposto nesta solução, assim como as suas responsabilidades e relações.

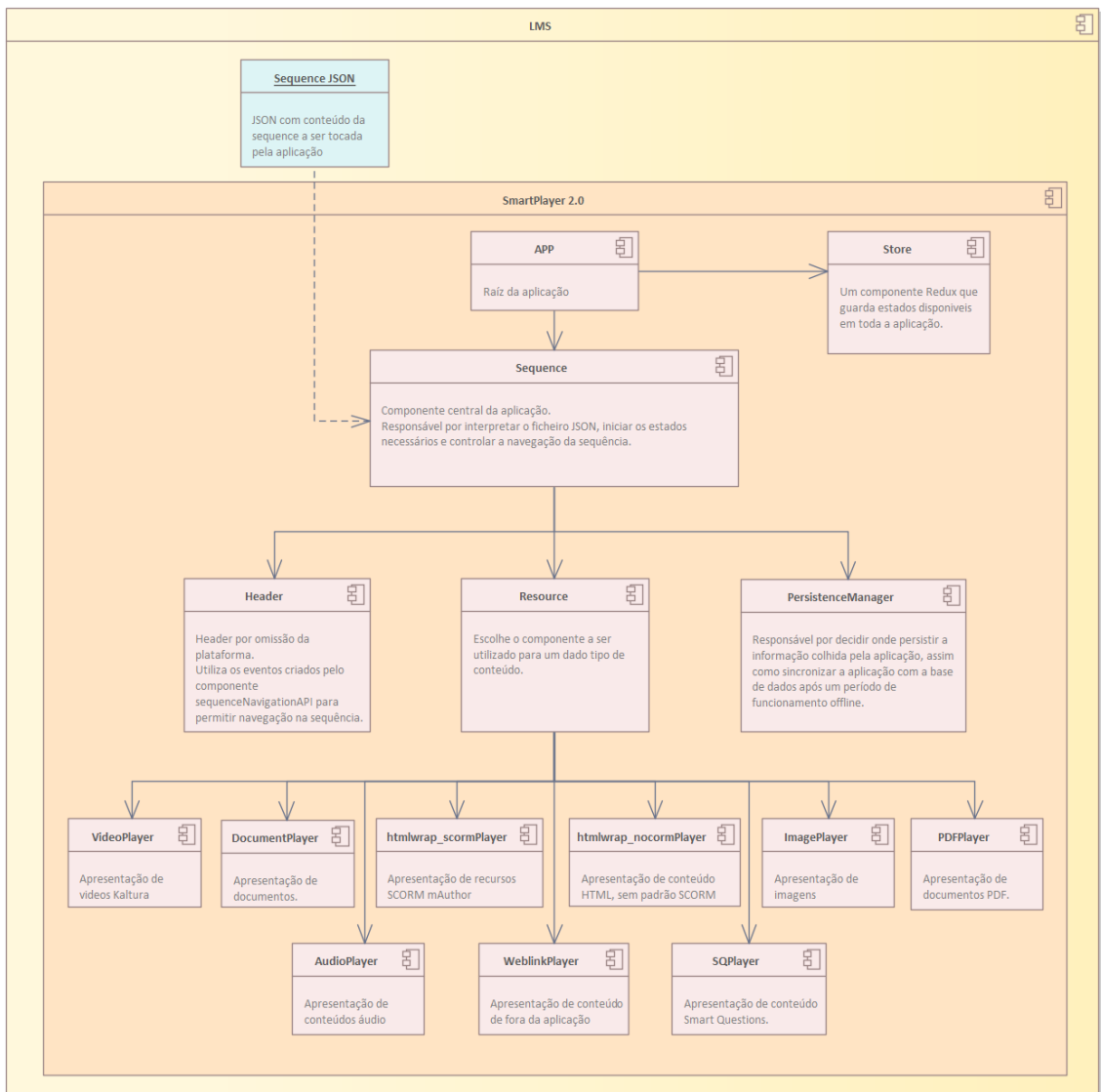


Figura 29 - Diagrama de componentes da aplicação.

4.4.1 JSON da Sequência

A aplicação foi desenhada com o objetivo de ser integrada numa plataforma LMS. Desta forma, é esperado que a plataforma que contém este projeto forneça dados necessários para a apresentação do conteúdo em formato JSON.

Para potenciar a criação de listas de reprodução com múltiplos recursos de diferentes tipos, a aplicação deve receber sempre uma lista de recursos, intitulada sequência. Caso esta lista contenha apenas um recurso, a aplicação não instância as opções de navegação.

```

{
  "sequenceId": "123456789",
  "sequenceName": "Uma Sequência de Recursos Educativos",
  "resources": [
    {
      "id": "123456",
      "resourceType": "pdf",
      "title": "Recurso Educativo 1",
      "source": "caminho/para/recursos1.pdf",
      "size": 0,
      "icon": "icon_pdf"
    },
    {
      "id": "1234567",
      "resourceType": "kaltura",
      "title": "Video Educativo 1",
      "source": "id_kaltura",
      "size": 0,
      "icon": "icon_video"
    }
  ]
}

```

Figura 30 - Exemplo de um ficheiro JSON com dados necessários ao funcionamento da aplicação projetada.

4.4.2 Componente Sequence

O componente *Sequence* serve como coração da aplicação tendo a função de adquirir os dados JSON necessários, guardar o principal estado da aplicação, *SequenceState*, e disponibilizar uma API para navegação numa sequência. Este componente é responsável por instanciar o componente *header*, *resource* e *persistence manager*.

4.4.3 Componente Resource

O componente *Resource* recebe como parâmetro um objeto *Resource*, com dados provenientes do JSON de sequência. Com estes dados o componente tem a função de identificar o tipo de recurso e instanciar o componente *player* que melhor o conseguir apresentar.

4.4.4 Componentes Player

Cada um dos componentes que contêm no seu nome a nomenclatura *Player*, tem a função de apresentar um, ou vários, tipos de conteúdo. Para tal, cada um destes componentes dependerá de um ou vários *media players* para a correta apresentação do conteúdo. Para garantir uma boa divisão de responsabilidades, estes componentes podem conter mais que um *Presentational*.

4.4.5 Componente *Header*

O componente *Header* é utilizado como cabeçalho por omissão desta aplicação. Este componente pode por isso ser escondido caso o LMS assim o requirite, permitindo que a plataforma crie o seu próprio cabeçalho com, por exemplo, o seu logotipo. Esta decisão permite um maior foco do módulo de distribuição na sua principal responsabilidade, a apresentação dos recursos educativos, fornecendo simultaneamente aos LMS que o contém maior liberdade para customização da apresentação e das funcionalidades específicas do LMS (partilha com outros utilizadores, adicionar a listas de favoritos, entre outros).

Este componente é principalmente usado para testes em ambiente de desenvolvimento.

4.4.6 *Store Redux*

Foi decidido utilizar a ferramenta Redux, para facilitar a submissão de resultados, uma vez que os dados são adquiridos nos componentes *Player* e submetidos pelo componente *PersistenceManager*. A *store Redux* permite assim simplificar o processo de propagação destes dados, uma vez que ultrapassa as restrições de propagação horizontal descritas na secção 4.3.2.

Para evitar o aumento da complexidade de todo o projeto, e não havendo necessidade funcional para tal, a *store Redux* é apenas utilizada para os estados associados à submissão de resultados e dados analíticos. O estado principal da aplicação, o *SequenceState* descrito na secção 4.4.2, será por isso iniciado e mantido dentro do componente *Sequence*, sendo os dados necessários propagados verticalmente pela aplicação.

4.4.7 Componente *PersistenceManager*

O *PersistenceManager* tem como responsabilidade a submissão de resultados obtidos pelos utilizadores, assim como a submissão de dados analíticos de potencial interesse para a plataforma. Cabe ainda a este componente perceber se a aplicação tem acesso à internet e decidir a submissão destes dados para o servidor ou guardá-los localmente, assim como detetar a existência de dados guardados localmente em sessões anteriores e submetê-los assim que a ligação for reestabelecida.

Este componente deve subscrever aos estados contidos na *Store Redux*.

4.5 Desenho da interface gráfica

A interface gráfica deste projeto é maioritariamente gerida pelos *media players* e pelo conteúdo apresentado por eles, existindo limite no seu nível de personalização. Os principais elementos de interface gráfica para este projeto são por isso os componentes *header*, tanto o componente por omissão incluído na aplicação, descrito na secção 4.4.5, como o componente criado especificamente para a Escola Virtual.

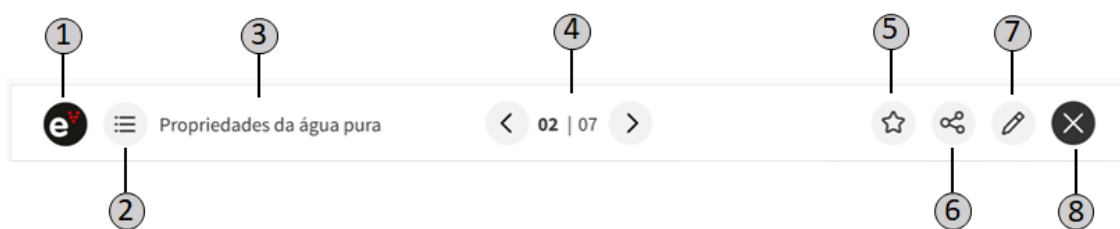


Figura 31 – Design do componente *header* a incluir no LMS da Escola Virtual. Criado pelos designers da Porto Editora para este projeto.

