

Memòria projecte Q?Web

Autor: Manel Cámara Gomis

Consultor: Manel Zaera

Juny 2012

Resum del projecte Q?Web

Q?Web és un projecte de programari emmarcat dins l'àmbit de l'anomenat 'Technology Enhanced Learning (TEL)'. Aquest àrea de coneixement fa referència a l'ús de les TIC's per donar suport a qualsevol activitat d'aprenentatge.

Dins el TEL, trobem programaris anomenats '**eines d'autor**', definits com qualsevol forma de programari que permet crear o editar **objectes** d'aprenentatge. A la vegada, trobem els propis 'objectes d'aprenentatge', definits com unitats pedagògiques reutilitzables i intercanviables, compostes per dues capes: els propis continguts docents, i una estructura tecnològica que permet la interacció amb ells.

El projecte Q?Web defineix dues línies independents, però complementàries, de desenvolupament de programari:

- **implementació d'una eina d'autor, Q?WebEditor**, en forma d'aplicació d'escriptori. L'eina ha de permetre als usuaris-docents generar objectes d'aprenentatge que segueixen l'estàndard SCORM. Aquest és un estàndard obert i àmpliament utilitzat dins la indústria TEL, i garanteix que els continguts pedagògics poden ser reutilitzats i intercanviats entre diferents sistemes de gestió d'aprenentatge (Learning Management Systems, LMS), com per exemple Moodle o Sakai. L'usuari de Q?WebEditor definirà els continguts de l'objecte a crear. Aquests, una vegada integrats en l'eina, seran transformats en un objecte d'aprenentatge SCORM-compatible. D'aquesta manera, l'objecte resultant podrà ser integrat en qualsevol LMS que segueixi aquest estàndard. En quant a la definició pràctica dels objectes pedagògics, Q?WebEditor pot generar dos tipus diferenciats:
 - **qüestionaris web d'autoavaluació**: l'eina genera un paquet SCORM que conté un qüestionari d'autoavaluació. Aquest, una vegada desplegat, presentarà a l'alumne un conjunt de preguntes/respostes de forma consecutiva. Cada pregunta estarà formada per un text i, opcionalment, complementada per una imatge. L'alumne, una vegada hagi finalitzat el test, obtindrà un informe de resultats.
 - **tutorials web**: l'eina genera un paquet SCORM que conté un tutorial web, format per un conjunt de pantalles compostes per text i imatge ('slides'). El tutorial està organitzat a mode de presentació, i, una vegada desplegat, permetrà visualitzar el conjunt d'slides, ja sigui seqüencialment, ja sigui de forma aleatòria, mitjançant un menú permanent d'accessos directes.
- **implementació d'una aplicació web, Q?WebScormApp**, que permet als usuaris-estudiants interactuar amb els objectes d'aprenentatge generats mitjançant Q?WebEditor. Així, l'objecte d'aprenentatge estarà format, de forma autocontinguda, per un conjunt de continguts i una aplicació web que permet interactuar amb ells. Per tal que els objectes pedagògics tinguin un àmbit d'utilització el més ampli possible, l'aplicació compleix amb 2 requisits:
 - l'objecte pedagògic és una aplicació web vàlida, i, per tant, pot ser visualitzada en un entorn online independent de qualsevol sistema de

gestió d'aprenentatge.

- l'objecte pedagògic segueix l'estàndard SCORM, i, per tant, pot ser integrat en qualsevol LMS compatible amb aquest estàndard.

Tots dos programaris, Q?WebEditor i Q?WebScormApp són software lliure, i es fan públics al repositori GitHub sota llicència GPL (veure [annexos](#))

Índex

1	Introducció	2
2	Justificació del TFC i context en el qual es desenvolupa: punt de partida i aportació del TFC	2
3	Objectius del TFC	3
4	Enfocament i mètode seguit	4
5	Planificació del projecte	6
6	Productes obtinguts	8
6.1	Programari Q?WebEditor	8
6.2	Programari Q?WebScormApp	9
6.3	Memòria del projecte	9
7	Arquitectura de les aplicacions i tecnologies emprades en el desenvolupament	9
7.1	Arquitectura de les aplicacions	9
7.2	Eines de programari emprades en el procés de desenvolupament	10
7.3	Tecnologies emprades en el procés de desenvolupament	11
8	Futures evolucions de Q?Web	13
9	Conclusions	13
10	Glossari	14
11	Bibliografia	15
12	Annexos	17
12.1	Tutorial Editor Q?Web (generació de qüestionaris d'autoavaluació)	18
12.2	Tutorial Editor Q?Web (generació de tutorials web)	26
12.3	Exemple de qüestionari d'autoavaluació	32
12.4	Exemple de fitxer xml de continguts	33
12.5	GitHub	34

1 Introducció

Dins els estudis d'Enginyeria Tècnica en Informàtica de Gestió, el Treball Final de Carrera (TFC) es planteja com una assignatura a cursar quan s'estan culminant aquests estudis, i està focalitzat en la realització d'un treball de síntesi dels coneixements adquirits al llarg de la carrera. Aquest treball ha de ser eminentment pràctic i ha d'estar vinculat amb algun camp relacionat amb el que seria l'exercici professional de la informàtica. Abans d'iniciar el TFC, l'estudiant és assignat a un àrea de coneixements determinada, dins la qual haurà de desenvolupar el seu treball.

En el cas d'aquest TFC, el projecte desenvolupat s'emmarca en l'àrea de Technology Enhanced Learning (TEL). Al llarg d'aquest document de memòria, es detalla de forma exhaustiva com s'ha gestionat el projecte informàtic Q?Web, des de la definició inicial fins els resultats aconseguits.

2 Justificació del TFC i context en el qual es desenvolupa: punt de partida i aportació del TFC.

Com hem vist al punt anterior, aquest TFC, en el seu punt de partida, ha estat assignat a l'àrea TEL. Essent així, per tal de delimitar aquest àmbit de coneixements i poder fer una proposta de programari adient, s'ha realitzat un estudi exploratori previ des de dos punts de vista:

- eines d'autor existents al mercat: repassant diferents aplicacions existents al mercat, que permeten als docents generar objectes d'aprenentatge, hem tingut la oportunitat d'experimentar amb l'aplicació Quandary, un programari de codi obert que permet crear 'laberints d'acció' (action mazes) en format Scorm. Aquests objectes poden prendre diferents formes: enquestes, qüestionaris, tutorials, històries interactives,... Quandary ha proporcionat la inspiració inicial per a plantejar el projecte Q?Web.
- estàndards de la indústria: l'estàndard SCORM (Sharable Content Object Reference Model) és considerat l'estàndard 'de facto' de la indústria. Aquest estàndard permet crear objectes pedagògics reutilitzables i compartibles entre qualsevol sistema LMS (Learning Management System) com els coneguts Moodle o Sakai.

Aquest projecte, inspirant-se en Quandary, i respectant l'estàndard Scorm, té com a finalitat el desenvolupament, des de zero, d'una aplicació d'escriptori que permeti a un docent, de forma fàcil i senzilla, crear objectes d'aprenentatge.

