

# Marques d'aigua per al seguiment de documents PDF

Alex Rojas Rodríguez

**Resum**— Aquest projecte té com objectiu principal la implementació d'una eina capaç de fer modificacions en un document PDF per posteriorment dur a terme un seguiment sobre el mateix. La motivació d'aquest projecte és l'increment, en la actualitat, de l'ús de la xarxa per a compartir documents, molts cops amb drets d'autor, i del poc control que tenim sobre aquets. Per a dur a terme el nostre objectiu farem ús de la llibreria iText, programada en Java, que ens permetrà poder fer un marcatge digital, o marca d'aigua, en un document un cop sigui descarregat. Per a la part pràctica, tenint en compte que aquest projecte es desenvolupa en un àmbit universitari, s'utilitzarà l'exemple d'un professor que penja els seus apunts a la plataforma del Campus Virtual. Aquest cas pràctic ens mostrarà el procés de marcatge i la posterior recuperació de les dades marcades en el document.

**Paraules clau**—Marques d'aigua, PDF, divulgació sense consentiment, Java, iText, seguretat, drets d'autor.

**Abstract**— This project's main objective is the implementation of a tool able to make changes in a PDF document for later conducted a follow-up on it. The motivation of this project is the increase, in the present, of use the network to share documents, often copyrighted, and we have little control over these last. In order to accomplish our goal we will use the iText library, programmed in Java, allowing us to make a digital mark, or watermark, in a document after it is downloaded. For the practical part, given that the project is developed in a university setting, used the example of a teacher who shared his notes on the Virtual Campus platform. This practice case will show the marking process and the subsequent recovery of the data marked in the document

**Index Terms**—Water marks, PDF, divulgation without consent, Java, iText, security, copyright.

---

## 1 INTRODUCCIÓ

**A**CTUALMENT existeix un intercanvi massiu de documents a través de la xarxa. No és nou el fet de que cada cop més, per certs aspectes quotidians de la nostra vida, s'està suprimint la utilització de documents físics i cada cop més entrem en contacte amb els documents digitals, ja sigui per omplir formularis a tercers, per viatjar, o fins i tot per rebre els resultats mèdics de la nostra última revisió.

Tot això sembla un avanç, més ben dit, és un avanç, però que passa quan es fa un ús il·lícit d'aquest?, que passa quan algú s'apropia de documents que no són de la seva propietat i en fan un ús indegut? com per exemple la divulgació sense consentiment, la còpia o fins i tot, en documents que ho permetin, adjudicar-se

els drets d'autor.

La divulgació sense permís explícit del autor o la apropiació indeguda dels drets d'aquest, esta contemplat en la llei, més específicament en la Llei de Propietat Intel·lectual [1]. Una pregunta que ens podria sorgir en un moment donat i relacionada amb el àmbit en el que es desenvolupa aquest projecte, l'universitari, és: *sota quins drets estan protegits els meus apunts, o els del professor?*

La resposta en forma d'article la podem trobar dins de la Llei de Propietat Intel·lectual i ens ve a dir que: *"tots els discursos i al·locucions, conferències, explicacions de càtedra i similars, estan protegits pel Dret d'Autor [...]"*<sup>1</sup>. Per tant, els nostres apunts no son altra cosa que una reproducció en sentit legal de la exposició del professor, que en el cas que les explicacions siguin originals, conserven tots els drets morals i d'exploatació.

- 
- E-mail de contacte: alex.rojas@e-campus.uab.cat
  - Menció realitzada: Tecnologies de la Informació.
  - Treball tutoritzat per: Jordi Duran Cals (Departament d'Enginyeria de la Informació i de les Comunicacions)
  - Curs 2015/16

<sup>1</sup> Els articles en qüestió són: 10.1 (a), 18, 19 i 20 de la Llei de Propietat Intel·lectual.

Aquest projecte busca donar una solució correcta a totes les possibles accions amb una intenció il·lícita a menor escala, fet per el qual el projecte només pretén ser una solució, per el moment, a favor de la protecció de documents del tipus PDF.

Aquesta solució ve motivada per certs estudis [2] que afirmen que es pot dissuadir el mal ús d'alguns documents mitjançant una marca d'aigua.

### 1.1 Estructura de l'article

Aquest article està estructurat de la següent manera:

En la primera part es farà una breu **introducció** al tema, fent ús d'alguns exemples per ficar-nos en situació. Dins d'aquesta primera part trobarem els **objectius**, on mitjançant una llista es mostraran en ordre de prioritat o necessitat. També farem referència a l'**estat de l'art** per conèixer que existeix i que ens ha motivat a dur a terme aquest treball.

La segona part mostrarà la **metodologia** utilitzada per el projecte, fent una breu presentació de com s'ha seguit aquesta metodologia i quins passos ha tingut dins del projecte.

En tercer lloc es dura a terme una breu descripció de com ha estat el **desenvolupament** i una justificació del perquè s'ha decidit desenvolupar cada part com s'ha fet. Es farà un repàs començant per la primera marca i acabant amb l'obtenció de la marca definitiva.

Com a quarta part mostrarem els **resultats**. Serà allà on es farà referència més explícitament quin ha estat el producte final del projecte. La última, i cinquena part, es dedicarà a les **conclusions**.

### 1.2 Objectius

Els objectius del projecte són els següents:

- Catalogar diferents tècniques de generació de marques d'aigua sobre documents.
- Analitzar l'estat de l'art de les diferents eines existents que possibiliten el tractament de marques d'aigua.
- Definir el conjunt d'informació necessària per a la traçabilitat del document.
- Avaluar el cost de generació de la marca al descarregar un document PDF, tenint en compte la mida d'aquest.
- Estudiar la possibilitat de securitzar la marca d'aigua, xifrant-la, i no permetre la seva eliminació.

Aquets objectius volen aconseguir que arribem des de la categorització de les marques d'aigua, com part inicial i preparatòria, fins a aconseguir realitzar una marca d'aigua en un document per poder fer després un seguiment d'aquest.

Per a la realització d'aquest projecte s'ha fet ús del llenguatge de programació Java, ajudant-nos del IDE<sup>2</sup> Eclipse [3].

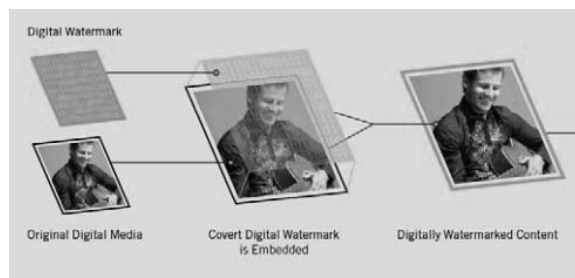


Fig 1 : Procés de marcatge digital [2]

La part relacionada amb el tractament del PDF i el posterior marcatge digital, com el que es pot veure a la Figura 1, ha estat desenvolupada utilitzant la llibreria iText [4].

### 1.3 Estat de l'art

Tenint en compte l'esmentat anteriorment i amb les tècniques avançades que existeixen avui en dia no es cap sorpresa el fet que ja existeixin eines que puguin o intentin garantir certs drets en documents publicats a la xarxa.

Com a institució a destacar trobem a la Digital Watermarking Alliance, en endavant DWA. La pròpia associació es defineix com una aliança internacional d'organitzacions de la indústria líder que ofereixen valuoses solucions de marcatge digital per a una ampla gama de clients i mercats de tot el món. Una aplicació amb un gran pes dins d'aquest treball i de la propietat de la DWA es la de: Forensics and Piracy deterrence [5]. L'objectiu d'aquesta aplicació és el mateix que el de aquest projecte, anar més enllà i poder saber mitjançant un marcatge digital qui ha estat qui ha divulgat el contingut.