A imagem acima representa a interface gráfica desenhada pela equipa de design da Porto Editora para o cabeçalho do novo módulo de distribuição da Escola Virtual. Começando pelos elementos mais à esquerda é possível ver que este cabeçalho contém o ícone da plataforma (1), seguido de um índice da sequência (2), e por último o título do recurso aberto de momento (3). Ao centro encontra-se o controlo de navegação (4). À direita ficam as opções da plataforma, guardar nos favoritos (5), partilha (6), edição (7) e o botão para fechar a apresentação de conteúdo (8).

De forma a melhorar a experiência de utilização é importante adicionar animações e *feedback* a cada um dos botões apresentados, desde alteração do cursor e cor do botão quando o cursor estiver suspenso em cima do mesmo, alterações de cor após clique ou a apresentação de animações enquanto uma nova página carrega. Quando o cursor é colocado em cima de um botão por alguns segundos uma pequena dica deve ser mostrada para contextualizar o utilizador sobre a funcionalidade. Para melhorar a acessibilidade da interface gráfica deve ser possível navegar e acionar todas as opções utilizando vários instrumentos de interface, como o rato e o teclado.

4.6 Autenticação e segurança

A autenticação e controlo de acesso ao conteúdo deve ser gerido pelo LMS que contém a aplicação. De forma a manter o conteúdo seguro é necessária uma codificação dos dados (JSON de sequência) a passar do LMS para dentro da aplicação, evitando a adulteração dos mesmos e potencial acesso ilícito a recursos da plataforma.

5 Desenvolvimento da solução

Neste capítulo é descrito o processo de desenvolvimento da solução proposta pelo presente documento, começando pela estruturação inicial da aplicação. Segue uma descrição da implementação de funções e utilidades necessárias ao correto funcionamento da aplicação, como os estados React criados, a API de controlo de navegação e um componente *Header*. Depois, é apresentada a implementação dos componentes para visualização de conteúdo e a integração de *media players* no projeto. Por fim é descrito o processo de integração da aplicação na plataforma Escola Virtual.

5.1 Estruturação da aplicação

Para iniciar o desenvolvimento foi criado um repositório GIT e iniciadas as *branches* primárias necessárias para o modelo de trabalho Gitflow, seguido da criação de uma nova aplicação React e execução do primeiro *commit*.

Numa primeira fase foi implementada uma estrutura esqueleto de toda a aplicação projetada no capítulo 4 deste documento, seguindo o padrão *container/presentational* para componentes *Class*. Foram também introduzidos alguns pacotes npm no projeto para facilitar a construção da aplicação, nomeadamente o *i18n*, um módulo de internacionalização simples para permitir a adaptação da aplicação para diferentes línguas (*i18n*, 2022), e o *react-icons*, uma biblioteca que agrega múltiplos conjuntos de ícones populares (*React Icons*, 2022).

De forma a criar um ambiente de desenvolvimento sem dependência da integração com um LMS, foi criado e introduzido na aplicação um JSON de sequência fictício, juntamente com um conjunto

variado de recursos retirados do LMS Escola Virtual. Este conteúdo é descartado no processo de *build*, não sendo por isso incluído na versão compilada da aplicação.

5.2 Criação do estado principal da aplicação

O principal estado da aplicação é chamado de *SequenceState*. Instanciado pelo componente *Sequence*, este estado contém todos os dados do JSON recebido assim como as variáveis relevantes à navegação, como o id atual do recurso.

```
export interface SequenceState {  
  2 references  
  hasChanged: boolean;  
  2 references  
  id: string;  
  2 references  
  name: string;  
  4 references  
  resources: ResourceData[];  
  12 references  
  resourceIds: number[];  
  12 references  
  currentResourceId: number;  
}
```

Figura 32 - *SequenceState* criado para a aplicação.

```
export class ResourceData {  
  1 reference  
  id: number = 0;  
  1 reference  
  externalId: string = "";  
  10 references  
  name: string = "";  
  4 references  
  type: string = "";  
  2 references  
  size: number = 0;  
  1 reference  
  sourceType?: string = "";  
  2 references  
  icon: string = "";  
  12 references  
  source: string = "";  
}
```

Figura 33 - Demonstração da class *ResourceData*, usada para tipificar os dados de cada Recurso no *SequenceState*.

5.3 Navegação de sequência e componente *header*

Dado o requisito de incorporar este módulo de distribuição de recursos em diferentes plataformas/aplicações, foi decidido criar uma API de navegação. Para atingir este objetivo foram criados três eventos HTML personalizados no componente Sequence.

- **sp-next:** apresenta o próximo recurso na sequência.
- **sp-previous:** apresenta o recurso anterior.
- **sp-moveToIndex:** apresenta o recurso cujo índice foi enviado como parâmetro.

Estes eventos podem ser lançados a partir de qualquer parte do DOM, incluindo fora da aplicação React.



Figura 34 - Componente *header* por omissão da aplicação.

O componente *header* demonstrado na figura acima foi desenvolvido para servir como cabeçalho por omissão da aplicação, pelos motivos explicados no capítulo 4.4.5. O componente inclui um índice representativo da sequência e as opções de navegação. Ambos os elementos utilizam os eventos descritos para navegar na sequência.

A interface gráfica deste componente baseia-se no *design* apresentado no capítulo 4.5, tendo sido simplificado, removendo os elementos específicos do LMS da Escola Virtual.

5.4 Apresentação de conteúdo

Esta secção foca nos tipos de recursos suportados pela aplicação, os componentes e *media players* utilizados para os apresentar, começando por descrever a utilização do padrão *Observer* para o redimensionamento dinâmico de conteúdo em relação ao tamanho do ecrã.

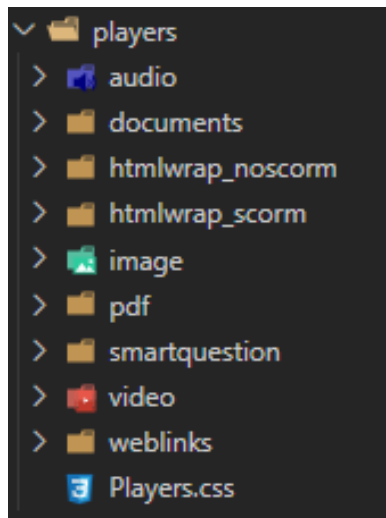


Figura 35 - Componentes *Player* criados no âmbito deste projeto.

5.4.1 Observers e redimensionamento

Para garantir a correta apresentação de alguns tipos de conteúdo, é necessário proceder ao seu redimensionamento. Para este fim foi utilizado o padrão *Observer*.

“O padrão *Observer* oferece um modelo de subscrição no qual objetos subscrevem um evento e são notificados quando o evento ocorre. Este padrão é a base da programação orientada a eventos, incluindo JavaScript. O padrão *Observer* facilita *design* orientado a objetos e promove acoplamento fraco”¹. Desta forma quando um sujeito, elemento a observar, sofre uma alteração, uma função é notificada e executada (Javascript Patterns - Javascript Observer, s.d.).

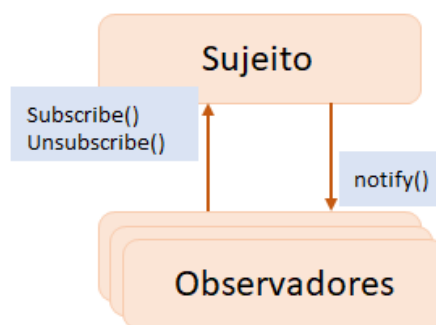


Figura 36 -Diagrama do funcionamento de um *observer*. Adaptado de (Javascript Patterns - Javascript Observer, s.d.)

¹ Tradução livre do autor. No original “*The Observer pattern offers a subscription model in which objects subscribe to an event and get notified when the event occurs. This pattern is the cornerstone of event driven programming, including JavaScript. The Observer pattern facilitates good object-oriented design and promotes loose coupling.*” (Javascript Patterns - Javascript Observer, s.d.)

Este padrão foi especialmente utilizado para garantir o correto dimensionamento de iframes. Desta forma uma iframe terá o mesmo comportamento que outros tipos de elemento HTML, redimensionando-se de acordo com o seu conteúdo. É de notar que esta abordagem só é possível pela garantia que o conteúdo dentro destas iframes se encontra no mesmo domínio que a aplicação.

```
var resizeObserver = new ResizeObserver(entries => {
  for (let entry of entries) {
    iframe.style.height = entry.contentRect.height+"px";
  }
});

resizeObserver.observe(iframeContentWindow.document.body);
```

Figura 37 – Exemplo da utilização do padrão *observer* para o redimensionamento de uma iframe de um recurso Smart Questions.

O código apresentado na imagem acima mostra a inicialização de um observador de eventos de redimensionamento. Este evento é disparado sempre que a largura e/ou altura de um dado elemento é alterada. Como é possível verificar, o sujeito escolhido para este observador foi o elemento `body` do DOM interno da iframe, sendo desta forma igualada a altura do conteúdo com a altura da iframe. Esta função é chamada aproveitando o evento `onload`, suportado nativamente pela tag HTML `iframe`.

5.4.2 Recursos áudio

Como explorado no capítulo 3.3.6, áudio é um dos tipos de media mais utilizado. Para a apresentação de conteúdo áudio foi escolhido o React Audio Player, também descrito no capítulo 3.3.6, tratando-se de um *media player* que cobre todos os requisitos do projeto.