A la pràctica, aquesta eina d'autor té les següents característiques:

- nom: Q?WebEditor
- funcionalitat principal: l'eina ha de permetre als usuaris-docents generar objectes

d'aprenentatge que segueixen l'estàndard SCORM. Aquest és un estàndard obert i àmpliament utilitzat dins la indústria TEL, i garanteix que els continguts pedagògics poden ser reutilitzats i intercanviats entre diferents sistemes de gestió d'aprenentatge (Learning Management Systems, LMS), com per exemple Moodle o Sakai. L'usuari de Q?WebEditor definirà els continguts de l'objecte a crear. Aquests, una vegada integrats en l'eina, seran transformats en un objecte d'aprenentatge SCORM-compatible. D'aquesta manera, l'objecte resultant podrà ser integrat en qualsevol LMS que segueixi aquest estàndard.

- tipologia dels objectes generats: en quant a la definició pràctica dels objectes pedagògics, Q?WebEditor ha de poder generar dos tipus diferenciats:
 - qüestionaris web d'autoavaluació: l'eina genera un paquet SCORM que conté un qüestionari d'autoavaluació. Aquest, una vegada desplegat, presenta a l'alumne un conjunt de preguntes/respostes de forma consecutiva. Cada pregunta està formada per un text i, opcionalment, complementada per una imatge. L'alumne, una vegada hagi finalitzat el test, obté un informe de resultats.
 - tutorials web: l'eina genera un paquet SCORM que conté un tutorial web, format per un conjunt de pantalles composades per text i imatge ('slides'). El tutorial està organitzat a mode de presentació, i, una vegada desplegat, permet visualitzar el conjunt d'slides, ja sigui seqüencialment, ja sigui de forma aleatòria, mitjançant un menú permanent d'accessos directes.
- llicència del programari resultant: el software resultant és programari lliure, i es publica a GitHub sota una llicència General Public License (GPL) (veure [annexos](#)). Aquesta llicència permet el lliure ús i distribució del programari, però obliga a distribuir-lo en les mateixes condicions GPL.

3 Objectius del TFC

Aquest projecte s'ha desenvolupat amb la finalitat d'acomplir els següents objectius:

- Objectius acadèmics de l'autor:
 - obtenir una nota satisfactòria al TFC, com a fita necessària per tal de finalitzar els estudis d'Enginyeria Tècnica Informàtica en la UOC
- Objectius personals de l'autor:
 - el projecte ha de servir per enfortir els coneixements sobre programació Java
 - el projecte ha de servir per enfortir els coneixements sobre aplicacions web dinàmiques
 - el projecte ha de servir per explorar, des d'un punt de vista tecnològic, un nou àrea d'interès, el Technology Enhanced Learning
 - el projecte ha de servir per guanyar experiència a l'hora d'enfrontar-se al

desenvolupament de programaris que resolguin problemes no trivials

- el projecte ha de servir per iniciar-se en el desenvolupament de programari de codi obert, al respecte d'utilitzar alguna de les llicències de programari lliure existents i publicant el codi font en algun repositori públic
- Objectius pràctics:
 - dissenyar i implementar una aplicació informàtica de 'codi obert' plenament funcional que, emmarcada en l'àrea TEL, permeti a un docent crear objectes d'aprenentatge en forma d'aplicació web compatible amb l'estàndard Scorm. Aquesta aplicació ha de seguir els següents criteris principals:
 - ha de desenvolupar-se utilitzant només eines i tecnologies de 'codi obert'
 - aplicació multiplataforma: ha de ser executable independentment del sistema operatiu de la màquina de l'usuari
 - aplicació internacionalitzada i que inclogui, d'inici, els idiomes castellà i català
 - aplicació autoexecutable, sense necessitat d'instal·lació en el S.O. de l'usuari
 - incorporar tutorials d'ús en la pròpia aplicació

4 Enfocament i mètode seguit

Aquest projecte s'ha desenvolupat, conceptualment, seguint els criteris que marca la 'metodologia àgil' i, en concret, la metodologia 'Scrum'.

El desenvolupament de programari 'àgil' és un corrent alternatiu al desenvolupament tradicional, caracteritzat per ser fortament normatiu i seguir una estricta planificació. En la taula següent es comparen les característiques principals d'ambdues aproximacions:

Comparació metodologies 'àgil' vs 'tradicional'	
se centra en els individus i la seva interacció	se centra en els processos i les eines
valora el programari que funcioni	valora la documentació exhaustiva
estableix una relació permanent amb el client	es guia per les especificacions del contracte amb el client
està obert al canvi	segueix de forma estricta allò planificat

Dins la metodologia 'àgil', es troba 'Scrum', un marc de treball per al desenvolupament de productes (de programari o altre tipus) caracteritzat per ser un procés iteratiu, que, a mida que avança, va produint entregues incrementals.

Scrum està caracteritzat per:

- conjunt de rols:
 - client
 - propietari del producte
 - scrum master
 - equip de desenvolupament

- cicle de treball:
 - product backlog: llista de requeriments, provinent del procés de presa de requisits amb el client. Cada requeriment es redacta com una 'història d'usuari'. El conjunt d'històries s'ordena per prioritats.
 - històries d'usuari: cada història d'usuari representa un bloc de treball. Té un redactat curt i senzill, però a la vegada representatiu de la funcionalitat que representa.
 - sprints: cada una de les iteracions al llarg del desenvolupament. Un sprint té com a objectiu finalitzar un grup de blocs de treball (un grup d'històries d'usuari) i produeix una entrega, que suposa un increment de funcionalitat del producte respecte a l'sprint anterior.

Aplicació pràctica de Scrum en aquest projecte

Aquest projecte, en ser un TFC amb uns requeriments i una planificació predefinita, no respon, estrictament parlant, a un projecte fruit del contracte amb un client. Així, tot i que al llarg del desenvolupament s'han tingut presents els conceptes de metodologia àgil i el marc de treball Scrum, el procés s'ha adaptat al marc de treball definit pel propi TFC. En concret, l'adaptació ha seguit les següents premisses:

- rols: el client d'aquest projecte ha estat el Director del TFC. L'equip de treball ha estat un únic desenvolupador, l'estudiant. En terminologia Scrum, l'estudiant ha estat, a la vegada, 'propietari del projecte', 'scrum master' i 'equip de desenvolupament'.
- presa de requisits del client (product backlog): la tradicional presa de requisits al client, que és qui defineix el producte a desenvolupar en un entorn client-proveïdor, ha estat substituïda per una proposta de projecte realitzada per l'estudiant, que ha comptat amb l'aprovació del Director del TFC (client)
- sprints (fases del projecte o iteracions): les fases i terminis del projecte venien ja prefixades en la pròpia definició del TFC. Al llarg del desenvolupament s'han respectat els terminis marcats per a les entregues, i s'ha consensuat amb el client, quan ha calgut, alguna extensió de dates.
- relació amb el client: la comunicació amb el client al llarg del projecte ha estat asíncrona. A més a més de les entregues definides per a cada iteració, s'han redactat i proporcionat al client informes setmanals de seguiment que han permès:

- tenir al client assabentat de l'evolució del projecte
- informar al client de canvis i/o novetats respecte a les funcionalitats del producte o respecte a l'ordre de prioritats dels diferents blocs de treball

Com a exemple, les històries d'usuari principals que participen en aquest projecte serien:

Com a	vull	per a
usuari	tenir una opció 'nou qüestionari'	generar objectes d'aprenentatge en forma de qüestionaris
usuari	tenir una opció 'editar qüestionari'	editar qüestionaris prèviament creats
usuari	tenir una opció 'tutorial'	saber com utilitzar l'editor Q?WebEditor

La metodologia seguida, conjuntament amb el marc de treball predefinit al TFC, s'ha traduït en la planificació del projecte que es detalla en el següent punt d'aquest document.