Per altra banda, i també relacionat amb el tema en qüestió, existeix una altra companyia anomenada Digimark, que ofereix el producte Guardian [6]. Aquest producte permet als seus clients acabar amb la pirateria de les seves creacions, utilitzant també una marca d'aigua.

Com podem veure en la actualitat existeix un sector de la tecnologia dedicat a la implantació de mesures per a protegir certs drets sobre material publicat a la xarxa.

En aquest apartat he fet referència a dues institucions importants però el fet de protegir el que es nostre ha donat peu a que existeixin moltes més institucions que vulguin donar la mateixa solució.

## 2 METODOLOGIA

La metodologia escollida per a la realització d'aquest projecte va ser Scrum[7-8]. El fet que ha ajudat a l'elecció de la metodologia és que cada cop més les metodologies àgils estan guanyant la partida a les al-

ment interactiu, Integrated Development Environment (IDE) en anglès, és una aplicació informàtica que proporciona serveis integrals per facilitar-li al desenvolupador o programador el desenvolupament de programari

<sup>2</sup> Un entorn de desenvolupament integrat o entorn de desenvolupa-

tres. Avui en dia moltes empreses apliquen aquestes metodologies per a la realització dels seus projectes. Un cop donades a conèixer les raons més professionals no s'han de deixar de banda les més importants, i les que realment decanten la balança. Aquestes raons estan relacionades amb la necessitat del projecte. El cas actual, el nostre projecte conté una gran part relacionada amb la cerca de informació, fent així que el desenvolupament d'aquest no pugui ser predictiu i depengui per tant dels resultats d'aquesta cerca, i s'hagi d'anar avaluant el procés de desenvolupament a mesura que avança el projecte.

És cert que Scrum és una metodologia que s'utilitza per el treball en equip però cal a dir que durant el transcurs del projecte s'ha adaptat aquesta metodologia a la situació del projecte, ja que al ser un equip de dues persones, s'han hagut de assumir més d'un rol per persona per arribar a cobrir tots els rols del scrum original. Aquesta variació en l'aplicació de l'Scrum no es nova i existeixen algunes persones que ja han fet referència al treballar sol utilitzant metodologies àgils [9].

Les modificacions esmentades anteriorment són les següents: Els rols procedents de la metodologia van ser assumits, com ja s'ha avançat anteriorment, per dues persones, alumne i tutor. El tutor va assumir el rol de *Product Owner*, que vindria a ser el que parla com el client. Per altre banda l'alumne va assumir els dos rols restants, *Scrum Master* i únic membre del *Development Team*. Les reunions diàries relacionades amb l'avanç del projecte van fer-se individualment, però no diàriament. Els sprints han tingut una duració de dues setmanes, durant els quals es treballaven diferents aspectes relacionats amb el projecte, tots classificats com històries d'usuari. En aquets sprints s'ha intentat que tant l'obertura, com el tancament dels mateixos, es facin amb la presència o amb el suport del tutor del treball de recerca.

Els primers dies es va definir el *product backlog* del projecte, creant així tantes històries d'usuari com objectius o necessitats teníem.

Al inici del projecte el *product backlog* era el següent:

- Web amb accés d'usuaris
- Web per a la descarrega de documents
- Catalogar diferents tècniques de generació de marques d'aigua
- Analitzar l'estat de l'art de les diferents eines existents que possibiliten el tractament de marques d'aigua
- Definir el conjunt d'informació necessària per a la traçabilitat del document
- Avaluat el cost de generació de la marca al descarregar un document PDF, tenint en compte la mida

Durant l'avanç del projecte aquest *product backlog* ha anat modificant-se i aquets canvis s'aniran fent vi-

sibles a mesura que s'avanci en aquest article. L'ampliació d'aquest *product backlog* normalment es produïa un cop arribat al *Sprint Planning Meeting*.

## 2.1 Aplicació del Scrum – Sprints

### 2.1.1 Sprint 1(12 d'Octubre al 25 d'Octubre)

La primera part del treball va estar dedicada al disseny del que seria el cas pràctic i a la recopilació d'informació relacionada amb el tema principal, el marcatge de PDF.

Per a la creació del cas pràctic es va voler relacionar amb el tema del que s'està parlant, els drets d'autor i la distribució sense consentiment, i es va voler que estigués orientat al màxim en l'àmbit universitari. Vam posar el cas de uns apunts penjats per un professor en el Campus Virtual, en endavant CV. Com ja he explicat anteriorment aquets apunts tenen unes lleis que els protegeixen, però i si algú vol fer-ne una distribució en una web d'un tercer i adjudicar-se el mèrit o simplement difondre la informació? Doncs el que es va decidir crear en aquest cas pràctic és una eina capaç de fer un marcatge d'un document PDF un cop es descarregat utilitzant la plataforma del CV. El disseny d'aquest cas pràctic s'ha intentat fer el més senzill possible, tenint en compte que la part important és el marcatge del document.

### 2.1.2 Sprint 2(26 d'Octubre al 9 de Novembre)

En la segona part, un cop disposàvem de un lloc web ben definit on provar la nostra aplicació i una bona documentació, vam començar a desenvolupar les parts bàsiques, com podrien ser: La descarrega de PDF, el *Login* dels usuaris del CV, el conjunt de documents a descarregar, etc.

### 2.1.3 Sprint 3(10 de Novembre al 23 de Novembre)

En aquest tercer sprint, o tercera part, ja ens centràvem en el marcatge i en analitzar els elements generats a partir d'aquets. Es va poder crear una marca visible i pública i tot i que no complís amb els requisits que buscàvem, (privada – invisible – robusta), ja era una marca en un document PDF. En aquest apartat també es va fer un anàlisi de l'estructura del PDF per tal de trobar la marca dins del document. Finalment es va poder trobar la marca però no es va poder extreure de l'arxiu. Per això es va tindre que pensar una altra solució.

### 2.1.4 Sprint 4(24 de Novembre al 6 de Desembre)

En aquest punt ja es disposava d'una marca però no s'havia aconseguit trobar-la dins del document, o més ben dit extreure-la. En el tema del marcatge es va millorar la marca, fent-la invisible. L'altre apartat important en aquest punt i que, com ja hem dit abans, no es va aconseguir en el Sprint anterior era la recuperació de la marca. Per a poder dur a terme aquesta funcio-

nalitat finalment es va fer ús de la llibreria iText, la mateixa que utilitzàvem per a crear la marca va ser l'encarregada de extreure-la.

Durant aquest sprint es van afegir dos noves històries d'usuari al product backlog

- Estudiar la possibilitat de xifrar la marca d'aigua i no permetre la seva eliminació.
- Crear un apartat dins de l'aplicació web que ens permeti recuperar la marca.

### 2.1.5 Sprint 5(7 de Desembre al 20 de Desembre)

El cinquè sprint va ser l'encarregat de portar la privacitat a la marca. Es va xifrar la marca mitjançant un algorisme de xifratge, amb el que fent servir una clau privada podíem fer que aquesta marca fos privada per els usuaris que no tinguessin permisos. En aquest Sprint també es van fer les proves de rendiment en la descàrrega.

### 2.1.6 Sprint 6(21 de Desembre al 3 de Gener)

En el sisè i últim Sprint es va acabar de definir l'aspecte de la pàgina que recuperava la marca, afegint en ella un uploader per a poder adjuntar el document que es vulgui comprovar. Per últim es van fer ajustos en la recuperació de la marca, com podrien ser afegir certs caràcters en el marcatge per a detectar millor la marca un cop estigui dins del document.