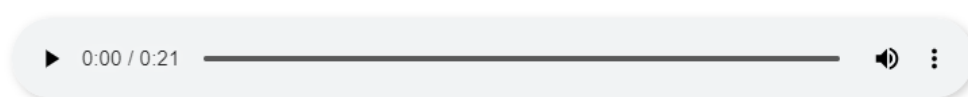


Figura 38 – Demonstração da utilização do *media player* React Audio Player no projeto

O componente é responsivo, ajustando-se perfeitamente a diferentes resoluções, tendo sido posicionado no centro da página e adicionada uma sombra para aumentar o contraste com o fundo branco.

5.4.3 Documentos

Este componente é responsável pela apresentação de todos os conteúdos catalogados como documentos ou *office*, contendo por isso dois *presentationals*.

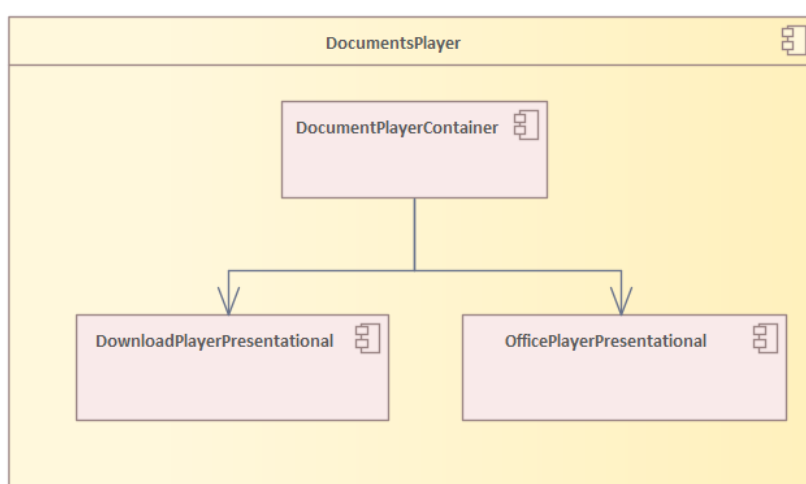


Figura 39 – Diagrama do componente DocumentsPlayer seguindo o padrão Container/Presentational com múltiplos presentational.

Recursos catalogados como documentos tem extensões variadas, podendo inclusivamente incluir arquivos .zip. Estes recursos podem ser submetidos pelos utilizadores da plataforma com perfil professor e partilhados com alunos, sendo certas extensões (.exe por exemplo) proibidas pela plataforma e as restantes devidamente analisados por um antivírus. Não existindo um *media player* para estes conteúdos, estes recursos são diretamente descarregados pelo utilizador.

Como demonstrado na imagem abaixo, o tamanho do conteúdo é apresentado, assim como a sua extensão, de forma permitir ao utilizador tomar uma decisão informada.

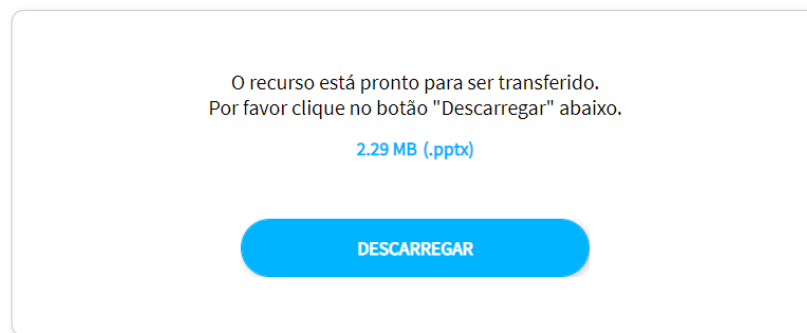


Figura 40 - Demonstração da funcionalidade de Download de documentos.

Quanto a documentos *Office*, a aplicação deteta a sua extensão e utiliza o Microsoft Office Viewer, descrito no capítulo 3.3.6, para a sua apresentação dentro da plataforma. Este *office viewer* foi escolhido pelas funcionalidades de tradução e acessibilidade. Tratando-se de um *player online*, quando o utilizador está *offline*, e assumindo que descarregou previamente os recursos, a opção de transferência descrita anteriormente é utilizada.

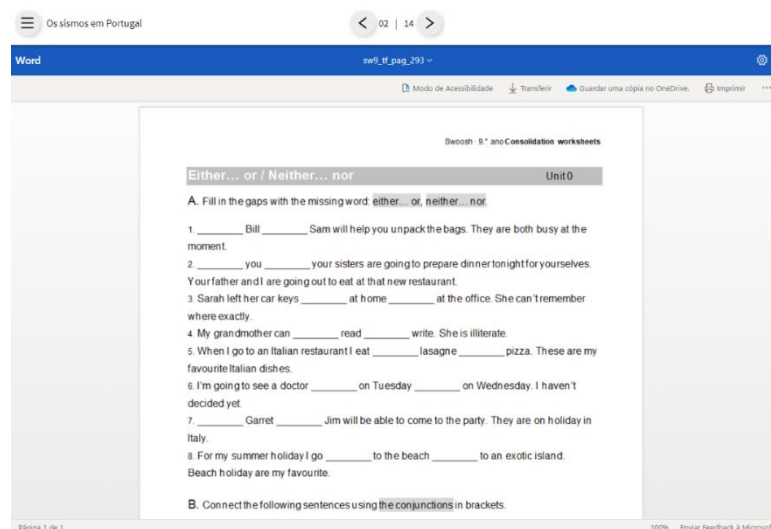


Figura 41 - Demonstração do Microsoft Office Viewer no módulo de distribuição desenvolvido.

5.4.4 Recursos `htmlwrap_noscom`

Conteúdo catalogado como “`htmlwrap_noscom`” é apresentado com recurso a uma `iframe`, não requerendo um *media player*. Trata-se de um tipo de recurso com uma grande variedade de fonte, funcionamento e estrutura, contendo *quizes*, apresentações interativas e jogos educativos HTML5 e WebGL.



Figura 42 - Demonstração de um jogo HTML5 para a disciplina e inglês.

Dada a variedade do conteúdo apresentado são um tipo de conteúdo difícil de corretamente redimensionar, contendo recursos que podem ou não ser responsivos e não respeitando uma regra única na sua estrutura. Para atingir a correta apresentação destes conteúdos foi utilizado o padrão *observer*, descrito anteriormente neste documento, tendo sido criados *observers* diferentes dependendo da estrutura detetada dentro do conteúdo da iframe.

```
if (isCanvasResource === true) {  
  // Resize Observer for canvas Resources (Games)  
  1 reference  
  var resizeCanvasObserver = new ResizeObserver(entries => {  
    for (let entry of entries) {  
      if (entry.contentRect.height > 0) {  
        iframe.style.height = entry.contentRect.height+"px";  
      }  
    }  
  });  
  noScormPlayer.classList.add("flex");  
  resizeCanvasObserver.observe(iframeBody.getElementsByTagName("canvas")[0]);  
}
```

Figura 43 - Exemplo de *observer* criado para o redimensionamento de recursos que utilizam o elemento HTML canvas.

O excerto de código apresentado na figura anterior é utilizado para o redimensionamento de jogos apresentados em elementos HTML Canvas, sendo detetada a sua presença e utilizada para iniciar o *observer* correto.

5.4.5 Recursos htmlwrap_scorm

Recursos “htmlwrap_scorm” do LMS da Escola Virtual são conteúdos criados na ferramenta de autoria mAuthor, descrita no capítulo 3.1.5. Estes recursos são apresentados com recurso ao icPlayer, descrito no capítulo 3.3.2. Este *media player* requer a utilização de uma *iframe*, uma vez que carrega múltiplas dependências externas com potencial para gerar conflitos com outras dependências da plataforma LMS, como é exemplo a versão específica do JQuery necessária (o *player* depende da versão 1.7.1, não sendo compatível com a versão mais recente presente na Escola Virtual).

Os ficheiros do icPlayer foram adicionados à pasta *public* do projeto, juntamente com um ficheiro html, intitulado *loadscorm.html*, utilizado para popular a *iframe* e iniciar o *player* dentro do mesmo domínio.

Estes recursos podem ter mais que um esquema de apresentação (*layout*), contendo uma série de funções que permitem a escolha do layout. Estes *layouts* têm como objetivo permitir a apresentação do conteúdo da forma mais apropriado à orientação e resolução do ecrã. Alguns destes layouts apresentam avisos ao utilizador sobre, por exemplo, a melhor orientação para ver o conteúdo.



Figura 44 - Demonstração do recurso mAuthor “A 1ª Guerra Mundial: as frentes do conflito” apresentado em dois *layouts* diferentes.

Mesmo aplicando o *layout* mais adequado à resolução e orientação do ecrã, estes recursos não são totalmente responsivos, sendo por isso necessário recorrer ao padrão *observer*, descrito no capítulo 5.4.1, e calcular uma escala que permita ao conteúdo ser corretamente dimensionado.

```
// Calculate content scale and position (based on new scale)
let contentScale = availableWidth / contentWidth;
let repositionTop = (contentHeight*contentScale - contentHeight) / 2 ;
let repositionLeft = (availableWidth - contentWidth ) / 2;
```

Figura 45 – Cálculo da escala baseado na largura disponível e largura do conteúdo.

Como demonstrado na figura acima, a escala é obtida pela divisão da largura disponível, pela largura do conteúdo. A transformação do conteúdo através de uma escala requer ainda o cálculo da posição do conteúdo. Isto acontece uma vez que a propriedade *css* utilizada, *transform*, aplica alterações à posição do elemento no DOM. O valor de reposicionamento horizontal é facilmente adquirido subtraindo à largura disponível a largura original do recurso, dividido por dois, dado que o valor de escala aplicado é baseado nas diferenças de largura do conteúdo. O valor de reposicionamento vertical é obtido subtraindo à nova altura do conteúdo, calculada multiplicando a altura original pela nova escala, e a altura original, dividido por dois.