5 Planificació del projecte

El projecte es planteja en 4 fases (iteracions o sprints en terminologia Scrum):

Planificació Fase 0: definició de tasques, temporalització i lliurables

Blocs de treball:

- recerca i estudi preliminar que ha de portar a elaborar un document de definició d'objectius i disseny d'un pla de treball en relació al projecte Q?Web

Lliurables:

- Document proposta de projecte i pla de treball

Temporalització:

- 2 setmanes

Planificació Fase 1: definició de tasques, temporalització i lliurables

Blocs de treball:

- definició de les funcionalitats que ha d'incloure el prototipus de programari a

lliurar en aquesta fase

- construcció del prototipus d'interfície gràfica d'usuari: pantalla menú general, pantalla d'alta de qüestionari, pantalla d'edició de qüestionari i sortida de l'aplicació
- implementació Java de les funcionalitats definides: aquesta implementació ha de generar (o permetre editar) un document xml que contindrà el conjunt de preguntes/ respostes d'un determinat qüestionari
- implementació AJAX (html+css+Javascript) d'un prototipus funcional d'aplicació web encarregat de presentar el qüestionari generat: en aquesta fase el prototipus web farà ús del fitxer de continguts xml de forma directa . Es deixa per a la següent fase la integració del fitxer xml de continguts en un paquet SCORM-compatible.

Lliurables:

- Informe d'evolució del projecte
- Esquema inicial de la memòria del projecte
- Planificació de la Fase 2: definició de tasques, temporalització i lliurables
- Prototipus funcional del programari Q?Web: tant de l'eina d'autor Q?WebEditor (Java) com de l'eina de visualització de qüestionaris web Q?WebScormPlayer.

Temporalització:

- 4 setmanes

Planificació Fase 2: definició de tasques, temporalització i lliurables

Blocs de treball:

- passar de treballar amb fitxers .xml a treballar amb fitxers .zip scorm-compatible
- implementar versió definitiva de la GUI de Q?WebEditor i de la seva lògica de negoci
- implementar versió definitiva de la GUI de aplicació web Q?WebScormPlayer per a visualitzar objectes d'aprenentatge
- realitzar un prototipus del tutorial de l'editor Q?Web
- redacció de l'esborrany de la memòria del projecte.

Lliurables:

- versió beta del programari Q?Web (Q?WebEditor & Q?WebScormPlayer)
- codi font de la versió beta del programari
- fitxers scorm d'exemple (un qüestionari d'autoavaluació i un tutorial)
- esborrany de la memòria del projecte

Temporalització:

- 5 setmanes

Planificació Fase 3 (final): definició de tasques, temporalització i lliurables.

Blocs de treball:

- transformar la versió Beta de l'editor+webApp en **versió RC** (Release Candidate).

Subtasques:

- realitzar procés de testing per tal de detectar i solucionar el major nombre de 'bugs' en les aplicacions
 - revisar i/o programar la lògica de les restriccions principals que ha de validar l'editor Q?Web. Per exemple, i en relació als qüestionaris d'autoavaluació, poden ser: validar qüestionaris buits, validar qüestionaris amb format incorrecte, validar preguntes sense respostes, validar preguntes amb un nombre de respostes vàlides diferents a 1,...
 - comentar tot el codi java
 - generar els Javadocs del codi java
- generar els **tutorials definitius** per als dos tipus d'objectes que genera l'editor (cadascun d'ells amb els dos idiomes actualment suportats per l'aplicació)
 - redactar la **versió definitiva de memòria** del projecte.

Lliurables:

- executable del programari final
- codi font del programari
- Javadocs del programari
- url's d'accés al repositori GitHub (veure [annexos](#)) per als dos projectes (Editor i ScormApp)
- fitxers scorm d'exemple
- memòria del projecte

6 Productes obtinguts

6.1 Programari Q?WebEditor

Aplicació d'escriptori en format zip que inclou:

- executable del programari Q?WebEditor.jar
- una còpia de l'aplicació web Q?WebScormApp

- codi font Java del programari, en format de projecte Eclipse. A la vegada, el codi font es publica a GitHub sota llicència GPL (veure [annexos](#))
- documentació del codi font, en format Javadocs
- tutorials del programari: en els dos idiomes suportats i per a cada un dels dos objectes d'aprenentatge que es poden generar
- exemples d'objectes pedagògics: CuestionarioHiragana.zip i TutorialQ?WebEditor.zip

6.2 Programari Q?WebScormApp

Aplicació web que inclou:

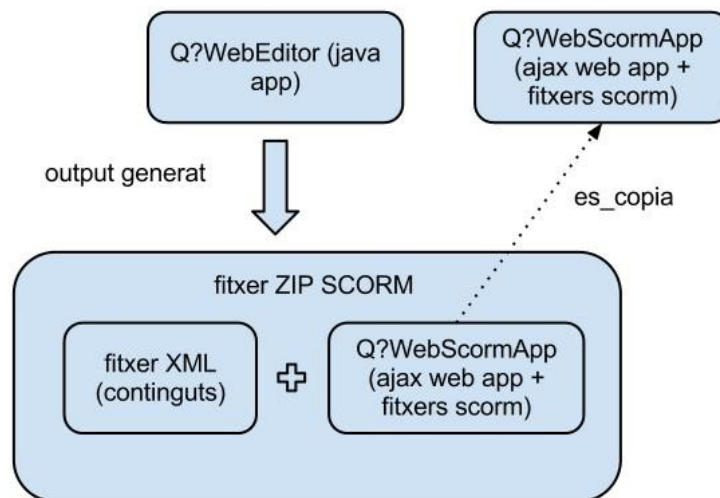
- directori 'war' amb l'aplicació web, formada per fitxers html, css i Javascript.
- codi font Java del programari, en format de projecte Eclipse. A la vegada, el codi font es publica a GitHub sota llicència GPL (veure annexos)

6.3 Memòria del projecte

Aquest mateix document, que recull de forma exhaustiva com s'ha desenvolupat el projecte Q?Web.

7 Arquitectura de les aplicacions i tecnologies emprades en el desenvolupament

7.1 Arquitectura de les aplicacions



El programari Q?Web compta amb dues aplicacions independents que treballen de forma conjunta per a generar l'objecte d'aprenentatge:

- Q?WebEditor: és una aplicació d'escriptori Java que, en essència, s'encarrega de generar un fitxer xml amb els continguts que el docent-usuari de l'aplicació ha introduït
- Q?WebScormApp: és una aplicació web Ajax que, quan és desplegada en un entorn web, presenta de forma interactiva els continguts que llegeix d'un fitxer xml generat amb Q?WebEditor. Aquesta aplicació inclou els fitxers de configuració necessaris per a fer l'aplicació scorm-compatible

Per tal d'obtenir un objecte d'aprenentatge Scorm amb Q?WebEditor, aquesta eina, en la seva estructura de directoris, compta amb una còpia de l'aplicació web Q?WebScormApp. Quan l'usuari de l'editor ha finalitzat la introducció de continguts, i demana a Q?WebEditor la generació de l'objecte, l'eina fa els següents passos, transparents per a l'usuari:

1. generar un fitxer xml de continguts en un directori temporal
2. fer una còpia de l'aplicació Q?WebScormApp al mateix directori temporal del punt anterior
3. 'empaquetar' tot el contingut del directori temporal anterior en un fitxer zip, que és l'output de l'aplicació

7.2 Eines de programari emprades en el procés de desenvolupament

- Eines de desenvolupament:
 - Eclipse 3.7 Indigo (www.eclipse.org): aquesta ha estat la IDE open source de desenvolupament de les dues aplicacions, Q?WebEditor i Q?