## 3 DESENVOLUPAMENT

### 3.1 Creació de la base del projecte

Per aquest apartat s'utilitzarà una estructura semblant a la que he esmentat anteriorment en l'apartat *Sprints*, però en aquest cas agruparem els sprints en tres parts: primera<sup>3</sup>, segona<sup>4</sup> i tercera part<sup>5</sup>.

La primera d'aquestes tres parts, i com ja s'ha dit amb anterioritat, ha estat la part de disseny. En aquest apartat era important tenir unes idees ben estructurades, ja que, tot i ser el principi del treball, era el que marcaria la correcta execució dels següents passos.

Centrant-nos en la part de disseny, i deixant per més endavant la part de documentació i els possibles objectius que vam assolir en aquesta, es parlarà de com ha estat el procés de construcció de la maqueta. Tenint en compte que en l'àmbit universitari la relació amb una eina com el CV és una cosa quotidiana no va ser gaire difícil decidir quina seria la estructura de la maqueta. Per això, va ser una dedició unànime amb el tutor que s'havia de disposar d'una estructura clàssica de CV, com la següent: Panell de registre en el que un usuari autoritzat s'autentica i accedeix dins l'eina. Un cop autenticat tindria accés a la pàgina principal en la que podria veure el llistat de documents. El disseny final del panell de Login es pot veure a la Figura 2.

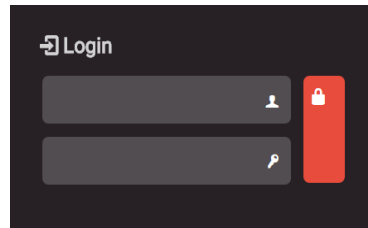


Fig 2 : Panell de registre de l'aplicació.

Per a la validació de l'usuari, i fer que realment es comprovi que l'usuari es autoritzat s'ha fet ús de una base de dades fictícia. Per aspectes relacionats amb el temps del projecte, i tenint en compte que la part principal d'aquest projecte era el marcatge del PDF, es va decidir que per l'emmagatzematge de les dades no s'utilitzaria una base de dades convencional sinó que es faria ús de una estructura de Java per a simular aquesta funcionalitat. La estructura escollida va ser la dels Beans<sup>6</sup> de Java.

S'han creat dues estructures que representaran els dos principals objectes del projecte, una per a l'usuari, nom i contrasenya, i una altra que correspon a un arxiu PDF, que té com a atributs una ruta fins al fitxer i una vista prèvia del PDF per mostrar en el llistat de documents. Aquests dos objectes han estat emmagatzemats a la vegada en dos Beans diferents més, un per a la base de dades de usuaris i un altre per a la base de dades de documents.

```
private static ArrayList<PdfBean> pdfDB;

@PostConstruct
private void init() {
    pdfDB = new ArrayList<PdfBean>();
    pdfDB.add(new PdfBean("TFG.pdf", "resources/previews/TFG.png"));
    pdfDB.add(new PdfBean("TFG2.pdf", "resources/previews/TFG2.png"));
    pdfDB.add(new PdfBean("OCA.pdf", "resources/previews/OCA.png"));
    pdfDB.add(new PdfBean("ESTEGO.pdf", "resources/previews/ESTEGO.png"));
    pdfDB.add(new PdfBean("TA.pdf", "resources/previews/TA.png"));
}
```

Fig 3 : Funció init() del Bean de la Base de Dades de documents.

En la Figura 3 podem veure una estructura que ens assegura que tindrem una "Base de Dades" cada cop que el nostre projecte sigui executat. Es crea una Llista, ArrayList, de l'estructura PdfBean que es l'estructura dels documents PDF, i es omplerta per tants documents com es vulgui. Un cop s'executa el nostre programa, i tenim la nostra classe Java configurada com @Singleton i @Startup<sup>7</sup>, aquesta ens crea la estructura i ens la fa accessible per a poder treballar amb els documents dels que disposa.

Pel que fa als usuaris, en la Figura 4, la funcionalitat és la mateixa, però en comptes de veure una llista de Beans de documents tindrem una llista de Beans d'usuaris en els que el seu constructor ens dirà quin és el seu usuari i la seva paraula de pas.

<sup>6</sup> Un JavaBean és una classe simple de Java que compleix certes normes amb els seus atributs i mètodes: Constructor sense arguments, atributs privats i mètodes getters i setter per aquets atributs.

<sup>7</sup> @Singleton: Només es crea una única instància del Bean en tot el projecte. @Startup: Es crea abans de que ningú el cridi, d'aquesta manera un cop executat el projecte ja se'n pot fer ús.

<sup>3</sup> Sprint I.

<sup>4</sup> Sprint II.

<sup>5</sup> Sprint III, Sprint IV, Sprint V i Sprint VI.

```
private static ArrayList<UserBean> usersDB;

@PostConstruct
private void init() {
    usersDB = new ArrayList<UserBean>();
    usersDB.add(new UserBean("Alex", "1234"));
    usersDB.add(new UserBean("Jordi", "4321"));
    usersDB.add(new UserBean("admin", "adminadmin"));
}
```

Fig 4 : Funció init() del Bean de la Base de Dades de usuaris.

Fins aquí del que es disposa es de dues pàgines web del tipus *.html*, una per el registre i una altra com a pàgina principal, i diferents estructures *.java* que controlen el funcionament intern del CV<sup>8</sup>.

### 3.1.1 Cerca d'informació

Per a poder entrar en el tema del marcatge del PDF era important disposar de coneixements per a poder decidir quin tipus de marca era millor per el nostre cas, i el que era més important encara, quina eina es volia utilitzar per a realitzar-la.

Es va tindre que fer una cerca per a buscar els diferents tipus de marques digitals que existien, i es va acabant classificant, depenent de certes característiques, de la manera que es mostra a la Figura 5:

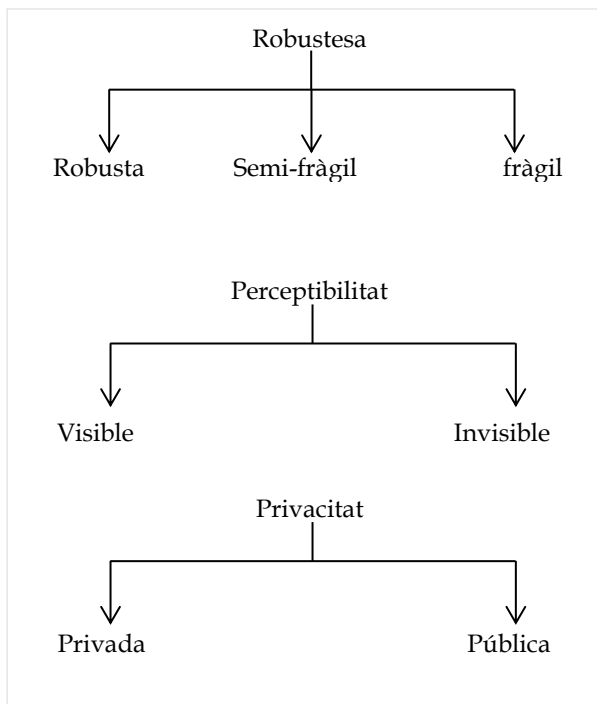


Fig 5 : Classificació de les marques d'aigua.