5.4.6 Imagens

Imagens são um tipo simples de conteúdo média facilmente representado em sites e aplicações web. De forma a melhorar a apresentação da imagem foi utilizado o pacote npm *react-image-zoom* (*react-image-zoom*, 2020), adicionando a possibilidade ao utilizador de aumentar o tamanho da mesma.

5.4.7 PDF

Para a apresentação de documentos PDF foi escolhido o componente *React PDF* descrito no capítulo 3.3.4. Dada a sua simplicidade, foi necessário desenvolver algumas funcionalidades extra para a apresentação, como um índice de páginas e navegação por número de página.

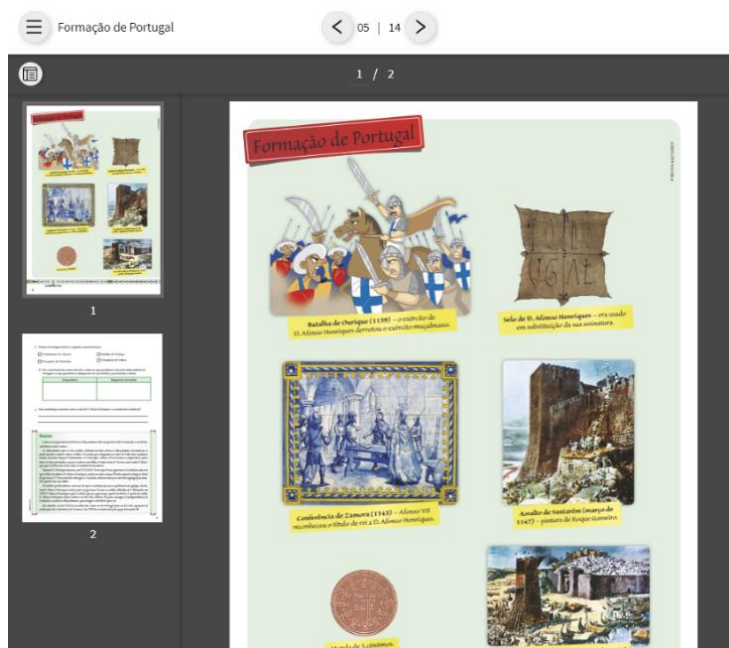


Figura 46 - Demonstração de um PDF apresentado pelo novo módulo de distribuição desenvolvido.

Para a apresentação do documento PDF este *player* depende de um *pdf.worker*, responsável pela análise e desenho das páginas do documento. Este *worker* foi descarregado do repositório recomendado na documentação do componente e adicionado à pasta *public* do projeto para que possa ser carregado durante a execução da aplicação.

Para redimensionamento deste componente foram utilizados *breakpoints* nos estilos *css* que alteram ligeiramente o layout apresentado. Desta forma o espaço disponível em ecrã é mais bem aproveitado.

5.4.8 Recursos Smart Questions

Como já referido no capítulo 3.1.5 e 3.3.3 desta dissertação, o Smart Questions (SQ) é um tipo de recurso avaliativo criado dentro da plataforma Escola Virtual, sendo o principal recurso avaliativo deste LMS. O recurso é criado dentro do módulo SQ do website e deve ser apresentado pelo módulo de distribuição de conteúdo de forma a permitir a sua inclusão em sequências de recursos.

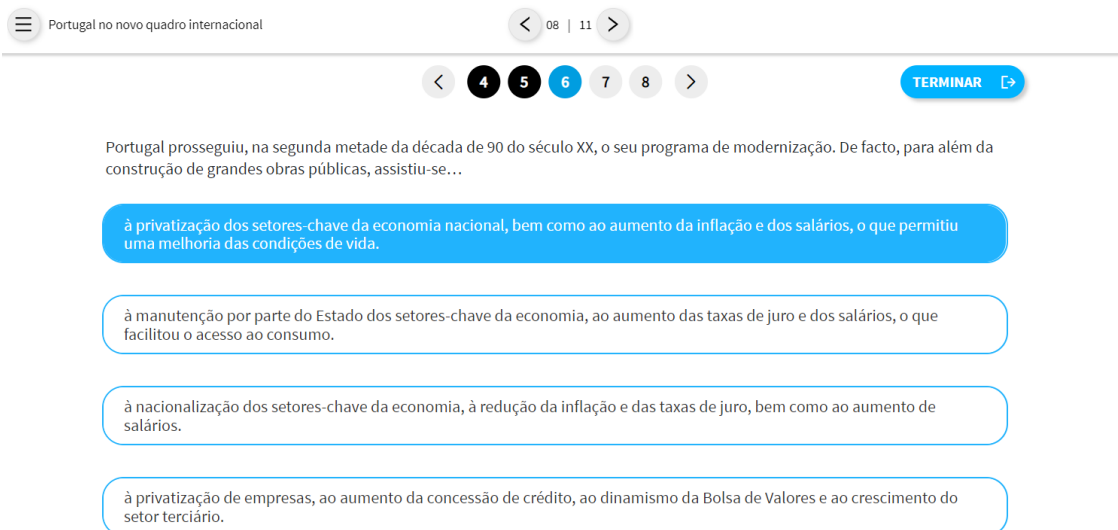


Figura 47 - Demonstração de um recurso SQ apresentado no novo módulo de distribuição.

Para a apresentação deste conteúdo recorreu-se ao uso de uma *iframe*. Tal como explorado no capítulo 5.4.1, o processo de redimensionamento da *iframe* deste componente é feito com recurso ao padrão *observer*.

5.4.9 Vídeos

Como descrito no capítulo 3.1.6 e 3.3.1, a Escola Virtual utiliza o Kaltura para gestão e apresentação da grande maioria do seu conteúdo vídeo. No entanto, a plataforma também contém conteúdo vídeo em formato *mp4*, sobretudo conteúdo submetido pelos utilizadores. Existe ainda conteúdo vídeo no formato de links para o Youtube, sendo catalogados como *weblinks*, e por isso tratados por outro *player*, descrito na secção 5.4.10.

Contendo dois *players* diferentes para a apresentação de vídeos, este componente contém dois *presentationals*.

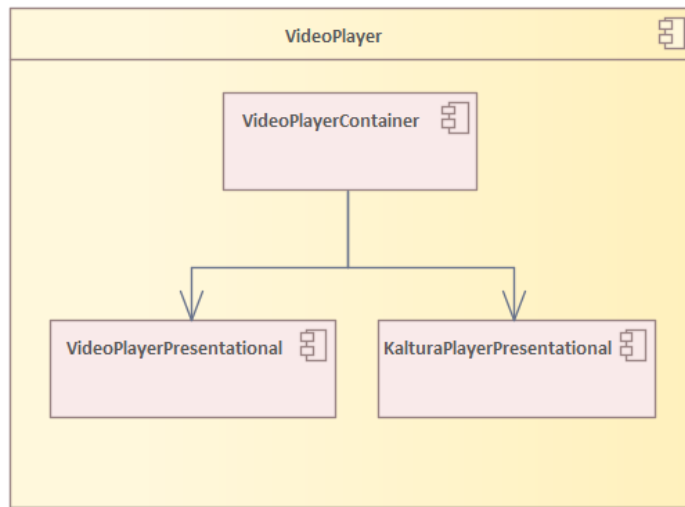


Figura 48 - Diagrama do componente VideoPlayer seguindo o padrão Container/Presentational com múltiplos presentational.

O Kaltura é um *player* HTML5 sendo por isso necessário recorrer a uma *iframe* para a sua correta apresentação. No entanto, tratando-se de um *player* completamente responsivo, mantendo o *aspect ratio* do vídeo em ecrãs de diferentes tamanhos e resoluções de ecrã, não é necessário proceder ao cálculo de escala ou redimensionamentos.



Figura 49 – Demonstração do *Player* Kaltura no novo módulo de distribuição de conteúdo.

Para a apresentação dos ficheiros mp4 é utilizada a *tag* HTML <video>, descrita no capítulo 3.3.1. Através da aplicação de alguns estilos css, foi possível fazer com que o elemento se ajustasse corretamente ao tamanho do ecrã, mantendo o seu *aspect ratio*.

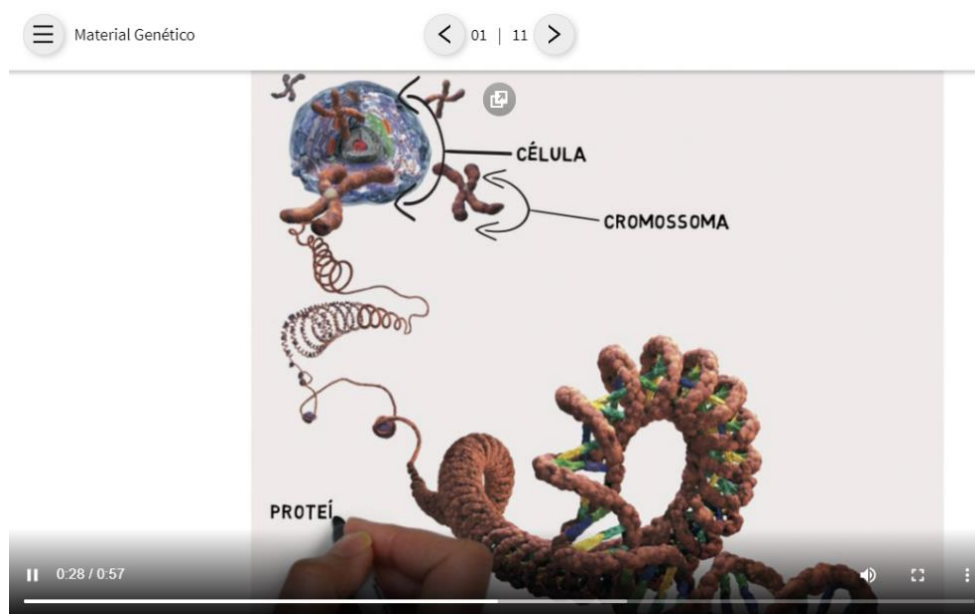


Figura 50 - Demonstração da apresentação de vídeos mp4 no módulo de distribuição de conteúdo.