WebScormApp

- GPE (Google Plugin for Eclipse, <https://developers.google.com/eclipse/>) plugin open source que s'integra amb Eclipse IDE i permet desenvolupar aplicacions web GWT (Google Web Toolkit)
- Netbeans 7.1 (www.netbeans.org): aquesta IDE open source s'ha utilitzat per a construir la GUI de Q?WebEditor
- Eines de validació d'objectes generats:
 - Reload Scorm Player (<http://www.reload.ac.uk/scormplayer.html>): eina open source que permet validar i visualitzar paquets scorm. S'ha utilitzat al llarg del desenvolupament per anar validant i interactuant amb els objectes d'aprenentatge generats en les proves
 - Demo Moodle (<http://demo.moodle.net/>): LMS open source que permet, utilitzant un compte demo, validar i interactuar amb objectes d'aprenentatge Scorm en un entorn de treball web real (veure annexos).
- Eines accessòries:
 - LibreOffice (<http://es.libreoffice.org/>): paquet d'ofimàtica open source que s'ha utilitzat per generar els diferents documents relacionats amb el projecte i, en concret, aquest mateix document de memòria.
 - Screen Hunter 5.1 Free (http://wisdom-soft.com/products/screenhunter_free.htm): eina freeware que permet fer captures de pantalla

7.3 *Tecnologies emprades en el procés de desenvolupament*

- Java 1.6 SE:

Q?WebEditor s'ha implementat amb llenguatge Java. Aquest fet ha aportat dues avantatges importants al projecte:

- programari multiplataforma: Java és un llenguatge 'portable', fent que les aplicacions Java siguin 'multiplataforma'. Així, Q?WebEditor pot ser directament executat en tot tipus de maquinari, independentment del sistema operatiu que utilitzi (Windows, Mac OSX, Linux,...)
- programari autoexecutable: Java permet 'empaquetar' una aplicació en un fitxer .jar executable. Q?WebEditor aprofita aquesta característica i és executat sense necessitat de fer cap instal·lació. Així, el programari és d'ús immediat i no afecta ni es veu afectat per les característiques del sistema operatiu de l'usuari.
- Java package org.apache.catalina.tomcat: a la versió 'Standard Edition' de Java s'ha afegit aquest package desenvolupat per l'organització Apache. D'aquesta

manera, el programari Q?WebEditor pot llençar instàncies del servlet container Tomcat des del servidor local de l'usuari. Aquesta característica s'ha utilitzar per implementar l'opció d'ajuda del programari, que és un tutorial web creat amb el propi Q?WebEditor.

- Google Web Toolkit (<https://developers.google.com/web-toolkit/>):

GWT és un conjunt d'eines de desenvolupament open source desenvolupat per Google. Permet implementar aplicacions web AJAX (html+css+javascript) utilitzant el llenguatge de programació Java i sense necessitat d'utilitzar el llenguatge Javascript. El propi Toolkit transforma, de forma transparent per al desenvolupador, el codi Java original en el codi Javascript de l'aplicació web. L'ús de GWT ha aportat al projecte els següents avantatges:

- estalvi de la corba d'aprenentatge que hagués suposat la programació en llenguatge Javascript
 - reutilització directa de classes 'bean' Java de Q?WebEditor en Q?WebScormApp: Cuestionario.java, Pregunta.java, Respuesta.java
 - reutilització, amb petites adaptacions, de les utilitats java per al treball amb XML de Q?WebEditor en Q?WebScormApp.
- Scorm (Sharable Content Object Reference Model): Scorm no és en sí mateix una tecnologia, sinó un conjunt d'especificacions, convertides en un estàndard de la indústria, que permet encapsular objectes d'aprenentatge de forma que siguin 'portables' entre qualsevol sistema LMS compatible amb l'estàndard. En essència, un paquet Scorm és un fitxer comprimit zip que consta de 3 elements:
 - el propi objecte d'aprenentatge: en el cas de Q?Web, el formaria el fitxer xml de continguts ([veure annexos](#)) i l'aplicació web Q?WebScormApp.
 - un 'manifest': en forma de fitxer xml ([veure annexos](#)), que conté 'metadades' que el sistema LMS necessita per saber interpretar l'objecte pedagògic
 - un conjunt de fulls d'estil en forma de fitxers xsd: que complementen el fitxer 'manifest' per tal que l'entorn virtual d'aprenentatge interpreti l'objecte.
- XML (*eXtensible Markup Language*): llenguatge de marques desenvolupat per W3C, amb la característica de ser, a la vegada, human-readable i machine-readable. XML ha estat triat com el vehicle d'intercanvi d'informació entre Q?WebEditor i Q?WebScormApp i el seu ús, en aquest projecte, ha permès:
 - separar totalment el desenvolupament dels dos programaris, fet que ha suposat poder treballar en tots dos de forma paral·lela
 - la característica 'human-readable' d'aquest llenguatge ha permès al desenvolupador configurar, de forma manual i quan ha calgut, els fitxers xml de continguts, facilitant i simplificant els passos intermedis de desenvolupament i les proves relacionades.

8 Futures evolucions de Q?Web

Partint de la base tecnològica desenvolupada al projecte Q?Web, es plantegen tot seguit algunes possibles línies d'evolució futura:

- versió web de l'editor Q?WebEditor: aprofitant l'experiència aconseguida amb el desenvolupament de Q?WebScormApp amb GWT, es planteja la possibilitat de replicar l'editor en una versió web
- aprofitar la funcionalitat d'instanciar el servlet container Tomcat, que implementa Q?WebEditor, per tal d'oferir a l'usuari la possibilitat de visualitzar els objectes d'aprenentatge a mida que es van desenvolupant, en un entorn web real. Això eliminaria la necessitat d'utilitzar eines externes, com Reload Scorm Player, per a validar els objectes.
- aprofitar que Q?WebEditor és una aplicació que implementa i18n (internacionalització) per ampliar els idiomes que la versió actual suporta amb l'anglès.

9 Conclusions

Aquest projecte culmina, en data 17/06/2012, amb l'entrega dels lliurables que s'havien definit en el seu inici. Com a autor del mateix, entenc que el projecte ha assolit els objectius marcats en la seva definició.