#### Robustesa:

La primera de les tres estructures que podem observar a la Figura 5 és la robustesa. Hi ha tres nivells de robustesa. La primera, la robusta, ens ve a dir que les marques són resistents tot i que es facin diferents transformacions en l'arxiu, com podrien ser còpies, modificacions en la mida, etc. Quan parlem de mar-

ques semi-fràgils ens referim a les marques que són resistents a transformacions lícites, però per el contrari, si la transformació es il·lícita, un atac per exemple, poden ser extretes. Aquestes últimes estan més extenses en els casos en el que es busca assegurar la integritat de les dades. Per últim en aquest apartat trobem les fràgils, que com el seu nom indica tenen una vida molt curta si li fem la més mínima modificació.

#### Visibilitat:

Una característica més, i molt important en algunes implementacions, és la visibilitat. Per a que ens pot servir que una marca sigui visible o invisible? En el cas de la visible ens dona més benefici si volem protegir l'arxiu en condicions físiques i no digitals, com per exemple en el cas d'un document que s'hagi d'imprimir. Tot i així, que sigui visible també permet gaudir de certa protecció en l'àmbit digital. En canvi, si volem que el nostre document no sigui replicat per la xarxa, per tant en l'àmbit digital, i a més a més la nostra marca conté informació sensible, com podrien ser noms, dates, IPs, i de més informació crítica, tindrem que utilitzar la invisibilitat. La invisibilitat ens garanteix, d'alguna manera, més privacitat.

#### Privacitat:

La última, la privacitat. Si volem que alguna cosa sigui privada no voldrem que ningú tingui accés a ella. En aquest cas la podríem comparar amb la visibilitat tot i que no necessàriament no ha de ser perceptible per l'ull humà sinó que es pot fer utilitzant algunes tècniques com per exemple el xifratge. Per altra banda si la nostra marca pot ser pública no hauríem de perquè ferla invisible, a no ser que existeixin necessitats afegides, com per exemple que s'hagi d'utilitzar el document imprès, per tant podria donar-se el cas que la marca sigui pública però que no es mostri al document de manera visual.

En el nostre cas, basant-nos en les nostres necessitats, vam haver de crear una marca el més robusta possible, invisible i privada.

Abans d'acabar aquesta primera part es va fer un avanç en la posada en marxa del lloc web, més específicament es va crear el mòdul encarregat de la descarrega. Per a la realització d'aquesta part es va implementar un Java Servlet<sup>9</sup> que permetés realitzar la descarrega del document que es demani en la petició web.

Aquest Servlet més endavant serà l'encarregat de cridar a la funció que ens genera la marca, ja que el procés de marcatge es produeix cada cop que un document és descarregat.

En el nostre cas el funcionament del servlet es el mostrat en la Figura 6.

<sup>9</sup> Un servlet és una classe que respon a un tipus particular de sol·licitud de xarxa (*request*) - més comunament una sol·licitud HTTP. Els servlets s'utilitzen generalment per implementar aplicacions web.

<sup>8</sup> El diagrama de classes es pot observar en l'apèndix 1.

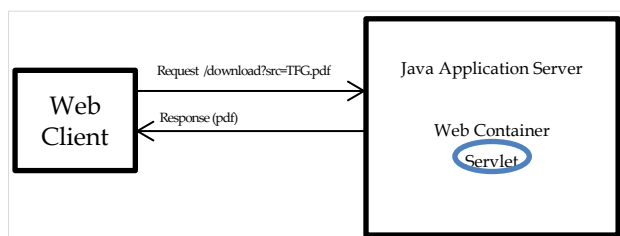


Fig 6 : Funcionament del servlet implementat.

En aquest punt disposàvem, com ja he repetit anteriorment, d'un entorn preparat per a poder treballar amb l'eina de marcatge digital, passem doncs a la explicació de com hem arribat a la marca.

### 3.2 Procés de marcatge digital

Per a la execució del procés de marcatge primer es va fer un estudi de quina era l'eina més viable per aconseguir l'objectiu principal. Pel que fa a el llenguatge java existeixen diferents llibreries que permeten interactuar amb els documents PDF, entre les que podem trobar *qoopa*, *PdfBox*, *iText*, *PDF Clown*, *Pdf-renderer*, *jPod*, *Pdfjet*, etc [10,11,4,12,13,14,15]. Finalment es va decidir treballar amb *iText* degut a que disposava de totes les funcionalitats necessàries per a l'execució de la nostra marca i a més a més de una documentació molt extensa que ens garanteix en major percentatge que altres que ens serà més fàcil treballar amb ella. També cal afegir que aquesta disposava d'una avantatge sobre les altres, i era que feia una setmana que treballàvem amb ella en busca de algun resultat.

Es va crear una classe Java encarregada del procés. Aquesta classe seria l'encarregada de contenir els mètodes per a el corresponent marcatge. És cridada des de el servlet i genera una marca, que conté una marca de temps i el nom de l'usuari que ha descarregat el fitxer. Per seguir amb la simulació d'un cas real com el CV, el nom que imprimim a la marca és el que ha entrat l'usuari, fent així un símil amb el NIU que s'utilitza a la universitat.

Necessitàvem una funció capaç de generar una marca en un PDF i que aquesta marca a més a més sigui invisible i privada. Anant per parts, per a la primera fase de desenvolupament de la marca es va posar com objectiu arribar a obtenir una marca, encara que sigui visible i pública.

### 3.3 Marca visible i pública

Un dels punts importants d'aquesta primera part és la decisió de marcar, no solament la primera pàgina, sinó que es fa un marcat en cadascuna de les pàgines del document. D'aquesta manera fem la nostra marca una mica més robusta, ja que contemplem la possibilitat que aquest document pugui ser dividit i per tant la possibilitat de perdre la marca si només l'apliquem a una pàgina.

El funcionament de la funció<sup>10</sup> de la marca, en aquest punt del desenvolupament del projecte, és el següent: Donat la ruta d'un document, el que volem descarregar, es fa una llegida d'aquest per a tenir la seva informació guardada en un objecte Java, en aquest cas l'objecte de la llibreria *iText PdfReader*. Un cop disposem d'aquesta informació podem procedir a preparar el marcatge, per això necessitem un objecte *PdfStamper*. El que farem serà guardar el pdf a descarregar amb les marques introduïdes en un document nou, això ens facilitarà tenir un document base del qual podrem anar generant còpies que es descarregaran els usuaris, aquest document nou es defineix en la construcció de l'objecte *PdfStamper*. Amb aquets dos objectes s'obtenen les funcions necessàries per poder generar un document marcat a partir de un document ja existent.

Per a crear la corresponent marca tenim la opció decidir tot tipus de configuracions, com on es vol col·locar dins de la pàgina, color, rotació, etc. Per a aquesta funcionalitat es disposa de un objecte *Rectangle* que permet treballar amb la pàgina en qüestió, ja siguin dimensions, marges, alguns colors, etc. Aquesta funcionalitat es interessant quan es vol que la marca sigui visible, ja que la marca hauria de complir certs requisits de estil, però per una marca invisible es podria suprimir. Tot i així més endavant es farà una explicació del perquè s'ha mantingut en el codi.

Per acabar, es fa un recorregut al llarg de totes les pàgines del document utilitzant una estructura *for* i es generarà una marca a cada iteració. Un cop s'ha enllestit el procés de marcatge es molt important per a la posterior descarrega tancar tant l'objecte del document marcat com l'objecte amb el document original, *PdfStamper* i *PdfReader* respectivament.

ginal a realitzar individualment i presentar i d  
un projecte en l'àmbit de les tecnologies esp  
sional en el que es sintetitzin i integrin les com  
all suposa una càrrega de treball personal de l'est  
some text  
TFG, com a "projecte aplicat", que no s'ha de ve  
tant, no es concreta ni limita gaire el tipus de  
ol problema, sistema o situació susceptible de s

Fig 7 : Exemple de marca mitjançant el codi de l'apèndix 2

### 3.4 Marca invisible i pública

Un cop es disposa d'una marca el procés cap a la personalització d'aquesta sembla un procés més senzill.