5.4.10 Weblinks

Os conteúdos catalogados como *weblinks* no LMS da Escola Virtual contêm hiperligações para múltiplos *websites*. Tratando-se de um tipo de recurso que, no caso concreto em estudo, pode ser criado pelos próprios utilizadores (com acesso à plataforma numa conta de perfil professor), não é possível garantir a correta apresentação dos mesmos dentro de uma *iframe*. É também importante minimizar o risco de fraude ou apresentação de conteúdo indevido, existindo para este fim uma série de regras para a criação deste tipo de recurso. Mesmo com estas regras implementadas, o módulo de distribuição deve precaver o utilizador da saída do domínio atual para um não controlado pela plataforma.

Existe uma hiperligação que pode ser facilmente identificada e tratada de forma a evitar a saída dos utilizadores de dentro da plataforma: os *weblinks* Youtube. Este componente contém por isso dois *Presentationals*.

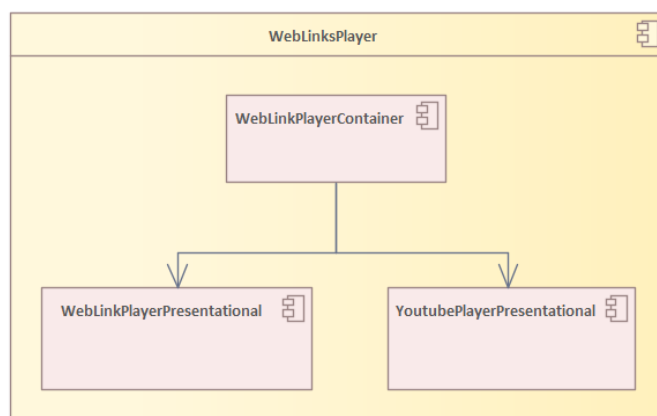


Figura 51 - Diagrama do componente WebLinksPlayer seguindo o padrão *Container/Presentational* com dois *presentational*.

Desta forma, quando o recurso contém um link para o Youtube, o vídeo é apresentado utilizando o Youtube Iframe Player API, descrito no capítulo 3.3.1. Para os restantes recursos deste tipo é apresentada uma página como demonstrado na imagem abaixo. Esta página mostra um aviso sobre a saída do domínio da plataforma, mostrando a hiperligação e permitindo ao utilizador tomar uma decisão consciente e informada sobre continuar.



Figura 52 - Demonstração da página de aviso para recursos *weblink*.

5.5 Submissão de resultados e dados analíticos

A submissão de resultados não foi concluída durante a elaboração deste projeto dissertação, tendo sido iniciada uma base para a sua implementação futura. Como descrito no capítulo 4.4.6, foi instanciada uma *store* Redux na raiz da aplicação e criados dois estados, *resultsState* e *analyticsState*, responsáveis por armazenar, respetivamente, os resultados e dados analíticos capturados pelos componentes *player*.

Foi também iniciado o desenvolvimento do componente para gestão de persistência, PersistenceManager. Este componente subscreve os estados Redux mencionados anteriormente tomando decisões sobre a sua persistência local ou em base de dados, dependendo da sua capacidade de ligação à internet.

5.6 Integração com a Escola Virtual

A aplicação criada foi integrada na Escola Virtual num ambiente de pré-produção, em conjunto com uma nova funcionalidade: pré-visualização de recursos. Foi escolhida como primeira abordagem a integração nesta nova funcionalidade dado o novo módulo de distribuição não possuir ainda a capacidade de submissão de resultados, e não sendo essa funcionalidade necessária para um contexto de pré-visualização.

Esta nova funcionalidade era um requisito difícil de alcançar com a módulo Smart Player anterior devido à sua estrutura dependente de iframes, sendo por isso uma boa demonstração das possibilidades abertas pela nova aplicação desenvolvida.

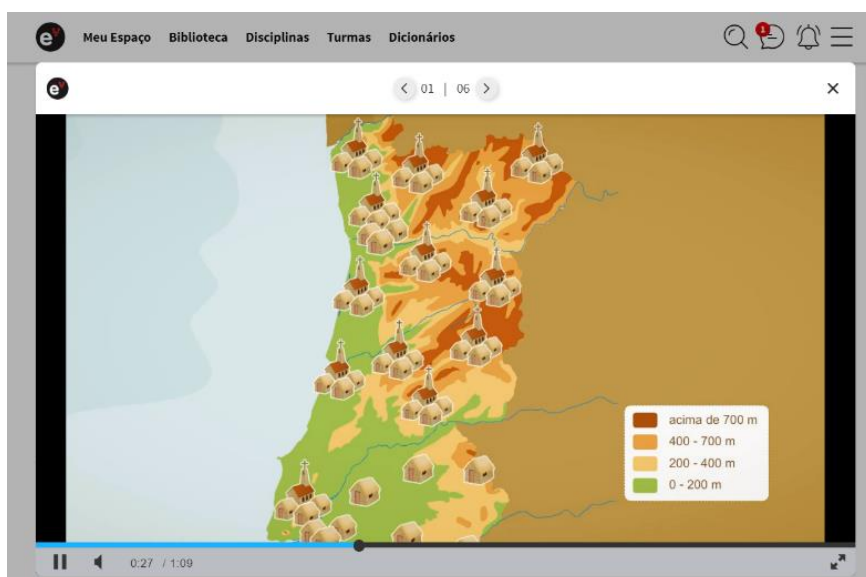


Figura 53 - Demonstração da funcionalidade "pré-visualizar" dentro da plataforma Escola Virtual, utilizando a aplicação de distribuição de conteúdo desenvolvida no decorrer deste projeto. Imagem retirada da plataforma Escola Virtual.

A figura acima demonstra a inclusão da aplicação no LMS da Escola Virtual. Para esta demonstração foi criado um componente *header* específico para a plataforma contendo o ícone

da plataforma. Tratando-se de uma funcionalidade de pré-visualização, as restantes funcionalidades do cabeçalho são escondidas.

5.6.1 Componente *header* na Escola Virtual

Como já mencionado, foi desenvolvido um cabeçalho para a aplicação na Escola Virtual. Todas as funcionalidades adicionadas a este elemento são opcionais, conseguindo assim servir todas as necessidades da plataforma.

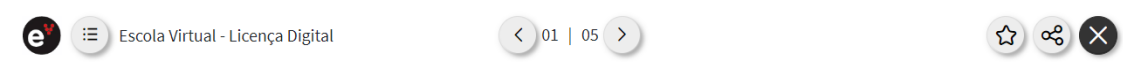


Figura 54 - Cabeçalho desenvolvido para a Escola Virtual.

Quando comparado com o *design* apresentado no capítulo 4.4.5, é possível identificar algumas alterações. Para aumentar o contraste entre o fundo branco e o cinzento dos elementos foi adicionado uma sombra em cada um deles. A maioria dos botões deste cabeçalho abrem menus com pelo menos duas opções.

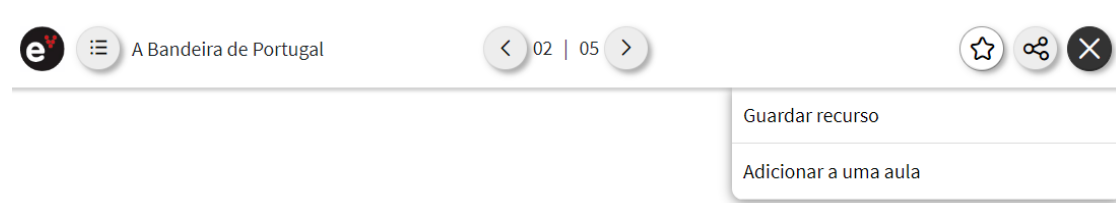


Figura 55 – Demonstração do menu “Guardar” do cabeçalho desenvolvido para a Escola Virtual.

Quando em dispositivos de menor dimensão, o *layout* apresentado é alterado, sendo alguma informação omitida nesse esquema de visualização, como o título do recurso. As opções presentes à direita são também concatenadas num menu único, com exceção da opção de fechar a apresentação de recursos.

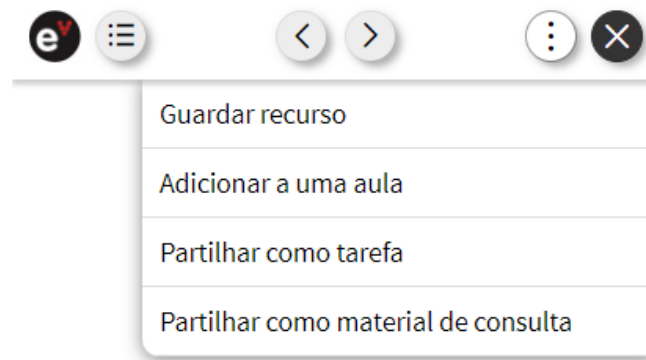


Figura 56- Demonstração do esquema de visualização do cabeçalho quando em layout mobile.