Per una banda, des d'un punt de vista personal, he pogut:

- enfortir i avançar en els meus coneixements previs de programació orientada a objectes Java
- superar la corba d'aprenentatge que m'ha suposat generar una aplicació web GWT no trivial
- posar en pràctica els dos punts anteriors dins un àmbit d'interès, TEL, en el qual voldria seguir experimentant en el futur

Per una altra banda, des d'un punt de vista pràctic, els resultats tangibles del projecte entenc que responen fidelment a allò que s'havia establert en la seva definició inicial:

- el programari final és plenament funcional: tot i que allargant el període de testing segur que apareixeran alguns 'bugs', el programari és prou estable per a treballar correctament amb les seves funcionalitats principals. Així, un docent pot utilitzar-lo, en condicions reals de treball, per generar objectes d'aprenentatge vàlids segons l'estàndard Scorm.
- el programari creat és software lliure, i s'ha desenvolupat utilitzant només eines de programari lliure. El producte final, tant en referència al seu ús com al seu codi font, s'ha obert al domini públic, sota llicència GPL, aprofitant el repositori GitHub.

10 Glossari

- **aplicació d'escriptori:** programari executable en un ordinador tipus PC
- **Ajax:** conjunt de tecnologies que permeten fer aplicacions web dinàmiques. Inclou pàgines amb llenguatge HTML, fulls d'estil Cascade Style Sheets (CSS) i fitxers Javascript
- **aplicació Java:** programari desenvolupat en llenguatge de programació Java
- **aplicació web:** programari que necessita d'un servidor web per a ser executat. L'execució pot ser local (en el propi ordinador de l'usuari) o remota, ja sigui en una Intranet o a Internet
- **beta (versió):** al llarg del desenvolupament de programari, el producte obtingut es va etiquetant, incrementalment, amb una versió. 'Beta' és una versió de programari propera a la definitiva en funcionalitats però no prou estable per a ser distribuïda
- **bugs:** errors o incidències que es detecten en versions no estables de programari
- **CSS:** tecnologia que permet separar, en fulls específics de definició, els estils (estètica) que es volen aplicar a una aplicació web
- **Demo:** s'anomena demo a una versió de prova d'un programari. Aquest tipus de versió acostumen a ser de lliure accés i tenen funcionalitats limitades.
- **Eclipse:** entorn de desenvolupament de programari (IDE) open source
- **eina d'autor:** en TEL, programari que permet generar objectes d'aprenentatge
- **estàndard SCORM:** estàndard de la indústria TEL que defineix les especificacions que han de complir els objectes d'aprenentatge per a fer-los portables entre diferents entorns d'aprenentatge.
- **GitHub:** repositori públic de projectes de desenvolupament de programari
- **Google Web Toolkit (GWT)**
- **jar (executable):** tipus de fitxer comprimit de Java que permet contenir una aplicació executable
- **Java:** llenguatge de programació open source i multiplataforma
- **Java package:** conjunt de classes Java
- **Javadoc:** utilitat que permet generar la documentació d'un conjunt de classes Java, API, en format HTML
- **Javascript:** llenguatge de programació interpretat que permet donar dinamisme a les pàgines web HTML
- **LibreOffice:** paquet d'ofimàtica de programari lliure
- **LMS (Learning Management System):** entorn virtual d'aprenentatge

- **Moodle:** uns dels LMS lliures més estesos al món
- **Netbeans:** entorn de desenvolupament de programari, IDE, open source
- **objecte d'aprenentatge:** en TEL, un objectes pedagògic format per uns continguts de coneixement i de pràctica
- **plugin:** mòdul de programari que es pot incorporar a un programari existent per tal d'incloure una nova funcionalitat
- **programari lliure:** programari de codi obert i lliure ús
- **servidor web:** software instal·lat en un maquinari que s'encarrega de rebre peticions de navegadors per a servir-los pàgines web
- **Release Candidate (versió):** versió d'un nou programari que, tot i trobar-se en fase de proves, és complerta a nivell de funcionalitats i estable. Correspon a l'estadi previ a la versió definitiva d'un programari.
- **Sakai:** juntament amb Moodle, un dels LMS lliures més estesos
- **Scrum:** model d'aplicació de 'programació àgil'
- **Tomcat:** servidor web que conté les especificacions per a donar servei a pàgines web programades en Java (Java Server Pages) i executar objectes Java (servlets)
- **war (estructura de directoris):** en un entorn Java, war és l'estructura estàndard de directoris que segueixen les aplicacions web
- **W3C:** World Wide Web Consortium és un consorci internacional que produeix recomanacions per a la World Wide Web.
- **zip:** el format de compressió més utilitzat per a comprimir arxius informàtics

11 Bibliografia

TFC-TEL UOC ETIG: http://cv.uoc.edu/tren/trenacc/web/GAT_EXP.PLANDOCENTE?any_academico=20121&cod_asignatura=05.202&idioma=CAT&pagina=PD_PREV_SECRE&cache=S

Technology Enhanced Learning (TEL): http://en.wikipedia.org/wiki/Technology-Enhanced_Learning

Scrum: [http://en.wikipedia.org/wiki/Scrum_\(development\)](http://en.wikipedia.org/wiki/Scrum_(development))

Scrum: <http://www.slideshare.net/FlowersInSpace/introduccion-a-scrum-con-caso-prctico-1516220>

Scrum: <http://www.genbetadev.com/metodologias-de-programacion/historias-de-usuario-una-forma-natural-de-analisis-funcional>

Eines d'autor (authoring tools): http://en.wikipedia.org/wiki/Authoring_system

Objectes d'aprenentatge (learning objects):

http://en.wikipedia.org/wiki/Learning_object

Entorns virtuals d'aprenentatge (Learning Management Systems, LMS):

http://en.wikipedia.org/wiki/Learning_Management_System

Moodle: <http://moodle.org/>

Quandary: <http://www.halfbakedsoftware.com/quandary.php>

Reload Scorm Player: <http://www.reload.ac.uk/scormplayer.html>

Agile programming: http://en.wikipedia.org/wiki/Agile_software_development

12 Annexos

12.1 Tutorial Editor Q?Web (generació de qüestionaris d'autoavaluació)	18
12.2 Tutorial Editor Q?Web (generació de tutorials web)	26
12.3 Exemple de qüestionari d'autoavaluació	32
12.4 Exemple de fitxer xml de continguts	33
12.5 <u>GitHub</u>	34

12.1 Tutorial Editor Q?Web (generació de qüestionaris d'autoavaluació)

Tutorial: Fer o editar qüestionaris d'autoavaluació amb QWebEditor

Fer o editar qüestionaris d'autoavaluació amb QWebEditor

Triar idioma i veure
presentació de
l'editor

Menú principal:
crear nou
qüestionari o editar
un existent

Configurar
qüestionari

Editar qüestionari

Editar pregunta

Editar resposta

Exemple
qüestionari resultat

Aquest tutorial explica com fer o editar qüestionaris d'autoavaluació amb Q?WebEditor

Continuar

Tutorial: Fer o editar qüestionaris d'autoavaluació amb QWebEditor

Triar idioma i veure presentació de l'editor

Triar idioma i veure presentació de l'editor		
Menú principal: crear nou qüestionari o editar un existent		
Configurar qüestionari		
Editar qüestionari		
Editar pregunta		
Editar resposta		
Exemple qüestionari resultat		