Per a fer invisible la marca es va utilitzar un objecte, també de la llibreria *iText*, anomenat *PdfGState*.

Aquest objecte ens permet afegir tot tipus de gràfics, en el cas del projecte actual s'utilitzarà per aplicar la transparència a la marca.

Ara en el codi de la marca s'afegeix l'objecte creat a partir del constructor de *PdfGState* i a més a més s'afegeix un paràmetre d'entrada a la funció que in-

<sup>10</sup> En l'apèndix 2 es pot veure el codi de la funció

dica si es vol o no transparència. D'aquesta manera s'obté una funció que es pot utilitzar tan per marques invisibles com visibles.

L'aplicació d'aquest objecte la podem veure a la Figura 8.

```

PdfGState gState = new PdfGState();
gState.setFillOpacity(0.5f);

if (transparency)
    gState.setFillOpacity(0.0f);

for (int i = 1; i <= numberOfPages; i++) {
    page = reader.getPageSizeWithRotation(i);
    x_position = (page.getLeft() + page.getRight()) / 2;
    y_position = (page.getTop() + page.getBottom()) / 2;
    content = stamper.getOverContent(i);
    content.saveState();
    content.setGState(gState);
    ColumnText.showTextAligned(content, Element.ALIGN_CENTER,
        content.getText(), x_position, y_position, 0);
    content.restoreState();
}

```

Fig 8 : Ampliació de la funció de marcatge per afegir invisibilitat.

### 3.5 Marca invisible i privada

Fins ara s'ha obtingut una marca invisible però segueix sense ser privada. El fet que no sigui privada implica que qualsevol persona amb capacitats per extreure aquesta marca podria conèixer la informació sensible que pot contenir.

El següent repte és el de la privacitat. Assolir-lo farà possible que aquesta marca, tot i ser recuperada per algú, no pugui revelar informació sensible que pugui afectar a una o algunes persones. Per a privatitzar la marca s'ha fet ús d'alguns paquets de Java que donen suport en temes de seguretat. Aquets paquets són en *java.security* i *javax.crypto*. El que s'ha fet es utilitzar un sistema de xifratge de clau privada i simètric, més específicament el *DES*<sup>11</sup>.

Quan es parla de clau privada, o sistemes de clau única, són sistemes que mantenen la clau en secret i aquesta clau es utilitzada tant per xifrar com per desxifrar. El fet que sigui simètric ens permet, dins del nostre projecte, que tenint la clau privada amb la que em xifrat la marca sigui relativament fàcil desxifrar-la. Aquest és un dels fets per els que s'ha escollit aquest algorisme, la facilitat en la implementació.

El codi de la funció per al xifratge de la marca es pot veure en l'apèndix 3.

El funcionament es el següent: A partir del paràmetre d'entrada *text* es genera un paràmetre de sortida *textResult*. Aquest últim es el paràmetre *text* xifrat. El primer que es pot observar es que es crea un objecte del tipus *Chiper*, que serà l'encarregat de xifrar el text, passant com a paràmetres del constructor el l'algorisme que es vol utilitzar<sup>12</sup>.

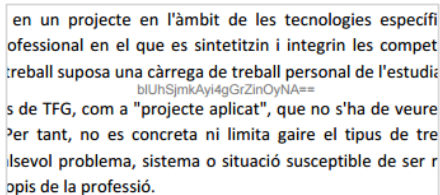


Fig 9 : Exemple de marca xifrada i visible.

Podem observar doncs que si algú es fes amb la marca i no tingués la clau privada no quedaria cap informació al descobert. És cert que existeixen diferents atacs que podrien vulnerar la seguretat de l'algorisme de xifrat, més endavant crearem una línia futura relacionada amb aquest aspecte.

En aquest apartat s'ha parlat de quin ha sigut el procés de marcatge, però com s'ha dit abans aquest procés es crida des de un Servlet. Per a poder veure el funcionament intern complert en forma de gràfic es pot consultar l'apèndix 4.

### 3.6 Recuperació de la marca

Aquest nou punt en el que recuperem la marca està molt relacionat amb el punt anterior, ja que en el punt anterior es fa un xifratge d'aquesta i ara al voler el text en pla l'hem de desxifrar.

Com s'ha dit anteriorment el des xifratge no és la part complicada, fet que es dedueix gràcies a que l'algorisme de xifratge és simètric i tenim la clau amb la que hem xifrat. La part complicada en aquest cas es la recuperació de la marca dins del PDF.

El primer que es va intentar fer va ser recuperar l'estructura del PDF un cop aquest estava marcat. Per a obtenir aquesta estructura es va fer ús de el sistema operatiu, en endavant S.O, *Kali Linux*<sup>13</sup>. Aquest S.O va permetre aconseguir l'objectiu utilitzant una funció interna, *pdf-parser*<sup>14</sup>. La marca va ser trobada però van aparèixer nous reptes difícils de resoldre. En primer cas les estructures venien comprimides i per tant la marca no corresponia al que s'esperava trobar, tot i així es va acabar localitzant.

Vist que era casi impossible extreure informació de l'estructura obtinguda a partir del S.O *Kali Linux* es va decidir buscar per una altra via, i finalment la mateixa llibreria *iText* va permetre la trobada i la possible extracció de la marca.

Per a la trobada de la marca es van seguir els següents passos:

#### Emmagatzemar informació del document:

Com en el cas de la generació de la marca, s'ha de emmagatzemar la informació del document en un objecte Java, i també com en el cas de la generació, s'ha escollit l'objecte *PdfReader* de la llibreria *iText*.

<sup>11</sup> L'algorisme DES (Data Encryption Standard) és un algorisme de xifratge estandaritzat al 1976. Actualment existeix una versió estesa d'aquest, el Triple DES.

<sup>12</sup> El mètode `getSecretKey(String secretKey)` és propi. Retorna la clau privada en format hash MD5.

<sup>13</sup> Kali Linux és una distribució Linux basada en Debian dirigida a testig avançat de penetracions i auditories de seguretat. Kali conté centenars d'eines que ens ajuden a executar diverses tasques relacionades amb la seguretat (<http://docs.kali.org/>)

<sup>14</sup> Accés a les eines de Kali Linux: <http://tools.kali.org/forensics/pdf-parser>.

### Trobar el número d'objectes existents al document:

Aquest objecte té un mètode, que s'anomena `getXrefSize()`, que ens retorna un valor numèric amb el número de objectes que té emmagatzemats la *Cross-reference table*<sup>15</sup>. Aquesta taula és un directori de tots els objectes que conté el document PDF, per tant si fem un recorregut a cadascun d'aquets objectes podem trobar si existeix o no la marca.

### Validació de l'objecte:

Per a cada iteració del bucle recuperem l'objecte en qüestió. Com la explicació d'aquest codi pot ser una mica costosa d'entendre només amb les línies Java, s'ha preparat un pseudocodi, en la Figura 10, amb la intenció de fer més visible el funcionament de la validació dels objectes:

```
per cada objecte:
  comprovar si es un objecte del que puguem treure informac
  si es així:
    Convertir l'objecte a una cadena de caràcters
    Comprovar si conté la marca:
      si es així:
        retornar la marca:
      si no es així:
        següent iteració
```

Fig 10 : Pseudocodi del mètode per recuperar la marca digital.

El codi font d'aquesta funció es pot observar en l'apèndix 5. En aquest codi observem que existeixen dos mètodes propis que són: *ValidateMark* i *decryptMark*. Aquets dos mètodes auxiliars permeten identificar la marca dins de cada objecte.