5.6.2 Compilação, disponibilização e integração

Como referido no capítulo 4.2.2 foi utilizada uma pipeline Jenkins para o processo de entrega e integração da aplicação no projeto Escola Virtual. Este processo está dividido em duas etapas: Compilação e Integração.

Compilação – Este processo é despoletado manualmente, gerando um ficheiro *.tar (Tape Archive File)* contendo a aplicação compilada.

Integração – Foi adicionada na pipeline de disponibilização da Escola Virtual uma etapa responsável por incluir a última versão compilada (com sucesso) da nova aplicação Smart Player.

5.7 Validação da aplicação

A aplicação de distribuição criada ao longo deste projeto para o LMS da Porto Editora foi entregue à equipa de controlo de qualidade da empresa, tendo sido extensivamente testada por várias pessoas, em múltiplos dispositivos eletrónicos, com diversos sistemas operativos e resoluções de ecrã.

Após a correção dos casos de erro reportados por esta equipa, a aplicação foi aprovada e disponibilizada no ambiente de produção, estando desde setembro deste ano (2022) disponível no site Escola Virtual.

Nome do Recurso	Tipo	ID Recurso	WEB	Android	IOS	Observações
Possessive Determiners	Mauthor Interactive	2994187 » » 2669015	NOK	OK	OK	Em Firefox este recurso não se encontra devidamente centrado » » Após alteração está a funcionar devidamente
Personal Pronouns	Mauthor Interactive	2994207 » » 2669007	NOK	OK	OK	Em Firefox este recurso não se encontra devidamente centrado » » Após alteração está a funcionar devidamente
Manual Interativo	Mauthor Interactive	3265324	OK	OK	OK	
O Sol e os Pontos Cardeais	Mauthor Duplo Layout	4002786	OK	OK	OK	
Distintuir naturalidade e nacionalidade	Mauthor Duplo Layout	4008027	OK	OK	OK	
Identificar a posição relativa entre retas	Mauthor Duplo Layout	4193364	OK	OK	OK	
Compreender um excerto: "O primeiro dia de aulas de Mariana"	Mauthor Duplo Layout	10986057	OK	OK	OK	
Reconhecer o significado do chocolate na vida de Mariana	Mauthor Duplo Layout	10986243	OK	OK	NOK	Em iOS o recurso não faz scroll em landscape e está desajustado
Resistência à Invasão Castelhana	Mauthor c/ Video	829161	OK	OK	OK	
Espetros e Aplicações da Espectroscopia	Mauthor c/ Video	17093	OK	OK	OK	
A Batalha de Aljubarrota	Kaltura	819279	OK	OK	OK	
Naturalidade e nacionalidade	Kaltura	4002442	OK	OK	OK	
Adverbs of Frequency	Kaltura c/interatividade	2669452	OK	OK	OK	
Audio - Regresso as Aulas	Audio	3097518	OK	OK	OK	
Mapa Mundo	Imagem 160:149	816169	OK	OK	OK	
Alfredo Cunha, A Cortina dos Dias	Imagem 709:1049	746193	OK	OK	OK	
Templo romano de Évora, Portugal	Imagem 59:40	5876993	OK	OK	OK	
Guia de Profissões	NoScorm Quiz	4948794	OK	OK	NOK	Em iOS não é possível efetuar scroll para visualizar o recursos integralmente.
Verb to be	NoScorm Digital Game	3703254	OK	OK	OK	
Itinerário na Serra	NoScorm Duplo Layout	4002794	OK	OK	OK	
Planificação a médio prazo F – Portugal Hoje	Office Word	2989650	OK	NOK	NOK	Em Android e iOS o download do recurso não está disponível.
ARA nº 8 - Portugal Hoje	Office Word	2989859	OK	NOK	NOK	Em Android e iOS o download do recurso não está disponível.
Ficha de Avaliação Trimestral 1	Office Word	3643619	OK	NOK	NOK	Em Android e iOS o download do recurso não está disponível.
Setores de atividade económica em Portugal (dados de 2012 a 2014)	Office Excel	2416714	OK	OK	OK	
Dados demográficos de Portugal continental (dados de 1960 a 2014)	Office Excel	2416718	OK	OK	OK	
Avaliação 6 A - Portugal hoje - grelhas	Office Excel	2870123	OK	OK	OK	
Recursos litológicos de Portugal	Office Power Point	1162564	OK	OK	OK	
Os sismos em Portugal	Office Power Point	1162677	OK	OK	OK	
Entidades de proteção e conservação da Natureza em Portugal	Office Power Point	1251461	OK	OK	OK	

Figura 57 – Excerto do ficheiro Excel entregue pela equipa de controlo de qualidade com os resultados dos primeiros testes funcionais efetuados num ambiente de pré-produção.

6 Conclusões

Neste capítulo será realizada uma reflexão sobre o projeto, iniciando com uma síntese do trabalho realizado. Em seguida são apresentados os objetivos concretizados e, para terminar, o trabalho futuro a realizar.

6.1 Síntese

Durante esta dissertação procurou-se responder à questão “Qual a melhor forma de distribuir e visualizar recursos educativos interativos numa plataforma de e-Learning?”. Para responder a esta pergunta foi feito um estudo e análise de plataformas LMS, de *media players*, de tecnologias e de *frameworks* de desenvolvimento para *front-end*, seguido do desenho e desenvolvimento de um módulo de distribuição de recursos educativos para a plataforma Escola Virtual da empresa Porto Editora.

6.2 Objetivos realizados

A aplicação desenvolvida para a Escola Virtual ao longo deste trabalho possui a capacidade de corretamente apresentar todos os tipos de conteúdo presentes no LMS, em diferentes dispositivos, sistemas operativos e resoluções, tal com requerido pela Porto Editora. Pode por isso considerar-se que foi obtido sucesso no desenvolvimento de uma aplicação para a distribuição e visualização de recursos educativos.

Tabela 6 - Grau de realização dos requisitos funcionais propostos

Requisito funcional	Grau de realização
Distribuição e apresentação de conteúdos educativos <i>online</i>	Implementado
Distribuição e apresentação de conteúdos educativos <i>offline</i>	Implementado
Captura de resultados e dados de utilização	Iniciado
Persistência de dados <i>online</i>	Iniciado
Persistência de dados <i>offline</i>	Iniciado

Através do estudo realizado ao longo deste projeto, assim como o trabalho de pesquisa e análise elaborado, é possível concluir que a biblioteca React é uma ferramenta flexível que permite a criação de aplicações modulares com boa separação de responsabilidades. Esta modularidade facilita a alteração, atualização e expansão das funcionalidades incluídas na aplicação, agilizando o processo de desenvolvimento e manutenção.

A possibilidade de facilmente integrar *media players* de várias fontes, assim como a existência de uma grande variedade de *media players* especificamente desenvolvidos para a biblioteca React, simplifica a construção de aplicações com a finalidade de distribuição e apresentação de conteúdos educativos, assim como a expansão gradual das suas funcionalidades.

6.3 Trabalho futuro

Embora a solução desenvolvida cumpra com o principal requisito para esta dissertação, a correta distribuição e apresentação de conteúdos educativos, há ainda funcionalidades secundárias e carência para a observância de todos os objetivos práticos propostos pelo Grupo Porto Editora.

Como mencionado anteriormente, as funcionalidades de captura e submissão de dados por parte da aplicação concebida encontram-se incompletas. A recolha de resultados e dados de utilização (analíticos), assim como o potencial para guardá-los localmente enquanto *offline* ou submetê-los para uma base de dados quando *online* é uma funcionalidade que acrescenta valor ao LMS da

Escola Virtual. Estes dados podem ser utilizados para a criação de um perfil cognitivo que melhor informe o LMS sobre o conteúdo a recomendar a cada um dos seus utilizadores, assim como adquirir dados importantes para que a organização decida o melhor tipo de conteúdo a produzir, refazer ou melhorar. Esta funcionalidade é a próxima etapa na maturação da aplicação desenvolvida.

Foi também identificada a possibilidade de expandir as capacidades de alguns dos componentes *player* implementados, na forma do acréscimo de novas ferramentas de usabilidade e acessibilidade como é exemplo o desenvolvimento de uma funcionalidade de ampliação ao PDF *viewer* utilizado.