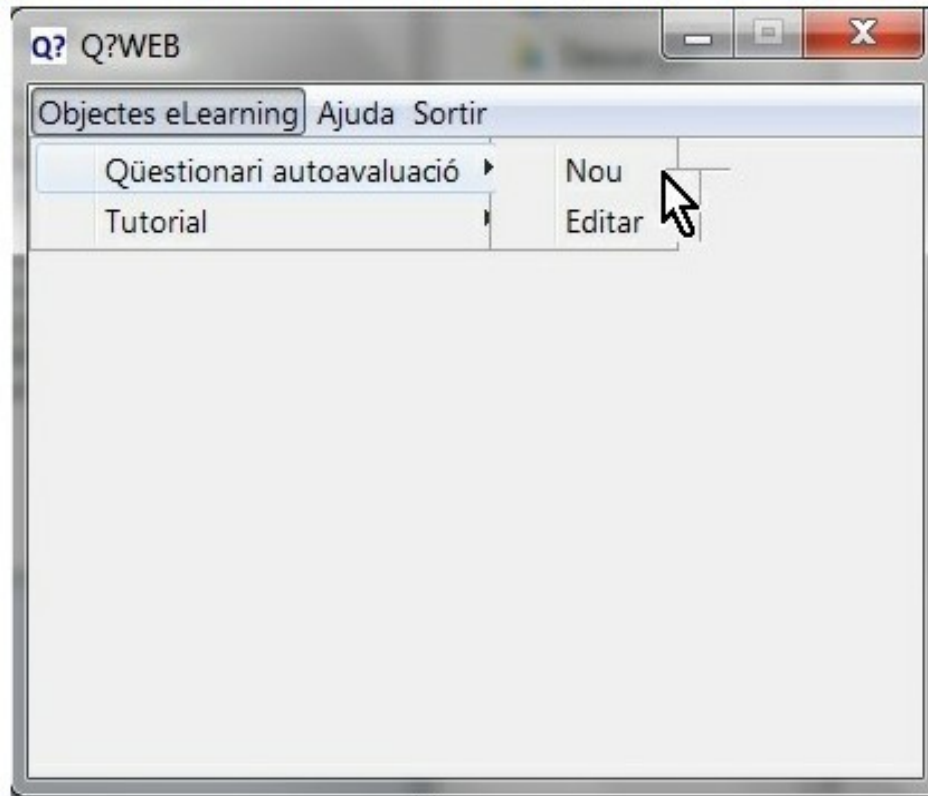
En executar Q?WebEditor, l'aplicació sol·licita que es triï l'idioma de l'editor. Una vegada fet, l'editor presenta una pantalla informativa amb dades sobre l'autor i la llicència de distribució del programari. En acceptar, l'aplicació presenta el menú principal.

Continuar

Tutorial: Fer o editar qüestionaris d'autoavaluació amb QWebEditor

Menú principal: crear nou qüestionari o editar un existent

- Triar idioma i veure presentació de l'editor
- Menú principal: crear nou qüestionari o editar un existent
- Configurar qüestionari
- Editar qüestionari
- Editar pregunta
- Editar resposta
- Exemple qüestionari resultat



Al menú principal es pot triar si es vol fer un qüestionari d'autoavaluació nou o editar un existent. En el primer cas, l'editor condueix a la pantalla de configuració d'un nou qüestionari (veure 'Configurar qüestionari'). En el segon cas, l'editor permet navegar per l'estructura d'arxius locals i localitzar el paquet scorm a editar (fitxer zip). Si el fitxer és vàlid, l'editor condueix a la pantalla 'Editar qüestionari', que ja té carregat en memòria el qüestionari font.

Continuar

Tutorial: Fer o editar qüestionaris d'autoavaluació amb QWebEditor

Configurar qüestionari

- Triar idioma i veure presentació de l'editor
- Menú principal: crear nou qüestionari o editar un existent
- Configurar qüestionari
- Editar qüestionari
- Editar pregunta
- Editar resposta
- Exemple qüestionari resultat

Q? Introducció de dades X

Títol qüestionari autoavaluació?

Text de presentació:

Idioma:

S'arriba a aquesta pantalla si l'usuari ha triat l'opció de fer un nou qüestionari. En aquesta, l'usuari pot especificar les dades bàsiques del nou qüestionari: títol, text de presentació i idioma.

Tutorial: Fer o editar qüestionaris d'autoavaluació amb QWebEditor

Editar qüestionari

- Triar idioma i veure presentació de l'editor
- Menú principal: crear nou qüestionari o editar un existent
- Configurar qüestionari
- Editar qüestionari
- Editar pregunta
- Editar resposta
- Exemple qüestionari resultat

Q? Q?Web - Edició de qüestionari X

Títol qüestionari: Ajuda

Idioma:

Text de presentació:

Preguntes: ▼

Visualitzador qüestionari:

- autoevaluacion
 - Autoavaluació Hiragana
 - Preguntes:
 - Com es pronuncia aquesta sil·laba Hiragana?
 - no.jpg
 - Com es pronuncia aquesta sil·laba Hiragana?
 - ka
 - true
 - ba
 - false
 - ka.jpg

És aquesta la pantalla principal en l'edició d'un qüestionari, ja sigui nou o existent. Permet:

- afegir una nova pregunta
- editar una pregunta existent (triant-la del desplegable 'preguntes')
- desar el qüestionari (sempre que tingui un mínim d'una pregunta amb al menys una resposta correcta)
- accedir a l'opció d'ajuda (obre ajuda en navegador web de l'usuari)
- sortir sense desar el qüestionari

La part inferior de la pantalla conté un visualitzador que permet monitoritzar l'estat i els canvis del qüestionari al llarg del procés d'edició.

Tutorial: Fer o editar qüestionaris d'autoavaluació amb QWebEditor

Editar pregunta

- Triar idioma i veure presentació de l'editor
- Menú principal: crear nou qüestionari o editar un existent
- Configurar qüestionari
- Editar qüestionari
- Editar pregunta
- Editar resposta
- Exemple qüestionari resultat

Q? Q?Web - Edició de pregunta Ajuda

Títol qüestionari:

Pregunta:

Imatge:

Respostes:

Visualitzador qüestionari:

- autoevaluacion
 - Autoavaluació Hiragana
 - Preguntes:
 - Com es pronuncia aquesta síl·laba Hiragana?
 - no.jpg
 - Com es pronuncia aquesta síl·laba Hiragana?
 - ka
 - true
 - ba
 - false
 - ka.jpg

S'arriba a aquesta pantalla des de 'Editar qüestionari'. Apareixerà buida en cas de nova pregunta, i amb dades carregades en cas d'edició de pregunta existent. Una pregunta es compon de: text, imatge (opcional) i conjunt de respostes. Les opcions disponibles són:

- inserir o modificar text
- triar o eliminar imatge (gif o jpg)
- desar la pregunta
- eliminar la pregunta
- afegir o editar resposta
- accedir a l'ajuda (s'obre al navegador web de l'usuari)
- sortir sense desar la pregunta (l'aplicació torna a 'Editar qüestionari')

La part inferior de la pantalla conté un visualitzador que permet monitoritzar l'estat i els canvis del qüestionari al llarg del procés d'edició.