El primer, *ValidateMark*, fa una validació de la *String* actual. Cal destacar que per la correcta validació de la marca i per a fer un patró que ajudi a aquesta validació, s'han afegit alguns caràcters abans de enviar la marca.

La marca té la següent estructura:

\* *Nom de l'usuari - Marca de temps* \*

El nom d'usuari està format per un *String*, que com ja hem dit amb anterioritat correspon a l'identificador de l'usuari. La segona part, la marca de temps, correspon a la data en format *yyyy/mm/dd hh:mm:ss*.

Aquesta estructura permet trobar de manera ràpida si existeix o no la marca. Per a la obtenció es fa ús de un parell de expressions regulars<sup>16</sup>: La primera per buscar l'objecte de la marca, ja que aquest sempre envolta la marca entre parèntesis. La segona, tenint en compte la estructura esmentada anteriorment, haurà

de buscar un text dins de asteriscs.

Es pot veure un exemple del objecte que conté la marca a continuació:

Marca :

\*Alex-2016/01/24 16:14:45\*

Marca xifrada:

Y+eFVkeDr4LLho0iTdS9gQxDZPB0pHHYzdLg45Ct1As=4

Objecte que conté la marca:

```
Q
q
q
/Xi0 gs
q
BT
1 0 0 1 202.99 421 Tm
/Xi1 8 Tf
(Y+eFVkeDr4LLho0iTdS9gf+gd/aufwo0zdLg45Ct1As=)Tj
ET
Q
Q
Q
```

Finalment aquesta part del projecte va comportar la ampliació de l'espai web. Es va haver de crear una nova pàgina que donat un document en format PDF es pogués trobar, si en té, la marca. El disseny final d'aquesta funcionalitat té l'aspecte de la Figura 11.

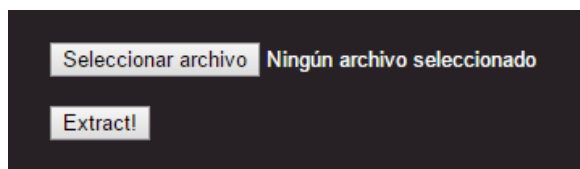


Fig 11 : Uploader del document marcat.

## 4 RESULTATS

Durant l'execució d'aquest projecte s'han assolit la major part dels objectius proposats inicialment.

El plantejament d'aquest projecte va ser, aleshores, una mica abstracte. Les idees relacionades amb l'abast del projecte deixaven una mica lliure el possible resultat d'aquest. Amb això no vull dir que els objectius del projecte no estiguessin clars sinó que en certa mesura es desconeixia si podrien ser resolts. Si s'afegeix la inexperiència amb el marcatge digital, justifiquen el fet de que el projecte tingués un final no previst.

Pel que fa al objectiu de la creació de la marca ha estat totalment completat. No només s'ha aconseguit marcar un document PDF sinó que a més s'ha pogut fer que aquesta marca sigui invisible. Un tema al que m'hagués agradat dedicar-li més temps és a la seguretat.

Moltes vegades en aquest tipus de projectes s'estableixen objectius, i un cop són assolits te'n adones que es podrien assolir alguns altres. La nostra

<sup>15</sup> Aquesta taula és similar a un directori. Conté la ubicació de cada objecte a l'arxiu PDF. En observar les entrades en aquesta taula, l'aplicació de lectura de PDF (per exemple, Adobe Reader), pot localitzar fàcilment un objecte dins de l'arxiu. Això estalvia temps a l'hora de llegir els objectes del PDF. La taula de referències creuades pot tenir una o més seccions. Cadascuna d'aquestes seccions pot tenir una o més subseccions.

<sup>16</sup> Una expressió regular és una cadena de text especial per a descriure un patró de recerca.



marca podríem dir que es mitjanament segura ja que ara mateix esta sent presentada com una marca invisible, privada i robusta, però cal a dir que no s'ha pogut testear en profunditat fins a quin punt compleix aquestes tres característiques. Per això li dedicaré una línia futura a les dues característiques que no hem pogut explorar en profunditat, la robustesa i la privacitat.

M'ajudaré de la metodologia utilitzada, aprofitant que ens permet recuperar cada instant del desenvolupament, per a fer un repàs de quin grau d'assoliment ha tingut el projecte i així poder tenir totes les idees ben ordenades. La primera part, i per tant primers objectius que es proposaven, era la creació de la interfície web. Aquesta ha estat creada tenint en compte que era un complement i no l'objectiu principal. S'ha intentat que sigui el més senzill possible però a la vegada el més útil. Les següents històries d'usuari estaven més relacionades amb aspectes de documentació, tot i així no trobo que s'hagin de passar per alt, ja que tot i que potser no reflexa un resultat tangible, el resultat intangible que ha causat en la meua persona ha estat significatiu.

Per últim, i la part principal del projecte que és el marcatge, s'ha assolit amb èxit. Un punt d'aquestes històries, que ha estat mencionat en algunes parts de aquest article, però que no s'ha arribat a fer una referència a la seva execució és el de el rendiment del marcatge per a documents de diferent dimensió. Aquesta referència s'ha decidit deixar-la per aquest apartat ja que no ha estat un desenvolupament, sinó que ha estat un resultat d'aquest. Per a les proves de rendiment s'han utilitzat vint fitxers de diferent dimensió. Aquets dimensions han anat des de 1 pàgina fins a 1094 pàgines.

Els resultats d'aquestes proves estan descrits de forma gràfica en la Figura 12. Com es pot observar en el gràfic de la Figura 12, el procés de marcatge afecta al temps de descarrega, però és suficient aquesta diferència com per que pugui ser significativa?

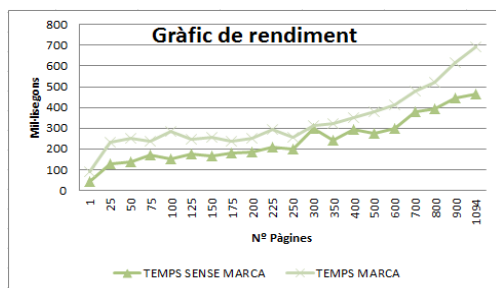


Fig 12 : Gràfic del rendiment en la descarrega.

Les proves han estat executades un parell de cops, fent després la mitja posterior entre els dos resultats per a intentar mantenir la major exactitud en les proves. La diferència entre els documents marcats i els no marcats no la podem catalogar de excessiva. Podem dir llavors que tenint en compte els resultats obtinguts i mostrats en la Figura 12, no existeix un cost

significatiu en el procés de descarrega quan s'ha de marcar un document.

Per últim en l'apartat de resultat, vull esmentar la fita relacionada amb la recuperació de la marca. S'ha aconseguit desenvolupar una eina afegida que ens permet, a partir d'una interfície web, fer un *upload* d'un document marcat i aconseguir si aquest document esta marcat o no, i en el cas que estigui marcat ens proporciona les dades de la marca, nom, data i hora. Per tant, es pot dir, amb més o menys seguretat, que el projecte ha complert amb els resultats proposats al inici.

## 5 CONCLUSIÓ

El marcatge de documents, en aquest cas de documents PDF, pot tenir una varietat molt amplia de beneficis. En el cas d'aquest projecte esta estrictament orientada a el seguiment d'aquest document un cop surt del nostre domini, però poden existir casos que no persegueixin el seguiment sinó que vulguin gaudir d'una certa integritat d'un producte com per exemple una data de caducitat, o en el cas de un producte que venen com original poder corroborar aquesta característica, etc.