7 Referências

#EstudoEmCasa 20-21. (2020). Obtido em janeiro de 2022, de *#EstudoEmCasa*: <https://www.rtp.pt/play/estudoemcasa/sobre>

@cyntler/react-doc-viewer. (9 de setembro de 2022). Obtido em setembro de 2022, de npm: <https://www.npmjs.com/package/@cyntler/react-doc-viewer>

A Git Origin Story. (27 de junho de 2018). *Linux Journal*. Obtido em agosto de 2022, de Linux Journal: <https://www.linuxjournal.com/content/git-origin-story>

Acessibilidade do sítio web Europarl. (s.d.). Obtido em julho de 2022, de Parlamento Europeu: <https://www.europarl.europa.eu/portal/pt/accessibility>

Adobe PDF Embed API. (s.d.). Obtido em fevereiro de 2022, de Adobe Developer: <https://developer.adobe.com/document-services/apis/pdf-embed/>

Adobe PDF Embed API Demo. (s.d.). Obtido em fevereiro de 2022, de Adobe Document Cloud: https://documentcloud.adobe.com/view-sdk-demo/index.html#/customize/IN_LINE/Bodea%20Brochure.pdf

Adobe PDF Embed API Quickstarts. (s.d.). Obtido em fevereiro de 2022, de Adobe Developer: <https://developer.adobe.com/document-services/docs/overview/pdf-embed-api/>

Angular versioning and releases. (s.d.). Obtido em maio de 2022, de angular: <https://angular.io/guide/releases>

angular.js v1.2.28. (15 de dezembro de 2014). Obtido de github:
<https://github.com/angular/angular.js/releases/tag/v1.2.28>

Atlassian - Gitflow. (s.d.). Obtido em abril de 2022, de Atlassian:
<https://www.atlassian.com/git/tutorials/comparing-workflows/gitflow-workflow>

Aula Digital - Banco de Recursos. (s.d.). Obtido em agosto de 2022, de Aula Digital:
<https://auladigital.leya.com/student/resourcebank>

Aula Digital - Oferta escolar para Alunos. (s.d.). Obtido em agosto de 2022, de Aula Digital:
<https://auladigital.leya.com/student/offer>

Aula Digital - Oferta Professores. (s.d.). Obtido em agosto de 2022, de Aula Digital:
<https://auladigital.leya.com/teacher/auladigital>

Aula Digital - Recursos. (s.d.). Obtido em agosto de 2022, de Aula Digital:
<https://auladigital.leya.com/recursos/auladigital>

Beetham, H., & Sharpe, R. (2007). *Rethinking Pedagogy for a Digital Age: Designing and Delivering E-Learning*. Routledge. Obtido em dezembro de 2021

Blackboard - Our purpose. (s.d.). Obtido em fevereiro de 2022, de Blackboard:
<https://www.blackboard.com/en-eu/about-us/our-purpose>

Blackboard - Supported File Types. (s.d.). Obtido em fevereiro de 2022, de Blackboard:
https://help.blackboard.com/Learn/Instructor/Ultra/Course_Content/Create_Content/Create_Course_Materials/Supported_File_Types

Capterra Blackboard Learn. (s.d.). Obtido em fevereiro de 2022, de Capterra:
<https://www.capterra.com/p/14500/Blackboard-LMS-for-Business/>

Concorrência notificada da venda da editora LeYa a grupo holandês. (15 de março de 2022). *Observador*. Obtido de <https://observador.pt/2022/03/15/concorrenca-notificada-da-venda-da-editora-leya-a-grupo-holandes/>

Create React App - Adding Typescript. (s.d.). Obtido em abril de 2022, de Create React App:
<https://create-react-app.dev/docs/adding-typescript/>

Easy LMS - What is a learning management system. (s.d.). Obtido em dezembro de 2021, de Easy LMS: <https://www.easy-lms.com/knowledge-center/lms-knowledge-center/what-is-a-learning-management-system/item10182>

Easy LMS - What is SCORM. (s.d.). Obtido em janeiro de 2022, de Easy LMS: <https://www.easy-lms.com/knowledge-center/lms-knowledge-center/what-is-scorm/item10195>

Edutechnica - 9th annual lms data update. (05 de novembro de 2021). (EdTech) Obtido em novembro de 2021, de Edutechnica: <https://edutechnica.com/2021/11/05/9th-annual-lms-data-update/>

Encyclopedia - LMS. (s.d.). Obtido em dezembro de 2021, de PC Magazine: <https://www.pcmag.com/encyclopedia/term/lms>

Encyclopedia - Media Player. (s.d.). Obtido em dezembro de 2021, de PC Magazine: <https://www.pcmag.com/encyclopedia/term/media-player>

Escola Virtual - Eu Leio. (2022). Obtido em fevereiro de 2022, de Escola Virtual: <https://www.escolavirtual.pt/Projetos-EV/euleio.htm>

Escola Virtual - EV 360º. (2022). Obtido em setembro de 2022, de Escola Virtual: <https://www.escolavirtual.pt/Instituicoes/ev360.htm>

Escola Virtual - Licença Individual Professor. (2022). Obtido em setembro de 2022, de Escola Virtual: <https://www.escolavirtual.pt/Professores/licencaindividual.htm>

Escola Virtual - Professor Adotante. (2022). Obtido em setembro de 2022, de Escola Virtual: <https://www.escolavirtual.pt/Professores/professoradotante.htm>

Escola Virtual - Subscrição Aluno e Pais. (2022). Obtido em setembro de 2022, de Escola Virtual: <https://www.escolavirtual.pt/Alunos-e-Pais/subscricao-premium.htm>

Escola Virtual - Subscrição Premium Aluno. (2022). Obtido em setembro de 2022, de Escola Virtual: <https://www.escolavirtual.pt/Alunos-e-Pais/subscricao-premium.htm>

Facebook, Microsoft e Zoom: quem ganha a guerra da videoconferência? (28 de fevereiro de 2020). *Exame*. Obtido em novembro de 2021, de <https://exame.com/tecnologia/mercado-de-videochamadas-tem-dominio-do-zoom-e-ameaca-do-facebook/>

Git. (s.d.). Obtido em setembro de 2022, de Git: <https://git-scm.com>

Git - A short History of Git. (s.d.). Obtido em agosto de 2022, de Git: <https://git-scm.com/book/en/v2/Getting-Started-A-Short-History-of-Git>

Godwin-Jones, R. (maio de 2004). Emerging Technologies: Learning Objects--Scorn or SCORM? *Language Learning & Technology*, 8(2), pp. 7-12. Obtido em janeiro de 2022

Gonçalves, P. (25 de março de 2020). Porto Editora responde a problemas na Escola Virtual: “São dezenas de milhares de novos alunos todos os dias”. (R. Ferreira, Entrevistador) Portugal: Exame Informática. Obtido em novembro de 2021, de <https://visao.sapo.pt/exameinformatica/noticias-ei/internet/2020-03-25-porto-editora-responde-problemas-escola-virtual/>

Google Docs About. (s.d.). Obtido em fevereiro de 2022, de Google: <https://www.google.com/docs/about/>

Google Play - Aula Digital. (s.d.). Obtido em setembro de 2022, de Google Play: https://play.google.com/store/apps/details?id=com.leya.educacao.manualdigital&hl=pt_PT

Google Play - Smart - Aula digital. (s.d.). Obtido em setembro de 2022, de Google Play: https://play.google.com/store/apps/details?id=com.leya.smart&hl=pt_PT

How to Meet WCAG. (julho de 2022). Obtido de Web Accessibility: <https://www.w3.org/WAI/WCAG21/quickref/>

HTML Audio. (s.d.). Obtido em abril de 2022, de w3 schools: https://www.w3schools.com/html/html5_audio.asp

HTML Video. (s.d.). Obtido em junho de 2022, de w3schools: https://www.w3schools.com/html/html5_video.asp

i18n. (maio de 2022). Obtido em maio de 2022, de npm: <https://www.npmjs.com/package/i18n>

IcPlayer - mAuthor Open Source Player. (s.d.). Obtido em março de 2022, de IcPlayer: <https://icplayer.com>

Infinitas Learning - About us. (s.d.). Obtido em abril de 2022, de Infinitas Learning: <https://www.infinitaslearning.com/about-us/>

Javascript Patterns - Javascript Observer. (s.d.). Obtido em julho de 2022, de dofactory: <https://www.dofactory.com/javascript/design-patterns/observer#:~:text=The%20Observer%20pattern%20offers%20a,design%20and%20promotes%20loose%20coupling>.

Kaltura Everything Video. (s.d.). Obtido em janeiro de 2022, de Kaltura: <https://corp.kaltura.com/company/about/>

Kaltura Player. (s.d.). Obtido em dezembro de 2021, de Kaltura: <https://corp.kaltura.com/video-content-management-system/player-studio/>

Kaltura Player ToolKit. (s.d.). (Kaltura) Obtido em dezembro de 2021, de Kaltura: <http://player.kaltura.com/docs/>

Kaltura Video Cloud Platform for Education. (s.d.). (Kaltura) Obtido em dezembro de 2021, de Kaltura: <https://corp.kaltura.com/education-video-platform/>

Kaltura Video Content Management System. (s.d.). Obtido em janeiro de 2022, de Kaltura: <https://corp.kaltura.com/video-content-management-system/>

Khronos Group. (s.d.). *WebGL Low-level 3D graphics api based on OpenGL ES.* Obtido em fevereiro de 2022, de Khronos: <https://www.khronos.org/webgl/>

Koen, P., Ajamian, G., Boyce, S., Clamen, A., Fisher, E., Fountoulakis, S., . . . Seibert, R. (2004). Fuzzy Front End: Effective Methods, Tools, and Techniques. Em P. Belliveau, A. Griffin, & S. Somermeyer, *The PDMA ToolBook 1 for New product Development* (p. 480). Obtido em 2022 de Janeiro

Koen, P., Ajamian, G., Burkart, R., Clamen, A., Davidson, J., D'Amore, R., . . . Wagner, K. (2001). Providing Clarity and a Common Language To the "Fuzzy Front End". *Research-Technology Management*, 44#2, 46-55. Obtido em janeiro de 2021

Lima, S. D. (30 de janeiro de 2020). *A reduced explanation of Redux. (Reducers, Actions, and Store).* Obtido em julho de 2022, de <https://levelup.gitconnected.com/a-reduced-explanation-of-redux-reducers-actions-and-store-9b7819c5d064>

mAuthor. (s.d.). Obtido em janeiro de 2022, de mAuthor: <https://www.mauthor.com>

mAuthor About us. (s.d.). Obtido em dezembro de 2021, de mauthor: https://www.mauthor.com/about_us

mAuthor SCORM packages. (s.d.). Obtido em fevereiro de 2022, de mAuthor: <https://www.mauthor.com/doc/en/page/SCORM-packages>

Microsoft File format reference for Word, Excel, and PowerPoint. (s.d.). Obtido em fevereiro de 2022, de Microsoft: <https://docs.microsoft.com/en-us/deployoffice/compat/office-file-format-reference>