Continuar

Tutorial: Fer o editar qüestionaris d'autoavaluació amb QWebEditor

Editar resposta

- Triar idioma i veure presentació de l'editor
- Menú principal: crear nou qüestionari o editar un existent
- Configurar qüestionari
- Editar qüestionari
- Editar pregunta
- Editar resposta
- Exemple qüestionari resultat

Q? Q?Web - Edició de resposta X

Títol qüestionari: Ajuda

Pregunta:

Resposta:

Tipus de resposta: Resposta correcta Resposta incorrecta

Visualitzador qüestionari:

- autoevaluacion
 - Autoavaluació Hiragana
 - Preguntes:
 - Com es pronuncia aquesta sí'laba Hiragana?
 - no.jpg
 - Com es pronuncia aquesta sí'laba Hiragana?
 - ka
 - true
 - ba
 - false
 - ka.jpg

S'arriba a aquesta pantalla des de 'Editar pregunta'. La pantalla estarà buida si la resposta és nova, i presentarà dades si s'està editant una pregunta existent. Una resposta es compon d'un text i un tipus (correcta o incorrecta). La pantalla permet:

- inserir o modificar text
- marcar o canviar tipus de resposta
- desar la resposta
- eliminar la resposta
- accedir a l'ajuda (s'obre al navegador de l'usuari)
- sortir sense desar la resposta (es torna a 'Editar pregunta')

Tutorial: Fer o editar qüestionaris d'autoavaluació amb QWebEditor

Exemple qüestionari resultat

- Triar idioma i veure presentació de l'editor
- Menú principal: crear nou qüestionari o editar un existent
- Configurar qüestionari
- Editar qüestionari
- Editar pregunta
- Editar resposta
- Exemple qüestionari resultat

Autoavaluació Hiragana

Pregunta: 1 / 2

Com es pronuncia aquesta síl·laba Hiragana?

か

Respostes:

ba

ka

Sortir Continuar

Autoavaluació Hiragana

Pregunta: 2 / 2

Com es pronuncia aquesta síl·laba Hiragana?

の

Respostes:

no

ta

Sortir Continuar

Informe de resultats

Autoavaluació Hiragana

Respostes correctes: 1 - Respostes incorrectes: 1

Com es pronuncia aquesta síl·laba Hiragana?

か

ba

ka

Com es pronuncia aquesta síl·laba Hiragana?

の

no

ta

Sortir

Aquesta pantalla mostra el resultat d'un qüestionari d'autoavaluació: pantalles pregunta/respostes i informe final de resultats.

Continuar

12.2 Tutorial Editor Q?Web (generació de tutorials web)

Tutorial: Fer o editar tutorials amb QWebEditor

Fer o editar tutorials amb QWebEditor

Triar idioma i veure presentació de l'editor
Menú principal: crear nou tutorial o editar un existent
Configurar tutorial
Editar tutorial
Editar pantalla

Aquest tutorial explica com fer o editar tutorials web amb Q?WebEditor

Continuar

Tutorial: Fer o editar tutorials amb QWebEditor

Triar idioma i veure presentació de l'editor

Triar idioma i veure presentació de l'editor

Menú principal:
crear nou tutorial o editar un existent

Configurar tutorial

Editar tutorial

Editar pantalla

Q? Q?WEB

Català

Español

OK

Q? Missatge

Copyright (C) 2012 Manel Cámara

Q?WebEditor is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or any later version.

Q?WebEditor is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details at <http://www.gnu.org/licenses/#GPL>

Aceptar

En executar Q?WebEditor, l'aplicació sol·licita que es triï l'idioma de l'editor. Una vegada fet, l'editor presenta una pantalla informativa amb dades sobre l'autor i la llicència de distribució del programari. En acceptar, l'aplicació presenta el menú principal.

Continuar

Tutorial: Fer o editar tutorials amb QWebEditor**Menú principal: crear nou tutorial o editar un existent**

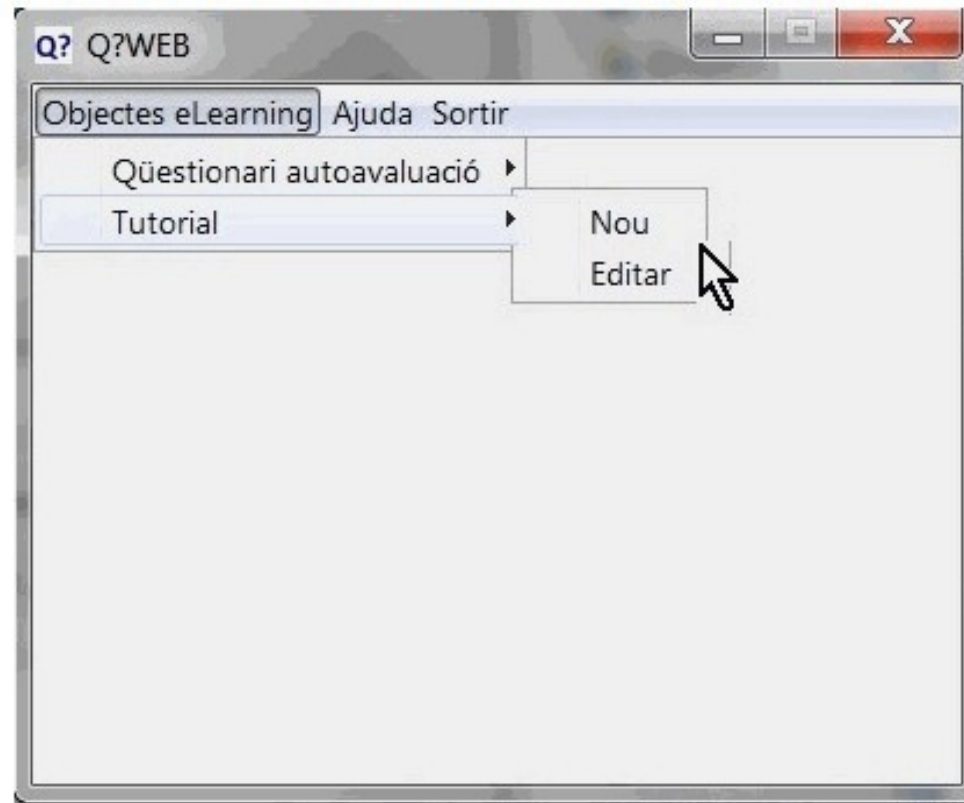
Triar idioma i veure presentació de l'editor

Menú principal: crear nou tutorial o editar un existent

Configurar tutorial

Editar tutorial

Editar pantalla



Al menú principal es pot triar si es vol fer un tutorial nou o editar un existent. En el primer cas, l'editor condueix a la pantalla de configuració d'un nou tutorial (veure 'Configurar tutorial'). En el segon cas, l'editor permet navegar per l'estructura d'arxius locals i localitzar el paquet scorm a editar (fitxer zip). Si el fitxer és vàlid, l'editor condueix a la pantalla 'Editar tutorial', que ja té carregat en memòria el tutorial font.

Continuar

Tutorial: Fer o editar tutorials amb QWebEditor

Configurar tutorial

Triar idioma i veure presentació de l'editor

Menú principal: crear nou tutorial o editar un existent

Configurar tutorial

Editar tutorial

Editar pantalla

Q? Introducció de dades

Títol tutorial?

Text de presentació:

Idioma:

S'arriba a aquesta pantalla si l'usuari ha triat l'opció de fer un nou tutorial. En aquesta, l'usuari pot especificar les dades bàsiques del nou tutorial: títol, text de presentació i idioma.