En aquest treball s'ha volgut aconseguir varis objectius, entre el que esta el principal que es el del marcatge, però existeixen molts que no podrien ser mostrats en aquest document sense fer referència a la pròpia persona.

En conclusió el projecte proporciona una eina que gestionada de manera correcta pot ajudar a garantir els drets que es buscaven protegir.

Pel que fa el tema en general, i allunyant-nos una mica e la part més tècnica, més informàtica, aquest treball m'ha fet donar-me compte que els aspectes relacionats els drets d'autor i la legalitat amb la que tractem certs documents estan ben protegits. Que existeixen organismes, com per exemple CEDRO<sup>17</sup>, que centren els seus esforços en aturar aquest mal ús de la xarxa i que penalitzen a qui no compleixen els drets. Això ho podem veure en forma de noticia[16,17] i no fa falta anar-se gaire lluny, la mateixa Universitat Autònoma ha estat sancionada per la publicació de certs documents en el CV al que accedien els alumnes. Per tant es pot veure que aquest tema no és un tema que ens agafi de nou sinó que realment es porta temps treballant en la defensa dels drets relacionats amb certs documents.

Finalment, diré que aquest tema és un tema delicat, ja que hem de tenir molt clar on està el límit, que moltes vegades és difús. Per tant, mesures com les que nosaltres intentem dur a terme són mesures que poden

<sup>17</sup> Ells mateixos es defineixen com : "Somos el Centro Español de Derechos Reprográficos (CEDRO), la asociación sin ánimo de lucro de autores y editores de libros, revistas y otras publicaciones, editadas en cualquier medio y soporte, que se encarga de defender y gestionar de forma colectiva sus derechos de propiedad intelectual de tipo patrimonial (reproducción, transformación, comunicación pública y distribución)." - [www.cedro.org](http://www.cedro.org)

ajudar a que aquest drets es segueixin defensant i que les persones siguin una mica més conscients dels drets que els envolten, tan seus com dels altres.

## 5.1 Línies Futures

En aquest apartat faré referència a les línies futures d'aquest projecte.

Trobo necessari plantejar línies futures ja que per molt ben definit que tinguem l'abast del nostre projecte sempre poden sorgir nous objectius al tancament d'aquest i no disposar de temps extra per implementar-los. Per tant dedicaré un apartat a cadascuna d'aquestes parts que han sorgit però que no s'han pogut implementar.

### 5.1.1 Seguretat

Els objectius del apartat de la seguretat, com he dit en altres apartats, es resumeixen en la privacitat i la robustesa, ja que la visibilitat es una característica que es pot observar fàcilment i la seva valoració és clara, si o no. Però quan parlem de robustesa o privacitat, com sabem si estem del tot segurs o el nostre producte es del tot robust? En la meua opinió no ho sabem, sempre haurem d'innovar per tal de que aquets aspectes es compleixin.

### 5.1.2 Privacitat

Quan es van decidir els objectius del projecte va sorgir el tema de la privacitat dins de la seguretat. Es va parlar que seria un apartat a tenir en compte un cop la part principal del projecte estigués acabada. Finalment se li va poder dedicar temps i es va xifrar aquesta marca. Tot i així, des de el meu punt de vista, aquest xifratge pot ser millorat, ja que l'algorisme utilitzat pot ser modificat per un de més segur. La meua proposta en aquest apartat de les línies futures es la implementació de l'algorisme AES en la funció de xifratge.

### 5.1.3 Robustesa

Entrem en un altre subapartat de la seguretat, i també difícil de mesurar a simple vista. En aquest cas més que millorar el que voldria és comprovar realment que la marca és robusta, que no es pugui aconseguir eliminar-la i obtenir un document sense marcar. Per tant la feina en aquest apartat és descobrir maneres de posar en perill la marca i veure si aquesta esta preparada per no ser vulnerada. Es podria començar fent un anàlisi del resultat de fer el *parseig* de el document mitjançant el S.O Kali Linux com el de l'apèndix 6.

### 5.1.4 Entorn Web

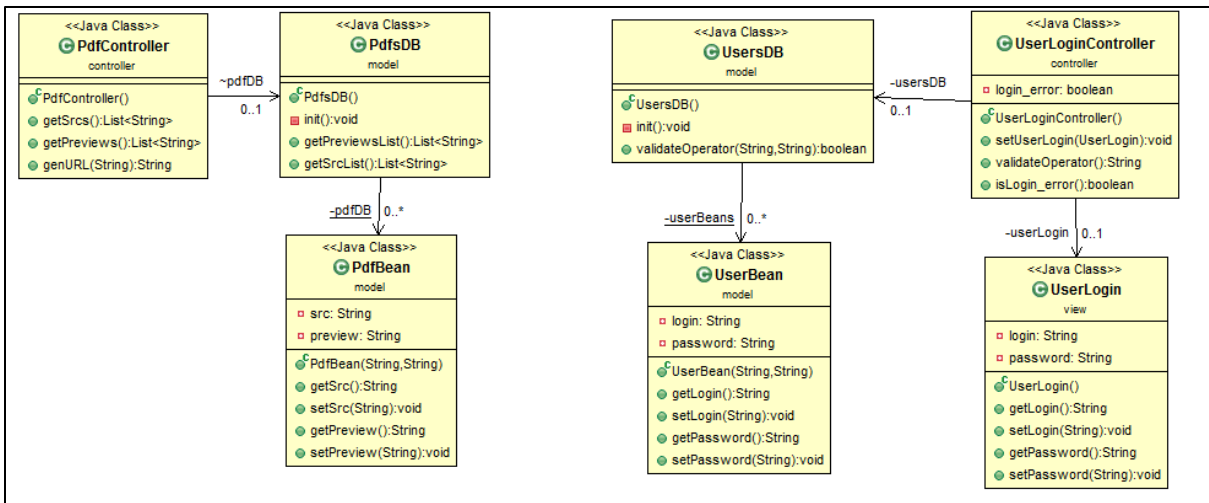
Una altra línia futura podria ser la implementació de l'eina a major escala, és a dir, proporcionar-li connexió a una base de dades en la qual es guardin els documents, els usuaris, tipus de marques, etc. I per tant facilitar la seva implementació en la xarxa permetent als usuaris poder fer un seguiment dels seus docu-

ments un cop els facin públics a la xarxa.