Microsoft Formatos e extensões de nomes de arquivo open xml. (s.d.). Obtido em fevereiro de 2022, de Microsoft: <https://support.microsoft.com/pt-br/office/formatos-e-extensões-de-nomes-de-arquivo-open-xml-5200d93c-3449-4380-8e11-31ef14555b18>

Moodle App. (s.d.). Obtido em fevereiro de 2022, de Moodle: <https://moodle.com/app/>

Moodle App Features. (s.d.). Obtido em fevereiro de 2022, de Moodle.org: https://docs.moodle.org/311/en/Moodle_app_features

Moodle Plugins - Media Player. (s.d.). Obtido em dezembro de 2021, de moodle.org: <https://moodle.org/plugins/browse.php?list=category&id=68>

moodle.org - Moodle SCORM Player. (s.d.). Obtido em janeiro de 2022, de moodle.org: https://docs.moodle.org/311/en/Moodle_app_SCORM_player

moodle.org - Moodle Statistics. (s.d.). Obtido em dezembro de 2021, de moodle.org: <https://stats.moodle.org>

nvie.com - A successful git branching model. (05 de janeiro de 2010). Obtido de nvie.com: <https://nvie.com/posts/a-successful-git-branching-model/>

Osterwalder, A., & Pigneur, Y. (2013). *Business Model Generation: A Handbook for Visionaries, Game Changers, and Challengers.* John Wiley & Sons. Obtido em fevereiro de 2022

Parisi, T. (2012). *WebGL: Up and Running.* O'Reilly Media, Inc. Obtido em fevereiro de 2022

Patterns - Container/Presentational Pattern. (s.d.). Obtido em abril de 2022, de Patterns: <https://www.patterns.dev/posts/presentational-container-pattern/>

PDFObject. (s.d.). Obtido em fevereiro de 2022, de PDFObject: <https://pdfobject.com>

Penfold, S. (21 de janeiro de 2022). *The Best Elearning Authoring Tools & Software*. Obtido em fevereiro de 2022, de Elucidat: <https://www.elucidat.com/blog/elearning-authoring-tools/>

Phalcon Docs - MVCC Applications. (s.d.). Obtido em janeiro de 2022, de Phalcon Docs: <https://docs.phalcon.io/3.4/en/application#mvc-applications>

Porto Editora a nossa história. (s.d.). Obtido em dezembro de 2021, de Porto Editora: <https://www.portoeditora.pt/sobre-nos/historial>

Porto Editora e LeYa dão acesso gratuito a plataformas virtuais para apoiar alunos sem aulas. (13 de março de 2020). *Meios & Publicidade*. Obtido em fevereiro de 2022, de <https://www.meiosepublicidade.pt/2020/03/porto-editora-da-acesso-gratuito-plataforma-escola-virtual/>

Queirós, L. M. (Ed.). (28 de fevereiro de 2022). Grupo LeYa vendido a empresa holandesa especializada no livro e no multimédia escolar. *Público*. Obtido em fevereiro de 2022, de <https://www.publico.pt/2022/02/28/culturaipsilon/noticia/operacoes-leya-portugal-mocambique-compradas-neerlandesa-infinitas-learning-1997112>

React - Team. (2022). Obtido em abril de 2022, de React: <https://reactjs.org/community/team.html>

React. (2022). Obtido em março de 2022, de React: <https://reactjs.org>

React Audio Player. (abril de 2021). Obtido em julho de 2022, de npm: <https://www.npmjs.com/package/react-audio-player>

React Docs - Components and props. (s.d.). Obtido em junho de 2022, de React Docs: <https://reactjs.org/docs/components-and-props.html>

React Docs - Hooks at a Glance. (s.d.). Obtido em julho de 2022, de React Docs: <https://reactjs.org/docs/hooks-overview.html>

React Docs - Introducing Hooks. (s.d.). Obtido em junho de 2022, de React Docs: <https://reactjs.org/docs/hooks-intro.html>

React Docs - Introducing JSX. (s.d.). Obtido de React Docs: <https://reactjs.org/docs/introducing-jsx.html>

React Docs - Rendering elements. (s.d.). Obtido em junho de 2022, de React Docs: <https://reactjs.org/docs/rendering-elements.html>

React Docs - State and Lifecycle. (s.d.). Obtido em julho de 2022, de React Docs: <https://reactjs.org/docs/state-and-lifecycle.html>

React Docs - Thinking in React. (s.d.). Obtido em julho de 2022, de React Docs: <https://reactjs.org/docs/thinking-in-react.html>

React H5 Audio Player. (setembro de 2022). Obtido em setembro de 2022, de npm: <https://www.npmjs.com/package/react-h5-audio-player>

React Icons. (04 de setembro de 2022). Obtido em setembro de 2022, de react-icons: <https://react-icons.github.io/react-icons>

React lifecycle methods diagram. (s.d.). Obtido em julho de 2022, de Projects: <https://projects.wojtekmaj.pl/react-lifecycle-methods-diagram/>

React PDF. (junho de 2022). Obtido em junho de 2022, de npm: <https://www.npmjs.com/package/react-pdf>

React PDF Viewer. (s.d.). Obtido em Abril de 2022, de React PDF Viewer: <https://react-pdf-viewer.dev>

React PDF Viewer - Plugins. (s.d.). Obtido em Abril de 2022, de React PDF Viewer: <https://react-pdf-viewer.dev/plugins/>

React PDF Viewer - Purchase. (s.d.). Obtido de React PDF Viewer: <https://react-pdf-viewer.dev/purchase/>

React Versions. (26 de abril de 2022). Obtido em maio de 2022, de React: <https://reactjs.org/versions/>

react-doc-viewer. (outubro de 2020). Obtido em fevereiro de 2022, de npm: <https://www.npmjs.com/package/react-doc-viewer#installation>

react-image-zoom. (novembro de 2020). Obtido em julho de 2022, de npm: <https://www.npmjs.com/package/react-image-zoom>

Redux. (s.d.). Obtido em junho de 2022, de Redux: <https://redux.js.org>

Rich, N., & Holweg, M. (janeiro de 2000). Value Analysis, Value Engineering. *INNOREGIO project*, 32. Obtido em janeiro de 2022

Roy, S. D. (julho de 2022). *What is Redux? Store, Actions, and Reducers*. Obtido de freeCodeCamp: <https://www.freecodecamp.org/news/what-is-redux-store-actions-reducers-explained/>

Runeson, P., & Höst, M. (19 de dezembro de 2008). Guidelines for conducting and reporting case study research in software engineering. (D. Sjoberg, Ed.) *Springerlink.com*. Obtido em dezembro de 2021

SCORM.com. (s.d.). Obtido em janeiro de 2022, de SCORM.com: <https://scorm.com>

SCORM.com - Evolution. (s.d.). Obtido em janeiro de 2022, de SCORM.com: <https://scorm.com/scorm-explained/business-of-scorm/scorm-versions/>

SCORM.com - Explained. (s.d.). Obtido em janeiro de 2022, de SCORM.com: <https://scorm.com/scorm-explained/one-minute-scorm-overview/>

Stack Overflow - Survey. (2020). Obtido em julho de 2022, de Stack Overflow: <https://insights.stackoverflow.com/survey/2020#most-loved-dreaded-and-wanted>

Stack Overflow - Survey. (2022). Obtido em agosto de 2022, de Stack Overflow: <https://survey.stackoverflow.co/2022/#most-loved-dreaded-and-wanted-language-love-dread>

The Software House - State of frontend. (2022). Obtido em setembro de 2022, de The Software House: <https://tsh.io/state-of-frontend/#report>

Typescript. (s.d.). Obtido em março de 2022, de Typescript: <https://www.typescriptlang.org>

Universidade do Minho - Blackboard-UMinho. (2022). Obtido em setembro de 2022, de Universidade do Minho: <https://elearning.uminho.pt>

unYLeYa O que é o e-Learning. (s.d.). Obtido em janeiro de 2022, de unYleYa: <https://unyleya.com.pt/pt/content/1-o-que-oe-learning>

Viage, H. (Ed.). (14 de dezembro de 2020). 53,6% das crianças tem dispositivo com acesso à internet em Portugal. *Forbes Portugal*. Obtido em dezembro de 2021, de Forbes Portugal: <https://www.forbespt.com/536-das-criancas-tem-dispositivo-com-acesso-a-internet-em-portugal/>

Visual Studio Marketplace - Git Graph. (abril de 2021). Obtido em maio de 2022, de Visual Studio Marketplace: <https://marketplace.visualstudio.com/items?itemName=mhutchie.git-graph>

Visual Studio Marketplace - GitLens. (setembro de 2022). Obtido em setembro de 2022, de Visual Studio Marketplace: <https://marketplace.visualstudio.com/items?itemName=eamodio.gitlens>

WebGL Getting a WebGL Implementation. (s.d.). Obtido em fevereiro de 2022, de WebGL: <https://get.webgl.org/get-a-webgl-implementation/#:~:text=WebGL%201.0%20is%20supported%20in,both%20desktop%20and%20mobile%20browsers.>

What is an Authoring Tool? (20 de dezembro de 2021). Obtido em dezembro de 2021, de Valamis: <https://www.valamis.com/hub/authoring-tool>

Woodall, T. (2003). Conceptualising 'value for the customer': An attributional, structural and dispositional analysis. *Academy of marketing science review 2003*. Obtido em janeiro de 2022

Youtube - IFrame Player API. (s.d.). Obtido em março de 2022, de Youtube: https://developers.google.com/youtube/iframe_api_reference