Tutorial: Fer o editar tutorials amb QWebEditor

Triar idioma i veure presentació de l'editor

Menú principal: crear nou tutorial o editar un existent

Configurar tutorial

Editar tutorial

Editar pantalla

Editar tutorial

Q? Q?Web - Edició de tutorial

Títol tutorial: Ajuda

Idioma:

Text de presentació:

Pantalles:

Visualitzador tutorial:

- tutorial
 - Fer tutorials amb Q?WebEditor
 - Pantalles:
 - Configuració dades bàsiques nou tutorial
 - ConfiguracioTutorial.jpg
 - Elecció idioma i presentació Q?WebEditor
 - PresentacioIdioma.jpg

És aquesta la pantalla principal en l'edició d'un tutorial, ja sigui nou o existent. Permet:

- afegir una nova pantalla
- editar una pantalla existent (triant-la del desplegable 'pantalles')
- desar el tutorial (sempre que tingui un mínim d'una pantalla)
- accedir a l'opció d'ajuda (obre ajuda en navegador web de l'usuari)
- sortir sense desar el tutorial

La part inferior de la pantalla conté un visualitzador que permet monitoritzar l'estat i els canvis del tutorial al llarg del procés d'edició.

Continuar

Tutorial: Fer o editar tutorials amb QWebEditor

- Triar idioma i veure presentació de l'editor
- Menú principal: crear nou tutorial o editar un existent
- Configurar tutorial
- Editar tutorial
- Editar pantalla

Editar pantalla

Q? Q?Web - Edició de pantalla tutorial Ajuda

Títol tutorial:

Títol pantalla:

Explicació pantalla: Aquesta pantalla permet crear o editar una nova pantalla del tutorial. Una pantalla (o slide) està formada per: títol, explicació i imatge (opcional). Les opcions disponibles són:
 - triar o eliminar imatge (format gif o jpg)
 - desar la pantalla
 - eliminar la pantalla

Imatge: EdicioPantalla.jpg

Visualitzador tutorial:

- tutorial
- + Fer tutorials amb Q?WebEditor

S'arriba a aquesta pantalla des de 'Editar tutorial'. Apareixerà buida en cas de nova pantalla, i amb dades carregades en cas d'edició de pantalla existent. Una pantalla es compon de: títol, text explicatiu i imatge (opcional). Les opcions disponibles són:

- inserir o modificar títol
- inserir o modificar text explicatiu
- triar o eliminar imatge (gif o jpg)
- desar la pantalla
- eliminar la pantalla
- accedir a l'ajuda (s'obre al navegador web de l'usuari)
- sortir sense desar la pantalla (l'aplicació torna a 'Editar tutorial')

La part inferior de la pantalla conté un visualitzador que permet monitoritzar l'estat i els canvis del tutorial al llarg del procés d'edició.

12.3 Exemple de qüestionari d'autoavaluació

The image displays three screenshots of a self-assessment quiz for Japanese Hiragana pronunciation.

Autoavaluació Hiragana - Pregunta: 1 / 2
 Com es pronuncia aquesta síl·laba Hiragana?
 か
 Respostes:
 ba
 ka
 Sortir Continuar

Autoavaluació Hiragana - Pregunta: 2 / 2
 Com es pronuncia aquesta síl·laba Hiragana?
 の
 Respostes:
 no
 ta
 Sortir Continuar

Informe de resultats
Autoavaluació Hiragana
 Respostes correctes: 1 - Respostes incorrectes: 1
 Com es pronuncia aquesta síl·laba Hiragana?
 か
 ba
 ka
 Com es pronuncia aquesta síl·laba Hiragana?
 の
 no
 ta
 Sortir

12.4 Exemple de fitxer xml de continguts

```

<?xml version="1.0" encoding="UTF-8"?>
- <questionario titulo="Autoavaluació Hiragana" tipoCuestionario="autoevaluacion"
  presentacion="Qüestionari d'autoavaluació del silabari japonès Hiragana" idioma="ca_ES">

- <pregunta>
  <textoPregunta>Com es pronuncia aquesta síl·laba Hiragana?</textoPregunta>
  <imagenPregunta>ka.jpg</imagenPregunta>
  - <respuesta>
    <tipoRespuesta>false</tipoRespuesta>
    <textoRespuesta>ba</textoRespuesta>
  </respuesta>
  - <respuesta>
    <tipoRespuesta>>true</tipoRespuesta>
    <textoRespuesta>ka</textoRespuesta>
  </respuesta>
</pregunta>
- <pregunta>
  <textoPregunta>Com es pronuncia aquesta síl·laba Hiragana?</textoPregunta>
  <imagenPregunta>no.jpg</imagenPregunta>
  - <respuesta>
    <tipoRespuesta>>true</tipoRespuesta>
    <textoRespuesta>no</textoRespuesta>
  </respuesta>
  - <respuesta>
    <tipoRespuesta>false</tipoRespuesta>
    <textoRespuesta>ta</textoRespuesta>
  </respuesta>
</pregunta>
</questionario>

```

12.5 GitHub: url's dels dos projectes i captura de pantalles

<https://github.com/manelcamara/QWebEditor>

Inc. [US] <https://github.com/manelcamara/QWebEditor>

The screenshot shows the GitHub interface for the repository 'manelcamara / QWebEditor'. At the top, there's the GitHub logo and navigation links like 'Signup and Pricing', 'Explore GitHub', 'Features', 'Blog', and 'Login'. Below the repository name, there are 'Watch' (1) and 'Fork' (1) buttons. A navigation bar includes 'Code', 'Network', 'Pull Requests' (0), 'Issues' (0), and 'Graphs'. The repository description is 'Projecte QWeb UOC Manel Cámara Juny 2012: programari QWebEditor'. There are buttons for 'Clone in Windows', 'ZIP', 'HTTP', and 'Git Read-Only' with the URL 'https://github.com/manelcamara/QwebEditor.git'. A 'Read-Only access' button is also present. Below this, there's a 'branch: master' dropdown, 'Files', 'Commits', 'Branches' (1), 'Tags', and 'Downloads' tabs. A section for 'Latest commit to the master branch' shows a commit titled 'Commit inicial' by 'MANEL' from 'an hour ago' with commit ID '1d0ff8200d'. At the bottom, there's a table for the repository files.

name	age	message	history
QwebEditor	an hour ago	Commit inicial [MANEL]	

<https://github.com/manelcamara/QWebScormApp>

Inc. [US] <https://github.com/manelcamara/QWebScormApp>

github

[Signup and Pricing](#)
[Explore GitHub](#)
[Features](#)
[Blog](#)
[Login](#)

PUBLIC
manelcamara / QWebScormApp

👁 Watch 1
🍴 Fork 1

Code
Network
Pull Requests 0
Issues 0
Graphs

Projecte QWeb UOC Manel Cámara Juny 2012: programari QWebScormApp

📦 Clone in Windows
📄 ZIP
HTTP
Git Read-Only
https://github.com/manelcamara/QWebScormApp.git
🔒 Read-Only access

🌿 branch: master ▾
Files
Commits
Branches 1
Tags
Downloads

🕒 Latest commit to the **master** branch

Commit inicial

👤 MANEL authored 22 minutes ago
📄 commit 3cff912686

QWebScormApp /

name	age	message	history
📁 QwebScormApp	22 minutes ago	Commit inicial [MANEL]	