## 6 BIBLIOGRAFIA

- [1] Llei de Propietat Intel·lectual, consultada el 17 de Octubre de 2015: <https://www.boe.es/buscar/act.php?id=BOE-A-1996-8930>
- [2] Digital Watermarking Alliance, consultada el 24 de Setembre de 2015: <http://www.digitalwatermarkingalliance.org/applications.asp>
- [3] Pàgina oficial del IDE Eclipse, consultada el 26 de Gener de 2016: <https://eclipse.org/>
- [4] Pàgina oficial de la llibreria iText, consultada el 24 de Setembre de 2015: <http://itextpdf.com/>
- [5] The deterrent effect of digital serial number on illegal file-sharing and downloading, consultada el 1 de Octubre de 2015: [http://www.digitalwatermarkingalliance.org/docs/papers/DWA\\_WhitePaper\\_PiracyDeterrence.pdf](http://www.digitalwatermarkingalliance.org/docs/papers/DWA_WhitePaper_PiracyDeterrence.pdf)
- [6] Producte The Guardian propietat de la empresa Digimark, consultada el 27 de Setembre de 2015: [www.digimarc.com/products/guardian](http://www.digimarc.com/products/guardian)
- [7] Scrum Methodology, consultada el 25 de Octubre de 2015: <http://scrummethodology.com/>
- [8] Informació sobre Scrum Wikipedia, consultada el 15 de Gener de 2016: <https://es.wikipedia.org/wiki/Scrum>
- [9] Blog Nebaris, "Cronicas ágiles de un desarrollador en solitario", consultada el 25 de Octubre de 2015: <http://www.nebaris.com/post/112/cronicas-agiles-de-un-desarrollador-solitario>
- [10] Pàgina oficial de la llibreria Qoppa software, consultada el 10 de Novembre de 2015: <https://www.qoppa.com/>
- [11] Pàgina oficial de la llibreria pdfBox, consultada el 10 de Novembre de 2015: <https://pdfbox.apache.org/>
- [12] Pàgina oficial de la llibreria PDF clown, consultada el 10 de Novembre de 2015: <http://pdfclown.org/>
- [13] Pàgina oficial de la llibreria Pdf-rendered, consultada el 10 de Novembre de 2015: <https://java.net/projects/pdf-renderer/>
- [14] Pàgina oficial de la llibreria jPod, consultada el 10 de Novembre de 2015: <http://sourceforge.net/projects/jpodlib/>
- [15] Pàgina oficial de la llibreria Pdfjet, consultada el 10 de Novembre de 2015: <http://pdfjet.com/os/edition.html>
- [16] Diari "el País", noticia del 4 de Novembre de 2014. "Condena a la UAB por violar los derechos de autor de su web", consultada el 25 d'Octubre de 2015: [http://ccaa.elpais.com/ccaa/2014/11/04/catalunya/1415100616\\_712247.html](http://ccaa.elpais.com/ccaa/2014/11/04/catalunya/1415100616_712247.html)
- [17] Diari "el periodico", noticia del 4 de Novembre de 2014. "La UAB deberá abonar 3 millones euros por vulnerar derechos de propiedad intelectual", consultada el 25 d'Octubre de 2015: <http://www.elperiodico.com/es/noticias/sociedad/uab-debera-abonar-millones-euros-por-vulnerar-derechos-propiedad-intelectual-3660394>

## 7 APÈNDIX



Apèndix 1: Diagrama de classes del funcionament bàsic del CV.

```

try {
    PdfReader reader = new PdfReader(src);
    int numberOfPages = reader.getNumberOfPages();

    PdfStamper stamper = new PdfStamper(reader, new FileOutputStream(dest));

    Font f = new Font(FontFamily.HELVETICA, 8);
    Phrase p = new Phrase("some text", f);

    for (int i = 1; i <= numberOfPages; i++) {
        page = reader.getPageSizeWithRotation(i);
        x_position = (page.getLeft() + page.getRight()) / 2;
        y_position = (page.getTop() + page.getBottom()) / 2;
        content = stamper.getOverContent(i);
        content.saveState();
        ColumnText.showTextAligned(content, Element.ALIGN_CENTER, p, x_position, y_position, 0);
        content.restoreState();
    }
    stamper.close();
    reader.close();
    return true;
} catch (Exception e) {
    e.printStackTrace();
    return false;
}

```

Apèndix 2: Funció de marcatge visible i pública

```

public String encryptMark(String text) {

    String secretKey = Ctes.SECRET_KEY;
    String textResult = "";

    try {

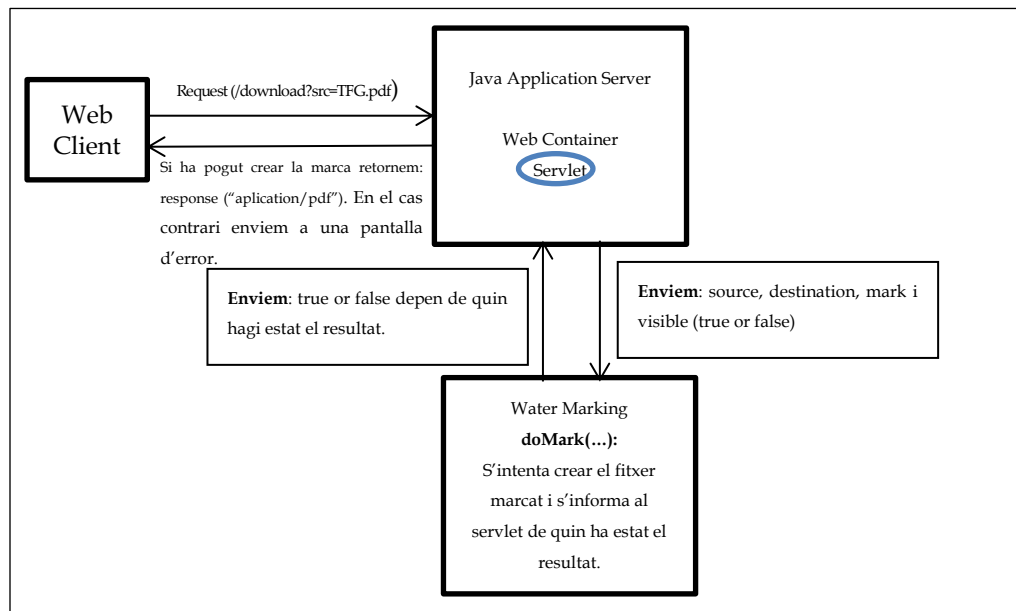
        Cipher cipher = Cipher.getInstance(Ctes.DES);
        cipher.init(Cipher.ENCRYPT_MODE, getSecretKey(secretKey));

        byte[] plainTextBytes = text.getBytes("utf-8");
        byte[] buf = cipher.doFinal(plainTextBytes);
        byte[] base64BytesTextResult = Base64.encodeBase64(buf);
        textResult = new String(base64BytesTextResult);

    } catch (Exception ex) {
        ex.printStackTrace();
    }
    return textResult;
}

```

Apèndix 3: Funció de xifratge



Apèndix 4: Gràfic del funcionament intern de la marca - Servlet i Marcatge.

```

for (int i = 1; i <= reader.getXrefSize(); i++) {
    obj = reader.getPdfObject(i);
    if (obj != null && obj.isStream()) {
        PRStream stream = (PRStream) obj;
        byte[] byteStream;
        try {
            byteStream = PdfReader.getStreamBytes(stream);
        } catch (UnsupportedPdfException e) {
            byteStream = PdfReader.getStreamBytesRaw(stream);
        }
        output = new String(byteStream);
        try {
            Pattern pattern = Pattern.compile("\\\\(([^\\]+)\\\\)");
            Matcher matcher = pattern.matcher(output);
            if (matcher.find()) {
                if (ValidateMark(decryptMark(matcher.group(0)))) {
                    return decryptMark(matcher.group(0));
                }
            }
        } catch (Exception e) {
            e.printStackTrace();
        }
    }
}
reader.close();

return marcaNotFound;

```

Apèndix 5: Codi de la funció encarregada de la recuperació de la marca.

```

obj 4 0
Type:
Referencing:
Contains stream

<<
  /Filter /FlateDecode
  /Length 10
>>

'x\x9c+\xe4\x02\x00\x00\xee\x00|\n'

obj 5 0
Type:
Referencing:
Contains stream

<<
  /Filter /FlateDecode
  /Length 94
>>

'x\x9cS\x08\xe4*\x04B\xfd\x88L#\x85\xf4b\xcb)\x84\xcbP\xc1\x00\x08\r\x15\x8c\x8c\x0c\xf5\x0c\x0c\x15L\x8c\x0c\x15Br
AJ\x8c\x15,\x14B\xd2\xb84\x1csR+\x0c\xac\x90\xa1axbIjQnbQvf^\xba~J~y^N~b\x8afH\x16\x97k\x08W
\x18\x02\x008\xf5\x1a\xfb\n'

```

Apèndix 6: Estructura comprimida del PDF